CPU 유니트

OMRON

SYSMAC CS 시리즈

CS1□-CPU□□□-□□

SYSMAC CJ 시리즈

CJ2H-CPU6□-EIP

CJ2H-CPU6□

CJ2M-CPU□□

CJ1□-CPU□□□-□

SYSMAC One NSJ 시리즈

NSJ□□-□□□□□-□□□

명령 참조 설명서

Man. No. SBCA-351G-K6

부탁의 말

(1) 본 설명서 내용의 일부 또는 전체를 무단으로 복사, 복제, 전재하는 것을 금지합니다.

(2) 본 설명서의 내용은 개선을 위해 예고 없이 사양 등이 변경되는 경우가 있으므로 미리 양지해 주시길 바랍니다.

(3) 본 설명서 내용에 만전을 기하였으나 의문이 가는 점이나 잘못된 점을 발견하면 번거로우시겠지만 설명서 말미에 기재된 당사 지점 또는 영업소로 문의해 주십시오.
문의하실 때는 설명서 말미에 기재된 설명서 번호도 함께 알려주십시오.

저작권 및 상표

- Windows는 마이크로소프트사의 등록상표입니다.
- 그 밖에 본문 중에 게재되어 있는 시스템명 및 제품명은 각 회사의 상표 또는 등록상표입니다.

SYSMAC CS 시리즈

CS1□-CPU□□□-□□

SYSMAC CJ 시리즈

CJ2H-CPU6□-EIP
CJ2H-CPU6□
CJ2M-CPU□□
CJ1□-CPU□□□-□

SYSMAC One NSJ 시리즈

NSJ□□-□□□□□-□□□

명령 참조 설명서

머리말

CS/CJ/NSJ 시리즈 PLC를 구입해 주셔서 감사합니다.

CS/CJ/NSJ 시리즈 PLC는 당사가 지닌 고도의 제어 기술과 풍부한 경험을 바탕으로 개발된 소형의 고기능 PLC입니다.

대상 시리즈와 형식의 정의

본 설명서에서 설명하는 대상 시리즈와 형식은 다음과 같습니다.

시리즈명	이름	형식
CJ 시리즈	CJ2H CPU 유니트	CJ2H-CPU6□-EIP CJ2H-CPU6□
	CJ2M CPU 유니트	CJ2M-CPU□□
	CJ1-H CPU 유니트	CJ1H-CPU□□H-R
		CJ1□-CPU□□H
		CJ1□-CPU□□P
	CJ1M CPU 유니트	CJ1M-CPU□□
	CJ1 CPU 유니트	CJ1□-CPU□□
CS 시리즈	CS1-H CPU 유니트	CS1G/H-CPU□□H
	CS1 CPU 유니트	CS1G/H-CPU□□(-V1)
	CS1D CPU 유니트	CS1D-CPU□□H
		CS1D-CPU□□S
		CS1D-CPU□□P
NSJ 시리즈	NSJ 유니트	NSJ5-TQ□□(B)-G5D
		NSJ5-SQ□□(B)-G5D
		NSJ8-TV□□(B)-G5D
		NSJ10-TV□□(B)-G5D
		NSJ12-TS□□(B)-G5D
		NSJ5-TQ□□(B)-M3D
		NSJ5-SQ□□(B)-M3D
		NSJ8-TV□□(B)-M3D

'**PLC**' 표기에 대해

이 설명서에서는 '프로그래머블 컨트롤러'의 약칭을 PC와 구별하기 위해 'PLC'라고 합니다. 단, 기존의 기능 이름 또는 소프트웨어의 메뉴 이름으로 'PC'를 사용하는 경우에는 제한적으로 'PC'를 사용하는 경우가 있습니다.
또한 개인용 컴퓨터(Personal Computer)는 약칭을 사용하지 않고 '컴퓨터'로 표시합니다.

대상 독자

이 설명서는 다음 사용자를 대상으로 한 것입니다.
전기 지식(전기 공사 또는 이와 동등한 지식)이 있는
- FA 기기 도입 담당자
- FA 시스템 설계자
- FA 현장 관리자

당부의 말

이 설명서는 CS/CJ/NSJ 시리즈 PLC를 사용하는 데 필요한 정보를 소개합니다. 이 설명서를 잘 읽고 충분히 숙지한 후 사용하십시오. 또한 읽은 후에도 이 설명서를 가까운 곳에 잘 보관해 두고 사용하십시오.

사용 시 동의 사항

1. 보증 내용

① 보증 기간
당사 제품의 보증 기간은 구입 후 또는 지정 장소에서 납품 후 1년입니다.

② 보증 범위
위 보증 기간 중에 당사의 책임으로 당사 제품에 고장이 발생한 경우에는 대체품을 제공하거나 제품 구입처에서 무상으로 수리해 드립니다.
단, 고장의 원인이 다음 항목에 해당하는 경우에는 이 보증 대상 범위에서 제외됩니다.

a) 카탈로그 또는 사용 설명서 등에 기재되어 있는 이외의 조건, 환경, 취급 및 사용에 의한 경우
b) 당사 제품 이외의 원인인 경우
c) 당사 직원 이외의 사람에 의한 개조 또는 수리에 의한 경우
d) 당사 제품 본래의 사용 방법 이외의 방법으로 사용한 경우
e) 제품 출하 당시 당사의 과학 기술 수준으로는 예견할 수 없었던 문제인 경우
f) 그 외 천재지변, 재해 등 당사의 책임이 아닌 원인에 의한 경우

또한 여기서의 보증은 당사 제품에 대한 보증만을 의미하는 것으로, 당사 제품의 고장에 의해 유발되는 손해는 보증 대상에서 제외됩니다.

2. 책임의 제한

① 당사 제품에 기인하여 발생한 특별 손해, 간접 손해 또는 소극적 손해에 관해서는 어떤 경우라도 책임을 지지 않습니다.

② 프로그래밍 가능한 당사 제품에 대해서는 당사 직원 이외의 사람이 실시한 프로그램 또는 그로 인해 발생한 결과에 대해 당사는 책임을 지지 않습니다.

3. 적합 용도의 조건

① 당사 제품을 다른 제품과 조합하여 사용하는 경우, 고객이 적합한 규격이나 법규 또는 규제를 확인하십시오. 또한 고객이 사용하는 시스템, 기계, 장치에 대한 당사 제품의 적합성은 고객이 직접 확인해야 합니다. 이러한 내용이 이루어지지 않은 경우 당사는 당사 제품의 적합성에 대해 책임을 지지 않습니다.

② 다음 용도로 사용되는 경우, 당사 영업 담당자에게 상담한 후 사양서 등에서 확인하는 한편, 정격 및 성능에 대해 여유가 있는 사용 방법이나 혹시 고장이 일어나도 위험을 최소화하는 안전 회로 등의 안전 대책을 마련하십시오.

a) 옥외의 용도, 잠재적인 화학적 오염 또는 전기적인 방해를 받는 용도 또는 카탈로그 및 사용 설명서에 기재되지 않은 조건이나 환경에서의 사용

b) 원자력 제어 설비, 소각 설비, 철도· 항공· 차량 설비, 의료용 기계, 오락 기계, 안전 장치 및 행정 기관이나 개별 업계의 규제에 따르는 설비

c) 인명이나 재산에 위험이 미치는 시스템, 기계, 장치

d) 가스, 수도, 전기 공급 시스템이나 24시간 연속 운전 시스템 등 높은 신뢰성이 요구되는 설비

e) 그 외 위의 a)~d)에 준하는 고도의 안전성이 필요한 용도

③ 고객이 당사 제품을 인명이나 재산에 중대한 영향을 미치는 용도로 사용하는 경우에는 시스템 전체에서 위험을 알리거나 여유 있는 설계로 안전성을 확보해야 하며, 당사 제품이 전체적으로 의도한 용도에 적합하게 배선 및 설치되었는지 반드시 사전에 확인하십시오.

④ 카탈로그 등에 기재된 활용 사례는 참고용이므로 이를 적용하고자 할 때에는 기기나 장치의 기능이나 안전성을 확인한 후 사용하십시오.

⑤ 당사 제품이 올바르게 사용되지 않아 고객 또는 제3자에게 예측하지 못한 손해가 발생하지 않도록 사용상의 금지 사항 및 주의 사항을 모두 숙지하고 준수하십시오.

4. 사양 변경

카탈로그와 사용 설명서에 기재된 제품의 사양 및 부속품은 개선 또는 그 밖의 이유로 필요에 따라 변경되는 경우가 있습니다. 당사 영업 담당자에게 상담하여 당사 제품의 실제 사양을 확인하시기 바랍니다.

5. 서비스의 범위

본 제품의 가격에는 기술자 파견 등의 서비스 비용은 포함되어 있지 않습니다.
고객이 원하는 사항이 있으면 당사 영업 담당자에게 상담하시기 바랍니다.

6. 적용 범위

이상의 내용은 일본 국내에서의 거래 및 사용을 전제로 합니다.

해외에서 사용하는 경우

본 제품 중 외환 및 해외 무역법의 규정에 따라 수출허가, 승인 대상 화물(또는 기술)에 해당하는 것을 수출(또는 비거주자에게 제공)하는 경우, 동법에 규정된 수출 허가, 승인(또는 직무 거래 허가)이 필요합니다.

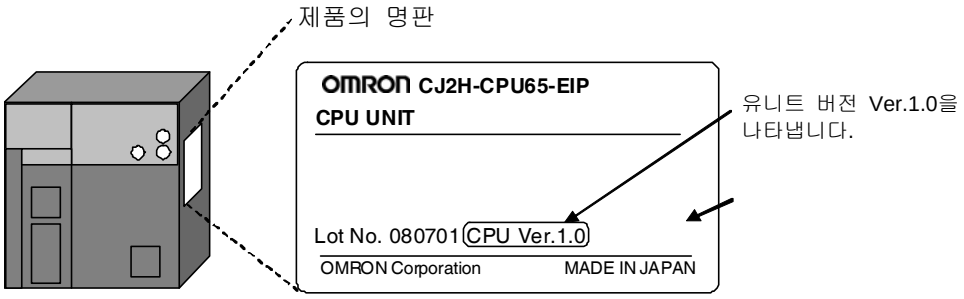
CS/CJ 시리즈의 '유니트 버전'

유니트 버전

CS/CJ/NSJ 시리즈는 버전 업그레이드 등에 따른 CPU 유니트의 탑재 성능 차이를 관리하기 위해 '유니트 버전'이라는 개념을 도입하였습니다.
명령어에 따라 특정 유니트 버전 이상에서만 사용할 수 있는 경우가 있으므로 주의하십시오.

1) 제품에서 표기 방법

유니트 버전을 통한 관리 대상 제품에는 유니트 버전이 'Ver.□.□' 형식으로 명판 위 Lot No. 오른쪽에 기재되어 있습니다.



2) 지원 소프트웨어에 의한 확인 방법

CX-Programmer Ver.4.0 이상에서는 다음의 몇 가지 방법으로 유니트 버전을 확인할 수 있습니다.

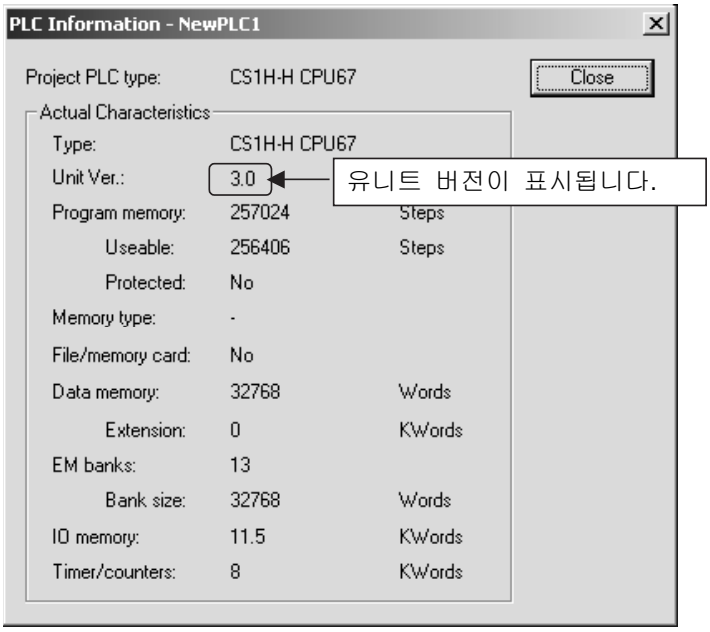
a) [PLC 정보]에서 확인하는 방법
b) [유니트 제조 정보]로 확인하는 방법(고기능 I/O 유니트, CPU 고기능 유니트도 확인 가능)

a) [PLC 정보]에서 확인하는 방법

1 ・PLC 기종 및 CPU 형식을 알고 있는 경우:
[PLC 변경] 대화 상자에서 선택하여 온라인으로 연결한 후 [PLC] 메뉴에서 [편집]－[정보]를 선택합니다.

・PLC 기종 및 CPU 형식이 정확하지 않은 경우
(단, 시리얼 연결로 직접 연결만 가능):
[PLC] 메뉴에서 [자동 온라인]을 선택하여 온라인으로 연결한 후 [PLC] 메뉴에서 [편집]－[정보]를 선택합니다.

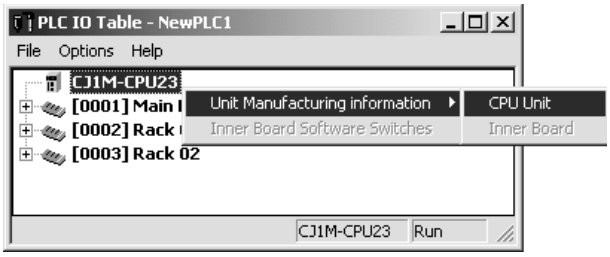
2 다음과 같이 [PLC 정보] 대화 상자가 표시됩니다.



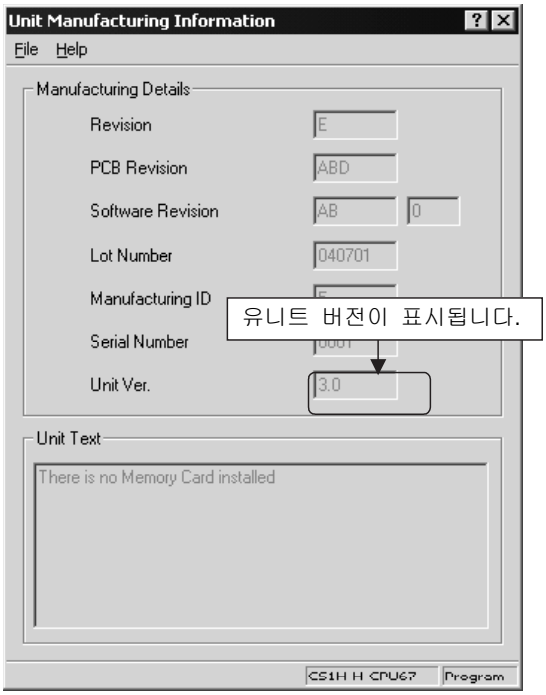
온라인으로 연결된 CPU 유니트의 유니트 버전을 확인하십시오.

b) [유니트 제조 정보]에서 확인하는 방법

1 I/O 테이블 창에서 CPU 유니트를 마우스 오른쪽 버튼으로 클릭하고 [유니트 제조 정보] | [CPU 유니트]를 선택합니다.



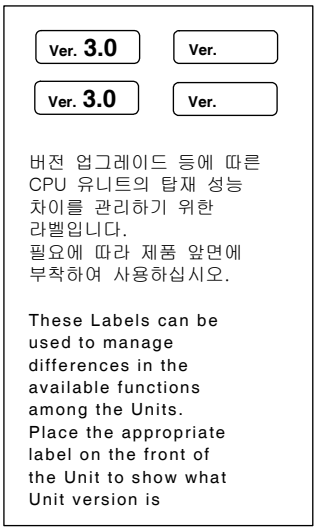
2 다음과 같은 [유니트 제조 정보] 대화 상자가 표시됩니다.



온라인으로 연결된 CPU 유니트의 유니트 버전을 확인하십시오.

3) 유니트 버전 라벨에 의한 식별

유니트 버전 라벨(아래 그림)이 제품에 첨부되어 있습니다.



유니트 버전을 이전의 CPU 유니트와 구별하고 싶을 때는 이 라벨을 CPU 유니트 앞면에 부착하십시오.

관련 설명서

CS/CJ/NSJ 시리즈의 PLC 본체 관련 설명서는 다음 표와 같이 구성되어 있으므로 참조하시기 바랍니다.

설명서 번호	형식	설명서 명칭	용도	내용
SBCA-351	CJ2H-CPU6□-EIP CJ2H-CPU6□ CJ2M-CPU□□ CS1G/H-CPU□□H CS1G/H-CPU□□-V1 CS1D-CPU□□H CS1D-CPU□□S CJ1H-CPU□□H-R CJ1G/H-CPU□□H CJ1G-CPU□□P CJ1M-CPU□□ CJ1G-CPU□□ NSJ□-□□□□(B)-G5D NSJ□-□□□□(B)-M3D	CS/CJ/NSJ 시리즈 명령 참조 설명서	명령어에 대해 자세히 알고 싶은 경우	각 명령어에 대한 자세한 내용을 설명합니다. 프로그래밍할 때 각 CPU 유니트의 사용 설명서와 함께 사용하십시오.
SBCA-349	CJ2H-CPU6□-EIP CJ2H-CPU6□ CJ2M-CPU□□	CJ 시리즈 CJ2 CPU 유니트 사용 설명서 하드웨어편	CJ2 CPU 유니트의 개요/설계/부착/보수 등의 기본적인 사양에 대해 알고 싶은 경우. 주로 하드웨어 관련 정보	CJ2 CPU 유니트의 PLC 본체에 대해 다음 내용을 설명합니다. ・개요 ・하드웨어의 사양, 설정 ・부착과 배선 ・문제 해결 사용 설명서 소프트웨어편(SBCA-350)과 함께 사용하십시오.
SBCA-350	CJ2H-CPU6□-EIP CJ2H-CPU6□ CJ2M-CPU□□	CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편	CJ2 CPU 유니트의 프로그래밍/시스템의 기동에 대해 알고 싶은 경우. 주로 소프트웨어 관련 정보	CJ2 CPU 유니트의 PLC 본체에 대해 다음 내용을 설명합니다. ・CPU 유니트의 동작 개요, 프로그래밍 지식 ・소프트웨어 설정, CPU 내장 기능 사용 설명서 하드웨어편(SBCA-349)과 함께 사용하십시오.
SBCA-313	CS1G/H-CPU□□H CS1G/H-CPU□□-V1 CS1D-CPU□□H CS1D-CPU□□S CJ1H-CPU□□H-R CJ1G/H-CPU□□H CJ1G-CPU□□P CJ1M-CPU□□ CJ1G-CPU□□ NSJ□-□□□□(B)-G5D NSJ□-□□□□(B)-M3D	CS/CJ/NSJ 시리즈 사용 설명서 프로그래밍편	CS/CJ/NSJ 시리즈의 각종 기능에 대해 알고 싶은 경우	CS/CJ/NSJ 시리즈의 PLC 본체에 대해 다음 내용을 설명합니다. ・프로그래밍 방법 ・태스크 기능 ・파일 메모리 기능 ・기타 각종 기능 사용 설명서 설치편(CS 시리즈: SBCA-301, CJ 시리즈: SBCA-312)과 함께 사용하십시오.

설명서 번호	형식	설명서 명칭	용도	내용
SBCA-301	CS1G/H-CPU□□H CS1G/H-CPU□□-V1	CS 시리즈 사용 설명서 설치편	CS 시리즈의 개요/설계/부착/보수 등의 기본적인 사양에 대해 알고 싶은 경우	CS 시리즈의 PLC 본체에 대해 다음 내용을 설명합니다. ・ 개요/특징 ・ 시스템 구성 설계 ・ 부착/배선 ・ I/O 메모리 할당 ・ 문제 발생 시 대처 방법 사용 설명서 프로그래밍편(SBCA-313)과 함께 사용하십시오.
SBCA-312	CJ1H-CPU□□H-R CJ1G/H-CPU□□H CJ1G-CPU□□P CJ1M-CPU□□ CJ1G-CPU□□	CJ 시리즈 사용 설명서 설치편	CJ 시리즈의 개요/설계/부착/보수 등의 기본적인 사양에 대해 알고 싶은 경우	CJ 시리즈의 PLC 본체에 대해 다음 내용을 설명합니다. ・ 개요/특징 ・ 시스템 구성 설계 ・ 부착/배선 ・ I/O 메모리 할당 ・ 문제 발생 시 대처 방법 사용 설명서 프로그래밍편(SBCA-313)과 함께 사용하십시오.
SBCA-315	CJ1M-CPU□□	CJ 시리즈 사용 설명서 내장 입출력 기능편	CJ1M CPU 유니트의 내장 입출력 기능에 대해 알고 싶은 경우	CJ1M CPU 유니트의 내장 입출력 기능에 대해 설명합니다.
SBCA-318	CS1D-CPU□□H CS1D-CPU□□S CS1D-DPL01 CS1D-PA/PD□□□	CS1D 듀플렉스 시스템 사용 설명서 설치편	CS1D 듀플렉스 시스템의 개요/설계/부착/보수 등의 기본적인 사양 및 이중화에 대해 알고 싶은 경우	CS1D 듀플렉스 시스템에 대해 다음 내용을 설명합니다. ・ 개요/특징 ・ 시스템 구성 설계 ・ 이중화 시스템 설계 ・ 부착/배선 ・ I/O 메모리 할당 ・ 문제 발생 시 대처 방법 또한, 유니트의 온라인 교환에 대한 프로그래밍 콘솔 조작 방법을 설명합니다. 사용 설명서 프로그래밍편(SBCA-313)과 함께 사용하십시오.
SBCA-304	CJ2H-CPU6□-EIP CJ2H-CPU6□ CJ2M-CPU□□ CS1G/H-CPU□□H CS1G/H-CPU□□-V1 CS1D-CPU□□H CS1D-CPU□□S CS1W-SCU□□-V1 CS1W-SCB□□-V1 CJ1H-CPU□□H-R CJ1G/H-CPU□□H CJ1G-CPU□□P CJ1M-CPU□□ CJ1G-CPU□□ CJ1W-SCU□□-V1 CP1H-X□□□□-□ CP1H-XA□□□□-□ CP1H-Y□□□□-□ CP1L-M/L□□□-□ CP1E-E□□D□-□ CP1E-N□□D□-□ NSJ□-□□□□(B)-G5D NSJ□-□□□□(B)-M3D	CS/CJ/CP/NSJ 시리즈 통신 명령 참조 설명서	CS/CJ/CP 시리즈 CPU 유니트, NSJ 시리즈의 통신 명령에 대해 자세히 알고 싶은 경우	1) C 모드 명령 및 2) FINS 명령에 대해 자세히 설명합니다. CPU 유니트의 통신 명령(C 모드 명령 또는 FINS 명령)에 대한 자세한 내용을 알고 싶을 때 참조하십시오. 주: 이 설명서에 기재된 통신 명령은 CPU 유니트의 통신 명령이며, 통신 경로는 관계없습니다. (CPU 유니트의 시리얼 통신 포트, 시리얼 커뮤니케이션 보드/유니트의 통신 포트, 통신 유니트 경유 등이 가능합니다.) 또한 고기능 I/O 유니트 또는 CPU 고기능 유니트 명령에 관해서는 각 유니트의 사용 설명서를 참조하십시오.

설명서 번호	형식	설명서 명칭	용도	내용
SBCA-342	NSJ5-TQ□□(B)-G5D/M3D NSJ5-SQ□□(B)-G5D/M3D NSJ8-TV□□(B)-G5D/M3D NSJ10-TV□□(B)-G5D NSJ12-TS□□(B)-G5D	NSJ 시리즈 NSJ 컨트롤러 사용 설명서 설치편	NSJ 시리즈의 개요/설계/부착/보수 등의 기본적인 사양에 대해 알고 싶은 경우	NSJ 시리즈에 대해 다음 내용을 설명합니다. ・ 개요/특징 ・ 시스템 구성 설계 ・ 부착/배선 ・ I/O 메모리 할당 ・ 문제 발생 시 대처 방법 CS/CJ 시리즈 사용 설명서 설치편(SBCA-312)/프로그래밍편(SBCA-313)/NS-V1/-V2 시리즈 설치 설명서(SBSA-517)와 함께 사용하십시오.
SBCA-337	WS02-CX□□-V□	CX-Programmer 조작 설명서	Windows PC용 프로그래밍 툴 CX-Programmer의 조작 방법에 대해 알고 싶은 경우	CX-Programmer의 조작 방법을 설명합니다. 프로그래밍할 때 각 CPU 유니트의 사용 설명서와 함께 사용하십시오.
SBCA-338	WS02-CX□□-V□	CX-Programmer CS/CJ/CP 시리즈 조작 설명서 펑션 블록편/구조적 텍스트편	펑션 블록 기능 및 ST 프로그래밍 기능을 사용하는 경우, 그 사양과 조작 방법에 대해 알고 싶을 때	펑션 블록 기능 및 ST 프로그래밍 기능의 사양과 조작 방법을 설명합니다. 펑션 블록 기능 및 ST 프로그래밍 기능을 사용할 때만 참조합니다.
SBCA-348	WS02-CX□□-V□	CX-Programmer 조작 설명서 SFC편	SFC 프로그래밍 기능을 사용하는 경우, 그 사양과 조작 방법에 대해 알고 싶을 때	SFC 프로그래밍 기능의 사양과 조작 방법을 설명합니다. SFC 프로그래밍 기능을 사용할 때만 참조합니다.
SBCA-303	CQM1H-PRO01 CQM1-PRO01 C200H-PRO27 +CS1W-KS001	CS/CJ 시리즈 프로그래밍 콘솔 조작 설명서	프로그래밍 콘솔의 조작 방법에 대해 알고 싶은 경우	프로그래밍 콘솔의 조작 방법을 설명합니다. 프로그래밍할 때 각 CPU 유니트의 사용 설명서와 함께 사용하십시오.
SBCD-300	CS1W-SCB□□-V1 CS1W-SCU□□-V1 CJ1W-SCU□□-V1	CS/CJ 시리즈 시리얼 커뮤니케이션 보드/유니트 사용 설명서	시리얼 커뮤니케이션 보드/유니트의 기능(하드웨어, 시리얼 통신 모드)에 대해 알고 싶은 경우 오므론이 제작한 구성 요소와 표준 시스템 프로토콜에 대해 알고 싶은 경우	1) 시리얼 커뮤니케이션 보드 및 시리얼 커뮤니케이션 유니트의 하드웨어와 소프트웨어 및 2) 표준 시스템 프로토콜에 대해 설명합니다. 주1: 시리얼 커뮤니케이션 보드/유니트의 통신 포트를 사용하여 상위 링크 모드에서 통신 명령 전송(송신 기능 포함)하는 경우, 통신 명령의 자세한 내용은 통신 명령 참조 설명서(SBCA-304)를 참조하십시오. 주2: 사용자가 작성한 프로토콜 매크로 기능에 대해 자세히 알고 싶은 경우는 CX-Protocol 조작 설명서(SBCA-307)를 참조하십시오.

설명서 번호	형식	설명서 명칭	용도	내용
SBCA-307	WS02-PSTC1-J	CX-Protocol 조작 설명서	프로토콜 매크로(송수신 시퀀스) 작성용 툴 CX-Protocol의 조작 방법에 대해 알고 싶은 경우	1) CX-Protocol의 조작 방법 및 2) 프로토콜 매크로에 대해 자세히 설명합니다. CX-Protocol을 사용하여 사용자가 시리얼 통신용 프로토콜을 작성하거나 표준 시스템 프로토콜을 사용자 정의하는 경우 참조하십시오.
			사용자가 작성하는 프로토콜 매크로 기능에 대해 자세히 알고 싶은 경우	
SBCA-347	CXONE-AL□□C-JV□/ AL□□D-JV□	CX-Integrator CS/CJ/CP/NSJ 시리즈용 네트워크 설정 툴 조작 설명서	네트워크를 설정/모니터링하고 싶은 경우	CX-Integrator의 조작 방법을 설명합니다.
SBCA-346	CXONE-AL□□C-JV□/ AL□□D-JV□	CX-One 설치 설명서	CX-One에서 소프트웨어를 설치하는 경우	FA 통합 툴 패키지 CX-One의 개요, CX-One 설치 방법에 대해 설명합니다.

안전상의 주의 사항

안전한 사용을 위한 표시 및 의미

이 사용 설명서에서는 CS/CJ 시리즈 PLC를 안전하게 사용할 수 있도록 주의 사항에 다음과 같은 표시와 그림 기호를 사용하였습니다.
여기에 표시된 주의 사항은 안전과 관련된 중요한 사항이므로 반드시 준수하십시오.
표시와 의미는 다음과 같습니다.

⚠ 경고	올바르게 취급하지 않으면 중경상을 입거나 경우에 따라 중상 또는 사망에 이를 위험이 있습니다. 또한 중대한 물적 손해가 발생할 수 있습니다.
⚠ 주의	올바르게 취급하지 않으면 경우에 따라 중경상을 입거나 물적 손해를 입을 수 있습니다.

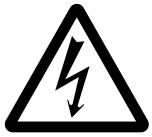
안전상의 중요 사항

제품을 안전하게 사용하기 위해 실시하거나 피해야 하는 사항을 나타냅니다.

사용상의 주의 사항

제품의 동작 불능, 오작동 또는 성능, 기능에 대한 악영향을 예방하기 위해 실시하거나 피해야 하는 사항을 나타냅니다.

그림 기호 설명



△ 기호는 주의 사항(경고 포함)을 의미합니다.
구체적인 내용은 △ 안의 기호와 문장으로 표시합니다.
왼쪽 그림은 '감전 주의'를 나타냅니다.



⃠ 기호는 금지 사항을 의미합니다.
구체적인 내용은 ⃠ 안의 기호와 문장으로 표시합니다.
왼쪽 그림은 '분해 금지'를 나타냅니다.



● 기호는 강제 사항을 의미합니다.
구체적인 내용은 ● 안의 기호와 문장으로 표시합니다.
왼쪽 그림은 '일반적인 강제 사항'을 나타냅니다.



△ 기호는 주의 사항(경고 포함)을 의미합니다.
구체적인 내용은 △ 안의 기호와 문장으로 표시합니다.
왼쪽 그림은 '일반적인 주의 사항'을 나타냅니다.



△ 기호는 주의 사항(경고 포함)을 의미합니다.
구체적인 내용은 △ 안의 기호와 문장으로 표시합니다.
왼쪽 그림은 '고온 주의'를 나타냅니다.

⚠ 경 고

전기가 흐를 때는 유니트를 분해하거나 내부에 접촉하지 마십시오.

감전의 위험이 있습니다.



전기가 흐를 때는 단자에 접촉하지 마십시오.

감전의 위험이 있습니다.



PLC(CPU 유니트 및 각 유니트 포함, 이하 **PLC)**의 고장 및 외부 요인에 의한 이상이 발생한 경우에도 시스템 전체가 안전하게 작동하도록 **PLC** 외부에서 안전 대책을 마련하십시오.
이상 동작에 의해 중대한 사고가 발생할 수 있습니다.

(1) 비상 정지 회로, 인터록 회로, 리미트 회로 등 안전 보호에 관한 회로는 반드시 **PLC** 외부의 제어 회로에 구성하십시오.

(2) **PLC**는 자가 진단 기능을 통해 이상을 검출한 경우 또는 운전 정지 고장 진단**(FALS)** 명령이 실행된 경우 운전을 정지하고 모든 출력을 **OFF**로 만듭니다.
단, **PLC**의 자가 진단 기능에서 검출되지 않는 입출력 제어부 또는 **I/O** 메모리 등에 이상이 있을 때는 의도되지 않은 출력을 하는 경우가 있습니다.
어느 경우든 시스템이 안전하게 동작하도록 **PLC** 외부에 대책을 마련하십시오.

(3) 출력 릴레이의 용착이나 연소, 출력 트랜지스터의 파괴 등에 의해 **PLC**의 출력이 **ON** 또는 **OFF**가 된 상태에서 그대로 있는 경우가 있습니다.
이런 경우 시스템이 안전하게 동작하도록 **PLC** 외부에 대책을 마련하십시오.

(4) 데이터 링크 통신이나 원격 **I/O** 통신에서 통신 이상이나 오작동이 발생해도 시스템 전체가 안전하게 작동하는 통신 시스템이나 프로그램으로 대책을 마련하십시오.



주변 툴을 사용하여 파일 메모리(메모리 카드 또는 **EM** 파일 메모리) 내의 데이터 파일을 **CPU** 유니트의 실제 **I/O(CIO)** 영역으로 전송할 때는 안전을 충분히 확인한 후 전송하십시오.
CPU 유니트의 동작 모드에 관계 없이 출력 유니트에 연결된 기기가 오작동을 일으킬 수 있습니다.



⚠ 주 의

신호선의 단선, 순간 정전에 의한 이상 신호 등에 대비하여 사용자가 페일 세이프**(fail-safe)** 대책을 마련하십시오.
이상 동작에 의해 중대한 사고가 발생할 수 있습니다.



사이클 시간이 연장되어도 영향이 없는지 확인한 후 온라인 편집을 실행하십시오.
입력 신호를 읽을 수 없는 경우가 있습니다.



다른 노드에 프로그램, **PLC** 시스템 설정, **I/O** 테이블, **I/O** 메모리, 파라미터를 전송하거나 변경할 때는 변경 대상 노드의 안전을 확인한 후 실행하십시오.
장치나 기계가 예상치 못한 동작을 할 수 있습니다.



CJ2 CPU 유니트에서 사용자 프로그램 또는 파라미터를 **CPU** 유니트에 입력하면 자동으로 플래시 메모리에 백업됩니다**(**플래시 메모리 기능**)**. 그러나 이 경우 **I/O** 메모리 영역**(DM/EM/HR** 포함**)**은 플래시 메모리에 기록되지 않습니다. 특히, 전원 **OFF** 복구 시 유지 영역인 **DM/EM/HR** 영역은 배터리에 의해 유지됩니다. 따라서 전원 이상이 발생한 경우 정확한 값을 유지하지 못할 가능성이 있습니다. 그러므로 프로그램에서 **DM/EM/HR** 영역의 내용을 사용하여 외부로 출력하는 경우 '배터리 이상 플래그**(A40204)**가 **1(ON)** 이면 외부로 출력하지 않음' 등의 대책을 마련하십시오.



AC 전원 단자의 나사는 이 설명서에 규정된 토크로 조이십시오.
나사가 헐거우면 발화 및 오작동을 일으킬 수 있습니다.



전기가 흐를 때 또는 전원을 **OFF**한 직후에는 전원 유니트에 접촉하지 마십시오.
화상의 위험이 있습니다.



외부 전원이 연결된 비절연 전원 타입의 유니트 **(CS1W-CLK12/52(-V1), CS1W-ETN01)**를 **PLC**에 장착한 경우, **PLC**에 **PC** 등의 주변 기기를 연결할 때는 주의하십시오. 외부 전원이 **24V**에 접지되어 있는 경우 주변 기기가 **0V**에 접지하면 단락됩니다. 따라서 주변 기기를 연결할 때는 외부 전원을 **0V**에 접지하거나 접지하지 마십시오.



안전상의 중요 사항

CS/CJ/NSJ 시리즈 PLC를 사용할 때는 다음 사항에 주의하십시오.

■ 전원

- 이 설명서에 지정된 전원 전압에서 사용하십시오.
- 사용하는 전원 유니트의 최대 공급 용량을 초과하지 않도록 시스템을 설계하십시오.
- 전원 사정이 나쁜 장소에서는 특히 정격 전압 및 주파수의 전원이 공급될 수 있도록 하여 사용하십시오.
- 다음과 같은 경우 PLC 본체의 전원을 OFF로 하십시오.
  - 전원 유니트나 I/O 유니트 등의 각종 유니트, CPU 유니트, 옵션 보드를 탈부착할 때
  - 장치를 조립할 때
  - 딥 스위치 또는 로터리 스위치를 설정할 때
  - 케이블을 연결하거나 배선할 때
  - 커넥터를 부착하거나 분리할 때
- 단자에서 단자로 점퍼 선을 사용하여 배선하면 각각의 전선에 가산된 전류가 흐릅니다. 점퍼 선으로 배선할 때는 모든 전선의 전류 용량에 주의하십시오.
- 교환 시기 알림 기능이 있는 전원 유니트를 사용할 때는 다음 사항에 주의하십시오.
  - 전원 유니트 전면 LED가 '0.0'과 'A02'를 교대로 표시하기 시작한 경우 또는 알람 출력이 자동으로 OFF가 된 경우 6개월 이내에 전원 유니트를 교환하십시오.
  - 알람 출력 케이블은 동력선이나 고압선에서 떨어뜨리십시오.
  - 알람 출력에 정격을 초과한 전압을 인가하거나 부하를 연결하지 마십시오.
  - 3개월 이상 보관해 두는 경우 교환 시기 알림 기능을 유지하기 위해 주위 온도 -20~+30℃, 상대 습도 25~70%인 장소에 보관하십시오.
  - 부착 상태에 따라 방열성이 악화되어 교환 시기 알림 신호에서 시기에 오차가 발생하거나 내부 소자의 성능 저하 또는 파손이 발생할 수 있습니다. 표준 부착 이외의 방법으로는 사용하지 마십시오.
- 전원을 OFF한 직후에는 전원 유니트의 단자에 접촉하지 마십시오. 잔류 전압에 의한 감전의 위험이 있습니다.
- PC와 PLC를 연결하는 경우 전위차에 의한 고장을 막기 위해 다음 사항에 주의하십시오.
  - 노트북 컴퓨터를 PLC에 연결하는 경우 PC의 전원 플러그를 AC 콘센트에서 빼십시오. AC 어댑터의 누설 전류에 의해 PC와 PLC 사이에 전위차가 나타나는 경우가 있습니다.
    PC와 PLC를 연결한 후 다시 AC 어댑터에서 전원을 공급하십시오.
  - PC에 FG 단자가 있는 경우에는 PLC의 FG(GR) 단자와 같은 전위가 되도록 연결하십시오.
- PC를 다른 위치에 접지하면 접지 조건에 따라 전위차가 발생할 가능성이 있습니다.

■ 설치

- 강한 고주파 노이즈가 발생하는 기기에서 멀리 떨어진 곳에 부착하십시오.
- 접지된 금속을 만지는 등 신체의 정전기를 방전한 후 유니트에 접촉하십시오.
- 단자대, 커넥터, 메모리 카드, 옵션 보드, 증설 케이블 등에 잠금 장치가 있는 경우 잠겨 있는지 반드시 확인한 후 사용하십시오.

- 전원 유니트나 CPU 유니트, I/O 유니트, 고기능 I/O 유니트, CPU 고기능 유니트를 연결할 때는 유니트끼리 연결한 후, 상하 슬라이더를 딸깍하고 소리가 날 때까지 밀어서 확실하게 잠그십시오.
- 잠그지 않으면 완전한 기능을 사용할 수 없는 경우가 있으므로 주의하십시오.

■ 배선

- 하드웨어 사용 설명서의 지시에 따라 정확하게 배선하십시오.
- 배선, 스위치 등의 설정을 충분히 확인한 후 전기가 흐르게 하십시오.
- 단자대의 나사, 케이블의 나사는 이 설명서에 규정된 토크로 조이십시오.
- 단자대, 커넥터는 장치 위치를 충분히 확인한 후 장착하십시오.
- 유니트 윗면에 방진 라벨을 부착한 상태로 배선하십시오.
- 배선을 완료한 후에는 방열을 위해 반드시 라벨을 떼어내고 사용하십시오.
- 압착 단자를 부착하여 배선하십시오. 꼬아서 만든 전선을 단자대에 직접 연결하지 마십시오.
- 입력 유니트에는 정격 입력 전압을 초과하는 전압을 인가하지 마십시오.
- 설치 공사를 할 때는 반드시 D종 접지(제3종 접지)를 실시하십시오.
  특히 전원 유니트의 LG 단자와 GR 단자를 단락한 경우 감전 방지를 위해 반드시 D종 접지(제3종 접지)를 실시하십시오.
- 출력 유니트에 최대 개폐 능력을 초과한 전압을 인가하거나 부하를 연결하지 마십시오.
- 케이블을 무리하게 구부리거나 당기지 마십시오.
- 케이블이나 코드에 물건을 올려놓지 마십시오.
- 연결 케이블은 설명서에 기재된 전용 케이블을 사용하거나 제작하십시오. 시판되는 일반 PC용 RS-232C 케이블을 사용하면 외부 기기 및 CPU 유니트가 고장날 수 있습니다.
- CPU 유니트의 RS-232C 포트의 6번 핀(+5V 전원)은 RS-422A 변환 어댑터(CJ1W-CIF11), RS-232C/RS-422A 변환 유니트(NT-AL001) 이외의 외부 연결 기기에 연결하지 마십시오. 외부 연결 기기 및 CPU 유니트가 고장날 수 있습니다.

■ 취급 주의 사항

- 내전압 시험 시, 스위치를 사용하여 전체 전압을 인가 또는 차단했을 때 발생하는 임펄스 전압에 의해 전원 유니트가 파손될 수 있습니다. 인가 전압은 시험기의 볼륨을 서서히 변화시키십시오.
- 절연 저항 시험 및 내전압 시험은 전원 유니트의 LG 단자를 GR 단자에서 분리한 후 실시하십시오.
- 운전을 시작하기 전에 딥 스위치나 데이터 메모리(DM)가 정확하게 설정되었는지 확인하십시오.
- 운전 재개에 필요한 데이터 메모리나 유지 릴레이의 내용, 프로그램, 파라미터 및 데이터는 교환한 CPU 유니트, 고기능 I/O 유니트, CPU 고기능 유니트에 전송한 후 운전을 재개하십시오.
- 다음 조작은 설비에 영향을 주지 않는지 확인한 후 실시하십시오.
  - PLC의 동작 모드 전환(전원 투입 시의 동작 모드 설정 포함)
  - 접점의 강제 세트/리셋
  - 현재값 또는 설정값 변경
- 본 제품을 분해하여 수리 또는 개조하지 마십시오.
- 제품을 떨어뜨리거나 비정상적인 진동 또는 충격을 주지 마십시오.
- 전기가 통하지 않는 상태에서 배터리 없이 방치하고, 배터리 장착 후 전원을 한번도 투입하지 않은 상태에서 방치하면 배터리 수명이 짧아질 수 있습니다.
- 배터리에 이상이 발생한 경우 또는 배터리에 이상이 없어도 배터리의 규정 백업 시간

이 경과된 경우 신속하게 배터리를 교환하십시오. 또한 교환용 배터리는 배터리 라벨에 표시된 제조일이 2년 이내인 것을 사용하십시오.

- 메모리의 내용이 파괴될 가능성이 있으므로 배터리 교환 전에는 5분 이상 전기가 통하게 하고, 전원을 끈 후 5분 이내에 새로운 배터리로 교환하십시오.
- 배터리 이상 플래그를 사용하는 활용 사례에서 전기가 통하는 상태로 배터리를 교환하는 경우, 배터리 이상을 검출하면 먼저 설비에 영향이 없는지 확인한 후 작업하십시오.
- 배터리는 전지액 누출, 파열, 발열, 발화 등의 위험이 있으므로 +, -의 단락, 충전, 분해, 가열, 불에 집어넣거나 강한 충격을 주는 일은 절대 하지 마십시오. 또한 바닥에 떨어뜨리는 등 강한 충격을 준 배터리는 전지액이 누출될 우려가 있으므로 절대 사용하지 마십시오.
- 숙련된 기술자가 배터리를 교환하도록 UL 규격에 규정되어 있습니다. 교환 작업은 숙련된 기술자가 담당하십시오. 또한 하드웨어 사용 설명서에 기재된 방법으로 교환하십시오.
- 본체 및 불필요한 배터리의 폐기는 지방자치단체의 규제를 받는 경우가 있습니다. 각 지방자치단체의 규정에 따라 폐기하십시오.



폐건전지 회수

- IOM(I/O 메모리) 유지 비트를 1(ON)로 설정한 경우 '운전' 또는 '모니터' 모드에서 '프로그램' 모드로 변경하면, 출력 릴레이는 클리어(OFF)되지 않고 직전 상태를 유지하므로 외부 부하의 상태에 주의하십시오. (또한 운전 정지 이상(FALS 명령 실행 포함)에 의해 운전이 정지되었을 때는 CPU 유니트의 내부 I/O 메모리 값이 유지되지만 출력 유니트의 출력 접점은 모두 OFF가 됩니다.)
- 부적절한 데이터 링크 테이블/파라미터가 설정되어 있으면 설비가 예기치 않게 동작할 위험이 있습니다.
  또한 적절한 데이터 링크 테이블/파라미터가 설정된 경우에도 설비에 영향을 주지 않는지 확인한 후, 데이터 링크를 기동/정지하십시오.
- 데이터 링크 통신의 경우 상태 플래그에 있는 PLC 본체 이상 정보를 참조하여 데이터 링크 대상 PLC에 이상이 발생하지 않은 경우에만 수신 데이터를 사용하는 프로그램을 작성하십시오.
  이상이 발생하면 데이터 링크 대상 PLC가 잘못된 데이터를 송신할 가능성이 있습니다.
- 라우팅 테이블을 주변 툴에서 PLC 본체로 전송하면 모든 CPU 고기능 유니트가 리셋됩니다. 이는 설정한 라우팅 테이블을 읽어와서 유효화로 하기 때문입니다. 리셋해도 설비에 영향을 주지 않는다는 것을 확인한 후 전송하십시오.
- PLC 운전 중에 태그 데이터 링크 파라미터를 전송할 수 있지만 태그 데이터 링크 파라미터 전송 중에는 관련 노드 간의 태그 데이터 링크가 정지됩니다. 설비에 영향을 주지 않는다는 것을 확인한 후 전송하십시오.
- 네트워크의 통신 장애가 발생한 경우 출력 상태는 사용되는 제품의 사양을 따릅니다.
  출력을 함께 하는 제품을 사용하는 경우 통신 장애 발생 시 어떻게 동작할지 확인한 후 안전 대책을 마련하십시오.

- 주변 툴(CX-Programmer/프로그래밍 콘솔)에 따라 AUTOEXEC.IOM 파일(전원 ON일 때 자동 전송용 데이터 파일)을 작성할 때는 반드시 쓰기 시작 주소에 D20000을 지정하고 쓰기 크기에 DM 영역 범위 내의 크기를 지정하십시오.
  D20000 이외를 지정하여 AUTOEXEC.IOM 파일을 작성한 경우에도 전원을 켜면 메모리 카드에서 파일의 데이터를 읽어와 CPU 유니트의 D20000을 시작 주소로 하여 전송됩니다. 이런 경우, D20000 이후를 잘못된 데이터로 덮어 쓰게 되므로 주의하십시오. 또한 크기가 DM 영역 범위를 초과한 경우(CX-Programmer만 가능) 초과분은 EM 영역을 덮어 쓰게 되므로 주의하십시오.
- CJ 시리즈 CPU 유니트의 경우, 공장 출하 시 배터리가 미리 장착되어 있습니다. 또한 내장 시계의 시각은 미리 설정되어 있습니다. 따라서, CS 시리즈 CS1 CPU 유니트와 달리 처음 사용할 때 메모리를 모두 삭제하거나 내장된 시계를 설정할 필요가 없습니다.
- CJ2 CPU 유니트는 사용자 프로그램, 파라미터 영역의 데이터를 내장 플래시 메모리에 백업합니다. 백업 중에는 CPU 유니트 전면의 BKUP LED가 켜집니다. LED가 켜진 상태에서 PLC 본체의 전원을 끄지 마십시오. 전원을 끄면 데이터가 백업되지 않습니다.
- 작성한 사용자 프로그램 및 유니트의 파라미터 설정은 동작을 충분히 확인한 후 본 운전으로 이동하십시오.
- I/O 테이블에 고기능 유니트를 설정한 후 유니트를 다시 시작할 때는 접속할 대상 기기의 안전을 충분히 확인한 후 실행하십시오.
- 래더, ST 프로그램에서 배열 변수의 첨자에 변수 또는 주소(ST 프로그램일 때는 변수만)를 지정하여 요소 번호를 간접 지정하는 경우 메모리 영역의 최대 범위를 초과하는 요소 번호를 지정하지 않도록 프로그래밍하십시오.
  최대 범위를 초과하는 요소 번호를 지정하면 배열 범위 밖 영역의 데이터를 읽고 써서 예기치 않게 동작할 위험이 있습니다.
- 래더 프로그램에서 주소의 오프셋 값에 변수 또는 주소를 지정하여 오프셋 값을 간접 지정하는 경우 시작 주소에 속하는 메모리 영역의 범위를 초과하지 않도록 프로그래밍하십시오.
  시작 주소에 속하는 메모리 영역의 범위를 초과하여 지정하면 다른 메모리 영역의 데이터를 읽고 써서 예기치 않게 동작할 위험이 있습니다.
- CS1/CJ1 CPU 유니트의 프로그램 파일(***.OBJ)은 메모리 카드를 사용하여 직접 CJ2 CPU 유니트로 전송할 수 없습니다. CX-Programmer에서 한번 CJ2 CPU 유니트용으로 교환한 후 전송하십시오.

■ 외부 회로

- PLC 본체의 전원을 켠 후 제어부의 전원을 켜도록 외부 회로를 구성하십시오. 제어부의 전원을 켠 후 PLC 본체의 전원을 켜면, PLC 본체의 전원을 켤 때 DC 출력 유니트 등의 출력이 순간적으로 오작동하여 제어부의 출력이 일시적으로 정상 동작을 하지 않을 수 있습니다.
- 외부 배선의 단락에 대비하여 전류 차단기 등의 안전 대책을 마련하십시오.
- 메모리 카드 액세스 중에는 PLC의 전원을 끄거나 그대로 메모리 카드를 빼지 마십시오. 메모리 카드를 사용할 수 없게 될 수 있습니다.
  메모리 카드를 뺄 때는 메모리 카드의 전원 공급 정지 푸시 버튼을 누르고 BUSY LED가 꺼졌는지 확인한 후 빼십시오.

사용상의 주의 사항

- 이 설명서의 지시에 따라 정확하게 설치하십시오.

- 다음과 같은 장소에는 설치하지 마십시오.
  - 햇빛이 닿는 장소
  - 주위 온도 및 상대 습도가 사양 값의 범위를 초과하는 장소
  - 온도 변화가 급격하여 결로가 발생하는 장소
  - 부식성 가스 및 가연성 가스가 있는 장소
  - 먼지, 염분, 철분이 많은 장소
  - 물, 기름, 약품 등의 거품이 날아오는 장소
  - 본체에 진동 및 충격이 직접 전해지는 장소

- 다음과 같은 장소에서 사용할 때는 충분한 차폐 대책을 마련하십시오.
  - 정전기 등에 의한 노이즈가 발생하는 장소
  - 강한 전계 및 자계가 발생하는 장소
  - 방사능에 피폭될 위험이 있는 장소
  - 전원선이나 동력선이 근처를 통과하는 장소

설명서 개정 이력

설명서 개정 기호는 앞 표지와 뒷표지 왼쪽 아래에 기재된 카탈로그 번호 끝에 표기하였습니다.



개정 기호	개정 연월	개정 이유 및 개정 페이지
–	2008년 6월	초판
B	2008년 12월	CJ 시리즈 CJ2 CPU 유니트 CJ2H-CPU6□ 추가, 기재 오류 수정
C	2009년 2월	유니트 간 동기 제어 관련 기술 추가
D	2009년 7월	다음 2가지 명령에 대한 기술 추가 ・ 아날로그 입력 직접 변환(AIDC) ・ 아날로그 출력 직접 변환(AODC)
E	2009년 9월	다음 2가지 시리얼 통신 명령 추가 ・ 시리얼 커뮤니케이션 유니트 시리얼 포트 직접 송신(DTXDU) ・ 시리얼 커뮤니케이션 유니트 시리얼 포트 직접 수신(DRXDU)
F	2009년 10월	EMBC에 EM 강제 세트/리셋 기능 관련 정보 추가 기술. 기재 오류 수정
G	2010년 2월	CJ 시리즈 CJ2M CPU 유니트 CJ2M-CPU□□ 추가, 다음 명령 추가 ・ 데이터 비교 명령 부호 있는 영역 비교(ZCPS) 부호 있는 2워드 영역 비교(ZCPSL) ・ I/O 유니트용 명령 위치 결정 고속 기동(NCDMV) 위치 결정 시작 트리거(NCDTR) 다음 명령에 CJ2M 관련 기술 추가 ・ 인터럽트 제어 명령 ・ 데이터 비교 명령 기재 오류 수정

제1장 명령어의 기본 지식 1

제2장 명령 목록 2

제3장 각 명령에 대한 설명 3

제4장 명령 처리 시간/스텝 수 목록 4

부록 부록

목차

제1장 명령어의 기본 지식

1-1 명령어의 기본 지식 1-2
1-2 피연산자 지정 방법 1-11
1-3 데이터 형식 1-19

제2장 명령 목록

명령 목록(CPU 유니트별 사용 가능 여부) 2-2

제3장 각 명령에 대한 설명

이 장을 보는 방법 3-2

시퀀스 입력 명령

로드 LD/로드 부정 LD NOT 3-8
논리곱 AND/부정 논리곱 AND NOT 3-10
논리합 OR/부정 논리합 OR NOT 3-12
로드 논리곱 AND LD/로드 논리합 OR LD 3-14
부정 NOT(520) 3-17
P.F. 기동 미분 UP(521)/P.F. 종료 미분 DOWN(522) 3-18
LD 비트 테스트 LD TST(350)/LD 비트 테스트 부정 LD TSTN(351) 3-20
AND 비트 테스트 AND TST(350)/AND 비트 테스트 부정 AND TSTN(351) 3-22
OR 비트 테스트 OR TST(350)/OR 비트 테스트 부정 OR TSTN(351) 3-24

시퀀스 출력 명령

출력 OUT/부정 출력 OUT NOT 3-28
일시 기억 릴레이 TR 3-30
유지 KEEP(011) 3-32
기동 미분 DIFU(013) 3-36
종료 미분 DIFD(014) 3-38
세트 SET/리셋 RSET 3-40
다중 비트 세트 SETA(530)/다중 비트 리셋 RSTA(531) 3-42
1비트 세트 SETB(532)/1비트 리셋 RSTB(533) 3-44
1비트 출력 OUTB(534) 3-46

시퀀스 제어 명령

종료 END(001) 3-52
기능 없음 NOP(000) 3-53
인터록 IL(002)/인터록 클리어 ILC(003) 3-54
멀티 인터록(미분 플래그 유지형) MILH(517)/
멀티 인터록(미분 플래그 비유지형) MILR(518)/
멀티 인터록 클리어 MILC(519) 3-57
점프 JMP(004)/점프 끝 JME(005) 3-66
조건부 점프 CJP(510)/조건부 부정 점프 CJPN(511) 3-69
복수 점프 JMP0(515)/복수 점프 끝 JME0(516) 3-72
반복 시작 FOR(512)/반복 종료 NEXT(513) 3-74
루프 중단 BREAK(514) 3-77

타이머/카운터 명령

100ms 타이머(타이머) TIM/TIMX(550) 3-88
10ms 타이머(고속 타이머) TIMH(015)/TIMHX(551) 3-91
1ms 타이머(초고속 타이머) TMHH(540)/TMHHX(552) 3-94
0.1ms 타이머 TIMU(541)/TIMUX(556) 3-96
0.01ms 타이머 TMUH(544)/TMUHX(557) 3-98
적산 타이머 TTIM(087)/TTIMX(555) 3-100
장시간 타이머 TIML(542)/TIMLX(553) 3-103
멀티 출력 타이머 MTIM(543)/MTIMX(554) 3-106
카운터 CNT/CNTX(546) 3-109
가역 카운터 CNTR(012)/CNTRX(548) 3-112
타이머/카운터 리셋 CNR(545)/CNRX(547) 3-115
타이머 리셋 TRSET(549) 3-117

데이터 비교 명령

심볼 비교 =, <>, <, <=, >, >=(300~328) 3-120
시각 비교 =DT, <> DT, < DT, <= DT, > DT, >= DT(341~346) 3-124
부호 없는 비교 CMP(020)/부호 없는 2워드 비교 CMPL(060) 3-128
부호 있는 BIN 비교 CPS(114)/부호 있는 BIN 2워드 비교 CPSL(115) 3-131
다채널 비교 MCMP(019) 3-134
테이블 일치 TCMP(085) 3-136
부호 없는 테이블 간 비교 BCMP(068) 3-138
확장 테이블 간 비교 BCMP2(502) 3-140
영역 비교 ZCP(088)/2워드 영역 비교 ZCPL(116) 3-143
부호 있는 영역 비교 ZCPS(117)/부호 있는 2워드 영역 비교 ZCPSL(118) 3-147

데이터 전송 명령

전송 MOV(021)/2워드 전송 MOVL(498) 3-152
부정 전송 MVN(022)/부정 2워드 전송 MVNL(499) 3-154
비트 전송 MOVB(082) 3-156
자릿수 전송 MOVD(083) 3-158
다중 비트 전송 XFRB(062) 3-160
블록 전송 XFER(070) 3-162
블록 설정 BSET(071) 3-164
데이터 교환 XCHG(073)/데이터 2워드 교환 XCGL(562) 3-166
데이터 분배 DIST(080) 3-168
데이터 추출 COLL(081) 3-170
인덱스 레지스터 설정 MOVR(560)/인덱스 레지스터 설정 MOVRW(561) 3-172

데이터 이동 명령

이동 레지스터 SFT(010) 3-176
좌우 이동 레지스터 SFTR(084) 3-178
비동기 이동 레지스터 ASFT(017) 3-180
워드 이동 WSFT(016) 3-182
1비트 좌측 이동 ASL(025)/1비트 좌측 이동(2워드) ASLL(570) 3-184
1비트 우측 이동 ASR(026)/1비트 우측 이동(2워드) ASRL(571) 3-186
CY가 있는 1비트 좌측 회전 ROL(027)/CY가 있는
1비트 좌측 회전(2워드) ROLL(572) 3-188
CY가 없는 1비트 좌측 회전 RLNC(574)/CY가 없는
1비트 좌측 회전(2워드) RLNL(576) 3-190
CY가 있는 1비트 우측 회전 ROR(028)/CY가 있는
1비트 우측 회전(2워드) RORL(573) 3-192

Y가 없는 1비트 우측 회전 RRNC(575)/CY가 없는
1비트 우측 회전(2워드) RRNL(577) 3-194
1자리 좌측 이동 SLD(074)/1자리 우측 이동 SRD(075) 3-196
N비트 데이터 좌측 이동 NSFL(578)/N비트 데이터 우측 이동 NSFR(579) 3-198
N비트 좌측 이동 NASL(580)/N비트 2워드 좌측 이동 NSLL(582) 3-200
N비트 우측 이동 NASR(581)/N비트 2워드 우측 이동 NSRL(583) 3-203

증가/감소 명령

BIN 증가 ++(590)/BIN 2워드 증가 ++L(591) 3-208
BIN 감소 －－(592)/BIN 2워드 감소 －－L(593) 3-211
BCD 증가 ++B(594)/BCD 2워드 증가 ++BL(595) 3-214
BCD 감소 －－B(596)/BCD 2워드 감소 －－BL(597) 3-217

사칙연산 명령

부호 있음, CY 없는 BIN 가산 +(400)/부호 있음, CY 없는 BIN 2워드 가산 +L(401) 3-222
부호 및 CY 있는 BIN 가산 +C(402)/부호 및 CY 있는 BIN 2워드 가산 +CL(403) 3-224
CY 없는 BCD 가산 +B(404)/CY 없는 BCD 2워드 가산 +BL(405) 3-227
CY 있는 BCD 가산 +BC(406)/CY 있는 BCD 2워드 가산 +BCL(407) 3-229
부호 있음, CY 없는 BIN 감산 －(410)/부호 있음, CY 없는 BIN 2워드 감산 －L(411) 3-231
부호 및 CY 있는 BIN 감산 －C(412)/부호 및 CY 있는 BIN 2워드 감산 －CL(413) 3-235
CY 없는 BCD 감산 －B(414)/CY 없는 BCD 2워드 감산 －BL(415) 3-238
CY 있는 BCD 감산 －BC(416)/CY 있는 BCD 2워드 감산 －BCL(417) 3-241
부호 있는 BIN 승산 *(420)/부호 있는 BIN 2워드 승산 *L(421) 3-243
부호 없는 BIN 승산 *U(422)/부호 없는 BIN 2워드 승산 *UL(423) 3-245
BCD 승산 *B(424)/BCD 2워드 승산 *BL(425) 3-247
부호 있는 BIN 제산/(430)/부호 있는 BIN 2워드 제산/L(431) 3-249
부호 없는 BIN 제산/U(432)/부호 없는 BIN 2워드 제산/UL(433) 3-251
BCD 제산/B(434)/BCD 2워드 제산/BL(435) 3-253

데이터 변환 명령

BCD→BIN 변환 BIN(023)/BCD→BIN 2워드 변환 BINL(058) 3-256
BIN→BCD 변환 BCD(024)/BIN→BCD 2워드 변환 BCDL(059) 3-258
2의 보수 변환 NEG(160)/2의 보수 변환(2워드) NEGL(161) 3-261
부호 확장 SIGN(600) 3-263
4→16/8→256 디코더 MLPX(076) 3-265
16→4/256→8 인코더 DMPX(077) 3-270
ASCII 코드 변환 ASC(086) 3-275
ASCII→HEX 변환 HEX(162) 3-279
비트 열→비트 행 변환 LINE(063) 3-283
비트 행→비트 열 변환 COLM(064) 3-285
부호 있는 BCD→BIN 변환 BINS(470)/부호 있는 BCD→BIN 2워드 변환 BISL(472) 3-287
부호 있는 BIN→BCD 변환 BCDS(471)/부호 있는 BIN→BCD 2워드 변환 BDSL(473) 3-291
그레이 코드 변환 GRY(474) 3-295
그레이 코드→BIN 변환 GRAY_BIN(478)/
그레이 코드→BIN 2워드 변환 GRAY_BINL(479) 3-298
BIN→그레이 코드 변환 BIN_GRAY(480)/
BIN→그레이 코드 2워드 변환 BIN_GRAYL(481) 3-300
숫자 4자리→ASCII 데이터 변환 STR4(601)/숫자 8자리→ASCII 데이터 변환 STR8(602)/
숫자 16자리→ASCII 데이터 변환 STR16(603) 3-302
ASCII 데이터→숫자 4자리 변환 NUM4(604)/ASCII 데이터→숫자 8자리 변환 NUM8(605)/
ASCII 데이터→숫자 16자리 변환 NUM16(606) 3-305

논리 연산 명령

워드 논리곱 ANDW(034)/워드 논리곱(2워드) ANDL(610) 3-310
워드 논리합 ORW(035)/워드 논리합(2워드) ORWL(611) 3-312
워드 배타적 논리합 XORW(036)/워드 배타적 논리합(2워드) XORL(612) 3-314
워드 배타적 논리합 부정 XNRW(037)/
워드 배타적 논리합 부정(2워드) XNRL(613) 3-316
비트 반전 COM(029)/비트 2워드 반전 COML(614) 3-318

특수 연산 명령

BIN 제곱근 연산 ROTB(620) 3-322
BCD 제곱근 연산 ROOT(072) 3-324
숫자 변환 APR(069) 3-326
부동 소수점 제산(BCD) FDIV(079) 3-335
비트 카운터 BCNT(067) 3-338

부동 소수점 변환 및 연산 명령

부동 소수점→16비트 BIN 변환 FIX(450)/
부동 소수점→32비트 BIN 변환 FIXL(451) 3-347
16비트 BIN→부동 소수점 변환 FLT(452)
32비트 BIN→부동 소수점 변환 FLTL(453) 3-349
부동 소수점 가산 +F(454)/부동 소수점 감산 －F(455)/
부동 소수점 승산 *F(456)/부동 소수점 제산/F(457) 3-351
각도→라디안 변환 RAD(458) 3-354
라디안→각도 변환 DEG(459) 3-356
SIN 연산 SIN(460)/COS 연산 COS(461)/TAN 연산 TAN(462) 3-358
고속 SIN 연산 SINQ(475)/고속 COS 연산 COSQ(476)/
고속 TAN 연산 TANQ(477) 3-361
SIN^{-1} 연산 ASIN(463)/COS^{-1} 연산 ACOS(464)/TAN^{-1} 연산 ATAN(465) 3-364
제곱근 연산 SQRT(466) 3-367
지수 연산 EXP(467) 3-369
로그 연산 LOG(468) 3-371
거듭제곱 연산 PWR(840) 3-373
단정도 부동 소수점 데이터 비교 =F, <>F, <F, <=F, >F, >=F(329~334) 3-375
부동 소수점<단>→문자열 변환 FSTR(448) 3-378
문자열→부동 소수점<단> 변환 FVAL(449) 3-382
부동 소수점 전송<단> MOVF(469) 3-386

배정도 부동 소수점 변환 및 연산 명령

부동 소수점→16비트 BIN 변환<배> FIXD(841)/
부동 소수점→32비트 BIN 변환<배> FIXLD(842) 3-396
16비트 BIN→부동 소수점 변환<배> DBL(843)/
32비트 BIN→부동 소수점 변환<배> DBLL(844) 3-398
부동 소수점 가산<배> +D(845)/부동 소수점 감산<배> －D(846)
부동 소수점 승산<배> *D(847)/부동 소수점 제산<배>/D(848) 3-400
각도→라디안 변환<배> RADD(849) 3-403
라디안→각도 변환<배> DEGD(850) 3-405
SIN 연산<배> SIND(851)/COS 연산<배> COSD(852)/
TAN 연산<배> TAND(853) 3-407
SIN^{-1} 연산<배> ASIND(854)/COS^{-1} 연산<배> ACOSD(855)/
TAN^{-1} 연산<배> ATAND(856) 3-410
제곱근 연산<배> SQRTD(857) 3-413
지수 연산<배> EXPD(858) 3-415

로그 연산<배> LOGD(859) 3-417
거듭제곱 연산<배> PWRD(860) 3-419
배정도 부동 소수점 데이터 비교 =D, <>D, <D, <=D, >D, >=D(335~340) 3-421

테이블 데이터 처리 명령

스택 영역 설정 SSET(630) 3-430
스택 데이터 저장 PUSH(632) 3-432
후입 선출 LIFO(634)/선입 선출 FIFO(633) 3-434
스택 데이터 수 출력 SNUM(638) 3-438
스택 데이터 참조 SREAD(639) 3-440
스택 데이터 갱신 SWRIT(640) 3-443
스택 데이터 삽입 SINS(641) 3-446
스택 데이터 삭제 SDEL(642) 3-449
테이블 영역 선언 DIM(631) 3-452
레코드 위치 설정 SETR(635) 3-454
레코드 위치 읽기 GETR(636) 3-456
데이터 검색 SRCH(181) 3-458
바이트 스와핑 SWAP(637) 3-461
최대값 검색 MAX(182)/최소값 검색 MIN(183) 3-463
2워드 데이터 최대값 검색 MAXL(174) 3-467
단정도 부동 소수점 데이터 최대값 검색 MAXF(176) 3-470
배정도 부동 소수점 데이터 최대값 검색 MAXD(178) 3-472
2워드 데이터 최소값 검색 MINL(175) 3-474
단정도 부동 소수점 데이터 최소값 검색 MINF(177) 3-476
배정도 부동 소수점 데이터 최소값 검색 MIND(179) 3-478
SUM 값 산출 SUM(184) 3-480
FCS 값 산출 FCS(180) 3-483

추적 명령

부호 없는 1W 레코드 검색 RSRCH <(360)/RSRCH <=(361)/
RSRCH =(362)/RSRCH >=(363)/RSRCH >(364) 3-495
부호 없는 2W 레코드 검색 RSRCH2 <(370)/RSRCH2 <=(371)/
RSRCH2 =(372)/RSRCH2 >=(373)/RSRCH2 >(374) 3-501
부호 없는 4W 레코드 검색 RSRCH4 <(380)/RSRCH4 <=(381)/
RSRCH4 =(382)/RSRCH4 >=(383)/RSRCH4 >(384) 3-504
부호 없는 1W 레코드 정렬 RSORT(203) 3-507
부호 없는 2W 레코드 정렬 RSORT2(204) 3-511
부호 없는 4W 레코드 정렬 RSORT4(205) 3-514

데이터 제어 명령

PID 연산 PID(190) 3-518
자동 조정 가능한 PID 연산 PIDAT(191) 3-527
상하한 리미트 제어 LMT(680) 3-534
데드 밴드 제어 BAND(681) 3-536
데드 존 제어 ZONE(682) 3-538
시분할 비례 출력 TPO(685) 3-540
스케일링 SCL(194) 3-546
스케일링 2 SCL2(486) 3-550
스케일링 3 SCL3(487) 3-553
데이터 평균화 AVG(195) 3-556

서브루틴 명령

서브루틴 호출 SBS(091) 3-561
매크로 MCRO(099) 3-566
서브루틴 진입 SBN(092)/서브루틴 반환 RET(093) 3-569
글로벌 서브루틴 호출 GSBS(750) 3-572
글로벌 서브루틴 진입 GSBN(751)/
글로벌 서브루틴 반환 GRET(752) 3-578

인터럽트 제어 명령

인터럽트 마스크 세트 MSKS(690) 3-585
인터럽트 마스크 읽기 MSKR(692) 3-591
인터럽트 해제 CLI(691) 3-595
인터럽트 태스크 실행 금지 DI(693) 3-599
인터럽트 태스크 실행 금지 해제 EI(694) 3-601

고속 카운터/펄스 출력 명령

동작 모드 제어 INI(880) 3-604
펄스 현재값 읽기 PRV(881) 3-607
펄스 주파수 변환 PRV2(883) 3-612
비교 테이블 등록 CTBL(882) 3-615
주파수 설정 SPED(885) 3-618
펄스량 설정 PULS(886) 3-621
위치 결정 PLS2(887) 3-623
주파수 가감속 제어 ACC(888) 3-628
원점 검색 ORG(889) 3-632
PWM 출력 PWM(891) 3-635

공정 진행 제어 명령

스텝 래더 영역 진행 SNXT(009)/스텝 래더 영역 정의 STEP(008) 3-639

I/O 유니트용 명령

I/O 리프레시 IORF(097) 3-650
고기능 I/O 유니트 매회 리프레시 FIORF(225) 3-653
CPU 고기능 유니트 매회 I/O 리프레시 DLNK(226) 3-656
7세그먼트 디코더 SDEC(078) 3-659
디지털 스위치 DSW(210) 3-661
10키 입력 TKY(211) 3-664
16키 입력 HKY(212) 3-667
행렬 입력 MTR(213) 3-670
7세그먼트 표시 7SEG(214) 3-674
아날로그 입력 직접 변환 AIDC(216) 3-677
아날로그 출력 직접 변환 AODC(217) 3-680
위치 결정 고속 기동 NCDMV(218) 3-683
위치 결정 시작 트리거 NCDTR(219) 3-688
인텔리전트 I/O 읽기 IORD(222) 3-691
인텔리전트 I/O 쓰기 IOWR(223) 3-694

시리얼 통신 명령

프로토콜 매크로 PMCR(260) 3-700
CPU 유니트 시리얼 커뮤니케이션 보드 시리얼 포트 출력 TXD(236) 3-706
CPU 유니트 시리얼 커뮤니케이션 보드 시리얼 포트 입력 RXD(235) 3-712
시리얼 커뮤니케이션 유니트 시리얼 포트 출력 TXDU(256) 3-720

시리얼 커뮤니케이션 유니트 시리얼 포트 입력 RXDU(255) 3-726
시리얼 커뮤니케이션 유니트 시리얼 포트 직접 송신 DTXDU(262) 3-733
시리얼 커뮤니케이션 유니트 시리얼 포트 직접 수신 DRXDU(261) 7-738
시리얼 포트 통신 설정 변경 STUP(237) 3-745
프로토콜 매크로 송신 2 PMCR2(264) 3-748

네트워크 통신용 명령

네트워크 송신 SEND(090) 3-774
네트워크 송신 2 SEND2(491) 3-779
네트워크 수신 RECV(098) 3-783
네트워크 수신 2 RECV2(492) 3-788
명령 송신 CMND(490) 3-791
명령 송신 2 CMND2(493) 3-798
범용 Explicit 메시지 송신 명령 EXPLT(720) 3-802
Explicit 읽기 명령 EGATR(721) 3-807
Explicit 쓰기 명령 ESATR(722) 3-812
Explicit CPU 유니트 데이터 읽기 명령 ECHRD(723) 3-816
Explicit CPU 유니트 데이터 쓰기 명령 ECHWR(724) 3-819

파일 메모리용 명령

데이터 파일 읽기 FREAD(700) 3-831
데이터 파일 쓰기 FWRIT(701) 3-836
텍스트 파일 쓰기 TWRIT(704) 3-841

표시 기능용 명령

메시지 표시 MSG(046) 3-848

시계 기능용 명령

캘린더 가산 CADD(730)/캘린더 감산 CSUB(731) 3-852
시분초→초 변환 SEC(065) 3-856
초→시분초 변환 HMS(066) 3-858
시계 보정 DATE(735) 3-860

디버그 처리 명령

트레이스 메모리 샘플링 TRSM(045) 3-864

고장 진단 명령

운전 계속 고장 진단 FAL(006) 3-868
운전 정지 고장 진단 FALS(007) 3-874
고장 지점 검출 FPD(269) 3-879

특수 명령

세트 캐리 STC(040)/클리어 캐리 CLC(041) 3-888
확장 DM 뱅크 전환 EMBC(281) 3-889
사이클 시간 감시 시간 설정 WDT(094) 3-891
조건 플래그 저장 CCS(282)/조건 플래그 로드 CCL(283) 3-893
CV→CS 주소 변환 FRMCV(284) 3-895
CS→CV 주소 변환 TOCV(285) 3-898
주변 서비스 금지 IOSP(287)/주변 서비스 금지 해제 IORS(288) 3-901

블록 프로그램 명령

블록 프로그램 BPRG(096)/블록 프로그램 종료 BEND(801) 3-908
블록 프로그램 일시 정지 BPPS(811)/블록 프로그램 재기동 BPRS(812) 3-910
조건부 종료 EXIT(806)/조건부 종료(부정) EXIT NOT(806) 3-912
조건 분기 블록 IF(802)/조건 분기 블록(부정) IF NOT(802)/
조건 분기 거짓 블록 ELSE(803)/조건 분기 블록 종료 IEND(804) 3-914
1스캔 조건 대기 WAIT(805)/1스캔 조건 대기(부정) WAIT NOT(805) 3-917
타이머 대기 TIMW(813)/TIMWX(816) 3-920
카운터 대기 CNTW(814)/CNTWX(818) 3-923
고속 타이머 대기 TMHW(815)/TMHWX(817) 3-925
반복 블록 LOOP(809)/반복 블록 종료 LEND(810)/
반복 블록 종료(부정) LEND NOT(810) 3-927

문자열 처리 명령

문자열 전송 MOV$(664) 3-934
문자열 연결 +$(656) 3-936
왼쪽에서 문자열 가져오기 LEFT$(652)/오른쪽에서 문자열 가져오기 RGHT$(653) 3-938
임의 위치에서 문자열 가져오기 MID$(654) 3-940
문자열 검색 FIND$(660) 3-942
문자열 길이 검출 LEN$(650) 3-944
문자열 치환 RPLC$(661) 3-946
문자열 삭제 DEL$(658) 3-948
문자열 교환 XCHG$(665) 3-950
문자열 클리어 CLR$(666) 3-952
문자열 삽입 INS$(657) 3-954
문자열 비교 =$, <>$, <$, <=$, >$, >=$(670~675) 3-956

태스크 제어 명령

태스크 실행 기동 TKON(820)/태스크 실행 대기 TKOF(821) 3-962

기종 변환용 명령

블록 전송 XFERC(565) 3-970
데이터 분배 DISTC(566) 3-972
데이터 추출 COLLC(567) 3-975
비트 전송 MOVBC(568) 3-979
비트 카운터 BCNTC(621) 3-981

펑션 블록용 특수 명령

변수 종류 취득 GETID(286) 3-984

SFC 관련 명령

스텝 활성화 명령 SA(784)/스텝 비활성화 명령 SE(785) 3-990
스텝 타이머 현재값 읽기 명령 TSR(780)/
스텝 타이머 현재값 변경 명령 TSW(781) 3-992
SFC 태스크 실행 명령 SFCON(789)/SFC 태스크 종료 명령 SFCOFF(790) 3-994
SFC 태스크 정지 명령(리셋 있음) SFCPR(793)/
SFC 태스크 정지 명령(리셋 없음) SFCPRN(791) 3-996

제4장 명령 처리 시간/스텝 수 목록

4-1 CJ 시리즈(CJ2) 명령 처리 시간/스텝 수 목록 4-2
4-2 CJ 시리즈(CJ1) 명령 처리 시간/스텝 수 목록 4-20
4-3 CS 시리즈 명령 처리 시간/스텝 수 목록 4-42

부록

명령 목록(펑션 번호 순서) 부록-2
ABC 순서 명령 목록 부록-11

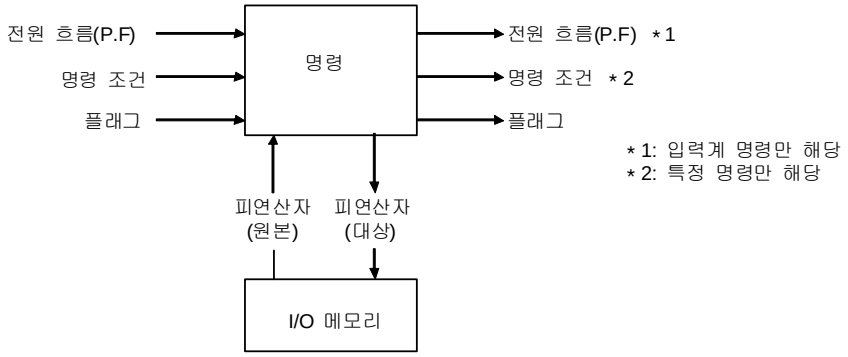
제1장

명령어의 기본 지식

1-1 명령어의 기본 지식

■ 명령의 구조

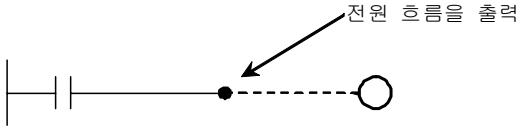
프로그램은 명령의 조합으로 구성됩니다.
명령은 다음과 같이 입력과 출력의 구조로 이루어집니다.



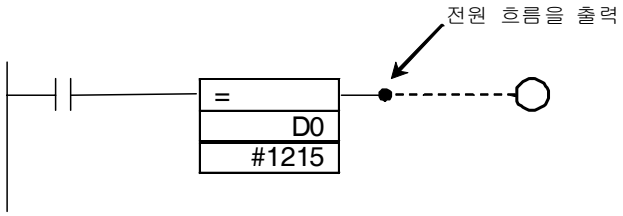
● 전원 흐름(P.F)

전원 흐름이란 평상시에 프로그램을 실행할 때 명령 기능의 실행/비실행을 제어하기 위한 정보입니다.
래더 프로그램에서는 모선에서의 신호 흐름을 나타냅니다.

- a) **입력부를 구성하는 입력계 명령의 경우**
  - 논리 시작을 나타내는 LD형 명령: 전원 흐름을 출력합니다.

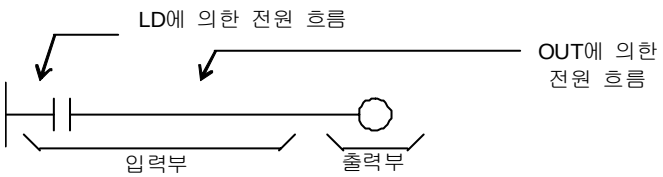


  - 다음 단계 연결형 명령: 입력 조건으로 전원 흐름을 입력하고, 다음 단계 연결형 명령 또는 출력 명령에 전원 흐름을 출력합니다.



- b) **출력부를 구성하는 출력계 명령의 경우**

  전원 흐름을 입력 조건으로 입력하여 각 기능을 실현합니다.



● 명령 조건

명령 조건이란 다음과 같은 특정 명령의 출력에서 전체 명령 실행과 관련된 특수한 상태입니다. 명령 실행 판정 시 전원 흐름(P.F.)보다 우선됩니다. 또한 명령 조건에 따라 실행되지 않는 명령이나 동작이 다른 명령이 있습니다.
명령 조건은 각 태스크의 선두에서 리셋(해제)됩니다. (따라서 태스크를 전환하면 리셋됩니다.)

한 조건에 대해 설정하는 명령과 해제하는 명령을 한 쌍으로 사용합니다.
이 한 쌍의 명령은 동일 태스크 내에 배치해야 합니다.

명령 조건	내용	설정 명령	해제 명령
인터록 중	프로그램의 일부에 인터록을 작동시킨 상태(출력 릴레이 OFF, 타이머 리셋, 카운터 유지 등의 특수한 상태)	IL 명령	ILC 명령
BREAK 중	FOR~NEXT 루프의 반복을 도중에 종료하는 상태(해당 루프의 NEXT까지의 명령을 실행하지 않음)	BREAK	NEXT
	JMP0~JME0의 점프를 실행하는 상태	JMP0	JME0
블록 프로그램 실행 중	BPRG~BEND까지의 프로그램을 실행하는 상태	BPRG	BEND

● 플래그

플래그란 명령 간의 인터페이스로 사용하는 비트 정보입니다.

입력 플래그		출력 플래그	
플래그명	내용	플래그명	내용
캐리(CY) 플래그	데이터 이동 명령, 가감산 명령에서 명시적이지 않은 피연산자로 사용되는 조건 플래그의 일종입니다.	조건 플래그	명령 실행 결과에 따라 갱신되는 플래그 및 상시 ON/상시 OFF 플래그의 총칭입니다. 사용자 프로그램에서는 주소 대신 P_On, P_Off, P_ER, P_CY, P_EQ 등의 라벨을 사용하여 지정합니다.
명령 전용 입력 플래그	FPD 명령용 티칭 플래그, 네트워크 통신 실행 가능 플래그 등	명령 전용 출력 플래그	메모리 카드 명령 실행 중 플래그, MSG 실행 완료 플래그 등

● **피연산자**

피연산자란 명령이 I/O 메모리 영역의 데이터 또는 상수를 지정하기 위해 미리 개별적으로 가지고 있는 파라미터를 말합니다. 피연산자 안에 주소 또는 상수를 입력하면 명령에서 연산을 실행할 수 있습니다.

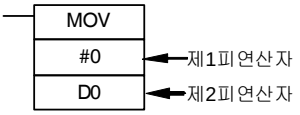
피연산자는 원본 피연산자, 대상 연산자 및 번호로 분류됩니다.

예:



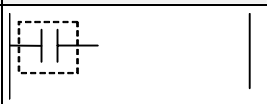
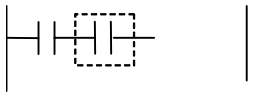
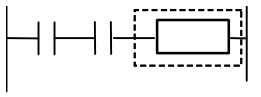
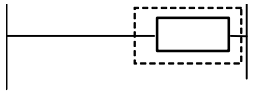
피연산자 분류		피연산자 기호	내용	
원본 피연산자	데이터를 읽어올 주소 또는 데이터 자신을 저장합니다.	S	원본 피연산자	제어 데이터(C) 이외의 원본 피연산자
		C	제어 데이터	원본 피연산자 중에 특히 1비트 또는 복수의 비트에 의해 여러 가지 의미를 갖는 데이터의 복합체
대상 피연산자	데이터를 기술할 주소를 저장합니다.	D	−	
번호	점프, 서브루틴 등, 각각의 명령에 특정 번호가 지정된 경우 해당 번호를 저장합니다.	N	−	

주: 기술된 위치에 따라 위에서부터 제1피연산자, 제2피연산자 등으로 부릅니다.



■ 명령의 배치 위치와 입력 조건

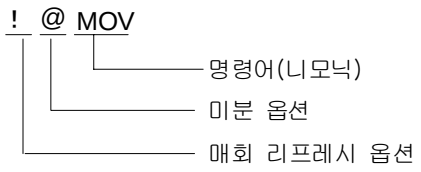
명령에는 다음과 같이 배치할 수 있는 위치가 있습니다.
또한 명령에는 입력 조건이 필요한 명령과 불필요한 명령이 있습니다.
각 명령별 내용은 제2장 '명령 목록'을 참조하십시오.

분류		배치 가능 위치	입력 조건 필요/불필요	그림	명령의 예
입력계	논리 시작형(LD형)	좌측 모선에 직접 연결 또는 회로 블록의 시작점	필요 없음		LD, LD TST, LD>(등의 심볼 비교)
	다음 단계 연결형	논리 시작형과 출력형의 중간 위치	필요		AND, OR, AND TST, AND>(등의 심볼 비교), UP, DOWN, NOT 등
출력계		우측 모선에 직접 연결	필요		OUT, MOV 등 많은 명령
			필요 없음		END, JME, FOR, ILC 등

■ 동작 옵션

명령에 다음의 동작 옵션을 추가하면 실행 조건을 미분형으로 수행하거나 명령 실행 시 데이터 리프레시(매회 리프레시)를 수행할 수 있습니다.

동작 옵션		옵션 기호	내용
미분 옵션	기동	@	입력 기동 미분형 명령이라고 합니다.
	종료	%	입력 종료 미분형 명령이라고 합니다.
매회 리프레시 옵션		!	명령 실행 시, 피연산자로 지정한 입출력 영역의 데이터를 I/O 리프레시합니다.



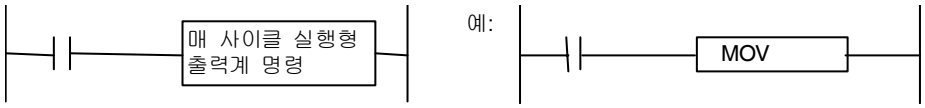
■ 실행 조건

기본 명령이나 응용 명령에는 다음의 두 타입이 있습니다.

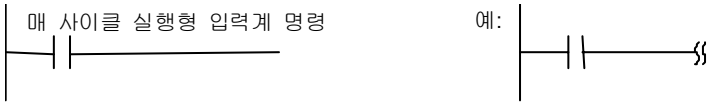
- 매 사이클 실행형
- 입력 미분형(1사이클 실행)

● 매 사이클 실행형 명령(옵션이 없는 일반 명령)

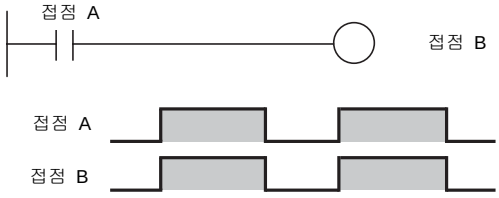
- 출력계 명령(입력 조건이 필요한 명령만 해당)인 경우
입력 조건이 성립(ON 또는 OFF)하는 동안 사이클마다 실행하는 명령입니다.



- 입력계 명령(논리 시작형 및 다음 단계 연결형 명령)인 경우
사이클마다 접점 읽기/비교/비트 테스트 등의 처리를 실행하고, 결과가 ON이면 입력 조건(전원 흐름)을 출력하는 명령입니다.



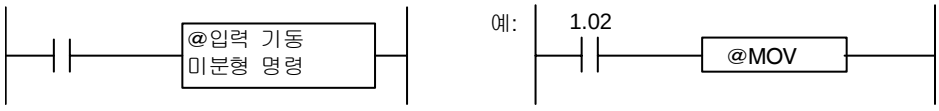
■시간 차트



● 입력 미분형 명령

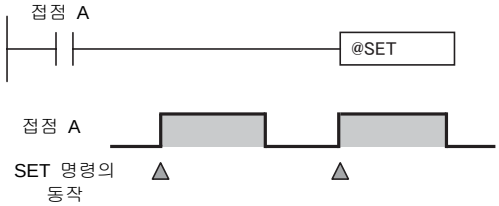
- 입력 기동 미분형 명령(명령어 앞에 '@' 옵션이 부가되는 명령)

- 출력계 명령인 경우
  입력 조건이 기동(OFF→ON)일 때만 실행되는 명령입니다. 다음 사이클에서는 실행되지 않습니다.

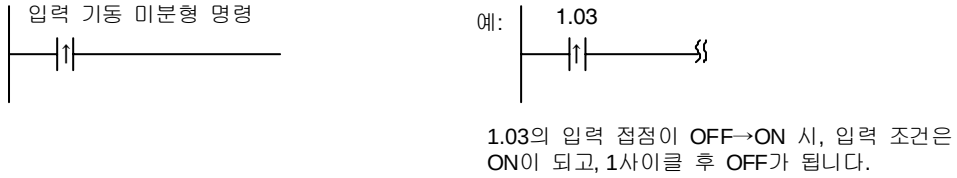


1.02가 OFF→ON 시에 1회만 MOV 명령을 실행합니다.

■시간 차트

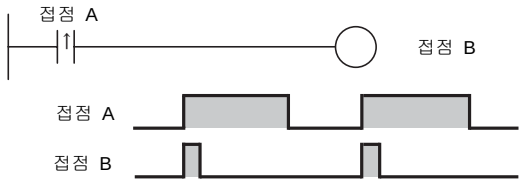


- 입력계 명령(논리 시작형 및 다음 단계 연결형 명령)인 경우
  사이클마다 접점 읽기/비교/비트 테스트 등의 처리를 실행하고, 결과가 OFF→ON으로 변하면 입력 조건(전원 흐름)을 출력하는 명령입니다.
  출력하는 입력 조건은(1사이클 ON한 후) 다음 사이클에서 OFF가 됩니다.

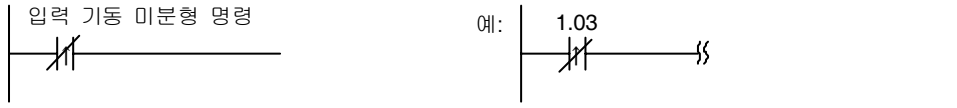


1.03의 입력 접점이 OFF→ON 시, 입력 조건은 ON이 되고, 1사이클 후 OFF가 됩니다.

■시간 차트

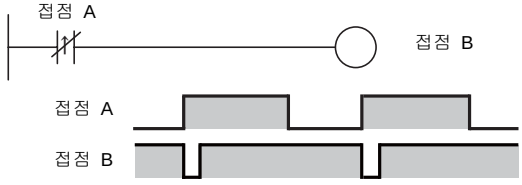


・ 입력계 명령(논리 시작형 및 다음 단계 연결형 명령)인 경우
사이클마다 접점 읽기/비교/비트 테스트 등의 처리를 실행하고, 결과가 OFF→ON으로 변하는 경우를 제외하고 입력 조건(전원 흐름)을 출력하는 명령입니다.



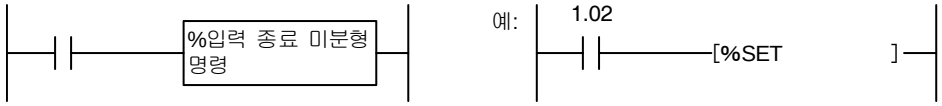
1.03의 입력 접점이 OFF→ON 시, OFF가 되고 1사이클 후에 ON이 됩니다.

■시간 차트



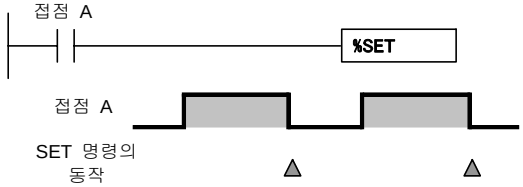
・ 입력 종료 미분형 명령(명령어 앞에 '%' 옵션이 부가되는 명령)

・ 출력계 명령인 경우
입력 조건이 종료(ON→OFF)일 때만 실행되는 명령입니다. 다음 사이클에서는 실행되지 않습니다.

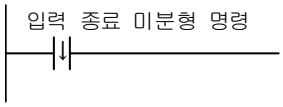


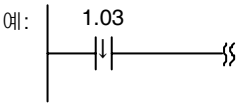
1.02가 ON→OFF 시, 1회만 SET 명령이 실행됩니다.

■시간 차트



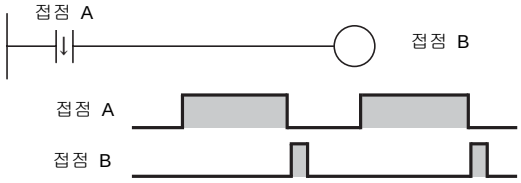
・ 입력계 명령(논리 시작형 및 다음 단계 연결형 명령)인 경우
사이클마다 접점 읽기/비교/비트 테스트 등의 처리를 실행하고, 결과가 ON→OFF로 변하면 입력 조건(전원 흐름)을 출력하는 명령입니다.
출력하는 입력 조건은(1사이클 ON한 후) 다음 사이클에서 OFF가 됩니다.





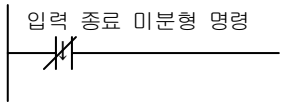
1.03의 입력 접점이 ON→OFF 시, ON이 되고, 1사이클 후에 OFF가 됩니다.

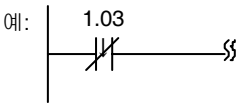
■시간 차트



주: 입력 종료 미분형의 옵션(%)은 입력 기동 미분 명령과 달리 LD, AND, OR, SET, RSET 명령에만 부가할 수 있습니다. 다른 명령에서 입력 종료 미분을 실행 조건으로 하고 싶은 경우는 DIFD 명령 또는 DOWN 명령과 조합하여 사용하십시오.
또한 NOT이 있는 명령은 CS1-H/CJ1-H/CJ1M/CS1D CPU 유니트만 부가할 수 있습니다.

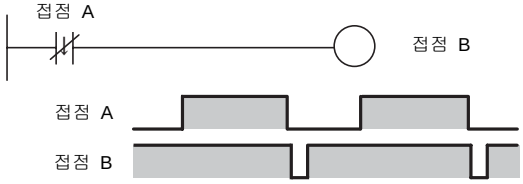
- 입력계 명령(논리 시작형 및 다음 단계 연결형 명령)인 경우
사이클마다 접점 읽기/비교/비트 테스트 등의 처리를 실행하고, 결과가 ON→OFF로 변하는 경우를 제외하고 입력 조건(전원 흐름)을 출력하는 명령입니다.





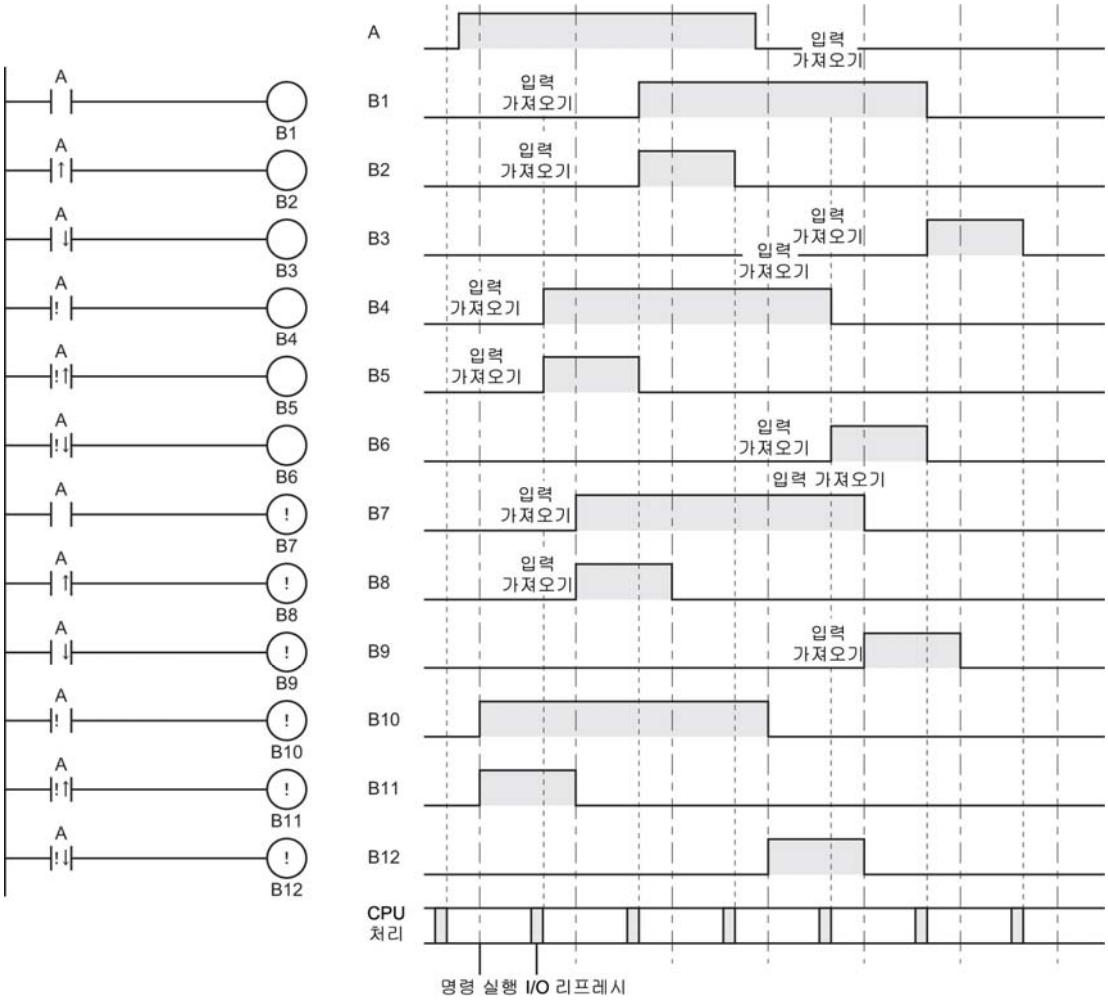
1.03의 입력 조건이 ON→OFF시, OFF가 되고, 1사이클 후에 입력 조건이ON이 됩니다.

■시간 차트



■ 각 입출력 명령의 동작 타이밍

LD 명령과 OUT 명령으로 구성된 프로그램에 대해 각 명령의 동작 차이를 타이밍 차트로 나타냅니다.



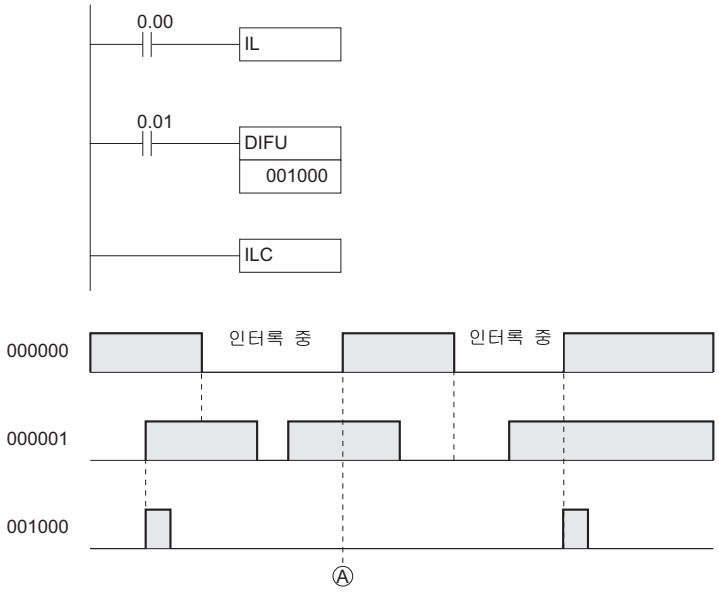
● 미분 명령

· 미분 명령은 이전 값이 ON인지, OFF인지의 플래그를 내부에 가지고 있습니다. 운전을 시작할 때는 기동 미분 명령(DIFU/@ 표시가 있는 명령)의 이전 값 플래그가 항상 ON으로, 종료 미분 명령(DIFD/% 표시가 있는 명령)의 이전 값 플래그는 항상 OFF로 설정되어 있습니다. 따라서 운전을 시작할 때 갑자기 미분 출력이 되는 경우는 없습니다.

· 기동 미분 명령(DIFU/@ 표시가 있는 명령)은 이전 값의 플래그가 OFF일 때만 입력이 ON이 된 시점에서 ON 출력을 합니다.

· IL-ILC 명령 내에서 사용할 때

다음의 예에서 미분 명령의 이전 값 플래그는 IL 직전의 값을 유지하며 인터록 중에는 갱신되지 않으므로 A 시점에서 미분 출력되지 않습니다.



· JMP-JME 명령 내에서 사용할 때

IL 명령과 동일하게 미분 명령의 이전 값 플래그는 JMP 명령 직전의 값을 유지하며 점프한 경우는 갱신되지 않습니다.

· 종료 미분 명령(DIFD/% 표시가 있는 명령)은 이전 값의 플래그가 ON일 때만 입력이 OFF된 시점에서 ON 출력을 합니다.

· 기동 미분, 종료 미분 명령은 모두 다음 번 사이클에서 출력을 OFF로 합니다.

참 고 **P_On(상시 ON 플래그), A200.11(운전 시작 시 1사이클 ON 플래그)을 입력 기동 미분형 명령의 입력 릴레이로 사용하지 마십시오.**
또한 P_Off(상시 OFF 플래그)를 입력 종료 미분형 명령의 입력 릴레이로 사용하지 마십시오.
두 경우 모두 명령이 실행되지 않습니다.

1-2 피연산자 지정 방법

■ I/O 메모리 영역 지정 방법

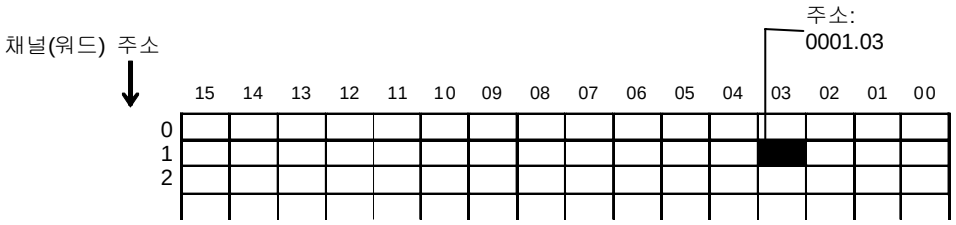
● 비트 주소 지정 방법

□□□□.□□
- □□□□: 채널(워드) 주소를 나타냅니다.
- □□: 비트 주소(00~15)를 나타냅니다.

예: 입출력 릴레이

1CH의 비트 03 표시 방법

1. 03
- 1: 채널(워드) 주소: 1CH
- 03: 비트 주소: 03

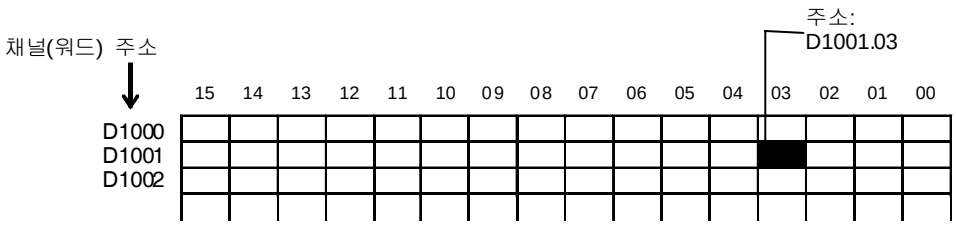


CJ2 CPU에서는 데이터 메모리(DM) 및 확장 데이터 메모리(EM)의 비트 주소를 지정할 수 있습니다.

예: DM 영역

D1001의 비트 03 표시 방법

D1001.03
- D1001: 채널(워드) 주소: D1001
- 03: 비트 주소: 03



● 채널(워드) 주소 지정 방법

□□□□CH
- □□□□: 채널(워드) 주소를 나타냅니다.

예: 입출력 릴레이

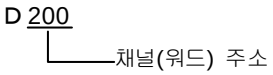
10CH(비트 00~15) 표시 방법

10 CH
- 10: 채널(워드) 주소

데이터 메모리는 D□□□□□ 또는 E□□□□□□로 표시합니다.

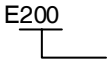
예: 데이터 메모리(DM)

D200 표시 방법

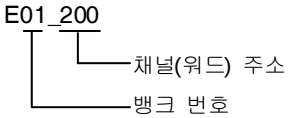


예: 확장 데이터 메모리(EM)

현재 뱅크의 200CH 표시 방법



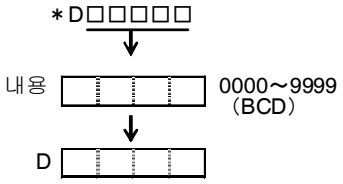
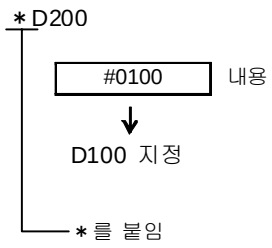
뱅크 번호 01의 200CH 표시 방법



■ 데이터 지정 방법

데이터 지정 방법	내용	예	명령에서 사용한 예
비트 주소 직접 지정	1비트(접점)를 지정하기 위해 채널 주소+비트 주소로 직접 지정합니다. □□□□.□□ 비트 주소 00～15 채널 주소 주: 타임업 플래그, 카운트 업 플래그, 태스크 플래그는 채널 주소/비트 주소를 구별하지 않습니다.	1.02 비트 주소 02 채널 주소 1CH	1.02
채널 주소 직접 지정	16비트(1워드=1채널)를 지정하기 위해 채널 주소를 직접 지정합니다. □□□□ 채널 주소	3 채널 주소 3CH D200 채널 주소 D200CH	MOV 3 D200
비트 주소 오프셋 지정(*1)	지정한 비트 주소를 선두로 하고 그 주소부터의 오프셋을 [ ] 안에 지정합니다. □□□□.□□ [□□] 오프셋 주소 상수 0～15 또는 CH 지정 I/O 메모리 주소 시작 비트 주소 시작 비트 주소에는 변수도 지정할 수 있습니다. 단, 주소와 변수 둘 다 영역 종류는 H/W/DM/EM 영역만 가능합니다. 오프셋 주소에는 상수 또는 CH 지정 I/O 메모리 주소를 지정할 수 있습니다. CH 지정 I/O 메모리 주소의 경우 해당 주소 내의 값을 간접 지정합니다.	10.00[2] 오프셋 비트 수 → 10.02 시작 비트 주소 10.00[W 0] 오프셋 비트 수 W0=&2 일 때 → 10.02 시작 비트 주소	10.00[2]
채널 주소 오프셋 지정(*1)	지정한 채널 주소를 선두로 하고 그 주소부터의 오프셋을 [ ] 안에 지정합니다. □□□□ [□□] 오프셋 주소 상수 0～ 또는 CH 지정 I/O 메모리 주소 시작 채널 주소 시작 채널 주소에는 변수도 지정할 수 있습니다. 단, 주소와 변수 둘 다 영역 종류는 H/W/DM/EM 영역만 가능합니다. 오프셋 주소에는 상수 또는 CH 지정 I/O 메모리 주소를 지정할 수 있습니다. CH 지정 I/O 메모리 주소의 경우 해당 주소 내의 값을 간접 지정합니다.	D0[2] 오프셋 CH 수 → D2 시작 채널 주소 D0[W0] 오프셋 CH 수 W0=&2 일 때 → D2 시작 채널 주소	MOV 3 D0[200]

데이터 지정 방법	내용	예	명령에서 사용한 예
DM/EM 간접 지정 BIN 모드	데이터 메모리(DM) 또는 확장 데이터 메모리(EM)의 채널 주소를 지정하려면 내용을 BIN 값(00000~32767)으로 읽어서 각 영역의 선두부터의 오프셋으로 간접 지정합니다. '@' 표시를 맨 앞에 붙입니다. @D□□□□□ 내용 [ ] 10진수 &0~32767 (16진수 #0000~7FFF) D [ ] 1) @D□□□□□의 내용이 10진수 &0~32767(16진수 #0000~7FFF)인 경우 D0~32767을 지정합니다.	@D300 [10진수 &256] 내용 (16진수 #0100) ↓ D00256 지정 @를 붙임	MOV #0001 @D300
	2) @D□□□□□의 내용이 10진수 &32768~65535(16진수 #8000~FFFF)인 경우 확장 데이터 메모리(EM)의 뱅크 번호 0에서 E0_0~E0_32767을 지정합니다.	@D300 [10진수 &32769] 내용 (16진수 #8001) ↓ E0_1 지정	
	3) @E□_□□□□□의 내용이 10진수 &0~32767(16진수 #0000~7FFF)인 경우 지정된 뱅크 번호 □에서 E□_0~E□_32767을 지정합니다.	@E1_200 [10진수 &257] 내용 (16진수 #0101) ↓ E1_257 지정	MOV #0001 @E1_200
	4) @E□_□□□□□의 내용이 10진수 &32768~65535(16진수 #8000~FFFF)인 경우 지정된 뱅크 번호 □의 다음 뱅크 번호에서 E(□+1)_0~E(□+1)_32767을 지정합니다.	@E1_200 [10진수 &32770] 내용 (16진수 #8002) ↓ E2_2 지정	
	주: BIN 모드에서 간접 지정하는 경우 데이터 메모리(DM)와 확장 데이터 메모리(EM) (뱅크 0~18Hex)를 연속된 주소로 취급합니다. @로 지정한 주소의 내용이 32767을 초과한 경우, 데이터 메모리일 때는 확장 데이터 메모리에서 뱅크 번호 0의 0을 시작점으로 간접 지정합니다. 예: 데이터 메모리(DM)의 내용이 32768인 경우 확장 데이터 메모리(EM)의 뱅크 번호 0의 E0_0을 지정합니다. 확장 데이터 메모리(EM)의 뱅크 번호 n일 때는 뱅크 번호 (n+1)의 0을 시작점으로 하여 간접 지정합니다. 예: 확장 데이터 메모리(EM)의 뱅크 2의 내용이 32768인 경우 확장 데이터 메모리(EM)의 뱅크 3의 E3_0을 지정합니다.		

데이터 지정 방법	내용	예	명령에서 사용한 예
DM/EM 간접 지정 BCD 모드	데이터 메모리(DM) 또는 확장 데이터 메모리(EM)의 채널 주소를 지정하려면 내용을 BCD 값(0000~9999)으로 읽어서 각 영역의 선두부터의 오프셋으로 간접 지정합니다. 또한 '*' 표시를 맨 앞에 붙입니다. 		MOV #0001 ＊D200

데이터 지정 방법	내용		예	명령에서 사용한 예
레지스터 직접 지정	인덱스 레지스터(IR) 또는 데이터 레지스터(DR)를 지정하려면 IR□(□: 0~15), DR□(□: 0~15)로 지정합니다.		IR0 IR1	MOVR 1.02 IR0 1CH 02비트의 I/O 메모리 실제 주소를 IR0에 저장합니다. MOVR 10 IR1 10CH의 I/O 메모리 실제 주소를 IR1에 저장합니다.
레지스터 간접 지정	간접 지정(오프셋 없음)	IR□의 내용을 I/O 메모리 실제 주소로 하는 접점 또는 채널을 지정합니다. ,IR□로 지정합니다 (접점/채널은 명령 및 피연산자로 지정).	,IR0 ,IR1	LD ,IR0 IR0의 내용을 I/O 메모리 실제 주소로 하는 접점을 로드합니다. MOV #0001 ,IR1 IR1의 내용을 I/O 메모리 실제 주소로 하는 채널에 #0001을 저장합니다.
	상수 오프셋 지정	IR□의 내용에 지정된 상수에 더하거나 뺀 값을 I/O 메모리 실제 주소로 하는 접점 또는 채널을 지정합니다. 상수 ,IR□로 지정합니다. 상수 오프셋의 범위는 −2048~+2047(10진수)이며, 명령을 실행할 때 BIN 데이터로 변환됩니다.	+5 ,IR0 31 ,IR1	LD +5 ,IR0 IR0의 내용+5를 I/O 메모리 실제 주소로 하는 접점을 로드합니다. MOV #0001 +31 ,IR1 IR1의 내용 +31을 I/O 메모리 실제 주소로 하는 채널에 #0001을 저장합니다.
	DR(데이터 레지스터) 오프셋 지정	IR□의 내용에 DR□를 더한(+) 값을 I/O 메모리 실제 주소로 하는 접점 또는 채널을 지정합니다. DR□,IR□로 지정합니다. DR(데이터 레지스터)의 값은 부호 있는 16진수로 취급합니다. 부호 있는 BIN 값이 음수인 경우 IR□의 내용을 감소시켜 이동합니다.	DR0 ,IR0 DR0 ,IR1	LD DR0 ,IR0 (IR0의 내용)+(DR0의 내용)을 I/O 메모리 실제 주소로 하는 접점을 로드합니다. MOV #0001 DR0 ,IR1 (IR1의 내용)+(DR0의 내용)을 I/O 메모리 실제 주소로 하는 채널에 #0001을 저장합니다.
	자동 증가	IR□의 내용을 I/O 메모리 실제 주소로 참조한 후 IR□의 내용을 1 또는 2씩 증가시킵니다. 1씩 증가시키는 경우: ,IR□+ 2씩 증가시키는 경우: ,IR□++ 로 지정합니다.	,IR0++ ,IR1+	LD ,IR0++ IR0의 내용을 I/O 메모리 실제 주소로 하는 접점을 로드한 후 IR0의 내용을 2 증가시킵니다. MOV #0001 ,IR1+ IR1의 내용을 I/O 메모리 실제 주소로 하는 채널에 #0001을 저장한 후 IR1의 내용을 1 증가시킵니다.
	자동 감소	IR□의 내용을 1 또는 2씩 감소시킨 후 그 값을 I/O 메모리 실제 주소로 참조합니다. 1씩 감소시키는 경우: ,−IR□ 2씩 감소시키는 경우: ,−−IR□ 로 지정합니다.	,−−IR0 ,−IR1	LD ,−−IR0 IR0의 내용을 2 감소시킨 후 그 값을 I/O 메모리 실제 주소로 하는 접점을 로드합니다. MOV #0001 ,−IR1 IR1의 내용을 1 감소시킨 후 그 값을 I/O 메모리 실제 주소로 하는 채널에 #0001을 저장합니다.

데이터 지정 방법	사용 피연산자	데이터 형식	기호	범위	명령에서 사용한 예
상수(16비트 데이터)	모든 BIN 데이터 또는 범위가 있는 BIN 데이터	부호 없는 BIN	#	#0000~#FFFF	MOV #0100 D0 D0에 16진수 #0100 (10진수 &256)을 저장합니다. + #0009 #0001 D1 D1에 16진수 #000A (10진수 &10)를 저장합니다.
		부호 있는 10진수	±	－32768~＋32767	MOV －100 D0 D0에 10진수 －100 (16진수 #FF9C)을 저장합니다. + －9 －1 D1 D1에 10진수 －10 (16진수 #FFF6)을 저장합니다.
		부호 없는 10진수	&	&0~&65535	MOV &256 D0 D0에 10진수 &256(16진수 #0100)을 저장합니다. + &9 &1 D1 D1에 10진수 &10(16진수 #000A)을 저장합니다.
	모든 BCD 데이터 또는 범위가 있는 BCD 데이터	BCD	#	#0000~#9999	MOV #0100 D0 D0에 #0100(BCD)을 저장합니다. +B #0009 #0001 D1 D1에 #0010(BCD)을 저장합니다.
상수(32비트 데이터)	모든 BIN 데이터 또는 범위가 있는 BIN 데이터	부호 없는 BIN	#	#00000000~#FFFFFFFF	MOVL #12345678 D0 D0~D1에 16진수 #12345678을 저장합니다. D1 D0 1234 5678
		부호 있는 10진수	＋	－2147483648~＋2147483647	MOVL －12345678 D0 D0~D1에 10진수 －12345678을 저장합니다.
		부호 없는 10진수	&	&0~&4294967295	MOVL &12345678 D0 D0~D1에 10진수 &12345678을 저장합니다.
	모든 BCD 데이터 또는 범위가 있는 BCD 데이터	BCD	#	#00000000~#99999999	MOVL #12345678 D0 D0~D1에 #12345678(BCD)을 저장합니다.

<table>
<tr><td>문자열
데이터</td><td>내용</td><td>기호</td><td>예</td><td>–</td></tr>
<tr><td></td><td>ASCII 코드(1바이트, 특수 문자는 제외)로 상위 바이트→하위 바이트의 순서에 따라 하위(낮은 숫자) 채널→상위 채널의 순서로 저장됩니다.
문자 수가 홀수인 경우 최종 채널의 하위 바이트의 빈 공간에 00Hex(NUL 코드)가 저장됩니다.
문자 수가 짝수인 경우에는 최종 채널＋1의 상위/하위 바이트에 0000Hex(NUL 코드 2개)가 저장됩니다.</td><td>↓</td><td>‘ABCDE’
↓
‘A’ ‘B’
‘C’ ‘D’
‘E’ NUL
‖
41 42
43 44
45 00

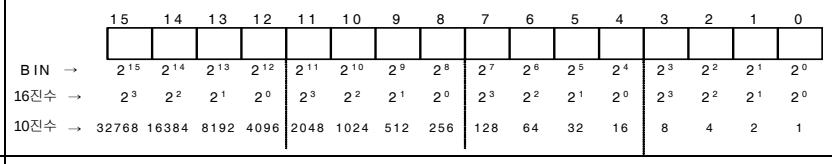
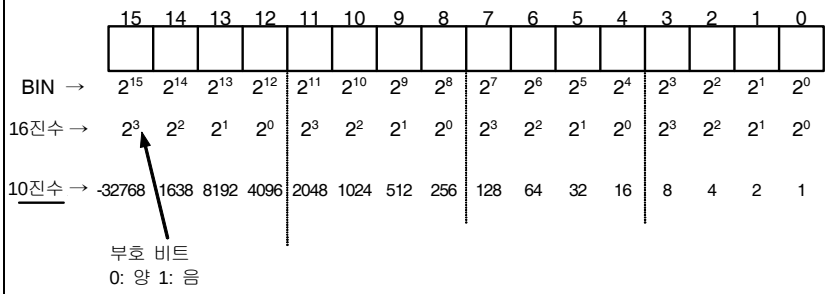
‘ABCD’
↓
‘A’ ‘B’
‘C’ ‘D’
NUL NUL
‖
41 42
43 44
00 00</td><td>MOV$ D100 D200
D100 41 42
D101 43 44
D102 45 00
↓
D200 41 42
D201 43 44
D202 45 00</td></tr>
<tr><td colspan="4">문자열로 취급되는 ASCII 코드는 다음과 같이 특수 문자를 제외한 영문자, 숫자, 가타카나, 기호입니다.</td></tr>
</table>

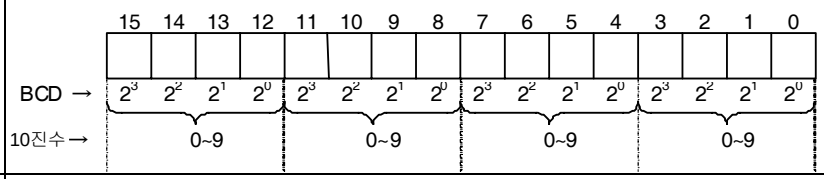
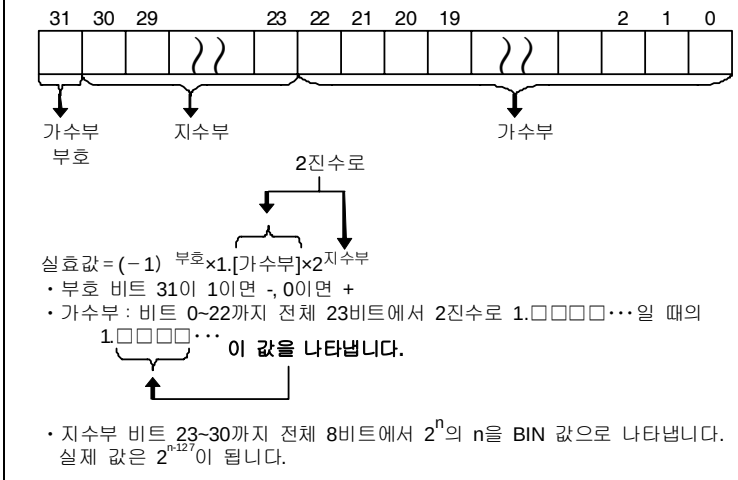
하위 4비트 \ 상위 4비트	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0			SP	0	@	P	`	p				ー	タ	ミ		
1			!	1	A	Q	a	q			。	ア	チ	ム		
2			″	2	B	R	b	r			「	イ	ツ	メ		
3			#	3	C	S	c	s			」	ウ	テ	モ		
4			$	4	D	T	d	t			、	エ	ト	ヤ		
5			%	5	E	U	e	u			・	オ	ナ	ユ		
6			&	6	F	V	f	v			ヲ	カ	ニ	ヨ		
7			′	7	G	W	g	w			ァ	キ	ヌ	ラ		
8			(	8	H	X	h	x			ィ	ク	ネ	リ		
9			)	9	I	Y	i	y			ゥ	ケ	ノ	ル		
A			*	:	J	Z	j	z			ェ	コ	ハ	レ		
B			+	;	K	[	k	{			ォ	サ	ヒ	ロ		
C			,	<	L	¥	l	|			ャ	シ	フ	ワ		
D			−	=	M	]	m	}			ュ	ス	ヘ	ン		
E			.	>	N	^	n	~			ョ	セ	ホ	゛		
F			/	?	O	_	o				ッ	ソ	マ	°		

*1 CJ2 CPU 유니트만 해당

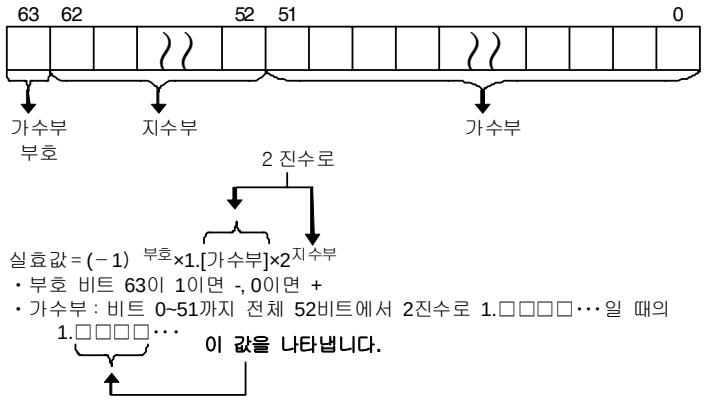
1-3 데이터 형식

CJ 시리즈에서 사용 가능한 데이터의 형식은 다음과 같습니다.

<table>
<tr><th>종류</th><th>데이터 형식</th><th>10진수</th><th>16진수
4자리</th></tr>
<tr><td>부호 없는
BIN
데이터</td><td> 
</td><td>&0
~
65535</td><td>#0000~
FFFF</td></tr>
<tr><td>부호 있는
BIN
데이터</td><td> 

최상위 비트를 부호 비트로 사용하는 2진수 16비트 데이터로 처리하지 않고, 그 값을 16진수 4자리로 표현합니다.
양수: 최상위 비트가 0(OFF)일 때 양의 값 또는 0을 의미합니다. 16진수 4자리 표현은 0000~7FFFHex가 됩니다.
음수: 최상위 비트가 1(ON)일 때 음의 값을 의미합니다. 16진수 4자리 표현은 8000~FFFFHex가 됩니다. 이때 음의 값(10진수)의 절대값은 2의 보수로 표현됩니다.</td><td>0~
－32768
0~
＋32767</td><td>음수:
#8000~
FFFF
양수:
#0000~
7FFF</td></tr>
<tr><td>BCD
데이터
(2진화
10진수)</td><td> 
</td><td>#0~
9999</td><td>#0000~
9999</td></tr>
<tr><td>단정도
부동
소수점
데이터</td><td> 

IEEE754 규격의 단정도를 준수합니다. 부동 소수점 변환 및 연산 명령에서만 사용합니다. CX-Programmer의 I/O 메모리 편집 및 모니터 화면에서 이 형식으로 설정/모니터링할 수 있습니다. 따라서 사용자가 이 형식에 대해 특별히 고려해야 할 필요는 없습니다. 2CH를 점유한다는 것만 고려하십시오.</td><td></td><td></td></tr>
</table>

종류	데이터 형식	10진수	16진수 4자리
배정도 부동 소수점 데이터	 IEEE754 규격의 배정도를 준수합니다. 부동 소수점 변환 및 연산 명령에서만 사용합니다. CX-Programmer의 I/O 메모리 편집 및 모니터 화면에서 이 형식으로 설정/모니터링할 수 있습니다. 따라서 사용자가 이 형식에 대해 특별히 고려해야 할 필요는 없습니다. 4CH를 점유한다는 것만 고려하십시오.		

참 고

・ 보수란?

일반적으로 기본수 x에 대한 보수는 지정된 수의 각 자리의 숫자를 x-1에서 빼고, 가장 낮은 자리에 1을 더한 수를 말합니다. (예: 7556에 대한 10의 보수는 9999 − 7556 + 1 = 2444)

보수는 덧셈으로 뺄셈을 표현하는 경우 등에 사용합니다.

예: 8954 − 7556 = 1398은 8954 + (75567에 대한 10의 보수) = 8954 + 2444 = 11398이 되며, 가장 높은 자리의 오버플로를 무시하면 뺄셈 결과인 1398을 얻을 수 있습니다.

・ 2의 보수란?

2의 보수는 기본수를 2로 한 보수입니다. 각 자리를 모두 1(= 2 − 1)로 한 값에서 원래의 수를 빼고 그 결과에 1을 더한 값입니다.

예: 2진수 1101에 대한 2의 보수는 1111(FHex) − 1101(DHex) + 1(1Hex) = 0011(3Hex)입니다. 16진수 4자리로 표현한 경우는 다음과 같습니다.

a Hex에 대한 2의 보수 b Hex는 FFFFHex − a Hex+0001Hex = b Hex입니다.

원래의 수 a Hex에서 2의 보수 b Hex를 구하려면 b Hex = 10000Hex − a Hex입니다.

예) 원래의 수 3039에서 2의 보수를 구하는 경우 10000Hex − 3039Hex = CFC7Hex입니다.

2의 보수 b Hex에서 원래의 수 a Hex를 구하는 경우도 마찬가지로 a Hex = 10000Hex − b Hex입니다.

예) 2의 보수 CFC7Hex에서 원래의 수를 구하려면 10000Hex − CFC7Hex = 3039Hex입니다.

또한 CJ 시리즈에는 NEG(2의 보수 변환)/NEGL(2의 보수 2워드 변환) 명령이 있습니다. 이 명령은 원래의 수에서 2의 보수를 구할 때 또는 2의 보수에서 원래의 수를 구할 때도 사용할 수 있습니다.

● **1CH** 데이터로 표현할 수 있는 값

숫자 (10진수)	BIN 표기			BCD 표기 (BCD 데이터)
	10진수 표기		16진수 표기	
	부호 없는 BIN 데이터	부호 있는 BIN 데이터		
1	&1	+1	#0001	#0001
2	&2	+2	#0002	#0002
3	&3	+3	#0003	#0003
4	&4	+4	#0004	#0004
5	&5	+5	#0005	#0005
6	&6	+6	#0006	#0006
7	&7	+7	#0007	#0007
8	&8	+8	#0008	#0008
9	&9	+9	#0009	#0009
10	&10	+10	#000A	#0010
11	&11	+11	#000B	#0011
12	&12	+12	#000C	#0012
13	&13	+13	#000D	#0013
14	&14	+14	#000E	#0014
15	&15	+15	#000F	#0015
16	&16	+16	#0010	#0016
:	:	:	:	:
9999	&9999	+9999	#270F	#9999
10000	&10000	+10000	#2710	표현 불가능
:	:	:	:	
32767	&32767	+32767	#7FFF	
32768	&32768	표현 불가능	#8000	
:	:		:	
65535	&65535		#FFFF	
-1	표현 불가능	-1	#FFFF	표현 불가능
:		:	:	
-32768		-32768	#8000	
-32769		표현 불가능	표현 불가능	

제2장

명령 목록

명령 목록(CPU 유니트별 사용 가능 여부)

명령 종류	명령 이름	니모닉	펑션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
시퀀스 입력 명령	로드	LD	–	논리 시작을 나타냅니다. 접점의 ON/OFF 내용을 전송합니다.	○	○	○	○	○	○	○	○	3-8
		@LD	–		○	○	○	○	○	○	○	○	
		%LD	–		○	○	○	○	○	○	○	○	
		!LD	–		○	○	○	○	○	○	×	○	
		!@LD	–		○	○	○	○	○	○	×	○	
		!%LD	–		○	○	○	○	○	○	×	○	
	로드 부정	LD NOT	–	논리 시작을 나타냅니다. 접점의 ON/OFF 내용을 반전하여 전송합니다.	○	○	○	○	○	○	○	○	3-8
		@LD NOT	–		○	○	○	○	○	○	○	×	
		%LD NOT	–		○	○	○	○	○	○	○	×	
		!LD NOT	–		○	○	○	○	○	○	×	○	
		!@LD NOT	–		○	○	○	○	○	○	×	×	
		!%LD NOT	–		○	○	○	○	○	○	×	×	
	논리곱	AND	–	접점의 ON/OFF 내용과 이전 입력 조건의 논리곱을 계산합니다.	○	○	○	○	○	○	○	○	3-10
		@AND	–		○	○	○	○	○	○	○	○	
		%AND	–		○	○	○	○	○	○	○	○	
		!AND	–		○	○	○	○	○	○	×	○	
		!@AND	–		○	○	○	○	○	○	×	○	
		!%AND	–		○	○	○	○	○	○	×	○	
	부정 논리곱	AND NOT	–	접점의 ON/OFF 내용을 반전하여 이전 입력 조건과의 논리곱을 계산합니다.	○	○	○	○	○	○	○	○	3-10
		@AND NOT	–		○	○	○	○	○	○	○	×	
		%AND NOT	–		○	○	○	○	○	○	○	×	
		!AND NOT	–		○	○	○	○	○	○	×	○	
		!@AND NOT	–		○	○	○	○	○	○	×	×	
		!%AND NOT	–		○	○	○	○	○	○	×	×	
	논리합	OR	–	접점의 ON/OFF 내용과 이전 입력 조건과의 논리합을 계산합니다.	○	○	○	○	○	○	○	○	3-12
		@OR	–		○	○	○	○	○	○	○	○	
		%OR	–		○	○	○	○	○	○	○	○	
		!OR	–		○	○	○	○	○	○	×	○	
		!@OR	–		○	○	○	○	○	○	×	○	
		!%OR	–		○	○	○	○	○	○	×	○	
	부정 논리합	OR NOT	–	접점의 ON/OFF 내용을 반전하여 이전 입력 조건과의 논리합을 계산합니다.	○	○	○	○	○	○	○	○	3-12
		@OR NOT	–		○	○	○	○	○	○	○	×	
		%OR NOT	–		○	○	○	○	○	○	○	×	
		!OR NOT	–		○	○	○	○	○	○	×	○	
		!@OR NOT	–		○	○	○	○	○	○	×	×	
		!%OR NOT	–		○	○	○	○	○	○	×	×	
	로드 논리곱	AND LD	–	회로 블록 간의 논리곱을 계산합니다.	○	○	○	○	○	○	○	○	3-14
	로드 논리합	OR LD	–	회로 블록 간의 논리합을 계산합니다.	○	○	○	○	○	○	○	○	3-14
	부정	NOT	520	입력 조건을 반전합니다.	○	○	○	○	○	○	○	○	3-17
	P.F. 기동 미분	UP	521	입력 신호가 기동될 때(OFF→ON) 1사이클 ON하고 다음 단계에 연결합니다.	○	○	○	○	○	○	○	○	3-18
	P.F. 종료 미분	DOWN	522	입력 신호가 종료될 때(ON→OFF) 1사이클 ON하고 다음 단계 이후로 연결합니다.	○	○	○	○	○	○	○	○	3-18
	LD 비트 테스트	LD TST	350	지정 비트가 1일 때 다음 단계에 LD 연결합니다.	○	○	○	○	○	○	○	○	3-20
	LD 비트 테스트 부정	LD TSTN	351	지정 비트가 0일 때 다음 단계에 LD 연결합니다.	○	○	○	○	○	○	○	○	3-20
	AND 비트 테스트	AND TST	350	지정 비트가 1일 때 다음 단계에 AND 연결합니다.	○	○	○	○	○	○	○	○	3-22
	AND 비트 테스트 부정	AND TSTN	351	지정 비트가 0일 때 다음 단계에 AND 연결합니다.	○	○	○	○	○	○	○	○	3-22
	OR 비트 테스트	OR TST	350	지정 비트가 1일 때 다음 단계에 OR 연결합니다.	○	○	○	○	○	○	○	○	3-24
	OR 비트 테스트 부정	OR TSTN	351	지정 비트가 0일 때 다음 단계에 OR 연결합니다.	○	○	○	○	○	○	○	○	3-24

명령 종류	명령 이름	니모닉	평션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
시퀀스 출력 명령	출력	OUT	–	논리 연산 처리 결과를 릴레이 출력합니다.	○	○	○	○	○	○	○	○	3-28
		!OUT	–		○	○	○	○	○	○	×	○	
	부정 출력	OUT NOT	–	논리 연산 처리 결과를 반전하여 릴레이 출력합니다.	○	○	○	○	○	○	○	○	3-28
		!OUT NOT	–		○	○	○	○	○	○	×	○	
	일시 기억 릴레이	TR	–	TR(일시 기억 릴레이)은 니모닉 프로그램에서 회로 중간의 ON/OFF 상태를 일시 기억할 때 사용합니다.	○	○	○	○	○	○	○	○	3-30
	유지	KEEP	011	래치 릴레이(자체 유지)의 동작을 수행합니다.	○	○	○	○	○	○	○	○	3-32
		!KEEP			○	○	○	○	○	○	×	○	
	기동 미분	DIFU	013	입력 신호가 기동될 때(OFF→ON) 지정한 접점을 1사이클 ON합니다.	○	○	○	○	○	○	○	○	3-36
		!DIFU			○	○	○	○	○	○	×	○	
	종료 미분	DIFD	014	입력 신호가 종료될 때(ON→OFF) 지정한 접점을 1사이클 ON합니다.	○	○	○	○	○	○	○	○	3-38
		!DIFD			○	○	○	○	○	○	×	○	
	세트	SET	–	입력 조건이 ON일 때 지정된 접점을 ON으로 설정합니다. 이후, 입력 조건의 ON/OFF에 관계 없이 지정된 접점은 ON 상태를 그대로 유지합니다.	○	○	○	○	○	○	○	○	3-40
		@SET	–		○	○	○	○	○	○	○	○	
		%SET	–		○	○	○	○	○	○	○	○	
		!SET	–		○	○	○	○	○	○	×	○	
		!@SET	–		○	○	○	○	○	○	×	○	
		!%SET	–		○	○	○	○	○	○	×	○	
	리셋	RSET	–	입력 조건이 ON일 때 지정된 접점을 OFF로 리셋합니다. 이후, 입력 조건의 ON/OFF에 관계 없이 지정된 접점은 OFF 상태를 그대로 유지합니다.	○	○	○	○	○	○	○	○	3-40
		@RSET	–		○	○	○	○	○	○	○	○	
		%RSET	–		○	○	○	○	○	○	○	○	
		!RSET	–		○	○	○	○	○	○	×	○	
		!@RSET	–		○	○	○	○	○	○	×	○	
		!%RSET	–		○	○	○	○	○	○	×	○	
	다중 비트 세트	SETA	530	연속으로 지정된 비트 수 분량의 비트를 ON합니다.	○	○	○	○	○	○	○	○	3-42
		@SETA			○	○	○	○	○	○	○	○	
	다중 비트 리셋	RSTA	531	연속으로 지정된 비트 수 분량의 비트를 OFF합니다.	○	○	○	○	○	○	○	○	3-42
		@RSTA			○	○	○	○	○	○	○	○	
	1비트 세트	SETB	532	입력 조건이 ON일 때 지정된 CH의 지정된 접점을 ON으로 설정합니다.	○	○	○	○	○	○	○	×	3-44
		@SETB			○	○	○	○	○	○	○	×	
		!SETB			○	○	○	○	○	○	×	×	
		!@SETB			○	○	○	○	○	○	×	×	
	1비트 리셋	RSTB	533	입력 조건이 ON일 때 지정된 CH의 지정된 접점을 OFF하고 리셋합니다.	○	○	○	○	○	○	○	×	3-44
		@RSTB			○	○	○	○	○	○	○	×	
		!RSTB			○	○	○	○	○	○	×	×	
		!@RSTB			○	○	○	○	○	○	×	×	
	1비트 출력	OUTB	534	논리 연산 처리 결과(입력 조건)를 지정된 CH의 지정된 비트에 출력합니다.	○	○	○	○	○	○	○	×	3-46
		@OUTB			○	○	○	○	○	○	○	×	
		!OUTB			○	○	○	○	○	○	×	×	
시퀀스 제어 명령	종료	END	001	한 프로그램의 종료를 나타냅니다.	○	○	○	○	○	○	○	○	3-52
	기능 없음	NOP	000	아무 기능도 없는 명령입니다.	○	○	○	○	○	○	○	○	3-53
	인터록	IL	002	입력 조건이 OFF이면 IL 명령에서 ILC 명령까지의 출력을 인터록합니다.	○	○	○	○	○	○	○	○	3-54
	인터록 클리어	ILC	003	인터록의 범위의 종료를 나타냅니다.	○	○	○	○	○	○	○	○	3-54
	멀티 인터록 (미분 플래그 유지형)	MILH	517	입력 조건이 OFF이면 MILH 명령에서 MILC 명령까지의 출력을 인터록합니다.	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-57
	멀티 인터록 (미분 플래그 비유지형)	MILR	518	입력 조건이 OFF이면 MILR 명령에서 MILC 명령까지의 출력을 인터록합니다.	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-57
	멀티 인터록 클리어	MILC	519	인터록 번호가 동일한 MILH 또는 MILR 명령에 의한 멀티 인터록 범위의 종료를 나타냅니다.	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-57
	점프	JMP	004	입력 조건이 OFF일 때 JME 명령까지 직접 점프합니다.	○	○	○	○	○	○	○	○	3-66

명령 종류	명령 이름	니모닉	펑션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
시퀀스 제어 명령	점프 종료	JME	005	JMP 또는 CJP 명령에 의한 점프의 종료 위치를 나타냅니다.	○	○	○	○	○	○	○	○	3-66
	조건부 점프	CJP	510	입력 조건이 ON일 때 JME 명령까지 직접 점프합니다.	○	○	○	○	○	○	○	○	3-69
	조건부 부정 점프	CJPN	511	점프 조건이 OFF일 때 JME 명령까지 직접 점프합니다.	○	○	○	○	○	○	○	○	3-69
	복수 점프	JMP0	515	JMP0 명령의 입력 조건이 OFF일 때 JMP0의 다음 명령에서 JME0 명령까지 NOP 처리합니다.	○	○	○	○	○	○	○	○	3-72
	복수 점프 종료	JME0	516	JMP0 명령에 의한 점프의 종료 위치를 나타냅니다.	○	○	○	○	○	○	○	○	3-72
	반복 시작	FOR	512	FOR~NEXT 사이의 프로그램을 무조건 지정된 횟수만큼 반복 실행한 후 NEXT 명령 이후의 명령을 실행합니다.	○	○	○	○	○	○	○	○	3-74
	반복 종료	NEXT	513	FOR-NEXT 루프의 종료를 나타냅니다.	○	○	○	○	○	○	○	○	3-74
	루프 중단	BREAK	514	FOR-NEXT 루프를 중단하고 NEXT 명령까지 NOP 처리합니다.	○	○	○	○	○	○	○	○	3-77
타이머/카운터 명령	100ms 타이머	TIM	–	감산식 ON 딜레이 0.1초 단위의 타이머 동작을 합니다.	○	○	○	○	○	○	○	○	3-88
		TIMX	550		○	○	○	○	○	○	○	×	3-88
	10ms 타이머 (고속 타이머)	TIMH	015	감산식 ON 딜레이 10ms(0.01초) 단위의 타이머 동작을 합니다.	○	○	○	○	○	○	○	○	3-91
		TIMHX	551		○	○	○	○	○	○	○	×	3-91
	1ms 타이머 (초고속 타이머)	TMHH	540	감산식 고속 ON 딜레이 1ms 단위의 타이머의 동작을 나타냅니다.	○	○	○	○	○	○	○	○	3-94
		TMHHX	552		○	○	○	○	○	○	○	×	3-94
	0.1ms 타이머	TIMU	541	감산식 ON 딜레이 0.1ms 단위의 타이머의 동작을 나타냅니다.	○	○	CJ1-H-R만 해당	×	×	×	×	×	3-96
		TIMUX	556		○	○	CJ1-H-R만 해당	×	×	×	×	×	3-96
	0.01ms 타이머	TMUH	544	감산식 ON 딜레이 0.01ms 단위의 타이머의 동작을 나타냅니다.	○	○	CJ1-H-R만 해당	×	×	×	×	×	3-98
		TMUHX	557		○	○	CJ1-H-R만 해당	×	×	×	×	×	3-98
	적산 타이머	TTIM	087	적산식 ON 딜레이 0.1초 단위의 타이머 동작을 합니다.	○	○	○	○	○	○	○	○	3-100
		TTIMX	555		○	○	○	○	○	○	○	×	3-100
	장시간 타이머	TIML	542	감산식 ON 딜레이 100ms 타이머입니다.	○	○	○	○	○	○	○	○	3-103
		TIMLX	553		○	○	○	○	○	○	○	×	3-103
	멀티 출력 타이머	MTIM	543	8점의 임의 타임업 플래그 값을 얻을 수 있는 누적식 ON 딜레이 0.1초 단위의 타이머입니다.	○	○	○	○	○	○	○	○	3-106
		MTIMX	554		○	○	○	○	○	○	○	×	3-106
	카운터	CNT	–	감산 카운터의 동작을 수행합니다.	○	○	○	○	○	○	○	○	3-109
		CNTX	546		○	○	○	○	○	○	○	×	3-109
	가역 카운터	CNTR	012	가감산 카운터 동작을 합니다.	○	○	○	○	○	○	○	○	3-112
		CNTRX	548	가산 카운트 입력의 기동으로 가산을 실행하고 감산 카운트 입력의 기동으로 감산을 실행합니다.	○	○	○	○	○	○	○	×	3-112
	타이머/카운터 리셋	CNR/ @CNR	545	지정 범위의 타이머/카운터의 업 플래그를 리셋합니다.	○	○	○	○	○	○	○	○	3-115
		CNRX/ @CNRX	547		○	○	○	○	○	○	○	×	3-115
	타이머 리셋	TRSET/ @TRSET	549	지정한 타이머를 리셋합니다.	○	○	×	×	×	×	×	×	3-117

명령 종류	명령 이름	니모닉	평션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
데이터 비교 명령	심볼 비교	= , <> , < , <= , > , >=	300~ 328	CH 데이터 또는 상수를 비교하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다.	○	○	○	○	○	○	○	○	3-120
	시각 비교	LD, AND, OR+=DT	341	2개의 시각 데이터(BCD 데이터)를 비교하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다.	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-124
		LD, AND, OR+<>DT	342		○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	
		LD, AND, OR+<DT	343		○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-124
		LD, AND, OR+<=DT	344		○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	
		LD, AND, OR+>DT	345		○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	
		LD, AND, OR+>=DT	346		○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	
	부호 없는 비교	CMP	020	CH 데이터 또는 상수를 부호 없는 BIN 16비트(16진수 4자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.	○	○	○	○	○	○	○	○	3-128
		!CMP			○	○	○	○	○	×	○	○	
	부호 없는 2워드 비교	CMPL	060	2CH분의 CH 데이터 또는 상수를 부호 없는 BIN 32비트(16진수 8자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.	○	○	○	○	○	○	○	○	3-128
	부호 있는 BIN 비교	CPS	114	CH 데이터 또는 상수를 부호 있는 BIN 16비트(최상위 비트를 부호 비트로 하는 16진수 4자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.	○	○	○	○	○	○	○	○	3-131
		!CPS			○	○	○	○	○	×	○	○	
	부호 있는 BIN 2워드 비교	CPSL	115	2CH분의 CH 데이터 또는 상수를 부호 있는 2워드 BIN 32비트(최상위 비트를 부호 비트로 하는 16진수 8자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.	○	○	○	○	○	○	○	○	3-131
	다채널 비교	MCMP	019	16CH분의 데이터를 CH 단위로 비교하여 지정된 CH의 해당 비트에 일치하는지 여부의 비교 결과를 출력합니다.	○	○	○	○	○	○	○	○	3-134
		@MCMP			○	○	○	○	○	○	○	○	
	테이블 일치	TCMP	085	비교 데이터 1CH를 비교 테이블 16CH분의 데이터와 각각 일치하는지 비교하여 출력 CH의 해당 비트에 1을 출력합니다.	○	○	○	○	○	○	○	○	3-136
		@TCMP			○	○	○	○	○	○	○	○	
	부호 없는 테이블 간 비교	BCMP	068	비교 데이터의 내용이 16쌍의 비교 데이터의 상하한 범위 안에 있는지 여부를 판별하여 범위 이내이면 출력 CH의 해당 비트에 1을 출력합니다.	○	○	○	○	○	○	○	○	3-138
		@BCMP			○	○	○	○	○	○	○	○	
	확장 테이블 간 비교	BCMP2	502	비교 데이터의 내용이 최대 256개인 비교 데이터 상하한 범위 안에 있는지 여부를 판별하여 범위 이내이면 D~D+최대 15CH의 해당 비트에 1을 출력합니다.	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-140
	영역 비교	ZCP	088	지정된 하나의 CH 데이터 또는 상수가 지정된 상한값과 하한값 사이에 있는지 여부를 부호 없는 BIN 16비트(16진수 4자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.	○	○	○	○	○	○	○	×	3-143
	2워드 영역 비교	ZCPL	116	지정된 하나의 2워드 CH 데이터 또는 상수가 지정된 상한값과 하한값 사이에 있는지 여부를 부호 없는 BIN 32비트(16진수 8자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.	○	○	○	○	○	○	○	×	3-143

명령 종류	명령 이름	니모닉	펑션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
데이터 비교 명령	부호 있는 영역 비교	ZCPS	117	지정된 하나의 CH 데이터 또는 상수가 지정된 상한값과 하한값 사이에 있는지 여부를 부호 있는 BIN 16비트(16진수 4자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.	Ver.1.3 이상	○	×	×	×	×	×	×	3-147
	부호 있는 2워드 영역 비교	ZCPSL	118	지정된 하나의 2워드 CH 데이터 또는 상수가 지정된 상한값과 하한값 사이에 있는지 여부를 부호 있는 BIN 32비트(16진수 8자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.	Ver.1.3 이상	○	×	×	×	×	×	×	3-147
데이터 전송 명령	전송	MOV	021	CH 데이터 또는 상수를 지정된 CH로 전송합니다.	○	○	○	○	○	○	○	○	3-152
		@MOV			○	○	○	○	○	○	○	○	
		!MOV			○	○	○	○	○	○	×	○	
		!@MOV			○	○	○	○	○	○	×	○	
	2워드 전송	MOVL/ @MOVL	498	2CH분의 CH 데이터 또는 상수를 32비트 단위로 전송 대상 CH에 출력합니다.	○	○	○	○	○	○	○	○	3-152
	부정 전송	MVN/ @MVN	022	CH 데이터 또는 상수의 비트 반전 데이터를 지정된 CH로 전송합니다.	○	○	○	○	○	○	○	○	3-154
	부정 2워드 전송	MVNL/ @MVNL	499	2CH분의 CH 데이터 또는 상수의 비트 반전 데이터를 32비트 단위로 지정된 CH에 출력합니다.	○	○	○	○	○	○	○	○	3-154
	비트 전송	MOVB/ @MOVB	082	지정한 비트를 전송합니다.	○	○	○	○	○	○	○	○	3-156
	자릿수 전송	MOVD/ @MOVD	083	자리(4비트) 단위로 전송하며 여러 자리의 전송도 가능합니다.	○	○	○	○	○	○	○	○	3-158
	다중 비트 전송	XFRB/ @XFRB	062	지정된 CH의 지정된 복수의 비트를 전송합니다.	○	○	○	○	○	○	○	○	3-160
	블록 전송	XFER/ @XFER	070	연속된 여러 CH 데이터를 일괄 전송합니다.	○	○	○	○	○	○	○	○	3-162
	블록 설정	BSET/ @BSET	071	연속된 복수의 CH 전체에 동일한 데이터를 설정합니다.	○	○	○	○	○	○	○	○	3-164
	데이터 교환	XCHG/ @XCHG	073	CH 간에 데이터를 교환합니다.	○	○	○	○	○	○	○	○	3-166
	데이터 2워드 교환	XCGL/ @XCGL	562	2CH의 CH 데이터끼리 32비트 단위로 교환합니다.	○	○	○	○	○	○	○	○	3-166
	데이터 분배	DIST/ @DIST	080	전송 대상을 기준으로 오프셋 지정된 채널로 전송 데이터를 전송합니다.	○	○	○	○	○	○	○	○	3-168
	데이터 추출	COLL/ @COLL	081	전송 원본을 기준으로 오프셋 지정된 채널의 내용을 지정 채널로 전송합니다.	○	○	○	○	○	○	○	○	3-170
	인덱스 레지스터 설정	MOVR/ @MOVR	560	지정된 CH 번호, 접점 번호의 I/O 메모리 실제 주소를 지정된 인덱스 레지스터에 설정합니다.	○	○	○	○	○	○	○	○	3-172
	인덱스 레지스터 설정	MOVRW/ @MOVRW	561	지정된 타이머/카운터 현재값의 I/O 메모리 실제 주소를 지정된 인덱스 레지스터에 설정합니다.	○	○	○	○	○	○	○	○	3-172
데이터 이동 명령	이동 레지스터	SFT	010	이동 레지스터 동작을 수행합니다.	○	○	○	○	○	○	○	○	3-176
	좌우 이동 레지스터	SFTR/ @SFTR	084	이동 방향을 전환할 수 있는 이동 레지스터 동작을 수행합니다.	○	○	○	○	○	○	○	○	3-178
	비동기 이동 레지스터	ASFT/ @ASFT	017	지정 CH 범위의 채널 데이터 중 16진수 #0000 이외의 채널 데이터에 대해 앞 또는 뒤 채우기를 실행하고 16진수 #0000의 채널 데이터와 위치를 교체합니다.	○	○	○	○	○	○	○	○	3-180
	워드 이동	WSFT/ @WSFT	016	CH 데이터 단위의 이동 동작을 나타냅니다.	○	○	○	○	○	○	○	○	3-182
	1비트 좌측 이동	ASL/ @ASL	025	CH 데이터를 왼쪽으로 1비트 이동합니다.	○	○	○	○	○	○	○	○	3-184
	1비트 좌측 이동(2워드)	ASLL/ @ASLL	570	2CH분의 데이터를 왼쪽으로 1비트 이동합니다.	○	○	○	○	○	○	○	○	3-184
	1비트 우측 이동	ASR/ @ASR	026	CH 데이터를 오른쪽으로 1비트 이동합니다.	○	○	○	○	○	○	○	○	3-186

명령 종류	명령 이름	니모닉	평션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
데이터 이동 명령	1비트 우측 이동(2워드)	ASRL/ @ASRL	571	2CH분의 데이터를 오른쪽으로 1비트 이동합니다.	○	○	○	○	○	○	○	○	3-186
	CY가 있는 1비트 좌측 회전	ROL/ @ROL	027	16비트의 채널 데이터를 캐리(CY) 플래그를 포함하여 1비트 좌측으로 회전합니다.	○	○	○	○	○	○	○	○	3-188
	CY가 있는 1비트 좌측 회전(2워드)	ROLL/ @ROLL	572	32비트의 채널 데이터를 캐리(CY) 플래그를 포함하여 1비트 좌측으로 회전합니다.	○	○	○	○	○	○	○	○	3-188
	CY가 없는 1비트 좌측 회전	RLNC/ @RLNC	574	16비트의 채널 데이터를 캐리(CY) 플래그를 포함하지 않고 1비트 좌측으로 회전합니다.	○	○	○	○	○	○	○	○	3-190
	CY가 없는 1비트 좌측 회전(2워드)	RLNL/ @RLNL	576	32비트의 채널 데이터를 캐리(CY) 플래그를 포함하지 않고 1비트 좌측으로 회전합니다.	○	○	○	○	○	○	○	○	3-190
	CY가 있는 1비트 우측 회전	ROR/ @ROR	028	16비트의 채널 데이터를 캐리(CY) 플래그를 포함하여 1비트 우측으로 회전합니다.	○	○	○	○	○	○	○	○	3-192
	CY가 있는 1비트 우측 회전(2워드)	RORL/ @RORL	573	32비트의 채널 데이터를 캐리(CY) 플래그를 포함하여 1비트 우측으로 회전합니다.	○	○	○	○	○	○	○	○	3-192
	CY가 없는 1비트 우측 회전	RRNC/ @RRNC	575	16비트의 채널 데이터를 캐리(CY) 플래그를 포함하지 않고 1비트 우측으로 회전합니다.	○	○	○	○	○	○	○	○	3-194
	CY가 없는 1비트 우측 회전(2워드)	RRNL/ @RRNL	577	32비트의 채널 데이터를 캐리(CY) 플래그를 포함하지 않고 1비트 우측으로 회전합니다.	○	○	○	○	○	○	○	○	3-194
	1자리 좌측 이동	SLD/ @SLD	074	연속된 채널의 데이터 내용을 1자리(4비트) 단위로 좌측 이동합니다.	○	○	○	○	○	○	○	○	3-196
	1자리 우측 이동	SRD/ @SRD	075	연속된 채널의 데이터 내용을 1자리(4비트) 단위로 우측 이동합니다.	○	○	○	○	○	○	○	○	3-196
	N비트 데이터 좌측 이동	NSFL/ @NSFL	578	지정된 비트 수 분량의 비트 데이터를 좌측으로 1비트 이동합니다.	○	○	○	○	○	○	○	○	3-198
	N비트 데이터 우측 이동	NSFR/ @NSFR	579	지정된 비트 수 분량의 비트 데이터를 우측으로 1비트 이동합니다.	○	○	○	○	○	○	○	○	3-198
	N비트 좌측 이동	NASL/ @NASL	580	16비트의 채널 데이터를 지정 비트 수만큼 좌측으로 이동합니다.	○	○	○	○	○	○	○	○	3-200
	N비트 2워드 좌측 이동	NSLL/ @NSLL	582	32비트의 채널 데이터를 지정 비트 수만큼 좌측으로 이동합니다.	○	○	○	○	○	○	○	○	3-200
	N비트 우측 이동	NASR/ @NASR	581	16비트의 채널 데이터를 지정 비트 수만큼 우측으로 이동합니다.	○	○	○	○	○	○	○	○	3-203
	N비트 2워드 우측 이동	NSRL/ @NSRL	583	32비트의 채널 데이터를 지정 비트 수만큼 우측으로 이동합니다.	○	○	○	○	○	○	○	○	3-203
증가/감소 명령	BIN 증가	++/ @++	590	16진수 4자리의 1CH 데이터에 1을 더합니다.	○	○	○	○	○	○	○	○	3-208
	BIN 2워드 증가	++L/ @++L	591	16진수 8자리의 2CH 데이터에 1을 더합니다.	○	○	○	○	○	○	○	○	3-208
	BIN 감소	−−/ @−−	592	16진수 4자리의 1CH 데이터에서 1을 뺍니다.	○	○	○	○	○	○	○	○	3-211
	BIN 2워드 감소	−−L/ @−−L	593	16진수 8자리의 2CH 데이터에서 1을 뺍니다.	○	○	○	○	○	○	○	○	3-211
	BCD 증가	++B/ @++B	594	BCD 4자리의 1CH 데이터에 1을 더합니다.	○	○	○	○	○	○	○	○	3-214
	BCD 2워드 증가	++BL/ @++BL	595	BCD 8자리의 2CH 데이터에 1을 더합니다.	○	○	○	○	○	○	○	○	3-214
	BCD 감소	−−B/ @−−B	596	BCD 4자리의 1CH 데이터에서 1을 뺍니다.	○	○	○	○	○	○	○	○	3-217
	BCD 2워드 감소	−−BL/ @−−BL	597	BCD 8자리의 2CH 데이터에서 1을 뺍니다.	○	○	○	○	○	○	○	○	3-217

명령 종류	명령 이름	니모닉	평션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
사칙연산 명령	부호 있음, CY 없는 BIN 가산	+/ @+	400	채널 데이터나 상수를 부호 있는 16진수 4자리로 가산합니다.	○	○	○	○	○	○	○	○	3-222
	부호 있음, CY 없는 BIN 2워드 가산	+L/ @+L	401	2CH 데이터나 상수를 부호 있는 16진수 8자리로 가산합니다.	○	○	○	○	○	○	○	○	3-222
	부호 및 CY 있는 BIN 가산	+C/ @+C	402	데이터나 상수를 캐리(CY) 플래그를 포함하여 부호 있는 16진수 4자리로 가산합니다.	○	○	○	○	○	○	○	○	3-224
	부호 및 CY 있는 BIN 2워드 가산	+CL/ @+CL	403	2CH분의 데이터나 상수를 캐리(CY) 플래그를 포함하여 부호 있는 16진수 8자리로 가산합니다.	○	○	○	○	○	○	○	○	3-224
	CY 없는 BCD 가산	+B/ @+B	404	채널 데이터나 상수를 BCD 4자리로 가산합니다.	○	○	○	○	○	○	○	○	3-227
	CY 없는 BCD 2워드 가산	+BL/ @+BL	405	2CH분의 채널 데이터나 상수를 BCD 8자리로 가산합니다.	○	○	○	○	○	○	○	○	3-227
	CY 있는 BCD 가산	+BC/ @+BC	406	채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 부호 있는 BCD 4자리로 가산합니다.	○	○	○	○	○	○	○	○	3-229
	CY 있는 BCD 2워드 가산	+BCL/ @+BCL	407	2CH분의 채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 BCD 8자리로 가산합니다.	○	○	○	○	○	○	○	○	3-229
	부호 있음, CY 없는 BIN 감산	−/ @−	410	채널 데이터나 상수를 부호 있는 16진수 4자리로 감산합니다.	○	○	○	○	○	○	○	○	3-231
	부호 있음, CY 없는 BIN 2워드 감산	−L/ @−L	411	2CH분의 채널 데이터나 상수를 부호 있는 16진수 8자리로 감산합니다.	○	○	○	○	○	○	○	○	3-231
	부호 및 CY 있는 BIN 감산	−C/ @−C	412	채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 부호 있는 16진수 4자리로 감산합니다.	○	○	○	○	○	○	○	○	3-235
	부호 및 CY 있는 BIN 2워드 감산	−CL/ @CL	413	2CH분의 채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 부호 있는 16진수 8자리로 감산합니다.	○	○	○	○	○	○	○	○	3-235
	CY 없는 BCD 감산	−B/ @−B	414	채널 데이터나 상수를 BCD 4자리로 감산합니다.	○	○	○	○	○	○	○	○	3-238
	CY 없는 BCD 2워드 감산	−BL/ @−BL	415	2CH분의 채널 데이터나 상수를 BCD 8자리로 감산합니다.	○	○	○	○	○	○	○	○	3-238
	CY 있는 BCD 감산	−BC/ @−BC	416	채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 BCD 4자리로 감산합니다.	○	○	○	○	○	○	○	○	3-241
	CY 있는 BCD 2워드 감산	−BCL/ @−BCL	417	2CH분의 채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 BCD 8자리로 감산합니다.	○	○	○	○	○	○	○	○	3-241
	부호 있는 BIN 승산	*/ @*	420	채널 데이터나 상수를 부호 있는 16진수 4자리로 곱합니다.	○	○	○	○	○	○	○	○	3-243
	부호 있는 BIN 2워드 승산	*L/ @*L	421	2CH분의 채널 데이터나 상수를 부호 있는 16진수 8자리로 곱합니다.	○	○	○	○	○	○	○	○	3-243
	부호 없는 BIN 승산	*U/ @*U	422	채널 데이터나 상수를 부호 없는 16진수 4자리로 곱합니다.	○	○	○	○	○	○	○	○	3-245
	부호 없는 BIN 2워드 승산	*UL/ @*UL	423	2CH분의 채널 데이터나 상수를 부호 없는 16진수 8자리로 곱합니다.	○	○	○	○	○	○	○	○	3-245
	BCD 승산	*B/ @*B	424	채널 데이터나 상수를 BCD 4자리로 곱합니다.	○	○	○	○	○	○	○	○	3-247
	BCD 2워드 승산	*BL/ @*BL	425	2CH분의 채널 데이터나 상수를 BCD 8자리로 곱합니다.	○	○	○	○	○	○	○	○	3-247
	부호 있는 BIN 제산	/ @/	430	채널 데이터나 상수를 부호 있는 16진수 4자리로 나눕니다.	○	○	○	○	○	○	○	○	3-249
	부호 있는 BIN 2워드 제산	/L @/L	431	2CH분의 채널 데이터나 상수를 부호 있는 16진수 8자리로 나눕니다.	○	○	○	○	○	○	○	○	3-249
	부호 없는 BIN 제산	/U @/U	432	채널 데이터나 상수를 부호 없는 16진수 4자리로 나눕니다.	○	○	○	○	○	○	○	○	3-251
	부호 없는 BIN 2워드 제산	/UL @/UL	433	2CH분의 채널 데이터나 상수를 부호 없는 16진수 8자리로 나눕니다.	○	○	○	○	○	○	○	○	3-251
	BCD 제산	/B @/B	434	채널 데이터나 상수를 BCD 4자리로 나눕니다.	○	○	○	○	○	○	○	○	3-253
	BCD 2워드 제산	/BL @/BL	435	2CH분의 채널 데이터나 상수를 BCD 8자리로 나눕니다.	○	○	○	○	○	○	○	○	3-253

명령 종류	명령 이름	니모닉	평션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
데이터 변환 명령	BCD→BIN 변환	BIN/ @BIN	023	BCD 데이터를 BIN 데이터로 변환합니다.	○	○	○	○	○	○	○	○	3-256
	BCD→BIN 2워드 변환	BINL/ @BINL	058	2워드 BCD 데이터를 BIN 데이터로 변환합니다.	○	○	○	○	○	○	○	○	3-256
	BIN→BCD 변환	BCD/ @BCD	024	BIN 데이터를 BCD 데이터로 변환합니다.	○	○	○	○	○	○	○	○	3-258
	BIN→BCD 2워드 변환	BCDL/ @BCDL	059	2워드 BIN 데이터를 BCD 데이터로 변환합니다.	○	○	○	○	○	○	○	○	3-258
	2의 보수 변환	NEG/ @NEG	160	BIN 16비트 데이터의 2의 보수를 계산합니다.	○	○	○	○	○	○	○	○	3-261
	2의 보수 2워드 변환	NEGL/ @NEGL	161	BIN 32비트 데이터의 2의 보수를 계산합니다.	○	○	○	○	○	○	○	○	3-261
	부호 확장	SIGN/ @SIGN	600	지정 CH의 데이터를 1CH의 부호 있는 BIN 데이터로 2CH에 부호 확장합니다.	○	○	○	○	○	○	○	○	3-263
	4→16/8→256 디코더	MLPX/ @MLPX	076	지정된 CH의 지정된 자리 또는 바이트를 읽어서(4→16 디코드 또는 8→256 디코드) 지정된 CH의 해당 비트에 1을 출력하고 다른 비트에는 0을 출력합니다.	○	○	○	○	○	○	○	○	3-265
	16→4/256→8 인코더	DMPX/ @DMPX	077	지정된 CH의 16비트 또는 256비트 중에 ON 상태인 최상위 비트나 최하위 비트를 읽어서 (16→4 인코드 또는 256→8 인코드) 지정된 CH의 지정된 자리 또는 바이트에 출력합니다.	○	○	○	○	○	○	○	○	3-270
	ASCII 코드 변환	ASC/ @ASC	086	지정된 16비트 데이터 자리를 ASCII 코드로 변환합니다.	○	○	○	○	○	○	○	○	3-275
	ASCII→HEX 변환	HEX/ @HEX	162	지정된 CH 데이터 자리의 내용을 8비트 ASCII 데이터로 처리하여 지정된 CH의 지정된 자리에 대응하는 문자로 출력합니다.	○	○	○	○	○	○	○	○	3-279
	비트 열→비트 행 변환	LINE/ @LINE	063	지정된 16CH분의 데이터 중 지정된 비트의 데이터만 지정된 다른 CH의 해당 비트로 설정합니다.	○	○	○	○	○	○	○	○	3-283
	비트 행→비트 열 변환	COLM/ @COLM	064	지정된 CH의 각 비트 데이터를 지정된 다른 CH에서 16CH의 지정된 비트로 설정합니다.	○	○	○	○	○	○	○	○	3-285
	부호 있는 BCD→BIN 변환	BINS/ @BINS	470	부호 있는 BCD 데이터를 부호 있는 BIN 데이터로 변환합니다.	○	○	○	○	○	○	○	○	3-287
	부호 있는 BCD→BIN 2워드 변환	BISL/ @BISL	472	부호 있는 2워드 BCD 데이터를 부호 있는 2워드 BIN 데이터로 변환합니다.	○	○	○	○	○	○	○	○	3-287
	부호 있는 BIN→BCD 변환	BCDS/ @BCDS	471	부호 있는 BIN 데이터를 부호 있는 BCD 데이터로 변환합니다.	○	○	○	○	○	○	○	○	3-290
	부호 있는 BIN→BCD 2워드 변환	BDSL/ @BDSL	473	부호 있는 2워드 BIN 데이터를 부호 있는 2워드 BCD 데이터로 변환합니다.	○	○	○	○	○	○	○	○	3-290
	그레이 코드 변환	GRY	474	지정 CH 내의 그레이 2진수 코드 데이터를 지정된 분해능에 따라 BIN 데이터, BCD 데이터 또는 각도(°) 데이터 중 하나로 변환합니다.	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-295
	그레이 코드→BIN 변환	GRAY_BIN/ @GRAY_BIN	478	지정한 그레이 코드(1W)를 BIN 코드(1W)로 변환하여 출력합니다.	○	○	×	×	×	×	×	×	3-298
	그레이 코드→BIN 2워드 변환	GRAY_BINL/@GRAY_BINL	479	지정한 그레이 코드(2W)를 BIN 코드(2W)로 변환하여 출력합니다.	○	○	×	×	×	×	×	×	3-298
	BIN→그레이 코드 변환	BIN_GRAY/ @BIN_GRAY	480	지정한 BIN 코드(1W)를 그레이 코드(1W)로 변환하여 출력합니다.	○	○	×	×	×	×	×	×	3-300
	BIN→그레이 코드 2워드 변환	BIN_GRAYL/@BIN_GRAYL	481	지정한 BIN 코드(2W)를 그레이 코드(2W)로 변환하여 출력합니다.	○	○	×	×	×	×	×	×	3-300

명령 종류	명령 이름	니모닉	펑션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
데이터 변환 명령	숫자 4자리→ ASCII 데이터 변환	STR4	601	숫자 데이터(16진수 4자리)를 ASCII 데이터(4문자)로 변환합니다.	○	○	Ver 4.0 이상	Ver 4.0 이상	Ver 4.0 이상	×	×	×	3-302
	숫자 8자리→ ASCII 데이터 변환	STR8	602	숫자 데이터(16진수 8자리)를 ASCII 데이터(8문자)로 변환합니다.	○	○	Ver 4.0 이상	Ver 4.0 이상	Ver 4.0 이상	×	×	×	3-302
	숫자 16자리→ ASCII 데이터 변환	STR16	603	숫자 데이터(16진수 16자리)를 ASCII 데이터(16문자)로 변환합니다.	○	○	Ver 4.0 이상	Ver 4.0 이상	Ver 4.0 이상	×	×	×	3-302
	ASCII 데이터 변환→ 숫자 4자리	NUM4	604	ASCII 데이터(4문자)를 숫자 데이터(16진수 4자리)로 변환합니다.	○	○	Ver 4.0 이상	Ver 4.0 이상	Ver 4.0 이상	×	×	×	3-305
	ASCII 데이터 변환→숫자 8자리	NUM8	605	ASCII 데이터(8문자)를 숫자 데이터(16진수 8자리)로 변환합니다.	○	○	Ver 4.0 이상	Ver 4.0 이상	Ver 4.0 이상	×	×	×	3-305
	ASCII 데이터 변환→숫자 16자리	NUM16	606	ASCII 데이터(16문자)를 숫자 데이터(16진수 16자리)로 변환합니다.	○	○	Ver 4.0 이상	Ver 4.0 이상	Ver 4.0 이상	×	×	×	3-305
논리 연산 명령	워드 논리곱	ANDW/ @ANDW	034	CH 데이터 단위에서 CH 데이터와 CH 데이터, 상수의 논리곱을 계산합니다.	○	○	○	○	○	○	○	○	3-310
	워드 논리곱(2워드)	ANDL/ @ANDL	610	2CH분의 CH 데이터와 CH 데이터, 상수의 논리곱을 계산합니다.	○	○	○	○	○	○	○	○	3-310
	워드 논리합	ORW/ @ORW	035	CH 데이터 단위에서 CH 데이터와 CH 데이터, 상수의 논리합을 계산합니다.	○	○	○	○	○	○	○	○	3-312
	워드 논리합(2워드)	ORWL/ @ORWL	611	2CH분의 CH 데이터와 CH 데이터, 상수의 논리합을 계산합니다.	○	○	○	○	○	○	○	○	3-312
	워드 배타적 논리합	XORW/ @XORW	036	CH 데이터 단위에서 CH 데이터와 CH 데이터, 상수의 배타적 논리합을 계산합니다.	○	○	○	○	○	○	○	○	3-314
	워드 배타적 논리합(2워드)	XORL/ @XORL	612	2CH의 CH 데이터에 대해 CH 데이터, 상수의 배타적 논리합을 계산합니다.	○	○	○	○	○	○	○	○	3-314
	워드 배타적 논리합 부정	XNRW/ @XNRW	037	CH 데이터 단위에서 CH 데이터와 CH 데이터, 상수의 배타적 논리합 부정을 계산합니다.	○	○	○	○	○	○	○	○	3-316
	워드 배타적 논리합 부정(2워드)	XNRL/ @XNRL	613	2CH분의 CH 데이터와 CH 데이터, 상수의 배타적 논리합을 계산합니다.	○	○	○	○	○	○	○	○	3-316
	비트 반전	COM/@COM	029	CH 데이터를 비트 반전합니다.	○	○	○	○	○	○	○	○	3-318
	비트 2워드 반전	COML/ @COML	614	2CH분의 CH 데이터에서 각 비트를 반전 출력합니다.	○	○	○	○	○	○	○	○	3-318
특수 연산 명령	BIN 제곱근 연산	ROTB/ @ROTB	620	지정된 BIN 데이터(32비트)를 제곱근 연산하고 결과의 정수부를 출력합니다.	○	○	○	○	○	○	○	○	3-322
	BCD 제곱근 연산	ROOT/ @ROOT	072	지정된 데이터를 BCD의 2워드 데이터로 제곱근 연산하고 결과의 정수부를 BCD 데이터로 출력합니다.	○	○	○	○	○	○	○	○	3-324
	숫자 변환	APR/ @APR	069	SIN, COS, 꺾은선 근사 계산을 수행합니다.	○	○	○	○	○	○	○	○	3-326
	부동 소수점 제산(BCD)	FDIV/ @FDIV	079	가수 7자리와 지수 1자리의 2CH(BCD 8자리) 부동 소수점을 나눕니다.	○	○	○	○	○	○	○	○	3-335
	비트 카운터	BCNT/ @BCNT	067	CH 데이터 내에서 "1"인 비트의 총 수를 계산합니다.	○	○	○	○	○	○	○	○	3-338
부동 소수점 변환 및 연산 명령	부동 소수점→ 16비트 BIN 변환	FIX/ @FIX	450	지정된 부동 소수점 32비트 데이터를 부호 있는 BIN 16비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-347
	부동 소수점→ 32비트 BIN 변환	FIXL/ @FIXL	451	지정된 부동 소수점 32비트 데이터를 부호 있는 BIN 32비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-347

명령 종류	명령 이름	니모닉	평션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
부동 소수점 변환 및 연산 명령	16비트 BIN→ 부동 소수점 변환	FLT/ @FLT	452	지정된 부호 있는 BIN 16비트 데이터를 부동 소수점 32비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-349
	32비트 BIN→ 부동 소수점 변환	FLTL/ @FLTL	453	지정된 부호 있는 BIN 32비트 데이터를 부동 소수점 32비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-349
	부동 소수점 가산	+F/ @+F	454	지정된 부동 소수점 데이터(32비트)를 가산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-351
	부동 소수점 감산	−F/ @−F	455	지정된 부동 소수점 데이터(32비트)를 감산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-351
	부동 소수점 승산	*F/ @*F	456	지정된 부동 소수점 데이터(32비트)를 곱하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-351
	부동 소수점 제산	/F @/F	457	지정된 부동 소수점 데이터(32비트)를 나눠서 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-351
	각도→라디안 변환	RAD/ @RAD	458	지정된 부동 소수점 데이터(32비트)를 도(°) 단위에서 라디안(rad) 단위로 변환하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-354
	라디안→각도 변환	DEG/ @DEG	459	지정된 부동 소수점 데이터(32비트)를 라디안(rad) 단위에서 도(°) 단위로 변환하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-356
	SIN 연산	SIN/ @SIN	460	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 SIN(사인) 값을 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-358
	COS 연산	COS/ @COS	461	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 COS(코사인) 값을 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-358
	TAN 연산	TAN/ @TAN	462	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 TAN(탄젠트) 값을 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-358
	고속 SIN 연산	SINQ/ @SINQ	475	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 SIN(사인) 값을 연산하여 결과를 고속으로 지정된 채널에 출력합니다.	○	○	CJ1-H-R만 해당	×	×	×	×	×	3-361
	고속 COS 연산	COSQ/ @COSQ	476	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 COS(코사인) 값을 연산하여 결과를 고속으로 지정된 채널에 출력합니다.	○	○	CJ1-H-R만 해당	×	×	×	×	×	3-361
	고속 TAN 연산	TANQ/ @TANQ	477	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 TAN(탄젠트) 값을 연산하여 결과를 고속으로 지정된 채널에 출력합니다.	○	○	CJ1-H-R만 해당	×	×	×	×	×	3-361
	SIN^{-1} 연산	ASIN/ @ASIN	463	지정된 부동 소수점 데이터(32비트)로 표시되는 SIN(사인) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-364
	COS^{-1} 연산	ACOS/ @ACOS	464	지정된 부동 소수점 데이터(32비트)로 표시되는 COS(코사인) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-364

명령 종류	명령 이름	니모닉	평션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
부동 소수점 변환 및 연산 명령	TAN^{-1} 연산	ATAN/ @ATAN	465	지정된 부동 소수점 데이터(32비트)로 표시되는 TAN(탄젠트) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-364
	제곱근 연산	SQRT/ @SQRT	466	지정된 부동 소수점 데이터(32비트)의 제곱근(√)을 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-367
	지수 연산	EXP/ @EXP	467	지정된 부동 소수점 데이터(32비트)의 지수(e를 밑수로 했을 때의 지수)를 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-369
	로그 연산	LOG/ @LOG	468	지정된 부동 소수점 데이터(32비트)의 자연로그(e를 밑수로 했을 때의 로그)를 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-371
	거듭제곱 연산	PWR/ @PWR	840	지정된 부동 소수점 데이터(32비트)의 거듭제곱 연산을 수행하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	○	3-373
	단정도 부동 소수점 데이터 비교	LD, AND, OR+=F	329	지정된 단정도 부동 소수점 데이터(32비트) 또는 상수를 비교하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다.	○	○	○	○	○	○	○	×	3-375
		LD, AND, OR+<>F	330		○	○	○	○	○	○	○	×	3-375
		LD, AND, OR+<F	331		○	○	○	○	○	○	○	×	3-375
		LD, AND, OR+<=F	332		○	○	○	○	○	○	○	×	3-375
		LD, AND, OR+>F	333		○	○	○	○	○	○	○	×	3-375
		LD, AND, OR+>=F	334		○	○	○	○	○	○	○	×	3-375
	부동 소수점<단>→문자열 변환	FSTR/ @FSTR	448	지정된 단정도 부동 소수점 데이터(32비트)를 소수점 형식 또는 지수 형식으로 표시하고 그것을 문자열 데이터(ASCII 코드)로 변환합니다.	○	○	○	○	○	○	○	×	3-378
	문자열→부동 소수점<단> 변환	FVAL/ @FVAL	449	지정된 소수점 형식 또는 지수 형식으로 나타낸 문자열(ASCII 코드)을 단정도 부동 소수점 데이터(32비트)로 변환하여 지정된 채널에 결과를 출력합니다.	○	○	○	○	○	○	○	×	3-382
	부동 소수점 전송<단>	MOVF/ @MOVF	469	지정된 부동 소수점 데이터(32비트)를 지정 채널로 전송합니다.	○	○	CJ1-H-R만 해당	×	×	×	×	×	3-386
배정도 부동 소수점 변환 및 연산 명령	부동 소수점→ 16비트 BIN 변환<배>	FIXD/ @FIXD	841	지정된 배정도 부동 소수점 데이터(64비트)를 부호 있는 BIN 16비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-396
	부동 소수점→ 32비트 BIN 변환<배>	FIXLD/ @FIXLD	842	지정된 배정도 부동 소수점 데이터(64비트)를 부호 있는 BIN 32비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-396
	16비트 BIN→ 부동 소수점 변환<배>	DBL/ @DBL	843	지정된 부호 있는 BIN 16비트 데이터를 배정도 부동 소수점 데이터(64비트)로 변환하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-398
	32비트 BIN→ 부동 소수점 변환<배>	DBLL/ @DBLL	844	지정된 부호 있는 BIN 32비트 데이터를 배정도 부동 소수점 데이터(64비트)로 변환하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-398
	부동 소수점 가산<배>	+D/ @+D	845	지정된 배정도 부동 소수점 데이터(64비트)를 더하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-400
	부동 소수점 감산<배>	−D/ @−D	846	지정된 배정도 부동 소수점 데이터(64비트)를 빼서 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-400
	부동 소수점 승산<배>	*D/ @*D	847	지정된 배정도 부동 소수점 데이터(64비트)를 곱해서 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-400

명령 종류	명령 이름	니모닉	평션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
배정도 부동 소수점 변환 및 연산 명령	부동 소수점 제산<배>	/D @/D	848	지정된 배정도 부동 소수점 데이터(64비트)를 나눠서 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-400
	각도→라디안 변환<배>	RADD/ @RADD	849	지정된 배정도 부동 소수점 데이터(64비트)를 도(°) 단위에서 라디안(rad) 단위로 변환하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-403
	라디안→각도 변환<배>	DEGD/ @DEGD	850	지정된 배정도 부동 소수점 데이터(64비트)를 라디안(rad) 단위에서 도(°) 단위로 변환하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-405
	SIN 연산<배>	SIND/ @SIND	851	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 각도(라디안 단위)의 SIN(사인) 값을 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-407
	COS 연산<배>	COSD/ @COSD	852	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 각도(라디안 단위)의 COS(코사인) 값을 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-407
	TAN 연산<배>	TAND/ @TAND	853	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 각도(라디안 단위)의 TAN(탄젠트) 값을 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-407
	SIN^{-1} 연산<배>	ASIND/ @ASIND	854	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 SIN(사인) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-410
	COS^{-1} 연산<배>	ACOSD/ @ACOSD	855	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 COS(코사인) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-410
	TAN^{-1} 연산<배>	ATAND/ @ATAND	856	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 TAN(탄젠트) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-410
	제곱근 연산<배>	SQRTD/ @SQRTD	857	지정된 배정도 부동 소수점 데이터(64비트)의 제곱근(√)을 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-413
	지수 연산<배>	EXPD/ @EXPD	858	지정된 배정도 부동 소수점 데이터(64비트)의 지수(e를 밑수로 했을 때의 지수)를 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-415
	로그 연산<배>	LOGD/ @LOGD	859	지정된 배정도 부동 소수점 데이터(64비트)의 자연 로그(e를 밑수로 했을 때의 지수)를 연산하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-417
	거듭제곱 연산<배>	PWRD/ @PWRD	860	지정된 배정도 부동 소수점 데이터(64비트)의 거듭제곱 연산을 수행하여 결과를 지정된 채널에 출력합니다.	○	○	○	○	○	○	○	×	3-419
	배정도 부동 소수점 데이터 비교	LD, AND, OR+=D	335	지정된 배정도 부동 소수점 데이터(64비트)를 비교하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다.	○	○	○	○	○	○	○	×	3-421
		LD, AND, OR+<>D	336		○	○	○	○	○	○	○	×	3-421
		LD, AND, OR+<D	337		○	○	○	○	○	○	○	×	3-421
		LD, AND, OR+<=D	338		○	○	○	○	○	○	○	×	3-421
		LD, AND, OR+>D	339		○	○	○	○	○	○	○	×	3-421
		LD, AND, OR+>=D	340		○	○	○	○	○	○	○	×	3-421

명령 종류	명령 이름	니모닉	펑션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
테이블 데이터 처리 명령	스택 영역 설정	SSET/ @SSET	630	지정된 CH에서 지정된 CH 수의 스택 영역을 설정하고, 동시에 데이터 저장 영역을 0으로 클리어합니다.	○	○	○	○	○	○	○	○	3-430
	스택 데이터 저장	PUSH/ @PUSH	632	지정된 스택 영역에 데이터를 저장합니다. 동시에 스택 포인터의 내용을 1 증가시킵니다.	○	○	○	○	○	○	○	○	3-432
	후입선출	LIFO/ @LIFO	634	지정된 스택 영역의 스택 포인터의 데이터를 1 감소시키고, 그 주소에서 데이터(마지막에 저장한 데이터)를 읽습니다.	○	○	○	○	○	○	○	○	3-434
	선입선출	FIFO/ @FIFO	633	지정된 스택 영역에서 처음 저장된 데이터를 읽습니다.	○	○	○	○	○	○	○	○	3-434
	스택 데이터 수 출력	SNUM/ @SNUM	638	지정된 스택 영역 내의 스택 데이터 수(CH 수)를 계산합니다.	○	○	○	○	○	○	○	×	3-438
	스택 데이터 참조	SREAD/ @SREAD	639	지정된 스택 영역에서 지정된 데이터(현재 스택 포인터가 있는 위치에서 지정된 개수 만큼 앞에 있는 데이터)를 읽습니다.	○	○	○	○	○	○	○	×	3-440
	스택 데이터 갱신	SWRIT/ @SWRIT	640	지정된 스택 영역에서 지정된 데이터(현재 스택 포인터가 있는 위치에서 지정된 개수 만큼 앞에 있는 데이터)의 위치에 다른 데이터를 덮어 씁니다.	○	○	○	○	○	○	○	×	3-443
	스택 데이터 삽입	SINS/ @SINS	641	지정된 스택 영역에서 지정된 데이터 위치(현재 스택 포인터가 있는 위치에서 지정된 개수 만큼 앞에 있는 위치)에 다른 데이터를 삽입합니다.	○	○	○	○	○	○	○	×	3-446
	스택 데이터 삭제	SDEL/ @SDEL	642	지정된 스택 영역에서 지정된 데이터 위치(현재 스택 포인터가 있는 위치에서 지정된 개수 만큼 앞에 있는 위치)의 데이터를 삭제합니다.	○	○	○	○	○	○	○	×	3-449
	테이블 영역 선언	DIM/ @DIM	631	테이블 영역을 등록합니다.	○	○	○	○	○	○	○	○	3-452
	레코드 위치 설정	SETR/ @SETR	635	지정된 레코드의 위치(레코드 선두의 I/O 메모리 실제 주소)를 지정된 인덱스 레지스터에 저장합니다.	○	○	○	○	○	○	○	○	3-454
	레코드 위치 읽기	GETR/ @GETR	636	지정된 인덱스 레지스터의 값(I/O 메모리 실제 주소)이 포함된 레코드의 번호를 출력합니다.	○	○	○	○	○	○	○	○	3-456
	데이터 검색	SRCH/ @SRCH	181	지정된 테이블 내의 데이터를 검색하고 일치하는 데이터가 있는 경우, 해당 데이터가 있는 CH(여러 개 있는 경우에는 하위 CH)의 I/O 메모리 실제 주소를 인덱스 레지스터 IR00에 출력합니다.	○	○	○	○	○	○	○	○	3-458
	바이트 스와핑	SWAP/ @SWAP	637	지정된 테이블 데이터의 지정된 CH 수의 상위 바이트와 하위 바이트를 교체합니다.	○	○	○	○	○	○	○	○	3-461
	최대값 검색	MAX/ @MAX	182	지정된 테이블 내의 최대값을 검색합니다.	○	○	○	○	○	○	○	○	3-463
	최소값 검색	MIN/ @MIN	183	지정된 테이블 내의 최소값을 검색합니다.	○	○	○	○	○	○	○	○	3-463
	2워드 데이터 최대값 검색	MAXL/ @MAXL	174	지정된 데이터 수만큼 2CH 데이터의 테이블 데이터로 간주하고, 그 중에서 최대값을 검색하여 출력합니다.	○	○	×	×	×	×	×	×	3-467
	단정도 부동 소수점 데이터 최대값 검색	MAXF/ @MAXF	176	지정된 데이터 수만큼 2CH 데이터의 단정도 부동 소수점 데이터로 간주하고, 그 중에서 최대값을 검색하여 출력합니다.	○	○	×	×	×	×	×	×	3-470
	배정도 부동 소수점 데이터 최대값 검색	MAXD/ @MAXD	178	지정된 데이터 수만큼 2CH 데이터의 배정도 부동 소수점 데이터로 간주하고, 그 중에서 최대값을 검색하여 출력합니다.	○	○	×	×	×	×	×	×	3-472

명령 종류	명령 이름	니모닉	평션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
테이블 데이터 처리 명령	2워드 데이터 최소값 검색	MINL/ @MINL	175	지정된 데이터 수만큼 2CH 데이터의 테이블 데이터로 간주하고, 그 중에서 최소값을 검색하여 출력합니다.	○	○	×	×	×	×	×	×	3-474
	단정도 부동 소수점 데이터 최소값 검색	MINF/ @MINF	177	지정된 데이터 수만큼 2CH 데이터의 단정도 부동 소수점 데이터로 간주하고, 그 중에서 최소값을 검색하여 출력합니다.	○	○	×	×	×	×	×	×	3-476
	배정도 부동 소수점 데이터 최소값 검색	MIND/ @MIND	179	지정된 데이터 수만큼 2CH 데이터의 배정도 부동 소수점 데이터로 간주하고, 그 중에서 최소값을 검색하여 출력합니다.	○	○	×	×	×	×	×	×	3-478
	SUM 값 산출	SUM/ @SUM	184	지정된 테이블 내의 SUM 값을 계산하고 2CH로 출력합니다.	○	○	○	○	○	○	○	○	3-480
	FCS 값 산출	FCS/ @FCS	180	지정된 테이블 내의 FCS 값을 계산하고 ASCII 코드 데이터로 변환하여 출력합니다.	○	○	○	○	○	○	○	○	3-483
추적 명령	부호 없는 1W 레코드 검색	RSRCH < , @RSRCH <	360	지정한 테이블 내의 여러 레코드 중에서 레코드 내 지정한 위치의 데이터(1W)를 검색하고, 일치하는 레코드가 있으면 레코드 번호와 레코드 데이터를 출력합니다.	○	○	×	×	×	×	×	×	3-495
		RSRCH <= , @RSRCH <=	361		○	○	×	×	×	×	×	×	3-495
		RSRCH = , @RSRCH =	362		○	○	×	×	×	×	×	×	3-495
		RSRCH >= , @RSRCH >=	363		○	○	×	×	×	×	×	×	3-495
		RSRCH > , @RSRCH >	364		○	○	×	×	×	×	×	×	3-495
	부호 없는 2W 레코드 검색	RSRCH2 < , @RSRCH2 <	370	지정한 테이블 내의 여러 레코드 중에서 레코드 내 지정한 위치의 데이터(2W)를 검색하고, 일치하는 레코드가 있으면 레코드 번호와 레코드 데이터를 출력합니다.	○	○	×	×	×	×	×	×	3-501
		RSRCH2 <= , @RSRCH2 <=	371		○	○	×	×	×	×	×	×	3-501
		RSRCH2 = , @RSRCH2 =	372		○	○	×	×	×	×	×	×	3-501
		RSRCH2 >= , @RSRCH2 >=	373		○	○	×	×	×	×	×	×	3-501
		RSRCH2 > , @RSRCH2 >	374		○	○	×	×	×	×	×	×	3-501
	부호 없는 4W 레코드 검색	RSRCH4 < , @RSRCH4 <	380	지정한 테이블 내의 여러 레코드 중에서 레코드 내 지정한 위치의 데이터(4W)를 검색하고, 일치하는 레코드가 있으면 레코드 번호와 레코드 데이터를 출력합니다.	○	○	×	×	×	×	×	×	3-504
		RSRCH4 <= , @RSRCH4 <=	381		○	○	×	×	×	×	×	×	3-504
		RSRCH4 = , @RSRCH4 =	382		○	○	×	×	×	×	×	×	3-504
		RSRCH4 >= , @RSRCH4 >=	383		○	○	×	×	×	×	×	×	3-504
		RSRCH4 > , @RSRCH4 >	384		○	○	×	×	×	×	×	×	3-504
	부호 없는 1W 레코드 정렬	RSORT/ @RSORT	203	지정한 테이블 내의 여러 레코드를 레코드 내 지정 위치의 데이터(1W)로 정렬합니다.	○	○	×	×	×	×	×	×	3-507
	부호 없는 2W 레코드 정렬	RSORT2/ @RSORT2	204	지정한 테이블 내의 여러 레코드를 레코드 내 지정 위치의 데이터(2W)로 정렬합니다.	○	○	×	×	×	×	×	×	3-511
	부호 없는 4W 레코드 정렬	RSORT4/ @RSORT4	205	지정한 테이블 내의 여러 레코드를 레코드 내 지정 위치의 데이터(4W)로 정렬합니다.	○	○	×	×	×	×	×	×	3-514
데이터 제어 명령	PID 연산	PID	190	지정된 파라미터에 따라 PID 연산(목표값 필터형 2 자유도 PID)을 수행하여 조작량을 출력합니다.	○	○	○	○	○	○	○	○	3-518
	자동 조정 가능한 PID 연산	PIDAT	191	지정된 파라미터에 따라 PID 연산을 수행합니다.	○	○	○	○	○	○	○	×	3-527
	상하한 리미트 제어	LMT/ @LMT	680	입력 데이터가 상하한 리미트 데이터의 범위 안인지 여부에 따라 출력 데이터를 제어합니다.	○	○	○	○	○	○	○	○	3-534
	데드 밴드 제어	BAND/ @BAND	681	입력 데이터가 불감대(Dead band)의 범위 안인지 여부에 따라 출력 데이터를 제어합니다.	○	○	○	○	○	○	○	○	3-536
	데드 존 제어	ZONE/ @ZONE	682	지정한 바이어스 값을 입력 데이터에 더하여 출력합니다.	○	○	○	○	○	○	○	○	3-538

명령 종류	명령 이름	니모닉	평션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
데이터 제어 명령	시분할 비례 출력	TPO	685	지정된 채널 내의 듀티비 또는 조작량을 입력하고 지정된 파라미터에 따라 듀티비를 시분할 비례 출력으로 변환하여 지정한 접점에 펄스 출력합니다.	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-540
	스케일링	SCL/ @SCL	194	지정한 1차 함수에 따라 부호 없는 BIN 데이터를 부호 없는 BCD 데이터로 스케일링(변환)합니다.	○	○	○	○	○	○	○	○	3-546
	스케일링 2	SCL2/ @SCL2	486	지정한 오프셋이 있는 1차 함수에 따라 부호 있는 BIN 데이터를 부호 있는 BCD 데이터로 스케일링(변환)합니다.	○	○	○	○	○	○	○	○	3-550
	스케일링 3	SCL3/ @SCL3	487	지정한 오프셋이 있는 1차 함수에 따라 부호 있는 BCD 데이터를 부호 있는 BIN 데이터로 스케일링(변환)합니다.	○	○	○	○	○	○	○	○	3-553
	데이터 평균화	AVG	195	CH 데이터의 지정한 사이클 횟수 분량의 평균값을 BIN으로 계산합니다.	○	○	○	○	○	○	○	○	3-556
서브루틴 명령	서브루틴 호출	SBS/ @SBS	091	지정된 번호의 서브루틴을 호출하는 프로그램을 실행합니다.	○	○	○	○	○	○	○	○	3-561
	매크로	MCRO/ @MCRO	099	인수가 있는 서브루틴 호출입니다.	○	○	○	○	○	○	○	○	3-566
	서브루틴 진입	SBN	092	지정된 번호의 서브루틴의 선두를 나타냅니다.	○	○	○	○	○	○	○	○	3-569
	서브루틴 반환	RET	093	서브루틴의 종료를 나타냅니다.	○	○	○	○	○	○	○	○	3-569
	글로벌 서브루틴 호출	GSBS/ @GSBS	750	지정된 번호의 글로벌 서브루틴을 호출하고 프로그램을 실행합니다.	○	○	○	○	○	○	○	×	3-572
	글로벌 서브루틴 진입	GSBN	751	지정된 번호의 글로벌 서브루틴의 선두를 나타냅니다.	○	○	○	○	○	○	○	×	3-578
	글로벌 서브루틴 반환	GRET	752	글로벌 서브루틴의 종료를 나타냅니다.	○	○	○	○	○	○	○	×	3-578
인터럽트 제어 명령	인터럽트 마스크 세트	MSKS/ @MSKS	690	인터럽트 요인을 제어합니다.	○	○	○	○	○	○	×	○	3-585
	인터럽트 마스크 읽기	MSKR/ @MSKR	692	MSKS 명령에 의해 지정된 인터럽트 제어 상태를 읽습니다.	○	○	○	○	○	○	×	○	3-591
	인터럽트 해제	CLI/ @CLI	691	I/O 인터럽트 입력 기억 해제/유지 지정 또는 예약 인터럽트의 첫 번째 인터럽트 시작 시간 지정을 수행합니다.	○	○	○	○	○	○	×	○	3-595
	인터럽트 태스크 실행 금지	DI/ @DI	693	모든 인터럽트 태스크(전원 OFF 인터럽트 태스크 제외)의 실행을 일시적으로 금지합니다.	○	○	○	○	○	○	○	○	3-599
	인터럽트 태스크 실행 금지 해제	EI	694	모든 인터럽트 태스크(전원 OFF 인터럽트 태스크 제외)의 실행 금지를 해제합니다.	○	○	○	○	○	○	○	○	3-601
고속 카운터/펄스 출력 명령	동작 모드 제어	INI/ @INI	880	비교 테이블과의 시작/정지, 고속 카운터 현재값 변경, 인터럽트 입력(카운터 모드)의 현재값 변경, 펄스 출력 현재값 변경(0으로 원점 확정), 펄스 출력 정지 중 하나를 실행합니다.	×	×	×	×	○	×	×	×	3-604
	펄스 현재값 읽기	PRV/ @PRV	881	고속 카운터 현재값, 펄스 출력 현재값, 인터럽트 입력(카운터 모드)의 현재값을 읽습니다.	×	×	×	×	○	×	×	×	3-607
	펄스 주파수 변환	PRV2	883	고속 카운터에 입력된 펄스 주파수를 읽어서 회전 속도(회전 수)로 변환하거나 카운터 현재값을 누적 회전 수로 변환하여 결과를 16진수 8자리로 출력합니다.	×	×	×	×	Ver 2.0 이상	×	×	×	3-612
	비교 테이블 등록	CTBL/ @CTBL	882	고속 카운터 현재값에 대해 목표값 일치 비교 또는 대역 비교를 실시합니다.	×	×	×	×	○	×	×	×	3-615
	주파수 설정	SPED/ @SPED	885	주파수를 지정하여 가감속 없이 펄스를 출력합니다.	×	×	×	×	○	×	×	×	3-618
	펄스량 설정	PULS/ @PULS	886	펄스 출력량을 설정합니다.	×	×	×	×	○	×	×	×	3-621

명령 종류	명령 이름	니모닉	평션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
고속 카운터/펄스 출력 명령	위치 결정	PLS2/ @PLS2	887	펄스 주파수와 가감속 비율을 지정하여 가감속이 있는 펄스 출력(가속 비율≠감속 비율)을 실시합니다.	×	×	×	×	○	×	×	×	3-623
	주파수 가감속 제어	ACC/ @ACC	888	펄스 주파수와 가감속 비율을 지정하여 가감속이 있는 펄스 출력(가속 비율=감속 비율)을 실시합니다.	×	×	×	×	○	×	×	×	3-628
	원점 검색	ORG/ @ORG	889	원점 검색 또는 원점 복귀를 실행합니다.	×	×	×	×	○	×	×	×	3-632
	PWM 출력	PWM/@PWM	891	듀티비 가변 펄스를 출력합니다.	×	×	×	×	○	×	×	×	3-635
공정 진행 제어 명령	스텝 래더 영역 진행	SNXT	009	배치 위치에 따라 다음 3가지 동작을 수행합니다. 1) 스텝 래더 영역으로 진행 2) 다음 공정 번호로 진행 3) 스텝 래더 영역의 종료로 진행	○	○	○	○	○	○	○	○	3-639
	스텝 래더 영역 정의	STEP	008	배치 위치 및 공정 번호 지정 유무에 따라 다음 2가지 동작을 수행합니다. 1) 지정된 공정의 시작 2) 스텝 래더 영역 전체의 종료 스텝 래더 영역은 STEP 명령(공정 번호 지정)~STEP 명령(공정 번호 없음)까지의 영역입니다.	○	○	○	○	○	○	○	○	3-639
I/O 유니트용 명령	I/O 리프레시	IORF/ @IORF	097	지정된 리프레시 하위 CH 번호부터 지정된 리프레시 상위 CH 번호까지의 I/O 채널 데이터를 리프레시합니다.	○	○	○	○	○	○	○	○	3-650
	고기능 I/O 유니트 매회 리프레시	FIORF/ @FIORF	225	지정된 고기능 I/O 유니트와의 I/O 리프레시를 매회 실행합니다.	○	○	CJ1-H-R만 해당	×	×	×	×	×	3-653
	CPU 고기능 유니트 매회 I/O 리프레시	DLNK/ @DLNK	226	지정된 CPU 고기능 유니트와의 I/O 리프레시를 매회 실행합니다.	○	○	○	○	○	○	○	×	3-656
	7세그먼트 디코더	SDEC/ @SDEC	078	채널 데이터에서 지정된 자리의 각 4비트 내용(0~F)을 8비트의 7세그먼트 데이터로 변환하여 지정된 CH 이후의 상위 또는 하위의 각 8비트에 출력합니다.	○	○	○	○	○	○	○	○	3-659
	디지털 스위치	DSW	210	입력 유니트 및 출력 유니트에 연결된 외부 디지털 스위치(또는 로터리 스위치)의 설정값을 읽어 지정된 CH 이후에 8자리 또는 4자리의 숫자(BCD 데이터)로 저장합니다.	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-661
	10키 입력	TKY	211	입력 유니트에 연결된 외부의 10키패드에서 숫자를 순서대로 읽어 지정된 CH 이후에 최대 8자리의 숫자(BCD 데이터)로 저장합니다.	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-664
	16키 입력	HYK	212	입력 유니트 및 출력 유니트에 연결된 외부의 16키보드에서 숫자를 순서대로 읽어 지정된 CH 이후에 최대 8자리의 숫자(BCD 데이터)로 저장합니다.	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-667
	행렬 입력	MTR	213	입력 유니트 및 출력 유니트에 연결된 외부의 8점×8열의 접점(행렬)에서 64점을 순서대로 읽어 지정된 CH 이후에 64점(4CH) 데이터로 저장합니다.	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-670
	7세그먼트 표시	7SEG	214	4자리 또는 8자리의 숫자(BCD 데이터)를 7세그먼트 표시기용 데이터로 변환하여 지정된 CH 이후에 출력합니다.	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-674
	아날로그 입력 직접 변환	AIDC/ @AIDC	216	직접 변환 모드의 아날로그 입력 유니트 CJ1W-AD042에서 지정된 아날로그 입력 번호의 입력 변환값을 읽습니다.	Ver. 1.1 이상	○	×	×	×	×	×	×	3-677

명령 종류	명령 이름	니모닉	펑션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
I/O 유니트용 명령	아날로그 출력 직접 변환	AODC/ @AODC	217	직접 변환 모드의 아날로그 출력 유니트 CJ1W-DA042V에 대해 지정된 아날로그 출력 번호의 출력 설정값을 출력합니다.	Ver. 1.1 이상	○	×	×	×	×	×	×	3-680
	NC 유니트 위치 결정 고속 기동	NCDMV @NCDMV	218	위치 제어 유니트(CJ1W-NC□□4, CJ1W-NC□81)의 지정된 축 번호에 대해 상대 이동 또는 절대 이동의 고속 PTP 기동을 합니다.	Ver. 1.3 이상	○	×	×	×	×	×	×	3-683
	NC 유니트 위치 결정 시작 트리거	NCDTR @NCDTR	219	위치 제어 유니트(CJ1W-NC□81)의 메모리 운전 기능에서 시퀀스의 시작 조건이 이 명령에서 지령 대기가 된 경우 해당 시퀀스를 시작합니다.	Ver. 1.3 이상	○	×	×	×	×	×	×	3-688
	인텔리전트 I/O 읽기	IORD/ @IORD	222	고기능 I/O 유니트 또는 CPU 고기능 유니트의 메모리 영역의 내용을 읽습니다.	○	○	○	○	○	○	○	○	3-691
	인텔리전트 I/O 쓰기	IOWR/ @IOWR	223	CPU 유니트의 I/O 메모리 영역의 내용을 고기능 I/O 유니트 또는 CPU 고기능 유니트에 출력합니다.	○	○	○	○	○	○	○	○	3-694
시리얼 통신 명령	프로토콜 매크로	PMCR/ @PMCR	260	시리얼 커뮤니케이션 보드(CS 시리즈만 해당) 또는 시리얼 커뮤니케이션 유니트에 등록된 송수신 시퀀스(프로토콜 데이터)를 호출하여 실행합니다.	○	○	○	○	○	○	○	○	3-700
	CPU 유니트 시리얼 커뮤니케이션 보드 시리얼 포트 출력	TXD/ @TXD	236	지정된 바이트 수 분량의 데이터를 그 상태 그대로 변환하지 않고 PLC 시스템 설정의 무수순 모드일 때의 시작 코드/종료 코드 지정에 따라 CPU 유니트 내장 RS-232C 포트(무수순 모드) 또는 시리얼 커뮤니케이션 보드 유니트 Ver.1.2 이상의 시리얼 통신 포트(무수순 모드)에서 송신합니다.	○	○	○	○	○	○	○	○	3-706
	CPU 유니트 시리얼 커뮤니케이션 보드 시리얼 포트 입력	RXD/ @RXD	235	CPU 유니트 내장 RS-232C 포트(무수순 모드) 또는 시리얼 커뮤니케이션 보드 유니트 Ver.1.2 이상의 시리얼 통신 포트(무수순 모드)에서 PLC 시스템 설정의 무수순 모드일 때의 시작 코드/종료 코드 지정에 따라 수신한 데이터를 지정된 시작 CH 번호부터 지정된 바이트 수만큼 출력합니다.	○	○	○	○	○	○	○	○	3-712
	시리얼 커뮤니케이션 유니트 시리얼 포트 출력	TXDU/ @TXDU	256	지정된 바이트 수 분량의 데이터를 그 상태 그대로 변환하지 않고 할당 DM 영역 설정의 무수순 모드일 때의 시작 코드/종료 코드 지정에 따라 시리얼 커뮤니케이션 유니트의 유니트 Ver.1.2 이상의 시리얼 통신 포트(무수순 모드)에서 송신합니다.	○	○	Ver. 3.0 이상	Ver. 3.0 이상	Ver. 3.0 이상	×	×	×	3-720
	시리얼 커뮤니케이션 유니트 시리얼 포트 입력	RXDU/ @RXDU	255	시리얼 커뮤니케이션 유니트의 유니트 Ver.1.2 이상의 시리얼 통신 포트에서 할당 DM 영역 설정의 무수순 모드일 때의 시작 코드/종료 코드 지정에 따라 수신한 데이터를 지정된 시작 CH 번호부터 지정된 바이트 수만큼 출력합니다.	○	○	Ver. 3.0 이상	Ver. 3.0 이상	Ver. 3.0 이상	×	×	×	3-726

명령 종류	명령 이름	니모닉	평션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
시리얼 통신 명령	시리얼 커뮤니케이션 유니트 시리얼 포트 직접 송신	DTXDU/ @DTXDU	262	시리얼 커뮤니케이션 유니트 CJ1W-SCU22/32/42의 시리얼 통신 포트(무수순 모드)에서 할당 DM 영역 설정의 무수순 모드일 때의 시작 코드/종료 코드에 따라 지정된 시작 CH 번호부터 지정된 바이트 수만큼 송신합니다. 이 명령은 명령 실행 시에 CPU 유니트에서 시리얼 커뮤니케이션 유니트로 송신 데이터를 전송하기 때문에 고속 데이터 송신이 가능합니다.	○	○	×	×	×	×	×	×	3-733
	시리얼 커뮤니케이션 유니트 시리얼 포트 직접 수신	DRXDU/ @DRXDU	261	시리얼 커뮤니케이션 유니트 CJ1W-SCU22/32/42의 시리얼 통신 포트(무수순 모드)에서 할당 DM 영역 설정의 무수순 모드일 때의 시작 코드/종료 코드 지정에 따라 수신한 데이터를 지정된 시작 CH 번호부터 지정된 바이트 수만큼 출력합니다. 이 명령은 명령 실행 시에 시리얼 커뮤니케이션 유니트에서 CPU 유니트로 수신 데이터를 전송하기 때문에 고속 데이터 전송이 가능합니다.	○	○	×	×	×	×	×	×	3-738
	시리얼 포트 통신 설정 변경	STUP/ @STUP	237	CPU 유니트, 시리얼 커뮤니케이션 유니트 또는 시리얼 커뮤니케이션 보드의 시리얼 포트(페리페럴 포트 포함) 통신 설정을 변경합니다.	○	○	○	○	○	○	○	○	3-745
	프로토콜 매크로 송신 2	PMCR2/ @PMCR2	264	시리얼 커뮤니케이션 유니트에 등록된 송수신 시퀀스(프로토콜 데이터)를 호출하여 실행합니다.	○	○	×	×	×	×	×	×	3-748
네트워크 통신용 명령	네트워크 송신	SEND/ @SEND	090	네트워크상의 노드에 데이터를 송신합니다.	○	○	○	○	○	○	○	○	3-774
	네트워크 송신 2	SEND2/ @SEND2	491	네트워크상의 노드에 데이터를 송신합니다.	○	○	×	×	×	×	×	×	3-779
	네트워크 수신	RECV/ @RECV	098	네트워크상의 노드에 대해 송신을 요구하고 데이터를 수신합니다.	○	○	○	○	○	○	○	○	3-783
	네트워크 수신 2	RECV2/ @RECV2	492	네트워크상의 노드에 대해 송신을 요구하고 데이터를 수신합니다.	○	○	×	×	×	×	×	×	3-788
	명령 송신	CMND/ @CMND	490	임의의 FINS 명령을 전송하고 응답을 수신합니다.	○	○	○	○	○	○	○	○	3-791
	명령 송신 2	CMND2/ @CMND2	493	임의의 FINS 명령을 전송하고 응답을 수신합니다.	○	○	×	×	×	×	×	×	3-798
	범용 Explicit 메시지 송신 명령	EXPLT	720	임의의 ServiceCode의 Explicit 메시지를 송신합니다(DeviceNet 네트워크만 해당).	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-802
	Explicit 읽기 명령	EGATR	721	Explicit 메시지를 통해 정보/상태 읽기(단일 속성 가져오기, 서비스 코드: 0EHex)를 실행합니다(DeviceNet 네트워크만 해당).	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-807
	Explicit 쓰기 명령	ESATR	722	Explicit 메시지를 통해 정보 쓰기(단일 속성 설정, 서비스 코드: 10Hex)를 실행합니다 (DeviceNet 네트워크만 해당).	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-812
	Explicit CPU 유니트 데이터 읽기 명령	ECHRD	723	Explicit 메시지를 지원하는 네트워크상의 CPU 유니트 데이터를 자체 CPU 유니트로 읽어옵니다(DeviceNet 네트워크만 해당).	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-816
	Explicit CPU 유니트 데이터 쓰기 명령	ECHWR	724	자체 CPU 유니트에서 Explicit 메시지를 지원하는 네트워크상의 CPU 유니트에 데이터를 씁니다 (DeviceNet 네트워크만 해당).	○	○	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	Ver 2.0 이상	×	×	3-819

명령 종류	명령 이름	니모닉	펑션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
파일 메모리용 명령	데이터 파일 읽기	FREAD/ @FREAD	700	파일 메모리 내의 지정 데이터 파일(I/O 메모리 파일)의 지정 데이터 또는 데이터 수를 CPU 유니트 내의 지정된 I/O 메모리 영역에 읽어옵니다.	○	○	○	○	○	○	○	○	3-831
	데이터 파일 쓰기	FWRIT/ @FWRIT	701	지정된 I/O 메모리 영역의 데이터를 지정된 데이터 파일(I/O 메모리 파일)에 씁니다.	○	○	○	○	○	○	○	○	3-836
	텍스트 파일 쓰기	TWRIT/ @TWRIT	704	I/O 메모리 영역에 있는 ASCII 데이터를 메모리 카드에 텍스트 파일로 저장합니다.	○	○	Ver 4.0 이상	Ver 4.0 이상	Ver 4.0 이상	×	×	×	3-841
표시 기능용 명령	메시지 표시	MSG/ @MSG	046	지정된 CH에서 16채널(워드)분의 메시지(ASCII 코드)를 읽어서 프로그래밍 콘솔에 표시합니다.	○	○	○	○	○	○	○	○	3-848
시계 기능용 명령	캘린더 가산	CADD/ @CADD	730	시각 데이터에 시간 데이터를 가산합니다.	○	○	○	○	○	○	○	○	3-852
	캘린더 감산	CSUB/ @CSUB	731	시각 데이터에서 시간 데이터를 감산합니다.	○	○	○	○	○	○	○	○	3-852
	시분초→초 변환	SEC/ @SEC	065	시분초 데이터를 초 데이터로 변환합니다.	○	○	○	○	○	○	○	○	3-856
	초→시분초 변환	HMS/ @HMS	066	초 데이터를 시분초 데이터로 변환합니다.	○	○	○	○	○	○	○	○	3-858
	시계 보정	DATE/ @DATE	735	지정된 시계 데이터에 따라 내부 시계의 값을 변경합니다.	○	○	○	○	○	○	○	○	3-860
디버그 처리 명령	트레이스 메모리 샘플링	TRSM	045	이 명령을 실행하면 접점 또는 채널 데이터를 샘플링하여 트레이스 메모리에 저장합니다.	○	○	○	○	○	○	○	○	3-864
고장 진단 명령	운전 계속 고장 진단	FAL/ @FAL	006	사용자 정의 운전 계속 이상을 등록 또는 해제합니다.	○	○	○	○	○	○	○	×	3-868
	운전 정지 고장 진단	FALS	007	사용자 정의 운전 정지 이상을 등록합니다.	○	○	○	○	○	○	○	×	3-874
	고장 지점 검출	FPD	269	지정한 1회로에 대해 시간 감시 진단 및 논리 진단을 수행합니다.	○	○	○	○	○	○	○	○	3-879
특수 명령	세트 캐리	STC/@STC	040	CY 플래그를 ON합니다.	○	○	○	○	○	○	○	○	3-888
	클리어 캐리	CLC/@CLC	041	CY 플래그를 OFF합니다.	○	○	○	○	○	○	○	○	3-888
	확장 DM 뱅크 전환	EMBC/ @EMBC	281	EM(확장 DM) 영역의 현재 뱅크를 전환합니다.	○	○	○	○	○	○	○	○	3-889
	사이클 시간 감시 시간 설정	WDT/ @WDT	094	이 명령을 실행한 사이클만 사이클 시간 감시 시간을 연장합니다.	○	○	○	○	○	○	○	○	3-891
	조건 플래그 저장	CCS/ @CCS	282	조건 플래그의 상태를 보관합니다.	○	○	○	○	○	○	○	×	3-893
	조건 플래그 로드	CCL/ @CCL	283	보관된 조건 플래그의 상태를 읽어옵니다.	○	○	○	○	○	○	○	×	3-893
	CV→CS 주소 변환	FRMCV/ @FRMCV	284	CVM1/CV 시리즈의 I/O 메모리 실제 주소를 CS/CJ 시리즈의 I/O 메모리 실제 주소로 변환합니다.	○	○	○	○	○	○	○	×	3-895
	CS→CV 주소 변환	TOCV/ @TOCV	285	CS/CJ 시리즈의 I/O 메모리 실제 주소를 CVM1/CV 시리즈의 I/O 메모리 실제 주소로 변환합니다.	○	○	○	○	○	○	○	×	3-898
	주변 서비스 금지	IOSP/ @IOSP	287	병렬 처리 모드 또는 주변 서비스 우선 모드에서 명령 실행 중인 주변 서비스를 금지합니다.	○	○	○	○	○	○	×	×	3-901
	주변 서비스 금지 해제	IORS	288	병렬 처리 모드 또는 주변 서비스 우선 모드에서 IOSP(주변 서비스 실행 금지) 명령에 의해 지정된 주변 서비스 실행 금지를 해제합니다.	○	○	○	○	○	○	×	×	3-901
블록 프로그램 명령	블록 프로그램	BPRG	096	지정된 번호의 블록 프로그램 영역의 선두를 나타냅니다.	○	○	○	○	○	○	○	○	3-908
	블록 프로그램 종료	BEND	801	블록 프로그램 영역의 종료를 나타냅니다.	○	○	○	○	○	○	○	○	3-908
	블록 프로그램 일시 정지	BPPS	811	실행 중인 지정된 블록 프로그램을 일시 정지시킵니다.	○	○	○	○	○	○	○	○	3-910
	블록 프로그램 재기동	BPRS	812	정지 중인 지정된 블록 프로그램 번호의 블록 프로그램을 재기동시켜서 실행합니다.	○	○	○	○	○	○	○	○	3-910

명령 종류	명령 이름	니모닉	평션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
블록 프로그램 명령	조건부 종료	EXIT	806	입력 조건이 ON일 때 EXIT 명령부터 BEND 명령까지의 프로그램을 실행시키지 않고 블록 프로그램을 종료합니다.	○	○	○	○	○	○	○	○	3-912
	조건부 종료(부정)	EXIT NOT	806	지정된 접점이 OFF일 때 EXIT 명령부터 BEND 명령까지의 프로그램을 실행시키지 않고 블록 프로그램을 종료합니다.	○	○	○	○	○	○	○	○	3-912
	조건 분기 블록	IF	802	입력 조건이 OFF이면 다음 명령 이후를 실행하고 ON이면 ELSE 명령까지의 명령을 무시합니다.	○	○	○	○	○	○	○	○	3-914
	조건 분기 블록(부정)	IF NOT	802	지정된 접점이 ON이면 다음 명령 이후를 실행하고 OFF이면 ELSE 명령까지의 명령을 무시합니다.	○	○	○	○	○	○	○	○	3-914
	조건 분기 거짓 블록	ELSE	803	IF 조건이 거짓일 때 실행되는 블록의 선두를 나타냅니다.	○	○	○	○	○	○	○	○	3-914
	조건 분기 블록 종료	IEND	804	조건 분기 블록 영역의 종료를 나타냅니다.	○	○	○	○	○	○	○	○	3-914
	1스캔 조건 대기	WAIT	805	입력 조건이 ON이 될 때까지 WAIT 명령부터 BEND 명령까지의 프로그램을 실행하지 않습니다.	○	○	○	○	○	○	○	○	3-917
	1스캔 조건 대기(부정)	WAIT NOT	805	지정된 접점이 OFF가 될 때까지 WAIT 명령부터 BEND 명령까지의 프로그램을 실행하지 않습니다.	○	○	○	○	○	○	○	○	3-917
	타이머 대기	TIMW	813	지정된 타이머가 타임업될 때까지 다음 명령부터 BEND 명령까지의 프로그램을 실행하지 않습니다.	○	○	○	○	○	○	○	○	3-920
		TIMWX	816		○	○	○	○	○	○	○	×	3-920
	카운터 대기	CNTW	814	지정된 카운터가 카운트 업될 때까지 다음 명령부터 BEND 명령까지의 프로그램을 실행하지 않습니다.	○	○	○	○	○	○	○	○	3-923
		CNTWX	818		○	○	○	○	○	○	○	×	3-923
	고속 타이머 대기	TMHW	815	지정된 타이머가 타임업될 때까지 다음 명령부터 BEND 명령까지의 프로그램을 실행하지 않습니다.	○	○	○	○	○	○	○	○	3-925
		TMHWX	817		○	○	○	○	○	○	○	×	3-925
	반복 블록	LOOP	809	반복 블록 영역(LOOP~LEND)의 선두를 나타냅니다.	○	○	○	○	○	○	○	○	3-927
	반복 블록 종료	LEND	810	반복 블록 영역(LOOP~LEND)의 종료를 나타냅니다.	○	○	○	○	○	○	○	○	3-927
	반복 블록 종료(부정)	LEND NOT	810	반복 블록 영역(LOOP~LEND NOT)의 종료를 나타냅니다.	○	○	○	○	○	○	○	○	3-927
문자열 처리 명령	문자열 전송	MOV$/ @MOV$	664	문자열을 전송합니다.	○	○	○	○	○	○	○	○	3-934
	문자열 연결	+$/ @+$	656	문자열과 문자열을 연결합니다.	○	○	○	○	○	○	○	○	3-936
	왼쪽에서 문자열 가져오기	LEFT$/ @LEFT$	652	문자열의 왼쪽(선두)부터 지정된 문자 수만큼 가져옵니다.	○	○	○	○	○	○	○	○	3-938
	오른쪽에서 문자열 가져오기	RGHT$/ @RGHT$	653	문자열의 오른쪽(끝)부터 지정된 문자 수만큼 가져옵니다.	○	○	○	○	○	○	○	○	3-938
	임의 위치에서 문자열 가져오기	MID$/ @MID$	654	문자열의 임의의 위치부터 지정된 수만큼 가져옵니다.	○	○	○	○	○	○	○	○	3-940
	문자열 검색	FIND$/ @FIND$	660	문자열에서 지정된 문자열을 검색하고 그 위치를 출력합니다.	○	○	○	○	○	○	○	○	3-942
	문자열 길이 검출	LEN$/ @LEN$	650	문자열의 길이를 계산합니다.	○	○	○	○	○	○	○	○	3-944
	문자열 치환	RPLC$/ @RPLC$	661	지정된 위치의 문자열을 지정된 문자로 치환합니다.	○	○	○	○	○	○	○	○	3-946
	문자열 삭제	DEL$/ @DEL$	658	문자열에서 지정된 문자열을 삭제합니다.	○	○	○	○	○	○	○	○	3-948
	문자열 교환	XCHG$/ @XCHG$	665	문자열과 문자열을 교환합니다.	○	○	○	○	○	○	○	○	3-950
	문자열 클리어	CLR$/ @CLR$	666	문자열을 모두 NUL(00Hex)로 클리어합니다.	○	○	○	○	○	○	○	○	3-952

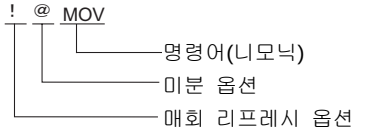
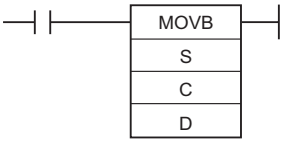
명령 종류	명령 이름	니모닉	평션 번호	기능 설명	CJ2H	CJ2M	CJ1-H/ CS1-H	CJ1M		CS1D		CJ1/ CS1	페이지
								-CPU 1□	-CPU 2□	-S	-H		
문자열 처리 명령	문자열 삽입	INS$/ @INS$	657	지정된 문자열 위치에 지정된 문자열을 삽입합니다.	○	○	○	○	○	○	○	○	3-954
	문자열 비교	LD, AND, OR+=$	670	2개의 문자열 데이터를 ASCII 코드의 크기로 선두 문자부터 비교(=, <>, <, <=, >, >= 중 하나)하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다.	○	○	○	○	○	○	○	○	3-956
		LD, AND, OR+<>$	671		○	○	○	○	○	○	○	○	3-956
		LD, AND, OR+<$	672		○	○	○	○	○	○	○	○	3-956
	문자열 비교	LD, AND, OR+>$	674	2개의 문자열 데이터를 ASCII 코드의 크기로 선두 문자부터 비교(=, <>, <, <=, >, >= 중 하나)하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다.	○	○	○	○	○	○	○	○	3-956
		LD, AND, OR+>=$	675		○	○	○	○	○	○	○	○	3-956
태스크 제어 명령	태스크 실행 기동	TKON/ @TKON	820	태스크를 실행 가능 상태로 설정합니다.	○	○	○	○	○	○	○	○	3-962
	태스크 실행 대기	TKOF/ @TKOF	821	태스크를 대기 상태로 설정합니다.	○	○	○	○	○	○	○	○	3-962
기종 변환용 명령	블록 전송	XFERC/ @XFERC	565	연속된 여러 CH 데이터를 일괄 전송합니다.	○	○	Ver 3.0 이상	Ver 3.0 이상	Ver 3.0 이상	×	×	×	3-970
	데이터 분배	DISTC/ @DISTC	566	전송 대상을 기준으로 오프셋 지정된 채널로 전송 데이터를 전송합니다.	○	○	Ver 3.0 이상	Ver 3.0 이상	Ver 3.0 이상	×	×	×	3-972
	데이터 추출	COLLC/ @COLLC	567	전송 원본을 기준으로 오프셋 지정된 채널의 내용을 지정 채널로 전송합니다.	○	○	Ver 3.0 이상	Ver 3.0 이상	Ver 3.0 이상	×	×	×	3-975
	비트 전송	MOVBC/ @MOVBC	568	지정한 비트를 전송합니다.	○	○	Ver 3.0 이상	Ver 3.0 이상	Ver 3.0 이상	×	×	×	3-979
	비트 카운터	BCNTC/ @BCNTC	621	CH 데이터 내에서 "1"인 비트의 총 수를 계산합니다.	○	○	Ver 3.0 이상	Ver 3.0 이상	Ver 3.0 이상	×	×	×	3-981
평션 블록용 특수 명령	변수 종류 취득	GETID/ @GETID	286	지정된 변수(또는 주소)의 FINS 명령의 변수 종류(I/O 메모리 영역 종류) 코드 및 CH 번호를 출력합니다.	○	○	Ver 3.0 이상	Ver 3.0 이상	Ver 3.0 이상	×	×	×	3-984
SFC 관련 명령	스텝 활성화 명령	SA/ @SA	784	지정된 스텝 및 서브 차트를 활성화 상태로 하고 작업의 실행을 시작합니다.	○	○	Ver 4.0 이상	Ver 4.0 이상	Ver 4.0 이상	×	×	×	3-990
	스텝 비활성화 명령	SE/ @SE	785	지정된 스텝 및 서브 차트를 비활성화 상태로 하고 작업의 실행을 정지시킵니다.	○	○	Ver 4.0 이상	Ver 4.0 이상	Ver 4.0 이상	×	×	×	3-990
	스텝 타이머 현재값 읽기 명령	TSR/ @TSR	780	S로 지정한 스텝의 스텝 타이머 현재값을 D에 저장합니다.	○	○	Ver 4.0 이상	Ver 4.0 이상	Ver 4.0 이상	×	×	×	3-992
	스텝 타이머 현재값 변경 명령	TSW/ @TSW	781	D로 지정한 스텝의 스텝 타이머 값을 S로 지정한 값으로 변경합니다.	○	○	Ver 4.0 이상	Ver 4.0 이상	Ver 4.0 이상	×	×	×	3-992
	SFC 태스크 실행 명령	SFCON	789	SFC 태스크 제어 명령에 의해 종료 또는 일시 정지된 SFC 태스크의 실행을 재개합니다.	○	○	Ver 4.0 이상	Ver 4.0 이상	Ver 4.0 이상	×	×	×	3-994
	SFC 태스크 종료 명령	SFCOFF	790	SFC 프로그램의 실행 상태를 종료시킵니다. 이때, 출력은 모두 유지된 상태에서 종료합니다. 종료한 SFC 태스크의 실행을 시작하면 초기화 스텝부터 처리가 시작됩니다.	○	○	Ver 4.0 이상	Ver 4.0 이상	Ver 4.0 이상	×	×	×	3-994
	SFC 태스크 정지 명령(리셋 있음)	SFCPR	793	N으로 지정한 SFC 태스크를 일시 정지합니다. 이때, 정지한 SFC 태스크의 출력은 모두 리셋됩니다.	○	○	Ver 4.0 이상	Ver 4.0 이상	Ver 4.0 이상	×	×	×	3-994
	SFC 태스크 정지 명령(리셋 없음)	SFCPRN	791	N으로 지정한 SFC 태스크를 일시 정지합니다. 이때, 정지한 SFC 태스크의 출력은 모두 유지됩니다.	○	○	Ver 4.0 이상	Ver 4.0 이상	Ver 4.0 이상	×	×	×	3-996

제3장

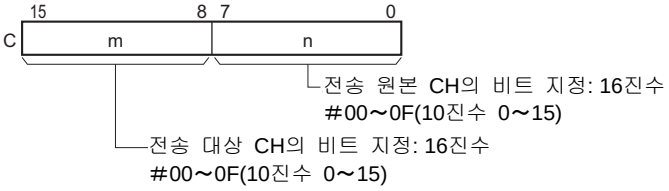
각 명령에 대한 설명

이 장을 보는 방법

명령어의 순서: 기능 순서로 배열되어 있습니다.
니모닉에서 찾을 때는 '부록 ABC 순서 명령 목록'을 참조하십시오.
각 명령어의 설명 항목은 다음과 같습니다.

항목	내용
명령 이름	명령어의 명칭을 나타냅니다. 예: 비트 전송
니모닉	니모닉을 나타냅니다. 예: MOVB
동작 옵션	동작 옵션별 니모닉을 나타냅니다. 미분 옵션 @: 입력 기동 미분형 명령이라고 합니다. %: 입력 종료 미분형 명령이라고 합니다. 매회 리프레시 옵션 ！: 명령 실행 시, 피연산자를 지정한 입출력 영역의 데이터를 I/O 리프레시 합니다. 
펑션 번호	펑션 번호를 나타냅니다.
기능	명령어의 기능을 나타냅니다.
심볼	CX-Programmer에서 사용하는 래더 심볼을 나타냅니다. 예: 
특정 영역 내의 사용 여부	다음의 각 영역 내에서 해당 명령을 사용할 수 있는지 여부를 나타냅니다. 펑션 블록 정의(FB) 정의 내, 블록 프로그램 영역, 공정 진행 프로그램 영역, 서브루틴 프로그램 영역, 인터럽트 태스크 프로그램 영역 SFC 작업/트랜지션 영역 가능: 사용 가능. 불가: 사용 불가능 예:

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명	피연산자의 내용, 데이터 형식, 크기를 나타냅니다. 컨트롤 데이터 등의 설명이 필요한 피연산자에 대해 해당 채널/비트의 의미를 나타냅니다. 예: 
영역 종류	각 피연산자에 지정할 수 있는 I/O 메모리의 영역 종류를 나타냅니다. 오른쪽 열의 '내용'은 영역 종류를 나타냅니다. 왼쪽 열의 S, C, D 등은 위의 피연산자 설명 항목에서 기재한 피연산자 기호를 나타냅니다. ○: 사용 가능. －: 사용 불가능 예:

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	－	○	－	－	－	－
C																		
D											—							

<table>
<tr><th>항목</th><th>내용</th></tr>
<tr><td>조건 플래그의 동작</td><td>해당 명령 실행 시(실행 직후), 조건 플래그가 어떻게 변하는지를 나타냅니다. 변하는 플래그만 기재됩니다. 기재되지 않은 플래그는 해당 명령 실행에 의해 변하지 않음을 의미합니다.
주: 단순히 'OFF'인 플래그는 명령 실행 시(실행 직후) 무조건 OFF로 된다는 것을 의미합니다.
예:
<table>
<tr><th>이름</th><th>라벨</th><th>내용</th></tr>
<tr><td>에러 플래그</td><td>P_ER</td><td>· C의 데이터가 범위 밖일 때 ON
· 그 이외일 때 OFF</td></tr>
<tr><td>=플래그</td><td>P_EQ</td><td>OFF</td></tr>
<tr><td>부정 플래그</td><td>P_N</td><td>OFF</td></tr>
</table>
</td></tr>
<tr><td>기능</td><td>명령어의 기능을 나타냅니다.</td></tr>
<tr><td>힌트</td><td>기본 기능 이외의 보충 설명을 나타냅니다.</td></tr>
<tr><td>사용상의 주의 사항</td><td>사용상의 주의 사항을 나타냅니다.</td></tr>
<tr><td>샘플 프로그램</td><td>피연산자에 구체적인 주소나 상수를 입력한 예를 표시합니다. 기본적으로 기능 설명과 대응합니다.</td></tr>
</table>

■ 상수 표기

이 장에서는 피연산자인 상수를 다음과 같이 표기합니다.

● '피연산자 설명' 및 '데이터 내용' 항목에서

· 비트 열을 나타내는(일반적으로 16진수로 입력) 피연산자인 경우

16진수로만 표기합니다. 예) MOV 명령의 피연산자 S: 상수의 범위로 #0000~FFFF만 표시합니다.
(단, 주변 툴에서는 접두사 &를 붙여서 10진수로 입력할 수도 있습니다.)

· 수를 나타내는(일반적으로 10진수로 입력) 피연산자인 경우(점프 명령 포함)

16진수 및 10진수로 표기합니다. 예) XFER 명령의 피연산자 W: 상수의 범위로 #0000~FFFF 및 10진수 &0~65535를 표시합니다.

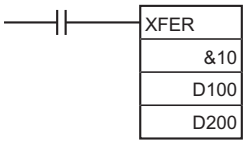
· 번호를 나타내는 피연산자의 경우(점프 명령 제외)

10진수로 표기합니다. 예) SBS 명령의 피연산자 N: 상수의 범위로 0~1023을 표시합니다.

● '동작 설명' 항목에서

'동작 설명' 래더도에서는 CX-Programmer에 따른 입력 표기로 나타냅니다.

주: 수를 나타내는(일반적으로 10진수로 입력) 피연산자인 경우 접두사 &를 붙인 10진수로 표시합니다.
예)



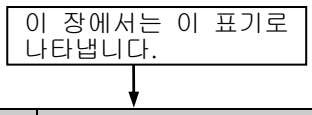
참고: 주변 툴을 사용한 상수 입력 방법은 다음과 같습니다.

주변 툴	CX-Programmer	프로그래밍 콘솔
비트 열을 나타내는 (일반적으로 16진수로 입력) 피연산자	접두사 &를 붙인 10진수를 입력합니다. 또는 접두사 #을 붙인 16진수로 입력합니다. (주)	[#상수] 키를 누르면 기본적으로 #가 붙은 16진수로 입력됩니다.
수를 나타내는(일반적으로 10진수로 입력) 피연산자		[표시 전환] 키를 누를 때마다 16진수(# 표시 있음)→부호 있는 10진수(+ − 표시 있음)→부호 없는 10진수(& 표시 있음)로 표시됩니다.
번호를 나타내는 피연산자 (점프 명령 제외)	접두사 #을 붙인 10진수를 입력합니다. (주)	그대로 10진수로 입력합니다. · 접두사 &가 자동으로 표시되는 경우: [표시 전환] 키를 누를 때마다 부호 없는 10진수(& 표시 있음)→16진수(# 표시 있음)→부호 있는 10진수(+ − 표시 있음)로 표시됩니다. · 접두사가 없는 경우: 10진수 입력만 가능합니다.

주: CX-Programmer에서는 피연산자 입력 시, 입력 가능한 데이터 범위(접두사 포함)가 표시됩니다.

■ 조건 플래그 표기

이 장의 설명에서는 조건 플래그를 C-X-Programmer의 표시 라벨로 표기합니다.
또한 프로그래밍 콘솔에서는 오른쪽에 나온 것처럼 표시됩니다.



이름	CX-Programmer의 이름	프로그래밍 콘솔의 표시 라벨
에러 플래그	P_ER	ER
액세스 에러 플래그	P_AER	AER
캐리 플래그	P_CY	CY
> 플래그	P_GT	>
= 플래그	P_EQ	=
< 플래그	P_LT	<
부정 플래그	P_N	N
오버 플로 플래그	P_OF	OF
언더 플로 플래그	P_UF	UF
≧ 플래그	P_GE	> =
≠ 플래그	P_NE	< >
≦ 플래그	P_LE	< =
상시 ON 플래그	P_On	ON
상시 OFF 플래그	P_Off	OFF

■ 심볼화 명령

C/CV 시리즈의 일부 명령어는 CS/CJ 시리즈에서 동등한 기능을 가진 다른 명령으로 바뀝니다.

	C/CV 시리즈	CS/CJ 시리즈
시퀀스 제어	JMP #0/JME #0	JMP0/JME0
데이터 비교	EQU	AND=
데이터 전송	MOVQ	MOV
증가/감소	INC	++B
	INCL	++BL
	INCB	++
	INBL	++L
	DEC	--B
	DECL	--BL
	DECB	--
	DCBL	--L
사칙연산	ADB	+C
	ADBL	+CL
	ADD	+BC
	ADDL	+BCL
	SBB	－C
	SBBL	－CL
	SUB	－BC
	SUBL	－BCL
	MBS	*
	MBSL	*L
	MLB	*U
	MUL	*B
	MULL	*BL
	DBS	/
	DBSL	/L
	DVB	/U
	DIV	/B
	DIVL	/BL
인터럽트 제어	INT	MSKS/MSKR/CLI DI/EI

시퀀스 입력 명령

니모닉	명령 이름	평션 번호	페이지
LD	로드	–	3-8
LD NOT	로드 부정	–	
AND	논리곱	–	3-10
AND NOT	부정 논리곱	–	
OR	논리합	–	3-12
OR NOT	부정 논리합	–	
AND LD	로드 논리곱	–	3-14
OR LD	로드 논리합	–	
NOT	부정	520	3-17
UP	P.F. 기동 미분	521	3-18
DOWN	P.F. 종료 미분	522	
LD TST	LD 비트 테스트	350	3-20
LD TSTN	LD 비트 테스트 부정	351	
AND TST	AND 비트 테스트	350	3-22
AND TSTN	AND LD 비트 테스트 부정	351	
OR TST	OR 비트 테스트	350	3-24
OR TSTN	OR 비트 테스트 부정	351	

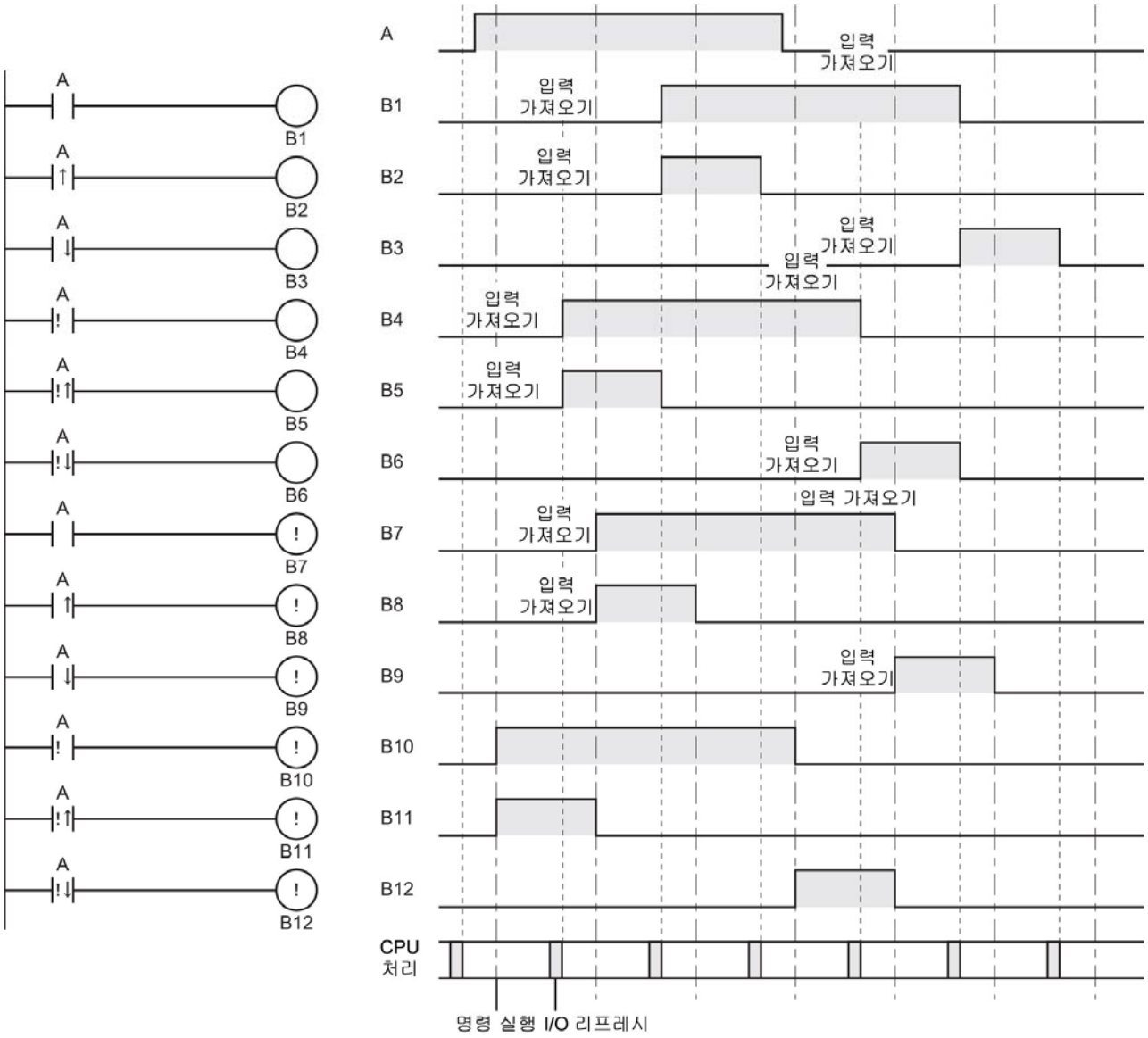
미분형 명령과 매회 리프레시형 명령

- LD/AND/OR 명령에는 일반적인 명령 외에 미분형 명령과 매회 리프레시형 명령, 그리고 두 가지를 조합한 명령이 있습니다.
- LD NOT/AND NOT/OR NOT/OUT/OUT NOT 명령에는 일반적인 명령 외에 매회 리프레시형 명령이 있습니다.
- 일반적인 명령과 미분형 명령, 매회 리프레시형 명령과 미분 매회 리프레시형 명령은 명령에서 처리하는 데이터 입출력 타이밍이 다릅니다.
- 일반적인 명령 및 미분형 명령은 해당 명령을 실행하기 이전의 I/O 리프레시 처리에 의해 입력된 데이터를 사용하여 실행되며, 그 결과는 다음 I/O 리프레시 처리에서 출력됩니다. 여기서 말하는 I/O 리프레시란 CPU 내부 메모리와 입출력 유니트가 데이터를 주고받는 것을 의미합니다.
- 매회 리프레시형 명령은 위의 I/O 리프레시에 더하여, 입력 명령에서는 명령 실행 직전에, 출력 명령에서는 명령 실행 직후에 명령에서 액세스한 채널에 대해서만 입출력 유니트와 데이터를 주고받습니다.
  접점의 매회 리프레시형 명령은 지정된 접점을 포함하여 16 비트분을 동시에 리프레시합니다.
  SYSBUS 원격 I/O 자국 장치의 유니트에 대해서는 매회 리프레시 명령을 사용할 수 없습니다.

명령의 형식	명령어	기능	I/O 리프레시
일반 명령	LD/AND/OR LD NOT/AND NOT/ OR NOT	지정된 접점의 ON/OFF 상태를 주기적 리프레시로 CPU에 전송하여 직후의 명령 실행에 반영합니다.	주기적 리프레시
	OUT/OUT NOT	명령 직후, 지정된 접점의ON/OFF 상태를 직후의 주기적 리프레시 시에 출력합니다.	
기동 미분형 명령	@LD/@AND/@OR	지정된 접점의 ON 기동(OFF→ON)이면 명령을 실행하고 1사이클 ON합니다.	
종료 미분형 명령	%LD/%AND/%OR	지정된 접점의 OFF 종료(ON→OFF)이면 명령을 실행하고 1사이클 ON합니다.	
매회 리프레시형 명령	!LD/!AND/!OR/ !LD NOT/!AND NOT/ !OR NOT	지정된 접점의 입력 데이터를 CPU에 전송하여 명령을 실행합니다.	명령 실행 전
	!OUT/!OUT NOT	명령을 실행한 후 지정된 접점에 데이터를 출력합니다.	명령 실행 후
기동 미분 및 매회 리프레시형 명령	!@LD/!@AND/!@OR	지정된 접점의 입력 데이터를 CPU에 전송하여 ON 기동(OFF→ON)이면 명령을 실행하고 1사이클 ON합니다.	명령 실행 전
종료 미분 및 매회 리프레시형 명령	!%LD/!%AND/!%OR	지정된 접점의 입력 데이터를 CPU에 전송하여 OFF 종료(ON→OFF)이면 명령을 실행하고 1사이클 ON합니다.	

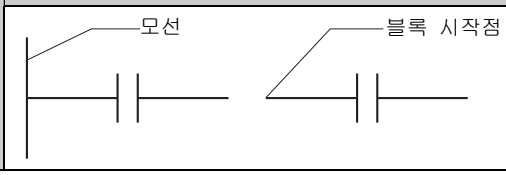
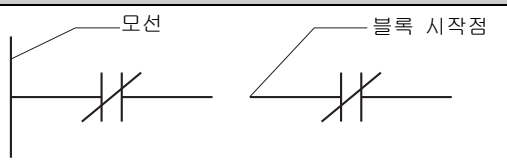
■ 각 입출력 명령의 동작 타이밍

LD 명령과 OUT 명령으로 구성된 프로그램에 대해 각 명령의 동작 차이를 타이밍 차트로 나타냅니다.



LD/LD NOT

명령 이름	니모닉	동작 옵션	펑션 번호	기능
로드	LD	@LD, %LD, !LD, !@LD, !%LD	–	논리 시작을 나타냅니다. 접점의 ON/OFF 내용을 전송합니다.
로드 부정	LD NOT	@LD NOT, %LD NOT, ! LD NOT, !@LD NOT, !%LD NOT	–	논리 시작을 나타냅니다. 접점의 ON/OFF 내용을 반전하여 전송합니다.

심볼	LD	LD NOT
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
–	–	BOOL	–

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
LD의 비트 피연산자	○	○	○	○	○	○	○*1	○*1	–	–	–	–	–	○	○	○	○	○
LD NOT의 비트 피연산자																		–

*1: CJ2 CPU 유니트만 사용 가능

조건 플래그의 동작

없음

기능

■ LD

모선에서 시작되는 최초의 a 접점 또는 회로 블록의 최초의 a 접점에서 사용합니다.
매회 리프레시를 지정하지 않은 경우, I/O 메모리의 지정된 비트 내용을 읽습니다. 매회 리프레시를 지정한 경우에는 기본 입력 유니트의 입력 단자의 실제 접점 상태를 읽습니다.

■ LD NOT

모선에서 시작되는 최초의 b 접점 또는 회로 블록의 최초의 b 접점에서 사용합니다.
매회 리프레시를 지정하지 않은 경우, I/O 메모리의 지정된 비트 내용을 반전하여 읽습니다. 매회 리프레시를 지정한 경우에는 기본 입력 유니트의 입력 단자의 실제 접점 상태를 반전하여 읽습니다.

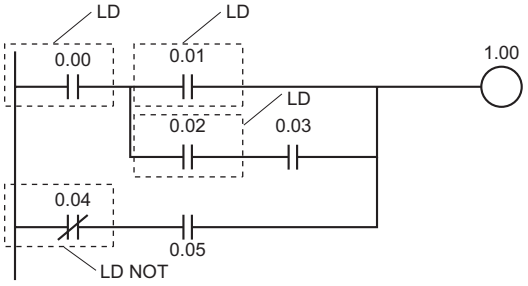
힌트

- LD/LD NOT 명령은 논리 시작을 나타내는 명령이며 다음 중 하나에 해당할 때 사용합니다.
  1) 모선에 직접 연결할 때
  2) AND LD 명령 또는 OR LD 명령에 의해 회로 블록을 연결했을 때(회로 블록의 선두에서 사용). (AND LD 명령 또는 OR LD 명령은 본 LD/LD NOT 명령에서 시작되는 회로 블록을 직렬 또는 병렬로 연결합니다.)
- 출력계 명령 중 모선에 직접 연결할 수 없는 명령은 입력 조건에 LD/LD NOT 명령이 하나 이상 반드시 필요합니다. LD/LD NOT 명령이 없으면 주변 툴을 사용한 프로그램 확인에서 회로 에러가 발생합니다.
- AND LD 명령 또는 OR LD 명령을 사용하여 회로 블록을 연결하는 경우, AND LD/OR LD 명령을 합한 수와 LD/LD NOT 명령의 합계에서 1을 뺀 수가 일치해야 합니다. 일치하지 않으면 회로 에러가 발생합니다. (자세한 내용은 AND LD/OR LD 항목을 참조하십시오.)

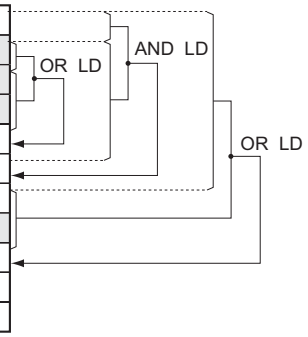
사용상의 주의 사항

- LD 명령은 기동 미분 또는 종료 미분을 지정(@ 또는 %)할 수 있습니다. 이 경우, 지정된 접점에서 OFF→ON 또는 ON→OFF의 변화가 있을 때만 1사이클 ON됩니다. 1사이클 후에는 OFF가 됩니다.
- LD/LD NOT 명령은 매회 리프레시를 지정(!)할 수 있습니다. 이 경우, 이 명령 실행 직전에 기본 입력 유니트(단, C200H 그룹 2 다점 입력 유니트, 원격 I/O 자국 장치의 기본 입력 유니트 제외)에 대해 IN 리프레시를 실행하고 그 내용을 직접 읽습니다.
- LD 명령은 기동 미분 또는 종료 미분과 매회 리프레시를 복합 지정(!@ 또는 !%)할 수 있습니다. 이 경우, 이 명령 실행 직전에 기본 입력 유니트에서 IN 리프레시로 직접 읽은 내용에 OFF→ON 또는 ON→OFF의 변화가 있을 때만 1사이클 ON됩니다. 1사이클 후에는 OFF가 됩니다.
- CJ2 CPU 유니트에서는 데이터 메모리(DM) 및 확장 데이터 메모리(EM)의 비트 주소가 가능하므로 LD/LD NOT 명령을 사용할 수 있습니다. 기타 기종은 LD/LD NOT 명령을 사용할 수 없으므로 LD TST 명령을 사용하십시오.

샘플 프로그램



명령	데이터
LD	0.00
LD	0.01
LD	0.02
AND	0.03
OR LD	—
AND LD	—
LD NOT	0.04
AND	0.05
OR LD	—
OUT	1.00



AND/AND NOT

명령 이름	니모닉	동작 옵션	평션 번호	기능
논리곱	AND	@AND, %AND, !AND, !@AND, !%AND	–	접점의 ON/OFF 내용과 이전 입력 조건의 논리곱을 계산합니다.
부정 논리곱	AND NOT	@AND NOT, %AND NOT, !AND NOT, !@AND NOT, !%AND NOT	–	접점의 ON/OFF 내용을 반전하여 이전 입력 조건과의 논리곱을 계산합니다.

심볼	AND	AND NOT
		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
–	–	BOOL	–

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
AND의 비트 피연산자	○	○	○	○	○	○	○*1	○*1	–	–	–	–	–	○	○	○	○	–
AND NOT의 비트 피연산자																		

*1: CJ2 CPU 유니트만 사용 가능

조건 플래그의 동작

없음

기능

■ AND

직렬 연결하는 a 접점에서 사용합니다. 모선에 직접 연결할 수는 없습니다.
또한 회로 블록의 선두에서 사용할 수도 없습니다. 매회 리프레시를 지정하지 않은 경우, I/O 메모리의 지정된 비트 내용을 읽습니다. 매회 리프레시를 지정한 경우에는 기본 입력 유니트의 입력 단자의 실제 접점 상태를 읽습니다.

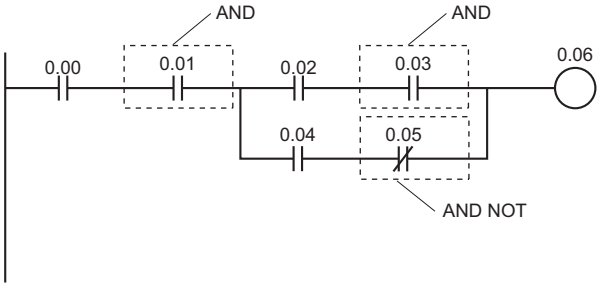
■ AND NOT

직렬 연결하는 b 접점에서 사용합니다. 모선에 직접 연결할 수는 없습니다.
또한 회로 블록의 선두에서 사용할 수도 없습니다. 매회 리프레시를 지정하지 않은 경우, I/O 메모리의 지정된 비트 내용을 읽습니다. 매회 리프레시를 지정한 경우에는 기본 입력 유니트의 입력 단자의 실제 접점 상태를 반전하여 읽습니다.

사용상의 주의 사항

- AND 명령은 기동 미분 또는 종료 미분을 지정(@ 또는 %)할 수 있습니다. 이 경우, 지정된 접점에서 OFF→ON 또는 ON→OFF의 변화가 있을 때만 1사이클 ON됩니다. 1사이클 후에는 OFF가 됩니다.
- AND/AND NOT 명령은 매회 리프레시를 지정(!)할 수 있습니다. 이 경우, 이 명령 실행 직전에 기본 입력 유니트(단, C200H 그룹 2 다점 입력 유니트, 원격 I/O 자국 장치의 기본 입력 유니트 제외)에 대해 IN 리프레시를 실행하고 그 내용을 직접 읽습니다.
- AND 명령은 기동 미분 또는 종료 미분과 매회 리프레시를 복합 지정(!@ 또는 !%)할 수 있습니다. 이 경우, 이 명령 실행 직전에 기본 입력 유니트에서 IN 리프레시로 직접 읽은 내용에 OFF→ON 또는 ON→OFF의 변화가 있을 때만 1사이클 ON됩니다. 1사이클 후에는 OFF가 됩니다.
- CJ2 CPU 유니트에서는 데이터 메모리(DM) 및 확장 데이터 메모리(EM)의 비트 주소가 가능하므로 AND/AND NOT 명령을 사용할 수 있습니다. 기타 기종은 AND/AND NOT 명령을 사용할 수 없으므로 AND TST 명령을 사용하십시오.

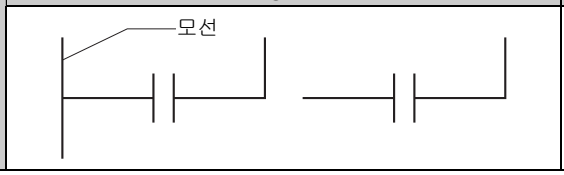
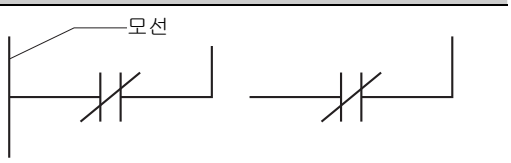
샘플 프로그램



명령	데이터
LD	0.00
AND	0.01
LD	0.02
AND	0.03
LD	0.04
AND NOT	0.05
OR LD	—
AND LD	—
OUT	0.06

OR/OR NOT

명령 이름	니모닉	동작 옵션	펑션 번호	기능
논리합	OR	@OR, %OR, !OR, !@OR, !%OR	–	접점의 ON/OFF 내용과 이전 입력 조건과의 논리합을 계산합니다.
부정 논리합	OR NOT	@OR NOT, %OR NOT, !OR NOT, !@OR NOT, !%OR NOT	–	접점의 ON/OFF 내용을 반전하여 이전 입력 조건과의 논리합을 계산합니다.

심볼	OR	OR NOT
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
–	–	BOOL	–

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
OR의 비트 피연산자	○	○	○	○	○	○	○*1	○*1	–	–	–	–	–	○	○	○	○	–
OR NOT의 비트 피연산자																		

*1: CJ2 CPU 유니트만 사용 가능

조건 플래그의 동작

없음

기능

■ OR

병렬 연결하는 a 접점에서 사용합니다.
(모선에 연결된 경우 또는 회로 블록 선두에서) LD/LD NOT 명령부터 시작하여 이 명령 직전까지의 회로와 OR(논리합)의 a 접점을 구성합니다.
매회 리프레시를 지정하지 않은 경우, I/O 메모리의 지정된 비트 내용을 읽습니다. 매회 리프레시를 지정한 경우에는 기본 입력 유니트의 입력 단자의 실제 접점 상태를 읽습니다.

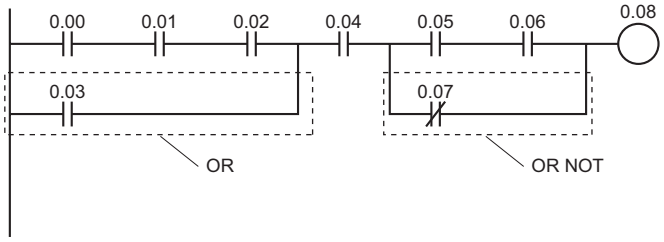
■ OR NOT

병렬 연결하는 b 접점에서 사용합니다.
(모선에 연결된 경우 또는 회로 블록 선두에서) LD/LD NOT 명령부터 시작하여 이 명령 직전까지의 회로와 OR(논리합)의 b 접점을 구성합니다.
매회 리프레시를 지정하지 않은 경우, I/O 메모리의 지정된 비트 내용을 읽습니다. 매회 리프레시를 지정한 경우에는 기본 입력 유니트의 입력 단자의 실제 접점 상태를 읽습니다.

사용상의 주의 사항

- OR 명령은 기동 미분 또는 종료 미분을 지정(@ 또는 %)할 수 있습니다. 이 경우, 지정된 접점에서 OFF→ON 또는 ON→OFF의 변화가 있을 때만 1사이클 ON됩니다. 1사이클 후에는 OFF가 됩니다.
- OR/OR NOT 명령은 매회 리프레시를 지정(!)할 수 있습니다. 이 경우, 이 명령 실행 직전에 기본 입력 유니트(단, C200H 그룹 2 다점 입력 유니트, 원격 I/O 자국 장치의 기본 입력 유니트 제외)에 대해 IN 리프레시를 실행하고 그 내용을 직접 읽습니다.
- OR 명령은 기동 미분 또는 종료 미분과 매회 리프레시를 복합 지정(!@ 또는 !%)할 수 있습니다. 이 경우, 이 명령 실행 직전에 기본 입력 유니트에서 IN 리프레시로 직접 읽은 내용에 OFF→ON 또는 ON→OFF의 변화가 있을 때만 1사이클 ON됩니다. 1사이클 후에는 OFF가 됩니다.
- CJ2 CPU 유니트에서는 데이터 메모리(DM) 및 확장 데이터 메모리(EM)의 비트 주소가 가능하므로 OR/OR NOT 명령을 사용할 수 있습니다. 기타 기종은 OR/OR NOT 명령을 사용할 수 없으므로 OR TST 명령을 사용하십시오.

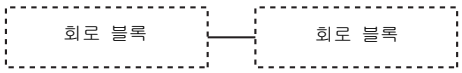
샘플 프로그램



명령	데이터
LD	0.00
AND	0.01
AND	0.02
OR	0.03
AND	0.04
LD	0.05
AND	0.06
OR NOT	0.07
AND LD	—
OUT	0.08

AND LD/OR LD

명령 이름	니모닉	동작 옵션	펑션 번호	기능
로드 논리곱	AND LD	−	−	회로 블록 간의 논리곱을 계산합니다.
로드 논리합	OR LD	−	−	회로 블록 간의 논리합을 계산합니다.

	AND LD	OR LD
심볼		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

조건 플래그의 동작

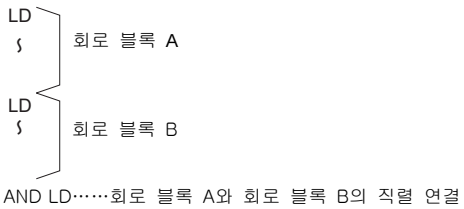
없음

기능

■ AND LD

이 명령 직전의 회로 블록과 회로 블록을 직렬로 연결합니다.

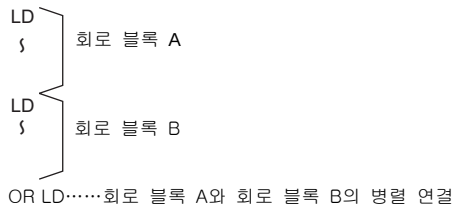
회로 블록은 LD/LD NOT 명령부터 시작하여 다음 LD/LD NOT 명령 직전까지의 회로를 가리킵니다.



■ OR LD

이 명령 직전의 회로 블록과 회로 블록을 병렬로 연결합니다.

회로 블록은 LD/LD NOT 명령부터 시작하여 다음 LD/LD NOT 명령 직전까지의 회로를 가리킵니다.



힌트

■ AND LD

· 3개 이상의 회로 블록을 직렬 연결하는 경우, 2개의 회로 블록을 이 명령으로 직렬 연결한 후, 다음 회로 블록을 이 명령으로 직렬 연결하는 형태로 순차 연결하는 것도 가능합니다. 또한 3개 이상의 회로 블록 뒤에 이 명령을 이어서 배치하여 일괄적으로 직렬 연결하는 것도 가능합니다.

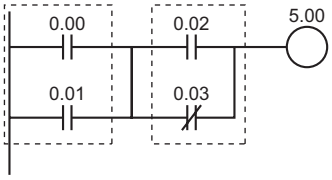
■ OR LD

· 3개 이상의 회로 블록을 병렬 연결하는 경우, 2개의 회로 블록을 이 명령으로 병렬 연결한 후, 다음 회로 블록을 이 명령으로 병렬 연결하는 형태로 순차 연결하는 것도 가능합니다. 또한 3개 이상의 회로 블록 뒤에 이 명령을 이어서 배치하여 일괄적으로 병렬 연결하는 것도 가능합니다.

사용상의 주의 사항

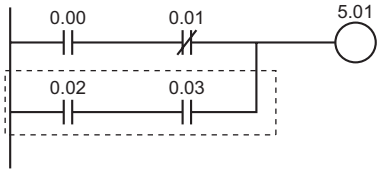
· AND LD 명령 또는 OR LD 명령을 사용하여 회로 블록을 연결하는 경우, AND LD/OR LD 명령을 합한 수와 LD/LD NOT 명령의 합계에서 1을 뺀 수가 같아야 합니다. 일치하지 않으면 회로 에러가 발생합니다.

■ AND LD



명령	데이터	
LD	0.00	
OR	0.01	
LD	0.02	←앞 블록에 직렬 연결하는 다음 블록의 최초 접점에서 사용합니다.
OR NOT	0.03	
AND LD	–	
OUT	5.00	

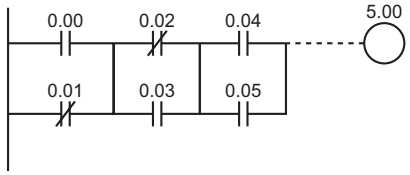
■ OR LD



명령	데이터	
LD	0.00	
AND NOT	0.01	
LD	0.02	←앞 블록에 직렬 연결하는 다음 블록의 최초 접점에서 사용합니다.
AND	0.03	
OR LD	–	
OUT	5.01	

샘플 프로그램

■ AND LD



코딩 ①

명령	데이터
LD	0.00
OR NOT	0.01
LD NOT	0.02
OR	0.03
AND LD	–
LD	0.04
OR	0.05
AND LD	–
:	:
OUT	5.00

코딩 ②

명령	데이터
LD	0.00
OR NOT	0.01
LD NOT	0.02
OR	0.03
LD	0.04
OR	0.05
:	:
AND LD	–
AND LD	–
:	:
OUT	5.00

· AND LD 명령은 몇 번이든 계속 사용할 수 있지만 ②의 방법으로 프로그래밍하는 경우 AND LD의 수는 그 앞에 있는 (LD 및 LD NOT의 수)에서 1을 뺀 수가 됩니다.

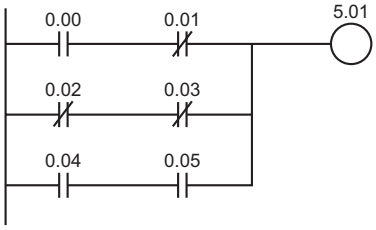
· ②의 경우, AND LD 앞에서는 LD, LD NOT의 개수를 합계 8개 이하로 하십시오.

· 9개 이상이 되면 ①의 방법으로 프로그래밍하십시오.

· 9개 이상일 때는 주변 툴을 사용한 프로그램 확인 시, 회로 에러가 발생합니다.

AND LD/OR LD

■ OR LD



코딩 ①

명령	데이터
LD	0.00
AND NOT	0.01
LD NOT	0.02
AND NOT	0.03
OR LD	–
LD	0.04
AND	0.05
OR LD	–
:	:
OUT	5.01

코딩 ②

명령	데이터
LD	0.00
AND NOT	0.01
LD NOT	0.02
AND NOT	0.03
LD	0.04
AND	0.05
:	:
OR LD	–
OR LD	–
:	:
OUT	5.01

- OR LD 명령은 몇 번이든 계속 사용할 수 있지만 ②의 방법으로 프로그래밍하는 경우 OR LD의 수는 그 앞에 있는 (LD 및 LD NOT의 수)에서 1을 뺀 수가 됩니다.
- ②의 경우, OR LD 앞에서는 LD, LD NOT의 개수를 합계 8개 이하로 하십시오.
- 9개 이상이 되면 ①의 방법으로 프로그래밍하십시오.
- ②의 방법에서 9개 이상일 때는 주변 툴을 사용한 프로그램 확인 시, 회로 에러가 발생합니다.

NOT

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부정	NOT	–	520	입력 조건을 반전합니다.

심볼	NOT
	

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

조건 플래그의 동작

없음

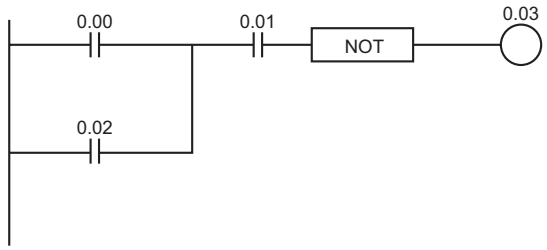
기능

입력 조건을 반전하여 다음 단계로 연결합니다.

사용상의 주의 사항

- 이 명령의 최종 단계에는 출력계 명령(OUT계 명령 및 다음 단계 연결형 명령 이외의 응용 명령)을 추가하십시오.
- 회로의 최종 단계에는 이 명령을 사용할 수 없습니다.

샘플 프로그램

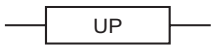


0.00, 0.01, 0.02의 논리 조건의 연산 결과를 반전하여 0.03에 출력합니다.

0.00	0.01	0.02	0.03
1	1	1	0
1	1	0	0
1	0	1	1
0	1	1	0
1	0	0	1
0	1	0	1
0	0	1	1
0	0	0	1

UP/DOWN

명령 이름	니모닉	동작 옵션	펑션 번호	기능
P.F. 기동 미분	UP	−	521	입력 신호가 기동될 때(OFF→ON) 1사이클 ON 하고 다음 단계에 연결합니다.
P. F. 종료 미분	DOWN	−	522	입력 신호가 종료될 때(ON→OFF) 1사이클 ON 하고 다음 단계 이후로 연결합니다.

심볼	UP	DOWN
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

조건 플래그의 동작

없음

기능

■ UP
다음 단계 연결형 기동 미분 명령입니다.

■ DOWN
다음 단계 연결형 종료 미분 명령입니다.

힌트

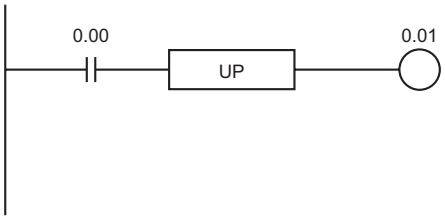
· 기동 미분 명령(DIFU)/종료 미분 명령(DIFD)에서도 논리 회로상에서 이전 단계까지의 입력 신호에 대한 미분을 취할 수 있지만 출력 명령이기 때문에 내부 보조 릴레이가 필요합니다. 이 명령에서는 직접 다음 단계로 연결할 수 있기 때문에 내부 보조 릴레이와 프로그램 스텝 수를 줄일 수 있습니다.

사용상의 주의 사항

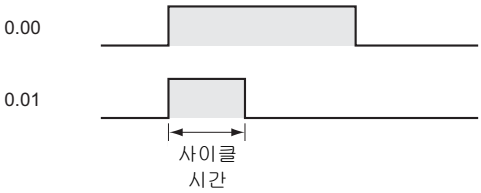
· 이 명령의 최종 단계에는 출력계 명령(OUT계 명령 및 다음 단계 연결형 명령 이외의 응용 명령)을 추가하십시오.
· 회로의 최종 단계에는 이 명령을 사용할 수 없습니다.
· IL-ILC 간, JMP-JME 명령 간, 서브루틴 명령 내에서 이 명령을 사용하는 경우 명령 실행 조건에 따라 동작이 안정되지 않는 경우가 있으므로 주의하십시오.
· 동일한 사이클 내에서 동일한 펑션 블록 인스턴스를 여러 번 실행한 경우, 이 명령의 동작이 일정치 않으므로 주의하십시오.
· 인스턴스는 EN이 OFF인 동안 실행되지 않습니다. 따라서 펑션 블록 정의 내에서 이 명령을 사용한 경우, 주의해야 합니다. 자세한 내용은 'CX-Programmer SYSMAC CS/CJ 시리즈 조작 설명서 펑션 블록편'에서 래더 언어 사용 시의 명령 제한 항목을 참조하십시오.
· 동일한 사이클 내에서 동일한 서브루틴을 여러 번 실행한 경우, 이 명령의 동작이 일정치 않으므로 주의하십시오.
· 서브루틴은 입력이 OFF인 동안 실행되지 않습니다. 따라서 서브루틴 영역 내에서 이 명령을 사용한 경우, 주의해야 합니다. 자세한 내용은 서브루틴 명령(SBS) 항목을 참조하십시오.

샘플 프로그램

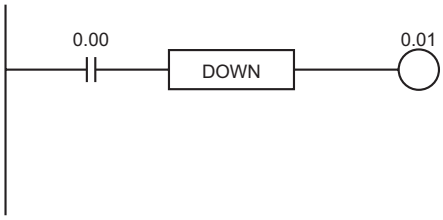
■ UP



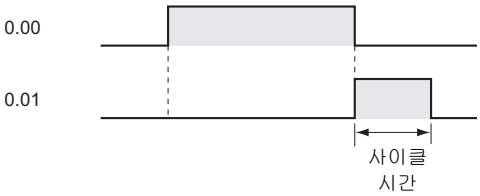
0.00이 OFF→ON이면 1사이클만 0.01이 ON으로 됩니다.



■ DOWN

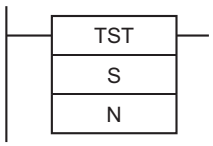
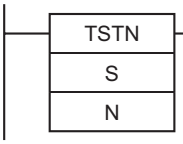


0.00이 ON→OFF이면 1사이클만 0.01이 ON으로 됩니다.



LD TST/LD TSTN

명령 이름	니모닉	동작 옵션	평션 번호	기능
LD 비트 테스트	LD TST	−	350	지정 비트가 1일 때 다음 단계에 LD 연결합니다.
LD 비트 테스트 부정	LD TSTN	−	351	지정 비트가 0일 때 다음 단계에 LD 연결합니다.

심볼	LD TST	LD TSTN
	 S: 테스트 데이터 채널 번호 N: 비트 위치 데이터	 S: 테스트 데이터 채널 번호 N: 비트 위치 데이터

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	테스트 데이터 채널 번호	WORD	1
N	비트 위치 데이터	UINT	1

N: 비트 위치 데이터

10진수 &0~15 또는 16진수 #0000~000F

채널을 지정할 때, 범위 밖의 값인 경우는 하위 4비트(16진수 #0~F)만 유효합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	−	○	−	○	−	−	−	−
N											○							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	변하지 않음*1
플래그	P_EQ	변하지 않음*1
부정 플래그	P_N	변하지 않음*1

*1 CS1/CJ1/CS1D(이중화 시스템용) CPU 유니트는 OFF

기능

■ LD TST

S의 N비트의 ON/OFF 내용을 입력 조건에 반영합니다. (LD 명령과 동일하게 취급하며, 이후에는 계속하여 다른 명령을 프로그래밍합니다.)

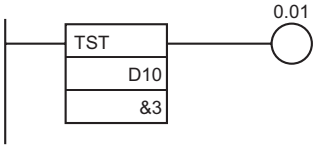
■ LD TSTN

S의 N비트의 ON/OFF 내용을 반전하여 입력 조건에 반영합니다. (LD 명령과 동일하게 취급하며, 이후에는 계속하여 다른 명령을 프로그래밍합니다.)

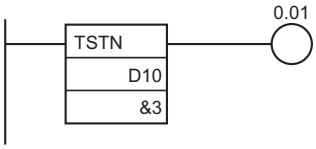
사용상의 주의 사항

- 이 명령의 최종 단계에는 출력계 명령(OUT계 명령 및 다음 단계 연결형 명령 이외의 응용 명령)을 추가하십시오.
- 회로의 최종 단계에는 이 명령을 사용할 수 없습니다.

샘플 프로그램



D10의 비트 3이 ON인지, OFF인지 확인하여 ON이면 0.01이 ON으로 됩니다.



D10의 비트 3이 ON인지, OFF인지 확인하여 OFF이면 0.01이 ON으로 됩니다.

AND TST/AND TSTN

명령 이름	니모닉	동작 옵션	펑션 번호	기능
AND 비트 테스트	AND TST	–	350	지정 비트가 1일 때 다음 단계에 AND 연결합니다.
AND LD 비트 테스트 부정	AND TSTN	–	351	지정 비트가 0일 때 다음 단계에 AND 연결합니다.

심볼	AND TST	AND TSTN
	AND TST S N S: 테스트 데이터 채널 번호 N: 비트 위치 데이터	AND TSTN S N S: 테스트 데이터 채널 번호 N: 비트 위치 데이터

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	테스트 데이터 채널 번호	WORD	1
N	비트 위치 데이터	UINT	1

N: 비트 위치 데이터

10진수 &0~15 또는 16진수 #0000~000F

채널을 지정할 때, 범위 밖의 값인 경우는 하위 4비트(16진수 #0~F)만 유효합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
N											○							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	변하지 않음*1
= 플래그	P_EQ	변하지 않음*1
부정 플래그	P_N	변하지 않음*1

*1 CS1/CJ1/CS1D(이중화 시스템용) CPU 유니트는 OFF

기능

■ AND TST

S의 N비트의 ON/OFF 내용과 입력 조건의 논리곱을 입력 조건에 반영합니다. (AND 명령과 동일하게 취급하며, 이후에는 계속하여 다른 명령을 프로그래밍합니다.)

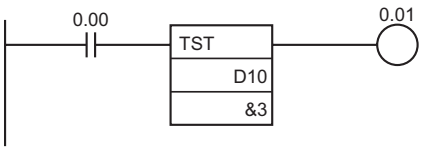
■ AND TSTN

S의 N비트의 ON/OFF 내용을 반전하여 입력 조건과의 논리곱을 입력 조건에 반영합니다. (AND 명령과 동일하게 취급하며, 이후에는 계속하여 다른 명령을 프로그래밍합니다.)

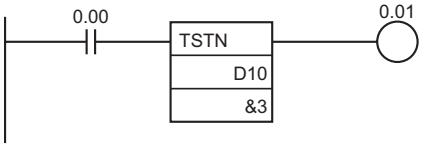
사용상의 주의 사항

- 이 명령의 최종 단계에는 출력계 명령(OUT계 명령 및 다음 단계 연결형 명령 이외의 응용 명령)을 추가하십시오.
- 회로의 최종 단계에는 이 명령을 사용할 수 없습니다.

샘플 프로그램



0.00이 ON이면 D10의 비트 3이 ON인지, OFF인지 확인합니다. 확인 결과, 비트 3이 ON이면 0.01이 ON으로 됩니다.



0.00이 ON이면 D10의 비트 3이 ON인지, OFF인지 확인합니다. 확인 결과, 비트 3이 OFF이면 0.01이 ON으로 됩니다.

OR TST/OR TSTN

명령 이름	니모닉	동작 옵션	펑션 번호	기능
OR 비트 테스트	OR TST	–	350	지정 비트가 1일 때 다음 단계에 OR 연결합니다.
OR 비트 테스트 부정	OR TSTN	–	351	지정 비트가 0일 때 다음 단계에 OR 연결합니다.

심볼	OR TST	OR TSTN
	TST S N S: 테스트 데이터 채널 번호 N: 비트 위치 데이터	TSTN S N S: 테스트 데이터 채널 번호 N: 비트 위치 데이터

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	테스트 데이터 채널 번호	WORD	1
N	비트 위치 데이터	UINT	1

N: 비트 위치 데이터

10진수 &0~15 또는 16진수 #0000~000F

채널을 지정할 때, 범위 밖의 값인 경우는 하위 4비트(16진수 #0~F)만 유효합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
N											○							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	변하지 않음*1
= 플래그	P_EQ	변하지 않음*1
부정 플래그	P_N	변하지 않음*1

*1 CS1/CJ1/CS1D(이중화 시스템용) CPU 유니트는 OFF

기능

■ OR TST

S의 N비트의 ON/OFF 내용과 입력 조건의 논리합을 입력 조건에 반영합니다. (OR 명령과 동일하게 취급하며, 이후에는 계속하여 다른 명령을 프로그래밍합니다.)

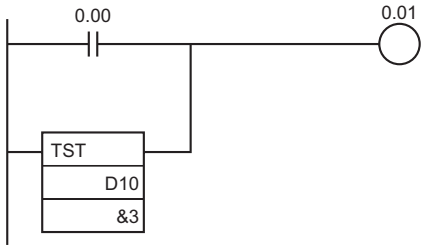
■ OR TSTN

S의 N비트의 ON/OFF 내용을 반전하여 입력 조건과의 논리합을 입력 조건에 반영합니다. (OR 명령과 동일하게 취급하며, 이후에는 계속하여 다른 명령을 프로그래밍합니다.)

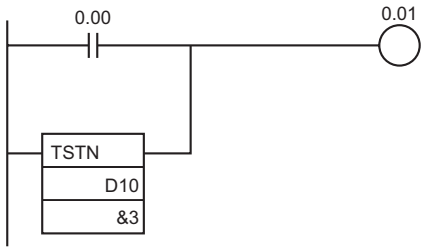
사용상의 주의 사항

- 이 명령의 최종 단계에는 출력계 명령(OUT계 명령 및 다음 단계 연결형 명령 이외의 응용 명령)을 추가하십시오.
- 회로의 최종 단계에는 이 명령을 사용할 수 없습니다.

샘플 프로그램



0.00과 D10의 비트 3의 논리합을 0.01에 출력합니다.



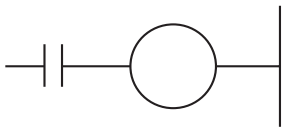
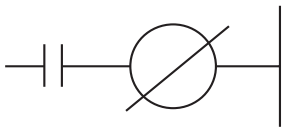
0.00과 D10의 비트 3의 부정 접점의 논리합을 0.01에 출력합니다.

시퀀스 출력 명령

니모닉	명령 이름	펑션 번호	페이지
OUT	출력	–	3-28
OUT NOT	부정 출력	–	
TR	일시 기억 릴레이	–	3-30
KEEP	유지	011	3-32
DIFU	기동 미분	013	3-36
DIFD	종료 미분	014	3-38
SET	세트	–	3-40
RSET	리셋	–	
SETA	다중 비트 세트	530	3-42
RSTA	다중 비트 리셋	531	
SETB	1비트 세트	532	3-44
RSTB	1비트 리셋	533	
OUTB	1비트 출력	534	3-46

OUT/OUT NOT

명령 이름	니모닉	동작 옵션	펑션 번호	기능
출력	OUT	!OUT	–	논리 연산 처리 결과를 릴레이 출력합니다.
부정 출력	OUT NOT	!OUT NOT	–	논리 연산 처리 결과를 반전하여 릴레이 출력합니다.

	OUT	OUT NOT
심볼		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
–	–	BOOL	–

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
OUT	○	○	○	○	–	–	○*1	○*1	–	–	–	–	–	○	–	–	–	○
OUT NOT																		

*1: CJ2 CPU 유니트만 사용 가능

조건 플래그의 동작

없음

기능

■ OUT

매회 리프레시를 지정하지 않은 경우 입력 조건(전원 흐름)의 내용을 I/O 메모리의 지정된 비트에 씁니다.
매회 리프레시를 지정한 경우에는 입력 조건(전원 흐름)의 내용을 I/O 메모리의 지정된 비트와 함께 기본 출력 유니트의 실제 출력 접점에도 씁니다.

■ OUT NOT

매회 리프레시를 지정하지 않은 경우 입력 조건(전원 흐름)의 내용을 반전하여 I/O 메모리의 지정된 비트에 씁니다.
매회 리프레시를 지정한 경우에는 입력 조건(전원 흐름)의 내용을 반전하여 I/O 메모리의 지정된 비트와 함께 기본 출력 유니트의 실제 출력 접점에도 씁니다.

힌트

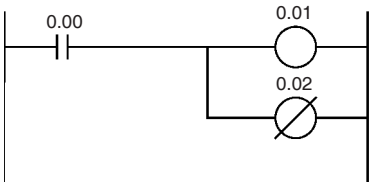
- 매회 리프레시를 지정(!OUT/!OUT NOT)할 수 있습니다. 이 경우, 이 명령 실행 직후에 직전의 입력 조건(전원 흐름)의 내용을 I/O 메모리의 지정된 비트에 쓰는 동시에 기본 출력 유니트(단, C200H 그룹 2 다점 출력 유니트, 원격 I/O 자국 장치의 기본 출력 유니트 제외)에 대해 OUT 리프레시를 실행합니다.
- SET/RESET 명령과 OUT 명령의 차이
  OUT 명령은 입력 조건이 ON일 때는 지정된 접점이 ON이 되고, 입력 조건이 OFF일 때는 지정된 접점이 OFF가 됩니다. SET/RSET 명령은 입력 조건이 ON일 때만 지정된 접점을 ON/OFF합니다. 입력 조건이 OFF일 때는 지정된 접점의 ON/OFF 상태가 변하지 않습니다.

사용상의 주의 사항

- CJ2 CPU 유니트에서는 OUT 명령을 데이터 메모리(DM) 및 확장 데이터 메모리(EM)에 대해 사용할 수 있습니다. 또한 CJ2 CPU 유니트 이외의 CPU인 경우, OUT 명령을 데이터 메모리(DM) 및 확장 데이터 메모리(EM)에 대해 사용할 수 없으므로 OUTB 명령을 사용하십시오.
  OUT 명령은 데이터 메모리(DM) 및 확장 데이터 메모리(EM)에 대해 사용할 수 없습니다. 이러한 영역에는 OUTB 명령을 사용하십시오.
- 이 명령에 인덱스 레지스터(간접 지정)를 지정하면 입력 조건이 OFF일 때도 동작하므로 주의하십시오.



샘플 프로그램



명령	데이터
LD	0.00
OUT	0.01
OUT NOT	0.02

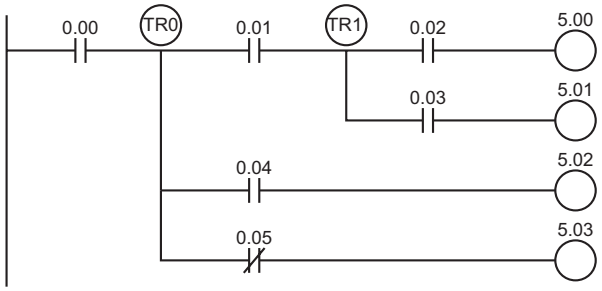
TR

명령 이름	니모닉	동작 옵션	평션 번호	기능
일시 기억 릴레이	TR	–	–	TR(일시 기억 릴레이)은 니모닉 프로그램에서 회로 중간의 ON/OFF 상태를 일시 기억할 때 사용합니다.

기능

TR(일시 기억 릴레이)은 니모닉 프로그램에서 회로 중간의 ON/OFF 상태를 일시 기억할 때 사용합니다.

래더도에서는 주변 툴에서 자동으로 처리하므로 사용하지 않습니다. 아래 설명의 경우는 알기 쉽게 설명하기 위해 래더도를 사용합니다.



명령	데이터
LD	0.00
OUT	TR0
AND	0.01
OUT	TR1
AND	0.02
OUT	5.00
LD	TR1
AND	0.03
OUT	5.01
LD	TR0
AND	0.04
OUT	5.02
LD	TR0
AND NOT	5.00
OUT	5.03

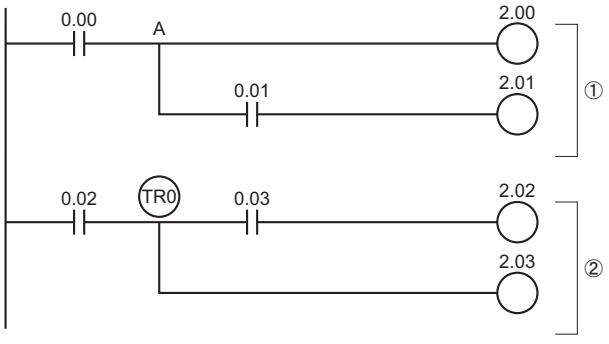
릴레이 번호	
일시 기억 릴레이	TR0~15

■ TR0~15 사용 방법

- TR0~15는 LD, OUT 명령 이외에서 사용할 수 없습니다.
- TR0~15의 릴레이 번호 사용 순서에는 제한이 없습니다.
- TR이 필요 없는 회로와 필요한 회로

①의 경우, A 점의 ON/OFF 상태는 출력 2.00과 같으므로 OUT2.00에 이어서 AND0.01, OUT2.01을 코딩할 수 있으며 TR은 필요 없습니다.

②의 경우, 분기점의 내용과 2.02의 출력 내용이 반드시 일치하지는 않으므로 TR에서 수신해야 합니다. ②의 회로를 ①로 재기입하면 프로그램 스텝 수가 줄어듭니다.

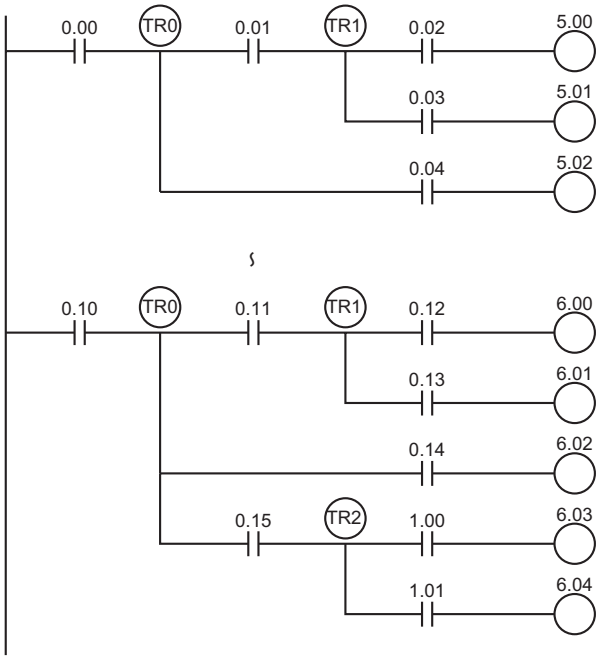


■ TR0~15의 개념

TR은 출력 분기가 많은 회로의 분기점에서 ON/OFF 상태를 기억(OUT TR0~15) 및 재현(LD TR0~15)하기 위해서만 사용되므로 일반적인 릴레이, 접점과는 달리 AND, OR 명령이나 NOT을 더한 명령에서 사용할 수 없습니다.

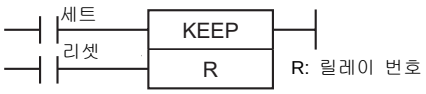
■ TR0~15 코일의 이중 사용

오른쪽 그림과 같이 출력 분기 회로가 많은 경우, 동일한 블록 안에서는 TR의 릴레이 번호를 중복해서 사용할 수 없지만 다른 블록에서는 사용할 수 있습니다.



KEEP

명령 이름	니모닉	동작 옵션	평션 번호	기능
유지	KEEP	!KEEP	011	래치 릴레이(자체 유지)의 동작을 수행합니다.

심볼	KEEP
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
R	릴레이 번호	BOOL	–

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
R	○	○	○	○	–	–	○*1	○*1	–	–	–	–	–	○	–	–	–	–

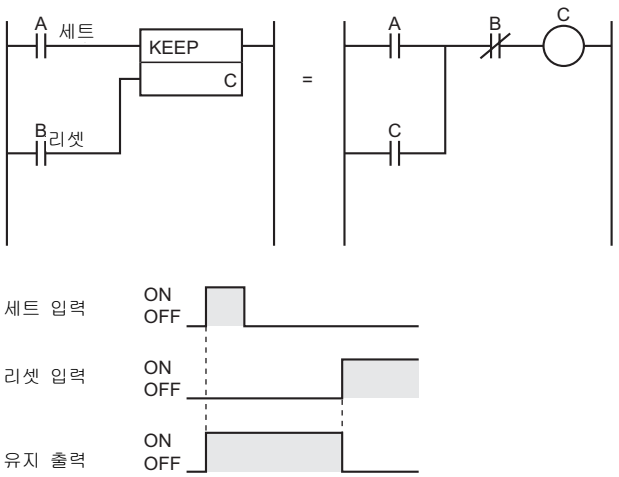
*1: CJ2 CPU 유니트만 사용 가능

조건 플래그의 동작

없음

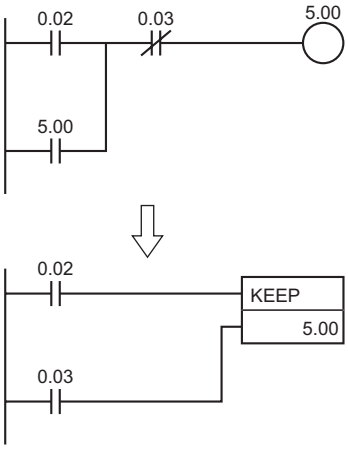
기능

세트 입력(입력 조건)이 ON이면 R로 지정한 릴레이의 ON 상태를 유지합니다. 리셋 입력이 ON이면 OFF 상태가 됩니다.

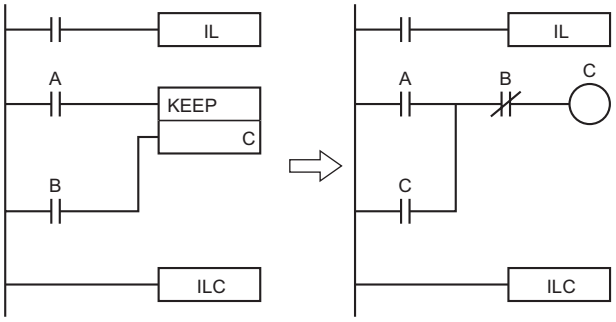


힌트

- KEEP 명령은 매회 리프레시형 명령(!KEEP)으로 사용할 수 있습니다.
  매회 리프레시형 명령(!KEEP)의 경우, R에 외부 I/O 할당된 출력 릴레이 영역을 지정합니다.
  R에 외부 출력을 지정하면 래치를 설정 또는 해제한 R을 명령 실행 시에 OUT 리프레시합니다(단, 그룹 2 다점 입출력 유니트, 고기능 I/O 유니트의 다점 입출력 유니트, SYSBUS 원격 I/O 자국 장치의 유니트는 제외).
- 자체 유지 회로의 KEEP 명령을 사용하면 다음과 같이 재기입할 수 있습니다.



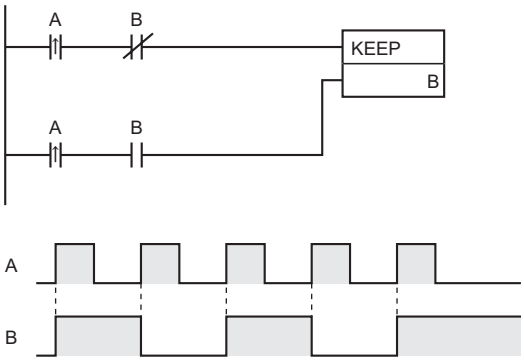
단, 위의 회로가 IL-ILC 사이에 있는 경우는 IL 조건이 OFF일 때 출력 릴레이를 사용한 회로(위쪽 회로)는 출력 5.00이 OFF가 되고, KEEP 명령을 사용한 회로는 상태를 유지하므로 주의하십시오.



IL(인터록) 중, C는 인터록 전의 상태 유지

IL(인터록) 중, C는 OFF

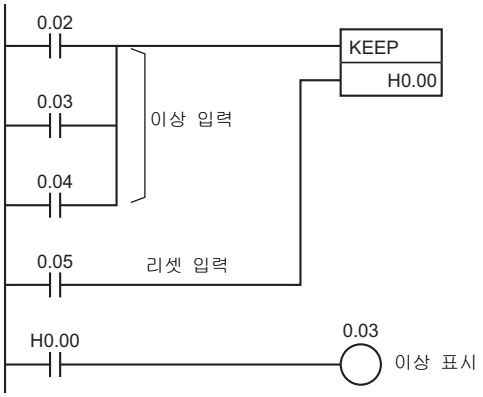
- KEEP 명령을 사용하여 플립플롭 회로를 작성할 수 있습니다.



KEEP

· 유지 릴레이를 KEEP 명령에서 사용하면 정전 시에도 직전의 상태를 기억할 수 있습니다.

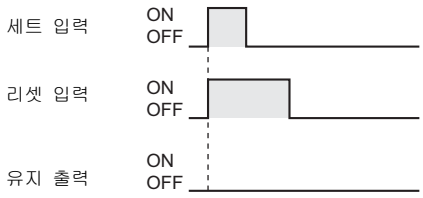
<정전 시 대책을 위한 이상 표시 회로 예>



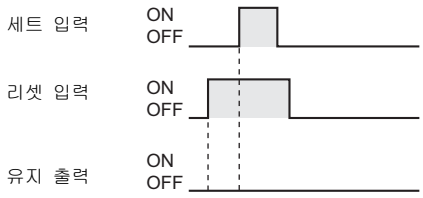
· IOM(I/O 메모리) 유지 비트가 ON이고, PLC 시스템 설정에서 IOM(I/O 메모리) 유지 비트를 유지로 설정한 경우(입력 릴레이는 정전 시에도 전원 OFF 직전의 상태를 유지하기 위해) 입출력 릴레이를 KEEP 명령에서 사용하면 유지 릴레이와 동일하게 정전 시에도 직전의 상태를 기억할 수 있습니다. 또한 PLC 시스템 설정을 설정한 경우, 다음 번에 전원을 켰을 때부터 유효해지므로 주의하십시오.

사용상의 주의 사항

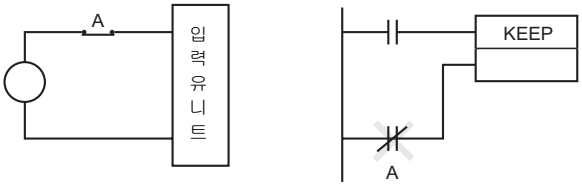
· 세트 입력(입력 조건)과 리셋 입력이 동시에 ON인 경우 리셋 입력이 우선합니다.



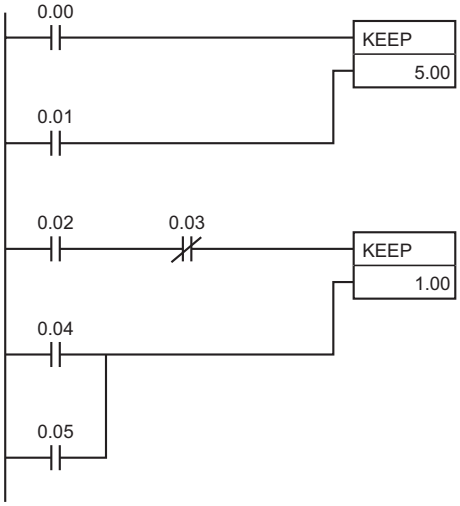
· 리셋 입력이 ON이면 세트 입력(입력 조건)은 수신되지 않습니다.



· KEEP 명령의 리셋 입력을 b 접점의 외부 기기에서 직접 전송하지 마십시오. AC 전원 OFF나 순간 정전일 때 PLC 본체의 내부 전원이 즉시 OFF되지 않고 입력 유니트의 입력이 먼저 OFF된 후, 그 결과 리셋 입력이 ON이 되어 리셋됩니다.



샘플 프로그램



0.00이 ON이면 5.00을 ON 상태로 유지합니다.
0.01이 ON이면 5.00을 OFF로 합니다.

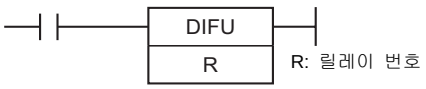
0.02가 ON이고 0.03이 OFF이면 1.00을 ON 상태로 유지합니다.
0.04 또는 0.05가 ON이면 1.00을 OFF로 합니다.

명령	데이터
LD	0.00
LD	0.01
KEEP(011)	5.00
LD	0.02
AND NOT	0.03
LD	0.04
OR	0.05
KEEP(011)	1.00

주: KEEP 명령은 래더도와 니모닉에서 입력 순서가 다릅니다.
래더도: 세트 입력→KEEP 명령→리셋 입력
니모닉: 세트 입력→리셋 입력→KEEP 명령

DIFU

명령 이름	니모닉	동작 옵션	평션 번호	기능
기동 미분	DIFU	!DIFU	013	입력 신호가 기동될 때(OFF→ON) 지정한 접점을 1사이클 ON합니다.

심볼	DIFU
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
R	릴레이 번호	BOOL	–

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
R	○	○	○	○	–	–	○*1	○*1	–	–	–	–	–	○	–	–	–	–

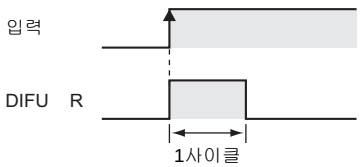
*1: CJ2 CPU 유니트만 사용 가능

조건 플래그의 동작

없음

기능

입력 신호가 기동될 때(OFF→ON) R로 지정한 접점을 1사이클 ON하고 1사이클 후 이 명령을 실행할 때 OFF로 합니다.



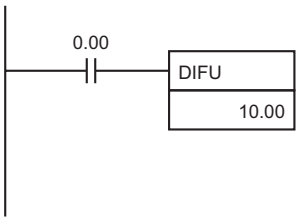
힌트

- 내부 보조 릴레이 등의 릴레이를 사용하지 않고 기동 미분을 다음 단계에 직접 연결하려는 경우, UP(기동 미분) 명령을 사용합니다.
- DIFU 명령은 매회 리프레시형 명령(!DIFU)으로 사용할 수 있습니다.
  매회 리프레시형 명령(!DIFU)의 경우, R에 외부 I/O 할당된 출력 릴레이 영역을 지정합니다(단, 그룹 2 다점 입출력 유니트, 고기능 I/O 유니트의 다점 입출력 유니트, SYSBUS 원격 I/O 자국 장치의 유니트는 제외).
  R에 외부 출력을 지정한 경우, ON으로 한 R을 해당 시점에서(명령 실행 시에) OUT 리프레시합니다.
  (ON으로 된 R은 평상시와 마찬가지로 1사이클 동안 ON이 됩니다.)

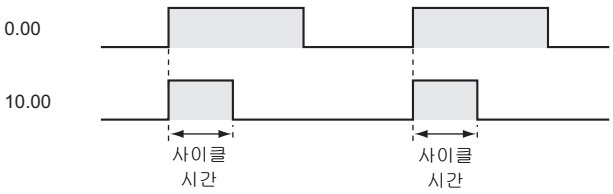
사용상의 주의 사항

- IL-ILC 명령 간, JMP-JME 명령 간 또는 서브루틴 명령 내에서 DIFU 명령을 사용하는 경우, 입력 조건에 따라 동작이 안정되지 않는 경우가 있으므로 주의하십시오.
- 인스턴스는 EN이 OFF인 동안 실행되지 않습니다. 따라서 펑션 블록 정의 내에서 이 명령을 사용한 경우, 주의해야 합니다. 자세한 내용은 'CX-Programmer SYSMAC CS/CJ 시리즈 조작 설명서 펑션 블록편'에서 래더 언어 사용 시의 명령 제한 항목을 참조하십시오.
- 동일한 사이클 내에서 동일한 펑션 블록 인스턴스를 여러 번 실행한 경우, 이 명령의 동작이 일정치 않으므로 주의하십시오.
- 서브루틴은 입력이 OFF인 동안 실행되지 않습니다. 따라서 서브루틴 영역 내에서 이 명령을 사용한 경우, 주의해야 합니다. 자세한 내용은 서브루틴 명령(SBS) 항목을 참조하십시오.
- 동일한 사이클 내에서 동일한 서브루틴을 여러 번 실행한 경우, 이 명령의 동작이 일정치 않으므로 주의하십시오.

샘플 프로그램



0.00이 OFF→ON으로 기동되면 10.00이 1사이클 동안만 ON으로 됩니다.



DIFD

명령 이름	니모닉	동작 옵션	펑션 번호	기능
종료 미분	DIFD	!DIFD	014	입력 신호가 종료될 때(ON→OFF) 지정한 접점을 1사이클 ON합니다.

심볼	DIFD
	

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
R	릴레이 번호	BOOL	–

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
R	○	○	○	○	–	–	○*1	○*1	–	–	–	–	–	○	–	–	–	–

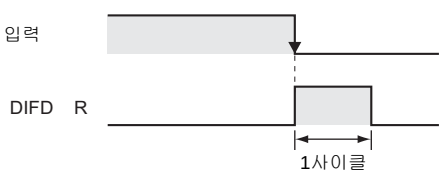
*1: CJ2 CPU 유니트만 사용 가능

조건 플래그의 동작

없음

기능

입력 신호가 종료될 때(ON→OFF) R로 지정한 접점을 1사이클 ON하고 1사이클 후 이 명령을 실행할 때 OFF로 합니다.



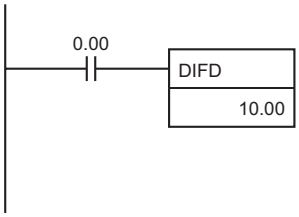
힌트

- 내부 보조 릴레이 등의 릴레이를 사용하지 않고 기동 미분을 다음 단계에 직접 연결하려는 경우는 DOWN(종료 미분) 명령을 사용합니다.
- IL-ILC 명령 간, JMP-JME 명령 간 또는 서브루틴 명령 내에서 DIFD 명령을 사용하는 경우, 입력 조건에 따라 동작이 안정되지 않는 경우가 있으므로 주의하십시오.
- DIFD 명령은 매회 리프레시형 명령(!DIFD)으로 사용할 수 있습니다.
  매회 리프레시형 명령(!DIFD)의 경우, R에 외부 I/O 할당된 출력 릴레이 영역을 지정합니다(단, 그룹 2 다점 입출력 유니트, 고기능 I/O 유니트의 다점 입출력 유니트, SYSBUS 원격 I/O 자국 장치의 유니트는 제외).
  R에 외부 출력을 지정한 경우, ON으로 한 R을 해당 시점에서(명령 실행 시에) OUT 리프레시합니다.
  (ON으로 된 R은 평상시와 마찬가지로 1사이클 동안 ON이 됩니다.)

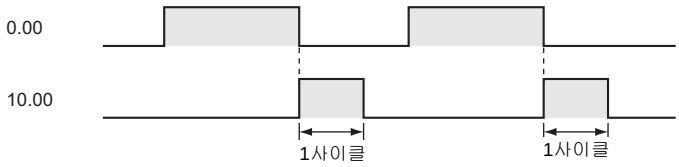
사용상의 주의 사항

- 동일한 사이클 내에서 동일한 펑션 블록 인스턴스를 여러 번 실행한 경우, 이 명령의 동작이 일정치 않으므로 주의하십시오.
- 인스턴스는 EN이 OFF인 동안 실행되지 않습니다. 따라서 펑션 블록 정의 내에서 이 명령을 사용한 경우, 주의해야 합니다. 자세한 내용은 'CX-Programmer SYSMAC CS/CJ 시리즈 조작 설명서 펑션 블록편'에서 래더 언어 사용 시의 명령 제한 항목을 참조하십시오.
- 동일한 사이클 내에서 동일한 서브루틴을 여러 번 실행한 경우, 이 명령의 동작이 일정치 않으므로 주의하십시오.
- 서브루틴은 입력이 OFF인 동안 실행되지 않습니다. 따라서 서브루틴 영역 내에서 이 명령을 사용한 경우, 주의해야 합니다. 자세한 내용은 서브루틴 명령(SBS) 항목을 참조하십시오.

샘플 프로그램



0.00이 ON→OFF로 기동되면 10.00이 1사이클 동안만 ON으로 됩니다.



SET/RSET

명령 이름	니모닉	동작 옵션	펑션 번호	기능
세트	SET	@SET, %SET, !SET, !@SET, !%SET	–	입력 조건이 ON일 때 지정된 접점을 ON으로 설정합니다. 이후, 입력 조건의 ON/OFF에 관계 없이 지정된 접점은 ON 상태를 그대로 유지합니다.
리셋	RSET	@RSET, %RSET, !RSET, !@RSET, !%RSET	–	입력 조건이 ON일 때 지정된 접점을 OFF로 리셋합니다. 이후, 입력 조건의 ON/OFF에 관계 없이 지정된 접점은 OFF 상태를 그대로 유지합니다.

심볼	SET	RSET
	SET R R: 릴레이 번호	RSET R R: 릴레이 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
R	릴레이 번호	BOOL	–

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
R	○	○	○	○	–	–	○*1	○*1	–	–	–	–	–	○	–	–	–	–

*1: CJ2 CPU 유니트만 사용 가능

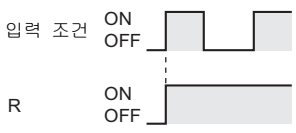
조건 플래그의 동작

없음

기능

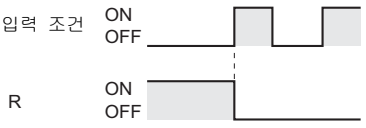
■ SET

입력 조건이 ON일 때 R로 지정한 접점을 ON으로 설정합니다. 입력 조건이 OFF이든, ON이든 상관 없이 지정된 접점 R은 ON 상태를 그대로 유지합니다. OFF로 하려면 RSET 명령을 사용합니다.



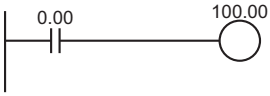
■ RSET

입력 조건이 ON일 때 R로 지정한 접점을 OFF합니다. 입력 조건이 OFF이든, ON이든 상관 없이 지정된 접점 R은 OFF 상태를 그대로 유지합니다. ON으로 하려면 SET 명령을 사용합니다.

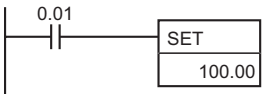


힌트

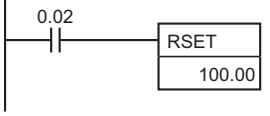
- SET/RESET 명령과 OUT 명령의 차이
  SET/RSET 명령은 입력 조건이 ON된 경우에만 지정된 접점을 ON/OFF합니다. 입력 조건이 OFF일 때는 지정된 접점의 ON/OFF 상태가 변하지 않습니다. OUT 명령은 입력 조건이 ON일 때는 지정된 접점이 ON이 되고, 입력 조건이 OFF일 때는 지정된 접점이 OFF가 됩니다.



100.00은 0.00의 ON/OFF에 의해 동작합니다.



0.01이 ON되면 100.00이 ON됩니다.



0.02가 ON될 때까지 100.00은 ON 상태를 유지합니다.

- KEEP 명령에서는 세트 입력과 리셋 입력을 동일한 위치에 기재해야 하지만 이 SET/RSET 명령은 따로 기재할 수 있습니다. 또한 같은 주소의 출력 릴레이에 대해 복수의 SET/RSET 명령을 사용할 수 있습니다.
- SET/RSET 명령은 매회 리프레시형 명령(!SET/!RSET)으로 사용할 수 있습니다.
  매회 리프레시형 명령(!SET/!RSET)의 경우, R에 외부 I/O 할당된 출력 릴레이 영역을 지정합니다(단, 그룹 2 다점 입출력 유니트, 고기능 I/O 유니트의 다점 입출력 유니트, SYSBUS 원격 I/O 자국 장치의 유니트는 제외).
  !SET(또는 !RSET 명령)로 R에 외부 출력을 지정한 경우, ON(또는 OFF)으로 한 R를 해당 시점에서(명령 실행 시) OUT 리프레시합니다. ON(또는 OFF)된 R은 평상시와 마찬가지로 RSET 명령(또는 SET명령)이 실행될 때까지 ON(또는 OFF)을 그대로 유지합니다.

사용상의 주의 사항

- CJ2 CPU 유니트에서는 데이터 메모리(DM) 및 확장 데이터 메모리(EM)의 비트 주소가 가능하므로 SET/RSET 명령을 사용할 수 있습니다. 기타 기종은 SET/RSET 명령을 사용할 수 없으므로 SETB/RSTB 명령을 사용하십시오.
- 이 SET/RSET 명령에서 타이머, 카운터의 세트/리셋은 실행할 수 없습니다.
- SET/RSET 명령을 IL-ILC/JMP-JME 명령 내에서 사용하는 경우, 인터록 조건/점프 조건이 OFF이면 지정된 출력 접점의 상태가 변하지 않습니다.

SETA/RSTA

명령 이름	니모닉	동작 옵션	평션 번호	기능
다중 비트 세트	SETA	@SETA	530	연속으로 지정된 비트 수 분량의 비트를 ON합니다.
다중 비트 리셋	RSTA	@RSTA	531	연속으로 지정 비트 수 분량의 비트를 OFF합니다.

심볼	SETA	RSTA
	SETA D N1 N2 D: 세트 하위 CH 번호 N1: 세트 시작 비트 위치 N2: 비트 수	RSTA D N1 N2 D: 리셋 하위 CH 번호 N1: 리셋 시작 비트 위치 N2: 비트 수

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
D	세트 하위 CH 번호	UINT	가변
N1	세트 시작 비트 위치	UINT	1
N2	비트 수	UINT	1

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
D	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
N1											○	○						
N2																		

조건 플래그의 동작

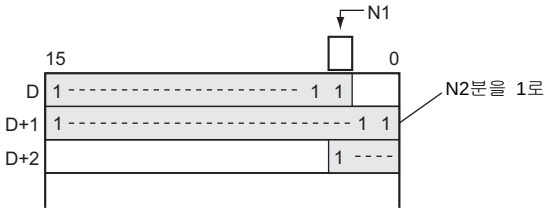
이름	라벨	내용
에러 플래그	P_ER	• N1의 데이터가 10진수 &0~15 또는 16진수 #0000~000F의 범위에 없을 때 ON • 그 이외일 때 OFF

기능

■ SETA

D로 지정한 하위 CH 번호의 N1로 지정한 시작 비트 위치(10진수 또는 16진수)에서 상위 비트 쪽에 연속으로 지정된 비트 수(N2) 분량을 ON합니다. 지정 범위 이외의 비트 데이터는 변경되지 않습니다. 또한 비트 수를 0으로 지정한 경우, 비트의 데이터는 변경되지 않습니다.

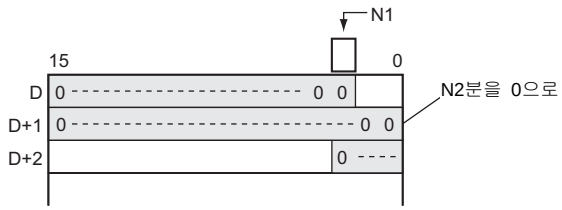
이 명령에서 ON으로 한 비트를 OFF로 할 때는 다중 비트 리셋(RSTA) 명령으로 제한하지 않고 일반 명령을 사용할 수 있습니다.



■ RSTA

D로 지정한 하위 CH 번호의 N1로 지정한 시작 비트 위치에서 상위 비트 쪽에 연속으로 지정된 비트 수(N2) 분량을 OFF합니다. 지정 범위 이외의 비트 데이터는 변경되지 않습니다. 또한 비트 수를 0으로 지정한 경우, 비트의 데이터는 변경되지 않습니다.

이 명령에서 OFF로 한 비트를 ON으로 할 때는 다중 비트 세트(SETA) 명령으로 제한하지 않고 일반 명령을 사용할 수 있습니다.



힌트

■ SETA

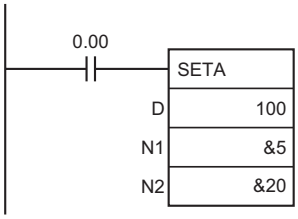
· SETA(다중 비트 세트) 명령에서는 데이터 메모리, 확장 데이터 메모리 등과 같이 채널(워드) 단위로 취급하는 영역 종류에 대해서도 지정 범위의 비트 영역을 일괄적으로 ON으로 설정할 수 있습니다.

■ RSTA

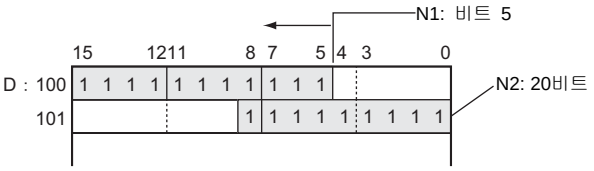
· RSTA(다중 비트 리셋) 명령에서는 데이터 메모리, 확장 데이터 메모리 등과 같이 채널(워드) 단위로 취급하는 영역 종류에 대해서도 지정 범위의 비트 영역을 일괄적으로 OFF로 설정할 수 있습니다.

샘플 프로그램

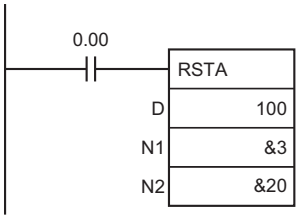
■ SETA



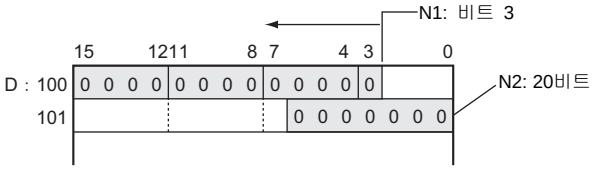
0.00이 ON이면 100CH의 비트 5에서 상위 측으로 20(16진수 #14)비트를 1(ON)로 합니다.



■ RSTA



0.00이 ON이면 100CH의 비트 3에서 상위 측으로 20(16진수 #14)비트를 0(OFF)으로 합니다.



SETB/RSTB

명령 이름	니모닉	동작 옵션	펑션 번호	기능
1비트 세트	SETB	@SETB, !SETB, !@SETB	532	입력 조건이 ON일 때 지정된 CH의 지정된 접점을 ON으로 설정합니다.
1비트 리셋	RSTB	@RSTB, !RSTB, !@RSTB	533	입력 조건이 ON일 때 지정된 CH의 지정된 접점을 OFF하고 리셋합니다.

심볼	SETB	RSTB
	SETB D N D: CH 주소 N: 비트 위치	RSTB D N D: CH 주소 N: 비트 위치

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
D	CH 주소	UINT	1
N	비트 위치	UINT	1

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
D	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
N											○							

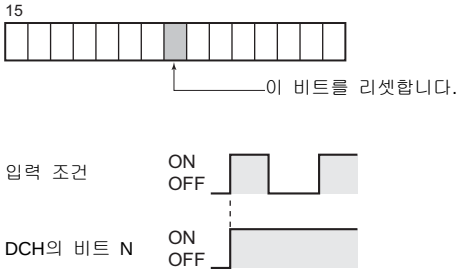
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· N이 10진수 &0~15 또는 16진수 #0000~000F 이외일 때 ON · 그 이외일 때 OFF

기능

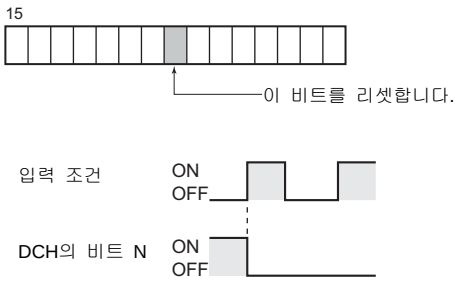
■ SETB

입력 조건이 ON일 때 D로 지정한 CH의 비트 주소 N을 ON으로 설정합니다. 입력 조건이 OFF가 되어도 D로 지정한 CH의 비트 주소 N은 변경되지 않습니다.



■ RSTB

입력 조건이 ON일 때 D로 지정한 CH의 비트 주소 N을 OFF합니다. 입력 조건이 OFF이든, ON이든 상관 없이 지정 접점 D로 지정된 CH의 비트 주소 N은 OFF 상태를 그대로 유지합니다. ON으로 하려면 SET 명령을 사용합니다.



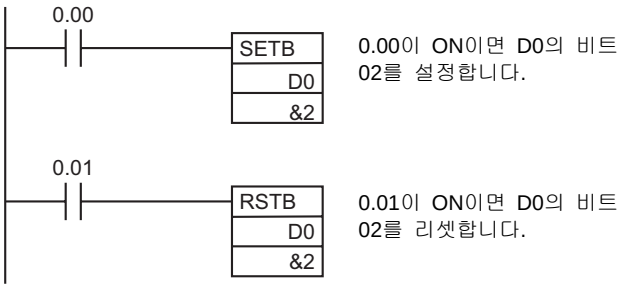
힌트

- SETB/RSTB 명령과 SET/RST 명령의 차이
  CIO, 내부 보조 릴레이(W), 유지 릴레이(H), 특수 보조 릴레이(A)에 대해서는 동일한 기능입니다.
  SETB/RSTB 명령은 SET/RST 명령에서 지정할 수 없는 데이터 메모리(D), 확장 데이터 메모리(E)의 지정된 비트에 대한 세트/리셋을 수행할 수 있습니다.
- SETB/RSTB 명령과 OUTB 명령의 차이
  SETB/RSTB 명령은 입력 조건이 ON된 경우에만 지정된 접점을 ON/OFF합니다.
  입력 조건이 OFF일 때는 지정된 접점의 ON/OFF 상태가 변하지 않습니다. OUTB 명령은 입력 조건이 ON일 때는 지정된 접점이 ON이 되고, 입력 조건이 OFF일 때는 지정된 접점이 OFF가 됩니다.
- KEEP 명령에서는 세트 입력과 리셋 입력을 동일한 위치에 기재해야 하지만 이 SETB/RSTB 명령은 따로 기재할 수 있습니다. 또한 같은 주소의 출력 릴레이에 대해 복수의 SETB/RSTB 명령을 사용할 수 있습니다.

사용상의 주의 사항

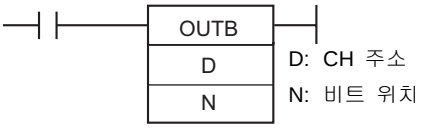
- SETB/RSTB 명령은 반드시 세트로 사용할 필요는 없습니다. (예: SETB 명령으로 세트한 비트를 리셋하기 위해 반드시 RSTB 명령을 사용할 필요는 없습니다.)
- 이 SETB/RSTB 명령에서 타이머, 카운터의 세트/리셋은 실행할 수 없습니다.
- SETB/RSTB 명령을 IL-ILC/JMP-JME 명령 내에서 사용하는 경우, 인터록 조건/점프 조건이 OFF이면 지정된 출력 접점의 상태가 변하지 않습니다.
- 매회 리프레시형 명령(!SETB/!RSTB)의 경우, R에 외부 I/O 할당된 출력 릴레이 영역을 지정합니다(단, 그룹 2 다점 입출력 유니트, 고기능 I/O 유니트의 다점 입출력 유니트, SYSBUS 원격 I/O 자국 장치의 유니트는 제외).
- !SETB(또는 !RSTB 명령)로 D CH의 비트 주소 N에 외부 출력을 지정한 경우, ON(또는 OFF)으로 한 D CH의 비트 주소 N을 해당 시점에서(명령 실행 시) OUT 리프레시합니다. ON(또는 OFF)된 D CH의 비트 주소 N은 평상시와 마찬가지로 RSTB 명령(또는 SETB명령)이 실행될 때까지 ON(또는 OFF)을 그대로 유지합니다.

샘플 프로그램



OUTB

명령 이름	니모닉	동작 옵션	평션 번호	기능
1비트 출력	OUTB	!OUTB	534	논리 연산 처리 결과(입력 조건)를 지정된 CH의 지정된 비트에 출력합니다.

심볼	OUTB
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
D	CH 주소	UINT	1
N	비트 위치	UINT	1

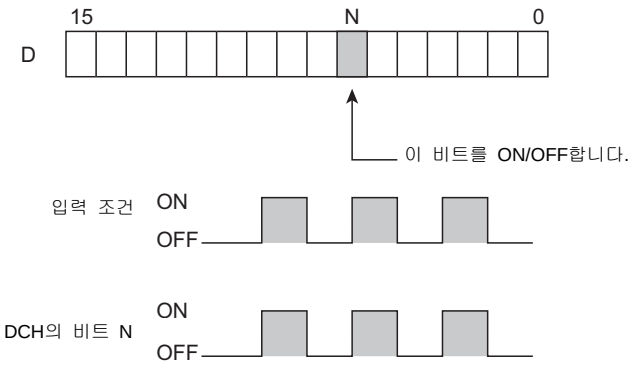
■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
D	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
N											○							

기능

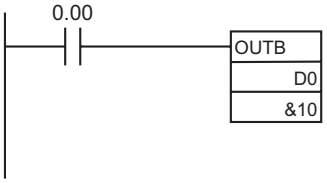
입력 조건이 ON일 때 D로 지정한 CH의 비트 주소 N을 ON합니다. 입력 조건이 OFF일 때 D로 지정한 CH의 비트 주소 N을 OFF합니다.

매회 리프레시를 지정하지 않은 경우 입력 조건(전원 흐름)의 내용을 I/O 메모리의 지정된 비트에 씁니다.
매회 리프레시를 지정한 경우에는 입력 조건(전원 흐름)의 내용을 I/O 메모리의 지정된 비트와 함께 기본 출력 유니트의 실제 출력 접점에도 씁니다.



힌트

· OUTB 명령과 SETB/RSTB의 차이
OUTB 명령은 입력 조건이 ON일 때는 지정된 접점이 ON이 되고, 입력 조건이 OFF일 때는 지정된 접점이 OFF가 됩니다.
SETB/RSTB 명령은 입력 조건이 ON된 경우에만 지정된 접점을 ON/OFF합니다. 입력 조건이 OFF일 때는 지정된 접점의 ON/OFF 상태가 변하지 않습니다.



0.00이 OFF이면 D0의 비트 10을 OFF합니다.

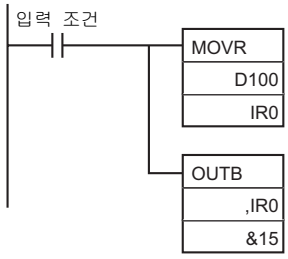
· OUTB 명령은 매회 리프레시형 명령(!OUTB)으로 사용할 수 있습니다. 이 경우, 이 명령 실행 직후에 직전의 입력 조건(전원 흐름)의 내용을 I/O 메모리의 지정된 비트에 쓰는 동시에, 기본 출력 유니트(단, C200H 그룹 2 다점 출력 유니트, 원격 I/O 자국 장치의 기본 출력 유니트 제외)에 대해 OUT 리프레시를 실행합니다.

사용상의 주의 사항

· 이 명령이 IL 명령~ILC 명령 사이의 프로그램 내에 기재되고 인터록 상태(IL 중)가 된 경우, OUT 명령과 동일하게 이 명령으로 지정된 비트는 OFF로 됩니다.

· 비트 위치 N을 채널 지정하는 경우 비트 00~03만 사용합니다.
예를 들어, N으로 지정한 채널의 내용이 16진수 #FFFA인 경우, 비트 10에 대해 ON/OFF합니다.

· 이 명령에 인덱스 레지스터(간접 지정)를 지정하면 입력 조건이 OFF일 때도 동작하므로 주의하십시오.



입력 조건이 OFF일 경우, MOVR 명령은 실행되지 않지만 인덱스 레지스터에 저장된 주소에 대해 OUTB 명령은 실행됩니다.

시퀀스 제어 명령

니모닉	명령 이름	펑션 번호	페이지
END	종료	001	3-52
NOP	기능 없음	000	3-53
IL	인터록	002	3-54
ILC	인터록 클리어	003	
MILH	멀티 인터록(미분 플래그 유지형)	517	3-57
MILR	멀티 인터록(미분 플래그 비유지형)	518	
MILC	멀티 인터록 클리어	519	
JMP	점프	004	3-66
JME	점프 종료	005	
CJP	조건부 점프	510	3-69
CJPN	조건부 부정 점프	511	
JMP0	복수 점프	515	3-72
JME0	복수 점프 종료	516	
FOR	반복 시작	512	3-74
NEXT	반복 종료	513	
BREAK	루프 중단	514	3-77

인터록 명령

■ 인터록 명령의 종류

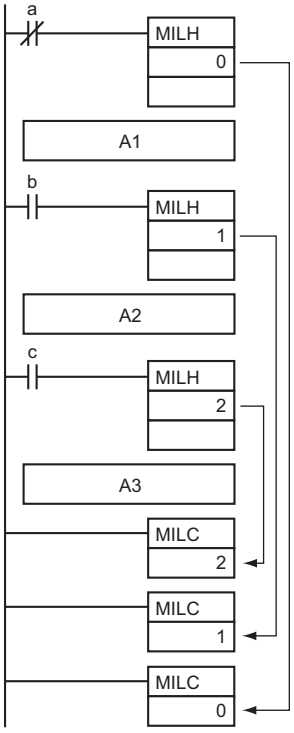
출력을 인터록하는 명령에는 다음과 같은 명령이 있습니다.

- 인터록(IL－ILC) 명령
- 멀티 인터록(MILH－MILC 또는 MILR－MILC) 명령(CS/CJ 시리즈 유니트 Ver.2.0 이상만 해당)
  멀티 인터록(미분 플래그 유지형) (MILH－MILC) 명령
  멀티 인터록(미분 플래그 비유지형) (MILR－MILC) 명령

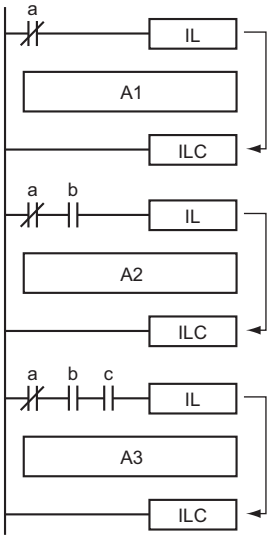
■ 인터록(IL－ILC) 명령과 멀티 인터록(MILH－MILC 또는 MILR－MILC) 명령의 차이점

인터록(IL－ILC) 명령은 중첩할 수 없습니다. 그에 비해 멀티 인터록(MILH－MILC 또는 MILR－MILC) 명령은 중첩 가능합니다. 따라서 다음과 같이 IL-ILC 명령에 비해 간단히 명령을 기술할 수 있습니다.

MILH-MILC 명령



IL-ILC 명령에서 구현



■ 멀티 인터록(미분 플래그 유지형) (MILH) 명령과 멀티 인터록(미분 플래그 비유지형) (MILR) 명령의 차이점

MILH 명령과 MILR 명령은 MILH－MILC 명령 또는 MILR－MILC 명령 사이에 미분 명령(DIFU/DIFD/@/% 표시가 있는 명령)이 존재하는 경우 해당 미분 명령의 동작이 서로 다릅니다.

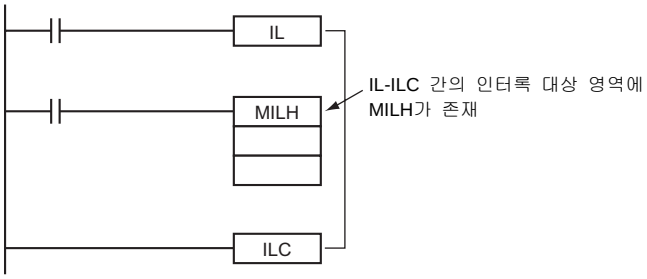
또한 MILH 명령은 미분 명령의 동작의 경우 IL 명령과 동일합니다.

자세한 내용은 MILH/MILR/MILC 명령 설명의 'MILH 명령과 MILR 명령의 차이점' 항목을 참조하십시오.

■ 사용 제한

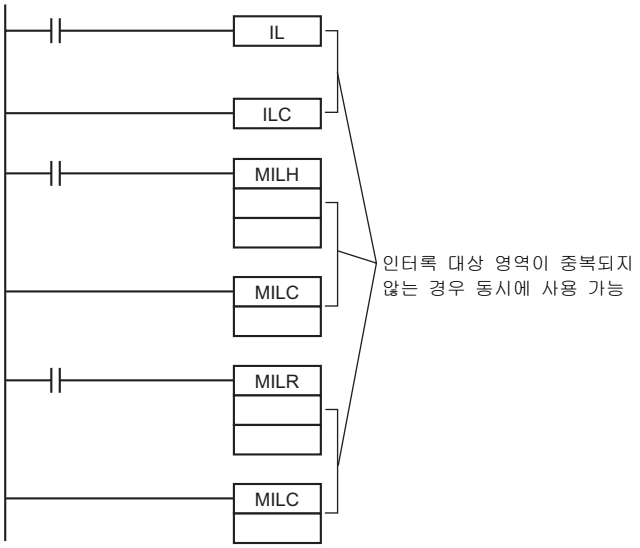
IL－ILC 명령, MILH－MILC 명령, MILR－MILC 명령을 혼합하여 사용하지 마십시오. 혼합하여 사용하는 경우 회로 동작을 보증할 수 없습니다. 혼합하여 사용한 경우의 동작에 대해서는 MILH/MILR/MILC 명령의 설명 항목을 참조하십시오.

예) IL－ILC 명령 사이에 MILH 명령 배치 불가



주: IL－ILC 명령, MILH－MILC 명령, MILR－MILC 명령의 각 인터록 대상 영역이 중복되지 않으면 동시에 사용할 수 있습니다.

예) IL－ILC, MILH－MILC, MILR－MILC를 중복하지 않고 배치하는 것은 가능



■ 인터록 명령과 점프 명령의 차이점

IL-ILC, MILH-MILC, MILR-MILC 명령은 다음과 같은 점에서 JMP-JME 명령과 다릅니다.

명령	명령의 실행	각 명령의 출력	OUT, OUTB, OUT NOT 명령을 지정한 릴레이	타이머계 명령(TTIM/TTIMX, MTIM/MTIMX 제외)의 상태
IL-ILC 명령 MILH-MILC 명령 MILR-MILC 명령	모든 명령을 실행 안 함(OUT, OUTB, OUT NOT, 타이머계 명령 이외)	모든 명령의 출력이 이전 상태를 유지(OUT, OUTB, OUT NOT, 타이머계 명령 이외)	OFF	리셋
JMP-JME 명령	모든 명령을 실행 안 함	모든 명령의 출력이 그 이전 상태를 유지	그 이전 상태 유지	기동 중인 타이머 (TIM/TIMX, TIMH/TIMHX, TMHH/TMHHX, TIMU/TIMUX, TMUH/TMUHX 명령만 해당)는 시간 계산 계속 수행(명령 실행 시 이외에도 현재값이 갱신되기 때문)

END

명령 이름	니모닉	동작 옵션	평션 번호	기능
종료	END	–	001	한 프로그램의 종료를 나타냅니다.

심볼	END
	END

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	불가능	불가능	불가능	가능	불가능

조건 플래그의 동작

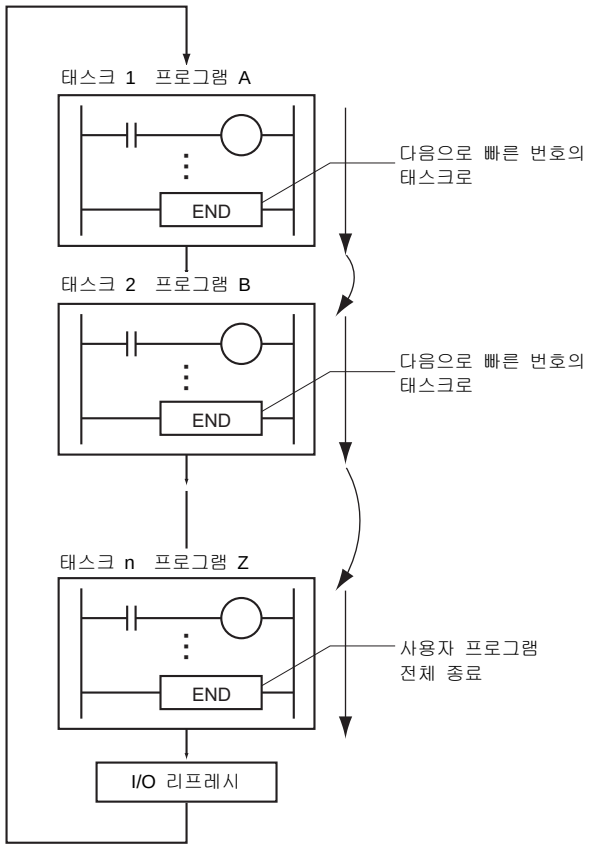
없음

기능

이 명령을 실행하여 하나의 프로그램에 대한 해당 사이클에서의 실행을 종료합니다. 따라서 END 명령 이후의 명령은 실행되지 않습니다.

사용자 프로그램 전체를 실행하는 경우에는 해당 사이클 내에서 기동된 사이클릭 태스크 중 다음으로 빠른 번호의 사이클릭 태스크 프로그램으로 실행 제어를 이동합니다.

이 명령이 기재된 프로그램의 사이클릭 태스크가 기동되어 있는 사이클릭 태스크 중에 가장 큰 태스크 번호인 경우, 이 명령은 사용자 프로그램 전체의 종료를 나타냅니다.



사용상의 주의 사항

· 한 프로그램의 마지막에는 반드시 END 명령을 넣으십시오. END 명령이 없으면 프로그램 에러가 발생합니다.

NOP

명령 이름	니모닉	동작 옵션	펑션 번호	기능
기능 없음	NOP	−	000	아무 기능도 없는 명령입니다.

심볼	NOP
	(래더도에는 표시되지 않습니다.)

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

조건 플래그의 동작

없음

기능

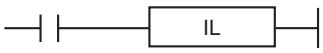
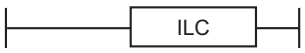
명령을 처리하지 않으며, 명령 삽입용으로 영역을 확보하기 위해 사용합니다.
니모닉 표시 시에만 사용할 수 있습니다.

힌트

· 접점을 삽입하고 싶은 위치에 NOP 명령을 기입해 두면 삽입한 후에도 프로그램 주소의 번호에 불일치가 생기지 않습니다.

IL/ILC

명령 이름	니모닉	동작 옵션	평션 번호	기능
인터록	IL	–	002	입력 조건이 OFF이면 IL 명령에서 ILC 명령까지의 출력을 인터록합니다.
인터록 클리어	ILC	–	003	인터록의 범위의 종료를 나타냅니다.

심볼	IL	ILC
		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	불가능	가능	가능	가능

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	변하지 않음*1
= 플래그	P_EQ	변하지 않음*1
부정 플래그	P_N	변하지 않음*1

*1: CS1/CJ1/CS1D(이중화 시스템용) CPU 유니트는 OFF

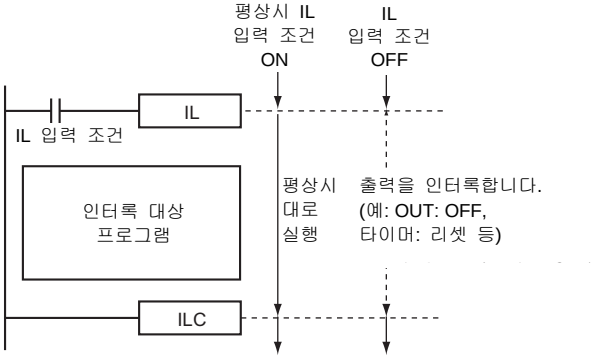
기능

IL 명령의 입력 조건이 OFF인 경우 IL 명령에서 ILC 명령까지 각 명령의 출력을 인터록합니다. IL 명령의 입력 조건이 ON인 경우에는 IL 명령에서 ILC 명령까지의 각 명령을 평상시대로 실행합니다.

인터록(IL) 상태에서 각 명령의 출력은 다음과 같습니다.

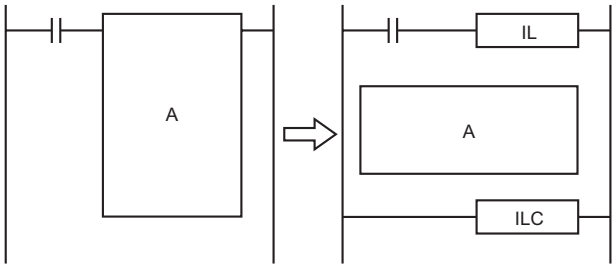
각 명령의 출력		상태
OUT, OUTB, OUT NOT 명령으로 지정된 릴레이		모두 OFF
타이머계 명령(100ms 타이머 TIM/TIMX, 10ms 타이머 TIMH/TIMHX, 1ms 타이머 TMHH/TMHHX, 장시간 타이머 TIML/TIMLX 명령)	현재값	타이머 설정값(리셋)
	업 플래그	OFF(리셋)
0.1ms 타이머 TIMU/TIMUX	참조 불가	
0.01ms 타이머 TMUH/TMUHX		
기타 명령(주)으로 지정된 릴레이, 채널		그 이전 상태 유지(명령 자체가 실행되지 않음)

주: TTIM/TTIMX, MTIM/MTIMX, SET, RSET, CNT/CNTX, CNTR/CNTRX, SFT, KEEP 등의 기타 모든 명령을 가리킵니다.



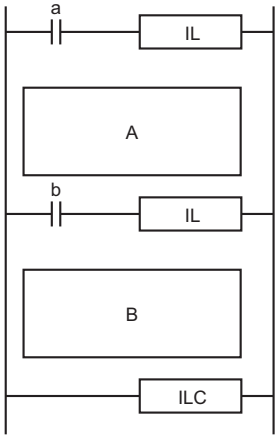
힌트

- IL-ILC 명령을 사용하면 프로그램 내에서 효율적인 회로 전환이 가능합니다.
- IL 명령 실행 중에도 ON을 유지하고 싶은 영역은 IL 명령 전에 SET 명령을 사용하여 ON으로 설정하십시오.
  동일한 입력 조건으로 복수의 처리를 실행하는 경우 해당하는 복수의 처리 직전과 직후에 각각 IL 명령과 ILC 명령을 배치하면 스텝 수를 줄일 수 있습니다.



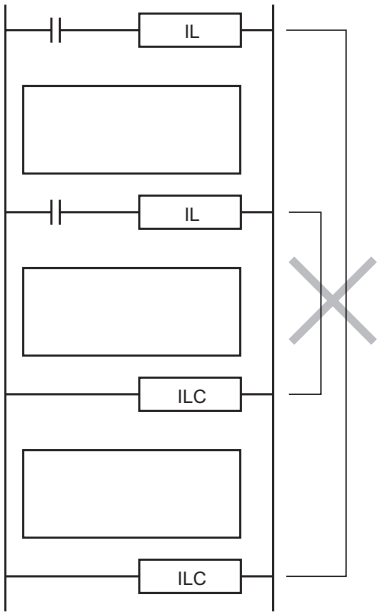
사용상의 주의 사항

- IL 명령에 의해 인터록이 된 경우에도 IL~ILC 간의 프로그램은 내부적으로 실행되므로 사이클 시간이 단축되지는 않습니다.
- IL 명령과 ILC 명령은 한 쌍으로 사용하십시오.
  한 쌍으로 사용하지 않으면(IL 명령과 ILC 명령 사이에 IL 명령이 있는 경우 등) 프로그램 확인 시, IL-ILC 에러가 발생합니다. 단, 동작은 오른쪽 그림과 같이 프로그램대로 동작합니다.



입력 조건		프로그램	
a	b	A	B
OFF	ON	인터록	인터록
OFF	OFF		
ON	OFF	비인터록	인터록
ON	ON	비인터록	비인터록

- IL 명령과 ILC 명령은 중첩(예: IL－IL－ILC－ILC)할 수 없습니다.
  중첩할 경우는 MLH－MILC 명령 또는 MILR－MILC 명령을 사용하십시오.

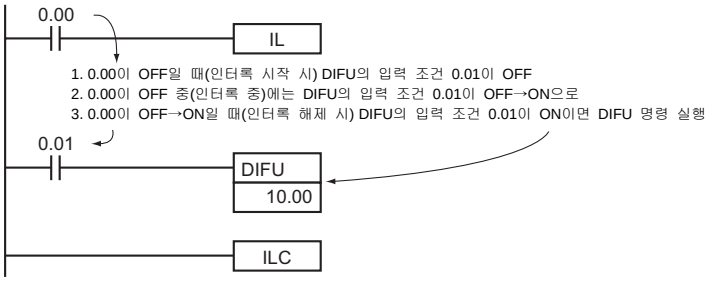


IL/ILC

· 미분 명령에 대한 동작

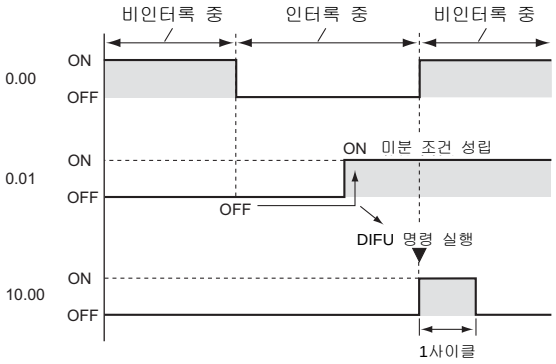
IL-ILC 명령 사이에 미분 명령(DIFU/DIFD/@/% 표시가 있는 명령)이 존재하는 경우, 입력 조건이 인터록 시작과 인터록 해제 시점 사이에 변하여 미분 조건이 성립하면 인터록 해제 시 미분 명령이 실행됩니다.

예: 기동 미분(DIFU) 명령의 경우 인터록 시작 시의 입력 조건이 OFF, 인터록 해제 시의 입력 조건이 ON이면 인터록 해제 시에 기동 미분(DIFU) 명령이 실행됩니다.



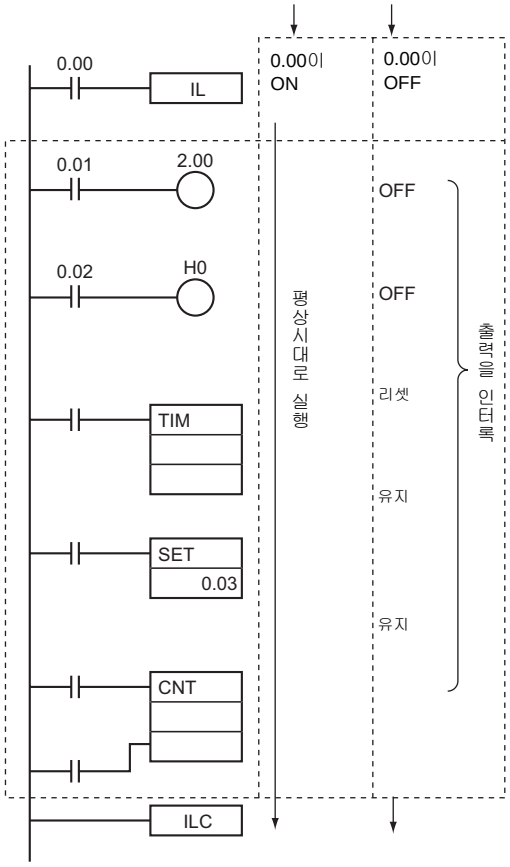
참고: IL 명령은 미분 명령의 동작에 대해 MILH 명령과 동일합니다.

타이밍 차트



샘플 프로그램

0.00이 OFF인 경우 IL ~ ILC 사이의 출력을 인터록합니다.
0.00이 ON이면 IL ~ ILC 사이의 명령을 평상시대로 실행합니다.



MILH/MILR/MILC

명령 이름	니모닉	동작 옵션	펑션 번호	기능
멀티 인터록(미분 플래그 유지형)	MILH	–	517	입력 조건이 OFF이면 MILH 명령에서 MILC 명령까지의 출력을 인터록합니다.
멀티 인터록(미분 플래그 비유지형)	MILR	–	518	입력 조건이 OFF이면 MILR 명령에서 MILC 명령까지의 출력을 인터록합니다.
멀티 인터록 클리어	MILC	–	519	인터록 번호가 동일한 MILH 또는 MILR 명령에 의한 멀티 인터록 범위의 종료를 나타냅니다.

심볼	MILH	MILR	MILC
	MILH N: 인터록 번호 D: 인터록 상태 출력 비트	MILR N: 인터록 번호 D: 인터록 상태 출력 비트	MILC N: 인터록 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	불가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	인터록 번호	–	1
D	인터록 상태 출력 비트	BOOL	–

N: 인터록 번호

0~15

한 쌍을 이루는 MILH(또는 MILR) 명령과 MILC 명령의 N(인터록 번호)을 일치시키십시오.

주: N(인터록 번호)의 사용 순서에는 대소 관계의 제한이 없습니다.

D: 인터록 상태 출력 비트

- 인터록 중이 아닐 때 ON
- 인터록 중일 때 OFF

MILH(또는 MILR) 명령에 의한 인터록 중에 이 비트를 강제 세트하면 인터록(IL)이 아닌 상태로 만들 수 있습니다. 반대로 인터록 중이 아닐 때 이 비트를 강제 리셋하면 인터록(IL) 상태로 만들 수도 있습니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
D	○	○	○	○							–			○*1				

*1: '1-1-7 자동 증가/감소 간접 인덱스 사용 시의 주의 사항'을 참조하십시오.

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF

MILH/MILR/MILC

기능

인터록 번호 N의 MILH(또는 MILR) 명령의 입력 조건이 OFF일 때, 해당 MILH(또는 MILR) 명령부터 동일한 인터록 번호 N의 MILC 명령까지의 각 명령에 대한 출력을 인터록합니다.

인터록 번호 N의 MILH(또는 MILR) 명령의 입력 조건이 ON일 때, 해당 MILH(또는 MILR) 명령부터 동일한 인터록 번호 N의 MILC 명령까지의 각 명령을 평상시대로 실행합니다.

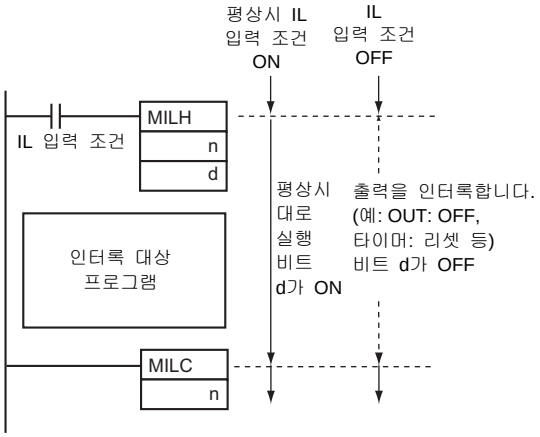
■ 인터록(IL) 상태

인터록(IL) 상태에서 각 명령의 출력은 다음과 같습니다.

각 명령의 출력		상태
OUT, OUTB, OUT NOT 명령으로 지정된 릴레이		모두 OFF
타이머계 명령(100ms 타이머 TIM/TIMX, 10ms 타이머 TIMH/ TIMHX, 1ms 타이머 TMHH/TMHHX, 장시간 타이머 TIML/TIMLX 명령)	현재값	타이머 설정값(리셋)
	업 플래그	OFF(리셋)
0.1ms 타이머 TIMU/TIMUX 0.01ms 타이머 TMUH/TMUHX	참조 불가	
기타 명령(주)으로 지정된 릴레이, 채널		그 이전 상태 유지(명령 자체가 실행되지 않음)

주: TTIM/TTIMX, MTIM/MTIMX, SET, RSET, CNT/CNTX, CNTR/CNTRX, SFT, KEEP 등의 기타 모든 명령을 가리킵니다.

또한 MILH 명령(또는 MILR 명령)의 제2피연산자 D(인터록 상태 출력 비트)에는 인터록 중일 때 OFF, 인터록 중이 아닐 때 ON을 출력합니다. 따라서 이 비트를 모니터링하면 각 인터록 번호별 인터록(IL) 상태를 모니터링할 수 있습니다.



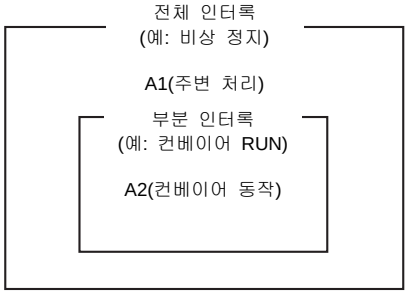
■ 중첩

중첩이란 MILH(또는 MILR) 명령과 MILC 명령 사이에 또 다른 MILH(또는 MILR) 명령과 MILC 명령이 존재하는 중첩 구조를 가리킵니다.

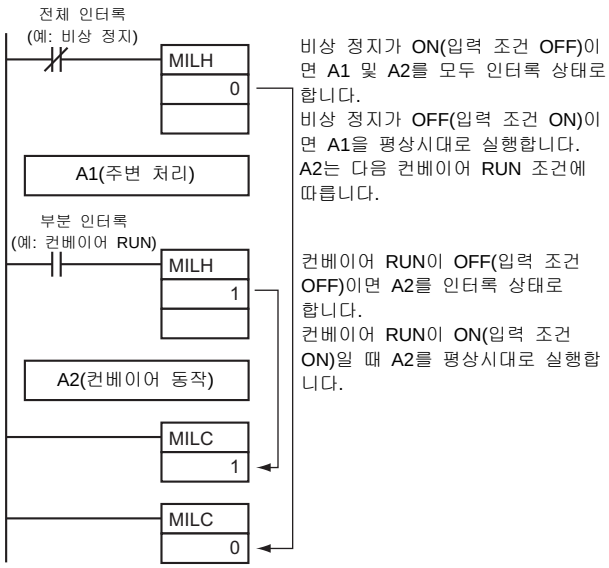
MILH(또는 MILR)－MILC 명령의 중첩(예: MILH n－MILH m－MILC m－MILC n)은 최대 16개까지 가능합니다.

중첩은 다음과 같은 용도로 사용합니다.

예 1) 전체를 인터록하는 조건과 일부분을 인터록하는 조건이 있는 경우(중첩 수 1회)



· 비상 정지 버튼의 ON에서 A1, A2를 인터록
· 컨베이어 RUN의 OFF에서 A2를 인터록



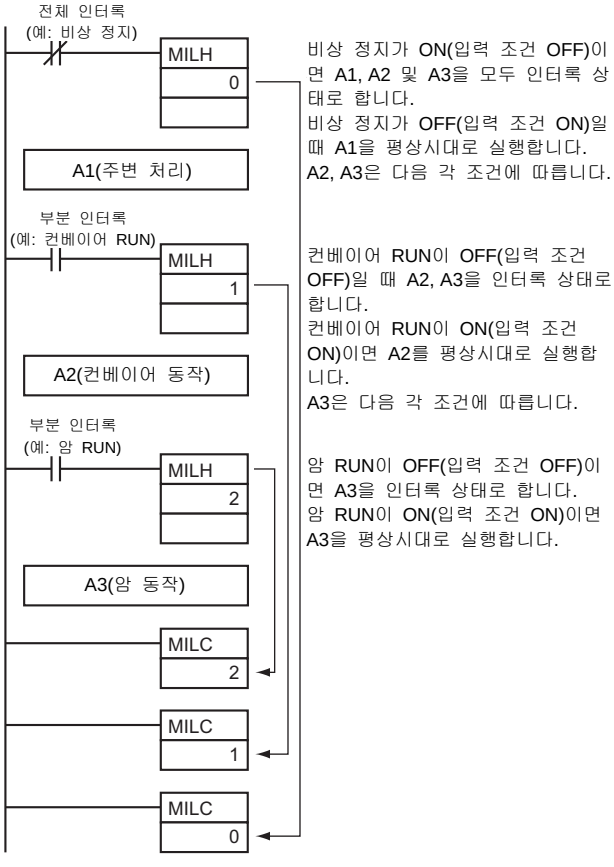
비상 정지가 ON(입력 조건 OFF)이면 A1 및 A2를 모두 인터록 상태로 합니다.
비상 정지가 OFF(입력 조건 ON)이면 A1을 평상시대로 실행합니다.
A2는 다음 컨베이어 RUN 조건에 따릅니다.

컨베이어 RUN이 OFF(입력 조건 OFF)이면 A2를 인터록 상태로 합니다.
컨베이어 RUN이 ON(입력 조건 ON)일 때 A2를 평상시대로 실행합니다.

예 2) 전체를 인터록하는 조건과 일부분을 인터록하는 이중의 조건이 있는 경우(중첩 수 2회)



· 비상 정지 버튼의 ON에서 A1, A2, A3을 인터록
· 컨베이어 RUN의 OFF에서 A2, A3을 인터록
· 암 RUN의 OFF에서 A3을 인터록



비상 정지가 ON(입력 조건 OFF)이면 A1, A2 및 A3을 모두 인터록 상태로 합니다.
비상 정지가 OFF(입력 조건 ON)일 때 A1을 평상시대로 실행합니다.
A2, A3은 다음 각 조건에 따릅니다.

컨베이어 RUN이 OFF(입력 조건 OFF)일 때 A2, A3을 인터록 상태로 합니다.
컨베이어 RUN이 ON(입력 조건 ON)이면 A2를 평상시대로 실행합니다.
A3은 다음 각 조건에 따릅니다.

암 RUN이 OFF(입력 조건 OFF)이면 A3을 인터록 상태로 합니다.
암 RUN이 ON(입력 조건 ON)이면 A3을 평상시대로 실행합니다.

MILH/MILR/MILC

■ MILH 명령과 MILR 명령의 차이점

MILH 명령과 MILR 명령은 MILC 명령까지의 각 명령 중에 미분 명령(DIFU/DIFD/@/% 표시가 있는 명령)이 존재하는 경우 동작이 서로 다릅니다.

MILR 명령의 경우, 인터록 중에(인터록 시작과 인터록 해제 시점 사이에 입력 조건이 바뀐 경우) 미분 조건이 성립해도 그 조건 성립은 취소되고, 인터록 해제 후 미분 명령은 실행되지 않습니다.

그에 비해 MILH 명령의 경우, 인터록 시작 시와 인터록 해제 시의 값에서 미분 조건이 성립하면 미분 조건 성립이 유효하게 되고, 인터록 해제 후 미분 명령(DIFU/DIFD/@/% 표시가 있는 명령)이 실행됩니다.

각 명령	입력 조건이 인터록 시작과 인터록 해제 시점 사이에 변하여 미분 명령(DIFU/DIFD/@/% 표시가 있는 명령)의 미분 조건이 성립할 때
MILH(멀티 인터록(미분 플래그 유지형)) 명령	인터록 해제 후 미분 명령이 실행됩니다.
MILR(멀티 인터록(미분 플래그 비유지형)) 명령	인터록 해제 후 미분 명령이 실행되지 않습니다.

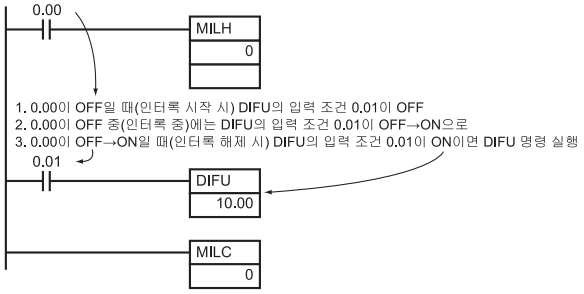
■ MILH 명령에서 미분 명령에 대한 동작

MILH-MILC 명령 사이에 미분 명령(DIFU/DIFD/@/% 표시가 있는 명령)이 존재하는 경우, 입력 조건이 인터록 시작 시와 인터록 해제 시의 값 사이에서 미분 조건이 성립하면 인터록 해제 시 미분 명령이 실행됩니다. 마찬가지로 인터록 시작 시 또는 인터록 해제 시와 같은 타이밍에 미분 조건이 성립한 경우 미분 명령이 실행됩니다.

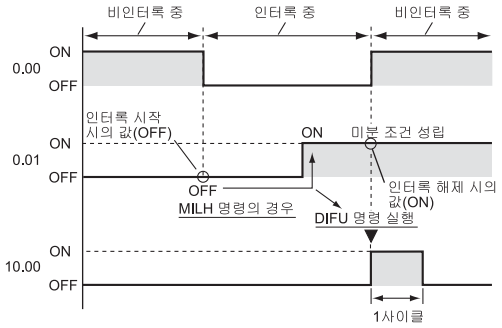
단, 인터록 상태에서 미분 조건이 성립해도 프로그램의 여러 조건에 의해 미분 조건이 성립하지 않아 인터록 해제 시 미분 조건이 성립하지 않은 경우 미분 명령은 실행되지 않습니다.

예: 기동 미분(DIFU) 명령의 경우 인터록 시작 시의 입력 조건이 OFF, 인터록 해제 시의 입력 조건이 ON이면 인터록 해제 시에 기동 미분(DIFU) 명령이 실행됩니다.

(또한 IL 명령은 미분 명령의 동작에 대해 MILH 명령과 동일합니다.)



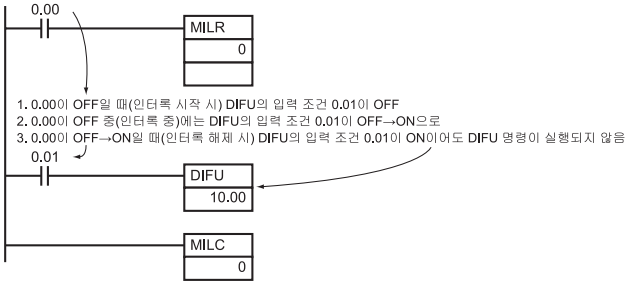
타이밍 차트



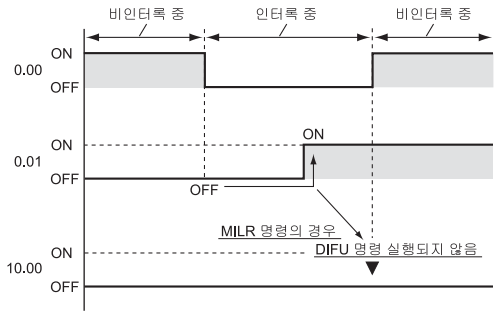
■ MILR 명령에서 미분 명령에 대한 동작

MILR－MILC 명령 사이에 미분 명령(DIFU/DIFD/@/% 표시가 있는 명령)이 존재하는 경우, 입력 조건이 인터록 시작 시와 인터록 해제 시의 값 사이에서 미분 조건이 성립해도 인터록 해제 시 미분 명령은 실행되지 않습니다. 마찬가지로 인터록 시작 시 또는 인터록 해제 시와 같은 타이밍에 미분 조건이 성립한 경우도 미분 명령은 실행되지 않습니다.

예: 기동 미분(DIFU) 명령의 경우 인터록 시작 시의 입력 조건이 OFF, 인터록 해제 시의 입력 조건이 ON이면 인터록 해제 시 기동 미분(DIFU) 명령은 실행되지 않습니다.



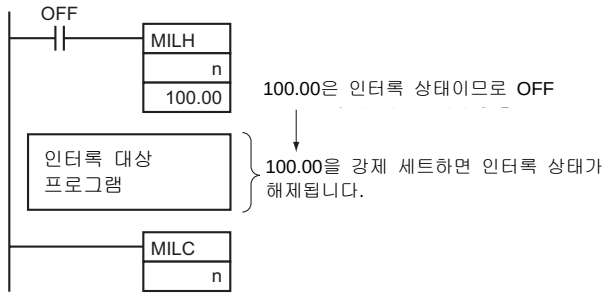
타이밍 차트



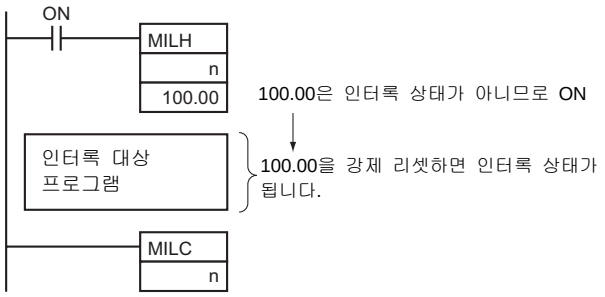
■ 주변 툴을 사용한 강제 세트/리셋 조작에 의한 인터록(IL) 상태 조작

주변 툴에서 MILH 명령(또는 MILR 명령)의 제2피연산자 D(인터록 상태 출력 비트)에 대해 강제 세트/리셋 조작을 하면 그 시점의 명령 실행에 의한 인터록 상태와 관계 없이 수동 조작으로 비인터록 상태/인터록 상태를 만들 수 있습니다.

강제 세트: 비인터록(IL) 상태가 됨

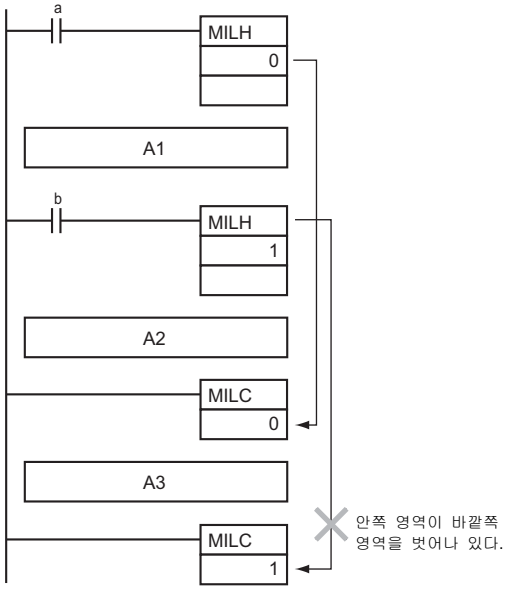


강제 리셋: 인터록(IL) 상태가 됨



힌트

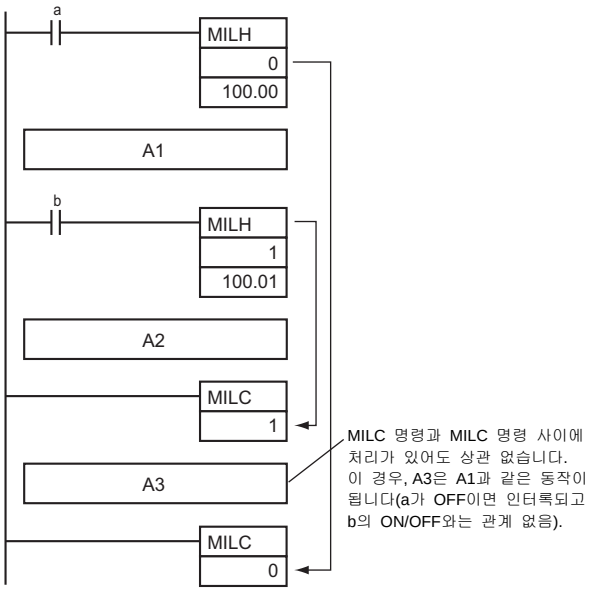
- MILH(또는 MILR) 명령에 의해 인터록 상태(IL 중)가 된 경우에도 MILH(또는 MILR)~MILC 사이의 프로그램은 내부적으로 실행되므로 사이클 시간이 단축되지는 않습니다.
- 안쪽의 MILH(또는 MILR)－MILC 명령 영역이 바깥쪽의 MILH(또는 MILR)－MILC 명령의 영역을 벗어나지 않도록 MILC 명령의 인터록 번호를 기재하십시오.



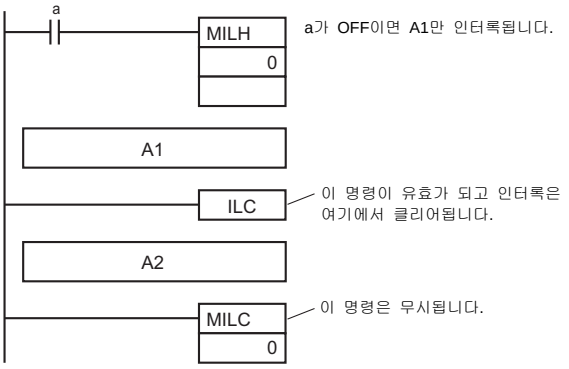
입력 조건		프로그램		
a	b	A1	A2	A3
OFF	ON	인터록	인터록	비인터록
	OFF			
ON	OFF	비인터록	인터록	인터록
	ON	비인터록	비인터록	비인터록

MILH/MILR/MILC

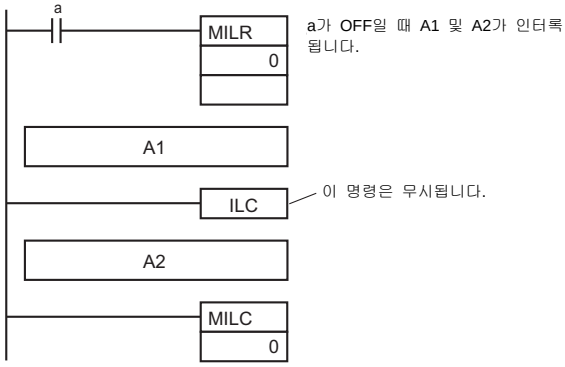
· MILC 명령과 MILC 명령 사이에 처리가 있어도 상관 없습니다.



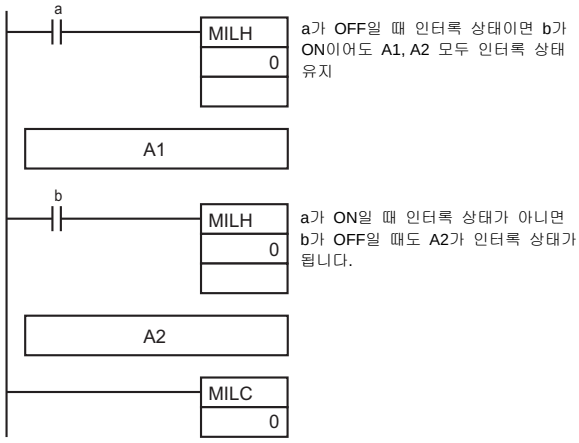
· MILH 명령-MILC 명령 사이에 ILC 명령이 존재하는 경우, MILH 명령-ILC 명령 사이가 인터록 대상이 됩니다.



· MILR 명령-MILC 명령 사이에 ILC 명령이 존재하는 경우, ILC 명령은 무시되고 MILR 명령-MILC 명령 사이가 인터록 대상이 됩니다.

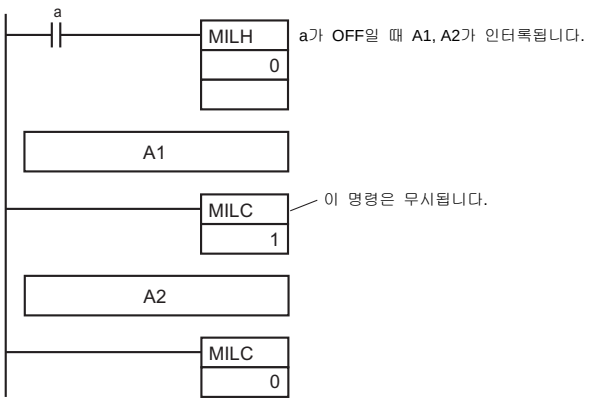


· MILH 명령-MILC 명령 사이에 같은 인터록 번호의 MILH(또는 MILR) 명령이 존재하는 경우,
  · 바깥쪽 MILH 명령에 의해 인터록 상태가 되었을 때는 안쪽의 MILH(또는 MILR) 명령에 의해 인터록 상태가 되지는 않습니다.
  · 바깥쪽 MILH 명령에 의해 비인터록 상태가 되었을 때는 안쪽의 MILH(또는 MILR) 명령에 의해 인터록 상태가 됩니다.



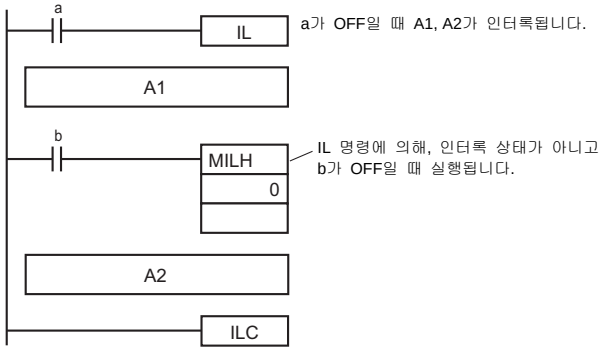
주: MILR 명령-MILC 명령 사이에 MILH(또는 MILR) 명령이 존재하는 경우도 동일합니다.

· MILH(또는 MILR) 명령-MILC 명령 사이에 다른 인터록 번호의 MILC 명령이 존재하는 경우 다른 인터록 번호의 MILC 명령은 무시됩니다.

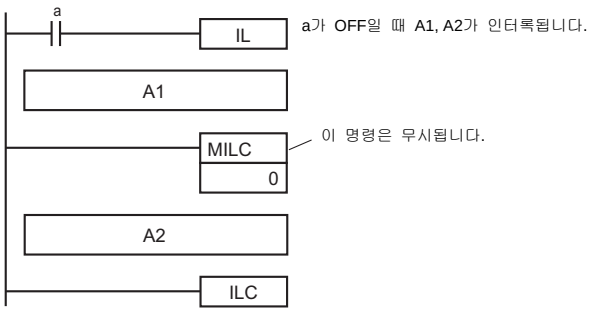


· IL 명령-ILC 명령 사이에 MILH 명령이 존재하는 경우, MILH 명령은 IL 명령에 의한 인터록 상태인지 여부를 스스로 판단하여 인터록 상태이면 아무 실행도 하지 않습니다. 이때 IL-ILC 명령 사이가 인터록 대상입니다.

또한 IL 명령에 의한 비인터록 상태에서 b가 OFF일 때는 MILH 명령이 실행되고 MILH-ILC 사이가 인터록 상태가 됩니다.

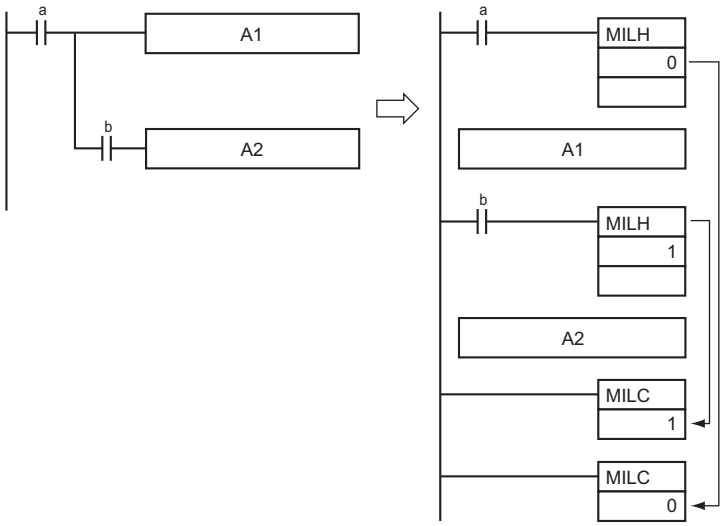


· IL 명령-ILC 명령 사이에 MILC 명령이 존재하는 경우, MILC 명령은 무시되고 IL-ILC 명령 사이가 인터록 대상이 됩니다.



· MILH(또는 MILR)-MILC 명령을 사용하면 프로그램 내에서 효율적인 회로 전환이 가능합니다.

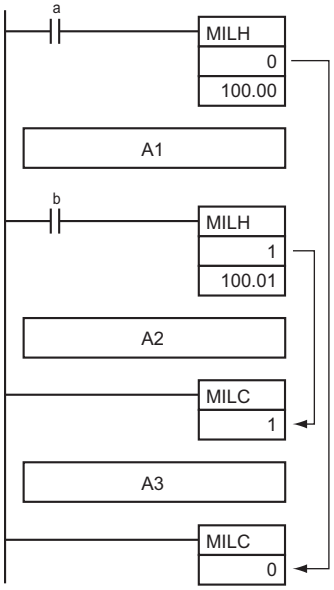
복합 조건에 맞춰 처리를 전환하고 싶은 경우, 각 처리 직전에 MILH(또는 MILR) 명령을 배치하고 마지막에는 MILC 명령을 배치합니다.



MILH/MILR/MILC

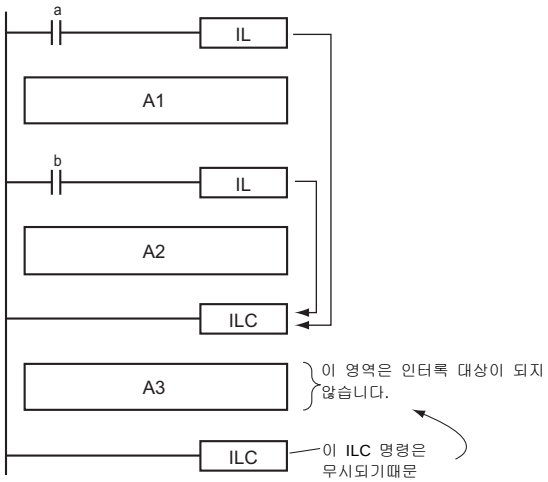
· MILH(또는 MILR)-MILC 명령은 IL-ILC 명령과 달리 중첩이 가능하므로 동일한 래더 프로그램을 기술해도 다음과 같이 동작이 달라집니다.

· MILH-MILC 명령



입력 조건		프로그램		
a	b	A1	A2	A3
OFF	ON	인터록	인터록	인터록
	OFF			
ON	OFF	비인터록	인터록	비인터록
ON	ON	비인터록	비인터록	비인터록

· IL-ILC 명령



입력 조건		프로그램		
a	b	A1	A2	A3
OFF	ON	인터록	인터록	인터록 대상 이외이므로 비인터록
	OFF			
ON	OFF	비인터록	인터록	
ON	ON	비인터록	비인터록	

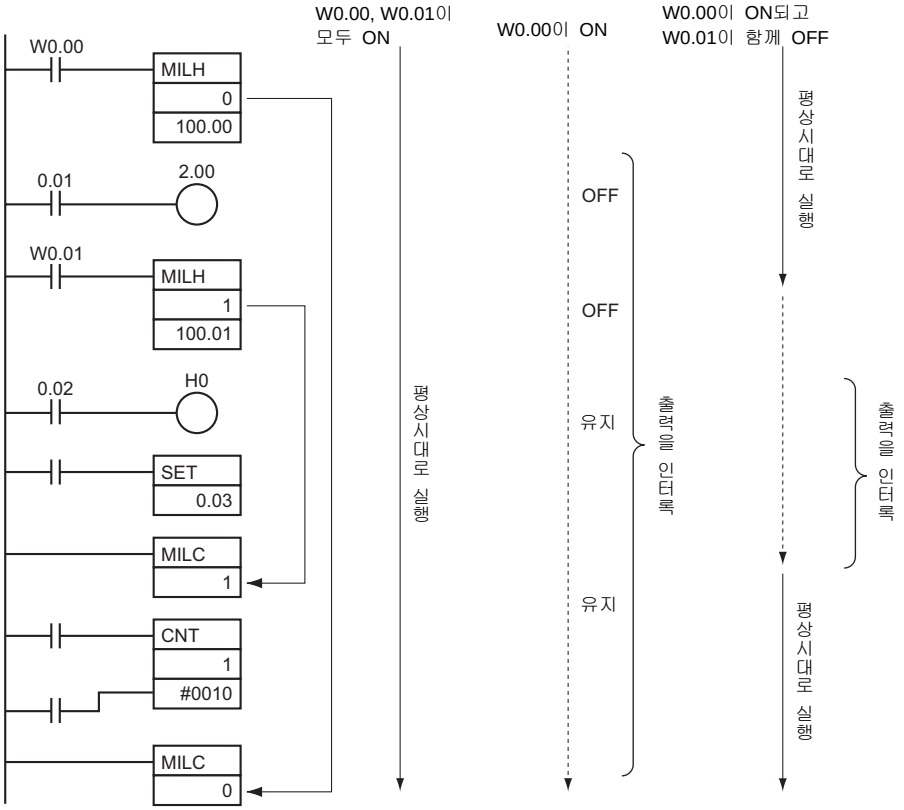
· MILH(또는 MILR) 명령 실행 중에도 ON을 유지하고 싶은 영역은 MILH(또는 MILR) 명령 전에 SET 명령을 사용하여 ON으로 설정하십시오.

샘플 프로그램

W0.00, W0.01이 모두 ON이면 MILH(인터록 번호 0)~MILC(인터록 번호 0) 사이의 명령을 평상시대로 실행합니다.

W0.00이 OFF이면 MILH(인터록 번호 0)~MILC(인터록 번호 0) 사이의 명령을 인터록합니다.

W0.00이 ON이고 W0.01이 OFF이면 MILH(인터록 번호 1)~MILC(인터록 번호 1) 사이의 명령을 인터록합니다. 그 이외는 평상시대로 실행합니다.



JMP/JME

명령 이름	니모닉	동작 옵션	펑션 번호	기능
점프	JMP	−	004	입력 조건이 OFF일 때 JME 명령까지 직접 점프합니다.
점프 종료	JME	−	005	JMP 또는 CJP 명령에 의한 점프의 종료 위치를 나타냅니다.

심볼	JMP	JME
	JMP N: 점프 번호	JME N: 점프 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	가능	불가능	가능	가능	불가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	점프 번호	UINT	1

N: 10진수 &0~1023 또는 16진수 #0000~00FF*1
JME 명령은 N에 상수 지정만 가능

*1: CJ1M-CPU11/21의 경우, 10진수 &0~255 또는 16진수 #0000~00FF

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
JMP	N	○	○	○	○	○	○	○	○	○	○	○*1	○	−	○	−	−	−	−
JME	N	−	−	−	−	−	−	−	−	−	−		−	−	−				

*1: CJ1M-CPU11/21의 경우, 10진수 &0~255 또는 16진수 #0000~00FF

조건 플래그의 동작

■ JMP

이름	라벨	내용
에러 플래그	P_ER	· N의 데이터가 10진수 &0~1023 또는 16진수 #0000~03FF*1의 범위에 없을 때 ON · 해당하는 점프 번호의 JME 명령이 존재하지 않을 때 ON · 해당하는 점프 번호의 JME 명령이 동일 태스크 내에 존재하지 않을 때 ON · 그 이외일 때 OFF

*1: CJ1M-CPU11/21의 경우, 10진수 &0~255 또는 16진수 #0000~00FF

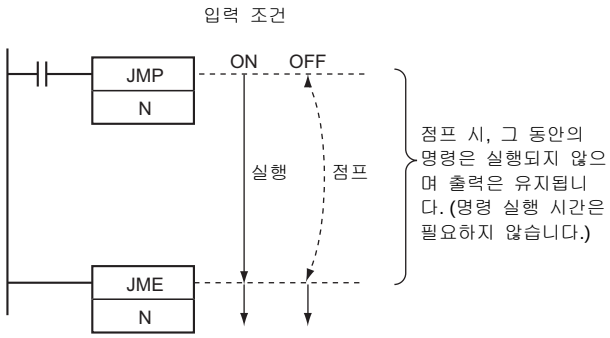
■ JME

없음

기능

JMP 명령의 입력 조건이 OFF이면 N으로 지정한 점프 번호를 가진 JME 명령까지 점프하고, 입력 조건이 ON이면 다음 명령 이후를 실행합니다.

입력 조건이 OFF이면 JMP-JME 사이의 명령은 실행되지 않습니다. 따라서 출력은 상태를 유지합니다. 단, 블록 프로그램 영역 안에서는 입력 조건에 관계없이 점프합니다.



힌트

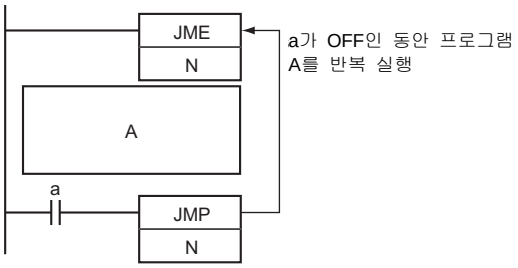
- JMP/CJP/CJPN 명령의 경우, 점프 조건이 성립하면 직접 JME 명령까지 점프합니다. JMP/CJP/CJPN-JME 사이의 명령을 실행하지 않으므로 그 사이의 명령 실행 시간이 걸리지 않습니다. 따라서 사이클 시간이 단축됩니다.

  그에 비해 JMP0 명령의 경우, 점프 조건이 성립하면 JMP0-JME0 사이는 NOP 처리(무기능 처리)를 실행합니다. 그 사이에 NOP 명령에 상당하는 처리 시간이 필요하므로 사이클 시간은 단축되지 않습니다.
- 점프 명령의 기능 비교

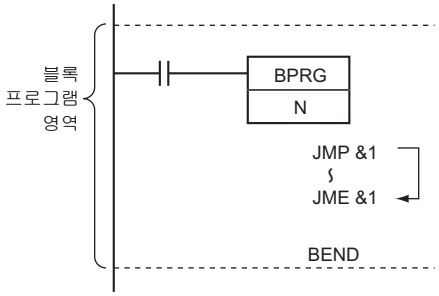
점프 명령	JMP-JME	CJP-JME	CJPN-JME	JMP0-JME0
점프하는 입력 조건	OFF	ON	OFF	OFF
사용 개수	합계 1024개 (CJ1M-CPU11/21은 256개)			제한 없음
점프 시 명령 처리	비실행			NOP 처리
점프 시 실행 시간	없음			NOP 명령 상당의 합계 시간
점프 시 명령 출력	직전의 상태 유지			
점프 시 기동 중인 타이머 현재값 갱신	계속 갱신			
블록 프로그램 영역 내 처리	무조건 점프	ON에서 점프	OFF에서 점프	사용 불가능

사용상의 주의 사항

- 점프되면 모든 명령의 출력(릴레이, 채널)이 그 이전 상태를 유지합니다. 단, TIM/TIMX 명령/TIMH/TIMHX 명령/TMHH/TMHHX 명령/TIMU/TIMUX 명령/TMUH/TMUHX 명령으로 기동된 타이머는 명령 실행 시 이외에도 현재값 갱신 처리가 수행되므로 시간 계산이 계속됩니다. (TIMU/TIMUX 명령, TMUH/TMUHX 명령을 기동 중에 점프하면 오차가 생기는 경우가 있으므로 주의하십시오.)
- 같은 번호를 가진 JME 명령이 2개 이상인 경우 프로그램 주소가 작은 JME 명령이 유효하게 됩니다. 이때, 주소가 큰 JME 명령은 무시됩니다.
- 프로그램 주소가 작은 쪽으로 점프하는 경우, JMP의 입력 조건이 OFF인 동안은 JMP-JME 사이를 반복해서 실행합니다. JMP의 입력 조건이 ON이 되면 반복을 종료합니다. 이런 경우, 계속 JMP의 입력 조건이 ON이 되지 않으면 END 명령을 실행하지 않고 사이클 시간이 초과될 가능성이 있으므로 주의하십시오.



- 블록 프로그램 영역 내에서는 JMP 명령 직전의 입력 조건과 관계 없이 항상 점프합니다.



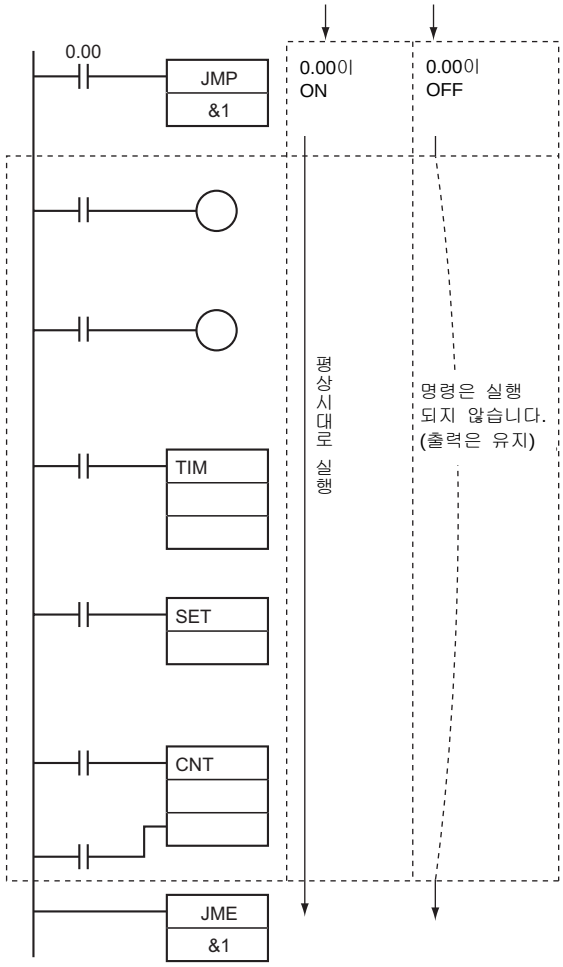
JMP/JME

- 점프 대상 JME 명령은 JMP 명령이 존재하는 같은 태스크 내에서 작성하십시오. 태스크 간에는 점프를 실행할 수 없습니다. JMP 명령과 JME 명령을 서로 다른 태스크에 기술하면 에러가 발생하여 에러 플래그가 ON이 됩니다.
- JMP/JME 명령 사이에서 미분 명령(DIFU/DIFD/@/% 표시가 있는 명령)을 사용한 경우, 입력 조건에 따라 동작이 안정되지 않는 경우가 있으므로 주의하십시오.

샘플 프로그램

0.00이 OFF이면 JMP~JME &1 사이의 명령을 실행하지 않고 출력을 유지합니다.

0.00이 ON이면 JMP~JME &1 사이의 명령을 평상시대로 실행합니다.



CJP/CJPN

명령 이름	니모닉	동작 옵션	평션 번호	기능
조건부 점프	CJP	–	510	입력 조건이 ON일 때 JME 명령까지 직접 점프합니다.
조건부 부정 점프	CJPN	–	511	점프 조건이 OFF일 때 JME 명령까지 직접 점프합니다.

심볼	CJP	CJPN
	CJP N N: 점프 번호	CJPN N N: 점프 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	가능	불가능	가능	가능	불가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	점프 번호	UINT	1

N: 점프 번호

10진수 &0~1023 또는 16진수 #0000~03FF*1

*1: CJ1M-CPU11/21의 경우, 10진수 &0~255 또는 16진수 #0000~00FF

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
CJP/ CJPN	N	○	○	○	○	○	○	○	○	○	○	○*1	○	–	○	–	–	–	–

*1: CJ1M-CPU11/21의 경우, 10진수 &0~255 또는 16진수 #0000~00FF

조건 플래그의 동작

■ CJP/CJPN

이름	라벨	내용
에러 플래그	P_ER	· 해당하는 점프 번호의 JME 명령이 존재하지 않을 때 ON · N의 데이터가 10진수 &0~1023 또는 16진수 #0000~03FF*1의 범위에 없을 때 ON · N의 점프 번호에 해당하는 JME 명령이 동일 태스크 내에 존재하지 않을 때 ON · 그 이외일 때 OFF

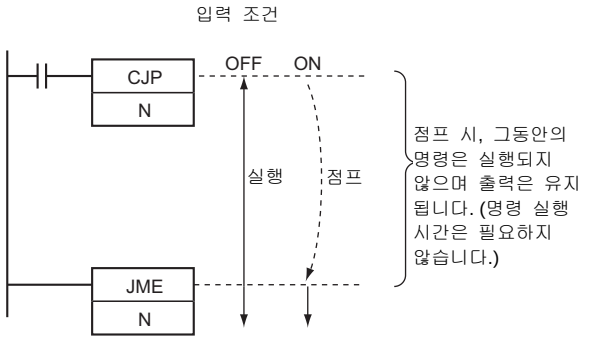
*1: CJ1M-CPU11/21의 경우, 10진수 &0~255 또는 16진수 #0000~00FF

CJP/CJPN

기능

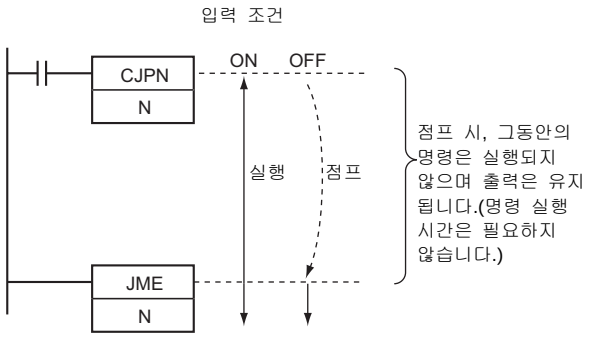
■ CJP

입력 조건이 ON이면 N의 점프 번호를 가진 JME 명령까지 점프합니다. 입력 조건이 OFF이면 다음 명령 이후를 실행합니다.



■ CJPN

입력 조건이 OFF이면 N의 점프 번호를 가진 JME 명령까지 점프합니다. 입력 조건이 ON이면 다음 명령 이후를 실행합니다.



힌트

- JMP/CJP/CJPN 명령의 경우, 점프 조건이 성립하면 직접 JME 명령까지 점프합니다. JMP/CJP/CJPN-JME 사이의 명령을 실행하지 않으므로 그 사이의 명령 실행 시간이 걸리지 않습니다. 따라서 사이클 시간이 단축됩니다.

