CPU 유니트
전원 유니트
기본 I/O 유니트

OMRON

PLC

SYSMAC CJ 시리즈

CJ2 CPU 유니트

CJ2H-CPU6□-EIP
CJ2H-CPU6□
CJ2M-CPU□□

사용 설명서
소프트웨어편

Man. No. SBCA-350F-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.

머리말

CJ 시리즈 PLC CJ2H-CPU6□(-EIP), CJ2M-CPU□□를 구입해 주셔서 감사합니다.
이 설명서에서는 CJ2H-CPU6□(-EIP), CJ2M-CPU1/3□을 사용하는 데 필요한 정보를 소개합니다.
이 설명서를 잘 읽고 충분히 숙지한 후 사용하십시오.

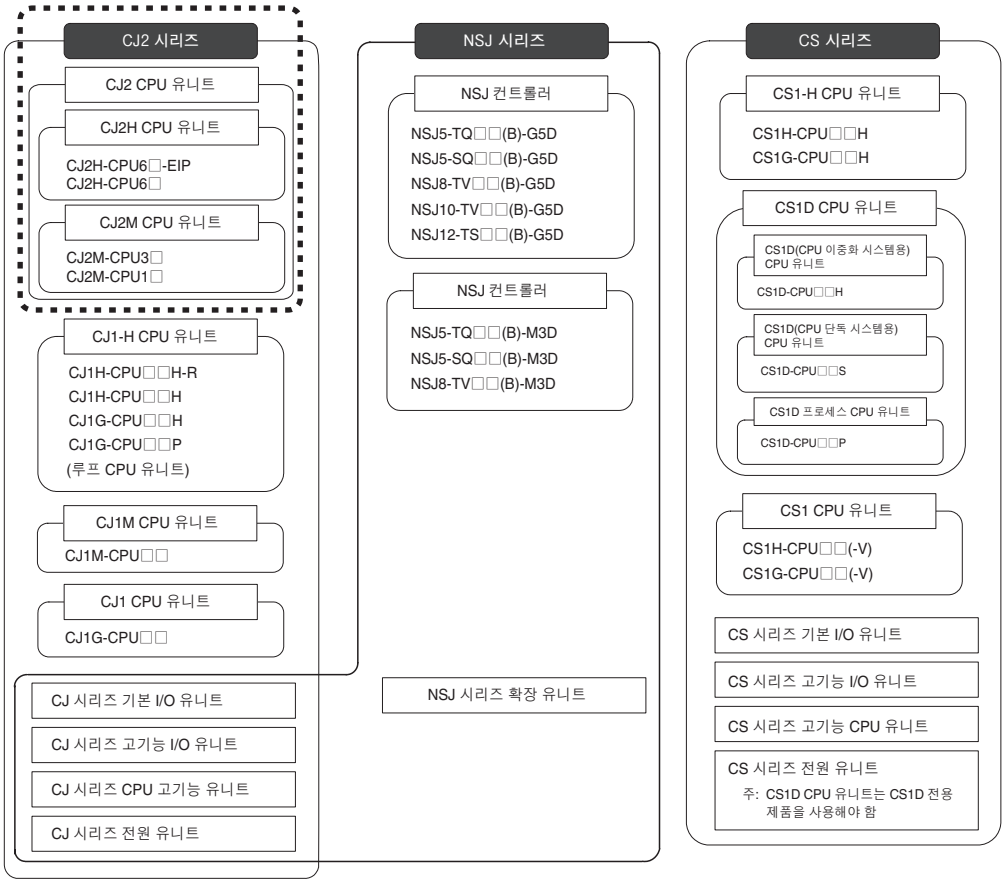
대상 독자

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

대상 제품

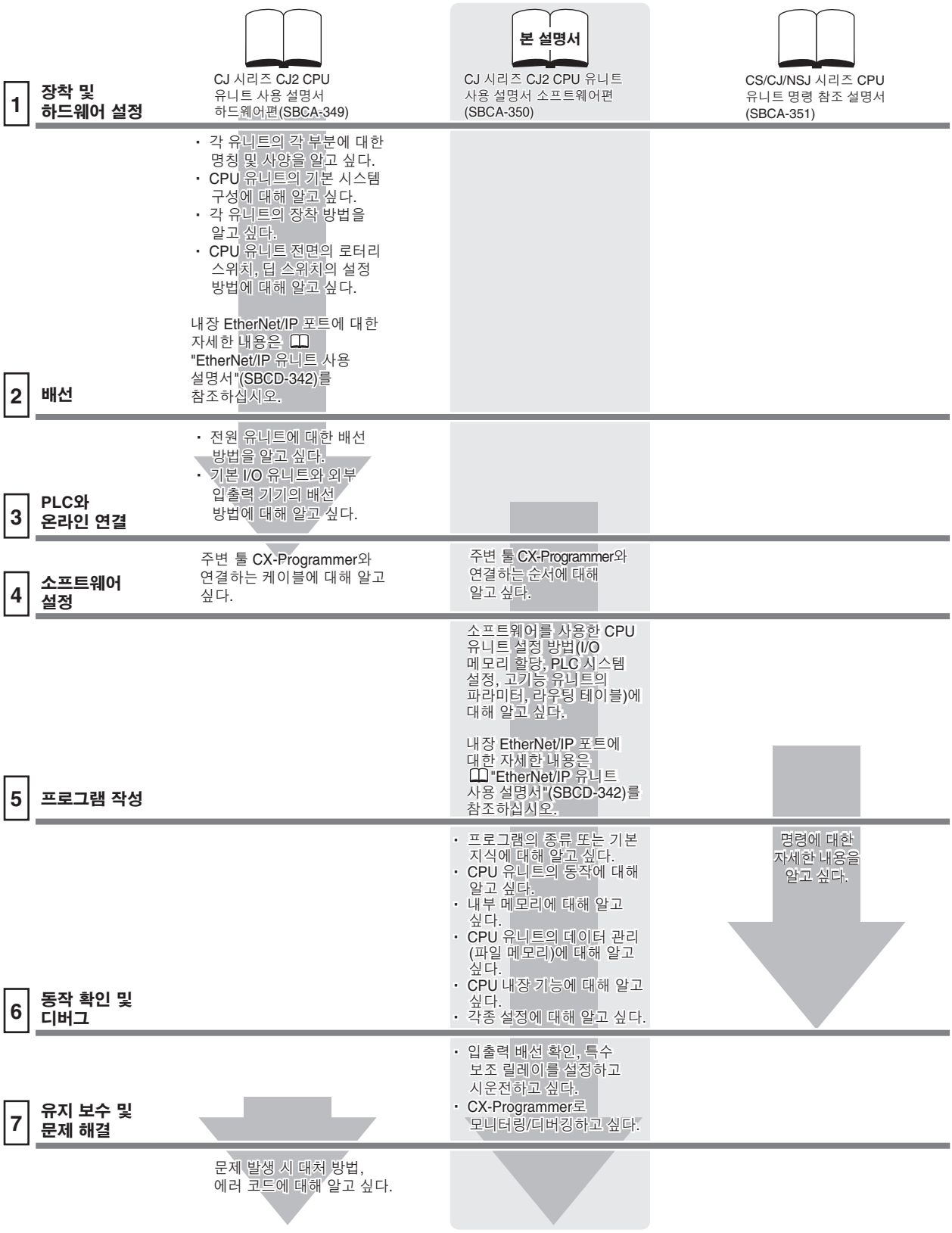
이 설명서에서 다루는 CJ2 CPU 유니트는 다음과 같은 CPU 유니트를 말합니다.
- CJ2H-CPU6□-EIP
- CJ2H-CPU6□
- CJ2M-CPU3□
- CJ2M-CPU1□

이 설명서에서는 CPU 유니트를 총칭하여 다음과 같이 표기하는 경우가 있습니다.
- CJ2H-CPU6□(-EIP): CJ2H-CPU6□-EIP 및 CJ2H-CPU6□ 총칭
- CJ2M-CPU□□: CJ2M-CPU3□ 및 CJ2M-CPU1□ 총칭



각 설명서별 구성

본 제품의 설명서는 다음과 같이 구성되었습니다.
목적에 따라 해당 설명서를 읽은 후 본 제품을 이용하시기 바랍니다.



각 장의 구성

CJ2 CPU 유니트의 설명서는 다음과 같은 장으로 구성되었습니다. 각각의 사용 상황에 맞춰 읽어보십시오.

사용 설명서 하드웨어편(설명서 번호: SBCA-349)

장	내용
1장 개요	CJ2 CPU 유니트의 개요, 특징, 사양 등에 대해 설명합니다.
2장 기본 시스템 및 구성 기기	CJ2 CPU 유니트의 시스템 구성에 대해 설명합니다.
3장 각 부분의 명칭과 기능	CPU 유니트와 구성 유니트 각 부분의 명칭과 기능에 대해 설명합니다.
4장 주변 툴	PLC 프로그래밍 및 디버그를 수행하는 주변 툴의 종류, PLC와 주변 툴을 연결하는 방법 등을 설명합니다.
5장 부착 및 설정	CPU 유니트, 구성 유니트의 부착과 배선 방법 또는 설치 장소에 대해 설명합니다.
6장 문제 해결	시스템 사용 시 발생하는 이상 상태의 확인 및 대처 방법에 대해 설명합니다.
7장 보수 점검	정기 점검 내용이나 배터리, 전원 유니트의 수명 또는 교환 방법에 대해 설명합니다.
8장 백업 조작	PLC의 데이터를 백업하는 조작에 대해 설명합니다.
부록	부하 단락 보호 기능과 단선 검출 기능, 유니트의 외형 규격, 운전 정지 이상/운전 계속 이상에 대한 자세한 내용, CPU 유니트의 시리얼 포트에 연결하는 방법, USB 드라이버의 설치 순서 등에 대해 설명합니다.