  그에 비해 JMP0 명령의 경우, 점프 조건이 성립하면 JMP0-JME0 사이는 NOP 처리(무기능 처리)를 실행합니다. 그 사이에 NOP 명령에 상당하는 처리 시간이 필요하므로 사이클 시간은 단축되지 않습니다.
- 점프 명령의 기능 비교

점프 명령	JMP-JME	CJP-JME	CJPN-JME	JMP0-JME0
점프하는 입력 조건	OFF	ON	OFF	OFF
사용 개수	합계 1024개 (CJ1M-CPU11/21은 256개)			제한 없음
점프 시 명령 처리	비실행			NOP 처리
점프 시 실행 시간	없음			NOP 명령 상당의 합계 시간
점프 시 명령 출력	직전의 상태 유지			
블록 프로그램 영역 내 처리	무조건 점프	ON에서 점프	OFF에서 점프	사용 불가능

사용상의 주의 사항

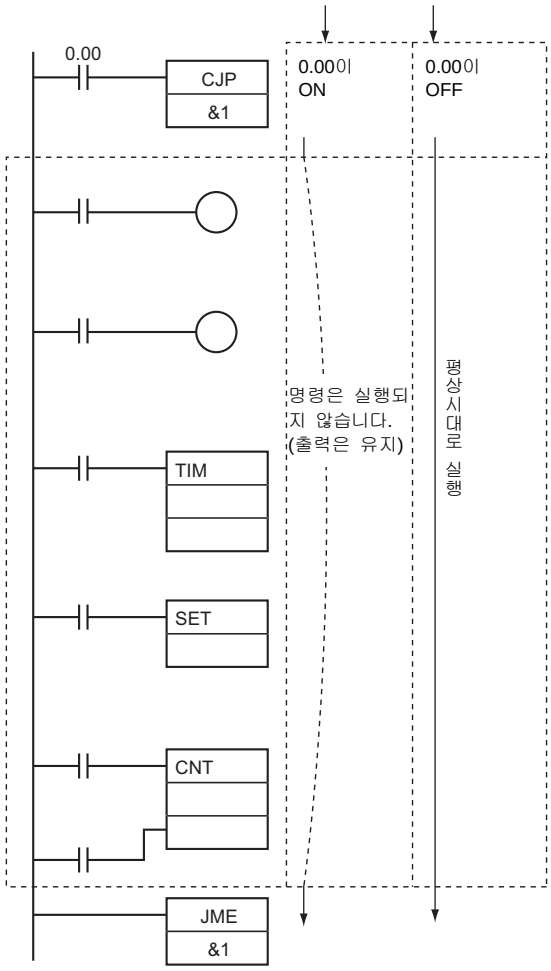
- CJP 명령은 입력 조건이 ON일 때 점프하며 JMP 명령과 반대로 동작합니다.
- 점프되면 모든 명령의 출력(릴레이, 채널)이 그 이전 상태를 유지합니다. 단, TIM/TIMX 명령/TIMH/TIMHX 명령/TMHH/TMHHX 명령/TIMU/TIMUX 명령/TMUH/TMUHX 명령으로 기동된 타이머는 명령 실행 시 이외에도 현재값 갱신 처리가 수행되므로 시간 계산이 계속됩니다. (TIMU/TIMUX 명령, TMUH/TMUHX 명령을 기동 중에 점프하면 오차가 생기는 경우가 있으므로 주의하십시오.)
- 같은 번호를 가진 JME 명령이 2개 이상인 경우 주소가 작은 JME 명령이 유효하게 됩니다. 이때, 주소가 큰 JME 명령은 무시됩니다.
- 프로그램 주소가 작은 쪽으로 점프하는 경우 CJP(CJPN)의 입력 조건이 ON(OFF)인 동안은 CJP(CJPN)-JME 사이를 반복해서 실행합니다. CJP(CJPN)의 입력 조건이 OFF(ON)가 되면 반복을 종료합니다. 이런 경우, 계속 CJP(CJPN)의 입력 조건이 OFF(ON)가 되지 않으면 END 명령을 실행하지 않고 사이클 시간이 초과될 가능성이 있으므로 주의하십시오.
- CJP(CJPN) 명령의 경우 CJP(CJPN) 조건이 ON(OFF)일 때 직접 JME 명령까지 점프하여 CJP(CJPN)-JME 사이의 명령을 실행하지 않으므로 그 사이의 명령 실행 시간이 걸리지 않습니다. 따라서 사이클 시간이 단축됩니다.
- 블록 프로그램 영역 내에서는 CJP(CJPN) 명령 직전의 입력 조건이 ON(OFF)일 때 점프합니다.

- 점프 대상 JME 명령은 CJP(CJPN) 명령이 존재하는 같은 태스크 내에서 작성하십시오. 태스크 간에는 점프를 실행할 수 없습니다. CJP(CJPN) 명령과 JME 명령을 서로 다른 태스크에 기술하면 에러가 발생하여 에러 플래그가 ON이 됩니다.
- CJP(CJPN)-JME 명령 사이에서 미분 명령(DIFU/DIFD/@/% 표시가 있는 명령)을 사용한 경우, 입력 조건에 따라 동작이 안정되지 않는 경우가 있으므로 주의하십시오.

샘플 프로그램

0.00이 ON이면 CJP~JME &1 사이의 명령을 실행하지 않고 출력을 유지합니다.

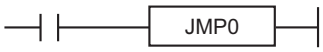
0.00이 OFF이면 CJP~JME &1 사이의 명령을 평상시대로 실행합니다.



주: CJPN 명령의 경우 입력 조건의 ON과 OFF가 반대로 됩니다.

JMP0/JME0

명령 이름	니모닉	동작 옵션	평션 번호	기능
복수 점프	JMP0	−	515	JMP0 명령의 입력 조건이 OFF일 때 JMP0의 다음 명령에서 JME0 명령까지 NOP 처리합니다.
복수 점프 종료	JME0	−	516	JMP0 명령에 의한 점프의 종료 위치를 나타냅니다.

심볼	JMP0	JME0
		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	불가능	가능	가능	가능

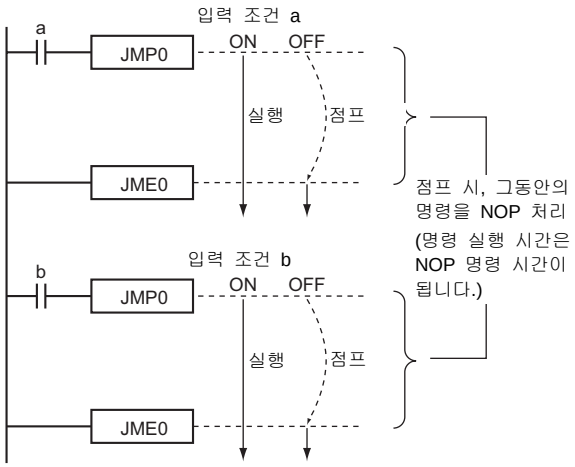
조건 플래그의 동작

없음

기능

JMP0 명령의 입력 조건이 OFF이면 JMP0 명령에서 JME0 명령까지 NOP 처리합니다. 입력 조건이 ON이면 다음 명령 이후를 실행합니다.

JMP/CJP/CJPN 명령과 달리, 점프 번호를 사용하지 않으므로 프로그램 중간의 어느 위치에서든지 사용할 수 있습니다.



힌트

· JMP/CJP/CJPN 명령의 경우, 점프 조건이 성립하면 직접 JME 명령까지 점프합니다. JMP/CJP/CJPN-JME 사이의 명령을 실행하지 않으므로 그 사이의 명령 실행 시간이 걸리지 않습니다. 따라서 사이클 시간이 단축됩니다.

그에 비해 JMP0 명령의 경우, 점프 조건이 성립하면 JMP0-JME0 사이는 NOP 처리(무기능 처리)를 실행합니다. 그 사이에 NOP 명령에 상당하는 처리 시간이 필요하므로 사이클 시간은 단축되지 않습니다.

· 점프 명령의 기능 비교

점프 명령	JMP-JME	CJP-JME	CJPN-JME	JMP0-JME0
점프하는 입력 조건	OFF	ON	OFF	OFF
사용 개수	합계 1024개(CJ1M-CPU11/21은 256개)			제한 없음
점프 시 명령 처리	비실행			NOP 처리
점프 시 실행 시간	없음			NOP 명령 상당의 합계 시간
점프 시 명령 출력	직전의 상태 유지			
블록 프로그램 영역 내 처리	무조건 점프	ON에서 점프	OFF에서 점프	사용 불가능

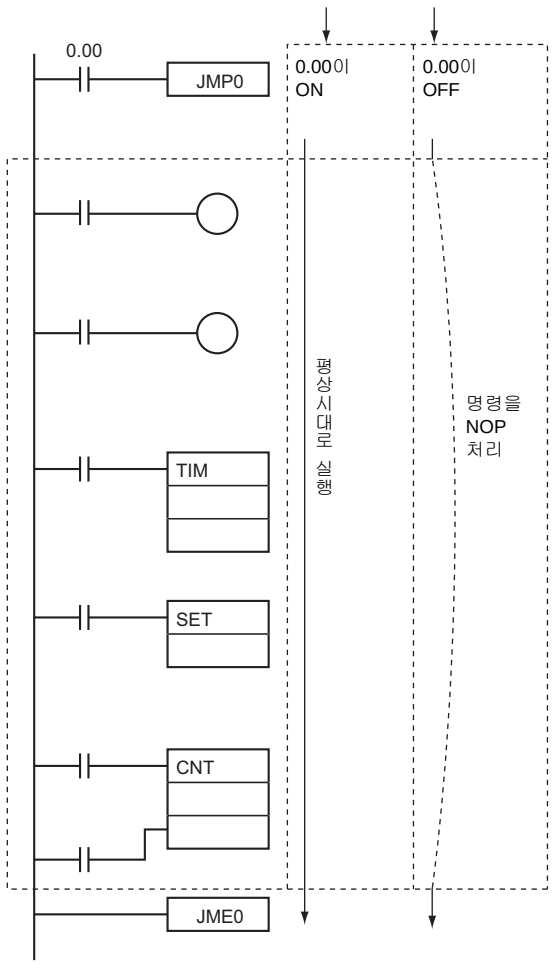
사용상의 주의 사항

- JMP/CJP/CJPN 명령이 직접 JME 명령까지 명령을 건너뛰고 실행하는 것과 달리, JMP0~JME0 명령 사이의 명령을 NOP 처리하므로 실행 시간이 필요합니다. 따라서 사이클 시간이 단축되지 않습니다. 또한 명령을 실행하지 않으므로 출력은 유지됩니다.
- JMP0 명령은 동일 프로그램상에서 여러 개 사용할 수 있습니다. 단, 여러 개 사용하는 경우 JMP0~JME0 사이가 중복되지 않도록 하십시오.
- 중첩(예: JMP0~JMP0~JME0~JME0)할 수 없습니다.
- JMP0/JME0 명령은 블록 프로그램 영역 내에서는 사용할 수 없습니다.
- 점프 대상 JME0 명령은 JMP0 명령이 존재하는 같은 태스크 내에서 작성하십시오. 태스크 간에는 점프를 실행할 수 없습니다.
- JMP0/JME0 명령 사이에서 미분 명령(DIFU/DIFD/@/% 표시가 있는 명령)을 사용한 경우, 입력 조건에 따라 동작이 안정되지 않는 경우가 있으므로 주의하십시오.

샘플 프로그램

0.00이 OFF이면 JMP0~JME0 사이의 명령을 NOP 처리합니다. (따라서 출력은 유지됩니다.)

0.00이 ON이면 JMP0~JME0 사이의 명령을 평상시대로 실행합니다.



FOR/NEXT

명령 이름	니모닉	동작 옵션	평션 번호	기능
반복 시작	FOR	−	512	FOR~NEXT 사이의 프로그램을 무조건 지정된 횟수만큼 반복 실행한 후 NEXT 명령 이후의 명령을 실행합니다.
반복 종료	NEXT	−	513	FOR-NEXT 루프의 종료를 나타냅니다.

심볼	FOR	NEXT
	FOR N N: 루프 횟수	NEXT

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	루프 횟수	UINT	1

N: 루프 횟수

10진수 &0~65535 또는 16진수 #0000~FFFF

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	○	○	○	○	○	○	○	○	○	○	○	○	−	○	−	−	−	−

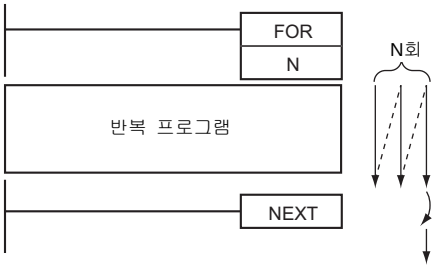
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 최대 중첩 레벨(15)을 초과하여 실행했을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	OFF
부정 플래그	P_N	OFF

기능

FOR~NEXT 사이의 프로그램을 무조건 N회 반복 실행한 후 NEXT 명령 이후의 명령을 실행합니다. 반복을 중단할 때는 BREAK 명령을 사용합니다. N에 0을 지정하면 FOR~NEXT 간의 명령을 NOP 처리합니다.

적은 프로그램 용량으로 테이블 데이터 등을 처리할 수 있습니다.



주: 1사이클에서 루프를 반복하므로 FOR-NEXT 명령 사이의 미분 접점은 해당 루프 중에 상시 ON 또는 상시 OFF가 됩니다.

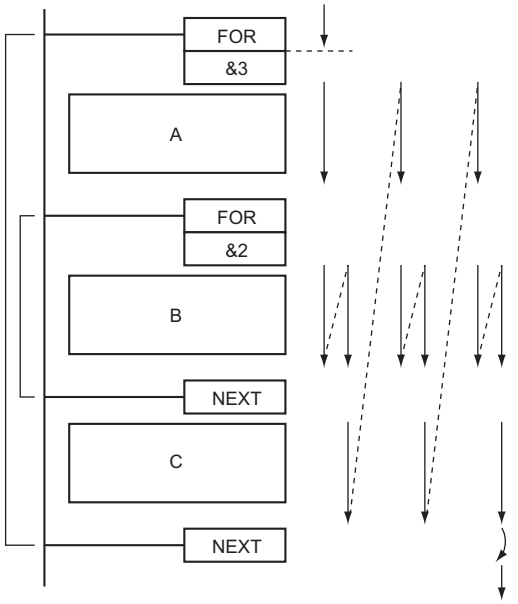
힌트

무조건이 아니라 특정 입력 조건이 성립할 때까지 프로그램을 반복하는 경우는 다음 2가지 방법이 있습니다.

1) 이 FOR/NEXT 명령을 사용하고 N에 최대 반복 횟수를 지정하여 반복 루프를 실행합니다. 반복을 N회 실행하기 전에 입력 조건이 성립하면 BREAK 명령을 실행하고 반복 루프를 강제 종료합니다.
2) 점프 명령(JME-JMP 명령 등)을 사용하고 JME 명령 뒤에 JMP 명령을 배치합니다. 특정 입력 조건이 성립할 때까지 JMP 명령에서 JME 명령(프로그램 주소가 빠른 명령)으로 점프를 반복합니다. 단, 이런 경우 입력 조건이 성립할 때까지 무제한 반복을 실행하므로 사이클 시간이 초과될 가능성이 있으므로 주의해야 합니다.

사용상의 주의 사항

- FOR 명령과 NEXT 명령은 동일 태스크 내에 기술하십시오. 서로 다른 태스크에 기술하면 반복이 실행되지 않습니다.
- FOR~NEXT의 중첩(예: FORn~FORm~NEXT~NEXT)은 최대 15개까지 가능합니다.

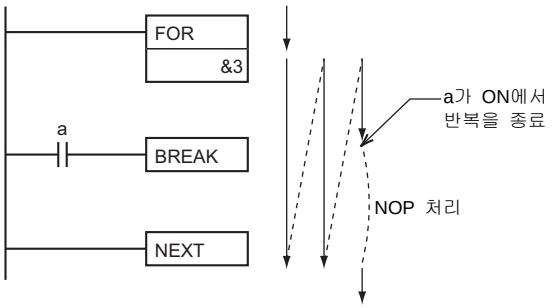


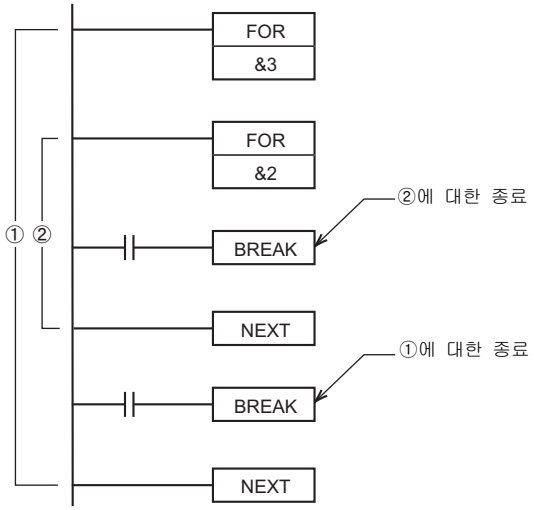
실행 순서: A→B→B→C⇢A→B→B→C⇢A→B→B→C

FOR/NEXT

· 반복 도중에 종료하는 경우는 반드시 BREAK 명령을 사용하십시오. 중첩되어 있는 경우, 해당 중첩 수만큼 BREAK 명령을 실행하십시오.

· BREAK 명령이 실행되면 해당 지점부터 NEXT 명령까지 NOP 처리됩니다.



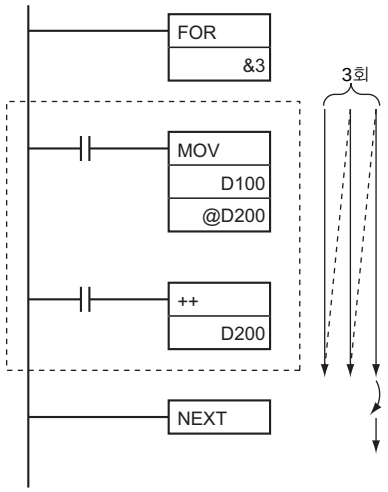


· FOR~NEXT 사이에서 점프 명령(JMP 등)을 실행하여 FOR~NEXT 밖으로 점프하지 마십시오.

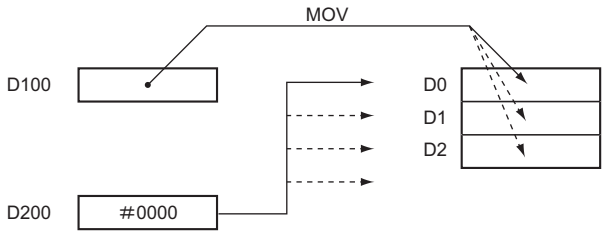
· FOR~NEXT 사이에서는 다음 명령을 사용할 수 없습니다.
  · 블록 프로그램
  · 복수 점프(JMP0)/복수 점프 종료(JME0)
  · 스텝 래더 진행(SNXT)/스텝 래더 영역 정의(STEP)

· FOR~NEXT 사이에서 미분 명령을 사용하는 경우, 해당 명령은 1회만 동작합니다.
  루프 횟수만큼 동작하지 않으므로 주의하십시오.
  · UP 명령, DOWN 명령
  · DIFU 명령, DIFD 명령
  · 기동 미분형 명령(동작 옵션 '@'를 포함하는 명령)
  · 종료 미분형 명령(동작 옵션 '%'를 포함하는 명령)

샘플 프로그램

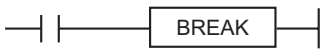


D200의 내용으로 간접 지정된 주소에 D100의 내용을 전송합니다.
++(증가) 명령으로 D200의 내용을 1 증가
이것을 3회 반복합니다.



BREAK

명령 이름	니모닉	동작 옵션	펑션 번호	기능
루프 중단	BREAK	−	514	FOR-NEXT 루프를 중단하고 NEXT 명령까지 NOP 처리합니다.

심볼	BREAK
	

특정 영역 내의 사용 여부

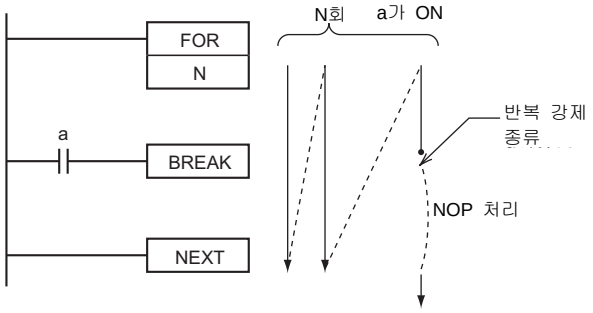
영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	가능	가능

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	OFF
부정 플래그	P_N	OFF

기능

FOR~NEXT 명령 사이의 프로그램 내에 배치합니다. 입력 조건이 ON일 때 FOR-NEXT 루프(반복 처리)를 강제 종료하고 NEXT 명령까지 NOP 처리합니다.



사용상의 주의 사항

- BREAK 명령은 하나의 중첩에 대해서만 사용할 수 있습니다. 다중 중첩을 종료하는 경우는 중첩 수만큼 BREAK 명령을 실행하십시오.
- BREAK 명령은 FOR~NEXT 명령 사이에서만 사용할 수 있습니다.

타이머/카운터 명령

니모닉	명령 이름	펑션 번호	페이지
TIM/TIMX	100ms 타이머	550	3-88
TIMH	10ms 타이머(고속 타이머)	015	3-91
TIMHX		551	
TMHH	1ms 타이머(초고속 타이머)	540	3-94
TMHHX		552	
TIMU	0.1ms 타이머	541	3-96
TIMUX		556	
TMUH	0.01ms 타이머	544	3-98
TMUHX		557	
TTIM	적산 타이머	087	3-100
TTIMX		555	
TIML	장시간 타이머	542	3-103
TIMLX		553	
MTIM	멀티 출력 타이머	543	3-106
MTIMX		554	
CNT/CNTX	카운터	546	3-109
CNTR	가역 카운터	012	3-112
CNTRX		548	
CNR	타이머/카운터 리셋	545	3-115
CNRX		547	
TRSET	타이머 리셋	549	3-117

타이머계 명령

타이머/카운터의 현재값 갱신 방식

■ 개요

타이머/카운터 관련 명령의 현재값 갱신 방식에는 'BCD 방식(모드)'과 'BIN 방식(모드)'이 있습니다.

방식	내용	설정 범위	설정값
BCD 방식	타이머 설정값을 BCD로 설정하는 방식입니다.	0~9,999	#0000~9999
BIN 방식	타이머 설정값을 BIN으로 설정하는 방식입니다.	0~65,535	&0~65535 또는 #0000~FFFF

타이머/카운터의 설정값을 채널(간접) 지정할 때도 이 타이머/카운터의 현재값 갱신 방식은 유효합니다. 간접 지정 대상 값을 BCD로 볼 것인가/BIN으로 볼 것인가에 대해서도 'BCD 방식(모드)'/'BIN 방식(모드)'이 유효합니다.

■ 대상 명령어

명령 분류	명령 이름	니모닉	
		BCD 방식인 경우	BIN 방식인 경우
타이머/카운터 명령	100ms 타이머(타이머)	TIM	TIMX(550)
	10ms 타이머(고속 타이머)	TIMH(015)	TIMHX(551)
	1ms 타이머(초고속 타이머)	TMHH(540)	TMHHX(552)
	0.1ms 타이머	TIMU(541)	TIMUX(556)
	0.01ms 타이머	TMUH(544)	TMUHX(557)
	적산 타이머(100ms)	TTIM(087)	TTIMX(555)
	장시간 타이머(100ms)	TIML(542)	TIMLX(553)
	멀티 출력 타이머(100ms)	MTIM(543)	MTIMX(554)
	카운터	CNT	CNTX(546)
	가역 카운터	CNTR(012)	CNTRX(548)
	타이머/카운터 리셋	CNR(545)	CNRX(547)
블록 프로그램 명령	100ms 타이머 대기(타이머 대기)	TIMW(813)	TIMWX(816)
	10ms 타이머 대기(고속 타이머 대기)	TMHW(815)	TMHWX(817)
	카운터 대기	CNTW(814)	CNTWX(818)

■ 현재값 갱신 설정 방법

· CS1-H/CS1D/CJ1-H/CJ1M CPU 유니트

하나의 프로젝트에서 BIN/BCD 중 하나를 선택해야 합니다. CX-Programmer의 PLC 속성에서 '타이머/카운터를 바이너리 모드로 실행'을 사용하여 방식을 선택하십시오.

· CJ2 CPU 유니트

하나의 프로젝트에서 BIN/BCD를 혼합하여 사용할 수 있습니다. CX-Programmer의 PLC 속성에서 갱신 방식을 지정할 필요가 없습니다.

기본 기능 목록

명령 이름	명령어	갱신	단위	최대 설정값	타이머 점수/명령	타이머 번호	타임업 플래그 갱신 타이밍	타이머 현재값 갱신 타이밍 (주)	리셋할 때	
									업 플래그	현재값
100ms 타이머	TIM	감산	100ms	999.9초	1점	사용	명령 실행 시	명령 실행 시 모든 사이클릭 태스크 종료 시 사이클 시간 80ms를 초과하는 경우 80ms마다 갱신	OFF	설정값
	TIMX			6553.5초						
10ms 타이머	TIMH	감산	10ms	99.99초	1점	사용	명령 실행 시	명령 실행 시 모든 사이클릭 태스크 종료 시 10ms마다 갱신	OFF	설정값
	TIMHX			655.35초						
1ms 타이머	TMHH	감산	1ms	9.999초	1점	사용	명령 실행 시	명령 실행 시 1ms마다 갱신	OFF	설정값
	TMHHX			65.535초						
0.1ms 타이머 *1	TIMU	감산	0.1ms	0.9999초	1점	사용	명령 실행 시	—	OFF	–
	TIMUX			6.5535초						

명령 이름	명령어	갱신	단위	최대 설정값	타이머 점수/명령	타이머 번호	타임업 플래그 갱신 타이밍	타이머 현재값 갱신 타이밍 (주)	리셋할 때	
									업 플래그	현재값
0.01ms 타이머 *1	TMUH	감산	0.01ms	0.09999초	1점	사용	명령 실행 시	—	OFF	–
	TMUHX			0.65535초						
적산 타이머	TTIM	누적	100ms	999.9초	1점	사용	명령 실행 시	명령 실행 시에만 갱신	OFF	0
	TTIMX			6553.5초						
장시간 타이머	TIML	감산	100ms	115일	1점	사용 안 함	명령 실행 시	명령 실행 시에만 갱신	OFF	설정값
	TIMLX			49710일						
멀티 출력 타이머	MTIM	누적	100ms	999.9초	8점	사용 안 함	명령 실행 시	명령 실행 시에만 갱신	OFF	0
	MTIMX			6553.5초						

주: 타이머 번호에 따라 갱신 타이밍이 달라집니다. 자세한 내용은 명령어의 설명을 참조하십시오.

각 조건의 동작 목록

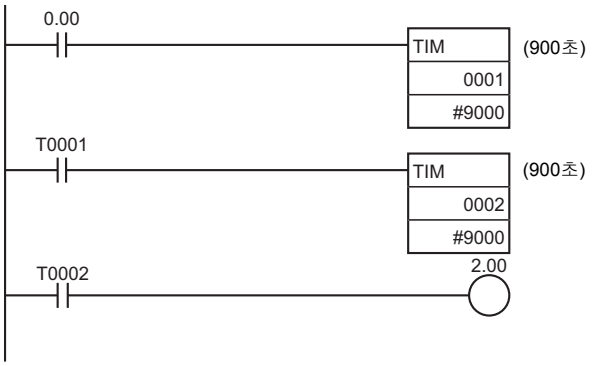
명령 이름	명령어	각 조건					강제 세트 시		강제 리셋 시	
		동작 모드 변경 시	전원 OFF 복구 시	CNR/CNRX 명령 실행 시	JMP-JME 명령에 의한 점프 시	IL-ILC 명령에 의한 인터록 시	업 플래그	현재값	업 플래그	현재값
100ms 타이머	TIM	현재값=0 업 플래그=OFF	현재값=0 업 플래그=OFF	현재값 9999 업 플래그=OFF	기동 중에 갱신	리셋(현재값=설정값, 업 플래그=OFF)	ON	0	OFF	설정값
	TIMX			현재값 FFFF 업 플래그=OFF						
10ms 타이머	TIMH	현재값=0 업 플래그=OFF	현재값=0 업 플래그=OFF	현재값 9999 업 플래그=OFF	기동 중에 갱신	리셋(현재값=설정값, 업 플래그=OFF)	ON	0	OFF	설정값
	TIMHX			현재값 FFFF 업 플래그=OFF						
1ms 타이머	TMHH	현재값=0 업 플래그=OFF	현재값=0 업 플래그=OFF	현재값 9999 업 플래그=OFF	기동 중에 갱신	리셋(현재값=설정값, 업 플래그=OFF)	ON	0	OFF	설정값
	TMHHX			현재값 FFFF 업 플래그=OFF						
0.1ms 타이머 *1	TIMU	현재값=0 업 플래그=OFF	현재값=0 업 플래그=OFF	현재값 9999 업 플래그=OFF	기동 중에 갱신	리셋(현재값=설정값, 업 플래그=OFF)	ON	– *2	OFF	– *2
	TIMUX			현재값 FFFF 업 플래그=OFF						
0.01ms 타이머 *1	TMUH	현재값=0 업 플래그=OFF	현재값=0 업 플래그=OFF	현재값 9999 업 플래그=OFF	기동 중에 갱신	리셋(현재값=설정값, 업 플래그=OFF)	ON	– *2	OFF	– *2
	TMUHX			현재값 FFFF 업 플래그=OFF						
적산 타이머	TTIM	현재값=0 업 플래그=OFF	현재값=0 업 플래그=OFF	현재값 9999 업 플래그=OFF	유지	유지	ON	0	OFF	0
	TTIMX			현재값 FFFF 업 플래그=OFF						
장시간 타이머	TIML	–	–	관계 없음	유지	리셋(현재값=설정값, 업 플래그=OFF)	–	–	–	–
	TIMLX									
멀티 출력 타이머	MTIM	–	–	관계 없음	유지	유지	–	–	–	–
	MTIMX									

*1: 0.1ms 타이머, 0.01ms 타이머에서는 타이머 현재값을 참조할 수 없습니다.

타이머/카운터 명령을 사용한 프로그램 예

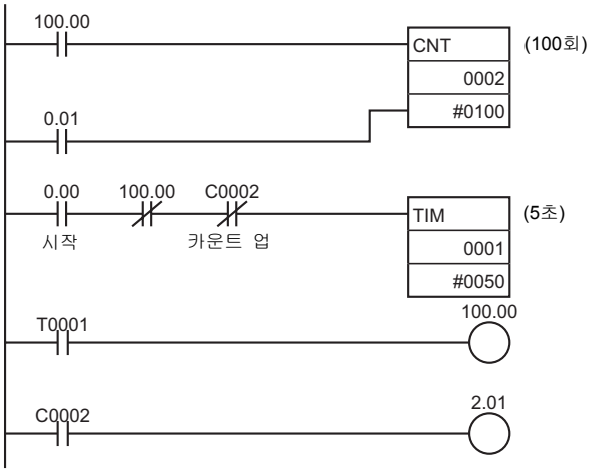
(1) 장시간 타이머

① TIM + TIM(예: 30분)



명령	데이터
LD	0.00
TIM	1
	#9000
LD	T0001
TIM	2
	#9000
LD	T0002
OUT	2.00

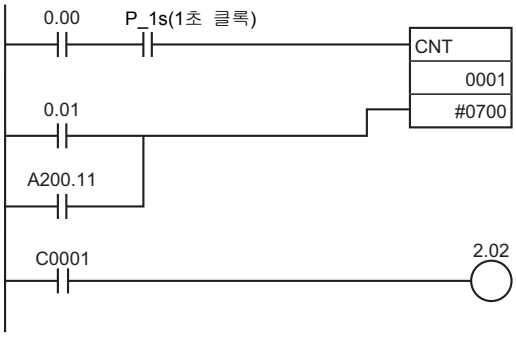
② TIM + CNT(예: 500초)



명령	데이터
LD	100.00
LD	0.01
CNT	2
	#0100
LD	0.00
AND NOT	100.00
ANT NOT	C0002
TIM	1
	#0050
LD	T0001
OUT	100.00
LD	C0002
OUT	2.01

- TIM1에서 5초마다 펄스를 발생시키고 CNT2에서 5초 간격의 펄스를 계산합니다.
- 타이머 시간은 (타이머 + 사이클 시간)×카운트 수가 됩니다. 이러한 예로 500초 타이머를 들 수 있습니다.
- 이 경우 카운터의 현재값은 전원이 꺼져도 데이터가 유지됩니다.

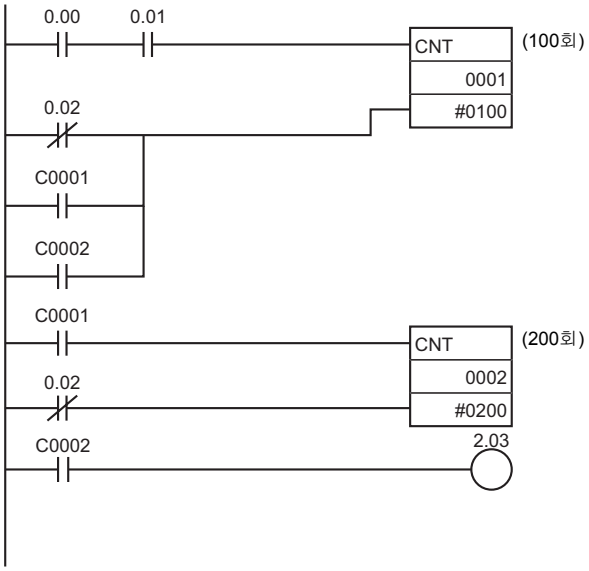
③ 클록 펄스 + 카운터(예: 700초)



명령	데이터
LD	0.00
AND	1s
LD NOT	0.01
CNT	1
	#0700
LD	C0001
OUT	2.02

- 내부 클록 펄스와 카운터를 조합하여 장시간 타이머를 구성할 수 있습니다.
- CNT1의 리셋 입력에 특수 보조 릴레이 A200.11(사이클 시간 ON)을 OR하면 프로그램 실행 시작 시, 설정값(0700)부터 다시 계산합니다.

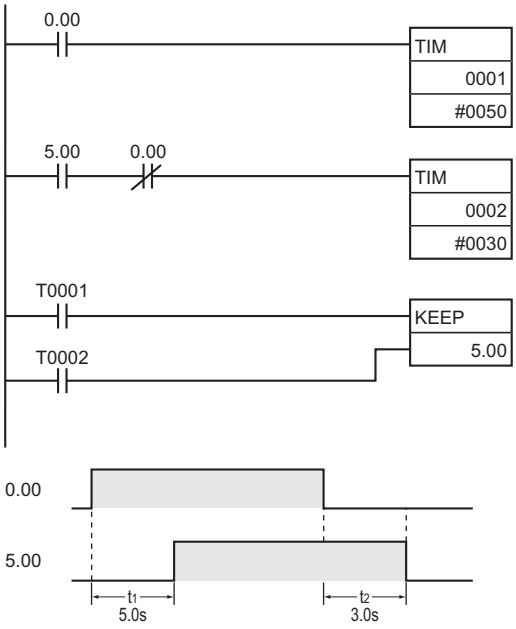
(2) 여러 자리 카운터(예: 20,000회)



명령	데이터
LD	0.00
AND	0.01
LD NOT	0.02
OR	C0001
OR	C0002
CNT	1
	#0100
LD	C0001
LD NOT	0.02
CNT	2
	#0200
LD	C0002
OUT	2.03

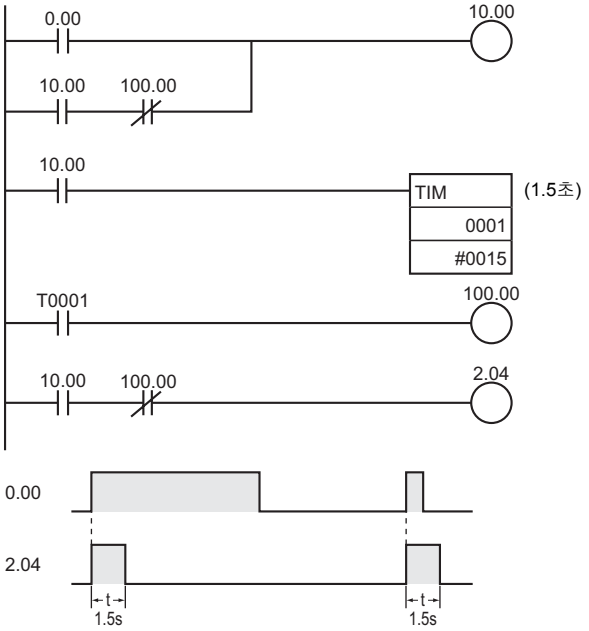
- 9999회 이상을 계산할 때는 카운터를 다단계로 프로그래밍 하면 됩니다.

(3) ON/OFF 딜레이 회로



명령	데이터
LD	0.00
TIM	1
	#0050
LD	5.00
AND NOT	0.00
TIM	2
	#0030
LD	T0001
LD	T0002
KEEP(011)	5.00

(4) 단일 쇼트 회로

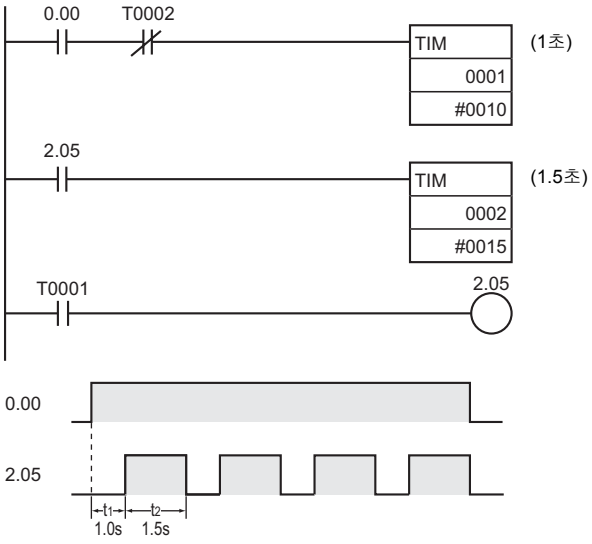


명령	데이터
LD	0.00
LD	10.00
AND NOT	100.00
OR LD	–
OUT	10.00
LD	10.00
TIM	1
	#0015
LD	T0001
OUT	100.00
LD	10.00
AND NOT	100.00
OUT	2.04

· 단일 쇼트는 입력이 ON된 후 TIM1의 설정 시간만큼 출력합니다.

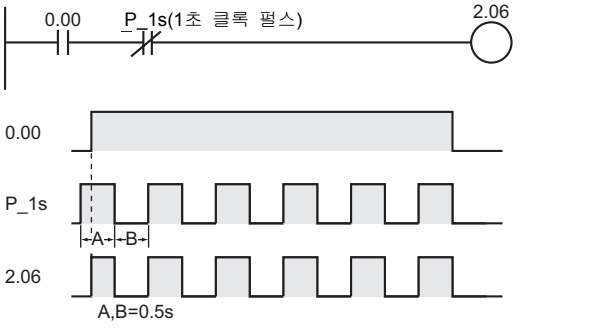
(5) 플리커 회로

① TIM 2개 사용 예



명령	데이터
LD	0.00
AND NOT	T0002
TIM	1
	#0010
LD	2.05
TIM	2
	#0015
LD	T0001
OUT	2.05

② 클록 펄스 사용 예



명령	데이터
LD	0.00
AND	1s
OUT	2.06

· 내부 클록 펄스(0.1초, 0.2초, 1초)를 이용하면 플리커 회로를 손쉽게 프로그래밍할 수 있습니다.

클록 펄스에 대해서는 "CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편"(SBCA-350)을 참조하십시오.

타이머/카운터 번호 간접 지정

- 타이머/카운터 번호는 인덱스 레지스터를 사용하여 간접적으로 사용할 수 있습니다.

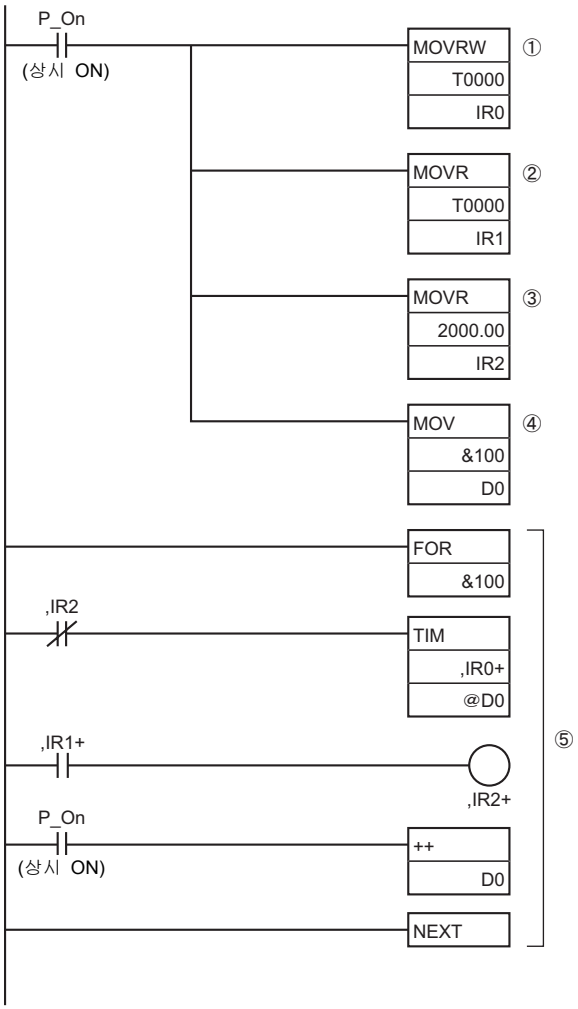
  인덱스 레지스터를 사용하여 간접 지정하는 경우, MOVRW(타이머/카운터 현재값용 인덱스 레지스터 설정) 명령을 사용하여 인덱스 레지스터에 지정된 번호에 해당하는 타이머/카운터의 현재값 영역의 I/O 메모리 실제 주소를 설정합니다.

  인덱스 레지스터를 사용한 타이머/카운터 번호 간접 지정은 타이머/카운터 번호를 가진 TIM/TIMX, TIMH/TIMHX, TMHH/TMHHX, TIMU/TIMUX, TMUH/TMUHX, TTIM/TTIMX, CNT/CNTX, CNTR/CNTRX, TIMW/TIMWX, CNTW/CNTWX, TMHW/TMHWX 명령에서 사용할 수 있습니다.
- 인덱스 레지스터를 사용하여 지정한 I/O 메모리 실제 주소가 잘못된 주소인 경우 해당 타이머/카운터 명령은 실행되지 않습니다.
- 인덱스 레지스터에 의한 타이머/카운터 번호의 간접 지정을 사용하면 다음 사용 예와 같이 프로그램 용량을 절감하거나 공통 서브루틴을 작성할 수 있습니다.

■ 사용 예: 복수의 TIM 명령을 간접 지정하여 기동하는 경우

D100	0010	…TIM0000의 설정값
D101	0100	…TIM0001의 설정값
D102	0050	…TIM0002의 설정값
⋮	⋮	⋮
D199	0999	…TIM0099의 설정값

- 왼쪽 예에서는T0000~T0099의 TIM 명령을 인덱스 레지스터 간접 및 간접 DM 지정을 사용하여 기동시킵니다. 각 TIM 명령의 설정값은 테이블 데이터로 D100~D199에 저장되어 있습니다.



① IR0에 TIM0000의 현재값 영역의 메모리 주소를 설정하고, 간접 지정을 통한 TIM 번호로 사용합니다. MOVRW 명령을 사용하는 현재값 영역 주소가 설정됩니다.

② IR1에 TIM0000의 플래그 주소를 설정합니다. MOVR 명령을 사용하면 플래그 주소가 설정됩니다.

③ IR2에 내부 보조 릴레이 2000.00의 플래그 주소를 설정합니다.

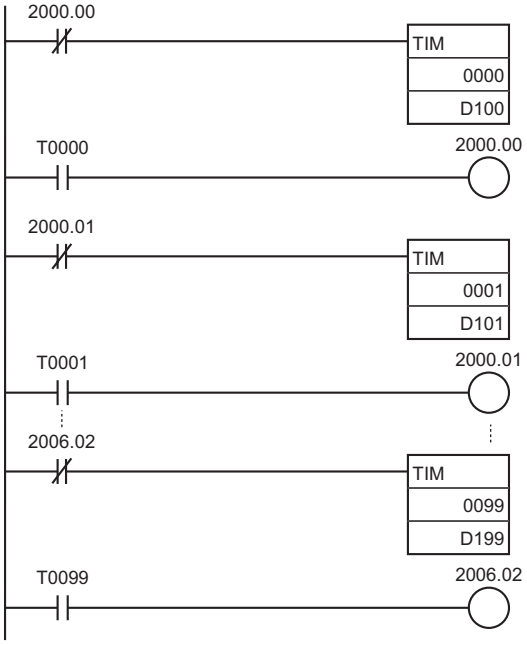
④ D0에 D100의 간접 지정을 위한 &100을 설정합니다.

⑤ IR0, IR1, IR2, D0의 값을 1씩 증가시키면서 TIM 명령의 실효를 100회 반복하고, TIM0000~TIM0099를 기동합니다.

위의 프로그램 중 ⑤ 부분만 살펴보면,

IR0	기동시키는 TIM의 현재값 메모리 주소
IR1	기동시키는 TIM의 플래그 주소
IR2	TIM 명령을 설정하기 위해 사용하는 내부 보조 릴레이의 메모리 주소
D0	기동시키는 TIM의 설정 저장 영역 DM 번호를 입력 파라미터로 한 100개의 TIM을 기동하는 공통 서브루틴으로 사용할 수 있습니다.

또한 위의 회로는 다음 회로와 등가가 됩니다.



■ 타이머 리셋 방법

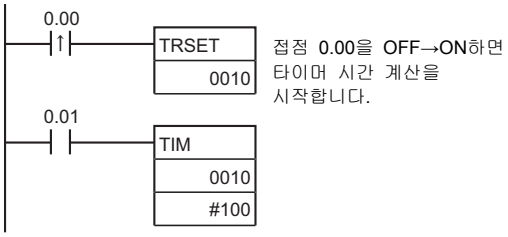
타이머 명령을 리셋하는 방법은 다음 3가지가 있습니다.

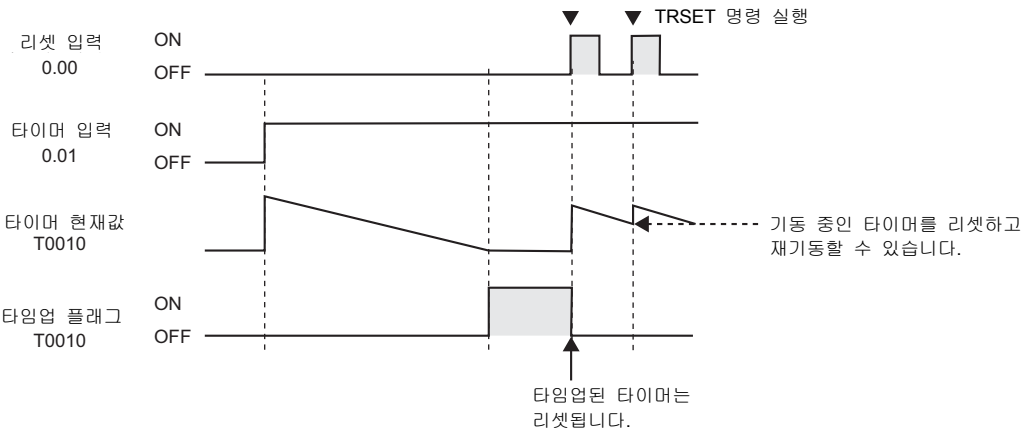
1. 타이머 리셋 명령 사용

타이머 리셋 명령(TRSET)을 실행할 때 지정된 타이머를 리셋합니다.

이 명령이 지정할 수 있는 타이머는 1점이지만 타이머/카운터 리셋 명령(CNR)보다 고속으로 처리를 수행합니다.

이미 시간 계산 중인 타이머에 대해 1사이클 내에서 타이머의 리셋→재기동을 실행할 때는 이 명령을 사용합니다.



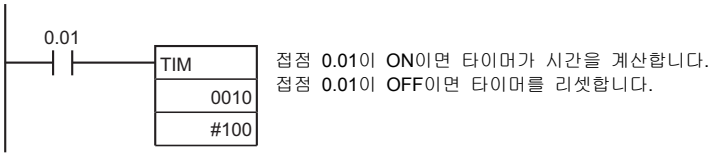


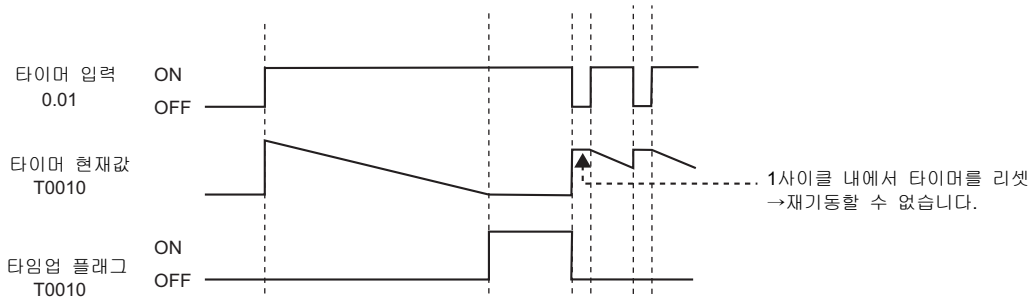
2. 타이머 명령의 입력을 OFF

타이머 명령의 입력을 OFF하면 타이머가 리셋됩니다.

타이머 명령의 입력을 다시 ON하면 타이머가 시간 계산을 실행합니다.

1사이클 내에서 타이머의 리셋→재기동을 수행할 수는 없습니다.



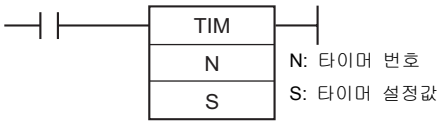
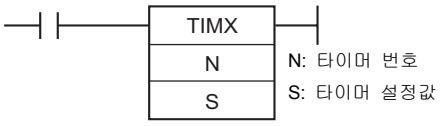


3. 타이머/카운터 리셋 명령 사용

타이머/카운터 리셋 명령(CNR/CNRX)을 실행할 때 지정된 타이머를 리셋합니다.
이 명령은 복수의 타이머를 지정할 수 있지만 타이머 리셋 명령(TRSET)보다 저속이 됩니다.
사용 방법은 타이머 리셋 명령(TRSET)과 동일합니다.

TIM/TIMX

명령 이름	니모닉	동작 옵션	평션 번호	기능
100ms 타이머	TIM/TIMX	–	550	감산식 ON 딜레이 0.1초 단위의 타이머 동작을 합니다.

심볼	TIM	TIMX
	BCD 방식일 때 	BIN 방식일 때 

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기
		TIM	TIMX	
N	타이머 번호	TIMER	TIMER	1
S	타이머 설정값	WORD	UINT	1

N: 타이머 번호
10진수 0~4095
S: 타이머 설정값(100ms 단위)
TIM(BCD): #0000~9999
TIMX(BIN): 10진수 &0~65535 또는 16진수 #0000~FFFF

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
N	–	–	–	–	○	–	–	–	–	–	–	–	–	○	–	–	–	–
S	○	○	○	○		○	○	○	○	○	○	○						

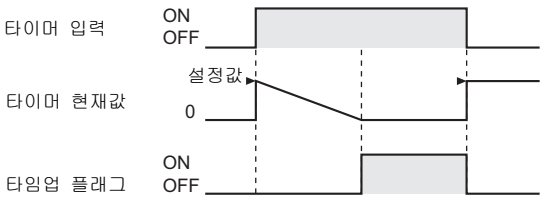
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· N의 타이머 번호가 간접 IR 지정인 경우, 주소가 타임업 플래그 또는 타이머의 현재값 주소가 아닐 때 ON · BCD 방식이고 S의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	변하지 않음*1
부정 플래그	P_N	변하지 않음*1

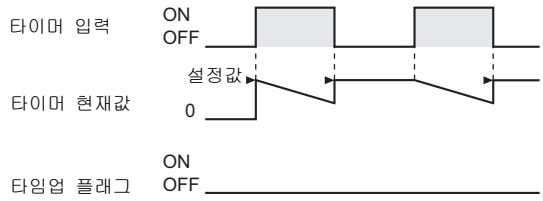
*1: CS1/CJ1/CS1D(이중화 시스템용) CPU 유니트는 OFF

기능

- 타이머 입력이 OFF일 때는 N으로 지정된 번호의 타이머를 리셋합니다. (타이머 현재값에 설정값 S를 대입하고 타임업 플래그를 OFF로 합니다.)
- 타이머 입력이 OFF→ON으로 된 시점에 타이머를 기동하고 타이머 현재값의 감산을 시작합니다. 타이머 입력이 ON인 동안, 타이머 현재값의 갱신을 실행하고 타이머 현재값이 0이 되면 타임업 플래그를 ON으로 합니다(타임업).
- 타임업된 후에는 타이머 현재값 및 타임업 플래그의 상태를 유지합니다. 재기동하려면 일단 타이머 입력을 OFF→ON으로 하거나 MOV 명령 등을 사용하여 타이머 현재값을 0 이외의 값으로 변경해야 합니다.
- 설정 시간은 다음과 같습니다.
  - TIM의 경우(BCD): 0~999.9초
  - TIMX의 경우(BIN): 0~6553.5초
- 타이머 정도는 −0.01~0초입니다.



(타임업되기 전에 타이머 입력이 OFF된 경우)



주: CS1D CPU 유니트에서는 타이머 정도가 ±(10ms + 사이클 시간)입니다.
CJ1-H-R CPU 유니트의 유니트 Ver.4.1의 타이머 정도는 −0.1~0초입니다.

힌트

TIM/TIMX 명령에서 사용되는 타이머 현재값 및 타임업 플래그는 다음의 타이밍에서 갱신됩니다.

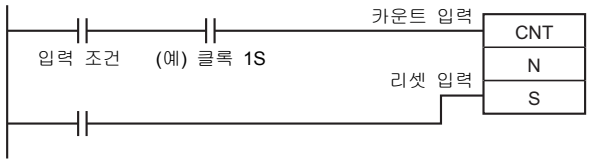
T0000~T2047의 경우

갱신 타이밍	내용
1) 각 명령 실행 시	· 명령을 실행할 때마다 현재값 갱신 · 현재값 = 0에서 타임업 플래그 ON, 0이 아니면 타임업 플래그 OFF
2) 모든 사이클릭 태스크 실행 종료 시	1사이클에 1회씩 현재값 갱신
3) 80ms마다 갱신	사이클 시간이 80ms를 초과하는 경우 80ms마다 현재값 갱신

T2048~T4095의 경우

갱신 타이밍	내용
각 명령 실행 시	· 명령을 실행할 때마다 현재값 갱신 · 현재값 = 0에서 타임업 플래그 ON, 0이 아니면 타임업 플래그 OFF

- 전원 OFF 시 타이머는 리셋(타이머 현재값=설정값, 타임업 플래그 OFF)됩니다. 전원 OFF 시에도 타이머 현재값을 유지하려면 타이머 명령을 사용하지 말고 다음과 같이 내부 클록과 카운터를 조합하십시오.



- 타이머 설정값이 #0000일 때는 명령 실행 시에 타임업됩니다.

사용상의 주의 사항

- 타이머 번호는 다른 타이머 명령과 공용입니다. 이들 명령에서 같은 타이머 번호를 동시에 동작시키면 오작동하므로 주의하십시오. 동시에 사용하면 프로그램 확인 시 '코일 이중 사용'이 표시됩니다. 또한 동시에 동작시키지 않는 조건이면 동일한 번호를 사용할 수 있습니다.
- 사이클 시간이 100ms를 초과하면 타이머 번호 2048~4095의 타이머는 정상적으로 동작하지 않습니다. 사이클 시간이 100ms를 초과할 때는 타이머 번호 0~2047의 타이머를 사용하십시오.
- 타이머 번호가 0~2047인 경우 타이머 명령은 태스크가 대기 중이어도 현재값을 갱신합니다. 타이머 번호가 2048~4095이면 태스크가 대기 중인 경우 현재값을 유지합니다.
- 타이머는 다음과 같은 경우에 리셋 또는 유지됩니다. (리셋할 때는 타이머 현재값 = 설정값, 타임업 플래그 = OFF가 됩니다.)

TIM/TIMX

	동작 모드 변경 시 (프로그램←→운전 또는 모니터) (주 1)	전원 OFF 복구시 (주 2)	CNR/CNRX(타이머/카운터 리셋 명령) 실행(주 3)	IL-ILC 명령에 의한 인터록 시	JMP-JME 명령에 의한 점프 시
현재값	0	0	BCD 방식일 때: #9999 BIN 방식일 때: #FFFF	설정값	현재값은 계속 갱신됨
타임업 플래그	OFF	OFF	OFF	OFF	그 이전 상태를 유지

주 1: IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정한 경우는 동작 모드 변경 시에도 유지

주 2: IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정하고, PLC 시스템 설정에서 '전원 ON 시 IOM(I/O 메모리) 유지 비트 유지/비유지 설정'을 유지하는 것으로 설정한 경우는 전원 OFF 시에도 유지

주 3: TIM/TIMX 명령을 실행할 때 설정값이 설정됩니다.

- IL-ILC 명령에 의해 인터록된 경우는 리셋되어 현재값 = 설정값이 되고 타임업 플래그 = OFF가 됩니다.
- JMP/CJMP/CJPN-JME 명령에 의해 점프한 경우에도 기동 중인 타이머 번호 0~2047의 타이머 현재값은 갱신됩니다.*1 (명령은 실행되지 않지만 모든 태스크 실행 종료 후에 현재값이 갱신됩니다.)

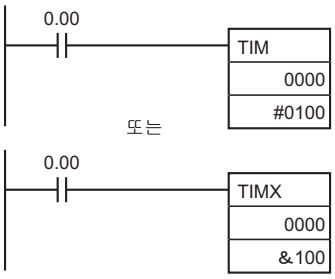
  *1: CS1D CPU 유니트에서는 갱신되지 않습니다.
- 강제 세트할 때는 업 플래그=ON, 현재값=0이 되고 강제 리셋할 때는 업 플래그=OFF, 현재값=설정값이 됩니다.
- 타임업 플래그는 명령 실행 시에만 갱신되기 때문에 타임업 플래그를 사용자 프로그램에서 전송하는 경우, 타이밍에 따라 1사이클 시간만큼 지연이 발생할 수 있으므로 주의하십시오.
- 온라인 편집에서 타이머 명령을 재기입할 때는 반드시 타임업 플래그를 리셋하십시오. 타임업 플래그를 리셋하지 않으면 재기입 후 타이머가 정상적으로 동작하지 않습니다.
- 유니트 간 동기 제어 기능을 사용할 때는 다음과 같은 제약이 있습니다.
  - 사이클 시간이 100ms를 초과하면 타이머가 정상적으로 동작하지 않습니다.
  - TIM/TIMX 명령이 정지한 태스크 내에 있거나 JMP/CJMP/CJPN-JME 명령으로 점프되어서 실행되지 않은 경우, 타이머 정도에 최대 -10ms의 오차가 발생합니다.
  - 타이머 현재값은 참조할 수 없는데, 참조하는 경우에는 잘못된 값을 읽어옵니다.

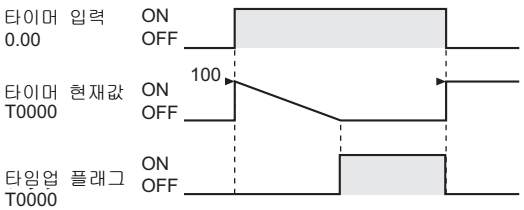
샘플 프로그램

타이머 입력 0.00이 OFF→ON으로 된 경우 타이머 현재값은 설정값에서 감산을 시작합니다.

타이머 현재값 = 0이 되면 타임업 플래그 T0000이 ON이 됩니다.

타이머 입력 0.00이 OFF가 되면, 타이머 현재값에 타이머 설정값이 다시 설정되고 타임업 플래그 T0000은 OFF가 됩니다.





TIMH/TIMHX

명령 이름	니모닉	동작 옵션	펑션 번호	기능
10ms 타이머	TIMH	–	015	감산식 ON 딜레이 10ms(0.01초) 단위의 타이머 동작을 합니다.
(고속 타이머)	TIMHX	–	551	

심볼	TIMH	TIMHX
	BCD 방식일 때	BIN 방식일 때
	TIMH N S N: 타이머 번호 S: 타이머 설정값	TIMHX N S N: 타이머 번호 S: 타이머 설정값

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기
		TIMH	TIMHX	
N	타이머 번호	TIMER	TIMER	1
S	타이머 설정값	WORD	UINT	1

N: 타이머 번호
10진수 0~4095
S: 타이머 설정값(10ms 단위)
TIMH(BCD): #0000~9999
TIMHX(BIN): 10진수 &0~65535 또는 16진수 #0000~FFFF

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
N	–	–	–	–	○	–	–	–	–	–	–	–	–	○	–	–	–	–
S	○	○	○	○		○	○	○	○	○	○	○						

조건 플래그의 동작

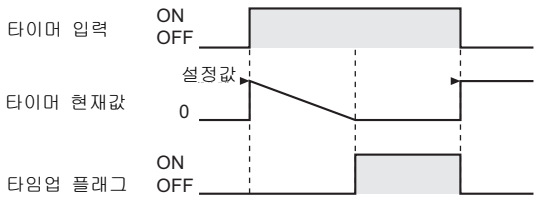
이름	라벨	내용
에러 플래그	P_ER	· N의 타이머 번호가 간접 IR 지정인 경우, 주소가 타임업 플래그 또는 타이머의 현재값 주소가 아닐 때 ON · BCD 방식이고 S의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	변하지 않음*1
부정	P_N	변하지 않음*1

*1: CS1/CJ1/CS1D(이중화 시스템용) CPU 유니트는 OFF

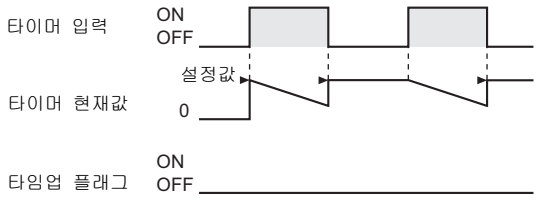
TIMH/TIMHX

기능

- 타이머 입력이 OFF일 때는 N으로 지정된 번호의 타이머를 리셋합니다(타이머 현재값에 설정값 S를 대입하고 타임업 플래그를 OFF로 합니다).
- 타이머 입력이 OFF→ON으로 된 시점에 타이머를 기동하고 타이머 현재값의 감산을 시작합니다. 타이머 입력이 ON인 동안, 타이머 현재값의 갱신을 실행하고 타이머 현재값이 0이 되면 타임업 플래그를 ON으로 합니다(타임업).
- 타임업된 후에는 타이머 현재값 및 타임업 플래그의 상태를 유지합니다. 재기동하려면 일단 타이머 입력을 OFF→ON으로 하거나 MOV 명령 등을 사용하여 타이머 현재값을 0 이외의 값으로 변경해야 합니다.
- 설정 시간은 다음과 같습니다.
  - BCD 방식인 경우 0~99.99초
  - BIN 방식인 경우 0~655.35초
- 타이머 정도는 -0.01~0초입니다. CS1D CPU 유니트에서는 타이머 정도가 ±(10ms+사이클 시간)입니다.



(타임업되기 전에 타이머 입력이 OFF된 경우)



힌트

TIMH/TIMHX 명령에서 사용되는 타이머 현재값 및 타임업 플래그는 다음의 타이밍에서 갱신됩니다.

T0000~T0255의 경우

갱신 타이밍	내용
1) 각 명령 실행 시	현재값=0에서 타임업 플래그 ON, 0이 아니면 타임업 플래그 OFF
2) 10ms마다 갱신	10ms마다 현재값 갱신

T0256~T2047의 경우

갱신 타이밍	내용
1) 각 명령 실행 시	· 명령을 실행할 때마다 현재값 갱신 · 현재값=0에서 타임업 플래그 ON, 0이 아니면 타임업 플래그 OFF
2) 모든 사이클릭 태스크 실행 종료 시	1사이클에 1회씩 현재값 갱신
3) 80ms마다 갱신	사이클 시간이 80ms를 초과하는 경우 80ms마다 현재값 갱신

T2048~T4095의 경우

갱신 타이밍	내용
각 명령 실행 시	· 명령을 실행할 때마다 현재값 갱신 · 현재값=0에서 타임업 플래그 ON, 0이 아니면 타임업 플래그 OFF

사용상의 주의 사항

- 타이머 번호는 다른 타이머 명령과 공용입니다. 이들 명령에서 같은 타이머 번호를 동시에 동작시키면 오작동하므로 주의하십시오. 동시에 사용하면 프로그램 확인 시 '코일 이중 사용'이 표시됩니다. 또한 동시에 동작시키지 않는 조건이면 동일한 번호를 사용할 수 있습니다.
- 사이클 시간이 100ms를 초과하면 타이머 번호 2048~4095의 타이머는 정상적으로 동작하지 않습니다. 사이클 시간이 100ms를 초과할 때는 타이머 번호 0~2047의 타이머를 사용하십시오.
- TIMH/TIMHX 명령은 타이머 번호 0~255에 대해서만 10ms마다 현재값을 갱신합니다. 사용자 프로그램에서 타이머 현재값을 참조하고 싶을 때는 타이머 번호 0~255를 사용하십시오.
- 타이머 번호가 0~2047인 경우 10ms 타이머 명령은 태스크가 대기 중이어도 현재값을 갱신합니다. 타이머 번호가 2048~4095이면 태스크가 대기 중인 경우 현재값을 유지합니다.
- CVM1/CV 시리즈와 달리, CS/CJ 시리즈에서는 타임업 플래그가 TIMH/TIMHX 명령을 실행할 때 갱신됩니다.

· 타이머는 다음과 같은 경우에 리셋 또는 유지됩니다. (리셋할 때는 타이머 현재값 = 설정값, 타임업 플래그 = OFF가 됩니다.)

	동작 모드 변경 시 (프로그램←→운전 또는 모니터) (주 1)	전원 OFF 복구 시 (주 2)	CNR/CNRX(타이머/카운터 리셋 명령) 실행 (주 3)	IL-ILC 명령에 의한 인터록 시	JMP-JME 명령에 의한 점프 시
현재값	0	0	BCD 방식일 때: #9999 BIN 방식일 때: #FFFF	설정값	현재값은 계속 갱신됨
타임업 플래그	OFF	OFF	OFF	OFF	그 이전 상태를 유지

주 1: IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정한 경우는 동작 모드 변경 시에도 유지

주 2: IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정하고, PLC 시스템 설정에서 '전원 ON 시 IOM(I/O 메모리) 유지 비트 유지/비유지 설정'을 유지하는 것으로 설정한 경우는 전원 OFF 시에도 유지

주 3: TIMH/TIMHX 명령을 실행할 때 설정값이 설정됩니다.

· JMP/CJMP/CJPN-JME 명령에 의해 점프한 경우에도 기동 중인 타이머 번호 0~2047의 타이머 현재값은 갱신됩니다.*1 (명령은 실행되지 않지만 10ms마다 또는 모든 태스크 실행 종료 후에 현재값이 갱신됩니다.)

*1: CS1D CPU 유니트에서는 갱신되지 않습니다.

· IL-ILC 명령에 의해 인터록된 경우는 리셋되어 현재값 = 설정값이 되고 타임업 플래그 = OFF가 됩니다.

· 강제 세트할 때는 업 플래그=ON, 현재값=0이 되고 강제 리셋할 때는 업 플래그=OFF, 현재값=설정값이 됩니다.

· 타임업 플래그는 명령 실행 시에만 갱신되기 때문에 타임업 플래그를 사용자 프로그램에서 전송하는 경우, 타이밍에 따라 1사이클 시간만큼 지연이 발생할 수 있으므로 주의하십시오.

· 온라인 편집에서 타이머 명령을 재기입할 때는 반드시 타임업 플래그를 리셋하십시오. 타임업 플래그를 리셋하지 않으면 재기입 후 타이머가 정상적으로 동작하지 않습니다.

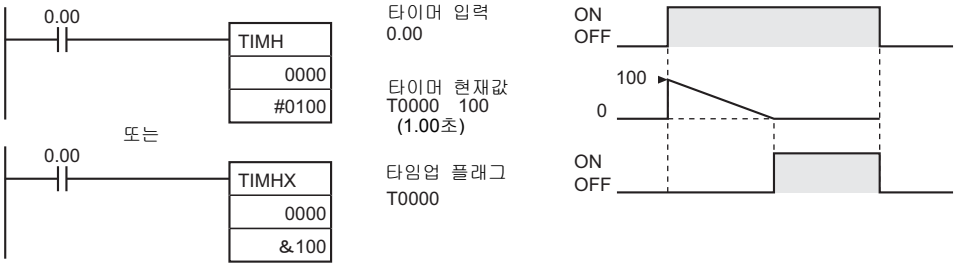
· 유니트 간 동기 제어 기능을 사용할 때는 다음과 같은 제약이 있습니다.
  · 사이클 시간이 100ms를 초과하면 타이머가 정상적으로 동작하지 않습니다.
  · TIMH/TIMHX 명령이 정지한 태스크 내에 있거나 JMP/CJMP/CJPN-JME 명령에서 점프되어서 실행되지 않은 경우, 타이머 정도에 최대 -10ms의 오차가 발생합니다.
  · 타이머 현재값은 참조할 수 없는데, 참조하는 경우에는 잘못된 값을 읽어옵니다.

샘플 프로그램

타이머 입력 0.00이 OFF→ON으로 된 경우 타이머 현재값은 설정값에서 감산을 시작합니다.

1초 후 타이머 현재값 = 0이 되면 타임업 플래그 T0000이 ON이 됩니다.

타이머 입력 0.00이 OFF가 되면, 타이머 현재값에 타이머 설정값이 다시 설정되고 타임업 플래그 T0000은 OFF가 됩니다.



TMHH/TMHHX

명령 이름	니모닉	동작 옵션	평션 번호	기능
1ms 타이머 (초고속 타이머)	TMHH	–	540	감산식 고속 ON 딜레이 1ms 단위의 타이머의 동작을 나타냅니다.
	TMHHX	–	552	

심볼	TMHH	TMHHX
	BCD 방식일 때 TMHH N: 타이머 번호 S: 타이머 설정값	BIN 방식일 때 TMHHX N: 타이머 번호 S: 타이머 설정값

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능*1	불가능	가능	가능	불가능	불가능*1

*1: CJ1-H-R/CJ2 CPU 유니트는 '가능'

피연산자 설명

피연산자	내용	데이터 형식		크기
		TMHH	TMHHX	
N	타이머 번호	TIMER	TIMER	1
S	타이머 설정값	WORD	UINT	1

N: 타이머 번호
10진수 0~15 또는 0~4095*1

S: 타이머 설정값
BCD: #0000~9999
BIN: 10진수 &0~65535 또는 16진수 #0000~FFFF

*1: CJ1-H-R/CJ2 CPU 유니트는 0~4095

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	○	–	–	–	–	–	–	–	–	○	–	–	–	–
S	○	○	○	○		○	○	○	○	○	○	○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· N의 타이머 번호가 간접 IR 지정인 경우, 주소가 타임업 플래그 또는 타이머의 현재값 주소가 아닐 때 ON · BCD 방식이고 S의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	변하지 않음*1
부정 플래그	P_N	변하지 않음*1

*1: CS1/CJ1/CS1D(이중화 시스템용) CPU 유니트는 OFF

기능

- 타이머 입력이 OFF일 때는 N으로 지정된 번호의 타이머를 리셋합니다. (타이머 현재값에 설정값 S를 대입하고 타임업 플래그를 OFF로 합니다.)
- 타이머 입력이 OFF→ON으로 된 시점에 타이머를 기동하고 타이머 현재값의 감산을 시작합니다. 타이머 입력이 ON인 동안, 타이머 현재값의 갱신을 실행하고 타이머 현재값이 0이 되면 타임업 플래그를 ON으로 합니다(타임업).
- 타임업된 후에는 타이머 현재값 및 타임업 플래그의 상태를 유지합니다. 재기동하려면 일단 타이머 입력을 OFF→ON으로 하거나 MOV 명령 등을 사용하여 타이머 현재값을 0 이외의 값으로 변경해야 합니다.
- 설정 시간은 다음과 같습니다.
  - BCD 방식인 경우 0~9.999초
  - BIN 방식인 경우 0~65.535초

주: 타이머 정도는 -0.001~0초입니다. CS1D CPU 유니트에서는 타이머 정도가 ±(10ms + 사이클 시간)입니다.
CJ1-H-R CPU 유니트의 유니트 Ver.4.1의 타이머 정도는 -0.01~0초입니다.

힌트

TMHH/TMHHX 명령에서 사용되는 타이머 현재값 및 타임업 플래그는 다음의 타이밍에서 갱신됩니다.

T0000~T0015의 경우*1

갱신 타이밍	내용
1) 각 명령 실행 시	현재값 = 0에서 타임업 플래그 ON, 0이 아니면 타임업 플래그 OFF
2) 1ms마다 갱신	1ms마다 현재값 갱신

*1: CJ-H-R CPU 유니트의 유니트 Ver.4.1에서는 사용할 수 없습니다.

T0016~T4095의 경우*2

갱신 타이밍	내용
각 명령 실행 시	현재값 = 0에서 타임업 플래그 ON, 0이 아니면 타임업 플래그 OFF

*2: CJ1-H-R/CJ2 CPU 유니트만 해당

사용상의 주의 사항

- 타이머 번호는 다른 타이머 명령과 공용입니다. 이들 명령에서 같은 타이머 번호를 동시에 동작시키면 오작동하므로 주의하십시오. 동시에 사용하면 프로그램 확인 시 '코일 이중 사용'이 표시됩니다. 또한 동시에 동작시키지 않는 조건이면 동일한 번호를 사용할 수 있습니다.
- 타임업 플래그는 명령 실행 시에만 갱신되기 때문에 타임업 플래그를 사용자 프로그램에서 전송하는 경우, 타이밍에 따라 1사이클만큼 지연이 발생할 수 있으므로 주의하십시오.
- TMHH/TMHHX 명령은 태스크가 대기 중이어도 현재값을 갱신합니다.
- 타이머는 다음과 같은 경우에 리셋 또는 유지됩니다. (리셋할 때는 타이머 현재값 = 설정값, 타임업 플래그 = OFF가 됩니다.)

	동작 모드 변경 시 (프로그램←→운전 또는 모니터) (주 1)	전원 OFF 복구 시 (주 2)	CNR/CNRX(타이머/카운터 리셋 명령) 실행(주 3)	IL-ILC 명령에 의한 인터록 시	JMP-JME 명령에 의한 점프 시
현재값	0	0	BCD 방식일 때: #9999 BIN 방식일 때: #FFFF	설정값	현재값은 계속 갱신됨
타임업 플래그	OFF	OFF	OFF	OFF	그 이전 상태를 유지

주 1: IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정한 경우는 동작 모드 변경 시에도 유지
주 2: IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정하고, PLC 시스템 설정에서 '전원 ON 시 IOM(I/O 메모리) 유지 비트 유지/비유지 설정'을 유지하는 것으로 설정한 경우는 전원 OFF 시에도 유지
주 3: TMHH/TMHHX 명령을 실행할 때 설정값이 설정됩니다.

- JMP/CJMP/CJPN-JME 명령에 의해 점프한 경우에도 기동 중인 타이머의 현재값은 갱신됩니다.*1 (명령은 실행되지 않지만 1ms마다 현재값이 갱신됩니다.)

  *1: CS1D CPU 유니트에서는 갱신되지 않습니다.
- IL-ILC 명령에 의해 인터록된 경우는 리셋되어 현재값 = 설정값이 되고 타임업 플래그 = OFF가 됩니다.
- 강제 세트할 때는 타임업 플래그=ON, 현재값=0이 되고 강제 리셋할 때는 타임업 플래그=OFF, 현재값=설정값이 됩니다.
- 온라인 편집에서 타이머 번호를 사용하는 타이머 명령 간에 재기입(TIM 명령←→TIMH 명령←→TMHH 명령 등)했을 때는 반드시 타임업 플래그를 리셋하십시오. 타임업 플래그를 리셋하지 않으면 재기입 후 타이머가 정상적으로 동작하지 않습니다.
- 유니트 간 동기 제어 기능을 사용할 때는 다음과 같은 제약이 있습니다.
  - 사이클 시간이 100ms를 초과하면 타이머가 정상적으로 동작하지 않습니다.
  - TMHH/TMHHX 명령이 정지한 태스크 내에 있거나 JMP/CJMP/CJPN-JME 명령에서 점프되어 실행되지 않은 경우, 타이머 정도에 최대 -1ms의 오차가 발생합니다.
  - 타이머 현재값은 참조할 수 없는데, 참조하는 경우에는 잘못된 값을 읽어옵니다.

TIMU/TIMUX

명령 이름	니모닉	동작 옵션	펑션 번호	기능
0.1ms 타이머	TIMU	–	541	감산식 ON 딜레이 0.1ms 단위의 타이머의 동작을 나타냅니다.
	TIMUX	–	556	

심볼	TIMU	TIMUX
	BCD 방식일 때 TIMU N: 타이머 번호 N S: 타이머 설정값 S	BIN 방식일 때 TIMUX N: 타이머 번호 N S: 타이머 설정값 S

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기
		TIMU	TIMUX	
N	타이머 번호	TIMER	TIMER	1
S	타이머 설정값	WORD	UINT	1

N: 타이머 번호
10진수 0~4095

S: 타이머 설정값
BCD: #0000~9999
BIN: 10진수 &0~65535 또는 16진수 #0000~FFFF

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	○	–	–	–	–	–	–	–	–	○	–	–	–	–
S	○	○	○	○		○	○	○	○	○	○	○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· N의 타이머 번호가 간접 IR 지정인 경우, 주소가 타임업 플래그 또는 타이머의 현재값 주소가 아닐 때 ON · BCD 방식이고 S의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	변하지 않음
부정 플래그	P_N	변하지 않음

기능

- 타이머 입력이 OFF일 때는 N으로 지정된 번호의 타이머를 리셋합니다. (타임업 플래그를 OFF로 합니다.)
- 타이머 입력이 OFF→ON으로 된 시점에 타이머를 기동하고 타이머 현재값의 감산을 시작합니다. 타이머 입력이 ON인 동안 타이머 설정값을 경과하면 타임업 플래그를 ON으로 합니다(타임업).
- 타임업된 후에는 타임업 플래그의 상태를 유지합니다. 재기동하려면 일단 타이머 입력을 OFF→ON으로 해야 합니다.
- 이 명령은 타임업 플래그만 참조하십시오. 타이머 현재값은 시스템에서 사용하므로 참조할 수 없습니다.
- 설정 시간은 다음과 같습니다.
  - BCD 방식인 경우 0~0.9999초
  - BIN 방식인 경우 0~6.5535초
- 타이머 정도는 -0.1~0ms입니다.

힌트

- TIMU/TIMUX 명령의 타임업 플래그 갱신

갱신 타이밍	내용
각 명령 실행 시	설정 시간을 경과했을 때 타임업 플래그ON, 경과하지 않았을 때 타임업 플래그 OFF

사용상의 주의 사항

- 타이머 번호는 다른 타이머 명령과 공용입니다. 이들 명령에서 같은 타이머 번호를 동시에 동작시키면 오작동하므로 주의하십시오. 동시에 사용하면 프로그램 확인 시 '코일 이중 사용'이 표시됩니다. 또한 동시에 동작시키지 않는 조건이면 동일한 번호를 사용할 수 있습니다.
- 타이머 현재값은 참조할 수 없습니다.
- 타임업 플래그는 명령 실행 시에만 갱신되기 때문에 타임업 플래그를 사용자 프로그램에서 전송하는 경우, 타이밍에 따라 1사이클 시간만큼 지연이 발생할 수 있으므로 주의하십시오.
- 사이클 시간이 100ms 이상이면 정상적으로 동작하지 않으므로 주의하십시오.
- 타이머는 다음과 같은 경우에 리셋 또는 유지됩니다. (리셋할 때는 타임업 플래그 = OFF가 됩니다.)

	동작 모드 변경 시 (프로그램←→운전 또는 모니터) (주 1)	전원 OFF 복구 시 (주 2)	CNR/CNRX(타이머/카운터 리셋 명령) 실행	IL-ILC 명령에 의한 인터록 시	JMP-JME 명령에 의한 점프 시
타임업 플래그	OFF	OFF	OFF	OFF	그 이전 상태를 유지

주 1: IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정한 경우는 동작 모드 변경 시에도 유지

주 2: IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정하고, PLC 시스템 설정에서 '전원 ON 시 IOM(I/O 메모리) 유지 비트 유지/비유지 설정'을 유지하는 것으로 설정한 경우는 전원 OFF 시에도 유지

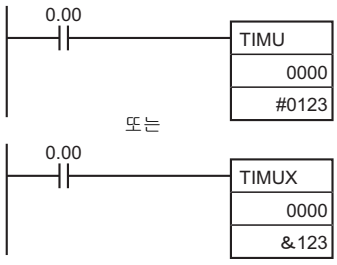
- IL-ILC 명령에 의해 인터록된 경우는 리셋되어 타임업 플래그 = OFF가 됩니다.
- JMP/CJMP/CJPN-JME 명령에 의해 점프했을 때 오차가 발생하는 경우가 있으므로 주의하십시오.
- 강제 세트할 때는 타임업 플래그=ON이 되고 강제 리셋할 때는 타임업 플래그=OFF가 됩니다.
- 온라인 편집에서 타이머 명령을 재기입할 때는 반드시 타임업 플래그를 리셋하십시오. 타임업 플래그를 리셋하지 않으면 재기입 후 타이머가 정상적으로 동작하지 않습니다.

샘플 프로그램

타이머 입력 0.00이 OFF→ON으로 된 경우 타이머의 감산을 시작합니다.

12.3ms 후에 타임업 플래그 T0000이 ON이 됩니다.

타이머 입력 0.00이 OFF가 되면 타임업 플래그 T0000이 OFF가 됩니다.



TMUH/TMUHX

명령 이름	니모닉	동작 옵션	평션 번호	기능
0.01ms 타이머	TMUH	–	544	감산식 ON 딜레이 0.01ms 단위의 타이머의 동작을 나타냅니다.
	TMUHX	–	557	

심볼	TMUH	TMUHX
	BCD 방식일 때 TMUH N S N: 타이머 번호 S: 타이머 설정값	BIN 방식일 때 TMUHX N S N: 타이머 번호 S: 타이머 설정값

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기
		TMUH	TMUHX	
N	타이머 번호	TIMER	TIMER	1
S	타이머 설정값	WORD	UINT	1

N: 타이머 번호
10진수 0~4095
S: 타이머 설정값
BCD: #0000~9999
BIN: 10진수 &0~65535 또는 16진수 #0000~FFFF

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	○	–	–	–	–	–	–	–	–	○	–	–	–	–
S	○	○	○	○		○	○	○	○	○	○	○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· N의 타이머 번호가 간접 IR 지정인 경우, 주소가 타임업 플래그 또는 타이머의 현재값 주소가 아닐 때 ON · BCD 방식이고 S의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	변하지 않음
부정 플래그	P_N	변하지 않음

기능

· 타이머 입력이 OFF일 때는 N으로 지정된 번호의 타이머를 리셋합니다. (타임업 플래그를 OFF로 합니다.)
· 타이머 입력이 OFF→ON으로 된 시점에 타이머를 기동하고 타이머 현재값의 감산을 시작합니다. 타이머 입력이 ON인 동안 타이머 설정값을 경과하면 타임업 플래그를 ON으로 합니다(타임업).

· 타임업된 후에는 타임업 플래그의 상태를 유지합니다. 재기동하려면 일단 타이머 입력을 OFF→ON으로 해야 합니다.
· 이 명령은 타임업 플래그만 참조하십시오. 타이머 현재값은 시스템에서 사용하므로 참조할 수 없습니다.
· 설정 시간은 다음과 같습니다.
  · BCD 방식인 경우 0~0.09999초
  · BIN 방식인 경우 0~0.65535초
· 타이머 정도는 -0.01~0ms입니다.

힌트

· TMUH/TMUHX 명령의 타임업 플래그 갱신

갱신 타이밍	내용
각 명령 실행 시	설정 시간을 경과했을 때 타임업 플래그 ON, 경과하지 않았을 때 타임업 플래그 OFF

사용상의 주의 사항

· 타이머 번호는 다른 타이머 명령과 공용입니다. 이들 명령에서 같은 타이머 번호를 동시에 동작시키면 오작동하므로 주의하십시오. 동시에 사용하면 프로그램 확인 시 '코일 이중 사용'이 표시됩니다. 또한 동시에 동작시키지 않는 조건이면 동일한 번호를 사용할 수 있습니다.
· 타이머 현재값은 참조할 수 없습니다.
· 타임업 플래그는 명령 실행 시에만 갱신되기 때문에 타임업 플래그를 사용자 프로그램에서 전송하는 경우, 타이밍에 따라 1사이클 시간만큼 지연이 발생할 수 있으므로 주의하십시오.
· 사이클 시간이 10ms 이상이면 정상적으로 동작하지 않으므로 주의하십시오.
· 타이머는 다음과 같은 경우에 리셋 또는 유지됩니다. (리셋할 때는 타임업 플래그 = OFF가 됩니다.)

	동작 모드 변경 시 (프로그램←→운전 또는 모니터) (주 1)	전원 OFF 복구 시 (주 2)	CNR/CNRX(타이머/카운터 리셋 명령) 실행	IL-ILC 명령에 의한 인터록 시	JMP-JME 명령에 의한 점프 시
타임업 플래그	OFF	OFF	OFF	OFF	그 이전 상태를 유지

주 1: IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정한 경우는 동작 모드 변경 시에도 유지
주 2: IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정하고, PLC 시스템 설정에서 '전원 ON 시 IOM(I/O 메모리) 유지 비트 유지/비유지 설정'을 유지하는 것으로 설정한 경우는 전원 OFF 시에도 유지

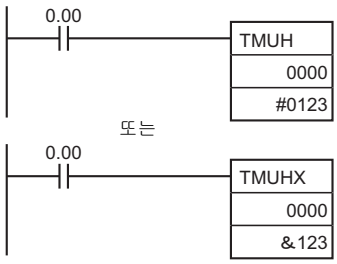
· IL-ILC 명령에 의해 인터록된 경우는 리셋되어 타임업 플래그 = OFF가 됩니다.
· JMP/CJMP/CJPN-JME 명령에 의해 점프했을 때 오차가 발생하는 경우가 있으므로 주의하십시오.
· 강제 세트할 때는 타임업 플래그=ON이 되고 강제 리셋할 때는 타임업 플래그=OFF가 됩니다.
· 온라인 편집에서 타이머 명령을 재기입할 때는 반드시 타임업 플래그를 리셋하십시오. 타임업 플래그를 리셋하지 않으면 재기입 후 타이머가 정상적으로 동작하지 않습니다.

샘플 프로그램

타이머 입력 0.00이 OFF→ON으로 된 경우 타이머의 감산을 시작합니다.

1.23ms 후에 타임업 플래그 T0000이 ON이 됩니다.

타이머 입력 0.00이 OFF가 되면 타임업 플래그 T0000이 OFF가 됩니다.



TTIM/TTIMX

명령 이름	니모닉	동작 옵션	평션 번호	기능
적산 타이머	TTIM	–	087	적산식 ON 딜레이 0.1초 단위의 타이머 동작을 합니다.
	TTIMX	–	555	

심볼	TTIM	TTIMX
	BCD 방식일 때 타이머 입력 → TTIM 리셋 입력 → N, S N: 타이머 번호 S: 타이머 설정값	BIN 방식일 때 타이머 입력 → TTIMX 리셋 입력 → N, S N: 타이머 번호 S: 타이머 설정값

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기
		TMUH	TMUHX	
N	타이머 번호	TIMER	TIMER	1
S	타이머 설정값	WORD	UINT	1

N: 타이머 번호
10진수 0~4095
S: 타이머 설정값
BCD: #0000~9999
BIN: 10진수 &0~65535 또는 16진수 #0000~FFFF

■ 영역 종류

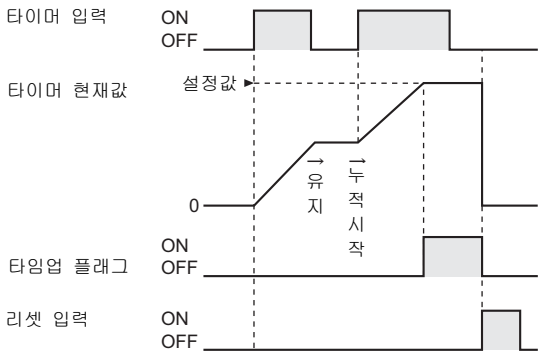
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	○	–	–	–	–	–	–	–	–	○	–	–	–	–
S	○	○	○	○		○	○	○	○	○	○	○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· N의 타이머 번호가 간접 IR 지정인 경우, 주소가 타임업 플래그 또는 타이머의 현재값 주소가 아닐 때 ON · BCD 방식이고 S의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF

기능

- 타이머 입력이 ON인 동안 현재값을 가산(누적)합니다. 타이머 입력이 OFF가 되면 누적을 정지하고 현재값을 유지합니다. 타이머 입력이 다시 ON이 되면 누적을 시작합니다. 타이머 현재값이 설정값에 도달하면 타임업 플래그가 ON이 됩니다.
- 타임업된 후에는 타이머 현재값 및 타임업 플래그의 상태를 유지합니다. 재기동하려면 MOV 명령 등에서 타이머 현재값을 설정값 이하로 하거나, 리셋 입력 ON 또는 CNR/CNRX 명령을 사용하여 타이머를 리셋합니다.
- 설정 시간은 다음과 같습니다.
  - BCD 방식인 경우 0~999.9초
  - BIN 방식인 경우 0~6553.5초
- 타이머 정도는 -0.01~0초입니다. CS1D CPU 유니트에서는 타이머 정도가 ±(10ms + 사이클 시간)입니다.



힌트

- 일반적인 TIM/TIMX 명령은 감산 카운트 방식으로 동작하므로 현재값은 카운트 업할 때까지 남은 시간을 나타냅니다. 그에 비해 TTIM/TTIMX(적산 타이머) 명령은 경과 시간이 현재값이므로 그대로 현재값을 연산에 사용하거나 표시 출력할 수 있습니다.

사용상의 주의 사항

- 타이머 번호는 다른 타이머 명령과 공용입니다. 이들 명령에서 같은 타이머 번호를 동시에 동작시키면 오작동하므로 주의하십시오. 동시에 사용하면 프로그램 확인 시 '코일 이중 사용'이 표시됩니다. 또한 동시에 동작시키지 않는 조건이면 동일한 번호를 사용할 수 있습니다.
- 타이머는 다음과 같은 경우에 리셋 또는 유지됩니다. (리셋할 때는 타이머 현재값 = 0, 타임업 플래그 = OFF가 됩니다.)

	동작 모드 변경 시 (프로그램←→운전 또는 모니터) (주 1)	전원 OFF 복구 시 (주 2)	CNR/CNRX(타이머/카운터 리셋 명령) 실행 (주 3)	IL-ILC 명령에 의한 인터록 시	JMP-JME 명령에 의한 점프 시
현재값	0	0	BCD 방식일 때: #9999 BIN 방식일 때: #FFFF	그 이전 상태를 유지	그 이전 상태를 유지
타임업 플래그	OFF	OFF	OFF	그 이전 상태를 유지	그 이전 상태를 유지

주 1: IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정한 경우는 동작 모드 변경 시에도 유지

주 2: IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정하고, PLC 시스템 설정에서 '전원 ON 시 IOM(I/O 메모리) 유지 비트 유지/비유지 설정'을 유지하는 것으로 설정한 경우는 전원 OFF 시에도 유지

주 3: TTIM/TTIMX 명령을 실행할 때 설정값이 설정됩니다.

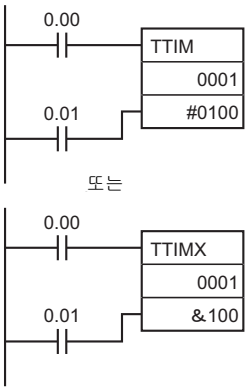
- IL-ILC 명령에 의해 인터록된 경우는 현재값을 유지하며 리셋되지 않습니다. 이로 인해 오차가 생기는 경우가 있으므로 주의하십시오.
- JMP-JME 명령에 의해 점프했을 때는 현재값을 유지합니다. 이로 인해 오차가 생기는 경우가 있으므로 주의하십시오.
- 강제 세트할 때는 타임업 플래그=ON, 현재값=0이 되고 강제 리셋할 때는 타임업 플래그=OFF, 현재값= 0이 됩니다. 또한 강제 세트/리셋은 리셋 입력/타이머 입력보다 우선됩니다.
- 현재값의 누적은 명령 실행 시에만 이루어지므로 사이클 시간이 100ms 이상이면 100ms 단위의 누적이므로 정상적으로 동작하지 않으므로 주의하십시오.
- 타임업 플래그는 명령 실행 시에만 갱신되기 때문에 타임업 플래그를 사용자 프로그램에서 전송하는 경우, 타이밍에 따라 1사이클 시간만큼 지연이 발생할 수 있으므로 주의하십시오.

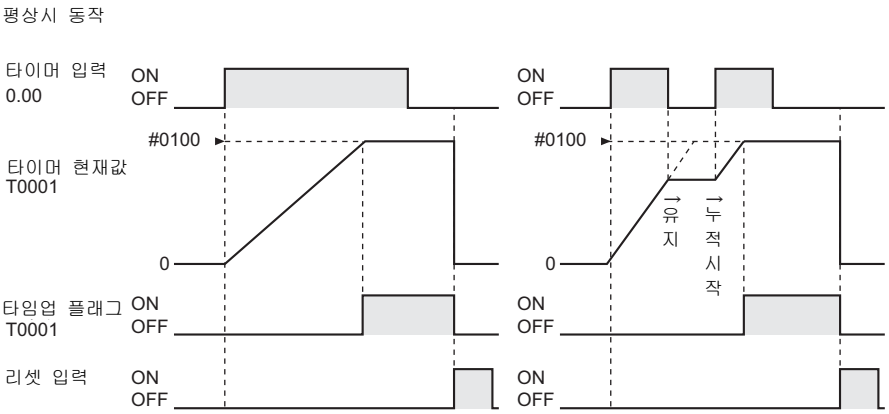
샘플 프로그램

타이머 입력 0.00이 ON이면 타이머 현재값이 적산 카운트를 시작하고, 타이머 현재값 = 타이머 설정값이 되면 타임업 플래그 T0001이 ON이 됩니다.

리셋 입력이 ON이 되면 타이머 현재값이 0이 되고 타임업 플래그 T0001은 OFF가 됩니다. (평상시에는 리셋 입력을 ON으로 한 후 타이머 입력을 ON으로 하여 사용합니다.)

설정값에 도달하기 전에 타이머 입력이 OFF가 되면 누적을 정지하고 타이머 현재값을 유지합니다. 타이머 입력이 다시 ON이 되면 누적을 시작합니다.





TIML/TIMLX

명령 이름	니모닉	동작 옵션	평션 번호	기능
장시간 타이머	TIML	–	542	감산식 ON 딜레이 100ms 타이머입니다.
	TIMLX	–	553	

심볼	TIML	TIMLX
	BCD 방식일 때 	BIN 방식일 때 

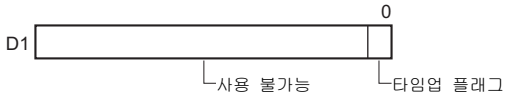
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	불가능	가능

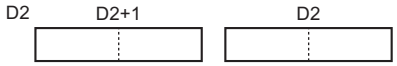
피연산자 설명

피연산자	내용	데이터 형식		크기
		TIML	TIMLX	
D1	타임업 플래그 CH 번호	WORD	UINT	1
D2	현재값 출력 하위 CH 번호	DWORD	UDINT	2
S	타이머 설정값 하위 CH 번호	DWORD	UDINT	2

D1: 타임업 플래그 CH 번호



D2: 현재값 출력 하위 CH 번호



S: 타이머 설정값 하위 CH 번호



D2, S의 범위

- BCD 방식일 때:
  BCD#00000000~99999999
- BIN 방식일 때:
  10진수 &00000000~4294967294
  또는 16진수 #00000000~FFFFFFFF

주: D2+1, D2 및 S+1, S는 각각 동일 영역 종류이어야 합니다.

■ 영역 종류

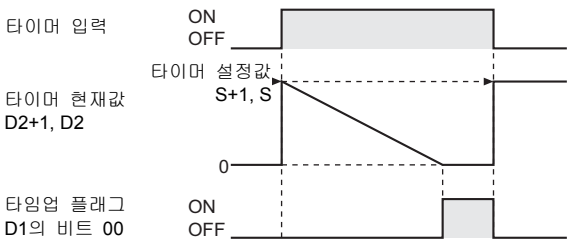
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
D1	○	○	○	○	–	–	○	○	○	○	–	–	–	○	–	–	–	–
D2																		
S					○	○					○							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· BCD 방식이고 D2의 데이터가 BCD가 아닐 때 ON · BCD 방식이고 S의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF

기능

- 타이머 입력이 OFF일 때는 타이머를 리셋합니다. (타이머 현재값 D2+1, D2에 설정값 S+1, S를 대입하고 타임업 플래그를 OFF로 합니다.)
- 타이머 입력이 OFF→ON으로 된 시점에 타이머를 기동하고 타이머 현재값 D2+1, D2의 감산을 시작합니다. 타이머 입력이 ON인 동안, 타이머 현재값의 갱신을 실행하고 타이머 현재값이 0이 되면 타임업 플래그를 ON으로 합니다(타임업).
- 타임업된 후에는 타이머 현재값 및 타임업 플래그의 상태를 유지합니다. 재기동하려면 일단 타이머 입력을 OFF→ON으로 하거나 MOV 명령 등을 사용하여 타이머 현재값 D2+1, D2를 0 이외의 값으로 변경해야 합니다.



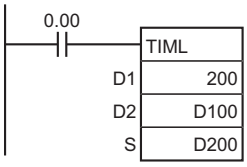
- 타이머 정도는 −0.01~0초입니다. CS1D CPU 유니트에서는 타이머 정도가 ±(10ms + 사이클 시간)입니다.
- 최대 시간 설정(초 단위)은 다음과 같습니다.
  - BCD 방식인 경우 115일
  - BIN 방식인 경우 4971일

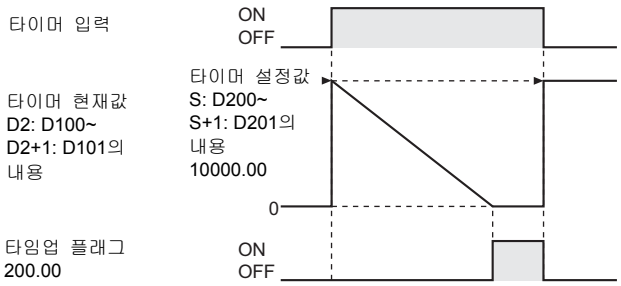
사용상의 주의 사항

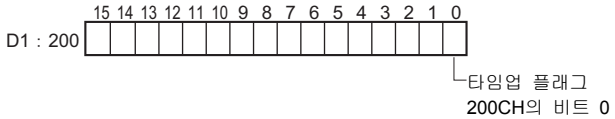
- 타이머 번호는 사용하지 않습니다. (타이머 현재값 영역을 사용한 타이머 현재값 갱신은 실행하지 않습니다.)
- 타임업 플래그를 일반적인 접점 영역으로 하여 강제 세트/리셋할 수 있지만 현재값은 변하지 않습니다.
- 현재값의 갱신은 명령 실행 시에만 이루어지며 사이클 시간이 100ms 이상이면 100ms 단위로 갱신되므로 정상적으로 동작하지 않을 수 있으니 주의하십시오.
- 타임업 플래그는 명령 실행 시에만 갱신되기 때문에 타임업 플래그를 사용자 프로그램에서 전송하는 경우, 타이밍에 따라 1사이클 시간만큼 지연이 발생할 수 있으므로 주의하십시오.
- IL-ILC 명령에 의해 인터록된 경우는 리셋되어 현재값 = 설정값이 되고 타임업 플래그 = OFF가 됩니다.
- JMP-JME 명령에 의해 점프했을 때는 현재값을 유지합니다. 이로 인해 오차가 생기는 경우가 있으므로 주의하십시오.
- 장시간 타이머 명령에서 사용하는 D1, D2, D2+1은 다른 명령에서 사용하지 마십시오. 다른 명령에서 사용하면 정상적인 타임업 동작이 보장되지 않으므로 주의하십시오.

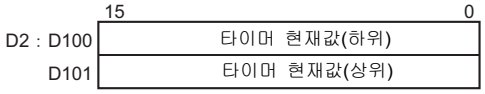
샘플 프로그램

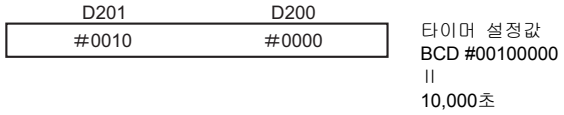
타이머 입력 0.00이 ON이면 타이머 현재값(D101, D100)이 타이머 설정값(D201, D200)이 되고 감산을 시작합니다. 타이머 현재값 = 0이 되면 타임업 플래그 200.00이 ON이 됩니다. 타이머 입력 0.00이 OFF가 되면 타임업 플래그 200.00이 OFF가 됩니다.





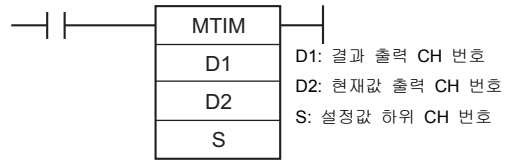
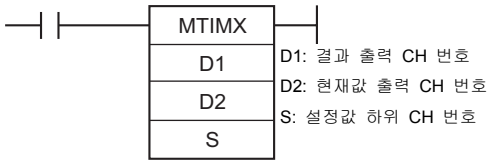






MTIM/MTIMX

명령 이름	니모닉	동작 옵션	펑션 번호	기능
멀티 출력 타이머	MTIM	–	543	8점의 임의 타임업 플래그 값을 얻을 수 있는 누적식 ON 딜레이 0.1초 단위의 타이머입니다.
	MTIMX	–	554	

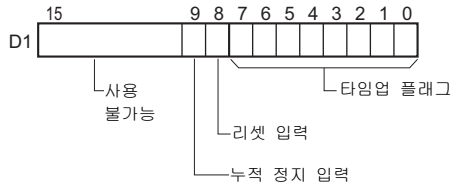
심볼	MTIM	MTIMX
	BCD 방식일 때 	BIN 방식일 때 

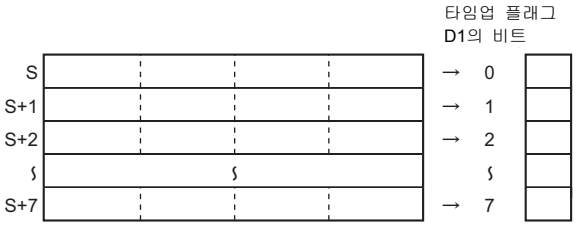
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기
		MTIM	MTIMX	
D1	결과 출력 CH 번호	UINT	UINT	1
D2	현재값 출력 CH 번호	WORD	UINT	1
S	설정값 하위 CH 번호	WORD	WORD	8





D2, S~S+7의 범위
- BCD 방식일 때:
  BCD#0000~9999
- BIN 방식일 때:
  10진수 &0~65535
  또는 16진수 #0000~FFFF

주: S~S+7은 각각 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
D1	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D2												○						
S												–						

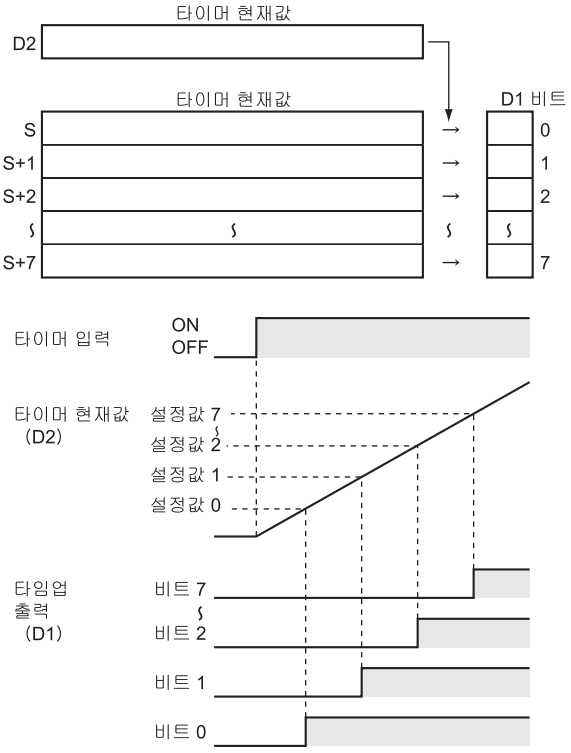
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	• BCD 방식이고 D2의 데이터가 BCD가 아닐 때 ON • 그 이외일 때 OFF

기능

- 입력 조건이 ON인 상태에서 누적 정지 입력 및 리셋 입력이 OFF이면 D2로 지정한 현재값을 누적합니다. 누적 정지 입력이 ON이 되면 누적을 정지하고 현재값을 유지합니다. 누적 정지 입력이 다시 OFF가 되면 누적을 시작합니다.
- S~S+7CH의 각 설정값에 대해 현재값≧ 설정값이 되면 해당하는 8점의 타임업 플래그가 ON이 됩니다.
- 현재값이 BCD 방식일 때: 9999, BIN 방식일 때: FFFF에 도달하면 0으로 돌아가서 타임업 플래그가 모두 OFF가 됩니다. 누적 중에 리셋 입력이 ON이 된 경우에도 현재값은 0으로 돌아가고 모든 타임업 플래그가 OFF가 됩니다.
- 설정값은 다음과 같습니다.
  - MTIM의 경우(BCD): 0~999.9초
  - MTIMX의 경우(BIN): 0~6553.5초
- 타이머 정도는 -0.01~0초입니다.

주: CS1D CPU 유니트에서는 타이머 정도가 ±(10ms + 사이클 시간)입니다.



- 누적 정지 입력과 리셋 입력의 관계에 따라 MTIM/MTIMX 명령은 다음과 같이 실행됩니다.

		누적 정지 입력(09비트)	
		0	1
리셋 입력 (08비트)	0	현재값을 갱신하고 설정값≦현재값이 되면 타임업 플래그의 해당 비트를 ON으로 설정합니다.	타이머 현재값 갱신을 정지합니다(NOP 처리).
	1	현재값, 타임업 플래그를 리셋하고 현재값 갱신을 정지합니다.	

- 누적 정지 입력, 리셋 입력은 MTIM/MTIMX 명령의 입력이 ON일 때만 유효합니다.

힌트

- D1(결과 출력) CH를 릴레이 영역에서 지정한 경우, 현재값 갱신 정지, 현재값 리셋은 SET/RSET 명령으로 ON/OFF할 수 있습니다.

사용상의 주의 사항

- 타이머 번호는 사용하지 않습니다. (타이머 현재값 영역을 사용한 타이머 현재값 갱신은 실행하지 않습니다.)
- 현재값이 BCD 방식일 때: #9999, BIN 방식일 때: #FFFF에 도달하면 #0000으로 돌아가서 타임업 플래그가 모두 OFF가 됩니다.
- BCD 방식의 경우 S~S+7의 설정값 데이터가 BCD가 아니면 해당 CH는 무시됩니다. 이때, 에러 플래그는 ON되지 않습니다.
- 타임업 플래그를 일반적인 접점 영역으로 하여 강제 세트/리셋할 수 있지만 현재값은 변하지 않습니다.
- 설정값을 8점 이하로 사용하는 경우

  S~S+7CH의 설정값에 0000이 있으면 그 CH 이후의 데이터는 무시됩니다.

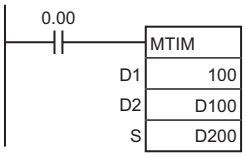
SCH	2CH	1	0	2	9
	3CH	2	5	0	6
	4CH	6	0	4	0
	5CH	0	0	0	0
	⟆	⟆			
S+7CH	9CH				

무시됩니다.

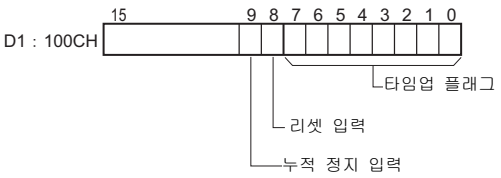
MTIM/MTIMX

- 현재값의 누적은 명령 실행 시에만 이루어지므로 사이클 시간이 100ms 이상이면 100ms 단위의 누적이므로 정상적으로 동작하지 않습니다. 따라서 고정도를 요구하는 경우는 사이클 시간의 영향을 받지 않도록 하기 위해 같은 TIM/MTIMX 명령을 필요에 따라 여러 개 프로그래밍하십시오.
- 타임업 플래그는 명령 실행 시에만 갱신되기 때문에 타임업 플래그를 사용자 프로그램에서 전송하는 경우, 타이밍에 따라 1사이클 시간만큼 지연이 발생할 수 있으므로 주의하십시오.
- IL-ILC 명령에 의해 인터록된 경우 현재값을 유지하며 리셋되지 않습니다. 이로 인해 오차가 생기는 경우가 있으므로 주의하십시오.
- JMP-JME 명령에 의해 점프했을 때는 현재값을 유지합니다. 이로 인해 오차가 생기는 경우가 있으므로 주의하십시오.
- 멀티 출력 타이머 명령에서 사용하는 D1, D2는 다른 명령에서 사용하지 마십시오. 다른 명령에서 사용하면 정상적인 타임업 동작이 보장되지 않으므로 주의하십시오.

샘플 프로그램



0.00이 ON 상태이고 100CH의 비트 9가 OFF 상태일 때 100CH의 비트 8이 ON→OFF되면 타이머가 기동합니다.
0000부터 카운트를 시작합니다. 타이머의 현재값이 누적되어 D100에 저장됩니다. D200~D207의 8CH의 각 타이머 설정값에 대해 타이머 현재값≧타이머 설정값이 되면 각 설정값에 대응하는 타임업 플래그 100CH의 비트 0~7이 ON이 됩니다.



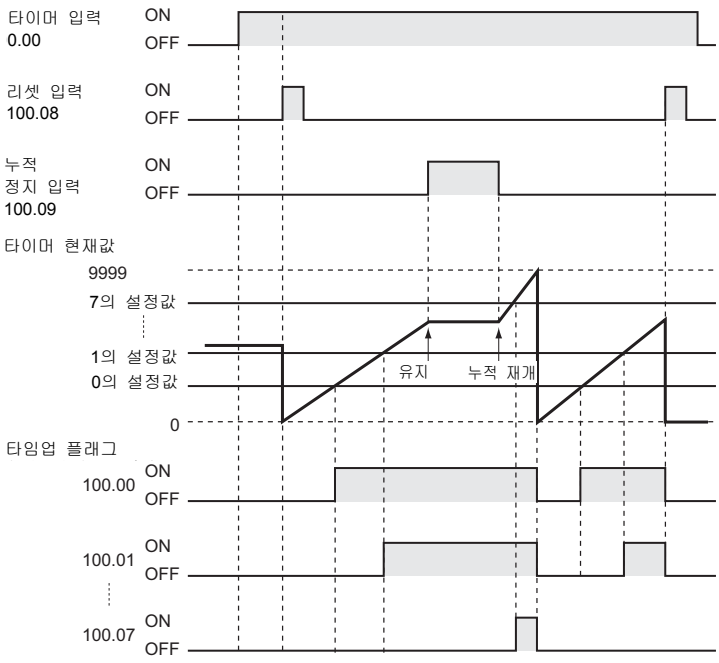
타이머 현재값 15 0 일 때

D2 : D100	0	1	0	0

(변경)

타이머 설정값

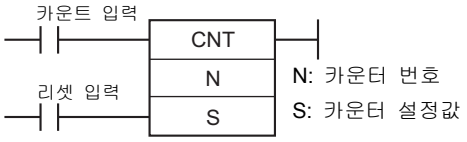
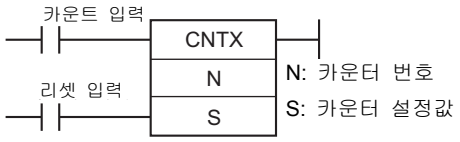
						타임업 출력 100CH의 비트
S : D200	0	0	8	0	→	1 0
S+1 : D201	0	0	9	0	→	1 1
S+2 : D202	0	1	0	0	→	1 2
S+3 : D203	0	1	1	0	→	0 3
S+4 : D204	0	1	2	0	→	0 4
S+5 : D205	0	1	3	0	→	0 5
S+6 : D206	0	1	5	0	→	0 6
S+7 : D207	1	0	0	0	→	0 7



주: 적산 동작 중에는 ON을 유지해야 합니다.

CNT/CNTX

명령 이름	니모닉	동작 옵션	펑션 번호	기능
카운터	CNT/CNTX	–	546	감산 카운터의 동작을 수행합니다.

심볼	CNT	CNTX
	BCD 방식일 때 	BIN 방식일 때 

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기
		CNT	CNTX	
N	카운터 번호	COUNTER	COUNTER	1
S	카운터 설정값	WORD	UINT	1

N: 카운터 번호
10진수 0~4095
S: 카운터 설정값
BCD: #0000~9999
BIN: 10진수 &0~65535 또는 16진수 #0000~FFFF

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	○	–	–	–	–	–	–	–	○	–	–	–	–
S	○	○	○	○	○		○	○	○	○	○	○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· N의 카운터 번호가 간접 IR 지정인 경우, 주소가 타이머의 현재값 주소가 아닐 때 ON · BCD 방식이고 S의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	변하지 않음*1
부정 플래그	P_N	변하지 않음*1

*1: CS1/CJ1/CS1D(이중화 시스템용) CPU 유니트는 OFF

CNT/CNTX

기능

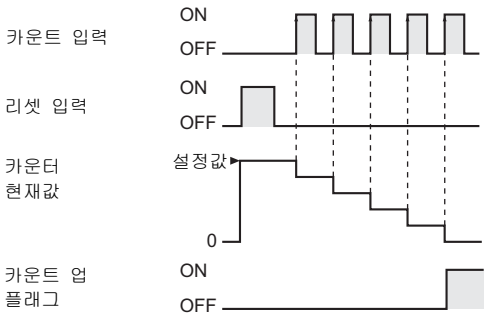
카운트 입력이 기동될 때마다 카운터 현재값이 감산 카운트됩니다. 카운터 현재값 = 0이 되면 카운트 업 플래그가 ON이 됩니다.

카운트 업 후, 리셋 입력 ON 또는 CNR/CNRX 명령을 사용하여 카운터 리셋을 실행하지 않으면 재기동되지 않습니다.

리셋 입력 ON이면 리셋이 되고(현재값=설정값, 카운트 업 플래그 = OFF) 카운트 입력은 무효가 됩니다.

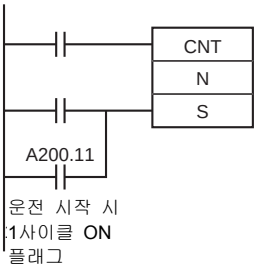
설정값은 다음과 같습니다.

- BCD 방식인 경우 0~9999회
- BIN 방식인 경우 0~65535회



힌트

- 카운터 현재값은 전원이 꺼져도 유지됩니다. 따라서 전원을 켰을 때 현재값을 설정값으로 시작하고 싶은 경우(전원 OFF 직전의 카운터 현재값에서 시작하고 싶지 않은 경우)는 카운터 리셋 입력에 1사이클 ON 플래그 A200.11(특수 보조 릴레이: 운전 시작 시의 1사이클 동안만 ON)을 삽입하십시오.



사용상의 주의 사항

- 카운터 번호는 카운터 명령, 가역 카운터 명령, 블록 프로그램의 카운터 대기 명령에서 공용입니다. 이들 명령에서 같은 카운터 번호를 동시에 동작시키면 오작동하므로 주의하십시오. 동시에 사용하면 프로그램 확인 시 '코일 이중 사용'이 표시됩니다. 또한 동시에 동작시키지 않는 조건이면 동일한 번호를 사용할 수 있습니다.

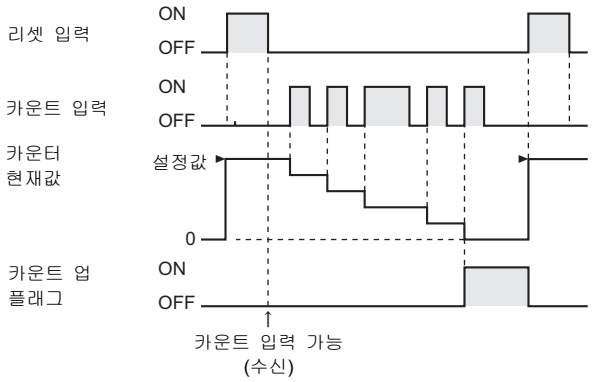
카운터 현재값, 카운트 업 플래그의 갱신 타이밍

	갱신 타이밍
카운터 현재값	카운트 입력 기동 시
카운트 업 플래그	명령 실행 시마다(현재값이 0인 경우 ON, 0 이외인 경우 OFF)

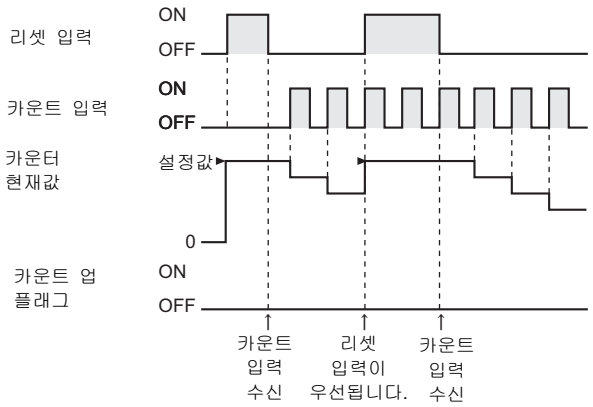
강제 세트/리셋 시의 카운터 현재값, 카운트 업 플래그의 상태

	강제 세트 시	강제 리셋 시
카운터 현재값	0	설정값
카운트 업 플래그	ON	OFF

- 다음과 같이 카운트 입력 OFF→ON 시작 전에 리셋 입력을 반드시 OFF→ON으로 하여 리셋하십시오. 또한 카운트 입력의 OFF→ON 전에 리셋 입력을 ON→OFF로 하십시오. 리셋 입력이 ON인 동안은 카운트 입력이 수신되지 않습니다.



· 리셋 입력과 카운트 입력이 동시에 ON으로 되면 리셋 입력이 우선되고 카운터는 리셋(카운터 현재값 = 설정값, 카운트 업 플래그 = OFF)됩니다.



· 온라인 편집에서 카운터를 추가하는 경우는 카운터를 리셋한 후 사용하십시오. 카운터 현재값이 이전 값을 유지하고 있어서 프로그램 변경 후의 카운터가 정상적으로 동작하지 않는 경우가 있습니다.

CNTR/CNTRX

명령 이름	니모닉	동작 옵션	평션 번호	기능
가역 카운터	CNTR	–	012	가감산 카운터 동작을 합니다. 가산 카운트 입력의 기동으로 가산을 실행하고 감산 카운트 입력의 기동으로 감산을 실행합니다.
	CNTRX	–	548	

심볼	CNTR	CNTRX
	BCD 방식일 때 가산 카운트 감산 카운트 리셋 입력 CNTR N S N: 카운터 번호 S: 카운터 설정값	BIN 방식일 때 가산 카운트 감산 카운트 리셋 입력 CNTRX N S N: 카운터 번호 S: 카운터 설정값

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기
		CNTR	CNTRX	
N	카운터 번호	COUNTER	COUNTER	1
S	카운터 설정값	WORD	UINT	1

N: 카운터 번호
10진수 0~4095
S: 카운터 설정값
BCD: #0000~9999
BIN: 10진수 &0~65535 또는 16진수 #0000~FFFF

■ 영역 종류

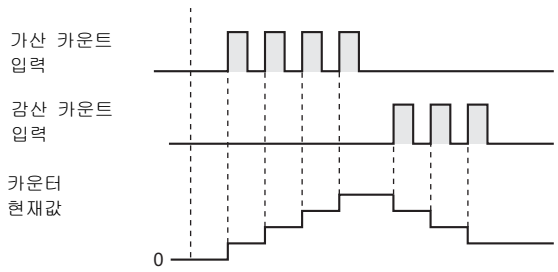
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	○	–	–	–	–	–	–	–	○	–	–	–	–
S	○	○	○	○	○		○	○	○	○	○	○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· N의 카운터 번호가 간접 IR 지정인 경우, 주소가 타이머의 현재값 주소가 아닐 때 ON · BCD 방식이고 S의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF

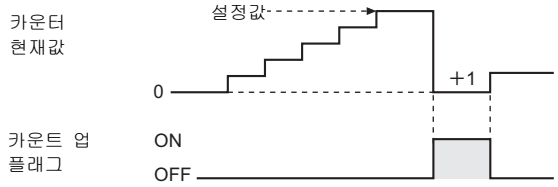
기능

가산 카운트 입력의 기동으로 가산을 실행하고 감산 카운트 입력의 기동으로 감산을 실행합니다. 가산에 의해 현재값이 설정값에서 0으로 자리올림되면 카운트 업 플래그가 ON이 되고 0에서 1로 가산되면 OFF가 됩니다. 또한 감산에 의해 현재값이 0에서 설정값으로 자리내림되면 ON이 되고 설정값에서 1씩 감산될 때 OFF가 됩니다.

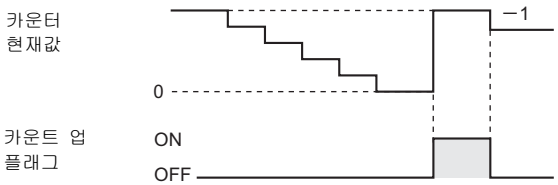


카운트업 플래그의 동작

・가산을 사용할 때



・감산을 사용할 때

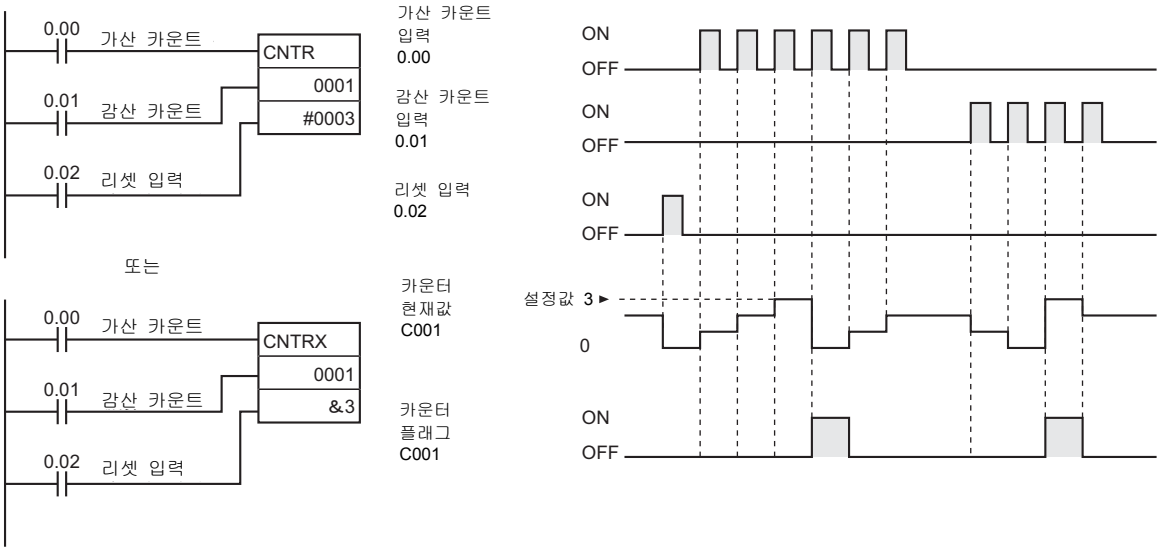


사용상의 주의 사항

- 카운터 번호는 카운터 명령, 가역 카운터 명령, 카운터 대기 명령에서 공용입니다. 이들 명령에서 같은 카운터 번호를 동시에 동작시키면 오작동하므로 주의하십시오. 동시에 사용하면 프로그램 확인 시 '코일 이중 사용'이 표시됩니다. 또한 동시에 동작시키지 않는 조건이면 동일한 번호를 사용할 수 있습니다.
- 가산, 감산 양쪽의 입력이 동시에 기동하는 경우는 계산하지 않습니다. 리셋 입력 ON일 때는 현재값=0으로 하고 카운트 입력은 무효가 됩니다. 카운트 업 플래그는 자리올림(가산 시) 및 자리내림(감산 시)에서 ON이 되고 그 이외에는 OFF가 됩니다.
- CNTR/CNTRX 명령은 래더도와 니모닉에서 입력 순서가 다음과 같이 달라집니다.
  - 래더도:
    가산 입력→CNTR/CNTRX 명령→감산 입력→리셋 입력
  - 니모닉:
    가산 입력→감산 입력→리셋 입력→CNTR/CNTRX 명령

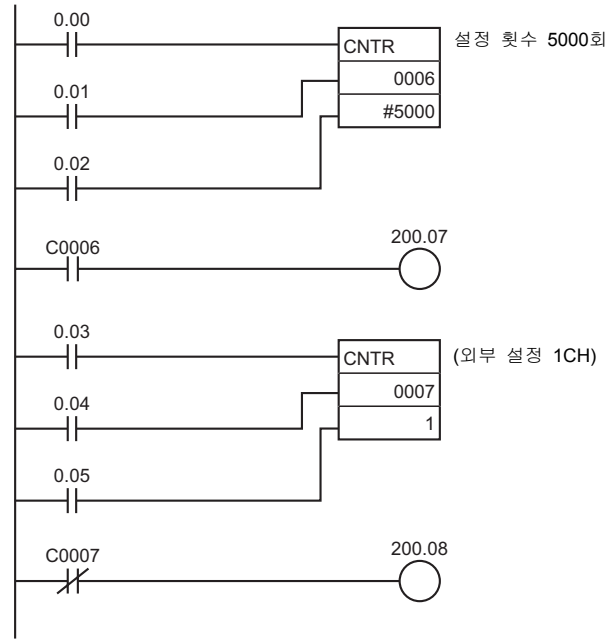
샘플 프로그램

리셋 입력 0.02가 ON이 되면 카운터 현재값은 0이 됩니다. 가산 카운트 입력 0.00이 OFF→ON으로 될 때마다 카운터 현재값이 1씩 증가합니다. 카운터 현재값이 설정값 3인 상태에서 가산 카운트 입력 0.00이 OFF→ON으로 되면 카운터 현재값은 0이 되고, 동시에 카운터 플래그가 ON이 됩니다. 감산 카운트 입력 0.01이 OFF→ON으로 될 때마다 카운터 현재값이 1씩 감소합니다. 카운터 현재값이 0인 상태에서 다음에 가산 카운트 입력 0.01이 OFF→ON으로 되면 카운터 현재값은 설정값 3이 되고, 동시에 카운터 플래그가 ON이 됩니다.

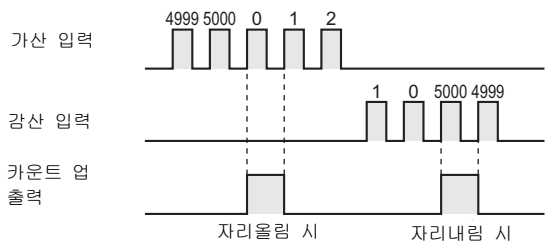


가산, 감산 카운트 입력은 신호 기동(OFF→ON) 시 1회 계산됩니다. 양쪽 입력이 동시에 ON일 때는 둘 다 계산하지 않습니다. 또한 리셋 입력이 ON일 때 현재값은 0이 되며 카운트 입력이 수신되지 않습니다.

CNTR0007의 동작은 입력 릴레이 0.03이 ON일 때 1CH의 내용에 따라 설정값이 결정됩니다. 이때 입력 릴레이 1CH에 외부 설정기를 연결하여 가변 설정의 가역 카운터로 사용할 수 있습니다.



명령	데이터
LD	0.00
LD	0.01
LD	0.02
CNTR(012)	0006
	#5000
LD	C0006
OUT	200.07
LD	0.03
LD	0.04
LD	0.05
CNTR(012)	0007
	1
LD NOT	C0007
OUT	200.08



CNR/CNRX

명령 이름	니모닉	동작 옵션	평션 번호	기능
타이머/카운터 리셋	CNR	@CNR	545	지정 범위의 타이머/카운터의 업 플래그를 리셋합니다.
	CNRX	@CNRX	547	

심볼	CNR	CNRX
	BCD 방식일 때 CNR D1 D2 D1: 타이머/카운터 번호 1 D2: 타이머/카운터 번호 2	BIN 방식일 때 CNRX D1 D2 D1: 타이머/카운터 번호 1 D2: 타이머/카운터 번호 2

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
D1	타이머/카운터 번호 1	TIMER/COUNTER*1	가변
D2	타이머/카운터 번호 2	TIMER/COUNTER*1	가변

*1: D1과 D2가 동일 변수일 때만 유효

D1: 타이머/카운터 번호 1
T0000~T4095 또는 C0000~C4095

D2:타이머/카운터 번호 2
T0000~T4095 또는 C0000~C4095

주: D1과 D2는 동일 영역 종류(타이머 또는 카운터 중 하나)이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
D1	–	–	–	–	○	○	–	–	–	–	–	–	–	○	–	–	–	–
D2																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· D1의 타이머 번호/카운터 번호가 간접 IR 지정인 경우, 주소가 타이머 영역/카운터 번호가 아닐 때 ON · D2의 타이머 번호/카운터 번호가 간접 IR 지정인 경우, 주소가 타이머 영역/카운터 번호가 아닐 때 ON · D1과 D2가 동일 영역일 때 ON · 그 이외일 때 OFF

기능

번호 D1의 타이머/카운터에서 번호 D2의 타이머/카운터까지의 업 플래그를 리셋합니다. 동시에, 현재값을 최대값(BCD 방식일 때: #9999, BIN 방식일 때: #FFFF)으로 설정합니다. (번호 D1~D2의 타이머/카운터 명령이 실행되면 현재값에 설정값이 설정됩니다.)

사용상의 주의 사항

· 리셋 대상 타이머/카운터는 다음과 같습니다.

대상 명령		CNR 명령 실행 시의 동작
BCD	TIM(100ms 타이머) TIMH(10ms 타이머) TMHH(1ms 타이머) TTIM(적산 타이머) TIMW(블록 프로그램의 타이머 대기) TMHW(블록 프로그램의 고속 타이머 대기) CNT(카운터) CNTR(가역 카운터) CNTW(카운터 대기)	현재값을 최대값(BCD#9999)으로 하고 업 플래그를 OFF로 합니다.
	TIMU(0.1ms 타이머) TMUH(0.01ms 타이머)	업 플래그를 OFF로 합니다. (현재값 참조 불가)

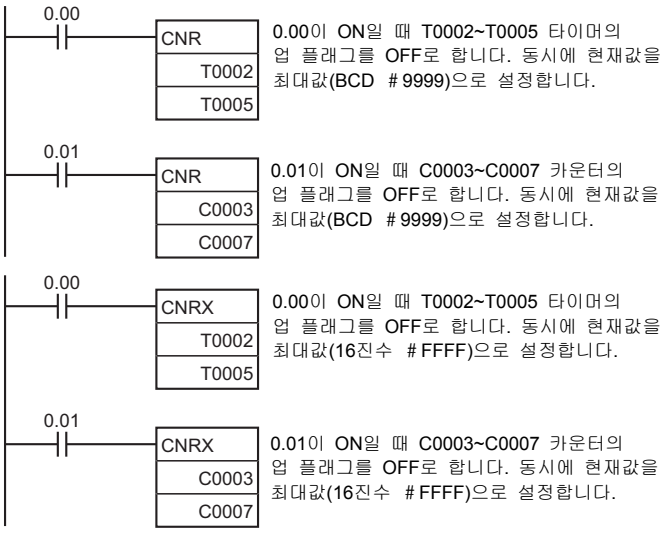
대상 명령		CNRX 명령 실행 시의 동작
BIN	TIMX(100ms 타이머) TIMHX(10ms 타이머) TMHHX(1ms 타이머) TTIMX(적산 타이머) TIMWX(블록 프로그램의 타이머 대기) TMHWX(블록 프로그램의 고속 타이머 대기) CNTX(카운터) CNTRX(가역 카운터) CNTWX(카운터 대기)	현재값을 최대값(16진수 #FFFF)으로 하고 업 플래그를 OFF로 합니다.
	TIMU(0.1ms 타이머) TMUHX(0.01ms 타이머)	업 플래그를 OFF로 합니다. (현재값 참조 불가)

단, TIML/TIMLX(장시간 타이머) 명령 및 MTIM/MTIMX(멀티 출력 타이머) 명령은 리셋 대상에서 제외됩니다.

· 이 명령은 명령에 대해 리셋을 실행하는 것이 아니라 명령이 사용하는 타이머/카운터의 현재값을 최대값으로 하여 업 플래그를 리셋하는 명령입니다. 명령에 대한 리셋과는 동작이 다르므로 주의하십시오. (예: TIM/TIMX 명령에 대한 리셋이 현재값 = 설정값으로 하여 업 플래그 = OFF로 하는데 비해, 이 CNR/CNRX 명령에 의한 리셋은 현재값 = 최대값으로 하고 업 플래그 = OFF로 합니다.)

· D1 > D2의 D1, D2를 지정한 경우, D1의 타이머/카운터 번호의 업 플래그만 리셋합니다.

샘플 프로그램



TRSET

명령 이름	니모닉	동작 옵션	펑션 번호	기능
타이머 리셋	TRSET	@TRSET	549	지정한 타이머를 리셋합니다.

심볼	TRSET
	

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	타이머 번호	TIMER	1

■ 영역 종류

내용	채널 주소							간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
N	–	–	–	–	○	–	–	–	–	–	–	–	○	–	–	–	–

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 타이머 번호가 간접 IR 지정인 경우, 주소가 타이머의 업 플래그 또는 현재값 주소가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	변하지 않음
부정 플래그	P_N	변하지 않음

기능

지정한 타이머를 리셋합니다.

데이터 비교 명령

니모닉	명령 이름	펑션 번호	페이지
= , <> , < , <= , > , >=	심볼 비교	300~328	3-120
=DT,<> DT,< DT,<= DT, > DT,>= DT	시각 비교	341~346	3-124
CMP	부호 없는 비교	020	3-128
CMPL	부호 없는 2워드 비교	060	
CPS	부호 있는 BIN 비교	114	3-131
CPSL	부호 있는 BIN 2워드 비교	115	
MCMP	다채널 비교	019	3-134
TCMP	테이블 일치	085	3-136
BCMP	부호 없는 테이블 간 비교	068	3-138
BCMP2	확장 테이블 간 비교	502	3-140
ZCP	영역 비교	088	3-143
ZCPL	2워드 영역 비교	116	
ZCPS	부호 있는 영역 비교	117	3-147
ZCPSL	부호 있는 2워드 영역 비교	118	

= , <> , < , <= , > , >=

명령 이름	니모닉	동작 옵션	펑션 번호	기능
심볼 비교	= , <> , < , <= , > , >=	–	300~328	CH 데이터 또는 상수를 비교하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다. 데이터 형식(부호 없음, 부호 있음), 데이터 길이(워드, 2워드)가 일치하는지 여부를 비교하는 것도 가능합니다.

부동 소수점 변환 명령에 대한 자세한 내용은 단정도 부동 소수점 데이터 비교를 참조하고, 배정도 부동 소수점 변환 명령에 대한 자세한 내용은 배정도 부동 소수점 변환 명령을 참조하십시오.

심볼	= , <> , < , <= , > , >=		
	LD(로드) 연결형	AND(직렬) 연결형	OR(병렬) 연결형
	 S1: 비교 데이터 1 S2: 비교 데이터 2	 S1: 비교 데이터 1 S2: 비교 데이터 2	 S1: 비교 데이터 1 S2: 비교 데이터 2

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식				크기	
		부호 없음	2워드, 부호 없음	부호 있음	2워드, 부호 있음	워드	2워드
S1	비교 데이터 1	UINT	UDINT	INT	DINT	1	2
S2	비교 데이터 2	UINT	UDINT	INT	DINT	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
데이터 길이: 워드	S1	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
	S2																		
데이터 길이: 2워드	S1	○	○	○	○	○	○	○	○	○	○	○	–	○	○	–	–	–	–
	S2																		

조건 플래그의 동작

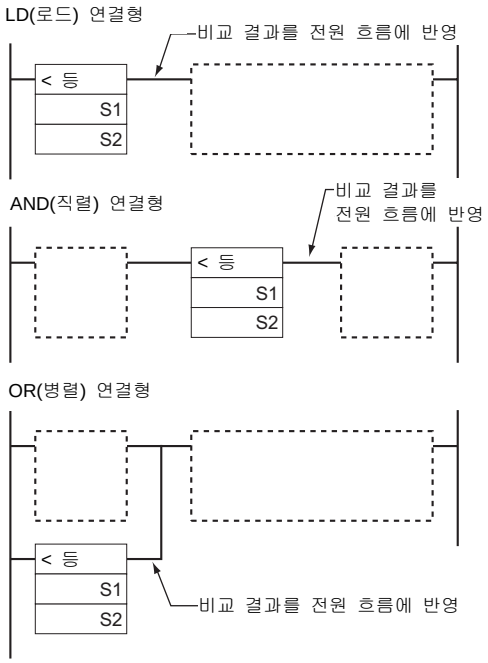
이름	라벨	내용	
		데이터 길이: 워드	데이터 길이: 2워드
에러 플래그	P_ER	변하지 않음*1	변하지 않음*1
> 플래그	P_GT	· 비교 결과가 S1 > S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1 > S2+1, S2일 때 ON · 그 이외일 때 OFF
≧ 플래그	P_GE	· 비교 결과가 S1≧S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1≧S2+1, S2일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 비교 결과가 S1 = S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1 = S2+1, S2일 때 ON · 그 이외일 때 OFF

이름	라벨	내용	
		데이터 길이: 워드	데이터 길이: 2워드
≠ 플래그	P_NE	· 비교 결과가 S1≠S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1≠S2+1, S2일 때 ON · 그 이외일 때 OFF
< 플래그	P_LT	· 비교 결과가 S1＜S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1＜S2+1, S2일 때 ON · 그 이외일 때 OFF
≦ 플래그	P_LE	· 비교 결과가 S1≦S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1≦S2+1, S2일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	변하지 않음*1	변하지 않음*1

*1: CS1/CJ1/CS1D(이중화 시스템용) CPU 유니트는 OFF

기능

S1과 S2를 부호 없음 또는 부호 있음으로 비교하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다.
LD, AND, OR 명령과 동일하게 취급하며, 각 명령 후에는 계속하여 다른 명령을 프로그래밍합니다.
LD형의 경우: 모선에 직접 연결할 수 있습니다.
AND형의 경우: 모선에 직접 연결할 수 없습니다.
OR형의 경우: 모선에 직접 연결할 수 있습니다.



■ 심볼과 옵션

비교 명령은 심볼과 옵션을 조합하여 72종류의 니모닉으로 표현됩니다.

심볼 (래더에서는 LD, AND, OR 없음)		옵션 (데이터 형식)		옵션 (데이터 길이)
LD=, AND=, OR=, LD<>, AND<>, OR<> LD<, AND<, OR< LD<=, AND<=, OR<= LD>, AND>, OR> LD>=, AND>=, OR>=	+	없음(부호 없음) S(부호 있음)	+	없음(워드) L(2워드)

기능	데이터 형식 및 길이	니모닉	이름	펑션 번호
S1≦S2일 때 참(ON)	부호 없는 워드 형식	LD =	LD 일치	300
		AND =	AND 일치	300
		OR =	OR 일치	300
	부호 없는 2워드 형식	LD =L	LD 2워드 일치	301
		AND =L	AND 2워드 일치	301
		OR =L	OR 2워드 일치	301
	부호 있는 워드 형식	LD =S	부호 있는 LD 일치	302
		AND =S	부호 있는 AND 일치	302
		OR =S	부호 있는 OR 일치	302
	부호 있는 2워드 형식	LD =SL	부호 있는 LD 2워드 일치	303
		AND=SL	부호 있는 AND 2워드 일치	303
		OR=SL	부호 있는 OR 2워드 일치	303

= , <> , < , <= , > , >=

기능	데이터 형식 및 길이	니모닉	이름	평션 번호
S1≠S2일 때 참(ON)	부호 없는 워드 형식	LD <>	LD 불일치	305
		AND <>	AND 불일치	305
		OR <>	OR 불일치	305
	부호 없는 2워드 형식	LD <>L	LD 2워드 불일치	306
		AND <>L	AND 2워드 불일치	306
		OR <>L	OR 2워드 불일치	306
	부호 있는 워드 형식	LD <>S	부호 있는 LD 불일치	307
		AND <>S	부호 있는 AND 불일치	307
		OR <>S	부호 있는 OR 불일치	307
	부호 있는 2워드 형식	LD <>SL	부호 있는 LD 2워드 불일치	308
		AND <>SL	부호 있는 AND 2워드 불일치	308
		OR <>SL	부호 있는 OR 2워드 불일치	308
S1＜S2일 때 참(ON)	부호 없는 워드 형식	LD <	LD 미만	310
		AND <	AND 미만	310
		OR <	OR 미만	310
	부호 없는 2워드 형식	LD <L	LD 2워드 미만	311
		AND <L	AND 2워드 미만	311
		OR <L	OR 2워드 미만	311
	부호 있는 워드 형식	LD<S	부호 있는 LD 미만	312
		AND <S	부호 있는 AND 미만	312
		OR<S	부호 있는 OR 미만	312
	부호 있는 2워드 형식	LD <SL	부호 있는 LD 2워드 미만	313
		AND <SL	부호 있는 AND 2워드 미만	313
		OR <SL	부호 있는 OR 2워드 미만	313
S1≦S2일 때 참(ON)	부호 없는 워드 형식	LD <=	LD 이하	315
		AND <=	AND 이하	315
		OR <=	OR 이하	315
	부호 없는 2워드 형식	LD<=L	LD 2워드 이하	316
		AND<=L	AND 2워드 이하	316
		OR<=L	OR 2워드 이하	316
	부호 있는 워드 형식	LD <=S	부호 있는 LD 이하	317
		AND <=S	부호 있는 AND 이하	317
		OR <=S	부호 있는 OR 이하	317
	부호 있는 2워드 형식	LD <=SL	부호 있는 LD 2워드 이하	318
		AND <=SL	부호 있는 AND 2워드 이하	318
		OR<=SL	부호 있는 OR 2워드 이하	318
S1＞S2일 때 참(ON)	부호 없는 워드 형식	LD >	LD 초과	320
		AND >	AND 초과	320
		OR >	OR 초과	320
	부호 없는 2워드 형식	LD >L	LD 2워드 초과	321
		AND >L	AND 2워드 초과	321
		OR >L	OR 2워드 초과	321
	부호 있는 워드 형식	LD >S	부호 있는 LD 초과	322
		AND >S	부호 있는 AND 초과	322
		OR >S	부호 있는 OR 초과	322
	부호 있는 2워드 형식	LD >SL	부호 있는 LD 2워드 초과	323
		AND>SL	부호 있는 AND 2워드 초과	323
		OR>SL	부호 있는 OR 2워드 초과	323
S1≧S2일 때 참(ON)	부호 없는 워드 형식	LD >=	LD 이상	325
		AND>=	AND 이상	325
		OR >=	OR 이상	325
	부호 없는 2워드 형식	LD >=L	LD 2워드 이상	326
		AND >=L	AND 2워드 이상	326
		OR >=L	OR 2워드 이상	326
	부호 있는 워드 형식	LD >=S	부호 있는 LD 이상	327
		AND>=S	부호 있는 AND 이상	327
		OR>=S	부호 있는 OR 이상	327
	부호 있는 2워드 형식	LD>=SL	부호 있는 LD 2워드 이상	328
		AND>=SL	부호 있는 AND 2워드 이상	328
		OR >=SL	부호 있는 OR 2워드 이상	328

- 부호 없는 비교 명령(S 옵션 없음)에서는 부호 없는 데이터(10진수 &0~&65535 또는 16진수 #0000~FFFF) 및 BCD 데이터를 사용할 수 있습니다.
- 부호 있는 비교 명령(S 옵션 있음)에서는 부호 있는 데이터(10진수 -32768~+32767 또는 16진수 #0000~FFFF)를 사용할 수 있습니다.

힌트

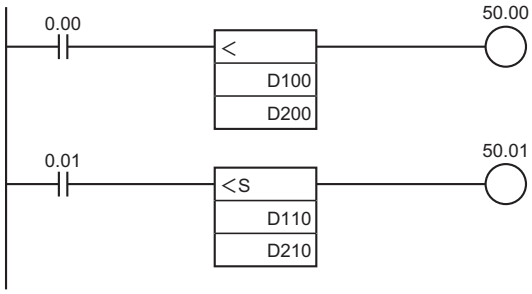
- CMP 명령이나 CMPL 명령과 달리, 비교 결과가 직접 다음 단계 회로에 입력 조건으로 반영되므로 조건 플래그를 전송할 필요가 없으며 프로그램을 간결하게 작성할 수 있습니다.

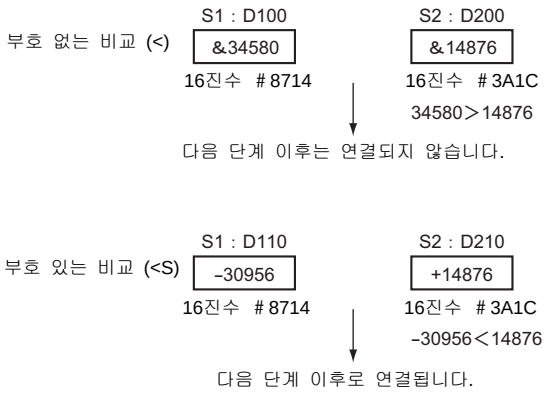
사용상의 주의 사항

- 이 명령의 최종 단계에는 출력계 명령(OUT계 명령 및 다음 단계 연결형 명령 이외의 응용 명령)을 추가하십시오.
- 회로의 최종 단계에는 이 명령을 사용할 수 없습니다.

샘플 프로그램

■ AND 연결형 < 명령, <S 명령의 경우





- < 명령의 동작

  0.00이 ON일 때 데이터 메모리 D100의 데이터와 D200의 데이터가 부호 없는 BIN 4자리로 비교됩니다.
  비교 결과가 (D100의 데이터)＜(D200의 데이터)인 경우, 다음 단계 이후가 연결되어 출력 릴레이 50.00이 ON됩니다.
  비교 결과가 (D100의 데이터)＜(D200의 데이터)가 아닌 경우는 다음 단계 이후로 연결되지 않습니다.

- <S 명령의 동작

  0.01이 ON일 때 데이터 메모리 D110의 데이터와 D210의 데이터가 부호 있는 BIN 4자리로 비교됩니다.
  비교 결과가 (D110의 데이터)＜(D210의 데이터)인 경우, 다음 단계 이후가 연결되어 출력 릴레이 50.01이 ON됩니다.
  비교 결과가 (D110의 데이터)＜(D210의 데이터)가 아닌 경우는 다음 단계 이후로 연결되지 않습니다.

=DT , <> DT , < DT , <= DT , > DT , >= DT

명령 이름	니모닉	동작 옵션	평션 번호	기능
시각 비교	=DT <> DT < DT <= DT > DT >= DT	–	341 342 343 344 345 346	2개의 시각 데이터(BCD 데이터)를 비교하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다.

심볼	=DT , <> DT , < DT , <= DT , > DT , >= DT



특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

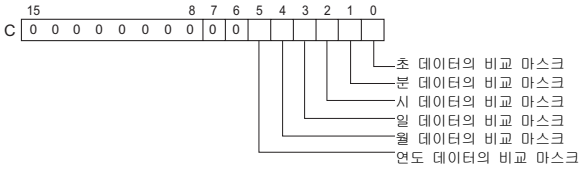
피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	WORD	1
S1	현재 시각 데이터 하위 CH 번호	WORD	3
S2	비교 시각 데이터 하위 CH 번호	WORD	3

C: 제어 데이터

연, 월, 일, 시, 분, 초 중 어느 것을 비교 마스크(대상에서 제외)로 할지 비트 00~05를 사용하여 각각 지정합니다. 전체 마스크(비트 00~05가 모두 1)의 경우, 명령이 실행되지 않고 다음 단계 이후에 연결되지 않습니다.

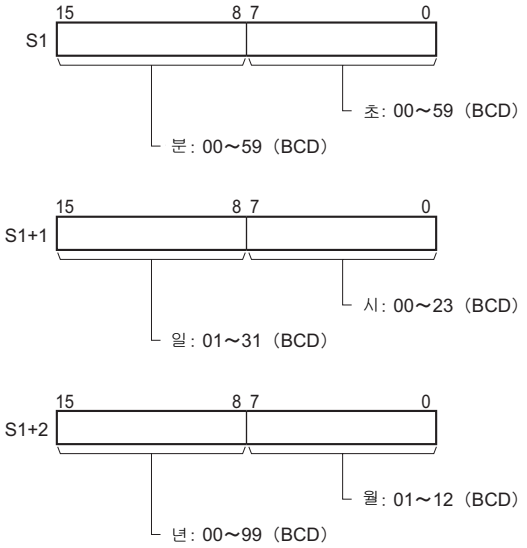
이때, 에러 플래그는 ON이 됩니다.



S1: 현재 시각 데이터 하위 CH 번호

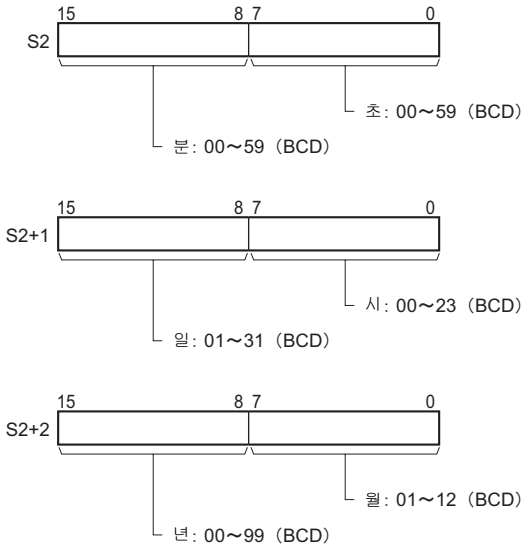
현재 시각(연, 월, 일, 시, 분, 초) 데이터를 아래의 S1~S1+2에 저장합니다.

CPU 유니트의 내부 시계 데이터를 비교 기준으로 하는 경우, S1 = A351CH로 하여 CPU 유니트의 내부 시계 데이터(A351~353CH)를 지정합니다.



S2: 비교 시각 데이터 하위 CH 번호

비교 시각(연, 월, 일, 시, 분, 초) 데이터를 아래의 S2~S2+2에 저장합니다.



주: 연도는 서력 아래 2자리를 사용합니다. 또한 97년과 98년을 경계로 98이면 서기 1998년, 97이면 서기 2097년으로 지정됩니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	–	–	–	–	○	–	–	○	–	–	–	–
S1,S2							○	○	○	○	–							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 전체 마스크일 때 ON · 그 이외일 때 OFF
> 플래그	P_GT	· 비교 결과가 S1＞S2일 때 ON · 그 이외일 때 OFF
≧ 플래그	P_GE	· 비교 결과가 S1≧S2일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 비교 결과가 S1=S2일 때 ON · 그 이외일 때 OFF
≠ 플래그	P_NE	· 비교 결과가 S1≠S2일 때 ON · 그 이외일 때 OFF
< 플래그	P_LT	· 비교 결과가 S1＜S2일 때 ON · 그 이외일 때 OFF
≦ 플래그	P_LE	· 비교 결과가 S1≦S2일 때 ON · 그 이외일 때 OFF

=DT , <> DT , < DT , <= DT , > DT , >= DT

기능

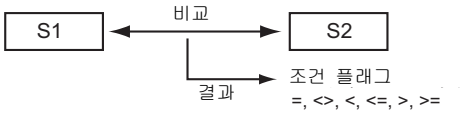
C에서 0(비교 대상)으로 지정한 시각 항목(연, 월, 일, 시, 분, 초)에 대해 S1과 S2를 BCD로 비교하고, 결과가 참이면 다음 단계 이후로 연결합니다. 이와 동시에 조건 플래그(=, <>, <, <=, >, >=)에도 반영합니다.

시각 비교 명령은 18종류의 니모닉으로 표현됩니다.

C에서 1(비교 마스크)로 지정된 항목은 비교를 수행하지 않습니다.

또한 명령 실행 후 각 조건 플래그가 다음과 같이 비교 결과에 따라 ON/OFF됩니다.

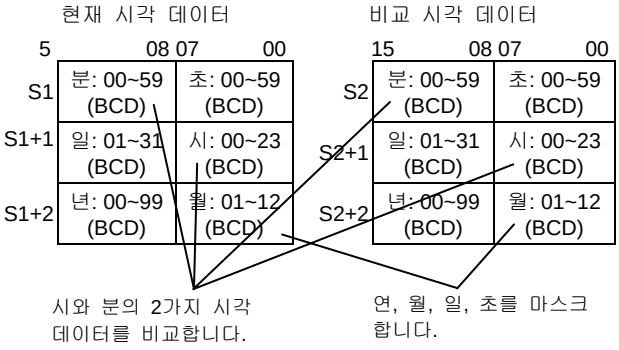
비교 결과	=	<>	<	<=	>	>=
S1=S2	ON	OFF	OFF	ON	OFF	ON
S1>S2	OFF	ON	OFF	OFF	ON	ON
S1<S2	OFF	ON	ON	ON	OFF	OFF



■ 시각 데이터의 비교 마스크

시각 데이터의 비교 마스크란 비교 대상 이외의 시각 데이터를 마스크(숨김)하는 기능을 말합니다. 이를 통해, C에서 0(비교 대상)으로 지정한 '연, 월, 일, 시, 분, 초'의 6개 시각 데이터를 개별적으로 비교할 수 있습니다.

예: C=#39(2진수에서는 111001이므로, 년: 1, 월: 1, 일: 1, 시: 0, 분: 0, 초: 1)인 경우 시, 분 데이터만 비교 대상이 됩니다(0 = 비교 대상, 1 = 마스크).
따라서 매일 a시 b분에 ON되는 캘린더 타이머 기능을 수행할 수 있습니다.



힌트

- 기존의 데이터 비교 명령에서는 16비트 단위로 데이터를 비교합니다. 시각 비교 명령을 사용하면 시각 데이터를 8비트 단위로 한정하여 비교할 수 있습니다.
- CPU 유니트의 내부 시계 데이터는 다음의 특수 보조 릴레이에 BCD로 저장됩니다.
- 위의 내부 시계 데이터는 주변 툴(프로그래밍 콘솔 포함)에 의한 설정, DATE(시계 보정) 명령 실행, FINS 명령의 시간 정보 쓰기(#0702) 중 하나를 통해 설정합니다.

비교

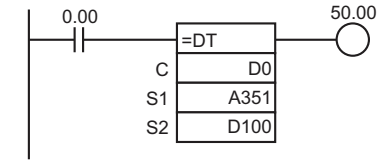
주소		내용
채널	비트	
A351CH	00~07	초(00~59) (BCD)
	08~15	분(00~59) (BCD)
A352CH	00~07	시(00~23) (BCD)
	08~15	일(01~31) (BCD)
A353CH	00~07	월(01~12) (BCD)
	08~15	년(00~99) (BCD)

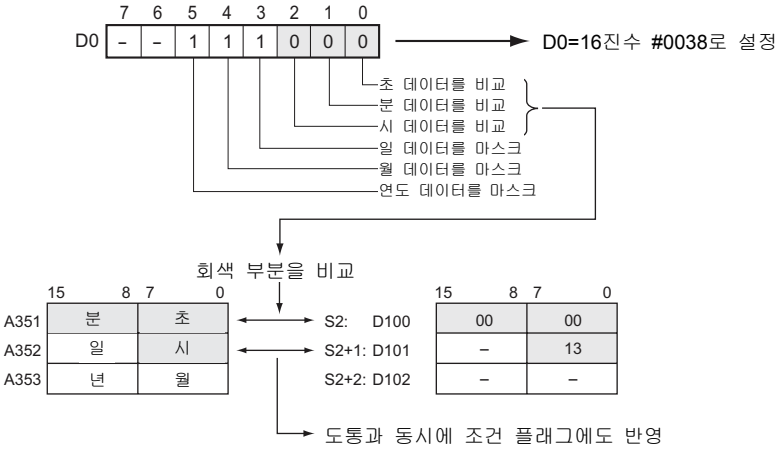
사용상의 주의 사항

- 이 명령의 최종 단계에는 출력계 명령(OUT계 명령 및 다음 단계 연결형 명령 이외의 응용 명령)을 추가하십시오.
- 회로의 최종 단계에는 이 명령을 사용할 수 없습니다.

샘플 프로그램

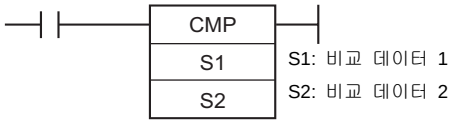
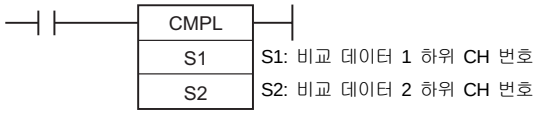
0.00이 ON인 경우 시각이 13시 0분 0초이면 50.00을 ON으로 합니다.
CPU 유니트 내장 시계의 A351~A352의 현재 시각과 D100~D102의 설정 시각을 시, 분, 초로 비교합니다.





CMP/CMPL

명령 이름	니모닉	동작 옵션	평션 번호	기능
부호 없는 비교	CMP	!CMP	020	CH 데이터 또는 상수를 부호 없는 BIN 16비트 (16진수 4자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.
부호 없는 2워드 비교	CMPL	–	060	2CH분의 CH 데이터 또는 상수를 부호 없는 BIN 32비트(16진수 8자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.

심볼	CMP	CMPL
		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		CMP	CMPL	CMP	CMPL
S1	CMP: 비교 데이터 1 CMPL: 비교 데이터 1 하위 CH 번호	UINT	UDINT	1	2
S2	CMP: 비교 데이터 2 CMPL: 비교 데이터 2 하위 CH 번호	UINT	UDINT	1	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
CMP S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
CMPL S1,S2												–	○					

조건 플래그의 동작

이름	라벨	내용	
		CMP	CMPL
에러 플래그	P_ER	변하지 않음*1	변하지 않음*1
> 플래그	P_GT	· 비교 결과가 S1 > S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1 > S2+1, S2일 때 ON · 그 이외일 때 OFF
≧ 플래그	P_GE	· 비교 결과가 S1≧S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1≧S2+1, S2일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 비교 결과가 S1=S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1 = S2+1, S2일 때 ON · 그 이외일 때 OFF

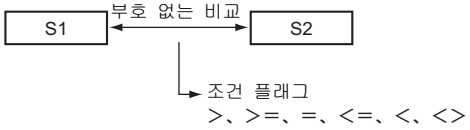
이름	라벨	내용	
		CMP	CMPL
≠ 플래그	P_NE	· 비교 결과가 S1≠S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1≠S2+1, S2일 때 ON · 그 이외일 때 OFF
< 플래그	P_LT	· 비교 결과가 S1<S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1<S2+1, S2일 때 ON · 그 이외일 때 OFF
≦ 플래그	P_LE	· 비교 결과가 S1≦S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1≦S2+1, S2일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	변하지 않음*1	변하지 않음*1

*1: CS1/CJ1/CS1D(이중화 시스템용) CPU 유니트는 OFF

■ CMP 명령 실행 후 >, > =, =, < =, <, < >의 각 조건 플래그가 ON/OFF됩니다.

비교 결과	>	> =	=	< =	<	< >
S1 > S2	ON	ON	OFF	OFF	OFF	ON
S1 = S2	OFF	ON	ON	ON	OFF	OFF
S1 < S2	OFF	OFF	OFF	ON	ON	ON

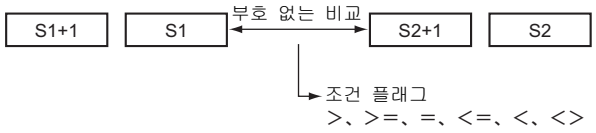
주: −은 경우에 따라 다름



■ CMPL 명령 실행 후 >, > =, =, < =, <, < >의 각 조건 플래그가 ON/OFF됩니다.

비교 결과	>	> =	=	< =	<	< >
S1+1,S1 > S2+1,S2	ON	ON	OFF	OFF	OFF	ON
S1+1,S1 = S2+1,S2	OFF	ON	ON	ON	OFF	OFF
S1+1,S1 < S1+1,S1	OFF	OFF	OFF	ON	ON	ON

주: −은 경우에 따라 다름



기능

■ CMP

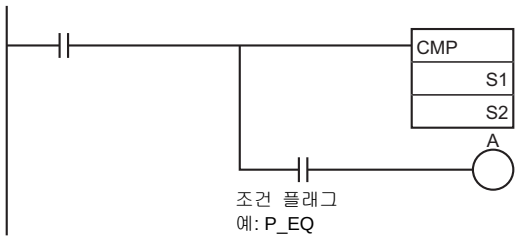
S1과 S2를 부호 없는 BIN(16진수 4자리)으로 비교하여 결과를 조건 플래그(>, > =, =, < >, <, < =)에 반영합니다.

■ CMPL

S1과 S2를 2워드 데이터로 처리하여 부호 없는 BIN(16진수 8자리)으로 비교하여 결과를 조건 플래그(>, > =, =, < >, <, < =)에 반영합니다.

사용상의 주의 사항

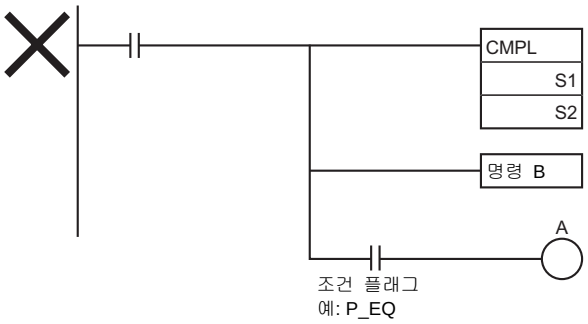
· 이 명령을 실행하면 조건 플래그에 비교 결과가 반영되므로 예와 같이 CMP(CMPL) 명령과 동일한 입력 조건의 출력 분기에서 결과를 입력하십시오.



S1과 S2가 일치하면 P_EQ 플래그가 ON이 되고 A가 ON이 됩니다.

CMP/CMPL

· 조건 플래그를 다른 명령 뒤에 배치하면 그 명령의 실행 결과에 의해 조건 플래그가 바뀌므로 조건 플래그는 반드시 CMP(CMPL) 명령 직후에 배치하십시오.

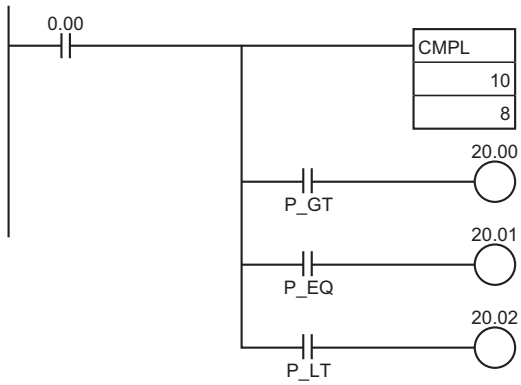


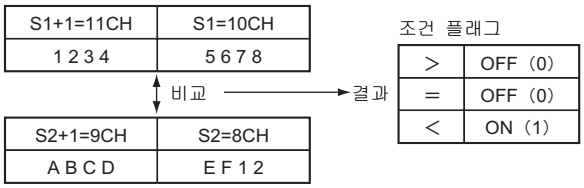
조건 플래그는 명령 B의 실행 결과를 수신합니다.

· CMP 명령은 매회 리프레시형 명령(!CMP)으로 사용할 수 있습니다.
매회 리프레시형 명령인 !CMP의 경우, S1 또는 S2(S1 및 S2도 가능)에 외부 I/O 할당된 입력 릴레이 영역을 지정합니다(단, 그룹 2 다점 입출력 유니트, 고기능 I/O 유니트의 다점 입출력 유니트, SYSBUS 원격 I/O 자국 장치의 유니트는 제외).
명령 실행 시 S1 또는 S2의 값을 IN 리프레시하고 그 값을 비교합니다.

샘플 프로그램

· 입력 릴레이 0.00이 ON일 때 11, 10CH와 9, 8CH의 데이터 내용을 16진수 8자리로 비교합니다.
· 비교 결과, 11, 10CH가 크면 20.00이 ON되고, 같으면 20.01이 ON되며, 11, 10CH가 작으면 20.02가 ON됩니다.





CPS/CPSL

명령 이름	니모닉	동작 옵션	평션 번호	기능
부호 있는 BIN 비교	CPS	!CPS	114	CH 데이터 또는 상수를 부호 있는 BIN 16비트(최상위 비트를 부호 비트로 하는 16진수 4자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.
부호 있는 BIN 2워드 비교	CPSL	−	115	2CH분의 CH 데이터 또는 상수를 부호 있는 2워드 BIN 32비트(최상위 비트를 부호 비트로 하는 16진수 8자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.

심볼	CPS	CPSL
	CPS S1 S2 S1: 비교 데이터 1 S2: 비교 데이터 2	CPSL S1 S2 S1: 비교 데이터 1 하위 CH 번호 S2: 비교 데이터 2 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		CPS	CPSL	CPS	CPSL
S1	CPS: 비교 데이터 1 CPSL: 비교 데이터 1 하위 CH 번호	INT	DINT	1	2
S2	CPS: 비교 데이터 2 CPSL: 비교 데이터 2 하위 CH 번호	INT	DINT	1	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
CPS S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	−	○	−	−	−	−
CPSL S1,S2												−						

조건 플래그의 동작

이름	라벨	내용	
		CPS	CPSL
에러 플래그	P_ER	변하지 않음*1	변하지 않음*1
> 플래그	P_GT	· 비교 결과가 S1 > S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1 > S2+1, S2일 때 ON · 그 이외일 때 OFF
≧ 플래그	P_GE	· 비교 결과가 S1≧S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1≧S2+1, S2일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 비교 결과가 S1=S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1 = S2+1, S2일 때 ON · 그 이외일 때 OFF

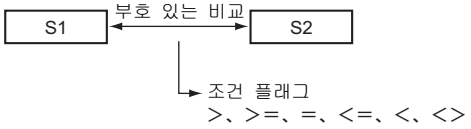
CPS/CPSL

이름	라벨	내용	
		CPS	CPSL
≠ 플래그	P_NE	· 비교 결과가 S1≠S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1≠S2+1, S2일 때 ON · 그 이외일 때 OFF
< 플래그	P_LT	· 비교 결과가 S1<S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1<S2+1, S2일 때 ON · 그 이외일 때 OFF
≦ 플래그	P_LE	· 비교 결과가 S1≦S2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S1+1, S1≦S2+1, S2일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	변하지 않음*1	변하지 않음*1

*1: CS1/CJ1/CS1D(이중화 시스템용) CPU 유니트는 OFF

■ CPS 명령 실행 후 >, > =, =, < =, <, < >의 각 조건 플래그가 ON/OFF됩니다.

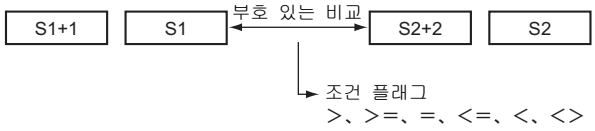
비교 결과	>	> =	=	< =	<	< >
S1 > S2	ON	ON	OFF	OFF	OFF	ON
S1 = S2	OFF	ON	ON	ON	OFF	OFF
S1 < S2	OFF	OFF	OFF	ON	ON	ON



주: −은 경우에 따라 다름. 비교 데이터 1, 2에서 지정할 수 있는 범위는 부호 있는 BIN(10진수 -32768~32767 또는 16진수 #8000~7FFF)입니다.

■ CPSL 명령 실행 후 >, > =, =, < =, <, < >의 각 조건 플래그가 ON/OFF됩니다.

비교 결과	>	> =	=	< =	<	< >
S1+1,S1 > S2+1,S2	ON	ON	OFF	OFF	OFF	ON
S1+1,S1 = S2+1,S2	OFF	ON	ON	ON	OFF	OFF
S1+1,S1 < S1+1,S1	OFF	OFF	OFF	ON	ON	ON



주: −은 경우에 따라 다름. 비교 데이터 1, 2에서 지정할 수 있는 범위는 부호 있는 BIN(10진수 -2147483648~2147483647 또는 16진수 #80000000~7FFFFFFF)입니다.

기능

■ CPS

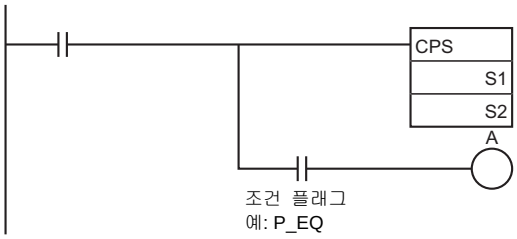
S1과 S2를 부호 없는 BIN(최상위 비트를 부호 비트로 하는 16진수 4자리)으로 비교하여 결과를 조건 플래그(>, > =, =, < >, <, < =)에 반영합니다.

■ CPSL

S1과 S2를 부호 있는 2워드 BIN(최상위 비트를 부호 비트로 하는 16진수 8자리)으로 비교하여 결과를 조건 플래그(>, > =, =, < >, <, < =)에 반영합니다.

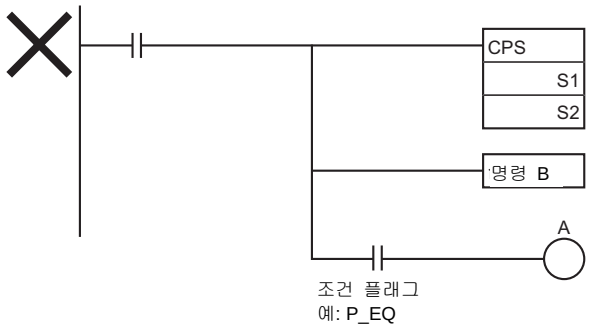
사용상의 주의 사항

· 이 명령을 실행하면 조건 플래그에 비교 결과가 반영되므로 예와 같이 CPS 명령과 동일한 입력 조건의 출력 분기에서 결과를 입력하십시오.



S1과 S2가 일치하면 P_EQ 플래그가 ON이 되고 A가 ON이 됩니다.

· 조건 플래그를 다른 명령 뒤에 배치하면 그 명령의 실행 결과에 의해 조건 플래그가 바뀌므로 조건 플래그는 반드시 CPS 명령 직후에 배치하십시오.

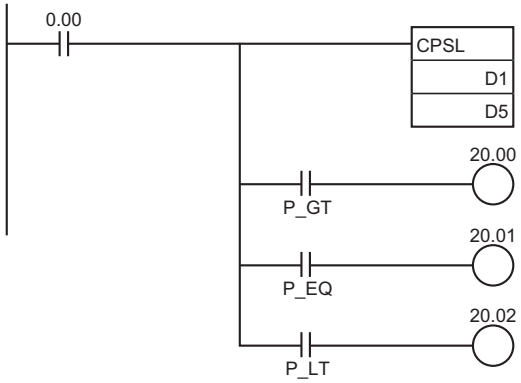


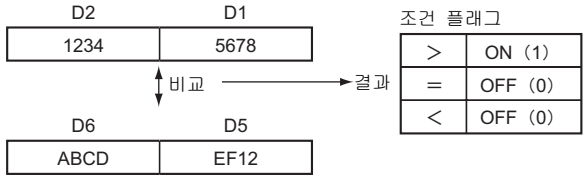
조건 플래그는 명령 B의 실행 결과를 수신합니다.

· CPS 명령은 매회 리프레시형 명령(!CPS)으로 사용할 수 있습니다.
매회 리프레시형 명령(!CPS)의 경우, S1 또는 S2(S1 및 S2도 가능)에 외부 I/O 할당된 입력 릴레이 영역을 지정합니다(단, 그룹 2 다점 입출력 유니트, 고기능 I/O 유니트의 다점 입출력 유니트, SYSBUS 원격 I/O 자국 장치의 유니트는 제외).
명령 실행 시 S1 또는 S2의 값을 IN 리프레시하고 그 값을 비교합니다.

샘플 프로그램

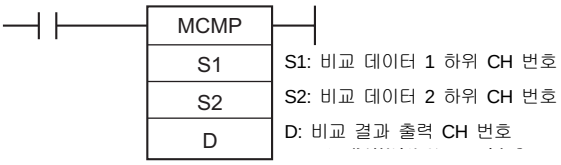
0.00이 ON일 때 데이터 메모리 D2, D1의 데이터 내용과 D6, D5의 데이터 내용이 부호 있는 BIN으로 비교됩니다.
비교 결과, D2, D1의 데이터 내용이 크면 조건 플래그 >가 ON이 되고 출력 릴레이 20.00이 ON됩니다.
D2, D1의 데이터 내용과 D6, D5의 데이터 내용이 같으면 =가 ON이 되고 20.01이 ON됩니다.
D2, D1의 데이터 내용보다 D6, D5의 데이터 내용이 크면 <가 ON이 되고 20.02가 ON됩니다.





MCMP

명령 이름	니모닉	동작 옵션	평션 번호	기능
다채널 비교	MCMP	@MCMP	019	16CH분의 데이터를 CH 단위로 비교하여 지정된 CH의 해당 비트에 일치하는지 여부의 비교 결과를 출력합니다.

	MCMP
심볼	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S1	비교 데이터 1 하위 CH 번호	WORD	16
S2	비교 데이터 2 하위 CH 번호	WORD	16
D	비교 결과 출력 CH 번호	UINT	1

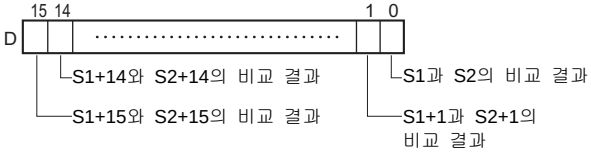
S1: 비교 데이터 1 하위 CH 번호

S1	비교 데이터 0
S1+1	비교 데이터 1
≀	≀
S1+15	비교 데이터 15

S2: 비교 데이터 2 하위 CH 번호

S2	비교 데이터 0
S2+1	비교 데이터 1
≀	≀
S2+15	비교 데이터 15

D: 비교 결과 출력 CH 번호



주: S1~S1+15 및 S2~S2+15는 각각 동일 영역 종류이어야 합니다.

■ 영역 종류

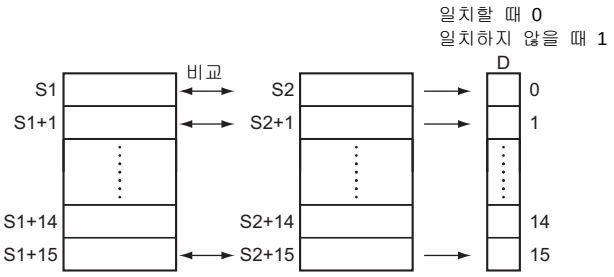
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S1,S2	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D												○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 비교 결과가 0일 때(비교 결과, 모든 데이터가 일치할 때) ON · 그 이외일 때 OFF

기능

S1로 지정한 16채널(워드)분의 데이터와 S2로 지정한 16채널분의 데이터를 각각 채널 단위로 비교하고, 일치한 경우는 0, 일치하지 않은 경우는 1을 D의 해당 비트에 출력합니다.
S1과 S2를 비교하여 일치하면 DCH의 비트 0에 0을 출력하고 일치하지 않으면 1을 출력합니다.
S1+1과 S2+1을 비교하여 일치하면 DCH의 비트 1에 0을 출력하고 일치하지 않으면 1을 출력합니다. 이하 동일.
S1+15와 S2+15를 비교하여 일치하면 DCH의 비트 15에 0을 출력하고 일치하지 않으면 1을 출력합니다.

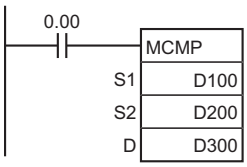


힌트

· 이 명령 실행 후 = 플래그가 ON되면 16채널분의 데이터가 일치함을 나타냅니다.

샘플 프로그램

0.00이 ON일 때 D100~D115와 D200~D215의 각 16CH를 비교하여 일치하는 경우 0, 일치하지 않는 경우 1의 결과를 D300의 비트 0~15에 저장합니다.

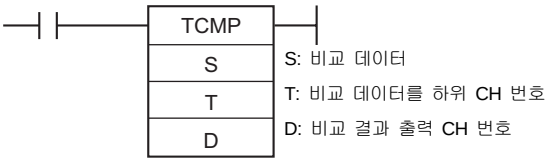


S1 : D100		S2 : D200		D : D300 ↓ 비트	
D100	1 2 3 4	D200	1 2 3 4	0	0
D101	5 6 7 8	D201	9 A B C	1	1
D102	9 A B C	D202	5 6 7 8	1	2
D103	D E F 0	D203	D E F 0	0	3
D104	0 2 0 0	D204	0 2 0 0	0	4
D105	1 2 3 4	D205	5 6 7 8	1	5
D106	5 6 7 8	D206	9 A B C	1	6
D107	9 A B C	D207	D E F 0	1	7
D108	D E F 0	D208	1 2 3 4	1	8
D109	1 2 3 4	D209	5 6 7 8	1	9
D110	5 6 7 8	D210	0 2 0 0	1	10
D111	9 A B C	D211	1 2 3 4	1	11
D112	0 2 0 0	D212	0 2 0 0	0	12
D113	1 2 3 4	D213	5 6 7 8	1	13
D114	5 6 7 8	D214	9 A B C	1	14
D115	9 A B C	D215	1 2 3 4	1	15

TCMP

명령 이름	니모닉	동작 옵션	평션 번호	기능
테이블 일치	TCMP	@TCMP	085	비교 데이터 1CH를 비교 테이블 16CH분의 데이터와 각각 일치하는지 비교하여 출력 CH의 해당 비트에 1을 출력합니다.

심볼	TCMP



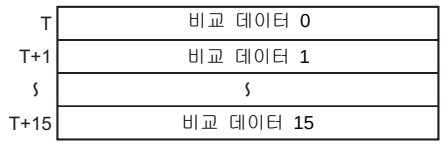
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

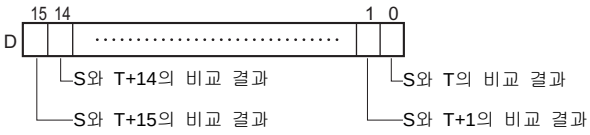
피연산자 설명

피연산자	내용	데이터 형식	크기
S	비교 데이터	WORD	1
T	비교 테이블 하위 CH 번호	WORD	16
D	비교 결과 출력 CH 번호	UINT	1

T: 비교 테이블 하위 CH 번호



D: 비교 결과 출력 CH 번호



■ 영역 종류

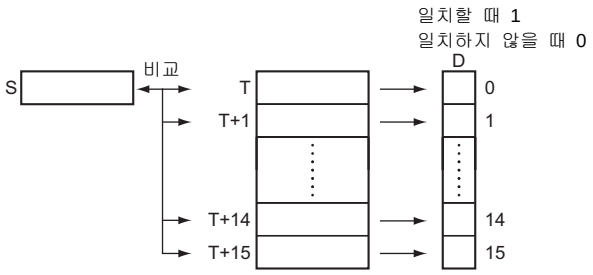
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
T											–	–						
D												○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 비교 결과가 0일 때(비교 테이블과 비교한 결과, S의 데이터가 모두 일치하지 않을 때) ON · 그 이외일 때 OFF

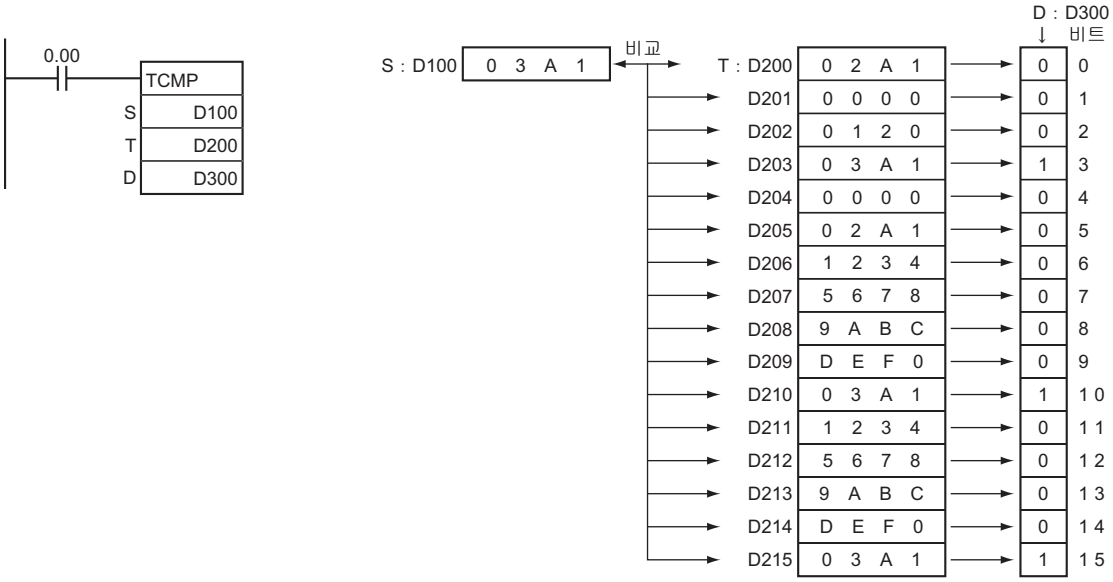
기능

S로 지정한 1채널(워드)과 T~T+15로 지정한 16채널(워드)분의 데이터를 각각 비교하고, 일치한 경우는 1, 일치하지 않은 경우는 0을 D의 해당 비트에 출력합니다.
S와 T를 비교하여 일치하면 DCH의 비트 0에 1을 출력하고 일치하지 않으면 0을 출력합니다.
S와 T+1을 비교하여 일치하면 DCH의 비트 1에 1을 출력하고 일치하지 않으면 0을 출력합니다.
이하 동일합니다.
S와 T+15를 비교하여 일치하면 DCH의 비트 15에 1을 출력하고 일치하지 않으면 0을 출력합니다.



샘플 프로그램

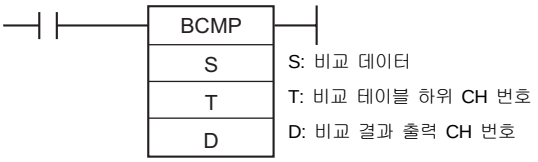
0.00이 ON일 때 D100과 D200~D215를 비교하여 일치하는 경우 1, 일치하지 않는 경우 0을 D300의 비트 0~15에 저장합니다.



BCMP

명령 이름	니모닉	동작 옵션	평션 번호	기능
부호 없는 테이블 간 비교	BCMP	@BCMP	068	비교 데이터의 내용이 16쌍의 비교 데이터의 상하한 범위 안에 있는지 여부를 판별하여 범위 이내이면 출력 CH의 해당 비트에 1을 출력합니다.

심볼	BCMP



특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

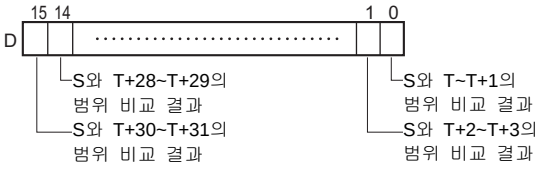
피연산자 설명

피연산자	내용	데이터 형식	크기
S	비교 데이터	WORD	1
T	비교 테이블 하위 CH 번호	WORD	32
D	비교 결과 출력 CH 번호	UINT	1

T: 비교 테이블 하위 CH 번호

T	하한값 0
T+1	상한값 0
T+2	하한값 1
T+3	상한값 1
∫	∫
T+30	하한값 15
T+31	상한값 15

D: 비교 결과 출력 CH 번호



■ 영역 종류

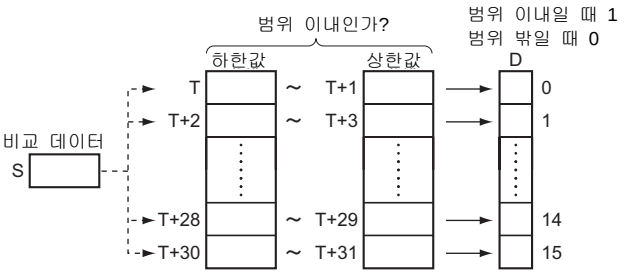
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
T											–	–						
D												○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 비교 결과가 0일 때(비교 테이블과 비교한 결과, S의 데이터가 모두 범위 밖일 때) ON · 그 이외일 때 OFF

기능

T로 지정한 32워드의 데이터를 16쌍의 상한값, 하한값 데이터로 처리하여 S가 각각의 범위 안(상한값, 하한값 포함)에 있는지 비교합니다. 비교 결과, 범위 안(일치 포함)이면 1을, 범위 밖(일치 포함 안 함)이면 0을 D의 해당 비트에 출력합니다.
T, T+2, ···, T+28, T+30이 하한값이 됩니다.
T+1, T+3, ···, T+29, T+31은 상한값이 됩니다.
S가 T(하한값)~T+1(상한값), T+2(하한값)~T+3(상한값), ···의 범위 안에 있는지 여부를 비교하여 결과를 DCH의 비트 0, 1, ···에 출력합니다.

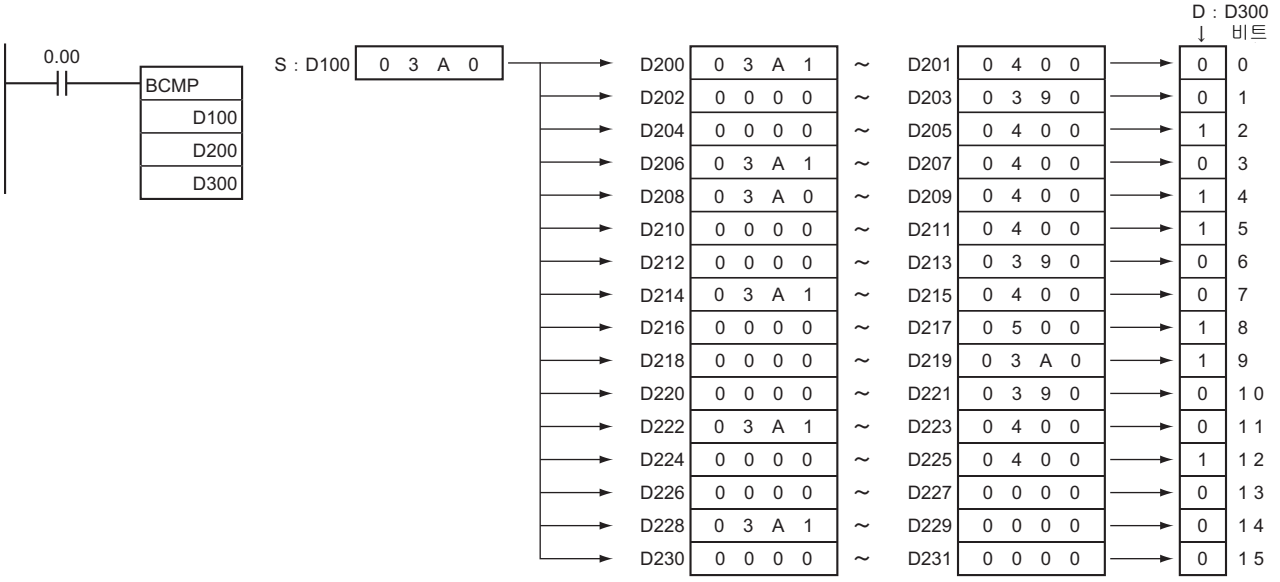


예:
T≦S≦T+1일 때 DCH의 비트 0에 1을 출력
T+2≦S≦T+3일 때 DCH의 비트 1에 1을 출력
이하 동일.
S＜T, T+1＜S일 때 DCH의 비트 0에 0을 출력
S＜T+2, T+3＜S일 때 DCH의 비트 1에 0을 출력
이하 동일.

주: 하한값과 상한값이 반대(T＞T+1 등)인 경우, 에러는 되지 않고, D의 해당 비트에 0(범위 밖)을 출력합니다.

샘플 프로그램

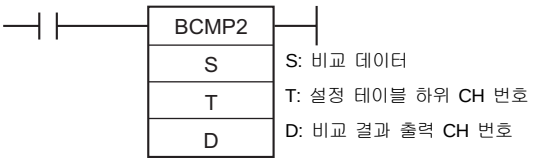
0.00이 ON일 때 D100의 내용이 D200의 내용을 하한값으로 하고 D201의 내용을 상한값으로 한 범위 내인 경우, D300의 비트 0에 1을 저장하고 범위 밖일 때는 0을 저장합니다. 동일하게 D100의 내용이 D202~D203, ···D214~D215의 범위 안에 있으면 1을, 범위 밖에 있으면 0을 D300의 비트 1···15에 저장합니다.



BCMP2

명령 이름	니모닉	동작 옵션	펑션 번호	기능
확장 테이블 간 비교	BCMP2	@BCMP2	502	비교 데이터의 내용이 최대 256개인 비교 데이터 상하한 범위 안에 있는지 여부를 판별하여 범위 이내이면 D~D+최대 15CH의 해당 비트에 1을 출력합니다.

심볼	BCMP2



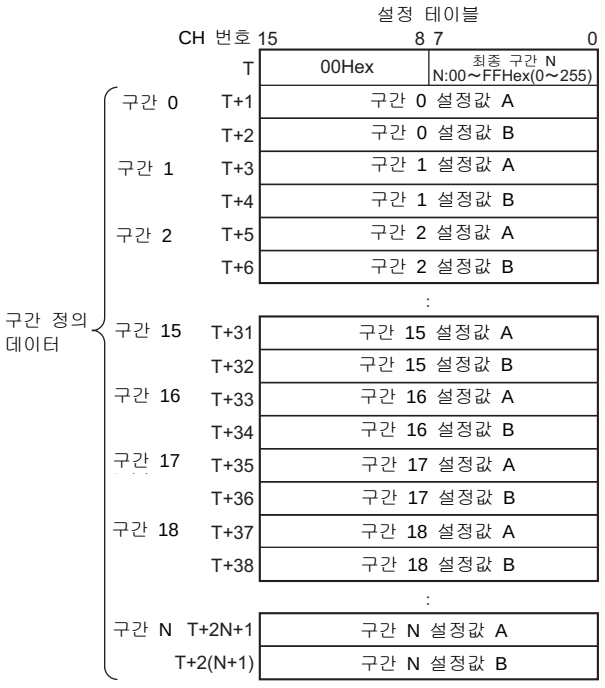
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

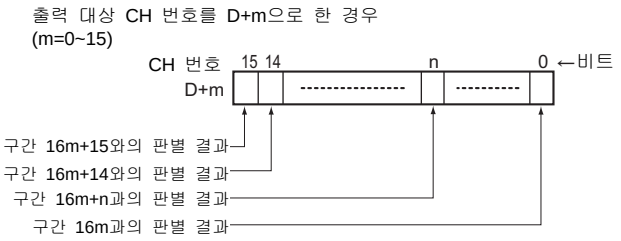
피연산자 설명

피연산자	내용	데이터 형식	크기
S	비교 데이터	WORD	1
T	설정 테이블 하위 CH 번호	WORD	가변
D	비교 결과 출력 CH 번호	WORD	가변

T: 설정 테이블 하위 CH 번호



D: 비교 결과 출력 CH 번호



■ 영역 종류

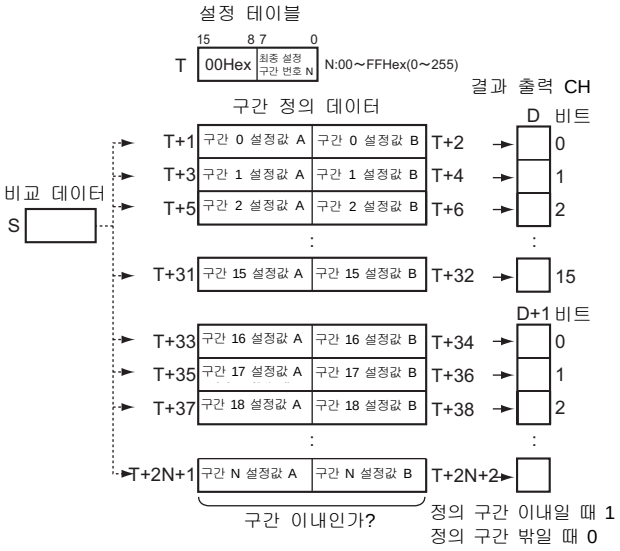
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
T											–	–						
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF

기능

비교 데이터 S가 T+1 이후 각 구간(최대 256개)의 설정 범위 안에 있는지 판별합니다. 판별 결과, 비교 데이터 S가 설정 범위(정의 구간) 안에 있으면 1을, 설정 범위 밖에 있으면 0을 D~D+최대 15CH의 해당 비트에 출력합니다. 또한 최종 구간 N은 T의 하위 바이트로 지정합니다(N = 0~255). 그리고 T의 상위 바이트는 반드시 0으로 하십시오.



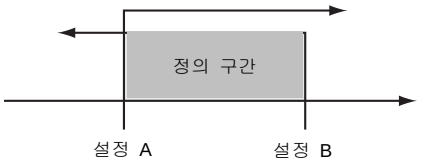
■ 구간의 개수

구간의 개수는 설정 테이블의 시작 채널로 설정합니다. 구간은 최대 256개까지 등록할 수 있습니다.

■ 구간 설정

정의 구간은 설정값 A와 설정값 B의 대소 관계에 따라 다음과 같이 결정됩니다.

· 설정값 A≦설정값 B인 경우
  설정값 A≦정의 구간≦설정값 B



· 설정값 A≦설정값 B인 경우
  정의 구간≦설정값 B, 설정값 A≦정의 구간



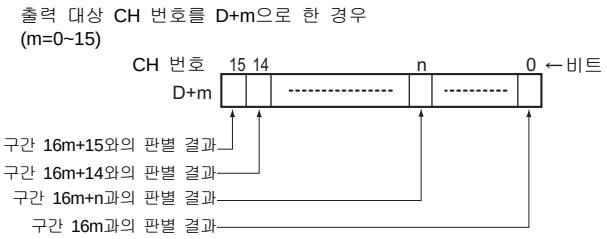
· T+1≦T+2인 경우
  T+1≦S≦T+2일 때 DCH의 비트 0에 1을 출력
  T+3≦S≦T+4일 때 DCH의 비트 1에 1을 출력
  S<T+5, T+6<S일 때 DCH의 비트 2에 0을 출력
  S<T+7, T+8<S일 때 DCH의 비트 3에 0을 출력

· T+1>T+2인 경우
  S≦T+2, T+1≦S일 때 DCH의 비트 0에 1을 출력
  S≦T+4, T+3≦S일 때 DCH의 비트 1에 1을 출력
  T+6<S<T+5일 때 DCH의 비트 2에 0을 출력
  T+8<S<T+7일 때 DCH의 비트 3에 0을 출력

BCMP2

■ 판별 결과 저장 대상

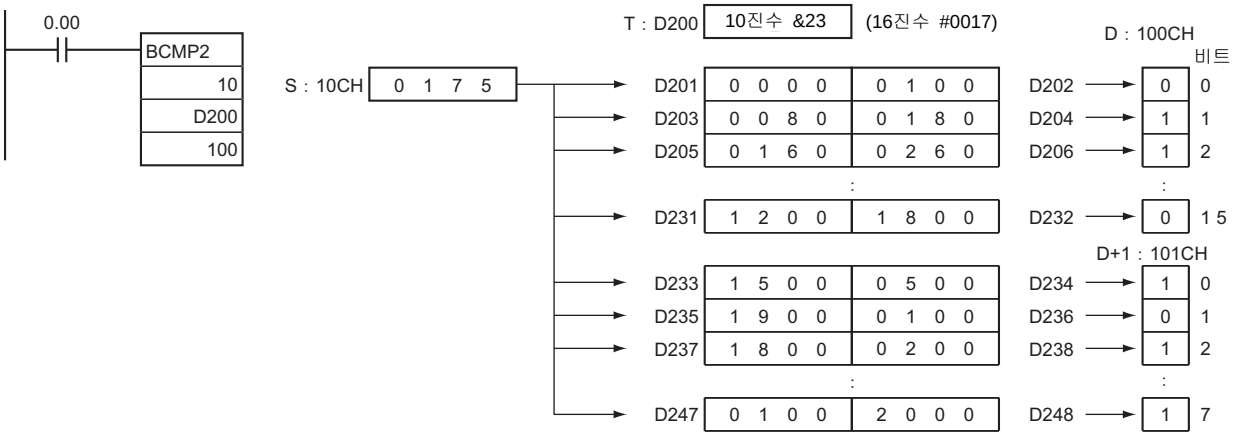
판별 결과는 D의 해당 비트에 출력됩니다.
설정 테이블의 구간 정의 수가 16을 초과하는 경우(최종 설정 구간 번호 N이 16 이상인 경우), 결과 출력 CH는 D로 지정한 채널 이후에 연속된 복수 CH에 걸쳐 출력합니다.



샘플 프로그램

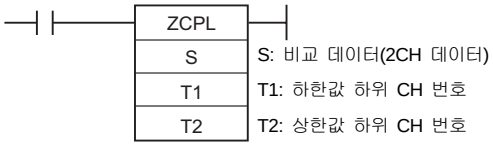
0.00이 ON일 때 10CH의 내용이 D200부터 구간 정의 데이터에 설정된 구간(최종 설정 구간N = 10진수 &23(16진수 #0017)의 24개)의 정의 구간 안에 있는지 여부를 판별하여 결과를 D 이후에 출력합니다.

[D201과 D202]의 정의 구간 안에 있는 경우 100CH의 비트 0에 1을 저장하고, 정의 구간 밖에 있는 경우에는 0을 저장합니다. 마찬가지로 10CH의 내용이 [D203과 D204], … [D247과 D248]의 정의 구간 안에 있으면 1을, 구간 밖에 있으면 0을 100CH의 비트 1…101CH의 비트 7에 저장합니다.



ZCP/ZCPL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
영역 비교	ZCP	–	088	지정된 하나의 CH 데이터 또는 상수가 지정된 상한값과 하한값 사이에 있는지 여부를 부호 없는 BIN 16비트(16진수 4자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.
2워드 영역 비교	ZCPL	–	116	지정된 하나의 2워드 CH 데이터, 또는 상수가 지정된 상한값과 하한값 사이에 있는지 여부를 부호 없는 BIN 32비트(16진수 8자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.

심볼	ZCP	ZCPL
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		ZCP	ZCPL	ZCP	ZCPL
S	ZCP: 비교 데이터(1CH 데이터) ZCPL: 비교 데이터(2CH 데이터)	UINT	UDINT	1	2
T1	ZCP: 하한값 ZCPL: 하한값 하위 CH 번호	UINT	UDINT	1	2
T2	ZCP: 상한값 ZCPL: 상한값 하위 CH 번호	UINT	UDINT	1	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
ZCP: S,T1,T2	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
ZCPL: S,T1,T2												–	○					

조건 플래그의 동작

이름	라벨	내용	
		ZCP	ZCPL
에러 플래그	P_ER	T1＞T2를 설정했을 때 ON	T1＋1, T1＞T2＋1, T2를 설정했을 때 ON
＞ 플래그	P_GT	· 비교 결과가 S＞T2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S+1, S＞T2+1, T2일 때 ON · 그 이외일 때 OFF

ZCP/ZCPL

이름	라벨	내용	
		ZCP	ZCPL
≧ 플래그	P_GE	· CJ2H CPU 유니트(유니트Ver.1.0~1.2), CS1-H/CJ1-H/CJ1M/CS1D CPU 유니트: 유지 · CJ2H CPU 유니트(유니트 Ver.1.3 이상), CJ2M CPU 유니트: · 비교 결과가 T1≦S일 때 ON · 그 이외일 때 OFF	· CJ2H CPU 유니트(유니트 Ver.1.0~1.2), CS1-H/CJ1-H/CJ1M/CS1D CPU 유니트: 유지 · CJ2H CPU 유니트(유니트 Ver.1.3 이상), CJ2M CPU 유니트: · 비교 결과가 T1+1, T1≦S+1, S일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 비교 결과가 T1≦S≦T2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 T1+1, T1≦S+1, S≦T2+1, T2일 때 ON · 그 이외일 때 OFF
≠ 플래그	P_NE	· CJ2H CPU 유니트(유니트 Ver.1.0~1.2), CS1-H/CJ1-H/CJ1M/CS1D CPU 유니트: 유지 · CJ2H CPU 유니트(유니트 Ver.1.3 이상), CJ2M CPU 유니트 · 비교 결과가 S＜T1일 때 ON · 비교 결과가 S＞T2일 때 ON · 그 이외일 때 OFF	· CJ2H CPU 유니트(유니트 Ver.1.0~1.2), CS1-H/CJ1-H/CJ1M/CS1D CPU 유니트: 유지 · CJ2H CPU 유니트(유니트 Ver.1.3 이상), CJ2M CPU 유니트 · 비교 결과가 S+1, S＜T1+1, T1일 때 ON · 그 이외일 때 OFF
< 플래그	P_LT	· 비교 결과가 S＜T1일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S+1, S＜T1+1, T1일 때 ON · 그 이외일 때 OFF
≦ 플래그	P_LE	· CJ2H CPU 유니트(유니트 Ver.1.0~1.2), CS1-H/CJ1-H/CJ1M/CS1D CPU 유니트: 유지 · CJ2H CPU 유니트(유니트 Ver.1.3 이상), CJ2M CPU 유니트 · 비교 결과가 S≦T2일 때 ON · 그 이외일 때 OFF	· CJ2H CPU 유니트(유니트 Ver.1.0~1.2), CS1-H/CJ1-H/CJ1M/CS1D CPU 유니트: 유지 · CJ2H CPU 유니트(유니트 Ver.1.3 이상), CJ2M CPU 유니트 · 비교 결과가 T1+1, T1≦S+1, S일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	유지	유지

기능

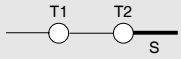
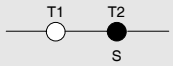
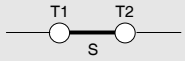
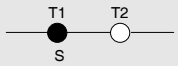
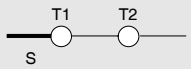
■ ZCP

비교 데이터 S가 하한값 T1~상한값 T2의 범위 안(T1≦S≦T2)에 들어가는지 여부를 부호 없는 BIN(16진수 4자리)으로 비교하여 결과를 조건 플래그에 반영합니다.

■ ZCPL

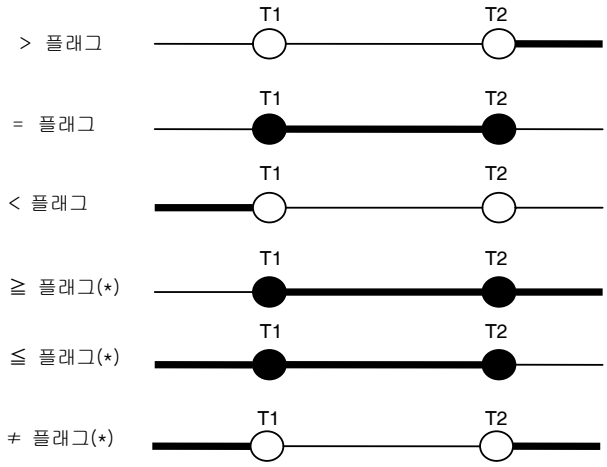
비교 데이터 S가 하한값 T1~상한값 T2의 범위 안(T1+1, T1≦S+1, S≦T2+1, T2)에 들어가는지 여부를 부호 없는 BIN(16진수 8자리)으로 비교하여 결과를 조건 플래그에 반영합니다.

비교 결과와 각 플래그의 동작

비교 결과	> 플래그	= 플래그	< 플래그	≧ 플래그(*) (> 플래그와 = 플래그의 논리합)	≦ 플래그(*) (< 플래그와 = 플래그의 논리합)	≠ 플래그(*) (> 플래그와 < 플래그의 논리합)
S > T2 	ON	OFF	OFF	ON	OFF	ON
S = T2 	OFF	ON			ON	OFF
T1 < S < T2 						
S = T1 						
S < T1 		OFF	ON	OFF		ON

* CJ2H CPU 유니트(유니트 Ver1.0~1.2), CS1-H/CJ1-H/CJ1M/CS1D CPU 유니트에서는 ON/OFF되지 않습니다. CJ2H CPU 유니트 (유니트 Ver.1.3 이상) CJ2M CPU 유니트에서만 ON/OFF됩니다.

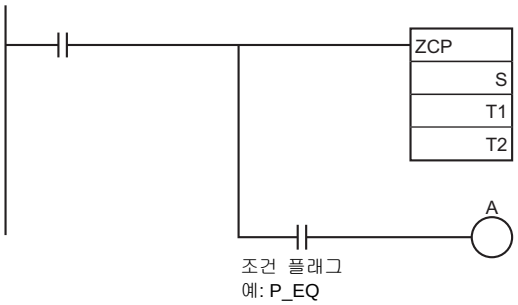
각 플래그가 ON되는 것은 다음과 같이 수직선에서 S의 값이 굵은 실선(—)의 범위 또는 검은 동그라미(●) 중 하나에 존재한다는 것을 의미합니다.



* CJ2H CPU 유니트(유니트 Ver1.0~1.2),
CS1-H/CJ1-H/CJ1M/CS1D CPU 유니트에서는 ON/OFF하지 않습니다

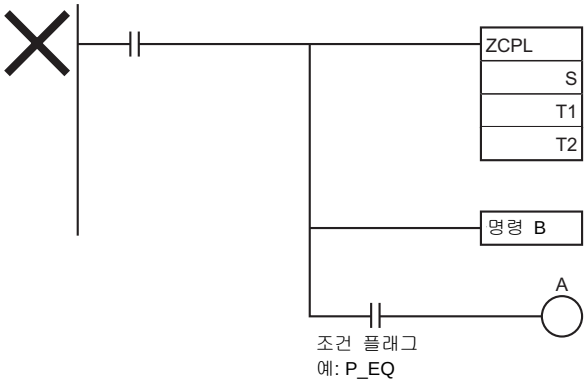
사용상의 주의 사항

· 이 명령을 실행하면 조건 플래그에 비교 결과가 반영되므로 예와 같이 ZCP(ZCPL) 명령과 동일한 입력 조건의 출력 분기에서 결과를 입력하십시오.



T1≦S≦T2일 때 P_EQ 플래그가 ON이 되고 A가 ON이 됩니다.

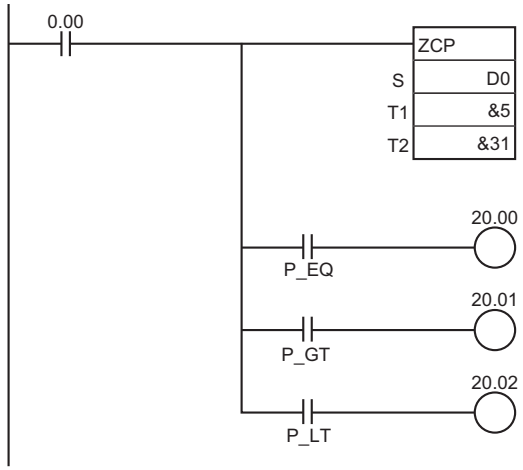
· 조건 플래그를 다른 명령 뒤에 배치하면 그 명령의 실행 결과에 의해 조건 플래그가 바뀌므로 조건 플래그는 반드시 ZCP(ZCPL) 명령 직후에 배치하십시오.

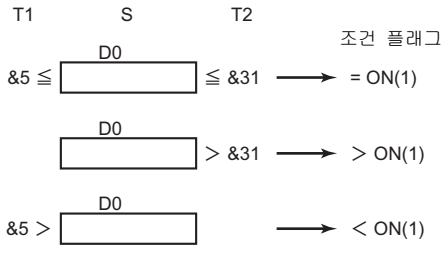


조건 플래그는 명령 B의 실행 결과를 수신합니다.

샘플 프로그램

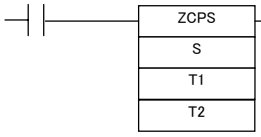
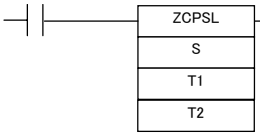
· 입력 릴레이 0.00이 ON일 때 D0의 내용이 10진수 &5~31 또는 16진수 #0005~001F 사이에 있는지 여부를 비교합니다.
· D0의 내용＞31일 때 20.01이 ON되고, 5≦D0의 내용≦31일 때 20.00이 ON되며, D0의 내용＜5일 때 20.02가 ON됩니다.





ZCPS/ZCPSL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부호 있는 영역 비교	ZCPS	–	117	지정된 하나의 CH 데이터 또는 상수가 지정된 상한값과 하한값 사이에 있는지 여부를 부호 있는 BIN 16비트(16진수 4자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.
부호 있는 2워드 영역 비교	ZCPSL	–	118	지정된 하나의 2워드 CH 데이터, 또는 상수가 지정된 상한값과 하한값 사이에 있는지 여부를 부호 있는 BIN 32비트(16진수 8자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.

심볼	ZCPS	ZCPSL
	 S: 비교 데이터(1CH 데이터) T1: 하한값 T2: 상한값	 S: 비교 데이터(2CH 데이터) T1: 하한값 하위 CH 번호 T2: 상한값 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		ZCPS	ZCPSL	ZCPS	ZCPSL
S	ZCPS: 비교 데이터(1CH 데이터) ZCPSL: 비교 데이터(2CH 데이터)	INT	DINT	1	2
T1	ZCPS: 하한값 ZCPSL: 하한값 하위 CH 번호	INT	DINT	1	2
T2	ZCPS: 상한값 ZCPSL: 상한값 하위 CH 번호	INT	DINT	1	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
ZCPS: S,T1,T2	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
ZCPSL: S,T1,T2												–	○					

조건 플래그의 동작

이름	라벨	내용	
		ZCPS	ZCPSL
에러 플래그	P_ER	T1＞T2를 설정했을 때 ON	T1＋1, T1＞T2＋1, T2를 설정했을 때 ON
＞ 플래그	P_GT	· 비교 결과가 S＞T2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S+1, S＞T2+1, T2일 때 ON · 그 이외일 때 OFF
≧ 플래그	P_GE	· 비교 결과가 T1≦S일 때 ON · 그 이외일 때 OFF	· 비교 결과가 T1+1, T1≦S+1, S일 때 ON · 그 이외일 때 OFF

ZCPS/ZCPSL

이름	라벨	내용	
		ZCPS	ZCPSL
= 플래그	P_EQ	· 비교 결과가 T1≦S≦T2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 T1+1, T1≦S+1, S≦T2+1, T2일 때 ON · 그 이외일 때 OFF
≠ 플래그	P_NE	· 비교 결과가 S＜T1일 때 ON · 비교 결과가 S＞T2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S+1, S＜T1+1, T1일 때 ON · 비교 결과가 S+1, S＞T2+1, T2일 때 ON · 그 이외일 때 OFF
< 플래그	P_LT	· 비교 결과가 S＜T1일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S+1, S＜T1+1, T1일 때 ON · 그 이외일 때 OFF
≦ 플래그	P_LE	· 비교 결과가 S≦T2일 때 ON · 그 이외일 때 OFF	· 비교 결과가 S+1, S≦T2+1, T2일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	유지	유지

기능

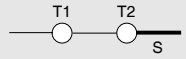
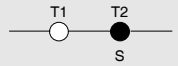
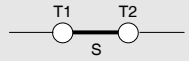
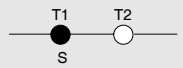
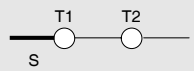
■ ZCPS

비교 데이터 S가 하한값 T1~상한값 T2의 범위 안(T1≦S≦T2)에 들어가는지 여부를 부호 있는 BIN(16진수 4자리)으로 비교하여 결과를 조건 플래그에 반영합니다.

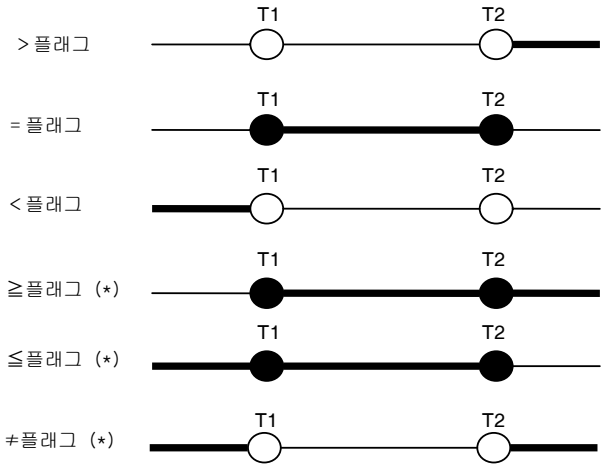
■ ZCPSL

비교 데이터 S가 하한값 T1~상한값 T2의 범위 안(T1+1, T1≦S+1, S≦T2+1, T2)에 들어가는지 여부를 부호 있는 BIN(16진수 8자리)으로 비교하여 결과를 조건 플래그에 반영합니다.

비교 결과와 각 플래그의 동작

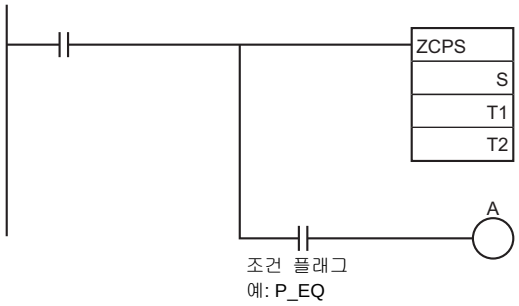
비교 결과	＞ 플래그	＝ 플래그	＜ 플래그	≧ 플래그(*) (＞ 플래그와 ＝ 플래그의 논리합)	≦ 플래그(*) (＜ 플래그와 ＝ 플래그의 논리합)	≠ 플래그(*) (＞ 플래그와 ＜ 플래그의 논리합)
S＞T2 	ON	OFF	OFF	ON	OFF	ON
S＝T2 	OFF	ON			ON	OFF
T1＜S＜T2 						
S＝T1 						
S＜T1 		OFF	ON	OFF		ON

각 플래그가 ON되는 것은 다음과 같이 수직선에서 S의 값이 굵은 실선(—)의 범위 또는 검은 동그라미(●) 중 하나에 존재한다는 것을 의미합니다.



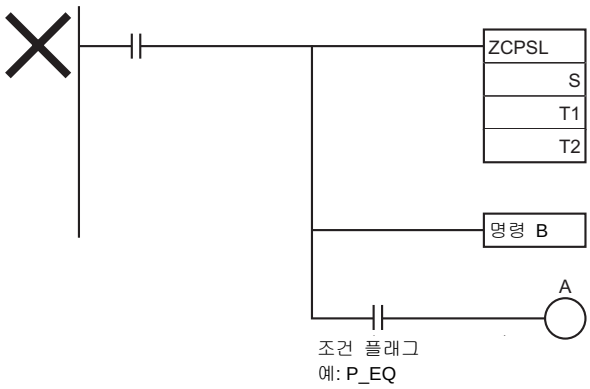
사용상의 주의 사항

- 이 명령을 실행하면 조건 플래그에 비교 결과가 반영되므로 예와 같이 ZCPS(ZCPSL) 명령과 동일한 입력 조건의 출력 분기에서 결과를 입력하십시오.



T1≦S≦T2일 때 P_EQ 플래그가 ON이 되고 A가 ON이 됩니다.

- 조건 플래그를 다른 명령 뒤에 배치하면 그 명령의 실행 결과에 의해 조건 플래그가 바뀌므로 조건 플래그는 반드시 ZCPS(ZCPSL) 명령 직후에 배치하십시오.

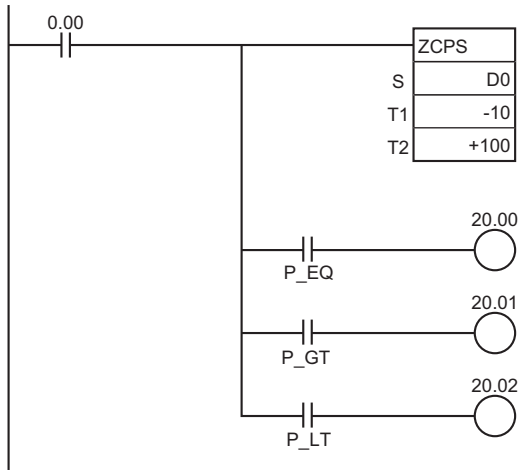


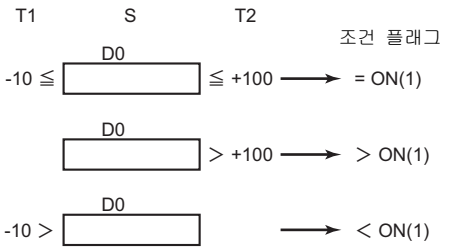
조건 플래그는 명령 B의 실행 결과를 수신합니다.

ZCPS/ZCPSL

샘플 프로그램

- 입력 릴레이 0.00이 ON일 때 D0의 내용이 부호 있는 10진수 －10~＋100(부호 있는 16진수 #FFF6~0064) 사이에 있는지 여부를 비교합니다.
- D0의 내용＞+100일 때 20.01이 ON되고, -10≦D0의 내용≦100일 때 20.00이 ON되며, D0의 내용＜-10일 때 20.02가 ON됩니다.



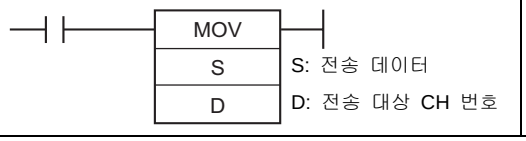
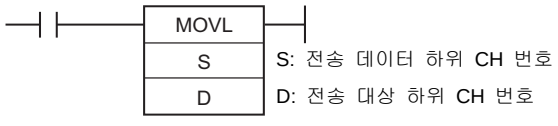


데이터 전송 명령

니모닉	명령 이름	펑션 번호	페이지
MOV	전송	021	3-152
MOVL	2워드 전송	498	
MVN	부정 전송	022	3-154
MVNL	부정 2워드 전송	499	
MOVB	비트 전송	082	3-156
MOVD	자릿수 전송	083	3-158
XFRB	다중 비트 전송	062	3-160
XFER	블록 전송	070	3-162
BSET	블록 설정	071	3-164
XCHG	데이터 교환	073	3-166
XCGL	데이터 2워드 교환	562	
DIST	데이터 분배	080	3-168
COLL	데이터 추출	081	3-170
MOVR	인덱스 레지스터 설정	560	3-172
MOVRW	인덱스 레지스터 설정	561	

MOV/MOVL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
전송	MOV	@MOV, !MOV, !@MOV	021	CH 데이터 또는 상수를 지정된 CH로 전송합니다.
2워드 전송	MOVL	@MOVL	498	2CH분의 CH 데이터 또는 상수를 32비트 단위로 전송 대상 CH에 출력합니다.

심볼	MOV	MOVL
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		MOV	MOVL	MOV	MOVL
S	MOV: 전송 데이터 MOVL: 전송 데이터 하위 CH 번호	WORD	DWORD	1	2
D	MOV: 전송 대상 CH 번호 MOVL: 전송 대상 하위 CH 번호	WORD	DWORD	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
MOV	S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
	D											–							
MOVL	S	○	○	○	○	○	○	○	○	○	○	○	–	○	○	–	–	–	–
	D											–							

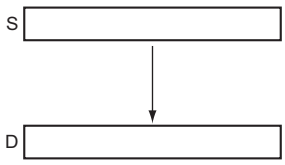
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 전송된 데이터(D)가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 전송된 데이터(D)의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

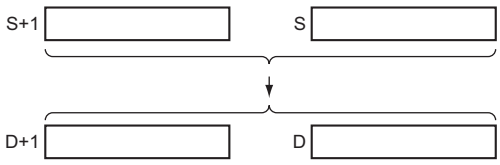
■ MOV

S를 D에 전송합니다. S가 상수인 경우, 데이터 설정에 사용할 수 있습니다.



■ MOVL

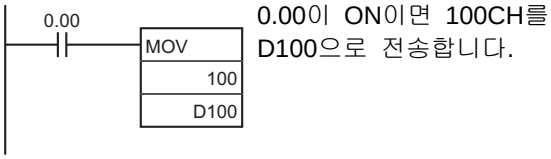
S를 2워드 데이터로 D+1, D에 전송합니다. S, S+1이 상수인 경우, 데이터 설정에 사용할 수 있습니다.



사용상의 주의 사항

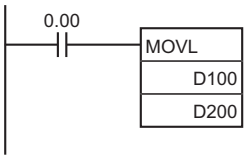
- MOV 명령은 매회 리프레시형 명령(!MOV)으로 사용할 수 있습니다.
- 매회 리프레시형 명령(!MOV)의 경우, S에 외부 I/O 할당된 입력 릴레이 영역을 지정할 수 있습니다. 또한 D에 외부 I/O 할당된 출력 릴레이 영역을 지정할 수 있습니다(단, 그룹 2 다점 입출력 유니트, 고기능 I/O 유니트의 다점 입출력 유니트, SYSBUS 원격 I/O 자국 장치의 유니트는 제외).
  S에 외부 입력을 지정한 경우, 명령 실행 시 S의 값을 IN 리프레시하고 그 값을 D에 전송합니다. D에 외부 출력을 지정한 경우에는 명령 실행 시 S의 값을 D에 전송하고 즉시 OUT 리프레시합니다. S를 IN 리프레시하고 동시에 D를 OUT 리프레시하는 것도 가능합니다.

샘플 프로그램

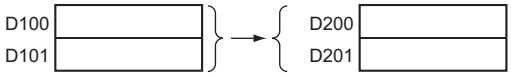


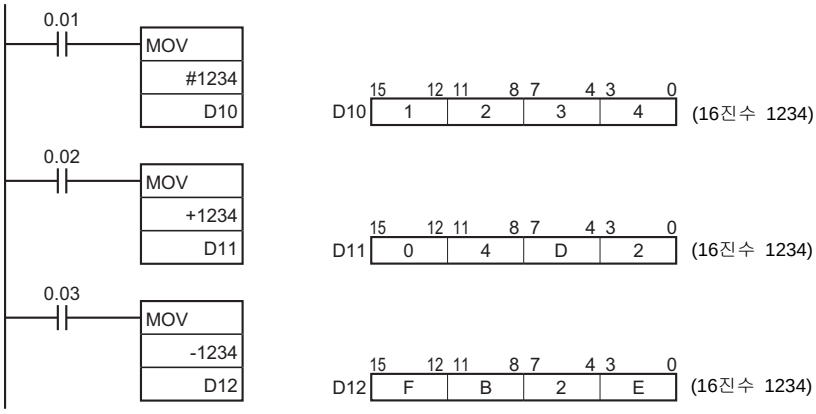
0.00이 ON이면 100CH를 D100으로 전송합니다.





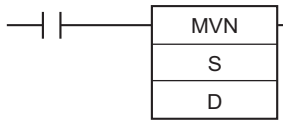
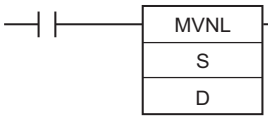
0.00이 ON이면 D100~D101을 D200~D201로 전송합니다.





MVN/MVNL

명령 이름	니모닉	동작 옵션	평션 번호	기능
부정 전송	MVN	@MVN	022	CH 데이터 또는 상수의 비트 반전 데이터를 지정된 CH로 전송합니다.
부정 2워드 전송	MVNL	@MVNL	499	2CH분의 CH 데이터 또는 상수의 비트 반전 데이터를 32비트 단위로 지정된 CH에 출력합니다.

심볼	MVN	MVNL
	 S: 전송 데이터 D: 전송 대상 CH 번호	 S: 전송 데이터 하위 CH 번호 D: 전송 대상 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		MVN	MVNL	MVN	MVNL
S	MVN: 전송 데이터 MVNL: 전송 데이터 하위 CH 번호	WORD	DWORD	1	2
D	MVN: 전송 대상 CH 번호 MVNL: 전송 대상 하위 CH 번호	WORD	DWORD	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
MVN	S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
	D											–							
MVNL	S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
	D											–							

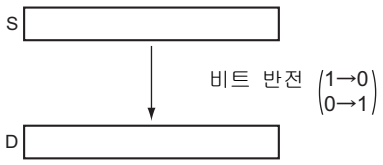
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 전송된 데이터(D)가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 전송된 데이터(D)의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

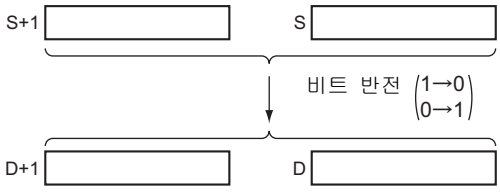
■ MVN

S의 16비트를 비트 반전하여 D에 전송합니다.

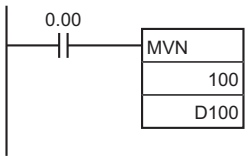


■ MVNL

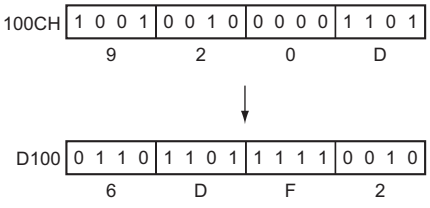
S를 2워드 데이터로 비트 반전하여 D+1, D에 전송합니다.

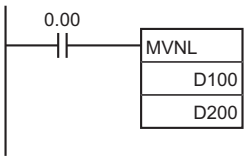


샘플 프로그램

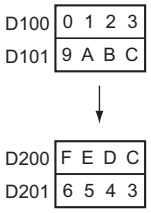


0.00이 ON이면 100CH의 각 비트를 반전하여 D100으로 전송합니다.





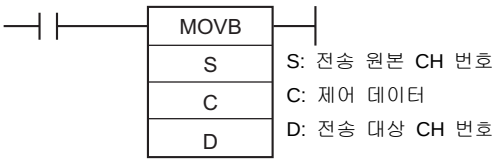
0.00이 ON이면 D100~101의 각 비트를 반전하여 D200~D201로 전송합니다.



MOVB

명령 이름	니모닉	동작 옵션	평션 번호	기능
비트 전송	MOVB	@MOVB	082	지정한 비트를 전송합니다.

심볼	MOVB



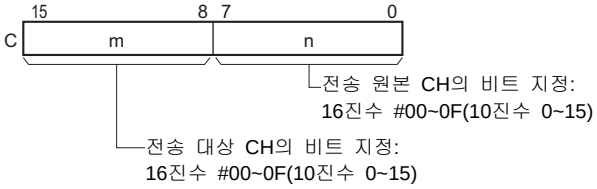
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	전송 원본 CH 번호	WORD	1
C	제어 데이터	UINT	1
D	전송 대상 CH 번호	WORD	1

C: 제어 데이터



■ 영역 종류

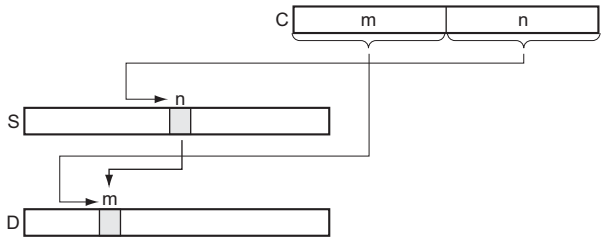
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
C																		
D											–							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C의 데이터가 범위 밖일 때 ON · 그 이외일 때 OFF

기능

S의 지정된 비트 위치(C의 n)의 내용(0/1)을 D의 지정된 비트 위치(C의 m)에 전송합니다.



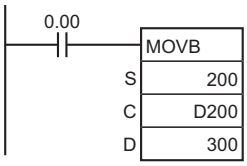
힌트

· S와 D에 동일 채널을 지정하면 비트 위치 변경 등에 사용할 수 있습니다.

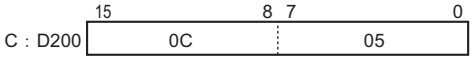
사용상의 주의 사항

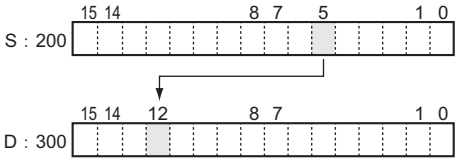
· 전송 대상 CH의 데이터는 전송된 비트 이외에는 변경되지 않습니다.

샘플 프로그램



0.00이 ON일 때 제어 데이터(C)의 내용이 다음과 같은 경우 200CH의 비트 5를 300CH의 비트 12로 전송합니다.

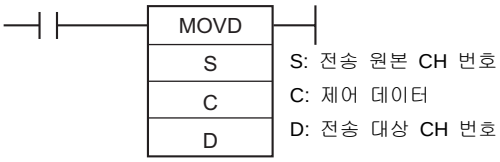




MOVD

명령 이름	니모닉	동작 옵션	평션 번호	기능
자릿수 전송	MOVD	@MOVD	083	자리(4비트) 단위로 전송하며 여러 자리의 전송도 가능합니다.

심볼	MOVD



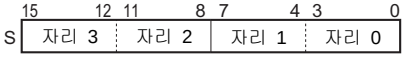
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

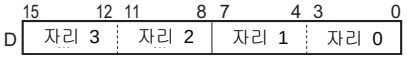
피연산자	내용	데이터 형식	크기
S	전송 원본 CH 번호	WORD	1
C	제어 데이터	UINT	1
D	전송 대상 CH 번호	UINT	1

S: 전송 원본 CH 번호



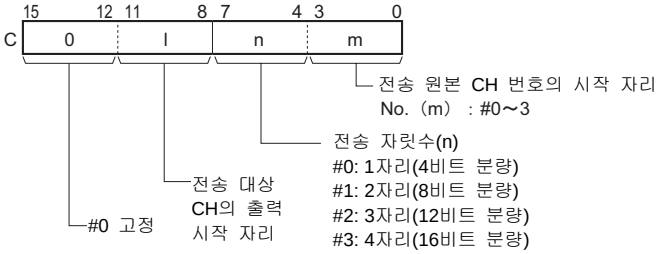
시작 자리에서 상위 측 자리가 전송 원본이 됨
(자리 3의 다음은 자리 0으로 돌아감)

D: 전송 대상 CH 번호



시작 자리에서 상위 측 자리가 전송 대상이 됨
(자리 3의 다음은 자리 0으로 돌아감)

C: 제어 데이터



■ 영역 종류

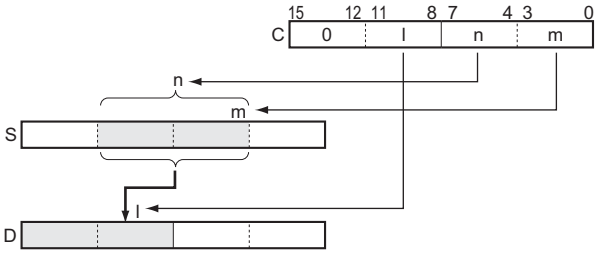
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
C																		
D											–							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C의 데이터가 범위 밖일 때 ON · 그 이외일 때 OFF

기능

1자리를 4비트로 하여 S의 지정된 전송 시작 자리(C의 m)부터 지정된 전송 자릿수(C의 n)만큼의 내용을 D의 지정된 출력 시작 자리(C의 I) 이후에 전송합니다.



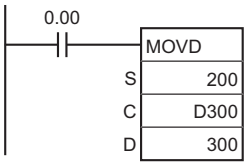
힌트

· S와 D에 동일 채널을 지정하면 채널 내의 비트 위치 변환 등에 사용할 수 있습니다.

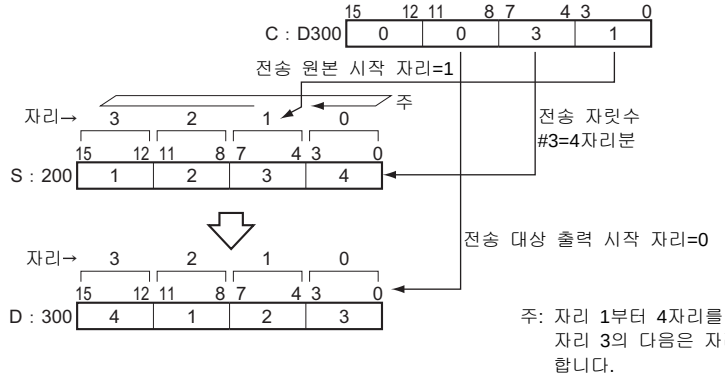
사용상의 주의 사항

· 전송 대상 CH의 데이터는 전송된 자리 이외에는 변경되지 않습니다.
· 여러 자리를 전송하는 경우, 전송 대상 CH 내의 최상위 자리에서 밀려나는 자리는 동일 CH의 최하위 자리 쪽으로 우회하여 전송됩니다.

샘플 프로그램



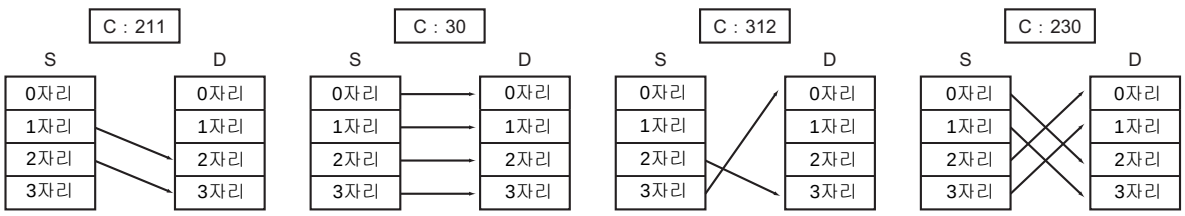
0.00이 ON일 때 제어 데이터(C)의 내용이 다음과 같은 경우 200CH의 자리 1에서 상위 쪽 4자리의 데이터를 300CH의 자리 0에서 상위 쪽 4자리로 전송합니다.



주: 자리 1부터 4자리를 전송할 때는 자리 3의 다음은 자리 0을 전송합니다.

■ 여러 자리(자릿수) 전송 예

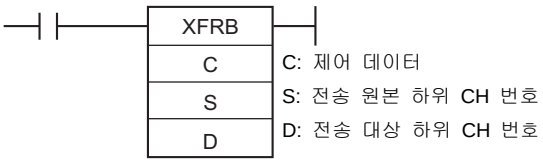
여러 자리를 전송하는 경우, 전송 원본의 시작 자리 번호와 전송 대상의 출력 시작 자리 번호는 하위 자리 쪽을 지정합니다.



XFRB

명령 이름	니모닉	동작 옵션	평션 번호	기능
다중 비트 전송	XFRB	@XFRB	062	지정된 CH의 지정된 복수의 비트를 전송합니다.

심볼	XFRB



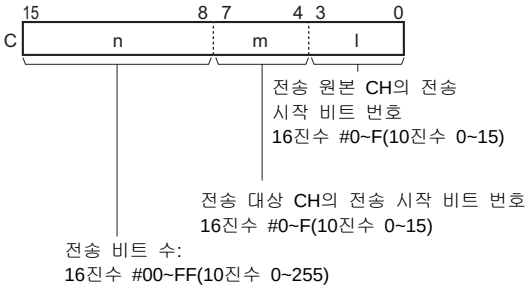
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

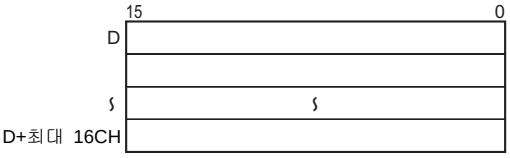
피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	UINT	1
S	전송 원본 하위 CH 번호	WORD	가변
D	전송 대상 하위 CH 번호	WORD	가변

C: 제어 데이터

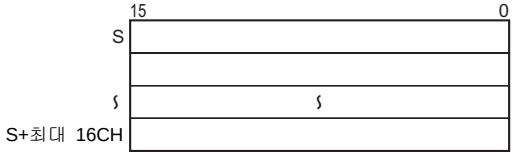


D: 전송 대상 하위 CH 번호



주: S~S+최대 16CH 및 D~D+최대 16CH는 각각 동일 영역 종류이어야 합니다.

S: 전송 원본 하위 CH 번호



■ 영역 종류

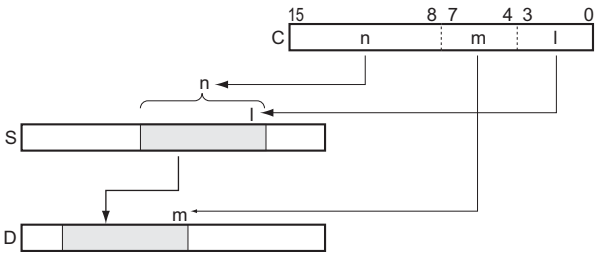
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
S											–	–						
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF

기능

S로 지정한 전송 원본 하위 CH 번호의 지정된 시작 비트 위치(C의 l)부터 지정된 비트 수(C의 n)만큼의 데이터를 D로 지정된 전송 대상 하위 CH 번호의 지정된 시작 비트 위치(C의 m) 이후에 전송합니다.



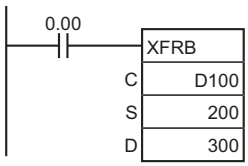
힌트

- 여러 채널에 걸쳐 있는 데이터를 1개 명령으로 최대 255비트분까지 전송할 수 있습니다.
- 여러 채널에 걸쳐 데이터를 전송하면 데이터를 앞부터 채울 수 있으므로 데이터 영역을 효율적으로 사용할 수 있습니다. (특히, 위치 결정 등의 작업 시 위치 데이터의 사용이 쉬워집니다.)
- 전송 원본과 전송 대상이 중복되어도 정상적으로 전송할 수 있으므로 ANDW 명령과 조합하면 m비트분을 n비트 이동하는 것과 같이 이동 명령을 대행할 수도 있습니다.
- 전송 원본과 전송 대상의 데이터 영역을 중복하여 지정할 수도 있습니다.

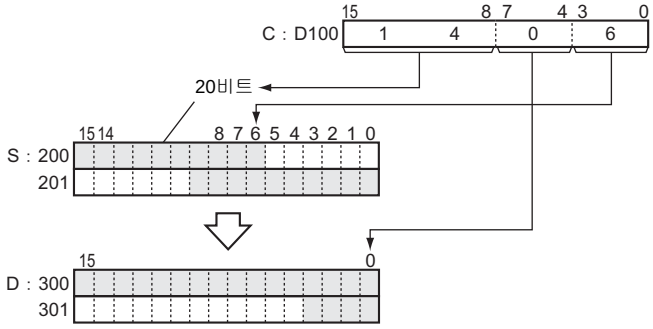
사용상의 주의 사항

- 전송 원본, 전송 대상의 데이터 범위가 영역의 최대 범위를 초과하지 않도록 하십시오.
- 전송 비트 수(C의 n)가 0인 경우 전송하지 않습니다.
- 전송 대상 CH의 내용은 전송된 비트 이외에는 변경되지 않습니다.

샘플 프로그램

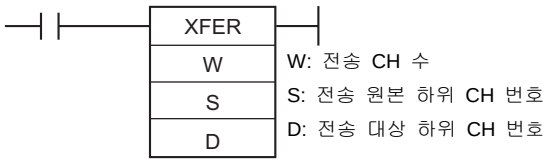


0.00이 ON일 때 제어 데이터(C)의 내용이 다음과 같은 경우 200CH의 비트 6에서 상위 쪽 20비트를 300CH의 비트 0에서 상위 쪽 20비트로 전송합니다.



XFER

명령 이름	니모닉	동작 옵션	평션 번호	기능
블록 전송	XFER	@XFER	070	연속된 여러 CH 데이터를 일괄 전송합니다.

	XFER
심볼	

특정 영역 내의 사용 여부

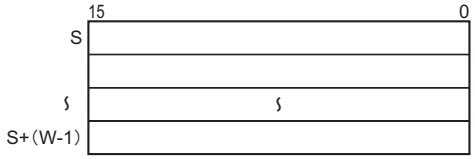
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

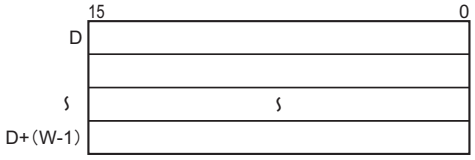
피연산자 설명

피연산자	내용	데이터 형식	크기
W	전송 CH 수	UINT	1
S	전송 원본 하위 CH 번호	WORD	가변
D	전송 대상 하위 CH 번호	WORD	가변

W: 전송 CH 수

10진수 &0~65535 또는 16진수 #0000~FFFF





■ 영역 종류

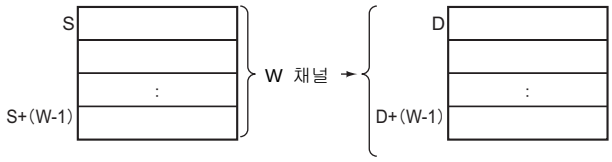
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
W	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
S											–	–						
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF

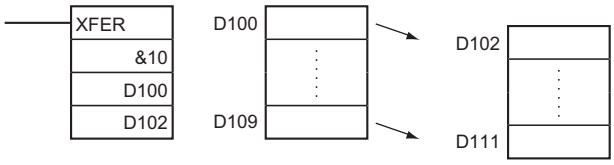
기능

S로 지정한 전송 원본 하위 CH 번호부터 W로 지정한 데이터 수(10진수 또는 16진수)만큼 D로 지정한 전송 대상 하위 CH 번호 이후에 전송합니다.



힌트

· S와 D에 같은 영역 종류를 지정하고 XFER 명령을 사용하여 데이터를 이동할 수도 있습니다.

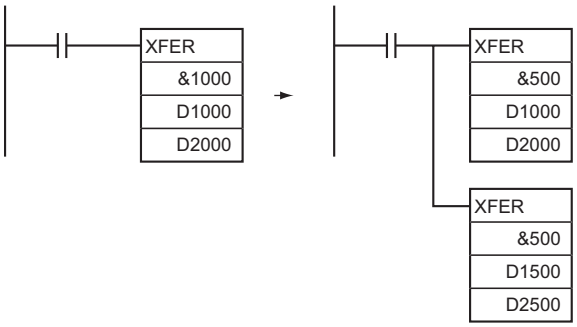


· 전송 원본과 전송 대상의 데이터 영역을 중복하여 지정할 수도 있습니다(워드 이동 동작).

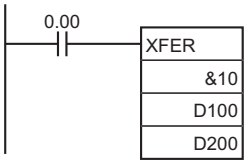
사용상의 주의 사항

· 전송 원본, 전송 대상 CH가 데이터 영역을 초과하지 않도록 하십시오.

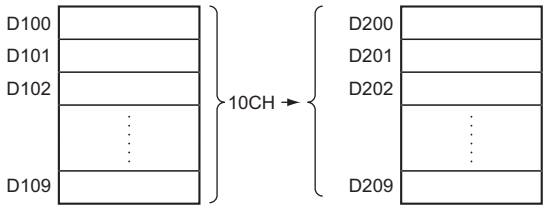
· 대량의 채널을 블록 전송하는 경우 명령 실행 시간이 걸립니다. 이 명령 실행 중에 인터럽트가 발생해도 명령 실행이 끝날 때까지 인터럽트 태스크는 기동되지 않습니다.
또한 이 명령 실행 중에 전원 OFF가 발생하는 경우 블록 전송 도중에 실행이 종료되는 경우가 있습니다.
명령을 분할하면 이와 같은 문제를 피할 수 있습니다.



샘플 프로그램

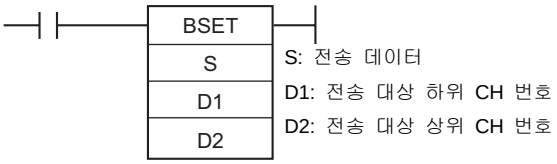


0.00이 ON이면 D100~D109의 10CH를 D200~D209로 전송합니다.



BSET

명령 이름	니모닉	동작 옵션	평션 번호	기능
블록 설정	BSET	@BSET	071	연속된 복수의 CH 전체에 동일한 데이터를 설정합니다.

심볼	BSET
	

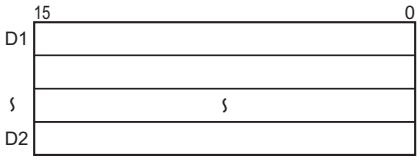
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	전송 데이터	WORD	1
D1	전송 대상 하위 CH 번호	WORD	가변
D2	전송 대상 상위 CH 번호	WORD	가변

D1: 전송 대상 하위 CH 번호
D2: 전송 대상 상위 CH 번호



주: D1~D2는 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
D1											–	–						
D2																		

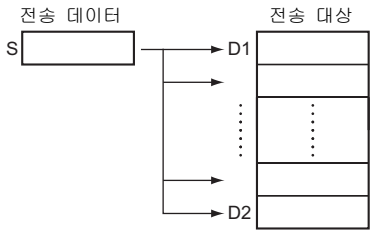
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· D1 > D2일 때 ON · 그 이외일 때 OFF

주: D1≦D2이어야 합니다. D1 > D2인 경우 에러가 되고 에러 플래그가 ON이 됩니다.

기능

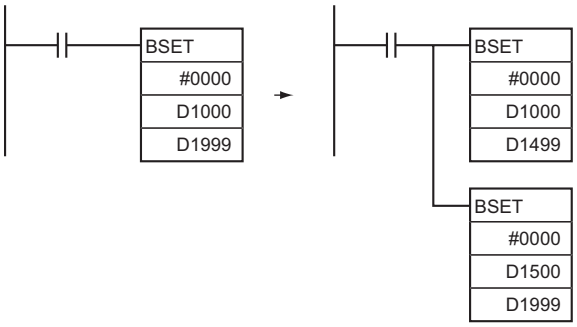
D1로 지정한 전송 대상 하위 CH 번호부터 D2로 지정한 전송 대상 상위 CH 번호까지 S를 출력합니다.



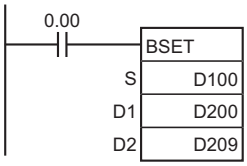
사용상의 주의 사항

대량의 채널을 블록 설정하는 경우 명령 실행 시간이 걸립니다.
이 명령 실행 중에 인터럽트가 발생해도 명령 실행이 끝날 때까지 인터럽트 태스크는 기동되지 않습니다.
또한 이 명령을 실행할 때 전원 OFF가 발생하는 경우 블록 설정 도중에 실행이 종료되는 경우가 있습니다.
명령을 분할하면 이와 같은 문제를 피할 수 있습니다.

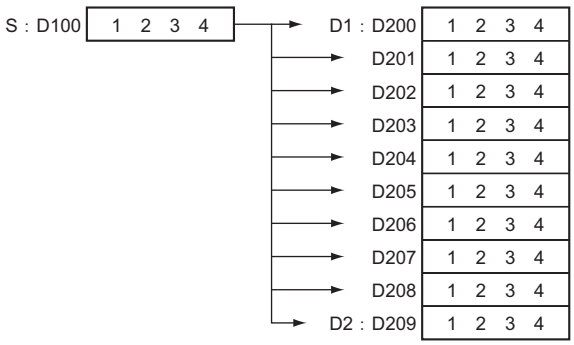
예)



샘플 프로그램



0.00이 ON이면 D100의 내용을 D200~D209의 10CH로 전송합니다.



XCHG/XCGL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
데이터 교환	XCHG	@XCHG	073	CH 간에 데이터를 교환합니다.
데이터 2워드 교환	XCGL	@XCGL	562	2CH의 CH 데이터끼리 32비트 단위로 교환합니다.

심볼	XCHG	XCGL
	XCHG D1 D2 D1: 교환 CH 번호 1 D2: 교환 CH 번호 2	XCGL D1 D2 D1: 교환 하위 CH 번호 1 D2: 교환 하위 CH 번호 2

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		XCHG	XCGL	XCHG	XCGL
D1	XCHG: 교환 CH 번호 1 XCGL: 교환 하위 CH 번호 1	WORD	DWORD	1	2
D2	XCHG: 교환 CH 번호 2 XCGL: 교환 하위 CH 번호 2	WORD	DWORD	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
XCHG	D1,D2	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
XCGL	D1,D2												–	○					

조건 플래그의 동작

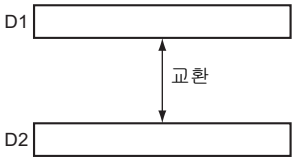
이름	라벨	내용
에러 플래그	P_ER	변하지 않음*1
= 플래그	P_EQ	변하지 않음*1
부정 플래그	P_N	변하지 않음*1

*1: CS1/CJ1/CS1D(이중화 시스템용) CPU 유니트는 OFF

기능

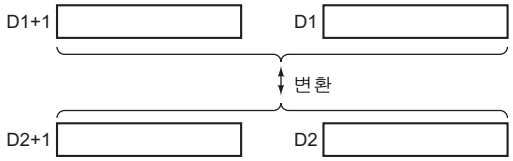
■ **XCHG**

D1과 D2를 교환합니다.



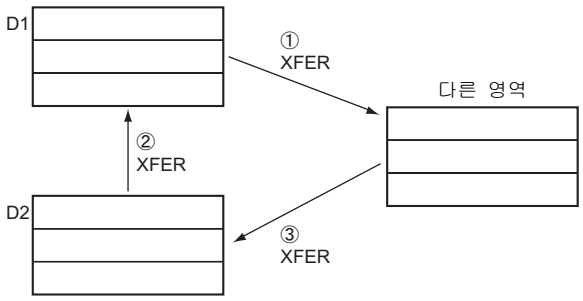
■ **XCGL**

D1과 D2를 2워드 데이터로 교환합니다.

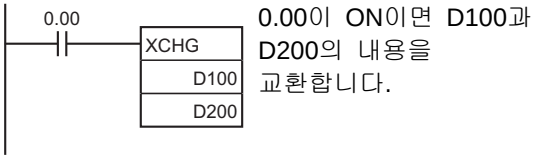


힌트

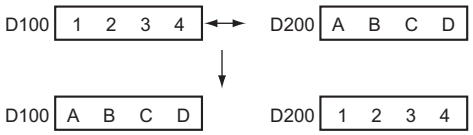
3CH 이상의 채널 데이터를 일괄 교환하려는 경우는 다음과 같이 일단 XFER(블록 전송) 명령을 사용하여 다른 영역을 통해 데이터를 교환하십시오.

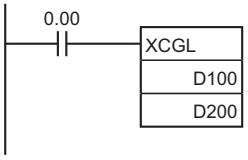


샘플 프로그램

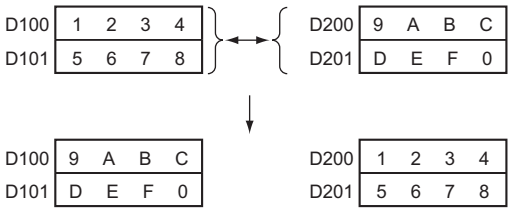


0.00이 ON이면 D100과 D200의 내용을 교환합니다.





0.00이 ON이면 D100~D101과 D200~D201의 내용을 교환합니다.



DIST

명령 이름	니모닉	동작 옵션	평션 번호	기능
데이터 분배	DIST	@DIST	080	전송 대상을 기준으로 오프셋 지정된 채널로 전송 데이터를 전송합니다.

심볼	DIST



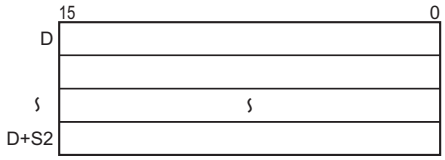
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S1	전송 데이터	WORD	1
D	전송 대상 기준 CH 번호	WORD	1
S2	오프셋 데이터	UINT	1

D: 전송 대상 기준 CH 번호



S2: 오프셋 데이터

10진수 &0~65535 또는 16진수 #0000~FFFF

주: D~D+S2는 동일 영역 종류이어야 합니다.

■ 영역 종류

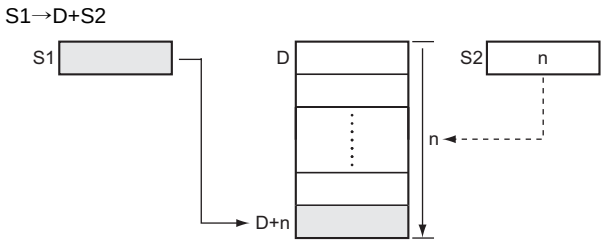
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S1	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
D											–	–						
S2											○	○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 전송 데이터가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 전송 데이터의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

D로 지정한 전송 대상 기준 CH 번호부터 S2로 지정한 오프셋 데이터만큼 이동한 주소에 S1을 전송합니다.



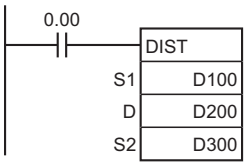
힌트

· 오프셋 데이터(S2)의 내용을 변경하면 DIST 명령 하나로 임의의 위치에 데이터를 전송(분배)할 수 있습니다.

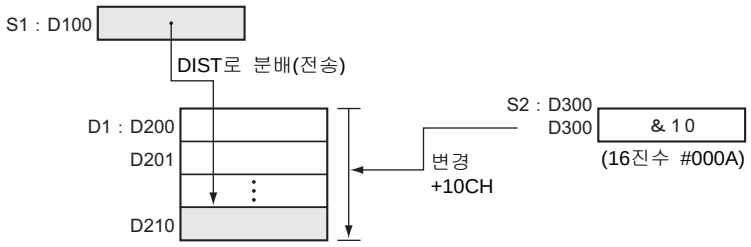
사용상의 주의 사항

· 오프셋 데이터(S2)의 내용은 전송 대상의 영역 범위를 벗어나지 않도록 하십시오.

샘플 프로그램



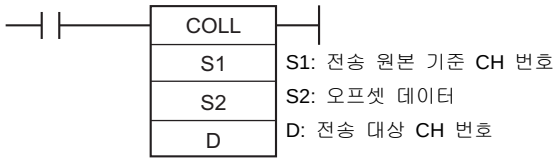
0.00이 ON이면 D200에 D300의 내용(예: &10(16진수 #000A))을 가산한 주소(D210)로 D100의 내용을 전송합니다. (D300의 내용을 변경하면 D100의 내용을 임의의 주소로 분배할 수 있습니다.)



COLL

명령 이름	니모닉	동작 옵션	평션 번호	기능
데이터 추출	COLL	@COLL	081	전송 원본을 기준으로 오프셋 지정된 채널의 내용을 지정 채널로 전송합니다.

심볼	COLL



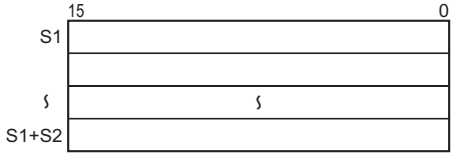
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S1	전송 원본 기준 CH 번호	WORD	1
S2	오프셋 데이터	WORD	1
D	전송 대상 CH 번호	WORD	1

S1: 전송 원본 기준 CH 번호



S2: 오프셋 데이터

10진수 &0~65535 또는 16진수 #0000~FFFF

주: D~D+S는 동일 영역 종류이어야 합니다.

■ 영역 종류

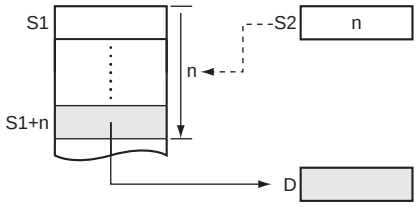
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S1	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S2											○	○						
D											–							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 전송 데이터가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 전송 데이터의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

S1로 지정한 전송 원본 기준 CH에서 S2로 지정한 오프셋 데이터만큼 이동한 주소의 데이터를 D에 전송합니다.



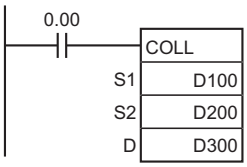
힌트

· 오프셋 데이터(S2)의 내용을 변경하면 COLL 명령 하나로 임의의 위치에서 데이터를 가져올(추출) 수 있습니다.

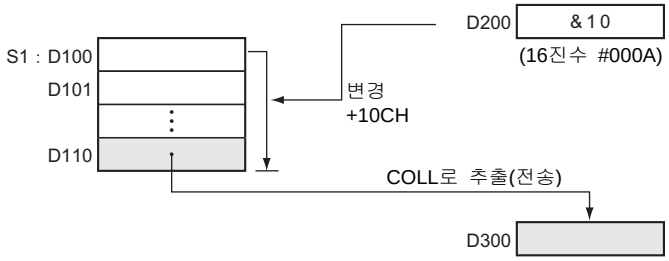
사용상의 주의 사항

· 오프셋 데이터(S2)의 내용은 전송 원본의 영역 범위를 벗어나지 않도록 하십시오.

샘플 프로그램

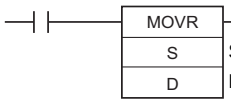
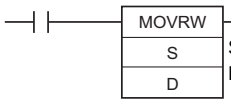


0.00이 ON이면 D100에 D200의 내용(예: &10(16진수 #000A))을 가산한 주소(D110)의 내용을 D300으로 전송합니다. (D200의 내용을 변경하면 임의의 주소에서 추출할 수 있습니다.)



MOVR/MOVRW

명령 이름	니모닉	동작 옵션	평션 번호	기능
인덱스 레지스터 설정	MOVR	@MOVR	560	지정된 CH 번호, 접점 번호의 I/O 메모리 실제 주소를 지정된 인덱스 레지스터에 설정합니다.
인덱스 레지스터 설정	MOVRW	@MOVRW	561	지정된 타이머/카운터 현재값의 I/O 메모리 실제 주소를 지정된 인덱스 레지스터에 설정합니다.

심볼	MOVR	MOVRW
	 S: 지정 CH 번호/접점 번호 D: 전송 대상 인덱스 레지스터 번호	 S: 지정 타이머/카운터 번호 D: 전송 대상 인덱스 레지스터 번호

특정 영역 내의 사용 여부

■ MOVR

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

■ MOVRW

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	가능	가능	가능	가능	불가능

피연산자 설명

피연산자	내용	데이터 형식		크기
		MOVR	MOVRW	
S	MOVR: 지정 CH 번호/접점 번호 MOVRW: 지정 타이머/카운터 번호	BOOL	UINT	1
D	전송 대상 인덱스 레지스터 번호	WORD	WORD	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
MOVR	S	○	○	○	○	○	○	○	○	–	–	–	–	–	–	○	–	–	–
	D	–	–	–	–	–	–	–	–					○		–			
MOVRW	S	–	–	–	–	○	○	–	–	–	–	–	–	–	–	–	–	–	–
	D					–	–							○					

조건 플래그의 동작

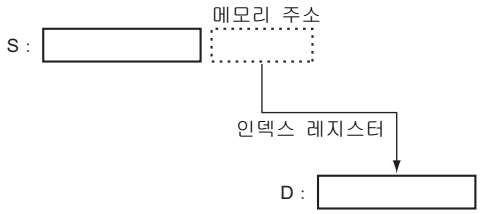
이름	라벨	내용
에러 플래그	P_ER	변하지 않음*1
= 플래그	P_EQ	변하지 않음*1
부정 플래그	P_N	변하지 않음*1

*1: CS1/CJ1/CS1D(이중화 시스템용) CPU 유니트는 OFF

기능

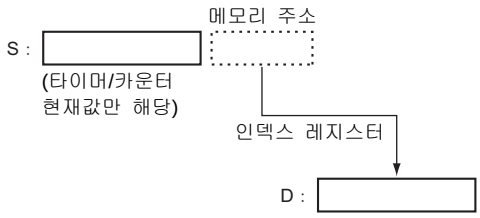
■ MOVR

S의 I/O 메모리 실제 주소를 D로 지정된 인덱스 레지스터에 전송합니다.



■ MOVRW

S(타이머 또는 카운터)의 I/O 메모리 실제 주소(타이머 현재값 영역 또는 카운터 현재값 영역)를 D로 지정된 인덱스 레지스터에 전송합니다.



사용상의 주의 사항

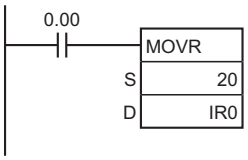
■ MOVR

- 인덱스 레지스터(IR0~15)에 대한 I/O 메모리 실제 주소 설정(타이머/카운터의 현재값은 제외)은 이 명령으로 수행합니다.
- 일반적인 I/O 메모리의 주소(영역 종류별 주소)에서 S를 지정하면 이 값을 자동으로 I/O메모리 실제 주소로 변환하여 D에 저장합니다.
- 이 명령에서 S에 타이머/카운터를 지정하면 타이머/카운터의 업 플래그의 I/O메모리 실제 주소를 D에 저장합니다.
- 타이머/카운터의 현재값의 I/O 메모리 실제 주소는 이 명령이 아니라 MOVRW 명령으로 설정합니다.
- 인터럽트 태스크에서의 IR은 일정치 않습니다. 따라서 인터럽트 태스크 중에는 반드시 이 명령을 사용하여 설정하십시오. 또한 인터럽트 태스크에서 실행된 IR/DR에는 사이클릭 태스크가 영향을 주지 않습니다.

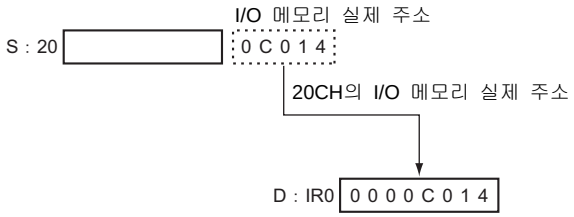
■ MOVRW

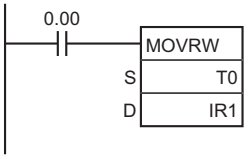
- 인덱스 레지스터(IR0~15)에 대한 타이머/카운터 현재값의 I/O 메모리 실제 주소 설정은 이 명령으로 수행합니다.
- 일반적인 타이머/카운터 주소로 S를 지정하면 이 값을 자동으로 타이머/카운터 현재값의 I/O 메모리 실제 주소로 변환하여 D에 저장합니다.
- 타이머/카운터의 업 플래그의 I/O 메모리 실제 주소는 이 명령이 아니라 MOVR 명령으로 설정합니다.

샘플 프로그램

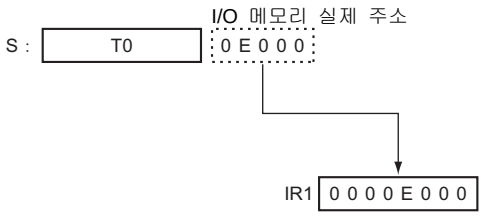


0.00이 ON이면 20CH의 I/O 메모리 실제 주소를 인덱스 레지스터 IR0에 저장합니다.





0.00이 ON이면 타이머 현재값 T0000의 I/O 메모리 실제 주소를 인덱스 레지스터 IR1에 저장합니다.



구체적인 I/O 메모리 실제 주소의 값은 'SYSMAC CS/CJ 시리즈 사용 설명서 설치편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'에 기재되어 있습니다.

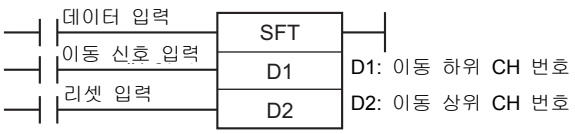
데이터 이동 명령

니모닉	명령 이름	펑션 번호	페이지
SFT	이동 레지스터	010	3-176
SFTR	좌우 이동 레지스터	084	3-178
ASFT	비동기 이동 레지스터	017	3-180
WSFT	워드 이동	016	3-182
ASL	1비트 좌측 이동	025	3-184
ASLL	1비트 좌측 이동(2워드)	570	
ASR	1비트 우측 이동	026	3-186
ASRL	1비트 우측 이동(2워드)	571	
ROL	CY가 있는 1비트 좌측 회전	027	3-188
ROLL	CY가 있는 1비트 좌측 회전(2워드)	572	
RLNC	CY가 없는 1비트 좌측 회전	574	3-190
RLNL	CY가 없는 1비트 좌측 회전(2워드)	576	
ROR	CY가 있는 1비트 우측 회전	028	3-192
RORL	CY가 있는 1비트 우측 회전(2워드)	573	
RRNC	CY가 없는 1비트 우측 회전	575	3-194
RRNL	CY가 없는 1비트 우측 회전(2워드)	577	
SLD	1자리 좌측 이동	074	3-196
SRD	1자리 우측 이동	075	
NSFL	N비트 데이터 좌측 이동	578	3-198
NSFR	N비트 데이터 우측 이동	579	
NASL	N비트 좌측 이동	580	3-200
NSLL	N비트 2워드 좌측 이동	582	
NASR	N비트 우측 이동	581	3-203
NSRL	N비트 2워드 우측 이동	583	

SFT

명령 이름	니모닉	동작 옵션	평션 번호	기능
이동 레지스터	SFT	−	010	이동 레지스터 동작을 수행합니다.

심볼	SFT



특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능*1	불가능	가능	가능	가능	가능

*1: 평션 블록 인스턴스 할당 영역에 CIO, 내부 보조 릴레이(WR), 유지 릴레이(HR)가 지정된 경우만 가능

피연산자 설명

피연산자	내용	데이터 형식	크기
D1	이동 하위 CH 번호	UINT	가변
D2	이동 상위 CH 번호	UINT	가변

■ 영역 종류

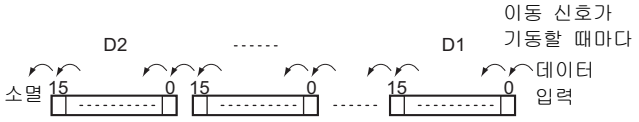
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
D1	○	○	○	−	−	−	−	−	−	−	−	−	−	○	−	−	−	−
D2																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 간접 IR 지정으로 D1, D2의 주소가 CIO, WR, HR 이외의 영역이 되었을 때 ON · 그 이외일 때 OFF

기능

이동 신호 입력을 기동(OFF→ON)할 때, D1에서 D2까지 모두 좌측 방향(최하위 비트→최상위 비트)으로 1비트 이동하여 최하위 비트에 데이터 입력의 ON/OFF 내용을 반영합니다.



사용상의 주의 사항

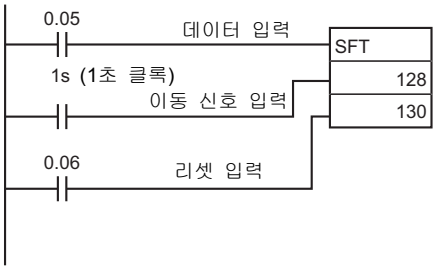
· D1, D2의 범위가 중복된 복수의 SFT 명령은 사용할 수 없습니다. (사용한 경우, 결과가 일정치 않습니다.)
· D, D2는 동일 영역 종류이어야 합니다.
· 이동 범위에서 오버 플로된 비트의 내용은 소멸됩니다.
· 리셋 입력 ON인 경우, D1로 지정한 이동 하위 CH 번호부터 D2로 지정한 이동 상위 CH 번호까지 리셋(=0)합니다. 리셋 입력은 다른 입력에 우선합니다.
· 이동 범위 설정은 기본적으로 D1≦D2이지만 D1>D2로 지정된 경우도 에러가 되지 않고 D1만 1채널(워드) 이동합니다.

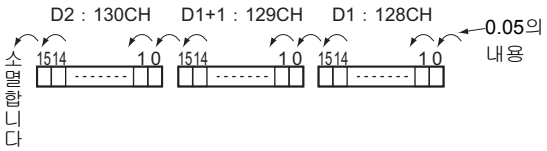
샘플 프로그램

■ 16비트를 초과하는 이동 레지스터 예

128~130CH의 48비트를 사용한 이동 레지스터입니다.

이동 신호 입력용으로 클록 펄스 1s를 사용하면 1초마다 입력 릴레이 0.05의 내용이 128.00~130.15로 이동 됩니다.

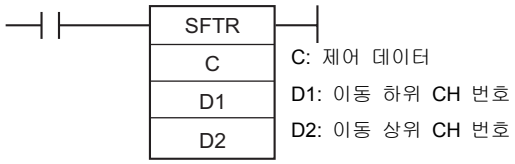




SFTR

명령 이름	니모닉	동작 옵션	평션 번호	기능
좌우 이동 레지스터	SFTR	@SFTR	084	이동 방향을 전환할 수 있는 이동 레지스터 동작을 수행합니다.

심볼	SFTR



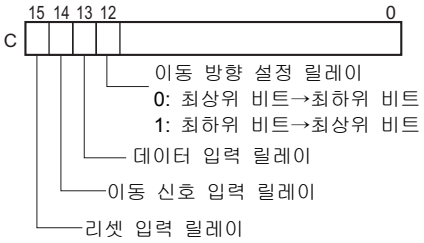
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	UINT	1
D1	이동 하위 CH 번호	UINT	가변
D2	이동 상위 CH 번호	UINT	가변

C: 제어 데이터



주: D1, D2는 동일 영역 종류이어야 합니다.

■ 영역 종류

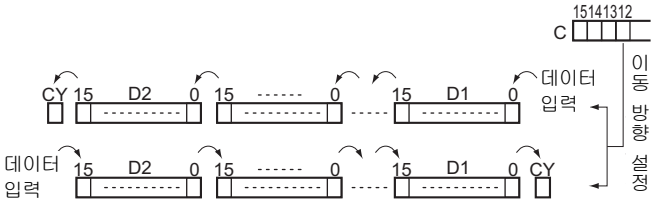
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
D1												–						
D2																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· D1 > D2일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 캐리(CY) 플래그에 1이 이동되었을 때 ON · 캐리(CY) 플래그에 0이 이동되었을 때 OFF · 리셋 입력이 1일 때 OFF

기능

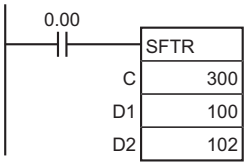
이동 신호 입력 릴레이(C의 비트 14)가 ON일 때 D1에서 D2까지 이동 방향 설정 릴레이(C의 비트 12)로 지정한 방향으로 1비트 이동하고, 최하위 비트 또는 최상위 비트에 데이터 입력 릴레이(C의 비트 13)의 ON/OFF 내용을 반영합니다. 이동 범위에서 오버 플로된 비트의 내용은 캐리 플래그(CY)에 반영됩니다.



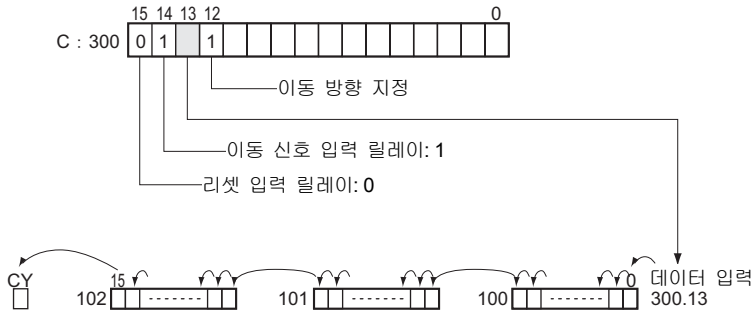
주:

- 위의 이동 동작은 리셋 입력 릴레이(C의 비트 15) OFF 시의 조건입니다.
- 리셋 입력 릴레이(C의 비트 15) ON 시에는 D1에서 D2까지 모두 리셋(= 0)합니다.

샘플 프로그램



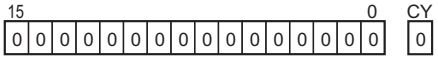
- 이동 동작
  리셋 입력 릴레이 300.15가 OFF인 경우 0.00이 ON인 상태에서 이동 신호 입력 릴레이 300.14가 ON이면, 300.12가 지정하는 방향(예 1: 최하위 비트→최상위 비트 방향)으로 100~102CH의 3CH를 1비트 이동하여 최하위 비트 100.00에 데이터 입력 릴레이 300.13의 내용을 설정합니다.
  오버 플로된 102.15의 내용은 CY(캐리 플래그)로 설정됩니다.



- 리셋 동작
  리셋 입력 릴레이 300.15가 ON인 경우 0.00이 ON인 상태에서 300.14가 ON이면 100~102CH의 3CH와 CY 플래그를 모두 OFF로 리셋합니다.

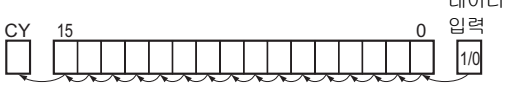
■ 제어 데이터의 내용과 동작

(1) 리셋 시…리셋 입력 릴레이(C의 비트 15)가 ON일 때



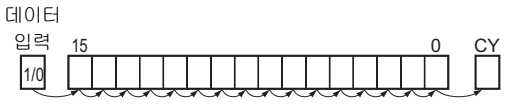
D1~D2CH의 모든 비트 내용 및 CY 플래그는 0이 되고 다른 입력은 수신되지 않습니다.

(2) 좌측 이동 시(최상위 비트←최하위 비트)…이동 방향 설정 릴레이(C의 비트 12)가 ON일 때



이동 신호 입력 릴레이(C의 비트 14)가 ON일 때 데이터 입력 릴레이(C의 비트 13)의 내용(1/0)이 D1CH의 비트 0으로 이동하고, 각 비트의 내용은 각각 1비트씩 상위 비트 측으로, D2CH의 비트 15의 내용은 CY 플래그로 이동합니다.

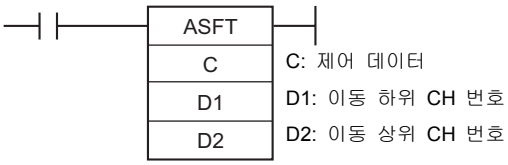
(3) 우측 이동 시(최상위 비트→최하위 비트)…이동 방향 설정 릴레이(C의 비트 12)가 OFF일 때



이동 신호 입력 릴레이(C의 비트 14)가 ON일 때 데이터 입력 릴레이(C의 비트 13)의 내용(1/0)이 D2CH의 비트 15로 이동하고, 각 비트의 내용은 각각 1비트씩 하위 비트 측으로, D1CH의 비트 0의 내용은 CY 플래그로 이동합니다.

ASFT

명령 이름	니모닉	동작 옵션	평션 번호	기능
비동기 이동 레지스터	ASFT	@ASFT	017	지정 CH 범위의 채널 데이터 중 16진수 #0000 이외의 채널 데이터에 대해 앞 또는 뒤 채우기를 실행하고 16진수 #0000의 채널 데이터와 위치를 교체합니다.

심볼	ASFT
	

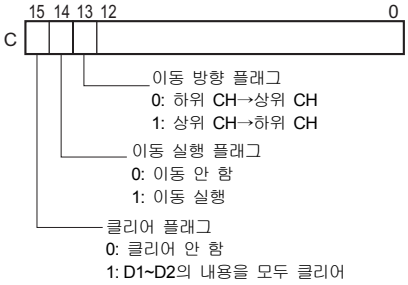
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	UINT	1
D1	이동 하위 CH 번호	UINT	가변
D2	이동 상위 CH 번호	UINT	가변

C: 제어 데이터



주: D1, D2는 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
D1												–						
D2																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· D1 > D2일 때 ON · 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF

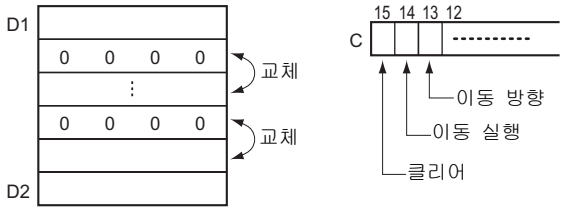
기능

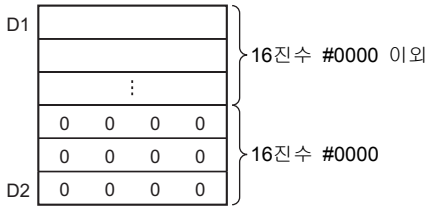
이동 실행 플래그(C의 비트 14)가 ON일 때 D1에서 D2 범위의 채널 데이터 중, 0 이외의 데이터를 이동 방향 플래그(C의 비트 13)로 지정한 방향으로 채우므로 명령 실행 1회마다 0 이외의 데이터와 인접한 0의 채널 데이터를 교체합니다.

이렇게 하면 D1에서 D2 범위의 데이터를 0과 0 이외의 데이터로 구분할 수 있습니다.

주:

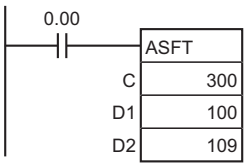
· 클리어 플래그(C의 비트 15)가 ON일 때 D1에서 D2까지의 범위를 0으로 클리어합니다. 클리어 플래그는 이동 실행 플래그(C의 비트 14)보다 우선됩니다.





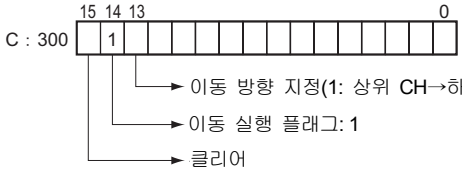
이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

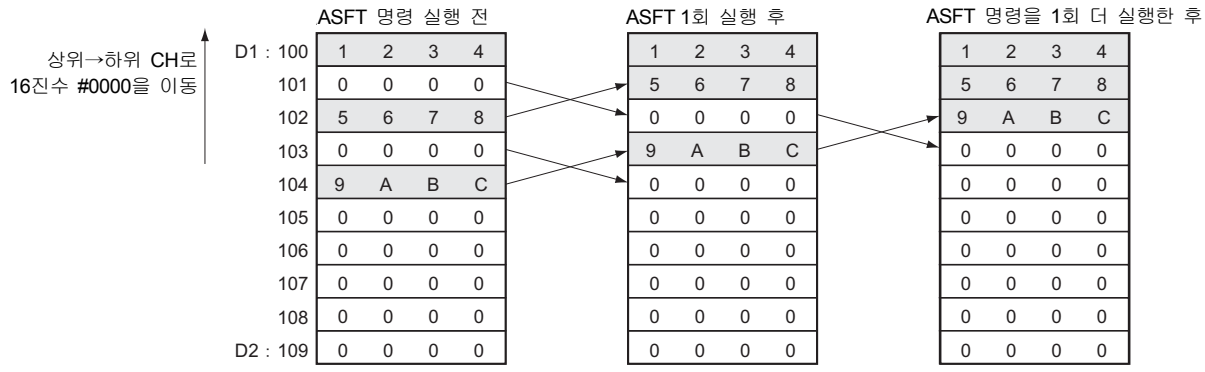
샘플 프로그램



· 이동 동작

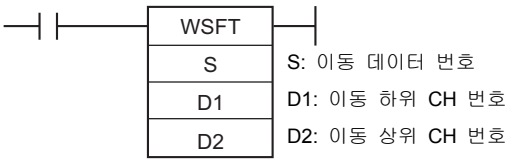
0.00이 ON인 상태에서 이동 실행 플래그 300.14가 ON이면 100~109CH의 10CH 중 16진수 #0000 이외의 채널 데이터를 이동 방향 플래그 300.13에서 지정한 방향(예: 1: 상위→하위)으로 이동합니다. (16진수 #0000 이외에 하위가 되는 채널에 있는 경우, 16진수 #0000과 해당 채널은 서로 교체됩니다.)





WSFT

명령 이름	니모닉	동작 옵션	평션 번호	기능
워드 이동	WSFT	@WSFT	016	CH 데이터 단위의 이동 동작을 나타냅니다.

심볼	WSFT
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	이동 데이터 번호	WORD	1
D1	이동 하위 CH 번호	UINT	가변
D2	이동 상위 CH 번호	UINT	가변

■ 영역 종류

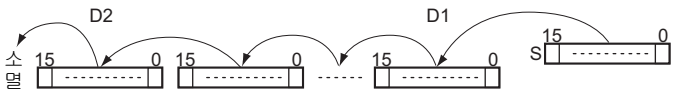
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
D1											–	–						
D2																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· D1 > D2일 때 ON · 그 이외일 때 OFF

기능

D1에서 D2까지 상위 CH로 1워드씩 이동하여 최하위 CH(D1)에 S로 지정한 데이터를 출력합니다. 이때 원래의 최상위 CH(D2)의 데이터는 소멸됩니다.

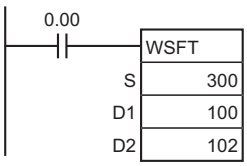


사용상의 주의 사항

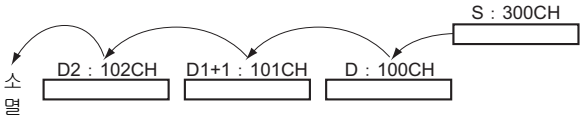
· D, D2는 동일 영역 종류이어야 합니다.

· 대량의 데이터를 이동하는 경우 명령 실행 시간이 걸립니다. 따라서 이 명령 실행 시에 전원 OFF가 발생한 경우 이동 동작이 도중에 실행 종료되는 경우가 있으므로 주의하십시오.

샘플 프로그램

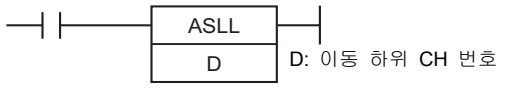


0.00이 ON이면 100~102CH까지 1워드(CH)씩 상위 쪽으로 이동합니다.
100CH에는 300CH의 내용이 저장되고 102CH의 내용은 소멸됩니다.



ASL/ASLL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
1비트 좌측 이동	ASL	@ASL	025	CH 데이터를 왼쪽으로 1비트 이동합니다.
1비트 좌측 이동(2워드)	ASLL	@ASLL	570	2CH분의 데이터를 왼쪽으로 1비트 이동합니다.

심볼	ASL	ASLL
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		ASL	ASLL	ASL	ASLL
D	ASL: 이동 CH 번호 ASLL: 이동 하위 CH 번호	UINT	UDINT	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	글록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
ASL	D	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
ASLL	D												–						

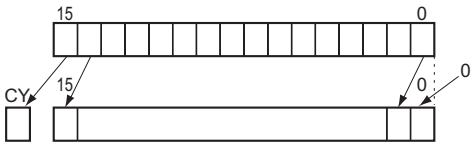
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 이동 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 캐리(CY) 플래그에 1이 이동되었을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 이동 결과에 의해 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF

기능

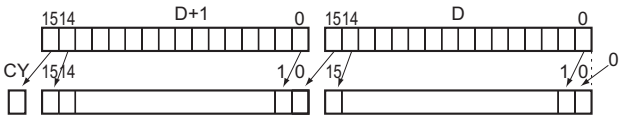
■ ASL

D를 왼쪽 방향(최하위 비트→최상위 비트)으로 1비트 이동합니다. 최하위 비트에는 0이 설정됩니다. 최상위 비트는 캐리 플래그(CY)로 이동됩니다.

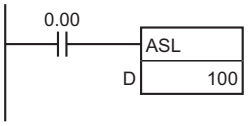


■ ASLL

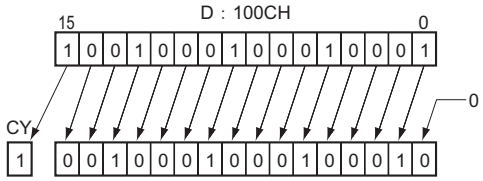
D를 2워드 데이터로 하여 왼쪽 방향(최하위 비트→최상위 비트)으로 1비트 이동합니다. DCH의 최하위 비트에는 0이 설정되고, D+1CH의 최상위 비트는 캐리 플래그(CY)로 이동됩니다.

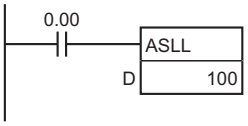


샘플 프로그램

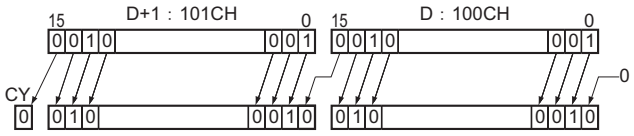


0.00이 ON이면 100CH를 왼쪽으로 1비트 이동합니다. 100.00에는 0이 설정되고 100.15의 내용이 CY 플래그에 설정됩니다.



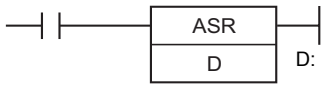
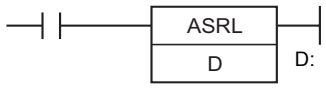


0.00이 ON이면 100CH와 101CH를 왼쪽으로 1비트 이동합니다. 100.00에는 0이 설정되고 101.15의 내용이 CY 플래그에 설정됩니다.



ASR/ASRL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
1비트 우측 이동	ASR	@ASR	026	CH 데이터를 오른쪽으로 1비트 이동합니다.
1비트 우측 이동 (2워드)	ASRL	@ASRL	571	2CH분의 데이터를 오른쪽으로 1비트 이동합니다.

심볼	ASR	ASRL
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		ASR	ASRL	ASR	ASRL
D	ASR: 이동 CH 번호 ASRL: 이동 하위 CH 번호	UINT	UDINT	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
ASR	D	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
ASRL	D												–						

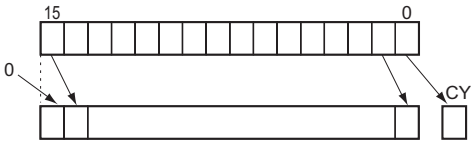
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 이동 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 캐리(CY) 플래그에 1이 이동되었을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	OFF

기능

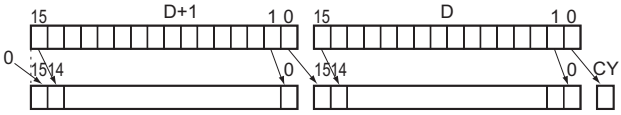
■ ASR

D를 오른쪽 방향(최상위 비트→최하위 비트)으로 1비트 이동합니다. 최상위 비트에는 0이 설정됩니다. 최하위 비트는 캐리 플래그(CY)로 이동됩니다.

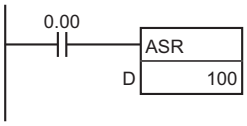


■ ASRL

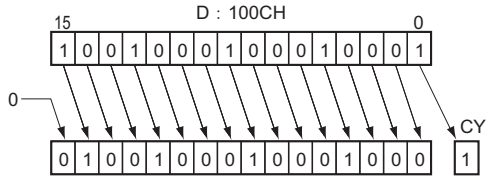
D를 2워드 데이터로 하여 오른쪽 방향(최상위 비트→최하위 비트)으로 1비트 이동합니다. D+1CH의 최상위 비트에는 0이 설정되고, DCH의 최하위 비트는 캐리 플래그(CY)로 이동됩니다.

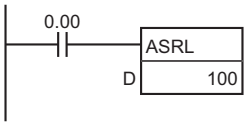


샘플 프로그램

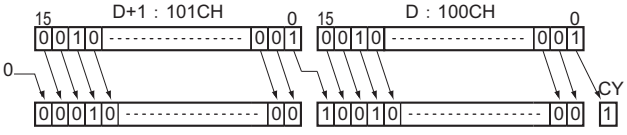


0.00이 ON이면 100CH를 오른쪽으로 1비트 이동합니다. 100.00에는 0이 설정되고 100.15의 내용이 CY 플래그에 설정됩니다.



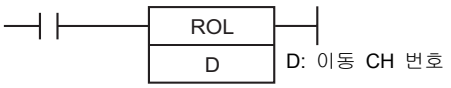
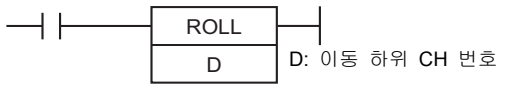


0.00이 ON이면 100CH와 101CH를 오른쪽으로 1비트 이동합니다. 100.00에는 0이 설정되고 100.15의 내용이 CY 플래그에 설정됩니다.



ROL/ROLL

명령 이름	니모닉	동작 옵션	평션 번호	기능
CY가 있는 1비트 좌측 회전	ROL	@ROL	027	16비트의 채널 데이터를 캐리(CY) 플래그를 포함하여 1비트 좌측으로 회전합니다.
CY가 있는 1비트 좌측 회전(2워드)	ROLL	@ROLL	572	32비트의 채널 데이터를 캐리(CY) 플래그를 포함하여 1비트 좌측으로 회전합니다.

심볼	ROL	ROLL
		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		ROL	ROLL	ROL	ROLL
D	ROL: 이동 CH 번호 ROLL: 이동 하위 CH 번호	UINT	UDINT	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
ROL	D	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
ROLL	D												–						

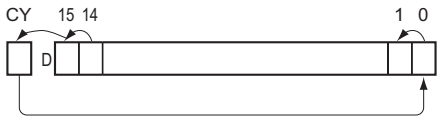
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 이동 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 캐리(CY) 플래그에 1이 이동되었을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 이동 결과에 의해 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF

기능

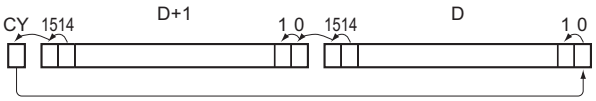
■ ROL

캐리(CY) 플래그도 포함하여 D를 왼쪽 방향(최하위 비트→최상위 비트)으로 1비트 회전합니다.



■ ROLL

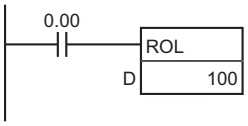
D를 2워드 데이터로 하여 캐리(CY) 플래그도 포함하여 왼쪽 방향(최하위 비트→최상위 비트)으로 1비트 회전합니다.



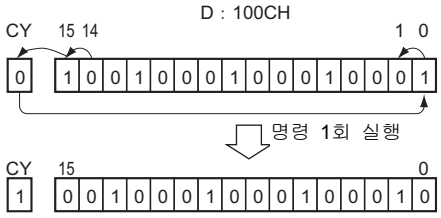
힌트

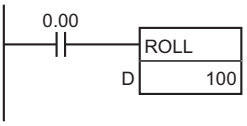
· 이 명령 직전에 세트 캐리(STC), 클리어 캐리(CLC) 명령을 사용하여 캐리(CY) 플래그의 내용을 1, 0으로 설정할 수 있습니다.

샘플 프로그램

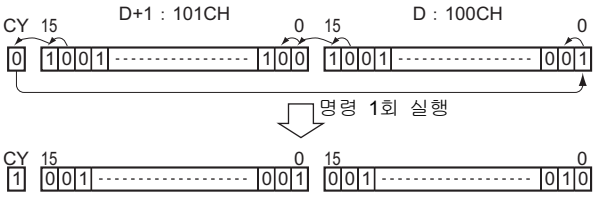


0.00이 ON이면 CY 플래그를 포함하여 100CH를 왼쪽으로 1비트 회전합니다.
100.15의 내용이 CY 플래그에 설정되고 CY 플래그의 내용은 100.00에 설정됩니다.



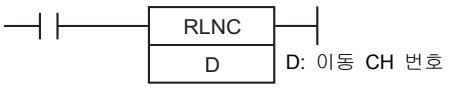
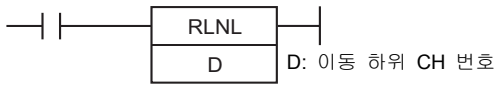


0.00이 ON이면 CY 플래그를 포함하여 100CH와 101CH를 왼쪽으로 1비트 회전합니다.
101.15의 내용이 CY 플래그에 설정되고 CY 플래그의 내용은 100.00에 설정됩니다.



RLNC/RLNL

명령 이름	니모닉	동작 옵션	평션 번호	기능
CY가 없는 1비트 좌측 회전	RLNC	@RLNC	574	16비트의 채널 데이터를 캐리(CY) 플래그를 포함하지 않고 1비트 좌측으로 회전합니다.
CY가 없는 1비트 좌측 회전(2워드)	RLNL	@RLNL	576	32비트의 채널 데이터를 캐리(CY) 플래그를 포함하지 않고 1비트 좌측으로 회전합니다.

심볼	RLNC	RLNL
		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		RLNC	RLNL	RLNC	RLNL
D	RLNC: 이동 CH 번호 RLNL: 이동 하위 CH 번호	UINT	UDINT	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
RLNC	D	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
RLNL	D												–						

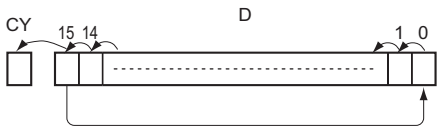
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 이동 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 캐리(CY) 플래그에 1이 이동되었을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 이동 결과에 의해 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF

기능

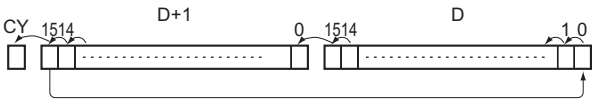
■ RLNC

D를 왼쪽 방향(최하위 비트→최상위 비트)으로 1비트 회전합니다. D의 최상위 비트의 데이터는 최하위 비트로 이동하고, 동시에 CY 플래그에도 출력됩니다.

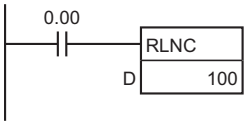


■ RLNL

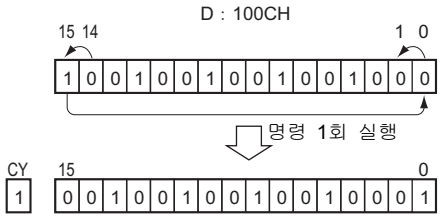
D를 2워드 데이터로 하여 모두 왼쪽 방향(최하위 비트→최상위 비트)으로 1비트 이동합니다. D+1의 최상위 비트의 데이터는 D의 최하위 비트로 이동하고, 동시에 CY 플래그에도 출력됩니다.

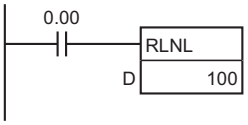


샘플 프로그램

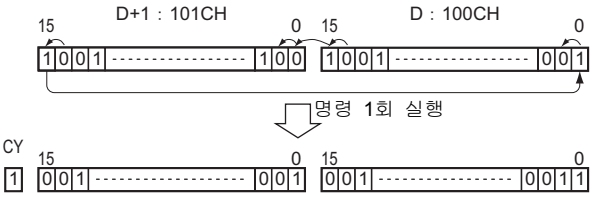


0.00이 ON이면 CY 플래그를 포함하지 않고 100CH를 왼쪽으로 1비트 회전합니다.
100.15의 내용이 100.00에 설정됩니다.





0.00이 ON이면 CY 플래그를 포함하지 않고 100CH와 101CH를 왼쪽으로 1비트 회전합니다.
101.15의 내용이 100.00에 설정됩니다.



ROR/RORL

명령 이름	니모닉	동작 옵션	평션 번호	기능
CY가 있는 1비트 우측 회전	ROR	@ROR	028	16비트의 채널 데이터를 캐리(CY) 플래그를 포함하여 1비트 우측으로 회전합니다.
CY가 있는 1비트 우측 회전(2워드)	RORL	@RORL	573	32비트의 채널 데이터를 캐리(CY) 플래그를 포함하여 1비트 우측으로 회전합니다.

심볼	ROR	RORL
	ROR D D: 이동 CH 번호	RORL D D: 이동 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		ROR	RORL	ROR	RORL
D	ROR: 이동 CH 번호 RORL: 이동 하위 CH 번호	UINT	UDINT	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	글록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
ROL	D	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
ROLL	D												–						

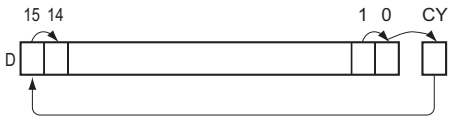
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 이동 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 캐리(CY) 플래그에 1이 이동되었을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 이동 결과에 의해 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF

기능

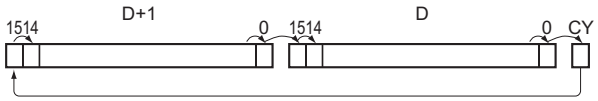
■ ROR

캐리(CY) 플래그도 포함하여 D를 오른쪽 방향(최상위 비트→최하위 비트)으로 1비트 회전합니다.



■ RORL

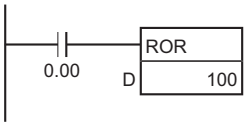
D를 2워드 데이터로 하여 캐리(CY) 플래그도 포함하여 오른쪽 방향(최상위 비트→최하위 비트)으로 1비트 회전합니다.



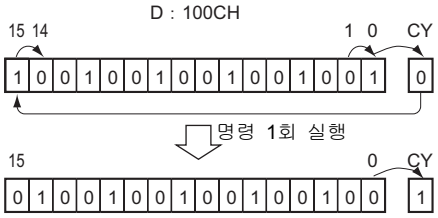
힌트

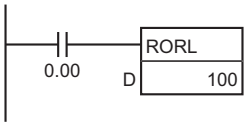
· 이 명령 직전에 세트 캐리(STC), 클리어 캐리(CLC) 명령을 사용하여 캐리(CY) 플래그의 내용을 1, 0으로 설정할 수 있습니다.

샘플 프로그램

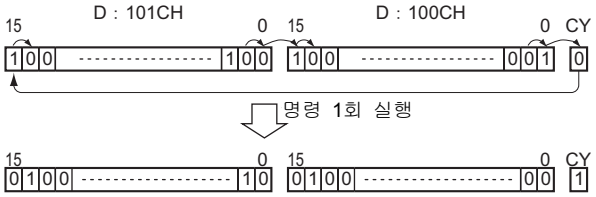


0.00이 ON이면 CY 플래그를 포함하여 100CH를 오른쪽으로 1비트 회전합니다.
100.00의 내용이 CY 플래그에 설정되고 CY 플래그의 내용은 100.15에 설정됩니다.



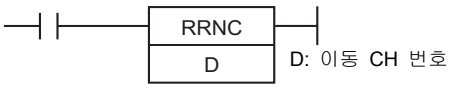
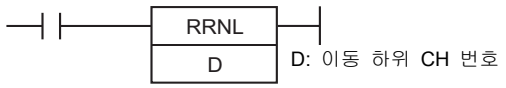


0.00이 ON이면 CY 플래그를 포함하여 100CH와 101CH를 오른쪽으로 1비트 회전합니다.
101.00의 내용이 CY 플래그에 설정되고 CY 플래그의 내용은 101.15에 설정됩니다.



RRNC/RRNL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
CY가 없는 1비트 우측 회전	RRNC	@RRNC	575	16비트의 채널 데이터를 캐리(CY) 플래그를 포함하지 않고 1비트 우측으로 회전합니다.
CY가 없는 1비트 우측 회전(2워드)	RRNL	@RRNL	577	32비트의 채널 데이터를 캐리(CY) 플래그를 포함하지 않고 1비트 우측으로 회전합니다.

심볼	RRNC	RRNL
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		RRNC	RRNL	RRNC	RRNL
D	RRNC: 이동 CH 번호 RRNL: 이동 하위 CH 번호	UINT	UDINT	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	글록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
RRNC	D	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
RRNL	D												–						

조건 플래그의 동작

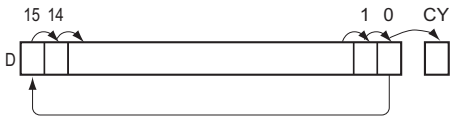
이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 이동 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 캐리(CY) 플래그에 1이 이동되었을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 이동 결과에 의해 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF

기능

■ RRNC

D를 오른쪽 방향(최상위 비트→최하위 비트)으로 1비트 회전합니다(캐리(CY) 플래그 포함 안 함).

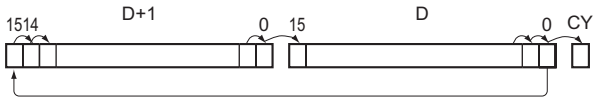
D의 최상위 비트의 데이터는 최하위 비트로 이동하고, 동시에 CY 플래그에도 출력합니다.



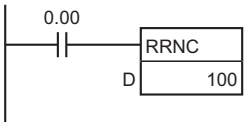
■ RRNL

D를 2워드 데이터로 하여 모두 오른쪽 방향(최상위 비트→최하위 비트)으로 1비트 이동합니다(캐리(CY) 플래그 포함 안 함).

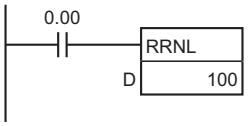
D+1의 최상위 비트의 데이터는 D의 최하위 비트로 이동하고, 동시에 CY 플래그에도 출력됩니다.



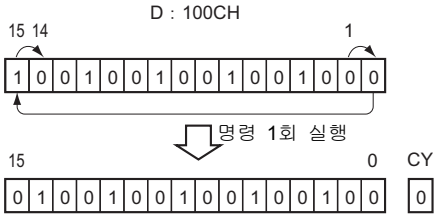
샘플 프로그램

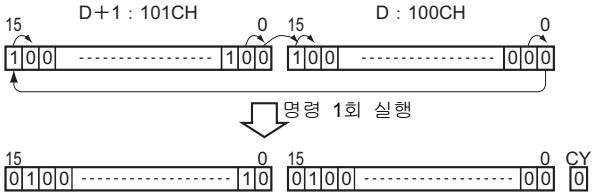


0.00이 ON이면 CY 플래그를 포함하지 않고 100CH를 오른쪽으로 1비트 회전합니다.
100.00의 내용이 100.15에 설정됩니다.



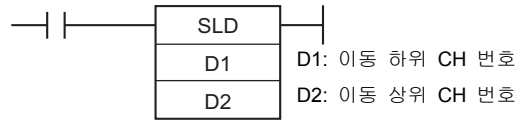
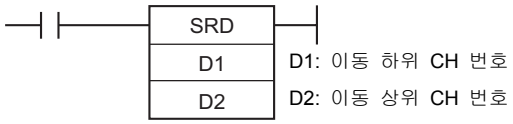
0.00이 ON이면 CY 플래그를 포함하지 않고 100CH와 101CH를 오른쪽으로 1비트 회전합니다.
100.00의 내용이 101.15에 설정됩니다.





SLD/SRD

명령 이름	니모닉	동작 옵션	평션 번호	기능
1자리 좌측 이동	SLD	@SLD	074	연속된 채널의 데이터 내용을 1자리(4비트) 단위로 좌측 이동합니다.
1자리 우측 이동	SRD	@SRD	075	연속된 채널의 데이터 내용을 1자리(4비트) 단위로 우측 이동합니다.

심볼	SLD	SRD
		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
D1	이동 하위 CH 번호	UINT	가변
D2	이동 상위 CH 번호	UINT	가변

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
D	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· D1 > D2일 때 ON · 그 이외일 때 OFF

기능

■ SLD

D1에서 D2까지의 범위를 자리(4비트) 단위로 상위 자리 쪽으로 이동합니다. 이때 최하위 자리(D1의 비트 3-0)에는 0이 들어가고, 원래의 최상위 자리(D2의 비트 15-12)의 데이터는 소멸됩니다.

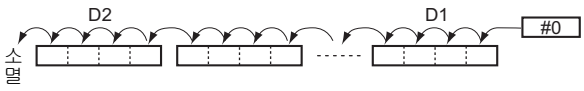
■ SRD

D1에서 D2까지의 범위를 자리(4비트) 단위로 하위 자리 쪽으로 이동합니다. 이때 최상위 자리(D2의 비트 15-12)에는 0이 들어가고, 원래의 최하위 자리(D1 비트 3-0)의 데이터는 소멸됩니다.

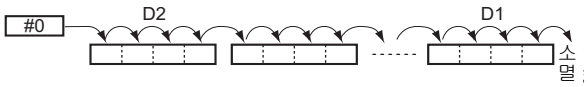
사용상의 주의 사항

- D, D2는 동일 영역 종류이어야 합니다.
- 대량의 데이터를 이동하는 경우 명령 실행 시간이 걸립니다. 따라서 이 명령 실행 시에 전원 OFF가 발생하는 경우 이동 동작 도중에 실행 종료될 수 있으므로 주의하십시오.

- SLD

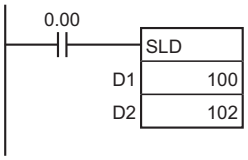


- SRD

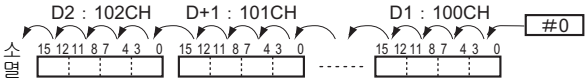


샘플 프로그램

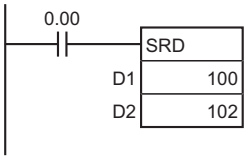
■ SLD



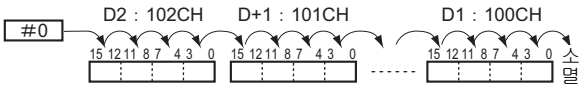
0.00이 ON이면 100~102의 각 자리(4비트)를 상위 쪽으로 이동합니다.
100CH의 비트 0~3에 #0이 설정되고 102CH의 비트 12~15는 소멸됩니다.



■ SRD

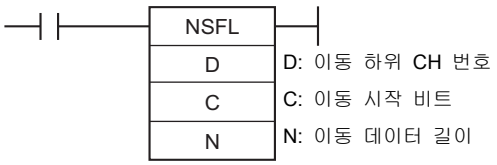
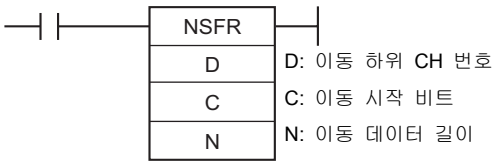


0.00이 ON이면 100~102의 각 자리(4비트)를 하위 쪽으로 이동합니다.
102CH의 비트 12~15에 #0이 설정되고 100CH의 비트 0~3은 소멸됩니다.



NSFL/NSFR

명령 이름	니모닉	동작 옵션	평션 번호	기능
N비트 데이터 좌측 이동	NSFL	@NSFL	578	지정된 비트 수 분량의 비트 데이터를 좌측으로 1비트 이동합니다.
N비트 데이터 우측 이동	NSFR	@NSFR	579	지정된 비트 수 분량의 비트 데이터를 우측으로 1비트 이동합니다.

	NSFL	NSFR
심볼		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
D	이동 하위 CH 번호	UINT	가변
C	이동 시작 비트	UINT	1
N	이동 데이터 길이	UINT	1

C: 이동 시작 비트
10진수 &0~15 또는 16진수 #0000~000F

N: 이동 데이터 길이
10진수 &0~65535 또는 16진수 #0000~FFFF

주: D~D+최대 65535CH의 이동 대상은 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
D	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
C											○	○						
N																		

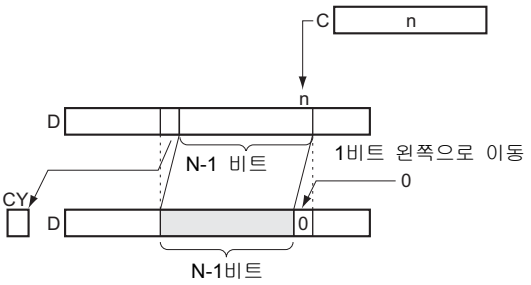
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C의 데이터가 10진수 &0~15 또는 16진수 #0000~000F의 범위에 없을 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 캐리(CY) 플래그에 1이 이동되었을 때 ON · 그 이외일 때 OFF

기능

■ NSFL

D로 지정한 이동 하위 CH의 이동 시작 비트(C)부터 이동 데이터 길이(N)만큼의 비트 데이터를 모두 왼쪽 방향(상위 CH, 상위 비트 측)으로 1비트 이동합니다. 이때 이동 시작 비트에는 0이 들어가고, 이동 범위의 최상위 비트의 내용은 캐리(CY) 플래그로 이동합니다.

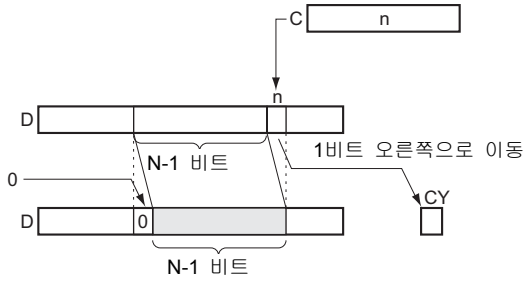


주:

- 이동 데이터 길이(N)가 0인 경우, 이동 시작 비트의 데이터를 캐리(CY) 플래그에 복사합니다. 이동 시작 비트의 데이터 내용은 변경되지 않습니다.
- 이동 범위의 최하위 CH 및 최상위 CH의 데이터는 이동 대상 비트 이외에는 변경되지 않습니다.

■ NSFR

D로 지정한 이동 하위 CH의 이동 시작 비트(C)부터 이동 데이터 길이(N)만큼의 비트 데이터를 모두 오른쪽 방향(하위 CH, 하위 비트 측)으로 1비트 이동합니다. 이때 이동 시작 비트에는 0이 들어가고, 이동 범위의 최하위 비트의 내용은 캐리(CY) 플래그로 이동합니다.

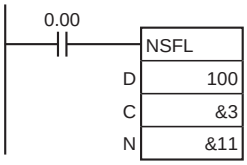


주:

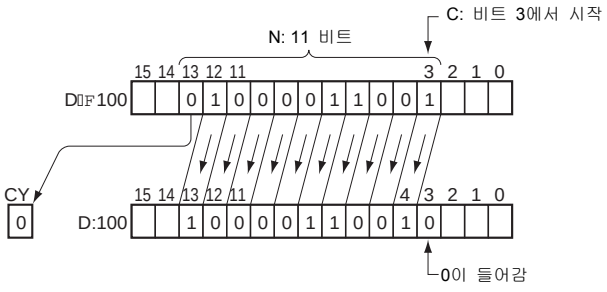
- 이동 데이터 길이(N)가 0인 경우, 이동 시작 비트의 데이터를 캐리(CY) 플래그에 복사합니다. 이동 시작 비트의 데이터 내용은 변경되지 않습니다.
- 이동 범위의 최하위 CH 및 최상위 CH의 데이터는 이동 대상 비트 이외에는 변경되지 않습니다.

샘플 프로그램

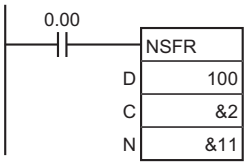
■ NSFL



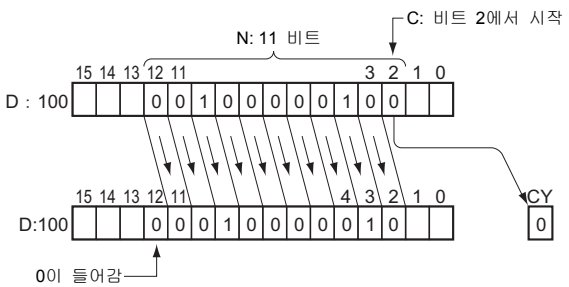
0.00이 ON이면 이동 시작 위치 비트 3부터 이동 데이터 길이 11비트(16진수 #000B)를 모두 왼쪽 방향(최하위 비트→최상위 비트)으로 1비트 이동합니다.



■ NSFR



0.00이 ON이면 이동 시작 위치 비트 2부터 이동 데이터 길이 11비트(16진수 #000B)를 모두 오른쪽 방향(최상위 비트→최하위 비트)으로 1비트 이동합니다.
이동 시작 비트인 100CH의 비트 12에는 0이 들어갑니다.
이동 대상 범위의 최하위 비트인 100CH의 비트 2의 내용은 CY 플래그에 복사됩니다.



NASL/NSLL

명령 이름	니모닉	동작 옵션	평션 번호	기능
N비트 좌측 이동	NASL	@NASL	580	16비트의 채널 데이터를 지정 비트 수만큼 좌측으로 이동합니다.
N비트 2워드 좌측 이동	NSLL	@NSLL	582	32비트의 채널 데이터를 지정 비트 수만큼 좌측으로 이동합니다.

심볼	NASL	NSLL
	NASL D: 이동 CH 번호 C: 제어 데이터	NSLL D: 이동 하위 CH 번호 C: 제어 데이터

특정 영역 내의 사용 여부

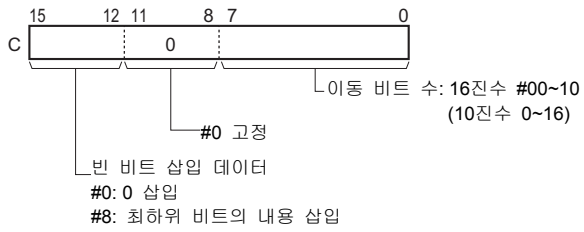
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

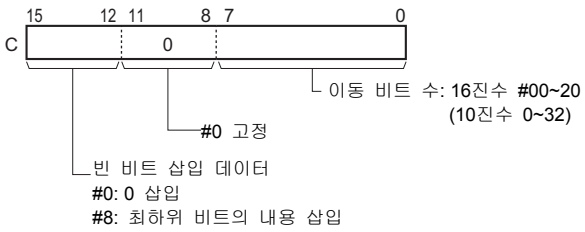
피연산자	내용	데이터 형식		크기	
		NASL	NSLL	NASL	NSLL
D	NASL: 이동 CH 번호 NSLL: 이동 하위 CH 번호	UINT	UDINT	1	2
C	제어 데이터	UINT	UINT	1	1

C: 제어 데이터

· NASL



· NSLL



■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
NASL	D	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
	C											○							
NSLL	D	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
	C											○	○						

조건 플래그의 동작

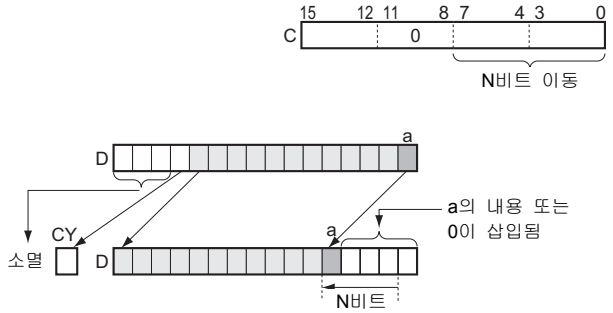
이름	라벨	내용
에러 플래그	P_ER	· 제어 데이터 C(이동 비트 수)가 범위 밖일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 이동 결과가 0일 때 ON · 그 이외일 때 OFF

이름	라벨	내용
캐리 플래그	P_CY	· 캐리(CY) 플래그에 1이 이동되었을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 이동 결과에 의해 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF

기능

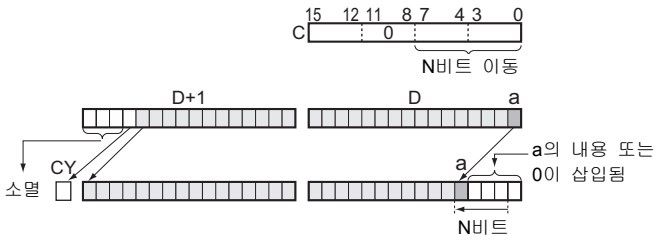
■ NASL

C로 지정한 비트 수(BIN)만큼 왼쪽 방향(최하위 비트→최상위 비트)으로 D를 이동합니다. 이때, 지정된 CH의 최하위 비트부터 이동 비트 수 분량의 비트에는 빈 비트 삽입 데이터(C)에 따른 데이터가 들어갑니다.



■ NSLL

D를 2워드 데이터로 하여, C로 지정한 비트 수(BIN)만큼 모두 왼쪽 방향(최하위 비트→최상위 비트)으로 이동합니다. 이때, 지정된 CH의 최하위 비트부터 이동 비트 수 분량의 비트에는 빈 비트 삽입 데이터(C)에 따른 데이터가 들어갑니다.

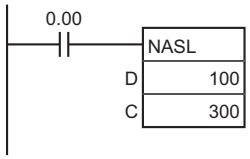


사용상의 주의 사항

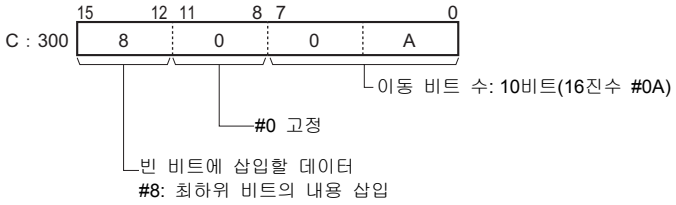
- 지정된 CH에서 오버 플로된 비트의 경우 마지막 비트의 내용이 캐리(CY) 플래그로 이동되고, 그 이외일 때는 소멸됩니다.
- 이동 비트 수(C)가 0인 경우에는 이동 동작이 수행되지 않습니다. 단, 지정된 채널의 데이터에 따라 각 플래그를 ON/OFF합니다.

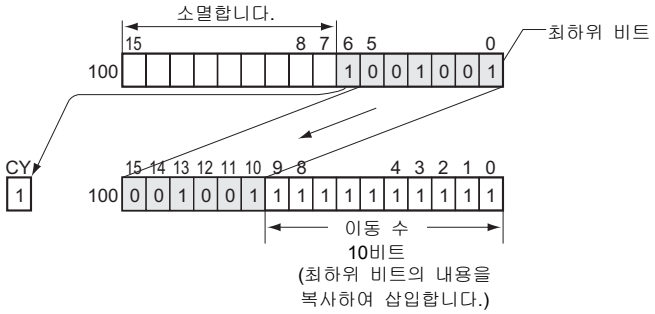
NASL/NSLL

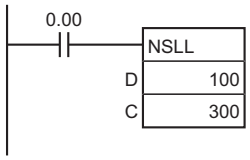
샘플 프로그램



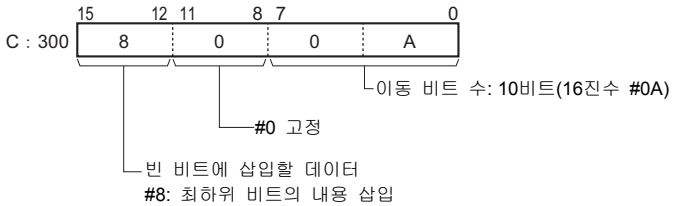
0.00이 ON이면 100CH의 데이터를 300CH(제어 데이터)의 비트 0~7에서 지정한 비트 수(예: 10비트)만큼 왼쪽 방향(최하위 비트→최상위 비트)으로 이동합니다.
빈 비트에는 100CH의 비트 0의 내용을 복사하여 삽입합니다.
오버 플로된 비트는 최하위 비트가 CY에 설정되고 나머지는 소멸됩니다.

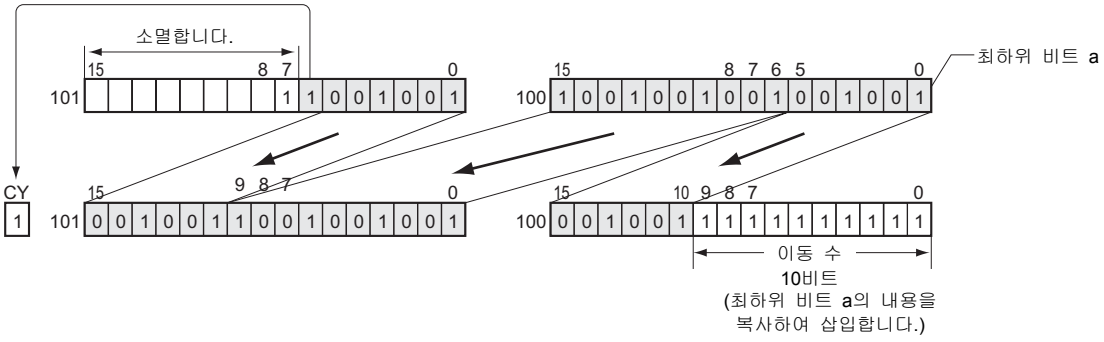






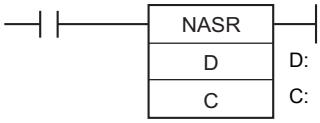
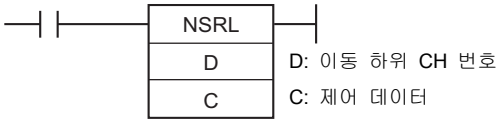
0.00이 ON이면 100CH~101CH의 2CH 데이터를 300CH(제어 데이터)의 비트 0~7에서 지정한 비트 수(예: 10비트)만큼 왼쪽 방향(최하위 비트→최상위 비트)으로 이동합니다.
빈 비트에는 100CH의 비트 0의 내용이 복사되어 삽입됩니다.
오버 플로된 비트는 최하위 비트가 CY에 설정되고 나머지는 소멸됩니다.





NASR/NSRL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
N비트 우측 이동	NASR	@NASR	581	16비트의 채널 데이터를 지정 비트 수만큼 우측으로 이동합니다.
N비트 2워드 우측 이동	NSRL	@NSRL	583	32비트의 채널 데이터를 지정 비트 수만큼 우측으로 이동합니다.

심볼	NASR	NSRL
		

특정 영역 내의 사용 여부

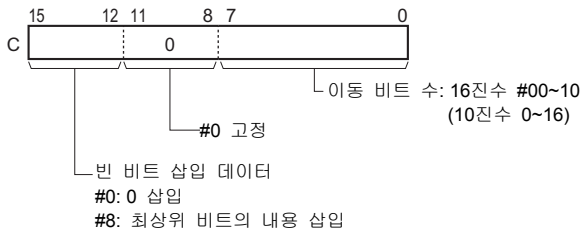
영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

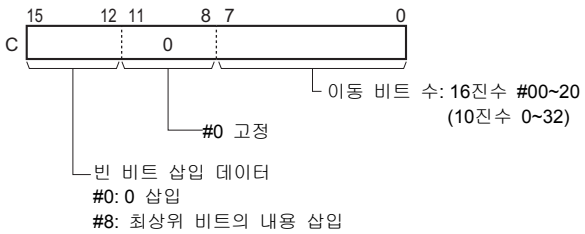
피연산자	내용	데이터 형식		크기	
		NASR	NSRL	NASR	NSRL
D	NASR: 이동 CH 번호 NSRL: 이동 하위 CH 번호	UINT	UDINT	1	2
C	제어 데이터	UINT	UINT	1	1

C: 제어 데이터

· NASR



· NSRL



■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
NASR	D	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
	C											○							
NSRL	D	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
	C											○	○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 제어 데이터(이동 비트 수)가 범위 밖일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 이동 결과가 0일 때 ON · 그 이외일 때 OFF

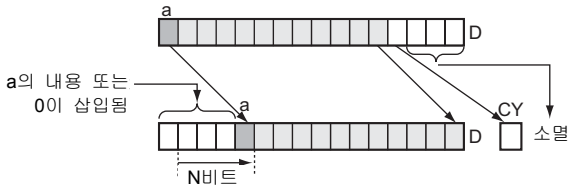
NASR/NSRL

이름	라벨	내용
캐리 플래그	P_CY	· 캐리(CY) 플래그에 1이 이동되었을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 이동 결과에 의해 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF

기능

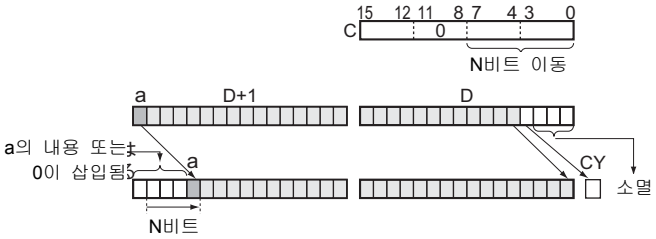
■ NASR

C로 지정한 비트 수(BIN)만큼 오른쪽 방향(최상위 비트→최하위 비트)으로 D를 이동합니다. 이때, 지정된 CH의 최상위 비트부터 이동 비트 수 분량의 비트에는 빈 비트 삽입 데이터(C)에 따른 데이터가 들어갑니다.



■ NSRL

D를 2워드 데이터로 하여, C로 지정한 비트 수(BIN)만큼 모두 오른쪽 방향(최상위 비트→최하위 비트)으로 이동합니다. 이때, 지정된 CH의 최상위 비트부터 이동 비트 수 분량의 비트에는 빈 비트 삽입 데이터(C)에 따른 데이터가 들어갑니다.

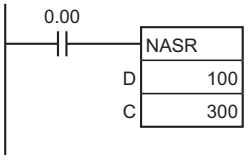


사용상의 주의 사항

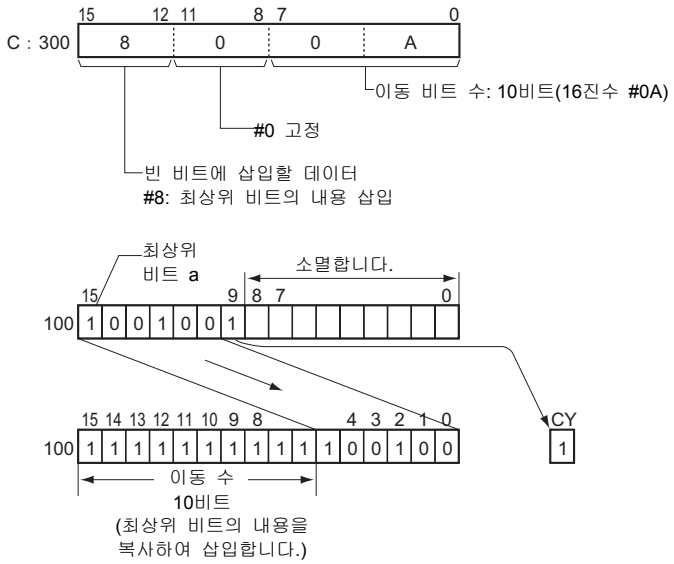
· 지정된 CH에서 오버 플로된 비트의 경우 마지막 비트의 내용이 캐리(CY) 플래그로 이동되고, 그 이외일 때는 소멸됩니다.

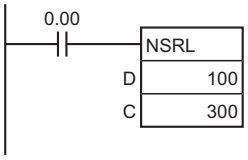
· 이동 비트 수(C)가 0인 경우에는 이동 동작이 수행되지 않습니다. 단, 지정된 채널의 데이터에 따라 각 플래그를 ON/OFF합니다.

샘플 프로그램

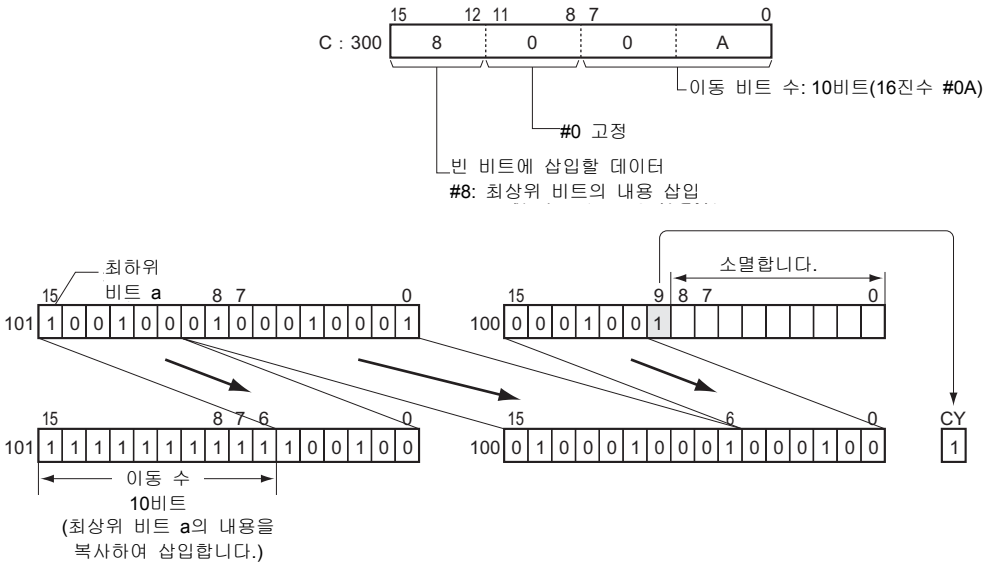


0.00이 ON이면 100CH의 데이터를 300CH(제어 데이터)의 비트 0~7에서 지정한 비트 수(예: 10비트)만큼 오른쪽 방향(최상위 비트→최하위 비트)으로 이동합니다.
빈 비트에는 100CH의 비트 15의 내용이 복사되어 삽입됩니다.
오버 플로된 비트는 마지막 비트가 CY에 설정되고 나머지는 소멸됩니다.





0.00이 ON이면 100CH~101CH의 2CH 데이터를 300CH(제어 데이터)의 비트 0~7에서 지정한 비트 수(예: 10비트)만큼 오른쪽 방향(최상위 비트→최하위 비트)으로 이동합니다.
빈 비트에는 101CH의 비트 15의 내용이 삽입됩니다.
오버 플로된 비트는 마지막 비트가 CY에 설정되고 나머지는 소멸됩니다.



증가/감소 명령

니모닉	명령 이름	펑션 번호	페이지
＋＋	BIN 증가	590	3-208
＋＋L	BIN 2워드 증가	591	
－－	BIN 감소	592	3-211
－－L	BIN 2워드 감소	593	
＋＋B	BCD 증가	594	3-214
＋＋BL	BCD 2워드 증가	595	
－－B	BCD 감소	596	3-217
－－BL	BCD 2워드 감소	597	

++/++L

명령 이름	니모닉	동작 옵션	펑션 번호	기능
BIN 증가	++	@++	590	16진수 4자리의 1CH 데이터에 1을 더합니다.
BIN 2워드 증가	++L	@++L	591	16진수 8자리의 2CH 데이터에 1을 더합니다.

심볼	++	++L
	++ / D (D: 데이터 CH 번호)	++L / D (D: 데이터 하위 CH 번호)

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		++	++L	++	++L
D	++: 데이터 CH 번호 ++L: 데이터 하위 CH 번호	UINT	UDINT	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
++	D	○	○	○	○	○	○	○	○	○	○	-	○	-	○	-	-	-	-
++L	D												-	○					

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 가산 결과, 자리올림이 있을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

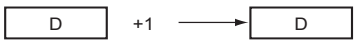
기능

■ ++

D로 지정한 데이터를 BIN 연산(+1)합니다.

++의 경우 입력 조건이 ON인 동안(OFF가 될 때까지)은 1사이클마다 1을 더합니다.

@++는 입력 조건이 기동될 때만(1사이클만) 1을 더합니다.

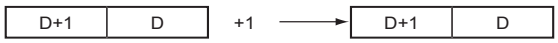


■ ++L

D로 지정한 데이터를 2워드 데이터로 BIN 연산(+1)합니다.

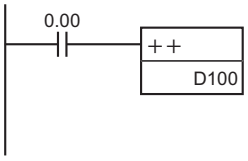
++L의 경우 입력 조건이 ON인 동안(OFF가 될 때까지)은 1사이클마다 1을 더합니다.

@++L은 입력 조건이 기동될 때만(1사이클만) 1을 더합니다.



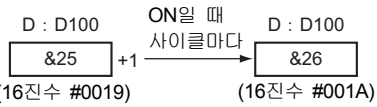
샘플 프로그램

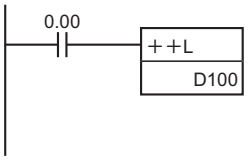
■ ++, ++L 명령일 때



0.00이 ON이면 사이클마다 D100의 데이터 내용에 1을 더합니다.
(0.00이 OFF가 될 때까지 1사이클마다 1을 더합니다.)

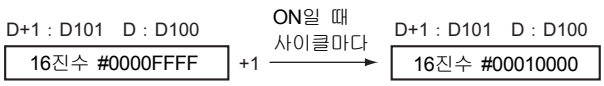
예

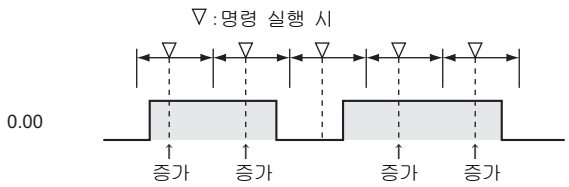




0.00이 ON이면 사이클마다 D101, D100의 데이터 내용에 1을 더합니다.
(0.00이 OFF가 될 때까지 1사이클마다 1을 더합니다.)

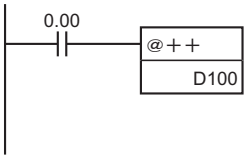
예





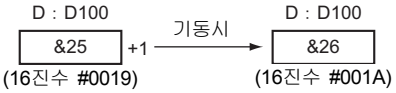
++/++L

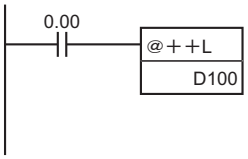
■ @++, @++L 명령일 때



0.00이 OFF→ON으로 기동되었을 때만 D100의 데이터 내용에 1을 더합니다.

예

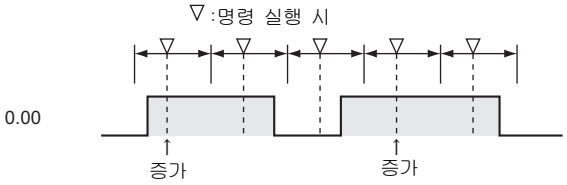




0.00이 OFF→ON으로 기동되었을 때만 D101, D100의 데이터 내용에 1을 더합니다.

예





−−/−−L

명령 이름	니모닉	동작 옵션	펑션 번호	기능
BIN 감소	−−	@−−	592	16진수 4자리의 1CH 데이터에서 1을 뺍니다.
BIN 2워드 감소	−−L	@−−L	593	16진수 8자리의 2CH 데이터에서 1을 뺍니다.

심볼	−−	−−L
	−− / D — D: 데이터 CH 번호	−−L / D — D: 데이터 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		−−	−−L	−−	−−L
D	−−: 데이터 CH 번호 −−L: 데이터 하위 CH 번호	UINT	UDINT	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
−−	D	○	○	○	○	○	○	○	○	○	○	-	○	-	○	-	-	-	-
−−L	D												-	○					

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 감산 결과, 자리내림이 있을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

−−/−−L

기능

■ −−

D로 지정한 데이터를 BIN 연산(-1)합니다.

−−의 경우 입력 조건이 ON인 동안(OFF가 될 때까지)은 1사이클마다 1을 뺍니다. @−−는 입력 조건이 기동될 때만(1사이클만) 1을 뺍니다.



■ −−L

D로 지정한 데이터를 2워드 데이터로 BIN 연산(-1)합니다.

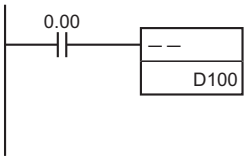
−−L의 경우 입력 조건이 ON인 동안(OFF가 될 때까지)은 1사이클마다 1을 뺍니다.

@−−L은 입력 조건이 기동될 때만(1사이클만) 1을 뺍니다.



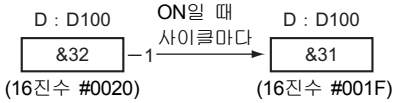
샘플 프로그램

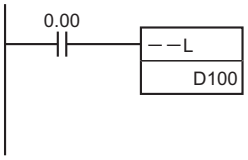
■ −−, −−L 명령일 때



0.00이 ON이면 사이클마다 D100의 데이터 내용에서 1을 뺍니다.
(0.00이 OFF가 될 때까지 1사이클마다 1을 뺍니다.)

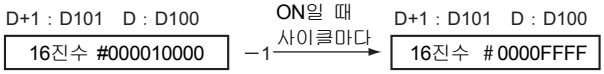
예

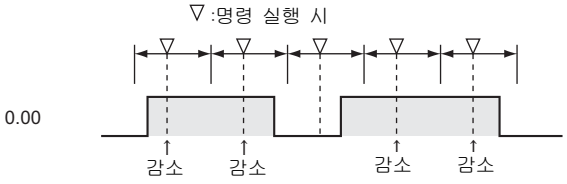




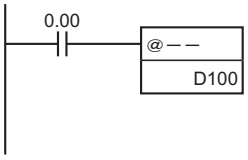
0.00이 ON이면 사이클마다 D101, D100의 데이터 내용에서 1을 뺍니다.
(0.00이 OFF가 될 때까지 1사이클마다 1을 뺍니다.)

예



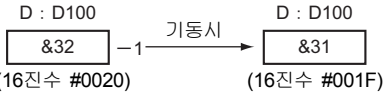


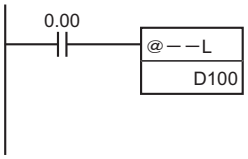
■ @−−, @−−L 명령일 때



0.00이 OFF→ON으로 기동되었을 때만 D100의 데이터 내용에서 1을 뺍니다.

예

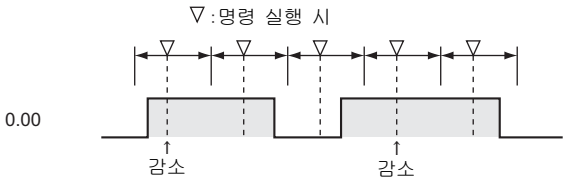




0.00이 OFF→ON으로 기동되었을 때만 D101, D100의 데이터 내용에서 1을 뺍니다.

예





++B/++BL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
BCD 증가	++B	@++B	594	BCD 4자리의 1CH 데이터에 1을 더합니다.
BCD 2워드 증가	++BL	@++BL	595	BCD 8자리의 2CH 데이터에 1을 더합니다.

심볼	++B	++BL
	++B D: 데이터 CH 번호	++BL D: 데이터 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		++B	++BL	++B	++BL
D	++B: 데이터 CH 번호 ++BL: 데이터 하위 CH 번호	WORD	DWORD	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
++B	D	○	○	○	○	○	○	○	○	○	○	-	○	-	○	-	-	-	-
++BL	D												-						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· D의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 가산 결과, 자리올림이 있을 때 ON · 그 이외일 때 OFF

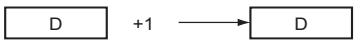
기능

■ ++B

D로 지정한 데이터를 BCD 연산(+1)합니다.

++B의 경우 입력 조건이 ON인 동안(OFF가 될 때까지)은 1사이클마다 1을 더합니다.

@++B는 입력 조건이 기동될 때만(1사이클만) 1을 더합니다.

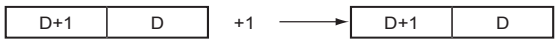


■ ++BL

D로 지정한 데이터를 2워드 데이터로 BCD 연산(+1)합니다.

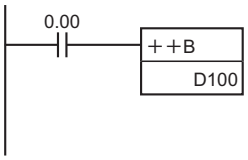
++BL의 경우 입력 조건이 ON인 동안(OFF가 될 때까지)은 1사이클마다 1을 더합니다.

@++BL은 입력 조건이 기동될 때만(1사이클만) 1을 더합니다.



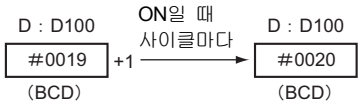
샘플 프로그램

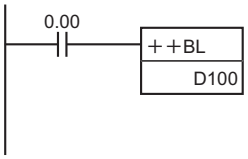
■ ++B, ++BL 명령일 때



0.00이 ON이면 사이클마다 D100의 데이터 내용에 1을 더합니다.
(0.00이 OFF가 될 때까지 1사이클마다 1을 더합니다.)

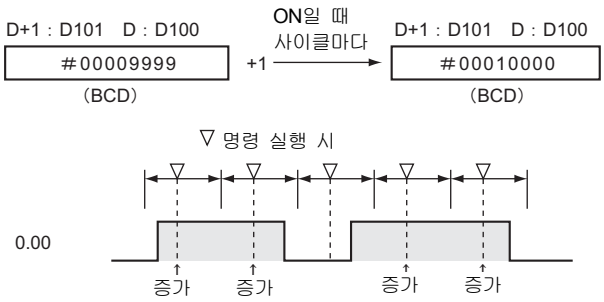
예





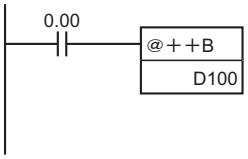
0.00이 ON이면 사이클마다 D101, D100의 데이터 내용에 1을 더합니다.
(0.00이 OFF가 될 때까지 1사이클마다 1을 더합니다.)

예



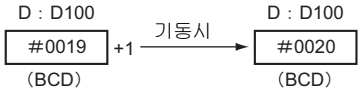
++B/++BL

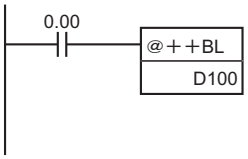
■ @++B, @++BL 명령일 때



0.00이 OFF→ON으로 기동되었을 때만 D100의 데이터 내용에 1을 더합니다.

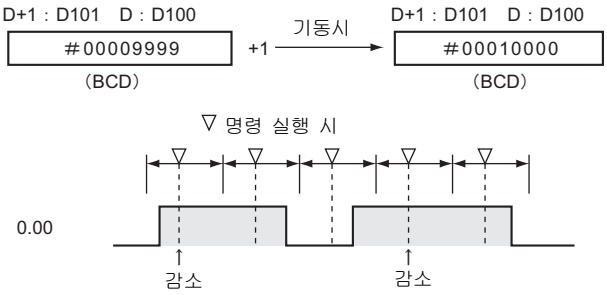
예





0.00이 OFF→ON으로 기동되었을 때만 D101, D100의 데이터 내용에 1을 더합니다.

예



－－B/－－BL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
BCD 감소	－－B	@－－B	596	BCD 4자리의 1CH 데이터에서 1을 뺍니다.
BCD 2워드 감소	－－BL	@－－BL	597	BCD 8자리의 2CH 데이터에서 1을 뺍니다.

심볼	－－B	－－BL
	－－B D D: 데이터 CH 번호	－－BL D D: 데이터 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		－－B	－－BL	－－B	－－BL
D	－－B: 데이터 CH 번호 －－BL: 데이터 하위 CH 번호	WORD	DWORD	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
－－B	D	○	○	○	○	○	○	○	○	○	○	-	○	-	○	-	-	-	-
－－BL	D												-						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· D의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 감산 결과, 자리내림이 있을 때 ON · 그 이외일 때 OFF

−−B/−−BL

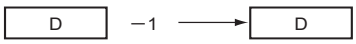
기능

■ −−B

D로 지정한 데이터를 BCD 연산(-1)합니다.

−−B의 경우 입력 조건이 ON인 동안(OFF가 될 때까지)은 1사이클마다 1을 뺍니다.

@−−B는 입력 조건이 기동될 때만(1사이클만) 1을 뺍니다.

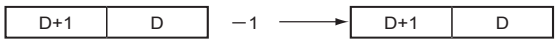


■ −−BL

D로 지정한 데이터를 2워드 데이터로 BCD 연산(-1)합니다.

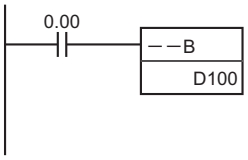
−−BL의 경우 입력 조건이 ON인 동안(OFF가 될 때까지)은 1사이클마다 1을 뺍니다.

@−−BL은 입력 조건이 기동될 때만(1사이클만) 1을 뺍니다.



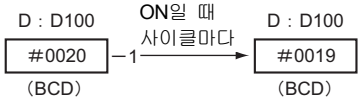
샘플 프로그램

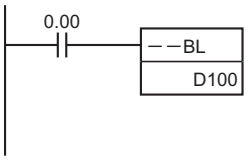
■ −−B, −−BL 명령일 때



0.00이 ON이면 사이클마다 D100의 데이터 내용에서 1을 뺍니다.
(0.00이 OFF가 될 때까지 1사이클마다 1을 뺍니다.)

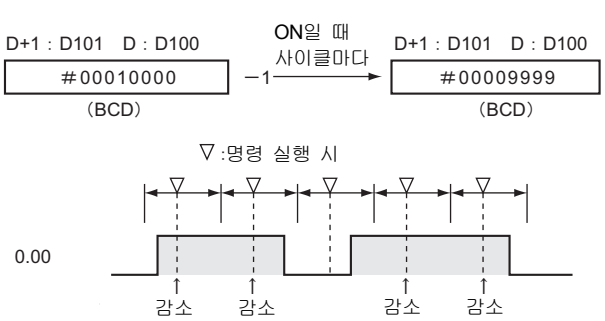
예



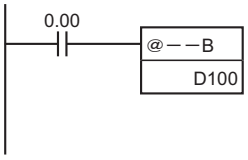


0.00이 ON이면 사이클마다 D101, D100의 데이터 내용에서 1을 뺍니다.
(0.00이 OFF가 될 때까지 1사이클마다 1을 뺍니다.)

예

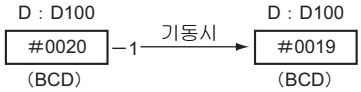


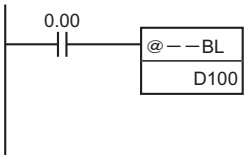
■ @－－B, @－－BL 명령일 때



0.00이 OFF→ON으로 기동되었을 때만 D100의 데이터 내용에서 1을 뺍니다.

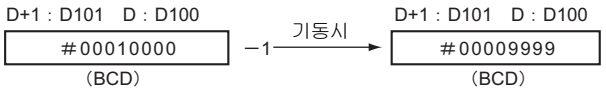
예

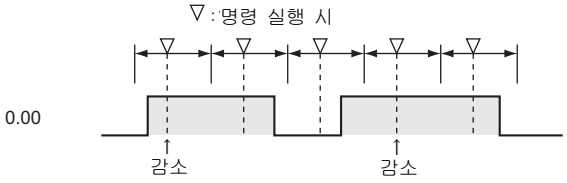




0.00이 OFF→ON으로 기동되었을 때만 D101, D100의 데이터 내용에서 1을 뺍니다.

예



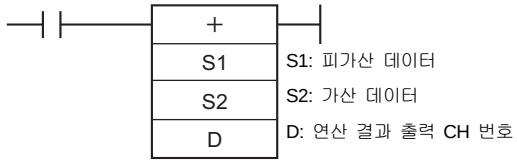


사칙연산 명령

니모닉	명령 이름	평션 번호	페이지
+	부호 있음, CY 없는 BIN 가산	400	3-222
+L	부호 있음, CY 없는 BIN 2워드 가산	401	
+C	부호 및 CY 있는 BIN 가산	402	3-224
+CL	부호 및 CY 있는 BIN 2워드 가산	403	
+B	CY 없는 BCD 가산	404	3-227
+BL	CY 없는 BCD 2워드 가산	405	
+BC	CY 있는 BCD 가산	406	3-229
+BCL	CY 있는 BCD 2워드 가산	407	
−	부호 있음, CY 없는 BIN 감산	410	3-231
−L	부호 있음, CY 없는 BIN 2워드 감산	411	
−C	부호 및 CY 있는 BIN 감산	412	3-235
−CL	부호 및 CY 있는 BIN 2워드 감산	413	
−B	CY 없는 BCD 감산	414	3-238
−BL	CY 없는 BCD 2워드 감산	415	
−BC	CY 있는 BCD 감산	416	3-241
−BCL	CY 있는 BCD 2워드 감산	417	
*	부호 있는 BIN 승산	420	3-243
*L	부호 있는 BIN 2워드 승산	421	
*U	부호 없는 BIN 승산	422	3-245
*UL	부호 없는 BIN 2워드 승산	423	
*B	BCD 승산	424	3-247
*BL	BCD 2워드 승산	425	
/	부호 있는 BIN 제산	430	3-249
/L	부호 있는 BIN 2워드 제산	431	
/U	부호 없는 BIN 제산	432	3-251
/UL	부호 없는 BIN 2워드 제산	433	
/B	BCD 제산	434	3-253
/BL	BCD 2워드 제산	435	

+/+L

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부호 있음, CY 없는 BIN 가산	+	@+	400	채널 데이터나 상수를 부호 있는 16진수 4자리로 가산합니다.
부호 있음, CY 없는 BIN 2워드 가산	+L	@+L	401	2CH 데이터나 상수를 부호 있는 16진수 8자리로 가산합니다.

심볼	+	+L
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		+	+L	+	+L
S1	+: 피가산 데이터 +L: 피가산 데이터 하위 CH 번호	INT	DINT	1	2
S2	+: 가산 데이터 +L: 가산 데이터 하위 CH 번호	INT	DINT	1	2
D	+: 연산 결과 출력 CH 번호 +L: 연산 결과 출력 하위 CH 번호	INT	DINT	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
+	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	-	○	-	-	-	-
	D											-							
+L	S1,S2	○	○	○	○	○	○	○	○	○	○	○	-	○	○	-	-	-	-
	D											-							

조건 플래그의 동작

이름	라벨	내용	
		+	+L
에러 플래그	P_ER	OFF	OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 가산 결과, 자리올림이 있을 때 ON · 그 이외일 때 OFF	· 가산 결과, 자리올림이 있을 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 양수+양수의 결과가 음수 범위 (16진수 #8000~FFFF)에 있을 때 ON · 그 이외일 때 OFF	· 양수+양수의 결과가 음수 범위 (16진수 #80000000~FFFFFFFF)에 있을 때 ON · 그 이외일 때 OFF

이름	라벨	내용	
		+	+L
언더 플로 플래그	P_UF	· 음수+음수의 결과가 양수 범위 (16진수 #0000~7FFF)에 있을 때 ON · 그 이외일 때 OFF	· 음수+음수의 결과가 양수 범위 (16진수 #00000000~7FFFFFFF)에 있을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 결과의 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF	· 결과의 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF

기능

■ +

S1로 지정한 데이터와 S2로 지정한 데이터를 BIN 가산하여 결과를 D에 출력합니다.

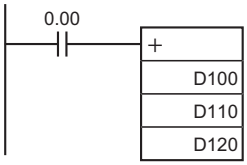


■ +L

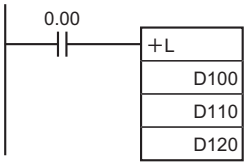
S1로 지정한 데이터와 S2로 지정한 데이터를 2워드 데이터로 BIN 가산하여 결과를 D에 출력합니다.



샘플 프로그램



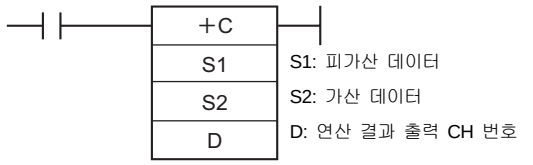
0.00이 ON이면 D100과 D110이 부호 있는 BIN 4자리로 가산되어 결과가 D120에 출력됩니다.



0.00이 ON이면 D101, D100의 2채널의 데이터와 D111, D110의 2채널의 데이터가 부호 있는 BIN 8자리로 가산되어 결과가 D121, D120에 출력됩니다.

+C/+CL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부호 및 CY 있는 BIN 가산	+C	@+C	402	데이터나 상수를 캐리(CY) 플래그를 포함하여 부호 있는 16진수 4자리로 가산합니다.
부호 및 CY 있는 BIN 2워드 가산	+CL	@+CL	403	2CH분의 데이터나 상수를 캐리(CY) 플래그를 포함하여 부호 있는 16진수 8자리로 가산합니다.

심볼	+C	+CL
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		+C	+CL	+C	+CL
S1	+C: 피가산 데이터 +CL: 피가산 데이터 하위 CH 번호	INT	DINT	1	2
S2	+C: 가산 데이터 +CL: 가산 데이터 하위 CH 번호	INT	DINT	1	2
D	+C: 연산 결과 출력 CH 번호 +CL: 연산 결과 출력 하위 CH 번호	INT	DINT	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
+C	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	-	○	-	-	-	-
	D											-							
+CL	S1,S2	○	○	○	○	○	○	○	○	○	○	○	-	-	○	-	-	-	-
	D											-							

조건 플래그의 동작

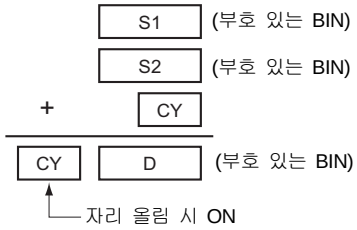
이름	라벨	내용	
		+C	+CL
에러 플래그	P_ER	OFF	OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 가산 결과, 자리올림이 있을 때 ON · 그 이외일 때 OFF	· 가산 결과, 자리올림이 있을 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 양수+양수+캐리 플래그의 결과가 음수 범위(16진수 #8000~FFFF)에 있을 때 ON · 그 이외일 때 OFF	· 양수+양수+캐리 플래그의 결과가 음수 범위(16진수 #80000000~FFFFFFFF)에 있을 때 ON · 그 이외일 때 OFF

이름	라벨	내용	
		+C	+CL
언더 플로 플래그	P_UF	· 음수+음수+캐리 플래그의 결과가 양수 범위(16진수 #0000~7FFF)에 있을 때 ON · 그 이외일 때 OFF	· 음수+음수+캐리 플래그의 결과가 양수 범위(16진수 #00000000~7FFFFFFF)에 있을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 결과의 최상위 비트가 이 되었을 때 ON · 그 이외일 때 OFF	· 결과의 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF

기능

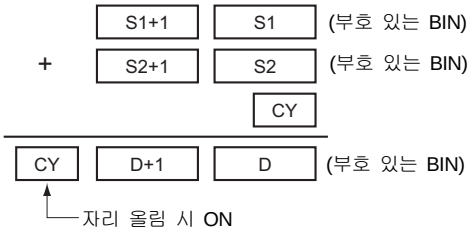
■ +C

S1로 지정한 데이터와 S2로 지정한 데이터를 캐리(CY) 플래그도 포함하여 BIN 가산한 결과를 D에 출력합니다.



■ +CL

S1로 지정한 데이터와 S2로 지정한 데이터를 2워드 데이터로 캐리(CY) 플래그도 포함하여 BIN 가산한 결과를 D+1, D에 출력합니다.

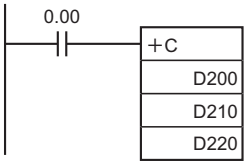


힌트

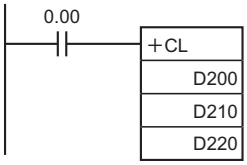
· 캐리(CY) 플래그를 클리어하고 싶은 경우는 CLC(클리어 캐리) 명령을 실행하십시오.

+C/+CL

샘플 프로그램



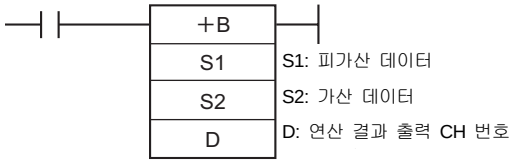
0.00이 ON이면 D200, D210 및 CY 플래그가 부호 있는 BIN 4자리로 가산되어 결과가 D220에 출력됩니다.



0.00이 ON이면 D201, D200과 D211, D210 및 CY 플래그가 부호 있는 BIN 8자리로 가산되어 결과가 D221, D220에 출력됩니다.

+B/+BL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
CY 없는 BCD 가산	+B	@+B	404	채널 데이터나 상수를 BCD 4자리로 가산합니다.
CY 없는 BCD 2워드 가산	+BL	@+BL	405	2CH분의 채널 데이터나 상수를 BCD 8자리로 가산합니다.

심볼	+B	+BL
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		+B	+BL	+B	+BL
S1	+B: 피가산 데이터 +BL: 피가산 데이터 하위 CH 번호	WORD	DWORD	1	2
S2	+B: 가산 데이터 +BL: 가산 데이터 하위 CH 번호	WORD	DWORD	1	2
D	+B: 연산 결과 출력CH 번호 +BL: 연산 결과 출력 하위 CH 번호	WORD	DWORD	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
+B	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	-	○	-	-	-	-
	D											-							
+BL	S1,S2	○	○	○	○	○	○	○	○	○	○	○	-	-	○	-	-	-	-
	D											-							

조건 플래그의 동작

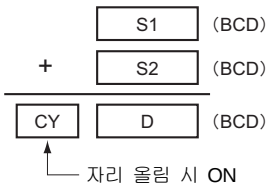
이름	라벨	내용	
		+B	+BL
에러 플래그	P_ER	· S1의 데이터가 BCD가 아닐 때 ON · S2의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF	· S1+1, S1의 데이터가 BCD가 아닐 때 ON · S2+1, S2의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 가산 결과, 자리올림이 있을 때 ON · 그 이외일 때 OFF	· 가산 결과, 자리올림이 있을 때 ON · 그 이외일 때 OFF

+B/+BL

기능

■ +B

S1로 지정한 데이터와 S2로 지정한 데이터를 BCD 가산하여 결과를 D에 출력합니다.

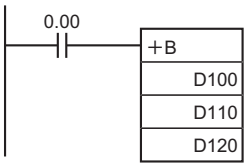


■ +BL

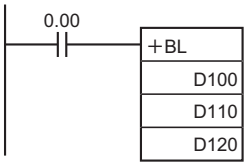
S1로 지정한 데이터와 S2로 지정한 데이터를 2워드 데이터로 BCD 가산하여 결과를 D+1, D에 출력합니다.



샘플 프로그램



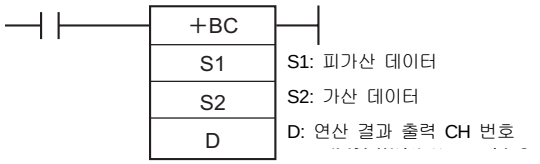
0.00이 ON이면 D100과 D110이 BCD 4자리로 가산되어 결과가 D120에 출력됩니다.



0.00이 ON이면 D101, D100과 D111, D110이 BCD 8자리로 가산되어 결과가 D121, D120에 출력됩니다.

+BC/+BCL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
CY 있는 BCD 가산	+BC	@+BC	406	채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 부호 있는 BCD 4자리로 가산합니다.
CY 있는 BCD 2워드 가산	+BCL	@+BCL	407	2CH분의 채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 BCD 8자리로 가산합니다.

심볼	+BC	+BCL
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		+BC	+BCL	+BC	+BCL
S1	피가산 데이터	WORD	DWORD	1	2
S2	가산 데이터	WORD	DWORD	1	2
D	+BC: 연산 결과 출력 CH 번호 +BCL: 연산 결과 출력 하위 CH 번호	WORD	DWORD	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
+BC	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	-	○	-	-	-	-
	D											-							
+BCL	S1,S2	○	○	○	○	○	○	○	○	○	○	○	-	-	○	-	-	-	-
	D											-							

조건 플래그의 동작

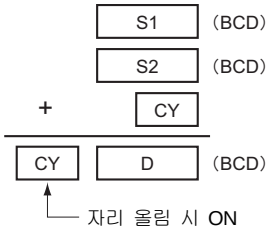
이름	라벨	내용	
		+BC	+BCL
에러 플래그	P_ER	· S1의 데이터가 BCD가 아닐 때 ON · S2의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF	· S1+1, S1의 데이터가 BCD가 아닐 때 ON · S2+1, S2의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 가산 결과, 자리올림이 있을 때 ON · 그 이외일 때 OFF	· 가산 결과, 자리올림이 있을 때 ON · 그 이외일 때 OFF

+BC/+BCL

기능

■ +BC

S1로 지정한 데이터와 S2로 지정한 데이터를 캐리(CY) 플래그도 포함하여 BCD 가산한 결과를 D에 출력합니다.



■ +BCL

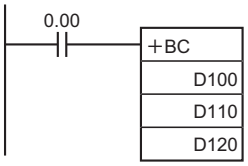
S1로 지정한 데이터와 S2로 지정한 데이터를 2워드 데이터로 캐리(CY) 플래그도 포함하여 BCD 가산한 결과를 D+1, D에 출력합니다.



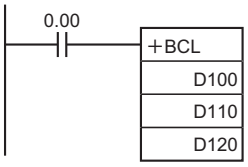
힌트

· 캐리(CY) 플래그를 클리어하고 싶은 경우는 CLC(클리어 캐리) 명령을 실행하십시오.

샘플 프로그램



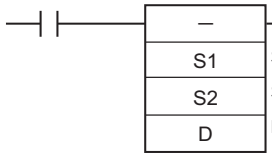
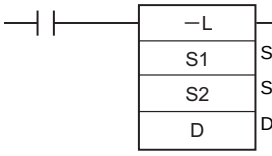
0.00이 ON이면 D100, D110 및 CY 플래그가 BCD 4자리로 가산되어 결과가 D120에 출력됩니다.



0.00이 ON이면 D101, D100과 D111, D110 및 CY 플래그가 BCD 8자리로 가산되어 결과가 D121, D120에 출력됩니다.

−/−L

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부호 있음, CY 없는 BIN 감산	−	@−	410	채널 데이터나 상수를 부호 있는 16진수 4자리로 감산합니다.
부호 있음, CY 없는 BIN 2워드 감산	−L	@−L	411	2CH분의 채널 데이터나 상수를 부호 있는 16진수 8자리로 감산합니다.

심볼	−	−L
	 S1: 피감산 데이터 S2: 감산 데이터 D: 연산 결과 출력 CH 번호	 S1: 피감산 데이터 하위 CH 번호 . S2: 감산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		−	−L	−	−L
S1	−: 피감산 데이터 −L: 피감산 데이터 하위 CH 번호	INT	DINT	1	2
S2	−: 감산 데이터 −L: 감산 데이터 하위 CH 번호	INT	DINT	1	2
D	−: 연산 결과 출력 CH 번호 −L: 연산 결과 출력 하위 CH 번호	INT	DINT	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
−	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	-	○	-	-	-	-
	D											-							
−L	S1,S2	○	○	○	○	○	○	○	○	○	○	○	-	○	○	-	-	-	-
	D											-							

조건 플래그의 동작

이름	라벨	내용	
		−	−L
에러 플래그	P_ER	OFF	OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 감산 결과, 자리내림이 있을 때 ON · 그 이외일 때 OFF	· 감산 결과, 자리내림이 있을 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 양수-음수의 결과가 음수 범위(16진수 #8000~FFFF)에 있을 때 ON · 그 이외일 때 OFF	· 양수-음수의 결과가 음수 범위(16진수 #80000000~FFFFFFFF)에 있을 때 ON · 그 이외일 때 OFF

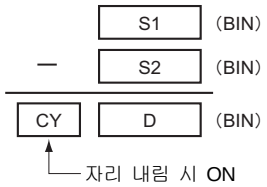
−/−L

이름	라벨	내용	
		−	−L
언더 플로 플래그	P_UF	· 음수-양수의 결과가 양수 범위(16진수 #0000~7FFF)에 있을 때 ON · 그 이외일 때 OFF	· 음수-양수의 결과가 양수 범위(16진수 #00000000~7FFFFFFF)에 있을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 결과의 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF	· 결과의 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF

기능

■ −

S1로 지정한 데이터와 S2로 지정한 데이터를 BIN 감산하여 결과를 D에 출력합니다. 결과가 음수가 되는 경우에는 2의 보수로 D에 출력됩니다.



■ −L

S1로 지정한 데이터와 S2로 지정한 데이터를 2워드 데이터로 BIN 감산하여 결과를 D+1, D에 출력합니다. 결과가 음수가 되는 경우에는 2의 보수로 D+1, D에 출력됩니다.



힌트

· 2의 보수란?

2진수의 각 자리를 1에서 빼서 그 결과에 1을 더한 값입니다.

예) 2진수 1101에 대한 2의 보수는 1111(16진수 #F)-1101(16진수 #D)+1(16진수 #1) = 0011(16진수 #3)입니다.

예) 16진수 4자리의 16진수 #3039에 대한 2의 보수는 16진수 #FFFF-16진수 #3039+16진수 #0001 = 16진수 #CFC7입니다.

따라서 16진수 4자리의 경우, 16진수 a에 대한 2의 보수는 16진수 #FFFF-16진수 a+16진수 #0001 = 16진수 b가 됩니다. 2의 보수인 16진수 b에서 원래의 수 16진수 a를 구할 때는 16진수 a = 16진수 #10000-16진수 b입니다.

예) 2의 보수인 16진수 #CFC7에서 원래의 수를 구하려면 16진수 #10000-16진수 #CFC7 = 16진수 #3039가 됩니다.

숫자 예 1)

부호 있는 데이터로 본 경우		부호 없는 데이터로 본 경우
16진수 #FFFF →	−1	65535
−)16진수 #0001 →	−) +1	−) 00001
16진수 #FFFE →	−2*1	65534*2
부정 플래그 ON		
캐리 OFF		

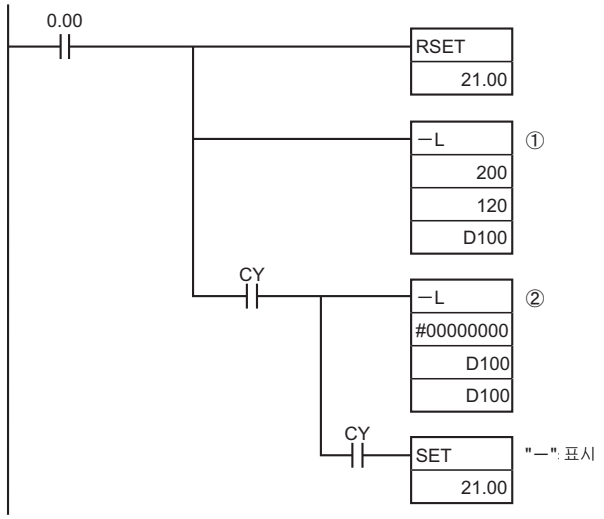
* 1: 부정 플래그가 ON이므로 결과(16진수 #FFFE)는 음수(2의 보수)이며 −2를 나타냅니다.
* 2: 캐리가 OFF이고 결과(16진수 #FFFE)는 부호 없는 데이터의 양수로 65534가 됩니다.

숫자 예 2)

부호 있는 데이터로 본 경우		부호 없는 데이터로 본 경우
16진수 #FFFD →	−3	65533
−)16진수 #FFFF →	−) −1	−) 65535
16진수 #FFFE →	−2*3	65534*4
부정 플래그 ON		
캐리 ON		

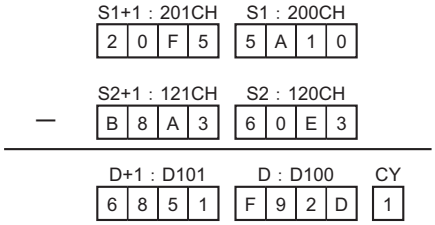
* 3: 부정 플래그가 ON이므로 결과(16진수 #FFFE)는 음수(2의 보수)이며 −2를 나타냅니다.
* 4: 캐리가 ON이므로 결과(16진수 #FFFE)는 음수(2의 보수)가 되며 원래의 수로 변환하면 −2가 됩니다.

(예) 16진수: 20F55A10 − B8A360E3 = -97AE06D3

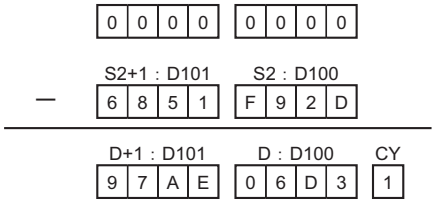


201, 200CH에 의해 121, 120CH의 BCD 8자리 감산을 수행합니다. 결과는 D101, D100의 8자리에 출력됩니다. 결과가 음수이면 #00000000-보수 = 원래의 수인 프로그램 ②가 실행되고, 다시 D101, D100에 원래의 수가 출력됩니다.

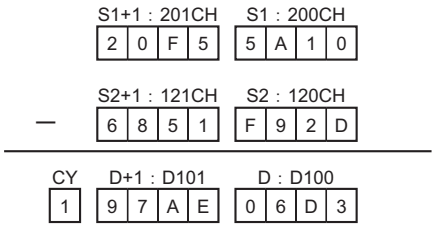
①의 감산



②의 감산…CY플래그가 1이므로 보수를 원래의 수로 수정합니다.



최종 감산 결과

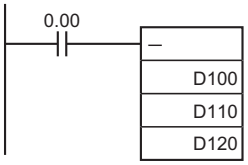


CY 플래그가 ON이므로 실제 숫자는 -97AE06D3을 나타냅니다.

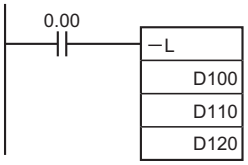
D101, D100의 내용이 음수라는 것을 표시하는 경우는 21.00에 출력합니다.

−/−L

샘플 프로그램



0.00이 ON이면 D100에서 D110이 16진수 4자리로 감산되어 결과가 D120에 출력됩니다.



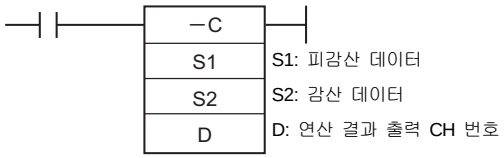
0.00이 ON이면 D101, D100에서 D111, D110이 16진수 8자리로 감산되어 결과가 D121, D120에 출력됩니다.

감산 결과가 음수(S1＜S2 또는 S1+1, S1＜S2+1, S2)이면, 감산 결과는 2의 보수로 출력됩니다. 이때, CY 플래그는 ON이 됩니다.

2의 보수를 원래의 수로 변환하는 경우는 CY 플래그를 입력 조건으로 하고, 다시 0에서 감산 결과를 빼는 프로그램이 필요합니다. 이때 음수라는 것을 나타내기 위한 표시 등에 CY 플래그를 사용합니다.

－C/－CL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부호 및 CY 있는 BIN 감산	－C	@－C	412	채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 부호 있는 16진수 4자리로 감산합니다.
부호 및 CY 있는 BIN 2워드 감산	－CL	@－CL	413	2CH분의 채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 부호 있는 16진수 8자리로 감산합니다.

심볼	－C	－CL
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		－C	－CL	－C	－CL
S1	－C: 피감산 데이터 －CL: 피감산 데이터 하위 CH 번호	INT	DINT	1	2
S2	－C: 감산 데이터 －CL: 감산 데이터 하위 CH 번호	INT	DINT	1	2
D	－C: 연산 결과 출력 CH 번호 －CL: 연산 결과 출력 하위 CH 번호	INT	DINT	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
－C	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	-	○	-	-	-	-
	D											-							
－CL	S1,S2	○	○	○	○	○	○	○	○	○	○	○	-	-	○	-	-	-	-
	D											-							

조건 플래그의 동작

이름	라벨	내용	
		－C	－CL
에러 플래그	P_ER	OFF	OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 감산 결과, 자리내림이 있을 때 ON · 그 이외일 때 OFF	· 감산 결과, 자리내림이 있을 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 양수-음수-캐리 플래그의 결과가 음수 범위(16진수 #8000~FFFF)에 있을 때 ON · 그 이외일 때 OFF	· 양수-음수-캐리 플래그의 결과가 음수 범위(16진수 #80000000~FFFFFFFF)에 있을 때 ON · 그 이외일 때 OFF

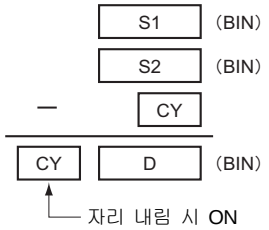
−C/−CL

이름	라벨	내용	
		−C	−CL
언더 플로 플래그	P_UF	· 음수-양수-캐리 플래그의 결과가 양수 범위(16진수 #0000~7FFF)에 있을 때 ON · 그 이외일 때 OFF	· 음수-양수-캐리 플래그의 결과가 양수 범위(16진수 #00000000~7FFFFFFF)에 있을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 결과의 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF	· 결과의 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF

기능

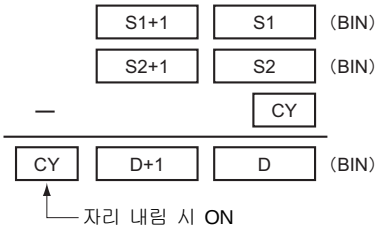
■ −C

S1로 지정한 데이터와 S2로 지정한 데이터를 캐리(CY) 플래그도 포함하여 BIN 감산한 결과를 D에 출력합니다. 결과가 음수가 되는 경우에는 2의 보수로 D에 출력됩니다.



■ −CL

S1로 지정한 데이터와 S2로 지정한 데이터를 2워드 데이터로 캐리(CY) 플래그도 포함하여 BIN 감산한 결과를 D+1, D에 출력합니다. 결과가 음수가 되는 경우에는 2의 보수로 D+1, D에 출력됩니다.



힌트

· 캐리(CY) 플래그를 클리어하고 싶은 경우는 CLC(클리어 캐리) 명령을 실행하십시오.

· 2의 보수란?

2진수의 각 자리를 1에서 빼서 그 결과에 1을 더한 값입니다.

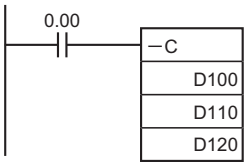
예) 2진수 1101에 대한 2의 보수는 1111(16진수 #F)-1101(16진수 #D)+1(16진수 #1) = 0011(16진수 #3)입니다.

예) 16진수 4자리의 16진수 #3039에 대한 2의 보수는 16진수 #FFFF-16진수 #3039+16진수 #0001 = 16진수 #CFC7입니다.

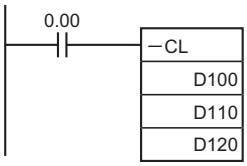
따라서 16진수 4자리의 경우, 16진수 a에 대한 2의 보수는 16진수 #FFFF-16진수 a+16진수 #0001 = 16진수 b가 됩니다. 2의 보수 16진수 b에서 원래의 수 16진수 a를 구할 때는 16진수 a = 16진수 #10000-16진수 b입니다.

예) 2의 보수인 16진수 #CFC7에서 원래의 수를 구하려면 16진수 #10000-16진수 #CFC7 = 16진수 #3039가 됩니다.

샘플 프로그램



0.00이 ON이면 D100에서 D110 및 CY 플래그가 16진수 4자리로 감산되어 결과가 D120에 출력됩니다.



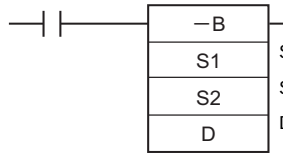
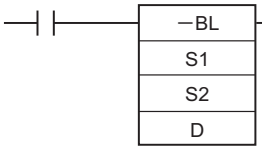
0.00이 ON이면 D101, D100에서 D111, D110 및 CY 플래그가 16진수 8자리로 감산되어 결과가 D121, D120에 출력됩니다.

감산 결과가 음수(S1＜S2 또는 S1+1, S1＜S2+1, S2)이면, 감산 결과는 2의 보수로 출력됩니다. 이때, CY 플래그는 ON이 됩니다.

2의 보수를 원래의 수로 변환하는 경우는 CY 플래그를 입력 조건으로 하고, 다시 0에서 감산 결과를 빼는 프로그램이 필요합니다. 이때 음수라는 것을 나타내기 위한 표시 등에 CY 플래그를 사용합니다.

－B/－BL

명령 이름	니모닉	동작 옵션	평션 번호	기능
CY 없는 BCD 감산	－B	@－B	414	채널 데이터나 상수를 BCD 4자리로 감산합니다.
CY 없는 BCD 2워드 감산	－BL	@－BL	415	2CH분의 채널 데이터나 상수를 BCD 8자리로 감산합니다.

심볼	－B	－BL
	 S1: 피감산 데이터 S2: 감산 데이터 D: 연산 결과 출력 CH 번호	 S1: 피감산 데이터 하위 CH 번호 S2: 감산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		－B	－BL	－B	－BL
S1	－B: 피감산 데이터 －BL: 피감산 데이터 하위 CH 번호	WORD	DWORD	1	2
S2	－B: 감산 데이터 －BL: 감산 데이터 CH 번호	WORD	DWORD	1	2
D	－B: 연산 결과 출력 CH 번호 －BL: 연산 결과 출력 하위 CH 번호	WORD	DWORD	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
－B	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	-	○	-	-	-	-
	D											-							
－BL	S1,S2	○	○	○	○	○	○	○	○	○	○	○	-	-	○	-	-	-	-
	D											-							

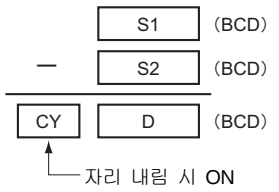
조건 플래그의 동작

이름	라벨	내용	
		－B	－BL
에러 플래그	P_ER	· S1의 데이터가 BCD가 아닐 때 ON · S2의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF	· S1+1, S1의 데이터가 BCD가 아닐 때 ON · S2+1, S2의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 감산 결과, 자리내림이 있을 때 ON · 그 이외일 때 OFF	· 감산 결과, 자리내림이 있을 때 ON · 그 이외일 때 OFF

기능

■ －B

S1로 지정한 데이터와 S2로 지정한 데이터를 BCD 감산하여 결과를 D에 출력합니다. 결과가 음수가 되는 경우에는 10의 보수로 D에 출력됩니다.



■ －BL

S1로 지정한 데이터와 S2로 지정한 데이터를 2워드 데이터로 BCD 감산하여 결과를 D+1, D에 출력합니다. 결과가 음수가 되는 경우에는 10의 보수로 D+1, D에 출력됩니다.



힌트

· 10의 보수란?

각 자리를 9에서 빼서 그 결과에 1을 더한 값입니다.

예) 7556에 대한 10의 보수는 9999-7556+1 = 2444입니다.

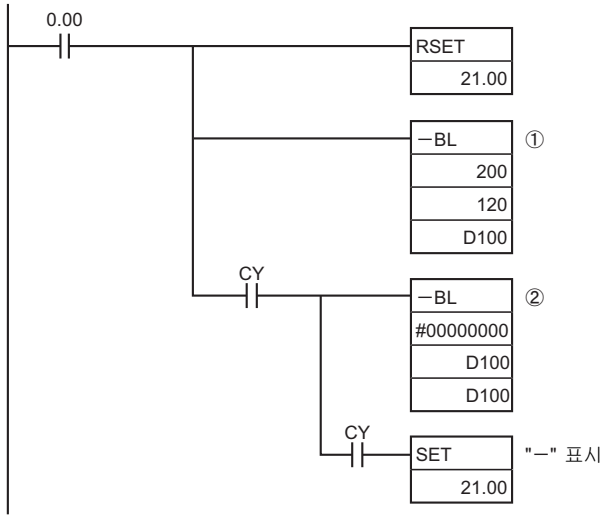
따라서 4자리 숫자 A에 대한 10의 보수는 9999-A+1 = B가 됩니다.

10의 보수 B에서 원래의 수 A를 구할 때는 A = 10000-B입니다.

예) 10의 보수 2444에서 원래의 수를 구하는 경우 10000-2444 = 7556입니다.

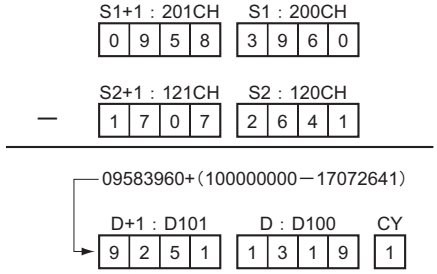
−B/−BL

(예) BCD: 9,583,960-17,072,641 = -7,488,681

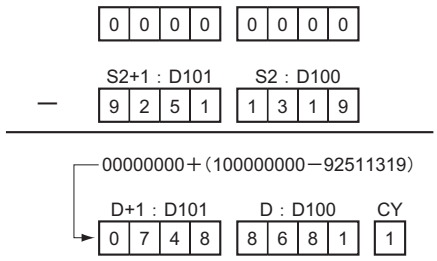


201, 200CH에 의해 121, 120CH의 BCD 8자리 감산을 수행합니다. 결과는 D101, D100의 8자리에 출력됩니다. 결과가 음수이면 #00000000-보수 = 원래의 수인 프로그램 ②가 실행되고, 다시 D101, D100에 원래의 수가 출력됩니다.

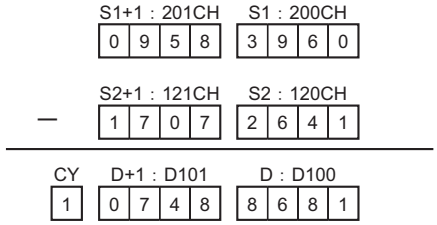
①의 감산



②의 감산…CY플래그가 1이므로 보수를 원래의 수로 수정합니다.



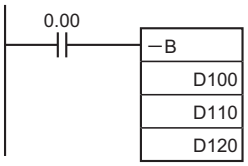
최종 감산 결과



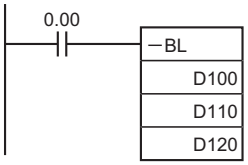
CY 플래그가 ON이므로 실제 숫자는 -7,488,681을 나타냅니다.

D101, D100의 내용이 음수라는 것을 표시하는 경우는 21.00에 출력합니다.

샘플 프로그램



0.00이 ON이면 D100에서 D110이 BCD 4자리로 감산되어 결과가 D120에 출력됩니다.



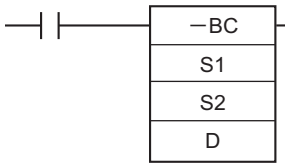
0.00이 ON이면 D101, D100에서 D111, D110이 BCD 8자리로 감산되어 결과가 D121, D120에 출력됩니다.

감산 결과가 음수(S1 < S2 또는 S1+1, S1 < S2+1, S2)이면, 감산 결과는 10의 보수로 출력됩니다. 이때, CY 플래그는 ON이 됩니다.

10의 보수를 원래의 수로 변환하는 경우는 CY 플래그를 입력 조건으로 하고, 다시 0에서 감산 결과를 빼는 프로그램이 필요합니다. 이때 음수라는 것을 나타내기 위한 표시 등에 CY 플래그를 사용합니다.

－BC/－BCL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
CY 있는 BCD 감산	－BC	@－BC	416	채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 BCD 4자리로 감산합니다.
CY 있는 BCD 2워드 감산	－BCL	@－BCL	417	2CH분의 채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 BCD 8자리로 감산합니다.

심볼	－BC	－BCL
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		－BC	－BCL	－BC	－BCL
S1	－BC: 피감산 데이터 －BCL: 피감산 데이터 하위 CH 번호	WORD	DWORD	1	2
S2	－BC: 감산 데이터 －BCL: 감산 데이터 하위 CH 번호	WORD	DWORD	1	2
D	－BC: 연산 결과 출력 CH 번호 －BCL: 연산 결과 출력 하위 CH 번호	WORD	DWORD	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
－BC	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	-	○	-	-	-	-
	D											-							
－BCL	S1,S2	○	○	○	○	○	○	○	○	○	○	○	-	-	○	-	-	-	-
	D											-							

조건 플래그의 동작

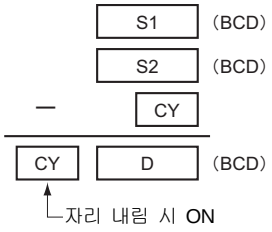
이름	라벨	내용	
		－BC	－BCL
에러 플래그	P_ER	· S1의 데이터가 BCD가 아닐 때 ON · S2의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF	· S1+1, S1의 데이터가 BCD가 아닐 때 ON · S2+1, S2의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 감산 결과, 자리내림이 있을 때 ON · 그 이외일 때 OFF	· 감산 결과, 자리내림이 있을 때 ON · 그 이외일 때 OFF

−BC/−BCL

기능

■ −BC

S1로 지정한 데이터와 S2로 지정한 데이터를 캐리(CY) 플래그도 포함하여 BCD 감산한 결과를 D에 출력합니다. 결과가 음수가 되는 경우에는 10의 보수로 D에 출력됩니다.



■ −BCL

S1로 지정한 데이터와 S2로 지정한 데이터를 2워드 데이터로 캐리(CY) 플래그도 포함하여 BCD 감산한 결과를 D+1, D에 출력합니다. 결과가 음수가 되는 경우에는 10의 보수로 D+1, D에 출력됩니다.



힌트

· 캐리(CY) 플래그를 클리어하고 싶은 경우는 CLC(클리어 캐리) 명령을 실행하십시오.

· 10의 보수란?

각 자리를 9에서 빼서 그 결과에 1을 더한 값입니다.

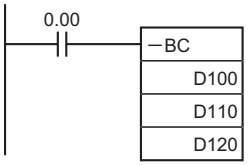
예) 7556에 대한 10의 보수는 9999-7556+1 = 2444입니다.

따라서 4자리 숫자 A에 대한 10의 보수는 9999-A+1 = B가 됩니다.

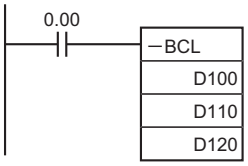
10의 보수 B에서 원래의 수 A를 구할 때는 A = 10000 − B가 됩니다.

예) 10의 보수 2444에서 원래의 수를 구하는 경우 10000-2444 = 7556입니다.

샘플 프로그램



0.00이 ON이면 D100에서 D110 및 CY 플래그가 BCD 4자리로 감산되어 결과가 D120에 출력됩니다.



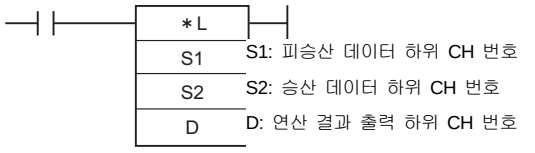
0.00이 ON이면 D101, D100에서 D111, D110 및 CY 플래그가 BCD 8자리로 감산되어 결과가 D121, D120에 출력됩니다.

감산 결과가 음수(S1 < S2 또는 S1+1, S1 < S2+1, S2)이면, 감산 결과는 10의 보수로 출력됩니다. 이때, CY 플래그는 ON이 됩니다.

10의 보수를 원래의 수로 변환하는 경우는 CY 플래그를 입력 조건으로 하고, 다시 0에서 감산 결과를 빼는 프로그램이 필요합니다. 이때 음수라는 것을 나타내기 위한 표시 등에 CY 플래그를 사용합니다.

* / * L

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부호 있는 BIN 승산	*	@ *	420	채널 데이터나 상수를 부호 있는 16진수 4자리로 곱합니다.
부호 있는 BIN 2워드 승산	* L	@ * L	421	2CH분의 채널 데이터나 상수를 부호 있는 16진수 8자리로 곱합니다.

심볼	*	* L
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		*	* L	*	* L
S1	*: 피승산 데이터 * L: 피승산 데이터 하위 CH 번호	INT	DINT	1	2
S2	*: 승산 데이터 * L: 승산 데이터 하위 CH 번호	INT	DINT	1	2
D	연산 결과 출력 하위 CH 번호	DINT	LINT	2	4

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
*	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	-	○	-	-	-	-
	D											-	-						
* L	S1,S2	○	○	○	○	○	○	○	○	○	○	○	-	-	○	-	-	-	-
	D											-							

조건 플래그의 동작

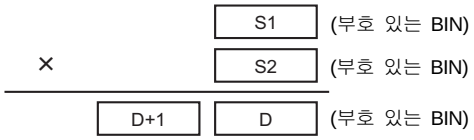
이름	라벨	내용	
		*	* L
에러 플래그	P_ER	OFF	OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 결과의 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF	· 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

* / *L

기능

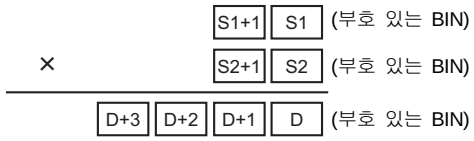
■ *

S1로 지정한 데이터와 S2로 지정한 데이터의 부호 있는 BIN 곱셈을 수행하여 결과를 D+1, D에 출력합니다.

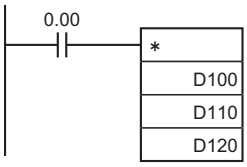


■ *L

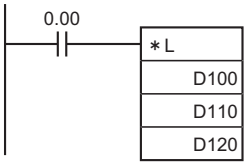
S1로 지정한 데이터와 S2로 지정한 데이터를 2워드 데이터로 하여, 부호 있는 BIN 곱셈을 수행한 결과를 D+3, D+2, D+1, D에 출력합니다.



샘플 프로그램



0.00이 ON이면 D100과 D110이 부호 있는 16진수 4자리로 곱해져 결과가 D121, D120에 출력됩니다.

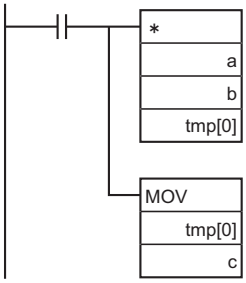


0.00이 ON이면 D101, D100과 D111, D110이 부호 있는 16진수 8자리로 곱해져 결과가 D123, D122, D121, D120에 출력됩니다.

■ 펑션 블록 내 정의에서 사용한 예

배열 변수를 사용하여 곱셈 결과의 1채널분만 가져옵니다.

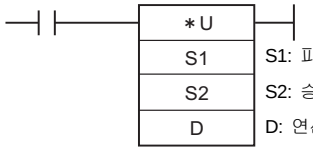
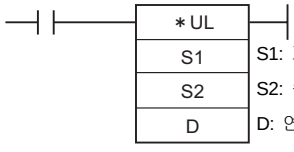
a * b → c



사용 변수:
- 비승산 데이터 a(데이터 형식 INT)
- 승산 데이터 b(데이터 형식 INT)
- 결과 c(데이터 형식 INT)
- 임시 변수 tmp(데이터 형식 WORD, 배열 2 요소)

* U/ * UL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부호 없는 BIN 승산	* U	@ * U	422	채널 데이터나 상수를 부호 없는 16진수 4자리로 곱합니다.
부호 없는 BIN 2워드 승산	* UL	@ * UL	423	2CH분의 채널 데이터나 상수를 부호 없는 16진수 8자리로 곱합니다.