사용 설명서 소프트웨어편(설명서 번호: SBCA-350) (본 설명서)

장	내용
1장 개요	CJ2 CPU 유니트의 개요, 특징, 사양 등에 대해 설명합니다.
2장 CPU 유니트의 내부 메모리	CPU 내부 메모리의 종류와 저장된 데이터에 대해 설명합니다.
3장 CPU 유니트의 동작	CPU 유니트의 내부 동작에 대해 설명합니다.
4장 CPU 유니트의 초기 설정	CPU 유니트의 초기 설정에 대해 설명합니다.
5장 프로그램 지식	프로그램의 종류나 프로그램을 구성하는 변수 또는 명령 등, 프로그램에 대해 자세히 설명합니다.
6장 I/O 메모리	I/O 메모리 내 각 영역의 종류와 자세한 내용에 대해 설명합니다.
7장 파일 조작	CPU 유니트에 저장할 수 있는 파일의 종류와 저장 장소, 파일 조작에 대해 설명합니다.
8장 I/O 할당 및 유니트 설정	CPU 유니트 및 유니트와 데이터를 교환하는 I/O 할당에 대한 자세한 내용, 각 유니트의 설정 등에 대해 설명합니다.
9장 PLC 시스템 설정	CPU 유니트의 기본 설정을 수행하는 PLC 시스템 설정에 대한 자세한 내용을 설명합니다.
10장 CPU 내장 기능	CPU 유니트 본체에서 사용할 수 있는 기능에 대해 설명합니다.
11장 주변 툴/기기와 연결 기능	CJ2 CPU 유니트와 CX-Programmer 등의 주변 툴 또는 기타 기기를 연결하는 방법을 설명합니다.
12장 CPU와 사이클 시간	사이클 시간의 모니터링 방법과 계산 방법을 설명합니다.
부록	프로그램에서 사용할 수 있는 명령 목록, 명령 처리 시간/스텝 수 목록, 특수 보조 릴레이 목록, I/O 메모리의 실효 메모리 주소, 전원 OFF 시 CPU 유니트 동작, CJ/CS 시리즈의 비교 결과 등을 설명합니다.

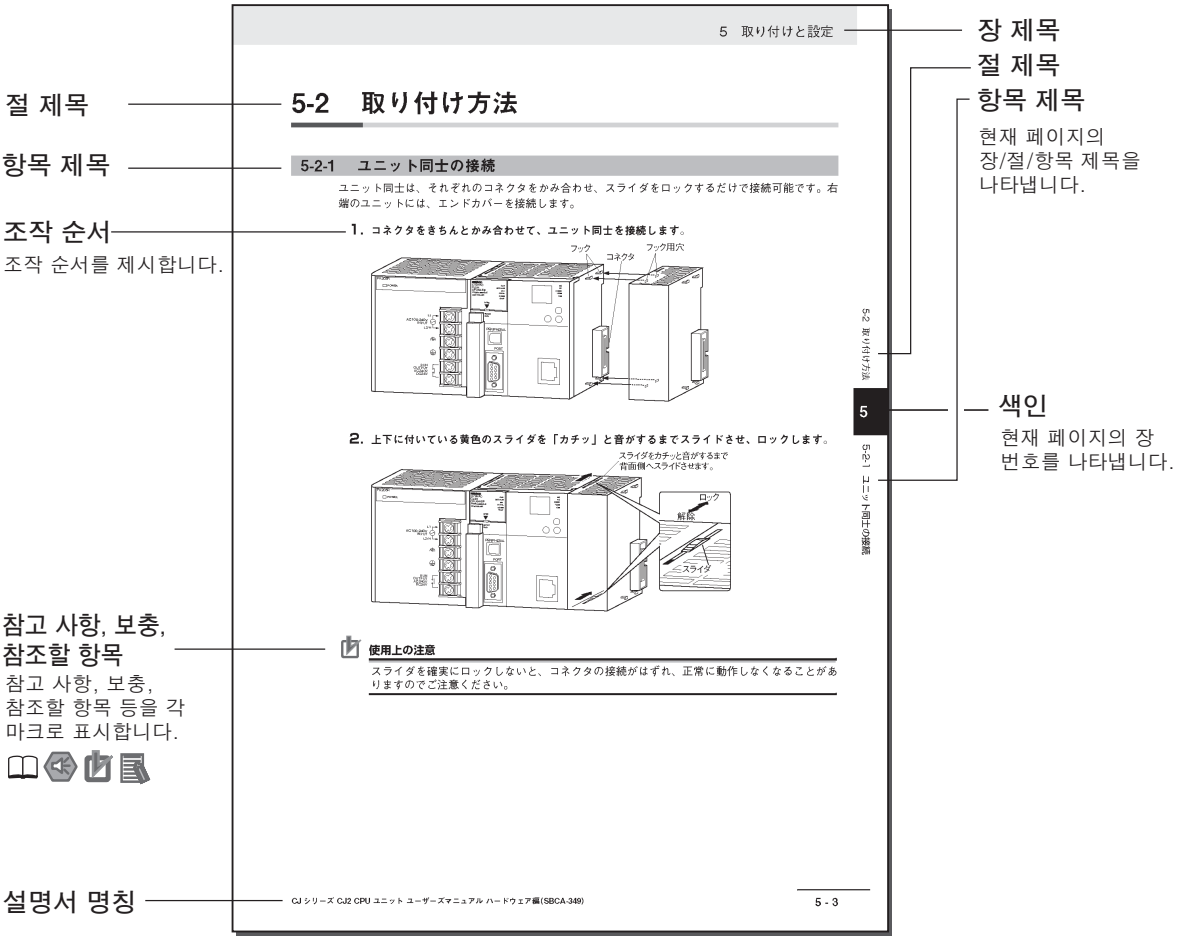
명령 참조 설명서(설명서 번호: SBCA-351)

장	내용
제1장 명령어의 기본 지식	CS/CJ/NSJ 시리즈 CPU 유니트의 래더 프로그램 설계에 대한 기초 지식을 설명합니다.
제2장 명령 목록	CS/CJ/NSJ 시리즈 CPU 유니트에서 사용할 수 있는 명령 목록을 설명합니다.
제3장 각 명령에 대한 설명	CS/CJ/NSJ 시리즈 CPU 유니트에서 사용할 수 있는 명령의 기능, 피연산자의 내용 및 샘플 프로그램에 대해 설명합니다.
제4장 명령 처리 시간/스텝 수 목록	CS/CJ/NSJ 시리즈 CPU 유니트에서 사용하는 각 명령의 처리 시간에 대해 설명합니다.
부록	CS/CJ/NSJ 시리즈 CPU 유니트의 명령을 펑션 번호 순서 및 니모닉 순서로 정렬한 목록, ASCII 코드 목록에 대해 설명합니다.

설명서 읽는 방법

페이지 구성과 기호

본 설명서의 각 페이지 구성과 기호는 다음과 같습니다.



이 페이지는 설명용 예이므로 실제 내용과 다릅니다.

마크

본 설명서에서 사용하는 마크에는 다음과 같은 의미가 있습니다.

안전상의 중요 사항
제품을 안전하게 사용하기 위해 실시하거나 피해야 하는 사항을 나타냅니다.

사용상의 주의 사항
제품의 동작 불능, 오작동 또는 성능 및 기능에 미치는 악영향을 예방하기 위해 실시하거나 피해야 하는 사항을 나타냅니다.

참고
필요에 따라 읽어야 하는 항목입니다.
알아두면 편리한 정보나 사용 시 참고가 되는 내용에 대해 설명합니다.

자세한 정보 또는 관련 정보가 기재되어 있는 위치를 나타냅니다.