심볼	* U	* UL
	 S1: 피승산 데이터 S2: 승산 데이터 D: 연산 결과 출력 하위 CH 번호	 S1: 피승산 데이터 하위 CH 번호 S2: 승산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		* U	* UL	* U	* UL
S1	* U: 피승산 데이터 * UL: 피승산 데이터 하위 CH 번호	UINT	UDINT	1	2
S2	* U: 승산 데이터 * UL: 승산 데이터 하위 CH 번호	UINT	UDINT	1	2
D	연산 결과 출력 하위 CH 번호	UDINT	ULINT	2	4

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
* U	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	-	○	-	-	-	-
	D											-	-						
* UL	S1,S2	○	○	○	○	○	○	○	○	○	○	○	-	-	○	-	-	-	-
	D											-							

조건 플래그의 동작

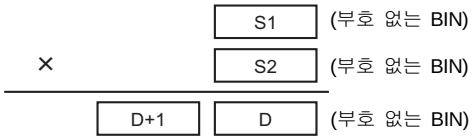
이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

＊U/＊UL

기능

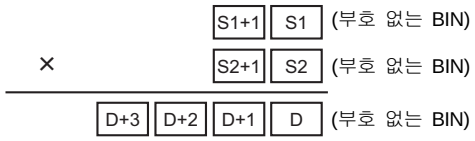
■ ＊U

S1로 지정한 데이터와 S2로 지정한 데이터를 BIN 곱셈하여 결과를 D+1, D에 출력합니다.

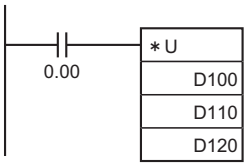


■ ＊UL

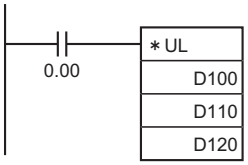
S1로 지정한 데이터와 S2로 지정한 데이터를 2워드 데이터로 하여, 부호 없는 BIN 곱셈을 수행한 결과를 D+3, D+2, D+1, D에 출력합니다.



샘플 프로그램



0.00이 ON이면 D100과 D110이 부호 없는 16진수 4자리로 곱해져 결과가 D121, D120에 출력됩니다.

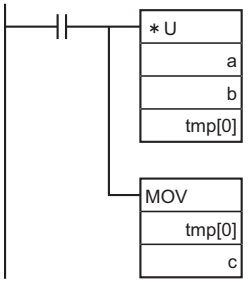


0.00이 ON이면 D101, D100과 D111, D110이 부호 없는 16진수 8자리로 곱해지고 결과가 D123, D122, D121, D120에 출력됩니다.

■ 펑션 블록 내 정의에서 사용한 예

배열 변수를 사용하여 곱셈 결과의 1채널분만 가져옵니다.

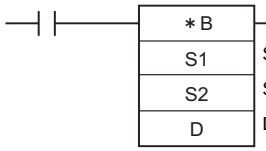
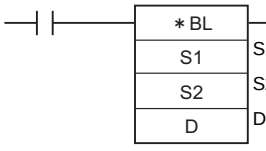
a ＊ b → c



사용 변수:
- 비승산 데이터 a(데이터 형식 UNIT)
- 승산 데이터 b(데이터 형식 UNIT)
- 결과 c(데이터 형식 UNIT)
- 임시 변수 tmp(데이터 형식 WORD, 배열 2 요소)

* B/ * BL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
BCD 승산	* B	@ * B	424	채널 데이터나 상수를 BCD 4자리로 곱합니다.
BCD 2워드 승산	* BL	@ * BL	425	2CH분의 채널 데이터나 상수를 BCD 8자리로 곱합니다.

심볼	* B	* BL
	 S1: 피승산 데이터 S2: 승산 데이터 D: 연산 결과 출력 하위 CH 번호	 S1: 피승산 데이터 하위 CH 번호 S2: 승산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		* B	* BL	* B	* BL
S1	* B: 피승산 데이터 * BL: 피승산 데이터 하위 CH 번호	WORD	DWORD	1	2
S2	* B: 승산 데이터 * BL: 승산 데이터 하위 CH 번호	WORD	DWORD	1	2
D	연산 결과 출력 하위 CH 번호	DWORD	LWORD	2	4

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
* B	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	-	○	-	-	-	-
	D											-	-						
* BL	S1,S2	○	○	○	○	○	○	○	○	○	○	○	-	-	○	-	-	-	-
	D											-							

조건 플래그의 동작

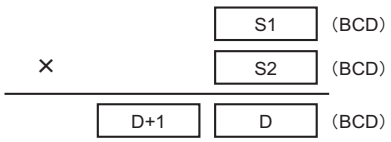
이름	라벨	내용	
		* B	* BL
에러 플래그	P_ER	· S1의 데이터가 BCD가 아닐 때 ON · S2의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF	· S1+1, S1의 데이터가 BCD가 아닐 때 ON · S2+1, S2의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF

＊B/＊BL

기능

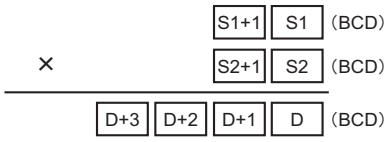
■ ＊B

S1로 지정한 데이터와 S2로 지정한 데이터를 BCD 곱셈하여 결과를 D+1, D에 출력합니다.

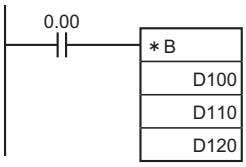


■ ＊BL

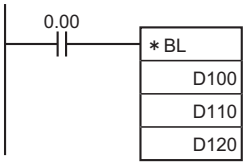
S1로 지정한 데이터와 S2로 지정한 데이터를 2워드 데이터로 BCD 곱셈하여 결과를 D+3, D+2, D+1, D에 출력합니다.



샘플 프로그램



0.00이 ON이면 D100과 D110이 BCD 4자리로 곱해져 결과가 D121, D120에 출력됩니다.

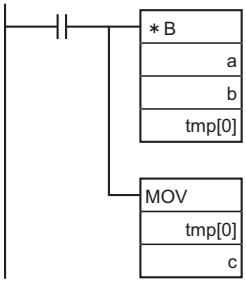


0.00이 ON이면 D101, D100과 D111, D110이 BCD 8자리로 곱해져 결과가 D123, D122, D121, D120에 출력됩니다.

■ 펑션 블록 내 정의에서 사용한 예

배열 변수를 사용하여 곱셈 결과의 1채널분만 가져옵니다.

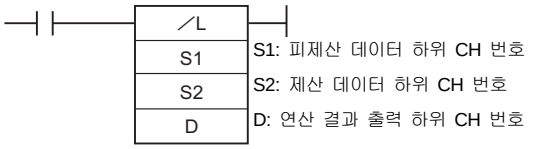
a ＊ b → c



사용 변수:
- 비승산 데이터 a(데이터 형식 WORD)
- 승산 데이터 b(데이터 형식 WORD)
- 결과 c(데이터 형식 WORD)
- 임시 변수 tmp(데이터 형식 WORD, 배열 2 요소)

/ , /L

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부호 있는 BIN 제산	/	@ /	430	채널 데이터나 상수를 부호 있는 16진수 4자리로 나눕니다.
부호 있는 BIN 2워드 제산	/L	@ /L	431	2CH분의 채널 데이터나 상수를 부호 있는 16진수 8자리로 나눕니다.

심볼	/	/L
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		/	/L	/	/L
S1	/: 피제산 데이터 /L: 피제산 데이터 하위 CH 번호	INT	DINT	1	2
S2	/: 제산 데이터 /L: 제산 데이터 하위 CH 번호	INT	DINT	1	2
D	연산 결과 출력 하위 CH 번호	DWORD	LWORD	2	4

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
/	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	-	○	-	-	-	-
	D											-	-						
/L	S1,S2	○	○	○	○	○	○	○	○	○	○	○	-	-	○	-	-	-	-
	D											-							

조건 플래그의 동작

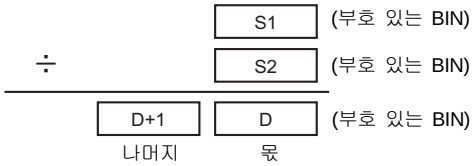
이름	라벨	내용
에러 플래그	P_ER	· 제산 데이터가 0일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과 몫이 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 결과의 몫의 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF

/ , /L

기능

■ /

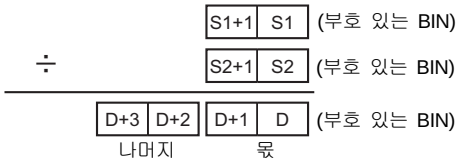
부호 있는 BIN 데이터(16비트)로 S1÷S2를 계산하여 D에 몫(16비트)을 출력하고 D+1에 나머지(16비트)를 출력합니다.



주: 16진수 #8000÷16진수 #FFFF의 나눗셈은 일정치 않습니다.

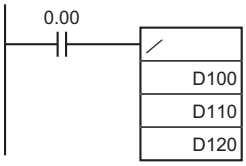
■ /L

부호 있는 BIN 데이터(32비트)로 S1÷S2를 계산하여 D+1, D에 몫(32비트)을 출력하고 D+3, D+2에 나머지(32비트)를 출력합니다.

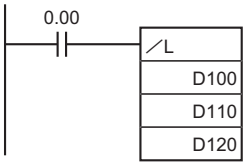


주: 16진수 #80000000÷16진수 #FFFFFFFF의 나눗셈은 일정치 않습니다.

샘플 프로그램



0.00이 ON이면 D100이 D110에서 부호 있는 16진수 4자리로 나눠지고 연산 결과의 몫이 D120에, 나머지가 D121에 출력됩니다.

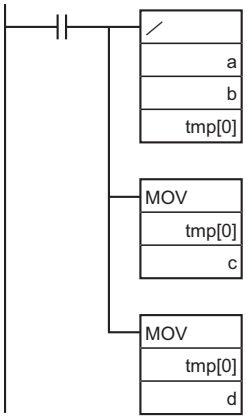


0.00이 ON이면 D101, D100이 D111, D110에서 부호 있는 16진수 8자리로 나눠지고 연산 결과의 몫이 D121, D120에, 나머지가 D123, D122에 출력됩니다.

■ 펑션 블록 내 정의에서 사용한 예

배열 변수를 사용하여 몫과 나머지를 가져옵니다.

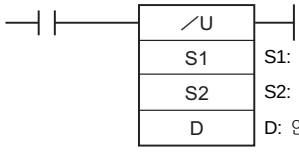
a／b→c…d



사용 변수:
- 비제산 데이터 a(데이터 형식 INT)
- 제산 데이터 b(데이터 형식 INT)
- 몫 c(데이터 형식 INT)
- 나머지 d(데이터 형식 INT)
- 임시 변수 tmp(데이터 형식 WORD, 배열 2 요소)

/ U , / UL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부호 없는 BIN 제산	/ U	@ / U	432	채널 데이터나 상수를 부호 없는 16진수 4자리로 나눕니다.
부호 없는 BIN 2워드 제산	/ UL	@ / UL	433	2CH분의 채널 데이터나 상수를 부호 없는 16진수 8자리로 나눕니다.

심볼	/ U	/ UL
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		/ U	/ UL	/ U	/ UL
S1	/ U: 피제산 데이터 / UL: 피제산 데이터 하위 CH 번호	UINT	UDINT	1	2
S2	/ U: 제산 데이터 / UL: 제산 데이터 하위 CH 번호	UINT	UDINT	1	2
D	연산 결과 출력 하위 CH 번호	DWORD	LWORD	2	4

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
/ U	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	-	○	-	-	-	-
	D											-	-						
/ UL	S1,S2	○	○	○	○	○	○	○	○	○	○	○	-	-	○	-	-	-	-
	D											-							

조건 플래그의 동작

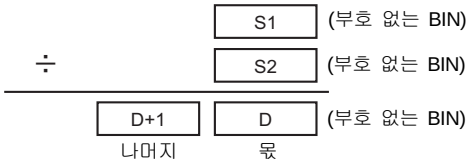
이름	라벨	내용
에러 플래그	P_ER	· 제산 데이터가 0일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과 몫이 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 결과의 몫의 최상위 비트가 1이 되었을 때 ON · 그 이외일 때 OFF

/U, /UL

기능

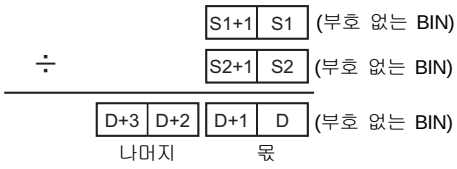
■ /U

부호 없는 BIN 데이터(16비트)로 S1÷S2를 계산하여 D에 몫(16비트)을 출력하고 D+1에 나머지(16비트)를 출력합니다.

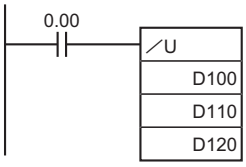


■ /UL

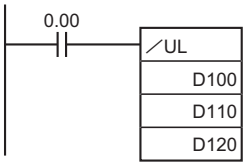
부호 없는 BIN 데이터(32비트)로 S1÷S2를 계산하여 D+1, D에 몫(32비트)을 출력하고 D+3, D+2에 나머지(32비트)를 출력합니다.



샘플 프로그램



0.00이 ON이면 D100이 D110에서 부호 없는 16진수 4자리로 나눠지고 연산 결과의 몫이 D120에, 나머지가 D121에 출력됩니다.

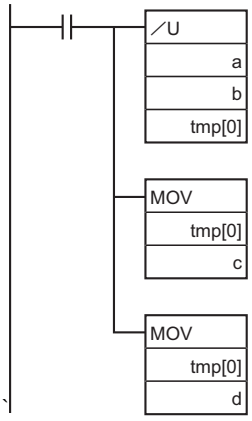


0.00이 ON이면 D101, D100이 D111, D110에서 부호 없는 16진수 8자리로 나눠지고 연산 결과의 몫이 D121, D120에, 나머지가 D123, D122에 출력됩니다.

■ 펑션 블록 내 정의에서 사용한 예

배열 변수를 사용하여 몫과 나머지를 가져옵니다.

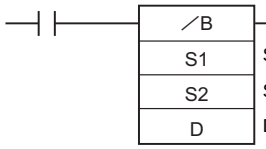
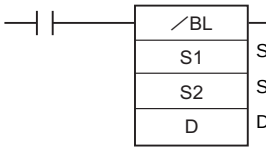
a／b→c…d



사용 변수:
- 비제산 데이터 a(데이터 형식 UNIT)
- 제산 데이터 b(데이터 형식 UNIT)
- 몫 c(데이터 형식 UNIT)
- 나머지 d(데이터 형식 UNIT)
- 임시 변수 tmp(데이터 형식 WORD, 배열 2 요소)

/B, /BL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
BCD 제산	/B	@/B	434	채널 데이터나 상수를 BCD 4자리로 나눕니다.
BCD 2워드 제산	/BL	@/BL	435	2CH분의 채널 데이터나 상수를 BCD 8자리로 나눕니다.

심볼	/B	/BL
	 S1: 피제산 데이터 S2: 제산 데이터 D: 연산 결과 출력 하위 CH 번호	 S1: 피제산 데이터 하위 CH 번호 S2: 제산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		/B	/BL	/B	/BL
S1	/B: 피제산 데이터 /BL: 피제산 데이터 하위 CH 번호	WORD	DWORD	1	2
S2	/B: 제산 데이터 /BL: 제산 데이터 하위 CH 번호	WORD	DWORD	1	2
D	연산 결과 출력 하위 CH 번호	DWORD	LWORD	2	4

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
/B	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	-	○	-	-	-	-
	D											-	-						
/BL	S1,S2	○	○	○	○	○	○	○	○	○	○	○	-	-	○	-	-	-	-
	D											-							

조건 플래그의 동작

이름	라벨	내용	
		/B	/BL
에러 플래그	P_ER	· S1의 데이터가 BCD가 아닐 때 ON · S2의 데이터가 BCD가 아닐 때 ON · 제산 데이터가 0일 때 ON · 그 이외일 때 OFF	· S1+1, S1의 데이터가 BCD가 아닐 때 ON · S2+1, S2의 데이터가 BCD가 아닐 때 ON · 제산 데이터가 0일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과 몫이 0일 때 ON · 그 이외일 때 OFF	· 연산 결과 몫이 0일 때 ON · 그 이외일 때 OFF

/B , /BL

기능

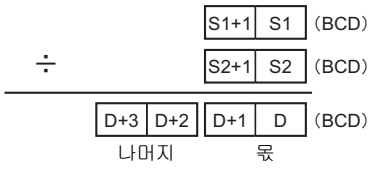
■ /B

BCD 데이터(16비트)로 S1÷S2를 계산하여 D에 몫(16비트)을 출력하고 D+1에 나머지(16비트)를 출력합니다.

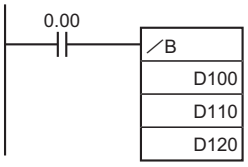


■ /BL

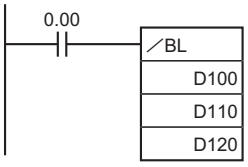
BCD 데이터(32비트)로 S1÷S2를 계산하여 D+1, D에 몫(32비트)을 출력하고 D+3, D+2에 나머지(32비트)를 출력합니다.



샘플 프로그램



0.00이 ON이면 D100이 D110에서 BCD 4자리로 나눠지고 연산 결과의 몫이 D120에, 나머지가 D121에 출력됩니다.

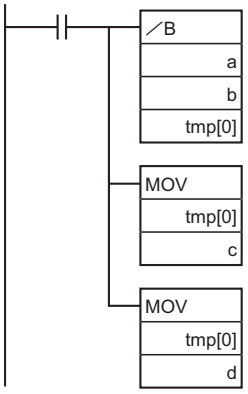


0.00이 ON이면 D101, D100이 D111, D110에서 BCD 8자리로 나눠지고 연산 결과의 몫이 D121, D120에, 나머지가 D123, D122에 출력됩니다.

■ 펑션 블록 내 정의에서 사용한 예

배열 변수를 사용하여 몫과 나머지를 가져옵니다.

a / b → c ··· d



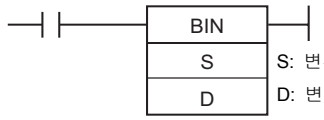
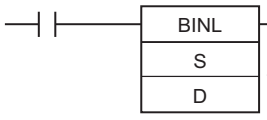
사용 변수:
- 비제산 데이터 a(데이터 형식 WORD)
- 제산 데이터 b(데이터 형식 WORD)
- 몫 c(데이터 형식 WORD)
- 나머지 d(데이터 형식 WORD)
- 임시 변수 tmp(데이터 형식 WORD, 배열 2 요소)

데이터 변환 명령

니모닉	명령 이름	펑션 번호	페이지
BIN	BCD→BIN 변환	023	3-256
BINL	BCD→BIN 2워드 변환	058	
BCD	BIN→BCD 변환	024	3-258
BCDL	BIN→BCD 2워드 변환	059	
NEG	2의 보수 변환	160	3-261
NEGL	2의 보수 변환(2워드)	161	
SIGN	부호 확장	600	3-263
MLPX	4→16/8→256 디코더	076	3-265
DMPX	16→4/256→8 인코더	077	3-270
ASC	ASCII 코드 변환	086	3-275
HEX	ASCII→HEX 변환	162	3-279
LINE	비트 열→비트 행 변환	063	3-283
COLM	비트 행→비트 열 변환	064	3-285
BINS	부호 있는 BCD→BIN 변환	470	3-287
BISL	부호 있는 BCD→BIN 2워드 변환	472	
BCDS	부호 있는 BIN→BCD 변환	471	3-291
BDSL	부호 있는 BIN→BCD 2워드 변환	473	
GRY	그레이 코드 변환	474	3-295
GRAY_BIN	그레이 코드→BIN 변환	478	3-298
GRAY_BINL	그레이 코드→BIN 2워드 변환	479	
BIN_GRAY	BIN→그레이 코드 변환	480	3-300
BIN_GRAYL	BIN→그레이 코드 2워드 변환	481	
STR4	숫자 4자리→ASCII 데이터 변환	601	3-302
STR8	숫자 8자리→ASCII 데이터 변환	602	
STR16	숫자 16자리→ASCII 데이터 변환	603	
NUM4	ASCII 데이터→숫자 4자리 변환	604	3-305
NUM8	ASCII 데이터→숫자 8자리 변환	605	
NUM16	ASCII 데이터→숫자 16자리 변환	606	

BIN/BINL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
BCD→BIN 변환	BIN	@BIN	023	BCD 데이터를 BIN 데이터로 변환합니다.
BCD→BIN 2워드 변환	BINL	@BINL	058	2워드 BCD 데이터를 BIN 데이터로 변환합니다.

심볼	BIN	BINL
	 S: 변환 데이터 CH 번호 D: 변환 결과 출력 CH 번호	 S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		BIN	BINL	BIN	BINL
S	BIN: 변환 데이터 CH 번호 BINL: 변환 데이터 하위 CH 번호	WORD	DWORD	1	2
D	BIN: 변환 결과 출력 CH 번호 BINL: 변환 결과 출력 하위 CH 번호	UINT	UDINT	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
BIN	S	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
	D																		
BINL	S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
	D																		

조건 플래그의 동작

이름	라벨	내용	
		BIN	BINL
에러 플래그	P_ER	· S의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF	· S+1, S의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과가 0일 때 ON · 그 이외일 때 OFF	· 변환 결과가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	OFF	OFF

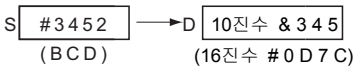
기능

■ BIN

S로 지정한 BCD 데이터를 BIN 변환하여 D에 출력합니다.



(예) S(BCD)→D(BIN)

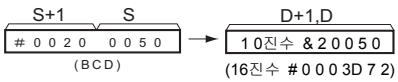


■ BINL

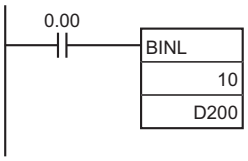
S+1, S로 지정한 BCD 데이터를 2워드 데이터로 BIN 변환하여 결과를 D+1, D에 출력합니다.



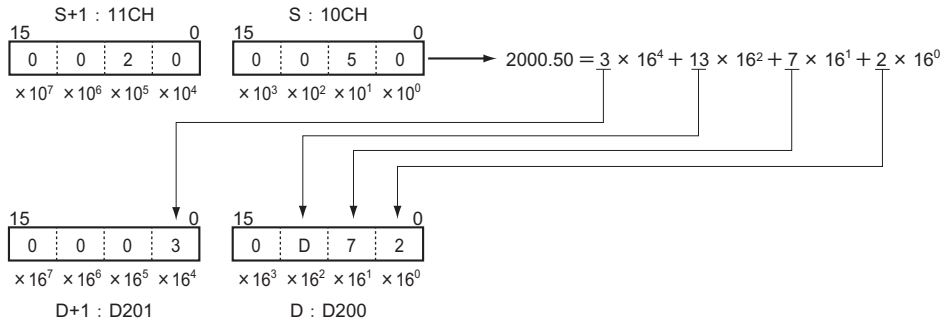
(예) S + 1, S(BCD)→D + 1, D(BIN)



샘플 프로그램



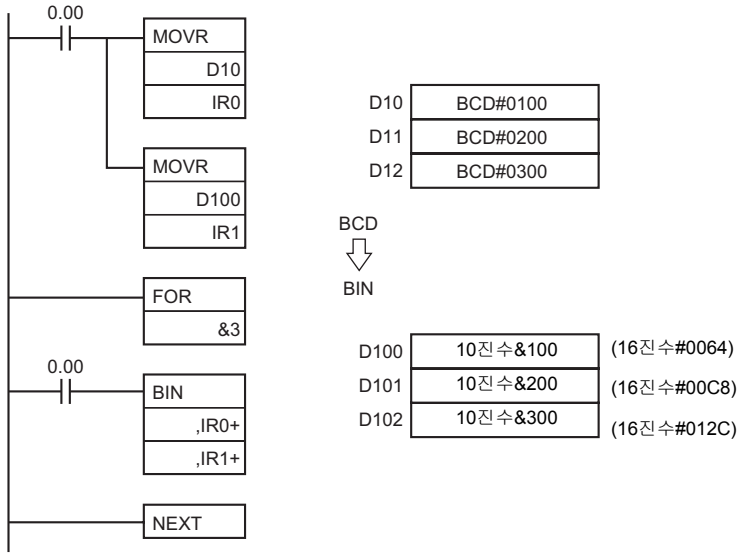
0.00이 ON이면 11CH, 10CH의 BCD 8자리 데이터를 BIN 32비트 데이터로 변환하여 데이터 메모리 D201, D200에 출력합니다.



■ N채널분의 BCD 데이터를 BIN 데이터로 변환

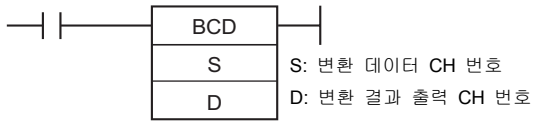
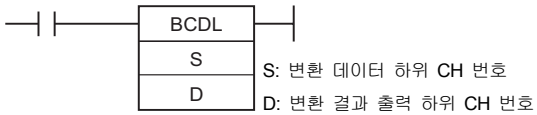
・ N= 3 의 예

0.00이 ON되면 D10부터 3채널분의 BCD 데이터를 1채널마다 BIN 데이터로 변환합니다. 변환된 각 BIN 데이터는 D100 이후에 저장됩니다.



BCD/BCDL

명령 이름	니모닉	동작 옵션	평션 번호	기능
BIN→BCD 변환	BCD	@BCD	024	BIN 데이터를 BCD 데이터로 변환합니다.
BIN→BCD 2워드 변환	BCDL	@BCDL	059	2워드 BIN 데이터를 BCD 데이터로 변환합니다.

심볼	BCD	BCDL
		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		BCD	BCDL	BCD	BCDL
S	BCD: 변환 데이터 CH 번호 BCDL: 변환 데이터 하위 CH 번호	UINT	UDINT	1	2
D	BCD: 변환 결과 출력 CH 번호 BCDL: 변환 결과 출력 하위 CH 번호	WORD	DWORD	1	2

S: 변환 데이터 CH 번호

- BCD

  10진수 &0~9999 또는 16진수 #0000~270F(변환하여 BCD #0000~9999의 값)

- BCDL

  10진수 &0~99999999 또는 16진수 #00000000~05F5E0FF(변환하여 BCD #00000000~99999999의 값)

주: S~S+1 및 D~D+1은 각각 동일 영역 종류이어야 합니다.

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
BCD	S	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
	D																		
BCDL	S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
	D																		

조건 플래그의 동작

이름	라벨	내용	
		BCD	BCDL
에러 플래그	P_ER	· S의 데이터가 10진수 &0~9999 또는 16진수 #0000~270F의 범위가 아닐 때 ON · 그 이외일 때 OFF	· S+1, S의 데이터가 10진수 &0~99999999 또는 16진수 #00000000~05F5E0FF의 범위가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과가 0일 때 ON · 그 이외일 때 OFF	· 변환 결과가 0일 때 ON · 그 이외일 때 OFF

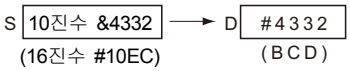
기능

■ BCD

S로 지정한 BIN 데이터를 BCD 변환하여 결과를 D에 출력합니다.



(예) S(BIN)→D(BCD)

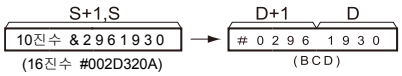


■ BCDL

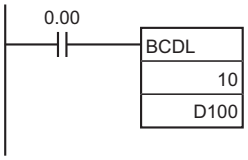
S+1, S로 지정한 BIN 데이터를 2워드 데이터로 BCD 변환하여 결과를 D+1, D에 출력합니다.



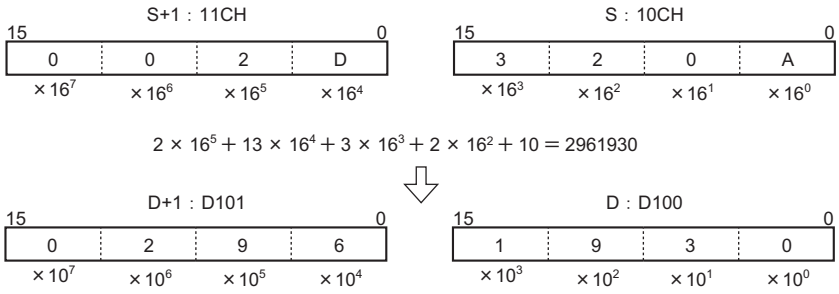
(예) S + 1, S(BIN)→D + 1, D(BCD)



샘플 프로그램



0.00이 ON이면 11CH, 10CH의 BIN 32비트 데이터를 BCD 8자리 데이터로 변환하여 데이터 메모리 D101, D100에 출력합니다.

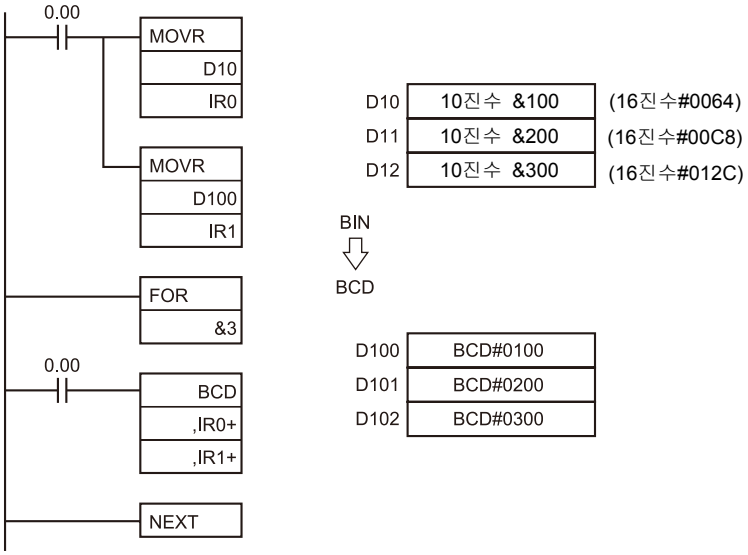


BCD/BCDL

■ N채널분의 BIN 데이터를 BCD 데이터로 변환

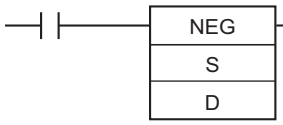
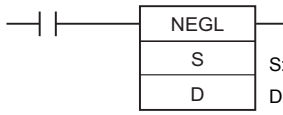
N= 3 의 예

0.00이 ON되면 D10부터 3채널분의 BIN 데이터를 1채널마다 BCD 데이터로 변환합니다. 변환된 각 BCD 데이터는 D100 이후에 저장됩니다.



NEG/NEGL

명령 이름	니모닉	동작 옵션	평션 번호	기능
2의 보수 변환	NEG	@NEG	160	BIN 16비트 데이터의 2의 보수를 계산합니다.
2의 보수 변환(2워드)	NEGL	@NEGL	161	BIN 32비트 데이터의 2의 보수를 계산합니다.

심볼	NEG	NEGL
	 S: 변환 데이터 CH 번호 D: 변환 결과 출력 CH 번호	 S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		NEG	NEGL	NEG	NEGL
S	NEG: 변환 데이터 CH 번호 NEGL: 변환 데이터 하위 CH 번호	WORD	DWORD	1	2
D	NEG: 변환 결과 출력 CH 번호 NEGL: 변환 결과 하위 CH 번호	UINT	UDINT	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
NEG	S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
	D											–							
NEGL	S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
	D											–							

조건 플래그의 동작

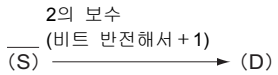
이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 변환 결과가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 변환 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

NEG/NEGL

기능

■ NEG

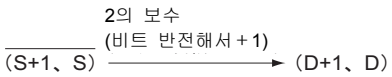
S로 지정한 데이터를 비트 반전하여 1을 더한 결과를 D에 출력합니다.



주: 16진수 #8000의 변환 결과는 16진수 #8000이 됩니다.

■ NEGL

S로 지정한 데이터를 2워드 데이터로 비트 반전하여 1을 더합니다. 결과를 D+1, D에 출력합니다.

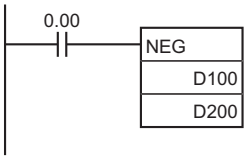


주: 16진수 #80000000의 변환 결과는 16진수 #80000000이 됩니다.

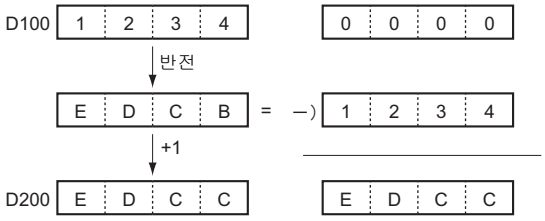
힌트

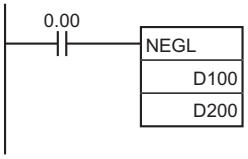
· 비트 반전하여 1을 더하는 조작은 0에서 S의 내용을 빼는 조작에 해당합니다.

샘플 프로그램

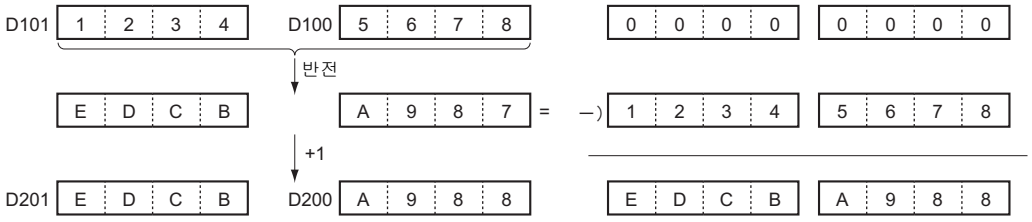


0.00이 ON이면 D100의 내용의 2의 보수를 계산하여 D200에 출력합니다.



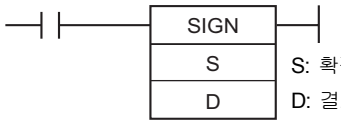


0.00이 ON이면 D101, D100의 내용의 2의 보수를 계산하여 D201, D200에 출력합니다.



SIGN

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부호 확장	SIGN	@SIGN	600	지정 CH의 데이터를 1CH의 부호 있는 BIN 데이터로 2CH에 부호 확장합니다.

심볼	SIGN
	

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	확장 데이터 CH 번호	WORD	1
D	결과 출력 하위 CH 번호	DINT	2

D: 결과 출력 하위 CH 번호
D: S의 내용
D+1: 16진수 #0000 또는 16진수 #FFFF

주: D+1과 D는 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
D											–	–						

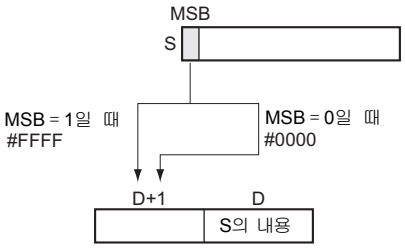
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 확장 결과, D+1, DCH의 내용이 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

SIGN

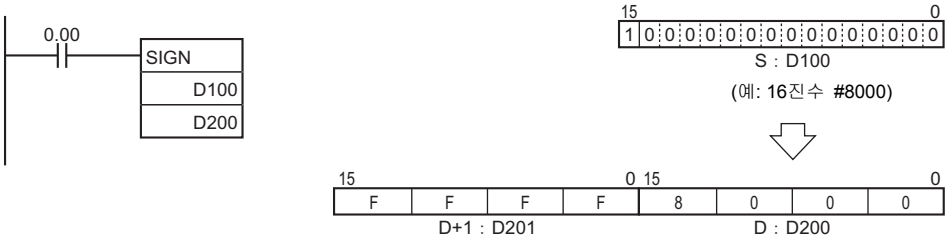
기능

S의 부호 비트(MSB)의 내용이 1일 때 D+1CH에 16진수 #FFFF를 출력하고 0일 때는 D+1CH에 16진수 #0000을 출력합니다. D에는 S의 내용이 그대로 출력됩니다.
그 결과, S의 1CH 데이터가 2CH로 부호 확장되어 D+1, D에 출력됩니다.



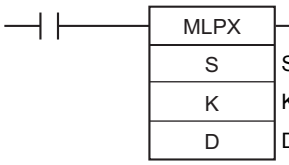
샘플 프로그램

0.00이 ON일 때 D100의 내용을 부호 확장하여 D200에 출력합니다.



MLPX

명령 이름	니모닉	동작 옵션	평션 번호	기능
4→16/8→256 디코더	MLPX	@MLPX	076	지정된 CH의 지정된 자리 또는 바이트를 읽어서(4→16 디코드 또는 8→256 디코드) 지정된 CH의 해당 비트에 1을 출력하고 다른 비트에는 0을 출력합니다.

심볼	MLPX
	

특정 영역 내의 사용 여부

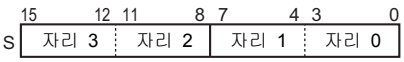
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	변환 데이터 CH 번호	UINT	1
K	제어 데이터(자리 지정)	UINT	1
D	변환 결과 출력 하위 CH 번호	UINT	가변

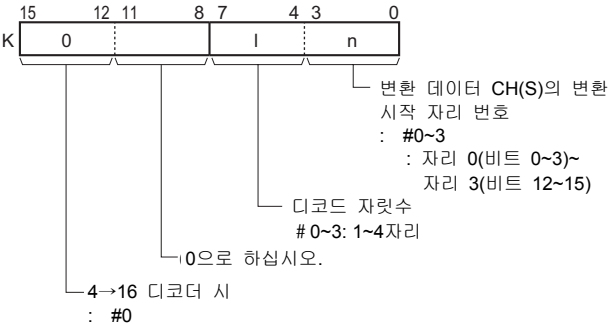
■ 4→16 디코더 시

S: 변환 데이터 CH 번호



시작 자리에서 상위 측 자리를 디코드 대상으로 합니다.
(자리 3의 다음은 자리 0으로 돌아감)

K: 제어 데이터



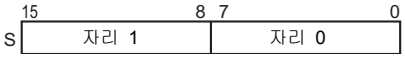
D: 변환 결과 출력 하위 CH 번호

D: 디코드 1번째 자리 디코드 결과
D+1: 디코드 2번째 자리 디코드 결과
D+2: 디코드 3번째 자리 디코드 결과
D+3: 디코드 4번째 자리 디코드 결과

주: D~최대 D+3은 동일 영역 종류이어야 합니다.

■ 8→256 디코더 시

S: 변환 데이터 CH 번호



시작 자리에서 상위 측 자리를 디코드 대상으로 합니다.
(자리 1의 다음은 자리 0으로 돌아감)

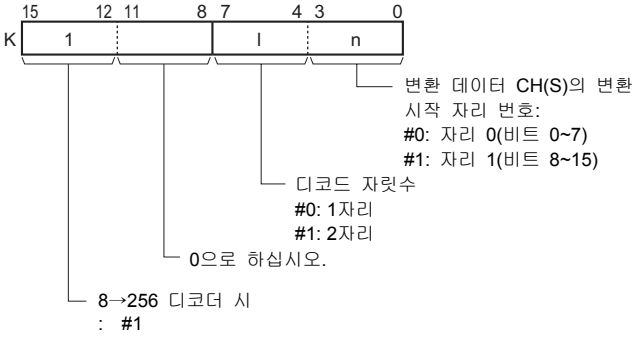
D: 변환 결과 출력 하위 CH 번호

D+15~D: 디코드 1번째 자리 디코드 결과

D+31~D+16: 디코드 2번째 자리 디코드 결과

주: D~최대 D+31은 동일 영역 종류이어야 합니다.

K: 제어 데이터



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
K											○							
D											–	–						

조건 플래그의 동작

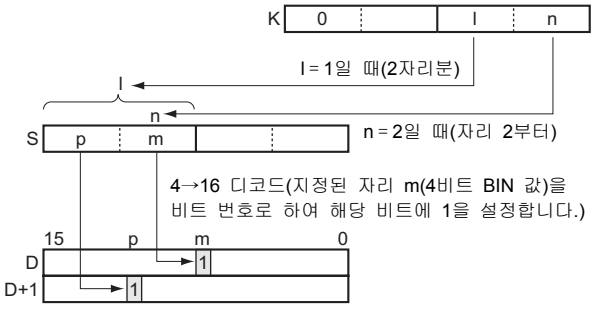
이름	라벨	내용
에러 플래그	P_ER	· K의 데이터가 범위 밖일 때 ON · 그 이외일 때 OFF

기능

변환 종류(K)에 따라 4→16 디코더 또는 8→256 디코더를 지정합니다.

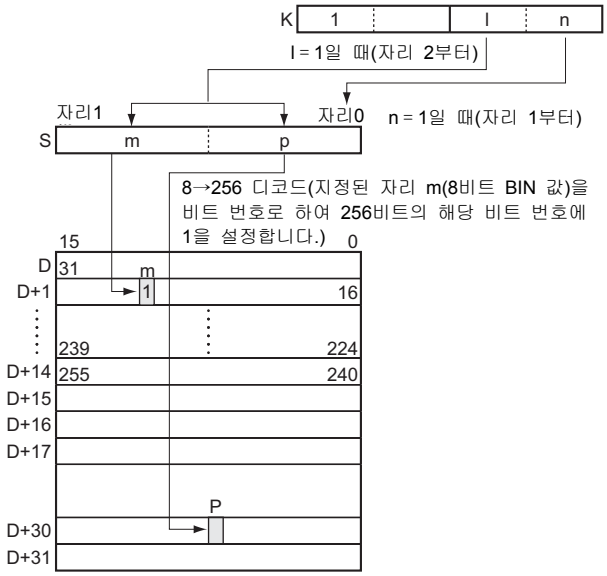
■ 4→16 디코더 지정 시

S로 지정한 데이터의 K로 지정(n 자리부터 I+1개분) 한 각 자리(4비트)의 내용(16진수 #0~F)을 비트 위치(0-15)로 간주하여, D로 지정되는 각 CH(16비트) 이후의 해당 비트에 1을 출력하고 그 이외의 비트에는 0을 출력합니다.



■ 8→256 디코더 지정 시

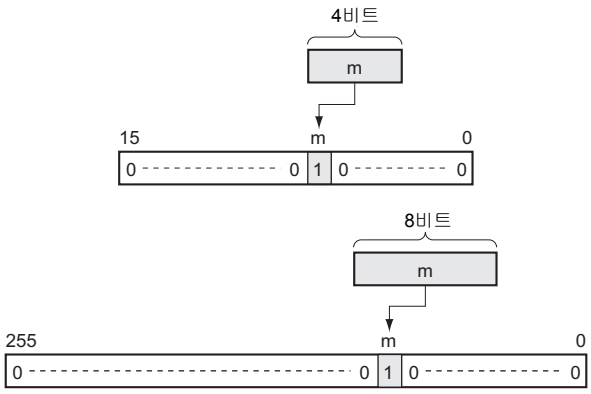
S로 지정한 데이터의 K로 지정(n 자리부터 I+1개분) 한 각 자리(8비트)의 내용(16진수 #00~FF)을 비트 위치(0~255)로 간주하여, D로 지정되는 각 CH(16비트) 이후의 해당 비트에 1을 출력하고 그 이외의 비트에는 0을 출력합니다.



힌트

· 4→16 디코드란 오른쪽 그림과 같이 4비트 BIN 값을 비트 번호로 하여 16비트 중 해당 비트 번호를 1로 설정하고, 그밖에는 0으로 설정하는 것을 말합니다.

· 8→256 디코드란 오른쪽 그림과 같이 8비트 BIN 값을 비트 번호로 하여 256비트 중 해당 비트 번호를 1로 설정하고, 그밖에는 0으로 설정하는 것을 말합니다.



사용상의 주의 사항

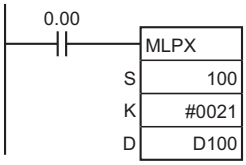
· 4→16 디코더 지정 시

디코드 자릿수(K)에서 여러 자리의 변환을 지정한 경우 변환 대상은 S의 변환 시작 자리(K의 n)에서 상위 자리쪽(자리 3의 다음은 자리 0으로 돌아감) 순서가 되며, 변환 결과는 K에서 상위 CH쪽(1자리당 1CH)의 순서로 저장됩니다.

· 8→256 디코더 지정 시

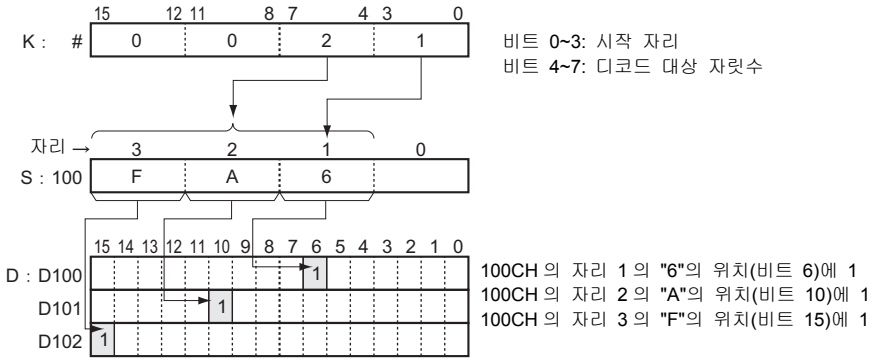
디코드 자릿수(K의 n)에서 여러 자리의 변환을 지정한 경우 변환 대상은 S의 변환 시작 자리(K)에서 상위 자리쪽(자리 1의 다음은 자리 0으로 돌아감)의 순서가 되며, 변환 결과는 K에서 상위 CH쪽(1자리당 16CH)의 순서로 저장됩니다.

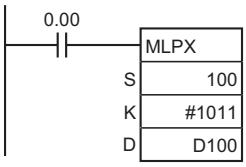
샘플 프로그램



・4→16 디코더

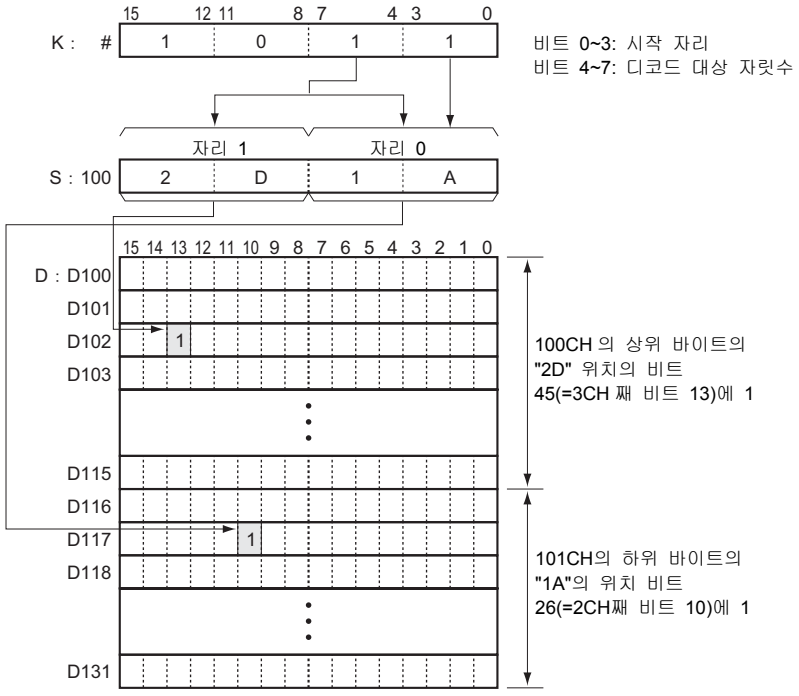
0.00이 ON이면 100CH의 1번째 자리부터 3자리(K의 내용에 따라 다름)의 각 값(16진수)을 비트 번호로 하여 D100~D102의 3CH의 해당 비트에 1을 설정합니다.





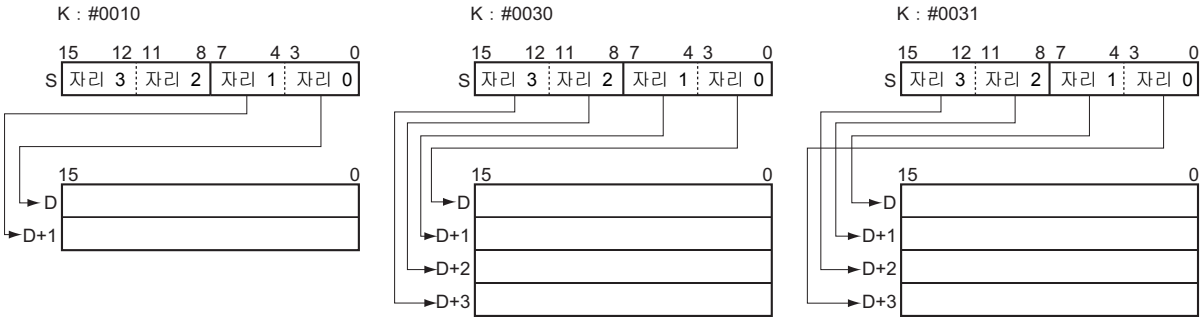
・8 →256 디코더

0.00이 ON이면 100CH의 상위 바이트에서 2바이트(K의 내용에 따라 다름)의 각 값(16진수)을 비트 번호로 하여 D100~D115와 D116~D131의 각 256비트의 해당 비트(하위→상위 순서로 16CH를 연속된 256비트로 간주)에 1을 설정합니다.

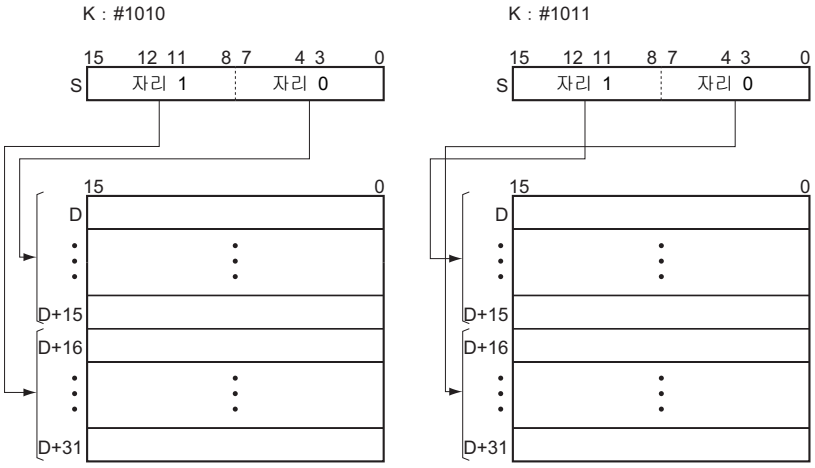


■ 여러 자리의 디코드 예

· 4→16 디코드 예

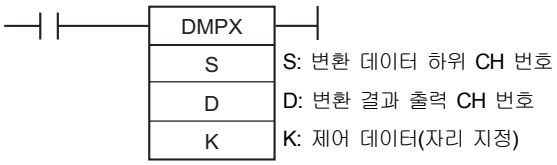


· 8→256 디코드 예



DMPX

명령 이름	니모닉	동작 옵션	펑션 번호	기능
16→4/256→8 인코더	DMPX	@DMPX	077	지정된 CH의 16비트 또는 256비트 중에 ON 상태인 최상위 비트나 최하위 비트를 읽어서(16→4 인코드 또는 256→8 인코드) 지정된 CH의 지정된 자리 또는 바이트에 출력합니다.

심볼	DMPX
	

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

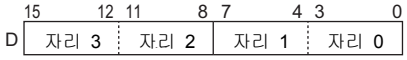
피연산자	내용	데이터 형식	크기
S	변환 데이터 하위 CH 번호	UINT	가변
D	변환 결과 출력 CH 번호	UINT	1
K	제어 데이터(자리 지정)	UINT	1

■ 16→4 인코더 시

S: 변환 데이터 하위 CH 번호

S: 인코드 1번째 자리 인코드 대상
S+1: 인코드 2번째 자리 인코드 대상
S+2: 인코드 3번째 자리 인코드 대상
S+3: 인코드 4번째 자리 인코드 대상

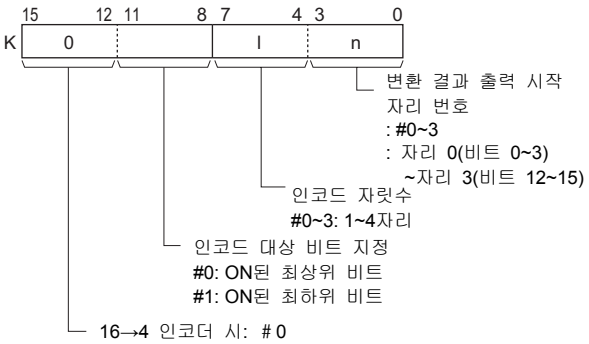
D: 변환 결과 출력 CH 번호



S부터 S+3의 각 인코드 결과를 시작 자리에서 상위 측에 저장(자리 3의 다음은 자리 0으로 돌아감)

주: S~최대 S+3은 동일 영역 종류이어야 합니다.

K: 제어 데이터



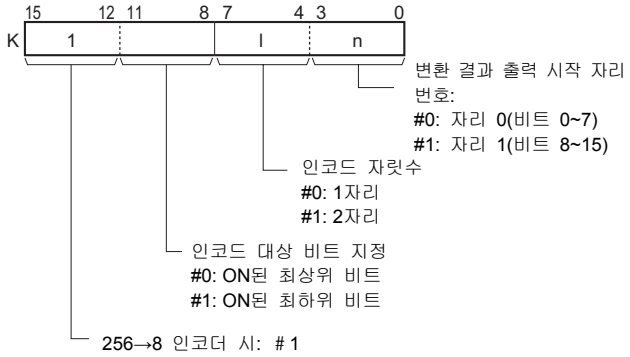
■ 256→8 인코더 시

S: 변환 데이터 하위 CH 번호

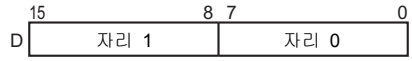
S+15~S: 인코드 1번째 자리 인코드 대상

S+31~S+16: 인코드 2번째 자리 인코드 대상

K: 제어 데이터



D: 변환 결과 출력 CH 번호



S~S+15, S+16~S+31의 각 인코드 결과를 시작 자리에서 상위 측에 저장(자리 1의 다음은 자리 0으로 돌아감)

주: S~최대 S+31은 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D											–	○						
K											○	○						

조건 플래그의 동작

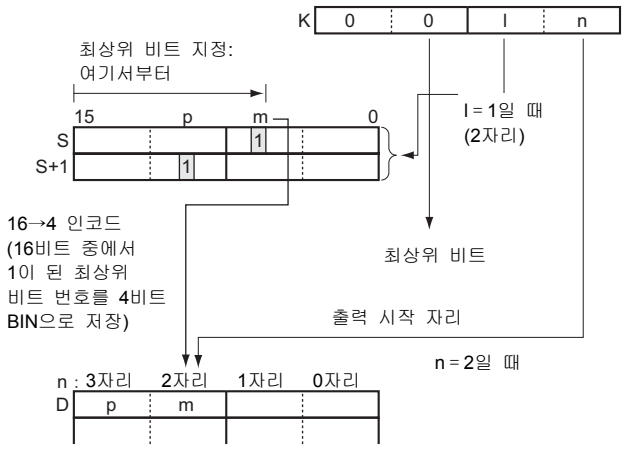
이름	라벨	내용
에러 플래그	P_ER	· S로 지정한 변환 데이터 중 하나가 1CH에서도 0(인코드 가능한 비트가 존재하지 않음)일 때 ON · K의 데이터가 범위 밖일 때 ON · 그 이외일 때 OFF

기능

변환 종류(K)에 따라 16→4 인코더 또는 256→8 인코더를 지정합니다.

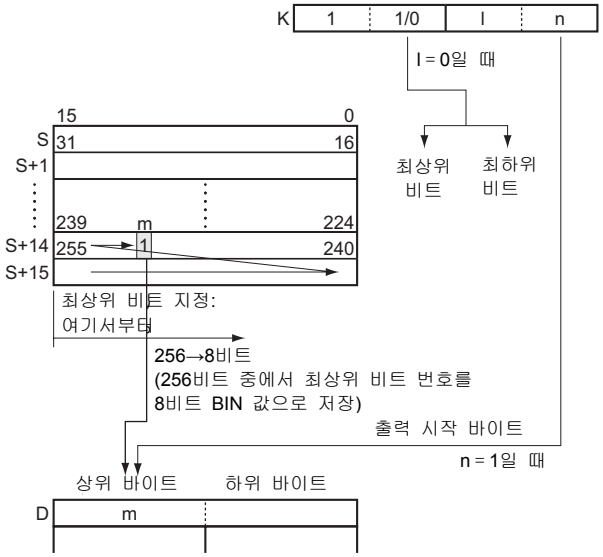
■ 16→4 인코더 지정 시

S에서 K의 I로 지정한 수의 각 16비트(1채널) 중에서 인코드 대상 비트(ON된 비트 중 최상위 비트 또는 최하위 비트)의 비트 번호(0~15)를 읽어 K의 n으로 지정한 D의 출력 시작 자리(4비트) 이후에 비트 번호(16진수 #0~F)를 출력합니다.



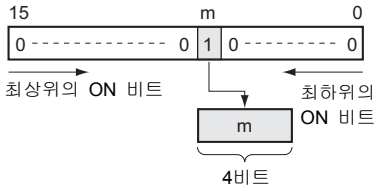
■ 256→8 인코더 지정 시

S에서 K의 I로 지정한 수의 각 256비트(16채널) 중에서 인코드 대상 비트(ON된 비트 중 최상위 비트 또는 최하위 비트)의 비트 번호(0~255)를 읽어 K의 n으로 지정한 D의 출력 시작 자리(8비트) 이후에 비트 번호(16진수 #00~FF)를 출력합니다.

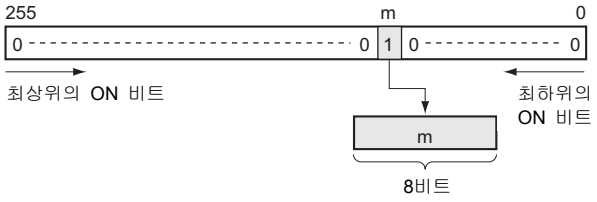


힌트

· 16→4 인코드란 오른쪽 그림과 같이 16비트 중 1이 설정된 최상위 또는 최하위 비트 번호(m)를 4비트 BIN 값으로 변환하는 것을 말합니다.



· 256→8 인코드란 오른쪽 그림과 같이 256비트 중 1이 설정된 최상위 또는 최하위 비트 번호(m)를 8비트 BIN 값으로 변환하는 것을 말합니다.

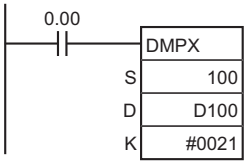


사용상의 주의 사항

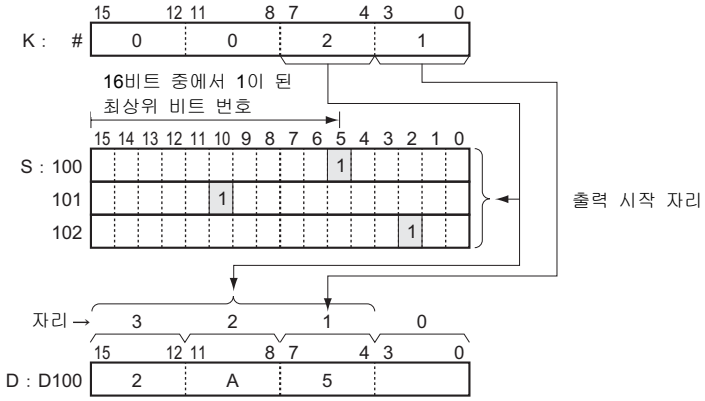
· 16→4 인코더 지정 시
출력 대상 이외의 데이터는 변경하지 않습니다. 인코드 번호(K)에서 여러 자리의 변환을 지정한 경우 변환 대상은 S에서 상위 CH쪽(1자리당 1CH)의 순서가 되며, 변환 결과는 D의 출력 시작 자리(K)에서 상위 자리쪽(자리 3의 다음은 자리 0으로 돌아감)의 순서로 저장됩니다.

· 256→8 인코더 지정 시
출력 대상 이외의 데이터는 변경하지 않습니다. 인코드 자릿수(K)에서 여러 자리의 변환을 지정한 경우 변환 대상은 S에서 상위 CH쪽(1바이트당 16CH)의 순서가 되며, 변환 결과는 D의 출력 시작 자리(K)에서 상위 자리쪽(자리 1의 다음은 자리 0으로 돌아감)의 순서로 저장됩니다.

샘플 프로그램

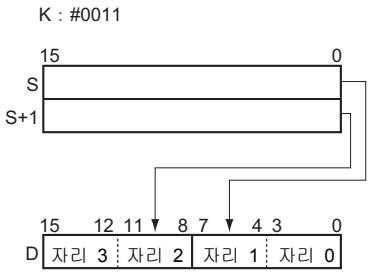


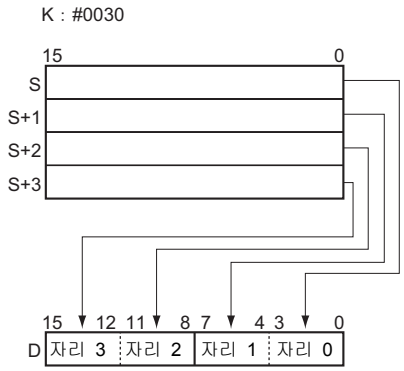
・16→4 인코더
0.00이 ON이면 100CH~102CH의 각 3CH의 16비트 중에서 1이 된 각 최상위 비트 번호를 D100의 1~3자리에 16진수로 저장합니다.

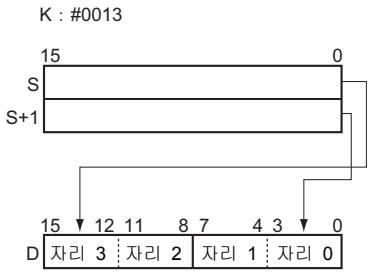


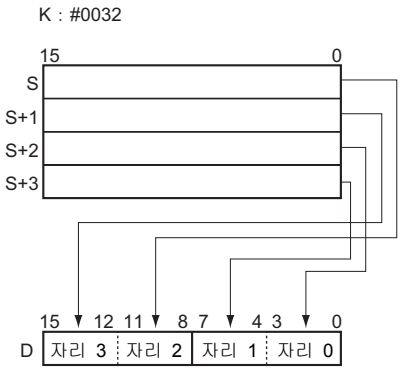
■ 여러 자리의 인코드 예

・ 16→4 인코드 예



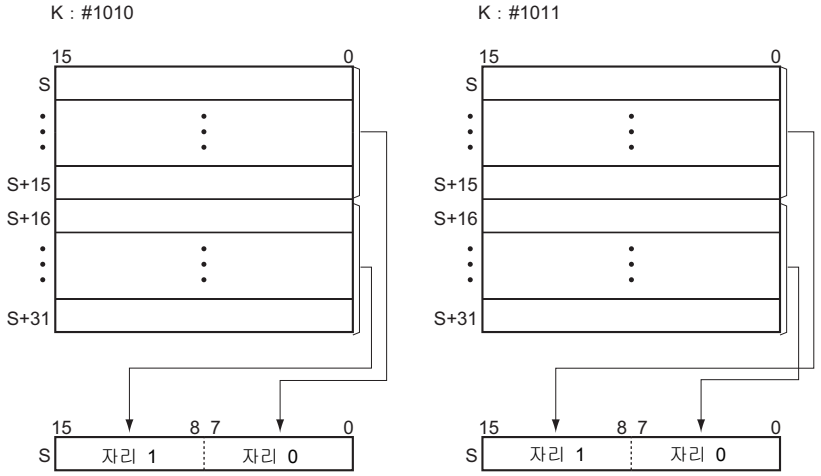






DMPX

· 256→8 인코드 예



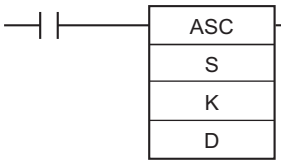
변환 데이터에 0이 포함되어도 그 밖의 데이터에 대해 인코드하는 경우, 복수의 DMPX 명령으로 분할하십시오.

(예) 4자리를 지정한 16→4 인코더인 경우

```
DMPX D0 D100 #0030
        ↓
DMPX D0 D100 #0000
DMPX D1 D100 #0001
DMPX D2 D100 #0002
DMPX D3 D100 #0003
```

ASC

명령 이름	니모닉	동작 옵션	평션 번호	기능
ASCII 코드 변환	ASC	@ASC	086	지정된 16비트 데이터 자리를 ASCII 코드로 변환합니다.

심볼	ASC
	

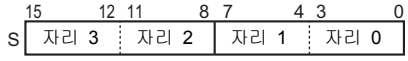
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

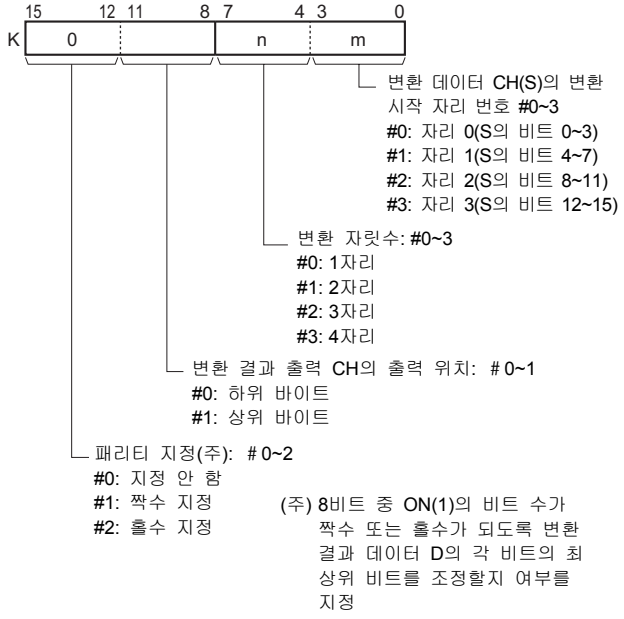
피연산자	내용	데이터 형식	크기
S	변환 데이터 CH 번호	UINT	1
K	자리 지정 데이터	UINT	1
D	변환 결과 출력 하위 CH 번호	UINT	가변

S: 변환 데이터 CH 번호

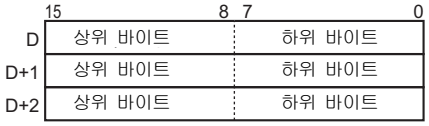


변환 시작 자리에서 상위 자리 측을 HEX 데이터로 간주하여 ASCII 코드로 변환(자리 3의 다음은 자리 0으로 돌아감)

K: 자리 지정 데이터



D: 변환 결과 출력 하위 CH 번호



D의 출력 시작 바이트에서 상위 CH 측으로 하위→ 상위 바이트 순서로 ASCII 코드를 저장

주: D~최대 D+2는 동일 영역 종류이어야 합니다.

ASC

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
K											○							
D											–	–						

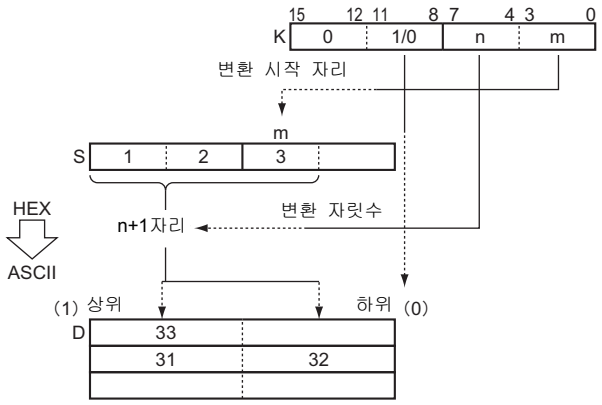
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· K의 데이터가 범위 밖일 때 ON · 그 이외일 때 OFF

기능

S를 4자리의 HEX 데이터로 처리하여 변환 시작 자리 번호(K의 m) 및 변환 자릿수(K의 n)로 지정한 자리의 데이터(16진수 #0~F)를 8비트의 ASCII 코드 데이터("0"(16진수 #30)~ "9"(16진수 #39), "A"(16진수 #41)~ "F"(16진수 #46))로 변환하여 결과를 D에서 K로 지정한 출력 위치(상위 또는 하위부터 저장)에 출력합니다.

또한 ASCII 코드 데이터의 최상위 비트는 패리티 지정(K의 비트 12~15)이 가능하고 홀수 또는 짝수 패리티 비트(8비트 중 1의 비트 수를 홀수 또는 짝수로 조정)로 변환할 수 있습니다.



힌트

· 패리티 비트는 데이터 전송 시의 오류 검출을 위해 데이터에 추가하는 비트입니다. 이 비트를 추가하면 데이터 중 1의 개수를 항상 홀수 또는 짝수로 전송하여 수신 측에서 1의 개수가 홀수 또는 짝수가 아닌 경우 전송에 오류가 발생했다고 판단합니다.

· CJ2 CPU 유니트 및 유니트 버전 4.0 이상의 CS1-H/CJ1-H CPU 유니트의 경우, 숫자 4자리/8자리/16자리→ASCII 데이터 변환 명령(STR4/STR8/STR16)이 있습니다.

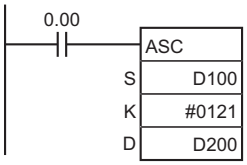
사용상의 주의 사항

· 변환 자릿수(K)에 여러 자리의 변환을 지정한 경우 변환 대상 자리는 시작 자리에서 상위 자리쪽의 순서가 되며(자리 3의 다음은 자리 0으로 돌아감), 변환 결과는 D의 출력 위치에서 상위 CH쪽(8비트 단위)의 순서로 저장됩니다.

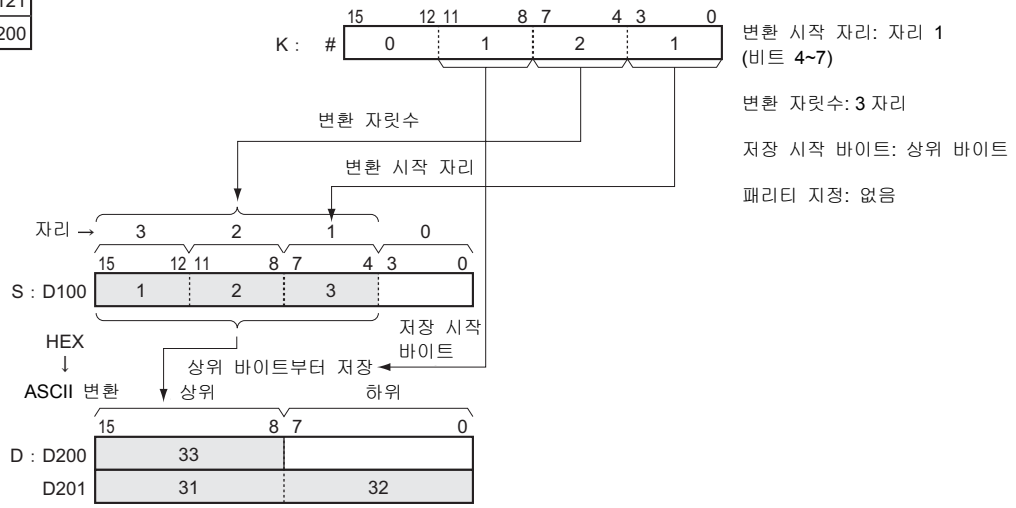
· 변환 결과 출력 CH의 데이터 중 출력 대상이 아닌 위치의 데이터는 유지됩니다.

· 여러 자리의 변환인 경우, D+1, D+2CH가 영역을 초과하지 않도록 주의하십시오.

샘플 프로그램



0.00이 ON이면 D100을 4자리 HEX(16진수) 데이터로 간주하여 제어 데이터의 내용에 따라 자리 1~3의 3자리를 ASCII 코드로 변환하여 D200의 상위 바이트에서 3바이트에 저장합니다.



■ ASCII 코드 변환 예

변환 데이터 자리 내용					변환 출력 데이터								
숫자	비트 내용				코드	(MSB) 비트 내용 (LSB)							
0	0	0	0	0	#30	*	0	1	1	0	0	0	0
1	0	0	0	1	#31	*	0	1	1	0	0	0	1
2	0	0	1	0	#32	*	0	1	1	0	0	1	0
3	0	0	1	1	#33	*	0	1	1	0	0	1	1
4	0	1	0	0	#34	*	0	1	1	0	1	0	0
5	0	1	0	1	#35	*	0	1	1	0	1	0	1
6	0	1	1	0	#36	*	0	1	1	0	1	1	0
7	0	1	1	1	#37	*	0	1	1	0	1	1	1
8	1	0	0	0	#38	*	0	1	1	1	0	0	0
9	1	0	0	1	#39	*	0	1	1	1	0	0	1
A	1	0	1	0	#41	*	1	0	0	0	0	0	1
B	1	0	1	1	#42	*	1	0	0	0	0	1	0
C	1	1	0	0	#43	*	1	0	0	0	0	1	1
D	1	1	0	1	#44	*	1	0	0	0	1	0	0
E	1	1	1	0	#45	*	1	0	0	0	1	0	1
F	1	1	1	1	#46	*	1	0	0	0	1	1	0

*: 패리티 비트…패리티 지정에 의해 변경됩니다.

■ 패리티 지정과 패리티 비트의 내용

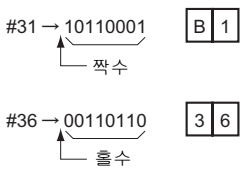
· 패리티 지정 0…패리티 지정 안 함

패리티 비트의 내용은 0이 출력됩니다.

· 패리티 지정 1…짝수 패리티 지정

패리티 비트의 내용은 다른 7비트 중 ON된 비트의 수가 홀수이면 ON이 되고 짝수이면 OFF가 되며, 8비트 중 ON된 비트의 합계가 짝수가 되도록 변경됩니다.

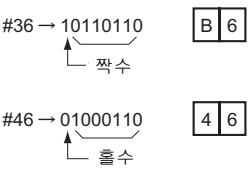
예:



· 패리티 지정 2…홀수 패리티 지정

패리티 비트의 내용은 다른 7비트 중 ON된 비트의 수가 짝수이면 ON이 되고 홀수이면 OFF가 되며, 8비트 중 ON된 비트의 합계가 홀수가 되도록 변경됩니다.

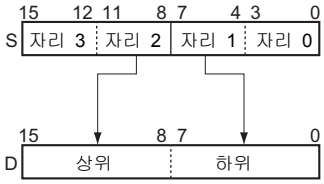
예:



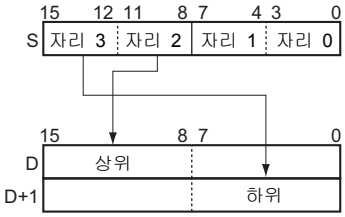
ASC

■ 여러 자리의 ASCII 코드 변환 예

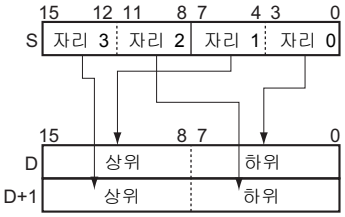
K : #0011



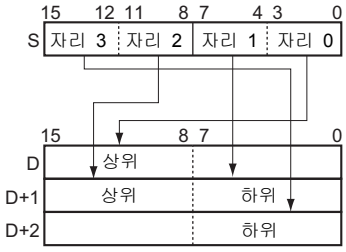
K : #0112



K : #0030

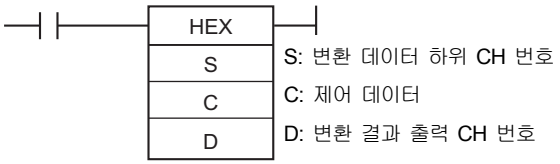


K : #0130



HEX

명령 이름	니모닉	동작 옵션	펑션 번호	기능
ASCII→HEX 변환	HEX	@HEX	162	지정된 CH 데이터 자리의 내용을 8비트 ASCII 데이터로 처리하여 지정된 CH의 지정된 자리에 대응하는 문자로 출력합니다.

심볼	HEX
	

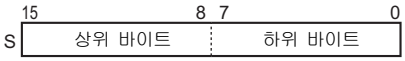
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

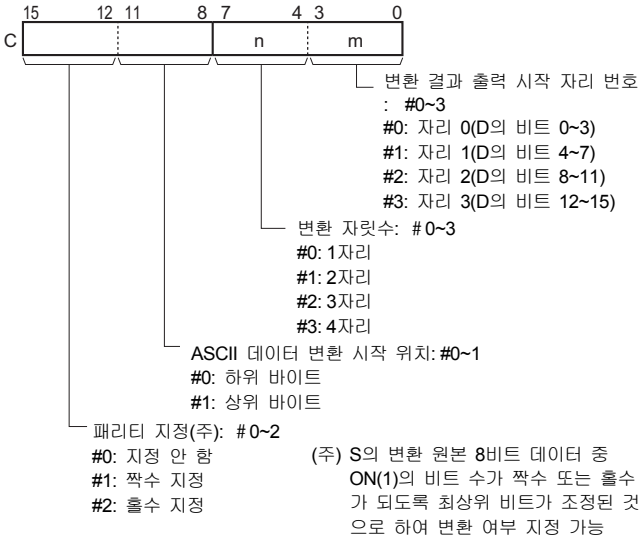
피연산자	내용	데이터 형식	크기
S	변환 데이터 하위 CH 번호	UINT	가변
C	제어 데이터	UINT	1
D	변환 결과 출력 CH 번호	UINT	1

S: 변환 데이터 하위 CH 번호



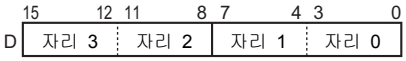
변환 시작 바이트에서 상위 측을 ASCII 코드로 간주하여 HEX로 변환(상위 바이트의 다음은 하위 바이트로 돌아감)

C: 제어 데이터



(주) S의 변환 원본 8비트 데이터 중 ON(1)의 비트 수가 짝수 또는 홀수가 되도록 최상위 비트가 조정된 것으로 하여 변환 여부 지정 가능

D: 변환 결과 출력 CH 번호



HEX로 변환한 결과를 시작 자리에서 상위 측에 저장 (자리 3의 다음은 자리 0으로 돌아감)

HEX

■ 영역 종류

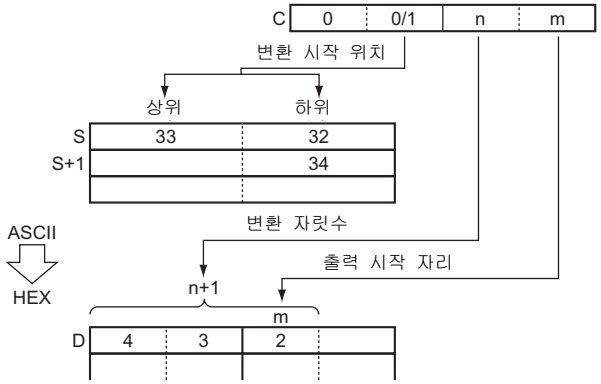
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
C											○	○						
D											–	–						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S의 ASCII 코드 데이터가 패리티 에러일 때 ON · S의 데이터가 변환할 수 없는 ASCII 코드 데이터일 때 ON · C의 데이터가 범위 밖일 때 ON · 그 이외일 때 OFF

기능

S를 상위 8비트, 하위 8비트의 ASCII 코드 데이터("0"(16진수 #30)－"9"(16진수 #39), "A"(16진수 #41)－"F"(16진수 #46))로 처리하여 HEX 데이터로 변환한 결과를 D에 출력합니다. 변환 시작 위치(C)로 지정한 데이터를 1자리(4비트)의 HEX 데이터(16진수 #0-F)로 변환하여 결과를 D의 지정된 출력 시작 자리(C)에 출력합니다. ASCII 코드 데이터의 최상위 비트를 패리티 지정(C)에 따른 홀수 또는 짝수 패리티 비트로 간주하여 데이터를 변환할 수 있습니다.



힌트

· 패리티 비트는 데이터 전송 시의 오류 검출을 위해 데이터에 추가하는 비트입니다. 이 비트를 추가하면 데이터 중 1의 개수를 항상 홀수 또는 짝수로 전송하여 수신 측에서 1의 개수가 홀수 또는 짝수가 아닌 경우 전송에 오류가 발생했다고 판단합니다.

· CJ2 CPU 유니트 및 유니트 버전 4.0 이상의 CS1-H/CJ1-H CPU 유니트의 경우, ASCII 데이터→숫자 4자리/8자리/16자리 변환 명령(NUM4/NUM8/NUM16)이 있습니다.

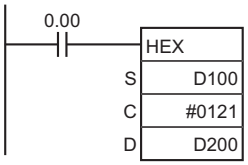
사용상의 주의 사항

· 변환 자릿수(C)에서 여러 자리의 변환을 지정한 경우 변환 대상은 S의 출력 시작 위치(C)에서 상위 CH쪽의 순서가 되며, 변환 결과는 D의 출력 시작 자리(C)에서 상위 자리쪽(자리 3의 다음은 자리 0으로 돌아감)의 순서로 저장됩니다.

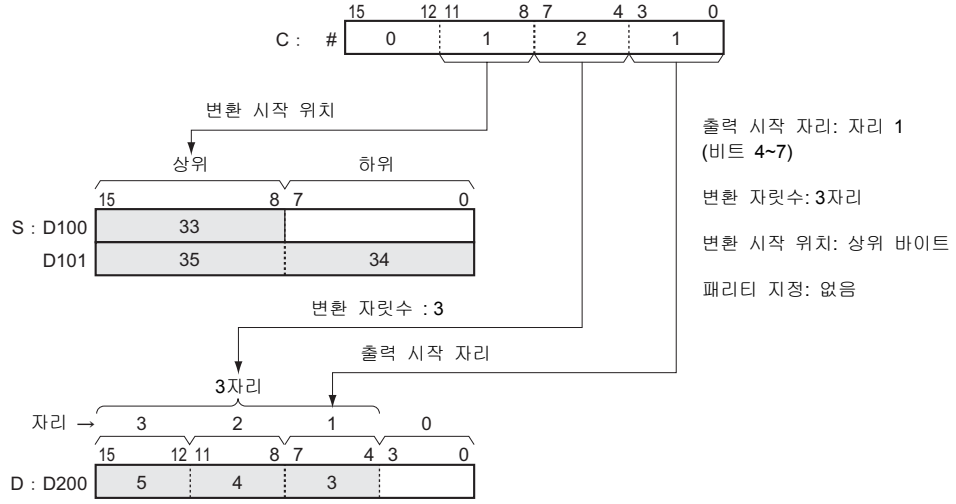
· 변환 결과 출력 CH의 데이터 중 출력 대상이 아닌 자리의 데이터는 유지됩니다(이전 그대로임).

· S로 지정할 수 있는 ASCII 코드 데이터는 다음 범위에 있어야 합니다.
  · 16진수 #30(ASCII 문자 0)~16진수 #39(ASCII 문자 9)
  · 16진수 #41(ASCII 문자 A)~16진수 #46(ASCII 문자 F) (패리티 비트 제외)

샘플 프로그램



0.00이 ON이면 D100~D101을 ASCII 코드 데이터로 간주하고 #0121에 기초하여 D100의 상위 바이트에서 3바이트(3문자 ASCII 코드)를 HEX(16진수) 데이터로 변환하여 D200의 자리 1~3의 3자리에 저장합니다.



■ 패리티 지정

0: 지정 안 함

변환 데이터의 패리티 비트가 0일 때만 변환을 실행합니다. 0이 아니면 에러 플래그가 ON이 되고 변환은 실행되지 않습니다.

1: 짝수 지정

변환 데이터(8비트)의 '1'의 개수가 짝수일 때만 변환을 실행합니다. '1'의 개수가 홀수이면 에러 플래그가 ON이 되고 변환은 실행되지 않습니다.

예:

#31 10110001 ('1'의 개수가 짝수) → 변환 실행 → '1'

00110001 ('1'의 개수가 홀수) → 변환 실행 → 변환되지 않음

2: 홀수 지정

변환 데이터(8비트)의 '1'의 개수가 홀수일 때만 변환을 실행합니다. '1'의 개수가 짝수이면 에러 플래그가 ON이 되고 변환은 실행되지 않습니다.

예:

#41 11000001 ('1'의 개수가 홀수) → 변환 실행 → 'A'

01000001 ('1'의 개수가 짝수) → 변환 실행 → 변환되지 않음

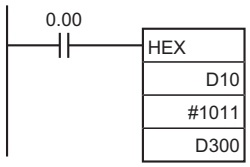
변환 예

변환 데이터									변환 결과(Hex 데이터)				
ASCII 코드	(MSB) 비트 내용 (LSB)								숫자	비트 내용			
$30	*	0	1	1	0	0	0	0	0	0	0	0	0
$31	*	0	1	1	0	0	0	1	1	0	0	0	1
$32	*	0	1	1	0	0	1	0	2	0	0	1	0
$33	*	0	1	1	0	0	1	1	3	0	0	1	1
$34	*	0	1	1	0	1	0	0	4	0	1	0	0
$35	*	0	1	1	0	1	0	1	5	0	1	0	1
$36	*	0	1	1	0	1	1	0	6	0	1	1	0
$37	*	0	1	1	0	1	1	1	7	0	1	1	1
$38	*	0	1	1	1	0	0	0	8	1	0	0	0
$39	*	0	1	1	1	0	0	1	9	1	0	0	1
$41	*	1	0	0	0	0	0	1	A	1	0	1	0
$42	*	1	0	0	0	0	1	0	B	1	0	1	1
$43	*	1	0	0	0	0	1	1	C	1	1	0	0

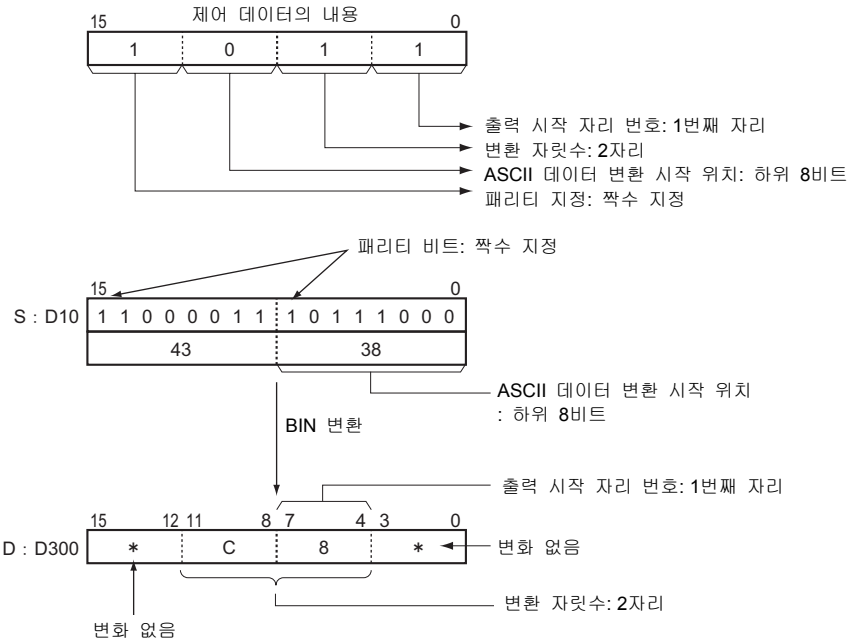
HEX

변환 데이터									변환 결과(Hex 데이터)				
ASCII 코드	(MSB) 비트 내용 (LSB)								숫자	비트 내용			
$44	*	1	0	0	0	1	0	0	D	1	1	0	1
$45	*	1	0	0	0	1	0	1	E	1	1	1	0
$46	*	1	0	0	0	1	1	0	F	1	1	1	1

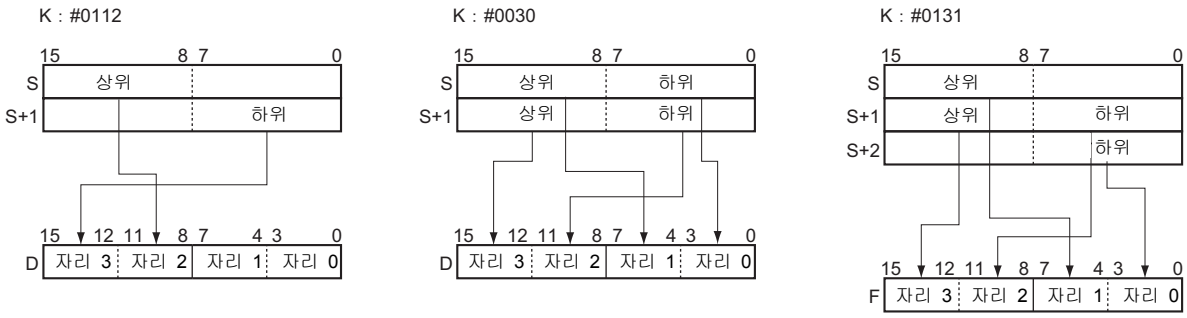
*: 패리티 비트…패리티 지정에 의해 변경됩니다.



0.00이 ON이면 제어 데이터 내용에 따라 데이터 메모리 D10의 ASCII 코드 데이터가 BIN 데이터로 변환되어 결과가 D300에 출력됩니다.



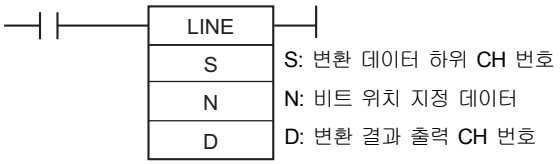
■ 여러 바이트의 ASCII 코드를 HEX 데이터로 변환한 예



LINE

명령 이름	니모닉	동작 옵션	평션 번호	기능
비트 열→비트 행 변환	LINE	@LINE	063	지정된 16CH분의 데이터 중 지정된 비트의 데이터만 지정된 다른 CH의 해당 비트로 설정합니다.

심볼	LINE



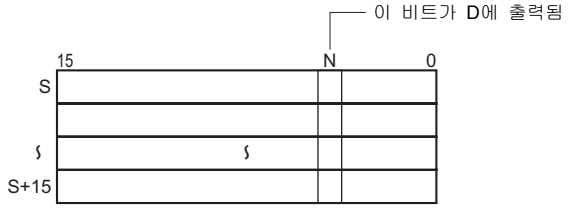
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

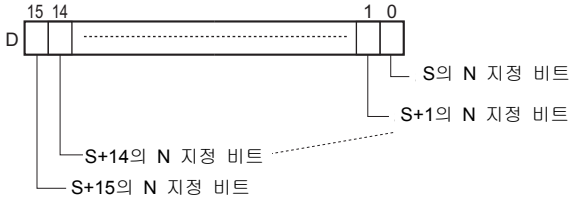
피연산자	내용	데이터 형식	크기
S	변환 데이터 하위 CH 번호	WORD	16
N	비트 위치 지정 데이터	UINT	1
D	변환 결과 출력 CH 번호	UINT	1

S: 변환 데이터 하위 CH 번호



주: S~S+15는 동일 영역 종류이어야 합니다.

D: 변환 결과 출력 CH 번호



N: 비트 위치 지정 데이터

10진수 &0~15 또는 16진수 #0000~000F

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
N											○	○						
D											–							

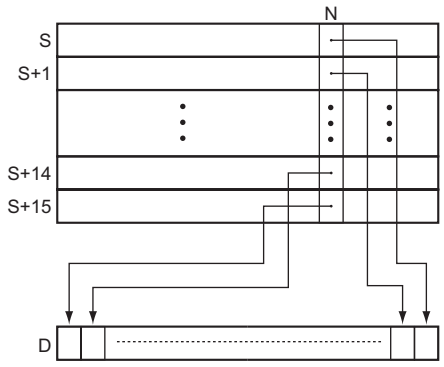
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· N의 데이터가 10진수 &0~15 또는 16진수 #0000~000F의 범위에 없을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과가 0일 때 ON · 그 이외일 때 OFF

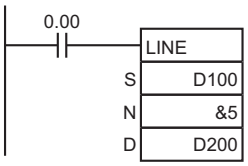
LINE

기능

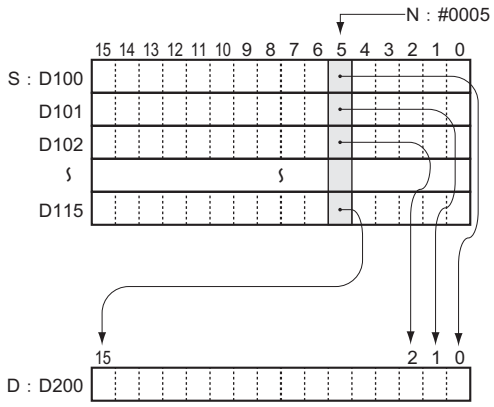
S로 지정한 변환 데이터 하위 CH 번호부터 16CH분의 데이터에서 N으로 지정한 비트 위치의 ON/OFF 내용을 추출하여 16비트의 데이터로 D에 출력합니다.



샘플 프로그램

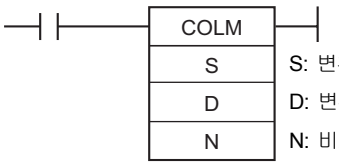


0.00이 ON이면 D100~D115의 비트 5의 비트 열 데이터를 00~15비트 행 데이터로 하여 D200으로 전송합니다.



COLM

명령 이름	니모닉	동작 옵션	펑션 번호	기능
비트 행→비트 열 변환	COLM	@COLM	064	지정된 CH의 각 비트 데이터를 지정된 다른 CH에서 16CH의 지정된 비트로 설정합니다.

심볼	COLM
	

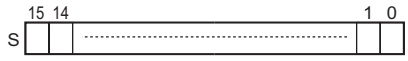
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

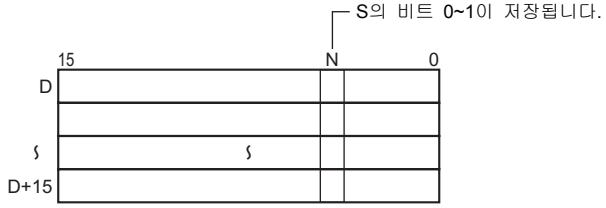
피연산자 설명

피연산자	내용	데이터 형식	크기
S	변환 데이터 하위 CH 번호	WORD	1
D	변환 결과 출력 하위 CH 번호	WORD	16
N	비트 위치 지정 데이터	UINT	1

S: 변환 데이터 하위 CH 번호



D: 변환 결과 출력 하위 CH 번호



D의 N 지정 비트: S의 비트 0을 출력
D+1의 N 지정 비트: S의 비트 1을 출력
:
:
:
D+15의 N 지정 비트: S의 비트 15를 출력

N: 비트 위치 지정 데이터
10진수 &0~15 또는 16진수 #0000~000F

주: D~D+15는 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
D											–	–						
N											○	○						

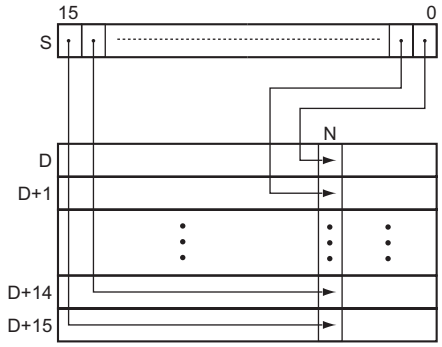
COLM

조건 플래그의 동작

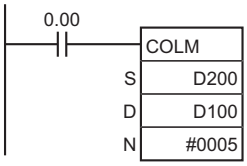
이름	라벨	내용
에러 플래그	P_ER	· N의 데이터가 10진수 &0~15 또는 16진수 #0000~000F의 범위에 없을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과, D~D+15의 지정 비트가 모두 0이 되었을 때 ON · 그 이외일 때 OFF

기능

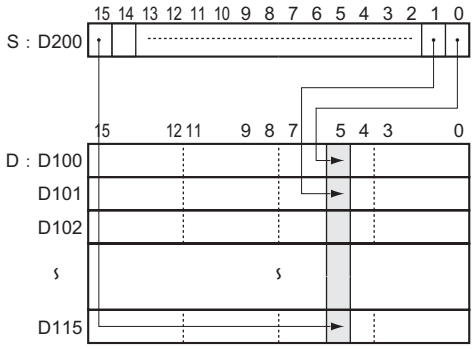
S(16비트)의 각 비트 내용(비트 0에서 비트 15)을 D로 지정한 변환 결과 출력 하위 CH 번호부터 16CH 분의 데이터에 대해 지정된 비트 위치(N)에 각각 출력합니다. 변환 결과 출력 CH의 지정된 비트 이외의 데이터는 유지됩니다(이전 그대로임).



샘플 프로그램



0.00이 ON이면 D200의 0~15비트 행 데이터를 D100~D115CH의 비트 5의 비트 열 데이터로 하여 전송합니다.



BINS/BISL

명령 이름	니모닉	동작 옵션	평션 번호	기능
부호 있는 BCD→BIN 변환	BINS	@BINS	470	부호 있는 BCD 데이터를 부호 있는 BIN 데이터로 변환합니다.
부호 있는 BCD→BIN 2워드 변환	BISL	@BISL	472	부호 있는 2워드 BCD 데이터를 부호 있는 2워드 BIN 데이터로 변환합니다.

심볼	BINS	BISL
	BINS C: C: 데이터 형식 지정 번호 S: S: 변환 데이터 CH 번호 D: D: 변환 결과 출력 CH 번호	BISL C: C: 데이터 형식 지정 번호 S: S: 변환 데이터 하위 CH 번호 D: D: 변환 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

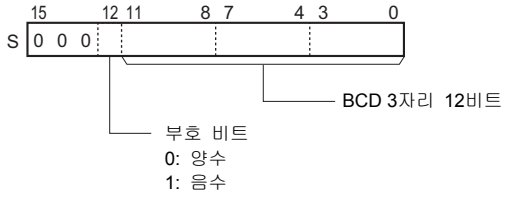
피연산자	내용	데이터 형식		크기	
		BINS	BISL	BINS	BISL
C	데이터 형식 지정 번호	UINT	UINT	1	1
S	변환 데이터 CH 번호	WORD	DWORD	1	2
D	변환 결과 출력 CH 번호	INT	DINT	1	2

C: 데이터 형식 지정 번호
10진수 &0~3 또는 16진수 #0000~0003

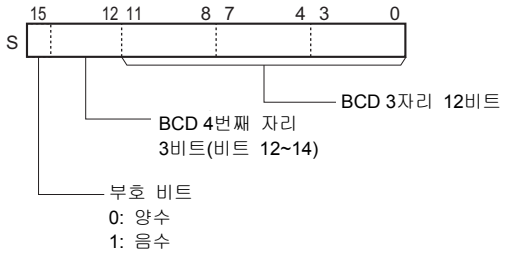
S: 변환 데이터 CH 번호

■ **BINS**

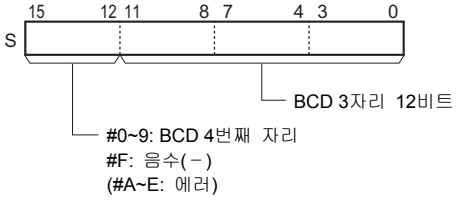
・ C = 0일 때
S의 값: －999~999(BCD)



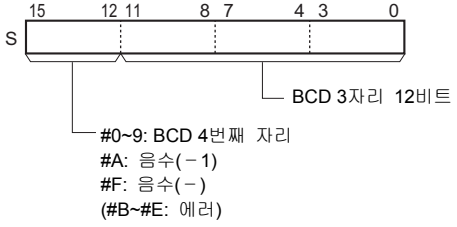
・ C = 1일 때
S의 값: －7999~7999(BCD)



・ C = 2일 때
S의 값: －999~9999(BCD)



・ C = 3일 때
S의 값: －1999~9999(BCD)

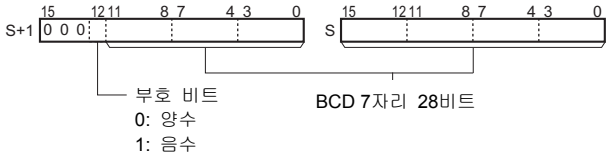


BINS/BISL

■ **BISL**

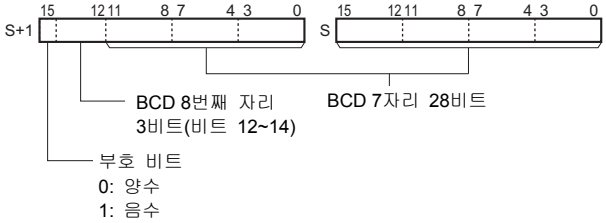
· C = 0일 때

S의 값: −9999999~9999999(BCD)



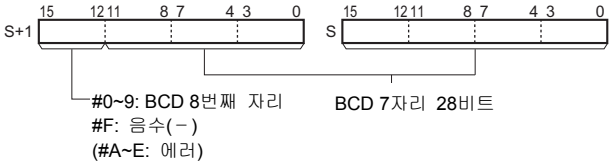
· C = 1일 때

S의 값: −79999999~79999999(BCD)



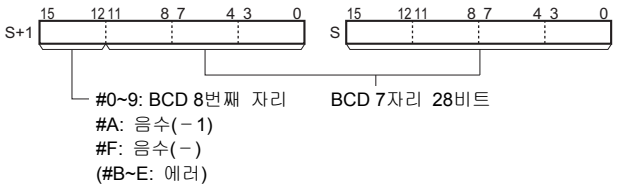
· C = 2일 때

S의 값: −9999999~99999999(BCD)



· C = 3일 때

S의 값: −19999999~99999999(BCD)



■ **영역 종류**

· BINS

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	○	−	○	−	−	−	−
S											−							
D																		

· BISL

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	○	−	○	−	−	−	−
S											−	−						
D																		

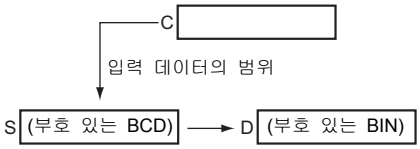
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 변환 데이터의 최상위 자리가 A~E일 때 ON(C에 2를 지정한 경우) · 변환 데이터의 최상위 자리가 B~E일 때 ON(C에 3를 지정한 경우) · C의 데이터가 0~3의 범위에 없을 때 ON · S의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 변환 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

■ BINS

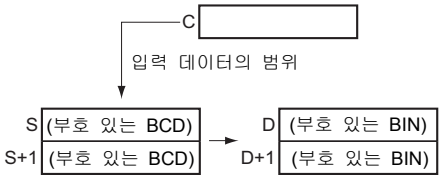
S를 C로 지정한 형식의 부호 있는 BCD 데이터로 하여, 부호 있는 BIN 데이터(16비트)로 변환한 결과를 D에 출력합니다.



데이터 형식 지정 번호	변환 데이터(BCD)	변환 결과 데이터(BIN 16진수)
C	S	D
0	－999~－1	#FC19~FFFF
	0~999	#0000~03E7
1	－7999~－1	#E0C1~FFFF
	0~7999	#0000~1F3F
2	－999~－1	#FC19~FFFF
	0~9999	#0000~270F
3	－1999~－1	#F831~FFFF
	0~9999	#0000~270F

■ BISL

S+1, S를 C로 지정한 형식의 부호 있는 BCD 데이터로 하여, 부호 있는 BIN 데이터(32비트)로 변환한 결과를 D+1, D에 출력합니다. 결과가 음수가 되는 경우에는 2의 보수로 표현합니다.



데이터 형식 지정 번호	변환 데이터(BCD)	변환 결과 데이터(BIN 16진수)
C	S	D
0	－9999999~－1	#FF676981~FFFFFFFF
	0~9999999	#00000000~0098967F
1	－79999999~－1	#FB3B4C01~FFFFFFFF
	0~79999999	#00000000~04C4B3FF
2	－9999999~－1	#FF676981~FFFFFFFF
	0~99999999	#00000000~05F5E0FF
3	－19999999~－1	#FECED301~FFFFFFFF
	0~99999999	#00000000~05F5E0FF

힌트

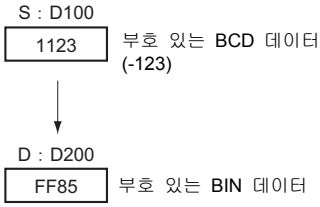
· 고기능 I/O 유니트 중에는 부호 있는 BCD 데이터를 출력 데이터로 하는 경우가 있습니다. 이 경우, BINS/BISL 명령을 사용하여 부호 있는 BIN 데이터로 변환하면 연산이 쉬워집니다.

샘플 프로그램

· BINS 명령의 동작

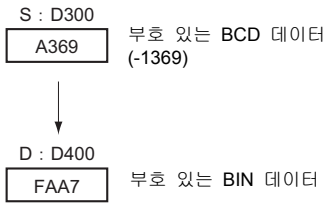
입력 릴레이 0.00이 ON이 되면 먼저 D100의 데이터 형식과 범위가 데이터 형식 지정 0000에 기초하는지 확인합니다. 확인 결과가 정상이므로 데이터 메모리 D100의 부호 있는 BCD 데이터가 부호 있는 BIN 데이터로 변환되어 결과가 D200에 출력됩니다.





입력 릴레이 0.01이 ON이 되면 먼저 D300의 데이터 형식과 범위가 데이터 형식 지정 0003에 기초하는지 확인합니다. 확인 결과가 정상이므로 데이터 메모리 D300의 부호 있는 BCD 데이터가 부호 있는 BIN 데이터로 변환되어 결과가 D400에 출력됩니다.

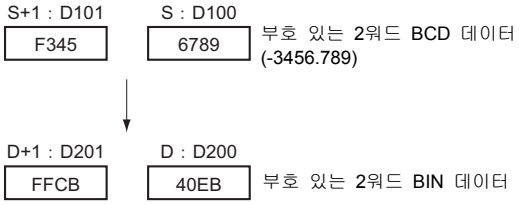




· BISL 명령의 동작

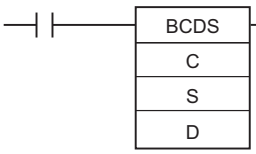
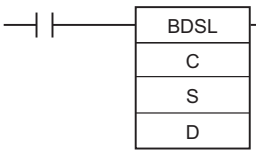
입력 릴레이 0.00이 ON이 되면 먼저 D101, D100의 데이터 형식과 범위가 데이터 형식 지정 0002에 기초하는지 확인합니다. 확인 결과가 정상이므로 데이터 메모리 D101, D100의 부호 있는 BCD 데이터가 부호 있는 BIN 데이터로 변환되어 결과가 D201, D200에 출력됩니다.





BCDS/BDSL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부호 있는 BIN→BCD 변환	BCDS	@BCDS	471	부호 있는 BIN 데이터를 부호 있는 BCD 데이터로 변환합니다.
부호 있는 BIN→BCD 2워드 변환	BDSL	@BDSL	473	부호 있는 2워드 BIN 데이터를 부호 있는 2워드 BCD 데이터로 변환합니다.

심볼	BCDS	BDSL
	 C: 데이터 형식 지정 번호 S: 변환 데이터 CH 번호 D: 변환 결과 출력 CH 번호	 C: 데이터 형식 지정 번호 S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		BCDS	BDSL	BCDS	BDSL
C	데이터 형식 지정 번호	UINT	UINT	1	1
S	변환 데이터 CH 번호	INT	DINT	1	2
D	변환 결과 출력 CH 번호	WORD	DWORD	1	2

C: 데이터 형식 지정 번호
10진수 &0~3 또는 16진수 #0000~0003

S: 변환 데이터 CH 번호

· BCDS

데이터 형식 지정 번호 C	변환 데이터 S
0	10진수 －999~－1 16진수 #FC19~FFFF
	10진수 0~+999 16진수 #0000~03E7
1	10진수 －7999~－1 16진수 #E0C1~FFFF
	10진수 0~+7999 16진수 #0000~1F3F
2	10진수 －999~－1 16진수 #FC19~FFFF
	10진수 0~+9999 16진수 #0000~270F
3	10진수 －1999~－1 16진수 #F831~FFFF
	10진수 0~+9999 16진수 #0000~270F

· BDSL

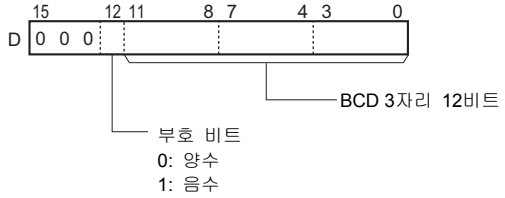
데이터 형식 지정 번호 C	변환 데이터 S
0	10진수 －9999999~－1 16진수 #FF676981~FFFFFFFF
	10진수 0~9999999 16진수 #00000000~0098967F
1	10진수 －79999999~－1 16진수 #FB3B4C01~FFFFFFFF
	10진수 0~79999999 16진수 #00000000~04C4B3FF
2	10진수 －9999999~－1 16진수 #FF676981~FFFFFFFF
	10진수 0~99999999 16진수 #00000000~05F5E0FF
3	10진수 －19999999~－1 16진수 #FECED301~FFFFFFFF
	10진수 0~99999999 16진수 #00000000~05F5E0FF

BCDS/BDSL

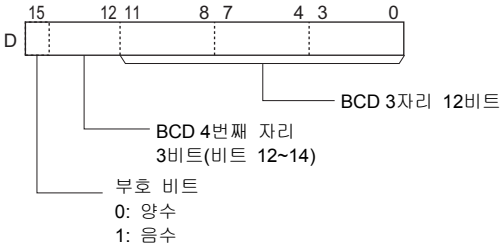
D: 변환 결과 출력 CH 번호

■ BCDS

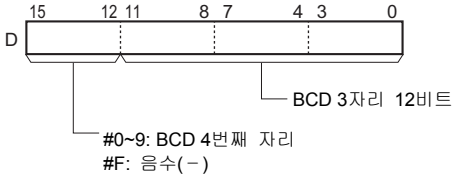
· C = 0일 때 D의 값: −999~999(BCD)



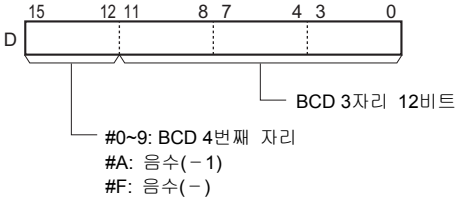
· C = 1일 때 D의 값: −7999~7999(BCD)



· C = 2일 때 D의 값: −999~9999(BCD)

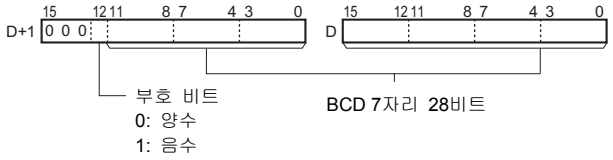


· C = 3일 때 D의 값: −1999~9999(BCD)

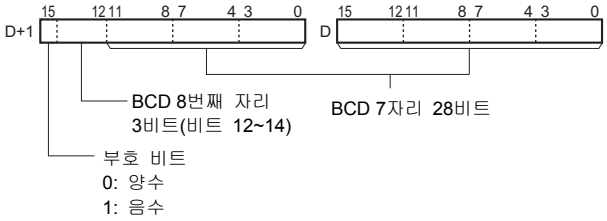


■ BDSL

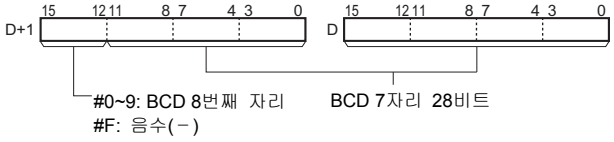
· C = 0일 때
D의 값: −9999999~9999999(BCD)



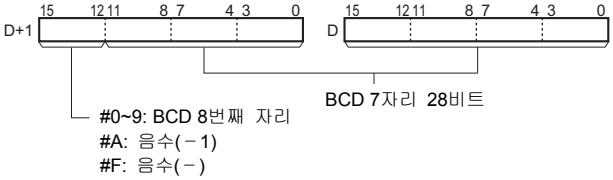
· C = 1일 때
D의 값: −79999999~79999999(BCD)



· C = 2일 때
D의 값: −9999999~99999999(BCD)



· C = 3일 때
D의 값: −19999999~99999999(BCD)



■ 영역 종류

· BCDS

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	○	−	○	−	−	−	−
S											−							
D																		

· BDSL

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	○	−	○	−	−	−	−
S											−	−						
D																		

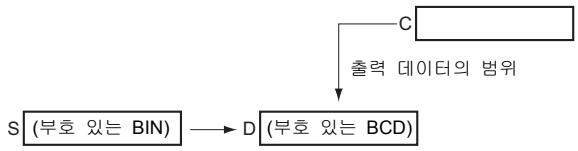
조건 플래그의 동작

이름	라벨	내용	
		BCDS	BDSL
에러 플래그	P_ER	· 변환 데이터가 16진수 #0000~03E7 또는 16진수 #FC19~FFFF의 범위에 없을 때 ON(C에 0 지정 시) · 변환 데이터가 16진수 #0000~1F3F 또는 16진수 #E0C1~FFFF의 범위에 없을 때 ON(C에 1 지정 시) · 변환 데이터가 16진수 #0000~270F 또는 16진수 #FC19~FFFF의 범위에 없을 때 ON(C에 2 지정 시) · 변환 데이터가 16진수 #0000~270F 또는 16진수 #F831~FFFF의 범위에 없을 때 ON(C에 3 지정 시) · C의 데이터가 0~3의 범위에 없을 때 ON · 그 이외일 때 OFF	· 변환 데이터가 16진수 #00000000~0098967F 또는 16진수 #FF676981~FFFFFFFF의 범위에 없을 때 ON(C에 0 지정 시) · 변환 데이터가 16진수 #00000000~04C4B3FF 또는 16진수 #FB3B4C01~FFFFFFFF의 범위에 없을 때 ON(C에 1 지정 시) · 변환 데이터가 16진수 #00000000~05F5E0FF 또는 16진수 #FF676981~FFFFFFFF의 범위에 없을 때 ON(C에 2 지정 시) · 변환 데이터가 16진수 #00000000~05F5E0FF 또는 16진수 #FECED301~FFFFFFFF의 범위에 없을 때 ON(C에 3 지정 시) · C의 데이터가 0~3의 범위에 없을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과가 0일 때 ON · 그 이외일 때 OFF	
부정 플래그	P_N	· 결과의 부호 비트가 1일 때 ON(S에 0, 1 지정 시) · 결과의 최상위 자리가 16진수 #F일 때 ON(S에 2 지정 시) · 그 이외일 때 OFF	

기능

■ BCDS

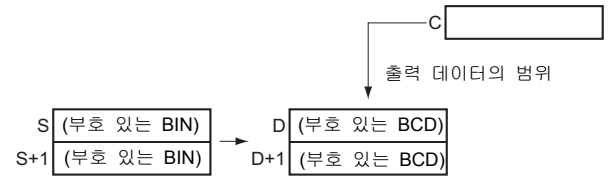
S로 지정한 부호 있는 BIN 데이터(16비트)를 C로 지정하는 형식의 부호 있는 BCD 데이터로 변환하여 결과를 D에 출력합니다.



데이터 형식 지정 번호	변환 데이터(BIN)	변환 결과 데이터(BCD)
C	S	D
0	10진수 －999~－1 16진수 #FC19~FFFF	－999~－1
	10진수 0~+999 16진수 #0000~03E7	0~999
1	10진수 －7999~－1 16진수 #E0C1~FFFF	－7999~－1
	10진수 0~+7999 16진수 #0000~1F3F	0~7999
2	10진수 －999~－1 16진수 #FC19~FFFF	－999~－1
	10진수 0~+9999 16진수 #0000~270F	0~9999
3	10진수 －1999~－1 16진수 #F831~FFFF	－1999~－1
	10진수 0~+9999 16진수 #0000~270F	0~9999

■ BDSL

S+1, S로 지정한 부호 있는 BIN 데이터(32비트)를 C로 지정한 형식의 부호 있는 BCD 데이터로 변환하여 결과를 D+1, D에 출력합니다.



데이터 형식 지정 번호	변환 데이터(BIN 16진수)	변환 결과 데이터(BCD)
C	S	D
0	10진수 −9999999~−1 16진수 #FF676981~FFFFFFFF	−9999999~−1
	10진수 0~9999999 16진수 #00000000~0098967F	0~9999999
1	10진수 −79999999~−1 16진수 #FB3B4C01~FFFFFFFF	−79999999~−1
	10진수 0~79999999 16진수 #00000000~04C4B3FF	0~79999999
2	10진수 −9999999~−1 16진수 #FF676981~FFFFFFFF	−9999999~−1
	10진수 0~99999999 16진수 #00000000~05F5E0FF	0~99999999
3	10진수 −19999999~−1 16진수 #FECED301~FFFFFFFF	−19999999~−1
	10진수 0~99999999 16진수 #00000000~05F5E0FF	0~99999999

사용상의 주의 사항

· 부호 있는 BCD 데이터에서 −0은 +0으로 취급합니다.

힌트

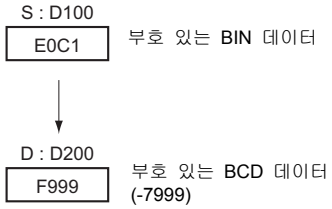
· 고기능 I/O 유니트 중에는 부호 있는 BCD 데이터를 입력 데이터로 하는 경우가 있습니다.
이 경우, BCDS/BDSL 명령을 사용하여 부호 있는 BIN 데이터로 변환하면 연산이 편리해집니다.

샘플 프로그램

■ BCDS

0.00이 ON이 되면 먼저 D100의 데이터 범위가 데이터 형식 지정 0001에 기초하는지 확인합니다. 확인 결과가 정상이므로 데이터 메모리 D100의 부호 있는 BIN 데이터가 부호 있는 BCD 데이터로 변환되어 결과가 D200에 출력됩니다.

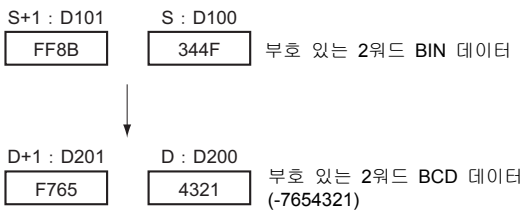




■ BDSL

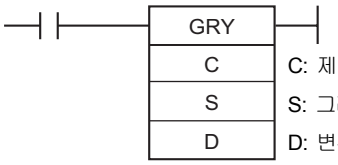
0.00이 ON이 되면 먼저 D101, D100의 데이터 범위가 데이터 형식 지정 0003에 기초하는지 확인합니다. 확인 결과가 정상이므로 데이터 메모리 D101, D100의 부호 있는 2워드 BIN 데이터가 부호 있는 2워드 BCD 데이터로 변환되어 결과가 D201, D200에 출력됩니다.





GRY

명령 이름	니모닉	동작 옵션	평션 번호	기능
그레이 코드 변환	GRY	@GRY	474	지정된 CH 내의 그레이 2진수 코드 데이터를 지정된 분해능에 따라 BIN 데이터, BCD 데이터 또는 각도(°) 데이터 중 하나로 변환합니다.

심볼	GRY
	

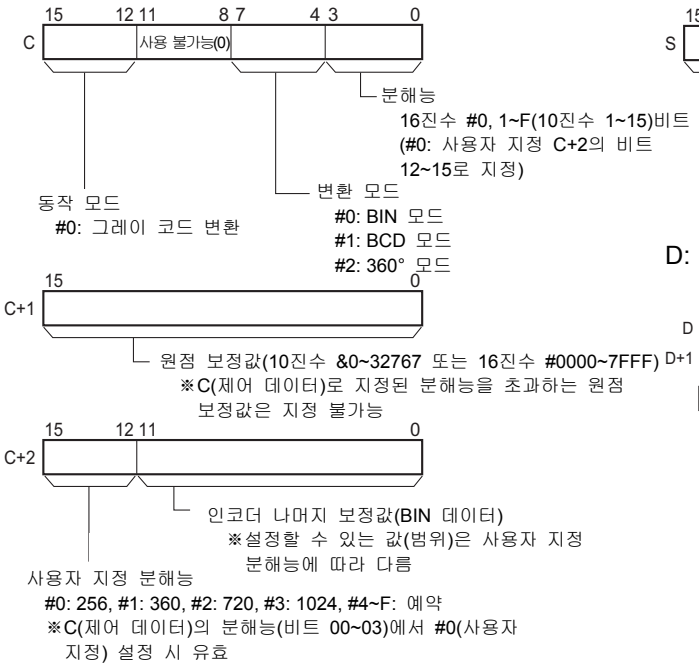
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

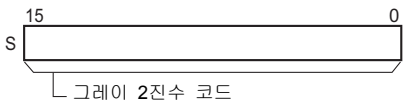
피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	–	3
S	그레이 2진수 코드 데이터	–	1
D	변환 결과 출력 하위 CH 번호	–	2

C: 제어 데이터

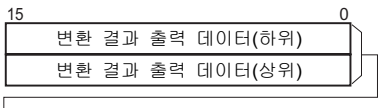


S: 그레이 2진수 코드



그레이 2진수 코드
C(제어 데이터)의 분해능(비트 00~03)에서 지정된 비트 수만큼 입력 범위로 합니다.
※지정된 분해능 이외(입력 범위 밖)의 비트는 무시됩니다.
(예: 분해능 #8의 경우, S가 #FFFF일 때 #00FF로 간주합니다.)

D: 변환 결과 출력 하위 CH 번호



C(제어 데이터)의 분해능(비트 00~03) 및 변환 모드(비트 04~07) 지정에 따라 변환된 데이터를 출력 시작 CH에서 상위 CH 측으로, 하위→상위 데이터 순서로 저장합니다.
BIN 모드 시: 10진수 &0~32767 또는 16진수 00000000~ # 00007FFF
BCD 모드 시: #00000000~#00032767
360° 모드 시: #00000000~#00003599(0.0~359.9°, 0.1° 단위 BCD 데이터)

GRY

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S											○	○						
D											–	–						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C의 동작 모드 지정(비트 12~15)에서 0(그레이 코드 변환) 이외가 지정되었을 때 ON · C+1의 원점 보정값이 분해능(사용자 지정 시도 포함) 이상의 값이 설정되었을 때 ON · C의 변환 모드 지정(비트 04~07)에서 0(BIN), 1(BCD), 2(360°) 이외가 지정되었을 때 ON · C의 분해능(비트 00~03)에서 0(사용자 지정)을 설정한 경우, 사용자 지정 분해능을 초과하는 인코더 나머지 보정값이 지정되었을 때 ON · C의 분해능(비트 00~03)에서 0(사용자 지정)을 설정한 경우, BIN 데이터로 변환한 결과<인코더 나머지 보정값일 때 ON · C의 분해능(비트 00~03)에서 0(사용자 지정)을 설정한 경우, BIN 데이터로 변환한 결과<분해능일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	OFF
부정 플래그	P_N	OFF

기능

S로 지정한 CH 내의 그레이 2진수 코드 데이터를 C로 설정한 분해능 및 다음의 변환 모드(BIN, BCD, 360°)에 따라 변환하여 결과를 D에 출력합니다.

변환 모드	내용
BIN 모드	그레이 2진수 코드를 BIN 데이터(16진수 #00000000~00007FFF)로 변환한 값을 원점 및 나머지 보정하여 D에 출력합니다.
BCD 모드	그레이 2진수 코드를 BIN 데이터로 변환한 값을 원점 및 나머지 보정한 후, 다시 BCD 데이터(BCD #00000000~00032767)로 변환하여 D에 출력합니다.
360° 모드	그레이 2진수 코드를 BIN 데이터로 변환한 값을 원점 및 나머지 보정한 후, 다시 각도 데이터 (#00000000~00003599: 0.1° 단위 0.0~359.9°에 대응)로 변환하여 D에 출력합니다.

힌트

· 일반적으로 그레이 2진수 코드 출력 타입의 절대값 인코더에서 병렬 신호(2n)를 DC 입력 유니트로 전송할 때 사용합니다.

주: S로 지정한 CH가 입력 유니트에 할당된 CH인 경우 변환 대상인 입력 데이터는 최대 1CPU 유니트 사이클 이전의 그레이 코드 값이 되므로 주의하십시오.

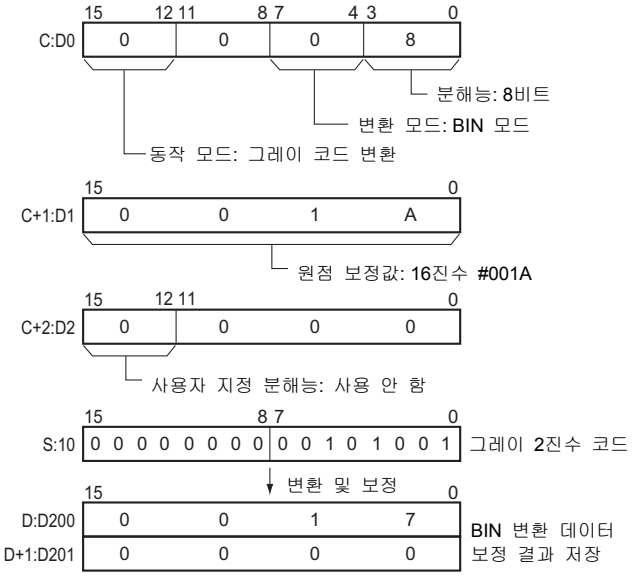
샘플 프로그램

0.00이 ON일 때 10CH에 저장된 그레이 2진수 코드의 데이터가 제어 데이터(D0)의 내용에 따라 변환되어 결과가 D200에 출력됩니다.



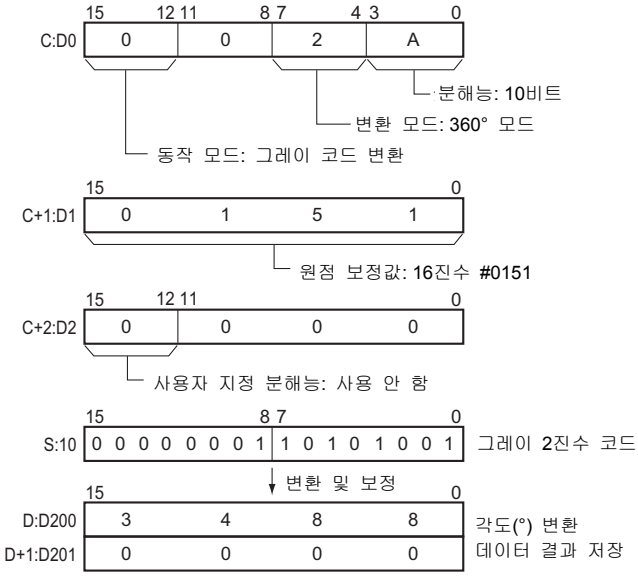
(예 1)

8비트 분해능에서 BIN 데이터로 변환, 원점 보정 16진수 #001A의 경우



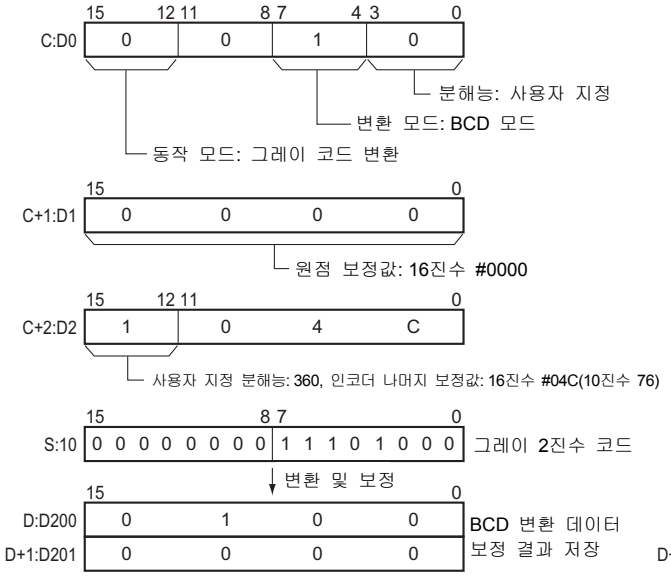
(예 2)

10비트 분해능에서 각도(°) 데이터로 변환, 원점 보정 16진수 #0151의 경우



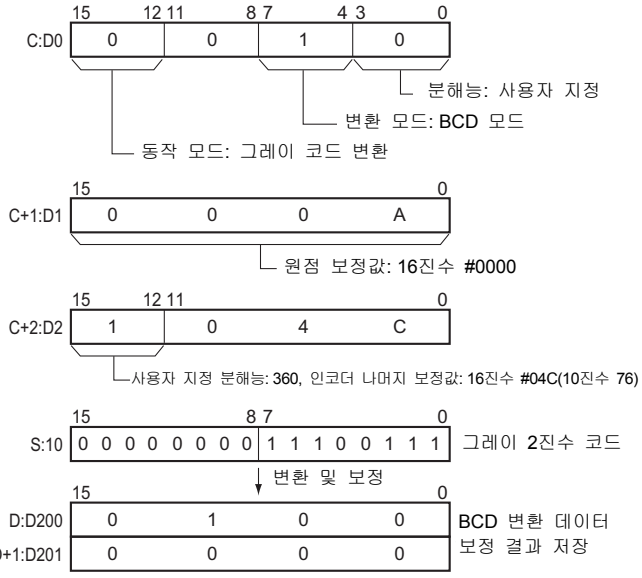
(예 3)

오므론이 제작한 절대값 인코더 E6C2-AG5C를 사용할 때(분해능 360/회전, 인코더 나머지 보정값 76) BCD 데이터로 변환, 원점 보정값 16진수 #0000의 경우



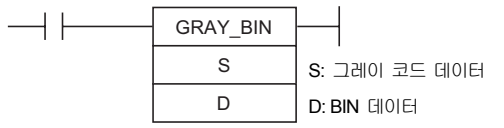
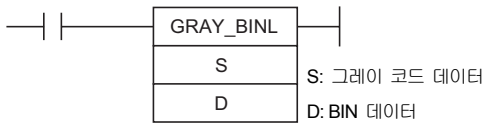
(예 4)

오므론이 제작한 절대값 인코더 E6C2-AG5C를 사용할 때(분해능 360/회전, 인코더 나머지 보정값 76) BCD 데이터로 변환, 원점 보정값 16진수 #000A의 경우



GRAY_BIN/GRAY_BINL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
그레이 코드→BIN 변환	GRAY_BIN	@GRAY_BIN	478	지정한 그레이 코드(1W)를 BIN 코드(1W)로 변환하여 출력합니다.
그레이 코드→BIN 2워드 변환	GRAY_BINL	@GRAY_BINL	479	지정한 그레이 코드(2W)를 BIN 코드(2W)로 변환하여 출력합니다.

심볼	GRAY_BIN	GRAY_BINL
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		GRAY_BIN	GRAY_BINL	GRAY_BIN	GRAY_BINL
S	그레이 코드 데이터	WORD	DWORD	1	2
D	BIN 데이터	WORD	DWORD	1	2

■ 영역 종류

내용	채널 주소							간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
D										–							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 변환 결과가 음수일 때(그레이 코드 데이터가 #8000~#FFFF일 때) ON · 그 이외일 때 OFF
= 플래그	P_EQ	변하지 않음
부정 플래그	P_N	변하지 않음

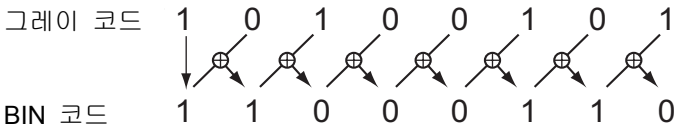
기능

■ GRAY-BIN

S로 지정한 그레이 코드(1W)를 BIN 코드(1W)로 변환하여 D에 출력합니다.

■ GRAY-BINL

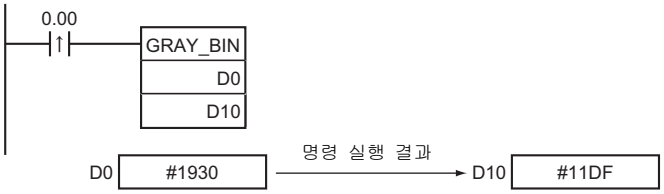
S로 지정한 그레이 코드(2W)를 BIN 코드(2W)로 변환하여 D에 출력합니다.



샘플 프로그램

■ GRAY-BIN

접점 0.00을 ON하면 D0에 저장된 그레이 코드를 바이너리 코드로 변환하여 D10에 저장합니다.



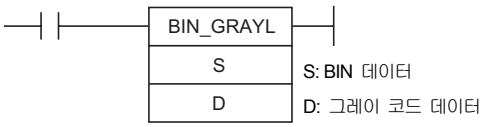
■ GRAY-BINL

접점 0.00을 ON하면 D1~D0에 저장된 그레이 코드를 바이너리 코드로 변환하여 D11~D10에 저장합니다.



BIN_GRAY/BIN_GRAYL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
BIN→그레이 코드 변환	BIN_GRAY	@BIN_GRAY	480	지정한 BIN 코드(1W)를 그레이 코드(1W)로 변환하여 출력합니다.
BIN→그레이 코드 2워드 변환	BIN_GRAYL	@BIN_GRAYL	481	지정한 BIN 코드(2W)를 그레이 코드(2W)로 변환하여 출력합니다.

심볼	BIN_GRAY	BIN_GRAYL
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		BIN_GRAY	BIN_GRAYL	BIN_GRAY	BIN_GRAYL
S	BIN 데이터	WORD	DWORD	1	2
D	그레이 코드 데이터	WORD	DWORD	1	2

■ 영역 종류

내용	채널 주소							간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
D										–							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· BIN 코드가 음수일 때(그레이 코드 데이터가 #8000~#FFFF일 때) ON · 그 이외일 때 OFF
= 플래그	P_EQ	변하지 않음
부정 플래그	P_N	변하지 않음

기능

■ BIN_GRAY

S로 지정한 BIN 코드(1W)를 그레이 코드(1W)로 변환하여 D에 출력합니다.

■ BIN_GRAYL

S로 지정한 BIN 코드(2W)를 그레이 코드(2W)로 변환하여 D에 출력합니다.

샘플 프로그램

■ BIN_GRAY

접점 0.01을 ON하면 D10에 저장된 바이너리 코드를 그레이 코드로 변환하여 D0에 저장합니다.



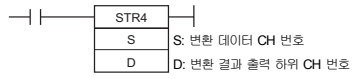
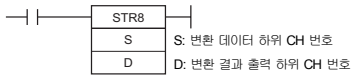
■ BIN_GRAYL

접점 0.01을 ON하면 D11-D10에 저장된 바이너리 코드를 그레이 코드로 변환하여 D1-D0에 저장합니다.



STR4/STR8/STR16

명령 이름	니모닉	동작 옵션	평션 번호	기능
숫자 4자리→ ASCII 데이터 변환	STR4	@STR4	601	숫자 데이터(16진수 4자리)를 ASCII 데이터(4문자)로 변환합니다.
숫자 8자리→ASCII 데이터 변환	STR8	@STR8	602	숫자 데이터(16진수 8자리)를 ASCII 데이터(8문자)로 변환합니다.
숫자 16자리→ ASCII 데이터 변환	STR16	@STR16	603	숫자 데이터(16진수 16자리)를 ASCII 데이터(16문자)로 변환합니다.

심볼	STR4	STR8	STR16
			

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식			크기		
		STR4	STR8	STR16	STR4	STR8	STR16
S	변환 데이터 CH 번호	UINT	UDINT	ULINT	1	2	4
D	변환 결과 출력 하위 CH 번호	WORD	WORD	WORD	2	4	8

■ 영역 종류

· STR4, STR8

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
D											–							

· STR16

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		

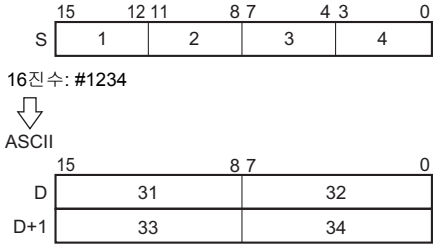
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 변환 결과가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 변환 데이터의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

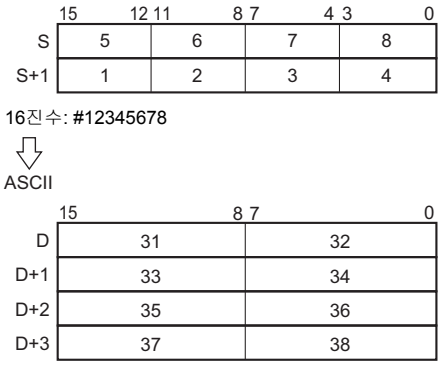
■ STR4

S로 지정한 숫자 데이터(16진수 4자리 #0000~FFFF)를 ASCII 데이터(4문자)로 변환하여 결과를 D, D＋1에 출력합니다.



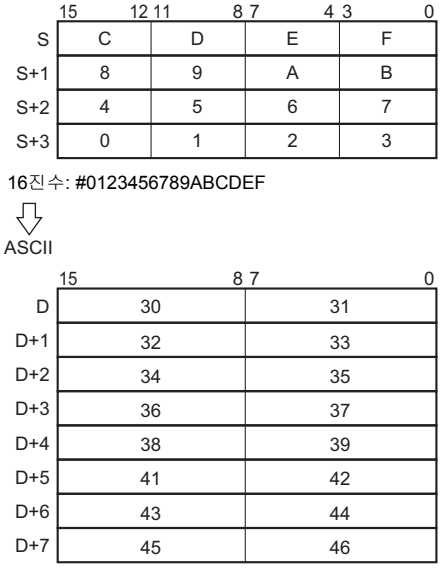
■ STR8

S로 지정한 숫자 데이터(16진수 8자리 #00000000~FFFFFFFF)를 ASCII 데이터(8문자)로 변환하여 결과를 D, D＋1, D+2, D+3에 출력합니다.



■ STR16

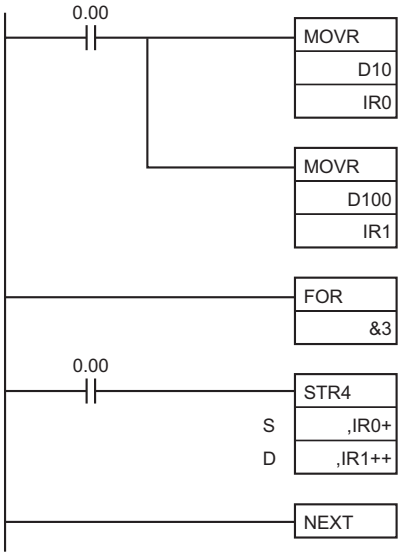
S로 지정한 숫자 데이터(16진수 16자리 #0000000000000000~FFFFFFFFFFFFFFFF)를 ASCII 데이터(16문자)로 변환하여 결과를 D, D＋1, D+2, D+3, D+4, D+5, D+6, D+7에 출력합니다.



샘플 프로그램

■ 3채널분의 숫자 데이터를 ASCII 데이터로 변환하는 경우

0.00이 ON되면 D10부터 3채널분의 숫자 데이터를 1채널마다 ASCII 데이터로 변환합니다. 변환된 각 ASCII 데이터는 D100 이후에 저장됩니다.



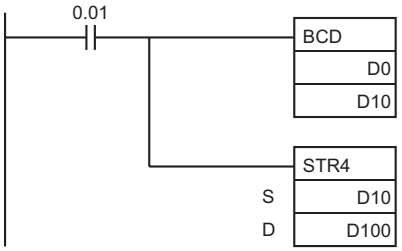
	15 12	11 8	7 4	3 0
S:D10	0	1	2	3
S+1:D11	4	5	6	7
S+2:D12	8	9	A	B

16진수 ⇩ ASCII

	15 8	7 0
D:D100	30	31
D+1:D101	32	33
D+2:D102	34	35
D+3:D103	36	37
D+4:D104	38	39
D+5:D105	41	42

■ 숫자 데이터(10진수)를 BCD 데이터 형식으로 하여 ASCII 데이터로 변환하는 경우

0.01이 ON되면 변환 데이터가 되는 D0의 값(예: 10진수 &1234)을 BCD 데이터로 변환하여 D10에 일단 저장합니다. 그 후 ASCII 데이터로 변환하여 결과를 D100에 출력합니다.



	15 12	11 8	7 4	3 0	
D0	0	4	D	2	10진수 &1234(16진수 #04D2)

BIN(16진수) ⇩ BCD

	15 12	11 8	7 4	3 0
S:D10	1	2	3	4

BCD ⇩ ASCII(BCD 형식)

	15 8	7 0
D:D100	31	32
D+1:D101	33	34

NUM4/NUM8/NUM16

명령 이름	니모닉	동작 옵션	펑션 번호	기능
ASCII 데이터→ 숫자 4자리 변환	NUM4	@NUM4	604	ASCII 데이터(4문자)를 숫자 데이터(16진수 4자리)로 변환합니다.
ASCII 데이터→ 숫자 8자리 변환	NUM8	@NUM8	605	ASCII 데이터(8문자)를 숫자 데이터(16진수 8자리)로 변환합니다.
ASCII 데이터→ 숫자 16자리 변환	NUM16	@NUM16	606	ASCII 데이터(16문자)를 숫자 데이터(16진수 16자리)로 변환합니다.

심볼	NUM4	NUM8	NUM16
	NUM4 S D S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	NUM8 S D S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	NUM16 S D S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식			크기		
		NUM4	NUM8	NUM16	NUM4	NUM8	NUM16
S	변환 데이터 하위 CH 번호	WORD	WORD	WORD	2	4	8
D	변환 결과 출력 CH 번호	UINT	UDINT	ULINT	1	2	4

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 변환 데이터가 16진수 숫자 ('0'-'9', 'a'-'f', 'A'-'F') 이외의 문자를 포함할 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 변환 데이터의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

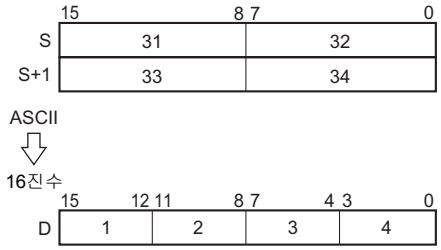
NUM4/NUM8/NUM16

기능

■ NUM4

· S로 지정한 ASCII 데이터(4문자)를 숫자 데이터(16진수 4자리)로 변환하여 결과를 D에 출력합니다.

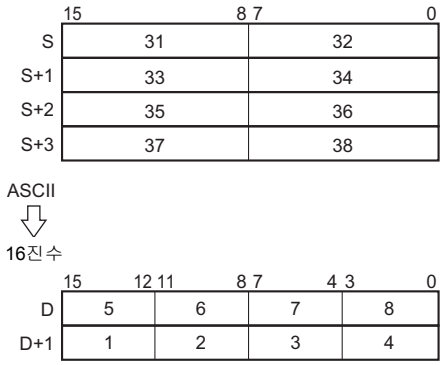
· ASCII 데이터가 16진수 숫자 이외의 문자를 포함하는 경우는 에러 플래그가 ON이 됩니다. 이 경우, 명령은 실행되지 않습니다.



■ NUM8

· S로 지정한 ASCII 데이터(8문자)를 숫자 데이터(16진수 8자리)로 변환하여 결과를 D, D+1에 출력합니다.

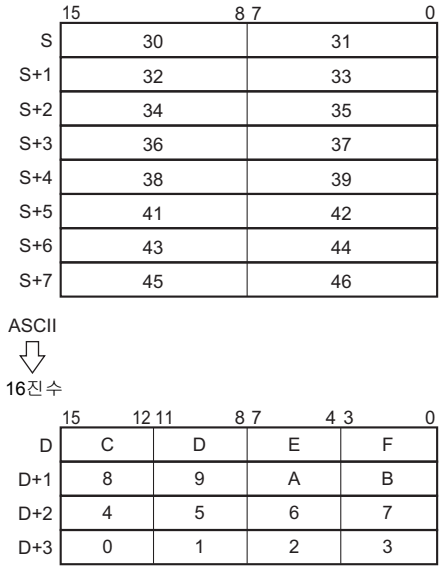
· ASCII 데이터가 16진수 숫자 이외의 문자를 포함하는 경우는 에러 플래그가 ON이 됩니다. 이 경우, 명령은 실행되지 않습니다.



■ NUM16

· S로 지정한 ASCII 데이터(16문자)를 숫자 데이터(16진수 16자리)로 변환하여 결과를 D, D+1, D+2, D+3에 출력합니다.

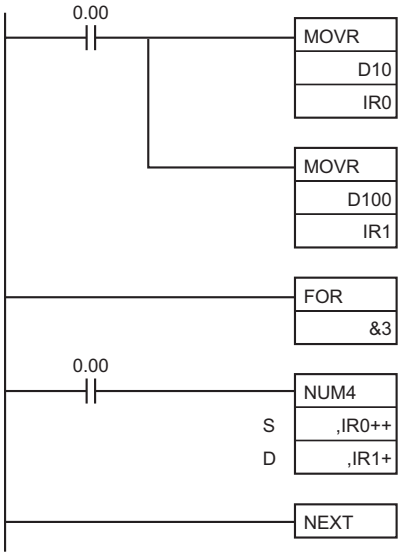
· ASCII 데이터가 16진수 숫자 이외의 문자를 포함하는 경우는 에러 플래그가 ON이 됩니다. 이 경우, 명령은 실행되지 않습니다.



샘플 프로그램

■ 4자리 숫자를 나타내는 ASCII 데이터 3개를 숫자 데이터로 변환하는 경우

0.00이 ON되면 D10부터 6채널분의 ASCII 데이터를 2채널마다 숫자 데이터로 변환합니다.
변환된 각 숫자 데이터는 D100 이후에 저장됩니다.



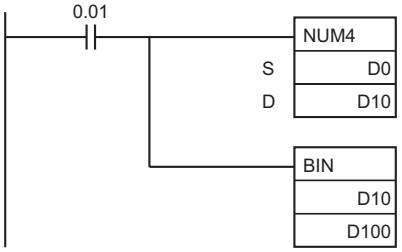
	15 8	7 0
S:D10	31	32
S+1:D11	41	42
S+2:D12	38	39
S+3:D13	45	46
S+4:D14	30	30
S+5:D15	30	30

ASCII ⇩ 16진수

	15 12	11 8	7 4	3 0
D:D100	1	2	A	B
D+1:D101	8	9	E	F
D+2:D102	0	0	0	0

■ BCD 형식의 ASCII 데이터를 숫자 데이터(10진수)로 변환하는 경우

0.01이 ON되면 D0을 하위 CH로 하는 ASCII 데이터를 BCD 데이터로 변환하여 D10에 일단 저장합니다.
그 후 BIN 데이터로 변환하여 결과를 D100에 출력합니다.



	15 8	7 0
S:D0	31	32
S+1:D1	33	34

ASCII(BCD 형식) ⇩ BCD

	15 12	11 8	7 4	3 0
D:10	1	2	3	4

BCD ⇩ BIN(16진수)

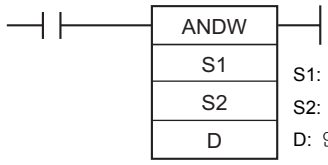
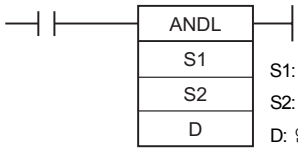
	15 12	11 8	7 4	3 0	
D:100	0	4	D	2	10진수 &1234(16진수 #04D2)

논리 연산 명령

니모닉	명령 이름	펑션 번호	페이지
ANDW	워드 논리곱	034	3-310
ANDL	워드 논리곱(2워드)	610	
ORW	워드 논리합	035	3-312
ORWL	워드 논리합(2워드)	611	
XORW	워드 배타적 논리합	036	3-314
XORL	워드 배타적 논리합(2워드)	612	
XNRW	워드 배타적 논리합 부정	037	3-316
XNRL	워드 배타적 논리합 부정(2워드)	613	
COM	비트 반전	029	3-318
COML	비트 2워드 반전	614	

ANDW/ANDL

명령 이름	니모닉	동작 옵션	평션 번호	기능
워드 논리곱	ANDW	@ANDW	034	CH 데이터 단위에서 CH 데이터와 CH 데이터, 상수의 논리곱을 계산합니다.
워드 논리곱(2워드)	ANDL	@ANDL	610	2CH분의 CH 데이터와 CH 데이터, 상수의 논리곱을 계산합니다.

심볼	ANDW	ANDL
	 S1: 연산 데이터 1 S2: 연산 데이터 2 D: 연산 결과 출력 CH 번호	 S1: 연산 데이터 1 하위 CH 번호 S2: 연산 데이터 2 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		ANDW	ANDL	ANDW	ANDL
S1	ANDW: 연산 데이터 1 ANDL: 연산 데이터 1 하위 CH 번호	WORD	DWORD	1	2
S2	ANDW: 연산 데이터 2 ANDL: 연산 데이터 2 하위 CH 번호	WORD	DWORD	1	2
D	연산 결과 출력 CH 번호	WORD	DWORD	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
ANDW	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
	D											–							
ANDL	S1,S2	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
	D											–							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 연산 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

■ ANDW

S1로 지정한 데이터와 S2로 지정한 데이터의 논리곱을 계산하여 결과를 D에 출력합니다.

S1 · S2 → D

S1	S2	D
1	1	1
1	0	0
0	1	0
0	0	0

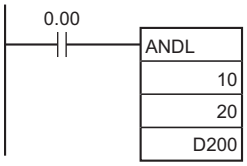
■ ANDL

S1로 지정한 데이터와 S2로 지정한 데이터를 2워드 데이터로 논리곱을 계산하여 결과를 D+1, D에 출력합니다.

(S1+1, S1) · (S2+1, S2) → (D+1, D)

S1+1, S1	S2+1, S2	D+1, D
1	1	1
1	0	0
0	1	0
0	0	0

샘플 프로그램



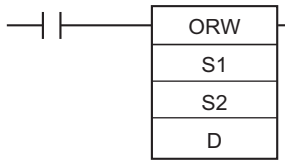
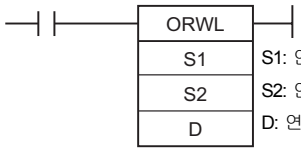
0.00이 ON이면 11, 10CH와 21, 20CH의 대응 비트별로 논리곱을 계산하여 데이터 메모리 D201, D200의 대응 비트에 출력합니다.

S1 : 10CH S1+1 : 11CH			S2 : 20CH S2+1 : 21CH			D : D200 D+1 : D201	
10.00	0	∧	20.00	1	⇨	00	0
10.01	1		20.01	1		01	1
10.02	0		20.02	0		02	0
10.03	1		20.03	0		03	0
10.04	0		20.04	1		04	0
11.13	1		21.13	1		13	1
11.14	1		21.14	0		14	0
11.15	0		21.15	0		15	0

주: ∧ 기호는 논리곱을 나타냅니다.

ORW/ORWL

명령 이름	니모닉	동작 옵션	평션 번호	기능
워드 논리합	ORW	@ORW	035	CH 데이터 단위에서 CH 데이터와 CH 데이터, 상수의 논리합을 계산합니다.
워드 논리합(2워드)	ORWL	@ORWL	611	2CH분의 CH 데이터와 CH 데이터, 상수의 논리합을 계산합니다.

심볼	ORW	ORWL
	 S1: 연산 데이터 1 S2: 연산 데이터 2 D: 연산 결과 출력 CH 번호	 S1: 연산 데이터 1 하위 CH 번호 S2: 연산 데이터 2 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		ORW	ORWL	ORW	ORWL
S1	ORW: 연산 데이터 1 ORWL: 연산 데이터 1 하위 CH 번호	WORD	DWORD	1	2
S2	ORW: 연산 데이터 2 ORWL: 연산 데이터 2 하위 CH 번호	WORD	DWORD	1	2
D	연산 결과 출력 CH 번호	WORD	DWORD	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
ORW	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
	D											–							
ORWL	S1,S2	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
	D											–							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 연산 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

■ ORW

S1로 지정한 데이터와 S2로 지정한 데이터의 논리합을 계산하여 결과를 D에 출력합니다.

S1+S2 → D

S1	S2	D
1	1	1
1	0	1
0	1	1
0	0	0

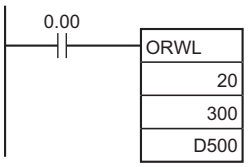
■ ORWL

S1로 지정한 데이터와 S2로 지정한 데이터를 2워드 데이터로 논리합을 계산하여 결과를 D+1, D에 출력합니다.

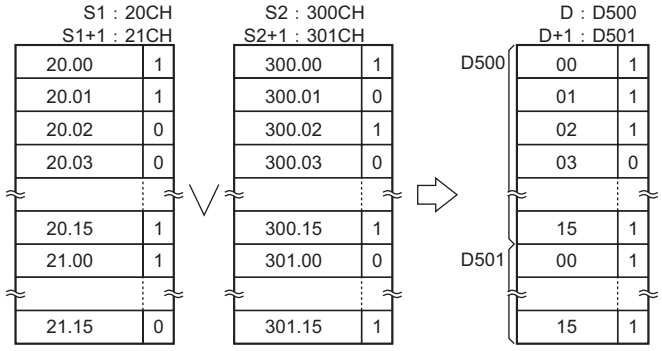
(S1+1, S1)+(S2+1, S2) → (D+1, D)

S1+1, S1	S2+1, S2	D+1, D
1	1	1
1	0	1
0	1	1
0	0	0

샘플 프로그램



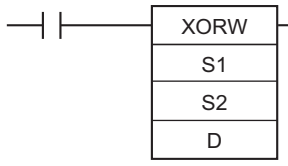
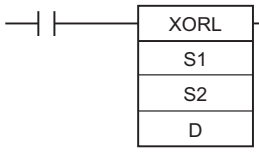
0.00이 ON이면 21, 20CH와 301, 300CH의 대응 비트별로 논리합을 계산하여 데이터 메모리 D501, D500의 대응 비트에 출력합니다.



주: ∨ 기호는 논리합을 나타냅니다.

XORW/XORL

명령 이름	니모닉	동작 옵션	평션 번호	기능
워드 배타적 논리합	XORW	@XORW	036	CH 데이터 단위에서 CH 데이터와 CH 데이터, 상수의 배타적 논리합을 계산합니다.
워드 배타적 논리합 (2워드)	XORL	@XORL	612	2CH의 CH 데이터에 대해 CH 데이터, 상수의 배타적 논리합을 계산합니다.

심볼	XORW	XORL
	 S1: 연산 데이터 1 S2: 연산 데이터 2 D: 연산 결과 출력 CH 번호	 S1: 연산 데이터 1 하위 CH 번호 S2: 연산 데이터 2 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		XORW	XORL	XORW	XORL
S1	XORW: 연산 데이터 1 XORL: 연산 데이터 1 하위 CH 번호	WORD	DWORD	1	2
S2	XORW: 연산 데이터 2 XORL: 연산 데이터 2 하위 CH 번호	WORD	DWORD	1	2
D	XORW: 연산 결과 출력 CH 번호 XORL: 연산 결과 출력 하위 CH 번호	WORD	DWORD	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
XORW	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
	D											–							
XORL	S1,S2	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
	D											–							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 연산 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

■ XORW

S1로 지정한 데이터와 S2로 지정한 데이터의 배타적 논리합을 계산하여 결과를 D에 출력합니다.

$S1 \cdot \overline{S2}+\overline{S1} \cdot S2 \rightarrow D$

S1	S2	D
1	1	0
1	0	1
0	1	1
0	0	0

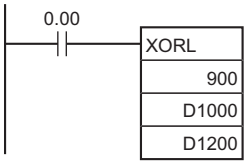
■ XORL

S1로 지정한 데이터와 S2로 지정한 데이터를 2워드 데이터로 배타적 논리합을 계산하여 결과를 D+1, D에 출력합니다.

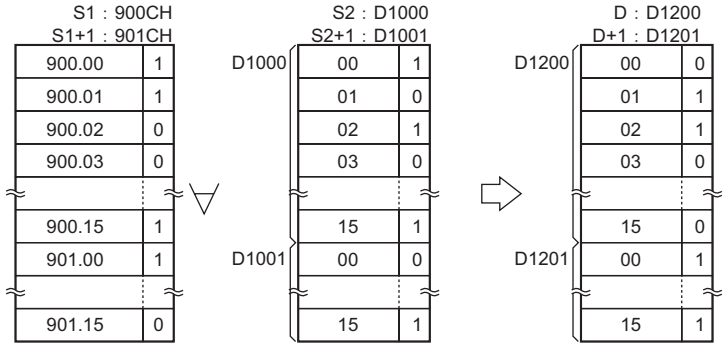
$(S1+1, S1)\cdot\overline{(S2+1, S2)}+\overline{(S1+1, S1)}\cdot(S2+1, S2) \rightarrow (D+1, D)$

S1+1, S1	S2+1, S2	D+1, D
1	1	0
1	0	1
0	1	1
0	0	0

샘플 프로그램



0.00이 ON이면 901, 900CH와 데이터 메모리 D1001, D1000의 대응 비트의 배타적 논리합을 계산하여 데이터 메모리 D1201, D1200의 대응 비트에 출력합니다.



주: ∀ 기호는 배타적 논리합을 나타냅니다.

XNRW/XNRL

명령 이름	니모닉	동작 옵션	평션 번호	기능
워드 배타적 논리합 부정	XNRW	@XNRW	037	CH 데이터 단위에서 CH 데이터와 CH 데이터, 상수의 배타적 논리합 부정을 계산합니다.
워드 배타적 논리합 부정(2워드)	XNRL	@XNRL	613	2CH분의 CH 데이터와 CH 데이터, 상수의 배타적 논리합을 계산합니다.

심볼	XNRW	XNRL
		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		XNRW	XNRL	XNRW	XNRL
S1	XNRW: 연산 데이터 1 XNRL: 연산 데이터 1 하위 CH 번호	WORD	DWORD	1	2
S2	XNRW: 연산 데이터 2 XNRL: 연산 데이터 2 하위 CH 번호	WORD	DWORD	1	2
D	XNRW: 연산 결과 출력 CH 번호 XNRL: 연산 결과 출력 하위 CH 번호	WORD	DWORD	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
XNRW	S1,S2	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
	D											–							
XNRL	S1,S2	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
	D											–							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 연산 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

■ XNRW

S1로 지정한 데이터와 S2로 지정한 데이터의 배타적 부정 논리합을 계산하여 결과를 D에 출력합니다.

$S1 \cdot S2+\overline{S1} \cdot \overline{S2} \rightarrow D$

S1	S2	D
1	1	1
1	0	0
0	1	0
0	0	1

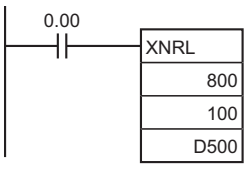
■ XNRL

S1로 지정한 데이터와 S2로 지정한 데이터를 2워드 데이터로 배타적 부정 논리합을 계산하여 결과를 D+1, D에 출력합니다.

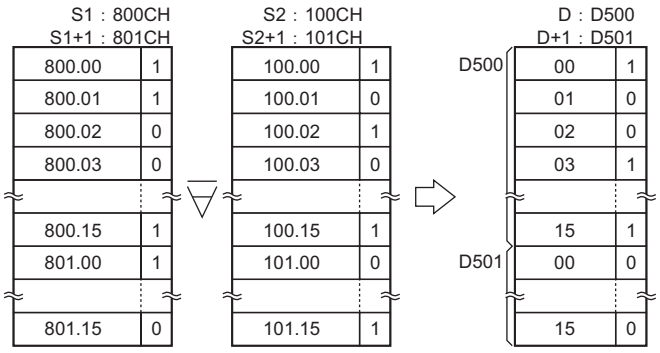
$(S1+1, S1)\cdot(S2+1, S2)+\overline{(S1+1, S1)}\cdot\overline{(S2+1, S2)} \rightarrow (D+1, D)$

S1+1, S1	S2+1, S2	D+1, D
1	1	1
1	0	0
0	1	0
0	0	1

샘플 프로그램



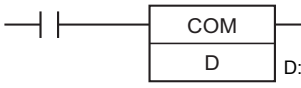
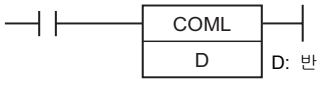
0.00이 ON이면 801, 800CH와 101, 100CH의 대응 비트의 배타적 부정 논리합을 계산하여 데이터 메모리 D501, D500의 대응 비트에 출력합니다.



주: $\overline{\forall}$ 기호는 배타적 부정 논리합을 나타냅니다.

COM/COML

명령 이름	니모닉	동작 옵션	평션 번호	기능
비트 반전	COM	@COM	029	CH 데이터를 비트 반전합니다.
비트 2워드 반전	COML	@COML	614	2CH분의 CH 데이터에서 각 비트를 반전 출력합니다.

심볼	COM	COML
	 D: 반전 데이터 CH 번호	 D: 반전 데이터 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		COM	COML	COM	COML
D	COM: 반전 데이터 CH 번호 COML: 반전 데이터 하위 CH 번호	WORD	DWORD	1	2

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
COM	D	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
COML	D												–						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 연산 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

■ COM

D로 지정한 데이터의 각 비트를 반전합니다.

$\overline{D} \rightarrow D$ $\begin{pmatrix} 1 \rightarrow 0 \\ 0 \rightarrow 1 \end{pmatrix}$

주: COM 명령을 사용한 경우, 입력 ON 시 1사이클마다 실행되므로 주의하십시오.

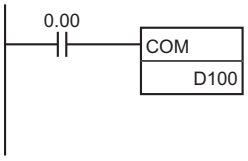
■ COML

D로 지정한 데이터를 2워드 데이터로 각 비트를 반전합니다.

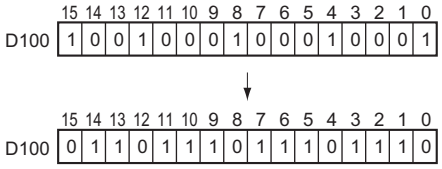
$\overline{(D+1,\ D)} \rightarrow (D+1,\ D)$

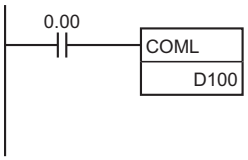
주: COML 명령을 사용한 경우, 입력 ON 시 1사이클마다 실행되므로 주의하십시오.

샘플 프로그램

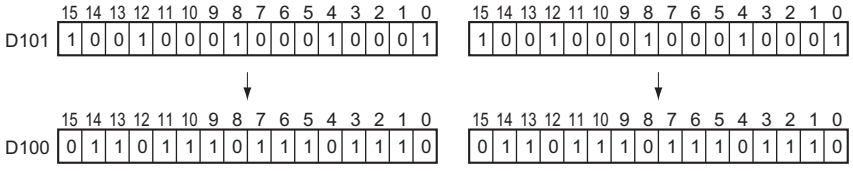


0.00이 ON이면 D100의 각 비트를 모두 반전합니다.





0.00이 ON이면 D101과 D100의 각 비트를 모두 반전합니다.

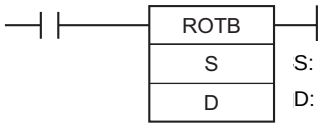


특수 연산 명령

니모닉	명령 이름	펑션 번호	페이지
ROTB	BIN 제곱근 연산	620	3-322
ROOT	BCD 제곱근 연산	072	3-324
APR	숫자 변환	069	3-326
FDIV	부동 소수점 제산(BCD)	079	3-335
BCNT	비트 카운터	067	3-338

ROTB

명령 이름	니모닉	동작 옵션	평션 번호	기능
BIN 제곱근 연산	ROTB	@ROTB	620	지정된 BIN 데이터(32비트)를 제곱근 연산하고 결과의 정수부를 출력합니다.

심볼	ROTB
	 S: 연산 데이터 하위 CH 번호 D: 연산 결과 출력 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	연산 데이터 하위 CH 번호	UDINT	2
D	연산 결과 출력 CH 번호	UINT	1

■ 영역 종류

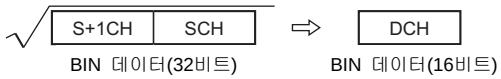
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
D											–	○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S, S+1의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· S, S+1의 데이터가 10진수 &1073741824~2147483647 또는 16진수 #40000000~7FFFFFFF의 범위 이내일 때 ON · 그 이외일 때 OFF
언더 플로 플래그	P_UF	OFF
부정 플래그	P_N	OFF

기능

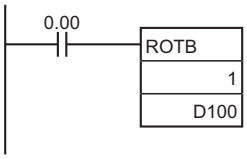
S로 지정한 BIN 데이터(32비트: 10진수 &0~1073741823 또는 16진수 #00000000~3FFFFFFF)를 제곱근 연산하여 결과의 정수부를 D에 출력합니다.



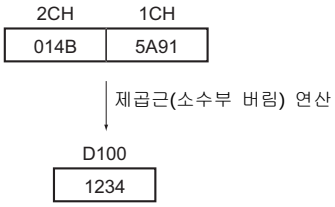
주:

- 소수점 이하 데이터는 버립니다.
- 입력 데이터(S+1, S)의 지정 범위는 10진수 &0~1073741823 또는 16진수 #00000000~3FFFFFFF입니다. 10진수 &1073741824~2147483647 또는 16진수 #40000000~7FFFFFFF가 지정된 경우는 1073741823으로 제곱근 연산을 수행합니다.
- 이 명령의 피연산자 S+1, S, D는 모두 BIN이므로 주의하십시오. 입력 데이터가 BCD인 경우는 ROOT를 사용하십시오.

샘플 프로그램

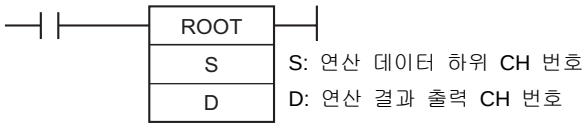


0.00이 ON이면 2CH, 1CH 데이터의 제곱근 연산을 수행하여 D100에 정수부만(소수부 버림) 출력합니다.



ROOT

명령 이름	니모닉	동작 옵션	평션 번호	기능
BCD 제곱근 연산	ROOT	@ROOT	072	지정된 데이터를 BCD의 2워드 데이터로 제곱근 연산하고 결과의 정수부를 BCD 데이터로 출력합니다.

심볼	ROOT
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	연산 데이터 하위 CH 번호	DWORD	2
D	연산 결과 출력 CH 번호	WORD	1

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
D											–	○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S, S+1의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF

기능

S로 지정된 데이터를 BCD의 2워드 데이터로 제곱근 연산하고 결과의 정수부를 BCD 데이터로 D에 출력합니다.

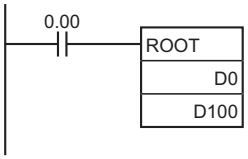


주:

· 소수점 이하 데이터는 버립니다.

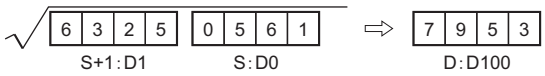
· 이 명령의 피연산자 S+1, S, D는 모두 BCD이므로 주의하십시오. 입력 데이터가 BIN인 경우는 ROTB를 사용하십시오.

샘플 프로그램



0.00이 ON이면 D1, D0 데이터의 제곱근을 계산하여 D100에 출력합니다.

(例) $\sqrt{63,250,561} \rightarrow 7,953$

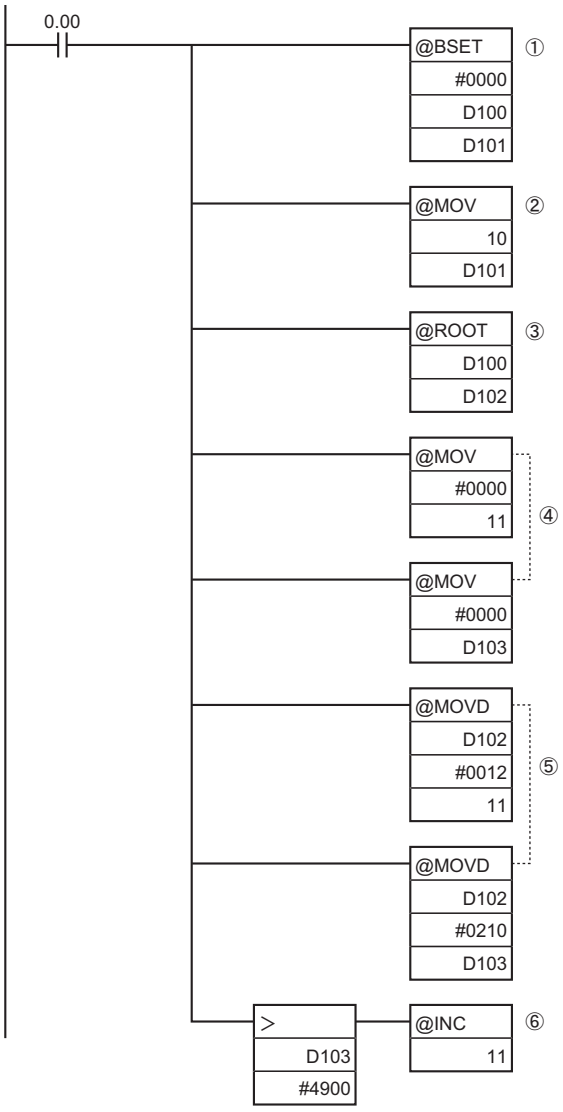


7953.221…
소수점 이하는 버립니다.

응용 예

10CH의 BCD 4자리 데이터의 제곱근을 계산하여 소수점 이하를 반올림한 후 11CH에 BCD 4자리로 출력합니다.

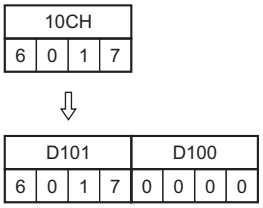
(例) $\sqrt{6017} = 77.56\cdots\cdots \rightarrow 78$
소수점 이하 반올림



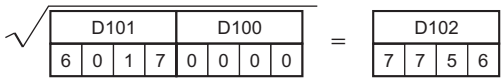
ROOT 명령은 변환 계산 데이터 8자리가 대상이 되므로 데이터 메모리 D101, D100을 사용합니다.

① D101, D100을 클리어(00000000)합니다.

② 내부 보조 릴레이 10CH의 내용을 D101에 전송합니다.



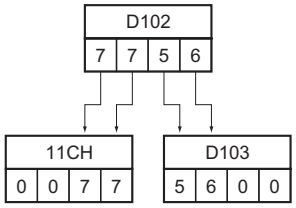
③ D101, 100의 8자리 제곱근을 계산하여 D102에 출력합니다.



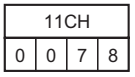
7756.932…
소수점 이하는 버립니다.

④ 11CH, D103을 클리어(0000)합니다.

⑤ D102의 내용을 2자리씩 11CH, D103에 전송합니다.



⑥ D103의 내용이 #4900보다 크면 11CH에 1이 가산됩니다.

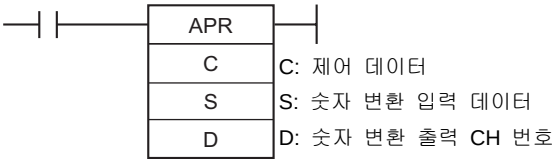


11CH에 연산 결과 0078이 출력됩니다.

APR

명령 이름	니모닉	동작 옵션	평션 번호	기능
숫자 변환	APR	@APR	069	SIN, COS, 꺾은선 근사 계산을 수행합니다.

심볼	APR



특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	UINT	가변
S	숫자 변환 입력 데이터	WORD	1
D	숫자 변환 출력 CH 번호	WORD	1

■ SIN 계산 지정 시

피연산자	값	데이터 범위
C	&0 또는 #0000	–
S	BCD #0000~0900	0°~90°
D	BCD #0000~9999 BCD #9999	0.0000~0.9999 1.0000

・SIN 계산(C = 0) 시:
S로 지정한 각도 데이터(×10－1단위의 BCD) 0000~0900(0°~90°)에 대한 SIN 값을 계산하여 결과를 소수점 이하 4자리를 나타내는 BCD 데이터 0000~9999(0.0000~0.9999)로 D에 출력합니다.
소수점 이하 5자리 이후는 버립니다.

■ COS 계산 지정 시

피연산자	값	데이터 범위
C	&1 또는 #0001	–
S	BCD #0000~0900	0°~90°
D	BCD #0000~999 BCD #9999	0.0000~0.9999 1.0000

・COS 계산 지정(C = 1) 시:
S로 지정한 각도 데이터(×10－1단위의 BCD) 0000~0900(0°~90°)에 대한 COS 값을 계산하여 결과를 소수점 이하 4자리를 나타내는 BCD 데이터 0000~9999(0.0000~0.9999)로 D에 출력합니다.
소수점 이하 5자리 이후는 버립니다.

주:
・SIN90° 및 COS0°에 대한 연산 결과는 1.0000이지만 9999(= 0.9999)를 출력합니다.

■ 꺾은선 근사 지정 시

피연산자	값	데이터 범위
C	CH 번호	−
S	부호 없는 BCD 16비트 데이터	#0000~9999
	부호 없는 BIN 16비트 데이터	&0000~65535
	부호 있는 BIN 16비트 데이터	−32768~32767
	부호 있는 BIN 32비트 데이터	−2147483648~2147483647
	단정도 부동 소수점 데이터	−∞, -3.402823×10^{38}~-1.175494×10^{-38}, $+1.175494\times10^{-38}$~$+3.402823\times10^{38}$, +∞
D	부호 없는 BCD 16비트 데이터	#0000~9999
	부호 없는 BIN 16비트 데이터	&0000~65535
	부호 있는 BIN 16비트 데이터	−32768~32767
	부호 있는 BIN 32비트 데이터	−2147483648~2147483647
	단정도 부동 소수점 데이터	−∞, -3.402823×10^{38}~-1.175494×10^{-38}, $+1.175494\times10^{-38}$~$+3.402823\times10^{38}$, +∞

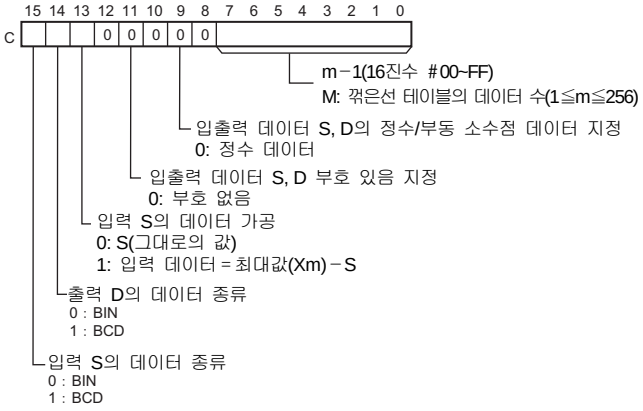
꺾은선 근사
C에 채널 주소가 입력되면 꺾은선 근사로 지정됩니다. C에서는 C에 입력한 채널 주소를 기점으로 한 꺾은선 테이블의 데이터 수 또는 입출력 데이터 종류(BCD/BIN) 등을 지정합니다.

입출력 데이터로 사용할 수 있는 데이터 종류는 다음 5가지입니다.

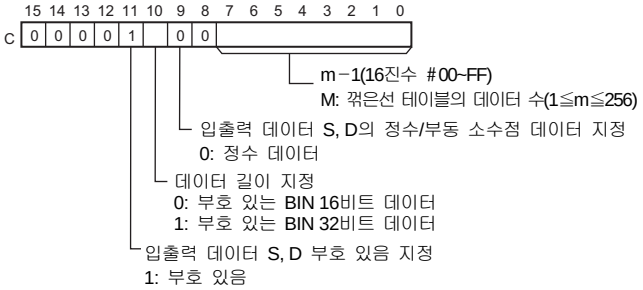
- 부호 없는 BCD 16비트 데이터
- 부호 없는 BIN 16비트 데이터
- 부호 있는 BIN 16비트 데이터(CJ2/CS1-H/CJ1-H/CJ1M/CS1D CPU 유니트만 해당)
- 부호 있는 BIN 32비트 데이터(CJ2/CS1-H/CJ1-H/CJ1M/CS1D CPU 유니트만 해당)
- 단정도 부동 소수점 데이터(CJ2/CS1-H/CJ1-H/CJ1M/CS1D CPU 유니트만 해당)

APR

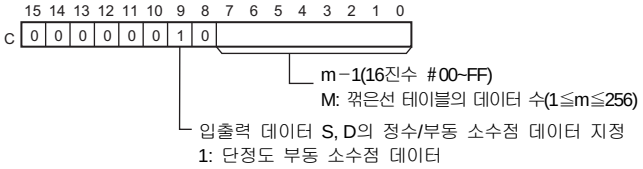
· 부호 없는 정수 데이터(BIN 또는 BCD) 지정 시



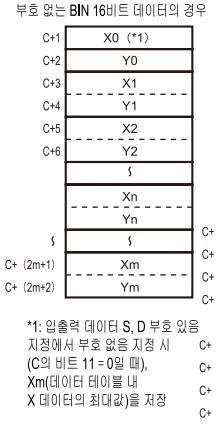
· 부호 있는 정수 데이터(BIN) 지정 시



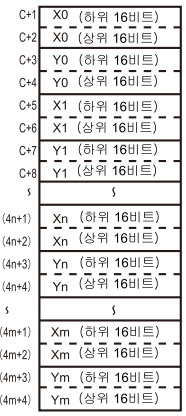
· 단정도 부동 소수점 데이터 지정 시



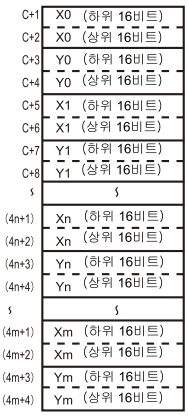
· BCD 16비트 데이터, 부호 있는/부호 없는 BIN 16비트 데이터의 경우



부호 있는 BIN 32비트 데이터의 경우



단정도 부동 소수점 데이터의 경우



*1: 입출력 데이터 S, D 부호 있음 지정에서 부호 없음 지정 시 (C의 비트 11 = 0일 때), Xm(데이터 테이블 내 X 데이터의 최대값)을 저장

주: X1 < X2 < Xm...입니다.
꺾은선 데이터(Xm, Ym)는 제어 데이터에서의 입출력 지정에 관계 없이 BIN 데이터를 저장합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
S												○						
D											–							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C의 데이터가 0~1의 범위에 없을 때 ON(상수 지정 시) · C의 꺾은선 데이터가 X1≦X2≦···≦Xm이 아닐 때 ON(꺾은선 근사 계산 지정 시) · S의 데이터가 BCD가 아닐 때 ON(꺾은선 근사 계산을 지정하고 입력 종류 BCD를 지정했을 때) · S의 데이터가 1워드 상수를 지정했을 때(꺾은선 근사 계산을 지정하고 입출력 데이터 S, D를 부동 소수점 데이터로 지정했을 때) · S의 데이터가 0000~0900의 BCD 데이터가 아닐 때 ON(SIN/COS 계산 지정 시) · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· D의 데이터의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

■ 꺾은선 근사 계산 지정 시(C = CH 지정):

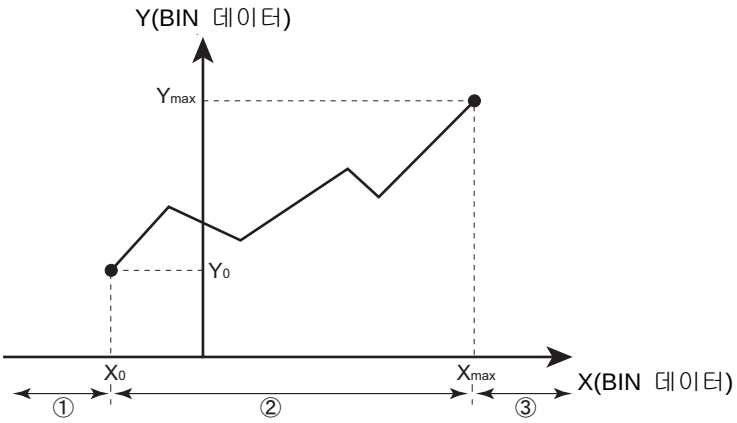
다음 변환식을 사용하여, S로 지정한 입력 데이터를 C로 지정한 꺾은선 데이터(Xn, Yn)에 기초하여 근사 계산을 수행하고 결과를 D로 지정한 채널에 출력합니다.

① $S < X_0$일 때 변환 결과 = Y_0

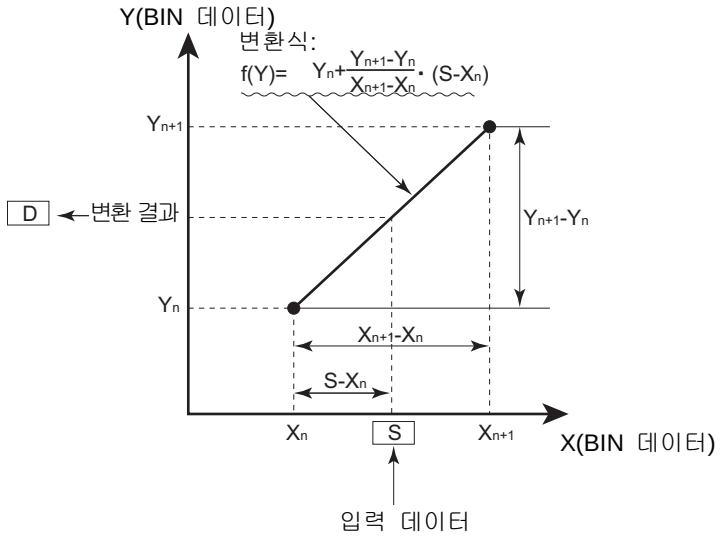
② $X_0 \leqq S \leqq X_{max}$일 때 $X_n < S < X_{n+1}$이면

변환 결과 = $Y_n + [\{Y_{n+1} - Y_n\}/\{X_{n+1} - X_n\}] \times \{\text{입력 데이터 } S - X_n\}$

③ $X_{max} < S$일 때 변환 결과 = Y_{max}



또한 꺾은선 테이블 데이터에는 최대 256개의 데이터를 저장할 수 있습니다.



APR

각 데이터 종류를 사용하기 위해서는 다음 설정을 수행하십시오.

■ 부호 없는 BCD 16비트 데이터의 경우

입력 데이터와 출력 데이터 중 한쪽 또는 양쪽에 BCD 데이터를 사용할 수 있습니다.

또한 입력 데이터에 'S로 지정한 값을 그대로 사용'하거나 'Xm－S'(꺾은선 데이터 X의 최대값－S로 지정한 값)를 선택할 수 있습니다.

설정 항목	꺾은선 데이터의 시작 채널 비트	설정값
입력 데이터 S의 BIN/BCD 지정	15	0: BIN 1: BCD
출력 데이터 D의 BIN/BCD 지정	14	0: BIN 1: BCD
입력 데이터 S의 데이터 가공	13	0: 그대로 사용 1: 입력 데이터 = Xm-S
입출력 데이터 S, D 부호 있음 지정	11	0: 부호 없음
입출력 데이터 S, D 부호 있음 지정 시 데이터 길이 지정	10	무효 (데이터 길이는 16비트로 고정)
입출력 데이터 S, D의 정수/부동 소수점 데이터 지정	09	0: 정수 데이터

■ 부호 없는 BIN 16비트 데이터의 경우

입력 데이터와 출력 데이터 중 한쪽 또는 양쪽에 BIN 데이터를 사용할 수 있습니다.

또한 입력 데이터에 'S로 지정한 값을 그대로 사용'하거나 'Xm－S'(꺾은선 데이터 X의 최대값－S로 지정한 값)를 선택할 수 있습니다.

설정 항목	꺾은선 데이터의 시작 채널 비트	설정값
입력 데이터 S의 데이터 종류	15	0: BIN 1: BCD
출력 데이터 D의 데이터 종류	14	0: BIN 1: BCD
입력 데이터 S의 데이터 가공	13	0: 그대로 사용 1: 입력 데이터 = Xm-S
입출력 데이터 S, D 부호 있음 지정	11	0: 부호 없음
입출력 데이터 S, D 부호 있음 지정 시 데이터 길이 지정	10	무효 (데이터 길이는 16비트로 고정)
입출력 데이터 S, D의 정수/부동 소수점 데이터 지정	09	0: 정수 데이터

■ 부호 있는 BIN 16비트 데이터의 경우

(CJ2/CS1-H/CJ1-H/CJ1M/CS1D CPU 유니트만 해당)

설정 항목	꺾은선 데이터의 시작 채널 비트	설정값
입력 데이터 S의 데이터 종류	15	0: BIN
출력 데이터 D의 데이터 종류	14	0: BIN
입력 데이터 S의 데이터 가공	13	0
입출력 데이터 S, D 부호 있음 지정	11	1: 부호 있음
입출력 데이터 S, D 부호 있음 지정 시 데이터 길이 지정	10	0: 부호 있음 (BIN 16비트 데이터)
입출력 데이터 S, D의 정수/부동 소수점 데이터 지정	09	0: 정수 데이터

■ 부호 있는 BIN 32비트 데이터의 경우

(CS1/CJ1 CPU 유니트는 불가능)

설정 항목	꺾은선 데이터의 시작 채널 비트	설정값
입력 데이터 S의 데이터 종류	15	0: BIN
출력 데이터 D의 데이터 종류	14	0: BIN
입력 데이터 S의 데이터 가공	13	0
입출력 데이터 S, D 부호 있음 지정	11	1: 부호 있음
입출력 데이터 S, D 부호 있음 지정 시 데이터 길이 지정	10	0: 부호 있음 (BIN 32비트 데이터)
입출력 데이터 S, D의 정수/부동 소수점 데이터 지정	09	0: 정수 데이터

주: '입출력 데이터 S, D 부호 있음 지정 시 데이터 길이 지정'에서 부호 있는 BIN 32비트 데이터를 지정하고, S에 16상수를 사용한 경우 입력 데이터를 32비트 데이터로 변환(부호 확장)하여 계산합니다.

■ 단정도 부동 소수점 데이터의 경우

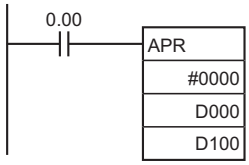
(CJ2/CS1-H/CJ1-H/CJ1M/CS1D CPU 유니트만 해당)

설정 항목	꺾은선 데이터의 시작 채널 비트	설정값
입력 데이터 S의 데이터 종류	15	0
출력 데이터 D의 데이터 종류	14	0
입력 데이터 S의 데이터 가공	13	0
입출력 데이터 S, D 부호 있음 지정	11	0
입출력 데이터 S, D 부호 있음 지정 시 데이터 길이 지정	10	0
입출력 데이터 S, D의 정수/부동 소수점 데이터 지정	09	1: 부동 소수점 데이터

주: '입출력 데이터 S, D의 정수/부동 소수점 데이터 지정'에서 단정도 부동 소수점 데이터를 지정한 경우 입력 데이터 S에 상수를 지정할 수 없습니다.

샘플 프로그램

■ SIN 계산(C=0)



APR 명령의 동작

・ 제어 데이터 C는 &0 또는 #0000을 설정하십시오.
(예) sin30° = 0.5000

입력 데이터 S: D0

0	$\times 10^{1}$	$\times 10^{0}$	$\times 10^{-1}$
0	3	0	0

$\times 10^{-1}$ 단위의 BCD 데이터로 설정하십시오.
설정 범위: #0000~0900 (0°~90°)

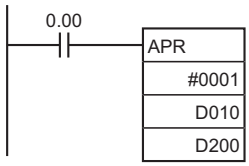
⇨

출력 데이터 D: D100

$\times 10^{-1}$	$\times 10^{-2}$	$\times 10^{-3}$	$\times 10^{-4}$
5	0	0	0

소수점 이하 4자리의 BCD 데이터로 출력됩니다.
소수점 이하 5자리 이후는 버립니다.

■ COS 계산(C=1)



APR 명령의 동작

・ 제어 데이터 C는 &1 또는 #0001을 설정하십시오.
(예) cos30° = 0.8660

입력 데이터 S: D10

0	$\times 10^{1}$	$\times 10^{0}$	$\times 10^{-1}$
0	3	0	0

$\times 10^{-1}$ 단위의 BCD 데이터로 설정하십시오.
설정 범위: #0000~0900 (0°~90°)

⇨

출력 데이터 D: D200

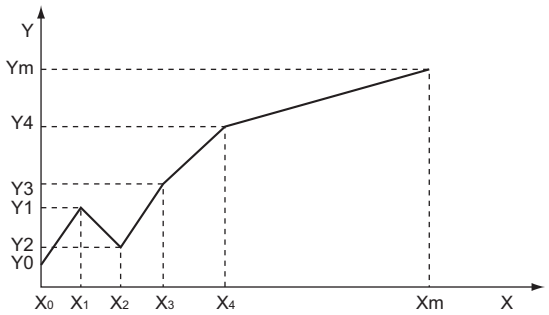
$\times 10^{-1}$	$\times 10^{-2}$	$\times 10^{-3}$	$\times 10^{-4}$
8	6	6	0

소수점 이하 4자리의 BCD 데이터로 출력됩니다.
소수점 이하 5자리 이후는 버립니다.

■ 꺾은선 근사 계산(C: CH 지정 시)

・ 부호 없는 BCD/BIN 16비트 데이터의 경우

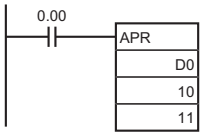
제어 데이터 C에 설정한 CH 이후에 꺾은선 데이터를 설정해두면 그 데이터에 기초하여 S CH의 입력 데이터에 대한 근사값을 계산한 후 D CH에 출력합니다.

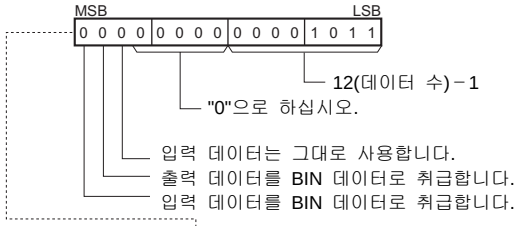


APR

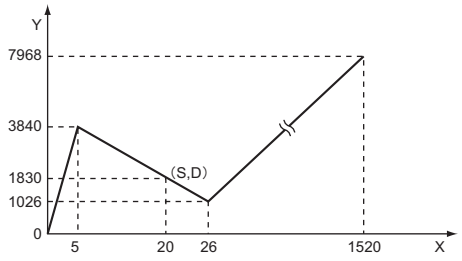
- Yn = f(Xn) Yo = f(Xo)
- Xn − 1 < Xn으로 하십시오.
- Xn, Yn은 모두 BIN 데이터로 입력하십시오.

C+1	Xm(데이터 테이블 내 X 데이터의 최대값)
C+2	Y_0
C+3	X_1
C+4	Y_1
C+5	X_2
C+6	Y_2
	≀
	Xn
	Yn
≀	≀
C+（2m+1）	Xm
C+（2m+2）	Ym





D0	000B	
D1	&1520	····· X_{12}
D2	&0	····· Y_0
D3	&5	····· X_1
D4	&3840	····· Y_1
D5	&26	····· X_2
D6	&1026	····· Y_2
≀	≀	
D25	&1520	····· X_{12}
D26	&7968	····· Y_{12}



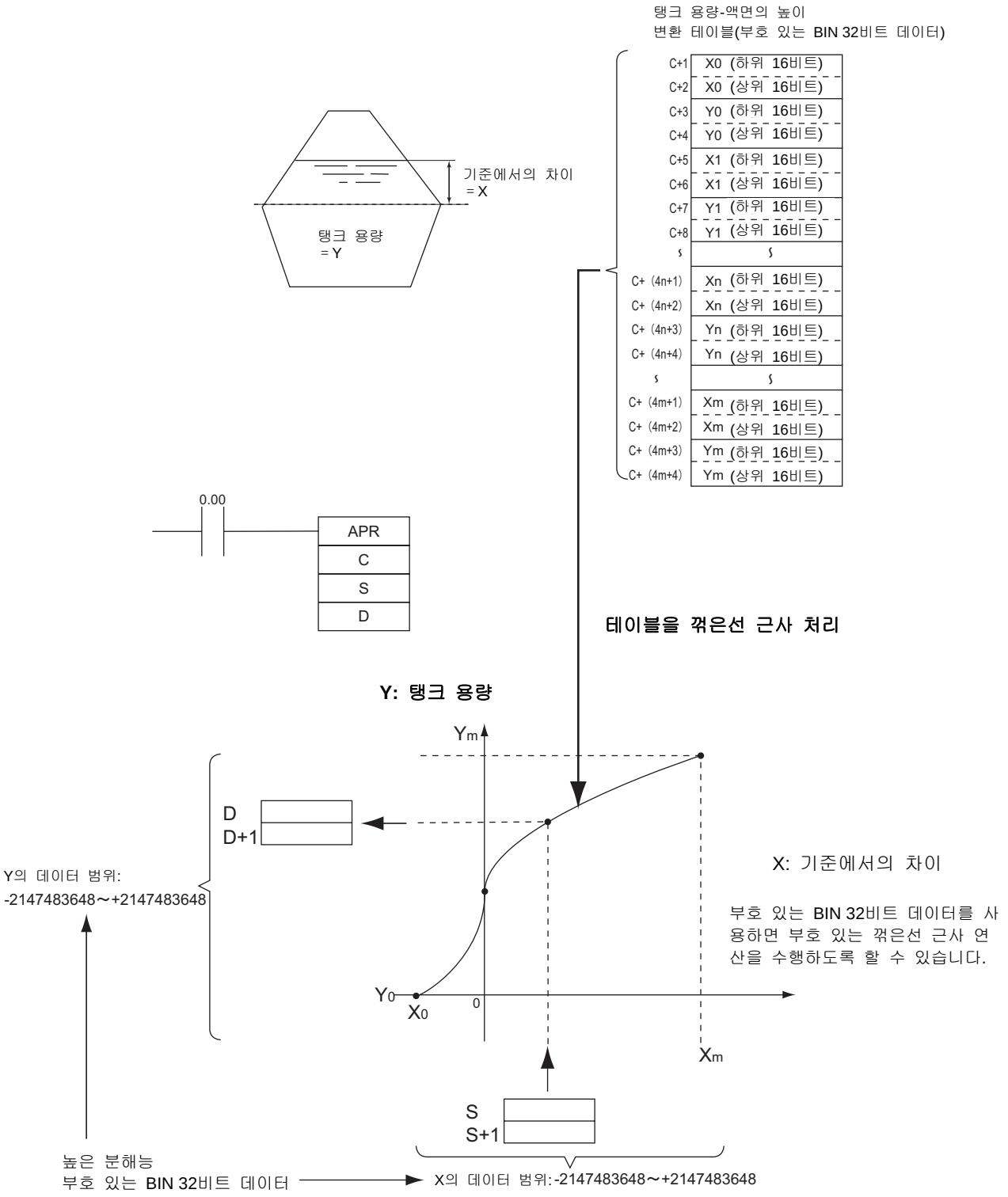
입력 데이터(BIN) S:0010CH &20 ⇨ 출력 데이터(BIN) D:0011CH &1830

(계산식)

$$Y = 3840 + \frac{1026 - 3840}{26 - 5} \times (20 - 5)$$

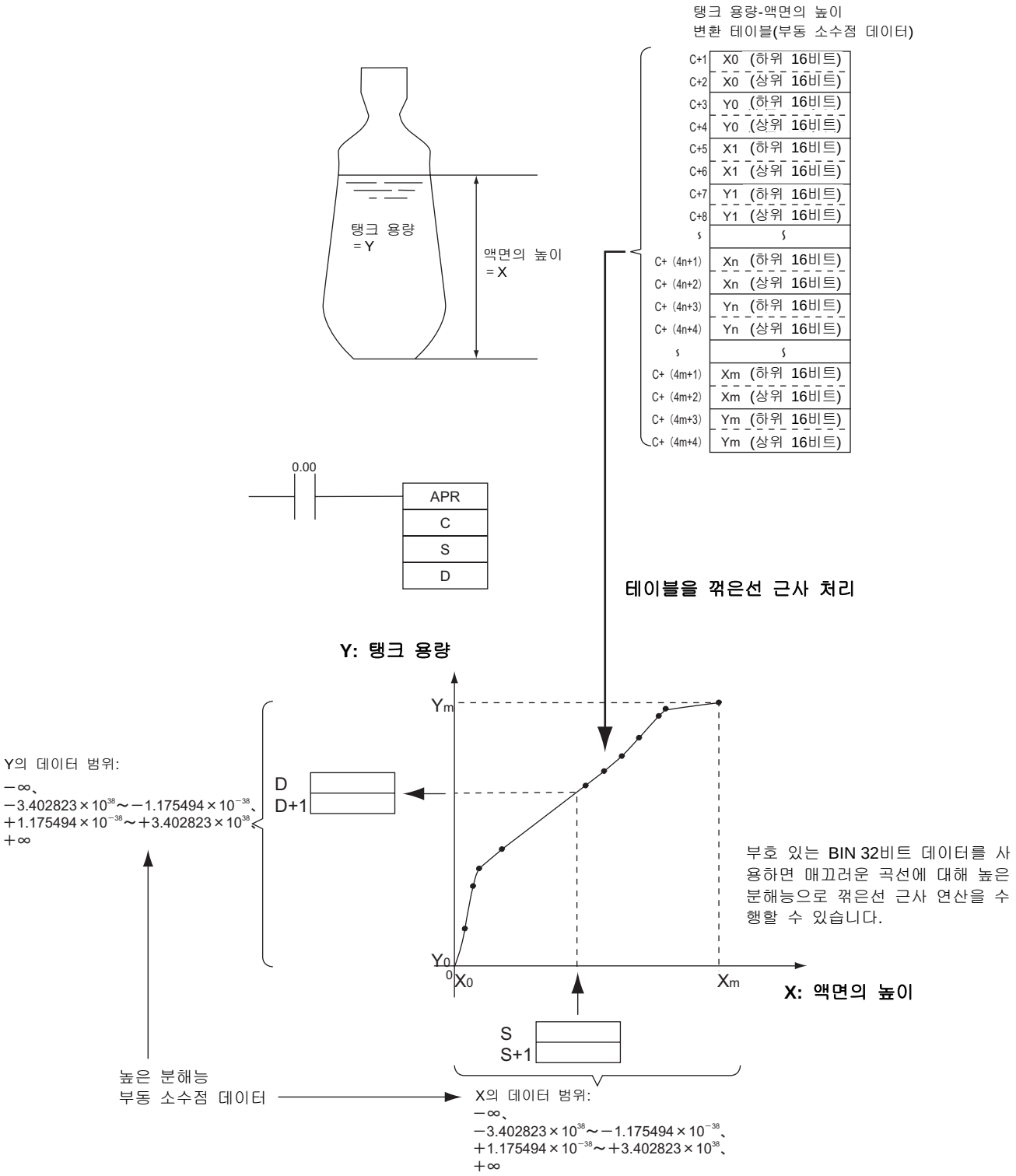
$$= 1830$$

· 부호 있는 BIN 16/32비트 데이터의 경우(CJ2/CS1-H/CJ1-H/CJ1M/CS1D CPU 유니트만 해당)
탱크 형태에 맞춰 액면의 높이→탱크 용량으로 변환



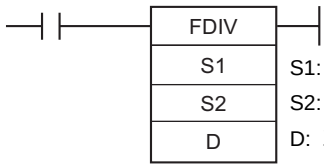
APR

· 부동 소수점 데이터의 경우(CJ2/CS1-H/CJ1-H/CJ1M/CS1D CPU 유니트만 해당)
탱크 형태에 맞춰 액면의 높이→탱크 용량으로 변환



FDIV

명령 이름	니모닉	동작 옵션	평션 번호	기능
부동 소수점 제산 (BCD)	FDIV	@FDIV	079	가수 7자리와 지수 1자리의 2CH(BCD 8자리) 부동 소수점을 나눕니다.

심볼	FDIV
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S1	피제산 데이터 하위 CH 번호	UDINT	2
S2	제산 데이터 하위 CH 번호	UDINT	2
D	제산 결과 출력 하위 CH 번호	UDINT	2

■ 영역 종류

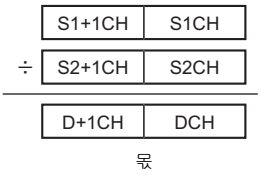
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S1	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S2																		
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 피제산 데이터의 가수부가 BCD 데이터가 아닐 때 ON · 제산 데이터의 가수부가 BCD 데이터가 아닐 때 ON · 제산 데이터가 0일 때 ON · 연산 결과가 유효 자릿수를 초과했을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF

기능

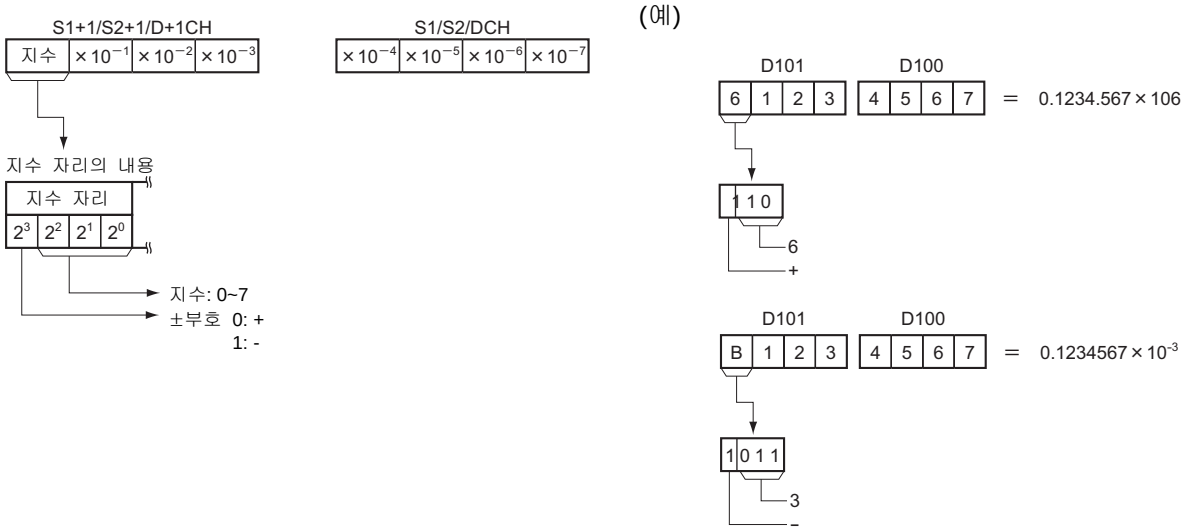
S1로 지정한 데이터와 S2로 지정한 데이터를 [지수부 1자리+가수부 BCD 7자리]의 부동 소수점 데이터(32비트: BCD)로 나누고, 몫을 부동 소수점 데이터로 D+1, D에 출력합니다.



주:
· S2+1CH의 최상위 자리는 지수 자리가 되고 0~F가 유효하게 됩니다. 지수 자리 이외는 연산 데이터가 BCD가 되도록 설정하십시오.
· S1+1, S2+1CH, D+1CH가 영역을 초과하지 않도록 주의하십시오.
· 몫은 유효 숫자 7자리가 출력되고 8자리 이후는 버립니다.

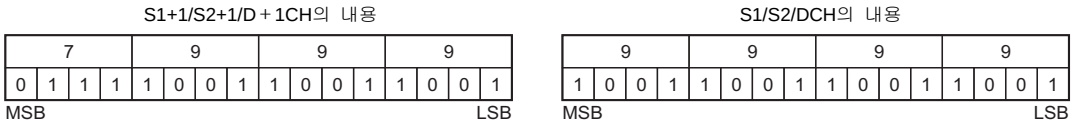
FDIV

■ 부동 소수점 표시 방법

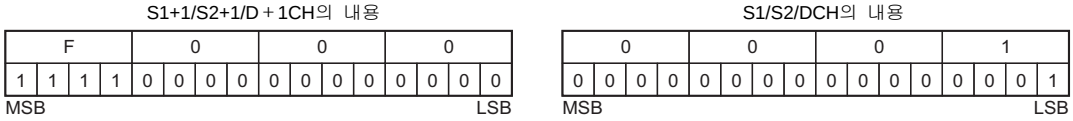


■ 피제산 데이터, 제산 데이터, 몫의 유효 범위

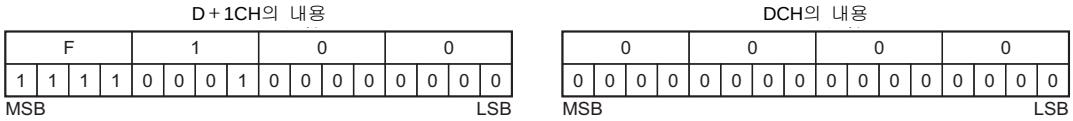
· 값 0.9999999×10^{7}



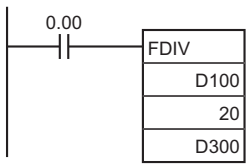
· 최대값 0.0000001×10^{-7}(피제산 데이터, 제산 데이터)



· 최대값 0.1000000×10^{-7}(몫)

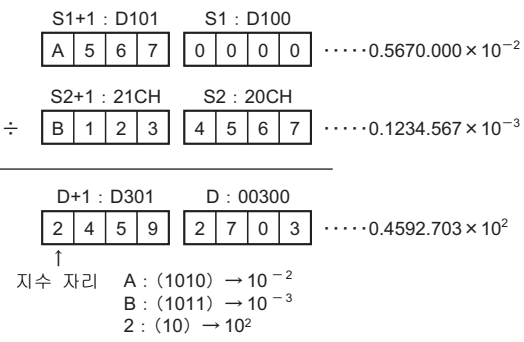


샘플 프로그램



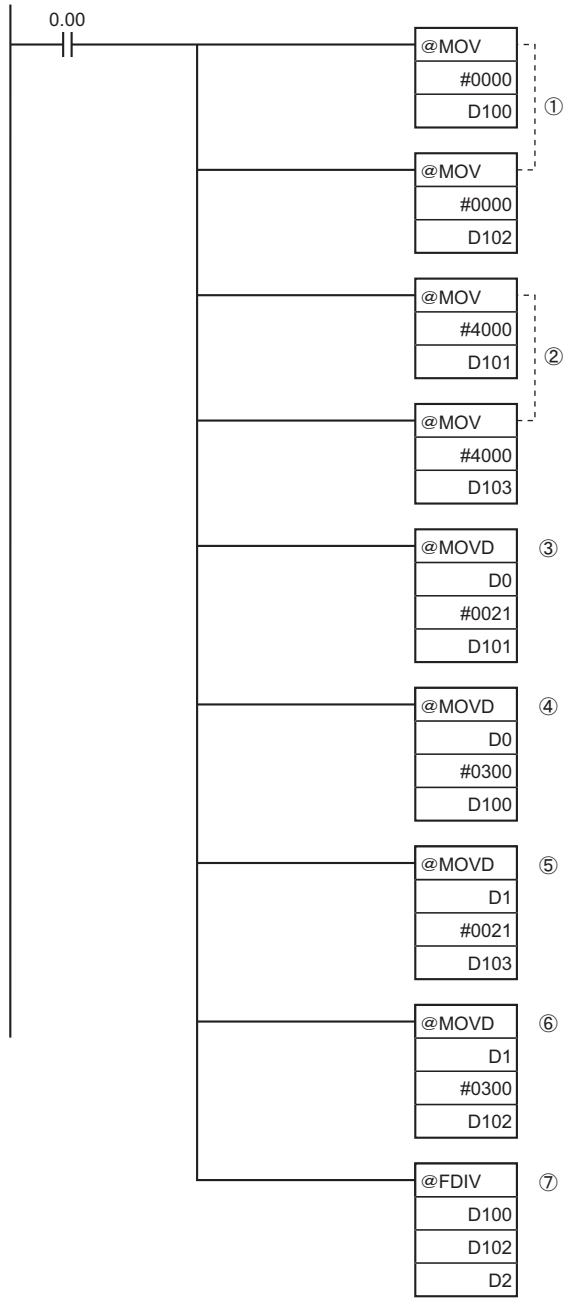
0.00이 ON이면 D101, D100의 내용을 21, 20CH의 내용으로 나누고 몫을 D301, D300에 출력합니다.

(예) $0.567 \times 10^{-2} \div 0.1234.567 \times 10^{-3} = 0.4592.703 \times 10^{2}$

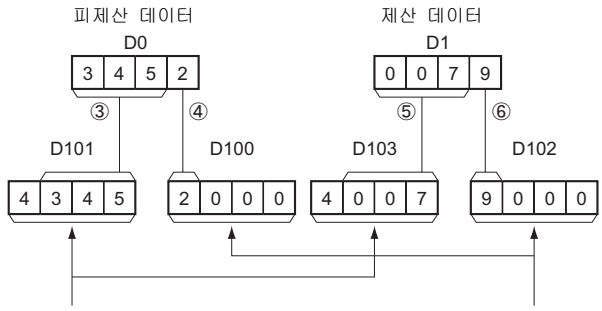


응용 예

데이터 메모리 D0의 내용을 D1의 내용으로 부동 소수점 제산(BCD)하여 결과를 D2, D3에 출력합니다.
(예) $0.3452\times10^{4}\div0.0079\times10^{4}=0.436962\times10^{2}$



부동 소수점 제산 명령은 각각 2CH씩 데이터로 사용하므로 데이터 메모리 D100~D103을 연산용으로 설정합니다.



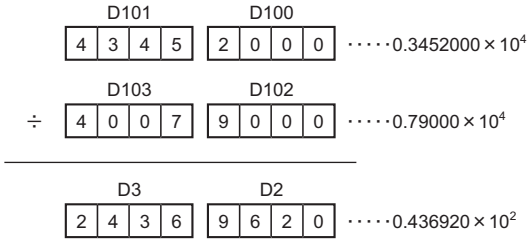
② #4000을 전송합니다. ① #0000을 전송합니다.

〔3452＝0.3452×104〕 〔D100 D102를〕
〔0079＝0.0079×104〕 〔클리어합니다.〕

③~⑥ 자릿수 전송을 통해 지정된 자리에 데이터를 전송합니다.

⑦ D101, D100÷D103, D102의 부동 소수점 제산을 수행합니다.

연산 결과, 몫은 D3, D2에 출력됩니다.



BCNT

명령 이름	니모닉	동작 옵션	평션 번호	기능
비트 카운터	BCNT	@BCNT	067	CH 데이터 내에서 "1"인 비트의 총 수를 계산합니다.

	BCNT
심볼	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
W	카운트 CH 수	UINT	1
S	카운트 하위 CH 번호	UINT	가변
D	카운트 결과 출력 CH 번호	UINT	1

W: 카운트 CH 수

10진수 &1~65535 또는 16진수 #0001~FFFF

■ 영역 종류

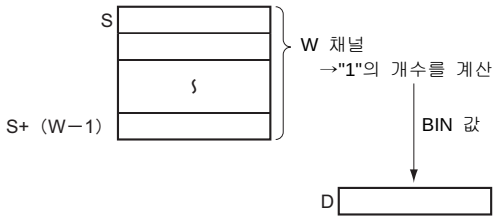
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
W	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
S											–	–						
D												○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· W의 데이터가 0일 때 ON. 또는 카운트 결과 D의 데이터가 65535 (16진수 #FFFF)를 초과했을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 카운트 결과(D의 데이터)가 0일 때 ON · 그 이외일 때 OFF

기능

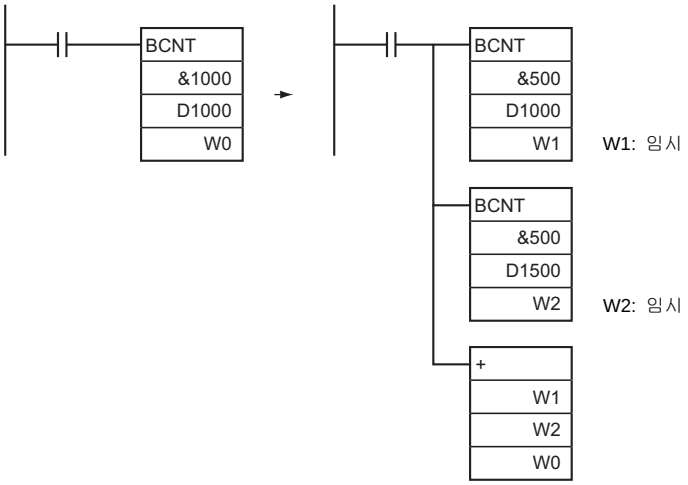
S로 지정한 카운트 하위 CH 번호부터 지정된 CH 수 (W) 분량의 데이터에 대해 "1"인 비트의 총 수를 계산하여 결과를 BIN 값으로 D에 출력합니다.



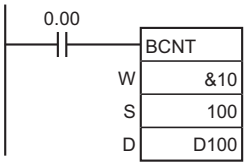
사용상의 주의 사항

대량의 채널을 비트 카운트하는 경우 명령 실행에 시간이 걸립니다.
이 명령 실행 중에 인터럽트가 발생해도 명령 실행이 끝날 때까지 인터럽트 태스크는 기동되지 않습니다.
명령을 분할하면 이와 같은 현상을 피할 수 있습니다.

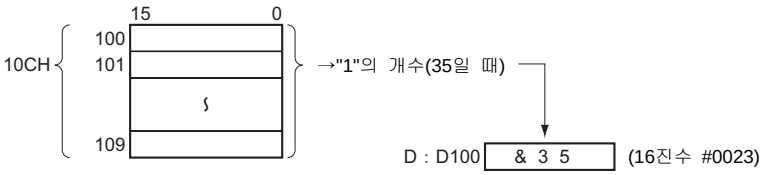
예)



샘플 프로그램



0.00이 ON이면 100CH부터 10CH의 데이터 중 "1"의 개수를 계산하여 BIN 값으로 D100에 저장합니다.



부동 소수점 변환 및 연산 명령

니모닉	명령 이름	펑션 번호	페이지
FIX	부동 소수점→16비트 BIN 변환	450	3-347
FIXL	부동 소수점→32비트 BIN 변환	451	
FLT	16비트 BIN→부동 소수점 변환	452	3-349
FLTL	32비트 BIN→부동 소수점 변환	453	
＋F	부동 소수점 가산	454	3-351
－F	부동 소수점 감산	455	
＊F	부동 소수점 승산	456	
／F	부동 소수점 제산	457	
RAD	각도 →라디안 변환	458	3-354
DEG	라디안→각도 변환	459	3-356
SIN	SIN 연산	460	3-358
COS	COS 연산	461	
TAN	TAN 연산	462	
SINQ	고속 SIN 연산	475	3-361
COSQ	고속 COS 연산	476	
TANQ	고속 TAN 연산	477	
ASIN	SIN^{-1} 연산	463	3-364
ACOS	COS^{-1} 연산	464	
ATAN	TAN^{-1} 연산	465	
SQRT	제곱근 연산	466	3-367
EXP	지수 연산	467	3-369
LOG	로그 연산	468	3-371
PWR	거듭제곱 연산	840	3-373
=F,<>F,<F,<=F,>F,>=F	단정도 부동 소수점 데이터 비교	329~334	3-375
FSTR	부동 소수점<단>→문자열 변환	448	3-378
FVAL	문자열→부동 소수점<단> 변환	449	3-382
MOVF	부동 소수점 전송<단>	469	3-386

부동 소수점 변환 및 연산 명령

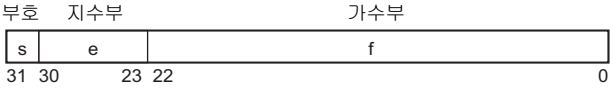
■ 데이터 형식

부동 소수점 데이터는 실수를 부호, 가수, 지수로 표현한 데이터입니다. 임의의 데이터를 부동 소수점 형식으로 나타내면 다음과 같습니다.

실수값 = $(-1)^{s}2^{e-127}(1.f)$

s: 부호
e: 지수
f: 가수

부동 소수점 데이터의 형식은 IEEE754 규격의 단정도를 준수합니다. 단정도 데이터는 32비트로 표현되며 형식은 다음과 같습니다.



데이터	비트 수	내용
s: 부호	1	0: 양, 1: 음
e: 지수	8	지수부 e의 값은 0~255입니다. e에서 127을 뺀 값이 실제 지수가 됩니다. 따라서 지수는 -127~128이 됩니다. e = 0, e = 255일 때는 특수 숫자를 나타냅니다.
f: 가수	23	2진수에서 부동 소수점의 가수부를 나타냅니다. 정규화 표현을 사용하면 2.0 > 1.f ≧ 1.0을 나타냅니다.

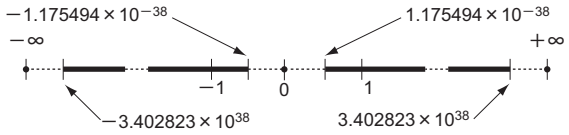
■ 유효 자릿수

부동 소수점 데이터의 유효 자릿수는 7자리입니다.

■ 부동 소수점 데이터에서 표현 방법

· 표현 가능한 수

- $-\infty$
- $-3.402823\times10^{38} \leqq$ 값 $\leqq -1.175494\times10^{-38}$
- 0
- $+1.175494\times10^{-38} \leqq$ 값 $\leqq +3.402823\times10^{38}$
- $+\infty$
- 비수(NaN)



· 특수 숫자

비수(NaN), ±∞, 0의 형식은 다음과 같이 정의합니다.

· 비수(NaN)	e = 255, f ≠ 0
· $+\infty$	e = 255, f = 0, s = 0
· $-\infty$	e = 255, f = 0, s = 1
· 0	e = 0

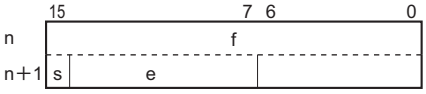
비수(NaN)란 부동 소수점 데이터로 처리되지 않는 값입니다.
단, 부동 소수점 및 연산 명령의 결과가 비수(NaN)가 되지는 않습니다.

■ 부동 소수점 데이터 형식의 저장 방법

주변 툴 CX-Programmer의 I/O 메모리 편집 화면에서 데이터 형식을 '부동 소수점'으로 지정하면 입력된 소수점 데이터가 위의 IEEE754 규격의 데이터 형식으로 자동 변환되어 I/O 메모리에 저장됩니다. 또한 위의 IEEE754 규격의 데이터 형식으로 저장된 데이터는 자동으로 소수점 데이터로 모니터링됩니다.

따라서 사용자가 앞에서 설명한 IEEE754 규격의 데이터 형식을 고려할 필요는 없습니다.

부동 소수점 데이터가 2CH를 점유한다는 점만 고려하십시오.



■ 참고

부동 소수점 수로 표현하는 값의 종류

지수부(e) / 가수부(f)	0	0도 모든 비트 1(255)도 아님	모든 비트 1(255)
0	0	정규화 수	무한대
0 이외	비정규화 수		비수(NaN)

주: 비정규화 수는 정규화 수로 표현할 수 없는 범위의 절대값이 작은 부동 소수점 수를 표현하지만 정규화 수에 비해 유효 자릿수가 적어집니다. 따라서 연산 결과가 비정규화 수가 되는 경우 또는 중간 결과가 비정규화 수가 되는 경우, 결과의 유효 자릿수는 보장되지 않으므로 주의하십시오.

(1) 정규화 수

일반적인 실수값을 표현합니다.

부호부는 0(양) 또는 1(음)로 값의 부호를 나타냅니다.

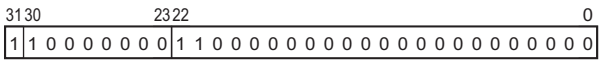
지수부의 값은 1~254입니다. 실제 지수는 이 값에서 127을 뺀 값이며 범위는 −126~127입니다.

가수부의 값은 0~$(2^{23}-1)$이 됩니다. 실제 가수는 2^{23}의 비트를 1로 가정하고 바로 다음에 소수점이 있다고 해석합니다.

정규화 수를 표현하는 값은 다음과 같습니다.

$(-1)^{(부호부)} \times 2^{(지수부)-127} \times (1 + <가수부> \times 2^{-23})$

예



부호: −

지수: 128 − 127 = 1

가수: $1 + (2^{22} + 2^{21}) \times 2^{-23} = 1 + (2^{-1} + 2^{-2}) = 1 + 0.75 = 1.75$

값: $-1.75 \times 2^{1} = -3.5$

(2) 비정규화 수

절대값이 작은 실수값을 표현합니다.

부호부는 0(양) 또는 1(음)로 값의 부호를 나타냅니다.

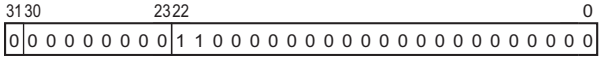
지수부는 0이고 실제 지수는 −126입니다.

가수부는 1~$2^{23}-1$이며, 실제 가수는 2^{23}의 비트를 0으로 가정하고 바로 다음에 소수점이 있다고 해석합니다.

비정규화 수를 표현하는 값은 다음과 같습니다.

$(-1)^{(부호부)} \times 2^{-126} \times (<가수부> \times 2^{-23})$

예



부호: +

지수: −126

가수: $0 + (2^{22} + 2^{21}) \times 2^{-23} = 0 + (2^{-1} + 2^{-2}) = 0 + 0.75 = 0.75$

값: 0.75×2^{-126}

(3) 0

부호부는 0(양) 또는 1(음)이며, 각각 +0.0, -0.0을 나타냅니다.

지수부와 가수부는 모두 0입니다.

+0.0, −0.0은 둘 다 값으로 0.0을 나타냅니다. 0의 부호에 의한 각 연산에서 기능의 차이는 '부동 소수점의 사칙연산'을 참조하십시오.

(4) 무한대

부호부는 0(양) 또는 1(음)이며, 각각 +∞, −∞를 나타냅니다.

지수부는 255(2^8-1)입니다.

가수부는 0입니다.

(5) 비수(NaN)

'0.0/0.0', '∞/∞', '∞−∞' 등, 결과가 숫자 또는 무한대에 대응하지 않는 연산 결과를 얻게 됩니다.

지수부는 255(2^8-1)입니다.

가수부는 0 이외의 값입니다.

주: 비수(NaN)의 부호 및 가수 필드(0 이외)의 값에 대해서는 규정되어 있지 않습니다.

■ 부동 소수점의 사칙연산

(1) 결과값의 반올림 방법

부동 소수점 수의 사칙연산 결과에 대한 정확한 값이 내부 표현의 가수의 유효 숫자를 초과한 경우는 다음 규칙에 따라 반올림합니다.

① 결과의 값은 그 값에서 가장 가까운 2개의 부동 소수점 수의 내부 표현 중 가까운 쪽을 향해 반올림됩니다.

② 결과의 값이 그 값에서 가장 가까운 2개의 부동 소수점 수의 정중앙에 있는 경우는 가수의 마지막 자리가 0이 되는 방향으로 반올림됩니다.

(2) 오버 플로, 언더 플로, 무효 연산일 때의 처리

실행 시의 오버 플로, 언더 플로, 무효 연산에 대해서는 다음과 같이 처리합니다.

① 오버 플로의 경우는 결과의 부호에 따라 양 또는 음의 무한대가 됩니다.

② 언더 플로의 경우는 결과의 부호에 따라 양 또는 음의 0이 됩니다.

③ 무효 연산은 부호가 반대인 무한대를 더한 경우, 부호가 같은 무한대를 뺀 경우, 0과 무한대를 곱한 경우, 0을 0으로 또는 무한대를 무한대로 나눈 경우에 발생합니다.
위와 같은 경우, 결과는 비수(NaN)가 됩니다.

④ 부동 소수점 수에서 정수로 변환한 경우 오버 플로가 발생하면 결과의 값은 보장되지 않습니다.

(3) 특수한 값의 연산에 대한 주의 사항

특수한 값(0, 무한대, 비수(NaN))의 연산에 대한 주의 사항은 다음과 같습니다.

① 양의 0과 음의 0의 합은 양의 0이 됩니다.

② 부호가 같은 0의 차는 양의 0이 됩니다.

③ 피연산자의 한쪽 또는 양쪽에 비수(NaN)가 포함된 연산의 결과는 항상 비수(NaN)가 됩니다.

④ 비교 연산에서는 양의 0과 음의 0이 같은 것으로 취급됩니다.

⑤ 피연산자의 한쪽 또는 양쪽이 비수(NaN)인 비교 연산, 등가 연산의 결과는 '<>'에 대해서 항상 참이 되고 그 밖에는 항상 거짓이 됩니다.

■ 부동 소수점 데이터의 연산 결과

(1) 오버 플로

연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 큰 경우 오버 플로가 됩니다. 이런 경우, 연산 결과는 다음과 같습니다.

- 연산 결과가 양일 때: +∞
- 연산 결과가 음일 때: −∞

(2) 연산 결과가 0이 되는 경우

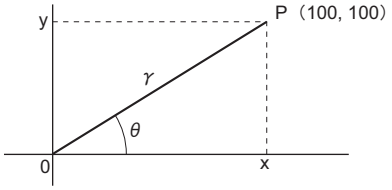
연산 결과의 가수부 f와 지수부 e가 양쪽 모두 0인 경우 연산 결과는 0이 됩니다. 또한 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최소값보다 작은 경우 언더 플로가 됩니다. 이 경우에도 연산 결과는 0이 됩니다.

(3) 연산 결과가 0일 때 ON이 되는 조건 플래그인 = 플래그의 경우 연산 결과의 가수부 f와 지수부 e가 모두 0일 때만 ON이 됩니다.

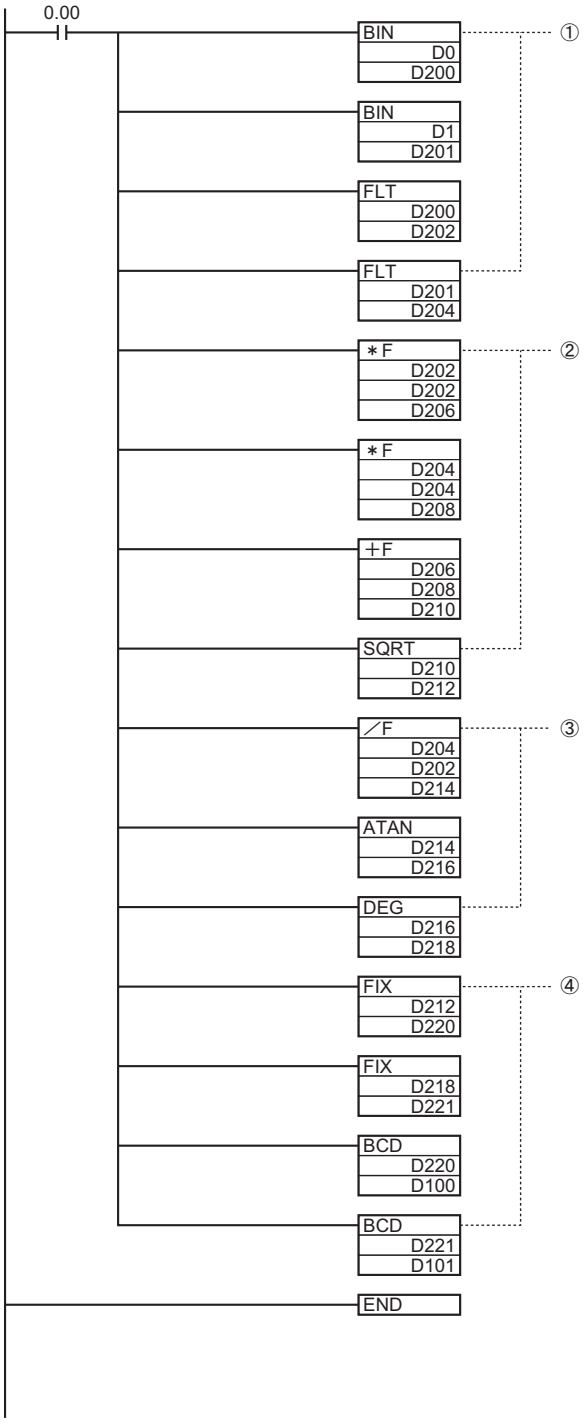
■ 부동 소수점 변환 및 연산의 예

X축, Y축의 좌표(x, y)가 데이터 메모리 D0과 D1에 BCD 4자리 데이터로 지정되었을 때 원점에서의 거리 r과 각도 θ(도 단위)를 구하여 D100, D101에 출력하는 프로그램의 예를 나타냅니다.

단, 연산 결과는 소수점 이하를 버린 값을 저장합니다.



■ 프로그램 예



・ 계산식

거리 $r = \sqrt{x^2 + y^2}$

각도 $\theta = \tan^{-1}\left(\frac{y}{x}\right)$

예: 좌표 (100, 100)이 지정된 경우를 생각해 보겠습니다. 위의 계산식으로 거리 r과 각도 θ를 구할 수 있습니다.

거리 $r = \sqrt{100^2 + 100^2} = 141.4214$

각도 $\theta = \tan^{-1}\left(\frac{100}{100}\right) \times 180 \div \pi = 45.0$

・ 데이터 메모리 내용

D0 | #0100 | x (BCD) → D100 | #0141 | r (BCD)

D1 | #0100 | y (BCD) → D101 | #0045 | θ (BCD)

① BCD 데이터에서 부동 소수점 데이터로 변환합니다.

- 작업 영역으로 D200 이후를 사용하고 있습니다.
- BIN 명령을 사용하여 일단 BIN 데이터로 변환하고 FLT 명령을 사용하여 부동 소수점 데이터로 변환합니다.
- x 값을 부동 소수점 데이터로 변환한 값이 D203, D202에 출력됩니다.
- y 값을 부동 소수점 데이터로 변환한 값이 D205, D204에 출력됩니다.

② 거리 r을 구하기 위해 부동 소수점 연산 명령을 사용하여 $x^2 + y^2$의 제곱근을 연산합니다. 연산 결과는 D213, D212에 부동 소수점 데이터로 출력됩니다.

③ 각도 θ를 구하기 위해 부동 소수점 연산 명령을 사용하여 $\tan^{-1}(y/x)$를 연산합니다.

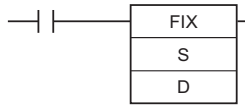
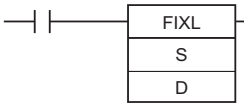
- 연산 결과는 D219, D218에 부동 소수점 데이터로 출력됩니다.
- ATAN 명령은 결과를 라디안 단위로 출력하므로 DEG 명령을 사용하여 도 단위로 변환합니다.

④ 부동 소수점 데이터에서 BCD 데이터로 변환합니다.

- FIX 명령을 사용하여 BIN 데이터로 변환한 후 BCD 명령을 사용하여 BCD 데이터로 변환합니다.
- 거리 r은 BCD 데이터로 D100에 출력됩니다.
- 각도 θ는 BCD 데이터로 D101에 출력됩니다.

FIX/FIXL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부동 소수점→16비트 BIN 변환	FIX	@FIX	450	지정된 부동 소수점 32비트 데이터를 부호 있는 BIN 16비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다.
부동 소수점→32비트 BIN 변환	FIXL	@FIXL	451	지정된 부동 소수점 32비트 데이터를 부호 있는 BIN 32비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다.

심볼	FIX	FIXL
	 S: 부동 소수점 데이터 하위 CH 번호 D: 변환 결과 출력 CH 번호	 S: 부동 소수점 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		FIX	FIXL	FIX	FIXL
S	부동 소수점 데이터 하위 CH 번호	REAL	REAL	2	2
D	변환 결과 출력 CH 번호	INT	DINT	1	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–						
D											–	○	–	○	–	–	–	–

조건 플래그의 동작

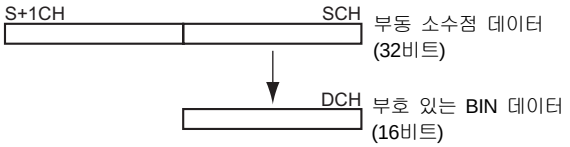
이름	라벨	내용
에러 플래그	P_ER	· FIX · S+1, S의 데이터가 −32768~+32767의 범위 안에 없을 때 ON · FIXL · S+1, S의 데이터가 −2147483648~+2147483647의 범위 안에 없을 때 ON · S+1, S의 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 변환 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

FIX/FIXL

기능

■ FIX

S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)의 정수부를 부호 있는 BIN(16비트)으로 변환하여 결과를 D에 출력합니다.



부동 소수점 데이터의 정수부를 부호 있는 BIN 데이터로 변환하여 결과를 지정된 채널에 출력합니다.
소수점 이하는 버립니다.

예:

부동 소수점 데이터		변환 결과(부호 있는 BIN 16비트)
3.5	→	3
−3.5	→	−3

■ FIXL

S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)의 정수부를 부호 있는 BIN(32비트)으로 변환하여 결과를 D+1, D에 출력합니다.



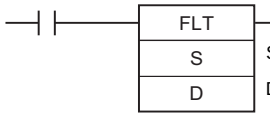
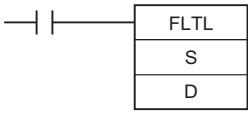
부동 소수점 데이터의 정수부를 부호 있는 BIN 2워드 데이터로 변환하여 결과를 지정된 채널에 출력합니다.
소수점 이하는 버립니다.

예:

부동 소수점 데이터		변환 결과(부호 있는 BIN 32비트)
2147483640.5	→	2147483640
−2147483640.5	→	−2147483640

FLT/FLTL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
16비트 BIN→부동 소수점 변환	FLT	@FLT	452	지정된 부호 있는 BIN 16비트 데이터를 부동 소수점 32비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다.
32비트 BIN→부동 소수점 변환	FLTL	@FLTL	453	지정된 부호 있는 BIN 32비트 데이터를 부동 소수점 32비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다.

심볼	FLT	FLTL
	 S: BIN 데이터 D: 변환 결과 출력 하위 CH 번호	 S: BIN 2 워드 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		FLT	FLTL	FLT	FLTL
S	BIN 데이터	INT	DINT	1	2
D	변환 결과 출력 하위 CH 번호	REAL	REAL	2	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
D											–	–						

조건 플래그의 동작

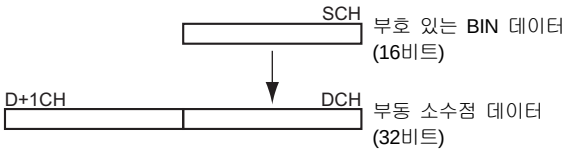
이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 변환 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 변환 결과가 음수가 되었을 때 ON · 그 이외일 때 OFF

FLT/FLTL

기능

■ FLT

S로 지정한 부호 있는 BIN 데이터(16비트)를 단정도 부동 소수점 데이터(32비트: IEEE754)로 변환하여 결과를 D+1, D에 출력합니다. 부동 소수점 데이터는 소수점 이하가 1자리인 0이 됩니다.



S에는 −32768~32767의 범위 내의 BIN 데이터를 지정할 수 있습니다.

−32768~32767의 범위 밖의 BIN 데이터를 변환하고 싶은 경우는 FLTL 명령을 사용하십시오.

예:

부호 있는 BIN 16비트		부동 소수점 데이터
3	→	3.0
−3	→	−3.0

■ FLTL

S로 지정한 부호 있는 BIN 데이터(32비트)를 단정도 부동 소수점 데이터(32비트: IEEE754)로 변환하여 결과를 D+1, D에 출력합니다. 부동 소수점 데이터는 소수점 이하가 1자리인 0이 됩니다.



- S에는 −2147483648~2147483647 범위 내의 BIN 2워드 데이터를 지정할 수 있습니다.
- 부동 소수점의 유효 자릿수는 24비트입니다. 따라서 16777215(24비트~최대값)를 초과한 값을 FLTL 명령으로 변환한 경우는 변환 결과에 오차가 생깁니다.

예:

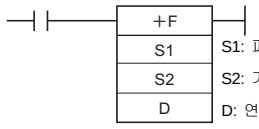
부호 있는 BIN 32비트		부동 소수점 데이터
16777215	→	16777215.0
−16777215	→	−16777215.0

+F, －F, *F, /F

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부동 소수점 가산	＋F	@＋F	454	지정된 부동 소수점 데이터(32비트)를 가산하여 결과를 지정된 채널에 출력합니다.
부동 소수점 감산	－F	@－F	455	지정된 부동 소수점 데이터(32비트)를 감산하여 결과를 지정된 채널에 출력합니다.
부동 소수점 승산	＊F	@＊F	456	지정된 부동 소수점 데이터(32비트)를 곱하여 결과를 지정된 채널에 출력합니다.
부동 소수점 제산	／F	@／F	457	지정된 부동 소수점 데이터(32비트)를 나눠서 결과를 지정된 채널에 출력합니다.

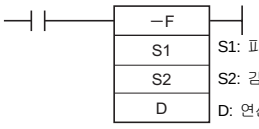
심볼

+F



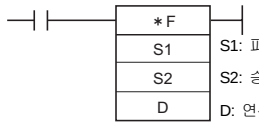
S1: 피가산 부동 소수점 데이터 하위 CH 번호
S2: 가산 부동 소수점 데이터 하위 CH 번호
D: 연산 결과 출력 하위 CH 번호

－F



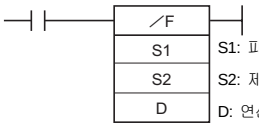
S1: 피감산 부동 소수점 데이터 하위 CH 번호
S2: 감산 부동 소수점 데이터 하위 CH 번호
D: 연산 결과 출력 하위 CH 번호

＊F



S1: 피승산 부동 소수점 데이터 하위 CH 번호
S2: 승산 부동 소수점 데이터 하위 CH 번호
D: 연산 결과 출력 하위 CH 번호

／F



S1: 피제산 부동 소수점 데이터 하위 CH 번호
S2: 제산 부동 소수점 데이터 하위 CH 번호
D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S1	＋F: 피가산 부동 소수점 데이터 하위 CH 번호 －F: 피감산 부동 소수점 데이터 하위 CH 번호 ＊F: 피승산 부동 소수점 데이터 하위 CH 번호 ／F: 피제산 부동 소수점 데이터 하위 CH 번호	REAL	2
S2	＋F: 가산 부동 소수점 데이터 하위 CH 번호 －F: 감산 부동 소수점 데이터 하위 CH 번호 ＊F: 승산 부동 소수점 데이터 하위 CH 번호 ／F: 제산 부동 소수점 데이터 하위 CH 번호	REAL	2
D	연산 결과 출력 하위 CH 번호	REAL	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	＊DM ＊EM		DR	IR 직접	IR 간접				
S1, S2	○	○	○	○	○	○	○	○	○	○	○	－	－	○	－	－	－	－
D											－							

+F, −F, *F, /F

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S1 또는 S2가 부동 소수점 데이터로 처리되지 않을 때 ON · +F · 피가산 데이터 또는 가산 데이터가 비수(NaN)일 때 ON · +∞와 −∞를 더하려고 한 경우 ON · −F · 피감산 데이터 또는 감산 데이터가 비수(NaN)일 때 ON · +∞와 +∞ 또는 −∞와 −∞를 빼려고 한 경우 ON · *F · 피승산 데이터 또는 승산 데이터가 비수(NaN)일 때 ON · 0과 +∞ 또는 −∞를 곱하려고 한 경우 ON · /F · 피제산 데이터 또는 제산 데이터가 비수(NaN)일 때 ON · 피제산 데이터와 제산 데이터가 모두 0 또는 +/−∞인 경우 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 클 때 ON · 그 이외일 때 OFF
언더 플로 플래그	P_UF	· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최소값보다 작을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 연산 결과가 음수가 되었을 때 ON · 그 이외일 때 OFF

기능

S1로 지정한 데이터와 S2로 지정한 데이터를 단정도 부동 소수점 데이터(32비트: IEEE754)로 가산(+F), 감산(−F), 승산(*F), 제산(/F)하여 결과를 D+1, D에 출력합니다.

· +F



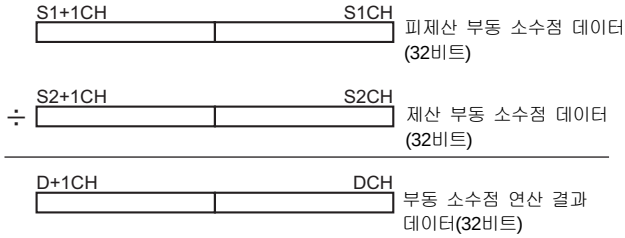
· −F



· *F



· /F



· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 큰 경우 오버 플로 플래그가 ON이 됩니다. 이 경우, 연산 결과는 ±∞로 출력됩니다.

· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최소값보다 작은 경우 언더 플로 플래그가 ON이 됩니다. 이 경우, 연산 결과는 부동 소수점 데이터의 0으로 출력됩니다.

■ 연산 규칙

연산하는 부동 소수점 데이터의 조합에 맞춰 다음 표에 따른 결과를 출력합니다.

● 부동 소수점 가산(+F)

가산 데이터 \ 피가산 데이터	0	숫자	+∞	−∞	비수(NaN)
0	0	숫자	+∞	−∞	ER
숫자	숫자	*1	+∞*2	−∞*2	
+∞	+∞	+∞*2	+∞	ER	
−∞	−∞	−∞*2	ER	−∞	
비수(NaN)					

*1: 이 경우는 0(언더 플로의 결과 UF가 0이 되는 경우 포함) 또는 숫자, +∞, −∞ 중 하나의 결과를 출력합니다.

*2: CJ1H-CPU□□H-R CPU 유니트는 불안정한 값을 출력합니다.
ER: 에러 플래그가 ON이 되고 명령이 실행되지 않습니다.

● 부동 소수점 감산(−F)

감산 데이터 \ 피감산 데이터	0	숫자	+∞	−∞	비수(NaN)
0	0	숫자	+∞	−∞	ER
숫자	숫자	*1	+∞*2	−∞*2	
+∞	−∞*2	−∞*2	ER	−∞	
−∞	+∞	+∞	+∞	ER	
비수(NaN)					

*1: 이 경우는 0(언더 플로의 결과 UF가 0이 되는 경우 포함) 또는 숫자, +∞, −∞ 중 하나의 결과를 출력합니다.

*2: CJ1H-CPU□□H-R CPU 유니트는 불안정한 값을 출력합니다.
ER: 에러 플래그가 ON이 되고 명령이 실행되지 않습니다.

● 부동 소수점 승산(*F)

승산 데이터 \ 피승산 데이터	0	숫자	+∞	−∞	비수(NaN)
0	0	0	ER	ER	ER
숫자	0	*1	+ / −∞*2	+ / −∞*2	
+∞	ER	+ / −∞*2	+∞	−∞	
−∞	ER	+ / −∞*2	−∞	+∞	
비수(NaN)					

*1: 이 경우는 0(언더 플로의 결과 UF가 0이 되는 경우 포함) 또는 숫자, +∞, −∞ 중 하나의 결과를 출력합니다.

*2: CJ1H-CPU□□H-R CPU 유니트는 불안정한 값을 출력합니다.
ER: 에러 플래그가 ON이 되고 명령이 실행되지 않습니다.

● 부동 소수점 제산(/F)

제산 데이터 \ 피제산 데이터	0	숫자	+∞	−∞	비수(NaN)
0	ER	+ / −∞*3	+∞*3	−∞*3	ER
숫자	0	*2	+ / −∞	+ / −∞	
+∞	0	0*1 *3	ER	ER	
−∞	0	0*1 *3	ER	ER	
비수(NaN)					

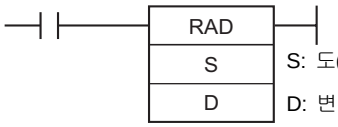
*1: 언더 플로의 결과 0이 됩니다.

*2: 이 경우는 0(언더 플로의 결과 UF가 0이 되는 경우 포함) 또는 숫자, +∞, −∞ 중 하나의 결과를 출력합니다.

*3: CJ1H-CPU□□H-R CPU 유니트는 불안정한 값을 출력합니다.
ER: 에러 플래그가 ON이 되고 명령이 실행되지 않습니다.

RAD

명령 이름	니모닉	동작 옵션	평션 번호	기능
각도 →라디안 변환	RAD	@RAD	458	지정된 부동 소수점 데이터(32비트)를 도(°) 단위에서 라디안(rad) 단위로 변환하여 결과를 지정된 채널에 출력합니다.

심볼	RAD
	 S: 도(°) 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	도(°) 데이터 하위 CH 번호	REAL	2
D	변환 결과 출력 하위 CH 번호	REAL	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
D											–							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 각도 데이터가 부동 소수점 데이터로 처리되지 않을 때 ON · 각도 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 변환 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 클 때 ON · 그 이외일 때 OFF
언더 플로 플래그	P_UF	· 변환 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최소값보다 작을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 변환 결과가 음수가 되었을 때 ON · 그 이외일 때 OFF

기능

S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)로 표현되는 각도 데이터를 도 단위에서 라디안 단위로 변환하여 결과를 D+1, D에 출력합니다.



RAD 명령은 다음 식에 기초하여 도에서 라디안으로 변환을 수행합니다.

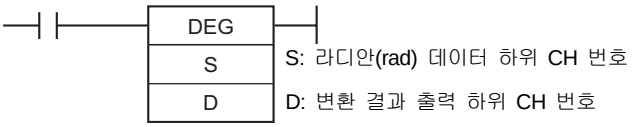
도(°)×π /180 = 라디안(rad)

주:

- 변환 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 큰 경우 오버 플로 플래그가 ON이 됩니다. 이때, 변환 결과는 ±∞로 출력됩니다.
- 변환 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최소값보다 작은 경우 언더 플로 플래그가 ON이 됩니다. 이 경우, 변환 결과는 부동 소수점 데이터의 0으로 출력됩니다.

DEG

명령 이름	니모닉	동작 옵션	평션 번호	기능
라디안→각도 변환	DEG	@DEG	459	지정된 부동 소수점 데이터(32비트)를 라디안(rad) 단위에서 도(°) 단위로 변환하여 결과를 지정된 채널에 출력합니다.

심볼	DEG
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	라디안(rad) 데이터 하위 CH 번호	REAL	2
D	변환 결과 출력 하위 CH 번호	REAL	2

■ 영역 종류

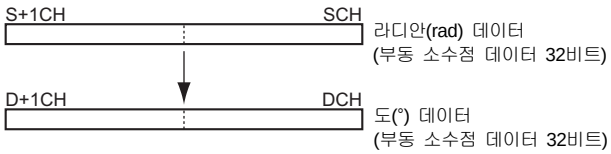
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
D											–							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 각도 데이터가 부동 소수점 데이터로 처리되지 않을 때 ON · 각도 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 변환 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 클 때 ON · 그 이외일 때 OFF
언더 플로 플래그	P_UF	· 변환 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최소값보다 작을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 변환 결과가 음수가 되었을 때 ON · 그 이외일 때 OFF

기능

S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)로 표현되는 각도 데이터를 라디안 단위에서 도 단위로 변환하여 결과를 D+1, D에 출력합니다.



DEG 명령은 다음 식에 기초하여 라디안에서 도로 변환을 수행합니다.

라디안(rad)×180/π = 도(°)

주:

- 변환 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 큰 경우 오버 플로 플래그가 ON이 됩니다. 이때, 변환 결과는 ±∞로 출력됩니다.
- 변환 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최소값보다 작은 경우 언더 플로 플래그가 ON이 됩니다. 이 경우, 변환 결과는 부동 소수점 데이터의 0으로 출력됩니다.

SIN/COS/TAN

명령 이름	니모닉	동작 옵션	평션 번호	기능
SIN 연산	SIN	@SIN	460	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 SIN(사인) 값을 연산하여 결과를 지정된 채널에 출력합니다.
COS 연산	COS	@COS	461	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 COS(코사인) 값을 연산하여 결과를 지정된 채널에 출력합니다.
TAN 연산	TAN	@TAN	462	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 TAN(탄젠트) 값을 연산하여 결과를 지정된 채널에 출력합니다.

심볼	SIN	COS	TAN
	SIN S: 각도(rad) 데이터 하위 CH 번호 D D: 변환 결과 출력 하위 CH 번호	COS S: 각도(rad) 데이터 하위 CH 번호 S D D: 변환 결과 출력 하위 CH 번호	TAN S: 각도(rad) 데이터 하위 CH 번호 S D D: 변환 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	각도(rad) 데이터 하위 CH 번호	REAL	2
D	변환 결과 출력 하위 CH 번호	REAL	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
D											–							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 각도 데이터가 부동 소수점 데이터로 처리되지 않을 때 ON · 각도 데이터가 비수(NaN)일 때 ON · 각도 데이터의 절대값이 0~65535의 범위 안에 없을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	OFF
언더 플로 플래그	P_UF	OFF
부정 플래그	P_N	· 연산 결과가 음수일 때 ON · 그 이외일 때 OFF

기능

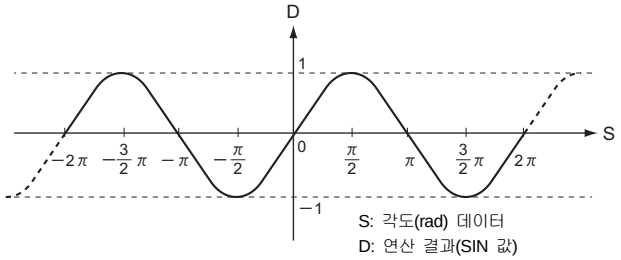
■ SIN

S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)로 표현되는 각도(라디안 단위)에 대한 SIN(사인) 값을 연산하여 결과를 D+1, D에 출력합니다.

SIN(S+1 | S) → D+1 | D

S에는 라디안(rad) 단위의 각도 데이터를 지정하십시오.
또한 도(°)와 라디안(rad)의 변환에 대해서는 RAD 명령, DEG 명령을 참조하십시오.

● 입력 데이터와 연산 결과의 관계



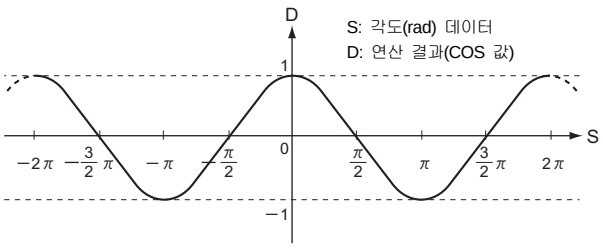
■ COS

S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)로 표현되는 각도(라디안 단위)에 대한 COS(코사인) 값을 연산하여 결과를 D+1, D에 출력합니다.

COS(S+1 | S) → D+1 | D

S에는 라디안(rad) 단위의 각도 데이터를 지정하십시오.
또한 도(°)와 라디안(rad)의 변환에 대해서는 RAD 명령, DEG 명령을 참조하십시오.

● 입력 데이터와 연산 결과의 관계



■ TAN

S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)로 표현되는 각도(라디안 단위)에 대한 TAN(탄젠트) 값을 연산하여 결과를 D+1, D에 출력합니다.

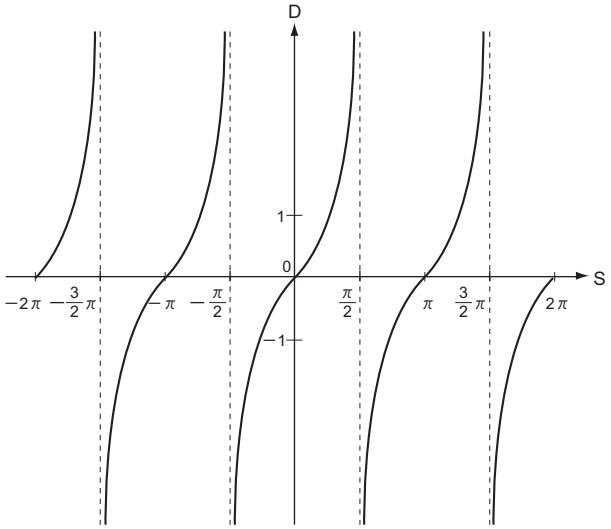
TAN(S+1 | S) → D+1 | D

S에는 라디안(rad) 단위의 각도 데이터를 지정하십시오.
또한 도(°)와 라디안(rad)의 변환에 대해서는 RAD 명령, DEG 명령을 참조하십시오.

- 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 큰 경우 오버 플로(OF)가 ON이 됩니다. 이런 경우, 연산 결과는 ±∞로 출력됩니다.

SIN/COS/TAN

● 입력 데이터와 연산 결과의 관계



S: 각도(rad) 데이터
D: 연산 결과(TAN 값)

SINQ/COSQ/TANQ

명령 이름	니모닉	동작 옵션	평션 번호	기능
고속 SIN 연산	SINQ	@SINQ	475	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 SIN(사인) 값을 연산하여 결과를 고속으로 지정된 채널에 출력합니다.
고속 COS 연산	COSQ	@COSQ	476	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 COS(코사인) 값을 연산하여 결과를 고속으로 지정된 채널에 출력합니다.
고속 TAN 연산	TANQ	@TANQ	477	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 TAN(탄젠트) 값을 연산하여 결과를 고속으로 지정된 채널에 출력합니다.

심볼	SINQ	COSQ	TANQ
	SINQ S: 각도(rad) 데이터 하위 CH 번호 D D: 변환 결과 출력 하위 CH 번호	COSQ S: 각도(rad) 데이터 하위 CH 번호 D D: 변환 결과 출력 하위 CH 번호	TANQ S: 각도(rad) 데이터 하위 CH 번호 D D: 변환 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	각도(rad) 데이터 하위 CH 번호	REAL	2
D	변환 결과 출력 하위 CH 번호	REAL	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
D											–							

기능

■ SINQ

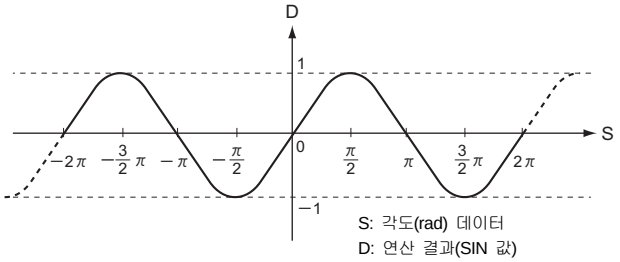
S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)로 표현되는 각도(라디안 단위)에 대한 SIN(사인) 값을 연산하여 결과를 D+1, D에 출력합니다.

SIN(S+1 | S) → D+1 | D

- S에는 라디안(rad) 단위의 각도 데이터를 지정하십시오. 또한 도(°)와 라디안(rad)의 변환에 대해서는 RAD 명령, DEG 명령을 참조하십시오.
- 각도 데이터는 −65535~65535의 범위 내에서 지정하십시오. −65535~65535의 범위 밖에서 데이터를 지정한 경우 불안정한 값이 출력됩니다. 이때, 에러 플래그는 ON되지 않습니다.

SINQ/COSQ/TANQ

● 입력 데이터와 연산 결과의 관계



S: 각도(rad) 데이터
D: 연산 결과(SIN 값)

■ COSQ

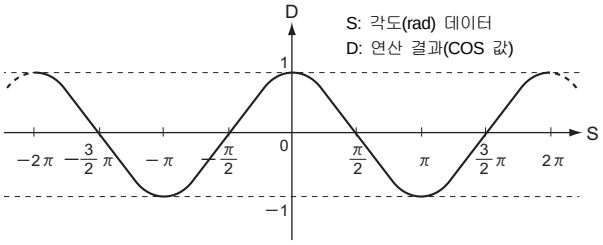
S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)로 표현되는 각도(라디안 단위)에 대한 COS(코사인) 값을 연산하여 결과를 D+1, D에 출력합니다.

COS(S+1 | S) → D+1 | D

· S에는 라디안(rad) 단위의 각도 데이터를 지정하십시오.
또한 도(°)와 라디안(rad)의 변환에 대해서는 RAD 명령, DEG 명령을 참조하십시오.

· 각도 데이터는 −65535~65535의 범위 내에서 지정하십시오. −65535~65535의 범위 밖에서 데이터를 지정한 경우 불안정한 값이 출력됩니다. 이때, 에러 플래그는 ON되지 않습니다.

● 입력 데이터와 연산 결과의 관계



■ TANQ

S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)로 표현되는 각도(라디안 단위)에 대한 TAN(탄젠트) 값을 연산하여 결과를 D+1, D에 출력합니다.

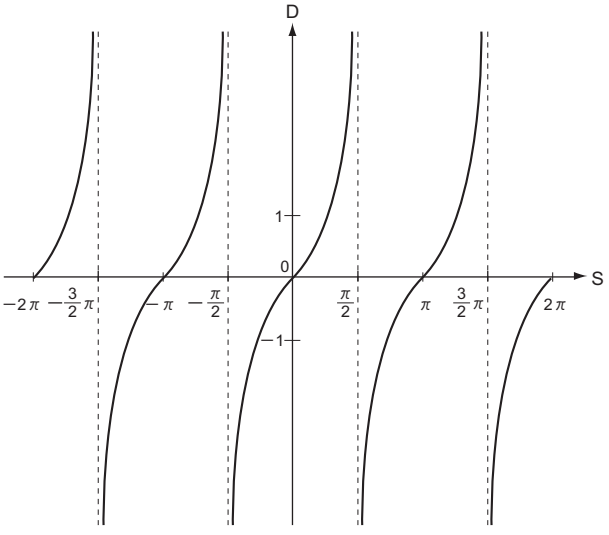
TAN(S+1 | S) → D+1 | D

· S에는 라디안(rad) 단위의 각도 데이터를 지정하십시오.
또한 도(°)와 라디안(rad)의 변환에 대해서는 RAD 명령, DEG 명령을 참조하십시오.

· 각도 데이터는 −65535~65535의 범위 내에서 지정하십시오. −65535~65535의 범위 밖에서 데이터를 지정한 경우 불안정한 값이 출력됩니다. 이때, 에러 플래그는 ON되지 않습니다.

· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 큰 경우는 ±∞ 또는 0으로 출력됩니다.

● 입력 데이터와 연산 결과의 관계



S: 각도(rad) 데이터
D: 연산 결과(TAN 값)

사용상의 주의 사항

■ SINQ, COSQ

이 명령은 SIN/COS 명령과 다음과 같은 점이 다릅니다.

- 실행 성능이 향상되었습니다.
- 명령 길이가 8스텝입니다.
- 조건 플래그를 갱신하지 않습니다.
- 각도 데이터가 범위 밖일 때는 불안정한 값을 출력합니다.
- 프로그래밍 콘솔에서 입력 및 출력할 수 없으며 대신 '?'가 표시됩니다.

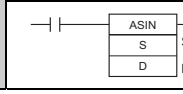
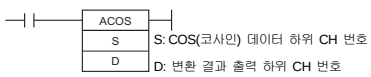
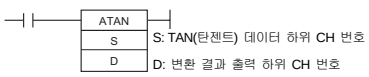
■ TANQ

이 명령은 TAN 명령과 다음과 같은 점이 다릅니다.

- 실행 성능이 향상되었습니다.
- 명령 길이가 15스텝입니다.
- 조건 플래그를 갱신하지 않습니다.
- 각도 데이터가 범위 밖일 때는 불안정한 값을 출력합니다.
- 프로그래밍 콘솔에서 입력 및 출력할 수 없으며 대신 '?'가 표시됩니다.
- $\frac{n}{2}\pi$ (n = . . . , -3, -1, 1, 3. . .)일 때는 불안정한 값을 출력합니다.

ASIN/ACOS/ATAN

명령 이름	니모닉	동작 옵션	평션 번호	기능
SIN^{-1} 연산	ASIN	@ASIN	463	지정된 부동 소수점 데이터(32비트)로 표시되는 SIN(사인) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다.
COS^{-1} 연산	ACOS	@ACOS	464	지정된 부동 소수점 데이터(32비트)로 표시되는 COS(코사인) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다.
TAN^{-1} 연산	ATAN	@ATAN	465	지정된 부동 소수점 데이터(32비트)로 표시되는 TAN(탄젠트) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다.

심볼	ASIN	ACOS	ATAN
			

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	ASIN: SIN(사인) 데이터 하위 CH 번호 ACOS: COS(코사인) 데이터 하위 CH 번호 ATAN: TAN(탄젠트) 데이터 하위 CH 번호	REAL	2
D	변환 결과 출력 하위 CH 번호	REAL	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
D											–							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S가 부동 소수점 데이터로 처리되지 않을 때 ON · ASIN · SIN 데이터가 비수(NaN)일 때 ON · SIN 데이터가 −1.0~1.0의 범위 안에 없을 때 ON · ACOS · COS 데이터가 비수(NaN)일 때 ON · COS 데이터가 −1.0~1.0의 범위 안에 없을 때 ON · ATAN · TAN 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	OFF
언더 플로 플래그	P_UF	OFF

이름	라벨	내용
부정 플래그	P_N	· ASIN, ATAN ▪ 연산 결과가 음수가 되었을 때 ON ▪ 그 이외일 때 OFF · ACOS OFF

기능

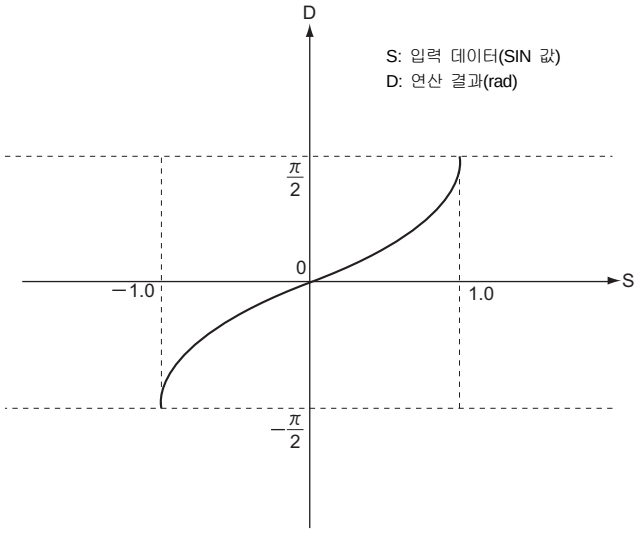
■ ASIN

S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)로 표현되는 SIN(사인) 값에 대한 각도(라디안 단위)를 연산하여 결과를 D+1, D에 출력합니다.

SIN^{-1}(S+1 : S) → D+1 : D

연산 결과는 $-\pi/2$~$\pi/2$의 범위 내의 각도(라디안 단위) 데이터로 D+1, D CH에 출력됩니다.

● 입력 데이터와 연산 결과의 관계



■ ACOS

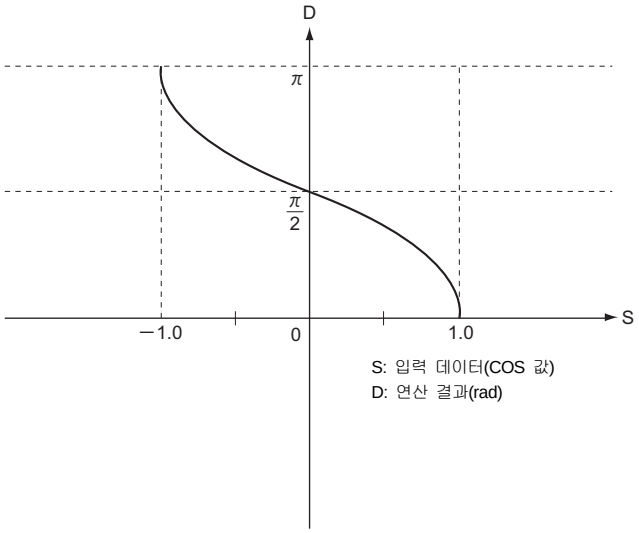
S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)로 표현되는 COS(코사인) 값에 대한 각도(라디안 단위)를 연산하여 결과를 D+1, D에 출력합니다.

COS^{-1}(S+1 : S) → D+1 : D

연산 결과는 0~π의 범위 내의 각도(라디안 단위) 데이터로 D+1, D CH에 출력됩니다.

ASIN/ACOS/ATAN

● 입력 데이터와 연산 결과의 관계



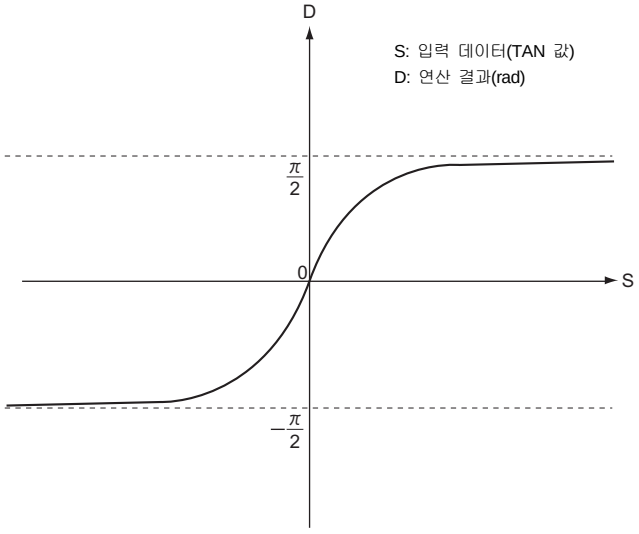
■ ATAN

S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)로 표현되는 TAN(탄젠트) 값에 대한 각도(라디안 단위)를 연산하여 결과를 D+1, D에 출력합니다.

TAN⁻¹(S+1 | S) → D+1 | D

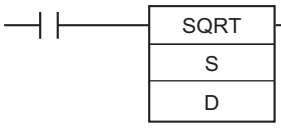
연산 결과는 −π/2~π/2의 범위 내의 각도(라디안 단위) 데이터로 D+1, D CH에 출력됩니다.

● 입력 데이터와 연산 결과의 관계



SQRT

명령 이름	니모닉	동작 옵션	평션 번호	기능
제곱근 연산	SQRT	@SQRT	466	지정된 부동 소수점 데이터(32비트)의 제곱근(√)을 연산하여 결과를 지정된 채널에 출력합니다.

심볼	SQRT
	 S: 입력 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	입력 데이터 하위 CH 번호	REAL	2
D	연산 결과 출력 하위 CH 번호	REAL	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
D											–							

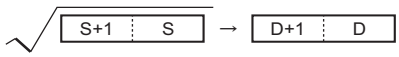
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 입력 데이터가 부동 소수점 데이터로 처리되지 않을 때 ON · 입력 데이터가 음수일 때 ON · 입력 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 클 때 ON · 그 이외일 때 OFF
언더 플로 플래그	P_UF	OFF
부정 플래그	P_N	OFF

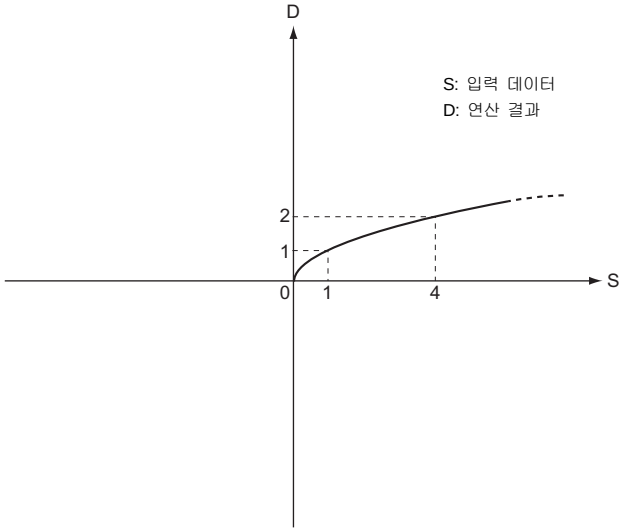
SQRT

기능

S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)로 표현되는 입력 데이터에 대한 제곱근을 연산하여 결과를 D+1, D에 출력합니다.



■ 입력 데이터와 연산 결과의 관계



EXP

명령 이름	니모닉	동작 옵션	펑션 번호	기능
지수 연산	EXP	@EXP	467	지정된 부동 소수점 데이터(32비트)의 지수(e를 밑수로 했을 때의 지수)를 연산하여 결과를 지정된 채널에 출력합니다.

심볼	EXP
	 S: 입력 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	입력 데이터 하위 CH 번호	REAL	2
D	연산 결과 출력 하위 CH 번호	REAL	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
D											–							

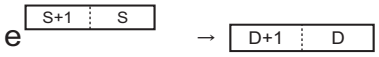
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 입력 데이터가 부동 소수점 데이터로 처리되지 않을 때 ON · 입력 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 클 때 ON · 그 이외일 때 OFF
언더 플로 플래그	P_UF	· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최소값보다 작을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	OFF

EXP

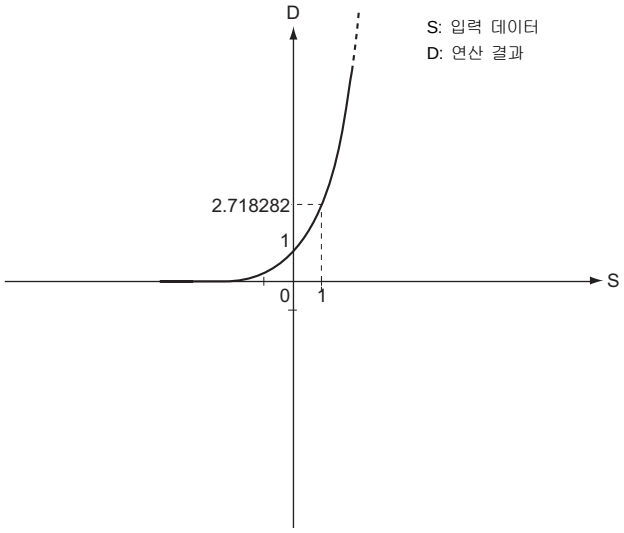
기능

S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)로 표현되는 입력 데이터에 대한 지수(e를 밑수로 했을 때의 지수)를 연산하여 결과를 D+1, D에 출력합니다.



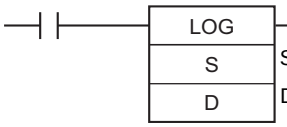
EXP 명령은 밑수(e)를 2.718282로 하여 지수 연산합니다.

■ 입력 데이터와 연산 결과의 관계



LOG

명령 이름	니모닉	동작 옵션	펑션 번호	기능
로그 연산	LOG	@LOG	468	지정된 부동 소수점 데이터(32비트)의 자연 로그(e를 밑수로 했을 때의 로그)를 연산하여 결과를 지정된 채널에 출력합니다.

심볼	LOG
	 S: 입력 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	입력 데이터 하위 CH 번호	REAL	2
D	연산 결과 출력 하위 CH 번호	REAL	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
D											–							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 입력 데이터가 부동 소수점 데이터로 처리되지 않을 때 ON · 입력 데이터가 음수일 때 ON · 입력 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 클 때 ON · 그 이외일 때 OFF
언더 플로 플래그	P_UF	OFF
부정 플래그	P_N	· 연산 결과가 음수가 되었을 때 ON · 그 이외일 때 OFF

LOG

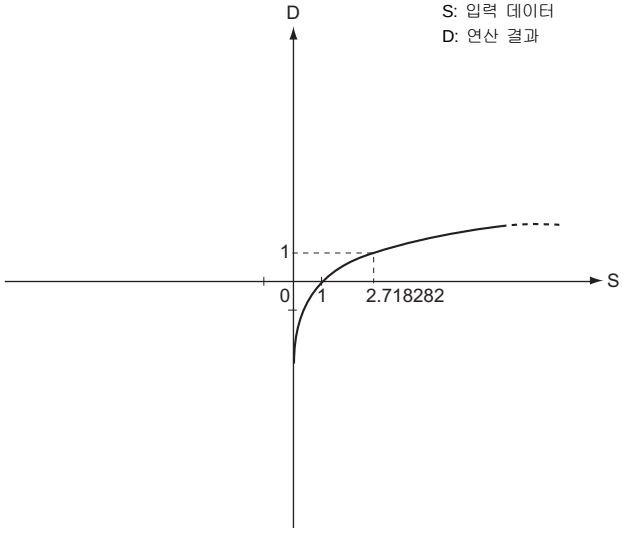
기능

S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)로 표현되는 입력 데이터에 대한 자연 로그(e를 밑수로 했을 때의 로그)를 연산하여 결과를 D+1, D에 출력합니다.

$\log_e$ [S+1 | S] → [D+1 | D]

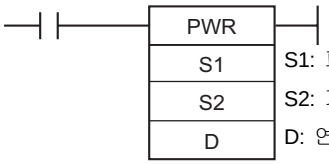
LOG 명령은 밑수(e)를 2.718282로 하여 지수 연산합니다.

■ 입력 데이터와 연산 결과의 관계



PWR

명령 이름	니모닉	동작 옵션	펑션 번호	기능
거듭제곱 연산	PWR	@PWR	840	지정된 부동 소수점 데이터(32비트)의 거듭제곱 연산을 수행하여 결과를 지정된 채널에 출력합니다.

심볼	PWR
	 S1: 피거듭제곱 데이터 하위 CH 번호 S2: 거듭제곱 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S1	피거듭제곱 데이터 하위 CH 번호	REAL	2
S2	거듭제곱 데이터 하위 CH 번호	REAL	2
D	연산 결과 출력 하위 CH 번호	REAL	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S1, S2	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
D											–							

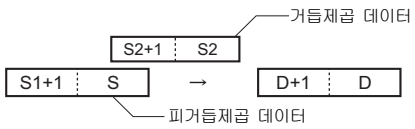
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 피거듭제곱 데이터 또는 거듭제곱 데이터가 부동 소수점 데이터로 처리되지 않을 때 ON · 피거듭제곱 데이터 또는 거듭제곱 데이터가 비수(NaN)일 때 ON · 피거듭제곱 데이터가 0이고 거듭제곱 데이터가 0 이하일 때 ON · 피거듭제곱 데이터가 음수이고 거듭제곱 데이터가 정수가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 클 때 ON · 그 이외일 때 OFF
언더 플로 플래그	P_UF	· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최소값보다 작을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 연산 결과가 음수가 되었을 때 ON · 그 이외일 때 OFF

PWR

기능

S1로 지정한 데이터와 S2로 지정한 데이터를 단정도 부동 소수점 데이터(32비트: IEEE754)로 거듭제곱 연산하여 결과를 D+1, D에 출력합니다.



예: 부동 소수점 데이터 연산 결과

S1: 3.1 → 3.1^3 → D: 29.791

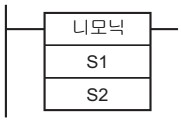
S2: 3.0

=F, <>F, <F, <=F, >F, >=F

명령 이름	니모닉	동작 옵션	펑션 번호	기능
단정도 부동 소수점 데이터 비교	=F <>F <F <=F >F >=F	–	329 330 331 332 333 334	지정된 단정도 부동 소수점 데이터(32비트) 또는 상수를 비교하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다.

심볼	단정도 부동 소수점 데이터 비교

LD(로드) 연결형



AND(직렬) 연결형



OR(병렬) 연결형



특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S1	비교 데이터 1	REAL	2
S2	비교 데이터 2	REAL	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S1	○	○	○	○	○	○	○	○	○	○	○	–	○	○	–	–	–	–
S2																		

조건 플래그의 동작

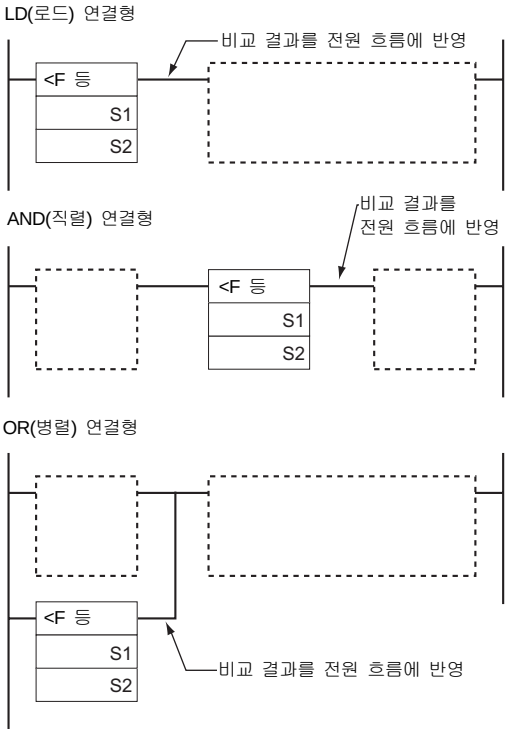
이름	라벨	내용
에러 플래그	P_ER	OFF
> 플래그	P_GT	· 비교 결과가 S1+1, S1 > S2+1, S2일 때 ON · 그 이외일 때 OFF
≧ 플래그	P_GE	· 비교 결과가 S1+1, S1≧S2+1, S2일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 비교 결과가 S1+1, S1 = S2+1, S2일 때 ON · 그 이외일 때 OFF
≠ 플래그	P_NE	· 비교 결과가 S1+1, S1≠ S2+1, S2일 때 ON · 그 이외일 때 OFF
< 플래그	P_LT	· 비교 결과가 S1+1, S1 < S2+1, S2일 때 ON · 그 이외일 때 OFF
≦ 플래그	P_LE	· 비교 결과가 S1+1, S1≦S2+1, S2일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	유지

=F, <>F, <F, <=F, >F, >=F

기능

S1로 지정한 데이터와 S2로 지정한 데이터를 단정도 부동 소수점 데이터(32비트: IEEE754)로 비교하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다.
비교 데이터 S1, S2를 지정하려면 데이터가 저장된 채널의 시작 주소를 피연산자 S1, S2에 입력합니다.
또한 비교 데이터에 상수를 사용하는 경우는 S1, S2에 16진수 8자리 숫자를 직접 입력합니다.

LD, AND, OR 명령과 동일하게 취급하며, 각 명령 후에는 계속하여 다른 명령을 프로그래밍합니다.
LD형의 경우: 모선에 직접 연결할 수 있습니다.
AND형의 경우: 모선에 직접 연결할 수 없습니다.
OR형의 경우: 모선에 직접 연결할 수 있습니다.



■ 심볼과 옵션

비교 명령은 심볼과 옵션을 조합하여 18종류의 니모닉으로 표현됩니다.

심볼 (래더에서는 LD, AND, OR 없음)		옵션 (데이터 형식)
LD=, AND=, OR=, LD<>, AND<>, OR<>, LD<, AND<, OR<, LD<=, AND<=, OR<=, LD>, AND>, OR>, LD>=, AND>=, OR>=	+	F(단정도 부동 소수점)

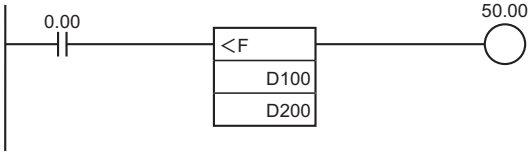
기능	니모닉	이름	펑션 번호
S1 = S2일 때 참(ON)	LD = F	LD 부동 소수점 데이터 일치	329
	AND = F	AND 부동 소수점 데이터 일치	329
	OR = F	OR 부동 소수점 데이터 일치	329
S1≠ S2일 때 참(ON)	LD <> F	LD 부동 소수점 데이터 불일치	330
	AND <> F	AND 부동 소수점 데이터 불일치	330
	OR <> F	OR 부동 소수점 데이터 불일치	330
S1 < S2일 때 참(ON)	LD < F	LD 부동 소수점 데이터 미만	331
	AND < F	AND 부동 소수점 데이터 미만	331
	OR < F	OR 부동 소수점 데이터 미만	331
S1≦S2일 때 참(ON)	LD <= F	LD 부동 소수점 데이터 이하	332
	AND <= F	AND 부동 소수점 데이터 이하	332
	OR <= F	OR 부동 소수점 데이터 이하	332
S1 > S2일 때 참(ON)	LD > F	LD 부동 소수점 데이터 초과	333
	AND > F	AND 부동 소수점 데이터 초과	333
	OR > F	OR 부동 소수점 데이터 초과	333
S1≧S2일 때 참(ON)	LD >= F	LD 부동 소수점 데이터 이상	334
	AND>= F	AND 부동 소수점 데이터 이상	334
	OR >= F	OR 부동 소수점 데이터 이상	334

사용상의 주의 사항

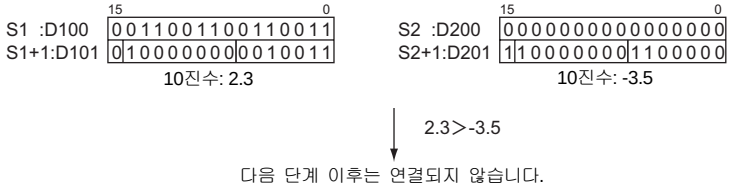
- 이 명령의 최종 단계에는 출력계 명령(OUT계 명령 및 다음 단계 연결형 명령 이외의 응용 명령)을 추가하십시오.
- 회로의 최종 단계에는 이 명령을 사용할 수 없습니다.

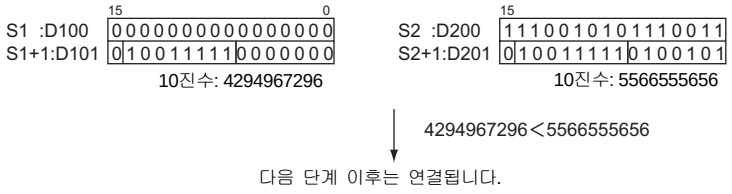
샘플 프로그램

0.00이 ON일 때 데이터 메모리 D100, 101에 저장된 부동 소수점 데이터와 D200, 201에 저장된 부동 소수점 데이터를 비교합니다.
비교 결과가 (D100, 101의 데이터) < (D200, 201의 데이터)인 경우, 다음 단계 이후가 연결되어 출력 릴레이 50.00이 ON됩니다.
비교 결과가 (D100, 101의 데이터) < (D200, 201의 데이터)가 아닌 경우는 다음 단계 이후로 연결되지 않습니다.



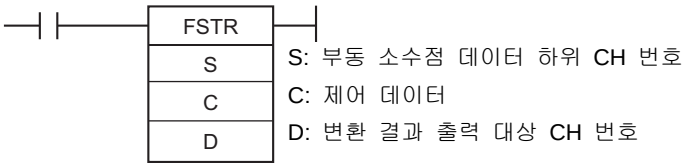
단정도 부동 소수점 비교(<F)





FSTR

명령 이름	니모닉	동작 옵션	평션 번호	기능
부동 소수점<단>→ 문자열 변환	FSTR	@FSTR	448	지정된 단정도 부동 소수점 데이터(32비트)를 소수점 형식 또는 지수 형식으로 표시하고 그것을 문자열 데이터(ASCII 코드)로 변환합니다.

심볼	FSTR
	

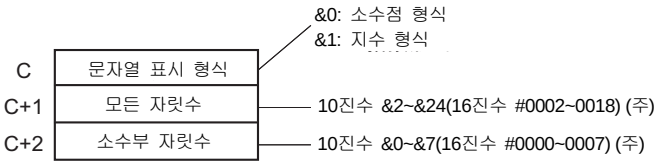
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	부동 소수점 데이터 하위 CH 번호	REAL	2
C	제어 데이터	UINT	3
D	변환 결과 출력 대상 CH 번호	UINT	가변

C: 제어 데이터



주: 모든 자릿수, 소수부 자릿수에는 제한이 있습니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
C											–							
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S+1, S의 데이터가 비수(NaN)일 때 ON · S+1, S의 데이터가 +/-∞일 때 ON · C로 지정하는 문자열 표시 형식 지정이 0000, 0001 이외일 때 ON · C+1의 전체 자릿수가 범위 밖('변환 후 문자열의 자릿수 제한' 참조)일 때 ON · C+2로 지정하는 소수부 자릿수가 범위 밖('변환 후 문자열의 자릿수 제한' 참조)일 때 ON
= 플래그	P_EQ	· 변환 결과가 "0"일 때 ON · 그 이외일 때 OFF

기능

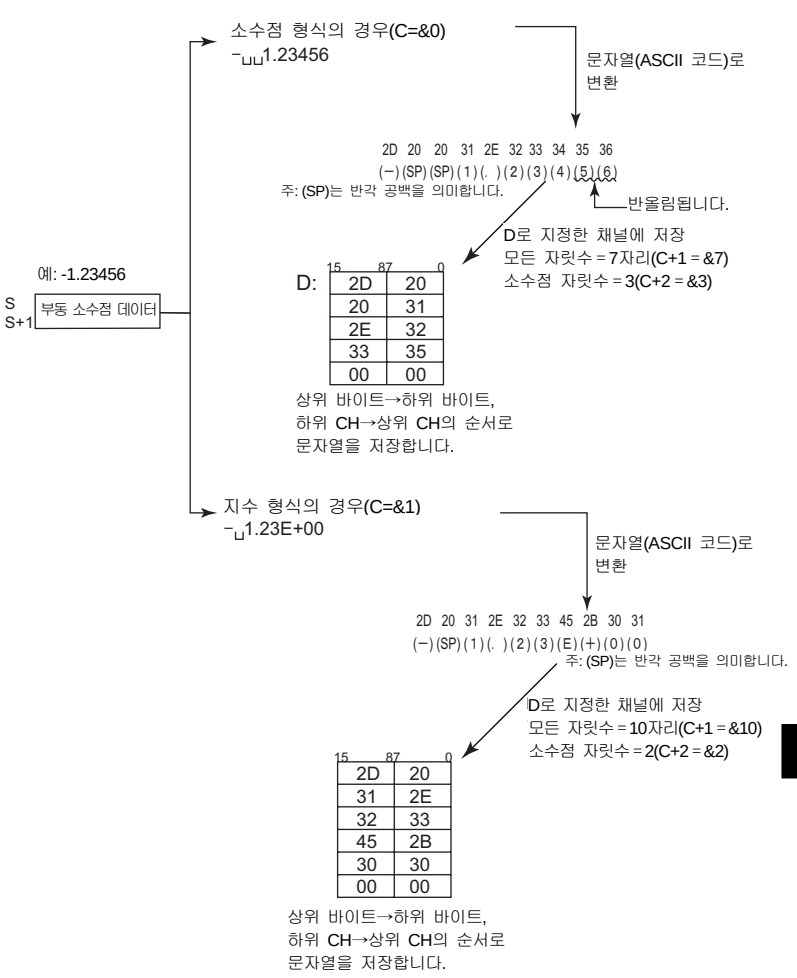
S로 지정한 단정도 부동 소수점 데이터(32 비트: IEEE754)를 C~C+2의 내용에 따라 소수점 형식 또는 지수 형식으로 나타내고, 그것을 문자열(ASCII 코드)로 변환한 후 D로 지정된 채널에 결과를 출력합니다.

· C(변환 형식)를 사용하여, S+1, S의 부동 소수점 데이터를 소수점 형식/지수 형식 중 어떤 형식으로 나타낼지 지정합니다.
  · 소수점 형식
    실수를 정수부와 소수부로 표현하는 형식입니다.
    예) 124.56
  · 지수 형식
    실수를 정수부, 소수부 및 지수부로 표현하는 형식입니다.
    예) 1.2456E-2(1.2456×10-2)

· C+1(전체 자릿수)을 사용하여, 변환 후 문자열의 수(부호, 숫자, 소수점, 반각 공백 문자 포함)를 지정합니다.

· C+2(소수부 자릿수)를 사용하여, 변환 후 문자열의 소수부 자릿수(문자 수)를 지정합니다.

D 이후의 저장 순서는 D의 상위 바이트→D의 하위 바이트→D+1의 상위 바이트→D+1의 하위 바이트→‥‥(이하 동일) 순서가 됩니다.

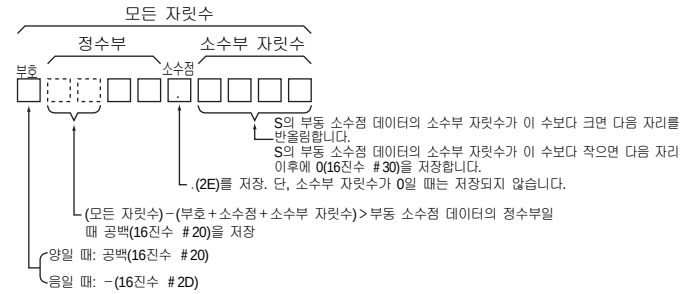


FSTR

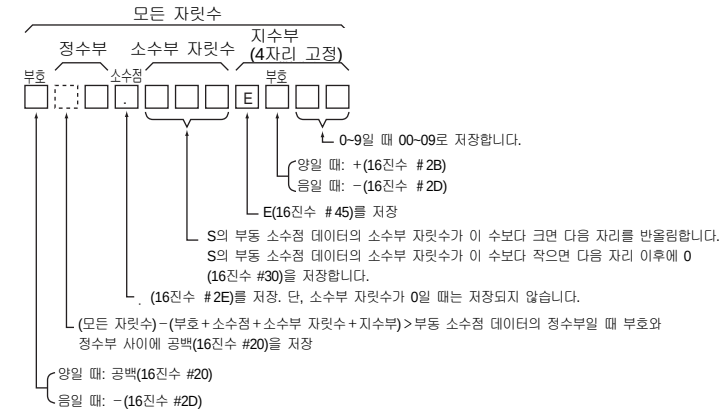
■ 변환 후 문자열 저장에 대한 자세한 내용

변환 후의 문자열은 오른쪽 그림과 같이 D에 저장됩니다.
또한 소수점 형식과 지수 형식의 경우는 문자열 저장 방법이 다릅니다.

소수점 형식의 경우(C=0)



지수 형식의 경우(C=16진수 #1)



주: 변환 후 문자열의 마지막 처리(종료 코드)는 다음과 같이 저장됩니다.
- 모든 자릿수가 홀수인 경우: 마지막 문자열 뒤에 "00Hex"를 저장합니다.
- 모든 자릿수가 짝수인 경우: 마지막 문자열 뒤에 "0000Hex"를 저장합니다.

■ 변환 후 문자열의 자릿수 제한

변환 후 문자열의 자릿수(문자 수)에는 제한이 있습니다. 또한 제한에 맞지 않는 경우 에러 플래그가 ON이 됩니다.

· 전체 자릿수(모든 문자 수)에 대한 제한

1) 변환 후의 문자열이 소수점 형식(C=0)인 경우
   - 소수부 자릿수가 0인 경우: 2≦전체 자릿수≦24
   - 소수부 자릿수가 0 이외인 경우: (소수부 자릿수+3)≦전체 자릿수≦24

2) 변환 후의 문자열이 지수 형식(C=1)인 경우
   - 소수부 자릿수가 0인 경우: 6≦전체 자릿수≦24
   - 소수부 자릿수가 0 이외인 경우: (소수부 자릿수+7)≦전체 자릿수≦24

· 정수부 자릿수(정수부 문자 수)의 제한

1) 변환 후의 문자열이 소수점 형식(C=0)인 경우
   - 소수부 자릿수가 0인 경우: 1≦정수부 자릿수≦24
   - 소수부 자릿수가 0 이외인 경우: 1≦정수부 자릿수≦24−소수부 자릿수−2

2) 변환 후의 문자열이 지수 형식(C=1)인 경우
   1(고정)

· 소수부 자릿수에 대한 제한

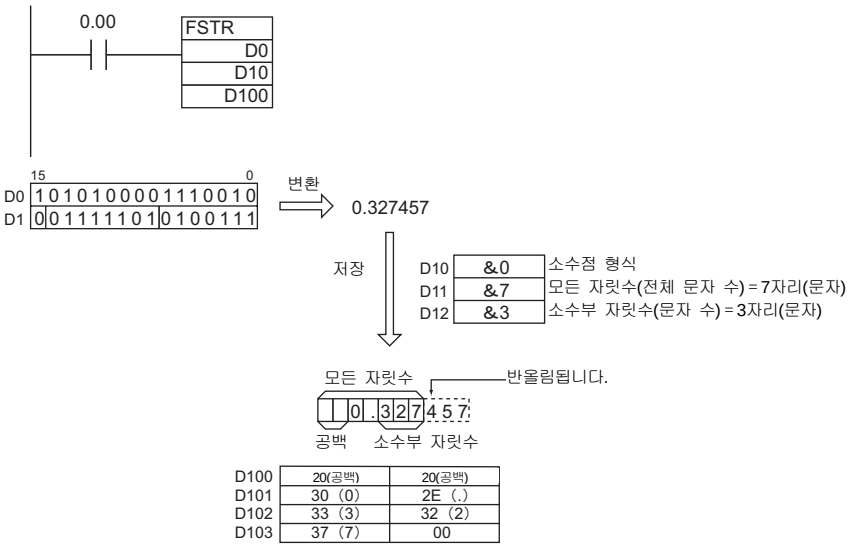
1) 변환 후의 문자열이 소수점 형식(C=0)인 경우
   소수부 자릿수≦7이고 (소수부 자릿수)≦(전체 자릿수−3)

2) 변환 후의 문자열이 지수 형식(C=1)인 경우
   소수부 자릿수≦7이고 (소수부 자릿수)≦(전체 자릿수−7)

샘플 프로그램

■ 부동 소수점 데이터를 소수점 형식의 문자열로 변환

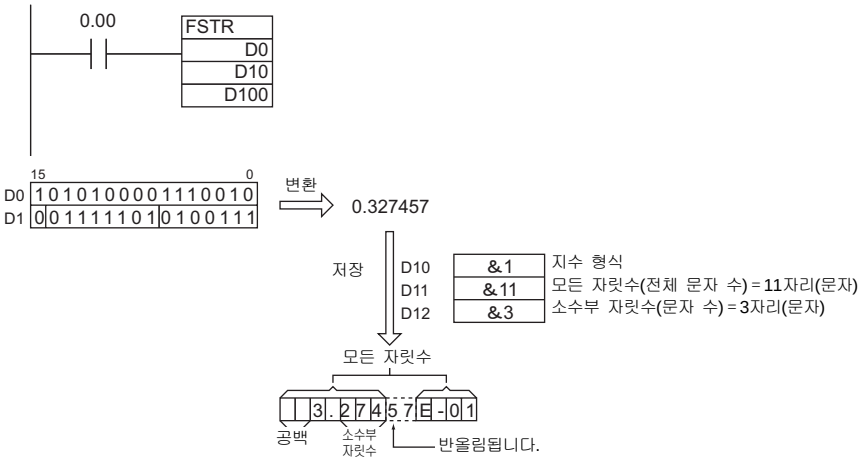
0.00이 ON되면 D1, D0에 저장된 부동 소수점 데이터를 소수점 형식의 문자열로 변환하고, D10 이후에 저장된 제어 데이터의 내용(소수점 형식, 전체 자릿수 = 7자리, 소수부 자릿수 = 3자리)에 따라, 변환된 문자열을 D100 이후에 저장합니다.



D100	20(공백)	20(공백)
D101	30 (0)	2E (.)
D102	33 (3)	32 (2)
D103	37 (7)	00

■ 부동 소수점 데이터를 지수 형식의 문자열로 변환

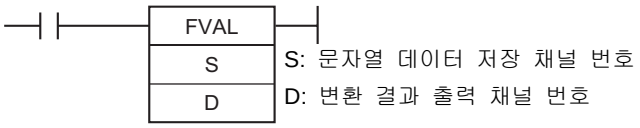
0.00이 ON되면 D0에 저장된 부동 소수점 데이터를 지수 형식의 문자열로 변환하고, D10 이후에 저장된 제어 코드의 내용(지수 형식, 전체 자릿수 = 11자리, 소수부 자릿수 = 3자리)에 따라, 변환된 문자열을 D100 이후에 저장합니다.



D100	20(공백)	20(공백)
D101	33 (3)	2E (.)
D102	32 (2)	37 (7)
D103	35 (5)	45 (E)
D104	2D (-)	30 (0)
D105	31 (1)	00

FVAL

명령 이름	니모닉	동작 옵션	평션 번호	기능
문자열→부동 소수점<단> 변환	FVAL	@FVAL	449	지정된 소수점 형식 또는 지수 형식으로 나타낸 문자열(ASCII 코드)을 단정도 부동 소수점 데이터(32비트)로 변환하여 지정된 채널에 결과를 출력합니다.

심볼	FVAL
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	문자열 데이터 저장 채널 번호	UINT	가변
D	변환 결과 출력 채널 번호	REAL	2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S로 지정한 문자열 데이터의 정수부, 소수부에 #30~39(문자 0~9) 이외의 문자가 있을 때 ON · S로 지정한 문자열 데이터의 지수부에 “#45(문자 E), #2B(문자 +)”, “#45(문자 E), #2D(문자 −)” 이외의 문자가 있을 때 ON · S로 지정한 문자열에 지수부가 여러 개 있을 때 ON · 변환 후의 데이터가 +/-∞일 때 ON · S로 지정한 문자열 데이터의 문자 수가 0일 때 ON · S로 지정한 문자열 데이터의 25문자 이내에 #00이 없을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과가 “0”일 때 ON · 그 이외일 때 OFF

기능

S로 지정한 CH에 저장된 문자열 데이터(ASCII 코드)를 단정도 부동 소수점 데이터(32비트: IEEE754)로 변환하여 D로 지정된 채널에 결과를 출력합니다.

변환 대상이 되는 문자열 데이터는 다음 중 하나의 형식으로 표현된 ASCII 코드입니다.

부호, 소수점, 지수부를 제외하고 6문자까지가 유효합니다. 6번째 문자 이후는 무시됩니다.

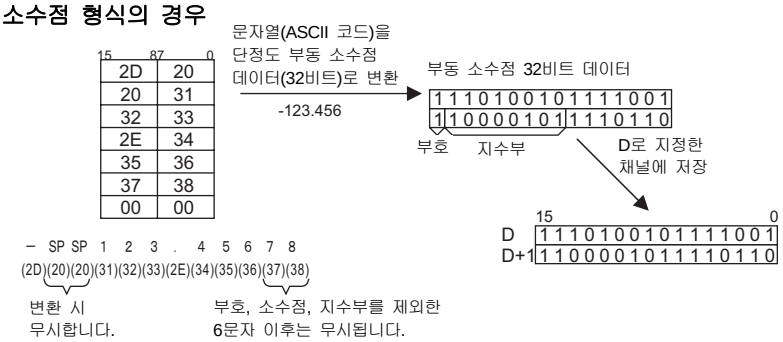
· 소수점 형식
실수를 정수부와 소수부로 표현하는 형식입니다.
예) 124.56

· 지수 형식
실수를 정수부와 소수부 및 지수부로 표현하는 형식입니다.
예) 1.2456E-2(1.2456×10-2)

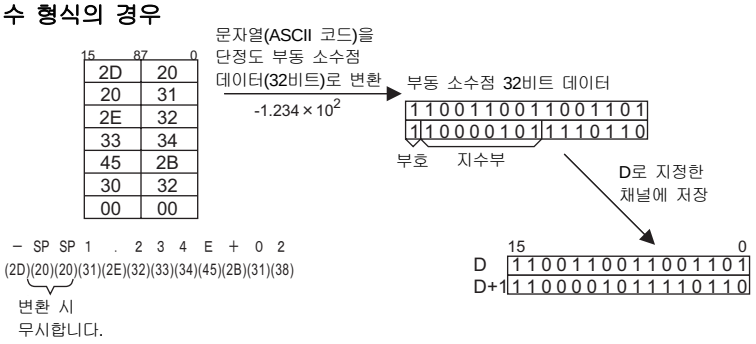
또한 저장된 문자열 데이터의 형식은 자동 판별됩니다.

S 이후의 문자열 저장 순서는 S의 상위 바이트→S의 하위 바이트→S+1의 상위 바이트→S+1의 하위 바이트→····(이하 동일) 순서입니다.

소수점 형식의 경우



지수 형식의 경우



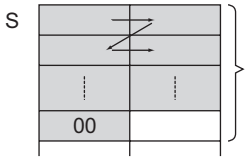
■ 문자열 저장에 대한 자세한 내용

문자열은 다음 표와 같이 변환됩니다.

또한 소수점 형식과 지수 형식의 경우는 변환 조건이 달라집니다.

FVAL

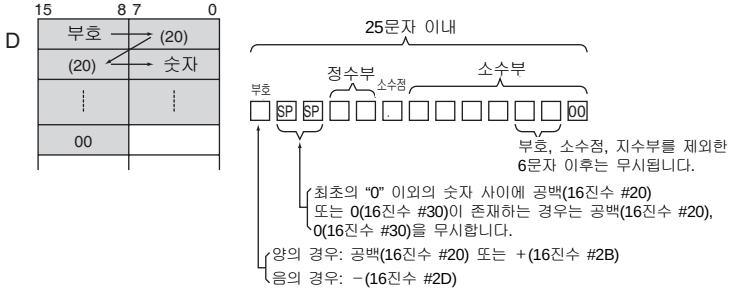
변환 대상



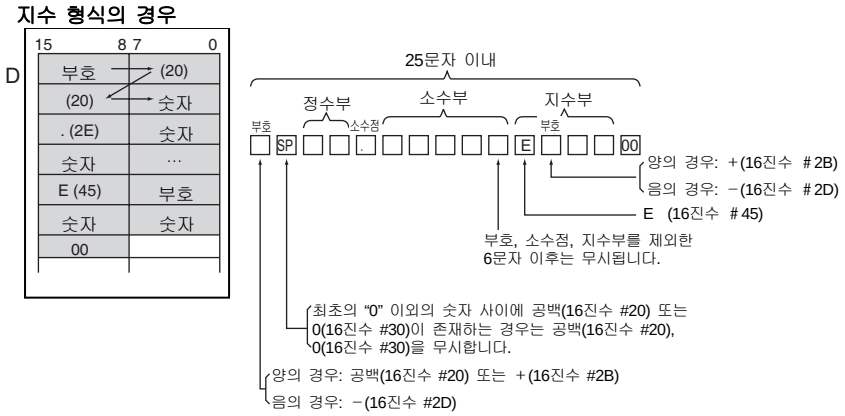
S의 상위 바이트에서 NUL 문자(16진수 #00)까지의 문자열을 부동 소수점 데이터로 변환합니다.
단, NUL 문자(16진수 #00)를 포함하여 25문자 이내이어야 합니다.

NUL 문자(16진수 #00)까지(25문자 이내)

소수점 형식의 경우



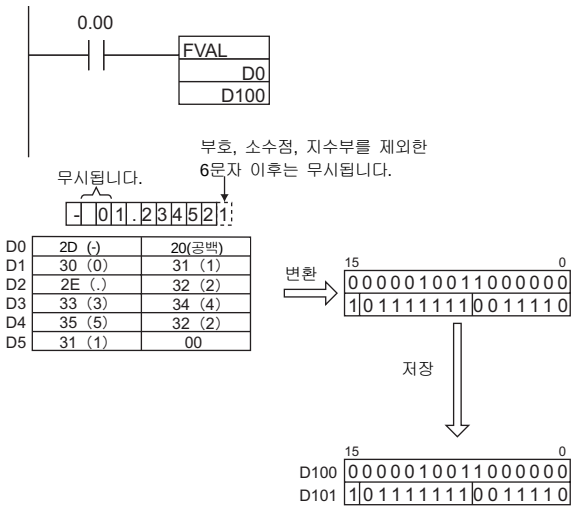
지수 형식의 경우



샘플 프로그램

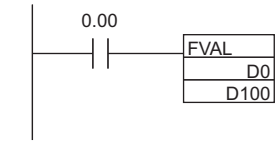
■ 소수점 형식의 문자열을 부동 소수점 데이터로 변환하는 경우

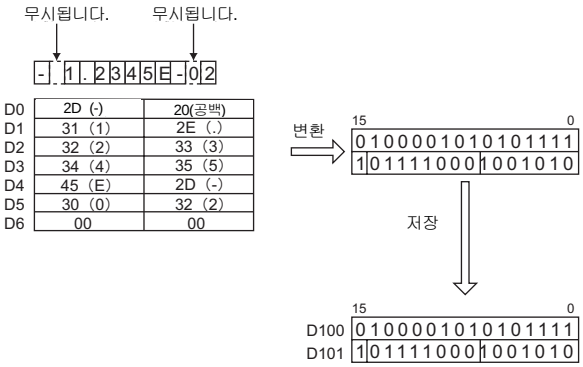
0.00이 ON되면 D0 이후에 저장된 소수점 형식의 문자열을 부동 소수점 데이터로 변환하고, 변환된 부동 소수점 데이터를 D100 이후에 저장합니다.



■ 지수 형식의 문자열을 부동 소수점 데이터로 변환하는 경우

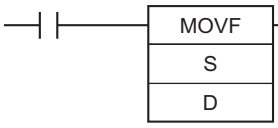
0.00이 ON되면 D0에 저장된 지수 형식의 문자열을 부동 소수점 데이터로 변환하고, 변환된 부동 소수점 데이터를 D100 이후에 저장합니다.





MOVF

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부동 소수점 전송<단>	MOVF	@MOVF	469	지정된 부동 소수점 데이터(32비트)를 지정 채널로 전송합니다.

심볼	MOVF
	 S: 전송 데이터 하위 CH 번호 D: 전송 대상 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	전송 데이터 하위 CH 번호	REAL	2
D	전송 대상 하위 CH 번호	REAL	2

■ 영역 종류

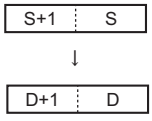
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
D											–							

조건 플래그의 동작

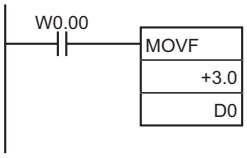
이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 전송 데이터가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 전송 데이터가 음수일 때 ON · 그 이외일 때 OFF

기능

S로 지정한 단정도 부동 소수점 데이터(32비트: IEEE754)를 D+1, D에 출력합니다.



샘플 프로그램



접점 W0.00을 ON하면 D0~D1에 '+3.0'을 부동 소수점 형식(IEEE754 규격)에 따라 저장합니다.

D1	D0
#4040	#0000

부동 소수점 형식에서 '+3.0'을 의미합니다.

배정도 부동 소수점 변환 및 연산 명령

니모닉	명령 이름	펑션 번호	페이지
FIXD	부동 소수점→16비트 BIN 변환<배>	841	3-396
FIXLD	부동 소수점→32비트 BIN 변환<배>	842	
DBL	16비트 BIN→부동 소수점 변환<배>	843	3-398
DBLL	32비트 BIN→부동 소수점 변환<배>	844	
＋D	부동 소수점 가산<배>	845	3-400
－D	부동 소수점 감산<배>	846	
＊D	부동 소수점 승산<배>	847	
/D	부동 소수점 제산<배>	848	
RADD	각도 →라디안 변환<배>	849	3-403
DEGD	라디안→각도 변환<배>	850	3-405
SIND	SIN 연산<배>	851	3-407
COSD	COS 연산<배>	852	
TAND	TAN 연산<배>	853	
ASIND	SIN^{-1} 연산<배>	854	3-410
ACOSD	COS^{-1} 연산<배>	855	
ATAND	TAN^{-1} 연산<배>	856	
SQRTD	제곱근 연산<배>	857	3-413
EXPD	지수 연산<배>	858	3-415
LOGD	로그 연산<배>	859	3-417
PWRD	거듭제곱 연산<배>	860	3-419
=D,<>D,<D,<=D, >D,>=D	배정도 부동 소수점 데이터 비교	335~340	3-421

배정도 부동 소수점 변환 및 연산 명령

■ 데이터 형식

배정도 부동 소수점 데이터는 실수를 부호, 가수, 지수로 표현한 데이터입니다.

또한 임의의 데이터를 부동 소수점 형식으로 나타내면 다음과 같습니다.

실수값 = $(-1)^s \, 2^{e-1023} \times (1.f)$

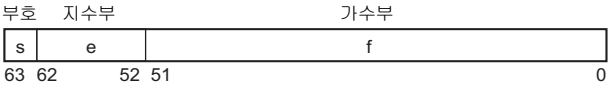
s: 부호

e: 지수부

f: 가수부

여기서, (1.f)는 가수를 정규화하여 1의 자리가 1이고, 나머지 소수부가 f임을 나타냅니다.

CS/CJ 시리즈는 IEEE754 규격의 배정도를 준수한 배정도 부동 소수점 데이터 형식을 사용합니다.
배정도 데이터는 64비트로 표현되며 형식은 다음과 같습니다.



데이터	비트 수	내용
s: 부호	1	0: 양, 1: 음
e: 지수부	11	지수부 e의 값은 0~2047입니다. e에서 1023을 뺀 값이 실제 지수가 됩니다. 따라서 지수는 −1023~1024가 됩니다. e = 0, e = 1024일 때는 특수 숫자를 나타냅니다.
f: 가수부	52	2진수에서 부동 소수점의 가수부를 나타냅니다. 또한 가수를 정규화했을 때 가수의 범위는 1.0≦1.f < 2.0이 됩니다.

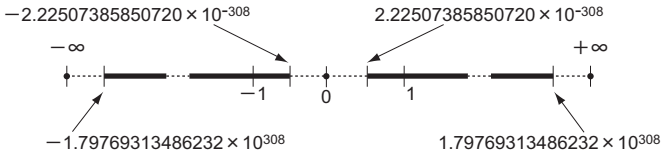
■ 유효 자릿수

배정도 부동 소수점 데이터의 유효 자릿수는 15자리입니다.

■ 배정도 부동 소수점 데이터에서 표현 방법

・ 표현 가능한 수

- $-\infty$
- $-1.79769313486232 \times 10^{308}$≦값≦$-2.22507385850720 \times 10^{-308}$
- 0
- $+2.22507385850720 \times 10^{-308}$≦값≦$+1.79769313486232 \times 10^{308}$
- $+\infty$
- 비수(NaN)



・ 특수 숫자

비수(NaN), ±∞, 0의 형식은 다음과 같이 정의합니다.

・ 비수(NaN)	e = 2047, f ≠ 0
・ $+\infty$	e = 2047, f = 0, s = 0
・ $-\infty$	e = 2047, f = 0, s = 1
・ 0	e = 0, f = 0

비수(NaN)란 부동 소수점 데이터로 처리되지 않는 값입니다.

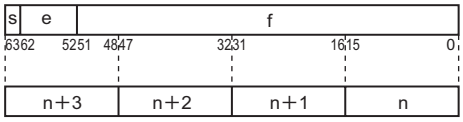
단, 배정도 부동 소수점 및 연산 명령의 결과가 비수(NaN)가 되지는 않습니다.

■ 배정도 부동 소수점 데이터 형식의 저장 방법

주변 툴 CX-Programmer의 I/O 메모리 편집 화면에서 데이터 형식을 '2워드 부동 소수점'으로 지정하면 입력된 소수점 데이터가 위의 IEEE754 규격의 데이터 형식으로 자동 변환되어 I/O 메모리에 저장됩니다. 또한 위의 IEEE754 규격의 데이터 형식으로 저장된 데이터는 자동으로 소수점 데이터로 모니터링됩니다.

따라서 사용자가 앞에서 설명한 IEEE754 규격의 데이터 형식을 고려할 필요는 없습니다.

배정도 부동 소수점 데이터가 4CH를 점유한다는 점만 고려하십시오.



■ 참고

부동 소수점 수로 표현하는 값의 종류

지수부(e) / 가수부(f)	0	0도 모든 비트 1(1024)도 아님	모든 비트 1(1024)
0	0	정규화 수	무한대
0 이외	비정규화 수		비수(NaN)

주: 정규화 수는 정규화 수로 표현할 수 없는 범위의 절대값이 작은 부동 소수점 수를 표현하지만 정규화 수에 비해 유효 자릿수가 적어집니다. 따라서 연산 결과가 비정규화 수가 되는 경우 또는 중간 결과가 비정규화 수가 되는 경우, 결과의 유효 자릿수는 보장되지 않으므로 주의하십시오.

(1) 정규화 수

일반적인 실수값을 표현합니다.

부호부는 0(양) 또는 1(음)로 값의 부호를 나타냅니다.

지수부의 값은 1~2046입니다. 실제 지수는 이 값에서 1023을 뺀 값이며 범위는 −1022~1023입니다.

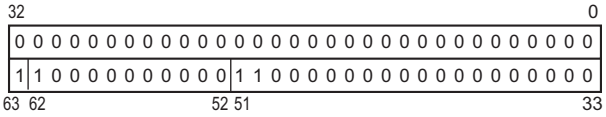
가수부의 값은 0~($2^{52}-1$)이 됩니다. 실제 가수는 2^{52}의 비트를 1로 가정하고 바로 다음에 소수점이 있다고 해석합니다.

정규화 수를 표현하는 값은

$(-1)^{(부호부)} \times 2^{(지수부)-1023} \times (1 + <가수부> \times 2^{-52})$

와 같습니다.

예



부호: −

지수: $1024 - 1023 = 1$

가수: $1 + (2^{51} + 2^{50}) \times 2^{-52} = 1 + (2^{-1} + 2^{-2}) = 1 + 0.75 = 1.75$

값: $-1.75 \times 2^1 = -3.5$

(2) 비정규화 수

절대값이 작은 실수값을 표현합니다.

부호부는 0(양) 또는 1(음)로 값의 부호를 나타냅니다.

지수부는 0이고 실제 지수는 −1022입니다.

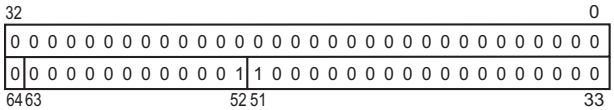
가수부는 1~$2^{52}-1$이며, 실제 가수는 2^{52}의 비트를 0으로 가정하고 바로 다음에 소수점이 있다고 해석합니다.

비정규화 수를 표현하는 값은

$(-1)^{(부호부)} \times 2^{-1022} \times (<가수부> \times 2^{-52})$

와 같습니다.

예



부호: +

지수: −1022

가수: $0 + (2^{51} + 2^{50}) \times 2^{-52} = 0 + (2^{-1} + 2^{-2}) = 0 + 0.75 = 0.75$

값: $0.75 \times 2^{-1022} = 1.668805 \times 10^{-308}$

(3) 0

부호부는 0(양) 또는 1(음)이며, 각각 +0.0, -0.0을 나타냅니다.

지수부와 가수부는 모두 0입니다.

+0.0, −0.0은 둘 다 값으로 0.0을 나타냅니다. 0의 부호에 의한 각 연산에서 기능의 차이는 '부동 소수점의 사칙연산'을 참조하십시오.

(4) 무한대

부호부는 0(양) 또는 1(음)이며, 각각 +∞, −∞를 나타냅니다.

지수부는 2047($2^{11} - 1$)입니다.

가수부는 0입니다.

(5) 비수(NaN)

'0.0/0.0', '∞/∞', '∞ − ∞' 등, 결과가 숫자 또는 무한대에 대응하지 않는 연산 결과를 얻게 됩니다.

지수부는 2047($2^{11} - 1$)입니다.

가수부는 0 이외의 값입니다.

주: 비수(NaN)의 부호 및 가수 필드(0 이외)의 값에 대해서는 규정되어 있지 않습니다.

■ 부동 소수점의 사칙연산

(1) 결과값의 반올림 방법

부동 소수점 수의 사칙연산 결과에 대한 정확한 값이 내부 표현의 가수의 유효 숫자를 초과한 경우는 다음 규칙에 따라 반올림합니다.

① 결과의 값은 그 값에서 가장 가까운 2개의 부동 소수점 수의 내부 표현 중 가까운 쪽을 향해 반올림됩니다.

② 결과의 값이 그 값에서 가장 가까운 2개의 부동 소수점 수의 정중앙에 있는 경우는 가수의 마지막 자리가 0이 되는 방향으로 반올림됩니다.

(2) 오버 플로, 언더 플로, 무효 연산일 때의 처리

실행 시의 오버 플로, 언더 플로, 무효 연산에 대해서는 다음과 같이 처리합니다.

① 오버 플로의 경우는 결과의 부호에 따라 양 또는 음의 무한대가 됩니다.

② 언더 플로의 경우는 결과의 부호에 따라 양 또는 음의 0이 됩니다.

③ 무효 연산은 부호가 반대인 무한대를 더한 경우, 부호가 같은 무한대를 뺀 경우, 0과 무한대를 곱한 경우, 0을 0으로 또는 무한대를 무한대로 나눈 경우에 발생합니다.
위와 같은 경우, 결과는 비수(NaN)가 됩니다.

④ 부동 소수점 수에서 정수로 변환한 경우 오버 플로가 발생하면 결과의 값은 보장되지 않습니다.

(3) 특수한 값의 연산에 대한 주의 사항

특수한 값(0, 무한대, 비수(NaN))의 연산에 대한 주의 사항은 다음과 같습니다.

① 양의 0과 음의 0의 합은 양의 0이 됩니다.

② 부호가 같은 0의 차는 양의 0이 됩니다.

③ 피연산자의 한쪽 또는 양쪽에 비수(NaN)가 포함된 연산의 결과는 항상 비수(NaN)가 됩니다.

④ 비교 연산에서는 양의 0과 음의 0이 같은 것으로 취급됩니다.

⑤ 피연산자의 한쪽 또는 양쪽이 비수(NaN)인 비교 연산, 등가 연산의 결과는 '<>'에 대해서 항상 참이 되고 그 밖에는 항상 거짓이 됩니다.

■ 배정도 부동 소수점 데이터의 연산 결과

(1) 오버 플로

연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 큰 경우 오버 플로가 됩니다. 이런 경우, 연산 결과는 다음과 같습니다.

- 연산 결과가 양일 때: +∞
- 연산 결과가 음일 때: −∞

(2) 연산 결과가 0이 되는 경우

연산 결과의 가수부 f와 지수부 e가 양쪽 모두 0인 경우 연산 결과는 0이 됩니다. 또한 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최소값보다 작은 경우 언더 플로가 됩니다. 이 경우에도 연산 결과는 0이 됩니다.

(3) 연산 결과가 0일 때 ON이 되는 조건 플래그인 = 플래그의 경우 연산 결과의 가수부 f와 지수부 e가 모두 0일 때만 ON이 됩니다.

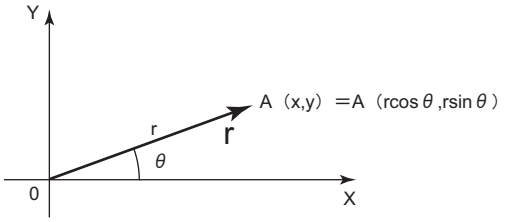
■ 단정도 부동 소수점 변환 및 연산 결과와 배정도 부동 소수점 변환 및 연산 결과의 비교

여기서는 극좌표 형식으로 표현된 벡터

$$\mathbf{r} = re^{j\left(\frac{\pi}{360}\right)\theta}$$

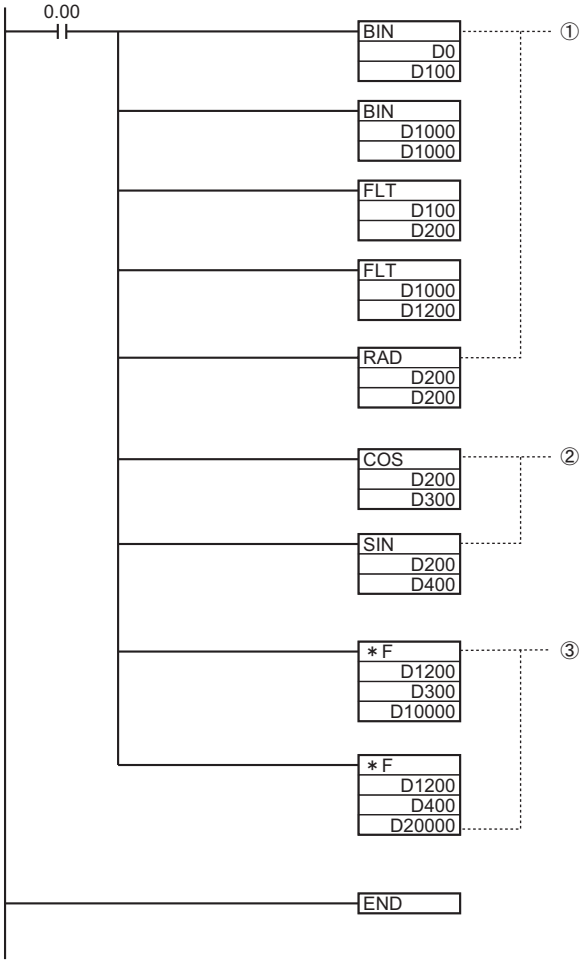
를 X-Y 직교 좌표계상의 좌표 A(x,y)로 변환하는 예로, 단정도 부동 소수점과 배정도 부동 소수점의 연산 결과의 차이를 나타냅니다.

또한 여기서는 θ = 20°(도), r = 10으로 하여 θ를 D0에, r를 D1000에 BCD 4자리로 저장한 것으로 합니다.



● 단정도 부동 소수점 변환 및 연산을 사용한 경우

· 프로그램 예



① BCD 데이터에서 단정도 부동 소수점 데이터(32비트: IEEE754)로 변환합니다.

- BIN 명령을 사용하여 일단 BIN 데이터로 변환하고, FLT 명령을 사용하여 단정도 부동 소수점 데이터(32비트: IEEE754)로 변환합니다.
- θ를 단정도 부동 소수점 데이터(32비트: IEEE754)로 변환한 값이 D200, D201에 출력됩니다.
- D200, D201에 저장된 각도를 라디안으로 변환합니다.
- r를 단정도 부동 소수점 데이터(32비트: IEEE754)로 변환한 값이 D1200, D1201에 출력됩니다.

② cosθ, sinθ의 값을 단정도 부동 소수점으로 산출합니다.

- cosθ의 값이 D300, D301에 출력됩니다.
- sinθ의 값이 D400, D401에 출력됩니다.

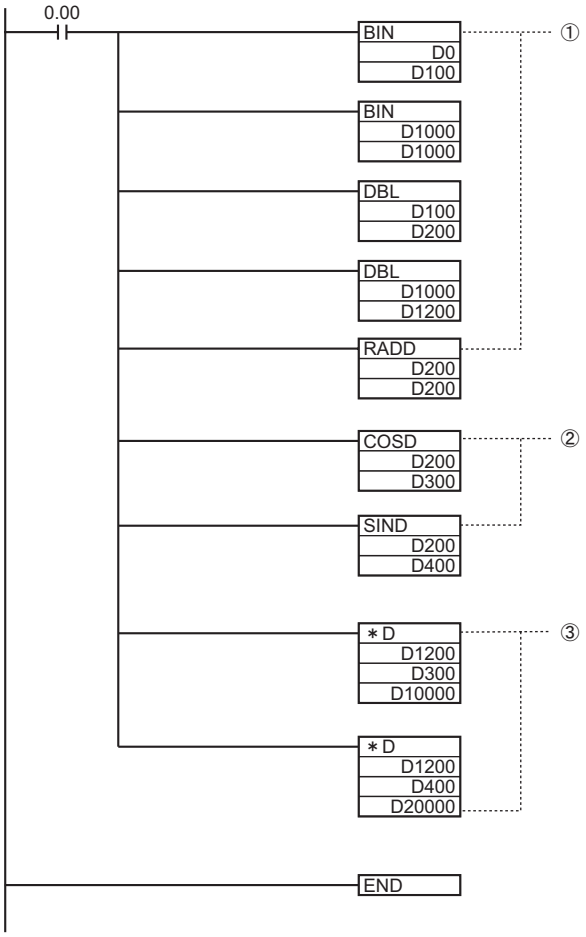
③ r×cosθ, r×sinθ를 단정도 부동 소수점으로 산출합니다.

- r×cosθ의 값이 D10000~D10001에 출력됩니다.
- r×sinθ의 값이 D20000~D20001에 출력됩니다.
- 연산 결과 A(x,y)

A의 좌표	단정도 부동 소수점 데이터 (32비트: IEEE754)	실수값
x	4116 59CF	3.4202015399933
y	405A E495	9.3969259262085

● 배정도 부동 소수점 변환 및 연산을 사용한 경우

· 프로그램 예



① BCD 데이터에서 배정도 부동 소수점 데이터(64비트: IEEE754)로 변환합니다.

- BIN 명령을 사용하여 일단 BIN 데이터로 변환하고, DBL 명령을 사용하여 배정도 부동 소수점 데이터(64비트: IEEE754)로 변환합니다.
- θ를 배정도 부동 소수점 데이터(64비트: IEEE754)로 변환한 값이 D200~D203에 출력됩니다.
- D200~D203에 저장된 각도를 라디안으로 변환합니다.
- r를 배정도 부동 소수점 데이터(64비트: IEEE754)로 변환한 값이 D1200~D1203에 출력됩니다.

② cosθ, sinθ의 값을 배정도 부동 소수점으로 산출합니다.

- cosθ의 값이 D300~D303에 출력됩니다.
- sinθ의 값이 D400~D403에 출력됩니다.

③ r×cosθ, r×sinθ를 배정도 부동 소수점으로 산출합니다.

- r×cosθ의 값이 D10000~D10003에 출력됩니다.
- r×sinθ의 값이 D20000~D20003에 출력됩니다.
- 연산 결과 A(x,y)

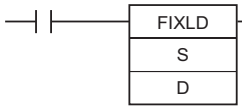
A의 좌표	배정도 부동 소수점 데이터 (64비트: IEEE754)	실수값
x	4022 CB39 E973 5C32	3.4202014332567
y	400B 5C92 91AC 8EEB	9.3969262078591

● 연산 결과의 비교

연산 결과의 실수값을 비교하면 배정도 부동 소수점 데이터를 사용한 쪽이 더 고정도의 결과를 얻을 수 있다는 것을 알 수 있습니다.

FIXD/FIXLD

명령 이름	니모닉	동작 옵션	평션 번호	기능
부동 소수점→16비트 BIN 변환<배>	FIXD	@FIXD	841	지정된 배정도 부동 소수점 데이터(64비트)를 부호 있는 BIN 16비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다.
부동 소수점→32비트 BIN 변환<배>	FIXLD	@FIXLD	842	지정된 배정도 부동 소수점 데이터(64비트)를 부호 있는 BIN 32비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다.

심볼	FIXD	FIXLD
	 S: 부동 소수점 데이터 하위 CH 번호 D: 변환 결과 출력 CH 번호	 S: 부동 소수점 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		FIXD	FIXLD	FIXD	FIXLD
S	부동 소수점 데이터 하위 CH 번호	LREAL	LREAL	4	4
D	변환 결과 출력 CH 번호	INT	DINT	1	2

■ 영역 종류

· FIXD

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	글록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D												○						

· FIXLD

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	글록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		

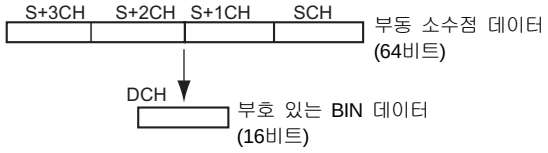
조건 플래그의 동작

이름	라벨	내용	
		FIXD	FIXLD
에러 플래그	P_ER	· S의 데이터가 부동 소수점 데이터로 처리되지 않을 때 ON · S의 데이터가 비수(NaN)일 때 ON · S의 데이터가 -32768~+32767의 범위 안에 없을 때 ON · 그 이외일 때 OFF	· S의 데이터가 부동 소수점 데이터로 처리되지 않을 때 ON · S의 데이터가 비수(NaN)일 때 ON · S의 데이터가 -2147483648~+2147483647의 범위 안에 없을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과가 0일 때 ON · 그 이외일 때 OFF	
부정 플래그	P_N	· 변환 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF	

기능

■ FIXD

S로 지정한 배정도 부동 소수점 데이터(64비트: IEEE754)의 정수부를 부호 있는 BIN(16비트)으로 변환하여 결과를 D에 출력합니다.



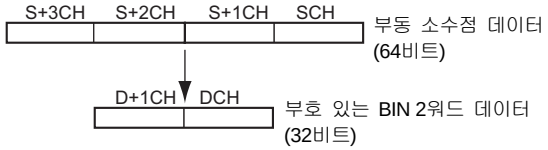
부동 소수점 데이터의 정수부를 BIN 데이터로 변환하여 결과를 지정된 채널에 출력합니다. 소수점 이하는 버립니다.

예:

부동 소수점 데이터		변환 결과 (부호 있는 BIN 16비트)
3.5	→	3
-3.5	→	-3

■ FIXLD

S로 지정한 배정도 부동 소수점 데이터(64비트: IEEE754)의 정수부를 부호 있는 BIN(32비트)으로 변환하여 결과를 D+1, D에 출력합니다.



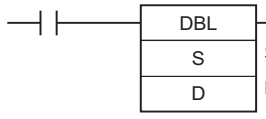
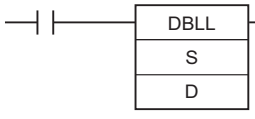
부동 소수점 데이터의 정수부를 부호 있는 BIN 2워드 데이터로 변환하여 결과를 지정된 채널에 출력합니다. 소수점 이하는 버립니다.

예:

부동 소수점 데이터		변환 결과 (부호 있는 BIN 32비트)
2147483640.5	→	2147483640
-2147483640.5	→	-2147483640

DBL/DBLL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
16비트 BIN→부동 소수점 변환<배>	DBL	@DBL	843	지정된 부호 있는 BIN 16비트 데이터를 배정도 부동 소수점 데이터(64비트)로 변환하여 결과를 지정된 채널에 출력합니다.
32비트 BIN→부동 소수점 변환<배>	DBLL	@DBLL	844	지정된 부호 있는 BIN 32비트 데이터를 배정도 부동 소수점 데이터(64비트)로 변환하여 결과를 지정된 채널에 출력합니다.

심볼	DBL	DBLL
	 S: BIN 데이터 D: 변환 결과 출력 하위 CH 번호	 S: BIN 2워드 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식		크기	
		DBL	DBLL	DBL	DBLL
S	DBL: BIN 데이터 DBLL: BIN 2워드 데이터 하위 CH 번호	INT	DINT	1	2
D	변환 결과 출력 하위 CH 번호	LREAL	LREAL	4	4

■ 영역 종류

· DBL

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
D											–	–						

· DBLL

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
D											–							

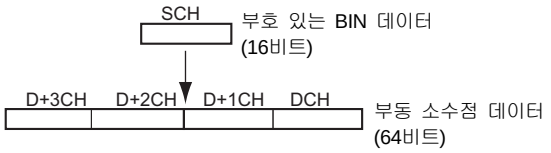
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 변환 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 변환 결과가 음수가 되었을 때 ON · 그 이외일 때 OFF

기능

■ DBL

S로 지정한 부호 있는 BIN 데이터(16비트)를 배정도 부동 소수점 데이터(64비트: IEEE754)로 변환하여 결과를 D+3, D+2, D+1, D에 출력합니다. 또한 결과의 소수점 이하 1자리는 0이 됩니다.



S에는 -32768~32767의 범위 내의 BIN 데이터를 지정할 수 있습니다.

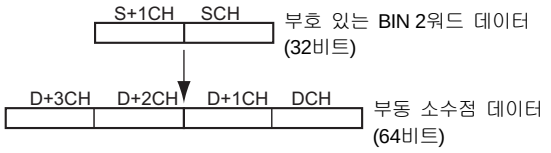
또한 -32768~32767의 범위 밖의 BIN 데이터를 변환하고 싶은 경우는 DBLL 명령을 사용하십시오.

예:

부호 있는 BIN 16비트		부동 소수점 데이터
3	→	3
-3	→	-3

■ DBLL

S로 지정한 BIN 데이터(32비트)를 배정도 부동 소수점 데이터(64비트: IEEE754)로 변환하여 결과를 D+3, D+2, D+1, D에 출력합니다. 또한 결과의 소수점 이하 1자리는 0이 됩니다.



S에는 -2147483648~2147483647의 범위 내의 BIN 2워드 데이터를 지정할 수 있습니다.

예:

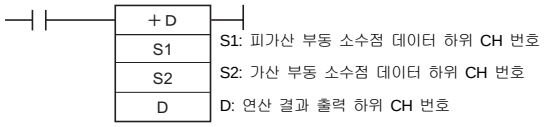
부호 있는 BIN 32비트		부동 소수점 데이터
16777215	→	16777215
-16777215	→	-16777215

+D, －D, *D, /D

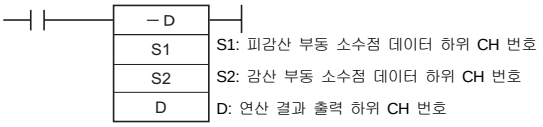
명령 이름	니모닉	동작 옵션	펑션 번호	기능
부동 소수점 가산<배>	＋D	@＋D	845	지정된 배정도 부동 소수점 데이터(64비트)를 더하여 결과를 지정된 채널에 출력합니다.
부동 소수점 감산<배>	－D	@－D	846	지정된 배정도 부동 소수점 데이터(64비트)를 빼서 결과를 지정된 채널에 출력합니다.
부동 소수점 승산<배>	＊D	@＊D	847	지정된 배정도 부동 소수점 데이터(64비트)를 곱해서 결과를 지정된 채널에 출력합니다.
부동 소수점 제산<배>	/D	@/D	848	지정된 배정도 부동 소수점 데이터(64비트)를 나눠서 결과를 지정된 채널에 출력합니다.

심볼

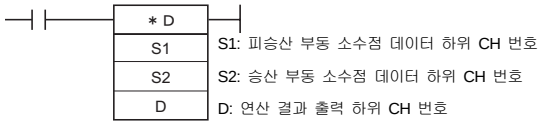
＋D



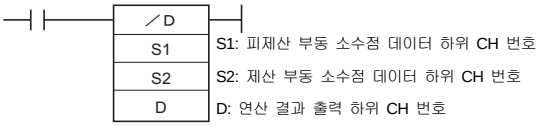
－D



＊D



/D



특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S1	＋D: 피가산 부동 소수점 데이터 하위 CH 번호 －D: 피감산 부동 소수점 데이터 하위 CH 번호 ＊D: 피승산 부동 소수점 데이터 하위 CH 번호 /D: 피제산 부동 소수점 데이터 하위 CH 번호	LREAL	4
S2	＋D: 가산 부동 소수점 데이터 하위 CH 번호 －D: 감산 부동 소수점 데이터 하위 CH 번호 ＊D: 승산 부동 소수점 데이터 하위 CH 번호 /D: 제산 부동 소수점 데이터 하위 CH 번호	LREAL	4
D	연산 결과 출력 하위 CH 번호	LREAL	4

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	＊DM ＊EM		DR	IR 직접	IR 간접				
S1, S2	○	○	○	○	○	○	○	○	○	○	－	－	－	○	－	－	－	－
D																		

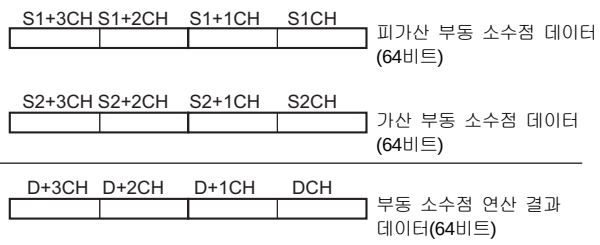
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S1 또는 S2가 부동 소수점 데이터로 처리되지 않을 때 ON · +D ▪ 피가산 데이터 또는 가산 데이터가 비수(NaN)일 때 ON ▪ +∞와 −∞를 더하려고 한 경우 ON · −D ▪ 피감산 데이터 또는 감산 데이터가 비수(NaN)일 때 ON ▪ +∞와 +∞ 또는 −∞와 −∞를 빼려고 한 경우 ON · *D ▪ 피승산 데이터 또는 승산 데이터가 비수(NaN)일 때 ON ▪ 0과 +∞ 또는 −∞를 곱하려고 한 경우 ON · /D ▪ 피제산 데이터 또는 제산 데이터가 비수(NaN)일 때 ON ▪ 피제산 데이터와 제산 데이터가 모두 0 또는 ±∞인 경우 ON ·그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 클 때 ON · 그 이외일 때 OFF
언더 플로 플래그	P_UF	· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최소값보다 작을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 연산 결과가 음수가 되었을 때 ON · 그 이외일 때 OFF

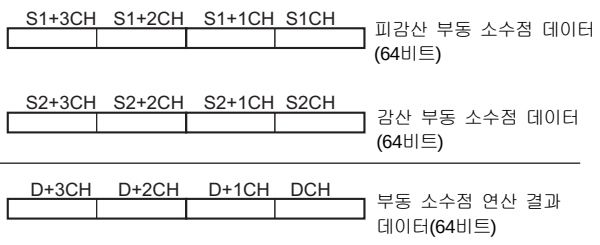
기능

S1로 지정한 데이터와 S2로 지정한 데이터를 배정도 부동 소수점 데이터(64비트: IEEE754)로 가산(+D), 감산(−D), 승산(*D), 제산(/D)하여 결과를 D+3, D+2, D+1, D에 출력합니다.

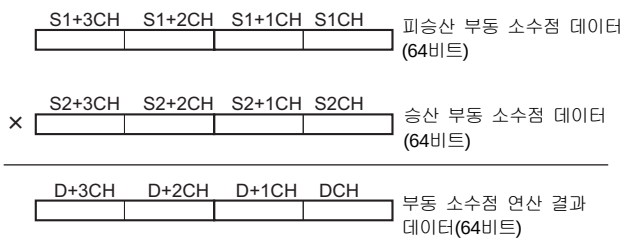
· +D



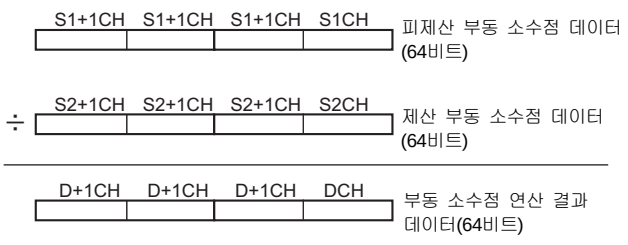
· −D



· *D



· /D



+D, −D, *D, /D

■ 연산 규칙

연산하는 부동 소수점 데이터의 조합에 맞춰 다음 표에 따른 결과를 출력합니다.

● 부동 소수점 가산<배> (+D)

가산 데이터 \ 피가산 데이터	0	숫자	+∞	−∞	비수(NaN)
0	0	숫자	+∞	−∞	
숫자	숫자	* *	+∞	−∞	
+∞	+∞	+∞	+∞	ER	
−∞	−∞	−∞	ER	−∞	
비수(NaN)					ER

* *: 이 경우는 0(언더 플로의 결과가 0이 되는 경우 포함) 또는 숫자, +∞, −∞ 중 하나의 결과를 출력합니다.
ER: 에러 플래그가 ON이 되고 명령이 실행되지 않습니다.

● 부동 소수점 감산<배> (−D)

감산 데이터 \ 피감산 데이터	0	숫자	+∞	−∞	비수(NaN)
0	0	숫자	+∞	−∞	
숫자	숫자	* *	+∞	−∞	
+∞	−∞	−∞	ER	−∞	
−∞	+∞	+∞	+∞	ER	
비수(NaN)					ER

* *: 이 경우는 0(언더 플로의 결과가 0이 되는 경우 포함) 또는 숫자, +∞, −∞ 중 하나의 결과를 출력합니다.
ER: 에러 플래그가 ON이 되고 명령이 실행되지 않습니다.

● 부동 소수점 승산<배> (*D)

승산 데이터 \ 피승산 데이터	0	숫자	+∞	−∞	비수(NaN)
0	0	0	ER	ER	
숫자	0	* *	+/−∞	+/−∞	
+∞	ER	+/−∞	+∞	−∞	
−∞	ER	+/−∞	−∞	+∞	
비수(NaN)					ER

* *: 이 경우는 0(언더 플로의 결과가 0이 되는 경우 포함) 또는 숫자, +∞, −∞ 중 하나의 결과를 출력합니다.
ER: 에러 플래그가 ON이 되고 명령이 실행되지 않습니다.

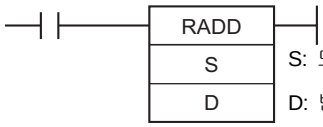
● 부동 소수점 제산<배> (/D)

제산 데이터 \ 피제산 데이터	0	숫자	+∞	−∞	비수(NaN)
0	ER	+/−∞	+∞	−∞	
숫자	0	* *	+/−∞	+/−∞	
+∞	0	0*	ER	ER	
−∞	0	0*	ER	ER	
비수(NaN)					ER

*: 언더 플로의 결과 0이 됩니다.
* *: 이 경우는 0(언더 플로의 결과가 0이 되는 경우 포함) 또는 숫자, +∞, −∞ 중 하나의 결과를 출력합니다.
ER: 에러 플래그가 ON이 되고 명령이 실행되지 않습니다.

RADD

명령 이름	니모닉	동작 옵션	펑션 번호	기능
각도 →라디안 변환 <배>	RADD	@RADD	849	지정된 배정도 부동 소수점 데이터(64비트)를 도(°) 단위에서 라디안(rad) 단위로 변환하여 결과를 지정된 채널에 출력합니다.

심볼	RADD
	

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	도(°) 데이터 하위 CH 번호	LREAL	4
D	변환 결과 출력 하위 CH 번호	LREAL	4

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 각도 데이터가 부동 소수점 데이터로 처리되지 않을 때 ON · 각도 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 변환 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 클 때 ON · 그 이외일 때 OFF
언더 플로 플래그	P_UF	· 변환 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최소값보다 작을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 변환 결과가 음수가 되었을 때 ON · 그 이외일 때 OFF

기능

S로 지정한 배정도 부동 소수점 데이터(64비트: IEEE754)로 표현되는 각도 데이터를 도 단위에서 라디안 단위로 변환하여 결과를 D+3, D+2, D+1, D에 출력합니다.

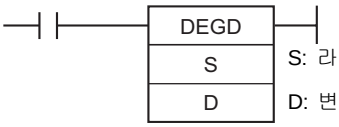


RADD 명령은 다음 식에 기초하여 도에서 라디안으로 변환을 수행합니다.

도(°)×π/180 = 라디안(rad)

DEGD

명령 이름	니모닉	동작 옵션	펑션 번호	기능
라디안→각도 변환 <배>	DEGD	@DEGD	850	지정된 배정도 부동 소수점 데이터(64비트)를 라디안(rad) 단위에서 도(°) 단위로 변환하여 결과를 지정된 채널에 출력합니다.

심볼	DEGD
	 S: 라디안(rad) 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	라디안(rad) 데이터 하위 CH 번호	LREAL	4
D	변환 결과 출력 하위 CH 번호	LREAL	4

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		

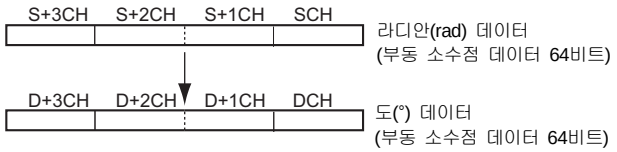
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 각도 데이터가 부동 소수점 데이터로 처리되지 않을 때 ON · 각도 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 변환 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 클 때 ON · 그 이외일 때 OFF
언더 플로 플래그	P_UF	· 변환 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최소값보다 작을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 변환 결과가 음수가 되었을 때 ON · 그 이외일 때 OFF

DEGD

기능

S로 지정한 배정도 부동 소수점 데이터(64비트: IEEE754)로 표현되는 각도 데이터를 라디안 단위에서 도 단위로 변환하여 결과를 D+3, D+2, D+1, D에 출력합니다.



DEGD 명령은 다음 식에 기초하여 라디안에서 도로 변환을 수행합니다.

라디안(rad)×180/π = 도(°)

SIND/COSD/TAND

명령 이름	니모닉	동작 옵션	평션 번호	기능
SIN 연산<배>	SIND	@SIND	851	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 각도(라디안 단위)의 SIN(사인) 값을 연산하여 결과를 지정된 채널에 출력합니다.
COS 연산<배>	COSD	@COSD	852	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 각도(라디안 단위)의 COS(코사인) 값을 연산하여 결과를 지정된 채널에 출력합니다.
TAN 연산<배>	TAND	@TAND	853	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 각도(라디안 단위)의 TAN(탄젠트) 값을 연산하여 결과를 지정된 채널에 출력합니다.

심볼	SIND	COSD	TAND
	SIND S: 각도(rad) 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	COSD S: 각도(rad) 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	TAND S: 각도(rad) 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	각도(rad) 데이터 하위 CH 번호	LREAL	4
D	변환 결과 출력 하위 CH 번호	LREAL	4

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 각도 데이터가 부동 소수점 데이터로 처리되지 않을 때 ON · 각도 데이터가 비수(NaN)일 때 ON · 각도 데이터의 절대값이 0~65535의 범위 안에 없을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· SIND, COSD 유지 · TAND ▪ 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 클 때 ON ▪ 그 이외일 때 OFF
언더 플로 플래그	P_UF	유지
부정 플래그	P_N	· 연산 결과가 음수일 때 ON · 그 이외일 때 OFF

SIND/COSD/TAND

기능

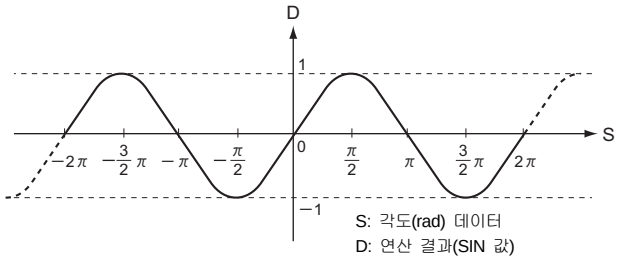
■ SIND

S로 지정한 배정도 부동 소수점 데이터(64비트: IEEE754)로 표현되는 각도(라디안 단위)에 대한 SIN(사인) 값을 연산하여 결과를 D+3, D+2, D+1, D에 출력합니다.



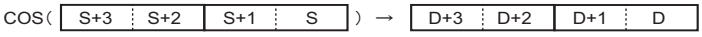
· S에는 라디안(rad) 단위의 각도 데이터를 지정하십시오.
  또한 도(°)와 라디안(rad)의 변환에 대해서는 RADD 명령, DEGD 명령을 참조하십시오.
· 각도 데이터의 절대값이 65536 이상이면 에러 플래그가 ON이 되고 명령은 실행되지 않습니다.

● 입력 데이터와 연산 결과의 관계



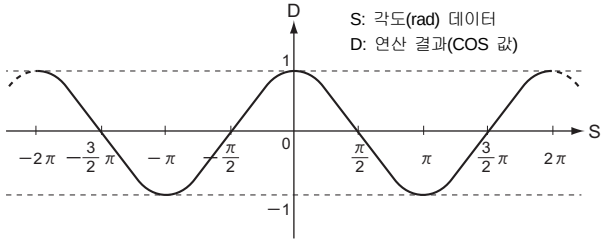
■ COSD

S로 지정한 배정도 부동 소수점 데이터(64비트: IEEE754)로 표현되는 각도(라디안 단위)에 대한 COS(코사인) 값을 연산하여 결과를 D+3, D+2, D+1, D에 출력합니다.



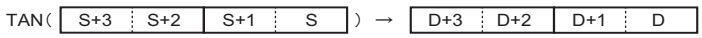
· S에는 라디안(rad) 단위의 각도 데이터를 지정하십시오.
  또한 도(°)와 라디안(rad)의 변환에 대해서는 RADD 명령, DEGD 명령을 참조하십시오.
· 각도 데이터의 절대값이 65536 이상이면 에러 플래그가 ON이 되고 명령은 실행되지 않습니다.

● 입력 데이터와 연산 결과의 관계



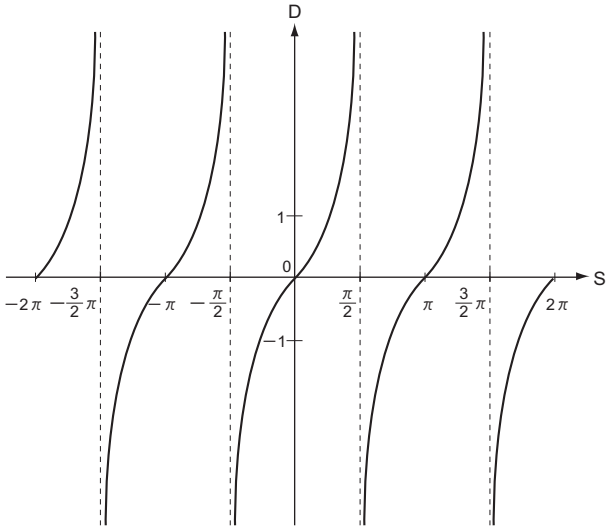
■ TAND

S로 지정한 배정도 부동 소수점 데이터(64비트: IEEE754)로 표현되는 각도(라디안 단위)에 대한 TAN(탄젠트) 값을 연산하여 결과를 D+3, D+2, D+1, D에 출력합니다.



· S에는 라디안(rad) 단위의 각도 데이터를 지정하십시오.
  또한 도(°)와 라디안(rad)의 변환에 대해서는 RAD 명령, DEG 명령을 참조하십시오.
· 각도 데이터의 절대값이 65536 이상이면 에러 플래그가 ON이 되고 명령은 실행되지 않습니다.
· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 큰 경우 오버 플로 플래그가 ON이 됩니다. 이런 경우, 연산 결과는 ±∞로 출력됩니다.

● 입력 데이터와 연산 결과의 관계



S: 각도(rad) 데이터
D: 연산 결과(TAN 값)

ASIND/ACOSD/ATAND

명령 이름	니모닉	동작 옵션	평션 번호	기능
SIN^{-1} 연산<배>	ASIND	@ASIND	854	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 SIN(사인) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다.
COS^{-1} 연산<배>	ACOSD	@ACOSD	855	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 COS(코사인) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다.
TAN^{-1} 연산<배>	ATAND	@ATAND	856	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 TAN(탄젠트) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다.

심볼	ASIND	ACOSD	ATAND
	ASIND S: S: SIN(사인) 데이터 하위 CH 번호 D: D: 변환 결과 출력 하위 CH 번호	ACOSD S: S: COS(코사인) 데이터 하위 CH 번호 D: D: 변환 결과 출력 하위 CH 번호	ATAND S: S: TAN(탄젠트) 데이터 하위 CH 번호 D: D: 변환 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	ASIND: SIN(사인) 데이터 하위 CH 번호 ACOSD: COS(코사인) 데이터 하위 CH 번호 ATAND: TAN(탄젠트) 데이터 하위 CH 번호	LREAL	4
D	변환 결과 출력 하위 CH 번호	LREAL	4

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		

조건 플래그의 동작

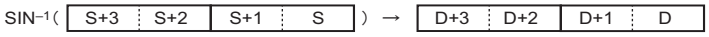
이름	라벨	내용
에러 플래그	P_ER	· S가 부동 소수점 데이터로 처리되지 않을 때 ON · ASIND ・ SIN 데이터가 비수(NaN)일 때 ON ・ SIN 데이터가 −1.0~1.0의 범위 안에 없을 때 ON · ACOSD ・ COS 데이터가 비수(NaN)일 때 ON ・ COS 데이터가 −1.0~1.0의 범위 안에 없을 때 ON · ATAND ・ TAN 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF

이름	라벨	내용
오버 플로 플래그	P_OF	유지
언더 플로 플래그	P_UF	유지
부정 플래그	P_N	· ASIND, ATAND ▪ 연산 결과가 음수가 되었을 때 ON ▪ 그 이외일 때 OFF · ACOSD 유지

기능

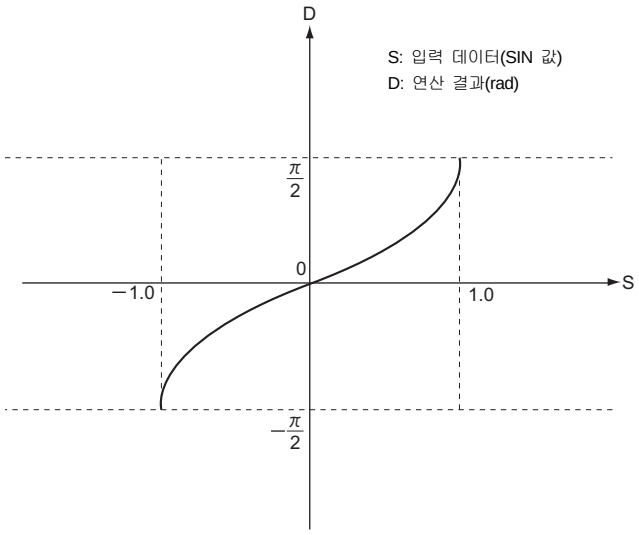
■ ASIND

S로 지정한 배정도 부동 소수점 데이터(64비트: IEEE754)로 표현되는 SIN(사인) 값에 대한 각도(라디안 단위)를 연산하여 결과를 D+3, D+2, D+1, D에 출력합니다.



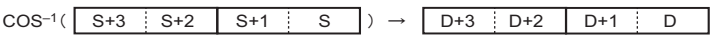
연산 결과는 −π/2~π/2의 범위 내의 각도(라디안 단위) 데이터로 D+3, D+2, D+1, D CH에 출력됩니다.

● 입력 데이터와 연산 결과의 관계



■ ACOSD

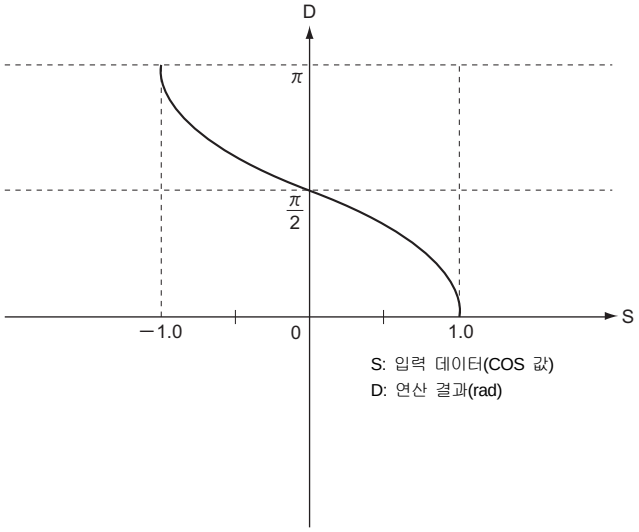
S로 지정한 배정도 부동 소수점 데이터(64비트: IEEE754)로 표현되는 COS(코사인) 값에 대한 각도(라디안 단위)를 연산하여 결과를 D+3, D+2, D+1, D에 출력합니다.



연산 결과는 0~π의 범위 내의 각도(라디안 단위) 데이터로 D+3, D+2, D+1, D CH에 출력됩니다.

ASIND/ACOSD/ATAND

● 입력 데이터와 연산 결과의 관계



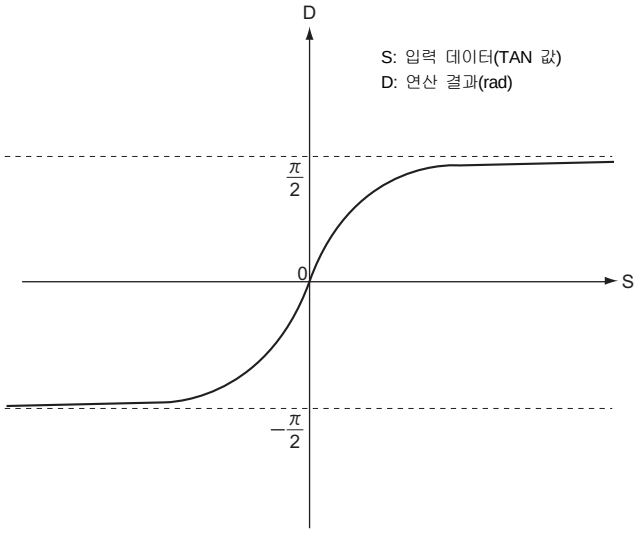
■ ATAND

S로 지정한 배정도 부동 소수점 데이터(64비트: IEEE754)로 표현되는 TAN(탄젠트) 값에 대한 각도(라디안 단위)를 연산하여 결과를 D+3, D+2, D+1, D에 출력합니다.

TAN⁻¹(S+3 | S+2 | S+1 | S) → D+3 | D+2 | D+1 | D

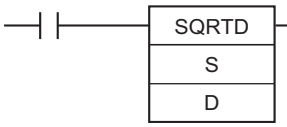
연산 결과는 −π/2~π/2의 범위 내의 각도(라디안 단위) 데이터로 D+3, D+2, D+1, D CH에 출력됩니다.

● 입력 데이터와 연산 결과의 관계



SQRTD

명령 이름	니모닉	동작 옵션	펑션 번호	기능
제곱근 연산<배>	SQRTD	@SQRTD	857	지정된 배정도 부동 소수점 데이터(64비트)의 제곱근(√)을 연산하여 결과를 지정된 채널에 출력합니다.

심볼	SQRTD
	 S: 입력 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	입력 데이터 하위 CH 번호	LREAL	4
D	연산 결과 출력 하위 CH 번호	LREAL	4

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		

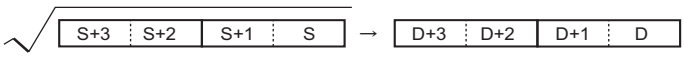
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 입력 데이터가 부동 소수점 데이터로 처리되지 않을 때 ON · 입력 데이터가 음수일 때 ON · 입력 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 클 때 ON · 그 이외일 때 OFF
언더 플로 플래그	P_UF	유지
부정 플래그	P_N	유지

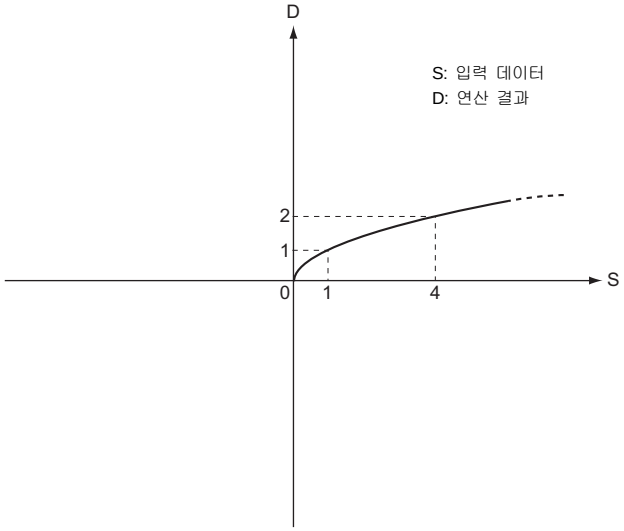
SQRTD

기능

S로 지정한 배정도 부동 소수점 데이터(64비트: IEEE754)로 표현되는 입력 데이터에 대한 제곱근을 연산하여 결과를 D+3, D+2, D+1, D에 출력합니다.

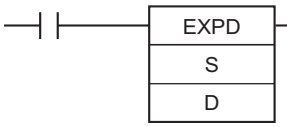


■ 입력 데이터와 연산 결과의 관계



EXPD

명령 이름	니모닉	동작 옵션	펑션 번호	기능
지수 연산<배>	EXPD	@EXPD	858	지정된 배정도 부동 소수점 데이터(64비트)의 지수(e를 밑수로 했을 때의 지수)를 연산하여 결과를 지정된 채널에 출력합니다.

심볼	EXPD
	 S: 입력 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	입력 데이터 하위 CH 번호	LREAL	4
D	연산 결과 출력 하위 CH 번호	LREAL	4

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		

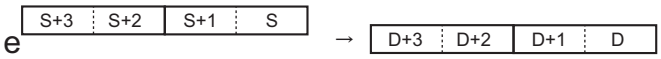
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 입력 데이터가 부동 소수점 데이터로 처리되지 않을 때 ON · 입력 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 클 때 ON · 그 이외일 때 OFF
언더 플로 플래그	P_UF	· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최소값보다 작을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	유지

EXPD

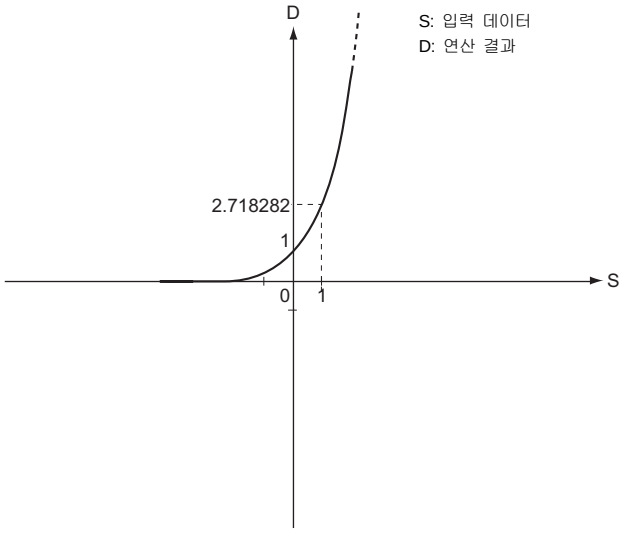
기능

S로 지정한 배정도 부동 소수점 데이터(64비트: IEEE754)로 표현되는 입력 데이터에 대한 지수(e를 밑수로 했을 때의 지수)를 연산하여 결과를 D+3, D+2, D+1, D에 출력합니다.



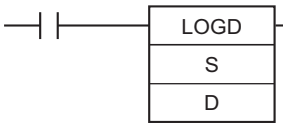
· EXPD 명령은 밑수(e)를 2.718282로 하여 지수 연산합니다.

■ 입력 데이터와 연산 결과의 관계



LOGD

명령 이름	니모닉	동작 옵션	평션 번호	기능
로그 연산<배>	LOGD	@LOGD	859	지정된 배정도 부동 소수점 데이터(64비트)의 자연 로그(e를 밑수로 했을 때의 지수)를 연산하여 결과를 지정된 채널에 출력합니다.

심볼	LOGD
	 S: 입력 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	입력 데이터 하위 CH 번호	LREAL	4
D	연산 결과 출력 하위 CH 번호	LREAL	4

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 입력 데이터가 부동 소수점 데이터로 처리되지 않을 때 ON · 입력 데이터가 음수일 때 ON · 입력 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 클 때 ON · 그 이외일 때 OFF
언더 플로 플래그	P_UF	유지
부정 플래그	P_N	· 연산 결과가 음수가 되었을 때 ON · 그 이외일 때 OFF

LOGD

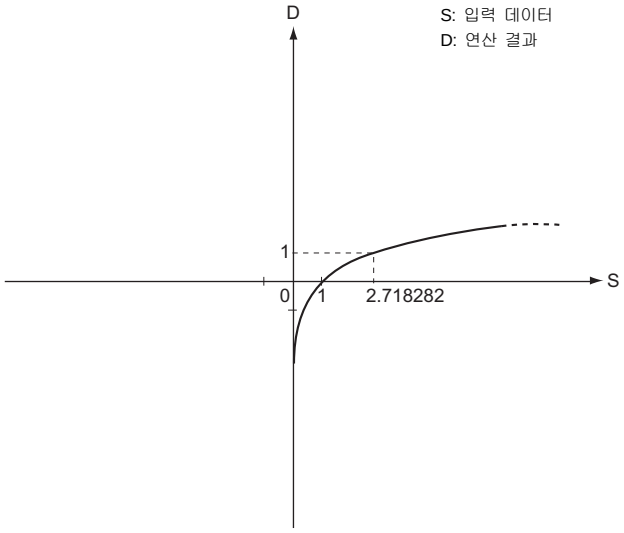
기능

S로 지정한 배정도 부동 소수점 데이터(64비트: IEEE754)로 표현되는 입력 데이터에 대한 자연 로그(e를 밑수로 했을 때의 로그)를 연산하여 결과를 D+3, D+2, D+1, D에 출력합니다.

$\log_e$ [S+3 | S+2 | S+1 | S] → [D+3 | D+2 | D+1 | D]

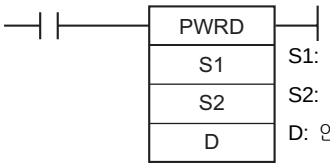
· LOGD 명령은 밑수(e)를 2.718282로 하여 지수 연산합니다.

■ 입력 데이터와 연산 결과의 관계



PWRD

명령 이름	니모닉	동작 옵션	펑션 번호	기능
거듭제곱 연산<배>	PWRD	@PWRD	860	지정된 배정도 부동 소수점 데이터(64비트)의 거듭제곱 연산을 수행하여 결과를 지정된 채널에 출력합니다.

심볼	PWRD
	 S1: 피거듭제곱 데이터 하위 CH 번호 S2: 거듭제곱 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S1	피거듭제곱 데이터 하위 CH 번호	LREAL	4
S2	거듭제곱 데이터 하위 CH 번호	LREAL	4
D	연산 결과 출력 하위 CH 번호	LREAL	4

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S1, S2	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 피거듭제곱 데이터 또는 거듭제곱 데이터가 부동 소수점 데이터로 처리되지 않을 때 ON · 피거듭제곱 데이터 또는 거듭제곱 데이터가 비수(NaN)일 때 ON · 피거듭제곱 데이터가 0이고 거듭제곱 데이터가 0 이하일 때 ON · 피거듭제곱 데이터가 음수이고 거듭제곱 데이터가 정수가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과(부동 소수점 데이터)의 지수부와 가수부가 모두 0(부동 소수점 데이터의 0)일 때 ON · 그 이외일 때 OFF
오버 플로 플래그	P_OF	· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최대값보다 클 때 ON · 그 이외일 때 OFF
언더 플로 플래그	P_UF	· 연산 결과의 절대값이 부동 소수점 데이터로 표현할 수 있는 최소값보다 작을 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 연산 결과가 음수가 되었을 때 ON · 그 이외일 때 OFF

PWRD

기능

S1로 지정한 데이터와 S2로 지정한 데이터를 배정도 부동 소수점 데이터(64비트: IEEE754)로 거듭제곱 산하여 결과를 D+3, D+2, D+1, D에 출력합니다.



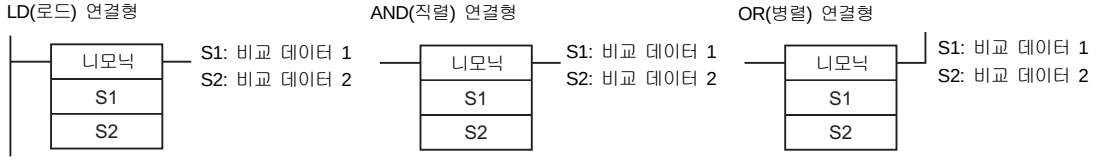
예:

부동 소수점 데이터		연산 결과
S1: 3.1 S2: 3.0	→ 3.1^3 →	D: 29.791

=D, <>D, <D, <=D, >D, >=D

명령 이름	니모닉	동작 옵션	펑션 번호	기능
배정도 부동 소수점 데이터 비교	=D <>D <D <=D >D >=D	–	335 336 337 338 339 340	지정된 배정도 부동 소수점 데이터(64비트)를 비교하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다.

심볼	배정도 부동 소수점 데이터 비교



특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S1	비교 데이터 1	LREAL	4
S2	비교 데이터 2	LREAL	4

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S1	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S2																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
> 플래그	P_GT	· 비교 결과가 S1+3, S1+2, S1+1, S1 > S2+3, S2+2, S2+1, S2일 때 ON · 그 이외일 때 OFF
≧ 플래그	P_GE	· 비교 결과가 S1+3, S1+2, S1+1, S1≧S2+3, S2+2, S2+1, S2일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 비교 결과가 S1+3, S1+2, S1+1, S1 = S2+3, S2+2, S2+1, S2일 때 ON · 그 이외일 때 OFF
≠ 플래그	P_NE	· 비교 결과가 S1+3, S1+2, S1+1, S1≠S2+3, S2+2, S2+1, S2일 때 ON · 그 이외일 때 OFF
< 플래그	P_LT	· 비교 결과가 S1+3, S1+2, S1+1, S1 < S2+3, S2+2, S2+1, S2일 때 ON · 그 이외일 때 OFF
≦ 플래그	P_LE	· 비교 결과가 S1+3, S1+2, S1+1, S1≦S2+3, S2+2, S2+1, S2일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	유지

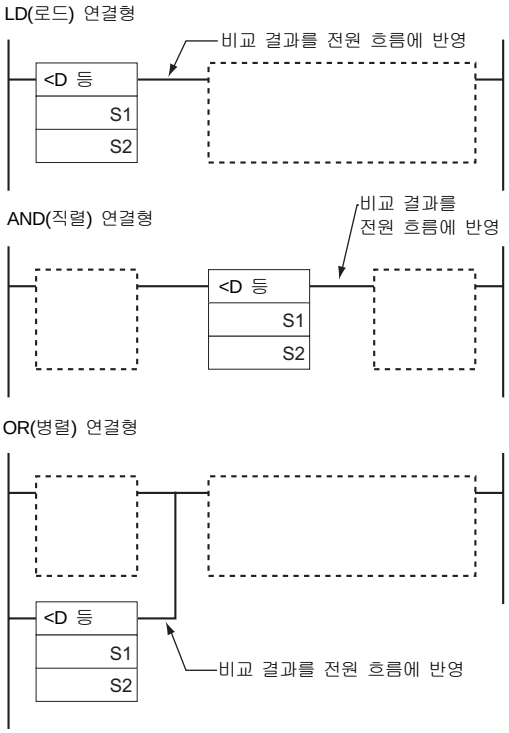
=D, <>D, <D, <=D, >D, >=D

기능

S1로 지정한 데이터와 S2로 지정한 데이터를 배정도 부동 소수점 데이터(64비트: IEEE754)로 비교하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다. 비교 데이터 S1, S2를 지정하려면 데이터가 저장된 채널의 시작 주소를 피연산자 S1, S2에 입력합니다. 또한 상수는 비교 데이터로 사용할 수 없습니다.

LD, AND, OR 명령과 동일하게 취급하며, 각 명령 후에는 계속하여 다른 명령을 프로그래밍합니다.

- LD형의 경우: 모선에 직접 연결할 수 있습니다.
- AND형의 경우: 모선에 직접 연결할 수 없습니다.
- OR형의 경우: 모선에 직접 연결할 수 있습니다.



■ 심볼과 옵션

비교 명령은 심볼과 옵션을 조합하여 18종류의 니모닉으로 표현됩니다.

심볼 (래더에서는 LD, AND, OR 없음)		옵션 (데이터 형식)
LD=, AND=, OR=, LD<>, AND<>, OR<>, LD<, AND<, OR<, LD<=, AND<=, OR<=, LD>, AND>, OR>, LD>=, AND>=, OR>=	+	D(배정도 부동 소수점)

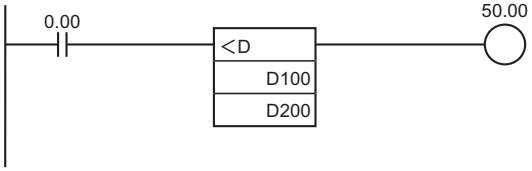
기능	니모닉	이름	펑션 번호
S1 = S2일 때 참(ON)	LD = D	LD 부동 소수점 데이터 일치(배)	335
	AND = D	AND 부동 소수점 데이터 일치(배)	335
	OR = D	OR 부동 소수점 데이터 일치(배)	335
S1≠ S2일 때 참(ON)	LD <> D	LD 부동 소수점 데이터 불일치(배)	336
	AND <> D	AND 부동 소수점 데이터 불일치(배)	336
	OR <> D	OR 부동 소수점 데이터 불일치(배)	336
S1 < S2일 때 참(ON)	LD < D	LD 부동 소수점 데이터 미만(배)	337
	AND < D	AND 부동 소수점 데이터 미만(배)	337
	OR < D	OR 부동 소수점 데이터 미만(배)	337
S1≦S2일 때 참(ON)	LD <= D	LD 부동 소수점 데이터 이하(배)	338
	AND <= D	AND 부동 소수점 데이터 이하(배)	338
	OR <= D	OR 부동 소수점 데이터 이하(배)	338
S1 > S2일 때 참(ON)	LD > D	LD 부동 소수점 데이터 초과(배)	339
	AND > D	AND 부동 소수점 데이터 초과(배)	339
	OR > D	OR 부동 소수점 데이터 초과(배)	339
S1≧S2일 때 참(ON)	LD >= D	LD 부동 소수점 데이터 이상(배)	340
	AND>= D	AND 부동 소수점 데이터 이상(배)	340
	OR >= D	OR 부동 소수점 데이터 이상(배)	340

사용상의 주의 사항

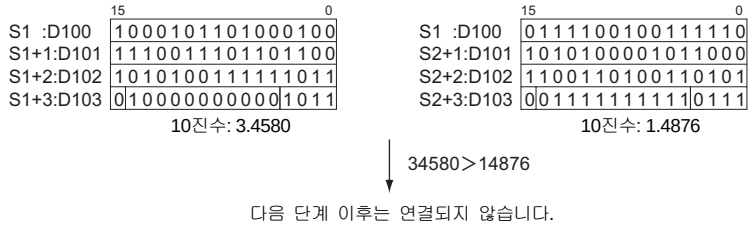
- 이 명령의 최종 단계에는 출력계 명령(OUT계 명령 및 다음 단계 연결형 명령 이외의 응용 명령)을 추가하십시오.
- 회로의 최종 단계에는 이 명령을 사용할 수 없습니다.

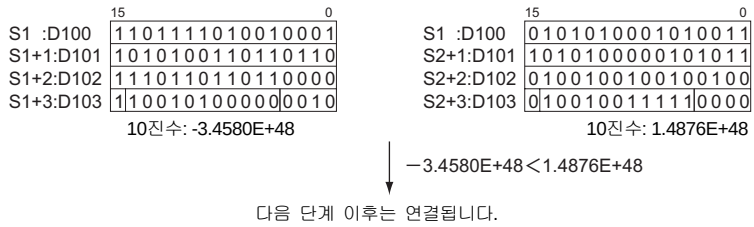
샘플 프로그램

0.00이 ON일 때 데이터 메모리 D100~103에 저장된 부동 소수점 데이터와 D200~203에 저장된 부동 소수점 데이터를 비교합니다. 비교 결과가 (D100~103의 데이터) < (D200~203의 데이터)인 경우, 다음 단계 이후가 연결되어 출력 릴레이 50.00이 ON됩니다.
비교 결과가 (D100~103의 데이터) < (D200~203의 데이터)가 아닌 경우는 다음 단계 이후로 연결되지 않습니다.



부호 없는 비교(<)





테이블 데이터 처리 명령

니모닉	명령 이름	펑션 번호	페이지
SSET	스택 영역 설정	630	3-430
PUSH	스택 데이터 저장	632	3-432
LIFO	후입선출	634	3-434
FIFO	선입선출	633	
SNUM	스택 데이터 수 출력	638	3-438
SREAD	스택 데이터 참조	639	3-440
SWRIT	스택 데이터 갱신	640	3-443
SINS	스택 데이터 삽입	641	3-446
SDEL	스택 데이터 삭제	642	3-449
DIM	테이블 영역 선언	631	3-452
SETR	레코드 위치 설정	635	3-454
GETR	레코드 위치 읽기	636	3-456
SRCH	데이터 검색	181	3-458
SWAP	바이트 스와핑	637	3-461
MAX	최대값 검색	182	3-463
MIN	최소값 검색	183	
MAXL	2워드 데이터 최대값 검색	174	3-467
MAXF	단정도 부동 소수점 데이터 최대값 검색	176	3-470
MAXD	배정도 부동 소수점 데이터 최대값 검색	178	3-472
MINL	2워드 데이터 최소값 검색	175	3-474
MINF	단정도 부동 소수점 데이터 최소값 검색	177	3-476
MIND	단정도 부동 소수점 데이터 최소값 검색	179	3-478
SUM	SUM 값 산출	184	3-480
FCS	FCS 값 산출	180	3-483

테이블 데이터 처리 명령

테이블 데이터 처리 명령에서 사용할 수 있는 영역은 다음 3가지 영역입니다.

1) 스택 영역

2) 테이블 영역

3) 각 명령으로 지정한 영역

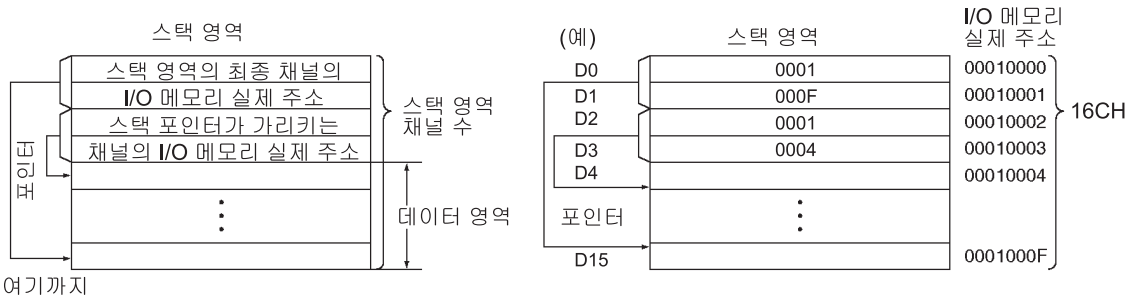
1), 2)는 모두 일반적인 I/O 메모리 영역을 특정 영역 지정 명령(각각 SSET 명령, DIM 명령)으로 확보합니다.

3)은 각 명령별로 피연산자를 사용하여 영역을 지정합니다.

영역 종류	용도	관련 명령
스택 영역	선입선출, 후입선출 등의 처리	SSET, PUSH, FIFO, LIFO, SREAD, SWRIT, SINS, SDEL, SNUM 명령
테이블 영역	레코드 단위의 데이터 처리	DIM, SETR, GETR 명령
각 명령 지정 영역	최대값, 최소값 검색이나 SUM 계산 등의 처리	MIN, MAX, SRCH, SUM, FCS, SWAP 명령

■ 스택 영역

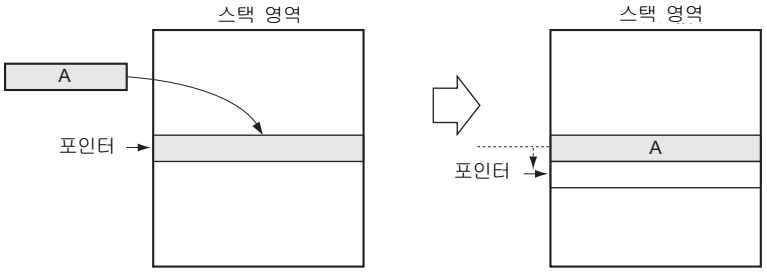
스택 영역은 다음 그림과 같은 구조를 가진 데이터 영역입니다. SSET 명령을 사용하여 I/O 메모리의 일부를 스택 영역으로 정의할 수 있습니다. 스택 영역 시작부에는 포인터용 주소가 저장되고, 데이터를 저장할 때마다 포인터 주소에 1을 더하여 다음에 저장해야 하는 주소를 가리킵니다.



PUSH 명령으로 스택 영역에 데이터를 차례로 저장합니다. 그 후, 저장해둔 데이터는 FIFO 명령 또는 LIFO 명령을 사용하여 먼저 입력한 것부터 가져오거나 나중에 입력한 것부터 가져옵니다.

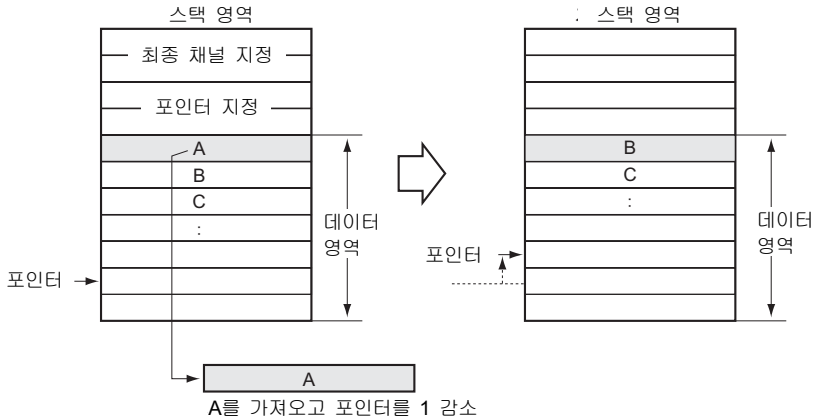
● 스택 데이터 저장(PUSH 명령)

포인터의 위치에 데이터를 저장하고 포인터를 1 증가시킵니다.



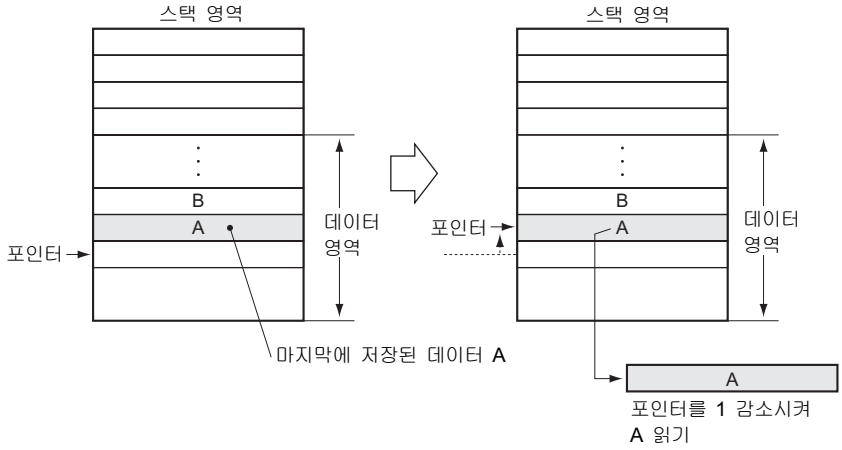
● 선입선출(FIFO 명령)

먼저 저장한 데이터부터 꺼냅니다. 꺼낸 데이터는 소멸하고 포인터에서 1을 뺍니다.



● 후입선출(LIFO 명령)

나중에 저장한 데이터부터 읽습니다. 포인터에서 1을 뺀 다음 그 주소의 데이터(마지막에 저장된 데이터)를 읽습니다. 이때, 읽은 주소의 데이터는 클리어되지 않습니다.



스택된 데이터 수를 알 때는 SNUM 명령을 사용합니다.

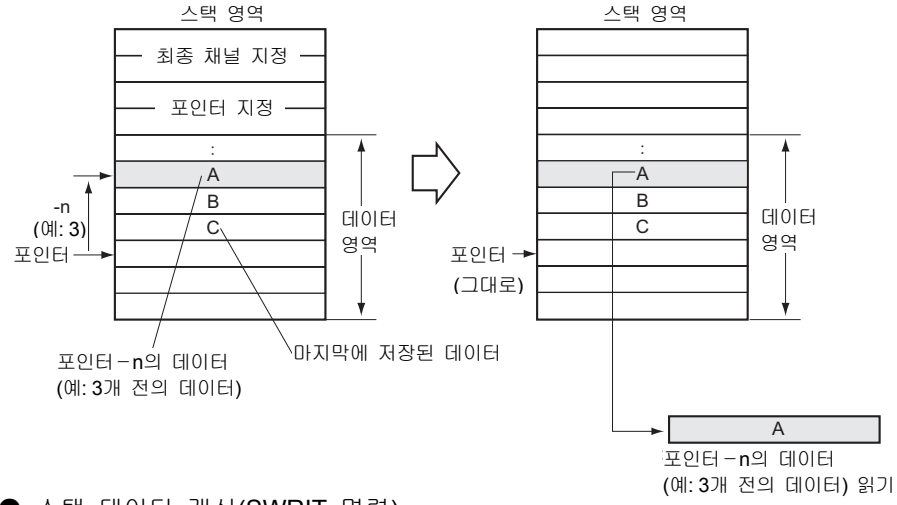
(예: 컨베이어 위의 물품 수를 알고 있을 때)

그리고 스택되어 있는 중간 데이터를 읽기(참조), 변경(갱신), 삽입, 삭제할 때는 각각 SREAD, SWRIT, SINS, SDEL 명령을 사용합니다.

(예: 컨베이어 위를 지나가는 물품을 추적하는 경우, 스택된 데이터 = 물품으로 취급하여 물품의 투입, 배출, 변경 등을 관리)

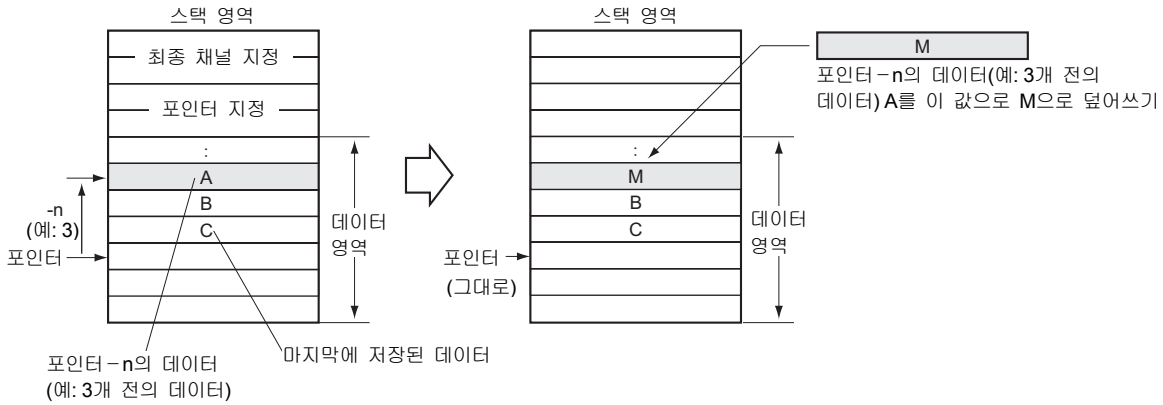
● 스택 데이터 참조(SREAD 명령)

현재 포인터가 있는 위치에서 거슬러 올라가 n개 이전의 데이터를 읽습니다.



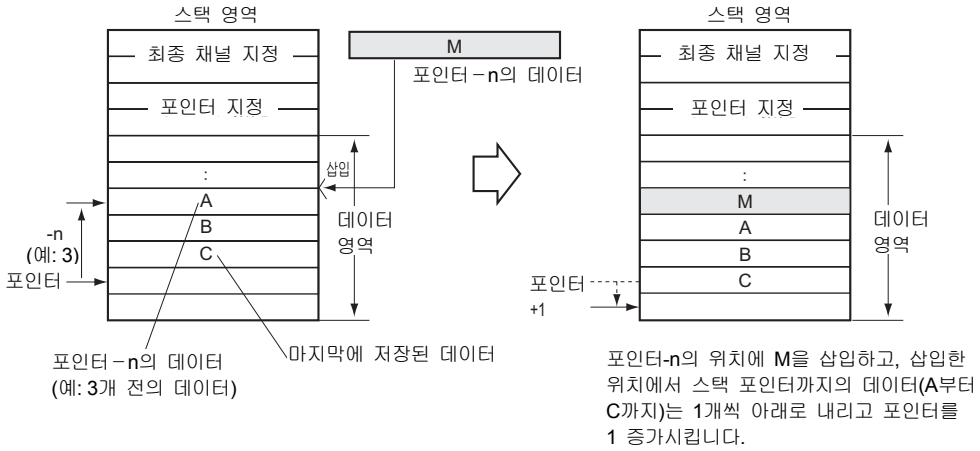
● 스택 데이터 갱신(SWRIT 명령)

현재 포인터가 있는 위치에서 거슬러 올라가 n개 이전의 데이터를 덮어 씁니다.



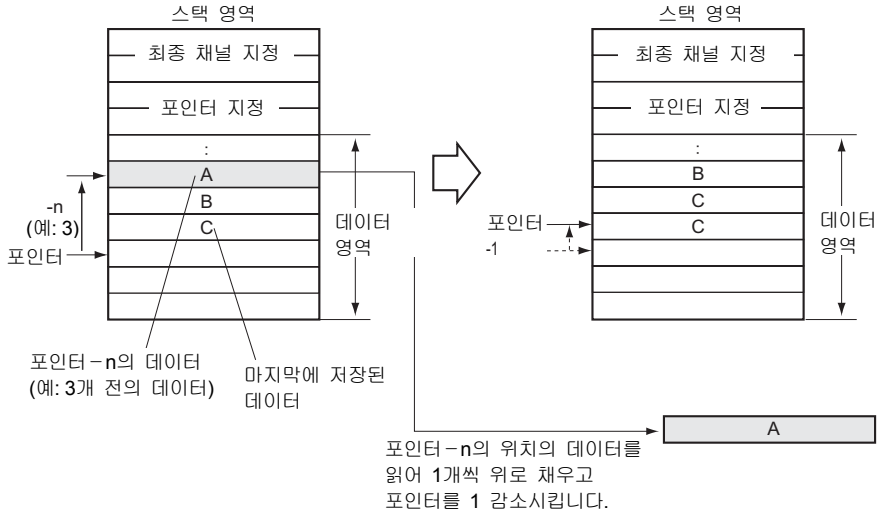
● 스택 데이터 삽입(SINS 명령)

현재 포인터가 있는 위치에서 거슬러 올라가 n개 이전의 위치에 데이터를 삽입합니다.