목차 구성

1 개요

2 CPU 유니트 내부 메모리

3 CPU 유니트의 동작

4 CPU 유니트의 초기 설정

5 프로그램 지식

6 I/O 메모리

7 파일 조작

8 I/O 할당 및 유니트 설정

9 PLC 시스템 설정

10 CPU 내장 기능

11 주변 툴/기기와 연결

12 CPU와 사이클 시간

부록 부록

목차

머리말 1
대상 독자 1
대상 제품 1
각 설명서별 구성 2
각 장의 구성 3
설명서 읽는 방법 5
페이지 구성과 기호 5
마크 5
목차 구성 7
사용 시 동의 사항 15
안전상의 주의 사항 17
안전한 사용을 위한 표시 및 의미 17
그림 기호 설명 18
안전상의 중요 사항 21
사용상의 주의 사항 25
법규와 규격 26
해외에서 사용하는 경우 26
EC 지령에 대한 적합성 26
선박 규격에 대한 적합성 27
상표 27
CJ2 CPU 유니트의 유니트 버전 28
유니트 버전 28
유니트 버전의 종류 31
관련 설명서 33

제1장 개요 1-1

1-1 CJ2 CPU 유니트의 개요 1-2
1-1-1 개요 1-2
1-1-2 CJ2 CPU 유니트의 특징 1-4
1-2 기본적인 사용 순서 1-10

제2장 CPU 유니트 내부 메모리 2-1

2-1 CPU 유니트의 내부 메모리 2-2
2-1-1 CPU 유니트의 메모리 구성 2-2
2-1-2 메모리 영역과 저장 데이터 2-3
2-1-3 주변 툴에서 데이터 전송 2-4

제3장 CPU 유니트의 동작 3-1

3-1 CPU 유니트의 내부 동작 3-2
3-1-1 내부 동작의 개요 3-2
3-1-2 사이클 시간 3-4
3-1-3 전원 OFF 시의 처리 3-7
3-2 유니트의 동작 모드 3-8
3-2-1 동작 모드 3-8
3-2-2 동작 모드 확인 방법 3-9
3-2-3 동작 모드의 전환 3-10
3-2-4 동작 모드의 자세한 내용 3-14

제4장 CPU 유니트의 초기 설정 4-1

4-1 CPU 유니트의 초기 설정에 대한 개요 4-2
4-1-1 CPU 유니트의 초기 설정에 대한 개요 4-2

4-2 PLC 시스템 설정 4-7

4-3 I/O 테이블 작성 4-8
4-3-1 I/O 테이블 4-8
4-3-2 자동 할당 4-9
4-3-3 수동 할당 4-9

4-4 라우팅 테이블 설정 4-10
4-4-1 라우팅 테이블 4-10
4-4-2 라우팅 테이블이 필요한 경우 4-12
4-4-3 라우팅 테이블의 설정/전송 방법 4-13

4-5 고기능 유니트용 할당 DM 영역 설정 4-14
4-5-1 고기능 유니트용 할당 DM 영역 설정 4-14
4-5-2 고기능 유니트용 할당 DM 영역 설정 방법 4-14

4-6 CPU 고기능 유니트 시스템 설정 영역 4-15
4-6-1 CPU 고기능 유니트 시스템 설정 영역 4-15
4-6-2 CPU 고기능 유니트 시스템 설정 영역의 설정 방법 4-15

제5장 프로그램 지식 5-1

5-1 프로그램 5-3
5-1-1 프로그램 언어 5-3
5-1-2 래더 언어 5-6
5-1-3 ST 언어 5-8
5-1-4 SFC 언어 5-9

5-2 태스크 5-11
5-2-1 태스크 5-11
5-2-2 사이클릭 태스크 5-14
5-2-3 인터럽트 태스크 5-19
5-2-4 태스크 설계 방법 5-30

5-3 섹션 5-40
5-3-1 섹션이란? 5-40

5-4 펑션 블록(FB) 기능 5-42
5-4-1 펑션 블록이란? 5-42
5-4-2 펑션 블록의 특징 5-43
5-4-3 펑션 블록 관련 사양 5-44

5-5 변수 5-47
5-5-1 변수란? 5-47
5-5-2 변수의 종류 5-48
5-5-3 글로벌 변수 5-50
5-5-4 로컬 변수 5-50
5-5-5 네트워크 변수(CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당) 5-51
5-5-6 펑션 블록 내 변수 5-55
5-5-7 변수의 데이터 형식 5-56
5-5-8 변수의 주소 자동 할당 5-61

5-6 명령 5-62
5-6-1 명령의 기본 지식 5-62
5-6-2 피연산자 지정 방법 5-70
5-6-3 데이터 형식 5-77
5-6-4 I/O 리프레시 타이밍 5-80

5-7 인덱스 레지스터 기능 5-85
5-7-1 인덱스 레지스터란? 5-85
5-7-2 인덱스 레지스터 사용 방법 5-86
5-7-3 인덱스 레지스터 관련 처리 5-89
5-7-4 인덱스 레지스터를 모니터링하는 방법 5-90
5-7-5 인덱스 레지스터/데이터 레지스터의 태스크 간 공통화 5-91
5-8 주소의 오프셋 지정 5-92
5-8-1 주소의 오프셋 지정이란? 5-92
5-8-2 주소의 오프셋 용도의 예 5-95
5-9 작성한 프로그램 확인 5-96
5-9-1 CX-Programmer의 입력 조작 시 등의 확인 5-96
5-9-2 CX-Programmer에서 프로그램 확인 5-96
5-9-3 시뮬레이터 기능을 사용한 디버그 5-97
5-9-4 명령 실행 시 확인 5-100
5-10 프로그램 작성 시 주의 사항 5-103
5-10-1 조건 플래그 사용 시 주의 사항 5-103
5-10-2 특정 프로그램 영역에 관한 주의 사항 5-109

제6장 I/O 메모리 6-1

6-1 I/O 메모리 영역의 개요 6-2
6-1-1 I/O 메모리 영역이란? 6-2
6-1-2 I/O 메모리 영역 목록 6-3
6-1-3 I/O 메모리 값의 유지 6-5
6-2 입출력 릴레이 6-7
6-3 데이터 링크 릴레이 6-11
6-4 동기 데이터 링크 릴레이 6-12
6-5 CPU 고기능 유니트 릴레이 6-13
6-6 고기능 I/O 유니트 릴레이 6-14
6-7 시리얼 PLC 링크 릴레이 6-15
6-8 DeviceNet 릴레이 6-16
6-9 내부 보조 릴레이 6-17
6-10 유지 릴레이 6-18
6-11 특수 보조 릴레이 6-19
6-12 일시 기억 릴레이 6-20
6-13 데이터 메모리 6-21
6-14 확장 데이터 메모리 6-24
6-15 타이머 6-28
6-16 카운터 6-30
6-17 태스크 플래그 6-31
6-18 인덱스 레지스터 6-32
6-19 데이터 레지스터 6-37
6-20 조건 플래그 6-39
6-21 클록 펄스 6-41

제7장 파일 조작 7-1

7-1 파일 메모리란? 7-2
7-1-1 파일 메모리의 매체 7-2
7-1-2 매체의 초기화 7-3
7-1-3 메모리 카드를 사용하는 경우 주의 사항 7-5

7-2 파일 메모리용 파일 종류 7-8
7-2-1 파일 종류 7-8
7-2-2 파일 작성 및 매체에 저장하는 순서 7-11

7-3 파일 메모리 조작 방법 7-12
7-3-1 파일 메모리 조작의 종류 7-12
7-3-2 파일 메모리 조작 및 파일 7-14
7-3-3 파일의 사용 제한 7-21
7-3-4 파일 크기 7-21
7-3-5 주변 툴과 파일 메모리의 관계 7-22

제8장 I/O 할당 및 유니트 설정 8-1

8-1 I/O 할당 8-2
8-1-1 I/O 할당이란? 8-2
8-1-2 자동 할당 8-5
8-1-3 수동 할당 8-9
8-1-4 I/O 테이블 관련 이상과 주의 사항 8-17

8-2 고기능 유니트 설정 8-20
8-2-1 파라미터 설정 8-20
8-2-2 CPU 유니트와 데이터 교환 8-24

제9장 PLC 시스템 설정 9-1

9-1 PLC 시스템 설정의 개요 9-2

9-2 PLC 시스템 설정 목록 9-3

9-3 PLC 시스템 설정의 설명 9-4
9-3-1 전원 ON 시 동작 설정 9-4
9-3-2 CPU 유니트 설정 9-7
9-3-3 타이머/인터럽트/동기 제어 9-12
9-3-4 고기능 I/O 리프레시 9-17
9-3-5 유니트 설정 9-19
9-3-6 시리얼 포트 9-21
9-3-7 주변 서비스 9-27
9-3-8 FINS 금지 9-28

제10장 CPU 내장 기능 10-1

10-1 시계 기능 10-3
10-1-1 시계 기능 10-3
10-1-2 시각 기록 기능 10-4
10-1-3 프리 러닝 타이머 기능 10-6

10-2 사이클 시간 관련/고속 처리 기능 10-7
10-2-1 사이클 시간 일정화 기능 10-7
10-2-2 사이클 시간 감시 기능 10-8
10-2-3 사이클 시간 모니터링 기능 10-9
10-2-4 펄스 캐치 기능 10-9
10-2-5 백그라운드 처리 실행 기능 10-10
10-2-6 고속 인터럽트 기능 10-18

10-3 운전 시작 시 동작 설정 및 보수 기능 10-20
10-3-1 동작 모드 변경 및 전원 ON 시의 유지 설정 기능 10-20
10-3-2 전원 OFF 감지 지연 시간 설정 10-22
10-3-3 전원 OFF 금지 구간 설정 10-23
10-3-4 운전 중 출력 기능 10-24
10-3-5 전원 ON 시 자동 전송 기능 10-25

10-4 유니트 관련 관리 기능 10-32
10-4-1 기본 I/O 유니트 관련 10-32
10-4-2 CPU 고기능 유니트 관련 10-33
10-4-3 고기능 I/O 유니트 관련 10-34