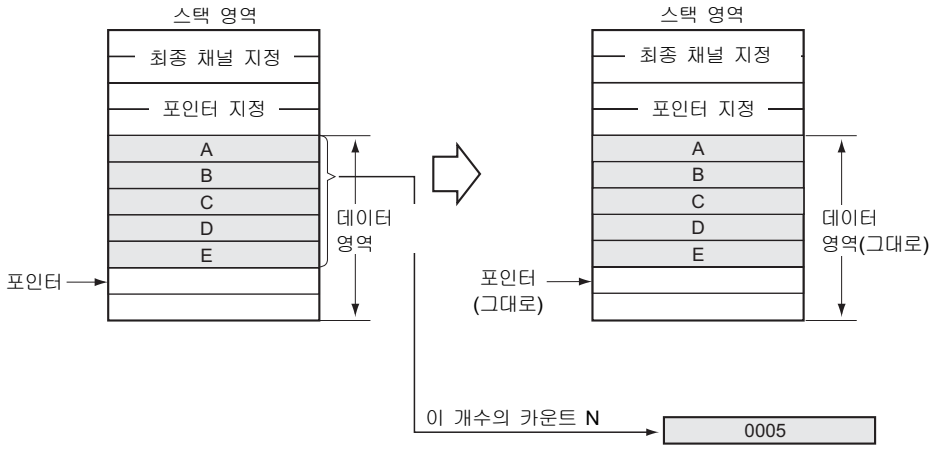
● 스택 데이터 삭제(SDEL 명령)

현재 포인터가 있는 위치에서 거슬러 올라가 n개 이전의 데이터를 삭제합니다.



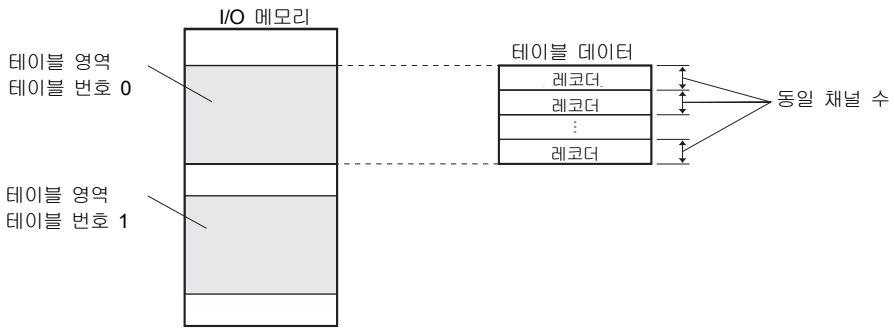
● 스택 데이터 수 계산(SNUM 명령)

스택 포인터부터 스택 영역 선두까지의 스택 데이터 수를 계산합니다.



■ 테이블 영역

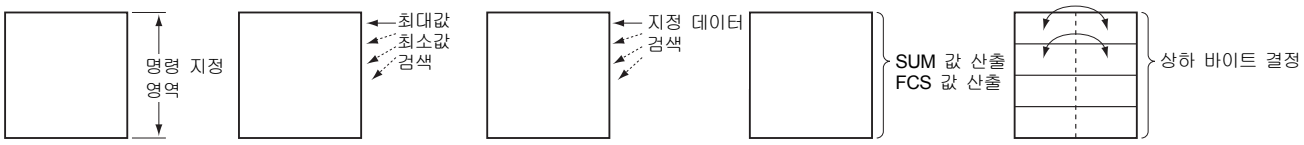
채널 수가 동일한 복수의 레코드군으로 이루어진 일련의 데이터를 테이블 데이터라고 합니다. I/O 메모리의 지정 영역에 저장된 테이블 데이터를 테이블 영역으로 등록할 수 있습니다(DIM 명령). 테이블 영역은 최대 16개 등록할 수 있으며 0~15의 테이블 번호로 참조합니다.



일반적으로 테이블 데이터에는 인덱스 레지스터(IR)를 사용한 간접 참조가 편리합니다. 등록된 테이블 영역의 경우 SETR(레코드 위치 결정) 명령 및 GETR(레코드 위치 읽기) 명령으로 레코드의 주소를 쉽게 산출할 수 있습니다.

■ 각 명령 지정 영역

지정 범위의 데이터 중 최대값(MAX 명령), 최소값(MIN 명령), 지정 데이터의 검색(SRCH 명령), SUM값 산출(SUM 명령), FCS 값 산출(FCS 명령), 상하 바이트 교환(SWAP 명령)을 수행할 수 있습니다.



SSET

명령 이름	니모닉	동작 옵션	펑션 번호	기능
스택 영역 설정	SSET	@SSET	630	지정된 CH에서 지정된 CH 수의 스택 영역을 설정하고, 동시에 데이터 저장 영역을 0으로 클리어합니다.

심볼	SSET
	

특정 영역 내의 사용 여부

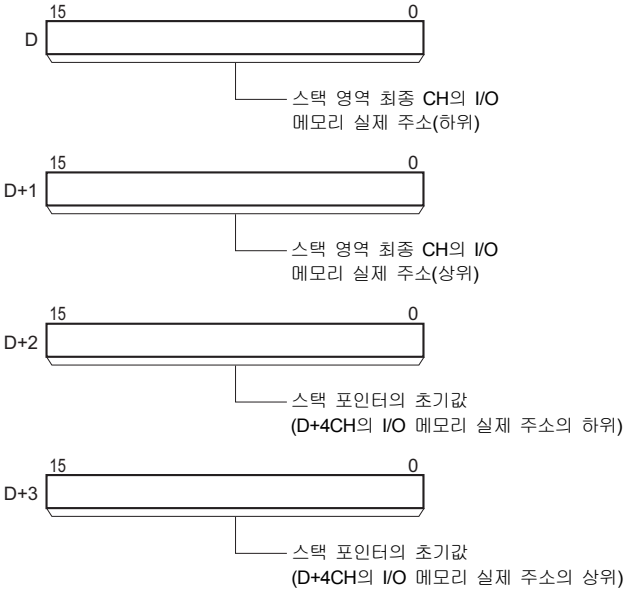
영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

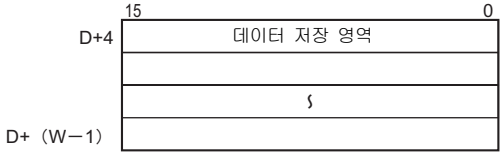
피연산자	내용	데이터 형식	크기
D	스택 영역 하위 CH 번호	UINT	가변
W	영역 CH 번호	UINT	1

D: 스택 영역 하위 CH 번호

D~D+3: 스택 관리 정보(4CH 고정)



D+4~: 데이터 저장 영역



주: D~D+(W－1)은 동일 영역 종류이어야 합니다.

W: 영역 CH 수

10진수 &5~65535 또는 16진수 #0005~FFFF

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
D	○	○	○	○	○	○	○	○	○	○	－	－	－	○	－	－	－	－
W											○	○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· W의 데이터가 10진수 &5~65535 또는 16진수 #0005~FFFF의 범위에 없을 때 ON · 그 이외일 때 OFF

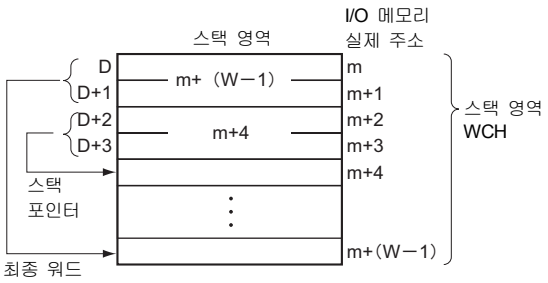
주:

· 영역 CH 수(W)에는 위의 스택 관리 정보(스택 영역 최종 메모리 주소+스택 포인터)를 포함하여 지정하므로 5 이상의 값을 지정합니다.

기능

D로 지정한 스택 영역 하위 CH 번호부터 W로 지정한 영역 CH 수 분량을 스택 영역으로 확보합니다. 확보된 스택 영역 선두의 2CH(D+1, D+0)에는 스택 영역 최종 CH의 I/O 메모리 실제 주소(32비트)를 출력하고, 다음 2CH(D+3, D+2)에는 스택 포인터로 스택 영역 선두+4CH(D+4), 즉 데이터 저장 영역의 선두 I/O 메모리 실제 주소(32비트)를 출력합니다.

또한 동시에 데이터 저장 영역(D+4~D+(W－1))은 모두 0으로 클리어됩니다.



힌트

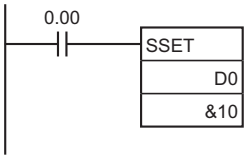
· 이 명령은 포인터를 가진 스택 영역을 확보하기 위한 명령입니다. 이 명령을 사용하여 스택 영역을 확보한 후,

1) PUSH(스택 데이터 저장) 명령을 사용하여, 확보된 스택 영역에 데이터를 1CH(채널)씩 저장합니다.
2) 그 후, 먼저 입력된 데이터에서 가져올 때는 FIFO(선입선출) 명령을 실행합니다. 읽을 때는 나중에 입력된 데이터부터 LIFO(후입선출) 명령을 실행합니다.
   PUSH, FIFO, LIFO 명령을 실행하면 스택 관리 정보 내의 스택 포인터 값이 자동으로 갱신됩니다. 평상시에는 스택 관리 정보를 사용자 프로그램에서 고려해야 할 필요가 없습니다. 위 명령 이외에 스택의 내용을 참조하는 경우는 스택 포인터의 값을 인덱스 레지스터(IR)에 설정하여 간접 참조하십시오.

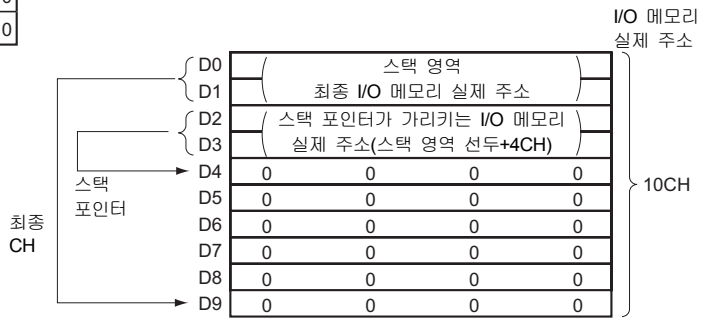
관련 명령

· PUSH(스택 데이터 저장) 명령: 지정한 스택 영역에 데이터를 저장합니다.
· FIFO(선입선출) 명령: 지정된 스택 영역에서 처음 저장한 데이터를 읽습니다.
· LIFO(후입선출) 명령: 지정된 스택 영역에서 마지막에 저장된 데이터를 읽습니다.

샘플 프로그램



입력 릴레이 0.00이 ON일 때 D0~D9의 10CH를 스택 영역으로 설정합니다.



D0과 D1에는 스택 영역 최종 CH의 I/O 메모리 실제 주소가 자동으로 저장됩니다.
D2와 D3에는 스택 포인터의 I/O 메모리 실제 주소가 저장됩니다.
(스택 영역 선두 D4의 I/O 메모리 실제 주소가 자동으로 저장됩니다.)

PUSH

명령 이름	니모닉	동작 옵션	평션 번호	기능
스택 데이터 저장	PUSH	@PUSH	632	지정된 스택 영역에 데이터를 저장합니다. 동시에 스택 포인터의 내용을 1 증가시킵니다.

심볼	PUSH
	PUSH D: 스택 영역 하위 CH 번호 S: 저장 데이터 CH 영역

특정 영역 내의 사용 여부

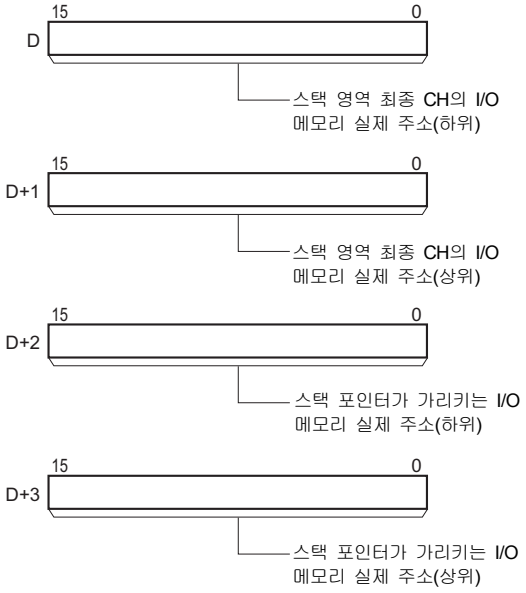
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

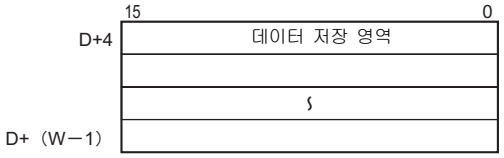
피연산자	내용	데이터 형식	크기
D	스택 영역 하위 CH 번호	UINT	가변
S	저장 데이터 CH 번호	UINT	1

D: 스택 영역 하위 CH 번호

D~D+3: 스택 관리 정보(4CH 고정)



D+4~: 데이터 저장 영역



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
D	○	○	○	○	○	○	○	○	○	○	−	−	−	○	−	−	−	−
S											○	○						

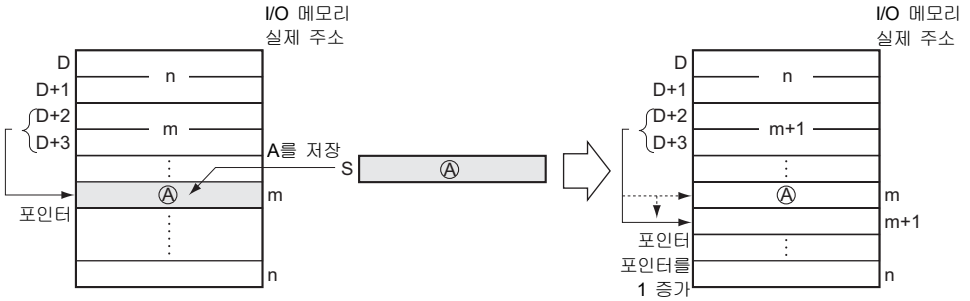
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 스택 포인터(D+3, D+2)로 표시한 주소가 스택 영역의 최종 CH를 초과했을 때 (스택 오버 플로) ON · 그 이외일 때 OFF

기능

D로 지정한 스택 영역의 스택 포인터(D+3, D+2)로 표시한 주소에 S로 지정한 데이터(1CH)를 출력하고 스택 포인터의 내용을 1 증가시킵니다.

스택 영역은 SSET(스택 영역 설정) 명령으로 미리 설정해야 합니다.

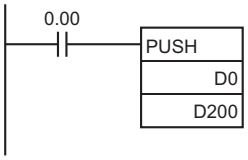


힌트

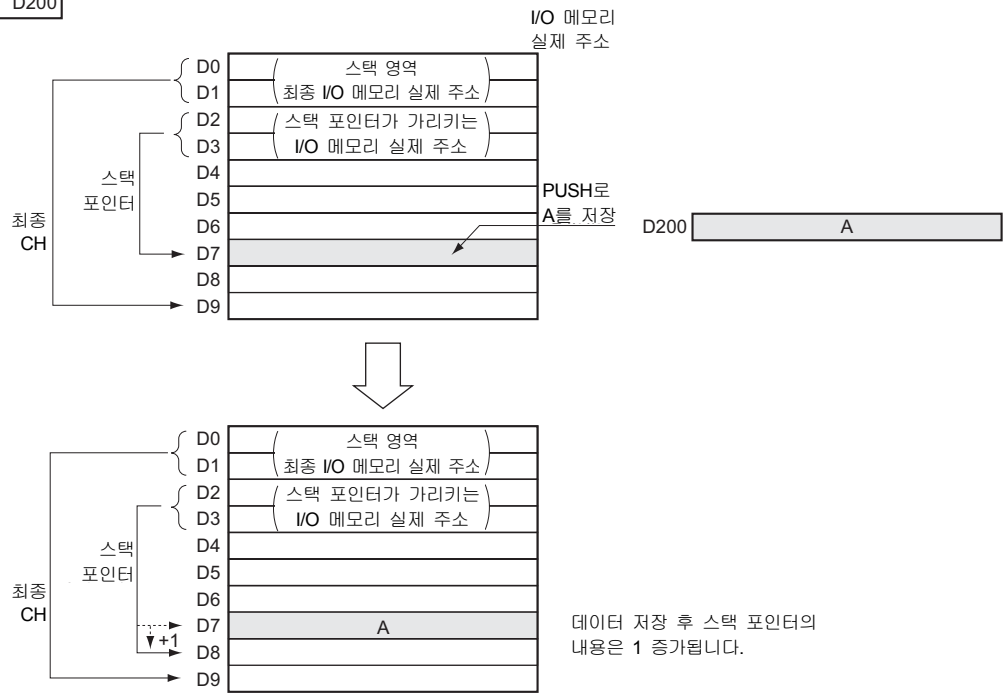
· 스택 포인터는 이 명령으로 데이터를 저장한 후, 최종 데이터의 다음 주소를 지정합니다.

· 이 명령으로 데이터를 저장한 후, FIFO(선입선출) 명령 또는 LIFO(후입선출) 명령을 사용할 수 있습니다.

샘플 프로그램



입력 릴레이 0.00이 ON이면 D0부터 시작하는 스택 영역의 현재 스택 포인터가 가리키는 위치에 D200의 내용을 저장합니다. 현재 스택 포인터가 D7을 가리키고 있는 경우 D7에 저장합니다.



LIFO/FIFO

명령 이름	니모닉	동작 옵션	평션 번호	기능
후입선출	LIFO	@LIFO	634	지정된 스택 영역의 스택 포인터의 데이터를 1 감소시키고, 그 주소에서 데이터(마지막에 저장한 데이터)를 읽습니다.
선입선출	FIFO	@FIFO	633	지정된 스택 영역에서 처음 저장된 데이터를 읽습니다.

심볼	LIFO	FIFO
	LIFO S: 스택 영역 하위 CH 번호 D: 출력 대상 CH 번호	FIFO S: 스택 영역 하위 CH 번호 D: 출력 대상 CH 번호

특정 영역 내의 사용 여부

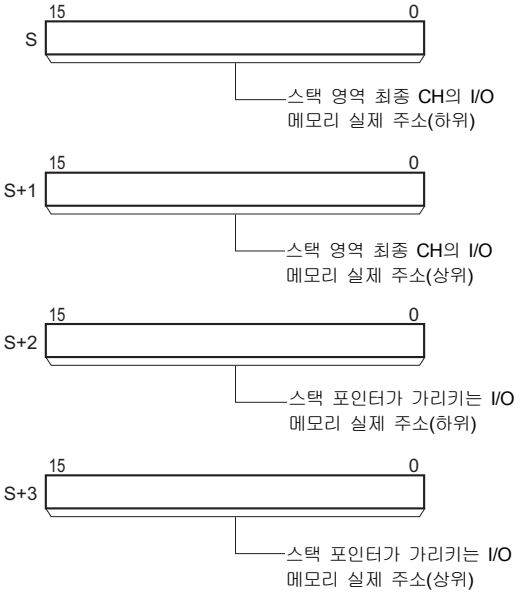
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

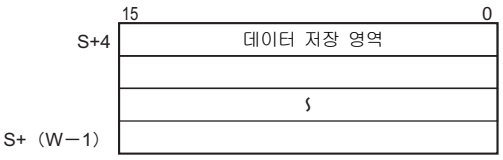
피연산자	내용	데이터 형식	크기
S	스택 영역 하위 CH 번호	UINT	가변
D	출력 대상 CH 번호	UINT	1

S: 스택 영역 하위 CH 번호

S~S+3: 스택 관리 정보(4CH 고정)



S+4~: 데이터 저장 영역



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D												○						

조건 플래그의 동작

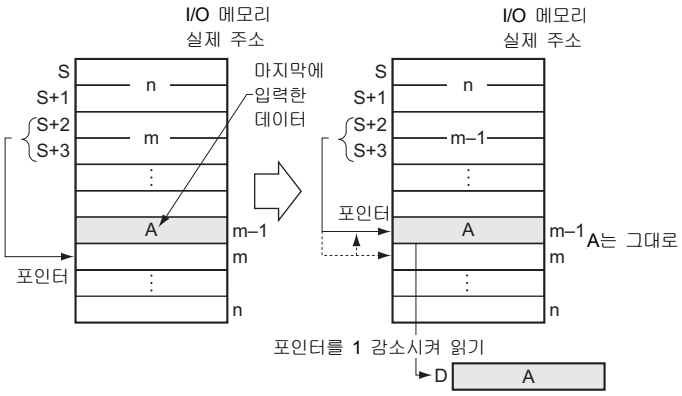
이름	라벨	내용
에러 플래그	P_ER	· 스택 포인터(S+3, S+2)의 내용이 데이터 저장 영역의 선두 I/O 메모리 실제 주소 이하일 때(스택 언더 플로) ON · 그 이외일 때 OFF

기능

■ LIFO

S로 지정한 스택 영역의 스택 포인터(S+3, S+2)의 데이터를 1 감소시키고 그 주소에서 데이터를 읽어서 D에 출력합니다. 읽은 위치의 내용은 변경되지 않습니다.

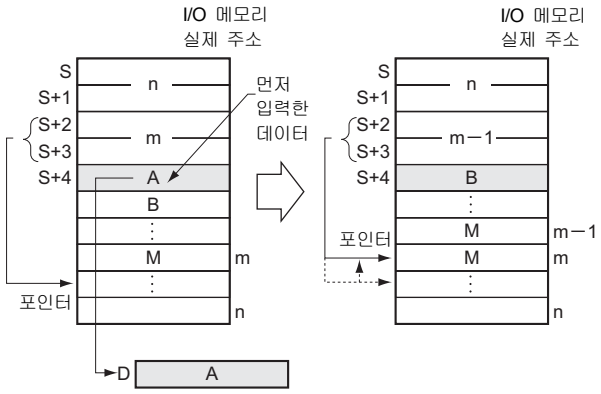
스택 영역은 SSET(스택 영역 설정) 명령으로 미리 설정해야 합니다.



■ FIFO

S로 지정한 스택 영역의 데이터 저장 영역 선두(S+4)에서 데이터를 읽어서 D에 출력합니다. 그 후, 스택 포인터(S+3, S+2)의 내용을 1 감소시키고 데이터 저장 영역 선두+1(S+5)~스택 포인터가 가리키는 위치의 데이터를 하위 CH 측으로 1CH(채널) 이동합니다. 이때, 데이터 저장 영역 선두(S+4)의 원래 데이터는 소멸됩니다. 또한 스택 포인터의 위치 데이터는 이동 후에도 변경되지 않습니다.

스택 영역은 SSET(스택 영역 설정) 명령으로 미리 설정해야 합니다.



힌트

■ LIFO

· 이 명령은 PUSH(스택 데이터 저장) 명령과 조합하여 사용합니다. PUSH 명령으로 데이터를 저장한 후 이 명령을 사용하면 후입선출 처리가 가능합니다.

· PUSH 명령으로 데이터를 저장한 후 스택 포인터는 최종 데이터의 다음 주소를 지정합니다. 이 명령은 스택 포인터의 내용을 1 감소시키고 그 주소의 데이터(최종 데이터)를 읽습니다.

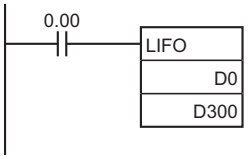
■ FIFO

· 이 명령은 PUSH(스택 데이터 저장) 명령과 조합하여 사용합니다. PUSH 명령으로 데이터를 저장한 후, 이 명령을 사용하면 선입선출 처리가 가능합니다.

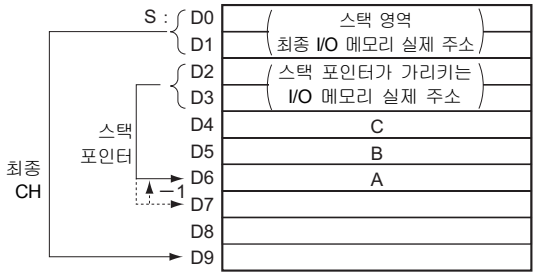
· 이 명령은 스택 영역의 선두 데이터를 가져온 후 가져온 데이터를 삭제하여 앞 채우기를 실행합니다.

샘플 프로그램

■ LIFO

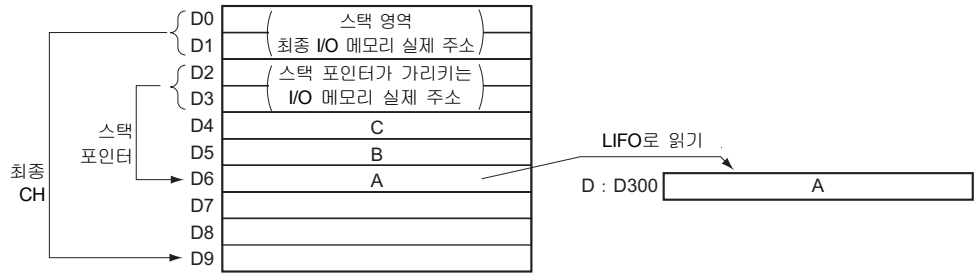


0.00이 ON이면 D0부터 시작하는 스택 영역의 스택 포인터를 1 감소시킨 데이터 1CH(D6)를 읽어 D300에 저장합니다.

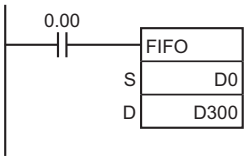




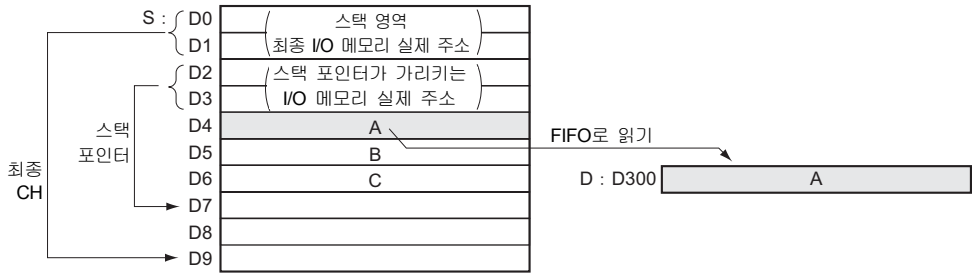
스택 포인터의 내용을 1 감소시키고 D6의 데이터를 읽습니다.
읽은 D6의 내용(A)은 그대로 남아 있습니다.



■ FIFO

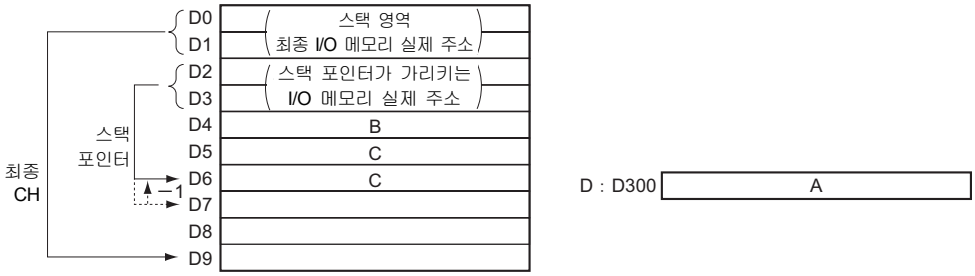


0.00이 ON일 때 D0부터 시작하는 스택 영역의 선두 위치(스택 포인터 I/O 메모리 실제 주소 저장 위치+1CH)의 1CH(D4)를 가져와 D300에 저장합니다.



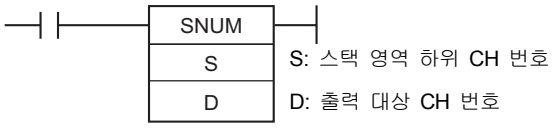


데이터를 읽은 후 스택 포인터의 내용은 1 감소되고, D4의 내용은 소멸되며, D5의 내용이 앞으로 당겨서 이동됩니다. (D5~D6이 D4~D5로 이동합니다.)



SNUM

명령 이름	니모닉	동작 옵션	평션 번호	기능
스택 데이터 수 출력	SNUM	@SNUM	638	지정된 스택 영역 내의 스택 데이터 수(CH 수)를 계산합니다.

심볼	SNUM
	

특정 영역 내의 사용 여부

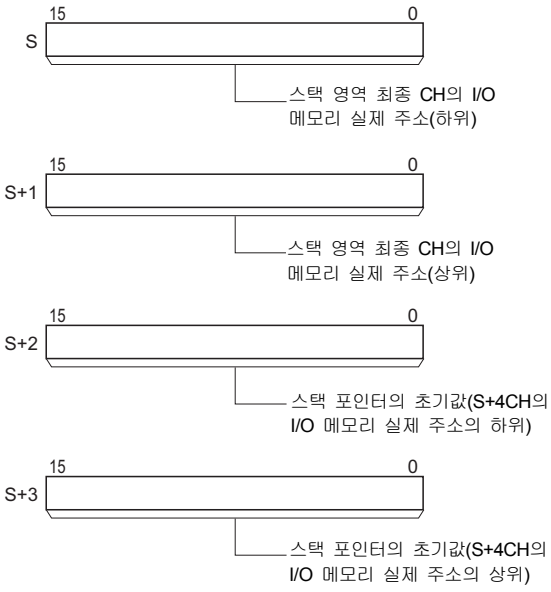
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

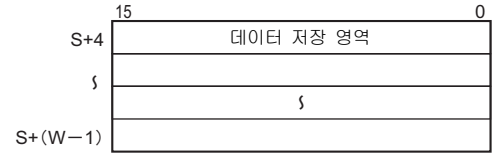
피연산자	내용	데이터 형식	크기
S	스택 영역 하위 CH 번호	UINT	가변
D	출력 대상 CH 번호	UINT	1

S: 스택 영역 하위 CH 번호

S~S+3: 스택 관리 정보(4CH 고정)



S+4~: 데이터 저장 영역



■ 영역 종류

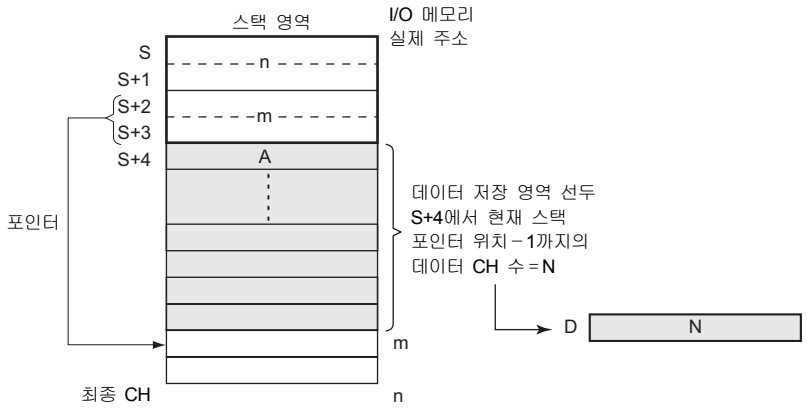
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D												○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF
= 플래그	P_EQ	· 데이터 저장 영역의 데이터 수(D에 대한 출력값)가 0(데이터 없음)일 때 ON · 그 이외일 때 OFF

기능

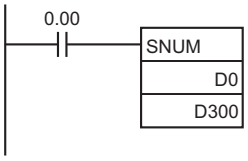
S로 지정한 스택 영역의 데이터 저장 영역 선두(S+4)부터 현재 스택 포인터(S+3, S+2)가 가리키는 위치에서 1 감소시킨 위치까지의 데이터의 CH 수를 계산하여 그 CH 수를 D에 출력합니다. 이때, 데이터 저장 영역의 데이터 및 스택 포인터의 위치는 변경되지 않습니다. 스택 영역은 SSET (스택 영역 설정) 명령으로 미리 설정해야 합니다.



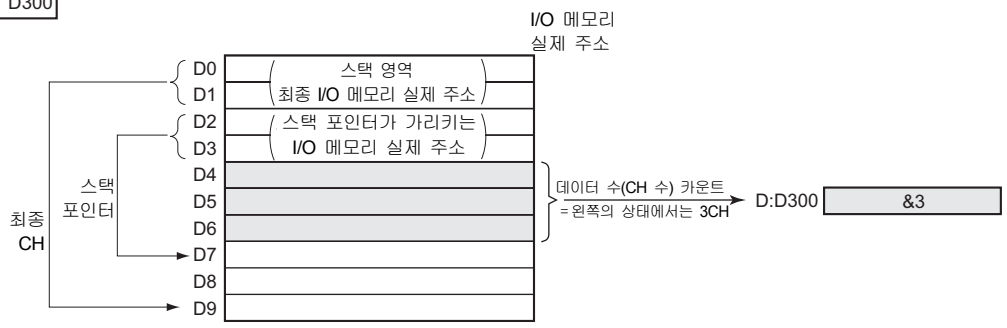
힌트

· 이 명령은 현재 컨베이어 위에 있는 워크의 수를 계산하는 경우 등에 사용합니다.

샘플 프로그램

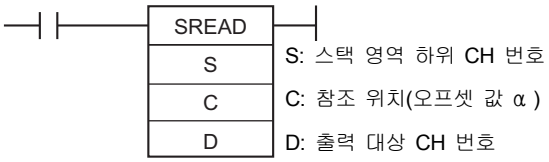


입력 릴레이 0.00이 ON이면 D0에서 시작하는 스택 영역의 데이터 저장 영역 선두 CH의 D4부터 현재 스택 포인터(D7)가 가리키는 위치－1(D6)까지의 데이터 수(CH 수)를 카운트 하여 D300에 저장합니다.



SREAD

명령 이름	니모닉	동작 옵션	평션 번호	기능
스택 데이터 참조	SREAD	@SREAD	639	지정된 스택 영역에서 지정된 데이터(현재 스택 포인터가 있는 위치에서 지정된 개수 만큼 앞에 있는 데이터)를 읽습니다.

	SREAD
심볼	

특정 영역 내의 사용 여부

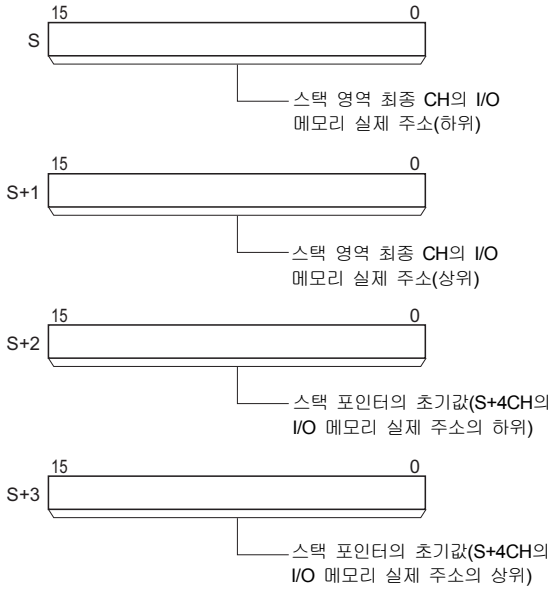
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

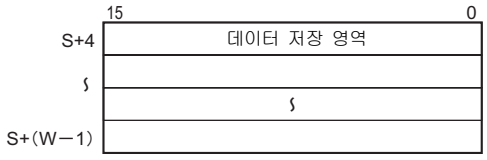
피연산자	내용	데이터 형식	크기
S	스택 영역 하위 CH 번호	UINT	가변
C	참조 위치(오프셋 값 α)	UINT	1
D	출력 대상 CH 번호	UINT	1

S: 스택 영역 하위 CH 번호

S~S+3: 스택 관리 정보(4CH 고정)



S+4~: 데이터 저장 영역



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
C											○	○						
D											–							

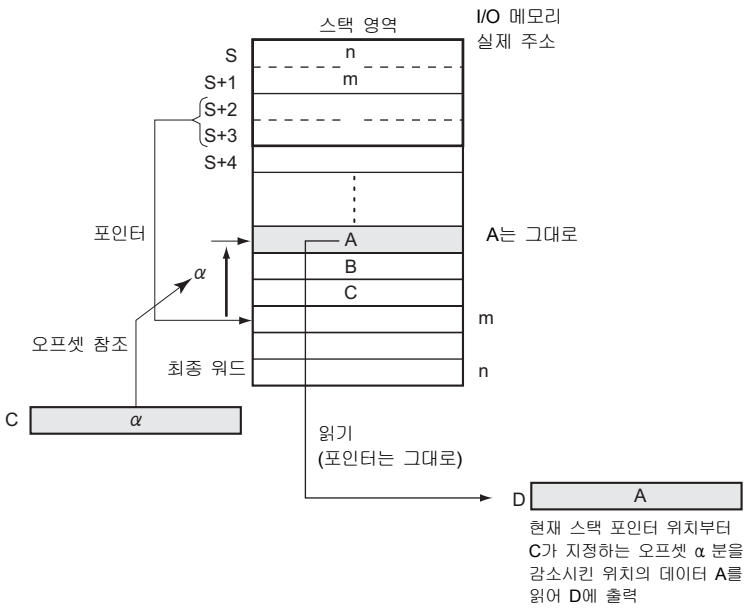
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 지정된 읽기 위치가 스택 데이터 영역 밖일 때 ON · 참조 위치가 0 또는 스택 데이터의 최대 영역(#FFFB)을 초과했을 때 ON · 스택 포인터가 이미 데이터 저장 선두의 I/O 메모리 실제 주소 이하일 때 (스택 언더 플로) ON · 그 이외일 때 OFF
= 플래그	P_EQ	· D의 데이터(출력 데이터)가 0일 때 ON · 그 이외일 때 OFF

기능

S로 지정한 스택 영역의 스택 포인터(S+3, S+2)가 가리키는 위치부터, C로 지정한 CH 수(오프셋)만큼 감소시킨 위치의 데이터를 읽어서 D에 출력합니다. 이때, 데이터 저장 영역의 데이터 및 스택 포인터의 위치는 변경되지 않습니다.

스택 영역은 SSET(스택 영역 설정) 명령으로 미리 설정해야 합니다.



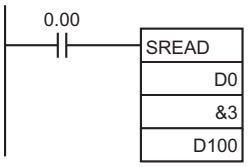
힌트

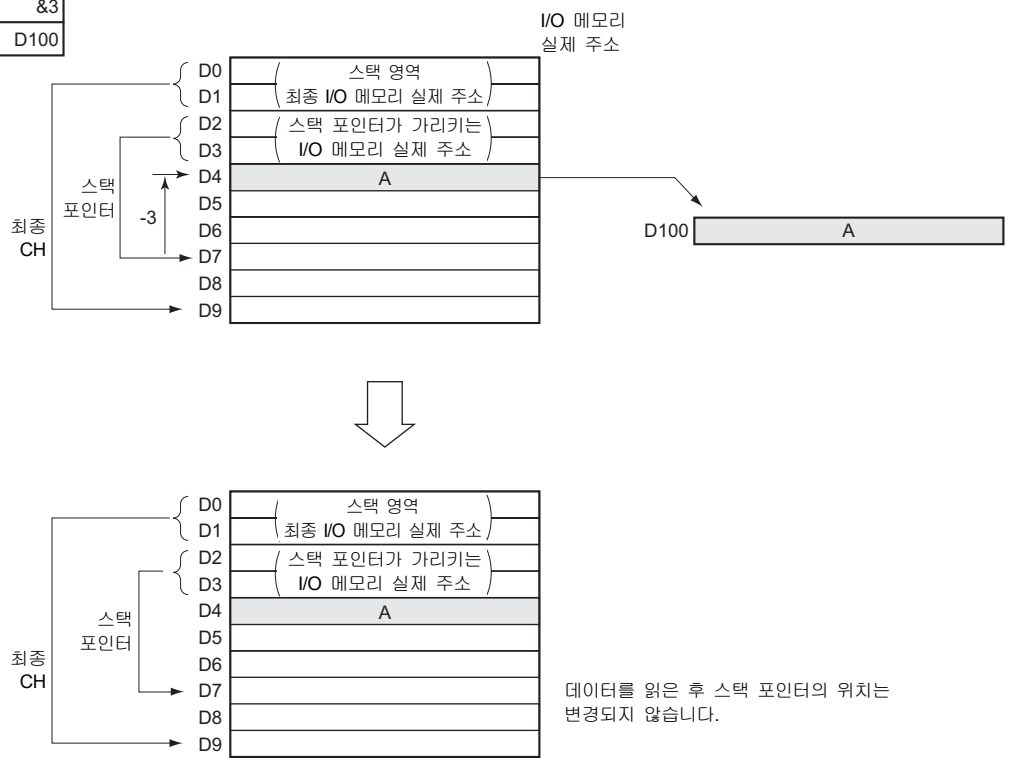
· 이 명령은 현재 컨베이어 위에 있는 워크 중 특정 위치(최종 투입 워크부터 지정된 개수 이전)의 워크에 대한 데이터를 읽는 경우 등에 사용합니다.

SREAD

샘플 프로그램

입력 릴레이 0.00이 ON이면 D0에서 시작하는 스택 영역의 현재 스택 포인터가 가리키는 위치(D7)부터 3CH분을 감소시킨 위치(D4)의 내용을 읽어 D100에 저장합니다. 현재 스택 포인터의 위치는 변경되지 않습니다.



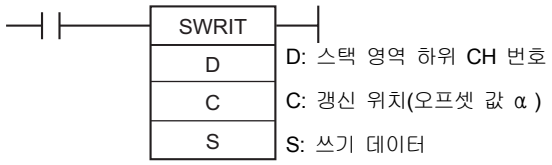


데이터를 읽은 후 스택 포인터의 위치는 변경되지 않습니다.

SWRIT

명령 이름	니모닉	동작 옵션	펑션 번호	기능
스택 데이터 갱신	SWRIT	@SWRIT	640	지정된 스택 영역에서 지정된 데이터(현재 스택 포인터가 있는 위치에서 지정된 개수 만큼 앞에 있는 데이터)의 위치에 다른 데이터를 덮어 씁니다.

심볼	SWRIT



특정 영역 내의 사용 여부

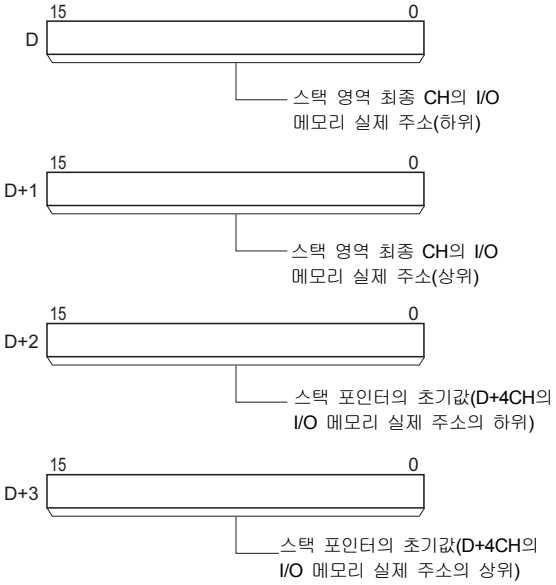
영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

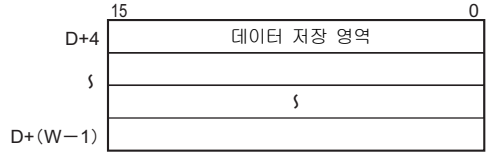
피연산자	내용	데이터 형식	크기
D	스택 영역 하위 CH 번호	UINT	가변
C	갱신 위치(오프셋 값 α)	UINT	1
S	쓰기 데이터	UINT	1

D: 스택 영역 하위 CH 번호

D~D+3: 스택 관리 정보(4CH 고정)



D+4~: 데이터 저장 영역



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
D	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
C											○	○						
S																		

SWRIT

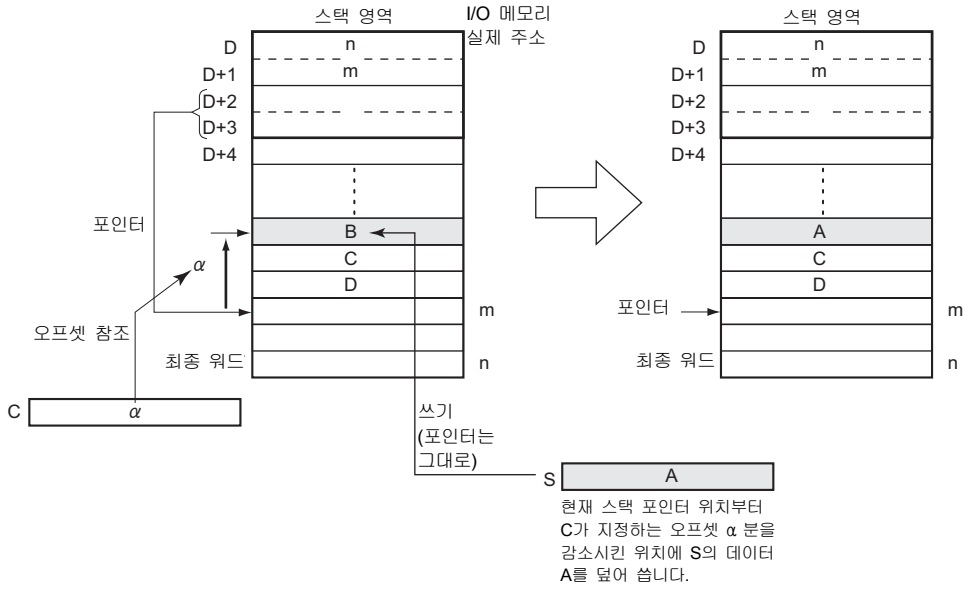
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 지정된 쓰기 위치가 스택 데이터 영역 밖일 때 ON · 갱신 위치가 0 또는 스택 데이터의 최대 영역(#FFFB)을 초과했을 때 ON · 스택 포인터가 이미 데이터 저장 선두의 I/O 메모리 실제 주소 이하일 때 (스택 언더 플로) ON · 그 이외일 때 OFF

기능

D로 지정한 스택 영역의 스택 포인터(D+3, D+2)가 가리키는 위치부터 C로 지정한 CH 수(오프셋)만큼 감소시킨 위치에 S로 지정한 데이터를 덮어 씁니다. 이때, 데이터 저장 영역의 데이터 및 스택 포인터의 위치는 변경되지 않습니다.

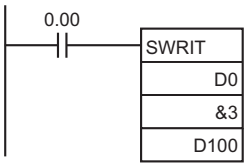
스택 영역은 SSET(스택 영역 설정) 명령으로 미리 설정해야 합니다.



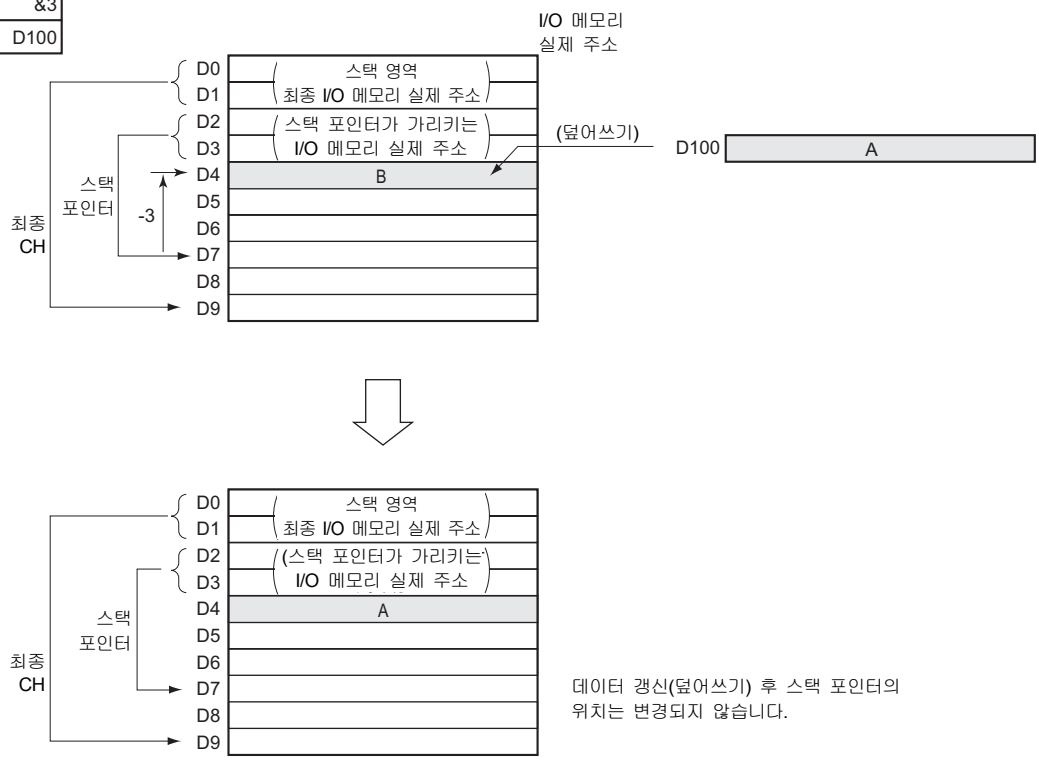
힌트

· 이 명령은 현재 컨베이어 위에 있는 워크 중 특정 위치(최종 투입 워크부터 지정된 개수 이전)의 워크에 대한 데이터를 갱신(변경)하는 경우 등에 사용합니다.

샘플 프로그램



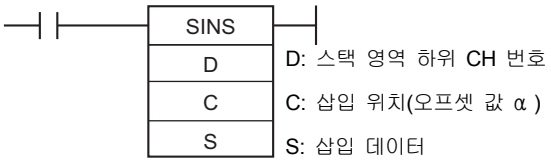
입력 릴레이 0.00이 ON이면 D0에서 시작하는 스택 영역의 현재 스택 포인터가 가리키는 위치(D7)부터 3CH분을 감소시킨 위치(D4)에 D100의 내용을 저장합니다. 스택 포인터의 위치는 변경되지 않습니다.



데이터 갱신(덮어쓰기) 후 스택 포인터의 위치는 변경되지 않습니다.

SINS

명령 이름	니모닉	동작 옵션	평션 번호	기능
스택 데이터 삽입	SINS	@SINS	641	지정된 스택 영역에서 지정된 데이터 위치(현재 스택 포인터가 있는 위치에서 지정된 개수 만큼 앞에 있는 위치)에 다른 데이터를 삽입합니다.

심볼	SINS
	

특정 영역 내의 사용 여부

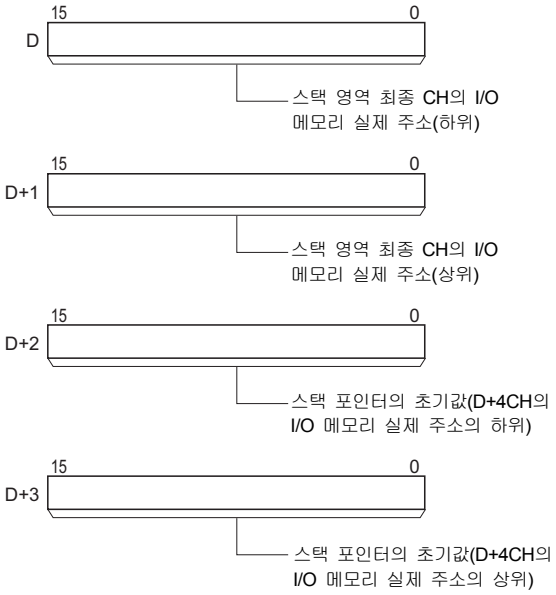
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

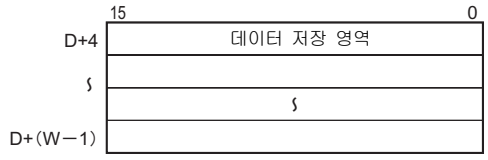
피연산자	내용	데이터 형식	크기
D	스택 영역 하위 CH 번호	UINT	가변
C	삽입 위치(오프셋 값 α)	UINT	1
S	삽입 데이터	UINT	1

D: 스택 영역 하위 CH 번호

D~D+3: 스택 관리 정보(4CH 고정)



D+4~: 데이터 저장 영역



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
D	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
C											○	○						
S																		

조건 플래그의 동작

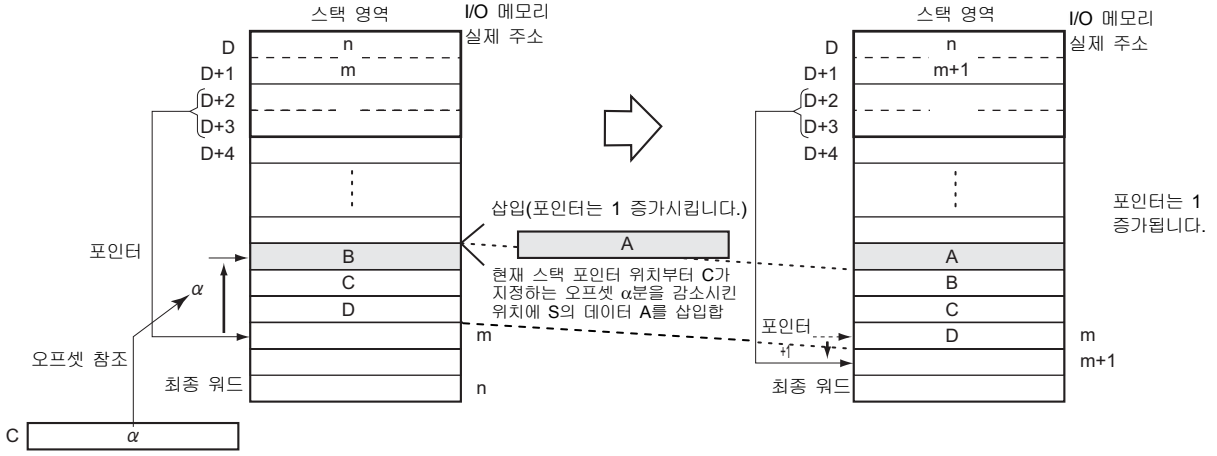
이름	라벨	내용
에러 플래그	P_ER	· 스택 포인터(D+3, D+2)로 표시한 주소가 스택 영역의 최종 CH를 초과했을 때 (스택 오버 플로) ON · 삽입 위치가 스택 데이터의 최대 영역-1(#FFFA)을 초과했을 때 ON · 그 이외일 때 OFF

기능

D로 지정한 스택 영역의 스택 포인터(D+3, D+2)가 가리키는 위치부터 C로 지정한 CH 수(오프셋)만큼 감소시킨 위치에 S로 지정한 데이터 1CH를 삽입합니다. 이때, 삽입한 위치부터 스택 포인터에서 1을 감소시킨 위치까지의 데이터는 모두 +1CH(아래로) 이동합니다(스택 포인터 위치의 내용은 덮어씀). 동시에 스택 포인터(D+3, D+2)의 데이터를 1 증가시킵니다.

1CH를 삽입하려면 데이터 저장 영역의 마지막 1CH에 데이터가 저장되어 있지 않아야 합니다. 저장되어 있는 경우, 에러가 발생하며 삽입되지 않습니다.

스택 영역은 SSET(스택 영역 설정) 명령으로 미리 설정해야 합니다.

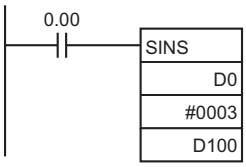


힌트

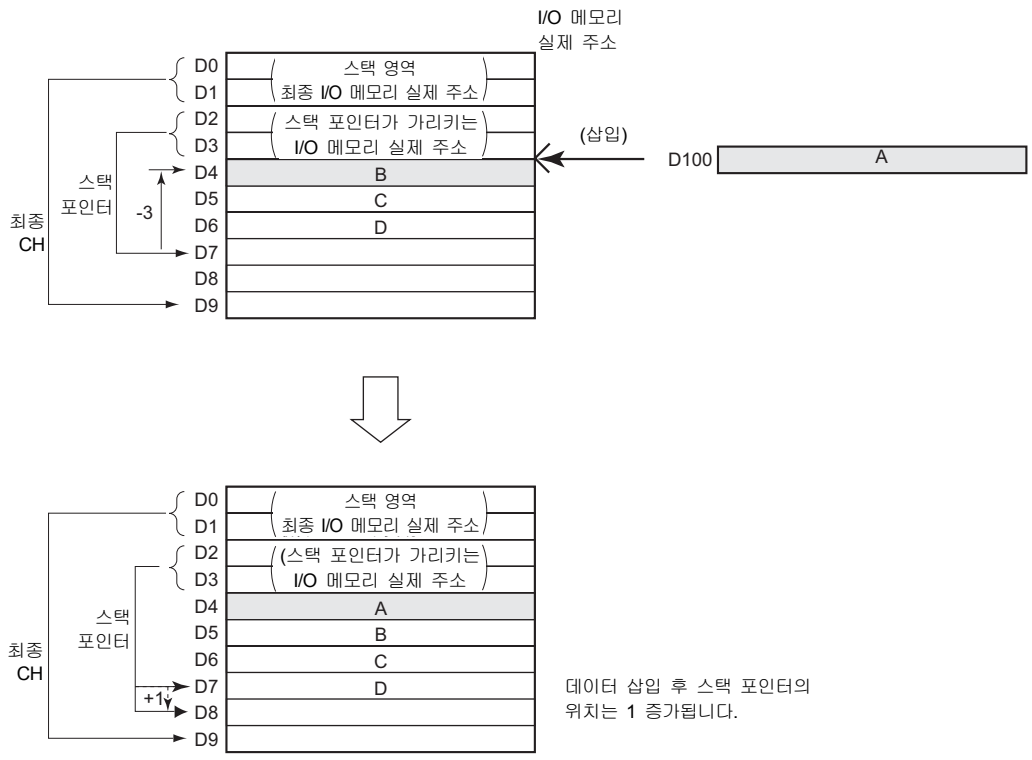
· 이 명령은 현재 컨베이어 위에 있는 워크 중 특정 위치(최종 투입 워크에서 지정된 개수 이전 위치)의 워크를 삽입하는 경우 등에 사용합니다.

SINS

샘플 프로그램



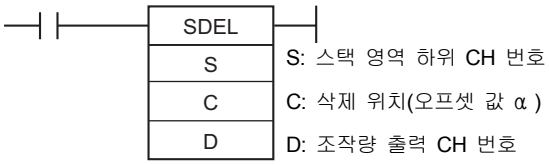
입력 릴레이 0.00이 ON이면 D0에서 시작하는 스택 영역의 현재 스택 포인터가 가리키는 위치(D7)부터 3CH분을 감소시킨 위치(D4)에 D100의 내용을 삽입합니다. 이때, 삽입한 위치(D4)부터 스택 포인터-1(D6)까지의 데이터는 모두 +1CH 아래로 이동합니다(D7의 내용은 덮어씀). 동시에 스택 포인터(D7)가 1 증가하여 D8을 가리킵니다.



데이터 삽입 후 스택 포인터의 위치는 1 증가됩니다.

SDEL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
스택 데이터 삭제	SDEL	@SDEL	642	지정된 스택 영역에서 지정된 데이터 위치(현재 스택 포인터가 있는 위치에서 지정된 개수 만큼 앞에 있는 위치)의 데이터를 삭제합니다.

심볼	SDEL
	

특정 영역 내의 사용 여부

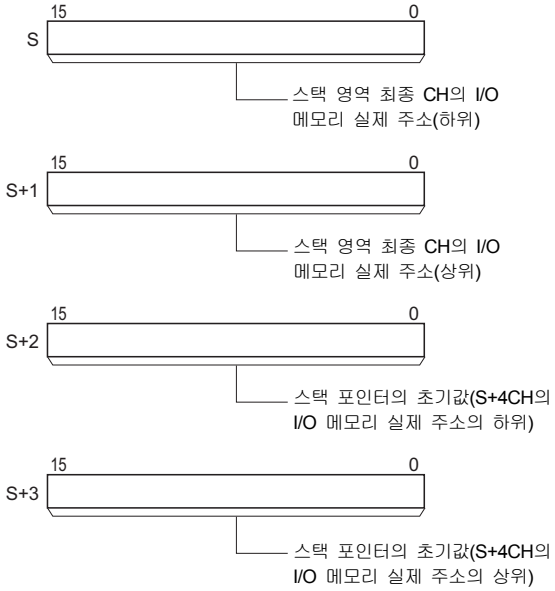
영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

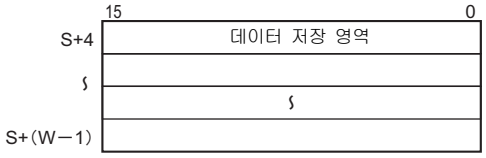
피연산자	내용	데이터 형식	크기
S	스택 영역 하위 CH 번호	UINT	가변
C	삭제 위치(오프셋 값 α)	UINT	1
D	조작량 출력 CH 번호	UINT	1

S: 스택 영역 하위 CH 번호

S~S+3: 스택 관리 정보(4CH 고정)



S+4~: 데이터 저장 영역



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
C											○	○						
D											–							

SDEL

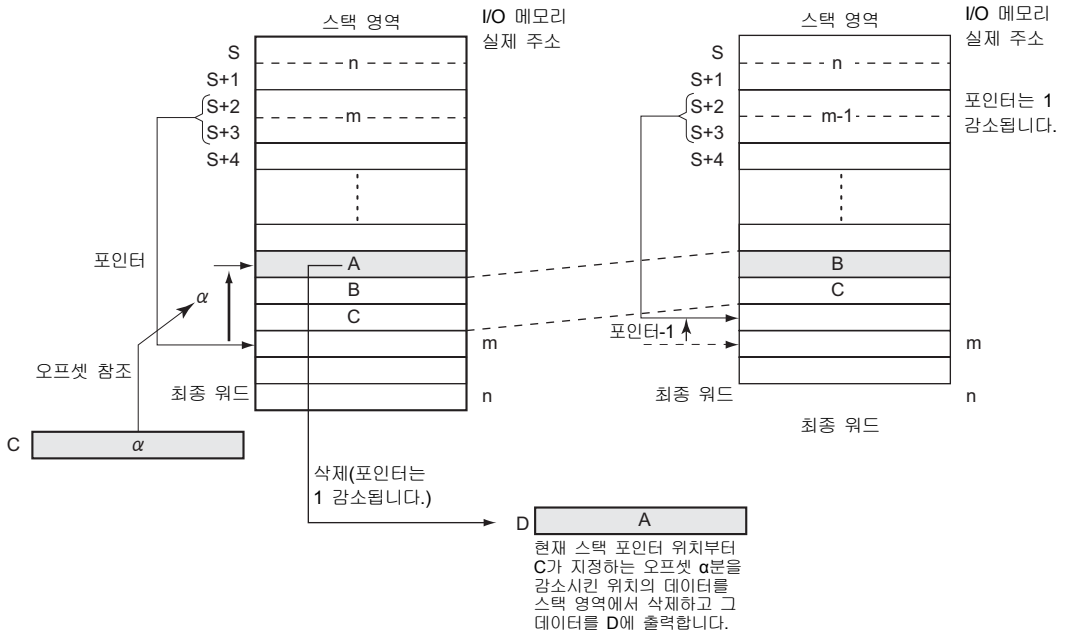
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 스택 포인터(S+3, S+2)로 가리키는 주소가 데이터 저장 영역 선두(S+4)의 I/O 메모리 실제 주소 이하일 때(스택 언더 플로) ON · 삭제 위치가 0 또는 스택 데이터의 최대 영역(#FFFB)을 초과했을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 출력(저장) 데이터가 0일 때 ON · 그 이외일 때 OFF

기능

S로 지정한 스택 영역의 스택 포인터(S+3, S+2)가 가리키는 위치부터 C로 지정한 CH 수(오프셋)만큼 감소시킨 위치의 데이터 1CH를 스택 영역에서 삭제하고, 그 데이터를 가져와서 D에 출력합니다. 이때, 삭제한 위치 +1CH부터 스택 포인터－1까지의 데이터는 모두 －1CH(위로) 이동합니다. 동시에 스택 포인터(S+3, S+2)의 데이터를 1 감소시킵니다. 이 경우, 스택 포인터－1의 데이터는 변경되지 않습니다.

스택 영역은 SSET(스택 영역 설정) 명령으로 미리 설정해야 합니다.

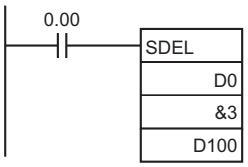


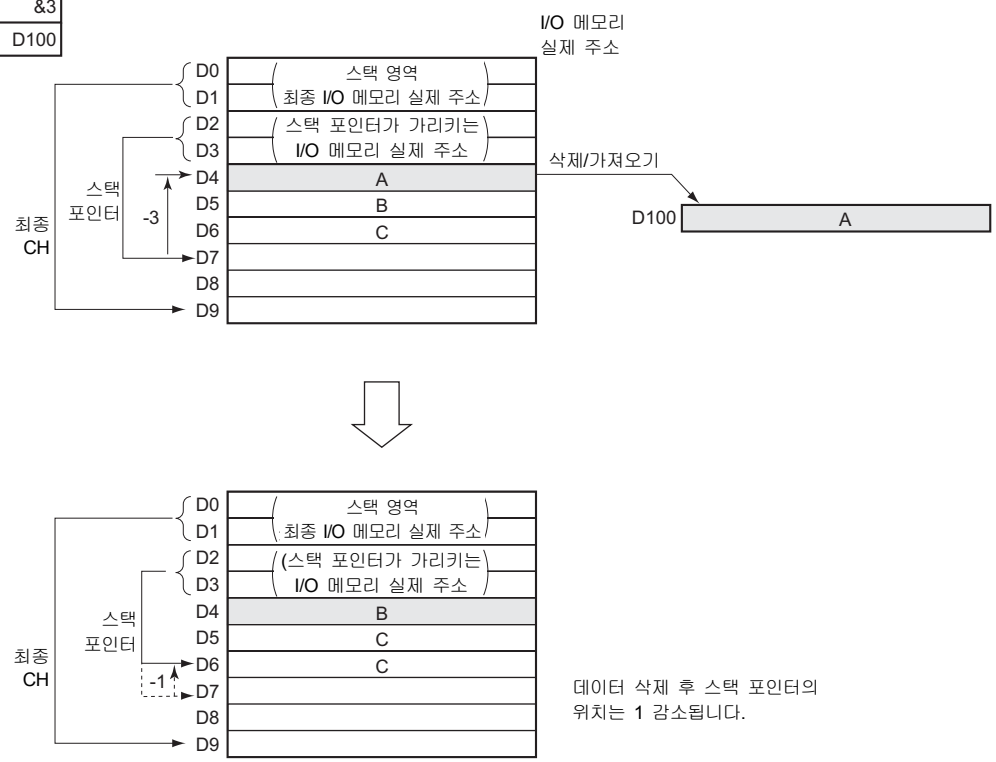
힌트

· 이 명령은 현재 컨베이어 위에 있는 워크 중 특정 위치(최종 투입 워크에서 지정된 개수 이전 위치)의 워크를 배출하는 경우 등에 사용합니다.

샘플 프로그램

입력 릴레이 0.00이 ON이면 D0에서 시작하는 스택 영역의 현재 스택 포인터가 가리키는 위치(D7)부터 3CH만큼 감소시킨 위치(D4)에서 D100의 내용을 삭제하고 그 데이터를 D100에 저장합니다. 이때, 삭제한 위치+1(D5)부터 스택 포인터－1(D6)까지의 데이터는 모두 －1CH 위로 이동합니다. 동시에 스택 포인터(D7)는 1 감소하여 D6을 가리킵니다.

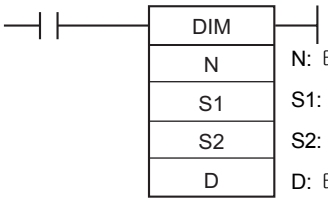




데이터 삭제 후 스택 포인터의 위치는 1 감소됩니다.

DIM

명령 이름	니모닉	동작 옵션	평션 번호	기능
테이블 영역 선언	DIM	@DIM	631	레코드 길이와 레코드 수를 선언하고 테이블 영역을 등록합니다. 16개의 테이블 영역을 정의할 수 있습니다.

	DIM
심볼	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	테이블 번호	–	1
S1	레코드 길이	UINT	1
S2	레코드 수	UINT	1
D	테이블 영역 하위 CH 번호	UINT	가변

N: 테이블 번호
0~15
S1: 레코드 길이
10진수 &1~65535 또는 16진수 #0001~FFFF
S2: 레코드 수
10진수 &1~65535 또는 16진수 #0001~FFFF

■ 영역 종류

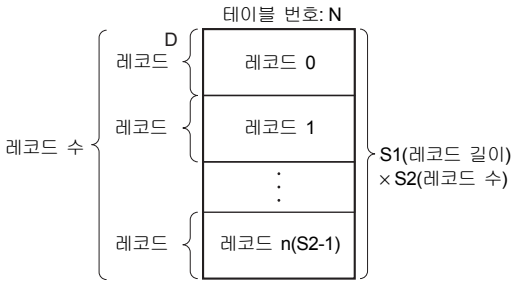
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
S1, S2	○	○	○	○	○	○	○	○	○	○		○						
D											–	–		○				

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S1, S2 중 하나의 데이터가 0일 때 ON · 그 이외일 때 OFF

기능

D로 지정한 테이블 영역 하위 CH 번호부터 레코드 길이(S1)×레코드 수(S2) 분량의 영역을 N으로 지정한 번호의 테이블 영역으로 등록합니다. 테이블 영역의 데이터는 변경되지 않습니다. 등록한 테이블의 레코드는 레코드 번호(0~(레코드 수－1))를 통해 참조합니다.



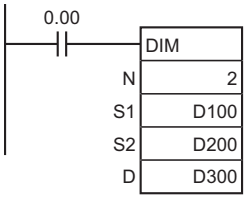
힌트

- 이 명령은 테이블 데이터에 대한 주소를 쉽게 산출하기 위한 명령인 SETR(레코드 위치 설정) 명령, GETR(레코드 위치 읽기) 명령과 조합하여 사용합니다.
- 이 명령으로 데이터를 레코드로 분할한 후 SETR(레코드 위치 설정) 명령을 사용하여 알고 싶은 레코드 번호의 시작 주소를 인덱스 레지스터에 저장합니다. 이 내용을 포인터로 하여 인덱스 레지스터를 피연산자로 간접 지정할 수 있는 명령(일반적인 명령)에서 필요한 주소의 데이터에 대한 읽기/쓰기, 검색, 비교 등의 각종 처리를 수행합니다.
  그 결과, 온도, 압력 등의 설정값을 하나의 레코드로 처리하여 품종별 레코드 테이블로 저장해 두고, 특정 조건에서 특정 품종(레코드 번호)의 설정 온도를 읽어오는 등의 각종 처리를 쉽게 수행할 수 있습니다.

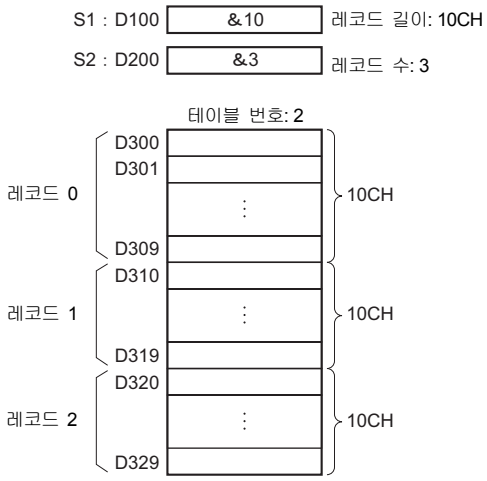
관련 명령

- SETR(레코드 위치 설정) 명령: 지정된 레코드 번호의 시작 주소에 대한 I/O 메모리 실제 주소를 인덱스 레지스터(IR)에 설정합니다.
- GETR(레코드 위치 읽기) 명령: 지정된 인덱스 레지스터(IR)의 값(I/O 메모리 실제 주소)이 포함된 레코드 번호를 출력합니다.

샘플 프로그램



0.00이 ON이면 D300부터 30CH 분량(레코드 길이 10CH×레코드 수 3)을 테이블 번호 2의 테이블 영역으로 등록합니다.



SETR

명령 이름	니모닉	동작 옵션	평션 번호	기능
레코드 위치 설정	SETR	@SETR	635	지정된 레코드의 위치(레코드 선두의 I/O 메모리 실제 주소)를 지정된 인덱스 레지스터에 저장합니다.

심볼	SETR
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	테이블 번호	–	1
S1	레코드 번호	UINT	1
D	출력 대상 인덱스 레지스터	WORD	2

N: 테이블 번호

0~15

S1: 레코드 번호

10진수 &0~65534 또는 16진수 #0000~FFFE

D: 출력 대상 인덱스 레지스터

IR0~15

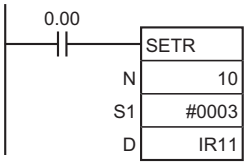
■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
S1	○	○	○	○	○	○	○	○	○	○		○		○				
D	–	–	–	–	–	–	–	–	–	–	–	–	○	–				

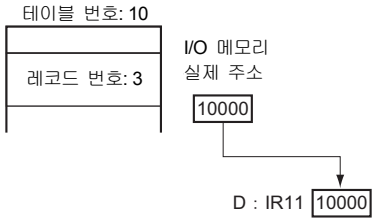
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 지정된 테이블 영역이 등록되지 않았을 때 ON · 지정된 레코드 번호가 등록된 레코드 수를 초과하는 경우 ON · 그 이외일 때 OFF

샘플 프로그램

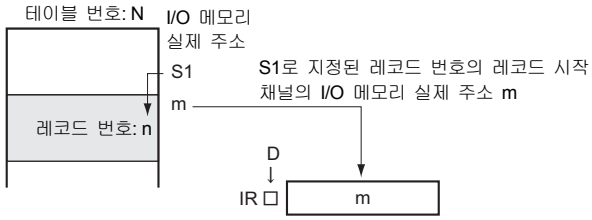


0.00이 ON이면 테이블 번호 10의 레코드 번호 3의 시작 주소의 I/O 메모리 실제 주소를 인덱스 레지스터 IR11에 저장합니다.



기능

N으로 지정한 번호를 가진 테이블 영역의 지정된 레코드(S1) 선두에 해당하는 I/O 메모리 실제 주소를 D로 지정한 인덱스 레지스터에 출력합니다.



주:

- 테이블 영역은 테이블 영역 선언(DIM) 명령으로 미리 등록해야 합니다.
- 레코드 번호는 선두의 레코드를 0으로 지정합니다.

GETR

명령 이름	니모닉	동작 옵션	평션 번호	기능
레코드 위치 읽기	GETR	@GETR	636	지정된 인덱스 레지스터의 값(I/O 메모리 실제 주소)이 포함된 레코드의 번호를 출력합니다.

심볼	GETR
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	테이블 번호	–	1
S1	인덱스 레지스터	UINT	2
D	레코드 위치 출력 CH 번호	WORD	1

N: 테이블 번호

0~15

S1: 인덱스 레지스터

IR0~15

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
S1													○					
D	○	○	○	○	○	○	○	○	○	○	–	○	–	○				

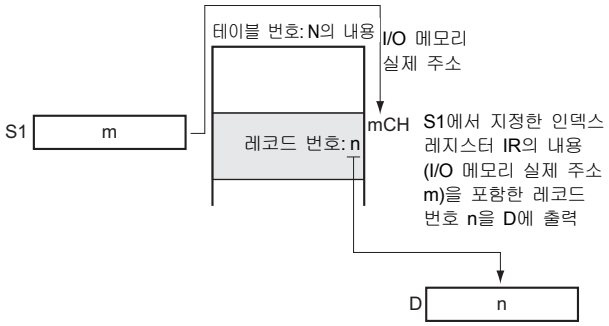
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 인덱스 레지스터가 나타내는 주소가 지정된 테이블 영역이 아닐 때 ON · 지정된 테이블 영역이 등록되지 않았을 때 ON · 그 이외일 때 OFF

기능

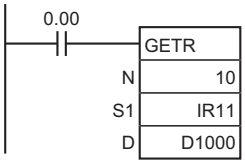
테이블 번호 N의 내용 중에서 S1로 지정한 인덱스 레지스터에 저장된 값(I/O 메모리 실제 주소)이 포함된 레코드 번호를 D에 출력합니다.

인덱스 레지스터에 저장된 I/O 메모리 실제 주소는 해당 레코드의 선두가 아니어도 됩니다.

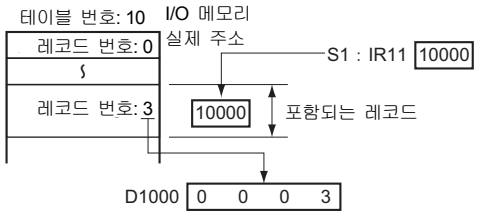


주: 테이블 영역은 테이블 영역 선언(DIM) 명령으로 미리 등록해야 합니다.

샘플 프로그램



0.00이 ON이면 인덱스 레지스터 IR11에 저장된 값을 I/O 메모리 실제 주소로 하는 주소를 포함하는 레코드 번호 3을 D1000에 저장합니다.



SRCH

명령 이름	니모닉	동작 옵션	평션 번호	기능
데이터 검색	SRCH	@SRCH	181	지정된 테이블 내의 데이터를 검색합니다.

심볼	SRCH	
	SRCH	
	W	W: 테이블 길이 지정 데이터
	S1	S1: 테이블 하위 CH 번호
	S2	S2: 검색 데이터

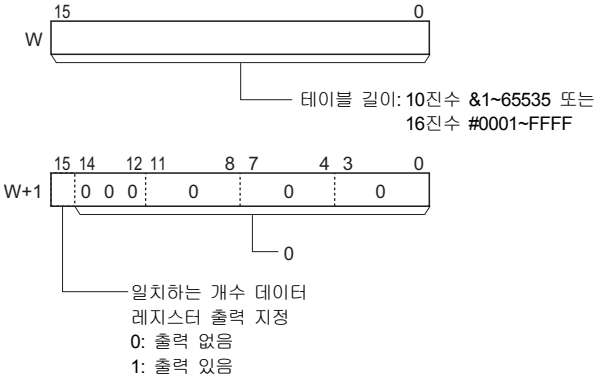
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

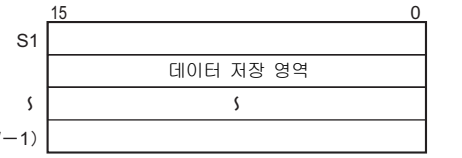
피연산자 설명

피연산자	내용	데이터 형식	크기
W	테이블 길이 지정 데이터	UDINT	2
S1	테이블 하위 CH 번호	UINT	가변
S2	검색 데이터	WORD	1

W: 테이블 길이 지정 데이터



S1: 테이블 하위 CH 번호



주: W~W+1 및 S1~S1+(W－1)은 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
W	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
S1											–							
S2											○	○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· W의 테이블 길이 데이터(비트 00~15)가 10진수 &1~65535 또는 16진수 #0001~FFFF 범위 안에 없을 때 ON · 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 검색 결과, S2의 데이터와 동일한 데이터가 테이블 내에 있을 때 ON · 그 이외일 때 OFF

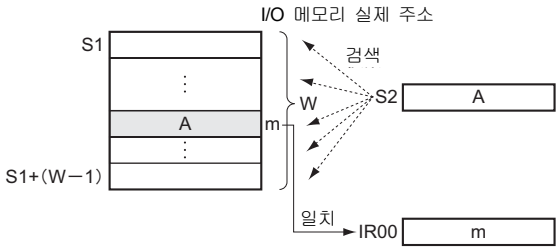
관련 특수 보조 릴레이

이름	주소	내용
백그라운드 실행 시 IR00 출력	A595~A596	인덱스 레지스터 출력을 지정한 명령을 백그라운드 처리로 실행한 경우, 결과를 이 영역에 출력하며 IR00에는 출력하지 않습니다. A595: 하위, A596: 상위
백그라운드 실행 시 DR00 출력	A597	데이터 레지스터 출력을 지정한 명령을 백그라운드 처리로 실행한 경우, 결과를 이 영역에 출력하며 DR00에는 출력하지 않습니다.
백그라운드 처리 시 = 플래그 출력	A598.01	SRCH 명령을 백그라운드 처리로 실행한 경우 데이터를 검색한 결과, 일치하는 데이터가 있을 때 이 플래그를 1(ON)로 설정합니다.
백그라운드 처리 중 ER/AER 플래그 출력	A395.10	백그라운드 처리 중에 명령 에러 또는 잘못된 영역 액세스 에러가 발생한 경우, 이 플래그가 1(ON)이 됩니다.

기능

S1로 지정된 테이블 하위 CH 번호부터 테이블 길이(W) 분량의 테이블 데이터에 대해 CH(채널) 단위로 지정된 데이터(S2)를 검색하여, 일치하는 데이터가 있는 경우 해당 데이터가 있는 CH(여러 개 있는 경우는 하위 CH)의 I/O 메모리 실제 주소를 인덱스 레지스터 IR0에 출력합니다. 동시에 = 플래그를 ON합니다.

일치하는 개수 데이터 레지스터 출력 지정(W+1의 비트 15)이 출력 있음(1)으로 지정된 경우, 일치하는 개수를 10진수 &0~65535 또는 16진수 #0000~FFFF로 데이터 레지스터 DR0에 출력합니다. 출력 없음(0)으로 지정된 경우, DR0은 변경되지 않습니다.



힌트

· 이 명령은 1레코드가 1CH인 테이블 데이터에 대해 검색을 수행하는 명령입니다. 1레코드가 복수 CH인 테이블 데이터에 대한 검색은 DIM, SETR, GETR 명령이나 FOR~NEXT, BREAK 명령 및 인덱스 레지스터(IR)에 의한 간접 참조를 조합하여 구현할 수 있습니다.

· 이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

사용상의 주의 사항

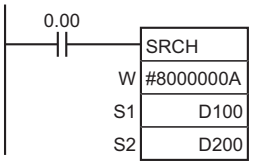
· 일치하는 데이터가 있는지 여부는 이 명령을 수행한 직후, = 플래그를 읽어서 판단하십시오.

· 일치하는 데이터가 없는 경우는 IR0과 DR0의 값이 변경되지 않습니다.

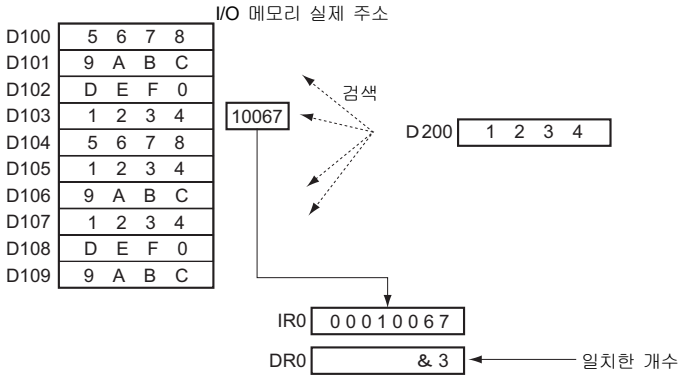
· PLC 시스템 설정에서 백그라운드 처리를 지정한 경우, 일치하는 데이터가 있는 CH의 실제 I/O 메모리 실제 주소는 IR0에 출력되지 않고 대신 특수 보조 릴레이 A595/A596에 출력됩니다.

· PLC 시스템 설정에서 백그라운드 처리를 지정하고 일치하는 개수의 데이터 레지스터 출력을 지정한 경우, 일치하는 개수는 DR0에 출력되지 않고 대신 특수 보조 릴레이 A597에 출력됩니다.

샘플 프로그램



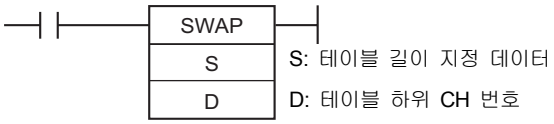
0.00이 ON이면 D100부터 10CH의 데이터에서 D200의 내용과 같은 내용을 검색하여 일치하는 것 중 가장 작은(하위의) I/O 메모리 실제 주소를 인덱스 레지스터 IR0에 저장합니다. 검색 결과, 일치한 개수를 데이터 레지스터 DR0에 저장합니다.



테이블 길이 지정 데이터를 10진수 &10 또는 16진수 #000A로 한 경우, 데이터 레지스터 DR0에 일치하는 개수를 출력하지 않습니다.

SWAP

명령 이름	니모닉	동작 옵션	평션 번호	기능
바이트 스와핑	SWAP	@SWAP	637	지정된 테이블 데이터의 지정된 CH 수의 상위 바이트와 하위 바이트를 교체합니다.

심볼	SWAP
	

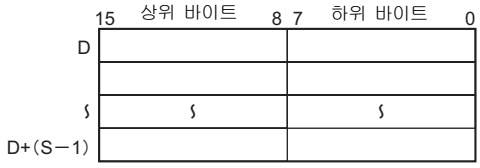
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	테이블 길이 지정 데이터	UINT	1
D	테이블 하위 CH 번호	UINT	가변

S: 테이블 길이 지정 데이터
10진수 &1~65535 또는 16진수 #0001~FFFF
D: 테이블 하위 CH 번호



주: D~D+(S－1)은 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
D											–	–						

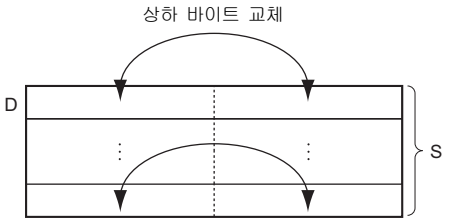
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S의 데이터가 0일 때 ON · 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF

SWAP

기능

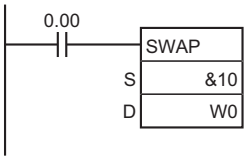
D로 지정된 테이블 데이터 선두 CH부터 지정된 테이블 길이(S) 분량의 테이블 데이터에 대해 각 데이터(16비트)의 상위 8비트와 하위 8비트를 교체하여 결과를 D에 출력합니다.



힌트

- 문자열(ASCII 코드)의 저장 순서를 반대로 할 때 등에 사용합니다.
- 이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

샘플 프로그램



0.00이 ON이면 W0CH~W9CH의 10CH에 대해 각 데이터의 상위 바이트와 하위 바이트를 교체합니다.

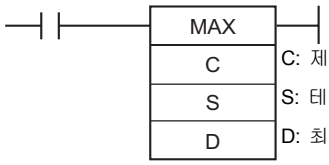
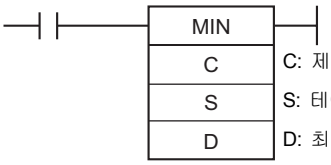
	15	8	7	0
W0	4	1	4	2
W1	4	3	4	4
W2	4	5	4	6
≀		≀		
W9	3	0	3	1

⇨

	15	8	7	0
W0	4	2	4	1
W1	4	4	4	3
W2	4	6	4	5
≀		≀		
W9	3	1	3	0

MAX/MIN

명령 이름	니모닉	동작 옵션	펑션 번호	기능
최대값 검색	MAX	@MAX	182	지정된 테이블 내의 최대값을 검색합니다.
최소값 검색	MIN	@MIN	183	지정된 테이블 내의 최소값을 검색합니다.

심볼	MAX	MIN
		

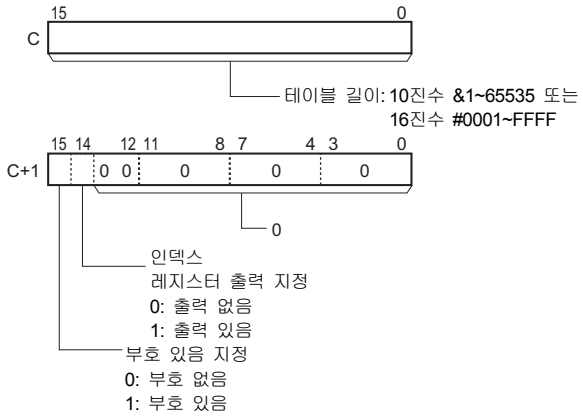
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

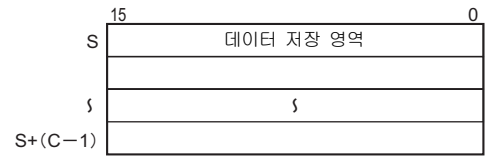
피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	UDINT	2
S	테이블 하위 CH 번호	UINT	가변
D	MAX: 최대값 출력 대상 CH 번호 MIN: 최소값 출력 대상 CH 번호	UINT	1

C: 제어 데이터



S: 테이블 하위 CH 번호



주: C~C+1 및 S~S+(C－1)은 각각 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
S											–							
D												○						

MAX/MIN

조건 플래그의 동작

이름	라벨	내용	
		MAX	MIN
에러 플래그	P_ER	· C의 테이블 길이 데이터(비트 00~15)가 10진수 &1~65535 또는 16진수 #0001~FFFF 범위 안에 없을 때 ON · 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF	
= 플래그	P_EQ	· 최대값이 0일 때 ON · 그 이외일 때 OFF	· 최소값이 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 최대값의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF	· 최소값의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

이름	주소	내용
백그라운드 실행 시 IR0 출력	A595~A596	인덱스 레지스터 출력을 지정한 명령을 백그라운드 처리로 실행한 경우, 결과를 이 영역에 출력하며 IR0에는 출력하지 않습니다. A595: 하위, A596: 상위
백그라운드 처리 중 ER/AER 플래그 출력	A395.10	백그라운드 처리 중에 명령 에러 또는 잘못된 영역 액세스 에러가 발생한 경우, 이 플래그가 1(ON)이 됩니다.

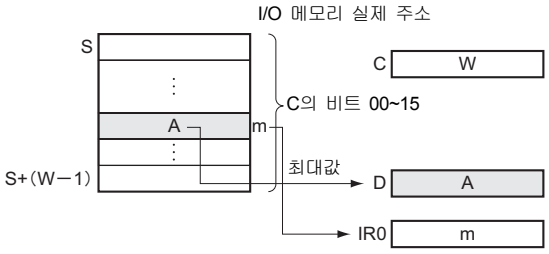
기능

■ MAX

S로 지정된 테이블 하위 CH 번호부터 C로 지정된 테이블 길이(워드 수) 분량을 테이블 데이터로 간주하고, 그 중에서 최대값을 검색하여 D에 출력합니다.

인덱스 레지스터 출력 지정(C+1)을 '출력 있음'로 지정한 경우, 최대값이 있는 CH(복수의 CH에 있는 경우는 하위 CH)의 I/O 메모리 실제 주소를 IR0에 출력합니다.

부호 있음 지정(C+1) 시에는 테이블의 데이터를 부호 있는 BIN 데이터(음수는 2의 보수)로 취급합니다.

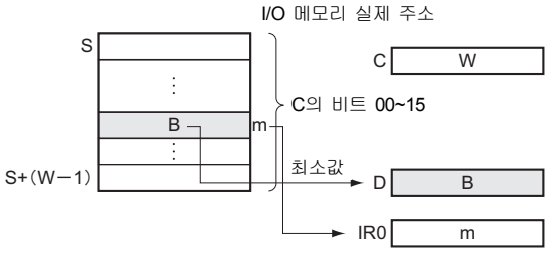


■ MIN

S로 지정된 테이블 하위 CH 번호부터 C로 지정된 테이블 길이(워드 수) 분량을 테이블 데이터로 간주하고, 그 중에서 최소값을 검색하여 D에 출력합니다.

인덱스 레지스터 출력 지정(C+1) 시, 최소값이 있는 CH(복수의 CH에 있는 경우는 하위 CH)의 I/O 메모리 실제 주소를 IR0에 출력합니다.

부호 있음 지정(C+1) 시에는 테이블의 데이터를 부호 있는 BIN 데이터(음수는 2의 보수)로 취급합니다.



힌트

- 부호 있음 지정(C+1의 비트 15)에 '부호 있음'(1)을 지정한 경우, 16진수 #8000~FFFF의 숫자를 음수로 취급합니다. 따라서 부호 지정 유무에 따라 검색 결과가 달라집니다.
- 이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

■ MAX

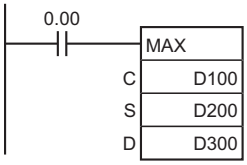
· PLC 시스템 설정에서 백그라운드 처리를 지정하고 인덱스 레지스터 출력을 지정한 경우, 최대값이 있는 CH의 실제 I/O 메모리 실제 주소는 IR0에 출력되지 않고 대신 특수 보조 릴레이 A595/A596에 출력됩니다.

■ MIN

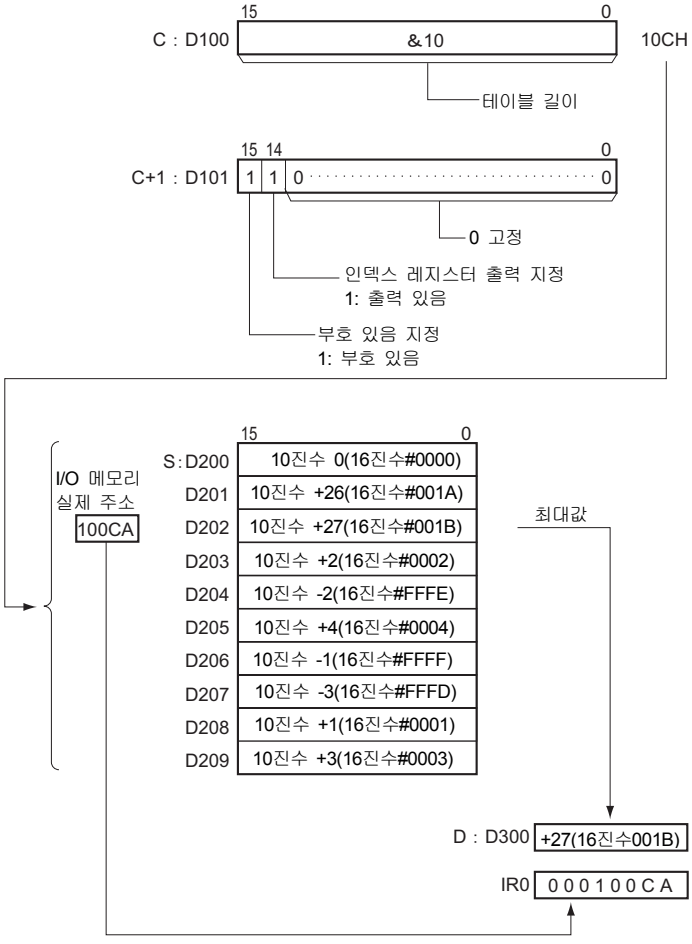
· PLC 시스템 설정에서 백그라운드 처리를 지정하고 인덱스 레지스터 출력을 지정한 경우, 최소값이 있는 CH의 실제 I/O 메모리 실제 주소는 IR0에 출력되지 않고 대신 특수 보조 릴레이 A595/A596에 출력됩니다.

샘플 프로그램

■ MAX

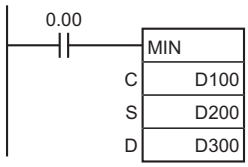


0.00이 ON이면 D200부터 10CH(D100으로 지정)의 테이블 중에서 최대값을 검색하여 그 값을 D300에 저장합니다. 동시에 최대값이 있던 주소 D202의 I/O 메모리 실제 주소를 IR0에 저장합니다.

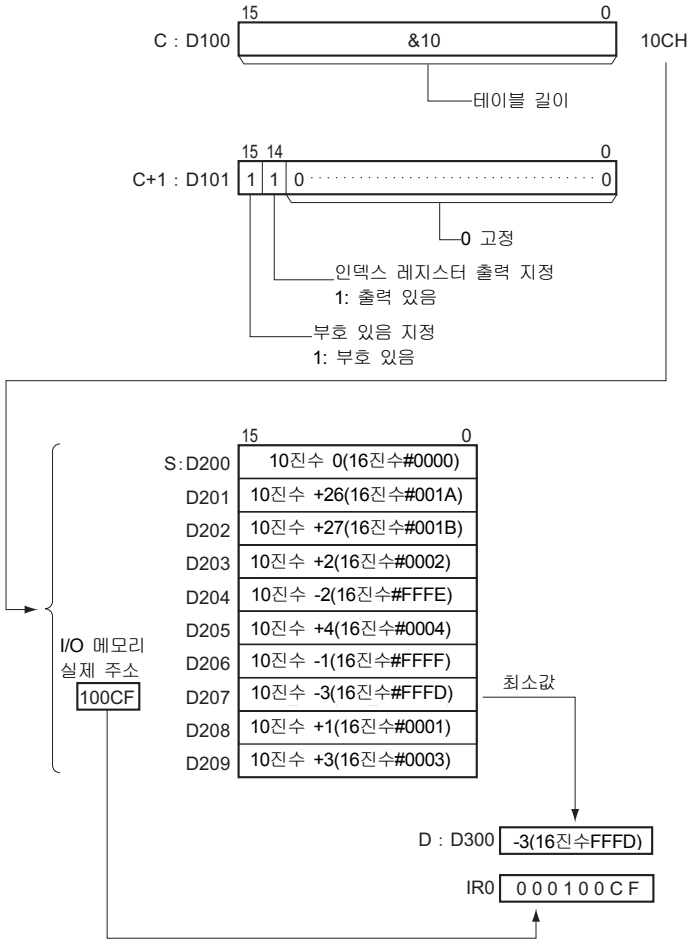


MAX/MIN

■ MIN

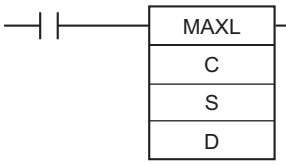


0.00이 ON이면 D200부터 10CH(D100으로 지정)의 테이블 중에서 최소값을 검색하여 그 값을 D300에 저장합니다. 동시에 최소값이 있던 주소 D207의 I/O 메모리 실제 주소를 IR0에 저장합니다.



MAXL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
2워드 데이터 최대 값 검색	MAXL	@MAXL	174	지정된 데이터 수만큼 2CH 데이터의 테이블 데이터로 간주하고, 그 중에서 최대값을 검색하여 출력합니다.

심볼	MAXL
	

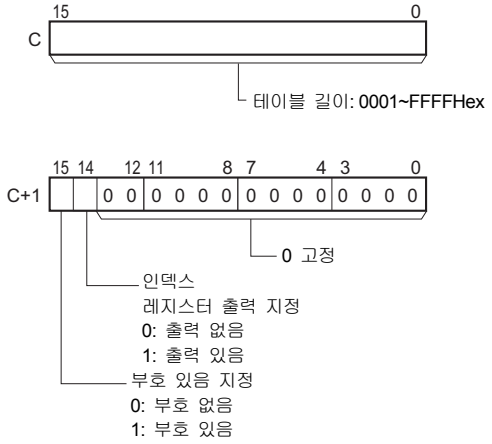
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

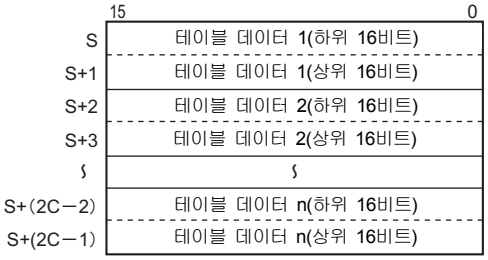
피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	UDINT	2
S	테이블 데이터	UDINT	가변
D	최대값 출력 대상 하위 CH 번호	UDINT	2

C: 제어 데이터



S: 테이블 하위 CH 번호



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
S											–							
D																		

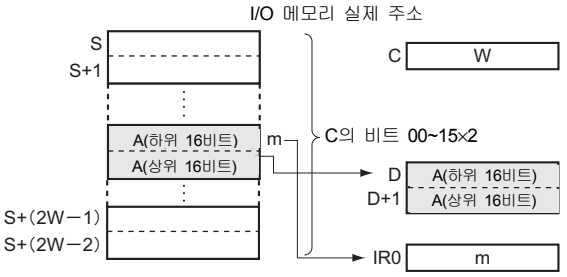
MAXL

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C의 테이블 길이 데이터(비트 00~15)가 0001~FFFFHex의 범위 안에 없을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 최대값이 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 최대값의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

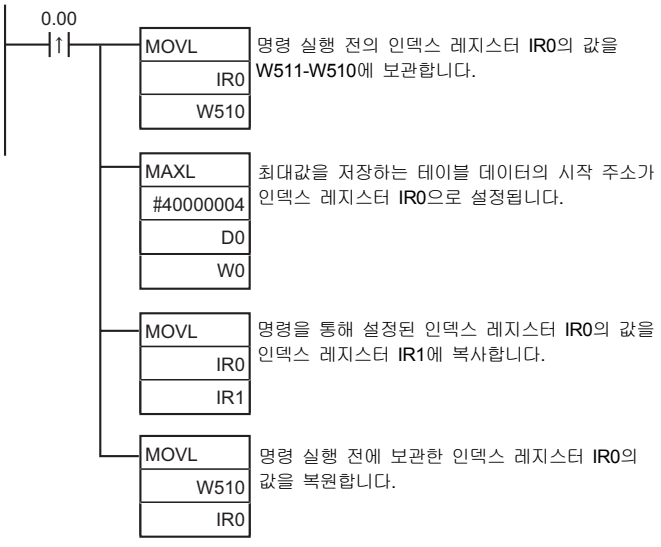
기능

S로 지정된 테이블 하위 CH 번호부터 C로 지정된 데이터 수만큼 2CH 데이터의 테이블 데이터로 간주하고, 그 중에서 최대값을 검색하여 D+1, D에 출력합니다.
인덱스 레지스터 출력 지정(C+1)을 '출력 있음'으로 지정한 경우, 최대값이 있는 CH(복수의 CH에 있는 경우는 하위 CH)의 I/O 메모리 실제 주소를 IR0에 출력합니다.
부호 있음 지정(C+1) 시에는 테이블의 데이터를 부호 있는 BIN 데이터(음수는 2의 보수)로 취급합니다.



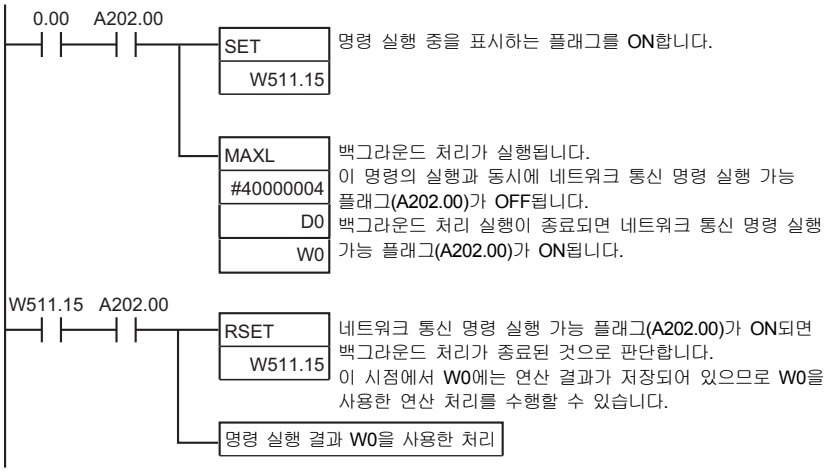
■ 인덱스 레지스터의 보관/복구(MAXL, MAXF, MAXD, MINL, MINF, MIND 공통)

이 명령은 최대값 또는 최소값을 저장하는 테이블 데이터의 시작 주소를 인덱스 레지스터 IR0에 저장할 수 있습니다. 프로그램 내에서 인덱스 레지스터 IR0을 다른 용도로 사용하는 경우는 이 명령 전후에 인덱스 레지스터 IR0의 값을 보관/복구하십시오.



■ 데이터 백그라운드 처리(MAXL, MAXF, MAXD, MINL, MINF, MIND 공통)

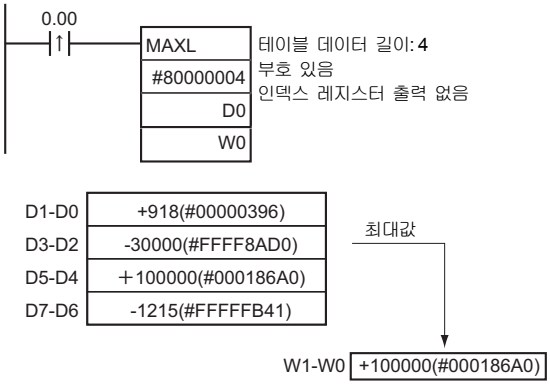
이 명령의 실행 시간은 테이블 데이터의 크기에 비례하여 길어집니다. 테이블 데이터의 크기가 길어지면 CPU 유니트의 사이클 시간 편차를 억제하기 위해 '백그라운드 처리 실행'이 유효합니다.



샘플 프로그램

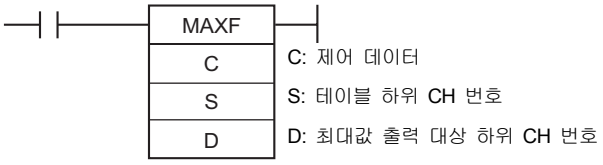
접점 0.00을 ON하면 D0을 선두로 한 테이블 데이터 4개분을 부호 있는 2CH 데이터로 간주하여 최대값을 검색합니다.

그 결과는 W0-W1에 설정됩니다.



MAXF

명령 이름	니모닉	동작 옵션	펑션 번호	기능
단정도 부동 소수점 데이터 최대값 검색	MAXF	@MAXF	176	지정된 데이터 수만큼 2CH 데이터의 단정도 부동 소수점 데이터로 간주하고, 그 중에서 최대값을 검색하여 출력합니다.

심볼	MAXF
	

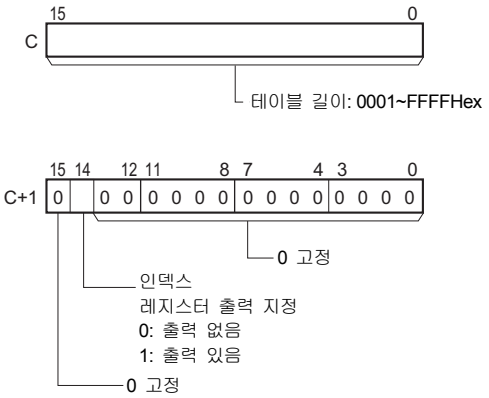
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

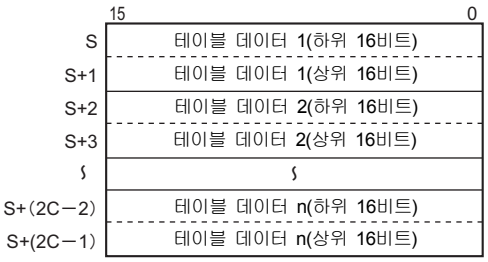
피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	DWORD	2
S	테이블 데이터	REAL	가변
D	최대값 출력 대상 하위 CH 번호	REAL	2

C: 제어 데이터



S: 테이블 하위 CH 번호



■ 영역 종류

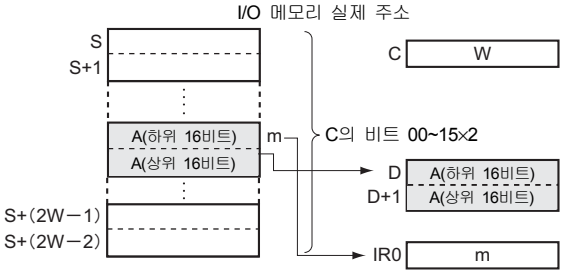
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
S											–							
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C의 테이블 길이 데이터(비트 00~15)가 0001~FFFFHex의 범위 안에 없을 때 ON · 테이블 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 최대값이 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 최대값이 음수일 때 ON · 그 이외일 때 OFF

기능

S로 지정된 테이블 하위 CH 번호부터 C로 지정된 데이터 수만큼 단정도 부동 소수점 데이터(32비트: IEEE754)의 테이블 데이터로 간주하고, 그 중에서 최대값을 검색하여 D+1, D에 출력합니다.
인덱스 레지스터 출력 지정(C+1)을 '출력 있음'으로 지정한 경우, 최대값이 있는 CH(복수의 CH에 있는 경우는 하위 CH)의 I/O 메모리 실제 주소를 IR0에 출력합니다.

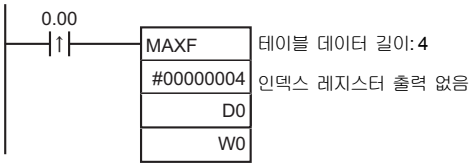


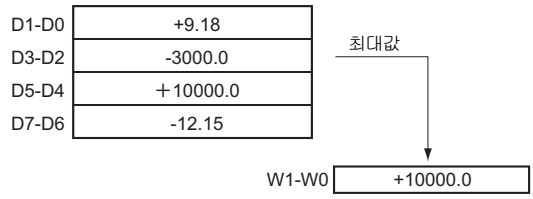
힌트

· '인덱스 레지스터의 보관/복구' 및 '백그라운드 처리'에 대해서는 MAXL을 참조하십시오.

샘플 프로그램

접점 0.00을 ON하면 D0을 선두로 한 테이블 데이터 4개분을 단정도 부동 소수점 데이터로 간주하여 최대값을 검색합니다.
그 결과는 W0-W1에 설정됩니다.





MAXD

명령 이름	니모닉	동작 옵션	평션 번호	기능
배정도 부동 소수점 데이터 최대값 검색	MAXD	@MAXD	178	지정된 데이터 수만큼 2CH 데이터의 배정도 부동 소수점 데이터로 간주하고, 그 중에서 최대값을 검색하여 출력합니다.

심볼	MAXD
	

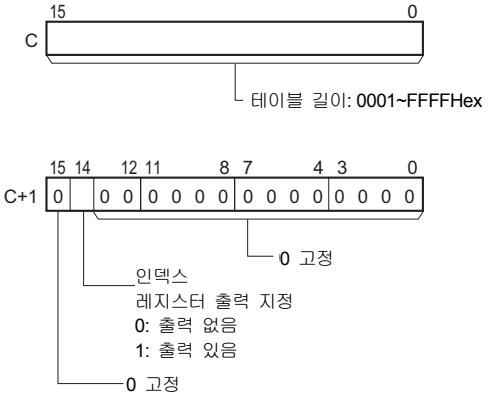
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

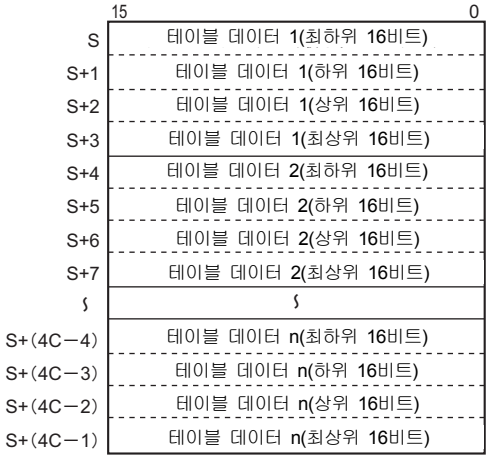
피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	DWORD	2
S	테이블 데이터	LREAL	가변
D	최대값 출력 대상 하위 CH 번호	LREAL	4

C: 제어 데이터



S: 테이블 하위 CH 번호



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
S											–							
D																		

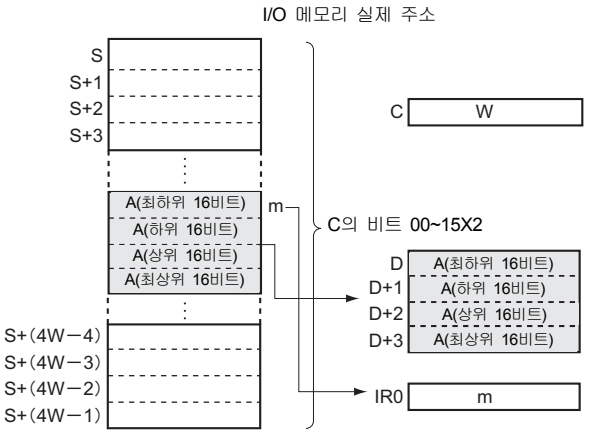
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C의 테이블 길이 데이터(비트 00~15)가 0001~FFFFHex의 범위 안에 없을 때 ON · 테이블 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 최대값이 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 최대값이 음수일 때 ON · 그 이외일 때 OFF

기능

S로 지정된 테이블 하위 CH 번호부터 C로 지정된 데이터 수만큼 배정도 부동 소수점 데이터(64비트: IEEE754)의 테이블 데이터로 간주하고, 그 중에서 최대값을 검색하여 D~D+3에 출력합니다.

인덱스 레지스터 출력 지정(C+1)을 '출력 있음'으로 지정한 경우, 최대값이 있는 CH(복수의 CH에 있는 경우는 하위 CH)의 I/O 메모리 실제 주소를 IR0에 출력합니다.



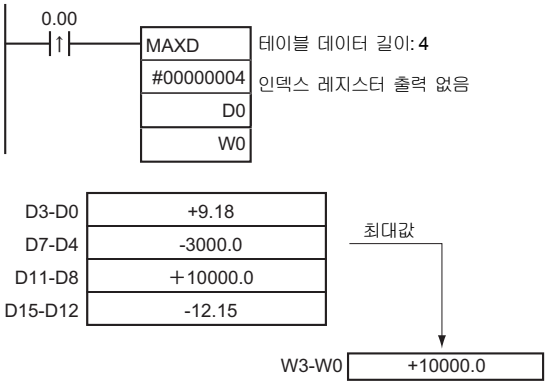
힌트

· '인덱스 레지스터의 보관/복구' 및 '백그라운드 처리'에 대해서는 MAXL을 참조하십시오.

샘플 프로그램

접점 0.00을 ON하면 D0을 선두로 한 테이블 데이터 4개분을 배정도 부동 소수점 데이터로 간주하여 최대값을 검색합니다.

그 결과는 W0-W3에 설정됩니다.



MINL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
2워드 데이터 최소 값 검색	MINL	@MINL	175	지정된 데이터 수만큼 2CH 데이터의 테이블 데이터로 간주하고, 그 중에서 최소값을 검색하여 출력합니다.

심볼	MINL
	

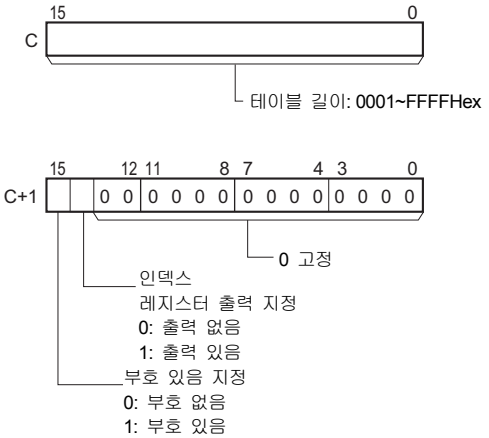
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

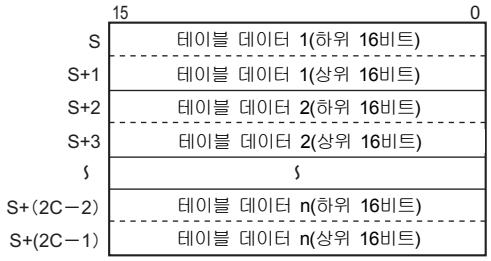
피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	UDINT	2
S	테이블 데이터	UDINT	가변
D	최소값 출력 대상 하위 CH 번호	UDINT	2

C: 제어 데이터



S: 테이블 하위 CH 번호



■ 영역 종류

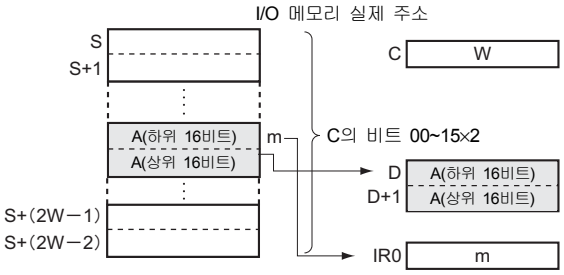
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
S											–							
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C의 테이블 길이 데이터(비트 00~15)가 0001~FFFFHex의 범위 안에 없을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 최소값이 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 최소값의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

S로 지정된 테이블 하위 CH 번호부터 C로 지정된 데이터 수만큼 2CH 데이터의 테이블 데이터로 간주하고, 그 중에서 최소값을 검색하여 D+1, D에 출력합니다.
인덱스 레지스터 출력 지정(C+1)을 '출력 있음'으로 지정한 경우, 최소값이 있는 CH(복수의 CH에 있는 경우는 하위 CH)의 I/O 메모리 실제 주소를 IR0에 출력합니다.
부호 있음 지정(C+1) 시에는 테이블의 데이터를 부호 있는 BIN 데이터(음수는 2의 보수)로 취급합니다.

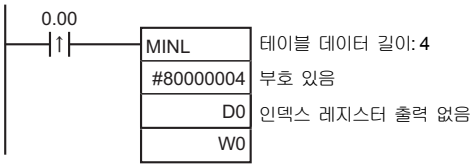


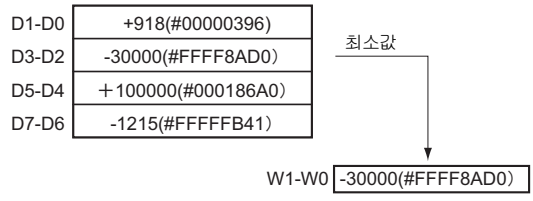
힌트

· '인덱스 레지스터의 보관/복구' 및 '백그라운드 처리'에 대해서는 MAXL을 참조하십시오.

샘플 프로그램

접점 0.00을 ON하면 D0을 선두로 한 테이블 데이터 4개분을 부호 있는 2CH 데이터로 간주하여 최소값을 검색합니다.
그 결과는 W0-W1에 설정됩니다.





MINF

명령 이름	니모닉	동작 옵션	평션 번호	기능
단정도 부동 소수점 데이터 최소값 검색	MINF	@MINF	177	지정된 데이터 수만큼 2CH 데이터의 단정도 부동소수점 데이터로 간주하고, 그 중에서 최소값을 검색하여 출력합니다.

심볼	MINF
	

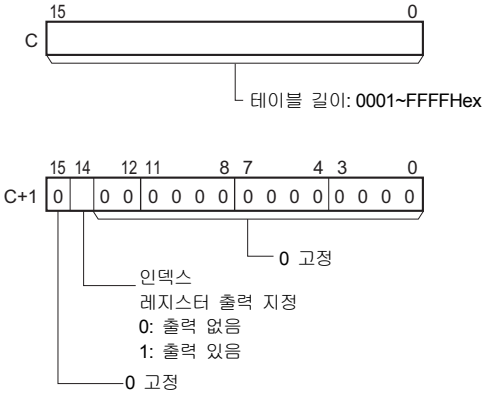
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

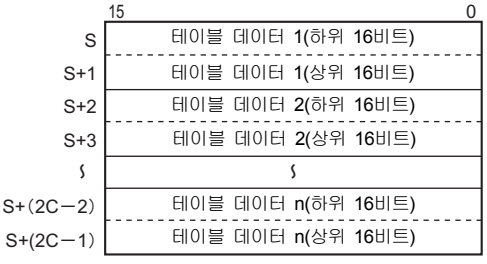
피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	DWORD	2
S	테이블 데이터	REAL	가변
D	최소값 출력 대상 하위 CH 번호	REAL	2

C: 제어 데이터



S: 테이블 하위 CH 번호



■ 영역 종류

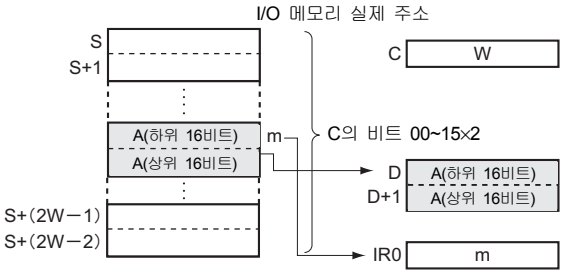
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	글록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
S											–							
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C의 테이블 길이 데이터(비트 00~15)가 0001~FFFFHex의 범위 안에 없을 때 ON · 테이블 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 최소값이 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 최소값이 음수일 때 ON · 그 이외일 때 OFF

기능

S로 지정된 테이블 하위 CH 번호부터 C로 지정된 데이터 수만큼 단정도 부동 소수점 데이터(32비트: IEEE754)의 테이블 데이터로 간주하고, 그 중에서 최소값을 검색하여 D+1, D에 출력합니다.
인덱스 레지스터 출력 지정(C+1)을 '출력 있음'으로 지정한 경우, 최소값이 있는 CH(복수의 CH에 있는 경우는 하위 CH)의 I/O 메모리 실제 주소를 IR0에 출력합니다.

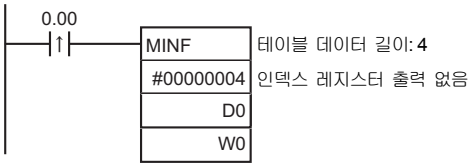


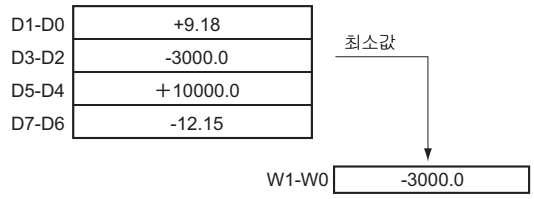
힌트

· '인덱스 레지스터의 보관/복구' 및 '백그라운드 처리'에 대해서는 MAXL을 참조하십시오.

샘플 프로그램

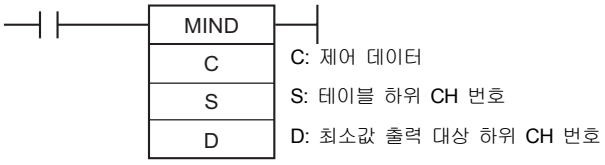
접점 0.00을 ON하면 D0을 선두로 한 테이블 데이터 4개분을 단정도 부동 소수점 데이터로 간주하여 최소값을 검색합니다.
그 결과는 W0-W1에 설정됩니다.





MIND

명령 이름	니모닉	동작 옵션	펑션 번호	기능
배정도 부동 소수점 데이터 최소값 검색	MIND	@MIND	179	지정된 데이터 수만큼 2CH 데이터의 배정도 부동 소수점 데이터로 간주하고, 그 중에서 최소값을 검색하여 출력합니다.

심볼	MIND
	

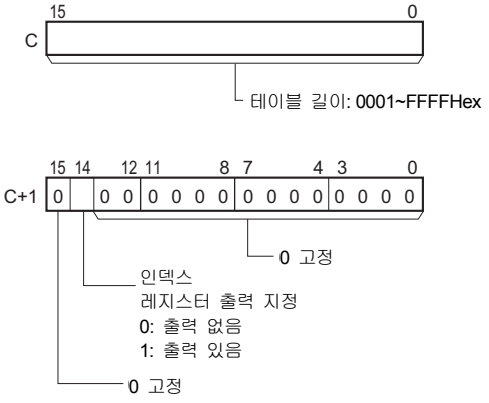
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

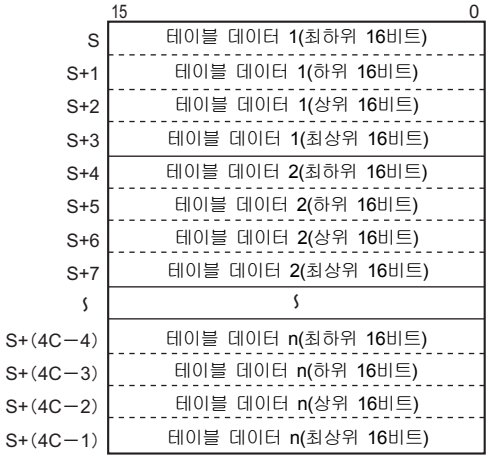
피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	DWORD	2
S	테이블 데이터	LREAL	가변
D	최소값 출력 대상 하위 CH 번호	LREAL	4

C: 제어 데이터



S: 테이블 하위 CH 번호



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
S											–							
D																		

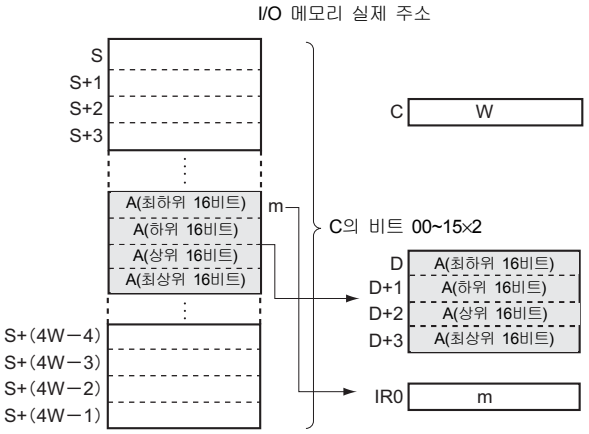
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C의 테이블 길이 데이터(비트 00~15)가 0001~FFFFHex의 범위 안에 없을 때 ON · 테이블 데이터가 비수(NaN)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 최소값이 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 최소값이 음수일 때 ON · 그 이외일 때 OFF

기능

S로 지정된 테이블 하위 CH 번호부터 C로 지정된 데이터 수만큼 배정도 부동 소수점 데이터(64비트: IEEE754)의 테이블 데이터로 간주하고, 그 중에서 최소값을 검색하여 D~D+4에 출력합니다.

인덱스 레지스터 출력 지정(C+1)을 '출력 있음'으로 지정한 경우, 최소값이 있는 CH(복수의 CH에 있는 경우는 하위 CH)의 I/O 메모리 실제 주소를 IR0에 출력합니다.



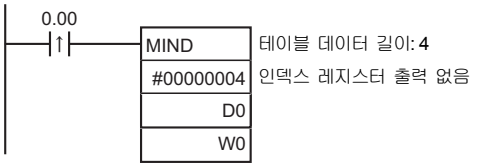
힌트

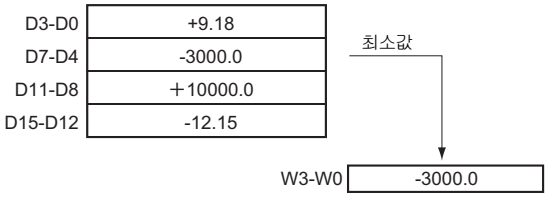
· '인덱스 레지스터의 보관/복구' 및 '백그라운드 처리'에 대해서는 MAXL을 참조하십시오.

샘플 프로그램

접점 0.00을 ON하면 D0을 선두로 한 테이블 데이터 4개분을 배정도 부동 소수점 데이터로 간주하여 최소값을 검색합니다.

그 결과는 W0-W3에 설정됩니다.

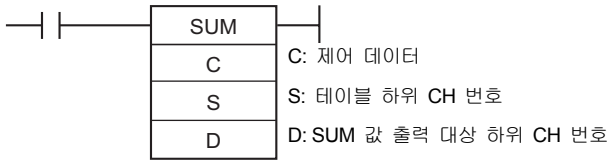




SUM

명령 이름	니모닉	동작 옵션	평션 번호	기능
SUM 값 산출	SUM	@SUM	184	지정된 테이블 내의 SUM 값을 계산하고 2CH로 출력합니다.

심볼	SUM



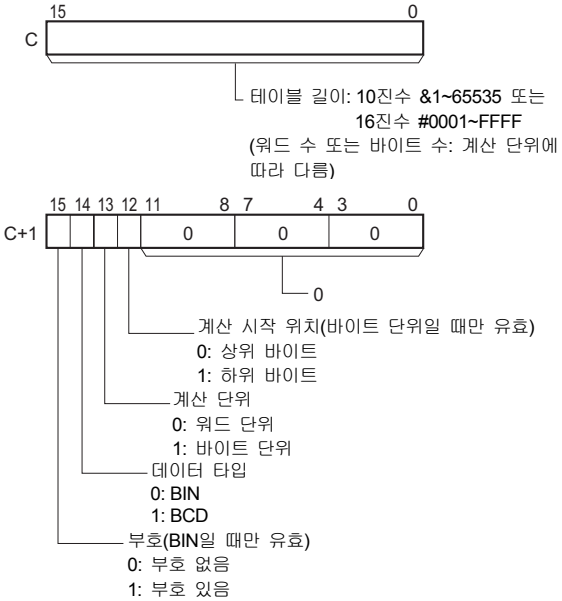
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

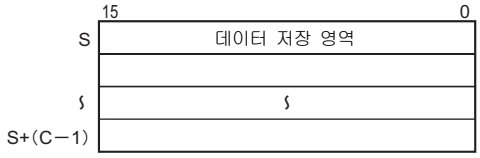
피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	UDINT	2
S	테이블 하위 CH 번호	UINT	가변
D	SUM 값 출력 대상 하위 CH 번호	UDINT	2

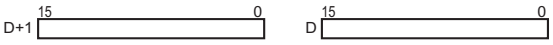
C: 제어 데이터



S: 테이블 하위 CH 번호



D: SUM 값 출력 대상 하위 CH 번호



D에 하위 4자리, D+1에 상위 4자리가 저장됩니다.

주: C~C+1 및 S~S+(C－1)은 각각 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
S											–							
D											–							

조건 플래그의 동작

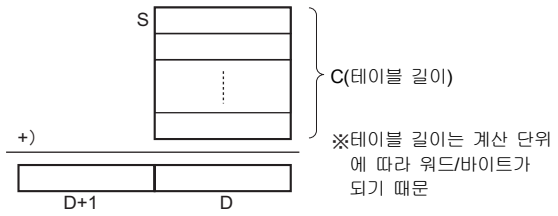
이름	라벨	내용
에러 플래그	P_ER	· C의 테이블 길이 데이터(비트 00~15)가 10진수 &1~65535 또는 16진수 #0001~FFFF 범위 안에 없을 때 ON · 테이블 내에 BIN 데이터가 있을 때 ON(데이터 타입 BCD 지정 시) · 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과, SUM 값이 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 연산 결과, SUM 값의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

S로 지정된 테이블 하위 CH 번호부터 C로 지정된 테이블 길이 분량을 테이블 데이터로 간주하고, C+1로 지정된 계산 단위(워드 또는 바이트) 및 데이터 타입(BIN 또는 BCD)에 따라 SUM 값을 연산하여 결과를 D+1, D에 출력합니다.

계산 단위로 바이트를 지정한 경우, C+1의 계산 시작 위치 지정(상위 바이트 또는 하위 바이트)을 유효로 합니다.

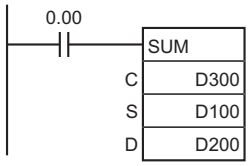
데이터 타입으로 BIN을 지정한 경우에는 C+1의 부호 지정(부호 있음 또는 부호 없음)을 유효로 합니다.



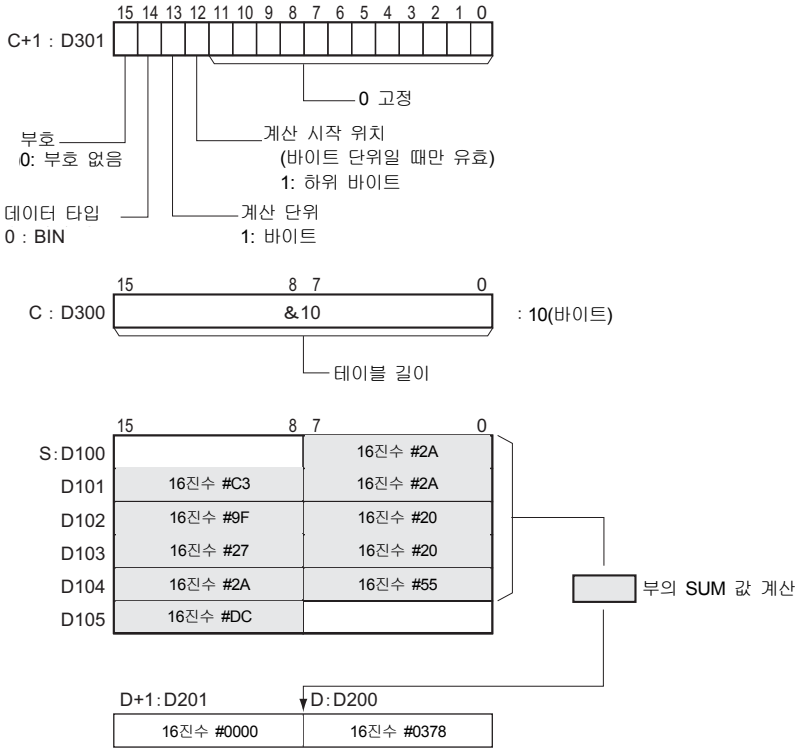
힌트

· 이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

샘플 프로그램

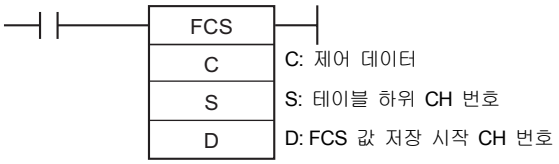


0.00이 ON이면 D100의 하위 바이트부터, D300이 지정한 바이트 수의 데이터에 대해 부호 없는 BIN 데이터로 SUM 값을 계산하여 D201, D200에 저장합니다.



FCS

명령 이름	니모닉	동작 옵션	평션 번호	기능
FCS 값 산출	FCS	@FCS	180	지정된 테이블 내의 FCS 값을 계산하고 ASCII 코드 데이터로 변환하여 출력합니다.

심볼	FCS
	

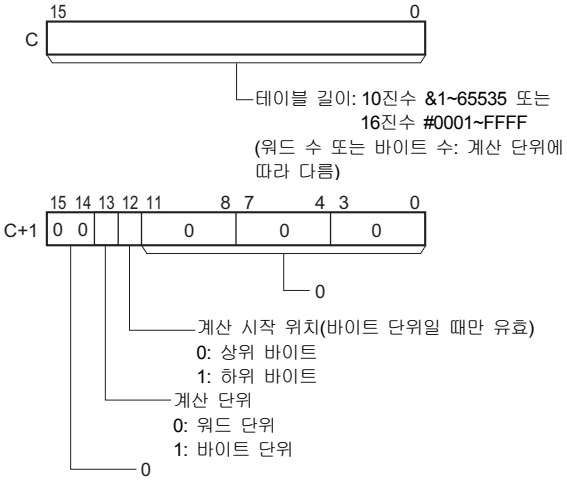
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

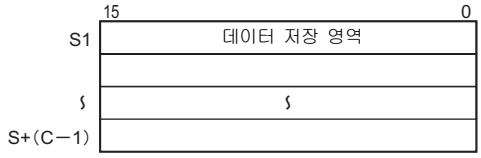
피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	UDINT	2
S	테이블 하위 CH 번호	UINT	가변
D	FCS 값 저장 시작 CH 번호	UINT	가변

C: 제어 데이터

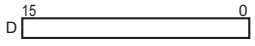


S: 테이블 하위 CH 번호

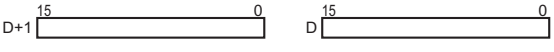


D: FCS 값 저장 시작 CH 번호

계산 단위에 바이트 단위 지정 시



계산 단위에 워드 단위 지정 시



D에 하위 4자리, D+1에 상위 4자리가 저장됩니다.

주: C~C+1 및 S~S+(C의 데이터－1)은 각각 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
S											–							
D																		

FCS

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C의 테이블 길이 데이터(비트 00~15)가 10진수 &1~65535 또는 16진수 #0001~FFFF 범위 안에 없을 때 ON · 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF

기능

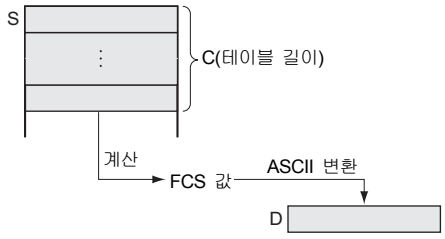
S로 지정된 테이블 하위 CH 번호부터 C로 지정된 테이블 길이 분량을 테이블 데이터로 간주하고, C+1로 지정된 계산 단위(워드 단위 또는 바이트 단위)로 FCS 값을 연산하여 ASCII 코드 데이터로 변환한 후

· 계산 단위로 바이트 단위를 지정했을 때: D에 출력합니다.

· 계산 단위로 워드 단위를 지정했을 때: D+1, D에 출력합니다.

계산 단위로 바이트 단위를 지정한 경우, C+1에서 계산 시작 위치(상위 바이트 또는 하위 바이트)를 지정합니다.

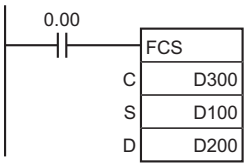
테이블 길이(C)의 단위(워드 또는 바이트)는 계산 단위(C+1)에 따릅니다.



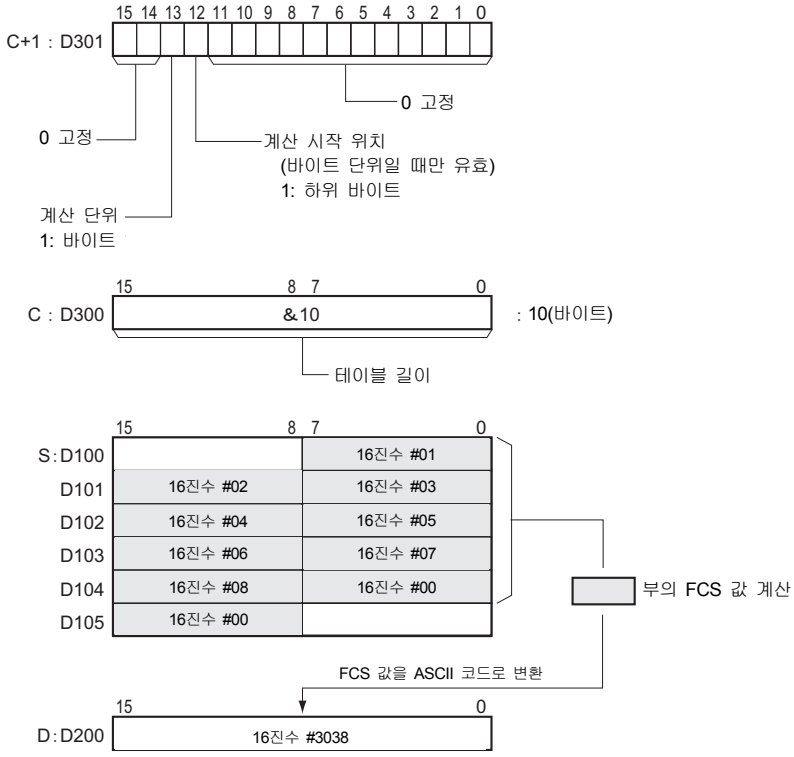
힌트

· 이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

샘플 프로그램



0.00이 ON이면 D100의 하위 바이트부터, D300이 지정한 바이트 수의 데이터에 대해 BIN 데이터로 FCS 값을 계산하여 D200에 저장합니다.



추적 명령

니모닉	명령 이름	펑션 번호	페이지
RSRCH<,RSRCH<=, RSRCH=,RSRCH> RSRCH>=, RSRCH>	부호 없는 1W 레코드 검색	360~364	3-495
RSRCH2<,RSRCH2<=, RSRCH2=,RSRCH2>=, RSRCH2>,	부호 없는 2W 레코드 검색	370~374	3-501
RSRCH4<,RSRCH4<=, RSRCH4=,RSRCH4>, RSRCH4>=	부호 없는 4W 레코드 검색	380~384	3-504
RSORT	부호 없는 1W 레코드 정렬	203	3-507
RSORT2	부호 없는 2W 레코드 정렬	204	3-511
RSORT4	부호 없는 4W 레코드 정렬	205	3-514

추적 명령

생산 라인에 투입되는 워크의 생산 정보를 일반적으로 '추적 데이터'라고 합니다. 추적 데이터는 '품종', '형식', '수량', '제조일', '제조 등급', '제조 공장' 등과 같은 여러 정보로 구성됩니다. (하나의 워크에 대한 정보의 집합체를 레코드라고 합니다.)
추적 명령을 사용하면 추적 데이터에서 필요한 레코드를 간단하게 검색하거나 레코드를 지정된 추적 데이터 정보로 정렬할 수 있습니다.

추적 명령의 샘플 프로그램을 아래에 기재되어 있습니다. 샘플 프로그램에서 사용하는 추적 명령의 동작 사양에 대해서는 각 명령에 대한 설명을 참조하십시오.

1. 검색 조건과 일치하는 추적 데이터를 추출하는 경우
2. 추적 데이터를 정렬한 후 검색하는 경우
3. 레코드 수가 많은 추적 데이터를 정렬하는 경우

■ 선반 번호나 배송지 코드에 대해 검색 및 정렬하는 경우

추적 데이터



다음과 같은 경우 무효 추적 데이터라고 정의합니다.

- 선반 번호가 #0000일 때
- 배송지 코드가 #FFFFFFFF일 때

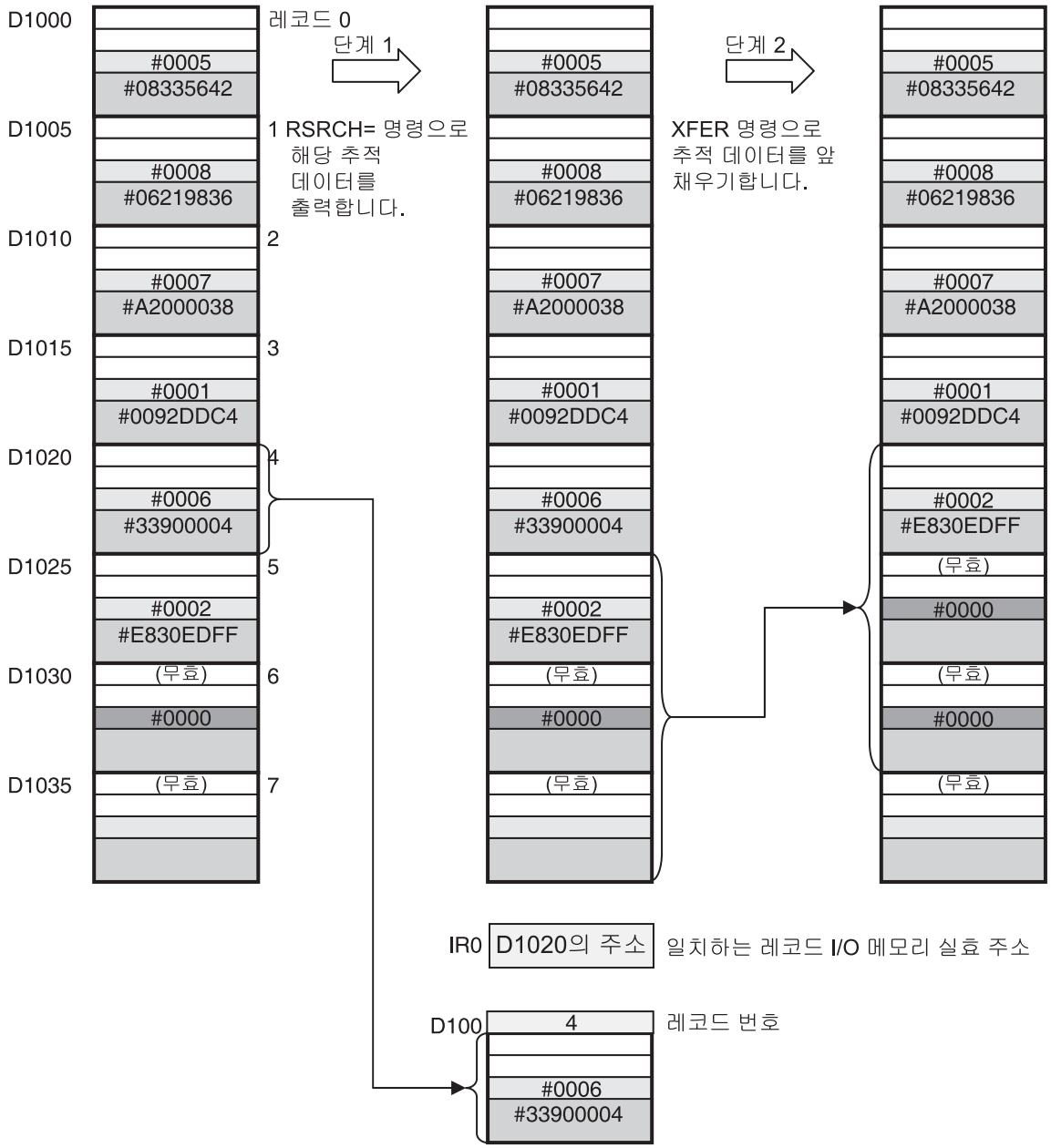




■ 검색 조건과 일치하는 추적 데이터를 추출하는 경우

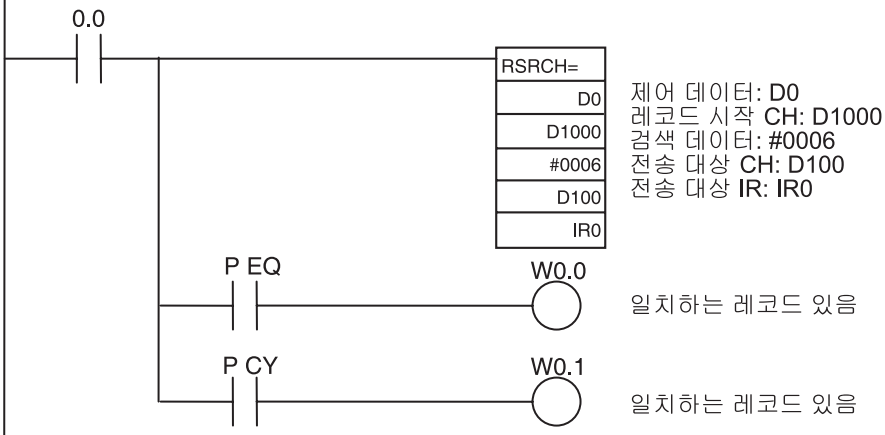
테이블에서 선반 번호가 #0006인 추적 데이터를 검색하여 추출을 수행합니다.

레코드 시작 CH	D1000
레코드 길이	5
레코드 수	8
검색 결과 전송 대상 CH	D100
검색 대상	선반 번호
검색 종료 설정	유효 종료 데이터 설정(선반 번호=#0000)
검색 방법	선형 탐색(선두 레코드부터 순서대로 검색)
전송 대상 인덱스 레지스터 번호	IR0

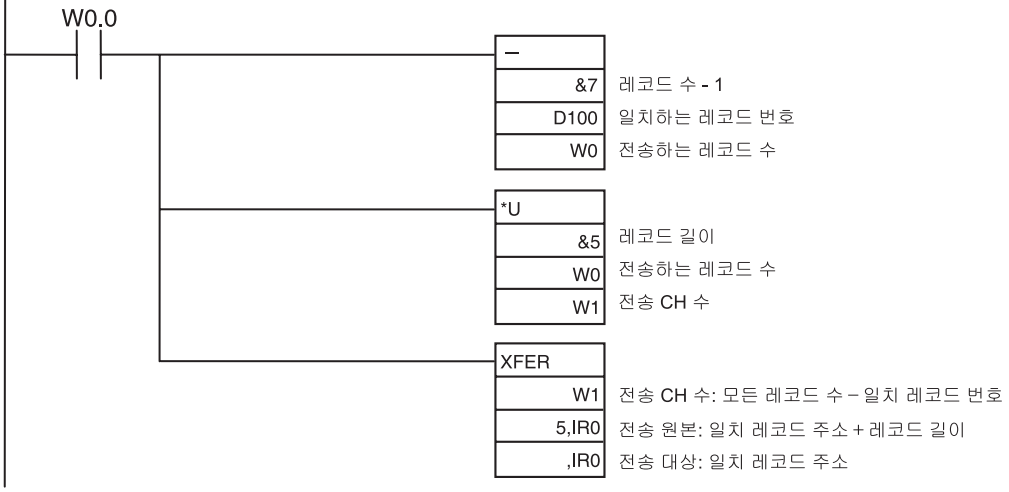


C: D0	0 0 0 8	레코드 수 = 8CH
C+1: D1	0 0 0 5	레코드 수 = 5CH
C+2: D2	0 0 0 2	검색 대상 데이터 오프셋 = 2CH
C+3: D3	8 0 0 1	인덱스 레지스터 출력 지정: 출력 있음 검색 방식 = 선형 검색/검색 종료 설정 = 유효: 종료 데이터 지정
C+4: D4	0 0 0 0	종료 데이터 = #0000
C+5: D5	0 0 0 0	#0000 고정

1단계: RSRCH= 명령으로 해당 추적 데이터를 출력합니다.



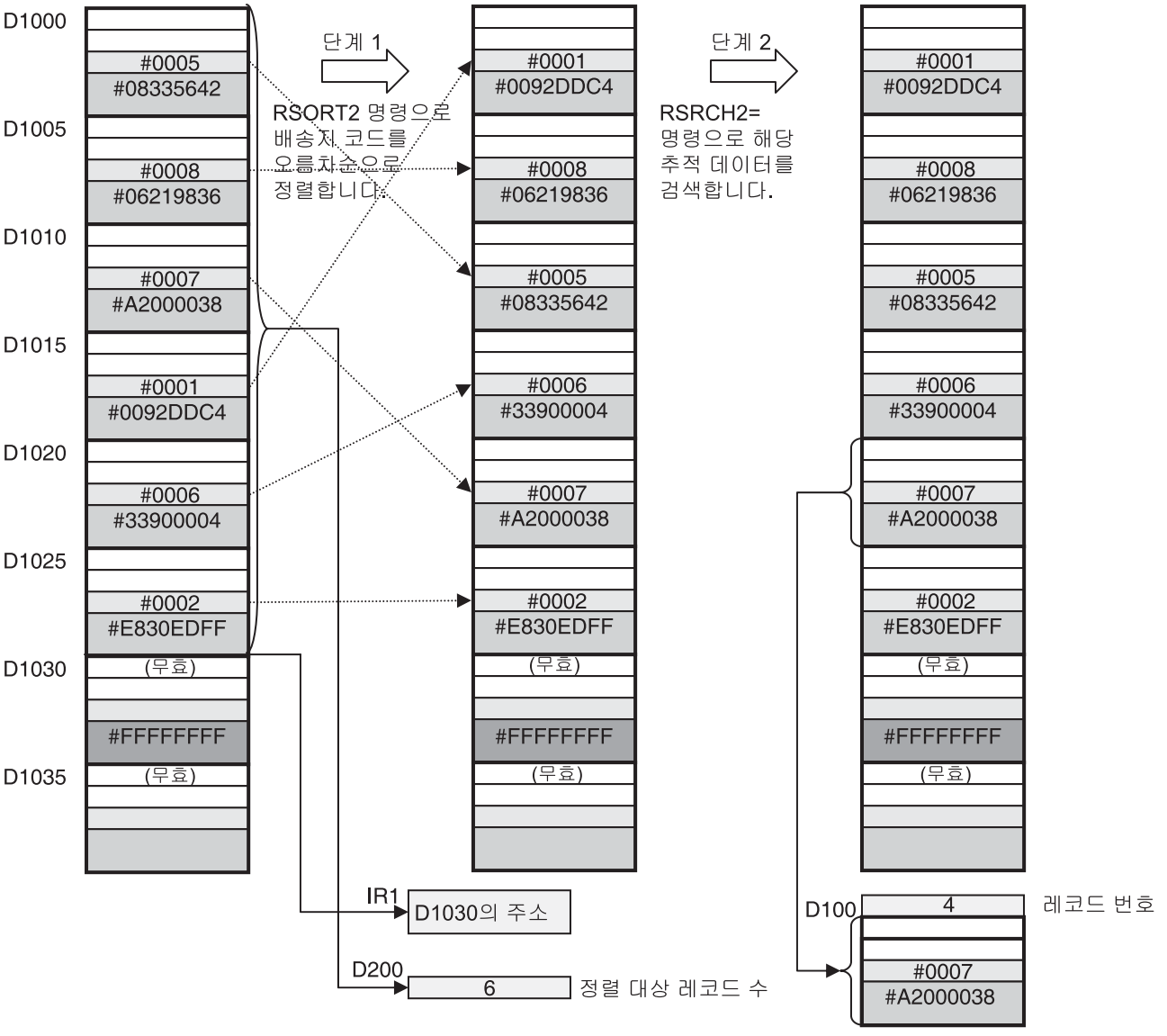
2단계: XFER 명령으로 추적 데이터를 앞 채우기합니다.



■ 추적 데이터를 정렬한 후 검색하는 경우

테이블을 배송지 코드로 정렬한 후 배송지가 #A2000038인 추적 데이터를 검색합니다.

레코드 시작 CH	D1000
레코드 길이	5
레코드 수	8
정렬 결과 출력 CH	D200
정렬 대상	배송지 코드
정렬 옵션	오름차순, 분할 정렬 설정 없음(1사이클에 완료)
정렬 종료 설정	유효 종료 데이터 설정(배송지 코드=#FFFFFFFF)
정렬 결과 인덱스 레지스터 출력	IR1
검색 결과 전송 대상 CH	D100
검색 대상	배송지 코드
검색 종료 설정	유효 종료 I/O 메모리 실제 주소 설정(정렬의 출력 결과에 따라 다름)
검색 방법	이분 탐색
검색 결과 인덱스 레지스터 출력	없음



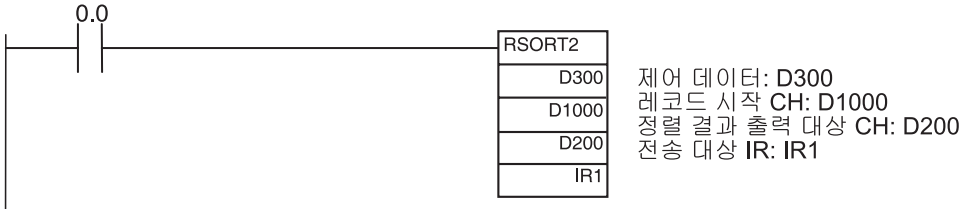
RSORT2

C: D300	0 0 0 8	레코드 수 = 8CH
C+1: D301	0 0 0 5	레코드 길이 = 5CH
C+2: D302	0 0 0 3	정렬 대상 데이터 오프셋 = 3CH
C+3: D303	0 0 0 0	분할 정렬 설정 = 무효/오름차순
C+4: D304	8 0 0 1	인덱스 레지스터 출력 지정 = 출력 있음/정렬 종료 설정 = 유효
C+5: D305	F F F F	종료 데이터 = #FFFF
C+6: D306	F F F F	종료 데이터 = #FFFF
C+7: D307 ~ C+86: D386		시스템 워크 영역

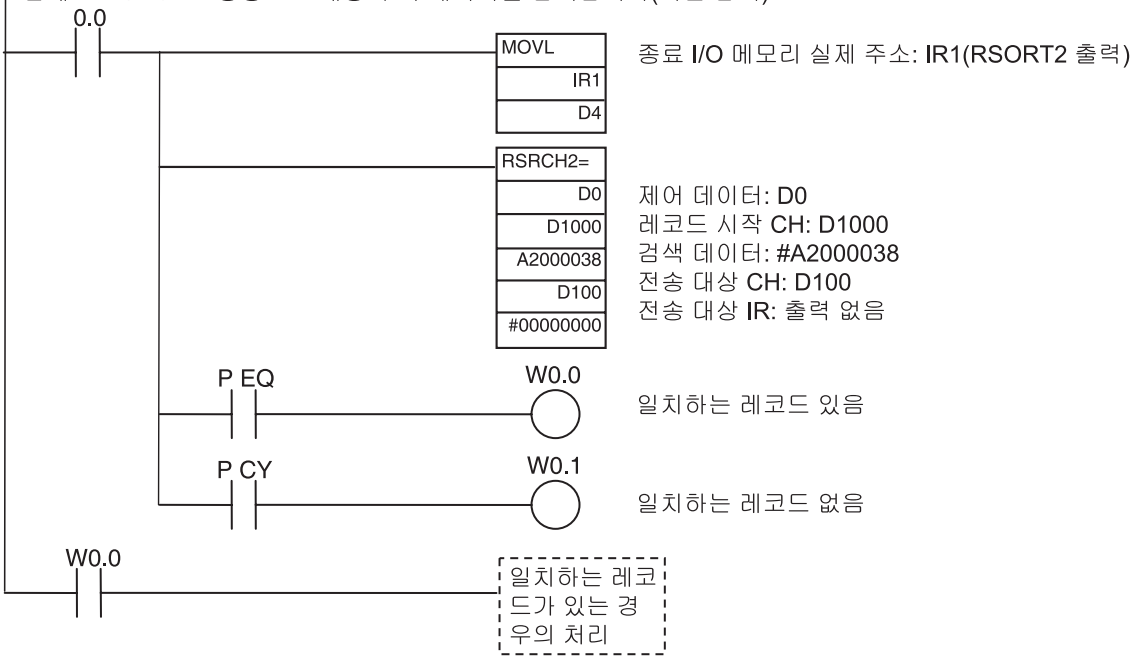
RSRCH2

C: D0	0 0 0 8	레코드 수 = 8CH
C+1: D1	0 0 0 5	레코드 길이 = 5CH
C+2: D2	0 0 0 3	검색 대상 데이터 오프셋 = 3H
C+3: D3	0 0 1 2	인덱스 레지스터 출력 지정 = 있음 검색 방식 = 이분 탐색 오름차순/검색 종료 설정 = 유효: 종료 I/O 메모리 실제 주소 지정
C+4: D4	0 4 0 6	검색 종료 I/O 메모리 실제 주소(하위)
C+5: D5	0 0 0 1	검색 종료 I/O 메모리 실제 주소(상위)

단계 1: RSORT2 명령으로 배송지 코드를 오름차순으로 정렬합니다.



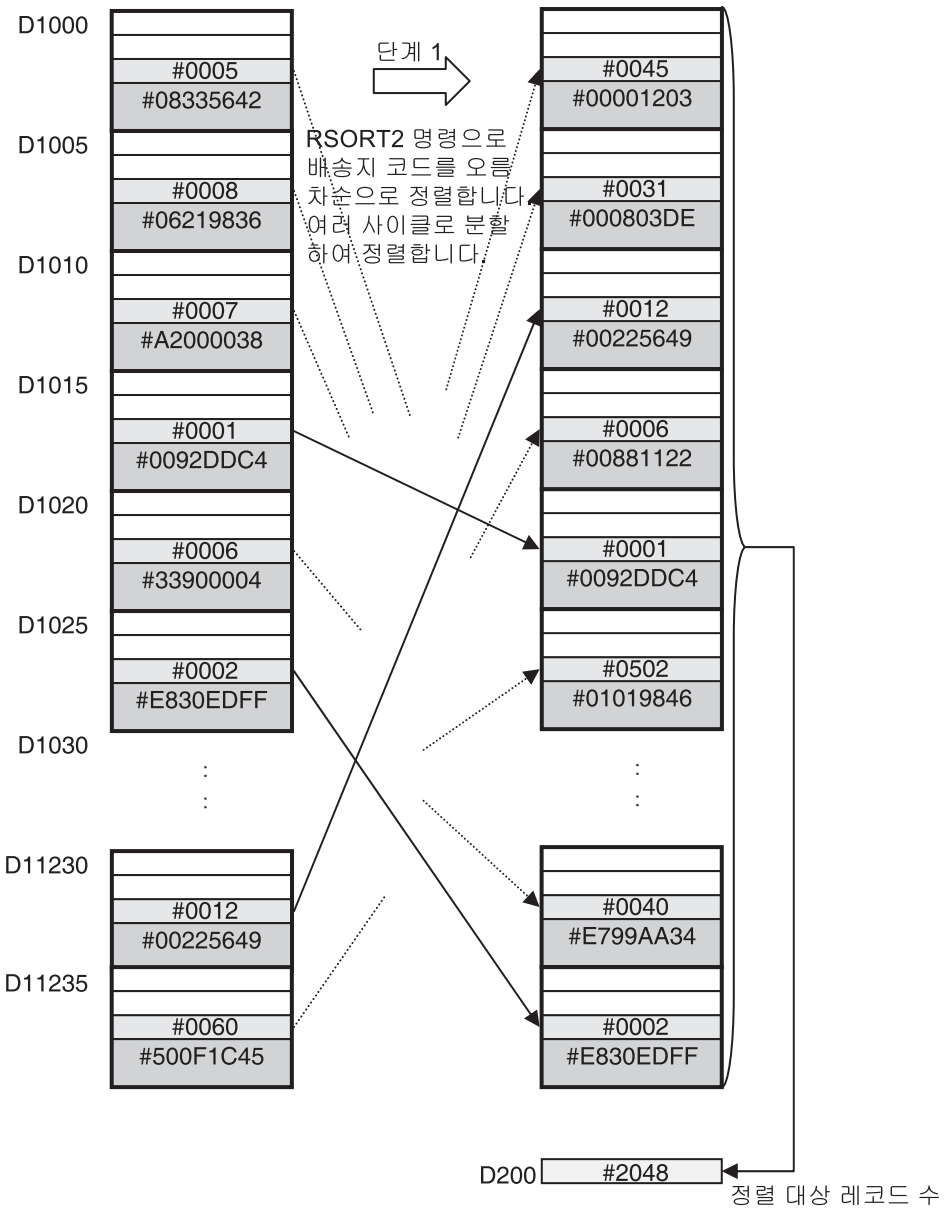
단계 2: RSRCH2=명령으로 해당 추적 데이터를 검색합니다(이분 탐색).



■ 레코드 수가 많은 추적 데이터를 정렬하는 경우(분할 정렬)

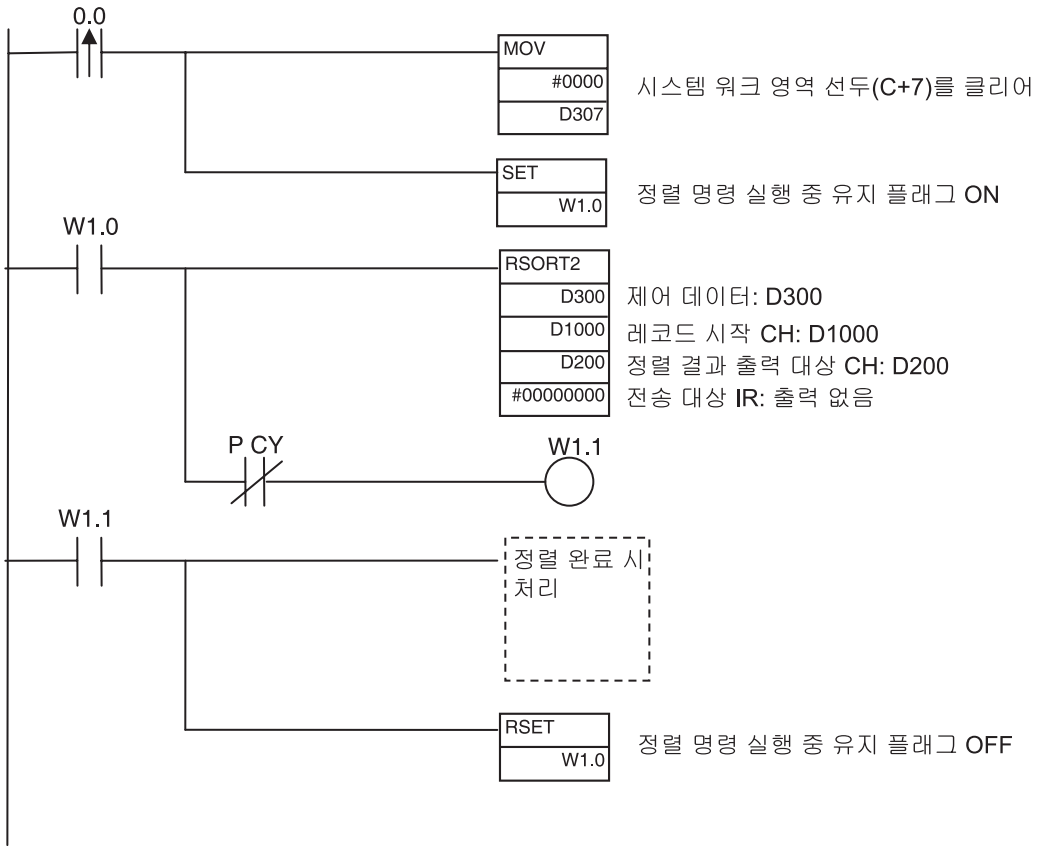
2048개의 레코드가 등록된 추적 데이터를 배송지 코드로 정렬합니다. 사이클 시간을 단축하기 위해 여러 사이클로 분할하여 정렬합니다.

레코드 시작 CH	D1000
레코드 길이	5
레코드 수	2048
정렬 결과 출력 CH	D200
정렬 대상	배송지 코드
정렬 옵션	오름차순, 분할 정렬 설정 있음
정렬 종료 설정	없음
정렬 결과 인덱스 레지스터 출력	없음



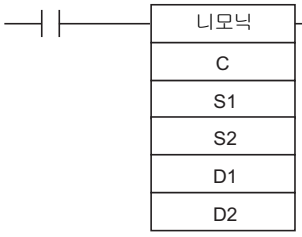
C: D300	2 0 4 8	레코드 수 = 2048CH
C+1: D301	0 0 0 5	레코드 길이 = 5CH
C+2: D302	0 0 0 3	정렬 대상 데이터 오프셋 = 3CH
C+3: D303	8 0 0 0	분할 정렬 설정 = 유효/오름차순
C+4: D304	0 0 0 0	인덱스 레지스터 출력 지정 = 출력 없음/정렬 종료 설정 = 무효
C+5: D305	0 0 0 0	#0000 고정
C+6: D306	0 0 0 0	#0000 고정
C+7: D307 ~ C+86: D386		시스템 워크 영역

단계 1: RSORT2 명령으로 배송지 코드를 오름차순으로 정렬합니다.



RSRCH< , RSRCH<= , RSRCH= , RSRCH> , RSRCH>=

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부호 없는 1W 레코드 검색	RSRCH< RSRCH<= RSRCH= RSRCH> RSRCH>=	@RSRCH< @RSRCH<= @RSRCH= @RSRCH> @RSRCH>=	360 361 362 363 364	지정한 테이블 내의 여러 레코드 중에서 레코드 내 지정한 위치의 데이터(1W)를 검색하고, 일치하는 레코드가 있으면 레코드 번호와 레코드 데이터를 출력합니다.

심볼	RSRCH<, RSRCH<=, RSRCH=, RSRCH>, RSRCH>=
	 C: 제어 데이터 S1: 검색 시작 레코드의 시작 CH 번호 S2: 검색 데이터 저장 CH 번호 D1: 전송 대상 CH 번호 D2: 전송 대상 인덱스 레지스터 번호

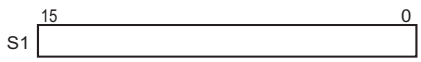
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

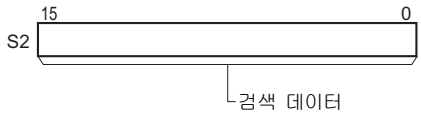
피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	WORD	6
S1	검색 시작 레코드의 시작 CH 번호	WORD	가변
S2	검색 데이터 저장 CH 번호	WORD	1
D1	전송 대상 CH 번호	WORD	가변
D2	전송 대상 인덱스 레지스터 번호	DWORD	2

S1: 검색 시작 레코드의 시작 CH 번호

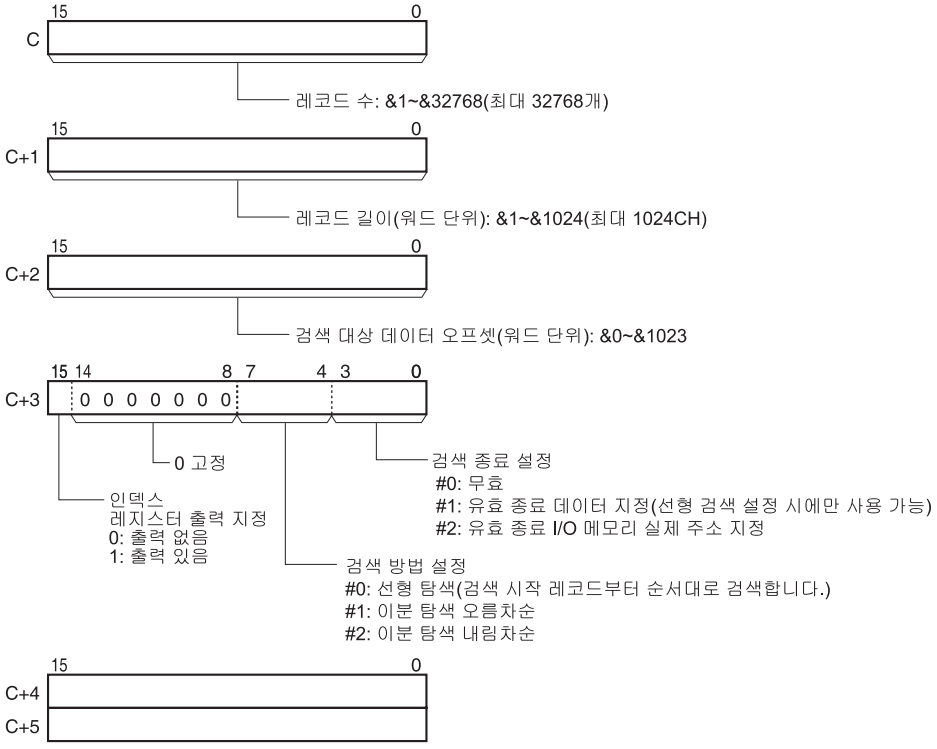


S2: 검색 데이터 저장 CH 번호



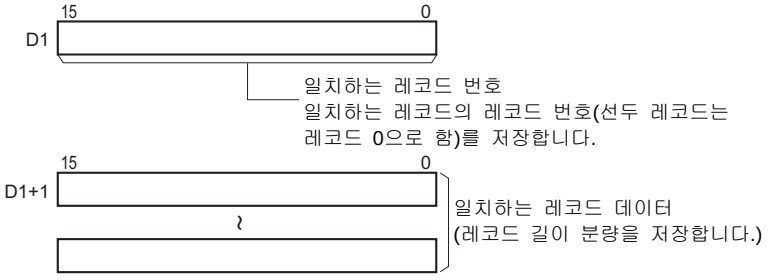
RSRCH< , RSRCH<= , RSRCH= , RSRCH> , RSRCH>=

C: 제어 데이터

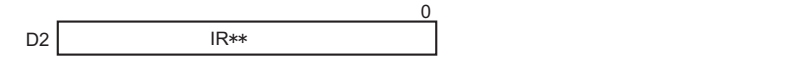


■ 검색 종료 설정 무효의 경우
C+4: 0 고정
C+5: 0 고정
■ 검색 종료 설정이 유효이고 종료 데이터를 지정하는 경우
C+4: 검색 종료 데이터(#0000~FFFF)
C+5: 0 고정
■ 검색 종료 설정이 유효이고 종료 I/O 메모리 실제 주소를 지정하는 경우
C+4: 검색 종료 I/O 메모리 실제 주소(하위)
C+5: 검색 종료 I/O 메모리 실제 주소(상위)

D1: 전송 대상 CH 번호



D2: 전송 대상 인덱스 레지스터 번호



일치하는 레코드의 I/O 메모리 실제 주소를 인덱스 레지스터에 저장합니다.
C+3의 인덱스 레지스터 출력 지정을 '출력 없음'으로 설정한 경우는 D2에 #00000000을 지정하십시오.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S1																		
S2											○	○						
D1											–	–						
D2	–	–	–	–	–	–	–	–	–	–	○		○	–				

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 레코드 길이(C+1)가 1~1024가 아닐 때 ON · 레코드 수(C)×레코드 길이(C+1)가 32768을 초과했을 때 ON · 레코드 수(C)×레코드 길이(C+1)가 0일 때 ON · 검색 데이터 오프셋(C+2)이 레코드 길이(C+1)보다 1 작은 값을 초과했을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 검색 데이터를 찾았을 때 ON · 검색 데이터를 찾지 못했을 때 OFF
캐리 플래그	P_CY	[선형 탐색 시] · 최종 레코드 또는 검색 종료 설정 조건과 일치하는 레코드까지 검색을 마치기 전에 검색 데이터를 찾았을 때 ON · 최종 레코드 또는 검색 종료 설정 조건과 일치하는 레코드까지 검색을 마쳤을 때 OFF [이분 탐색 시] · 반드시 OFF

기능

S1로 지정된 시작 레코드부터 C로 지정된 레코드 수만큼 S2의 검색 데이터로 레코드를 검색합니다. 검색 조건이 일치한 경우 = 플래그가 ON이 되며, 일치한 레코드 번호를 D1에 출력하고 일치한 레코드 데이터는 D1+1 이후에 출력합니다. 또한 C+3의 인덱스 레지스터 출력 지정을 '출력 있음'으로 설정한 경우는 일치한 레코드 선두의 I/O 메모리 실제 주소를 IR에 출력합니다. 일치한 레코드가 여러 개 있으면 일치한 레코드 중 시작 레코드에 가까운 레코드를 출력합니다.
조건이 일치하지 않은 경우는 D1 이후에 아무 것도 출력되지 않습니다(명령 실행 전의 값을 유지).

검색 방법에는 2종류가 있으며 C＋3의 검색 방법 결정에 따라 지정됩니다.

· 선형 탐색법

검색 시작 레코드부터 순서대로 검색합니다. 검색 데이터가 정렬되지 않은 경우에 사용합니다. C로 지정한 최대 레코드 수만큼 검색하기 전에 검색 데이터를 찾은 경우 캐리 플래그를 ON합니다(검색하지 않은 레코드가 있음을 나타냄).

· 이분 탐색법(오름차순/내림차순)

검색 데이터가 오름차순 또는 내림차순으로 정렬된 경우, 이분 탐색법을 사용하도록 지정하면 명령 실행 처리를 고속으로 수행할 수 있습니다.

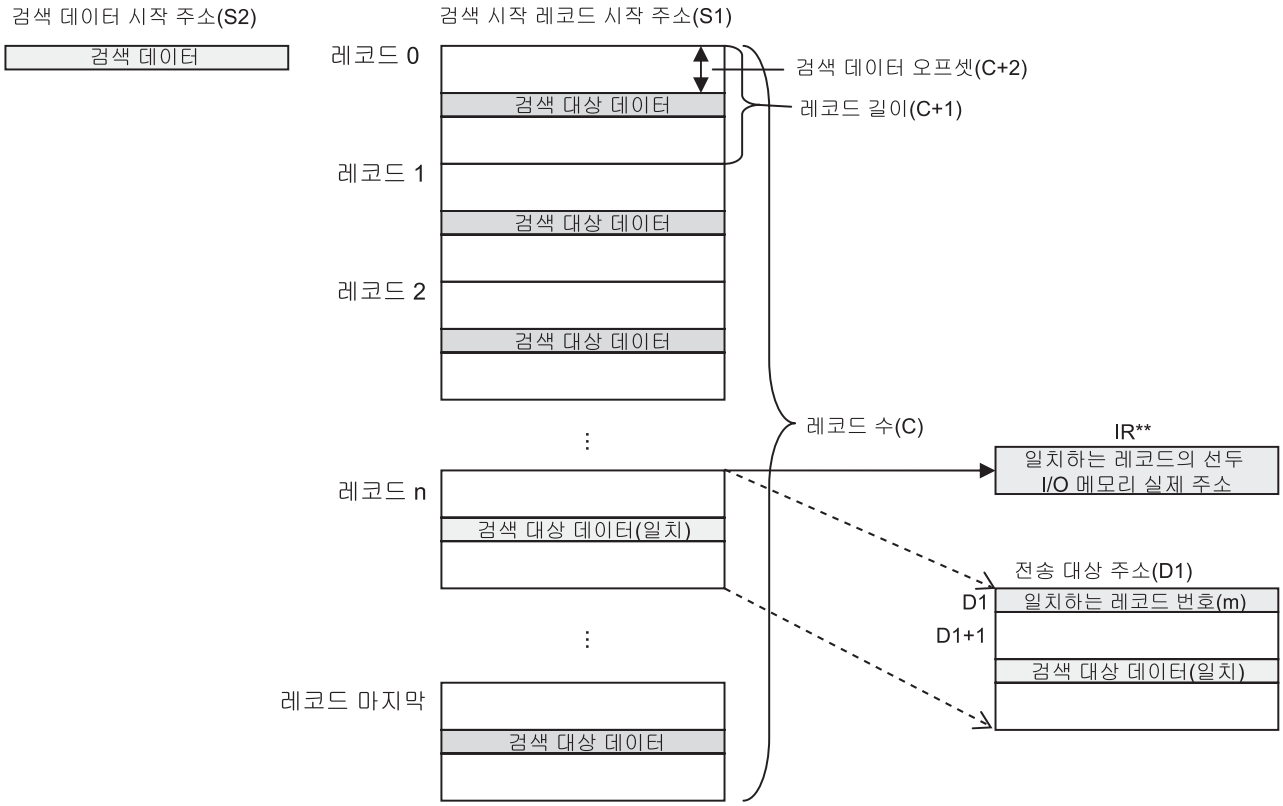
사용상의 주의 사항

· 검색 방법을 '이분 탐색'으로 설정한 경우 검색 데이터가 오름차순 또는 내림차순으로 정렬되어 있지 않으면 명령 실행 결과가 불안정해집니다.

· 검색 종료를 '유효 종료 데이터 지정'으로 설정한 경우는 이분 탐색으로 검색할 수 없습니다. 검색 종료를 '유효 종료 데이터 지정'으로 설정하고, 검색 방법을 '이분 탐색'으로 설정한 경우에는 선형 탐색으로 검색합니다.

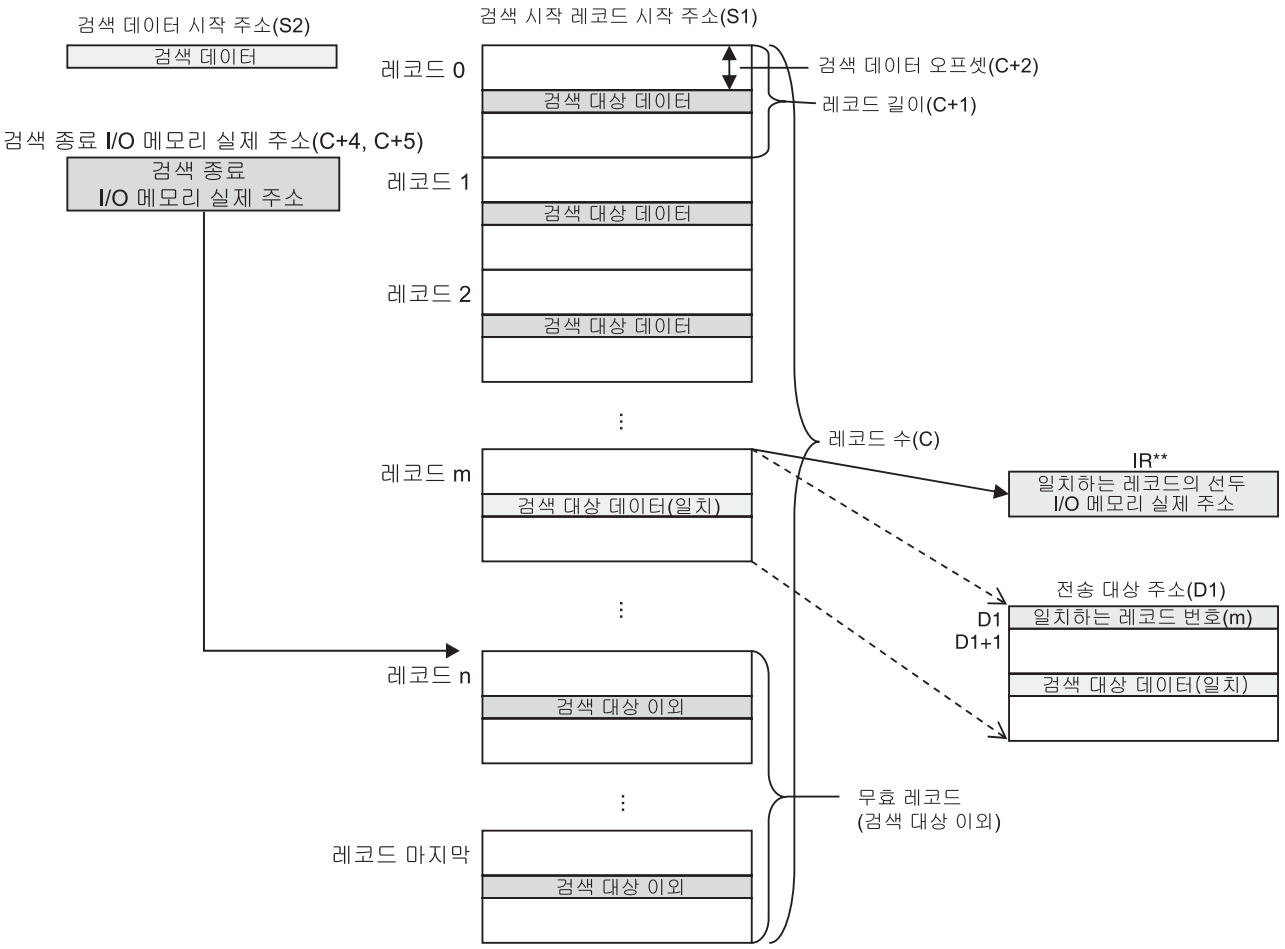
■ 검색 종료 설정이 무효인 경우

레코드 수(C)를 검색 범위로 판단하고 레코드 수(C)만큼 탐색합니다.



■ 검색 종료 설정이 유효이고 종료 데이터를 지정하는 경우

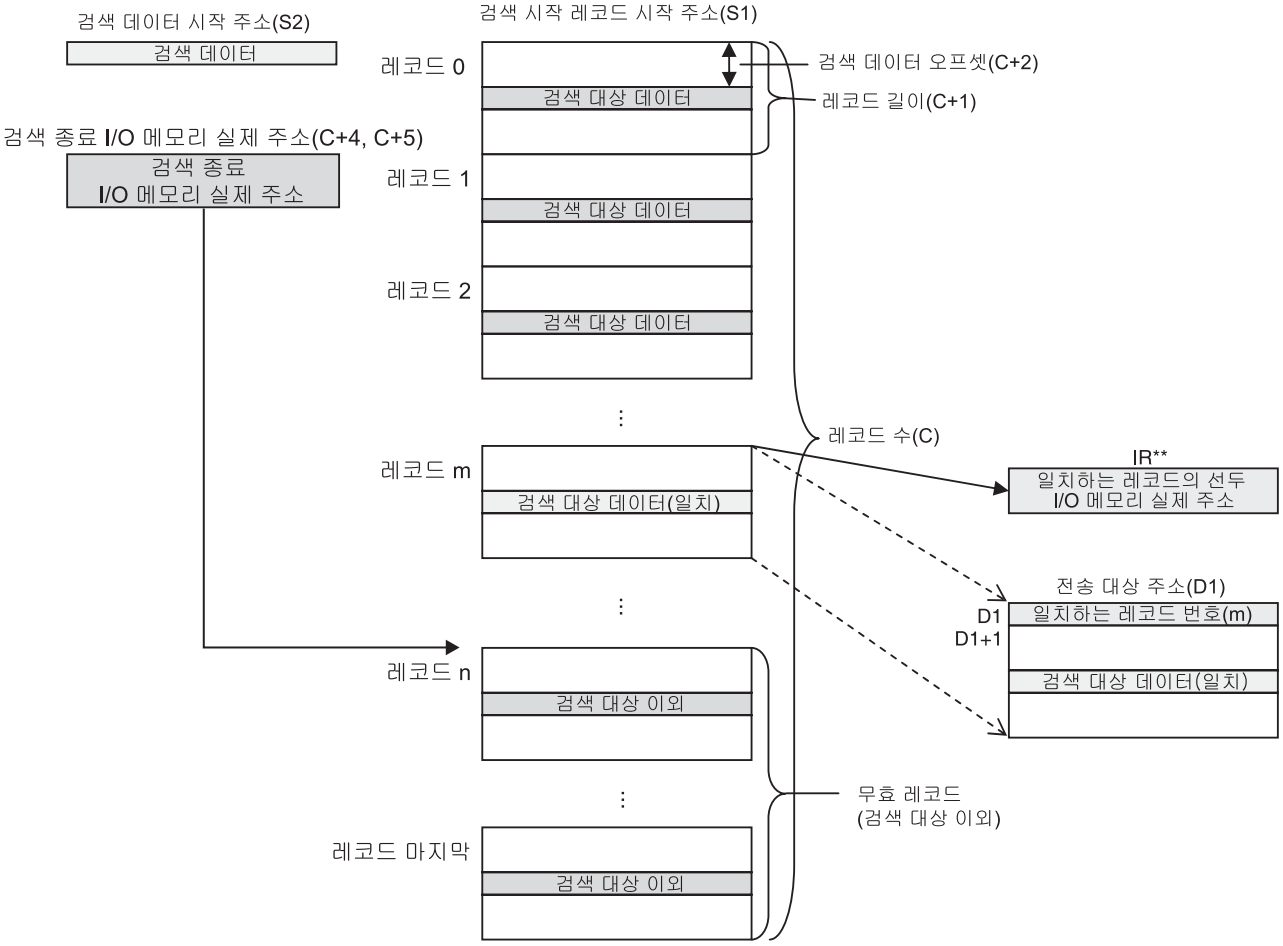
C+3의 검색 종료 설정에서 '유효 종료 데이터 지정'을 지정한 경우, C+4에 있는 검색 종료 데이터를 찾은 레코드를 검색 대상 레코드의 끝이라고 판단하고, 종료 레코드 직전의 레코드까지 검색합니다. 종료 데이터를 찾지 못한 경우는 레코드 수(C)를 검색 범위로 판단합니다.



RSRCH< , RSRCH<= , RSRCH= , RSRCH> , RSRCH>=

■ 검색 종료 설정이 유효이고 종료 I/O 메모리 실제 주소를 지정하는 경우

C+3의 검색 종료 설정에서 '유효 종료 I/O 메모리 실제 주소 지정'을 지정한 경우, C+4, C+5로 나타내는 I/O 메모리 실제 주소까지 검색 대상으로 합니다. 종료 I/O 메모리 실제 주소 이후의 레코드는 검색 대상에서 제외됩니다. 종료 I/O 메모리 실제 주소가 최종 레코드 이후에 설정된 경우는 레코드 수(C)를 검색 범위로 판단합니다.



RSRCH2< , RSRCH2<= , RSRCH2= , RSRCH2> , RSRCH2>=

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부호 없는 2W 레코드 검색	RSRCH2< RSRCH2<= RSRCH2= RSRCH2> RSRCH2>=	@RSRCH2< @RSRCH2<= @RSRCH2= @RSRCH2> @RSRCH2>=	370 371 372 373 374	지정한 테이블 내의 여러 레코드 중에서 레코드 내 지정한 위치의 데이터(2W)를 검색하고, 일치하는 레코드가 있으면 레코드 번호와 레코드 데이터를 출력합니다.

심볼	RSRCH2<, RSRCH2<=, RSRCH2=, RSRCH2>, RSRCH2>=



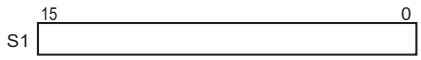
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

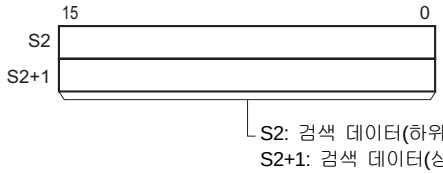
피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	WORD	6
S1	검색 시작 레코드의 시작 CH 번호	WORD	가변
S2	검색 데이터 저장 CH 번호	DWORD	2
D1	전송 대상 CH 번호	WORD	가변
D2	전송 대상 인덱스 레지스터 번호	DWORD	2

S1: 검색 시작 레코드의 시작 CH 번호

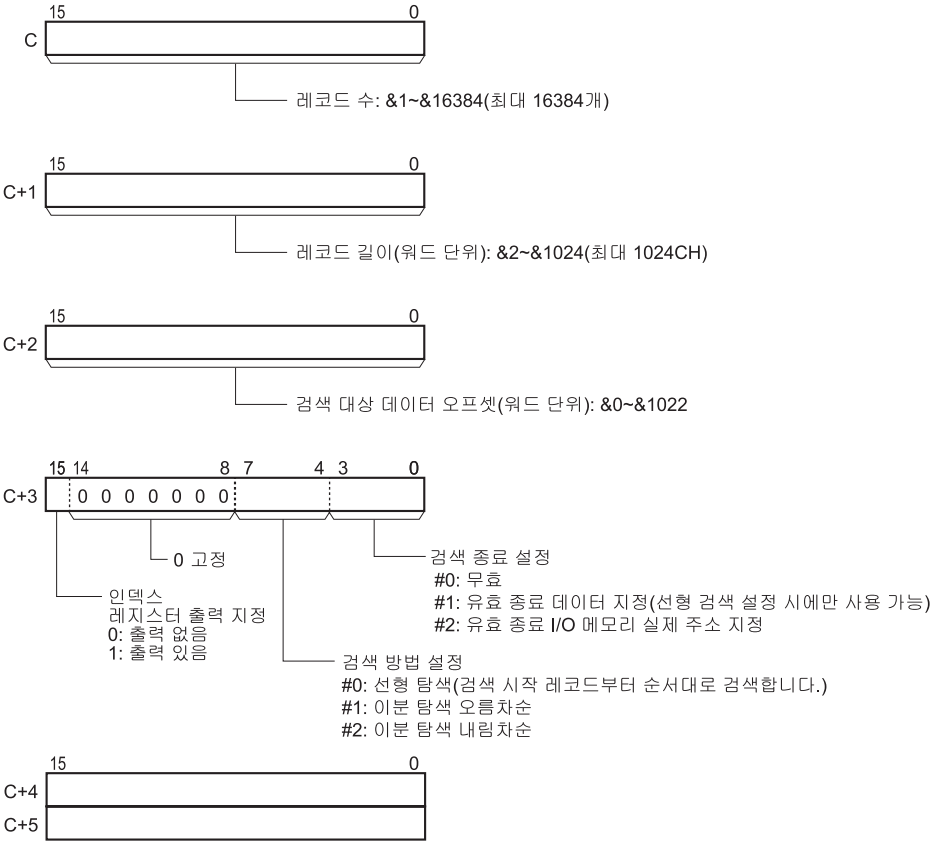


S2: 검색 데이터 저장 CH 번호



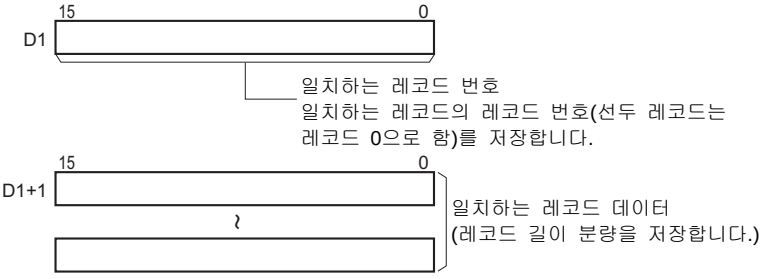
RSRCH2< , RSRCH2<= , RSRCH2= , RSRCH2> , RSRCH2>=

C: 제어 데이터

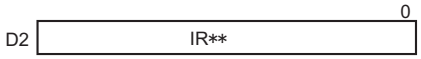


■ 검색 종료 설정 무효의 경우
C+4: 0 고정
C+5: 0 고정
■ 검색 종료 설정이 유효이고 종료 데이터를 지정하는 경우
C+4,C+5: 검색 종료 데이터(#00000000~FFFFFFFF)
■ 검색 종료 설정이 유효이고 종료 I/O 메모리 실제 주소를 지정하는 경우
C+4: 검색 종료 I/O 메모리 실제 주소(하위)
C+5: 검색 종료 I/O 메모리 실제 주소(상위)

D1: 전송 대상 CH 번호



D2: 전송 대상 인덱스 레지스터 번호



일치하는 레코드의 I/O 메모리 실제 주소를 인덱스
레지스터에 저장합니다.
C+3의 인덱스 레지스터 출력 지정을 '출력 없음'으로
설정한 경우는 D2에 #00000000을 지정하십시오.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S1																		
S2											○							
D1																		
D2	–	–	–	–	–	–	–	–	–	–			○	–				

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 레코드 길이(C+1)가 2~1024가 아닐 때 ON · 레코드 수(C)×레코드 길이(C+1)가 32768을 초과했을 때 ON · 레코드 수(C)×레코드 길이(C+1)가 0일 때 ON · 검색 데이터 오프셋(C+2)이 레코드 길이(C+1)보다 2 작은 값을 초과했을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 검색 데이터를 찾았을 때 ON · 검색 데이터를 찾지 못했을 때 OFF
캐리 플래그	P_CY	[선형 탐색 시] · 최종 레코드 또는 검색 종료 설정 조건과 일치하는 레코드까지 검색을 마치기 전에 검색 데이터를 찾았을 때 ON · 최종 레코드 또는 검색 종료 설정 조건과 일치하는 레코드까지 검색을 마쳤을 때 OFF [이분 탐색 시] · 반드시 OFF

기능

S1로 지정된 시작 레코드부터 C로 지정된 레코드 수만큼 S2, S2+1의 검색 데이터로 레코드를 검색합니다. 검색 조건이 일치한 경우 = 플래그가 ON이 되며, 일치한 레코드 번호를 D1에 출력하고 일치한 레코드 데이터는 D1+1 이후에 출력합니다. 또한 C+3의 인덱스 레지스터 출력 지정을 '출력 있음'으로 설정한 경우는 일치한 레코드 선두의 I/O 메모리 실제 주소를 IR에 출력합니다. 이때, = 플래그는 ON이 됩니다. 일치한 레코드가 여러 개 있으면 일치한 레코드 중 시작 레코드에 가까운 레코드를 출력합니다.

조건이 일치하지 않은 경우는 D1 이후에 아무 것도 출력되지 않습니다(명령 실행 전의 값을 유지).

검색 방법에는 2종류가 있으며 C+3의 검색 방법 결정에 따라 지정됩니다.

· 선형 탐색법

검색 시작 레코드부터 순서대로 검색합니다. 검색 데이터가 정렬되지 않은 경우에 사용합니다. C로 지정한 최대 레코드 수만큼 검색하기 전에 검색 데이터를 찾은 경우 캐리 플래그를 ON합니다(검색하지 않은 레코드가 있음을 나타냄).

· 이분 탐색법(오름차순/내림차순)

검색 데이터가 오름차순 또는 내림차순으로 정렬된 경우, 이분 탐색법을 사용하도록 지정하면 명령 실행 처리를 고속으로 수행할 수 있습니다.

사용상의 주의 사항

· 검색 방법을 '이분 탐색'으로 설정한 경우 검색 데이터가 오름차순 또는 내림차순으로 정렬되어 있지 않으면 명령 실행 결과가 불안정해집니다.

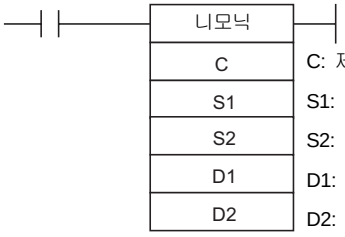
· 검색 종료를 '유효 종료 데이터 지정'으로 설정한 경우는 이분 탐색으로 검색할 수 없습니다. 검색 종료를 '유효 종료 데이터 지정'으로 설정하고, 검색 방법을 '이분 탐색'으로 설정한 경우에는 선형 탐색으로 검색합니다.

RSRCH4< , RSRCH4<= , RSRCH4= , RSRCH4> , RSRCH4>=

명령 이름	니모닉	동작 옵션	평션 번호	기능
부호 없는 4W 레코드 검색	RSRCH4< RSRCH4<= RSRCH4= RSRCH4> RSRCH4>=	@RSRCH4< @RSRCH4<= @RSRCH4= @RSRCH4> @RSRCH4>=	380 381 382 383 384	지정한 테이블 내의 여러 레코드 중에서 레코드 내 지정한 위치의 데이터(4W)를 검색하고, 일치하는 레코드가 있으면 레코드 번호와 레코드 데이터를 출력합니다.

심볼

RSRCH4<, RSRCH4<=, RSRCH4=, RSRCH4>, RSRCH4>=



C: 제어 데이터
S1: 검색 시작 레코드의 시작 CH 번호
S2: 검색 데이터 저장 CH 번호
D1: 전송 대상 CH 번호
D2: 전송 대상 인덱스 레지스터 번호

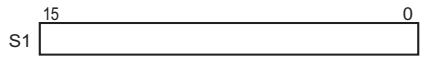
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

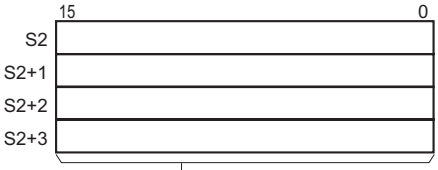
피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	WORD	8
S1	검색 시작 레코드의 시작 CH 번호	WORD	가변
S2	검색 데이터 저장 CH 번호	LWORD	4
D1	전송 대상 CH 번호	WORD	가변
D2	전송 대상 인덱스 레지스터 번호	DWORD	2

S1: 검색 시작 레코드의 시작 CH 번호



S2: 검색 데이터 저장 CH 번호

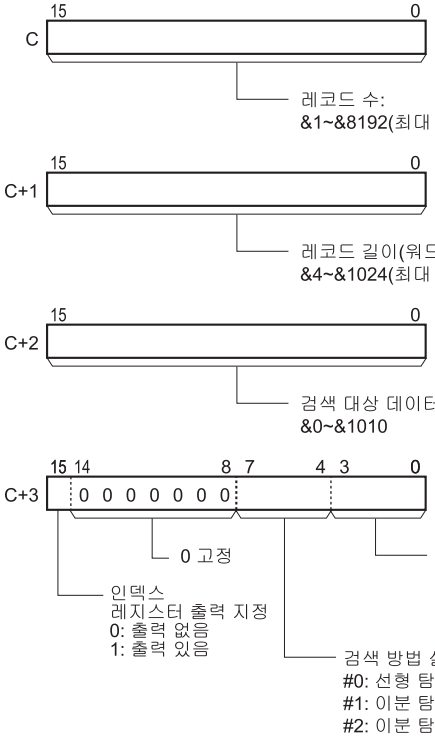


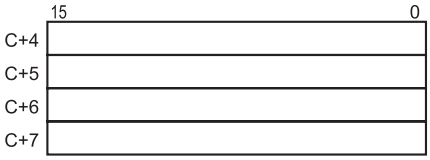
S2 : 검색 데이터 (S2CH)
S2+1 : 검색 데이터(S2+1CH)
S2+2 : 검색 데이터(S2+2CH)
S2+3 : 검색 데이터(S2+3CH)

예를 들어, 검색 데이터의 값이 #0123456789ABCDEF일 때

S2	# C D E F
S2+1	# 8 9 A B
S2+2	# 4 5 6 7
S2+3	# 0 1 2 3

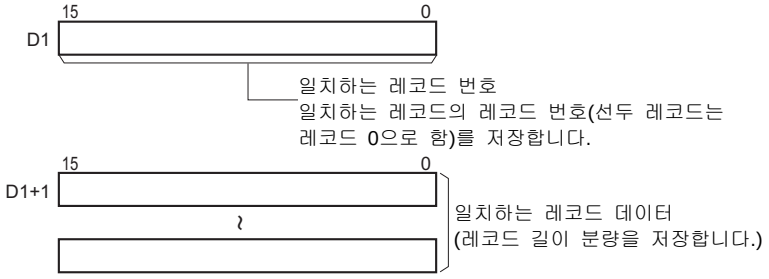
C: 제어 데이터



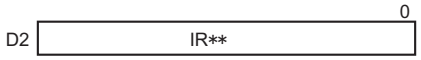


■ 검색 종료 설정이 무효인 경우
C+4: 0 고정
C+5: 0 고정
C+6: 0 고정
C+7: 0 고정
■ 검색 종료 설정이 유효이고 종료 데이터를 지정하는 경우
C+4~C+7: 검색 종료 데이터
(#0000000000000000~FFFFFFFFFFFFFFFF)
C+4에서 최하위 16비트→하위 16비트→상위 16비트
→최상위 16비트 순서로 저장합니다.
■ 검색 종료 설정이 유효이고 종료 I/O 메모리 실제 주소를 지정하는 경우
C+4: 검색 종료 I/O 메모리 실제 주소(하위)
C+5: 검색 종료 I/O 메모리 실제 주소(상위)
C+6: 0 고정
C+7: 0 고정

D1: 전송 대상 CH 번호



D2: 전송 대상 인덱스 레지스터 번호



일치하는 레코드의 I/O 메모리 실제 주소를 인덱스
레지스터에 저장합니다.
C+3의 인덱스 레지스터 출력 지정을 '출력 없음'으로
설정한 경우는 D2에 #00000000을 지정하십시오.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S1																		
S2																		
D1																		
D2	–	–	–	–	–	–	–	–	–	–	○		○	–				

RSRCH4< , RSRCH4<= , RSRCH4= , RSRCH4> , RSRCH4>=

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 레코드 길이(C+1)가 4~1024가 아닐 때 ON · 레코드 수(C)×레코드 길이(C+1)가 32768을 초과했을 때 ON · 레코드 수(C)×레코드 길이(C+1)가 0일 때 ON · 검색 데이터 오프셋(C+2)이 레코드 길이(C+1)보다 4 작은 값을 초과했을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 검색 데이터를 찾았을 때 ON · 검색 데이터를 찾지 못했을 때 OFF
캐리 플래그	P_CY	[선형 탐색 시] · 최종 레코드 또는 검색 종료 설정 조건과 일치하는 레코드까지 검색을 마치기 전에 검색 데이터를 찾았을 때 ON · 최종 레코드 또는 검색 종료 설정 조건과 일치하는 레코드까지 검색을 마쳤을 때 OFF [이분 탐색 시] · 반드시 OFF

기능

S1로 지정된 시작 레코드부터 C로 지정된 레코드 수만큼 S2, S2+1, S2+2, S2+3의 검색 데이터로 레코드를 검색합니다. 검색 조건이 일치한 경우 = 플래그가 ON이 되며, 일치한 레코드 번호를 D1에 출력하고 일치한 레코드 데이터는 D1+1 이후에 출력합니다. 또한 C+3의 인덱스 레지스터 출력 지정을 '출력 있음'으로 설정한 경우는 일치한 레코드 선두의 I/O 메모리 실제 주소를 IR에 출력합니다. 이때, = 플래그는 ON이 됩니다. 일치한 레코드가 여러 개 있으면 일치한 레코드 중 시작 레코드에 가까운 레코드를 출력합니다.

조건이 일치하지 않은 경우는 D1 이후에 아무 것도 출력되지 않습니다(명령 실행 전의 값을 유지).

검색 방법에는 2종류가 있으며 C＋3의 검색 방법 결정에 따라 지정됩니다.

· 선형 탐색법

검색 시작 레코드부터 순서대로 검색합니다. 검색 데이터가 정렬되지 않은 경우에 사용합니다. C로 지정한 최대 레코드 수만큼 검색하기 전에 검색 데이터를 찾은 경우 캐리 플래그를 ON합니다(검색하지 않은 레코드가 있음을 나타냄).

· 이분 탐색법(오름차순/내림차순)

검색 데이터가 오름차순 또는 내림차순으로 정렬된 경우, 이분 탐색법을 사용하도록 지정하면 명령 실행 처리를 고속으로 수행할 수 있습니다.

사용상의 주의 사항

· 검색 방법을 '이분 탐색'으로 설정한 경우 검색 데이터가 오름차순 또는 내림차순으로 정렬되어 있지 않으면 명령 실행 결과가 불안정해집니다.

· 검색 종료를 '유효 종료 데이터 지정'으로 설정한 경우는 이분 탐색으로 검색할 수 없습니다. 검색 종료를 '유효 종료 데이터 지정'으로 설정하고, 검색 방법을 '이분 탐색'으로 설정한 경우에는 선형 탐색으로 검색합니다.

RSORT

명령 이름	니모닉	동작 옵션	평션 번호	기능
부호 없는 1W 레코드 정렬	RSORT	@RSORT	203	지정한 테이블 내의 여러 레코드를 레코드 내 지정 위치의 데이터(1W)로 정렬합니다.

심볼	RSORT
	

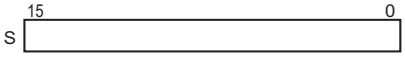
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

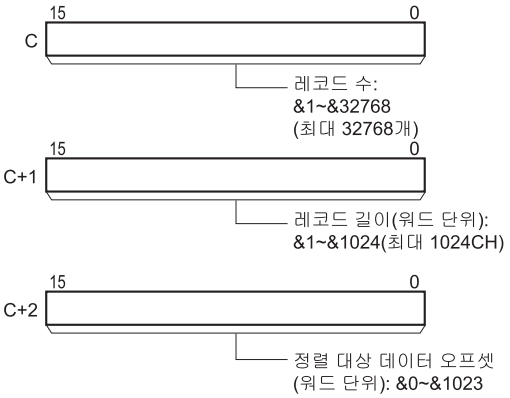
피연산자 설명

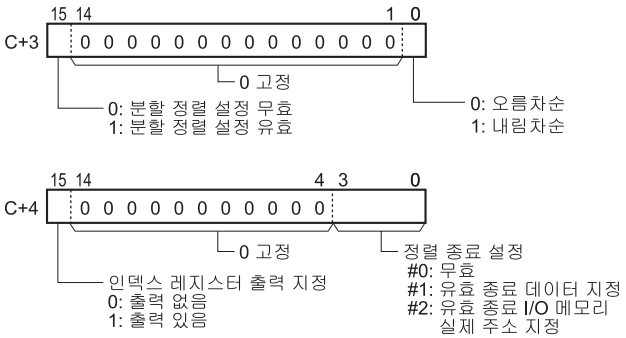
피연산자	내용	데이터 형식	크기
C	제어 데이터	WORD	87
S	정렬 시작 레코드의 시작 CH 번호	WORD	가변
D1	정렬 결과 출력 대상 CH 번호	WORD	1
D2	전송 대상 인덱스 레지스터 번호	DWORD	2

S: 정렬 시작 레코드의 시작 CH 번호

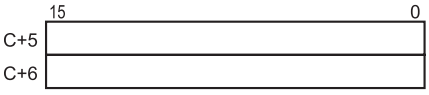


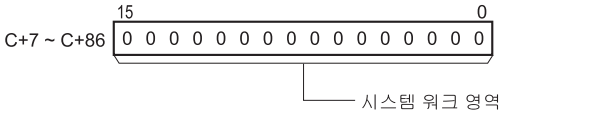
C: 제어 데이터





RSORT





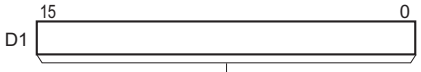
시스템 워크 영역
정렬을 시작할 때는 C+7을
반드시 0으로 클리어하십시오.

■ 정렬 종료 설정이 무효인 경우
C+5: 0 고정
C+6: 0 고정

■ 정렬 종료 설정이 유효이고 종료 데이터를 지정하는 경우
C+5: 정렬 종료 데이터(#0000~FFFF)
C+6: 0 고정

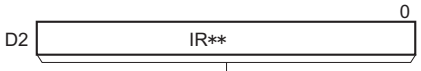
■ 정렬 종료 설정이 유효이고 종료 I/O 메모리 실제 주소를 지정하는 경우
C+5: 정렬 종료 I/O 메모리 실제 주소(하위)
C+6: 정렬 종료 I/O 메모리 실제 주소(상위)

D1: 정렬 결과 출력 대상 CH 번호



정렬 대상 레코드 수
정렬 대상이 된 레코드 수를 저장합니다.

D2: 전송 대상 인덱스 레지스터 번호



정렬 대상이 된 범위의 최종 I/O 메모리 실제 주소+1
을 인덱스 레지스터에 저장합니다.
C+4의 인덱스 레지스터 출력 지정을 '출력 없음'으로
설정한 경우는 D2에 #00000000을 지정하십시오.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S																		
D1												○						
D2	–	–	–	–	–	–	–	–	–	–	○	–	○	–				

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 레코드 길이(C+1)가 1~1024가 아닐 때 ON · 레코드 수(C)×레코드 길이(C+1)가 32768을 초과했을 때 ON · 레코드 수(C)×레코드 길이(C+1)가 0일 때 ON · 정렬 대상 데이터 오프셋(C+2)이 레코드 길이(C+1)보다 작은 값을 초과했을 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 정렬 중일 때 ON · 정렬 완료 시 OFF

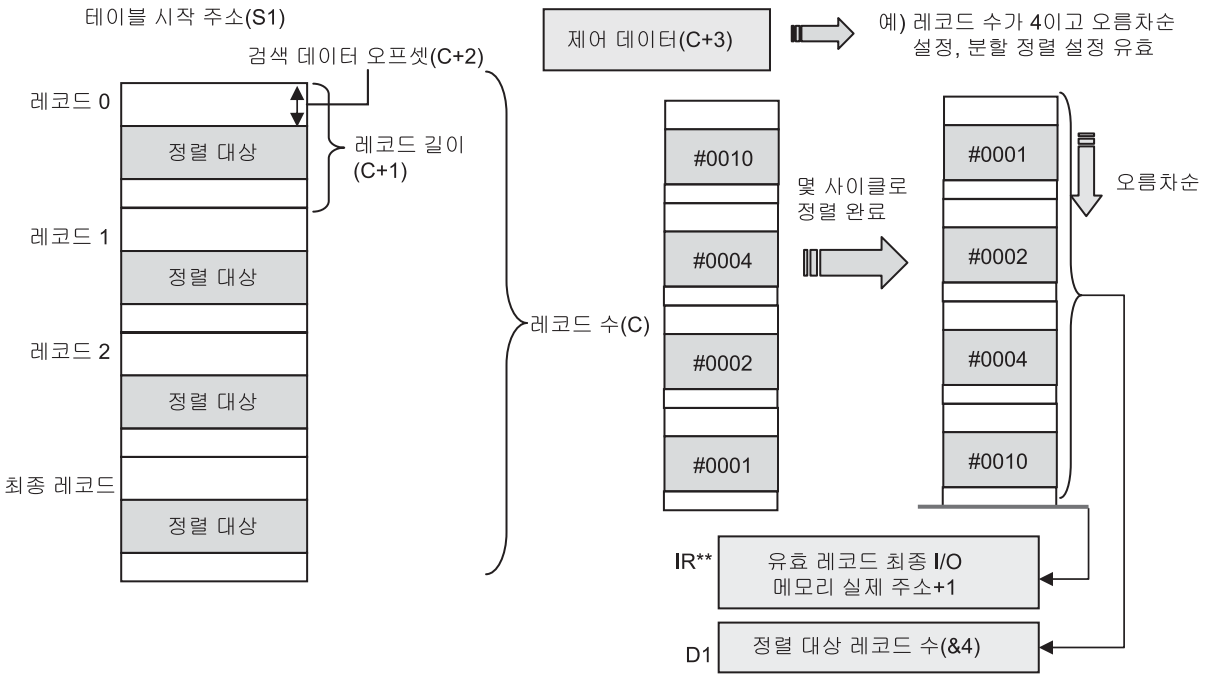
기능

S로 지정된 시작 레코드부터 C의 지정된 레코드 수까지의 데이터를 지정된 위치의 데이터(1W)를 키로 하여 레코드 단위로 정렬합니다. 정렬이 종료되면 캐리 플래그가 OFF가 되고 정렬 대상 레코드 수를 D1에 출력합니다. 또한 C+4의 인덱스 레지스터 출력 지정을 '출력 있음'으로 설정한 경우는 정렬 대상이 된 범위의 최종 I/O 메모리 실제 주소+1을 IR에 출력합니다. 정렬 방향은 C+3의 설정값에 따라 각각 오름차순/내림차순으로 정렬합니다.

C+3의 분할 정렬 설정을 유효로 한 경우는 여러 사이클에 걸쳐 정렬을 실행합니다. 정렬 실행 중에는 캐리 플래그가 ON되고 정렬을 완료하면 OFF됩니다. 정렬 완료 전에 RSORT 명령 실행 조건이 OFF가 된 경우는 다음에 실행 조건이 ON이 되면 이어서 정렬을 실행합니다.

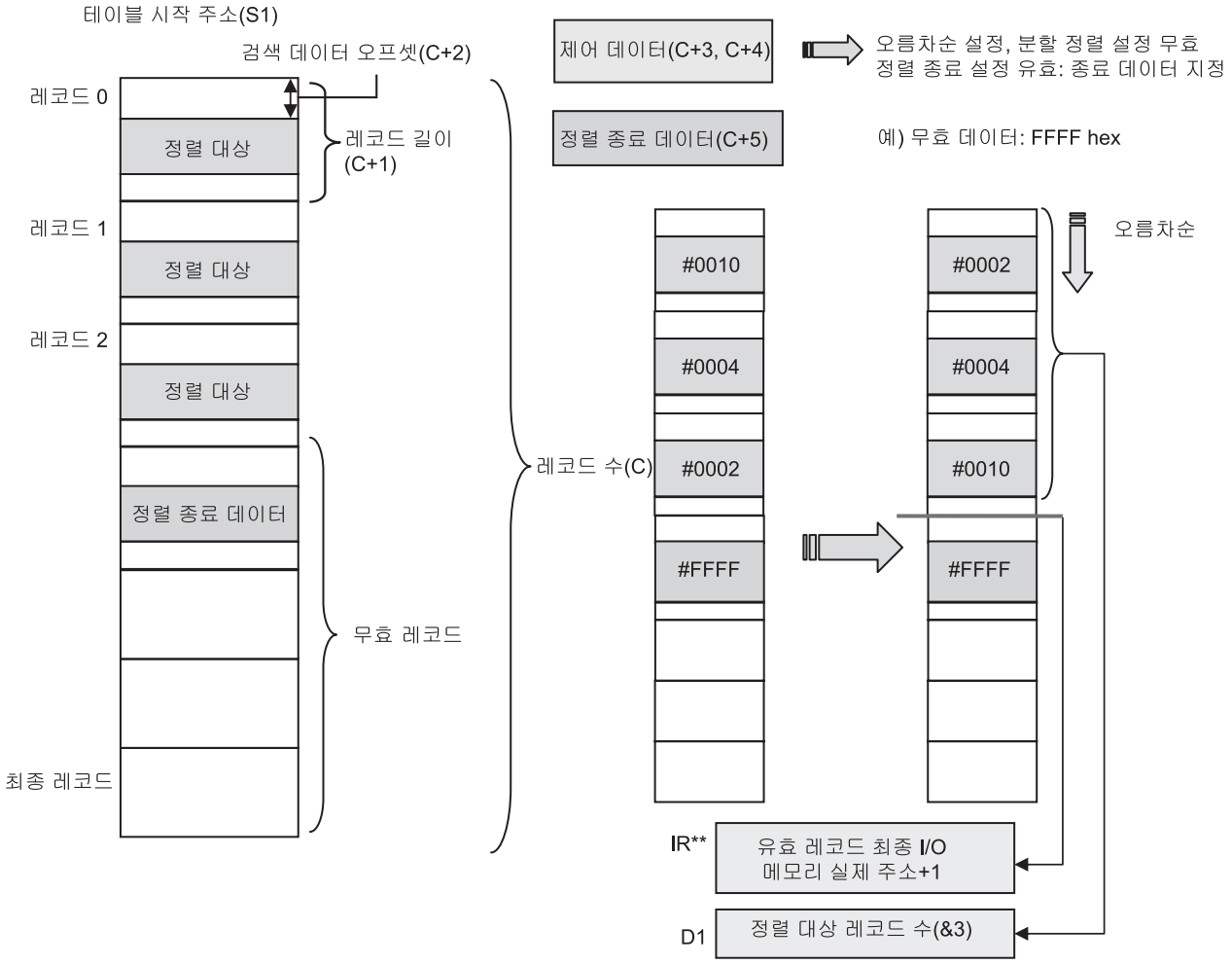
■ 정렬 종료 설정이 무효인 경우

레코드 수(C)를 정렬 범위로 판단하고 레코드 수(C)만큼 정렬합니다.



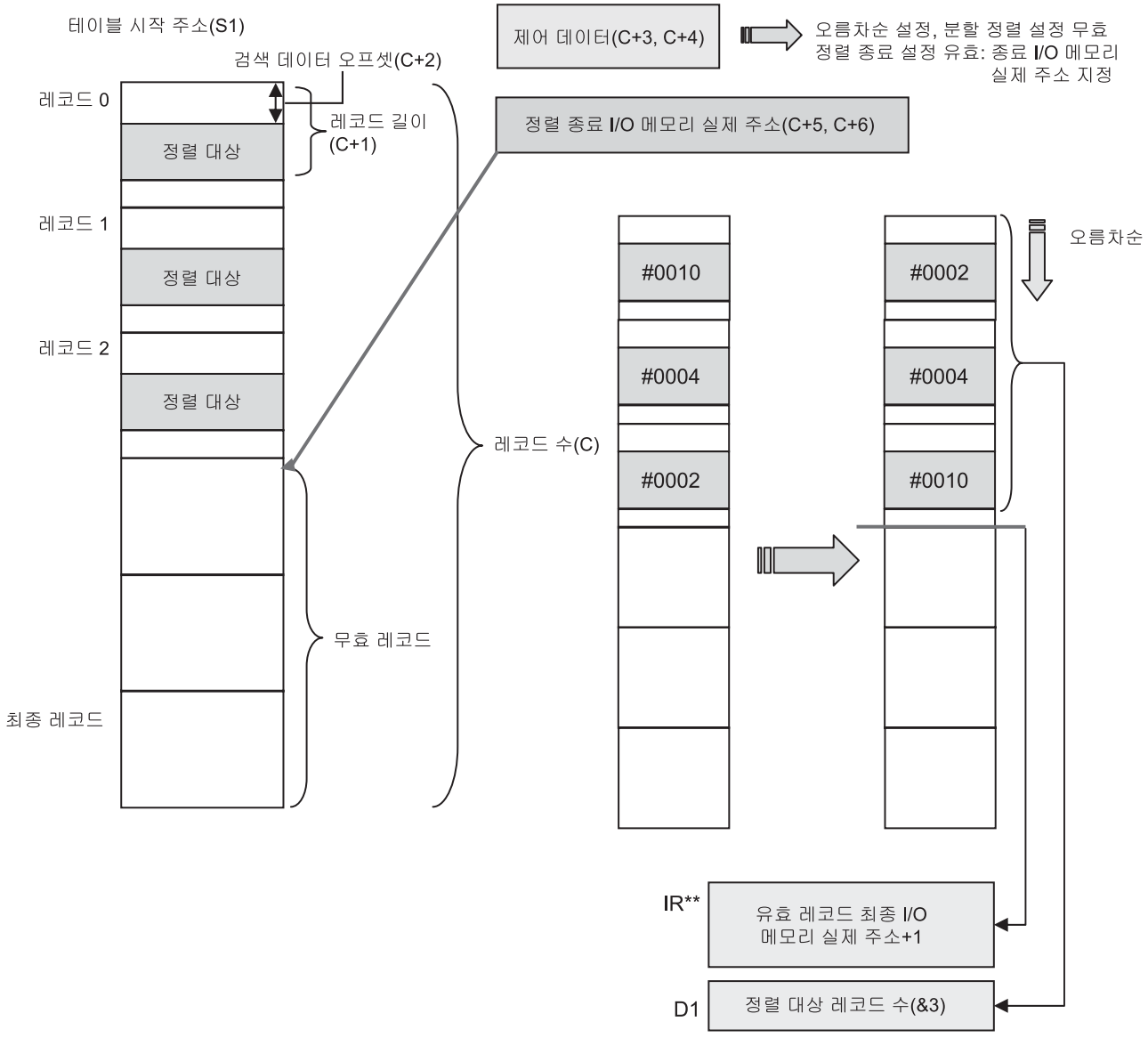
■ 정렬 종료 설정이 유효이고 종료 데이터를 지정하는 경우

C+4의 정렬 종료 설정에서 '유효 종료 데이터 지정'을 지정한 경우, C+4에 있는 정렬 종료 데이터를 찾은 레코드를 정렬 대상 레코드의 끝이라고 판단하고, 종료 레코드 직전의 레코드까지 정렬합니다. 종료 데이터를 찾지 못한 경우는 레코드 수(C)를 정렬 범위로 판단합니다.



■ 정렬 종료 설정이 유효이고 종료 I/O 메모리 실제 주소를 지정하는 경우

C+4의 정렬 종료 설정에서 '유효 종료 I/O 메모리 실제 주소 지정'을 지정한 경우, C+5, C+6으로 나타내는 I/O 메모리 실제 주소까지 정렬 대상으로 합니다. 종료 I/O 메모리 실제 주소 이후의 레코드는 정렬 대상에서 제외됩니다. 종료 I/O 메모리 실제 주소가 최종 레코드 이후에 설정된 경우는 레코드 수(C)를 정렬 범위로 판단합니다.



사용상의 주의 사항

· 정렬을 시작할 때는 반드시 C+7을 클리어하십시오.

RSORT2

명령 이름	니모닉	동작 옵션	펑션 번호	기능
부호 없는 2W 레코드 정렬	RSORT2	@RSORT2	204	지정한 테이블 내의 여러 레코드를 레코드 내 지정 위치의 데이터(2W)로 정렬합니다.

심볼	RSORT2



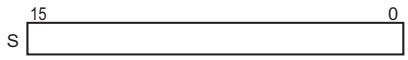
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

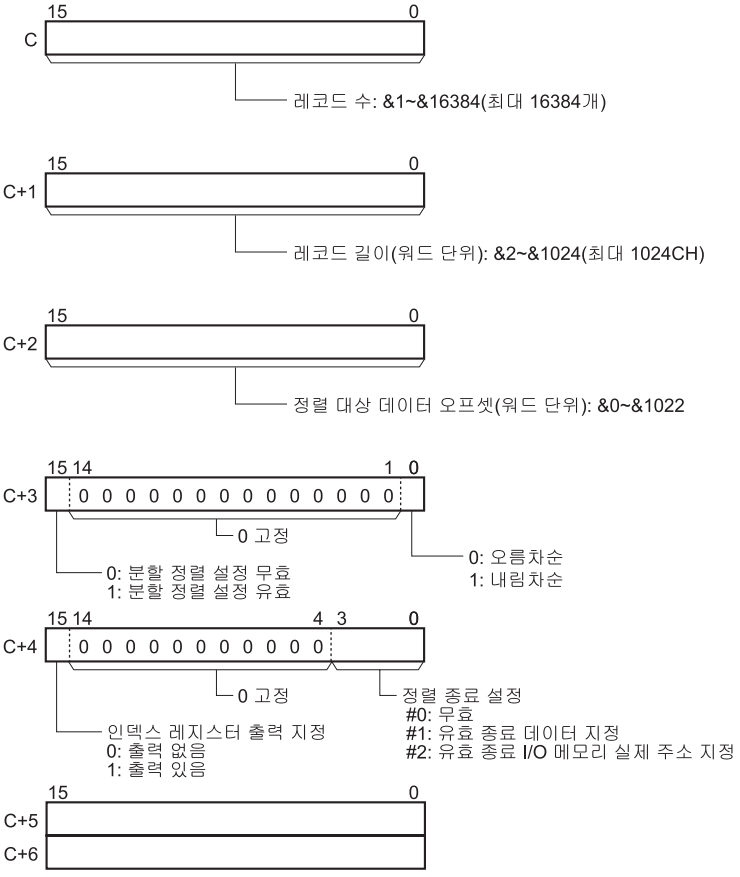
피연산자	내용	데이터 형식	크기
C	제어 데이터	WORD	87
S	정렬 시작 레코드의 시작 CH 번호	WORD	가변
D1	정렬 결과 출력 대상 CH 번호	WORD	1
D2	전송 대상 인덱스 레지스터 번호	DWORD	2

S: 정렬 시작 레코드의 시작 CH 번호



RSORT2

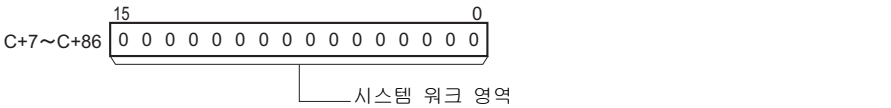
C: 제어 데이터



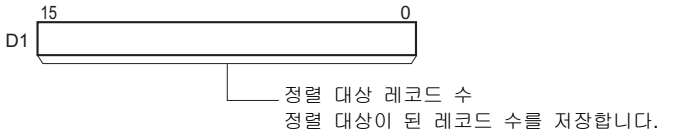
■ 정렬 종료 설정이 무효인 경우
C+5: 0 고정
C+6: 0 고정

■ 정렬 종료 설정이 유효이고 종료 데이터를 지정하는 경우
C+5: 정렬 종료 데이터
(00000000 ~ FFFFFFFF)
C+6: 정렬 종료 데이터
(00000000 ~ FFFFFFFF)

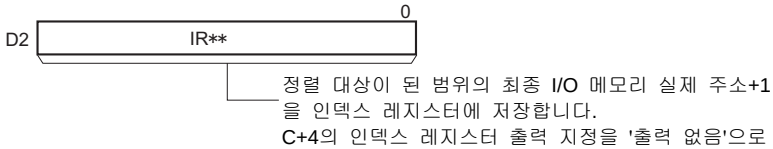
■ 정렬 종료 설정이 유효이고 종료 I/O 메모리 실제 주소를 지정하는 경우
C+5: 정렬 종료 I/O 메모리 실제 주소(하위)
C+6: 정렬 종료 I/O 메모리 실제 주소(상위)



D1: 정렬 결과 출력 대상 CH 번호



D2: 전송 대상 인덱스 레지스터 번호



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S																		
D1												○						
D2	–	–	–	–	–	–	–	–	–	–	○	–	○	–				

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 레코드 길이(C+1)가 2~1024가 아닐 때 ON · 레코드 수(C)×레코드 길이(C+1)가 32768을 초과했을 때 ON · 레코드 수(C)×레코드 길이(C+1)가 0일 때 ON · 정렬 대상 데이터 오프셋(C+2)이 레코드 길이(C+1)보다 2 작은 값을 초과했을 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 정렬 중일 때 ON · 정렬 완료 시 OFF

기능

S로 지정된 시작 레코드부터 C의 지정된 레코드 수까지의 데이터를 지정된 위치의 데이터(2W)를 키로 하여 레코드 단위로 정렬합니다. 정렬이 종료되면 캐리 플래그가 OFF가 되고 정렬 대상 레코드 수를 D1에 출력합니다. 또한 C+4의 인덱스 레지스터 출력 지정을 '출력 있음'으로 설정한 경우는 정렬 대상이 된 범위의 최종 I/O 메모리 실제 주소+1을 IR에 출력합니다. 정렬 방향은 C+3의 설정값에 따라 각각 오름차순/내림차순으로 정렬합니다.

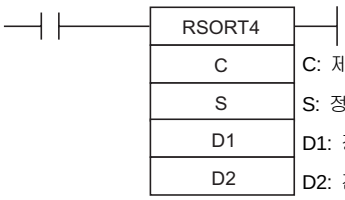
C+3의 분할 정렬 설정을 유효로 한 경우는 여러 사이클에 걸쳐 정렬을 실행합니다. 정렬 실행 중에는 캐리 플래그가 ON되고 정렬을 완료하면 OFF됩니다. 정렬 완료 전에 RSORT2 명령 실행 조건이 OFF가 된 경우는 다음에 실행 조건이 ON이 되면 이어서 정렬을 실행합니다.

사용상의 주의 사항

· 정렬을 시작할 때는 반드시 C+7을 클리어하십시오.

RSORT4

명령 이름	니모닉	동작 옵션	평션 번호	기능
부호 없는 4W 레코드 정렬	RSORT4	@RSORT4	205	지정한 테이블 내의 여러 레코드를 레코드 내 지정 위치의 데이터(4W)로 정렬합니다.

심볼	RSORT4
	

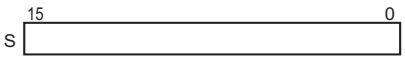
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

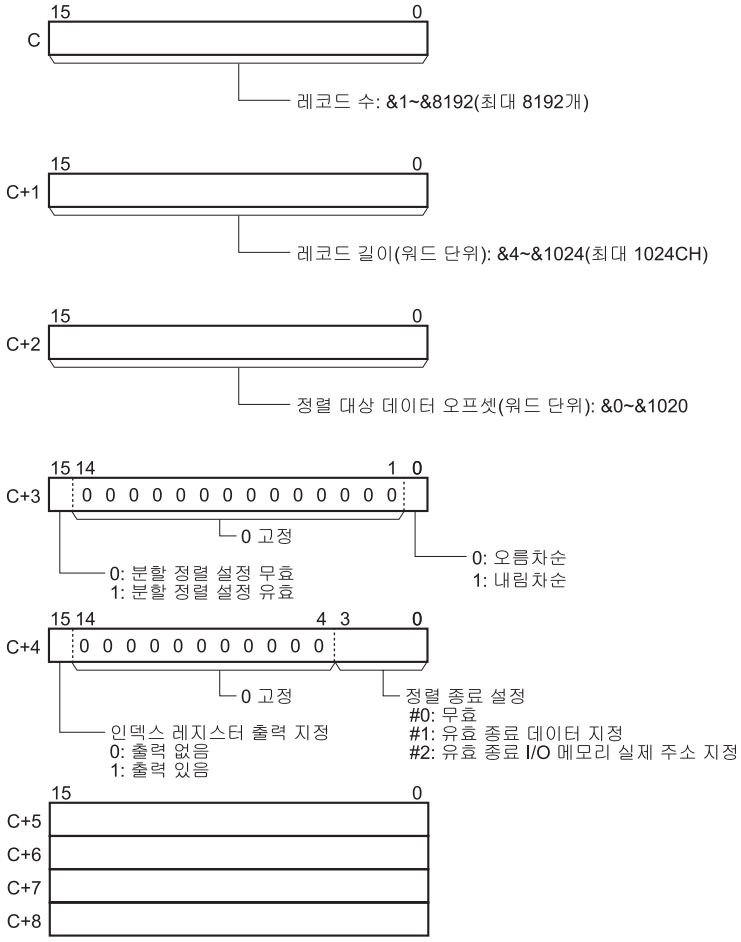
피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	WORD	89
S	정렬 시작 레코드의 시작 CH 번호	WORD	가변
D1	정렬 결과 출력 대상 CH 번호	WORD	1
D2	전송 원본 인덱스 레지스터 번호	DWORD	2

S: 정렬 시작 레코드의 시작 CH 번호



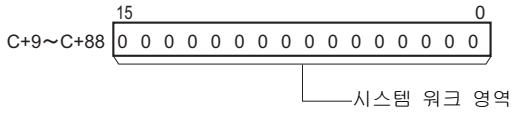
C: 제어 데이터



■ 정렬 종료 설정이 무효인 경우
C+5: 0 고정
C+6: 0 고정

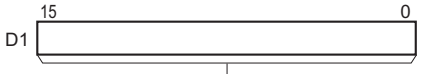
■ 정렬 종료 설정이 유효이고 종료 데이터를 지정하는 경우
C+5~C+88: 정렬 종료 데이터
(0000000000000000 ~ FFFFFFFFFFFFFFFF)
C+5에서 최하위 16비트→하위 16비트→상위 16비트→최상위 16비트 순서로 저장합니다.

■ 정렬 종료 설정이 유효이고 종료 I/O 메모리 실제 주소를 지정하는 경우
C+5: 정렬 종료 I/O 메모리 실제 주소(하위)
C+6: 정렬 종료 I/O 메모리 실제 주소(상위)
C+7: 0 고정
C+8: 0 고정



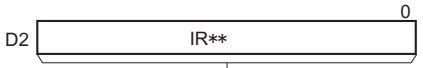
시스템 워크 영역
정렬을 시작할 때는 C+7을 반드시 0으로 클리어하십시오.

D1: 정렬 결과 출력 대상 CH 번호



정렬 대상 레코드 수
정렬 대상이 된 레코드 수를 저장합니다.

D2: 전송 대상 인덱스 레지스터 번호



정렬 대상이 된 범위의 최종 I/O 메모리 실제 주소+1을
인덱스 레지스터에 저장합니다.
C+4의 인덱스 레지스터 출력 지정을 '출력 없음'으로
설정한 경우는 D2에 #00000000을 지정하십시오.

RSORT4

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S																		
D1												○						
D2	–	–	–	–	–	–	–	–	–	–	○	–	○	–				

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 레코드 길이(C+1)가 4~1024가 아닐 때 ON · 레코드 수(C)×레코드 길이(C+1)가 32768을 초과했을 때 ON · 레코드 수(C)×레코드 길이(C+1)가 0일 때 ON · 정렬 대상 데이터 오프셋(C+2)이 레코드 길이(C+1)보다 4 작은 값을 초과했을 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 정렬 중일 때 ON · 정렬 완료 시 OFF

기능

S로 지정된 시작 레코드부터 C의 지정된 레코드 수까지의 데이터를 지정된 위치의 데이터(4W)를 키로 하여 레코드 단위로 정렬합니다. 정렬이 종료되면 캐리 플래그가 OFF가 되고 정렬 대상 레코드 수를 D1에 출력합니다. 또한 C+4의 인덱스 레지스터 출력 지정을 '출력 있음'으로 설정한 경우는 정렬 대상이 된 범위의 최종 I/O 메모리 실제 주소+1을 IR에 출력합니다. 정렬 방향은 C+3의 설정값에 따라 각각 오름차순/내림차순으로 정렬합니다.

C+3의 분할 정렬 설정을 유효로 한 경우는 여러 사이클에 걸쳐 정렬을 실행합니다. 정렬 실행 중에는 캐리 플래그가 ON되고 정렬을 완료하면 OFF됩니다. 정렬 완료 전에 RSORT4 명령 실행 조건이 OFF가 된 경우는 다음에 실행 조건이 ON이 되면 이어서 정렬을 실행합니다.

사용상의 주의 사항

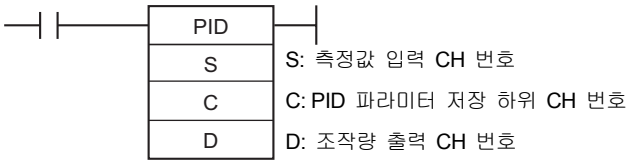
· 정렬을 시작할 때는 반드시 C+7을 클리어하십시오.

데이터 제어 명령

니모닉	명령 이름	펑션 번호	페이지
PID	PID 연산	190	3-518
PIDAT	자동 조정 가능한 PID 연산	191	3-527
LMT	상하한 리미트 제어	680	3-534
BAND	데드 밴드 제어	681	3-536
ZONE	데드 존 제어	682	3-538
TPO	시분할 비례 출력	685	3-540
SCL	스케일링	194	3-546
SCL2	스케일링 2	486	3-550
SCL3	스케일링 3	487	3-553
AVG	데이터 평균화	195	3-556

PID

명령 이름	니모닉	동작 옵션	펑션 번호	기능
PID 연산	PID	–	190	지정된 파라미터에 따라 PID 연산(목표값 필터형 2 자유도 PID)을 수행하여 조작량을 출력합니다.

심볼	PID
	

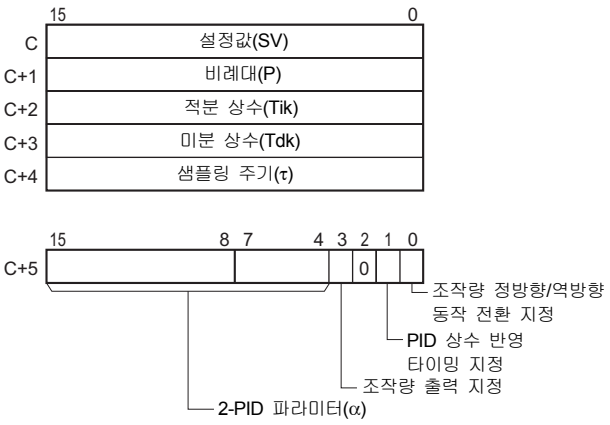
특정 영역 내의 사용 여부

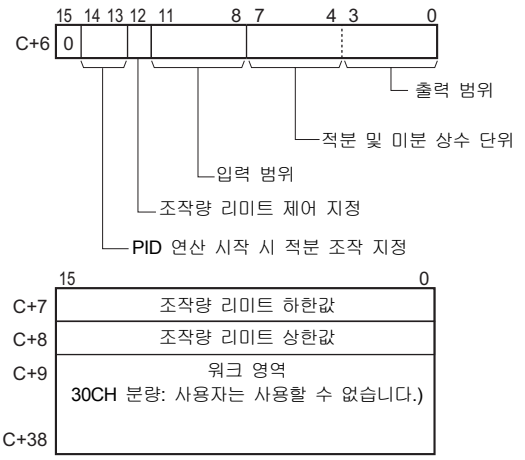
영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	측정값 입력 CH 번호	UINT	1
C	PID 파라미터 저장 하위 CH 번호	WORD	39
D	조작량 출력 CH 번호	UINT	1

C: PID 파라미터 저장 하위 CH 번호





■ 영역 종류

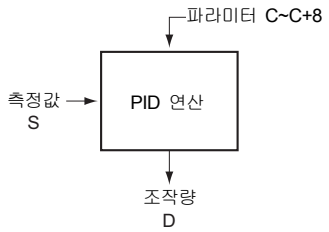
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
C												–						
D												○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C의 데이터가 범위 밖일 때 ON · 실제 샘플링 주기가 설정된 샘플링 주기의 2배를 초과했을 때 ON · 그 이외일 때 OFF
> 플래그	P_GT	· PID 연산된 조작량이 조작 출력 리미트 상한값을 상회할 때 ON · 그 이외일 때 OFF
< 플래그	P_LT	· PID 연산된 조작량이 조작 출력 리미트 하한값을 하회할 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· PID 연산을 실행했을 때 ON · 그 이외일 때 OFF

기능

C로 지정한 파라미터(설정값, PID 상수 등)에 따라 S를 측정값 입력으로 한 PID 연산(목표값 필터형 2 자유도 PID 연산)을 수행하여 D에 조작량을 출력합니다.



입력 조건이 기동(OFF→ON)될 때 파라미터를 전송하여 정상 범위 밖이면 에러 플래그를 ON으로 합니다.

정상 범위이면 그 때의 조작량을 초기값으로 PID 처리를 수행합니다.

입력 조건이 ON이면 지정된 샘플링 주기마다의 측정값을 입력으로 하여 연산을 수행합니다.

입력 조건이 OFF이면 PID 연산을 정지합니다. D의 조작량은 그 때의 값으로 유지됩니다. 변경해야 할 때는 래더 프로그램 또는 수동 조작으로 변경하십시오.

· 입력 범위(C+6의 비트 8~11) 지정을 통해 S의 측정값으로 입력된 16비트 중 유효로 할 비트 수를 지정합니다. 예를 들어, 입력 범위를 12비트(#4)로 지정하면 16진수 #0000~0FFF 범위의 값이 측정값으로 유효하게 됩니다. 입력 범위로 지정한 비트보다 큰 값이 입력되었을 때는 입력 범위의 최대값으로 보정합니다. 이 때, 에러 플래그는 ON되지 않습니다.

· 설정값의 범위도 입력 범위에 따릅니다.

· 측정값 및 설정값은 #0~입력 범위 최대값의 부호 없는 BIN 값(16진수)입니다.

· 출력 범위(C+6의 비트 0~3) 지정을 통해 조작량의 16비트 중 유효로 할 비트 수를 지정합니다. 예를 들어, 출력 범위를 12비트(#4)로 지정하면 16진수 #0000~0FFF 범위의 값이 조작량으로 출력됩니다.

· 비례 동작만인 경우 측정값 = 설정값일 때의 조작량을 지정할 수 있습니다.

  0: 출력 0%

  1: 출력 50%

· 비례 동작의 정방향 동작, 역방향 동작을 지정할 수 있습니다.

· 조작량의 상하한을 지정할 수 있습니다.

· 샘플링 주기는 10ms 단위로 지정할 수 있습니다(0.01~99.99s). 단, 샘플링 주기와 PID 명령 실행 시(사이클 시간마다)의 조합으로 실제 PID 연산을 실행할지 여부가 결정됩니다.

· PID 연산 실행 중에 비례대(P), 적분 상수(Tik), 미분 상수(Tdk)를 변경한 경우, 해당 변경 사항을 샘플링 주기별로 PID 연산에 반영/반영하지 않음 중 하나를 지정할 수 있습니다. PID 상수 반영 타이밍을 지정(C+5의 비트 1)하여 수행합니다.

주: PID 연산 실행 중에 변경 사항을 반영할 수 있는 것은 P, Tik, Tdk의 각 파라미터뿐입니다.

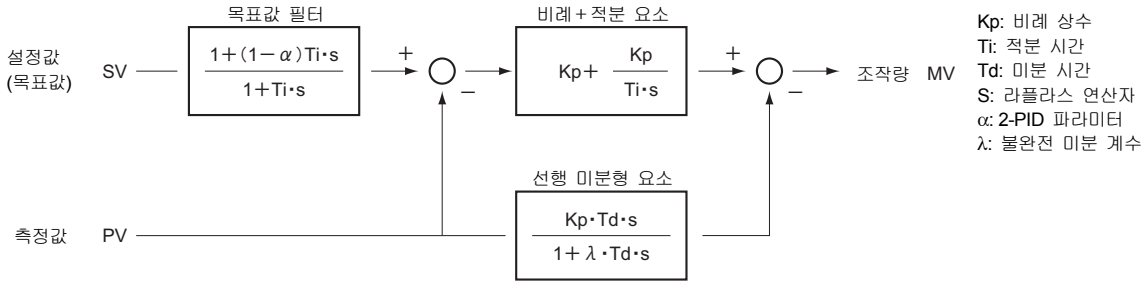
성능 사양

항목			사양
PID 연산 방식		–	목표값 필터형 2 자유형 PID 방식(정방향 동작/역방향 동작)
PID 제어 루프 수		–	제한 없음(1명령 1루프)
샘플링 제어		τ	0.01~99.99s
PID 상수	비례대	P	0.1~999.9%
	적분 상수	Tik	1~8191, 9999(샘플링 주기의 배수, 9999일 때 적분 동작 없음)
	미분 상수	Tdk	0~8191(샘플링 주기의 배수, 0일 때 미분 동작 없음)
설정 수		SV	0~65535(단, 입력 범위 최대값까지 유효)
설정값		PV	0~65535(단, 입력 범위 최대값까지 유효)
조작값		MV	0~65535(단, 출력 범위 최대값까지 유효)

연산 방식

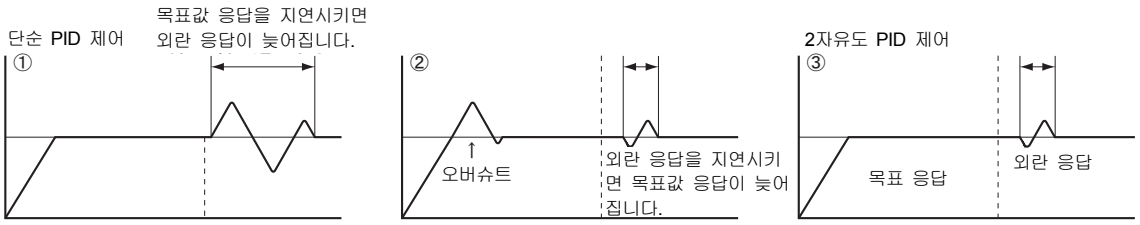
PID 제어의 연산은 목표값 필터형 2 자유도 PID 제어 방식으로 수행됩니다.

목표값 필터형 2 자유도 PID 블록도



· 단순 PID 제어에서는 오버슈트가 나타나는 것을 방지하면 외란에 대한 안정이 늦어지고(①), 반대로 외란에 대한 안정을 빠르게 하면 오버슈트가 나타나 목표값에 대한 응답이 늦어집니다(②).

그에 비해 2 자유도 PID 제어의 경우는 오버슈트가 없고 목표값에 대한 응답도 빠르며 외란에 대한 안정도 빠르게 수행할 수 있습니다(③).



PID 파라미터 설정 내용

제어 데이터	항목	내용	설정 범위	입력 조건 ON인 상태에서 변경 가능 여부
C	설정값(SV)	제어 대상의 목표값입니다.	입력 범위에서 비트 수 분량의 BIN 데이터 (0~지정된 입력 범위 최대값)	가능
C+1	비례대(P)	비례 제어 범위/제어 범위 전체로 나타내는 P 제어용 파라미터입니다.	10진수 &1~9999 또는 16진수 #0001~270F(0.1% 단위, 0.1~999.9%)	입력 조건 ON인 경우 C+5의 비트 1이 1일 때 가능
C+2	적분 상수(Tik)	적분 동작 효과의 크기를 정확히 나타내는 상수입니다. 이 값을 크게 하면 적분 효과가 약해집니다.	10진수 &1~8191 또는 16진수 #0001~1FFF 10진수 &9999 또는 16진수 #270F(적분 동작 없음 설정) 적분 및 미분 상수 단위 지정이 "1"일 때: 1~8191배 "9"일 때: 0.1~819.1s(주 1)	
C+3	미분 상수 (Tdk)	미분 동작의 크기를 나타내는 상수입니다. 이 값을 크게 하면 미분 효과가 강해집니다.	10진수 &1~8191 또는 16진수 #0001~1FFF 10진수 &0 또는 16진수 #0000(미분 동작 없음 설정) 적분 및 미분 상수 단위 지정이 "1"일 때: 1~8191배 "9"일 때: 0.1~819.1s(주 1)	
C+4	샘플링 주기 (τ)	PID 연산을 수행하는 주기를 설정합니다.	10진수 &1~9999 또는 16진수 #0001~270F(10ms 단위, 0.01~99.99s)	불가능
C+5의 비트 4~15	2-PID 파라미터(α)	입력 필터 계수입니다. 일반적으로 0.65로 사용하십시오. 값이 0에 가까울수록 필터 성능이 떨어집니다.	16진수 #000: α = 0.65(16진수 3자리) 16진수 #100~163으로 하면 하위 2자리의 값이 α = 0.00~0.99를 의미합니다. (주 2)	
C+5의 비트 3	조작량 출력 지정	측정값 = 설정값일 때의 조작량을 지정합니다. 주: 이 설정은 '적분 동작 없음'일 때 유효합니다.	0, 1 0: 출력 0% 1: 출력 50%	
C+5의 비트 1	PID 상수 반영 타이밍 지정	P(비례대), Tik(적분 상수), Tdk(미분 상수)의 각 파라미터를 언제 PID 연산에 반영시킬 지를 지정합니다.	0, 1 0: 입력 조건 기동 시에만 1: 입력 조건 기동 시 및 샘플링 주기마다	가능
C+5의 비트 0	조작량 정/역동작 전환 지정	비례 동작의 방향을 결정하는 파라미터입니다.	0, 1 0: 역방향 동작 1: 정방향 동작	불가능
C+6 비트 14, 13	PID 연산 시작 시 적분 조작 지정 (유니트 Ver.4.0 이상만)	PID 연산 시작(입력 OFF→ON) 시의 적분 조작량의 초기 상태를 지정합니다.	비트 14 = 0, 비트 13 = 0: 연산을 시작할 때의 조작량 출력 지정과 동일한 적분 조작량부터 시작합니다(기존과 동일). 비트 14 = 0 또는 1, 비트 13 = 1: 조작량 출력이 갑자기 바뀌지 않고 연속적으로 변화하는 적분 조작량부터 시작합니다(범프리스 처리). 비트 14 = 1, 비트 13 = 0: 적분 조작량=0부터 시작합니다.	불가능
C+6의 비트 12	조작량 리미트 제어 지정	조작량을 리미트 제어할지 여부를 지정입니다.	0, 1 0: 무효(리미트 제어 안 함) 1: 유효(리미트 제어) (주 3)	
C+6의 비트 8~11	입력 범위	입력 데이터의 비트 수입니다.	(16진수 1자리) 0: 8비트, 1: 9비트, 2: 10비트 3: 11비트, 4: 12비트, 5: 13비트 6: 14비트, 7: 15비트, 8: 16비트	
C+6의 비트 4~7	적분 및 미분 상수 단위 지정	적분 상수 및 미분 상수의 시간 단위 지정입니다.	9(16진수 1자리) 1: 샘플링 주기 배수 지정 적분 및 미분 시간을 샘플링 주기의 지정 배수 시간으로 지정합니다. 9: 시간 지정 적분 및 미분 시간을 100ms 단위로 지정합니다. (주 1)	

PID

제어 데이터	항목	내용	설정 범위	입력 조건 ON인 상태에서 변경 가능 여부
C+6의 비트 0~3	출력 범위	출력 데이터의 비트 수입니다.	설정 범위는 입력 범위와 동일합니다.	
C+7	조작량 리미트 하한값	조작량을 리미트 제어했을 때의 리미트 하한값입니다.	10진수 &0~65535 또는 16진수 #0000~FFFF(주 3)	
C+8	조작량 리미트 상한값	조작량을 리미트 제어했을 때의 리미트 상한값입니다.	10진수 &0~65535 또는 16진수 #0000~FFFF(주 3)	

주 1: 적분 및 미분 상수 단위 지정을 시간 지정("9")으로 한 경우는 적분 시간 및 미분 시간 설정을 샘플링 주기의 1~8191배에 해당하는 시간 이내로 설정하십시오.

주 2: 000을 설정하면 2-PID 파라미터 α가 0.65가 됩니다. 평상시에는 000으로 설정하십시오.

주 3: 조작량 리미트 제어 지정을 유효("1")로 한 경우에는 각 값을 다음과 같이 설정하십시오.
0000≦조작량 리미트 하한값≦조작량 리미트 상한값≦지정 출력 범위 최대값

샘플링 주기와 사이클 시간의 관계

샘플링 주기는 10ms 단위로 지정할 수 있습니다(0.01~99.99s). 단, 샘플링 주기와 PID 명령 실행 시(사이클 시간마다)의 조합에 의해 실제 PID 연산이 결정됩니다. 샘플링 주기와 사이클 시간의 관계는 다음과 같습니다.

1) 샘플링 주기＜사이클 시간일 때

샘플링 주기마다 PID 연산을 하지 않고 사이클 시간마다 PID 연산을 합니다.

2) 샘플링 주기≧사이클 시간일 때

사이클마다 PID 연산을 실행하지 않고 사이클 시간(PID 명령-PID 명령 간의 시간)의 누적값≧샘플링 주기가 되었을 때 PID 명령이 실행됩니다. 누적값의 초과분(사이클 시간의 누적값－샘플링 주기)은 다음 누적값으로 넘어갑니다.

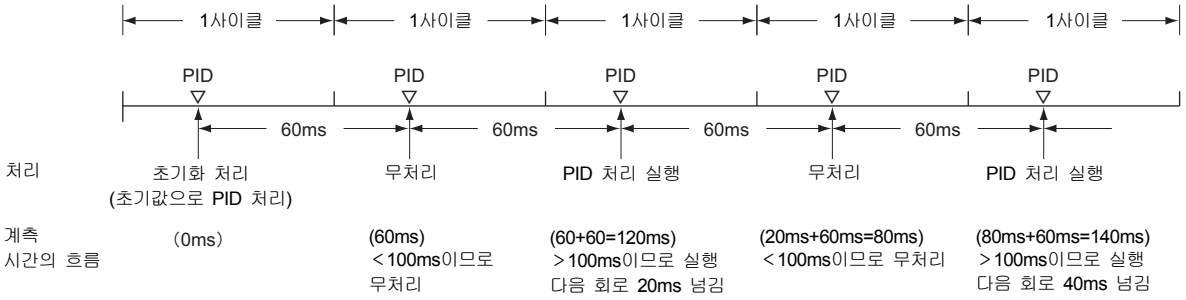
예를 들어, 샘플링 주기 100ms에서 사이클 시간이 60ms로 일정한 경우,

초기화 실행 후의 1회째는 60ms＜100ms이므로 PID 명령이 실행되지 않습니다.

2회째는 60ms+60ms＞100ms가 되므로 PID 명령을 실행합니다. (초과분인 120ms－100ms＝20ms는 다음 회로 넘어갑니다.)

3회째는 이전 회에서 넘어온 분량을 더하여 20ms+60ms로 계산되며, 80ms＜100ms이므로 PID 명령은 실행되지 않습니다.

4회째는 80ms+60ms＞100ms이므로 PID 명령을 실행합니다. (초과분인 140ms－100ms＝40ms는 다음 회로 넘어갑니다.) 이하 동일합니다.



PID 제어

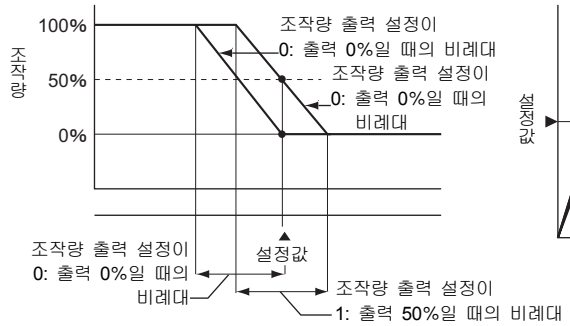
■ 비례 동작(P)

설정값에 대해 비례대가 있으며, 그 중에서 조작량(제어량)이 편차에 비례하는 동작을 비례 동작이라고 합니다.

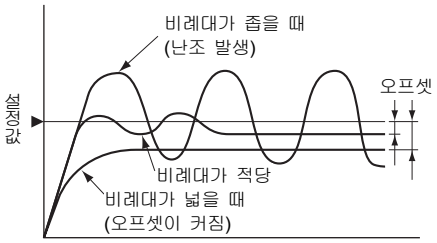
현재값이 비례대보다 작은 경우 조작량은 100%입니다. 비례대 내에서는 조작량이 편차에 비례하여 서서히 작아지고, 설정값과 현재값이 일치하면(편차가 0이 되는 경우) 조작량은 조작량 출력 지정 파라미터에 따라 0% 또는 50%가 됩니다(역방향 동작의 경우).

비례대는 입력 범위 전체 폭에 대한 백분율로 나타냅니다. 비례 동작에서는 오프셋(잔류 편차)이 발생하지만 비례대를 작게 하여 오프셋을 감소시킬 수 있습니다. 단, 너무 작으면 난조를 보입니다.

비례 동작(역방향 동작 시)



비례대 조정



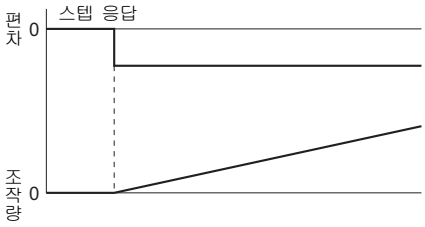
■ 적분 동작(I)

비례 동작에서는 오프셋이 발생합니다. 따라서 비례 동작에 적분 동작을 조합하면 시간 경과에 따라 오프셋이 감소하여 측정값과 설정값이 일치하게 됩니다.

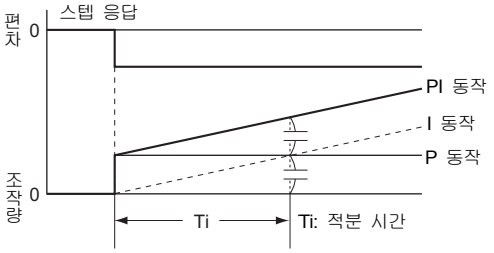
· 적분 시간

적분 동작의 강도를 나타내는 단위로, 그림과 같이 스텝상의 편차에 대해 적분의 조작량이 비례 동작과 동일한 조작량에 도달할 때까지의 시간을 말합니다. 적분 시간이 짧을수록 적분 동작에 의한 정정이 강해집니다. 단, 너무 짧으면 정정 동작이 너무 강해져서 난조가 발생하는 요인이 될 수 있습니다.

적분 동작



PI 동작과 적분 시간



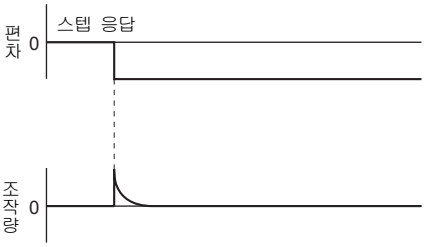
■ 미분 동작(D)

비례 동작이나 적분 동작은 제어 결과에 대한 정정 동작이므로 항상 응답이 지연됩니다. 미분 동작은 이런 결점을 보완하기 위해 급격한 외란에 대해 큰 조작량을 부여하여 원래 상태로 빨리 돌아가도록 작동합니다. 편차가 생기는 경사(미분 계수)에 비례한 조작량으로 정정 동작을 수행합니다.

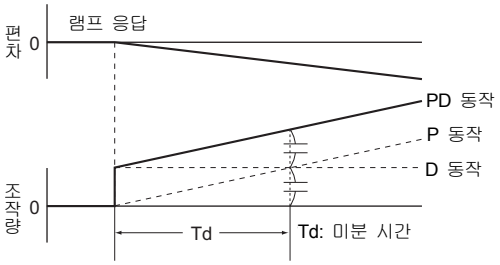
· 미분 시간

미분 동작의 강도를 나타내는 단위로, 그림과 같이 램프상의 편차에 대해 미분의 조작량이 비례 동작과 동일한 조작량에 도달할 때까지의 시간을 말합니다. 미분 시간이 길수록 미분 동작에 의한 정정이 강해집니다.

미분 시간



PD 동작과 미분 시간



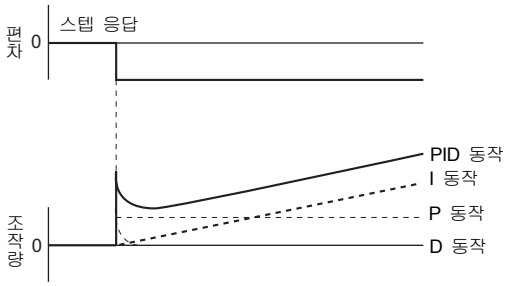
PID 동작

PID 동작은 비례 동작(P), 적분 동작(I) 및 미분 동작(D)을 조합한 동작으로, 낭비되는 시간이 있는 제어 대상에도 우수한 제어 결과를 얻습니다. 비례 동작으로 난조가 발생하지 않는 매끄러운 제어를 실행하고 적분 동작으로 오프셋을 자동으로 수정하며, 미분 동작으로 외란에 대한 응답을 빠르게 합니다.

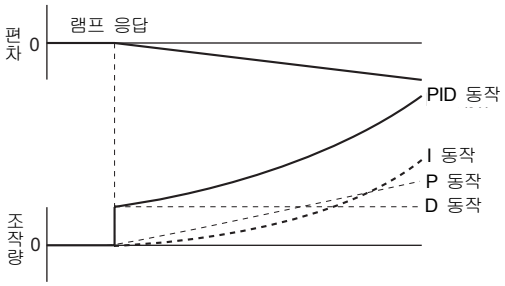
스텝상 편차, 램프상 편차에 대한 PID 동작의 조작량 변화는 다음과 같습니다.

PID

PID 동작의 스텝 응답



PD 동작의 램프 응답

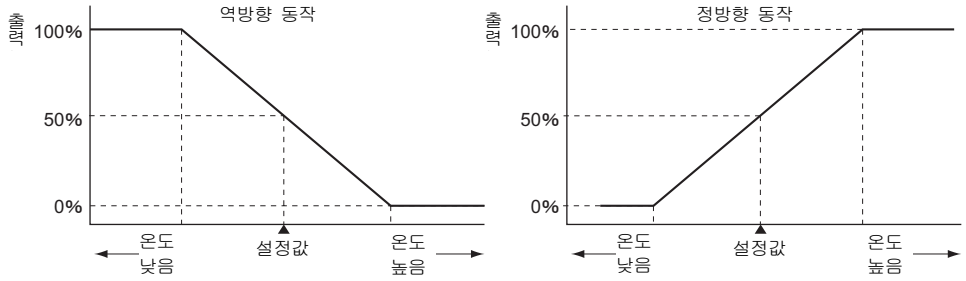


동작 방향

PID 제어에서는 다음 2가지 중 하나의 제어 방향을 선택합니다.

역방향 동작: 설정값보다 측정값이 작으면 조작량을 증가시킵니다(가열).

정방향 동작: 설정값보다 측정값이 크면 조작량을 증가시킵니다(냉각).

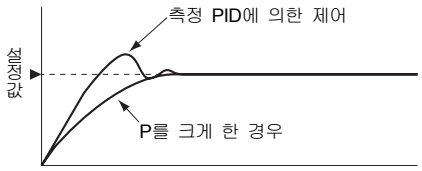


(조작량 출력 지정 50%일 때)

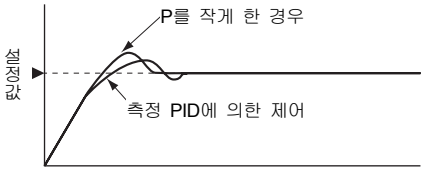
PID 파라미터 조정

PID 파라미터와 제어 상태의 일반적인 관계는 다음과 같습니다. PID 명령을 사용할 때 참고하십시오.

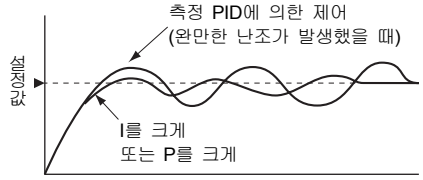
- 안정될 때까지 시간(정정 시간)이 좀 걸려도 오버 슈트가 발생하지 않도록 하려면 비례대(P)를 크게 설정합니다.



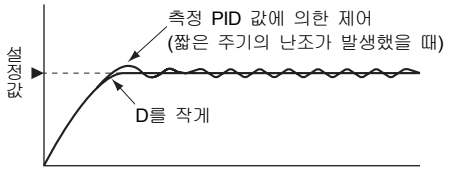
- 오버슈트가 발생해도 빠르게 안정적인 제어 상태로 만들고 싶으면 비례대(P)를 작게 설정합니다. 단, 너무 작으면 난조가 발생하는 원인이 됩니다.



- 완만한 난조가 발생하거나 오버슈트 및 언더 슈트를 반복하면서 수렴할 때는 적분 동작이 너무 강한 것일 수 있습니다. 적분 시간(I)을 크게 하거나 비례대(P)를 크게 하면 난조를 줄일 수 있습니다.



- 짧은 주기의 난조가 발생했을 때는 제어계의 응답이 빠르고 미분 동작이 너무 강한 것일 수 있습니다. 이런 경우는 미분 시간(D)을 작게 설정하십시오.



힌트

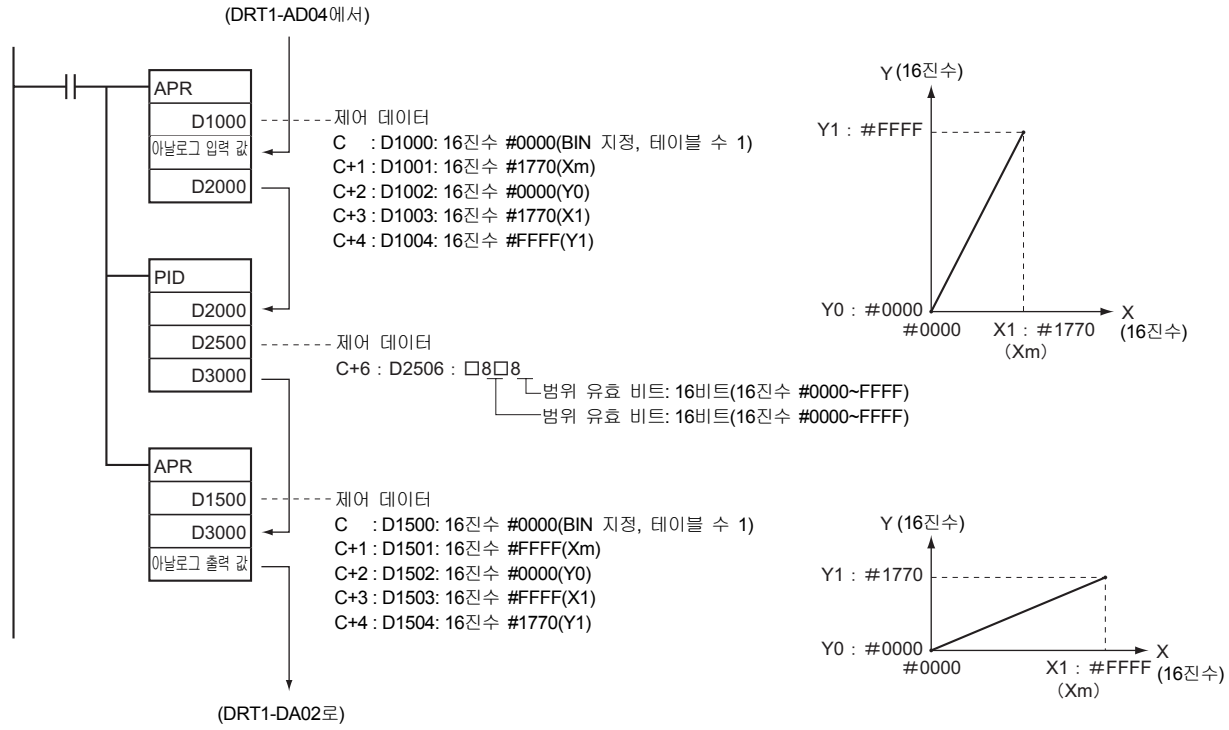
- PID 상수 반영 타이밍 지정(C+5의 비트 1)은 로트 번호 001201□□□□(2000년 12월 1일 이후 제조분)의 CS1 CPU 유니트, CS1-H CPU 유니트 및 CJ 시리즈에서 사용할 수 있습니다.
- PID 연산 명령의 측정값 및 조작 출력량의 범위는 제어 데이터의 C+6의 비트 8~11 및 C+6의 비트 0~3이며, 16비트 내 유효 비트 수로 지정합니다(다음 표 참조).

C+6의 비트 8~11 C+6의 비트 0~3	유효 비트 수	데이터 범위
0	8	16진수 #0000~00FF
1	9	16진수 #0000~01FF
2	10	16진수 #0000~03FF
3	11	16진수 #0000~07FF
4	12	16진수 #0000~0FFF
5	13	16진수 #0000~1FFF
6	14	16진수 #0000~3FFF
7	15	16진수 #0000~7FFF
8	16	16진수 #0000~FFFF

사용하는 아날로그 입력 유니트 및 아날로그 출력 유니트의 데이터 범위가 이 유효 비트 수로 지정 가능한 데이터 범위에 없는 경우, 다음과 같이 PID 연산 명령의 앞과 뒤에서 APR(숫자 변환) 명령을 사용하여 데이터 범위가 일치하도록 숫자를 변환하십시오.

예: CompoBus/D 슬레이브의 아날로그 입력 유니트 DRT1-AD04, 아날로그 출력 유니트 DRT1-DA02를 사용할 때

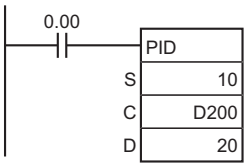
DRT1-AD04, DRT1-DA02의 데이터 범위는 모두 16진수 #0000~1770이므로 이 데이터 범위를 유효 비트 수로 지정할 수 없습니다. 따라서 다음과 같이 APR(숫자 변환) 명령을 사용하여 예를 들어, 아날로그 입력 유니트의 명령을 APR 명령으로 16진수 #0000~FFFF로 변환하여 PID 연산 명령의 측정값으로 출력합니다. 또한 PID 연산 명령의 조작 출력량을 APR 명령으로 16진수 #0000~1770으로 변환하여 아날로그 출력 유니트에 출력합니다.



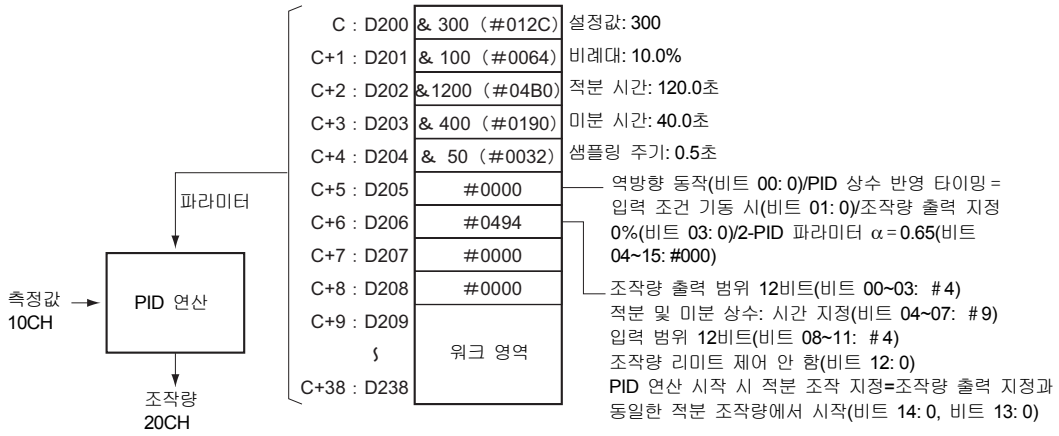
사용상의 주의 사항

- IL-ILC 명령 간, JMP-JME 명령 간 또는 서브루틴 명령 내에서 PID 명령을 사용하는 경우, 입력 조건에 따라 동작이 안정되지 않는 경우가 있으므로 주의하십시오.
- PID 파라미터(C~C+38) 중 입력 조건을 ON으로 한 상태에서 변경할 수 있는 값은 C의 설정값(SV)뿐입니다. 그 밖의 값을 변경했을 때는 반드시 입력 조건을 OFF→ON으로 기동하십시오.
- PID 명령은 입력 조건의 기동을 STOP→RUN으로 처리하여 실행됩니다. 입력 조건이 기동될 때 C+9~C+38을 초기화(클리어)한 후, 다음 사이클 이후 입력 조건이 ON된 상태이면 PID 연산을 실행합니다.
  따라서 상시 ON 플래그(ON)를 PID 명령의 입력 조건으로 하는 경우는 C+9~C+38을 운전 시작 시 초기화(클리어)하는 처리를 별도로 설정하십시오.
- 복수의 PID 명령에서 PID 파라미터 저장 CH를 공유하여 사용할 수 없습니다. 복수의 PID에서 같은 파라미터를 사용하는 경우에도 반드시 다른 CH를 지정하십시오.

샘플 프로그램



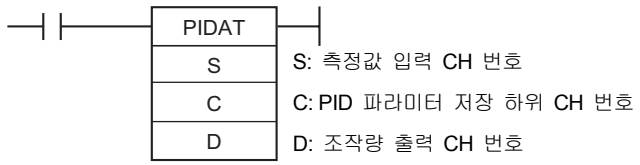
- 0.00이 기동(OFF→ON)되었을 때 D200~D208에 설정된 다음 파라미터에 따라 D209~D238의 워크 영역의 초기화(클리어)를 수행합니다. 초기화 종료 후 PID 연산을 수행하여 조작량을 20CH에 출력합니다.
- 0.00이 ON이면 D200~D208에 설정된 파라미터에 따라 PID 연산을 샘플링 주기 간격으로 실행하여 조작량을 20CH에 출력합니다.
- 비례대(P), 적분 상수(Tik), 미분 상수(Tdk)의 각 파라미터를 0.00의 기동 이후에 변경한 경우, PID 연산에는 반영되지 않습니다.



주: 0.00이 OFF일 때, 이 상태에서 20CH에 값을 쓰면 수동 제어와 같은 동작을 하도록 할 수 있습니다. 이 수동 제어 상태에서 PID 명령을 실행하여 자동 제어로 이동할 때 조작량의 갑작스런 변화를 제어합니다. (이동 시의 조작량을 이전 적분 조작량으로 하여 PID 연산을 시작합니다.)

PIDAT

명령 이름	니모닉	동작 옵션	펑션 번호	기능
자동 조정 가능한 PID 연산	PIDAT	–	191	지정된 파라미터에 따라 PID 연산을 수행합니다.

심볼	자동 조정 가능한 PID 연산
	

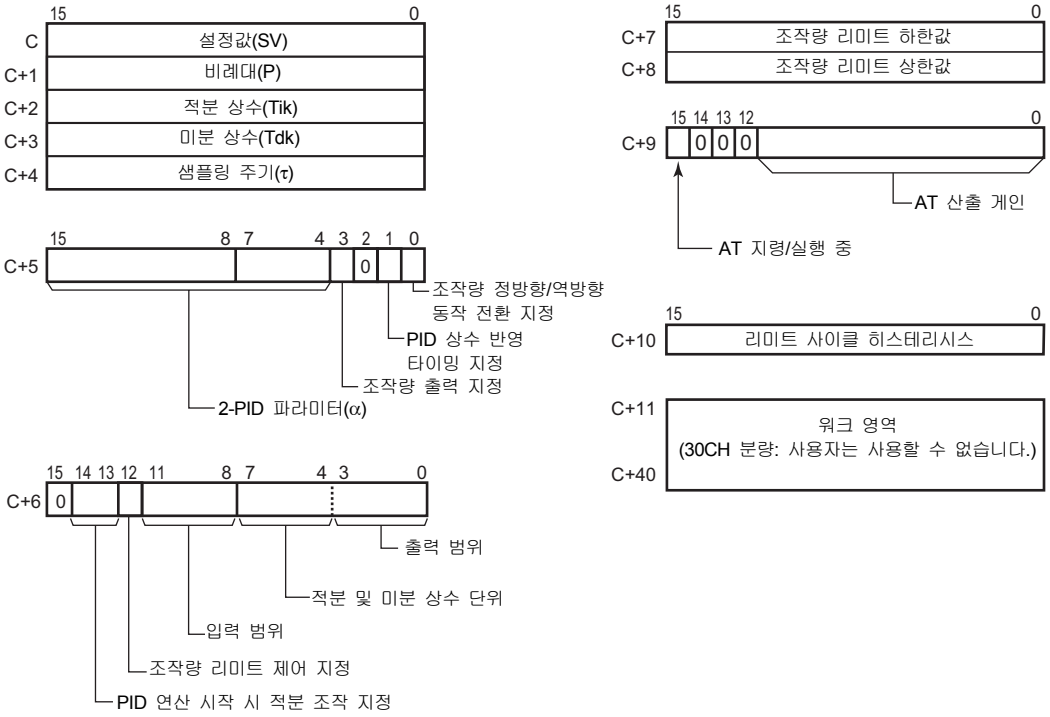
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	측정값 입력 CH 번호	UINT	1
C	PID 파라미터 저장 하위 CH 번호	WORD	41
D	조작량 출력 CH 번호	UINT	1

C: PID 파라미터 저장 하위 CH 번호



PIDAT

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
C												–						
D												○						

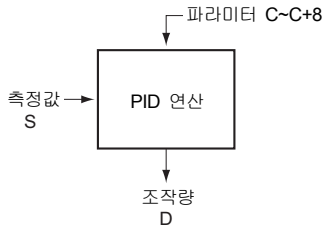
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C의 데이터가 범위 밖일 때 ON · 실제 샘플링 주기가 설정된 샘플링 주기의 2배를 초과했을 때 ON · 자동 조정 실행 시 이상이 발생했을 때 ON · 그 이외일 때 OFF
> 플래그	P_GT	· PID 연산된 조작량이 조작 출력 리미트 상한값을 상회할 때 ON · 그 이외일 때 OFF
< 플래그	P_LT	· PID 연산된 조작량이 조작 출력 리미트 하한값을 하회할 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· PID 연산을 실행했을 때 ON · 그 이외일 때 OFF

기능

C로 지정한 파라미터(설정값, PID 상수 등)에 따라 S를 측정값 입력으로 한 PID 연산(목표값 필터형 2 자유도 PID 연산)을 수행하여 D에 조작량을 출력합니다.
입력 조건이 기동(OFF→ON)될 때 파라미터를 전송하여 정상 범위 밖이면 에러 플래그를 ON으로 합니다.
정상 범위이면 그 때의 조작량을 초기값으로 PID 처리를 수행합니다.
입력 조건이 ON이면 지정된 샘플링 주기마다의 측정값을 입력으로 하여 연산을 수행합니다.

입력 조건이 OFF이면 PID 연산을 정지합니다. D의 조작량은 그 때의 값으로 유지됩니다. 변경해야 할 때는 래더 프로그램 또는 수동 조작으로 변경하십시오.



· 특정 사이클에서 AT 지령/실행 중(C+9의 비트 15)이 1(ON)이 되면(사이클마다 이 비트 확인), PID 상수의 자동 조정(AT)을 실행하기 시작합니다. (또한 AT 실행 중에는 SV 변경이 반영되지 않습니다.)

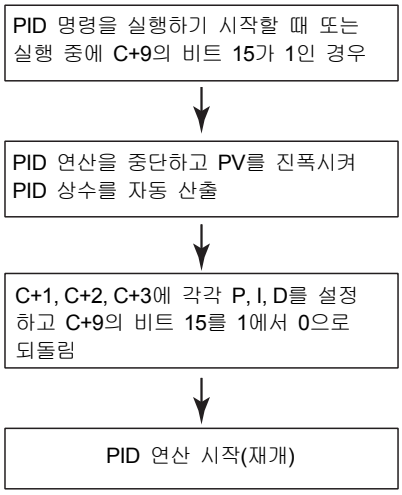
  AT는 리미트 사이클 규칙에 따릅니다. 조작량을 강제로 변경(최대 조작량←→최소 조작량)시켜 제어 대상의 특성을 관측하고, 그 관측 결과를 바탕으로 P, I, D 상수를 자동 연산하여 각각 C+1, C+2, C+3에 자동 저장합니다.

  동시에 AT 지령/실행 중(C+9의 비트 15)을 1(ON)에서 0(OFF)으로 설정하고, 자동 연산된 PID 상수(C+1, C+2, C+3)의 값을 바탕으로 PID 연산을 시작합니다.

· PIDAT 명령을 실행하기 시작할 때 C+9의 비트 15가 1인 경우:
  AT를 처음 실행한 후, 산출된 PID 상수로 PID 연산을 시작합니다.

· PIDAT 명령 실행 중에 C+9의 비트 15를 0에서 1로 설정한 경우:
  사용자가 미리 저장했던 PID 상수에 대한 PID 연산을 중단하고, AT를 실행한 후 산출된 PID 상수로 PID 연산을 시작(재개)합니다.

AT 실행 순서



주 1: AT 실행 중에 C+9의 비트 15가 1→0이 되어 AT를 중지한 경우, AT 실행 시작 직전의 PID 파라미터로 PID 연산을 시작합니다.

주 2: AT 실행 에러가 된 경우에도 AT 실행 시작 직전의 PID 파라미터로 PID 연산을 시작합니다. 1), 2)에서 모두 P, I, D의 각 파라미터는 AT 중지 시점의 것을 사용하여 연산합니다.

- 입력 범위(C+6의 비트 8~11) 지정을 통해 S의 측정값으로 입력된 16비트 중 유효로 할 비트 수를 지정합니다. 예를 들어, 입력 범위를 12비트(#4)로 지정하면 16진수 #0000~0FFF 범위의 값이 측정값으로 유효하게 됩니다. 입력 범위로 지정한 비트보다 큰 값이 입력되었을 때는 입력 범위의 최대값으로 보정합니다. 이 때, 에러 플래그는 ON되지 않습니다.
- 설정값의 범위도 입력 범위에 따릅니다.
- 측정값 및 설정값은 16진수 #0000~입력 범위 최대값의 부호 없는 BIN 값입니다.
- 출력 범위(C+6의 비트 0~3) 지정을 통해 조작량의 16비트 중 유효로 할 비트 수를 지정합니다. 예를 들어, 출력 범위를 12비트(#4)로 지정하면 16진수 #0000~0FFF 범위의 값이 조작량으로 출력됩니다.
- 비례 동작만인 경우 측정값 = 설정값일 때의 조작량을 0%와 50% 중 하나로 지정할 수 있습니다.
- 비례 동작의 정방향 동작, 역방향 동작을 지정할 수 있습니다.
- 조작량의 상하한을 지정할 수 있습니다.
- 샘플링 주기는 10ms 단위로 지정할 수 있습니다(0.01~99.99s). 단, 실제 PID 연산은 샘플링 주기와 PIDAT 명령 실행 시(사이클 시간마다)의 조합으로 실행 여부가 결정됩니다.
- PID 연산 실행 중에 비례대(P), 적분 상수(Tik), 미분 상수(Tdk)를 변경한 경우, 해당 변경 사항을 샘플링 주기별로 PID 연산에 반영/반영하지 않음 중 하나를 지정할 수 있습니다. PID 상수 반영 타이밍을 지정(C+5의 비트 1)하여 수행합니다.

힌트

- PIDAT 명령(191)은 PID 명령(190)에 AT(자동 조정) 기능을 추가한 명령입니다. PID 제어 동작 자체는 PID 명령(190)과 동일합니다. PID 제어 동작에 대한 자세한 내용은 PID 명령(190) 항목을 참조하십시오.
- PIDAT 명령은 입력 조건의 기동을 STOP→RUN으로 처리하여 실행됩니다. 입력 조건이 기동될 때 C+11~C+40을 초기화(클리어)한 후, 다음 사이클 이후 입력 조건이 ON된 상태이면 PIDAT 연산을 실행합니다.
  따라서 상시 ON 플래그(ON)를 PIDAT 명령의 입력 조건으로 하는 경우는 C+11~C+40을 운전 시작 시 초기화(클리어)하는 처리를 별도로 설정하십시오.

사용상의 주의 사항

- IL-ILC 명령 간, JMP-JME 명령 간 또는 서브루틴 명령 내에서 PIDAT 명령을 사용하는 경우, 입력 조건에 따라 동작이 안정되지 않는 경우가 있으므로 주의하십시오.
- 복수의 PIDAT 명령에서 PID 파라미터 저장 CH를 공유하여 사용할 수 없습니다. 복수의 PIDAT에서 같은 파라미터를 사용하는 경우에도 반드시 다른 CH를 지정하십시오.
- 수동으로 PID 상수를 변경할 때는 이 PID 상수 반영 타이밍 지정(C+5의 비트 1)을 1(C+1, C+2, C+3의 값을 샘플링 주기별로 PID 연산에 반영)로 합니다. 또한 이 지정은 AT 실행 후 PID 상수를 수동 조정하려는 경우에도 유효합니다.
- PID 파라미터(C~C+38) 중 입력 조건을 ON으로 한 상태에서 변경할 수 있는 값은 다음 파라미터입니다. 그 밖의 값을 변경했을 때는 반드시 입력 조건을 OFF→ON으로 기동하십시오.
  - C의 설정값(SV) (단, PID 연산 실행 중에만. AT 실행 중에는 변경한 값이 반영되지 않습니다.)
  - PID 상수의 반영 타이밍 지정
  - PID 상수 반영 타이밍(C+5의 비트 1)이 1(C+1, C+2, C+3의 값을 샘플링 주기별로 반영)일 때, C+1, C+2, C+3의 각 P, I, D 상수
  - C+9의 비트 15의 AT 지령/실행 중 비트
  - C+9의 비트 0~14의 AT 산출 게인 및 C+10의 리미트 사이클 히스테리시스(단, 어느 경우든 AT 시작 시가 1일 때 전송됩니다.)

PID 파라미터 설정 내용

제어 데이터	항목	내용	설정 범위	입력 조건 ON인 상태에서 변경 가능 여부
C	설정값(SV)	제어 대상의 목표값입니다.	입력 범위에서 비트 수 분량의 BIN 데이터 (0~지정된 입력 범위 최대값)	가능(단, AT 실행 중에는 불가능)
C+1	비례대(P)	비례 제어 범위/제어 범위 전체로 나타내는 P 제어용 파라미터입니다.	10진수 &1~9999 또는 16진수 #0001~270F(0.1% 단위, 0.1~999.9%)	입력 조건 ON인 경우 C+5의 비트 1이 1일 때 가능
C+2	적분 상수(Tik)	적분 동작 효과의 크기를 정확히 나타내는 상수입니다. 이 값을 크게 하면 적분 효과가 약해집니다.	10진수 &1~8191 또는 16진수 #0001~1FFF, 10진수 &9999 또는 16진수 #270F(적분 동작 없음 설정), 적분 및 미분 상수 단위 지정이 "1"일 때: 1~8191배, "9"일 때: 0.1~819.1s(주 1)	
C+3	미분 상수 (Tdk)	미분 동작의 크기를 나타내는 상수입니다. 이 값을 크게 하면 미분 효과가 강해집니다.	10진수 &1~8191 또는 16진수 #0001~1FFF, 10진수 &0 또는 16진수 #0000(미분 동작 없음 설정), 적분 및 미분 상수 단위 지정이 "1"일 때: 1~8191배, "9"일 때: 0.1~819.1s(주 1)	
C+4	샘플링 주기 (τ)	PID 연산을 수행하는 주기를 설정합니다.	10진수 &1~9999 또는 16진수 #0001~270F(10ms 단위, 0.01~99.99s)	불가능
C+5의 비트 4~15	2-PID 파라미터(α)	입력 필터 계수입니다. 일반적으로 0.65로 사용하십시오. 값이 0에 가까울수록 필터 성능이 떨어집니다.	16진수 #000: α = 0.65(16진수 3자리), 16진수 #100~163으로 하면 하위 2자리의 값이 α = 0.00~0.99를 의미합니다. (주 2)	
C+5의 비트 3	조작량 출력 지정	측정값 = 설정값일 때의 조작량을 지정합니다. 주: 이 설정은 '적분 동작 없음'일 때 유효합니다.	0, 1 0: 출력 0% 1: 출력 50%	
C+5의 비트 1	PID 상수 반영 타이밍 지정	P(비례대), Tik(적분 상수), Tdk(미분 상수)의 각 파라미터를 언제 PID 연산에 반영시킬 지를 지정합니다.	0, 1 0: 입력 조건 기동 시에만 1: 입력 조건 기동 시 및 샘플링 주기마다	가능
C+5의 비트 0	조작량 정/역 동작 전환 지정	비례 동작의 방향을 결정하는 파라미터입니다.	0, 1 0: 역방향 동작 1: 정방향 동작	불가능
C+6의 비트 14, 13	PID 연산 시작 시 적분 조작 지정(유니트 Ver.4.0 이상만)	PID 연산 시작(입력 OFF→ON) 시의 적분 조작량의 초기 상태를 지정합니다.	비트 14 = 0, 비트 13 = 0: 연산을 시작할 때의 조작량 출력 지정과 동일한 적분 조작량부터 시작합니다(기존과 동일). 비트 14 = 0 또는 1, 비트 13 = 1: 조작량 출력을 갑자기 바꾸지 않고 연속적으로 변화하는 적분 조작량부터 시작합니다(범프리스 처리). 비트 14 = 1, 비트 13 = 0: 적분 조작량=0부터 시작합니다.	불가능
C+6의 비트 12	조작량 리미트 제어 지정	조작량을 리미트 제어할지 여부를 지정입니다.	0, 1 0: 무효(리미트 제어 안 함) 1: 유효(리미트 제어) (주 3)	
C+6의 비트 8~11	입력 범위	입력 데이터의 비트 수입니다.	(16진수 1자리) 0: 8비트, 1: 9비트, 2: 10비트 3: 11비트, 4: 12비트, 5: 13비트 6: 14비트, 7: 15비트, 8: 16비트	
C+6의 비트 4~7	적분 및 미분 상수 단위 지정	적분 상수 및 미분 상수의 시간 단위 지정입니다.	9(16진수 1자리) 1: 샘플링 주기 배수 지정 적분 및 미분 시간을 샘플링 주기의 지정 배수 시간으로 지정합니다. 9: 시간 지정 적분 및 미분 시간을 100ms 단위로 지정합니다(주 1).	
C+6의 비트 0~3	출력 범위	출력 데이터의 비트 수입니다.	설정 범위는 입력 범위와 동일합니다.	
C+7	조작량 리미트 하한값	조작량을 리미트 제어했을 때의 리미트 하한값입니다.	10진수 &0~65535 또는 16진수 #0000~FFFF(주 3)	
C+8	조작량 리미트 상한값	조작량을 리미트 제어했을 때의 리미트 상한값입니다.	10진수 &0~65535 또는 16진수 #0000~FFFF(주 3)	

제어 데이터	항목	내용	설정 범위	입력 조건 ON인 상태에서 변경 가능 여부
C+9의 비트 15	AT 지령/실행 중	PID 상수의 AT(자동 조정) 실행 지령과 AT 실행 중 플래그를 겸합니다. · AT 실행 시, 1로 설정합니다(PIDAT 명령 실행 중에도 유효합니다). · AT가 종료되면 자동으로 0으로 돌아갑니다. 주: AT 실행 중에 1에서 0으로 설정하면 AT가 중지되고 AT 실행 시작 직전의 PID 파라미터로 PID 연산이 시작됩니다. 단, P, I, D의 각 파라미터는 중지 시점의 값이 유효하게 됩니다.	0, 1 · AT 실행 지령으로 0→1: AT 실행 지시(PIDAT 명령 실행 시 1일 때도 AT 실행 지시) 1→0: AT 중지 지시 또는 AT 종료 시 자동으로 변경됨 · AT 실행 중 플래그로 0: AT 비실행 중 1: AT 실행 중	가능
C+9의 비트 0~11	AT 산출 게인	AT에 의한 PID 조정의 산출 결과의 자동 저장 값에 대한 기여도를 사용자 정의로 조정할 때 설정합니다. 일반적으로 기본값 그대로 사용합니다. · 안정성을 중시하는 경우에는 크게 설정합니다. · 속응성을 중시하는 경우에는 작게 설정합니다.	10진수 &0 또는 16진수 #0000: 1.00 (기본값) 10진수 &1~1000 또는 16진수 #0001~03E8: 0.01~10.00(0.01 단위)	가능(단, 반영 타이밍은 AT 시작 시)
C+10	리미트 사이클 히스테리시스	SV에서 리미트 사이클을 발생시키는 경우의 히스테리시스를 설정합니다. 기본적으로 역방향 동작의 경우, SV－0.20%의 히스테리시스로 MV를 ON시킵니다. PV가 불안정하여 정상적인 리미트 사이클을 발생시킬 수 없는 경우, 이 값을 크게 설정합니다. 단, 필요 이상으로 크게 설정하면 AT 정도가 악화됩니다.	10진수 &0 또는 16진수 #0000: 0.20% (기본값) 10진수 &1~1000 또는 16진수 #0001~03E8: 0.01~10.00%(0.01 단위) 16진수 #FFFF: 0.00% 주: 입력 범위에 대한 %	

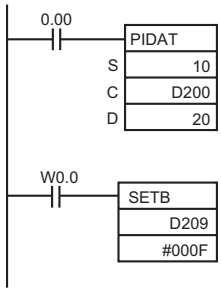
주 1: 적분 및 미분 상수 단위 지정을 시간 지정("9")으로 한 경우는 적분 시간 및 미분 시간 설정을 샘플링 주기의 1~8191배에 해당하는 시간 이내로 설정하십시오.

주 2: 000을 설정하면 2-PID 파라미터 α가 0.65가 됩니다. 평상시에는 000으로 설정하십시오.

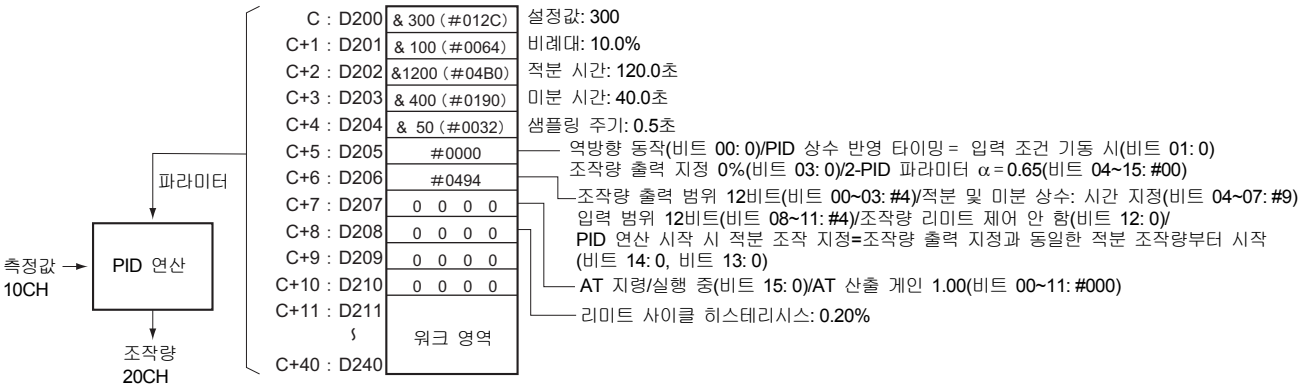
주 3: 조작량 리미트 제어 지정을 유효("1")로 한 경우에는 각 값을 다음과 같이 설정하십시오.
0000≦조작량 리미트 하한값≦조작량 리미트 상한값≦지정 출력 범위 최대값

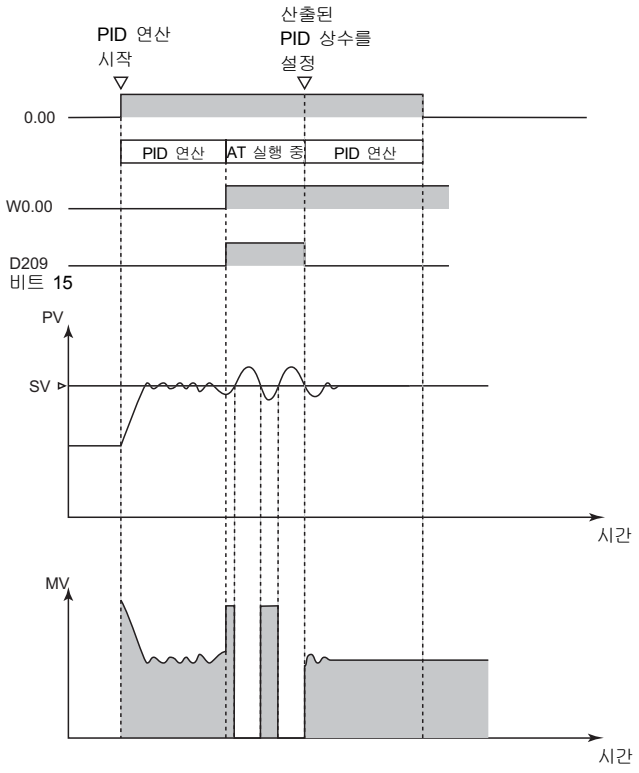
샘플 프로그램

■ PID 명령 실행 후 다른 조건에서 AT를 실행하는 경우

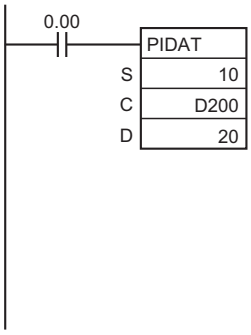


- 0.00의 기동(OFF→ON)되었을 때 D200~D208에 설정된 다음 파라미터에 따라 D211~D240의 워크 영역의 초기화(클리어)를 수행합니다. 초기화 종료 후 PID 연산을 수행하여 조작량을 20CH에 출력합니다.
- 0.00이 ON이면 D200~D210에 설정된 파라미터에 따라 PID 연산을 샘플링 주기 간격으로 실행하여 조작량을 20CH에 출력합니다.
- 비례대(P), 적분 상수(Tik), 미분 상수(Tdk)의 각 파라미터를 0.00의 기동 이후에 변경한 경우, PID 연산에는 반영되지 않습니다.
- W0.0이 ON→OFF로 종료되면 SETB 명령을 사용하여 D209(C+9)의 비트를 1(ON)로 하고 AT 실행을 시작합니다. AT 실행 종료 후 산출된 P, I, D 상수를 각각 C+1, C+2, C+3에 설정하고, 해당 PID 상수에 기초하여 PID 연산을 실행합니다.

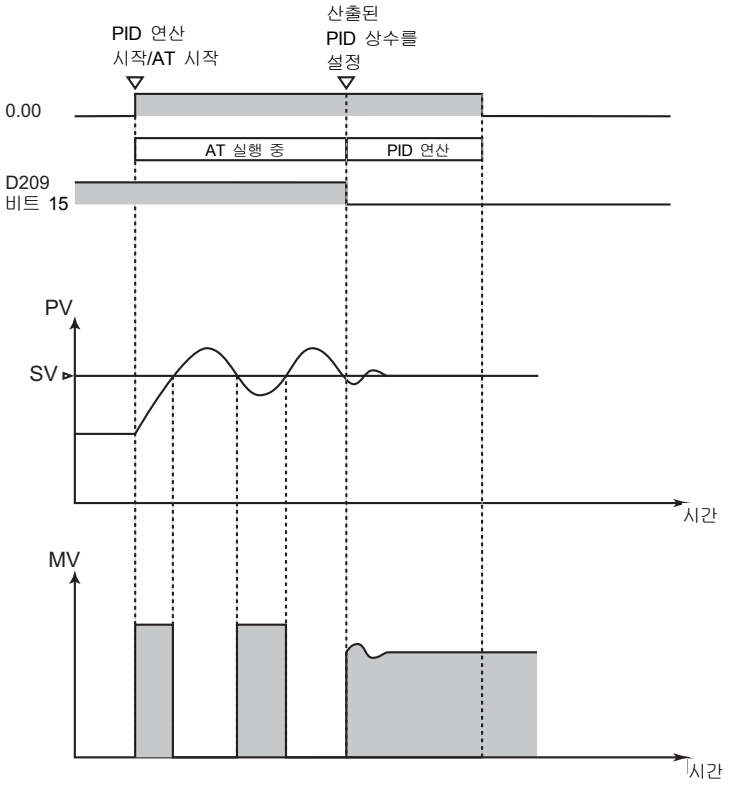




■ PID 명령 실행 시 즉시 AT를 실행하는 경우

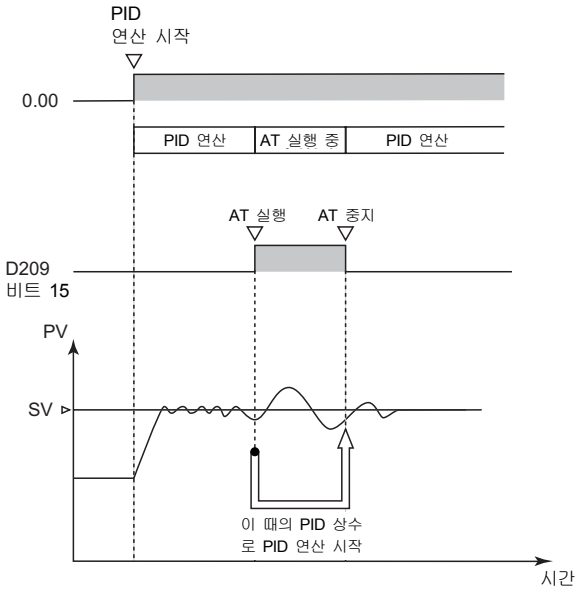


- 0.00이 ON인 경우 D209(C+9)의 비트 15가 1(ON)이면 먼저 AT가 실행됩니다. AT 실행 종료 후 산출된 P, I, D 상수를 각각 C+1, C+2, C+3에 설정하고, 해당 PID 상수에 기초하여 PID 연산을 실행합니다.



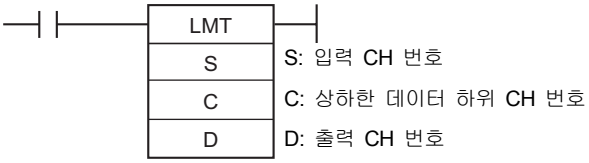
■ AT 중에 AT를 중지한 경우

AT 실행 중에 D209(C+9)의 비트 15를 1(ON)에서 0(OFF)으로 하면 AT는 중지되고 AT 실행 시작 직전의 PID 상수로 PID 연산을 재개합니다.



LMT

명령 이름	니모닉	동작 옵션	평션 번호	기능
상하한 리미트 제어	LMT	@LMT	680	입력 데이터가 상하한 리미트 데이터의 범위 안인지 여부에 따라 출력 데이터를 제어합니다.

	LMT
심볼	

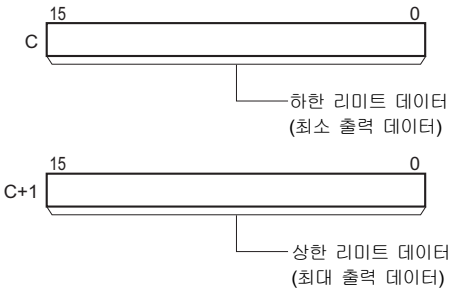
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	입력 CH 번호	INT	1
C	리미트 데이터 하위 CH 번호	DINT	2
D	출력 CH 번호	INT	1

C: 리미트 데이터 하위 CH 번호



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
C											–	–						
D												○						

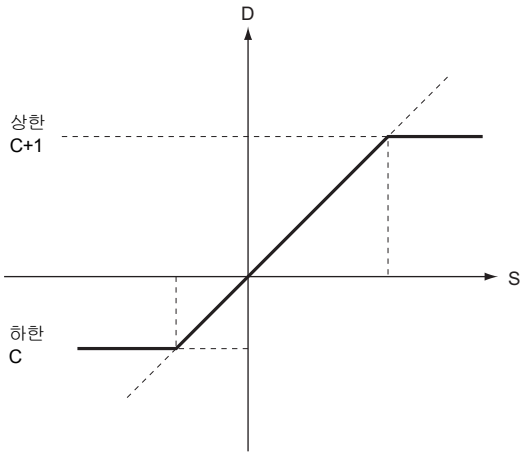
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 상한 리미트 데이터 < 하한 리미트 데이터일 때 ON · 그 이외일 때 OFF
> 플래그	P_GT	· S > 상한 리미트 데이터일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 결과가 0일 때 ON · 그 이외일 때 OFF

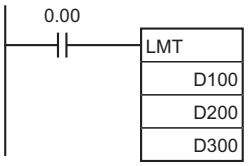
이름	라벨	내용
< 플래그	P_LT	· S<하한 리미트 데이터일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

S(부호 있는 BIN 데이터: 10진수 −32768~+32767 또는 16진수 #0000~FFFF)에 대해 하한 리미트 데이터≦S≦상한 리미트 데이터일 때 S를 D에 출력합니다. S>상한 리미트 데이터일 때 상한 리미트 데이터를 D에 출력합니다. S<하한 리미트 데이터일 때 하한 리미트 데이터를 D에 출력합니다.



샘플 프로그램

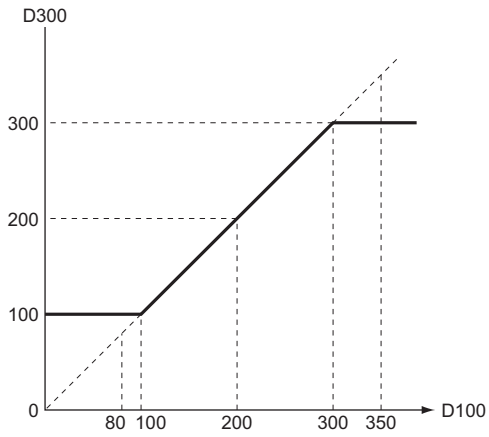


C : D200	+100 (#0064)	하한 리미트 100
D201	+300 (#012C)	상한 리미트 300

예: D100이 +80(＃0050)일 때
80＜하한 리미트 100이므로 D300에 +100(#0064)

D100이 +200(#00C8)일 때
하한 리미트 100＜200＜상한 리미트 300이므로 D300에 +200(#00C8) 출력

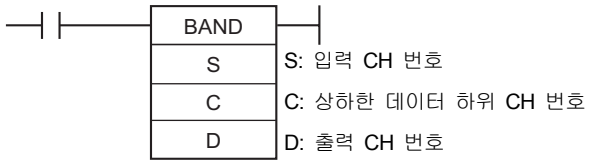
D100이 +350(#015E)일 때
상한 리미트 300＜350이므로 D300에 +300(#012C) 출력



BAND

명령 이름	니모닉	동작 옵션	평션 번호	기능
데드 밴드 제어	BAND	@BAND	681	입력 데이터가 불감대(Dead band)의 범위 안인지 여부에 따라 출력 데이터를 제어합니다.

심볼	BAND



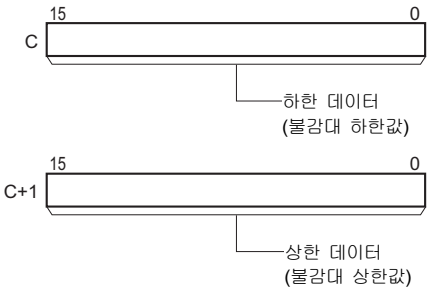
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	입력 CH 번호	INT	1
C	상하한 데이터 하위 CH 번호	UINT	2
D	출력 CH 번호	UINT	1

C: 상하한 데이터 하위 CH 번호



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
C											–	–						
D												○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 상한 데이터 < 하한 데이터일 때 ON · 그 이외일 때 OFF
> 플래그	P_GT	· S > 상한 데이터일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 결과가 0일 때 ON · 그 이외일 때 OFF

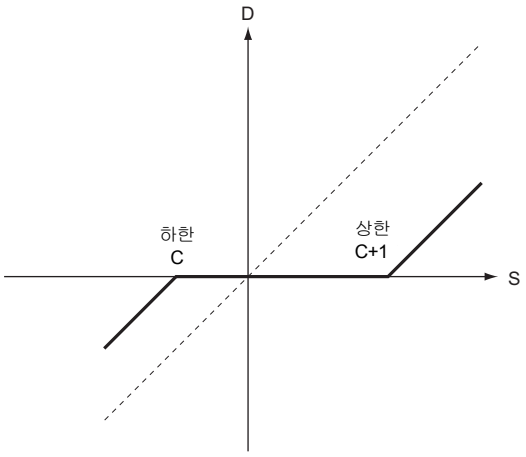
이름	라벨	내용
< 플래그	P_LT	· S<하한 데이터일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

S로 지정한 데이터(부호 있는 BIN 데이터: 10진수 −32768~+32767 또는 16진수 #0000~FFFF)에 대해 하한 데이터<S<상한 데이터일 때(불감대 이내) 0을 D에 출력합니다.

S>상한 데이터인 경우 (S로 지정한 데이터)−(상한 데이터)를 D에 출력합니다.

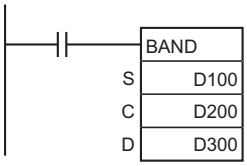
S<하한 데이터인 경우 (S로 지정한 데이터)−(하한 데이터)를 D에 출력합니다.



출력 데이터가 −32768(16진수 #8000)보다 작아진 경우 또는 +32767(16진수 #7FFF)보다 커진 경우는 부호가 반전됩니다.

예) 하한 데이터 −32768(16진수 #8000), 입력 데이터 256(16진수 #0100)일 때
출력 데이터: −32768(#8000) − 256(#0100) = 32512(#7F00)

샘플 프로그램

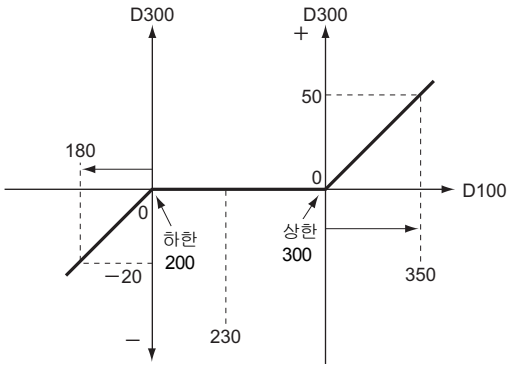


C : D200	+200(#00C8)	하한 데이터 200
D201	+300(#012C)	상한 데이터 300

예. S: D100이 +180(#00B4)일 때
180<하한 데이터 200이므로 180 − 200 = -20(#FFEC)을 D300에 출력

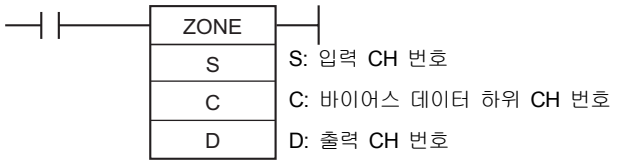
D100이 +230(#00E6)일 때
하한 데이터 200<230<상한 데이터 300이므로 0을 D300에 출력

D100이 +350(#015E)일 때
상한 데이터 300<350이므로 350 − 300 = +50(#0032)을 D300에 출력



ZONE

명령 이름	니모닉	동작 옵션	평션 번호	기능
데드 존 제어	ZONE	@ZONE	682	지정한 바이어스 값을 입력 데이터에 더하여 출력합니다.

심볼	ZONE
	

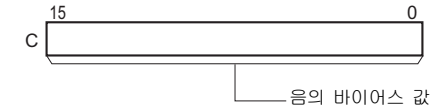
특정 영역 내의 사용 여부

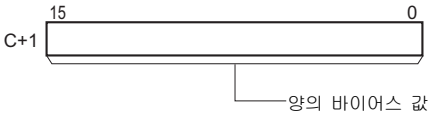
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	입력 CH 번호	INT	1
C	바이어스 데이터 하위 CH 번호	UDINT	2
D	출력 CH 번호	UINT	1

C: 바이어스 데이터 하위 CH 번호





■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
C											–	–						
D												○						

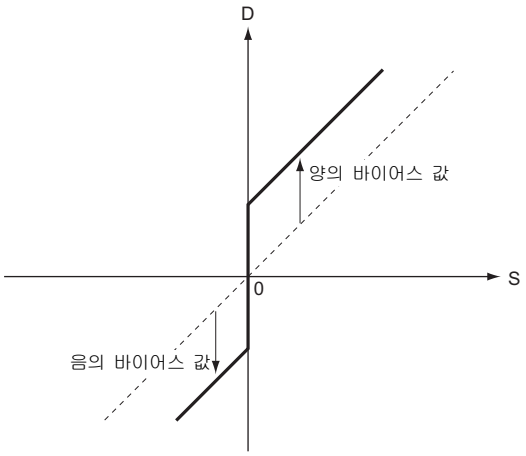
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 양의 바이어스 값<음의 바이어스 값일 때 ON · 그 이외일 때 OFF
> 플래그	P_GT	· S>0일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 결과가 0일 때 ON · 그 이외일 때 OFF
< 플래그	P_LT	· S<0일 때 ON · 그 이외일 때 OFF

이름	라벨	내용
부정 플래그	P_N	· 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

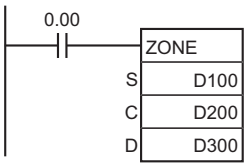
S로 지정한 입력 데이터(부호 있는 BIN 데이터: 10진수 −32768~+32767 또는 16진수 #0000~FFFF)에 대해 입력 데이터<0인 경우, 입력 데이터+음의 바이어스 값(C+0)을 D에 출력합니다. 입력 데이터>0인 경우, 입력 데이터+양의 바이어스 값(C+1)을 D에 출력합니다. 입력 데이터=0이면 0을 D에 출력합니다.



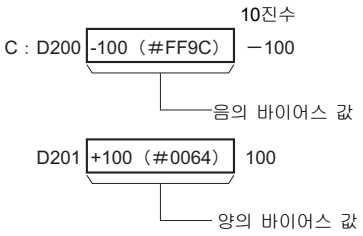
출력 데이터가 −32768(16진수 #8000)보다 작아진 경우 또는 32767(16진수 #7FFF)보다 커진 경우는 부호가 반전됩니다.

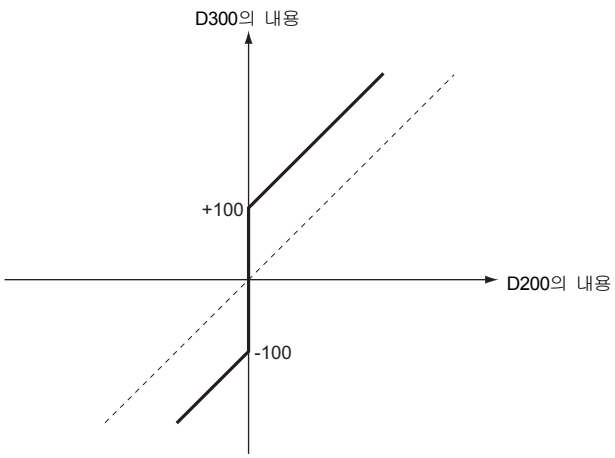
예) 음의 바이어스 값 −256(16진수 #FF00), 입력 데이터 −32767(16진수 #8000)일 때
출력 데이터: −32767(#8000) + −256(#FF00) = 32512(#7F00)

샘플 프로그램



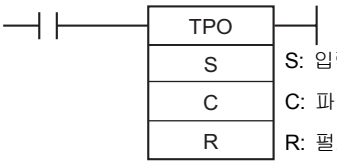
0.00이 ON이면 D100의 값(부호 있는 BIN 데이터)에 대해 그 값이 0보다 작을 때 −100을 더한 값을 D300에 저장합니다.
그 값이 0일 때는 &0을 D300에 저장합니다.
그 값이 0보다 클 때는 +100을 더한 값을 D300에 저장합니다.





TPO

명령 이름	니모닉	동작 옵션	평션 번호	기능
시분할 비례 출력	TPO	–	685	지정된 채널 내의 듀티비 또는 조작량을 입력하고 지정된 파라미터에 따라 듀티비를 시분할 비례 출력으로 변환하여 지정한 접점에 펄스 출력합니다.

심볼	TPO
	 S: 입력 듀티비 또는 조작량 저장 CH 번호 C: 파라미터 저장 하위 CH 번호 R: 펄스 출력 릴레이 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	입력 듀티비 또는 조작량 저장 CH 번호	UINT	1
C	파라미터 저장 하위 CH 번호	WORD	7
R	펄스 출력 릴레이 번호	BOOL	–

S: 입력 듀티비 또는 조작량 저장 CH 번호

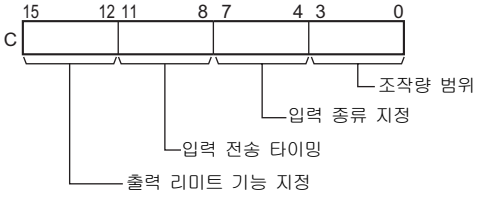
입력 듀티비 또는 조작량 저장 채널을 지정합니다.

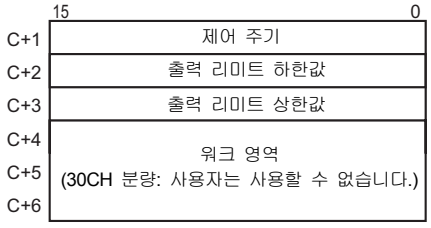
- 입력 듀티비: 10진수 &0~10000 또는 16진수 #0000~2710(0.00~100.00%)
- 입력 조작량*1: 10진수 &0~최대 65535 또는 16진수 #0000~최대 FFFF(C의 비트 00~03으로 유효 자릿수로서의 조작량 범위를 지정합니다. PID 연산 명령의 조작량 출력의 출력 범위에 일치시킵니다.)

*1: S를 입력 조작량으로 처리하는 경우, PID 연산 명령(PID) (또는 자동 조정 가능한 PID 연산 명령(PIDAT))에 의한 조작량이 저장되는 채널 번호를 지정합니다.

C: 파라미터 저장 하위 CH 번호

C의 비트 04~07(입력 종류 지정)을 사용하여 입력 듀티비, 입력 조작량 중 어느 데이터로 처리할지 지정합니다(C의 비트 04~07이 0인 경우 듀티비, C의 비트 04~07이 1인 경우는 조작량).





주: 자세한 내용은 뒤에 설명하는 파라미터 설정 내용 참조

R: 펄스 출력 릴레이 번호

펄스 출력의 출력 대상 접점을 지정합니다.

평상시에는 트랜지스터 출력 유니트의 할당 접점을 지정하고 해당 트랜지스터 출력 유니트에 SSR(솔리드 스테이트 릴레이)를 연결합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
C					○	○	○	○	○	○	–	–						
R					–	–	–	–	–	–								

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S가 범위 밖인 경우 ON (주) 범위는 입력 종류에 따라 바뀜 · C가 범위 밖인 경우 ON (주) 조작량 범위는 입력 종류가 조작량으로 지정된 경우만 해당 · 제어 주기(C+1)가 범위 밖일 때 ON · 출력 리미트 기능이 유효이고 출력 리미트 하한값(C+2), 출력 리미트 상한값(C+3)이 범위 밖일 때 ON · 출력 리미트 기능이 유효이고 출력 리미트 하한값(C+2)≦출력 리미트 상한값(C+3)이 아닐 때 ON · 그 이외일 때 OFF

기능

S로 지정한 채널 번호 내의 듀티비 또는 조작량을 입력하고 C~C+3으로 지정한 파라미터에 따라 듀티비를 시분할 비례 출력*1으로 변환하여 R로 지정한 접점에 펄스 출력합니다.

*1: 시분할 비례 출력이란 입력값(S)에 따라 ON과 OFF의 시간비를 비례적으로 변화시키는 출력입니다. ON과 OFF의 시간비를 변화시키는 주기를 '제어 주기'라고 하며 C+1로 지정합니다.

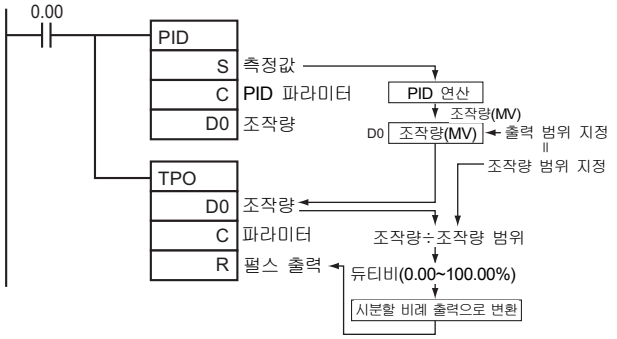
예) 제어 주기 = 1s일 때, 입력값이 50%인 경우 0.5s 동안 ON, 0.5s 동안 OFF가 되고, 입력값이 80%인 경우는 0.8s 동안 ON, 0.2s 동안 OFF가 됩니다.

일반적으로 PID 연산 명령(PID) (또는 자동 조정 가능한 PID 연산 명령(PIDAT))과 세트로 사용하여 조작량 출력 결과(D)를 이 명령의 S로 지정합니다.

그리고 R에는 트랜지스터 출력 유니트의 할당 접점을 지정하고, 그 트랜지스터 출력 유니트에 SSR(솔리드 스테이트 릴레이)를 연결하여 히터의 시분할 비례 제어(ON과 OFF의 시간비에 대한 비례적인 제어)를 수행합니다.

■ PID 연산 명령과 조합한 경우의 예

PID 연산 명령과 조합한 경우는 입력 조작량을 바탕으로 조작량을 조작 범위로 나눈 값을 듀티비로 하고, 그 듀티비를 시분할 비례 출력으로 변환하여 펄스 출력합니다.

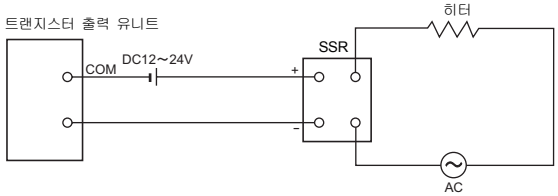


이 경우, PID 연산 명령의 출력 범위와 이 명령의 조작량 범위를 일치시켜 둡니다.

예) 출력 범위 = 조작량 범위 지정이 12비트(16진수 #0000~0FFF)의 경우 PID 연산 명령에 의한 조작량을 16진수 #0FFF로 나눈 값을 듀티비로 합니다. 이 명령은 그 값을 시분할 비례 출력으로 변환합니다.

■ 외부 배선의 예

트랜지스터 출력 유니트와 SSR를 다음과 같이 연결합니다.



TPO

파라미터 설정 내용

	항목	내용	설정 범위	입력 조건 ON인 상태에서 변경 가능 여부
C의 비트 0~3	조작량 범위	입력 데이터의 비트 수입니다.	(16진수 1자리) 0: 8비트, 1: 9비트, 2: 10비트 3: 11비트, 4: 12비트, 5: 13비트 6: 14비트, 7: 15비트, 8: 16비트	가능
C의 비트 4~7	입력 종류 지정	입력(S) 내의 데이터가 듀티비인지, 조작량인지 선택합니다.	0: 듀티비 주: S 값의 범위 : 듀티비: 10진수 &0~10000 또는 16진수 #0000~2710(0~100.00%) 1: 조작량 주: S 값의 범위: 10진수 &0~65535 또는 16진수 #0000~FFFF 상한은 조작량 범위(C의 비트 0~3)에 따라 달라짐	가능
C의 비트 8~11	입력 전송 타이밍 지정	입력 전송 타이밍을 지정합니다.	0: 제어 주기마다 1: 하방 우선 2: 상방 우선 3: 상시	
C의 비트 12~15	출력 리미트 기능 지정	출력 리미트 기능의 유효/무효를 지정합니다.	0: 무효(리미트 제어 안 함) 1: 유효(리미트 제어) (주)	가능
C+1	제어 주기	제어 주기(ON과 OFF의 시간비를 변화시키는 주기)	10진수 &100~9999 또는 16진수 #0064~270F(1.00~99.99초) 주: 예를 들어, 1초는 &1이 아니라 10진수 &100으로 설정한다는 점에 주의하십시오.	가능
C+2	출력 리미트 하한값	출력을 리미트 제어했을 때의 리미트 하한값입니다.	10진수 &0~10000 또는 16진수 #0000~2710(0~100.00%)	가능
C+3	출력 리미트 상한값	출력을 리미트 제어했을 때의 리미트 상한값입니다.	10진수 &0~10000 또는 16진수 #0000~2710(0~100.00%)	가능
C+4 C+5 C+6	워크 영역	시스템이 사용하는 영역입니다. 사용자는 이 채널을 사용할 수 없습니다.	사용 불가능	–

주: 출력 리미트 제어 지정을 유효("1")로 한 경우에는 각 값을 다음과 같이 설정하십시오.
10진수 &0≦출력 리미트 하한값≦출력 리미트 상한값≦10진수 &10000

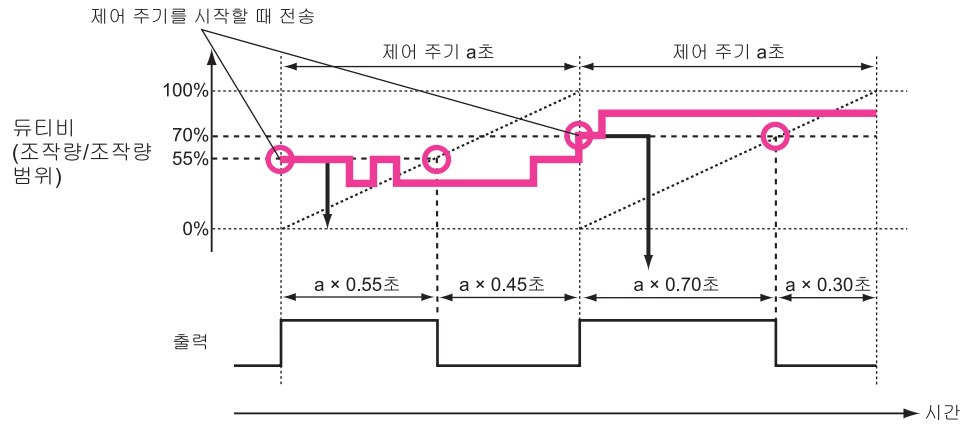
기능 설명

- 입력 조건이 ON인 동안 명령을 실행합니다.
- 명령 실행을 시작하면 듀티비에 맞춰 출력 접점(R)을 ON/OFF합니다.
- 파라미터(C~C+3)는 명령을 실행할 때마다 실시간으로 전송됩니다. 변경하는 경우는 동시에 변경하십시오 (각 파라미터 간의 정합성이 깨지지 않도록).
- 명령 실행 타이밍에서 출력(R)의 ON/OFF가 수행됩니다. 또한 출력(R)의 ON/OFF 타이밍 정도는 최대 10ms입니다.
- 입력 조건이 OFF가 되면 명령 실행을 중단합니다. 이때, 경과 시간이 리셋되고 제어 주기가 초기화됩니다.
- 입력 종류 지정(C의 비트 04~07)을 통해 입력(S) 내의 데이터가 듀티비인지, 조작량인지 선택합니다. 입력(S) 내의 데이터가 조작량인 경우, 듀티비는 조작량(S)÷조작량 범위(C의 비트 0~3에 따라 결정)로 계산됩니다.
- 입력(S)의 전송 타이밍은 입력 전송 타이밍 지정(C의 비트 8~11)을 통해 지정합니다.

'0: 제어 주기마다'인 경우	제어 주기 시작 시점에서 듀티비 입력을 전송하고 주기 도중의 듀티비 변화는 무시합니다.
'1: 하방 우선'인 경우	제어 주기 시작 시점의 듀티비보다 듀티비 입력이 하강된 경우, 하강된 값을 우선하여 ON 시간을 하방 수정합니다.
'2: 상방 우선'인 경우	제어 주기 시작 시점의 듀티비보다 듀티비 입력이 상승된 경우, 상승된 값을 우선하여 ON 시간을 상방 수정합니다.
'3: 상시'인 경우	명령이 실행될 때마다 실시간으로 듀티비를 전송하며 제어 주기 내에서 ON/OFF를 반복합니다.

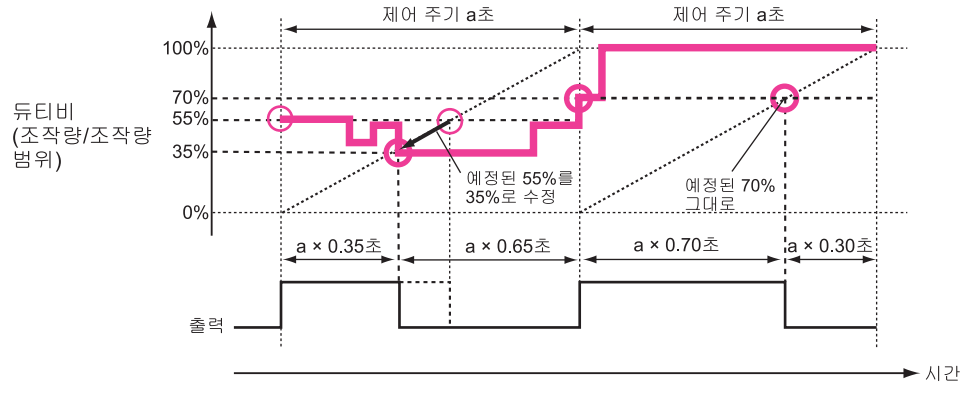
아래에서 그림으로 설명합니다.

- 입력 전송 타이밍이 '0: 제어 주기마다'로 지정된 경우



제어 주기를 시작할 때의 듀티비에 따라 해당 주기의 출력을 결정합니다.
참고: 일반적인 용도로 사용합니다.

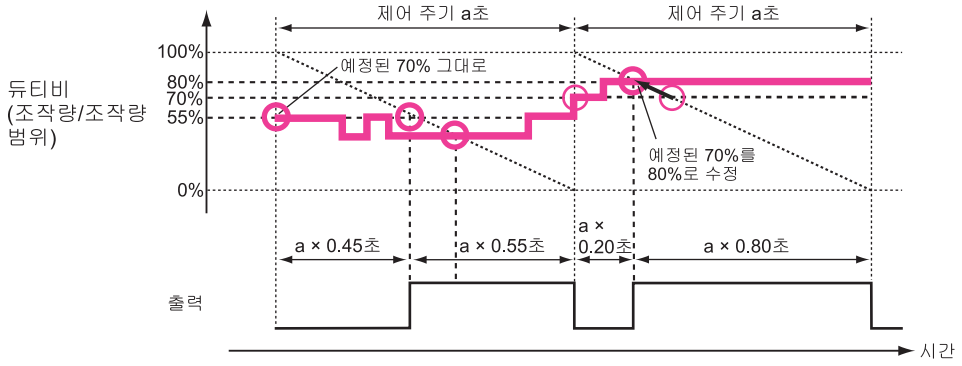
- 입력 전송 타이밍이 '1: 하방 우선'으로 지정된 경우



제어 주기 시작 시의 듀티비보다 빨리 하강한 경우만 하강된 듀티비로 수정하여 OFF로 합니다.
참고: 시분할 비례로 가열 동작을 할 때 제어 주기가 비교적 긴 경우, 가열에 의한 오버슈트를 방지하고 싶은 경우 등에 사용합니다.

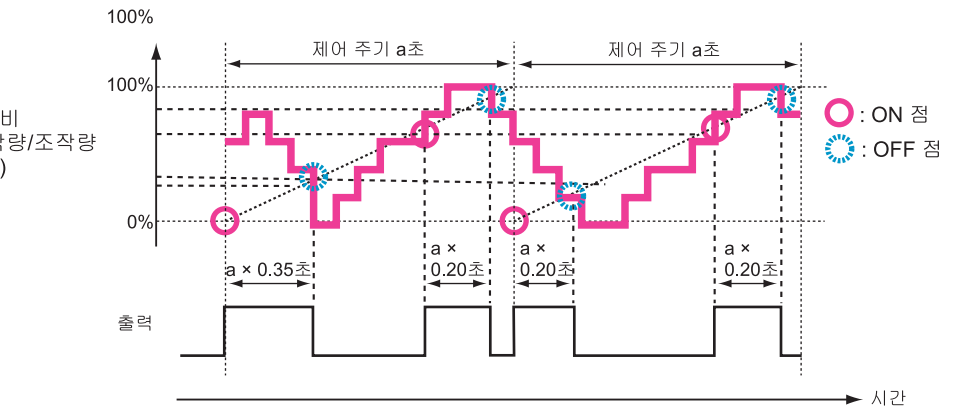
TPO

· 입력 전송 타이밍이 '2: 상방 우선'으로 지정된 경우



제어 주기 시작 시의 듀티비보다 빨리 상승한 경우만 상승된 듀티비로 수정하여 ON으로 합니다.
(이 경우만 출력의 ON, OFF 순서가 OFF→ON 순서로 됩니다.)
참고: 시분할 비례로 냉각 동작을 할 때 제어 주기가 비교적 긴 경우, 냉각에 의한 언더슈트를 방지하고 싶은 경우 등에 사용합니다.

· 입력 전송 타이밍이 '3: 상시'로 지정된 경우

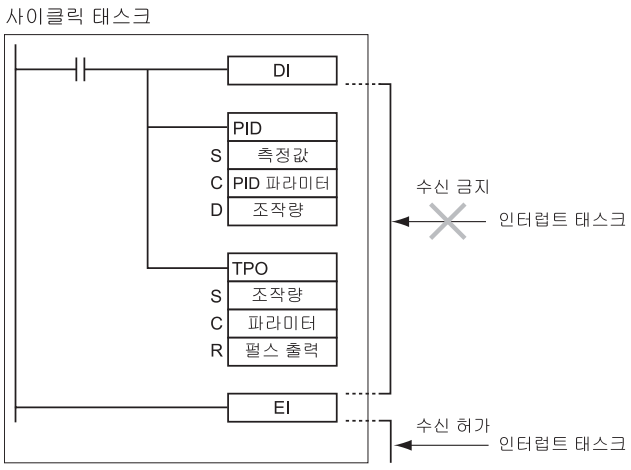


듀티비 변화를 실시간으로 따릅니다.
제어 주기 시작 시의 듀티비보다 빨리 하강한 경우 하강된 듀티비로 수정하여 OFF로 합니다. 그 후 상승하면 다시 수정하여 ON으로 하며, 이것을 반복합니다.
참고: 제어 주기가 비교적 긴 경우이고 듀티비 변화가 심한 경우, 제어성을 향상시키고 싶거나 조명/전력을 세밀하게 제어하고 싶은 경우 등에 사용합니다.

· 출력 리미트 기능(C의 비트 12~15)을 유효로 하면 출력이 출력 리미트 하한값(C+2)과 출력 리미트 상한값(C+3)의 범위 밖이 되었을 때, 출력을 제한(포화)시킬 수 있습니다.

사용상의 주의 사항

· 사이클릭 태스크 내에서 이 명령을 PID 명령과 조합하여 사용하고 인터럽트 태스크를 사용하는 경우, PID 명령 및 이 명령 전에 DI(인터럽트 태스크 실행 금지)를 실행하여 인터럽트 태스크의 실행을 금지하십시오. PID 명령과 이 명령 사이에 인터럽트가 들어오면 제어 주기가 어긋날 수 있습니다.

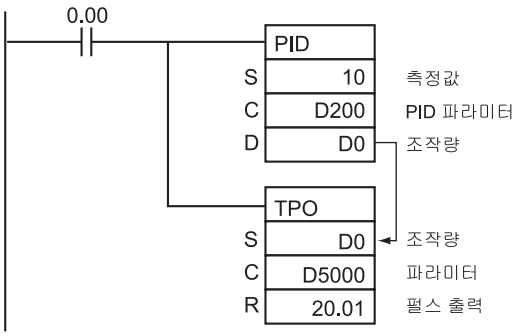


샘플 프로그램

■ PID 명령과 조합하여 사용하는 경우

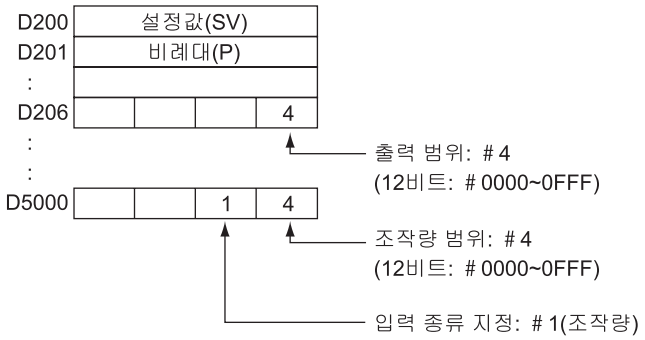
0.00이 ON인 동안 PID 연산 명령에서 출력 조작량(D0에 저장)을 입력하고, 이를 바탕으로 듀티비를 계산(조작량÷조작량 범위)한 후, 그 결과를 시분할 비례 출력으로 변환하여 20CH 비트 01에 펄스 출력합니다.

또한 트랜지스터 출력 유니트를 20CH로 할당해 두고 그 비트 01의 단자에 SSR(솔리드 스테이트 릴레이)를 연결하여 히터 제어합니다.



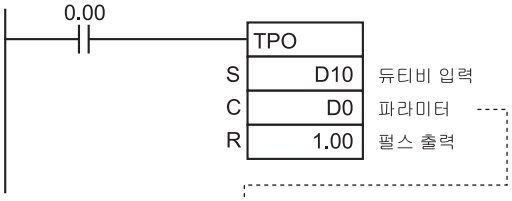
00000의 OFF→ON으로 기동 시 파라미터를 전송하고 10CH를 측정값(PV)으로 하여 PID 연산을 시작하고 조작량(MV) D0에 저장합니다.

D0에 저장된 조작량(MV)을 조작량 범위(예: 12비트 지정 시 #0FFF)로 나눈 비율을 듀티비로 하고, 그것을 시분할 비례 출력으로 변환하여 20CH 비트 01에 출력합니다.



■ 이 명령 단독으로 사용하는 경우

0.00이 ON인 동안 D10 내의 듀티비를 입력하고, 그 값을 시분할 비례 출력으로 변환하여 1CH 비트 00에 펄스 출력합니다. 제어 주기 = 1초, 출력 리미트 기능 유효(하한 20.00%, 상한 80.00%) 시.

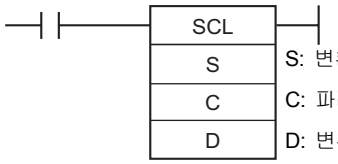


D10에 저장된 듀티비를 입력하고, 그것을 시분할 비례 출력으로 변환하여 1CH 비트 00에 출력합니다.

D0	1 1 0 0	듀티비 지정, 명령 실행 시 전송, 출력 리미트 기능 유효
D1	&100(#0064)	제어 주기 = 1.00초
D2	&200(#07D0)	출력 리미트 하한=20.00%
D3	&8000(#1F40)	출력 리미트 상한=80.00%
D4	설정 안 함	
D5	설정 안 함	
D6	설정 안 함	
:		
:		
D10	&0~&10000 (#0000~2710)	0 ~ 100.00%

SCL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
스케일링	SCL	@SCL	194	지정한 1차 함수에 따라 부호 없는 BIN 데이터를 부호 없는 BCD 데이터로 스케일링(변환)합니다.

심볼	SCL
	

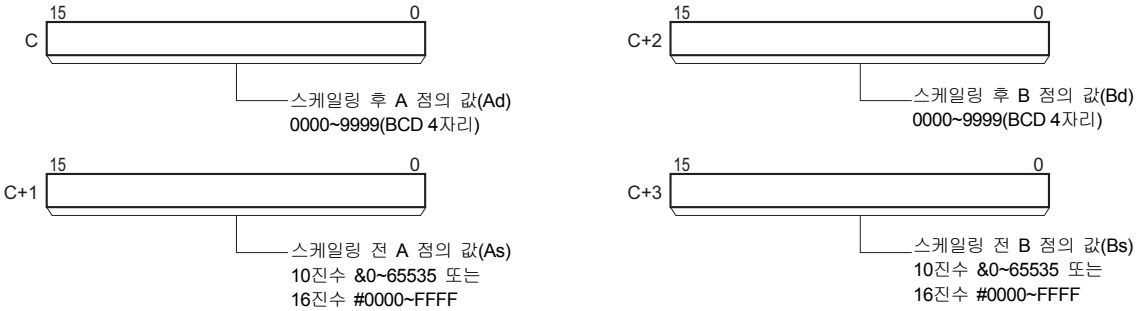
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	변환 대상 CH 번호	UINT	1
C	파라미터 저장 하위 CH 번호	LWORD	4
D	변환 결과 저장 CH 번호	WORD	1

C: 파라미터 저장 하위 CH 번호



주: C~C+3은 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
C												–						
D												○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· Ad(C)와 Bd(C+2)의 값이 BCD가 아닐 때 ON · As(C+1)와 Bs(C+3)의 값이 같을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과가 0일 때 ON · 그 이외일 때 OFF

기능

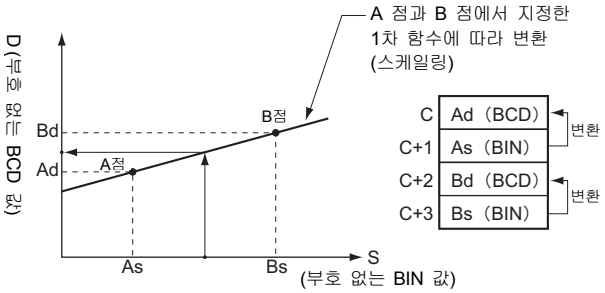
S로 지정한 부호 없는 BIN 데이터를 C로 지정한 파라미터(A 점 및 B 점의 스케일링 전후의 각 값)에 의해 결정되는 일차 함수에 따라 부호 없는 BCD 데이터로 변환하여 결과를 D에 출력합니다.

변환식: $D = Bd - \frac{(Bd - Ad)}{(Bs - As)\text{의 BCD 변환 값}} \times (Bs - S)\text{의 BCD 변환 값}$

A 점과 B 점은 순기울기뿐만 아니라 역기울기에서도 가능합니다. 따라서 역스케일링이 가능합니다.

· 변환 결과에서 소수점 이하의 데이터는 반올림됩니다.

· 변환 결과가 0000보다 작은 경우는 0000을 출력하고 9999보다 큰 경우는 9999를 출력합니다.



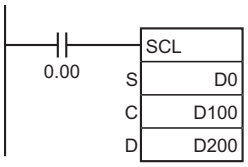
힌트

· 이 명령은 아날로그 입력 유니트의 아날로그 신호 변환 결과를 사용자 정의 스케일로 변환하는 경우 등에 사용합니다(예: 1~5V에 대한 10진수 &0~4000 또는 16진수 #0000~0FA0을 50~200℃로 스케일링).

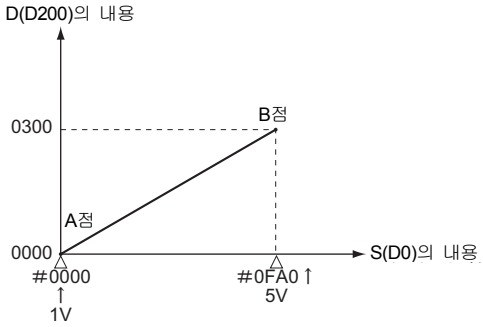
· 이 명령은 부호 없는 BIN을 부호 없는 BCD로 스케일링(변환)합니다.
따라서 원래 데이터 S에 음수가 있는 경우, 그에 대해 값을 증가(최대 마이너스 값을 더함)시킨 후 스케일링하도록 프로그램을 작성해야 합니다.
또한 스케일링 결과 D에는 음수를 출력할 수 없습니다. 스케일링(변환) 결과가 음수인 경우 0이 출력됩니다.

샘플 프로그램

아날로그 신호 1~5V에 대해 16진수 #0000~0FA0의 값이 D0에 저장되면 그 값을 0~300의 BCD 값으로 변환(스케일링)합니다.



0.00이 ON이면 아날로그 입력 유니트의 D0의 값을 A 점(#0000→0(BCD))과 B 점(＃0FA0→300(BCD))에 의해 결정되는 1차 함수에 따라 스케일링하여 D200에 저장합니다.



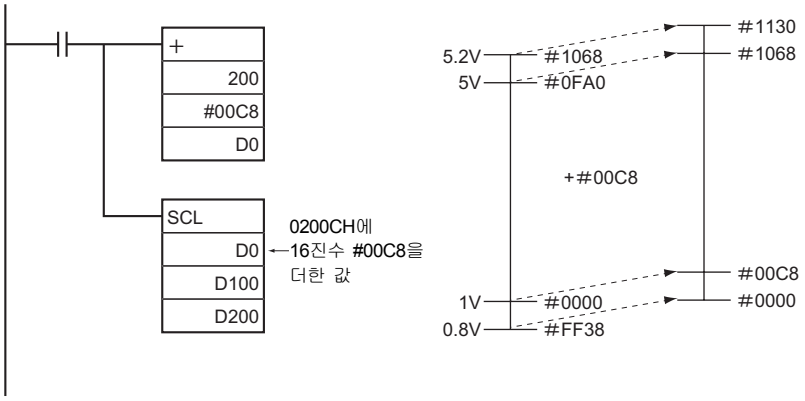
C : D100	0 0 0 0	Ad (BCD)
C+1 : D101	0 0 0 0	As (BIN)
C+2 : D102	0 3 0 0	Bd (BCD)
C+3 : D103	0 F A 0	Bs (BIN)

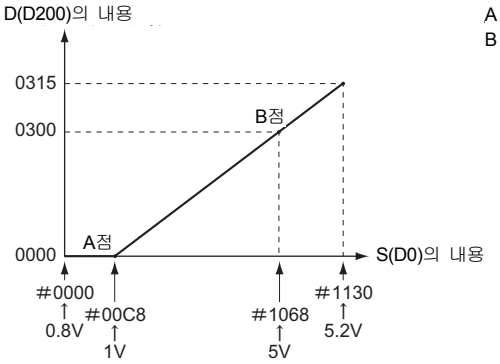
참고

실제 아날로그 입력 유니트의 변경값은 0.8V~5.2V에 대해 16진수 #FF38~1068의 값이 저장됩니다.

그러나 SCL 명령은 S의 값을 16진수 #0000~FFFF의 부호 없는 16진수 값으로 취급하기 때문에 1V(#0000) 미만의 부호 있는 16진수 값 #FF38~FFFF에 대해 정확하게 스케일링할 수 없습니다.

따라서 실제로 아날로그 입력 유니트의 변환값을 SCL 명령으로 스케일링하는 경우, 16진수 #FF38이 #0000이 되도록 전체에 16진수 #00C8을 더한 후 SCL 명령을 실행합니다.



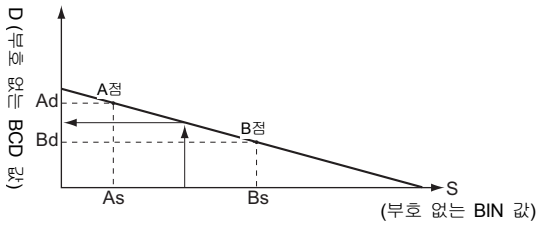


A 점(16진수 #00C8→0000(BCD))
B 점(16진수 #1068→0300(BCD))

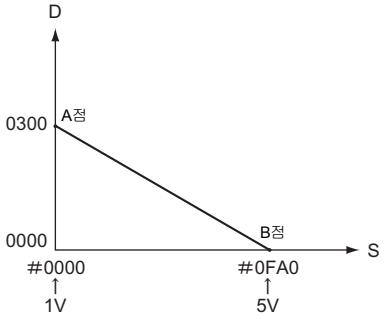
C : D100	0	0	0	0	Ad (BCD)
C+1 : D101	0	0	C	8	As (BIN)
C+2 : D102	0	3	0	0	Bd (BCD)
C+3 : D103	1	0	6	8	Bs (BIN)

단, SCL 명령의 경우 D의 값에는 0~9999의 부호 없는 BCD 값을 저장합니다. 따라서 0.8~5.2V에 대해 원래 −15~+315의 스케일링을 해야 하지만 변환 결과가 음수인 경우, 0(BCD)을 저장하므로 위의 그림과 같이 16진수 #0000~00C8에 대해서는 0(BCD)으로 일정해집니다.

참고: 역스케일링에 대해 As < Bs, Ad > Bd도 가능합니다.

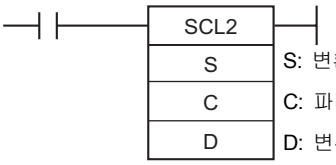


이 경우, 예를 들어 1~5V(16진수 #0000~0FA0)에 대해 0300~0000의 BCD로 변환(역스케일링)하는 것이 가능합니다.



SCL2

명령 이름	니모닉	동작 옵션	평션 번호	기능
스케일링 2	SCL2	@SCL2	486	지정한 오프셋이 있는 1차 함수에 따라 부호 있는 BIN 데이터를 부호 있는 BCD 데이터로 스케일링(변환)합니다.

	SCL2
심볼	

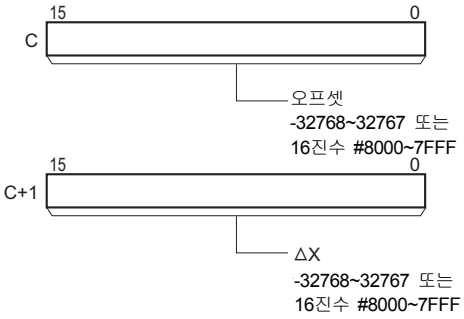
특정 영역 내의 사용 여부

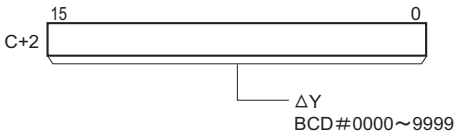
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	변환 대상 CH 번호	INT	1
C	파라미터 저장 하위 CH 번호	WORD	3
D	변환 결과 저장 CH 번호	WORD	1

C: 파라미터 저장 하위 CH 번호





주: C~C+2는 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
C												–						
D												○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· ΔX(C+1)의 데이터가 0일 때 ON · ΔY(C+2)의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과가 0일 때 ON · 그 이외일 때 OFF

이름	라벨	내용
캐리 플래그	P_CY	· 변환 결과가 음수가 되었을 때 ON · 그 이외일 때 OFF

기능

S로 지정한 부호 있는 BIN 데이터를 C로 지정한 파라미터(기울기와 오프셋)로 결정되는 1차 함수에 따라 부호 있는 BCD 데이터(BCD 데이터는 절대값이고, CY 플래그로 음/양 판별. ON: 음, OFF: 양)로 변환하여 결과를 D에 출력합니다.

변환식: $D = \frac{\Delta Y}{\Delta X\text{의 BCD 변환값}} X \{(S\text{의 BCD 변환값}) - (\text{오프셋의 BCD 변환값})\}$

주 : $\frac{\Delta Y}{\Delta X}$ 가 기울기입니다.

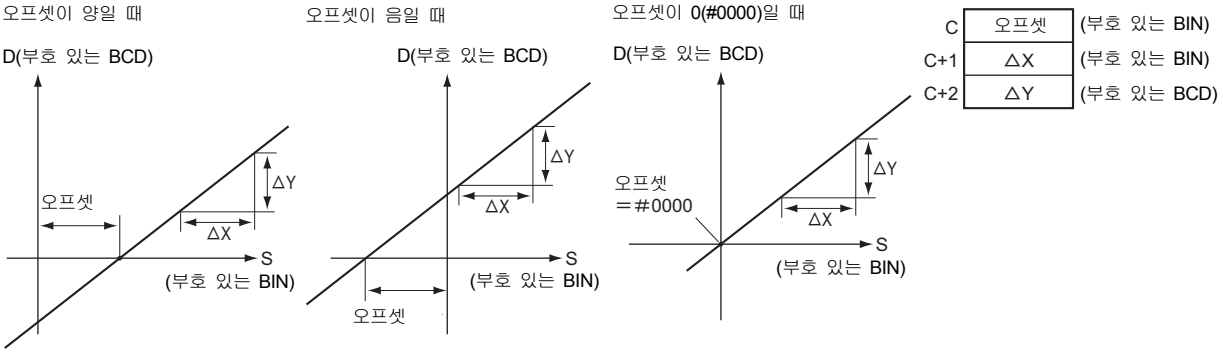
오프셋은 양수, 0, 음수가 가능합니다.

기울기도 양수, 0, 음수가 가능합니다. 따라서 역스케일링도 가능합니다.

· 변환 결과에서 소수점 이하의 데이터는 반올림됩니다.

· D(BCD 데이터)는 절대값을 나타내며 캐리 플래그로 음과 양을 나타냅니다. 따라서 변환 결과는 -9999~9999의 범위에서 출력됩니다.

· 변환 결과가 상한(9999)을 초과하는 경우 9999를 출력하고, 하한을 초과하는 경우는 -9999를 출력합니다.



힌트

· 이 명령은 아날로그 입력 유니트의 아날로그 신호 변환 결과를 사용자 정의 스케일로 변환하는 경우 등에 사용합니다(예: 1~5V에 대한 10진수 &0~4000 또는 16진수 #0000~0FA0을 -100~200℃로 스케일링).

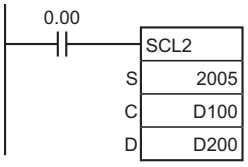
· 이 명령은 부호 있는 BIN을 부호 있는 BCD로 스케일링(변환)합니다.
따라서 원래 데이터 S에 음수가 있는 경우에도 그대로 스케일링할 수 있습니다.
또한 스케일링 결과 D와 캐리 플래그에서 스케일 결과에 음수를 출력하는 것도 가능합니다.

SCL2

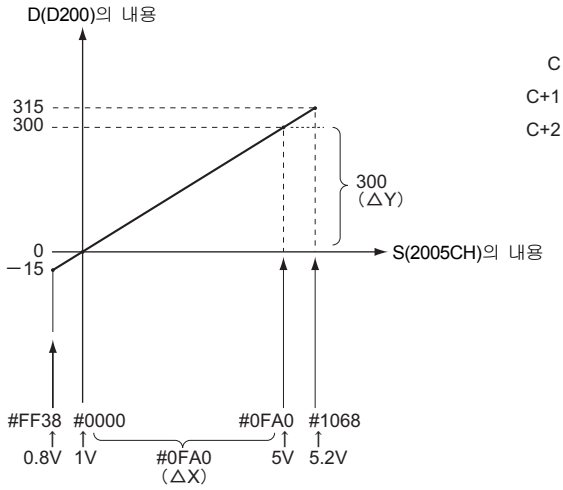
샘플 프로그램

■ 아날로그 신호 1~5V에 대해 0~300의 스케일링을 할 때

아날로그 신호 1~5V에 대해 부호 있는 16진수 #0000~0FA0의 값이 2005CH에 저장되면 그 값을 0~300의 BCD 값으로 변환(스케일링)합니다.



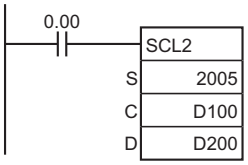
0.00이 ON이면 아날로그 입력 유니트의 2005CH의 값을 오프셋 = #0000, △X = 16진수 #0FA0, △Y = BCD #0300에 의해 결정되는 1차 함수에 따라 스케일링하여 D200에 저장합니다.



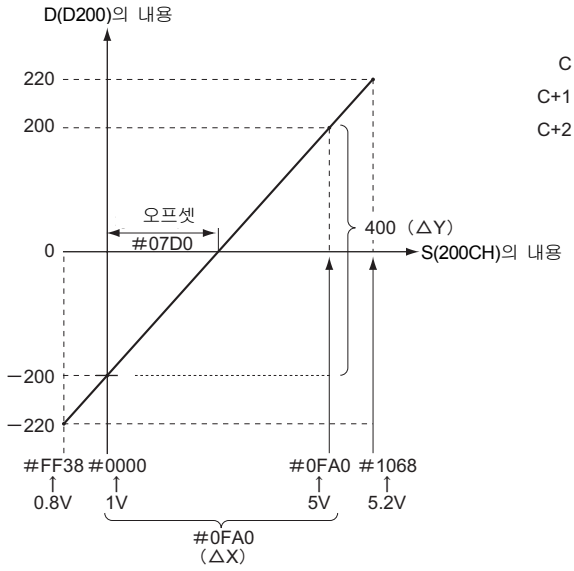
C : D100	0	0	0	0	오프셋
C+1 : D101	0	F	A	0	△X
C+2 : D102	0	3	0	0	△Y

■ 아날로그 신호 1~5V에 대해 -200~+200을 스케일링할 때

아날로그 신호 1~5V에 대해 부호 있는 BIN 값 10진수 &0~4000 또는 16진수 #0000~0FA0의 값이 2005CH에
저장되면 그 값을 -0200~0200의 BCD 값으로 변환(스케일링)합니다.



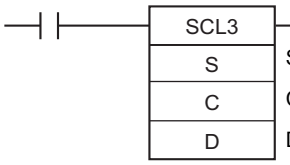
0.00이 ON이면 아날로그 입력 유니트의 2005CH의 값을 오프셋 = 16진수 #07D0 △X = 16진수 #0FA0 △Y = BCD #0400에 의해 결정되는 1차 함수에 따라 스케일링하여 D200에 저장합니다.



C : D100	0	7	D	0	오프셋
C+1 : D101	0	F	A	0	△X
C+2 : D102	0	4	0	0	△Y

SCL3

명령 이름	니모닉	동작 옵션	펑션 번호	기능
스케일링 3	SCL3	@SCL3	487	지정한 오프셋이 있는 1차 함수에 따라 부호 있는 BCD 데이터를 부호 있는 BIN 데이터로 스케일링(변환)합니다.

심볼	SCL3
	

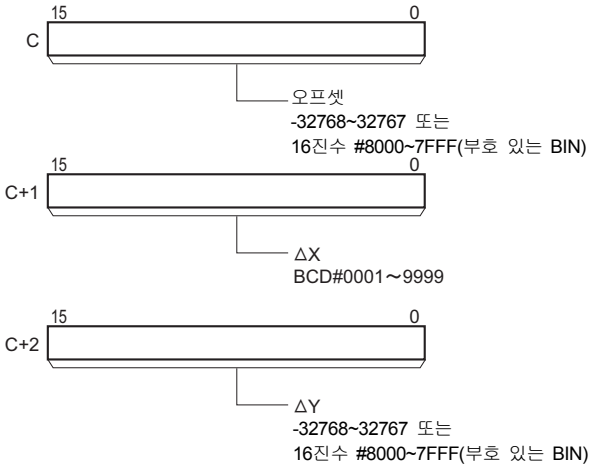
특정 영역 내의 사용 여부

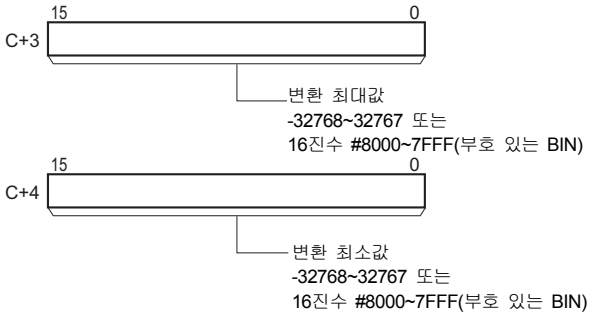
영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	변환 대상 CH 번호	WORD	1
C	파라미터 저장 하위 CH 번호	WORD	5
D	변환 결과 저장 CH 번호	INT	1

C: 파라미터 저장 하위 CH 번호





주: C~C+4는 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
C												–						
D												○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S의 데이터가 BCD가 아닐 때 ON · ΔX(C+1)의 데이터가 0001~9999의 BCD가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 변환 결과가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 변환 결과의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

기능

S로 지정한 부호 있는 BCD 데이터(BCD 데이터는 절대값이고 CY 플래그로 음양 판별. ON: 음 OFF: 양)를 C로 지정한 파라미터(기울기와 오프셋)로 결정되는 1차 함수에 따라 부호 있는 BIN 데이터로 변환하여 결과를 D에 출력합니다.

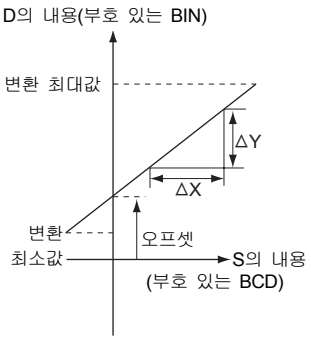
CY 플래그는 STC 명령(040)/CLC 명령(041)으로 ON/OFF하십시오.

변환식: $D = \frac{\Delta Y}{\Delta X\text{의 BCD 변환값}} \times (S\text{의 BIN 변환값}) + (\text{오프셋})$

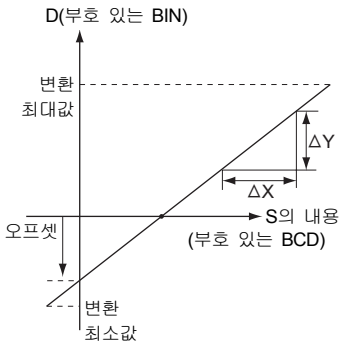
주 : $\frac{\Delta Y}{\Delta X}$ 가 기울기입니다.

- 오프셋은 양수, 0, 음수가 가능합니다.
- 기울기도 양수, 0, 음수가 가능합니다. 따라서 역스케일링도 가능합니다.
- S의 BCD 데이터는 절대값을 나타내며 명령 실행 시의 캐리(CY) 플래그에 의해 음과 양을 구별합니다. 따라서 변환 대상 데이터는 −9999~9999의 범위가 됩니다.
- 변환 결과에서 소수점 이하의 데이터는 반올림됩니다.
- 변환 결과가 변환 최대값(C+3)을 초과하는 경우 변환 최대값을 출력하고, 변환 최소값(C+4)을 초과하는 경우는 변환 최소값을 출력합니다.

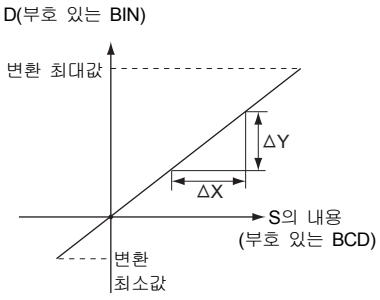
오프셋이 양일 때



오프셋이 음일 때



오프셋이 0(#0000)일 때

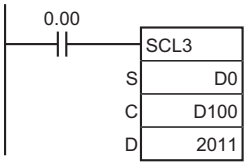


힌트

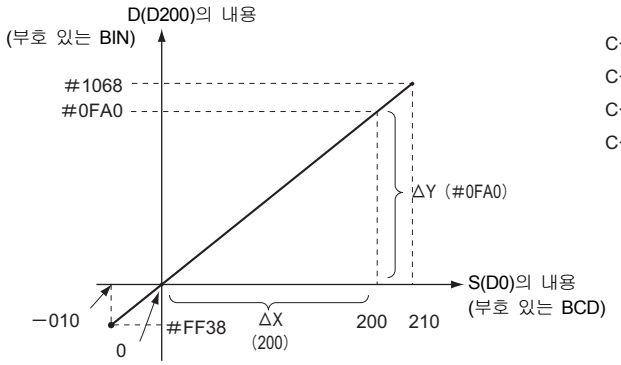
- 이 명령은 사용자 정의 스케일을 아날로그 출력 유니트용 부호 있는 BIN 데이터로 변환하는 경우 등에 사용합니다(예: 0~200℃를 10진수 &0~4000 또는 16진수 #0000~0FA0으로 변환하여 아날로그 출력 신호 1~5V를 아날로그 출력 유니트에서 출력).

샘플 프로그램

0~200의 값을 아날로그 신호(예를 들어 1~5V)로 스케일링할 때, 부호 있는 BCD 값 0~200의 값을 아날로그 출력 유니트용 데이터 부호 있는 BIN 값 0~0FA0으로 변환(스케일링)합니다.



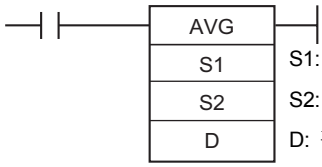
0.00이 ON이면 D0의 값을 오프셋 = #0000, △X = BCD #0200, △Y = 16진수 #0FA0에 의해 결정되는 1차 함수에 따라 스케일링하여 2011CH에 저장합니다.



C : D100	0	0	0	0	오프셋
C+1 : D101	0	2	0	0	△X
C+2 : D102	0	F	A	0	△Y
C+3 : D103	1	0	6	8	변환 최대값
C+4 : D104	F	F	3	8	변환 최소값

AVG

명령 이름	니모닉	동작 옵션	평션 번호	기능
데이터 평균화	AVG	–	195	CH 데이터의 지정한 사이클 횟수 분량의 평균값을 BIN으로 계산합니다.

심볼	AVG
	 S1: 현재값 입력 CH 번호(대상 CH) S2: 평균값 계산 사이클 횟수 D: 평균값 저장 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

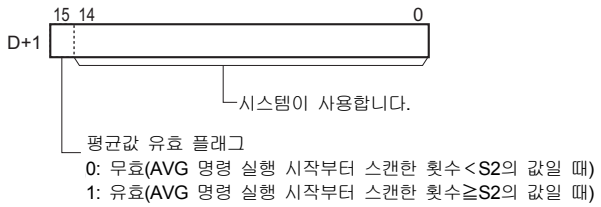
피연산자	내용	데이터 형식	크기
S1	현재값 입력 CH 번호(대상 CH)	UINT	1
S2	평균값 계산 사이클 횟수	UINT	1
D	평균값 저장 하위 CH 번호	UINT	가변

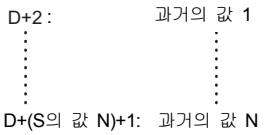
S2: 평균값 계산 사이클 횟수

10진수 &1~64 또는 16진수 #0001~0040

D: 평균값 저장 하위 CH 번호

D+1: 작업 데이터(사용자 쓰기 불가능)





주: D~D+(S의 값N)+1은 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S1, S2	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
D											–	–						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S2의 데이터가 0일 때 ON · 그 이외일 때 OFF

기능

S1로 지정한 부호 없는 BIN 데이터(10진수 &0~65535 또는 16진수 #0000~FFFF)를 지정한 사이클 횟수(S2)만큼 저장 포인터(D+1의 비트 00~07)를 갱신하면서 과거의 값으로 D+2 이후에 순차 저장합니다.

그 동안 S1의 데이터는 그대로 D에 출력하고, 평균값 유효 플래그(D+1의 비트 15)를 0(OFF)으로 설정합니다.

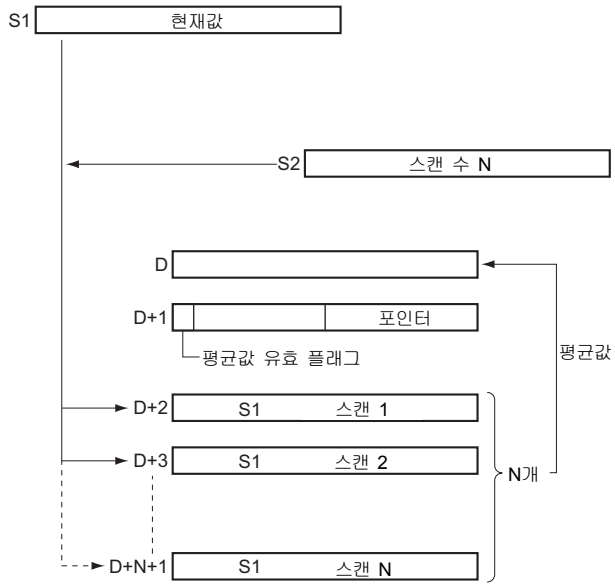
지정된 사이클 횟수(S2) 분량의 과거의 S1 값이 D+2 이후에 저장되면 해당 과거 값의 평균값을 연산하고 결과를 부호 없는 BIN 데이터로 D에 출력합니다. 이때, 평균값 유효 플래그(D+1의 비트 15)를 1(ON)로 설정합니다.

이후에는 스캔할 때마다 최신 S2 스캔 분량의 데이터를 사용하여 평균값이 계산되어 D에 출력됩니다.

지정된 사이클 횟수(S2)는 최대 64입니다.
64를 초과하는 값을 지정한 경우는 64로 동작합니다.

과거 값의 저장 포인터는 S2-1까지 도달하면 다시 0부터 시작됩니다.

평균값에서 소수점 이하의 데이터는 반올림됩니다.

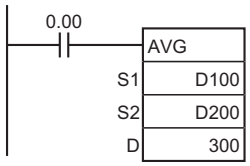


사용상의 주의 사항

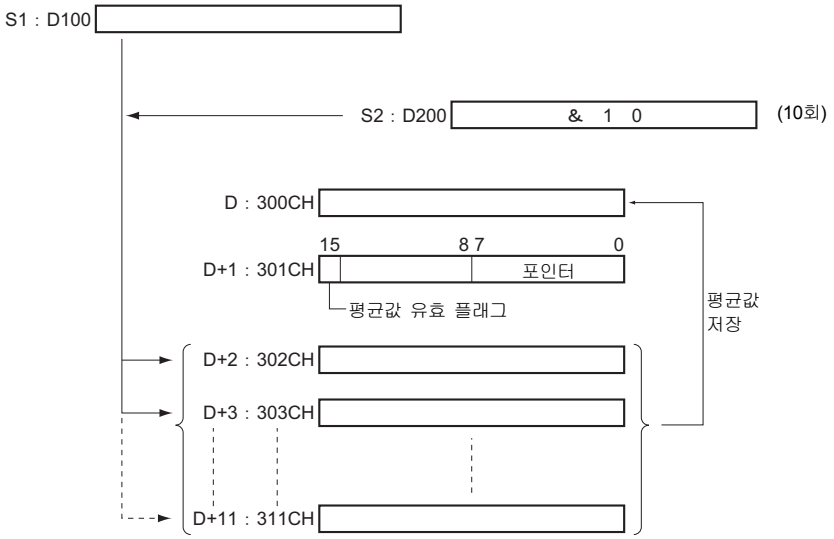
첫 회 입력 조건이 기동되면 이 명령에 의해 작업 데이터(D+1)를 초기화(0으로 클리어)합니다.

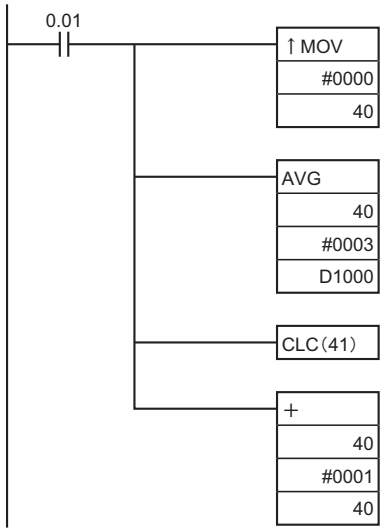
단, 운전 시작 1사이클째인 경우 이 명령은 명령 실행 시 작업 데이터(D+1)를 초기화하지 않습니다.
따라서 운전 시작 1사이클째부터 이 명령을 실행하는 경우는 프로그램을 통해 D+1을 클리어하십시오.

샘플 프로그램

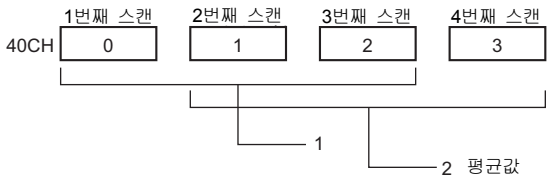


0.00이 ON이면 D100의 내용을 D200의 내용(예: &10 즉, 10회)의 스캔 분량만큼 302~311CH의 10CH에 순차 저장합니다. 해당 10CH 분량의 평균값을 300CH에 저장하고 301CH의 비트 15를 ON으로 합니다.





- 입력 0.01이 ON이 되면 40CH의 내용은 0부터 스캔할 때마다 1씩 가산됩니다.
- 1, 2번째 스캔에서 AVG 명령은 40CH의 내용을 D1002, D1003에 저장하고 D1001도 갱신합니다. (40CH의 내용을 가산하는 것은 AVG 명령의 결과가 변경되는 것을 확인할 수 있도록 하려는 것으로, 특별한 의미는 없습니다.)
- 3번째 스캔에서 AVG 명령은 최신 평균값을 계산하여 D1000에 저장합니다.



D1000	0	1	1	2	평균값
D1001	1	2	8000	8001	작업용 데이터
D1002	0	0	0	3	과거의 전송
D1003	–	1	1	1	데이터 저장 영역
D1004	–	–	2	2	

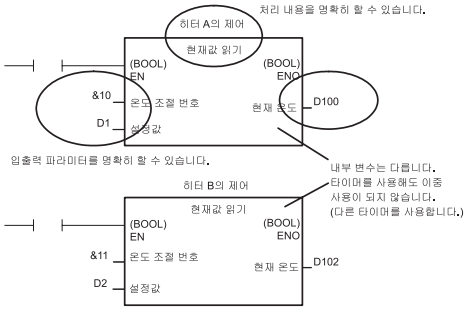
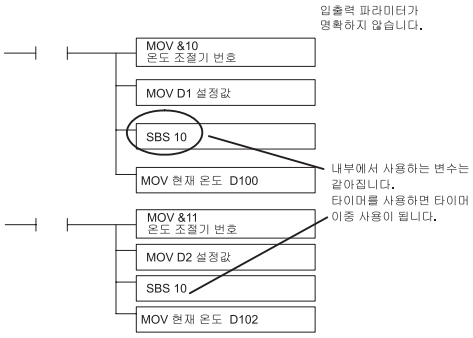
서브루틴 명령

니모닉	명령 이름	펑션 번호	페이지
SBS	서브루틴 호출	091	3-561
MCRO	매크로	099	3-566
SBN	서브루틴 진입	092	3-569
RET	서브루틴 반환	093	
GSBS	글로벌 서브루틴 호출	750	3-572
GSBN	글로벌 서브루틴 진입	751	3-578
GRET	글로벌 서브루틴 반환	752	

서브루틴 명령

CS/CJ 시리즈에는 프로그램의 구조화 수단으로 펑션 블록과 서브루틴 기능이 있습니다.
각각의 장단점은 다음과 같습니다.

■ 펑션 블록과 서브루틴의 차이

	펑션 블록	서브루틴
사용 가능한 CPU 유니트의 유니트 버전	CS1-H, CJ1-H, CJ1M CPU 유니트: 유니트 버전 3.0 이상 CJ2 CPU 유니트: 모든 유니트 버전	모든 유니트 버전
처리 이름	○ 인스턴스명을 붙일 수 있습니다.	× 이름을 붙일 수 없습니다.
입출력의 명확화	○ FB의 입력 변수/출력 변수로 명확화할 수 있습니다.	× 명확화할 수 없습니다.
변수 이름	○ 변수에 이름을 붙일 수 있습니다.	○ 변수명을 붙일 수 있습니다.
내부 변수	○ 있음 이름이 같은 인스턴스는 같은 변수에 액세스하고 이름이 다른 인스턴스는 다른 변수에 액세스합니다. 따라서 펑션 블록 내부에서 호출원(인스턴스)별로 데이터를 유지할 수 있습니다. 펑션 블록 내에서 타이머를 사용해도 이중 사용이 되지 않습니다.	× 없음 모든 서브루틴이 같은 변수에 액세스합니다. 따라서 서브루틴에서 호출원별로 데이터를 유지할 수 없습니다. 서브루틴 내에서 타이머를 사용하면 타이머 이중 사용이 됩니다.
프로그램 기술 방법	○ 래더 또는 ST 언어를 사용한 변수 프로그래밍	× 래더를 사용한 주소 프로그래밍
그림 설명	○ 처리 내용이나 입출력 파라미터가 알기 쉬워집니다. (시인성을 높일 수 있습니다.) 	× 래더를 사용한 주소 프로그래밍 
보호	○ 펑션 블록별로 읽기 금지를 설정할 수 있습니다. (CX-Programmer 6.1 이상에서 가능)	× 보호 기능은 없습니다.
저장	○ 펑션 블록 단위로 저장할 수 있습니다.	× 서브루틴 단위로 저장할 수 없습니다.

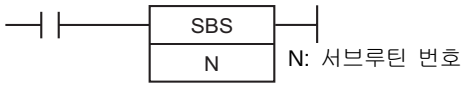
또한 서브루틴 기능은 일반적인 서브루틴과 글로벌 서브루틴의 2종류가 있습니다.
각각의 차이는 다음과 같습니다.

■ 일반적인 서브루틴과 글로벌 서브루틴의 차이

	일반적인 서브루틴	글로벌 서브루틴
기능	동일 태스크 내에 있는 서브루틴을 호출할 수 있습니다. 예를 들어, 태스크 번호 0에 있는 서브루틴은 태스크 번호 1부터 호출할 수 없습니다.	모든 태스크에서 서브루틴을 호출할 수 있습니다. 인터럽트 태스크 번호 0에 있는 서브루틴은 어느 태스크에서나 호출할 수 있습니다.
조합 사용 명령	SBS(서브루틴 호출) 명령 SBN(서브루틴 진입) 명령 RET(서브루틴 반환) 명령	GSBS(글로벌 서브루틴 호출) 명령 GSBN(글로벌 서브루틴 진입) 명령 GRET(글로벌 서브루틴 반환) 명령
서브루틴 프로그램 기술	서브루틴을 사용하는 태스크의 마지막(END 명령 앞)에 기술합니다.	인터럽트 태스크 번호 0의 마지막(END 명령 앞)에 기술합니다.

SBS

명령 이름	니모닉	동작 옵션	펑션 번호	기능
서브루틴 호출	SBS	@SBS	091	지정된 번호의 서브루틴을 호출하는 프로그램을 실행합니다.

심볼	SBS
	

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	가능	가능	가능	가능	불가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	서브루틴 번호*1	–	1

N: 서브루틴 번호
10진수 0~1023*1

*1: CJ1M-CPU11/21의 경우 10진수 0~255

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–

조합 사용 명령

SBN(서브루틴 진입) 명령과 RET(서브루틴 반환) 명령

조건 플래그의 동작

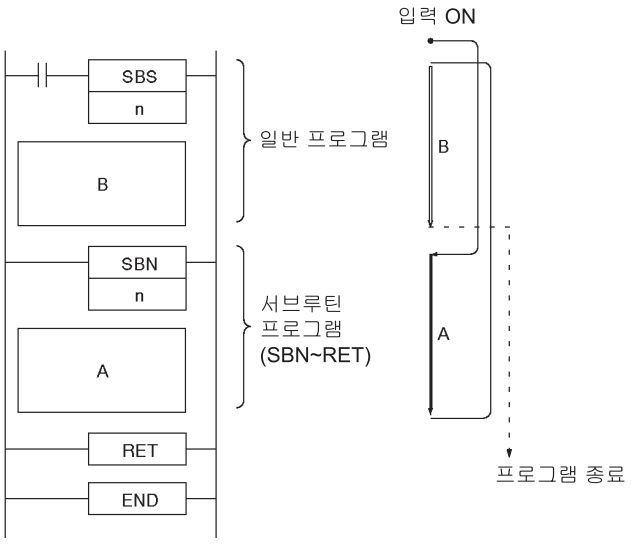
이름	라벨	내용
에러 플래그	P_ER	· 중첩 수가 16단을 초과했을 때 ON · 지정 대상 서브루틴이 존재하지 않을 때 ON · 서브루틴이 자신을 호출했을 때 ON · 실행 중인 서브루틴을 호출했을 때 ON · 지정 대상 서브루틴이 자체 태스크 내에서 정의되지 않았을 때 ON · 그 이외일 때 OFF

SBS

기능

N으로 지정한 번호의 서브루틴 영역(SBN 명령~RET 명령 사이의 영역)의 프로그램을 호출합니다. 서브루틴 영역을 실행 종료한 후 이 명령의 다음 명령으로 돌아갑니다.

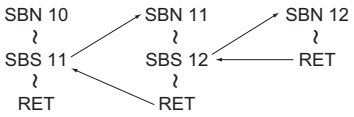
동일한 서브루틴을 여러 번 호출할 수도 있습니다.

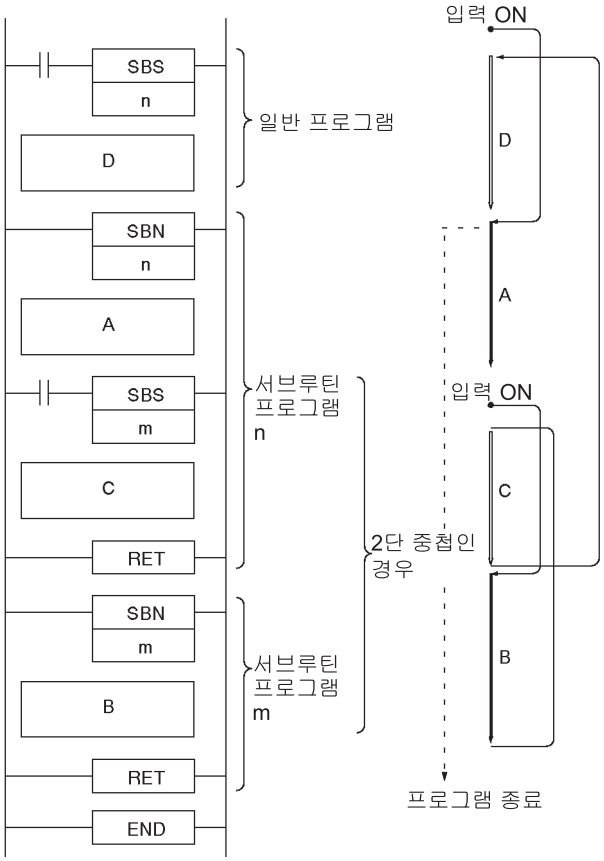


서브루틴은 16단까지 중첩할 수 있습니다.

중첩이란 서브루틴 프로그램(SBN~RET) 안에 서브루틴 호출(SBS)이 삽입된 상태를 말합니다.

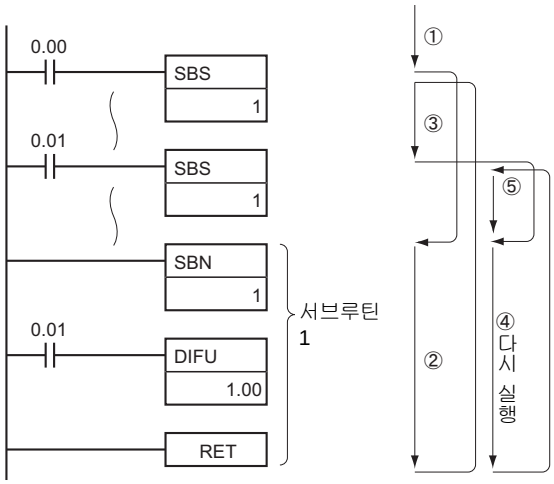
예)



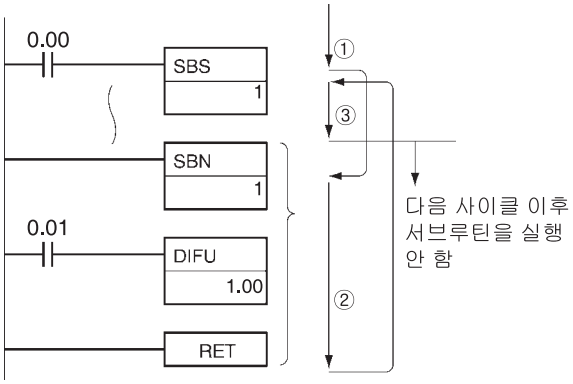


사용상의 주의 사항

- 서브루틴 번호는 하나의 서브루틴에 할당되는 유일한 번호이므로 번호를 중복하여 사용하지 마십시오.
- 미분형 명령(UP, DOWN, DIFU, DIFD, 동작 옵션 @, %가 있는 명령)을 서브루틴 내에서 사용하는 경우 다음과 같은 점에 주의해야 합니다.
  - 동일한 사이클 내에서 동일한 서브루틴을 여러 번 실행한 경우 서브루틴 내의 입력 미분형 명령(DIFU, DIFD, 동작 옵션 @, %가 있는 명령)의 동작이 일정치 않으므로 주의하십시오.
  - 다음 예에서는 입력 릴레이 0.00이 ON되면 서브루틴 1이 실행되고, 입력 릴레이 0.01이 OFF→ON이 된 경우에는 DIFU 명령에 의해 1.00이 ON됩니다. 그러나 동일한 사이클 내에서 입력 릴레이 0.01이 ON된 경우, 서브루틴 1이 다시 실행되므로 이번에는 DIFU 명령이 0.01의 기동을 검출하지 못하여 1.00이 OFF가 됩니다.



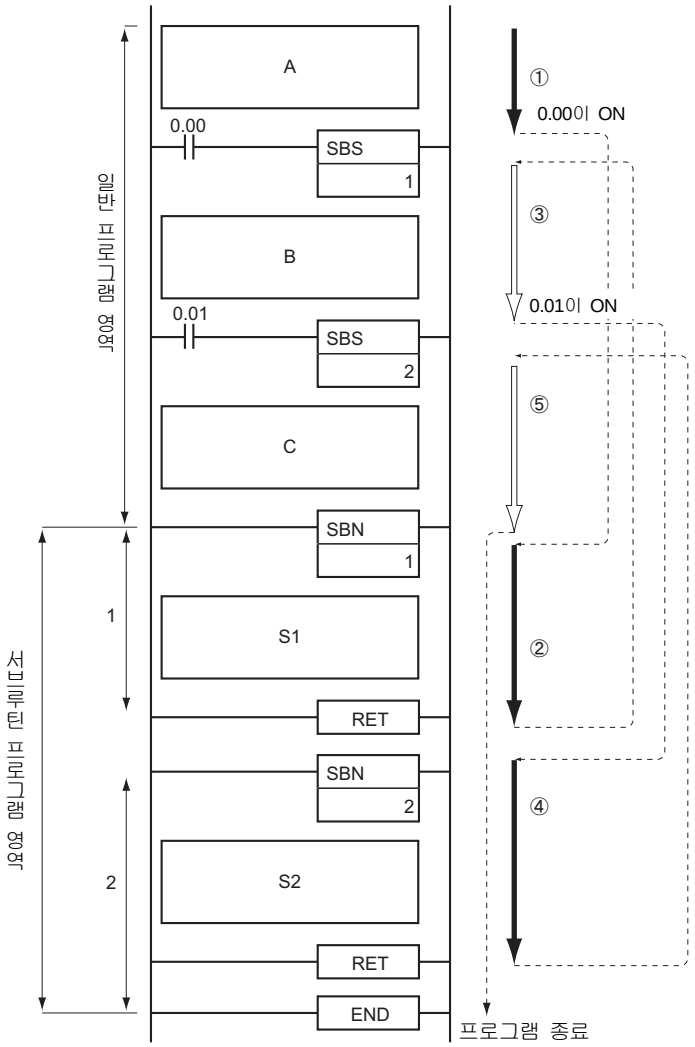
  - 이와 반대로 서브루틴 내의 미분 명령(DIFU, DIFD 명령)이 실행되고 그 출력이 ON된 상태에서 같은 서브루틴을 다음 번부터 호출하지 않은 경우, 미분 명령(DIFU, DIFD 명령)의 출력은 ON된 그대로 OFF되지 않으므로 주의하십시오.
  - 다음 예에서는 0.00이 ON되면 서브루틴 1이 실행되고, 0.01이 OFF→ON이 된 경우에는 DIFU 명령에 의해 1.00이 ON됩니다. 이후의 사이클에서 0.00이 OFF인 상태에서 서브루틴 1이 실행되지 않으면 DIFU 명령의 출력 1.00은 ON 그대로 유지됩니다.
  - IL-ILC 명령에 의해 인터록 중에는 SBS 명령이 NOP 처리됩니다.



샘플 프로그램

■ 2개의 서브루틴 호출 명령이 있을 때

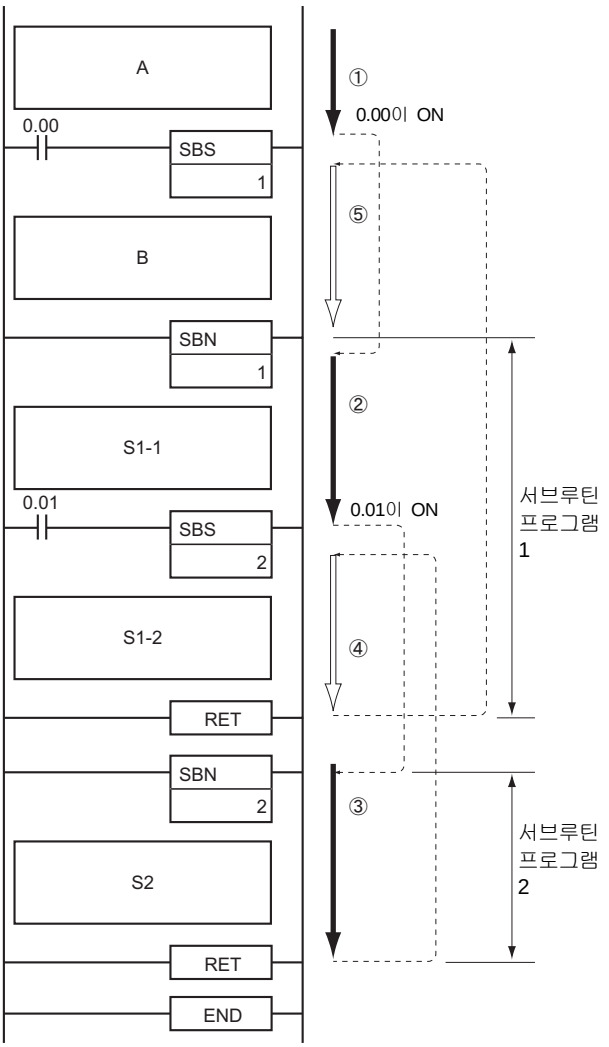
・ 중첩하지 않은 경우



0.00이 ON이면 서브루틴 번호 1의 서브루틴 프로그램을 실행한 후, SBS1 명령의 다음 명령으로 돌아갑니다.
0.01이 ON이면 서브루틴 번호 2의 서브루틴 프로그램을 실행한 후, SBS2 명령의 다음 명령으로 돌아갑니다.

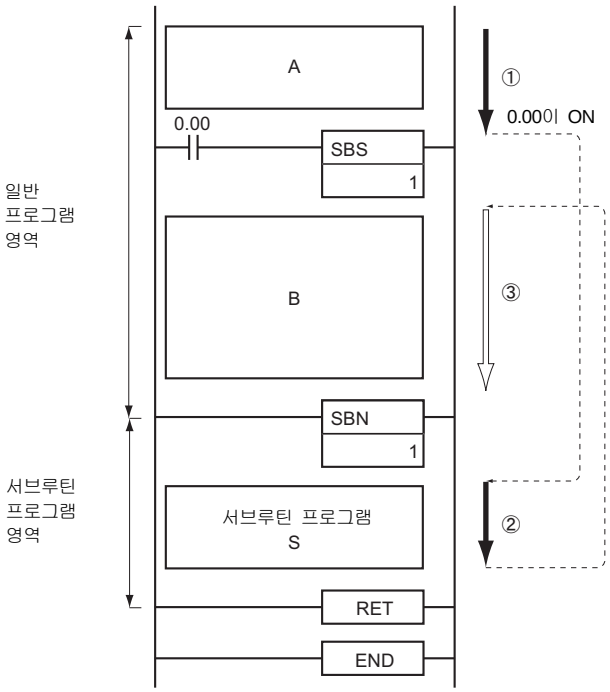
0.00	0.01	실행 순서
ON	ON	A→S1→B→S2→C
ON	OFF	A→S1→B→C
OFF	ON	A→B→S2→C
OFF	OFF	A→B→C

・ 2단 중첩인 경우



0.00이 ON이면 서브루틴 번호 1의 서브루틴 프로그램을 실행하고, 0.01이 ON이면 서브루틴 번호 2의 서브루틴 프로그램을 실행한 후, SBS2의 다음 명령으로 돌아갑니다. SBN2 직전까지의 명령을 실행한 후, SBS1의 다음 명령으로 돌아갑니다. SBN1 직전의 명령까지 실행하고 종료합니다.

0.00	0.01	실행 순서
ON	ON	A→S1-1→S2→S1-2→B
ON	OFF	A→S1-1→S1-2→B
OFF	ON	A→B
OFF	OFF	A→B



0.00이 ON이면 서브루틴 번호 1의 서브루틴 프로그램을 실행한 후, 이 명령의 다음 명령으로 돌아갑니다. SBN1 직전의 명령까지 실행하고 종료합니다.

0.00	실행 순서
ON	A→S→B
OFF	A→B

MCRO

명령 이름	니모닉	동작 옵션	평션 번호	기능
매크로	MCRO	@MCRO	099	인수가 있는 서브루틴 호출입니다.

심볼	MCRO
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	가능	가능	가능	가능	불가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	서브루틴 번호	–	1
S	인수 데이터 하위 CH 번호	–	4
D	반환값 데이터 하위 CH 번호	–	4

N: 서브루틴 번호
10진수 0~1023*1

*1: CJ1M-CPU11/21의 경우 10진수 0~255

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
S	○	○	○	○	○	○	○	○	○	○	–			○				
D																		

조합 사용 명령

SBN(서브루틴 진입) 명령과 RET(서브루틴 반환) 명령

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 중첩 수가 16단을 초과했을 때 ON · 지정 대상 서브루틴이 존재하지 않을 때 ON · 서브루틴이 자신을 호출했을 때 ON · 실행 중인 서브루틴을 호출했을 때 ON · 지정 대상 서브루틴이 자체 태스크 내에서 정의되지 않았을 때 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

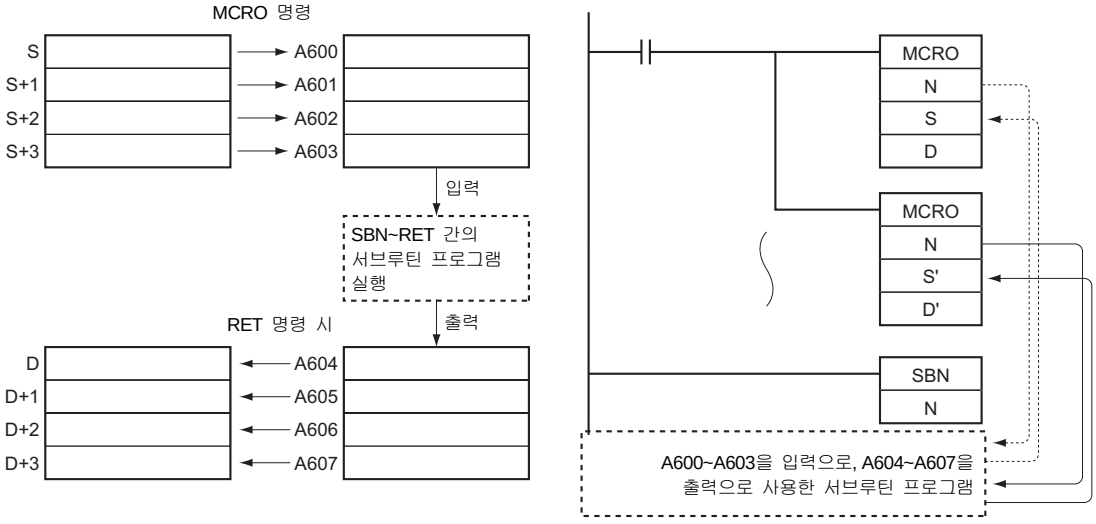
이름	주소	내용
MCRO 명령용 인수 영역	A600~A603	MCRO 명령 실행 시에 인수 데이터 하위 CH 번호에서 4CH의 인수 데이터가 여기에 복사됩니다. 복사된 인수 4CH는 서브루틴으로 전달됩니다.
MCRO 명령용 변수 영역	A604~A607	MCRO 명령의 서브루틴 실행 후 서브루틴에서 반환한 결과 4CH가 전달됩니다. 전달된 반환 결과는 MCRO 명령의 파라미터로 반환 데이터의 하위 CH 번호부터 4CH에 저장됩니다.

기능

N으로 지정한 번호의 서브루틴 영역(SBN 명령~RET 명령 사이의 영역)의 프로그램을 호출합니다.

SBS(서브루틴 호출) 명령과 달리, S로 지정한 인수 데이터와 D로 지정한 반환값 데이터를 사용하여 서브루틴 영역 프로그램과 데이터를 주고받을 수 있습니다. 그 결과, 하나의 서브루틴 영역 프로그램을 주소만 바꾼 여러 종류의 회로로 구분하여 사용할 수 있습니다.

S~S+3CH의 데이터를 A600~A603CH(MCRO 명령용 인수 영역)에 복사하여 지정된 번호의 서브루틴을 호출합니다. A600~A603CH의 데이터를 입력 데이터로 하고, A604~A607CH(MCRO 명령용 반환 영역)를 출력 데이터로 하는 서브루틴 프로그램을 실행한 후, A604~A607CH의 데이터를 D~D+3CH에 복사하여 MCRO 명령의 다음 명령으로 돌아갑니다.



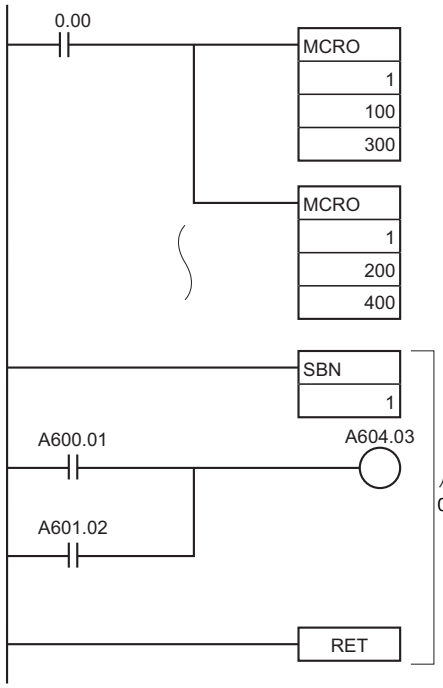
힌트

이 매크로 명령은 서브루틴 프로그램의 일종입니다. 서브루틴 번호로 지정한 서브루틴 프로그램을 MCRO(매크로) 명령이 실행될 때 지정하는 입출력 번호로 변경하여 실행합니다.
동일한 동작 패턴 및 구조이고 주소만 다른 복수의 회로가 있는 경우, 이 MCRO(매크로) 명령을 사용하면 해당 복수의 회로를 여러 개의 MCRO 명령과 하나의 회로로 묶을 수 있습니다.
단, MCRO(매크로) 명령으로 묶을 수 있는 회로는 완전히 같고 주소만 다른 회로이어야 합니다.

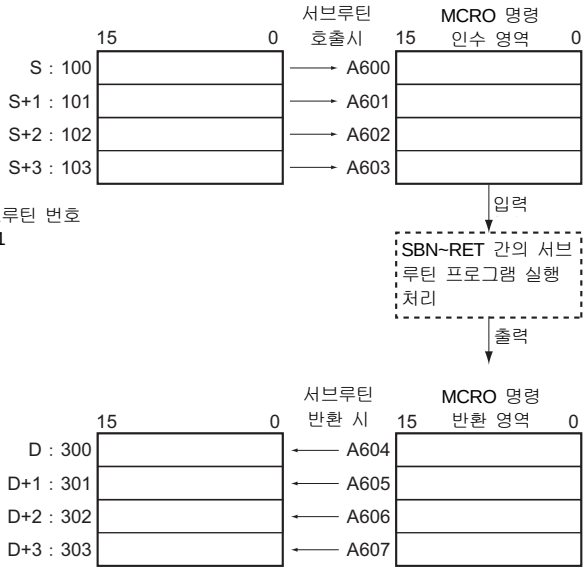
사용상의 주의 사항

- MCRO 명령을 통해 호출하는 서브루틴 프로그램에는 입력 주소에 인수 영역(A600~A603CH)의 4CH 데이터(채널, 비트 등)를 사용하고, 출력 주소에는 반환 영역(A604~A607CH)의 4CH 데이터(채널, 비트 등)를 사용해야 합니다. 이 4CH 이상의 데이터는 주고받을 수 없으므로 주의하십시오.
- MCRO 명령은 중첩이 가능합니다. 단, 인수 영역(A600~A603CH) 및 반환 영역(A604~A607CH)이 각각 하나밖에 없으므로 중첩할 때는 데이터 보관 처리 프로그램이 필요합니다.
- 인터럽트 태스크에서 MCRO 명령을 사용하는 경우 인터럽트 발생 시에 인수 영역(A600~A603CH) 및 반환 영역(A604~A607CH)을 덮어 쓸 가능성이 있습니다. 따라서 인터럽트 태스크에서 사용할 때는 데이터 보관 처리 프로그램이 필요합니다.

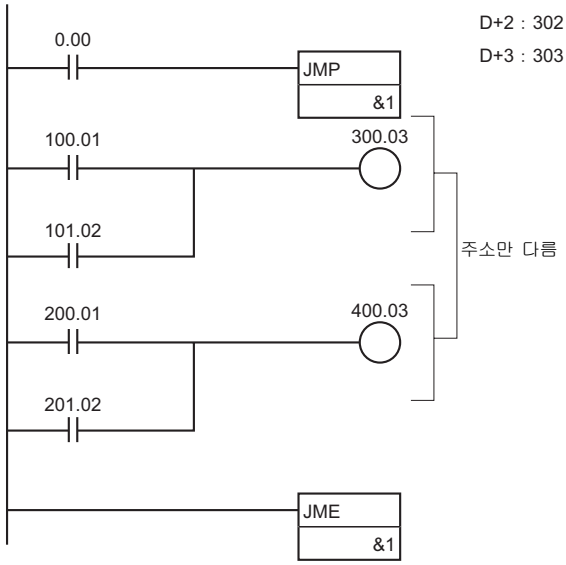
샘플 프로그램



0.00이 ON이면 서브루틴 번호 1의 프로그램에 ① 100~103CH의 데이터를 인수로 전달하여 서브루틴 프로그램을 실행한 후, 반환 데이터를 300~303CH에 저장합니다.
② 다음으로 200~203CH의 데이터를 인수로 전달하여 서브루틴 프로그램을 실행한 후, 반환 데이터를 400~403CH에 저장합니다.



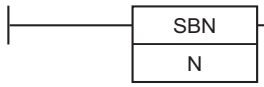
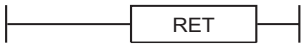
|| (동등)



마찬가지로 200~203CH→A600~A603, A604~A607→400~403CH로 전송

SBN/RET

명령 이름	니모닉	동작 옵션	펑션 번호	기능
서브루틴 진입	SBN	−	092	지정된 번호의 서브루틴의 선두를 나타냅니다.
서브루틴 반환	RET	−	093	서브루틴의 종료를 나타냅니다.

심볼	SBN	RET
		

특정 영역 내의 사용 여부

■ SBN

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	불가능	불가능	불가능	가능	불가능

■ RET

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	불가능	불가능	가능	가능	불가능

피연산자 설명

피연산자	내용	데이터 형식	크기
		SBN	
N	서브루틴 번호	−	1

■ SBN

N: 서브루틴 번호

10진수 0~1023*1

*1: CJ1M-CPU11/21의 경우 10진수 0~255

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
SBN	N	−	−	−	−	−	−	−	−	−	−	○	−	−	−	−	−	−	−

조합 사용 명령

SBS(서브루틴 호출) 명령 또는 MCRO(매크로) 명령

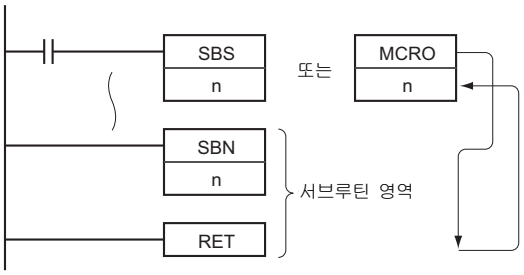
조건 플래그의 동작

■ SBN/ RET
없음

기능

■ SBN
서브루틴 영역의 선두를 나타냅니다.

최초의 SBN 명령 이후는 서브루틴 영역이 되며, 서브루틴 영역은 SBS 명령 또는 MCRO 명령에 의해서만 실행됩니다.

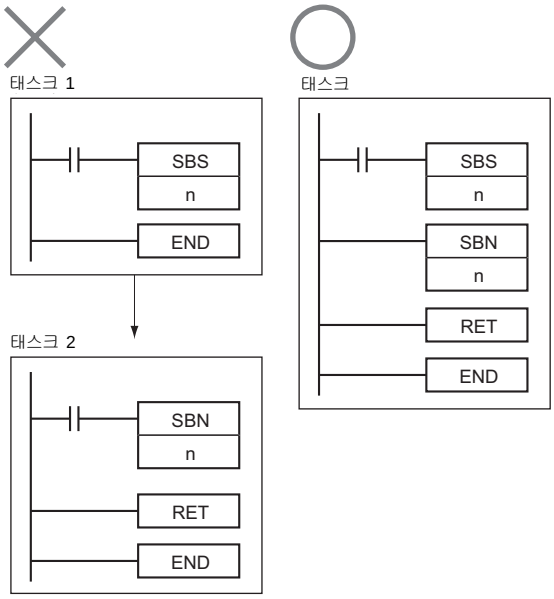


■ RET
서브루틴 영역의 실행을 종료하고 호출원인 SBS 명령 또는 MCRO 명령의 다음 명령으로 돌아갑니다.

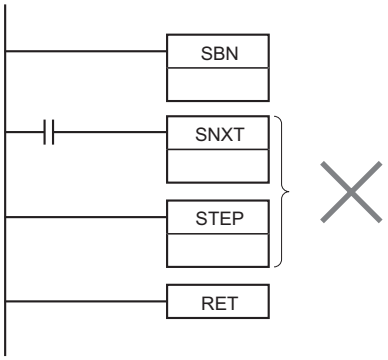
MCRO 명령에서 서브루틴이 호출된 경우는 A604~A607(MCRO 명령용 반환 영역)의 값을 D로 지정한 반환 데이터 하위 CH 번호 이후에 씁니다.

사용상의 주의 사항

· 서브루틴 영역(SBN~RET)은 동일한 번호의 SBS 명령 또는 MCRO 명령과 동일한 프로그램(태스크) 내에 배치하십시오. 태스크를 벗어나 서브루틴을 실행할 수는 없습니다.

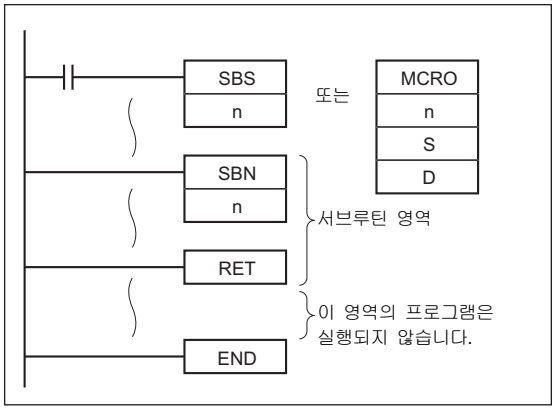


· 공정 진행 명령(STEP 명령, SNXT 명령)은 서브루틴 영역 내에서 사용할 수 없으므로 주의하십시오.



· 서브루틴 영역(SBN~RET)은 각 태스크에 할당되는 각 프로그램의 마지막(END 명령) 직전(복수의 서브루틴이 있는 경우는 통합하여), 일반 프로그램 뒤에 배치하십시오. 서브루틴 영역(SBN~RET) 이후에 일반 프로그램을 배치하면 해당하는 일반 프로그램은 실행되지 않고 무효가 됩니다.

프로그램(태스크)



주:

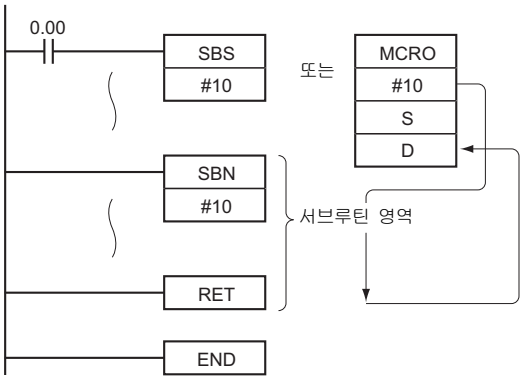
참고: 주변 툴을 사용한 N 데이터의 입력 방법은 다음과 같습니다.

· CX-Programmer의 경우: 0~1023으로 입력

· 프로그래밍 콘솔의 경우: 0000~1023으로 입력

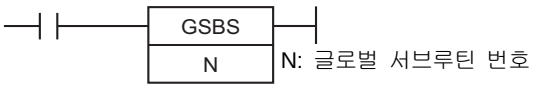
샘플 프로그램

0.00이 ON이면 서브루틴 번호 10의 서브루틴 영역(SBN~RET)의 프로그램을 실행한 후 SBS 또는 MCRO 명령의 다음 명령으로 돌아갑니다.



GSBS

명령 이름	니모닉	동작 옵션	평션 번호	기능
글로벌 서브루틴 호출	GSBS	@GSBS	750	지정된 번호의 글로벌 서브루틴을 호출하고 프로그램을 실행합니다.

심볼	GSBS
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	가능	가능	가능	가능	불가능

피연산자 설명

피연산자	내용	데이터 형식	크기
		SBN	
N	글로벌 서브루틴 번호	–	1

N: 글로벌 서브루틴 번호
10진수 0~1023*1

*1: CJ1M-CPU11/21의 경우 10진수 0~255

주: 글로벌 서브루틴 번호와 일반 서브루틴 번호(SBS 명령 또는 MCRO 명령의 피연산자)는 공통입니다. (동일한 번호를 사용하지 마십시오.)

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–

조합 사용 명령

GSBN(글로벌 서브루틴 진입) 명령과 GRET(글로벌 서브루틴 반환) 명령

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 일반 서브루틴과 합한 전체 중첩 수가 16단을 초과했을 때 ON · 지정 대상 글로벌 서브루틴이 존재하지 않을 때 ON · 글로벌 서브루틴에서 자신을 호출했을 때 ON · 실행 중인 글로벌 서브루틴을 호출했을 때 ON · 지정 대상 글로벌 서브루틴이 인터럽트 태스크 번호 0에 정의되어 있지 않을 때 ON · 그 이외일 때 OFF

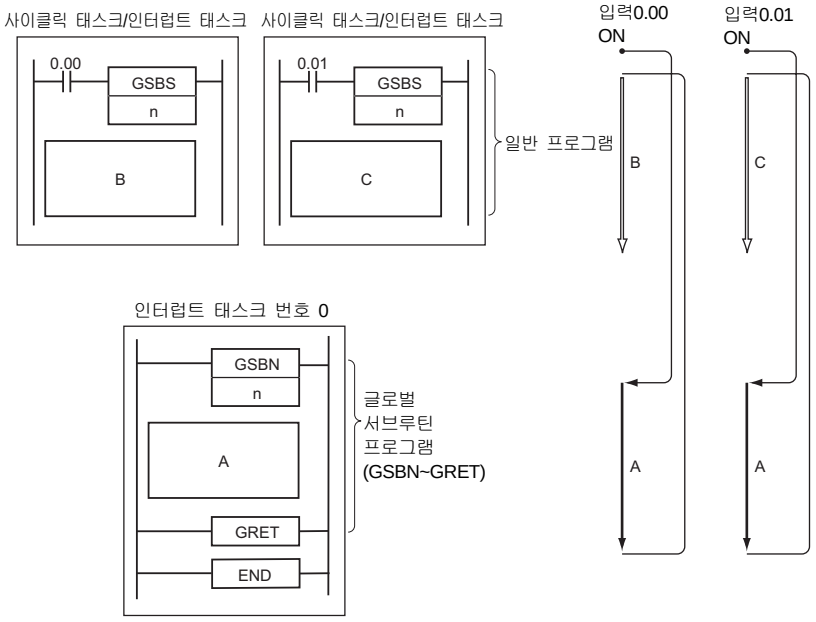
기능

N으로 지정한 번호의 글로벌 서브루틴 영역(GSBN 명령~GRET 명령 사이의 영역)의 프로그램을 호출합니다. 글로벌 서브루틴 영역을 실행 종료한 후 이 명령의 다음 명령으로 돌아갑니다.

동일한 글로벌 서브루틴 영역(GSBN 명령~GRET 명령 사이의 영역)을 여러 번 호출하는 것도 가능합니다.

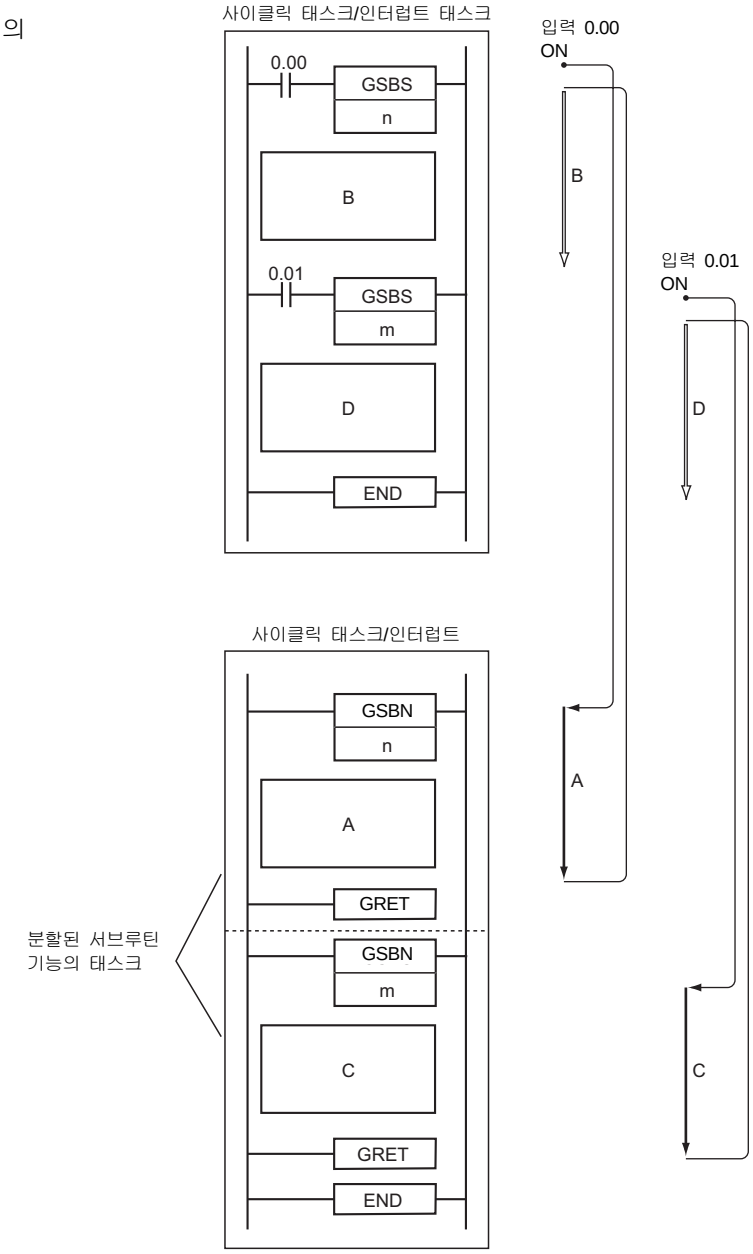
이 명령을 여러 개의 태스크로 기술하여 동일한 번호의 글로벌 서브루틴을 호출할 수 있습니다. 태스크 공통의 표준 회로를 글로벌 서브루틴으로 하면 프로그램을 모듈화할 수 있게 됩니다.

글로벌 서브루틴 영역(GSBN 명령~GRET 명령 사이의 영역)은 인터럽트 태스크 번호 0에만 정의(기술)할 수 있습니다. 그 밖의 태스크로 정의한 경우 GSBS 명령을 실행하려고 하면 에러가 발생하여 에러 플래그가 ON이 됩니다.
또한 이 GSBS 명령은 사이클릭 태스크 및 인터럽트 태스크(추가 태스크 포함) 중 어느 태스크에 기술해도 상관없습니다.



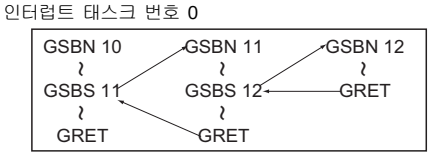
GSBS

인터럽트 태스크 번호 0 내에 복수의 글로벌 서브루틴 영역(GSBN 명령~GRET 명령)을 정의(기술)하는 것도 가능합니다.



서브루틴 영역(SBN 명령~RET 명령), 글로벌 서브루틴 영역(GSBN 명령~GRET 명령) 내에 SBS 명령 또는 GSBS 명령을 기술하여 서브루틴 및 글로벌 서브루틴을 중첩할 수 있습니다.

단, 중첩은 최대 16단까지 가능합니다.



사용상의 주의 사항

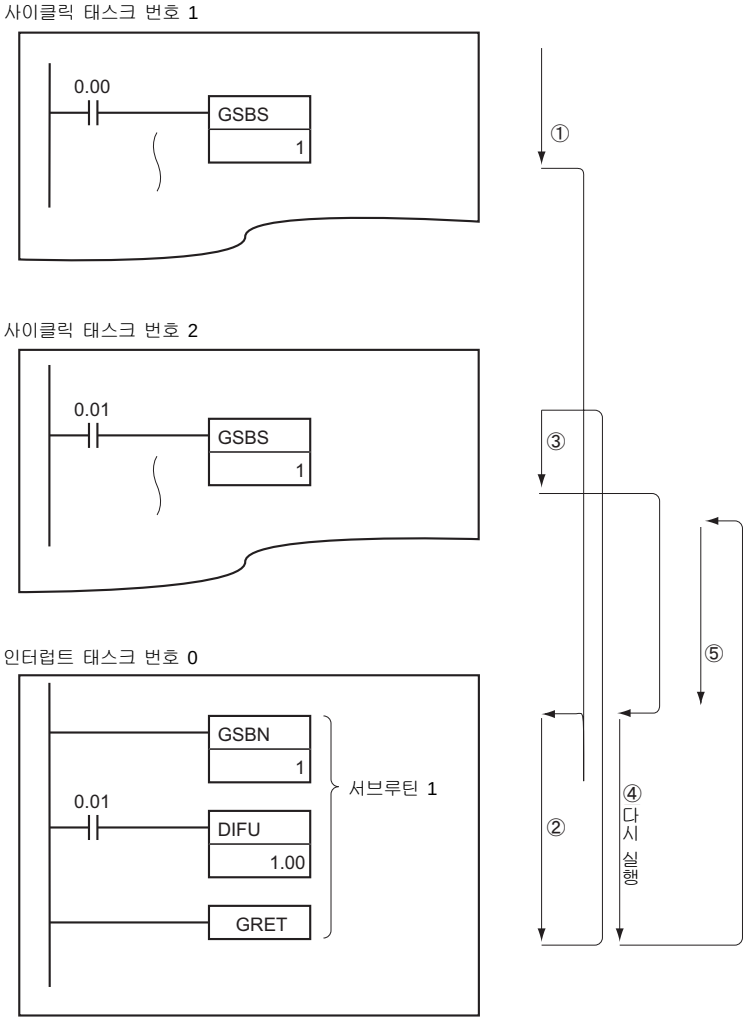
글로벌 서브루틴 영역(GSBN 명령~GRET 명령 사이의 영역)은 일반 서브루틴 호출(SBS) 명령으로 호출할 수 없습니다.

인터록 중에는 GSBS 명령이 실행되지 않으므로 글로벌 서브루틴 영역 내에서도 인터록되지 않습니다.

입력 미분형 명령(UP, DOWN, DIFU, DIFD, 동작 옵션 @, %가 있는 명령)을 서브루틴 내에서 사용하는 경우 다음과 같은 점에 주의해야 합니다.

· 동일한 사이클 내에서 동일한 글로벌 서브루틴을 여러 번 실행한 경우, 글로벌 서브루틴 내의 입력 미분형 명령(DIFU, DIFD, 동작 옵션 @, %가 있는 명령)의 동작이 일정치 않으므로 주의하십시오.

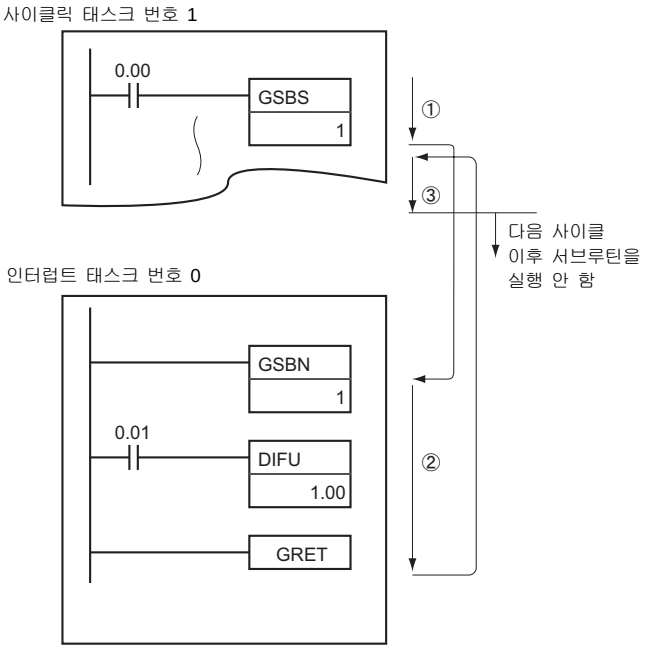
다음 예에서는 입력 릴레이 0.00이 ON되면 글로벌 서브루틴 1이 실행되고, 입력 릴레이 0.01이 OFF→ON이 된 경우에는 DIFU 명령에 의해 1.00이 ON됩니다. 그러나 동일한 사이클 내에서 입력 릴레이 0.01이 ON된 경우, 글로벌 서브루틴 1이 다시 실행되므로 이번에는 DIFU 명령이 0.01의 기동을 검출하지 못하여 1.00이 OFF가 됩니다.



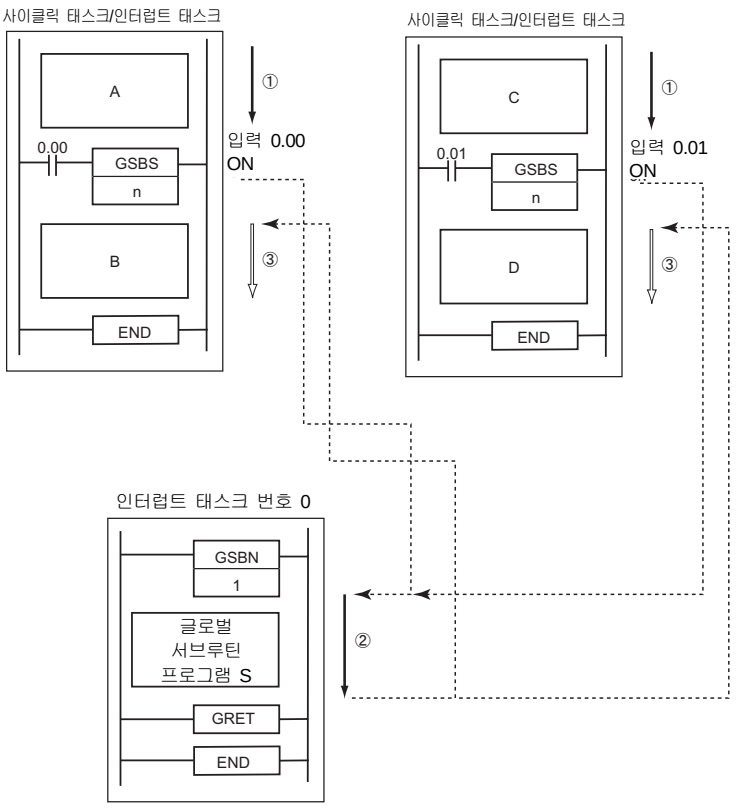
GSBS

· 이와 반대로 글로벌 서브루틴 내의 미분 명령(DIFU, DIFD 명령)이 실행되고 그 출력이 ON된 상태에서 같은 글로벌 서브루틴을 다음 번부터 호출하지 않은 경우, 미분 명령(DIFU, DIFD 명령)의 출력은 ON된 그대로 OFF되지 않으므로 주의하십시오.

다음 예에서는 0.00이 ON되면 글로벌 서브루틴 1이 실행되고, 0.01이 OFF→ON이 된 경우에는 DIFU 명령에 의해 1.00이 ON됩니다. 이후의 사이클에서 0.00이 OFF인 상태로 글로벌 서브루틴 1이 실행되지 않으면 DIFU 명령의 출력 1.00은 ON 그대로 유지됩니다.



샘플 프로그램



0.00이 ON이면 글로벌 서브루틴 번호 1의 글로벌 서브루틴 프로그램을 실행한 후, GSBS 명령의 다음 명령으로 돌아갑니다.

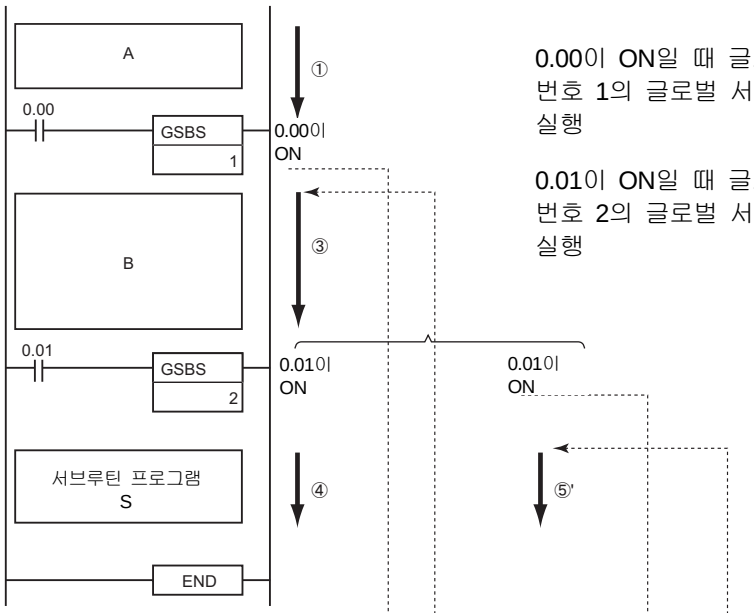
0.00	실행 순서
ON	A→S→B
OFF	A→B

0.01이 ON이면 글로벌 서브루틴 번호 1의 글로벌 서브루틴 프로그램을 실행한 후, GSBS 명령의 다음 명령으로 돌아갑니다.

0.01	실행 순서
ON	C→S→D
OFF	C→D

인터럽트 태스크 번호 0 내에 복수의 글로벌 서브루틴을 기술할 수 있습니다. 이 경우, 인터럽트 태스크 번호 0을 분할하여 서브루틴 기능의 태스크로 사용할 수 있습니다.

사이클릭 태스크/인터럽트 태스크

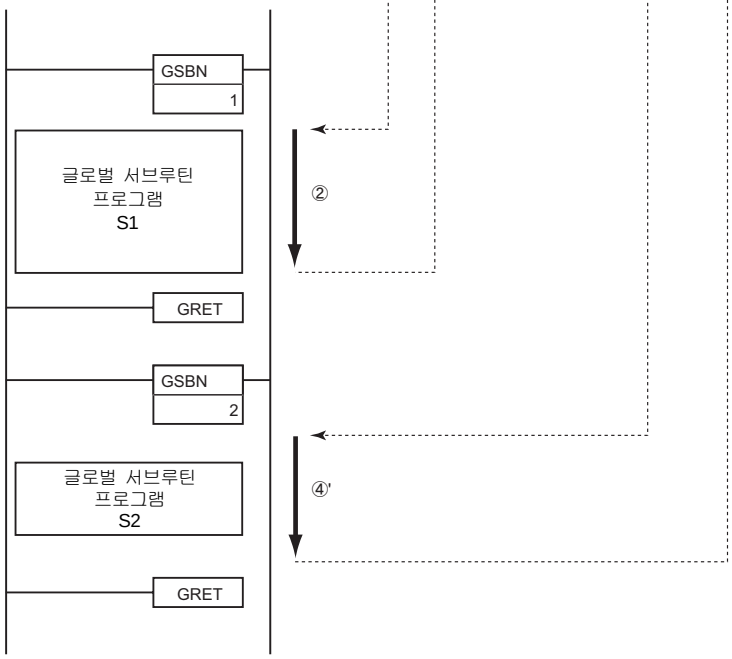


0.00이 ON일 때 글로벌 서브루틴 번호 1의 글로벌 서브루틴 프로그램 실행

0.01이 ON일 때 글로벌 서브루틴 번호 2의 글로벌 서브루틴 프로그램 실행

참고:
자체 태스크만 사용하는 서브루틴은 '서브루틴'을 사용하고, 다른 태스크와 공통으로 사용하는 서브루틴에는 '글로벌 서브루틴'을 사용하면 태스크의 영향을 고려한 디버그가 가능합니다.

인터럽트 태스크 번호 0



GSBN/GRET

명령 이름	니모닉	동작 옵션	평션 번호	기능
글로벌 서브루틴 진입	GSBN	−	751	지정된 번호의 글로벌 서브루틴의 선두를 나타냅니다.
글로벌 서브루틴 반환	GRET	−	752	글로벌 서브루틴의 종료를 나타냅니다.

심볼	GSBN	GRET
		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	불가능	불가능	불가능	가능	불가능

피연산자 설명

피연산자	내용	데이터 형식	크기
		GSBN	
N	글로벌 서브루틴 번호	−	1

N: 글로벌 서브루틴 번호
10진수 0~1023*1

*1: CJ1M-CPU11/21의 경우 10진수 0~255

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	−	−	−	−	−	−	−	−	−	−	○	−	−	−	−	−	−	−

조합 사용 명령

■ GSBN/GRET

· GSBS(글로벌 서브루틴 호출) 명령

조건 플래그의 동작

■ GSBN/GRET

없음

기능

■ GSBN

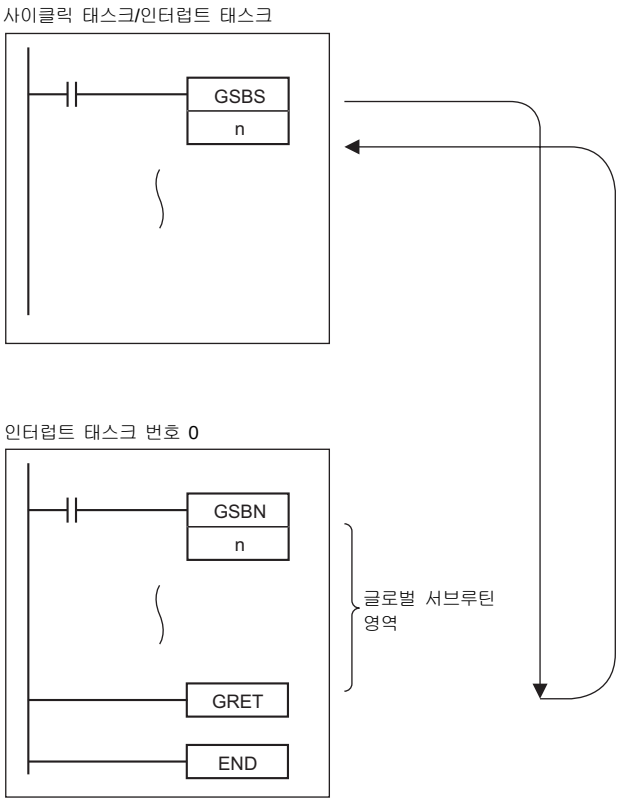
글로벌 서브루틴 영역의 선두를 나타냅니다.

최초의 GSBN 명령 이후는 글로벌 서브루틴 영역이 되며, 글로벌 서브루틴 영역은 GSBS 명령에 의해서만 실행됩니다.

■ GRET

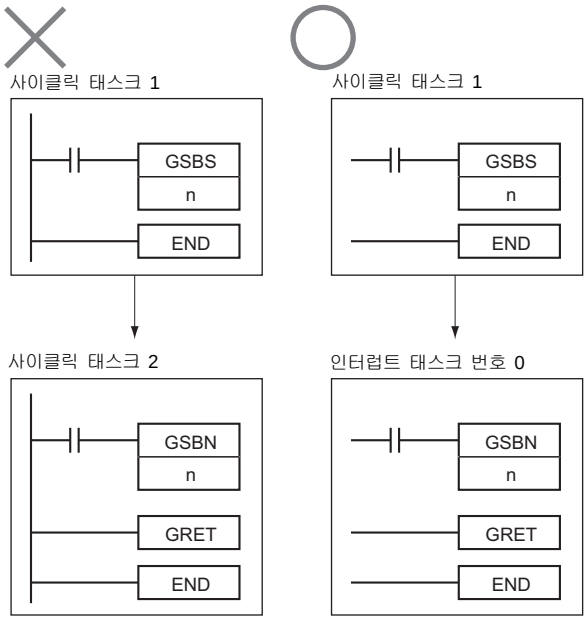
글로벌 서브루틴 영역의 종료를 나타냅니다.

글로벌 서브루틴 영역의 실행을 종료하고 호출원인 GSBS 명령의 다음 명령으로 돌아갑니다.



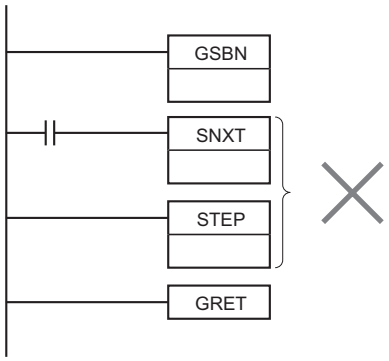
사용상의 주의 사항

· 글로벌 서브루틴 영역(GSBN~GRET)은 반드시 인터럽트 태스크 번호 0에 배치하십시오.



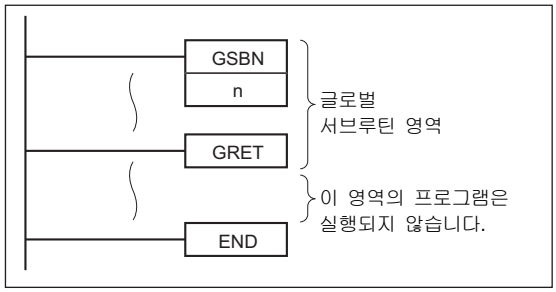
GSBN/GRET

· 공정 진행 명령(STEP 명령, SNXT 명령)은 글로벌 서브루틴 영역 내에서 사용할 수 없으므로 주의하십시오.



· 글로벌 서브루틴 영역(GSBN~GRET)은 인터럽트 태스크 번호 0에 할당되는 프로그램의 마지막(END 명령) 직전(복수의 글로벌 서브루틴이 있는 경우는 통합하여), 일반 프로그램 뒤에 배치하십시오. 글로벌 서브루틴 영역(GSBN~GRET) 이후에 일반 프로그램을 배치하면 해당하는 일반 프로그램은 실행되지 않고 무효가 됩니다.

프로그램(인터럽트 태스크 번호 0)



주:

참고: 주변 툴을 사용한 N 데이터의 입력 방법은 다음과 같습니다.

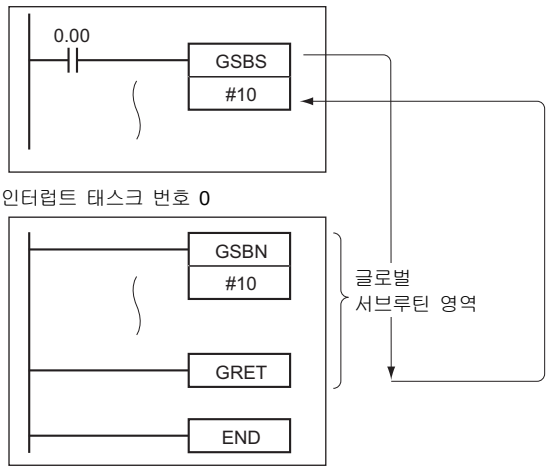
· CX-Programmer의 경우: 0~1023으로 입력

· 프로그래밍 콘솔의 경우: 0000~1023으로 입력

샘플 프로그램

0.00이 ON이면 서브루틴 번호 10의 서브루틴 영역(GSBN~GRET)의 프로그램을 실행한 후, GSBS 명령의 다음 명령으로 돌아갑니다.

·사이클릭 태스크/인터럽트 태스크



인터럽트 제어 명령

니모닉	명령 이름	펑션 번호	페이지
MSKS	인터럽트 마스크 세트	690	3-585
MSKR	인터럽트 마스크 읽기	692	3-591
CLI	인터럽트 해제	691	3-595
DI	인터럽트 태스크 실행 금지	693	3-599
EI	인터럽트 태스크 실행 금지 해제	694	3-601

인터럽트 제어 명령

CS/CJ 시리즈에는 다음과 같은 인터럽트 기능이 있습니다.

인터럽트 종류	인터럽트 조건	설정 방법
I/O 인터럽트	CPU 장치의 인터럽트 입력 유니트에서 인터럽트 입력 기동/종료 시	MSKS 명령으로 어느 인터럽트 입력 유니트의 어느 입력 번호를 인터럽트 유효화할지 지정합니다.
예약 인터럽트	정시(일정 시간 간격)	MSKS 명령으로 예약 인터럽트의 시간 간격을 지정하여 내부 타이머를 기동합니다.
전원 OFF 인터럽트	전원 OFF 확정 시(기본 전원 OFF 감지 시간+전원 OFF 감지 지연 시간 후)	PLC 시스템 설정의 '전원 OFF 인터럽트 태스크 유무 설정'에서 전원 OFF 인터럽트 태스크를 유효로 합니다. MSKS 명령은 사용하지 않습니다.
외부 인터럽트	CPU 장치의 고기능 I/O 유니트, CPU 장치의 CPU 고기능 유니트, INNER 보드(CS 시리즈만 해당)에서 인터럽트 요구 시	인터럽트 기능에 적합한 고기능 I/O 유니트, CPU 고기능 유니트, INNER 보드의 설정을 통해 인터럽트 조건과 인터럽트 태스크 번호를 지정합니다. MSKS 명령은 사용하지 않습니다.
입력 인터럽트	CPU 유니트 내장 인터럽트 입력의 기동/종료 시 (CJ1M-CPU2□만 해당)	MSKS 명령으로 어느 내장 입력을 인터럽트 유효로 할지 지정합니다.
고속 카운터 인터럽트	CPU 유니트 내장 고속 카운터 입력에 대해 목표값 일치 비교 또는 대역 비교 조건 성립 시 (CJ1M-CPU2□만 해당)	CTBL(비교 테이블 등록) 명령으로 목표값과 인터럽트 태스크 번호를 지정합니다. MSKS 명령은 사용하지 않습니다.

명령의 개요

■ 인터럽트 마스크 세트: MSKS 명령

I/O 인터럽트, 입력 인터럽트 및 예약 인터럽트의 인터럽트 요인은 초기 상태(운전 시작 시)에서 마스크(금지)가 실행됩니다. MSKS 명령을 실행하면 I/O 인터럽트 및 입력 인터럽트에 대한 인터럽트 입력의 수신 가능/금지의 제어 또는 예약 인터럽트에 대한 내부 타이머의 기동/정지가 가능해집니다.

■ 인터럽트 해제: CLI 명령

I/O 인터럽트 입력 및 입력 인터럽트 요인의 해제/유지 지정, 예약 인터럽트의 첫 번째 인터럽트 시작 시간 지정 또는 고속 카운터 인터럽트 요인의 해제/유지 지정(CJ1M CPU 유니트만 해당)을 실행합니다.

■ 인터럽트 마스크 읽기: MSKR 명령

MSKS 명령에 의해 지정된 인터럽트 제어 상태를 읽습니다.

■ 인터럽트 태스크 실행 금지: DI 명령

모든 인터럽트 태스크(전원 OFF 인터럽트 태스크 제외)의 실행을 일시적으로 금지합니다. 금지 중에 발생한 인터럽트 요인은 기억됩니다.

■ 인터럽트 태스크 실행 금지 해제: EI 명령

DI 명령으로 설정한 모든 인터럽트 태스크(전원 OFF 인터럽트 태스크 제외)의 실행 금지 상태를 해제합니다. 금지 중에 기억된 인터럽트 요인이 존재하는 경우 해당하는 인터럽트 태스크가 실행됩니다.

인터럽트 태스크의 사용상 주의 사항

■ 모든 인터럽트 공통

- 인터럽트 태스크 중에 고기능 I/O 유니트에 대해 IORF/FIORF(*1)/IORD/IOWR 명령을 실행하는 경우, 반드시 PLC 시스템 설정의 '고기능 I/O 주기적 리프레시 개별 금지 설정'으로 고기능 I/O 유니트에 대한 주기적 리프레시를 금지(1: 주기적 리프레시 안 함)로 설정하십시오.
  고기능 I/O 유니트와의 주기적 리프레시(0: 주기적 리프레시 수행)로 설정한 상태에서 인터럽트 태스크 중에 다음 중 하나를 실행하면 '다중 리프레시'가 되어 운전 계속 이상이 발생하고 인터럽트 태스크 이상 플래그(A402.13)가 ON이 됩니다.

  1) IORF/FIORF 명령을 통해 동일한 고기능 I/O 유니트에 대해 I/O 리프레시를 실행했을 때

  2) IORD/IOWR 명령을 통해 동일한 고기능 I/O 유니트의 메모리 영역에 대해 읽기 또는 쓰기를 실행했을 때

・C200H 고기능 I/O 유니트 또는 SYSBUS 원격 I/O 자국 장치를 연결한 경우 인터럽트 태스크를 10ms 이상 실행하지 마십시오. C200H 고기능 I/O 유니트 또는 SYSBUS 원격 I/O 자국 장치와의 I/O 리프레시 중에 인터럽트 태스크가 10ms 이상 실행되면 운전 계속 이상의 '인터럽트 태스크 이상 플래그'(A402.13)가 ON이 됩니다(CS 시리즈만 해당).

■ I/O 인터럽트

・I/O 인터럽트 태스크의 기동 요인은 CS/CJ 시리즈 기본 I/O 유니트(인터럽트 입력 유니트)와 C200H 기본 I/O 유니트(인터럽트 입력 유니트)의 인터럽트 입력입니다. INNER 보드, 고기능 I/O 유니트의 인터럽트 입력은 대상에서 제외됩니다.

・인터럽트 입력 유니트는 CPU 장치의 다음 장소에 장착하십시오. 그 이외의 위치에 장착한 경우 I/O 인터럽트 태스크를 기동할 수 없습니다.

  ・CJ2H CPU 유니트
    ・CJ2H-CPU6□-EIP : 슬롯 번호 0~3
    ・CJ2H-CPU6□ : 슬롯 번호 0~4
  ・CJ2M CPU 유니트 : 슬롯 번호 0~4
  ・CJ1H CPU 유니트 : 슬롯 번호 0~3
  ・CJ1M CPU 유니트 : 슬롯 번호 0~2

・인터럽트 유니트 번호는 왼쪽부터 장착 순서에 따라 연속 번호로 할당됩니다.

・CS1W-INT01과 C200HS-INT01은 혼재하여 사용할 수 없습니다.

・인터럽트 마스크를 해제하는 경우는 해당 시점까지 검출한 인터럽트 입력이 클리어됩니다.

・인터럽트 입력 요인은 대응하는 I/O 인터럽트 태스크의 실행 종료까지 기억되므로 실행 중에 다시 동일한 번호의 인터럽트 입력이 있어도 무시됩니다.

■ 예약 인터럽트

・예약 인터럽트 태스크의 실행 시간은 예약 인터럽트 설정 시간보다 짧게 설정하십시오.

・MSKS 명령의 예약 인터럽트 처리 지정은 어디까지나 예약 인터럽트의 간격을 설정하는 것이므로 첫 번째 인터럽트 시작 시간을 보장하는 것은 아닙니다. 첫 번째 인터럽트 시작 시간을 예약 인터럽트 시간과 통일시켜 예약 인터럽트 처리를 수행할 때는 MSKS 명령 앞에서 CLI 명령으로 '예약 인터럽트의 첫 번째 인터럽트 시작 시간 지정'을 설정하십시오. 단, CJ1M/CJ2M CPU 유니트는 리셋 지정을 하면 CLI 명령을 사용하지 않은 경우에도 첫 번째 인터럽트 시작 시간이 보장됩니다.

・예약 인터럽트의 단위 시간은 PLC 시스템 설정의 '예약 인터럽트 단위 시간 설정'에서 설정합니다.

・CJ2H CPU 유니트에서 유니트 간 동기 제어 기능을 사용할 때는 예약 인터럽트 태스크를 사용할 수 없습니다.

관련 특수 보조 릴레이

이름	주소	내용
인터럽트 태스크 최대값 처리 시간	A440	인터럽트 태스크 최대 처리 시간이 0.1ms 단위의 BIN 데이터로 저장됩니다. 이 값은 운전을 시작할 때 클리어됩니다.
최대 처리 시간 인터럽트 태스크 번호	A441	최대 처리 시간의 인터럽트 태스크 번호가 BIN 데이터로 저장됩니다. 태스크 0~255(16진수 0~FF)에 대응하여 #8000~80FF가 됩니다. 운전 시작 후 처음으로 인터럽트가 발생하면 A441CH의 비트 15가 1(ON)이 됩니다. 그 후 발생한 인터럽트 태스크에서 최대 처리 시간이 16진수 하위 2자리에 저장됩니다. 이 값은 운전을 시작할 때 클리어됩니다.
인터럽트 태스크 이상 플래그 다중 리프레시 이상 플래그	A402.13	다음과 같은 경우 (ON)이 됩니다. 1) C200H 고기능 I/O 유니트 또는 SYSBUS 원격 I/O 자국 장치와 I/O 리프레시 중에 인터럽트 태스크를 10ms 이상 실행했을 때(CS 시리즈만 해당) 2) PLC 시스템 설정의 '인터럽트 태스크 이상 검출 유무 설정'을 '검출'로 설정한 경우, 동일한 고기능 I/O 유니트에 대해 다음과 같은 상태가 되었을 때 ・인터럽트 태스크에서 실행한 IORF/FIORF/IORD/IOWR 명령과 사이클릭 태스크에서 실행한 IORF/FIORF/IORD/IOWR 명령이 경합했을 때 ・인터럽트 태스크에서 실행한 IORF/FIORF/IORD/IOWR 명령과 I/O 리프레시 (END 리프레시)가 경합했을 때 주: PLC 시스템 설정에서 '고기능 I/O 유니트와 주기적으로 갱신'으로 설정한 상태에서 인터럽트 태스크 중에 동일한 고기능 I/O 유니트에 대해 IORF/FIORF/IORD/IOWR 명령을 실행하면 다중 리프레시 상태가 되어 인터럽트 태스크 이상이 됩니다.
인터럽트 태스크 이상 요인 플래그	A426.15	위의 1), 2)에 대한 식별용 플래그
인터럽트 태스크 이상 발생 태스크 번호	A426.00~426.11	1)인 경우, 인터럽트 태스크 번호 2)인 경우, 다중 리프레시 대상의 고기능 I/O 유니트의 기기 번호 저장

관련 PLC 시스템 설정

■ 예약 인터럽트

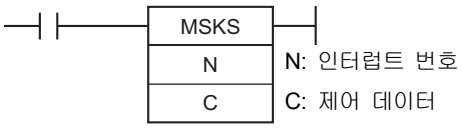
이름	내용	설정
예약 인터럽트 단위 시간 설정	MSKS(인터럽트 마스크 세트) 명령으로 지정하는 예약 인터럽트 설정 시간의 단위 시간을 설정합니다.	10ms(초기값) 1.0ms 0.1ms

■ 전원 OFF 인터럽트

이름	내용	설정
전원 OFF 인터럽트 태스크 유무 설정	전원 OFF 인터럽트 태스크를 '유효/사용'으로 설정하는 경우, 전원 OFF가 발생하면 전원 OFF 인터럽트 태스크가 기동합니다.	무효(초기값) 유효
전원 OFF 감지 지연 시간 설정	기본 전원 OFF 감지 시간(AC 전원: 10~25ms 불확정, DC 전원: 2~5ms 불확정)에 이 값을 더한 시간이 경과한 후, 전원 OFF로 판단합니다.	0~10ms (1ms 단위)

MSKS

명령 이름	니모닉	동작 옵션	평션 번호	기능
인터럽트 마스크 세트	MSKS	@MSKS	690	인터럽트 요인을 제어합니다.

심볼	MSKS
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	인터럽트 번호	–	1
C	제어 데이터	UINT	1

(1) I/O 인터럽트 및 입력 인터럽트의 인터럽트 입력을 허가/금지하는 경우

피연산자	내용	
N	인터럽트 입력 접점	인터럽트 번호
	인터럽트 입력 유니트(CS1W-INT01 또는 CJ1W-INT01)의 입력 접점(16점/1유니트)	0: 인터럽트 입력 유니트 번호 0 I/O 인터럽트 0~15(인터럽트 태스크 번호 100~115) 1: 인터럽트 입력 유니트 번호 1 I/O 인터럽트 16~31(인터럽트 태스크 번호 116~131)
	인터럽트 입력 유니트(C200HS-INT01)의 입력 접점 (8점/1유니트)	0: 인터럽트 입력 유니트 번호 0 I/O 인터럽트 0~7(인터럽트 태스크 번호 100~107) 1: 인터럽트 입력 유니트 번호 1 I/O 인터럽트 8~15(인터럽트 태스크 번호 108~115) 2: 인터럽트 입력 유니트 번호 2 I/O 인터럽트 16~23(인터럽트 태스크 번호 116~123) 3: 인터럽트 입력 유니트 번호 3 I/O 인터럽트 24~31(인터럽트 태스크 번호 124~131)
	CJ1M CPU 유니트(CJ1M-CPU2□)의 내장 입력 접점 (4점/1유니트)	6: 입력 인터럽트 0(인터럽트 태스크 번호 140) 7: 입력 인터럽트 1(인터럽트 태스크 번호 141) 8: 입력 인터럽트 2(인터럽트 태스크 번호 142) 9: 입력 인터럽트 3(인터럽트 태스크 번호 143)

MSKS

피연산자	내용	
C	**인터럽트 입력 접점**	**제어 데이터(인터럽트 마스크 데이터)**
	인터럽트 입력 유니트(CS1W-INT01 또는 CJ1W-INT01)의 입력 접점(16점/1유니트)	16진수 #0000~FFFF 0~15비트가 각 인터럽트 입력에 대응 각 비트의 의미 1: 수신 금지(마스크) 0: 수신 허가
	인터럽트 입력 유니트(C200HS-INT01)의 입력 접점(8점/1유니트)	16진수 #0000~00FF 하위 0~7비트가 각 인터럽트 입력에 대응 각 비트의 의미 1: 수신 금지(마스크) 0: 수신 허가
	CJ1M CPU 유니트(CJ1M-CPU2□)의 내장 입력 접점(4점/1유니트)	10진수 &0(16진수 #0000): 인터럽트 허가(직접 모드) 10진수 &1(16진수 #0001): 인터럽트 금지 10진수 &2(16진수 #0002): 인터럽트 허가(감산 및 카운터 모드) 10진수 &3(16진수 #0003): 인터럽트 허가(가산 및 카운터 모드)

(2) I/O 인터럽트 및 입력 인터럽트의 인터럽트 입력의 에지 검출 방향을 지정하는 경우 (인터럽트 입력 유니트(C200HS-INT01) 사용 시 제외)

데이터	내용	
N	**인터럽트 입력 접점**	**인터럽트 번호**
	인터럽트 입력 유니트(CS1W-INT01 또는 CJ1W-INT01)의 입력 접점(16점/1유니트)	2: 인터럽트 입력 유니트 번호 0 I/O 인터럽트 0~15(인터럽트 태스크 번호 100~115) 3: 인터럽트 입력 유니트 번호 1 I/O 인터럽트 16~31(인터럽트 태스크 번호 116~131)
	CJ1M CPU 유니트(CJ1M-CPU2□)의 내장 입력 접점(4점/1유니트)	10: 입력 인터럽트 0(인터럽트 태스크 번호 140) 11: 입력 인터럽트 1(인터럽트 태스크 번호 141) 12: 입력 인터럽트 2(인터럽트 태스크 번호 142) 13: 입력 인터럽트 3(인터럽트 태스크 번호 143)
C	**인터럽트 입력 접점**	**제어 데이터(인터럽트 입력 에지 검출 방향 지정)**
	인터럽트 입력 유니트(CS1W-INT01 또는 CJ1W-INT01)의 입력 접점(16점/1유니트)	16진수 #0000~FFFF 0~15비트가 각 인터럽트 입력에 대응 각 비트의 의미 0: 기동 검출 1: 종료 검출
	CJ1M CPU 유니트(CJ1M-CPU2□)의 내장 입력 접점(4점/1유니트)	10진수 &0(16진수 #0000): 기동 검출 10진수 &1(16진수 #0001): 종료 검출

주: 에지 검출 지정을 변경하는 경우는 해당 시점까지 검출한 인터럽트 입력이 클리어됩니다.

(3) 예약 인터럽트의 내부 타이머를 기동/정지하는 경우

피연산자	내용	
N	인터럽트 번호	
	4: 예약 인터럽트 0(인터럽트 태스크 번호 2) 5: 예약 인터럽트 1(인터럽트 태스크 번호 3) 주: CJ1M-CPU11/21에서는 예약 인터럽트 0만 사용 가능 주: CJ2H CPU 유니트에서 유니트 간 동기 제어 기능을 사용할 때는 예약 인터럽트(0, 1)는 사용 불가능	
C	예약 인터럽트 단위 시간(PLC 시스템 설정)	제어 데이터(예약 인터럽트 설정 시간)
	전체 설정 공통	10진수 &0(16진수 #0000): 인터럽트 금지(내부 타이머 정지)
	10ms	10진수 &1~9999(16진수 #0001~270F): 인터럽트 허가(인터럽트 간격 10~99990ms에서 내부 타이머 기동)
	1ms	10진수 &1~9999(16진수 #0001~270F): 인터럽트 허가(인터럽트 간격 1~9999ms에서 내부 타이머 기동)
	0.1ms	CJ1M CPU 유니트의 경우 10진수 &5~9999(16진수 #0005~270F): 인터럽트 허가(인터럽트 간격 시간 0.5~999.9ms에서 내부 타이머 기동) 주: 10진수 &1~4(16진수 #0001~0004)는 사용 불가능. 명령 에러가 됩니다.
		CJ1-H-R CPU 유니트, CJ2H CPU 유니트의 경우 10진수 &2~9999(16진수 #0002~270F): 인터럽트 허가(인터럽트 간격 시간 0.2~999.9ms에서 내부 타이머 기동) 주: 10진수 &1(16진수 #0001)은 사용 불가능. 명령 에러가 됩니다. CJ2M CPU 유니트의 경우 10진수 &4~9999(16진수 #0004~270F): 인터럽트 허가(인터럽트 간격 시간 0.4~999.9ms에서 내부 타이머 기동) 주: 10진수 &1~3(16진수 #0001~0003)은 사용 불가능. 명령 에러가 됩니다.

주: CJ2H CPU 유니트(유니트 Ver.1.1 이상)에서 인터럽트 고속 기동 기능을 사용하면 예약 인터럽트 0(인터럽트 태스크 번호 2)에 대해서만 10진수 &1(16진수 #0001)의 설정 즉, 0.1ms의 인터럽트 간격 시간을 설정할 수 있습니다.

(4) 리셋 지정에서 예약 인터럽트의 내부 타이머를 기동하는 경우
(CJ1M CPU 유니트, CJ2M CPU 유니트의 경우만 해당)

피연산자	내용	
N	인터럽트 번호	
	14: 예약 인터럽트 0(인터럽트 태스크 번호 2) 15: 예약 인터럽트 1(인터럽트 태스크 번호 3) 주: CJ1M-CPU11/21에서는 예약 인터럽트 0만 사용 가능	
C	예약 인터럽트 단위 시간(PLC 시스템 설정)	제어 데이터(예약 인터럽트 설정 시간)
	전체 설정 공통	10진수 &0(16진수 #0000): 인터럽트 금지(내부 타이머 정지)
	10ms	10진수 &1~9999(16진수 #0001~270F): 인터럽트 허가 (내부 타이머 값을 리셋한 후 인터럽트 간격 10~99990ms에서 내부 타이머 기동)
	1ms	10진수 &1~9999(16진수 #0001~270F): 인터럽트 허가 (내부 타이머 값을 리셋한 후 인터럽트 간격 1~9999ms에서 내부 타이머 기동)
	0.1ms	CJ1M CPU 유니트의 경우 10진수 &5~9999(16진수 #0005~270F): 인터럽트 허가 (내부 타이머 값을 리셋한 후 인터럽트 간격 시간 0.5~999.9ms에서 내부 타이머 기동) 주: 10진수 &1~4(16진수 #0001~0004)는 사용 불가능. 명령 에러가 됩니다.
		CJ2M CPU 유니트의 경우 10진수 &4~9999(16진수 #0004~270F): 인터럽트 허가(인터럽트 간격 시간 0.4~999.9ms에서 내부 타이머 기동) 주: 10진수 &1~3(16진수 #0001~0003)은 사용 불가능. 명령 에러가 됩니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
C	○	○	○	○	○	○	○	○	○	○		○		○				

MSKS

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· N의 데이터가 0~5(CJ1M CPU 유니트의 경우 0~15, CJ2M CPU 유니트의 경우 0~15, 100~107, 110~117)의 범위에 없을 때 ON · I/O 인터럽트 지정: C200HS-INT01 유니트의 입력 인터럽트 사용 시, C의 내용이 16진수 #0000~00FF의 범위에 없을 때 ON · I/O 인터럽트 지정: CJ1M/CJ2M CPU 유니트 내장 입력 인터럽트 사용 시, C의 내용이 0~3의 범위에 없을 때 ON · 예약 인터럽트 지정: 10ms/1ms 단위에서 C의 내용이 10진수 &0~9999(16진수 #0000~270F)의 범위에 없을 때 ON · 예약 인터럽트 지정: CJ1M CPU 유니트에서 0.1ms 단위인 경우, 10진수 &0, &5~9999(16진수 #0000, #0005~270F)의 범위에 없을 때 ON · 예약 인터럽트 지정: CJ1-H-R/CJ2H/CJ2M CPU 유니트에서 0.1ms 단위인 경우, 10진수 &0, &2~9999(16진수 #0000, #0002~270F)의 범위에 없을 때 ON(주) · CJ2H CPU 유니트에서 유니트 간 동기 제어 기능 사용 시, N의 데이터가 4, 5(예약 인터럽트 지정)일 때 ON
= 플래그	P_EQ	OFF
부정 플래그	P_N	OFF

주: CJ2H CPU 유니트(유니트 Ver.1.1 이상)에서 인터럽트 고속 기동 기능을 사용하면 예약 인터럽트 0(N = 4)에 대해서만 10진수 &1(16진수 #0001)을 설정해도 명령 에러가 되지 않고 에러 플래그는 OFF가 됩니다.

관련 PLC 시스템 설정

이름	내용	설정
예약 인터럽트 단위 시간 설정	MSKS(인터럽트 마스크 세트) 명령으로 지정하는 예약 인터럽트 설정 시간의 단위 시간을 설정합니다.	10ms(초기값) 1.0ms 0.1ms

관련 특수 보조 릴레이

이름	주소	내용
다중 리프레시 이상 플래그	A402.13	다음과 같은 경우 1(ON)이 됩니다. 1) PLC 시스템 설정의 '인터럽트 태스크 이상 검출 유무 설정'을 '검출'로 설정한 경우, 동일한 고기능 I/O 유니트에 대해 다음과 같은 상태가 되었을 때 ・ 인터럽트 태스크에서 실행한 IORF/FIORF/IORD/IOWR 명령과 사이클릭 태스크에서 실행한 IORF/FIORF/IORD/IOWR 명령이 경합했을 때 ・ 인터럽트 태스크에서 실행한 IORF/FIORF/IORD/IOWR 명령과 I/O 리프레시(END 리프레시)가 경합했을 때 2) C200H 고기능 I/O 유니트 또는 SYSBUS 원격 I/O 자국 장치와 I/O 리프레시 중에 인터럽트 태스크를 10ms 이상 실행했을 때(CS 시리즈만 해당)
다중 리프레시 이상 요인 플래그	A426.15	A402.13(다중 리프레시 이상 플래그)이 ON일 때의 이상 요인을 저장합니다. 1: 다중 리프레시 0: C200H 고기능 I/O 유니트 또는 SYSBUS 원격 I/O 자국 장치와 I/O 리프레시 중에 인터럽트 태스크를 10ms 이상 실행(CS 시리즈만 해당)
다중 리프레시 이상 발생 기기 번호	A426.00 ~A426.11	A402.13(다중 리프레시 이상 플래그)이 ON일 때 A426.15(다중 리프레시 이상 요인 플래그)의 상태에 따라 다음 정보가 BIN 12비트에 저장됩니다. ・ A426.15가 1(ON)인 경우, 다중 리프레시된 고기능 I/O 유니트의 기기 번호 ・ A426.15가 0(OFF)인 경우, 10ms 이상 실행된 인터럽트 태스크의 태스크 번호(CS 시리즈만 해당)

기능

N의 값을 통해 어느 인터럽트에 대해 어떤 설정을 수행할지 지정합니다.

(1) I/O 인터럽트 및 입력 인터럽트의 인터럽트 입력 허가/금지(N=0~3, 6~9, 100~107)

- N으로 지정한 인터럽트 입력 접점에 대해 C의 비트로 인터럽트 수신 허가/금지를 설정합니다. 이를 통해 각 I/O 인터럽트 태스크 및 입력 인터럽트 태스크의 실행 여부를 제어할 수 있습니다.
- 인터럽트 입력 허가를 지정하면 해당 시점까지 검출한 인터럽트 요인이 클리어됩니다.

(2) I/O 인터럽트 및 입력 인터럽트의 에지 검출 방향 지정(N=2, 3, 10~13, 110~117)

- N으로 지정한 인터럽트 입력 접점에 대해 C의 비트로 에지 검출 방향(입력 인터럽트의 신호를 기동/종료 중 어디에서 검출할지)을 지정합니다.
- 에지 검출 지정은 인터럽트 입력의 허가/금지 처리와 조합하여 사용합니다. 또한 기동/종료를 지정하는 MSKS 명령을 실행하지 않으면 초기 설정으로 기동을 검출하는 동작을 합니다.
- 에지 검출 지정을 변경하면 해당 시점까지 검출한 인터럽트 요인이 클리어됩니다.

(3) 예약 인터럽트의 내부 타이머 기동/정지(N=4, 5)

- N으로 지정한 예약 인터럽트의 인터럽트 간격을 C로 지정한 예약 인터럽트 설정 시간으로 설정하는 동시에 내부 타이머를 기동합니다. 또한 내부 타이머를 정지할 수도 있습니다. 이를 통해 예약 인터럽트 태스크의 실행 여부를 제어할 수 있습니다.
- 이 명령으로 내부 타이머를 재기동하면 이 명령 실행부터 첫 번째 예약 인터럽트 태스크가 기동될 때까지의 인터럽트 시작 시간이 일정치 않게 됩니다. 이는 이전 기동 시 사용한 내부 타이머의 현재값을 계속 사용하기 때문입니다.
  따라서 인터럽트 시작 시간을 지정하려는 경우에는 이 명령과 함께 CLI 명령을 사용하십시오.

(4) 예약 인터럽트의 내부 타이머 기동(리셋 지정) (N=14, 15)

- N으로 지정한 예약 인터럽트의 인터럽트 간격을 C로 지정한 시간으로 설정하는 동시에 내부 타이머의 현재값을 리셋하고 기동합니다. 이렇게 하면 (3)과 달리 이 명령 실행 후의 인터럽트 시작 시간이 보장됩니다(CJ1M CPU 유니트, CJ2M CPU 유니트만 해당).

주 1: CJ1M-CPU11/21에서는 예약 인터럽트 태스크를 1개(예약 인터럽트 0(인터럽트 태스크 번호 2))만 사용할 수 있습니다.

주 2: 예약 인터럽트 설정 시간으로 지정하는 숫자의 단위는 PLC 시스템 설정의 '예약 인터럽트 단위 시간 설정'을 통해 설정합니다(초기값: 10ms).

힌트

- 인터럽트 태스크의 최대 처리 시간은 특수 보조 릴레이 A440CH(최대 처리 시간의 인터럽트 태스크 실행 시)에 저장됩니다. 또한 동시에 해당 인터럽트 태스크의 번호가 특수 보조 릴레이 A441CH의 하위 바이트에 저장됩니다.

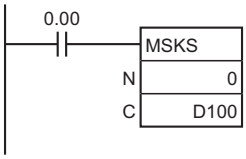
사용상의 주의 사항

- 예약 인터럽트 태스크의 실행 시간은 예약 인터럽트의 시간 간격보다 짧게 설정하십시오.
- MSKS 명령의 예약 인터럽트 설정 지정은 어디까지나 예약 인터럽트의 간격을 설정하는 것이므로 MSKS 명령 실행 후 첫 번째 예약 인터럽트까지의 시간을 설정하는 것은 아닙니다.
  MSKS 명령 실행 시의 내부 타이머는 이전 정지 시의 값부터 계속 시간을 계산하므로 첫 번째 예약 인터럽트까지의 시간은 일정치 않습니다. 첫 번째 예약 인터럽트까지의 시간을 규정하여 예약 인터럽트 처리를 수행할 때는 MSKS 명령 앞에서 CLI 명령으로 '예약 인터럽트의 첫 번째 인터럽트 시작 시간 지정'을 설정하십시오. 단, CJ1M CPU 유니트 또는 CJ2M CPU 유니트는 리셋 지정을 하면 CLI 명령을 사용하지 않은 경우에도 첫 번째 인터럽트까지의 시간이 예약 인터럽트의 시간 간격과 같게 설정됩니다.
- CJ2H CPU 유니트에서 유니트 간 동기 제어 기능을 사용할 때는 예약 인터럽트 태스크를 사용할 수 없습니다. MSKS 명령의 인터럽트 번호에 N=4~5를 지정하면 명령 에러가 됩니다.

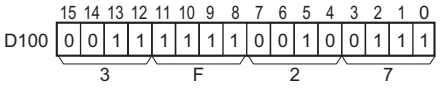
MSKS

샘플 프로그램

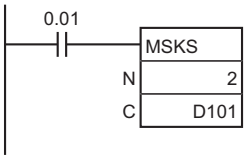
■ I/O 인터럽트 지정 시(CS1W-INT01 또는 CJ1W-INT01 사용 시)



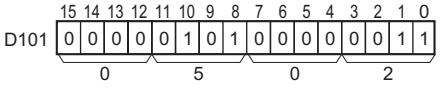
0.00이 ON이면 인터럽트 입력 유니트 번호 0의 인터럽트 수신 허가 및 금지를 설정합니다.



0: 허가/1: 금지

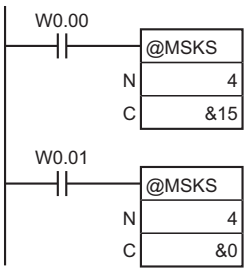


0.01이 ON이면 인터럽트 입력 유니트 번호 0의 입력 기동/종료 검출을 설정합니다.

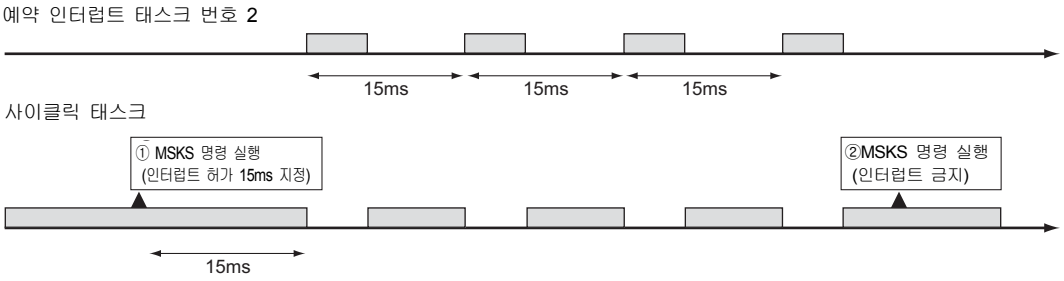


0: 기동 검출/1: 종료 검출

■ 예약 인터럽트 지정 시

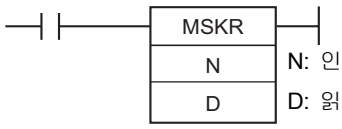


① W0.00이 OFF→ON이면 예약 인터럽트 0의 인터럽트 간격 시간을 15ms로 설정하여 내부 타이머를 기동합니다(단위 설정 시간: 1ms일 때).
② W0.01이 OFF→ON이면 예약 인터럽트 0의 내부 타이머를 정지하여 예약 인터럽트 발생을 중지합니다.



MSKR

명령 이름	니모닉	동작 옵션	펑션 번호	기능
인터럽트 마스크 읽기	MSKR	@MSKR	692	MSKS 명령에 의해 지정된 인터럽트 제어 상태를 읽습니다.

심볼	MSKR
	

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	인터럽트 번호	–	1
D	읽기 결과 출력 CH 번호	UINT	1

(1) 현재 설정된 I/O 인터럽트 및 입력 인터럽트의 인터럽트 마스크 상태를 읽는 경우

피연산자	내용	
N	인터럽트 입력 접점	인터럽트 번호
	인터럽트 입력 유니트(CS1W-INT01 또는 CJ1W-INT01)의 입력 접점(16점/1유니트)	0: 인터럽트 입력 유니트 번호 0 I/O 인터럽트 0~15(인터럽트 태스크 번호 100~115) 1: 인터럽트 입력 유니트 번호 1 I/O 인터럽트 16~31(인터럽트 태스크 번호 116~131)
	인터럽트 입력 유니트(C200HS-INT01)의 입력 접점(8점/1유니트)	0: 인터럽트 입력 유니트 번호 0 I/O 인터럽트 0~7(인터럽트 태스크 번호 100~107) 1: 인터럽트 입력 유니트 번호 1 I/O 인터럽트 8~15(인터럽트 태스크 번호 108~115) 2: 인터럽트 입력 유니트 번호 2 I/O 인터럽트 16~23(인터럽트 태스크 번호 116~123) 3: 인터럽트 입력 유니트 번호 3 I/O 인터럽트 24~31(인터럽트 태스크 번호 124~131)
	CJ1M CPU 유니트(CJ1M-CPU2□)의 내장 입력 접점(4점/1유니트)	6: 입력 인터럽트 0(인터럽트 태스크 번호 140) 7: 입력 인터럽트 1(인터럽트 태스크 번호 141) 8: 입력 인터럽트 2(인터럽트 태스크 번호 142) 9: 입력 인터럽트 3(인터럽트 태스크 번호 143)

MSKR

피연산자	내용	
D	**인터럽트 입력 접점**	**읽기 상태(인터럽트 마스크 상태)**
	인터럽트 입력 유니트(CS1W-INT01 또는 CJ1W-INT01)의 입력 접점(16점/1유니트)	16진수 #0000~FFFF 0~15비트가 각 인터럽트 입력에 대응 각 비트의 의미 1: 수신 금지(마스크) 0: 수신 허가
	인터럽트 입력 유니트(C200HS-INT01)의 입력 접점(8점/1유니트)	16진수 #0000~00FF 하위 0~7비트가 각 인터럽트 입력에 대응 각 비트의 의미 1: 수신 금지(마스크) 0: 수신 허가
	CJ1M CPU 유니트(CJ1M-CPU2□)의 내장 입력 접점(4점/1유니트)	10진수 &0(16진수 #0000): 인터럽트 허가(직접 모드) 10진수 &1(16진수 #0001): 인터럽트 금지 10진수 &2(16진수 #0002): 인터럽트 허가 (감산 및 카운터 모드) 10진수 &3(16진수 #0003): 인터럽트 허가 (가산 및 카운터 모드)

(2) 현재 설정된 I/O 인터럽트 및 입력 인터럽트의 에지 검출 방향 지정 상태를 읽는 경우
(인터럽트 입력 유니트(C200HS-INT01) 사용 시 제외)

피연산자	내용	
N	**인터럽트 입력 접점**	**인터럽트 번호**
	인터럽트 입력 유니트(CS1W-INT01 또는 CJ1W-INT01)의 입력 접점(16점/1유니트)	2: 인터럽트 입력 유니트 번호 0 I/O 인터럽트 0~15(인터럽트 태스크 번호 100~115) 3: 인터럽트 입력 유니트 번호 1 I/O 인터럽트 16~31(인터럽트 태스크 번호 116~131)
	CJ1M CPU 유니트(CJ1M-CPU2□)의 내장 입력 접점(4점/1유니트)	10: 입력 인터럽트 0(인터럽트 태스크 번호 140) 11: 입력 인터럽트 1(인터럽트 태스크 번호 141) 12: 입력 인터럽트 2(인터럽트 태스크 번호 142) 13: 입력 인터럽트 3(인터럽트 태스크 번호 143)
D	**인터럽트 입력 접점**	**읽기 결과(인터럽트 입력 에지 검출 방향 지정)**
	인터럽트 입력 유니트(CS1W-INT01 또는 CJ1W-INT01)의 입력 접점(16점/1유니트)	16진수 #0000~FFFF 0~15비트가 각 인터럽트 입력에 대응 각 비트의 의미 0: 기동 검출 1: 종료 검출
	CJ1M CPU 유니트(CJ1M-CPU2□)의 내장 입력 접점(4점/1유니트)	10진수 &0(16진수 #0000): 기동 검출 10진수 &1(16진수 #0001): 종료 검출

(3) 예약 인터럽트의 내부 타이머 설정값을 읽는 경우

피연산자	내용	
N	**인터럽트 번호**	
	4: 예약 인터럽트 0(인터럽트 태스크 번호 2) 5: 예약 인터럽트 1(인터럽트 태스크 번호 3) 주: CJ1M-CPU11/21에서는 예약 인터럽트 0만 사용 가능	
D	**예약 인터럽트 단위 시간(PLC 시스템 설정)**	**읽기 결과(예약 인터럽트 설정 시간)**
	전체 설정 공통	10진수 &0(16진수 #0000): 인터럽트 금지(내부 타이머 정지 중)
	10ms	10진수 &1~9999(16진수 #0001~270F): 인터럽트 허가(인터럽트 간격 10~99990ms에서 내부 타이머 기동 중)
	1ms	10진수 &1~9999(16진수 #0001~270F): 인터럽트 허가(인터럽트 간격 1~9999ms에서 내부 타이머 기동 중)
	0.1ms	10진수 &1~9999(16진수 #0001~270F): 인터럽트 허가(인터럽트 간격 0.1~999.9ms에서 내부 타이머 기동 중)

(4) 예약 인터럽트의 내부 타이머 현재값을 읽는 경우(CJ1M CPU 유니트, CJ2M CPU 유니트만 해당)

피연산자	내용	
N	인터럽트 번호	
	14: 예약 인터럽트 0(인터럽트 태스크 번호 2) 15: 예약 인터럽트 1(인터럽트 태스크 번호 3) 주: CJ1M-CPU11/21에서는 예약 인터럽트 0만 사용 가능	
D	예약 인터럽트 단위 시간(PLC 시스템 설정)	읽기 결과(내부 타이머 현재값)
	10ms	10진수 &0~9999(16진수 #0000~270F): 내부 타이머 현재값(0~99990ms)
	1ms	10진수 &0~9999(16진수 #0000~270F): 내부 타이머 현재값(0~9999ms)
	0.1ms	10진수 &0~9999(16진수 #0000~270F): 내부 타이머 현재값(0.0~999.9ms)

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
D	○	○	○	○	○	○	○	○	○	○	–	○		○				

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· N의 데이터가 0~5(CJ1M CPU 유니트의 경우 0~15, CJ2M CPU 유니트의 경우 0~15, 100~107, 110~117)의 범위 내에 없을 때 ON · I/O 인터럽트 지정: CJ2M CPU 유니트에서 인터럽트 입력 유니트가 장착되지 않았을 때 ON · 그 이외일 때 OFF

기능

MSKS(인터럽트 마스크 세트) 명령에 의해 지정된 인터럽트 요인의 상태를 읽습니다. N의 값을 통해 어느 인터럽트에 대해 어떤 상태를 읽을지 지정합니다.

(1) I/O 인터럽트 및 입력 인터럽트의 인터럽트 마스크 상태(N=0~3, 6~9, 100~107)
N으로 지정한 인터럽트 입력 접점에 대해 현재 설정된 인터럽트의 수신 허가/금지 상태를 D의 비트에 출력합니다.

(2) I/O 인터럽트 및 입력 인터럽트의 인터럽트 입력 에지 검출 방향 지정 상태(N=2, 3, 10~13, 110~117)
N으로 지정한 인터럽트 입력 접점에 대해 현재 설정된 에지 검출 방향 지정(기동/종료) 상태를 D의 비트에 출력합니다.

(3) 예약 인터럽트의 내부 타이머 설정값(N=4, 5)
N으로 지정한 예약 인터럽트에 대해 내부 타이머에 설정된 예약 인터럽트 설정 시간을 D에 출력합니다.

(4) 예약 인터럽트의 내부 타이머 현재값(N=14, 15)
N으로 지정한 예약 인터럽트의 내부 타이머 현재값을 D에 출력합니다. 내부 타이머의 현재값은 예약 인터럽트 처리 시작(MSKS 명령 실행)부터 경과한 시간 또는 이전 예약 인터럽트 기동부터의 경과 시간 값입니다(CJ1M CPU 유니트, CJ2M CPU 유니트만 해당).

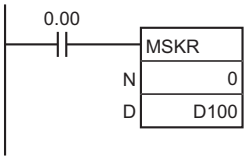
주 1: CJ1M-CPU11/21에서는 예약 인터럽트 태스크를 1개(예약 인터럽트 0(인터럽트 태스크 번호 2))만 사용할 수 있습니다.

주 2: 예약 인터럽트 설정 시간 또는 내부 타이머 현재값으로 읽는 숫자의 단위는 PLC 시스템 설정의 '예약 인터럽트 단위 시간 설정'을 통해 설정합니다(초기값: 10ms).

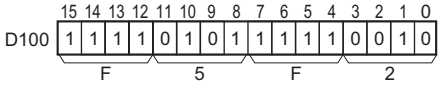
주 3: 이 명령은 사이클릭 태스크와 인터럽트 태스크 내에서 모두 사용할 수 있습니다.

샘플 프로그램

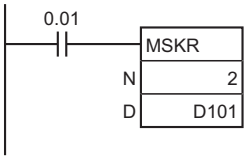
■ I/O 인터럽트 지정 시(CS1W-INT01 또는 CJ1W-INT01 사용 시)



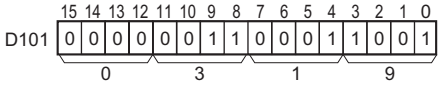
0.00이 ON이면 인터럽트 입력 유니트 번호 0의 인터럽트 수신 허가/금지 상태를 읽어 D100에 저장합니다.



0: 인터럽트 수신/1: 인터럽트 금지

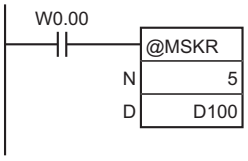


0.01이 ON이면 인터럽트 입력 유니트 번호 0의 인터럽트 입력의 에지 검출 방향이 기동 지정인지, 종료 지정인지를 읽어 D101에 저장합니다.

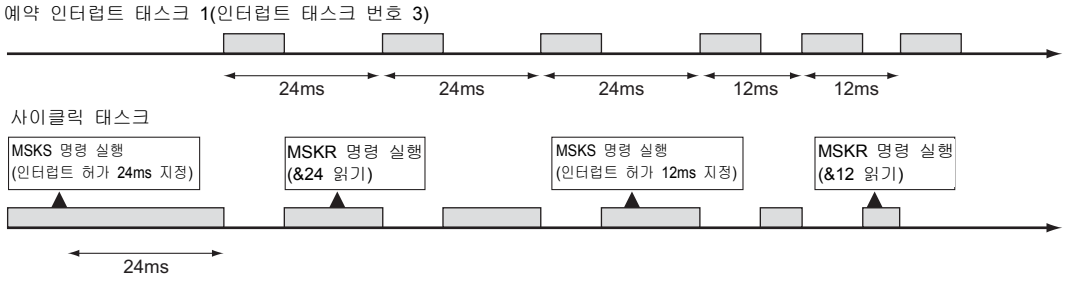


0: 기동 지정/1: 종료 지정

■ 예약 인터럽트 지정 시

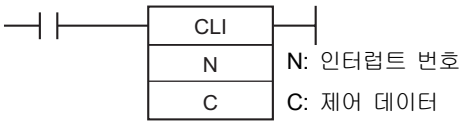


예약 인터럽트 1의 내부 타이머가 동작 중이고 W0.00이 OFF→ON 일 때 현재 설정된 예약 인터럽트 설정 시간을 D100에 저장합니다.



CLI

명령 이름	니모닉	동작 옵션	펑션 번호	기능
인터럽트 해제	CLI	@CLI	691	인터럽트 입력 기억 해제/유지 지정 또는 예약 인터럽트의 첫 번째 인터럽트 시작 시간 지정을 수행합니다.

심볼	CLI
	

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	인터럽트 번호	–	1
C	제어 데이터	UINT	1

(1) I/O 인터럽트 및 입력 인터럽트의 인터럽트 요인의 기억 해제/기억 유지를 지정하는 경우

피연산자	내용	
N	**인터럽트 입력 접점**	**인터럽트 번호**
	인터럽트 입력 유니트(CS1W-INT01 또는 CJ1W-INT01)의 입력 접점(16점/1유니트)	0: 인터럽트 입력 유니트 번호 0 I/O 인터럽트 0~15(인터럽트 태스크 번호 100~115) 1: 인터럽트 입력 유니트 번호 1 I/O 인터럽트 16~31(인터럽트 태스크 번호 116~131)
	인터럽트 입력 유니트(C200HS-INT01)의 입력 접점(8점/1유니트)	0: 인터럽트 입력 유니트 번호 0 I/O 인터럽트 0~7(인터럽트 태스크 번호 100~107) 1: 인터럽트 입력 유니트 번호 1 I/O 인터럽트 8~15(인터럽트 태스크 번호 108~115) 2: 인터럽트 입력 유니트 번호 2 I/O 인터럽트 16~23(인터럽트 태스크 번호 116~123) 3: 인터럽트 입력 유니트 번호 3 I/O 인터럽트 24~31(인터럽트 태스크 번호 124~131)
	CJ1M CPU 유니트(CJ1M-CPU2□)의 내장 입력 접점(4점/1유니트)	6: 입력 인터럽트 0(인터럽트 태스크 번호 140) 7: 입력 인터럽트 1(인터럽트 태스크 번호 141) 8: 입력 인터럽트 2(인터럽트 태스크 번호 142) 9: 입력 인터럽트 3(인터럽트 태스크 번호 143)
C	**인터럽트 입력 접점**	**제어 데이터(요인 처리 데이터)**
	인터럽트 입력 유니트(CS1W-INT01 또는 CJ1W-INT01)의 입력 접점(16점/1유니트)	16진수 #0000~FFFF 0~15비트가 각 인터럽트 입력에 대응 각 비트의 의미 1: 기억 해제 0: 기억 유지
	인터럽트 입력 유니트(C200HS-INT01)의 입력 접점(8점/1유니트)	16진수 #0000~00FF 하위 0~7비트가 각 인터럽트 입력에 대응 각 비트의 의미 1: 기억 해제 0: 기억 유지
	CJ1M CPU 유니트(CJ1M-CPU2□)의 내장 입력 접점(4점/1유니트)	10진수 &0(16진수 #0000): 기억 해제 10진수 &1(16진수 #0001): 기억 유지

(2) 예약 인터럽트의 첫 번째 인터럽트 시작 시간을 지정하는 경우

피연산자	내용	
N	**인터럽트 번호**	
	4: 예약 인터럽트 0(인터럽트 태스크 번호 2) 5: 예약 인터럽트 1(인터럽트 태스크 번호 3) 주: CJ1M-CPU11/21에서는 예약 인터럽트 0만 사용 가능 주: CJ2H CPU 유니트에서 유니트 간 동기 제어 기능을 사용할 때는 예약 인터럽트(0, 1)는 사용 불가능	
C	**예약 인터럽트 단위 시간(PLC 시스템 설정)**	**제어 데이터(첫 번째 인터럽트 시작 시간)**
	10ms	10진수 &0~9999(16진수 #0000~270F): 첫 번째 인터럽트 시작 시간(0~99990ms)
	1ms	10진수 &0~9999(16진수 #0000~270F): 첫 번째 인터럽트 시작 시간(0~9999ms)
	0.1ms	10진수 &0~9999(16진수 #0000~270F): 첫 번째 인터럽트 시작 시간(0~999.9ms)

(3) 고속 카운터 인터럽트 요인의 기억 해제/기억 유지를 지정하는 경우(CJ1M/CPU 유니트만 해당)

피연산자	내용
N	**인터럽트 번호**
	10: 고속 카운터 입력 0 11: 고속 카운터 입력 1
C	**요인 기억 데이터**
	10진수 &0(16진수 #0000): 기억 해제 10진수 &1(16진수 #0001): 기억 유지

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
C	○	○	○	○	○	○	○	○	○	○		○		○				

조건 플래그의 동작

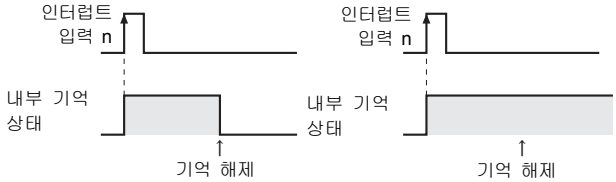
이름	라벨	내용
에러 플래그	P_ER	· N의 데이터가 0~5(CJ1M CPU 유니트의 경우 0, 1, 4~11)의 범위 내에 없을 때 ON · C의 데이터가 16진수 #0000~00FF의 범위 내에 없을 때 ON(I/O 인터럽트 지정 시에 C200HS-INT01을 사용한 경우에만 해당) · C의 데이터가 0~1의 범위 내에 없을 때 ON(I/O 인터럽트 지정 시에 CJ1M CPU 유니트 내장 입력 인터럽트를 사용한 경우 또는 고속 카운터 인터럽트 지정 시) · C의 데이터가 10진수 &0~9999 또는 16진수 #0000~270F의 범위 내에 없을 때 ON(예약 인터럽트 지정 시) · CJ2H CPU 유니트에서 유니트 간 동기 제어 기능 사용 시, N의 데이터가 4, 5(예약 인터럽트 지정)일 때 ON · 그 이외일 때 OFF

기능

N의 값을 통해 I/O 인터럽트 및 입력 인터럽트 요인의 기억 해제/기억 유지를 수행할지, 예약 인터럽트의 첫 번째 인터럽트 시작 시간을 설정할지, 고속 카운터 인터럽트 요인의 기억 해제/기억 유지(CJ1M CPU 유니트 내장 고속 카운터 입력 사용 시에만 해당)를 수행할지 여부를 지정합니다.

(1) I/O 인터럽트 및 입력 인터럽트 요인의 기억 해제/유지(N=0~3, 6~9, 100~107)

N으로 지정된 인터럽트 입력에 대해 C의 비트로 인터럽트 요인의 기억을 해제 또는 유지합니다.

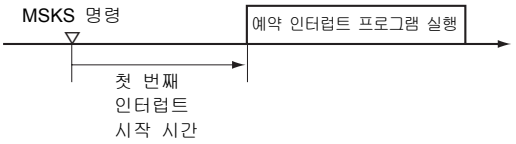


인터럽트 태스크 실행 중에 다른 인터럽트 입력이 수신된 경우, 대응하는 인터럽트 태스크의 실행이 보류되고 일단 해당 인터럽트 요인이 내부에 기억됩니다. 실행 중인 인터럽트 태스크가 종료되면 해당 시점에서 기억이 유지된 모든 인터럽트 요인을 대상으로 우선 순위가 높은 것부터 실행됩니다.

실행이 보류된 I/O 인터럽트 태스크 또는 입력 인터럽트 태스크를 취소하고 싶으면 이 CLI(인터럽트 해제) 명령을 통해 내부에 기억된 입력 인터럽트 요인을 해제합니다.

(2) 예약 인터럽트의 첫 번째 인터럽트 시작 시간 설정(N=4, 5)

N으로 지정된 예약 인터럽트에 대해 C의 값을 첫 번째 인터럽트 시작 시간으로 지정합니다.



주 1: CJ1M-CPU11/21에서는 예약 인터럽트 태스크를 1개(예약 인터럽트 0(인터럽트 태스크 번호 2))만 사용할 수 있습니다.

주 2: 첫 번째 인터럽트 시작 시간으로 지정하는 숫자의 단위는 PLC 시스템 설정의 '예약 인터럽트 단위 시간 설정'을 통해 설정합니다(초기값: 10ms).

(3) 고속 카운터 인터럽트 요인의 기억 해제/유지(N = 10~13)

N으로 지정된 내장 고속 카운터 인터럽트(목표값 일치 비교 또는 대역 비교)에 대해 인터럽트 요인의 기억을 해제 또는 유지합니다(CJ1M CPU 유니트만 해당).

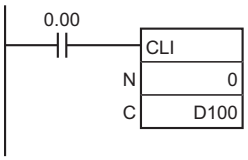
주 3: CJ1M CPU 유니트에서는 고속 카운터 입력 0, 1만 사용할 수 있습니다.

사용상의 주의 사항

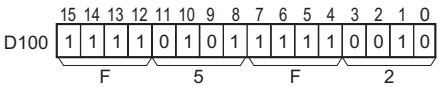
CJ2H CPU 유니트에서 유니트 간 동기 제어 기능을 사용할 때는 예약 인터럽트 태스크를 사용할 수 없습니다. CLI 명령의 인터럽트 번호에 N=4~5를 지정하면 명령 에러가 됩니다.

샘플 프로그램

■ I/O 인터럽트 지정 시(CS1W-INT01/CJ1W-INT01 사용 시)



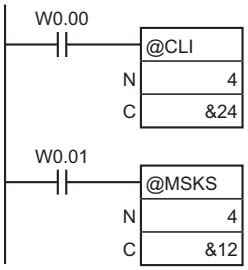
0.00이 ON인 경우 인터럽트 입력 유니트 번호 0 각 인터럽트 입력에 대해 인터럽트 요인의 기억을 해제 또는 유지합니다.



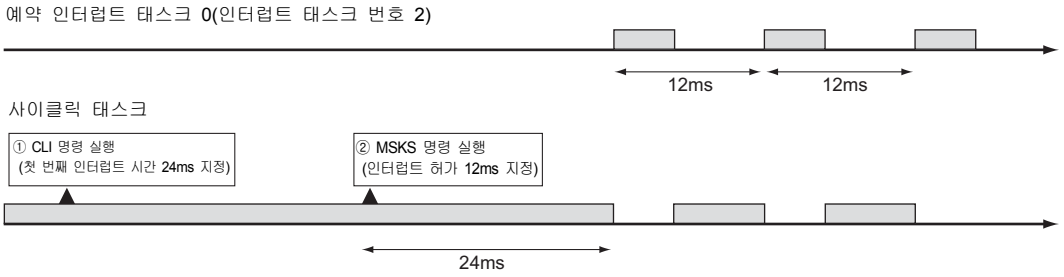
0: 인터럽트 입력 기억 유지/1: 인터럽트 입력 기억 해제

CLI

■ 예약 인터럽트 지정 시

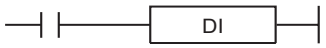


① W0.00이 OFF→ON이면 예약 인터럽트 0의 첫 번째 인터럽트 시작 시간을 24ms로 설정합니다(예약 인터럽트 단위 시간 설정: 1ms일 때).
② W0.01이 OFF→ON이면 예약 인터럽트 0의 인터럽트 간격 시간을 12ms로 설정하여 내부 타이머를 기동합니다(예약 인터럽트 단위 시간 설정: 1ms일 때).



DI

명령 이름	니모닉	동작 옵션	평션 번호	기능
인터럽트 태스크 실행 금지	DI	@DI	693	모든 인터럽트 태스크(전원 OFF 인터럽트 태스크 제외)의 실행을 일시적으로 금지합니다.

심볼	DI
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	불가능	가능

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 인터럽트 태스크에서 실행했을 때 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

이름	주소	내용
전원 OFF 금지 구간 설정	A530CH	16진수 #A5A5: 전원 OFF 금지 구간 설정 유효 DI－EI 명령 사이는 전원 OFF 처리(전원 OFF 인터럽트 태스크 실행은 제외)를 마스크하고 EI 명령까지 실행합니다. 16진수 #A5A5 이외: 전원 OFF 금지 구간 설정 무효

기능

사이클릭 태스크 내에서 사용하며, 전원 OFF 인터럽트 태스크 이외의 모든 인터럽트 태스크(I/O 인터럽트 태스크, 예약 인터럽트 태스크, 외부 인터럽트 태스크, 입력 인터럽트 태스크, 고속 카운터 인터럽트 태스크)의 실행을 인터럽트 태스크 실행 금지 해제(EI)가 실행될 때까지 일시적으로 금지(보류)합니다.

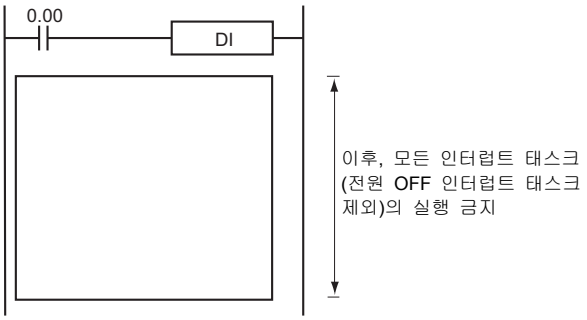
사이클릭 태스크의 프로그램 처리로 인터럽트 태스크를 실행하고 싶지 않은 타이밍에 사용합니다.
금지 중에 발생한 인터럽트 요인은 내부에서 기억되며, 금지가 해제된 시점에서 대응하는 인터럽트 태스크가 실행됩니다.

CS1-H/CJ1-H/CJ1M/CS1D(CPU 단독 시스템용)/CJ2 CPU 유니트의 경우, 전원 OFF 금지 구간 설정 유효(A530CH(전원 OFF 금지 구간 설정)가 16진수 #A5A5)일 때 전원 OFF 처리 실행도 동시에 금지합니다. 이 명령 실행 후 전원 OFF를 확정해도 EI 명령 또는 최종 태스크의 END 명령까지의 명령군을 실행한 후 CPU 유니트를 리셋합니다.
단, 전원 OFF 인터럽트 태스크가 유효로 설정된 경우는 전원 OFF 인터럽트 태스크를 실행한 후 CPU 유니트를 리셋합니다.
자세한 내용은 SYSMAC CS/CJ 시리즈 사용 설명서 설치편의 '10-3-1 전원 OFF 시의 명령 실행' 또는 SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편의 '12-2-2 전원 OFF 시의 명령 실행'을 참조하십시오.

사용상의 주의 사항

· 금지 상태는 EI(인터럽트 태스크 실행 금지 해제) 명령으로 해제합니다.
· 이 명령은 인터럽트 태스크 내에서 실행할 수 없습니다.

샘플 프로그램



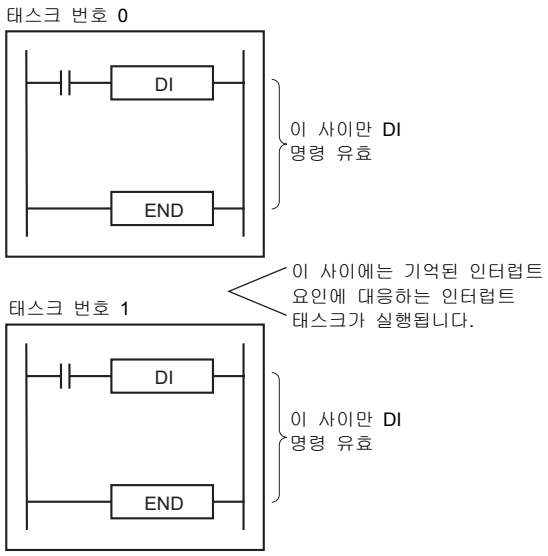
0.00이 ON일 때 전원 OFF 인터럽트 태스크 이외의 모든 인터럽트 태스크의 실행을 금지합니다.

주: CS1-H/CJ1-H/CJ1M/CS1D(CPU 단독 시스템용)/CJ2 CPU 유니트에서 전원 OFF 인터럽트 태스크가 무효인 경우, 전원 OFF 처리도 동시에 금지할 수 있습니다.

■ 이 명령을 통해 전원 OFF 인터럽트 태스크 이외의 모든 인터럽트 태스크를 실행 금지하는 경우

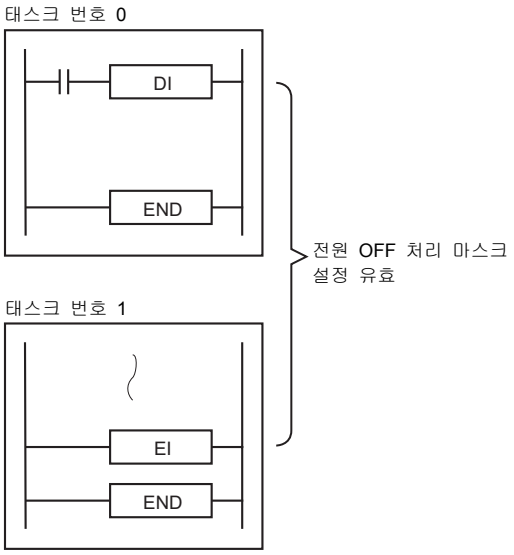
사이클릭 태스크 실행 중에는 금지를 실행할 수 없습니다.
여러 사이클릭 태스크에서 금지하려는 경우는 각각의 사이클릭 태스크에 이 명령을 삽입하십시오.

단, 하나의 사이클릭 태스크가 종료하면 반드시 금지 상태가 해제되므로 기억된 인터럽트 요인이 존재하는 경우는 이 타이밍에 인터럽트 태스크가 실행됩니다.



■ 이 명령을 통해 전원 OFF 처리 실행을 금지한 경우

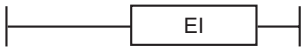
사이클릭 태스크 실행 중에 금지할 수 있습니다.
(기동 중인 모든 태스크 실행 후에 해제됩니다.)



주: CS1-H/CJ1-H/CJ1M/CS1D(CPU 단독 시스템용)/CJ2 CPU 유니트이고, 전원 OFF 인터럽트 태스크가 무효이며, A530CH가 16진수 #A5A5인 경우는 DI 명령~EI 명령 사이에서 전원 OFF가 확정되면 EI 명령이 실행될 때 CPU 유니트를 리셋합니다.

EI

명령 이름	니모닉	동작 옵션	펑션 번호	기능
인터럽트 태스크 실행 금지 해제	EI	−	694	모든 인터럽트 태스크(전원 OFF 인터럽트 태스크 제외)의 실행 금지를 해제합니다.

심볼	EI
	

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	불가능	가능

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 인터럽트 태스크에서 실행했을 때 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

이름	주소	내용
전원 OFF 금지 구간 설정	A530CH	16진수 #A5A5: 전원 OFF 금지 구간 설정 유효 DI−EI 명령 사이 또는 DI−END 명령 사이는 전원 OFF 처리(전원 OFF 인터럽트 태스크 실행은 제외)를 마스크하고 EI 명령 또는 END 명령까지 실행합니다. 16진수 #A5A5 이외: 전원 OFF 처리 마스크 무효

기능

· 사이클릭 태스크 내에서 사용하며, 전원 OFF 인터럽트 태스크 이외에 DI(인터럽트 태스크 실행 금지) 명령에 의해 실행 금지된 모든 인터럽트 태스크(I/O 인터럽트 태스크, 예약 인터럽트 태스크, 외부 인터럽트 태스크, 입력 인터럽트 태스크, 고속 카운터 인터럽트 태스크)의 실행 금지를 해제합니다.

· CS1-H/CJ1-H/CJ1M/CS1D(CPU 단독 시스템용)/CJ2 CPU 유니트이고 DI 명령에서 전원 OFF 처리가 금지된 경우는 전원 OFF 처리 실행 금지도 해제합니다. DI 명령 실행 후 전원 OFF를 확정해도 그 시점에서 CPU 유니트를 리셋하지 않고 DI 명령의 다음부터 이 명령 직전까지의 명령군을 모두 실행한 후에 CPU 유니트를 리셋합니다.
DI 명령으로 전원 OFF 처리를 금지하는 방법은 DI 명령을 참조하십시오.

사용상의 주의 사항

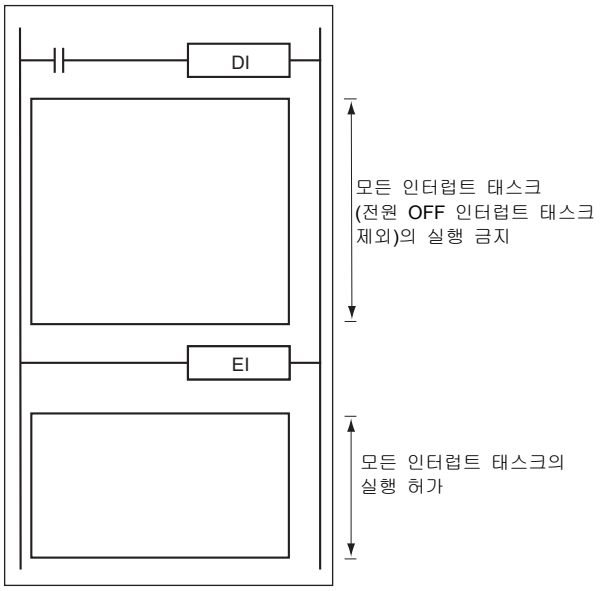
· 이 명령은 입력 조건(전원 흐름)이 필요하지 않습니다.

· 이 명령은 DI(인터럽트 태스크 실행 금지 해제) 명령으로 금지 상태를 해제합니다. MSKS(인터럽트 마스크 세트) 명령을 통해 I/O 인터럽트 가능(수신 가능) 또는 예약 인터럽트 처리 가능으로 설정하지 않은 인터럽트 태스크 번호인 경우에는 이 명령을 통해 수신 가능하게 되지 않습니다.

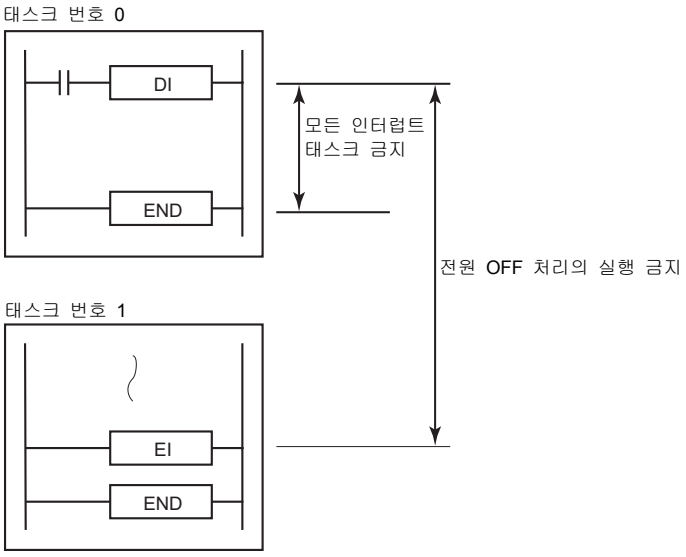
· 이 명령은 인터럽트 태스크 내에서 실행할 수 없습니다.

EI

샘플 프로그램



주: CS1-H/CJ1-H/CJ1M/CS1D(CPU 단독 시스템용)/CJ2 CPU 유니트이고 전원 OFF 인터럽트 태스크가 무효인 경우는 전원 OFF 처리의 실행 금지도 동시에 해제합니다.

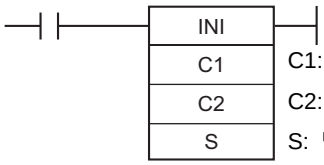


고속 카운터/펄스 출력 명령

니모닉	명령 이름	펑션 번호	페이지
INI	동작 모드 제어	880	3-604
PRV	펄스 현재값 읽기	881	3-607
PRV2	펄스 주파수 변환	883	3-612
CTBL	비교 테이블 등록	882	3-615
SPED	주파수 설정	885	3-618
PULS	펄스량 설정	886	3-621
PLS2	위치 결정	887	3-623
ACC	주파수 가감속 제어	888	3-628
ORG	원점 검색	889	3-632
PWM	PWM 출력	891	3-635

INI

명령 이름	니모닉	동작 옵션	평션 번호	기능
동작 모드 제어	INI	@INI	880	비교 테이블과의 시작/정지, 고속 카운터 현재값 변경, 인터럽트 입력(카운터 모드)의 현재값 변경, 펄스 출력 현재값 변경(0으로 원점 확정), 펄스 출력 정지 중 하나를 실행합니다.

심볼	INI
	 C1: 포트 지정 C2: 제어 데이터 S: 변경 데이터 저장 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C1	포트 지정	WORD	1
C2	제어 데이터	UINT	1
S	변경 데이터 저장 하위 CH 번호	DWORD	2

C1: 포트 지정

포트 지정(16진수)	
#0000	펄스 출력 0
#0001	펄스 출력 1
#0010	고속 카운터 입력 0
#0011	고속 카운터 입력 1
#0100	인터럽트 입력 0(카운터 모드)
#0101	인터럽트 입력 1(카운터 모드)
#0102	인터럽트 입력 2(카운터 모드)
#0103	인터럽트 입력 3(카운터 모드)
#1000	PWM 출력 0
#1001	PWM 출력 1

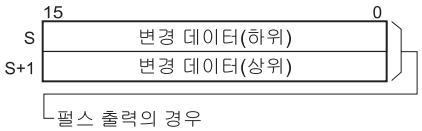
C2: 제어 데이터

제어 데이터(16진수)	
#0000	비교 시작
#0001	비교 정지
#0002	현재값 변경
#0003	펄스 출력 정지

S: 변경 데이터 저장 하위 CH 번호

현재값 변경(C2 = #0002)을 지정했을 때 변경 데이터를 저장합니다.

현재값 변경 이외를 지정한 경우 이 피연산자의 값은 사용하지 않습니다.



펄스 출력의 경우
-2147.483648~2147.483647 또는 16진수 #80000000~7FFFFFFF
고속 카운터 입력의 경우
-2147.483648~2147.483647 또는 16진수 #80000000~7FFFFFFF
10진수 &0~4294.967295 또는 16진수 #00000000~FFFFFFFF
인터럽트 입력(카운터 모드)의 경우
10진수 &0~65535 또는 16진수 #00000000~0FFFF

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C1, C2	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
S	○	○	○	○	○	○	○		○	○	–			○				

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C1, C2, S로 지정할 수 있는 범위를 초과했을 때 ON · C1과 C2가 대응하지 않는 경우 ON · 비교 테이블이 등록되지 않은 상태에서 비교 시작을 지정한 경우 ON · 펄스 출력 중인 포트에 현재값 변경을 지정한 경우 ON · 고속 카운터로 설정되지 않은 포트에 대해 고속 카운터 현재값 변경을 지정한 경우 ON · 인터럽트 입력(카운터 모드)의 현재값 변경 시, 범위 밖의 값을 지정한 경우 ON · CTBL 명령 실행 중에 인터럽트가 발생하여 인터럽트 태스크 내에서 고속 카운터 입력 지정의 INI 명령을 실행한 경우 ON · 인터럽트 입력(카운터 모드)으로 설정되지 않은 포트에 대해 명령을 실행한 경우 ON · 그 이외일 때 OFF

기능

C1로 지정된 포트에 대해 C2로 지정된 제어를 실행합니다.

지정할 수 있는 C1과 C2의 조합은 다음과 같습니다.

C1(포트 지정)	C2(제어 데이터)			
	비교 시작 (#0000)	비교 정지 (#0001)	현재값 변경 (#0002)	펄스 출력 정지 (#0003)
펄스 출력 (#0000, #0001)	×	×	○	○
고속 카운터 입력 (#0010, #0011)	○	○	○	×
인터럽트 입력(카운터 모드) (#0100, #0101, #0102, #0103)	×	×	○	×
PWM 출력 (#1000, #1001)	×	×	×	○

■ 비교 시작(C2 = #0000)

비교 테이블 등록(CTBL) 명령으로 등록한 비교 테이블과 고속 카운터 현재값의 비교를 시작합니다.

주: 비교 테이블이 등록되지 않았을 때는 이 지령이 에러가 되어 실행할 수 없습니다.

■ 비교 정지(C2 = #0001)

비교 테이블 등록(CTBL) 명령으로 등록한 비교 테이블과 고속 카운터 현재값의 비교를 정지합니다.

INI

■ 현재값 변경(C2 = #0002)

제어 대상			제어 내용	변경 가능 범위
펄스 출력 (C1=#0000, #0001)			펄스 출력 현재값을 변경합니다. S+1, S에 변경할 값을 설정합니다. 주: 이 지령은 펄스 출력 정지 중에만 실행할 수 있습니다. 펄스 출력 중에 실행한 경우는 에러가 됩니다.	-2147483648~2147483647 또는 16진수 #80000000~7FFFFFFF
고속 카운터 입력 (C1=#0010, #0011)	선형 모드 시	위상차 입력/가감산 펄스 입력/펄스 + 방향 입력	고속 카운터 현재값을 변경합니다. S+1, S에 변경할 값을 설정합니다. 주: 지정한 포트가 고속 카운터 설정이 되지 않은 경우, 이 지령은 에러가 됩니다.	-2147483648~2147483647 또는 16진수 #80000000~7FFFFFFF
		가산 펄스 입력		10진수 &0~4294967295 또는 16진수 #00000000~FFFFFFFF
	링 모드 시			10진수 &0~4294967295 또는 16진수 #00000000~FFFFFFFF
인터럽트 입력(카운터 모드) (C1=#0100, #0101, #0102, #0103)			인터럽트 입력(카운터 모드) 현재값을 변경합니다. S+1, S에 변경할 값을 설정합니다.	10진수 &0~65535 또는 16진수 #00000000~0000FFFF 주: 위 범위 밖의 값을 지정한 경우 에러

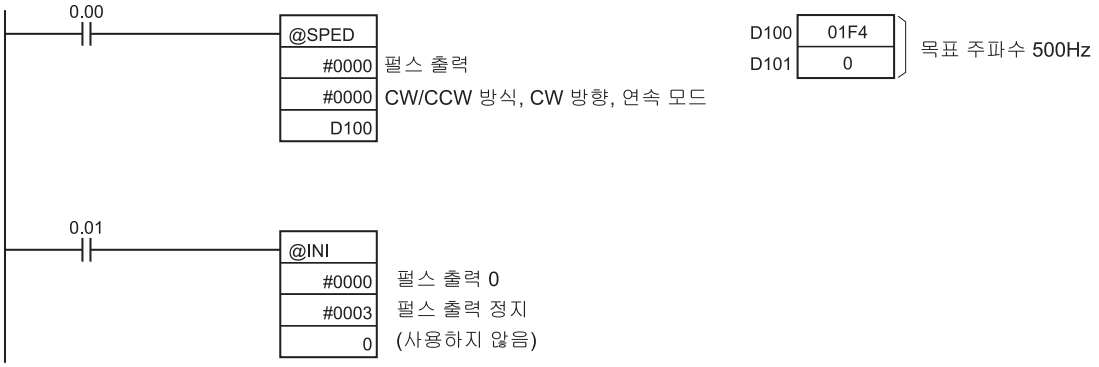
■ 펄스 출력 정지(C1 = #1000, #1001, C2 = #0003)

지정한 포트의 펄스 출력을 정지합니다(즉시 정지).

또한 펄스 출력 정지 중에 이 지령을 실행하면 펄스량 설정이 클리어됩니다.

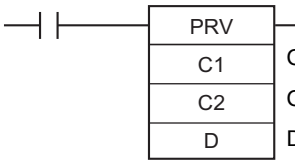
샘플 프로그램

0.00이 OFF→ON이면 SPED 명령을 사용하여 펄스 출력 0에서 연속 모드로 500Hz의 펄스 출력을 시작합니다. 0.01이 OFF→ON이 되면 INI 명령을 사용하여 펄스 출력을 정지합니다.



PRV

명령 이름	니모닉	동작 옵션	펑션 번호	기능
펄스 현재값 읽기	PRV	@PRV	881	고속 카운터 현재값, 펄스 출력 현재값, 인터럽트 입력(카운터 모드)의 현재값을 읽습니다.

심볼	PRV
	

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C1	포트 지정	–	1
C2	제어 데이터	–	1
D	현재값 저장 하위 CH 번호	WORD	가변

C1: 포트 지정

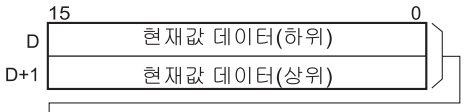
포트 지정(16진수)	
#0000	펄스 출력 0
#0001	펄스 출력 1
#0010	고속 카운터 입력 0
#0011	고속 카운터 입력 1
#0100	인터럽트 입력 0(카운터 모드)
#0101	인터럽트 입력 1(카운터 모드)
#0102	인터럽트 입력 2(카운터 모드)
#0103	인터럽트 입력 3(카운터 모드)
#1000	PWM 출력 0
#1001	PWM 출력 1

C2: 제어 데이터

제어 데이터(16진수)	
#0000	현재값 읽기
#0001	상태 읽기
#0002	대역 비교 결과 읽기
#00□3	C1 = #0000 또는 #0001의 경우: 펄스 출력 0 또는 1의 주파수 읽기 C1 = #0010의 경우: 고속 카운터 입력 0의 주파수 읽기 #0003: 일반 방식 #0013: 고주파수 대응 및 10ms 샘플링 방식(CJ1M CPU 유니트의 유니트 Ver.3.0 이상만 가능) #0023: 고주파수 대응 및 100ms 샘플링 방식(CJ1M CPU 유니트의 유니트 Ver.3.0 이상만 가능) #0033: 고주파수 대응 및 1s 샘플링 방식(CJ1M CPU 유니트의 유니트 Ver.3.0 이상만 가능)

PRV

D: 현재값 저장 하위 CH 번호



2채널 출력
펄스 출력, 고속 카운터 입력에 대한 현재값 읽기의 경우
입력 주파수 읽기(고속 카운터 0만 해당)의 경우



1채널 출력
인터럽트 입력(카운터 모드)에 대한 현재값 읽기의 경우
상태 읽기의 경우
대역 비교 결과 읽기의 경우

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C1, C2	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
D	○	○	○	○	○	○	○		○	○	–			○				

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C1, C2로 지정할 수 있는 범위를 초과했을 때 ON · C1과 C2가 대응하지 않는 경우 ON · 대역 비교 동작 중이 아닐 때 대역 비교 결과 읽기를 지정한 경우 ON · 고속 카운터 0 이외에 대해 입력 주파수 읽기를 지정한 경우 ON · 고속 카운터로 설정되지 않은 포트에 대해 명령을 실행한 경우 ON · 인터럽트 입력(카운터 모드)으로 설정되지 않은 포트에 대해 명령을 실행한 경우 ON · 그 이외일 때 OFF

기능

C1로 지정된 포트에 대해 C2로 지정된 데이터를 읽습니다.
지정할 수 있는 C1과 C2의 조합은 다음과 같습니다.

C1(포트 지정)	C2(제어 데이터)		
	현재값 읽기 (#0000)	상태 읽기 (#0001)	대역 비교 결과 읽기 (#0002)
펄스 출력 (#0000, #0001)	○	○	×
고속 카운터 입력 (#0010, #0011)	○	○	○
인터럽트 입력 (카운터 모드) (#0100, #0101, #0102, #0103)	○	×	×
PWM 출력 (#1000, #1001)	×	○	×

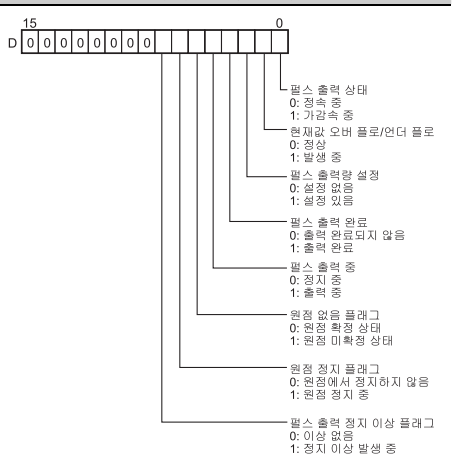
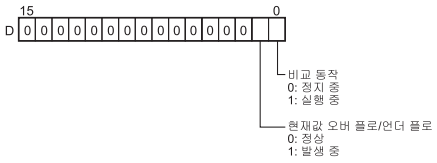
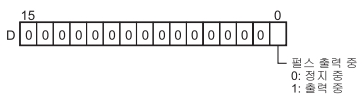
C1(포트 지정)	주파수 읽기(#00□3)			
	일반 방식 (#0003)	10ms 샘플링 방식(#0013)	100ms 샘플링 방식 (#0023)	1s 샘플링 방식 (#0033)
펄스 출력 (#0000, #0001)	○*	×	×	×
고속 카운터 입력 (#0010, #0011)	○ (고속 카운터 0만 가능)	○* (고속 카운터 0만 가능)	○* (고속 카운터 0만 가능)	○* (고속 카운터 0만 가능)
인터럽트 입력 (카운터 모드) (#0100, #0101, #0102, #0103)	×	×	×	×
PWM 출력 (#1000, #1001)	×	×	×	×

* CJ1M CPU 유니트의 유니트 Ver.3.0 이상만 가능

■ 현재값 읽기(C2 = #0000)

대상		내용	읽기 결과 범위
펄스 출력 (C1=#0000, #0001)		펄스 출력 현재값을 읽어 D+1, D에 저장합니다.	-2147483648~2147483647 또는 16진수 #80000000~7FFFFFFF
고속 카운터 입력 (C1=#0010, #0011)	선형 모드 시	고속 카운터 현재값을 읽어 D+1, D에 저장합니다.	-2147483648~2147483647 또는 16진수 #80000000~7FFFFFFF
	링 모드 시		10진수 &0~4294967295 또는 16진수 #00000000~FFFFFFFF
인터럽트 입력(카운터 모드) (C1 = #0100, #0101, #0102, #0103)		인터럽트 입력(카운터 모드)의 현재값을 읽어 D에 저장합니다.	10진수 &0~65535 또는 16진수 #00000000~0000FFFF

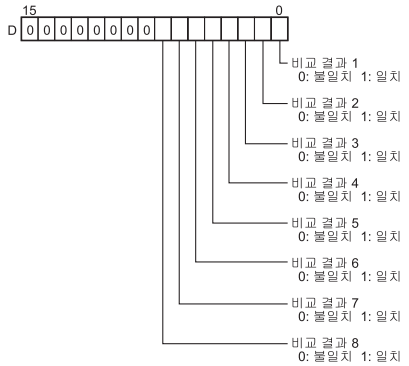
■ 상태 읽기(C2 = #0001)

대상	내용	읽기 결과 범위
펄스 출력	펄스 출력 상태를 읽어 D에 저장합니다.	
고속 카운터 입력	고속 카운터의 상태를 읽어 D에 저장합니다.	
PWM 출력	PWM 출력 상태를 읽어 D에 저장합니다.	

PRV

■ 대역 비교 결과 읽기(C2 = #0002)

고속 카운터를 대역 비교형으로 비교할 때 PRV 명령 실행 시의 비교 결과를 읽어 D에 저장합니다.



■ 펄스 출력 또는 고속 카운터 주파수 읽기(C2 = #00□3)

펄스 출력 0 또는 1에서 출력되는 펄스의 주파수(Hz) 또는 고속 카운터 0에 입력되는 펄스의 주파수(Hz)를 읽어 D+1, D에 저장합니다.

C1 = #0000 또는 #0001의 경우: 펄스 출력 0 또는 1의 주파수 읽기

변환 결과 = 10진수 &0~100000 또는 16진수 #00000000~000186A0

C1 = #0010의 경우: 고속 카운터 입력 0의 주파수 읽기

카운터 입력 방식이 위상차 4체배 이외일 때:

변환 결과 = 10진수 &0~100000 또는 16진수 #00000000~000186A0

주: 100kHz를 넘어가는 주파수가 입력된 경우, 출력되는 값은 최대값(10진수 &100000)으로 고정됩니다.

카운터 입력 방식이 위상차 4체배 일 때:

변환 결과 = 10진수 &0~200000 또는 16진수 #00000000~00030D40

주: 200kHz를 넘어가는 주파수가 입력된 경우, 출력되는 값은 최대값(10진수 &200000)으로 고정됩니다.

● 펄스 주파수 산출 방식

CJ1M CPU 유니트의 유니트 Ver.3.0 이상인 경우, 펄스 출력 0 또는 1에서 출력되거나 고속 카운터 0에 입력되는 펄스의 주파수(Hz)를 산출하는 방식에는 크게 다음의 2가지 방식이 있습니다.

- 일반 방식(기존 방식): 주파수의 크기에 관계 없이 항상 1펄스당 카운트 클록의 카운트 수로 산출하는 방식입니다. 높은 주파수에서 펄스의 기동/종료가 둔화되어 오차가 발생합니다(참고: 100kHz에서 최대 1%의 오차).
- 고주파수 대응 방식: 고주파수와 저주파수에서 계측 방식을 전환하는 방식입니다.

1kHz 이상의 고주파수인 경우, 일정 시간(샘플링 시간) 내의 카운트 펄스를 계측하여 주파수를 산출합니다. 샘플링 시간의 폭에 맞춰 다음 3종류에서 선택할 수 있습니다. C2의 아래 2번째 자리를 사용하여 지정합니다.

- 고주파수 대응 및 10ms 샘플링 방식(10ms당 펄스 수를 카운트하는 방식):
  C2 = 16진수 #0013(참고: 1kHz에서 최대 10%의 오차)
- 고주파수 대응 및 100ms 샘플링 방식(100ms당 펄스 수를 카운트하는 방식):
  C2 = 16진수 #0023(참고: 1kHz에서 최대 1%의 오차)
- 고주파수 대응 및 1s 샘플링 방식(1s당 펄스 수를 카운트하는 방식):
  C2 = 16진수 #0033(참고: 1kHz에서 최대 0.1%의 오차)

1kHz 미만의 저주파수는 일반 방식으로 주파수를 산출합니다.

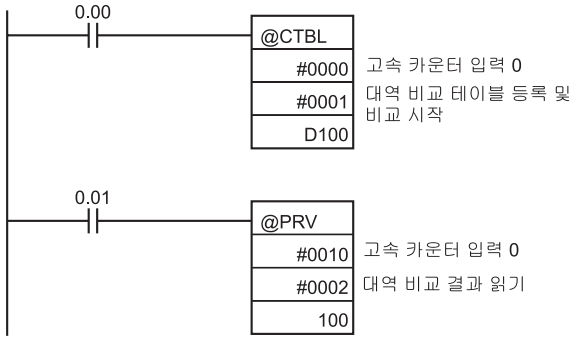
사용상의 주의 사항

- C1=0010 Hex(고속 카운터 입력 0)를 지정하고 C2=0013 Hex, 0023 Hex 또는 0033 Hex(주파수 읽기의 고주파수 대응)를 지정한 경우, 카운터 값을 리셋하면 해당 샘플링 시간 내에 읽은 데이터가 일정치 않게 되므로 주의하십시오.

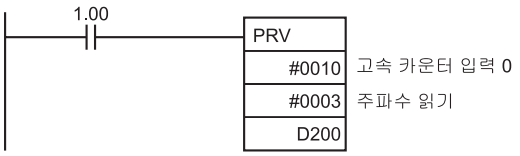
샘플 프로그램

0.00이 OFF→ON이면 CTBL 명령으로 고속 카운터 입력 0에 대해 대역 비교 테이블을 등록하여 비교를 시작합니다.

0.01이 OFF→ON이 되면 PRV 명령으로 해당 시점의 대역 비교 결과를 0100에 읽어옵니다.

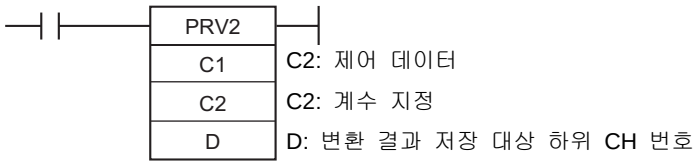


1.00이 ON이면 PRV 명령으로 해당 시점에서 고속 카운터 입력 0에 입력되는 펄스의 주파수를 읽어 D201, D200에 16진수로 출력합니다.



PRV2

명령 이름	니모닉	동작 옵션	평션 번호	기능
펄스 주파수 변환	PRV2	@PRV2	883	고속 카운터에 입력된 펄스 주파수를 읽어서 회전 속도(회전 수)로 변환하거나 카운터 현재값을 누적 회전 수로 변환하여 결과를 16진수 8자리로 출력합니다.

	PRV2
심볼	

특정 영역 내의 사용 여부

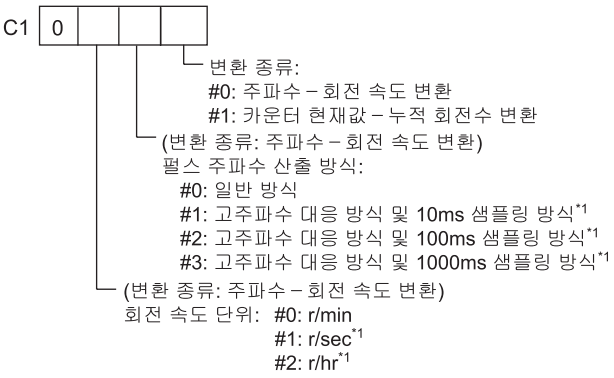
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C1	제어 데이터	WORD	1
C2	계수 지정	UINT	1
D	변환 결과 저장 대상 하위 CH 번호	UDINT	2

C1: 제어 데이터

제어 데이터(16진수)	
#0□ * 0	주파수 – 회전 속도 변환 (□는 단위, *는 주파수 산출 방식을 지정)
#0001	카운터 현재값 – 누적 회전 수 변환

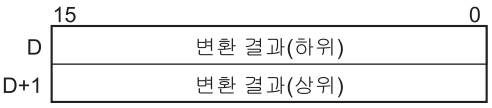


*1: CJ1M CPU 유니트의 유니트 Ver.3.0 이상만 가능

C2: 계수 지정

10진수 &1~65535 또는 16진수 #0001~FFFF: 1회전당 펄스 수

D: 변환 결과 저장 대상 하위 CH 번호



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
C1	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
C2	○	○	○	○	○	○	○	–	○	○		○						
D											–	–		○				

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 고속 카운터 0을 사용하지 않는 설정이 되었을 때 ON · 제어 데이터가 범위 밖인 경우 ON · 계수 지정이 0인 경우 ON · 위 이외의 경우 OFF

기능

고속 카운터 0에 입력된 펄스 주파수를 C2로 지정한 계수를 사용하여 C1의 변환 방법으로 D에 출력합니다. C1을 통해 다음 2가지 변환 방법 중 하나를 선택합니다.

■ 주파수 – 회전 속도 변환(C1 = 16진수 #0□ * 0)의 경우

(□는 단위, *는 주파수 산출 방식을 지정)

1) 회전 속도 단위:

회전 속도 단위에 따라 다음 3종류가 있습니다.

· □ = 0일 때 단위 r/min: 주파수 데이터와 1회전당 펄스 수에서 회전 속도(r/min)를 산출합니다.

변환 결과(r/min) = (주파수/1회전당 펄스 수)×60

· □ = 1일 때 단위 r/s: 주파수 데이터와 1회전당 펄스 수에서 회전 속도(r/sec)를 산출합니다(CJ1M CPU 유니트의 유니트 Ver.3.0 이상만 가능).

변환 결과(r/sec) = (주파수/1회전당 펄스 수)

· □ = 2일 때 단위 r/hr: 주파수 데이터와 1회전당 펄스 수에서 회전 속도(r/hr)를 산출합니다(CJ1M CPU 유니트의 유니트 Ver.3.0 이상만 가능).

변환 결과(r/hr) = (주파수/1회전당 펄스 수)×60×60

· 카운터 입력 방식이 위상차 4체배 이외일 때:

변환 결과 = 10진수 &0~100000 또는 16진수 #00000000~000186A0

주: 100kHz를 넘어가는 주파수가 입력된 경우, 출력되는 값은 최대값(10진수 &100000)으로 고정됩니다.

· 카운터 입력 방식이 위상차 4체배 일 때:

변환 결과 = 10진수 &0~200000 또는 16진수 #00000000~00030D40

주: 200kHz를 넘어가는 주파수가 입력된 경우, 출력되는 값은 최대값(10진수 &200000)으로 고정됩니다.

2) 펄스 주파수 산출 방식:

CJ1M CPU 유니트의 유니트 Ver.3.0 이상인 경우, 고속 카운터 0에 입력되는 펄스의 주파수(Hz)를 산출하는 방식에는 크게 다음의 2가지 방식이 있습니다.

· 일반 방식(기존 방식):

C1 = 16진수 #0□00일 때

주파수의 크기에 관계 없이 항상 1펄스당 카운트 클록의 카운트 수로 산출하는 방식입니다. 높은 주파수에서 펄스의 기동/종료가 둔화되어 오차가 발생합니다(참고: 100kHz에서 최대 1%의 오차).

· 고주파수 대응 방식:

고주파수와 저주파수에서 계측 방식을 전환하는 방식입니다(CJ1M CPU 유니트의 유니트 Ver.3.0 이상만 가능).

1kHz 이상의 고주파수인 경우, 일정 시간(샘플링 시간) 내의 카운트 펄스를 계측하여 주파수를 산출합니다. 샘플링 시간의 폭에 맞춰 다음 3종류에서 선택할 수 있습니다. C1의 아래 2번째 자리를 사용하여 지정합니다.

· 고주파수 대응 및 10ms 샘플링 방식(10ms당 펄스 수를 카운트하는 방식): C1 = 16진수 #0□10일 때(참고: 1kHz에서 최대 10%의 오차)

PRV2

- 고주파수 대응 및 100ms 샘플링 방식(100ms당 펄스 수를 카운트하는 방식): C2 = 16진수 #0□20일 때 (참고: 1kHz에서 최대 1%의 오차)
- 고주파수 대응 및 1s 샘플링 방식(1s당 펄스 수를 카운트하는 방식): C2 = 16진수 #0□30일 때(참고: 1kHz에서 최대 0.1%의 오차)
  1kHz 미만의 저주파수는 일반 방식으로 주파수를 산출합니다.

■ 카운터 현재값 – 누적 회전 수 변환(C2 = 16진수 #0001)의 경우

카운터 현재값과 1회전당 펄스 수에서 누적 회전 수를 산출합니다.

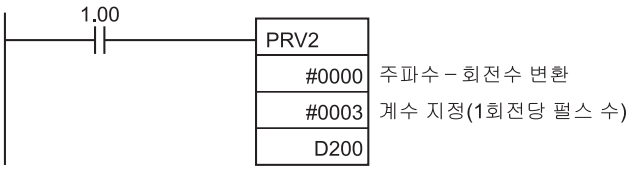
변환 결과 = (카운터 현재값/1회전당 펄스 수)

사용상의 주의 사항

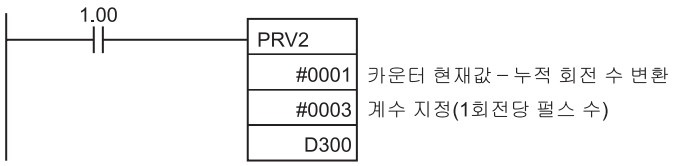
- C1에서 주파수 – 회전 속도 변환과 고주파수 대응 방식을 지정한 경우, 카운터 값을 리셋하면 해당 샘플링 시간 내에 읽은 데이터가 일정치 않게 되므로 주의하십시오.

샘플 프로그램

1.00이 ON이면 PRV2 명령으로 해당 시점에서 고속 카운터 입력 0에 입력되는 펄스의 주파수를 읽어 회전 속도(r/min)로 변환하여 D201, D200에 16진수로 출력합니다.

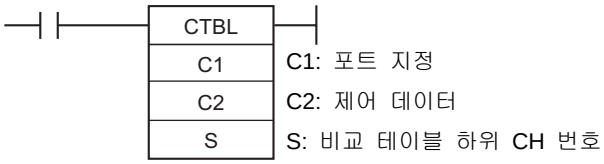


1.00이 ON이면 PRV2 명령으로 해당 시점의 카운터 현재값을 읽어 누적 회전 수로 변환하여 D301, D300에 16진수로 출력합니다.



CTBL

명령 이름	니모닉	동작 옵션	평션 번호	기능
비교 테이블 등록	CTBL	@CTBL	882	고속 카운터 현재값에 대해 목표값 일치 비교 또는 대역 비교를 실시합니다.

심볼	CTBL
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C1	포트 지정	–	1
C2	제어 데이터	–	1
S	비교 테이블 하위 CH 번호	LWORD	가변

C1: 포트 지정

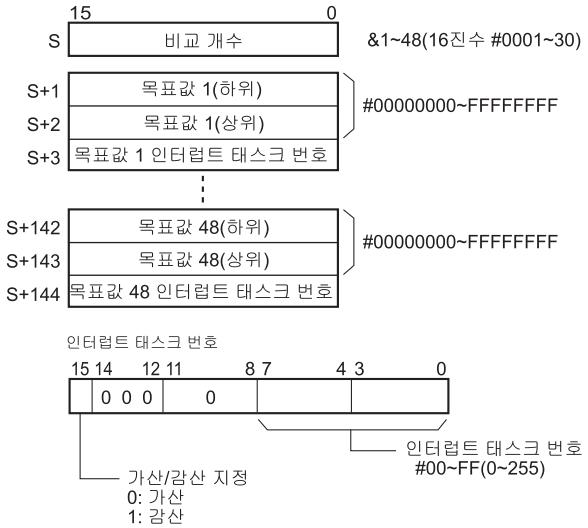
포트 지정(16진수)	
#0000	고속 카운터 입력 0
#0001	고속 카운터 입력 1

C2: 제어 데이터

제어 데이터(16진수)	
#0000	목표값 일치 비교 테이블 등록 및 비교 시작
#0001	대역 비교 테이블 등록 및 비교 시작
#0002	목표값 일치 비교 테이블 등록만
#0003	대역 비교 테이블 등록만

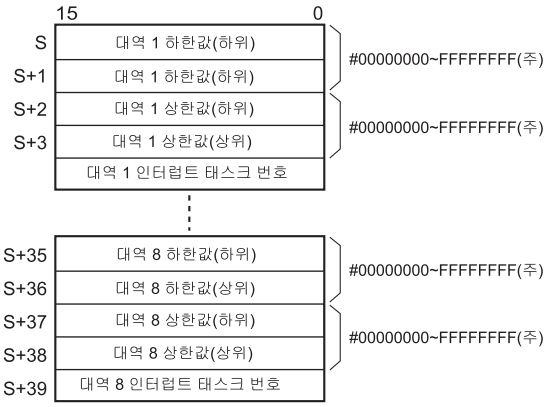
S: 비교 테이블 하위 CH 번호

・ 목표값 일치 비교 테이블 지정 시
S의 비교 개수에 따라 4~145 채널의 가변 길이가 됩니다.



CTBL

· 대역 비교 테이블 지정 시
반드시 8개의 대역을 지정하여 40채널의 고정 길이가 됩니다.
설정값이 8개 미만인 경우는 인터럽트 태스크 번호에 16진수 #FFFF를 지정합니다.



인터럽트 태스크 번호
#0000~00FF: 인터럽트 태스크 번호 0~255
#AAAA: 인터럽트 태스크를 기동 안 함
#FFFF: 이 대역의 설정을 무효로 함

주: 대역의 상한값과 하한값은 상한값≥하한값을 만족하도록 설정하십시오.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C1,C2	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
S	○	○	○	○	○	○	○		○	○	–			○				

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C1, C2로 지정할 수 있는 범위를 초과했을 때 ON · 목표값 일치 비교에서 비교 개수에 0개를 설정한 경우 ON · 목표값 일치 비교에서 비교 개수에 48개를 초과한 값을 설정한 경우 ON · 목표값 일치 비교에서 동일한 목표값을 여러 번 설정한 경우 ON · 대역 비교에서 상한값과 하한값을 역전하여 설정한 경우 ON · 대역 비교에서 모든 대역의 설정값을 무효로 한 경우 ON · 고속 카운터가 가산 펄스 모드로 설정되었는데 일치 조건으로 감산 시를 지정한 경우 ON · 고속 카운터의 링 모드로 설정되었을 때, 링 최대값을 초과하는 값을 지정하여 명령을 실행한 경우 ON · 고속 카운터로 설정되지 않은 포트에 대해 명령을 실행한 경우 ON · 비교 동작 실행 중에 다른 비교 방법으로 명령을 실행한 경우 ON · 그 이외일 때 OFF

기능

C1로 지정된 포트에 대해 C2로 지정된 방식에 따라 고속 카운터 현재값과 비교하기 위한 테이블의 등록과 비교를 시작합니다.

또한 한번 등록한 비교 테이블은 다른 테이블을 등록하거나 '프로그램' 모드로 이동할 때까지 유효합니다.

CTBL 명령은 1회 실행하면 지정된 조건으로 비교 동작을 시작합니다. 따라서 기본적으로 입력 미분형(@ 표시 있음) 또는 1사이클 ON의 입력 조건에서 사용합니다(C2에서 비교 시작을 지정한 경우).

주: 비교 테이블에 프로그램으로 등록되지 않은 인터럽트 태스크 번호가 지정된 경우, 인터럽트 발생 시에 프로그램 에러(운전 정지 이상)가 발생합니다.

■ 비교 테이블 등록(C2 = #0002, #0003)

고속 카운터 현재값과 비교하기 위한 테이블 등록만 수행합니다.

이 경우, INI 명령을 실행하면 비교가 시작됩니다.

■ 비교 테이블 등록 및 비교 시작(C2 = #0000, #0001)

고속 카운터 현재값과 비교하기 위한 테이블을 등록하고 비교를 시작합니다.

■ 비교의 정지

비교 동작을 정지하기 위해서는 CTBL 명령으로 비교를 시작한 경우나 INI 명령으로 비교를 시작한 경우 모두 INI 명령을 사용합니다.

■ 목표값 일치 비교

고속 카운터 현재값이 테이블의 목표값에 일치하면 지정된 인터럽트 태스크를 실행합니다.

- 같은 인터럽트 태스크 번호를 여러 개 비교에서 사용하는 것도 가능합니다.
- 일치 조건으로 가산 카운트 시의 일치와 감산 카운트 시의 일치를 지정할 수 있습니다.

  테이블 내 인터럽트 태스크 번호의 최상위 비트에 0을 지정하면 가산, 1을 지정하면 감산이 됩니다.
- 비교 테이블에는 최대 48개의 목표값을 등록할 수 있습니다. 또한 테이블에는 시작 채널에 지정한 개수 분량을 설정합니다(테이블의 데이터 길이: 가변 길이).
- 목표값과의 비교는 테이블에 등록된 값 전부에 대해 실행합니다.

주 1: 테이블 내에서 동일한 목표값을 여러 번 지정하면 에러가 됩니다.

주 2: 고속 카운터가 가산 펄스 모드로 설정된 경우, 일치 조건으로 감산 시를 지정하면 에러가 됩니다.

주 3: 목표값과 일치한 상태에서 카운트 방향(가산/감산)이 바뀌면 해당 방향의 다음 목표값과 일치시킬 수 없습니다. 목표값이 카운트 값의 변화에서 최저나 최대가 되지 않도록 설정하십시오.

■ 대역 비교

고속 카운터 현재값이 상한값과 하한값 사이에 들어가면 지정된 인터럽트 태스크를 실행합니다.

- 같은 인터럽트 태스크 번호를 여러 개 비교에서 사용하는 것도 가능합니다.
- 비교 테이블에는 8개의 대역(상한값과 하한값의 세트)을 등록할 수 있습니다. 대역은 중복하여 지정할 수 있습니다.
- 8개 미만인 경우에는 인터럽트 태스크 번호에 16진수 #FFFF를 지정하여 해당 대역의 설정값을 무시합니다(테이블의 데이터 길이: 고정(40CH)).
- 인터럽트 태스크의 실행은 조건 일치 기동 시에만 실행됩니다.

  1사이클 비교 실행에서 조건 일치 기동이 된 대역이 여러 개 있는 경우는 테이블의 선두에서 가까운 것을 우선적으로 실행하며 나머지는 다음 사이클 이후에 실행됩니다. 또한 인터럽트 태스크를 기동할 필요가 없는 경우는 인터럽트 태스크 번호에 16진수 #AAAA를 지정하십시오. 이 경우는 PRV 명령의 실행(대역 비교 결과 읽기) 또는 대역 비교 조건 일치 플래그를 참조하여 비교 결과를 확인합니다.

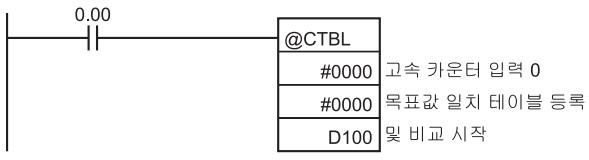
주 1: 상한값과 하한값을 역전하여 지정한 경우는 에러가 됩니다.

샘플 프로그램

0.00이 OFF→ON이면 CTBL 명령으로 고속 카운터 입력 0에 대해 목표값 일치 비교 테이블을 등록하여 비교를 시작합니다.

고속 카운터 현재값이 가산 방향으로 카운트하여 500이 된 시점에서 목표값 1과 일치시키기 위해 인터럽트 태스크 번호 1이 실행됩니다.

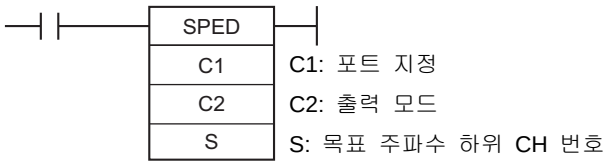
그리고 가산 카운트를 수행하여 1000이 된 시점에서 목표값 2와 일치시키기 위해 인터럽트 태스크 번호 2가 실행됩니다.



D100	&2	목표값의 개수 2
D101-D102	&500(#000001F4)	목표값 1
D103	&1	인터럽트 태스크 번호 1
D104-D105	&100(#000003E8)	목표값 2
D106	&2	인터럽트 태스크 번호 2(가산 시)

SPED

명령 이름	니모닉	동작 옵션	펑션 번호	기능
주파수 설정	SPED	@SPED	885	주파수를 지정하여 가감속 없이 펄스를 출력합니다.

심볼	SPED
	

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

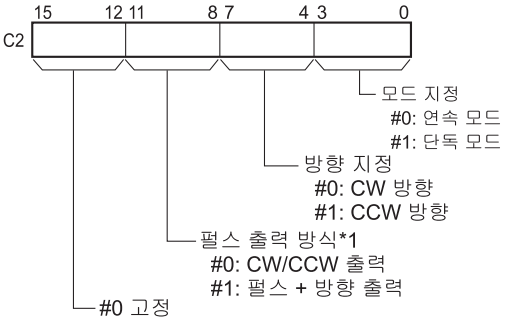
피연산자 설명

피연산자	내용	데이터 형식	크기
C1	포트 지정	UINT	1
C2	출력 모드	WORD	1
S	목표 주파수 하위 CH 번호	UDINT	2

C1: 포트 지정

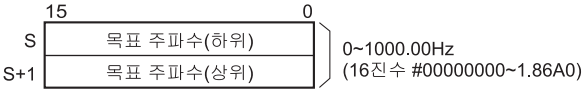
포트 지정(16진수)	
#0000	펄스 출력 0
#0001	펄스 출력 1

C2: 모드 지정



*1: 펄스 출력 0과 1을 동시에 동작시키는 경우, 동일한 방식으로 설정하십시오.

S: 목표 주파수 하위 CH 번호



출력 주파수를 1Hz 단위로 지정합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C1, C2	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
S	○	○	○	○	○	○	○		○	○	–			○				

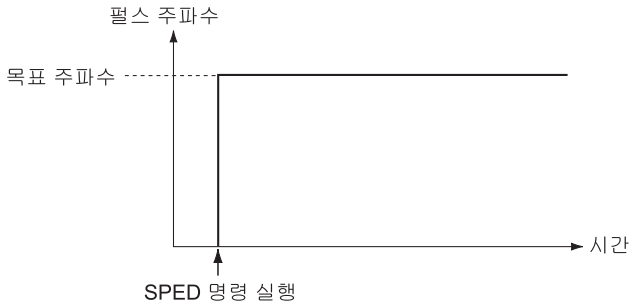
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C1, C2, S로 지정할 수 있는 범위를 초과했을 때 ON · 이미 PLS2, ORG 명령으로 펄스 출력 중인 포트에 이 명령을 사용했을 때 ON · 이미 SPED, ACC 명령으로 펄스 출력 중인 포트에 이 명령으로 단독/연속 지정을 바꿔서 사용했을 때 ON · 사이클릭 태스크로 펄스 출력을 제어하는 명령의 실행 중에 인터럽트가 발생하여 인터럽트 태스크 내에서 이 명령을 실행한 경우 ON · 원점 미확정 시, 단독 모드 및 절대 펄스 지정으로 이 명령을 실행했을 때 ON · 그 이외일 때 OFF

기능

C1로 지정한 포트에서 C2로 지정한 방식을 사용하여 S로 지정한 '목표 주파수'로 펄스 출력을 시작합니다.

SPED 명령은 1회 실행하면 지정된 조건으로 펄스 출력을 시작합니다. 따라서 기본적으로 입력 미분형 (@ 표시 있음) 또는 1사이클 ON의 입력 조건에서 사용합니다.



단독 모드에서는 미리 PULS 명령으로 설정된 펄스량을 출력하면 자동으로 펄스 출력을 정지합니다.

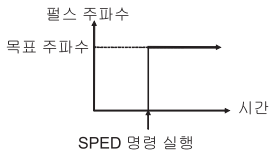
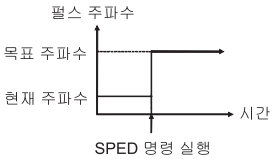
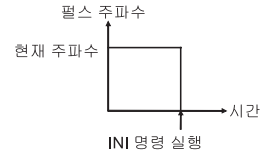
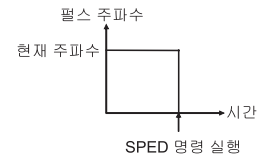
연속 모드에서는 펄스 출력 정지를 실행할 때까지 펄스 출력을 계속합니다.

단독 모드에서 펄스 출력 중에 연속 모드로 변경하거나 연속 모드에서 펄스 출력 중에 단독 모드로 변경하는 것은 에러가 되므로 실행할 수 없습니다.

■ 연속 모드(속도 제어)

펄스 출력 정지를 실행할 때까지 펄스 출력을 계속합니다.

주: '프로그램' 모드로 이동하면 펄스 출력이 즉시 정지합니다.

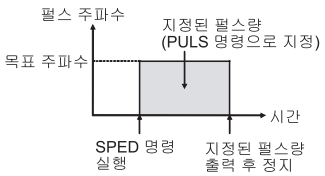
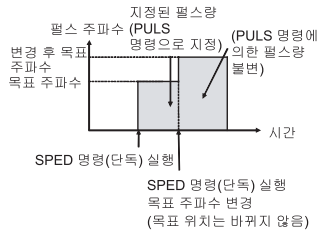
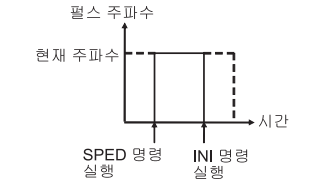
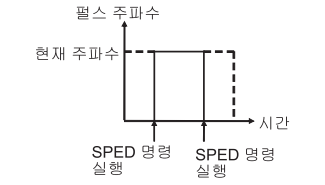
조작	동작 내용	사용 예	주파수의 변화	설명	사용 순서 명령어
펄스 출력 시작	속도 지정 출력	속도(주파수)를 스텝 형상으로 변화시킬 때		지정 주파수의 펄스를 출력합니다.	SPED(연속)
설정 변경	스텝 형상의 속도 변경	운전 중에 속도를 변경할 때		펄스 출력 중인 주파수를 스텝 형상(상방향 또는 하방향)으로 변경합니다.	SPED(연속) ↓ SPED(연속)
펄스 출력 정지	펄스 출력 정지	즉시 정지		펄스 출력을 즉시 정지합니다.	SPED(연속) ↓ INI
	펄스 출력 정지	즉시 정지		펄스 출력을 즉시 정지합니다.	SPED(연속) ↓ SPED(연속, 목표 주파수 0Hz)

■ 단독 모드(위치 결정)

PULS 명령으로 설정한 펄스 출력량만큼 펄스를 출력합니다.

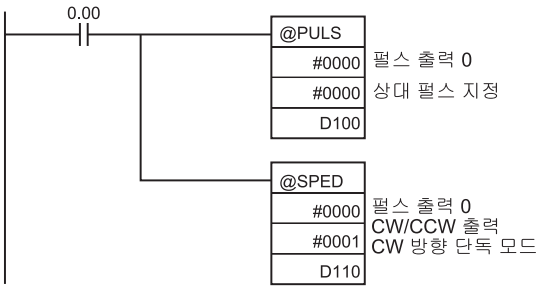
주:

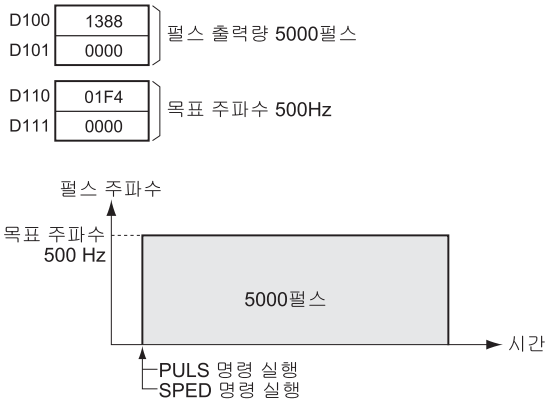
- '프로그램' 모드로 이동하면 펄스 출력이 즉시 정지합니다.
- 펄스 출력을 정지한 후 다시 출력할 때는 PULS 명령을 다시 사용하여 펄스 출력량을 설정해야 합니다.
- PULS 명령으로 미리 펄스 출력량을 설정해야 합니다. 출력량을 설정하지 않은 상태에서는 SPED 명령을 실행해도 펄스가 출력되지 않습니다.
- PULS 명령을 통해 절대 펄스 지정에서 이동 펄스량이 설정된 경우, 이 SPED 명령의 피연산자로 지정한 방향 지정(CW 방향/CCW 방향)은 무시됩니다.

조작	동작 내용	사용 예	주파수의 변화	설명	사용 순서 명령어
펄스 출력 시작	속도 지정 출력	가감속 없는 위치 결정을 할 때		펄스 출력을 지정한 주파수에서 시작하여 지정된 펄스량을 출력하면 즉시 정지시킵니다. 주: 위치 결정(펄스 출력) 중에 목표 위치(펄스량) 변경은 불가능합니다.	PULS ↓ SPED(단독)
설정 변경	스텝 형상의 속도 변경	운전 중에 스텝 형상으로 속도를 변경하려는 경우		위치 결정 중에 SPED 명령을 실행하고 펄스 출력 중인 주파수를 스텝 형상(상방향 또는 하방향)으로 변경합니다. 이 경우, 목표 위치(펄스량)는 바뀌지 않습니다.	PULS ↓ SPED(단독) ↓ SPED(단독)
펄스 출력 정지	펄스 출력 정지(펄스 출력량 비유지)	즉시 정지		펄스 출력을 즉시 정지합니다. 이 때, 현재 펄스 출력량은 클리어됩니다.	PULS ↓ SPED(단독) ↓ INI
	펄스 출력 정지(펄스 출력량 비유지)	즉시 정지		펄스 출력을 즉시 정지합니다. 이 때, 현재 펄스 출력량은 클리어됩니다.	PULS ↓ SPED(단독) ↓ SPED(단독, 목표 주파수 0Hz)

샘플 프로그램

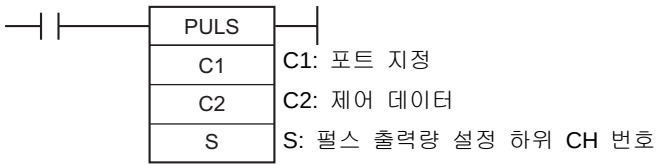
0.00이 OFF→ON이면 PULS 명령을 사용하여 펄스 출력 0의 펄스 출력량을 상대 펄스 지정으로 5000 펄스로 설정합니다. 동시에 SPED 명령으로 CW/CCW 방식, CW 방향, 단독 모드로 목표 주파수 500Hz의 펄스 출력을 시작합니다.





PULS

명령 이름	니모닉	동작 옵션	평션 번호	기능
펄스량 설정	PULS	@PULS	886	펄스 출력량을 설정합니다.

심볼	PULS
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C1	포트 지정	UINT	1
C2	제어 데이터	UINT	1
S	펄스 출력량 설정 하위 CH 번호	DINT	2

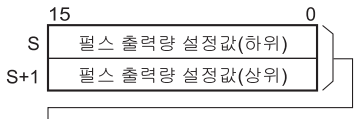
C1: 포트 지정

포트 지정(16진수)	
#0000	펄스 출력 0
#0001	펄스 출력 1

C2: 제어 데이터

제어 데이터(16진수)	
#0000	상대 펄스 지정
#0001	절대 펄스 지정

S: 펄스 출력량 설정 하위 CH 번호



상대 펄스 지정 시:
0~2147.483647(16진수 #00000000~7FFFFFFF)
절대 펄스 지정 시:
-2147.483648 ~ +2147.483647(16진수 #80000000~7FFFFFFF)

실제로 출력되는 이동 펄스량은 상대 펄스 지정 시와 절대 펄스 지정 시, 다음과 같습니다.

- 상대 펄스 지정 시
  이동 펄스량 = 펄스 출력량 설정값
- 절대 펄스 지정 시
  이동 펄스량 = 펄스 출력량 설정값 – 현재값

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C1, C2	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
S	○	○	○	○	○	○	○		○	○				○				

PULS

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C1, C2, S로 지정할 수 있는 범위를 초과했을 때 ON · 이미 펄스 출력 중인 포트에 이 명령을 사용했을 때 ON · 사이클릭 태스크로 펄스 출력을 제어하는 명령의 실행 중에 인터럽트가 발생하여 인터럽트 태스크 내에서 이 명령을 실행한 경우 ON · 그 이외일 때 OFF

기능

C1로 지정된 포트에 대해 C2, S로 지정하는 방식/펄스 출력량을 설정합니다.

PULS 명령으로 설정된 펄스 출력량은 주파수 설정(SPED) 명령 또는 주파수 가감속 제어(ACC) 명령을 단독 모드에서 실행하면 출력됩니다.

주:

· 펄스 출력 중에 이 PULS 명령을 실행한 경우 에러가 되며 펄스 출력량은 재설정할 수 없습니다.

따라서 이 명령은 기본적으로 입력 미분형(@ 표시 있음) 또는 1사이클 ON의 입력 조건에서 사용합니다.

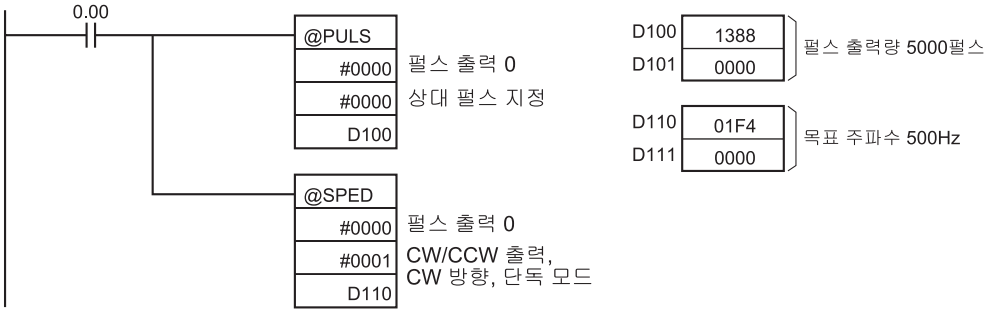
· 이 명령의 실행 후, 동작 모드 제어(INI) 명령으로 펄스 출력의 현재값을 변경해도 산출된 이동 펄스량은 변경되지 않습니다.

· 절대 펄스 지정에서 이동 펄스량을 설정한 경우, SPED 명령 및 ACC 명령의 피연산자로 지정한 방향 지정(CW 방향/CCW 방향)은 무시됩니다.

· 펄스 출력량의 현재값 범위(−2147483648~2147483647) 밖으로 이동하는 지정도 가능합니다.

샘플 프로그램

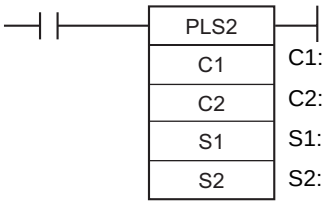
0.00이 OFF→ON이면 PULS 명령을 사용하여 펄스 출력 0의 펄스 출력량을 상대 펄스 지정으로 5000 펄스로 설정합니다. 동시에 SPED 명령으로 CW/CCW 방식, CW 방향, 단독 모드로 목표 주파수 500Hz의 펄스 출력을 시작합니다.



PLS2

명령 이름	니모닉	동작 옵션	펑션 번호	기능
위치 결정	PLS2	@PLS2	887	펄스 주파수와 가감속 비율을 지정하여 가감속이 있는 펄스 출력(가속 비율≠감속 비율)을 실시합니다.

심볼	PLS2



C1: 포트 지정
C2: 제어 데이터
S1: 설정 테이블 하위 CH 번호
S2: 기동 주파수 하위 CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

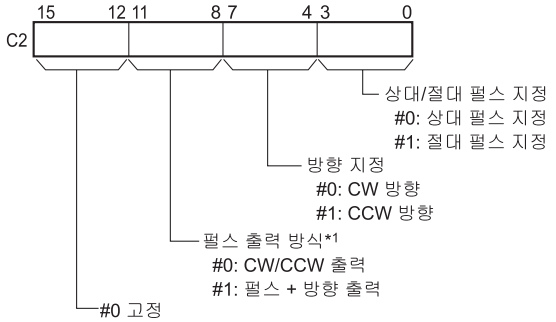
피연산자 설명

피연산자	내용	데이터 형식	크기
C1	포트 지정	–	1
C2	제어 데이터	–	1
S1	설정 테이블 하위 CH 번호	WORD	6
S2	기동 주파수 하위 CH 번호	UDINT	2

C1: 포트 지정

포트 지정(16진수)	
#0000	펄스 출력 0
#0001	펄스 출력 1

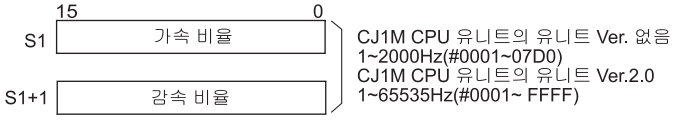
C2: 제어 데이터



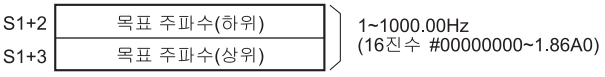
* 1: 펄스 출력 0과 1을 동시에 동작시키는 경우, 동일한 방식으로 설정하십시오.

PLS2

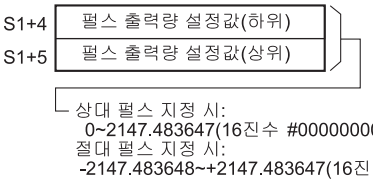
S1: 설정 테이블 하위 CH 번호



펄스 제어 주기(4ms)마다 주파수 증감량을 각각 1Hz 단위로 지정합니다.



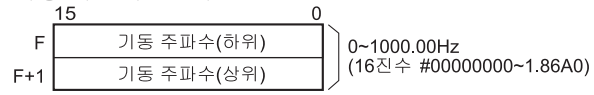
가속, 감속 후의 주파수를 1Hz 단위로 지정합니다.



실제로 출력되는 이동 펄스량은 상대 펄스 지정 시와 절대 펄스 지정 시, 다음과 같습니다.

- 상대 펄스 지정 시
  이동 펄스량 = 펄스 출력량 설정값
- 절대 펄스 지정 시
  이동 펄스량 = 펄스 출력량 설정값 – 현재값

S2: 기동 주파수 하위 CH 번호



기동 시의 주파수를 1Hz 단위로 지정합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C1, C2	–	–	–	–	–	–	–		–	–	○			–				
S1	○	○	○	○	○	○	○	–	○	○	–	–	–	○	–	–	–	–
S2											○							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C1, C2, S1, S2로 지정할 수 있는 범위를 초과했을 때 ON · 이미 SPED, ORG 명령으로 펄스 출력 중인 포트에 이 명령을 사용했을 때 ON · 사이클릭 태스크로 펄스 출력을 제어하는 명령의 실행 중에 인터럽트가 발생하여 인터럽트 태스크 내에서 이 명령을 실행한 경우 ON · 원점 미확정 시, 절대 펄스 지정으로 이 명령을 실행했을 때 ON · 그 이외일 때 OFF

기능

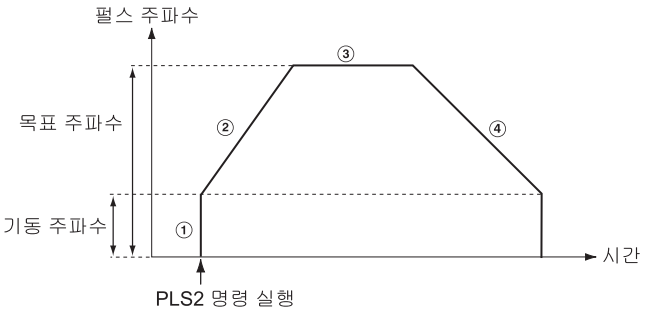
C1로 지정한 포트에서 C2로 지정한 방식을 사용하여 S2로 지정한 기동 주파수로 펄스 출력을 시작합니다(오른쪽 그림 ①).

펄스 제어 주기(4ms)마다, S1로 지정한 가속 비율에 따라 S1로 지정한 목표 주파수에 도달할 때까지 주파수를 증가시킵니다(오른쪽 그림 ②).

목표 주파수에 도달하면 가속을 정지하고 등속으로 펄스 출력을 계속합니다(오른쪽 그림 ③).

S1로 지정된 펄스 출력량과 감속 비율에서 계산한 감속 포인트(주파수를 감소시키는 타이밍)가 되면 펄스 제어 주기(4ms)마다 주파수를 감소시켜 기동 주파수에 도달하면 출력을 정지합니다(오른쪽 그림 ④).

PLS2 명령은 1회 실행하면 지정된 조건으로 펄스 출력을 시작합니다. 따라서 기본적으로 입력 미분형(@ 표시 있음) 또는 1사이클 ON의 입력 조건에서 사용합니다.



PLS2 명령은 위치 결정만 가능합니다.

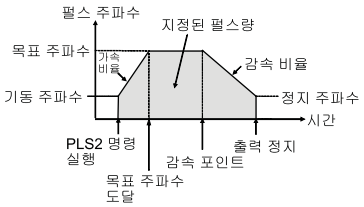
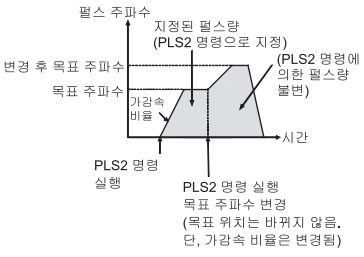
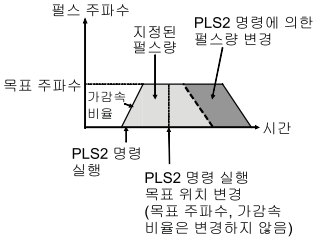
CJ1M CPU 유니트의 경우, ACC 명령(단독 또는 연속 모드)에 의한 펄스 출력 중(등속뿐만 아니라 가속 또는 감속 중에도 가능)에 이 PLS2 명령을 실행할 수 있습니다(*1)(*2). 또한 반대로 이 PLS2 명령에 의한 펄스 출력 중(등속뿐만 아니라 가속 또는 감속 중에도 가능)에 ACC 명령(단독 모드)을 실행할 수도 있습니다.

*1: ACC 명령(연속 모드)으로 속도 제어 중에 ACC 명령과 같은 목표 주파수를 지정하여 PLS2 명령을 실행하면 인터럽트 일정 크기 이송도 가능합니다.
이때 PLS2 명령에 의한 가속은 수행되지 않지만 가속 비율을 0으로 하면 에러 플래그가 ON되어 PLS2 명령이 실행되지 않습니다. 가속 비율에는 0 이외의 값을 설정하십시오.

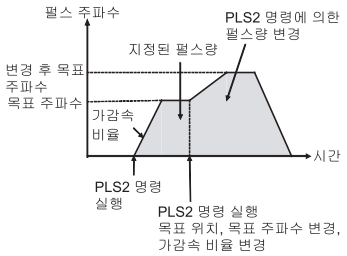
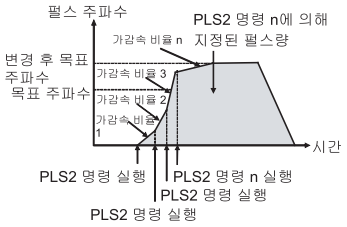
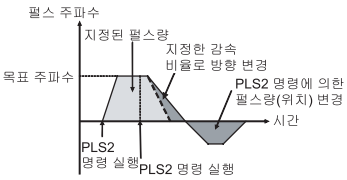
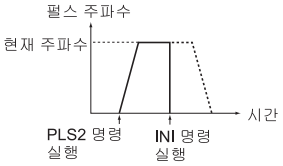
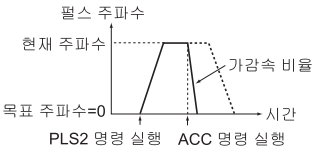
*2: ACC 명령(단독 또는 연속 모드)을 목표 주파수 0Hz로 실행하고, 펄스 출력의 정지 직전에 이 PLS2 명령을 실행해도 목표 주파수는 변경되지 않고 펄스 출력은 그대로 정지합니다. PLS2 명령은 펄스 출력의 정지 직전에는 실행되지 않으므로 펄스 출력 정지 후에 실행하십시오.

■ 단독 모드(위치 결정)

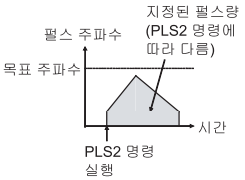
주: '프로그램' 모드로 이동하면 펄스 출력이 즉시 정지합니다.

조작	동작 내용	사용 예	주파수의 변화	설명	사용 순서 명령어
펄스 출력 시작	자세한 사다리꼴 제어	사다리꼴 가감속 위치를 결정할 때(가속 비율과 감속 비율은 개별. 기동 속도 있음). 위치 결정 중 펄스량 변경 가능		일정 비율로 주파수를 가속 또는 감속합니다. 지정된 펄스량을 출력하면 정지됩니다(*1). 주: 위치 결정(펄스 출력) 중에 목표 위치(펄스량)의 변경이 가능합니다.	PLS2
설정 변경	기울기 있는 속도 변경(가속 비율≠감속 비율) 변경	위치 결정 중 목표 속도(주파수)의 변경(가속 비율과 감속 비율이 개별)		위치 결정 중에 PLS2 명령을 실행하고 가속 비율, 감속 비율, 목표 주파수를 변경할 수 있습니다. 주: 의도적으로 목표 위치를 바꿀 수 없도록 PLS2 명령의 피연산자로 절대 지정을 통해 원래와 같은 위치를 지정해야 합니다.	PLS2 ↓ PLS2 PULS ↓ ACC(단독) ↓ PLS2
	목표 위치 변경	위치 결정 중 목표 위치의 변경(다중 기동)		위치 결정 중에 PLS2 명령을 실행하고 목표 위치(펄스량), 가속 비율, 감속 비율, 목표 주파수를 변경할 수 있습니다. 주: 변경 후에 등속역을 확보할 수 없는 경우, 에러가 되어 실행을 무시하고 원래 동작을 계속합니다.	PLS2 ↓ PLS2 PULS ↓ ACC(단독) ↓ PLS2

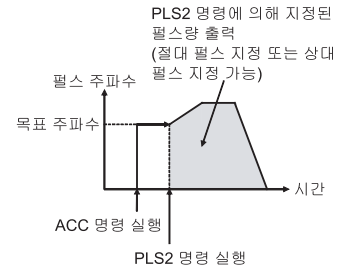
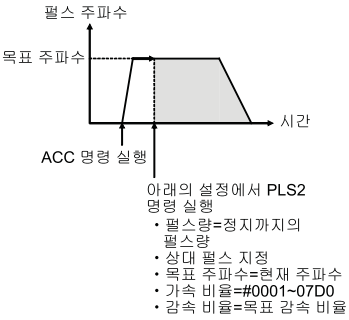
PLS2

조작	동작 내용	사용 예	주파수의 변화	설명	사용 순서 명령어
	목표 위치+기울기 있는 속도 변경	위치 결정 중 목표 위치, 목표 속도(주파수)의 변경(다중 기동)		위치 결정 중에 PLS2 명령을 실행하고 목표 위치(펄스량), 가속 비율, 감속 비율, 목표 주파수를 변경할 수 있습니다. 주: 변경 후에 등속역을 확보할 수 없는 경우, 에러가 되어 실행을 무시하고 원래 동작을 계속합니다.	PULS ↓ ACC(단독) ↓ PLS2 --- PLS2 ↓ PLS2
		위치 결정 중 가감속의 변경(다중 기동)		위치 결정 중에 PLS2 명령을 실행하고 가속 비율, 감속 비율을 변경할 수 있습니다.	PULS ↓ ACC(단독) ↓ PLS2 --- PLS2 ↓ PLS2
	방향 변경	위치 결정 중 방향의 변경		절대 펄스 지정에 따른 위치 결정 중에 PLS2 명령을 실행하고 절대 위치를 지정하여 방향을 변경할 수 있습니다.	PLS2 ↓ PLS2 --- PULS ↓ ACC(단독) ↓ PLS2
펄스 출력 정지	펄스 출력 정지(펄스 출력량 비유지)	즉시 정지		펄스 출력을 즉시 정지합니다. 이 때, 현재 펄스 출력량은 클리어됩니다.	PLS2 ↓ INI
	기울기 있는 펄스 출력 정지(펄스 출력량 비유지)	감속 정지		펄스 출력을 감속 정지합니다.	PLS2 ↓ ACC(단독, 목표 주파수 0Hz)

*1: 삼각 제어
가속 및 감속에만 필요한 펄스량이 목표 주파수에 도달하는 펄스량을 충족하지 않는 경우에도(에러가 되지 않음), 자동으로 가감속 시간을 단축하여 가감속만의 삼각 형상의 제어를 실행합니다.

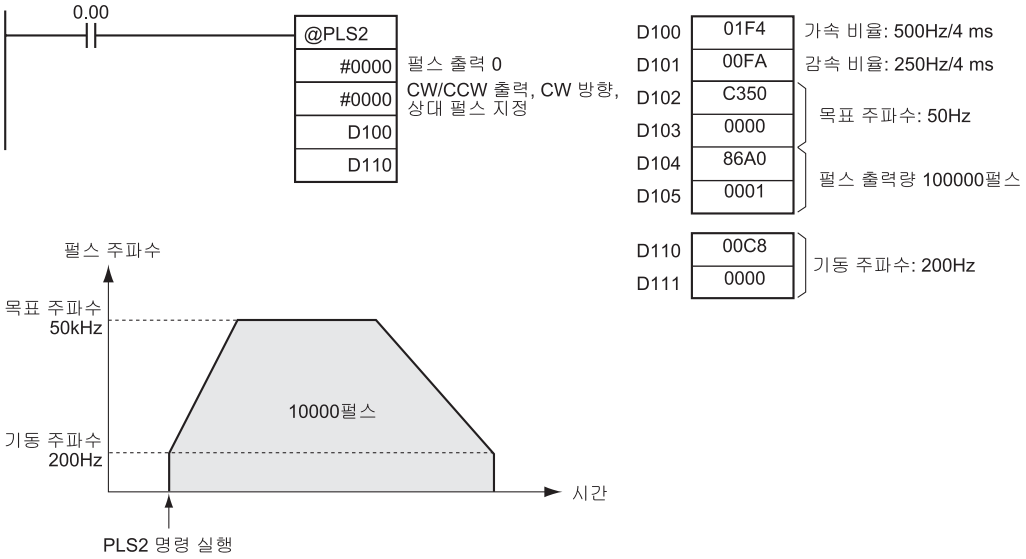


■ 연속 모드(속도 제어)→단독 모드(위치 결정)

사용 예	주파수의 변화	설명	사용 순서 명령어
속도 제어 중 일정량 이송 위치 결정으로 변경		ACC 명령으로 속도 제어 중에 PLS2 명령을 실행하고 위치 결정으로 변경합니다.	ACC(연속) ↓ PLS2
인터럽트 일정 크기 이송			

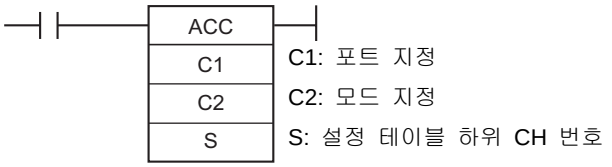
샘플 프로그램

0.00이 OFF→ON이면 PLS2 명령을 사용하여 펄스 출력 0에서 상대 펄스 지정으로 100000펄스의 출력을 시작합니다. 기동 주파수인 200Hz부터 가속 비율 500Hz/4ms로 목표 주파수인 50kHz까지 가속합니다. 그 후, 감속 포인트부터 감속 비율 250Hz/4ms로 감속하고 기동 주파수인 200Hz까지 감속되면 펄스 출력을 정지합니다.



ACC

명령 이름	니모닉	동작 옵션	펑션 번호	기능
주파수 가감속 제어	ACC	@ACC	888	펄스 주파수와 가감속 비율을 지정하여 가감속이 있는 펄스 출력(가속 비율=감속 비율)을 실시합니다.

	ACC
심볼	

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

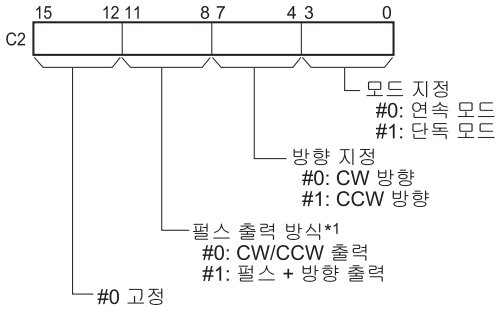
피연산자 설명

피연산자	내용	데이터 형식	크기
C1	포트 지정	–	1
C2	모드 지정	–	1
S	설정 테이블 하위 CH 번호	WORD	3

C1: 포트 지정

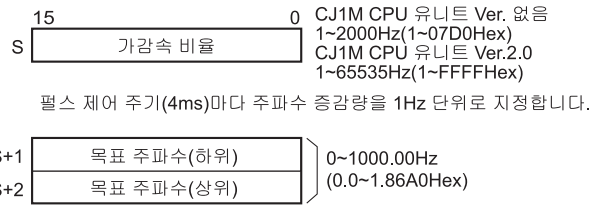
포트 지정(16진수)	
#0000	펄스 출력 0
#0001	펄스 출력 1

C2: 모드 지정



* 1: 펄스 출력 0과 1을 동시에 동작시키는 경우, 동일한 방식으로 설정하십시오.

S: 설정 테이블 하위 CH 번호



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C1, C2	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
S	○	○	○	○	○	○	○		○	○	–			○				

조건 플래그의 동작

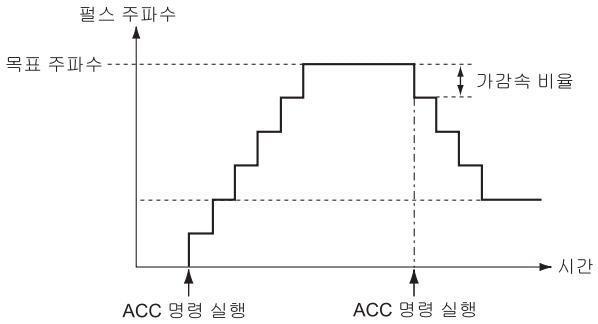
이름	라벨	내용
에러 플래그	P_ER	· C1, C2, S로 지정할 수 있는 범위를 초과했을 때 ON · 이미 ORG 명령으로 펄스 출력 중인 포트에 이 명령을 사용했을 때 ON · 이미 SPED, ACC, PLS2 명령으로 출력 중인 포트에 이 명령으로 단독/연속 지정을 바꿔서 사용했을 때 ON · 사이클릭 태스크로 펄스 출력을 제어하는 명령의 실행 중에 인터럽트가 발생하여 인터럽트 태스크 내에서 이 명령을 실행한 경우 ON · 원점 미확정 시, 단독 모드 및 절대 펄스 지정으로 이 명령을 실행했을 때 ON · 그 이외일 때 OFF

기능

C1로 지정한 포트에서 C2로 지정한 방식을 사용하여 S로 지정한 '목표 주파수', '가감속 비율'로 펄스 출력을 시작합니다.

또한 펄스 제어 주기(4ms)마다, S로 지정된 가감속 비율에 따라 S+2, S+1로 지정된 목표 주파수에 도달할 때까지 주파수를 가감속합니다.

ACC 명령은 1회 실행하면 지정된 조건으로 펄스 출력을 시작합니다. 따라서 기본적으로 입력 미분형(@ 표시 있음) 또는 1사이클 ON의 입력 조건에서 사용합니다.



단독 모드에서는 미리 PULS 명령으로 설정된 펄스량을 출력하면 자동으로 펄스 출력을 정지합니다.

연속 모드에서는 펄스 출력 정지를 실행할 때까지 펄스 출력을 계속합니다.

단독 모드에서 펄스 출력 중에 연속 모드로 변경하거나 연속 모드에서 펄스 출력 중에 단독 모드로 변경하는 것은 에러가 되므로 실행할 수 없습니다.

CJ1M CPU 유니트의 경우, 이 ACC 명령(단독 또는 연속 모드)에 의한 펄스 출력 중(등속뿐만 아니라 가속 또는 감속 중에도 가능)에 이 PLS2 명령을 실행할 수 있습니다(*1). 또한 반대로 이 PLS2 명령에 의한 펄스 출력 중(등속뿐만 아니라 가속 또는 감속 중에도 가능)에 PLS2 명령 또는 ACC 명령(단독 모드)을 실행할 수도 있습니다.

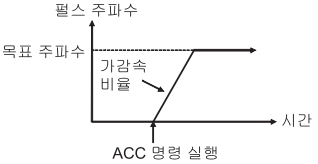
이 ACC 명령(단독 또는 연속 모드)을 목표 주파수 0Hz로 실행하고, 펄스 출력의 정지 직전에 ACC 명령이나 PLS2 명령을 실행해도 목표 주파수는 변경되지 않고 펄스 출력은 그대로 정지합니다. ACC 명령이나 PLS2 명령은 펄스 출력의 정지 직전에는 실행되지 않으므로 펄스 출력 정지 후에 실행하십시오.

*1: ACC 명령(연속 모드)으로 속도 제어 중에 ACC 명령과 같은 목표 주파수를 지정하여 PLS2 명령을 실행하면 인터럽트 일정 크기 이송도 가능합니다.
이때 PLS2 명령에 의한 가속은 수행되지 않지만 가속 비율을 0으로 하면 에러 플래그가 ON되어 PLS2 명령이 실행되지 않습니다. 가속 비율에는 0 이외의 값을 설정하십시오.

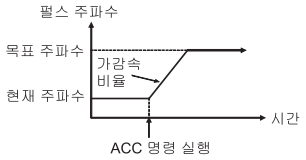
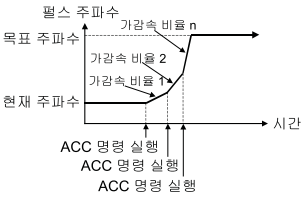
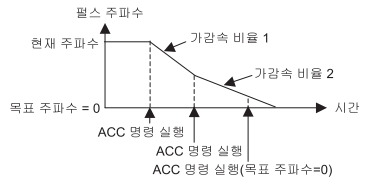
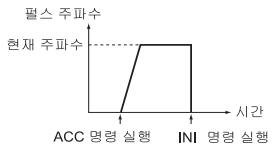
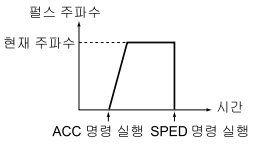
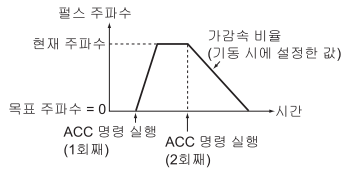
■ 연속 모드(속도 제어)

펄스 출력 정지를 실행할 때까지 펄스 출력을 계속합니다.

주: '프로그램' 모드로 이동하면 펄스 출력이 즉시 정지합니다.

조작	동작 내용	사용 예	주파수의 변화	설명	사용 순서 명령어
펄스 출력 시작	가속도/속도 지정 출력	속도(주파수)를 일정 비율로 가속시킬 때		일정 비율로 주파수를 변경시킨 펄스를 출력합니다.	ACC(연속)

ACC

조작	동작 내용	사용 예	주파수의 변화	설명	사용 순서 명령어
설정 변경	기울기 있는 속도 변경	운전 중에 속도를 완만하게 변경하고 싶을 때		현재의 주파수에서 일정 비율로 주파수를 가속 또는 감속시킵니다.	ACC(연속) 또는 SPED(연속) ↓ ACC(연속)
		기울기의 연속으로 속도를 변화시키고 싶을 때		가속 또는 감속 중에 가속 비율 또는 감속 비율을 변경시킵니다.	ACC(연속) ↓ ACC(연속)
		속도를 변화시키면서 정지시키고 싶을 때		감속 중에 감속 비율을 변경시킵니다. 주: 목표 주파수를 0Hz로 설정한 경우, 감속 비율은 동작 중의 값이 그대로 유효하게 됩니다.	ACC(연속) ↓ ACC(연속) ↓ ACC(연속, 목표 주파수 0Hz)
펄스 출력 정지	펄스 출력 정지	즉시 정지		펄스 출력을 즉시 정지합니다.	ACC(연속) ↓ INI
	펄스 출력 정지	즉시 정지		펄스 출력을 즉시 정지합니다.	ACC(연속) ↓ SPED(연속, 목표 주파수 0Hz)
	기울기 있는 펄스 출력 정지	감속 정지		펄스 출력을 감속 정지합니다. 주: 2회째 ACC 명령의 목표 주파수가 0Hz일 때는 1회째 ACC 명령의 감속 비율로 감속합니다.	ACC(연속) ↓ ACC(연속, 목표 주파수 0Hz)

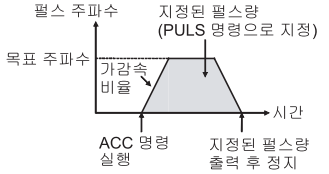
■ 단독 모드(위치 결정)

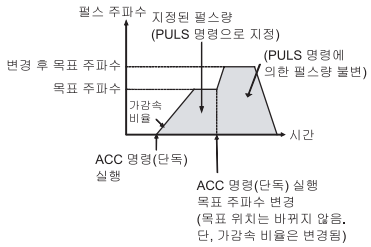
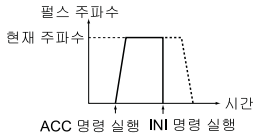
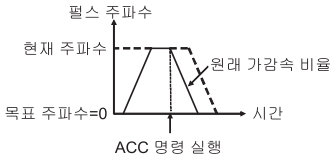
PULS 명령으로 설정한 펄스 출력량만큼 펄스를 출력합니다.

설정된 펄스 출력량과 가감속 비율에서 계산한 감속 포인트(주파수를 감소시키는 타이밍)가 되면 4ms마다 가감속 비율로 설정된 주파수만큼 주파수를 감소시켜, 펄스 출력량에 도달한 시점에서 펄스 출력량을 정지합니다.

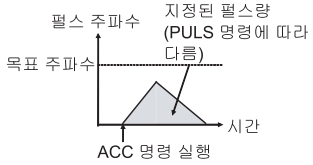
주:

- '프로그램' 모드로 이동하면 펄스 출력이 즉시 정지합니다.
- 펄스 출력을 정지한 후 다시 출력할 때는 PULS 명령을 다시 사용하여 펄스 출력량을 설정해야 합니다.
- PULS 명령으로 미리 펄스 출력량을 설정해야 합니다. 출력량을 설정하지 않은 상태에서는 ACC 명령을 실행해도 펄스가 출력되지 않습니다.
- PULS 명령으로 절대 펄스 지정에서 이동 펄스량이 설정된 경우, 이 ACC 명령의 피연산자로 지정한 방향 지정(CW 방향/CCW 방향)은 무시됩니다.

조작	동작 내용	사용 예	주파수의 변화	설명	사용 순서 명령어
펄스 출력 시작	단순한 사다리꼴 제어	사다리꼴 가감속의 위치를 결정할 때 (가속 비율과 감속 비율이 일치. 기동 속도 없음). 위치 결정 중 펄스량 변경 불가능		일정 비율로 주파수를 가속 또는 감속하고 지정된 펄스량을 출력하면 즉시 정지시킵니다(*1). 주: 위치 결정(펄스 출력) 중에 목표 위치(펄스량) 변경은 불가능합니다.	PULS ↓ ACC(단독)

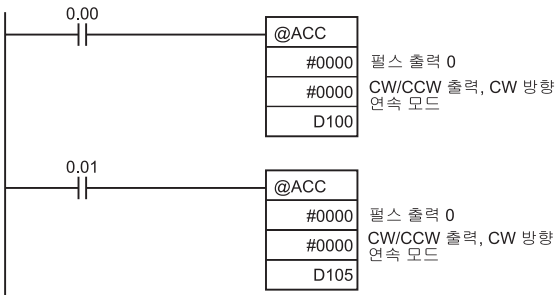
조작	동작 내용	사용 예	주파수의 변화	설명	사용 순서 명령어
설정 변경	기울기 있는 속도 변경(가속 비율 = 감속 비율) 변경	위치 결정 중 목표 속도(주파수)의 변경(가속 비율과 감속 비율이 동일)		위치 결정 중에 ACC 명령을 실행하여 가감속 비율, 목표 주파수를 변경할 수 있습니다. 이 경우, 목표 위치(펄스량)는 바뀌지 않습니다.	PULS ↓ ACC(단독) 또는 SPED(단독) ↓ ACC(단독)
					PLS2 ↓ ACC(단독)
펄스 출력 정지	펄스 출력 정지(펄스 출력량 비유지)	즉시 정지		펄스 출력을 감속 정지합니다. 주: ACC 명령 실행 시의 가감속 비율은 동작 중의 값이 그대로 유효하게 됩니다. 따라서 SPED 명령으로 기동된 경우는 가감속 비율이 무효이므로 즉시 정지됩니다.	PULS ↓ ACC(단독) ↓ INI
	기울기 있는 펄스 출력 정지(펄스 출력량 비유지)	감속 정지			PULS ↓ ACC(단독) 또는 SPED(단독) ↓ ACC(단독, 목표 주파수 0Hz)
					PLS2 ↓ ACC(단독, 목표 주파수 0Hz)

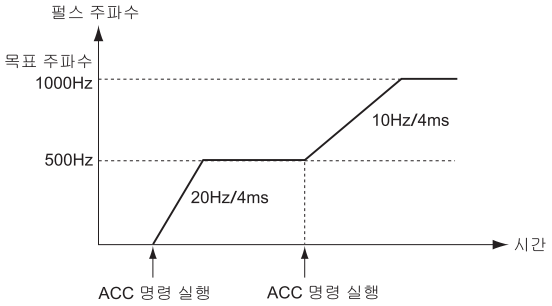
*1: 삼각 제어
가속 및 감속에만 필요한 펄스량이 목표 주파수에 도달하는 펄스량을 충족하지 않는 경우에도(에러가 되지 않음), 자동으로 가감속 시간을 단축하여 가감속만의 삼각 형상의 제어를 실행합니다.



샘플 프로그램

0.00이 OFF→ON이면 ACC 명령을 사용하여 펄스 출력 0의 포트에서 CW/CCW 방식, CW방향, 연속 모드로 가감속 비율 20Hz, 목표 주파수 500Hz의 펄스 출력을 시작합니다. 그 후, 0.01이 OFF→ON이 되면 다시 ACC 명령을 사용하여 가감속 비율 10Hz, 목표 주파수 1000Hz로 변경합니다.



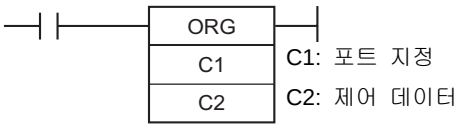


D100	0014	가속 비율 20Hz
D101	01F4	목표 주파수 500Hz
D102	0000	

D105	000A	가속 비율 10Hz
D106	03E8	목표 주파수 1000Hz
D107	0000	

ORG

명령 이름	니모닉	동작 옵션	평션 번호	기능
원점 검색	ORG	@ORG	889	원점 검색 또는 원점 복귀를 실행합니다.

심볼	ORG
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

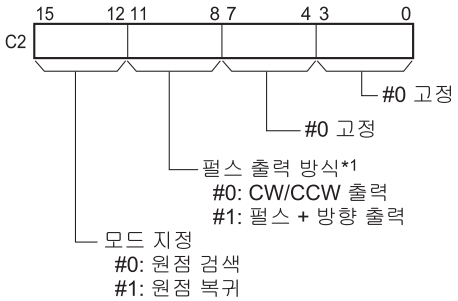
피연산자 설명

피연산자	내용	데이터 형식	크기
C1	포트 지정	–	1
C2	제어 데이터	–	1

C1: 포트 지정

포트 지정(16진수)	
#0000	펄스 출력 0
#0001	펄스 출력 1

C2: 제어 데이터



* 1: 펄스 출력 0과 1을 동시에 동작시키는 경우, 동일한 방식으로 설정하십시오.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C1, C2	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C1, C2로 지정할 수 있는 범위를 초과했을 때 ON · 이미 SPED, ACC, PLS2 명령으로 펄스 출력 중인 포트에 이 명령을 사용했을 때 ON · 사이클릭 태스크로 펄스 출력을 제어하는 명령의 실행 중에 인터럽트가 발생하여 인터럽트 태스크 내에서 이 명령을 실행한 경우 ON · PLC 시스템 설정의 원점 결정 관련 파라미터가 범위 밖인 경우 ON · PLC 시스템 설정의 원점 검색 관련 속도가 원점 검색 고속 속도≦원점 검색 근방 속도가 되거나 원점 검색 근방 속도≦원점 검색 기동 속도가 된 경우 ON · PLC 시스템 설정의 원점 복귀 관련 속도가 원점 복귀 목표 속도≦원점 복귀 기동 속도인 경우 ON · 원점 미확정 상태에서 원점 복귀를 지정한 경우 ON, 그 이외일 때 OFF

기능

C1로 지정된 포트에서 C2로 지정된 방식을 사용하여 펄스를 출력하고 원점 검색 또는 원점 복귀를 수행합니다.

이 ORG 명령을 실행하는 경우 PLC 시스템 설정을 통해 다음 파라미터를 미리 설정해야 합니다.

원점 검색	원점 복귀
· 원점 검색 기능 사용 설정	· 원점 복귀 목표 속도
· 동작 모드 선택	· 원점 검색/원점 복귀 기동 속도
· 원점 검색 동작 지정	· 원점 복귀 가속 비율
· 원점 검출 방법 지정	· 원점 복귀 감속 비율
· 원점 검색 방향 지정	
· 원점 검색/원점 복귀 기동 속도	
· 원점 검색 고속 속도	
· 원점 검색 근방 속도	
· 원점 검색 가속 비율	
· 원점 검색 감속 비율	
· 원점 보정 데이터	
· 원점 근방 입력 신호 종류	
· 원점 입력 신호 종류	
· 한계 입력 신호 종류	
· 위치 결정 감시 시간	

ORG 명령은 1회 실행하면 지정된 조건으로 원점 결정 동작을 시작합니다. 따라서 기본적으로 입력 미분형(@ 표시 있음) 또는 1사이클 ON의 입력 조건에서 사용합니다.

■ 원점 검색(C2 비트 12~15 = 0)

지정된 출력 방식으로 원점 검색 기동 속도의 펄스 출력을 시작합니다(오른쪽 그림 ①).

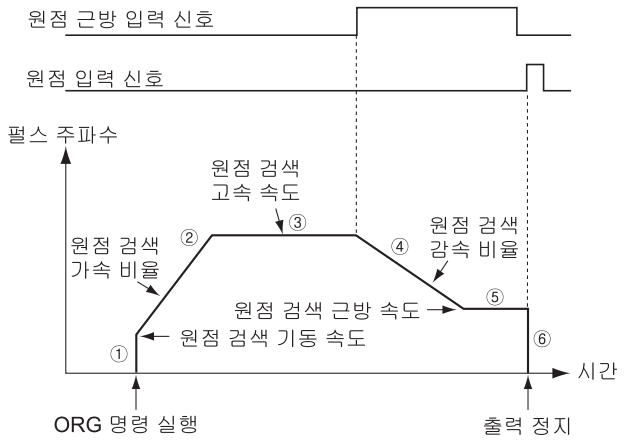
원점 검색 가속 비율에 따라 원점 검색 고속 속도까지 가속하고(오른쪽 그림 ②), 원점 근방 입력 신호가 ON될 때까지 등속으로 동작합니다(오른쪽 그림 ③).

원점 근방 입력 신호가 ON되면 원점 검색 감속 비율에 따라 원점 검색 근방 속도까지 감속하고(오른쪽 그림 ④), 원점 입력 신호가 ON될 때까지 등속으로 동작합니다(오른쪽 그림 ⑤).

원점 입력 신호가 ON되면 펄스 출력을 정지합니다(오른쪽 그림 ⑥).

원점 검색이 완료되면 편차 카운터 리셋 출력이 ON됩니다.

위와 같은 동작은 동작 모드 또는 원점 검출 방법 등의 파라미터에 따라 달라집니다.



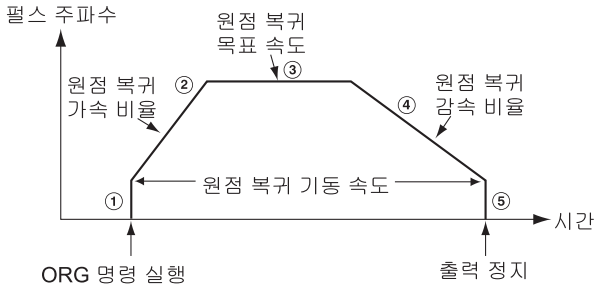
ORG

■ 원점 복귀(C2 비트 12~15 = 1)

지정된 출력 방식으로 원점 복귀 기동 속도의 펄스 출력을 시작합니다(오른쪽 그림 ①).

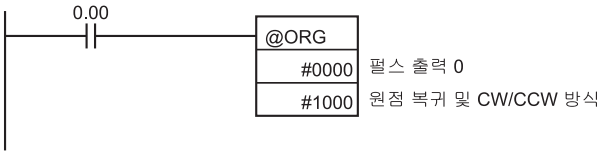
원점 복귀 가속 비율에 따라 원점 복귀 목표 속 속도까지 가속하여(오른쪽 그림 ②) 등속으로 동작합니다(오른쪽 그림 ③).

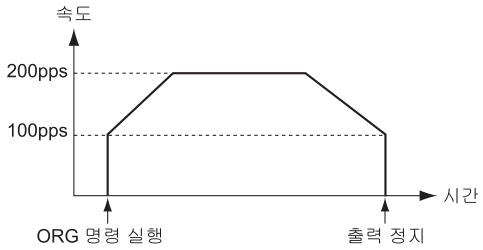
원점 위치까지의 펄스 출력량과 원점 복귀 감속 비율에서 계산한 감속 포인트(주파수를 감소시키는 타이밍)가 되면 감속을 시작하여(오른쪽 그림 ④) 원점 복귀 기동 속도까지 감속되면(원점에 도달하면) 펄스 출력을 정지합니다(오른쪽 그림 ⑤).



샘플 프로그램

0.00이 OFF→ON이면 ORG 명령을 사용하여 펄스 출력 0에 대해 CW/CCW 방식으로 펄스를 출력하고 원점 복귀를 수행합니다. PLC 시스템 설정에 따라 기동 속도는 100pps, 목표 속도는 200pps, 가속 비율과 감속 비율은 50Hz/4ms가 됩니다.



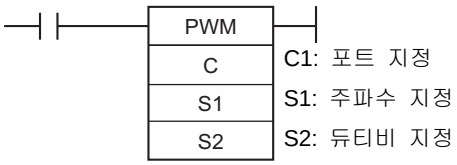


PLC 시스템 설정

펄스 출력 0 원점 검색/원점 복귀 기동 속도	100pps
펄스 출력 0 원점 복귀 목표 속도	200pps
펄스 출력 0 원점 복귀 가속 비율	50Hz/4ms
펄스 출력 0 원점 복귀 감속 비율	50Hz/4ms

PWM

명령 이름	니모닉	동작 옵션	평션 번호	기능
PWM 출력	PWM	@PWM	891	듀티비 가변 펄스를 출력합니다.

심볼	PWM
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C	포트 지정	–	1
S1	주파수 지정	–	1
S2	듀티비 지정	–	1

C: 포트 지정

포트 지정(16진수)	
#0000	펄스 출력 0(듀티비 1% 단위)
#0001	펄스 출력 1(듀티비 1% 단위)

· CJ1M CPU 유니트의 유니트 Ver.2.0 이상만 해당

포트 지정(16진수)	
#1000	펄스 출력 0(듀티비 0.1% 단위)
#1001	펄스 출력 1(듀티비 0.1% 단위)

주: CJ1M-CPU21에서는 펄스 출력 0만 사용 가능

S1: 주파수 지정

10진수 &1~65535 또는 16진수 #0001~FFFF: 0.1Hz~6553.5Hz(0.1Hz 단위로 지정 가능)

단, 출력 회로의 제한에 의해 실제로 출력되는 PWM 파형의 정도(ON 듀티+5%, -0%)가 보장되는 것은 0.1~1000.0Hz입니다.

S2: 듀티비 지정

CJ1M CPU 유니트의 유니트 Ver.2.0

10진수 &0~1000 또는 16진수 #0000~03E8: 0.0~100.0%(0.1% 단위로 지정 가능)

CJ1M CPU 유니트의 유니트 버전 표기가 없는 타입

10진수 &0~100 또는 16진수 #0000~0064: 0~100%(1% 단위로 지정 가능)

듀티비(펄스 주기에 대한 ON 시간의 비율)를 백분율로 지정합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
S1, S2	○	○	○	○	○	○	○		○	○		○		○				

PWM

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C, S1, S2로 지정할 수 있는 범위를 초과했을 때 ON · 이미 ORG 명령으로 펄스 출력 중인 포트에 이 명령을 사용했을 때 ON · 사이클릭 태스크로 펄스 출력을 제어하는 명령의 실행 중에 인터럽트가 발생하여 인터럽트 태스크 내에서 이 명령을 실행한 경우 ON · 그 이외일 때 OFF

기능

C로 지정된 포트에서 S1로 지정된 주파수, S2로 지정된 듀티비의 펄스를 출력합니다.

PWM 명령에 의한 가변 듀티비 펄스 출력 중에 듀티비 지정을 변경하여 PWM 명령을 실행하면 펄스 출력을 정지하지 않고 듀티비를 변경할 수 있습니다. 단, 주파수 변경은 무시되어 실행할 수 없습니다.
또한 PWM 명령은 1회 실행하면 지정된 조건으로 펄스 출력을 시작합니다. 따라서 기본적으로 입력 미분형 (@ 표시 있음) 또는 1사이클 ON의 입력 조건에서 사용합니다.

펄스 출력의 정지는 다음 중 한 방법으로 실행합니다.

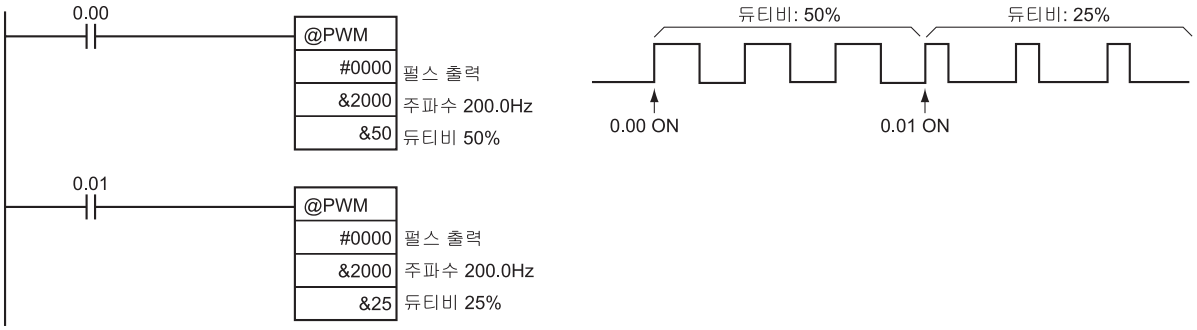
(1) INI 명령(C2=#0003: 펄스 출력 정지)을 실행

(2) '프로그램' 모드로 이동

샘플 프로그램

0.00이 OFF→ON이면 PWM 명령을 사용하여 펄스 출력 0에 대해 주파수 200Hz, 듀티비 50%로 펄스 출력을 시작합니다.

0.01이 OFF→ON이 되면 듀티비 25%로 변경합니다.

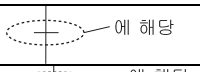
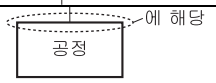


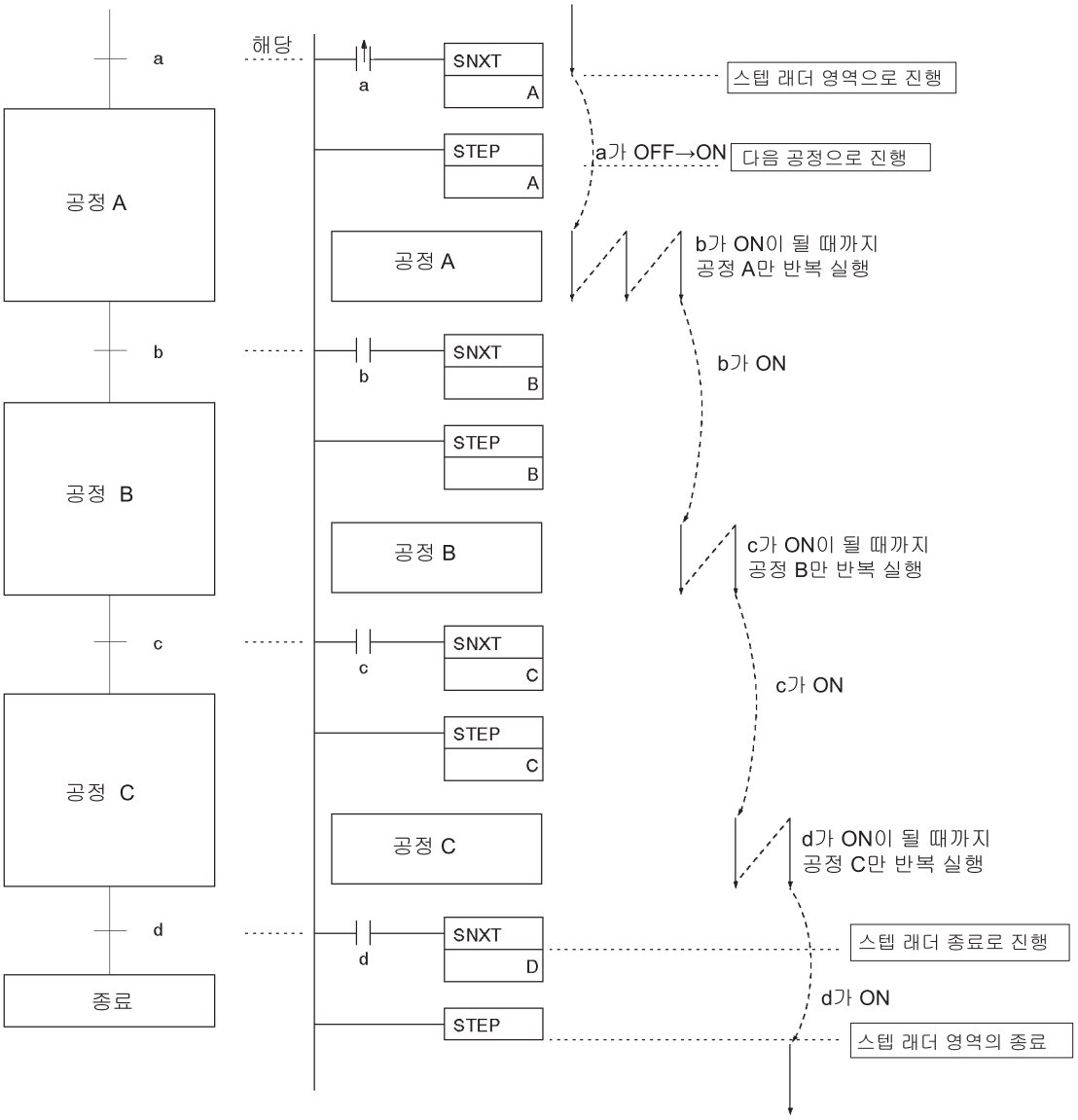
공정 진행 제어 명령

니모닉	명령 이름	펑션 번호	페이지
SNXT	스텝 래더 영역 진행	009	3-639
STEP	스텝 래더 영역 정의	008	

공정 진행 제어 명령이란?

CS/CJ 시리즈에서는 SNXT(스텝 래더 진행) 명령과 STEP(스텝 래더 영역) 명령을 조합하여 공정 진행 프로그램을 작성할 수 있습니다.

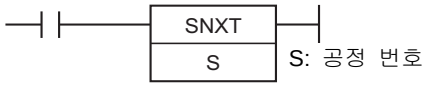
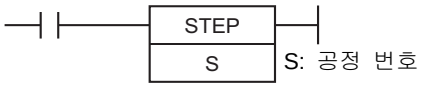
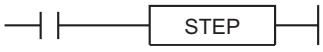
명령	동작	공정도
SNXT(스텝 래더 진행) 명령	다음 공정으로 진행을 제어합니다.	
STEP(스텝 래더 영역) 명령	공정의 시작을 나타냅니다. 다음 공정으로 진행하기 위한 조건이 성립할 때까지 해당 공정의 프로그램만 반복 실행합니다.	



주: A, B, C, D의 공정 진행 번호는 내부 보조 릴레이 WR을 사용합니다.

SNXT/STEP

명령 이름	니모닉	동작 옵션	펑션 번호	기능
스텝 래더 영역 진행	SNXT	–	009	스텝 래더 영역의 공정 진행을 제어합니다.
스텝 래더 영역 정의	STEP	–	008	스텝 래더 영역에서 공정의 시작과 종료를 정의합니다.

	SNXT	STEP
심볼		스텝 래더 영역 시작 명령 (공정 번호를 지정한 경우)  스텝 래더 영역 종료 명령 (공정 번호를 지정하지 않은 경우) 

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	불가능	가능	불가능	불가능	불가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	공정 번호	–	1

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	–	○	–	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–

조건 플래그의 동작

이름	라벨	내용	
		SNXT	STEP
에러 플래그	P_ER	· S로 지정한 데이터가 내부 보조 릴레이 WR 영역의 범위에 없을 때 ON · 인터럽트 태스크로 사용될 때 ON · 그 이외일 때 OFF	· S의 데이터가 WR 영역에 없을 때 ON · 인터럽트 태스크로 사용될 때 ON · 그 이외일 때 OFF

SNXT/STEP

기능

■ SNXT

배치 위치에 따라 다음 3가지 동작을 수행합니다.

1) 스텝 래더 영역으로 진행

2) 다음 공정 번호로 진행

3) 스텝 래더 영역의 종료로 진행

스텝 래더 영역은 STEP 명령(공정 번호 지정)~STEP 명령(공정 번호 없음)까지의 영역입니다.

● 스텝 래더 영역으로 진행

(전체 스텝 래더 영역의 앞에 배치했을 때)

공정 번호 S로 지정한 릴레이를 OFF→ON으로 하고 공정 S(STEP S 명령 이후)로 진행합니다.

입력 조건을 기동 미분형으로 설정하십시오. (SNXT 명령을 스텝 래더 영역 밖에 배치하면 SET 명령과 동일한 동작이 되기 때문입니다.)

● 다음 공정으로 진행

(전체 스텝 래더 영역의 중간에 배치했을 때)

직전의 공정 번호(릴레이)를 ON→OFF로 하고, 다음 공정의 공정 번호(릴레이) S를 OFF→ON으로 하여 공정 S(STEP S 명령 이후)로 진행합니다.

● 스텝 래더 영역의 종료로 진행

(전체 스텝 래더 영역의 마지막에 배치했을 때)

직전의 공정 번호(릴레이)를 ON→OFF로 합니다. S로 지정하는 공정 번호는 더미 릴레이입니다. (단, ON이 되므로 ON이 되어도 문제 없는 릴레이를 지정하십시오.)

■ STEP

배치 위치 및 공정 번호 지정 유무에 따라 다음 2가지 동작을 수행합니다.

1) 지정된 공정의 시작

2) 스텝 래더 영역 전체 종료

● 지정된 공정의 시작

(각 공정의 직전에 배치하고 S를 지정했을 때)

SNXT 명령을 통해 공정 번호 S가 OFF→ON으로 기동했을 때 이 명령의 다음 명령부터 실행합니다. 동시에 A200.12(스텝 래더 1사이클 ON 플래그)를 ON으로 합니다.

다음 사이클 이후, 다음 공정의 이동 조건이 성립할 때까지(SNXT 명령에 의한 다음 공정 번호의 OFF→ON까지) 현재 공정을 반복 실행합니다.

SNXT 명령을 통해 다음 공정 번호가 OFF→ON으로 기동되었을 때 이 명령으로 지정한 공정 번호 S는 리셋(ON→OFF)되고 현재 공정 번호 S의 공정은 IL 중(인터록 중)이 됩니다.

STEP 명령이 지정하는 공정 번호 S의 ON/OFF 상태에 따라 공정 번호 S의 공정 내 명령 및 출력은 다음과 같아집니다. 공정 번호 S가 ON→OFF로 리셋되면 인터록 상태(표 1 참조)가 됩니다.

공정 번호(릴레이)의 상태	동작
ON	공정 내의 명령을 평상시 동작으로 실행합니다.
ON→OFF	공정 내에서 사용한 릴레이 및 명령을 표 1의 IL(인터록) 상태로 설정합니다.
OFF	공정 내의 명령을 NOP 처리합니다.

표 1: IL(인터록) 상태

각 명령의 출력		상태
OUT, OUT NOT 명령으로 지정된 릴레이		모두 OFF
타이머계 명령(100ms 타이머 TIM/TIMX, 10ms 타이머 TIMH/ TIMHX, 1ms 타이머 TMHH/TMHHX, 장시간 타이머 TIML/TIMLX 명령)	현재값	#0000(리셋)
	업 플래그	OFF(리셋)
0.1ms 타이머 TIMU/TIMUX ＊10.01ms 타이머 TMUH/TMUHX *1	참조 불가	
기타 명령(주)으로 지정된 릴레이, 채널		그 이전 상태를 유지 (명령 자체가 실행되지 않음)

주: TTIM/TTIMX, MTIM/MTIMX, SET, RSET, CNT/CNTX, CNTR/CNTRX, SFT, KEEP 등의 기타 모든 명령을 가리킵니다.

*1: CJ2, CJ1-H-R CPU 유니트만 해당

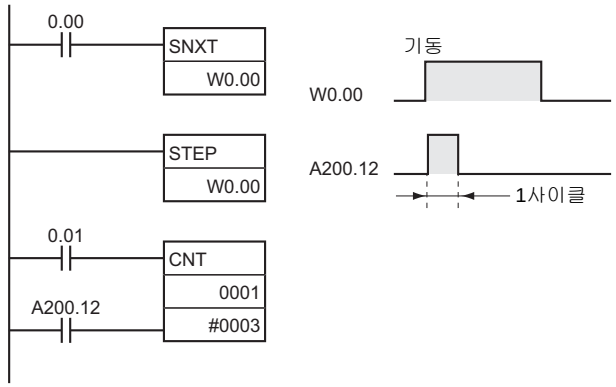
각 공정의 직전에 반드시 삽입합니다.

스텝 래더 영역의 처음에 배치하면 스텝 래더 영역의 시작을 나타냅니다.

· 스텝 래더 영역 전체 종료
(전체 스텝 래더 영역의 마지막에 배치하고 S를 지정하지 않았을 때)
SNXT 명령을 통해 SNXT 명령 직전의 공정 번호가 ON→OFF로 종료되면 스텝 래더 영역 전체가 종료됩니다.

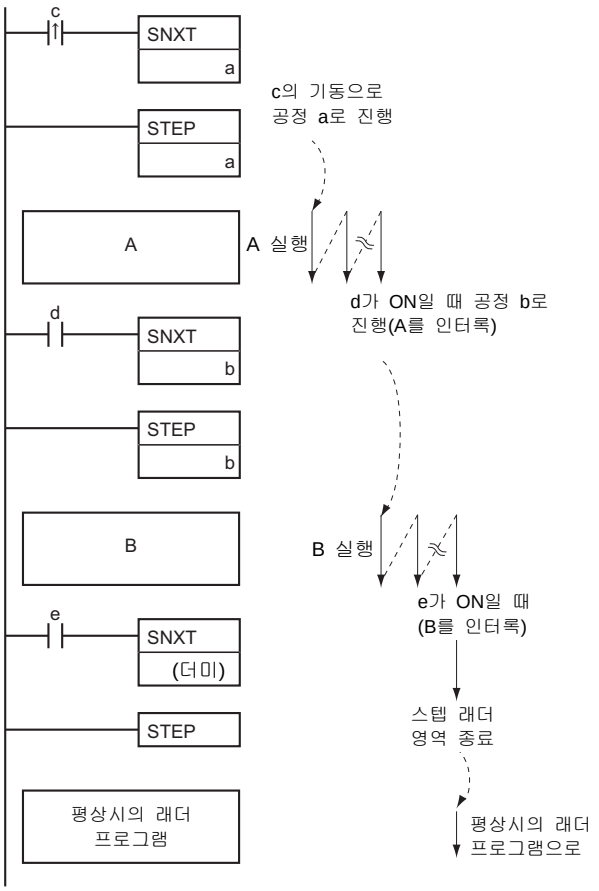
힌트

공정을 시작할 때 1회 1사이클 동안만 ON이 되는 특수 보조 릴레이 '스텝 래더 1사이클 ON 플래그' (A200.12)가 있습니다. 이 플래그를 사용하여 공정에 들어갔을 때의 초기화 처리 등을 수행하십시오.



관련 특수 보조 릴레이

이름	주소	내용
스텝 래더 1사이클 ON 플래그	A200.12	STEP 명령을 통해 1개 공정을 기동한 경우 1사이클, 1(ON)이 됩니다. 공정 기동 시의 카운터 초기화 등에 사용합니다.



SNXT/STEP

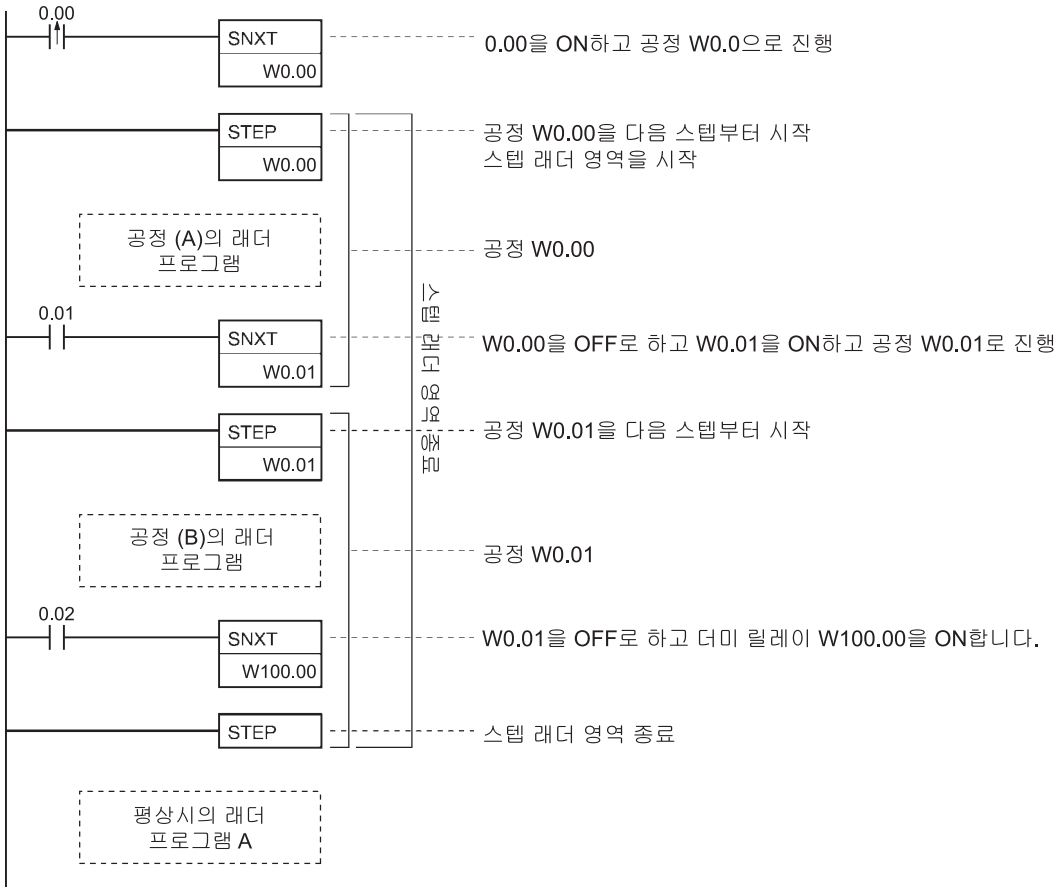
사용상의 주의 사항

- SNXT/STEP 명령으로 공정 번호 S에 지정할 수 있는 영역 종류는 내부 보조 릴레이 WR만 가능하므로 주의하십시오.
- SNXT/STEP 명령에서 S에 공정 번호를 지정하는 주소는 일반 래더 회로에서 사용하는 주소와 중복하여 사용할 수 없습니다. 중복 사용한 경우 코일 2중 사용 에러가 됩니다.
- 공정 내에 SBS(서브루틴 호출) 명령이 있는 경우는 공정 번호가 ON→OFF가 되어도 서브루틴 내의 출력은 IL(인터록 상태)이 되지 않으므로 주의하십시오.
- SNXT 명령은 입력의 기동(OFF→ON) 1회만 실행됩니다.
- 최종 공정의 마지막에는 SNXT 명령을 삽입하며 해당 공정 번호는 공정 번호로 사용하지 않는 내부 보조 릴레이 WR를 더미 릴레이 번호로 사용하십시오. 사용하는 릴레이를 공정 번호에 지정하면 해당 공정을 동작시키게 됩니다.
- 서브루틴 내, 인터럽트 태스크 내, 블록 프로그램 영역 내에서는 사용할 수 없습니다.
- 동일 사이클 내에서 2군데 이상의 스텝 래더 영역을 동시에 동작시킬 수 없습니다.

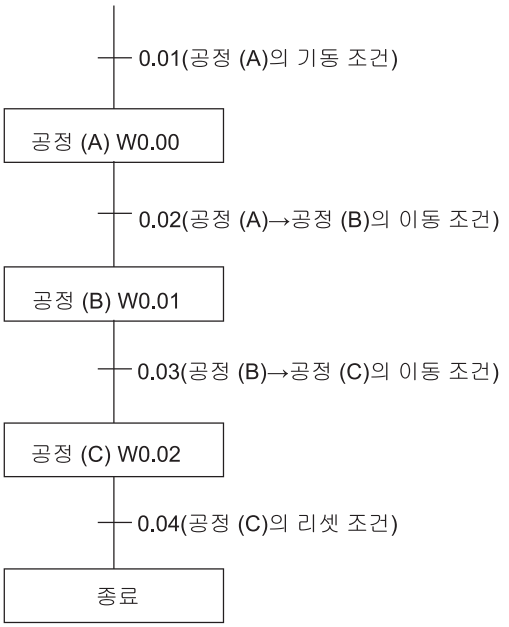
스텝 래더 프로그램 영역 내에서 사용할 수 없는 명령은 다음과 같습니다.

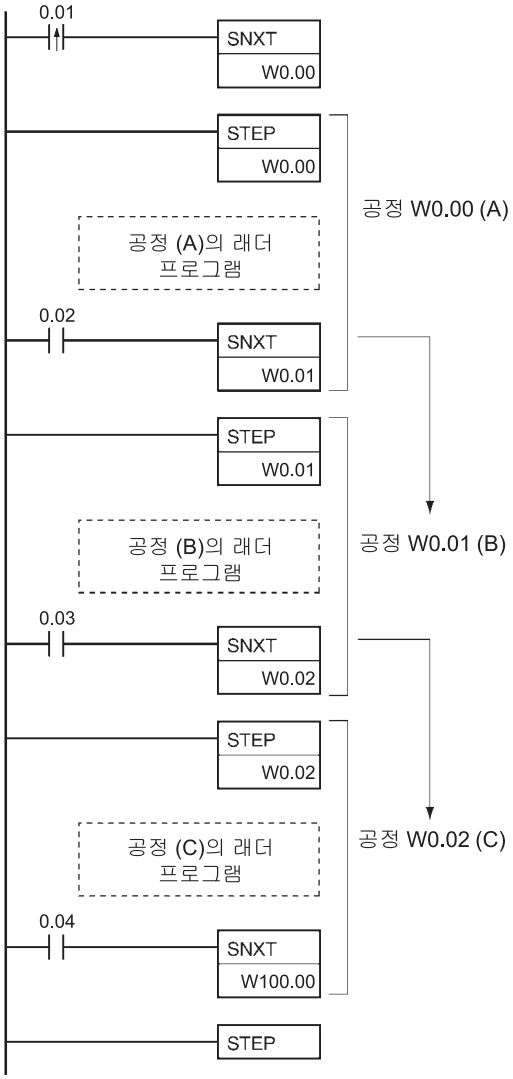
기능별 분류	니모닉	명령
시퀀스 제어	END	종료
	IL	인터록
	ILC	인터록 클리어
	JMP	점프
	JME	점프 종료
	CJP	조건부 점프
	CJPN	조건부 부정 점프
	JMP0	복수 점프
	JME0	복수 점프 종료
서브루틴	SBN	서브루틴 엔트리
	RET	서브루틴 반환

샘플 프로그램

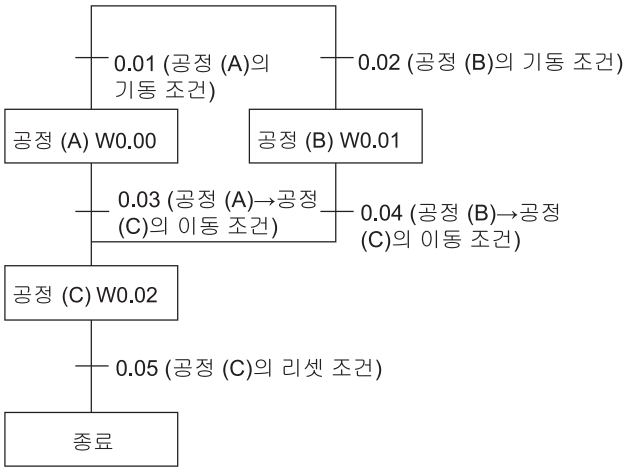


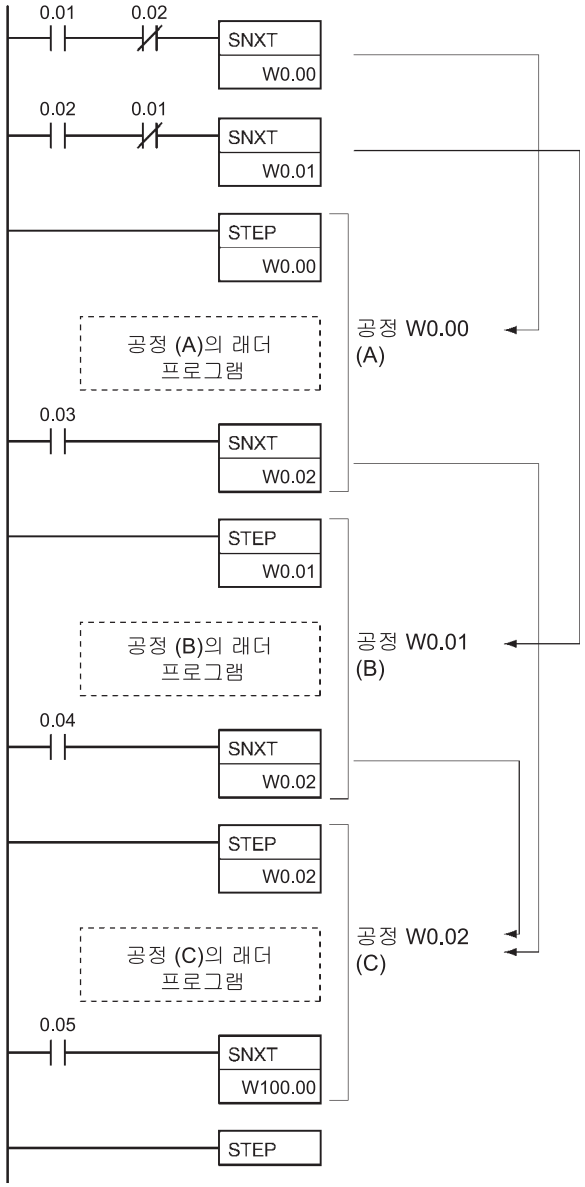
(1) 순서 제어


0.01(공정 (A)의 기동 조건)
공정 (A) W0.00
0.02(공정 (A)→공정 (B)의 이동 조건)
공정 (B) W0.01
0.03(공정 (B)→공정 (C)의 이동 조건)
공정 (C) W0.02
0.04(공정 (C)의 리셋 조건)
종료


0.01
SNXT
W0.00
STEP
W0.00
공정 W0.00 (A)
공정 (A)의 래더 프로그램
0.02
SNXT
W0.01
STEP
W0.01
공정 W0.01 (B)
공정 (B)의 래더 프로그램
0.03
SNXT
W0.02
STEP
W0.02
공정 W0.02 (C)
공정 (C)의 래더 프로그램
0.04
SNXT
W100.00
STEP

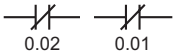
(2) 선택 분기 제어





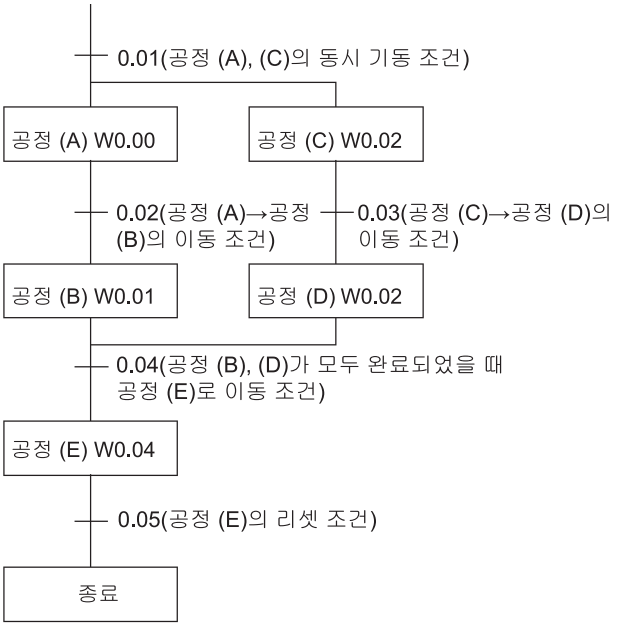
● 참고

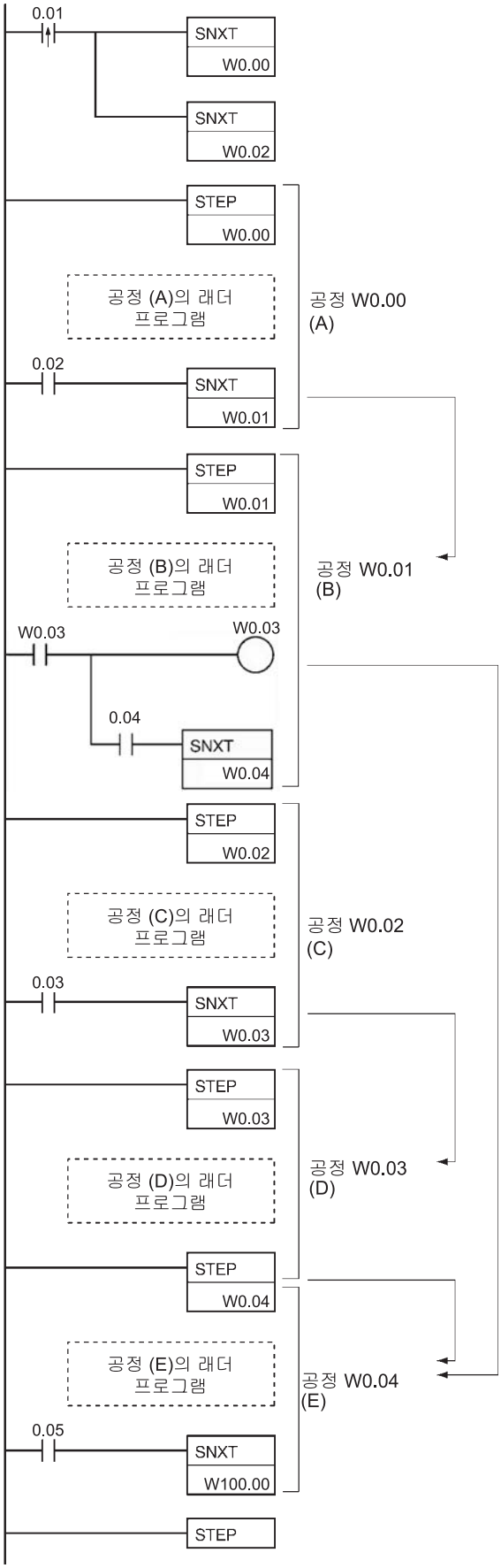
- 위 예에서 SNXT의 W0.02와 같이 선택 분기를 통해 다음 공정을 기동하는 경우, 코일 이중 사용 상태가 되지만 CX-Programmer의 프로그램 확인에서는 코일 이중 사용 에러가 발생하지 않습니다.
- 스텝 래더 회로에서 코일 이중 사용 에러가 되는 것은 일반 래더 회로에서 사용한 릴레이 번호를 스텝 래더 명령에서 사용했을 때뿐입니다.
- 위의 프로그램은 공정 A, B를 동시에 진행할 수 없는 경우입니다.
- 공정 A, B의 동시 진행이 가능한 경우,



을 삭제하십시오.

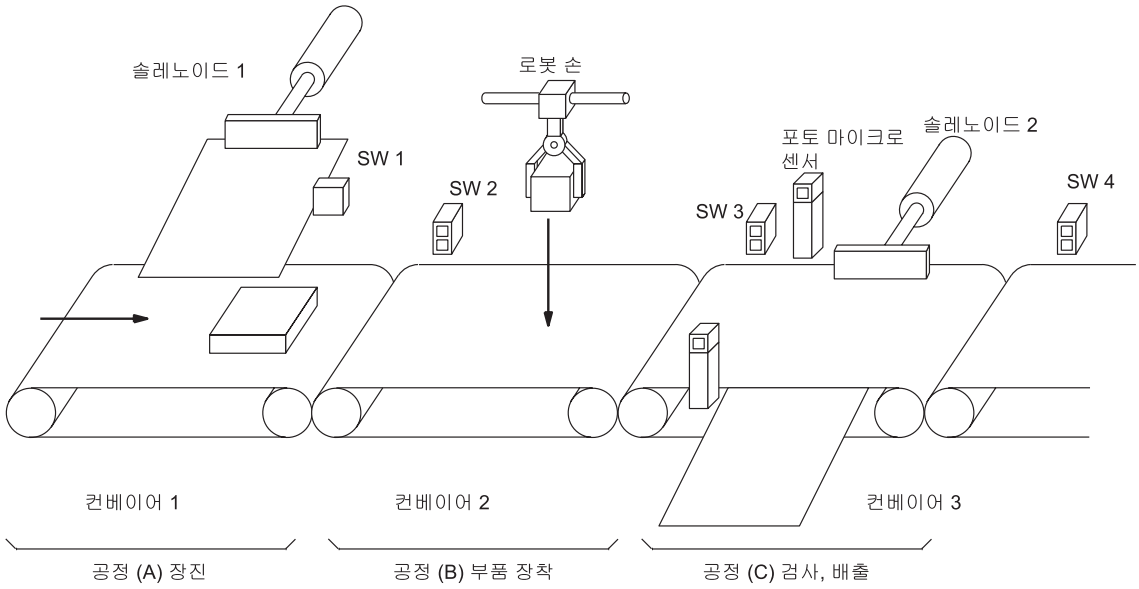
(3) 병렬 분기 합류 제어

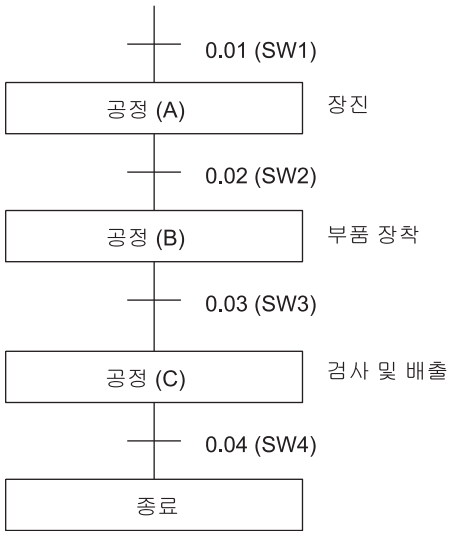

0.01(공정 (A), (C)의 동시 기동 조건)
공정 (A) W0.00
공정 (C) W0.02
0.02(공정 (A)→공정 (B)의 이동 조건)
0.03(공정 (C)→공정 (D)의 이동 조건)
공정 (B) W0.01
공정 (D) W0.02
0.04(공정 (B), (D)가 모두 완료되었을 때 공정 (E)로 이동 조건)
공정 (E) W0.04
0.05(공정 (E)의 리셋 조건)
종료

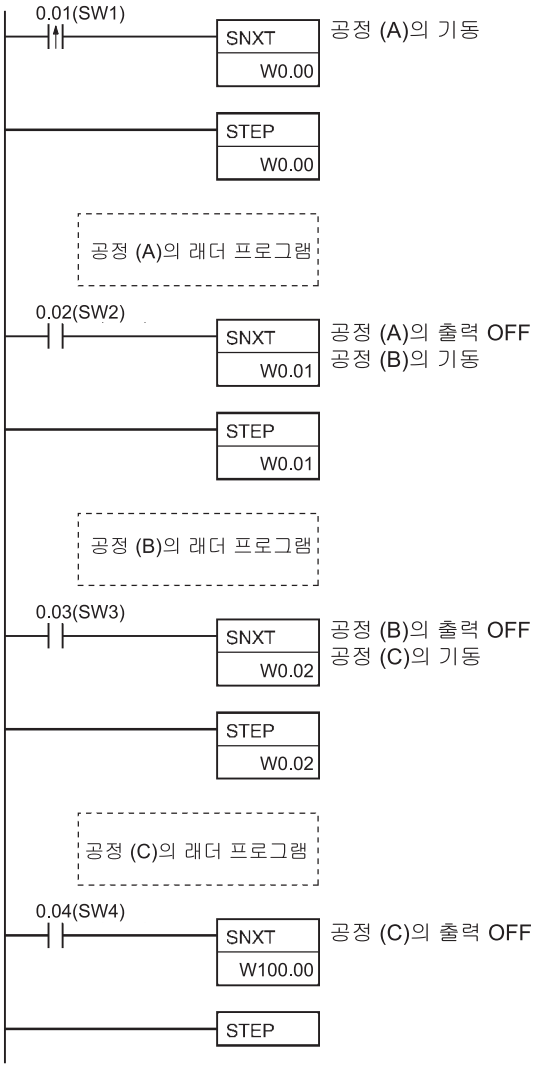

0.01
SNXT W0.00
SNXT W0.02
STEP W0.00
공정 (A)의 래더 프로그램
공정 W0.00 (A)
0.02
SNXT W0.01
STEP W0.01
공정 (B)의 래더 프로그램
공정 W0.01 (B)
W0.03
W0.03
0.04
SNXT W0.04
STEP W0.02
공정 (C)의 래더 프로그램
공정 W0.02 (C)
0.03
SNXT W0.03
STEP W0.03
공정 (D)의 래더 프로그램
공정 W0.03 (D)
STEP W0.04
공정 (E)의 래더 프로그램
공정 W0.04 (E)
0.05
SNXT W100.00
STEP

응용 예

(1) 순서 제어



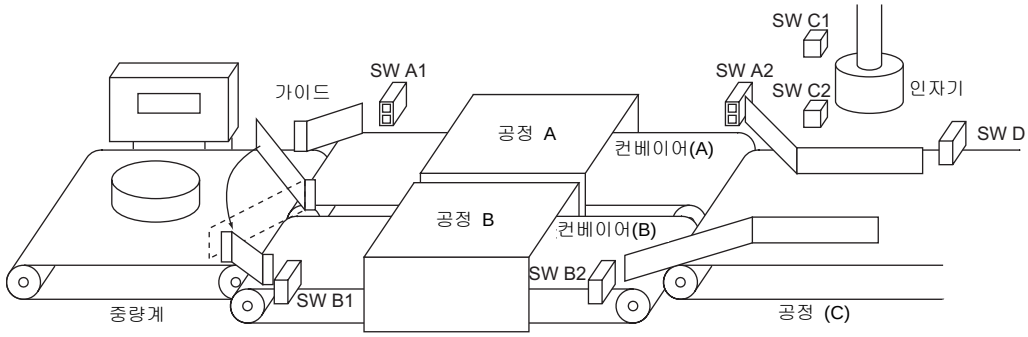


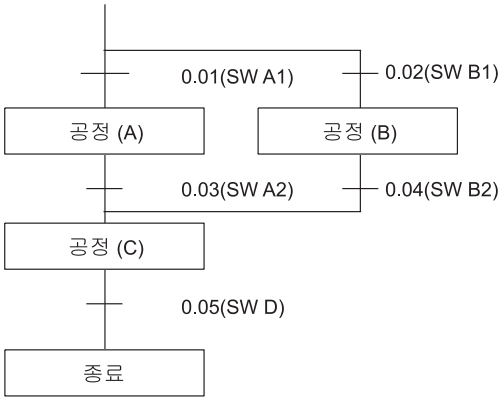


동작 설명
① SW1의 ON에 의해
  솔레노이드 1 작동
  컨베이어 1 작동 } 공정 (A)
② SW2의 ON에 의해 이전 공정 정지
  로봇 손 작동
  컨베이어 2 작동 } 공정 (B)
③ SW3의 ON에 의해 이전 공정 정지
  포토 마이크로 센서 작동(부품 검사용)
  컨베이어 3 작동
  솔레노이드 2 작동(불량품 배출) } 공정 (C)
④ SW4의 ON에 의해 이전 공정 정지

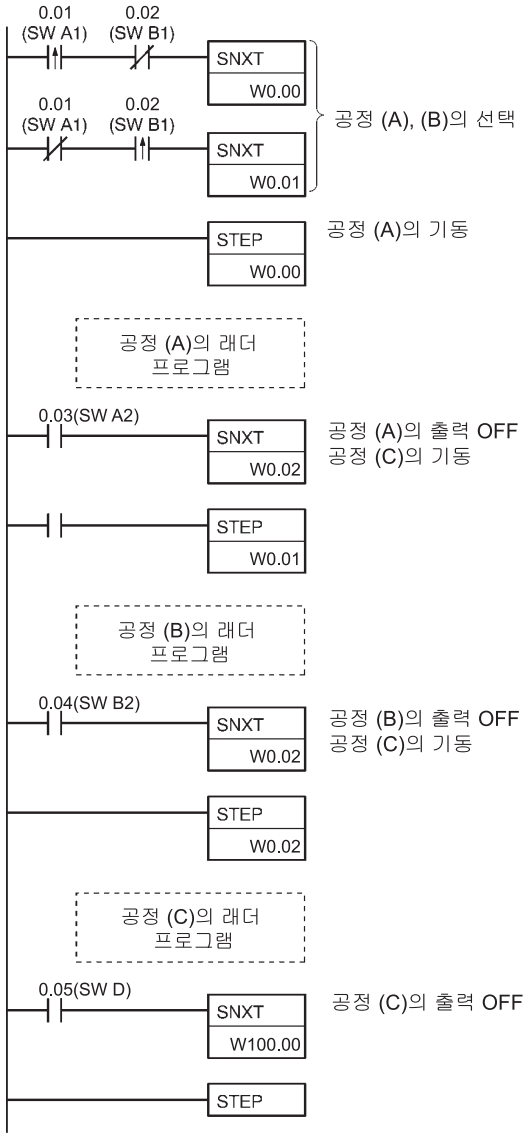
참고
공정 내에서 다른 공정을 시작하면(SNXT 명령 기동 시) 자체 공정의 모든 출력이 해당 사이클 내에서 OFF됩니다.

(2) 선택 분기 제어

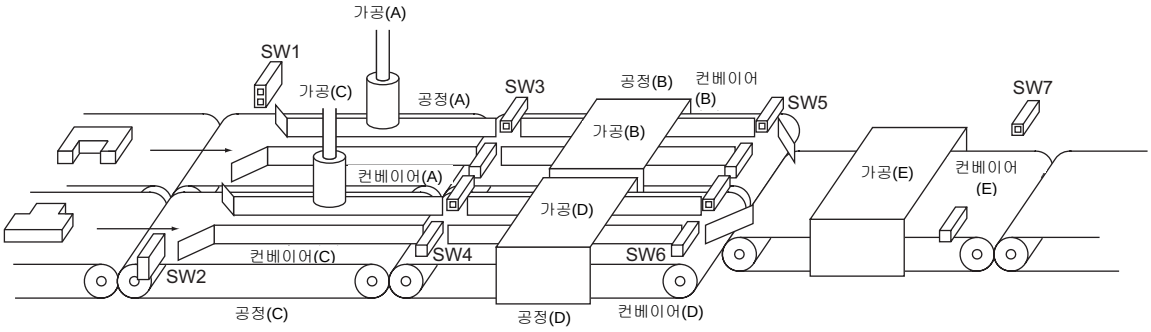


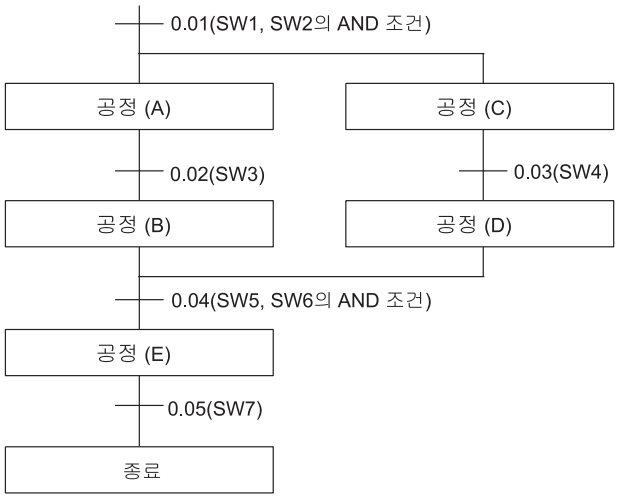


동작 설명
중량에 따라 가이드에 의해 제품이 분류됩니다.
① SWA1 ON에 의해
컨베이어 (A) 작동
가공기 (A) 작동 } 공정 (A)
② SWB1 ON에 의해
컨베이어 (B) 작동
가공기 (B) 작동 } 공정 (B)
③ SWA2 ON에 의해 공정 A 정지
인자기 작동(down)
SWC2 ON에 의해 UP } 공정 (C)
④ SWB2 ON에 의해 공정 B 정지
인자기 작동(down)
SWC2 ON에 의해 UP } 공정 (C)
⑤ SWD ON에 의해 인자기 정지



(3) 병렬 분기 합류 제어

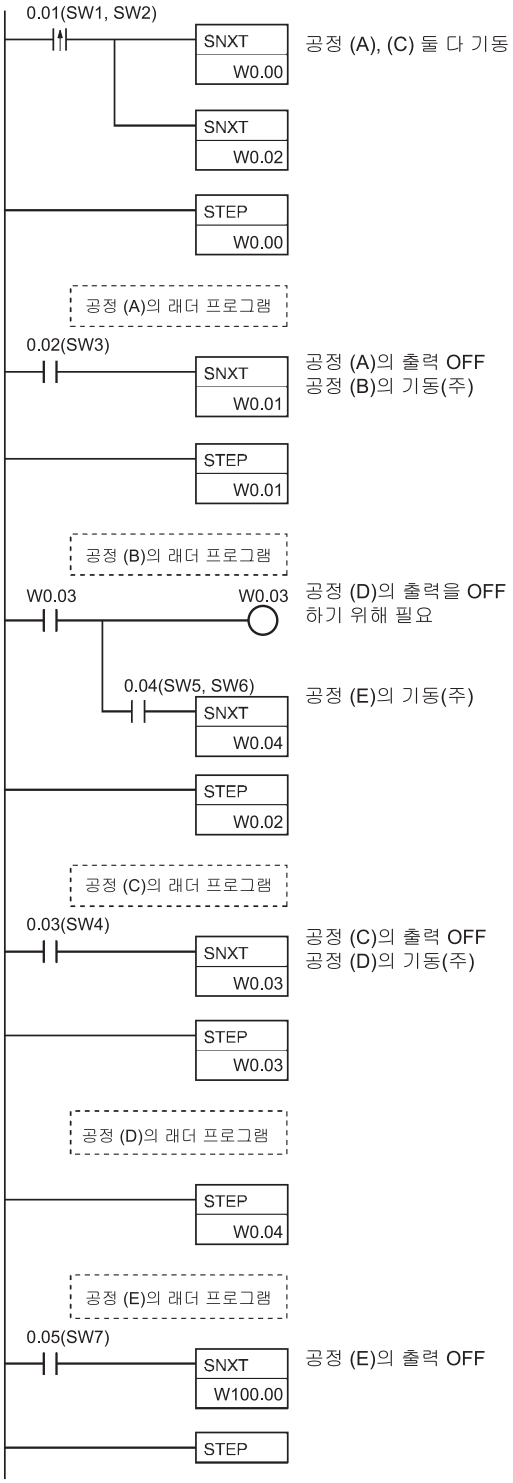




동작 설명
① SW1, SW2의 ON에 의해
　컨베이어 (A) 작동 ┐
　가공 (A) 작동 ┘ 공정 (A)
　컨베이어 (C) 작동 ┐
　가공 (C) 작동 ┘ 공정 (C)
② SW3의 ON에 의해
　공정 (A) 정지 ┐
　컨베이어 (B) 작동 ├ 공정 (B)
　가공 (B) 작동 ┘
③ SW4의 ON에 의해
　공정 (C) 정지 ┐
　컨베이어 (D) 작동 ├ 공정 (D)
　가공 (D) 작동 ┘
④ SW5, SW6의 ON에 의해
　공정 (B) 정지 ┐
　공정 (D) 정지 ├ 공정 (E)
　컨베이어 (E) 작동 ┘
　가공 (E) 작동 ┘
⑤ SW7의 ON에 의해
　공정 (E) 정지

주: 공정 (B), 공정 (D)가 작동 중이고 SW5, SW6이 ON이면 공정 (B), (D)를 종료로 판단합니다. SNXT W0.04의 실행을 통해 공정 (B)의 출력은 OFF되고 W0.03이 OFF가 됩니다. STEP W0.03은 ON→OFF로의 기동을 판단하여 공정 (D)의 출력을 OFF합니다.

참고
STEP 명령은 ON→OFF로 기동될 때 자체 공정의 출력을 모두 OFF합니다.

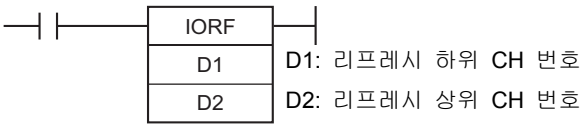


I/O 유니트용 명령

니모닉	명령 이름	펑션 번호	페이지
IORF	I/O 리프레시	097	3-650
FIORF	고기능 I/O 유니트 매회 리프레시	225	3-653
DLNK	CPU 고기능 유니트 매회 I/O 리프레시	226	3-656
SDEC	7세그먼트 디코더	078	3-659
DSW	디지털 스위치	210	3-661
TKY	10키 입력	211	3-664
HKY	16키 입력	212	3-667
MTR	행렬 입력	213	3-670
7SEG	7세그먼트 표시	214	3-674
AIDC	아날로그 입력 직접 변환	216	3-677
AODC	아날로그 출력 직접 변환	217	3-680
NCDMV	위치 결정 고속 기동	218	3-683
NCDTR	위치 결정 시작 트리거	219	3-688
IORD	인텔리전트 I/O 읽기	222	3-691
IOWR	인텔리전트 I/O 쓰기	223	3-694

IORF

명령 이름	니모닉	동작 옵션	평션 번호	기능
I/O 리프레시	IORF	@IORF	097	지정된 I/O 채널 데이터를 리프레시합니다.

심볼	IORF
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	가능	가능	가능	가능	불가능

피연산자 설명

피연산자	내용	데이터 형식	크기
D1	리프레시 하위 CH 번호	–	가변
D2	리프레시 상위 CH 번호	–	가변

D1: 리프레시 하위 CH 번호
CIO 영역 내, 입출력 릴레이 영역(0000~0999CH) 또는 고기능 I/O 유니트 릴레이 영역(2000~2959CH)
D2: 리프레시 상위 CH 번호
CIO 영역 내, 입출력 릴레이 영역(0000~0999CH) 또는 고기능 I/O 유니트 릴레이 영역(2000~2959CH)

주: D1과 D2는 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
D1, D2	○	–	–	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· D1 > D2일 때 ON · D1의 주소와 D2의 주소가 다른 영역일 때 ON · CS1D CPU 유니트에서 활성-대기 사이에서 동기화되지 않았을 때 ON · 그 이외일 때 OFF

IORF 명령에 의한 리프레시 대상 유니트

장치 위치	CPU 장치 또는 증설 장치(SYSBUS 원격 I/O 자국 장치 제외)		리프레시 여부 ○: 가능 ×: 불가능
유니트 종류	기본 I/O 유니트	CS 시리즈 기본 I/O 유니트	○
		C200H 기본 I/O 유니트(CS 시리즈만 해당)	○
		C200H 그룹 2 다점 I/O의 기본 I/O 유니트(CS 시리즈만 해당)	○
		CJ 시리즈 기본 I/O 유니트	○
	고기능 I/O 유니트(다점 I/O 유니트를 포함하는 모든 고기능 I/O 유니트의 할당 릴레이 영역)		○

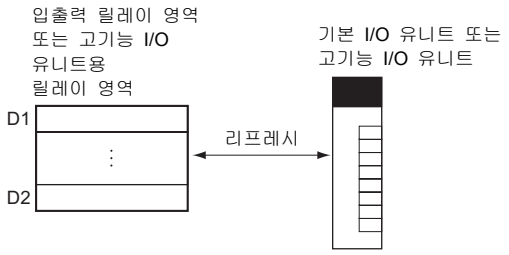
주: 매회 리프레시(!) 지정에 따라 리프레시 가능한 유니트와 IORF(I/O 리프레시) 명령으로 리프레시 가능한 유니트가 달라지므로 주의하십시오.

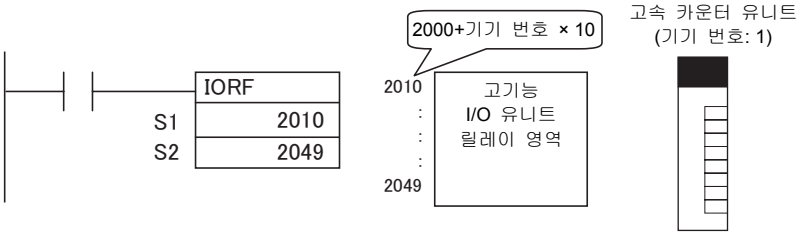
기능

D1로 지정한 리프레시 하위 CH 번호부터 D2로 지정한 리프레시 상위 CH 번호까지의 I/O 채널 데이터를 리프레시합니다.

리프레시 대상은 CPU 장치 및 증설 장치에 장착된 기본 I/O 유니트와 고기능 I/O 유니트입니다. 기본 I/O 유니트용 입출력 릴레이 영역(0000~0999CH)과 고기능 I/O 유니트 릴레이 영역(2000~2959CH)이 대상 범위입니다.

고기능 I/O 유니트를 리프레시하는 경우, 해당 영역의 시작 채널을 지정 범위에 포함시켜야 합니다.
또한 지정된 고기능 I/O 유니트는 해당 영역을 모두 리프레시합니다.
예를 들어 4호기를 점유 중인 고속 카운터 유니트의 기기 번호가 1인 경우, 고기능 I/O 유니트 릴레이는 2010~2049CH의 40CH분이 되지만 IORF 명령에서 2010CH를 지정해야 합니다.
그 결과, 2010~2049CH가 모두 리프레시됩니다.





힌트

이 명령과 FIORF/DLNK 명령의 차이는 다음 표와 같습니다.

명령	대상 유니트
IORF 명령	· 기본 I/O 유니트의 I/O 리프레시 · 고기능 I/O 유니트의 할당 릴레이 영역 및 할당 DM 영역의 I/O 리프레시
FIORF 명령	· 고기능 I/O 유니트의 할당 릴레이 영역 및 할당 DM 영역의 I/O 리프레시
DLNK 명령	· CPU 고기능 유니트의 할당 릴레이 영역(25CH)의 I/O 리프레시 · CPU 고기능 유니트의 할당 DM 영역(100CH)의 I/O 리프레시 · 데이터 링크, DeviceNet 원격 I/O 통신 등과 같은 유니트 고유의 리프레시

사용상의 주의 사항

· 이 명령을 인터럽트 태스크 중에 사용할 수 있습니다. (그 결과, 인터럽트에 의한 특정 입출력의 고속 응답 처리를 실현할 수 있습니다.)
단, 이런 경우 반드시 PLC 시스템 설정의 '고기능 I/O 유니트 주기적인 리프레시 유무 지정'을 고기능 I/O 유니트와 주기적인 리프레시 금지(1: 주기적 리프레시 안 함)로 설정하십시오.
고기능 I/O 유니트와의 주기적 리프레시(0: 주기적 리프레시 수행)로 설정한 상태에서 인터럽트 태스크 중에 다음 중 하나를 실행하면 '다중 리프레시'가 되어 운전 계속 이상이 발생하고 다중 리프레시 이상 플래그(A402.13)가 ON이 됩니다.

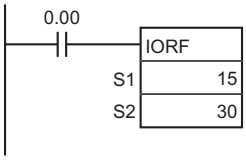
IORF

1)IORF/FIORF 명령을 통해 동일한 고기능 I/O 유니트에 대해 I/O 리프레시를 실행했을 때

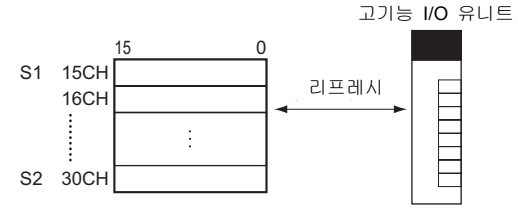
2)IORD/IOWR 명령을 통해 동일한 고기능 I/O 유니트의 메모리 영역에 대해 읽기 또는 쓰기를 실행했을 때

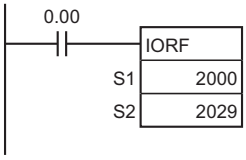
- 고기능 I/O 유니트에 대한 주기적인 리프레시를 금지하는 경우, 운전 시작 이후 11초 이하의 간격으로 해당 유니트에 대한 IORF 명령 또는 FIORF 명령(*1)이 실행되도록 하십시오.
  주기적인 리프레시를 금지한 고기능 I/O 유니트에 대해 11초 이상 IORF 명령 또는 FIORF 명령이 실행되지 않는 경우에는 고기능 I/O 유니트에서 CPU 유니트 감시 이상이 발생합니다(ERH LED 또는 RUN LED 점등).
- 리프레시 대상 범위(D1~D2) 내에서 유니트가 장착되지 않은 채널에 대해서는 아무 처리도 하지 않고, 유니트가 장착된 유니트에 대해서만 리프레시 처리를 수행합니다.
- 입출력 릴레이 영역(0000-0999CH)과 고기능 I/O 유니트 릴레이 영역(2000-2959CH)에 걸쳐 지정하면 에러가 됩니다.
- 리프레시 대상 유니트에 C200H 시리즈 고기능 I/O 유니트와 CS 시리즈 고기능 I/O 유니트가 혼재해도 됩니다(CS 시리즈만 해당).
- C200H 그룹 2 다점 I/O 유니트에 대해서는 유니트 단위의 리프레시를 수행합니다. 단, 리프레시 하위 CH 번호가 점유 채널 중간부터 지정된 경우는 해당 유니트 영역의 리프레시를 수행하지 않습니다. 리프레시 상위 CH 번호가 점유 채널 중간까지 지정된 경우는 해당 유니트의 전체 영역을 리프레시합니다(CS 시리즈만 해당).
- I/O 조회 이상이 발생한 유니트에 대해서는 리프레시 처리를 수행하지 않습니다(CS 시리즈만 해당).
- IORF 명령을 사용한 I/O 리프레시 처리 중에 I/O 버스 이상이 발생한 경우 I/O 리프레시 처리를 중지합니다.

샘플 프로그램

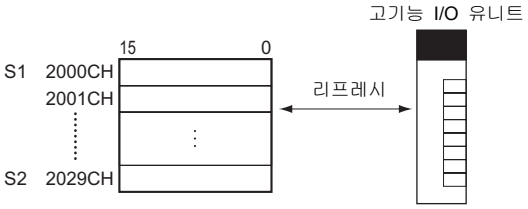


입출력 릴레이 영역의 리프레시
0.00이 ON이면 15~30CH의 16CH(256점)의
입출력 릴레이 영역을 리프레시합니다.



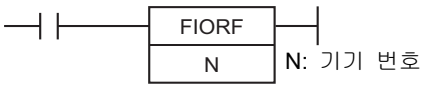


고기능 I/O 유니트 릴레이 영역의 리프레시
0.00이 ON이면 2000~2029CH의 30CH(480점)의
고기능 I/O 유니트 릴레이 영역을 리프레시합니다.



FIORF

명령 이름	니모닉	동작 옵션	평션 번호	기능
고기능 I/O 유니트 매회 리프레시	FIORF	@FIORF	225	지정된 고기능 I/O 유니트와의 I/O 리프레시를 매회 실행합니다.

심볼	FIORF
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	기기 번호	UNIT	1

N: 기기 번호

고기능 I/O 유니트의 기기 번호(10진수 &0~95 또는 16진수 #0000~005F)

주: 여러 기기를 점유하는 고기능 I/O 유니트에 대해서는 최하위 기기 번호를 지정하십시오.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 기기 번호가 10진수 &0~95 또는 16진수 #0000~005F 이외일 때 ON · N으로 지정한 기기 번호의 고기능 I/O 유니트가 존재하지 않을 때 ON · N으로 지정한 기기 번호가 여러 기기를 점유하는 고기능 I/O 유니트의 최하위 번호가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 리프레시가 정상적으로 종료되었을 때 ON · 고기능 I/O 유니트의 주기적 리프레시 중에 이 명령을 실행했을 때 OFF

FIORF 명령에 의한 리프레시 대상 유니트

장치 위치	유니트 종류	리프레시 여부 ○: 가능 ×: 불가능
CPU 장치 또는 증설 장치 (SYSBUS 원격 I/O 자국 장치 제외)	기본 I/O 유니트	×
	고기능 I/O 유니트의 다음 영역(유니트 단위로 일괄 리프레시) · 할당 릴레이 영역 · 할당 DM 영역	○
	CPU 고기능 유니트	×

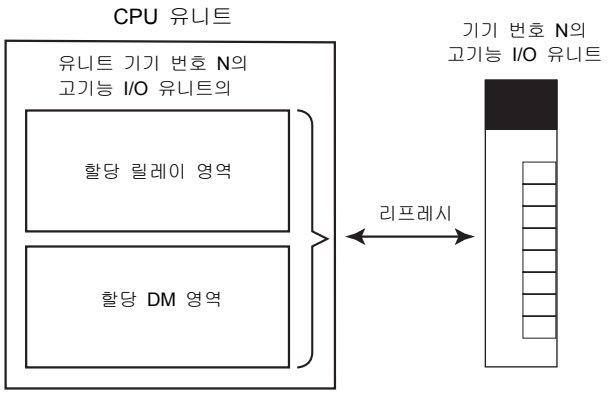
FIORF

기능

N으로 지정한 기기 번호의 고기능 I/O 유니트와 할당 릴레이 영역 및 할당 DM 영역의 매회 리프레시를 실행합니다.

매회 리프레시를 실행하는 영역은 각 고기능 I/O 유니트의 사용 설명서를 참조하십시오.

특정 고기능 I/O 유니트와의 주기적인 I/O 리프레시 시간의 사이클 시간으로 가산하지 않고 필요할 때만 고기능 I/O 유니트와 I/O 리프레시를 수행하려면, PLC 시스템 설정에서 기기 번호의 고기능 I/O의 주기적 리프레시 유무 설정을 없음으로 설정한 후 이 FIORF 명령을 실행합니다.



힌트

· FIORF 명령과 IORF/DLNK 명령의 차이

이 명령과 IORF/DLNK 명령의 차이는 다음 표와 같습니다.

명령	대상 유니트
IORF 명령	· 기본 I/O 유니트의 I/O 리프레시 · 고기능 I/O 유니트의 할당 릴레이 영역 및 할당 DM 영역의 I/O 리프레시
FIORF 명령	· 고기능 I/O 유니트의 할당 릴레이 영역 및 할당 DM 영역의 I/O 리프레시
DLNK 명령	· CPU 고기능 유니트의 할당 릴레이 영역(25CH)의 I/O 리프레시 · CPU 고기능 유니트의 할당 DM 영역(100CH)의 I/O 리프레시 · 데이터 링크, DeviceNet 원격 I/O 통신 등과 같은 유니트 고유의 리프레시

· 이 명령을 통한 고기능 I/O 유니트의 I/O 리프레시는 IORF 명령을 사용한 I/O 리프레시와 다음과 같은 점에서 다릅니다.
  · 명령 실행 시간이 고속
  · 기기 번호로 리프레시 범위 지정 가능

사용상의 주의 사항

· 이 명령을 인터럽트 태스크 중에 사용할 수 있습니다. (그 결과, 인터럽트에 의한 특정 입출력의 고속 응답 처리를 실현할 수 있습니다.)
단, 이런 경우 반드시 PLC 시스템 설정의 '고기능 I/O 유니트 주기적인 리프레시 유무 지정'을 고기능 I/O 유니트와 주기적인 리프레시 금지(1: 주기적 리프레시 안 함)로 설정하십시오.
고기능 I/O 유니트와의 주기적 리프레시(0: 주기적 리프레시 수행)로 설정한 상태에서 인터럽트 태스크 중에 다음 중 하나를 실행하면 '다중 리프레시'가 되어 운전 계속 이상이 발생하고 인터럽트 태스크 이상 플래그(A402.13)가 ON이 됩니다.

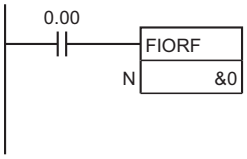
  1) IORF/FIORF 명령을 통해 동일한 고기능 I/O 유니트에 대해 I/O 리프레시를 실행했을 때

  2) IORD/IOWR 명령을 통해 동일한 고기능 I/O 유니트의 메모리 영역에 대해 읽기 또는 쓰기를 실행했을 때

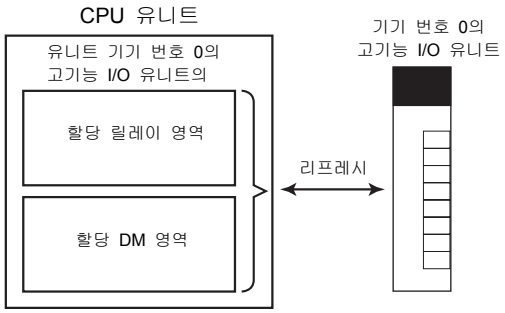
· 고기능 I/O 유니트에 대한 주기적인 리프레시를 금지하는 경우, 운전 시작 이후 11초 이하의 간격으로 해당 유니트에 대한 IORF 명령 또는 FIORF 명령이 실행되도록 하십시오.
주기적인 리프레시를 금지한 고기능 I/O 유니트에 대해 11초 이상 IORF 명령 또는 FIORF 명령이 실행되지 않는 경우에는 고기능 I/O 유니트에서 CPU 유니트 감시 이상이 발생합니다(ERH LED 또는 RUN LED 점등).

· FIORF 명령을 사용한 I/O 리프레시 처리 중에 I/O 버스 이상이 발생한 경우 I/O 리프레시 처리를 중지합니다.

샘플 프로그램



0.00이 ON이면 기기 번호 0의 고기능 I/O 유니트의 할당 릴레이 영역 및 할당 DM 영역을 매회 I/O 리프레시합니다.



DLNK

명령 이름	니모닉	동작 옵션	평션 번호	기능
CPU 고기능 유니트 매회 I/O 리프레시	DLNK	@DLNK	226	지정된 CPU 고기능 유니트와의 I/O 리프레시를 매회 실행합니다.

심볼	DLNK
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	유니트 번호	UINT	1

N: 유니트 번호
CPU 고기능 유니트의 유니트 번호(10진수 &0~15 또는 16진수 #0000~000F)

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 유니트 번호가 10진수 &0~15 또는 16진수 #0000~000F 이외일 때 ON · N으로 지정한 유니트 번호의 CPU 고기능 유니트가 존재하지 않을 때 ON · CS1D CPU 유니트에서 활성-대기 사이에서 동기화되지 않았을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· CPU 고기능 유니트가 데이터 갱신 중이어서 매회 I/O 리프레시를 실행하지 못했을 때 OFF · N으로 지정한 유니트 번호의 CPU 고기능 유니트가 CPU 고기능 유니트 이상 또는 CPU 고기능 유니트 설정 이상일 때 OFF · 인터럽트 태스크에서 이 명령을 사용한 경우 일반 I/O 리프레시와 경합하여 다중 리프레시가 발생했을 때 OFF · 리프레시가 정상적으로 종료되었을 때 ON

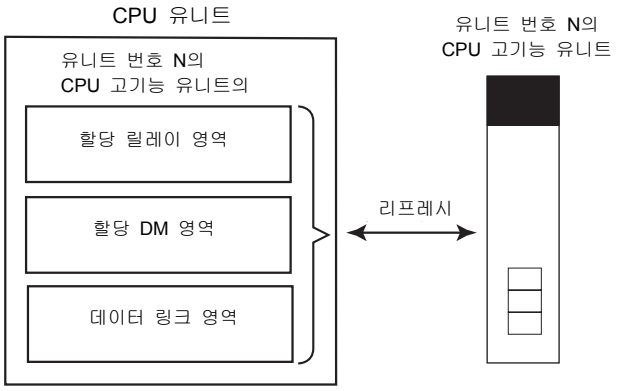
기능

N으로 지정한 유니트 번호의 CPU 고기능 유니트와 아래 ①, ②의 매회 I/O 리프레시를 실행합니다.
단, 매회 I/O 리프레시의 실행 조건에 대해서는 뒤에 설명하는 힌트를 참조하십시오.

① CPU 고기능 유니트 할당 릴레이 영역(25CH) 및 할당 DM 영역(100CH)의 I/O 리프레시

② 다음의 데이터 링크, DeviceNet 원격 I/O 통신 등과 같은 유니트 고유의 리프레시(①과 함께 실행됨)

유니트 종류	유니트 고유의 리프레시
Controller Link 또는 SYMAC LINK 유니트	데이터 링크 리프레시
DeviceNet 유니트(주: C200H용 DeviceNet 마스터 유니트는 제외)	원격 I/O 통신의 리프레시



힌트

· DLNK 명령과 IORF/FIORF 명령의 차이

이 명령과 IORF/FIORF 명령의 차이는 다음 표와 같습니다.

명령	대상 유니트
IORF 명령	· 기본 I/O 유니트의 I/O 리프레시 · 고기능 I/O 유니트의 할당 릴레이 영역 및 할당 DM 영역의 I/O 리프레시
FIORF 명령	· 고기능 I/O 유니트의 할당 릴레이 영역 및 할당 DM 영역의 I/O 리프레시
DLNK 명령	· CPU 고기능 유니트의 할당 릴레이 영역(25CH)의 I/O 리프레시 · CPU 고기능 유니트의 할당 DM 영역(100CH)의 I/O 리프레시 · 데이터 링크, DeviceNet 원격 I/O 통신 등과 같은 유니트 고유의 리프레시

· 이 명령에 의한 리프레시는 CPU 유니트⇔CPU 고기능 유니트 간의 리프레시입니다. 다음 사항에 주의하십시오.

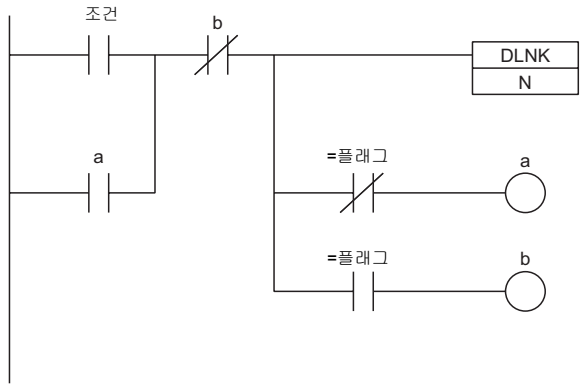
1. 데이터 링크 또는 DeviceNet 원격 I/O 통신의 통신 데이터 교환의 경우, 이 명령을 실행할 때 즉시 통신 대상과 데이터 교환이 실행되는 것이 아닙니다. 네트워크의 특정 통신 사이클 시간의 주기로 송수신 권한을 실제로 획득해야 비로소 데이터 교환이 실행됩니다. 따라서 최대로 통신 사이클 시간에 해당하는 지연이 발생합니다.
2. 이 명령은 CPU 고기능 유니트가 데이터 교환을 수행하는 동안 리프레시를 실행할 수 없습니다. 따라서 높은 빈도로 이 명령을 실행하면 리프레시가 실행되지 않습니다. 이 명령을 사용할 때는 통신 사이클 이하의 주기로 이 명령을 실행하지 않도록 권장합니다.

사용상의 주의 사항

· 이 명령은 CPU 고기능 유니트에 대해 I/O 리프레시를 실행하는 명령입니다. 이 명령을 실행할 때는 CPU 고기능 유니트와의 데이터 교환(예를 들어, Controller Link 유니트에서는 데이터 링크)에 시간이 필요합니다.

DLNK

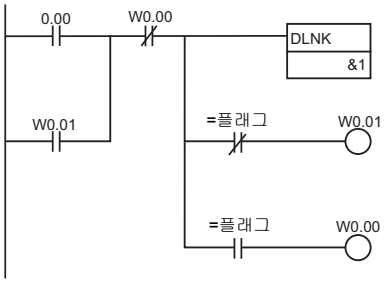
· CPU 고기능 유니트가 데이터 갱신 중인 경우 이 명령은 실행되지 않고 = 플래그가 OFF가 됩니다. 재시도를 실행할 수 있도록 다음의 재시도 처리를 래더 프로그램 내에 삽입할 것을 권장합니다.



· CPU 고기능 유니트 이상(A402.07이 ON) 또는 CPU 고기능 유니트 설정 이상(A402.03이 ON)이 발생한 유니트에 대해서는 I/O 리프레시 처리를 수행하지 않습니다.

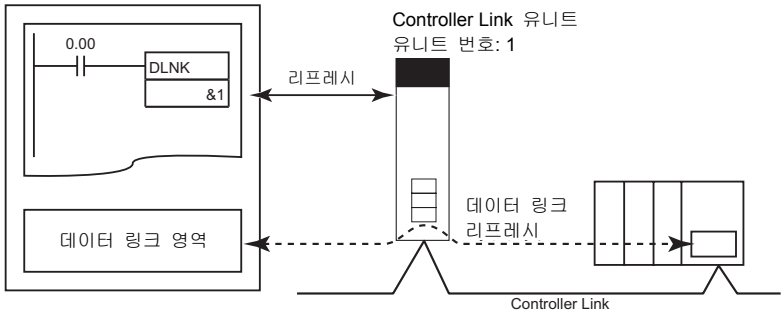
· 이 명령을 사용한 I/O 리프레시 처리 중에 I/O 버스 이상이 발생한 경우 I/O 리프레시 처리를 중지합니다.

샘플 프로그램



0.00이 ON이면 유니트 번호 1의 CPU 고기능 유니트(예: Controller Link 유니트)와의 I/O 리프레시(예: 데이터 링크의 PC 내의 리프레시)를 매회 실행합니다.

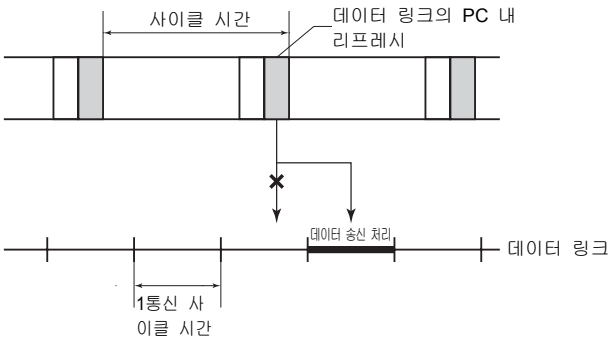
Controller Link 유니트가 데이터 갱신 중인 경우에는 ER 플래그가 ON이 되고 W0.01이 ON이 되어 재시도합니다. 정상 종료된 후 ER 플래그가 ON이 되고 재시도를 중지합니다.



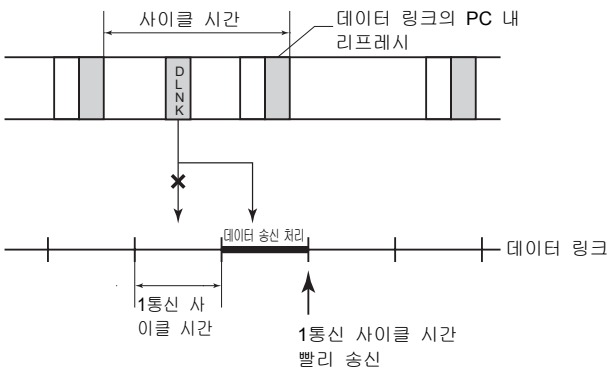
* 1: 실제로 데이터 링크 영역의 리프레시가 실행되는 타이밍은 다음과 같습니다.
  - 송신의 경우: 다음에 토큰을 취득하여 데이터를 네트워크로 송신할 때입니다(최대 1통신 사이클 시간 후에 데이터 송신).
  - 수신의 경우: 직전에 토큰을 취득하여 네트워크에서 수신한 데이터를 입력합니다(최대 1통신 사이클 시간 전의 데이터 수신).

데이터 수신 처리의 예

기존 I/O 리프레시 타이밍의 데이터 링크인 경우



DLNK 명령 실행 타이밍의 데이터 링크인 경우



SDEC

명령 이름	니모닉	동작 옵션	평션 번호	기능
7세그먼트 디코더	SDEC	@SDEC	078	채널 데이터에서 지정된 자리의 각 4비트 내용(0~F)을 8비트의 7세그먼트 데이터로 변환하여 지정된 CH 이후의 상위 또는 하위의 각 8비트에 출력합니다.

심볼	SDEC
	

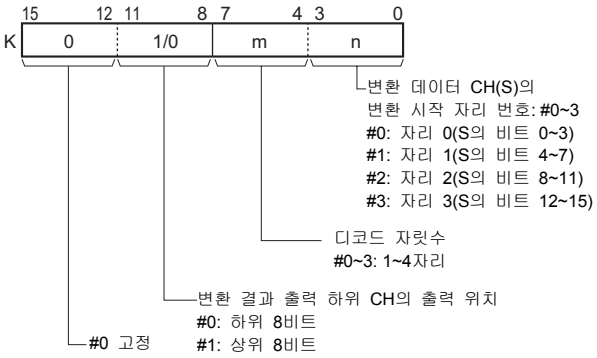
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	변환 데이터 CH 번호	UINT	1
K	자리 지정 데이터	UINT	1
D	변환 결과 출력 하위 CH 번호	UINT	가변

K: 자리 지정 데이터



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
K											○							
D											–	–						

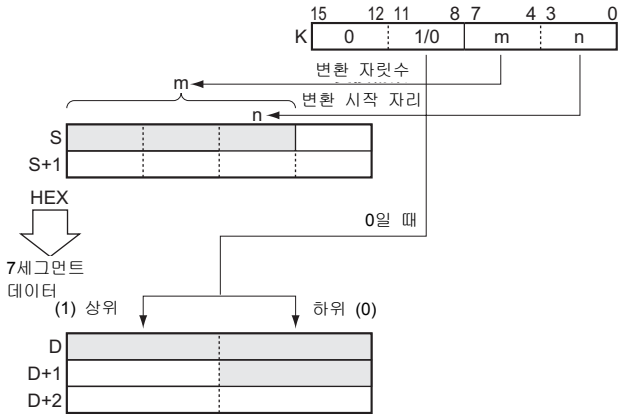
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· K의 데이터가 범위 밖일 때 ON · 그 이외일 때 OFF

SDEC

기능

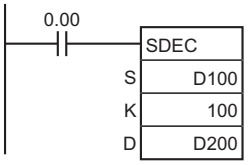
S로 지정한 데이터를 4자리의 16진수 데이터(#0000~FFFF)로 처리하고, 변환 시작 자리 번호(K)로 지정한 자리의 데이터(#0~F)를 7세그먼트 데이터로 변환하여 결과를 D의 지정한 출력 위치(K)에 출력합니다.



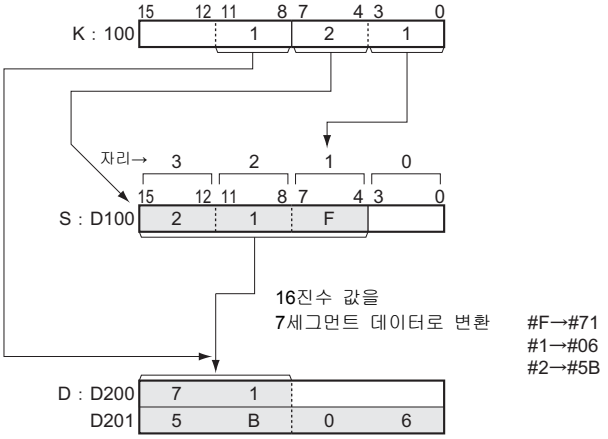
사용상의 주의 사항

- 변환 자릿수(K)에서 여러 자리의 변환을 지정한 경우 변환 대상 자리는 시작 자리에서 상위 자리 쪽의 순서가 됩니다. (자리3의 다음은 자리0으로 돌아갑니다.)
- 변환 결과는 D의 출력 위치에서 상위 CH 쪽(8비트 단위) 순서로 저장됩니다. 변환 결과 출력 CH의 데이터 중 출력 대상이 아닌 위치의 데이터는 변경되지 않습니다.

샘플 프로그램



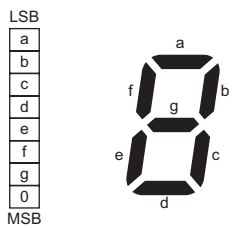
0.00이 ON이면 D100의 1CH를 4자리의 16진수(#0~F)로 간주하고 100CH (자리 지정 데이터)의 비트 0~3이 지정하는 자리 위치(예: 자리 1)에서 비트 4~7이 지정하는 자릿수를 7세그먼트 데이터로 변환하여 D200의 지정된 바이트 위치(100CH의 비트 8~11에서 지정)부터 저장합니다.



■ 변환 데이터의 내용과 디코드 출력의 관계

자리(4비트)의 내용 0~F가 8비트의 7세그먼트 데이터로 변환된 경우 다음 표와 같아집니다.

변환 원본 데이터					변환 결과 데이터(8비트)								7세그먼트 표시
숫자	비트 내용					g	f	e	d	c	b	a	16진수
0	0	0	0	0	0	0	1	1	1	1	1	1	3F
1	0	0	0	1	0	0	0	0	0	1	1	0	06
2	0	0	1	0	0	1	0	1	1	0	1	1	5B
3	0	0	1	1	0	1	0	0	1	1	1	1	4F
4	0	1	0	0	0	1	1	0	0	1	1	0	66
5	0	1	0	1	0	1	1	0	1	1	0	1	6D
6	0	1	1	0	0	1	1	1	1	1	0	1	7D
7	0	1	1	1	0	0	1	0	0	1	1	1	27
8	1	0	0	0	0	1	1	1	1	1	1	1	7F
9	1	0	0	1	0	1	1	0	1	1	1	1	6F
A	1	0	1	0	0	1	1	1	0	1	1	1	77
B	1	0	1	1	0	1	1	1	1	1	0	0	7C
C	1	1	0	0	0	0	1	1	1	0	0	1	39
D	1	1	0	1	0	1	0	1	1	1	1	0	5E
E	1	1	1	0	0	1	1	1	1	0	0	1	79
F	1	1	1	1	0	1	1	1	0	0	0	1	71



DSW

명령 이름	니모닉	동작 옵션	평션 번호	기능
디지털 스위치	DSW	–	210	입력 유니트 및 출력 유니트에 연결된 외부 디지털 스위치(또는 로터리 스위치)의 설정값을 읽어 지정된 CH 이후에 8자리 또는 4자리의 숫자(BCD 데이터)로 저장합니다.

	DSW	
심볼	DSW	
	I	I: 데이터 라인 입력(D0~3) CH 번호
	O	O: 제어 신호(CS/RD) 출력 CH 번호
	D	D: 데이터 저장 시작 CH 번호
	C1	C1: 자릿수 지정
	C2	C2: 워크 영역 시작 CH 번호

특정 영역 내의 사용 여부

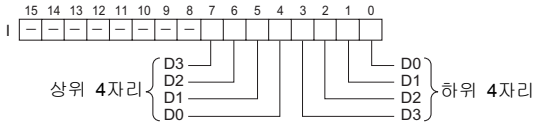
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
I	데이터 라인 입력(D0~3) CH 번호	UINT	1
O	제어 신호(CS/RD) 출력 CH 번호	UINT	1
D	데이터 저장 시작 CH 번호	WORD	가변
C1	자릿수 지정	UINT	1
C2	워크 영역 시작 CH 번호	WORD	1

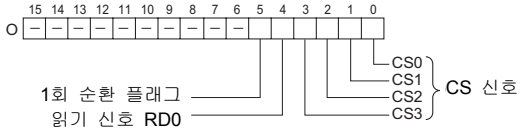
I: 데이터 라인 입력(D0~3) CH 번호

평상시에는 입력 릴레이의 입력 유니트 할당 CH 를 지정하고 이 입력 유니트에 디지털 스위치의 데이터 라인 D0~D3을 연결합니다.



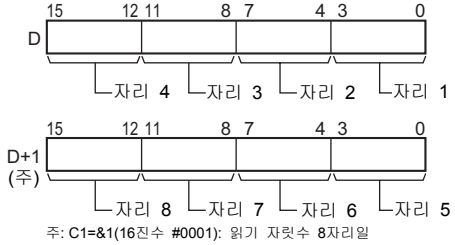
O: 제어 신호(CS/RD) 출력 CH 번호

평상시에는 출력 릴레이의 출력 유니트 할당 CH 를 지정하고 이 출력 유니트에 디지털 스위치의 제어 신호(CS 신호 및 RD 신호)를 연결합니다.



D: 데이터 저장 시작 CH 번호

외부 디지털 스위치의 설정값을 저장하는 시작 채널 번호를 지정합니다.



주: C1=&1(16진수 #0001): 읽기 자릿수 8자리일

C1: 자릿수 지정

외부 디지털 스위치에서 읽을 자릿수를 지정합니다.

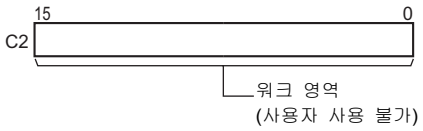
&0 또는 #0000: 4자리 읽기

&1 또는 #0001: 8자리 읽기

DSW

C2: 워크 영역 시작 CH 번호

워크 영역입니다. 이 명령은 이 1CH를 시스템적으로 사용하며, 다른 용도로는 사용할 수 없습니다.



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
I, O	○	○	○	○	○	○	○	○	○	○	–	○		○				
D												–	–		–	–	–	–
C1	–	–	–	–	–	–	–	–	–	–	○			–				
C2	○	○	○	○	○	○	○	○	○		–	○		○				

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF

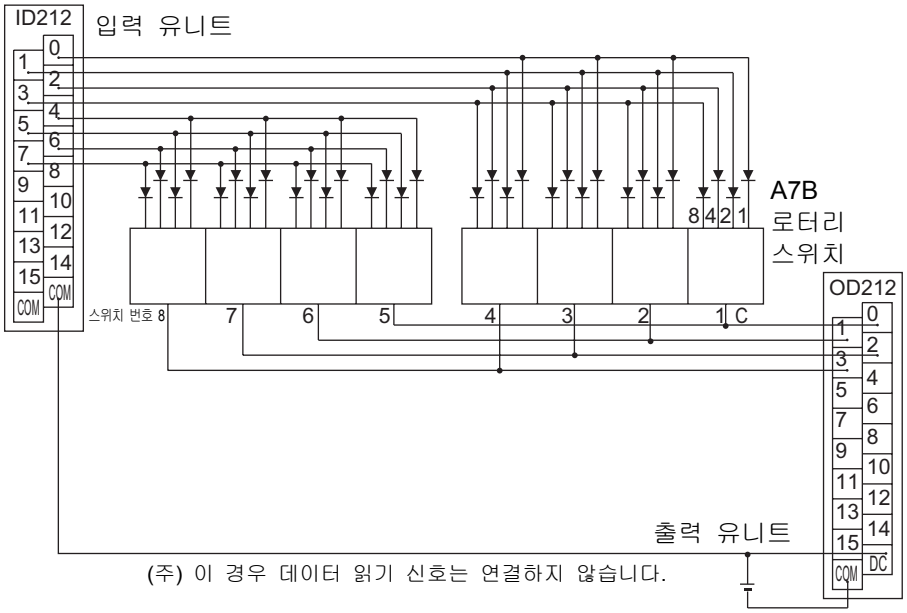
기능

O로 지정한 CH의 비트 00~04로 제어 신호를 출력하고, I로 지정한 CH 내 외부 디지털 스위치의 데이터 라인(하위 4자리 D0~3, 상위 4자리 D0~3)에서 C1로 지정한 자릿수(8자리 또는 4자리) 분량을 읽어 D 이후에 저장합니다. (4자리일 때는 DCH에 저장하고 8자리일 때는 D, D+1CH에 저장합니다.) 이 명령은 CPU 유니트의 16사이클에 1회이며 외부 디지털 스위치(또는 로터리 스위치)의 4자리 또는 8자리의 설정값을 읽습니다. 또한 16사이클 시간마다 제어 신호가 1회 순환하면 1회 순환 플래그(OCH 비트 05)가 1사이클 ON이 됩니다.

■ 외부와 연결 방법

외부의 디지털 스위치 또는 로터리 스위치를 다음과 같이 입력 유니트의 접점 0~7, 출력 유니트의 접점 0~4에 연결합니다.

예) 로터리 스위치

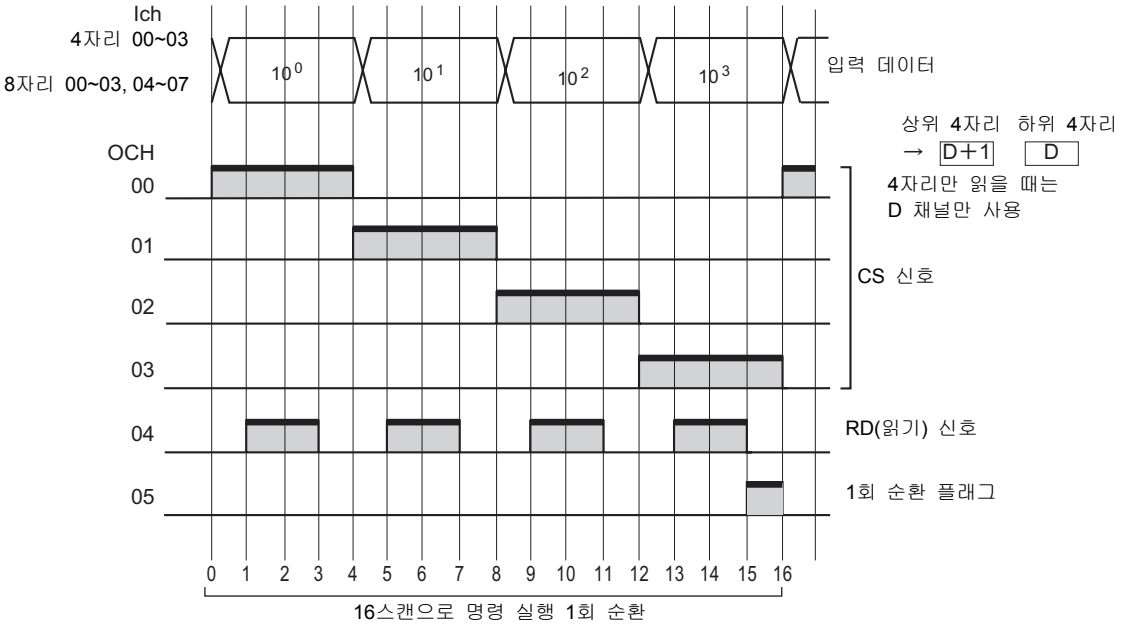


사용할 수 있는 입출력 유니트는 다음과 같습니다.

SYSBUS 원격 I/O 증설 장치 이외의 기본 I/O 유니트 또는 다점 I/O 유니트(고기능 I/O 유니트)

- 입력 유니트: 8점 이상의 DC 입력 유니트
- 출력 유니트: 8점 이상의 트랜지스터 출력 유니트

■ 타이밍 차트



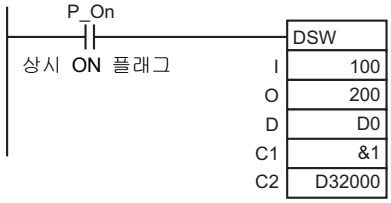
사용상의 주의 사항

- C2의 워크 영역 1CH를 다른 명령으로 읽고 쓰지 마십시오.
  다른 명령으로 읽고 쓴 경우, 이 명령이 정상적으로 동작하지 않으므로 주의하십시오.
  운전 시작 1사이클째인 경우 이 명령은 명령 실행 시 워크 영역의 1CH를 초기화하지 않습니다. 따라서 운전 시작 1사이클째부터 이 명령을 실행하는 경우는 프로그램을 통해 워크 영역을 클리어하십시오.
- 이 명령은 1회 명령 실행 후 반드시 외부 디지털 스위치(또는 로터리 스위치)를 연결한 입력 유니트/출력 유니트와의 I/O 리프레시가 실행되어야 합니다.
  따라서 사이클 시간이 짧으면 입출력 유니트의 ON/OFF 응답 시간과의 관계에서 정확하게 동작하지 않는 경우가 있습니다. 이런 경우 ON/OFF 응답 시간이 짧은 입출력 유니트를 사용하거나 고정 사이클 시간 기능 등을 통해 사이클 시간을 조정하십시오.
  또한 외부 디지털 스위치(또는 로터리 스위치)는 다음과 같은 유니트에 연결하지 마십시오.
  - SYSBUS 원격 I/O 자국 장치의 기본 I/O 유니트 및 다점 I/O 유니트(고기능 I/O 유니트)
  - 통신 슬레이브(DeviceNet 슬레이브 또는 CompoBus/S 슬레이브 등)
- 이 명령은 16사이클에서 4자리 또는 8자리의 데이터를 읽습니다. 그 후 다시 처음으로 돌아가 데이터 읽기를 수행합니다.
- 실행을 시작할 때는(명령 실행 정지 시점의 상태와 관계 없이) 항상 최초 사이클에서 읽기를 시작합니다.
- 이 명령은 C200HX/HG/HE, CQM1H 시리즈와 달리 프로그램 내에서 사용 횟수 제한이 없습니다.

샘플 프로그램

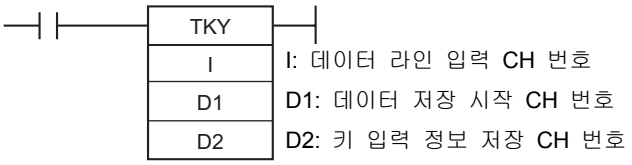
100CH(예: CS1W-ID211: 16점 DC 입력 유니트 할당)와
200CH(예: CS1W-OD211: 16점 트랜지스터 출력 유니트)에
연결된 디지털 스위치에서 8자리의 숫자를 항상 읽어 D0~D3에 저장합니다.

D32000은 워크 영역이 됩니다.



TKY

명령 이름	니모닉	동작 옵션	평션 번호	기능
10키 입력	TKY	@TKY	211	입력 유니트에 연결된 외부의 10키패드에서 숫자를 순서대로 읽어 지정된 CH 이후에 최대 8자리의 숫자(BCD 데이터)로 저장합니다.

	TKY
심볼	

특정 영역 내의 사용 여부

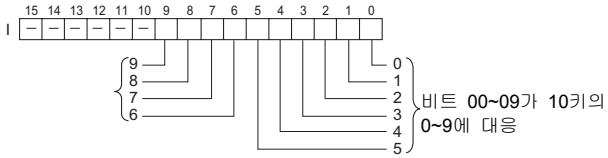
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
I	데이터 라인 입력 CH 번호	UINT	1
D1	데이터 저장 시작 CH 번호	UDINT	2
D2	키 입력 정보 저장 CH 번호	UINT	1

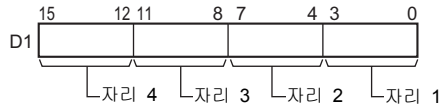
I: 데이터 라인 입력 CH 번호

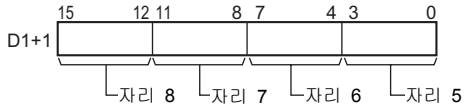
평상시에는 입력 릴레이의 입력 유니트 할당 CH를 지정하고 이 입력 유니트에 외부의 10키패드의 데이터 라인 0~9를 연결합니다.



D1: 데이터 저장 시작 CH 번호

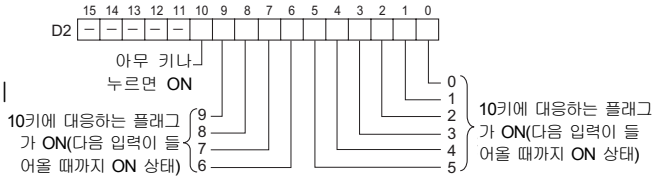
10키패드에서 최대 8자리의 숫자를 저장하는 시작 채널 번호를 지정합니다.





D2: 키 입력 정보 저장 CH 번호

10키패드의 입력 상태에 따라 대응하는 비트(비트 00~09가 10키 0~9에 대응)가 ON 또는 OFF 됩니다. 아무 키나 누르는 동안 비트 10이 ON이 됩니다.



주: 이 명령은 다른 편의 명령(예: HKY)과 달리 워크 영역을 지정할 필요가 없습니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
I	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
D1												–						
D2												○						

조건 플래그의 동작

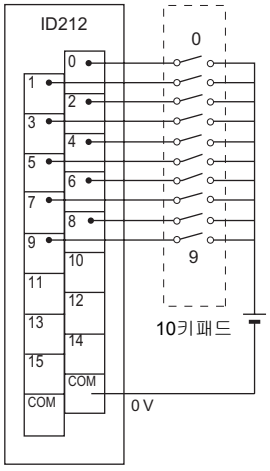
이름	라벨	내용
에러 플래그	P_ER	OFF

기능

I로 지정한 CH 내의 외부 10키패드 데이터가 변경될 때마다 그 시점의 숫자(0~9)를 순서대로 읽어 최대 8자리분을 D1(하위 4자리), D1+1(상위 4자리)에 저장합니다. 또한 읽는 동안 10키가 변경될 때마다의 상태는 D2에 저장됩니다.

■ 외부와 연결 방법

외부의 10키패드를 다음과 같이 입력 유니트의 접점 0~9에 연결합니다.



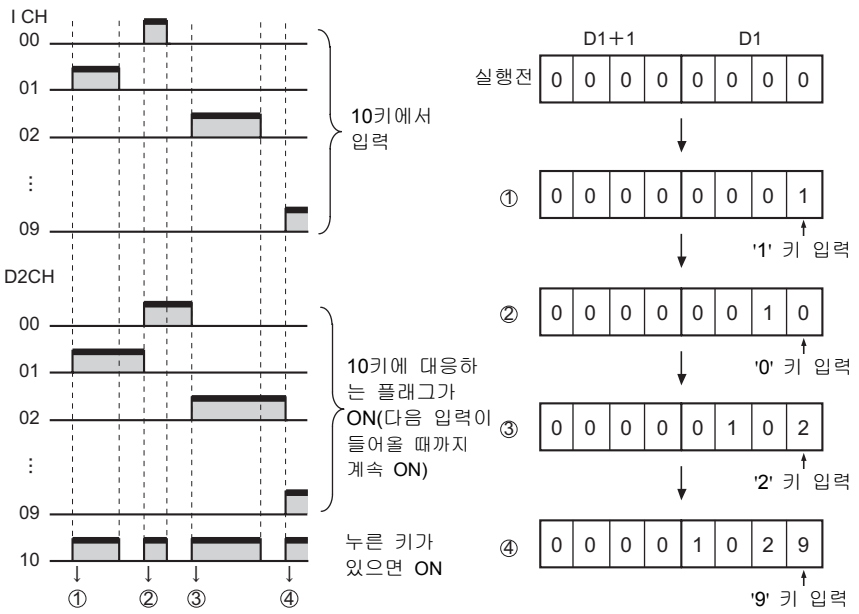
입력 유니트

사용할 수 있는 입력 유니트는 다음과 같습니다.

SYSBUS 원격 I/O 증설 장치 이외의 기본 입력 유니트 또는 다점 입력 유니트(고기능 I/O 유니트)

· 입력 유니트: 16점 이상의 DC 입력 유니트

■ 타이밍 차트



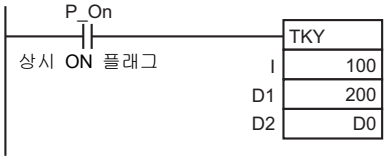
사용상의 주의 사항

- 이 명령은 1회 명령 실행 후 반드시 10키패드를 연결한 입력 유니트/출력 유니트와의 I/O 리프레시가 실행되어야 합니다.
  따라서 사이클 시간이 짧으면 입출력 유니트의 ON/OFF 응답 시간과의 관계에서 정확하게 동작하지 않는 경우가 있습니다. 이런 경우 ON/OFF 응답 시간이 짧은 입출력 유니트를 사용하거나 고정 사이클 시간 기능 등을 통해 사이클 시간을 조정하십시오.
  또한 10키패드는 다음과 같은 유니트에 연결하지 마십시오.
  - SYSBUS 원격 I/O 자국 장치의 기본 I/O 유니트 및 다점 I/O 유니트(고기능 I/O 유니트)
  - 통신 슬레이브(DeviceNet 슬레이브 또는 CompoBus/S 슬레이브 등)
- 10키패드에서 하나의 숫자를 입력하면 저장된 숫자가 1자리(4비트)씩 상위로 이동하고 최하위 자리에 마지막으로 입력한 숫자가 저장됩니다.

  10키패드에서 8자리 이상의 숫자가 입력되면 최상위 자리부터 없어집니다.
- 이 명령의 실행을 시작할 때는(명령 실행 정지 시점의 상태와 관계 없이) 항상 최초 사이클에서 읽기를 시작합니다.
- 10키패드로 하나의 키를 누르는 동안 다른 키의 입력은 수신되지 않습니다.
- 이 명령은 프로그램 내에서 사용 횟수 제한이 없습니다(C200HX/HG/HE, CQM1H 시리즈와 다름).

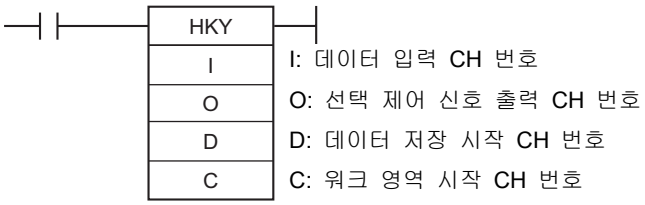
샘플 프로그램

100CH(예: CS1W-ID211: 16점 DC 입력 유니트 할당)에 연결된 10키패드에서 8자리의 숫자를 항상 읽어 200~201H에 저장합니다.



HKY

명령 이름	니모닉	동작 옵션	평션 번호	기능
16키 입력	HKY	–	212	입력 유니트 및 출력 유니트에 연결된 외부의 16키보드에서 숫자를 순서대로 읽어 지정된 CH 이후에 최대 8자리의 숫자(BCD 데이터)로 저장합니다.

	HKY
심볼	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
I	데이터 입력 CH 번호	UINT	1
O	선택 제어 신호 출력 CH 번호	UINT	1
D	데이터 저장 시작 CH 번호	WORD	3
C	워크 영역 시작 CH 번호	WORD	1

I: 데이터 입력 CH 번호

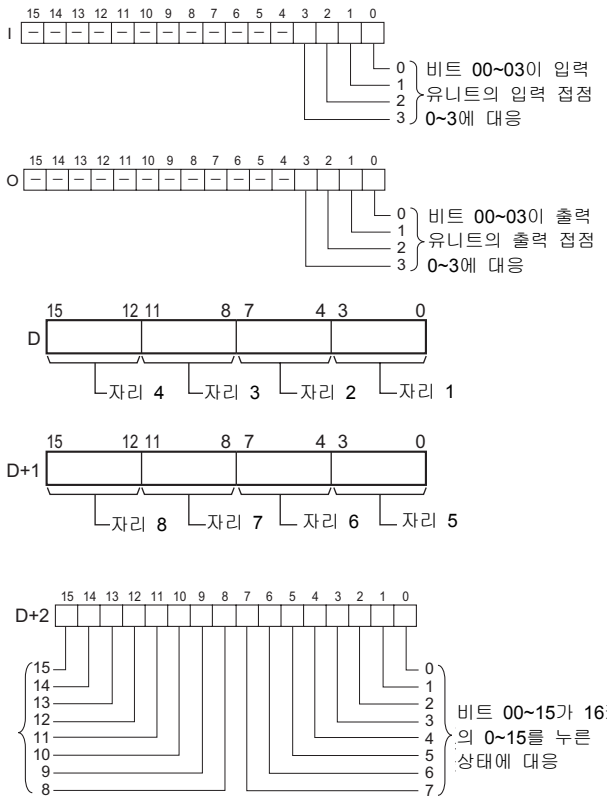
평상시에는 입력 릴레이의 입력 유니트 할당 CH를 지정하고 이 입력 유니트에 16키보드의 데이터 라인 0~3을 연결합니다.

O: 선택 제어 신호 출력 CH 번호

평상시에는 출력 릴레이의 출력 유니트 할당 CH를 지정하고 이 출력 유니트에 16키보드의 선택 제어 신호를 연결합니다.

D: 데이터 저장 시작 CH 번호

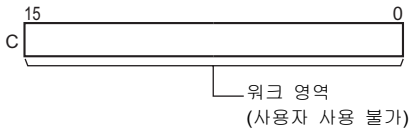
16키보드에서 입력되는 최대 8자리의 숫자를 저장하는 시작 채널 번호를 지정합니다.



HKY

C: 워크 영역 시작 CH 번호

워크 영역으로 사용하는 채널 번호를 지정합니다.
이 명령은 이 1CH를 시스템적으로 사용하며,
다른 용도로는 사용할 수 없습니다.



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
I, O	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
D												–						
C												○						

조건 플래그의 동작

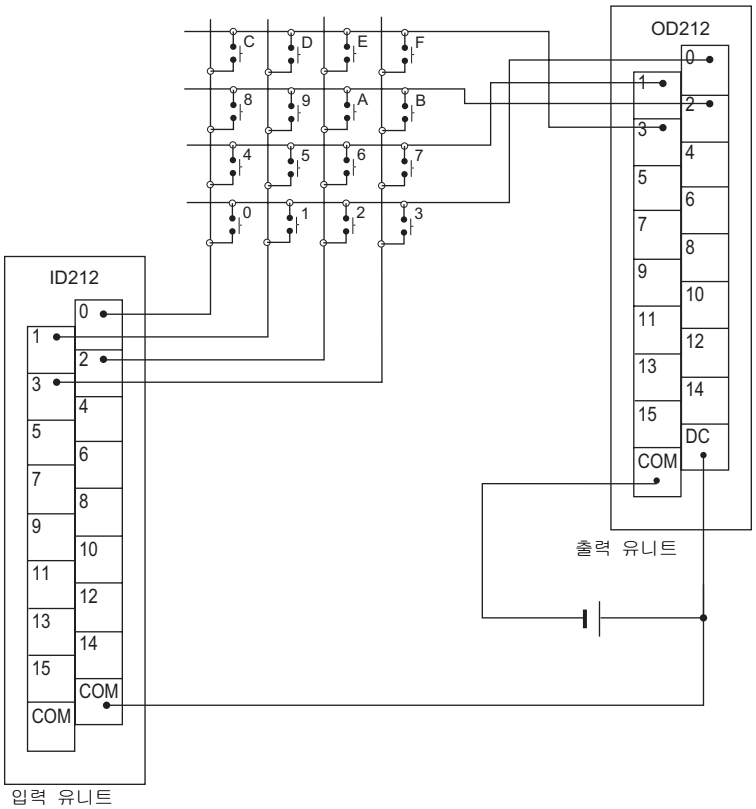
이름	라벨	내용
에러 플래그	P_ER	OFF

기능

O로 지정한 CH의 비트 00~03으로 선택 제어 신호를 출력하고, I로 지정한 CH 내의 비트 00~03에서 데이터를 순서대로 읽어 최대 8자리의 숫자로 D 이후에 저장합니다. 이 명령은 CPU 유니트의 3~12사이클에 1회, 하나의 키 상태를 읽습니다. 또한 읽는 동안 16키를 누르는 상태에 맞춰 D+2의 대응하는 비트가 ON/OFF됩니다.

■ 외부와 연결 방법

외부의 16키보드를 다음과 같이 입력 유니트의 접점 0~3, 출력 유니트의 접점 0~3에 연결합니다.

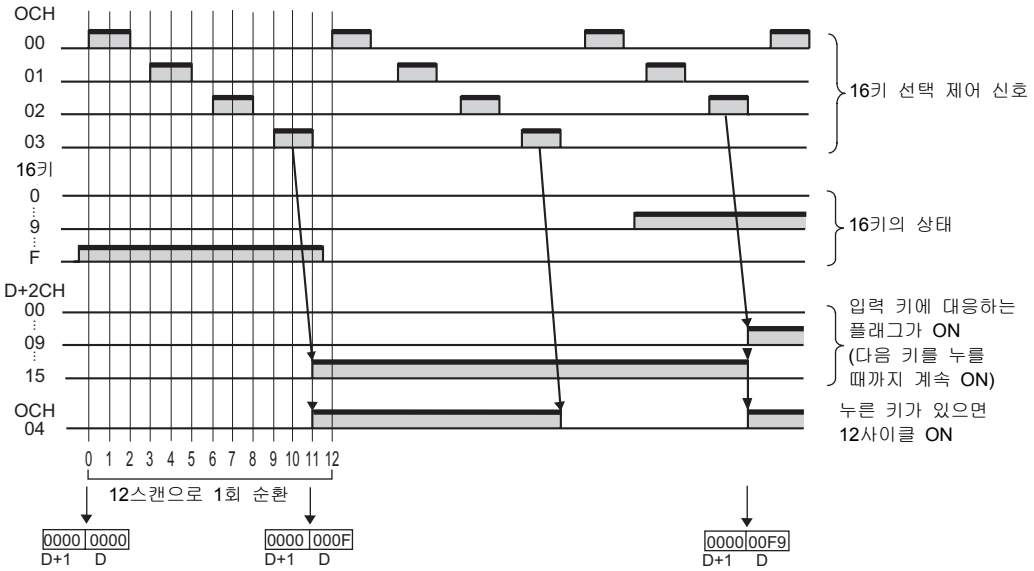


사용할 수 있는 입출력 유니트는 다음과 같습니다.

SYSBUS 원격 I/O 증설 장치 이외의 기본 I/O 유니트 또는 다점 I/O 유니트(고기능 I/O 유니트)

- 입력 유니트: 8점 이상의 DC 입력 유니트
- 출력 유니트: 8점 이상의 트랜지스터 출력 유니트

■ 타이밍 차트



사용상의 주의 사항

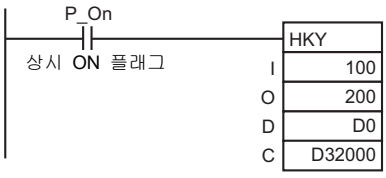
- C로 지정한 워크 영역의 1CH를 다른 명령으로 읽고 쓰지 마십시오.
  다른 명령으로 읽고 쓴 경우, 이 명령이 정상적으로 동작하지 않으므로 주의하십시오.
  운전 시작 1사이클째인 경우 이 명령은 명령 실행 시 워크 영역의 1CH를 초기화하지 않습니다. 따라서 운전 시작 1사이클째부터 이 명령을 실행하는 경우는 프로그램을 통해 워크 영역을 클리어하십시오.
- 이 명령은 1회 명령 실행 후 반드시 16키보드를 연결한 입력 유니트/출력 유니트와의 I/O 리프레시가 실행되어야 합니다.
  따라서 사이클 시간이 짧으면 입출력 유니트의 ON/OFF 응답 시간과의 관계에서 정확하게 동작하지 않는 경우가 있습니다. 이런 경우 ON/OFF 응답 시간이 짧은 입출력 유니트를 사용하거나 고정 사이클 시간 기능 등을 통해 사이클 시간을 조정하십시오.
  또한 16키보드는 다음과 같은 유니트에 연결하지 마십시오.
  - SYSBUS 원격 I/O 자국 장치의 기본 I/O 유니트 및 다점 I/O 유니트(고기능 I/O 유니트)
  - 통신 슬레이브(DeviceNet 슬레이브 또는 CompoBus/S 슬레이브 등)
- 이 명령은 어떤 선택 제어 신호 출력이 ON일 때 ON이 된 입력이 무엇인지에 따라 누른 키를 특정합니다. 따라서 누른 키를 특정할 수 있는 타이밍에 따라 3~12사이클의 어느 한 주기에 하나의 숫자(BCD 데이터 1자리)를 읽습니다. 그 후 다시 처음으로 돌아가 동일한 읽기 처리를 수행합니다.
- 이 명령의 실행을 시작할 때는(명령 실행 정지 시점의 상태와 관계 없이) 항상 최초 사이클에서 읽기를 시작합니다.
- 16키보드에서 하나의 숫자를 입력하면 저장된 숫자가 1자리(4비트)씩 상위로 이동하고 마지막으로 입력한 숫자가 최하위 자리에 저장됩니다.

  16키보드에서 8자리 이상의 숫자가 입력되면 최상위 자리부터 없어집니다.
- 16키보드로 하나의 키를 누르는 동안 다른 키의 입력은 수신되지 않습니다.
- 이 명령은 CQM1H 시리즈와 달리 프로그램 내에서 사용 횟수 제한이 없습니다.

샘플 프로그램

100CH(예: CS1W-ID211: 16점 DC 입력 유니트 할당)와 200CH(예: CS1W-OD211: 16점 트랜지스터 출력 유니트)에 연결된 16키보드에서 8자리의 숫자를 항상 읽어 D0~D1CH에 저장합니다.

또한 오른쪽 그림은 D(워크 영역)를 D32000으로 한 예입니다.



MTR

명령 이름	니모닉	동작 옵션	평션 번호	기능
행렬 입력	MTR	–	213	입력 유니트 및 출력 유니트에 연결된 외부의 8점×8열의 접점(행렬)에서 64점을 순서대로 읽어 지정된 CH 이후에 64점(4CH) 데이터로 저장합니다.

	MTR	
심볼	MTR	
	I	I: 데이터 입력 CH 번호
	O	O: 데이터 선택 신호 출력 CH 번호
	D	D: 입력 데이터 저장 시작 CH 번호
	C	C: 워크 영역 시작 CH 번호

특정 영역 내의 사용 여부

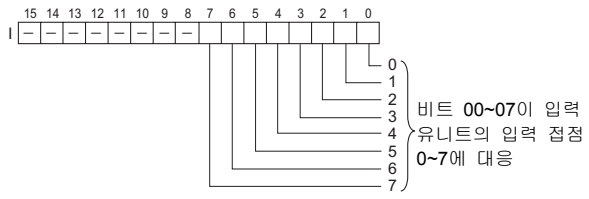
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
I	데이터 입력 CH 번호	UINT	1
O	데이터 선택 신호 출력 CH 번호	UINT	1
D	입력 데이터 저장 시작 CH 번호	ULINT	4
C	워크 영역 시작 CH 번호	WORD	1

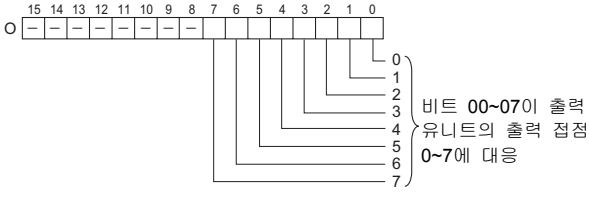
I: 데이터 입력 CH 번호

평상시에는 입력 릴레이의 입력 유니트 할당 CH를 지정하고 이 입력 유니트에 8점의 입력 신호를 연결합니다.



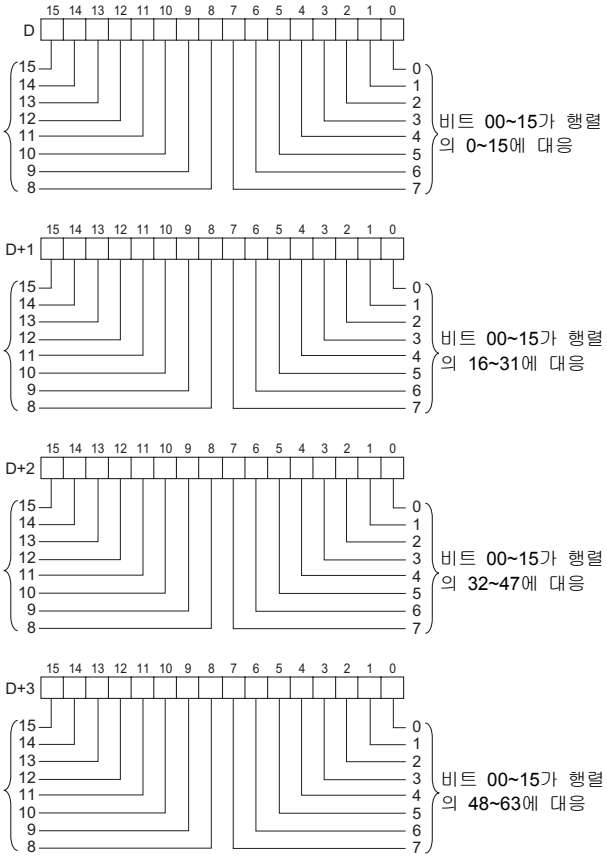
O: 데이터 선택 신호 출력 CH 번호

평상시에는 출력 릴레이의 출력 유니트 할당 CH를 지정하고 이 출력 유니트에 8점의 데이터 선택용 신호를 연결합니다.



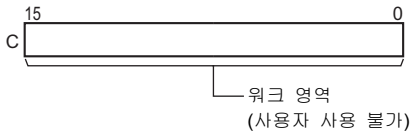
D: 입력 데이터 저장 시작 CH 번호

8점×8열의 접점(행렬)에서의 16점 데이터(4CH)를 저장하는 시작 채널 번호를 지정합니다.



C: 워크 영역 시작 CH 번호

워크 영역으로 사용하는 채널 번호를 지정합니다. 이 명령은 이 1CH를 시스템적으로 사용하며, 다른 용도로는 사용할 수 없습니다.



■ **영역 종류**

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
I, O	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
D												–						
C												○						

조건 플래그의 동작

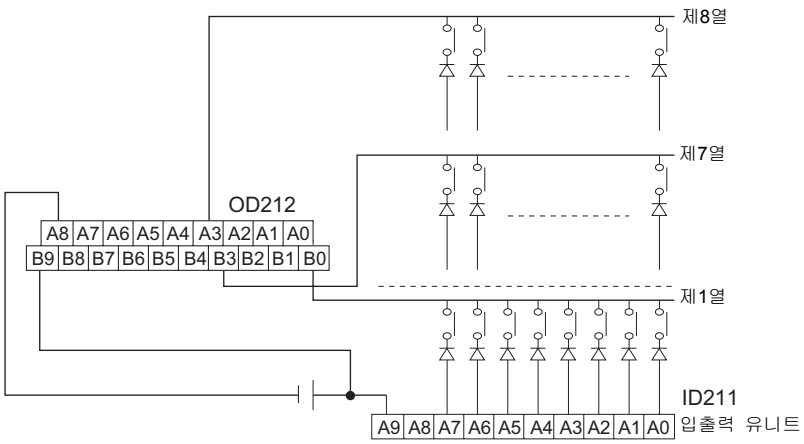
이름	라벨	내용
에러 플래그	P_ER	OFF

기능

O로 지정한 CH의 비트 00~07로 선택 신호를 출력하고, I로 지정한 CH 내의 비트 00~07에서 데이터를 순서대로 읽어 16점 데이터(4CH)로 D 이후에 저장합니다. 이 명령은 CPU 유니트의 24사이클에 1회, 하나의 행렬 상태를 읽습니다.
또한 24사이클 시간마다 데이터 선택용 신호가 1회 순환하면 키 스캔 1회 순환 플래그(OCH 비트 08)가 1사이클 ON이 됩니다.

■ 외부와 연결 방법

외부의 8점×8열의 행렬을 오른쪽 그림과 같이 입력 유니트의 접점 0~7, 출력 유니트의 접점 0~7에 연결합니다.

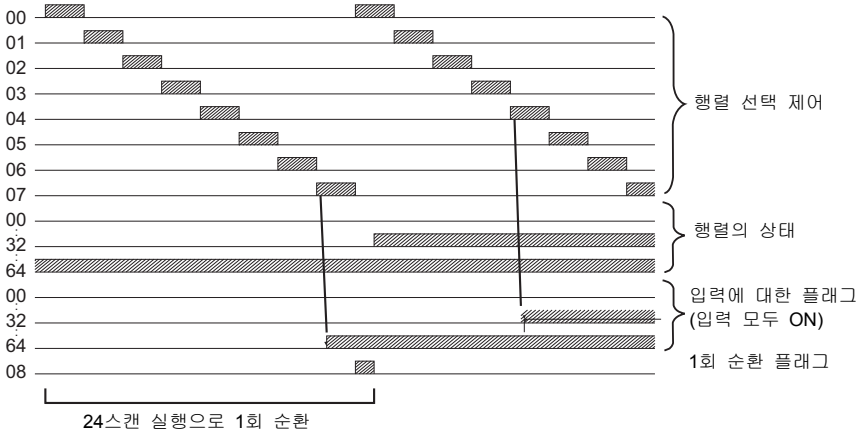


사용할 수 있는 입출력 유니트는 다음과 같습니다.

SYSBUS 원격 I/O 증설 장치 이외의 기본 I/O 유니트 또는 다점 I/O 유니트(고기능 I/O 유니트)

- 입력 유니트: 8점 이상의 DC 입력 유니트
- 출력 유니트: 8점 이상의 트랜지스터 출력 유니트

■ 타이밍 차트

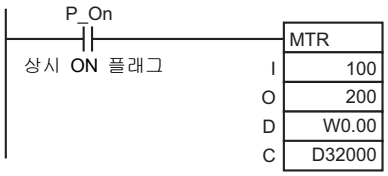


사용상의 주의 사항

- C로 지정한 워크 영역의 1CH를 다른 명령으로 읽고 쓰지 마십시오. 다른 명령으로 읽고 쓴 경우, 이 명령이 정상적으로 동작하지 않으므로 주의하십시오.
- 운전 시작 1사이클째인 경우 이 명령은 명령 실행 시 워크 영역의 1CH를 초기화하지 않습니다. 따라서 운전 시작 1사이클째부터 이 명령을 실행하는 경우는 프로그램을 통해 워크 영역을 클리어하십시오.
- 이 명령은 1회 명령 실행 후 반드시 외부 행렬을 연결한 입력 유니트/출력 유니트와의 I/O 리프레시가 실행되어야 합니다.
  따라서 사이클 시간이 짧으면 입출력 유니트의 ON/OFF 응답 시간과의 관계에서 정확하게 동작하지 않는 경우가 있습니다. 이런 경우 ON/OFF 응답 시간이 짧은 입출력 유니트를 사용하거나 고정 사이클 시간 기능 등을 통해 사이클 시간을 조정하십시오.
  또한 외부 행렬은 다음과 같은 유니트에 연결하지 마십시오.
  - SYSBUS 원격 I/O 자국 장치의 기본 I/O 유니트 및 다점 I/O 유니트(고기능 I/O 유니트)
  - 통신 슬레이브(DeviceNet 슬레이브 또는 CompoBus/S 슬레이브 등)
- 이 명령의 실행을 시작할 때는(명령 실행 정지 시점의 상태와 관계 없이) 항상 최초 사이클에서 읽기를 시작합니다.
- 이 명령은 C200HX/HG/HE, CQM1H 시리즈와 달리 프로그램 내에서 사용 횟수 제한이 없습니다.

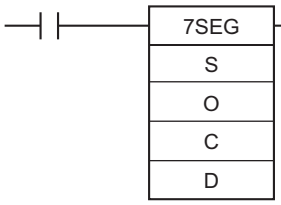
샘플 프로그램

100CH(예: CS1W-ID211: 16점 DC 입력 유니트 할당)와 200CH(예: CS1W-OD211: 16점 트랜지스터 출력 유니트)에 연결된 8점×8열의 접점(행렬)에서 64점 데이터를 항상 읽어 W0~W3CH에 저장합니다. 또한 오른쪽 그림은 D(워크 영역)를 D32000으로 한 예입니다.



7SEG

명령 이름	니모닉	동작 옵션	평션 번호	기능
7세그먼트 표시	7SEG	–	214	4자리 또는 8자리의 숫자(BCD 데이터)를 7세그먼트 표시기용 데이터로 변환하여 지정된 CH 이후에 출력합니다.

	7SEG
심볼	 S: 표시 데이터 저장 시작 CH 번호 O: 데이터 출력/래치 출력 저장 CH 번호 C: 표시 자릿수 및 출력 논리 선택 데이터 D: 워크 영역 시작 CH 번호

특정 영역 내의 사용 여부

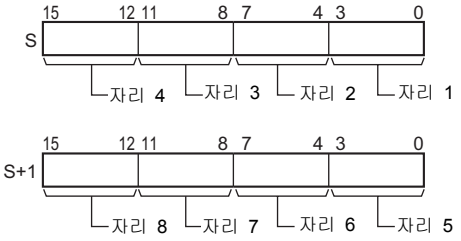
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	불가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	표시 데이터 저장 시작 CH 번호	WORD	가변
O	데이터 출력/래치 출력 저장 CH 번호	UINT	1
C	표시 자릿수 및 출력 논리 선택 데이터	#+10진수만 해당	1
D	워크 영역 시작 CH 번호	WORD	1

S: 표시 데이터 저장 시작 CH 번호

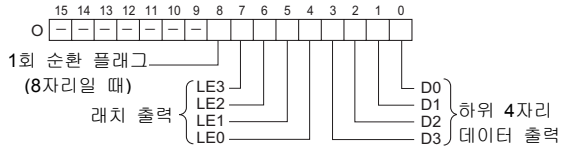
7세그먼트 표시기용 데이터를 저장하는 시작 채널 번호를 지정합니다.



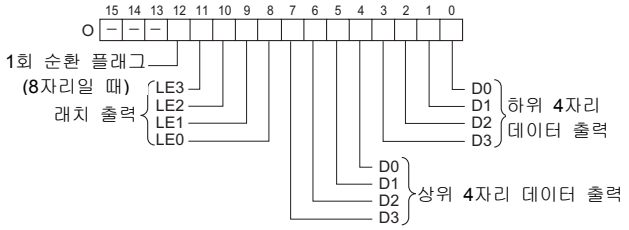
O: 데이터 출력/래치 출력 저장 CH 번호

평상시에는 출력 릴레이의 출력 유니트 할당 CH를 지정하고 이 출력 유니트에 7세그먼트 표시기를 연결합니다.

4자리 지정 시:



8자리 지정 시:



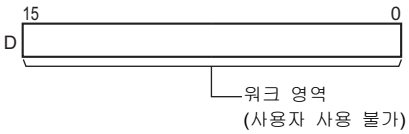
C: 표시 자릿수 및 출력 논리 선택 데이터

표시 자릿수가 4자리인지, 8자리인지 및 7세그먼트 표시기의 데이터 입력 측과 사용 출력 유니트 측의 상호 논리(트랜지스터 출력의 음의 논리(NPN)/양의 논리(PNP))가 같은지 다른지 선택합니다. 다음 #0000~0007 중 하나를 지정하십시오.

표시 자릿수	7세그먼트 표시기의 데이터 입력과 출력 유니트의 논리	7세그먼트 표시기의 래치 입력과 출력 유니트의 논리	C(표시 자릿수 및 출력 논리 선택 데이터)
4자리(4자리 1쌍)	같음	같음	#0000
		다름	#0001
	다름	같음	#0002
		다름	#0003
8자리(4자리 2쌍)	같음	같음	#0004
		다름	#0005
	다름	같음	#0006
		다름	#0007

D: 워크 영역 시작 CH 번호

워크 영역으로 사용하는 채널 번호를 지정합니다. 이 명령은 이 1CH를 시스템적으로 사용하며, 다른 용도로는 사용할 수 없습니다.



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
O												○						
C	–	–	–	–	–	–	–	–	–	–	○	–		–				
D	○	○	○	○	○	○	○	○	○	○	–	○		○				

조건 플래그의 동작

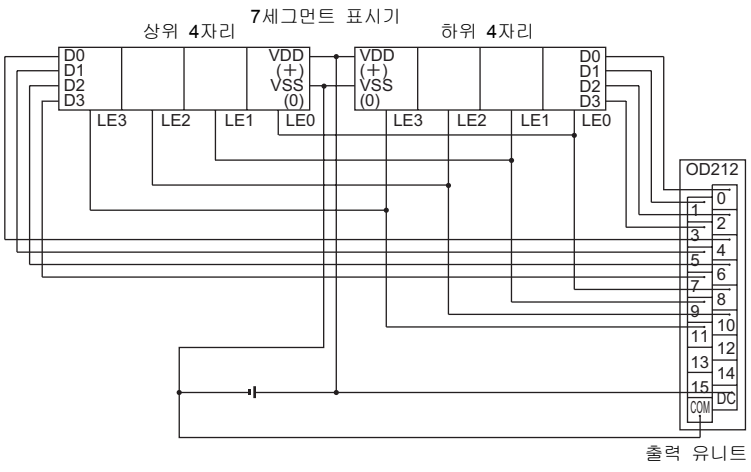
이름	라벨	내용
에러 플래그	P_ER	OFF

기능

C로 지정한 표시 자릿수 및 출력 논리를 바탕으로 S로 지정한 4자리(SCH) 또는 8자리(S, S+1CH)의 BCD 데이터를 7세그먼트 표시기용 데이터(하위 4자리 D0~3, 상위 4자리 D0~3, 래치 출력 신호 LE0~3)로 변환하여 O에 출력합니다.
이 명령은 CPU 유니트의 12사이클에 1회, 7세그먼트 표시기용 데이터를 출력합니다.
또한 12사이클 시간마다 래치 출력 신호가 1회 순환하면 1회 순환 플래그(4자리 지정 시: OCH 비트 08, 8자리 지정 시 OCH 비트 12)가 1사이클 ON이 됩니다.

■ 외부와 연결 방법

7세그먼트 표시기를 다음 그림과 같이 출력 유니트에 연결합니다. 그림은 8자리 표시일 때의 예입니다. 4자리 표시일 때는 데이터 출력(D0~D3)을 출력 접점 0~3에 연결하고 래치 출력(LE0~LE3)을 출력 접점 4~7에 연결합니다. 8자리 표시일 때는 출력 접점 12에, 4자리 표시일 때는 출력 접점 8에, 데이터 표시가 1회 순환했을 때 ON이 되는 출력이 수행되지만 연결할 필요는 없습니다.



7SEG

사용할 수 있는 출력 유니트는 다음과 같습니다.

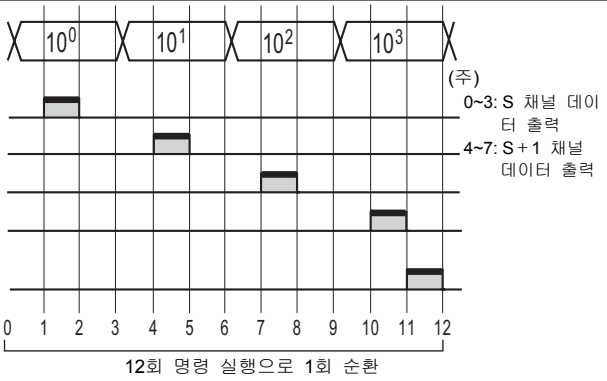
SYSBUS 원격 I/O 증설 장치 이외의 기본 출력 유니트 또는 다점 출력 유니트(고기능 I/O 유니트)

· 4자리일 때: 8점 이상의 트랜지스터 출력 유니트

· 8자리일 때: 16점 이상의 트랜지스터 출력 유니트

■ 타이밍 차트

기능 설명	대응하는 0 채널의 비트 위치		출력 상태(데이터 래치 출력의 논리는 C의 설정에 따라 달라짐)
	4자리 1쌍	4자리 2쌍	
데이터 출력	0～3	0～3 4～7	
래치 출력 0	4	8	
래치 출력 1	5	9	
래치 출력 2	6	10	
래치 출력 3	7	11	
1회 순환 플래그	8	12	



사용상의 주의 사항

· D로 지정한 워크 영역의 1CH를 다른 명령으로 읽고 쓰지 마십시오.
다른 명령으로 읽고 쓴 경우, 이 명령이 정상적으로 동작하지 않으므로 주의하십시오.
운전 시작 1사이클째인 경우 이 명령은 명령 실행 시 워크 영역의 1CH를 초기화하지 않습니다. 따라서 운전 시작 1사이클째부터 이 명령을 실행하는 경우는 프로그램을 통해 워크 영역을 클리어하십시오.

· 이 명령은 1회 명령 실행 후 반드시 7세그먼트 표시기를 연결한 출력 유니트와의 I/O 리프레시가 실행되어야 합니다. 따라서 사이클 시간이 짧으면 입출력 유니트의 ON/OFF 응답 시간과의 관계에서 정확하게 동작하지 않는 경우가 있습니다. 이런 경우 ON/OFF 응답 시간이 짧은 입출력 유니트를 사용하거나 고정 사이클 시간 기능 등을 통해 사이클 시간을 조정하십시오. 7세그먼트 표시기는 다음과 같은 유니트에 연결하지 마십시오.
  · SYSBUS 원격 I/O 자국 장치의 기본 I/O 유니트 및 다점 I/O 유니트(고기능 I/O 유니트)
  · 통신 슬레이브(DeviceNet 슬레이브 또는 CompoBus/S 슬레이브 등)

· 이 명령은 12사이클에서 4자리 또는 8자리 데이터를 출력합니다. 그 후 다시 처음으로 돌아가 데이터 출력을 수행합니다.

· 이 명령의 실행을 시작할 때는(명령 실행 정지 시점의 상태와 관계 없이) 항상 최초 사이클에서 출력을 시작 합니다.

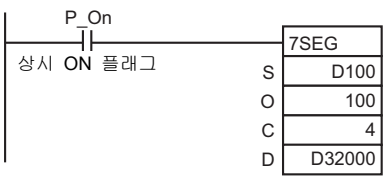
· 연결된 7세그먼트 표시기의 자릿수가 4자리 또는 8자리보다 작은 경우에도 4자리 또는 8자리분의 데이터가 출력됩니다.

샘플 프로그램

D100 내의 BCD 8자리 숫자를 100CH(예: CS1W-OD211: 16점 트랜지스터 출력 유니트 할당)에 연결된 7세그먼트 표시기에 항상 표시합니다.

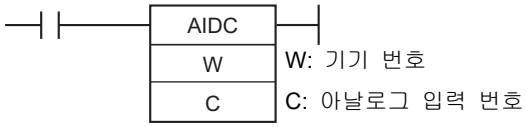
8자리(4자리 2쌍), 7세그먼트 표시기의 데이터 입력과 출력 유니트의 논리가 같고 7세그먼트 표시기의 래치 입력과 출력 유니트의 논리가 같으므로 C(표시 자릿수 및 출력 논리 선택 데이터) = #0004로 합니다.

또한 오른쪽 그림은 D(워크 영역)를 D32000으로 한 예입니다.



AIDC

명령 이름	니모닉	동작 옵션	펑션 번호	기능
아날로그 입력 직접 변환	AIDC	@AIDC	216	직접 변환 모드의 아날로그 입력 유니트 CJ1W-AD042에서 지정된 아날로그 입력 번호의 입력 변환값을 읽습니다.

심볼	AIDC
	

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
W	기기 번호	UINT	1
C	아날로그 입력 번호	UINT	1

W: 기기 번호
기기 번호 0~95를 10진수 &0~95(16진수 #0000~005F)로 지정

C: 아날로그 입력 번호
상수로 10진수 &0~4(16진수 #0000~0004)만 지정할 수 있습니다.
10진수 &0(16진수 #0000): 아날로그 입력 번호 1~사용하는 아날로그 입력 점수 분량 모두
10진수 &1(16진수 #0001): 아날로그 입력 번호 1만
10진수 &2(16진수 #0002): 아날로그 입력 번호 2만
10진수 &3(16진수 #0003): 아날로그 입력 번호 3만
10진수 &4(16진수 #0004): 아날로그 입력 번호 4만

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
W	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
C	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· W의 기기 번호가 10진수 &0~95(16진수 #0000~005F)의 범위 안이 아닐 때 ON · 지정된 유니트가 아날로그 입력 유니트 CJ1W-AD042가 아닐 때 ON · 지정된 아날로그 입력 유니트 CJ1W-AD042가 초기 처리 중 또는 재시작 중일 때 ON · W로 지정된 기기 번호의 유니트가 없을 때 ON · 이 명령을 중복 실행(사이클릭 태스크 내에서 이 명령 실행 중에 인터럽트 태스크 내에서도 이 명령을 실행)했을 때 ON · 사용하지 않는 아날로그 입력 번호(1~4)를 C에 지정했을 때 ON · 아날로그 입력 유니트 CJ1W-AD042가 직접 변환 모드가 아닐 때 ON · 아날로그 입력 유니트 CJ1W-AD042가 DM 파라미터 설정 이상 발생 중(ERC-LED 점등 중)일 때 ON · 아날로그 컨버터 이상일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 읽기가 정상적으로 종료되었을 때 ON · 읽기가 정상적으로 종료되지 않았을 때 OFF

관련 특수 보조 릴레이

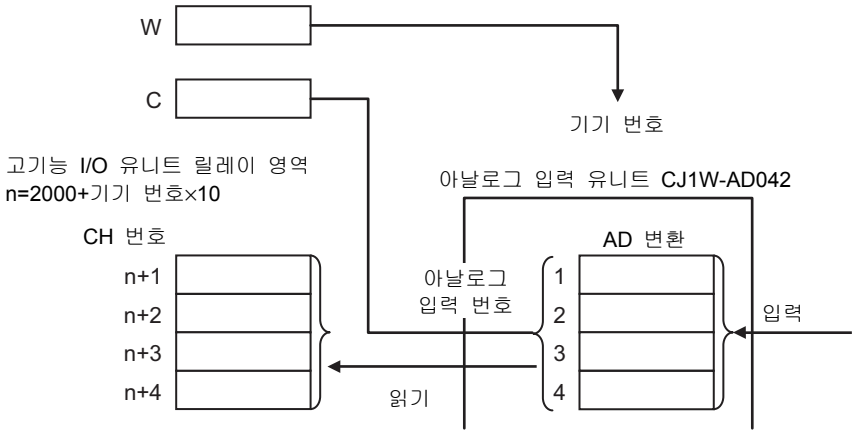
이름	주소	내용
고기능 I/O 유니트 초기화 중 플래그	A330.00~ A335.15	고기능 I/O 유니트 초기(초기화) 처리 중에 1(ON)이 됩니다. 0: 초기 처리 중이 아님 1: 초기 처리 중

기능

W의 기기 번호로 지정한 아날로그 입력 유니트 CJ1W-AD042에 대해 C로 지정한 아날로그 입력 번호의 AD 변환을 실행시키고, 그 변환 결과를 즉시 읽어 고기능 I/O 유니트의 할당 릴레이 영역에 저장합니다.

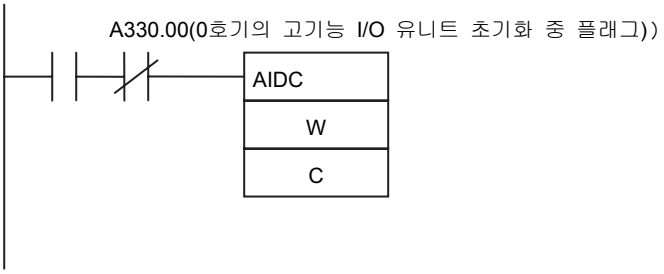
읽은 값의 데이터 크기는 아날로그 입력 번호 0의 경우 4CH, 각 아날로그 입력 번호(1~4)의 경우 1CH입니다. 아날로그 입력 번호 0의 경우 사용하는 아날로그 입력 점수 이후의 CH에는 0이 저장됩니다.

이때 지정하는 아날로그 입력 유니트 CJ1W-AD042는 미리 직접 변환 모드로 설정되어 있어야 합니다.
또한 직접 변환 모드나 아날로그 입력 점수에 대해서는 'CJ 시리즈 아날로그 입출력 유니트 사용 설명서'(번호 SBCC-845)를 참조하십시오.



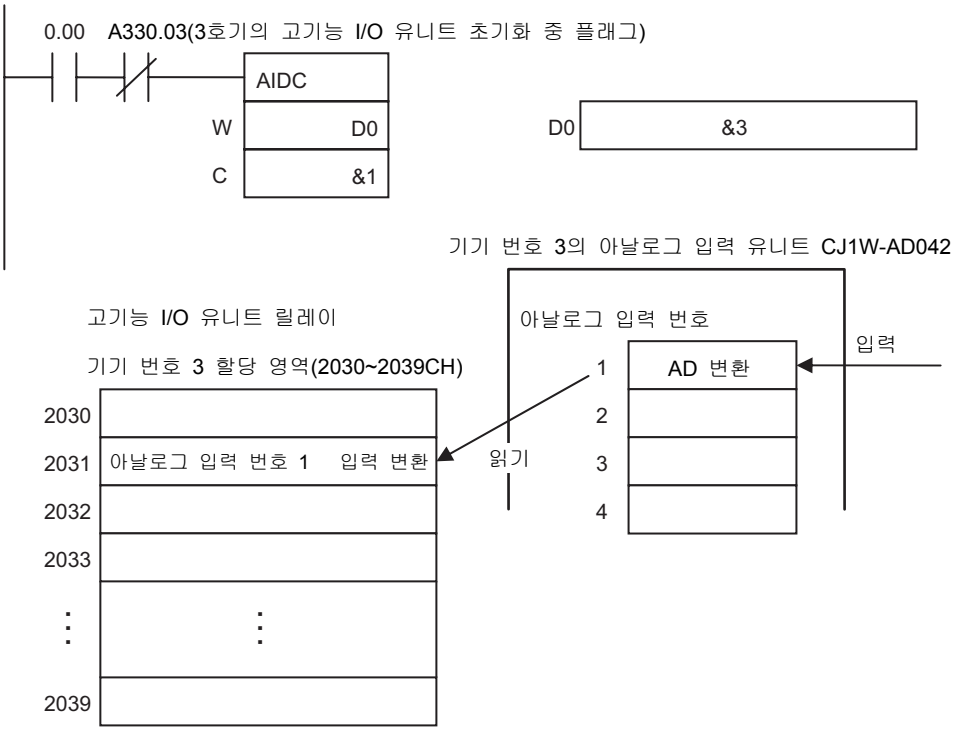
사용상의 주의 사항

- 이 명령은 반드시 아날로그 입력 유니트 CJ1W-AD042에 대해 사용하십시오. 그 이외의 고기능 I/O 유니트에 대해 이 명령을 사용하면 이 명령의 실행 시간이 약 1ms가 되거나 유니트 동작이 정지(ERH LED 점등)하는 경우가 있습니다.
- 이 명령을 실행할 때 아날로그 입력 유니트 CJ1W-AD042는 직접 변환 모드로 설정해 두십시오. 직접 변환 모드가 아니면 이 명령이 실행되지 않고 에러 플래그가 ON이 됩니다.
- 이 명령을 사이클릭 태스크와 인터럽트 태스크 양쪽에서 사용하지 마십시오. 이 명령을 중복 실행한 경우 인터럽트 태스크 중인 이 명령이 실행되지 않고 에러 플래그가 ON이 될 수 있습니다.
- 지정한 아날로그 입력 유니트 CJ1W-AD042가 초기(초기화) 처리 중에 이 명령을 사용하면 이 명령이 실행되지 않고 에러 플래그가 ON이 됩니다. 따라서 이 명령의 실행 조건에는 다음 그림과 같이 고기능 I/O 유니트 초기화 중 플래그를 b 접점으로 삽입하십시오.



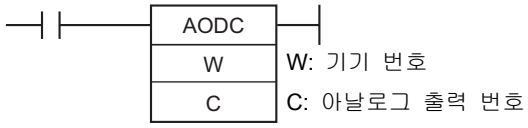
샘플 프로그램

접점 0.00이 ON이면 기기 번호 3의 아날로그 입력 유니트 CJ1W-AD042의 아날로그 입력 번호 1의 입력 변환값을 기기 번호 3의 고기능 I/O 유니트의 할당 릴레이 영역 2031CH에 저장합니다.



AODC

명령 이름	니모닉	동작 옵션	평션 번호	기능
아날로그 출력 직접 변환	AODC	@AODC	217	직접 변환 모드의 아날로그 출력 유니트 CJ1W-DA042V에 대해 지정된 아날로그 출력 번호의 출력 설정값을 출력합니다.

심볼	AODC
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
W	기기 번호	UINT	1
C	아날로그 출력 번호	UINT	1

W: 기기 번호
기기 번호 0~95를 10진수 &0~95(16진수 #0000~005F)로 지정

C: 아날로그 출력 번호
상수로 10진수 &0~4(16진수 #0000~0004)만 지정할 수 있습니다.
10진수 &0(16진수 #0000): 아날로그 출력 번호 1~사용하는 아날로그 출력 점수 분량 모두
10진수 &1(16진수 #0001): 아날로그 출력 번호 1만
10진수 &2(16진수 #0002): 아날로그 출력 번호 2만
10진수 &3(16진수 #0003): 아날로그 출력 번호 3만
10진수 &4(16진수 #0004): 아날로그 출력 번호 4만

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
W	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
C	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–

조건 플래그의 동작

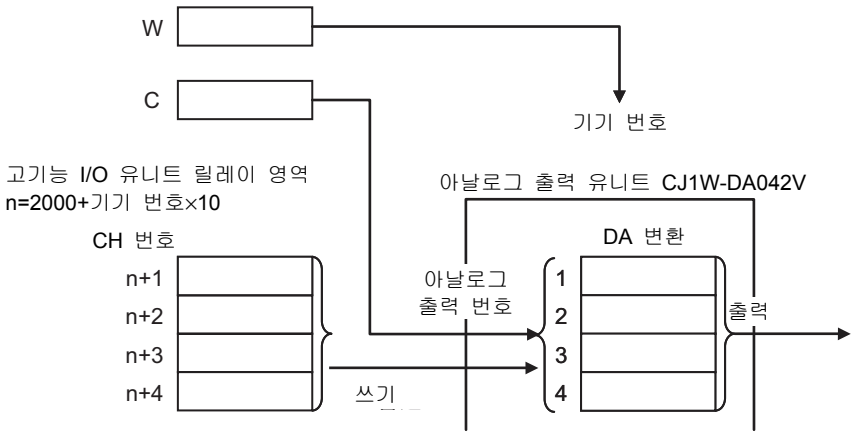
이름	라벨	내용
에러 플래그	P_ER	· W의 기기 번호가 10진수 &0~95(16진수 #0000~005F)의 범위 안이 아닐 때 ON · 지정된 유니트가 아날로그 출력 유니트 CJ1W-DA042V가 아닐 때 ON · 지정된 아날로그 출력 유니트 CJ1W-DA042V가 초기 처리 중 또는 재시작 중일 때 ON · W로 지정된 기기 번호의 유니트가 없을 때 ON · 이 명령을 중복 실행(사이클릭 태스크 내에서 이 명령 실행 중에 인터럽트 태스크 내에서도 이 명령을 실행)했을 때 ON · 사용하지 않는 아날로그 출력 번호(1~4)를 C에 지정했을 때 ON · 아날로그 출력 유니트 CJ1W-DA042V가 직접 변환 모드가 아닐 때 ON · 지정된 아날로그 출력 번호에 해당하는 할당 릴레이 영역의 출력 설정값이 범위 밖의 값일 때 ON. 또한 사용하는 아날로그 출력 번호 모두에서 출력하는 경우, 어느 하나의 출력 설정값이라도 범위 밖의 값일 때 ON · 아날로그 출력 유니트 CJ1W-DA042V가 DM 파라미터 설정 이상 발생 중(ERC-LED 점등 중)일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 출력이 정상적으로 종료되었을 때 ON · 출력이 정상적으로 종료되지 않았을 때 OFF

관련 특수 보조 릴레이

이름	주소	내용
고기능 I/O 유니트 초기화 중 플래그	A330.00~ A335.15	고기능 I/O 유니트 초기(초기화) 처리 중에 1(ON)이 됩니다. 0: 초기 처리 중이 아님 1: 초기 처리 중

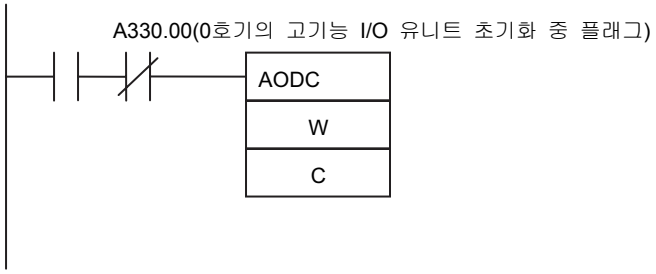
기능

W의 기기 번호로 지정한 아날로그 출력 유니트 CJ1W-DA042V에 대해 해당하는 고기능 I/O 유니트 릴레이 영역의 출력 설정값의 DA 변환을 실행시키고, 그 변환값을 즉시 C로 지정한 아날로그 출력 번호에서 출력합니다.
출력 값의 데이터 크기는 아날로그 출력 번호 0의 경우 4CH, 각 아날로그 출력 번호(1~4)의 경우 1CH입니다.
이때 지정하는 아날로그 출력 유니트 CJ1W-DA042V는 미리 직접 변환 모드로 설정되어 있어야 합니다.
또한 직접 변환 모드나 아날로그 출력 점수에 대해서는 'CJ 시리즈 아날로그 입출력 유니트 사용 설명서'(번호 SBCC-845)를 참조하십시오.



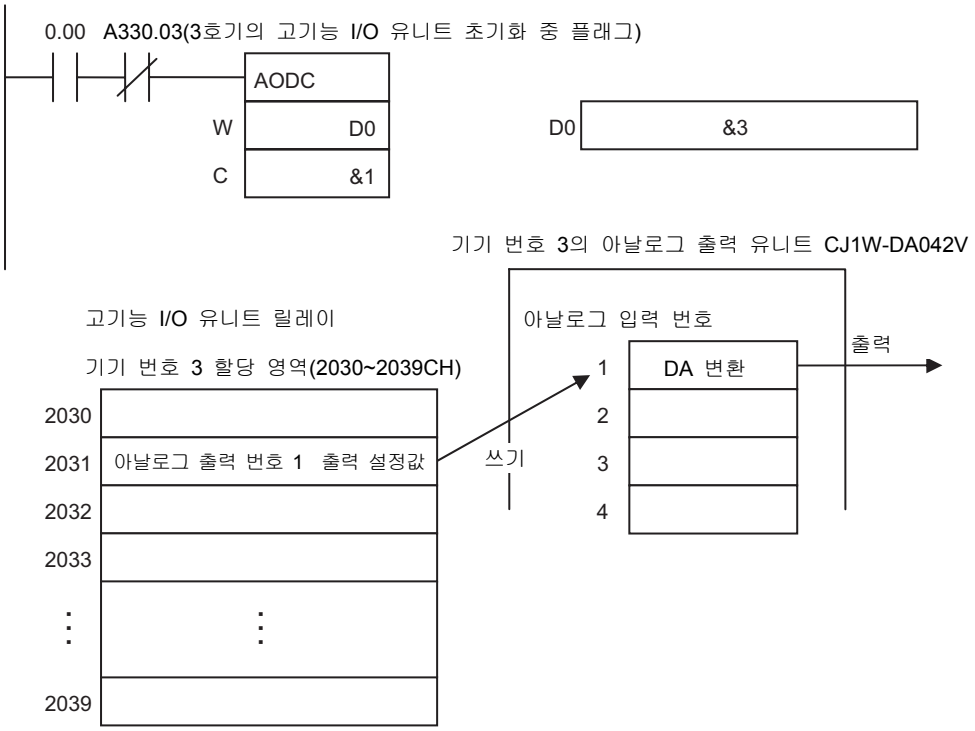
사용상의 주의 사항

- 이 명령은 반드시 아날로그 출력 유니트 CJ1W-DA042V에 대해 사용하십시오. 그 이외의 고기능 I/O 유니트에 대해 이 명령을 사용하면 이 명령의 실행 시간이 약 1ms가 되거나 유니트 동작이 정지(ERH LED 점등)하는 경우가 있습니다.
- 이 명령을 실행할 때 아날로그 출력 유니트 CJ1W-DA042V는 직접 변환 모드로 설정해 두십시오. 직접 변환 모드가 아니면 이 명령이 실행되지 않고 에러 플래그가 ON이 됩니다.
- 이 명령을 사이클릭 태스크와 인터럽트 태스크 양쪽에서 사용하지 마십시오. 이 명령을 중복 실행한 경우 인터럽트 태스크 중인 이 명령이 실행되지 않고 에러 플래그가 ON이 될 수 있습니다.
- 지정된 아날로그 출력 유니트 CJ1W-DA042V가 초기(초기화) 처리 중일 때는 이 명령이 동작하지 않습니다. 따라서 이 명령의 실행 조건에는 다음 그림과 같이 고기능 I/O 유니트 초기화 중 플래그를 b 접점으로 삽입하십시오.



샘플 프로그램

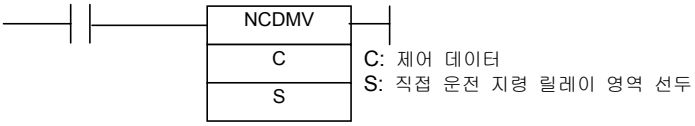
접점 0.00이 ON일 때 기기 번호 3의 아날로그 출력 유니트 CJ1W-DA042V의 아날로그 출력 번호 1에서 기기 번호 3의 고기능 I/O 유니트의 할당 릴레이 영역 2031CH에 저장된 출력 설정값을 출력합니다.



NCDMV

명령 이름	니모닉	동작 옵션	평션 번호	기능
NC 유니트 위치 결정 고속 기동	NCDMV	@NCDMV	218	위치 제어 유니트(CJ1W-NC□□4, CJ1W-NC□81)의 지정된 축 번호에 대해 상대 이동 또는 절대 이동의 고속 PTP 기동을 합니다.

심볼	NCDMV



특정 영역 내의 사용 여부

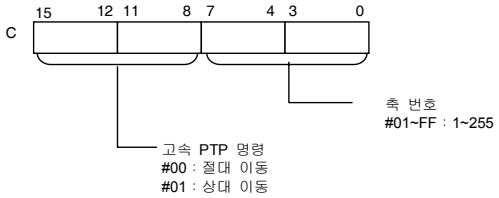
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	DWORD	2
S	직접 운전 지령 릴레이 영역 시작 CH 번호	WORD	가변

C: 제어 데이터

C : 고속 PTP 명령+축 번호



C+1 : 대상 위치 제어 유니트
- CJ1W-NC□□4(고기능 I/O 유니트)의 경우
  16진수 #0000~005F(기기 번호 0~95)
- CJ1W-NC□81(CPU 고기능 유니트)의 경우
  16진수 #8000~800F(유니트 번호 0~F)

S: 직접 운전 지령 릴레이 영역 시작 주소
위치 제어 유니트의 공통 파라미터로 설정한 직접 운전 지령 릴레이 영역 시작 주소를 지정합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
C	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
S	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C+1의 기기 번호 또는 유니트 번호가 범위 밖일 때 ON · C+1로 지정된 유니트가 위치 제어 유니트 (CJ1W-NC214/234/414/434, CJ1W-NC281/481/881)가 아닐 때 ON · C+1로 지정된 기기 번호(또는 유니트 번호)의 유니트가 없을 때 ON · C의 축 번호가 00일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 위치 제어 유니트 측에서 기동이 정상적으로 종료되었을 때 ON · 지정된 위치 제어 유니트에서 C의 고속 PTP 명령이 잘못되었을 때 OFF · 지정된 위치 제어 유니트에서 C의 축 번호가 범위 안에 없을 때 OFF · 지정된 위치 제어 유니트에서 C의 축 번호가 무효 축일 때 OFF · 위치 제어 유니트가 준비되지 않은 상태에서 이 명령이 실행되었을 때 OFF

관련 특수 보조 릴레이

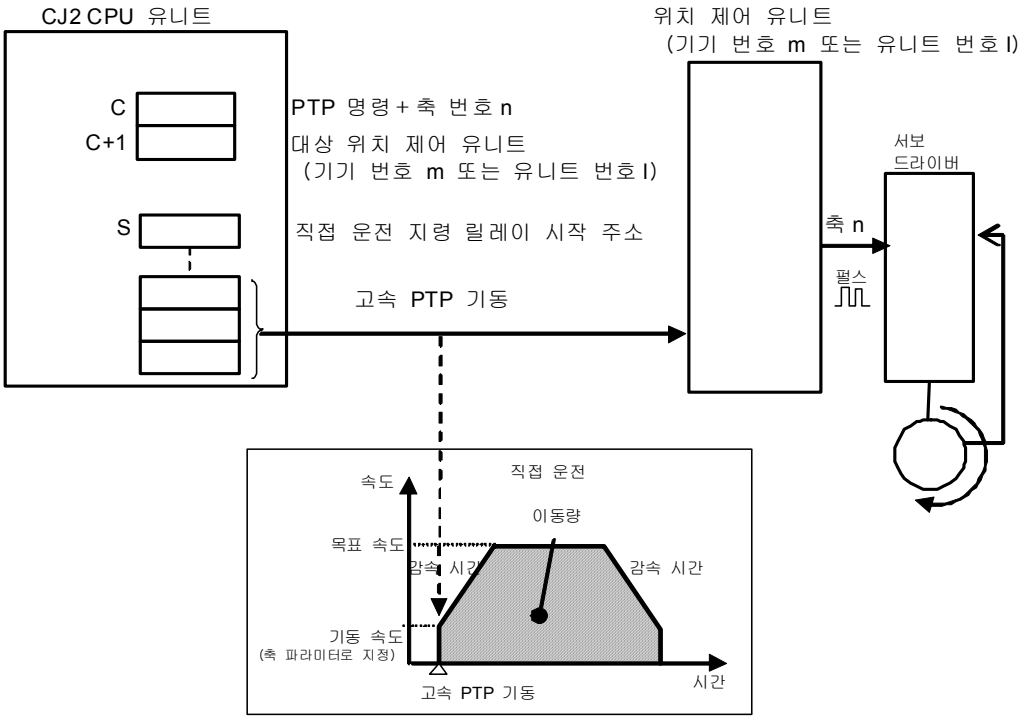
이름	주소	내용
고기능 I/O 유니트 초기화 중 플래그	A330.00~ A335.15	고기능 I/O 유니트 초기(초기화) 처리 중에 1(ON)이 됩니다. 0: 초기 처리 중이 아님 1: 초기 처리 중
CPU 고기능 유니트 초기화 중 플래그	A302.00~ A302.15	CPU 고기능 유니트가 초기(초기화) 처리 중에 1(ON)이 됩니다. 0: 초기 처리 중이 아님 1: 초기 처리 중

기능

C+1로 지정한 위치 제어 유니트의, C로 지정한 축 번호에 대해 상대 이동 또는 절대 이동으로 고속 PTP 기동을 합니다.

이 명령 하나로 기동할 수 있는 축 수는 1축뿐입니다.

위치 제어 유니트 측에서 정상으로 기동되었다는 것은 조건 플래그의 = 플래그 ON을 통해 확인합니다.



대상 위치 제어 유니트는 다음과 같습니다.

CJ1W-NC214/234/414/434(고기능 I/O 유니트)
CJ1W-NC281/481/881(CPU 고기능 유니트)

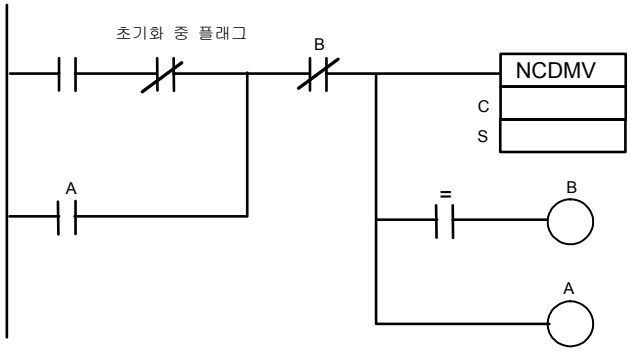
직접 운전의 자세한 사용 방법은 "SYSMAC CJ 시리즈 위치 제어 유니트 사용 설명서"(SBCE-323)의 '6장 직접 운전'을 참조하십시오.

사용상의 주의 사항

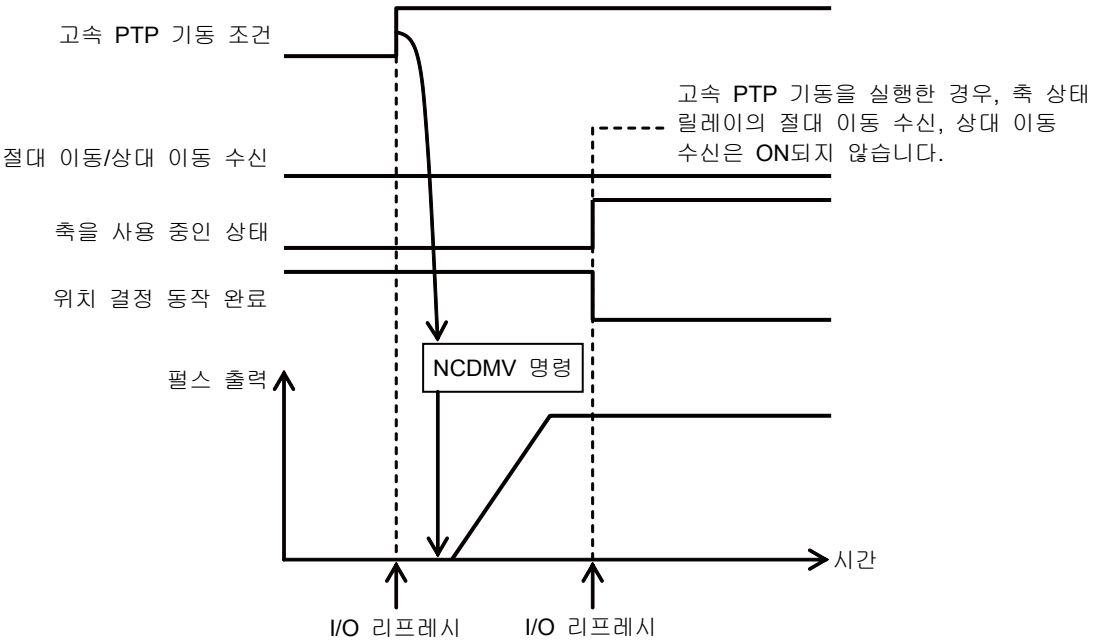
- 위치 제어 유니트는 1제어 주기에서 이 명령을 1명령만 실행할 수 있습니다.
  이로 인해 고속 PTP 기동을 연속 실행한 경우 등, 타이밍에 따라 위치 제어 유니트가 명령을 수신하지 않고 종료(= 플래그가 OFF)되는 경우가 있습니다.
  따라서 조건 플래그의 = 플래그가 ON이 될 때까지 이 명령을 사이클마다 실행하도록 다음 그림과 같은 자체 유지 회로를 반드시 구성하십시오.

NCDMV

- 지정된 위치 제어 유니트가 초기(초기화) 처리 중일 때는 이 명령이 동작하지 않습니다. 따라서 이 명령의 실행 조건에는 다음 그림과 같이 특수 보조 릴레이의 초기화 중 플래그(고기능 I/O 유니트 초기화 중 플래그 A330.00~A335.15, CPU 고기능 유니트 초기화 중 플래그 A302.00~A302.15)를 b 접점으로 삽입하십시오.



- 고속 PTP 기동을 사용하는 경우 펄스 출력은 이 명령의 실행 시점에서 시작됩니다. 단, 축을 사용 중인 상태 등의 축 상태 변화는 그 직후인 I/O 리프레시 시점에 반영됩니다.

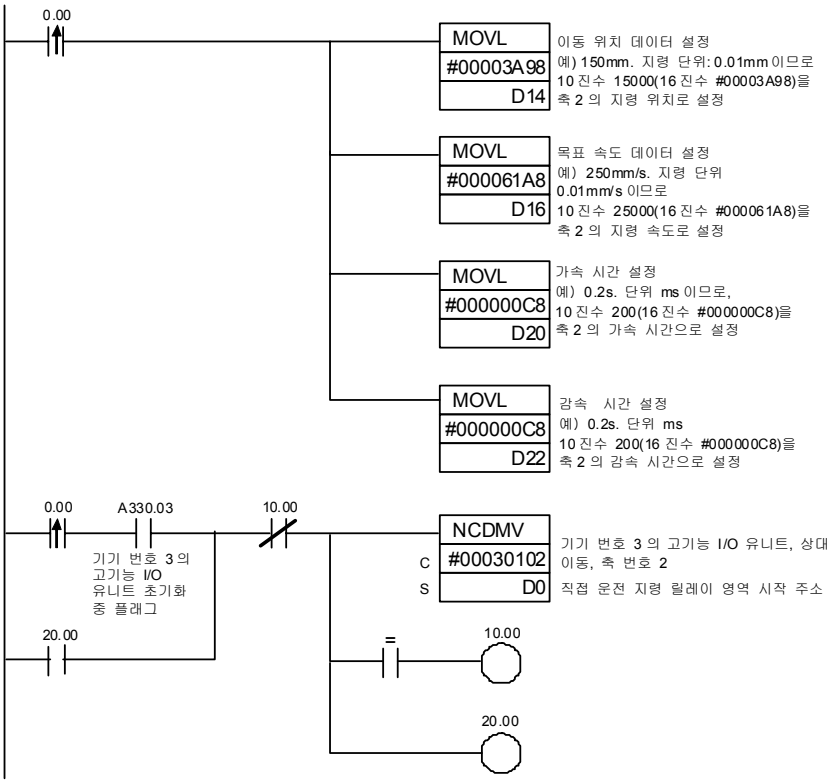


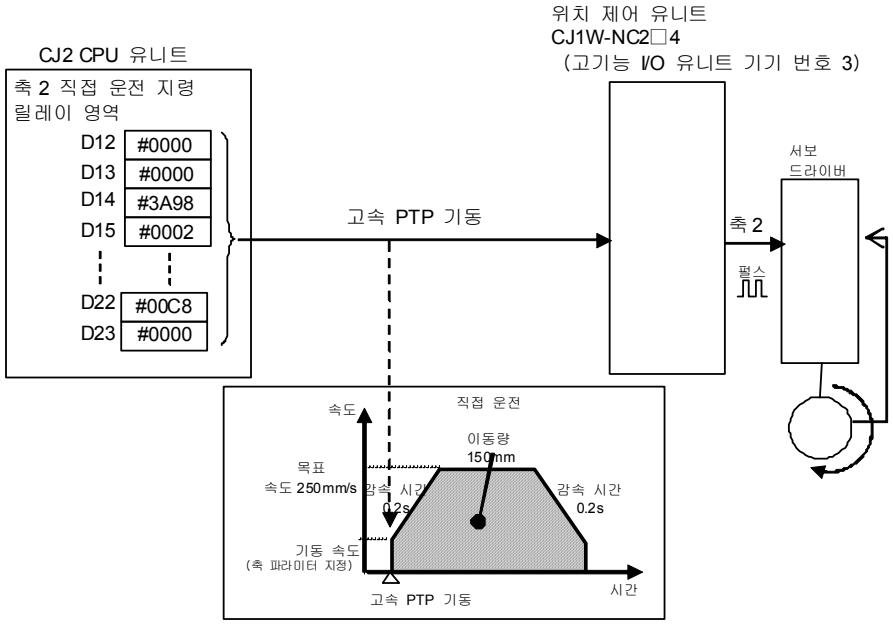
- 인터럽트 태스크 중에 이 명령을 사용할 수 있습니다. (이를 통해 인터럽트에 의한 고속 PTP 기동을 구현할 수 있습니다.)
  단, 이 경우, 사이클릭 태스크에서 동일한 유니트에 대해 이 명령(및 IORD/IOWR 명령)을 실행하지 마십시오. 사이클릭 태스크에서 이 명령을 실행하는 중에 인터럽트 태스크에서도 중복 실행하려고 하면 다중 리프레시 이상(운전 계속 이상)이 발생합니다(A402.13이 ON).

샘플 프로그램

접점 0.00을 OFF→ON으로 하면 기기 번호 3의 위치 제어 유니트 CJ1W-NC2□4(고기능 I/O 유니트)의 축 번호 2에 대해 직접 운전의 상대 이동을 실행합니다.

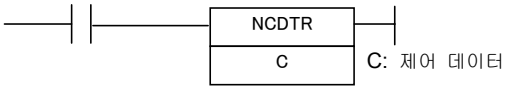
직접 운전 지령 릴레이 영역 시작 채널이 D0일 때 지령 릴레이 영역 시작 채널 D12~D23의 '직접 운전 지령 설정'을 바탕으로 고속 PTP 기동을 시작합니다.





NCDTR

명령 이름	니모닉	동작 옵션	평션 번호	기능
NC 유니트 위치 결정 시작 트리거	NCDTR	@NCDTR	219	위치 제어 유니트(CJ1W-NC□81)의 메모리 운전 기능에서 시퀀스의 시작 조건이 이 명령에서 지령 대기가 된 경우 해당 시퀀스를 시작합니다.

심볼	NCDTR
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	DWORD	2

C: 제어 데이터

C : 위치 제어 유니트 내 태스크 번호
위치 제어 유니트의 메모리 운전 파라미터로 설정한 태스크를 10진수 &1~255 또는 16진수 #0001~00FF로 지정합니다.

C+1 : 대상 위치 제어 유니트
16진수 #8000~800F(유니트 번호 0~F)로 지정합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
C	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C+1의 유니트 번호가 범위 밖일 때 ON · C+1로 지정된 유니트가 위치 제어 유니트 (CJ1W-NC281/481/881)가 아닐 때 ON · C+1로 지정된 유니트 번호의 유니트가 없을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 위치 제어 유니트 측에서 기동이 정상적으로 종료되었을 때 ON · 지정된 위치 제어 유니트에서 C의 태스크 번호가 범위 안에 없을 때 OFF · 지정된 위치 제어 유니트에서 해당 태스크가 명령에 의한 시작 대기가 아닐 때 OFF · 위치 제어 유니트가 준비되지 않은 상태에서 이 명령이 실행되었을 때 OFF

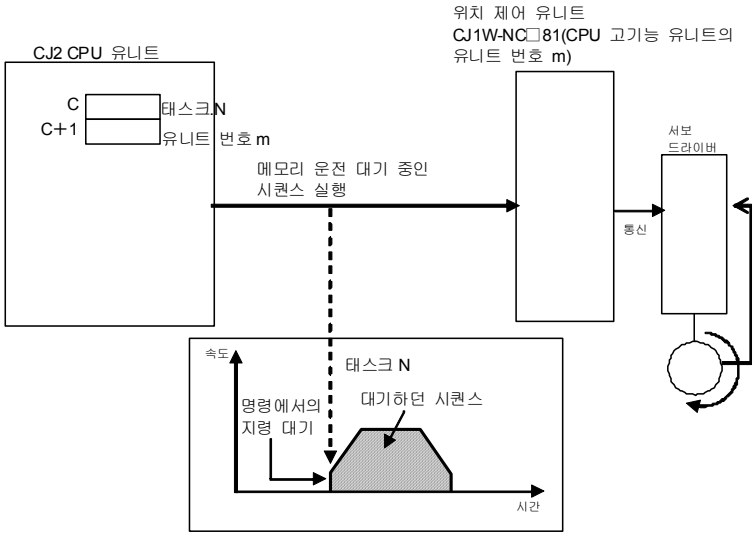
관련 특수 보조 릴레이

이름	주소	내용
CPU 고기능 유니트 초기화 중 플래그	A302.00~ A302.15	CPU 고기능 유니트가 초기(초기화) 처리 중에 1(ON)이 됩니다. 0: 초기 처리 중이 아님 1: 초기 처리 중

기능

C+1로 지정한 위치 제어 유니트의, C로 지정한 메모리 운전의 태스크에 대해 명령에서 지령 입력 대기로 대기하고 있는 시퀀스를 시작합니다.

위치 제어 유니트 측에서 정상으로 시퀀스가 기동되었다는 것은 조건 플래그의 = 플래그 ON을 통해 확인합니다.



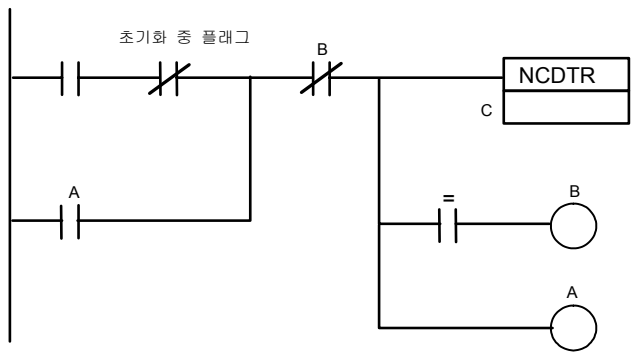
대상 위치 제어 유니트는 다음과 같습니다.

CJ1W-NC281/481/881(CPU 고기능 유니트)

위치 제어 유니트의 메모리 운전 사양은 "SYSMAC CJ 시리즈 위치 제어 유니트(CJ1W-NC281/481/881) 사용 설명서"(SBCE-323)의 '7장 메모리 운전'을 참조하십시오.

사용상의 주의 사항

- 위치 제어 유니트는 1제어 주기에서 이 명령을 1명령만 실행할 수 있습니다.
  이로 인해 복수의 태스크에 대해 이 명령을 실행하는 등, 타이밍에 따라 위치 제어 유니트가 명령을 수신하지 않고 종료(= 플래그가 OFF)되는 경우가 있습니다.
  따라서 조건 플래그의 = 플래그가 ON이 될 때까지 이 명령을 사이클마다 실행하도록 다음 그림과 같은 자체 유지 회로를 반드시 구성하십시오.
- 지정된 위치 제어 유니트가 초기(초기화) 처리 중일 때는 이 명령이 동작하지 않습니다. 따라서 이 명령의 실행 조건에는 다음 그림과 같이 초기화 중 플래그(CPU 고기능 유니트 초기화 중 플래그 (A302.00~A302.15))를 b 접점으로 삽입하십시오.

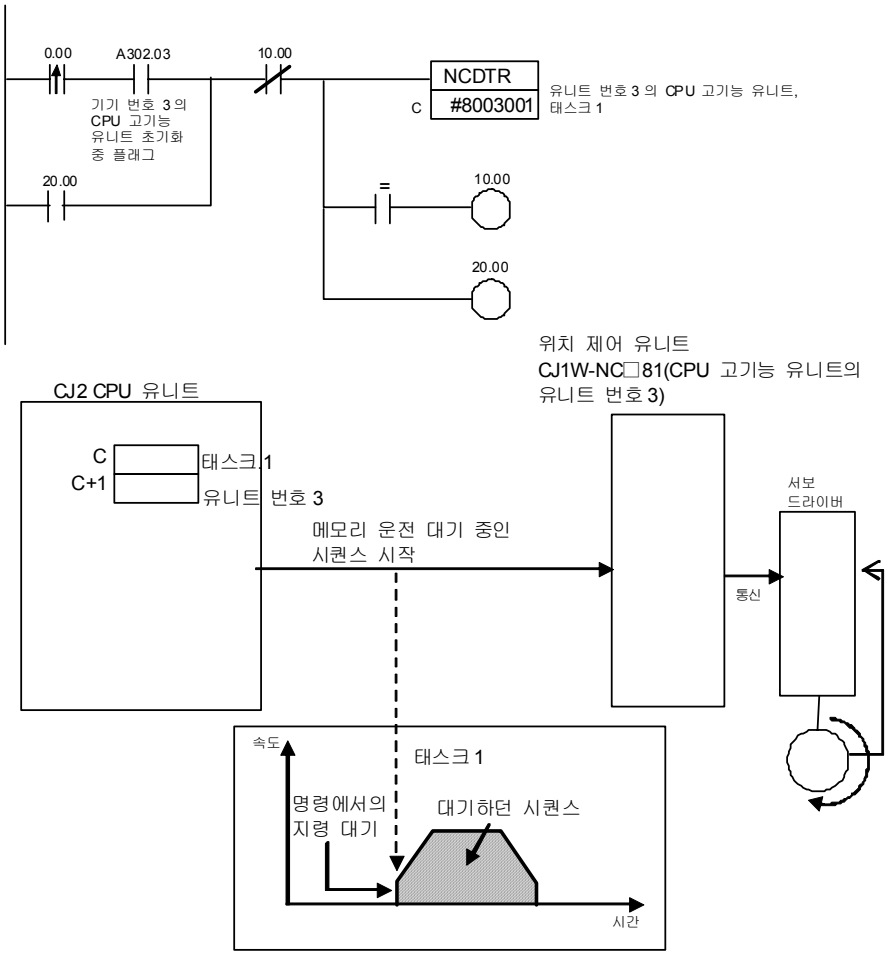


NCDTR

- 인터럽트 태스크 중에 이 명령을 사용할 수 있습니다.
  단, 그 경우 사이클릭 태스크에서 동일한 유니트에 대해 이 명령(및 IORD/IOWR 명령)을 실행하지 마십시오.
  사이클릭 태스크에서 이 명령을 실행하는 중에 인터럽트 태스크에서도 중복 실행하려고 하면 다중 리프레시 이상(운전 계속 이상)이 발생합니다(A402.13이 ON).

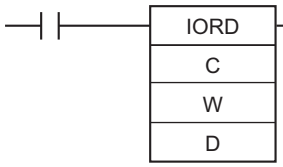
샘플 프로그램

접점 0.00을 ON하면 유니트 번호 3의 위치 제어 유니트의 태스크 1에 대해 명령에서 지령 입력 대기로 대기하고 있는 시퀀스를 시작합니다.



IORD

명령 이름	니모닉	동작 옵션	평션 번호	기능
인텔리전트 I/O 읽기	IORD	@IORD	222	고기능 I/O 유니트 또는 CPU 고기능 유니트의 메모리 영역의 내용을 읽습니다.

심볼	IORD
	 C: 제어 데이터 W: 전송 원본 기기 번호(또는 유니트 번호) 및 전송 CH 수 D: 전송 대상 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	UINT	1
W	전송 원본 기기 번호(또는 유니트 번호) 및 전송 CH 수	UDINT	2
D	전송 대상 하위 CH 번호	UINT	가변

주: 피연산자(W)에 CPU 고기능 유니트의 유니트 번호를 지정한 이 명령을 사용자 프로그램에 기술하고, 대응하지 않는 CPU 유니트에 전송하여 운전 중에 명령의 실행 조건이 성립하면 에러가 발생하고 에러 플래그는 ON이 됩니다.

C: 제어 데이터

고기능 I/O 유니트(또는 CPU 고기능 유니트)에 따라 달라집니다.

W: 전송 원본 기기 번호(또는 유니트 번호) 및 전송 CH 수

고기능 I/O 유니트의 경우, 기기 번호 0~95를 10진수 &0~95 또는 16진수 #0000~005F로 지정. CPU 고기능 유니트의 경우, 유니트 번호 0~F를 유니트 번호+16진수 #8000(16진수 #8000~800F)으로 지정

W+1: 전송 CH 수 10진수 &1~128 또는 16진수 #0001~0080

고기능 I/O 유니트(또는 CPU 고기능 유니트)에 따라 달라집니다.



W+1이 상위 4자리,
W가 하위 4자리

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
W												–						
D											–							

조건 플래그의 동작

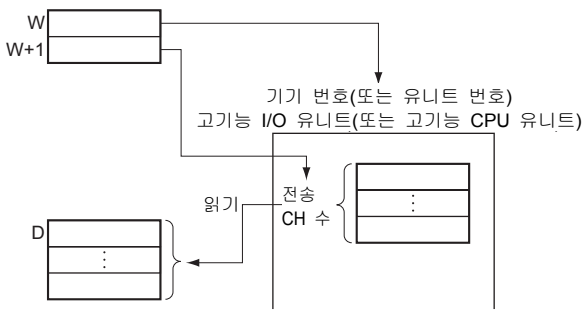
이름	라벨	내용
에러 플래그	P_ER	· W의 전송 CH 수 데이터가 10진수 &1~128 또는 16진수 #0001~0080의 범위 안에 없을 때 ON · W의 기기 번호 지정 또는 유니트 번호 지정이 10진수 &0~95 또는 16진수 #0000~005F의 범위 내에도 없고, 16진수 #8000~800F의 범위 내에도 없을 때 ON · 지정된 고기능 I/O 유니트가 SYSBUS에 있을 때 ON · IORD 명령에 대응하지 않는 고기능 I/O 유니트(또는 CPU 고기능 유니트)를 지정했을 때 ON · CS1D CPU 유니트에서 활성－대기 사이에서 동기화되지 않았을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 읽기가 정상적으로 종료되었을 때 ON · 읽기가 정상적으로 종료되지 않았을 때 OFF

기능

W의 기기 번호(또는 유니트 번호)로 지정한 고기능 I/O 유니트(또는 CPU 고기능 유니트)의 메모리 영역의 내용을 전송 CH 수만큼 읽어 D에 출력합니다.