10-5 메모리 관리 기능 10-35
10-5-1 자동 백업 기능 10-35
10-5-2 EM 파일 메모리 기능 10-38
10-5-3 코멘트 기억 기능 10-38
10-5-4 프로그램 교체 기능 10-40

10-6 보호 기능 10-46
10-6-1 딥 스위치에 의한 쓰기 방지 기능 10-46
10-6-2 비밀번호를 사용한 읽기 방지 기능 10-47
10-6-3 제조 로트 번호에 따른 운전 금지 기능 10-53
10-6-4 FINS 쓰기 방지 기능 10-54
10-6-5 PLC 이름 기능 10-58

10-7 디버그 기능 10-61
10-7-1 강제 세트/리셋 10-61
10-7-2 테스트 입력 10-62
10-7-3 미분 모니터 10-62
10-7-4 온라인 편집 10-63
10-7-5 부하 차단 10-65
10-7-6 데이터 트레이스 10-66
10-7-7 에러 발생 시 정지 위치 저장 기능 10-70
10-7-8 고장 진단 명령(FAL 명령, FALS 명령) 기능 10-71
10-7-9 시스템 이상을 고의로 발생시키는 기능 10-72
10-7-10 고장 지점 검출 명령(FPD 명령) 기능 10-73

10-8 유니트 간 동기 제어 기능 10-75
10-8-1 개요 10-75
10-8-2 유니트 간 동기 제어의 자세한 구조 10-78
10-8-3 유니트 간 동기 제어 기능의 사양 10-81
10-8-4 동기 데이터 링크 릴레이 10-82
10-8-5 유니트 간 동기 제어 기능의 사용상 제한 사항 10-86
10-8-6 사용 순서 10-88
10-8-7 PLC 시스템 설정 10-89
10-8-8 동기 인터럽트 태스크의 작성 10-91
10-8-9 유니트 간 동기 제어 동작의 조정/문제 해결 10-92

제11장 주변 툴/기기와 연결 11-1

11-1 CX-Programmer에서 PLC로 연결하는 방법 11-2
11-1-1 개요 11-2
11-1-2 CX-Programmer에서 액세스 가능한 시스템 구성 11-4
11-1-3 CX-Programmer에서 PLC에 대한 액세스 조작 11-8
11-1-4 자동 온라인 연결 11-11

11-2 시리얼 통신 11-15
11-2-1 시리얼 통신 11-15

11-3 통신 네트워크 11-28

제12장 CPU와 사이클 시간 12-1

12-1 사이클 시간의 모니터링 12-2
12-1-1 사이클 시간의 모니터링 방법 12-2

12-2 사이클 시간 계산 방법 12-4
12-2-1 CPU 유니트의 동작 흐름 12-4
12-2-2 사이클 시간 계산의 개요 12-5
12-2-3 각 유니트의 I/O 리프레시 시간 12-7
12-2-4 사이클 시간 계산 예 12-10
12-2-5 온라인 편집 시의 사이클 시간 연장 시간 12-11
12-2-6 입출력 응답 시간 12-12
12-2-7 시리얼 PLC 링크의 응답 성능 12-13

부록 ..... 부록 -1

부록-1 명령 목록 부록-2
부록-1-1 시퀀스 입력 명령 부록-3
부록-1-2 시퀀스 출력 명령 부록-5
부록-1-3 시퀀스 제어 명령 부록-7
부록-1-4 타이머/카운터 명령 부록-10
부록-1-5 데이터 비교 명령 부록-14
부록-1-6 데이터 전송 명령 부록-17
부록-1-7 데이터 이동 명령 부록-19
부록-1-8 증가/감소 명령 부록-22
부록-1-9 사칙연산 명령 부록-23
부록-1-10 데이터 변환 명령 부록-28
부록-1-11 논리 연산 명령 부록-34
부록-1-12 특수 연산 명령 부록-35
부록-1-13 부동 소수점 변환 및 연산 명령 부록-36
부록-1-14 배정도 부동 소수점 변환 및 연산 명령 부록-40
부록-1-15 테이블 데이터 처리 명령 부록-43
부록-1-16 추적 명령 부록-47
부록-1-17 데이터 제어 명령 부록-49
부록-1-18 서브루틴 명령 부록-52
부록-1-19 인터럽트 제어 명령 부록-53
부록-1-20 공정 진행 제어 명령 부록-54
부록-1-21 I/O 유니트용 명령 부록-55
부록-1-22 시리얼 통신 명령 부록-58
부록-1-23 네트워크 통신용 명령 부록-60
부록-1-24 파일 메모리용 명령 부록-62
부록-1-25 표시 기능용 명령 부록-63
부록-1-26 시계 기능 명령 부록-63
부록-1-27 디버그 처리 명령 부록-64
부록-1-28 고장 진단 명령 부록-64
부록-1-29 특수 명령 부록-65
부록-1-30 블록 프로그램 명령 부록-66
부록-1-31 문자열 처리 명령 부록-70
부록-1-32 태스크 제어 명령 부록-72
부록-1-33 기종 변환용 명령 부록-73
부록-1-34 펑션 블록용 특수 명령 부록-74

부록-2 명령 처리 시간/스텝 수 목록 부록-75
부록-2-1 시퀀스 입력 명령 부록-76
부록-2-2 시퀀스 출력 명령 부록-77
부록-2-3 시퀀스 제어 명령 부록-77
부록-2-4 타이머/카운터 명령 부록-78
부록-2-5 데이터 비교 명령 부록-79
부록-2-6 데이터 전송 명령 부록-80
부록-2-7 데이터 이동 명령 부록-81
부록-2-8 증가/감소 명령 부록-81
부록-2-9 사칙연산 명령 부록-82
부록-2-10 데이터 변환 명령 부록-83
부록-2-11 논리 연산 명령 부록-84
부록-2-12 특수 연산 명령 부록-84
부록-2-13 부동 소수점 변환 및 연산 명령 부록-85
부록-2-14 배정도 부동 소수점 변환 및 연산 명령 부록-86
부록-2-15 테이블 데이터 처리 명령 부록-87
부록-2-16 추적 명령 부록-88
부록-2-17 데이터 제어 명령 부록-88
부록-2-18 서브루틴 명령 부록-88
부록-2-19 인터럽트 제어 명령 부록-89
부록-2-20 공정 진행 제어 명령 부록-89
부록-2-21 I/O 유니트용 명령 부록-89
부록-2-22 시리얼 통신 명령 부록-90
부록-2-23 네트워크 통신 명령 부록-90
부록-2-24 파일 메모리용 명령 부록-90
부록-2-25 표시 기능용 명령 부록-90
부록-2-26 시계 기능 명령 부록-91
부록-2-27 디버그 처리 명령 부록-91
부록-2-28 고장 진단 명령 부록-91
부록-2-29 특수 명령 부록-91
부록-2-30 블록 프로그램 명령 부록-92
부록-2-31 문자열 처리 명령 부록-93
부록-2-32 태스크 제어 명령 부록-93
부록-2-33 기종 변환용 명령 부록-93
부록-2-34 펑션 블록용 특수 명령 부록-94
부록-2-35 SFC 제어 명령 부록-94
부록-2-36 펑션 블록 사용 시의 인스턴스 실행 시간 부록-94

부록-3 특수 보조 릴레이 목록 부록-96
부록-3-1 읽기 전용 영역 부록-96
부록-3-2 읽기/쓰기 가능 영역 부록-114
부록-3-3 특수 보조 릴레이 보충 설명 부록-121

부록-4 I/O 메모리 실제 주소 부록-124
부록-4-1 I/O 메모리 실제 주소 부록-124
부록-4-2 I/O 메모리 실제 주소 목록 부록-125

부록-5 전원 OFF 시 PLC의 동작 부록-126
부록-5-1 전원 OFF 시의 동작 부록-126
부록-5-2 전원 OFF 시의 명령 실행 부록-128

부록-6 Windows XP(SP2 이상)/Vista에서 EtherNet/IP를 연결하기 전에 부록-130
부록-6-1 Windows 방화벽 설정 변경 순서 부록-130

부록-7 CJ/CS 시리즈 비교 표 부록-132

부록-8 유니트 버전에 따른 지원 기능 목록 부록-136

색인 부록-139

사용 시 동의 사항

● 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 또는 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.

● 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.

● 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.

- **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.

- **서비스의 범위**

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

안전상의 주의 사항

안전한 사용을 위한 표시 및 의미

이 사용 설명서에서는 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 및 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)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.



- 비상 정지 회로, 인터록 회로, 리미트 회로 등 안전 보호에 관한 회로는 반드시 PLC 외부의 제어 회로에 구성하십시오.
- PLC는 자가 진단 기능을 통해 이상을 검출한 경우 또는 운전 정지 고장 진단(FALS) 명령이 실행된 경우 운전을 정지하고 모든 출력을 OFF로 만듭니다. 단, PLC의 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.
- 출력 릴레이의 용착이나 연소, 출력 트랜지스터의 파괴 등에 의해 PLC의 출력이 ON 또는 OFF가 된 상태에서 그대로 있는 경우가 있습니다.
  이런 경우 시스템이 안전하게 동작하도록 PLC 외부에 대책을 마련하십시오.
- 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 실행하십시오.
CPU 유니트의 동작 모드에 관계 없이 출력 유니트에 연결된 기기가 오작동을 일으킬 수 있습니다.