대상이 되는 고기능 I/O 유니트(또는 CPU 고기능 유니트)는 CPU 장치 및 증설 장치에 장착된 유니트뿐입니다.

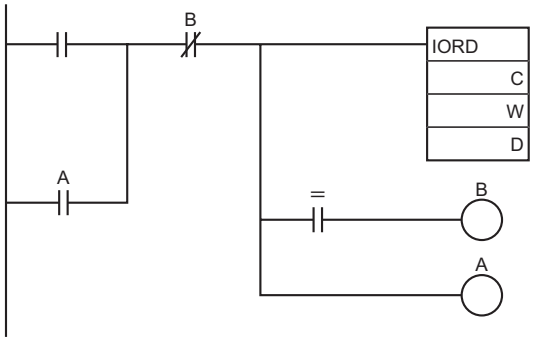
또한 IORD 명령의 자세한 사용 방법은 읽기 대상이 되는 각 고기능 I/O 유니트(또는 CPU 고기능 유니트)의 사용 설명서를 참조하십시오.



힌트

· 이 명령을 실행하면 조건 플래그에 실행 결과가 반영됩니다.
특히 읽기가 완료되면 = 플래그가 ON이 됩니다.
= 플래그 등의 조건 플래그는 IORD 명령과 동일한 입력 조건의 출력 분기에서 입력하십시오.

· 고기능 I/O 유니트(또는 CPU 고기능 유니트)가 다른 작업 중인 경우, 읽기는 실행되지 않습니다. 따라서 읽기가 실행될 때까지 이 명령을 사이클마다 실행하도록 다음과 같이 = 플래그를 사용하여 자체 유지 회로를 구성하십시오.



· 입력 조건이 성립하면 접점 A에 의해 자체 유지. = 플래그가 ON이 될 때까지 IORD 명령을 사이클마다 실행합니다. 읽기가 완료되어 = 플래그가 ON이 되면 접점 B가 ON이 되어 자체 유지를 해제합니다.

· 조건 플래그를 다른 명령 뒤에 배치하면 그 명령의 실행 결과에 의해 조건 플래그가 바뀌므로 조건 플래그는 반드시 IORD 명령 직후에 배치하십시오.

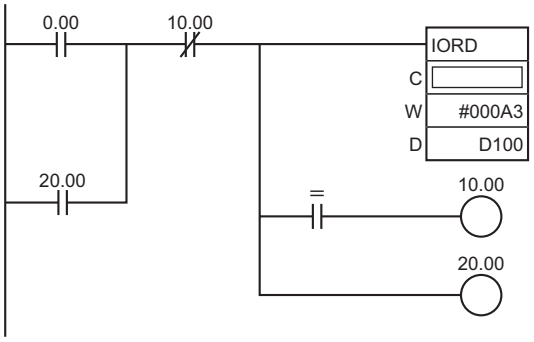
사용상의 주의 사항

· 이 명령을 인터럽트 태스크 중에 사용할 수 있습니다. (그 결과, 인터럽트에 의한 특정 입출력의 고속 응답 처리를 실현할 수 있습니다.)
단, 이런 경우 반드시 PLC 시스템 설정의 '고기능 I/O 유니트 주기적인 리프레시 유무 지정'을 고기능 I/O 유니트와 주기적인 리프레시 금지(1: 주기적 리프레시 안 함)로 설정하십시오.
고기능 I/O 유니트와의 주기적 리프레시(0: 주기적 리프레시 수행)로 설정한 상태에서 인터럽트 태스크 중에 다음 중 하나를 실행하면 '다중 리프레시'가 되어 운전 계속 이상이 발생하고 인터럽트 태스크 이상 플래그(A402.13)가 ON이 됩니다.

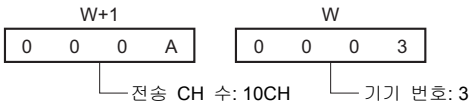
1) IORF/FIORF 명령을 통해 동일한 고기능 I/O 유니트에 대해 I/O 리프레시를 실행했을 때

2) IORD/IOWR 명령을 통해 동일한 고기능 I/O 유니트의 메모리 영역에 대해 읽기 또는 쓰기를 실행했을 때

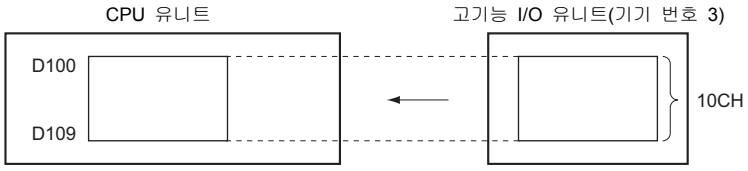
샘플 프로그램



0.00이 ON이면 기기 번호 3의 고기능 I/O 유니트에서 10CH를 읽어 D100~D109에 저장합니다.

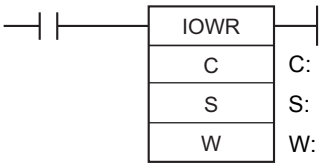


제어 코드(C)의 내용은 각 고기능 I/O 유니트에 따라 다릅니다.



IOWR

명령 이름	니모닉	동작 옵션	평션 번호	기능
인텔리전트 I/O 쓰기	IOWR	@IOWR	223	CPU 유니트의 I/O 메모리 영역의 내용을 고기능 I/O 유니트 또는 CPU 고기능 유니트에 출력합니다.

	IOWR
심볼	 C: 제어 데이터 S: 전송 원본 하위 CH 번호 W: 전송 대상 기기 번호(또는 유니트 번호) 및 전송 CH 수

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	UINT	1
S	전송 원본 하위 CH 번호	WORD	가변
W	전송 대상 기기 번호(또는 유니트 번호) 및 전송 CH 수	UDINT	2

주: 피연산자(W)에 CPU 고기능 유니트의 유니트 번호를 지정한 이 명령을 사용자 프로그램에 기술하고, 대응하지 않는 CPU 유니트에 전송하여 운전 중에 명령의 실행 조건이 성립하면 에러가 발생하고 에러 플래그는 ON이 됩니다.

C: 제어 데이터

고기능 I/O 유니트(또는 CPU 고기능 유니트)에 따라 달라집니다.

W: 전송 대상 기기 번호(또는 유니트 번호) 및 전송 CH 수

고기능 I/O 유니트의 경우, 기기 번호 0~95를 10진수 &0~95 또는 16진수 #0000~005F로 지정. CPU 고기능 유니트의 경우, 유니트 번호 0~F를 유니트 번호+16진수 #8000(16진수 #8000~800F)으로 지정

W+1: 전송 CH 수 10진수 &1~128 또는 16진수 #0001~0080

고기능 I/O 유니트(또는 CPU 고기능 유니트)에 따라 달라집니다.



W+1이 상위 4자리,
W가 하위 4자리

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
S												–						
W																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· W의 전송 CH 수 데이터가 10진수 &1~128 또는 16진수 #0001~0080의 범위 안에 없을 때 ON · W의 기기 번호 지정 또는 유니트 번호 지정이 10진수 &0~95 또는 16진수 #0000~005F의 범위 내에도 없고, 16진수 #8000~800F의 범위 내에도 없을 때 ON · S가 상수로 지정된 경우이고 W의 전송 CH 수 데이터가 1이 아닐 때 ON · 지정된 고기능 I/O 유니트가 SYSBUS에 있을 때 ON · IOWR 명령에 대응하지 않는 고기능 I/O 유니트(또는 CPU 고기능 유니트)를 지정했을 때 ON · CS1D CPU 유니트에서 활성－대기 사이에서 동기화되지 않았을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 쓰기가 정상적으로 종료되었을 때 ON · 쓰기가 정상적으로 종료되지 않았을 때 ON

주 :

· 전송 CH 수(W+1)를 "0001"로 지정한 경우는 S의 데이터를 상수로 지정할 수 있습니다. 전송 CH 수가 "0001"이 아닐 때 S에 상수를 지정한 경우는 에러가 됩니다.

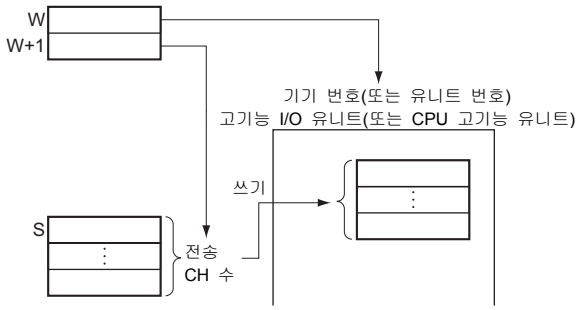
· 고기능 I/O 유니트(또는 CPU 고기능 유니트)가 다른 작업 중인 상태에서 정상적으로 쓰기가 실행되지 않은 경우는 = 플래그가 OFF가 됩니다.

· 고기능 I/O 유니트(또는 CPU 고기능 유니트)에 I/O 유니트 조회 이상, 고기능 I/O 유니트(또는 CPU 고기능 유니트) 설정 이상, 고기능 I/O 유니트(또는 CPU 고기능 유니트) 이상이 발생한 경우는 에러가 됩니다.

기능

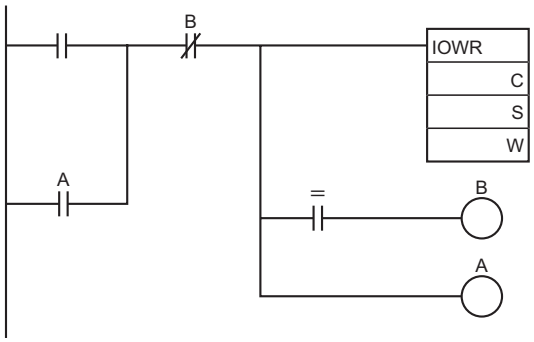
S의 전송 원본 하위 CH 번호부터 전송 CH 수(W) 분량의 데이터를, W의 기기 번호(또는 유니트 번호)로 지정한 고기능 I/O 유니트(또는 CPU 고기능 유니트)의 메모리 영역에 출력합니다. 대상이 되는 고기능 I/O 유니트(또는 CPU 고기능 유니트)는 CPU 장치 및 증설 장치에 장착된 유니트뿐입니다.

또한 IOWR 명령의 자세한 사용 방법은 쓰기 대상이 되는 각 고기능 I/O 유니트(또는 CPU 고기능 유니트)의 사용 설명서를 참조하십시오.



힌트

- 이 명령을 실행하면 조건 플래그에 실행 결과가 반영됩니다.
  특히 읽기가 완료되면 **=** 플래그가 ON이 됩니다.
  = 플래그 등의 조건 플래그는 IOWR 명령과 동일한 입력 조건의 출력 분기에서 입력하십시오.
- 고기능 I/O 유니트(또는 CPU 고기능 유니트)가 다른 작업 중인 경우, 읽기는 실행되지 않습니다. 따라서 쓰기가 실행될 때까지 이 명령을 사이클마다 실행하도록 다음과 같이 = 플래그를 사용하여 자체 유지 회로를 구성하십시오.



- 입력 조건이 성립하면 접점 A에 의해 자체 유지. = 플래그가 ON이 될 때까지 IOWR 명령을 사이클마다 실행합니다. 쓰기가 완료되어 **=** 플래그가 ON이 되면 접점 B가 ON이 되어 자체 유지를 해제합니다.
- 조건 플래그를 다른 명령 뒤에 배치하면 그 명령의 실행 결과에 의해 조건 플래그가 바뀌므로 조건 플래그는 반드시 IOWR 명령 직후에 배치하십시오.

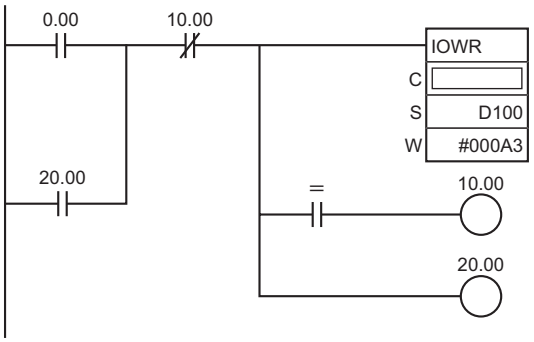
사용상의 주의 사항

- 이 명령을 인터럽트 태스크 중에 사용할 수 있습니다. (그 결과, 인터럽트에 의한 특정 입출력의 고속 응답 처리를 실현할 수 있습니다.)
  단, 이런 경우 반드시 PLC 시스템 설정의 '고기능 I/O 유니트 주기적인 리프레시 유무 지정'을 고기능 I/O 유니트와 주기적인 리프레시 금지(1: 주기적 리프레시 안 함)로 설정하십시오.
  고기능 I/O 유니트와의 주기적 리프레시(0: 주기적 리프레시 수행)로 설정한 상태에서 인터럽트 태스크 중에 다음 중 하나를 실행하면 '다중 리프레시'가 되어 운전 계속 이상이 발생하고 인터럽트 태스크 이상 플래그(A402.13)가 ON이 됩니다.

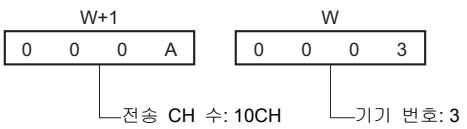
  1) IORF/FIORF 명령을 통해 동일한 고기능 I/O 유니트에 대해 I/O 리프레시를 실행했을 때

  2) IORD/IOWR 명령을 통해 동일한 고기능 I/O 유니트의 메모리 영역에 대해 읽기 또는 쓰기를 실행했을 때

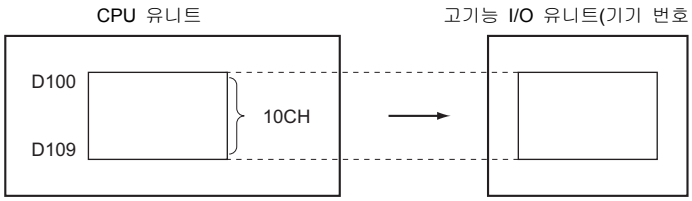
샘플 프로그램



0.00이 ON이면 기기 번호 3의 고기능 I/O 유니트에 대해 D100~D109의 10CH를 씁니다.



제어 코드(C)의 내용은 각 고기능 I/O 유니트에 따라 다릅니다.



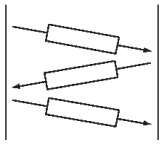
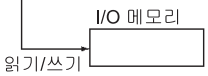
시리얼 통신 명령

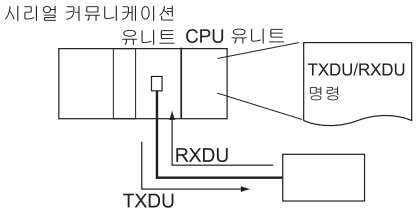
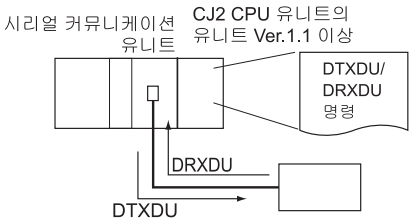
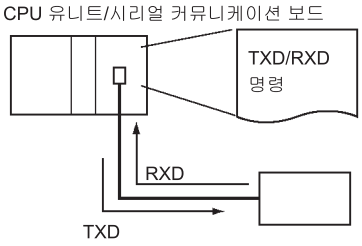
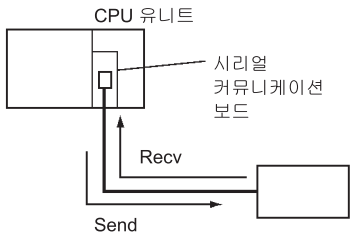
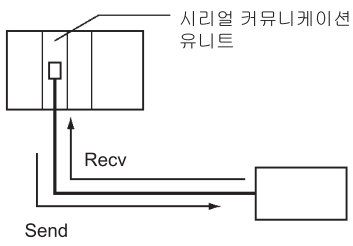
니모닉	명령 이름	펑션 번호	페이지
PMCR	프로토콜 매크로	260	3-700
TXD	CPU 유니트 시리얼 커뮤니케이션 보드 시리얼 포트 출력	236	3-706
RXD	CPU 유니트 시리얼 커뮤니케이션 보드 시리얼 포트 입력	235	3-712
TXDU	시리얼 커뮤니케이션 유니트 시리얼 포트 출력	256	3-720
RXDU	시리얼 커뮤니케이션 유니트 시리얼 포트 입력	255	3-726
DTXDU	시리얼 커뮤니케이션 유니트 시리얼 포트 직접 송신	262	3-733
DRXDU	시리얼 커뮤니케이션 유니트 시리얼 포트 직접 수신	261	3-738
STUP	시리얼 포트 통신 설정 변경	237	3-745
PMCR2	프로토콜 매크로 송신 2	264	3-748

시리얼 통신 명령

시리얼 통신 명령에는 무수순 모드에서 범용 외부 기기와 데이터 송신 또는 수신을 수행하는 TXD/RXD/TXDU/RXDU/DTXDU/DRXDU 명령과 사용자 정의 프로토콜로 범용 외부 기기와 데이터 송신 또는 수신을 수행하는 PMCR 명령이 있습니다.

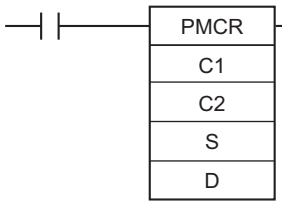
- TXD/RXD 명령은 CPU 유니트 내장 시리얼 포트, 시리얼 커뮤니케이션 보드(유니트 Ver.1.2 이상)의 시리얼 포트 전용 명령입니다.
- TXDU/RXDU 명령은 시리얼 커뮤니케이션 유니트(유니트 Ver.1.2 이상) 전용 명령입니다.
- DTXDU/DRXDU 명령은 CJ2H CPU 유니트의 유니트 Ver.1.1 이상에 시리얼 커뮤니케이션 유니트 CJ1W-SCU22/32/42를 장착했을 때의 전용 명령입니다. 고속으로 송수신이 가능합니다.
- PMCR/PMCR2 명령은 시리얼 커뮤니케이션 유니트/보드의 각 시리얼 포트 전용 명령입니다.

명령	통신 프레임	기능
TXD/RXD/TXDU/RXDU/DTXDU/DRXDU 명령	다음 6종류 중 하나 • 시작 코드/종료 코드 없음: 데이터 • 시작 코드/종료 코드 모두 있음: ST | 데이터 | ED • 시작 코드만 있음: ST | 데이터 • 종료 코드 CR+LF: 데이터 | CR | LF • 종료 코드만 있음: 데이터 | ED • 시작 코드 있음, 종료 코드 CR+LF: ST | 데이터 | CR | LF	• 일방적으로 송신 또는 수신 • 송신 딜레이 시간 설정 가능
PMCR/PMCR2 명령	다음 프레임(메시지)을 외부 기기에 맞춰 자유롭게 작성 헤더 | 주소 | 데이터 | 에러확인 | 터미네이터 • 송수신 스텝이 가능  	• 송수신 등의 스텝을 최대 16개 준비 • 응답 내용에 맞는 스텝 전환, 재시도 처리 가능 • 송수신 시간 감시 가능 • PLC와의 읽기/쓰기용 변수 사용 가능 • 반복 변수 준비 등

명령	모드	통신 포트
TXDU/RXDU 명령	무수순	시리얼 커뮤니케이션 유니트(유니트 Ver.1.2 이상)의 시리얼 포트 
DTXDU/DRXDU 명령	무수순	시리얼 커뮤니케이션 유니트 CJ1W-SCU22/32/42의 시리얼 포트 
TXD/RXD 명령	무수순	CPU 유니트 내장 시리얼 포트, 시리얼 커뮤니케이션 보드(유니트 Ver.1.2 이상)의 시리얼 포트 
PMCR 명령 PMCR2 명령	프로토콜 매크로	시리얼 커뮤니케이션 보드 (CS 시리즈만 해당)  시리얼 커뮤니케이션 유니트 

PMCR

명령 이름	니모닉	동작 옵션	펑션 번호	기능
프로토콜 매크로	PMCR	@PMCR	260	시리얼 커뮤니케이션 보드(CS 시리즈만 해당) 또는 시리얼 커뮤니케이션 유니트에 등록된 송수신 시퀀스(프로토콜 데이터)를 호출하여 실행합니다.

	PMCR
심볼	 C1: 제어 데이터(통신 포트 번호, 시리얼 포트 번호, 상대방 기기 주소) C2: 제어 데이터(송수신 시퀀스 번호) S: 송신 데이터 시작 CH 번호 D: 수신 데이터 저장 시작 CH 번호

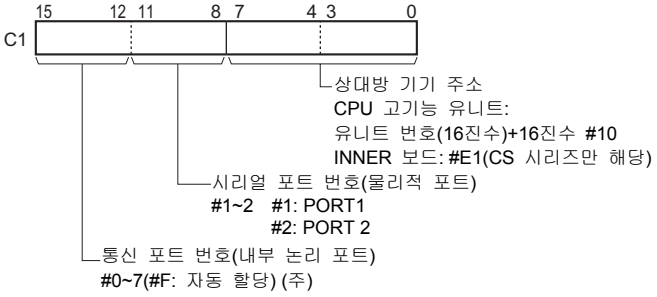
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

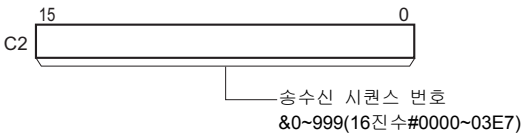
피연산자	내용	데이터 형식	크기
C1	제어 데이터(통신 포트 번호, 시리얼 포트 번호, 상대방 기기 주소)	UINT	1
C2	제어 데이터(송수신 시퀀스 번호)	UINT	1
S	송신 데이터 시작 CH 번호	UINT	가변
D	수신 데이터 저장 시작 CH 번호	UINT	가변

C1: 제어 데이터(통신 포트 번호, 시리얼 포트 번호, 상대방 기기 주소)



주: 자동 할당은 CJ2 CPU 유니트 및 로트 번호 020601(2002년 6월 1일 제조) 이후의 CS1-H/CJ1-H/CJ1M/CS1D(CPU 단독 시스템용) CPU 유니트에서만 사용할 수 있습니다.

C2: 제어 데이터(송수신 시퀀스 번호)

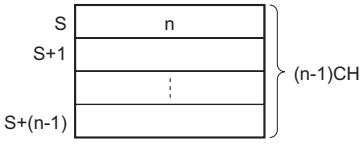


송신 영역 지정

S: 송신 데이터 시작 CH 번호

송신 영역으로 확보할 채널 수를 n으로 S에 설정합니다(S 자체도 포함).

n = 10진수 &0~250 또는 16진수 #0000~00FA가 가능. 그리고 S+1~S+(n-1)에 송신 데이터를 준비해 둡니다.



직접 지정, 링크 채널 지정 등, 실행하는 시퀀스 내에 피연산자를 지정하지 않는 경우는 S에 상수(#0000)를 지정하십시오. 상수 이외의 채널 등을 지정한 경우는 해당 채널의 내부 데이터가 반드시 0000이 되어야 합니다.

다른 상수나 채널을 설정하면 에러(에러 플래그가 ON)가 되어 PMCR 명령이 실행되지 않습니다.

수신 영역 지정

D: 수신 데이터 저장 시작 CH 번호

- PMCR 명령 실행 전의 설정(수신 영역의 유지 지정)

  수신에 실패한 경우 수신 영역의 값을 유지하려면(힌트 항목 참조), 미리 D에 채널 수 m을 지정하고 D+1~D+(m-1)에 유지하려는 데이터를 설정해 둡니다.

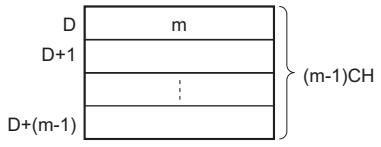
  m = 10진수 &2~250 또는 16진수 #0002~00FA가 가능.

  m에 0 또는 1을 지정한 경우 수신에 실패하면 수신 영역이 0으로 클리어됩니다.

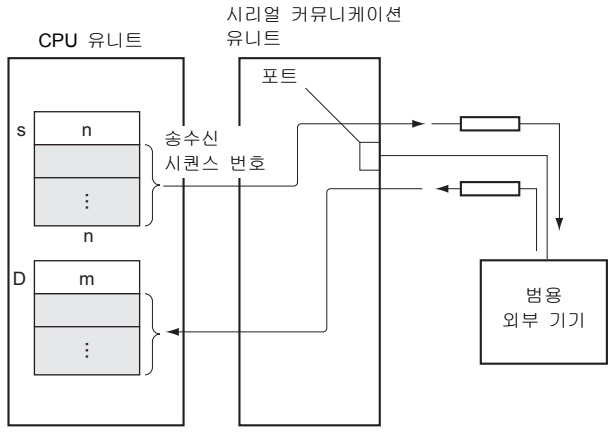
- 프로토콜 매크로 실행 완료 시

  수신한 데이터의 채널 수가 m으로 D에 자동 저장됩니다(m = 10진수 &0~250 또는 16진수 #0000~00FA).

  D+1~D+(m-1)에는 실제로 수신한 데이터가 저장됩니다.



직접 지정, 링크 채널 지정 등, 실행하는 시퀀스 내에 피연산자를 지정하지 않는 경우는 D에 상수(#0000)를 지정하십시오. 상수 이외의 채널 등을 지정한 경우는 해당 채널의 내부 데이터가 반드시 0000이 되어야 합니다.



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C1, C2	○	○	*1 ○	○	○	○	○	*2 ○	○	○	○	○	–	○	–	–	–	–
S												–						
D																		

*1: H512~H1535는 사용할 수 없습니다.

*2: EM D 뱅크 이상은 사용할 수 없습니다.

PMCR

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 명령 실행 시, 지정한 통신 포트(내부 논리 포트)의 네트워크 통신 명령 실행 가능 플래그가 OFF되었을 때 ON · C1의 데이터가 범위 밖일 때 ON · S, D의 데이터 CH 수가 249CH를 초과했을 때 ON(채널 지정 시) · 인터럽트 태스크에서 실행했을 때 ON(CJ2 CPU 유니트에서 인터럽트 고속 기동 기능 사용 시) · 그 이외일 때 OFF

관련 특수 보조 릴레이

이름	주소	내용
네트워크 통신 명령 실행 가능 플래그	A202.00~ A202.07	· 네트워크 통신(PMCR 명령 포함)이 실행 가능할 때 1(ON)이 됩니다. · 각 비트가 통신 포트(내부 논리 포트) 번호에 대응합니다. 비트 00~07: 통신 포트 번호 00~07 · 네트워크 통신 실행 중에 0(OFF), 실행 종료(정상이든 이상이든) 후에는 1(ON)이 됩니다.
네트워크 통신 실행 에러 플래그	A219.00~ A219.07	· 네트워크 통신 실행 시 에러(이상)가 발생한 경우 1(ON)이 됩니다. · 각 비트가 통신 포트(내부 논리 포트) 번호에 대응합니다. 비트 00~07: 통신 포트 번호 00~07 · 다음 네트워크 통신 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 통신 명령 실행에서 0(OFF)이 됩니다.
네트워크 통신 응답 코드	A203CH~ A210CH	· 네트워크 통신이 실행된 경우 응답 코드(종료 코드)가 저장됩니다. · 각 CH는 통신 포트(내부 논리 포트) 번호에 대응합니다. A203~210CH: 통신 포트 번호 00~07 · 통신 명령 실행 중에는 #0000이 되고 통신 명령 실행이 종료되면 반영됩니다. 운전 시작 시에 클리어됩니다.

통신 응답 코드 목록

코드(16진수)	내용
#1106	해당 프로그램 번호 없음 · 등록되지 않은 '송수신 시퀀스 번호'가 지정되었습니다. · '송수신 시퀀스 번호'를 수정하거나 CX-Protocol로 시퀀스를 추가하십시오.
#2201	실행 중이므로 동작 불가능 이미 프로토콜을 실행 중이어서 실행할 수 없습니다. '프로토콜 매크로 실행 중 플래그' b 접점을 추가하십시오.
#2202	정지 중이므로 실행 불가능 프로토콜 전환 중이어서 실행할 수 없습니다. '시리얼 설정 변경 중 플래그'의 b 접점을 추가하십시오.
#2401	등록 테이블 없음 · 프로토콜 매크로 데이터에서 SUM 값 이상이 발생했거나 데이터 전송 중입니다. · CX-Protocol로 프로토콜 매크로 데이터를 전송하십시오.

그 밖의 응답 코드는 'SYSMAC CS/CJ 시리즈 통신 명령 참조 설명서(SBCA-304)'를 참조하십시오.

관련 **INNER** 보드 릴레이 영역**(CS** 시리즈만 해당**)**

주소	이름		내용
1909.15	프로토콜 매크로 실행 중 플래그	포트 1용	PMCR 명령이 실행되면 1(ON)이 됩니다. 실행에 실패하는 경우 0(OFF) 그대로이고 송수신 시퀀스가 종료(End 또는 Abort에 의해)된 경우에는 0(OFF)이 됩니다.
1919.15		포트 2용	

관련 CPU 고기능 유니트 릴레이 영역

n=1500+25×유니트 번호

주소	이름		내용
n+9CH의 비트 15	프로토콜 매크로 실행 중 플래그	포트 1용	PMCR 명령이 실행되면 1(ON)이 됩니다. 실행에 실패하는 경우 0(OFF) 그대로이고 송수신 시퀀스가 종료(End 또는 Abort에 의해)된 경우에는 0(OFF)이 됩니다.
n+19CH의 비트 15		포트 2용	

기능

C2로 지정한 번호의 송수신 시퀀스를 C1의 비트 12~15로 지정한 통신 포트(내부 논리 포트) 0~7 중 하나를 사용하여 C1의 비트 0~7로 지정한 기기 주소(장치)의 시리얼 포트 중, C1의 비트 8~11로 지정한 시리얼 포트(물리적 포트)에서 실행합니다.

송신 메시지 내의 변수가 피연산자 지정인 경우, S+1CH부터 S의 내용을 채널 수로 한 데이터를 송신 영역으로 사용합니다.

수신 메시지 내의 변수가 피연산자 지정인 경우, 수신에 성공했을 때 D+1CH 이후에 수신하고, D에 해당 수신 데이터의 채널 수(D 자체 포함)가 자동으로 저장됩니다.

수신 처리에 실패하면 PMCR 명령 실행 전에 설정해둔 D+1CH 이후의 데이터를 수신 버퍼에서 읽어 다시 D+1CH 이후에 저장합니다. (이 기능을 통해 수신에 실패했을 때도 현재값 데이터가 0으로 클리어되지 않고 직전의 수신 데이터로 유지됩니다.)

주: 통신 포트(내부 논리 포트) 번호를 자동 할당("F")으로 사용하는 경우에 대해서는 3-736의 '내부 논리 포트 자동 할당 기능'의 설명을 참조하십시오.

힌트

■ 수신 영역의 유지 지정 방법

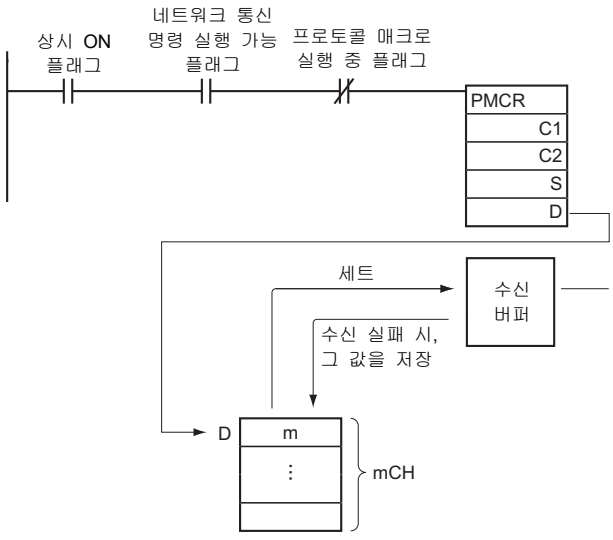
수신 버퍼는 PMCR 명령을 실행할 때 송수신 시퀀스 실행 직전에 일단 0으로 클리어됩니다. 따라서 다음과 같이 정상적으로 현재값 데이터 등을 읽는 래더 프로그램을 작성한 경우, 수신 에러 등에 의해 데이터가 전송되지 않았을 때 현재값 데이터가 순간적으로 0으로 클리어됩니다.

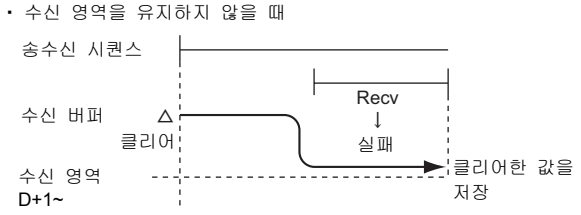
따라서 수신 에러가 발생해도 직전의 수신 데이터를 유지하도록 일단 0으로 클리어한 후 송수신 시퀀스 실행 직전에 수신 영역의 선두 m의 CH 수만큼 CPU 유니트의 I/O 메모리에서 읽어 설정해 두는 기능이 있습니다.

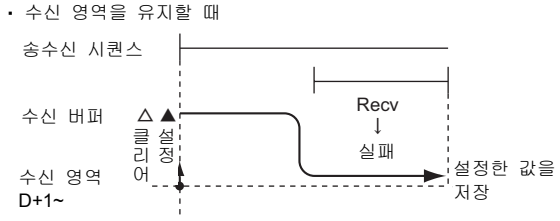
이 기능을 통해 수신에 실패했을 때도 현재값 데이터가 0으로 클리어되지 않고 직전의 수신 데이터가 유지됩니다.

유지하려는 데이터의 CH 수만큼 m으로 지정하십시오. (0 또는 1로 지정된 경우는 직전의 수신 데이터가 유지되지 않고 모두 0으로 클리어됩니다.)

예: 1회만 송수신을 수행하여 수신 데이터를 전송하는 PMCR 명령을 래더 프로그램에서 항상(또는 정기적으로) 실행하는 프로토콜



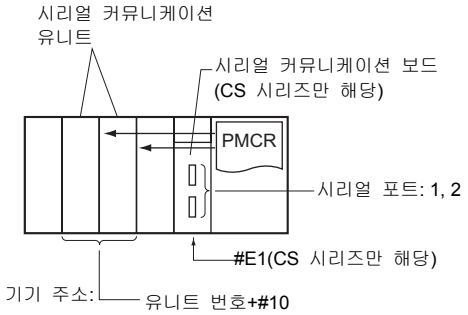




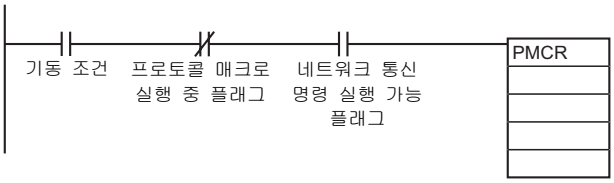
사용상의 주의 사항

- 실제로 S로 지정한 송신 영역의 데이터를 송신하는 것은 송신 메시지 내 변수의 읽기 옵션R( )에 따라 결정됩니다. 또한 실제로 D로 지정한 수신 영역에 데이터를 수신하는 것은 수신 메시지 내 변수의 쓰기 옵션W( )에 따라 결정됩니다.
- 메시지 내의 변수 (R( )) 또는 (W( ))의 지정 방법은 CX-Protocol의 조작 설명서를 참조하십시오.
- PMCR 명령은 시리얼 커뮤니케이션 보드(CS 시리즈만 해당) 또는 시리얼 커뮤니케이션 유니트에 대해 실행할 수 있습니다. 시리얼 커뮤니케이션 유니트는 CPU 장치 또는 증설 장치에 최대 16대를 장착할 수 있습니다. 따라서 어느 유니트/보드의 어느 시리얼 포트에 대해 PMCR 명령을 실행할지 식별하기 위해 상대방 기기 주소를 C1의 비트 0~7에서 지정하고 상대방의 시리얼 포트 번호를 C1의 비트 8~11에서 지정합니다. 기기 주소 지정 방법은 다음과 같습니다.

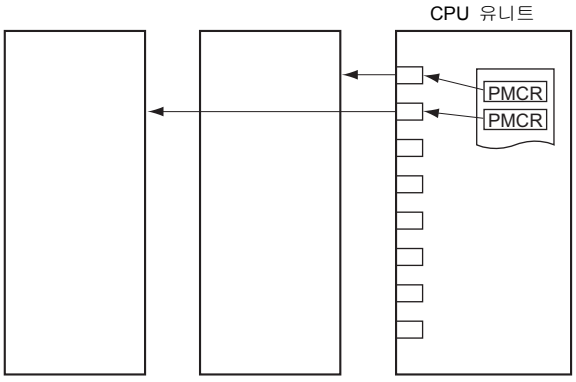
유니트(장치) 종류	기기 주소
시리얼 커뮤니케이션 보드 (CS 시리즈만 해당)	16진수 #E1
시리얼 커뮤니케이션 유니트	유니트 번호(16진수)+16진수 #10



- '프로토콜 매크로 실행 중 플래그'는 PMCR 명령 실행 시작에서 ON되며, 송수신 시퀀스의 실행이 종료하여 지정된 릴레이 영역에 수신한 데이터가 완전히 저장된 단계에서 OFF가 됩니다. 따라서 PMCR 명령을 통해 지정된 시리얼 포트(물리적 포트)에서 송수신 시퀀스를 실행 중에는 다른 PMCR 명령을 동일한 시리얼 포트에서 실행하지 않도록 PMCR 명령의 입력 조건에 이 '프로토콜 매크로 실행 중 플래그'를 b 접점으로 삽입하십시오.

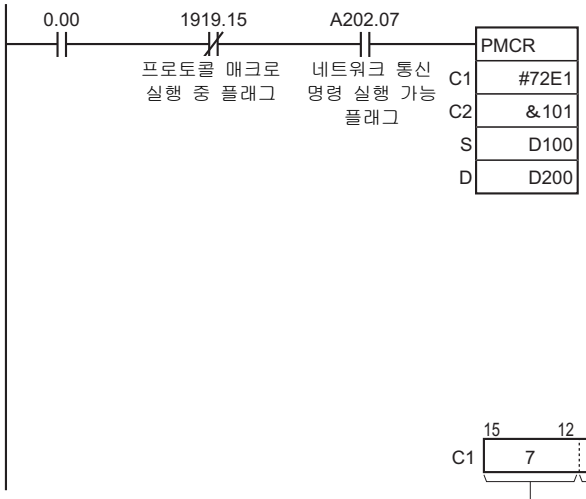


- PMCR 명령(내부적으로 FINS 명령을 전송했기 때문)은 네트워크 통신 명령의 SEND/RECV/CMND 명령과 동일하게 내부 논리 포트 0~7을 사용하여 시리얼 커뮤니케이션 보드(CS 시리즈만 해당) 또는 시리얼 커뮤니케이션 유니트에 대해 송수신 시퀀스 실행 지시를 내립니다. 따라서 네트워크 통신 명령이나 다른 PMCR 명령에 의해 내부 논리 포트가 사용될 때는 해당 통신 포트를 사용하여 PMCR 명령을 실행할 수 없습니다.
  그러므로 PMCR 명령의 입력 조건에는 C1의 비트 12~15로 지정한 사용 통신 포트 번호에 대응한 특수 보조 릴레이의 '네트워크 통신 실행 가능 플래그'(A202.00~A202.07)를 a 접점으로 삽입하십시오.

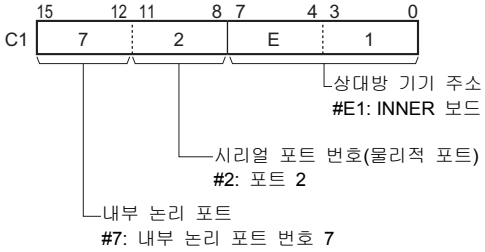


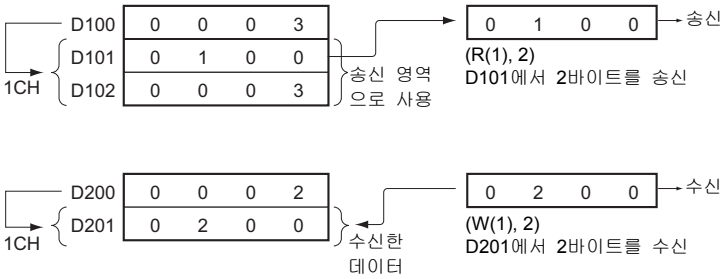
- C2의 데이터가 범위 밖인 경우 에러 플래그는 ON으로 되지 않습니다. 특수 보조 릴레이의 '네트워크 통신 응답 코드'(A203~210CH)에 종료 코드가 저장됩니다.

샘플 프로그램



0.00 및 A202.07(내부 논리 포트 번호 7용 네트워크 통신 명령 실행 가능 플래그)이 ON이고 1919.15(프로토콜 매크로 실행 중 플래그)가 OFF일 때 송수신 시퀀스 번호 101(&101 또는 #0065)을 실행합니다.
송신 메시지 내의 변수가 피연산자 지정인 경우, D101부터 2CH(D100의 내용이 #0003)의 데이터를 송신 영역으로 사용합니다.
수신 메시지 내의 변수가 피연산자 지정인 경우, 수신한 데이터가 D201 이후에 저장되고, 수신한 데이터의 CH 수가 자동으로(D200을 포함한 수로) D200에 저장됩니다.

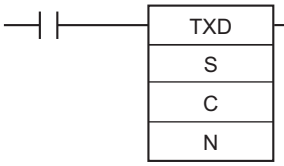




주: 실제 송/수신은 위와 같이 송신 메시지 내 변수의 읽기 옵션 R( ) 및 수신 메시지 내 변수의 쓰기 옵션 W( )로 구성됩니다.

TXD

명령 이름	니모닉	동작 옵션	평션 번호	기능
CPU 유니트 시리얼 커뮤니케이션 보드 시리얼 포트 출력	TXD	@TXD	236	지정된 바이트 수 분량의 데이터를 그 상태 그대로 변환하지 않고 PLC 시스템 설정의 무수순 모드일 때의 시작 코드/종료 코드 지정에 따라 CPU 유니트 내장 RS-232C 포트(무수순 모드) 또는 시리얼 커뮤니케이션 보드 유니트 Ver.1.2 이상의 시리얼 통신 포트(무수순 모드)에서 송신합니다.

심볼	TXD
	 S: 송신 데이터 시작 CH 번호 C: 제어 데이터 N: 송신 바이트 수 10진수 &0~256(16진수 #0000~0100)

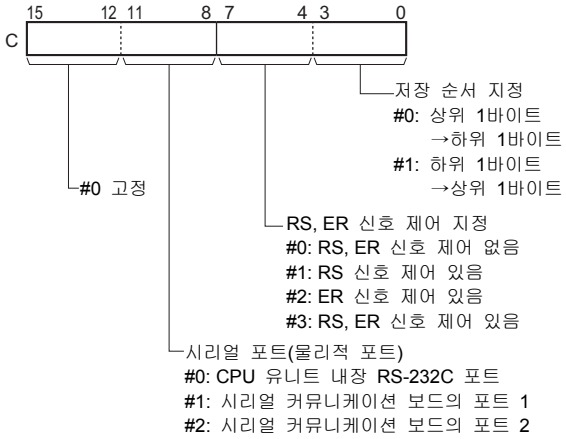
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	송신 데이터 시작 CH 번호	UINT	가변
C	제어 데이터	UINT	1
N	송신 바이트 수 10진수 &0~256(16진수 #0000~0100)	UINT	1

C: 제어 데이터



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
C											○	○						
N																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· CPU 유니트 내장 RS-232C 포트 지정에서 PLC 시스템 설정이 무수순 모드가 아닐 때 ON · 시리얼 커뮤니케이션 보드의 시리얼 포트 지정 시, 할당 DM 영역 설정이 무수순 모드가 아닐 때 또는 시리얼 커뮤니케이션 보드가 유니트 Ver.이 없는 타입(무수순에 대응되지 않는 타입)일 때 ON · C의 데이터가 범위 밖일 때 ON · N의 데이터가 10진수 &0~256 또는 16진수 #0000~0100의 범위에 없을 때 ON · 송신 준비 플래그(CPU 유니트 내장 포트: A392.05, 시리얼 커뮤니케이션 보드의 시리얼 포트 1: A356.05, 시리얼 포트 2: A356.13)가 OFF인 경우 송신하려고 했을 때 ON · 사이클릭 태스크에서 시리얼 커뮤니케이션 보드의 포트를 지정한 TXD/RXD 명령 실행 중에 인터럽트가 발생하여 인터럽트 태스크가 기동하고, 인터럽트 태스크 내에서도 시리얼 커뮤니케이션 보드의 포트를 지정한 TXD/RXD 명령을 실행했을 때 ON(인터럽트 태스크 내 TXD/RXD 명령의 에러 플래그) · 시리얼 커뮤니케이션이 재시작 중에 시리얼 커뮤니케이션 보드의 시리얼 포트 지정에서 TXD 명령을 실행했을 때 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

· CPU 유니트 내장 RS-232C 포트의 경우

이름	주소	내용
RS-232C 포트 송신 준비 플래그	A392.05	무수순 모드에서 송신 가능할 때 1(ON)이 됩니다.

· 시리얼 커뮤니케이션 보드(유니트 Ver.1.2 이상) 포트 1의 경우

이름	주소	내용
포트 1 송신 준비 플래그	A356.05	무수순 모드에서 송신 가능할 때 1(ON)이 됩니다.

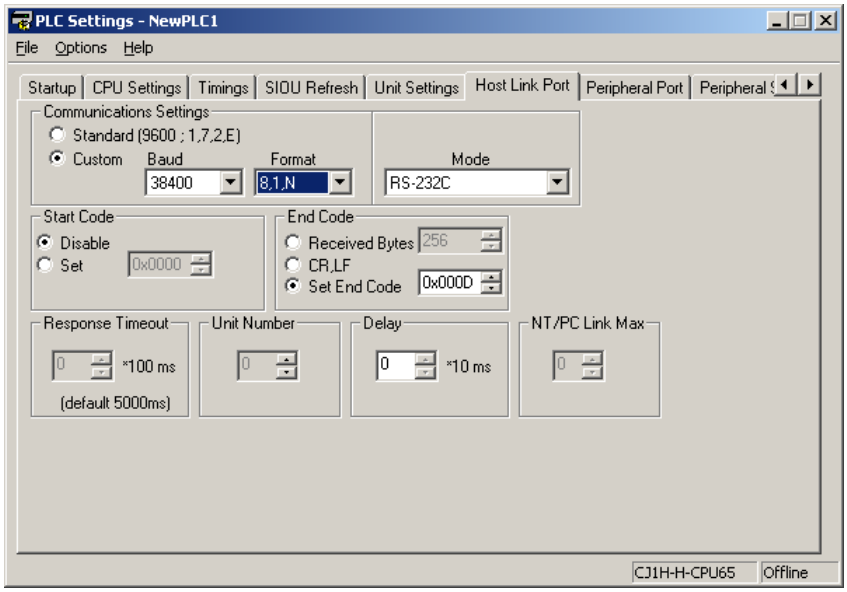
· 포트 2의 경우

이름	주소	내용
포트 2 송신 준비 플래그	A356.13	무수순 모드에서 송신 가능할 때 1(ON)이 됩니다.

이름		주소	내용
포트 1/2	INNER 보드 서비스 불가능 플래그	A424.04	무수순 모드를 지원하지 않는 보드(유니트 Ver. 표기가 없는 타입)에 대해 TXD 명령을 실행했을 때 ON

관련 PLC 시스템 설정(CPU 유니트 내장 RS-232C 포트 사용 시)

■ PLC 시스템 설정(CX-Programmer의 경우)



■ 관련 PLC 시스템 설정

프로그래밍 콘솔 주소		내용	설정
채널	비트		
+162	0~15	무수순 모드 시 송신 딜레이 시간	16진수 #0000~210F: 10진수 0~99990ms(10ms 단위)
+164	8~15	무수순 모드 시 시작 코드	16진수 #00~FF
	0~7	무수순 모드 시 종료 코드	16진수 #00~FF
+165	12	무수순 모드 시 시작 코드 유무 설정	16진수 #0: 없음 16진수 #1: 있음
	8~9	무수순 모드 시 종료 코드 유무 설정	16진수 #0: 없음 16진수 #1: 있음 16진수 #2: CR+LF 지정
	0~7	무수순 모드 시 송신 데이터 수 설정	16진수 #00(초기값): 256 16진수 #01~FF: 10진수 1~255

■ 할당 DM 설정 영역(시리얼 커뮤니케이션 보드의 시리얼 포트 사용 시)

이름		비트	내용	설정
포트 1	포트 2			
D32002	D32012	15	무수순 모드 시 송신 딜레이 시간	0: 기본값(0ms) 1: 임의 설정
		14~00	무수순 모드 시 송신 딜레이 임의 설정 시간	16진수 #0000~7530: 10진수 0~300000ms [10ms 단위]
D32004	D32014	15~08	무수순 모드 시 시작 코드	16진수 #00~FF
		07~00	무수순 모드 시 종료 코드	16진수 #00~FF
D32005	D32015	15~12	무수순 모드 시 시작 코드 유무 설정	16진수 #0: 없음 16진수 #1: 있음
		11~08	무수순 모드 시 종료 코드 유무 설정	16진수 #0: 없음 16진수 #1: 있음 16진수 #2: CR+LF 지정
		07~00	수신 데이터 수	16진수 #00(초기값): 256바이트 16진수 #01~FF: 1~255바이트

기능

- S로 지정한 송신 데이터 시작 CH 번호부터 N으로 지정한 송신 바이트 수 분량의 데이터를 변환하지 않고 그대로 C의 비트 8~11로 지정한 CPU 유니트 내장 RS-232C 포트(무수순 모드) 또는 시리얼 커뮤니케이션 보드의 시리얼 포트(무수순 모드)에 출력합니다.
- PLC 시스템 설정(CPU 유니트 내장 포트 시) 또는 할당 DM 영역 설정(시리얼 커뮤니케이션 보드의 시리얼 포트 시)으로 송신 메시지의 프레임 형식을 다음과 같이 설정할 수 있습니다. 이 설정에서 시작 코드 및 종료 코드를 지정한 경우, 지정된 시작 및 종료 코드를 부가하여 송신합니다.

1)시작 코드 설정 유무(있는 경우, 코드 #00~FF)

2)종료 코드 설정 유무(있는 경우, 코드 #00~FF 또는 CR+LF)

· C의 비트 0~3으로 지정한 저장 순서 지정에 따른 순서로 송신합니다.

· C의 비트 4~7로 지정한 RS, ER 신호의 제어 지정은 다음과 같이 반영됩니다.

1)RS 신호의 제어를 지정한 경우, S의 비트 15의 내용을 RS 신호에 반영합니다.

2)ER 신호의 제어를 지정한 경우, S의 비트 15의 내용을 ER 신호에 반영합니다.

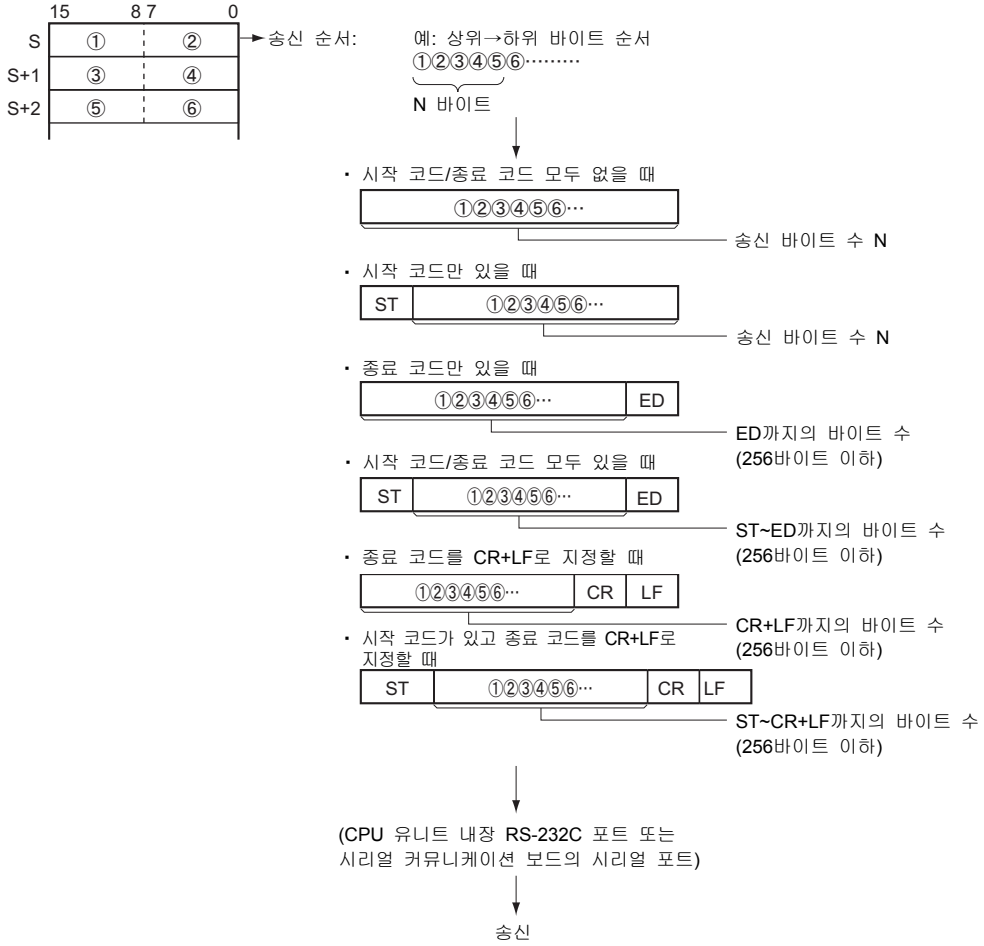
3)RS 신호 및 ER 신호의 제어를 지정한 경우, S의 비트 15의 내용을 RS 신호에, S의 비트 14의 내용을 ER 신호에 반영합니다.

RS, ER 신호의 제어 지정을 1~3Hex로 한 경우, 송신 준비 플래그(CPU 유니트 내장 포트: A392.05, 시리얼 커뮤니케이션 보드의 시리얼 포트 1: A356.05, 시리얼 포트 2: A356.13)와 관계 없이 명령을 실행합니다.

· 송신 가능한 바이트 수는 최대 259바이트(송신 데이터부: 최대 256바이트, 시작 코드부: 1바이트, 종료 코드부: CR+LF 지정 2바이트)입니다.

· N에는 시작 코드 및 종료 코드를 포함하지 않는 송신 데이터부의 크기를 지정하십시오.

■ 시작 코드/종료 코드 설정과 송신 데이터



힌트

· TXD 명령으로 데이터를 송신할 상대 기기에 따라 송신 간격을 설정해야 하는 경우가 있습니다. 이 경우, 송신 딜레이 시간을 설정하여 송신 간격을 조정하십시오.

사용상의 주의 사항

· 사이클릭 태스크와 인터럽트 태스크 양쪽에서 시리얼 커뮤니케이션 보드의 시리얼 포트(포트 1/2) 대상의 TXD 명령/RXD 명령을 기술하지 마십시오.

사이클릭 태스크로 시리얼 커뮤니케이션 보드의 시리얼 포트(포트 1/2) 대상의 TXD 명령/RXD 명령을 실행하는 중에 인터럽트 태스크를 기동한 경우, 해당 인터럽트 태스크 내에서 시리얼 커뮤니케이션 보드의 시

TXD

리얼 포트(포트 1/2) 대상의 TXD 명령/RXD 명령을 실행할 수 없습니다. 에러가 되어 인터럽트 태스크 내 TXD/RXD 명령 직후의 에러 플래그가 ON이 됩니다(사이클릭 태스크와 인터럽트 태스크 사이에 TXD 명령 /RXD 명령, 포트 1/2의 어떤 조합도 불가능).

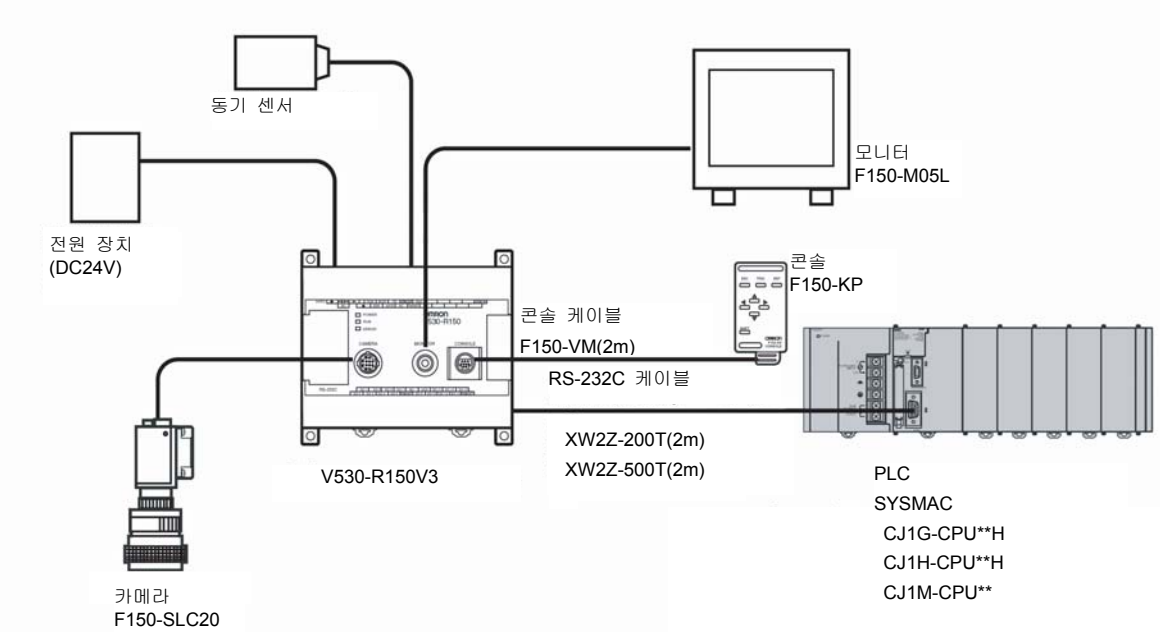
- 이 명령은 무수순 모드로 설정한 CPU 유니트 내장 RS-232C 포트 또는 시리얼 커뮤니케이션 보드의 시리얼 포트에만 사용할 수 있습니다.
- 특수 보조 릴레이의 송신 준비 플래그(CPU 유니트 내장 포트: A392.05, 시리얼 커뮤니케이션 보드의 시리얼 포트 1: A356.05, 시리얼 포트 2: A356.13)가 ON일 때만 송신 가능합니다.
- N에 0을 지정한 경우 송신하지 않습니다.
- 무수순 모드를 지원하지 않는 보드(유니트Ver. 표기가 없는 타입)에 대해 TXD 명령을 실행한 경우, INNER 보드 서비스 불가능 플래그(A424CH 비트 04: 운전 계속 이상) 및 에러 플래그가 ON이 됩니다.

샘플 프로그램

■ 데이터를 송신하는 경우

범용 외부 기기와 통신하는 예로, 2차원 코드 리더(V530-R150V3)에 데이터를 송신합니다.

- 하드웨어 구성



이 사례에서는 CPU 유니트 내장 RS-232C 포트에 연결했습니다.
먼저 코드 리더의 읽는 조건을 설정하십시오.

- 통신 설정

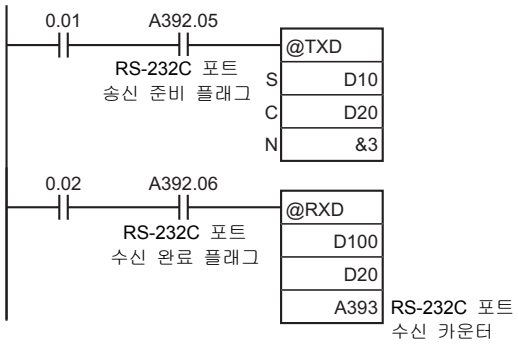
코드 리더의 통신 설정은 다음과 같은 공장 출하 시의 값으로 설정합니다.

통신 모드	무수순 모드
통신 속도	38400bps
데이터 비트 길이	8비트
패리티 확인	없음
정지 비트	1비트
시작 코드	없음
종료 코드	#000D(CR)

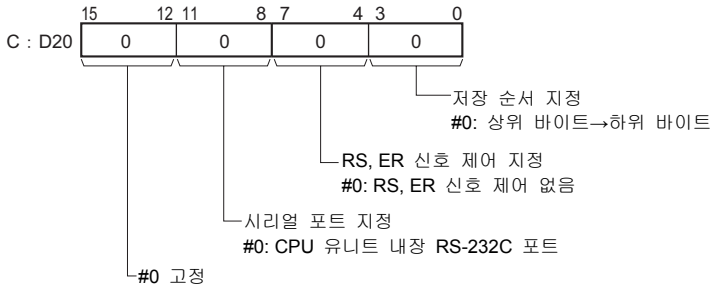
PLC 시스템 설정에서 PLC 측 통신 설정도 동일하게 구성하십시오.
이 경우, 종료 코드만 있는 설정이 됩니다.

· 프로그램 예

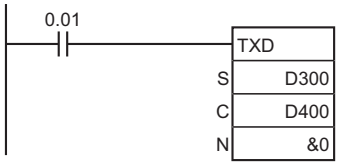
RS-232C 포트 송신 준비 플래그 A392.05가 ON일 때 0.01을 ON하면 D10의 상위 바이트에서 3바이트를 변환하지 않고 그대로 CPU 유니트 내장 RS-232C 포트에 연결된 코드 리더에 RS-232C로부터의 트리거 입력이 되는 '일반 읽기 명령' "@GL"을 송신합니다.



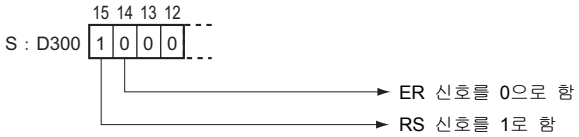


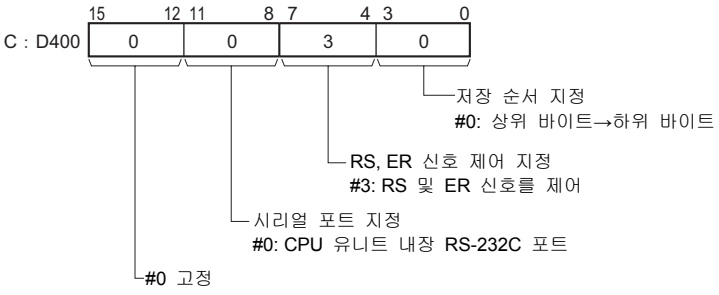


■ 신호 제어를 하는 경우



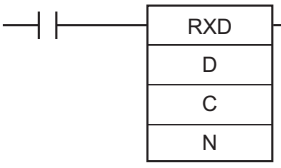
0.01이 ON이면 D300의 비트 15의 내용을 RS 신호에, 비트 14의 내용을 ER 신호에 반영합니다.





RXD

명령 이름	니모닉	동작 옵션	평션 번호	기능
CPU 유니트 시리얼 커뮤니케이션 보드 시리얼 포트 입력	RXD	@RXD	235	CPU 유니트 내장 RS-232C 포트(무수순 모드) 또는 시리얼 커뮤니케이션 보드 유니트 Ver.1.2 이상의 시리얼 통신 포트(무수순 모드)에서 PLC 시스템 설정의 무수순 모드일 때의 시작 코드/종료 코드 지정에 따라 수신한 데이터를 지정된 시작 CH 번호부터 지정된 바이트 수만큼 출력합니다.

심볼	RXD
	 D: 수신 데이터 저장 시작 CH 번호 C: 제어 데이터 N: 저장 바이트 수 10진수 &0~256(16진수 #0000~0100)

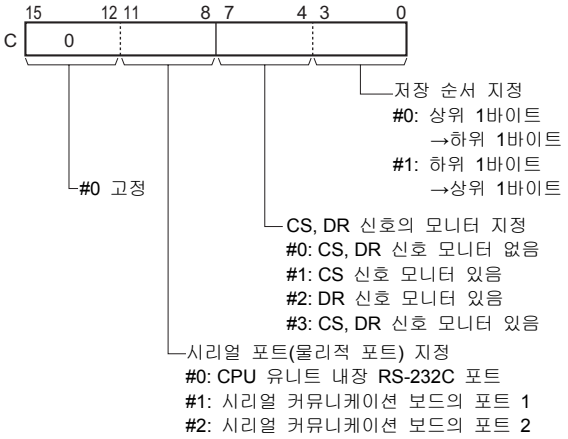
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
D	수신 데이터 저장 시작 CH 번호	UINT	가변
C	제어 데이터	UINT	1
N	저장 바이트 수 10진수 &0~256(16진수 #0000~0100)	UINT	1

C: 제어 데이터



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
D	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
C											○	○						
N																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· CPU 유니트 내장 RS-232C 포트 지정에서 PLC 시스템 설정이 무수순 모드가 아닐 때 ON · 시리얼 커뮤니케이션 보드의 시리얼 포트 지정 시, 할당 DM 영역 설정이 무수순 모드가 아닐 때 또는 시리얼 커뮤니케이션 보드가 유니트 Ver.이 없는 타입(무수순에 대응되지 않는 타입)일 때 ON · C의 데이터가 범위 밖일 때 ON · N의 데이터가 10진수 &0~256 또는 16진수 #0000~0100의 범위에 없을 때 ON · CS1D CPU 유니트에서 활성-대기 사이에서 동기화되지 않았을 때 ON · 사이클릭 태스크에서 시리얼 커뮤니케이션 보드의 포트를 지정한 TXD/RXD 명령 실행 중에 인터럽트가 발생하여 인터럽트 태스크가 기동하고, 인터럽트 태스크 내에서도 시리얼 커뮤니케이션 보드의 포트를 지정한 TXD/RXD 명령을 실행했을 때 ON(인터럽트 태스크 내 TXD/RXD 명령의 에러 플래그) · 시리얼 커뮤니케이션 보드가 재시작 중에 시리얼 커뮤니케이션 보드의 시리얼 포트 지정에서 RXD 명령을 실행했을 때 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

■ CPU 유니트 내장 RS-232C 포트 사용 시

이름	주소	내용
RS-232C 포트 수신 완료 플래그	A392.06	무수순 모드에서 수신을 완료했을 때 1(ON)이 됩니다. · 수신 바이트 수 지정 시: 지정한 바이트 수를 수신하면 ON · 인코드 지정 시: 인코드 수신 또는 256바이트 수신에서 ON
RS-232C 포트 수신 초과 플래그	A392.07	무수순 모드일 때 수신 데이터 수를 초과하여 수신한 경우 1(ON)이 됩니다. · 수신 바이트 수 지정 시: 수신 완료 후 RXD 명령을 실행하기 전에 데이터를 수신하면 ON · 인코드 지정 시: 인코드 수신 후 RXD 명령을 실행하기 전에 데이터를 수신하면 ON 인코드를 수신하지 않는 경우 256바이트 수신 후 257바이트째가 인코드가 아닐 때 ON
RS-232C 포트 수신 카운터	A393.00~393.15	무수순 모드일 때 수신한 데이터의 바이트 수를 16진수로 표시합니다.

■ 시리얼 커뮤니케이션 보드의 시리얼 포트 사용 시

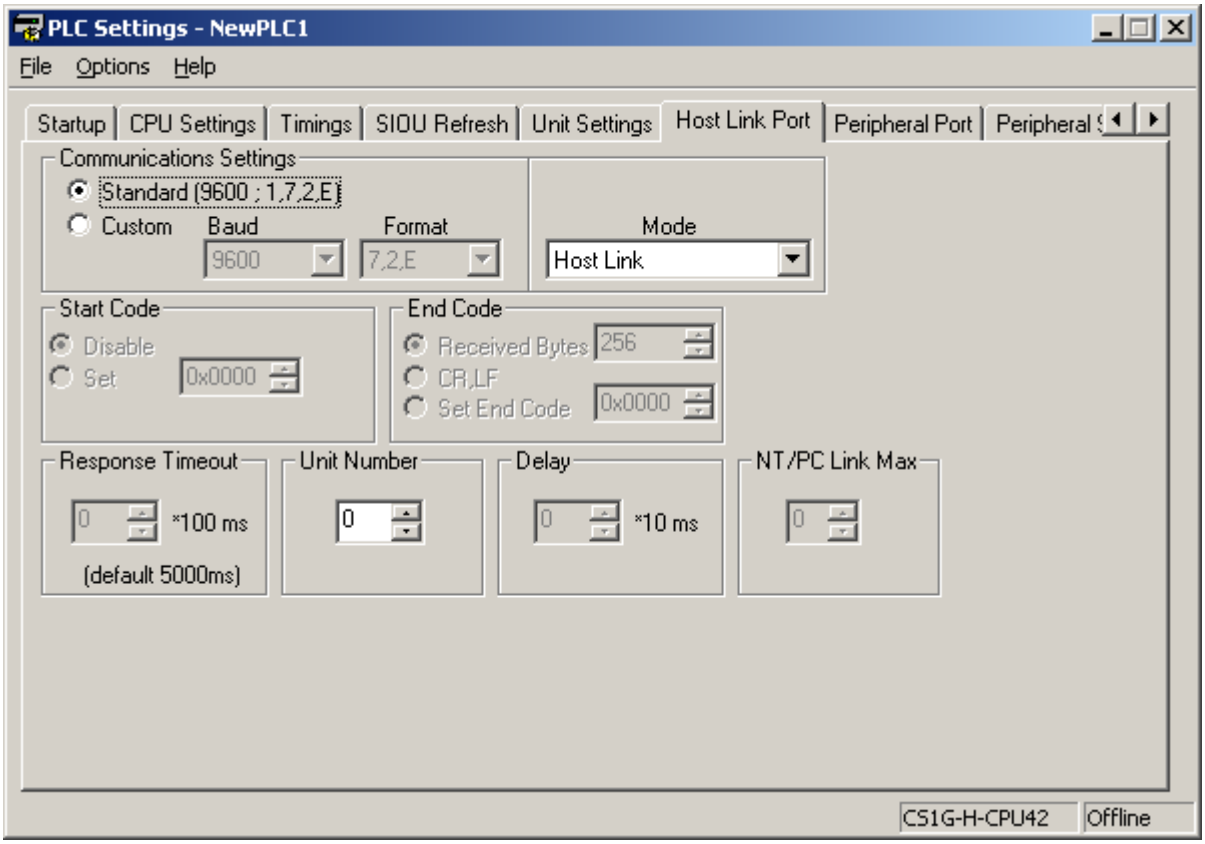
이름		주소	내용
포트 1	수신 완료 플래그	A356.06	무수순 모드에서 수신을 완료했을 때 1(ON)이 됩니다. · 수신 바이트 수 지정 시: 지정한 바이트 수를 수신하면 ON · 인코드 지정 시: 인코드 수신 또는 256바이트 수신에서 ON
	수신 초과 플래그	A356.07	무수순 모드일 때 수신 데이터 수를 초과하여 수신한 경우 1(ON)이 됩니다. · 수신 바이트 수 지정 시: 수신 완료 후 RXDU 명령으로 데이터를 읽기 전에 다음 데이터를 수신했을 때 ON · 인코드를 수신하지 않는 경우 256바이트 수신 후 257바이트째가 인코드가 아닐 때 ON
	수신 카운터	A357.00~A357.15	무수순 모드일 때 시리얼 포트가 수신한 데이터의 바이트 수(0~256)를 16진수로 표시합니다.
	오버런 에러	1908CH 비트 04	RXD 명령 실행 전에 수신 버퍼에 260바이트 이상의 데이터를 수신했을 때 ON
포트 2	수신 완료 플래그	A356.14	무수순 모드에서 수신을 완료했을 때 1(ON)이 됩니다. · 수신 바이트 수 지정 시: 지정한 바이트 수를 수신하면 ON · 인코드 지정 시: 인코드 수신 또는 256바이트 수신에서 ON
	수신 초과 플래그	A356.15	무수순 모드일 때 수신 데이터 수를 초과하여 수신한 경우 1(ON)이 됩니다. · 수신 바이트 수 지정 시: 수신 완료 후 RXDU 명령으로 데이터를 읽기 전에 다음 데이터를 수신했을 때 ON · 인코드를 수신하지 않는 경우 256바이트 수신 후 257바이트째가 인코드가 아닐 때 ON
	수신 카운터	A358.00~A358.15	무수순 모드일 때 시리얼 포트가 수신한 데이터의 바이트 수(0~256)를 16진수로 표시합니다.

RXD

이름		주소	내용
	오버런 에러	1918CH 비트 04	RXD 명령 실행 전에 수신 버퍼에 260바이트 이상의 데이터를 수신했을 때 ON
시리얼 포트 1/2	INNER 보드 서비스 불가능 플래그	A424CH 비트 04	무수순 모드를 지원하지 않는 보드(유니트 Ver. 표기가 없는 타입)에 대해 RXD 명령을 실행했을 때 ON
시리얼 포트 1/2	INNER 보드 서비스 불가능 플래그	A424CH 비트 04	무수순 모드를 지원하지 않는 보드(유니트 Ver. 표기가 없는 타입)에 대해 RXD 명령을 실행했을 때 ON

관련 PLC 시스템 설정(CPU 유니트 내장 RS-232C 포트 사용 시)

■ 시스템 설정(CX-Programmer의 경우)



■ PLC 시스템 설정(프로그래밍 콘솔의 경우)

프로그래밍 콘솔 주소		내용	설정
채널	비트		
162	0~15	무수순 모드 시 송신 딜레이 시간	16진수 #0000~210F: 10진수 0~99990ms(10ms 단위)
164	8~15	무수순 모드 시 시작 코드	16진수 #00~FF
	0~7	무수순 모드 시 종료 코드	16진수 #00~FF
165	12	무수순 모드 시 시작 코드 유무 설정	16진수 #0: 없음 16진수 #1: 있음
	8~9	무수순 모드 시 종료 코드 유무 설정	16진수 #0: 없음 16진수 #1: 있음 16진수 #2: CR+LF 지정
	0~7	무수순 모드 시 송신 데이터 수 설정	16진수 #00(초기값): 256 16진수 #01~FF: 10진수 1~255

■ 할당 DM 설정 영역(시리얼 커뮤니케이션 보드의 시리얼 포트 사용 시)

이름		비트	내용	설정
포트 1	포트 2			
D32002	D32012	15	무수순 모드 시 송신 딜레이 시간	0: 기본값(0ms) 1: 임의 설정
		14~00	무수순 모드 시 송신 딜레이 임의 설정 시간	16진수 #0000~7530: 10진수 0~300000ms [10ms 단위]
D32004	D32014	15~08	무수순 모드 시 시작 코드	16진수 #00~FF
		07~00	무수순 모드 시 종료 코드	16진수 #00~FF
D32005	D32015	15~12	무수순 모드 시 시작 코드 유무 설정	16진수 #0: 없음 16진수 #1: 있음
		11~08	무수순 모드 시 종료 코드 유무 설정	16진수 #0: 없음 16진수 #1: 있음 16진수 #2: CR+LF 지정
		07~00	수신 데이터 수	16진수 #00(초기값): 256바이트 16진수 #01~FF: 1~255바이트

기능

- C의 비트 8~11로 지정한 CPU 유니트 내장 RS-232C 포트(무수순 모드) 또는 시리얼 커뮤니케이션 보드의 시리얼 포트(무수순 모드)에서 수신 완료 데이터를 D로 지정한 수신 데이터 저장 시작 CH 번호부터 N으로 지정한 저장 바이트 수만큼 저장합니다. 수신 완료 데이터가 N으로 지정한 저장 바이트 수만큼 존재하지 않는 경우는 존재하는 수신 완료 데이터만큼 저장합니다.
- PLC 시스템 설정(CPU 내장 포트 시) 또는 할당 DM 영역 설정(시리얼 커뮤니케이션 보드의 시리얼 포트 시)을 통해 수신 메시지의 프레임 형식을 다음과 같이 설정할 수 있습니다.

  1)시작 코드 설정 유무(있는 경우, 코드 #00~FF)

  2)종료 코드 설정 유무(있는 경우, 코드 #00~FF 또는 CR+LF, 없는 경우 수신 데이터 수 #00~FF: 1~256, #00은 256을 나타냅니다.)
- C의 비트 0~3으로 지정한 저장 순서 지정에 따른 순서로 수신 데이터를 D 이후에 저장합니다.

- 수신 완료 플래그가 ON이 되는 경우

  1)PLC 시스템 설정(CPU 내장 포트 시) 또는 할당 DM 영역 설정(시리얼 커뮤니케이션 보드의 시리얼 포트 시)을 통해 지정한 수신 바이트 수를 수신했을 때 수신 완료 플래그(*1)가 ON이 됩니다. 이때, 수신 카운터(*2)는 PLC 시스템 설정으로 지정된 무수순 모드 시의 수신 데이터 수와 같아집니다.

  2)PLC 시스템 설정(CPU 내장 포트 시) 또는 할당 DM 영역 설정(시리얼 커뮤니케이션 보드의 시리얼 포트 시)을 통해 종료 코드를 지정한 경우, 해당 종료 코드를 수신했을 때 수신 완료 플래그(*1)가 ON이 됩니다.

  3)256바이트의 데이터를 수신하면 수신 완료 플래그(*1)가 ON이 됩니다.
  수신 완료 플래그(*1)가 ON인 상태에서 무수순 수신 명령을 실행하여 데이터를 수신하면 수신 초과 플래그(*3)가 ON이 됩니다.
- 데이터를 259바이트 수신하면 수신을 정지합니다. 그 이후의 수신이 있으면 오버런 에러 플래그(*5) 및 전송 에러 발생 플래그(*6)가 1(ON)이 됩니다. CPU 유니트 내장 시리얼 포트에서 오버런 에러나 패리티 에러가 발생한 경우, 수신이 정지됩니다. 수신을 시작하기 위해서는 시리얼 포트를 재기동해야 합니다.
- 지정 바이트 수 이상의 수신 데이터는 시리얼 커뮤니케이션 보드의 시리얼 포트 대상인 경우, RXD 명령 실행 시 파기됩니다. 그에 비해 CPU 유니트 내장 RS-232C 포트 대상인 경우에는 RXD 명령 실행 시 파기되지 않고 수신 버퍼에 남아 있습니다.
- 이 RXD 명령을 실행하면 수신 버퍼에서 D에 데이터가 저장되고 수신 완료 플래그(*1)가 OFF가 되며 수신 카운터(*2)는 0이 됩니다. 또한 수신 초과 플래그가 ON인 경우는 OFF가 됩니다.
- CPU 유니트 내장 RS-232C 포트의 경우, RS-232C 포트 재시작 플래그(*4)를 ON으로 했을 때도 수신 완료 플래그(*1)가 OFF가 되고 수신 카운터(*2)는 0이 됩니다. 이때, 수신 초과 플래그가 ON인 경우는 OFF가 됩니다.
- C의 비트 4~7로 지정한 CS, DR 신호의 모니터 지정은 다음과 같이 반영됩니다.

  1)CS 신호의 모니터를 지정한 경우, CS 신호의 상태를 D로 지정한 채널의 비트 15에 출력합니다.

  2)DR 신호의 모니터를 지정한 경우, DR 신호의 상태를 D로 지정한 채널의 비트 15에 출력합니다.

  3)CS 신호 및 DR 신호의 모니터를 지정한 경우, CS 신호의 상태를 D로 지정한 채널의 비트 15에, DR 신호의 상태를 D로 지정한 채널의 비트 14에 출력합니다.
- CS, DR 신호의 모니터 지정을 #1~3으로 한 경우, 수신 완료 플래그(*1)와 관계 없이 데이터를 읽습니다. CS 신호 또는 DR 신호 모니터를 지정한 경우 수신 데이터의 저장 동작은 수행하지 않습니다.
- 수신 가능한 바이트 수는 최대 259바이트(수신 데이터부: 최대 256바이트, 시작 코드부: 1바이트, 종료 코드부: CR+LF 지정 2바이트)입니다.

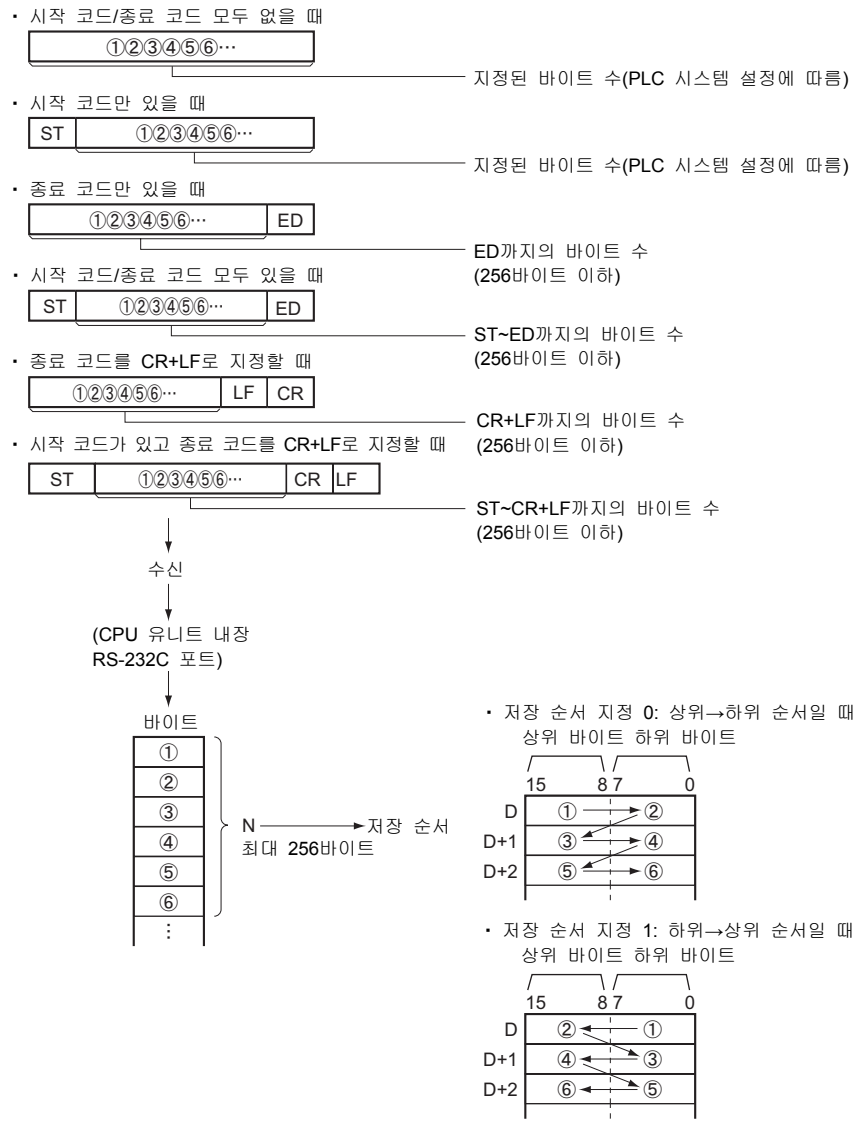
RXD

· N에는 시작 코드 및 종료 코드를 포함하지 않는 수신 데이터부의 크기를 지정하십시오.

관련 특수 보조 릴레이/할당 릴레이의 주소

*1: 수신 완료 플래그	내장 RS-232C	A392CH 비트 06
	포트 1	A356CH 비트 06
	포트 2	A356CH 비트 14
*2: 수신 카운터	내장 RS-232C	A393CH
	포트 1	A357CH
	포트 2	A358CH
*3: 수신 초과 플래그	내장 RS-232C	A392CH 비트 07
	포트 1	A356CH 비트 07
	포트 2	A356CH 비트 15
*4: RS-232C 포트 재시작 플래그	내장 RS-232C	A526CH 비트 00
*5: 오버런 에러	포트 1	1908CH 비트 04
	포트 2	1918CH 비트 04
*6: 전송 에러 발생	포트 1	1908CH 비트 15
	포트 2	1918CH 비트 15
*7: INNER 보드 서비스 불가능 플래그	포트 1/2	A424CH 비트 04

■ 시작 코드/종료 코드 설정과 수신 데이터



힌트

· CPU 유니트 내장 시리얼 포트에 대한 RXD 명령의 경우, RXD 명령 실행 후 수신 버퍼를 클리어하지 않습니다. 따라서 복수의 RXD 명령을 사용한 분할 수신이 가능합니다.
그에 비해 시리얼 커뮤니케이션 보드(유니트 Ver.1.2 이상)의 시리얼 포트에 대한 RXD 명령의 경우, RXD 명령 실행 후 수신 버퍼가 클리어됩니다. 따라서 복수의 RXD 명령을 사용한 분할 수신은 불가능하므로 주의하십시오.

사용상의 주의 사항

· 사이클릭 태스크와 인터럽트 태스크 양쪽에서 시리얼 커뮤니케이션 보드의 시리얼 포트(포트 1/2) 대상의 TXD 명령/RXD 명령을 기술하지 마십시오.
사이클릭 태스크로 시리얼 커뮤니케이션 보드의 시리얼 포트(포트 1/2) 대상의 TXD 명령/RXD 명령을 실행하는 중에 인터럽트 태스크를 기동한 경우, 해당 인터럽트 태스크 내에서 시리얼 커뮤니케이션 보드의 시리얼 포트(포트 1/2) 대상의 TXD 명령/RXD 명령을 실행할 수 없습니다. 에러가 되어 인터럽트 태스크 내 TXD/RXD 명령 직후의 에러 플래그가 ON이 됩니다(사이클릭 태스크와 인터럽트 태스크 사이에 TXD 명령/RXD 명령, 포트 1/2의 어떤 조합도 불가능).

· 무수순 모드로 설정한 CPU 유니트 내장 RS-232C 포트 또는 시리얼 커뮤니케이션 보드의 시리얼 포트에만 사용할 수 있습니다.

· 수신 완료 플래그(CPU 유니트 내장 RS-232C: A392.06, 시리얼 커뮤니케이션 보드의 포트 1: A356.06, 포트 2: A356.14)가 1(ON)일 때 이 명령을 실행하고 데이터를 수신하도록 하십시오(수신 버퍼에서).

· 데이터를 수신할 때 RXD 명령으로 수신 데이터를 읽지 않으면 다음 데이터를 수신할 수 없습니다.
수신 완료 플래그가 ON이 되면 다음 수신까지 RXD 명령으로 수신 데이터를 읽으십시오.

· CPU 유니트 내장 시리얼 포트에 대한 RXD 명령의 경우, RXD 명령 실행 후 수신 버퍼를 클리어하지 않습니다. 따라서 복수의 RXD 명령을 사용한 분할 수신이 가능합니다.
그에 비해 시리얼 커뮤니케이션 보드(유니트 Ver.1.2 이상)의 시리얼 포트에 대한 RXD 명령의 경우, RXD 명령 실행 후 수신 버퍼가 클리어됩니다. 따라서 복수의 RXD 명령을 사용한 분할 수신은 불가능하므로 주의하십시오.

· N에 0을 지정한 경우 D 이후에 데이터를 전송하지 않습니다. 이 경우, 수신 완료 플래그(*1)와 수신 초과 플래그(*3)가 OFF되고 수신 카운터(*2)는 0이 됩니다.

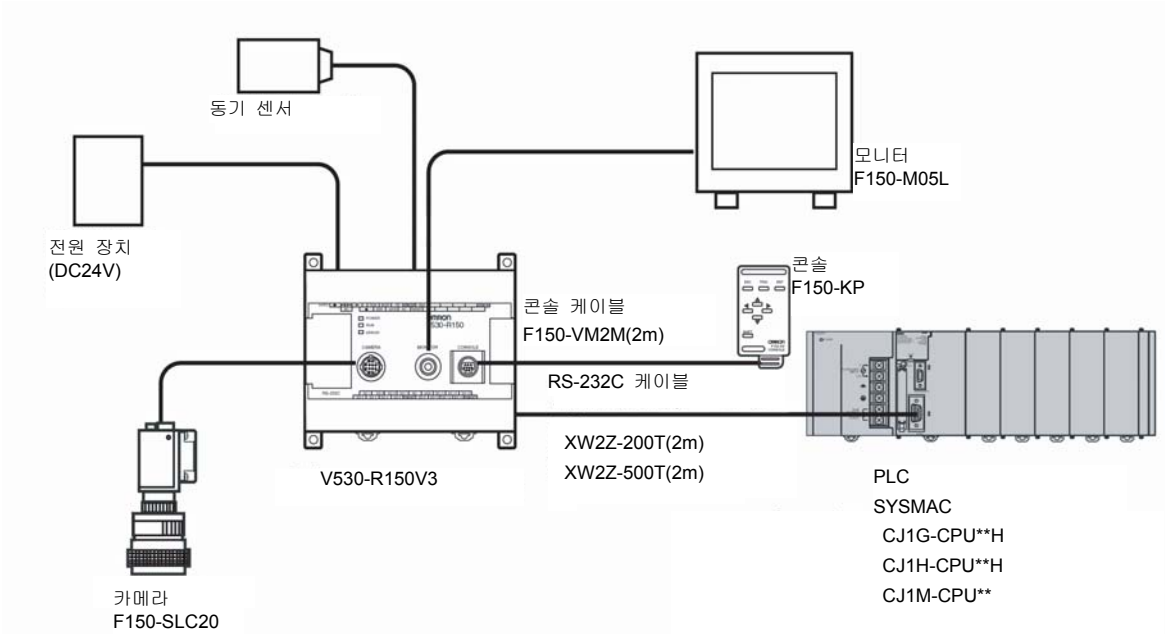
· 무수순 모드를 지원하지 않는 보드(유니트 Ver. 표기가 없는 타입)에 대해 RXD 명령을 실행한 경우, INNER 보드 서비스 불가능 플래그(운전 계속 이상 *7) 및 에러 플래그가 ON이 됩니다.

샘플 프로그램

■ 데이터를 수신하는 경우

범용 외부 기기와 통신하는 예로, 2차원 코드 리더(V530-R150V3)에서 데이터를 수신합니다.

· 하드웨어 구성



이 사례에서는 CPU 유니트 내장 RS-232C 포트에 연결했습니다.
먼저 코드 리더의 읽는 조건을 설정하십시오.

RXD

· 통신 설정

코드 리더의 통신 설정은 다음과 같은 공장 출하 시의 값으로 설정합니다.

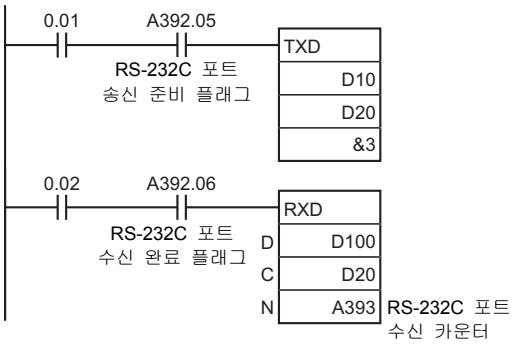
통신 모드	무수순 모드
통신 속도	38400bps
데이터 비트 길이	8비트
패리티 확인	없음
정지 비트	1비트
시작 코드	없음
종료 코드	#000D(CR)

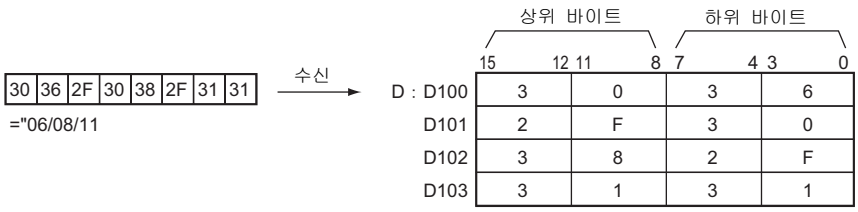
PLC 시스템 설정에서 PLC의 통신 설정도 동일하게 구성하십시오.

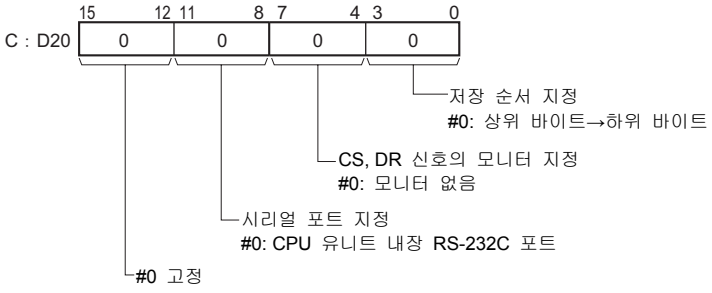
이 경우, 종료 코드만 있는 설정이 됩니다.

· 프로그램 예

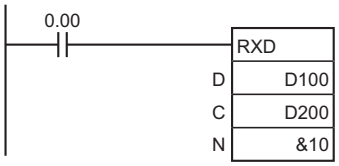
RS-232C 포트 수신 완료 플래그 A392.06이 ON일 때 0.02를 ON하면 CPU 유니트 내장 RS-232C 포트에 연결된 코드 리더에서 수신하여 읽은 결과를 D100의 상위 바이트에 RS-232C 포트 수신 카운터 A393CH의 바이트 수만큼 저장합니다.



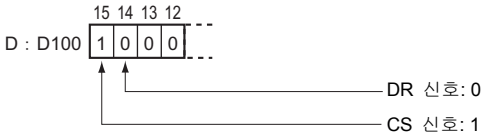


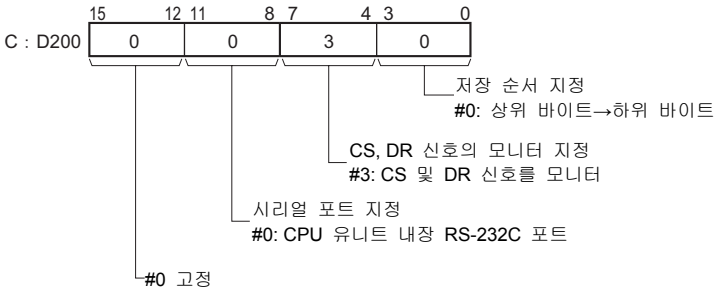


■ 신호 모니터를 하는 경우



0.00이 ON이면 CS 신호를 D100의 비트 15에, DR 신호를 비트 14에 반영합니다.

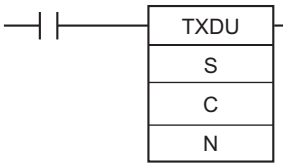




TXDU

명령 이름	니모닉	동작 옵션	평션 번호	기능
시리얼 커뮤니케이션 유니트 시리얼 포트 출력	TXDU	@TXDU	256	지정된 바이트 수 분량의 데이터를 변환하지 않고 그대로 할당 DM 영역 설정의 무수순 모드일 때의 시작 코드/종료 코드 지정에 따라 시리얼 커뮤니케이션 유니트의 유니트 Ver.1.2 이상의 시리얼 통신 포트(무수순 모드)에서 송신합니다.

심볼	TXDU



S: 송신 데이터 시작 CH 번호
C: 제어 데이터 시작 CH 번호
N: 송신 바이트 수
10진수 &0~256(16진수 #0000~0100)

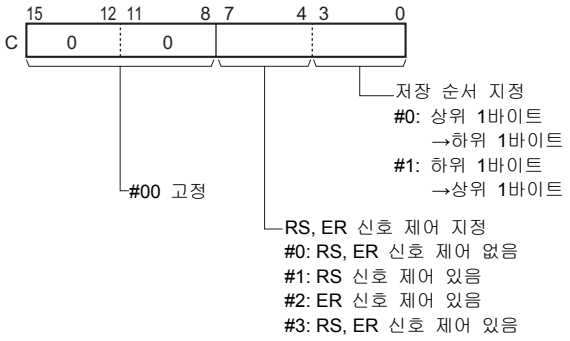
특정 영역 내의 사용 여부

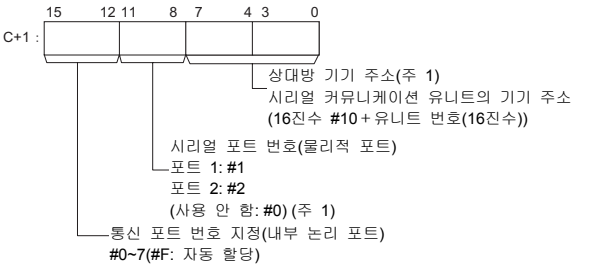
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	송신 데이터 시작 CH 번호	UINT	가변
C	제어 데이터 시작 CH 번호	UDINT	2
N	송신 바이트 수 10진수 &0~256(16진수 #0000~0100)	UINT	1

C: 제어 데이터 시작 CH 번호





주 1: 시리얼 포트 번호(물리적 포트)에 #0을 지정하고 상대방 기기 주소에 시리얼 포트의 기기 주소(포트 1: 16진수 #80+16진수 #04×유니트 번호(16진수), 포트 2: 16진수 #81+16진수 #04×유니트 번호(16진수))를 직접 지정할 수도 있습니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
C											○							
N												○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 명령 실행 시, 지정된 통신 포트(내부 논리 포트)의 네트워크 통신 명령 실행 가능 플래그가 OFF되었을 때 또는 통신 포트를 모두 사용 중일 때 ON · C의 데이터가 범위 밖일 때 ON · N의 데이터가 10진수 &0~256 또는 16진수 #0000~0100의 범위에 없을 때 ON · 인터럽트 태스크에서 실행했을 때 ON(CJ2 CPU 유니트에서 인터럽트 고속 기동 기능 사용 시) · 그 이외일 때 OFF

할당 DM 설정 영역

■ m=D30000+100×유니트 번호 CH

이름		비트	내용	설정
포트 1	포트 2			
m+2	m+12	15	무수순 모드 시 송신 딜레이 시간	기본값(0ms) 1: 임의 설정
		14~00	무수순 모드 시 송신 딜레이 임의 설정 시간	16진수 #0000~7530(10진수 0~300,000ms) [10ms 단위]
m+4	m+14	15~08	무수순 모드 시 시작 코드	16진수 #00~FF
		07~00	무수순 모드 시 종료 코드	16진수 #00~FF
m+5	m+15	15~12	무수순 모드 시 시작 코드 유무 설정	16진수 #0: 없음 16진수 #1: 있음
		11~08	무수순 모드 시 종료 코드 유무 설정	16진수 #0: 없음 16진수 #1: 있음 16진수 #2: CR+LF 지정
		07~00	수신 데이터 수	16진수 #00(초기값): 256바이트 16진수 #01~FF: 1~256바이트

관련 특수 보조 릴레이

이름	주소	내용
네트워크 통신 명령 실행 가능 플래그	A202.00~ A202.07	네트워크 통신(TXDU 명령 포함)이 실행 가능할 때 1(ON)이 됩니다. 각 비트가 통신 포트(내부 논리 포트) 번호에 대응합니다. 비트 00~07: 통신 포트 번호 00~07 네트워크 중에 0(OFF), 실행 종료(정상이든 이상이든) 후에는 1(ON)이 됩니다.
네트워크 통신 응답 코드	A203CH~ A210CH	네트워크 통신이 실행된 경우 응답 코드(종료 코드)가 저장됩니다. 각 CH는 통신 포트(내부 논리 포트) 번호에 대응합니다. A203~210CH: 통신 포트 번호 00~07 통신 명령 실행 중에는 #0000이 되고 통신 명령 실행이 종료되면 반영됩니다. 이 값은 운전을 시작할 때 클리어됩니다.
네트워크 통신 실행 에러 플래그	A219.00~ A219.07	네트워크 통신 실행 시 에러(이상)가 발생한 경우 1(ON)이 됩니다. 각 비트가 통신 포트(내부 논리 포트) 번호에 대응합니다. 비트 00~07: 통신 포트 번호 00~07 다음 네트워크 통신 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 통신 명령 실행에서 0(OFF)이 됩니다.

통신 응답 코드 목록

코드(16진수)	내용
#0401	정의되지 않은 명령(시리얼 통신 모드가 프로토콜 매크로, NT 링크, 루프백 테스트 모드, 시리얼 게이트웨이 중 하나인 경우 등)
#0205	응답 시간 초과(시리얼 통신 모드가 상위 링크 모드인 경우 등)
#1001	명령 길이가 최대 명령 길이를 초과
#1002	명령 길이가 최소 명령 길이를 만족하지 못 함
#1003	지정된 데이터 수와 송신 데이터 수가 일치하지 않음
#1004	명령 형식이 다름
#110C	그 밖의 파라미터 에러
#2201	실행 중이므로 동작 불가능(송신 중이므로 동작 불가능)
#2202	정지 중이므로 실행 불가능(프로토콜 전환 중이기 때문에 실행 불가능)

관련 **CPU** 고기능 유니트 릴레이 영역

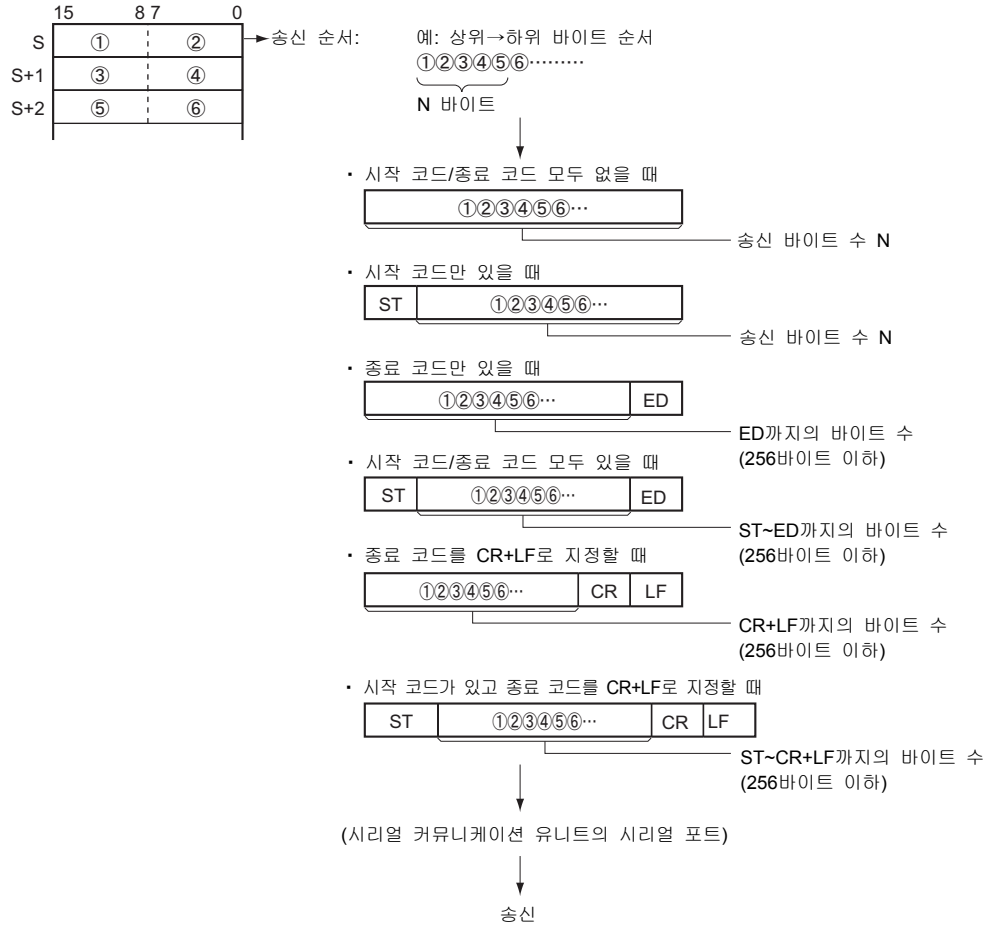
■ n = 1500 + 25×유니트 번호

이름		비트	내용
포트 1	포트 2		
n+9	n+19	05	TXDU 명령 실행 중 플래그 1: 실행 중, 0: 비실행 중

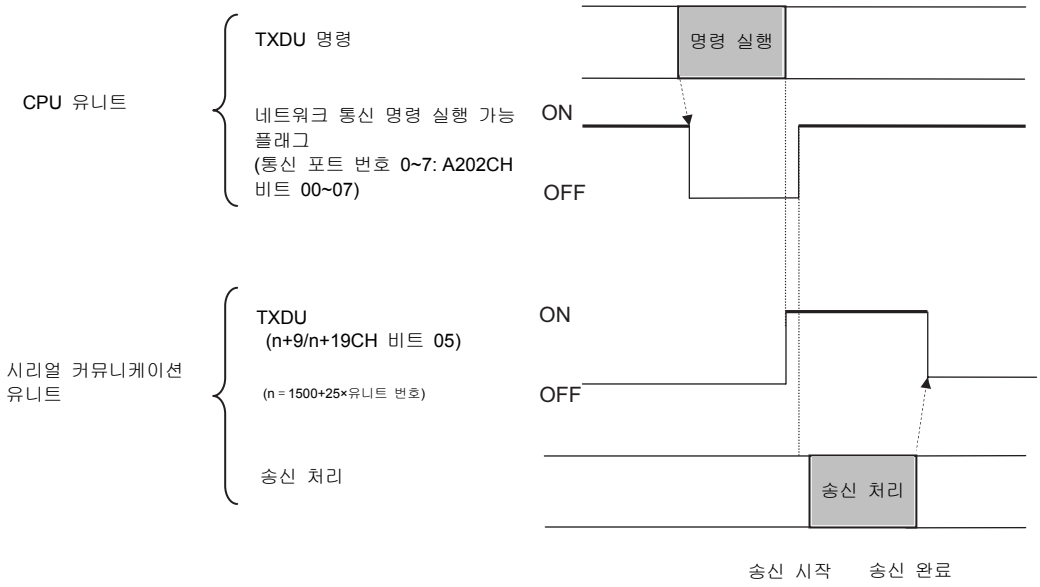
기능

- S로 지정한 송신 데이터 시작 CH 번호부터 N으로 지정한 송신 바이트 수 분량의 데이터를 변환하지 않고 그대로 C+1의 비트 0~7로 지정한 기기 주소의 시리얼 커뮤니케이션 유니트의 C+1의 비트 8~11로 지정한 시리얼 포트(무수순 모드)에 출력합니다.
- 할당 DM 영역 설정에서 송신 메시지의 프레임 형식을 다음과 같이 설정할 수 있습니다. 이 설정에서 시작 코드 및 종료 코드를 지정한 경우, 지정된 시작 및 종료 코드를 부가하여 송신합니다.
  1) 시작 코드 설정 유무(있는 경우, 코드 16진수 #00~FF)
  2) 종료 코드 설정 유무(있는 경우, 코드 16진수 #00~FF 또는 CR+LF)
- C의 비트 0~3으로 지정한 저장 순서 지정에 따른 순서로 송신합니다.
- C의 비트 4~7로 지정한 RS, ER 신호의 제어 지정은 다음과 같이 반영됩니다.
  1) RS 신호의 제어를 지정한 경우, S의 비트 15의 내용을 RS 신호에 반영합니다.
  2) ER 신호의 제어를 지정한 경우, S의 비트 15의 내용을 ER 신호에 반영합니다.
  3) RS 신호 및 ER 신호의 제어를 지정한 경우, S의 비트 15의 내용을 RS 신호에, S의 비트 14의 내용을 ER 신호에 반영합니다.
- 송신 가능한 바이트 수는 최대 259바이트(송신 데이터부: 최대 256바이트, 시작 코드부: 1바이트, 종료 코드부: CR+LF 지정 2바이트)입니다.
- N에는 시작 코드 및 종료 코드를 포함하지 않는 송신 데이터부의 크기를 지정하십시오.
- C+1의 비트 12~15로 지정한 통신 포트(내부 논리 포트) 0~7 중 하나를 사용합니다. 또한 통신 포트(내부 논리 포트) 번호를 자동 할당("F")으로 사용하는 경우에 대해서는 3-736의 '내부 논리 포트 자동 할당 기능'의 설명을 참조하십시오.

■ 시작 코드/종료 코드 설정과 송신 데이터



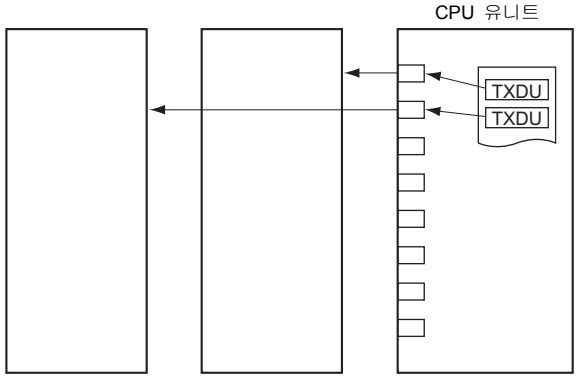
■ 송신 실행 플래그의 동작



힌트

· TXDU 명령으로 데이터를 송신하는 경우 범용 외부 기기에 따라 송신 간격을 설정해야 합니다. 이 경우, 할당 DM 영역 설정을 통해 송신 딜레이 시간을 설정하여 송신 간격을 조정하십시오.

TXDU



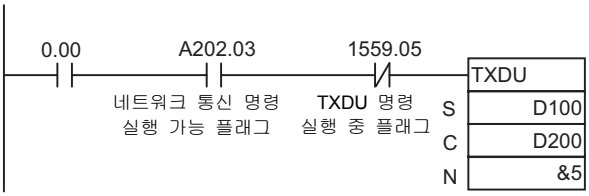
사용상의 주의 사항

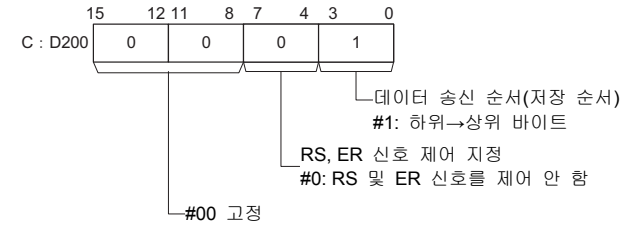
- 이 명령은 무수순 모드로 설정한 시리얼 커뮤니케이션 유니트의 시리얼 포트에만 사용할 수 있습니다.
- 사용하는 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그(A202.00~202.07)가 1(ON)이고, TXDU 명령 실행 중 플래그(할당 DM 영역 설정)가 0(OFF)일 때만 송신 가능합니다.
- N에 0을 지정한 경우 송신하지 않습니다.
- TXDU 명령(내부적으로는 FINS 명령을 전송했기 때문)은 네트워크 통신 명령의 SEND/RECV/CMND 명령 및 시리얼 통신 명령의 PMCR/RXDU 명령과 동일하게 통신 포트(내부 논리 포트) 0~7을 사용하여 시리얼 커뮤니케이션 유니트(유니트 Ver.1.2 이상)에 대해 송신 시퀀스 실행 지시를 내립니다. 따라서 네트워크 통신 명령 또는 시리얼 통신 명령의 PMCR/RXDU 명령이나 다른 TXDU 명령에 의해 통신 포트(내부 논리 포트)가 사용될 때는 해당 통신 포트를 사용하여 TXDU 명령을 실행할 수 없습니다.

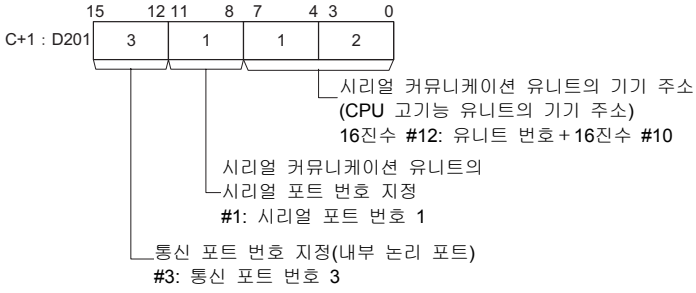
  그러므로 TXDU 명령의 입력 조건에는 C1의 비트 12~15로 지정한 사용 통신 포트 번호에 대응한 특수 보조 릴레이의 '네트워크 통신 실행 가능 플래그'(A202.00~A202.07)를 a 접점으로 삽입하십시오.
- TXDU 명령은 TXDU 명령 실행 플래그(n+9/n+19CH 비트 05)가 1(ON)인 동안은 실행할 수 없습니다(n = 1500+25×유니트 번호).
  따라서 TXDU 명령의 입력 조건에는 TXDU 명령 실행 중 플래그를 b 접점으로 삽입하십시오.

샘플 프로그램

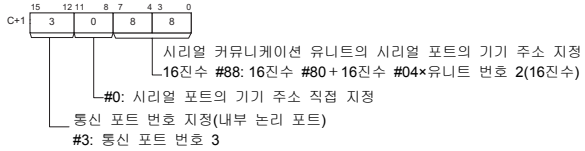
0.00이 ON이고 네트워크 통신 명령 실행 가능 플래그 A202.03이 ON이며 TXDU 명령 실행 중 플래그 1559.05가 OFF인 경우 통신 포트(내부 논리 포트) 번호 3에서 D100의 하위 바이트부터 5바이트를 변환하지 않고 그대로 유니트 번호 2의 시리얼 커뮤니케이션 유니트의 시리얼 포트 번호 1에 연결된 범용 외부 기기(예: 프린터 등)에 데이터를 출력합니다.

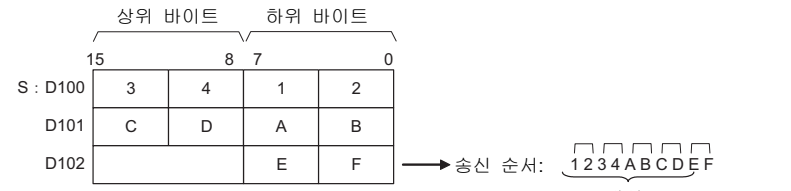




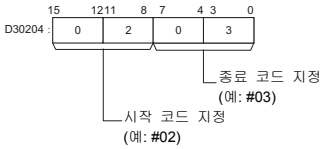


참고:
C+1의 비트 8~11의 시리얼 포트 번호(물리적 포트)에 0을 지정하고, C+1의 비트 0~7에 시리얼 포트의 기기 주소(포트 1: 16진수 #80+16진수 #04×유니트 번호(16진수), 포트 2: 16진수 #81+16진수 #04×유니트 번호(16진수))를 직접 지정할 수도 있습니다.

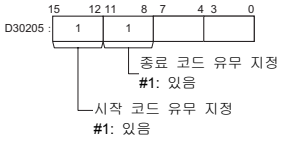




참고:
할당 DM 영역 설정
・ 시작 코드/종료 코드 지정



・ 시작 코드/종료 코드 유무 지정



시작 코드/종료 코드 모두 있음으로 지정했을 때(할당 DM 영역)

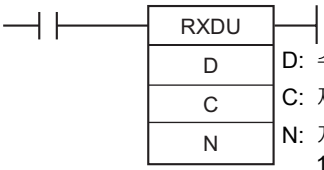
ST	12	34	AB	CD	EF	ED

ST: 시작 코드, 예: #02
ED: 종료 코드, 예: #03

데이터 송신

RXDU

명령 이름	니모닉	동작 옵션	평션 번호	기능
시리얼 커뮤니케이션 유니트 시리얼 포트 입력	RXDU	@RXDU	255	시리얼 커뮤니케이션 유니트의 유니트 Ver.1.2 이상의 시리얼 통신 포트에서 할당 DM 영역 설정의 무수순 모드일 때의 시작 코드/종료 코드 지정에 따라 수신한 데이터를 지정된 시작 CH 번호부터 지정된 바이트 수만큼 출력합니다.

	RXDU
심볼	 D: 수신 데이터 저장 시작 CH 번호 C: 제어 데이터 시작 CH 번호 N: 저장 바이트 수 10진수 &0~255(16진수 #0000~0100)

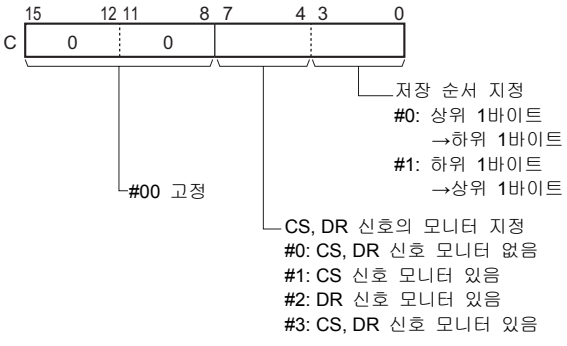
특정 영역 내의 사용 여부

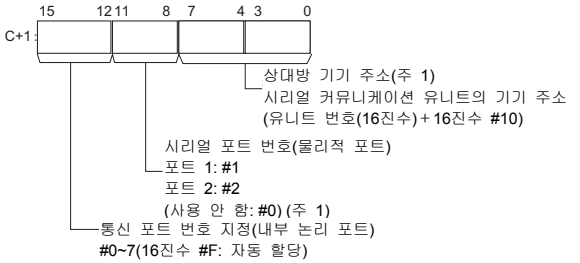
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
D	수신 데이터 저장 시작 CH 번호	UINT	가변
C	제어 데이터 시작 CH 번호	UDINT	2
N	저장 바이트 수 10진수 &0~255(16진수 #0000~0100)	UINT	1

C: 제어 데이터 시작 CH 번호





주 1: 시리얼 포트 번호(물리적 포트)에 #00을 지정하고 상대방 기기 주소에 시리얼 포트의 기기 주소(시리얼 포트 1: 16진수 #80+16진수 #04×유니트 번호(16진수), 시리얼 포트 2: 16진수 #81+16진수 #04×유니트 번호(16진수))를 직접 지정할 수도 있습니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
D	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
C											○							
N												○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 명령 실행 시, 지정된 통신 포트 번호(내부 논리 포트)의 네트워크 통신 명령 실행 가능 플래그 A202.00~07이 OFF로 되었을 때 또는 통신 포트를 모두 사용 중일 때 ON · C의 데이터가 범위 밖일 때 ON · N의 데이터가 10진수 &0~256 또는 16진수 #0000~0100의 범위에 없을 때 ON · 인터럽트 태스크에서 실행했을 때 ON(CJ2 CPU 유니트에서 인터럽트 고속 기동 기능 사용 시) · 그 이외일 때 OFF

할당 DM 설정 영역

■ m = D30000 + 100 × 유니트 번호 CH

이름		비트	내용	설정
포트 1	포트 2			
m+4	m+14	15~08	무수순 모드 시 시작 코드	16진수 #00~FF
		07~00	무수순 모드 시 종료 코드	16진수 #00~FF
m+5	m+15	15~12	무수순 모드 시 시작 코드 유무 설정	16진수 #0: 없음 16진수 #1: 있음
		11~08	무수순 모드 시 종료 코드 유무 설정	16진수 #0: 없음 16진수 #1: 있음 16진수 #2: CR+LF 지정
		07~00	수신 데이터 수	16진수 #00(초기값): 256바이트 16진수 #01~FF: 1~256바이트

관련 특수 보조 릴레이

이름	주소	내용
네트워크 통신 명령 실행 가능 플래그	A202.00~ A202.07	네트워크 통신(RXDU 명령 포함)이 실행 가능할 때 1(ON)이 됩니다. 각 비트가 통신 포트(내부 논리 포트) 번호에 대응합니다. 비트 00~07: 통신 포트 번호 00~07 네트워크 중에 0(OFF), 실행 종료(정상이든 이상이든) 후에는 1(ON)이 됩니다.
네트워크 통신 응답 코드	A203CH~ A210CH	네트워크 통신이 실행된 경우 응답 코드(종료 코드)가 저장됩니다. 각 CH는 통신 포트(내부 논리 포트) 번호에 대응합니다. A203~210CH: 통신 포트 번호 00~07 통신 명령 실행 중에는 #0000이 되고 통신 명령 실행이 종료되면 반영됩니다. 이 값은 운전을 시작할 때 클리어됩니다.
네트워크 통신 실행 에러 플래그	A219.00~ A219.07	네트워크 통신 실행 시 에러(이상)가 발생한 경우 1(ON)이 됩니다. 각 비트가 통신 포트(내부 논리 포트) 번호에 대응합니다. 비트 00~07: 통신 포트 번호 00~07 다음 네트워크 통신 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 통신 명령 실행에서 0(OFF)이 됩니다.

통신 응답 코드 목록

코드(16진수)	내용
#0401	정의되지 않은 명령(시리얼 통신 모드가 프로토콜 매크로, NT 링크, 루프백 테스트 모드, 시리얼 게이트웨이 중 하나인 경우 등)
#0205	응답 시간 초과(시리얼 통신 모드가 상위 링크 모드인 경우 등)
#1001	명령 길이가 최대 명령 길이를 초과
#1002	명령 길이가 최소 명령 길이를 만족하지 못 함
#1004	명령 형식이 다름
#110C	그 밖의 파라미터 에러
#2201	실행 중이므로 동작 불가능 (송신 중이므로 동작 불가능)
#2202	정지 중이므로 실행 불가능 (프로토콜 전환 중이기 때문에 실행 불가능)

관련 CPU 고기능 유니트 릴레이 영역

■ n = 1500 + 25×유니트 번호

이름		비트	내용
포트 1	포트 2		
n+9CH	n+19CH	06	수신 완료 플래그 1: 수신 완료 0: 수신하지 않았거나 수신 중 0→1: 보드/유니트가 지정된 수의 수신 데이터를 수신 완료했을 때 1→0: RXD 또는 RXDU 명령 실행에 의해 CPU 유니트가 데이터를 수신하여 I/O 메모리에 쓰기 종료 직후
n+9CH	n+19CH	07	수신 초과 플래그 1: 보드/유니트가 지정된 수신 데이터 수 이상을 수신 0: 보드/유니트가 지정된 수신 데이터 수 이상을 수신하지 않음 0→1: 보드/유니트가 수신 완료 후 데이터를 다시 수신했을 때 1→0: RXD 또는 RXDU 명령 실행에 의해 CPU 유니트가 데이터를 수신하여 I/O 메모리에 쓰기 종료 직후
n+10CH	n+20CH	00~15	수신 카운터 수신한 데이터의 바이트 수를 10진수 &0~256 또는 16진수 #0000~0100으로 표시합니다.
n+8CH	n+18CH	04	오버런 에러 RXDU 명령 실행 전에 수신 버퍼에 260바이트 이상의 데이터를 수신했을 때 ON 주: 오버런 에러가 발생하여 ON이 된 경우는 전원 OFF→ON 또는 포트 재시작을 통해서만 이 플래그를 OFF로 할 수 있음

기능

- C+1의 비트 0~7로 지정한 기기 주소에 해당하는 시리얼 커뮤니케이션 유니트의 C+1의 비트 8~11로 지정한 시리얼 포트(무수순 모드)에서 수신 완료 데이터를 D로 지정한 수신 데이터 저장 시작 CH 번호부터 N으로 지정한 저장 바이트 수만큼 저장합니다. 수신 완료 데이터가 N으로 지정한 저장 바이트 수만큼 존재하지 않는 경우는 존재하는 수신 완료 데이터만큼 저장합니다.
- 할당 DM 영역 설정에서 수신 메시지의 프레임 형식을 다음과 같이 설정할 수 있습니다.

  1)시작 코드 설정 유무(있는 경우, 코드 #00~FF)

  2)종료 코드 설정 유무(있는 경우, 코드 16진수 #00~FF 또는 CR+LF, 없는 경우, 수신 데이터 수 16진수 #00~FF: 1~256, #00은 256을 나타냅니다.)
- C의 비트 0~3으로 지정한 저장 순서 지정에 따른 순서로 수신 데이터를 D 이후에 저장합니다.
- 수신 완료 플래그(*1)가 ON이 되는 경우

  1)할당 DM 영역 설정으로 지정한 수신 바이트 수를 수신하면 수신 완료 플래그가 ON이 됩니다. 이때, 수신 카운터(*2)는 할당 DM 영역 설정으로 지정된 무수순 모드 시의 수신 데이터 수와 같아집니다.

  2)할당 DM 영역 설정에서 종료 코드를 지정한 경우, 해당 종료 코드를 수신하면 수신 완료 플래그가 ON이 됩니다.

  3)256바이트의 데이터를 수신하면 수신 완료 플래그가 ON이 됩니다.
- 수신 초과 플래그(*3)가 ON이 되는 경우

  1)수신 완료 플래그(*1)가 ON인 상태에서 무수순 수신 명령(RXDU 명령)을 실행하지 않고 다시 데이터를 수신하면 수신 초과 플래그가 ON이 됩니다.

  2)할당 DM 영역 설정으로 지정한 수신 데이터 수를 초과해서 수신하면 수신 초과 플래그가 ON이 됩니다.
- 데이터를 259바이트 수신하면 수신을 정지합니다. 그 이후의 수신이 있으면 오버런 에러 플래그(*5) 및 전송 에러 발생 플래그(*6)가 1(ON)이 됩니다.
- 지정 바이트 수 이상의 수신 데이터는 다음 RXDU 명령 실행 시 파기됩니다.
- 이 RXDU 명령을 실행하면 수신 버퍼에서 D에 데이터가 저장되고 수신 완료 플래그(*1)가 OFF가 되며 수신 카운터(*2)는 0이 됩니다. 또한 수신 초과 플래그가 ON인 경우는 OFF가 됩니다.
- C의 비트 4~7로 지정한 CS, DR 신호의 모니터 지정은 다음과 같이 반영됩니다.

  1)CS 신호의 모니터를 지정한 경우, CS 신호의 상태를 D로 지정한 채널의 비트 15에 출력합니다.

  2)DR 신호의 모니터를 지정한 경우, DR 신호의 상태를 D로 지정한 채널의 비트 15에 출력합니다.

  3)CS 신호 및 DR 신호의 모니터를 지정한 경우, CS 신호의 상태를 D로 지정한 채널의 비트 15에, DR 신호의 상태를 D로 지정한 채널의 비트 14에 출력합니다.
- CS, DR 신호의 모니터 지정을 #1~3으로 한 경우, 수신 완료 플래그(*1)와 관계 없이 데이터를 읽습니다. CS 신호 또는 DR 신호 모니터를 지정한 경우 수신 데이터의 저장 동작은 수행하지 않습니다.
- 수신 가능한 바이트 수는 최대 259바이트(수신 데이터부: 최대 256바이트, 시작 코드부: 1바이트, 종료 코드부: CR+LF 지정 2바이트)입니다.

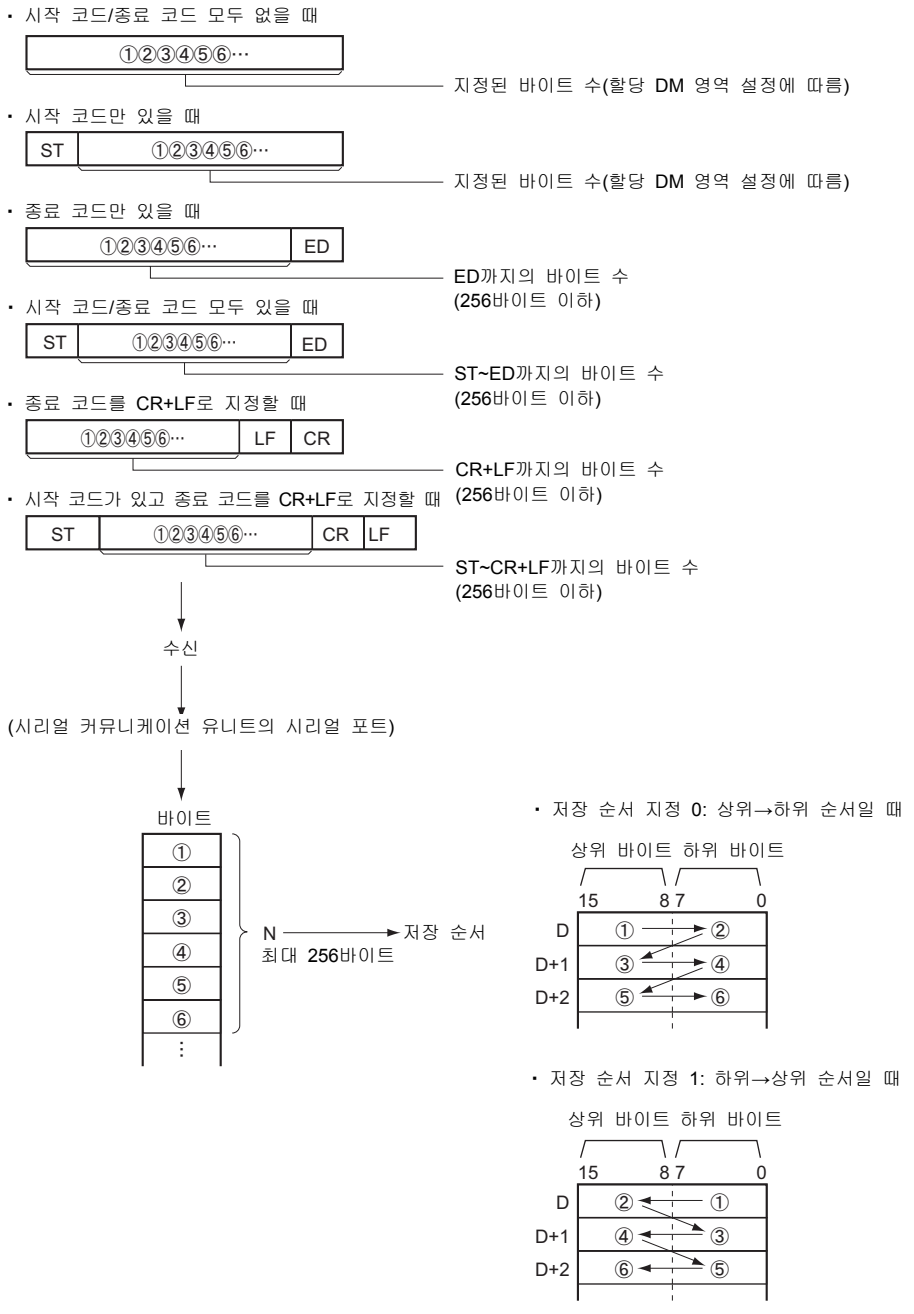
- N에는 시작 코드 및 종료 코드를 포함하지 않는 수신 데이터부의 크기를 지정하십시오.
- C+1의 비트 12~15로 지정한 통신 포트(내부 논리 포트) 0~7 중 하나를 사용합니다.
- 통신 포트(내부 논리 포트) 번호를 자동 할당("F")으로 사용하는 경우에 대해서는 3-736의 '내부 논리 포트 자동 할당 기능'의 설명을 참조하십시오.

관련 특수 보조 릴레이/할당 릴레이의 주소

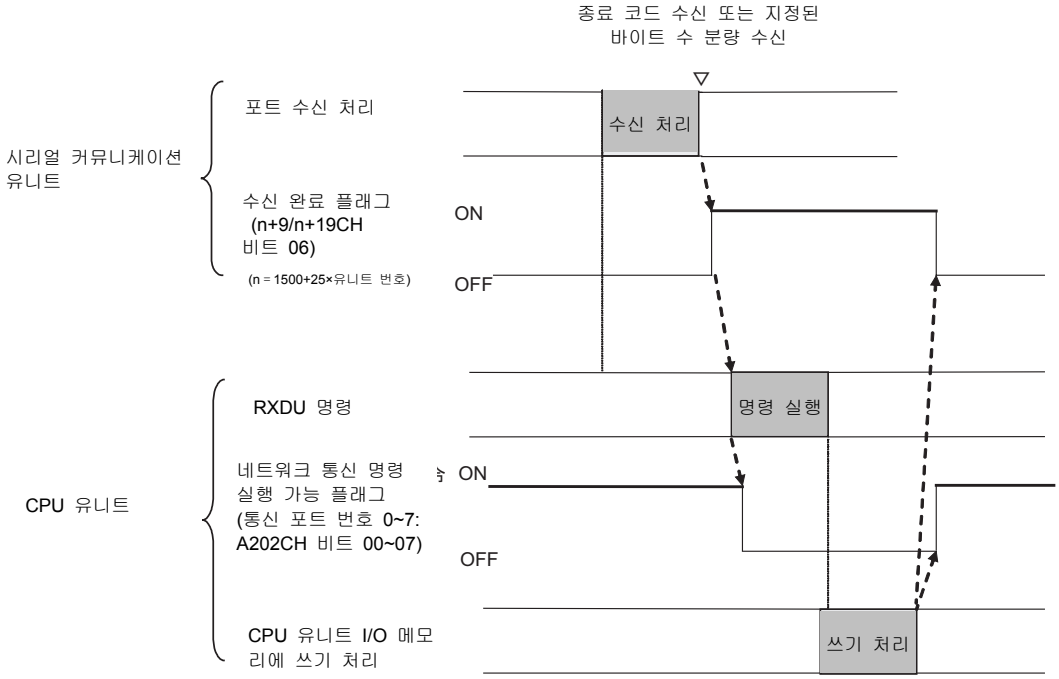
*1: 수신 완료 플래그	포트 1	n+9CH 비트 06
	포트 2	n+19CH 비트 06
*2: 수신 카운터	포트 1	n+10CH
	포트 2	n+20CH
*3: 수신 초과 플래그	포트 1	n+9CH 비트 07
	포트 2	n+19CH 비트 07
*4: 오버런 에러	포트 1	n+8CH 비트 04
	포트 2	n+18CH 비트 04
*5: 전송 에러 발생	포트 1	n+8CH 비트 15
	포트 2	n+18CH 비트 15

(n = 1500+25×유니트 번호)

■ 시작 코드/종료 코드 설정과 송신 데이터



■ 수신 실행 플래그의 동작



힌트

· 수신 완료 플래그(할당 릴레이 영역)가 1(ON)일 때 이 명령을 실행하고 데이터를 수신하도록 하십시오 (수신 버퍼에서).

사용상의 주의 사항

· 데이터를 수신할 때 RXDU 명령으로 수신 데이터를 읽지 않으면 다음 데이터를 수신할 수 없습니다. 수신 완료 플래그가 ON이 되면 다음 수신까지 RXDU 명령으로 수신 데이터를 읽으십시오.

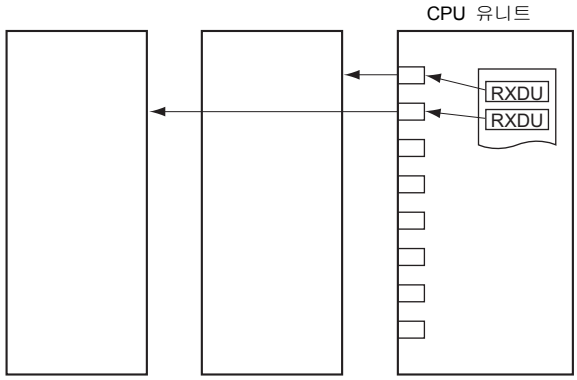
· 시리얼 커뮤니케이션 유니트의 시리얼 포트에 대한 RXDU 명령의 경우, RXDU 명령 실행 후 수신 버퍼가 클리어됩니다. 따라서 복수의 RXDU 명령을 사용한 분할 수신은 불가능하므로 주의하십시오.

· 무수순 모드로 설정한 시리얼 커뮤니케이션 유니트의 시리얼 포트에만 사용할 수 있습니다.

· N에 0을 지정한 경우 D 이후에 데이터를 전송하지 않습니다. 이 경우, 수신 완료 플래그(*1)와 수신 초과 플래그(*3)가 OFF되고 수신 카운터(*2)는 0이 됩니다.

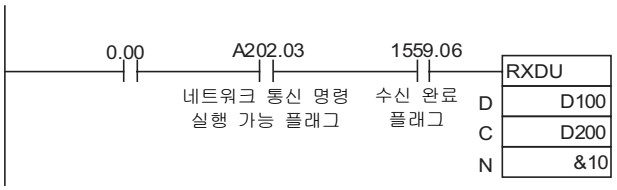
· RXDU 명령(내부적으로는 FINS 명령을 전송했기 때문)은 네트워크 통신 명령의 SEND/RECV/CMND 명령 및 시리얼 통신 명령의 PMCR/TXDU 명령과 동일하게 통신 포트(내부 논리 포트) 0~7을 사용하여 시리얼 커뮤니케이션 보드(CS 시리즈만 해당) 또는 시리얼 커뮤니케이션 유니트에 대해 송수신 시퀀스 실행 지시를 내립니다. 따라서 네트워크 통신 명령/PMCR/TXDU 명령이나 다른 RXDU 명령에 의해 통신 포트(내부 논리 포트)가 사용될 때는 해당 통신 포트를 사용하여 RXDU 명령을 실행할 수 없습니다.

  그러므로 RXDU 명령의 입력 조건에는 C1의 비트 12~15로 지정한 사용 통신 포트 번호에 대응한 특수 보조 릴레이의 '네트워크 통신 실행 가능 플래그'(A202.00~A202.07)를 a 접점으로 삽입하십시오.

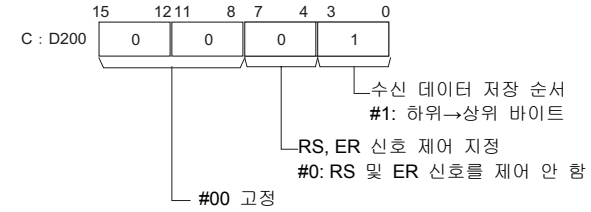


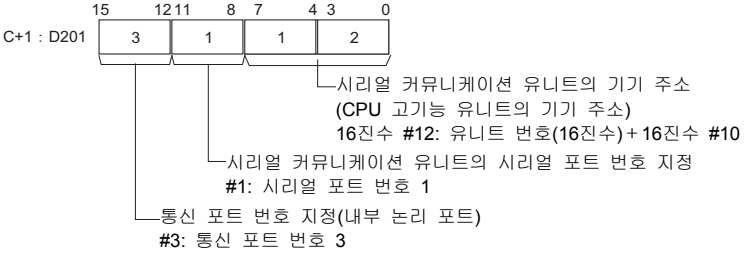
· RXDU 명령은 수신 완료 플래그(n+9/n+19CH 비트 06)가 0(OFF)인 동안은 실행할 수 없습니다.
따라서 RXDU 명령의 입력 조건에는 수신 완료 플래그를 a 접점으로 삽입하십시오.

샘플 프로그램

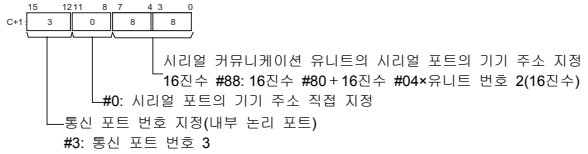


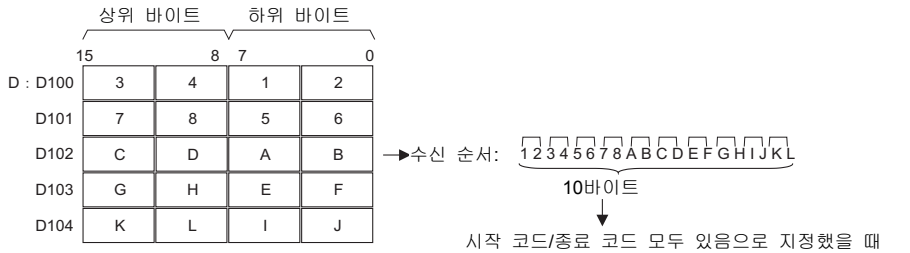
0.00이 ON이고 네트워크 통신 명령 실행 가능 플래그 A202.03이 ON이며 수신 완료 플래그 1559.06이 ON인 경우, 유니트 번호 2의 시리얼 커뮤니케이션 유니트의 시리얼 포트 번호 1에 연결된 범용 외부 기기(예: 바코드 리더 등)에서 수신한 데이터를 변환하지 않고 그대로 통신 포트(내부 논리 포트) 번호 3을 사용하여 D100의 하위 바이트부터 10바이트에 저장합니다.





참고:
C+1로 시리얼 커뮤니케이션 유니트의 시리얼 포트의 기기 주소를 직접 지정할 수도 있습니다.





시작 코드/종료 코드 모두 있음으로 지정했을 때 (PLC 시스템 설정에서)

ST	12	34	56	78	AB	CD	EF	GH	IJ	KL	ED

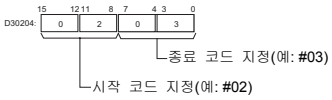
ST: 시작 코드, 예: #02
ED: 종료 코드, 예: #03

↓

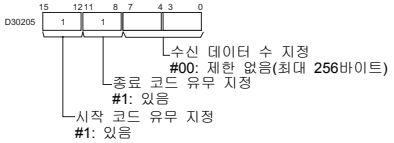
데이터 수신

참고:
할당 DM 영역 설정

- 시작 코드/종료 코드 지정

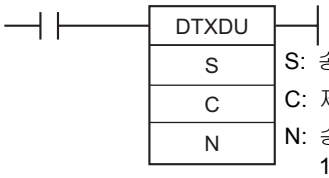


- 시작 코드/종료 코드 유무 지정



DTXDU

명령 이름	니모닉	동작 옵션	펑션 번호	기능
시리얼 커뮤니케이션 유니트 시리얼 포트 직접 송신	DTXDU	@DTXDU	262	시리얼 커뮤니케이션 유니트 CJ1W-SCU22/32/42의 시리얼 통신 포트(무수순 모드)에서 할당 DM 영역 설정의 무수순 모드일 때의 시작 코드/종료 코드 지정에 따라 지정된 시작 CH 번호부터 지정된 바이트 수만큼 송신합니다. 이 명령은 명령 실행 시에 CPU 유니트에서 시리얼 커뮤니케이션 유니트로 송신 데이터를 전송하기 때문에 고속 데이터 송신이 가능합니다.

심볼	DTXDU
	 S: 송신 데이터 시작 CH 번호 C: 제어 데이터 시작 CH 번호 N: 송신 바이트 수 10진수 &0~256(16진수 #0000~0100)

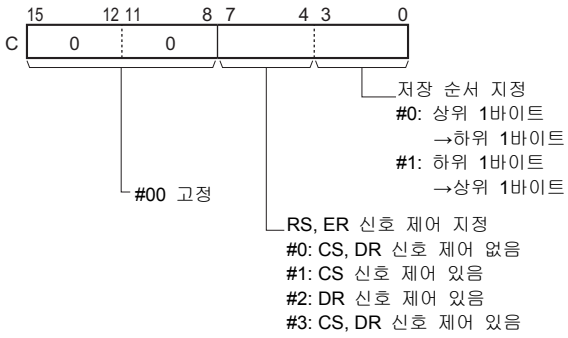
특정 영역 내의 사용 여부

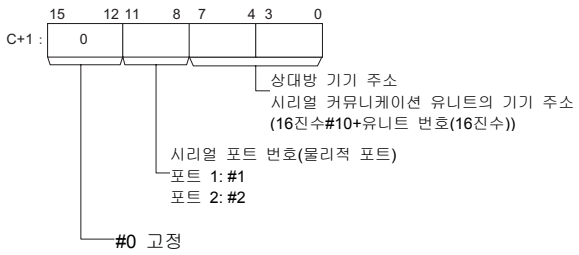
영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	송신 데이터 시작 CH 번호	UINT	가변
C	제어 데이터 시작 CH 번호	UDINT	2
N	송신 바이트 수 10진수 &0~256(16진수 #0000~0100)	UINT	1

C: 제어 데이터 시작 CH 번호





■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
C											○							
N												○						

DTXDU

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	・C의 데이터가 범위 밖일 때 ON ・N의 데이터가 10진수 &0~256 또는 16진수 #0000~0100의 범위에 없을 때 ON ・사이클릭 태스크 내에서 DTXDU/DRXDU 명령을 실행하는 중에 인터럽트 태스크가 기동되어 인터럽트 태스크 내에서도 이 명령을 실행했을 때 ON ・지정 유니트가 CJ1W-SCU22/32/42가 아닐 때 ON(이때, 명령 처리 시간이 최대 1ms가 됩니다.) ・지정된 시리얼 커뮤니케이션 유니트가 초기 처리 중일 때 ON ・시리얼 커뮤니케이션 유니트의 포트 지정이 무수순 모드가 아닐 때 ON ・송신 준비가 아닌 상태에서 이 명령이 실행되었을 때 ON ・그 이외일 때 OFF

할당 DM 설정 영역

■ m=D30000+100×유니트 번호 CH

이름		비트	내용	설정
포트 1	포트 2			
m+2	m+12	15	무수순 모드 시 송신 딜레이 시간	기본값(0ms) 1: 임의 설정
		14~00	무수순 모드 시 송신 딜레이 임의 설정 시간	16진수 #0000~7530(10진수 0~300,000ms) [10ms 단위]
m+4	m+14	15~08	무수순 모드 시 시작 코드	16진수 #00~FF
		07~00	무수순 모드 시 종료 코드	16진수 #00~FF
m+5	m+15	15~12	무수순 모드 시 시작 코드 유무 설정	16진수 #0: 없음 16진수 #1: 있음
		11~08	무수순 모드 시 종료 코드 유무 설정	16진수 #0: 없음 16진수 #1: 있음 16진수 #2: CR+LF 지정
		07~00	데이터 수	16진수 #00(초기값): 256바이트 16진수 #01~FF: 1~256바이트

관련 CPU 고기능 유니트 릴레이 영역

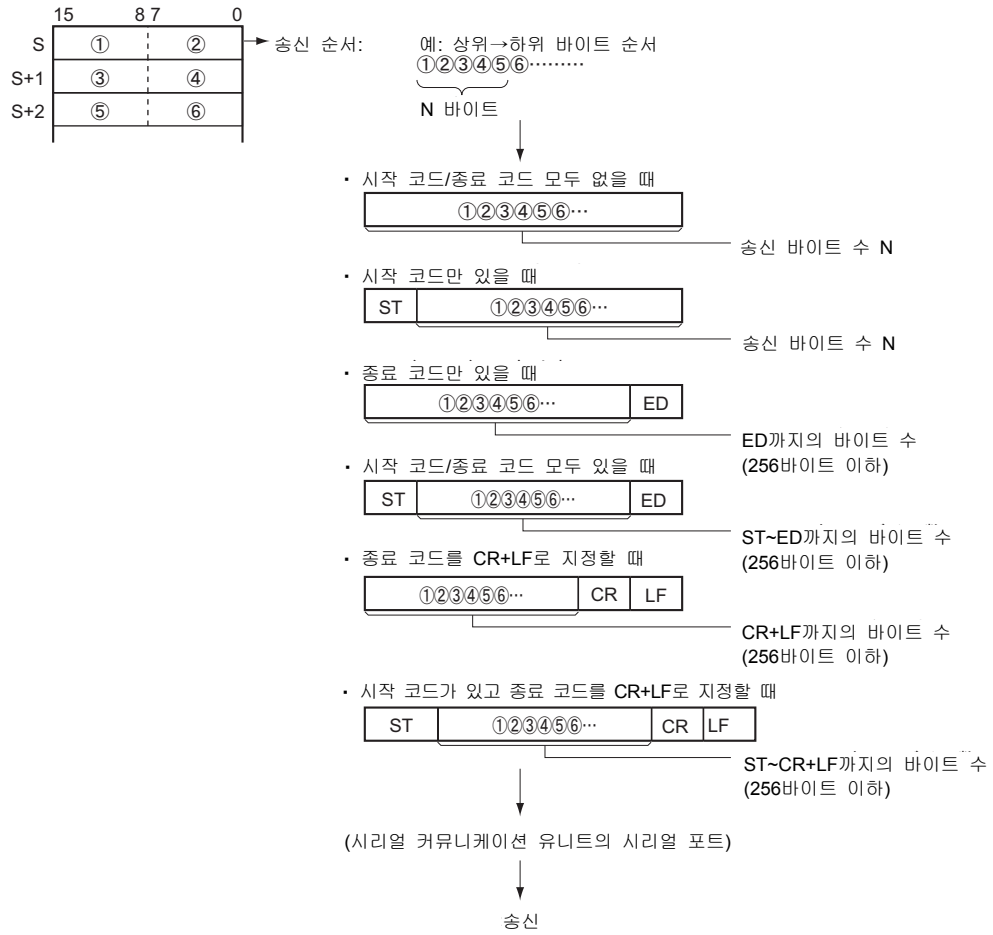
■ n = 1500 + 25×유니트 번호

이름		비트	내용
포트 1	포트 2		
n+9	n+19	04	DTXDU 명령용 송신 준비 플래그 1: 송신 가능, 0: 송신 불가능

기능

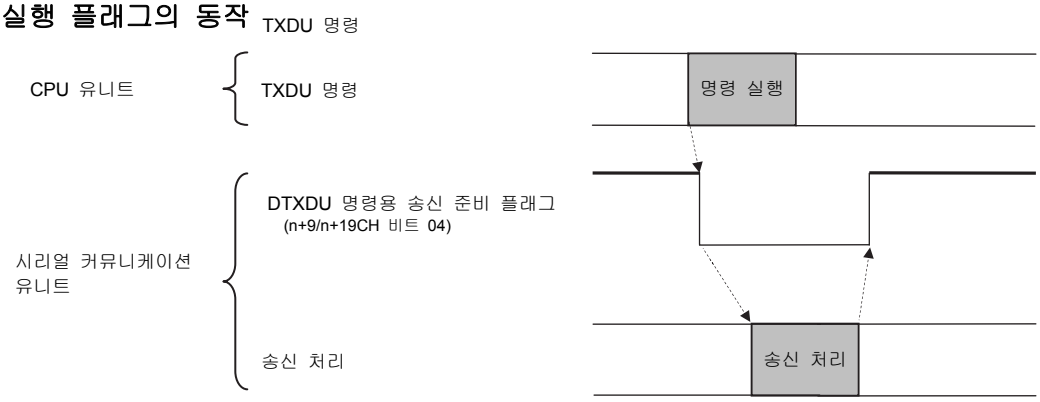
- S로 지정한 송신 데이터 시작 CH 번호부터 N으로 지정한 송신 바이트 수 분량의 데이터를 변환하지 않고 그대로 C+1의 비트 0~7로 지정한 기기 주소에 해당하는 시리얼 커뮤니케이션 유니트(CJ1W-SCU22/32/42)의 C+1의 비트 8~11로 지정한 시리얼 포트(무수순 모드)에 출력합니다.
- 이 명령은 명령 실행 시에 CPU 유니트에서 시리얼 커뮤니케이션 유니트로 송신 데이터를 전송하기 때문에 고속 데이터 송신이 가능합니다.
- 할당 DM 영역 설정에서 송신 메시지의 프레임 형식을 다음과 같이 설정할 수 있습니다. 이 설정에서 시작 코드 및 종료 코드를 지정한 경우, 지정된 시작 및 종료 코드를 부가하여 송신합니다.
  1) 시작 코드 설정 유무(있는 경우, 코드 16진수 #00~FF)
  2) 종료 코드 설정 유무(있는 경우, 코드 16진수 #00~FF 또는 CR+LF)
- C의 비트 0~3으로 지정한 저장 순서 지정에 따른 순서로 송신합니다.
- C의 비트 4~7로 지정한 RS, ER 신호의 제어 지정은 다음과 같이 반영됩니다.
  1) RS 신호의 제어를 지정한 경우, S의 비트 15의 내용을 RS 신호에 반영합니다.
  2) ER 신호의 제어를 지정한 경우, S의 비트 15의 내용을 ER 신호에 반영합니다.
  3) RS 신호 및 ER 신호의 제어를 지정한 경우, S의 비트 15의 내용을 RS 신호에, S의 비트 14의 내용을 ER 신호에 반영합니다.
- 송신 가능한 바이트 수는 최대 259바이트(송신 데이터부: 최대 256바이트, 시작 코드부: 1바이트, 종료 코드부: CR+LF 지정 2바이트)입니다.
- N에는 시작 코드 및 종료 코드를 포함하지 않는 송신 데이터부의 크기를 지정하십시오.

■ 시작 코드/종료 코드 설정과 송신 데이터



DTXDU

■ 송신 실행 플래그의 동작



주: DTXDU 명령은 매회 실행형 명령이므로 명령을 실행한 후 즉시 송신 처리가 실행됩니다.

힌트

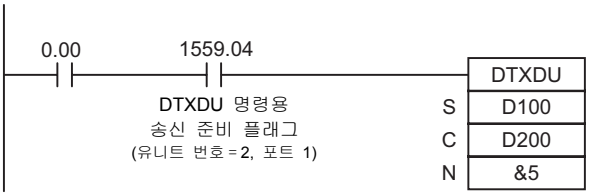
- DTXDU 명령으로 데이터를 송신하는 경우 범용 외부 기기에 따라 송신 간격을 설정해야 합니다. 이 경우, 할당 DM 영역 설정을 통해 송신 딜레이 시간을 설정하여 송신 간격을 조정하십시오.

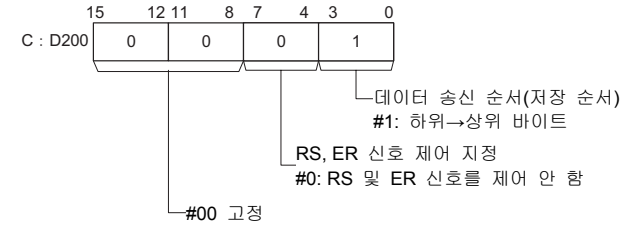
사용상의 주의 사항

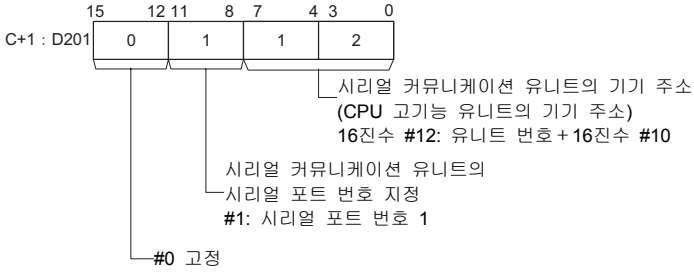
- 이 명령은 반드시 고속 시리얼 커뮤니케이션 유니트(CJ1W-SCU22/32/42)에 대해 사용하십시오. 다른 고기능 유니트에 대해 잘못 사용한 경우 이 명령의 실행 시간이 약 1ms가 되어 사이클 시간이 연장됩니다.
- 이 명령은 CJ2H CPU 유니트의 유니트 Ver.1.1 이상에 장착되어 있으며, 무수순 모드로 설정한 시리얼 커뮤니케이션 유니트 CJ1W-SCU22/32/42의 시리얼 포트에만 사용 가능합니다.
- N에 0을 지정한 경우 송신하지 않습니다.
- DTXDU 명령은 DTXDU 명령용 송신 준비 플래그(n+9/n+19CH 비트 04)가 0(OFF)인 동안은 실행할 수 없습니다(n = 1500+25×유니트 번호).
  따라서 DTXDU 명령의 입력 조건에는 DTXDU 명령용 송신 준비 플래그를 a 접점으로 삽입하십시오.

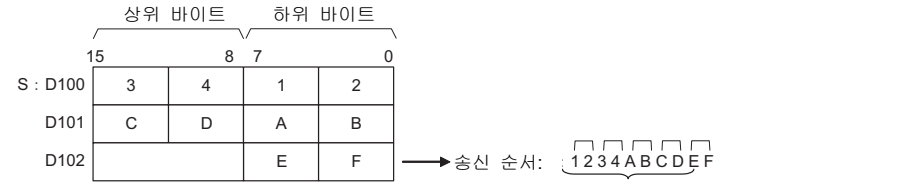
샘플 프로그램

0.00이 ON이고 DTXDU 명령용 송신 준비 플래그 1559.04가 OFF인 경우, D100의 하위 바이트부터 5바이트를 변환하지 않고 그대로 유니트 번호 2의 시리얼 커뮤니케이션 유니트의 시리얼 포트 번호 1에 연결된 범용 외부 기기에 데이터를 출력합니다.



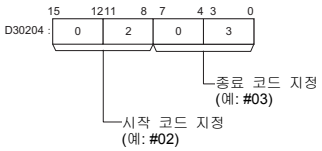




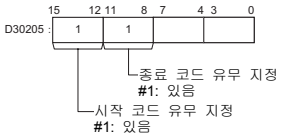


참고:
할당 DM 영역 설정

• 시작 코드/종료 코드 지정



• 시작 코드/종료 코드 유무 지정



시작 코드/종료 코드 모두 있음으로 지정했을 때
(할당 DM 영역)

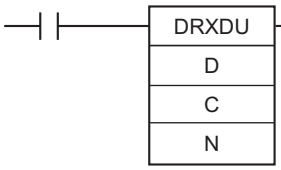
ST	12	34	AB	CD	EF	ED

ST: 시작 코드, 예: #02
ED: 종료 코드, 예: #03

데이터 송신

DRXDU

명령 이름	니모닉	동작 옵션	평션 번호	기능
시리얼 커뮤니케이션 유니트 시리얼 포트 직접 수신	DRXDU	@DRXDU	261	시리얼 커뮤니케이션 유니트 CJ1W-SCU22/32/42의 시리얼 통신 포트(무수순 모드)에서 할당 DM 영역 설정의 무수순 모드일 때의 시작 코드/종료 코드 지정에 따라 수신한 데이터를 지정된 시작 CH 번호부터 지정된 바이트 수만큼 출력합니다. 이 명령은 명령 실행 시에 시리얼 커뮤니케이션 유니트에서 CPU 유니트로 수신 데이터를 전송하기 때문에 고속 데이터 전송이 가능합니다.

심볼	DRXDU
	 D: 수신 데이터 시작 CH 번호 C: 제어 데이터 시작 CH 번호 N: 저장 바이트 수 10진수 &0~256(16진수 #0000~0100)

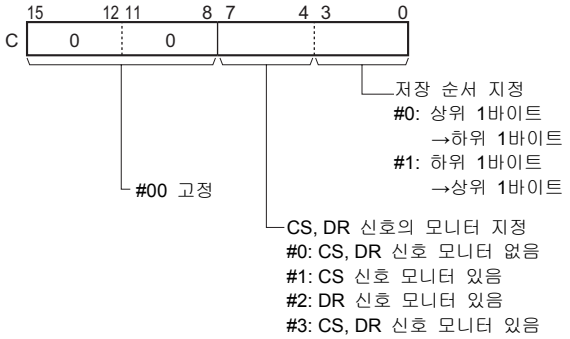
특정 영역 내의 사용 여부

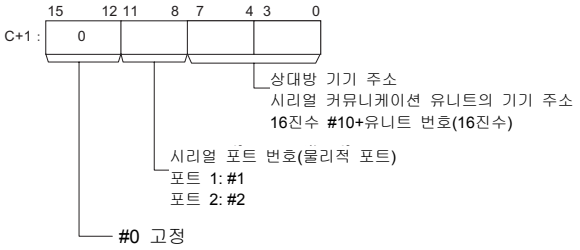
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
D	수신 데이터 저장 시작 CH 번호	UINT	가변
C	제어 데이터 시작 CH 번호	UDINT	2
N	저장 바이트 수 10진수 &0~255(16진수 #0000~0100)	UINT	1

C: 제어 데이터 시작 CH 번호





■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
D	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
C											○							
N												○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	・C의 데이터가 범위 밖일 때 ON ・N의 데이터가 10진수 &0~256 또는 16진수 #0000~0100의 범위에 없을 때 ON ・사이클릭 태스크 내에서 DTXDU/DRXDU 명령을 실행하는 중에 인터럽트 태스크가 기동되어 인터럽트 태스크 내에서도 이 명령을 실행했을 때 ON ・지정 유니트가 CJ1W-SCU22/32/42가 아닐 때 ON(이때, 명령 처리 시간이 최대 1ms가 됩니다.) ・지정된 시리얼 커뮤니케이션 유니트가 초기 처리 중일 때 ON ・시리얼 커뮤니케이션 유니트의 포트 지정이 무수순 모드가 아닐 때 ON ・그 이외일 때 OFF

할당 DM 설정 영역

■ m = D30000 + 100 × 유니트 번호 CH

이름		비트	내용	설정
포트 1	포트 2			
m+4	m+14	15~08	무수순 모드 시 시작 코드	16진수 #00~FF
		07~00	무수순 모드 시 종료 코드	16진수 #00~FF
m+5	m+15	15~12	무수순 모드 시 시작 코드 유무 설정	16진수 #0: 없음 16진수 #1: 있음
		11~08	무수순 모드 시 종료 코드 유무 설정	16진수 #0: 없음 16진수 #1: 있음 16진수 #2: CR+LF 지정
		07~00	수신 데이터 수	16진수 #00(초기값): 256바이트 16진수 #01~FF: 1~256바이트
m+25	m+35	15~08	인터럽트 통지가 있는 경우에 기동하는 외부 인터럽트 태스크 번호	16진수 #00~FF: 0~255바이트
		07~05	예약	
		04	데이터 수신 시 CPU 유니트에 대한 인터럽트 통지 유무	16진수 #0: 없음 16진수 #1: 있음
		03~01	예약	
		00	DRXDU 명령 실행 직후의 수신 버퍼 클리어 유무	16진수 #0: 없음 16진수 #1: 있음

관련 CPU 고기능 유니트 릴레이 영역

■ n = 1500 + 25×유니트 번호

이름		비트	내용
포트 1	포트 2		
n+9CH	n+19CH	06	수신 완료 플래그 1: 수신 완료 0: 수신하지 않았거나 수신 중 0→1: 보드/유니트가 지정된 수의 수신 데이터를 수신 완료했을 때 1→0: RXDU 또는 DRXDU 명령 실행에 의해 CPU 유니트가 데이터를 수신하여 I/O 메모리에 쓰기 종료 직후
n+9CH	n+19CH	07	수신 초과 플래그 1: 보드/유니트가 지정된 수신 데이터 수 이상을 수신 0: 보드/유니트가 지정된 수신 데이터 수 이상을 수신하지 않음 0→1: 보드/유니트가 수신 완료 후 데이터를 다시 수신했을 때 1→0: RXDU 또는 DRXDU 명령 실행에 의해 CPU 유니트가 데이터를 수신하여 I/O 메모리에 쓰기 종료 직후
n+10CH	n+20CH	00~15	수신 카운터 수신한 데이터의 바이트 수를 10진수 &0~256 또는 16진수 #0000~0100으로 표시합니다.
n+8CH	n+18CH	04	오버런 에러 RXD, RXDU 또는 DRXDU 명령 실행 전에 수신 버퍼에 260바이트 이상의 데이터를 수신했을 때 ON 주: 오버런 에러가 발생하여 ON이 된 경우는 전원 OFF→ON 또는 포트 재시작을 통해서만 이 플래그를 OFF로 할 수 있음

기능

- C+1의 비트 0~7로 지정한 기기 주소의 시리얼 커뮤니케이션 유니트(CJ1W-SCU22/32/42)의 C+1의 비트 8~11로 지정한 시리얼 포트(무수순 모드)에서 수신 완료 데이터를 D로 지정한 수신 데이터 저장 시작 CH 번호부터 N으로 지정한 저장 바이트 수만큼 저장합니다. 수신 완료 데이터가 N으로 지정한 저장 바이트 수만큼 존재하지 않는 경우는 존재하는 수신 완료 데이터만큼 저장합니다.
- 이 명령은 명령 실행 시에 시리얼 커뮤니케이션 유니트에서 CPU 유니트로 수신 데이터를 전송하기 때문에 고속 데이터 전송이 가능합니다.
- 할당 DM 영역 설정에서 수신 메시지의 프레임 형식을 다음과 같이 설정할 수 있습니다.
  1) 시작 코드 설정 유무(있는 경우, 코드 #00~FF)
  2) 종료 코드 설정 유무(있는 경우, 코드 #00~FF 또는 CR+LF, 없는 경우, 수신 데이터 수 16진수 #00~FF: 1~256, #00은 256을 나타냅니다.)
- C의 비트 0~3으로 지정한 저장 순서 지정에 따른 순서로 수신 데이터를 D 이후에 저장합니다.
- 수신 완료 플래그(*1)가 ON이 되는 경우
  1) 할당 DM 영역 설정으로 지정한 수신 바이트 수를 수신하면 수신 완료 플래그가 ON이 됩니다. 이때, 수신 카운터(*2)는 할당 DM 영역 설정으로 지정된 무수순 모드 시의 수신 데이터 수와 같아집니다.
  2) 할당 DM 영역 설정에서 종료 코드를 지정한 경우, 해당 종료 코드를 수신하면 수신 완료 플래그가 ON이 됩니다.
  3) 256바이트의 데이터를 수신하면 수신 완료 플래그가 ON이 됩니다.
- 수신 초과 플래그(*3)가 ON이 되는 경우
  1) 수신 완료 플래그(*1)가 ON인 상태에서 무수순 수신 명령(RXDU 또는 DRXDU 명령)을 실행하지 않고 다시 데이터를 수신하면 수신 초과 플래그가 ON이 됩니다.
  2) 할당 DM 영역 설정으로 지정한 수신 데이터 수를 초과해서 수신하면 수신 초과 플래그가 ON이 됩니다.
- 데이터를 259바이트 수신하면 수신을 정지합니다. 그 이후의 수신이 있으면 오버런 에러 플래그(*5) 및 전송 에러 발생 플래그(*6)가 1(ON)이 됩니다.
- 지정 바이트 수 이상의 수신 데이터는 수신 버퍼를 클리어 실행/비실행 설정에 따라 다음 DRXDU 명령 실행 시 파기 실행/비실행을 선택할 수 있습니다.
- 이 DRXDU 명령을 실행하면 수신 버퍼에서 D에 데이터가 저장되고 수신 완료 플래그(*1)가 OFF가 되며 수신 카운터(*2)는 0이 됩니다. 또한 수신 초과 플래그가 ON인 경우는 OFF가 됩니다.
- C의 비트 4~7로 지정한 CS, DR 신호의 모니터 지정은 다음과 같이 반영됩니다.
  1) CS 신호의 모니터를 지정한 경우, CS 신호의 상태를 D로 지정한 채널의 비트 15에 출력합니다.
  2) DR 신호의 모니터를 지정한 경우, DR 신호의 상태를 D로 지정한 채널의 비트 15에 출력합니다.
  3) CS 신호 및 DR 신호의 모니터를 지정한 경우, CS 신호의 상태를 D로 지정한 채널의 비트 15에, DR 신호의 상태를 D로 지정한 채널의 비트 14에 출력합니다.
- CS, DR 신호의 모니터 지정을 #1~3으로 한 경우, 수신 완료 플래그(*1)와 관계 없이 데이터를 읽습니다. CS 신호 또는 DR 신호 모니터를 지정한 경우 수신 데이터의 저장 동작은 수행하지 않습니다.

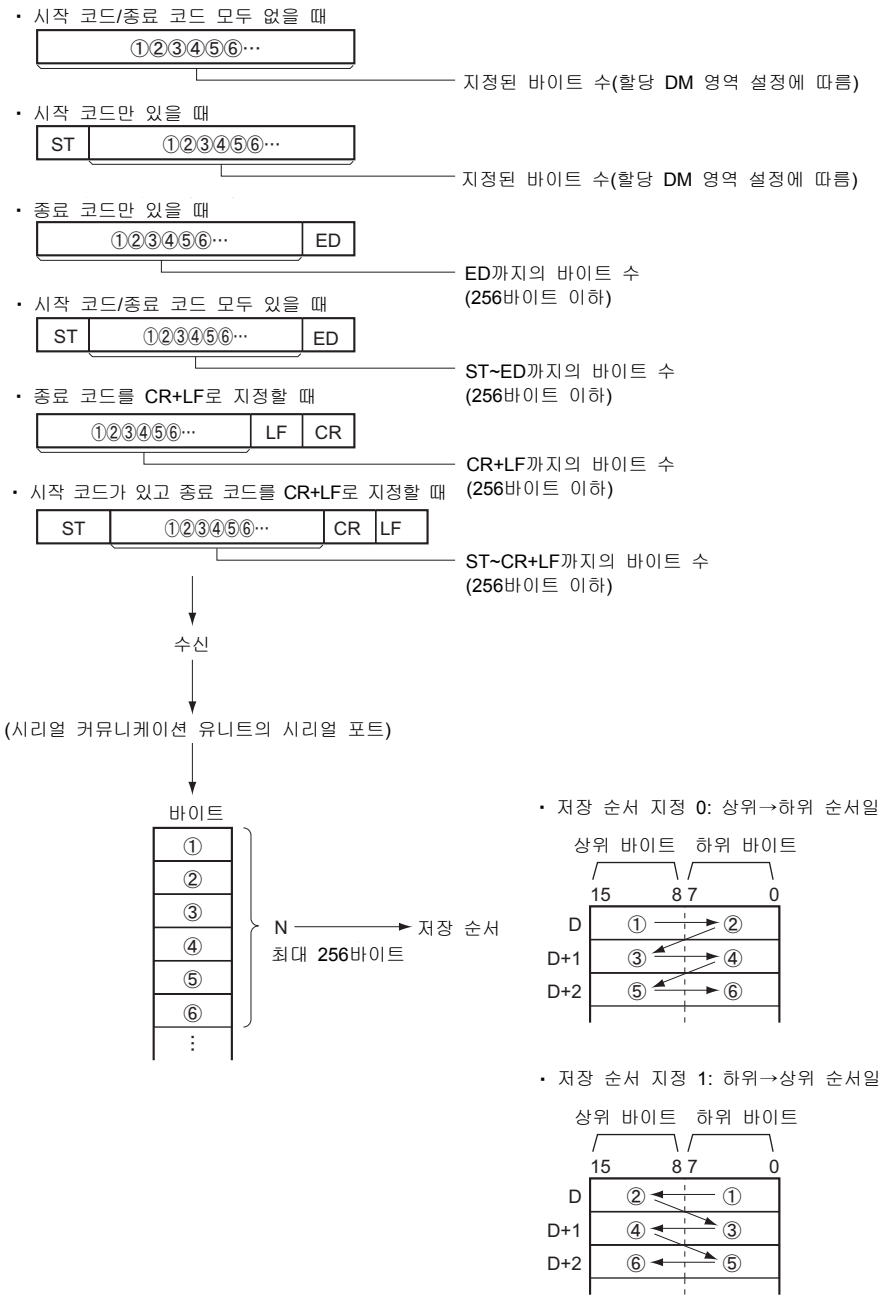
- 수신 가능한 바이트 수는 최대 259바이트(수신 데이터부: 최대 256바이트, 시작 코드부: 1바이트, 종료 코드부: CR+LF 지정 2바이트)입니다.
- N에는 시작 코드 및 종료 코드를 포함하지 않는 수신 데이터부의 크기를 지정하십시오.
- 수신 버퍼 클리어 타이밍
  DRXDU 명령에서는 할당 DM(m+25CH/m+35CH 비트 00)의 설정을 통해 DRXDU 명령 실행 후 수신 버퍼를 클리어할지 여부를 선택할 수 있습니다.

관련 특수 보조 릴레이/할당 릴레이의 주소

*1: 수신 완료 플래그	포트 1	n+9CH 비트 06
	포트 2	n+19CH 비트 06
*2: 수신 카운터	포트 1	n+10CH
	포트 2	n+20CH
*3: 수신 초과 플래그	포트 1	n+9CH 비트 07
	포트 2	n+19CH 비트 07
*4: 오버런 에러	포트 1	n+8CH 비트 04
	포트 2	n+18CH 비트 04
*5: 전송 에러 발생	포트 1	n+8CH 비트 15
	포트 2	n+18CH 비트 15

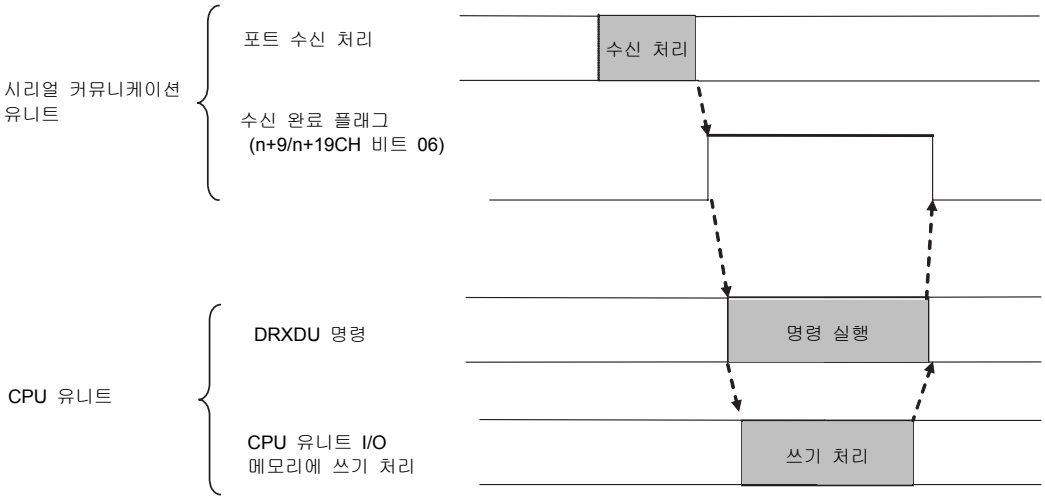
(n = 1500+25×유니트 번호)

■ 시작 코드/종료 코드 설정과 송신 데이터



DRXDU

■ 수신 실행 플래그의 동작



힌트

· 수신 완료 플래그(할당 릴레이 영역)가 1(ON)일 때 이 명령을 실행하고 데이터를 수신하도록 하십시오 (수신 버퍼에서).

사용상의 주의 사항

· 이 명령은 반드시 고속 시리얼 커뮤니케이션 유니트(CJ1W-SCU22/32/42)에 대해 사용하십시오. 다른 고기능 유니트에 대해 잘못 사용한 경우 이 명령의 실행 시간이 약 1ms가 되어 사이클 시간이 연장됩니다.

· 데이터를 수신할 때 RXDU 또는 DRXDU 명령으로 수신 데이터를 읽지 않으면 다음 데이터를 수신할 수 없습니다. 수신 완료 플래그가 ON이 되면 다음 수신까지 RXDU 또는 DRXDU 명령으로 수신 데이터를 읽으십시오.

· 이 명령은 CJ2H CPU 유니트의 유니트 Ver.1.1 이상에 장착되어 있으며, 무수순 모드로 설정한 시리얼 커뮤니케이션 유니트 CJ1W-SCU22/32/42의 시리얼 포트에만 사용 가능합니다.

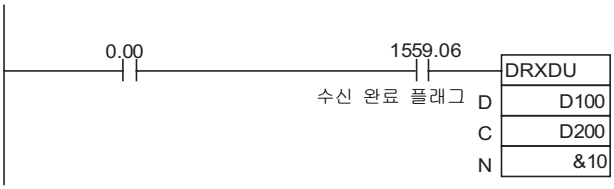
· N에 0을 지정한 경우 D 이후에 데이터를 전송하지 않습니다. 이 경우, 수신 완료 플래그(*1)와 수신 초과 플래그(*3)가 OFF되고 수신 카운터(*2)는 0이 됩니다.

· 사이클릭 태스크의 DRXDU 명령 입력 조건에는 수신 완료 플래그를 a 접점으로 삽입하십시오.
외부 인터럽트 태스크에서 DRXDU 명령을 사용하는 경우, 수신 완료 플래그를 사용할 수 없습니다. (외부 인터럽트가 발생한 경우는 데이터 수신이 완료되었기 때문에 수신 완료 플래그가 필요 없습니다.)

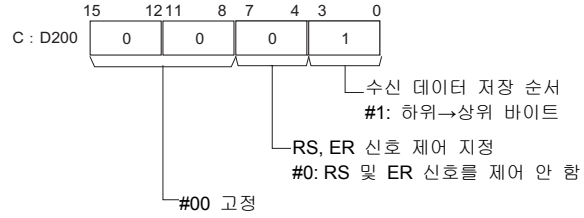
· 시리얼 커뮤니케이션 유니트에서 CPU 유니트에 대해 인터럽트 통지 중에 시리얼 커뮤니케이션 유니트를 재시작하지 마십시오. 시스템의 동작이 불안정해질 수 있습니다.

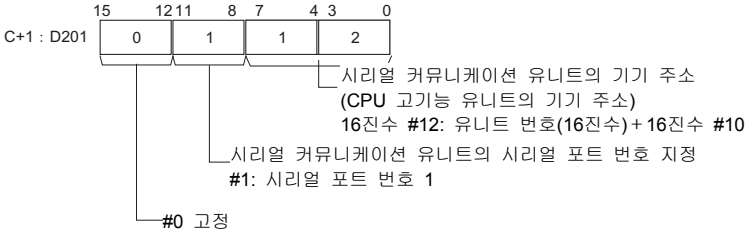
샘플 프로그램

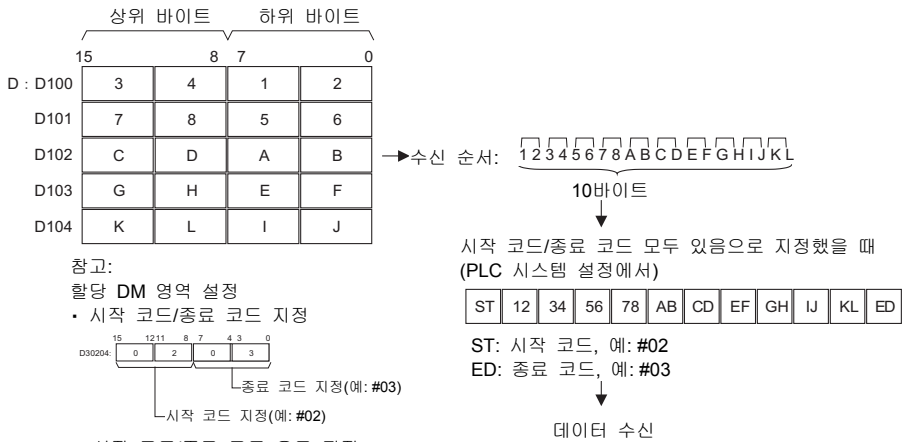
【사이클릭 태스크에서의 사용 방법】



0.00이 ON이고 수신 완료 플래그 1559.06이 ON인 경우, 유니트 번호 2의 시리얼 커뮤니케이션 유니트의 시리얼 포트 번호 1에 연결된 범용 외부 기기에서 수신한 데이터를 변환하지 않고 그대로 D100의 하위 바이트부터 10바이트에 저장합니다.



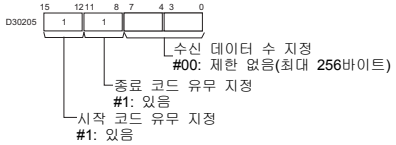




참고:
할당 DM 영역 설정

・ 시작 코드/종료 코드 지정

・ 시작 코드/종료 코드 유무 지정

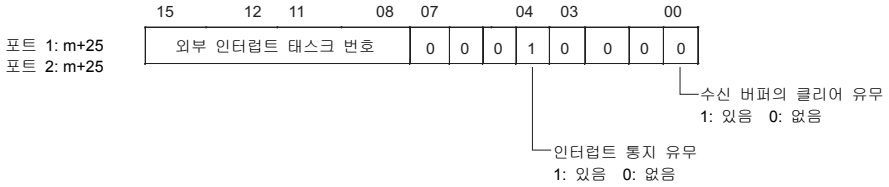


DRXDU

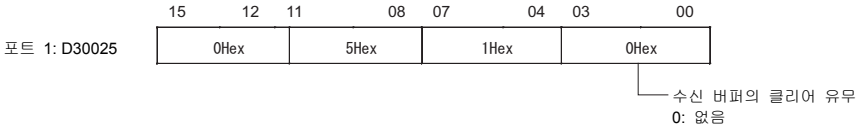
【인터럽트 태스크에서의 사용 방법】
유니트 번호 2의 시리얼 커뮤니케이션 유니트가 범용 외부 기기(예: 바코드 리더 등)에서 수신한 경우, CPU 유니트 내 인터럽트 태스크 5를 기동하고 D00100의 하위 바이트부터 10바이트에 저장합니다.

1. 시스템 설정 영역(할당 DM 영역)의 설정

   시스템 설정 영역(할당 DM 영역)의 m＋25CH, m＋35CH에 다음을 설정합니다.
   m＝D30000＋100×유니트 번호



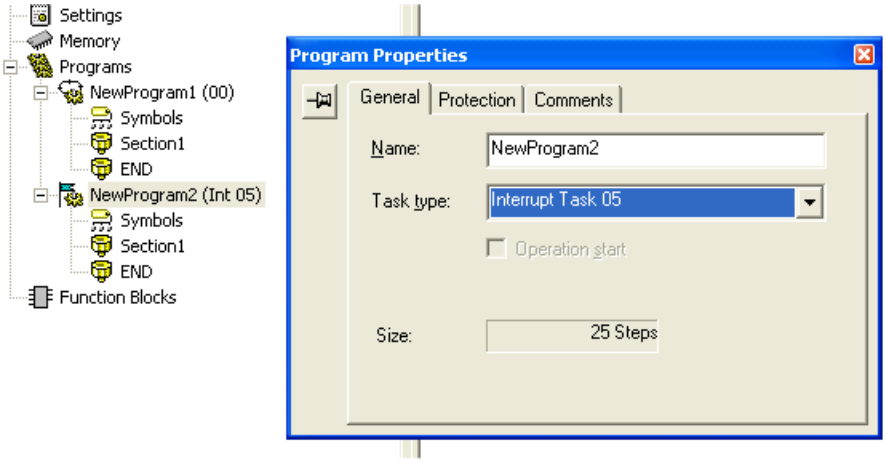
   예) 외부 인터럽트 태스크 번호 5일 때 시리얼 커뮤니케이션 유니트와의 유니트 번호＝0일 때



   또한 위의 할당 DM 영역은 I/O 테이블 창의 시리얼 커뮤니케이션 유니트의 [고기능 유니트 설정 편집]에서도 설정할 수 있습니다.

2. CX-Programmer의 프로젝트 트리에서 외부 인터럽트 태스크에 할당하려는 프로그램을 선택하고 마우스 오른쪽 버튼으로 클릭하여 [속성]을 선택합니다.

   이 예에서는 [태스크 종류]로 '인터럽트 태스크 05'를 선택합니다.



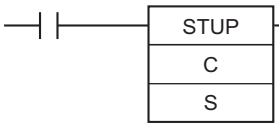
3. 외부 인터럽트 태스크에 할당된 프로그램에 다음 내용을 기술합니다.



* 수신 완료 플래그는 사용하지 않습니다.

STUP

명령 이름	니모닉	동작 옵션	평션 번호	기능
시리얼 포트 통신 설정 변경	STUP	@STUP	237	CPU 유니트, 시리얼 커뮤니케이션 유니트 또는 시리얼 커뮤니케이션 보드의 시리얼 포트(페리페럴 포트 포함) 통신 설정을 변경합니다.

심볼	STUP
	 C: 포트 지정 S: 설정 데이터 시작 CH 번호

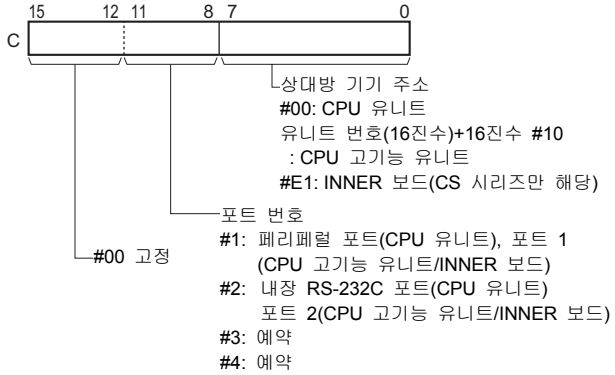
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C	포트 지정	UINT	1
S	설정 데이터 시작 CH 번호	UINT	10

C: 포트 지정



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
S												–						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C의 데이터가 범위 밖일 때 ON · 명령 실행 시, 이미 포트 설정 변경 중 플래그가 ON되어 있을 때 ON · 인터럽트 태스크에서 실행되었을 때 ON · 그 이외일 때 OFF

STUP

관련 특수 보조 릴레이

이름	주소	내용
페리페럴 포트 설정 변경 중 플래그	A619.01	페리페럴 포트의 통신 조건 설정을 변경하는 중에 1(ON)이 됩니다.
RS-232C 포트 설정 변경 중 플래그	A619.02	RS-232C 포트의 통신 조건 설정을 변경하는 중에 1(ON)이 됩니다.
시리얼 커뮤니케이션 유니트 0호기 포트 1~15호기 포트 4용 시리얼 설정 변경 중 플래그	A620.01~ A620.04 A635.01~ A635.04	시리얼 커뮤니케이션 유니트의 해당 포트의 설정 변경 중에 1(ON)이 됩니다.
시리얼 커뮤니케이션 보드의 포트 1~포트 4용 시리얼 설정 변경 중 플래그(CS 시리즈만 해당)	A636.01~ A636.04	시리얼 커뮤니케이션 보드의 해당 포트의 설정 변경 중에 1(ON)이 됩니다.

주: 포트 3/포트 4에 대응하는 설정 변경 중 플래그는 예비용으로, 현재 사용되지 않습니다.

기능

S로 지정한 설정 데이터 시작 CH 번호부터 10채널 분량(S~S+9CH)의 데이터를 지정된 기기 주소의 유니트(장치)의 아래와 같은 통신 설정 영역에 저장합니다. S에 상수 #0000을 지정하면 해당 포트의 통신 설정을 기본값으로 합니다.

기기 주소	유니트	포트 번호	시리얼 포트	시리얼 포트의 통신 설정 영역
16진수 #00	CPU 유니트	1	페리페럴 포트	PLC 시스템 설정 내의 통신 포트 설정(페리페럴 포트) 영역 CJ2 CPU 유니트는 지정 불가
		2	내장 RS-232C 포트	PLC 시스템 설정 내의 통신 포트 설정(내장 RS-232C 포트) 영역
16진수 #10+유니트 번호(16진수)	시리얼 커뮤니케이션 유니트(CPU 고기능 유니트)	1	포트 1 지정 시	D30000+100×유니트 번호를 선두로 한 10CH
		2	포트 2 지정 시	D30000+100×유니트 번호+10을 선두로 한 10CH
16진수 #E1	시리얼 커뮤니케이션 보드(INNER 보드) (CS 시리즈만 해당)	1	포트 1 지정 시	D32000을 선두로 한 10CH
		2	포트 2 지정 시	D32010을 선두로 한 10CH

S~S+9CH의 10CH에는 다음 표의 데이터를 저장해 둡니다.

CPU 유니트의 페리페럴 포트	PLC 시스템 설정 프로그래밍 콘솔 주소+144~153
CPU 유니트의 내장 RS-232C	PLC 시스템 설정 프로그래밍 콘솔 주소+160~169
시리얼 커뮤니케이션 유니트의 포트 1	m~m+9(m=DM30000×유니트 번호)
시리얼 커뮤니케이션 유니트의 포트 2	m+10~m+9(m=DM30000×유니트 번호)
시리얼 커뮤니케이션 보드의 포트 1	D32000~D32009
시리얼 커뮤니케이션 보드의 포트 2	D32010~D32019

STUP 실행 시, 특수 보조 릴레이의 해당 포트 설정 변경 중 플래그(A619~636CH)가 ON이 되고(설정 변경 중에는 계속 ON 상태임), 설정이 완료되면 OFF가 됩니다.

주: STUP 명령으로 CPU 유니트의 통신 설정을 변경한 후, 전원을 OFF→ON으로 했을 때의 통신 설정은 CPU 유니트의 타입에 따라 다음과 같이 달라집니다.

CPU 유니트 타입	전원 OFF→ON 후의 통신 설정
CS1-H/CJ1-H/CJ1M/ CS1D/CJ2 CPU 유니트	STUP 명령으로 변경하기 전의 설정값으로 돌아갑니다.
CS1/CJ1 CPU 유니트	STUP 명령으로 변경한 후의 설정값이 유지됩니다.

힌트

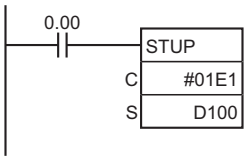
- 이 명령은 운전 중에 특정 조건에서 프로토콜 모드를 변경하는 경우 등에 사용합니다(예를 들어, 프로토콜 매크로 모드에서 모뎀과 회선 연결의 송수신 시퀀스를 실행하고, 특정 조건이 성립할 때 운전 상태에서 프로토콜을 상위 링크로 전환하여 상위 PC에서 모니터/프로그래밍하는 경우 등).
- 통신 시스템 설정 내용은 프로토콜 모드, 통신 속도, 데이터 형식, 프로토콜 매크로 전송 방식, 프로토콜 매크로 송수신 데이터 최대 길이 등으로 구성됩니다.

  자세한 내용은 다음의 설명서를 참조하십시오.
  - CS/CJ 시리즈 사용 설명서 프로그래밍편(SBCA-313) '6-3 시리얼 통신 기능'
  - CS/CJ 시리즈 사용 설명서 설치편(SBCA-312) '7-1 PLC 시스템 설정'
  - CS/CJ 시리즈 시리얼 커뮤니케이션 유니트/보드 사용 설명서(SBCD-300)
  - SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편(SBCA-350)

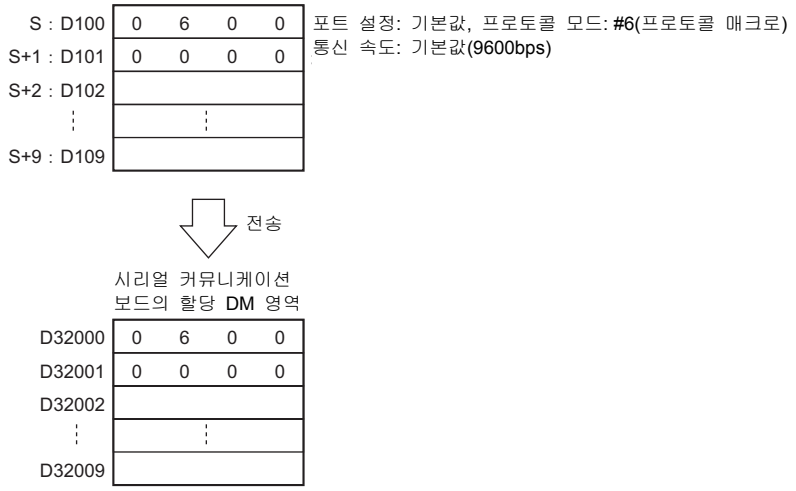
    '4-2 PLC 시스템 설정'

    '11-2 네트워크'

샘플 프로그램

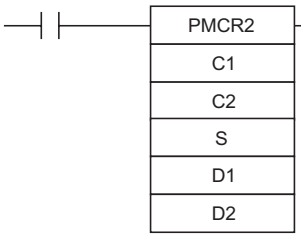


0.00이 ON이면 INNER 보드(시리얼 커뮤니케이션 보드)의 시리얼 포트 1의 통신 설정을 D100~D109의 10CH의 내용으로 변경합니다.
예: 프로토콜 모드를 프로토콜 매크로로 변경합니다.



PMCR2

명령 이름	니모닉	동작 옵션	펑션 번호	기능
프로토콜 매크로 송신 2	PMCR2	@PMCR2	264	시리얼 커뮤니케이션 유니트에 등록된 송수신 시퀀스(프로토콜 데이터)를 호출하여 실행합니다.

	PMCR2
심볼	 C1: 제어 데이터 1 C2: 제어 데이터 2 S: 송신 데이터 시작 채널 D1: 수신 데이터 저장 시작 채널 D2: 통신 정보

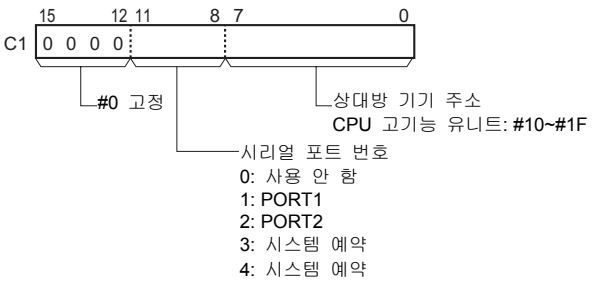
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

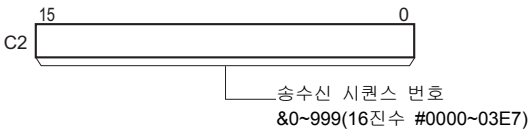
피연산자 설명

피연산자	내용	데이터 형식	크기
C1	제어 데이터 1	WORD	1
C2	제어 데이터 2	WORD	1
S	송신 데이터 시작 채널	INT	가변
D1	수신 데이터 저장 시작 채널	WORD	가변
D2	통신 정보	WORD	2

C1: 제어 데이터 1

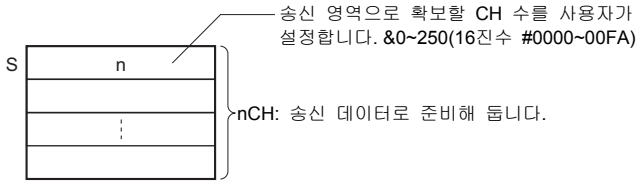


C2: 제어 데이터 2



■ 송신 영역 지정

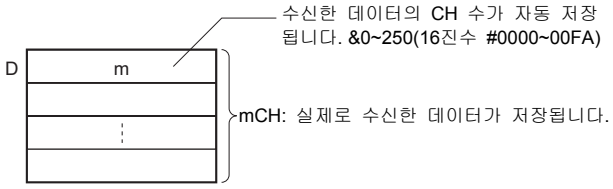
S: 송신 데이터 시작 채널

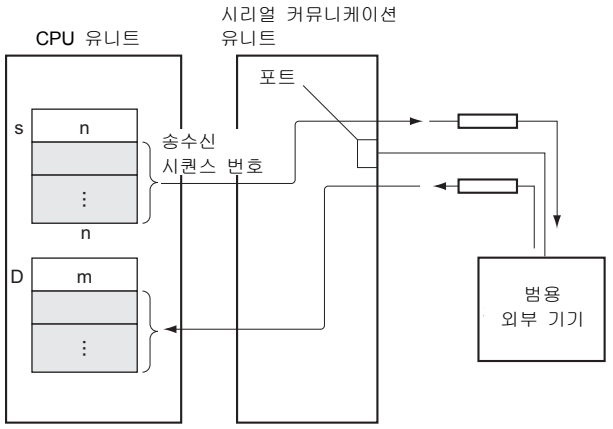


직접 지정, 링크 채널 지정 등, 실행하는 시퀀스 내에 피연산자를 지정하지 않는 경우는 S에 상수(#0000)를 지정하십시오. 상수 이외의 채널 등을 지정한 경우는 해당 채널의 값이 반드시 0000이 되어야 합니다. 다른 상수나 채널을 설정하면 에러(에러 플래그가 ON)가 되어 PMCR 명령이 실행되지 않습니다.

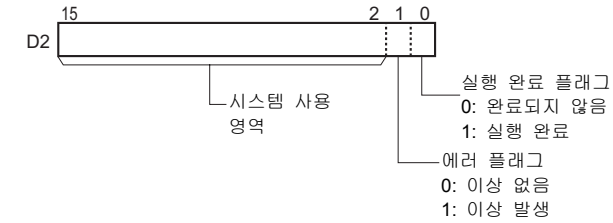
■ 수신 영역 지정

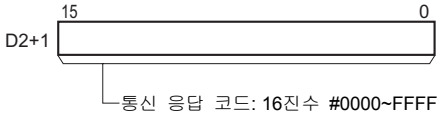
D1: 수신 데이터 저장 시작 채널
• 프로토콜 매크로 실행 완료 시





D2: 통신 정보





· PMCR2 명령 실행 전의 설정

D를 선두로 한 m으로 지정한 채널 수의 데이터를, 수신 실패 시의 수신 영역 유지 데이터로 설정하십시오.
m = 10진수 &2~250 또는 16진수 #0002~00FA가 가능. 단, m에 0 또는 1을 설정하면 수신 영역은 PMCR 명령 실행 시, 송수신 시퀀스 실행 직전에 0으로 클리어됩니다.

직접 지정, 링크 채널 지정 등, 실행하는 시퀀스 내에 피연산자를 지정하지 않는 경우는 D에 상수(#0000)를 지정하십시오. 상수 이외의 채널 등을 지정한 경우는 해당 채널의 값이 반드시 0000이 되어야 합니다.

■ 영역 종류

내용	채널 주소							간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
C1	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
C2																	
S																	
D1																	
D2										–	–						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C로 지정한 데이터가 범위 밖일 때 ON · S, D로 지정된 CH 수가 249CH를 초과했을 때(CH 지정 시) · 빈 내부 논리 포트가 없을(A211=0) 때 · 그 이외일 때 OFF

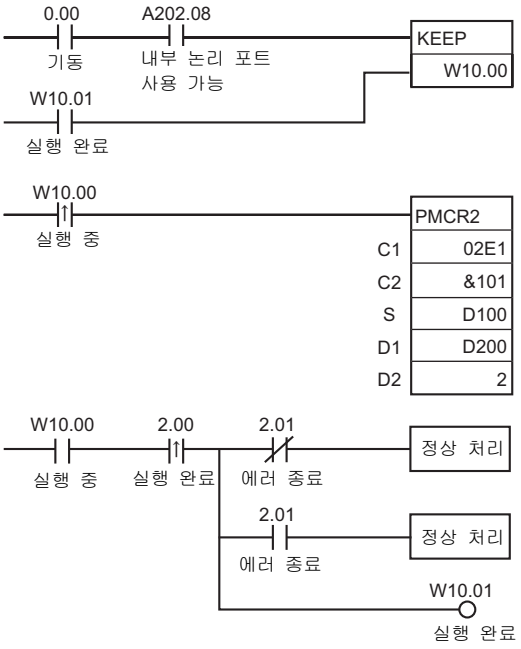
PMCR2

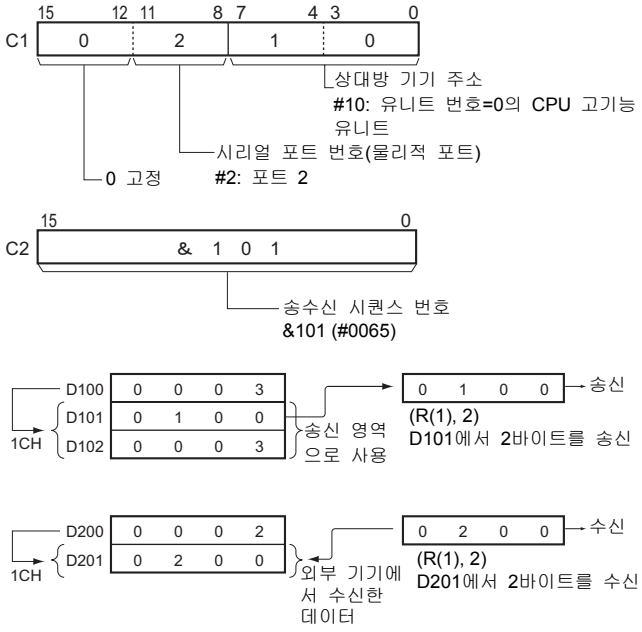
기능

시리얼 커뮤니케이션 유니트에 등록된 송수신 시퀀스를 호출하여 실행합니다.
C2로 지정한 시퀀스를 C1로 지정한 물리적 통신 포트에서 실행합니다. 송신 메시지 내의 변수가 피연산자 지정인 경우, S+1CH부터 S의 내용을 채널 수로 한 데이터가 송신 영역으로 사용됩니다. 수신 메시지 내의 변수가 피연산자 지정인 경우, D+1CH 이후에 수신하고 D에 해당 수신 데이터의 채널 수가 저장됩니다.
송신을 시작하면 D2의 실행 완료 플래그를 OFF로 합니다. 응답이 돌아오면 송신 결과의 정상/이상이 D2의 에러 플래그에 반영됩니다. 통신 응답은 D2＋1에 기록됩니다.

샘플 프로그램

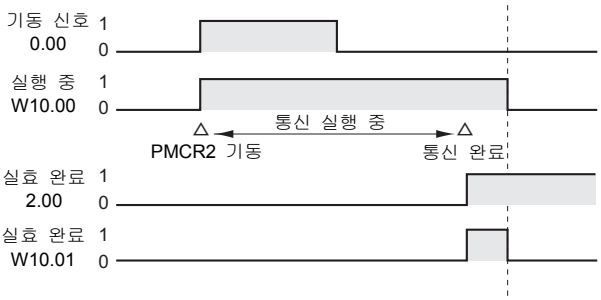
0.00(기동) 및 A202.08(내부 논리 포트 사용 가능)이 ON이면 W10.00(실행 중)이 ON이 됩니다. W10.00(실행 중)이 ON이면 송수신 시퀀스 번호 101(&101 또는 #0065)을 실행합니다. 송신 메시지 내의 변수가 피연산자 지정인 경우, D101부터 2CH(D100의 내용이 #0003)의 데이터를 송신 영역으로 사용합니다.
수신 메시지 내의 변수가 피연산자 지정인 경우, 수신한 데이터가 D201 이후에 저장되고, 수신한 데이터의 CH 수가 자동으로 D200에 저장됩니다(D200을 포함한 수).
정상일 때는 2.00(실행 완료)이 ON되고 정상 처리를 실행합니다. 이상일 때는 2.01(에러 종료)이 ON되고 이상 처리를 실행한 후 종료합니다.





주: 실제 송/수신은 위와 같이 송신 메시지 내 변수의 읽기 옵션 R( ) 및 수신 메시지 내 변수의 쓰기 옵션 W( )로 구성됩니다.

■ 타임 차트



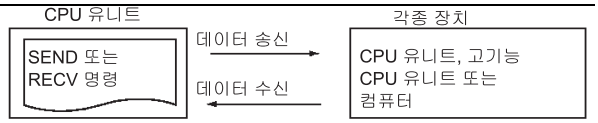
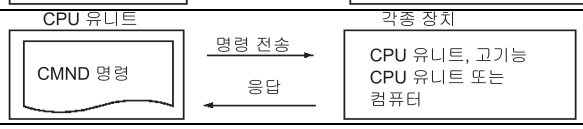
네트워크 통신용 명령

니모닉	명령 이름	펑션 번호	페이지
SEND	네트워크 송신	090	3-774
SEND2	네트워크 송신 2	491	3-779
RECV	네트워크 수신	098	3-783
RECV2	네트워크 수신 2	492	3-788
CMND	명령 송신	490	3-791
CMND2	명령 송신 2	493	3-798
EXPLT	범용 Explicit 메시지 송신 명령	720	3-802
EGATR	Explicit 읽기 명령	721	3-807
ESATR	Explicit 쓰기 명령	722	3-812
ECHRD	Explicit CPU 유니트 데이터 읽기 명령	723	3-816
ECHWR	Explicit CPU 유니트 데이터 쓰기 명령	724	3-819

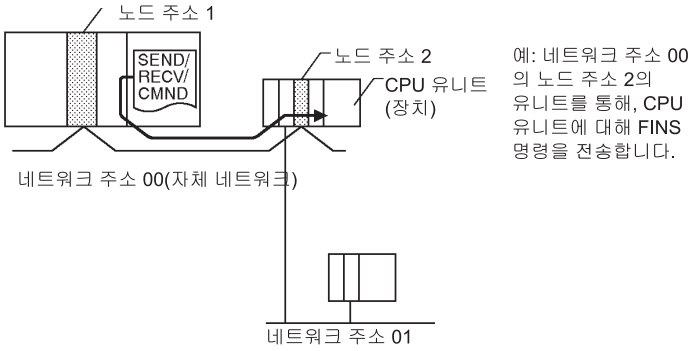
네트워크 통신 명령의 개요

네트워크 또는 CPU 장치의 각종 장치(CPU 유니트, CPU 고기능 유니트, 컴퓨터 등)에 대해 필요할 때(조건 성립 시) 데이터 송수신 또는 제어(모드 변경 등)를 수행하기 위한 명령입니다.

네트워크 통신 명령에는 다음 2종류가 있습니다.

메시지 내용	네트워크 통신 명령	내용
데이터 송/수신용 명령 전송 (FINS 명령)	SEND/SEND2/RECV/ RECV2 명령	
임의의 명령 전송 (FINS 명령)	CMND/CMND2 명령	

위의 네트워크 통신 명령으로 전송하는 명령은 'FINS 명령'이라고 하며, FA 제어 장치용 통신 명령입니다. 'FINS 명령'에 대한 자세한 내용은 'SYSMAC CS/CJ 시리즈 통신 명령 참조 설명서'(SBCA-304)를 참조하십시오. 'FINS 명령'은 '어느 네트워크의 어느 노드의 어느 기기 주소(장치)에 대해' 명령을 전송할지 지정하는 것만으로 상대가 네트워크에 있든지, CPU 장치에 있든지 상관 없이 자유롭게 동일한 통신(명령 전송/응답 수신)을 수행할 수 있습니다.

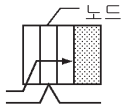
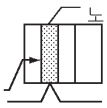
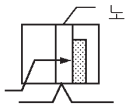


네트워크 주소: 네트워크의 주소(자체 네트워크 = 16진수 #00)

노드 주소: 네트워크상의 논리 주소

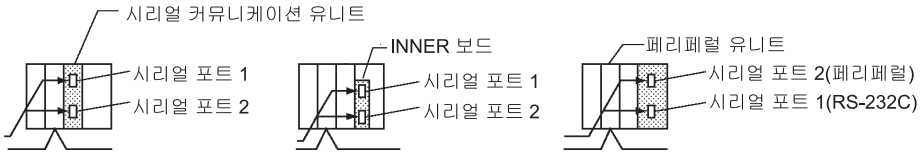
기기 주소: 상대 장치 식별용 주소

- CPU 유니트: 16진수 #00
- CPU 고기능 유니트: 16진수 #10 + 유니트 번호(16진수)
- 고기능 I/O 유니트: 16진수 #20 + 기기 번호(16진수)
- INNER 보드: 16진수 #E1
- 컴퓨터: 16진수 #01

	CPU 유니트	CPU 고기능 유니트	INNER 보드	컴퓨터
기기 주소	#00	유니트 번호+ # 10	#E1	#01
내용				

■ 기기 주소

기기 주소에는 상대 장치의 시리얼 포트(번호 1~4)를 지정할 수도 있습니다.



- 시리얼 포트의 기기 주소:
  - 시리얼 커뮤니케이션 유니트의 경우

    포트 1: 16진수 #80 + 16진수 #04×유니트 번호(16진수)

유니트 번호	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
16진수	80	84	88	8C	90	94	98	9C	A0	A4	A8	AC	B0	B4	B8	BC
10진수	128	132	136	140	144	148	152	156	160	164	168	172	176	180	184	188

    포트 2: 16진수 #81 + 16진수 #04×유니트 번호(16진수)

유니트 번호	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
16진수	81	85	89	8D	91	95	99	9D	A1	A5	A9	AD	B1	B5	B9	BD
10진수	129	133	137	141	145	149	153	157	161	165	169	173	177	181	185	189

  - 시리얼 커뮤니케이션 보드의 경우

    포트 1: 16진수 #E4(10진수로 228)

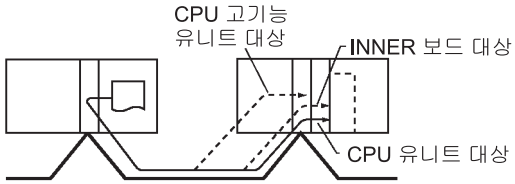
    포트 2: 16진수 #E5(10진수로 229)
- CPU 유니트의 경우

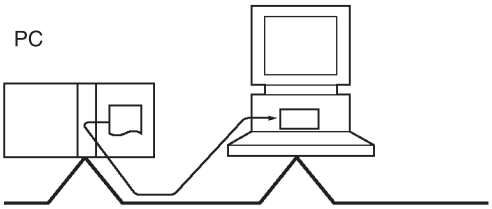
  페리페럴 포트: 16진수 #FD(10진수로 253)

  RS-232C 포트: 16진수 #FC(10진수로 252)

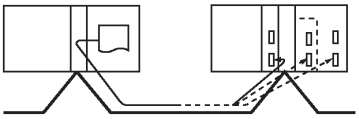
네트워크 통신 패턴

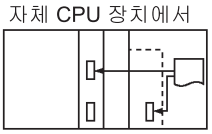
- 네트워크상의 장치에 대한 PLC
- CPU 유니트 대상
- CPU 고기능 유니트 대상
- INNER 보드 대상





- 네트워크상의 장치의 시리얼 포트에 대한 PLC
- CPU 유니트, CPU 고기능 유니트, INNER 보드의 각 시리얼 포트 대상

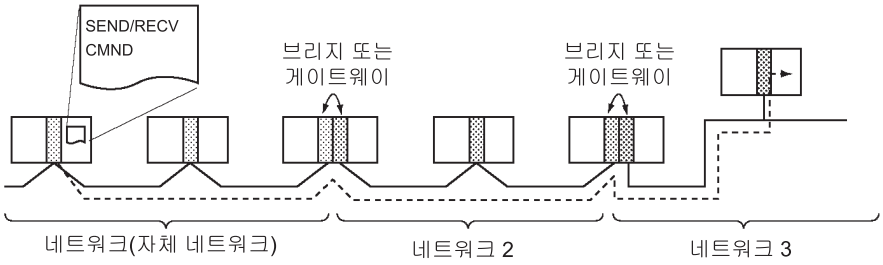




자세한 내용은 각 네트워크(Controller Link, Ethernet)용 유니트의 사용 설명서를 참조하십시오.

■ 네트워크

자체 네트워크(데이터를 발신하는 네트워크)를 포함하여 8계층까지 네트워크를 벗어날 수 있습니다.

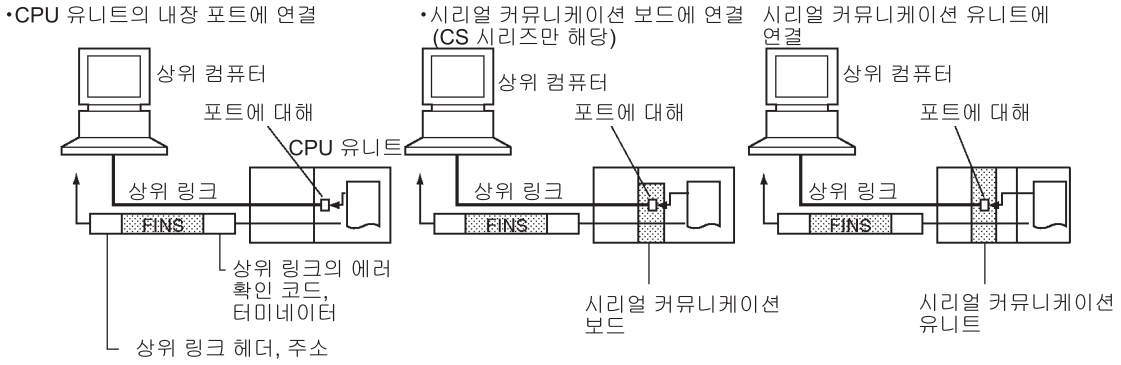


네트워크를 벗어나 통신하기 위해서는 목표로 하는 네트워크상의 노드에 도달하기 위한 경로를 나타낸 테이블을 각 PLC(CPU 유니트)에 등록해야 합니다. 이 테이블을 '라우팅 테이블'이라고 합니다. 라우팅 테이블은 다음 2종류의 테이블로 구성됩니다.

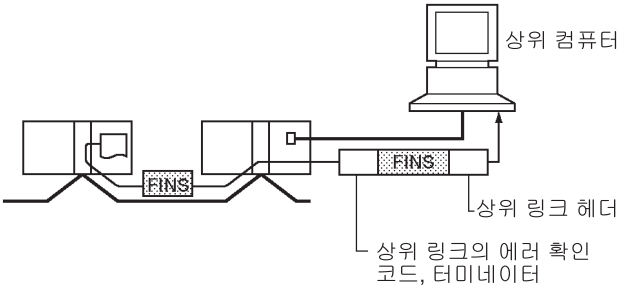
- 자체 네트워크 테이블 : 자체 PLC에 장착된 유니트 번호와 네트워크 주소의 대응표
- 중계 네트워크 테이블 : 자체 PLC에는 연결되어 있지 않지만 목표로 하는 네트워크(최종 네트워크)와 거기에 이르는 경로의 최초 중계점(먼저 가야 할 곳)의 네트워크 주소와 노드 주소의 대응표

상위 링크로 연결한 상위 컴퓨터(송신 기능)

상위 링크 모드의 시리얼 포트에 대해 SEND/SEND2, RECV/RECV2, CMND/CMND2 명령을 실행하면 FINS 명령을 상위 링크의 헤더, 터미네이터 등에 포함시켜 상위 컴퓨터에 송신합니다.

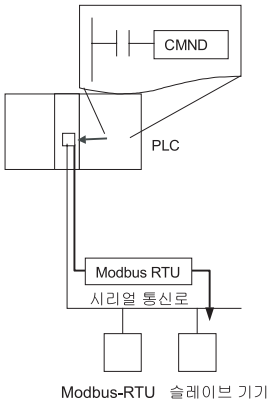


주: 네트워크를 통해 위의 송신 기능을 구현할 수도 있습니다. 이 경우, 네트워크로는 FINS 명령이 전달되고 상위 링크로는 상위 링크의 헤더, 터미네이터로 감싼 FINS 명령이 전달됩니다.

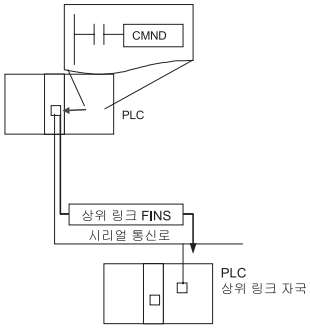


시리얼 포트(시리얼 게이트웨이 기능)에 연결된 기기(구성 요소 또는 상위 링크 자국의 PLC)에 대한 FINS 명령 전송(또는 데이터 송수신)

- 시리얼 게이트웨이 기능을 지원하는 시리얼 포트에 대해 CMND/CMND2 명령을 실행하고, 시리얼 게이트웨이 기능을 통해 CompoWay/F, Modbus-RTU, Modbus-ASCII 명령으로 변환하는 것도 가능합니다.

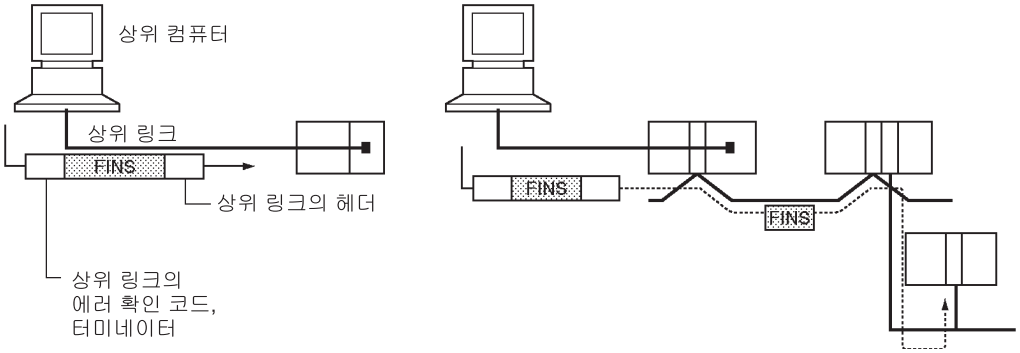


- 시리얼 게이트웨이 기능을 지원하는 시리얼 포트에 대해 CMND/CMND2 명령 또는 SEND/SEND2, RECV/RECV2 명령을 실행하고, 시리얼 게이트웨이 기능을 통해 임의의 FINS 명령을 전송하거나 데이터를 송수신하는 것도 가능합니다.



■ 상위 링크로 연결한 상위 컴퓨터에서 FINS 명령 전송

송신 기능과 반대로, 상위 링크로 연결한 상위 컴퓨터에서 직접 연결되어 있는 PLC(CPU 유니트) 또는 네트워크상의 장치(CPU 유니트, 유니트, PC 등)에 대해 FINS 명령을 전송하는 것도 가능합니다. 이 경우도 FINS 명령을 상위 링크의 헤더, 터미네이터, 에러 코드 확인에 포함시켜 전송합니다.



네트워크 통신용 명령

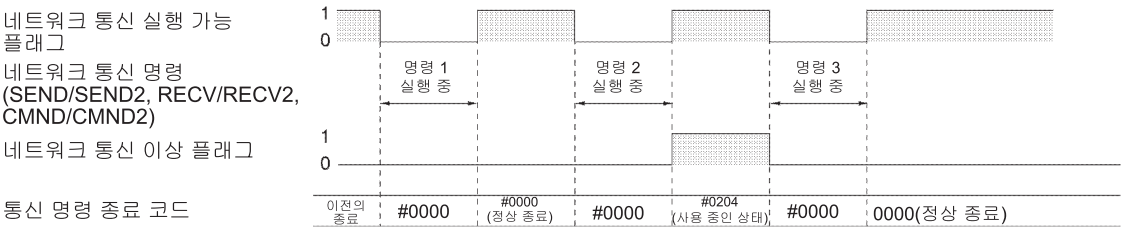
CS/CJ 시리즈는 동시에 최대 8개까지 명령에 의한 네트워크 통신이 가능합니다. CJ2 시리즈는 최대 64개까지 동시에 통신할 수 있는 'CJ2 전용 네트워크 통신용 명령'도 있습니다.

	니모닉	명령 이름	특징
CJ2 전용 네트워크 통신용 명령	SEND2	네트워크 송신	최대 64개까지 동시 사용 가능. 통신 완료 및 에러 검출은 명령으로 지정한 채널을 이용합니다.
	RECV2	네트워크 수신	
	CMND2	명령 송신	
	PMCR2	프로토콜 매크로	
CJ1/CS1/CJ2 공통 네트워크 통신용 명령	SEND	네트워크 송신	최대 8개까지 동시 사용 가능. 통신 완료 및 에러 검출은 AR (특수 보조 릴레이)를 이용합니다. 내부 논리 포트를 지정하거나 내부 논리 포트 자동 할당 기능을 사용해야 합니다.
	RECV	네트워크 수신	
	CMND	명령 송신	
	EXPLT	범용 Explicit 메시지 송신	
	EGATR	Explicit 읽기	
	ESATR	Explicit 쓰기	
	ECHRD	Explicit CPU 유니트 데이터 읽기	
	ECHWR	Explicit CPU 유니트 데이터 쓰기	
	PMCR	프로토콜 매크로	
	TXDU	시리얼 커뮤니케이션 유니트의 시리얼 포트 출력	
	RXDU	시리얼 커뮤니케이션 유니트 시리얼 포트 입력	

통신 플래그의 동작

- 네트워크 통신 실행 가능 플래그는 송수신 실행 중에 0(OFF), 실행 종료(정상이든 이상이든) 후에는 1(ON)이 됩니다.
- 네트워크 통신 이상 플래그는 다음 송수신 실행까지 상태를 유지합니다.
- 네트워크 통신 이상 플래그는 이상 종료가 발생해도 다음 통신 명령 실행에서 0(OFF)이 됩니다.

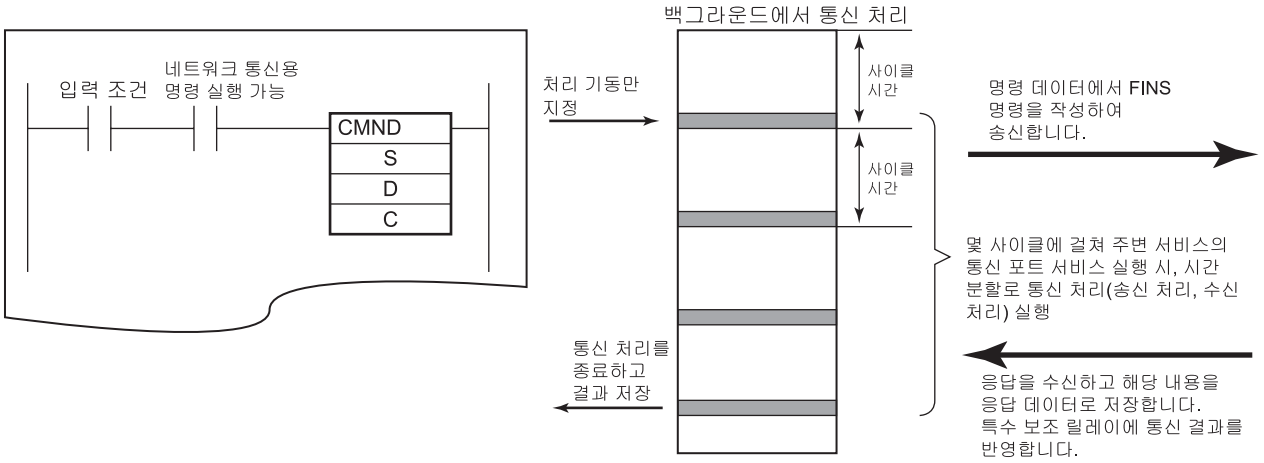
(예)



주소	비트	이름	내용
A202	00~07	네트워크 통신 명령 실행 가능 플래그	통신 명령이 실행 가능한 경우 1(ON)이 됩니다. 각 비트 00~07이 내부 논리 포트 번호 0~7에 대응합니다. 통신 명령 실행 후 이 플래그가 OFF→ON되면 네트워크 통신의 완료를 확인할 수 있습니다. 통신 명령 실행 시에는 0(OFF)이 됩니다.
	15	네트워크 통신 자동 할당 가능 플래그	내부 논리 포트 자동 할당에서 통신 명령을 실행할 수 있는 경우, 즉 자동 할당으로 사용할 수 있는 빈 내부 논리 포트가 있는 경우 1(ON)이 됩니다. 참고: 동시에 9개 이상의 통신 명령을 사용하는 경우, 이 플래그를 통해 자동 할당으로 사용할 수 있는 빈 내부 논리 포트가 있는지 확인한 후 통신 명령을 실행합니다.
A203~ A210CH	–	네트워크 통신 응답 코드	네트워크 통신이 실행된 경우의 응답 코드가 설정됩니다.
A214CH	00~07	네트워크 통신 완료 시 1사이클 ON 플래그	통신 완료 시 1사이클만 1(ON)이 됩니다. 각 비트 00~07이 내부 논리 포트 번호 0~7에 대응합니다. 주: 이 플래그는 통신 명령을 실행한 다음 사이클부터 유효가 되므로 통신 명령을 실행한 사이클에서 참조하지 않도록 래더 프로그램에서 1사이클 대기한 후 참조해야 합니다. 참고: A218CH(사용 내부 논리 포트 번호)에 저장된 사용 내부 논리 포트 번호를 바탕으로 비트 00~07 중 하나를 지정하여 참조합니다.
	08~15	사용 금지	–
A215CH	00~07	네트워크 통신 완료 시 1사이클 ON 에러 플래그	통신 완료 시 에러이면 1사이클만 1(ON)이 됩니다. 각 비트 00~07이 내부 논리 포트 번호 0~7에 대응합니다. 이 플래그가 1(ON)이 된 경우, 네트워크 통신 응답 코드(A203~A210CH)를 참조하여 에러 요인을 확인하십시오. 주: 이 플래그는 통신 명령을 실행한 다음 사이클부터 유효가 되므로 통신 명령을 실행한 사이클에서 참조하지 않도록 래더 프로그램에서 1사이클 대기한 후 참조해야 합니다. 참고: A218CH(사용 내부 논리 포트 번호)에 저장된 사용 내부 논리 포트 번호를 바탕으로 비트 00~07 중 하나를 지정하여 참조합니다.
	08~15	사용 금지	–
A216~ A217CH	–	네트워크 통신 응답 코드 저장 주소	통신 명령을 실행한 경우, 특수 보조 릴레이의 임의의 채널 주소에 통신 응답 코드가 자동으로 저장됩니다. 해당 임의의 채널 주소의 I/O 메모리 실제 주소가 여기에 저장됩니다. 참고: 여기에 저장된 I/O 메모리 실제 주소를 인덱스 레지스터 IR에 전송하여 인덱스 레지스터 간접 지정을 통해 임의의 채널 주소에 저장된 통신 응답 코드를 읽을 수 있습니다.
A218CH	–	사용 내부 논리 포트 번호	통신 명령을 실행한 경우, 사용한 내부 논리 포트 번호가 여기에 저장됩니다. 0~7: 내부 논리 포트 번호 0~7
A219CH	00~07	네트워크 통신 실행 에러 플래그	네트워크 통신 실행 중에 에러가 발생한 경우, 1(ON)이 됩니다. 이 플래그가 1(ON)이 된 경우, 네트워크 통신 응답 코드(A203~A210CH)를 참조하여 에러 요인을 확인하십시오. 각 비트 00~07이 내부 논리 포트 번호 0~7에 대응합니다. 통신 명령 실행 시에는 0(OFF)이 됩니다.

네트워크 통신용 명령의 실행 타이밍

네트워크 통신용 명령에서는 입력 조건이 성립하면 통신 처리를 기동하기만 하며, 실제 통신 처리는 백그라운드에서 주변 서비스 처리로 실행됩니다.

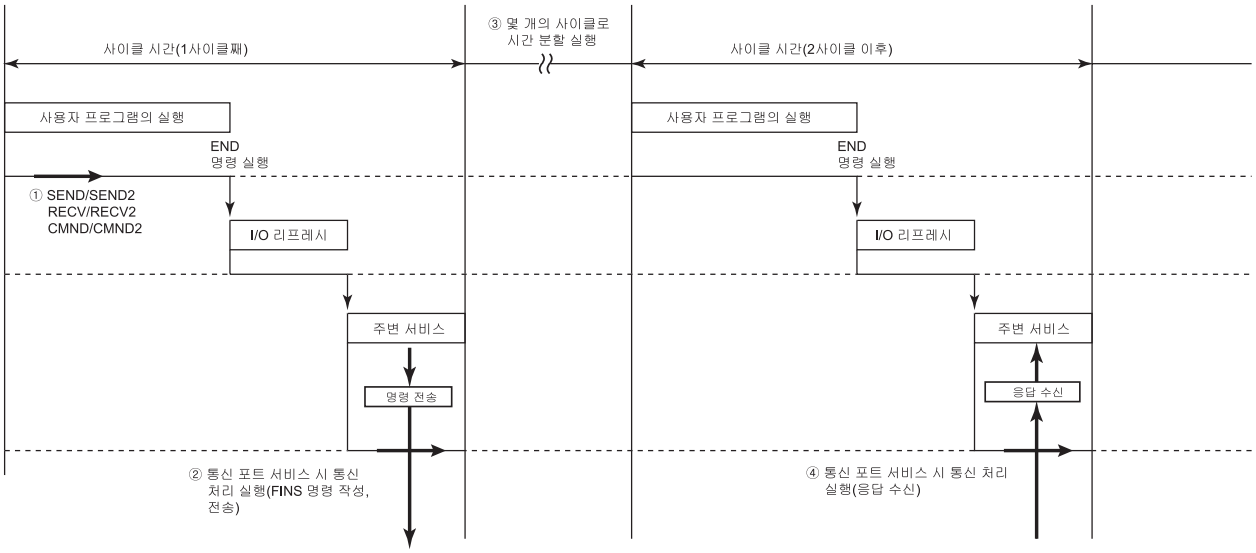


다음 동작이 실행됩니다.

① 입력 조건이 성립하고 이 시점의 네트워크 통신 명령 실행 가능 플래그(A202.00~A202.07)가 1(ON)이면 각 명령은 네트워크 명령 실행 가능 플래그(A202.00~A202.07)를 0(OFF), 네트워크 통신 실행 에러 플래그(A219.00~A219.07)를 0(OFF), 네트워크 통신 응답 코드(A203~A210CH)를 #0000으로 하고, C를 읽어 통신 처리(FINS 명령 전송/응답 수신 처리)를 기동합니다.

② 주변 서비스 처리로 통신 명령 처리를 실행합니다. 이 처리는 여러 사이클에 걸쳐 실행됩니다.

③ 응답이 돌아오면 통신 포트 서비스 내에서 피연산자로 지정된 응답 데이터를 갱신합니다. 이 타이밍에서 특수 보조 릴레이의 네트워크 통신 명령 실행 가능 플래그(A202.00~A202.07)를 1(ON)로 하고, 네트워크 통신 실행 에러 플래그(A219.00~A219.07), 네트워크 통신 응답 코드(A203~A210CH)를 갱신합니다.

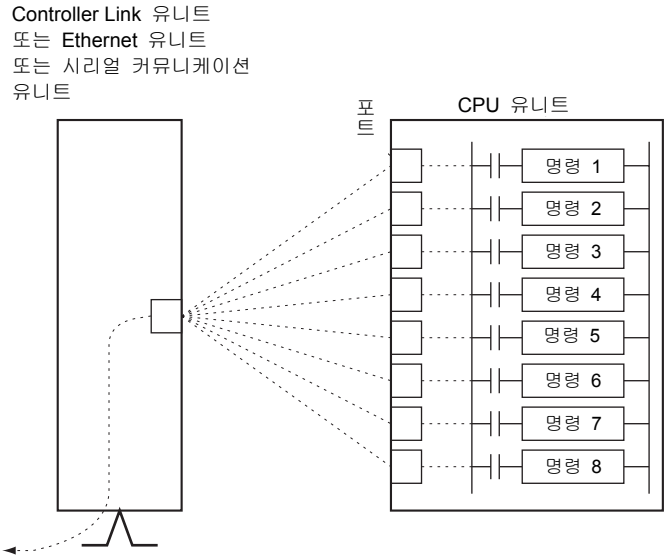


CJ1/CS1/CJ2 공통 네트워크 통신용 명령

내부 논리 포트

8개의 논리상의 통신 포트인 내부 논리 포트가 있습니다. 8개의 통신 명령을 동시에 실행할 수 있지만 1개의 내부 논리 포트는 한번에 1개 명령만 실행할 수 있습니다. 9개 이상의 통신 명령을 실행할 때는 배타적 제어를 구성해야 합니다.

이 내부 논리 포트는 네트워크 통신 명령(SEND/SEND2, RECV/RECV2, CMND/CMND2 등), 시리얼 통신 명령(TXDU, RXDU) 및 프로토콜 매크로 명령(PMCR/PMCR2)과 공용입니다. 따라서 이들 명령에서 중복된 포트를 지정할 수 없으므로 주의하십시오.

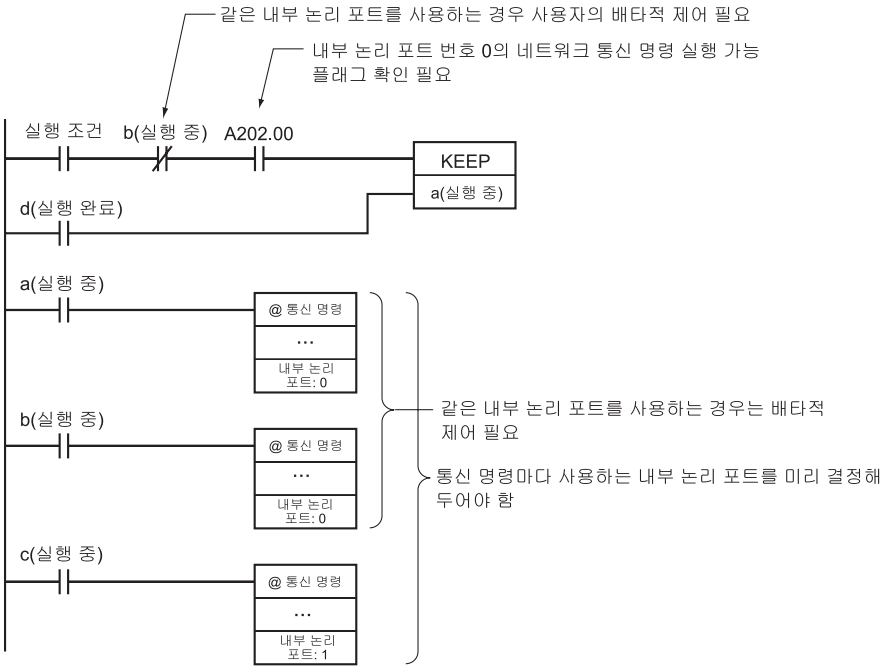


CJ1/CS1/CJ2 공통 네트워크 통신용 명령은 내부 논리 포트를 지정하는 경우와 CPU 유니트가 자동으로 할당되는 경우의 2종류가 있습니다.

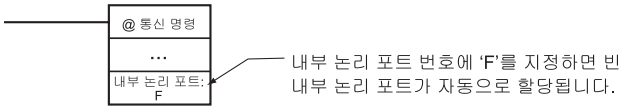
각 내부 논리 포트는 동시에 사용할 수 없으므로 내부 논리 포트를 지정하는 경우, 다음 중 하나 또는 양쪽 처리가 필요합니다.

- 프로그래밍할 때 빈 내부 논리 포트 번호를 확인하면서 명령의 피연산자에 사용 내부 논리 포트 번호를 지정
- 래더 프로그램에서 명령을 실행하기 전에 해당 내부 논리 포트 번호가 비어 있는지 여부를 확인하는 처리를 추가

자동 할당 기능을 사용하지 않을 때(예)



그에 비해 CJ2 CPU 유니트 및 로트 번호 020601(2002년 6월 1일 제조) 이후의 CS1-H/CJ1-H/CJ1M/CS1D(CPU 단독 시스템용) CPU 유니트의 경우, 내부 논리 포트 번호로 '0'~'7'이 아니라 'F'를 지정하면 해당 시점에서 비어 있는 내부 논리 포트 번호를 자동으로 검출하여 할당해서 사용할 수 있습니다.



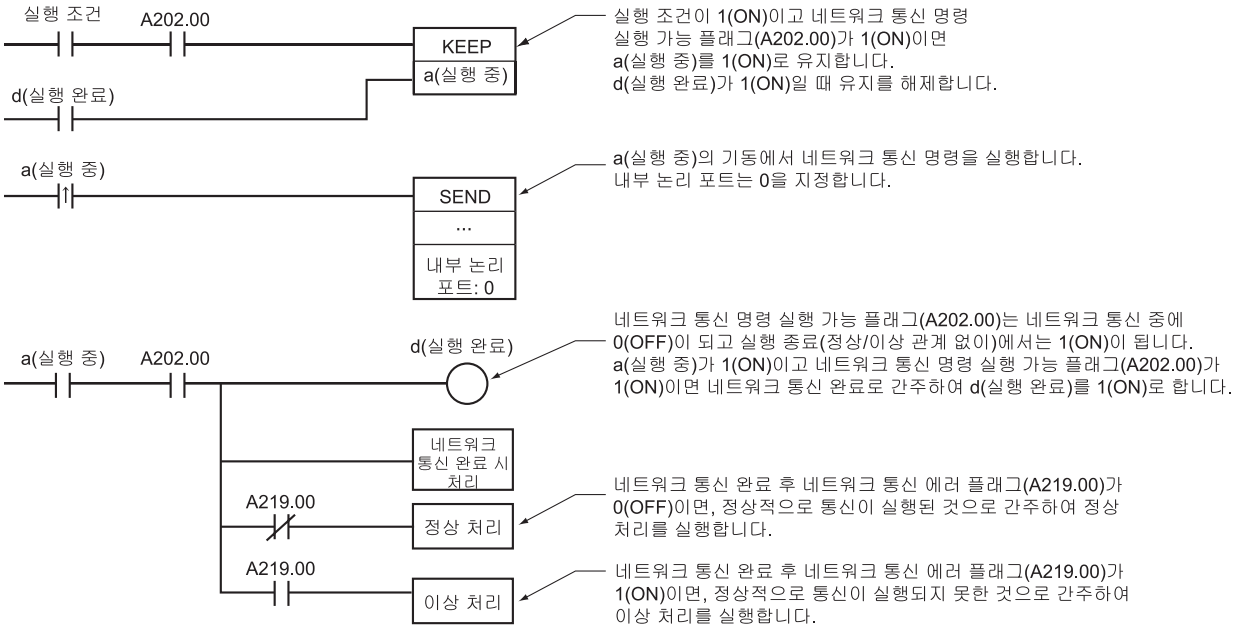
자동 할당 기능을 사용하면 내부 논리 포트를 고려하여 프로그래밍할 필요가 없습니다.

다음과 같은 차이점이 있습니다.

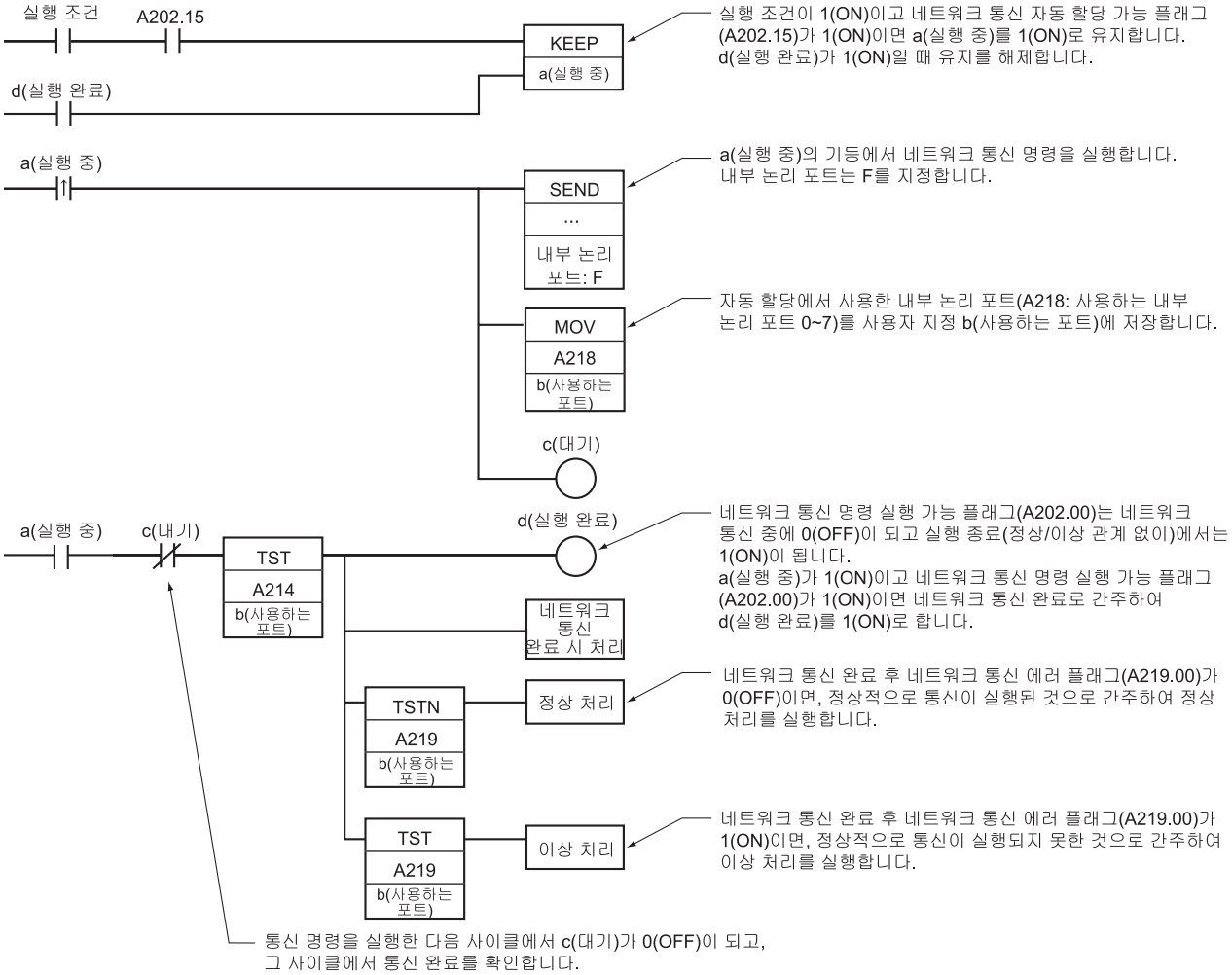
항목	내부 논리 포트 사용자 지정 시	내부 논리 포트 자동 할당 시
제어 데이터 내 '내부 논리 포트 번호' 지정	"0"~"7"	"F"
배타적 제어	필요	필요 없음 단, 동시에 9개 이상 통신 명령이 동작할 가능성이 있는 경우는 배타적 제어가 필요합니다.
각종 플래그 사용 방법	지정된 내부 논리 포트에 대응하는 플래그를 LD/LD NOT 명령으로 판별합니다.	'사용 내부 논리 포트 번호'(A218CH)와 TST/TSTN 명령을 사용하여 판별합니다.
네트워크 통신 응답 코드	사용자 지정된 내부 논리 포트에 대응하는 응답 코드를 참조합니다.	'네트워크 통신 응답 코드 저장 주소'(A216CH/217CH에 I/O 메모리 실제 주소 저장)와 인덱스 레지스터 간접 지정을 사용하여 응답 코드를 참조합니다.

래더 프로그램 예

■ 내부 논리 포트를 지정한 경우



■ 내부 논리 포트를 자동 할당으로 한 경우



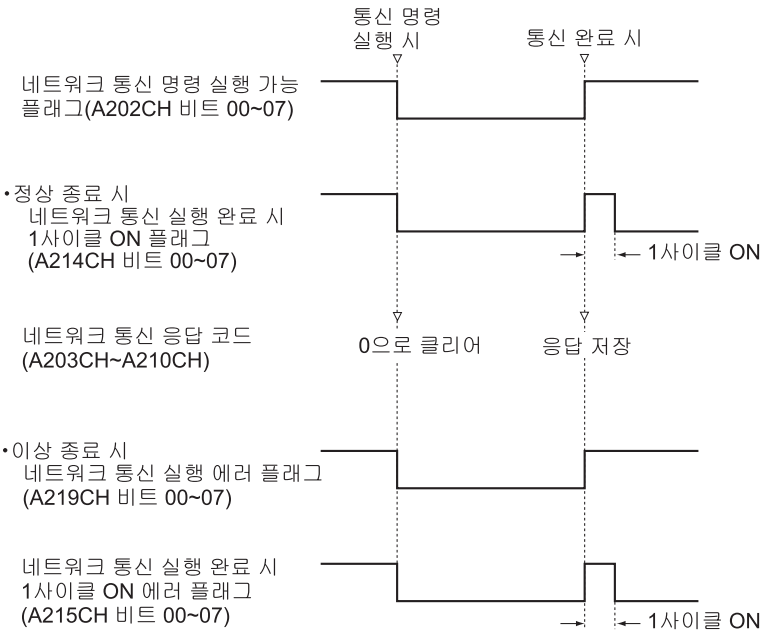
내부 논리 포트를 사용자가 지정하는 경우

네트워크 통신용 명령의 제어 데이터에서 사용하는 내부 논리 포트를 지정합니다.
명령이 처리하는 동안 같은 내부 논리 포트를 다른 명령에서 사용할 수 없습니다.
따라서 배타적 제어가 필요합니다.

■ 내부 논리 포트 사용자 지정에서 사용하는 특수 보조 릴레이

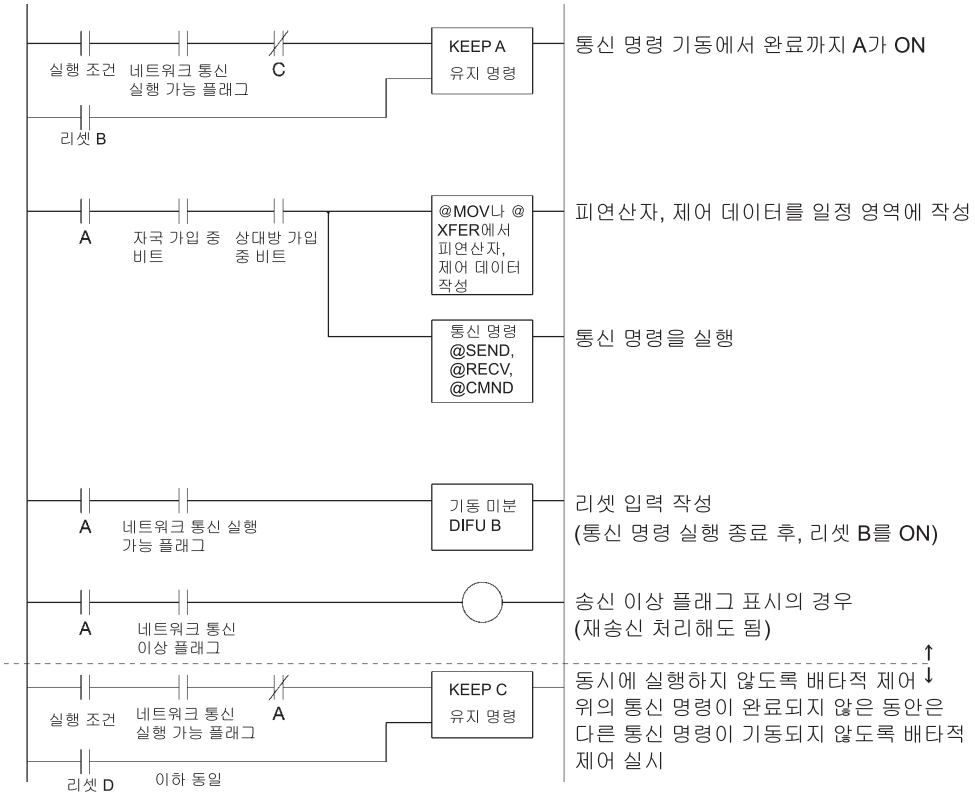
주소	비트	이름	내용
A202	00~07	네트워크 통신 명령 실행 가능 플래그	통신 명령이 실행 가능한 경우 1(ON)이 됩니다. 각 비트 00~07이 내부 논리 포트 번호 0~7에 대응합니다. 통신 명령 실행 후 이 플래그가 OFF→ON되면 네트워크 통신의 완료를 확인할 수 있습니다. 통신 명령 실행 시에는 0(OFF)이 됩니다.
A203~ A210CH	–	네트워크 통신 응답 코드	네트워크 통신이 실행된 경우의 응답 코드가 설정됩니다.
A219CH	00~07	네트워크 통신 실행 에러 플래그	네트워크 통신 실행 중에 에러가 발생한 경우, 1(ON)이 됩니다. 이 플래그가 1(ON)이 된 경우, 네트워크 통신 응답 코드(A203~A210CH)를 참조하여 에러 요인을 확인하십시오. 각 비트 00~07이 내부 논리 포트 번호 0~7에 대응합니다. 통신 명령 실행 시에는 0(OFF)이 됩니다.

● 각종 플래그의 동작



사용 방법

1. 명령의 피연산자에서 내부 논리 포트로 16진수 #0~7을 지정합니다.
2. 네트워크 통신 실행 가능 플래그(A202.00~A202.07)를 사용하여 배타적 제어를 구성한 후 사용하십시오.



내부 논리 포트의 자동 할당 기능

■ 개요

다음 명령은 내부 논리 번호 0~7의 8개 중 하나를 사용하여 네트워크 통신 또는 시리얼 통신을 실행합니다.

· 네트워크 통신용 명령(SEND, RECV, CMND)

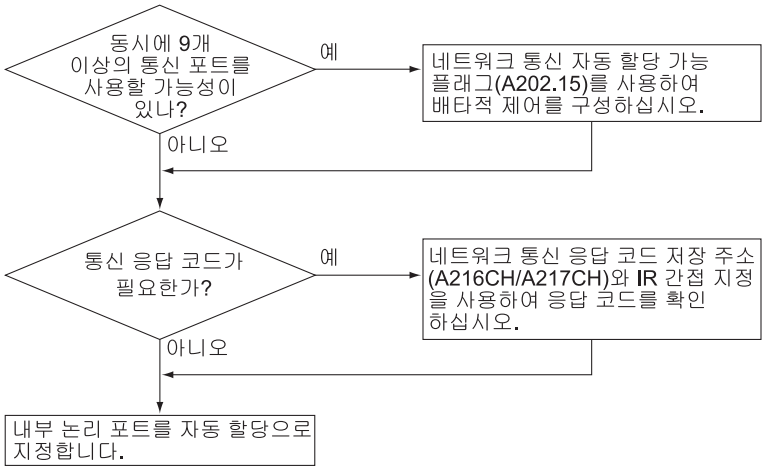
· 시리얼 통신 명령(PMCR/PMCR2, TXDU, RXDU)

(아래의 설명에서는 위의 명령을 총칭하여 '통신 명령'이라고 기술합니다.)

■ 내부 논리 포트 자동 할당에서 사용하는 특수 보조 릴레이

주소	비트	이름	내용
A202CH	15	네트워크 통신 자동 할당 가능 플래그	내부 논리 포트 자동 할당에서 통신 명령을 실행할 수 있는 경우, 즉 자동 할당으로 사용할 수 있는 빈 내부 논리 포트가 있는 경우 1(ON)이 됩니다. 참고: 동시에 9개 이상의 통신 명령을 사용하는 경우, 이 플래그를 통해 자동 할당으로 사용할 수 있는 빈 내부 논리 포트가 있는지 확인한 후 통신 명령을 실행합니다.
A214CH	00~07	네트워크 통신 완료 시 1사이클 ON 플래그	통신 완료 시 1사이클만 1(ON)이 됩니다. 각 비트 00~07이 내부 논리 포트 번호 0~7에 대응합니다. 주: 이 플래그는 통신 명령을 실행한 다음 사이클부터 유효가 되므로 통신 명령을 실행한 사이클에서 참조하지 않도록 래더 프로그램에서 1사이클 대기한 후 참조해야 합니다. 참고: A218CH(사용 내부 논리 포트 번호)에 저장된 사용 내부 논리 포트 번호를 바탕으로 비트 00~07 중 하나를 지정하여 참조합니다.
	08~15	사용 금지	–
A215CH	00~07	네트워크 통신 완료 시 1사이클 ON 에러 플래그	통신 완료 시 에러이면 1사이클만 1(ON)이 됩니다. 각 비트 00~07이 내부 논리 포트 번호 0~7에 대응합니다. 이 플래그가 1(ON)이 된 경우, 네트워크 통신 응답 코드(A203~A210CH)를 참조하여 에러 요인을 확인하십시오. 주: 이 플래그는 통신 명령을 실행한 다음 사이클부터 유효가 되므로 통신 명령을 실행한 사이클에서 참조하지 않도록 래더 프로그램에서 1사이클 대기한 후 참조해야 합니다. 참고: A218CH(사용 내부 논리 포트 번호)에 저장된 사용 내부 논리 포트 번호를 바탕으로 비트 00~07 중 하나를 지정하여 참조합니다.
	08~15	사용 금지	–
A216~ A217CH	–	네트워크 통신 응답 코드 저장 주소	통신 명령을 실행한 경우, 특수 보조 릴레이의 임의의 채널 주소에 통신 응답 코드가 자동으로 저장됩니다. 해당 임의의 채널 주소의 I/O 메모리 실제 주소가 여기에 저장됩니다. 참고: 여기에 저장된 I/O 메모리 실제 주소를 인덱스 레지스터 IR에 전송하여 인덱스 레지스터 간접 지정을 통해 임의의 채널 주소에 저장된 통신 응답 코드를 읽을 수 있습니다.
A218CH	–	사용 내부 논리 포트 번호	통신 명령을 실행한 경우, 사용한 내부 논리 포트 번호가 여기에 저장됩니다. 0~7: 내부 논리 포트 번호 0~7

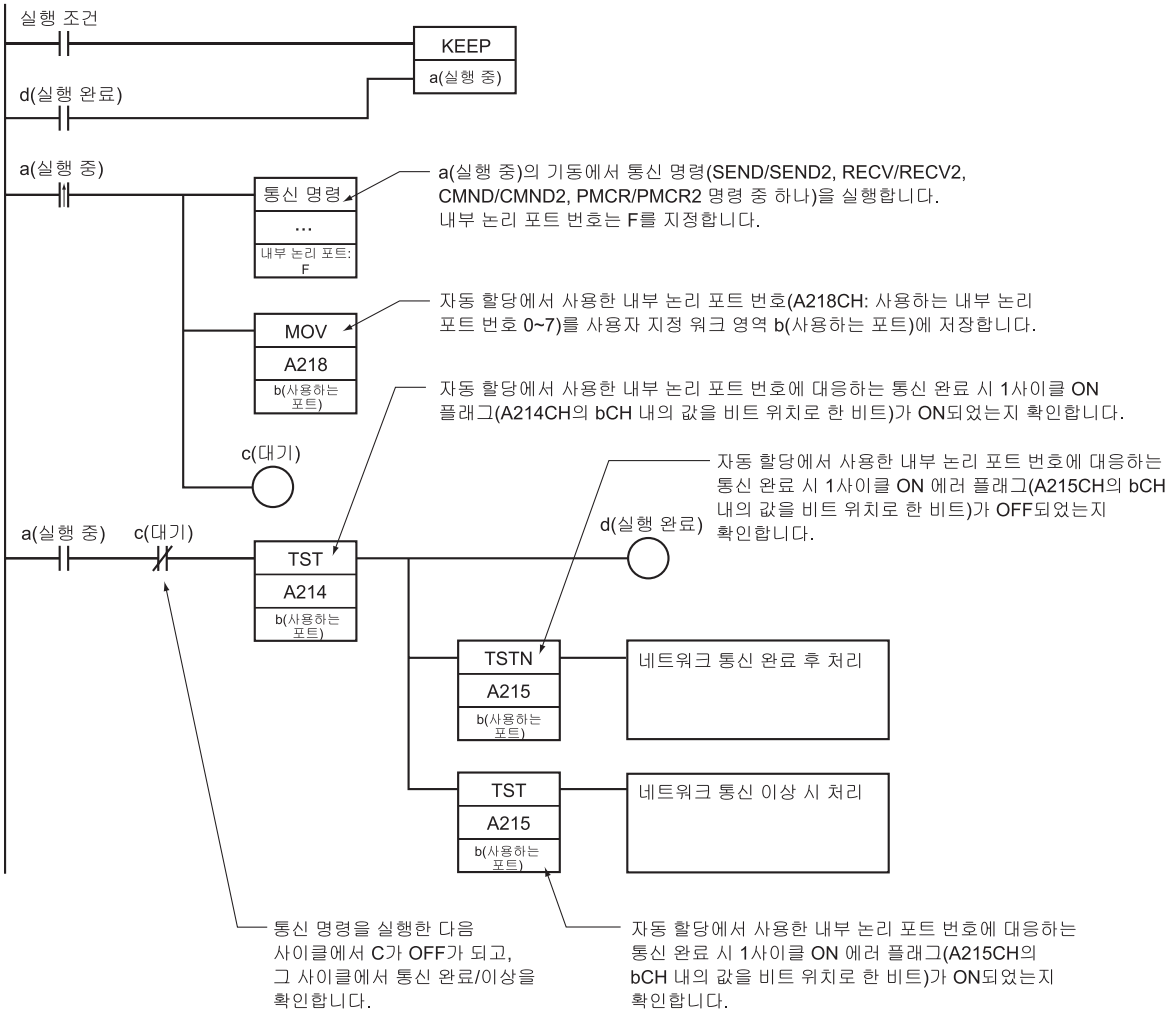
네트워크 통신 자동 할당 가능 플래그(A202.15), 네트워크 통신 응답 코드 저장 주소(A216CH/A217CH)를 사용할지 여부는 다음과 같이 판단합니다.



사용 방법

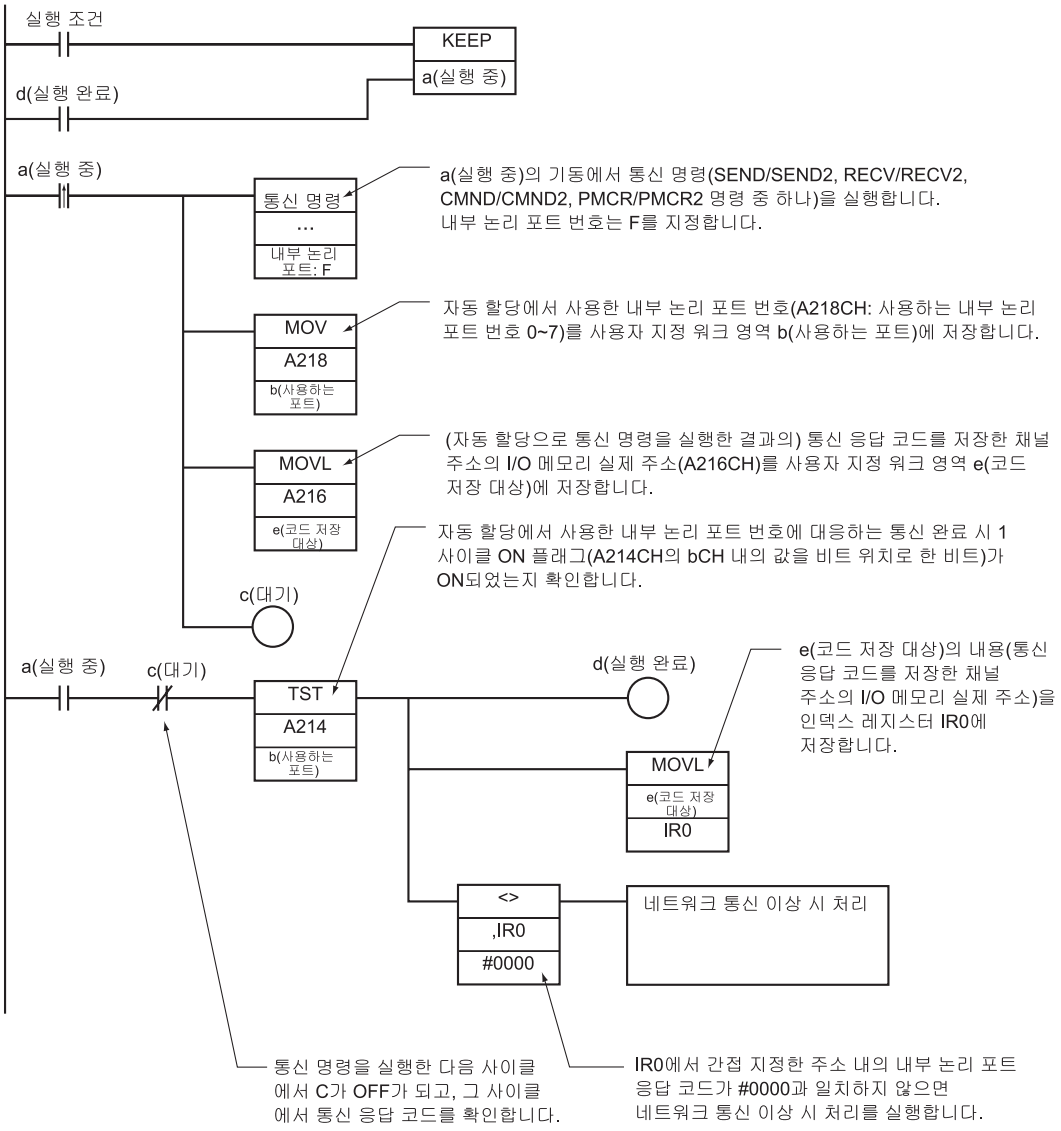
1. 명령의 피연산자에서 내부 논리 포트 번호로 16진수 #F를 지정합니다.
2. 반드시 다음과 같은 래더 프로그램을 구성하여 사용하십시오.

■ 통신 명령 실행 후 네트워크 통신 완료/이상 시 처리를 수행하는 경우



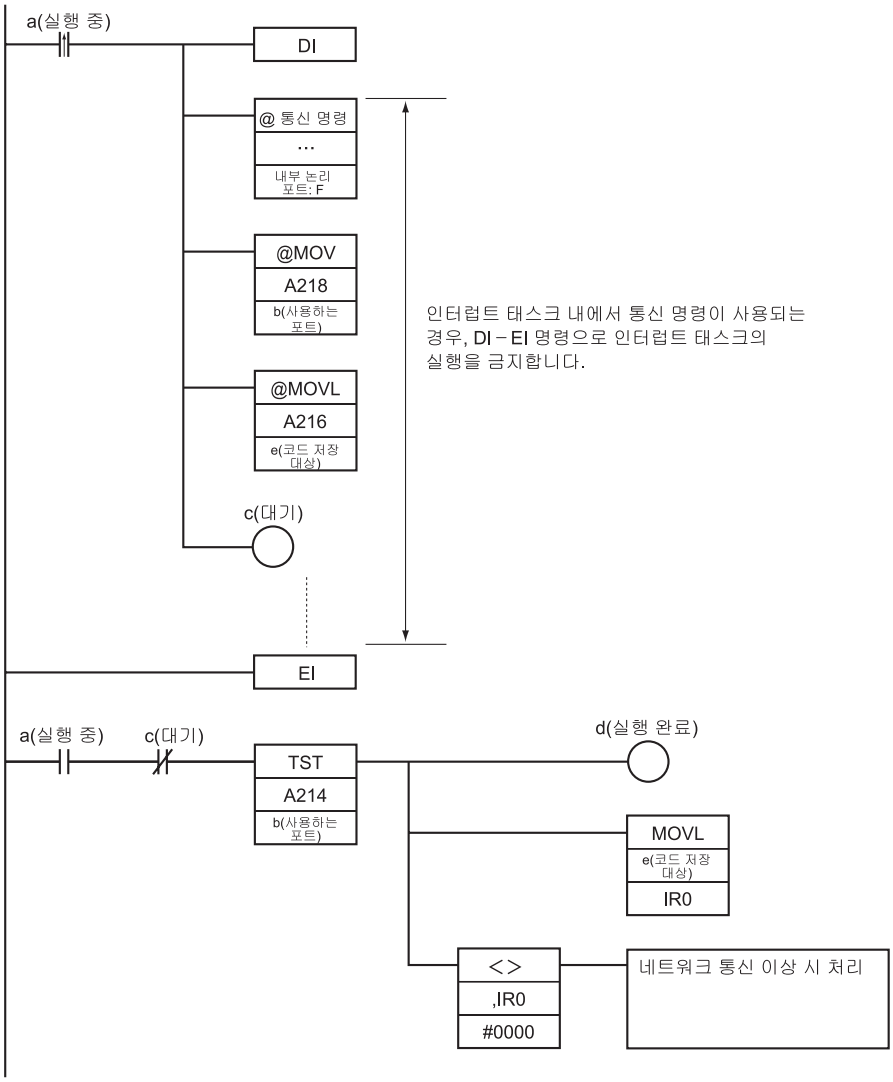
■ 통신 명령 실행 후 네트워크 통신 응답 코드를 참조하는 경우

네트워크 통신 응답 코드는 주로 에러 발생 시의 요인을 판단하기 위해 사용하지만 다음과 같이 #0000인지 여부를 확인하여 정상적으로 종료했는지 확인하는 데 사용할 수도 있습니다.

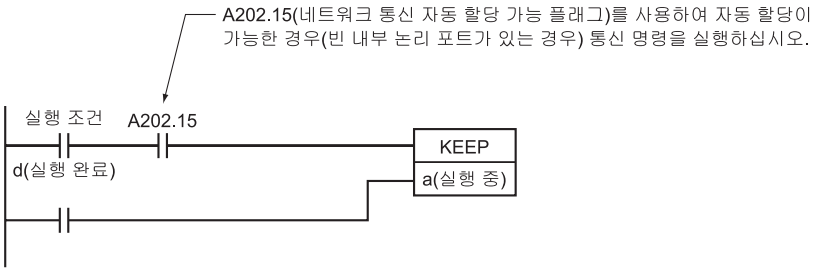


주 1: 인터럽트 태스크 내에서도 통신 명령을 사용하는 경우(내부 논리 포트 자동 할당 또는 내부 논리 포트 사용자 지정 중 어느 것이든), 사이클릭 태스크 내의 내부 논리 포트 자동 할당을 사용한 통신 명령 실행 전후에 다음과 같이 DI 명령과 EI 명령을 사용하여 인터럽트 태스크의 실행을 금지하십시오.

사이클릭 태스크

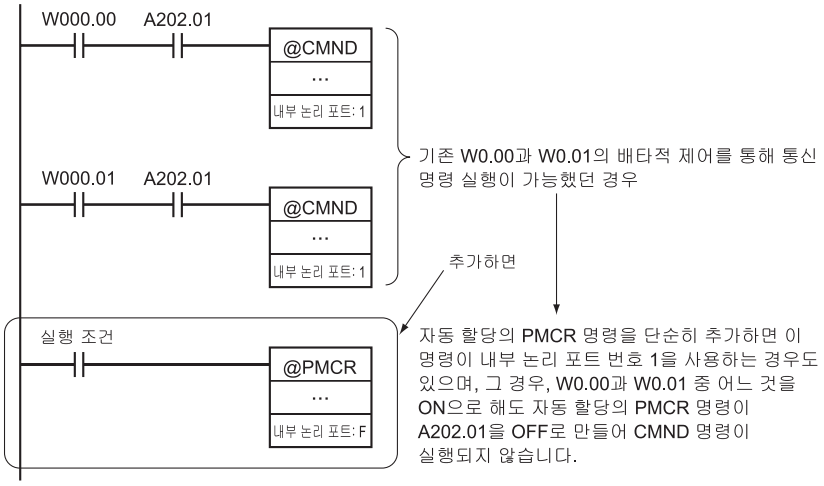


주 2: 동시에 9개 이상의 통신 명령을 실행할 가능성이 있는 경우는 자동 할당을 수행할 때도 실행하기 전에 다음과 같이 빈 내부 논리 포트가 있는지 확인하는 래더 프로그램을 구성하십시오.



주 3: 내부 논리 포트 사용자 지정과 내부 논리 포트 자동 할당을 통신 명령을 통해 혼재시킬 수 있습니다. 단, 각 통신 명령의 실행 조건에 따라 내부 논리 포트를 자동 할당한 통신 명령이 사용자 지정 내부 논리 포트를 사용할 가능성이 있습니다. 따라서 특히 다음과 같이 기존 래더 프로그램에 자동 할당 통신 명령을 추가할 때는 주의해야 합니다.

예)



CJ2 CPU 유니트 전용 네트워크용 통신 명령(SEND2, CMND2, PMCR2, RECV2) 공통 사양

■ 논리 포트 할당

내부 논리 포트(*1)는 기존의 네트워크 통신 명령의 경우 명령을 실행할 때 피연산자를 통해 지정했지만, 이 명령에서는 64개까지 동시에 사용할 수 있으며 포트 번호를 자동 할당하는 새로운 내부 논리 포트(8~71)를 사용할 수 있게 되었습니다. 따라서 내부 논리 포트의 번호를 고려해야 할 필요가 없습니다.

명령 실행 시 내부 논리 포트가 할당될 때마다 내부 논리 포트 사용 가능 수(A211)를 감소시키고, 해당하는 논리 포트의 응답 반환을 통해 증가시켜 통신 성능을 감시할 수 있습니다.

내부 논리 포트 번호	용도	취득 방법
0－7	네트워크 통신 명령, 시리얼 통신 명령, 간이 백업 (SEND, CMND, PMCR, TXDU 등)	피연산자로 지정 0~7: 지정된 번호의 논리 포트에 할당 F: 0~7 사이에서 자동 할당
8－71	새로운 네트워크 통신 명령(SEND2, CMND2, PMCR2, RECV2)	내부적으로 자동 할당(실행 가능한 수를 A211에 출력)

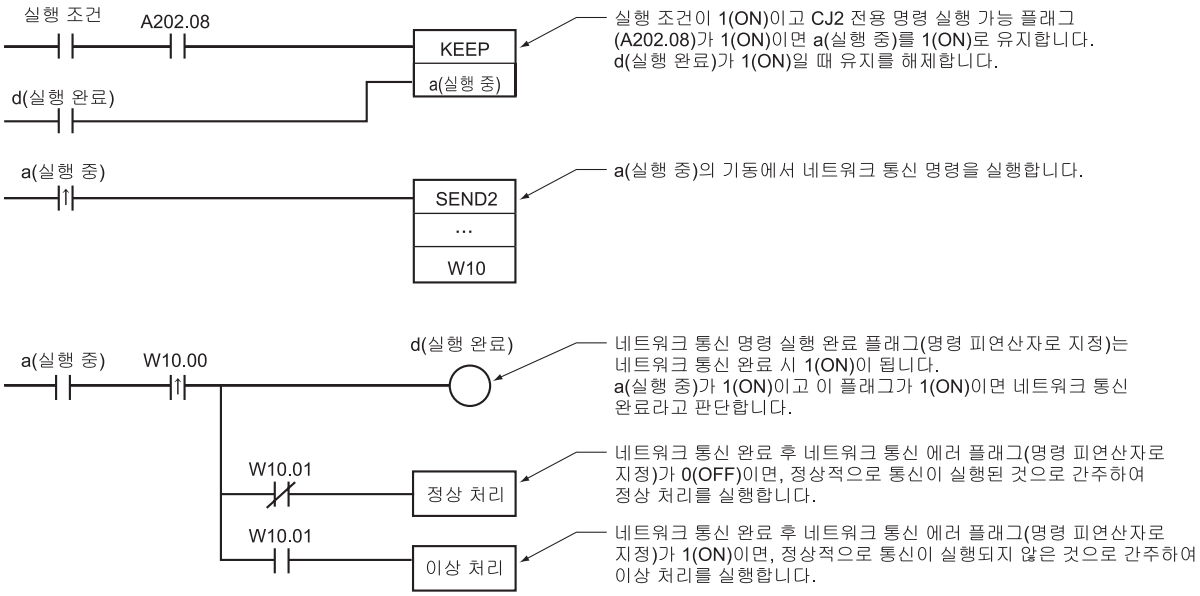
*1: 여러 사이클에 걸친 처리(통신 처리, 파일계 처리 등)의 동시 실행을 관리하는 가상의 내부 논리 포트를 말합니다.

■ 특수 보조 릴레이

기존 명령의 통신 실행 가능 플래그(A202)는 신규 명령에서 명령 피연산자의 통신 정보 D2에 해당합니다.

특수 보조 릴레이	CJ1/CS1/CJ2 시리즈 공통 통신 명령에서의 용도	CJ2 시리즈 전용 명령에서 대응하는 CH, 비트
A202.00~A202.07	통신 명령 실행 가능 플래그(내부 논리 포트 0~7)	명령 피연산자 D2 비트 0 (실행 가능이 아니라 실행 완료에서 ON)
A202.8	사용 안 함	CJ2 전용 명령 실행 가능 플래그(초기값=1)
A203~210	내부 논리 포트 0~7에 대한 통신 응답 코드	명령 피연산자 D2＋1
A211	사용 안 함	사용 가능한 CJ2 전용 명령 수(＝0이 되면 A202.08을 OFF)
A213	Explicit 통신 실행 에러 플래그	
A214.00~A214.07	네트워크 통신 완료 시 1사이클 ON 플래그	—
A215.00~A215.07	네트워크 통신 완료 시 1사이클 ON 에러 플래그	—
A216-217	네트워크 통신 응답 코드 주소	—
A218	사용 중인 네트워크 내부 논리 포트 번호	—
A219	통신 실행 중 에러 플래그	명령 피연산자 D2 비트 1

사용 방법



Explicit 메시지 통신 명령

■ Explicit 메시지 통신 방법

PLC에서 Explicit 메시지를 전송하는 경우, 다음 2가지 방법이 있습니다.

- CMND/CMND2 명령을 사용하여 FINS 명령 #2801(Explicit 메시지 송신 명령)을 전송하는 방법
- 다음과 같은 전용 Explicit 메시지 통신 명령을 사용하는 방법(주)

주: Explicit 메시지 통신 명령은 CS/CJ 시리즈 유니트 Ver.2.0 이상에서만 사용 가능합니다.

■ Explicit 메시지 통신 명령 목록

아래의 Explicit 전용 명령을 Explicit 메시지 통신 명령이라고 합니다.

니모닉(펑션 번호)	명령 이름	개요
EXPLT(720)	범용 Explicit 메시지 송신 명령	임의의 Service 코드의 Explicit 메시지를 송신합니다. 주: 기능적으로는 CMND/CMND2 명령을 사용하여 FINS 명령 #2801을 전송하는 방법과 동등합니다.
EGATR(721)	Explicit 읽기 명령	Service 코드가 #0E(각종 정보, 상태의 읽기: Get Attribute single)의 Explicit 메시지를 송신합니다.
ESATR(722)	Explicit 쓰기 명령	Service 코드가 #10(각종 정보의 쓰기: Set Attribute single)의 Explicit 메시지를 송신합니다.
ECHRD(723)	Explicit CPU 유니트 데이터 읽기 명령	Explicit 메시지를 통해 네트워크의 CPU 유니트의 데이터를 읽습니다.
ECHWR(724)	Explicit CPU 유니트 데이터 쓰기 명령	Explicit 메시지를 통해 네트워크의 CPU 유니트의 데이터를 씁니다.

■ Explicit 메시지 통신 명령의 특징

- Explicit 메시지 통신에서도 간단하게 CPU 유니트 간의 데이터 읽기/쓰기 처리를 수행할 수 있습니다.
- 특히, Explicit 읽기 명령/쓰기 명령의 경우 Service 코드를 입력하지 않고 ClassID 이하를 기입하면 됩니다.
- 또한 Explicit 읽기 명령/쓰기 명령의 경우, 피연산자에 자체 및 다른 CPU 유니트의 I/O 메모리 영역을 직접 지정할 수도 있습니다.

  Service 코드 16진수 #1E(CH 단위의 데이터 읽기), 16진수 #1F(CH 단위의 데이터 쓰기)에서의 CMND/CMND2 명령 기술과 같이 영역 종류를 코드로 지정하고 채널 번호를 16진수로 지정할 필요가 없습니다.

■ 동작

'Explicit 통신 실행 에러 플래그'에서 정상 종료(OFF)인지, 이상 종료(ON)인지 판별합니다.

이상 종료(ON)인 경우, 그리고 FINS의 '네트워크 통신 실행 에러 플래그'를 통해 Explicit 메시지 자체가 전송되지 않았는지(ON 시), 아니면 전송되었는데 송신한 Explicit 메시지에 에러가 있었는지(OFF 시) 판별합니다.

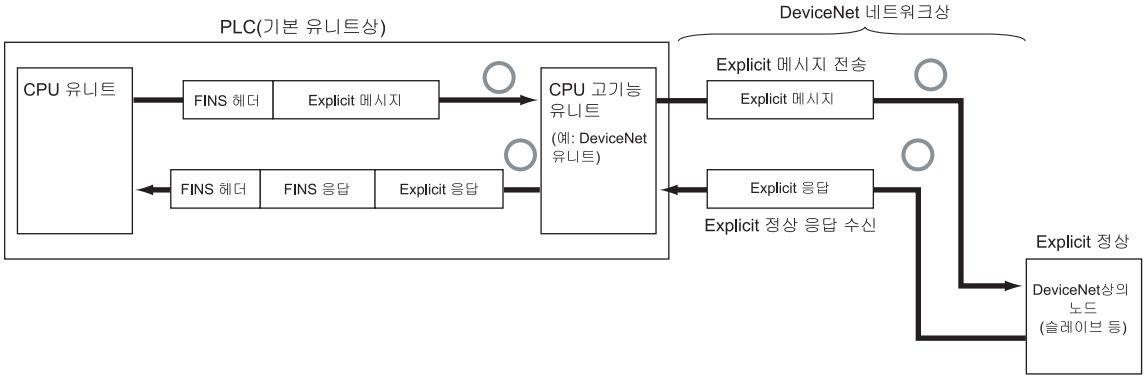
또한 '네트워크 통신 응답 코드'에는 정상 종료인 경우 #0000, Explicit의 에러인 경우 Explicit 메시지의 에러 코드(Error Code), FINS의 에러인 경우 FINS 메시지의 종료 코드가 저장됩니다.

조건		Explicit 통신 실행 에러 플래그(A213CH 비트 00~07: 내부 논리 포트 번호 0~7)	네트워크 통신 실행 에러 플래그(A219CH 비트 00~07: 내부 논리 포트 번호 0~7)	네트워크 통신 응답 코드 (A203~210CH: 내부 논리 포트 번호 0~7)
1) 정상 종료		0(OFF)	0(OFF)	#0000
2) 이상 종료	a) Explicit 메시지 자체를 전송하지 않았을 때	1(ON)	1(ON)	FINS 메시지의 종료 코드
	b) Explicit 메시지가 전송되었지만 Explicit 에러 응답이 돌아왔을 때		0(OFF)	Explicit 메시지의 에러 코드(Error Code)

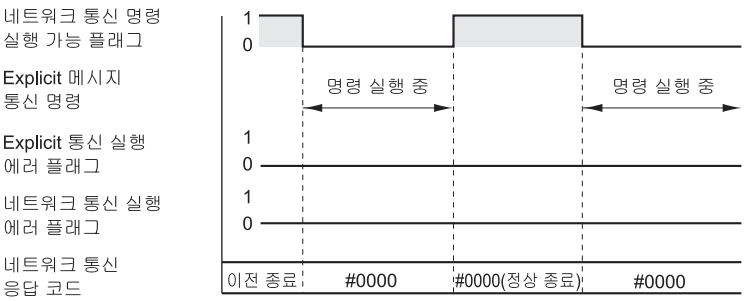
■ 정상 종료 시:

Explicit 메시지가 전송되고 정상 응답이 돌아옵니다.

Explicit 통신 실행 에러 플래그(A213CH 비트 00~07: 내부 논리 포트 번호 0~7)가 0(OFF)이 되고, 네트워크 통신 응답 코드(A203~210CH: 내부 논리 포트 번호 0~7)에 Explicit 메시지의 정상 응답 코드 #0000이 저장됩니다.



통신 플래그의 동작



■ 이상 종료 시:

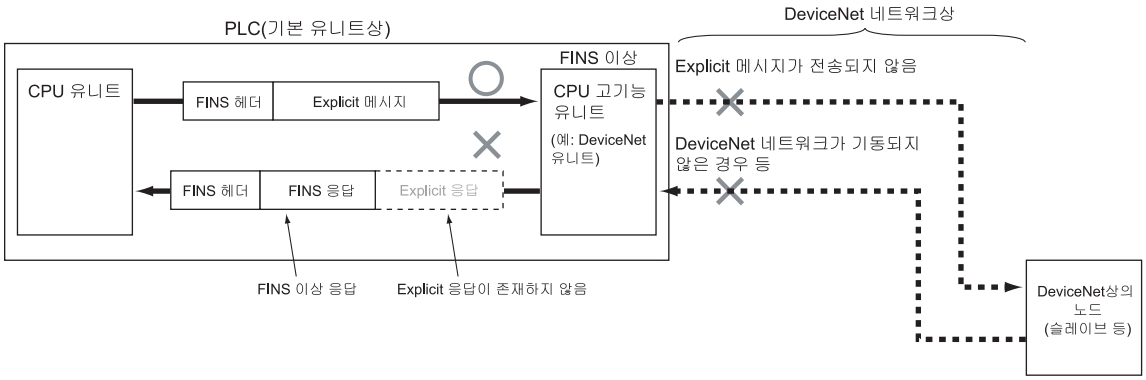
이 경우는 다음 a), b)의 2가지 경우가 있습니다.

a) Explicit 메시지 자체가 전송되지 않은 경우

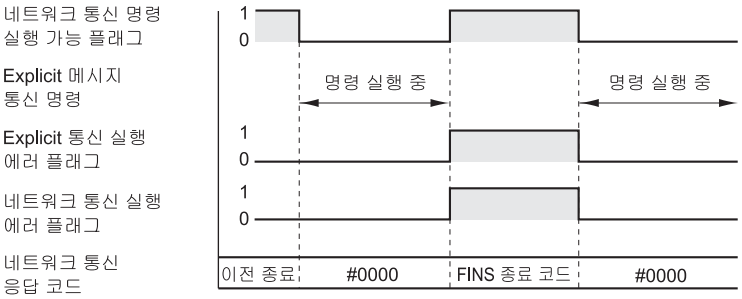
네트워크 자체가 기동하지 않은 경우 등에는 네트워크에 Explicit 메시지 자체가 송신되지 않습니다.

이 경우, Explicit 통신 실행 에러 플래그(A213CH 비트 00~07: 내부 논리 포트 번호 0~7)가 1(ON)이 되고 네트워크 통신 실행 에러 플래그(A219CH 비트 00~07: 내부 논리 포트 번호 0~7)는 1(ON)이 됩니다.

이때, 네트워크 통신 응답 코드(A203~210CH: 내부 논리 포트 번호 0~7)에는 FINS 메시지의 에러 코드가 저장됩니다.



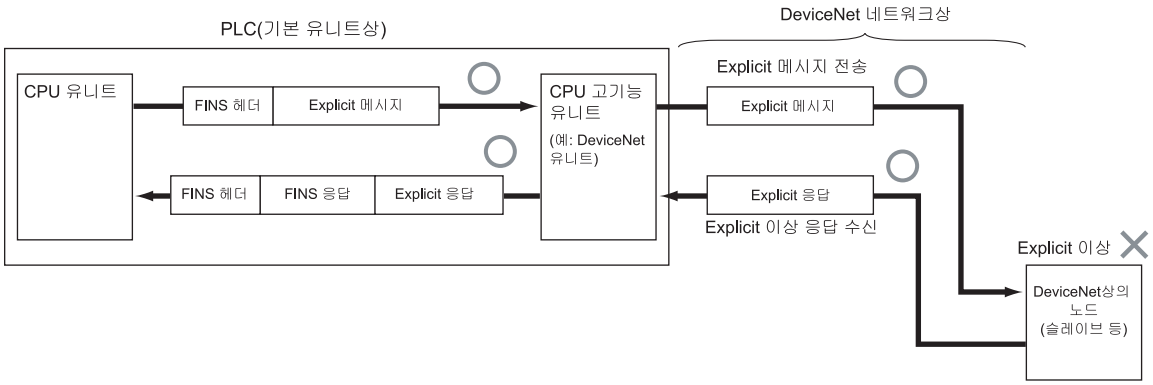
통신 플래그의 동작



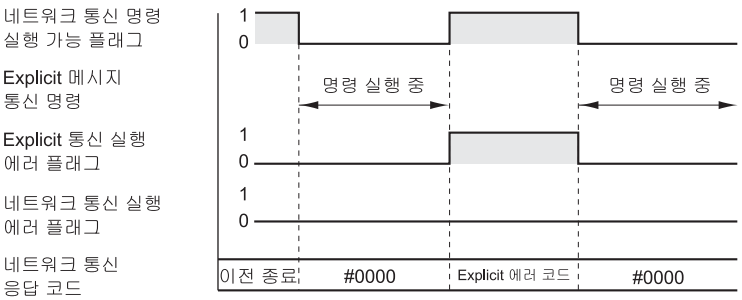
b) Explicit 메시지가 전송되었지만 에러 응답이 돌아온 경우

Explicit 메시지가 전송되었지만 Explicit 명령 프레임 자체에 에러(지원하지 않는 코드, 잘못된 크기 등)가 있는 경우, Explicit 통신 실행 에러 플래그(A213CH 비트 00~07: 내부 논리 포트 번호 0~7)가 1(ON)이 되고 네트워크 통신 실행 에러 플래그(A219CH 비트 00~07: 내부 논리 포트 번호 0~7)는 0(OFF)이 됩니다.

이때, 네트워크 통신 응답 코드(A203~210CH: 내부 논리 포트 번호 0~7)에는 Explicit 메시지의 에러 코드(Error Code)가 저장됩니다.

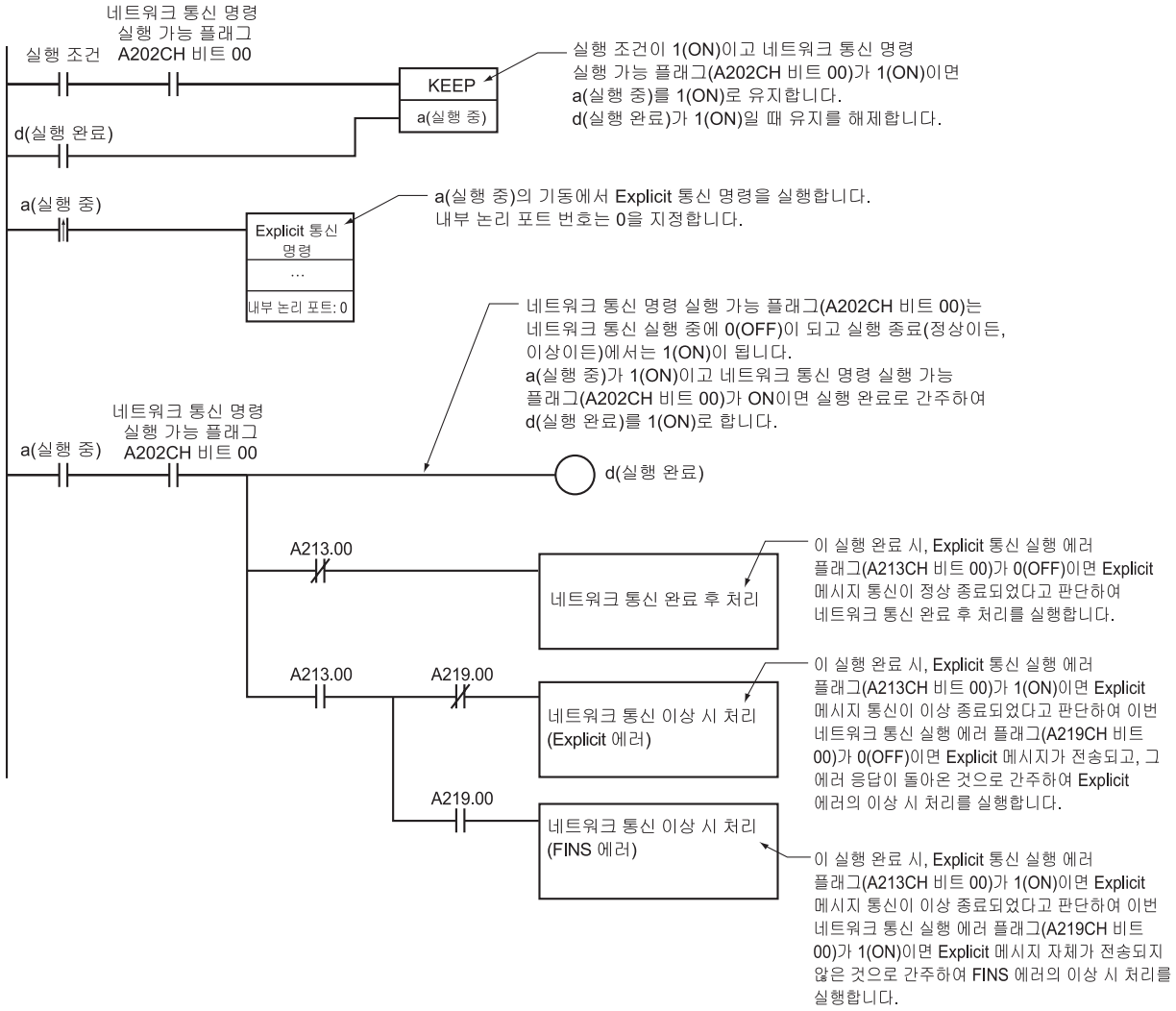


통신 플래그의 동작

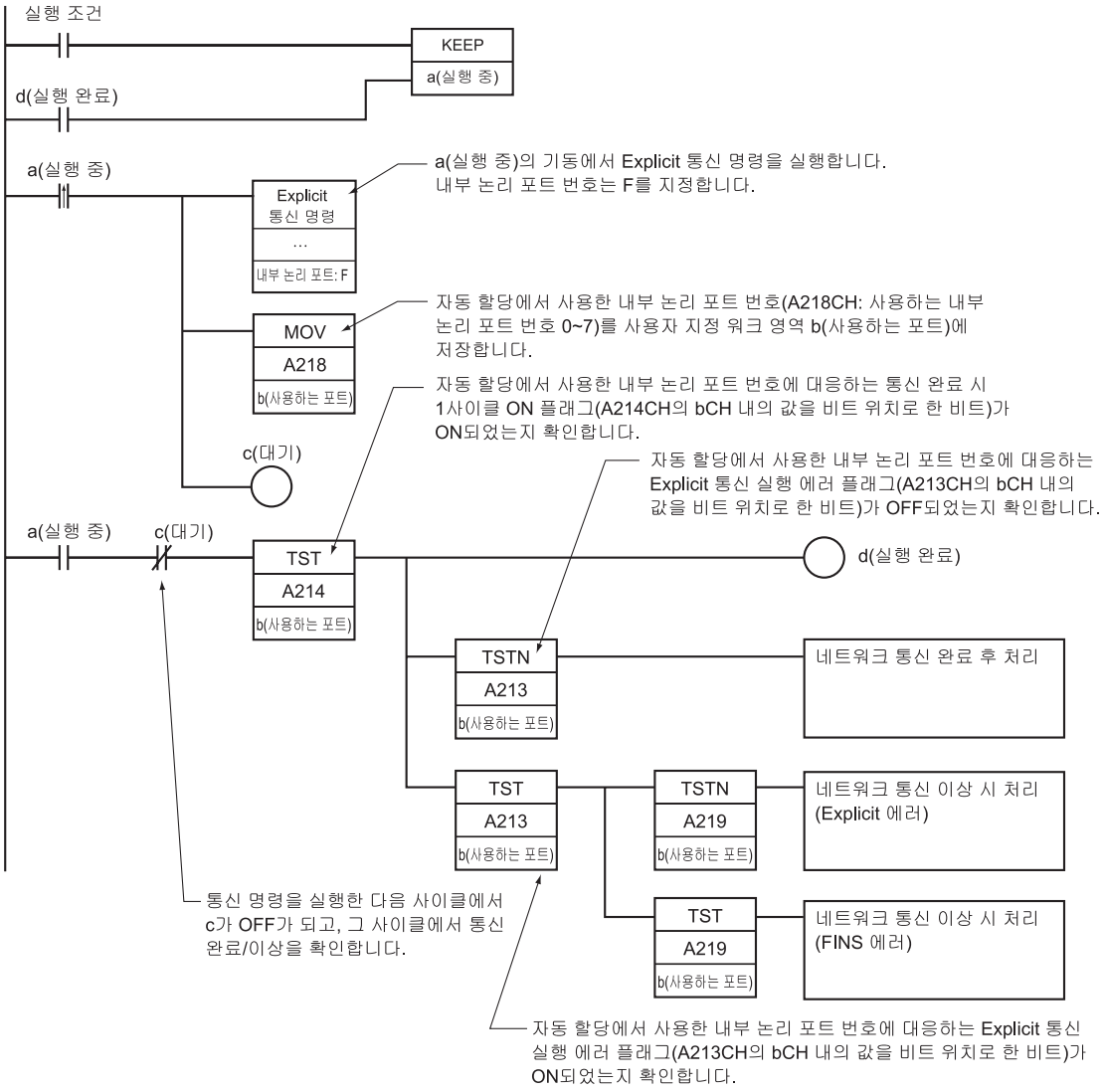


래더 프로그램 예

■ 내부 논리 포트를 사용자가 지정한 경우

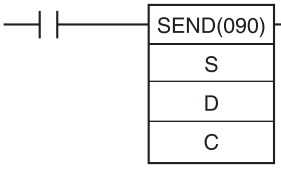


■ 내부 논리 포트를 자동 할당으로 한 경우



SEND

명령 이름	니모닉	동작 옵션	평션 번호	기능
네트워크 송신	SEND	@SEND	090	네트워크상의 노드에 데이터를 송신합니다.

심볼	SEND
	 S: 송신 원본(로컬 노드)의 송신 시작 CH 번호 D: 송신 대상(상대 노드)의 수신 시작 CH 번호 C: 제어 데이터 하위 CH 번호

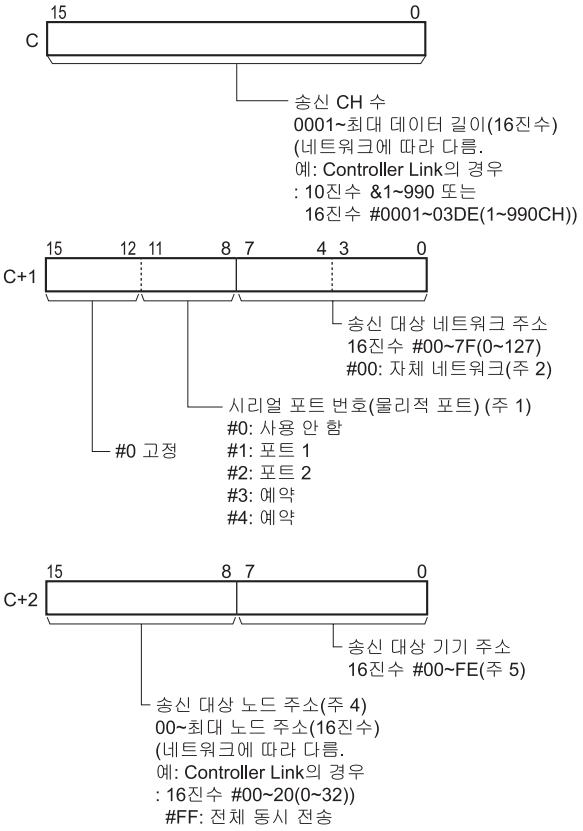
특정 영역 내의 사용 여부

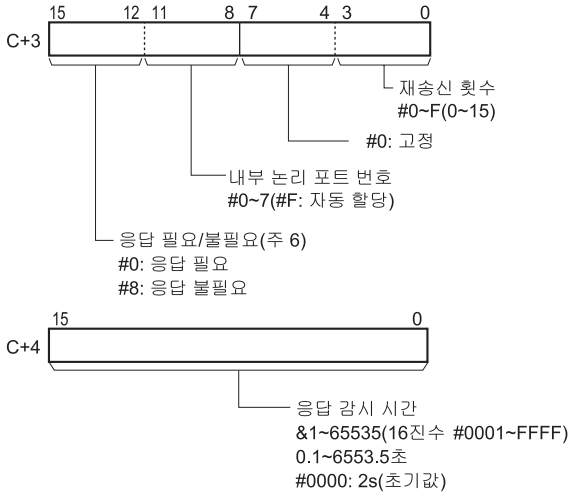
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	송신 원본(로컬 노드)의 송신 시작 CH 번호	UINT	가변
D	송신 대상(상대 노드)의 수신 시작 CH 번호	UINT	가변
C	제어 데이터 하위 CH 번호	WORD	5

C: 제어 데이터 하위 CH 번호





주 1: 상위 링크의 송신 기능을 통해 시리얼 포트를 경유하는 상위 컴퓨터를 대상으로 데이터를 송신하는 경우, 다음 두 가지 방법 중 하나를 사용합니다.

1) C+2의 비트 00~07의 송신 대상 기기 주소에 유니트/보드(CPU 유니트, 시리얼 커뮤니케이션 유니트/보드)의 기기 주소를 지정하여 C+1의 비트 8~11(시리얼 포트 번호)에 시리얼 통신의 포트 번호(포트 번호 1: #1, 포트 번호 2: #2)를 설정하는 방법

기기 주소 C+2 비트 00~07	유니트(장치)	시리얼 포트 번호 C+1 비트 08~11	시리얼 포트
16진수 #00	CPU 유니트	#1	내장 RS-232C 포트 사용
		#2	페리페럴 포트 사용
16진수 #10+유니트 번호(16진수)	시리얼 커뮤니케이션 유니트(CPU 고기능 유니트)	#1	포트 1 사용
		#2	포트 2 사용
16진수 #E1	시리얼 커뮤니케이션 보드(INNER 보드) (CS 시리즈만 해당)	#1	포트 1 사용
		#2	포트 2 사용

2) C+2의 비트 00~07의 송신 대상 기기 주소에 다음과 같은 시리얼 포트의 기기 주소를 직접 설정하는 방법(이 경우, C+1의 비트 08~11의 시리얼 포트 번호(물리적 포트)는 0: 사용 안 함으로 설정)

・ 시리얼 커뮤니케이션 유니트의 경우

시리얼 커뮤니케이션 유니트의 시리얼 포트	시리얼 포트의 기기 주소	예) 유니트 번호 1의 경우
포트 1	16진수 #80+16진수 #04×유니트 번호(16진수)	#80+#04×1=#84(10진수로 132)
포트 2	16진수 #81+16진수 #04×유니트 번호(16진수)	#81+#04×1=#85(10진수로 133)

・ 시리얼 커뮤니케이션 보드의 경우

시리얼 커뮤니케이션 보드의 시리얼 포트	시리얼 포트의 기기 주소
포트 1	16진수 #E4(10진수로 228)
포트 2	16진수 #E5(10진수로 229)

・ CPU 유니트의 경우

CPU 유니트의 시리얼 포트	시리얼 포트의 기기 주소
페리페럴 포트	16진수 #FD(10진수로 253)
RS-232C 포트	16진수 #FC(10진수로 252)

주 2: 시리얼 게이트웨이 기능(상위 링크 FINS로 변환)으로 라우팅 테이블 없이 시리얼 포트를 지정하는 경우는 송신 대상 네트워크 주소에 시리얼 포트의 기기 주소를 설정합니다.

주 3: 여러 대의 CPU 고기능 유니트가 장착된 경우, 유니트 번호가 가장 작은 유니트의 네트워크 주소가 됩니다.

주 4: 전체 동시 송신의 경우, 16진수 #FF로 설정합니다.
로컬 노드 내부로 송신하는 경우에는 16진수 #00으로 설정합니다.

주 5: 기기 주소:

- CPU 유니트: 16진수 #00
- CPU 고기능 유니트: 16진수 #10+유니트 번호(16진수)
- 고기능 I/O 유니트: 16진수 #20+기기 번호(16진수) (C200H 시리즈 고기능 I/O 유니트는 제외)
- INNER 보드: 16진수 #E1(CS 시리즈만 해당)

- 컴퓨터: 16진수 #01
- 네트워크에 연결된 유니트(상대 유니트의 유니트 번호를 설정할 필요 없음): 16진수 #FE

시리얼 포트의 기기 주소:

- 시리얼 커뮤니케이션 유니트의 경우
  포트 1: 16진수 #80 + 16진수 #04×유니트 번호(16진수)
  포트 2: 16진수 #81 + 16진수 #04×유니트 번호(16진수)
- 시리얼 커뮤니케이션 보드의 경우
  포트 1: 16진수 #E4(10진수로 228)
  포트 2: 16진수 #E5(10진수로 229)
- CPU 유니트의 경우
  페리페럴 포트: 16진수 #FD(10진수로 253)
  RS-232C 포트: 16진수 #FC(10진수로 252)

주 6: 전체 동시 송신하는 경우는 응답이 필요한 것으로 설정해도 응답이 돌아오지 않습니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	* ○	○	○	–	–	–	○	–	–	–	–
D																		
C																		

* CJ2 CPU 유니트만 EM D 뱅크를 지정할 수 있습니다. CS1/CJ1은 상대 노드가 CJ2 CPU 유니트일 때도 D 뱅크 이후는 지정할 수 없습니다.

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C로 지정한 시리얼 포트 번호가 0~4가 아닐 때 ON · C로 지정한 내부 논리 포트 번호에 대응하는 네트워크 명령 실행 가능 플래그가 OFF일 때 ON · 인터럽트 태스크에서 실행했을 때 ON(CJ2 CPU 유니트에서 인터럽트 고속 기동 기능 사용 시) · 그 이외일 때 OFF

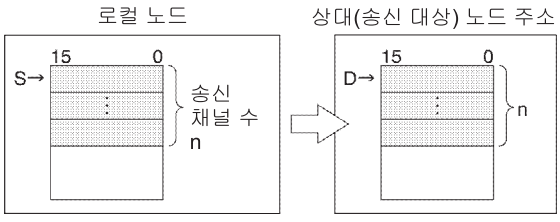
관련 특수 보조 릴레이

이름	주소	내용
네트워크 통신 명령 실행 가능 플래그	A202.00~ A202.07	네트워크 통신(PMCR 명령 포함)이 실행 가능할 때 1(ON)이 됩니다. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 네트워크 통신 실행 중에 0(OFF), 실행 종료(정상이든 이상이든) 후에는 1(ON)이 됩니다.
네트워크 통신 실행 에러 플래그	A219.00~ A219.07	네트워크 통신 실행 시 에러(이상)가 발생한 경우 1(ON)이 됩니다. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 다음 네트워크 통신 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 통신 명령 실행에서 0(OFF)이 됩니다.
네트워크 통신 응답 코드	A203CH~ A210CH	네트워크 통신이 실행된 경우 응답 코드(종료 코드)가 저장됩니다. 각 CH는 내부 논리 포트 번호에 대응합니다. A203~210CH: 내부 논리 포트 번호 00~07 통신 명령 실행 중에는 #0000이 되고 통신 명령 실행 종료 시에 반영됩니다. 명령을 실행하면 클리어됩니다.

주: 네트워크 통신 응답 코드에 대해서는 'CS/CJ 시리즈 통신 명령 참조 설명서'(번호 SBCA-304)의 FINS 명령 종료 코드 목록을 참조하십시오.

기능

· S로 지정한 송신 원본(로컬 노드)의 송신 시작 CH 번호부터 송신 CH 수 분량의 데이터를 CPU 버스 또는 네트워크를 통해 송신하고, C로 지정한 송신 대상 네트워크 주소의 송신 대상 노드 주소를 통해 지정된 송신 대상 기기 주소(장치)의 D로 지정한 수신 시작 CH부터 씁니다.

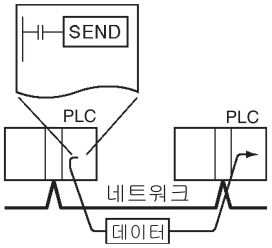


- C로 지정한 내부 논리 포트 번호가 실행 가능(네트워크 통신 명령 실행 가능 플래그가 ON)한 경우, 지정된 내부 논리 포트 번호의 네트워크 통신 실행 에러 플래그(내부 논리 포트 번호 00~07: A219.00~A219.07)와 네트워크 통신 실행 가능 플래그(내부 논리 포트 번호 00~07: A202.00~A202.07)를 OFF하고 네트워크 통신 응답 코드에 #0000을 씁니다. 그 후, 네트워크상의 노드에 데이터를 송신합니다.
- D에 EM 영역(현재 뱅크)을 지정한 경우, 송신 대상(상대 노드) EM 영역의 현재 뱅크에 기입됩니다.
- 자체 네트워크 이외로 데이터를 송신(네트워크 이탈)하는 경우, 라우팅 테이블(통신 상대의 PLC가 연결된 네트워크까지의 통신 경로를 등록한 테이블)을 각 네트워크의 PLC(CPU 유니트)에 등록해야 합니다.
- 송신 대상 노드 주소가 #FF일 때는 지정된 네트워크 주소의 모든 노드에 송신합니다(전체 동시 전송).
- 응답이 필요한 것으로 설정한 경우는 응답 감시 시간 이내에 응답이 없으면 응답이 있을 때까지 지정된 재송신 횟수만큼 데이터를 재송신합니다.
- 전체 동시 전송을 지정한 경우는 응답이 없으며 재송신도 하지 않습니다.
- 장치뿐만 아니라 장치의 시리얼 포트에 대해서도 이 명령을 실행할 수 있습니다.

Controller Link 네트워크나 Ethernet상의 PLC, PC 또는 상위 링크 모드로 시리얼 포트에 연결된 상위 컴퓨터에 대해 데이터를 송신할 수 있습니다.

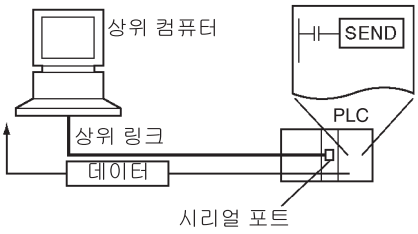
■ 네트워크상의 PLC(CPU 유니트) 또는 PC로 데이터 송신

Controller Link 네트워크나 Ethernet상의 PLC(CPU 유니트) 또는 PC에 대해 이 SEND 명령을 실행한 경우, 지정된 I/O 메모리 영역의 데이터를 상대방의 지정된 I/O 메모리 영역으로 송신할 수 있습니다.



■ 상위 링크로 연결된 상위 컴퓨터로 데이터 송신(송신 기능)

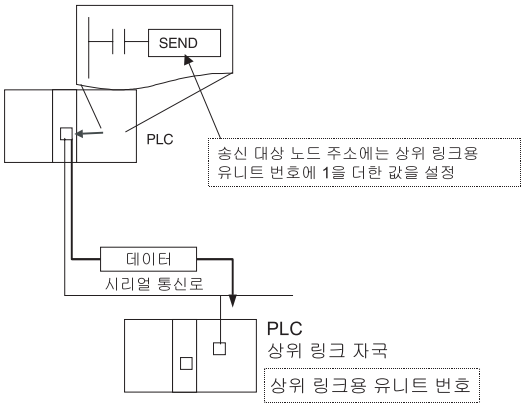
CPU 유니트 내장 시리얼 포트, 시리얼 커뮤니케이션 보드(CS 시리즈만 해당), 시리얼 커뮤니케이션 유니트 중 하나가 상위 컴퓨터(PC 등)와 상위 링크 모드로 1 : 1 연결되어 있을 때, 해당 유니트 중 하나의 시리얼 포트에 대해 SEND 명령을 실행한 경우, PLC 측이 최초의 송신권을 가지며 상위 컴퓨터로 CPU 유니트의 지정된 I/O 메모리 영역의 데이터를 송신할 수 있습니다. (자체 PLC 이외에 네트워크상의 PLC에 연결된 상위 컴퓨터에 송신하는 것도 가능합니다.)



■ 시리얼 포트(시리얼 게이트웨이 기능)에 연결된 상위 링크 자국의 PLC로 데이터 송신

시리얼 커뮤니케이션 보드/유니트의 시리얼 통신로에 상위 링크 자국으로 연결된 PLC로 시리얼 게이트웨이 기능을 통해 데이터를 송신하는 것도 가능합니다.

단, 이 경우, 송신 대상 노드 주소에는 상위 링크용 유니트 번호에 1을 더한 값을 설정한다는 점에 주의해야 합니다.



SEND

힌트

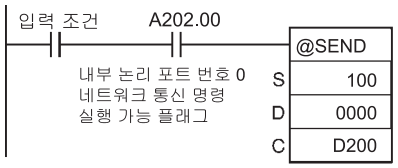
■ 상위 링크로 연결된 상위 컴퓨터로 데이터 송신(송신 기능)

SEND 명령을 CPU 유니트 내장 시리얼 포트, 시리얼 커뮤니케이션 보드(CS 시리즈만 해당), 시리얼 커뮤니케이션 유니트 중 한 유니트의 시리얼 포트에 대해 실행하면 그 중 한 유니트의 시리얼 포트에서 상위 컴퓨터에 대해 명령(FINS 메시지를 상위 링크의 헤더나 터미네이터에 포함시킨 명령)을 전송합니다. 전송하는 FINS 명령은 I/O 메모리 영역 쓰기 명령(명령 코드 "0102")이고 상위 링크 헤더 코드는 "OF"입니다.

상위 컴퓨터 측에서는 수신한 해당 명령(상위 링크에 포함시킨 FINS 명령)을 처리하는 프로그램을 작성해야 합니다.

자체 PLC상의 수신처인 경우, C+1의 송신 대상 네트워크 주소를 자체 네트워크(#00)에 설정하고, C+2의 송신 대상 노드 주소를 자체 PLC(#00)에 설정합니다. C+2의 송신 대상 기기 주소는 CPU 유니트인 경우 #00, 시리얼 커뮤니케이션 보드인 경우(CS 시리즈만 해당) #E1, 시리얼 커뮤니케이션 유니트인 경우 해당 유니트 번호+#10을 지정합니다.

· SEND 명령(상위 컴퓨터로 데이터 송신)의 경우:



입력 조건이 ON이고 내부 논리 포트 번호 0 네트워크 통신 명령 실행 가능 플래그가 ON일 때, 릴레이 영역의 100CH~109CH(10CH분)의 데이터를 네트워크 주소 0, 노드 주소 0, 기기 주소 16진수 #10의 시리얼 커뮤니케이션 유니트의 포트 1에 연결된 상위 컴퓨터에 전송합니다.

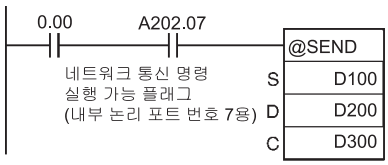
주: 상위 컴퓨터 측에서는 데이터를 수신하여 응답을 PLC로 전송하는 프로그램이 필요합니다.

C	D200	0 0 : 0 A	전송 채널 수 10CH
C+1	D201	0 1 : 0 0	시리얼 커뮤니케이션 유니트의 포트 1(비트 10~08), 송신 대상 네트워크 주소 0(비트 07~00)
C+2	D202	0 0 : 1 0	송신 대상 노드 주소 0(비트 15~08), 송신 대상 기기 주소 16진수 #10(비트 07~00)
C+3	D203	0 0 : 0 0	응답 필요(비트 15), 내부 논리 포트 0(비트 11~08), 재송신 횟수 0회(비트 04~00)
C+4	D204	0 0 : 0 0	응답 감시 시간 2초(기본값: #0000)

사용상의 주의 사항

- 1개의 내부 논리 포트에 대해 한번에 1개 명령만 실행할 수 있습니다. 따라서 각 포트에 대응된 네트워크 통신 명령 실행 가능 플래그가 0(OFF)인 경우는 해당 포트에 대해 이 명령을 실행하지 않도록 네트워크 통신 명령 실행 가능 플래그(A202.00~A202.07)를 a 접점의 입력 조건으로 하십시오.
- 내부 논리 포트 번호를 자동 할당("F")으로 사용하는 경우에 대해서는 3-736의 '내부 논리 포트 자동 할당 기능'의 설명을 참조하십시오.
- 내부 논리 포트 번호는 네트워크 통신 명령과 PMCR(프로토콜 매크로) 명령에서 공용입니다. 따라서 동일한 내부 논리 포트 번호를 사용하여 PMCR 명령과 네트워크 통신 명령을 동시에 실행할 수 없으므로 주의하십시오.
- 노이즈 등으로 인해 통신 도중에 송신 메시지나 응답이 소실되는 경우 등을 고려하여 SEND 명령을 실행할 때는 재송신 횟수를 0 이외로 설정하고, 응답이 감시 시간을 초과해도 돌아오지 않는 경우 재송신 처리를 수행할 것을 권장합니다.

샘플 프로그램

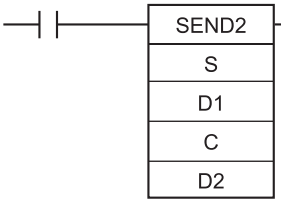


0.00 및 A202.07(내부 논리 포트 번호 7용 네트워크 통신 명령 실행 가능 플래그)이 ON일 때, D100~10CH(D100~D109)의 데이터를 자체 네트워크상에 있는 노드 주소 3의 CPU 유니트의 D200부터 10CH(D200~D209)에 송신합니다. 응답 감시 시간 10초 후 응답이 돌아오지 않은 경우는 재송신 처리를 최대 3회까지 실행합니다.

C: D300	0 0 : 0 A	송신 CH = 10CH
C+1: D301	0 0 : 0 0	자체 네트워크상의 장치 자체에 대해
C+2: D302	0 3 : 0 0	송신 대상 노드 주소 = #03, 기기 주소 = #00(CPU 유니트)
C+3: D303	0 7 : 0 3	응답 필요, 내부 논리 포트 7, 재송신 횟수 = #03
C+4: D304	0 0 : 6 4	응답 감시 시간 = #0064(10초)

SEND2

명령 이름	니모닉	동작 옵션	펑션 번호	기능
네트워크 송신 2	SEND2	@SEND2	491	네트워크상의 노드에 데이터를 송신합니다.

	SEND2	
심볼		
	S	S: 송신 요구 대상의 송신 시작 채널
	D1	D1: 송신 요구 원본의 수신 시작 채널
	C	C: 제어 데이터
	D2	D2: 통신 정보

특정 영역 내의 사용 여부

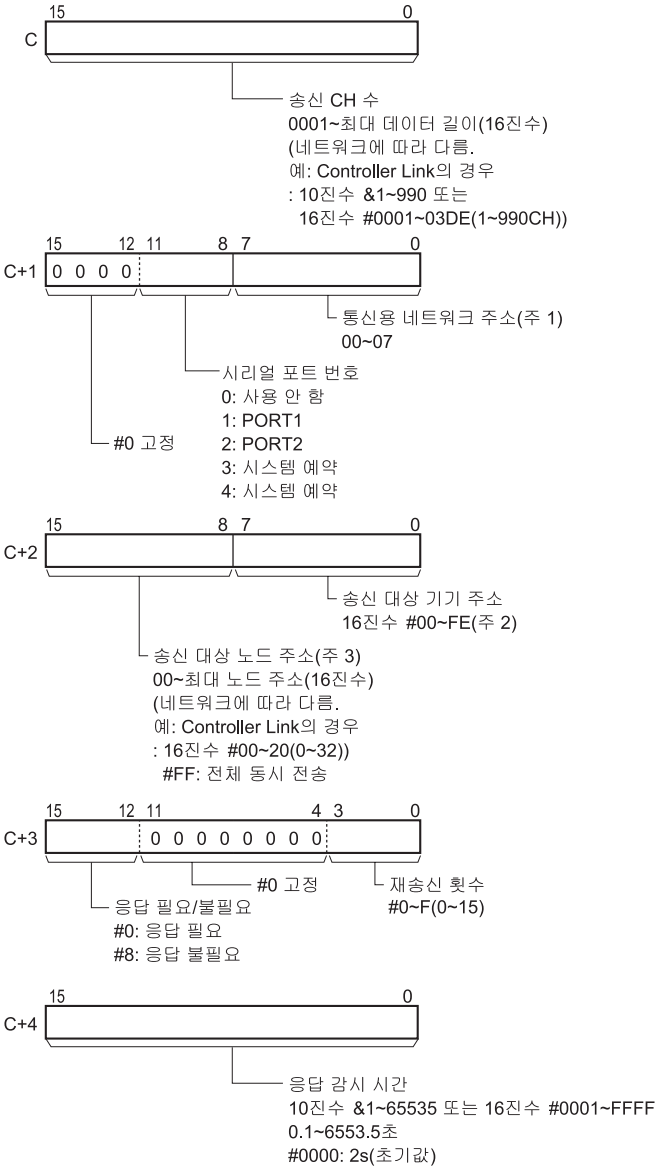
영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	송신 명령 시작 채널	WORD	가변
D1	응답 저장 대상의 시작 채널	WORD	가변
C	제어 데이터	WORD	6
D2	통신 정보	WORD	2

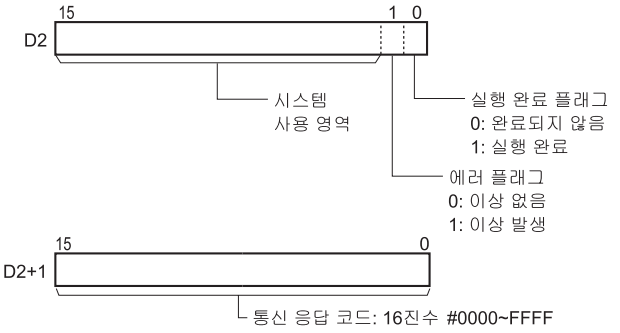
SEND2

C: 제어 데이터



D2: 통신 정보

D1, D2는 명령을 실행하기 전에 0으로 클리어 하십시오.



주 1: 통신용 네트워크 주소는 라우팅 테이블로 설정하십시오.
단, 자체 네트워크를 경유하여 명령을 실행하는 경우는 라우팅 테이블이 필요하지 않습니다. 통신용 네트워크 주소: 00으로 설정하십시오.
자체 네트워크란 여러 대의 CPU 고기능 유니트가 장착된 경우, 유니트 번호가 가장 작은 유니트가 연결된 네트워크를 말합니다.

주 2: 기기 주소:

- CPU 유니트: 16진수 #00
- CPU 고기능 유니트: 16진수 #10+유니트 번호(16진수)
- 고기능 I/O 유니트: 16진수 #20+기기 번호(16진수) (C200H 시리즈 고기능 I/O 유니트는 제외)
- INNER 보드: 16진수 #E1(CS 시리즈만 해당)
- 컴퓨터: 16진수 #01
- 네트워크에 연결된 유니트(상대 유니트의 유니트 번호를 특정할 필요 없음): 16진수 #FE

시리얼 포트의 기기 주소:

- 시리얼 커뮤니케이션 유니트의 경우
  포트 1: 16진수 #80+16진수 #04×유니트 번호(16진수)
  포트 2: 16진수 #81+16진수 #04×유니트 번호(16진수)
- 시리얼 커뮤니케이션 보드의 경우
  포트 1: 16진수 #E4(10진수로 228)
  포트 2: 16진수 #E5(10진수로 229)
- CPU 유니트의 경우
  페리페럴 포트: 16진수 #FD(10진수로 253)
  RS-232C 포트: 16진수 #FC(10진수로 252)

주 3: 전체 동시 송신의 경우, 16진수 #FF로 설정합니다.
로컬 노드 내부로 송신하는 경우에는 16진수 #00으로 설정합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
D1																		
C												–						
D2																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C로 지정한 시리얼 포트 번호 0~4 Hex가 아닐 때 ON · 빈 내부 논리 포트가 없을 때(A211=0) ON · 그 이외일 때 OFF

기능

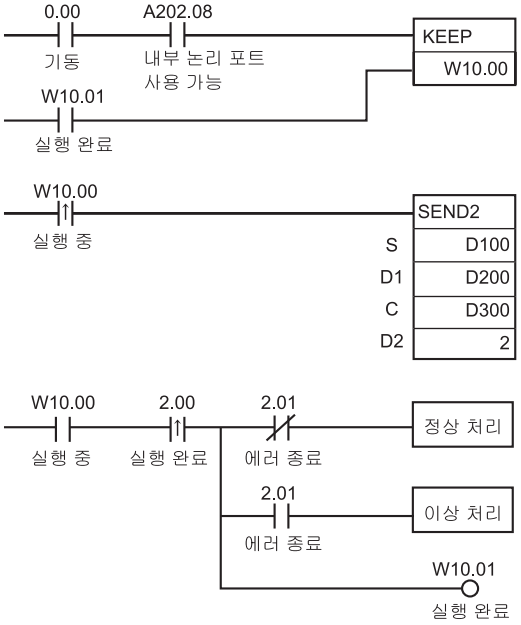
S로 지정한 송신 원본(로컬 노드)의 송신 시작 CH 번호부터 송신 CH 수 분량의 데이터를 송신하고, C로 지정한 송신 대상 네트워크 주소의 송신 대상 노드 주소를 통해 지정된 송신 대상 기기 주소(장치)의 D로 지정한 수신 시작 CH부터 씁니다.

송신을 시작하면 D2의 실행 완료 플래그를 OFF로 합니다. 응답이 돌아오면 송신 결과의 정상/이상이 D2의 에러 플래그에 반영됩니다. 통신 응답은 D2＋1에 기록됩니다.
기존 SEND 명령의 통신 가능 플래그(A202) 및 네트워크 통신 실행 에러 플래그(A219)가 피연산자 D2에 해당하고, 네트워크 통신 응답 코드(A203~210)가 D2+1에 해당합니다.

SEND2

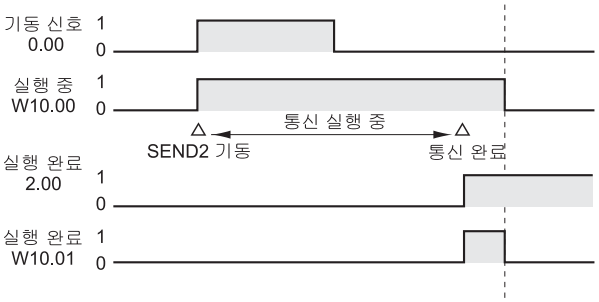
샘플 프로그램

0.00(기동) 및 A202.08(내부 논리 포트 사용 가능)이 ON이면 W10.00(실행 중)이 ON이 됩니다. W10.00(실행 중)이 ON이면 D100~10CH(D100~D109)의 데이터를 자체 네트워크상의 노드 주소 3의 CPU 유니트의 D200부터 10CH(D200~D209)로 송신합니다. 응답 감시 시간 10초 후 응답이 돌아오지 않은 경우는 재송신 처리를 최대 3회까지 실행합니다.
정상일 때는 2.00(실행 완료)이 ON되고 정상 처리를 실행합니다. 이상일 때는 2.01(에러 종료)이 ON되고 이상 처리를 실행한 후 종료합니다.



C: D300	0	0	1	0	송신 CH 수 = 10CH
C+1: D301	0	0	0	0	자체 네트워크상의 장치 자체에 대해
C+2: D302	0	3	0	0	송신 대상 노드 주소 = #03, 기기 주소 = #00(CPU 유니트)
C+3: D303	0	0	0	3	응답 필요, 재송신 횟수 = #03
C+4: D304	0	1	0	0	응답 감시 시간 = #0064(10초)

■ 타임 차트



RECV

명령 이름	니모닉	동작 옵션	펑션 번호	기능
네트워크 수신	RECV	@RECV	098	네트워크상의 노드에 대해 송신을 요구하고 데이터를 수신합니다.

심볼	RECV
	RECV(098) S: 송신 요구 대상(상대 노드)의 송신 시작 CH 번호 D: 송신 요구 원본(로컬 노드)의 수신 시작 CH 번호 C: 제어 데이터 하위 CH 번호

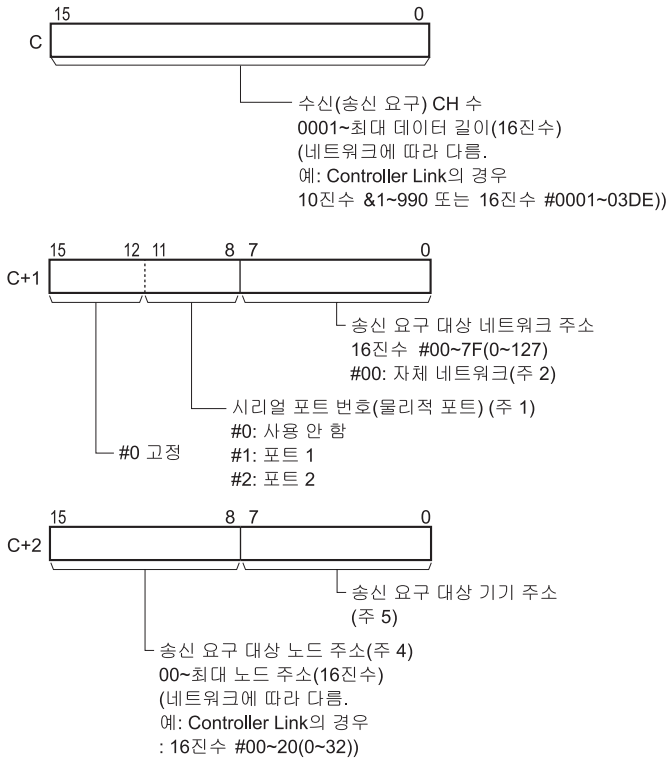
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

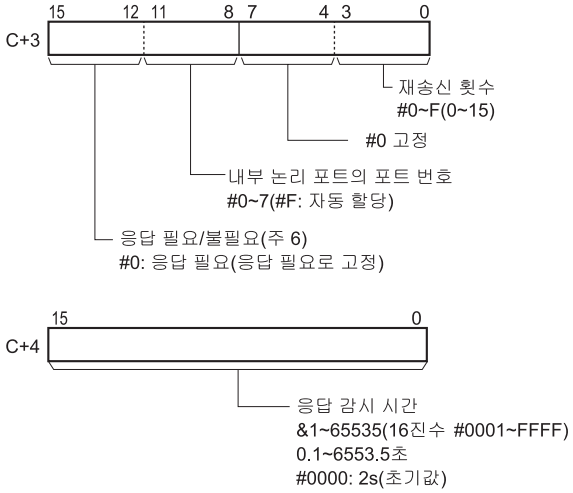
피연산자 설명

피연산자	내용	데이터 형식	크기
S	송신 요구 대상(상대 노드)의 송신 시작 CH 번호	UINT	가변
D	송신 요구 원본(로컬 노드)의 수신 시작 CH 번호	UINT	가변
C	제어 데이터 하위 CH 번호	WORD	5

C: 제어 데이터 하위 CH 번호



RECV



주 1: 상위 링크의 송신 기능을 통해 시리얼 포트를 경유하는 상위 컴퓨터에서 데이터를 수신하는 경우, 다음 두 가지 방법 중 하나를 사용합니다.

1) C+2의 비트 00~07의 송신 요구 대상 기기 주소에 유니트/보드(CPU 유니트, 시리얼 커뮤니케이션 유니트/보드)의 기기 주소를 지정하여 C+1의 비트 8~11(시리얼 포트 번호)에 시리얼 통신의 포트 번호(포트 번호 1: #1, 포트 번호 2: #2)를 설정하는 방법

기기 주소 C+2 비트 00~07	유니트(장치)	시리얼 포트 번호 C+1 비트 08~11	시리얼 포트
16진수 #00	CPU 유니트	#1	내장 RS-232C 포트 사용
		#2	페리페럴 포트 사용
16진수 #10 +유니트 번호(16진수)	시리얼 커뮤니케이션 유니트(CPU 고기능 유니트)	#1	포트 1 사용
		#2	포트 2 사용
16진수 #E1	시리얼 커뮤니케이션 보드(INNER 보드) (CS 시리즈만 해당)	#1	포트 1 사용
		#2	포트 2 사용

2) C+2의 비트 00~07의 송신 요구 대상 기기 주소에 다음과 같은 시리얼 포트의 기기 주소를 직접 설정하는 방법(이 경우, C+1의 비트 08~11의 시리얼 포트 번호(물리적 포트)는 0: 사용 안 함으로 설정)

・ 시리얼 커뮤니케이션 유니트의 경우

시리얼 커뮤니케이션 유니트의 시리얼 포트	시리얼 포트의 기기 주소	예) 유니트 번호 1의 경우
포트 1	16진수 #80 + 16진수 #04 × 유니트 번호(16진수)	#80 + #04 × 1 = #84 (10진수로 132)
포트 2	16진수 #81 + 16진수 #04 × 유니트 번호(16진수)	#81 + #04 × 1 = #85 (10진수로 133)

・ 시리얼 커뮤니케이션 보드의 경우

시리얼 커뮤니케이션 보드의 시리얼 포트	시리얼 포트의 기기 주소
포트 1	16진수 #E4(10진수로 228)
포트 2	16진수 #E5(10진수로 229)

・ CPU 유니트의 경우

CPU 유니트의 시리얼 포트	시리얼 포트의 기기 주소
페리페럴 포트	16진수 #FD(10진수로 253)
RS-232C 포트	16진수 #FC(10진수로 252)

주 2: 시리얼 게이트웨이 기능(상위 링크 FINS로 변환)으로 라우팅 테이블 없이 시리얼 포트를 지정하는 경우는 송신 요구 대상 네트워크 주소에 시리얼 포트의 기기 주소를 설정합니다.

주 3: 여러 대의 CPU 고기능 유니트가 장착된 경우, 유니트 번호가 가장 작은 유니트의 네트워크 주소가 됩니다.

주 4: RECV 명령에서는 전체 동시 송신이 불가능합니다. 로컬 노드 내부로 송신을 요구하는 경우 #00을 설정합니다.

주 5: 기기 주소

- CPU 유니트: 16진수 #00
- CPU 고기능 유니트: 16진수 #10 + 유니트 번호(16진수)
- 고기능 I/O 유니트: 16진수 #20+기기 번호(16진수)
  (C200H 시리즈 고기능 I/O 유니트 제외)

- INNER 보드: 16진수 #E1(CS 시리즈만 해당)
- 컴퓨터: 16진수 #01
- 네트워크에 연결된 유니트(상대 유니트의 유니트 번호를 특정할 필요 없음): 16진수 #FE

시리얼 포트의 기기 주소:

- 시리얼 커뮤니케이션 유니트의 경우
  포트 1: 16진수 #80 + 16진수 #04 × 유니트 번호(16진수)
  포트 2: 16진수 #81 + 16진수 #04 × 유니트 번호(16진수)
- 시리얼 커뮤니케이션 보드의 경우
  포트 1: 16진수 #E4(10진수로 228)
  포트 2: 16진수 #E5(10진수로 229)
- CPU 유니트의 경우
  페리페럴 포트: 16진수 #FD(10진수로 253)
  RS-232C 포트: 16진수 #FC(10진수로 252)

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	* ○	○	○	−	−	−	○	−	−	−	−
D																		
C																		

* CJ2 CPU 유니트만 EM D 뱅크를 지정할 수 있습니다. CS1/CJ1은 상대 노드가 CJ2 CPU 유니트일 때도 D 뱅크 이후는 지정할 수 없습니다.

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C로 지정한 시리얼 포트 번호가 0~4가 아닐 때 ON · C로 지정한 내부 논리 포트 번호에 대응하는 네트워크 명령 실행 가능 플래그가 OFF일 때 ON · 인터럽트 태스크에서 실행했을 때 ON(CJ2 CPU 유니트에서 인터럽트 고속 기동 기능 사용 시) · 그 이외일 때 OFF

관련 특수 보조 릴레이

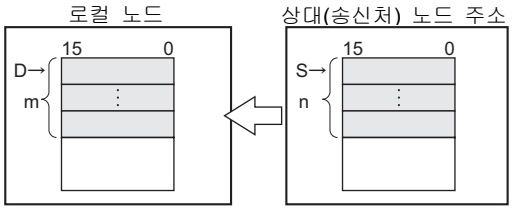
이름	주소	내용
네트워크 통신 명령 실행 가능 플래그	A202.00~ A202.07	네트워크 통신(PMCR 명령 포함)이 실행 가능할 때 1(ON)이 됩니다. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 네트워크 통신 실행 중에 0(OFF), 실행 종료(정상이든 이상이든) 후에는 1(ON)이 됩니다.
네트워크 통신 실행 에러 플래그	A219.00~ A219.07	네트워크 통신 실행 시 에러(이상)가 발생한 경우 1(ON)이 됩니다. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 다음 네트워크 통신 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 통신 명령 실행에서 0(OFF)이 됩니다.
네트워크 통신 응답 코드	A203CH~ A210CH	네트워크 통신이 실행된 경우 응답 코드(종료 코드)가 저장됩니다. 각 CH는 내부 논리 포트 번호에 대응합니다. A203~210CH: 내부 논리 포트 번호 00~07 통신 명령 실행 중에는 #0000이 되고 통신 명령 실행이 종료되면 반영됩니다. 운전 시작 시에 클리어됩니다.

주:
· 네트워크 통신 응답 코드에 대해서는 'CS/CJ 시리즈 통신 명령 참조 설명서'(번호 SBCA-304)의 FINS 명령 종료 코드 목록을 참조하십시오.

RECV

기능

C로 지정한 네트워크 주소의 송신 요구 대상(상대방) 노드 주소를 경유하여 송신 요구 대상 기기 주소(장치)에서, S로 지정한 송신 시작 CH부터 C로 지정한 수신 CH 수 분량의 데이터를 CPU 버스 또는 네트워크를 통해 수신하고 D로 지정한 송신 요구 원본(로컬 노드)의 수신 시작 CH부터 씁니다.

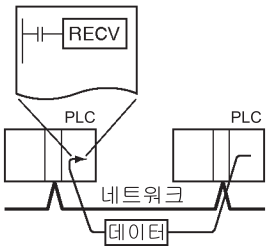


- C로 지정한 내부 논리 포트 번호가 실행 가능(네트워크 통신 명령 실행 가능 플래그가 ON)한 경우, 지정된 내부 논리 포트 번호의 네트워크 통신 실행 에러 플래그(내부 논리 포트 번호 00~07: A219.00~A219.07)와 네트워크 통신 실행 가능 플래그(내부 논리 포트 번호 00~07: A202.00~A202.07)를 OFF하고 네트워크 통신 응답 코드에 #0000을 기입합니다. 그 후, 네트워크상의 노드의 데이터를 수신합니다.
- D에 EM 영역(현재 뱅크)을 지정한 경우, 송신 원본(상대 노드)의 EM 영역의 현재 뱅크 데이터를 수신합니다.
- 자체 네트워크 이외로 데이터를 수신(네트워크 이탈)하는 경우, 라우팅 테이블(통신 상대의 PLC가 연결된 네트워크까지의 통신 경로를 등록한 테이블)을 각 네트워크의 PLC(CPU 유니트)에 등록해야 합니다.
- RECV 명령은 응답으로 데이터를 회신해야 하므로 응답이 필요한 것으로 설정합니다.
  응답이 필요한 것으로 설정한 경우는 응답 감시 시간 이내에 응답이 없으면 응답이 있을 때까지 지정된 재송신 횟수만큼 송신을 요구합니다.
- 장치뿐만 아니라 장치의 시리얼 포트에 대해서도 이 명령을 실행할 수 있습니다.

  Controller Link 네트워크나 Ethernet상의 PLC, PC 또는 상위 링크 모드로 시리얼 포트에 연결된 상위 컴퓨터에서 데이터를 수신할 수 있습니다.

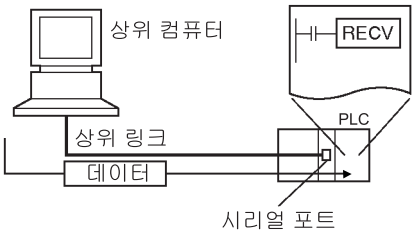
■ 네트워크상의 PLC(CPU 유니트) 또는 PC에서 데이터 수신

Controller Link 네트워크나 Ethernet상의 PLC(CPU 유니트) 또는 PC에 대해 이 RECV 명령을 실행한 경우, 상대방의 지정된 I/O 메모리 영역의 데이터를 자신의 지정된 I/O 메모리 영역으로 수신할 수 있습니다.



■ 상위 링크로 연결된 상위 컴퓨터에서 데이터 수신(송신 기능)

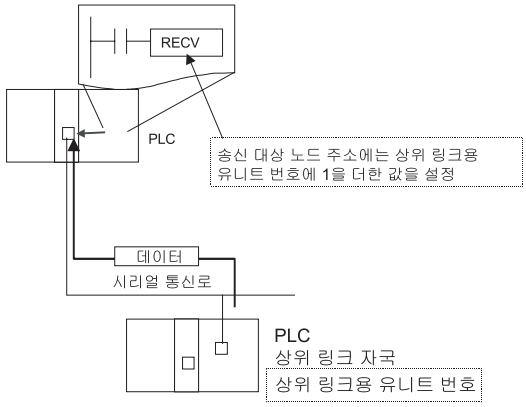
CPU 유니트 내장 시리얼 포트, 시리얼 커뮤니케이션 보드(CS 시리즈만 해당), 시리얼 커뮤니케이션 유니트 중 하나가 상위 컴퓨터(PC 등)와 상위 링크 모드로 1 : 1 연결되어 있을 때, 해당 유니트 중 하나의 포트에 대해 RECV 명령을 실행한 경우, PLC 측이 최초의 명령 송신권을 가지며 상위 컴퓨터에서 CPU 유니트의 지정된 I/O 메모리 영역에 데이터를 수신할 수 있습니다. (자체 PLC 이외에 네트워크상의 PLC에 연결된 상위 컴퓨터에서 수신하는 것도 가능합니다.)



■ 시리얼 포트(시리얼 게이트웨이 기능)에 연결된 상위 링크 자국의 PLC에서 데이터 수신

시리얼 커뮤니케이션 보드/유니트의 시리얼 통신로에 상위 링크 자국으로 연결된 PLC에서 시리얼 게이트웨이 기능을 통해 데이터를 수신하는 것도 가능합니다.

단, 이 경우, 송신 대상 노드 주소에는 상위 링크용 유니트 번호에 1을 더한 값을 설정한다는 점에 주의해야 합니다.



힌트

■ 상위 링크로 연결된 상위 컴퓨터에서 데이터 수신(송신 기능)

RECV 명령을 CPU 유니트 내장 시리얼 포트, 시리얼 커뮤니케이션 보드(CS 시리즈만 해당), 시리얼 커뮤니케이션 유니트 중 한 유니트의 시리얼 포트에 대해 실행하면 그 중 한 유니트의 시리얼 포트에서 상위 컴퓨터에 대해 명령(FINS 메시지를 상위 링크의 헤더나 터미네이터에 포함시킨 명령)을 전송합니다. 전송하는 FINS 명령은 I/O 메모리 영역 읽기 명령(명령 코드 "#0101")이고 상위 링크 헤더 코드는 "#0F"입니다.

상위 컴퓨터 측에서는 수신한 해당 명령(상위 링크에 포함시킨 FINS 명령)을 처리하는 프로그램을 작성해야 합니다.

자체 PLC상의 수신처인 경우, C+1의 송신 대상 네트워크 주소를 자체 네트워크(#00)에 설정하고, C+2의 송신 대상 노드 주소를 자체 PLC(#00)에 설정합니다. C+2의 송신 대상 기기 주소는 CPU 유니트인 경우 #00, INNER 보드인 경우 #E1, 범용 시리얼 유니트인 경우 해당 유니트 번호+#10(16진수)을 지정합니다.

사용상의 주의 사항

- 1개의 내부 논리 포트에 대해 한번에 1개 명령만 실행할 수 있습니다. 따라서 각 포트에 대응된 네트워크 통신 명령 실행 가능 플래그가 0(OFF)인 경우는 해당 포트에 대해 이 명령을 실행하지 않도록 네트워크 통신 명령 실행 가능 플래그(A202.00~A202.07)를 a 접점의 입력 조건으로 하십시오.
- 내부 논리 포트 번호를 자동 할당("F")으로 사용하는 경우에 대해서는 3-736의 '내부 논리 포트 자동 할당 기능'의 설명을 참조하십시오.
- 내부 논리 포트 번호는 네트워크 통신 명령과 PMCR(프로토콜 매크로) 명령에서 공용입니다. 따라서 동일한 내부 논리 포트 번호를 사용하여 PMCR 명령과 네트워크 통신 명령을 동시에 실행할 수 없으므로 주의하십시오.
- 노이즈 등으로 인해 통신 도중에 송신 요구 메시지나 응답이 소실되는 경우 등을 고려하여 RECV 명령을 실행할 때는 재송신 횟수를 0 이외로 설정하고, 응답이 감시 시간을 초과해도 돌아오지 않는 경우 재송신 처리를 수행할 것을 권장합니다.

RECV2

명령 이름	니모닉	동작 옵션	평션 번호	기능
네트워크 수신 2	RECV2	@RECV2	492	네트워크상의 노드에 대해 송신을 요구하고 데이터를 수신합니다.

심볼	RECV2
	

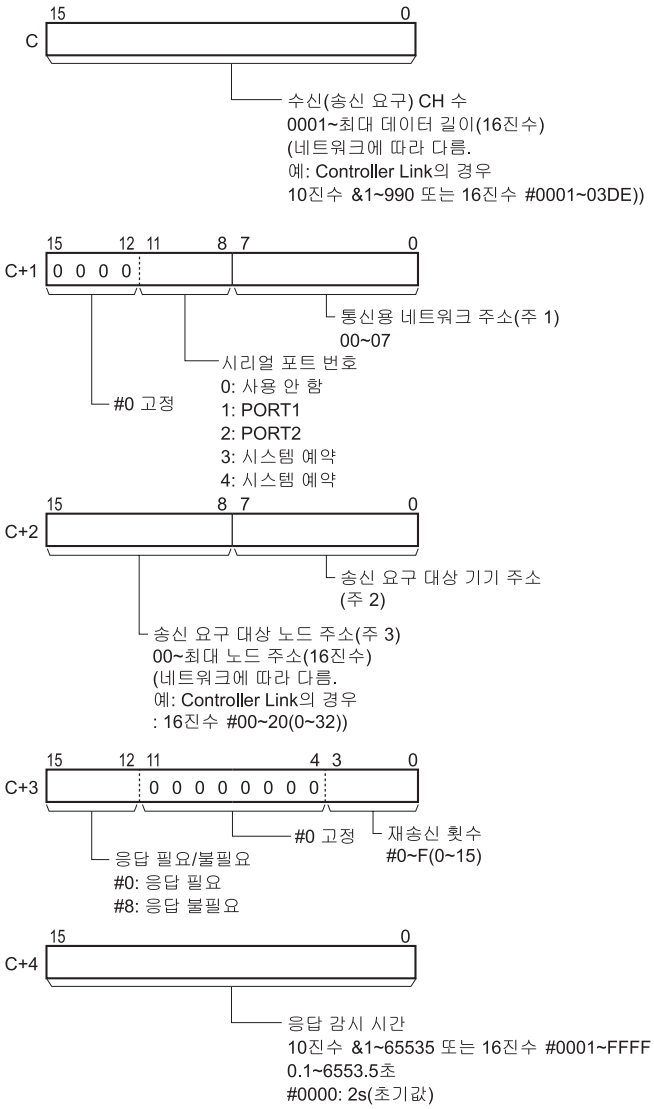
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

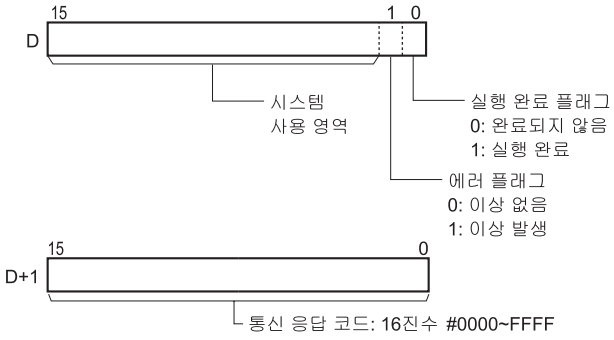
피연산자	내용	데이터 형식	크기
S	송신 요구 대상 송신 시작 채널	WORD	가변
D1	송신 요구 원본 수신 시작 채널	WORD	가변
C	제어 데이터	WORD	5
D2	통신 정보	WORD	2

C: 제어 데이터



D2: 통신 정보

D1, D2는 명령을 실행하기 전에 0으로 클리어 하십시오.



주 1: 통신용 네트워크 주소는 라우팅 테이블로 설정하십시오.
단, 자체 네트워크를 경유하여 명령을 실행하는 경우는 라우팅 테이블이 필요하지 않습니다. 통신용 네트워크 주소: 00으로 설정하십시오.
자체 네트워크란 여러 대의 CPU 고기능 유니트가 장착된 경우, 유니트 번호가 가장 작은 유니트가 연결된 네트워크를 말합니다.

주 2: 기기 주소:

- CPU 유니트: 16진수 #00
- CPU 고기능 유니트: 16진수 #10+유니트 번호(16진수)
- 고기능 I/O 유니트: 16진수 #20+기기 번호(16진수) (C200H 시리즈 고기능 I/O 유니트는 제외)
- INNER 보드: 16진수 #E1(CS 시리즈만 해당)
- 컴퓨터: 16진수 #01
- 네트워크에 연결된 유니트(상대 유니트의 유니트 번호를 특정할 필요 없음): 16진수 #FE

시리얼 포트의 기기 주소:

- 시리얼 커뮤니케이션 유니트의 경우
  포트 1: 16진수 #80+16진수 #04×유니트 번호(16진수)
  포트 2: 16진수 #81+16진수 #04×유니트 번호(16진수)
- 시리얼 커뮤니케이션 보드의 경우
  포트 1: 16진수 #E4(10진수로 228)
  포트 2: 16진수 #E5(10진수로 229)
- CPU 유니트의 경우
  페리페럴 포트: 16진수 #FD(10진수로 253)
  RS-232C 포트: 16진수 #FC(10진수로 252)

주 3: RECV2 명령에서는 전체 동시 송신이 불가능합니다. 로컬 노드 내부로 송신을 요구하는 경우 #00을 설정합니다.

RECV2

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
D1																		
C												–						
D2																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C로 지정한 시리얼 포트 번호가 0~4Hex가 아닐 때 ON · 빈 내부 논리 포트가 없을 때(A211=0) ON · 그 이외일 때 OFF

기능

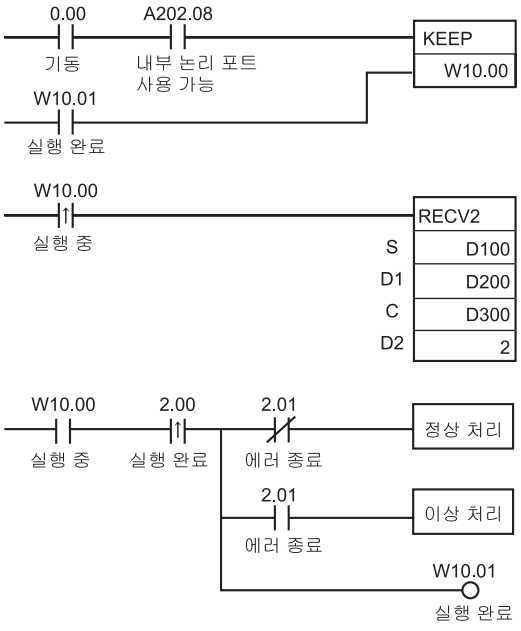
C로 지정한 네트워크 주소의 송신 요구 대상에서 노드 주소를 경유한 송신 요구 대상 기기 주소의 S로 지정한 송신 시작 CH부터 C로 지정한 수신 CH 수 분량의 데이터를 CPU 버스 또는 네트워크를 통해 수신하여 D로 지정한 송신 요구 원본(로컬 노드)의 수신 시작 CH부터 씁니다.

송신을 시작하면 D2의 실행 완료 플래그를 OFF로 합니다. 응답이 돌아오면 송신 결과의 정상/이상이 D2의 에러 플래그에 반영됩니다. 통신 응답은 D2＋1에 기록됩니다.
기존 RECV 명령의 통신 가능 플래그(A202) 및 네트워크 통신 실행 에러 플래그(A219)가 피연산자 D2에 해당하고, 네트워크 통신 응답 코드(A203~210)가 D2+1에 해당합니다.

샘플 프로그램

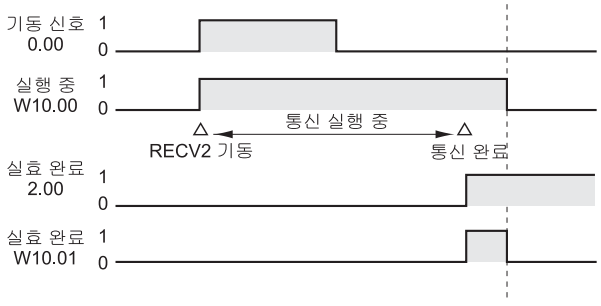
0.00(기동) 및 A202.08(내부 논리 포트 사용 가능)이 ON이면 W10.00(실행 중)이 ON이 됩니다. W10.00(실행 중)이 ON이면 자체 네트워크상의 노드 주소 3의 CPU 유니트 D100부터 10CH(D100~D109)의 데이터를 D200~10CH(D200~D209)에 수신합니다. 응답 감시 시간 10초 후 응답이 돌아오지 않은 경우는 재송신 처리를 최대 3회까지 실행합니다.

정상일 때는 2.00(실행 완료)이 ON되고 정상 처리를 수행합니다. 이상일 때는 2.01(에러 종료)이 ON되고 이상 처리를 수행한 후 종료합니다.



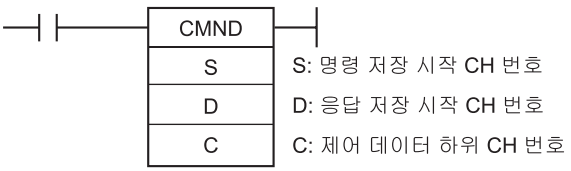
C: D300	0	0	1	0	송신 CH 수 = 10CH
C+1: D301	0	0	0	0	자체 네트워크상의 장치 자체에 대해
C+2: D302	0	3	0	0	송신 대상 노드 주소 = #03, 기기 주소 = #00(CPU 유니트)
C+3: D303	0	0	0	3	응답 필요, 재송신 횟수 = #03
C+4: D304	0	1	0	0	응답 감시 시간 = #0064(10초)

■ 타임 차트



CMND

명령 이름	니모닉	동작 옵션	펑션 번호	기능
명령 송신	CMND	@CMND	490	임의의 FINS 명령을 전송하고 응답을 수신합니다.

심볼	CMND
	

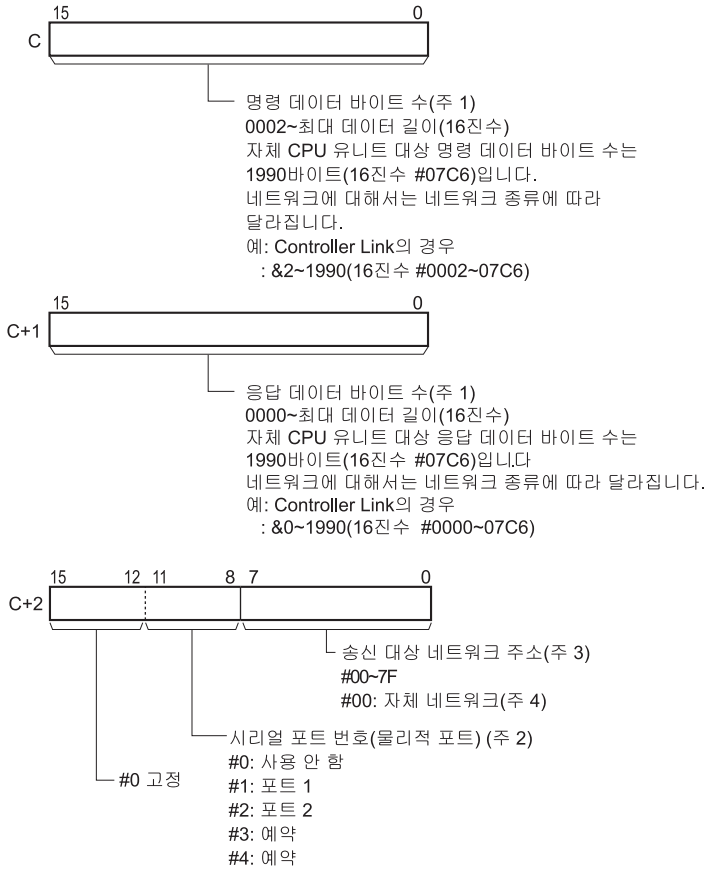
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

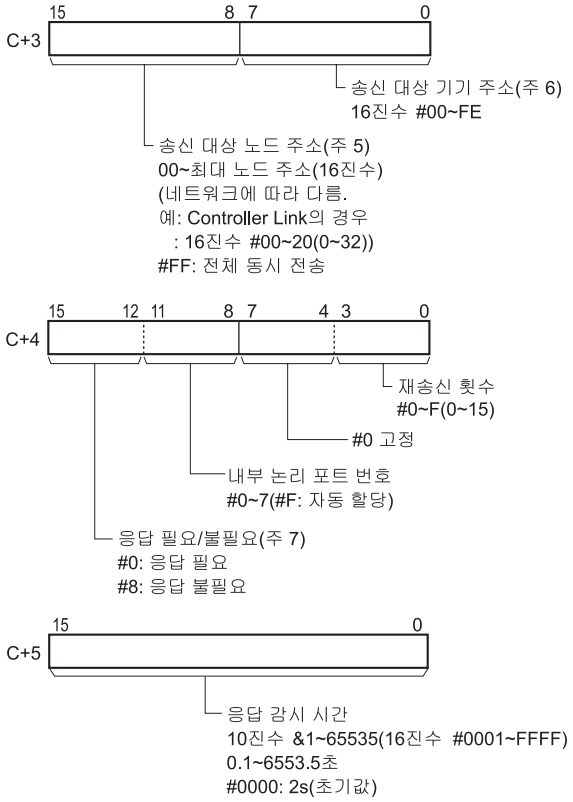
피연산자 설명

피연산자	내용	데이터 형식	크기
S	명령 저장 시작 CH 번호	UINT	가변
D	응답 저장 시작 CH 번호	UINT	가변
C	제어 데이터 하위 CH 번호	WORD	6

C: 제어 데이터 하위 CH 번호



CMND



주 1: 명령 데이터 바이트 수 및 응답 데이터 바이트 수의 최대 데이터 길이에 대해서는 사용하는 네트워크의 유니트 설명서를 참조하십시오.

명령 데이터 바이트 수 및 응답 데이터 바이트 수의 최대 데이터 길이는 FINS 명령의 경로가 되는 네트워크 중 데이터 길이가 최소인 네트워크에 의해 결정됩니다.

응답 데이터 바이트 수보다 긴 응답 데이터가 반송된 경우, 응답 데이터는 저장되지 않습니다. 응답 데이터 바이트 수보다 짧은 응답 데이터가 반송된 경우에는 응답 데이터를 수신하며 나머지 영역은 이전 상태 그대로 유지됩니다.

주 2: 상위 링크의 송신 기능 또는 시리얼 게이트웨이 기능(CompoWay/F, Modbus-RTU 또는 Modbus-ASCII로 변환)에 의해 시리얼 포트 대상으로 FINS 명령을 전송하는 경우, 다음 두 가지 방법 중 하나를 사용합니다.

1) C+3의 비트 00~07의 송신 대상 기기 주소에 유니트/보드(CPU 유니트, 시리얼 커뮤니케이션 유니트/보드)의 기기 주소를 지정하여 C+2의 비트 8~11(시리얼 포트 번호)에 시리얼 통신의 포트 번호(포트 번호 1: #1, 포트 번호 2: #2)를 설정하는 방법

송신 대상 기기 주소 C+3 비트 00~07	유니트(장치)	시리얼 포트 번호 C+2 비트 08~11	시리얼 포트
16진수 #00	CPU 유니트	#1	내장 RS-232C 포트 사용
		#2	페리페럴 포트 사용
16진수 #10+유니트 번호(16진수)	시리얼 커뮤니케이션 유니트(CPU 고기능 유니트)	#1	포트 1 사용
		#2	포트 2 사용
16진수 #E1	시리얼 커뮤니케이션 보드(INNER 보드) (CS 시리즈만 해당)	#1	포트 1 사용
		#2	포트 2 사용

2) C+3의 비트 00~07의 송신 대상 기기 주소에 다음과 같은 시리얼 포트의 기기 주소를 직접 설정하는 방법(이 경우, C+2의 비트 08~11의 시리얼 포트 번호(물리적 포트)는 0: 사용 안 함으로 설정)

・ 시리얼 커뮤니케이션 유니트의 경우

시리얼 커뮤니케이션 유니트의 시리얼 포트	시리얼 포트의 기기 주소	예) 유니트 번호 1의 경우
포트 1	16진수 #80+16진수 #04×유니트 번호(16진수)	#80+#04×1=#84(10진수로 132)
포트 2	16진수 #81+16진수 #04×유니트 번호(16진수)	81+#04×1=#85(10진수로 133)

・ 시리얼 커뮤니케이션 보드의 경우

시리얼 커뮤니케이션 보드의 시리얼 포트	시리얼 포트의 기기 주소
포트 1	16진수 #E4(10진수로 228)
포트 2	16진수 #E5(10진수로 229)

・CPU 유니트의 경우

CPU 유니트의 시리얼 포트	시리얼 포트의 기기 주소
페리페럴 포트	16진수 #FD(10진수로 253)
RS-232C 포트	16진수 #FC(10진수로 252)

주 3: 시리얼 게이트웨이 기능(상위 링크 FINS로 변환)으로 라우팅 테이블 없이 시리얼 포트를 지정하는 경우는 송신 대상 네트워크 주소에 시리얼 포트의 기기 주소를 설정합니다.

주 4: 여러 대의 CPU 고기능 유니트가 장착된 경우, 유니트 번호가 가장 작은 유니트의 네트워크 주소가 됩니다.

주 5: 전체 동시 송신의 경우, #FF로 설정합니다. 로컬 노드 내부로 송신하는 경우 #00을 설정합니다.
시리얼 게이트웨이 기능(상위 링크 FINS로 변환)에서 시리얼 포트를 지정하는 경우는 송신 대상 노드 주소에 송신 대상 PLC의 상위 링크용 유니트 번호에 1을 더한 값(1~32)을 설정합니다.

주 6: 기기 주소

・CPU 유니트: #00(16진수)
・CPU 고기능 유니트: 유니트 번호+#10(16진수)
・고기능 I/O 유니트: 기기 번호+#20(16진수)
(C200H 시리즈 고기능 I/O 유니트 제외)
・INNER 보드: #E1(16진수) (CS 시리즈만 해당)
・컴퓨터: #01(16진수)
・네트워크에 연결된 유니트(상대 유니트의 유니트 번호를 특정할 필요 없음): #FE(16진수)

시리얼 포트의 기기 주소:

・시리얼 커뮤니케이션 유니트의 경우
포트 1: #80(16진수)+#04(16진수)×유니트 번호
포트 2: #81(16진수)+#04(16진수)×유니트 번호
・시리얼 커뮤니케이션 보드의 경우
포트 1: #E4(16진수) (10진수로 228)
포트 2: #E5(16진수) (10진수로 229)
・CPU 유니트의 경우
페리페럴 포트: #FD(16진수) (10진수로 253)
RS-232C 포트: #FC(16진수) (10진수로 252)

주 7: 전체 동시 송신하는 경우는 응답이 필요한 것으로 설정해도 응답이 돌아오지 않습니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		
C																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	・C로 지정한 시리얼 포트 번호가 0~4가 아닐 때 ON ・C로 지정한 내부 논리 포트 번호에 대응하는 네트워크 명령 실행 가능 플래그가 OFF일 때 ON ・자체 CPU 유니트 대상 FINS 명령을 전송했을 때 파일 메모리 조작 실행 중 플래그(A343.13)가 ON인 경우 ON ・인터럽트 태스크에서 실행했을 때 ON(CJ2 CPU 유니트에서 인터럽트 고속 기동 기능 사용 시) ・그 이외일 때 OFF

CMND

관련 특수 보조 릴레이

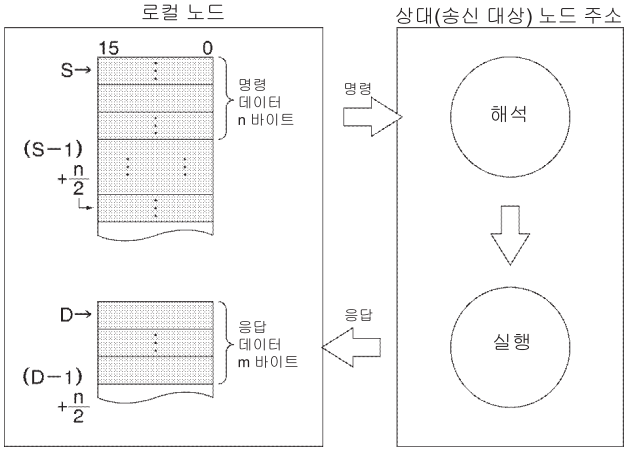
이름	주소	내용
네트워크 통신 명령 실행 가능 플래그	A202.00~ A202.07	네트워크 통신(PMCR 명령 포함)이 실행 가능할 때 1(ON)이 됩니다. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 네트워크 통신 실행 중에 0(OFF), 실행 종료(정상이든 이상이든) 후에는 1(ON)이 됩니다.
네트워크 통신 실행 에러 플래그	A219.00~ A219.07	네트워크 통신 실행 시 에러(이상)가 발생한 경우 1(ON)이 됩니다. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 다음 네트워크 통신 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 통신 명령 실행에서 0(OFF)이 됩니다.
네트워크 통신 응답 코드	A203CH~ A210CH	네트워크 통신이 실행된 경우 응답 코드(종료 코드)가 저장됩니다. 각 CH는 내부 논리 포트 번호에 대응합니다. A203~210CH: 내부 논리 포트 번호 00~07 통신 명령 실행 중에는 #0000이 되고 통신 명령 실행이 종료되면 반영됩니다. 이 값은 운전을 시작할 때 클리어됩니다.
파일 메모리 조작 실행 중 플래그	A343.13	자체 CPU 유니트 대상 CMND 명령 실행 중(파일 메모리 관련 이외의 FINS 명령 전송 포함) 또는 다음 중 하나의 파일 메모리 관련 조작 실행 중에 1(ON). 아무것도 실행하지 않을 때는 0(OFF) · FREAD/FWRIT 명령 실행 · 특수 보조 릴레이에 의한 프로그램 교체 중 · 간이 백업 조작 실행 중

주:

· 네트워크 통신 응답 코드에 대해서는 'CS/CJ 시리즈 통신 명령 참조 설명서'(번호 SBCA-304)의 FINS 명령 종료 코드 목록을 참조하십시오.

기능

S로 지정한 명령 저장 시작 CH 번호부터 명령 데이터 바이트 수만큼 임의의 FINS 명령 데이터를 CPU 버스 또는 네트워크를 통해, C로 지정한 네트워크 주소의 노드 주소를 사용하여 지정된 기기 주소(장치)에 송신합니다. 그 응답은 D를 선두로 하여 응답 데이터 바이트 수만큼 저장합니다.



· C로 지정한 내부 논리 포트 번호가 실행 가능(네트워크 통신 명령 실행 가능 플래그가 ON)한 경우, 지정된 내부 논리 포트 번호의 네트워크 통신 실행 에러 플래그(포트 번호 00~07: A219.00~A219.07)와 네트워크 통신 실행 가능 플래그(내부 논리 포트 번호 00~07: A202.00~A202.07)를 OFF하고 네트워크 통신 응답 코드에 #0000을 씁니다. 그 후, 네트워크상의 노드에 FINS 명령을 전송합니다.

· 송신 대상 노드는 라우팅 테이블(통신 상대의 PLC가 연결된 네트워크까지의 통신 경로를 등록한 테이블)을 사용하여 찾습니다.

· 장치뿐만 아니라 장치의 시리얼 포트에 대해서도 이 명령을 실행할 수 있습니다.

· CPU 유니트가 이 명령을 실행하여 자신에 대해 FINS 명령을 전송하는 것(자체 CPU 유니트 대상 FINS 명령 전송)도 가능합니다(단, CS 시리즈 CS1 CPU 유니트(-V□ 없음)는 제외). 이 경우, 제어 데이터(C)의 값은 다음과 같이 설정하십시오.

  · 송신 대상 네트워크 주소(C+2의 비트 0~7): #00(자체 네트워크)

  · 시리얼 포트 번호(C+2의 비트 8~11): #0(사용 안 함)

  · 송신 대상 기기 주소(C+3의 비트 0~7): #00(CPU 유니트)

  · 송신 대상 노드 주소(C+3의 비트 8~15): #00(로컬 노드 내)

  · 재송신 횟수(C+4의 비트 0~3): #0(재송신 횟수 지정은 무효이므로 0회로 설정)

  · 응답 감시 시간(C+5의 비트 0~15): 10진수 &0~65535 또는 16진수 #0000~FFFF 가능(단, 0을 설정한 경우에는 2s가 아니라 6553.5s에서 동작합니다.)

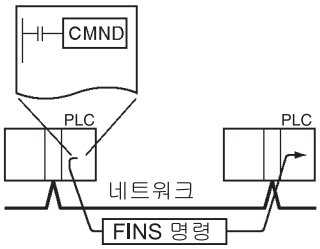
· 송신 대상 노드 주소가 #FF일 때는 지정된 네트워크 주소의 모든 노드에 동일한 명령을 전송합니다(전체 동시 전송).

- 응답이 필요한 것으로 설정한 경우는 응답 감시 시간 이내에 응답이 없으면 응답이 있을 때까지 지정된 재송신 횟수만큼 명령을 전송합니다. 응답이 존재하지 않는 명령에 대해서는 응답 필요 없음으로 설정하십시오.
- 전체 동시 전송을 지정한 경우는 응답이 없으며 재송신도 하지 않습니다.
- 설정한 응답 데이터 바이트 수보다 많은 데이터가 응답으로 반환된 경우는 에러가 됩니다.

Controller Link 네트워크나 Ethernet상의 PLC(CPU 유니트, CPU 고기능 유니트 또는 INNER 보드(CS 시리즈만 해당)), PC 또는 상위 링크 모드로 시리얼 포트에 연결된 상위 컴퓨터에 대해 임의의 FINS 명령을 전송할 수 있습니다.

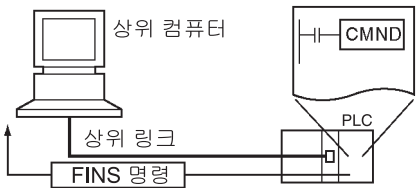
■ 네트워크상의 PLC(CPU 유니트, CPU 고기능 유니트 또는 INNER 보드(CS 시리즈만 해당)) 또는 PC로 임의의 FINS 명령 전송

Controller Link 네트워크나 Ethernet상의 PLC(CPU 유니트, CPU 고기능 유니트 또는 INNER 보드(CS 시리즈만 해당)) 또는 PC에 대해 CMND 명령을 실행한 경우, 임의의 FINS 명령을 상대방에게 전송할 수 있습니다.



■ 상위 링크로 연결된 상위 컴퓨터로 임의의 FINS 명령 전송(송신 기능)

CPU 유니트 내장 시리얼 포트, 시리얼 커뮤니케이션 보드(CS 시리즈만 해당), 시리얼 커뮤니케이션 유니트 중 하나가 상위 컴퓨터(PC 등)와 상위 링크 모드로 1 : 1 연결되어 있을 때, 해당 유니트 중 하나의 시리얼 포트(C+2의 비트 8~11에서 #1 또는 #2 지정)에 대해 CMND 명령을 실행한 경우, PLC 측이 최초의 송신권을 가지며 상위 컴퓨터로 임의의 FINS 명령을 전송할 수 있습니다. (자체 PLC 이외에 네트워크상의 PLC에 연결된 상위 컴퓨터에 대해 명령을 전송하는 것도 가능합니다.)



■ 시리얼 포트(시리얼 게이트웨이 기능)에 연결된 기기(구성 요소 또는 상위 링크 자국의 PLC)에 대한 FINS 명령 전송

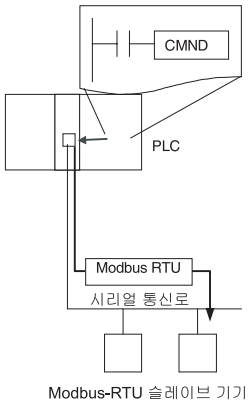
- 구성 요소(CompoWay/F, Modbus-RTU 또는 Modbus-ASCII로 변환할 때):

  시리얼 커뮤니케이션 보드/유니트의 시리얼 포트 또는 CPU 유니트의 시리얼 포트(페리페럴 포트 또는 RS-232C 포트)에 다음 FINS 명령을 전송하여 시리얼 게이트웨이 기능을 통해 CompoWay/F, Modbus-RTU 또는 Modbus-ASCII 명령으로 변환 하는 것도 가능합니다.

  CompoWay/F 명령으로 변환할 때: #2803

  Modbus-ASCII 명령으로 변환할 때: #2805(CPU 유니트의 시리얼 포트는 불가)

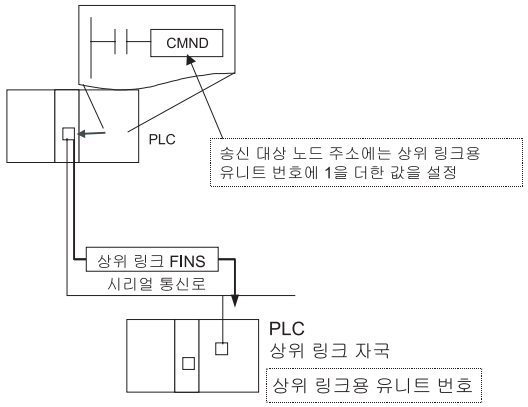
  Modbus-RTU 명령으로 변환할 때: #2804(CPU 유니트의 시리얼 포트는 불가)



· 상위 링크 자국으로서의 PLC 대상(상위 링크 FINS로 변환할 때):

시리얼 커뮤니케이션 보드/유니트의 시리얼 통신로에 상위 링크 자국으로 연결된 PLC로 시리얼 게이트웨이 기능을 통해 임의의 FINS 명령을 전송하는 것도 가능합니다.

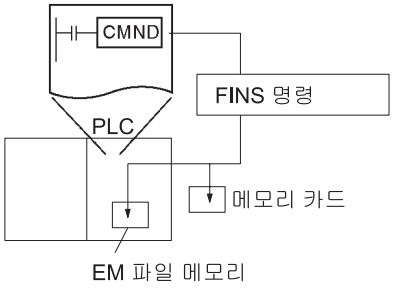
단, 이 경우, 송신 대상 노드 주소에는 상위 링크용 유니트 번호에 1을 더한 값을 설정한다는 점에 주의해야 합니다.



■ 자체 CPU 유니트에 임의의 FINS 명령 전송

CPU 유니트가 자신에 대해 임의의 FINS 명령을 전송하는 것도 가능합니다(단, CS 시리즈 CS1 CPU 유니트(-V□ 없음)는 제외). 주로 파일 메모리 관련 FINS 명령(명령 코드 #22□□: 파일 메모리의 포맷, 파일 삭제, 파일 복사 등)을 전송하여 파일 메모리를 조작할 때 사용합니다(자세한 내용은 SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편의 '5-2 파일 조작 방법' 또는 SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편 '7-3 파일 메모리용 조작' 참조).

또한 자체 CPU 유니트 대상 FINS 명령을 실행하는 중에는(파일 메모리 관련 이외의 FINS 명령을 전송한 경우도) 파일 메모리 조작 실행 중 플래그(A343.13)가 1(ON)이 됩니다. 따라서 반드시 파일 메모리 조작 실행 중 플래그(A343.13)를 b 접점의 입력 조건으로 하여 배타적 제어를 하십시오.



힌트

· 이 명령으로 FINS 명령의 I/O 메모리 영역 쓰기 명령(명령 코드: #0102)을 전송하는 것과 SEND 명령은 동등합니다. 또한 이 명령으로 FINS 명령의 I/O 메모리 영역 읽기 명령(명령 코드: #0101)을 전송하는 것과 RECV 명령은 동등합니다.

· 상위 링크로 연결된 상위 컴퓨터로 FINS 명령 전송(송신 기능)

CMND 명령을 CPU 유니트 내장 시리얼 포트, 시리얼 커뮤니케이션 보드(CS 시리즈만 해당), 시리얼 커뮤니케이션 유니트 중 한 유니트의 시리얼 포트(C+2의 비트 8~11에서 #1 또는 #2를 지정)에 대해 실행하면 그 중 한 유니트의 시리얼 포트에서 상위 컴퓨터에 대해 명령(FINS 메시지를 상위 링크의 헤더나 터미네이터에 포함시킨 명령)을 전송합니다. 전송하는 FINS 명령은 임의로 지정할 수 있습니다. 상위 링크 헤더 코드는 "#0F"입니다.
상위 컴퓨터 측에서는 수신한 해당 명령(상위 링크에 포함시킨 FINS 명령)을 처리하는 프로그램을 작성해야 합니다.
자체 PLC상의 수신처인 경우, C+1의 송신 대상 네트워크 주소를 자체 네트워크(#00)에 설정하고, C+2의 송신 대상 노드 주소를 자체 PLC(#00)에 설정합니다. C+2의 송신 대상 기기 주소는 CPU 유니트의 경우 #00, 시리얼 커뮤니케이션 보드(CS 시리즈만 해당)의 경우 #E1, 시리얼 커뮤니케이션 유니트의 경우, 해당 유니트 번호+#10을 지정합니다.

사용상의 주의 사항

· 1개의 내부 논리 포트에 대해 한번에 1개 명령만 실행할 수 있습니다. 따라서 각 포트에 대응된 네트워크 통신 명령 실행 가능 플래그가 0(OFF)인 경우는 해당 포트에 대해 이 명령을 실행하지 않도록 네트워크 통신 명령 실행 가능 플래그(A202.00~A202.07)를 a 접점의 입력 조건으로 하십시오.

· 내부 논리 포트 번호를 자동 할당("F")으로 사용하는 경우에 대해서는 3-736의 '내부 논리 포트 자동 할당 기능'의 설명을 참조하십시오.

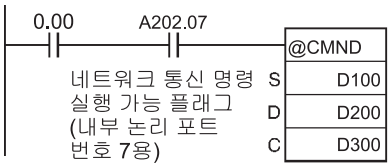
· 내부 논리 포트 번호는 네트워크 통신 명령과 시리얼 통신 명령(TXDU, RXDU, PMCR)에서 공용입니다. 따라서 동일한 내부 논리 포트 번호를 사용하여 시리얼 통신 명령과 네트워크 통신 명령을 동시에 실행할 수 없으므로 주의하십시오.

· 자체 CPU 유니트 대상 FINS 명령을 전송하는 경우는 네트워크 통신 명령 실행 가능 플래그(A202.00~A202.07)를 a 접점, 파일 메모리 조작 실행 중 플래그(A343.13)를 b 접점의 입력 조건으로 하십시오.

· 노이즈 등으로 인해 통신 도중에 FINS 명령 메시지나 응답이 소실되는 경우 등을 고려하여 CMND 명령을 실행할 때는 재송신 횟수를 0 이외로 설정하고, 응답이 감시 시간을 초과해도 돌아오지 않는 경우 재송신 처리를 수행할 것을 권장합니다.

샘플 프로그램

■ 네트워크상의 다른 CPU 유니트에 FINS 명령 전송

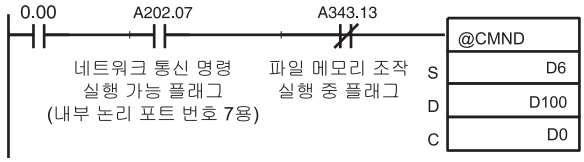


0.00 및 A202.07(내부 논리 포트 번호 7용 네트워크 통신 명령 실행 가능 플래그)이 ON일 때, 자체 네트워크상의 노드 주소 3의 CPU 유니트에 대해 FINS 명령 #0101(I/O 메모리 영역 읽기 명령)을 송신하고, 응답을 D200~D211에 저장합니다.
(FINS 명령은 D10~D19의 10CH를 읽습니다.
응답은 명령 코드 #0101의 2바이트와 종료 코드의 2바이트 뒤에 10CH의 데이터가 돌아옵니다: 합계 14CH = 28바이트)

	15 87 0	
S: D100	0 1 0 1	명령 코드 #0101 = I/O 메모리 영역 읽기
S+1:D101	8 2 0 0	D10(I/O 메모리 종류 = #82, 주소 = #000A00)
S+2:D102	0 A 0 0	
S+3:D103	0 0 0 A	읽기 요소 수 = #000A(10)

	15 87 0	
C: D300	0 0 0 8	명령 데이터 바이트 수 = #0008(8)
C+1:D301	0 0 1 8	응답 데이터 바이트 수 = #0018(24)
C+2:D302	0 0 0 0	자체 네트워크상의 장치 자체에 대해
C+3:D303	0 3 0 0	송신 대상 노드 주소 = #03, 기기 주소 = #00(CPU 유니트)
C+4:D304	0 7 0 3	응답 필요, 내부 논리 포트 7, 재송신 횟수 = #03
C+5:D305	0 0 6 4	응답 감시 시간 = #0064(10초)

■ 자체 CPU 유니트에FINS 명령 전송



0.00 및 A202.07(내부 논리 포트 번호 7용 네트워크 통신 명령 실행 가능 플래그)이 ON이고 A343.13(파일 메모리 조작 실행 중 플래그)이 OFF이면 자체 CPU 유니트 대상으로 FINS 명령 #2215(디렉토리 작성/삭제)를 전송하고 그 응답을 D100~D101에 저장합니다.

FINS 명령은 CPU 유니트에 장착된 메모리 카드 내, 예를 들어 '\OMRON' 디렉토리 아래에 'CS1'이라는 디렉토리를 작성합니다. 응답으로 명령 코드 #2215의 2바이트와 종료 코드의 2바이트가 반환됩니다.

	15 8 7 0	
S: D6	2 2 1 5	명령 코드=#2215(디렉토리 작성/삭제)
S+1: D7	8 0 0 0	디스크 번호=#8000(메모리 카드)
S+2: D8	0 0 0 0	파라미터=#0000(디렉토리 작성)
S+3: D9	4 3 5 3	하위 디렉토리명 = 'CS1□□□□□□.□□□' (□: 공백코드)
S+4: D10	3 1 2 0	
S+5: D11	2 0 2 0	
S+6: D12	2 0 2 0	
S+7: D13	2 E 2 0	
S+8: D14	2 0 2 0	
S+9: D15	0 0 0 6	디렉토리 길이=#0006(6문자)
S+10:D16	5 C 4 F	디렉토리 절대 경로 = '\OMRON'
S+11:D17	4 D 5 2	
S+12:D18	4 F 4 E	

	15 8 7 0	
C: D0	0 0 1 A	명령 데이터 바이트 수=#001A(26바이트)
C+1: D1	0 0 0 4	응답 데이터 바이트 수=#0004(4바이트)
C+2: D2	0 0 0 0	송신 대상 주소= #0000(자체 네트워크)
C+3: D3	0 0 0 0	#00(로컬 노드)/#00(CPU 유니트)
C+4: D4	0 7 0 0	응답 필요/내부 논리 포트 번호 7/재송신 횟수=0회
C+5: D5	0 0 0 0	응답 감시 시간=#0000(2초)

CMND2

명령 이름	니모닉	동작 옵션	펑션 번호	기능
명령 송신 2	CMND2	@CMND2	493	임의의 FINS 명령을 전송하고 응답을 수신합니다.

심볼	CMND2	
	CMND2	
	S	S: 송신 명령 시작 채널
	D1	D1: 응답 저장 대상 시작 채널
	C	C: 제어 데이터
	D2	D2: 통신 정보

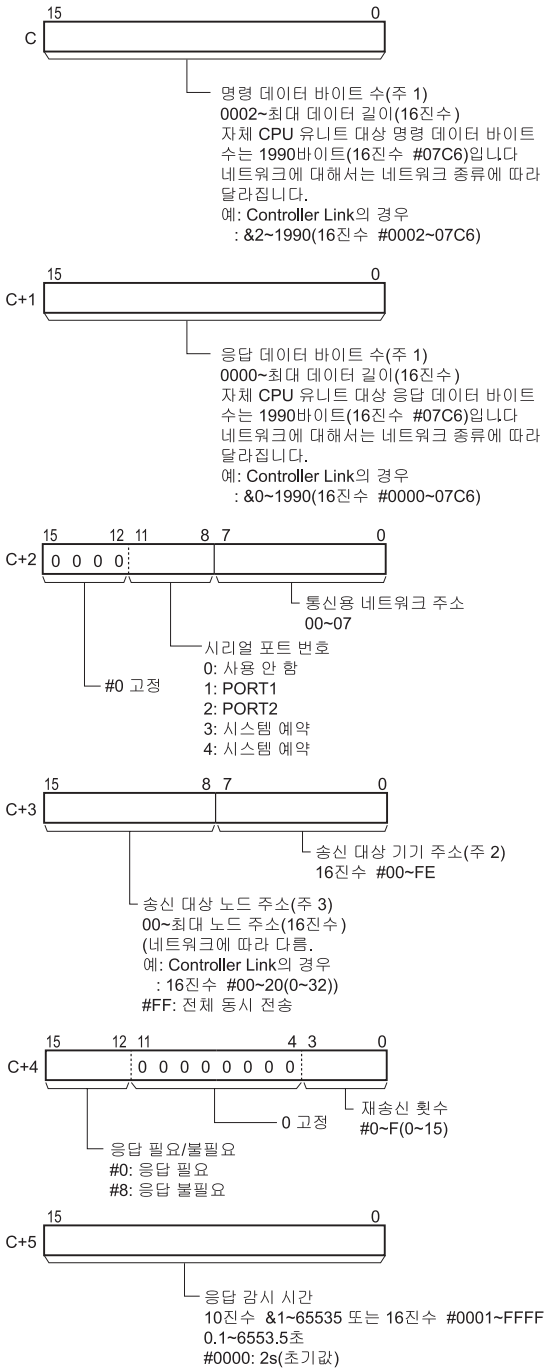
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

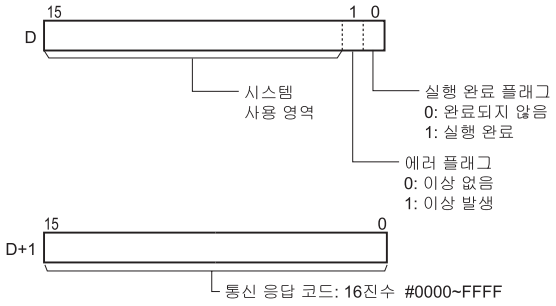
피연산자	내용	데이터 형식	크기
S	송신 명령 시작 채널	WORD	가변
D1	응답 저장 대상의 시작 채널	WORD	가변
C	제어 데이터	WORD	6
D2	통신 정보	WORD	2

C: 제어 데이터



D2: 통신 정보

D1, D2는 명령을 실행하기 전에 0으로 클리어하십시오.



주 1: 명령 데이터 바이트 수 및 응답 데이터 바이트 수의 최대 데이터 길이에 대해서는 사용하는 네트워크의 유니트 설명서를 참조하십시오.
명령 데이터 바이트 수 및 응답 데이터 바이트 수의 최대 데이터 길이는 FINS 명령의 경로가 되는 네트워크 중 데이터 길이가 최소인 네트워크에 의해 결정됩니다.
응답 데이터 바이트 수보다 긴 응답 데이터가 반송된 경우, 응답 데이터는 저장되지 않습니다. 응답 데이터 바이트 수보다 짧은 응답 데이터가 반송된 경우에는 응답 데이터를 수신하며 나머지 영역은 이전 상태 그대로 유지됩니다.

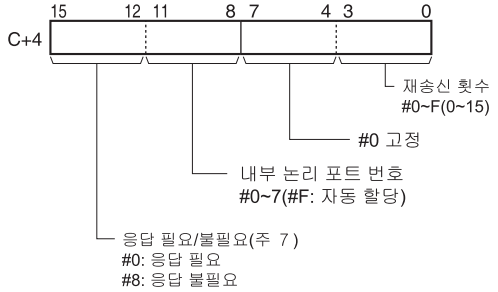
주 2: 기기 주소
- CPU 유니트: #00(16진수)
- CPU 고기능 유니트: 유니트 번호+#10(16진수)
- 고기능 I/O 유니트: 기기 번호+#20(16진수)
  (C200H 시리즈 고기능 I/O 유니트 제외)
- INNER 보드: #E1(16진수) (CS 시리즈만 해당)
- 컴퓨터: #01(16진수)
- 네트워크에 연결된 유니트(상대 유니트의 유니트 번호를 특정할 필요 없음): #FE(16진수)

시리얼 포트의 기기 주소:
- 시리얼 커뮤니케이션 유니트의 경우
  포트 1: #80(16진수)+#04(16진수)×유니트 번호
  포트 2: #81(16진수)+#04(16진수)×유니트 번호
- 시리얼 커뮤니케이션 보드의 경우

포트 1: #E4(16진수) (10진수로 228)
포트 2: #E5(16진수) (10진수로 229)

・CPU 유니트의 경우
페리페럴 포트: #FD(16진수) (10진수로 253)
RS-232C 포트: #FC(16진수) (10진수로 252)

주 3: 전체 동시 송신의 경우, #FF로 설정합니다. 로컬 노드 내부로 송신하는 경우 #00을 설정합니다.
시리얼 게이트웨이 기능(상위 링크 FINS로 변환)에서 시리얼 포트를 지정하는 경우는 송신 대상 노드 주소에 송신 대상 PLC의 상위 링크용 유니트 번호에 1을 더한 값(1~32)을 설정합니다.



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
D1																		
C												–						
D2																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	・C로 지정한 시리얼 포트 번호가 0~4Hex가 아닐 때 ON ・빈 내부 논리 포트가 없을 때(A211=0) ON ・그 이외일 때 OFF

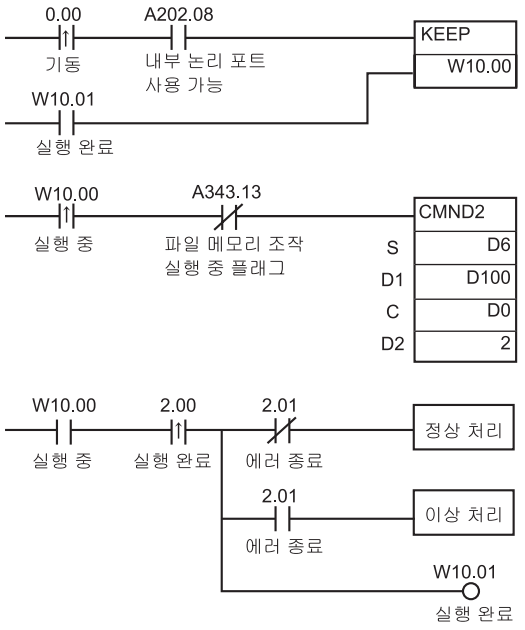
기능

S로 지정한 명령 저장 시작 CH 번호에서 명령 데이터 바이트 수만큼 임의의 FINS 명령 데이터를 CPU 버스 또는 네트워크를 통해 C로 지정한 네트워크 주소의 노드를 통해 지정된 기기 주소에 송신합니다. 그 응답은 D를 시작 주소로 하여 저장합니다.
CPU 유니트가 자신에 대해 임의의 FINS 명령을 전송하는 것도 가능합니다.
주로 파일 메모리 관련 FINS 명령을 전송하여 파일 메모리를 조작하는 데 사용합니다.
또한 자체 CPU 유니트 대상 FINS 명령을 실행하는 중(파일 메모리 관련 이외의 FINS 명령을 전송한 경우도 포함)에는 파일 메모리 조작 실행 중 플래그(A343.13)가 1(ON)이 되므로 반드시 파일 메모리 조작 실행 중 플래그(A343.13)를 b 접점의 입력 조건으로 해야 합니다.
따라서 자체 CPU 유니트에 대한 CMND 명령, CMND2 명령은 동시에 실행할 수 없습니다.

송신을 시작하면 D2의 실행 완료 플래그를 OFF로 합니다. 응답이 돌아오면 송신 결과의 정상/이상이 D2의 에러 플래그에 반영됩니다. 통신 응답은 D2 + 1에 기록됩니다.
기존 CMND 명령의 통신 가능 플래그(A202) 및 네트워크 통신 실행 에러 플래그(A219)가 피연산자 D2에 해당하고, 네트워크 통신 응답 코드(A203~210)가 D2+1에 해당합니다.

샘플 프로그램

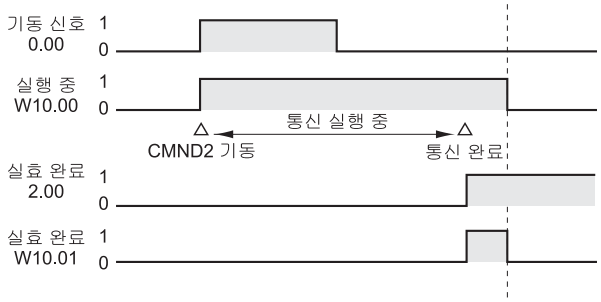
0.00(기동) 및 A202.08(CJ2 전용 네트워크 통신 명령 사용 가능)이 ON이면 W10.00(실행 중)이 ON이 됩니다. W10.00(실행 중)이 ON이고 A343.13(파일 메모리 조작 실행 중 플래그)이 OFF이면 자체 CPU 유니트 대상으로 FINS 명령 #2215(디렉토리 작성/삭제)를 전송하고 그 응답을 D100~D101에 저장합니다.
FINS 명령은 CPU 유니트에 장착된 메모리 카드 내, 예를 들어 '\OMRON' 디렉토리 아래에 'CS1'이라는 디렉토리를 작성합니다. 응답으로 명령 코드 #2215의 2바이트와 종료 코드의 2바이트가 반환됩니다.
정상일 때는 2.00(실행 완료)이 ON되고 정상 처리를 실행합니다. 이상일 때는 2.01(에러 종료)이 ON되고 이상 처리를 실행한 후 종료합니다.



		15 87 0	
S:	D6	2 2 1 5	명령 코드=#2215(디렉토리 작성/삭제)
S+1:	D7	8 0 0 0	디스크 번호=#8000(메모리 카드)
S+2:	D8	0 0 0 0	파라미터=#0000(디렉토리 작성)
S+3:	D9	4 3 5 3	하위 디렉토리명 = 'CS1□□□□□□.□□□' (□: 공백코드)
S+4:	D10	3 1 2 0	
S+5:	D11	2 0 2 0	
S+6:	D12	2 0 2 0	
S+7:	D13	2 E 2 0	
S+8:	D14	2 0 2 0	
S+9:	D15	0 0 0 6	디렉토리 길이=#0006(6문자)
S+10:	D16	5 C 4 F	디렉토리 절대 경로 = '\OMRON'
S+11:	D17	4 D 5 2	
S+12:	D18	4 F 4 E	

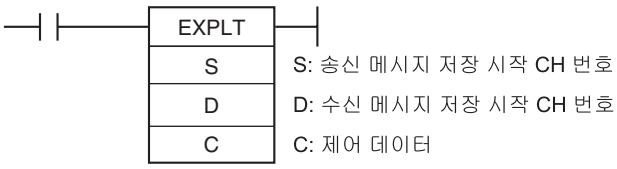
		15 87 0	
C:	D0	0 0 1 A	명령 데이터 바이트 수=#001A(26바이트)
C+1:	D1	0 0 0 4	응답 데이터 바이트 수=#0004(4바이트)
C+2:	D2	0 0 0 0	송신 대상 주소= #0000(자체 네트워크)
C+3:	D3	0 0 0 0	#00(로컬 노드)/#00(CPU 유니트)
C+4:	D4	0 7 0 0	응답 필요/내부 논리 포트 번호 7/재송신 횟수=0회
C+5:	D5	0 0 0 0	응답 감시 시간=#0000(2초)

■ 타임 차트



EXPLT

명령 이름	니모닉	동작 옵션	평션 번호	기능
범용 Explicit 메시지 송신 명령	EXPLT	@EXPLT	720	임의의 ServiceCode의 Explicit 메시지를 송신합니다.

심볼	EXPLT
	

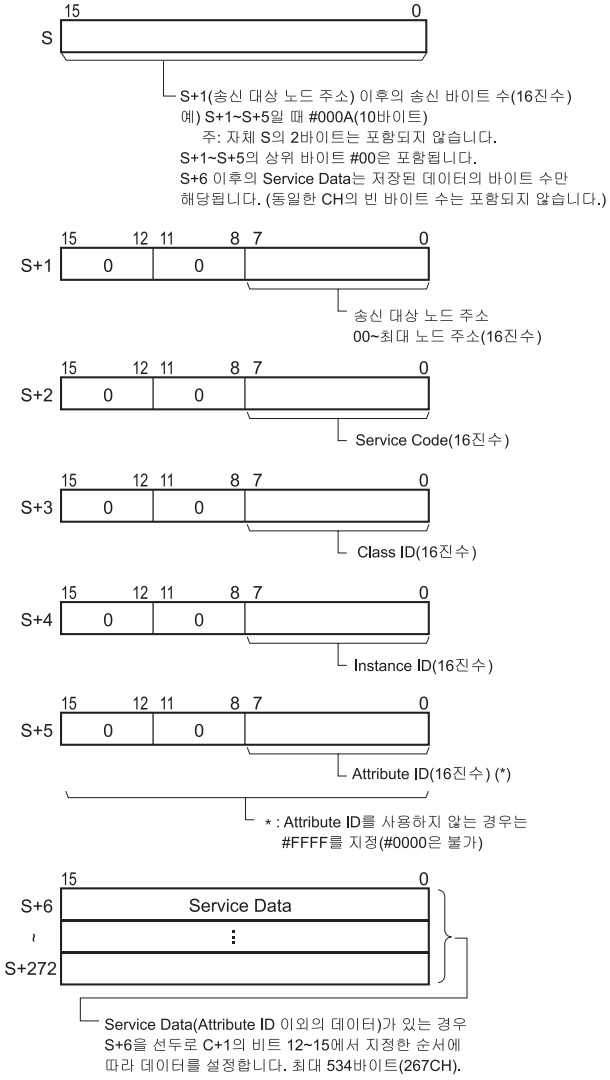
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

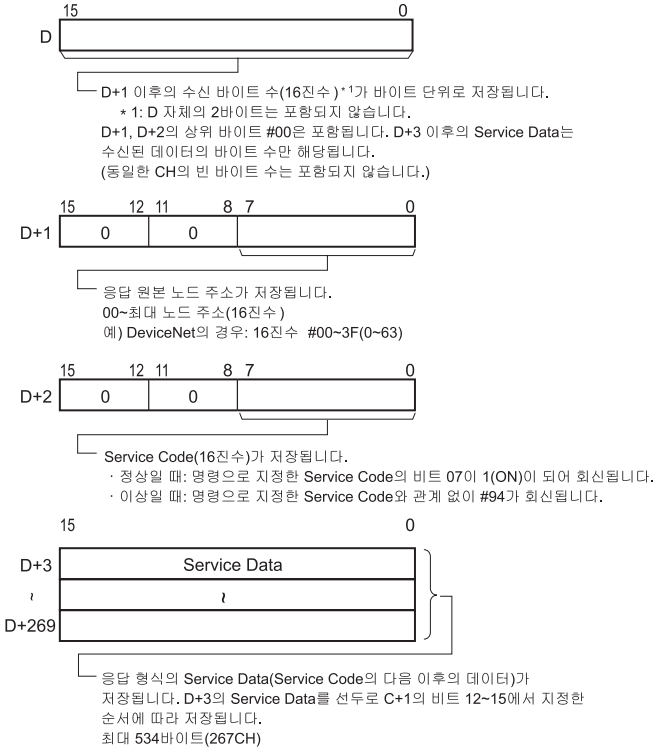
피연산자 설명

피연산자	내용	데이터 형식	크기
S	송신 메시지 저장 시작 CH 번호	WORD	가변
D	수신 메시지 저장 시작 CH 번호	WORD	가변
C	제어 데이터	LWORD	4

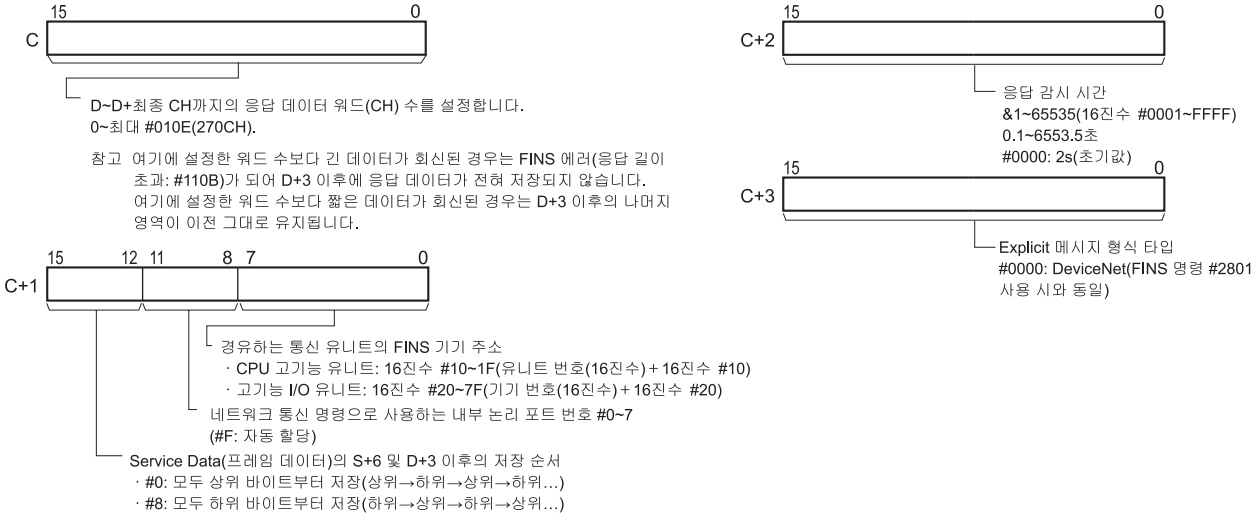
S: 송신 메시지 저장 시작 CH 번호



D: 수신 메시지 저장 시작 CH 번호



C: 제어 데이터



EXPLT

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		
C																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C로 지정한 내부 논리 포트 번호에 대응하는 네트워크 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

'Explicit 통신 실행 에러 플래그'에서 정상 종료(OFF)인지, 이상 종료(ON)인지 판별합니다.

이상 종료(ON)인 경우, 그리고 FINS의 '네트워크 통신 실행 에러 플래그'를 통해 Explicit 메시지 자체가 전송되지 않았는지(ON 시), 아니면 전송되었는데 송신한 Explicit 메시지에 에러가 있었는지(OFF 시) 판별합니다. 또한 '네트워크 통신 응답 코드'에는 정상 종료인 경우 #0000, Explicit의 에러인 경우 Explicit 메시지의 에러 코드(Error Code), FINS의 에러인 경우 FINS 메시지의 종료 코드가 저장됩니다.
동작에 대한 자세한 내용은 네트워크 통신 명령에 대한 전체 설명 앞부분의 'Explicit 메시지 통신 명령'을 참조하십시오.

이름	주소	내용
네트워크 통신 명령 실행 가능 플래그	A202.00~ A202.07	네트워크 통신(PMCR 명령 포함)이 실행 가능할 때 1(ON)이 됩니다. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 네트워크 통신 실행 중에 0(OFF), 실행 종료(정상이든, 이상이든) 후에는 1(ON)이 됩니다.
Explicit 통신 실행 에러 플래그	A213.00~ A213.07	Explicit 메시지 통신 실행 시 에러(이상)가 발생한 경우 1(ON)이 됩니다*1. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 *1: Explicit 메시지 자체가 전송되지 않은 경우, 전송되었지만 Explicit 이상 응답이 발생한 경우에 1(ON)이 됩니다. 다음 Explicit 메시지 통신 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 Explicit 메시지 통신 명령 실행에서 0(OFF)이 됩니다.
네트워크 통신 실행 에러 플래그	A219.00~ A219.07	Explicit 메시지 자체가 전송되지 않은 경우 1(ON)이 됩니다. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 다음 네트워크 통신 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 통신 명령 실행에서 0(OFF)이 됩니다.
네트워크 통신 응답 코드	A203CH~ A210CH	네트워크 통신이 실행된 경우 다음 코드가 저장됩니다. 각 CH는 내부 논리 포트 번호에 대응합니다. A203~210CH: 내부 논리 포트 번호 00~07 · Explicit 통신 실행 에러 플래그가 0(OFF)일 때: #0000이 저장됩니다. · Explicit 통신 실행 에러 플래그가 1(ON)이고 네트워크 통신 실행 에러 플래그가 1(ON)일 때: FINS의 종료 코드가 저장됩니다. · Explicit 통신 실행 에러 플래그가 1(ON)이고 네트워크 통신 실행 에러 플래그가 0(OFF)일 때: Explicit의 에러 코드(Error Code)가 저장됩니다. 통신 실행 중에는 # 0000이 되고 통신 명령 실행이 종료되면 반영됩니다. 운전 시작 시에 클리어됩니다.

기능

S+2~S+최종 CH까지의 범용 Explicit 명령을 C+1 비트 00~07로 지정한 FINS 기기 주소의 통신 유니트를 통해 S+1로 지정한 네트워크상의 노드 주소로 전송합니다.

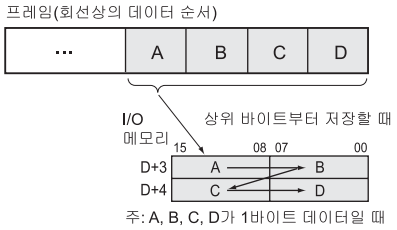
수신한 Explicit 응답은 D+2 이후에 저장합니다.

주 1: S에 저장하는 송신 바이트 수는 S+1~S+5의 10바이트 및 S+6 이후의 Service Data의 데이터 바이트 수입니다. (예: Service Data가 1바이트인 경우, 데이터의 저장 순서에 관계 없이 10+1=11바이트이므로 S에는 &11 또는 #000B를 설정합니다.)

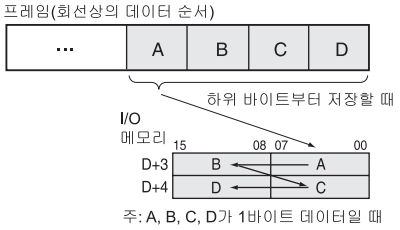
주 2: D에 저장된 수신 바이트 수는 D+1~D+2의 4바이트 및 D+3 이후의 Service Data의 데이터 바이트 수입니다. (예: Service Data가 1바이트인 경우, 데이터의 저장 순서에 관계 없이 4+1=5바이트이므로 D에는 &5 또는 #0005가 저장됩니다.)

또한 Service Data의 S+6 및 D+3 이후의 데이터 저장 순서는 C+1 비트 12~15로 지정합니다.

・ 상위 바이트부터 저장하는 경우: C+1 비트 12~15 = #0으로 합니다.

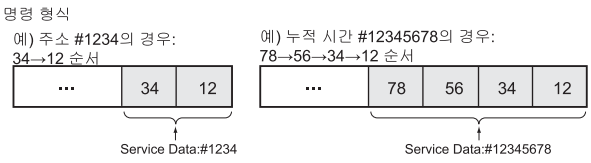


・ 하위 바이트부터 저장하는 경우: C+1 비트 12~15 = #8로 합니다.



사용상의 주의 사항

저장 원본 데이터의 순서는 Explicit 메시지의 프레임상의 순서(회선상의 데이터 순서)라는 점에 주의하십시오.



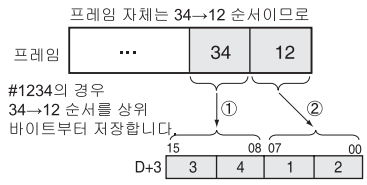
예를 들어, Service Data의 데이터가 2바이트 또는 4바이트 단위인 경우, 프레임상의 순서(회선상의 데이터 순서)는 다음과 같이 2자리씩 하위 자리→상위 자리의 순서가 됩니다.

따라서 Service Data 내 데이터가 2바이트 또는 4바이트 단위인 경우, I/O 메모리에 저장하는 방법은 다음과 같습니다.

1) 2바이트 단위 데이터인 경우

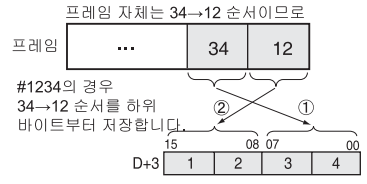
・ 상위 바이트부터 저장할 때(C+1 비트 12~15 = #0)

예) #1234의 데이터를 D+3~에 저장할 때



・ 하위 바이트부터 저장할 때(C+1 비트 12~15 = #8)

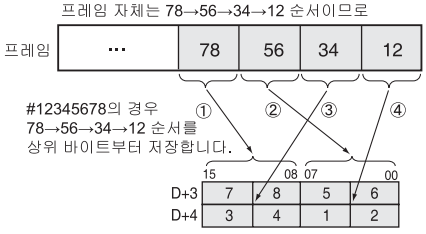
예) #1234의 데이터를 D+3에 저장할 때



2) 4바이트 단위 데이터인 경우

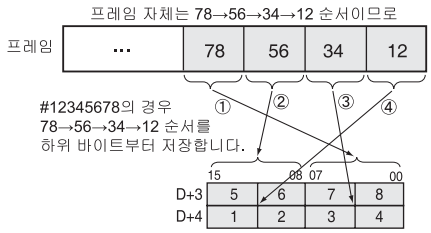
・ 상위 바이트부터 저장할 때(C+1 비트 12~15 = #0)

예) #12345678의 데이터를 D+3~에 저장할 때



EXPLT

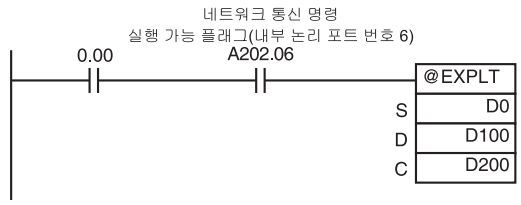
· 하위 바이트부터 저장할 때(C+1 비트 12~15 = #1)
예) #12345678의 데이터를 D+3~에 저장할 때



주: 위의 내용은 D+3 이후의 예이지만 S+6 이후도 동일합니다.

샘플 프로그램

DRT2 슬레이브(원격 I/O 터미널)의 ON 누적 시간 또는 접점 동작 횟수 읽기



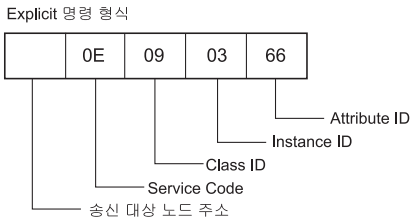
0.00 및 A202.06(내부 논리 포트 번호 6용 네트워크 통신 명령 실행 가능 플래그)이 ON일 때

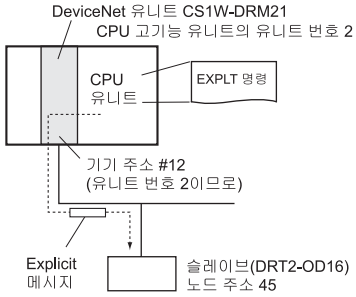
DRT2 슬레이브(원격 I/O 터미널)의 ON 누적 시간 또는 접점 동작 횟수를 읽습니다.

번호 3의 ON 누적 시간(단위: 초) 또는 접점 동작 횟수(단위: 회)를 읽습니다.

Service Code: #0E, Class ID = #09, Instance ID = #03, Attribute ID = #66

예를 들어, 응답으로 ON 누적 시간(단위: 초)이 2752039초인 경우





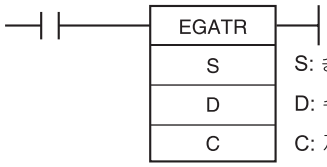
S:	D0	0	0	0	A	송신 크기: S+1~S+5이므로 5CH = #A(10바이트)
S+1:	D1	0	0	2	D	슬레이브의 노드 주소 = #2D(45)
S+2:	D2	0	0	0	E	Service Code = #0E
S+3:	D3	0	0	0	9	Class ID = #09
S+4:	D4	0	0	0	3	Instance ID = #03(입력 번호 3)
S+5:	D5	0	0	6	6	Attribute ID = #66

D:	D100	0	0	0	8	수신 바이트로 응답 프레임상의 8바이트 = #08이 저장되는
D+1:	D101	0	0	2	D	슬레이브의 노드 주소 = #2D(45)가 회신됩니다.
D+2:	D102	0	0	8	E	Service Code = #8E(정상)
D+3:	D103	2	7	F	E	Service Data = #0029FE27(10진수로 2752039초)
D+4:	D104	2	9	0	0	

C:	D200	0	0	0	4	D~D+4CH이므로 5CH = #0005 설정
C+1:	D201	0	6	1	2	데이터 저장 순서 = #0(상위 바이트부터), 사용하는 내부 논리 포트 번호 = #6, DeviceNet 유니트의 기기 주소 = #12(유니트 번호 2)
C+2:	D202	0	0	0	0	응답 감시 시간 = #0000(2초)
C+3:	D203	0	0	0	0	Explicit 형식 타입 지정 = #0000(DeviceNet)

EGATR

명령 이름	니모닉	동작 옵션	평션 번호	기능
Explicit 읽기 명령	EGATR	@EGATR	721	Explicit 메시지를 통해 정보/상태 읽기(Get Attribute single, ServiceCode: 0EHex)를 실행합니다.

심볼	EGATR
	

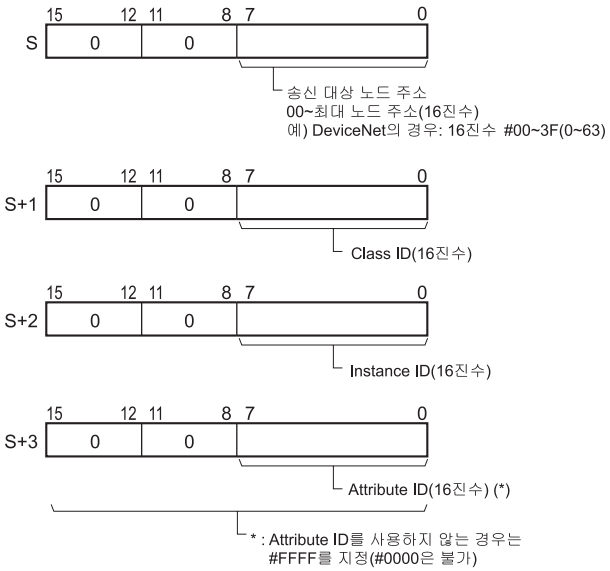
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

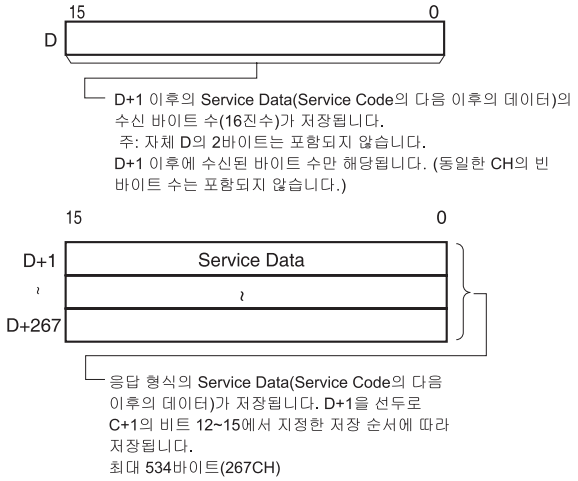
피연산자 설명

피연산자	내용	데이터 형식	크기
S	송신 메시지 저장 시작 CH 번호	ULINT	4
D	수신 메시지 저장 시작 CH 번호	WORD	가변
C	제어 데이터	LWORD	4

S: 송신 메시지 저장 시작 CH 번호

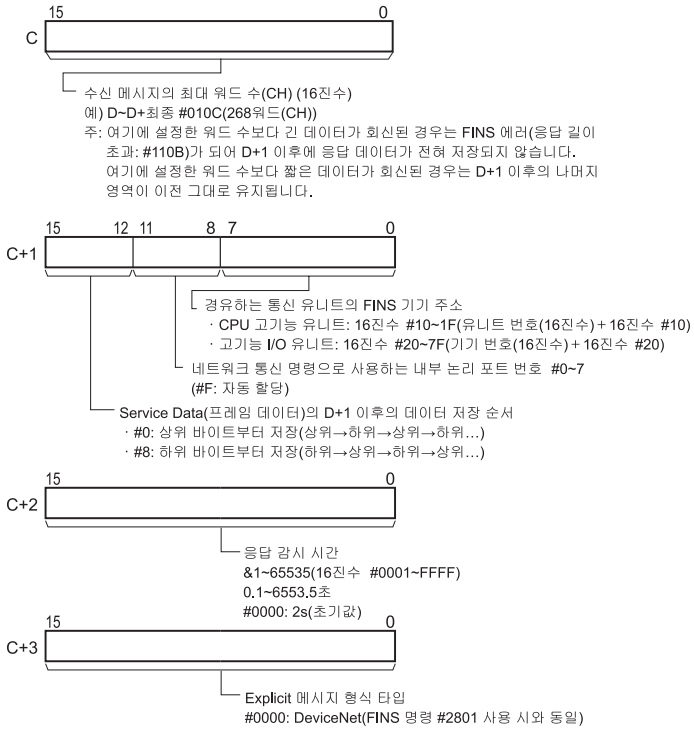


D: 수신 메시지 저장 시작 CH 번호



EGATR

C: 제어 데이터



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		
C																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C로 지정한 내부 논리 포트 번호에 대응하는 네트워크 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

'Explicit 통신 실행 에러 플래그'에서 정상 종료(OFF)인지, 이상 종료(ON)인지 판별합니다.

이상 종료(ON)인 경우, 그리고 FINS의 '네트워크 통신 실행 에러 플래그'를 통해 Explicit 메시지 자체가 전송되지 않았는지(ON 시), 아니면 전송되었는데 송신한 Explicit 메시지에 에러가 있었는지(OFF 시) 판별합니다. 또한 '네트워크 통신 응답 코드'에는 정상 종료인 경우 #0000, Explicit의 에러인 경우 Explicit 메시지의 에러 코드(Error Code), FINS의 에러인 경우 FINS 메시지의 종료 코드가 저장됩니다.
동작에 대한 자세한 내용은 네트워크 통신 명령에 대한 전체 설명 앞부분의 'Explicit 메시지 통신 명령'을 참조하십시오.

이름	주소	내용
네트워크 통신 명령 실행 가능 플래그	A202.00~ A202.07	네트워크 통신(PMCR 명령 포함)이 실행 가능할 때 1(ON)이 됩니다. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 네트워크 통신 실행 중에 0(OFF), 실행 종료(정상이든, 이상이든) 후에는 1(ON)이 됩니다.

이름	주소	내용
Explicit 통신 실행 에러 플래그	A213.00~A213.07	Explicit 메시지 통신 실행 시 에러(이상)가 발생한 경우 1(ON)이 됩니다*1. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 *1: Explicit 메시지 자체가 전송되지 않는 경우, 전송되었지만 Explicit 이상 응답이 발생한 경우에 1(ON)이 됩니다. 다음 Explicit 메시지 통신 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 Explicit 메시지 통신 명령 실행에서 0(OFF)이 됩니다.
네트워크 통신 실행 에러 플래그	A219.00~A219.07	Explicit 메시지 자체가 전송되지 않은 경우 1(ON)이 됩니다. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 다음 네트워크 통신 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 통신 명령 실행에서 0(OFF)이 됩니다.
네트워크 통신 응답 코드	A203CH~A210CH	네트워크 통신이 실행된 경우 다음 코드가 저장됩니다. 각 CH는 내부 논리 포트 번호에 대응합니다. A203~210CH: 내부 논리 포트 번호 00~07 ・ Explicit 통신 실행 에러 플래그가 0(OFF)일 때: #0000이 저장됩니다. ・ Explicit 통신 실행 에러 플래그가 1(ON)이고 네트워크 통신 실행 에러 플래그가 1(ON)일 때: FINS의 종료 코드가 저장됩니다. ・ Explicit 통신 실행 에러 플래그가 1(ON)이고 네트워크 통신 실행 에러 플래그가 0(OFF)일 때: Explicit의 에러 코드(Error Code)가 저장됩니다. 통신 실행 중에는 #0000이 되고 통신 명령 실행이 종료되면 반영됩니다. 운전 시작 시에 클리어됩니다.

기능

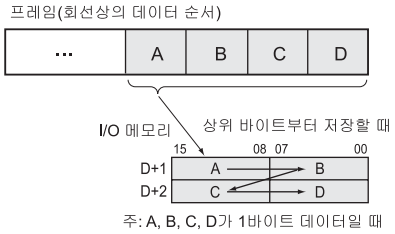
S+1~S+3 내의 정보/상태 읽기(ServiceCode: #0E)의 Explicit 명령을 C+1 비트 00~07로 지정한 FINS 기기 주소의 통신 유니트를 통해 S로 지정한 네트워크상의 노드 주소로 전송합니다.

수신한 Explicit 응답 내의 ServiceData(Service Code의 다음 이후의 데이터)는 D+1 이후에 저장됩니다.

주: D에 저장된 수신 바이트 수는 Service Data의 데이터 바이트 수입니다. (예: Service Data가 1바이트인 경우, 데이터의 저장 순서에 관계 없이 D에는 #0001이 저장됩니다.)

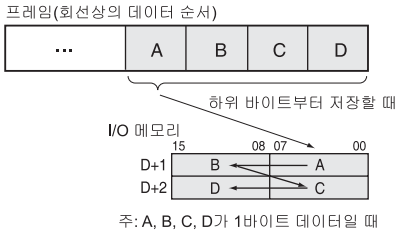
또한 Service Data의 D+1 이후의 데이터 저장 순서는 C+1 비트 12~15로 지정합니다.

・ 상위 바이트부터 저장하는 경우: C+1 비트 12~15 = #0으로 합니다.



주: A, B, C, D가 1바이트 데이터일 때

・ 하위 바이트부터 저장하는 경우: C+1 비트 12~15 = #8로 합니다.

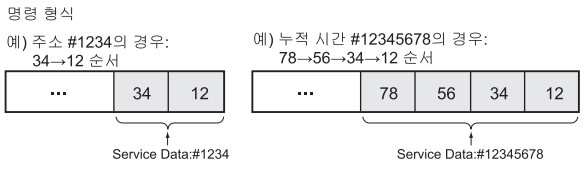


주: A, B, C, D가 1바이트 데이터일 때

사용상의 주의 사항

저장 원본 데이터의 순서는 Explicit 메시지의 프레임상의 순서(회선상의 데이터 순서)라는 점에 주의하십시오.

예를 들어, Service Data의 데이터가 2바이트 또는 4바이트 단위인 경우, 프레임상의 순서(회선상의 데이터 순서)는 다음과 같이 2자리씩 하위 자리→상위 자리의 순서가 됩니다.

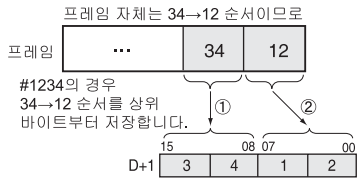


따라서 Service Data 내 데이터가 2바이트 또는 4바이트 단위인 경우, I/O 메모리에 저장하는 방법은 다음과 같습니다.

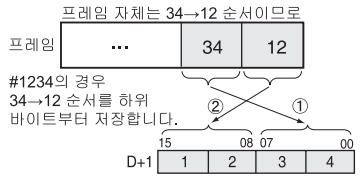
EGATR

1) 2바이트 단위 데이터인 경우

· 상위 바이트부터 저장할 때(C+1 비트 12~15 = #0)
예) #1234의 데이터를 D+1에 저장할 때

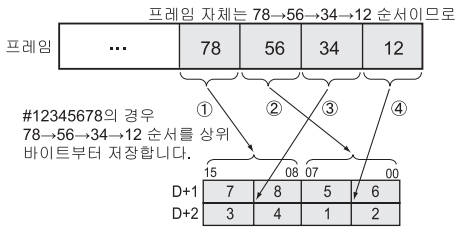


· 하위 바이트부터 저장할 때(C+1 비트 12~15 = #8)
예) #1234의 데이터를 D+1에 저장할 때

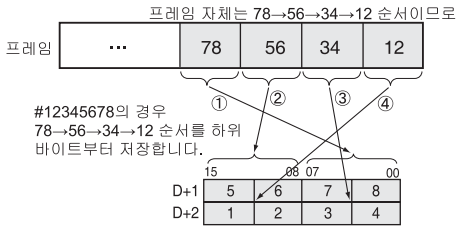


2) 4바이트 단위 데이터인 경우

· 상위 바이트부터 저장할 때(C+1 비트 12~15 = #0)
예) #12345678의 데이터를 D+1~에 저장할 때

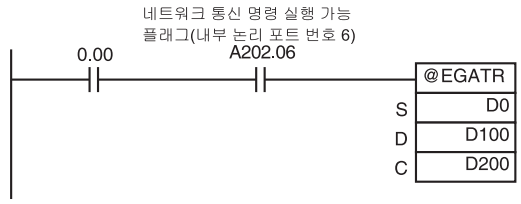


· 하위 바이트부터 저장할 때(C+1 비트 12~15 = #1)
예) #12345678의 데이터를 D+1~에 저장할 때



샘플 프로그램

DRT2 슬레이브(원격 I/O 터미널)의 범용 상태 읽기

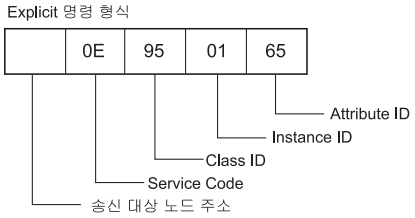


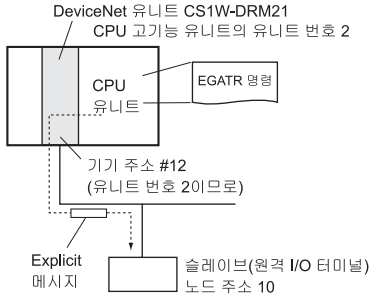
0.00 및 A202.06(내부 논리 포트 번호 6용 네트워크 통신 명령 실행 가능 플래그)이 ON일 때

DRT2 슬레이브(원격 I/O 터미널)의 범용 상태를 읽습니다.

(Service Code: #0E), Class ID = #95, Instance ID = #01, Attribute ID = #65

범용 상태는 1바이트로 회신됩니다.

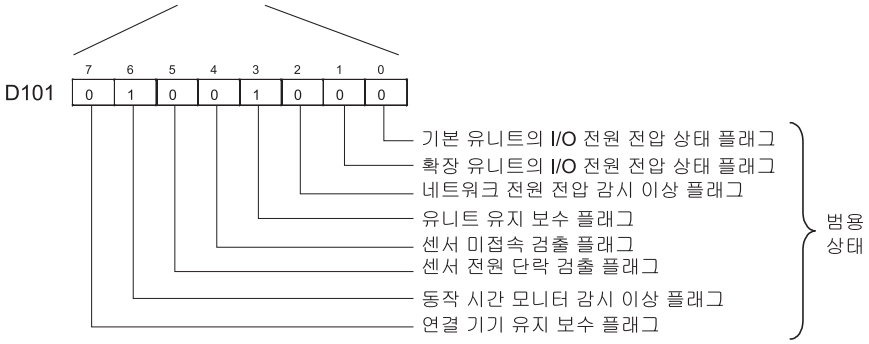




S:	D0	0	0	0	A	슬레이브의 노드 주소 = #A(10)
S+1:	D1	0	0	9	5	Class ID = #95
S+2:	D2	0	0	0	1	Instance ID = #01
S+3:	D3	0	0	6	5	Attribute ID = #65

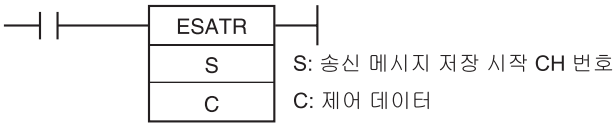
C:	D200	0	0	0	2	D~D+1CH 2워드 = #0002 설정
C+1:	D201	8	6	1	2	데이터 저장 순서 = #8(하위 바이트부터), 사용하는 내부 논리 포트 번호 = #6, DeviceNet 유니트의 기기 주소 = #12(유니트 번호 2)
C+2:	D202	0	0	0	0	응답 감시 시간 = #0000(2초)
C+3:	D203	0	0	0	0	Explicit 형식 타입 지정 = #0000(DeviceNet)

D:	D100	0	0	0	1	D+1의 하위 바이트뿐이므로 1바이트 = #1이 회신됩니다. 범용 상태가 비트 00~07에 회신됩니다. C+1 비트의 비트 12~15에서 하위 바이트 지정이므로 비트 00~07에 저장됩니다.
D+1:	D101	0	0	4	8	



ESATR

명령 이름	니모닉	동작 옵션	평션 번호	기능
Explicit 쓰기 명령	ESATR	@ESATR	722	Explicit 메시지를 통해 정보 쓰기(Set Attribute single, ServiceCode: 10Hex)를 실행합니다.

심볼	ESATR
	

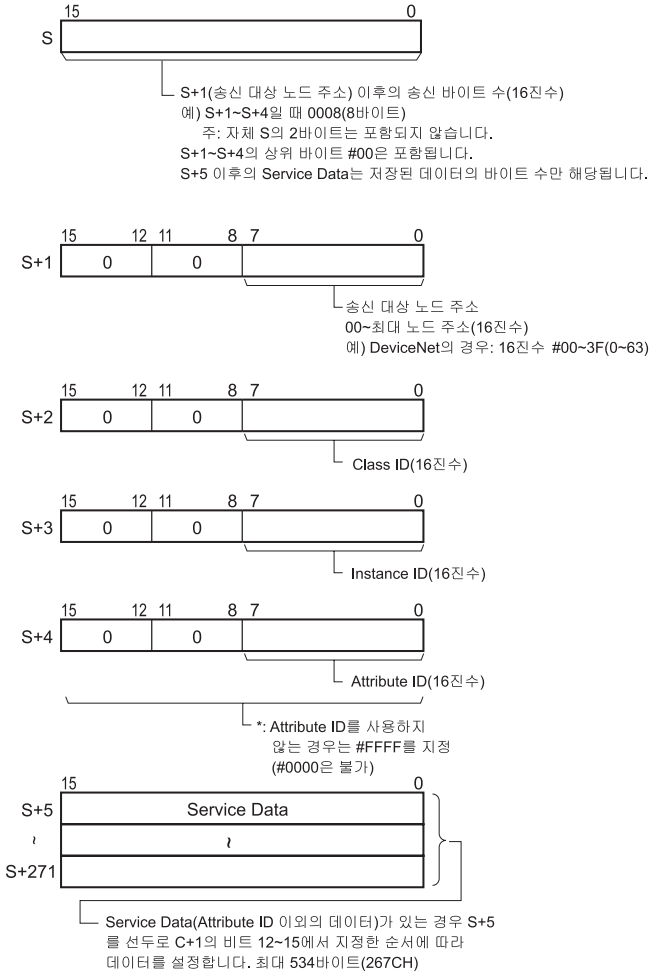
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

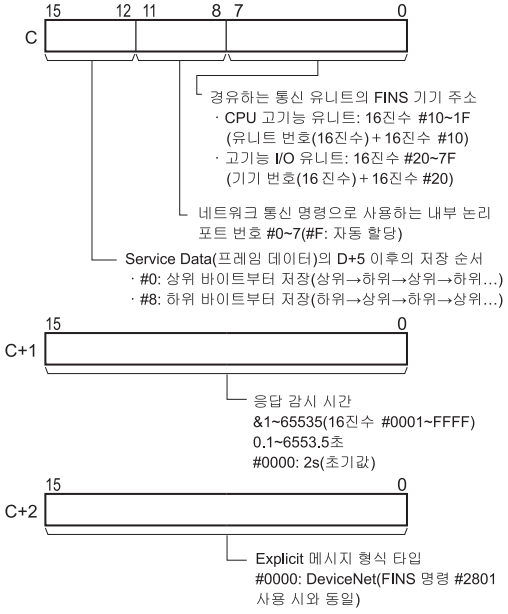
피연산자 설명

피연산자	내용	데이터 형식	크기
S	송신 메시지 저장 시작 CH 번호	WORD	가변
C	제어 데이터	WORD	3

S: 송신 메시지 저장 시작 CH 번호



C: 제어 데이터



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
C																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C로 지정한 내부 논리 포트 번호에 대응하는 네트워크 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

'Explicit 통신 실행 에러 플래그'에서 정상 종료(OFF)인지, 이상 종료(ON)인지 판별합니다.

이상 종료(ON)인 경우, 그리고 FINS의 '네트워크 통신 실행 에러 플래그'를 통해 Explicit 메시지 자체가 전송되지 않았는지(ON 시), 아니면 전송되었는데 송신한 Explicit 메시지에 에러가 있었는지(OFF 시) 판별합니다. 또한 '네트워크 통신 응답 코드'에는 정상 종료인 경우 #0000, Explicit의 에러인 경우 Explicit 메시지의 에러 코드(Error Code), FINS의 에러인 경우 FINS 메시지의 종료 코드가 저장됩니다.
동작에 대한 자세한 내용은 네트워크 통신 명령에 대한 전체 설명 앞부분의 'Explicit 메시지 통신 명령'을 참조하십시오.

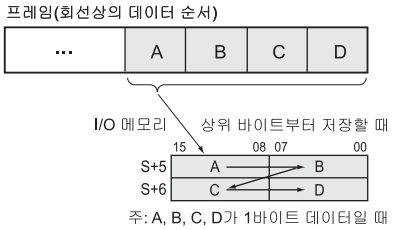
이름	주소	내용
네트워크 통신 명령 실행 가능 플래그	A202.00~ A202.07	네트워크 통신(PMCR 명령 포함)이 실행 가능할 때 1(ON)이 됩니다. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 네트워크 통신 실행 중에 0(OFF), 실행 종료(정상이든, 이상이든) 후에는 1(ON)이 됩니다.
Explicit 통신 실행 에러 플래그	A213.00~ A213.07	Explicit 메시지 통신 실행 시 에러(이상)가 발생한 경우 1(ON)이 됩니다*1. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 *1: Explicit 메시지 자체가 전송되지 않는 경우, 전송되었지만 Explicit 이상 응답이 발생한 경우에 1(ON)이 됩니다. 다음 Explicit 메시지 통신 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 Explicit 메시지 통신 명령 실행에서 0(OFF)이 됩니다.
네트워크 통신 실행 에러 플래그	A219.00~ A219.07	Explicit 메시지 자체가 전송되지 않은 경우 1(ON)이 됩니다. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 다음 네트워크 통신 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 통신 명령 실행에서 0(OFF)이 됩니다.
네트워크 통신 응답 코드	A203CH~ A210CH	네트워크 통신이 실행된 경우 다음 코드가 저장됩니다. 각 CH는 내부 논리 포트 번호에 대응합니다. A203~210CH: 내부 논리 포트 번호 00~07 · Explicit 통신 실행 에러 플래그가 0(OFF)일 때: #0000이 저장됩니다. · Explicit 통신 실행 에러 플래그가 1(ON)이고 네트워크 통신 실행 에러 플래그가 1(ON)일 때: FINS의 종료 코드가 저장됩니다. · Explicit 통신 실행 에러 플래그가 1(ON)이고 네트워크 통신 실행 에러 플래그가 0(OFF)일 때: Explicit의 에러 코드(Error Code)가 저장됩니다. 통신 실행 중에는 #0000이 되고 통신 명령 실행이 종료되면 반영됩니다. 운전 시작 시에 클리어됩니다.

기능

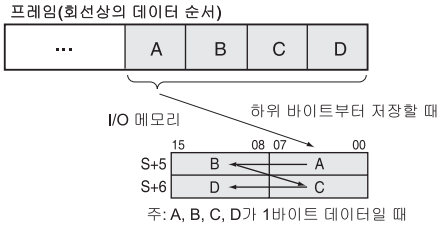
S+2~S+최종 CH 내의 정보 쓰기(ServiceCode: #10)의 Explicit 명령을 C 비트 00~07로 지정한 FINS 기기 주소의 통신 유니트를 통해 S+1로 지정한 네트워크상의 노드 주소로 전송합니다.

또한 Service Data의 S+5 이후의 데이터 저장 순서는 C 비트 12~15로 지정합니다.

· 상위 바이트부터 저장하는 경우: C 비트 12~15 = #0으로 합니다.

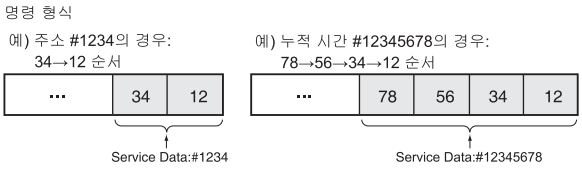


· 하위 바이트부터 저장하는 경우: C 비트 12~15 = #8로 합니다.



사용상의 주의 사항

저장 원본 데이터의 순서는 Explicit 메시지의 프레임상의 순서(회선상의 데이터 순서)라는 점에 주의하십시오.

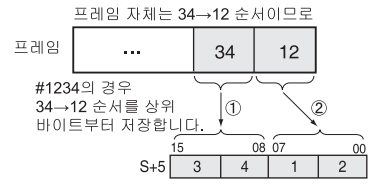


예를 들어, Service Data의 데이터가 2바이트 또는 4바이트 단위인 경우, 프레임상의 순서(회선상의 데이터 순서)는 다음과 같이 2자리씩 하위 자리→상위 자리의 순서가 됩니다.

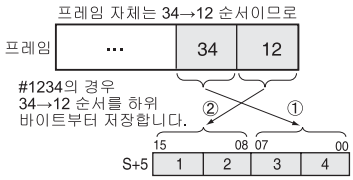
따라서 Service Data 내 데이터가 2바이트 또는 4바이트 단위인 경우, I/O 메모리에 저장하는 방법은 다음과 같습니다.

1) 2바이트 단위 데이터인 경우

· 상위 바이트부터 저장할 때(C 비트 12~15 = #0)
예) #1234의 데이터를 S+5에 저장할 때

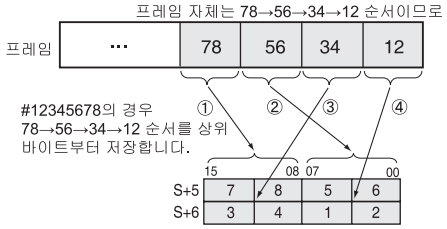


· 하위 바이트부터 저장할 때(C 비트 12~15 = #8)
예) #1234의 데이터를 S+5에 저장할 때

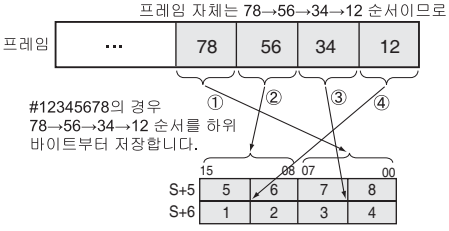


2) 4바이트 단위 데이터인 경우

· 상위 바이트부터 저장할 때(C 비트 12~15 = #8)
예) #12345678의 데이터를 S+5~에 저장할 때

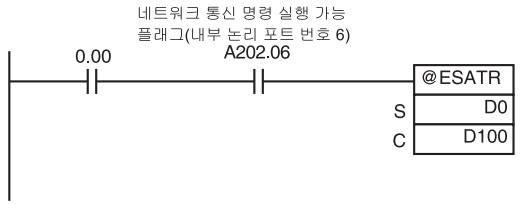


· 하위 바이트부터 저장할 때(C+1 비트 12~15 = #1)
예) #12345678의 데이터를 S+5~에 저장할 때



샘플 프로그램

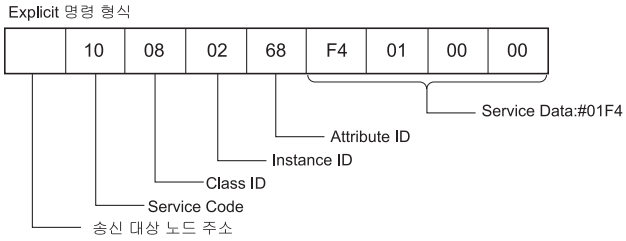
DRT2 슬레이브(원격 I/O 터미널)의 접점 동작 횟수 설정값 쓰기

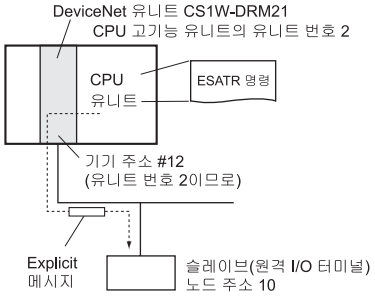


0.00 및 A202.06(내부 논리 포트 번호 6용 네트워크 통신 명령 실행 가능 플래그)이 ON일 때 DRT2 슬레이브(원격 I/O 터미널)의 입력 번호 2의 접점 동작 횟수 설정값을 씁니다.

(Service Code: #10,) Class ID = #08, Instance ID = #02, Attribute ID = #68

접점 동작 횟수 설정값은 500(#000001F4)으로 합니다. 따라서, Service Data = #F4010000



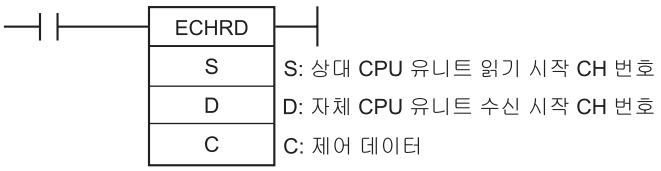


S:	D0	0	0	0	C	S+1~S+6이므로 6CH=#C(12바이트)
S:+1	D1	0	0	0	A	슬레이브의 노드 주소=#A(10)
S+2:	D2	0	0	0	8	Class ID=#08
S+3:	D3	0	0	0	2	Instance ID=#02
S+4:	D4	0	0	6	8	Attribute ID=#68
S+5:	D5	0	1	F	4	Service Data=#F401
S+6:	D6	0	0	0	0	

C:	D201	8	6	1	2	데이터 저장 순서=#8(하위 바이트부터) 사용하는 내부 논리 포트 번호=#6
C+1:	D202	0	0	0	0	DeviceNet 유니트의 기기 주소=#12(유니트 번호 2)
C+2:	D203	0	0	0	0	응답 감시 시간=#0000(2초) Explicit 형식 타입 지정 = #0000(DeviceNet)

ECHRD

명령 이름	니모닉	동작 옵션	평션 번호	기능
Explicit CPU 유니트 데이터 읽기 명령	ECHRD	@ECHRD	723	Explicit 메시지를 지원하는 네트워크상의 CPU 유니트 데이터를 자체 CPU 유니트로 읽어옵니다.

심볼	ECHRD
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

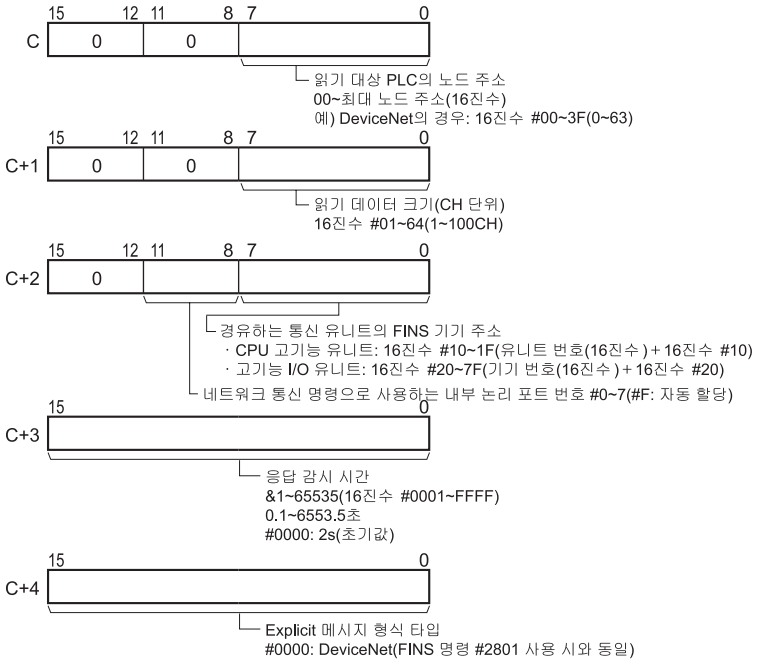
피연산자 설명

피연산자	내용	데이터 형식	크기
S	상대 CPU 유니트 읽기 시작 CH 번호	UINT	1
D	자체 CPU 유니트 수신 시작 CH 번호	UINT	1
C	제어 데이터	WORD	5

S: 상대 CPU 유니트 읽기 시작 CH 번호
S에 상대 CPU 유니트에서 읽으려는 시작 채널 번호를 지정합니다.

D: 자체 CPU 유니트 수신 시작 CH 번호
D에 읽은 데이터를 저장할 자체 CPU 유니트의 시작 채널 번호를 지정합니다.

C: 제어 데이터



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	* ○	○	○	–	–	–	○	–	–	–	–
D																		
C																		

* CJ2 CPU 유니트만 EM D 뱅크를 지정할 수 있습니다. CS1/CJ1은 상대 CPU 유니트가 CJ2 CPU 유니트일 때도 D 뱅크 이후는 지정할 수 없습니다.

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C로 지정한 내부 논리 포트 번호에 대응하는 네트워크 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

'Explicit 통신 실행 에러 플래그'에서 정상 종료(OFF)인지, 이상 종료(ON)인지 판별합니다.

이상 종료(ON)인 경우, 그리고 FINS의 '네트워크 통신 실행 에러 플래그'를 통해 Explicit 메시지 자체가 전송되지 않았는지(ON 시), 아니면 전송되었는데 송신한 Explicit 메시지에 에러가 있었는지(OFF 시) 판별합니다. 또한 '네트워크 통신 응답 코드'에는 정상 종료인 경우 #0000, Explicit의 에러인 경우 Explicit 메시지의 에러 코드(Error Code), FINS의 에러인 경우 FINS 메시지의 종료 코드가 저장됩니다.

동작에 대한 자세한 내용은 네트워크 통신 명령에 대한 전체 설명 앞부분의 'Explicit 메시지 통신 명령'을 참조하십시오.

이름	주소	내용
네트워크 통신 명령 실행 가능 플래그	A202.00~ A202.07	네트워크 통신(PMCR 명령 포함)이 실행 가능할 때 1(ON)이 됩니다. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 네트워크 통신 실행 중에 0(OFF), 실행 종료(정상이든, 이상이든) 후에는 1(ON)이 됩니다.
Explicit 통신 실행 에러 플래그	A213.00~ A213.07	Explicit 메시지 통신 실행 시 에러(이상)가 발생한 경우 1(ON)이 됩니다*1. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 *1: Explicit 메시지 자체가 전송되지 않는 경우, 전송되었지만 Explicit 이상 응답이 발생한 경우에 1(ON)이 됩니다. 다음 Explicit 메시지 통신 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 Explicit 메시지 통신 명령 실행에서 0(OFF)이 됩니다.
네트워크 통신 실행 에러 플래그	A219.00~ A219.07	Explicit 메시지 자체가 전송되지 않은 경우 1(ON)이 됩니다. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 다음 네트워크 통신 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 통신 명령 실행에서 0(OFF)이 됩니다.
네트워크 통신 응답 코드	A203CH~ A210CH	네트워크 통신이 실행된 경우 다음 코드가 저장됩니다. 각 CH는 내부 논리 포트 번호에 대응합니다. A203~210CH: 내부 논리 포트 번호 00~07 · Explicit 통신 실행 에러 플래그가 0(OFF)일 때: #0000이 저장됩니다. · Explicit 통신 실행 에러 플래그가 1(ON)이고 네트워크 통신 실행 에러 플래그가 1(ON)일 때: FINS의 종료 코드가 저장됩니다. · Explicit 통신 실행 에러 플래그가 1(ON)이고 네트워크 통신 실행 에러 플래그가 0(OFF)일 때: Explicit의 에러 코드(Error Code)가 저장됩니다. 통신 실행 중에는 #0000이 되고 통신 명령 실행이 종료되면 반영됩니다. 운전 시작 시에 클리어됩니다.

기능

네트워크에서 C로 지정한 노드 주소의 CPU 유니트의 S로 지정한 읽기 시작 CH에서 C+1로 지정한 송신 CH 수 분량의 데이터를 읽어 자체 CPU 유니트의 D 이후에 저장합니다.

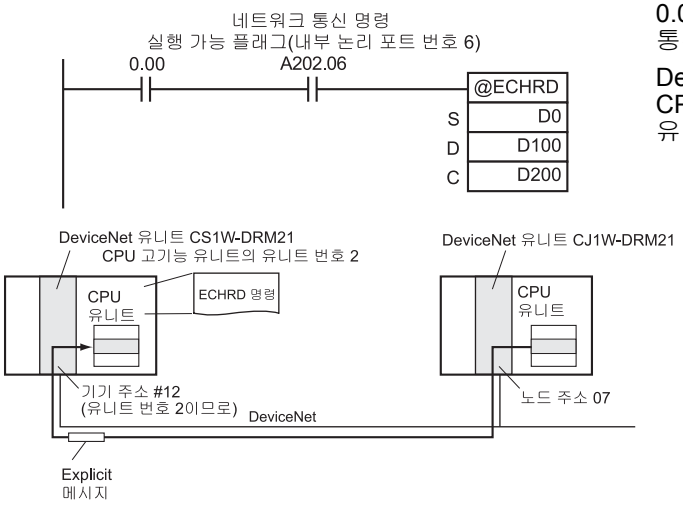
ECHRD

힌트

· 이 명령은 Service 코드가 #1C(바이트 단위의 데이터 읽기: PLC 바이트 read)의 Explicit 메시지를 송신합니다.

샘플 프로그램

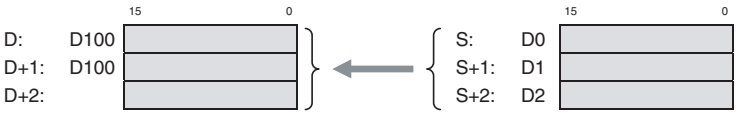
DeviceNet 네트워크상의 CJ 시리즈 CPU 유니트의 I/O 메모리를 읽어 자체 CPU 유니트의 I/O 메모리에 저장합니다.



0.00 및 A202.06(내부 논리 포트 번호 6용 네트워크 통신 명령 실행 가능 플래그)이 ON일 때

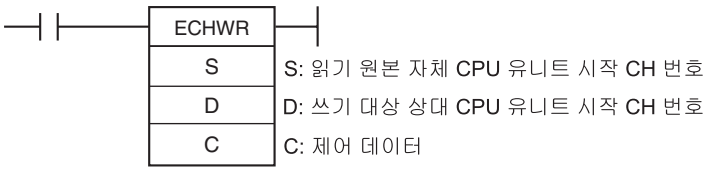
DeviceNet 네트워크상의 노드 주소 07의 CJ 시리즈 CPU 유니트의 I/O 메모리 D0~D2를 읽어 자체 CPU 유니트의 D100~D102에 저장합니다.

	15 8	7 0	
C: D200	00	07	읽기 대상 PLC의 노드 주소 = #7
C+1: D201	00	03	읽기 데이터 크기(CH 수) = #3
C+2: D202	06	12	사용하는 내부 논리 포트 번호 = 6 DeviceNet 유니트의 기기 주소 = #12 (유니트 번호 2)
C+3: D203	00	00	응답 감시 시간 = #0000(2초)
C+4: D204	00	00	Explicit 형식 타입 지정 = #0000 (DeviceNet)



ECHWR

명령 이름	니모닉	동작 옵션	평션 번호	기능
Explicit CPU 유니트 데이터 쓰기 명령	ECHWR	@ECHWR	724	자체 CPU 유니트에서 Explicit 메시지를 지원하는 네트워크상의 CPU 유니트에 데이터를 씁니다.

심볼	ECHWR
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

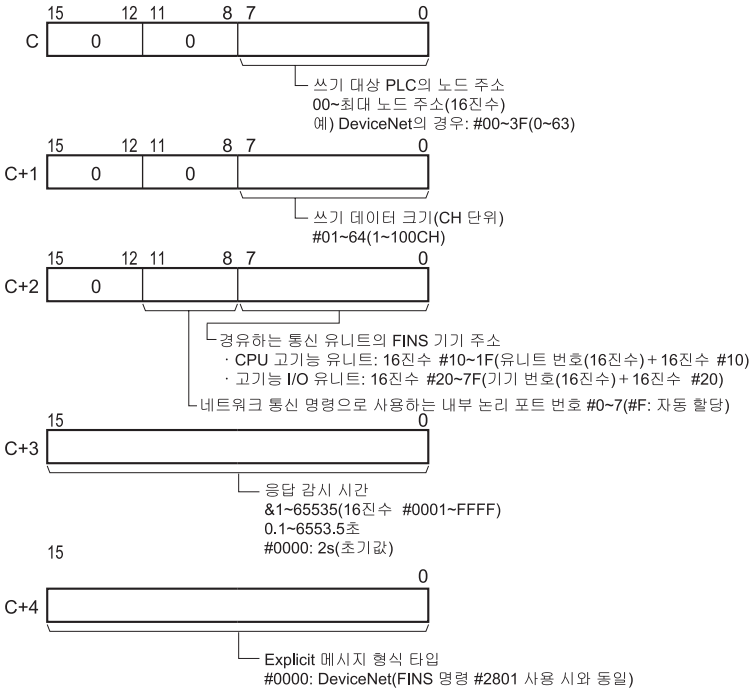
피연산자 설명

피연산자	내용	데이터 형식	크기
S	쓰기 원본 자체 CPU 유니트 시작 CH 번호	UINT	1
D	쓰기 대상 상대 CPU 유니트 시작 CH 번호	UINT	1
C	제어 데이터	WORD	5

S: 쓰기 원본 자체 CPU 유니트 시작 CH 번호
S에 쓰기 원본의 자체 CPU 유니트의 시작 채널 번호를 지정합니다.

D: 쓰기 대상 상대 CPU 유니트 시작 CH 번호
D에 쓰기 대상의 상대 CPU 유니트의 시작 채널 번호를 지정합니다.

C: 제어 데이터



ECHWR

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	* ○	○	○	–	–	–	○	–	–	–	–
D																		
C																		

* CJ2 CPU 유니트만 EM D 뱅크 이후를 지정할 수 있습니다. CS1/CJ1은 상대 CPU 유니트가 CJ2 CPU 유니트일 때도 D 뱅크 이후를 지정할 수 없습니다.

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C로 지정한 내부 논리 포트 번호에 대응하는 네트워크 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

'Explicit 통신 실행 에러 플래그'에서 정상 종료(OFF)인지, 이상 종료(ON)인지 판별합니다.

이상 종료(ON)인 경우, 그리고 FINS의 '네트워크 통신 실행 에러 플래그'를 통해 Explicit 메시지 자체가 전송되지 않았는지(ON 시), 아니면 전송되었는데 송신한 Explicit 메시지에 에러가 있었는지(OFF 시) 판별합니다. 또한 '네트워크 통신 응답 코드'에는 정상 종료인 경우 #0000, Explicit의 에러인 경우 Explicit 메시지의 에러 코드(Error Code), FINS의 에러인 경우 FINS 메시지의 종료 코드가 저장됩니다.
동작에 대한 자세한 내용은 네트워크 통신 명령에 대한 전체 설명 앞부분의 'Explicit 메시지 통신 명령'을 참조하십시오.

이름	주소	내용
네트워크 통신 명령 실행 가능 플래그	A202.00~ A202.07	네트워크 통신(PMCR 명령 포함)이 실행 가능할 때 1(ON)이 됩니다. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 네트워크 통신 실행 중에 0(OFF), 실행 종료(정상이든, 이상이든) 후에는 1(ON)이 됩니다.
Explicit 통신 실행 에러 플래그	A213.00~ A213.07	Explicit 메시지 통신 실행 시 에러(이상)가 발생한 경우 1(ON)이 됩니다*1. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 *1: Explicit 메시지 자체가 전송되지 않는 경우, 전송되었지만 Explicit 이상 응답이 발생한 경우에 1(ON)이 됩니다. 다음 Explicit 메시지 통신 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 Explicit 메시지 통신 명령 실행에서 0(OFF)이 됩니다.
네트워크 통신 실행 에러 플래그	A219.00~ A219.07	Explicit 메시지 자체가 전송되지 않은 경우 1(ON)이 됩니다. 각 비트는 내부 논리 포트 번호에 대응합니다. 비트 00~07: 내부 논리 포트 번호 00~07 다음 네트워크 통신 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 통신 명령 실행에서 0(OFF)이 됩니다.
네트워크 통신 응답 코드	A203CH~ A210CH	네트워크 통신이 실행된 경우 다음 코드가 저장됩니다. 각 CH는 내부 논리 포트 번호에 대응합니다. A203~210CH: 내부 논리 포트 번호 00~07 · Explicit 통신 실행 에러 플래그가 0(OFF)일 때: #0000이 저장됩니다. · Explicit 통신 실행 에러 플래그가 1(ON)이고 네트워크 통신 실행 에러 플래그가 1(ON)일 때: FINS의 종료 코드가 저장됩니다. · Explicit 통신 실행 에러 플래그가 1(ON)이고 네트워크 통신 실행 에러 플래그가 0(OFF)일 때: Explicit의 에러 코드(Error Code)가 저장됩니다. 통신 실행 중에는 #0000이 되고 통신 명령 실행이 종료되면 반영됩니다. 운전 시작 시에 클리어됩니다.

기능

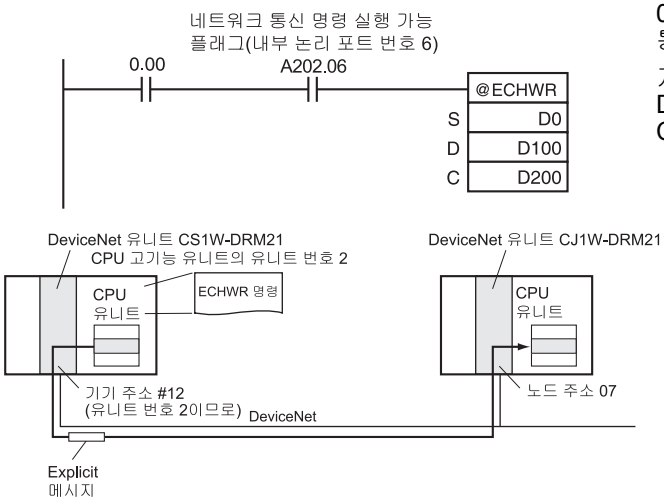
자체 CPU 유니트의 S로 지정한 쓰기 원본 시작 CH에서 C+1로 지정한 CH 수 분량의 데이터를 네트워크상의 C로 지정한 노드 주소의 CPU 유니트의 D로 지정한 쓰기 선두 이후에 씁니다.

힌트

· 이 명령은 Service 코드가 #1E(바이트 단위의 데이터 쓰기: PLC 바이트 write)의 Explicit 메시지를 송신합니다.

샘플 프로그램

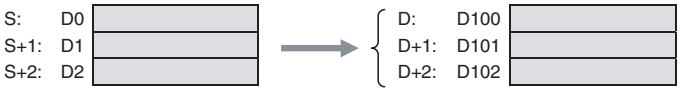
자체 CPU 유니트의 I/O 메모리를 DeviceNet 네트워크상의 CJ 시리즈 CPU 유니트의 I/O 메모리에 씁니다.



0.00 및 A202.06(내부 논리 포트 번호 6용 네트워크 통신 명령 실행 가능 플래그)이 ON일 때

자체 CPU 유니트의 I/O 메모리 D0~D2를 읽어 DeviceNet 네트워크상의 노드 주소 07의 CJ 시리즈 CPU 유니트 D100~D102에 저장합니다.

	15 8	7 0	
C: D200	0 0	0 7	읽기 대상 PLC의 노드 주소 = #7
C+1: D201	0 0	0 3	읽기 데이터 크기(CH 수) = #3
C+2: D202	0 6	1 2	사용하는 내부 논리 포트 번호 = 6 DeviceNet 유니트의 기기 주소 = #12 (유니트 번호 2)
C+3: D203	0 0	0 0	응답 감시 시간 = #0000(2초)
C+4: D204	0 0	0 0	Explicit 형식 타입 지정 = #0000 (DeviceNet)



파일 메모리용 명령

니모닉	명령 이름	펑션 번호	페이지
FREAD	데이터 파일 읽기	700	3-831
FWRIT	데이터 파일 쓰기	701	3-836
TWRIT	텍스트 파일 쓰기	704	3-841

사용자 프로그램의 명령에 의한 파일 조작

■ 개요

사용자 프로그램에서 다음과 같은 명령을 사용하여 파일을 조작할 수 있습니다.

· FWRIT/TWRIT 명령:

I/O 메모리의 지정된 데이터를 메모리 카드 또는 EM 파일 메모리에 1개 파일(데이터 파일)로 작성할 수 있습니다.

또한 이미 있는 파일에 대해 추가하거나 임의의 위치에서 덮어 쓸 수도 있습니다.

· FREAD 명령:

메모리 카드 또는 EM 파일 메모리에 존재하는 데이터 파일을 읽어 지정된 I/O 메모리에 반영할 수 있습니다.

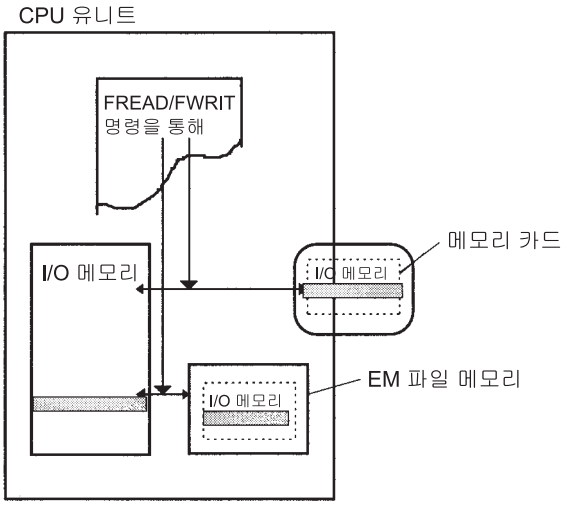
또한 파일에 대해 임의의 위치에서 읽을 수 있습니다.

주: 이 경우, 파일 단위의 전송(읽기/쓰기)이 아니라 파일 내의 지정된 시작 위치에서 지정된 수의 데이터를 읽고 씁니다.

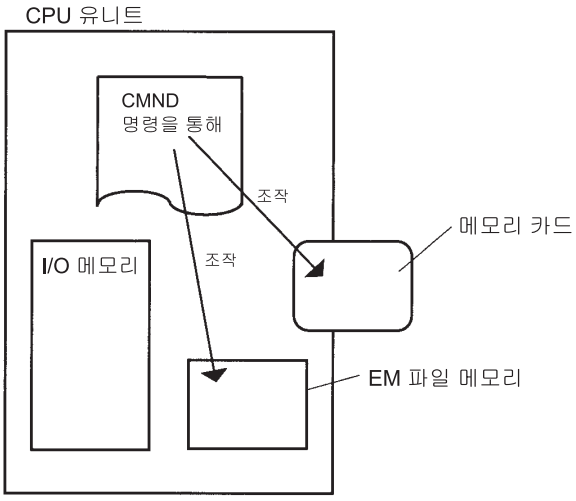
· CMND 명령:

CPU 유니트가 자신에 대해 메모리 카드 조작 관련 FINS 명령을 전송하면 메모리 카드 또는 EM 파일 메모리에 대해 파일 포맷, 삭제, 복사, 파일명 변경 등의 각종 조작을 할 수 있습니다.

I/O 메모리와 파일 메모리 간의 읽기/쓰기
: FREAD/FWRIT 명령으로 실행



파일 메모리에 대한 조작
: CMND 명령으로 실행



■ 파일 메모리용 명령

● 명령

명령 이름	명령어	내용
데이터 파일 읽기	FREAD 명령	지정된 데이터 파일의 데이터 또는 데이터의 요소 수를 지정된 I/O 메모리에 읽습니다.
데이터 파일 쓰기	FWRIT 명령	지정된 I/O 메모리 영역의 데이터를 지정된 데이터 파일로 작성합니다.
텍스트 파일 쓰기	TWRIT 명령	지정된 I/O 메모리 영역에 있는 ASCII 문자열을 텍스트 파일로 작성합니다.

데이터 파일의 데이터 형식은 BIN 데이터(.IOM), 구분이 없거나 탭으로 구분된 텍스트(.TXT), 쉼표 구분 텍스트(.CSV)를 FREAD/FWRIT 명령의 피연산자를 통해 선택할 수 있습니다. 구체적으로 다음과 같습니다.

● FREAD/FWRIT 명령의 피연산자

제어 데이터	내용	입력 가능 범위
비트 12~15	데이터 형식([ ] 안은 확장자명) 0Hex: BIN[.IOM] 1Hex: 구분 없는 워드[.TXT] 2Hex: 구분 없는 LONG 워드[.TXT] 3Hex: 쉼표 구분 워드[.CSV] 4Hex: 쉼표 구분 LONG 워드[.CSV] 5Hex: 탭 구분 워드[.TXT] 6Hex: 탭 구분 LONG 워드[.TXT]	0~6Hex

제어 데이터	내용	입력 가능 범위
비트 08~11	행바꿈 유무/행바꿈 위치 0Hex: 행바꿈 없음 8Hex: 10필드마다 행바꿈 9Hex: 1필드마다 행바꿈 AHex: 2필드마다 행바꿈 BHex: 4필드마다 행바꿈 CHex: 5필드마다 행바꿈 DHex: 16필드마다 행바꿈	0Hex, 8~DHex

힌트

파일 메모리용 명령(쓰기)을 통해 파일 메모리에 작성되는 파일의 날짜는 CPU 유니트에 내장된 시계를 따릅니다.

이 FREAD/FWRIT 명령은 다른 FREAD/FWRIT 명령, 자체 CPU 유니트 대상 CMND 명령, 특수 보조 릴레이에 의한 프로그램 교체, 간이 백업 조작과 동시에 실행할 수 없습니다. 따라서 프로그램에서 아래의 '파일 메모리 조작 실행 중 플래그'를 사용하여 배타적으로 제어해야 합니다.

또한 FREAD 명령 실행 시, 도중에 에러가 되는 데이터(16진수값 이외의 데이터 또는 4자리/8자리 이외의 데이터)가 있는 경우, 읽기는 도중에 종료되지만 그 이전의 데이터는 읽습니다. 이때, '파일 읽기 이상 플래그'(A343.10)가 1(ON)이 됩니다. (에러 플래그(P_ER)는 ON이 되지 않습니다.)

● 관련 특수 보조 릴레이

이름	주소	내용
메모리 카드 종류	A343.00~343.02	장착된 메모리 카드의 종류
EM 파일 메모리화 영역의 포맷 이상 플래그	A343.06	EM 파일 메모리화 시작 뱅크 번호의 영역이 포맷 이상일 때 1(ON)이 됩니다. 주: 포맷이 정상적으로 종료되면 0(OFF)으로 되돌아갑니다.
메모리 카드 포맷 이상 플래그	A343.07	메모리 카드 장착 시, 메모리 카드가 포맷되지 않았을 때 또는 포맷 이상일 때 1(ON)이 됩니다.
파일 쓰기 이상 플래그	A343.08	파일 쓰기에서 이상이 발생했을 때 1(ON)이 됩니다.
파일 쓰기 불가 플래그	A343.09	파일 쓰기 실행 시, 쓰기 방지되어 있거나 용량 초과 때문에 쓰기 작업을 수행할 수 없는 경우 1(ON)이 됩니다.
파일 읽기 이상 플래그	A343.10	파일을 읽을 때 이상이 발생한 경우(파일이 손상되거나 파일을 읽을 때 잘못된 데이터가 있는 경우) 1(ON)
파일 없음 플래그	A343.11	파일을 읽을 때 읽어올 파일이 존재하지 않는 경우 또는 파일을 쓸 때 지정한 디렉토리가 존재하지 않는 경우 1(ON)
파일 메모리 조작 실행 중 플래그	A343.13	다음 중 하나의 파일 메모리 관련 조작 실행 중에 1(ON). 아무 것도 실행하지 않을 때는 0(OFF) · 메모리 카드 인식 시작 시 · 자체 CPU 유니트 대상 CMND 명령 실행 중 · FREAD/FWRIT 명령 실행 · 특수 보조 릴레이에 의한 프로그램 교체 중 · 간이 백업 조작 실행 중
파일 액세스 중 플래그	A343.14	파일 데이터를 액세스하는 중에 1(ON)이 됩니다.
메모리 카드 인식 완료 플래그	A343.15	메모리 카드를 인식할 수 있으면 1(ON)이 됩니다. 메모리 카드를 인식할 수 없으면 0(OFF)이 됩니다.
남은 데이터 파일 전송 데이터 수	A346~347CH	데이터 파일 전송 상황을 알리기 위한 2워드 채널(32비트)입니다. BIN 형식(.IOM)의 파일인 경우, 1CH를 읽을 때마다 감산됩니다. TXT 형식(.TXT) 및 CSV 형식(.CSV)의 파일인 경우, 1필드(구분 요소)를 읽을 때마다 감산됩니다.

■ CMND 명령

● 명령

명령 이름	명령어	내용
명령 송신	CMND 명령	CPU 유니트가 자신에 대해 파일 메모리 관련 FINS 명령을 전송하면 자신의 파일 메모리에 대해 포맷, 삭제 등의 각종 조작을 할 수 있습니다.

주: 제어 데이터(C)는 다음과 같이 설정합니다.
송신 대상 네트워크 주소: 00Hex(자체 네트워크), 시리얼 포트 번호: 0Hex(사용 안 함), 송신 대상 기기 주소: 00Hex(CPU 유니트), 송신 대상 노드 주소: 00Hex(로컬 노드 내), 재송신 횟수: 0Hex(재송신 횟수 지정은 무효이므로 0회로 설정)

● 파일 메모리 관련 FINS 명령

주: 파일 메모리 관련 이외의 FINS 명령도 전송 가능합니다.
FINS 명령에 대한 자세한 내용은 'SYSMAC CS/CJ 시리즈 통신 명령 참조 설명서'(SBCA-304)를 참조하십시오.

이 자체 CPU 유니트 대상 CMND 명령은 다른 자체 CPU 유니트 대상 CMND 명령, 파일 메모리용 명령, 특수 보조 릴레이에 의한 프로그램 교체, 간이 백업 조작과 동시에 실행할 수 없습니다. 따라서 프로그램에서 아래의 '파일 메모리 조작 실행 중 플래그'를 사용하여 배타적으로 제어해야 합니다.

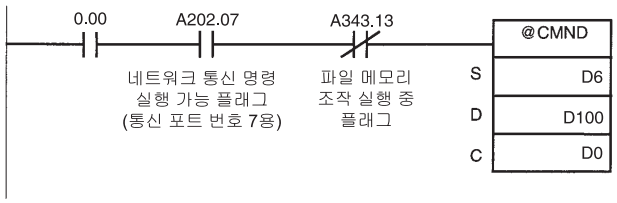
또한 자체 CPU 유니트 대상 CMND 명령이 실행되지 못한 경우 ER 플래그가 ON이 됩니다.

● 관련 특수 보조 릴레이

이름	주소	내용
파일 메모리 조작 실행 중 플래그	A343.13	다음 중 하나의 파일 메모리 관련 조작 실행 중에 1(ON). 아무 것도 실행하지 않을 때는 0(OFF) · 메모리 카드 인식 시작 시 · 자체 CPU 유니트 대상 CMND 명령 실행 중 · FREAD/FWRIT 명령 실행 · 특수 보조 릴레이에 의한 프로그램 교체 중 · 간이 백업 조작 실행 중
메모리 카드 인식 완료 플래그	A343.15	메모리 카드를 인식하면 1(ON)이 됩니다. 메모리 카드를 인식할 수 없으면 0(OFF)이 됩니다.

● 프로그램 예

(예) 메모리 카드에 디렉토리 작성



0.00 및 A202.07(통신 포트 번호 7용 네트워크 통신 명령 실행 가능 플래그)이 ON이고 A343.13 (파일 메모리 조작 실행 중 플래그)이 OFF이면 자체 CPU 유니트 대상으로 FINS 명령 2215Hex (디렉토리 작성/삭제)를 전송하고 그 응답을 D100~D101에 저장합니다. FINS 명령은 CPU 유니트에 장착된 메모리 카드 내, 예를 들어 '\OMRON' 디렉토리 아래에 'CS1'이라는 디렉토리를 작성합니다. 응답으로 명령 코드 2215Hex의 2바이트와 종료 코드의 2바이트가 반환됩니다.

		15 8	7 0	
S:	D6	22	15	명령 코드=2215 Hex(디렉토리 작성/삭제)
S+1:	D7	80	00	디스크 번호=8000 Hex(메모리 카드)
S+2:	D8	00	00	파라미터=0000 Hex(디렉토리 작성)
S+3:	D9	43	53	하위 디렉토리명='CS1□□□□□.□□□' (□: 공백 코드)
S+4:	D10	31	20	
S+5:	D11	20	20	
S+6:	D12	20	20	
S+7:	D13	2E	20	
S+8:	D14	20	20	
S+9:	D15	00	06	디렉토리 길이=0006 Hex(6문자)
S+10:	D16	5C	4F	디렉토리 절대 경로='₩OMRON'
S+11:	D17	4D	52	
S+12:	D18	4F	4E	

		15 8	7 0	
C:	D0	00	1A	명령 데이터 바이트 수=001A Hex(26바이트)
C+1:	D1	00	04	응답 데이터 바이트 수=0004 Hex(4바이트)
C+2:	D2	00	00	송신 대상 주소=0000 Hex(자체 네트워크)
C+3:	D3	00	00	00 Hex(로컬 노드)/00 Hex(CPU 유니트)
C+4:	D4	07	00	응답 필요/통신 포트 번호 7/재송신 횟수=0회
C+5:	D5	00	00	응답 감시 시간=0000 Hex(6553.5초)

힌트

자체 CPU 유니트 대상 CMND 명령은 위의 파일 메모리 관련 FINS 명령 이외의 다른 FINS 명령을 자신에 대해 전송할 수도 있습니다. 단, 그 경우도 위의 파일 메모리 조작 실행 중 플래그를 통해 배타적으로 제어하여 사용해야 합니다.

■ 순서

● 메모리 카드 내 데이터 파일 읽기/쓰기 또는 파일을 조작할 때

1. CPU 유니트에 메모리 카드를 장착(포맷 완료된 카드를 사용)
2. 데이터 파일 쓰기(FWRIT) 명령으로 I/O 메모리의 지정된 영역에 파일명을 붙인 후 메모리 카드에 저장

주: 메모리 카드에 기록한 TXT 또는 CSV 형식의 데이터를 표 계산 소프트웨어로 읽는 경우, PC의 카드용 슬롯에 메모리 카드 어댑터 HMC AP001을 통해 메모리 카드를 장착한 후 Windows 기능을 사용하여 읽을 수 있습니다.

데이터 파일 읽기 명령으로 메모리 카드에서 데이터 파일을 CPU 유니트 내의 I/O 메모리로 읽기

자체 CPU 유니트 대상 명령 송신(CMND) 명령을 통해 메모리 카드 내의 파일 조작

● EM 파일 메모리 내의 데이터 파일 읽기/쓰기 또는 파일을 조작할 때

1. PLC 시스템 설정을 통해 EM 영역의 지정된 뱅크 이후를 파일 메모리로 전환
2. CX-Programmer로 EM 파일 메모리를 초기화
3. 데이터 파일 쓰기 명령으로 I/O 메모리의 지정된 영역에 파일명을 붙여 EM 파일 메모리에 저장

데이터 파일 읽기 명령으로 EM 파일 메모리에서 데이터 파일을 CPU 유니트 내의 I/O 메모리로 읽기

자체 CPU 유니트 대상 명령 송신(CMND) 명령을 통해 EM 파일 메모리 내의 파일 조작

메모리 카드를 사용하는 경우 주의 사항

메모리 카드를 사용하기 전에 다음 사항을 확인하십시오.

1) 포맷

메모리 카드는 포맷(초기화)하여 출하되므로 구입한 후 포맷할 필요가 없습니다.

사용한 후 포맷하는 경우는 반드시 CX-Programmer 또는 프로그래밍 콘솔을 사용하여 CPU 유니트에서 포맷하십시오.

노트북 컴퓨터 등으로 메모리 카드를 직접 포맷한 경우, CPU 유니트가 메모리 카드를 인식하지 못할 수 있습니다. 이런 경우에는 다시 CPU 유니트를 사용하여 메모리 카드를 포맷해도 사용할 수 없으므로 주의하십시오.

2) 루트 디렉토리에 저장할 수 있는 파일 수

하드 디스크와 동일하게 메모리 카드의 루트 디렉토리에 저장할 수 있는 파일 수에는 제한이 있습니다. 메모리 카드의 종류/형식에 따라 다르지만 128~512개까지 저장할 수 있습니다.

따라서 시간별로 로그 파일을 작성하는 응용 프로그램인 경우 루트 디렉토리가 아닌 하위 디렉토리에 파일을 저장하십시오.

하위 디렉토리는 미리 PC에 작성하거나 CMND 명령을 이용하여 작성하십시오.

CMND 명령을 사용한 구체적인 프로그램에 대해서는 이 설명서의 '3-763 명령 송신 CMND(490)'를 참조하십시오.

3) 쓰기 횟수

일반적으로 플래시 메모리에는 쓰기 횟수의 제한이 있습니다.

메모리 카드도 10만회의 보증 제한이 있습니다.

가령 10분 간격으로 끊임없이 쓰기를 실시하면 2년 이내에 10만 회에 도달하게 됩니다.

4) 1개 파일의 최소 크기

DM 영역의 수 CH 분량의 데이터와 같이 작은 파일을 여러 개 저장하면 카드 용량의 상한까지 저장할 수 없습니다.

예를 들어, 할당 단위 크기가 4096바이트인 메모리 카드인 경우 내부 구조 관계로 한 파일의 데이터량이 아무리 작아도 4096바이트 크기를 사용하게 됩니다.

따라서 DM 영역의 10CH 분량의 데이터를 메모리 카드에 저장하면 파일 크기는 68바이트이지만 할당 단위 크기가 4096바이트인 메모리 카드에서는 68바이트의 파일이 실질적으로 4096바이트 용량을 사용합니다.

이와 같이 한 파일의 크기가 작은 경우는 카드의 사용 효율이 나빠집니다.

단, 할당 단위 크기가 크면 카드 사용 효율은 나빠지지만 카드의 액세스 속도는 좋아집니다.

또한 사용 중인 메모리 카드의 할당 단위 크기는 PC의 DOS 프롬프트에서 메모리 카드에 대해 CHKDSK를 실행하여 확인할 수 있습니다.

이 설명서에는 구체적인 방법이 기재되어 있지 않습니다.

할당 단위 크기에 대한 자세한 내용은 PC 관련 일반 서적을 참조하십시오.

5) 메모리 카드 액세스 중의 주의 사항

PLC 본체가 메모리 카드에 액세스 중이면 BUSY LED(메모리 카드 액세스 중 LED)가 점등합니다.

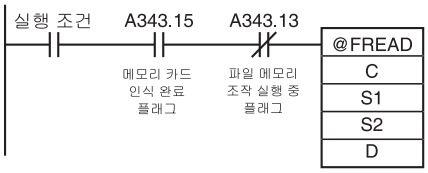
① BUSY LED 점등 중에는 PLC 본체의 전원을 끄지 마십시오.

② BUSY LED가 점등되어 있는 상태에서 그대로 메모리 카드를 빼지 마십시오. 메모리 카드의 전원 공급 정지 푸시 버튼을 누르고 BUSY LED(메모리 카드 액세스 중 LED)가 꺼진 후 메모리 카드를 빼십시오.

위의 ① 또는 ②를 실행하면 최악의 경우, 해당 메모리 카드를 사용할 수 없게 됩니다.

③ 메모리 카드는 카드의 라벨 면을 우측으로 해서 삽입하십시오. 잘못된 방향으로 무리하여 삽입하면 PLC 본체 및 메모리 카드가 고장날 가능성이 있습니다.

④ 메모리 카드를 장착한 후 CPU 유니트가 인식을 완료할 때까지는 몇 초 정도 걸립니다. 따라서 전원 ON 시 또는 메모리 카드 장착 후 즉시 메모리 카드에 액세스하는 경우는 다음과 같이 메모리 카드 인식 완료 플래그(A343.15)를 a 접점의 입력 조건으로 하십시오.



명령의 종류

■ FWRIT 명령

I/O 메모리의 지정된 데이터를 데이터 파일로 작성할 수 있습니다.

파일 형식은 바이너리 파일과 CSV 파일이 있습니다.

또한 이미 있는 파일에 대해 추가하거나 임의의 위치에서 덮어쓸 수도 있습니다.

■ FREAD 명령

데이터 파일을 읽어 지정된 I/O 메모리에 반영할 수 있습니다.

파일 형식은 바이너리 파일과 CSV 파일이 있습니다.

또한 파일에 대해 임의의 위치에서 읽을 수 있습니다.

■ TWRIT 명령

I/O 메모리에 있는 ASCII 문자 데이터를 텍스트 파일로 작성할 수 있습니다.

또한 이미 있는 파일에 대해 추가하거나 덮어쓸 수도 있습니다.

■ CMND 명령

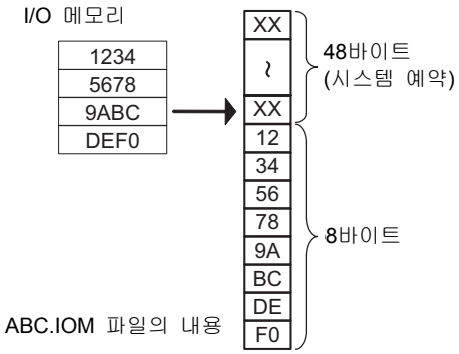
메모리 카드를 조작하는 FINS 명령을 전송하면 파일 포맷, 삭제, 복사, 파일명 변경 등 각종 조작을 할 수 있습니다.

데이터 파일 내의 데이터 구조

■ BIN 형식(.IOM)의 경우

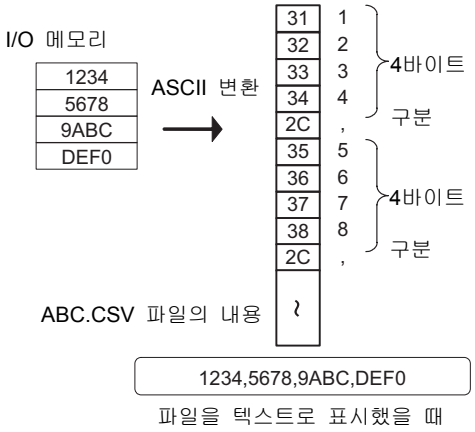
예: I/O 메모리의 4채널 데이터가 #1234, #5678, #9ABC, #DEF0을 ABC.IOM 파일에 저장한 경우

단, 평상시 조작에서는 사용자가 이 데이터 형식을 고려할 필요가 없습니다.



■ 워드 CSV 형식(.CSV)의 경우

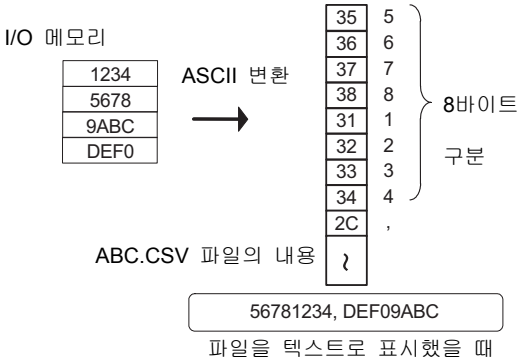
예: I/O 메모리의 4채널 데이터가 #1234, #5678, #9ABC, #DEF0을 ABC.CSV 파일에 저장한 경우



워드 TXT 형식(.TXT)의 경우도 동일합니다.

■ LONG 워드 CSV 형식(.CSV)의 경우

예: I/O 메모리의 4채널 데이터가 #1234, #5678, #9ABC, #DEF0을 ABC.CSV 파일에 저장한 경우



LONG 워드 TXT 형식(.TXT)의 경우도 동일합니다.

관련 특수 보조 릴레이

■ 메모리 카드 인식

이름	주소	내용
메모리 카드 종류	A343.00~ A343.02	장착된 메모리 카드의 종류가 저장됩니다(0: 없음, 4: 플래시 ROM).
메모리 카드 포맷 이상 플래그	A343.07	메모리 카드 장착 시, 메모리 카드가 포맷되지 않았을 때 또는 포맷 이상일 때 1(ON)이 됩니다.
메모리 카드 인식 완료 플래그	A343.15	메모리 카드를 인식할 수 있으면 1(ON)이 됩니다. 메모리 카드를 인식할 수 없으면 0(OFF)이 됩니다.

■ 명령 관련

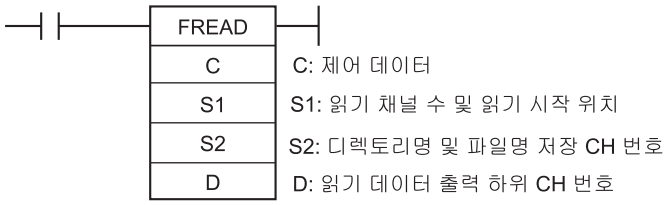
이름	주소	내용
파일 쓰기 이상 플래그	A343.08	파일 쓰기에서 이상이 발생했을 때 1(ON)이 됩니다. 쓰기 파일이 쓰기 방지(읽기 전용)되어 있을 때도1(ON)이 됩니다.
파일 쓰기 불가 플래그	A343.09	파일 쓰기 실행 시, 용량 초과 때문에 쓰기 작업을 수행할 수 없는 경우 1(ON)이 됩니다.
파일 읽기 이상 플래그	A343.10	파일을 읽을 때 해당 파일의 데이터 형식이 잘못되었거나 파일이 손상되어 읽을 수 없는 경우 1(ON)이 됩니다.
파일 없음 플래그	A343.11	파일을 읽을 때 읽어올 파일이 존재하지 않는 경우 또는 지정한 파일이 존재하지 않는 경우 1(ON)이 됩니다.
파일 메모리 조작 실행 중 플래그	A343.13	다음 중 하나의 파일 메모리 관련 조작 실행 중에 1(ON). 아무 것도 실행하지 않을 때는 0(OFF). · 자체 CPU 유니트 대상 CMND 명령 실행 중 · FREAD/FWRIT/TWRIT 명령 실행 · 특수 보조 릴레이에 의한 프로그램 교체 중 · 간이 백업 조작 실행 중
파일 액세스 중 플래그	A343.14	파일 데이터를 실제 액세스하는 중에 1(ON)이 됩니다.
남은 데이터 파일 전송 데이터 수	A346~A347	데이터 파일 전송 상황을 알리기 위한 2워드 채널(32비트)입니다. 데이터 파일 읽기(FREAD) 명령 또는 데이터 파일 쓰기(FWRIT) 명령의 실행 시작 시, 명령 피연산자를 통해 읽기 또는 쓰기 채널 수 또는 필드 수가 저장됩니다. 실제로 전송되면 1 채널 또는 1필드 전송될 때마다 감산됩니다. A346CH: 하위, A347CH: 상위의 32비트 BIN 값입니다.

■ EM 파일 메모리 관련

이름	주소	내용
EM 파일 메모리화 영역 포맷 이상 플래그	A343.06	EM 파일 메모리화 시작 뱅크 번호의 영역이 포맷 이상일 때 1(ON)이 됩니다.
EM 파일 포맷 시작 뱅크 번호	A344	EM 파일 영역으로 포맷된 시작 뱅크 번호가 저장됩니다. 포맷되지 않은 경우는 #FFFF가 저장됩니다.

FREAD

명령 이름	니모닉	동작 옵션	펑션 번호	기능
데이터 파일 읽기	FREAD	@FREAD	700	파일 메모리 내의 지정 데이터 파일(I/O 메모리 파일)의 지정 데이터 또는 데이터 수를 CPU 유니트 내의 지정된 I/O 메모리 영역에 읽어옵니다.

심볼	FREAD
	

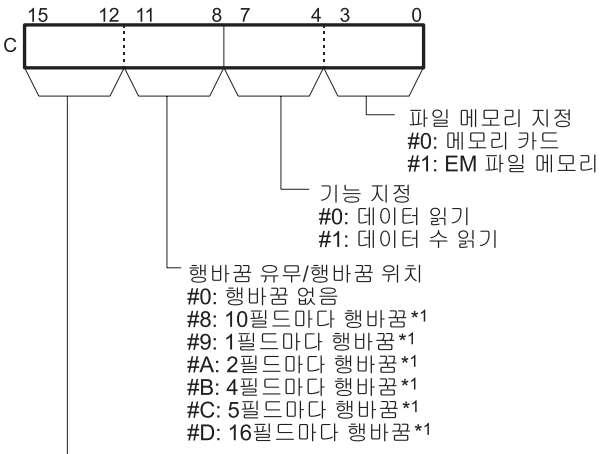
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	UINT	가변
S1	읽기 채널 수 및 읽기 시작 위치	LWORD	4
S2	디렉토리명 및 파일명 저장 CH 번호	UINT	가변
D	읽기 데이터 출력 하위 CH 번호	UINT	가변

C: 제어 데이터



데이터 형식	확장자	1필드(구분 요소)당 I/O 메모리상의 크기
#0: BIN	.IOM	-
#1: 구분 없는 워드*1	.TXT	1CH
#2: 구분 없는 LONG 워드*1	.TXT	2CH
#3: 쉼표 구분 워드*1	.CSV	1CH
#4: 쉼표 구분 LONG 워드*1	.CSV	2CH
#5: 탭 구분 워드*1	.TXT	1CH
#6: 탭 구분 LONG 워드*1	.TXT	2CH

*1: CS 시리즈 CS1 CPU 유니트(-V□ 없음)에서는 사용할 수 없습니다.

주 1: 워드 형식의 경우: 1필드 = I/O 메모리의 1CH, LONG 워드 형식의 경우: 1필드 = I/O 메모리의 2CH

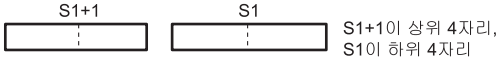
주 2: 행이 구분되는 데이터를 읽는 경우는 행바꿈 있음(C의 비트 08~11이 #8~D 중 하나)을 지정해야 합니다.

FREAD

주 3: LONG 워드 형식의 경우, I/O 메모리의 상위→하위 CH 순서로 저장됩니다(예: #12345678을 D00000: #5678, D00001: #1234로 저장).

S1: 읽기 채널 수 및 읽기 시작 위치

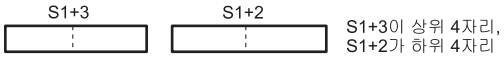
· S1+1, S1: 읽기 데이터 수



S1+1이 상위 4자리, S1이 하위 4자리

데이터 형식	C의 비트 12~15	S1+1, S1의 내용
BIN 형식의 경우	#0(BIN)	읽기 채널 수(몇 CH분을 파일 메모리에서 읽을지 지정) 10진수 &0~1073741823 또는 16진수 #00000000~3FFFFFFF
워드 형식의 경우	#1(구분 없음) #3(쉼표 구분) #5(탭 구분)	읽기 필드 수(1필드 = 1CH이므로 CH 수와 일치) 10진수 &0~536870911 또는 16진수 #00000000~1FFFFFFF
LONG 워드 형식의 경우	#2(구분 없음) #4(쉼표 구분) #6(탭 구분)	읽기 필드 수(1필드 = 2CH이므로 CH 수/2와 일치) 10진수 &0~268435455 또는 16진수 #00000000~0FFFFFFF

· S1+3, S1+2: 읽기 시작 위치
파일 메모리 선두부터 읽기 시작 위치 지정



S1+3이 상위 4자리, S1+2가 하위 4자리

형식	C의 비트 12~15	S1+3, S1+2의 내용
BIN 형식의 경우	#0(BIN)	선두에서 몇 CH 더한 위치에서 읽을지 지정 10진수 &0~1073741823 또는 16진수 #00000000~3FFFFFFF
워드 형식의 경우	#1(구분 없음) #3(쉼표 구분) #5(탭 구분)	선두에서 몇 필드 더한 위치에서 읽을지 지정(1필드 = 1CH이므로 CH를 더한 경우와 일치) 10진수 &0~536870911 또는 16진수 #00000000~1FFFFFFF
LONG 워드 형식의 경우	#2(구분 없음) #4(쉼표 구분) #6(탭 구분)	선두에서 몇 필드 더한 위치에서 읽을지 지정(1필드 = 2CH이므로 몇 CH/2를 더한 경우와 일치) 10진수 &0~268435455 또는 16진수 #00000000~0FFFFFFF 필드

주:

· 위의 값은 BIN 형식 또는 TXT/CSV 형식에서 행바꿈 없음(C의 비트 08~11이 #0)을 지정했을 때만 유효합니다. 행바꿈 있음(C의 비트 08~11이 #8~D 중 하나)을 지정했을 때는 반드시 S1+3, S1+2를 #00000000으로 설정하십시오.

· S1~S1+3은 동일 영역 종류이어야 합니다. 이러한 데이터는 기능 지정이 데이터 읽기일 때만 유효합니다.

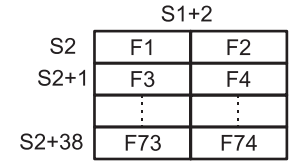
S2: 디렉토리명 및 파일명 저장 CH 번호

· 디렉토리명 및 파일명의 문자열 데이터. ASCII 코드로 입력.

· 디렉토리명: 1~65문자, 65문자 미만이어도 공백을 채우지 않습니다. 루트 디렉토리(\)에서 절대 경로로 지정하며 선두는 루트 디렉토리의 \(#5C)부터 시작합니다.

· 파일명: 식별자: 1~8문자, 8문자 미만이어도 공백을 채우지 않습니다. 파일명의 마지막에 NUL(#00)을 추가합니다(NUL은 8문자에 포함되지 않음). 확장자: .IOM으로 고정되어 있으므로 지정하지 않습니다.

· 디렉토리명과 파일명 사이는 \(#5C)로 구분합니다.



F1부터 순서대로 상위→하위 바이트,
하위→상위 채널에 저장.
파일명 마지막의 NUL(#00)을 포함하여
디렉토리명+파일명으로 최대 74문자
(바이트)

D: 읽기 데이터 출력 하위 CH 번호

· 데이터 읽기의 경우, D를 선두로, S1+1, S1로 지정하여 읽은 CH 수가 저장됩니다.

· 데이터 수 읽기의 경우, D+1, D에 16진수 #00000000~7FFFFFF의 값이 저장됩니다. D+1이 상위 4자리, D가 하위 4자리입니다(S1+1, S1과 동일한 BIN 형식).

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	–	–	○	–	–	○	–	–	–	–
S1, S2									○	○	–							
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C로 지정한 파일 메모리가 존재하지 않을 때 ON · C로 지정한 데이터가 범위 밖일 때 ON · S2로 지정한 파일명 지정 문자열이 조건을 만족하지 않을 때 ON · 파일 메모리 조작 실행 중 플래그가 ON일 때 ON · C에 상수 이외를 지정했을 때 ON(CS 시리즈 CS1 CPU 유니트(-V□ 없음)만 해당) · S1로 지정한 데이터가 범위 밖일 때 ON(CS 시리즈 CS1 CPU 유니트(-V□ 없음) 제외) · D로 지정한 영역이 잘못된 영역인 경우 ON · CS1D CPU 유니트에서 활성－대기 사이에서 동기화되지 않았을 때 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

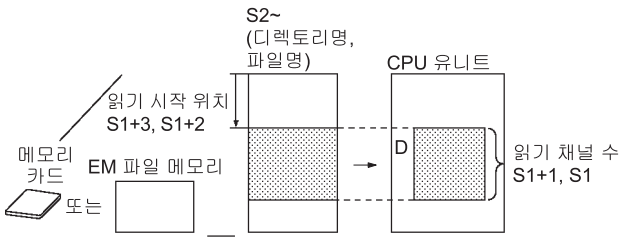
이름	주소	내용
메모리 카드 종류	A343.00~ A343.02	장착된 메모리 카드의 종류가 저장됩니다. (0: 없음, 4: 플래시 ROM)
메모리 카드 포맷 이상 플래그	A343.07	메모리 카드 장착 시, 메모리 카드가 포맷되지 않았을 때 또는 포맷 이상일 때 1(ON)이 됩니다.
파일 읽기 이상 플래그	A343.10	파일을 읽을 때 해당 파일의 데이터 형식이 잘못되었거나 파일이 손상되어 읽을 수 없는 경우 1(ON)이 됩니다.
파일 없음 플래그	A343.11	파일을 읽을 때 읽어올 파일이 존재하지 않는 경우 또는 지정한 파일이 존재하지 않는 경우 1(ON)이 됩니다.
파일 메모리 조작 실행 중 플래그	A343.13	다음 중 하나의 파일 메모리 관련 조작 실행 중에 1(ON). 아무 것도 실행하지 않을 때는 0(OFF). · 자체 CPU 유니트 대상 CMND 명령 실행 중 · FREAD/FWRIT 명령 실행 · 특수 보조 릴레이에 의한 프로그램 교체 중 · 간이 백업 조작 실행 중
파일 액세스 중 플래그	A343.14	파일 데이터를 실제 액세스하는 중에 1(ON)이 됩니다.
메모리 카드 인식 완료 플래그	A343.15	메모리 카드를 인식할 수 있으면 1(ON)이 됩니다. 메모리 카드를 인식할 수 없으면 0(OFF)이 됩니다.
EM 파일 메모리화 영역 포맷 이상 플래그	A343.06	EM 파일 메모리화 시작 뱅크 번호의 영역이 포맷 이상일 때 1(ON)이 됩니다.
남은 데이터 파일 전송 데이터 수	A346~A347	데이터 파일 전송 상황을 알리기 위한 2워드 채널(32비트)입니다. 데이터 파일 읽기(FREAD) 명령 또는 데이터 파일 쓰기(FWRIT) 명령의 실행 시작 시, 명령 피연산자를 통해 읽기 또는 쓰기 채널 수 또는 필드 수가 저장됩니다. 실제로 전송되면 1 채널 또는 1필드 전송될 때마다 감산됩니다. A346CH: 하위, A347CH: 상위의 32비트 BIN 값입니다.

FREAD

기능

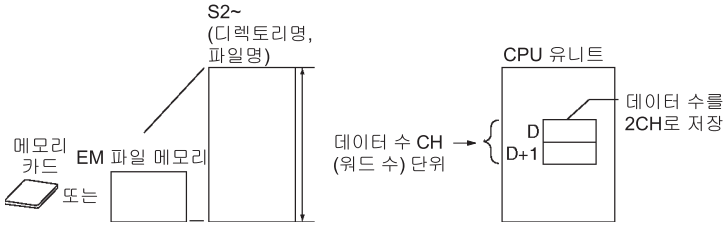
■ 데이터 읽기의 경우(C의 비트 4~7: #0)

C로 지정한 파일 메모리(메모리 카드 또는 EM 파일 메모리) 내의 S2로 지정한 파일명의 데이터 파일(확장자 .IOM, .TXT, .CSV 중 하나)의 S1+3, S1+2로 지정한 읽기 시작 위치(CH 오프셋)에서 S1+1, S1로 지정한 읽기 데이터 수 분량의 데이터를 채널 또는 필드 단위로 읽어 D를 선두로 한 지정된 I/O 메모리 영역에 출력합니다.



■ 데이터 수 읽기의 경우(C의 비트 4~7: #1)

C로 지정한 파일 메모리(메모리 카드 또는 EM 파일 메모리) 내의 S2로 지정한 파일명의 데이터 파일(확장자 .IOM, .TXT, .CSV 중 하나)의 데이터 수(몇 CH 또는 몇 필드의 데이터인지)를 읽어 D로 지정한 I/O 메모리 영역(D+1, D)에 BIN 형식으로 출력합니다.



힌트

- 래더 프로그램에서 파일 메모리 관련 다른 조작(파일 메모리 포맷, 파일 복사, 삭제 등)을 실행하는 경우는 CMND 명령(명령 송신 명령)을 통해 자체 CPU 유니트 대상으로 파일 메모리 관련 FINS 명령을 전송하십시오.
- S2 이후에 디렉토리명 및 파일명을 ASCII 코드로 상위 바이트→하위 바이트 순서, 하위 채널→상위 채널 순서에 따라 앞부터 채워서 저장하십시오.

  예: 디렉토리명: \ABC, 파일명: XYZ.IOM의 경우

S2	'\'	'A'
S2+1	'B'	'C'
S2+2	'\'	'X'
S2+3	'Y'	'Z'
S2+4	NUL	

→

S2	5C	41
S2+1	42	43
S2+2	5C	58
S2+3	59	5A
S2+4	00	

로 저장합니다.

사용상의 주의 사항

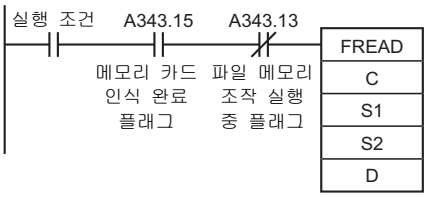
- 연산 처리에서 이 명령의 처리는 파일 메모리의 읽기 처리를 기동시키는 것뿐입니다. 설명서 뒷부분에 나온 명령 처리 시간은 이 기동 시간을 나타냅니다. 실제로 읽기(전송) 처리는 주변 서비스의 파일 액세스 처리에 의해 수행됩니다. 따라서 어떤 사이클에서 이 명령이 1회 실행되면 다음 사이클에서 입력 조건 OFF가 되어 실행되지 않아도 읽기 처리는 1회의 기동에 따라 계속 실행됩니다.

  또한 실제 전송이 완료될 때까지의 시간은 데이터의 양이나 파일 액세스 처리에 지정된 서비스 시간에 따라 달라집니다. 기준으로, 사이클 시간 10ms, 루트 디렉토리의 파일에 액세스하는 조건에서 전송 채널 수 1024CH일 때 0.92s, 9999CH일 때 4.64s가 됩니다.
- 읽기(전송) 처리가 종료되면 파일 메모리 조작 실행 중 플래그(A343.13)가 0(OFF)이 됩니다. 따라서 이 플래그(A343.13)를 사용하여 파일 메모리용 명령 간에 배타적으로 제어하십시오.
- 읽기 채널 수가 읽기 출력 대상의 영역 종류 최대 범위를 초과한 경우에도 실제 RAM상의 연속 위치(I/O 메모리 실제 주소의 순서)에 데이터가 저장됩니다.
  예: 읽기 출력 대상이 내부 보조 릴레이 WR 영역이고 읽기 채널 수가 WR 영역을 초과하는 경우, I/O 메모리 실제 주소에 따른 영역 종류(이 경우, 타이머 현재값)에 연속하여 값이 저장됩니다.
- 이 명령 실행 시작 시, '남은 데이터 파일 전송 데이터 수' 영역(A346~347CH)에 채널 수가 저장되며 실제로 읽으면 1 채널 또는 1 필드 읽을 때마다 감산됩니다. 이 영역의 값을 확인하면 지정된 채널 수 또는 지정된 필드 수의 읽기가 실제로 성공했는지 여부를 확인할 수 있습니다.
- 읽기 채널 수보다 데이터 파일의 채널 수가 작은 경우는 데이터 파일의 채널 수만큼 읽으며 에러가 되지 않습니다.
- 읽기 시작 위치가 데이터 파일의 채널 수보다 큰 경우는 '파일 읽기 이상 플래그'를 ON하고 읽기를 실행하지 않습니다.
- 지정한 데이터 파일이 파일 메모리에 존재하지 않는 경우 또는 지정된 영역이 존재하지 않는 경우는 '파일 없음 플래그'(A343.11)를 ON하고 읽기는 실행하지 않습니다.

- 명령 실행 중에는 '파일 메모리 조작 실행 중 플래그'(A343.13)를 ON합니다. 실행 시작 시, 이미 플래그가 ON되어 있는 경우는 에러가 되어 에러 플래그가 ON이 됩니다.
- 다음 중 하나인 경우 '파일 읽기 이상 플래그'(A343.10)를 ON하고 읽기를 실행하지 않습니다.
  - 실제 파일 형식이 지정된 파일 형식이 아닌 경우
    특히, TXT, CSV 형식의 경우 다음 조건을 만족하지 않는 경우

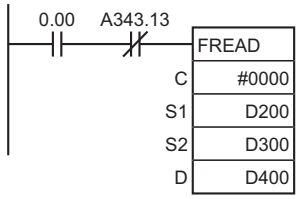
문자 코드:	반각 16진수 값(0~9, A~F, a~f)
구분(필드) 크기:	워드 형식: 16진수 값 4자리 LONG 워드 형식: 16진수 값 8자리

  (이 경우, 다음 값 이외의 값을 읽은 시점에서 읽기가 도중에 종료되지만 그 이전의 값은 읽어옵니다.)
  - 데이터 파일이 손상된 경우
- 메모리 카드를 장착한 후 CPU 유니트가 인식을 완료할 때까지는 몇 초 정도 걸립니다. 따라서 전원 ON 시 또는 메모리 카드 장착 후 즉시 메모리 카드에 액세스하는 경우에는 오른쪽 그림과 같이 '메모리 카드 인식 완료 플래그'(A343.15)를 a 접점의 입력 조건으로 하십시오.



샘플 프로그램

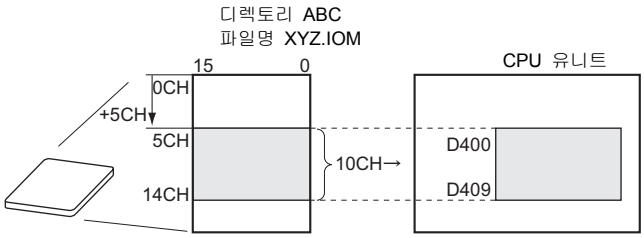
0.00이 ON이면 메모리 카드 내의 루트 디렉토리의 디렉토리 ABC의 파일명 XYZ.IOM의 선두+5CH부터 10CH의 데이터를 읽어 D400~D409에 저장합니다.



C: # 0 0 0 0 파일 메모리 = 메모리 카드, 기능 지정 = 데이터 읽기

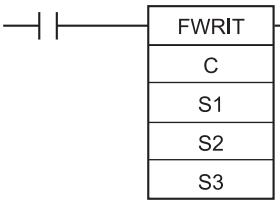


S2:D300	5	C	4	1	디렉토리명: \ABC
S2+1:D301	4	2	4	3	파일명: XYZ
S2+2:D302	5	C	5	8	
S2+3:D303	5	9	5	A	
S2+4:D304	0	0	(주)		주: 무시합니다.



FWRIT

명령 이름	니모닉	동작 옵션	평션 번호	기능
데이터 파일 쓰기	FWRIT	@FWRIT	701	지정된 I/O 메모리 영역의 데이터를 지정된 데이터 파일(I/O 메모리 파일)에 씁니다.

심볼	FWRIT
	

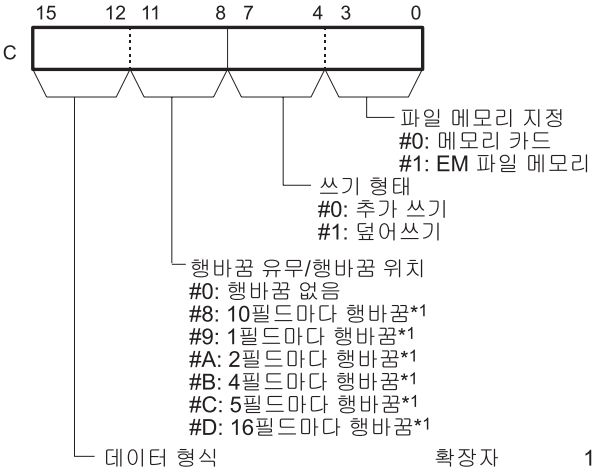
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	UINT	가변
S1	쓰기 채널 수 및 쓰기 시작 위치	LWORD	4
S2	디렉토리명 및 파일명 저장 CH 번호	UINT	가변
S3	읽기 원본 데이터 출력 하위 CH 번호	UINT	가변

C: 제어 데이터



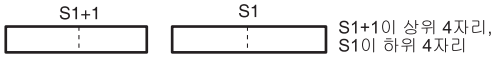
*1: CS 시리즈 CS1 CPU 유니트(-V□ 없음)에서는 사용할 수 없습니다.

주 1: 워드 형식의 경우: 1필드 = I/O 메모리의 1CH, LONG 워드 형식의 경우: 1필드 = I/O 메모리의 2CH

주 2: LONG 워드 형식의 경우, I/O 메모리의 상위→하위 CH 순서로 저장됩니다(예: #12345678을 D00000: #5678, D00001: #1234로 저장).

S1: 쓰기 채널 수 및 쓰기 시작 위치

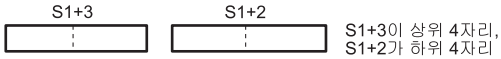
· S1+1, S1: 쓰기 데이터 수



데이터 형식	C의 비트 12~15	S1+1, S1의 내용
BIN 형식의 경우	#0(BIN)	쓰기 채널 수(몇 CH분을 파일 메모리에 쓸지 지정) 10진수 &0~1073741823 또는 16진수 #00000000~3FFFFFFF
워드 형식의 경우	#1(구분 없음) #3(쉼표 구분) #5(탭 구분)	쓰기 필드 수(1필드 = 1CH이므로 CH 수와 일치) 10진수 &0~536870911 또는 16진수 #00000000~1FFFFFFF
LONG 워드 형식의 경우	#2(구분 없음) #4(쉼표 구분) #6(탭 구분)	쓰기 필드 수(1필드 = 2CH이므로 CH 수/2와 일치) 10진수 &0~268435455 또는 16진수 #00000000~0FFFFFFF

· S1+3, S1+2: 쓰기 시작 위치

파일 메모리 선두부터 쓰기 시작 위치 지정



데이터 형식	C의 비트 12~15	S1+1, S1의 내용
BIN 형식의 경우	#0(BIN)	선두에서 몇 CH 더한 위치에 쓸지 지정 10진수 &0~1073741823 또는 16진수 #00000000~3FFFFFFF
워드 형식의 경우	#1(구분 없음) #3(쉼표 구분) #5(탭 구분)	선두에서 몇 필드 더한 위치에 쓸지 지정(1필드 = 1CH이므로 CH를 더한 경우와 일치) 10진수 &0~536870911 또는 16진수 #00000000~1FFFFFFF
LONG 워드 형식의 경우	#2(구분 없음) #4(쉼표 구분) #6(탭 구분)	선두에서 몇 필드 더한 위치에 쓸지 지정(1필드 = 2CH이므로 CH÷2를 더한 경우와 일치) 10진수 &0~268435455 또는 16진수 #00000000~0FFFFFFF 필드

주:

· 위의 값은 BIN 형식의 경우, 덮어쓰기를 실행할 때만 유효합니다.
TXT 또는 CSV 형식의 경우, 행바꿈 없음(C의 비트 08~11이 #0)을 지정하고 덮어쓰기를 실행(C의 비트 04~07이 #1)할 때만 유효합니다.
행바꿈 있음(C의 비트 08~11이 #8~D 중 하나)을 지정했을 때는 반드시 S1+3, S1+2를 #00000000으로 설정하십시오.

· S1~S1+3은 동일 영역 종류이어야 합니다.

S2: 디렉토리명 및 파일명 저장 CH 번호

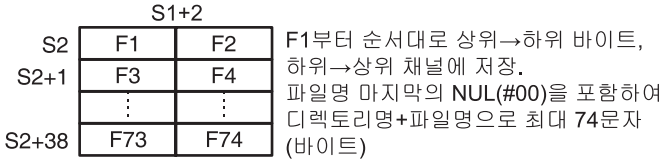
· 디렉토리명 및 파일명의 문자열 데이터. ASCII 코드로 입력

· 디렉토리명: 1~65문자, 65문자 미만이어도 공백을 채우지 않으며 루트 디렉토리(\)에서 절대 경로로 지정합니다.

· 선두는 루트 디렉토리의 \(#5C)부터 시작합니다.

· 파일명: 식별자: 1~8문자, 8문자 미만이어도 공백을 채우지 않습니다. 파일명의 마지막에 NUL(#00)을 추가합니다(NUL은 8문자에 포함되지 않음). 확장자: .IOM으로 고정되어 있으므로 지정하지 않습니다.

· 디렉토리명과 파일명 사이는 \(#5C)로 구분합니다.



S3: 쓰기 원본 데이터 하위 CH 번호

쓰기 원본 데이터 하위 CH 번호(I/O 메모리에서 어느 영역 종류의 어느 주소의 데이터를 파일 메모리에 쓸지 지정)

FWRIT

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	–	–	○	–	–	○	–	–	–	–
S1, S2									○	○	–							
S3																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· C로 지정한 파일 메모리가 존재하지 않을 때 ON · C로 지정한 데이터가 범위 밖일 때 ON · S2로 지정한 파일명 지정 문자열이 조건을 만족하지 않을 때 ON · 파일 메모리 조작 실행 중 플래그가 ON일 때 ON · C에 상수 이외를 지정했을 때 ON(CS 시리즈 CS1 CPU 유니트(-V□ 없음)만 해당) · S1로 지정한 데이터가 범위 밖일 때 ON(CS 시리즈 CS1 CPU 유니트(-V□ 없음) 제외) · S3으로 지정한 영역이 잘못된 영역인 경우 ON · CS1D CPU 유니트에서 활성－대기 사이에서 동기화되지 않았을 때 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

이름	주소	내용
메모리 카드 종류	A343.00~ A343.02	장착된 메모리 카드의 종류가 저장됩니다(0: 없음, 4: 플래시 ROM).
메모리 카드 포맷 이상 플래그	A343.07	메모리 카드 장착 시, 메모리 카드가 포맷되지 않았을 때 또는 포맷 이상일 때 1(ON)이 됩니다.
파일 쓰기 이상 플래그	A343.08	파일 쓰기에서 이상이 발생했을 때 1(ON)이 됩니다. 쓰기 파일이 쓰기 방지(읽기 전용)되어 있을 때도 1(ON)이 됩니다.
파일 쓰기 불가 플래그	A343.09	파일 쓰기 실행 시, 용량 초과 때문에 쓰기 작업을 수행할 수 없는 경우 1(ON)이 됩니다.
파일 없음 플래그	A343.11	파일 쓰기 실행 시, 지정된 디렉토리가 존재하지 않을 때 1(ON)이 됩니다.
파일 메모리 조작 실행 중 플래그	A343.13	다음 중 하나의 파일 메모리 관련 조작 실행 중에 1(ON). 아무 것도 실행하지 않을 때는 0(OFF). · 자체 CPU 유니트 대상 CMND 명령 실행 중 · FREAD/FWRIT 명령 실행 · 특수 보조 릴레이에 의한 프로그램 교체 중 · 간이 백업 조작 실행 중
파일 액세스 중 플래그	A343.14	파일 데이터를 실제 액세스하는 중에 1(ON)이 됩니다.
메모리 카드 인식 완료 플래그	A343.15	메모리 카드를 인식할 수 있으면 1(ON)이 됩니다. 메모리 카드를 인식할 수 없으면 0(OFF)이 됩니다.
EM 파일 메모리화 영역 포맷 이상 플래그	A343.06	EM 파일 메모리화 시작 뱅크 번호의 영역이 포맷 이상일 때 1(ON)이 됩니다.
남은 데이터 파일 전송 데이터 수	A346~A347	데이터 파일 전송 상황을 알리기 위한 2워드 채널(32비트)입니다. 데이터 파일 쓰기(FWRIT) 명령 실행 시작 시, 명령 피연산자에 의해 읽기 또는 쓰기 채널 수가 저장됩니다. 실제로 전송되면 감산됩니다. A346CH: 하위, A347CH: 상위의 32비트 BIN 값입니다.

기능

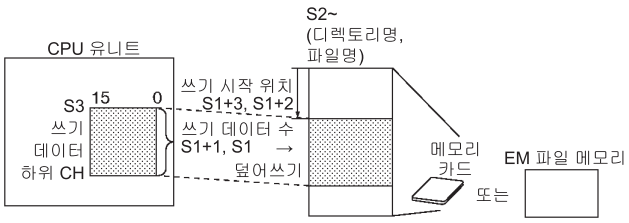
■ 작성이 끝난 파일에 기록하는 경우

기록 형태로 덮어쓰기와 추가하여 쓰기가 가능합니다. C의 비트 4~7로 지정합니다.

· 덮어쓰기의 경우(C의 비트 4~7: #1)

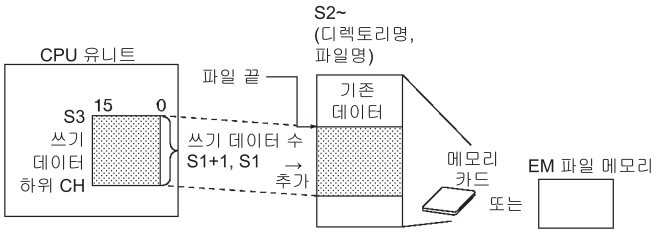
S3으로 지정한 I/O 메모리 영역의 시작 CH에서 S1+1, S1로 지정한 채널 수를 C로 지정한 파일 메모리(메모리 카드 또는 EM 파일 메모리) 내의 S2로 지정한 파일명의 데이터 파일(확장자 .IOM, .TXT, .CSV 중 하나)의 S1+3, S1+2로 지정한 쓰기 시작 위치(CH 오프셋)부터 덮어 씁니다.

이 경우, 기존 파일의 용량 범위 내에서 덮어 쓰며 용량을 초과하는 분량은 기록되지 않습니다.



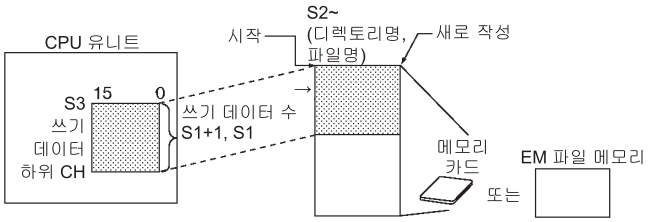
· 추가하여 쓰기(C의 비트 4~7: #0)

S3으로 지정한 I/O 메모리 영역의 시작 CH에서 S1+1, S1로 지정한 채널 수를 C로 지정한 파일 메모리(메모리 카드 또는 EM 파일 메모리) 내의 기존 데이터의 끝에 추가하여 지정된 데이터 형식으로 기록합니다.



■ 파일을 새로 작성하여 기록하는 경우

지정한 데이터 파일이 파일 메모리에 존재하지 않는 경우, 데이터 파일을 새로 작성(확장자 .IOM, .TXT, .CSV 중 하나)하여 선두부터 지정된 데이터 형식으로 데이터를 기록합니다. 이 경우, 기록 형태의 지정은 #0(추가하여 쓰기), #1(덮어쓰기) 모두 가능합니다.



■ 데이터 형식, 행바꿈 유무 및 행바꿈 위치, 쓰기 필드 수

· 워드 형식의 경우, I/O 메모리의 데이터를 I/O 메모리의 하위 CH→상위 CH 순서로 1CH마다(필드로) 구분하여 저장합니다. LONG 워드 형식의 경우, I/O 메모리의 하위 CH→상위 CH 순서로 2CH마다(2CH 이내는 상위 CH→하위 CH 순서) 구분하여(필드로) 저장합니다.

· 구분하지 않는 경우 필드 사이를 채워서 연속으로 저장합니다. 쉼표로 구분하는 경우 필드 사이에 쉼표 코드를 삽입해서 저장합니다. 탭으로 구분하는 경우에는 필드 사이에 탭 코드를 삽입해서 저장합니다.

· 행을 구분하는 경우, 지정된 필드 수에 따라 행바꿈 코드를 삽입해서 저장합니다. 행을 구분하지 않는 경우에는 연속으로 저장합니다.

· 쓰기 크기로 지정한 필드 수를 I/O 메모리에서 읽어, 위의 데이터 형식으로 파일 메모리에 기록합니다.

· 지정한 디렉토리가 존재하지 않는 경우는 '파일 없음 플래그'(A343.11)가 1(ON)이 되고 파일 쓰기는 실행되지 않습니다.

예: 디렉토리명: \ABC, 파일명: XYZ의 경우

S2	'\'	'A'
S2+1	'B'	'C'
S2+2	'\'	'X'
S2+3	'Y'	'Z'
S2+4	NUL	

→

S2	5C	41
S2+1	42	43
S2+2	5C	58
S2+3	59	5A
S2+4	00	

로 저장합니다.

힌트

· 래더 프로그램에서 파일 메모리 관련 다른 조작(파일 메모리 포맷, 파일 복사, 삭제 등)을 실행하는 경우는 CMND 명령(명령 송신 명령)을 통해 자체 CPU 유니트 대상으로 파일 메모리 관련 FINS 명령을 전송하십시오.

사용상의 주의 사항

· 연산 처리에서 이 명령의 처리는 파일 메모리의 쓰기 처리를 기동시키는 것뿐입니다. 설명서 뒷부분에 나온 명령 처리 시간은 이 기동 시간을 나타냅니다. 실제로 쓰기(전송) 처리는 주변 서비스의 파일 액세스 처리에 의해 수행됩니다. 따라서 어떤 사이클에서 이 명령이 1회 실행되면 다음 사이클에서 입력 조건이 OFF가 되어 실행되지 않아도 쓰기 처리는 1회의 기동에 따라 계속 실행됩니다.

  또한 실제 전송이 완료될 때까지의 시간은 데이터의 양이나 파일 액세스 처리에 지정된 서비스 시간에 따라 달라집니다. 기준으로, 사이클 시간 10ms, 루트 디렉토리의 파일에 액세스하는 조건에서 전송 채널 수 1024CH일 때 1.97s[새 파일], 1.33s[기존 파일], 9999CH일 때 6.64s[새 파일], 6.12s[기존 파일]이 됩니다.

· 쓰기(전송) 처리가 종료되면 파일 메모리 조작 실행 중 플래그(A343.13)가 0(OFF)이 됩니다. 따라서 이 플래그(A343.13)를 사용하여 파일 메모리 명령 간에 배타적으로 제어하십시오.

· 쓰기 채널 수가 쓰기 원본의 영역 종류 최대 범위를 초과한 경우에도 실제 RAM상의 연속 위치(I/O 메모리 실제 주소의 순서)의 데이터를 I/O 메모리에서 읽어와 파일 메모리에 기록합니다.

  예: 쓰기 원본이 내부 보조 릴레이 WR 영역에서 쓰기 채널 수가 WR 영역을 초과한 경우, I/O 메모리 실제 주소의 다음 영역 종류(이 경우, 타이머 현재값)의 값을 읽어 파일 메모리에 연속으로 기록합니다.

· 이 명령 실행 시작 시, '남은 데이터 파일 전송 데이터 수' 영역(A346~347CH)에 쓰기 채널 수 또는 필드 수(S1+1, S1)가 저장되며 실제로 쓰기 처리되면 감산됩니다. 이 영역의 값을 확인하면 지정된 채널 수의 쓰기가 실제로 성공했는지 여부를 확인할 수 있습니다.

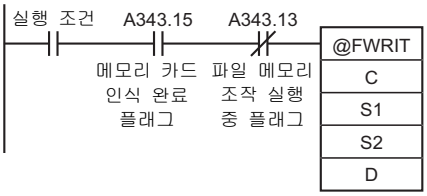
· 명령 실행 중에는 '파일 메모리 조작 실행 중 플래그'(A343.13)가 1(ON)이 됩니다. 명령 실행 시작 시, 이미 이 플래그가 1(ON)인 경우는 에러가 되어 에러 플래그가 ON이 됩니다.

· 쓰기 시작 위치가 데이터 파일의 채널 수보다 큰 경우는 '파일 쓰기 이상 플래그'(A343.08)를 ON하고 쓰기는 실행하지 않습니다

· 용량 부족, 쓰기 방지 요인으로 데이터를 추가하거나 덮어 쓸 수 없는 경우 또는 파일을 작성할 수 없는 경우는 '파일 쓰기 불가 플래그'(A343.09)를 ON하고 쓰기는 실행하지 않습니다

· 파일 형식이 데이터 파일의 파일 형식(BIN)이 아닌 경우나 데이터 파일이 손상된 경우는 '파일 쓰기 이상 플래그'(A343.08)를 ON하고 쓰기는 실행하지 않습니다

· 메모리 카드를 장착한 후 CPU 유니트가 인식을 완료할 때까지는 몇 초 정도 걸립니다. 따라서 전원 ON 시 또는 메모리 카드 장착 후 즉시 메모리 카드에 액세스하는 경우에는 다음과 같이 '메모리 카드 인식 완료 플래그'(A343.15)를 a 접점의 입력 조건으로 하십시오.



· S3 이후의 쓰기 원본 데이터는 이 명령이 실행된 후 주변 서비스의 파일 액세스 처리를 통해 참조됩니다. 따라서 파일 메모리의 쓰기 처리가 완료되지 않은 동안, 쓰기 원본 데이터를 변경하면 변경된 내용이 기록될 가능성이 있습니다.

TWRIT

명령 이름	니모닉	동작 옵션	평션 번호	기능
텍스트 파일 쓰기	TWRIT	@TWRIT	704	I/O 메모리 영역에 있는 ASCII 데이터를 메모리 카드에 텍스트 파일로 저장합니다.

심볼	TWRIT	
	TWRIT	
	C	C: 제어 데이터
	S1	S1: 쓰기 바이트 수
	S2	S2: 디렉토리명 및 파일명 저장 시작 CH 번호
	S3	S3: 쓰기 데이터 저장 시작 CH 번호
	S4	S4: 구분 문자

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	UINT	1
S1	쓰기 바이트 수	UINT	1
S2	디렉토리명 및 파일명 저장 시작 CH 번호	WORD	1
S3	쓰기 데이터 저장 시작 CH 번호	WORD	1
S4	구분 문자	WORD	1

C: 제어 데이터

- #0000: 추가하여 쓰기
- #0001: 새로 작성 또는 덮어쓰기

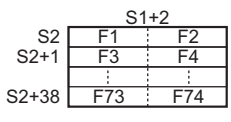
S1: 쓰기 바이트 수

10진수 &0~255 또는 16진수 #0000~#00FF

S2: 디렉토리명 및 파일명 저장 시작 CH 번호

- 디렉토리명 및 파일명의 문자열 데이터. ASCII 코드로 입력
- 디렉토리명: 1~65문자, 65문자 미만이어도 공백을 채우지 않습니다. 루트 디렉토리(\)에서 절대 경로로 지정합니다.
- 선두는 루트 디렉토리의 \(#5C)부터 시작합니다.
- 파일명: 식별자: 1~8문자, 8문자 미만이어도 공백을 채우지 않습니다. 파일명의 마지막에 NUL(#00)을 추가합니다(NUL은 8문자에 포함되지 않음). 확장자: 없음
- 디렉토리명과 파일명 사이는 \(#5C)로 구분합니다.

주: S2 이후(디렉토리명 및 파일명)는 동일 영역 종류이어야 합니다.



F1부터 순서대로 상위→하위 바이트, 하위→상위 채널에 저장. 파일명 마지막의 NUL(#00)을 포함하여 디렉토리명+파일명으로 최대 74문자(바이트)

S3: 쓰기 데이터 저장 시작 CH 번호

쓰기 원본 데이터 하위 CH 번호(I/O 메모리에서 어느 영역 종류의 어느 주소의 데이터를 쓸지 지정)

주: S3 이후의 쓰기 원본 영역은 동일한 영역 종류이어야 합니다. 단, I/O 메모리 실제 주소 순서의 데이터를 파일로 쓰기 처리합니다.

TWRIT

S4: 구분 문자

쓰기 데이터에 대해 구분 문자(최대 2바이트)를 지정하는 경우 ASCII 코드를 지정합니다. 구분 문자가 필요 없는 경우는 #0000을 지정하십시오.

최대 2바이트를 지정할 수 있습니다. 1바이트를 지정하는 경우는 하위 바이트에 #00을 지정하십시오.

대표적인 구분 문자

16진수 #2C00: 쉼표 문자(1바이트)

16진수 #0A00: 행바꿈(1바이트)

16진수 #0D0A: 행바꿈 복귀(2바이트)

16진수 #0C00: 페이지 바꿈(1바이트)

16진수 #0900: 탭(1바이트)

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
S1																		
S2, S3											–							
S4											○							

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 메모리 카드가 존재하지 않을 때 ON · C로 지정한 데이터가 범위 밖일 때 ON · S2로 지정한 파일명 지정 문자열이 조건을 만족하지 않을 때 ON · 파일 메모리 조작 실행 중 플래그가 ON일 때 ON · S3으로 지정한 영역이 잘못된 영역인 경우 ON · CS1D CPU 유니트에서 활성－대기 사이에서 동기화되지 않았을 때 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

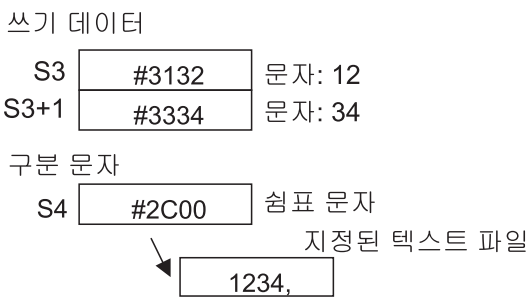
이름	주소	내용
메모리 카드 포맷 이상 플래그	A343.07	메모리 카드 장착 시, 메모리 카드가 포맷되지 않았을 때 또는 포맷 이상일 때 1(ON)이 됩니다.
파일 쓰기 이상 플래그	A343.08	파일 쓰기에서 이상이 발생했을 때 1(ON)이 됩니다.
파일 쓰기 불가 플래그	A343.09	파일 쓰기 실행 시, 쓰기 방지되어 있거나 용량 초과 때문에 쓰기 작업을 수행할 수 없는 경우 1(ON)이 됩니다.
파일 없음 플래그	A343.11	파일 쓰기 실행 시, 지정된 디렉토리가 존재하지 않을 때 1(ON)이 됩니다.
파일 메모리 조작 실행 중 플래그	A343.13	다음 중 하나의 파일 메모리 관련 조작 실행 중에 1(ON). 아무 것도 실행하지 않을 때는 0(OFF) · 자체 CPU 유니트 대상 CMND 명령 실행 중 · FREAD/FWRIT/TWRIT 명령 실행 중 · 특수 보조 릴레이에 의한 프로그램 교체 중 · 간이 백업 조작 실행 중
파일 액세스 중 플래그	A343.14	파일 데이터를 실제 액세스하는 중에 1(ON)이 됩니다.
메모리 카드 인식 완료 플래그	A343.15	메모리 카드를 인식할 수 있으면 1(ON)이 됩니다. 메모리 카드를 인식할 수 없으면 0(OFF)이 됩니다.
남은 데이터 파일 전송 데이터 수	A346~A347	데이터 파일 전송 상황을 알리기 위한 2워드 채널(32비트)입니다. 명령 실행 시작 시, 쓰기 바이트 수가 저장됩니다. 실제로 전송되면 감산됩니다. A346CH: 하위, A347CH: 상위의 32비트 BIN 값입니다.

기능

S3으로 지정한 시작 CH에 있는 ASCII 데이터를 S1로 지정한 바이트 수만큼 S2로 지정한 파일명의 텍스트 파일(확장자 .TXT)로 메모리 카드에 기록합니다.

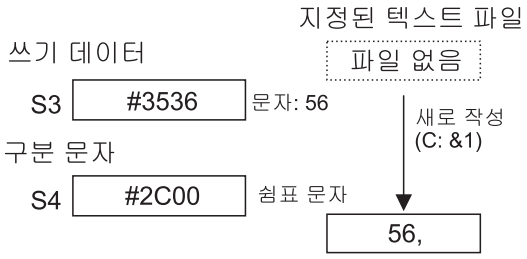
텍스트 파일의 마지막에 S4로 나타내는 구분 문자를 부가할 수 있습니다.

작성한 텍스트 파일은 텍스트 에디터 등으로 참조할 수 있습니다.



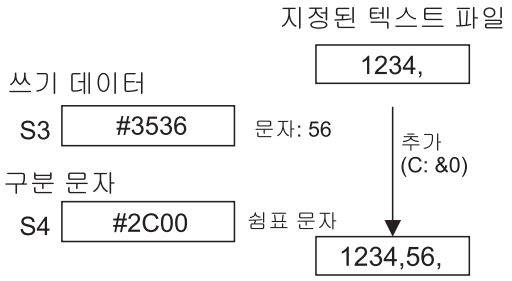
· 파일을 새로 작성하는 경우

C(제어 데이터)에 1을 지정합니다.

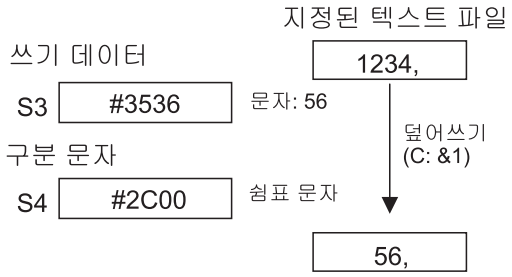


· 작성이 끝난 파일에 기록하는 경우

기존 파일에 추가하는 경우는 C(제어 데이터)에 0을 지정합니다.



기존 파일에 덮어 쓸 때는C(제어 데이터)에 1을 지정합니다.



■ 데이터 형식

· I/O 메모리의 데이터는 ASCII 코드로 상위 바이트→하위 바이트 순서, 하위 채널→상위 채널 순서에 따라 앞부터 채워서 저장하십시오.

문자열 12345678을 쓰는 경우

S3	#3132	문자: 12
S3+1	#3334	문자: 34
S3+2	#3536	문자: 56
S3+3	#3738	문자: 78

■ 디렉토리명 및 파일명(S2)

· 디렉토리명은 루트 디렉토리(\)에서 절대 경로로 지정합니다. 루트 디렉토리의 \(#5C)는 선두에 필요합니다. 또한 디렉토리 길이는 최대 65문자이며 65문자 미만인 경우에도 공백으로 채우지 않습니다. 하위 디렉토리와는 \(#5C)를 사용하여 구분합니다. 사용할 수 있는 문자는 "a~z", "A~Z", "0~9"이고 ASCII 코드로 설정합니다.

· 파일명은 1~8문자의 ASCII 코드로 설정합니다. a~z, A~Z, 0~9의 ASCII 코드를 사용하십시오. 8문자 미만인 경우에도 공백으로 채우지 않습니다. 마지막에는 반드시 NUL(#00)을 삽입합니다.

· 확장자인 .TXT는 고정되어 있으므로 지정하지 않습니다.

· S2 이후에 디렉토리명 및 파일명을 ASCII 코드로 상위 바이트→하위 바이트 순서, 하위 채널→상위 채널 순서에 따라 앞부터 채워서 저장하십시오.

· 지정한 디렉토리가 존재하지 않는 경우는 '파일 없음 플래그'(A343.11)가 1(ON)이 되고 파일 쓰기는 실행되지 않습니다.

예: 디렉토리명: \ABC, 파일명: XYZ의 경우

S2	'\'	'A'
S2+1	'B'	'C'
S2+2	'\'	'X'
S2+3	'Y'	'Z'
S2+4	NUL	

→

S2	5C	41
S2+1	42	43
S2+2	5C	58
S2+3	59	5A
S2+4	00	

로 저장합니다.

힌트

· 래더 프로그램에서 파일 메모리 관련 다른 조작(파일 메모리 포맷, 파일 복사, 삭제 등)을 실행하는 경우는 CMND 명령(명령 송신 명령)을 통해 자체 CPU 유니트 대상으로 파일 메모리 관련 FINS 명령을 전송하십시오.

사용상의 주의 사항

· 연산 처리에서 이 명령의 처리는 파일 메모리의 쓰기 처리를 기동시키는 것뿐입니다. 설명서 뒷부분에 나온 명령 처리 시간은 이 기동 시간을 나타냅니다. 실제로 쓰기(전송) 처리는 주변 서비스의 파일 액세스 처리에 의해 수행됩니다. 따라서 어떤 사이클에서 이 명령이 1회 실행되면 다음 사이클 이후에서 입력 조건이 OFF가 되어도 쓰기 처리는 1회의 기동에 따라 계속 실행됩니다.
또한 실제 전송이 완료될 때까지의 시간은 데이터의 양이나 파일 액세스 처리에 지정된 서비스 시간에 따라 달라집니다. 기준으로, 사이클 시간 10ms, 루트 디렉토리의 파일에 액세스하는 조건에서 쓰기 바이트 수 100바이트일 때 440ms[신규 파일], 260ms[기존 파일에 추가], 255바이트일 때 450ms[새 파일], 270ms[기존 파일에 추가]가 됩니다.
단, 이러한 값은 메모리 카드의 종류나 메모리 카드 내의 파일 수에 따라 크게 달라집니다.

· 쓰기(전송) 처리가 종료되면 파일 메모리 조작 실행 중 플래그(A343.13)가 0(OFF)이 됩니다. 따라서 이 플래그(A343.13)를 사용하여 파일 메모리 명령 간에 배타적으로 제어하십시오.

· 쓰기 바이트 수가 쓰기 원본의 영역 종류 최대 범위를 초과한 경우에도 실제로 RAM상의 연속 위치(I/O 메모리 실제 주소의 순서)의 데이터를 I/O 메모리에서 읽어와 파일 메모리에 기록합니다.
예: 쓰기 원본이 내부 보조 릴레이 WR 영역에서 쓰기 채널 수가 WR 영역을 초과한 경우, I/O 메모리 실제 주소의 다음 영역 종류(이 경우, 타이머 현재값)의 값을 읽어 파일 메모리에 연속으로 기록합니다. 실제 RAM상의 연속 위치에 대해서는 부록의 '메모리 맵'을 참조하십시오.

· 이 명령 실행 시작 시, '남은 데이터 파일 전송 데이터 수' 영역(A346~347CH)에 쓰기 바이트 수가 저장되며 실제로 쓰기 처리되면 감산됩니다. 이 영역의 값을 확인하면 지정된 채널 수의 쓰기가 실제로 성공했는지 여부를 확인할 수 있습니다.

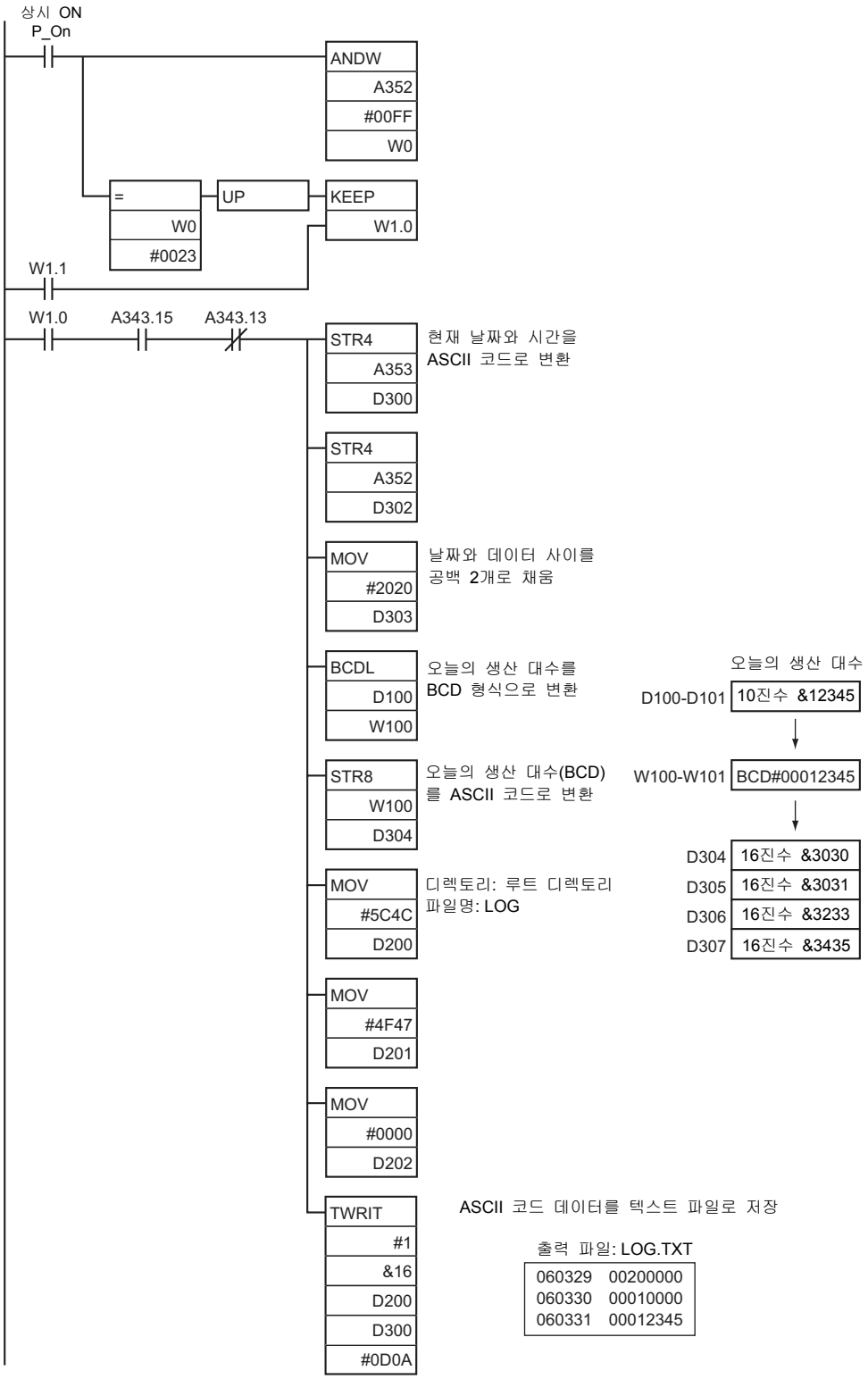
· 명령 실행 중에는 '파일 메모리 조작 실행 중 플래그'(A343.13)가 1(ON)이 됩니다. 명령 실행 시작 시, 이미 이 플래그가 1(ON)인 경우는 에러가 되어 에러 플래그가 ON이 됩니다.

· 용량 부족, 쓰기 방지 요인으로 데이터를 추가하거나 덮어 쓸 수 없는 경우 또는 파일을 작성할 수 없는 경우는 '파일 쓰기 불가 플래그'(A343.09)를 ON하고 쓰기는 실행하지 않습니다

· 메모리 카드를 장착한 후 CPU 유니트가 인식을 완료할 때까지는 몇 초 정도 걸립니다. 따라서 전원 ON 시 또는 메모리 카드 장착 후 즉시 메모리 카드에 액세스하는 경우에는 '메모리 카드 인식 완료 플래그'(A343.15)를 a 접점의 입력 조건으로 하십시오.

샘플 프로그램

D100~D101에 1일 생산 대수가 16진수 8자리로 저장되어 있습니다.
매일 23시 0분에 생산 대수 데이터를 BCD 형식으로 변환하여
메모리 카드 내 루트 디렉토리의 파일명 LOG.TXT에 추가하여 씁니다.

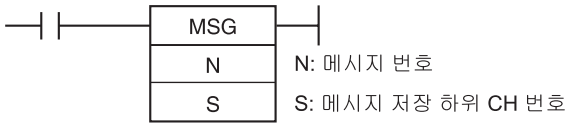


표시 기능용 명령

니모닉	명령 이름	펑션 번호	페이지
MSG	메시지 표시	046	3-848

MSG

명령 이름	니모닉	동작 옵션	평션 번호	기능
메시지 표시	MSG	@MSG	046	지정된 CH에서 16채널(워드)분의 메시지(ASCII 코드)를 읽어서 프로그래밍 콘솔에 표시합니다.

심볼	MSG
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	메시지 번호	UINT	1
S	메시지 저장 하위 CH 번호	UINT	가변

N: 메시지 번호
10진수 &0~7 또는 16진수 #0000~0007
S: 메시지 저장 하위 CH 번호
메시지를 표시할 때는 CH 지정
메시지 표시를 해제할 때는 16진수 #0000~FFFF

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
S												–						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· N의 데이터가 0~7의 범위 안에 없을 때 ON · 그 이외일 때 OFF

기능

입력 조건이 ON이면 N으로 지정한 메시지 번호에 대해 S로 지정한 메시지 저장 하위 CH 번호부터 16CH분의 ASCII 코드 데이터(NUL을 포함하여 최대 32문자)를 등록합니다. 메시지가 등록되면 프로그래밍 콘솔을 연결할 때 이상 메시지가 프로그래밍 콘솔에 계속 표시됩니다.

메시지 등록 후 메시지 저장 영역의 내용을 재기입하면 표시 메시지를 바꿀 수 있습니다.

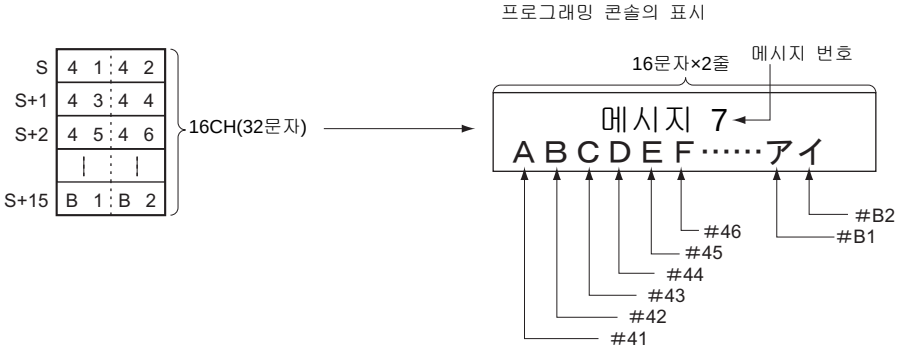
일단 등록한 메시지를 해제하기 위해서는 해제하려는 메시지 번호를 N으로 지정하고 S에 상수(#0000~FFFF 중 하나)를 지정하여 이 명령을 실행합니다.

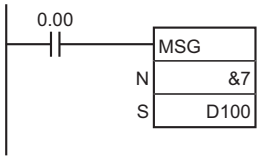
프로그램 실행 중에 등록된 메시지는 프로그램의 실행이 정지되어도 유지됩니다. 단, 다음 프로그램 실행 시작 시에는 모든 메시지가 해제됩니다.

사용상의 주의 사항

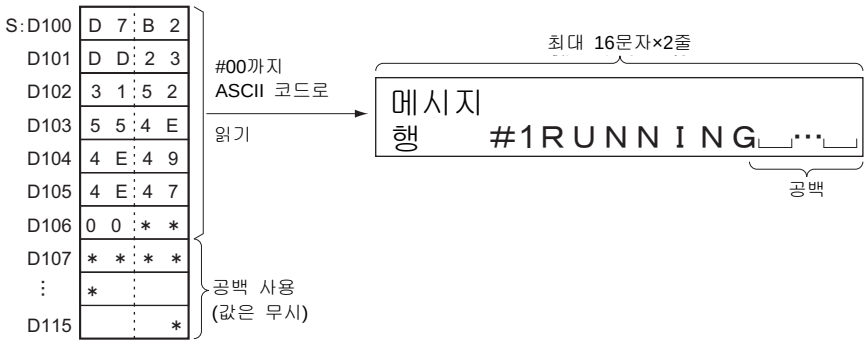
- 이미 메시지가 등록된 경우에도 이후의 명령 실행을 우선하여 등록을 갱신합니다.
- #00 이후를 프로그래밍 콘솔에서 공백으로 치환합니다.
- 메시지 표시 순서는 상위 바이트→하위 바이트 순서입니다.

샘플 프로그램





0.00이 ON이면 메시지 번호 7로 D100~D115의 16CH 데이터를 32문자 분량의 ASCII 코드로 간주하여 주변 툴에 표시합니다.



■ ASCII 코드 목록표

하위 4비트 \ 상위 4비트	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0			Sp	0	@	P	‘	p				ー	タ	ミ		
1			!	1	A	Q	a	q			。	ア	チ	ム		
2			”	2	B	R	b	r			「	イ	ツ	メ		
3			#	3	C	S	c	s			」	ウ	テ	モ		
4			$	4	D	T	d	t			、	エ	ト	ヤ		
5			%	5	E	U	e	u			・	オ	ナ	ユ		
6			&	6	F	V	f	v			ヲ	カ	ニ	ヨ		
7			’	7	G	W	g	w			ァ	キ	ヌ	ラ		
8			(	8	H	X	h	x			ィ	ク	ネ	リ		
9			)	9	I	Y	i	y			ゥ	ケ	ノ	ル		
A			*	:	J	Z	j	z			ェ	コ	ハ	レ		
B			+	;	K	[	k	{			ォ	サ	ヒ	ロ		
C			,	<	L	¥	l	|			ャ	シ	フ	ワ		
D			ー	=	M	]	m	}			ュ	ス	ヘ	ン		
E			.	>	N	^	n	~			ョ	セ	ホ	゛		
F			/	?	O	_	o				ッ	ソ	マ	°		

시계 기능용 명령

니모닉	명령 이름	펑션 번호	페이지
CADD	캘린더 가산	730	3-852
CSUB	캘린더 감산	731	
SEC	시분초→초 변환	065	3-856
HMS	초→시분초 변환	066	3-858
DATE	시계 보정	735	3-860

CADD/CSUB

명령 이름	니모닉	동작 옵션	펑션 번호	기능
캘린더 가산	CADD	@CADD	730	시각 데이터에 시간 데이터를 가산합니다.
캘린더 감산	CSUB	@CSUB	731	시각 데이터에서 시간 데이터를 감산합니다.

	CADD	CSUB
심볼	CADD S1 S1: 피가산 데이터(시각) 하위 CH 번호 S2 S2: 가산 데이터(시간) 하위 CH 번호 D D: 연산 결과(시각) 출력 하위 CH 번호	CSUB S1 S1: 피감산 데이터(시각) 하위 CH 번호 S2 S2: 감산 데이터(시간) 하위 CH 번호 D D: 연산 결과(시각) 출력 하위 CH 번호

특정 영역 내의 사용 여부

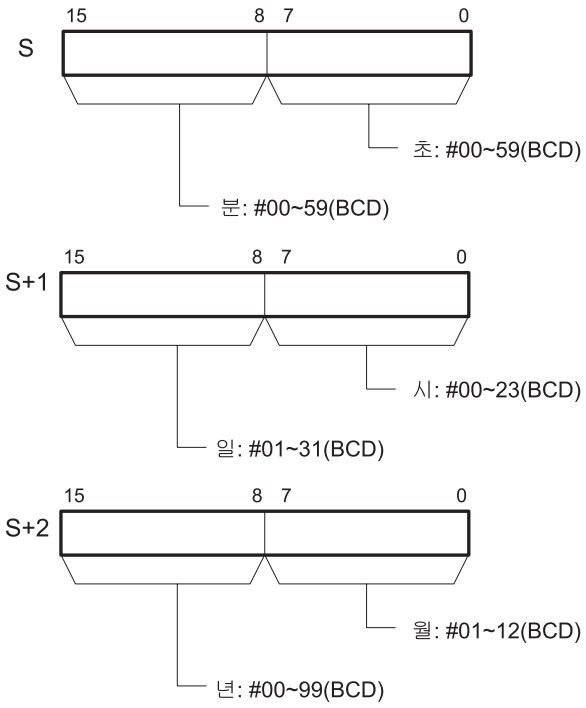
영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

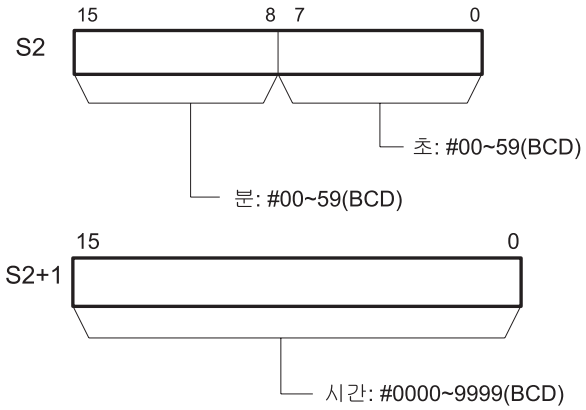
피연산자	내용	데이터 형식	크기
S1	CADD: 피가산 데이터(시각) 하위 CH 번호 CSUB: 피감산 데이터(시각) 하위 CH 번호	WORD	3
S2	CADD: 가산 데이터(시간) 하위 CH 번호 CSUB: 감산 데이터(시간) 하위 CH 번호	DWORD	2
D	연산 결과(시각) 출력 하위 CH 번호	WORD	3

· CADD

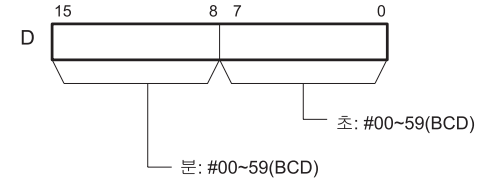
S1: 피가산 데이터(시각) 하위 CH 번호

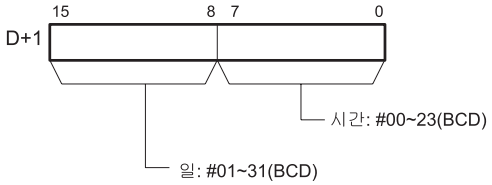


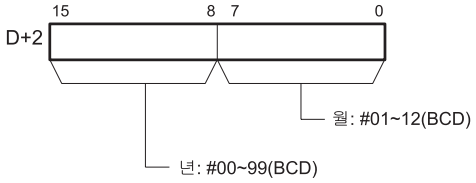
S2: 가산 데이터(시간) 하위 CH 번호



D: 연산 결과(시각) 출력 하위 CH 번호

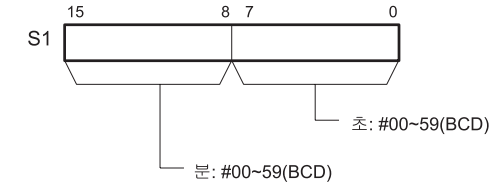


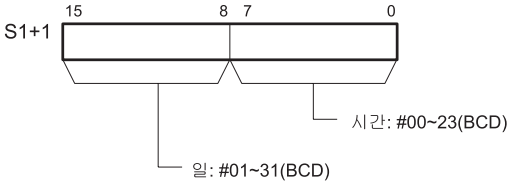


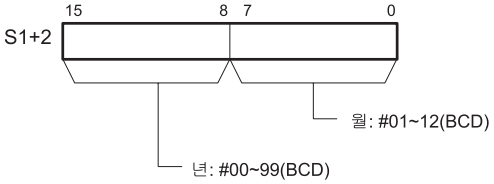


· CSUB

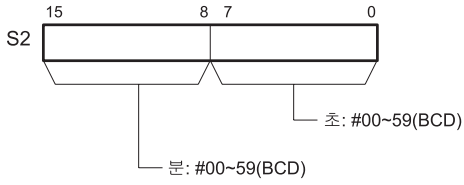
S1: 피감산 데이터(시각) 하위 CH 번호

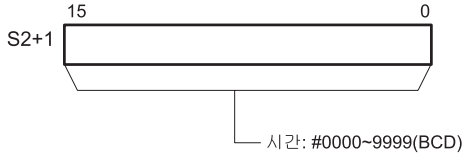




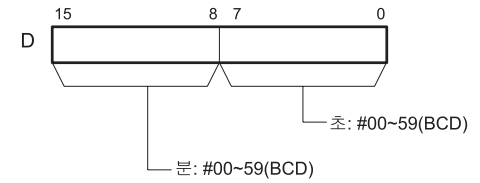


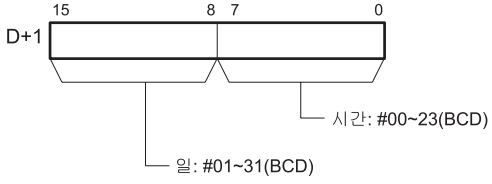
S2: 감산 데이터(시간) 하위 CH 번호

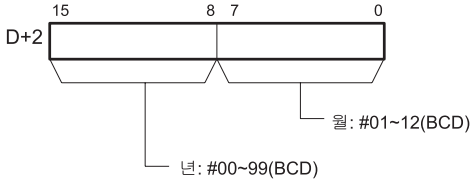




D: 연산 결과(시각) 출력 하위 CH 번호







CADD/CSUB

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S1	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S2											○							
D											–							

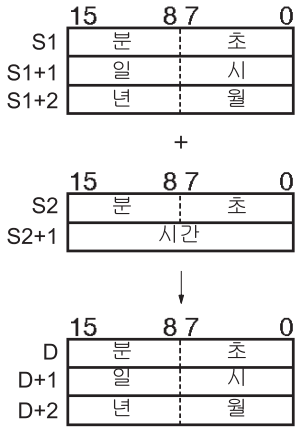
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S1의 시각 데이터가 범위 밖일 때 ON · S2의 시간 데이터가 범위 밖일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· CSUB 명령에서 연산 결과가 0일 때 ON · 그 이외일 때 OFF

기능

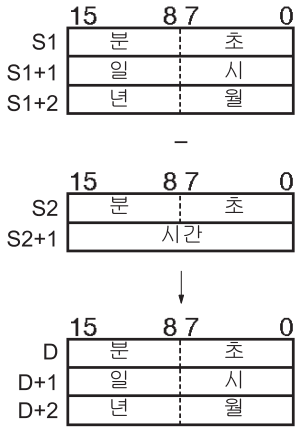
■ CADD

S1로 지정한 시각 데이터(연, 월, 일, 시, 분, 초)와 S2로 지정한 시간 데이터(시간, 분, 초)를 가산하여 결과를 시각 데이터(연, 월, 일, 시, 분, 초)로 D에 출력합니다.



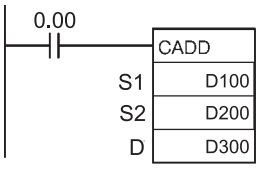
■ CSUB

S1로 지정한 시각 데이터(연, 월, 일, 시, 분, 초)에서 S2로 지정한 시간 데이터(시간, 분, 초)를 감산하여 결과를 시각 데이터(연, 월, 일, 시, 분, 초)로 D에 출력합니다.

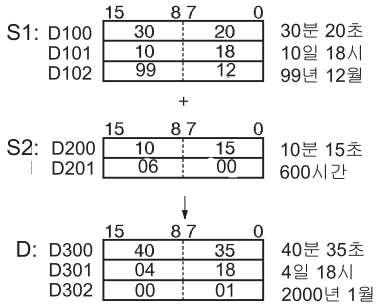


샘플 프로그램

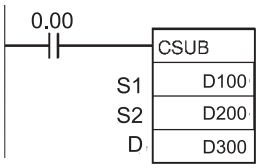
■ CADD



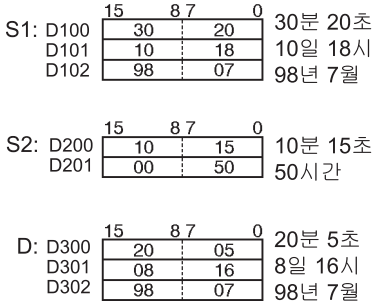
0.00이 ON이면 D100~D102의 시각 데이터(연월일시분초)와 D200~D201의 시간 데이터(시간 및 분초)를 더해서 D300~D302의 시각 데이터(연월일시분초)에 저장합니다.



■ CSUB

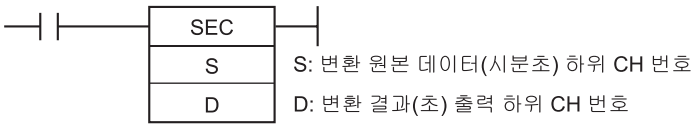


0.00이 ON이면 D100~D102의 시각 데이터(연월일시분초)에서 D200~D201의 시간 데이터(시간 및 분초)를 빼서 D300~D302의 시각 데이터(연월일시분초)에 저장합니다.



SEC

명령 이름	니모닉	동작 옵션	평션 번호	기능
시분초→초 변환	SEC	@SEC	065	시분초 데이터를 초 데이터로 변환합니다.

심볼	SEC
	

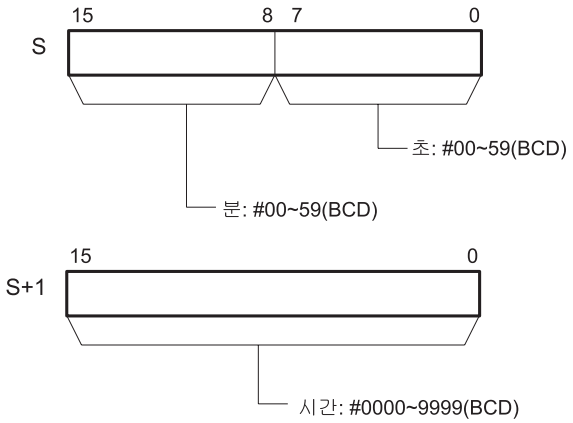
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

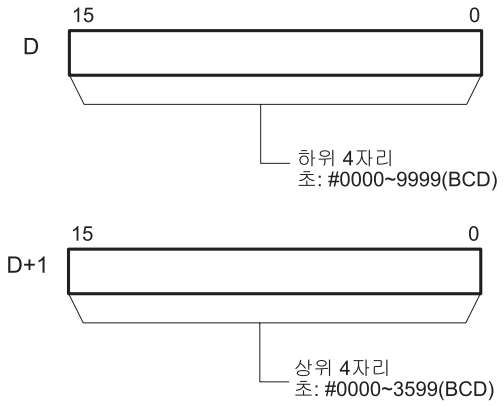
피연산자 설명

피연산자	내용	데이터 형식	크기
S	변환 원본 데이터(시분초) 하위 CH 번호	DWORD	2
D	변환 결과(초) 출력 하위 CH 번호	DWORD	2

S: 변환 원본 데이터(시분초) 하위 CH 번호



D: 변환 결과(초) 출력 하위 CH 번호



주: S~S+1, D~D+1은 각각 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
D											–							

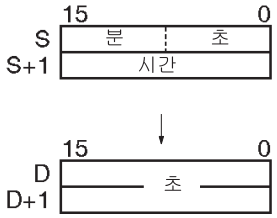
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S+1(시간 데이터)이 BCD 데이터가 아닐 때 ON · S의 분 데이터(비트 08~15)가 00~59의 BCD 데이터가 아닐 때 ON · S의 초 데이터(비트 00~07)가 00~59의 BCD 데이터가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF

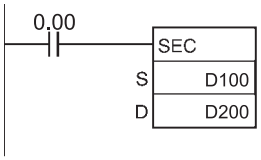
기능

S로 지정한 시분초 데이터(BCD 8자리)를 초 데이터(BCD 8자리)로 변환하여 결과를 D+1, D에 출력합니다.

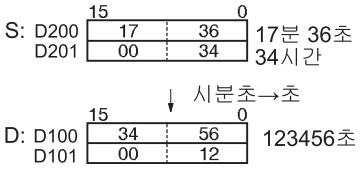
시분초 데이터의 최대값은 9999시간 59분 59초(35,999,999초)입니다.



샘플 프로그램

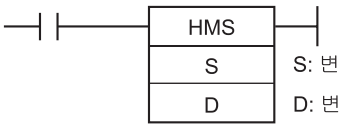


0.00이 ON이면 D200~D201의 시분초 데이터(예: 34시간 17분 36초)를 초 데이터로 변환하여 D100~D101에 저장합니다.



HMS

명령 이름	니모닉	동작 옵션	평션 번호	기능
초→시분초 변환	HMS	@HMS	066	초 데이터를 시분초 데이터로 변환합니다.

심볼	HMS
	 S: 변환 원본 데이터(초) 하위 CH 번호 D: 변환 결과(시분초) 출력 하위 CH 번호

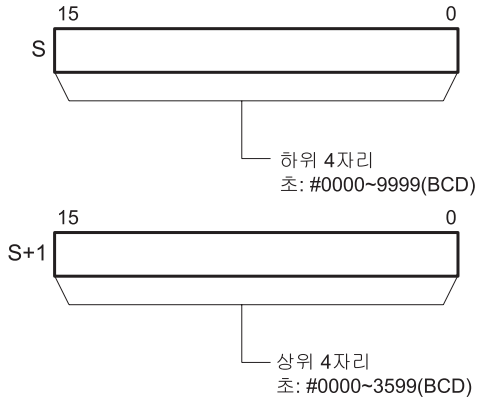
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

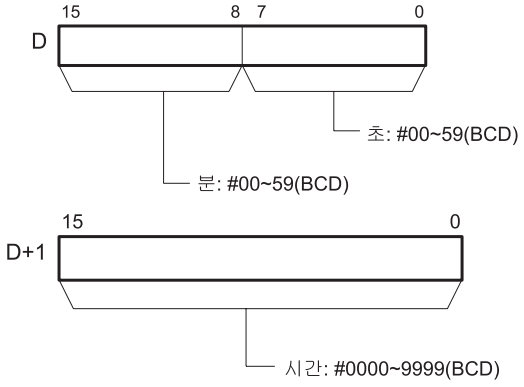
피연산자 설명

피연산자	내용	데이터 형식	크기
S	변환 데이터(초) 하위 CH 번호	DWORD	2
D	변환 결과(시분초) 출력 하위 CH 번호	DWORD	2

S: 변환 데이터(초) 하위 CH 번호



D: 변환 결과(시분초) 출력 하위 CH 번호



주: S~S+1, D~D+1은 각각 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	–	–	○	–	–	–	–
D											–							

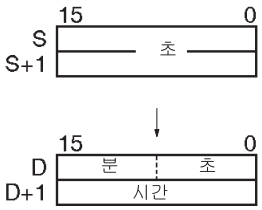
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S+1, S의 데이터가 0~35999999의 BCD 데이터가 아닐 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF

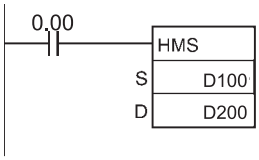
기능

S로 지정한 초 데이터(BCD 8자리)를 시분초 데이터 (BCD 8자리)로 변환하여 결과를 D+1, D에 출력합니다.

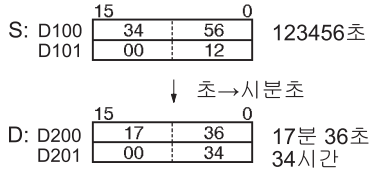
초 데이터의 최대값은 35.999.999초(9999시간 59분 59초)입니다.



샘플 프로그램

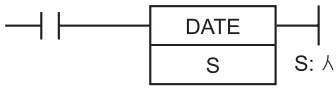


0.00이 ON이면 D100~D101의 초 데이터(예: 1234.56초)를 시분초 데이터로 변환하여 D200~D201에 저장합니다.



DATE

명령 이름	니모닉	동작 옵션	평션 번호	기능
시계 보정	DATE	@DATE	735	지정된 시계 데이터에 따라 내부 시계의 값을 변경합니다.

심볼	DATE
	

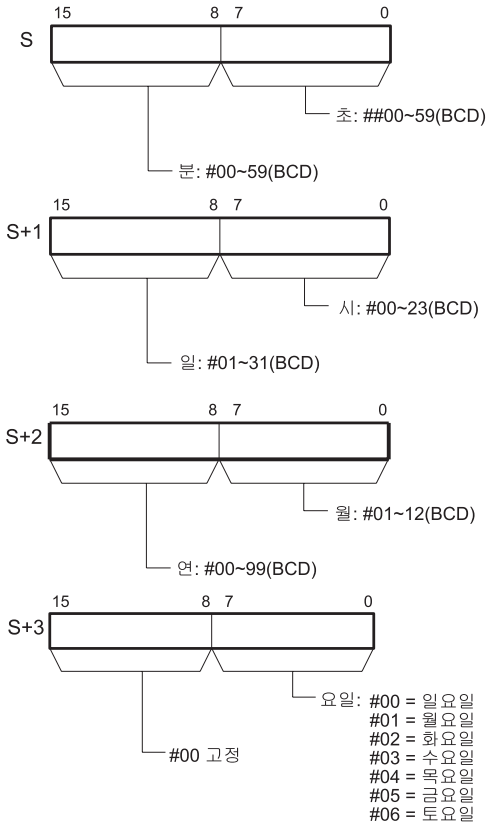
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	시계 데이터 하위 CH 번호	LWORD	4

S: 시계 데이터 하위 CH 번호



주: S~S+3은 각각 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–

조건 플래그의 동작

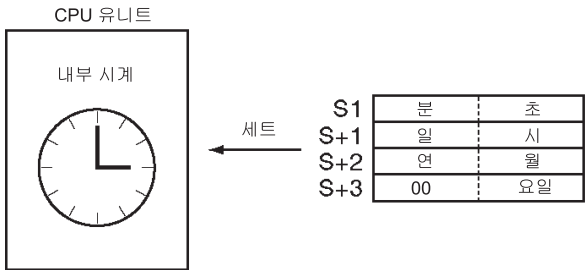
이름	라벨	내용
에러 플래그	P_ER	· S+3, S+2, S+1, S의 데이터가 범위 밖일 때 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

이름	주소	내용
시계 데이터	A351 ~A354CH	A351.00~A351.07: 초(00~59) (BCD) A351.08~A351.15: 분(00~59) (BCD) A352.00~A352.07: 시(00~23) (BCD) A352.08~A352.15: 일(01~31) (BCD) A353.00~A353.07: 월(01~12) (BCD) A353.08~A353.15: 년(00~99) (BCD) A354.00~A354.07: 요일(00~06) (BCD) 00: 일요일, 01: 월요일, 02: 화요일, 03: 수요일, 04: 목요일, 05: 금요일, 06: 토요일

기능

S~S+3으로 지정한 시계 데이터(4채널)에 따라 내부 시계의 값을 변경합니다. 변경된 값은 즉시 특수 보조 릴레이의 시계 데이터 영역(A351~354CH)에 반영됩니다.



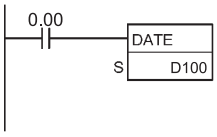
힌트

· 시계 데이터는 기타 주변 툴(프로그래밍 콘솔 포함) 및 FINS 명령의 시간 정보 쓰기(#0702)를 통해 설정할 수 있습니다.

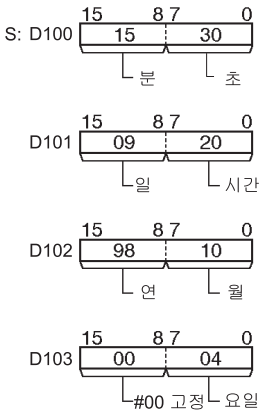
사용상의 주의 사항

· 일시를 실제 달력에 없는 날짜로 설정해도 에러가 발생하지 않습니다(예: 11월 31일 등).

샘플 프로그램



0.00이 ON이면 내부 시계를 1998년 10월 9일 20시 15분 30초, 목요일로 설정합니다.



디버그 처리 명령

니모닉	명령 이름	평션 번호	페이지
TRSM	트레이스 메모리 샘플링	045	3-864

TRSM

명령 이름	니모닉	동작 옵션	평션 번호	기능
트레이스 메모리 샘플링	TRSM	–	045	이 명령을 실행하면 접점 또는 채널 데이터를 샘플링하여 트레이스 메모리에 저장합니다.

심볼	TRSM
	

특정 영역 내의 사용 여부

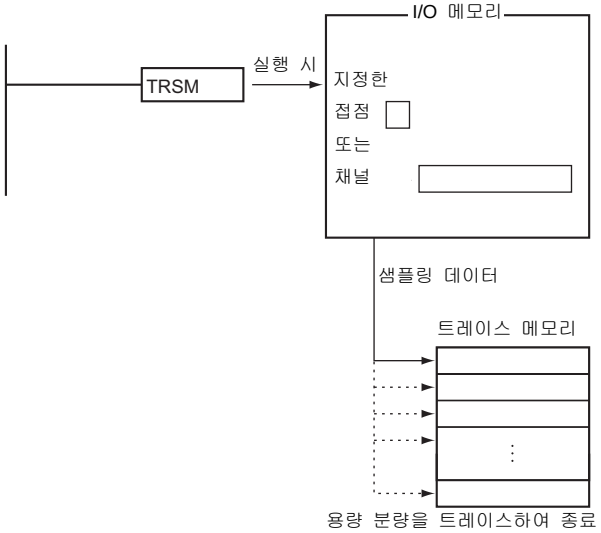
영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

조건 플래그의 동작

없음

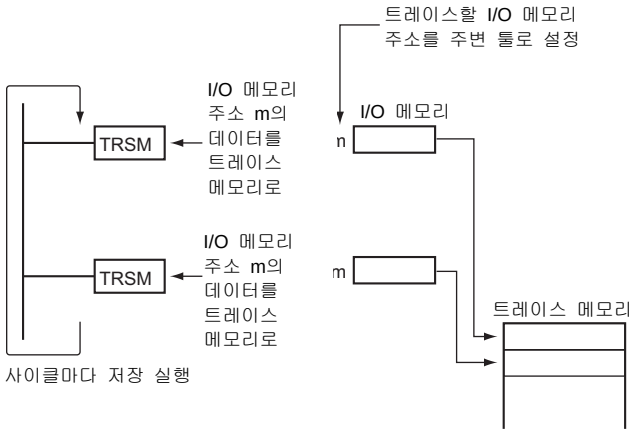
기능

이 명령을 실행할 때마다, 주변 툴로 미리 지정해 둔 I/O 메모리가 있는 접점 또는 채널 데이터의 현재값을 읽어 트레이스 메모리에 순서대로 저장합니다. 트레이스 메모리 용량에 해당하는 분량의 데이터를 읽으면 읽기를 종료합니다. 읽은 트레이스 메모리 내의 데이터는 필요할 때 주변 툴에서 모니터링할 수 있습니다.



- 이 명령의 기능은 데이터 트레이스 실행 시 샘플링하는 타이밍(샘플링 주기)을 지정하는 것뿐입니다. 그 밖의 설정 및 데이터 트레이스 실행 조작은 모두 주변 툴을 통해 실행됩니다.
- 이 명령은 입력 조건(전원 흐름)이 필요하지 않습니다. 항상 ON 입력으로 사용하십시오.
- 이 명령은 프로그램상의 각 조건(ON 조건)이 성립할 때 그 시점에서 특정 값의 상태를 파악하기 위해 사용합니다. 입력 조건이 사이클마다 ON이면 사이클마다 명령을 실행했을 때의 값을 트레이스 메모리에 저장합니다.

프로그램에 여러 개의 TRSM 명령을 배치할 수도 있습니다. 이 경우, 각 TRSM 명령을 실행할 때마다 같은 주소의 데이터를 트레이스 메모리에 저장합니다.



데이터 트레이스에 대한 자세한 내용은 주변 툴의 설명서를 참조하십시오.

힌트

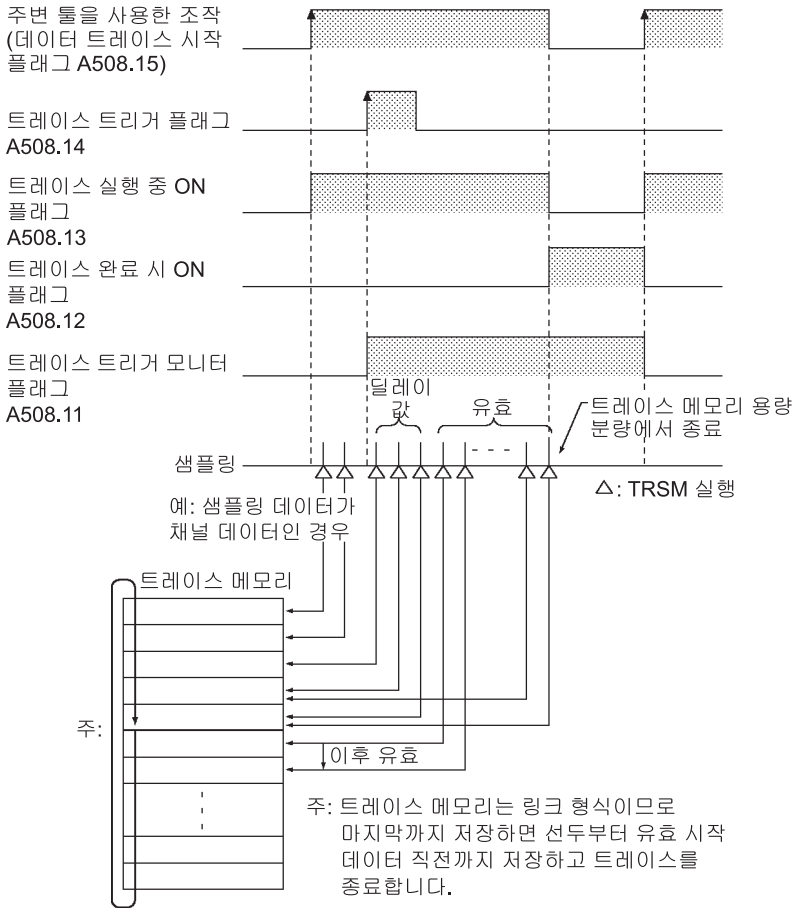
- 데이터 트레이스에는 이 TRSM 명령을 실행할 때마다의 샘플링 이외에 1사이클마다 실행되는 샘플링(모든 사용자 프로그램 실행 후의 샘플링) 및 정해진 주기(샘플링 주기)마다 실행되는 샘플링을 주변 툴로 지정할 수 있습니다.

- 주변 툴에서 사용 순서 개요는 다음과 같습니다.

1. 주변 툴로 다음 파라미터를 설정합니다.
   1) 샘플링 접점 번호 또는 샘플링 채널 번호
   2) 트리거 조건: 트레이스 메모리에 저장된 데이터를 유효화하는 조건을 설정합니다. 다음 3가지 중 하나를 지정합니다.
      - 트레이스 트리거 플래그 ON일 때
      - 특정 접점 기동 시
      - 특정 채널이 설정값과 일치할 때
   3) 샘플링 주기를 'TRSM 명령 실행 시'로 설정
   4) 딜레이값
2. 주변 툴로 실행 시작 조작을 수행하면 데이터 트레이스 시작 플래그(A508.15)가 0(OFF)→1(ON)이 되고 TRSM 명령 실행 시점의 샘플링이 실행되어 샘플링 데이터가 트레이스 메모리에 저장되기 시작합니다. 이와 동시에 트레이스 실행 중 ON 플래그(A508.13)가 ON이 됩니다.
3. 트리거 조건(트레이스 트리거 플래그/접점 기동/채널 값의 일치 중 하나)이 성립하면 그 시점부터 딜레이 횟수만큼 뒤 또는 앞의 시점(TRSM 명령 실행 시점)의 샘플링 데이터가 유효하게 됩니다. 동시에, 트레이스 트리거 모니터 플래그(A508.11)가 1(ON)이 됩니다.
4. 트레이스 메모리 용량에 해당하는 분량의 TRSM 명령을 실행하고 트레이스를 종료합니다. 트레이스가 종료되면 트레이스 완료 시 ON 플래그(A508.12)가 1(ON)이 되고 트레이스 실행 중 ON 플래그(A508.13)가 0(OFF)이 됩니다.
5. 주변 툴로 트레이스 메모리의 내용을 모니터링합니다.

TRSM

■ 타임 차트



사용상의 주의 사항

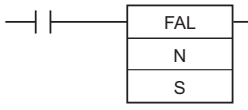
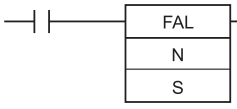
- 데이터 트레이스 중일 때 이외인 경우 또는 데이터 트레이스의 파라미터 설정에서 샘플링 주기가 TRSM 명령 실행 시 이외인 경우 NOP 처리됩니다.
- 사용자 프로그램 내에서 데이터 트레이스 시작 플래그(A508.15)를 ON/OFF시키지 마십시오.
- JMP-JME 사이에서 TRSM 명령을 사용한 경우, JMP 조건이 OFF이면 TRSM 명령이 실행되지 않습니다.

고장 진단 명령

니모닉	명령 이름	펑션 번호	페이지
FAL	운전 계속 고장 진단	006	3-868
FALS	운전 정지 고장 진단	007	3-874
FPD	고장 지점 검출	269	3-879

FAL

명령 이름	니모닉	동작 옵션	평션 번호	기능
운전 계속 고장 진단	FAL	@FAL	006	사용자 정의 운전 계속 이상을 등록 또는 해제합니다.

심볼	FAL	
	사용자 정의 운전 계속 이상을 등록/해제할 때	시스템에 의한 운전 계속 이상을 등록할 때
	 N: FAL 번호 S: 메시지 저장 대상 하위 CH 번호 메시지가 없을 때는 상수 (#0000~FFFF 중 하나)	 N: FAL 번호(=A529CH 내의 값) S: 발생시킬 고장 코드/이상 내용 저장 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	FAL 번호	상수만	1
S	메시지 저장 대상 하위 CH 번호 또는 메시지가 없을 때는 상수/발생시키는 고장 코드	WORD	가변

■ 사용자 정의 운전 계속 이상을 등록/해제할 때

주: 피연산자 N의 내용과 특수 보조 릴레이 A529CH(시스템 이상 발생 FAL/FALS 번호) 내의 값이 달라야 합니다.

	운전 계속 이상 등록	운전 계속 이상 해제
N	1~511: FAL 번호(FALS 번호와 공용)	0
S	CH 지정: S~S+7의 16문자 분량(ASCII 코드)을 주변 툴에 표시 #0000~FFFF: 메시지 없음	#FFFF: 일괄 해제 &1~511(10진수) #0001~01FF(16진수): 해제 대상 FAL 번호 위 이외의 값 또는 CH 지정: 중요도가 가장 높은 운전 계속 이상을 1개 해제

■ 시스템에 의한 운전 계속 이상을 등록할 때

주: 피연산자 N의 내용과 특수 보조 릴레이 A529CH(시스템 이상 발생 FAL/FALS 번호) 내의 값이 일치해야 합니다.

	운전 계속 이상 등록
N	1~511: FAL 번호(FALS 번호와 공용)
S	발생시키는 고장 코드(뒤의 설명 참조)
S+1	이상 내용 설정(뒤의 설명 참조)

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
S	○	○	○	○	○	○	○	○	○	○				○				

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· N의 데이터가 0~511의 범위 안에 없을 때 ON · 시스템에 의한 운전 계속 이상을 발생시킬 때, 발생시키는 고장 코드/이상 내용 저장 하위 CH 번호에 저장된 이상 코드(정지 이상, 계속 이상)가 해당되지 않는 경우 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

■ 사용자 정의 운전 계속 이상 관련

이름	주소	내용
FAL 이상 플래그	A402.15	FAL 명령이 실행되었을 때 1(ON)이 됩니다.
실행 FAL 번호	A360.01~ A391.15	FAL 명령이 실행되었을 때 해당 비트가 1(ON)이 됩니다. A360.01~A391.15가 FAL1~511에 대응합니다.

■ 시스템에 의한 운전 계속 이상 관련

이름	주소	내용
시스템 이상 발생 FAL/FALS 번호	A529CH	FAL 명령 또는 FALS 명령 실행을 사용하여 시스템 이상을 고의로 발생시키는 경우, 더미로 사용할 FAL/FALS 번호를 설정합니다. &1~511 또는 #0001~01FF

■ 사용자 정의/시스템 공통 운전 계속 이상 관련

이름	주소	내용
이상 내역 저장 영역	A100~ 199CH	이상(FAL 명령 실행 구성)이 발생하면 최신 20개의 고장 코드, 이상 발생 시각이 저장됩니다.
고장 코드	A400CH	이상(FAL 명령 실행 구성)이 발생하면 대응하는 고장 코드가 저장됩니다. #4101~42FF가 FAL1~511(16진수 #001~1FF)에 대응합니다. 주: 동시에 발생하면 중요도가 가장 높은 이상의 코드를 저장합니다.

운전 계속 이상 해제

■ 사용자 정의 운전 계속 이상 해제

N으로 지정하는 FAL 번호를 0으로 하면 운전 계속 이상이 해제됩니다.

S로 지정하는 데이터에 따라 다음과 같이 처리됩니다.

S	내용
10진수 &1~511 또는 16진수 #0001~01FF	해당 번호의 FAL 이상을 해제합니다.
16진수 #FFFF	모든 운전 계속 이상(시스템의 운전 계속 이상 포함)을 해제합니다.
16진수 #0200~FFFE 및 CH 지정	발생한 운전 계속 이상(시스템의 운전 계속 이상 포함) 중 중요도가 가장 높은 이상을 하나 해제합니다. 복수의 FAL 이상이 발생한 경우 가장 빠른 번호의 FAL을 해제합니다.

■ 시스템에 의한 운전 계속 이상 해제

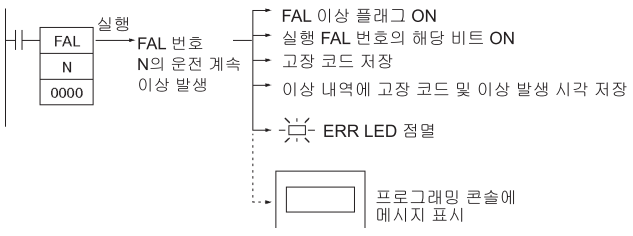
시스템에서 고의로 발생시킨 운전 계속 이상을 해제하는 방법은 ① PLC 본체의 전원 OFF→ON 또는 ② (PLC 본체의 전원이 ON 상태인 경우) 해당하는 이상이 실제로 발생했을 때와 같은 처리를 수행합니다.

기능

■ 사용자 정의 운전 계속 이상의 등록

FAL/FALS 번호	1~511(16진수 #001~1FF)
FAL 고장 코드	16진수 #4101~#42FF
특수 보조 릴레이의 실행 FAL 번호	A360.01~A391.15

N으로 지정한 데이터가 1~511이고 특수 보조 릴레이 A529CH(시스템 이상 발생 FAL/FALS 번호) 내의 값과 일치하지 않을 때 입력 조건이 성립하면 FAL 번호 N의 이상(운전 계속 이상)이 발생한 것으로 판단하여 아래와 같은 동작을 수행합니다.



FAL

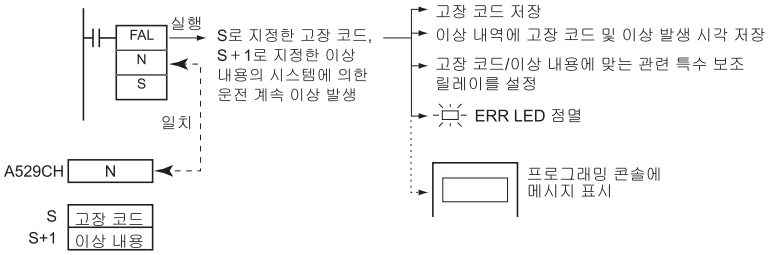
1) 특수 보조 릴레이 FAL 이상 플래그(A402.15)를 ON합니다(운전은 계속).
2) N으로 지정한 데이터가 1~511일 때는 특수 보조 릴레이의 실행 FAL 번호(A360~A391)의 해당 비트를 ON합니다(A360.01~A391.15: FAL 번호 1~511에 대응).
3) 특수 보조 릴레이의 고장 코드(A400)에 고장 코드(FAL 번호 1~511(10진수)에 대응되는 고장 코드 #4101~42FF)를 설정합니다.
4) 특수 보조 릴레이의 이상 내역 저장 영역(A100~199)에 고장 코드 및 이상 발생 시각을 저장합니다.

   주: 단, PLC 시스템 설정의 '이상 내역에 FAL 등록 안 함'을 선택하면 이상 내역 저장 영역에 고장 코드 및 이상 발생 시각을 저장하지 않습니다.
5) CPU 유니트의 ERR LED가 점멸하고 운전은 계속됩니다.
6) S에 의해 메시지가 CH 지정된 경우는 메시지를 등록(주변 툴에 표시)합니다.

   주: 이 명령으로 등록한 이상보다 중요도가 더 높은 이상(시스템에 의한 운전 정지 이상, FALS 번호 실행에 의한 운전 정지 이상 포함)이 동시에 발생한 경우는 고장 코드(A400)에 그 이상 코드가 설정됩니다.

■ 시스템에 의한 운전 계속 이상의 등록

N으로 지정한 데이터가 1~511이고 특수 보조 릴레이 A529CH(시스템 이상 발생 FAL/FALS 번호) 내의 값과 일치할 때 입력 조건이 성립하면 S, S+1로 지정한 고장 코드 및 이상 내용에 해당하는 시스템에 의한 운전 계속 이상을 고의로 발생시킵니다. 동시에 아래와 같은 동작을 수행합니다.



1) 특수 보조 릴레이의 고장 코드(A400)에 고장 코드를 설정합니다.
2) 특수 보조 릴레이의 이상 내역 저장 영역(A100~199)에 고장 코드 및 이상 발생 시각을 저장합니다.
3) 고장 코드/이상 내용에 맞는 관련 특수 보조 릴레이를 설정합니다.
4) CPU 유니트의 ERR LED가 점멸하고 운전은 계속됩니다.
5) 고의로 발생시킨 시스템에 의한 운전 계속 이상의 에러 메시지를 프로그래밍 콘솔에 표시합니다.

참고: FAL 명령을 통한 '시스템에 의한 운전 계속 이상 발생'은 프로그램을 디버그하는 경우 등에 사용합니다. 예를 들어, 고의로 이상을 발생시켜 PT(프로그래머블 터미널) 등의 MMI에 에러 메시지가 정확하게 표시되는지 여부를 확인할 수 있습니다.

주 1: 특수 보조 릴레이 A529CH(시스템 이상 발생 FAL/FALS 번호) 내의 값은 시스템에 의한 운전 계속/정지 이상을 고의로 발생시킬 때 더미로 사용하는 FAL/FALS 번호(FAL, FALS, FPD 명령에서 공통)입니다. (이 값은 더미이므로 실행 FAL 번호, 고장 코드에는 반영되지 않습니다.)
복수의 시스템에 의한 운전 계속/정지 이상을 고의로 발생시키는 경우는 A529CH 내의 같은 값을 N으로 지정하고, 다른 S, S+1의 내용을 통해 복수의 FAL/FALS/FPD 명령을 실행하십시오.

주 2: 이 명령으로 고의로 발생시킨 이상보다 중요도가 더 높은 이상(시스템에 의한 운전 정지 이상, FALS 번호 실행에 의한 운전 정지 이상 포함)이 동시에 발생한 경우는 고장 코드(A400)에 그 이상 코드가 설정됩니다.

주 3: 이 명령으로 고의로 발생시킨 이상을 해제하는 경우는 PLC 본체의 전원을 OFF로 한 후 다시 공급(ON)하십시오. PLC 본체의 전원을 ON 상태 그대로 둔 경우, 해당하는 이상이 실제로 발생했을 때와 동일한 처리를 수행해야 합니다.
자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 설치편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 하드웨어편'을 참조하십시오.

■ S, S+1 지정 방법

이상 명칭	S(16진수)	S+1
인터럽트 태스크 이상	#008B	비트 15 = OFF: 인터럽트 태스크 이상 비트 00~14: 이상 발생 인터럽트 태스크 번호 비트 15 = ON: 고기능 I/O 유니트 다중 리프레시 비트 00~14: 다중 리프레시 대상 고기능 I/O 유니트의 기기 번호
기본 I/O 이상	#009A	이상 유니트 장착 위치 비트 08~15: 기본 I/O 유니트 이상 랙 번호(BIN) 비트 00~07: 기본 I/O 유니트 이상 슬롯 번호(BIN)
PLC 시스템 설정 이상	#009B	PLC 시스템 설정 이상 위치 #0000~FFFF
I/O 조회 이상	#00E7	−(일정치 않음)
INNER 보드 계속 이상	#02F0	이상 요인 비트 00~03: 무효 비트 04~15: INNER 보드로 정의한 이상
CPU 고기능 유니트 이상	#0200	CPU 고기능 유니트 이상 기기 번호 10진수 &0~15 또는 16진수 #0000~000F
고기능 I/O 유니트 이상	#0300	고기능 I/O 유니트 이상 기기 번호 10진수 &0~95 또는 16진수 #0000~005F 16진수 #00FF(호기 미설정)

이상 명칭	S(16진수)	S+1
SYSBUS 이상	#00A0	SYSBUS 마스터 이상 호기 번호 10진수 &0~1 또는 16진수 #0000~0001
배터리 이상	#00F7	16진수 #0000
CPU 고기능 유니트 설정 이상	#0400	CPU 고기능 유니트 이상 유니트 번호 10진수 &0~15 또는 16진수 #0000~000F
고기능 I/O 유니트 설정 이상	#0500	고기능 I/O 유니트 이상 기기 번호 10진수 &0~95 또는 16진수 #0000~005F

■ FAL 명령 실행 시의 이상 내역 등록 유무 지정

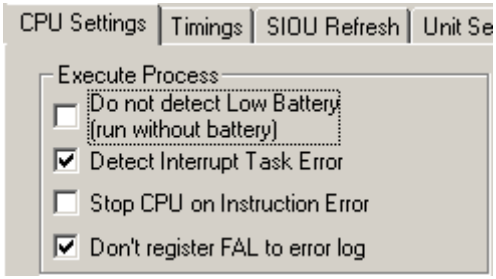
FAL 명령을 실행했을 때 사용자 정의 운전 계속 이상이 발생하지만 특수 보조 릴레이의 이상 내역 저장 영역(A100~199CH)에는 이상(고장 코드 및 이상 발생 시각)을 등록하지 않을 수 있습니다(주).

주: 이때, 특수 보조 릴레이 A402.15의 FAL 이상 플래그, A360~391CH의 실행 FAL 번호 및 A400CH의 고장 코드에는 반영됩니다.

이 기능은 이상 내역에 시스템에 의한 이상만 저장하려는 경우 사용합니다. 예를 들어, FAL 명령을 많이 사용하는 활용 사례에서 디버그 중에 이상 내역 저장 영역이 FAL 명령에 의한 이상으로 가득 차는 것을 방지할 경우 등에 사용합니다.

다음과 같은 PLC 시스템 설정으로 지정합니다.

・ PLC 시스템 설정(CX-Programmer의 경우)



・ PLC 시스템 설정(프로그래밍 콘솔의 경우)

프로그래밍 콘솔 주소	채널	+129
	비트	15
이름	FAL 이상에 대한 이상 내역 등록 유무 지정	
설정 데이터	0: 이상 내역에 등록 1: 이상 내역에 등록 안 함	
초기값	0: 이상 내역에 등록	
CPU 유니트 반영 타이밍	사이클마다(FAL 이상 발생 시)	

주: 위의 PLC 시스템 설정에서 '이상 내역에 등록 안 함'으로 설정한 경우에도 다음의 이상은 이상 내역에 등록됩니다.

- FALS 명령 실행에 의한 운전 정지 이상
- 시스템에 의한 운전 계속 이상
- 시스템에 의한 운전 정지 이상
- FAL(FPD) 명령 실행에 의해 고의로 발생시킨 시스템 운전 계속 이상
- FALS 명령 실행에 의해 고의로 발생시킨 시스템 운전 정지 이상

■ 사용자 정의 운전 계속 이상 발생 시의 메시지 표시

・ FAL 명령에서 명령 실행과 동시에 주변 툴에 메시지를 표시하려면 S에 CH 번호를 설정합니다. (메시지 표시가 필요 없는 경우, S에 상수 #0000~FFFF 중 하나를 설정하십시오.)

・ FAL 명령 실행에 의해 메시지가 등록됩니다. 메시지가 등록되면 프로그래밍 콘솔을 연결할 때 메시지가 표시됩니다.

・ S~S+7CH의 8CH에 ASCII 코드로 16문자 분량의 문자를 설정할 수 있습니다.

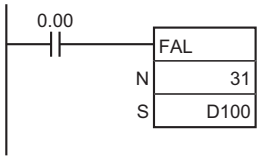
・ 메시지 표시 순서는 상위 바이트→하위 바이트입니다.

・ 메시지 구분은 NUL(#00)입니다. NUL을 제외하고 최대 16문자까지 표시할 수 있습니다.

・ FAL 명령 실행 후 메시지를 설정한 CH의 내용을 변경하면 변경 내용에 따라 메시지도 변경됩니다.

샘플 프로그램

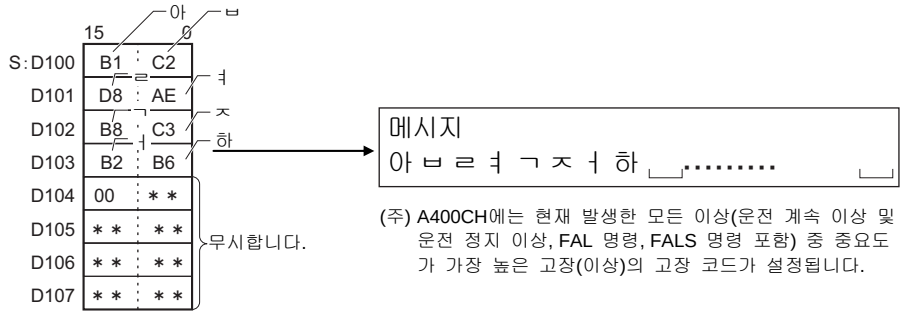
■ 등록할 때(N: 001~511 설정)



0.00이 ON이면 FAL 번호 31의 운전을 계속하는 고장(이상)이 발생했다고 판단하여 다음 처리를 실행합니다.

1) 특수 보조 릴레이 FAL 이상 플래그 A402.15를 ON
2) FAL 번호에 대응하는 특수 보조 릴레이 A361CH의 비트 15를 ON
3) 특수 보조 릴레이 고장 코드 A400에 고장 코드 #411F를 설정 (이 고장이 중요도가 가장 높을 때) (주)
4) 특수 보조 릴레이의 이상 내역 저장 영역(A100~199)에 고장 코드 및 이상 발생 시각을 저장
5) CPU 유니트의 ERR LED 점멸
6) D100~D107의 8CH 데이터를 16문자 ASCII 코드 데이터로 간주하고 주변 툴에 표시 (예: 압력 저하)

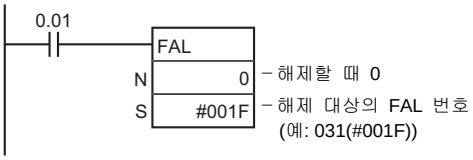
주: 메시지 표시가 필요 없는 경우, S에 상수(#0000~FFFF 중 하나)를 설정하십시오.



(주) A400CH에는 현재 발생한 모든 이상(운전 계속 이상 및 운전 정지 이상, FAL 명령, FALS 명령 포함) 중 중요도가 가장 높은 고장(이상)의 고장 코드가 설정됩니다.

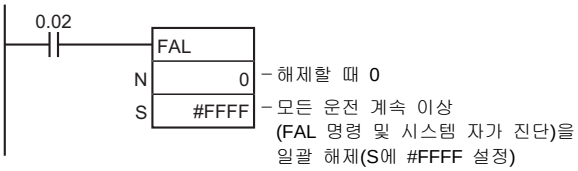
■ 해제할 때(N: 0 설정)

・ 발생한 대상 고장(이상)을 해제할 때(S: 해제 대상의 고장 코드 설정)



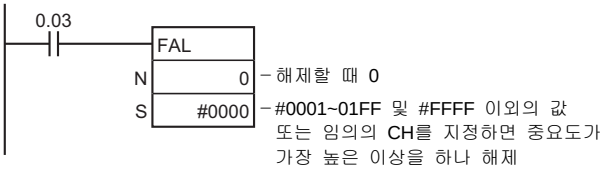
0.01이 ON이면 FAL 번호 31의 고장(이상)을 해제합니다.
A361CH의 비트 15가 OFF로 됩니다.
A402.15가 OFF로 됩니다.

・ 모든 운전 계속 이상(FAL 명령 및 시스템의 자가 진단에 의한 이상 포함)을 해제할 때(S: #FFFF 설정)



0.02가 ON이면 모든 운전 계속 이상을 일괄 해제합니다.
A360.01~A391.15의 모든 비트가 OFF로 됩니다.
A402.15가 OFF로 됩니다.

・ 현재 발생한 운전 계속 이상(FAL 명령 및 시스템의 자가 진단에 의한 이상 포함) 중 중요도가 가장 높은 이상을 하나 해제할 때(S: 위 이외의 설정)

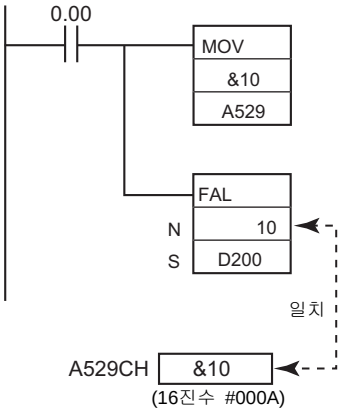


0.03이 ON이면 현재 발생한 운전 계속 이상 중 중요도가 가장 높은 이상을 하나 해제합니다.
A400CH의 고장 코드가 리셋됩니다.
FAL 명령에 의한 고장(이상)이 중요도가 가장 높을 때 A360.01~A391.15의 해당 비트가 OFF가 되고, A402.15가 OFF로 됩니다.

■ 시스템에 의한 운전 계속 이상을 발생시킬 때

0.00이 ON이면 CPU 고기능 유니트 설정 이상(이상 유니트 번호=1)을 발생시킵니다. 더미의 FAL 번호에는 10을 사용합니다(A529CH에 &10(16진수 #000A)을 저장해 둡니다). 동시에 다음의 처리를 수행합니다.

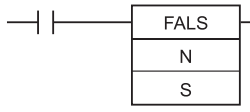
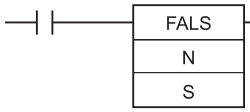
1) 특수 보조 릴레이 고장 코드 A400에 고장 코드 #0400을 설정(이 고장이 중요도가 가장 높을 때)
2) 특수 보조 릴레이의 이상 내역 저장 영역(A100~199)에 고장 코드 및 이상 발생 시각을 저장
3) 특수 보조 릴레이 A402.03(CPU 고기능 유니트 설정 이상 플래그)과 A427.01(CPU 고기능 유니트 설정 이상 유니트 번호 플래그의 비트 01이 유니트 번호 1에 대응)이 1(ON)이 됩니다.
4) CPU 유니트의 ERR LED 점멸
5) 프로그래밍 콘솔에 'CPU 고기능 설정 01'이 표시됩니다.



S : D200	# 0400	고장 코드: #0400(CPU 고기능 유니트 설정 이상)
D201	& 1	이상 유니트 번호: 1(16진수 #0001)

FALS

명령 이름	니모닉	동작 옵션	평션 번호	기능
운전 정지 고장 진단	FALS	–	007	사용자 정의 운전 정지 이상을 등록합니다.

심볼	FALS	
	사용자 정의 운전 계속 이상을 등록/해제할 때	시스템에 의한 운전 계속 이상을 등록할 때
	 N: FALS 번호 S: 메시지 저장 대상 하위 CH 번호 메시지가 없을 때는 상수 (#0000~FFFF 중 하나)	 N: FALS 번호 S: 발생시킬 고장 코드/이상 내용 저장 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	FALS 번호	상수만	1
S	메시지 저장 대상 하위 CH 번호 또는 메시지가 없을 때는 상수/발생시키는 고장 코드	WORD	가변

■ 사용자 정의 운전 정지 이상을 등록할 때

주: 피연산자 N의 내용과 특수 보조 릴레이 A529CH(시스템 이상 발생 FAL/FALS 번호) 내의 값이 달라야 합니다.

	운전 정지 이상 등록
N	1~511: FALS 번호(FAL 번호와 공용)
S	CH 지정: S~S+7의 16문자 분량(ASCII 코드)을 주변 툴에 표시 #0000~FFFF: 메시지 없음

■ 시스템에 의한 운전 정지 이상을 등록할 때

주: 피연산자 N의 내용과 특수 보조 릴레이 A529CH(시스템 이상 발생 FAL/FALS 번호) 내의 값이 일치해야 합니다.

	운전 정지 이상 등록
N	1~511: FALS 번호(FAL 번호와 공용)
S	발생시키는 고장 코드(뒤의 설명 참조)
S+1	이상 내용 설정(뒤의 설명 참조)

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
S	○	○	○	○	○	○	○	○	○	○				○				

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· N의 데이터가 1~511의 범위 안에 없을 때 ON · 시스템에 의한 운전 정지 이상을 발생시킬 때, 발생시키는 고장 코드/이상 내용 저장 하위 CH 번호에 저장된 이상 코드(정지 이상, 계속 이상)가 해당되지 않는 경우 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

■ 사용자 정의 운전 계속 이상 관련

이름	주소	내용
FAL 이상 플래그	A401.06	FALS 명령이 실행되었을 때 1(ON)이 됩니다.

■ 시스템에 의한 운전 정지 이상 관련

이름	주소	내용
시스템 이상 발생 FAL/FALS 번호	A529CH	FAL 명령 또는 FALS 명령 실행을 사용하여 시스템 이상을 고의로 발생시키는 경우, 더미로 사용할 FAL/FALS 번호를 설정합니다. 10진수 &1~511 또는 16진수 #0001~01FF

■ 사용자 정의/시스템 공통 운전 정지 이상 관련

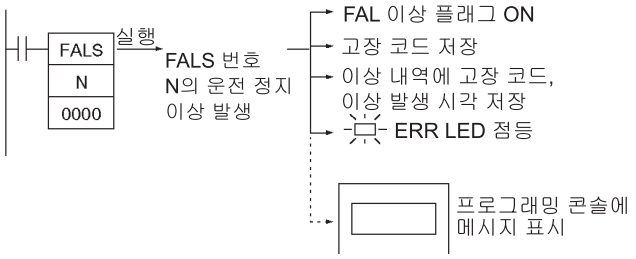
이름	주소	내용
이상 내역 저장 영역	A100~ 199CH	이상(FALS 명령 실행 포함)이 발생하면 최신 20개의 고장 코드, 이상 발생 시각이 저장됩니다.
고장 코드	A400CH	이상(FALS 명령 실행 포함)이 발생하면 대응하는 고장 코드가 저장됩니다. 16진수 #C101~C2FF가 FALS 1~511(16진수 #001~1FF)에 대응합니다. 주: 동시에 발생하면 중요도가 가장 높은 이상의 코드를 저장합니다.

기능

■ 사용자 정의 운전 정지 이상의 등록

FAL/FALS 번호	1~511(16진수 #001~1FF)
FAL 고장 코드	16진수 #C101~C2FF

FALS 번호 N이 특수 보조 릴레이 A529CH(시스템 이상 발생 FAL/FALS 번호) 내의 값과 일치하지 않을 때 입력 조건이 성립하면 FALS 번호 N의 사용자 정의 운전 정지 이상(운전을 정지시키는 이상)이 발생한 것으로 판단하여 다음의 동작을 수행합니다.



1) 특수 보조 릴레이 FALS 이상 플래그(A401.06)를 ON합니다(운전은 정지).
2) 특수 보조 릴레이의 고장 코드(A400)에 고장 코드(FALS 번호 1~511(16진수 #001~1FF)에 대응되는 고장 코드 16진수 #C101~C2FF)를 설정합니다. (단, 동시에 발생한 다른 고장 중에서 중요도가 가장 높은 경우만 저장합니다.)
3) 특수 보조 릴레이의 이상 내역 저장 영역(A100~199)에 고장 코드 및 이상 발생 시각을 저장합니다.
4) CPU 유니트의 ERR LED가 점등하고 운전을 정지합니다.
5) S에 의해 메시지가 CH 지정된 경우는 메시지를 등록(주변 툴에 표시)합니다.

주 1: 이 명령으로 등록한 이상보다 중요도가 더 높은 이상(시스템에 의한 운전 정지 이상 포함)이 동시에 발생한 경우는 고장 코드(A400)에 그 이상 코드가 설정됩니다.

주 2: 메시지 구분은 NUL(#00)입니다. NUL을 제외하고 최대 16문자까지 표시할 수 있습니다.

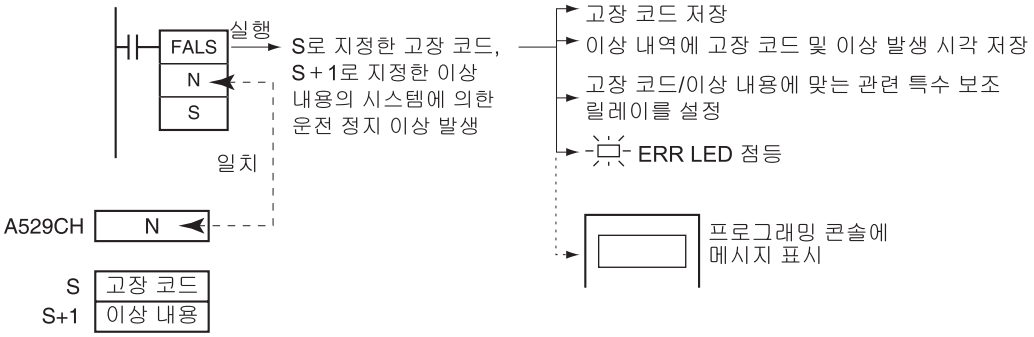
주 3: N의 데이터가 1~511의 범위에 없으면 에러가 되어 에러 플래그가 ON이 됩니다.

주 4: 사용자 정의 운전 정지 이상을 등록한 경우 I/O 메모리 및 출력 유니트의 출력 접점 상태는 다음과 같습니다.

		I/O 메모리	출력 유니트의 출력 접점
I/O 메모리 유지 플래그 (A500.12)	ON	유지	OFF
	OFF	유지	OFF

FALS

■ 시스템에 의한 운전 정지 이상의 등록



FALS 번호 N이 특수 보조 릴레이 A529CH(시스템 이상 발생 FAL/FALS 번호) 내의 값과 일치할 때 입력 조건이 성립하면 S로 지정한 고장 코드 및 S+1로 지정한 이상 내용에 해당하는 시스템에 의한 운전 정지 이상을 고의로 발생시킵니다. 동시에 아래와 같은 동작을 수행합니다.

1) 특수 보조 릴레이의 고장 코드(A400)에 고장 코드를 설정합니다.
2) 특수 보조 릴레이의 이상 내역 저장 영역(A100~199)에 고장 코드 및 이상 발생 시각을 저장합니다.
3) 고장 코드/이상 내용에 맞는 관련 특수 보조 릴레이를 설정합니다.
4) CPU 유니트의 ERR LED가 점등하고 운전을 정지합니다.
5) 고의로 발생시킨 시스템에 의한 운전 정지 이상의 에러 메시지를 프로그래밍 콘솔에 표시합니다.

주 1: 특수 보조 릴레이 A529CH(시스템 이상 발생 FAL/FALS 번호) 내의 값은 시스템 이상을 고의로 발생시킬 때 더미로 사용하는 FAL/FALS 번호(FAL, FALS, FPD 명령에서 공통)입니다. (이 값은 더미이므로 고장 코드에는 반영되지 않습니다.)
복수의 시스템 이상을 고의로 발생시키는 경우는 A529CH 내의 같은 값을 N으로 지정하고, 다른 S, S+1의 내용을 통해 FAL/FALS/FPD 명령을 실행하십시오.

주 2: 이 명령으로 고의로 발생시킨 이상보다 중요도가 더 높은 이상(시스템의 운전 정지 이상, FALS 번호 실행에 의한 운전 정지 이상 포함)이 동시에 발생한 경우는 고장 코드(A400)에 그 이상 코드가 설정됩니다.

주 3: 이 명령으로 고의로 발생시킨 이상을 해제하는 경우는 PLC 본체의 전원을 OFF로 한 후 다시 공급(ON)하십시오. PLC 본체의 전원을 ON 상태 그대로 둔 경우, 해당하는 이상이 실제로 발생했을 때와 동일한 처리를 수행해야 합니다.
자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 설치편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 하드웨어편'을 참조하십시오.

주 4: 시스템에 의한 운전 정지 이상을 고의로 발생시킨 경우, I/O 메모리 및 출력 유니트의 출력 접점 상태는 다음과 같습니다 (실제 시스템에 의한 운전 정지 이상과 동일).

		I/O 메모리	출력 유니트의 출력 접점
I/O 메모리 유지 플래그 (A500.12)	ON	유지	OFF
	OFF	클리어	OFF

■ S, S+1 지정 방법

이상 명칭	S	S+1
	고장 코드 설정(16진수)	이상 내용 설정
메모리 이상	#80F1	비트 00~09: 메모리 이상 발생 위치(아래에 기재) 비트 00: 사용자 프로그램 비트 04: PLC 시스템 설정 비트 05: 등록 I/O 테이블 비트 07: 라우팅 테이블 비트 08: CPU 고기능 유니트 시스템 설정 비트 09: 메모리 카드 전송 이상 비트 10~15: 무효
I/O 버스 이상	#80C0	비트 00~07: I/O 버스 이상 슬롯 번호 10진수 &0~9 또는 16진수 #00~09 특정하지 않았을 때: #0F 비트 08~15: I/O 버스 이상 랙 번호 10진수 &0~7 또는 16진수 #00~07 특정하지 않았을 때: 16진수 #0F
번호 이중 사용 에러	#80E9	CPU 고기능 유니트의 중복 유니트 번호 10진수 &0~15 또는 16진수 #0000~000F
		고기능 I/O 유니트의 중복 기기 번호 16진수 #8000~805F
증설 장치 중복 랙 번호	#80EA	중복 랙 번호(CH 할당 중복) 10진수 &0~6 또는 16진수 #0000~0006

이상 명칭	S 고장 코드 설정(16진수)	S+1 이상 내용 설정
INNER 보드 정지 이상	#82F0	이상 요인 비트 00~03: INNER 보드로 정의한 이상 비트 04~15: 무효
I/O 점수 초과	#80E1	비트 13~15: 이상 요인 비트 00~12: 상세 요인
		· I/O 총 점수 초과 비트 13~15: 000 비트 00~12: 총 I/O 점수(BIN)
		· 인터럽트 입력 점수 초과 비트 13~15: 001 비트 00~12: 인터럽트 입력 총 점수(BIN) 비트 00~12: 모두 0
		· 원격 I/O 슬레이브 유니트의 기기 번호 중복 또는 C500 원격 I/O 슬레이브 유니트의 점수 초과 비트 13~15: 010 비트 00~12: 중복된 원격 I/O 슬레이브 유니트의 기기 번호(BIN) (C500 원격 I/O 슬레이브 유니트에 320점을 초과하여 I/O가 장착된 경우의 기기 번호도 포함)
		· 전송 I/O의 기기 번호 중복 비트 13~15: 011 비트 00~12: 중복 전송 I/O의 기기 번호
		· 원격 I/O 마스터 유니트의 기기 번호 중복 또는 원격 I/O 마스터 유니트의 기기 번호 이상 비트 13~15: 100 비트 00~12: 중복 또는 이상이 발생한 원격 I/O 마스터 유니트의 기기 번호(BIN)
		· 장착 가능한 증설 장치 수 초과 비트 13~15: 101 비트 00~12: 증설 장치 장착 대수(BIN)
		· C200H 고기능 I/O 유니트를 인식 못하거나 원격 I/O를 인식 못함 비트 13~15: 110
I/O 설정 이상	#80E0	16진수 #0000
프로그램 에러	#80F0	비트 08~15: 이상 요인 이상 요인 비트 15: UM 초과 플래그 비트 14: 잘못된 명령 플래그 및 잘못된 개체 플래그 비트 13: 미분 초과 플래그 비트 12: 태스크 에러 플래그 비트 11: END 없음 플래그 비트 10: 잘못된 영역 액세스 에러 플래그 비트 09: DM/EM 간접 BCD 에러 플래그 비트 08: 명령 처리 에러 플래그 비트 00~07: 무효
사이클 시간 초과	#809F	16진수 #0000

■ 사용자 정의 운전 정지 이상 발생 시의 메시지 표시

· FALS 명령에서 명령 실행과 동시에 주변 툴에 메시지를 표시하려면 S에 CH 번호를 설정합니다.
(메시지 표시가 필요 없는 경우, S에 상수 #0000~FFFF 중 하나를 설정하십시오.)

· FALS 명령 실행에 의해 메시지가 등록됩니다. 메시지가 등록되면 프로그래밍 콘솔을 연결할 때 메시지가 표시됩니다.

· S~S+7CH의 8CH에 ASCII 코드로 16문자 분량의 문자를 설정할 수 있습니다.

· 메시지 표시 순서는 상위 바이트→하위 바이트입니다.

· 메시지 구분은 NUL(#00)입니다. NUL을 제외하고 최대 16문자까지 표시할 수 있습니다.

· FALS 명령 실행 후 메시지를 설정한 CH의 내용을 변경하면 변경 내용에 따라 메시지도 변경됩니다.

■ 시스템에 의한 운전 정지 이상 해제

시스템에서 고의로 발생시킨 운전 계속 이상을 해제하는 방법은 ① PLC 본체의 전원 OFF→ON 또는 ② (PLC 본체의 전원이 ON 상태인 경우) 해당하는 이상이 실제로 발생했을 때와 같은 처리를 수행합니다.

■ FALS 명령 해제 방법

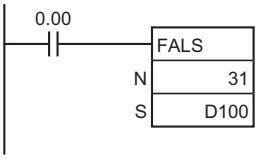
FALS 명령을 해제할 때는 이상 원인을 제거한 후 ① 주변 툴로 '이상 읽기/해제' 조작을 하거나 ② 전원을 OFF→ON으로 하십시오.

사용상의 주의 사항

· 시스템에 의한 운전 정지 이상의 등록을 실행한 경우, I/O 메모리 유지 플래그가 OFF이면 I/O 메모리가 클리어되므로 주의하십시오.

샘플 프로그램

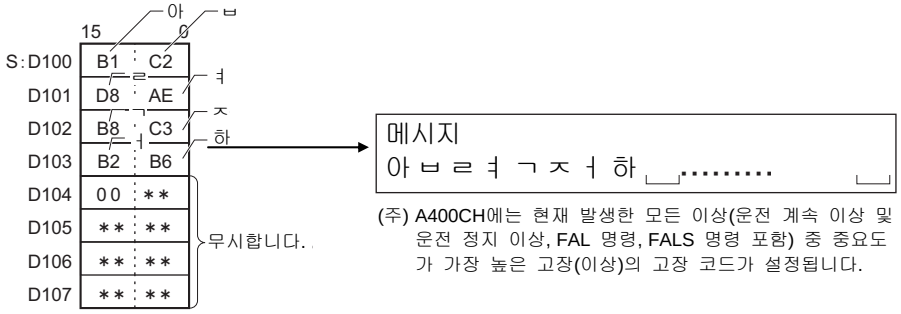
■ 사용자 정의 운전 정지 이상을 발생시킬 때



0.00이 ON이면 FALS 번호 031의 운전을 정지하는 고장(이상)이 발생했다고 판단하여 다음 처리를 실행합니다.

1) 특수 보조 릴레이 FALS 이상 플래그 A401.06을 ON
2) 특수 보조 릴레이 고장 코드 A400에 고장 코드 #C11F를 설정(이 고장이 중요도가 가장 높을 때) (주)
3) 특수 보조 릴레이의 이상 내역 저장 영역(A100~199)에 고장 코드 및 이상 발생 시각을 저장
4) CPU 유니트의 ERR LED 점등
5) D100~D107의 8CH 데이터를 16문자 ASCII 코드 데이터로 간주하고 주변 툴에 표시

주: 메시지 표시가 필요 없는 경우, S에 상수(#0000~FFFF 중 하나)를 설정하십시오.

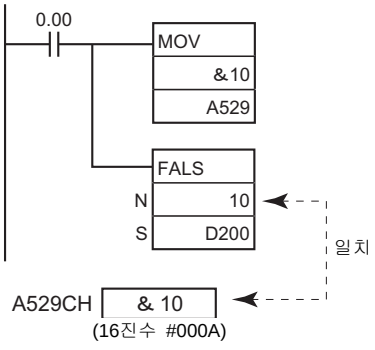


(주) A400CH에는 현재 발생한 모든 이상(운전 계속 이상 및 운전 정지 이상, FAL 명령, FALS 명령 포함) 중 중요도가 가장 높은 고장(이상)의 고장 코드가 설정됩니다.

■ 시스템에 의한 운전 정지 이상을 발생시킬 때

0.00이 ON이면 I/O 점수 초과 이상(장착 가능한 증설 장치 수 초과, 예: 9대)을 발생시킵니다. 더미의 FAL 번호에는 10을 사용합니다. (A529CH에 &10(16진수 #000A)을 저장해 둡니다.) 동시에 다음의 처리를 수행합니다.

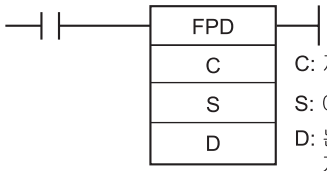
1) 특수 보조 릴레이 고장 코드 A400에 고장 코드 #80E1을 설정
   (이 고장이 중요도가 가장 높을 때)
2) 특수 보조 릴레이의 이상 내역 저장 영역(A100~199)에 고장 코드 및 이상 발생 시각을 저장
3) 특수 보조 릴레이 A401.11(I/O 점수 초과 플래그)이 1(ON)이 됩니다.
4) CPU 유니트의 ERR LED가 점등하고 운전이 정지됩니다.
5) 프로그래밍 콘솔에 'I/O 점수 초과'가 표시됩니다.



S:D200 # 80E1 ····· 고장 코드: #80E1(I/O 지점 수 초과 플래그)
D201 # A009 ····· 장착 증설 장치 수: 9

FPD

명령 이름	니모닉	동작 옵션	펑션 번호	기능
고장 지점 검출	FPD	–	269	지정한 1회로에 대해 시간 감시 진단 및 논리 진단을 수행합니다.

심볼	FPD
	

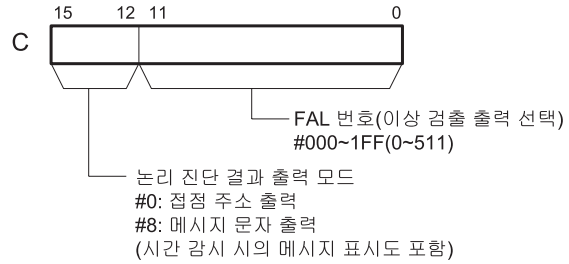
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	불가능	가능	가능	불가능	불가능

피연산자 설명

피연산자	내용	데이터 형식	크기
C	제어 데이터	–	1
S	이상 감시 시간 설정값	–	1
D	논리 진단 결과 출력 및 메시지 저장 하위 CH 번호	–	가변

C: 제어 데이터



S: 이상 감시 시간 설정값
&0~9999(10진수) 또는 #0000~270F(16진수)
설정 범위: 0~999.9초(단위 0.1초)
단, 0초를 지정한 경우 이상 감시 동작을 수행하지 않습니다.

D: 논리 진단 결과 출력 및 메시지 저장 하위 CH 번호
아래의 기능 설명을 참조하십시오.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
C	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–
S	○	○	○	○	○	○	○	○	○	○				○				
D											–							

조건 플래그의 동작

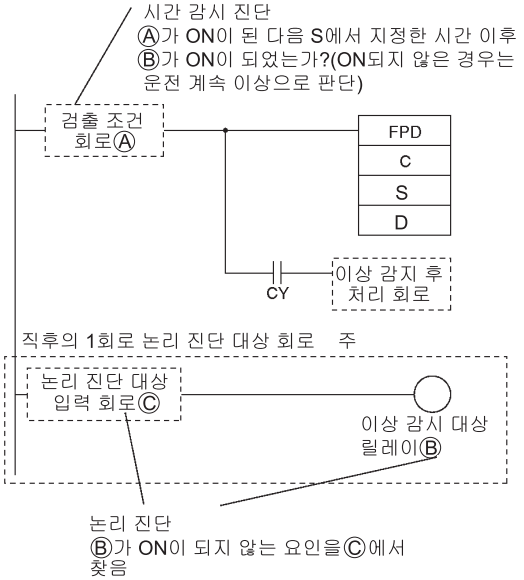
이름	라벨	내용
에러 플래그	P_ER	· C의 데이터가 범위 밖일 때 ON · S의 데이터가 범위 밖일 때 ON · 그 이외일 때 OFF
캐리 플래그	P_CY	· 지정된 이상 감시 시간이 경과하여 이상 감시 대상 릴레이가 0일 때 ON · 그 이외일 때 OFF

관련 특수 보조 릴레이

이름	주소	내용
FAL 이상 플래그	A402.15	시간 감시에서 FAL 이상을 등록했을 때 1(ON)이 됩니다.
실행 FAL 번호	A360.01~ A391.15	시간 감시에서 FAL 이상을 등록한 경우, N으로 지정한 데이터 10진수 &1~511(16진수 #0001~01FF)에 대응하는 비트가 1(ON)이 됩니다. A360.01~A391.15가 FAL001~511에 대응합니다.
이상 내역 저장 영역	A100~ 199CH	이상(시간 감시에 의한 이상 포함)이 발생하면 최신 20개의 고장 코드, 이상 발생 시각이 저장됩니다.
고장 코드	A400CH	이상(시간 감시에 의한 이상 포함)이 발생한 경우, N으로 지정한 데이터 10진수 &1~511(16진수 #0001~01FF)에 대응하는 고장 코드가 저장됩니다. 16진수 #4101~42FF가 FAL 1~511(16진수 #001~1FF)에 대응합니다. 주: 동시에 발생하면 중요도가 가장 높은 이상의 코드를 저장합니다.
FPD 명령용 티칭 플래그	A598.00	FPD 명령 실행 시에 이상 감시 시간을 자동 설정하여 동작(티칭 동작)시킬 때 1(ON)로 설정합니다.

기능

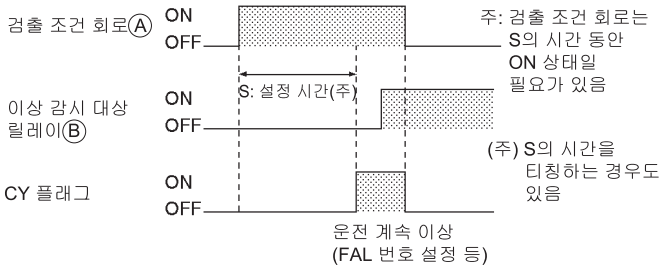
시간 감시 진단 동작과 논리 진단 동작을 수행합니다.
직후의 1회로를 다음과 같이 진단합니다.



주: FPD 명령 이후 최초의 LD(LD TR 제외) 또는 LD NOT 명령부터 시작하여 최초의 출력 명령(OUT TR 제외) 또는 응용 명령으로 끝나는 회로가 논리 대상 회로의 범위가 됩니다.

■ 시간 감시 진단

검출 조건 회로가 ON이 된 후, 이상 감시 대상 릴레이가 ON이 될 때까지의 시간을 감시하여 S로 설정한 시간을 초과해도 ON이 되지 않을 때 고장(운전 계속 이상)으로 판단합니다. CY 플래그를 ON합니다.

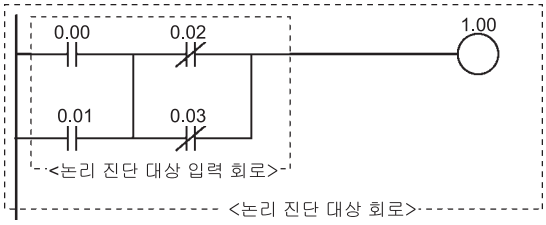


주:

- CY 플래그의 ON과 동시에 다음의 동작을 수행합니다. (C의 비트 00~11의 FAL 번호가 0일 때는 다음의 동작을 수행하지 않습니다.)
  1) 특수 보조 릴레이 FAL 이상 플래그(A402.15)를 ON합니다(운전은 계속).
  2) C로 지정한 FAL 번호가 1~511(16진수 #001~1FF)일 때는 특수 보조 릴레이의 실행 FAL 번호(A360~A391)의 해당 비트를 ON합니다(A360.01~A391.15: FAL 번호 001~511에 대응).
  3) 특수 보조 릴레이의 고장 코드(A400)에 고장 코드(FAL 번호 FALS 1~511(16진수 #001~1FF)에 대응되는 고장 코드 16진수 #4101~42FF)를 설정합니다. (단, 동시에 발생한 다른 고장 중에서 중요도가 가장 높은 경우에만 저장합니다.)
  4) 특수 보조 릴레이의 이상 내역 저장 영역(A100~199)에 고장 코드 및 이상 발생 시각을 저장합니다.
  5) CPU 유니트의 ERR LED가 점멸합니다.
  6) 논리 진단 결과 출력 모드가 메시지 문자 출력(C의 비트 2~10)의 내용을 운전 계속 이상 메시지로 표시합니다.
- 시간 진단을 하는 경우 검출 조건 회로는 S(설정 시간) 동안 ON 그대로 유지되어야 합니다.
- 검출 조건 회로 및 논리 진단 대상 입력 회로는 복수의 a 접점, b 접점으로 구성하십시오.
- 이상 검출 후 회로는 출력 명령을 사용하십시오. 특히 LD, LD NOT은 사용할 수 없으므로 주의하십시오.
- 이상 검출 후 처리 회로는 생략할 수 있습니다. 이상 검출 후 처리 회로는 출력 명령 또는 응용 명령으로 구성하십시오.

■ 논리 진단

검출 조건 회로(입력 조건)가 ON이면 사이클마다 이상 감시 대상 릴레이를 OFF로 하는 원인이 된 접점을 추출하여 D로 지정한 영역에 해당 접점의 정보를 저장합니다.



위의 예에서는 입력 릴레이 0.00~0.03이 모두 ON일 때는 0.02의 b 접점이 출력 릴레이 1.00이 ON이 되지 않는 원인으로 판단하여 결과가 출력됩니다. 논리 진단 결과, 접점 정보가 있는지 여부는 접점 정보 유무 플래그에 표시됩니다.

주:

- 이 논리 진단 동작은 검출 조건 회로가 ON인 동안 사이클마다 실행합니다. 시간 감시 진단 동작과 독립되어 있습니다.
- 논리 진단 대상 입력 회로는 복수의 a 접점, b 접점으로 구성하십시오.
- 이상 감시 대상 릴레이의 OFF 원인이 된 접점이 여러 개 존재하는 경우는 상단에 있고 모선에 가까운 1접점만 출력합니다.
- 논리 진단 추출 대상 접점은 LD, LD NOT, AND, AND NOT, OR, OR NOT(미분 및 매회 명령 포함)입니다. 그 이외의 명령 및 IR에 의한 간접 지정 접점 번호는 추출할 수 없습니다.
- 논리 진단 대상은 이 명령 이후 최초의 LD(LD TR 제외) 또는 LD NOT 명령부터 최초의 출력 명령(OUT TR 제외)까지의 1개 회로입니다.

- D 이후에 접점 정보를 저장하는 방법은 다음 2종류가 있습니다.
  1) 접점 주소 출력

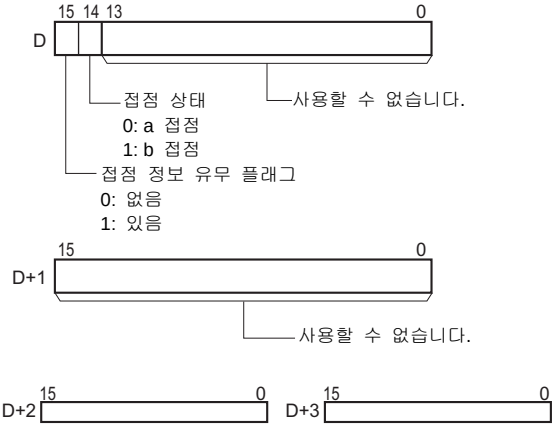
     접점 정보 유무를 D의 비트 15, 접점 상태(a 접점/b 접점의 종류)를 D의 비트 14, 접점의 I/O 메모리 실제 주소를 D+3, D+2의 2CH에 출력합니다.
  2) 메시지 문자 출력

     접점 정보 유무를 D의 비트 15, 접점 상태(a 접점/b 접점의 종류)를 D의 비트 14, 접점 번호의 ASCII 코드 6문자 분량을 D+2~D4의 3CH에 출력합니다.

■ D: 논리 진단 결과 출력 및 메시지 저장 하위 CH 번호

논리 진단 결과 출력 모드(C의 비트 12~15의 내용: 접점 주소 출력 또는 메시지 문자 출력)에 따라 설정 내용이 달라집니다.

· C의 비트 12~15 = 0(접점 주소 출력)일 때

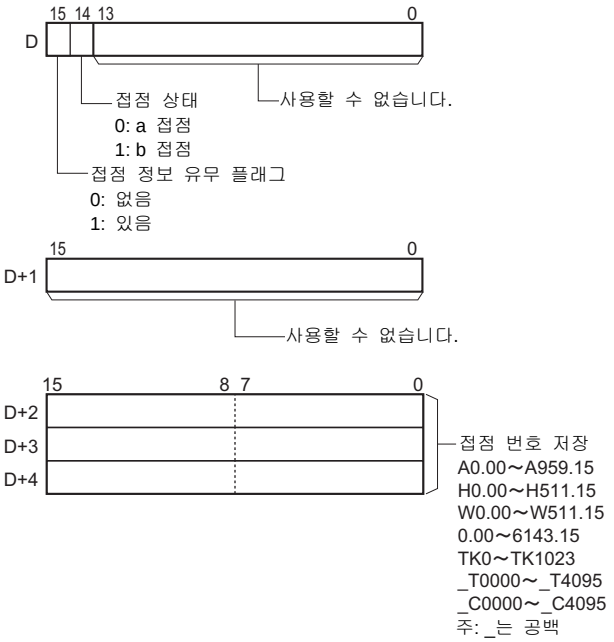


접점 주소가 I/O 메모리 실제 주소로 D+2, D+3의 2CH에 저장됩니다. 이 값을 인덱스 레지스터(IR)에 설정하면 접점을 간접 참조할 수 있습니다.

· C의 비트 12~15 = 8(메시지 문자 출력)일 때

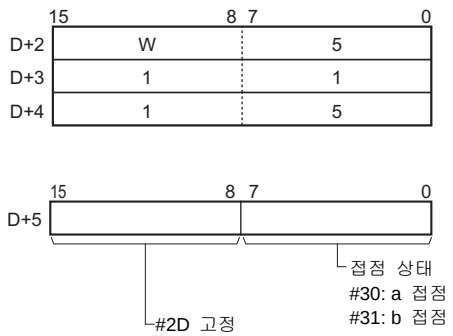
D~D+5에 논리 진단 결과가 출력됩니다.

D+6~D+10에 시간 감시 진단 주변 툴에 대한 메시지를 저장해 둡니다.



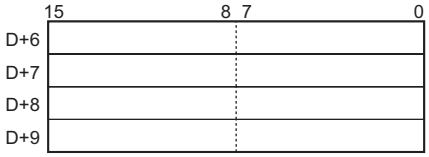
접점의 번호가 ASCII 코드 6문자로 D+2~D+4에 출력됩니다.

예: W511.15일 때 다음 순서로 각 문자/수를 ASCII 코드로 저장



· D+6~D+9:

사용자 정의 임의의 메시지 저장 영역입니다. C의 비트 12~15 = 8(메시지 문자 출력)일 때 D+6~D+9에 사용자 정의의 임의 메시지(ASCII 코드로 8문자까지)를 저장해 두고 시간 감시 결과, 운전 계속 이상이 되었을 때 주변 툴에 메시지를 표시할 수 있습니다. 메시지 마지막에는 NUL(#00)을 부가하여 구분하십시오.



힌트

■ FPD 명령의 시간 감시 진단 실행 시의 이상 내역 등록 유무 지정

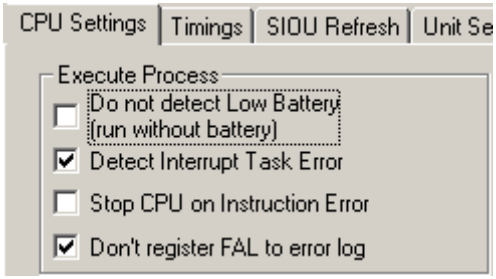
FPD 명령의 시간 감시 진단을 실행했을 때 사용자 정의 운전 계속 이상이 발생하지만 특수 보조 릴레이의 이상 내역 저장 영역(A100~199CH)에는 이상(고장 코드 및 이상 발생 시각)을 등록하지 않을 수 있습니다(주).

주: 이때, 특수 보조 릴레이 A402.15의 FAL 이상 플래그, A360~391CH의 실행 FAL 번호 및 A400CH의 고장 코드에는 반영됩니다.

이 기능은 이상 내역에 시스템에 의한 이상만 저장할 때 사용합니다. 예를 들어, FAL 명령을 많이 사용하는 활용 사례에서 디버그 중에 이상 내역 저장 영역이 FAL 명령에 의한 이상으로 가득 차는 것을 방지할 경우 등에 사용합니다.

다음과 같은 PLC 시스템 설정으로 지정합니다.

· PLC 시스템 설정(CX-Programmer의 경우)



· PLC 시스템 설정(프로그래밍 콘솔의 경우)

프로그래밍 콘솔 주소	채널	+129
	비트	15
이름	FAL 이상에 대한 이상 내역 등록 유무 지정	
설정 데이터	0: 이상 내역에 등록 1: 이상 내역에 등록 안 함	
초기값	0: 이상 내역에 등록	
CPU 유니트 반영 타이밍	사이클마다(FAL 이상 발생 시)	

주: 위의 PLC 시스템 설정에서 '이상 내역에 등록 안 함'으로 설정한 경우에도 다음의 이상은 이상 내역에 등록됩니다.

· FALS 명령 실행에 의한 운전 정지 이상
· 시스템에 의한 운전 계속 이상
· 시스템에 의한 운전 정지 이상
· FAL(FPD) 명령 실행에 의해 고의로 발생시킨 시스템 운전 계속 이상
· FALS 명령 실행에 의해 고의로 발생시킨 시스템 운전 정지 이상

■ 티칭에 의한 이상 감시 시간 설정값(S)의 자동 설정

S로 지정한 이상 감시 시간 설정값이 CH 지정이면 티칭을 통해 이상 감시 시간 설정값을 지정할 수 있습니다. 그 결과, 정상 동작 시의 감시 시간을 자동으로 설정할 수 있습니다.

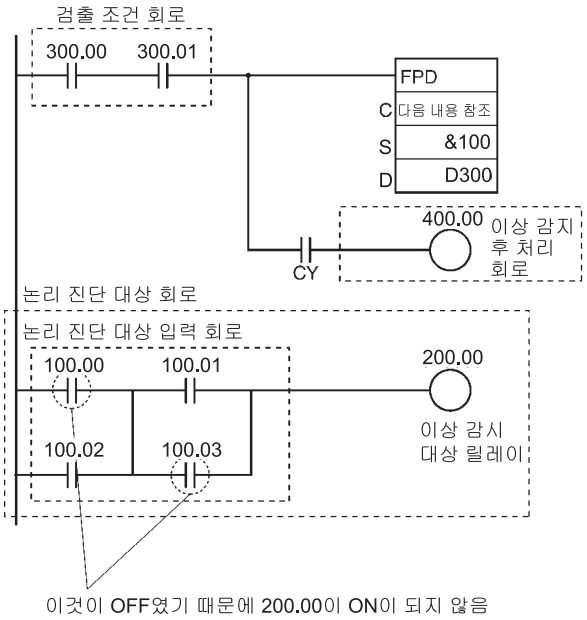
S가 CH 지정일 때 다음 동작을 수행합니다.

1) 사용자가 FPD 명령용 티칭 플래그(A598.00)를 ON으로 합니다.
2) A598.00이 ON인 동안에 검출 조건 회로가 ON이 된 후, 이상 감시 릴레이가 ON이 될 때까지의 시간을 계측합니다.
3) 계측한 시간이 설정값을 초과하면 1.5배의 값이 S에 저장됩니다.

주:

· FPD 명령은 프로그램에서 여러 개 사용할 수 있습니다. 단, D(논리 진단 결과 출력 및 메시지 저장 하위 CH 번호)는 다른 CH 번호를 지정하십시오.
· 감시 시간 갱신은 명령 실행 시에만 이루어지므로 사이클 시간이 100ms 이상이면 정상적으로 동작하지 않으므로 주의하십시오(100ms 단위의 갱신이기 때문).

샘플 프로그램



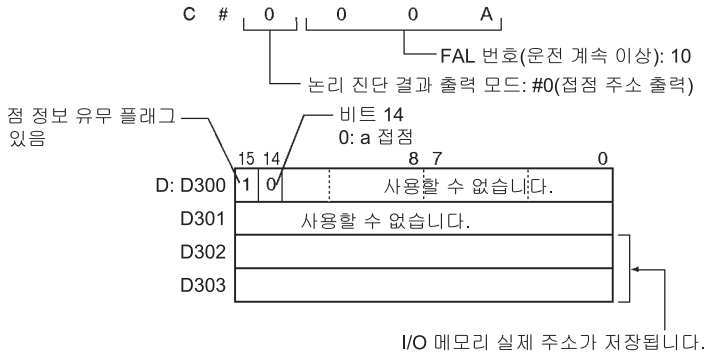
▪ 시간 감시 진단

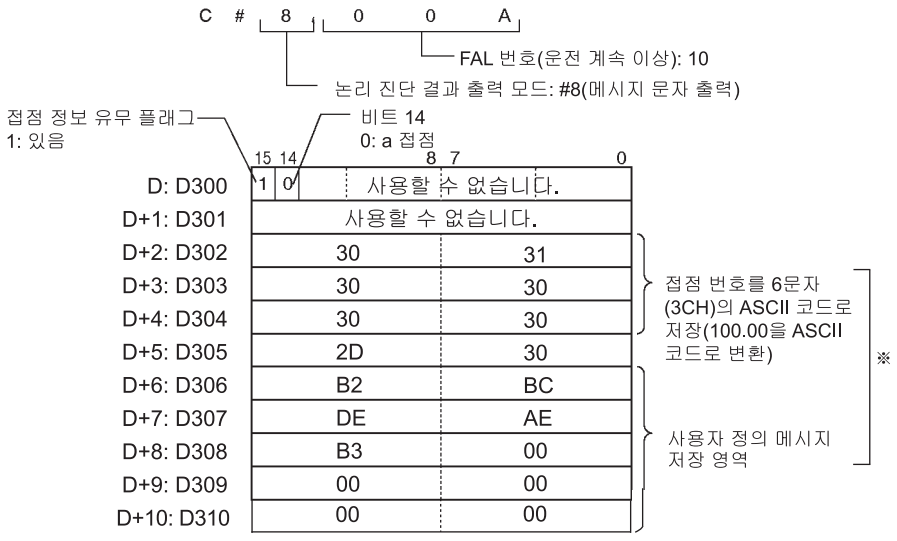
검출 조건 회로의 300.00과 300.01이 모두 ON이 되고 10초 후에 이상 감시 대상 릴레이 200.00이 ON이 되지 않은 경우 고장(이상)으로 판단합니다.

1) CY 플래그를 ON으로 합니다.
2) C의 비트 0~11의 FAL 번호(운전 계속 이상)가 00AHex(10)인 경우, 특수 보조 릴레이 A360의 비트 10을 ON으로 합니다. 동시에 특수 보조 릴레이 A400에 고장 코드 410A를 설정합니다. 또한 동시에 특수 보조 릴레이 A402.15를 ON으로 합니다.

▪ 논리 진단

논리 진단 대상 회로 중에 이상 감시 대상 릴레이 200.00을 OFF로 하는 원인이 된 접점이 100.00과 100.03인 경우, 접점 주소 출력의 경우 C의 비트 12~15의 논리 진단 결과 출력 모드가 #0(접점 출력)일 때, 100.00(100.03보다 상단에 있으므로)의 I/O 메모리 실제 주소가 D303, D302에 저장됩니다.

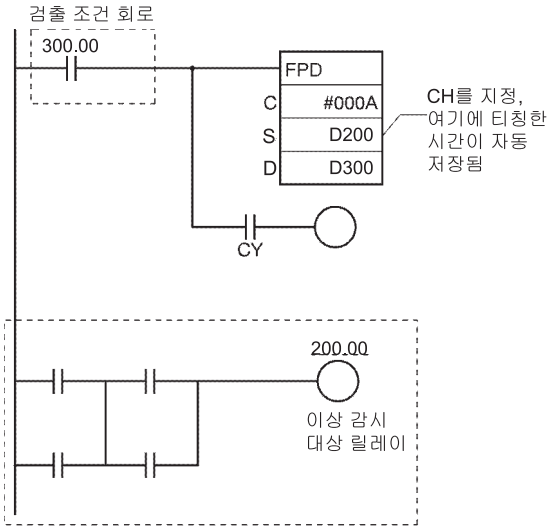




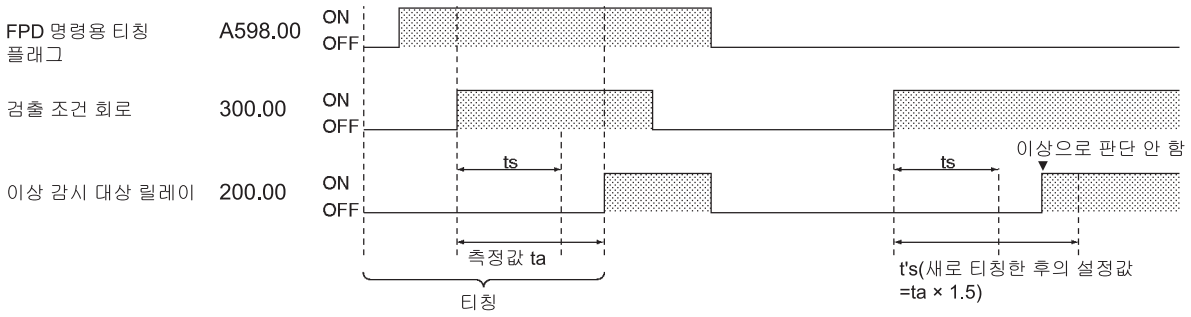
※: 시간 감시 진단으로 등록한 FAL 이상 메시지로 주변 툴에 표시됩니다.

주변 툴의 표시: 100.00 – 0 이상

- S의 이상 감시 시간 설정값을 티칭하는 경우 (S: 채널을 지정합니다.)



티칭을 시작하기 위해 특수 보조 릴레이 A598.00(FPD 명령용 티칭 플래그)을 ON으로 합니다.
A598.00이 ON인 동안 검출 조건 회로 300.00이 ON이 된 후, 이상 감시 대상 릴레이 200.00이 ON이 될 때까지의 시간을 계측합니다.
계측한 시간이 설정값을 초과하면 해당 시간의 1.5배의 값이 새로운 설정값으로 S에 저장됩니다.



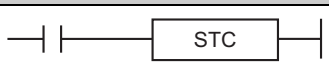
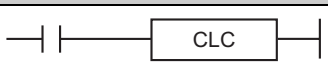
ts: S의 초기 설정값　t's: 새로 티칭한 후의 설정값
ta: 측정값
ta>ts의 경우　t's = ta×1.5로 함

특수 명령

니모닉	명령 이름	펑션 번호	페이지
STC	세트 캐리	040	3-888
CLC	클리어 캐리	041	
EMBC	확장 DM 뱅크 전환	281	3-889
WDT	사이클 시간 감시 시간 설정	094	3-891
CCS	조건 플래그 저장	282	3-893
CCL	조건 플래그 로드	283	
FRMCV	CV→CS 주소 변환	284	3-895
TOCV	CS→CV 주소 변환	285	3-898
IOSP	주변 서비스 금지	287	3-901
IORS	주변 서비스 금지 해제	288	

STC/CLC

명령 이름	니모닉	동작 옵션	평션 번호	기능
세트 캐리	STC	@STC	040	CY 플래그를 ON합니다.
클리어 캐리	CLC	@CLC	041	CY 플래그를 OFF합니다.

심볼	STC	CLC
		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

조건 플래그의 동작

이름	라벨	내용	
		STC	CLC
에러 플래그	P_ER	변하지 않음*1	변하지 않음*1
= 플래그	P_EQ	변하지 않음*1	변하지 않음*1
캐리 플래그	P_CY	ON	OFF
부정 플래그	P_N	변하지 않음*1	변하지 않음*1

*1: CS1-H/CJ1-H/CJ1M/CS1D(CPU 단독 시스템용) CPU 유니트는 변하지 않습니다.
CS1/CJ1/CS1D(이중화 시스템용) CPU 유니트는 OFF

기능

■ STC

입력 조건이 ON이면 조건 플래그의 캐리 플래그를 1(ON)로 합니다. 단, 이 명령으로 캐리 플래그를 ON한 후에도 평상시대로 다른 명령의 실행에 의해 캐리 플래그가 OFF/ON됩니다.

ROL, ROLL, ROR, RORL 명령에서는 캐리 플래그를 포함하여 회전 이동되므로 캐리 플래그를 조작하기 위해 이 STC 명령을 사용합니다.

■ CLC

입력 조건이 ON이면 조건 플래그의 캐리 플래그를 0(OFF)으로 합니다. 단, 이 명령으로 캐리 플래그를 OFF한 후에도 평상시대로 다른 명령의 실행에 의해 캐리 플래그가 ON/OFF됩니다.

+C, +CL, +BC, +BCL, -C, -CL, -BC, -BCL 명령에서는 캐리 플래그를 포함하여 연산하므로 직전 명령의 영향을 받지 않게 하기 위해 이 CLC 명령을 사용합니다.

ROL, ROLL, ROR, RORL 명령에서는 캐리 플래그를 포함하여 회전 이동되므로 캐리 플래그를 조작하기 위해 이 CLC 명령을 사용합니다.

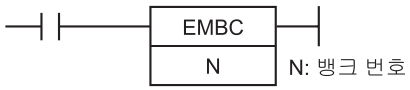
힌트

- 가산/감산 명령에는 다음과 같이 캐리 플래그를 포함하지 않고 연산하는 명령이 있습니다. 가산/감산 시에는 기본적으로 이들 명령을 사용합니다.

  +, +L, +B, +BL, −, −L, −B, −BL,

EMBC

명령 이름	니모닉	동작 옵션	펑션 번호	기능
확장 DM 뱅크 전환	EMBC	@EMBC	281	EM(확장 DM) 영역의 현재 뱅크를 전환합니다.

심볼	EMBC
	

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	뱅크 번호	UINT	1

N: 뱅크 번호

16진수 #0000~0018

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
N	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· N으로 지정한 뱅크 번호의 EM이 존재하지 않을 때 ON(파일 메모리화된 경우도 포함) · N으로 지정한 뱅크 번호의 EM이 주소 자동 할당 영역에 지정된 경우 ON(CJ2 CPU 유니트만 해당) · EM 강제 세트/리셋 기능이 설정되어 있을 때 ON(CJ2 CPU 유니트만 해당) · 인터럽트 태스크에서 실행했을 때 ON(CJ2 CPU 유니트에서 인터럽트 고속 기동 기능 사용 시) · 그 이외일 때 OFF

관련 특수 보조 릴레이

이름	주소	내용
현재 EM 뱅크	A301	현재 EM 뱅크 번호가 BIN으로 설정됩니다.

EMBC

기능

EM(확장 데이터 메모리)의 현재 뱅크를 N으로 지정한 뱅크 번호로 전환합니다. 이때, 현재 뱅크의 번호를 A301CH에 출력합니다.

주:

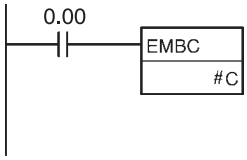
- 사이클릭 태스크 내에서 전환된 현재 EM 뱅크 번호는 태스크 전환 시에도 유지됩니다(사이클릭 태스크 공통). 따라서, 예를 들어 사이클릭 태스크 1의 EMBC 명령으로 현재 뱅크 번호를 B에서 C로 전환한 후, 사이클릭 태스크 2에서도 현재 뱅크 번호는 C 그대로입니다.
- 인터럽트 태스크 내에서 전환된 현재 EM 뱅크 번호는 해당 인터럽트 태스크 내에서만 유효합니다. 따라서 인터럽트 태스크가 종료하면 원래 뱅크 번호로 돌아갑니다.

힌트

- 확장 데이터 메모리(EM)에는 뱅크가 있습니다. 각 뱅크에는 각각 E00000~32767의 32768CH가 있습니다. 확장 데이터 메모리(EM)를 지정하는 경우 다음 2가지 지정 방법이 있습니다.
  1) 뱅크 번호를 포함하지 않고 E00000~32767로 지정한 경우, 현재 뱅크 내의 E00000~32767을 가리킵니다.
  2) 뱅크 번호를 포함하여 En_00000~32767(n=뱅크 번호)로 지정한 경우, 지정한 뱅크의 E00000~32767을 가리킵니다.

     1)의 지정 방법으로 지정한 경우 현재 뱅크를 전환하기 위해 EMBC 명령이 필요합니다.

샘플 프로그램

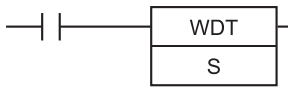


0.00이 ON이면 EM(확장 DM)의 현재 뱅크를 뱅크 번호 C로 전환합니다.
동시에 특수 보조 릴레이 A301CH에 뱅크 번호 #000C가 저장됩니다.

A301CH	0	0	0	C

WDT

명령 이름	니모닉	동작 옵션	평션 번호	기능
사이클 시간 감시 시간 설정	WDT	@WDT	094	이 명령을 실행한 사이클만 사이클 시간 감시 시간을 연장합니다.

심볼	WDT
	 S: 사이클 시간 감시 시간 연장 데이터

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	사이클 시간 감시 시간 연장 데이터	상수만	1

S: 사이클 시간 감시 시간 연장 데이터
10진수 &0~3999 또는 16진수 #0000~0F9F

■ 영역 종류

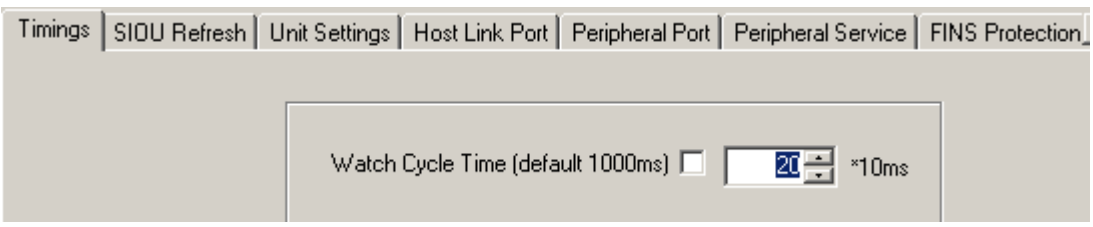
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 사이클 시간 감시 시간이 40s를 초과했을 때 ON · 그 이외일 때 OFF

관련 PLC 시스템 설정

· PLC 시스템 설정(CX-Programmer의 경우)



WDT

· PLC 시스템 설정(프로그래밍 콘솔의 경우)

이름	내용	설정
사이클 시간 감시 시간 설정	사이클 시간이 설정된 감시 시간을 초과했을 때, 사이클 시간 초과 플래그(운전 정지 이상)를 등록합니다.	0: 초기값(1,000ms) 1: 임의의 시간 설정
	사이클 시간 감시 시간을 설정합니다(위 설정을 '1: 임의의 시간 설정'으로 지정한 경우만 해당)	#0001~0FA0 : 1~40,000ms(10ms 단위)

주:

· 사이클 시간 감시 시간은 PLC 시스템 설정을 통해 1~40,000ms(10ms 단위) 범위에서 설정할 수 있습니다. 초기값은 1000ms(1s)입니다.

· 이 명령을 1사이클 내에서 여러 개 사용할 수도 있습니다. 이 경우, 각각의 연장 시간이 누적됩니다. 단, 누적값이 40,000ms(40s)를 초과할 수는 없습니다.

· 이 명령을 실행하려고 했을 때 누적값이 이미 40,000ms(40s)에 도달한 경우는 이 명령이 실행되지 않습니다.

관련 특수 보조 릴레이

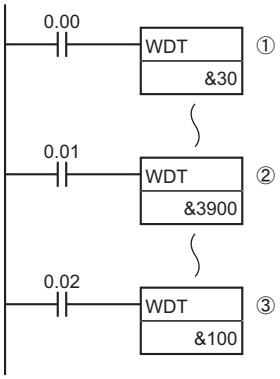
이름	주소	내용
사이클 시간 초과 플래그	A401.08	사이클 시간 현재값이 PLC 시스템 설정의 '사이클 시간 감시 시간'을 초과했을 때 1(ON)이 됩니다. CPU 유니트의 운전이 정지됩니다.
사이클 시간 최대값	A262~A263	사이클 시간 최대값이 BIN 32비트로 사이클마다 저장됩니다.
사이클 시간 현재값	A264~A265	사이클 시간 현재값이 BIN 32비트로 사이클마다 저장됩니다.

기능

이 명령 실행 사이클에 대해서만 PLC 시스템 설정으로 지정한 사이클 시간 감시 시간에 S로 지정한 데이터 ×10ms(0~39,990ms)만큼 연장합니다.

일시적으로 처리 내용이 증가하여 사이클 시간이 연장될 때 이 명령을 통해 사이클 시간이 초과되는 것을 막을 수 있습니다.

샘플 프로그램



WDT 명령의 동작

위의 예는 사이클 시간 감시 시간의 설정값이 기본값인 1000ms인 경우입니다.

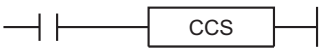
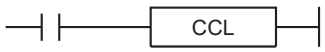
· 입력 릴레이 0.00이 ON이면 ①의 WDT 명령을 통해 사이클 시간 감시 시간이 300ms(30이므로, 30×10ms) 만큼 연장됩니다. 따라서 이 시점에서 사이클 시간 감시 시간의 누적값은 1300ms(1000ms+300ms)가 됩니다.

· 입력 릴레이 0.01이 ON일 때, ②의 WDT 명령을 통해 사이클 시간 감시 시간을 다시 39,000ms(3900이므로 3900×10ms) 연장하려고 합니다. 그러나 누적값이 40,300ms가 되므로 40,300ms에서 초과분인 300ms가 무시되어 결과적으로 이 시점에서는 38,700ms가 연장됩니다.

· 입력 릴레이 0.02가 ON이면 ③의 WDT 명령을 통해 사이클 시간 감시 시간을 다시 1,000ms(100이므로 100×10ms) 연장하려고 합니다. 그러나 이미 누적값이 40,000ms에 도달했으므로 ③의 WDT 명령은 실행되지 않습니다.

CCS/CCL

명령 이름	니모닉	동작 옵션	펑션 번호	기능
조건 플래그 저장	CCS	@CCS	282	조건 플래그의 상태를 보관합니다.
조건 플래그 로드	CCL	@CCL	283	보관된 조건 플래그의 상태를 읽어옵니다.

심볼	CCS	CCL
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

조건 플래그의 동작

■ CCS

변하지 않습니다.

■ CCL

읽은 값으로 변경됩니다.

기능

■ CCS

입력 조건이 ON이면 그 시점에서 다음의 모든 조건 플래그(상시 ON, 상시 OFF 제외)의 상태를 CPU 유니트 내의 보관 영역에 저장합니다.

조건 플래그: P_ER, P_CY, P_GT, P_EQ, P_LT, P_N, P_OF, P_UF, P_GE, P_NE, P_LE

보관 영역에 저장된 조건 플래그의 복구(읽기)는 CCL 명령(조건 플래그 로드 명령)을 통해서만 실행할 수 있습니다.

주: 이 CCS 명령을 실행하면 이전에 CCS 명령으로 저장한 내용을 덮어 쓰게 됩니다.

■ CCL

입력 조건이 ON이면 CPU 유니트의 보관 영역에 저장된 다음의 모든 조건 플래그(상시 ON, 상시 OFF 제외)의 상태를 복구(읽기)합니다.

조건 플래그: P_ER, P_CY, P_GT, P_EQ, P_LT, P_N, P_OF, P_UF, P_GE, P_NE, P_LE

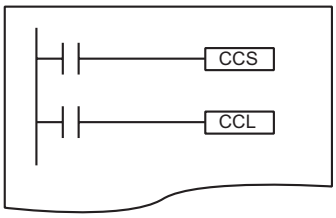
조건 플래그를 보관 영역에 대한 저장하는 작업은 CCS 명령(조건 플래그 저장 명령)을 통해서만 실행할 수 있습니다.

조건 플래그의 보관/복구는 다음 중 하나에서 실행할 수 있습니다.

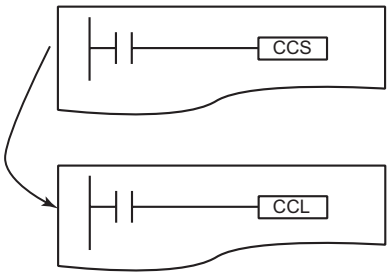
- 1개 태스크 내
- 사이클릭 태스크-사이클릭 태스크 사이
- 사이클 사이

CCS/CCL

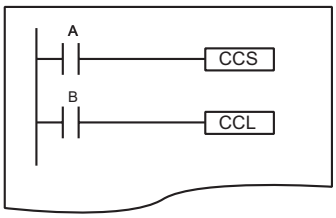
· 1개 태스크 내



· 사이클릭 태스크 사이



· 사이클 사이



CCS를 실행한 다음 사이클에서 CCL을 실행

단, 사이클릭 태스크와 인터럽트 태스크 사이에서는 보관/복구할 수 없습니다.

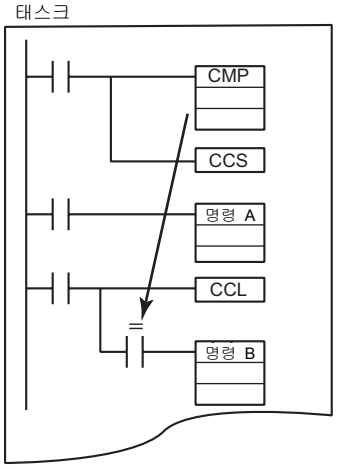
힌트

· 조건 플래그는 명령에서 공통으로 사용되기 때문에 각 명령 실행별로 해당 실행 결과에 따라 1사이클 내에서 변경됩니다. 그러므로 조건 플래그는 하나의 프로그램 내에서 간섭 받지 않도록 해당 읽기 위치에 주의해야 합니다.

따라서 어떤 명령을 실행한 직후의 조건 플래그 상태를 나중에 원하는 위치에서 읽으려면 이 CCS 명령/CCL 명령을 사용합니다.

예) 비교 명령 실행에 의한 = 플래그(비교 결과) 상태를 해당 비교 명령 직후가 아니라 나중에 읽고 싶을 때

태스크



…비교 결과를 조건 플래그에 저장합니다.
<이 비교 명령의 실행 결과를 명령 B로 수신하며 명령 A에 의해 간섭을 받지 않으려는 경우>

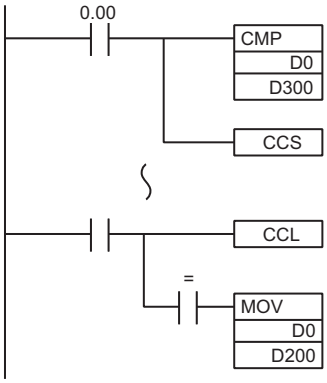
…조건 플래그의 상태를 보관합니다.

…조건 플래그의 상태를 복구합니다.

… = 플래그는 명령 A에 의한 간섭을 받지 않고, CCS 명령 직전의 CMP 명령 실행 결과가 됩니다.

· 태스크를 전환하면 조건 플래그가 모두 클리어됩니다. 따라서, 태스크 간(사이클릭 태스크-사이클릭 태스크 간)에 조건 플래그를 주고받거나 이후 사이클에서 조건 플래그를 주고받으려는 경우는 이 CCS 명령/CCL 명령을 사용하여 조건 플래그를 보관/복구시킵니다.

샘플 프로그램



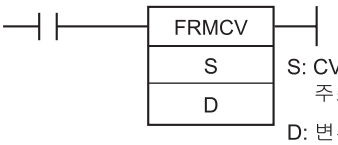
입력 릴레이 0.00이 ON이면 D0과 D300의 내용을 비교하여 그 결과(조건 플래그)를 CCS 명령으로 보관합니다.

보관된 조건 플래그를 CCL 명령으로 복구합니다.

예를 들어, 비교 결과의 = 플래그가 ON이면 MOV 명령을 실행합니다.

FRMCV

명령 이름	니모닉	동작 옵션	평션 번호	기능
CV→CS 주소 변환	FRMCV	@FRMCV	284	CVM1/CV 시리즈의 I/O 메모리 실제 주소를 CS/CJ 시리즈의 I/O 메모리 실제 주소로 변환합니다.

심볼	FRMCV
	 S: CVM1/CV 시리즈의 메모리 주소 저장 대상 채널 번호 D: 변환 결과 출력 대상 IR

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	CVM1/CV 시리즈의 메모리 주소 저장 대상 채널 번호	UINT	1
D	변환 결과 출력 대상 IR	IR*1만 가능	2

*1: D에는 인덱스 레지스터(IR0~15)만 지정할 수 있습니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	-	○	—	—	—	—
D	—	—	—	—	—	—	—	—	—	—	—	—	○	—				

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· CS/CJ 시리즈에는 존재하지 않는 다음 채널 번호의 I/O 메모리 실제 주소를 S에 지정했을 때 ON · 일시 기억 릴레이 TR(#09FF) · CPU 버스 링크 릴레이 G(#0A00~0AFF) · SFC 영역(#0D00~0E3F) 괄호 안은 I/O 메모리 실제 주소 · 그 이외일 때 OFF

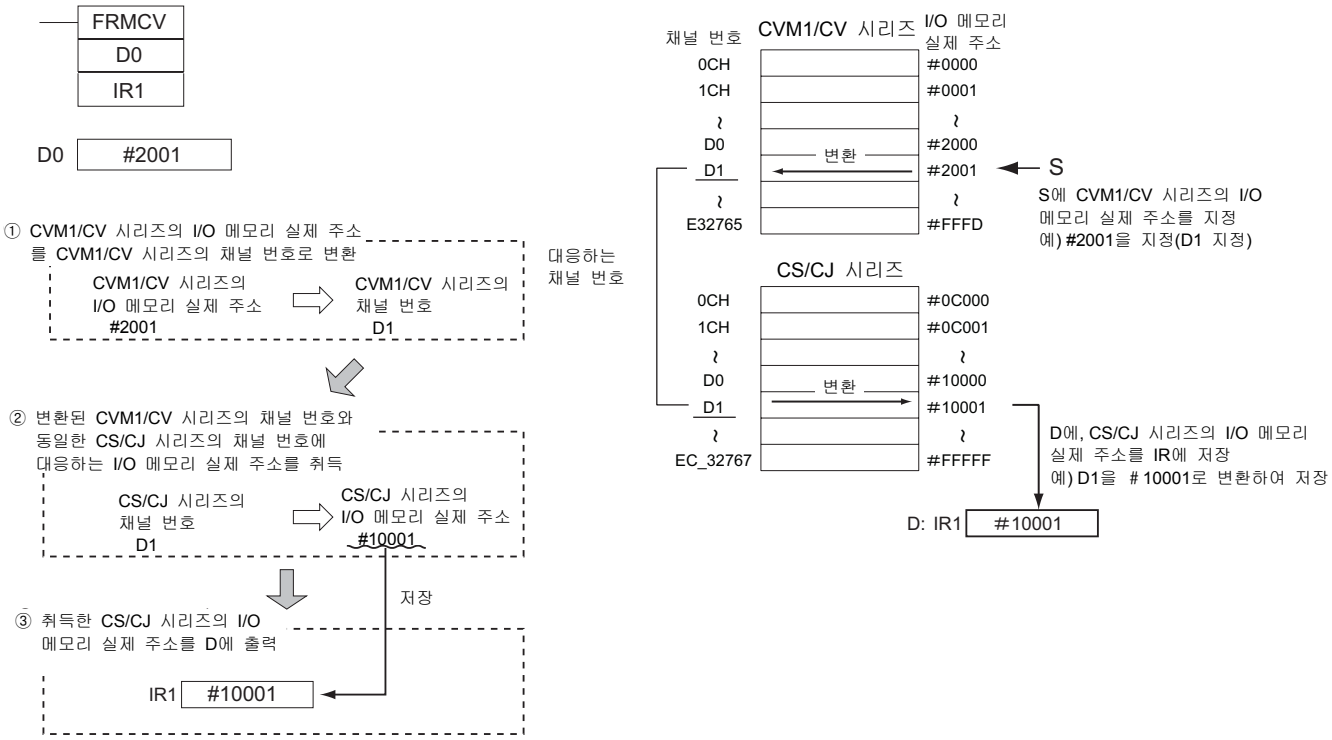
기능

FRMCV 명령은 입력 조건이 ON일 때 다음 동작을 실행합니다.

1. S로 지정한 CVM1/CV 시리즈의 I/O 메모리 실제 주소를 CVM1/CV 시리즈의 채널 번호로 일단 변환합니다.
2. 변환된 CVM1/CV 시리즈의 채널 번호와 동일한 CS/CJ 시리즈의 채널 번호에 대응하는 I/O 메모리 실제 주소를 취득합니다.
3. 취득한 CS/CJ 시리즈의 I/O 메모리 실제 주소를 D에 출력합니다.

FRMCV

예)



주: CVM1/CV 시리즈에 있고 CS/CJ 시리즈에는 없는 채널 번호의 경우, 에러 플래그가 ON이 되어 변환되지 않습니다.

힌트

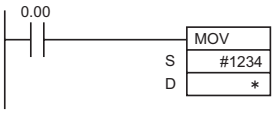
- 이 명령에 의해 인덱스 레지스터에 저장된 채널 번호는 일반적인 명령으로 인덱스 레지스터 간접 지정 (,IR 지정)을 수행하면 해당 I/O 메모리 실제 주소를 지정할 수 있습니다.
- CVM1/CV 시리즈에서 다음과 같은 프로그램을 작성하는 경우, 이 FRMCV 명령을 사용하여 해당 프로그램을 CS/CJ 시리즈용 프로그램으로 치환할 수 있습니다(뒤에 소개하는 샘플 프로그램 참조).
  - ① 간접 DM(*DM)의 BIN 지정을 사용하는 경우
    (DM 내의 I/O 메모리 실제 주소로 채널 번호를 간접 지정하는 경우)
  - ② CVM1/CV 시리즈의 I/O 메모리 실제 주소를 직접 값으로 사용하는 경우
    (MOV 명령 등으로 I/O 메모리 실제 주소를 IR에 직접 지정하여 저장하는 경우)

샘플 프로그램

■ CVM1/CV 시리즈에서 간접 DM(*DM)을 BIN 지정한 프로그램을 CS/CJ 시리즈용으로 치환하는 경우

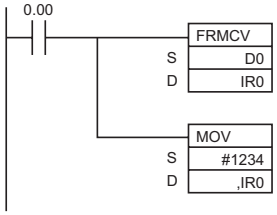
FRMCV 명령으로 S에 DM 영역을 지정하여 IR에 I/O 메모리 실제 주소를 저장한 후 IR 간접 지정을 실행합니다.

● CVM1/CV 시리즈의 프로그램
(간접 DM의 BIN 지정을 사용한 프로그램)



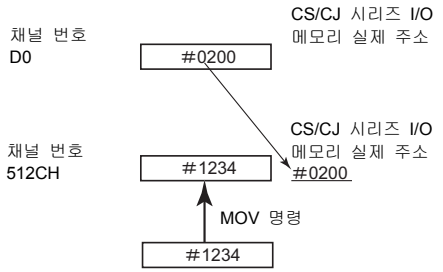
동등한 프로그램입니다.

● CS/CJ 시리즈의 프로그램

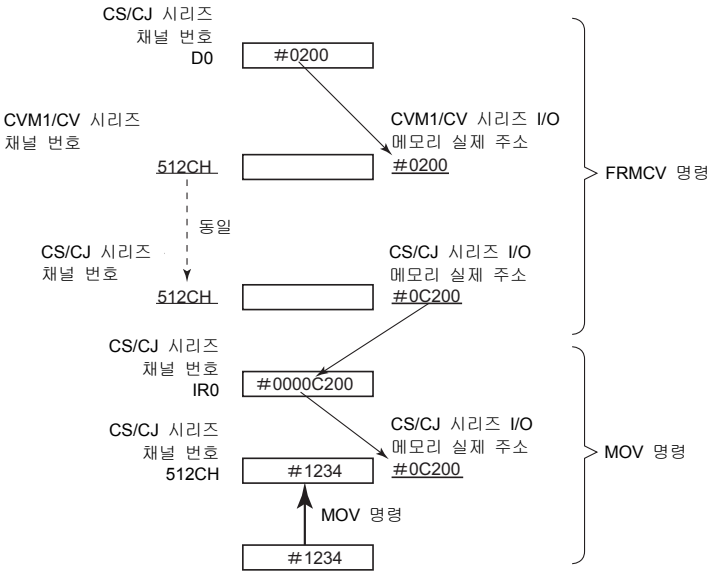


PLC 시스템 설정
간접 DM 데이터: BIN 지정 * DM 지정을 사용하는 경우, DM 내의 데이터를 I/O 메모리 실제 주소로 하여 해당 I/O 메모리 실제 주소에 대응하는 I/O 메모리 주소를 지정합니다.

예) D0 내의 값이 #0200인 경우, I/O 메모리 실제 주소 #0200에 대응하는 채널 번호는 512CH이므로 512CH에 #1234를 전송합니다.



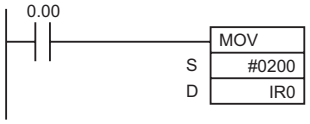
예) D0 내의 값이 #0200인 경우, CVM1/CV 시리즈의 I/O 메모리 실제 주소 #0200에 대응하는 CVM1/CV 시리즈의 채널 번호는 512CH이므로 그와 동일한 CS/CJ 시리즈의 512CH의 I/O 메모리 실제 주소 #0C200을 인덱스 레지스터 IR0에 저장합니다. MOV 명령에서 IR0을 전송 대상으로 간접 지정하면 512CH에 #1234가 전송됩니다.



■ CVM1/CV 시리즈에서 직접 I/O 메모리 실제 주소를 IR에 저장한 프로그램을 CS/CJ 시리즈용으로 치환하는 경우

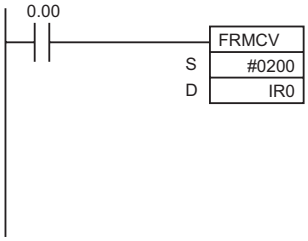
FRMCV 명령으로 S에 CVM1/CV 시리즈의 I/O 메모리 실제 주소를 직접 지정합니다.

● CVM1/CV 시리즈의 프로그램
(직접, I/O 메모리 실제 주소를 IR에 저장한 프로그램)

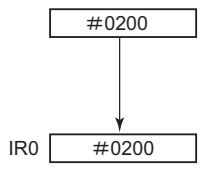


동등한 프로그램입니다.

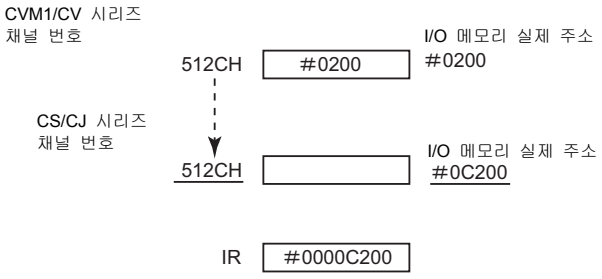
● CS/CJ 시리즈의 프로그램



I/O 메모리 실제 주소 #0200(512CH)을 인덱스 레지스터 IR0에 저장합니다.

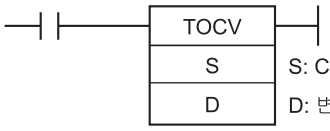


CVM1/CV 시리즈의 I/O 메모리 실제 주소 #0200의 채널 번호 512CH와 동일한 CS/CJ 시리즈의 I/O 메모리 실제 주소 #0C200을 인덱스 레지스터 IR0에 저장합니다.



TOCV

명령 이름	니모닉	동작 옵션	평션 번호	기능
CS→CV 주소 변환	TOCV	@TOCV	285	CS/CJ 시리즈의 I/O 메모리 실제 주소를 CVM1/CV 시리즈의 I/O 메모리 실제 주소로 변환합니다.

심볼	TOCV
	 S: CS/CJ 시리즈의 I/O 메모리 실제 주소 저장 대상 IR D: 변환 결과 출력 대상

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	CS 시리즈의 I/O 메모리 실제 주소 저장 대상 IR	IR*1만 가능	2
D	변환 결과 출력 대상	UINT	1

*1: S에는 인덱스 레지스터(IR0~15)만 지정할 수 있습니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	—	—	—	—	—	—	—	—	—	—	○	—	○	—	—	—	—	—
D	○	○	○	○	○	○	○	○	○	○	—	○	—	○				

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· CVM1/CV 시리즈에는 존재하지 않는 채널 번호의 I/O 메모리 실제 주소를 S에 지정했을 때 ON · 그 이외일 때 OFF

주

· S로 지정한 IR 내에 CVM1/CV 시리즈에 존재하지 않는 다음과 같은 채널 번호의 I/O 메모리 실제 주소가 저장되어 있으면 에러 플래그가 ON이 됩니다. 괄호 안은 I/O 메모리 실제 주소

TK 플래그(#0000B800~0000B801)

A512~1471(#0000BA40~0000BDFF), A10000~11535(#0000B200~0000B7FF)

CIO2556~6143(#0000C9FC~0000D7FF)

T1024~4095(#0000BE40~0000BEFF･ #0000E400~0000EFFF)

C1024~4095(#0000BF40~0000BFFF･ #0000F400~0000FFFF)

HR(#0000D800~0000D9FF)

WR(#0000DE00~0000DFFF)

D24576~32767(#00016000~00017FFF)

EM 뱅크 지정(#00018000~000F7FFF)

E32766~E32767(#000FFFFE~000FFFFF)

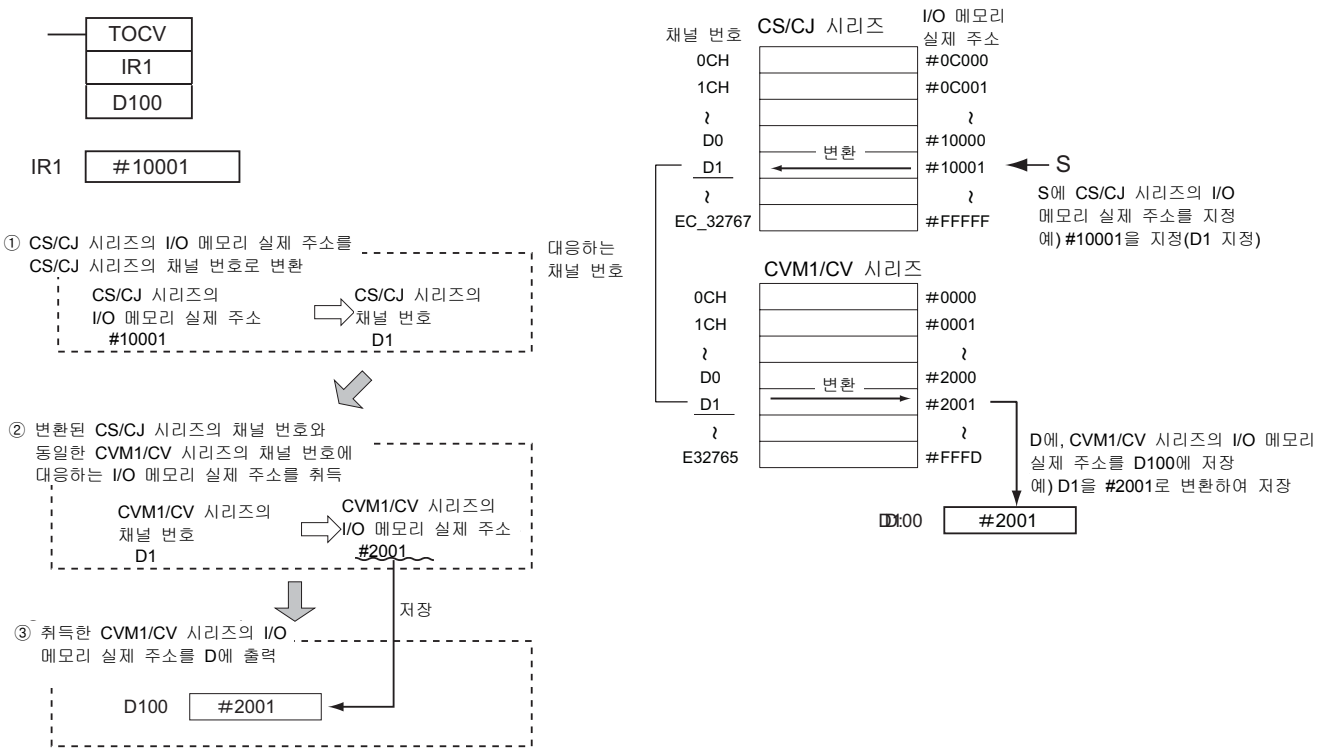
· S에 IR 이외의 영역이 지정된 경우 에러 플래그가 ON됩니다.

기능

TOCV 명령은 입력 조건이 ON일 때 다음 조작을 실행합니다.

1. S로 지정한 CS/CJ 시리즈의 I/O 메모리 실제 주소를 CS/CJ 시리즈의 채널 번호로 일단 변환합니다.
2. 변환된 CS/CJ 시리즈의 채널 번호와 동일한 CVM1/CV 시리즈의 채널 번호에 대응하는 I/O 메모리 실제 주소를 취득합니다.
3. 취득한 CVM1/CV 시리즈의 I/O 메모리 실제 주소를 D에 출력합니다.

예)



주: CS/CJ 시리즈에 있고 CVM1/CV 시리즈에는 없는 채널 번호의 경우, 에러 플래그가 ON이 되어 변환되지 않습니다.

힌트

이 명령 실행 후의 기본적인 사용 방법은 다음과 같습니다.

1. 이 명령을 통해 지정 주소에 저장한 CVM1/CV 시리즈의 I/O 메모리 실제 주소의 데이터를 CX-Programmer를 사용하여 CVM1/CV 시리즈에 전송합니다.
2. CVM1/CV 시리즈 측에서 인덱스 레지스터를 간접 지정(,IR 지정)하거나 간접 DM(* DM)을 BIN 지정하면 원래의 CS/CJ 시리즈와 동일한 채널 번호를 지정할 수 있습니다.

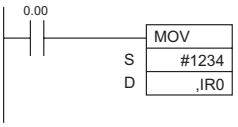
샘플 프로그램

CS/CJ 시리즈에서 인덱스 레지스터를 간접 지정(,IR 지정)한 프로그램을 CVM1/CV 시리즈용으로 치환하는 경우

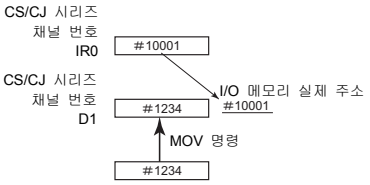
1. TOCV 명령으로 S에 IR(인덱스 레지스터)을 지정하고, 그 안의 I/O 메모리 실제 주소를 CVM1/CV 시리즈의 I/O 메모리 실제 주소로 변환합니다.
2. 해당 데이터를 CVM1/CV 시리즈의 I/O 메모리로 전송합니다.
3. CVM1/CV 시리즈의 프로그램 내에서 해당 데이터를 사용합니다.

TOCV

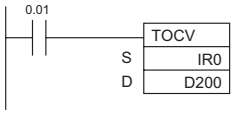
● CS/CJ 시리즈의 프로그램
(인덱스 레지스터 간접 지정을 사용한 프로그램)



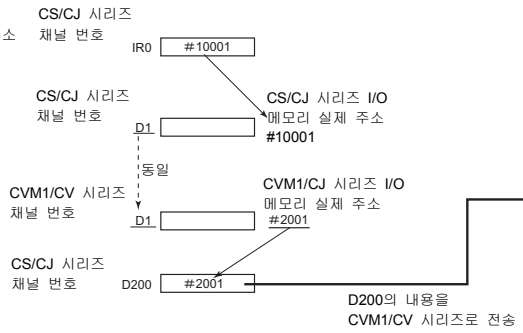
예) IR0 내의 값이 #10001인 경우, I/O 메모리 실제 주소 #10001에 대응하는 채널 번호는 D1이므로 D1에 #1234를 전송합니다.



● CS/CJ 시리즈의 프로그램

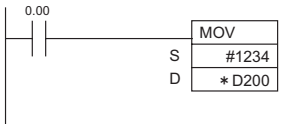


예) IR0 내의 값이 #10001인 경우, CS/CJ 시리즈의 I/O 메모리 실제 주소 #10001에 대응하는 CS/CJ 시리즈의 채널 번호는 D1이므로 그와 동일한 CVM1/CV 시리즈의 D1의 I/O 메모리 실제 주소 #2001을 D200에 저장합니다.



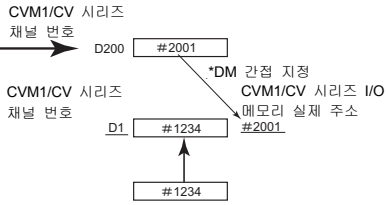
D200의 내용을 CVM1/CV 시리즈로 전송합니다.

● CVM1/CV 시리즈의 프로그램



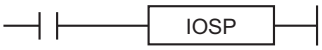
예) CVM1/CV 시리즈 측에서 MOV 명령으로 D200을 전송 대상으로 하여 BIN 지정의 간접 DM 지정을 사용하면 D1에 #1234를 전송할 수 있습니다.

PLC 시스템 설정
간접 DM 데이터: BIN 지정*DM 지정을 사용하는 경우 DM 내의 데이터를 I/O 메모리 실제 주소로 하여 해당 I/O 메모리 실제 주소에 대응하는 I/O 메모리 주소를 지정합니다.



IOSP/IORS

명령 이름	니모닉	동작 옵션	평션 번호	기능
주변 서비스 금지	IOSP	@IOSP	287	병렬 처리 모드 또는 주변 서비스 우선 모드에서 명령 실행 중인 주변 서비스를 금지합니다.
주변 서비스 금지 해제	IORS	—	288	병렬 처리 모드 또는 주변 서비스 우선 모드에서 IOSP(주변 서비스 실행 금지) 명령에 의해 지정된 주변 서비스 실행 금지를 해제합니다.

심볼	IOSP	IORS
		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

조건 플래그의 동작

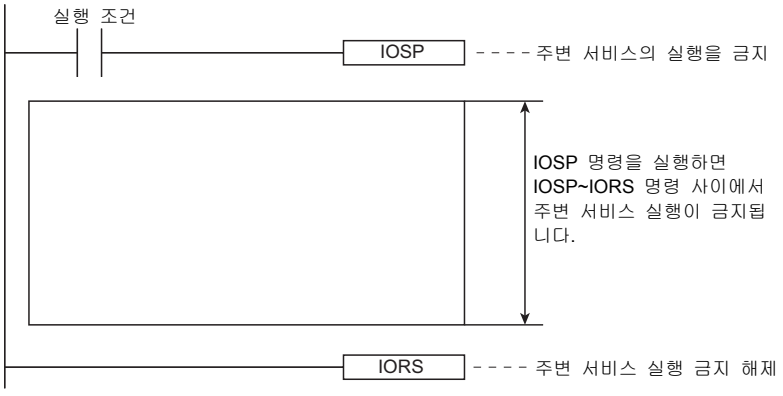
이름	라벨	내용
에러 플래그	P_ER	· 인터럽트 태스크에서 실행했을 때 ON · 그 이외일 때 OFF

기능

■ IOSP

병렬 처리 모드(I/O 메모리 동기 액세스 또는 비동기 액세스)로 사이클릭 태스크 내에서 사용하며, 다음과 같은 주변 서비스를 모두 금지합니다. 금지 상태는 IORS(주변 서비스 금지 해제) 명령으로 해제합니다.

- 고기능 I/O 유니트의 이벤트 서비스
- CPU 고기능 유니트의 이벤트 서비스
- 페리페럴 포트 서비스
- RS-232C 포트 서비스
- INNER 보드(CS 시리즈만 해당)의 이벤트 서비스
- 통신 포트 번호(내부 논리 포트)를 사용한 이벤트 서비스(명령의 백그라운드 처리 포함)



IOSP/IORS

이 명령을 통한 주변 서비스에 대한 실행 금지 상태는 IORS 명령의 실행, END(001) 명령의 실행 또는 운전 정지에 의해 해제됩니다.

주:

- 이 명령은 인터럽트 태스크 내에서 실행할 수 없습니다. 인터럽트 태스크 내에서 실행한 경우, 에러가 되어 에러 플래그가 ON 이 됩니다.
- 이 명령은 태스크 실행 중에 주변 서비스 실행 금지를 수행할 수 없습니다.

  여러 태스크에서 주변 서비스의 실행을 금지하려는 경우는 각각의 태스크에 이 명령을 삽입하십시오.

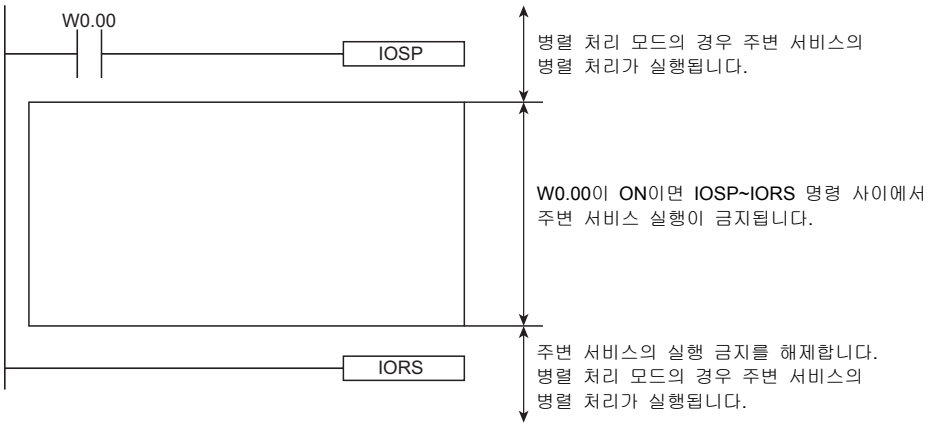
■ IORS

사이클릭 태스크 내에서 사용하여 IOSP(주변 서비스 실행 금지) 명령에 의해 지정된 주변 서비스 실행 금지 상태를 해제합니다.

주 1: 이 명령은 입력 조건(전원 흐름)이 필요하지 않습니다. 항상 ON 입력에서 사용하십시오.

주 2: 이 명령은 인터럽트 태스크 내에서 실행할 수 없습니다. 인터럽트 태스크 내에서 실행한 경우, 에러가 되어 에러 플래그가 ON이 됩니다.

샘플 프로그램



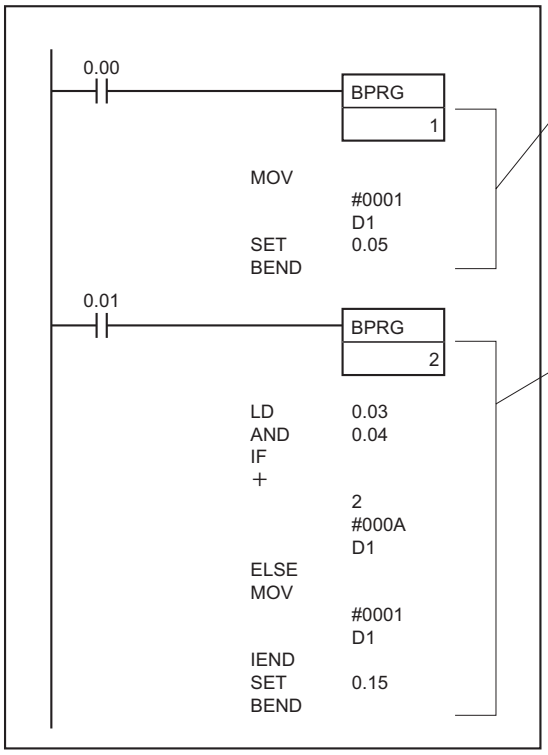
블록 프로그램 명령

니모닉	명령 이름	펑션 번호	페이지
BPRG	블록 프로그램	096	3-908
BEND	블록 프로그램 종료	801	
BPPS	블록 프로그램 일시 정지	811	3-910
BPRS	블록 프로그램 재기동	812	
EXIT	조건부 종료	806	3-912
EXIT NOT	조건부 종료(부정)	806	
IF	조건 분기 블록	802	3-914
IF NOT	조건 분기 블록(부정)	802	
ELSE	조건 분기 거짓 블록	803	
IEND	조건 분기 블록 종료	804	
WAIT	1스캔 조건 대기	805	3-917
WAIT NOT	1스캔 조건 대기(부정)	805	
TIMW	타이머 대기	813	3-920
TIMWX		816	
CNTW	카운터 대기	814	3-923
CNTWX		818	
TMHW	고속 타이머 대기	815	3-925
TMHWX		817	
LOOP	반복 블록	809	3-927
LEND	반복 블록 종료	810	
LEND NOT	반복 블록 종료(부정)	810	

블록 프로그램이란?

- CS/CJ 시리즈에서는 '블록 프로그램'이라는 영역을 사용자 프로그램 전체(전체 태스크 공통)에서 최대 128개를 작성할 수 있습니다.
- 블록 프로그램 영역은 하나의 입력 조건(전원 흐름)에 의해서만 기동됩니다. 기동 후, BPRG~BEND까지의 명령이 무조건 실행됩니다. (블록 프로그램 영역 내의 명령은 BPRG 명령을 제외하고 입력 조건(전원 흐름)과 관계 없이 실행됩니다.) 따라서 동일한 입력 조건 ON에서 동작시키는 명령을 이 블록 프로그램으로 묶을 수 있습니다.
- 전체를 하나의 블록으로 간주하여 래더를 통해 기동시키고 니모닉을 사용하여 그 안의 명령을 기술합니다. 따라서 블록 프로그램은 래더와 니모닉을 조합한 형태의 프로그램 영역이 됩니다.
- 블록 프로그램을 사용하면 래더로는 기술하기 어려운 조건 분기 또는 공정 진행 등의 논리 흐름을 간단하게 작성할 수 있습니다.

예



블록 프로그램 영역 번호 1

왼쪽 예에서는 0.00이 ON일 때 블록 프로그램 1을 실행

MOV, SET의 각 명령은 모두 무조건 실행되고 블록 프로그램 1이 종료됩니다.

블록 프로그램 영역 번호 2

0.01이 ON일 때 블록 프로그램 2를 실행

0.03이 ON이고 0.04가 ON이면 BIN 가산 (2CH＋#000A→D1)을 실행하고, 그렇지 않은 경우는 #0001을 D1에 전송합니다.
이어서 무조건 0.15를 설정하고 블록 프로그램 2가 종료됩니다.

태스크와의 관계

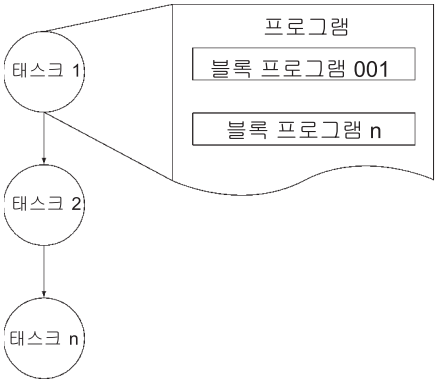
블록 프로그램은 태스크의 하위에 위치하며, 태스크에 의해 분할된 큰 단위의 프로그램을 다시 동일한 입력 조건(ON 조건)에서 동작하는 작은 단위의 블록으로 나눠서 실행을 제어할 때 사용합니다.

태스크와 마찬가지로 기동되지 않은 블록은 명령 실행 시간이 걸리지 않으므로 적절하게 사용하면 사이클 시간이 단축됩니다.

또한 태스크와 마찬가지로 블록 프로그램 내에서 기동 중인 다른 블록을 일시 정지/재기동시킬 수도 있습니다.

단, 태스크와 달리 블록 프로그램의 경우 일부 블록 프로그램 명령(IF, WAIT, EXIT, LEND) 이외에는 입력 조건을 사용할 수 없습니다(모두 실행됨). 또한 블록 프로그램 내에서는 입력의 기동/종료를 검출하는 명령 등, 일부 명령을 사용할 수 없으므로 주의하십시오.

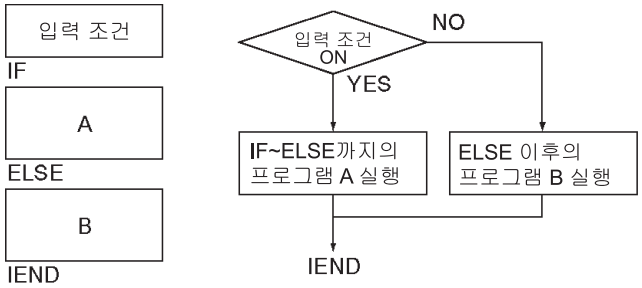
블록 프로그램은 사이클릭 태스크와 인터럽트 태스크에서 모두 사용할 수 있습니다. 단, 블록 프로그램의 번호 0~127은 태스크 간에 중복해서 사용할 수 없습니다.



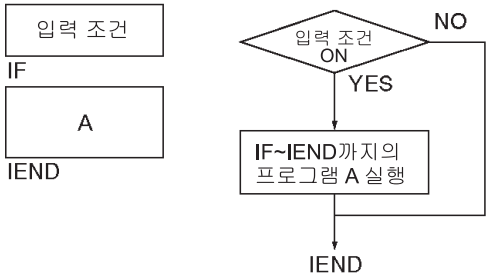
블록 프로그램 명령 사용 방법

- 블록 내에서 입력 조건(또는 접점)에 따른 실행/비실행은 기본적으로 IF/ELSE/IEND 명령을 사용합니다.

  A 또는 B 실행 시: IF A ELSE B IEND 순서



  A 또는 없음으로 전환 시: IF A IEND 순서



- 입력 조건(또는 접점)이 성립할 때까지 대기하고 입력 조건(또는 접점)이 성립되면 그 이후를 실행하는 경우(공정을 진행할 때 등) WAIT 명령을 사용합니다.
- 타임업까지 대기하고 타임업이 되면 그 이후를 실행하는 경우(타이머에 의해 공정 진행을 수행할 때 등) TIMW/TIMWX 명령 또는 TMHW/TMHWX 명령을 사용합니다.
- 카운트 업까지 대기하고 카운트 업이 되면 그 이후를 실행하는 경우(카운터에 의해 공정 진행을 수행할 때 등) CNTW/CNTWX 명령을 사용합니다.
- 블록 프로그램 내에서 입력 조건(또는 접점)이 성립될 때까지 반복 처리하고 싶은 경우는 LOOP/LEND 명령을 사용합니다.
- 입력 조건(또는 접점) 성립에 의해 블록 프로그램을 강제 종료시킬 때는 EXIT 명령을 사용합니다.

· 블록 프로그램 내에서 실행 중인 다른 블록 프로그램을 일시 정지시킬 때는 BPPS 명령을 사용하고, 재개시킬 때는 BPRS 명령을 사용합니다.

■ 블록 프로그램 내에서 입력 조건을 가질 수 있는 명령

블록 프로그램 내에서 입력 조건을 가질 수 있는 명령은 다음과 같습니다.

분류	이름	니모닉
블록 프로그램 명령	조건 분기 블록	IF
	1스캔 조건 대기	WAIT
	조건부 종료	EXIT
	반복 블록 종료	LEND
비블록 프로그램 명령	조건부 점프	CJP
	조건부 점프(부정)	CJPN

블록 프로그램 내에서 사용상의 제한이 있는 명령어

다음 명령어는 블록 프로그램 영역에서 단독으로 사용할 수 없습니다.

반드시 블록 프로그램 명령(IF, WAIT, EXIT, LEND) 또는 CJP, CJPN 명령과 조합하여 입력 조건으로 사용하십시오.

단독 또는 다른 명령과 조합하여 사용한 경우에는 이들 명령의 동작이 보장되지 않으므로 주의하십시오.

■ 사용상의 제한이 있는 명령

(IF, WAIT, EXIT, LEND의 입력 조건으로만 사용하는 명령)

니모닉	명령 이름
LD/LD NOT	로드/로드 부정
AND/AND NOT	논리곱/부정 논리곱
OR/OR NOT	논리합/부정 논리합
UP	P.F. 기동 미분
DOWN	P.F. 종료 미분
>, <, =, > =, < =, < >(S) (L)	심볼 표기의 다음 단계 연결형 비교 명령
LD TST/TST NOT	로드형 비트 테스트
AND TST/TST NOT	논리곱형 비트 테스트
OR TST/TST NOT	논리합형 비트 테스트
>$, <$, =$, > =$, < =$, < >$	문자열 비례 명령

○ 좋은 예

```
    ∫
LD    0.00          ┐IF의 입력
AND   1.00          │조건으로
TST   D0  #0010     ┘사용
IF
    ∫
```

× 나쁜 예

```
    ∫
LD    0.00          ┐MOV의 입력
AND   1.00          │조건으로
TST   D0  #0010     ┘사용 불가
MOV   #0000  10
    ∫
```

블록 프로그램 내에서 사용할 수 없는 명령어

블록 프로그램 영역 내에서는 다음과 같은 일부 명령을 사용할 수 없으므로 주의하십시오.

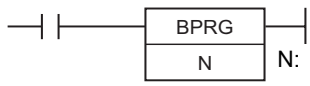
■ 블록 프로그램 내에서 사용할 수 없는 명령어

분류	니모닉	이름	대체 수단
시퀀스 출력	OUT	출력	SET, RSET 명령을 대신 사용하십시오.
	OUT NOT	부정 출력	
	DIFU	기동 미분	없음
	DIFD	종료 미분	없음
	KEEP	유지	없음
시퀀스 제어	FOR/NEXT	반복 시작/종료	블록 프로그램 명령의 LOOP/LEND 명령을 대신 사용하십시오.
	BREAK	루프 중단	
	IL/ILC	인터록/인터록 클리어	대신 블록을 분할하십시오.
	JMP0/JME0	복수 점프/복수 점프 끝	JMP/JME 명령 또는 CJP(N)/JME 명령을 사용하십시오(주).
	END	종료	블록 프로그램의 종료는 BEND 명령을 사용하십시오,
타이머/카운터 명령	TIM/TIMX	100ms 타이머	블록 프로그램 명령의 TIMW/TIMWX(타이머 대기), CNTW/CNTWX(카운터 대기), TMHW/TMHWX(고속 타이머 대기) 명령을 사용하십시오. 단, 타이머/카운터의 업까지 블록 프로그램 내의 다른 명령은 실행할 수 없습니다.
	TIMH/TIMHX	10ms 타이머	
	TMHH/TMHHX	1ms 타이머	
	TIMU/TIMUX	0.1ms 타이머	
	TMUH/TMUHX	0.01ms 타이머	
	TTIM/TTIMX	적산 타이머	
	TIML/TIMLX	장시간 타이머	
	MTIM/MTIMX	멀티 출력 타이머	
	CNT/CNTX	카운터	
	CNTR/CNTRX	가역 카운터	
서브루틴 명령	SBN/RET	서브루틴/서브루틴 반환	없음
데이터 이동 명령	SFT	이동 레지스터	다른 이동 명령
공정 진행 명령	STEP/SNXT	스텝 래더 영역 정의/스텝 래더 진행	블록 프로그램 명령의 WAIT(1스캔 조건 대기) 명령
데이터 제어 명령	PID	PID 연산	없음
고장 진단 명령	FPD	고장 지점 검출	없음
동작 옵션의 미분 동작 부가	@×××	입력 미분(기동)형 명령	없음
	%×××	입력 미분(종료)형 명령	없음

주: JMP/JME 명령은 사용할 수 있습니다. 단, 입력 조건은 없고 무조건 JME로 점프합니다.
CJP/JME 명령도 사용할 수 있습니다. 입력 조건은 뒤에 설명하며, 평상시대로 입력 조건에 따라 JME로 점프합니다.

BPRG/BEND

명령 이름	니모닉	동작 옵션	평션 번호	기능
블록 프로그램	BPRG	–	096	지정된 번호의 블록 프로그램 영역의 선두를 나타냅니다.
블록 프로그램 종료	BEND	–	801	블록 프로그램 영역의 종료를 나타냅니다.

	BPRG	BEND
심볼	 N: 블록 프로그램 번호	BEND

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	불가능	가능	가능	가능	불가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	블록 프로그램 번호	숫자	1

N: 블록 프로그램 번호
10진수 0~127

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
BPRG	N	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–

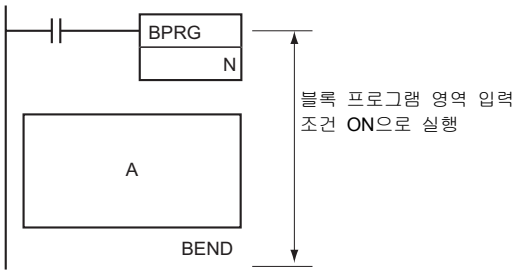
조건 플래그의 동작

이름	라벨	내용	
		BPRG	BEND
에러 플래그	P_ER	· 이미 블록 프로그램을 실행 중일 때 이 명령이 실행된 경우 ON · N의 데이터가 0~127의 범위 안에 없을 때 ON · 블록 프로그램 번호가 이중으로 사용된 경우 ON · 그 이외일 때 OFF	· 블록 프로그램이 실행되지 않을 때 이 명령이 실행된 경우 ON · 그 이외일 때 OFF

기능

BPRG 명령의 입력 조건이 ON이면 N으로 지정한 번호의 블록 프로그램 영역(BPRG 명령~BEND 명령)을 실행합니다.

BPRG 명령에서 BEND 명령까지의 각 명령은 입력 조건이 ON인 상태에서 실행됩니다(무조건).



BPRG 명령의 입력 조건이 OFF이면 N으로 지정한 번호의 블록 프로그램 영역을 실행하지 않습니다. 이 경우 블록 프로그램 영역 내의 명령 실행 시간이 걸리지 않습니다.

블록 프로그램 영역은 해당 BPRG 명령의 실행 조건이 ON이어도 다른 블록 프로그램 영역 내의 BPPS(블록 프로그램 일시 정지) 명령을 통해 실행을 정지시킬 수 있습니다.

사용상의 주의 사항

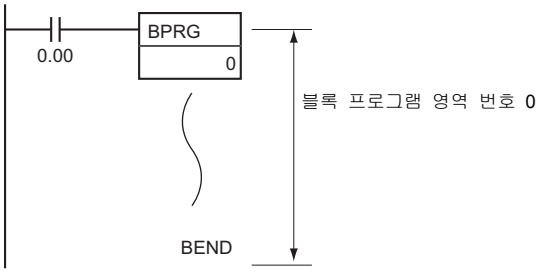
- 블록 프로그램 번호는 사용자 프로그램 전체에서 이중으로 사용할 수 없습니다.
- 블록 프로그램 영역은 중복할 수 없습니다.

예

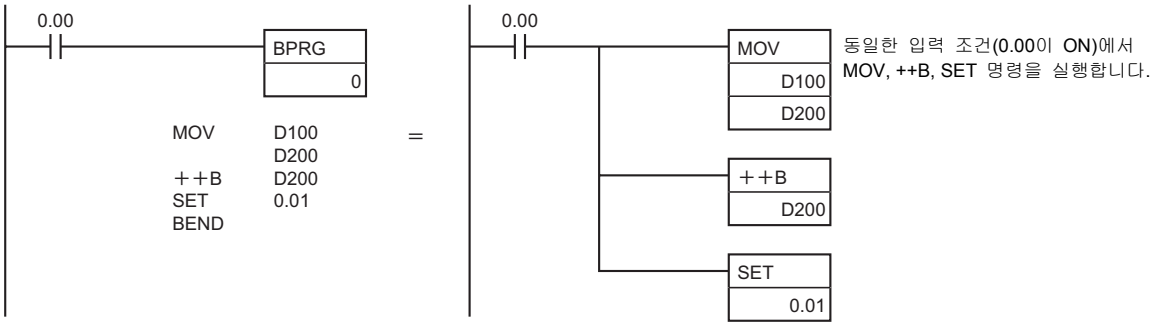
```
BPRG  0
  ≀
BPRG  1
  ≀
BEND
BEND
```

- IL-ILC 명령에 의한 인터록 시(IL 명령의 입력 조건이 OFF일 때)에는 BPRG~BEND 사이의 명령이 실행되지 않습니다.
- BPRG 명령과 BEND 명령은 동일 태스크 내에 배치하십시오.

샘플 프로그램



0.00이 ON이면 BPRG 명령부터 BEND 명령까지의 블록 프로그램(블록 번호 0)을 실행합니다.
0.00이 OFF일 때는 블록 번호 0의 블록 프로그램이 실행되지 않습니다.



BPPS/BPRS

명령 이름	니모닉	동작 옵션	평션 번호	기능
블록 프로그램 일시 정지	BPPS	–	811	실행 중인 지정된 블록 프로그램을 일시 정지시킵니다.
블록 프로그램 재기동	BPRS	–	812	정지 중인 지정된 블록 프로그램 번호의 블록 프로그램을 재기동시켜서 실행합니다.

심볼	BPPS	BPRS
	BPPS 블록 프로그램 번호 N	BPRS 블록 프로그램 번호 N

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	가능	가능	가능*1	가능*1	불가능

*1: 서브루틴 프로그램 영역, 인터럽트 태스크 프로그램 영역 내의 블록 프로그램 영역에서 사용 가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	블록 프로그램 번호	숫자	1

N: 블록 프로그램 번호
10진수 0~127

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 블록 프로그램 영역에 없을 때 ON · N의 데이터가 0~127의 범위 안에 없을 때 ON · 그 이외일 때 OFF

주: 블록 프로그램 영역에 없을 때 또는 N의 데이터가 0에서 127의 범위에 없을 때 에러가 되어 에러 플래그가 ON이 됩니다.

기능

■ BPPS

N으로 지정한 블록 프로그램을 일시 정지시킵니다. (N을 지정한 BPRG 명령의 입력 조건이 ON이어도 블록 프로그램 번호 N은 실행되지 않습니다.)

정지된 블록 프로그램은 BPRS(블록 프로그램 재기동) 명령을 통해 재기동되지 않는 한 정지된 그대로의 상태가 됩니다.

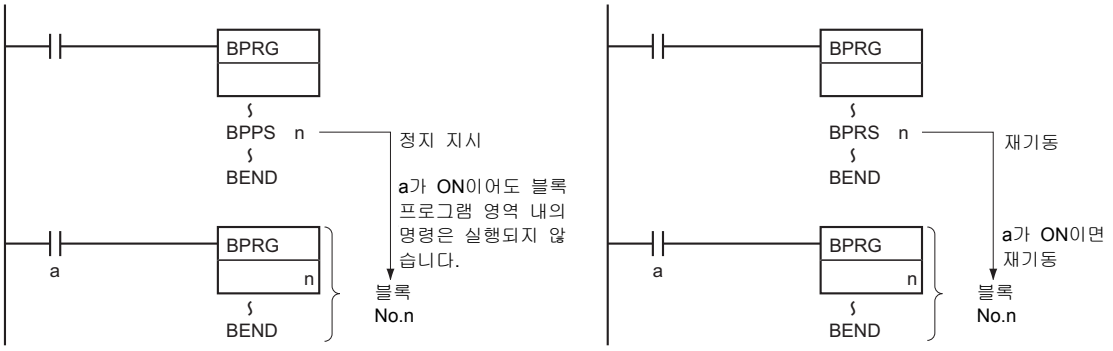
주: 이 명령을 사용하여 자체 블록 프로그램을 일시 정지시킬 수 있습니다. 이 경우, 다른 블록 프로그램 영역의 BPRS 명령에 의해 재기동하면 일시 정지한 다음 명령부터 재개됩니다.

■ BPRS

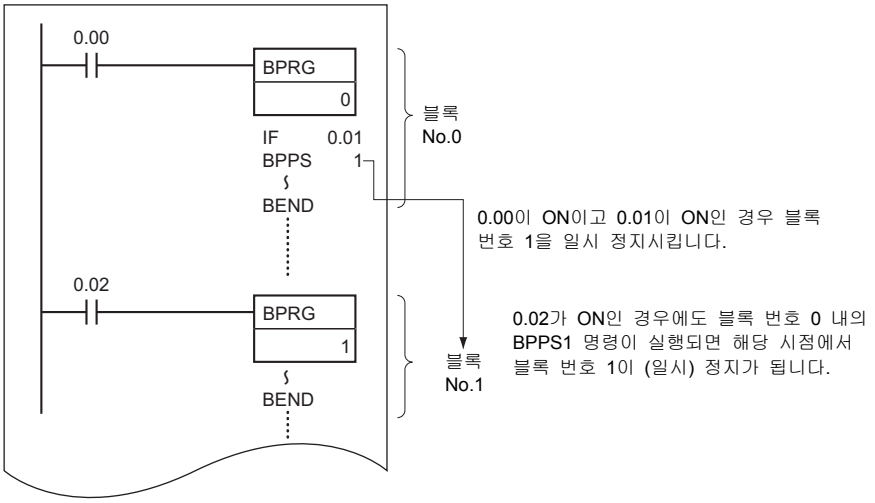
BPPS 명령에 의해 일시 정지되어 있으며 N으로 지정된 블록 프로그램을 재기동시킵니다. (N을 지정한 BPRG 명령의 입력 조건이 ON이면 블록 프로그램 번호 N이 실행됩니다.)

힌트

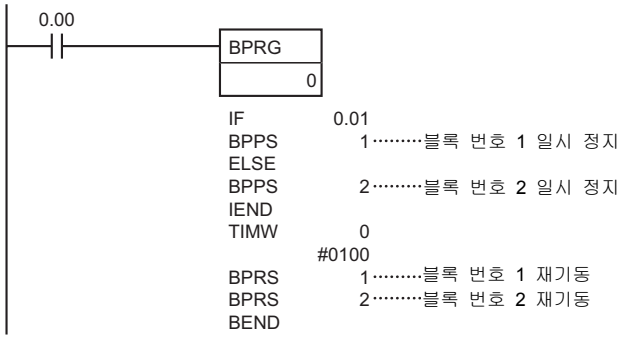
· BPPS 명령을 통해 일시 정지된 블록 프로그램에서도 TIMW/TIMWX 명령 및 TMHW/TMHWX 명령으로 기동된 타이머(타이머 번호 0~2047)는 현재값을 계속 갱신합니다.



샘플 프로그램



주: BPPS 명령 다음에 지정된 블록 번호가 있는 경우, 해당 블록은 실행되지 않습니다. 지정된 블록 번호 다음에 BPPS 명령이 있는 경우 다음 사이클에서 그 블록이 정지됩니다.



0.00이 ON이면 BPRG0이 실행됩니다.
접점 0.01이 ON이면 블록 번호 1의 블록 프로그램을 정지하고, OFF이면 블록 번호 2의 블록 프로그램을 정지합니다.
0.00이 ON이 되고 10초 후에 정지된 블록 번호 1의 블록 프로그램 또는 블록 번호 2의 블록 프로그램이 다시 시작됩니다.

코딩

주소	명령	데이터
000200	LD	0.00
000201	BPRG	0
000202	IF	0.01
000203	BPPS	1
000204	ELSE	—
000205	BPPS	2
000206	IEND	—
000207	TIMW	0
		#0100
000208	BPRS	1
000209	BPRS	2
000210	BEND	—

EXIT/EXIT NOT

명령 이름	니모닉	동작 옵션	펑션 번호	기능
조건부 종료	EXIT	–	806	입력 조건이 ON인 경우(피연산자를 지정하지 않는 경우) 또는 지정된 접점이 ON인 경우(피연산자를 지정한 경우), EXIT 명령부터 BEND 명령까지의 프로그램을 실행하지 않고 블록 프로그램을 종료합니다.
조건부 종료(부정)	EXIT NOT	–	806	지정된 접점이 OFF일 때 EXIT NOT 명령부터 BEND 명령까지의 프로그램을 실행하지 않고 블록 프로그램을 종료합니다.

심볼	피연산자를 지정하지 않는 경우(입력 조건 사용) 입력 조건 EXIT	피연산자를 지정하는 경우 EXIT 릴레이 번호 R EXIT NOT 릴레이 번호 R

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	가능	가능	가능*1	가능*1	불가능

*1: 서브루틴 프로그램 영역, 인터럽트 태스크 프로그램 영역 내의 블록 프로그램 영역에서 사용 가능

피연산자 설명

피연산자	내용	데이터 형식	크기
R	릴레이 번호	BOOL	1

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
R	○	○	○	○	○	○	○*1	○*1	–	–	–	–	–	○	○	○	○	–

*1: CJ2-CPU 유니트만 사용 가능

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 블록 프로그램 영역에 없을 때 ON · 그 이외일 때 OFF

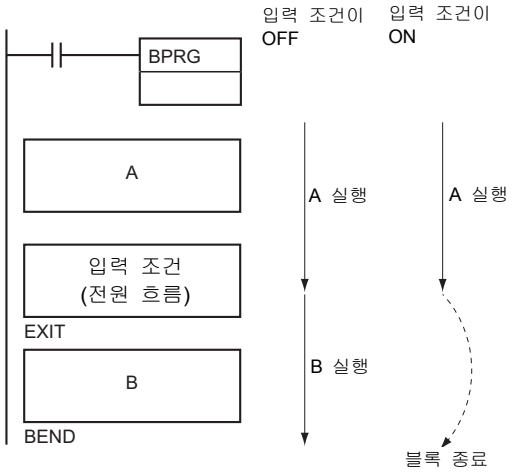
기능

■ 피연산자를 지정하지 않는 경우

EXIT 명령 앞에 입력 조건을 설정합니다. 입력 조건은 반드시 LD 명령부터 시작합니다.

입력 조건이 OFF이면 EXIT 이후를 실행합니다.

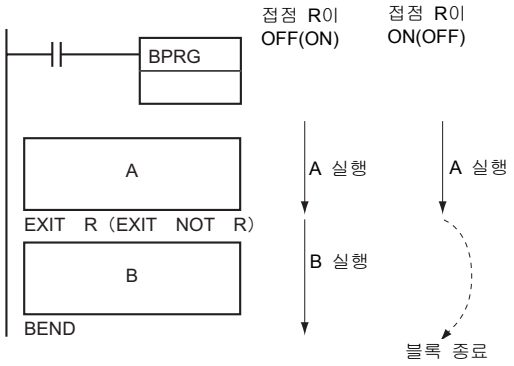
입력 조건이 ON이면 블록 프로그램을 종료합니다. (EXIT 이후 BEND까지의 모든 명령을 실행하지 않습니다.)



■ 피연산자를 지정하는 경우

R로 지정한 접점이 OFF(ON)이면 EXIT 이후를 실행합니다.

R로 지정한 접점이 ON(OFF)이면 블록 프로그램을 종료합니다. (EXIT 이후 BEND까지의 모든 명령을 실행하지 않습니다.)

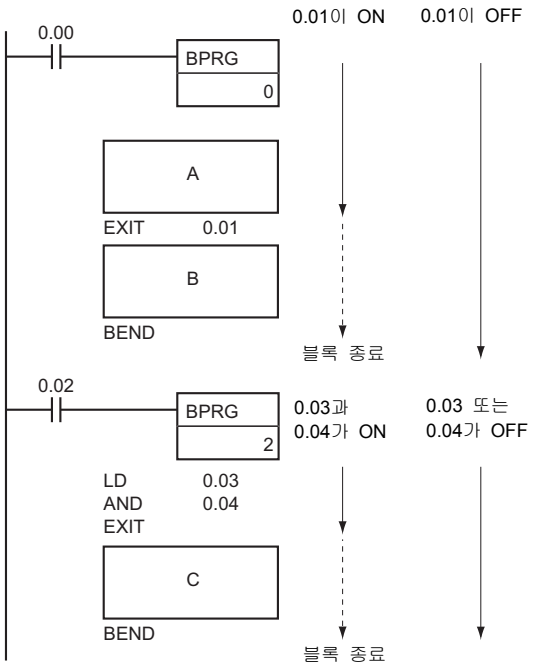


주: 괄호 안은 EXIT NOT 명령인 경우입니다.

샘플 프로그램

0.00이 ON이면 BPRG0이 실행됩니다. 프로그램 A가 실행된 후 EXIT 명령에서 접점 0.01이 ON이면 B는 실행되지 않고 블록 프로그램이 종료됩니다. OFF일 때는 B가 실행됩니다.

EXIT 명령은 IF~IEND 명령과 유사한 명령이지만 EXIT 명령의 경우 접점 조건이 ON이면 BEND까지의 프로그램이 실행되지 않으므로 그만큼 처리 시간이 단축됩니다.



코딩

주소	명령	피연산자
000200	LD	0.00
000201	BPRG	0
⋮	A	⋮
000210	EXIT	0.01
⋮	B	⋮
000220	BEND	—
000221	LD	0.02
000222	BPRG	2
000223	LD	0.03
000224	AND	0.04
000225	EXIT	—
⋮	C	⋮
000230	BEND	—

IF/IF NOT/ELSE/IEND

명령 이름	니모닉	동작 옵션	평션 번호	기능
조건 분기 블록	IF	–	802	입력 조건(피연산자를 지정하지 않는 경우) 또는 지정된 접점(피연산자를 지정하는 경우)이 OFF이면 다음 명령 이후를 실행하고, ON이면 ELSE 명령까지의 명령을 무시합니다.
조건 분기 블록 (부정)	IF NOT	–	802	지정된 접점이 ON이면 다음 명령 이후를 실행하고 OFF이면 ELSE 명령까지의 명령을 무시합니다.
조건 분기 거짓 블록	ELSE	–	803	IF 조건이 거짓일 때 실행되는 블록의 선두를 나타냅니다.
조건 분기 블록 종료	IEND	–	804	조건 분기 블록 영역의 종료를 나타냅니다.

심볼	IF 명령의 피연산자를 지정하지 않는 경우(입력 조건 사용)	IF 명령(IF NOT 명령)의 피연산자를 지정하는 경우
	입력 조건 IF : ELSE : IEND	IF 릴레이 번호 R IF NOT 릴레이 번호 R : ELSE : IEND

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	가능	가능	가능*1	가능*1	불가능

*1: 서브루틴 프로그램 영역, 인터럽트 태스크 프로그램 영역 내의 블록 프로그램 영역에서 사용 가능

피연산자 설명

피연산자	내용	데이터 형식	크기
R	릴레이 번호	BOOL	1

■ 영역 종류

내용		채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
		CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
IF, IF NOT	R	○	○	○	○	○	○	○*1	○*1	–	–	–	–	–	○	○	○	○	–

*1: CJ2-CPU 유니트만 사용 가능

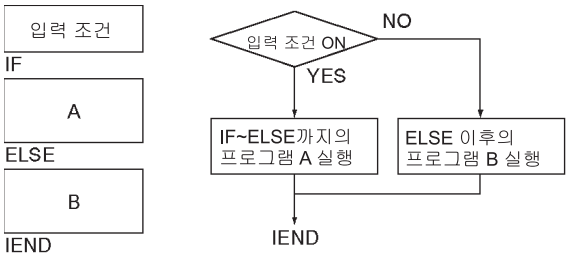
조건 플래그의 동작

이름	라벨	내용	
		IF/IF NOT	ELSE/IEND
에러 플래그	P_ER	· 블록 프로그램 영역에 없을 때 ON · 조건 분기의 중첩이 254개 이상일 때 ON · 그 이외일 때 OFF	· 블록 프로그램 영역에 없을 때 ON · 그 이외일 때 OFF

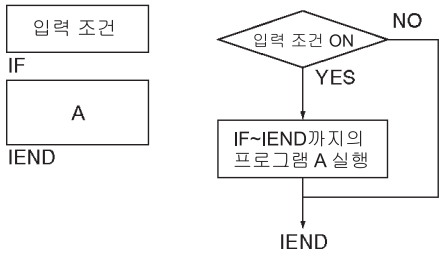
기능

■ IF 명령의 피연산자를 지정하지 않는 경우

IF 명령 앞에 입력 조건을 설정합니다. 입력 조건은 반드시 LD 명령부터 시작합니다. 입력 조건이 ON이면 다음 명령 이후(IF 이후 ELSE까지)를 실행하고, 입력 조건이 OFF이면 ELSE 명령 이후(ELSE 이후 IEND까지)의 명령을 실행합니다.

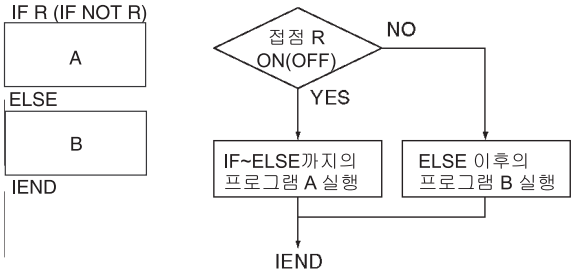


ELSE 명령을 생략한 경우, 입력 조건이 ON이면 다음 명령 이후(IF 이후 IEND까지)를 실행하고, 입력 조건이 OFF이면 IEND 명령 이후의 명령을 실행합니다.

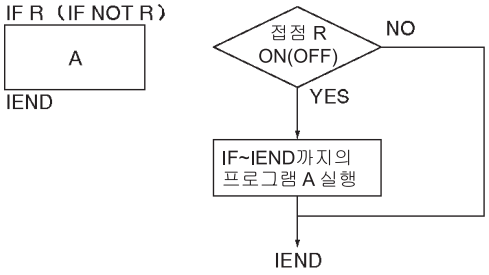


■ IF 명령(IF NOT 명령)의 피연산자를 지정하는 경우

IF 명령(IF NOT 명령)의 피연산자로 접점 R를 지정합니다. R로 지정한 접점이 ON(OFF)이면 다음 명령 이후(IF 이후 IEND까지)를 실행하고, OFF(ON)이면 ELSE 명령 이후(ELSE 이후 IEND까지)의 명령을 실행합니다.



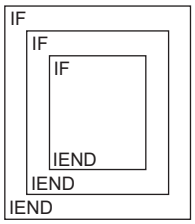
ELSE 명령을 생략한 경우, 접점 R이 ON(OFF)이면 다음 명령 이후(IF 이후 IEND까지)를 실행하고, 접점 R이 OFF(ON)이면 IEND 명령 이후의 명령을 실행합니다.



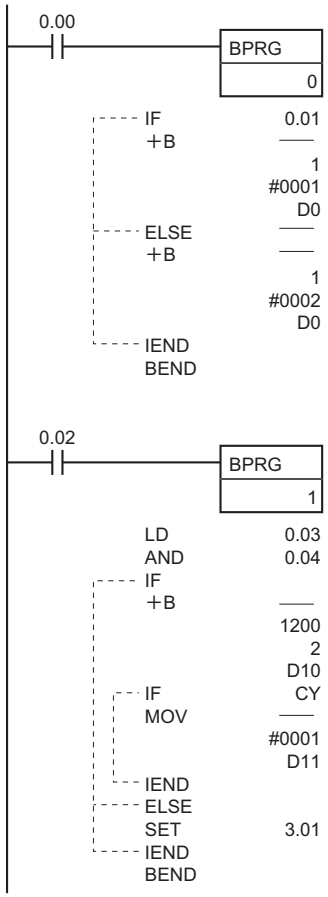
주: 설명 속의 괄호는 IF NOT 명령의 경우를 나타냅니다.

힌트

- 블록 프로그램 내에서는 각 명령이 무조건(입력 조건 없이) 실행됩니다. 입력 조건에 따라 실행/비실행을 전환하는 경우 이 명령을 사용합니다.
  프로그램 A와 B를 분기할 때: IF A ELSE B IEND
  프로그램 A와 없음을 전환할 때: IF A IEND
- 조건 분기 블록 중첩
  - IF~IEND의 블록 내에는 IF~IEND의 블록을 최대 253개까지 구성할 수 있습니다.



샘플 프로그램



코딩

주소	명령	데이터
000200	LD	0.00
000201	BPRG	0
000202	IF	0.01
000203	+B	—
		1
		#0001
		D0
000204	ELSE	—
000205	+B	—
		1
		#0002
		D0
000206	IEND	—
000207	BEND	—
000208	LD	0.02
000209	BPRG	1
000210	LD	0.03
000211	AND	0.04
000212	IF	—
000213	+B	—
		1200
		2
		D10
000214	IF	CY
000215	MOV	—
		#0001
		D11
000216	IEND	—
000217	ELSE	—
000218	SET	3.01
000219	IEND	—
000220	BEND	—

0.00이 ON이면 BPRG0이 실행됩니다.
0.01이 ON인 경우 1CH의 내용에 1이 가산(+B 실행)되어 결과가 데이터 메모리 D0에 출력됩니다.
0.01이 OFF인 경우 1CH의 내용에 2가 가산(+B 실행)되어 결과가 D0에 출력됩니다.

0.02가 ON이면 BPRG1이 실행됩니다.
0.03과 0.04가 모두 ON이면 1200CH의 내용과 2CH의 내용이 가산(+B 실행)되어 결과가 D10에 출력됩니다.
또한 자리올림이 있는 경우(CY 플래그가 ON)는 D11에 #0001이 전송(MOV 실행)됩니다. 0.03과 0.04 중 하나가 OFF이면 3.01이 ON됩니다 (SET 실행).

WAIT/WAIT NOT

명령 이름	니모닉	동작 옵션	펑션 번호	기능
1스캔 조건 대기	WAIT	−	805	입력 조건(피연산자를 지정하지 않는 경우) 또는 지정된 접점(피연산자를 지정하는 경우)이 ON이 될 때까지 WAIT 명령부터 BEND 명령까지의 프로그램을 실행하지 않습니다.
1스캔 조건 대기 (부정)	WAIT NOT	−	805	지정된 접점이 OFF가 될 때까지 WAIT NOT 명령부터 BEND 명령까지의 프로그램을 실행하지 않습니다.

심볼	피연산자를 지정하지 않는 경우(입력 조건 사용)	피연산자를 지정하는 경우
	입력 조건 WAIT	WAIT 릴레이 번호 R WAIT NOT 릴레이 번호 R

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	가능	가능	가능*1	가능*1	불가능

*1: 서브루틴 프로그램 영역, 인터럽트 태스크 프로그램 영역 내의 블록 프로그램 영역에서 사용 가능

피연산자 설명

피연산자	내용	데이터 형식	크기
R	릴레이 번호	BOOL	1

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
R	○	○	○	○	○	○	○*1	○*1	−	−	−	−	−	○	○	○	○	−

*1: CJ2-CPU 유니트만 사용 가능

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 블록 프로그램 영역에 없을 때 ON · 그 이외일 때 OFF

WAIT/WAIT NOT

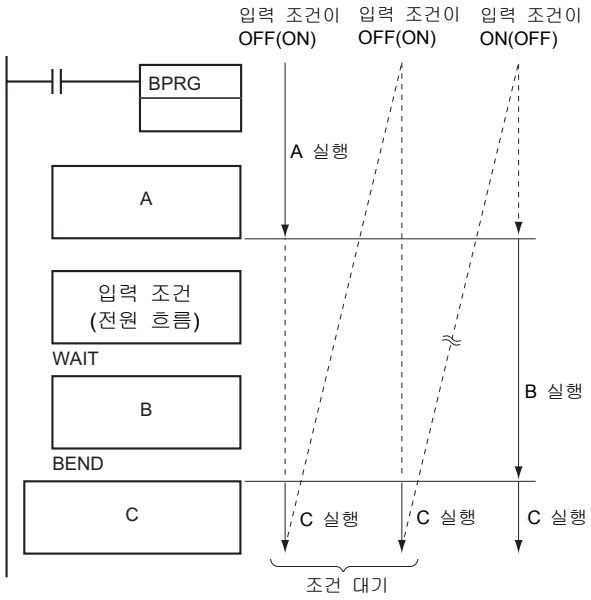
기능

■ 피연산자를 지정하지 않는 경우

WAIT 명령 앞에 입력 조건을 설정합니다. 입력 조건은 반드시 LD 명령부터 시작합니다.

입력 조건이 OFF이면 BEND 명령의 다음 명령으로 점프합니다. 다음 스캔부터는 블록 프로그램 내의 프로그램이 실행되지 않고 WAIT 명령의 입력 조건에 대한 조건 판단만 실행합니다.

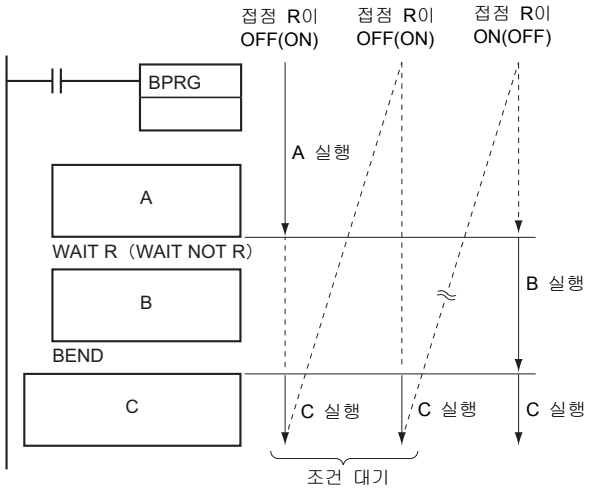
입력 조건이 ON이 되면 WAIT 명령부터 BEND 명령까지의 프로그램을 실행합니다.



■ 피연산자를 지정하는 경우

R로 지정한 접점이 OFF이면 BEND 명령의 다음 명령으로 점프합니다. 다음 스캔부터는 블록 프로그램이 WAIT 명령의 접점에 대한 ON/OFF 내용 판단만 실행합니다.

R로 지정한 접점이 ON(OFF)이면 WAIT 명령부터 BEND 명령까지의 프로그램을 실행합니다.



주: 괄호 안은 WAIT NOT 명령의 경우입니다.

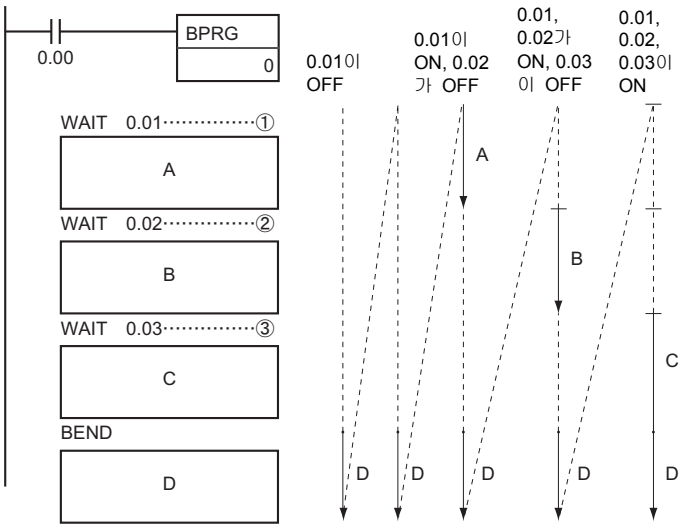
힌트

· WAIT 명령은 블록 프로그램 내에서 공정 진행 등을 수행할 때 사용합니다.

사용상의 주의 사항

· WAIT 명령에서는 입력 조건이 OFF이면 입력 조건이 ON이 될 때까지 이 명령 이외의 블록 프로그램의 명령을 실행하지 않는 대기 상태가 됩니다. 입력 조건이 ON이 된 시점에서 WAIT 명령 이후를 계속 실행합니다. 그러나 주변 툴 등에 의해 블록 프로그램이 포함된 태스크에 대해 온라인 편집을 실행한 경우, 이 명령에 의해 대기된 상태는 해제되고 다시 블록 프로그램 선두 영역부터 실행되므로 주의하십시오.

샘플 프로그램



0.00이 ON이면 BPRG00이 실행됩니다.

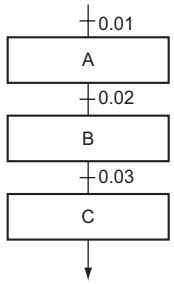
1) 최초의 WAIT 명령 ①에서 0.01이 OFF이면 A가 실행되지 않고 BEND로 점프합니다. 다음 사이클 이후, 0.01이 ON이 되는 것을 기다립니다. 0.01이 ON이 되면 A를 실행하고 다음 WAIT ②로 이동합니다.

2) 다음 WAIT 명령 ②에서 0.02가 OFF이면 B가 실행되지 않고 BEND로 점프합니다. 다음 사이클 이후, 0.02가 ON이 되는 것을 기다립니다. 0.02가 ON이 되면 B를 실행하고 다음 WAIT 명령 ③으로 이동합니다.

3) 다음 WAIT 명령 ③에서 동일하게 동작합니다.

접점 조건			프로그램 실행			
0.01	**0.02**	**0.03**	**0.00이 ON의 최초 사이클**	**다음 사이클**		
OFF	관계 없음	관계 없음	없음	없음 0.01 대기		0.01이 ON일 때 A →다음의 0.02 판단
ON	OFF	관계 없음	A	0.02 대기		0.02가 ON일 때 B →다음의 0.03 판단
ON	ON	OFF	A→B	0.03 대기	…	0.03이 ON일 때 C
ON	ON	ON	A→B→C	A→B→C		

이와 같이 WAIT 명령을 여러 개 사용하면 공정 진행 프로그램(A→B→C)을 실행할 수 있습니다.



TIMW/TIMWX

명령 이름	니모닉	동작 옵션	평션 번호	기능
타이머 대기	TIMW	–	813	지정된 타이머가 타임업될 때까지 다음 명령부터 BEND 명령까지의 프로그램을 실행하지 않습니다.
	TIMWX	–	816	

심볼	BCD 방식일 때		BIN 방식일 때	
	TIMW	N: 타이머 번호 S: 타이머 설정값	TIMWX	N: 타이머 번호 S: 타이머 설정값

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	가능	가능	가능*1	불가능	불가능

*1: 서브루틴 프로그램 영역 내의 블록 프로그램 영역에서 사용 가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	타이머 번호	TIMER	1
S	타이머 설정값	WORD	1

N: 타이머 번호
10진수 0~4095
S: 타이머 설정값
BCD: #0000~9999
BIN: 10진수 &0~65535 또는 16진수 #0000~FFFF

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	○	–	–	–	–	–	–	–	–	○	–	–	–	–
S	○	○	○	○		○	○	○	○	○	○	○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 블록 프로그램 영역에 없을 때 ON · N의 타이머 번호가 간접 IR 지정인 경우, 주소가 타이머의 현재값 주소가 아닐 때 ON · 타이머 현재값 갱신 방식이 BCD 방식이고 S의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF

기능

감산식 ON 딜레이 100ms 타이머입니다.

설정 시간은 다음과 같습니다.

- BCD 방식의 경우: 0~999.9초
- BIN 방식의 경우: 0~6553.5초

타이머 정도는 −0.01~0초입니다.

주: CS1D CPU 유니트에서는 타이머 정도가 ±(10ms + 사이클 시간)입니다.

블록 프로그램 기동 후 1회째 이 명령을 실행하는 경우

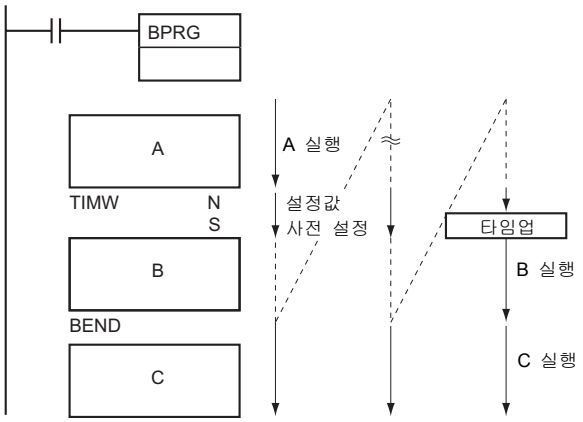
(1) 타임업 플래그를 리셋합니다.

(2) 타이머 설정값을 타이머 현재값 영역에 사전 설정합니다. (N으로 지정한 타이머 번호의 타이머가 기동 중이면 그대로 타이머 현재값을 갱신합니다.)

(3) 타이머 현재값을 갱신합니다.

타임업될 때까지 블록 프로그램은 이 명령만 실행하며 타이머 현재값을 갱신합니다.

타임업 후에는 타임업 플래그가 ON이 되고 다음 명령 이후가 실행됩니다.



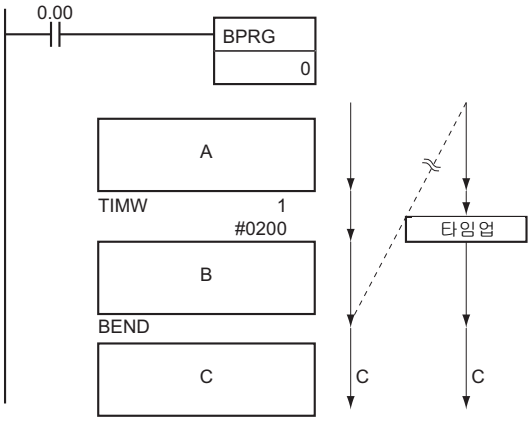
힌트

- 이 명령은 WAIT 명령의 입력 조건이 타이머(타임업)가 되는 명령입니다. 따라서 타이머에 의한 공정 진행 등에 사용할 수 있습니다.

사용상의 주의 사항

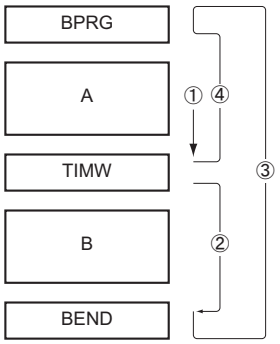
- TIMW 명령에서는 입력 조건의 타이머가 동작 중이면 타임업될 때까지 이 명령 이외의 블록 프로그램의 명령을 실행하지 않는 대기 상태가 됩니다. 타임업된 시점에서 TIMW 명령 이후를 계속 실행합니다. 그러나 주변 툴 등에 의해 블록 프로그램이 포함된 태스크에 대해 온라인 편집을 실행한 경우, 이 명령에 의해 대기된 상태는 해제되고 다시 블록 프로그램 선두 영역부터 실행되므로 주의하십시오.
- 타이머 번호가 0~2047인 경우 타이머 대기 명령은 태스크가 대기 중이어도 현재값을 갱신합니다. 타이머 번호가 2048~4095이면 태스크가 대기 중인 경우 현재값을 유지합니다.
- 타임업 플래그가 강제 세트되었을 때는 다음 명령 이후가 실행됩니다.
- 타임업 플래그가 강제 리셋되었을 때는 강제 리셋이 해제될 때까지 블록 프로그램이 이 명령만 실행합니다.
- 타이머 번호는 다른 타이머 명령과 공용입니다. 이들 명령에서 같은 타이머 번호를 동시에 동작시키면 오작동하므로 주의하십시오. 동시에 사용하면 프로그램 확인 시 '코일 이중 사용'이 표시됩니다. 또한 동시에 동작시키지 않는 조건이면 동일한 번호를 사용할 수 있습니다.
- 사이클 시간이 100ms 이상이면 정상적으로 동작하지 않으므로 주의하십시오.

샘플 프로그램



코딩

주소	명령	데이터
000200	LD	0.00
000201	BPRG	0
⋮	A	⋮
000210	TIMW	1
		#0200
⋮	B	⋮
000220	BEND	—



・TIMW 명령의 동작

0.00이 ON이면 BPRG0이 실행됩니다. 프로그램 A가 실행된 후 타이머 1에 설정값이 사전 설정되어 BEND로 점프합니다.

다음 사이클에서는 A가 실행되지 않고 TIMW 명령부터 실행됩니다. 그리고 타임업(20초)된 시점에서 프로그램 B가 실행됩니다.

타임업될 때까지 프로그램은 오른쪽 그림과 같이 ②→③→④→②를 반복합니다.

CNTW/CNTWX

명령 이름	니모닉	동작 옵션	펑션 번호	기능
카운터 대기	CNTW	–	814	지정된 카운터가 카운트 업될 때까지 다음 명령부터 BEND 명령까지의 프로그램을 실행하지 않습니다.
	CNTWX	–	818	

심볼	BCD 방식일 때		BIN 방식일 때	
	CNTW	N: 카운터 번호 S: 카운터 설정값 R: 카운트 입력	CNTWX	N: 카운터 번호 S: 카운터 설정값 R: 카운트 입력

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	가능	가능	가능*1	가능*1	불가능

*1: 서브루틴 프로그램 영역, 인터럽트 태스크 프로그램 영역 내의 블록 프로그램 영역에서 사용 가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	카운터 번호	COUNTER	1
S	카운터 설정값	WORD	1
R	카운트 입력	BOOL	1

N: 카운터 번호
10진수 0~4095
S: 카운터 설정값
BCD: #0000~9999
BIN: 10진수 &0~65535 또는 16진수 #0000~FFFF

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	○	–	–	–	–	–	–	–	○	–	–	–	–
S	○	○	○	○	○		○	○	○	○	○	○						
R							○*1	○*1	–	–	–	–		○	○	○	○	

*1: CJ2-CPU 유니트만 사용 가능

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 블록 프로그램 영역에 없을 때 ON · N의 카운터 번호가 간접 IR 지정인 경우, 주소가 카운터의 현재값 주소가 아닐 때 ON · 타이머 현재값 갱신 방식이 BCD 방식이고 S의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF

CNTW/CNTWX

기능

블록 프로그램 기동 후 1회째 이 명령을 실행하는 경우

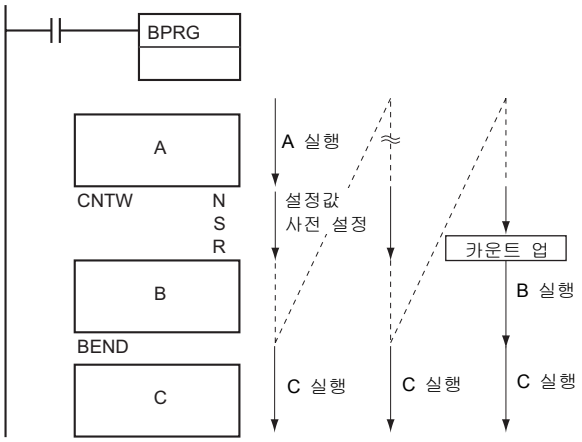
(1) 카운트 업 플래그를 리셋(=0)합니다.

(2) 카운터 설정값을 카운터 현재값 영역에 사전 설정합니다. (N으로 지정한 카운터 번호의 카운터가 기동 중이면 그대로 카운터 현재값을 갱신합니다.)

(3) 카운트 입력 R의 기동으로 카운터 현재값을 감산하여 갱신합니다.

카운트업될 때까지 블록 프로그램은 이 명령만 실행하며 카운터 현재값을 갱신합니다.

카운트 업 후에는 카운트 업 플래그가 ON이 되고 다음 명령 이후가 실행됩니다.



힌트

· 이 명령은 WAIT 명령의 입력 조건이 카운터(카운트 업)가 되는 명령입니다. 따라서 카운터에 의한 공정 진행 등에 사용할 수 있습니다.

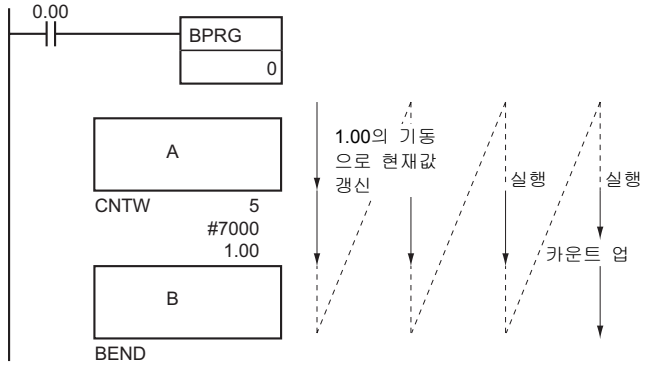
사용상의 주의 사항

· 카운트 업 플래그가 강제 세트되었을 때는 다음 명령 이후가 실행됩니다.

· 카운트 업 플래그가 강제 리셋되었을 때는 강제 리셋이 해제될 때까지 블록 프로그램이 이 명령만 실행합니다.

· 카운터 번호는 카운터 명령, 가역 카운터 명령, 블록 프로그램의 카운터 대기 명령과 공용입니다. 이들 명령에서 같은 카운터 번호를 동시에 동작시키면 오작동하므로 주의하십시오. 동시에 사용하면 프로그램 확인 시 '코일 이중 사용'이 표시됩니다. 또한 동시에 동작시키지 않는 조건이면 동일한 번호를 사용할 수 있습니다.

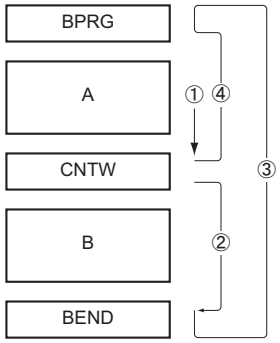
샘플 프로그램



코딩

주소	명령	데이터
000200	LD	0.00
000201	BPRG	0
⋮	A	⋮
000210	CNTW	5
		#7000
		1.00
⋮	B	⋮
000220	BEND	—

0.00이 ON이면 BPRG0이 실행됩니다. 프로그램 A가 실행된 후 카운터 5에 설정값이 사전 설정되어 BEND로 점프합니다.
다음 사이클에서는 A가 실행되지 않고 CNTW 명령부터 실행됩니다. 그리고 카운트 업(7000회)된 시점에서 프로그램 B가 실행됩니다.
카운트 업될 때까지 프로그램은 오른쪽 그림과 같이 ②→③→④→②를 반복합니다.



TMHW/TMHWX

명령 이름	니모닉	동작 옵션	펑션 번호	기능
고속 타이머 대기	TMHW	–	815	지정된 타이머가 타임업될 때까지 다음 명령부터 BEND 명령까지의 프로그램을 실행하지 않습니다.
	TMHWX	–	817	

심볼	BCD 방식일 때	BIN 방식일 때
	TMHW N S N: 카운터 번호 S: 카운터 설정값	TMHWX N S N: 카운터 번호 S: 카운터 설정값

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	가능	가능	가능*1	불가능	불가능

*1: 서브루틴 프로그램 영역 내의 블록 프로그램 영역에서 사용 가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	타이머 번호	TIMER	1
S	타이머 설정값	WORD	1

N: 타이머 번호
10진수 0~4095
S: 타이머 설정값
BCD: #0000~9999
BIN: 10진수 &0~65535 또는 16진수 #0000~FFFF

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
N	–	–	–	–	○	–	–	–	–	–	–	–	–	○	–	–	–	–
S	○	○	○	○		○	○	○	○	○	○	○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 블록 프로그램 영역에 없을 때 ON · N의 타이머 번호가 간접 IR 지정인 경우, 주소가 타이머의 현재값 주소가 아닐 때 ON · 타이머 현재값 갱신 방식이 BCD 방식이고 S의 데이터가 BCD가 아닐 때 ON · 그 이외일 때 OFF

TMHW/TMHWX

기능

감산식 ON 딜레이 10ms 타이머입니다.

설정 시간은 다음과 같습니다.

· BCD 방식의 경우: 0~99.99초

· BIN 방식의 경우: 0~655.35초

타이머 정도는 −0.01~0초입니다.

주: CS1D CPU 유니트에서는 타이머 정도가 ±(10ms + 사이클 시간)입니다.

블록 프로그램 기동 후 1회째 이 명령을 실행하는 경우

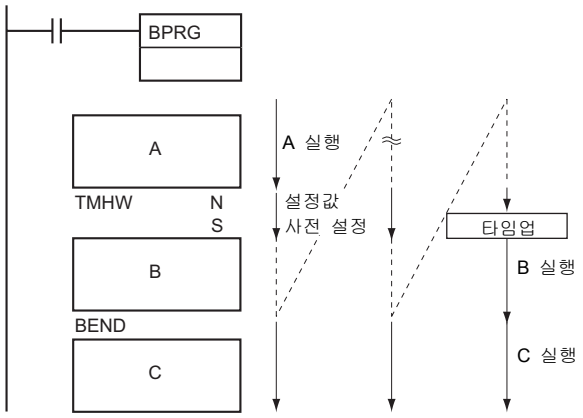
(1) 타임업 플래그를 리셋합니다.

(2) 타이머 설정값을 타이머 현재값 영역에 사전 설정합니다. (N으로 지정한 타이머 번호의 타이머가 기동 중이면 그대로 타이머 현재값을 갱신합니다.)

(3) 타이머 현재값을 갱신합니다.

타임업될 때까지 블록 프로그램은 이 명령만 실행합니다.

타임업 후에는 타임업 플래그가 ON이 되고 다음 명령 이후가 실행됩니다.



힌트

· 이 명령은 WAIT 명령의 입력 조건이 타이머(타임업)가 되는 명령입니다. 따라서 타이머에 의한 공정 진행 등에 사용할 수 있습니다.

사용상의 주의 사항

· 타이머 번호가 0~2047인 경우 고속 타이머 대기 명령은 태스크가 대기 중이어도 현재값을 갱신합니다. 타이머 번호가 2048~4095이면 태스크가 대기 중인 경우 현재값을 유지합니다.

· 타임업 플래그가 강제 세트되었을 때는 다음 명령 이후가 실행됩니다.

· 타임업 플래그가 강제 리셋되었을 때는 강제 리셋이 해제될 때까지 블록 프로그램이 이 명령만 실행합니다.

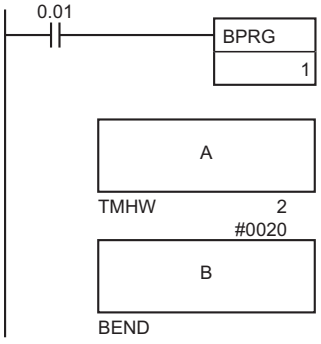
· 타이머 번호는 다른 타이머 명령과 공용입니다. 이들 명령에서 같은 타이머 번호를 동시에 동작시키면 오작동하므로 주의하십시오. 동시에 사용하면 프로그램 확인 시 '코일 이중 사용'이 표시됩니다. 또한 동시에 동작시키지 않는 조건이면 동일한 번호를 사용할 수 있습니다.

· 사이클 시간이 100ms 이상이면 정상적으로 동작하지 않으므로 주의하십시오.

샘플 프로그램

· TMHW 명령의 동작

0.01이 ON이면 BPRG1이 실행됩니다.
프로그램 A가 실행된 후 타이머 2에 설정값이 사전 설정되어 BEND로 점프합니다. 다음 사이클에서는 A가 실행되지 않고 TMHW 명령부터 실행됩니다. 타임업(0.2초)된 시점에서 프로그램 B가 실행됩니다.



코딩

주소	명령	데이터
000221	LD	0.01
000222	BPRG	1
⋮	A	⋮
000250	TMHW	2
		#0020
⋮	B	⋮
000281	BEND	—

LOOP/LEND/LEND NOT

명령 이름	니모닉	동작 옵션	펑션 번호	기능
반복 블록	LOOP	–	809	반복 블록 영역(LOOP~LEND)의 선두를 나타냅니다.
반복 블록 종료	LEND	–	810	반복 블록 영역(LOOP~LEND)의 종료를 나타냅니다. 입력 조건(피연산자를 지정하지 않는 경우) 또는 지정된 접점(피연산자를 지정하는 경우)이 ON이 될 때까지 LOOP~LEND를 반복합니다.
반복 블록 종료 (부정)	LEND NOT	–	810	반복 블록 영역(LOOP~LEND NOT)의 종료를 나타냅니다. 지정된 접점이 OFF가 될 때까지 LOOP~LEND NOT을 반복합니다.

	LOOP	LEND(LEND NOT)
심볼	LOOP	피연산자를 지정하지 않는 경우(입력 조건 사용) 입력 조건 LEND 피연산자를 지정하는 경우 LEND 릴레이 번호 R LEND NOT 릴레이 번호 R

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	불가능	가능	가능	가능*1	가능*1	불가능

*1: 서브루틴 프로그램 영역, 인터럽트 태스크 프로그램 영역 내의 블록 프로그램 영역에서 사용 가능

피연산자 설명

피연산자	내용	데이터 형식	크기
R	릴레이 번호	BOOL	1

■ 영역 종류

LEND/LEND NOT 명령

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
R	○	○	○	○	○	○	○*1	○*1	–	–	–	–	–	○	○	○	○	–

*1: CJ2-CPU 유니트만 사용 가능

조건 플래그의 동작

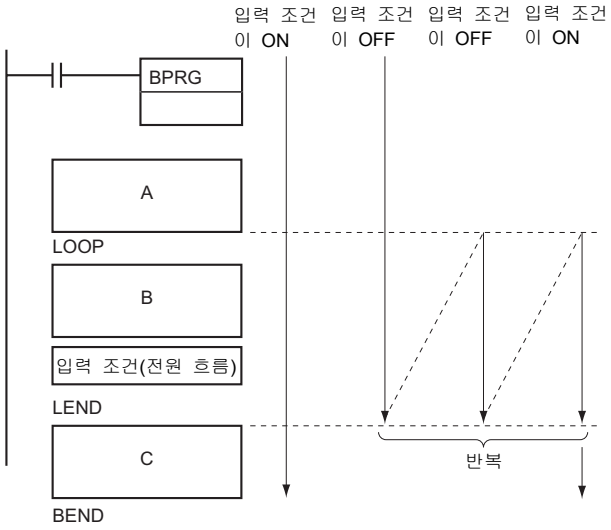
이름	라벨	내용
에러 플래그	P_ER	· 블록 프로그램 영역에 없을 때 ON · 그 이외일 때 OFF

LOOP/LEND/LEND NOT

기능

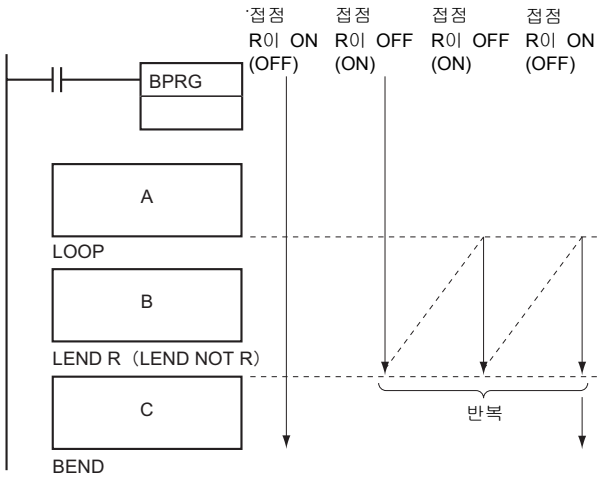
■ LEND 명령의 피연산자를 지정하지 않는 경우

LEND 명령 앞에 입력 조건을 설정합니다. 입력 조건은 반드시 LD 명령부터 시작합니다.
LEND 명령의 입력 조건이 OFF이면 LOOP 명령의 다음 명령으로 되돌아가 LOOP~LEND 사이를 반복합니다. ON이면 LOOP 명령의 다음 명령으로 돌아가지 않고 반복 블록을 종료한 후 다음 명령 이후를 실행합니다.



■ LEND 명령의 피연산자를 지정하는 경우

LEND 명령의 피연산자로 접점 R를 지정합니다. R로 지정한 접점이 OFF(ON)이면 LOOP 명령의 다음 명령으로 되돌아가 LOOP~LEND 사이를 반복합니다. ON(OFF)이면 LOOP 명령의 다음 명령으로 돌아가지 않고 반복 블록을 종료한 후 다음 명령 이후를 실행합니다.

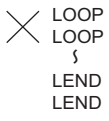


주: 괄호 안은 LEND NOT 명령인 경우입니다.

사용상의 주의 사항

LOOP~LEND 사이가 반복되는 동안은 I/O 리프레시되지 않으므로 필요에 따라 IORF(I/O 리프레시) 명령을 사용하십시오. 또한 사이클 시간을 초과하지 않도록 주의하십시오.

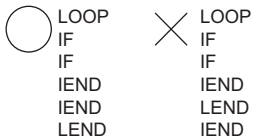
· 반복 블록을 중첩시킬 수 없습니다.



· LOOP와 LEND를 반대로 할 수 없습니다.



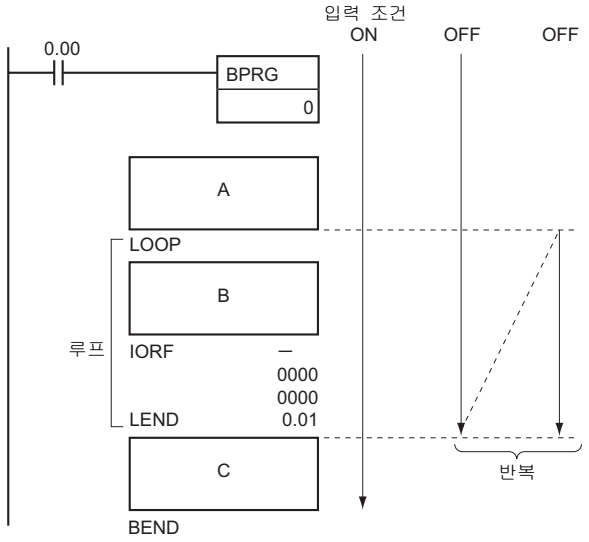
· LOOP~LEND 사이에서 조건 분기 블록을 중첩하는 경우, LOOP~LEND 사이에서 중첩을 완결시키십시오.



· LOOP 명령이 실행되지 않으면 LEND 명령은 NOP 처리가 됩니다.

샘플 프로그램

· LOOP 명령의 동작
0.00이 ON이면 BPRG0이 실행됩니다.
프로그램 A가 실행된 후 LOOP 명령이 이동하여 B가 실행됩니다.
I/O 리프레시 명령이 실행된 후 접점 0.01이 OFF이면 LOOP 명령으로 돌아가 프로그램 B가 반복 실행됩니다.
접점 0.01이 ON이면 프로그램 C가 실행되어 블록 프로그램이 종료됩니다.



코딩

주소	명령	데이터
000200	LD	0.00
000201	BPRG	0
⋮	A	⋮
000210	LOOP	—
⋮	B	⋮
000220	IORF	—
		0000
		0000
000221	LEND	0.01
⋮	C	⋮
000230	BEND	—

문자열 처리 명령

니모닉	명령 이름	펑션 번호	페이지
MOV$	문자열 전송	664	3-934
+$	문자열 연결	656	3-936
LEFT$	왼쪽에서 문자열 가져오기	652	3-938
RGHT$	오른쪽에서 문자열 가져오기	653	
MID$	임의 위치에서 문자열 가져오기	654	3-940
FIND$	문자열 검색	660	3-942
LEN$	문자열 길이 검출	650	3-944
RPLC$	문자열 치환	661	3-946
DEL$	문자열 삭제	658	3-948
XCHG$	문자열 교환	665	3-950
CLR$	문자열 클리어	666	3-952
INS$	문자열 삽입	657	3-954
=$, <>$, <$, <=$, >$, >=$	문자열 비교	670~675	3-956

문자열 처리

선두부터 NUL 코드(#00)까지의 데이터를 ASCII 코드(1바이트, 특수 문자 제외)로 표시한 문자열 데이터로 처리합니다.

저장 순서는 상위 바이트→하위 바이트 순서로, 하위 채널→상위 채널 순서입니다.

(예) 문자열 ABCDE의 경우

A	B	=	41	42
C	D		43	44
E	NUL		45	00

문자 수가 홀수인 경우 최종 채널의 하위 바이트의 빈 공간에 #00(NUL 코드)이 저장됩니다.

(예) 문자열 ABCD의 경우

A	B	=	41	42
C	D		43	44
NUL	NUL		00	00

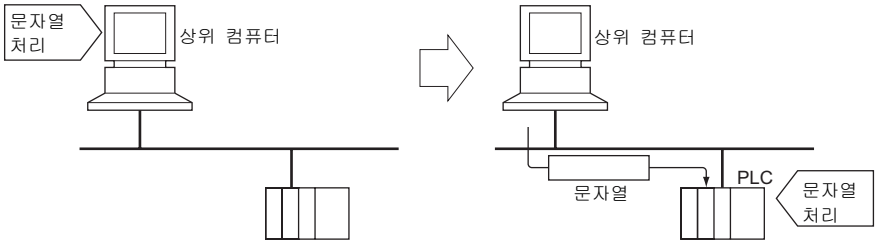
문자 수가 짝수인 경우에는 최종 채널+1의 상위/하위 바이트에 #0000(NUL 코드 2개)이 저장됩니다.

(예) MOV$ D0 D100

D0	41	42	→	D100	41	42
D1	43	44		D101	43	44
D2	45	NUL		D102	45	NUL

위와 같이 문자열을 지정하면 시작 CH만 지정해도 NUL(#00)이 있는 위치까지의 데이터를 한 덩어리의 문자열(ASCII 코드) 데이터로 처리할 수 있습니다.

문자열 처리 명령은 기존 상위 컴퓨터에서 수행하던 각종 문자열 처리(생산 품종 데이터 등의 처리)를 PLC 측에서 수행하는 경우 등에 사용할 수 있습니다.



예를 들어, 상위 컴퓨터에서 품종명별 생산 계획 데이터 등을 PLC에 전송하고, PLC 측에서 품종명 등의 문자열에 대해 삽입, 치환 등의 각종 처리를 수행할 수 있습니다.

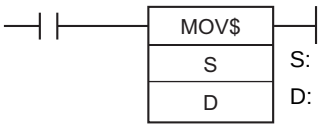
이를 통해 상위 컴퓨터의 데이터 처리 부담을 줄일 수 있습니다.

문자열 처리 명령에서 사용하는 ASCII 코드는 다음과 같습니다.
특수 문자를 제외하고 영문자, 숫자, 가타카나, 기호입니다.

하위 4비트 \ 상위 4비트	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0			SP	0	@	P	‘	p				ー	タ	ミ		
1			!	1	A	Q	a	q			。	ア	チ	ム		
2			”	2	B	R	b	r			「	イ	ツ	メ		
3			#	3	C	S	c	s			」	ウ	テ	モ		
4			$	4	D	T	d	t			、	エ	ト	ヤ		
5			%	5	E	U	e	u			・	オ	ナ	ユ		
6			&	6	F	V	f	v			ヲ	カ	ニ	ヨ		
7			’	7	G	W	g	w			ァ	キ	ヌ	ラ		
8			(	8	H	X	h	x			ィ	ク	ネ	リ		
9			)	9	I	Y	i	y			ゥ	ケ	ノ	ル		
A			*	:	J	Z	j	z			ェ	コ	ハ	レ		
B			+	;	K	[	k	{			ォ	サ	ヒ	ロ		
C			,	<	L	¥	l	|			ャ	シ	フ	ワ		
D			−	=	M	]	m	}			ュ	ス	ヘ	ン		
E			.	>	N	^	n	~			ョ	セ	ホ	゛		
F			/	?	O	_	o				ッ	ソ	マ	°		

MOV$

명령 이름	니모닉	동작 옵션	평션 번호	기능
문자열 전송	MOV$	@MOV$	664	문자열을 전송합니다.

심볼	MOV$
	 S: 전송 문자열 데이터 하위 CH 번호 D: 전송 대상 하위 CH 번호

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	전송 문자열 데이터 하위 CH 번호	UINT	가변
D	전송 대상 하위 CH 번호	UINT	가변

■ 영역 종류

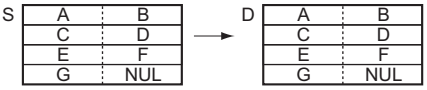
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S의 문자열 데이터가 4095문자를 초과할 때 ON · 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 전송 문자열이 '문자 없음(NUL만 있음)'일 때 ON · 그 이외일 때 OFF

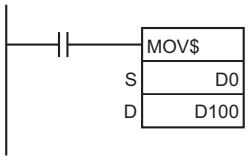
기능

S로 지정한 문자열 데이터를 그대로 문자열 데이터로(끝 부분의 NUL도 포함) D에 전송합니다. S의 문자열의 최대 문자 수는 4095문자입니다.



이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

샘플 프로그램



예 문자열: ABCDEF

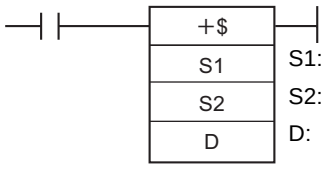
	15 8	7 0
S : D0	41	42
D1	43	44
D2	45	46
D3	00	00

→

	15 8	7 0
D : D100	41	42
D101	43	44
D102	45	46
D103	00	00

+$

명령 이름	니모닉	동작 옵션	평션 번호	기능
문자열 연결	+$	@+$	656	문자열과 문자열을 연결합니다.

심볼	+$
	 S1: 문자열 데이터 하위 CH 번호 1 S2: 문자열 데이터 하위 CH 번호 2 D: 연결 문자열 출력 하위 CH 번호

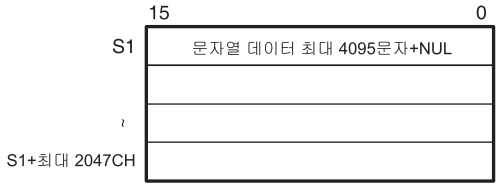
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

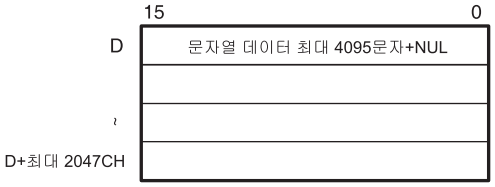
피연산자 설명

피연산자	내용	데이터 형식	크기
S1	문자열 데이터 하위 CH 번호 1	INT	가변
S2	문자열 데이터 하위 CH 번호 2	INT	가변
D	연결 문자열 출력 하위 CH 번호	INT	가변

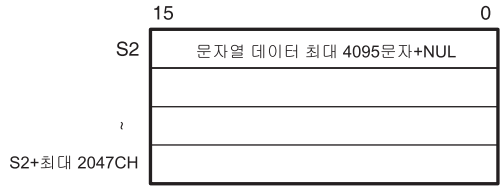
S1: 문자열 데이터 하위 CH 번호 1



D: 연결 문자열 출력 하위 CH 번호



S2: 문자열 데이터 하위 CH 번호 2



주:
・S1~S1+최대 2047CH, S2~S2+최대 2047CH 및 D~D+최대 2047CH는 각각 동일 영역 종류이어야 합니다.
・S2~S2+최대 2047CH와 D~D+최대 2047CH는 중복할 수 없습니다.

■ 영역 종류

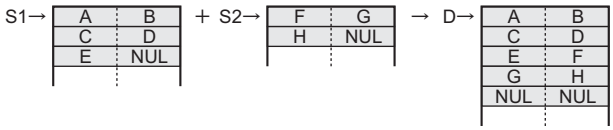
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S1	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S2																		
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S1, S2의 문자열 데이터가 4095문자를 초과할 때 ON · 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연결 결과가 '문자 없음(NUL만 있음)'일 때 ON · 그 이외일 때 OFF

기능

S1로 지정한 문자열과 S2로 지정한 문자열을 연결하여 결과를 문자열 데이터로(끝에 NUL 부가) D에 출력합니다.



S1과 S2의 문자열의 최대 문자 수는 4095문자입니다. 이를 초과하는 경우(4096문자까지 NUL이 존재하지 않는 경우)는 에러가 되어 에러 플래그가 ON이 됩니다. 또한 연결 결과의 최대 문자 수도 4095문자입니다. 이를 초과하는 경우 4095번째 문자까지(4096번째 문자에 NUL 부가) D에 출력합니다.

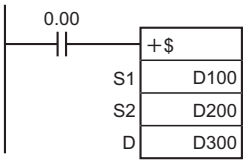
S1과 S2가 모두 NUL인 경우, NUL의 2문자 분량(#0000)을 D에 출력합니다.

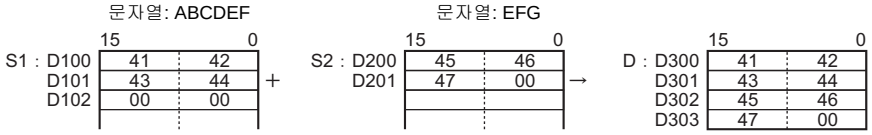
이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

사용상의 주의 사항

· D로 지정한 출력 대상 CH가 S2의 문자열 데이터 영역과 중복되지 않도록 하십시오. 중복되는 경우는 정상적으로 동작하지 않습니다.

샘플 프로그램





LEFT$/RGHT$

명령 이름	니모닉	동작 옵션	펑션 번호	기능
왼쪽에서 문자열 가져오기	LEFT$	@LEFT$	652	문자열의 왼쪽(선두)부터 지정된 문자 수만큼 가져옵니다.
오른쪽에서 문자열 가져오기	RGHT$	@RGHT$	653	문자열의 오른쪽(끝)부터 지정된 문자 수만큼 가져옵니다.

심볼	LEFT$	RGHT$
		

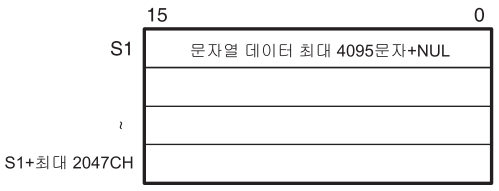
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

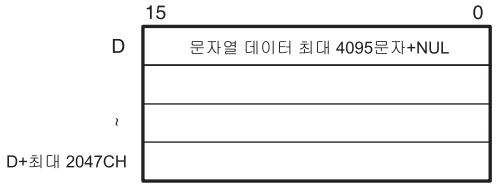
피연산자 설명

피연산자	내용	데이터 형식	크기
S1	문자열 데이터 하위 CH 번호	UINT	가변
S2	가져올 문자 수 데이터	UINT	1
D	문자열 출력 하위 CH 번호	UINT	가변

S1: 문자열 데이터 하위 CH 번호



D: 문자열 출력 하위 CH 번호



S2: 가져올 문자 수 데이터

10진수 &0~4095 또는 16진수 #0000~0FFF

주:

・S1~S1+최대 2047CH 및 D~D+최대 2047CH는 각각 동일 영역 종류이어야 합니다.

・S1~S1+최대 2047CH와 D~D+최대 2047CH는 중복할 수 있습니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S1	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S2											○	○						
D											–	–						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S1의 문자열 데이터가 4095문자를 초과할 때 ON · S2의 문자 수 데이터가 4095문자를 초과할 때 ON · 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 가져올 문자열이 '문자 없음(NUL만 있음)'일 때 ON · 그 이외일 때 OFF

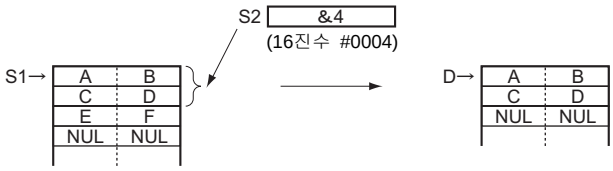
기능

■ LEFT$

S1로 지정한 문자열 데이터 하위 CH 번호부터 NUL(#00) 코드까지의 문자열의 왼쪽(선두)부터 S2로 지정한 문자 수만큼 읽어 결과를 문자열 데이터로 (끝에 NUL 부가) D에 출력합니다.

가져올 문자 수가 S1의 문자 수를 초과한 경우는 S1의 문자열 전체를 출력합니다.

가져올 문자 수에 0을 지정한 경우는 D에 NUL의 2문자 분량(#0000)을 출력합니다.



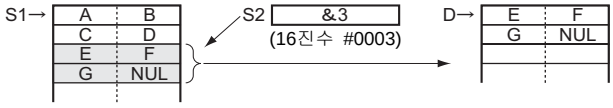
이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

■ RGHT$

S1로 지정한 문자열 데이터 하위 CH 번호부터 NUL(#00) 코드까지의 문자열의 오른쪽(끝)부터 S2로 지정한 문자 수만큼 읽어 결과를 문자열 데이터로 (끝에 NUL 부가) D에 출력합니다.

가져올 문자 수가 S1의 문자 수를 초과한 경우는 S1의 문자열 전체를 출력합니다.

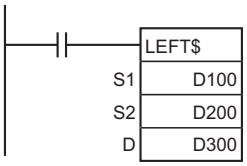
가져올 문자 수에 0을 지정한 경우는 D에 NUL의 2문자 분량(#0000)을 출력합니다.

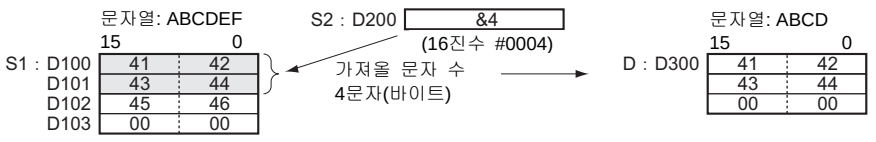


이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

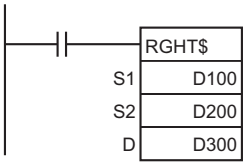
샘플 프로그램

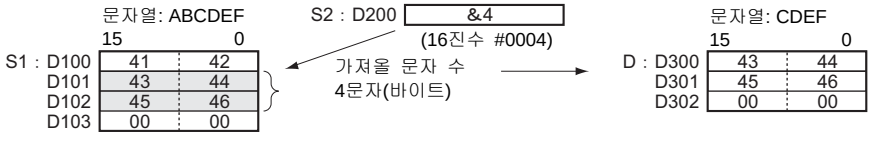
■ LEFT$





■ RGHT$

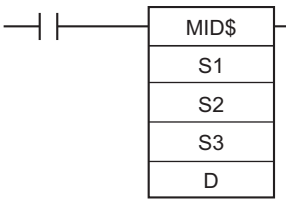




MID$

명령 이름	니모닉	동작 옵션	펑션 번호	기능
임의 위치에서 문자열 가져오기	MID$	@MID$	654	문자열의 임의의 위치부터 지정된 수만큼 가져옵니다.

심볼	MID$



S1: 문자열 데이터 하위 CH 번호
S2: 가져올 문자 수 데이터
S3: 가져올 시작 위치 데이터
D: 문자열 출력 하위 CH 번호

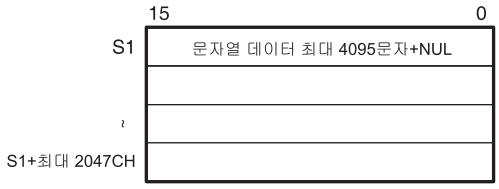
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

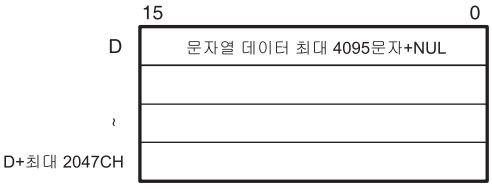
피연산자 설명

피연산자	내용	데이터 형식	크기
S1	문자열 데이터 하위 CH 번호	UINT	가변
S2	가져올 문자 수 데이터	UINT	1
S3	가져올 시작 위치 데이터	UINT	1
D	문자열 출력 하위 CH 번호	UINT	가변

S1: 문자열 데이터 하위 CH 번호



D: 문자열 출력 하위 CH 번호



S2: 가져올 문자 수 데이터
10진수 &0~4095 또는 16진수 #0000~0FFF

S3: 가져올 시작 위치 데이터
10진수 &1~4095 또는 16진수 #0001~0FFF

주:
- S1~S1+최대 2047CH 및 D~D+최대 2047CH는 각각 동일 영역 종류이어야 합니다.
- S1~S1+최대 2047CH와 D~D+최대 2047CH는 중복할 수 있습니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S1	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S2, S3											○	○						
D											–	–						

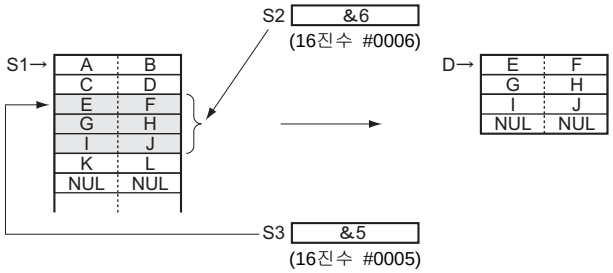
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S1의 문자열 데이터가 4095문자를 초과할 때 ON · S2의 데이터가 0~4095문자를 초과할 때 ON · S3의 데이터가 1~4095문자의 범위가 아닐 때 ON · S3의 데이터가 S1의 문자열 데이터의 문자 수를 초과했을 때 ON · 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 가져올 문자열이 '문자 없음(NUL만 있음)'일 때 ON · 그 이외일 때 OFF

기능

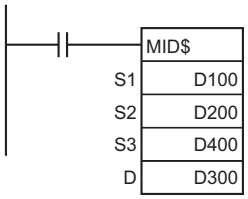
S1로 지정한 문자열 데이터 하위 CH 번호부터 NUL(#00) 코드까지의 문자열의 S3으로 지정한 시작 위치부터 S2로 지정한 문자 수만큼 읽어 결과를 문자열 데이터로(끝에 NUL 부가) D에 출력합니다. 가져올 문자 수가 S1의 문자열의 끝을 초과한 경우는 끝에 있는 문자열까지 출력합니다.

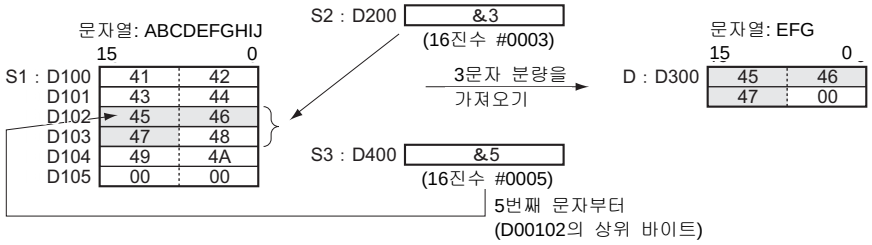
가져올 문자 수에 0을 지정한 경우는 D에 NUL의 2문자 분량(#0000)을 출력합니다.



이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

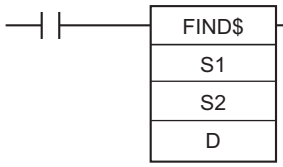
샘플 프로그램





FIND$

명령 이름	니모닉	동작 옵션	평션 번호	기능
문자열 검색	FIND$	@FIND$	660	문자열에서 지정된 문자열을 검색하고 그 위치를 출력합니다.

심볼	FIND$
	 S1: 피검색 문자열 데이터 하위 CH 번호 S2: 검색 문자열 데이터 하위 CH 번호 D: 검색 결과 출력 CH 번호

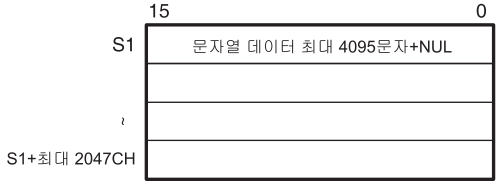
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

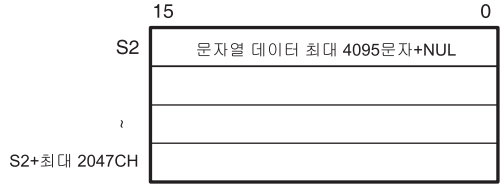
피연산자 설명

피연산자	내용	데이터 형식	크기
S1	피검색 문자열 데이터 하위 CH 번호	UINT	가변
S2	검색 문자열 데이터 하위 CH 번호	UINT	가변
D	검색 결과 출력 CH 번호	UINT	1

S1: 피검색 문자열 데이터 하위 CH 번호



S2: 검색 문자열 데이터 하위 CH 번호



주: S1~S1+최대 2047CH 및 S2~S2+최대 2047CH는 각각 동일 영역 종류이어야 합니다.