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



⚠ 주 의

사이클 시간이 연장되어도 영향이 없는지 확인한 후 온라인 편집을 실시하십시오.
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 배터리에 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.



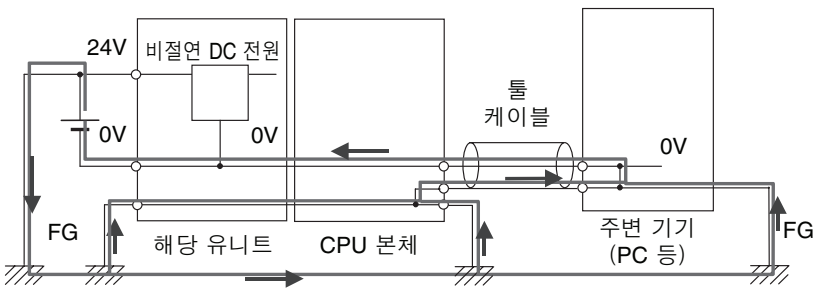
전기가 흐를 때 또는 전원을 OFF한 직후에는 전원 유니트에 접촉하지 마십시오.
화상의 위험이 있습니다.



비절연 전원 타입의 전원 유니트(CJ1W-PD022)를 장착한 PLC에 PC 등의 주변기기를 연결할 때는 외부 전원을 0V에 접지하거나 접지하지 마십시오.
주변기기의 접지 방법에 따라 외부 전원이 단락될 수도 있으므로 아래 그림과 같이 24V에는 절대 접지하지 마십시오.



《24V 전원이 단락된 사례》



안전상의 중요 사항

CJ 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

- 이 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 장착 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), 프로그램 가능 터미널(NV3W-M□20L) 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 메모리가 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 시간이 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 이 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)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 초과분은 EM 영역을 덮어 쓰게 되므로 주의하십시오.
- 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.
- 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 버튼을 누르고 BUSY LED가 꺼졌는지 확인한 후 빼십시오.
- 복원할 데이터를 선택할 때는 선택한 데이터가 정확한 복원 대상인지 충분히 확인한 후 다음 조작을 실시하십시오.
  선택한 데이터의 내용에 따라 복원 후 제어 대상이 예기치 않게 동작할 수 있습니다.
- 고기능 유니트에는 CPU 유니트에 저장된 파라미터(할당 DM 영역, 데이터 링크 테이블, Ethernet 설정 등)를 통해 작동하는 기능을 가진 것들이 있습니다. 해당 고기능 유니트를 대상으로 선택한 경우 PLC 백업 툴의 '정보'란에 제약 정보가 표시됩니다. 해당 제약 정보를 확인한 후 반드시 해당 고기능 유니트와 CPU 유니트를 모두 대상으로 선택하고 백업 또는 복원하십시오.
  한쪽 유니트의 데이터만 백업 또는 복원하여 장치를 가동하면 제어 대상이 예기치 않게 동작할 위험이 있습니다.
- 백업에 대한 제약이 있는 유니트가 포함된 백업 데이터를 복원하는 경우에는 PLC 백업 툴의 '정보'란에 제약 정보가 표시됩니다. 해당 제약 정보를 확인한 후 필요한 처리를 하십시오.
  복원 후 장치를 가동시킬 때 제어 대상이 예기치 않게 동작할 위험이 있습니다.
- PLC가 운전 중일 때 복원을 실행하는 경우 PLC의 작동이 정지되어도 문제가 없는지 확인하십시오.
  PLC가 의도하지 않은 타이밍에 정지하면 제어 대상이 예기치 않게 동작할 수 있습니다.
- 복원 후에는 반드시 PLC 전원을 다시 켜십시오.
  복원한 데이터가 반영되지 않고 제어 대상이 예기치 않게 동작할 수 있습니다.
- 강제 세트/리셋 정보는 백업할 수 있지만 복원되지는 않습니다.
  따라서 강제 세트/리셋 정보가 포함된 데이터를 복원한 후, 필요에 따라 CX-Programmer에서 강제 세트/리셋 조작을 한 후 운전하십시오.
  강제 상태가 서로 다르면 제어 대상이 예기치 않게 동작할 수 있습니다.
- 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.

● 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.

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

법규와 규격

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. 따라서 EC 지령 적합품인 CJ 시리즈를 사용하는 경우에도 고객이 기계 및 장비 전체에서 EC 지령의 적합성을 확인하여 대처해야 합니다.

EC 지령에 대한 적합성

적합 지령

- EMC 지령
- 저전압 지령

적합의 개념

● EMC 지령

오므론의 제품은 각종 기계, 제조 장치에 조립하여 사용하는 전기 기기이므로 조립한 기계 및 장치가 좀 더 간단하게 EMC 규격에 적합할 수 있도록 제품 자체와 관련된 EMC 규격(*)을 준수하기 위해 노력하고 있습니다.

그러나 고객의 기계 및 장치가 다양하며 EMC의 성능은 EC 지령 적합 제품을 조립한 기기 및 제어반의 구성, 배선 상태, 배치 상태 등에 따라 달라지므로 고객의 사용 상태에서의 적합성은 확인할 수 없습니다. 따라서 기계 및 장치 전체의 최종적인 EMC 적합성을 고객이 스스로 확인하시기 바랍니다.

* EMC (Electro-Magnetic Compatibility: 전자 환경 적합성) 관련 규격 중
EMS(Electro-Magnetic Susceptibility: 전자파 내성)에 관해서는 EN61000-6-2
(Electro-Magnetic Interference: 전자파 간섭)에 대해서는 EN61000-6-4,
EN61000-6-4 Radiated emission(복사성 방출)은 10m법에 따릅니다.

● 저전압 지령

전원 전압 50V AC~1000V AC 및 75V DC~1500V DC에서 작동하는 기기에 대해 필요한 안전성을 확보해야 합니다. 적용 규격은 EN61131-2입니다.

● EC 지령에 대한 적합성

CJ 시리즈는 EC 지령에 적합합니다. 그러나 고객의 기계 및 장치가 EC 지령에 적합하도록 하기 위해서는 다음 사항에 주의해야 합니다.

- CJ 시리즈는 반드시 제어반 내에 설치하십시오.
- DC 전원 유니트 및 I/O 유니트에 연결하는 DC 전원은 강화 절연 또는 이중 절연된 것을 사용하십시오.
- CJ 시리즈의 EC 지령 적합 제품은 EMI에 관한 공통 방출 규격(EN61000-6-4)에 적합하지만, 특히 Radiated emission(복사성 방출: 10m법)에 관해서는 사용하는 제어반의 구성, 연결되는 다른 기기와의 관계, 배선 등에 따라 달라질 수 있습니다.
  따라서 EC 지령 적합품인 CJ 시리즈를 사용하는 경우에도 고객이 기계 및 장비 전체에서 EC 지령의 적합성을 확인하여 대처해야 합니다.

선박 규격에 대한 적합성

본 제품은 각종 선박 규격에 적합합니다. 각 선박 규격에 적용하는 경우 사용 조건이 설정되어 설치 장소에 따라 사용할 수 없는 경우도 있으므로 당사 영업 담당자에게 문의하십시오.

각 선급 규격의 사용 조건(NK, LR)

● 선교(브리지) 및 갑판(데크) 이외에서의 사용 조건

- 본 제품은 반드시 제어반에 설치하십시오.
- 제어반의 개폐구 등의 간격은 개스킷 등으로 완전히 차폐하십시오.

● 선교(브리지)에서의 사용 조건(NK 일본해사협회에서만 인정 완료)

- 본 제품은 반드시 제어반에 설치하십시오.
- 제어반의 개폐구 등의 간격은 개스킷 등으로 완전히 차폐하십시오.
- 전원 라인에 다음과 같은 노이즈 필터를 연결하십시오.

노이즈 필터

제조업체	코셀사
형식	TAH-06-683

상표

SYSMAC은 오므론 주식회사에서 제조한 프로그램 가능 컨트롤러의 등록 상표입니다.
CX-One은 오므론 주식회사에서 제조한 프로그래밍 툴의 등록 상표입니다.
Windows는 마이크로소프트사의 등록 상표입니다.
본문에 게재된 기타 시스템 이름 및 제품명은 각 회사의 상표 또는 등록 상표입니다.

CJ2 CPU 유니트의 유니트 버전

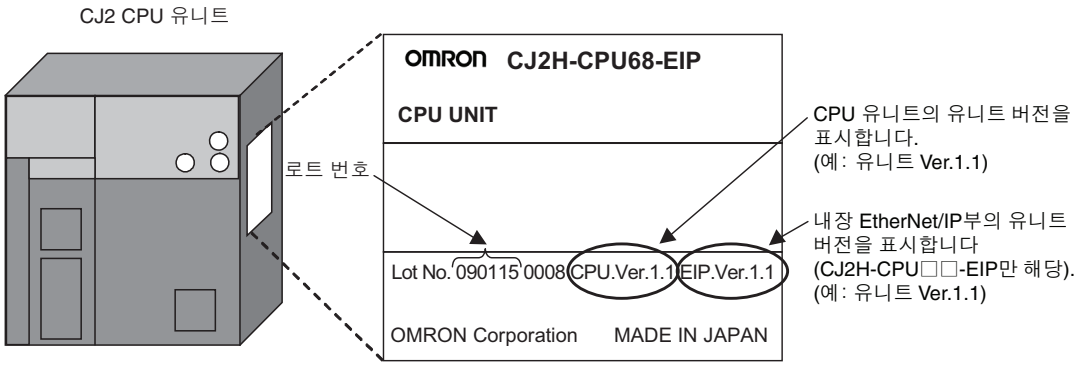
유니트 버전

CJ2 CPU 유니트는 버전 업그레이드 등에 따른 CPU 유니트의 탑재 성능 차이를 관리하기 위해 '유니트 버전'이라는 개념을 도입하였습니다.

제품에서 표기 방법

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

예: CJ2H-CPU68-EIP의 경우



지원 소프트웨어에 의한 확인 방법

CX-Programmer에서는 몇 가지 방법으로 유니트 버전을 확인할 수 있습니다.

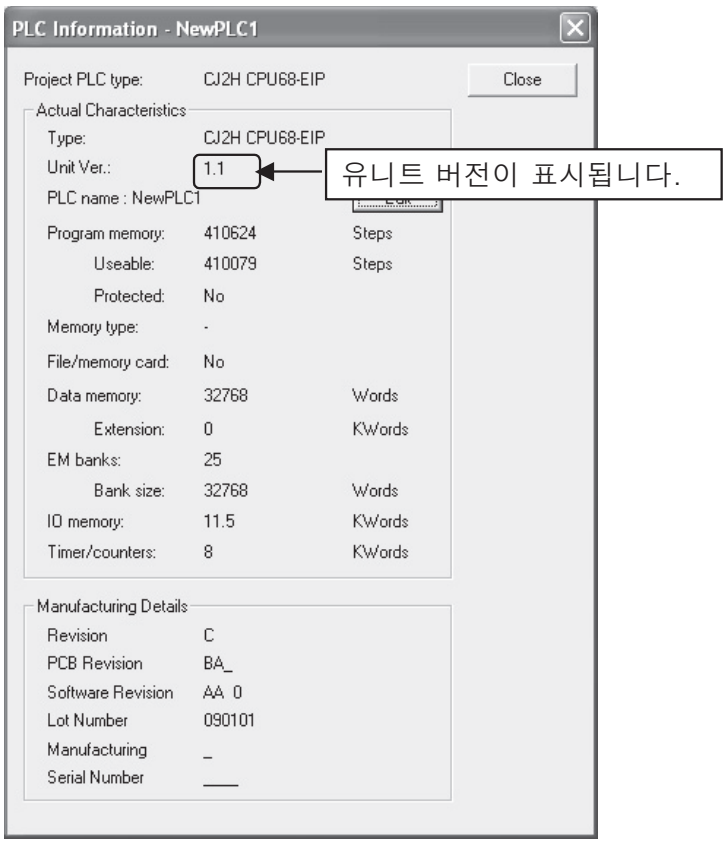
- [PLC 정보]에서 확인하는 방법
- [유니트 제조 정보]로 확인하는 방법(고기능 I/O 유니트, CPU 고기능 유니트도 확인 가능)

● [PLC 정보]에서 확인하는 방법

1 ▪ **PLC 기종 및 CPU 형식을 알고 있는 경우:**
[PLC 변경] 대화 상자에서 선택하여 온라인으로 연결하고 [PLC] 메뉴의 [편집] | [정보]를 선택하십시오.

▪ **PLC 기종 및 CPU 형식이 정확하지 않은 경우 (단, 시리얼 연결로 직접 연결만 가능):**
[PLC] 메뉴에서 [자동 온라인]을 선택하여 온라인으로 연결하고 [PLC] 메뉴의 [편집] | [정보]를 선택하십시오.

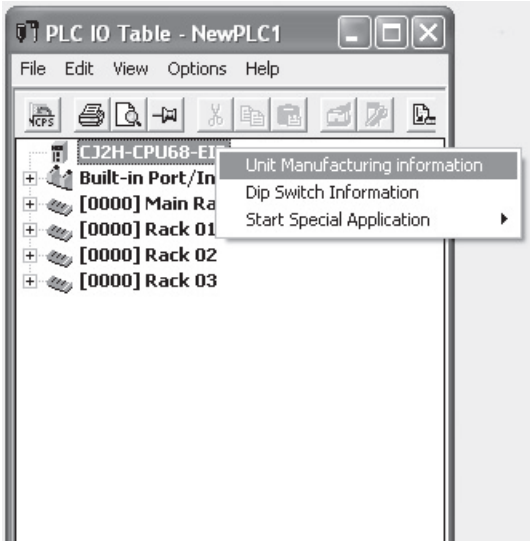
2 **다음과 같은 [PLC 정보] 대화 상자가 표시됩니다.**



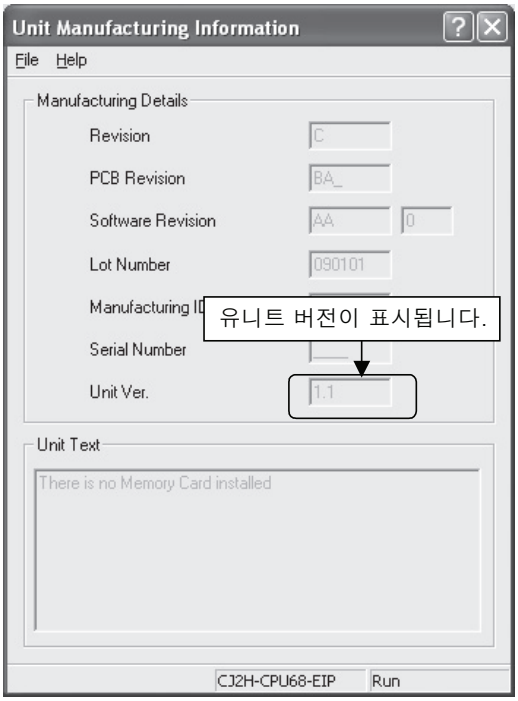
온라인으로 연결된 CPU 유니트의 유니트 버전을 선택하십시오.

- **[유니트 제조 정보]에서 확인하는 방법**

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



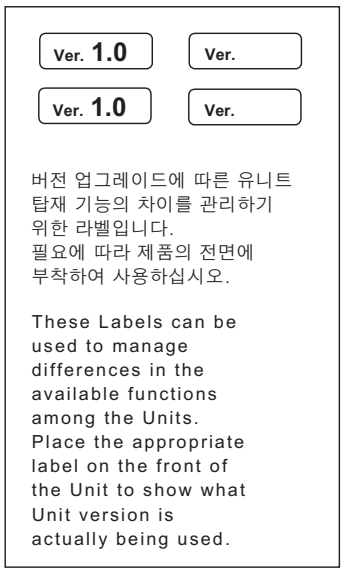
2 **다음과 같은 [유니트 제조 정보] 대화 상자가 표시됩니다.**



온라인으로 연결된 CPU 유니트의 유니트 버전을 선택하십시오.

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.

유니트 버전의 종류

기종	형식	유니트 Ver.
CJ2H CPU 유니트	CJ2H-CPU6□-EIP	유니트 Ver.1.0 (내장 EtherNet/IP부: 유니트 Ver.2.0)
		유니트 Ver.1.1 (내장 EtherNet/IP부: 유니트 Ver.2.0) 유니트 Ver.1.2 (내장 EtherNet/IP부: 유니트 Ver.2.0) 유니트 Ver.1.3 (내장 EtherNet/IP부: 유니트 Ver.2.0)
	CJ2H-CPU6□	유니트 Ver.1.1 유니트 Ver.1.2 유니트 Ver.1.3
CJ2M CPU 유니트	CJ2M-CPU3□	유니트 Ver.1.0 (내장 EtherNet/IP부: 유니트 Ver.2.0)
	CJ2M-CPU1□	유니트 Ver.1.0

유니트 버전과 주변 툴의 관계

유니트 버전과 CX-Programmer의 버전에는 다음과 같은 관계가 있습니다.
각 유니트 버전에서 강화된 기능에 대해서는 '부록-8 유니트 버전에 따른 지원 기능 목록'을 참조하십시오.