■ 영역 종류

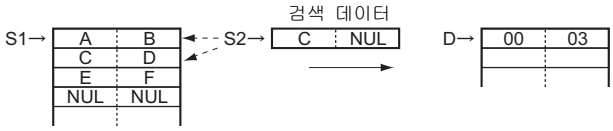
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S1	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S2																		
D																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S1, S2의 문자열 데이터가 4095문자를 초과할 때 ON · 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 검색 결과가 '문자 없음(NUL만 있음)'일 때 ON · 그 이외일 때 OFF

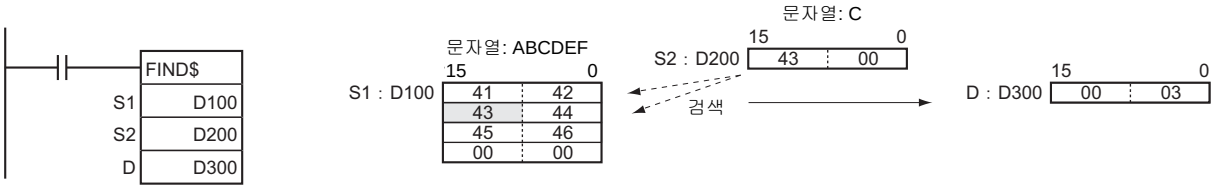
기능

S1로 지정한 문자열에서 S2로 지정한 문자열을 검색하여 결과(S1의 선두부터 몇 번째 문자인지)를 BIN 데이터로 D에 출력합니다. 일치하는 문자열이 없는 경우는 #0000을 D에 출력합니다.



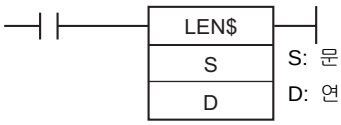
이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

샘플 프로그램



LEN$

명령 이름	니모닉	동작 옵션	평션 번호	기능
문자열 길이 검출	LEN$	@LEN$	650	문자열의 길이를 계산합니다.

심볼	LEN$
	 S: 문자열 데이터 하위 CH 번호 D: 연산 결과 출력 CH 번호

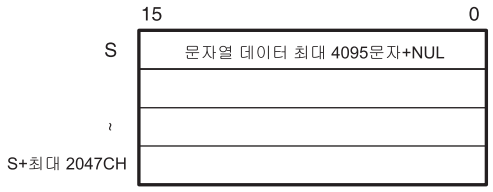
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	문자열 데이터 하위 CH 번호	UINT	가변
D	연산 결과 출력 CH 번호	UINT	1

S: 문자열 데이터 하위 CH 번호



주: S~S+최대 2047CH는 각각 동일 영역 종류이어야 합니다.

■ 영역 종류

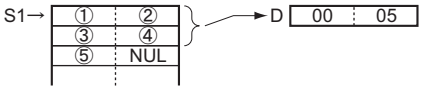
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
D												○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 연산 결과가 4095문자를 초과할 때 ON · 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 연산 결과가 0일 때 ON · 그 이외일 때 OFF

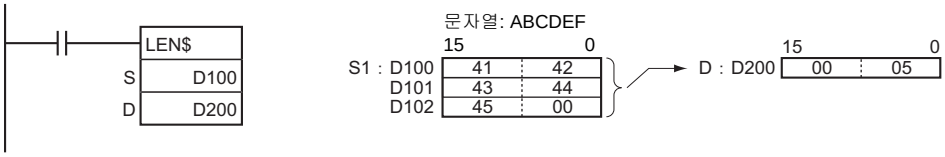
기능

S로 지정한 문자열 데이터 하위 CH 번호부터 NUL 코드(#00)까지의 문자 수(NUL 코드 자체는 포함 안 함)를 계산하여 결과를 BIN 데이터로 D에 출력합니다. 문자열 데이터의 선두가 NUL인 경우 연산 결과는 #0000이 됩니다.



이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

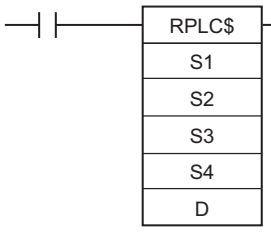
샘플 프로그램



RPLC$

명령 이름	니모닉	동작 옵션	펑션 번호	기능
문자열 치환	RPLC$	@RPLC$	661	지정된 위치의 문자열을 지정된 문자로 치환합니다.

심볼	RPLC$



S1: 문자열 데이터 하위 CH 번호
S2: 치환 문자열 데이터 하위 CH 번호
S3: 치환 문자 수 데이터
S4: 치환 시작 위치 데이터
D: 치환 결과 출력 하위 CH 번호

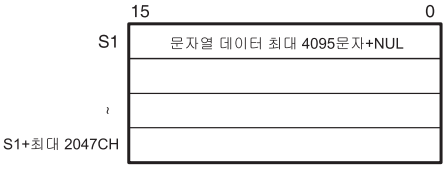
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

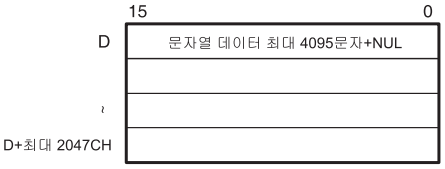
피연산자 설명

피연산자	내용	데이터 형식	크기
S1	문자열 데이터 하위 CH 번호	UINT	가변
S2	치환 문자열 데이터 하위 CH 번호	UINT	가변
S3	치환 문자 수 데이터	UINT	1
S4	치환 시작 위치 데이터	UINT	1
D	치환 결과 출력 하위 CH 번호	UINT	가변

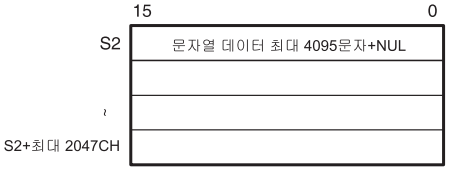
S1: 문자열 데이터 하위 CH 번호



D: 치환 결과 출력 하위 CH 번호



S2: 치환 문자열 데이터 하위 CH 번호



S3: 치환 문자 수 데이터
10진수 &0~4095 또는 16진수 #0000~0FFF

S4: 치환 시작 위치 데이터
10진수 &1~4095 또는 16진수 #0001~0FFF

주:
- S1~S1+최대 2047CH, S2~S2+최대 2047CH, D~D+최대 2047CH는 각각 동일 영역 종류이어야 합니다.
- D~D+최대 2047CH와 S2~S2+최대 2047CH는 중복할 수 없습니다.

■ 영역 종류

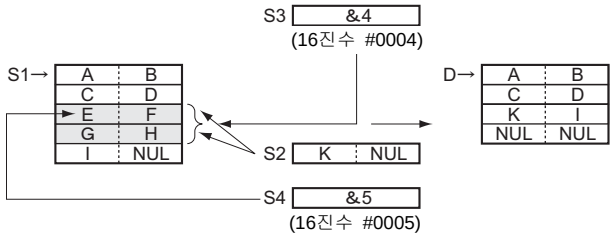
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S1, S2	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S3, S4											○	○						
D											–	–						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S1, S2의 문자열 데이터가 4095문자를 초과할 때 ON · S3의 데이터가 4095문자를 초과할 때 ON · S4의 데이터가 1~4095문자의 범위가 아닐 때 ON · 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 치환 결과가 '문자 없음(NUL만 있음)'일 때 ON · 그 이외일 때 OFF

기능

S1로 지정한 문자열의 S4로 지정한 시작 위치부터 S3으로 지정한 문자 수만큼 S2로 지정한 문자열로 치환하여 결과를 문자열 데이터로(끝에 NUL 부가) D에 출력합니다.



· 치환 결과의 최대 문자 수도 4095문자입니다. 이를 초과하는 경우 4095번째 문자까지(4096번째 문자에는 NUL 부가) D에 출력합니다.

· 치환 문자 수는 0~4095문자(16진수 #0000~0FFF)의 범위입니다. 0문자의 경우는 S1의 문자열을 그대로 D에 출력합니다. S2의 문자열이 NUL인 경우는 S1의 문자열의 지정 범위를 삭제하는 것과 동일한 동작이 됩니다.

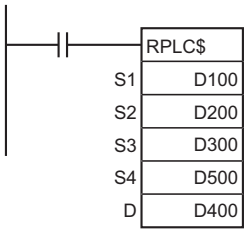
· S1의 문자열 선두부터 끝까지 NUL로 치환하는 경우, NUL의 2문자 분량(#0000)을 D에 출력합니다.

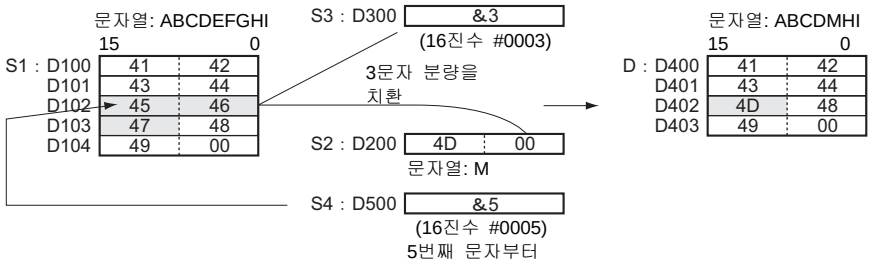
이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

사용상의 주의 사항

· D로 지정한 출력 대상 CH가 S2의 문자열 데이터 영역과 중복되지 않도록 하십시오. 중복되는 경우 정상적으로 동작하지 않습니다.

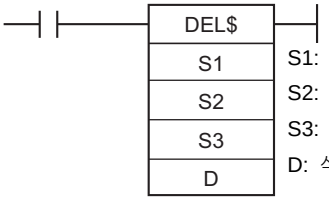
샘플 프로그램





DEL$

명령 이름	니모닉	동작 옵션	평션 번호	기능
문자열 삭제	DEL$	@DEL$	658	문자열에서 지정된 문자열을 삭제합니다.

	DEL$
심볼	 S1: 문자열 데이터 하위 CH 번호 S2: 문자 수 데이터 S3: 삭제 시작 위치 데이터 D: 삭제 결과 출력 하위 CH 번호

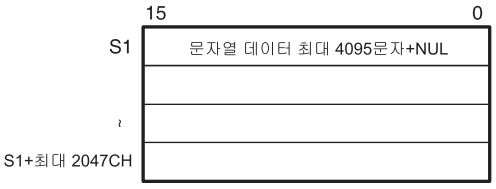
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

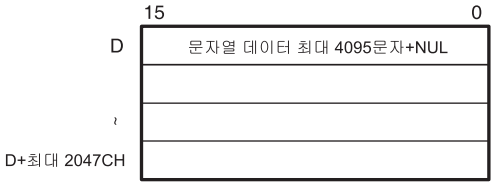
피연산자 설명

피연산자	내용	데이터 형식	크기
S1	문자열 데이터 하위 CH 번호	UINT	가변
S2	문자 수 데이터	UINT	1
S3	삭제 시작 위치 데이터	UINT	1
D	삭제 결과 출력 하위 CH 번호	UINT	가변

S1: 문자열 데이터 하위 CH 번호



D: 삭제 결과 출력 하위 CH 번호



S2: 문자 수 데이터

10진수 &0~4095 또는 16진수 #0000~0FFF

S3: 삭제 시작 위치 데이터

10진수 &1~4095 또는 16진수 #0001~0FFF

주:

- S1~S1+최대 2047CH, S2~S2+최대 2047CH, D~D+최대 2047CH는 각각 동일 영역 종류이어야 합니다.
- D~D+최대 2047CH와 S1~S1+최대 2047CH는 중복할 수 있습니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S1, S2	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S2, S3											○	○						
D											–	–						

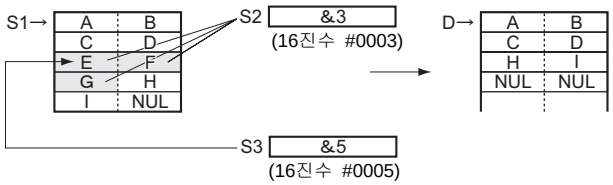
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S1의 문자열 데이터가 4095문자를 초과할 때 ON · S2의 데이터가 4095문자를 초과할 때 ON · S3의 데이터가 1~4095문자의 범위가 아닐 때 ON · S3의 데이터가 S1의 문자 수를 초과할 때 ON · 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 삭제 결과가 '문자 없음(NUL만 있음)'일 때 ON · 그 이외일 때 OFF

기능

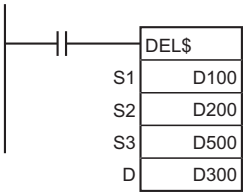
S1로 지정한 문자열의 S3으로 지정한 삭제 시작 위치부터 S2로 지정한 문자 수만큼 삭제하여 결과를 문자열 데이터로(끝에 NUL 부가) D에 출력합니다.

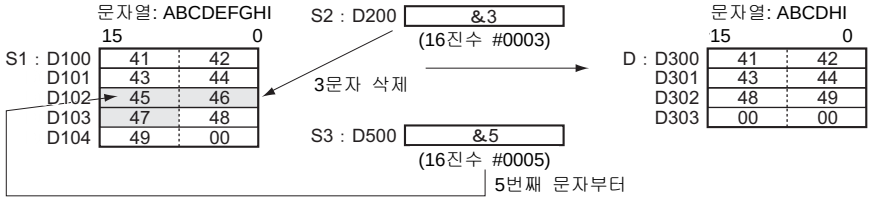
삭제 문자 수가 S1의 문자열 끝을 넘어간 경우 끝까지 삭제합니다. S1의 선두부터 끝까지 삭제하도록 지정하는 경우 #0000을 D에 출력합니다.



이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

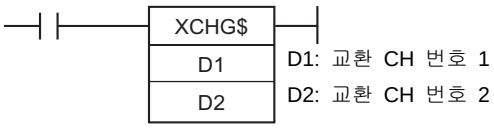
샘플 프로그램





XCHG$

명령 이름	니모닉	동작 옵션	평션 번호	기능
문자열 교환	XCHG$	@XCHG$	665	문자열과 문자열을 교환합니다.

심볼	XCHG$
	

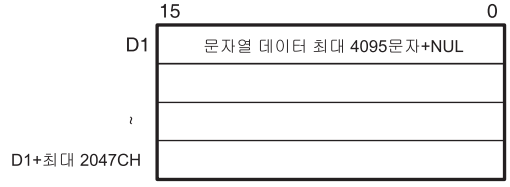
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

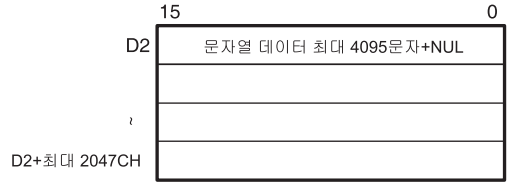
피연산자 설명

피연산자	내용	데이터 형식	크기
D1	교환 CH 번호 1	UINT	가변
D2	교환 CH 번호 2	UINT	가변

D1: 교환 CH 번호 1



D2: 교환 CH 번호 2



주:
- D1~D1+최대 2047CH, D2~D2+최대 2047CH는 각각 동일 영역 종류이어야 합니다.
- D1~D1+최대 2047CH와 D2~D2+최대 2047CH는 중복할 수 없습니다.

■ 영역 종류

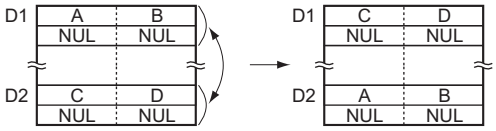
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
D1, D2	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· D1, D2의 문자열 데이터가 4095문자를 초과할 때 ON · D1, D2의 문자열 데이터 영역이 중복될 때 ON · 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF

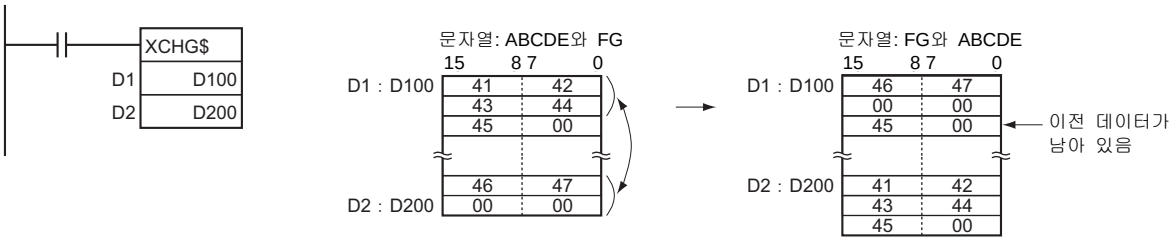
기능

D1로 지정한 문자열과 D2로 지정한 문자열을 교환합니다. D1과 D2 중 하나가 NUL인 경우 NUL2 문자 분량(#0000)을 나머지 한 쪽에 전송합니다.



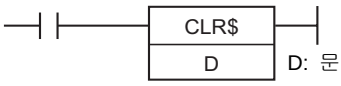
이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

샘플 프로그램



CLR$

명령 이름	니모닉	동작 옵션	평션 번호	기능
문자열 클리어	CLR$	@CLR$	666	문자열을 모두 NUL(00Hex)로 클리어합니다.

심볼	CLR$
	

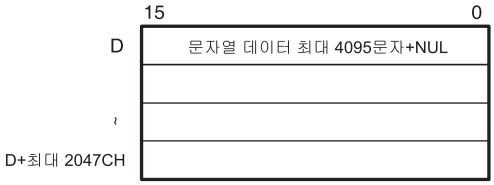
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
D	문자열 데이터 하위 CH 번호	UINT	가변

D: 문자열 데이터 하위 CH 번호



주: D~D+최대 2047CH는 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
D	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF

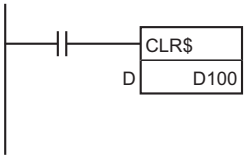
기능

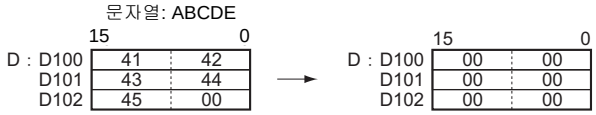
D로 지정한 문자열 데이터 하위 CH 번호부터 NUL 코드(#00)까지 모두 NUL(#00)로 클리어합니다. 클리어하는 최대 문자 수는 4096문자입니다. 4096번째 문자까지 NUL이 존재하지 않는 경우는 4096문자 분량을 클리어하고 그 이후는 클리어하지 않습니다.



이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

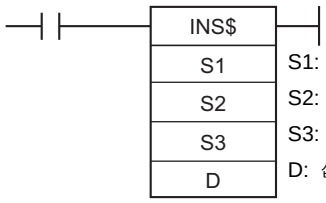
샘플 프로그램


CLR$
D
D100


문자열: ABCDE
15
0
D : D100
D101
D102
41
42
43
44
45
00
15
0
D : D100
D101
D102
00
00
00
00
00
00

INS$

명령 이름	니모닉	동작 옵션	펑션 번호	기능
문자열 삽입	INS$	@INS$	657	지정된 문자열 위치에 지정된 문자열을 삽입합니다.

	INS$
심볼	 S1: 피삽입 문자열 데이터 하위 CH 번호 S2: 삽입 문자열 데이터 하위 CH 번호 S3: 삽입 시작 위치 데이터 D: 삽입 결과 출력 하위 CH 번호

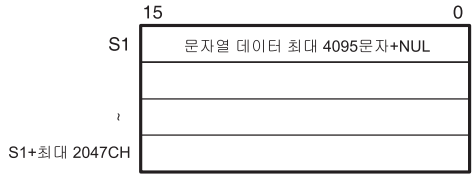
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

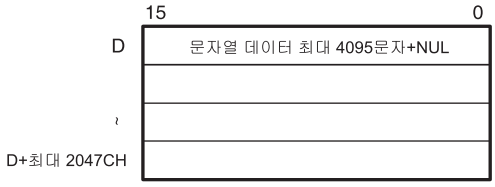
피연산자 설명

피연산자	내용	데이터 형식	크기
S1	피삽입 문자열 데이터 하위 CH 번호	UINT	가변
S2	삽입 문자열 데이터 하위 CH 번호	UINT	가변
S3	삽입 시작 위치 데이터 CH 번호	UINT	1
D	삽입 결과 출력 하위 CH 번호	UINT	가변

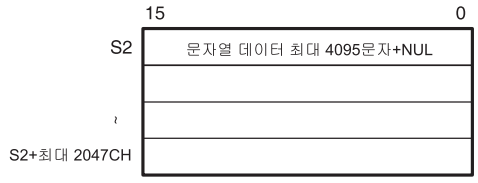
S1: 피삽입 문자열 데이터 하위 CH 번호



D: 삽입 결과 출력 하위 CH 번호



S2: 삽입 문자열 데이터 하위 CH 번호



S3: 삽입 시작 위치 데이터 CH 번호

10진수 &0~4095 또는 16진수 #0000~0FFF

주:

- S1~S1+최대 2047CH, S2~S2+최대 2047CH, D~D+최대 2047CH는 각각 동일 영역 종류이어야 합니다.
- D~D+최대 2047CH와 S2~S2+최대 2047CH는 중복할 수 없습니다. D~D+최대 2047CH와 S1~S1+최대 2047CH 및 S1~S1+최대 2047CH와 S2~S2+최대 2047CH는 중복할 수 있습니다.

■ 영역 종류

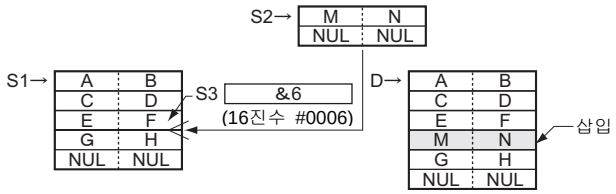
내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
S1, S2	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S3											○	○						
D											–	–						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S1, S2의 문자열 데이터가 4095문자를 초과할 때 ON · S3의 데이터가 4095문자를 초과할 때 ON · 백그라운드 처리를 지정한 경우에는 PLC 시스템 설정의 '백그라운드 처리에서 사용하는 통신 포트 번호'에서 지정한 통신 포트 번호에 대응하는 네트워크 통신 명령 실행 가능 플래그가 OFF일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· D에 #0000이 출력되었을 때 ON · 그 이외일 때 OFF

기능

S1로 지정한 문자열의 S3으로 지정한 시작 위치의 문자 뒤에 S2로 지정한 문자열을 삽입하여 결과를 문자열 데이터로(끝에 NUL 부가) D에 출력합니다.



· 삽입 결과의 최대 문자 수도 4095문자입니다. 이를 초과하는 경우 4095번째 문자까지(4096번째 문자에는 NUL 부가) D에 출력합니다.

· 삽입 시작 위치(S3)의 지정 범위는 0~4095번째 문자입니다. 0번째 문자를 지정한 경우는 S2와 S1의 연결과 동일한 동작이 됩니다.

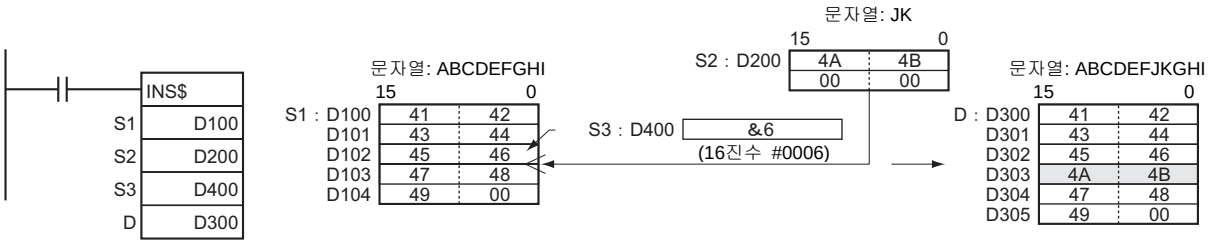
· S1과 S2 중 하나가 NUL인 경우, D에는 나머지 한 쪽의 문자열을 그대로 출력합니다. S1과 S2가 모두 NUL인 경우, NUL의 2문자 분량(#0000)을 D에 출력합니다.

이 명령은 백그라운드 처리를 지정할 수 있습니다. 자세한 내용은 'SYSMAC CS/CJ 시리즈 사용 설명서 프로그래밍편' 또는 'SYSMAC CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편'을 참조하십시오.

사용상의 주의 사항

· D로 지정한 출력 대상 CH가 S2의 문자열 데이터 영역과 중복되지 않도록 하십시오. 중복되는 경우는 정상적으로 동작하지 않습니다.

샘플 프로그램



=$, <>$, <$, <=$, >$, >=$

명령 이름	니모닉	동작 옵션	평션 번호	기능
문자열 비교	=$, <>$, <$, <=$, >$, >=$	–	670~ 675	2개의 문자열 데이터를 ASCII 코드의 크기로 선두 문자부터 비교(=, < >, <, <=, >, >= 중 하나)하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다.

심볼	LD(로드) 연결형	AND(직렬) 연결형	OR(병렬) 연결형
	니모닉 S1 S1: 문자열 데이터 하위 CH 번호 1 S2 S2: 문자열 데이터 하위 CH 번호 2	니모닉 S1 S1: 문자열 데이터 하위 CH 번호 1 S2 S2: 문자열 데이터 하위 CH 번호 2	니모닉 S1 S1: 문자열 데이터 하위 CH 번호 1 S2 S2: 문자열 데이터 하위 CH 번호 2

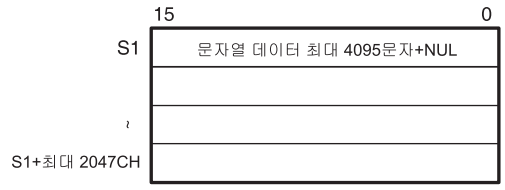
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

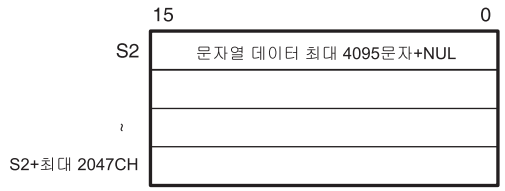
피연산자 설명

피연산자	내용	데이터 형식	크기
S1	문자열 데이터 하위 CH 번호 1	UINT	가변
S2	문자열 데이터 하위 CH 번호 2	UINT	가변

S1: 문자열 데이터 하위 CH 번호 1



S2: 문자열 데이터 하위 CH 번호 2



주:
- S1~S1+최대 2047CH, S2~S2+최대 2047CH는 각각 동일 영역 종류이어야 합니다.
- S1~S1+최대 2047CH와 S2~S2+최대 2047CH는 중복할 수 없습니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S1, S2	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S1, S2의 문자열 데이터가 4095문자를 초과하는 경우 ON · 그 이외일 때 OFF(에러일 때도 OFF)
> 플래그	P_GT	· 비교 결과가 S1>S2일 때 ON · 그 이외일 때 OFF(에러일 때도 OFF)
≧ 플래그	P_GE	· 비교 결과가 S1≧S2일 때 ON · 그 이외일 때 OFF(에러일 때도 OFF)
= 플래그	P_EQ	· 비교 결과가 S1=S2일 때 ON · 그 이외일 때 OFF(에러일 때도 OFF)
≠ 플래그	P_NE	· 비교 결과가 S1≠S2일 때 ON · 그 이외일 때 OFF(에러일 때도 OFF)
< 플래그	P_LT	· 비교 결과가 S1<S2일 때 ON · 그 이외일 때 OFF(에러일 때도 OFF)
≦ 플래그	P_LE	· 비교 결과가 S1≦S2일 때 ON · 그 이외일 때 OFF(에러일 때도 OFF)

주: 에러 플래그가 ON이면 비례 결과는 거짓으로 처리되어 다음 단계로 연결하지 않습니다.

기능

S1로 지정한 문자열과 S2로 지정한 문자열을 비교하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다.

S1과 S2의 문자열의 최대 문자 수는 각각 4095문자입니다.

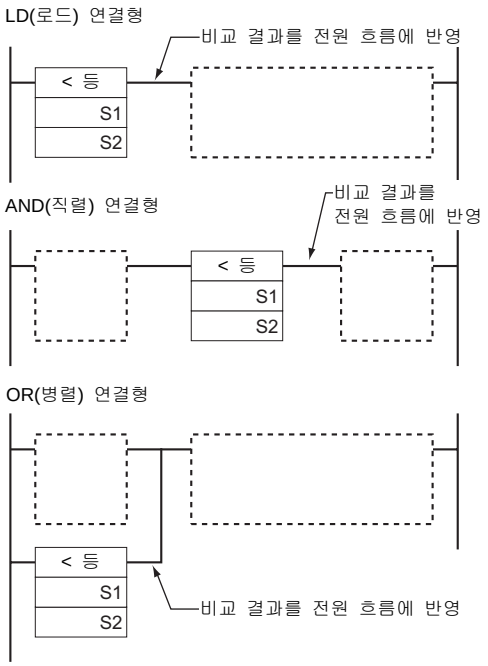
LD, AND, OR 명령과 동일하게 취급하며, 각 명령 후에는 계속하여 다른 명령을 프로그래밍합니다.

LD형의 경우: 모선에 직접 연결할 수 있습니다.

AND형의 경우: 모선에 직접 연결할 수 없습니다.

OR형의 경우: 모선에 직접 연결할 수 있습니다.

문자열 비교 명령은 18종류의 니모닉으로 표현됩니다.



=$, <>$, <$, <=$, >$, >=$

■ 문자열 비교 명령 목록

기능	니모닉	이름	펑션 번호
S1의 문자열=S2의 문자열일 때 참	LD =$	LD 문자열 일치	670
	AND =$	AND 문자열 일치	670
	OR =$	OR 문자열 일치	670
S1의 문자열≠S2의 문자열일 때 참	LD <>$	LD 문자열 불일치	671
	AND <>$	AND 문자열 불일치	671
	OR <>$	OR 문자열 불일치	671
S1의 문자열<$S2의 문자열일 때 참	LD <$	LD 문자열 미만	672
	AND <$	AND 문자열 미만	672
	OR <$	OR 문자열 미만	672
S1의 문자열≦S2의 문자열일 때 참	LD <=$	LD 문자열 이하	673
	AND <=$	AND 문자열 이하	673
	OR <=$	OR 문자열 이하	673
S1의 문자열＞S2의 문자열일 때 참	LD >$	LD 문자열 초과	674
	AND >$	AND 문자열 초과	674
	OR >$	OR 문자열 초과	674
S1의 문자열≧S2의 문자열일 때 참	LD >=$	LD 문자열 이상	675
	AND>=$	AND 문자열 이상	675
	OR >=$	OR 문자열 이상	675

비교 방법은 다음과 같습니다.

선두의 1문자(바이트)를 ASCII 코드로 서로 비교하여 ASCII 코드가 =이 아니라 어느 한 쪽이 크면 그 대소 관계가 해당 문자열 간의 대소 관계가 됩니다.

=이면 다음 문자를 비교하여 마찬가지로 =이 아니라 어느 한 쪽이 크면 그 대소 관계가 해당 문자열 간의 대소 관계가 됩니다. 이후 마찬가지로 문자를 순서대로 비교하고 NUL을 포함하여 모두 =이면 그 문자열들은 =의 관계가 됩니다.

또한 문자 수가 다른 경우, 긴 쪽 문자열 수에 맞춰 짧은 쪽 문자열을 NUL(#00)로 채운 후 비교합니다.

예: AD(#414400)와 BC(#424300)의 경우, 선두의 #41과 #42에서 AD＜BC

예: ADC(#41444300)와 B(#4200)의 경우, 선두의 #41과 #42에서 ADC＜B

예: ABC(#41424300)와 ABD(#41424400)의 경우, 선두의 #41은 일치, 다음의 #42도 일치, 다음의 #43과 #44부터 ABC＜ABD

예: ABC(#41424300)와 AB(#414200)의 경우, 선두의 #41은 일치, 다음의 #42도 일치, 다음의 #43과 #00부터 ABC＞AB

예: AB(#414200)와 AB(#414200)의 경우, 선두의 #41은 일치, 다음의 #42도 일치, 다음의 #00과 #00도 일치하므로 AB＝AB

LD, AND, OR 명령과 동일하게 취급하며, 각 명령 후에는 계속하여 다른 명령을 프로그래밍합니다.

LD형의 경우: 모선에 직접 연결할 수 있습니다.

AND형의 경우: 모선에 직접 연결할 수 없습니다.

OR형의 경우: 모선에 직접 연결할 수 있습니다.

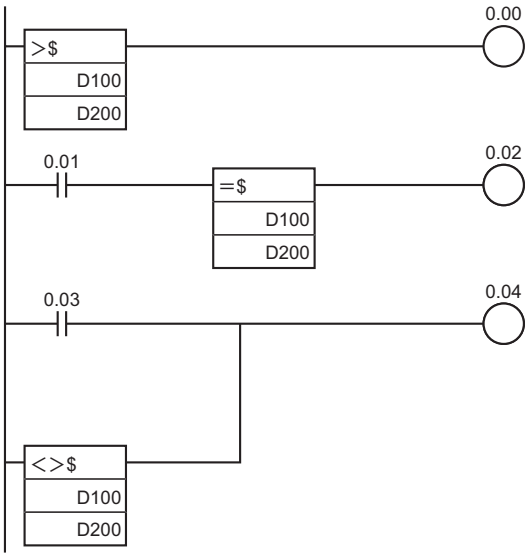
힌트

- 이 문자열 비교 명령은 문자열 데이터의 ASCII 코드 순서로 재배열하는 경우 등에 사용합니다. 예를 들어, 영문자의 경우 ASCII 코드의 소→대 순서가 A→Z까지의 알파벳 순서가 되므로 선두 문자부터 알파벳 순서로 재배열할 수 있습니다.

사용상의 주의 사항

- 이 명령의 최종 단계에는 출력계 명령(OUT계 명령 및 다음 단계 연결형 명령 이외의 응용 명령)을 추가하십시오.
- 회로의 최종 단계에는 이 명령을 사용할 수 없습니다.

샘플 프로그램

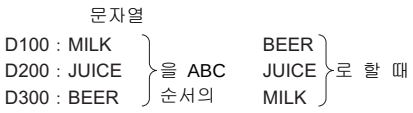


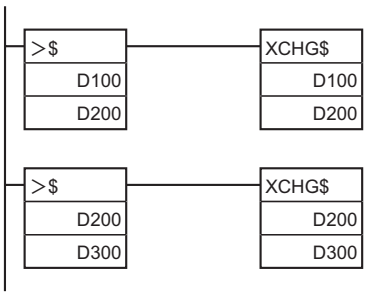
프로그램 주소	니모닉	데이터
000000	LD>$	– D100 D200
000001	OUT	0.00
000002	LD	0.01
000003	AND=$	– D100 D200
000004	OUT	0.02
000005	LD	0.03
000006	OR<>$	– D100 D200
000007	OUT	0.04

(예)

	>$ D100 D200	=$ D100 D200	<>$ D100 D200
문자열: ABDC (D100: 41 42 / D101: 44 43 / D102: 00 00) 문자열: ABC (D200: 41 42 / D201: 43 00) 일 때	ON	OFF	ON
문자열: ABC (D100: 41 42 / D101: 43 00) 문자열: ABC (D200: 41 42 / D201: 43 00) 일 때	OFF	ON	OFF

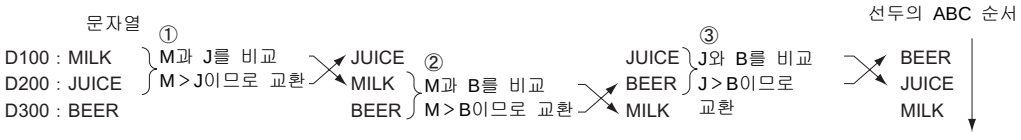
3개의 문자열을 재배열하는 예





…①③ D100과 D200을 선두로 한 2개의 문자열을 선두부터 ASCII 코드의 크기로 비교하여 D100의 문자열이 D200의 문자열보다 ASCII 코드로 큰 경우 2개의 문자열을 교환합니다.

…② D200과 D300을 선두로 한 2개의 문자열을 선두부터 ASCII 코드의 크기로 비교하여 D200의 문자열이 D230의 문자열보다 ASCII 코드로 큰 경우 2개의 문자열을 교환합니다.



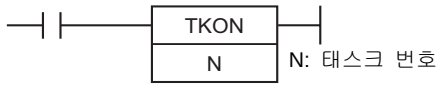
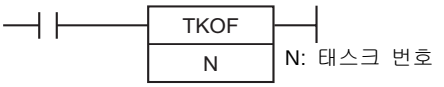
위와 같이 3개의 문자열을 ABC 순으로 재배열할 수 있습니다.

태스크 제어 명령

니모닉	명령 이름	펑션 번호	페이지
TKON	태스크 실행 기동	820	3-962
TKOF	태스크 실행 대기	821	

TKON/TKOF

명령 이름	니모닉	동작 옵션	평션 번호	기능
태스크 실행 기동	TKON	@TKON	820	태스크를 실행 가능 상태로 설정합니다.
태스크 실행 대기	TKOF	@TKOF	821	태스크를 대기 상태로 설정합니다.

심볼	TKON	TKOF
		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	태스크 번호	–	1

N: 태스크 번호

사이클릭 태스크의 경우: 10진수 0~127(사이클릭 태스크 번호 0~127에 대응)

추가 태스크의 경우: 10진수 8000~8255(인터럽트 태스크 번호 0~255에 대응)
(CS1-H/CJ1-H/CJ1M/CS1D/CJ2 CPU 유니트만 해당)

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
N	–	–	–	–	–	–	–	–	–	–	○	–	–	–	–	–	–	–

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· N의 데이터가 범위 밖일 때 ON · N으로 지정한 태스크가 존재하지 않을 때 ON · 인터럽트 태스크 내에서 실행되었을 때 ON · 그 이외일 때 OFF

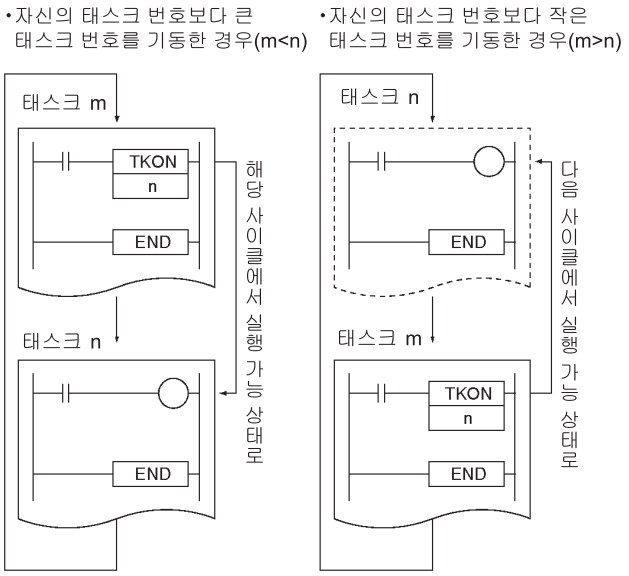
관련 릴레이

이름	주소	내용
태스크 플래그	TK00~127	· 사이클릭 태스크가 실행 가능 상태일 때 1(ON) · 사이클릭 태스크가 실행되지 않은 상태 또는 대기 상태일 때 0(OFF) · TK00~127이 사이클 실행 번호 0~127에 대응

기능

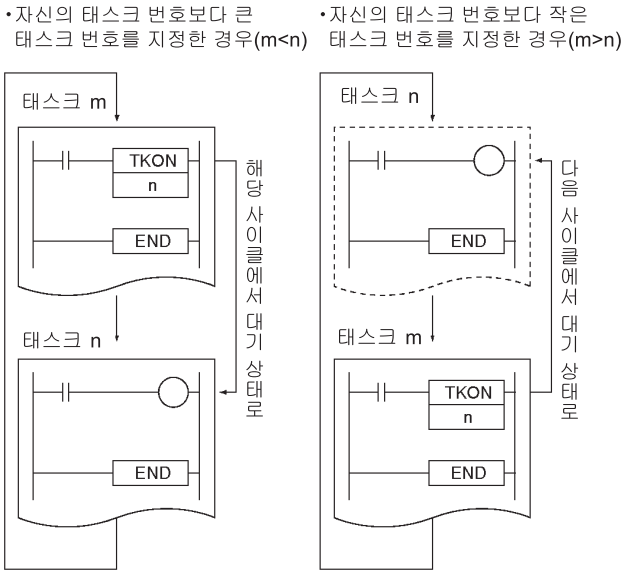
■ TKON

- N으로 지정한 사이클릭 태스크 또는 추가 태스크를 실행 가능 상태로 만듭니다. N = 0~127(사이클릭 태스크)인 경우, 동시에 대응하는 태스크 플래그(TK00~127)를 ON으로 합니다.
- 이 명령은 사이클릭 태스크와 추가 태스크로 실행할 수 있습니다. 인터럽트 태스크에서는 에러가 되어 실행할 수 없습니다.
- 이 명령에서 실행 가능 상태가 된 사이클릭 태스크 또는 추가 태스크는 TKOF 명령에 의해 대기 상태가 되지 않는 한, 다음 사이클 이후에도 실행 가능 상태가 됩니다.
- 한 태스크에서 다른 태스크를 지정하는 것도 가능합니다.
  단, 자체 태스크 번호보다 작은 태스크 번호를 실행 가능 상태로 지정한 경우는 그 태스크 번호가 해당 사이클에서 실행되지 않고 다음 사이클에서 실행됩니다.
  자체 태스크 번호보다 큰 태스크 번호를 실행 가능 상태로 지정한 경우는 그 태스크 번호가 해당 사이클에 실행됩니다. 또한 이미 태스크 플래그가 ON이 된 태스크에 대해 이 명령을 실행한 경우, NOP 처리됩니다.
  자체 태스크 번호를 실행 가능하게 지정한 경우, 이 명령은 NOP 처리됩니다.



■ TKOF

- N으로 지정한 사이클릭 태스크 또는 추가 태스크를 대기 상태로 만듭니다. N = 0~127(사이클릭 태스크)인 경우, 동시에 대응하는 태스크 플래그(TK00~127)를 OFF로 합니다.
  대기 상태란 해당 사이클 내에서 실행 상태가 되지 않는 것(프로그램을 실행 안 함)을 의미합니다.
- 이 명령은 사이클릭 태스크와 추가 태스크로 실행할 수 있습니다. 인터럽트 태스크에서는 에러가 되어 실행할 수 없습니다.
- 이 명령에서 대기 상태가 된 사이클릭 태스크 또는 추가 태스크는 TKON 명령에 의해 실행 가능 상태가 되지 않는 한, 다음 사이클 이후에도 대기 상태가 됩니다.
- 한 태스크 번호에서 다른 태스크 번호를 지정해도 됩니다.
  단, 자체 태스크 번호보다 작은 태스크 번호를 지정한 경우는 그 태스크 번호가 다음 사이클에서 대기 상태가 됩니다(해당 사이클에서는 이미 실행됨).
  자체 태스크 번호보다 큰 태스크 번호를 지정한 경우, 그 태스크 번호는 해당 사이클에서 대기 상태가 됩니다.
  자체 태스크 번호를 지정한 경우, 이 명령 실행 시 즉시 대기 상태가 되고 그 이후의 명령은 실행되지 않습니다.



힌트

- CX-Programmer에서 '운전 시작 시 기동' 옵션을 선택한 사이클릭 태스크는 운전을 시작할 때 자동으로 실행 가능 상태가 되고 기타 사이클릭 태스크는 실행되지 않은 상태가 됩니다.
  (또한 프로그래밍 콘솔로 메모리 전체 클리어 조작을 한 경우는 운전을 시작할 때 사이클릭 태스크 0만 실행 가능 상태가 됩니다.)
- 실행되지 않은 상태의 태스크는 TKON 명령에 의해 실행 가능 상태가 됩니다. 일단 실행 가능 상태가 된 사이클릭 태스크 또는 추가 태스크는 TKOF 명령에 의해 대기 상태가 됩니다.

TKON/TKOF

■ TKON

· 실행 가능 상태가 된 사이클릭 태스크/추가 태스크는 해당 사이클 내에서 태스크 번호 순서(소→대 순서)로 실행 상태가 됩니다. 따라서 실행 가능 상태가 된 태스크라도 해당 사이클 내에서 실행 순서가 돌아오기 전에 대기 상태가 된 경우는 실행되지 않습니다.

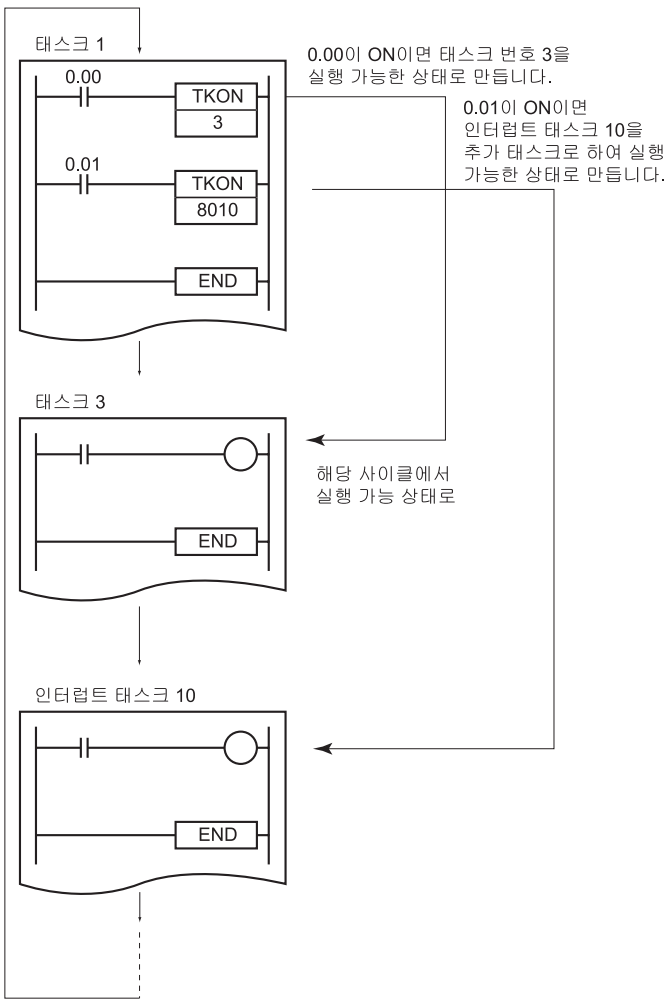
■ TKOF

· 이 명령에 의해 대기 상태가 되는 태스크는 실행 가능 상태가 된 사이클릭 태스크/추가 태스크입니다.

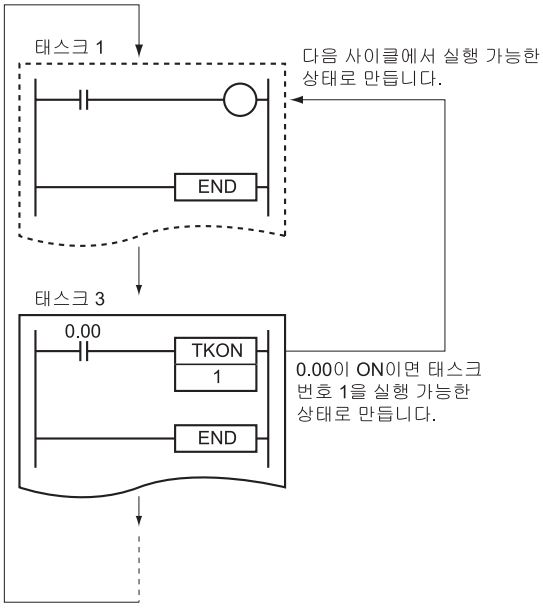
샘플 프로그램

■ TKON

(예 1)

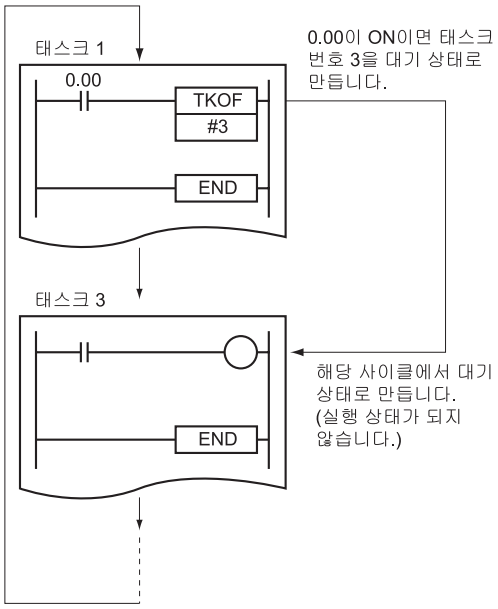


(예 2)

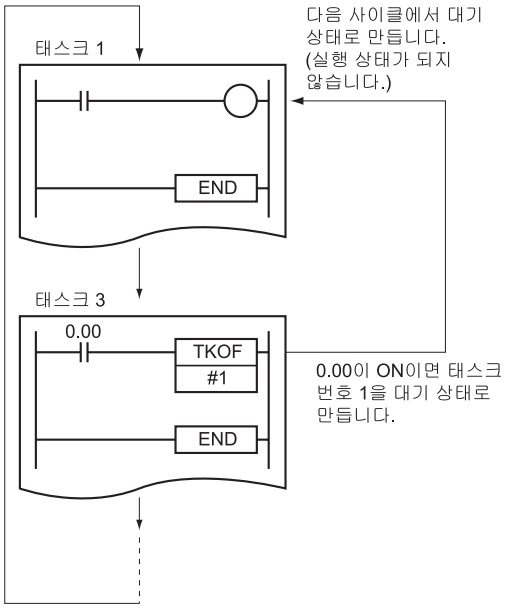


■ TKOF

(예 1)


태스크 1
0.00
TKOF
#3
END
0.00이 ON이면 태스크 번호 3을 대기 상태로 만듭니다.
태스크 3
END
해당 사이클에서 대기 상태로 만듭니다. (실행 상태가 되지 않습니다.)

(예 2)


다음 사이클에서 대기 상태로 만듭니다. (실행 상태가 되지 않습니다.)
태스크 1
END
태스크 3
0.00
TKOF
#1
END
0.00이 ON이면 태스크 번호 1을 대기 상태로 만듭니다.

기종 변환용 명령

니모닉	명령 이름	펑션 번호	페이지
XFERC	블록 전송	565	3-970
DISTC	데이터 분배	566	3-972
COLLC	데이터 추출	567	3-975
MOVBC	비트 전송	568	3-979
BCNTC	비트 카운터	621	3-981

기종 변환용 명령

기종 변환용 명령은 피연산자의 데이터 형식을 C 시리즈와 동일하게 BCD 형식으로 한 명령입니다(CS/CJ 시리즈에서는 BIN 형식). 다음 5가지 명령이 있습니다. 기종 변환용 명령의 니모닉(아래의 괄호 안)은 기존 CS/CJ 시리즈 명령의 니모닉에 C를 부가합니다.

- 블록 전송(XFERC)
- 데이터 분배(DISTC)
- 데이터 추출(COLLC)
- 비트 전송(MOVBC)
- 비트 카운터(BCNTC)

C 시리즈의 프로그램을 CS/CJ 시리즈의 프로그램으로 전환하는 경우, 이들 명령을 사용하면 피연산자에 이전과 동일한 데이터를 설정하여 프로그래밍할 수 있습니다.

또한 CX-Programmer Ver.5.0 이상에서 PLC 기종을 C 시리즈에서 CS/CJ 시리즈로 변경한 경우(주), 이들 명령의 니모닉도 자동으로 변환되므로(예: XFER→XFERC) 피연산자의 데이터를 수동으로 수정할 필요가 없습니다.

CX-Programmer Ver.4.0 이하에서 PLC 기종을 C 시리즈에서 CS/CJ 시리즈로 변경한 경우(주), 피연산자를 상수 지정할 때는 BCD→BIN으로 자동 변환되지만 채널 지정할 때는 해당 채널의 값을 BIN 데이터로 수동으로 변경해야 합니다.

주: [PLC 변경] 대화 상자에서 PLC 기종을 CS/CJ 시리즈로 변경합니다.

■ 기존 C 시리즈*1 명령과의 차이점 목록

*1: C200H, C1000H, C2000H, C200HS, C200HG/HE/HX(-Z), CQM1, CQM1H, CPM1/CPM1A, CPM2C, SRM1을 가리킵니다.

	기종 변환용 명령 (유니트 Ver.3.0 이상만 해당)		대응하는 C 시리즈 기존 명령		기종 변환용 명령의 C 시리즈 기존 명령과의 차이점(주)		CX-Programmer Ver.4.0 이하에서 PLC 기종을 C 시리즈에서 CS/CJ 시리즈로 변경한 경우	CX-Programmer Ver.5.0 이상에서 PLC 기종을 C 시리즈에서 CS/CJ 시리즈로 변경한 경우
명령 이름	니모닉	펑션 번호	니모닉	펑션 번호	C200H, C1000H, C2000H와의 차이점	C200HS, C200HG/HE/HX(-Z), CQM1, CQM1H, CPM1/CPM1A, CPM2C, SRM1과의 차이점		
블록 전송	XFERC	565	XFER	70	동일	동일	XFER 그대로 자동 변환되지만 제1피연산자(전송 CH 수)를 채널 지정하는 경우는 해당 채널의 값이 BIN 데이터가 되도록 프로그램을 수동으로 수정해야 합니다.	XFER이 XFERC로 자동 변환됩니다. 피연산자는 수정할 필요가 없습니다.
데이터 분배	DISTC	566	DIST	80	데이터 분배 동작 이외에 C200H, C1000H, C2000H가 아니었던 스택 PUSH 동작이 가능	동일(데이터 분배 동작, 스택 PUSH 동작이 가능)	DIST 그대로 자동 변환되지만 제3피연산자(S2: 오프셋 데이터)의 데이터를 채널 지정하는 경우는 해당 채널의 값이 BIN 데이터가 되도록 프로그램을 수동으로 수정해야 합니다.	DIST가 DISTC로 자동 변환됩니다. 피연산자는 수정할 필요가 없습니다.
데이터 추출	COLLC	567	COLL	81	데이터 추출 동작 이외에 C200H, C1000H, C2000H가 아니었던 스택 읽기 동작이 가능	동일(데이터 추출 동작, 스택 읽기 동작이 가능)	COLL 그대로 자동 변환되지만 제2피연산자(S2: 오프셋 데이터)를 채널 지정하는 경우는 해당 채널의 값이 BIN 데이터가 되도록 프로그램을 수동으로 수정해야 합니다.	COLL이 COLLC로 자동 변환됩니다. 피연산자는 수정할 필요가 없습니다.

	기종 변환용 명령 (유니트 Ver.3.0 이상만 해당)		대응하는 C 시리즈 기존 명령		기종 변환용 명령의 C 시리즈 기존 명령과의 차이점(주)		CX-Programmer Ver.4.0 이하에서 PLC 기종을 C 시리즈에서 CS/CJ 시리즈로 변경한 경우	CX-Programmer Ver.5.0 이상에서 PLC 기종을 C 시리즈에서 CS/CJ 시리즈로 변경한 경우
명령 이름	니모닉	펑션 번호	니모닉	펑션 번호	C200H, C1000H, C2000H와의 차이점	C200HS, C200HG/HE/HX(-Z), CQM1, CQM1H, CPM1/CPM1A, CPM2C, SRM1과의 차이점		
비트 전송	MOVBC	568	MOVB	82	동일	동일	MOVB 그대로 자동 변환되지만 제2피연산자(C: 제어 데이터)를 채널 지정하는 경우는 해당 채널의 값이 BIN 데이터가 되도록 프로그램을 수동으로 수정해야 합니다.	MOVB가 MOVBC로 자동 변환됩니다. 피연산자는 수정할 필요가 없습니다.
비트 카운터	BCNTC	621	BCNT	67	동일	동일	BCNT 그대로 자동 변환되지만 제1피연산자(카운트 CH 수)를 채널 지정하는 경우는 해당 채널의 값이 BIN 데이터가 되도록 프로그램을 수동으로 수정해야 합니다.	BCNT가 BCNTC로 자동 변환됩니다. 피연산자는 수정할 필요가 없습니다.

주) 조건 플래그의 동작은 다음과 같은 점에서 다릅니다. 자세한 내용은 각 명령어의 조건 플래그에 대한 설명을 참조하십시오.

- 모든 명령에서 DM 간접(*DM)의 내용이 BCD 데이터가 아닐 때 또는 DM 영역을 초과했을 때의 조건 플래그 동작이 다릅니다.
- DISTC 명령에서 스택 PUSH 동작과 관련된 조건 플래그의 동작이 C200H, C1000H, C2000H와 다릅니다.
- COLLC 명령에서 스택 읽기 동작과 관련된 조건 플래그의 동작이 C200H, C1000H, C2000H와 다릅니다.

■ 기존 CS/CJ 시리즈 명령과의 차이점 목록

	기종 변환용 명령		대응하는 CS/CJ 시리즈의 기존 명령		기종 변환용 명령의 기존 CS/CJ 시리즈 명령과의 차이점(주)
명령 이름	니모닉	펑션 번호	니모닉	펑션 번호	
블록 전송	XFERC	565	XFER	070	제1피연산자(W: 전송 CH 수)의 데이터가 정수(10진수 &0~65535 또는 16진수 #0000~FFFF)에서 BCD(#0000~9999)로 변경됨
데이터 분배	DISTC	566	DIST	080	데이터 분배 동작 이외에 CS/CJ 시리즈가 아닌 스택 PUSH 동작이 가능 제3피연산자(S2: 오프셋 데이터)의 데이터가 정수(10진수 &0~65535 또는 16진수 #0000~FFFF)에서 BCD(데이터 분배 동작 시: #0000~7999, 스택 PUSH 동작 시: #0000~9999)로 변경됨
데이터 추출	COLLC	567	COLL	081	데이터 추출 동작 이외에 CS/CJ 시리즈가 아닌 스택 읽기 동작이 가능 제2피연산자(S2: 오프셋 데이터)의 데이터가 정수(10진수 &0~65535 또는 16진수 #0000~FFFF)에서 BCD(데이터 분배 동작 시: #0000~7999, 스택 읽기 동작 시: FIFO: #9000~9999, LIFO: #8000~8999)로 변경됨
비트 전송	MOVBC	568	MOVB	082	제2피연산자(C: 제어 데이터)의 전송 원본 CH의 비트 지정, 전송 대상 CH의 비트 지정이 정수(&0~15(10진수) 또는 #00~0F(16진수))에서 BCD(#00~15)로 변경됨
비트 카운터	BCNTC	621	BCNT	067	제1피연산자(W: 카운트 CH 수)의 데이터가 정수(10진수 &0~65535 또는 16진수 #0000~FFFF)에서 BCD(#0000~9999)로 변경됨 제3피연산자(D: 카운트 결과 출력 CH 번호)에 대한 저장 데이터 형식이 정수(10진수 &0~65535 또는 16진수 #0000~FFFF)에서 BCD(#0000~9999)로 변경됨

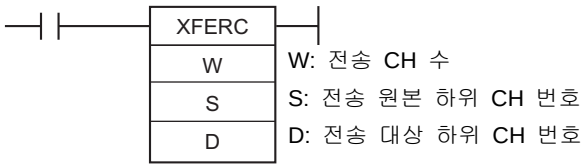
주) 조건 플래그의 동작은 다음과 같은 점에서 다릅니다. 자세한 내용은 각 명령어의 조건 플래그에 대한 설명을 참조하십시오.

- 위의 피연산자의 데이터가 BCD가 아닐 때는 에러 플래그가 ON이 됩니다.
- DISTC 명령에 스택 PUSH 동작과 관련된 조건 플래그의 동작이 추가되었습니다.
- COLLC 명령에 스택 읽기 동작과 관련된 조건 플래그의 동작이 추가되었습니다.

XFERC

명령 이름	니모닉	동작 옵션	평션 번호	기능
블록 전송	XFERC	@ XFERC	565	연속된 여러 CH 데이터를 일괄 전송합니다.

심볼	XFERC



특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

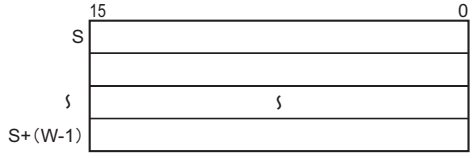
피연산자 설명

피연산자	내용	데이터 형식	크기
W	전송 CH 수	WORD	1
S	전송 원본 하위 CH 번호	WORD	가변
D	전송 대상 하위 CH 번호	WORD	가변

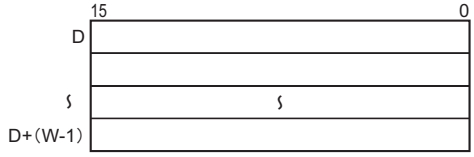
W: 전송 CH 수

BCD #0000~9999

S: 전송 원본 하위 CH 번호



D: 전송 대상 하위 CH 번호



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
W	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
S											–	–						
D																		

조건 플래그의 동작

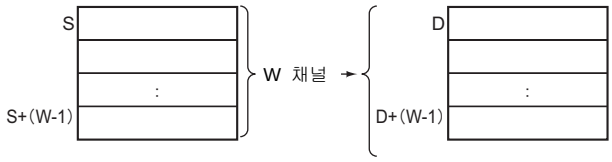
이름	라벨	내용
에러 플래그	P_ER	· W의 데이터가 BCD가 아닐 때 ON

주:

· C 시리즈의 XFER 명령의 경우, DM 간접(* DM)의 내용이 BCD 데이터가 아닐 때 또는 DM 영역을 초과했을 때 ON이 되지만, 이 명령에서는 ON이 되지 않습니다.

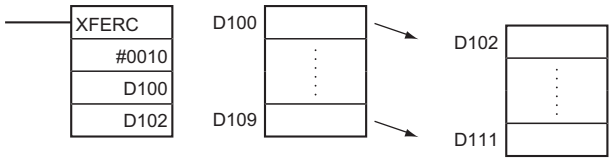
기능

S로 지정한 전송 원본 하위 CH 번호부터 W로 지정한 데이터 수(BCD)만큼 D로 지정한 전송 대상 하위 CH 번호 이후에 전송합니다.



힌트

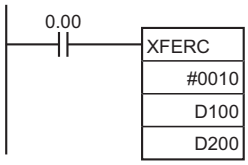
- S와 D에 같은 영역 종류를 지정하고 XFERC 명령을 사용하여 데이터를 이동할 수도 있습니다.



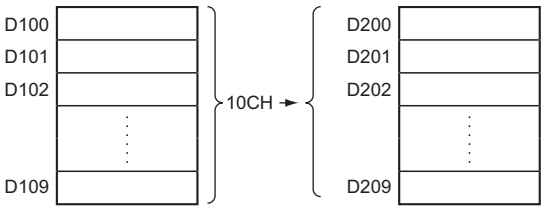
사용상의 주의 사항

- 대량의 채널을 블록 전송하는 경우 명령 실행 시간이 걸립니다. 따라서 이 명령 실행 시에 전원 OFF가 발생하는 경우 블록 전송 도중에 실행이 종료되는 경우가 있으므로 주의하십시오.
- 전송 원본, 전송 대상 CH가 데이터 영역을 초과하지 않도록 하십시오.

샘플 프로그램

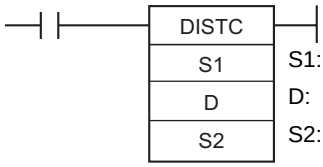


0.00이 ON이면 D100~D109의 10CH를 D200~D209로 전송합니다.



DISTC

명령 이름	니모닉	동작 옵션	평션 번호	기능
데이터 분배	DISTC	@DISTC	566	전송 대상을 기준으로 오프셋 지정된 채널로 전송 데이터를 전송합니다.

심볼	DISTC
	

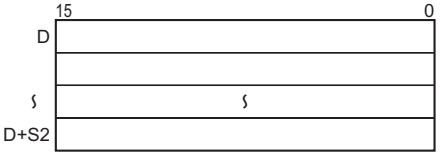
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S1	전송 데이터	WORD	1
D	전송 대상 기준 CH 번호	WORD	1
S2	오프셋 데이터	WORD	1

D: 전송 대상 기준 CH 번호



S2: 오프셋 데이터

- 데이터 분배 동작 시: BCD #0000~7999
- 스택 PUSH 동작 시: BCD #9000~9999

주: D~D+S2는 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S1	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
D											–	–						
S2											○	○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S2의 데이터가 BCD가 아닐 때 ON · 스택 PUSH 동작 시, 스택 포인터(D)의 데이터가 BCD가 아닐 때 ON · 스택 PUSH 동작 시, 전송 대상이 스택 데이터 영역을 초과했을 때 ON
= 플래그	P_EQ	· 전송 데이터 S1이 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 전송 데이터의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

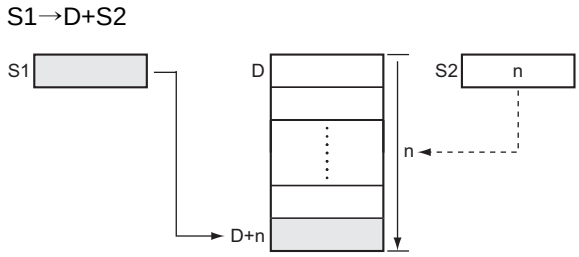
주:

· C 시리즈의 DIST 명령의 경우, DM 간접(* DM)의 내용이 BCD 데이터가 아닐 때 또는 DM 영역을 초과했을 때 에러 플래그가 ON이 되지만, 이 명령에서는 ON이 되지 않습니다.

기능

■ 데이터 분배 동작

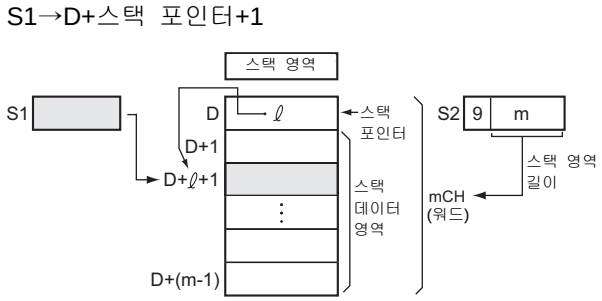
D로 지정한 전송 대상 기준 CH 번호부터 S2로 지정한 오프셋 데이터만큼 이동한 주소에 S1을 전송합니다.



■ 스택 PUSH 동작

S2의 최상위 BCD 1자리(비트 12~15)에 9(BCD)를 설정한 경우, 하위 BCD 3자리(비트 00~11)로 지정한 채널(워드) 수 분량(m)의 스택 영역이 전송 대상 기준 CH 번호 D 이후에 확보됩니다. 이때, D는 스택 포인터가 되고 D+1 이후가 스택 데이터 영역이 됩니다.

D로 지정한 전송 대상 기준 CH부터 스택 포인터(D의 내용: l)+1CH만큼 이동한 주소에 S1을 전송합니다.



스택 데이터 영역에 S1의 데이터가 전송될 때마다 D의 내용의 스택 포인터(l)가 자동으로 1씩 증가합니다.

주: 스택 PUSH 동작에서 사용할 때 한번 DISTC 명령으로 확보한 스택 영역을 다른 DISTC 명령에서 사용하고 싶은 경우, 반드시 스택 영역 길이를 최초에 DISTC 명령으로 확보한 값과 동일한 크기로 지정하십시오. 다른 크기로 지정하면 동작이 보장되지 않습니다.

힌트

· 스택 영역으로 전송한 데이터를 읽을 때는 COLLC 명령을 사용하면 편리합니다.

· 오프셋 데이터(S2)의 내용을 변경하면 DISTC 명령 하나로 임의의 위치에 데이터를 전송(분배)할 수 있습니다.

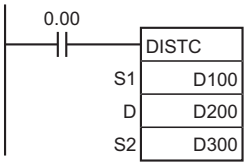
사용상의 주의 사항

· 오프셋 데이터(S2)의 내용은 전송 대상의 영역 범위를 벗어나지 않도록 하십시오.

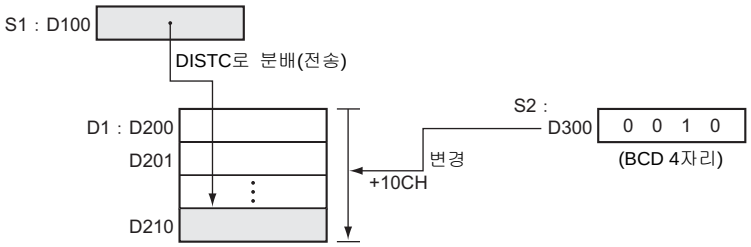
DISTC

샘플 프로그램

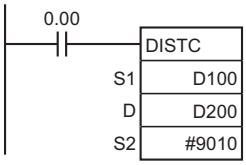
■ 데이터 분배 동작



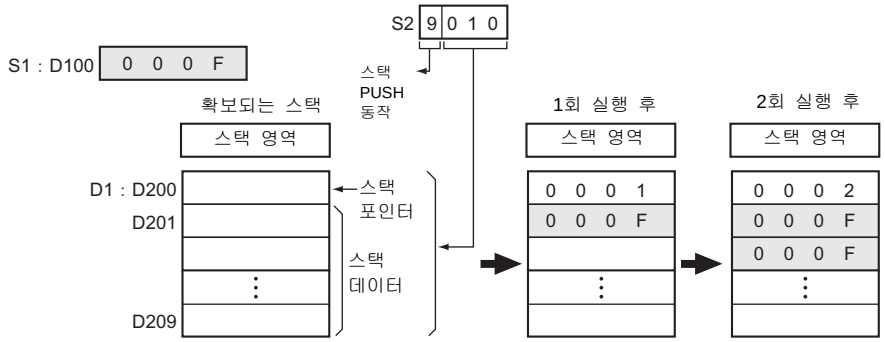
D300의 내용에서 최상위 1자리가 0이므로 데이터 분배 동작을 수행합니다.
0.00이 ON이면 D200에 D300의 내용(예: 10(BCD 4자리 #0010))을 가산한 주소(D210)로 D100의 내용을 전송합니다. (D300의 내용을 변경하면 D100의 내용을 임의의 주소로 분배할 수 있습니다.)



■ 스택 PUSH 동작



S2의 데이터에서 최상위 1자리가 9이므로 스택 PUSH 동작을 수행합니다.
0.00이 ON이면 D200을 선두로 하여 S2의 내용 #9010의 하위 3자리 #010으로 지정한 채널(워드) 수 분량의 스택 영역 D200~209가 확보됩니다.
동시에 D200+스택 포인터+1CH로 D100의 내용을 전송하고 스택 포인터의 값에 1을 더합니다.



COLLC

명령 이름	니모닉	동작 옵션	평션 번호	기능
데이터 추출	COLLC	@COLLC	567	전송 원본을 기준으로 오프셋 지정된 채널의 내용을 지정 채널로 전송합니다.

심볼	COLLC



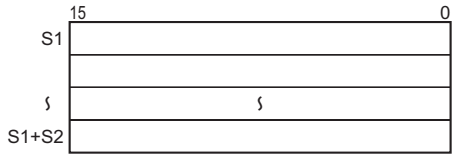
특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S1	전송 원본 기준 CH 번호	WORD	1
S2	오프셋 데이터	WORD	1
D	전송 대상 CH 번호	WORD	1

S1: 전송 원본 기준 CH 번호



S2: 오프셋 데이터

- 데이터 추출 동작 시: BCD #0000~7999
- 스택 읽기 동작 시 : FIFO: BCD #9000~9999
  : LIFO: BCD #8000~8999

주: D~D+S는 동일 영역 종류이어야 합니다.

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S1	○	○	○	○	○	○	○	○	○	○	–	–	–	○	–	–	–	–
S2											○	○						
D											–							

COLLC

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S2의 데이터가 BCD가 아닐 때 ON · 스택 읽기 동작 시, 스택 포인터(S1)의 데이터가 BCD가 아닐 때 ON · 스택 읽기 동작 시, 읽기 원본이 스택 데이터 영역을 초과했을 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 전송 데이터가 0일 때 ON · 그 이외일 때 OFF
부정 플래그	P_N	· 전송 데이터의 최상위 비트가 1일 때 ON · 그 이외일 때 OFF

주:

· C 시리즈의 COLL 명령의 경우, DM 간접(*DM)의 내용이 BCD 데이터가 아닐 때 또는 DM 영역을 초과했을 때 에러 플래그가 ON이 되지만, 이 명령에서는 ON이 되지 않습니다.

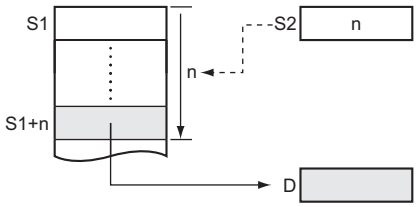
· 전송 데이터 내용의 최상위 비트가 1이면 부정 플래그가 ON이 됩니다.

기능

■ 데이터 추출 동작 시

S1로 지정한 전송 원본 기준 CH에서 S2로 지정한 오프셋 데이터만큼 이동한 주소의 데이터를 D에 전송합니다.

오프셋 데이터(S2)의 내용을 변경하면 COLL 명령 하나로 임의의 위치에서 데이터를 가져올(추출) 수 있습니다.



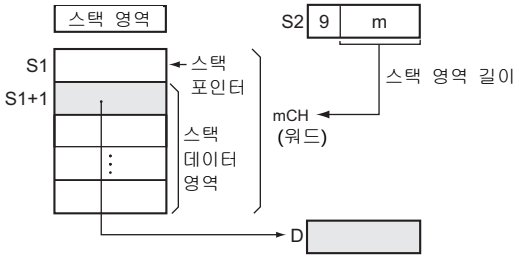
■ 스택 읽기 동작

· FIFO(선입선출) 방식

S2의 최상위 BCD 1자리(비트 12~15)에 #9(BCD)를 설정한 경우, FIFO(선입선출) 방식으로 하위 BCD 3자리(비트 00~11)로 지정한 채널(워드) 수 분량(m)의 스택 영역에서 데이터를 읽어 D에 전송합니다.

FIFO

S1+1CH에서 읽기

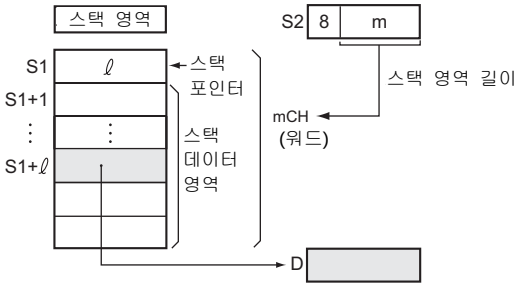


· LIFO(후입선출) 방식

S2의 최상위 BCD 1자리(비트 12~15)에 #8(BCD)을 설정한 경우, LIFO(후입선출) 방식으로 하위 BCD 3자리(비트 00~111)로 지정한 채널(워드) 수 분량(m)의 스택 영역에서 데이터를 읽어 D에 전송합니다.

LIFO

S1+스택 포인터 CH에서 읽기



어느 경우(방식)든 S1은 스택 포인터가 됩니다.

· FIFO 방식의 경우, S1+1에서 데이터를 읽습니다.

· LILO 방식의 경우, S1+(스택 포인터)에서 읽습니다.

· 어느 경우(방식)든 스택 영역에서 데이터를 읽을 때마다 스택 포인터의 값이 자동으로 1씩 감소됩니다.

주: 스택 읽기 동작에서 사용할 때 한번 DISTC 명령으로 확보한 스택 영역을 COLLC 명령에서 지정하는 경우, 반드시 스택 영역 길이를 최초에 DISTC 명령으로 확보한 값과 동일한 크기로 지정하십시오. 다른 크기로 지정하면 동작이 보장되지 않습니다.

힌트

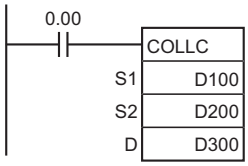
· 스택 영역에 데이터를 저장할 때는 DISTC 명령을 사용하면 편리합니다.

사용상의 주의 사항

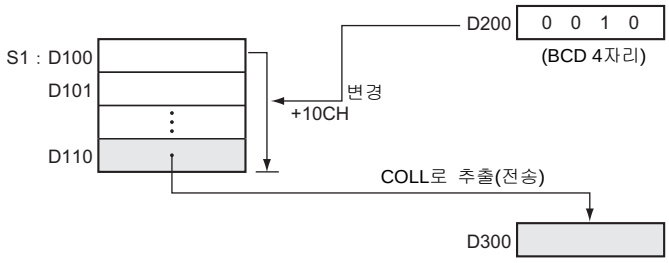
· 오프셋 데이터(S2)의 내용은 전송 원본의 영역 범위를 벗어나지 않도록 하십시오.

샘플 프로그램

■ 데이터 추출 동작

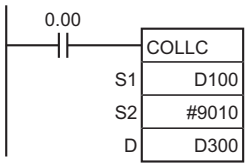


D200의 내용에서 최상위 1자리가 0이므로 데이터 추출 동작을 수행합니다.
0.00이 ON이면 D100에 D200의 내용(예: 10(BCD 4자리 #0010))을 가산한 주소(D110)의 내용을 D300으로 전송합니다. (D200의 내용을 변경하면 임의의 주소에서 추출할 수 있습니다.)

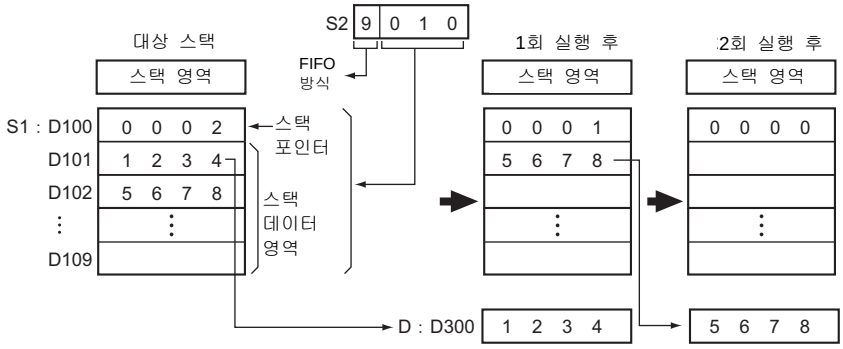


■ 스택 읽기 동작

· FIFO(선입선출) 방식

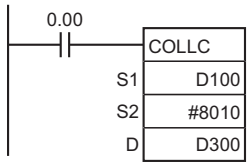


S2의 데이터에서 최상위 1자리가 #9이므로 FIFO 방식의 스택 추출 동작을 수행합니다.
0.00이 ON이면 S2의 데이터의 하위 3자리 #010으로 지정한 채널(워드) 수 분량의 스택 영역 D100~109가 확보됩니다.
동시에 S1+1의 D101의 내용을 D300으로 전송하고 스택 포인터의 값에서 1을 뺍니다.

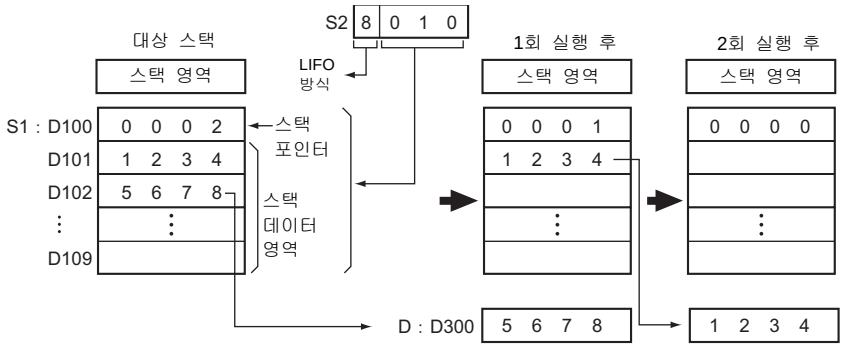


COLLC

· LIFO(후입선출) 방식

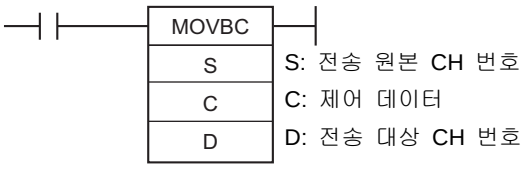


S2의 데이터에서 최상위 1자리가 #8이므로 LIFO 방식의 스택 추출 동작을 수행합니다.
0.00이 ON이면 S2의 데이터의 하위 3자리 #010으로 지정한 채널(워드) 수 분량의 스택 영역 D100~109가 확보됩니다.
동시에 S1+1의 D101의 내용을 D300으로 전송하고 스택 포인터의 값에서 1을 뺍니다.



MOVBC

명령 이름	니모닉	동작 옵션	펑션 번호	기능
비트 전송	MOVBC	@MOVBC	568	지정한 비트를 전송합니다.

심볼	MOVBC
	

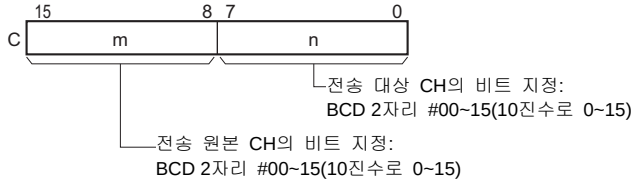
특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	전송 원본 CH 번호	WORD	1
C	제어 데이터	WORD	1
D	전송 대상 CH 번호	WORD	1

C: 제어 데이터



■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
C																		
D											–							

조건 플래그의 동작

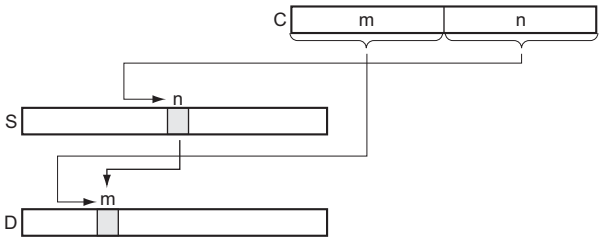
이름	라벨	내용
에러 플래그	P_ER	· C의 데이터가 범위 밖(BCD 데이터가 아닐 때 포함)일 때 ON · 그 이외일 때 OFF

주: C 시리즈의 MOVB 명령의 경우, DM 간접(* DM)의 내용이 BCD 데이터가 아닐 때 또는 DM 영역을 초과했을 때 ON이 되지만, 이 명령에서는 ON이 되지 않습니다.

MOVBC

기능

S의 지정된 비트 위치(C의 n)의 내용(0/1)을 D의 지정된 비트 위치(C의 m)에 전송합니다.

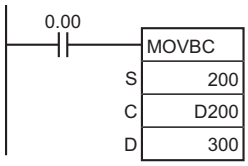


주:
- 전송 대상 CH의 데이터는 전송된 비트 이외에는 변경되지 않습니다.

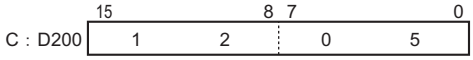
힌트

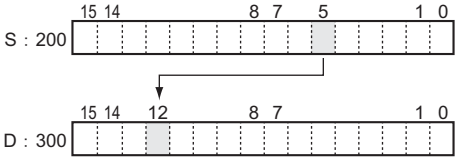
- S와 D에 동일 채널을 지정하면 비트 위치 변경 등에 사용할 수 있습니다.

샘플 프로그램



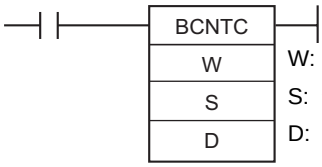
0.00이 ON일 때 제어 데이터(C)의 내용이 다음과 같은 경우 200CH의 비트 5를 300CH의 비트 12로 전송합니다.





BCNTC

명령 이름	니모닉	동작 옵션	평션 번호	기능
비트 카운터	BCNTC	@BCNTC	621	CH 데이터 내에서 "1"인 비트의 총 수를 계산합니다.

심볼	BCNTC
	

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
W	카운트 CH 수	WORD	1
S	카운트 하위 CH 번호	UNIT	가변
D	카운트 결과 출력 CH 번호	WORD	1

W: 카운트 CH 수
BCD #0001~9999

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 직접	IR 간접				
W	○	○	○	○	○	○	○	○	○	○	○	○	–	○	–	–	–	–
S											–	–						
D												○						

조건 플래그의 동작

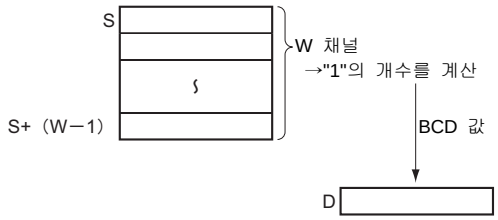
이름	라벨	내용
에러 플래그	P_ER	· W의 데이터가 BCD #0001~9999의 범위 안에 없을 때 ON · 카운트 결과(D의 데이터)가 BCD #9999를 초과할 때 ON · W의 데이터가 0일 때 ON · 그 이외일 때 OFF
= 플래그	P_EQ	· 카운트 결과(D의 데이터)가 0일 때 ON · 그 이외일 때 OFF

주:
· C 시리즈의 BCNT 명령의 경우, DM 간접(*DM)의 내용이 BCD 데이터가 아닐 때 또는 DM 영역을 초과했을 때 에러 플래그가 ON이 되지만, 이 명령에서는 ON이 되지 않습니다.

BCNTC

기능

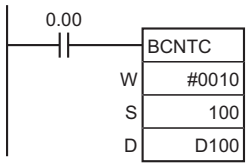
S로 지정한 카운트 하위 CH 번호부터 지정된 CH 수(W) 분량의 데이터에 대해 "1"인 비트의 총 수를 계산하여 결과를 BCD 값(#0000~9999)으로 D에 출력합니다.



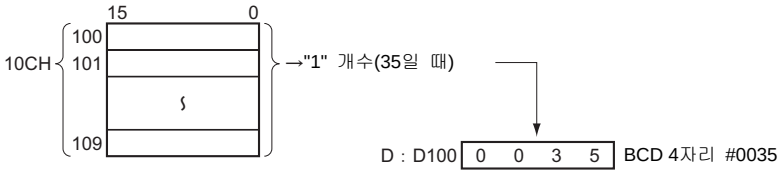
사용상의 주의 사항

· 카운트 종료 CH의 S+(W−1)이 S로 지정하는 영역 종류의 최대 범위를 초과하지 않도록 하십시오.

샘플 프로그램



0.00이 ON이면 100CH부터 10CH의 데이터 중 "1"의 개수를 세어 BCD 값으로 D100에 저장합니다.

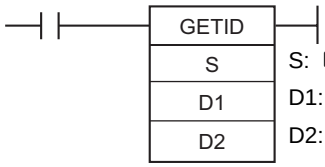


펑션 블록용 특수 명령

니모닉	명령 이름	펑션 번호	페이지
GETID	변수 종류 취득	286	3-984

GETID

명령 이름	니모닉	동작 옵션	펑션 번호	기능
변수 종류 취득	GETID	@ GETID	286	지정된 변수(또는 주소)의 FINS 명령의 변수 종류(I/O 메모리 영역 종류) 코드 및 CH 번호를 출력합니다.

	GETID
심볼	 S: 대상 변수 또는 주소 D1: 변수 종류(I/O 메모리 영역 종류) 코드 D2: CH 번호

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
S	대상 변수 또는 주소	WORD	1
D1	변수 종류(I/O 메모리 영역 종류) 코드	WORD	1
D2	CH 번호	WORD	1

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	○	○	○	○	○	○	–	○	–	○	–	–	–	–
D1																		
D2																		

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· S가 범위 밖일 때 ON · 그 이외일 때 OFF

기능

S로 지정한 변수 또는 주소의 변수 종류(I/O 메모리 영역 종류) 코드를 D1에 출력하고 동시에 CH 번호(오프셋)를 D2에 출력합니다.

· 변수 종류(I/O 메모리 영역 종류) 및 CH 번호:

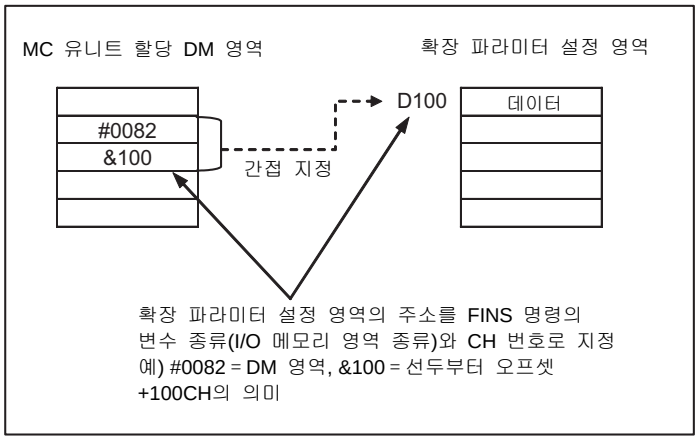
영역 종류		데이터 크기	변수 종류 코드: D1에 출력(16진수)	CH 번호: D2에 출력(10진수)
채널 I/O	CIO	채널	#00B0	&0000~6143
내부 보조 릴레이	W		#00B1	&000~511
유지 릴레이	H		#00B2	&000~511
데이터 메모리	D		#0082	&000~32767
확장 데이터 메모리	E 뱅크 번호 0~C		#0050~005C	&000~32767

힌트

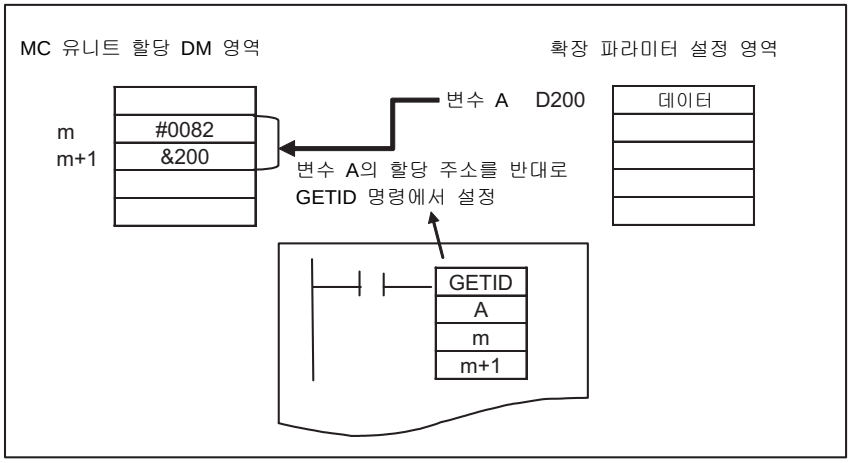
- 펑션 블록 내의 변수는 AT 지정을 하지 않는 한 CX-Programmer Ver.5.0 이상의 시스템에 의해 주소가 자동 할당됩니다.
  따라서, 예를 들어 고기능 유니트(예: MC 유니트)의 확장 파라미터 설정 영역을 간접 지정하는 경우, 확장 파라미터 설정 영역의 선두에 변수를 사용하면 간접 지정을 통해 해당 변수의 주소를 반대로 설정해야 합니다. 이런 경우 이 명령을 사용하여 변수의 할당 주소를 취득하여 반대로 설정합니다.

샘플 프로그램

● 통상



● 펑션 블록 사용 시



SFC 관련 명령

니모닉	명령 이름	펑션 번호	페이지
SA	스텝 활성화 명령	784	3-990
SE	스텝 비활성화 명령	785	
TSR	스텝 타이머 현재값 읽기 명령	780	3-992
TSW	스텝 타이머 현재값 변경 명령	781	
SFCON	SFC 태스크 실행 명령	789	3-994
SFCOFF	SFC 태스크 종료 명령	790	
SFCPR	SFC 태스크 정지 명령(리셋 있음)	793	3-996
SFCPRN	SFC 태스크 정지 명령(리셋 없음)	791	

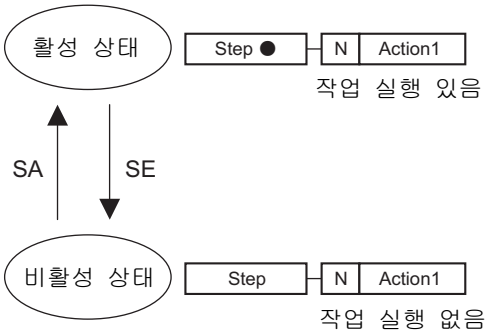
SFC 스텝 제어 명령

스텝 제어 명령은 SFC 프로그램의 스텝 및 서브 차트의 활성 상태를 제어합니다.

여기에는 다음 2가지 명령이 있습니다.

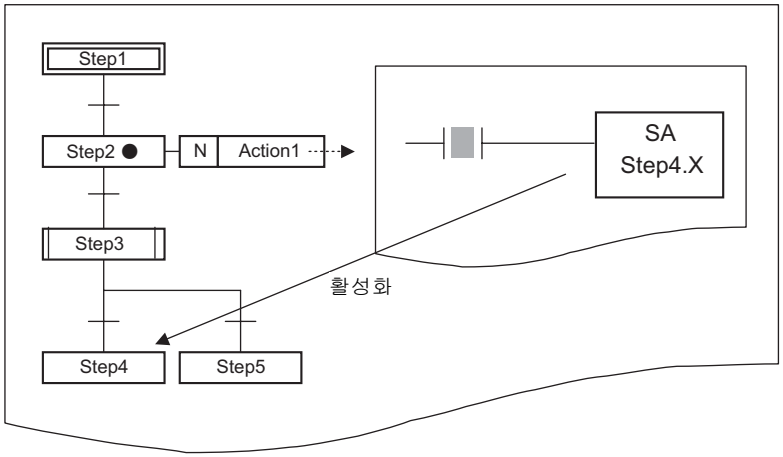
명령 이름	니모닉	개요
스텝 활성화 명령	SA	지정된 스텝 및 서브 차트를 활성화 상태로 하고 작업의 실행을 시작합니다.
스텝 비활성화 명령	SE	지정된 스텝 및 서브 차트를 비활성화 상태로 하고 작업의 실행을 정지시킵니다.

스텝 제어 명령을 통해 다음과 같이 상태의 이동을 실행합니다.

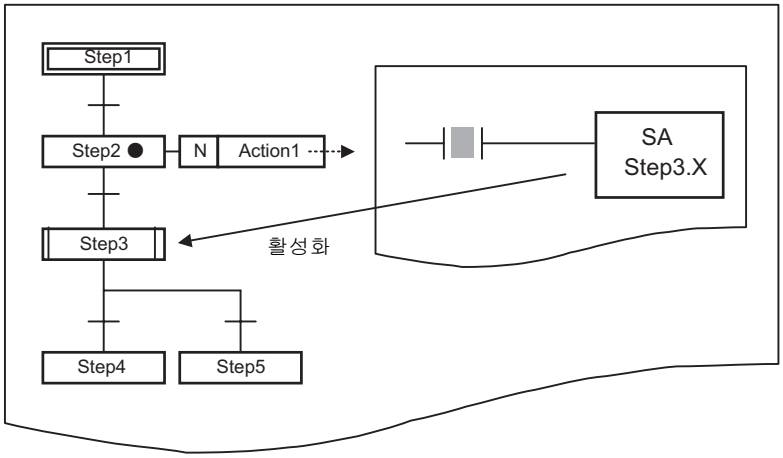


주: '●'는 스텝의 활성 상태를 나타냅니다.

예 1: Step2에 있는 Action1에서 Step4를 활성 상태로 만들 수 있습니다.



예 2: 서브 차트 스텝을 지정하면 서브 차트를 활성화시킬 수도 있습니다.

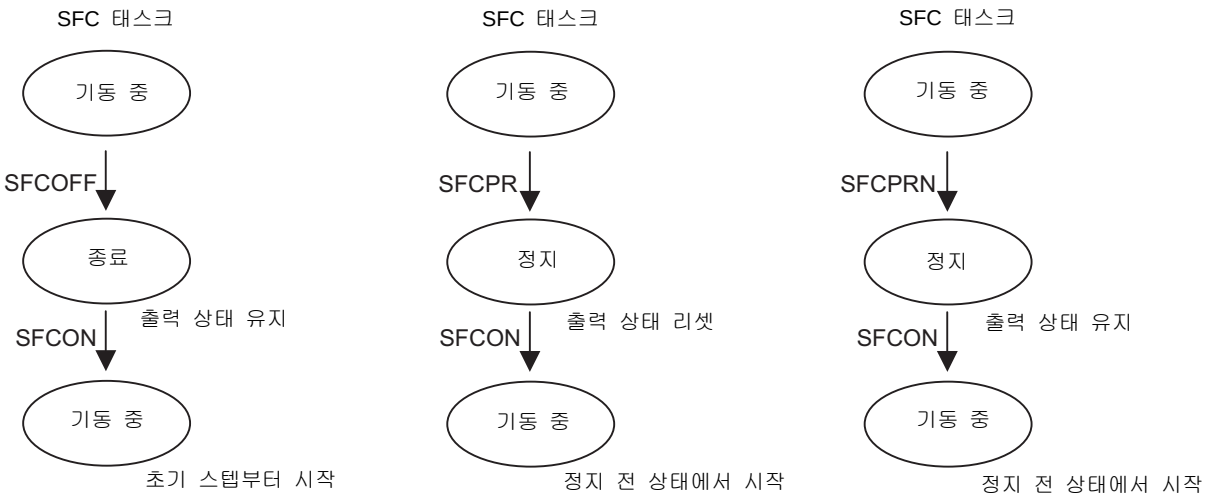


SFC 태스크 제어 명령

SFC 태스크 제어 명령을 사용하여 다음과 같은 제어를 수행할 수 있습니다.

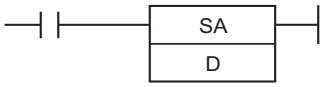
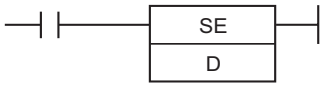
- SFC 태스크의 실행을 종료(SFCOFF)
- SFC 태스크의 실행을 일시 정지하고 출력을 리셋(SFCPR)
- SFC 태스크의 실행을 일시 정지하고 출력을 유지(SFCPRN)
- 종료/일시 정지한 SFC 태스크의 실행을 재개(SFCON)

SFC 실행 종료 SFCOFF	실행 중인 SFC 태스크를 종료합니다. SFCOFF로 지정된 태스크는 실행 중인 작업의 출력 상태를 모두 유지하고 처리를 종료합니다. SFCON 명령으로 실행을 재개한 경우 초기 스텝부터 처리를 시작합니다.
SFC 태스크 정지 (리셋 있음) SFCPR	실행 중인 SFC 태스크를 일시 정지합니다. SFCPR로 지정된 태스크는 실행 중인 태스크 내의 출력을 모두 리셋하고 처리를 일시 정지합니다. 이 명령 사용 시, 리셋 대상인 코일을 다른 태스크에서 사용하는 경우는 주의하십시오. SFCON 명령으로 실행을 재개한 경우, 이전에 정지한 스텝의 활성 상태에서 처리를 재개합니다. 정지한 스텝의 스텝 타이머는 정지 중에도 계속 카운트됩니다.
SFC 태스크 정지 (리셋 없음) SFCPRN	실행 중인 SFC 태스크를 일시 정지합니다. SFCPRN으로 지정된 태스크는 실행 중인 작업의 출력 상태를 모두 유지하고 처리를 일시 정지합니다. SFCON 명령으로 실행을 재개한 경우, 이전에 정지한 스텝의 활성 상태에서 처리를 재개합니다. 정지한 스텝의 스텝 타이머는 정지 중에도 계속 카운트됩니다.
SFC 실행 시작 SFCON	SFC 태스크 제어 명령에 의해 종료 또는 일시 정지된 SFC 태스크의 실행을 재개합니다.



SA/SE

명령 이름	니모닉	동작 옵션	평션 번호	기능
스텝 활성화 명령	SA	@SA	784	지정된 스텝 및 서브 차트를 활성화 상태로 하고 작업의 실행을 시작합니다.
스텝 비활성화 명령	SE	@SE	785	지정된 스텝 및 서브 차트를 비활성화 상태로 하고 작업의 실행을 정지시킵니다.

심볼	SA	SE
		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
D	스텝 플래그	BOOL	1

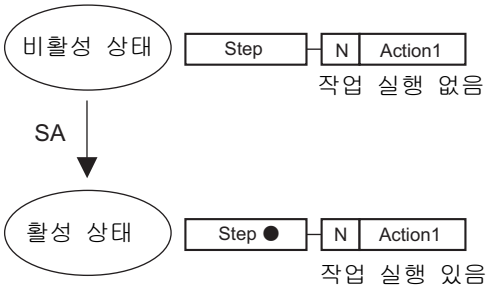
조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF

기능

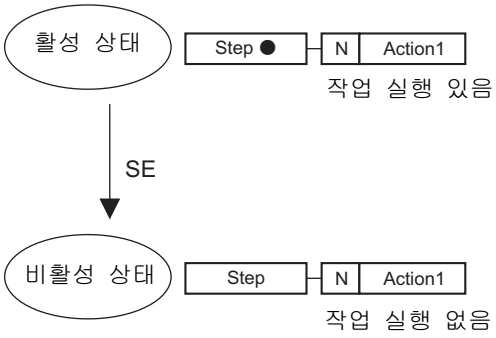
■ SA

D로 지정한 스텝을 활성화시킵니다.



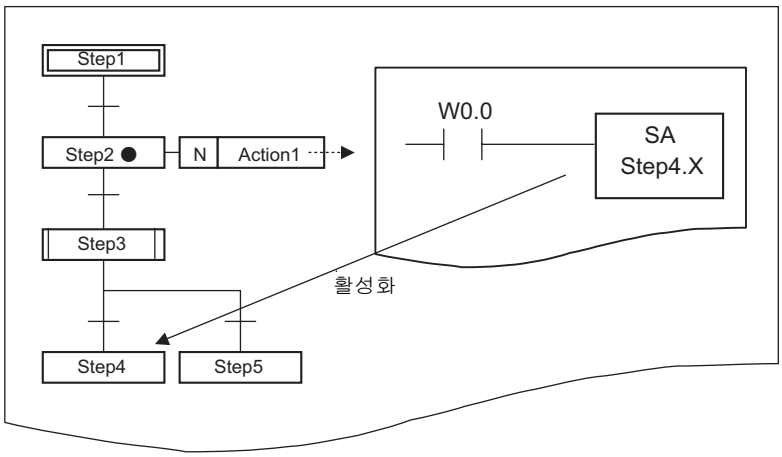
■ SE

D로 지정한 스텝을 비활성화시킵니다.

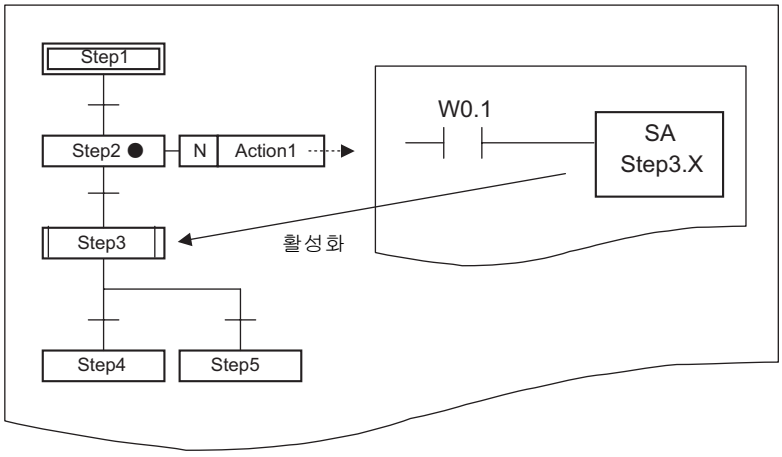


샘플 프로그램

접점 W0.0을 ON하면 Step4가 활성화됩니다.



접점 W0.1을 ON하면 Step3(서브 차트 스텝)이 활성화됩니다.

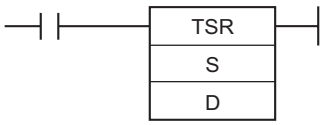
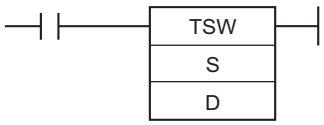


사용상의 주의 사항

- 스텝 플래그는 '[스텝명].X'로 지정합니다. 자세한 내용은 'CX-Programmer 조작 설명서 SFC편'을 참조하십시오.
- 서브 차트 내의 스텝은 지정할 수 없습니다.
- 이 명령은 프로그래밍 콘솔에서 입력 및 표시할 수 없습니다. 대신'?'가 표시됩니다.

TSR/TSW

명령 이름	니모닉	동작 옵션	평션 번호	기능
스텝 타이머 현재값 읽기 명령	TSR	@TSR	780	S로 지정한 스텝의 스텝 타이머 현재값을 D에 저장합니다.
스텝 타이머 현재값 변경 명령	TSW	@TSW	781	D로 지정한 스텝의 스텝 타이머 값을 S로 지정한 값으로 변경합니다.

심볼	TSR	TSW
		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	가능	가능

피연산자 설명

■ TSR

피연산자	내용	데이터 형식	크기
S	스텝 타이머	UINT	1
D	스텝 타이머 현재값 저장 대상 CH 번호	UINT	1

■ TSW

피연산자	내용	데이터 형식	크기
S	스텝 타이머 변경값 저장 대상 CH 번호	UINT	1
D	스텝 타이머	UINT	1

■ 영역 종류

내용	채널 주소								간접 DM/EM		상수	레지스터 참조			TK	조건 플래그	클록 펄스	TR
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	* DM * EM		DR	IR 직접	IR 간접				
S	○	○	○	○	–	–	○	○	–	–	–	–	–	–	–	–	–	–
D											–	○						

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	OFF

기능

■ TSR

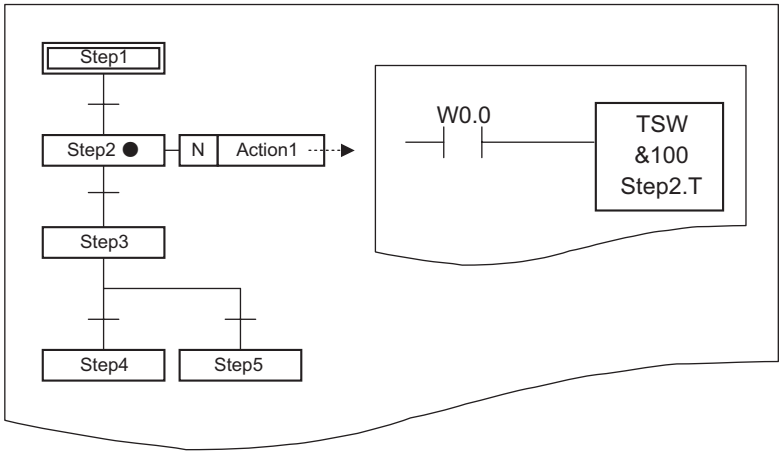
S로 지정한 스텝의 스텝 타이머 현재값을 D에 저장합니다.

■ TSW

D로 지정한 스텝의 스텝 타이머 값을 S로 지정한 값으로 변경합니다.

샘플 프로그램

접점 W0.0을 ON하면 Step2의 스텝 타이머 현재값이 100으로 변경됩니다.



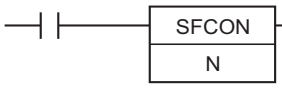
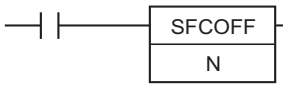
주: '●'는 스텝의 활성 상태를 나타냅니다.

사용상의 주의 사항

- 스텝 타이머는 '[스텝명].X'로 지정합니다. 자세한 내용은 'CX-Programmer 조작 설명서 SFC편'을 참조하십시오.
- 이 명령은 프로그래밍 콘솔에서 입력 및 표시할 수 없습니다. 대신 '?'가 표시됩니다.

SFCON/SFCOFF

명령 이름	니모닉	동작 옵션	평션 번호	기능
SFC 태스크 실행 명령	SFCON	−	789	SFC 태스크 제어 명령에 의해 종료 또는 일시 정지된 SFC 태스크의 실행을 재개합니다.
SFC 태스크 종료 명령	SFCOFF	−	790	SFC 프로그램의 실행 상태를 종료시킵니다. 이때, 출력은 모두 유지된 상태에서 종료합니다. 종료한 SFC 태스크의 실행을 시작하면 초기화 스텝부터 처리가 시작됩니다.

심볼	SFCON	SFCOFF
		

특정 영역 내의 사용 여부

영역	평션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	SFC 태스크 번호	−	1

사이클릭 태스크의 경우, N: &0~31(사이클릭 태스크 번호 0~31에 대응)
(CJ 시리즈 CJ2 CPU 유니트에서는 N: &0~127)
추가 태스크의 경우, N: &8000~8255(인터럽트 태스크 번호 0~255에 대응)

피연산자	태스크 이름
&0~&127	사이클릭 태스크 0~127
&8000	추가 태스크 0(인터럽트 태스크 0)
&8001	추가 태스크 1(인터럽트 태스크 1)
…	….
&8255	추가 태스크 255(인터럽트 태스크 255)

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	· 지정된 태스크가 존재하지 않을 때 OFF · 인터럽트 태스크 내에서 실행되었을 때 OFF

기능

■ SFCON

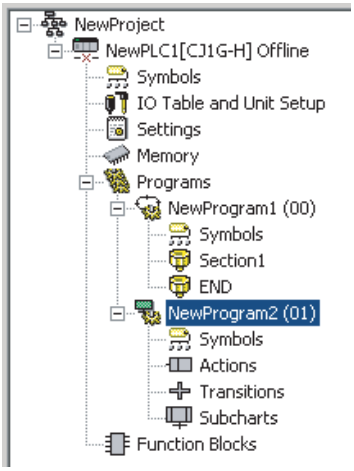
N으로 지정한 SFC 태스크를 기동합니다.
종료 상태의 SFC 태스크를 기동한 경우는 초기 스텝부터 실행을 시작합니다.
일시 정지 상태의 SFC 태스크를 기동한 경우는 정지 전의 상태에서 실행을 재개합니다.

■ SFCOFF

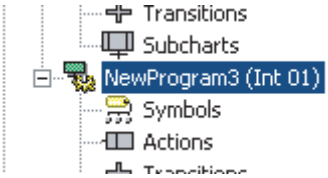
N으로 지정한 SFC 태스크를 종료합니다.
이 경우, 출력은 모두 유지됩니다.
SFCON 명령으로 실행을 재개한 경우, 초기 스텝부터 실행을 시작합니다.

사용상의 주의 사항

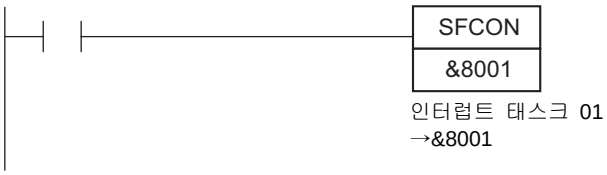
- SFC의 태스크 번호는 다음 그림과 같이 프로젝트 작업 영역의 SFC 프로그램 오른쪽에 표시되는 번호를 지정하십시오.



또한 다음 그림과 같이 인터럽트 태스크에 SFC 프로그램을 작성하고 추가 태스크로 사용할 수도 있습니다.



이 경우의 SFC 태스크 번호의 지정은 다음과 같습니다.

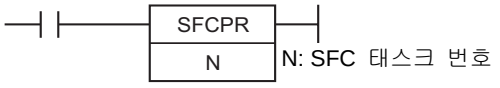
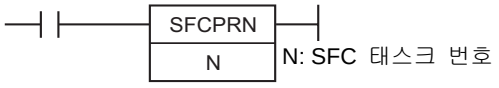


인터럽트 태스크 01
→&8001

- 자체 SFC 태스크에 대해서는 이 명령을 사용할 수 없습니다. 예를 들어, 사이클릭 태스크 0(태스크 번호 0)의 프로그램에서 이 명령이 태스크 번호 0을 지정할 수 없습니다.
- 이 명령은 프로그래밍 콘솔에서 입력 및 표시할 수 없습니다. 대신 '?'가 표시됩니다.

SFCPR/SFCPRN

명령 이름	니모닉	동작 옵션	펑션 번호	기능
SFC 태스크 정지 명령(리셋 있음)	SFCPR	−	793	N으로 지정한 SFC 태스크를 일시 정지합니다. 이때, 정지한 SFC 태스크의 출력은 모두 리셋됩니다.
SFC 태스크 정지 명령(리셋 없음)	SFCPRN	−	791	N으로 지정한 SFC 태스크를 일시 정지합니다. 이때, 정지한 SFC 태스크의 출력은 모두 유지됩니다.

심볼	SFCPR	SFCPRN
		

특정 영역 내의 사용 여부

영역	펑션 블록 정의 내	블록 프로그램 영역	공정 진행 프로그램 영역	서브루틴 프로그램 영역	인터럽트 태스크 프로그램 영역	SFC 작업/ 트랜지션 영역
사용	가능	가능	가능	가능	불가능	가능

피연산자 설명

피연산자	내용	데이터 형식	크기
N	SFC 태스크 번호	−	1

사이클릭 태스크의 경우, N: &0~31(사이클릭 태스크 번호 0~31에 대응)
(CJ 시리즈 CJ2 CPU 유니트에서는 N: &0~127)
추가 태스크의 경우, N: &8000~8255(인터럽트 태스크 번호 0~255에 대응)

피연산자	태스크 이름
&0~&127	사이클릭 태스크 0~127
&8000	추가 태스크 0(인터럽트 태스크 0)
&8001	추가 태스크 1(인터럽트 태스크 1)
…	….
&8255	추가 태스크 255(인터럽트 태스크 255)

조건 플래그의 동작

이름	라벨	내용
에러 플래그	P_ER	・지정된 태스크가 존재하지 않을 때 OFF ・인터럽트 태스크 내에서 실행되었을 때 OFF

기능

■ SFCPR

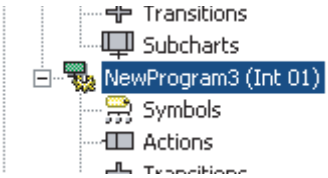
N으로 지정한 SFC 태스크를 일시 정지합니다.
이때, 정지한 SFC 태스크의 출력은 모두 리셋됩니다.
SFCON 명령으로 실행을 재개한 경우, 이전에 정지했을 때의 스텝 활성 상태에서 실행을 시작합니다.

■ SFCPRN

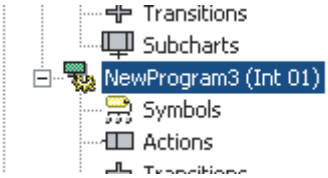
N으로 지정한 SFC 태스크를 일시 정지합니다.
이때, 정지한 SFC 태스크의 출력은 모두 유지됩니다.
SFCON 명령으로 실행을 재개한 경우, 이전에 정지했을 때의 스텝 활성 상태에서 실행을 시작합니다.

사용상의 주의 사항

- SFC의 태스크 번호는 다음 그림과 같이 프로젝트 작업 영역의 SFC 프로그램 오른쪽에 표시되는 번호를 지정하십시오.



또한 다음 그림과 같이 인터럽트 태스크에 SFC 프로그램을 작성하고 추가 태스크로 사용할 수도 있습니다.



이 경우의 SFC 태스크 번호의 지정은 다음과 같습니다.



- 자체 SFC 태스크에 대해 이 명령을 사용할 수 없습니다. 예를 들어, 사이클릭 태스크 0(태스크 번호 0)의 프로그램에서 이 명령이 태스크 번호 0을 지정할 수 없습니다.
- 이 명령은 프로그래밍 콘솔에서 입력 및 표시할 수 없습니다. 대신 '?'가 표시됩니다.
- SFCPR/SFCPRN에 의한 일시 정지인 경우에도 내부적으로 스텝 타이머가 갱신됩니다.
  단, 정지 중에는 표시가 갱신되지 않으므로 주의하십시오.
- SFCPR/SFCPRN을 사용할 때는 스텝 타이머가 정상적으로 동작하지 않는 경우가 있으므로 주의하십시오.

■ SFCPR

- SFCPR 명령은 정지 중인 SFC 태스크에 대해 실행하지 마십시오.(*1)
  정지 중인 SFC 태스크에 대해 SFCPR 명령을 실행한 경우, 해당 SFC 태스크에 대해 SFCON 명령을 실행해도 출력이 리셋되고 태스크도 기동되지 않습니다.
  이런 경우, 같은 태스크에 대해 SFCON 명령을 다시 한번 실행하면 SFC 태스크가 기동됩니다.

*1: 정지 중인 SFC 태스크에 대해 SFCPR 명령을 실행하지 않도록 하려면 태스크 플래그를 사용하여 태스크가 실행 중인지 여부를 확인하십시오. (단, 추가 태스크에서 SFC 프로그램을 사용하는 경우는 태스크 플래그를 사용할 수 없습니다.)

- 스텝이 상태 이동된 동일한 사이클에서 SFCPR이 실행된 경우, SFCPR에 의해 출력이 리셋되므로 비활성 상태가 된 스텝의 P0 작업이나 최종 스캔은 실행되지 않습니다.
- 스텝이 상태 이동된 동일한 사이클에서 SFCPR이 실행된 경우, 그 후의 SFCON에서 SFC 태스크의 실행을 시작해도 활성화된 스텝의 P, P1 작업은 실행되지 않습니다. 자세한 내용은 'CX-Programmer 조작 설명서 SFC편'을 참조하십시오.

제4장

명령 처리 시간/스텝 수 목록

4-1 CJ 시리즈(CJ2) 명령 처리 시간/스텝 수 목록

각 명령어의 처리 시간은 다음과 같습니다.
하나의 사용자 프로그램 전체(주)에서 각 명령어의 처리 시간을 합한 시간이 사이클 시간 중 '연산 처리 시간'입니다.
주: 정확하게는 사이클릭 태스크 중 실행 가능한 상태의 태스크와 인터럽트 조건이 성립하는 인터럽트 태스크에 할당된 프로그램입니다.

명령어의 처리 시간은 CPU 유니트의 형식에 따라 다르며 명령 실행 시의 조건에 따라서도 달라집니다.
또한 명령을 실행하지 않을 때도 처리 시간이 걸리므로 주의하십시오.
동시에 명령 길이(스텝)를 나타내는데, 이는 피연산자를 포함한 각 명령어가 사용자 프로그램 영역을 얼마나 점유하는지 나타냅니다. 스텝 수는 명령어 또는 표현 형식에 따라 달라집니다.
또한 스텝이 명령어의 수를 나타내는 것은 아니므로 주의하십시오.

주 1: 일부 명령어를 제외하고 미분 실행(기본 명령은 '↑ ↓', 응용 명령의 경우는 '@%') 옵션이 지원됩니다. 그 경우의 처리 시간에 다음 시간을 가산하십시오.
(단위: μ 초)

조건	CJ2H CPU 유니트	CJ2M CPU 유니트
	CPU6□(-EIP)	CPU□□
미분(↑ ↓)을 지원하는 경우	+0.24	+0.32
미분(@%)을 지원하는 경우	+0.24	+0.32

주 2: 명령 비실행 시의 처리 시간의 기준은 다음과 같습니다.

CJ2H CPU 유니트	CJ2M CPU 유니트
CPU6□(-EIP)	CPU□□
0.016~0.096	0.020~0.120

4-1-1 시퀀스 입력 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
로드	LD	1~2	0.016	0.040	
	!LD	2~14	0.99	1.260	
로드 부정	LD NOT	1~2	0.016	0.040	
	!LD NOT	2~14	0.99	1.260	
논리곱	AND	1~2	0.016	0.040	
	!AND	2~14	0.99	1.260	
부정 논리곱	AND NOT	1~2	0.016	0.040	
	!AND NOT	2~14	0.99	1.260	
논리합	OR	1~2	0.016	0.040	
	!OR	2~14	0.99	1.260	
부정 논리합	OR NOT	1~2	0.016	0.040	
	!OR NOT	2~14	0.99	1.260	
로드 논리곱	AND LD	1	0.016	0.040	
로드 논리합	OR LD	1	0.016	0.040	
부정	NOT	1	0.016	0.040	
P.F. 기동 미분	UP	3	0.26	0.36	
P.F. 종료 미분	DOWN	4	0.27	0.40	
LD 비트 테스트	LD TST	4	0.11	0.16	
LD 비트 테스트 부정	LD TSTN	4	0.11	0.16	
AND 비트 테스트	AND TST	4	0.11	0.16	
AND 비트 테스트 부정	AND TSTN	4	0.11	0.16	
OR 비트 테스트	OR TST	4	0.11	0.16	
OR 비트 테스트 부정	OR TSTN	4	0.11	0.16	

4-1-2 시퀀스 출력 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
출력	OUT	1~2	0.016	0.040	
	!OUT	2~14	0.99	1.320	
부정 출력	OUT NOT	1~2	0.016	0.040	
	!OUT NOT	2~14	0.99	1.320	
유지	KEEP	1~2	0.048	0.060	
	!KEEP	16	0.99	1.340	
기동 미분	DIFU	2~2	0.28	0.30	
종료 미분	DIFD	2~2	0.24	0.30	
세트	SET	1~2	0.016	0.040	
	!SET	2~14	0.99	1.360	
리셋	RSET	1~2	0.016	0.040	
	!RSET	2~14	0.99	1.360	
다중 비트 세트	SETA	4	3.68	4.12	1비트 세트
			15.5	24.4	1000비트 세트
다중 비트 리셋	RSTA	4	3.7	4.1	1비트 리셋
			15.5	24.4	1000비트 리셋
1비트 세트	SETB	2	0.19	0.280	
	!SETB	16	0.99	1.120	
1비트 리셋	RSTB	2	0.19	0.280	
	!RSTB	16	0.99	1.120	
1비트 출력	OUTB	2	0.19	0.280	
	!OUTB	16	0.99	1.180	

4-1-3 시퀀스 제어 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
종료	END	1	2.6	3.5	
기능 없음	NOP	1	0.016	0.040	
인터록	IL	1	0.048	0.060	
인터록 클리어	ILC	1	0.048	0.060	
멀티 인터록 (미분 플래그 유지형)	MILH	3	2.3	3.3	인터록 조건이 성립하지 않을 때(입력 조건 ON)
			3.4	4.6	인터록 조건이 성립할 때(입력 조건 OFF)
			3.8	5.2	인터록 중에 다시 인터록 조건이 성립할 때(입력 조건 OFF)
멀티 인터록 (미분 플래그 비유지형)	MILR	3	2.3	3.1	인터록 조건이 성립하지 않을 때(입력 조건 ON)
			3.4	4.5	인터록 조건이 성립할 때(입력 조건 OFF)
			3.8	5.1	인터록 중에 다시 인터록 조건이 성립할 때(입력 조건 OFF)
멀티 인터록 클리어	MILC	2	1.2	1.7	인터록 중이 아닐 때
			1.6	2.2	인터록 중일 때
점프	JMP	2	0.31	0.34	
점프 종료	JME	2	-	-	
조건부 점프	CJP	2	0.31	0.34	점프 조건이 성립할 때(입력 조건 ON)
조건부 부정 점프	CJPN	2	0.31	0.34	점프 조건이 성립할 때(입력 조건 OFF)
복수 점프	JMP0	1	0.048	0.060	
복수 점프 종료	JME0	1	0.048	0.060	
반복 시작	FOR	2	0.27	0.42	상수 지정
루프 중단	BREAK	1	0.048	0.060	
반복 종료	NEXT	1	0.14	0.16	루프를 계속할 때
			0.18	0.18	루프를 종료할 때

4-1-4 타이머/카운터 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
100ms 타이머	TIM	3	0.67	0.84	
	TIMX		0.67	0.84	
10ms 타이머	TIMH	3	0.67	0.84	
	TIMHX		0.67	0.84	
1ms 타이머	TMHH	3	0.67	0.84	
	TMHHX		0.67	0.84	
0.1ms 타이머	TIMU	3	0.67	0.84	
	TIMUX		0.67	0.84	
0.01ms 타이머	TMUH	3	0.67	0.84	
	TMUHX		0.67	0.84	
적산 타이머	TTIM	3	9.2	12.1	
			6.9	8.4	리셋할 때
			5.0	6.5	인터록할 때
	TTIMX	3	8.8	11.7	
			6.8	8.5	리셋할 때
			5.0	6.5	인터록할 때
장시간 타이머	TIML	4~5	5.8	7.0	
			3.9	4.1	인터록할 때
	TIMLX	4~5	5.7	7.0	
			3.6	3.7	인터록할 때
멀티 출력 타이머	MTIM	4	6.4	7.2	
			3.7	4.3	리셋할 때
	MTIMX	4	5.5	6.4	
			3.4	3.8	리셋할 때
타이머 리셋	TRSET	2	0.58	0.8	
카운터	CNT	3	0.51	0.58	
	CNTX		0.51	0.58	
가역 카운터	CNTR	3	9.1	11.8	
	CNTRX		8.0	10.3	
타이머/카운터 리셋	CNR	3	4.8	5.4	1CH 리셋
			2839	2555	1000CH 리셋
	CNRX	3	4.7	5.5	1CH 리셋
			2839	2555	1000CH 리셋

4-1-5 데이터 비교 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
심볼 비교(부호 없음)	=	4	0.08	0.16	
	<>				
	<				
	<=				
	>				
	>=				
심볼 비교(2워드, 부호 없음)	=L	4~7	0.08	0.24	
	<>L				
	<L				
	<=L				
	>L				
	>=L				
심볼 비교(부호 있음)	=S	4	0.08	0.16	
	<>S				
	<S				
	<=				
	>S				
	>=S				
심볼 비교(2워드, 부호 있음)	=SL	4~7	0.08	0.24	
	<>SL				
	<SL				
	<=SL				
	>SL				
	>=SL				
시각 비교	=DT	4	16.300	27.9	
	<>DT				
	<DT				
	<=DT				
	>DT				
	>=DT				
비교	CMP	3	0.06	0.080	
	!CMP	30	2.06	2.6	
2워드 비교	CMPL	3~5	0.064	0.120	
부호 있는 BIN 비교	CPS	3~5	0.064	0.080	
	!CPS	30	2.06	2.6	
부호 있는 BIN 2워드 비교	CPSL	3~5	0.064	0.120	
테이블 일치	TCMP	4	10.3	12.5	
다채널 비교	MCMP	4	15.2	20.3	
부호 없는 테이블 간 비교	BCMP	4	16.3	20.5	
확장 테이블 간 비교	BCMP2	4	5.0	5.1	데이터 수 1
			217.2	278	데이터 수 255
영역 비교	ZCP	3	0.14	0.400	
2워드 영역 비교	ZCPL	3~5	0.14	0.640	
부호 있는 영역 비교	ZCPS	3	0.14	0.400	
부호 있는 2워드 영역 비교	ZCPSL	3~5	0.14	0.640	

4-1-6 데이터 전송 명령

24.8 mm 명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
전송	MOV	3	0.05	0.12	
	!MOV	30	1.98	2.6	
2워드 전송	MOVL	3~4	0.05	0.20	
부정 전송	MVN	3	0.05	0.12	
부정 2워드 전송	MVNL	3~4	0.05	0.20	
비트 전송	MOVB	4	0.19	0.32	
자릿수 전송	MOVD	4	0.19	0.32	
다중 비트 전송	XFRB	4	6.6	9.4	1비트 전송
			85.8	119	255비트 전송
블록 전송	XFER	4	0.29	0.28	1CH 전송
			240.1	220	1000CH 전송
블록 설정	BSET	4	0.21	0.20	1CH 세트
			142.2	140	1000CH 세트
데이터 교환	XCHG	3	0.32	0.48	
데이터 2워드 교환	XCGL	3~4	0.12	0.29	
데이터 분배	DIST	4	4.5	4.7	
데이터 추출	COLL	4	4.6	4.7	
인덱스 레지스터 설정	MOVR	3	0.064	0.200	
인덱스 레지스터 설정	MOVRW	3	0.064	0.200	

4-1-7 데이터 이동 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
이동 레지스터	SFT	3	2.86	3.47	1CH 이동
			315	422	1000CH 이동
좌우 이동 레지스터	SFTR	4	6.22	6.38	1CH 이동
			319	422	1000CH 이동
비동기 이동 레지스터	ASFT	4	5.3	6.3	1CH 이동
			948	1285	1000CH 이동*1
워드 이동	WSFT	4	2.3	3.1	1CH 이동
			233	187	1000CH 이동
1비트 좌측 이동	ASL	2	0.18	0.260	
1비트 좌측 이동(2워드)	ASLL	2	0.32	0.420	
1비트 우측 이동	ASR	2	0.18	0.260	
1비트 우측 이동(2워드)	ASRL	2	0.32	0.420	
CY가 있는 1비트 좌측 회전	ROL	2	0.18	0.260	
CY가 있는 1비트 좌측 회전(2워드)	ROLL	2	0.32	0.420	
CY가 없는 1비트 좌측 회전	RLNC	2	0.18	0.260	
CY가 없는 1비트 좌측 회전(2워드)	RLNL	2	0.32	0.420	
CY가 있는 1비트 우측 회전	ROR	2	0.18	0.260	
CY가 있는 1비트 우측 회전(2워드)	RORL	2	0.32	0.420	
CY가 없는 1비트 우측 회전	RRNC	2	0.18	0.260	
CY가 없는 1비트 우측 회전(2워드)	RRNL	2	0.32	0.420	
1자리 좌측 이동	SLD	3	3.7	4.4	1CH 이동
			317.8	429	1000CH 이동
1자리 우측 이동	SRD	3	4.5	5.4	1CH 이동
			479.5	656	1000CH 이동
N비트 데이터 좌측 이동	NSFL	4	4.6	5.2	1비트 이동
			31.5	36.1	1000비트 이동
N비트 데이터 우측 이동	NSFR	4	4.5	5.2	1비트 이동
			39.0	50.2	1000비트 이동
N비트 좌측 이동	NASL	3	0.18	0.38	
N비트 2워드 좌측 이동	NSLL	3	0.32	0.54	
N비트 우측 이동	NASR	3	0.18	0.38	
N비트 2워드 우측 이동	NSRL	3	0.32	0.54	

*1: 데이터량에 따라 명령 실행 시간이 크게 변화하며 사이클 시간에 영향을 줍니다. 사이클 시간에 대한 영향을 줄이기 위해 '백그라운드 실행'을 수행할 수 있습니다.

4-1-8 증가/감소 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
BIN 증가	++	2	0.18	0.24	
BIN 2워드 증가	++L	2	0.18	0.24	
BIN 감소	-	2	0.18	0.24	
BIN 2워드 감소	--L	2	0.18	0.24	
BCD 증가	++B	2	3.0	3.4	
BCD 2워드 증가	++BL	2	3.2	3.5	
BCD 감소	--B	2	3.0	3.5	
BCD 2워드 감소	--BL	2	3.2	3.5	

4-1-9 사칙연산 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
부호 있음, CY 없는 BIN 가산	+	4	0.18	0.34	
부호 있음, CY 없는 BIN 2워드 가산	+L	4~6	0.18	0.24	
부호 및 CY 있는 BIN 가산	+C	4	0.18	0.34	
부호 및 CY 있는 BIN 2워드 가산	+CL	4~6	0.18	0.24	
CY 없는 BCD 가산	+B	4	4.0	4.8	
CY 없는 BCD 2워드 가산	+BL	4~6	4.9	6.0	
CY 있는 BCD 가산	+BC	4	4.4	5.2	
CY 있는 BCD 2워드 가산	+BCL	4~6	5.2	6.6	
부호 있음, CY 없는 BIN 감산	-	4	0.18	0.340	
부호 있음, CY 없는 BIN 2워드 감산	-L	4~6	0.18	0.24	
부호 및 CY 있는 BIN 감산	-C	4	0.18	0.340	
부호 및 CY 있는 BIN 2워드 감산	-CL	4~6	0.18	0.24	
CY 없는 BCD 감산	-B	4	4.1	4.9	
CY 없는 BCD 2워드 감산	-BL	4~6	4.9	5.9	
CY 있는 BCD 감산	-BC	4	4.5	5.2	
CY 있는 BCD 2워드 감산	-BCL	4~6	5.2	6.3	
부호 있는 BIN 승산	*	4	0.26	0.520	
부호 있는 BIN 2워드 승산	*L	4~6	3.6	3.9	
부호 없는 BIN 승산	*U	4	0.26	0.26	
부호 없는 BIN 2워드 승산	*UL	4~6	3.6	3.9	
BCD 승산	*B	4	3.6	4.6	
BCD 2워드 승산	*BL	4~6	4.9	6.2	
부호 있는 BIN 제산	/	4	0.29	0.540	
부호 있는 BIN 2워드 제산	/L	4~6	4.2	4.8	
부호 없는 BIN 제산	/U	4	0.29	0.540	
부호 없는 BIN 2워드 제산	/UL	4~6	3.8	4.2	
BCD 제산	/B	4	5.0	5.9	
BCD 2워드 제산	/BL	4~6	4.8	5.9	

4-1-10 데이터 변환 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
BCD→BIN 변환	BIN	3	0.18	0.280	
BCD→BIN 2워드 변환	BINL	3~4	3.3	3.5	
BIN→BCD 변환	BCD	3	0.19	0.300	
BIN→BCD 2워드 변환	BCDL	3~4	3.3	3.7	
2의 보수 변환	NEG	3	0.14	0.240	
2의 보수 2워드 변환	NEGL	3~4	0.26	0.440	
부호 확장	SIGN	3	0.26	0.340	
4→16/8→256 디코더	MLPX	4	0.17	0.280	4→16 디코드 시/자릿수 1자리
			0.42	0.770	4→16 디코드 시/자릿수 4자리
			1.14	1.760	8→256 디코드 시/자릿수 1자리
			2.17	3.370	8→256 디코드 시/자릿수 4자리
16→4/256→8 인코더	DMPX	4	3.3	4.6	16→4 디코드 시/자릿수 1자리
			3.7	5.2	16→4 디코드 시/자릿수 4자리
			17.3	26.3	256→8 디코드 시/자릿수 1자리
			35	47	256→8 디코드 시/자릿수 2자리
ASCII 코드 변환	ASC	4	4.0	4.5	1자리 ASCII 코드 변환
			4.6	5.2	4자리 ASCII 코드 변환
ASCII→HEX 변환	HEX	4	3.3	3.8	1자리 변환
비트 열→비트 행 변환	LINE	4	10.5	13.1	
비트 행→비트 열 변환	COLM	4	13.8	17.6	
부호 있는 BCD→BIN 변환	BINS	4	3.6	4.0	데이터 형식 지정 0
			3.6	4.0	데이터 형식 지정 1
			3.6	4.0	데이터 형식 지정 2
			3.6	4.0	데이터 형식 지정 3
부호 있는 BCD→BIN 2워드 변환	BISL	4~5	3.7	4.1	데이터 형식 지정 0
			3.6	4.1	데이터 형식 지정 1
			3.7	4.2	데이터 형식 지정 2
			3.7	4.2	데이터 형식 지정 3
부호 있는 BIN→BCD 변환	BCDS	4	3.7	4.0	데이터 형식 지정 0
			3.7	4.1	데이터 형식 지정 1
			3.7	4.2	데이터 형식 지정 2
			3.7	4.2	데이터 형식 지정 3
부호 있는 BIN→BCD 2워드 변환	BDSL	4~5	4.0	4.5	데이터 형식 지정 0
			4.0	4.6	데이터 형식 지정 1
			4.0	4.6	데이터 형식 지정 2
			4.1	4.6	데이터 형식 지정 3
그레이 코드 변환	GRY	4	26.5	49.1	8비트 BIN
			27.6	51.1	8비트 BCD
			30.9	57.2	8비트 각도
			35.3	66.0	15비트 BIN
			36.3	68.0	15비트 BCD
			39.6	74.0	15비트 각도
그레이 코드→BIN 변환	GRAY_BIN	3	0.1	0.3	
그레이 코드→BIN 2워드 변환	GRAY_BINL	3~4	0.1	0.4	
BIN→그레이 코드 변환	BIN_GRAY	3	0.1	0.3	
BIN→그레이 코드 2워드 변환	BIN_GRAYL	3~4	0.1	0.4	
숫자 4자리→ASCII 변환	STR4	3	8.4	14.2	
숫자 8자리→ASCII 변환	STR8	3~4	10.2	16.4	
숫자 16자리→ASCII 변환	STR16	3	15.8	28.2	
ASCII→숫자 4자리 변환	NUM4	3~4	10.5	18.5	
ASCII→숫자 8자리 변환	NUM8	3	14.8	27.1	
ASCII→숫자 16자리 변환	NUM16	3	27.4	52.0	

4-1-11 논리 연산 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
워드 논리곱	ANDW	4	0.14	0.340	
워드 논리곱(2워드)	ANDL	4~6	0.26	0.640	
워드 논리합	ORW	4	0.18	0.340	
워드 논리합(2워드)	ORWL	4~6	0.26	0.640	
워드 배타적 논리합	XORW	4	0.18	0.340	
워드 배타적 논리합(2워드)	XORL	4~6	0.26	0.640	
워드 배타적 논리합 부정	XNRW	4	0.18	0.340	
워드 배타적 논리합 부정(2워드)	XNRL	4~6	0.26	0.640	
비트 반전	COM	2	0.18	0.240	
비트 2워드 반전	COML	2	0.32	0.440	

4-1-12 특수 연산 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
BIN 제곱근 연산	ROTB	3	15.4	24.2	
BCD 제곱근 연산	ROOT	3	17.1	25.3	
숫자 변환	APR	4	4.6	5.3	SIN, COS 계산 지정
			5.7	6.9	꺾은선 근사 계산 지정
부동 소수점 제산(BCD)	FDIV	4	76	149	
비트 카운터	BCNT	4	0.24	0.360	1CH 카운트

4-1-13 부동 소수점 변환 및 연산 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
부동 소수점→16비트 BIN 변환	FIX	3~4	0.13	0.24	
부동 소수점→32비트 BIN 변환	FIXL	3~4	0.13	0.32	
16비트 BIN→부동 소수점 변환	FLT	3~4	0.13	0.30	
32비트 BIN→부동 소수점 변환	FLTL	3~4	0.13	0.32	
부동 소수점 가산	+F	4~6	0.24	0.66	
부동 소수점 감산	-F	4~6	0.24	0.66	
부동 소수점 제산	/F	4~6	0.4	0.9	
부동 소수점 승산	*F	4~6	0.24	0.66	
각도→라디안 변환	RAD	3~4	2.7	3.3	
라디안→각도 변환	DEG	3~4	3.0	3.2	
SIN 연산	SIN	3~4	3.8	4.3	0도 지정 시
			4.5	5.4	45도 지정 시
			5.0	6.0	90도 지정 시
고속 SIN 연산	SINQ	8~9	0.59	0.86	0, 45, 90도 지정 시
COS 연산	COS	3~4	3.7	4.3	0도 지정 시
			4.4	5.2	45도 지정 시
			5.3	6.7	90도 지정 시
고속 COS 연산	COSQ	8~9	0.59	0.86	0, 45, 90도 지정 시
TAN 연산	TAN	3~4	3.9	4.5	0도 지정 시
			6.1	8.2	45도 지정 시
고속 TAN 연산	TANQ	15~16	1.2	1.7	0, 45, 90도 지정 시
SIN^{-1} 연산	ASIN	3~4	5.8	7.1	0도 지정 시
			24.8	33.0	45도 지정 시
			5.6	7.0	90도 지정 시
COS^{-1} 연산	ACOS	3~4	5.3	6.8	0도 지정 시
			27.2	34.6	45도 지정 시
			6.4	7.1	90도 지정 시
TAN^{-1} 연산	ATAN	3~4	4.0	5.0	0도 지정 시
			5.6	7.0	45도 지정 시
제곱근 연산	SQRT	3~4	0.42	0.66	
지수 연산	EXP	3~4	3.8	4.5	
로그 연산	LOG	3~4	5.8	6.5	
거듭제곱 연산	PWR	4~6	35.7	56.6	
단정도 부동 소수점 데이터 비교	=F	4~6	0.13	0.26	
	<>F				
	<F				
	<=F				
	>F				
	>=F				
부동 소수점<단>→문자열 변환	FSTR	4~5	15.6	23.9	
문자열→부동 소수점<단> 변환	FVAL	3	21.2	31.4	
부동 소수점 전송<단>	MOVF	3~4	0.18	0.20	

4-1-14 배정도 부동 소수점 변환 및 연산 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
배정도 부동 소수점 데이터 비교	=D	4	5.1	6.7	
	<>D				
	<D				
	<=D				
	>D				
	>=D				
부동 소수점→16비트 BIN 변환<배>	FIXD	3	5.1	5.4	
부동 소수점→32비트 BIN 변환<배>	FIXLD	3	5.1	5.4	
16비트 BIN→부동 소수점 변환<배>	DBL	3	3.5	4.3	
32비트 BIN→부동 소수점 변환<배>	DBLL	3	3.5	4.3	
부동 소수점 가산<배>	+D	4	6.0	7.1	
부동 소수점 감산<배>	-D	4	6.1	7.1	
부동 소수점 승산<배>	*D	4	6.1	7.1	
부동 소수점 제산<배>	/D	4	6.4	7.5	
각도→라디안 변환<배>	RADD	3	6.1	6.5	
라디안→각도 변환<배>	DEGD	3	6.0	6.4	
SIN 연산<배>	SIND	3	14.7	21.5	0도 지정 시
			20.4	35.4	45도 지정 시
			18.5	35.0	90도 지정 시
COS 연산<배>	COSD	3	14.1	20.6	0도 지정 시
			19.6	29.9	45도 지정 시
			19.1	29.8	90도 지정 시
TAN 연산<배>	TAND	3	7.3	9.4	0도 지정 시
			27.4	50.3	45도 지정 시
SIN^{-1} 연산<배>	ASIND	3	7.5	9.8	0도 지정 시
			55.0	75.2	45도 지정 시
			6.1	8.3	90도 지정 시
COS^{-1} 연산<배>	ACOSD	3	8.3	10.9	0도 지정 시
			55.9	72.8	45도 지정 시
			43.7	72.8	90도 지정 시
TAN^{-1} 연산<배>	ATAND	3	6.1	7.4	0도 지정 시
			29.7	36.5	45도 지정 시
제곱근 연산<배>	SQRTD	3	16.6	23.4	
지수 연산<배>	EXPD	3	39.7	58.4	
로그 연산<배>	LOGD	3	35.5	52.2	
거듭제곱 연산<배>	PWRD	4	66	99	

4-1-15 테이블 데이터 처리 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
스택 영역 설정	SSET	3	7.6	9.4	스택 영역에 5CH 설정
			107	65	스택 영역에 1000CH 설정
스택 데이터 저장	PUSH	3	4.9	5.9	
선입선출	FIFO	3	4.8	5.0	스택 영역에 5CH 설정
			231	167	스택 영역에 1000CH 설정
후입선출	LIFO	3	5.3	7.1	
테이블 영역 선언	DIM	5	11.1	19.7	
레코드 위치 설정	SETR	4	3.8	5.5	
레코드 위치 읽기	GETR	4	4.6	7.9	
데이터 검색	SRCH	4	13.9	25.0	1CH 검색
			1940	3257	1000CH 검색*1
바이트 스와핑	SWAP	3	10.1	17.5	1CH 교체
			1421	2098	1000CH 교체*1
최대값 검색	MAX	4~5	4.8	5.8	데이터 수 1
			465	672	데이터 수 1000 *1
2워드 데이터 최대값 검색	MAXL	4~5	4.8	5.8	데이터 수 1
			465	773	데이터 수 1000 *1
단정도 부동 소수점 데이터 최대값 검색	MAXF	4~5	5.2	6.5	데이터 수 1
			682	1090	데이터 수 1000 *1
배정도 부동 소수점 데이터 최대값 검색	MAXD	4~5	5.4	6.4	데이터 수 1
			1435	2333	데이터 수 1000 *1
최소값 검색	MIN	4~5	4.8	5.8	데이터 수 1
			465	677	데이터 수 1000 *1
2워드 데이터 최소값 검색	MINL	4~5	4.8	5.9	데이터 수 1
			189	774	데이터 수 1000 *1
단정도 부동 소수점 데이터 최소값 검색	MINF	4~5	5.2	6.5	데이터 수 1
			683	1091	데이터 수 1000 *1
배정도 부동 소수점 데이터 최소값 검색	MIND	4~5	5.2	6.4	데이터 수 1
			1402	2303	데이터 수 1000 *1
SUM 값 산출	SUM	4~5	17.5	31.3	1CH 산출
			900	1696	1000CH 산출*1
FCS 값 산출	FCS	4~5	14.1	25.2	테이블 길이 1
			1235	2089	테이블 길이 1000 *1
스택 데이터 수 출력	SNUM	3	4.5	5.3	
스택 데이터 참조	SREAD	4	4.6	5.4	
스택 데이터 갱신	SWRIT	4	4.3	5.0	
스택 데이터 삽입	SINS	4	8.2	9.3	
			275	256	테이블 길이 1000
스택 데이터 삭제	SDEL	4	6.1	7.8	
			247	180	테이블 길이 1000

*1: 데이터량에 따라 명령 실행 시간이 크게 변화하며 사이클 시간에 영향을 줍니다. 사이클 시간에 대한 영향을 줄이기 위해 '백그라운드 실행'을 수행할 수 있습니다.

4-1-16 추적 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
부호 없는 1W 레코드 검색	RSRCH	6	13.9	15.9	레코드 수 1
			504	585	레코드 수 1000
부호 없는 2W 레코드 검색	RSRCH2	6	14.7	17.6	레코드 수 1
			838	932	레코드 수 1000
부호 없는 4W 레코드 검색	RSRCH4	6	17.0	19.1	레코드 수 1
			1544	1684	레코드 수 1000
부호 없는 1W 레코드 정렬	RSORT	5	149	156	100레코드
부호 없는 2W 레코드 정렬	RSORT2	5	250	249	분할 정렬 설정 무효
부호 없는 4W 레코드 정렬	RSORT4	5	457	440	99, 98, 97,..,0의 데이터를 0, 1, 2,...,99로 정렬 (최악값)

4-1-17 데이터 제어 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
PID 연산	PID	4	297	526	처음 실행할 때
			234	423	입력 ON이고 샘플링할 때
			71	117	입력 ON이고 샘플링하지 않을 때
			7.2	10.5	입력 OFF 시
자동 조정 가능한 PID 연산	PIDAT	4	302	600	처음 실행할 때
			237	428	입력 ON이고 샘플링할 때
			73	118	입력 ON이고 샘플링하지 않을 때
			7.3	10.5	입력 OFF 시
			120	203	AT에서 처음 실행할 때
상하한 리미트 제어	LMT	4~5	10.8	18.3	
데드 밴드 제어	BAND	4~5	11.2	19.2	
데드 존 제어	ZONE	4~5	10.9	17.7	
시분할 비례 출력	TPO	4	6.9	10.2	입력 OFF
			37	65	입력 ON 및 듀티 지정 또는 출력 리미트 무효
스케일링	SCL	4	7.6	9.3	
스케일링 2	SCL2	4	6.8	9.2	
스케일링 3	SCL3	4	7.8	9.9	
데이터 평균화	AVG	4	22	40	1회 평균
			212	351	64회 평균

4-1-18 서브루틴 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
서브루틴 호출	SBS	2	0.90	2.8	
서브루틴 진입	SBN	2	2.8	4.1	
서브루틴 반환	RET	1	0.43	2.0	
매크로	MCRO	4	16.8	21.7	
글로벌 서브루틴 호출	GSBS	2	0.90	2.8	
글로벌 서브루틴 진입	GSBN	2	2.7	3.6	
글로벌 서브루틴 반환	GRET	1	0.43	2.0	

4-1-19 인터럽트 제어 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
인터럽트 마스크 세트	MSKS	3	10.6	22.1	
인터럽트 마스크 읽기	MSKR	3	9.6	14.8	
인터럽트 해제	CLI	3	10.1	21.6	
인터럽트 태스크 실행 금지	DI	1	10.3	20.4	
인터럽트 태스크 실행 금지 해제	EI	1	9.3	16.0	

4-1-20 공정 진행 제어 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
스텝 래더 영역 정의	STEP	2	8.7	10.6	스텝 릴레이가 ON일 때
			8.7	9.8	스텝 릴레이가 OFF일 때
스텝 래더 영역 진행	SNXT	2	2.2	2.8	

4-1-21 I/O 유니트용 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
I/O 리프레시	IORF	3	10.1	12.2	1CH IN
			10.5	13.0	1CH OUT
고기능 I/O 유니트 매회 리프레시	FIORF	2	*1	*1	
CPU 고기능 유니트 매회 I/O 리프레시	DLNK	4	234	256	1CH 점유
7세그먼트 디코더	SDEC	4	2.5	3.3	
디지털 스위치	DSW	6	24.8	39.6	4자리 지정, 데이터 입력값: 0
			24.8	40.2	8자리 지정, 데이터 입력값: 00
10키 입력	TKY	4	7.2	9.7	데이터 입력값: 00
			6.5	8.6	데이터 입력값: FF
16키 입력	HYK	5	25.9	40.8	데이터 입력값: 00
			25.9	41.0	데이터 입력값: FF
행렬 입력	MTR	5	25.0	38.5	데이터 입력값: 00
			25.0	38.5	데이터 입력값: FF
7세그먼트 표시	7SEG	5	31.4	51.9	4자리 지정 시
			34.6	59.4	8자리 지정 시
아날로그 입력 직접 변환	AIDC	3	25.0	27.0	아날로그 입력 번호: 1, 사용하는 아날로그 입력 점수: 4
			38.8	41.6	아날로그 입력 번호: 0, 사용하는 아날로그 입력 점수: 4
아날로그 출력 직접 변환	AODC	3	23.1	24.4	아날로그 출력 번호: 1, 사용하는 아날로그 출력 점수: 4
			44.1	45.3	아날로그 출력 번호: 0, 사용하는 아날로그 출력 점수: 4
NC 유니트 위치 결정 고속 기동	NCDMV	4	81.7	95.3	
NC 유니트 위치 결정 시작 트리거	NCDTR	3	22.9	25.5	
인텔리전트 I/O 읽기	IORD	4	*1	*1	
인텔리전트 I/O 쓰기	IOWR	4	*1	*1	

* 1： IORD, IORW, FIORF 명령은 읽기 상대의 고기능 I/O 유니트에 따라 다릅니다.

4-1-22 시리얼 통신 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
프로토콜 매크로	PMCR	5	57.8	97.8	직접 지정
			77	132	피연산자 지정 송신 1CH, 수신 1CH
프로토콜 매크로 송신 2	PMCR2	6	49.5	96.0	직접 지정
			69	129	피연산자 지정 송신 1CH, 수신 1CH
시리얼 포트 출력	TXD	4	57.5	93.8	송신 바이트 수 1바이트
			517	947	송신 바이트 수 256바이트
시리얼 포트 입력	RXD	4	79	128	저장 바이트 수 1바이트
			570	1033	저장 바이트 수 256바이트
시리얼 커뮤니케이션 유니트 시리얼 포트 출력	TXDU	4	75	130	송신 바이트 수 1바이트
시리얼 커뮤니케이션 유니트 시리얼 포트 입력	RXDU	4	74	128	저장 바이트 수 1바이트
시리얼 커뮤니케이션 유니트 시리얼 포트 직접 송신	DTXDU	4	25.8	37.0	송신 바이트 수 1바이트
			179	203	송신 바이트 수 256바이트
시리얼 커뮤니케이션 유니트 시리얼 포트 직접 수신	DRXDU	4	27.8	39.7	저장 바이트 수 1바이트
			188	205	저장 바이트 수 256바이트
시리얼 통신 설정 변경	STUP	3	233	276	CPU 유니트의 COM 포트 대상

4-1-23 네트워크 통신용 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
네트워크 송신	SEND	4	44.3	79.4	
네트워크 송신 2	SEND2	5	43.4	82.8	
네트워크 수신	RECV	4	43.9	79.9	
네트워크 수신 2	RECV2	5	44.5	82.8	
명령 송신	CMND	4	52.7	95.1	
명령 송신 2	CMND2	5	53.0	98.1	
범용 Explicit 메시지 송신	EXPLT	4	78	134	
Explicit 읽기	EGATR	4	74	127	
Explicit 쓰기	ESATR	3	69	117	
Explicit CPU 유니트 데이터 읽기	ECHRD	4	65	110	
Explicit CPU 유니트 데이터 쓰기	ECHWR	4	64	110	

4-1-24 파일 메모리용 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
데이터 파일 읽기	FREAD	5	217	372	BIN 형식
데이터 파일 쓰기	FWRIT	5	216	366	BIN 형식
텍스트 파일 쓰기	TWRIT	5	205	370	

4-1-25 표시 기능용 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
메시지 표시	MSG	3	6.9	10.5	메시지 표시
			6.6	9.5	메시지 표시 해제

4-1-26 시계 기능 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
캘린더 가산	CADD	4	15.6	22.5	
캘린더 감산	CSUB	4	16.4	24.9	
시분초→초 변환	SEC	3	3.6	4.1	
초→시분초 변환	HMS	3	3.5	4.0	
시계 보정	DATE	2	29.6	53.2	

4-1-27 디버그 처리 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
트레이스 메모리 샘플링	TRSM	1	8.9	12.6	1점+0CH 샘플링 지정 시
			31.6	33.1	31점+6CH 샘플링 지정 시
			38.8	39.2	31점+16CH 샘플링 지정 시

4-1-28 고장 진단 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
운전 계속 고장 진단	FAL	3	7.9	14.7	이상 등록
			14.7	22.3	이상 해제(최우선)
			12.9	22.5	이상 해제(전체 이상)
			117	210	이상 해제(개별)
운전 정지 고장 진단	FALS	3	-	-	
고장 지점 검출	FPD	4	111	188	접점 주소 출력 시간 감시 중
			107	202	접점 주소 출력 이상을 처음 검출했을 때
			129	242	메시지 문자 출력 시간 감시 중
			159	244	메시지 문자 출력 이상을 처음 검출했을 때

4-1-29 특수 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
세트 캐리	STC	1	0.048	0.060	
클리어 캐리	CLC	1	0.048	0.060	
확장 DM 뱅크 전환	EMBC	2	7.6	14.6	
사이클 시간 감시 시간 설정	WDT	2	7.6	17.1	
조건 플래그 저장	CCS	1	5.8	8.3	
조건 플래그 로드	CCL	1	6.4	9.9	
CV→CS 주소 변환	FRMCV	3	9.4	15.7	
CS→CV 주소 변환	TOCV	3~4	10.3	18.2	
주변 서비스 금지	IOSP	1	-	-	
주변 서비스 금지 해제	IORS	1	-	-	

4-1-30 블록 프로그램 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
블록 프로그램	BPRG	2	7.8	14.1	
블록 프로그램 종료	BEND	1	8.8	13.4	
블록 프로그램 일시 정지	BPPS	2	5.4	8.4	
블록 프로그램 재기동	BPRS	2	3.6	4.8	
조건부 종료	입력 조건 EXIT	1	8.6	13.2	EXIT 실행(입력 조건 ON)
			2.0	2.6	EXIT 실행 안 함(입력 조건 OFF)
	EXIT 릴레이	2	9.8	14.8	EXIT 실행(릴레이 ON)
			3.6	4.2	EXIT 실행 안 함(릴레이 OFF)
조건부 종료(부정)	EXIT NOT 릴레이	2	3.6	4.3	EXIT 실행(릴레이 OFF)
			8.9	14.9	EXIT 실행 안 함(릴레이 ON)
조건 분기 블록	입력 조건 IF	1	1.9	2.4	IF 참(입력 조건 ON)
			3.8	6.4	IF 거짓(입력 조건 OFF)
조건 분기 블록	IF 릴레이	2	3.2	4.0	IF 참(릴레이 ON)
			5.1	8.0	IF 거짓(릴레이 OFF)
조건 분기 블록(부정)	IF NOT 릴레이	2	5.1	8.2	IF 참(릴레이 OFF)
			3.2	4.1	IF 거짓(릴레이 ON)
조건 분기 거짓 블록	ELSE	1	3.5	5.7	IF 참
			5.3	7.3	IF 거짓
조건 분기 블록 종료	IEND	1	5.3	8.5	IF 참
			2.0	2.4	IF 거짓
1스캔 조건 대기	입력 조건 WAIT	1	10.0	15.9	WAIT 실행 안 함(입력 조건 ON)
			1.4	1.9	WAIT 실행(입력 조건 OFF)
1스캔 조건 대기	WAIT 릴레이	2	9.2	13.5	WAIT 실행 안 함(릴레이 ON)
			2.6	3.7	WAIT 실행(릴레이 OFF)
1스캔 조건 대기(부정)	WAIT NOT 릴레이	2	9.2	13.5	WAIT 실행 안 함(릴레이 OFF)
			2.8	3.7	WAIT 실행(릴레이 ON)
100ms 타이머 대기	TIMW	3	15.6	22.9	처음 기동할 때
			16.0	23.2	처음 이후
	TIMWX	3	15.1	21.7	처음 기동할 때
			16.0	23.2	처음 이후
10ms 타이머 대기	TMHW	3	15.7	22.6	처음 기동할 때
			17.5	24.9	처음 이후
	TMHWX	3	15.2	22.1	처음 기동할 때
			16.4	23.4	처음 이후
카운터 대기	CNTW	4	13.7	20.5	처음 기동할 때
			13.4	19.8	처음 이후
	CNTWX	4	13.1	19.5	처음 기동할 때
			13.5	19.7	처음 이후
반복 블록	LOOP	1	4.6	9.1	
반복 블록 종료	입력 조건 LEND	1	4.2	8.6	LOOP 실행 안 함(입력 조건 ON)
			3.9	6.5	LOOP 실행(입력 조건 OFF)
반복 블록 종료	LEND 릴레이 번호	2	6.7	10.4	LOOP 실행 안 함(릴레이 ON)
			6.6	8.2	LOOP 실행(릴레이 OFF)
반복 블록 종료(부정)	LEND NOT 릴레이 번호	2	6.7	10.9	LOOP 실행 안 함(릴레이 OFF)
			6.6	8.2	LOOP 실행 안 함(릴레이 ON)

4-1-31 문자열 처리 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
문자열 전송	MOV$	3	31.5	58.3	1문자 전송*1
문자열 연결	+$	4	56	104	1문자+1문자*1
왼쪽에서 문자열 가져오기	LEFT$	4	33.5	62.2	2문자에서 1문자 가져오기*1
오른쪽에서 문자열 가져오기	RGHT$	4	33.4	62.1	2문자에서 1문자 가져오기*1
임의 위치에서 문자열 가져오기	MID$	5	32.3	60.8	3문자 중에서 1문자 가져오기*1
문자열 검색	FIND$	4	30.3	56.3	2문자에서 1문자 검색*1
문자열 길이 검출	LEN$	3	14.0	24.9	1문자 검출*1
문자열 치환	RPLC$	6	110	213	2문자 중 첫 번째 문자를 1문자 치환*1
문자열 삭제	DEL$	5	45.6	86.8	2문자 중 선두 1문자 삭제*1
문자열 교환	XCHG$	3	40.3	75.4	1문자와 1문자 교환*1
문자열 클리어	CLR$	2	15.9	28.3	1문자 클리어*1
문자열 삽입	INS$	5	85	162	2문자 중 첫 번째 문자 뒤에 1문자 삽입*1
문자열 비교	=$	4	27.0	50.9	1문자와 1문자 비교
	<>$				
	<$				
	<=$				
	>$				
	>=$				

*1: 데이터량에 따라 명령 실행 시간이 크게 변화하며 사이클 시간에 영향을 줍니다. 사이클 시간에 대한 영향을 줄이기 위해 '백그라운드 실행'을 수행할 수 있습니다.

4-1-32 태스크 제어 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
태스크 실행 기동	TKON	2	12.5	19.0	사이클릭 태스크 지정 시
			13.6	22.7	추가 태스크 지정 시
태스크 실행 대기	TKOF	2	240	393	사이클릭 태스크 지정 시
			15.5	25.8	추가 태스크 지정 시

4-1-33 기종 변환용 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
블록 전송	XFERC	4	6.7	8.2	1CH 전송
			362	409	1000CH 전송
데이터 분배	DISTC	4	4.6	5.3	데이터 분배 동작 시
			6.0	7.3	스택 동작 시
데이터 추출	COLLC	4	5.3	6.5	데이터 추출 동작 시
			4.5	13.0	스택 동작 시
			5.8	6.1	스택 동작 시(1CH분을 FIFO하는 경우)
			42	142	스택 동작 시(1000CH분을 FIFO하는 경우)
비트 전송	MOVBC	4	4.9	5.7	
비트 카운터	BCNTC	4	5.5	6.4	1CH 카운트
			873	974	1000CH 카운트

4-1-34 펑션 블록용 특수 명령

명령 이름	니모닉	명령 길이(스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
변수 종류 취득	GETID	4	7.6	12.5	

4-1-35 SFC 제어 명령

명령 이름	니모닉	명령 길이(스텝)	실행 시간(μs)		실행 조건
			CJ2H CPU6□(-EIP)	CJ2M CPU□□	
스텝 활성화	SA	2	9.1	11.8	
스텝 비활성화	SE	2	9.1	11.9	
스텝 타이머 현재값 읽기	TSR	3	3.9	4.2	
스텝 타이머 현재값 변경	TSW	3	5.7	8.2	
SFC 실행 시작	SFCON	2	14.0	20.4	
SFC 실행 종료	SFCOFF	2	249	402	
SFC 태스크 정지(리셋 있음)	SFCPR	2	249	405	
SFC 태스크 정지(리셋 없음)	SFCPRN	2	249	405	

4-1-36 펑션 블록 사용 시의 인스턴스 실행 시간

펑션 블록 정의를 작성하고 사용자 프로그램에 인스턴스를 첨부한 경우, 인스턴스 실행이 사이클 시간에 미치는 영향은 다음을 기준으로 하십시오.

인스턴스가 사이클 시간에 미치는 영향 시간
= 기동 시간 A
+입출력 파라미터 전송 처리 시간 B
+펑션 블록 정의 내 명령 실행 시간 C

A, B, C의 시간은 각각 다음을 기준으로 하십시오.

내용			CPU 유니트 형식	
			CJ2H-CPU6□H-EIP	CJ2M-CPU□□
A	기동 시간	입출력 파라미터 전송을 제외한 기동 시간	3.3μs	7.4μs
B	입출력 파라미터 전송 처리 시간 괄호 안은 데이터 형식	1비트(BOOL)의 입출력 변수인 경우	0.24μs	0.88μs
		1워드(INT, UINT, WORD)의 입출력 변수인 경우	0.19μs	0.88μs
		2워드(DINT, UDINT, DWORD, REAL)의 입출력 변수인 경우	0.19μs	1.2μs
		4워드(LINT, ULINT, LWORD, LREAL)의 입출력 변수인 경우	0.38μs	2.96μs
C	펑션 블록 정의 내 명령 실행 시간	명령 처리 시간의 합계(평상시 사용자 프로그램과 동일)		

예) CJ2H-CPU67H-EIP에서
데이터 형식이 1워드(INT)인 입력 변수 = 3개,
데이터 형식이 1워드(INT)인 출력 변수 = 2개,
펑션 블록 정의 부분의 명령 처리 시간 합계가 10μs인 경우:
하나의 인스턴스 실행 시간= 3.3μs+ (3+2)×0.19μs+ 10μs = 14.25μs입니다.

주: 동일한 펑션 블록 정의를 여러 개 첨부한 경우(인스턴스가 여러 개인 경우) 인스턴스 수에 따라 실행 시간이 늘어납니다.

참 고 **펑션 블록 사용 시의 프로그램 스텝 수**

펑션 블록 정의를 작성하고 사용자 프로그램에 인스턴스를 첨부했을 때 프로그램 스텝 수는 다음을 기준으로 하십시오.

스텝 수
= 인스턴스 수×(호출부 용량 m+입출력 파라미터 전송부 용량 n×파라미터 수)
+펑션 블록 정의 내 명령 스텝 수 p(주)

주: 동일 펑션 블록 정의를 여러 개 삽입한 경우(인스턴스가 여러 개인 경우), 2개째 이후의 인스턴스에서는 펑션 블록 정의 내 명령 스텝 수(p)가 소비되지 않습니다. 따라서 위 계산식에서는 펑션 블록 정의 내 명령 스텝 수(p)에 인스턴스 수를 곱하지 않았습니다.

내용			CJ2 CPU 유니트
m	호출부		57스텝
n	입출력 파라미터 전송부 괄호 안은 데이터 형식	1비트(BOOL)의 입출력 변수인 경우	6스텝
		1워드(INT, UINT, WORD)의 입출력 변수인 경우	6스텝
		2워드(DINT, UDINT, DWORD, REAL)의 입출력 변수인 경우	6스텝
		4워드(LINT, ULINT, LWORD, LREAL)의 입출력 변수인 경우	12스텝
p	펑션 블록 정의 내 명령 스텝 수	명령 스텝 수의 합계(평상시 사용자 프로그램과 동일)에 27스텝을 가산한 스텝 수	

예)
데이터 형식이 1워드(INT)인 입력 변수 = 5개,
데이터 형식이 1워드(INT)인 출력 변수 = 5개,
펑션 블록 정의 부분의 스텝 수가 100스텝인 경우:
1개 인스턴스의 스텝 수= 57+(5+5)×6스텝+100스텝 + 27스텝 = 244스텝입니다.

4-2 CJ 시리즈(CJ1) 명령 처리 시간/스텝 수 목록

각 명령어의 처리 시간은 다음과 같습니다.

하나의 사용자 프로그램 전체(주)에서 각 명령어의 처리 시간을 합한 시간이 사이클 시간 중 '연산 처리 시간'입니다.

주: 정확하게는 사이클릭 태스크 중 실행 가능한 상태의 태스크와 인터럽트 조건이 성립하는 인터럽트 태스크에 할당된 프로그램입니다.

명령어의 처리 시간은 CPU 유니트의 형식에 따라 다르며, 명령 실행 시의 조건에 따라서도 달라집니다.

또한 명령을 실행하지 않을 때도 처리 시간이 걸리므로 주의하십시오.

동시에 명령 길이(스텝)를 나타내는데, 이는 피연산자를 포함한 각 명령어가 사용자 프로그램 영역을 얼마나 점유하는지 나타냅니다. 스텝 수는 명령어 또는 표현 형식에 따라 달라집니다.

또한 스텝이 명령어의 수를 나타내는 것은 아니므로 주의하십시오.

주1: 일부 명령어를 제외하고 미분 실행(기본 명령은 '↑ ↓', 응용 명령의 경우는 '@%') 옵션이 지원됩니다. 그 경우의 처리 시간에 다음 시간을 가산하십시오.
(단위: μ초)

조건	CJ1-H CPU 유니트			CJ1M CPU 유니트	CJ1 CPU 유니트
	CPU6□H-R	CPU6□H	CPU4□H	CPU□□	CPU4□
미분(↑ ↓)을 지원하는 경우	+0.24	+0.24	+0.32	+0.5	+0.45
미분(@%)을 지원하는 경우	+0.24	+0.24	+0.32	+0.5	+0.33

주2: 명령 비실행 시의 처리 시간의 기준은 다음과 같습니다.

CJ1-H CPU 유니트			CJ1M CPU 유니트	CJ1 CPU 유니트
CPU6□H-R	CPU6□H	CPU4□H	CPU□□	CPU4□
0.016~0.096	0.018~0.108	0.02~0.12	0.05~0.30	0.12~0.72

4-2-1 시퀀스 입력 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
로드	LD	1	0.016	0.02	0.04	0.10	0.10	0.08	
	!LD	2	21.16	21.16	21.20	24.20	28.17	21.24	
로드 부정	LD NOT	1	0.016	0.02	0.04	0.10	0.10	0.08	
	!LD NOT	2	21.16	21.16	21.20	24.20	28.17	21.24	
논리곱	AND	1	0.016	0.02	0.04	0.10	0.10	0.08	
	!AND	2	21.16	21.16	21.20	24.20	28.17	21.24	
부정 논리곱	AND NOT	1	0.016	0.02	0.04	0.10	0.10	0.08	
	!AND NOT	2	21.16	21.16	21.20	24.20	28.17	21.24	
논리합	OR	1	0.016	0.02	0.04	0.10	0.10	0.08	
	!OR	2	21.16	21.16	21.20	24.20	28.17	21.24	
부정 논리합	OR NOT	1	0.016	0.02	0.04	0.10	0.10	0.08	
	!OR NOT	2	21.16	21.16	21.20	24.20	28.17	21.24	
로드 논리곱	AND LD	1	0.016	0.02	0.04	0.05	0.05	0.08	
로드 논리합	OR LD	1	0.016	0.02	0.04	0.05	0.05	0.08	
부정	NOT	1	0.016	0.02	0.04	0.05	0.05	0.08	
P.F. 기동 미분	UP	3	0.24	0.3	0.42	0.50	0.50	0.54	
P.F. 종료 미분	DOWN	4	0.24	0.3	0.42	0.50	0.50	0.54	
LD 비트 테스트	LD TST	4	0.11	0.14	0.24	0.35	0.35	0.37	
LD 비트 테스트 부정	LD TSTN	4	0.11	0.14	0.24	0.35	0.35	0.37	
AND 비트 테스트 부정	AND TSTN	4	0.11	0.14	0.24	0.35	0.35	0.37	
OR 비트 테스트	OR TST	4	0.11	0.14	0.24	0.35	0.35	0.37	
OR 비트 테스트 부정	OR TSTN	4	0.11	0.14	0.24	0.35	0.35	0.37	

4-2-2 시퀀스 출력 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
출력	OUT	1	0.016	0.02	0.04	0.35	0.35	0.21	
	!OUT	2	21.39	21.39	21.41	23.42	28.95	21.58	
부정 출력	OUT NOT	1	0.016	0.02	0.04	0.35	0.35	0.21	
	!OUT NOT	2	21.39	21.39	21.41	23.42	28.95	21.58	
유지	KEEP	1	0.048	0.06	0.08	0.40	0.40	0.29	
	!KEEP	1	21.42	21.43	21.45	23.47	29.0	21.66	
기동 미분	DIFU	2	0.21	0.24	0.40	0.50	0.50	0.54	
종료 미분	DIFD	2	0.21	0.24	0.40	0.50	0.50	0.54	
세트	SET	1	0.016	0.02	0.06	0.30	0.30	0.21	
	!SET	2	21.39	21.39	21.43	23.47	28.90	21.58	
리셋	RSET	1	0.016	0.02	0.06	0.30	0.30	0.21	
	!RSET	2	21.39	21.39	21.43	23.47	28.90	21.58	
다중 비트 세트	SETA	4	5.8	5.8	6.1	11.8	11.8	7.8	1비트 세트
			25.7	25.7	27.2	64.1	64.1	38.8	1000비트 세트
다중 비트 리셋	RSTA	4	5.7	5.7	6.1	11.8	11.8	7.8	1비트 리셋
			25.8	25.8	27.1	64.0	64.0	38.8	1000비트 리셋
1비트 세트	SETB	2	0.19	0.24	0.34	0.5	0.5	-	
	!SETB	3	21.63	21.48	21.88	23.81	23.81	-	
1비트 리셋	RSTB	2	0.19	0.24	0.34	0.5	0.5	-	
	!RSTB	3	21.64	21.48	21.88	23.81	23.81	-	
1비트 출력	OUTB	2	0.19	0.22	0.32	0.45	0.45	-	
	!OUTB	3	21.61	21.64	21.84	23.67	23.67	-	

4-2-3 시퀀스 제어 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
종료	END	1	5.5	5.5	6.0	7.9	7.9	4.0	
기능 없음	NOP	1	0.016	0.02	0.04	0.05	0.05	0.12	
인터록	IL	1	0.048	0.06	0.06	0.15	0.15	0.12	
인터록 클리어	ILC	1	0.048	0.06	0.06	0.15	0.15	0.12	
멀티 인터록(미분 플래그 유지형)	MILH	3	7.5	7.5	7.9	13.3	14.6	-	인터록 조건이 성립하지 않을 때(입력 조건 ON)
			8.9	8.9	9.7	16.6	18.3	-	인터록 조건이 성립할 때(입력 조건 OFF)
			6.1	6.1	6.5	10.3	11.7	-	인터록 중에 다시 인터록 조건이 성립할 때(입력 조건 OFF)
멀티 인터록(미분 플래그 비유지형)	MILR	3	7.5	7.5	7.9	13.3	14.6	-	인터록 조건이 성립하지 않을 때(입력 조건 ON)
			8.9	8.9	9.7	16.6	18.3	-	인터록 조건이 성립할 때(입력 조건 OFF)
			6.1	6.1	6.5	10.3	11.7	-	인터록 중에 다시 인터록 조건이 성립할 때(입력 조건 OFF)
멀티 인터록 클리어	MILC	2	5.0	5.0	5.6	8.3	12.5	-	인터록 중이 아닐 때
			5.7	5.7	6.2	9.6	14.2	-	인터록 중일 때
점프	JMP	2	0.31	0.38	0.48	0.95	0.95	8.1	
점프 종료	JME	2	-	-	-	-	-	-	
조건부 점프	CJP	2	0.31	0.38	0.48	0.95	0.95	7.4	점프 조건이 성립할 때(입력 조건 ON)
조건부 부정 점프	CJPN	2	0.31	0.38	0.48	0.95	0.95	8.5	점프 조건이 성립할 때(입력 조건 OFF)
복수 점프	JMP0	1	0.048	0.06	0.06	0.15	0.15	0.12	
복수 점프 종료	JME0	1	0.048	0.06	0.06	0.15	0.15	0.12	
반복 시작	FOR	2	0.18	0.21	0.21	1.00	1.00	0.21	상수 지정
루프 중단	BREAK	1	0.048	0.12	0.12	0.15	0.15	0.12	
반복 종료	NEXT	1	0.14	0.18	0.18	0.45	0.45	0.18	루프를 계속할 때
			0.18	0.22	0.22	0.55	0.55	0.22	루프를 종료할 때

4-2-4 타이머/카운터 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
100ms 타이머	TIM	3	0.45	0.56	0.88	1.30	1.30	0.42	
	TIMX		0.45	0.56	0.88	1.30	1.30	0.42	
10ms 타이머	TIMH	3	0.70	0.88	1.14	1.80	1.80	0.42	
	TIMHX		0.46	0.56	0.88	1.30	1.30	0.42	
1ms 타이머	TMHH	3	0.69	0.86	1.12	1.75	1.75	0.42	
	TMHHX		0.46	0.56	0.88	1.30	1.30	0.42	
0.1ms 타이머	TIMU	3	0.45	-	-	-	-	-	
	TIMUX		0.45	-	-	-	-	-	
00.1ms 타이머	TMUH	3	0.45	-	-	-	-	-	
	TMUHX		0.45	-	-	-	-	-	
적산 타이머	TTIM	3	16.1	16.1	17.0	27.4	30.9	21.4	
			10.9	10.9	11.4	19.0	21.2	14.8	리셋할 때
			8.5	8.5	8.7	15.0	16.6	10.7	인터록할 때
	TTIMX	3	16.1	16.1	17.0	27.4		21.4	
			10.9	10.9	11.4	19.0		14.8	리셋할 때
			8.5	8.5	8.7	15.0		10.7	인터록할 때
장시간 타이머	TIML	4~5	7.6	7.6	10.0	16.3	17.2	12.8	
			6.2	6.2	6.5	13.8	15.3	7.8	인터록할 때
	TIMLX	4~5	7.6	7.6	10.0	16.3		12.8	
			6.2	6.2	6.5	13.8		7.8	인터록할 때
멀티 출력 타이머	MTIM	4	20.9	20.9	23.3	38.55	43.3	26.0	
			5.6	5.6	5.8	12.9	13.73	7.8	리셋할 때
	MTIMX	4	20.9	20.9	23.3	38.55		26.0	
			5.6	5.6	5.8	12.9		7.8	리셋할 때
카운터	CNT	3	0.51	0.56	0.88	1.30	1.30	0.42	
	CNTX		0.51						
가역 카운터	CNTR	3	16.9	16.9	19.0	31.8	27.2	20.9	
	CNTRX		16.9						
타이머/카운터 리셋	CNR	3	9.9	9.9	10.6	14.7	17.93	13.9	1CH 리셋
			4160	4160	4160	6210	6300	5420	1000CH 리셋
	CNRX	3	9.9	9.9	10.6	14.7	17.93	13.9	1CH 리셋
			4160	4160	4160	6210	6300	5420	1000CH 리셋

4-2-5 데이터 비교 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
심볼 비교(부호 없음)	=	4	0.08	0.10	0.16	0.35	0.35	0.37	
	< >								
	<								
	< =								
	>								
	> =								
심볼 비교(2워드, 부호 없음)	= L	4~7	0.08	0.10	0.16	0.35	0.35	0.54	
	< > L								
	< L								
	< = L								
	> L								
	> = L								
심볼 비교(부호 있음)	= S	4	0.08	0.10	0.16	0.35	0.35	6.50	
	< > S								
	< S								
	< = S								
	> S								
	> = S								
심볼 비교(2워드, 부호 있음)	= SL	4~7	0.08	0.10	0.16	0.35	0.35	6.50	
	< > SL								
	< SL								
	< = SL								
	> SL								
	> = SL								
시각 비교	= DT	4	25.2	25.2	36.4	45.6	40.6	-	
	< > DT								
	< DT								
	< = DT								
	> DT								
	> = DT								
비교	CMP	3	0.032	0.04	0.04	0.10	0.10	0.29	
	!CMP	7	42.2	42.2	42.2	45.3	45.3	42.7	
2워드 비교	CMPL	3~5	0.064	0.08	0.08	0.50	0.50	0.46	
부호 있는 BIN 비교	CPS	3	0.064	0.08	0.08	0.30	0.30	6.50	
	!CPS	7	36.0	36.0	36.0	45.5	45.5	48.9	
부호 있는 BIN 2워드 비교	CPSL	3~5	0.064	0.08	0.08	0.50	0.50	6.50	
테이블 일치	TCMP	4	14.0	14.0	15.2	29.77	32.13	21.9	
다채널 비교	MCMP	4	20.5	20.5	22.8	45.80	48.67	31.2	
부호 없는 테이블 간 비교	BCMP	4	21.5	21.5	23.7	47.93	51.67	32.6	
확장 테이블 간 비교	BCMP2	4	8.4	-	-	13.20	19.33	-	데이터 수 1
			313.0	-	-	650.0	754.67	-	데이터 수 255
영역 비교	ZCP	3	5.3	5.3	5.4	11.53	12.43	-	
2워드 영역 비교	ZCPL	3~5	5.5	5.5	6.7	11.28	11.90	-	

4-2-6 데이터 전송 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
전송	MOV	3	0.14	0.18	0.20	0.30	0.30	0.29	
	!MOV	7	21.52	21.56	21.60	35.4	43.3	42.65	
2워드 전송	MOVL	3~4	0.26	0.32	0.34	0.60	0.60	0.50	
부정 전송	MVN	3	0.14	0.18	0.20	0.35	0.35	0.29	
부정 2워드 전송	MVNL	3~4	0.26	0.32	0.34	0.60	0.60	0.50	
비트 전송	MOVB	4	0.19	0.24	0.34	0.50	0.50	7.5	
자릿수 전송	MOVD	4	0.19	0.24	0.34	0.50	0.50	7.3	
다중 비트 전송	XFRB	4	10.1	10.1	10.8	20.9	22.1	13.6	1비트 전송
			186.4	186.4	189.8	253.3	329.7	269.2	255비트 전송
블록 전송	XFER	4	0.29	0.36	0.44	0.8	0.8	11.2	1CH 전송
			240.1	300.1	380.1	650.2	650.2	633.5	1000CH 전송
블록 설정	BSET	4	0.21	0.26	0.28	0.55	0.55	8.5	1CH 세트
			142.2	200.1	220.1	400.2	400.2	278.3	1000CH 세트
데이터 교환	XCHG	3	0.32	0.40	0.56	0.80	0.80	0.7	
데이터 2워드 교환	XCGL	3	0.61	0.76	1.04	1.5	1.5	1.3	
데이터 분배	DIST	4	5.1	5.1	5.4	6.6	12.47	7.0	
데이터 추출	COLL	4	5.1	5.1	5.3	6.5	12.77	7.1	
인덱스 레지스터 설정	MOVR	3	0.064	0.08	0.08	0.60	0.60	0.50	
인덱스 레지스터 설정	MOVRW	3	0.064	0.42	0.50	0.60	0.60	0.50	

4-2-7 데이터 이동 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
이동 레지스터	SFT	3	7.4	7.4	10.4	11.9	15.3	10.4	1CH 이동
			433	433	488	1390	1430	763	1000CH 이동
좌우 이동 레지스터	SFTR	4	6.9	6.9	7.2	11.4	15.5	9.6	1CH 이동
			615.3	615.3	680.2	1430	1550	859.6	1000CH 이동
비동기 이동 레지스터	ASFT	4	6.2	6.2	6.4	13.4	14.2	7.7	1CH 이동
			1220	1220	1220	2750	2990	2010	1000CH 이동*1
워드 이동	WSFT	4	4.5	4.5	4.7	9.6	12.3	7.8	1CH 이동
			171.5	171.5	171.7	928.0	933.3	781.7	1000CH 이동
1비트 좌측 이동	ASL	2	0.18	0.22	0.32	0.45	0.45	0.37	
1비트 좌측 이동(2워드)	ASLL	2	0.32	0.40	0.56	0.80	0.80	0.67	
1비트 우측 이동	ASR	2	0.18	0.22	0.32	0.45	0.45	0.37	
1비트 우측 이동(2워드)	ASRL	2	0.32	0.40	0.56	0.80	0.80	0.67	
CY가 있는 1비트 좌측 회전	ROL	2	0.18	0.22	0.32	0.45	0.45	0.37	
CY가 있는 1비트 좌측 회전(2워드)	ROLL	2	0.32	0.40	0.56	0.80	0.80	0.67	
CY가 없는 1비트 좌측 회전	RLNC	2	0.18	0.22	0.32	0.45	0.45	0.37	
CY가 없는 1비트 좌측 회전(2워드)	RLNL	2	0.32	0.40	0.56	0.80	0.80	0.67	
CY가 있는 1비트 우측 회전	ROR	2	0.18	0.22	0.32	0.45	0.45	0.37	
CY가 있는 1비트 우측 회전(2워드)	RORL	2	0.32	0.40	0.56	0.80	0.80	0.67	
CY가 없는 1비트 우측 회전	RRNC	2	0.18	0.22	0.32	0.45	0.45	0.37	
CY가 없는 1비트 우측 회전(2워드)	RRNL	2	0.32	0.40	0.56	0.80	0.80	0.67	
1자리 좌측 이동	SLD	3	5.9	5.9	6.1	7.6	12.95	8.2	1CH 이동
			561.1	561.1	626.3	1150	1270	760.7	1000CH 이동
1자리 우측 이동	SRD	3	6.9	6.9	7.1	8.6	15.00	8.7	1CH 이동
			760.5	760.5	895.5	1720	1820	1070	1000CH 이동
N비트 데이터 좌측 이동	NSFL	4	7.5	7.5	8.3	14.8	16.0	10.5	1비트 이동
			34.5	40.3	45.4	86.7	91.3	55.5	1000비트 이동
N비트 데이터 우측 이동	NSFR	4	7.5	7.5	8.3	14.7	15.9	10.5	1비트 이동
			48.2	50.5	55.3	114.1	119.6	69.3	1000비트 이동
N비트 좌측 이동	NASL	3	0.18	0.22	0.32	0.45	0.45	0.37	
N비트 2워드 좌측 이동	NSLL	3	0.32	0.40	0.56	0.80	0.80	0.67	
N비트 우측 이동	NASR	3	0.18	0.22	0.32	0.45	0.45	0.37	
N비트 2워드 우측 이동	NSRL	3	0.32	0.40	0.56	0.80	0.80	0.67	

*1: 데이터량에 따라 명령 실행 시간이 크게 변화하며 사이클 시간에 영향을 줍니다. 사이클 시간에 대한 영향을 줄이기 위해 '백그라운드 실행'을 수행할 수 있습니다.

4-2-8 증가/감소 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
BIN 증가	++	2	0.18	0.22	0.32	0.45	0.45	0.37	
BIN 2워드 증가	++L	2	0.18	0.40	0.56	0.80	0.80	0.67	
BIN 감소	--	2	0.18	0.22	0.32	0.45	0.45	0.37	
BIN 2워드 감소	--L	2	0.18	0.40	0.56	0.80	0.80	0.67	
BCD 증가	++B	2	5.7	6.4	4.5	12.3	14.7	7.4	
BCD 2워드 증가	++BL	2	5.6	5.6	4.9	9.24	10.8	6.1	
BCD 감소	--B	2	5.7	6.3	4.6	11.9	14.9	7.2	
BCD 2워드 감소	--L	2	5.3	5.3	4.7	9.0	10.7	7.1	

4-2-9 사칙연산 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
부호 있음, CY 없는 BIN 가산	+	4	0.18	0.18	0.20	0.30	0.30	0.37	
부호 있음, CY 없는 BIN 2워드 가산	+L	4~6	0.18	0.32	0.34	0.60	0.60	0.54	
부호 및 CY 있는 BIN 가산	+C	4	0.18	0.18	0.20	0.40	0.40	0.37	
부호 및 CY 있는 BIN 2워드 가산	+CL	4~6	0.18	0.32	0.34	0.60	0.60	0.54	
CY 없는 BCD 가산	+B	4	7.6	8.2	8.4	18.9	21.5	14.0	
CY 없는 BCD 2워드 가산	+BL	4~6	9.2	13.3	14.5	24.4	27.7	19.0	
CY 있는 BCD 가산	+BC	4	8.0	8.9	9.1	19.7	22.6	14.5	
CY 있는 BCD 2워드 가산	+BCL	4~6	9.6	13.8	15.0	25.2	28.8	19.6	
부호 있음, CY 없는 BIN 감산	-	4	0.18	0.18	0.20	0.3	0.3	0.37	
부호 있음, CY 없는 BIN 2워드 감산	-L	4~6	0.18	0.32	0.34	0.60	0.60	0.54	
부호 및 CY 있는 BIN 감산	-C	4	0.18	0.18	0.20	0.40	0.40	0.37	
부호 및 CY 있는 BIN 2워드 감산	-CL	4~6	0.18	0.32	0.34	0.60	0.60	0.54	
CY 없는 BCD 감산	-B	4	7.4	8.0	8.2	18.1	20.5	13.1	
CY 없는 BCD 2워드 감산	-BL	4~6	8.9	12.8	14.0	23.2	26.7	18.2	
CY 있는 BCD 감산	-BC	4	7.9	8.5	8.6	19.1	21.6	13.8	
CY 있는 BCD 2워드 감산	-BCL	4~6	9.4	13.4	14.7	24.3	27.7	18.8	
부호 있는 BIN 승산	*	4	0.26	0.38	0.40	0.65	0.65	0.58	
부호 있는 BIN 2워드 승산	*L	4~6	5.93	7.23	8.45	13.17	15.0	11.19	
부호 없는 BIN 승산	*U	4	0.26	0.38	0.40	0.75	0.75	0.58	
부호 없는 BIN 2워드 승산	*UL	4~6	5.9	7.1	8.3	13.30	15.2	10.63	
BCD 승산	*B	4	8.3	9.0	9.2	17.5	19.7	12.8	
BCD 2워드 승산	*BL	4~6	12.8	23.0	24.2	36.3	45.7	35.2	
부호 있는 BIN 제산	/	4	0.29	0.40	0.42	0.70	0.70	0.83	
부호 있는 BIN 2워드 제산	/L	4~6	7.2	7.2	8.4	13.7	15.5	9.8	
부호 없는 BIN 제산	/U	4	0.29	0.40	0.42	0.8	0.8	0.83	
부호 없는 BIN 2워드 제산	/UL	4~6	6.9	6.9	8.1	12.8	14.7	9.1	
BCD 제산	/B	4	8.6	8.6	8.8	19.3	22.8	15.9	
BCD 2워드 제산	/BL	4~6	13.1	17.7	18.9	27.1	34.7	26.2	

4-2-10 데이터 변환 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
BCD→BIN 변환	BIN	3	0.18	0.22	0.24	0.40	0.40	0.29	
BCD→BIN 2워드 변환	BINL	3~4	6.1	6.5	6.8	12.3	13.7	9.1	
BIN→BCD 변환	BCD	3	0.19	0.24	0.26	7.62	9.78	8.3	
BIN→BCD 2워드 변환	BCDL	3~4	6.7	6.7	7.0	10.6	12.8	9.2	
2의 보수 변환	NEG	3	0.14	0.18	0.20	0.35	0.35	0.29	
2의 보수 2워드 변환	NEGL	3~4	0.26	0.32	0.34	0.60	0.60	0.5	
부호 확장	SIGN	3	0.26	0.32	0.34	0.60	0.60	0.50	
4→16/8→256 디코더	MLPX	4	0.32	0.32	0.42	0.85	0.85	8.8	4→16 디코드 시/자릿수 1자리
			0.98	0.98	1.20	1.60	1.60	12.8	4→16 디코드 시/자릿수 4자리
			3.30	3.30	4.00	4.70	4.70	20.3	8→256 디코드 시/자릿수 1자리
			6.50	6.50	7.90	8.70	8.70	33.4	8→256 디코드 시/자릿수 4자리
16→4/256→8 인코더	DMPX	4	7.5	7.5	7.9	9.4	13.9	10.4	16→4 디코드 시/자릿수 1자리
			49.6	49.6	50.2	57.3	71.73	59.1	16→4 디코드 시/자릿수 4자리
			18.2	18.2	18.6	56.8	82.7	23.6	256→8 디코드 시/자릿수 1자리
			55.1	55.1	57.4	100.0	150.7	92.5	256→8 디코드 시/자릿수 2자리
ASCII 코드 변환	ASC	4	6.8	6.8	7.1	8.3	14.6	9.7	1자리 ASCII 코드 변환
			9.0	11.2	11.7	19.1	21.8	15.1	4자리 ASCII 코드 변환
ASCII→HEX 변환	HEX	4	7.1	7.1	7.4	12.1	15.6	10.1	1자리 변환
비트 열→비트 행 변환	LINE	4	16.6	19.0	23.1	37.0	40.3	29.1	
비트 행→비트 열 변환	COLM	4	18.4	23.2	27.5	45.7	48.2	37.3	
부호 있는 BCD→BIN 변환	BINS	4	6.8	8.0	8.3	16.2	17.0	12.1	데이터 형식 지정 0
			6.8	8.0	8.3	16.2	17.1	12.1	데이터 형식 지정 1
			7.1	8.3	8.6	16.5	17.7	12.7	데이터 형식 지정 2
			7.4	8.5	8.8	16.5	17.6	13.0	데이터 형식 지정 3
부호 있는 BCD→BIN 2워드 변환	BISL	4~5	6.9	9.2	9.6	18.4	19.6	13.6	데이터 형식 지정 0
			7.0	9.2	9.6	18.5	19.8	13.7	데이터 형식 지정 1
			7.3	9.5	9.9	18.6	20.1	14.2	데이터 형식 지정 2
			7.6	9.6	10.0	18.7	20.1	14.4	데이터 형식 지정 3
부호 있는 BIN→BCD 변환	BCDS	4	6.6	6.6	6.9	13.5	16.4	10.6	데이터 형식 지정 0
			6.7	6.7	7.0	13.8	16.7	10.8	데이터 형식 지정 1
			6.8	6.8	7.1	13.9	16.8	10.9	데이터 형식 지정 2
			7.1	7.2	7.5	14.0	17.1	11.5	데이터 형식 지정 3
부호 있는 BIN→BCD 2워드 변환	BDSL	4~5	7.6	8.1	8.4	11.4	12.5	11.6	데이터 형식 지정 0
			6.7	8.2	8.6	11.7	12.73	11.8	데이터 형식 지정 1
			6.7	8.3	8.7	11.8	12.8	12.0	데이터 형식 지정 2
			6.9	8.8	9.2	11.9	13.0	12.5	데이터 형식 지정 3
그레이 코드 변환	GRY	4	46.9	46.9	72.1	80.0	71.2	-	8비트 BIN
			49.6	49.6	75.2	83.0	75.6	-	8비트 BCD
			57.7	57.7	87.7	95.9	86.4	-	8비트 각도
			61.8	61.8	96.7	104.5	91.6	-	15비트 BIN
			64.5	64.5	99.6	107.5	96.1	-	15비트 BCD
			72.8	72.8	112.4	120.4	107.3	-	15비트 각도
			52.3	52.3	87.2	88.7	82.4	-	360 BIN
			55.1	55.1	90.4	91.7	86.8	-	360 BCD
			64.8	64.8	98.5	107.3	98.1	-	360 각도
숫자 4자리→ASCII 변환	STR4	3	13.79	13.79	20.24	22.16	19.88	-	
숫자 8자리→ASCII 변환	STR8	3~4	18.82	18.82	27.44	29.55	26.70	-	
숫자 16자리→ASCII 변환	STR16	3	30.54	30.54	44.41	48.16	44.10	-	
ASCII→숫자 4자리 변환	NUM4	3~4	18.46	18.46	27.27	29.13	26.88	-	
ASCII→숫자 8자리 변환	NUM8	3	27.27	27.27	40.29	42.69	39.71	-	
ASCII→숫자 16자리 변환	NUM16	3	52.31	52.31	78.25	82.21	74.23	-	

4-2-11 논리 연산 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
워드 논리곱	ANDW	4	0.14	0.18	0.20	0.30	0.30	0.37	
워드 논리곱(2워드)	ANDL	4~6	0.26	0.32	0.34	0.60	0.60	0.54	
워드 논리합	ORW	4	0.18	0.22	0.32	0.45	0.45	0.37	
워드 논리합(2워드)	ORWL	4~6	0.26	0.32	0.34	0.60	0.60	0.54	
워드 배타적 논리합	XORW	4	0.18	0.22	0.32	0.45	0.45	0.37	
워드 배타적 논리합(2워드)	XORL	4~6	0.26	0.32	0.34	0.60	0.60	0.54	
워드 배타적 논리합 부정	XNRW	4	0.18	0.22	0.32	0.45	0.45	0.37	
워드 배타적 논리합 부정(2워드)	XNRL	4~6	0.26	0.32	0.34	0.60	0.60	0.54	
비트 반전	COM	2	0.18	0.22	0.32	0.45	0.45	0.37	
비트 2워드 반전	COML	2	0.32	0.40	0.56	0.80	0.80	0.67	

4-2-12 특수 연산 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
BIN 제곱근 연산	ROTB	3	49.6	49.6	50.0	56.5	82.7	530.7	
BCD 제곱근 연산	ROOT	3	13.7	13.7	13.9	59.3	88.4	514.5	
숫자 변환	APR	4	6.7	6.7	6.9	14.0	15.0	32.3	SIN, COS 계산 지정
			17.2	17.2	18.4	32.2	37.9	78.3	꺾은선 근사 계산 지정
부동 소수점 제산(BCD)	FDIV	4	116.6	116.6	176.6	246.0	154.7	176.6	
비트 카운터	BCNT	4	0.24	0.3	0.38	0.65	0.65	22.1	1CH 카운트

4-2-13 부동 소수점 변환 및 연산 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
부동 소수점→16비트 BIN 변환	FIX	3~4	0.13	10.6	10.8	16.2	19.5	14.5	
부동 소수점→32비트 BIN 변환	FIXL	3~4	0.13	10.8	11.0	16.6	21.7	14.6	
16비트 BIN→부동 소수점 변환	FLT	3~4	0.13	8.3	8.5	12.2	14.6	11.1	
32비트 BIN→부동 소수점 변환	FLTL	3~4	0.13	8.3	8.5	14.0	15.8	10.8	
부동 소수점 가산	+F	4~6	0.24	8.0	9.2	13.3	15.7	10.2	
부동 소수점 감산	-F	4~6	0.24	8.0	9.2	13.3	15.8	10.3	
부동 소수점 제산	/F	4~6	0.4	8.7	9.9	14.0	17.6	12.0	
부동 소수점 승산	*F	4~6	0.24	8.0	9.2	13.2	15.8	10.5	
각도→라디안 변환	RAD	3~4	8.1	10.1	10.2	15.9	20.6	14.9	
라디안→각도 변환	DEG	3~4	8.0	9.9	10.1	15.7	20.4	14.8	
SIN 연산	SIN	3~4	42.0	42.0	42.2	47.9	70.9	61.1	
고속 SIN 연산	SINQ	8~9	0.59	-	-	-	-	-	
COS 연산	COS	3~4	31.5	31.5	31.8	41.8	51.0	44.1	
고속 COS 연산	COSQ	8~9	0.59	-	-	-	-	-	
TAN 연산	TAN	3~4	16.3	16.3	16.6	20.8	27.6	22.6	
고속 TAN 연산	TANQ	15~16	1.18	-	-	-	-	-	
SIN^{-1} 연산	ASIN	3~4	17.6	17.6	17.9	80.3	122.9	24.1	
COS^{-1} 연산	ACOS	3~4	20.4	20.4	20.7	25.3	33.5	28.0	
TAN^{-1} 연산	ATAN	3~4	16.1	16.1	16.4	45.9	68.9	16.4	
제곱근 연산	SQRT	3~4	0.42	19.0	19.3	26.2	33.2	28.1	
지수 연산	EXP	3~4	65.9	65.9	66.2	68.8	108.2	96.7	
로그 연산	LOG	3~4	12.8	12.8	13.1	69.4	103.7	17.4	
거듭제곱 연산	PWR	4~6	125.4	125.4	126.0	134.0	201.0	181.7	
단정도 부동 소수점 데이터 비교	=F	4~6	0.13	6.6	8.3	12.6	15.37	-	
	<>F								
	<F								
	<=F								
	>F								
	>=F								
부동 소수점<단>→문자열 변환	FSTR	4~5	48.5	48.5	48.9	58.4	85.7	-	
문자열→부동 소수점<단> 변환	FVAL	3	21.1	21.1	21.3	31.1	43.73	-	
부동 소수점 전송<단>	MOVF	3~4	0.18	-	-	-	-	-	

4-2-14 배정도 부동 소수점 변환 및 연산 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
배정도 부동 소수점 데이터 비교	=D	4	8.5	8.5	10.3	16.2	19.9	-	
	<>D								
	<D								
	<=D								
	>D								
	>=D								
부동 소수점→16비트 BIN 변환<배>	FIXD	3	11.0	11.7	12.1	16.1	21.6	-	
부동 소수점→32비트 BIN 변환<배>	FIXLD	3	10.2	11.6	12.1	16.4	21.7	-	
16비트 BIN→부동 소수점 변환<배>	DBL	3	9.9	9.9	10.0	14.3	16.5	-	
32비트 BIN→부동 소수점 변환<배>	DBLL	3	9.8	9.8	10.0	16.0	17.7	-	
부동 소수점 가산<배>	+D	4	11.2	11.2	11.9	18.3	23.6	-	
부동 소수점 감산<배>	-D	4	11.2	11.2	11.9	18.3	23.6	-	
부동 소수점 승산<배>	*D	4	12.0	12.0	12.7	19.0	25.0	-	
부동 소수점 제산<배>	/D	4	23.5	23.5	24.2	30.5	44.3	-	
각도→라디안 변환<배>	RADD	3	11.5	27.4	27.8	32.7	49.1	-	
라디안→각도 변환<배>	DEGD	3	11.2	11.2	11.9	33.5	48.4	-	
SIN 연산<배>	SIND	3	45.4	45.4	45.8	67.9	76.7	-	
COS 연산<배>	COSD	3	43.0	43.0	43.4	70.9	72.3	-	
TAN 연산<배>	TAND	3	19.8	20.1	20.5	97.9	157.0	-	
SIN^{-1} 연산<배>	ASIND	3	21.5	21.5	21.9	32.3	37.3	-	
COS^{-1} 연산<배>	ACOSD	3	24.7	24.7	25.1	29.9	42.5	-	
TAN^{-1} 연산<배>	ATAND	3	19.3	19.3	19.7	24.0	34.4	-	
제곱근 연산<배>	SQRTD	3	47.4	47.4	47.9	52.9	81.9	-	
지수 연산<배>	EXPD	3	121.0	121.0	121.4	126.3	201.3	-	
로그 연산<배>	LOGD	3	16.0	16.0	16.4	21.6	29.3	-	
거듭제곱 연산<배>	PWRD	4	223.9	223.9	224.2	232.3	373.4	-	

4-2-15 테이블 데이터 처리 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
스택 영역 설정	SSET	3	8.0	8.0	8.3	14.2	20.3	8.5	스택 영역에 5CH 설정
			231.6	231.6	251.8	426.5	435.3	276.8	스택 영역에 1000CH 설정
스택 데이터 저장	PUSH	3	6.5	6.5	8.6	15.7	16.4	9.1	
선입선출	FIFO	3	6.9	6.9	8.9	15.8	16.8	10.6	스택 영역에 5CH 설정
			352.6	352.6	434.3	728.0	732.0	1130	스택 영역에 1000CH 설정
후입선출	LIFO	3	7.0	7.0	9.0	16.6	17.2	9.9	
테이블 영역 선언	DIM	5	15.2	15.2	21.6	27.8	27.1	142.1	
레코드 위치 설정	SETR	4	5.4	5.4	5.9	12.8	13.2	7.0	
레코드 위치 읽기	GETR	4	7.8	7.8	8.4	16.1	18.3	11.0	
데이터 검색	SRCH	4	15.5	15.5	19.5	29.1	26.4	19.5	1CH 검색
			2420	2420	3340	4410	3600	3340	1000CH 검색*1
바이트 스와핑	SWAP	3	12.2	12.2	13.6	21.0	18.4	13.6	1CH 교체
			1940	1940	2820	3650	3150	2820	1000CH 교체*1
최대값 검색	MAX	4~5	19.2	19.2	24.9	35.3	32.0	24.9	데이터 수 1
			2390	2390	3360	4390	3570	3360	데이터 수 1000 *1
최소값 검색	MIN	4~5	19.2	19.2	25.3	35.4	31.9	25.3	데이터 수 1
			2390	2390	3330	4390	3580	3330	데이터 수 1000 *1
SUM 값 산출	SUM	4~5	28.2	28.2	38.5	49.5	44.1	38.3	1CH 산출
			1420	1420	1950	2330	2110	1950	1000CH 산출*1
FCS 값 산출	FCS	4~5	20.0	20.0	28.3	34.8	31.5	28.3	테이블 길이 1
			1650	1650	2480	3110	2770	2480	테이블 길이 1000 *1
스택 데이터 수 출력	SNUM	3	6.0	6.0	6.3	12.1	13.7	-	
스택 데이터 참조	SREAD	4	8.0	8.0	8.4	18.1	20.6	-	
스택 데이터 갱신	SWRIT	4	7.2	7.2	7.6	16.9	18.8	-	
스택 데이터 삽입	SINS	4	7.8	7.8	9.9	18.2	20.5	-	
			354.0	354.0	434.8	730.7	732.0	-	테이블 길이 1000
스택 데이터 삭제	SDEL	4	8.6	8.6	10.6	19.3	22.0	-	
			354.0	354.0	436.0	732.0	744.0	-	테이블 길이 1000

*1: 데이터량에 따라 명령 실행 시간이 크게 변화하며 사이클 시간에 영향을 줍니다. 사이클 시간에 대한 영향을 줄이기 위해 '백그라운드 실행'을 수행할 수 있습니다.

4-2-16 데이터 제어 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
PID 연산	PID	4	436.2	436.2	678.2	612.0	552.6	678.2	처음 실행할 때
			332.3	332.3	474.9	609.3	548.0	474.9	입력 ON이고 샘플링할 때
			97.3	97.3	141.3	175.3	162.0	141.3	입력 ON이고 샘플링하지 않을 때
자동 조정 가능한 PID 연산	PIDAT	4	446.3	446.3	712.5	765.3	700.0	-	처음 실행할 때
			339.4	339.4	533.9	620.7	558.0	-	입력 ON이고 샘플링할 때
			100.7	100.7	147.1	180.0	166.1	-	입력 ON이고 샘플링하지 않을 때
			189.2	189.2	281.6	233.7	225.1	-	AT에서 처음 실행할 때
			535.2	535.2	709.8	575.3	558.2	-	AT에서 샘플링할 때
상하한 리미트 제어	LMT	4~5	16.1	16.1	22.1	27.1	26.1	22.1	
데드 밴드 제어	BAND	4~5	17.0	17.0	22.5	27.4	26.6	22.5	
데드 존 제어	ZONE	4~5	15.4	15.4	20.5	28.0	26.4	20.5	
시분할 비례 출력	TPO	4	10.6	10.6	14.8	20.2	19.8	-	입력 OFF
			54.5	54.5	82.0	92.7	85.1	-	입력 ON 및 듀티 지정 또는 출력 리미트 무효
			61.0	61.0	91.9	102.5	95.3	-	입력 ON 및 조작량 지정과 출력 리미트 유효
스케일링	SCL	4	13.9	13.9	14.3	25.0	32.8	56.8	
스케일링 2	SCL2	4	12.2	12.2	12.6	22.3	29.1	50.7	
스케일링 3	SCL3	4	13.7	13.7	14.2	25.6	30.0	57.7	
데이터 평균화	AVG	4	36.3	36.3	52.6	62.9	59.1	53.1	1회 평균
			291.0	291.0	419.9	545.3	492.7	419.9	64회 평균

4-2-17 서브루틴 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
서브루틴 호출	SBS	2	0.90	1.26	1.96	2.04	2.04	17.00	
서브루틴 진입	SBN	2	-	-	-	-	-	-	
서브루틴 반환	RET	1	0.43	0.86	1.60	1.80	1.80	20.60	
매크로	MCRO	4	23.3	23.3	23.3	47.9	50.3	23.3	
글로벌 서브루틴 호출	GSBS	2	0.90	1.26	1.96	2.04	2.04	-	
글로벌 서브루틴 진입	GSBN	2	-	-	-	-	-	-	
글로벌 서브루틴 반환	GRET	1	0.43	0.86	1.60	1.80	1.80	-	

4-2-18 인터럽트 제어 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
인터럽트 마스크 세트	MSKS	3	25.6	25.6	38.4	44.7	42.9	39.5	
인터럽트 마스크 읽기	MSKR	3	11.9	11.9	11.9	16.9	15.9	11.9	
인터럽트 해제	CLI	3	27.4	27.4	41.3	42.7	44.5	41.3	
인터럽트 태스크 실행 금지	DI	1	15.0	15.0	16.8	30.3	28.5	16.8	
인터럽트 태스크 실행 금지 해제	EI	1	19.5	19.5	21.8	37.7	34.4	21.8	

4-2-19 고속 카운터/펄스 출력 명령(CJ1M-CPU21/22/23만 해당)

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
동작 모드 제어	INI	4	-	-	-	77.00	80.4	-	고속 카운터 비교 시작
			-	-	-	43.00	43.0	-	고속 카운터 비교 정지
			-	-	-	43.40	48.8	-	펄스 출력 현재값 변경
			-	-	-	51.80	50.8	-	고속 카운터 현재값 변경
			-	-	-	31.83	28.5	-	인터럽트 입력 카운터 모드 현재값 변경
			-	-	-	45.33	49.8	-	펄스 출력 정지
			-	-	-	36.73	30.5	-	PWM 출력 정지
펄스 현재값 읽기	PRV	4	-	-	-	42.40	43.9	-	펄스 출력 현재값 읽기
			-	-	-	53.40	65.9	-	고속 카운터 현재값 읽기
			-	-	-	33.60	30.5	-	인터럽트 입력 카운터 모드 현재값 읽기
			-	-	-	38.80	40.0	-	펄스 출력 상태 읽기
			-	-	-	39.30	66.9	-	고속 카운터 현재값 읽기
			-	-	-	38.30	34.5	-	PWM 상태 읽기
			-	-	-	117.73	145.7	-	고속 카운터 대역 비교 결과 읽기
			-	-	-	48.20	48.5	-	고속 카운터 0주파수 읽기
펄스 주파수 변환	PRV2	4	-	-	-	23.03	22.39	-	
비교 테이블 등록	CTBL	4	-	-	-	238.0	235.0	-	목표 일치 테이블 등록 및 비교 시작(1점)
			-	-	-	14420	9970	-	목표 일치 테이블 등록 및 비교 시작(48점)
			-	-	-	289.0	276.0	-	대역 비교 테이블 등록 및 비교 시작
			-	-	-	198.0	183.0	-	목표 일치 테이블 등록만(1점)
			-	-	-	14400	9610	-	목표 일치 테이블 등록만(48점)
			-	-	-	259.0	239.0	-	대역 비교 테이블 등록만
주파수 설정	SPED	4	-	-	-	56.00	89.3	-	연속 모드
			-	-	-	62.47	94.9	-	단독 모드
펄스량 설정	PULS	4	-	-	-	26.20	32.9	-	
위치 결정	PLS2	5	-	-	-	100.80	107.5	-	
주파수 가감속 제어	ACC	4	-	-	-	90.80	114.8	-	연속 모드
			-	-	-	80.00	122.1	-	단독 모드
원점 검색	ORG	3	-	-	-	106.13	116.0	-	원점 검색
			-	-	-	52.00	102.1	-	원점 복귀
PWM 출력	PWM	4	-	-	-	25.80	33.0	-	

4-2-20 공정 진행 제어 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
스텝 래더 영역 정의	STEP	2	17.4	17.4	20.7	35.9	37.1	27.1	스텝 릴레이가 ON일 때
			11.8	11.8	13.7	13.8	18.3	24.4	스텝 릴레이가 OFF일 때
스텝 래더 영역 진행	SNXT	2	6.6	6.6	7.3	12.1	14.0	10.0	

4-2-21 I/O 유니트용 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
I/O 리프레시	IORF	3	15.5	15.5	16.4	26.7	30.4	23.5	1CH IN
			17.20	17.20	18.40	29.7	35.0	25.6	1CH OUT
고기능 I/O 유니트 매회 리프레시	FIORF	2	*1	-	-	-	-	-	
CPU 고기능 유니트 매회 I/O 리프레시	DLNK	4	287.8	287.8	315.5	321.3	458.7	-	1CH 점유
7세그먼트 디코더	SDEC	4	6.5	6.5	6.9	8.1	15.7	14.1	
디지털 스위치	DSW	6	50.7	50.7	73.5	77.7	77.6	-	4자리 지정, 데이터 입력값: 0
			51.3	51.3	73.5	83.2	80.0	-	8자리 지정, 데이터 입력값: 00
10키 입력	TKY	4	9.7	9.7	13.2	18.7	18.6	-	데이터 입력값: 00
			10.7	10.7	14.8	20.2	19.1	-	데이터 입력값: FF
16키 입력	HYK	5	50.3	50.3	70.9	77.3	78.1	-	데이터 입력값: 00
			50.1	50.1	71.2	76.8	77.3	-	데이터 입력값: FF
행렬 입력	MTR	5	47.8	47.8	68.1	76.4	77.7	-	데이터 입력값: 00
			48.0	48.0	68.0	77.7	76.9	-	데이터 입력값: FF
7세그먼트 표시	7SEG	5	58.1	58.1	83.3	89.6	89.9	-	4자리 지정 시
			63.3	63.3	90.3	98.3	99.2	-	8자리 지정 시
인텔리전트 I/O 읽기	IORD	4	*1	*1	*1	*1	*1	*1	
인텔리전트 I/O 쓰기	IOWR	4	*1	*1	*1	*1	*1	*1	

*1: IORD, IORW, FIORF 명령은 읽기 상대의 고기능 I/O 유니트에 따라 다릅니다.

4-2-22 시리얼 통신 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
프로토콜 매크로	PMCR	5	100.1	100.1	142.1	158.4	206.0	276.8	직접 지정
			134.2	134.2	189.6	210.0	256.7	305.9	피연산자 지정 송신 1CH, 수신 1CH
시리얼 포트 출력	TXD	4	68.5	68.5	98.8	109.3	102.9	98.8	송신 바이트 수 1바이트
			734.3	734.3	1100	1230	1160	1100	송신 바이트 수 256바이트
시리얼 포트 입력	RXD	4	89.6	89.6	131.1	144.0	132.1	131.1	저장 바이트 수 1바이트
			724.2	724.2	1110	1310	1220	1110	저장 바이트 수 256바이트
시리얼 커뮤니케이션 유니트 시리얼 포트 출력	TXDU	4	131.5	131.5	202.4	213.4	208.6	-	송신 바이트 수 1바이트
시리얼 커뮤니케이션 유니트 시리얼 포트 입력	RXDU	4	131	131	200.8	211.8	206.8	-	저장 바이트 수 1바이트
시리얼 통신 설정 변경	STUP	3	341.2	341.2	400.0	504.7	524.7	440.4	CPU 유니트의 COM 포트 대상

4-2-23 네트워크 통신용 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
네트워크 송신	SEND	4	84.4	84.4	123.9	141.6	195.0	123.9	
네트워크 수신	RECV	4	85.4	85.4	124.7	142.3	196.7	124.7	
명령 송신	CMND	4	106.8	106.8	136.8	167.7	226.7	136.8	
범용 Explicit 메시지 송신	EXPLT	4	127.6	127.6	190.0	217.0	238.0	-	
Explicit 읽기	EGATR	4	123.9	123.9	185.0	210.0	232.7	-	
Explicit 쓰기	ESATR	3	110.0	110.0	164.4	188.3	210.3	-	
Explicit CPU 유니트 데이터 읽기	ECHRD	4	106.8	106.8	158.9	176.3	220.3	-	
Explicit CPU 유니트 데이터 쓰기	ECHWR	4	106.0	106.0	158.3	175.7	205.3	-	

4-2-24 파일 메모리용 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
데이터 파일 읽기	FREAD	5	391.4	391.4	632.4	657.3	641.3	684.1	BIN 형식(디렉토리명 + 파일명)이 2문자일 때
데이터 파일 쓰기	FWRIT	5	387.8	387.8	627.0	650.7	637.3	684.7	BIN 형식(디렉토리명 + 파일명)이 2문자일 때
텍스트 파일 쓰기	TWRIT	5	390.1	390.1	619.1	555.3	489.0	-	

4-2-25 표시 기능용 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
메시지 표시	MSG	3	10.1	10.1	14.2	16.8	17.3	14.3	메시지 표시
			8.4	8.4	11.3	14.7	14.7	11.3	메시지 표시 해제

4-2-26 시계 기능 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
캘린더 가산	CADD	4	34.0	38.3	201.9	217.0	194.0	209.5	
캘린더 감산	CSUB	4	29.6	38.6	170.4	184.7	167.0	184.1	
시분초→초 변환	SEC	3	7.8	21.4	29.3	36.1	35.4	35.8	
초→시분초 변환	HMS	3	7.7	22.2	30.9	45.1	45.7	42.1	
시계 보정	DATE	2	216.0	216.0	251.5	118.7	128.3	120.0	

4-2-27 디버그 처리 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
트레이스 메모리 샘플링	TRSM	1	80.4	80.4	120.0	207.0	218.3	120.0	1점+0CH 샘플링
			848.1	848.1	1060	1160	1100	1060	31점+6CH 샘플링

4-2-28 고장 진단 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
운전 계속 고장 진단	FAL	3	15.4	15.4	16.7	26.1	24.47	16.7	이상 등록
			179.8	179.8	244.8	294.0	264.0	244.8	이상 해제(최우선)
			432.4	432.4	657.1	853.3	807.3	657.1	이상 해제(전체 이상)
			161.5	161.5	219.4	265.7	233.0	219.4	이상 해제(개별)
운전 정지 고장 진단	FALS	3	-	-	-	-	-	-	
고장 지점 검출	FPD	4	140.9	140.9	202.3	220.7	250.0	202.3	접점 주소 출력 시간 감시 중
			163.4	163.4	217.6	250.3	264.3	217.6	접점 주소 출력 이상을 처음 검출했을 때
			185.2	185.2	268.9	220.7	321.7	268.9	메시지 문자 출력 시간 감시 중
			207.5	207.5	283.6	320.7	336.0	283.6	메시지 문자 출력 이상을 처음 검출했을 때

4-2-29 특수 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
세트 캐리	STC	1	0.048	0.06	0.06	0.15	0.15	0.12	
클리어 캐리	CLC	1	0.048	0.06	0.06	0.15	0.15	0.12	
확장 DM 뱅크 전환	EMBC	2	14.0	14.0	15.1	-	-	15.1	
사이클 시간 감시 시간 설정	WDT	2	15.0	15.0	19.7	23.6	22.0	19.7	
조건 플래그 저장	CCS	1	8.6	8.6	12.5	14.2	12.9	-	
조건 플래그 로드	CCL	1	9.8	9.8	13.9	16.3	15.7	-	
CV→CS 주소 변환	FRMCV	3	13.6	13.6	19.9	23.1	31.8	-	
CS→CV 주소 변환	TOCV	3~4	11.9	11.9	17.2	22.5	31.4	-	
주변 서비스 금지	IOSP	1	13.9	13.9	19.8	21.5	21.5	-	
주변 서비스 금지 해제	IORS	1	63.6	63.6	92.3	22.2	22.2	-	

4-2-30 블록 프로그램 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
블록 프로그램	BPRG	2	12.1	12.1	13.0	27.5	30.4	13.0	
블록 프로그램 종료	BEND	1	9.6	9.6	12.3	23.2	27.1	13.1	
블록 프로그램 일시 정지	BPPS	2	10.6	10.6	12.3	16.0	21.7	14.9	
블록 프로그램 재기동	BPRS	2	5.1	5.1	5.6	9.0	10.2	8.3	
조건부 종료	입력 조건 EXIT	1	10.0	10.0	11.3	23.8	26.0	12.9	EXIT 실행(입력 조건 ON)
			4.0	4.0	4.9	7.2	8.4	7.3	EXIT 실행 안 함(입력 조건 OFF)
	EXIT 릴레이	2	9.8	9.8	13.5	28.4	30.6	16.3	EXIT 실행(릴레이 ON)
			4.7	4.7	7.2	11.4	13.1	10.7	EXIT 실행 안 함(릴레이 OFF)
조건부 종료(부정)	EXIT NOT 릴레이	2	12.4	12.4	14.0	28.4	31.2	16.8	EXIT 실행(릴레이 OFF)
			7.1	7.1	7.6	11.8	13.5	11.2	EXIT 실행 안 함(릴레이 ON)
조건 분기 블록	입력 조건 IF	1	4.6	4.6	4.8	6.8	8.5	7.2	IF 참(입력 조건 ON)
			6.7	6.7	7.3	12.2	13.9	10.9	IF 거짓(입력 조건 OFF)
조건 분기 블록	IF 릴레이	2	6.8	6.8	7.2	11.0	12.7	10.4	IF 참(릴레이 ON)
			9.0	9.0	9.6	16.5	18.5	14.2	IF 거짓(릴레이 OFF)
조건 분기 블록(부정)	IF NOT 릴레이	2	7.1	7.1	7.6	11.5	13.1	10.9	IF 참(릴레이 OFF)
			9.2	9.2	10.1	16.8	18.9	14.7	IF 거짓(릴레이 ON)
조건 분기 거짓 블록	ELSE	1	6.2	6.2	6.7	11.4	12.6	9.9	IF 참
			6.8	6.8	7.7	13.4	15.0	11.2	IF 거짓
조건 분기 블록 종료	IEND	1	6.9	6.9	7.7	13.5	15.4	11.0	IF 참
			4.4	4.4	4.6	6.93	8.1	7.0	IF 거짓
1스캔 조건 대기	입력 조건 WAIT	1	12.6	12.6	13.7	28.6	34.0	16.7	WAIT 실행 안 함(입력 조건 ON)
			3.9	3.9	4.1	5.6	6.9	6.3	WAIT 실행(입력 조건 OFF)
1스캔 조건 대기	WAIT 릴레이	2	12.0	12.0	13.4	27.2	30.0	16.5	WAIT 실행 안 함(릴레이 ON)
			6.1	6.1	6.5	10.0	11.4	9.6	WAIT 실행(릴레이 OFF)
1스캔 조건 대기(부정)	WAIT NOT 릴레이	2	12.2	12.2	13.8	27.8	30.6	17.0	WAIT 실행 안 함(릴레이 OFF)
			6.4	6.4	6.9	10.5	11.8	10.1	WAIT 실행(릴레이 ON)
100ms 타이머 대기	TIMW	3	22.3	22.3	25.2	47.4	52.0	33.1	처음 기동할 때
			24.9	24.9	27.8	46.2	53.4	35.7	처음 이후
	TIMWX	3	22.3	22.3	25.2	47.4	52.0	33.1	처음 기동할 때
			24.9	24.9	27.8	46.2	53.4	35.7	처음 이후
10ms 타이머 대기	TMHW	3	25.8	25.8	27.9	47.9	53.7	34.1	처음 기동할 때
			20.6	20.6	22.7	40.9	46.2	28.9	처음 이후
	TMHWX	3	25.8	25.8	27.9	47.9	53.7	34.1	처음 기동할 때
			20.6	20.6	22.7	40.9	46.2	28.9	처음 이후
카운터 대기	CNTW	4	17.9	17.9	22.6	41.0	43.5	27.4	처음 기동할 때
			19.1	19.1	23.9	42.9	45.7	28.7	처음 이후
	CNTWX	4	17.9	17.9	22.6	41.0	43.5	27.4	처음 기동할 때
			19.1	19.1	23.9	42.9	45.7	28.7	처음 이후
반복 블록	LOOP	1	7.9	7.9	9.1	15.6	17.6	12.3	
반복 블록 종료	입력 조건 LEND	1	7.7	7.7	8.4	13.5	15.5	10.9	LOOP 실행 안 함(입력 조건 ON)
			6.8	6.8	8.0	17.5	19.8	9.8	LOOP 실행(입력 조건 OFF)
반복 블록 종료	LEND 릴레이 번호	2	9.9	9.9	10.7	17.5	19.9	14.4	LOOP 실행 안 함(릴레이 ON)
			8.9	8.9	10.3	21.6	24.5	13.0	LOOP 실행(릴레이 OFF)
반복 블록 종료(부정)	LEND NOT 릴레이 번호	2	10.2	10.2	11.2	21.9	24.9	14.8	LOOP 실행 안 함(릴레이 OFF)
			9.3	9.3	10.8	17.8	20.4	13.5	LOOP 실행 안 함(릴레이 ON)

4-2-31 문자열 처리 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
문자열 전송	MOV$	3	45.6	45.6	66.0	79.3	72.7	84.3	1문자 전송*1
문자열 연결	+$	4	86.5	86.5	126.0	152.0	137.0	167.8	1문자+1문자*1
왼쪽에서 문자열 가져오기	LEFT$	4	53.0	53.0	77.4	93.6	84.8	94.3	2문자에서 1문자 가져오기*1
오른쪽에서 문자열 가져오기	RGHT$	4	52.2	52.2	76.3	92.1	83.3	94.2	2문자에서 1문자 가져오기*1
임의 위치에서 문자열 가져오기	MID$	5	56.5	56.5	84.6	93.7	84.0	230.2	3문자 중에서 1문자 가져오기*1
문자열 검색	FIND$	4	51.4	51.4	77.5	89.1	96.7	94.1	2문자에서 1문자 검색*1
문자열 길이 검출	LEN$	3	19.8	19.8	28.9	33.8	30.1	33.4	1문자 검출*1
문자열 치환	RPLC$	6	175.1	175.1	258.7	300.7	267.7	479.5	2문자 중 첫 번째 문자를 1문자 치환*1
문자열 삭제	DEL$	5	63.4	63.4	94.2	111.3	99.3	244.6	2문자 중 선두 1문자 삭제*1
문자열 교환	XCHG$	3	60.6	60.6	87.2	105.2	95.3	99.0	1문자와 1문자 교환*1
문자열 클리어	CLR$	2	23.8	23.8	36.0	42.0	36.8	37.8	1문자 클리어*1
문자열 삽입	INS$	5	136.5	136.5	200.6	204.0	208.0	428.9	2문자 중 첫 번째 문자 뒤에 1문자 삽입*1
문자열 비교	=$	4	48.5	48.5	69.8	79.9	68.5	86.2	1문자와 1문자 비교
	<>$								
	<$								
	>$								
	>=$								

*1: 데이터량에 따라 명령 실행 시간이 크게 변화하며 사이클 시간에 영향을 줍니다. 사이클 시간에 대한 영향을 줄이기 위해 '백그라운드 실행'을 수행할 수 있습니다.

4-2-32 태스크 제어 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
태스크 실행 기동	TKON	2	6.4	6.4	10.2	11.2	9.55	3.2	사이클릭 태스크 지정 시
			20.6	20.6	30.1	34.6	34.9	-	추가 태스크 지정 시
태스크 실행 대기	TKOF	2	123.7	123.7	195.8	214.5	183.4	61.3	사이클릭 태스크 지정 시
			23.4	23.4	34.1	39.3	39.6	-	추가 태스크 지정 시

4-2-33 기종 변환용 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
블록 전송	XFERC	4	6.4	6.4	6.5	33.1	31.1	-	1CH 전송
			481.6	481.6	791.6	3056.1	2821.1	-	1000CH 전송
데이터 분배	DISTC	4	3.4	3.4	3.5	19	18.1	-	데이터 분배 동작 시
			5.9	5.9	7.3	39.5	38.5		스택 동작 시
데이터 추출	COLLC	4	3.5	3.5	3.85	24.9	29.7	-	데이터 추출 동작 시
			8	8	9.1	22.1	25.3	-	스택 동작 시
			8.3	8.3	9.6	25.5	31	-	스택 동작 시(1CH분을 FIFO하는 경우)
			2052.3	2052.3	2097.5	8310.1	7821.1	-	스택 동작 시(1000CH분을 FIFO하는 경우)
비트 전송	MOVBC	4	4.5	4.5	4.88	28.1	22.1	-	
비트 카운터	BCNTC	4	4.9	4.9	5	30.6	28.8	-	1CH 카운트
			1252.4	1252.4	1284.4	5814.1	5223.8	-	1000CH 카운트

4-2-34 펑션 블록용 특수 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시간(μs)						실행 조건
			CJ1H CPU6□ H-R	CJ1H CPU6□ H	CJ1G CPU4□ H	CJ1M CPU 12/13/ 22/23	CJ1M CPU 11/21	CJ1G CPU4□	
변수 종류 취득	GETID	4	14	14	22.2	23.4	21.3	-	

4-2-35 펑션 블록 사용 시의 인스턴스 실행 시간(CPU 유니트의 유니트 Ver.3.0 이상)

CS/CJ 시리즈 CPU 유니트의 유니트 Ver.3.0 이상에서 펑션 블록 정의를 작성하고 사용자 프로그램에 인스턴스를 첨부한 경우, 인스턴스 실행이 사이클 시간에 미치는 영향은 다음과 같습니다.

인스턴스가 사이클 시간에 미치는 영향 시간
= 기동 시간 A
+입출력 파라미터 전송 처리 시간 B
+펑션 블록 정의 내 명령 실행 시간 C

A, B, C의 시간은 각각 다음과 같습니다.

내용			CPU 유니트 형식			
			CJ1H-CPU6□H-R	CS1H-CPU6□H CJ1H-CPU6□H	CS1G-CPU4□H CJ1G-CPU4□H	CJ1M-CPU□□
A	기동 시간	입출력 파라미터 전송을 제외한 기동 시간	3.3μs	6.8μs	8.8μs	15.0μs
B	입출력 파라미터 전송 처리 시간 괄호 안은 데이터 형식	1비트(BOOL)의 입출력 변수인 경우	0.24μs	0.4μs	0.7μs	1.0μs
		1워드(INT, UINT, WORD)의 입출력 변수인 경우	0.19μs	0.3μs	0.6μs	0.8μs
		2워드(DINT, UDINT, DWORD, REAL)의 입출력 변수인 경우	0.19μs	0.5μs	0.8μs	1.1μs
		4워드(LINT, ULINT, LWORD, LREAL)의 입출력 변수인 경우	0.38μs	1.0μs	1.6μs	2.2μs
C	펑션 블록 정의 내 명령 실행 시간	명령 처리 시간의 합계(평상시 사용자 프로그램과 동일)				

예) CS1H-CPU63H에서
데이터 형식이 1워드(INT)인 입력 변수 = 3개,
데이터 형식이 1워드(INT)인 출력 변수 = 2개,
펑션 블록 정의 부분의 명령 처리 시간 합계가 10μs인 경우:
하나의 인스턴스 실행 시간= 6.8μs+ (3+2)×0.3μs+ 10μs = 18.3μs입니다.

주: 동일한 펑션 블록 정의를 여러 개 첨부한 경우(인스턴스가 여러 개인 경우) 인스턴스 수에 따라 실행 시간이 늘어납니다.

참 고 기존 기종에서 CS 시리즈로 프로그램 용량을 환산하는 기준
기존 기종(SYSMACα 또는 CVM1/CV 시리즈)의 프로그램 용량(단위: 워드)을 CS 시리즈의 프로그램 용량(단위: 스텝)으로 환산하는 경우, 그 기준은 다음과 같습니다.
기존 기종(SYSMACα 또는 CVM1/CV 시리즈)의 프로그램 용량(단위: 워드)에서 다음 표의 값(n)을 명령에 가산한 값이 CS 시리즈의 프로그램 용량(단위: 스텝)이 됩니다.

기존 기종의 워드 수 = a워드인 경우 → CS1에서의 스텝 수 = (a + n)스텝			
명령 분류	동작 옵션	SYSMACα →CS1로 변환할 때의 n의 값	CVM1/CV→CS1로 변환할 때의 n의 값
기본 명령	없음	OUT 명령, SET 명령, RSET 명령, KEEP 명령의 경우: -1 기타 명령: 0	0
	기동 미분 지원	없음	+1
	매회 리프레시 지원	없음	0
	기동 미분 및 매회 리프레시 지원	없음	+2
응용 명령	없음	0	-1
	기동 미분 지원	+1	0
	매회 리프레시 지원	없음	+3
	기동 미분 및 매회 리프레시 지원	없음	+4

예를 들어, OUT 명령의 경우(주소 0~25515 지정 시), SYSMAC 시리즈에서는 1명령당 2워드이고 CS 시리즈에서는 거기에서 1을 빼므로 1명령당 1스텝이 됩니다.
예를 들어, 매회 리프레시형 전송 명령인 !MOV의 경우 CVM1/CV 시리즈에서 1명령당 4워드이므로 거기에 3을 더하여 7스텝이 됩니다.

참 고 **펑션 블록 사용 시의 프로그램 스텝 수**
(CPU 유니트의 유니트 Ver.3.0 이상)

CS/CJ 시리즈 CPU 유니트의 유니트 Ver.3.0 이상에서 펑션 블록 정의를 작성하고 사용자 프로그램에 인스턴스를 첨부한 경우, 프로그램 스텝 수는 다음과 같습니다.

스텝 수
= 인스턴스 수×(호출부 용량 m+입출력 파라미터 전송부 용량 n×파라미터 수)
+펑션 블록 정의 내 명령 스텝 수 p(주)

주: 동일 펑션 블록 정의를 여러 개 삽입한 경우(인스턴스가 여러 개인 경우), 2개째 이후의 인스턴스에서는 펑션 블록 정의 내 명령 스텝 수(p)가 소비되지 않습니다. 따라서 위 계산식에서는 펑션 블록 정의 내 명령 스텝 수(p)에 인스턴스 수를 곱하지 않았습니다.

내용			CS/CJ 유니트 Ver.3.0 이상 공통
m	호출부		57스텝
n	입출력 파라미터 전송부 괄호 안은 데이터 형식	1비트(BOOL)의 입출력 변수인 경우	6스텝
		1워드(INT, UINT, WORD)의 입출력 변수인 경우	6스텝
		2워드(DINT, UDINT, DWORD, REAL)의 입출력 변수인 경우	6스텝
		4워드(LINT, ULINT, LWORD, LREAL)의 입출력 변수인 경우	12스텝
p	펑션 블록 정의 내 명령 스텝 수	명령 스텝 수의 합계(평상시 사용자 프로그램과 동일)에 27스텝을 가산한 스텝 수	

예)
데이터 형식이 1워드(INT)인 입력 변수 = 5개,
데이터 형식이 1워드(INT)인 출력 변수 = 5개,
펑션 블록 정의 부분의 스텝 수가 100스텝인 경우:
1개 인스턴스의 스텝 수= 57+(5+5)×6스텝+100스텝 + 27스텝 = 244스텝입니다.

4-3 CS 시리즈 명령 처리 시간/스텝 수 목록

CS 시리즈(CS1D CPU 유니트 제외 *1)의 각 명령어에 대한 처리 시간은 다음과 같습니다.

*1: CS1D CPU 유니트의 명령 처리 시간에 대해서는 'CS1D 듀플렉스 시스템 사용 설명서 설치편'(SBCA-318)을 참조하십시오.

하나의 사용자 프로그램 전체(주)에서 각 명령어의 처리 시간을 합한 시간이 사이클 시간 중 '연산 처리 시간'입니다.

주: 정확하게는 사이클릭 태스크 중 실행 가능한 상태의 태스크와 인터럽트 조건이 성립하는 인터럽트 태스크에 할당된 프로그램입니다.

명령어의 처리 시간은 CPU 유니트의 형식(CS1H-CPU6□/CS1H-CPU6□H/CS1G-CPU4□/CS1G-CPU4□H)에 따라 다르며, 명령 실행 시의 조건에 따라서도 달라집니다.
또한 명령을 실행하지 않을 때도 처리 시간이 걸리므로 주의하십시오.
동시에 명령 길이(스텝)를 나타내는데, 이는 피연산자를 포함한 각 명령어가 사용자 프로그램 영역을 얼마나 점유하는지 나타냅니다. 명령어에 따라 1명령어당 최소 1스텝에서 최대 7스텝의 용량이 필요합니다.
또한 스텝이 명령어의 수를 나타내는 것은 아니므로 주의하십시오.

주1 : ・기존 기종(C/CV 시리즈)에서는 프로그램 용량을 '워드' 단위로 표시하는데 비해 CS 시리즈에서는 '스텝' 단위로 표시합니다. 1스텝은 기본적으로 기존의 1워드에 해당합니다.
단, 1명령어당 점유 용량이 기존 기종과 CS 시리즈에서 서로 다르기 때문에 기존 기종의 사용자 프로그램을 CS 시리즈의 사용자 프로그램으로 변환한 경우, 정확히는 1워드 = 1스텝의 계산으로는 프로그램 용량에 약간 변화가 생깁니다. 자세한 내용은 4-61페이지의 '참고 기존 기종에서 CS 시리즈로 프로그램 용량을 환산하는 기준'을 참고하십시오.
・일부 명령어를 제외하고 미분 실행(기본 명령은 '↑ ↓', 응용 명령의 경우는 '@%') 옵션이 지원됩니다. 그 경우의 처리 시간에 다음 시간을 가산하십시오.

조건	CS1-H CPU 유니트		CS1 CPU 유니트	
	CPU6□H(μs)	CPU4□H(μs)	CPU6□(μs)	CPU4□(μs)
미분(↑ ↓)을 지원하는 경우	+0.24	+0.32	+0.41	+0.45
미분(@%)을 지원하는 경우	+0.24	+0.32	+0.29	+0.33

주 2: 명령 비실행 시의 처리 시간의 기준은 다음과 같습니다.

CS1-H CPU 유니트		CS1 CPU 유니트	
CPU6□H(μs)	CPU4□H(μs)	CPU6□(μs)	CPU4□(μs)
0.018~0.108	0.02~0.12	0.08~0.48	0.12~0.72

4-3-1 시퀀스 입력 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
로드	LD	1	0.02	0.04	0.04	0.08	
	!LD	2	21.16	21.20	21.20	21.24	대상 유니트가 CS 시리즈 유니트인 경우
			45.12	45.14	45.14	45.18	대상 유니트가 C200H 시리즈 유니트인 경우
로드 부정	LD NOT	1	0.02	0.04	0.04	0.08	
	!LD NOT	2	21.16	21.20	21.20	21.24	대상 유니트가 CS 시리즈 유니트인 경우
			45.12	45.14	45.14	45.18	대상 유니트가 C200H 시리즈 유니트인 경우
논리곱	AND	1	0.02	0.04	0.04	0.08	
	!AND	2	21.16	21.20	21.20	21.24	대상 유니트가 CS 시리즈 유니트인 경우
			45.12	45.14	45.14	45.18	대상 유니트가 C200H 시리즈 유니트인 경우
부정 논리곱	AND NOT	1	0.02	0.04	0.04	0.08	
	!AND NOT	2	21.16	21.20	21.20	21.24	대상 유니트가 CS 시리즈 유니트인 경우
			45.12	45.14	45.14	45.18	대상 유니트가 C200H 시리즈 유니트인 경우
논리합	OR	1	0.02	0.04	0.04	0.08	
	!OR	2	21.16	21.20	21.20	21.24	대상 유니트가 CS 시리즈 유니트인 경우
			45.12	45.14	45.14	45.18	대상 유니트가 C200H 시리즈 유니트인 경우
부정 논리합	OR NOT	1	0.02	0.04	0.04	0.08	
	!OR NOT	2	21.16	21.20	21.20	21.24	대상 유니트가 CS 시리즈 유니트인 경우
			45.12	45.14	45.14	45.18	대상 유니트가 C200H 시리즈 유니트인 경우
로드 논리곱	AND LD	1	0.02	0.04	0.04	0.08	
로드 논리합	OR LD	1	0.02	0.04	0.04	0.08	
부정	NOT	1	0.02	0.04	0.04	0.08	
P.F. 기동 미분	UP	3	0.3	0.42	0.46	0.54	
P.F. 종료 미분	DOWN	4	0.3	0.42	0.46	0.54	
LD 비트 테스트	LD TST	4	0.14	0.24	0.25	0.37	
LD 비트 테스트 부정	LD TSTN	4	0.14	0.24	0.25	0.37	
AND 비트 테스트 부정	AND TSTN	4	0.14	0.24	0.25	0.37	
OR 비트 테스트	OR TST	4	0.14	0.24	0.25	0.37	
OR 비트 테스트 부정	OR TSTN	4	0.14	0.24	0.25	0.37	

4-3-2 시퀀스 출력 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
출력	OUT	1	0.02	0.04	0.17	0.21	
	!OUT	2	21.39	21.41	21.54	21.58	대상 유니트가 CS 시리즈 유니트인 경우
			49.32	49.34	49.47	49.51	대상 유니트가 C200H 시리즈 유니트인 경우
부정 출력	OUT NOT	1	0.02	0.04	0.17	0.21	
	!OUT NOT	2	21.39	21.41	21.54	21.58	대상 유니트가 CS 시리즈 유니트인 경우
			49.32	49.34	49.47	49.51	대상 유니트가 C200H 시리즈 유니트인 경우
유지	KEEP	1	0.06	0.08	0.25	0.29	
기동 미분	DIFU	2	0.24	0.40	0.46	0.54	
종료 미분	DIFD	2	0.24	0.40	0.46	0.54	
세트	SET	1	0.02	0.06	0.17	0.21	
	!SET	2	21.39	21.41	21.54	21.58	대상 유니트가 CS 시리즈 유니트인 경우
			49.32	49.34	49.47	49.51	대상 유니트가 C200H 시리즈 유니트인 경우
리셋	RSET	1	0.02	0.06	0.17	0.21	채널 지정
	!RSET	2	21.39	21.41	21.54	21.58	대상 유니트가 CS 시리즈 유니트인 경우
			49.32	49.34	49.47	49.51	대상 유니트가 C200H 시리즈 유니트인 경우
다중 비트 세트	SETA	4	5.8	6.1	7.8	7.8	1비트 세트
			25.7	27.2	38.8	38.8	1000비트 세트
다중 비트 리셋	RSTA	4	5.7	6.1	7.8	7.8	1비트 리셋
			25.8	27.1	38.8	38.8	1000비트 리셋
1비트 세트	SETB	2	0.24	0.34	-	-	
	!SETB	3	21.48	21.88	-	-	
1비트 리셋	RSTB	2	0.24	0.34	-	-	
	!RSTB	3	21.68	21.88	-	-	
1비트 출력	OUTB	2	0.22	0.32	-	-	
	!OUTB	3	21.64	21.84	-	-	

4-3-3 시퀀스 제어 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
종료	END	1	5.5	6.0	4.0	4.0	
기능 없음	NOP	1	0.02	0.04	0.08	0.12	
인터록	IL	1	0.06	0.06	0.12	0.12	
인터록 클리어	ILC	1	0.06	0.06	0.12	0.12	
멀티 인터록(미분 플래그 유지형)	MILH	3	6.1	6.5	-	-	인터록 조건이 성립하지 않을 때(입력 조건 ON)
			7.5	7.9	-	-	인터록 조건이 성립할 때(입력 조건 OFF)
			8.9	9.7	-	-	인터록 중에 다시 인터록 조건이 성립할 때(입력 조건 OFF)
멀티 인터록(미분 플래그 비유지형)	MILR	3	6.1	6.5	-	-	인터록 조건이 성립하지 않을 때(입력 조건 ON)
			7.5	7.9	-	-	인터록 조건이 성립할 때(입력 조건 OFF)
			8.9	9.7	-	-	인터록 중에 다시 인터록 조건이 성립할 때(입력 조건 OFF)
멀티 인터록 클리어	MILC	2	5.0	5.6	-	-	인터록 중이 아닐 때
			5.7	6.2	-	-	인터록 중일 때
점프	JMP	2	0.38	0.48	8.1	8.1	
점프 종료	JME	2	-	-	-	-	
조건부 점프	CJP	2	0.38	0.48	7.4	7.4	점프 조건이 성립할 때(입력 조건 ON)
조건부 부정 점프	CJPN	2	0.38	0.48	8.5	8.5	점프 조건이 성립할 때(입력 조건 OFF)
복수 점프	JMP0	1	0.06	0.06	0.12	0.12	
복수 점프 종료	JME0	1	0.06	0.06	0.12	0.12	
반복 시작	FOR	2	0.12	0.21	0.12	0.21	상수 지정
루프 중단	BREAK	1	0.12	0.12	0.12	0.12	
반복 종료	NEXT	1	0.17	0.17	0.17	0.17	루프를 계속할 때
			0.12	0.12	0.12	0.12	루프를 종료할 때

4-3-4 타이머/카운터 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
100ms 타이머	TIM	3	**0.56**	**0.88**	**0.37**	**0.42**	
	TIMX	3	0.56	0.88	0.37	0.42	
10ms 타이머	TIMH	3	0.88	1.14	0.37	0.42	
	TIMHX	3	0.88	1.14	0.37	0.42	
1ms 타이머	TMHH	3	0.86	1.12	0.37	0.42	
	TMHHX	3	0.86	1.12	0.37	0.42	
적산 타이머	TTIM	3	16.1	17.0	21.4	21.4	
			10.9	11.4	14.8	14.8	리셋할 때
			8.5	8.7	10.7	10.7	인터록할 때
	TTIMX	3	16.1	17.0	21.4	21.4	
			10.9	11.4	14.8	14.8	리셋할 때
			8.5	8.7	10.7	10.7	인터록할 때
장시간 타이머	TIML	4~5	7.6	10.0	12.8	12.8	
			6.2	6.5	7.8	7.8	인터록할 때
	TIMLX	4~5	7.6	10.0	12.8	12.8	
			6.2	6.5	7.8	7.8	인터록할 때
멀티 출력 타이머	MTIM	4	20.9	23.3	26.0	26.0	
			5.6	5.8	7.8	7.8	리셋할 때
	MTIMX	4	20.9	23.3	26.0	26.0	
			5.6	5.8	7.8	7.8	리셋할 때
카운터	CNT	3	0.56	0.88	0.37	0.42	
	CNTX	3	0.56	0.88	0.37	0.42	
가역 카운터	CNTR	3	16.9	19.0	20.9	20.9	
	CNTRX	3	16.9	19.0	20.9	20.9	
타이머/카운터 리셋	CNR	3	9.9	10.6	13.9	13.9	1CH 리셋
			4160	4160	5420	5420	1000CH 리셋
	CNRX	3	9.9	10.6	13.9	13.9	1CH 리셋
			4160	4160	5420	5420	1000CH 리셋

4-3-5 데이터 비교 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
심볼 비교(부호 없음)	=	4	0.10	0.16	0.21	0.37	
	< >						
	<						
	< =						
	>						
	> =						
심볼 비교(2워드, 부호 없음)	= L	4~7	0.10	0.16	0.29	0.54	
	< > L						
	< L						
	< = L						
	> L						
	> = L						
심볼 비교(부호 있음)	= S	4	0.10	0.16	6.50	6.50	
	< > S						
	< S						
	< =						
	> S						
	> = S						
심볼 비교(2워드, 부호 있음)	= SL	4~7	0.10	0.16	6.50	6.50	
	< > SL						
	< SL						
	< = SL						
	> SL						
	> = SL						
시각 비교	= DT	4	25.1	36.4	-	-	
	< > DT	4	25.2	36.4	-	-	
	< DT	4	25.2	36.4	-	-	
	< = DT	4	25.2	36.4	-	-	
	> DT	4	25.1	36.4	-	-	
	> = DT	4	25.2	36.4	-	-	
비교	CMP	3	0.04	0.04	0.17	0.29	
	!CMP	7	42.14	42.14	42.57	42.69	대상 유니트가 CS 시리즈 유니트인 경우
			90.44	90.44	90.67	90.79	대상 유니트가 CS 시리즈 유니트인 경우
2워드 비교	CMPL	3~5	0.08	0.08	0.25	0.46	
부호 있는 BIN 비교	CPS	3	0.08	0.08	6.50	6.50	
	!CPS	7	35.98	35.98	48.9	48.9	대상 유니트가 CS 시리즈 유니트인 경우
			84.18	84.1 8	97.0	97.0	대상 유니트가 C200H 시리즈 유니트인 경우
부호 있는 BIN 2워드 비교	CPSL	3~5	0.08	0.08	6.50	6.50	
테이블 일치	TCMP	4	14.0	15.2	21.9	21.9	
다채널 비교	MCMP	4	20.5	22.8	31.2	31.2	
부호 없는 테이블 간 비교	BCMP	4	21.5	23.7	32.6	32.6	
확장 테이블 간 비교	BCMP2	4	8.4	9.3	-	-	데이터 수 1
			313.0	345.3	-	-	데이터 수 255
영역 비교	ZCP	3	5.3	5.4	-	-	
2워드 영역 비교	ZCPL	3~5	5.5	6.7	-	-	

4-3-6 데이터 전송 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
전송	MOV	3	0.18	0.20	0.25	0.29	
	!MOV	7	21.56	21.60	42.61	42.65	대상 유니트가 CS 시리즈 유니트인 경우
			90.70	90.72	90.77	90.83	대상 유니트가 C200H 시리즈 유니트인 경우
2워드 전송	MOVL	3~4	0.32	0.34	0.42	0.50	
부정 전송	MVN	3	0.18	0.20	0.25	0.29	
부정 2워드 전송	MVNL	3~4	0.32	0.34	0.42	0.50	
비트 전송	MOVB	4	0.24	0.34	7.5	7.5	
자릿수 전송	MOVD	4	0.24	0.34	7.3	7.3	
다중 비트 전송	XFRB	4	10.1	10.8	13.6	13.6	1비트 전송
			186.4	189.8	269.2	269.2	255비트 전송
블록 전송	XFER	4	0.36	0.44	11.2	11.2	1CH 전송
			300.1	380.1	633.5	633.5	1000CH 전송
블록 설정	BSET	4	0.26	0.28	8.5	8.5	1CH 세트
			200.1	220.1	278.3	278.3	1000CH 세트
데이터 교환	XCHG	3	0.40	0.56	0.5	0.7	
데이터 2워드 교환	XCGL	3~4	0.76	1.04	0.9	1.3	
데이터 분배	DIST	4	5.1	5.4	7.0	7.0	
데이터 추출	COLL	4	5.1	5.3	7.1	7.1	
인덱스 레지스터 설정	MOVR	3	0.08	0.08	0.42	0.50	
인덱스 레지스터 설정	MOVRW	3	0.42	0.50	0.42	0.50	

4-3-7 데이터 이동 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
이동 레지스터	SFT	3	7.4	10.4	10.4	10.4	1CH 이동
			433.2	488.0	763.1	763.1	1000CH 이동
좌우 이동 레지스터	SFTR	4	6.9	7.2	9.6	9.6	1CH 이동
			615.3	680.2	859.6	859.6	1000CH 이동
비동기 이동 레지스터	ASFT	4	6.2	6.4	7.7	7.7	1CH 이동
			1220	1220	2010	2010	1000CH 이동*1
워드 이동	WSFT	4	4.5	4.7	7.8	7.8	1CH 이동
			171.5	171.7	781.7	781.7	1000CH 이동
1비트 좌측 이동	ASL	2	0.22	0.32	0.29	0.37	
1비트 좌측 이동(2워드)	ASLL	2	0.40	0.56	0.50	0.67	
1비트 우측 이동	ASR	2	0.22	0.32	0.29	0.37	
1비트 우측 이동(2워드)	ASRL	2	0.40	0.56	0.50	0.67	
CY가 있는 1비트 좌측 회전	ROL	2	0.22	0.32	0.29	0.37	
CY가 있는 1비트 좌측 회전(2워드)	ROLL	2	0.40	0.56	0.50	0.67	
CY가 없는 1비트 좌측 회전	RLNC	2	0.22	0.32	0.29	0.37	
CY가 없는 1비트 좌측 회전(2워드)	RLNL	2	0.40	0.56	0.50	0.67	
CY가 있는 1비트 우측 회전	ROR	2	0.22	0.32	0.29	0.37	
CY가 있는 1비트 우측 회전(2워드)	RORL	2	0.40	0.56	0.50	0.67	
CY가 없는 1비트 우측 회전	RRNC	2	0.22	0.32	0.29	0.37	
CY가 없는 1비트 우측 회전(2워드)	RRNL	2	0.40	0.56	0.50	0.67	
1자리 좌측 이동	SLD	3	5.9	6.1	8.2	8.2	1CH 이동
			561.1	626.3	760.7	760.7	1000CH 이동
1자리 우측 이동	SRD	3	6.9	7.1	8.7	8.7	1CH 이동
			760.5	895.5	1070	1070	1000CH 이동
N비트 데이터 좌측 이동	NSFL	4	7.5	8.3	10.5	10.5	1비트 이동
			40.3	45.4	55.5	55.5	1000비트 이동
N비트 데이터 우측 이동	NSFR	4	7.5	8.3	10.5	10.5	1비트 이동
			50.5	55.3	69.3	69.3	1000비트 이동
N비트 좌측 이동	NASL	3	0.22	0.32	0.29	0.37	
N비트 2워드 좌측 이동	NSLL	3	0.40	0.56	0.50	0.67	
N비트 우측 이동	NASR	3	0.22	0.32	0.29	0.37	
N비트 2워드 우측 이동	NSRL	3	0.40	0.56	0.50	0.67	

*1: 데이터량에 따라 명령 실행 시간이 크게 변화하며 사이클 시간에 영향을 줍니다. 사이클 시간에 대한 영향을 줄이기 위해 '백그라운드 실행'을 수행할 수 있습니다.

4-3-8 증가/감소 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
BIN 증가	++	2	0.22	0.32	0.29	0.37	
BIN 2워드 증가	++L	2	0.40	0.56	0.50	0.67	
BIN 감소	--	2	0.22	0.32	0.29	0.37	
BIN 2워드 감소	--L	2	0.40	0.56	0.50	0.67	
BCD 증가	++B	2	6.4	4.5	7.4	7.4	
BCD 2워드 증가	++BL	2	5.6	4.9	6.1	6.1	
BCD 감소	--B	2	6.3	4.6	7.2	7.2	
BCD 2워드 감소	--BL	2	5.3	4.7	7.1	7.1	

4-3-9 사칙연산 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
부호 있음, CY 없는 BIN 가산	+	4	0.18	0.20	0.25	0.37	
부호 있음, CY 없는 BIN 2워드 가산	+L	4~6	0.32	0.34	0.42	0.54	
부호 및 CY 있는 BIN 가산	+C	4	0.18	0.20	0.25	0.37	
부호 및 CY 있는 BIN 2워드 가산	+CL	4~6	0.32	0.34	0.42	0.54	
CY 없는 BCD 가산	+B	4	8.2	8.4	14.0	14.0	
CY 없는 BCD 2워드 가산	+BL	4~6	13.3	14.5	19.0	19.0	
CY 있는 BCD 가산	+BC	4	8.9	9.1	14.5	14.5	
CY 있는 BCD 2워드 가산	+BCL	4~6	13.8	15.0	19.6	19.6	
부호 있음, CY 없는 BIN 감산	-	4	0.18	0.20	0.25	0.37	
부호 있음, CY 없는 BIN 2워드 감산	-L	4~6	0.32	0.34	0.42	0.54	
부호 및 CY 있는 BIN 감산	-C	4	0.18	0.20	0.25	0.37	
부호 및 CY 있는 BIN 2워드 감산	-CL	4~6	0.32	0.34	0.42	0.54	
CY 없는 BCD 감산	-B	4	8.0	8.2	13.1	13.1	
CY 없는 BCD 2워드 감산	-BL	4~6	12.8	14.0	18.2	18.2	
CY 있는 BCD 감산	-BC	4	8.5	8.6	13.8	13.8	
CY 있는 BCD 2워드 감산	-BCL	4~6	13.4	14.7	18.8	18.8	
부호 있는 BIN 승산	*	4	0.38	0.40	0.50	0.58	
부호 있는 BIN 2워드 승산	*L	4~6	7.23	8.45	11.19	11.19	
부호 없는 BIN 승산	*U	4	0.38	0.40	0.50	0.58	
부호 없는 BIN 2워드 승산	*UL	4~6	7.1	8.3	10.63	10.63	
BCD 승산	*B	4	9.0	9.2	12.8	12.8	
BCD 2워드 승산	*BL	4~6	23.0	24.2	35.2	35.2	
부호 있는 BIN 제산	/	4	0.40	0.42	0.75	0.83	
부호 있는 BIN 2워드 제산	/L	4~6	7.2	8.4	9.8	9.8	
부호 없는 BIN 제산	/U	4	0.40	0.42	0.75	0.83	
부호 없는 BIN 2워드 제산	/UL	4~6	6.9	8.1	9.1	9.1	
BCD 제산	/B	4	8.6	8.8	15.9	15.9	
BCD 2워드 제산	/BL	4~6	17.7	18.9	26.2	26.2	

4-3-10 데이터 변환 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
BCD→BIN 변환	BIN	3	0.22	0.24	0.25	0.29	
BCD→BIN 2워드 변환	BINL	3~4	6.5	6.8	9.1	9.1	
BIN→BCD 변환	BCD	3	0.24	0.26	8.3	8.3	
BIN→BCD 2워드 변환	BCDL	3~4	6.7	7.0	9.2	9.2	
2의 보수 변환	NEG	3	0.18	0.20	0.25	0.29	
2의 보수 2워드 변환	NEGL	3~4	0.32	0.34	0.42	0.50	
부호 확장	SIGN	3	0.32	0.34	0.42	0.50	
4→16/8→256 디코더	MLPX	4	0.32	0.42	8.8	8.8	4→16 디코드 시/자릿수 1자리
			0.98	1.20	12.8	12.8	4→16 디코드 시/자릿수 4자리
			3.30	4.00	20.3	20.3	8→256 디코드 시/자릿수 1자리
			6.50	7.90	33.4	33.4	8→256 디코드 시/자릿수 4자리
16→4/256→8 인코더	DMPX	4	7.5	7.9	10.4	10.4	16→4 디코드 시/자릿수 1자리
			49.6	50.2	59.1	59.1	16→4 디코드 시/자릿수 4자리
			18.2	18.6	23.6	23.6	256→8 디코드 시/자릿수 1자리
			55.1	57.4	92.5	92.5	256→8 디코드 시/자릿수 2자리
ASCII 코드 변환	ASC	4	6.8	7.1	9.7	9.7	1자리 ASCII 코드 변환
			11.2	11.7	15.1	15.1	4자리 ASCII 코드 변환
ASCII→HEX 변환	HEX	4	7.1	7.4	10.1	10.1	1자리 변환
비트 열→비트 행 변환	LINE	4	19.0	23.1	29.1	29.1	
비트 행→비트 열 변환	COLM	4	23.2	27.5	37.3	37.3	
부호 있는 BCD→BIN 변환	BINS	4	8.0	8.3	12.1	12.1	데이터 형식 지정 번호0
			8.0	8.3	12.1	12.1	데이터 형식 지정 번호1
			8.3	8.6	12.7	12.7	데이터 형식 지정 번호2
			8.5	8.8	13.0	13.0	데이터 형식 지정 번호3
부호 있는 BCD→BIN 2워드 변환	BISL	4~5	9.2	9.6	13.6	13.6	데이터 형식 지정 번호0
			9.2	9.6	13.7	13.7	데이터 형식 지정 번호1
			9.5	9.9	14.2	14.2	데이터 형식 지정 번호2
			9.6	10.0	14.4	14.4	데이터 형식 지정 번호3
부호 있는 BIN→BCD 변환	BCDS	4	6.6	6.9	10.6	10.6	데이터 형식 지정 번호0
			6.7	7.0	10.8	10.8	데이터 형식 지정 번호1
			6.8	7.1	10.9	10.9	데이터 형식 지정 번호2
			7.2	7.5	11.5	11.5	데이터 형식 지정 번호3
부호 있는 BIN→BCD 2워드 변환	BDSL	4~5	8.1	8.4	11.6	11.6	데이터 형식 지정 번호0
			8.2	8.6	11.8	11.8	데이터 형식 지정 번호1
			8.3	8.7	12.0	12.0	데이터 형식 지정 번호2
			8.8	9.2	12.5	12.5	데이터 형식 지정 번호3
그레이 코드 변환	GRY	4	46.9	72.1	-	-	8비트 BIN
			49.6	75.2	-	-	8비트 BCD
			57.7	87.7	-	-	8비트 각도
			61.8	96.7	-	-	15비트 BIN
			64.5	99.6	-	-	15비트 BCD
			72.8	112.4	-	-	15비트 각도
			52.3	87.2	-	-	360° BIN
			55.1	90.4	-	-	360° BCD
			64.8	98.5	-	-	360° 각도
숫자 4자리→ASCII 변환	STR4	3	13.79	20.24	-	-	
숫자 8자리→ASCII 변환	STR8	3~4	18.82	27.44	-	-	
숫자 16자리→ASCII 변환	STR16	3	30.54	44.41	-	-	
ASCII→숫자 4자리 변환	NUM4	3~4	18.46	27.27	-	-	
ASCII→숫자 8자리 변환	NUM8	3	27.27	40.29	-	-	
ASCII→숫자 16자리 변환	NUM16	3	52.31	78.25	-	-	

4-3-11 논리 연산 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
워드 논리곱	ANDW	4	0.18	0.20	0.25	0.37	
워드 논리곱(2워드)	ANDL	4~6	0.32	0.34	0.42	0.54	
워드 논리합	ORW	4	0.22	0.32	0.25	0.37	
워드 논리합(2워드)	ORWL	4~6	0.32	0.34	0.42	0.54	
워드 배타적 논리합	XORW	4	0.22	0.32	0.25	0.37	
워드 배타적 논리합(2워드)	XORL	4~6	0.32	0.34	0.42	0.54	
워드 배타적 논리합 부정	XNRW	4	0.22	0.32	0.25	0.37	
워드 배타적 논리합 부정(2워드)	XNRL	4~6	0.32	0.34	0.42	0.54	
비트 반전	COM	2	0.22	0.32	0.29	0.37	
비트 2워드 반전	COML	2	0.40	0.56	0.50	0.67	

4-3-12 특수 연산 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
BIN 제곱근 연산	ROTB	3	49.6	50.0	530.7	530.7	
BCD 제곱근 연산	ROOT	3	13.7	13.9	514.5	514.5	
숫자 변환	APR	4	6.7	6.9	32.3	32.3	SIN, COS 계산 지정
			17.2	18.4	78.3	78.3	꺾은선 근사 계산 지정
부동 소수점 제산(BCD)	FDIV	4	116.6	176.6	176.6	176.6	
비트 카운터	BCNT	4	0.3	0.38	22.1	22.1	1CH 카운트

4-3-13 부동 소수점 변환 및 연산 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
부동 소수점→16비트 BIN 변환	FIX	3~4	10.6	10.8	14.5	14.5	
부동 소수점→32비트 BIN 변환	FIXL	3~4	10.8	11.0	14.6	14.6	
16비트 BIN→부동 소수점 변환	FLT	3~4	8.3	8.5	11.1	11.1	
32비트 BIN→부동 소수점 변환	FLTL	3~4	8.3	8.5	10.8	10.8	
부동 소수점 가산	+F	4~6	8.0	9.2	10.2	10.2	
부동 소수점 감산	-F	4~6	8.0	9.2	10.3	10.3	
부동 소수점 제산	/F	4~6	8.7	9.9	12.0	12.0	
부동 소수점 승산	*F	4~6	8.0	9.2	10.5	10.5	
각도→라디안 변환	RAD	3~4	10.1	10.2	14.9	14.9	
라디안→각도 변환	DEG	3~4	9.9	10.1	14.8	14.8	
SIN 연산	SIN	3~4	42.0	42.2	61.1	61.1	
COS 연산	COS	3~4	31.5	31.8	44.1	44.1	
TAN 연산	TAN	3~4	16.3	16.6	22.6	22.6	
SIN^{-1} 연산	ASIN	3~4	17.6	17.9	24.1	24.1	
COS^{-1} 연산	ACOS	3~4	20.4	20.7	28.0	28.0	
TAN^{-1} 연산	ATAN	3~4	16.1	16.4	16.4	16.4	
제곱근 연산	SQRT	3~4	19.0	19.3	28.1	28.1	
지수 연산	EXP	3~4	65.9	66.2	96.7	96.7	
로그 연산	LOG	3~4	12.8	13.1	17.4	17.4	
거듭제곱 연산	PWR	4~6	125.4	126.0	181.7	181.7	
단정도 부동 소수점 데이터 비교	=F	4~6	6.6	8.3	-	-	
	<>F						
	<F						
	<=F						
	>F						
	>=F						
부동 소수점<단>→문자열 변환	FSTR	4	48.5	48.9	-	-	
문자열→부동 소수점<단> 변환	FVAL	3	21.1	21.3	-	-	

4-3-14 배정도 부동 소수점 변환 및 연산 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
배정도 부동 소수점 데이터 비교	=D	4	8.5	10.3	-	-	
	<>D						
	<D						
	<=D						
	>D						
	>=D						
부동 소수점→16비트 BIN 변환<배>	FIXD	3	11.7	12.1	-	-	
부동 소수점→32비트 BIN 변환<배>	FIXLD	3	11.6	12.1	-	-	
16비트 BIN→부동 소수점 변환<배>	DBL	3	9.9	10.0	-	-	
32비트 BIN→부동 소수점 변환<배>	DBLL	3	9.8	10.0	-	-	
부동 소수점 가산<배>	+D	4	11.2	11.9	-	-	
부동 소수점 감산<배>	-D	4	11.2	11.9	-	-	
부동 소수점 승산<배>	*D	4	12.0	12.7	-	-	
부동 소수점 제산<배>	/D	4	23.5	24.2	-	-	
각도→라디안 변환<배>	RADD	3	27.4	27.8	-	-	
라디안→각도 변환<배>	DEGD	3	11.2	11.9	-	-	
SIN 연산<배>	SIND	3	45.4	45.8	-	-	
COS 연산<배>	COSD	3	43.0	43.4	-	-	
TAN 연산<배>	TAND	3	20.1	20.5	-	-	
SIN^{-1} 연산<배>	ASIND	3	21.5	21.9	-	-	
COS^{-1} 연산<배>	ACOSD	3	24.7	25.1	-	-	
TAN^{-1} 연산<배>	ATAND	3	19.3	19.7	-	-	
제곱근 연산<배>	SQRTD	3	47.4	47.9	-	-	
지수 연산<배>	EXPD	3	121.0	121.4	-	-	
로그 연산<배>	LOGD	3	16.0	16.4	-	-	
거듭제곱 연산<배>	PWRD	4	223.9	224.2	-	-	

4-3-15 테이블 데이터 처리 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
스택 영역 설정	SSET	3	8.0	8.3	8.5	8.5	스택 영역에 5CH 설정
			231.6	251.8	276.8	276.8	스택 영역에 1000CH 설정
스택 데이터 저장	PUSH	3	6.5	8.6	9.1	9.1	
선입선출	FIFO	3	6.9	8.9	10.6	10.6	스택 영역에 5CH 설정
			352.6	434.3	1130	1130	스택 영역에 1000CH 설정
후입선출	LIFO	3	7.0	9.0	9.9	9.9	
테이블 영역 선언	DIM	5	15.2	21.6	142.1	142.1	
레코드 위치 설정	SETR	4	5.4	5.9	7.0	7.0	
레코드 위치 읽기	GETR	4	7.8	8.4	11.0	11.0	
데이터 검색	SRCH	4	15.5	19.5	19.5	19.5	1CH 검색
			2420	3340	3340	3340	1000CH 검색*1
바이트 스와핑	SWAP	3	12.2	13.6	13.6	13.6	1CH 교체
			1940	2820	2820	2820	1000CH 교체*1
최대값 검색	MAX	4~5	19.2	24.9	24.9	24.9	1CH 검색
			2390	3360	3360s	3360	1000CH 검색*1
최소값 검색	MIN	4~5	19.2	25.3	25.3	25.3	1CH 검색
			2390	3330	3330	3330	1000CH 검색*1
SUM 값 산출	SUM	4~5	28.2	38.5	38.5	38.3	1CH 산출
			1420	1950	1950	1950	1000CH 산출*1
FCS 값 산출	FCS	4~5	20.0	28.3	28.3	28.3	테이블 길이 1워드
			1650	2480	2480	2480	테이블 길이 1000워드*1
스택 데이터 수 출력	SNUM	3	6.0	6.3	-	-	
스택 데이터 참조	SREAD	4	8.0	8.4	-	-	
스택 데이터 갱신	SWRIT	4	7.2	7.6	-	-	
스택 데이터 삽입	SINS	4	7.8	9.9	-	-	
스택 데이터 삽입			354.0	434.8	-	-	테이블 길이 1000워드
스택 데이터 삭제	SDEL	4	8.6	10.6	-	-	
스택 데이터 삭제			354.0	436.0	-	-	테이블 길이 1000워드

*1: 데이터량에 따라 명령 실행 시간이 크게 변화하며 사이클 시간에 영향을 줍니다. 사이클 시간에 대한 영향을 줄이기 위해 '백그라운드 실행'을 수행할 수 있습니다.

4-3-16 데이터 제어 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
PID 연산	PID	4	436.2	678.2	678.2	678.2	처음 실행할 때
			332.3	474.9	474.9	474.9	입력 ON이고 샘플링할 때
			97.3	141.3	141.3	141.3	입력 ON이고 샘플링하지 않을 때
자동 조정 가능한 PID 연산	PIDAT	4	446.3	712.5	-	-	처음 실행할 때
			339.4	533.9	-	-	입력 ON이고 샘플링할 때
			100.7	147.1	-	-	입력 ON이고 샘플링하지 않을 때
			189.2	281.6	-	-	AT에서 처음 실행할 때
			535.2	709.8	-	-	AT에서 샘플링할 때
상하한 리미트 제어	LMT	4~5	16.1	22.1	22.1	22.1	
데드 밴드 제어	BAND	4~5	17.0	22.5	22.5	22.5	
데드 존 제어	ZONE	4~5	15.4	20.5	20.5	20.5	
시분할 비례 출력	TPO	4	10.6	14.8	-	-	입력 OFF
			54.5	82.0	-	-	입력 ON 및 듀티 지정 또는 출력 리미트 무효
			61.0	91.9	-	-	입력 ON 및 조작량 지정과 출력 리미트 유효
스케일링	SCL	4	37.1	53.0	56.8	56.8	
스케일링 2	SCL2	4	28.5	40.2	50.7	50.7	
스케일링 3	SCL3	4	33.4	47.0	57.7	57.7	
데이터 평균화	AVG	4	36.3	52.6	53.1	53.1	1회 평균
			291.0	419.9	419.9	419.9	64회 평균

4-3-17 서브루틴 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
서브루틴 호출	SBS	2	1.26	1.96	17.00	17.00	
서브루틴 진입	SBN	2	-	-	-	-	
서브루틴 반환	RET	1	0.86	1.60	20.60	20.60	
매크로	MCRO	4	23.3	23.3	23.3	23.3	
글로벌 서브루틴 진입	GSBN	2	-	-	-	-	
글로벌 서브루틴 반환	GRET	1	1.26	1.96	-	-	
글로벌 서브루틴 호출	GSBS	2	0.86	1.60	-	-	

4-3-18 인터럽트 제어 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
인터럽트 마스크 세트	MSKS	3	25.6	38.4	39.5	39.5	
인터럽트 마스크 읽기	MSKR	3	11.9	11.9	11.9	11.9	
인터럽트 해제	CLI	3	27.4	41.3	41.3	41.3	
인터럽트 태스크 실행 금지	DI	1	15.0	16.8	16.8	16.8	
인터럽트 태스크 실행 금지 해제	EI	1	19.5	21.8	21.8	21.8	

4-3-19 공정 진행 제어 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
스텝 래더 영역 정의	STEP	2	17.4	20.7	27.1	27.1	스텝 릴레이가 ON일 때
			11.8	13.7	24.4	24.4	스텝 릴레이가 OFF일 때
스텝 래더 영역 진행	SNXT	2	6.6	7.3	10.0	10.0	

4-3-20 I/O 유니트용 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
I/O 리프레시	IORF	3	15.5	16.4	23.5	23.5	대상 유니트가 CS 시리즈의 입력 1CH 유니트
			17.20	18.40	25.6	25.6	대상 유니트가 CS 시리즈의 출력 1CH 유니트
			58.5	63.2	81.7	81.7	대상 유니트가 C200H 시리즈의 입력 1CH 유니트
			62.6	67.0	86.7	86.7	대상 유니트가 C200H 시리즈의 출력 1CH 유니트
			319.9	320.7	377.5	377.6	대상 유니트가 CS 시리즈의 입력 60CH 유니트
			358.0	354.4	460.1	460.1	대상 유니트가 CS 시리즈의 출력 60CH 유니트
			303.3	343.9	357.1	357.1	대상 유니트가 C200H 시리즈의 입력 10CH 유니트
			348.2	376.6	407.5	407.5	대상 유니트가 C200H 시리즈의 출력 10CH 유니트
7세그먼트 디코더	SDEC	4	6.5	6.9	14.1	14.1	
디지털 스위치	DSW	6	50.7	73.5	-	-	4자리 지정, 데이터 입력값: 0
			51.3	73.5	-	-	8자리 지정, 데이터 입력값: 00
10키 입력	TKY	4	9.7	13.2	-	-	데이터 입력값: 00
			10.7	14.8	-	-	데이터 입력값: FF
16키 입력	HKY	5	50.3	70.9	-	-	데이터 입력값: 00
			50.1	71.2	-	-	데이터 입력값: FF
행렬 입력	MTR	5	47.8	68.1	-	-	데이터 입력값: 00
			48.0	68.0	-	-	데이터 입력값: FF
7세그먼트 표시	7SEG	5	58.1	83.3	-	-	4자리 지정 시
			63.3	90.3	-	-	8자리 지정 시
인텔리전트 I/O 읽기	IORD	4	*1	*1	*1	*1	
인텔리전트 I/O 쓰기	IOWR	4	*1	*1	*1	*1	
CPU 고기능 유니트 매회 I/O 리프레시	DLNK	4	287.8	315.5	-	-	1CH 점유

*1: IORD, IORW 명령은 읽기 상대의 고기능 I/O 유니트에 따라 다릅니다.

4-3-21 시리얼 통신 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
프로토콜 매크로	PMCR	5	100.1	142.1	276.8	276.8	직접 지정
			134.2	189.6	305.9	305.9	피연산자 지정 송신 1CH, 수신 1CH
시리얼 포트 출력	TXD	4	68.5	98.8	98.8	98.8	송신 바이트 수 1바이트
			734.3	1100	1100	1100	송신 바이트 수 256바이트
시리얼 포트 입력	RXD	4	89.6	131.1	131.1	131.1	저장 바이트 수 1바이트
			724.2	1110	1110	1110	저장 바이트 수 256바이트
시리얼 커뮤니케이션 유니트 시리얼 포트 출력	TXDU	4	131.5	202.4	-	-	송신 바이트 수 1바이트
시리얼 커뮤니케이션 유니트 시리얼 포트 입력	RXDU	4	131	200.8	-	-	저장 바이트 수 1바이트
시리얼 통신 설정 변경	STUP	3	341.2	400.0	440.4	440.4	

4-3-22 네트워크 통신용 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
네트워크 송신	SEND	4	84.4	123.9	123.9	123.9	
네트워크 수신	RECV	4	85.4	124.7	124.7	124.7	
명령 송신	CMND	4	106.8	136.8	136.8	136.8	
범용 Explicit 메시지 송신 명령	EXPLT	4	127.6	190.0	-	-	
Explicit 읽기 명령	EGATR	4	123.9	185.0	-	-	
Explicit 쓰기 명령	ESATR	3	110.0	164.4	-	-	
Explicit CPU 유니트 데이터 읽기 명령	ECHRD	4	106.8	158.9	-	-	
Explicit CPU 유니트 데이터 쓰기 명령	ECHWR	4	106.0	158.3	-	-	

4-3-23 파일 메모리용 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
데이터 파일 읽기	FREAD	5	391.4	632.4	684.1	684.1	BIN 형식
							(디렉토리명 + 파일명)이 2문자일 때
데이터 파일 쓰기	FWRIT	5	387.8	627.0	684.7	684.7	BIN 형식
							(디렉토리명 + 파일명)이 2문자일 때
텍스트 파일 쓰기	TWRIT	5	390.1	619.1	-	-	

4-3-24 표시 기능용 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
메시지 표시	MSG	3	10.1	14.2	14.3	14.3	메시지 표시
			8.4	11.3	11.3	11.3	메시지 표시 해제

4-3-25 시계 기능 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
캘린더 가산	CADD	4	38.3	201.9	209.5	209.5	
캘린더 감산	CSUB	4	38.6	170.4	184.1	184.1	
시분초→초 변환	SEC	3	21.4	29.3	35.8	35.8	
초→시분초 변환	HMS	3	22.2	30.9	42.1	42.1	
시계 보정	DATE	2	60.5	87.4	95.9	95.9	

4-3-26 디버그 처리 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
트레이스 메모리 샘플링	TRSM	1	80.4	120.0	120.0	120.0	1점+0CH 샘플링
			848.1	1060	1060	1060	31점+6CH 샘플링

4-3-27 고장 진단 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
운전 계속 고장 진단	FAL	3	15.4	16.7	16.7	16.7	이상 등록
			179.8	244.8	244.8	244.8	이상 해제(최우선)
			432.4	657.1	657.1	657.1	이상 해제(전체 이상)
			161.5	219.4	219.4	219.4	이상 해제(개별)
운전 정지 고장 진단	FALS	3	-	-	-	-	
고장 지점 검출	FPD	4	140.9	202.3	202.3	202.3	접점 주소 출력 시간 감시 중
			163.4	217.6	217.6	217.6	접점 주소 출력 이상을 처음 검출했을 때
			185.2	268.9	268.9	268.9	메시지 문자 출력 시간 감시 중
			207.5	283.6	283.6	283.6	메시지 문자 출력 이상을 처음 검출했을 때

4-3-28 특수 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
세트 캐리	STC	1	0.06	0.06	0.12	0.12	
클리어 캐리	CLC	1	0.06	0.06	0.12	0.12	
확장 DM 뱅크 전환	EMBC	2	14.0	15.1	15.1	15.1	
사이클 시간 감시 시간 설정	WDT	2	15.0	19.7	19.7	19.7	
조건 플래그 저장	CCS	1	8.6	12.5	-	-	
조건 플래그 로드	CCL	1	9.8	13.9	-	-	
CV→CS 주소 변환	FRMCV	3	13.6	19.9	-	-	
CS→CV 주소 변환	TOCV	3	11.9	17.2	-	-	
주변 서비스 금지	IOSP	1	13.9	19.8	-	-	
주변 서비스 금지 해제	IORS	1	63.6	92.3	-	-	

4-3-29 블록 프로그램 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
블록 프로그램	BPRG	2	12.1	13.0	13.0	13.0	
블록 프로그램 종료	BEND	1	9.6	12.3	13.1	13.1	
블록 프로그램 일시 정지	BPPS	2	10.6	12.3	14.9	14.9	
블록 프로그램 재기동	BPRS	2	5.1	5.6	8.3	8.3	
조건부 종료	입력 조건 EXIT	1	10.0	11.3	12.9	12.9	EXIT 실행(입력 조건 ON)
			4.0	4.9	7.3	7.3	EXIT 실행 안 함(입력 조건 OFF)
	EXIT 릴레이 번호	2	6.8	13.5	16.3	16.3	EXIT 실행(릴레이 ON)
			4.7	7.2	10.7	10.7	EXIT 실행 안 함(릴레이 OFF)
조건부 종료(부정)	EXIT NOT 릴레이 번호	2	12.4	14.0	16.8	16.8	EXIT 실행(릴레이 OFF)
			7.1	7.6	11.2	11.2	EXIT 실행 안 함(릴레이 ON)
조건 분기 블록	입력 조건 IF	1	4.6	4.8	7.2	7.2	IF 참(입력 조건 ON)
			6.7	7.3	10.9	10.9	IF 거짓(입력 조건 OFF)
	IF 릴레이 번호	2	6.8	7.2	10.4	10.4	IF 참(릴레이 ON)
			9.0	9.6	14.2	14.2	IF 거짓(릴레이 OFF)
조건 분기 블록(부정)	IF NOT 릴레이 번호	2	7.1	7.6	10.9	10.9	IF 참(릴레이 OFF)
			9.2	10.1	14.7	14.7	IF 거짓(릴레이 ON)
조건 분기 거짓 블록	ELSE	1	6.2	6.7	9.9	9.9	IF 참
			6.8	7.7	11.2	11.2	IF 거짓
조건 분기 블록 종료	IEND	1	6.9	7.7	11.0	11.0	IF 참
			4.4	4.6	7.0	7.0	IF 거짓
1스캔 조건 대기	입력 조건 WAIT	1	12.6	13.7	16.7	16.7	WAIT 실행 안 함(입력 조건 ON)
			3.9	4.1	6.3	6.3	WAIT 실행(입력 조건 OFF)
	WAIT 릴레이 번호	2	12.0	13.4	16.5	16.5	WAIT 실행 안 함(릴레이 ON)
			6.1	6.5	9.6	9.6	WAIT 실행(릴레이 OFF)
1스캔 조건 대기(부정)	WAIT NOT 릴레이 번호	2	12.2	13.8	17.0	17.0	WAIT 실행 안 함(릴레이 OFF)
			6.4	6.9	10.1	10.1	WAIT 실행(릴레이 ON)
100ms 타이머 대기	TIMW	3	22.3	25.2	33.1	33.1	처음 기동할 때
			24.9	27.8	35.7	35.7	평상시 실행할 때
	TIMWX	3	22.3	25.2	33.1	33.1	처음 기동할 때
			24.9	27.8	35.7	35.7	평상시 실행할 때
10ms 타이머 대기	TMHW	3	25.8	27.9	34.1	34.1	처음 기동할 때
			20.6	22.7	28.9	28.9	평상시 실행할 때
	TMHWX	3	25.8	27.9	34.1	34.1	처음 기동할 때
			20.6	22.7	28.9	28.9	평상시 실행할 때
카운터 대기	CNTW	4	17.9	22.6	27.4	27.4	처음 기동할 때
			19.1	23.9	28.7	28.7	평상시 실행할 때
	CNTWX	4	17.9	22.6	27.4	27.4	처음 기동할 때
			19.1	23.9	28.7	28.7	평상시 실행할 때
반복 블록	LOOP	1	7.9	9.1	12.3	12.3	
반복 블록 종료	입력 조건 LEND	1	7.7	8.4	10.9	10.9	LOOP 실행 안 함(입력 조건 ON)
			6.8	8.0	9.8	9.8	LOOP 실행(입력 조건 OFF)
	LEND 릴레이 번호	2	9.9	10.7	14.4	14.4	LOOP 실행 안 함(릴레이 ON)
			8.9	10.3	13.0	13.0	LOOP 실행(릴레이 OFF)
반복 블록 종료(부정)	LEND NOT 릴레이 번호	2	10.2	11.2	14.8	14.8	LOOP 실행 안 함(릴레이 OFF)
			9.3	10.8	13.5	13.5	LOOP 실행 안 함(릴레이 ON)

4-3-30 문자열 처리 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
문자열 전송	MOV$	3	45.6	66.0	84.3	84.3	1문자 전송*1
문자열 연결	+$	4	86.5	126.0	167.8	167.8	1문자+1문자*1
왼쪽에서 문자열 가져오기	LEFT$	4	53.0	77.4	94.3	94.3	2문자에서 1문자 가져오기*1
오른쪽에서 문자열 가져오기	RGHT$	4	52.2	76.3	94.2	94.2	2문자에서 1문자 가져오기*1
임의 위치에서 문자열 가져오기	MID$	5	56.5	84.6	230.2	230.2	3문자 중에서 1문자 가져오기*1
문자열 검색	FIND$	4	51.4	77.5	94.1	94.1	2문자에서 1문자 검색*1
문자열 길이 검출	LEN$	3	19.8	28.9	33.4	33.4	1문자 검출*1
문자열 치환	RPLC$	6	175.1	258.7	479.5	479.5	2문자 중 첫 번째 문자를 1문자 치환*1
문자열 삭제	DEL$	5	63.4	94.2	244.6	244.6	2문자 중 선두 1문자 삭제*1
문자열 교환	XCHG$	3	60.6	87.2	99.0	99.0	1문자와 1문자 교환*1
문자열 클리어	CLR$	2	23.8	36.0	37.8	37.8	1문자 클리어*1
문자열 삽입	INS$	5	136.5	200.6	428.9	428.9	2문자 중 첫 번째 문자 뒤에 1문자 삽입*1
문자열 비교	=$	4	48.5	69.8	86.2	86.2	1문자와 1문자 비교
	<>$						
	<$						
	>$						
	>=$						

*1: 데이터량에 따라 명령 실행 시간이 크게 변화하며 사이클 시간에 영향을 줍니다. 사이클 시간에 대한 영향을 줄이기 위해 '백그라운드 실행'을 수행할 수 있습니다.

4-3-31 태스크 제어 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
태스크 실행 기동	TKON	2	6.4	10.2	3.2	3.2	사이클릭 태스크 지정 시
			20.6	30.1	-	-	추가 태스크 지정 시
태스크 실행 대기	TKOF	2	123.7	195.8	61.3	61.3	사이클릭 태스크 지정 시
			23.4	34.1	-	-	추가 태스크 지정 시

4-3-32 기종 변환용 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
블록 전송	XFERC	4	6.4	6.5	-	-	1CH 전송
			481.6	791.6	-	-	1000CH 전송
데이터 분배	DISTC	4	3.4	3.5	-	-	데이터 분배 동작 시
			5.9	7.3	-	-	스택 동작 시
데이터 추출	COLLC	4	3.5	3.85	-	-	데이터 추출 동작 시
			8	9.1	-	-	스택 동작 시
			8.3	9.6	-	-	스택 동작 시(1CH분을 FIFO하는 경우)
			2052.3	2097.5	-	-	스택 동작 시(1000CH분을 FIFO하는 경우)
비트 전송	MOVBC	4	4.5	4.88	-	-	
비트 카운터	BCNTC	4	4.9	5	-	-	1CH 카운트
			1252.4	1284.4	-	-	1000CH 카운트

4-3-33 펑션 블록용 특수 명령

명령 이름	니모닉	명령 길이 (스텝)	실행 시(μs)				실행 조건
			CS1H CPU6□H	CS1G CPU4□H	CS1H CPU6□	CS1G CPU4□	
변수 종류 취득	GETID	4	14	22.2	-	-	

4-3-34 펑션 블록 사용 시의 인스턴스 실행 시간(CPU 유니트의 유니트 Ver.3.0 이상)

CS/CJ 시리즈 CPU 유니트의 유니트 Ver.3.0 이상에서 펑션 블록 정의를 작성하고 사용자 프로그램에 인스턴스를 첨부한 경우, 인스턴스 실행이 사이클 시간에 미치는 영향은 다음과 같습니다.

```
인스턴스가 사이클 시간에 미치는 영향 시간
        = 기동 시간 A
          +입출력 파라미터 전송 처리 시간 B
          +펑션 블록 정의 내 명령 실행 시간 C
```

A, B, C의 시간은 각각 다음과 같습니다.

내용			CPU 유니트 형식			
			CJ1H-CPU6□H-R	CS1H-CPU6□H CJ1H-CPU6□H	CS1G-CPU4□H CJ1G-CPU4□H	CJ1M-CPU□□
A	기동 시간	입출력 파라미터 전송을 제외한 기동 시간	3.3μs	6.8μs	8.8μs	15.0μs
B	입출력 파라미터 전송 처리 시간 괄호 안은 데이터 형식	1비트(BOOL)의 입출력 변수인 경우	0.24μs	0.4μs	0.7μs	1.0μs
		1워드(INT, UINT, WORD)의 입출력 변수인 경우	0.19μs	0.3μs	0.6μs	0.8μs
		2워드(DINT, UDINT, DWORD, REAL)의 입출력 변수인 경우	0.19μs	0.5μs	0.8μs	1.1μs
		4워드(LINT, ULINT, LWORD, LREAL)의 입출력 변수인 경우	0.38μs	1.0μs	1.6μs	2.2μs
C	펑션 블록 정의 내 명령 실행 시간	명령 처리 시간의 합계(평상시 사용자 프로그램과 동일)				

예) CS1H-CPU63H에서
데이터 형식이 1워드(INT)인 입력 변수 = 3개,
데이터 형식이 1워드(INT)인 출력 변수 = 2개,
펑션 블록 정의 부분의 명령 처리 시간 합계가 10μs인 경우:
하나의 인스턴스 실행 시간= 6.8μs+ (3+2)×0.3μs+ 10μs = 18.3μs입니다.

주: 동일한 펑션 블록 정의를 여러 개 첨부한 경우(인스턴스가 여러 개인 경우) 인스턴스 수에 따라 실행 시간이 늘어납니다.

참 고 **기존 기종에서 CS 시리즈로 프로그램 용량을 환산하는 기준**
기존 기종(SYSMACα 또는 CVM1/CV 시리즈)의 프로그램 용량(단위: 워드)을 CS 시리즈의 프로그램 용량(단위: 스텝)으로 환산하는 경우, 그 기준은 다음과 같습니다.
기존 기종(SYSMACα 또는 CVM1/CV 시리즈)의 프로그램 용량(단위: 워드)에서 다음 표의 값(n)을 명령에 가산한 값이 CS 시리즈의 프로그램 용량(단위: 스텝)이 됩니다.

기존 기종의 워드 수 = a워드인 경우 → CS1에서의 스텝 수 = (a + n)스텝			
명령 분류	동작 옵션	SYSMACα →CS1로 변환할 때의 n의 값	CVM1/CV→CS1로 변환할 때의 n의 값
기본 명령	없음	OUT 명령, SET 명령, RSET 명령, KEEP 명령의 경우: -1 기타 명령: 0	0
	기동 미분 지원	없음	+1
	매회 리프레시 지원	없음	0
	기동 미분 및 매회 리프레시 지원	없음	+2
응용 명령	없음	0	-1
	기동 미분 지원	+1	0
	매회 리프레시 지원	없음	+3
	기동 미분 및 매회 리프레시 지원	없음	+4

예를 들어, OUT 명령의 경우(주소 0~25515 지정 시), SYSMAC 시리즈에서는 1명령당 2워드이고 CS 시리즈에서는 거기에서 1을 빼므로 1명령당 1스텝이 됩니다.
예를 들어, 매회 리프레시형 전송 명령인 !MOV의 경우 CVM1/CV 시리즈에서 1명령당 4워드이므로 거기에 3을 더하여 7스텝이 됩니다.

참 고 **펑션 블록 사용 시의 프로그램 스텝 수 (CPU 유니트의 유니트 Ver.3.0 이상)**

CS/CJ 시리즈 CPU 유니트의 유니트 Ver.3.0 이상에서 펑션 블록 정의를 작성하고 사용자 프로그램에 인스턴스를 첨부한 경우, 프로그램 스텝 수는 다음과 같습니다.

스텝 수
= 인스턴스 수×(호출부 용량 m+입출력 파라미터 전송부 용량 n×파라미터 수)
+펑션 블록 정의 내 명령 스텝 수 p(주)

주: 동일 펑션 블록 정의를 여러 개 삽입한 경우(인스턴스가 여러 개인 경우), 2개째 이후의 인스턴스에서는 펑션 블록 정의 내 명령 스텝 수(p)가 소비되지 않습니다. 따라서 위 계산식에서는 펑션 블록 정의 내 명령 스텝 수(p)에 인스턴스 수를 곱하지 않았습니다.

내용			CS/CJ 유니트 Ver.3.0 이상 공통
m	호출부		57스텝
n	입출력 파라미터 전송부 괄호 안은 데이터 형식	1비트(BOOL)의 입출력 변수인 경우	6스텝
		1워드(INT, UINT, WORD)의 입출력 변수인 경우	6스텝
		2워드(DINT, UDINT, DWORD, REAL)의 입출력 변수인 경우	6스텝
		4워드(LINT, ULINT, LWORD, LREAL)의 입출력 변수인 경우	12스텝
p	펑션 블록 정의 내 명령 스텝 수	명령 스텝 수의 합계(평상시 사용자 프로그램과 동일)에 27스텝을 가산한 스텝 수	

예)
데이터 형식이 1워드(INT)인 입력 변수 = 5개,
데이터 형식이 1워드(INT)인 출력 변수 = 5개,
펑션 블록 정의 부분의 스텝 수가 100스텝인 경우:
1개 인스턴스의 스텝 수= 57+(5+5)×6스텝+100스텝 + 27스텝 = 244스텝입니다.

부록

부록 명령 목록(펑션 번호 순서)

펑션 번호	명령 이름	니모닉	기동 지정	종료 지정	매회 리프레시 지정	페이지
없음	로드	LD	@LD	%LD	!LD	3-8
없음	로드 부정	LD NOT	없음	없음	!LD NOT	3-8
없음	논리곱	AND	@AND	%AND	!AND	3-10
없음	부정 논리곱	AND NOT	없음	없음	!AND NOT	3-10
없음	논리합	OR	@OR	%OR	!OR	3-12
없음	부정 논리합	OR NOT	없음	없음	!OR NOT	3-12
없음	로드 논리곱	AND LD	없음	없음	없음	3-14
없음	로드 논리합	OR LD	없음	없음	없음	3-14
없음	출력	OUT	없음	없음	!OUT	3-28
없음	부정 출력	OUT NOT	없음	없음	!OUT NOT	3-28
없음	일시 기억 릴레이	TR	없음	없음	없음	3-30
없음	세트	SET	@SET	%SET	!SET	3-40
없음	리셋	RSET	@RSET	%RSET	!RSET	3-40
없음	타이머	TIM	없음	없음	없음	3-88
없음	카운터	CNT	없음	없음	없음	3-109
000	기능 없음	NOP	없음	없음	없음	3-53
001	종료	END	없음	없음	없음	3-52
002	인터록	IL	없음	없음	없음	3-54
003	인터록 클리어	ILC	없음	없음	없음	3-54
004	점프	JMP	없음	없음	없음	3-66
005	점프 종료	JME	없음	없음	없음	3-66
006	운전 계속 고장 진단	FAL	@FAL	없음	없음	3-868
007	운전 정지 고장 진단	FALS	없음	없음	없음	3-874
008	스텝 래더 영역 정의	STEP	없음	없음	없음	3-639
009	스텝 래더 영역 진행	SNXT	없음	없음	없음	3-639
010	이동 레지스터	SFT	없음	없음	없음	3-176
011	유지	KEEP	없음	없음	!KEEP	3-32
012	가역 카운터	CNTR	없음	없음	없음	3-112
013	기동 미분	DIFU	없음	없음	!DIFU	3-36
014	종료 미분	DIFD	없음	없음	!DIFD	3-38
015	10ms 타이머	TIMH	없음	없음	없음	3-91
016	워드 이동	WSFT	@WSFT	없음	없음	3-182
017	비동기 이동 레지스터	ASFT	@ASFT	없음	없음	3-180
019	다채널 비교	MCMP	@MCMP	없음	없음	3-134
020	부호 없는 비교	CMP	없음	없음	!CMP	3-128
021	전송	MOV	@MOV	없음	!MOV	3-152
022	부정 전송	MVN	@MVN	없음	없음	3-154
023	BCD→BIN 변환	BIN	@BIN	없음	없음	3-256
024	BIN→BCD 변환	BCD	@BCD	없음	없음	3-258
025	1비트 좌측 이동	ASL	@ASL	없음	없음	3-184
026	1비트 우측 이동	ASR	@ASR	없음	없음	3-186
027	CY가 있는 1비트 좌측 회전	ROL	@ROL	없음	없음	3-188
028	CY가 있는 1비트 우측 회전	ROR	@ROR	없음	없음	3-192
029	비트 반전	COM	@COM	없음	없음	3-318
034	워드 논리곱	ANDW	@ANDW	없음	없음	3-310
035	워드 논리합	ORW	@ORW	없음	없음	3-312
036	워드 배타적 논리합	XORW	@XORW	없음	없음	3-314
037	워드 배타적 논리합 부정	XNRW	@XNRW	없음	없음	3-316
040	세트 캐리	STC	@STC	없음	없음	3-888
041	클리어 캐리	CLC	@CLC	없음	없음	3-888
045	트레이스 메모리 샘플링	TRSM	없음	없음	없음	3-864
046	메시지 표시	MSG	@MSG	없음	없음	3-848
058	BCD→BIN 2워드 변환	BINL	@BINL	없음	없음	3-256
059	BIN→BCD 2워드 변환	BCDL	@BCDL	없음	없음	3-258
060	부호 없는 2워드 비교	CMPL	없음	없음	없음	3-128
062	다중 비트 전송	XFRB	@XFRB	없음	없음	3-160
063	비트 열→ 비트 행 변환	LINE	@LINE	없음	없음	3-283
064	비트 행→ 비트 열 변환	COLM	@COLM	없음	없음	3-285
065	시분초→ 초 변환	SEC	@SEC	없음	없음	3-856
066	초→시분초 변환	HMS	@HMS	없음	없음	3-858
067	비트 카운터	BCNT	@BCNT	없음	없음	3-338
068	부호 없는 테이블 간 비교	BCMP	@BCMP	없음	없음	3-138
069	숫자 변환	APR	@APR	없음	없음	3-326
070	블록 전송	XFER	@XFER	없음	없음	3-162
071	블록 설정	BSET	@BSET	없음	없음	3-164
072	BCD 제곱근 연산	ROOT	@ROOT	없음	없음	3-324
073	데이터 교환	XCHG	@XCHG	없음	없음	3-166
074	1자리 좌측 이동	SLD	@SLD	없음	없음	3-196
075	1자리 우측 이동	SRD	@SRD	없음	없음	3-196
076	4→16/8→256 디코더	MLPX	@MLPX	없음	없음	3-265
077	16→4/256→8 인코더	DMPX	@DMPX	없음	없음	3-270
078	7세그먼트 디코더	SDEC	@SDEC	없음	없음	3-659
079	부동 소수점 제산(BCD)	FDIV	@FDIV	없음	없음	3-335
080	데이터 분배	DIST	@DIST	없음	없음	3-168
081	데이터 추출	COLL	@COLL	없음	없음	3-170
082	비트 전송	MOVB	@MOVB	없음	없음	3-156
083	자릿수 전송	MOVD	@MOVD	없음	없음	3-158
084	좌우 이동 레지스터	SFTR	@SFTR	없음	없음	3-178
085	테이블 일치	TCMP	@TCMP	없음	없음	3-136

펑션 번호	명령 이름	니모닉	기동 지정	종료 지정	매회 리프레시 지정	페이지
086	ASCII 코드 변환	ASC	@ASC	없음	없음	3-275
087	적산 타이머	TTIM	없음	없음	없음	3-100
088	영역 비교	ZCP	없음	없음	없음	3-143
090	네트워크 송신	SEND	@SEND	없음	없음	3-774
091	서브루틴 호출	SBS	@SBS	없음	없음	3-561
092	서브루틴 진입	SBN	없음	없음	없음	3-569
093	서브루틴 반환	RET	없음	없음	없음	3-569
094	사이클 시간 감시 시간 설정	WDT	@WDT	없음	없음	3-891
096	블록 프로그램	BPRG	없음	없음	없음	3-908
097	I/O 리프레시	IORF	@IORF	없음	없음	3-650
098	네트워크 수신	RECV	@RECV	없음	없음	3-783
099	매크로	MCRO	@MCRO	없음	없음	3-566
114	부호 있는 BIN 비교	CPS	없음	없음	!CPS	3-131
115	부호 있는 BIN 2워드 비교	CPSL	없음	없음	없음	3-131
116	2워드 영역 비교	ZCPL	없음	없음	없음	3-143
117	부호 있는 영역 비교	ZCPS	없음	없음	없음	3-147
118	부호 있는 2워드 영역 비교	ZCPL	없음	없음	없음	3-147
160	2의 보수 변환	NEG	@NEG	없음	없음	3-261
161	2의 보수 변환(2워드)	NEGL	@NEGL	없음	없음	3-261
162	ASCII→HEX 변환	HEX	@HEX	없음	없음	3-279
174	2워드 데이터 최대값 검색	MAXL	@MAXL	없음	없음	3-467
175	2워드 데이터 최소값 검색	MINL	@MINL	없음	없음	3-474
176	단정도 부동 소수점 데이터 최대값 검색	MAXF	@MAXF	없음	없음	3-470
177	단정도 부동 소수점 데이터 최소값 검색	MINF	@MINF	없음	없음	3-476
178	배정도 부동 소수점 데이터 최대값 검색	MAXD	@MAXD	없음	없음	3-472
179	배정도 부동 소수점 데이터 최소값 검색	MIND	@MIND	없음	없음	3-478
180	FCS 값 산출	FCS	@FCS	없음	없음	3-483
181	데이터 검색	SRCH	@SRCH	없음	없음	3-458
182	최대값 검색	MAX	@MAX	없음	없음	3-463
183	최소값 검색	MIN	@MIN	없음	없음	3-463
184	SUM 값 산출	SUM	@SUM	없음	없음	3-480
190	PID 연산	PID	없음	없음	없음	3-518
191	자동 조정 가능한 PID 연산	PIDAT	없음	없음	없음	3-527
194	스케일링	SCL	@SCL	없음	없음	3-546
195	데이터 평균화	AVG	없음	없음	없음	3-556
203	부호 없는 1W 레코드 정렬	RSORT	@RSORT	없음	없음	3-507
204	부호 없는 2W 레코드 정렬	RSORT2	@RSORT2	없음	없음	3-511
205	부호 없는 4W 레코드 정렬	RSORT4	@RSORT4	없음	없음	3-514

펑션 번호	명령 이름	니모닉	기동 지정	종료 지정	매회 리프레시 지정	페이지
210	디지털 스위치	DSW	없음	없음	없음	3-661
211	10키 입력	TKY	@TKY	없음	없음	3-664
212	16키 입력	HKY	없음	없음	없음	3-667
213	행렬 입력	MTR	없음	없음	없음	3-670
214	7세그먼트 표시	7SEG	없음	없음	없음	3-674
216	아날로그 입력 직접 변환	AIDC	@AIDC	없음	없음	3-677
217	아날로그 출력 직접 변환	AODC	@AODC	없음	없음	3-680
218	NC 유니트 위치 결정 고속 기동	NCDMV	@NCDMV	없음	없음	3-683
219	NC 유니트 위치 결정 시작 트리거	NCDTR	@NCDTR	없음	없음	3-688
222	인텔리전트 I/O 읽기	IORD	@IORD	없음	없음	3-691
223	인텔리전트 I/O 쓰기	IOWR	@IOWR	없음	없음	3-694
225	고기능 I/O 유니트 매회 리프레시	FIORF	@FIORF	없음	없음	3-653
226	CPU 고기능 유니트 매회 I/O 리프레시	DLNK	@DLNK	없음	없음	3-656
235	CPU 유니트 시리얼 커뮤니케이션 보드 시리얼 포트 입력	RXD	@RXD	없음	없음	3-712
236	CPU 유니트 시리얼 커뮤니케이션 보드 시리얼 포트 출력	TXD	@TXD	없음	없음	3-706
237	시리얼 포트 통신 설정 변경	STUP	@STUP	없음	없음	3-745
255	시리얼 커뮤니케이션 유니트 시리얼 포트 입력	RXDU	@RXDU	없음	없음	3-726
256	시리얼 커뮤니케이션 유니트 시리얼 포트 출력	TXDU	@TXDU	없음	없음	3-720
260	프로토콜 매크로	PMCR	@PMCR	없음	없음	3-700
261	시리얼 커뮤니케이션 유니트 시리얼 포트 직접 수신	DRXDU	@DRXDU	없음	없음	3-738
262	시리얼 커뮤니케이션 유니트 시리얼 포트 직접 송신	DTXDU	@DTXDU	없음	없음	3-733
264	프로토콜 매크로 송신 2	PMCR2	@PMCR2	없음	없음	3-748
269	고장 지점 검출	FPD	없음	없음	없음	3-879
281	확장 DM 뱅크 전환	EMBC	@EMBC	없음	없음	3-889

펑션 번호	명령 이름	니모닉	기동 지정	종료 지정	매회 리프레시 지정	페이지
282	조건 플래그 저장	CCS	@CCS	없음	없음	3-893
283	조건 플래그 로드	CCL	@CCL	없음	없음	3-893
284	CV→CS 주소 변환	FRMCV	@FRMCV	없음	없음	3-895
285	CS→CV 주소 변환	TOCV	@TOCV	없음	없음	3-898
286	변수 종류 취득	GETID	@ GETID	없음	없음	3-984
287	주변 서비스 금지	IOSP	@IOSP	없음	없음	3-901
288	주변 서비스 금지 해제	IORS	없음	없음	없음	3-901
300	AND 일치	AND=	없음	없음	없음	3-120
300	LD 일치	LD=	없음	없음	없음	3-120
300	OR 일치	OR=	없음	없음	없음	3-120
301	AND 2워드 일치	AND=L	없음	없음	없음	3-120
301	LD 2워드 일치	LD=L	없음	없음	없음	3-120
301	OR 2워드 일치	OR=L	없음	없음	없음	3-120
302	AND 부호 있는 일치	AND=S	없음	없음	없음	3-120
302	LD 부호 있는 일치	LD =S	없음	없음	없음	3-120
302	OR 부호 있는 일치	OR=S	없음	없음	없음	3-120
303	AND 부호 있는 2워드 일치	AND=SL	없음	없음	없음	3-120
303	LD 부호 있는 2워드 일치	LD=SL	없음	없음	없음	3-120
303	OR 부호 있는 2워드 일치	OR=SL	없음	없음	없음	3-120
305	AND 불일치	AND<>	없음	없음	없음	3-120
305	LD 불일치	LD<>	없음	없음	없음	3-120
305	OR 불일치	OR<>	없음	없음	없음	3-120
306	AND 2워드 불일치	AND<>L	없음	없음	없음	3-120
306	LD 2워드 불일치	LD<>L	없음	없음	없음	3-120
306	OR 2워드 불일치	OR<>L	없음	없음	없음	3-120
307	AND 부호 있는 불일치	AND<>S	없음	없음	없음	3-120
307	LD 부호 있는 불일치	LD<>S	없음	없음	없음	3-120
307	OR 부호 있는 불일치	OR<>S	없음	없음	없음	3-120
308	AND 부호 있는 2워드 불일치	AND<>SL	없음	없음	없음	3-120
308	LD 부호 있는 2워드 불일치	LD<>SL	없음	없음	없음	3-120
308	OR 부호 있는 2워드 불일치	OR<>SL	없음	없음	없음	3-120
310	AND 미만	AND<	없음	없음	없음	3-120
310	LD 미만	LD<	없음	없음	없음	3-120
310	OR 미만	OR<	없음	없음	없음	3-120
311	AND 2워드 미만	AND<L	없음	없음	없음	3-120
311	LD 2워드 미만	LD<L	없음	없음	없음	3-120
311	OR 2워드 미만	OR<L	없음	없음	없음	3-120
312	AND 부호 있는 미만	AND<S	없음	없음	없음	3-120
312	LD 부호 있는 미만	LD<S	없음	없음	없음	3-120
312	OR 부호 있는 미만	OR<S	없음	없음	없음	3-120
313	AND 부호 있는 2워드 미만	AND<SL	없음	없음	없음	3-120
313	LD 부호 있는 2워드 미만	LD<SL	없음	없음	없음	3-120
313	OR 부호 있는 2워드 미만	OR<SL	없음	없음	없음	3-120
315	AND 이하	AND<=	없음	없음	없음	3-120
315	LD 이하	LD<=	없음	없음	없음	3-120
315	OR 이하	OR<=	없음	없음	없음	3-120
316	AND 2워드 이하	AND<=L	없음	없음	없음	3-120
316	LD 2워드 이하	LD<=L	없음	없음	없음	3-120
316	OR 2워드 이하	OR<=L	없음	없음	없음	3-120
317	AND 부호 있는 이하	AND<=S	없음	없음	없음	3-120
317	LD 부호 있는 이하	LD<=S	없음	없음	없음	3-120
317	OR 부호 있는 이하	OR<=S	없음	없음	없음	3-120
318	AND 부호 있는 2워드 이하	AND<=SL	없음	없음	없음	3-120
318	LD 부호 있는 2워드 이하	LD<=SL	없음	없음	없음	3-120
318	OR 부호 있는 2워드 이하	OR<=SL	없음	없음	없음	3-120
320	AND 초과	AND>	없음	없음	없음	3-120
320	LD 초과	LD>	없음	없음	없음	3-120
320	OR 초과	OR>	없음	없음	없음	3-120
321	AND 2워드 초과	AND>L	없음	없음	없음	3-120
321	LD 2워드 초과	LD>L	없음	없음	없음	3-120
321	OR 2워드 초과	OR>L	없음	없음	없음	3-120
322	AND 부호 있는 초과	AND>S	없음	없음	없음	3-120
322	LD 부호 있는 초과	LD>S	없음	없음	없음	3-120
322	OR 부호 있는 초과	OR>S	없음	없음	없음	3-120
323	AND 부호 있는 2워드 초과	AND>SL	없음	없음	없음	3-120
323	LD 부호 있는 2워드 초과	LD>SL	없음	없음	없음	3-120
323	OR 부호 있는 2워드 초과	OR>SL	없음	없음	없음	3-120
325	AND 이상	AND>=	없음	없음	없음	3-120
325	LD 이상	LD>=	없음	없음	없음	3-120
325	OR 이상	OR>=	없음	없음	없음	3-120
326	AND 2워드 이상	AND>=L	없음	없음	없음	3-120
326	LD 2워드 이상	LD>=L	없음	없음	없음	3-120
326	OR 2워드 이상	OR>=L	없음	없음	없음	3-120
327	AND 부호 있는 이상	AND>=S	없음	없음	없음	3-120
327	LD 부호 있는 이상	LD>=S	없음	없음	없음	3-120
327	OR 부호 있는 이상	OR>=S	없음	없음	없음	3-120
328	AND 부호 있는 2워드 이상	AND>=SL	없음	없음	없음	3-120

펑션 번호	명령 이름	니모닉	기동 지정	종료 지정	매회 리프레시 지정	페이지
328	LD 부호 있는 2워드 이상	LD>=SL	없음	없음	없음	3-120
328	OR 부호 있는 2워드 이상	OR>=SL	없음	없음	없음	3-120
329	AND 단정도 부동 소수점 일치	AND=F	없음	없음	없음	3-375
329	LD 단정도 부동 소수점 일치	LD=F	없음	없음	없음	3-375
329	OR 단정도 부동 소수점 일치	OR=F	없음	없음	없음	3-375
330	AND 단정도 부동 소수점 불일치	AND<>F	없음	없음	없음	3-375
330	LD 단정도 부동 소수점 불일치	LD<>F	없음	없음	없음	3-375
330	OR 단정도 부동 소수점 불일치	OR<>F	없음	없음	없음	3-375
331	AND 단정도 부동 소수점 미만	AND<F	없음	없음	없음	3-375
331	LD 단정도 부동 소수점 미만	LD<F	없음	없음	없음	3-375
331	OR 단정도 부동 소수점 미만	OR<F	없음	없음	없음	3-375
332	AND 단정도 부동 소수점 이하	AND<=	없음	없음	없음	3-375
332	LD 단정도 부동 소수점 이하	LD<=F	없음	없음	없음	3-375
332	OR 단정도 부동 소수점 이하	OR<=F	없음	없음	없음	3-375
333	AND 단정도 부동 소수점 초과	AND>F	없음	없음	없음	3-375
333	LD 단정도 부동 소수점 초과	LD>F	없음	없음	없음	3-375
333	OR 단정도 부동 소수점 초과	OR>F	없음	없음	없음	3-375
334	AND 단정도 부동 소수점 이상	AND>=F	없음	없음	없음	3-375
334	LD 단정도 부동 소수점 이상	LD>=F	없음	없음	없음	3-375
334	OR 단정도 부동 소수점 이상	OR>=F	없음	없음	없음	3-375
335	AND 배정도 부동 소수점 일치	AND=D	없음	없음	없음	3-421
335	LD 배정도 부동 소수점 일치	LD=D	없음	없음	없음	3-421
335	OR 배정도 부동 소수점 일치	OR=D	없음	없음	없음	3-421
336	AND 배정도 부동 소수점 불일치	AND<>D	없음	없음	없음	3-421
336	LD 배정도 부동 소수점 불일치	LD<>D	없음	없음	없음	3-421
336	OR 배정도 부동 소수점 불일치	OR<>D	없음	없음	없음	3-421
337	AND 배정도 부동 소수점 미만	AND<D	없음	없음	없음	3-421
337	LD 배정도 부동 소수점 미만	LD<D	없음	없음	없음	3-421
337	OR 배정도 부동 소수점 미만	OR<D	없음	없음	없음	3-421
338	AND 배정도 부동 소수점 이하	AND<=	없음	없음	없음	3-421
338	LD 배정도 부동 소수점 이하	LD<=D	없음	없음	없음	3-421
338	OR 배정도 부동 소수점 이하	OR<=D	없음	없음	없음	3-421
339	AND 배정도 부동 소수점 초과	AND>D	없음	없음	없음	3-421
339	LD 배정도 부동 소수점 초과	LD>D	없음	없음	없음	3-421
339	OR 배정도 부동 소수점 초과	OR>D	없음	없음	없음	3-421
340	AND 배정도 부동 소수점 이상	AND>=D	없음	없음	없음	3-421
340	LD 배정도 부동 소수점 이상	LD>=D	없음	없음	없음	3-421
340	OR 배정도 부동 소수점 이상	OR>=D	없음	없음	없음	3-421
341	AND 시각 일치	AND=DT	없음	없음	없음	3-124
341	LD 시각 일치	LD=DT	없음	없음	없음	3-124
341	OR 시각 일치	OR=DT	없음	없음	없음	3-124
342	AND 시각 불일치	AND<>DT	없음	없음	없음	3-124
342	LD 시각 불일치	LD<>DT	없음	없음	없음	3-124
342	OR 시각 불일치	OR<>DT	없음	없음	없음	3-124
343	AND 시각 미만	AND<DT	없음	없음	없음	3-124
343	LD 시각 미만	LD<DT	없음	없음	없음	3-124
343	OR 시각 미만	OR<DT	없음	없음	없음	3-124
344	AND 시각 이하	AND<=DT	없음	없음	없음	3-124
344	LD 시각 이하	LD<=DT	없음	없음	없음	3-124
344	OR 시각 이하	OR<=DT	없음	없음	없음	3-124
345	AND 시각 초과	AND>DT	없음	없음	없음	3-124
345	LD 시각 초과	LD>DT	없음	없음	없음	3-124
345	OR 시각 초과	OR>DT	없음	없음	없음	3-124
346	AND 시각 이상	AND>=DT	없음	없음	없음	3-124
346	LD 시각 이상	LD>=DT	없음	없음	없음	3-124
346	OR 시각 이상	OR>=DT	없음	없음	없음	3-124
350	LD 비트 테스트	LD TST	없음	없음	없음	3-20

펑션 번호	명령 이름	니모닉	기동 지정	종료 지정	매회 리프레시 지정	페이지
350	AND 비트 테스트	AND TST	없음	없음	없음	3-22
350	OR 비트 테스트	OR TST	없음	없음	없음	3-24
351	LD 비트 테스트 부정	LD TSTN	없음	없음	없음	3-20
351	AND 비트 테스트 부정	AND TSTN	없음	없음	없음	3-22
351	OR 비트 테스트 부정	OR TSTN	없음	없음	없음	3-24
360	부호 없는 1W 레코드 검색, 미만	RSRCH <	@RSRCH <	없음	없음	3-495
361	부호 없는 1W 레코드 검색, 이하	RSRCH <=	@RSRCH <=	없음	없음	3-495
362	부호 없는 1W 레코드 검색, 일치	RSRCH =	@RSRCH =	없음	없음	3-495
363	부호 없는 1W 레코드 검색, 이상	RSRCH >=	@RSRCH >=	없음	없음	3-495
364	부호 없는 1W 레코드 검색, 초과	RSRCH >	@RSRCH >	없음	없음	3-495
370	부호 없는 2W 레코드 검색, 미만	RSRCH2 <	@RSRCH 2 <	없음	없음	3-501
371	부호 없는 2W 레코드 검색, 이하	RSRCH2 <=	@RSRCH 2 <=	없음	없음	3-501
372	부호 없는 2W 레코드 검색, 일치	RSRCH2 =	@RSRCH 2 =	없음	없음	3-501
373	부호 없는 2W 레코드 검색, 이상	RSRCH2 >=	@RSRCH 2 >=	없음	없음	3-501
374	부호 없는 2W 레코드 검색, 초과	RSRCH2 >	@RSRCH 2 >	없음	없음	3-501
380	부호 없는 4W 레코드 검색, 미만	RSRCH4 <	@RSRCH 4 <	없음	없음	3-504
381	부호 없는 4W 레코드 검색, 이하	RSRCH4 <=	@RSRCH 4 <=	없음	없음	3-504
382	부호 없는 4W 레코드 검색, 일치	RSRCH4 =	@RSRCH 4 =	없음	없음	3-504
383	부호 없는 4W 레코드 검색, 이상	RSRCH4 >=	@RSRCH 4 >=	없음	없음	3-504
384	부호 없는 4W 레코드 검색, 초과	RSRCH4 >	@RSRCH 4 >	없음	없음	3-504
400	부호 있음, CY 없는 BIN 가산	+	@+	없음	없음	3-222
401	부호 있음, CY 없는 BIN 2워드 가산	+L	@+L	없음	없음	3-222
402	부호 및 CY 있는 BIN 가산	+C	@+C	없음	없음	3-224
403	부호 및 CY 있는 BIN 2워드 가산	+CL	@+CL	없음	없음	3-224
404	CY 없는 BCD 가산	+B	@+B	없음	없음	3-227
405	CY 없는 BCD 2워드 가산	+BL	@+BL	없음	없음	3-227
406	CY 있는 BCD 가산	+BC	@+BC	없음	없음	3-229
407	CY 있는 BCD 2워드 가산	+BCL	@+BCL	없음	없음	3-229
410	부호 있음, CY 없는 BIN 감산	-	@-	없음	없음	3-231
411	부호 있음, CY 없는 BIN 2워드 감산	-L	@-L	없음	없음	3-231
412	부호 및 CY 있는 BIN 감산	-C	@-C	없음	없음	3-235
413	부호 및 CY 있는 BIN 2워드 감산	-CL	@-CL	없음	없음	3-235
414	CY 없는 BCD 감산	-B	@-B	없음	없음	3-238
415	CY 없는 BCD 2워드 감산	-BL	@-BL	없음	없음	3-238
416	CY 있는 BCD 감산	-BC	@-BC	없음	없음	3-241
417	CY 있는 BCD 2워드 감산	-BCL	@-BCL	없음	없음	3-241
420	부호 있는 BIN 승산	*	@*	없음	없음	3-243
421	부호 있는 BIN 2워드 승산	*L	@*L	없음	없음	3-243
422	부호 없는 BIN 승산	*U	@*U	없음	없음	3-245
423	부호 없는 BIN 2워드 승산	*UL	@*UL	없음	없음	3-245
424	BCD 승산	*B	@*B	없음	없음	3-247
425	BCD 2워드 승산	*BL	@*BL	없음	없음	3-247
430	부호 있는 BIN 제산	/	@/	없음	없음	3-249
431	부호 있는 BIN 2워드 제산	/L	@/L	없음	없음	3-249
432	부호 없는 BIN 제산	/U	@/U	없음	없음	3-251
433	부호 없는 BIN 2워드 제산	/UL	@/UL	없음	없음	3-251
434	BCD 제산	/B	@/B	없음	없음	3-253
435	BCD 2워드 제산	/BL	@/BL	없음	없음	3-253
448	부동 소수점<단>→문자열 변환	FSTR	@FSTR	없음	없음	3-378
449	문자열→부동 소수점<단> 변환	FVAL	@FVAL	없음	없음	3-382
450	부동 소수점→16비트 BIN 변환	FIX	@FIX	없음	없음	3-347
451	부동 소수점→32비트 BIN 변환	FIXL	@FIXL	없음	없음	3-347
452	16비트 BIN→부동 소수점 변환	FLT	@FLT	없음	없음	3-349
453	32비트 BIN→부동 소수점 변환	FLTL	@FLTL	없음	없음	3-349
454	부동 소수점 가산	+F	@+F	없음	없음	3-351
455	부동 소수점 감산	-F	@-F	없음	없음	3-351
456	부동 소수점 승산	*F	@*F	없음	없음	3-351
457	부동 소수점 제산	/F	@/F	없음	없음	3-351
458	각도→라디안 변환	RAD	@RAD	없음	없음	3-354
459	라디안→각도 변환	DEG	@DEG	없음	없음	3-356
460	SIN 연산	SIN	@SIN	없음	없음	3-358
461	COS 연산	COS	@COS	없음	없음	3-358
462	TAN 연산	TAN	@TAN	없음	없음	3-358
463	SIN^{-1} 연산	ASIN	@ASIN	없음	없음	3-364
464	COS^{-1} 연산	ACOS	@ACOS	없음	없음	3-364
465	TAN^{-1} 연산	ATAN	@ATAN	없음	없음	3-364
466	제곱근 연산	SQRT	@SQRT	없음	없음	3-367
467	지수 연산	EXP	@EXP	없음	없음	3-369

펑션 번호	명령 이름	니모닉	기동 지정	종료 지정	매회 리프레시 지정	페이지
468	로그 연산	LOG	@LOG	없음	없음	3-371
469	부동 소수점 전송<단>	MOVF	@MOVF	없음	없음	3-386
470	부호 있는 BCD→BIN 변환	BINS	@BINS	없음	없음	3-287
471	부호 있는 BIN→BCD 변환	BCDS	@BCDS	없음	없음	3-290
472	부호 있는 BCD→BIN 2워드 변환	BISL	@BISL	없음	없음	3-287
473	부호 있는 BIN→BCD 2워드 변환	BDSL	@BDSL	없음	없음	3-290
474	그레이 코드 변환	GRY	@GRY	없음	없음	3-295
475	고속 SIN 연산	SINQ	@SINQ	없음	없음	3-361
476	고속 COS 연산	COSQ	@COSQ	없음	없음	3-361
477	고속 TAN 연산	TANQ	@TANQ	없음	없음	3-361
478	그레이 코드 →BIN 변환	GRAY_BIN	@GRAY_BIN	없음	없음	3-298
479	그레이 코드 →BIN 2워드 변환	GRAY_BINL	@GRAY_BINL	없음	없음	3-298
480	BIN→그레이 코드 변환	BIN_GRAY	@BIN_GRAY	없음	없음	3-300
481	BIN→그레이 코드 2워드 변환	BIN_GRAYL	@BIN_GRAYL	없음	없음	3-300
486	스케일링 2	SCL2	@SCL2	없음	없음	3-550
487	스케일링 3	SCL3	@SCL3	없음	없음	3-553
490	명령 송신	CMND	@CMND	없음	없음	3-791
491	네트워크 송신 2	SEND2	@SEND2	없음	없음	3-779
492	네트워크 수신 2	RECV2	@RECV2	없음	없음	3-788
493	명령 송신 2	CMND2	@CMND2	없음	없음	3-798
498	2워드 전송	MOVL	@MOVL	없음	없음	3-152
499	부정 2워드 전송	MVNL	@MVNL	없음	없음	3-154
502	확장 테이블 간 비교	BCMP2	@BCMP2	없음	없음	3-140
510	조건부 점프	CJP	없음	없음	없음	3-69
511	조건부 부정 점프	CJPN	없음	없음	없음	3-69
512	반복 시작	FOR	없음	없음	없음	3-74
513	반복 종료	NEXT	없음	없음	없음	3-74
514	루프 중단	BREAK	없음	없음	없음	3-77
515	복수 점프	JMP0	없음	없음	없음	3-72
516	복수 점프 종료	JME0	없음	없음	없음	3-72
517	멀티 인터록 (미분 플래그 유지형)	MILH	없음	없음	없음	3-57
518	멀티 인터록 (미분 플래그 비유지형)	MILR	없음	없음	없음	3-57
519	멀티 인터록 클리어	MILC	없음	없음	없음	3-57
520	부정	NOT	없음	없음	없음	3-17
521	P.F. 기동 미분	UP	없음	없음	없음	3-18
522	P.F. 종료 미분	DOWN	없음	없음	없음	3-18
530	다중 비트 세트	SETA	@SETA	없음	없음	3-42
531	다중 비트 리셋	RSTA	@RSTA	없음	없음	3-42

펑션 번호	명령 이름	니모닉	기동 지정	종료 지정	매회 리프레시 지정	페이지
532	1비트 세트	SETB	@SETB	없음	!SETB	3-44
533	1비트 리셋	RSTB	@RSTB	없음	!RSTB	3-44
534	1비트 출력	OUTB	@OUTB	없음	!OUTB	3-46
540	1ms 타이머	TMHH	없음	없음	없음	3-94
541	0.1ms 타이머	TIMU	없음	없음	없음	3-96
542	장시간 타이머	TIML	없음	없음	없음	3-103
543	멀티 출력 타이머	MTIM	없음	없음	없음	3-106
544	0.01ms 타이머	TMUH	없음	없음	없음	3-98
545	타이머/카운터 리셋	CNR	@CNR	없음	없음	3-115
546	카운터	CNTX	없음	없음	없음	3-109
547	타이머/카운터 리셋	CNRX	@CNRX	없음	없음	3-115
548	가역 카운터	CNTRX	없음	없음	없음	3-112
549	타이머 리셋	TRSET	@TRSET	없음	없음	3-117
550	100ms 타이머	TIMX	없음	없음	없음	3-88
551	10ms 타이머	TIMHX	없음	없음	없음	3-91
552	1ms 타이머	TMHHX	없음	없음	없음	3-94
553	장시간 타이머	TIMLX	없음	없음	없음	3-103
554	멀티 출력 타이머	MTIMX	없음	없음	없음	3-106
555	적산 타이머	TTIMX	없음	없음	없음	3-100
556	0.1ms 타이머	TIMUX	없음	없음	없음	3-96
557	0.01ms 타이머	TMUHX	없음	없음	없음	3-98
560	인덱스 레지스터 설정	MOVR	@MOVR	없음	없음	3-172
561	인덱스 레지스터 설정	MOVRW	@MOVRW	없음	없음	3-172
562	데이터 2워드 교환	XCGL	@XCGL	없음	없음	3-166
565	블록 전송	XFERC	@ XFERC	없음	없음	3-970
566	데이터 분배	DISTC	@DISTC	없음	없음	3-972
567	데이터 추출	COLLC	@COLLC	없음	없음	3-975
568	비트 전송	MOVBC	@MOVBC	없음	없음	3-979
570	1비트 좌측 이동(2워드)	ASLL	@ASLL	없음	없음	3-184
571	1비트 우측 이동(2워드)	ASRL	@ASRL	없음	없음	3-186
572	CY가 있는 1비트 좌측 회전(2워드)	ROLL	@ROLL	없음	없음	3-188
573	CY가 있는 1비트 우측 회전(2워드)	RORL	@RORL	없음	없음	3-192
574	CY가 없는 1비트 좌측 회전	RLNC	@RLNC	없음	없음	3-190
575	CY가 없는 1비트 우측 회전	RRNC	@RRNC	없음	없음	3-194
576	CY가 없는 1비트 좌측 회전(2워드)	RLNL	@RLNC	없음	없음	3-190
577	CY가 없는 1비트 우측 회전(2워드)	RRNL	@RRNL	없음	없음	3-194
578	N비트 데이터 좌측 이동	NSFL	@NSFL	없음	없음	3-198
579	N비트 데이터 우측 이동	NSFR	@NSFR	없음	없음	3-198
580	N비트 좌측 이동	NASL	@NASL	없음	없음	3-200
581	N비트 우측 이동	NASR	@NASR	없음	없음	3-203
582	N비트 2워드 좌측 이동	NSLL	@NSLL	없음	없음	3-200
583	N비트 2워드 우측 이동	NSRL	@NSRL	없음	없음	3-203

펑션 번호	명령 이름	니모닉	기동 지정	종료 지정	매회 리프레시 지정	페이지
590	BIN 증가	++	@++	없음	없음	3-208
591	BIN 2워드 증가	++L	@++L	없음	없음	3-208
592	BIN 감소	--	@--	없음	없음	3-211
593	BIN 2워드 감소	--L	@--L	없음	없음	3-211
594	BCD 증가	++B	@++B	없음	없음	3-214
595	BCD 2워드 증가	++BL	@++BL	없음	없음	3-214
596	BCD 감소	--B	@--B	없음	없음	3-217
597	BCD 2워드 감소	--BL	@--BL	없음	없음	3-217
600	부호 확장	SIGN	@SIGN	없음	없음	3-263
601	숫자 4자리 →ASCII 변환	STR4	@STR4	없음	없음	3-302
602	숫자 8자리 →ASCII 변환	STR8	@STR8	없음	없음	3-302
603	숫자 16자리 →ASCII 변환	STR16	@STR16	없음	없음	3-302
604	ASCII→숫자 4자리 변환	NUM4	@NUM4	없음	없음	3-305
605	ASCII→숫자 8자리 변환	NUM8	@NUM8	없음	없음	3-305
606	ASCII→숫자 16자리 변환	NUM16	@NUM16	없음	없음	3-305
610	워드 논리곱(2워드)	ANDL	@ANDL	없음	없음	3-310
611	워드 논리합(2워드)	ORWL	@ORWL	없음	없음	3-312
612	워드 배타적 논리합(2워드)	XORL	@XORL	없음	없음	3-314
613	워드 배타적 논리합 부정(2워드)	XNRL	@XNRL	없음	없음	3-316
614	비트 2워드 반전	COML	@COML	없음	없음	3-318
620	BIN 제곱근 연산	ROTB	@ROTB	없음	없음	3-322
621	비트 카운터	BCNTC	@BCNTC	없음	없음	3-999
630	스택 영역 설정	SSET	@SSET	없음	없음	3-430
631	테이블 영역 선언	DIM	@DIM	없음	없음	3-452
632	스택 데이터 저장	PUSH	@PUSH	없음	없음	3-432
633	선입선출	FIFO	@FIFO	없음	없음	3-434
634	후입선출	LIFO	@LIFO	없음	없음	3-434
635	레코드 위치 설정	SETR	@SETR	없음	없음	3-454
636	레코드 위치 읽기	GETR	@GETR	없음	없음	3-456
637	바이트 스와핑	SWAP	@SWAP	없음	없음	3-461
638	스택 데이터 수 출력	SNUM	@SNUM	없음	없음	3-438
639	스택 데이터 참조	SREAD	@SREAD	없음	없음	3-440
640	스택 데이터 갱신	SWRIT	@SWRIT	없음	없음	3-443
641	스택 데이터 삽입	SINS	@SINS	없음	없음	3-446
642	스택 데이터 삭제	SDEL	@SDEL	없음	없음	3-449
650	문자열 길이 검출	LEN$	@LEN$	없음	없음	3-944
652	왼쪽에서 문자열 가져오기	LEFT$	@LEFT$	없음	없음	3-938
653	오른쪽에서 문자열 가져오기	RGHT$	@RGHT$	없음	없음	3-938
654	임의 위치에서 문자열 가져오기	MID$	@MID$	없음	없음	3-940
656	문자열 연결	+$	@+$	없음	없음	3-936
657	문자열 삽입	INS$	@INS$	없음	없음	3-954
658	문자열 삭제	DEL$	@DEL$	없음	없음	3-948
660	문자열 검색	FIND$	@FIND$	없음	없음	3-942
661	문자열 치환	RPLC$	@RPLC$	없음	없음	3-946
664	문자열 전송	MOV$	@MOV$	없음	없음	3-934
665	문자열 교환	XCHG$	@XCHG$	없음	없음	3-950
666	문자열 클리어	CLR$	@CLR$	없음	없음	3-952
670	AND 문자열 일치	AND=$	없음	없음	없음	3-956
670	LD 문자열 일치	LD=$	없음	없음	없음	3-956
670	OR 문자열 일치	OR=$	없음	없음	없음	3-956
671	AND 문자열 불일치	AND<>$	없음	없음	없음	3-956
671	LD 문자열 불일치	LD<>$	없음	없음	없음	3-956
671	OR 문자열 불일치	OR<>$	없음	없음	없음	3-956
672	AND 문자열 미만	AND<$	없음	없음	없음	3-956
672	LD 문자열 미만	LD<$	없음	없음	없음	3-956
672	OR 문자열 미만	OR<$	없음	없음	없음	3-956
673	AND 문자열 이하	AND<=$	없음	없음	없음	3-956
673	LD 문자열 이하	LD<=$	없음	없음	없음	3-956
673	OR 문자열 이하	OR<=$	없음	없음	없음	3-956
674	AND 문자열 초과	AND>$	없음	없음	없음	3-956
674	LD 문자열 초과	LD>$	없음	없음	없음	3-956
674	OR 문자열 초과	OR>$	없음	없음	없음	3-956
675	AND 문자열 이상	AND>=$	없음	없음	없음	3-956
675	LD 문자열 이상	LD>=$	없음	없음	없음	3-956
675	OR 문자열 이상	OR>=$	없음	없음	없음	3-956
680	상하한 리미트 제어	LMT	@LMT	없음	없음	3-534
681	데드 밴드 제어	BAND	@BAND	없음	없음	3-536
682	데드 존 제어	ZONE	@ZONE	없음	없음	3-538
685	시분할 비례 출력	TPO	없음	없음	없음	3-540
690	인터럽트 마스크 세트	MSKS	@MSKS	없음	없음	3-585
691	인터럽트 해제	CLI	@CLI	없음	없음	3-595
692	인터럽트 마스크 읽기	MSKR	@MSKR	없음	없음	3-591
693	인터럽트 태스크 실행 금지	DI	@DI	없음	없음	3-599
694	인터럽트 태스크 실행 금지 해제	EI	없음	없음	없음	3-601
700	데이터 파일 읽기	FREAD	@FREAD	없음	없음	3-831
701	데이터 파일 쓰기	FWRIT	@FWRIT	없음	없음	3-836
704	텍스트 파일 쓰기	TWRIT	@TWRIT	없음	없음	3-841

펑션 번호	명령 이름	니모닉	기동 지정	종료 지정	매회 리프레시 지정	페이지
720	범용 Explicit 메시지 송신 명령	EXPLT	@EXPLT	없음	없음	3-802
721	Explicit 읽기 명령	EGATR	@EGATR	없음	없음	3-807
722	Explicit 쓰기 명령	ESATR	@ESATR	없음	없음	3-812
723	Explicit CPU 유니트 데이터 읽기 명령	ECHRD	@ECHRD	없음	없음	3-816
724	Explicit CPU 유니트 데이터 쓰기 명령	ECHWR	@ECHWR	없음	없음	3-819
730	캘린더 가산	CADD	@CADD	없음	없음	3-852
731	캘린더 감산	CSUB	@CSUB	없음	없음	3-852
735	시계 보정	DATE	@DATE	없음	없음	3-860
750	글로벌 서브루틴 호출	GSBS	@GSBS	없음	없음	3-572
751	글로벌 서브루틴 진입	GSBN	없음	없음	없음	3-578
752	글로벌 서브루틴 반환	GRET	없음	없음	없음	3-578
780	스텝 타이머 현재값 읽기 명령	TSR	@TSR	없음	없음	3-992
781	스텝 타이머 현재값 변경 명령	TSW	@TSW	없음	없음	3-992
784	스텝 활성화 명령	SA	@SA	없음	없음	3-990
785	스텝 비활성화 명령	SE	@SE	없음	없음	3-990
789	SFC 태스크 실행 명령	SFCON	없음	없음	없음	3-994
790	SFC 태스크 종료 명령	SFCOFF	없음	없음	없음	3-994
791	SFC 태스크 정지 명령(리셋 없음)	SFCPRN	없음	없음	없음	3-996
793	SFC 태스크 정지 명령(리셋 있음)	SFCPR	없음	없음	없음	3-996
801	블록 프로그램 종료	BEND	없음	없음	없음	3-908
802	조건 분기 블록	입력 조건 IF	없음	없음	없음	3-914
802	조건 분기 블록	IF 릴레이 번호	없음	없음	없음	3-914
802	조건 분기 블록(부정)	IF NOT 릴레이 번호	없음	없음	없음	3-914
803	조건 분기 거짓 블록	ELSE	없음	없음	없음	3-914
804	조건 분기 블록 종료	IEND	없음	없음	없음	3-914
805	1스캔 조건 대기	입력 조건 WAIT	없음	없음	없음	3-917
805	1스캔 조건 대기	WAIT 릴레이 번호	없음	없음	없음	3-917
805	1스캔 조건 대기(부정)	WAIT NOT 릴레이 번호	없음	없음	없음	3-917
806	조건부 종료	입력 조건 EXIT	없음	없음	없음	3-912
806	조건부 종료	EXIT 릴레이 번호	없음	없음	없음	3-912
806	조건부 종료(부정)	EXIT NOT 릴레이 번호	없음	없음	없음	3-912
809	반복 블록	LOOP	없음	없음	없음	3-927
810	반복 블록 종료	입력 조건 LEND	없음	없음	없음	3-927
810	반복 블록 종료	LEND 릴레이 번호	없음	없음	없음	3-927
810	반복 블록 종료(부정)	LEND NOT 릴레이 번호	없음	없음	없음	3-927
811	블록 프로그램 일시 정지	BPPS	없음	없음	없음	3-910
812	블록 프로그램 재기동	BPRS	없음	없음	없음	3-910
813	타이머 대기	TIMW	없음	없음	없음	3-920
814	카운터 대기	CNTW	없음	없음	없음	3-923
815	고속 타이머 대기	TMHW	없음	없음	없음	3-925
816	타이머 대기	TIMWX	없음	없음	없음	3-920
817	고속 타이머 대기	TMHWX	없음	없음	없음	3-925
818	카운터 대기	CNTWX	없음	없음	없음	3-923
820	태스크 실행 기동	TKON	@TKON	없음	없음	3-962
821	태스크 실행 대기	TKOF	@TKOF	없음	없음	3-962
840	거듭제곱 연산	PWR	@PWR	없음	없음	3-373
841	부동 소수점→16비트 BIN 변환<배>	FIXD	@FIXD	없음	없음	3-396
842	부동 소수점→32비트 BIN 변환<배>	FIXLD	@FIXLD	없음	없음	3-396
843	16비트 BIN→부동 소수점 변환<배>	DBL	@DBL	없음	없음	3-398
844	32비트 BIN→부동 소수점 변환<배>	DBLL	@DBLL	없음	없음	3-398
845	부동 소수점 가산<배>	+D	@+D	없음	없음	3-400
846	부동 소수점 감산<배>	−D	@−D	없음	없음	3-400
847	부동 소수점 승산<배>	*D	@*D	없음	없음	3-400
848	부동 소수점 제산<배>	/D	@/D	없음	없음	3-400
849	각도→라디안 변환<배>	RADD	@RADD	없음	없음	3-403
850	라디안→각도 변환<배>	DEGD	@DEGD	없음	없음	3-405
851	SIN 연산<배>	SIND	@SIND	없음	없음	3-407
852	COS 연산<배>	COSD	@COSD	없음	없음	3-407
853	TAN 연산<배>	TAND	@TAND	없음	없음	3-407
854	SIN^{-1} 연산<배>	ASIND	@ASIND	없음	없음	3-410
855	COS^{-1} 연산<배>	ACOSD	@ACOSD	없음	없음	3-410
856	TAN^{-1} 연산<배>	ATAND	@ATAND	없음	없음	3-410
857	제곱근 연산<배>	SQRTD	@SQRTD	없음	없음	3-413
858	지수 연산<배>	EXPD	@EXPD	없음	없음	3-415
859	로그 연산<배>	LOGD	@LOGD	없음	없음	3-417
860	거듭제곱 연산<배>	PWRD	@PWRD	없음	없음	3-419
880	동작 모드 제어	INI	@INI	없음	없음	3-604
881	펄스 현재값 출력	PRV	@PRV	없음	없음	3-607
882	비교 테이블 등록	CTBL	@CTBL	없음	없음	3-615
883	펄스 주파수 변환	PRV2	@PRV2	없음	없음	3-612
885	주파수 설정	SPED	@SPED	없음	없음	3-618
886	펄스량 설정	PULS	@PULS	없음	없음	3-621
887	위치 결정	PLS2	@PLS2	없음	없음	3-623

펑션 번호	명령 이름	니모닉	기동 지정	종료 지정	매회 리프레시 지정	페이지
888	주파수 제어	ACC	@ACC	없음	없음	3-628
889	원점 검색	ORG	@ORG	없음	없음	3-632
891	PWM 출력	PWM	@PWM	없음	없음	3-635

부록 ABC 순서 명령 목록

A

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
ACC	주파수 제어	888	@ACC	없음	없음	3-628
ACOS	COS^{-1} 연산	464	@ACOS	없음	없음	3-364
ACOSD	COS^{-1} 연산<배>	855	@ACOSD	없음	없음	3-410
AIDC	아날로그 입력 직접 변환	216	@AIDC	없음	없음	3-677
AND	논리곱	없음	@AND	%AND	!AND	3-10
AND LD	로드 논리곱	없음	없음	없음	없음	3-14
AND NOT	부정 논리곱	없음	없음	없음	!AND NOT	3-10
AND TST	AND 비트 테스트	350	없음	없음	없음	3-22
AND TSTN	AND 비트 테스트 부정	351	없음	없음	없음	3-22
AND<	AND 미만	310	없음	없음	없음	3-120
AND<$	AND 문자열 미만	672	없음	없음	없음	3-956
AND<=	AND 이하	315	없음	없음	없음	3-120
AND<=	AND 단정도 부동 소수점 이하	332	없음	없음	없음	3-375
AND<=	AND 배정도 부동 소수점 이하	338	없음	없음	없음	3-421
AND<=$	AND 문자열 이하	673	없음	없음	없음	3-956
AND<=DT	AND 시각 이하	344	없음	없음	없음	3-124
AND<=L	AND 2워드 이하	316	없음	없음	없음	3-120
AND<=S	AND 부호 있는 이하	317	없음	없음	없음	3-120
AND<=SL	AND 부호 있는 2워드 이하	318	없음	없음	없음	3-120
AND<>	AND 불일치	305	없음	없음	없음	3-120
AND<>$	AND 문자열 불일치	671	없음	없음	없음	3-956
AND<>D	AND 배정도 부동 소수점 불일치	336	없음	없음	없음	3-421
AND<>DT	AND 시각 불일치	342	없음	없음	없음	3-124
AND<>F	AND 단정도 부동 소수점 불일치	330	없음	없음	없음	3-375
AND<>L	AND 2워드 불일치	306	없음	없음	없음	3-120
AND<>S	AND 부호 있는 불일치	307	없음	없음	없음	3-120
AND<>SL	AND 부호 있는 2워드 불일치	308	없음	없음	없음	3-120
AND<D	AND 배정도 부동 소수점 미만	337	없음	없음	없음	3-421
AND<DT	AND 시각 미만	343	없음	없음	없음	3-124
AND<F	AND 단정도 부동 소수점 미만	331	없음	없음	없음	3-375
AND<L	AND 2워드 미만	311	없음	없음	없음	3-120
AND<S	AND 부호 있는 미만	312	없음	없음	없음	3-120
AND<SL	AND 부호 있는 2워드 미만	313	없음	없음	없음	3-120
AND=	AND 일치	300	없음	없음	없음	3-120
AND=$	AND 문자열 일치	670	없음	없음	없음	3-956
AND=D	AND 배정도 부동 소수점 일치	335	없음	없음	없음	3-421
AND=DT	AND 시각 일치	341	없음	없음	없음	3-124
AND=F	AND 단정도 부동 소수점 일치	329	없음	없음	없음	3-375
AND=L	AND 2워드 일치	301	없음	없음	없음	3-120
AND=S	AND 부호 있는 일치	302	없음	없음	없음	3-120
AND=SL	AND 부호 있는 2워드 일치	303	없음	없음	없음	3-120
AND>	AND 초과	320	없음	없음	없음	3-120
AND>$	AND 문자열 초과	674	없음	없음	없음	3-956
AND>=	AND 이상	325	없음	없음	없음	3-120
AND>=$	AND 문자열 이상	675	없음	없음	없음	3-956
AND>=D	AND 배정도 부동 소수점 이상	340	없음	없음	없음	3-421
AND>=DT	AND 시각 이상	346	없음	없음	없음	3-124
AND>=F	AND 단정도 부동 소수점 이상	334	없음	없음	없음	3-375
AND>=L	AND 2워드 이상	326	없음	없음	없음	3-120
AND>=S	AND 부호 있는 이상	327	없음	없음	없음	3-120
AND>=SL	AND 부호 있는 2워드 이상	328	없음	없음	없음	3-120

A

니모닉	명령 이름	평션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
AND>D	AND 배정도 부동 소수점 초과	339	없음	없음	없음	3-421
AND>DT	AND 시각 초과	345	없음	없음	없음	3-124
AND>F	AND 단정도 부동 소수점 초과	333	없음	없음	없음	3-375
AND>L	AND 2워드 초과	321	없음	없음	없음	3-120
AND>S	AND 부호 있는 초과	322	없음	없음	없음	3-120
AND>SL	AND 부호 있는 2워드 초과	323	없음	없음	없음	3-120
ANDL	워드 논리곱 (2워드)	610	@ANDL	없음	없음	3-310
ANDW	워드 논리곱	034	@ANDW	없음	없음	3-310
AODC	아날로그 출력 직접 변환	217	@AODC	없음	없음	3-680
APR	숫자 변환	069	@APR	없음	없음	3-326
ASC	ASCII 코드 변환	086	@ASC	없음	없음	3-275
ASFT	비동기 이동 레지스터	017	@ASFT	없음	없음	3-180
ASIN	SIN^{-1} 연산	463	@ASIN	없음	없음	3-364
ASIND	SIN^{-1} 연산<배>	854	@ASIND	없음	없음	3-410
ASL	1비트 좌측 이동	025	@ASL	없음	없음	3-184
ASLL	1비트 좌측 이동(2워드)	570	@ASLL	없음	없음	3-184
ASR	1비트 우측 이동	026	@ASR	없음	없음	3-186
ASRL	1비트 우측 이동(2워드)	571	@ASRL	없음	없음	3-186
ATAN	TAN^{-1} 연산	465	@ATAN	없음	없음	3-364
ATAND	TAN^{-1} 연산<배>	856	@ATAND	없음	없음	3-410
AVG	데이터 평균화	195	없음	없음	없음	3-556

B

니모닉	명령 이름	평션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
BAND	데드 밴드 제어	681	@BAND	없음	없음	3-536
BCD	BIN→BCD 변환	024	@BCD	없음	없음	3-258
BCDL	BIN→BCD 2워드 변환	059	@BCDL	없음	없음	3-258
BCDS	부호 있는 BIN→BCD 변환	471	@BCDS	없음	없음	3-290
BCMP	부호 없는 테이블 간 비교	068	@BCMP	없음	없음	3-138
BCMP2	확장 테이블 간 비교	502	@BCMP2	없음	없음	3-140
BCNT	비트 카운터	067	@BCNT	없음	없음	3-338
BCNTC	비트 카운터	621	@BCNTC	없음	없음	3-981
BDSL	부호 있는 BIN→BCD 2워드 변환	473	@BDSL	없음	없음	3-290
BEND	블록 프로그램 종료	801	없음	없음	없음	3-908
BIN	BCD→BIN 변환	023	@BIN	없음	없음	3-256
BIN_GRAY	BIN→그레이 코드 변환	480	@BIN_GRAY	없음	없음	3-300
BIN_GRAYL	BIN→그레이 코드 2워드 변환	481	@BIN_GRAYL	없음	없음	3-300
BINL	BCD→BIN 2워드 변환	058	@BINL	없음	없음	3-256
BINS	부호 있는 BCD→BIN 변환	470	@BINS	없음	없음	3-287
BISL	부호 있는 BCD→BIN 2워드 변환	472	@BISL	없음	없음	3-287
BPPS	블록 프로그램 일시 정지	811	없음	없음	없음	3-910
BPRG	블록 프로그램	096	없음	없음	없음	3-908
BPRS	블록 프로그램 재기동	812	없음	없음	없음	3-910
BREAK	루프 중단	514	없음	없음	없음	3-77
BSET	블록 설정	071	@BSET	없음	없음	3-164

C

니모닉	명령 이름	평션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
CADD	캘린더 가산	730	@CADD	없음	없음	3-852
CCL	조건 플래그 로드	283	@CCL	없음	없음	3-893
CCS	조건 플래그 저장	282	@CCS	없음	없음	3-893
CJP	조건부 점프	510	없음	없음	없음	3-69
CJPN	조건부 부정 점프	511	없음	없음	없음	3-69
CLC	클리어 캐리	041	@CLC	없음	없음	3-888
CLI	인터럽트 해제	691	@CLI	없음	없음	3-595
CLR$	문자열 클리어	666	@CLR$	없음	없음	3-952
CMND	명령 송신	490	@CMND	없음	없음	3-791
CMND2	명령 송신 2	493	@CMND2	없음	없음	3-798
CMP	부호 없는 비교	020	없음	없음	!CMP	3-128
CMPL	부호 없는 2워드 비교	060	없음	없음	없음	3-128
CNR	타이머/카운터 리셋	545	@CNR	없음	없음	3-115
CNRX	타이머/카운터 리셋	547	@CNRX	없음	없음	3-115
CNT	카운터	없음	없음	없음	없음	3-109
CNTR	가역 카운터	012	없음	없음	없음	3-112
CNTRX	가역 카운터	548	없음	없음	없음	3-112
CNTW	카운터 대기	814	없음	없음	없음	3-923
CNTWX	카운터 대기	818	없음	없음	없음	3-923
CNTX	카운터	546	없음	없음	없음	3-109
COLL	데이터 추출	081	@COLL	없음	없음	3-170
COLLC	데이터 추출	567	@COLLC	없음	없음	3-975
COLM	비트 행→비트 열 변환	064	@COLM	없음	없음	3-285
COM	비트 반전	029	@COM	없음	없음	3-318
COML	비트 2워드 반전	614	@COML	없음	없음	3-318

C

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
COS	COS 연산	461	@COS	없음	없음	3-358
COSD	COS 연산 <배>	852	@COSD	없음	없음	3-407
COSQ	고속 COS 연산	476	@COSQ	없음	없음	3-361
CPS	부호 있는 BIN 비교	114	없음	없음	!CPS	3-131
CPSL	부호 있는 BIN 2워드 비교	115	없음	없음	없음	3-131
CSUB	캘린더 감산	731	@CSUB	없음	없음	3-852
CTBL	비교 테이블 등록	882	@CTBL	없음	없음	3-615

D

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
DATE	시계 보정	735	@DATE	없음	없음	3-860
DBL	16비트 BIN→부동 소수점 변환<배>	843	@DBL	없음	없음	3-398
DBLL	32비트 BIN→부동 소수점 변환<배>	844	@DBLL	없음	없음	3-398
DEG	라디안→각도 변환	459	@DEG	없음	없음	3-356
DEGD	라디안→각도 변환<배>	850	@DEGD	없음	없음	3-405
DEL$	문자열 삭제	658	@DEL$	없음	없음	3-948
DI	인터럽트 태스크 실행 금지	693	@DI	없음	없음	3-599
DIFD	종료 미분	014	없음	없음	!DIFD	3-38
DIFU	기동 미분	013	없음	없음	!DIFU	3-36
DIM	테이블 영역 선언	631	@DIM	없음	없음	3-452
DIST	데이터 분배	080	@DIST	없음	없음	3-168
DISTC	데이터 분배	566	@DISTC	없음	없음	3-972
DLNK	CPU 고기능 유니트 매회 I/O 리프레시	226	@DLNK	없음	없음	3-656
DMPX	16→4/256→8 인코더	077	@DMPX	없음	없음	3-270
DOWN	P.F. 종료 미분	522	없음	없음	없음	3-18
DSW	디지털 스위치	210	없음	없음	없음	3-661
DRXDU	시리얼 커뮤니케이션 유니트 시리얼 포트 직접 수신	261	@DRXDU	없음	없음	7-738
DTXDU	시리얼 커뮤니케이션 유니트 시리얼 포트 직접 송신	262	@DTXDU	없음	없음	7-733

E

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
ECHRD	Explicit CPU 유니트 데이터 읽기 명령	723	@ECHRD	없음	없음	3-816
ECHWR	Explicit CPU 유니트 데이터 쓰기 명령	724	@ECHWR	없음	없음	3-819
EGATR	Explicit 읽기 명령	721	@EGATR	없음	없음	3-807
EI	인터럽트 태스크 실행 금지 해제	694	없음	없음	없음	3-601
ELSE	조건 분기 거짓 블록	803	없음	없음	없음	3-914
EMBC	확장 DM 뱅크 전환	281	@EMBC	없음	없음	3-889
END	종료	001	없음	없음	없음	3-52
ESATR	Explicit 쓰기 명령	722	@ESATR	없음	없음	3-812
EXIT NOT 릴레이 번호	조건부 종료 (부정)	806	없음	없음	없음	3-912
입력 조건 EXIT	조건부 종료	806	없음	없음	없음	3-912
EXIT 릴레이 번호	조건부 종료	806	없음	없음	없음	3-912
EXP	지수 연산	467	@EXP	없음	없음	3-369
EXPD	지수 연산 <배>	858	@EXPD	없음	없음	3-415
EXPLT	범용 Explicit 메시지 송신 명령	720	@EXPLT	없음	없음	3-802

F

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
FAL	운전 계속 고장 진단	006	@FAL	없음	없음	3-868
FALS	운전 정지 고장 진단	007	없음	없음	없음	3-874
FCS	FCS 값 산출	180	@FCS	없음	없음	3-483
FDIV	부동 소수점 제산(BCD)	079	@FDIV	없음	없음	3-335
FIFO	선입선출	633	@FIFO	없음	없음	3-434
FIND$	문자열 검색	660	@FIND$	없음	없음	3-942
FIORF	고기능 I/O 유니트 매회 리프레시	225	@FIORF	없음	없음	3-653
FIX	부동 소수점 →16비트 BIN 변환	450	@FIX	없음	없음	3-347
FIXD	부동 소수점 →16비트 BIN 변환<배>	841	@FIXD	없음	없음	3-396
FIXL	부동 소수점 →32비트 BIN 변환	451	@FIXL	없음	없음	3-347
FIXLD	부동 소수점 →32비트 BIN 변환<배>	842	@FIXLD	없음	없음	3-396
FLT	16비트 BIN→ 부동 소수점 변환	452	@FLT	없음	없음	3-349
FLTL	32비트 BIN→ 부동 소수점 변환	453	@FLTL	없음	없음	3-349

F

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
FSTR	부동 소수점<단> →문자열 변환	448	@FSTR	없음	없음	3-378
FVAL	문자열→ 부동 소수점 <단> 변환	449	@FVAL	없음	없음	3-382
FWRIT	데이터 파일 쓰기	701	@FWRIT	없음	없음	3-836
FOR	반복 시작	512	없음	없음	없음	3-74
FPD	고장 지점 검출	269	없음	없음	없음	3-879
FREAD	데이터 파일 읽기	700	@FREAD	없음	없음	3-831
FRMCV	CV→CS 주소 변환	284	@FRMCV	없음	없음	3-895

G

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
GETID	변수 종류 취득	286	@GETID	없음	없음	3-984
GETR	레코드 위치 읽기	636	@GETR	없음	없음	3-456
GRET	글로벌 서브 루틴 반환	752	없음	없음	없음	3-578
GRAY_BIN	그레이 코드 →BIN 변환	478	@GRAY_ BIN	없음	없음	3-298
GRAY_BINL	그레이 코드 →BIN 2워드 변환	479	@GRAY_ BINL	없음	없음	3-298
GRY	그레이 코드 변환	474	@GRY	없음	없음	3-295
GSBN	글로벌 서브루틴 진입	751	없음	없음	없음	3-578
GSBS	글로벌 서브루틴 호출	750	@GSBS	없음	없음	3-572

H

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
HEX	ASCII→HE X 변환	162	@HEX	없음	없음	3-279
HKY	16키 입력	212	없음	없음	없음	3-667
HMS	초→시분초 변환	066	@HMS	없음	없음	3-858

I

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
IEND	조건 분기 블록 종료	804	없음	없음	없음	3-914
IF NOT 릴레이 번호	조건 분기 블록(부정)	802	없음	없음	없음	3-914
입력 조건 IF	조건 분기 블록	802	없음	없음	없음	3-914
IF 릴레이 번호	조건 분기 블록	802	없음	없음	없음	3-914
IL	인터록	002	없음	없음	없음	3-54
ILC	인터록 클리어	003	없음	없음	없음	3-54
INI	동작 모드 제어	880	@INI	없음	없음	3-604
INS$	문자열 삽입	657	@INS$	없음	없음	3-954
IORD	인텔리전트 I/O 읽기	222	@IORD	없음	없음	3-691
IORF	I/O 리프레시	097	@IORF	없음	없음	3-650
IORS	주변 서비스 금지 해제	288	없음	없음	없음	3-901
IOSP	주변 서비스 금지	287	@IOSP	없음	없음	3-901
IOWR	인텔리전트 I/O 쓰기	223	@IOWR	없음	없음	3-694

J

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
JME	점프 종료	005	없음	없음	없음	3-66
JME0	복수 점프 종료	516	없음	없음	없음	3-72
JMP	점프	004	없음	없음	없음	3-66
JMP0	복수 점프	515	없음	없음	없음	3-72

K

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
KEEP	유지	011	없음	없음	!KEEP	3-32

L

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
LD	로드	없음	@LD	%LD	!LD	3-8
LD=DT	LD 시각 일치	341	없음	없음	없음	3-124
LD =S	LD 부호 있는 일치	302	없음	없음	없음	3-120
LD NOT	로드 부정	없음	없음	없음	!LD NOT	3-8
LD TST	LD 비트 테스트	350	없음	없음	없음	3-20
LD TSTN	LD 비트 테스트 부정	351	없음	없음	없음	3-20
LD<	LD 미만	310	없음	없음	없음	3-120
LD<$	LD 문자열 미만	672	없음	없음	없음	3-956
LD<=	LD 이하	315	없음	없음	없음	3-120
LD<=$	LD 문자열 이하	673	없음	없음	없음	3-956
LD<=D	LD 배정도 부동 소수점 이하	338	없음	없음	없음	3-421
LD<=DT	LD 시각 이하	344	없음	없음	없음	3-124

L

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
LD<=F	LD 단정도 부동 소수점 이하	332	없음	없음	없음	3-375
LD<=L	LD 2워드 이하	316	없음	없음	없음	3-120
LD<=S	LD 부호 있는 이하	317	없음	없음	없음	3-120
LD<=SL	LD 부호 있는 2워드 이하	318	없음	없음	없음	3-120
LD<>	LD 불일치	305	없음	없음	없음	3-120
LD<>$	LD 문자열 불일치	671	없음	없음	없음	3-956
LD<>D	LD 배정도 부동 소수점 불일치	336	없음	없음	없음	3-421
LD<>DT	LD 시각 불일치	342	없음	없음	없음	3-124
LD<>F	LD 단정도 부동 소수점 불일치	330	없음	없음	없음	3-375
LD<>L	LD 2워드 불일치	306	없음	없음	없음	3-120
LD<>S	LD 부호 있는 불일치	307	없음	없음	없음	3-120
LD<>SL	LD 부호 있는 2워드 불일치	308	없음	없음	없음	3-120
LD<D	LD 배정도 부동 소수점 미만	337	없음	없음	없음	3-421
LD<DT	LD 시각 미만	343	없음	없음	없음	3-124
LD<F	LD 단정도 부동 소수점 미만	331	없음	없음	없음	3-375
LD<L	LD 2워드 미만	311	없음	없음	없음	3-120
LD<S	LD 부호 있는 미만	312	없음	없음	없음	3-120
LD<SL	LD 부호 있는 2워드 미만	313	없음	없음	없음	3-120
LD=	LD 일치	300	없음	없음	없음	3-120
LD=$	LD 문자열 일치	670	없음	없음	없음	3-956
LD=D	LD 배정도 부동 소수점 일치	335	없음	없음	없음	3-421
LD=F	LD 단정도 부동 소수점 일치	329	없음	없음	없음	3-375
LD=L	LD 2워드 일치	301	없음	없음	없음	3-120
LD=SL	LD 부호 있는 2워드 일치	303	없음	없음	없음	3-120
LD>	LD 초과	320	없음	없음	없음	3-120
LD>$	LD 문자열 초과	674	없음	없음	없음	3-956
LD>=	LD 이상	325	없음	없음	없음	3-120
LD>=$	LD 문자열 이상	675	없음	없음	없음	3-956
LD>=D	LD 배정도 부동 소수점 이상	340	없음	없음	없음	3-421
LD>=DT	LD 시각 이상	346	없음	없음	없음	3-124
LD>=F	LD 단정도 부동 소수점 이상	334	없음	없음	없음	3-375
LD>=L	LD 2워드 이상	326	없음	없음	없음	3-120
LD>=S	LD 부호 있는 이상	327	없음	없음	없음	3-120
LD>=SL	LD 부호 있는 2워드 이상	328	없음	없음	없음	3-120
LD>D	LD 배정도 부동 소수점 초과	339	없음	없음	없음	3-421
LD>DT	LD 시각 초과	345	없음	없음	없음	3-124
LD>F	LD 단정도 부동 소수점 초과	333	없음	없음	없음	3-375
LD>L	LD 2워드 초과	321	없음	없음	없음	3-120
LD>S	LD 부호 있는 초과	322	없음	없음	없음	3-120
LD>SL	LD 부호 있는 2워드 초과	323	없음	없음	없음	3-120
LEFT$	왼쪽에서 문자열 가져오기	652	@LEFT$	없음	없음	3-938
LEN$	문자열 길이 검출	650	@LEN$	없음	없음	3-944
LEND NOT 릴레이 번호	반복 블록 종료(부정)	810	없음	없음	없음	3-927
입력 조건 LEND	반복 블록 종료	810	없음	없음	없음	3-927
LEND 릴레이 번호	반복 블록 종료	810	없음	없음	없음	3-927
LIFO	후입선출	634	@LIFO	없음	없음	3-434
LINE	비트 열→ 비트 행 변환	063	@LINE	없음	없음	3-283
LMT	상하한 리미트 제어	680	@LMT	없음	없음	3-534
LOG	로그 연산	468	@LOG	없음	없음	3-371
LOGD	로그 연산 <배>	859	@LOGD	없음	없음	3-417
LOOP	반복 블록	809	없음	없음	없음	3-927

M

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
MAX	최대값 검색	182	@MAX	없음	없음	3-463
MAXD	배정도 부동 소수점 데이터 최대값 검색	178	@MAXD	없음	없음	3-472
MAXF	단정도 부동 소수점 데이터 최대값 검색	176	@MAXF	없음	없음	3-470
MAXL	2워드 데이터 최대값 검색	174	@MAXL	없음	없음	3-467
MCMP	다채널 비교	019	@MCMP	없음	없음	3-134
MCRO	매크로	099	@MCRO	없음	없음	3-566
MID$	임의 위치에서 문자열 가져오기	654	@MID$	없음	없음	3-940
MILC	멀티 인터록 클리어	519	없음	없음	없음	3-57

M

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
MILH	멀티 인터록(미분 플래그 유지형)	517	없음	없음	없음	3-57
MILR	멀티 인터록(미분 플래그 비유지형)	518	없음	없음	없음	3-57
MIN	최소값 검색	183	@MIN	없음	없음	3-463
MIND	배정도 부동 소수점 데이터 최소값 검색	179	@MIND	없음	없음	3-478
MINF	단정도 부동 소수점 데이터 최소값 검색	177	@MINF	없음	없음	3-476
MINL	2워드 데이터 최소값 검색	175	@MINL	없음	없음	3-474
MLPX	4→16/8→256 디코더	076	@MLPX	없음	없음	3-265
MOV	전송	021	@MOV	없음	!MOV	3-152
MOV$	문자열 전송	664	@MOV$	없음	없음	3-934
MOVB	비트 전송	082	@MOVB	없음	없음	3-156
MOVBC	비트 전송	568	@MOVBC	없음	없음	3-979
MOVD	자릿수 전송	083	@MOVD	없음	없음	3-158
MOVF	부동 소수점 전송<단>	469	@MOVF	없음	없음	3-386
MOVL	2워드 전송	498	@MOVL	없음	없음	3-152
MOVR	인덱스 레지스터 설정	560	@MOVR	없음	없음	3-172
MOVRW	인덱스 레지스터 설정	561	@MOVRW	없음	없음	3-172
MSG	메시지 표시	046	@MSG	없음	없음	3-848
MSKR	인터럽트 마스크 읽기	692	@MSKR	없음	없음	3-591
MSKS	인터럽트 마스크 세트	690	@MSKS	없음	없음	3-585
MTIM	멀티 출력 타이머	543	없음	없음	없음	3-106
MTIMX	멀티 출력 타이머	554	없음	없음	없음	3-106
MTR	행렬 입력	213	없음	없음	없음	3-670
MVN	부정 전송	022	@MVN	없음	없음	3-154
MVNL	부정 2워드 전송	499	@MVNL	없음	없음	3-154

N

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
NASL	N비트 좌측 이동	580	@NASL	없음	없음	3-200
NASR	N비트 우측 이동	581	@NASR	없음	없음	3-203
NEG	2의 보수 변환	160	@NEG	없음	없음	3-261
NCDMV	NC 유니트 위치 결정 고속 기동	218	@NCDMV	없음	없음	3-683
NCDTR	NC 유니트 위치 결정 시작 트리거	219	@NCDTR	없음	없음	3-688
NEGL	2의 보수 변환(2워드)	161	@NEGL	없음	없음	3-261
NEXT	반복 종료	513	없음	없음	없음	3-74
NOP	기능 없음	000	없음	없음	없음	3-53
NOT	부정	520	없음	없음	없음	3-17
NSFL	N비트 데이터 좌측 이동	578	@NSFL	없음	없음	3-198
NSFR	N비트 데이터 우측 이동	579	@NSFR	없음	없음	3-198
NSLL	N비트 2워드 좌측 이동	582	@NSLL	없음	없음	3-200
NSRL	N비트 2워드 우측 이동	583	@NSRL	없음	없음	3-203
NUM4	ASCII→숫자 4자리 변환	604	@NUM4	없음	없음	3-305
NUM8	ASCII→숫자 8자리 변환	605	@NUM8	없음	없음	3-305
NUM16	ASCII→숫자 16자리 변환	606	@NUM16	없음	없음	3-305

O

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
OR	논리합	없음	@OR	%OR	!OR	3-12
ORG	원점 검색	889	@ORG	없음	없음	3-632
OR LD	로드 논리합	없음	없음	없음	없음	3-14
OR NOT	부정 논리합	없음	없음	없음	!OR NOT	3-12
OR TST	OR 비트 테스트	350	없음	없음	없음	3-24
OR TSTN	OR 비트 테스트 부정	351	없음	없음	없음	3-24
OR<	OR 미만	310	없음	없음	없음	3-120
OR<$	OR 문자열 미만	672	없음	없음	없음	3-956
OR<=	OR 이하	315	없음	없음	없음	3-120
OR<=$	OR 문자열 이하	673	없음	없음	없음	3-956
OR<=D	OR 배정도 부동 소수점 이하	338	없음	없음	없음	3-421
OR<=DT	OR 시각 이하	344	없음	없음	없음	3-124
OR<=F	OR 단정도 부동 소수점 이하	332	없음	없음	없음	3-375
OR<=L	OR 2워드 이하	316	없음	없음	없음	3-120
OR<=S	OR 부호 있는 이하	317	없음	없음	없음	3-120
OR<=SL	OR 부호 있는 2워드 이하	318	없음	없음	없음	3-120
OR<>	OR 불일치	305	없음	없음	없음	3-120
OR<>$	OR 문자열 불일치	671	없음	없음	없음	3-956
OR<>D	OR 배정도 부동 소수점 불일치	336	없음	없음	없음	3-421
OR<>DT	OR 시각 불일치	342	없음	없음	없음	3-124
OR<>F	OR 단정도 부동 소수점 불일치	330	없음	없음	없음	3-375
OR<>L	OR 2워드 불일치	306	없음	없음	없음	3-120
OR<>S	OR 부호 있는 불일치	307	없음	없음	없음	3-120

O

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
OR<>SL	OR 부호 있는 2워드 불일치	308	없음	없음	없음	3-120
OR<D	OR 배정도 부동 소수점 미만	337	없음	없음	없음	3-421
OR<DT	OR 시각 미만	343	없음	없음	없음	3-124
OR<F	OR 단정도 부동 소수점 미만	331	없음	없음	없음	3-375
OR<L	OR 2워드 미만	311	없음	없음	없음	3-120
OR<S	OR 부호 있는 미만	312	없음	없음	없음	3-120
OR<SL	OR 부호 있는 2워드 미만	313	없음	없음	없음	3-120
OR=	OR 일치	300	없음	없음	없음	3-120
OR=$	OR 문자열 일치	670	없음	없음	없음	3-956
OR=D	OR 배정도 부동 소수점 일치	335	없음	없음	없음	3-421
OR=DT	OR 시각 일치	341	없음	없음	없음	3-124
OR=F	OR 단정도 부동 소수점 일치	329	없음	없음	없음	3-375
OR=L	OR 2워드 일치	301	없음	없음	없음	3-120
OR=S	OR 부호 있는 일치	302	없음	없음	없음	3-120
OR=SL	OR 부호 있는 2워드 일치	303	없음	없음	없음	3-120
OR>	OR 초과	320	없음	없음	없음	3-120
OR>$	OR 문자열 초과	674	없음	없음	없음	3-956
OR>=	OR 이상	325	없음	없음	없음	3-120
OR>=$	OR 문자열 이상	675	없음	없음	없음	3-956
OR>=D	OR 배정도 부동 소수점 이상	340	없음	없음	없음	3-421
OR>=DT	OR 시각 이상	346	없음	없음	없음	3-124
OR>=F	OR 단정도 부동 소수점 이상	334	없음	없음	없음	3-375
OR>=L	OR 2워드 이상	326	없음	없음	없음	3-120
OR>=S	OR 부호 있는 이상	327	없음	없음	없음	3-120
OR>=SL	OR 부호 있는 2워드 이상	328	없음	없음	없음	3-120
OR>D	OR 배정도 부동 소수점 초과	339	없음	없음	없음	3-421
OR>DT	OR 시각 초과	345	없음	없음	없음	3-124
OR>F	OR 단정도 부동 소수점 초과	333	없음	없음	없음	3-375
OR>L	OR 2워드 초과	321	없음	없음	없음	3-120
OR>S	OR 부호 있는 초과	322	없음	없음	없음	3-120
OR>SL	OR 부호 있는 2워드 초과	323	없음	없음	없음	3-120
ORW	워드 논리합	035	@ORW	없음	없음	3-312
ORWL	워드 논리합 (2워드)	611	@ORWL	없음	없음	3-312
OUT	출력	없음	없음	없음	!OUT	3-28
OUT NOT	부정 출력	없음	없음	없음	!OUT NOT	3-28
OUTB	1비트 출력	534	@OUTB	없음	!OUTB	3-46

P

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
PID	PID 연산	190	없음	없음	없음	3-518
PIDAT	자동 조정 가능한 PID 연산	191	없음	없음	없음	3-527
PLS2	위치 결정	887	@PLS2	없음	없음	3-623
PMCR	프로토콜 매크로	260	@PMCR	없음	없음	3-700
PMCR2	프로토콜 매크로 송신 2	264	@PMCR2	없음	없음	3-748
PRV	펄스 현재값 출력	881	@PRV	없음	없음	3-607
PRV2	펄스 주파수 변환	883	@PRV2	없음	없음	3-612
PULS	펄스량 설정	886	@PULS	없음	없음	3-621
PUSH	스택 데이터 저장	632	@PUSH	없음	없음	3-432
PWM	PWM 출력	891	@PWM	없음	없음	3-635
PWR	거듭제곱 연산	840	@PWR	없음	없음	3-373
PWRD	거듭제곱 연산<배>	860	@PWRD	없음	없음	3-419

R

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
RAD	각도→라디안 변환	458	@RAD	없음	없음	3-354
RADD	각도→라디안 변환<배>	849	@RADD	없음	없음	3-403
RECV	네트워크 수신	098	@RECV	없음	없음	3-783
RECV2	네트워크 수신 2	492	@RECV2	없음	없음	3-788
RET	서브루틴 반환	093	없음	없음	없음	3-569
RGHT$	오른쪽에서 문자열 가져오기	653	@RGHT$	없음	없음	3-938
RLNC	CY가 없는 1비트 좌측 회전	574	@RLNC	없음	없음	3-190
RLNL	CY가 없는 1비트 좌측 회전(2워드)	576	@RLNC	없음	없음	3-190
ROL	CY가 있는 1비트 좌측 회전	027	@ROL	없음	없음	3-188
ROLL	CY가 있는 1비트 좌측 회전(2워드)	572	@ROLL	없음	없음	3-188
ROOT	BCD 제곱근 연산	072	@ROOT	없음	없음	3-324
ROR	CY가 있는 1비트 우측 회전	028	@ROR	없음	없음	3-192
RORL	CY가 있는 1비트 우측 회전(2워드)	573	@RORL	없음	없음	3-192
ROTB	BIN 제곱근 연산	620	@ROTB	없음	없음	3-322
RPLC$	문자열 치환	661	@RPLC$	없음	없음	3-946

R

니모닉	명령 이름	평션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
RRNC	CY가 없는 1비트 우측 회전	575	@RRNC	없음	없음	3-194
RRNL	CY가 없는 1비트 우측 회전(2워드)	577	@RRNL	없음	없음	3-194
RSET	리셋	없음	@RSET	%RSET	!RSET	3-40
RSORT	부호 없는 1W 레코드 정렬	203	@RSORT	없음	없음	3-507
RSORT2	부호 없는 2W 레코드 정렬	204	@RSORT 2	없음	없음	3-511
RSORT4	부호 없는 4W 레코드 정렬	205	@RSORT 4	없음	없음	3-514
RSRCH <	부호 없는 1W 레코드 검색, 미만	360	@RSRCH <	없음	없음	3-495
RSRCH <=	부호 없는 1W 레코드 검색, 이하	361	@RSRCH <=	없음	없음	3-495
RSRCH =	부호 없는 1W 레코드 검색, 일치	362	@RSRCH =	없음	없음	3-495
RSRCH >=	부호 없는 1W 레코드 검색, 이상	363	@RSRCH >=	없음	없음	3-495
RSRCH >	부호 없는 1W 레코드 검색, 초과	364	@RSRCH >	없음	없음	3-495
RSRCH2 <	부호 없는 2W 레코드 검색, 미만	370	@RSRCH 2 <	없음	없음	3-501
RSRCH2 <=	부호 없는 2W 레코드 검색, 이하	371	@RSRCH 2 <=	없음	없음	3-501
RSRCH2 =	부호 없는 2W 레코드 검색, 일치	372	@RSRCH 2 =	없음	없음	3-501
RSRCH2 >=	부호 없는 2W 레코드 검색, 이상	373	@RSRCH 2 >=	없음	없음	3-501
RSRCH2 >	부호 없는 2W 레코드 검색, 초과	374	@RSRCH 2 >	없음	없음	3-501
RSRCH4 <	부호 없는 4W 레코드 검색, 미만	380	@RSRCH 4 <	없음	없음	3-504
RSRCH4 <=	부호 없는 4W 레코드 검색, 이하	381	@RSRCH 4 <=	없음	없음	3-504
RSRCH4 =	부호 없는 4W 레코드 검색, 일치	382	@RSRCH 4 =	없음	없음	3-504
RSRCH4 >=	부호 없는 4W 레코드 검색, 이상	383	@RSRCH 4 >=	없음	없음	3-504
RSRCH4 >	부호 없는 4W 레코드 검색, 초과	384	@RSRCH 4 >	없음	없음	3-504
RSTA	다중 비트 리셋	531	@RSTA	없음	없음	3-42
RSTB	1비트 리셋	533	@RSTB	없음	!RSTB	3-44
RXD	CPU 유니트 시리얼 커뮤니케이션 보드 시리얼 포트 입력	235	@RXD	없음	없음	3-712
RXDU	시리얼 커뮤니케이션 유니트 시리얼 포트 입력	255	@RXDU	없음	없음	3-726

S

니모닉	명령 이름	평션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
SA	스텝 활성화 명령	784	@SA	없음	없음	3-990
SBN	서브루틴 진입	092	없음	없음	없음	3-569
SBS	서브루틴 호출	091	@SBS	없음	없음	3-561
SCL	스케일링	194	@SCL	없음	없음	3-546
SCL2	스케일링 2	486	@SCL2	없음	없음	3-550
SCL3	스케일링 3	487	@SCL3	없음	없음	3-553
SDEC	7세그먼트 디코더	078	@SDEC	없음	없음	3-659
SDEL	스택 데이터 삭제	642	@SDEL	없음	없음	3-449
SE	스텝 비활성화 명령	785	@SE	없음	없음	3-990
SEC	시분초→초 변환	065	@SEC	없음	없음	3-856
SEND	네트워크 송신	090	@SEND	없음	없음	3-774
SEND2	네트워크 송신 2	491	@SEND2	없음	없음	3-779
SET	세트	없음	@SET	%SET	!SET	3-40
SETA	다중 비트 세트	530	@SETA	없음	없음	3-42
SETB	1비트 세트	532	@SETB	없음	!SETB	3-44
SETR	레코드 위치 설정	635	@SETR	없음	없음	3-454
SFCOFF	SFC 태스크 종료 명령	790	없음	없음	없음	3-994
SFCON	SFC 태스크 실행 명령	789	없음	없음	없음	3-994
SFCPR	SFC 태스크 정지 명령 (리셋 있음)	793	없음	없음	없음	3-996
SFCPRN	SFC 태스크 정지 명령 (리셋 없음)	791	없음	없음	없음	3-996
SFT	이동 레지스터	010	없음	없음	없음	3-176
SFTR	좌우 이동 레지스터	084	@SFTR	없음	없음	3-178
SIGN	부호 확장	600	@SIGN	없음	없음	3-263
SIN	SIN 연산	460	@SIN	없음	없음	3-358
SIND	SIN 연산 <배>	851	@SIND	없음	없음	3-407
SINQ	고속 SIN 연산	475	@SINQ	없음	없음	3-361
SINS	스택 데이터 삽입	641	@SINS	없음	없음	3-446
SLD	1자리 좌측 이동	074	@SLD	없음	없음	3-196
SNUM	스택 데이터 수 출력	638	@SNUM	없음	없음	3-438
SNXT	스텝 래더 영역 진행	009	없음	없음	없음	3-639
SPED	주파수 설정	885	@SPED	없음	없음	3-618
SQRT	제곱근 연산	466	@SQRT	없음	없음	3-367
SQRTD	제곱근 연산 <배>	857	@SQRTD	없음	없음	3-413
SRCH	데이터 검색	181	@SRCH	없음	없음	3-458
SRD	1자리 우측 이동	075	@SRD	없음	없음	3-196
SREAD	스택 데이터 참조	639	@SREAD	없음	없음	3-440

S

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
SSET	스택 영역 설정	630	@SSET	없음	없음	3-430
STC	세트 캐리	040	@STC	없음	없음	3-888
STEP	스텝 래더 영역 정의	008	없음	없음	없음	3-639
STR4	숫자 4자리 →ASCII 변환	601	@STR4	없음	없음	3-302
STR8	숫자 8자리 →ASCII 변환	602	@STR8	없음	없음	3-302
STR16	숫자 16자리 →ASCII 변환	603	@STR16	없음	없음	3-302
STUP	시리얼 포트 통신 설정 변경	237	@STUP	없음	없음	3-745
SUM	SUM 값 산출	184	@SUM	없음	없음	3-480
SWAP	바이트 스와핑	637	@SWAP	없음	없음	3-461
SWRIT	스택 데이터 갱신	640	@SWRIT	없음	없음	3-443

T

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
TAN	TAN 연산	462	@TAN	없음	없음	3-358
TAND	TAN 연산 <배>	853	@TAND	없음	없음	3-407
TANQ	고속 TAN 연산	477	@TANQ	없음	없음	3-361
TCMP	테이블 일치	085	@TCMP	없음	없음	3-136
TIM	100ms 타이머	없음	없음	없음	없음	3-88
TIMH	10ms 타이머	015	없음	없음	없음	3-91
TIMHX	10ms 타이머	551	없음	없음	없음	3-91
TIML	장시간 타이머	542	없음	없음	없음	3-103
TIMLX	장시간 타이머	553	없음	없음	없음	3-103
TIMU	0.1ms 타이머	541	없음	없음	없음	3-96
TIMUX	0.1ms 타이머	556	없음	없음	없음	3-96
TIMW	타이머 대기	813	없음	없음	없음	3-920
TIMWX	타이머 대기	816	없음	없음	없음	3-920
TIMX	100ms 타이머	550	없음	없음	없음	3-88
TKOF	태스크 실행 대기	821	@TKOF	없음	없음	3-962
TKON	태스크 실행 기동	820	@TKON	없음	없음	3-962
TKY	10키 입력	211	@TKY	없음	없음	3-664
TMHH	1ms 타이머	540	없음	없음	없음	3-94
TMHHX	1ms 타이머	552	없음	없음	없음	3-94
TMHW	고속 타이머 대기	815	없음	없음	없음	3-925
TMHWX	고속 타이머 대기	817	없음	없음	없음	3-925
TMUH	0.01ms 타이머	544	없음	없음	없음	3-98
TMUHX	0.01ms 타이머	557	없음	없음	없음	3-98
TOCV	CS→CV 주소 변환	285	@TOCV	없음	없음	3-898
TPO	시분할 비례 출력	685	없음	없음	없음	3-540
TR	일시 기억 릴레이	없음	없음	없음	없음	3-30
TRSET	타이머 리셋	549	@TRSET	없음	없음	3-117
TRSM	트레이스 메모리 샘플링	045	없음	없음	없음	3-864
TSR	스텝 타이머 현재값 읽기 명령	780	@TSR	없음	없음	3-992
TST	비트 테스트	350	없음	없음	없음	3-22
TSTN	비트 테스트 부정	351	없음	없음	없음	3-22
TSW	스텝 타이머 현재값 변경 명령	781	@TSW	없음	없음	3-992
TTIM	적산 타이머	087	없음	없음	없음	3-100
TTIMX	적산 타이머	555	없음	없음	없음	3-100
TWRIT	텍스트 파일 쓰기	704	@TWRIT	없음	없음	3-841
TXD	CPU 유니트 시리얼 커뮤니케이션 보드 시리얼 포트 출력	236	@TXD	없음	없음	3-706
TXDU	시리얼 커뮤니케이션 유니트 시리얼 포트 출력	256	@TXDU	없음	없음	3-720

U

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
UP	P.F. 기동 미분	521	없음	없음	없음	3-18

W

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
WAIT NOT 릴레이 번호	1스캔 조건 대기(부정)	805	없음	없음	없음	3-917
입력 조건 WAIT	1스캔 조건 대기	805	없음	없음	없음	3-917
WAIT 릴레이 번호	1스캔 조건 대기	805	없음	없음	없음	3-917
WDT	사이클 시간 감시 시간 설정	094	@WDT	없음	없음	3-891
WSFT	워드 이동	016	@WSFT	없음	없음	3-182

X

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
XCGL	데이터 2워드 교환	562	@XCGL	없음	없음	3-166
XCHG	데이터 교환	073	@XCHG	없음	없음	3-166
XCHG$	문자열 교환	665	@XCHG$	없음	없음	3-950
XFER	블록 전송	070	@XFER	없음	없음	3-162
XFRB	다중 비트 전송	062	@XFRB	없음	없음	3-160
XFERC	블록 전송	565	@ XFERC	없음	없음	3-970
XNRL	워드 배타적 논리합 부정(2워드)	613	@XNRL	없음	없음	3-316
XNRW	워드 배타적 논리합 부정	037	@XNRW	없음	없음	3-316
XORL	워드 배타적 논리합(2워드)	612	@XORL	없음	없음	3-314
XORW	워드 배타적 논리합	036	@XORW	없음	없음	3-314

Z

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
ZCP	영역 비교	088	없음	없음	없음	3-143
ZCPL	2워드 영역 비교	116	없음	없음	없음	3-143
ZONE	데드 존 제어	682	@ZONE	없음	없음	3-538

기타 기호

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
7SEG	7세그먼트 표시	214	없음	없음	없음	3-674
+	부호 있음, CY 없는 BIN 가산	400	@+	없음	없음	3-222
+$	문자열 연결	656	@+$	없음	없음	3-936
++	BIN 증가	590	@++	없음	없음	3-208
++B	BCD 증가	594	@++B	없음	없음	3-214
++BL	BCD 2워드 증가	595	@++BL	없음	없음	3-214
++L	BIN 2워드 증가	591	@++L	없음	없음	3-208
+B	CY 없는 BCD 가산	404	@+B	없음	없음	3-227
+BC	CY 있는 BCD 가산	406	@+BC	없음	없음	3-229
+BCL	CY 있는 BCD 2워드 가산	407	@+BCL	없음	없음	3-229
+BL	CY 없는 BCD 2워드 가산	405	@+BL	없음	없음	3-227
+C	부호 및 CY 있는 BIN 가산	402	@+C	없음	없음	3-224
+CL	부호 및 CY 있는 BIN 2워드 가산	403	@+CL	없음	없음	3-224
+D	부동 소수점 가산<배>	845	@+D	없음	없음	3-400
+F	부동 소수점 가산	454	@+F	없음	없음	3-351
+L	부호 있음, CY 없는 BIN 2워드 가산	401	@+L	없음	없음	3-222
-	부호 있음, CY 없는 BIN 감산	410	@-	없음	없음	3-231
--	BIN 감소	592	@--	없음	없음	3-211
--B	BCD 감소	596	@--B	없음	없음	3-217
--BL	BCD 2워드 감소	597	@--BL	없음	없음	3-217
--L	BIN 2워드 감소	593	@--L	없음	없음	3-211
-B	CY 없는 BCD 감산	414	@-B	없음	없음	3-238
-BC	CY 있는 BCD 감산	416	@-BC	없음	없음	3-241
-BCL	CY 있는 BCD 2워드 감산	417	@-BCL	없음	없음	3-241
-BL	CY 없는 BCD 2워드 감산	415	@-BL	없음	없음	3-238
-C	부호 및 CY 있는 BIN 감산	412	@-C	없음	없음	3-235
-CL	부호 및 CY 있는 BIN 2워드 감산	413	@-CL	없음	없음	3-235
-D	부동 소수점 감산<배>	846	@-D	없음	없음	3-400
-F	부동 소수점 감산	455	@-F	없음	없음	3-351
-L	부호 있음, CY 없는 BIN 2워드 감산	411	@-L	없음	없음	3-231
*	부호 있는 BIN 승산	420	@*	없음	없음	3-243
*B	BCD 승산	424	@*B	없음	없음	3-247
*BL	BCD 2워드 승산	425	@*BL	없음	없음	3-247
*D	부동 소수점 승산<배>	847	@*D	없음	없음	3-400
*F	부동 소수점 승산	456	@*F	없음	없음	3-351
*L	부호 있는 BIN 2워드 승산	421	@*L	없음	없음	3-243
*U	부호 없는 BIN 승산	422	@*U	없음	없음	3-245
*UL	부호 없는 BIN 2워드 승산	423	@*UL	없음	없음	3-245
/	부호 있는 BIN 제산	430	@/	없음	없음	3-249
/B	BCD 제산	434	@/B	없음	없음	3-253
/BL	BCD 2워드 제산	435	@/BL	없음	없음	3-253
/D	부동 소수점 제산<배>	848	@/D	없음	없음	3-400
/F	부동 소수점 제산	457	@/F	없음	없음	3-351
/L	부호 있는 BIN 2워드 제산	431	@/L	없음	없음	3-249

기타 기호

니모닉	명령 이름	펑션 번호	기동 지정	종료 지정	매회 리프레시 지정	페이지
/U	부호 없는 BIN 제산	432	@/U	없음	없음	3-251
/UL	부호 없는 BIN 2워드 제산	433	@/UL	없음	없음	3-251

ASCII 코드 목록표

하위 4비트 \ 상위 4비트	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0			SP	0	@	P	`	p				ー	タ	ミ		
1			!	1	A	Q	a	q			。	ア	チ	ム		
2			″	2	B	R	b	r			「	イ	ツ	メ		
3			#	3	C	S	c	s			」	ウ	テ	モ		
4			$	4	D	T	d	t			、	エ	ト	ヤ		
5			%	5	E	U	e	u			・	オ	ナ	ユ		
6			&	6	F	V	f	v			ヲ	カ	ニ	ヨ		
7			′	7	G	W	g	w			ァ	キ	ヌ	ラ		
8			(	8	H	X	h	x			ィ	ク	ネ	リ		
9			)	9	I	Y	i	y			ゥ	ケ	ノ	ル		
A			*	:	J	Z	j	z			ェ	コ	ハ	レ		
B			+	;	K	[	k	{			ォ	サ	ヒ	ロ		
C			,	<	L	¥	l	|			ャ	シ	フ	ワ		
D			ー	=	M	]	m	}			ュ	ス	ヘ	ン		
E			.	>	N	^	n	~			ョ	セ	ホ	゛		
F			／	?	O	_	o				ッ	ソ	マ	゜		

본지는 주로 종류 선정에 필요한 내용을 게재하고 있으며, 사용상의 주의 사항 등은 게재되어 있지 않습니다.
사용하실 때에는 반드시 제품에 동봉되어 있는 매뉴얼을 읽어주시기 바랍니다.

- 이 카탈로그에 기재된 활용 사례는 참고용이므로 채택하실 때는 기기 및 장치 기능이나 안전성을 확인한 후 사용해주십시오.
- 이 카탈로그에 기재되지 않은 조건이나 환경에서 사용하거나 원자력 제어, 철도, 항공, 차량, 연소 장치, 의료 기기, 오락 기계, 안전 기기, 그 밖에 인명이나 재산에 큰 영향을 미치는 등 안전성이 각별히 요구되는 용도로 사용할 것을 검토하는 경우는 정격이나 성능에 여유가 있는 사용 방법이나 장애 시 안전 대책 등을 마련하는 한편, 당사 영업 담당자에게 상담하여 사양서 등을 확인하시기 바랍니다.

Industrial Web ▶ http://www.ia.omron.co.kr

한국 오므론 제어기기 주식회사

137-920 서울시 서초구 서초동 1303-22 교보타워 B동 21층 TEL : 02-3483-7789 FAX : 02-3483-7788

카탈로그 번호 SBCA-351G-K6 2010년 9월 현재

© OMRON Corporation 2008-2010 All Rights Reserved.
예고없이 사양 등을 변경하는 경우가 있으므로 양해 부탁드립니다.