● 유니트 버전과 주변 툴의 관계

CPU 유니트		사용하는 기능		필요한 주변 툴					
				CX-Programmer					프로그래밍 콘솔
				Ver.7.1 이하	Ver.8.0	Ver.8.2 이상	Ver.9.0	Ver.9.1 이상	
CH2H	CJ2H-CPU6□-EIP 유니트 Ver.1.0	유니트 Ver.1.0의 기능		×	○	○	○	○	× (*3)
	CJ2H-CPU6□-EIP 유니트 Ver.1.1	유니트 Ver.1.1에서 강화된 기능	사용	×	×	○ (*2)	○	○	
			사용 안 함	×	○ (*1)	○	○	○	
	CJ2H-CPU6□ 유니트 Ver.1.1	유니트 Ver.1.1에서 강화된 기능	사용	×	×	○ (*2)	○	○	
			사용 안 함	×	×	○	○	○	
	CJ2H-CPU6□-EIP 유니트 Ver.1.2	유니트 Ver.1.2에서 강화된 기능	사용	×	×	×	○	○	
			사용 안 함	×	○ (*1)	○ (*1)	○	○	
	CJ2H-CPU6□ 유니트 Ver.1.2	유니트 Ver.1.2에서 강화된 기능	사용	×	×	×	○	○	
			사용 안 함	×	○ (*1)	○ (*1)	○	○	
	CJ2H-CPU6□-EIP 유니트 Ver.1.3	유니트 Ver.1.3에서 강화된 기능	사용	×	×	×	×	○	
			사용 안 함	×	○ (*1)	○ (*1)	○	○	
	CJ2H-CPU6□ 유니트 Ver.1.3	유니트 Ver.1.3에서 강화된 기능	사용	×	×	×	×	○	
			사용 안 함	×	○ (*1)	○ (*1)	○	○	
CJ2M	CJ2M-CPU□□ 유니트 Ver.1.0	유니트 Ver.1.0의 기능		×	×	×	×	○	

*1 버전 업그레이드로 강화된 기능을 사용하지 않는 경우 CX-Programmer의 버전을 높일 필요가 없습니다.
*2 유니트 Ver.1.1에서 강화된 기능을 사용하는 경우 CX-Programmer Ver.8.2 이상을 사용해야 합니다. 단, 고속 인터럽트 기능과 고정 사이클 시간 운전 중 변경 기능에 대해서만 CX-Programmer Ver.8.02에서 지원합니다.
*3 CJ2 CPU 유니트에서는 프로그래밍 콘솔을 사용할 수 없습니다.

● PLC 기종의 풀다운 목록

CX-Programmer의 [PLC 변경] 대화 상자에 있는 [PLC 기종] 풀다운 목록에서는 유니트 버전을 식별할 수 없습니다.

유니트 버전에 관계 없이 다음 중에서 선택하십시오.

시리즈	CPU 유니트 타입	CPU 유니트 형식	CX-Programmer Ver.9.0 이상의 [PLC 변경] 대화 상자에서 PLC 기종 목록 표기
CJ 시리즈	CJ2H CPU 유니트	CJ2H-CPU6□-EIP CJ2H-CPU6□	CJ2H
	CJ2M CPU 유니트	CJ2M-CPU3□ CJ2M-CPU1□	CJ2M

유니트 버전과 관련된 CX-Programmer 조작 문제 해결

현상	원인	대처법
 위의 에러 메시지를 표시한 후 출력 창의 [컴파일] 탭에 컴파일 에러 내용이 표시됩니다.	버전 업그레이드한 유니트에서만 지원하는 명령이 포함된 사용자 프로그램을 이전 버전의 CPU 유니트에 다운로드 하려고 했습니다.	사용자 프로그램을 재검토하십시오. 또는 다운로드하려는 CPU 유니트를 버전 업그레이드한 CPU 유니트로 교환하십시오.
	버전 업그레이드한 유니트에서만 지원하는 PLC 시스템 설정 데이터(기본값 이외)를 이전 버전의 CPU 유니트에 다운로드 하려고 했습니다.	PLC 시스템 설정 데이터를 재검토하십시오. 또는 다운로드하려는 CPU 유니트를 버전 업그레이드한 CPU 유니트로 교환하십시오.
프로그램을 PLC에서 PC로 전송하면 프로그램 내에 "????"가 표시됩니다.	버전 업그레이드한 CX-Programmer에서만 지원하는 명령이 포함된 사용자 프로그램을 이전 버전의 CX-Programmer에서 업로드하려고 했습니다.	이전 버전의 CX-Programmer에서는 업로드할 수 없습니다. 버전 업그레이드한 CX-Programmer를 사용하여 업로드하십시오.

관련 설명서

CJ 시리즈 CJ2 CPU 유니트의 PLC 본체 관련 설명서는 다음 표와 같이 구성되어 있으므로 참조하시기 바랍니다.

설명서 명칭	설명서 번호	형식	용도	내용
CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편 (본 설명서)	SBCA-350	CJ2H-CPU6□-EIP CJ2H-CPU6□ CJ2M-CPU□□	CJ2 CPU 유니트의 소프트웨어 사양에 대해 알고 싶은 경우	CJ2 CPU 유니트에 대해 다음 내용을 설명합니다. • CPU 유니트의 동작 • 내부 메모리 • 프로그램 • 여러 가지 설정 • CPU 내장 기능 사용 설명서 하드웨어편(SBCA-349)과 함께 사용하십시오.
CJ 시리즈 CJ2 CPU 유니트 사용 설명서 하드웨어편	SBCA-349	CJ2H-CPU6□-EIP CJ2H-CPU6□ CJ2M-CPU□□	CJ2 CPU 유니트의 하드웨어 사양에 대해 알고 싶은 경우	CJ2 CPU 유니트에 대해 다음 내용을 설명합니다 • 개요/특징 • 기본 시스템의 구성 • 각 부분의 명칭과 기능 • 부착 및 설정 방법 • 문제 발생 시 대처 방법 사용 설명서 소프트웨어편(SBCA-350)과 함께 사용하십시오.
EtherNet/IP 유니트 사용 설명서	SBCD-342	CJ2H-CPU6□-EIP CJ2M-CPU3□ CS1W-EIP21 CJ1W-EIP21	CJ2 CPU 유니트의 EtherNet/IP 내장 포트를 사용하는 경우	EtherNet/IP 내장 포트/유니트에 대해 설명합니다. 기본 설정, 태그 데이터 링크, FINS 통신, 기타 기능에 대해 설명합니다.
CS/CJ/NSJ 시리즈 명령 참조 설명서	SBCA-351	CJ2H-CPU6□-EIP CJ2H-CPU6□ CJ2M-CPU□□ CS1G/H-CPU□□H CS1G/H-CPU□□-V1 CJ1G/H-CPU□□H CJ1G-CPU□□ CJ1M-CPU□□ NSJ□-□□□□(B)-G5D NSJ□-□□□□(B)-M3D	명령어에 대해 자세히 알고 싶은 경우	각 명령어에 대한 자세한 내용을 설명합니다. 프로그래밍할 때 사용 설명서 소프트웨어편(SBCA-350)과 함께 사용하십시오.
CS/CJ/CP/NSJ 시리즈 통신 명령 참조 설명서	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 시리즈 CPU 유니트, NSJ 시리즈의 통신 명령에 대해 자세히 알고 싶은 경우	1) C 모드 명령 및 2) FINS 명령에 대한 자세한 내용을 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 관계 없습니다(CPU 유니트의 시리얼 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.
CX-One 설치 설명서	SBCA-346	CXONE- AL□□C-V□/AL□□D-V□	CX-One에서 소프트웨어를 설치하는 경우	FA 통합 툴 패키지 CX-One의 개요, CX-One 설치 방법에 대해 설명합니다.

설명서 명칭	설명서 번호	형식	용도	내용
CX-Programmer 조작 설명서	SBCA-337	WS02-CX□□-V□	Windows PC용 프로그래밍 툴 CX-Programmer의 조작 방법에 대해 알고 싶은 경우	CX-Programmer의 조작 방법을 설명합니다. 프로그래밍할 때 CJ2 사용 설명서 소프트웨어편(SBCA-350), 명령 참조 설명서 (SBCA-351)와 함께 사용하십시오.
CX-Programmer 조작 설명서 펑션 블록 편/구조적 텍스트편	SBCA-338			
CX-Programmer 조작 설명서 SFC편	SBCA-348			
CX-Simulator 조작 설명서	SBCA-310	WS02-SIMC1-J	Windows PC용 시뮬레이션 툴 CX-Simulator의 조작 방법에 대해 알고 싶은 경우 CX-Programmer Ver.6.1 이상에서 CX-Programmer의 시뮬레이션 기능을 사용하는 경우	CX-Simulator의 조작 방법을 설명합니다. 시뮬레이션할 때 CX-Programmer 조작 설명서(SBCA-337), CJ2 사용 설명서 소프트웨어편(SBCA-350), 명령 참조 설명서 (SBCA-351)와 함께 사용하십시오.
CX-Integrator CS/CJ/CP/NSJ 시리즈용 네트워크 설정 툴 조작 설명서	SBCA-347	CXONE-AL□□C-V□/AL□□D-V□	네트워크를 설정/모니터링하고 싶은 경우	CX-Integrator의 조작 방법을 설명합니다.

1

개요

이 장에서는 CJ2 CPU 유니트의 개요, 특징, 사양 등에 대해 설명합니다.

1-1 CJ2 CPU 유니트의 개요 1-2
1-1-1 개요 1-2
1-1-2 CJ2 CPU 유니트의 특징 1-4
1-2 기본적인 사용 순서 1-10

1-1 CJ2 CPU 유니트의 개요

1-1-1 개요

SYSMAC CJ2는 다음과 같은 특징을 가진 다기능 CPU 유니트입니다.

- **고속, 대용량 메모리**
  기본 성능을 고속화하고 메모리를 대용량화한 결과 머신 제어를 여유 있게 처리합니다.

- **EtherNet/IP 포트 탑재(CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당)**
  오픈 네트워크인 EtherNet/IP를 탑재하여 주변 툴 연결에서 PLC 간 데이터 링크, PLC 간 메시지 통신, FTP 전송 등과 같은 범용 이더넷 통신이 가능합니다.

- **툴 I/F의 범용 네트워크화**
  범용 네트워크인 USB, EtherNet/IP 포트는 시판되는 케이블을 사용할 수 있으므로 주변 툴/기기와 간단하게 연결할 수 있습니다(EtherNet/IP 포트는 CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당).

- **태그 액세스(CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당)**
  CPU 유니트 본체에 태그 이름 서버가 내장되어 태그명과 I/O 주소를 대응시킵니다. 그 결과, 외부 기기에서 태그명을 이용한 액세스가 가능해지므로 I/O 주소를 알아야 할 필요가 없습니다.

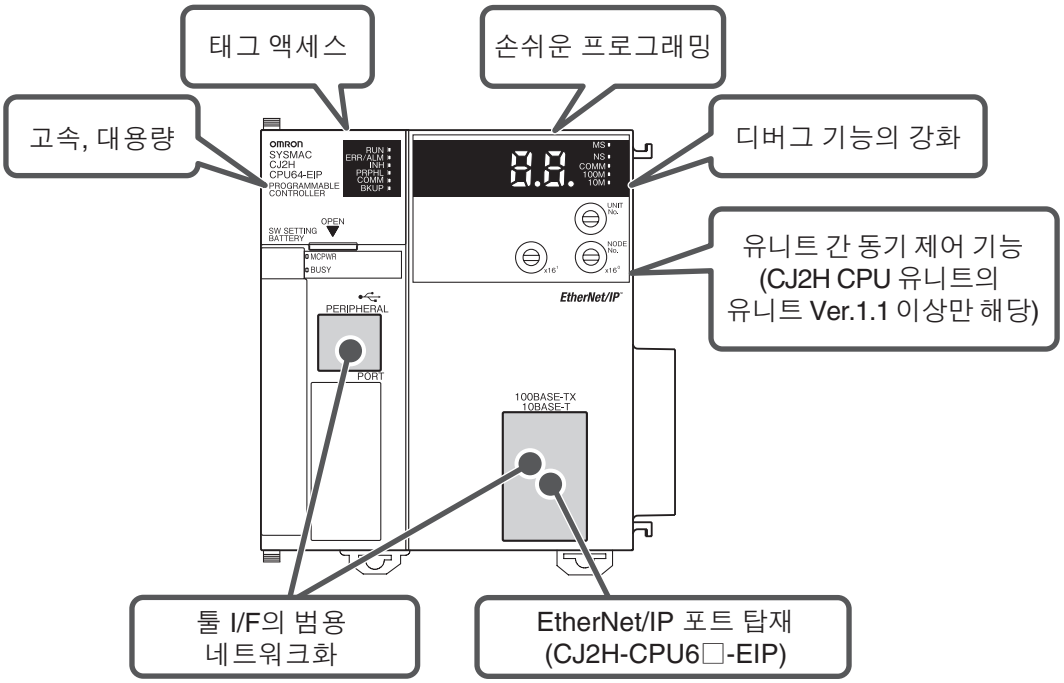
- **유니트 간 동기 제어 기능 추가(CJ2H CPU 유니트의 유니트 Ver.1.1 이상)**
  고기능 유니트의 처리 타이밍을 동기화하여 입력→연산→출력까지의 편차를 최소화한 결과, 응용 프로그램의 성능을 유지하기 쉬워졌습니다.

- **손쉬운 프로그래밍**
  DM/EM 영역의 비트 지정, 주소 오프셋 지정, 배열 변수 등 가독성이 높은 프로그래밍 환경을 제공합니다.

- **디버그 기능의 강화**
  온라인 편집, 데이터 트레이스 기능을 강화하여 디버그 효율을 크게 높였습니다.

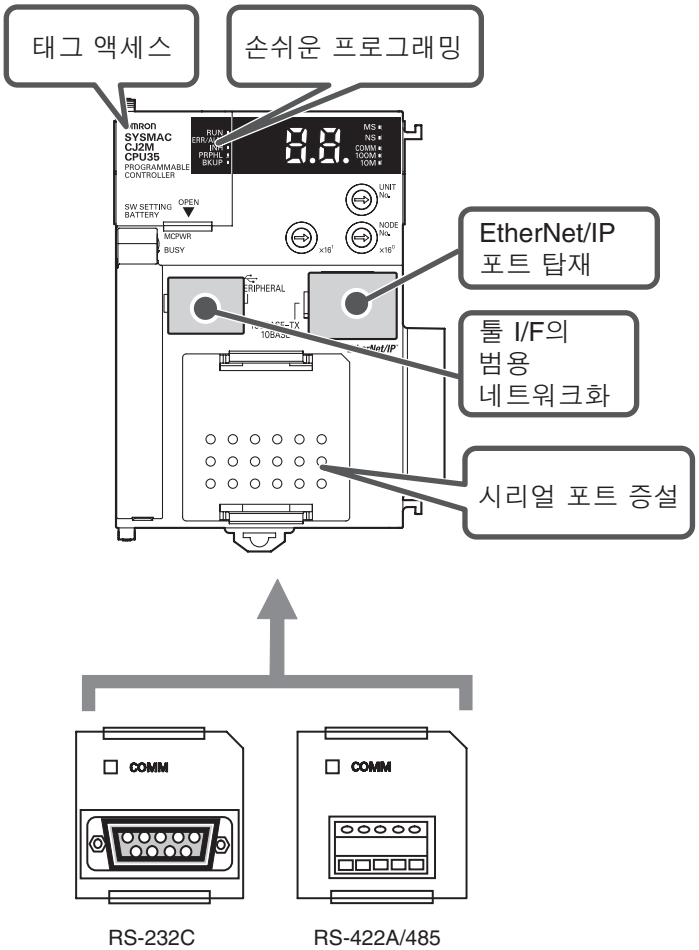
- **시리얼 포트의 증설(CJ2M-CPU3□만 해당)**
  CJ2M CPU 유니트의 표준 타입(CJ2M-CPU3□)인 경우, 표준 탑재된 EtherNet/IP 포트에 추가로 별도 판매하는 RS-232C 옵션 보드 또는 RS-422A/485 옵션 보드를 선택하여 시리얼 포트를 증설할 수 있습니다.

CJ2H CPU 유니트

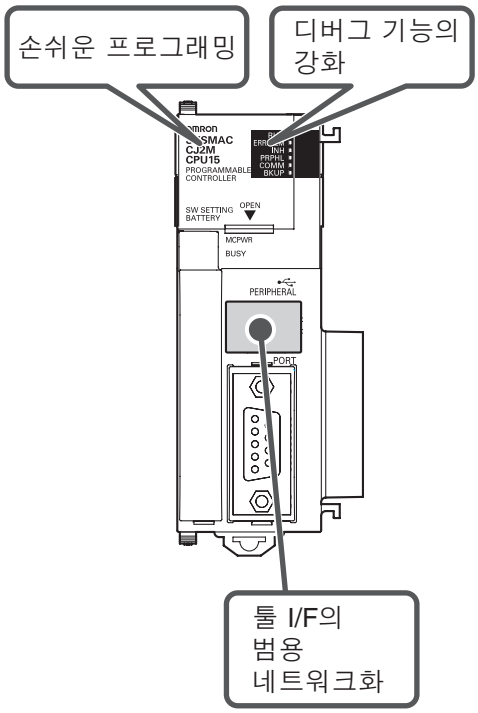


CJ2M CPU 유니트

표준 타입
(CJ2M-CPU3□)



간편 타입
(CJ2M-CPU1□)



1-1-2 CJ2 CPU 유니트의 특징

고속, 대용량 메모리

- **고속 처리**
  기본 명령 CJ2H: 0.016 μs~, CJ2M: 0.04 μs~, 응용 명령 CJ2H: 0.05 μs~, CJ2M: 0.06 μs~, 매회 리프레시 명령 CJ2H: 0.99 μs~, CJ2M: 1.26 μs~의 고속 처리가 가능합니다.

- **대용량 프로그램**
  프로그램 용량 최대 400K 스텝의 대용량 타입을 준비했습니다.

- **펑션 블록 정의 전용 영역(CJ2M만 해당)**
  CJ2M CPU 유니트의 경우, 펑션 블록 정의 전용 저장 영역으로 'FB 프로그램 영역'이 있습니다(20K 스텝).
  따라서 펑션 블록을 사용하여 기존 프로그램을 부품화, 구조화한 경우에도 사용자 프로그램 영역의 사용량이 줄어듭니다.
  또한 FB 프로그램 영역의 용량을 초과하는 펑션 블록의 정의는 사용자 프로그램 영역에 저장됩니다.

- **대용량 데이터 메모리**
  EM(확장 데이터 메모리) 최대 800K 워드(25뱅크)의 대용량 타입이 준비되어 있습니다.

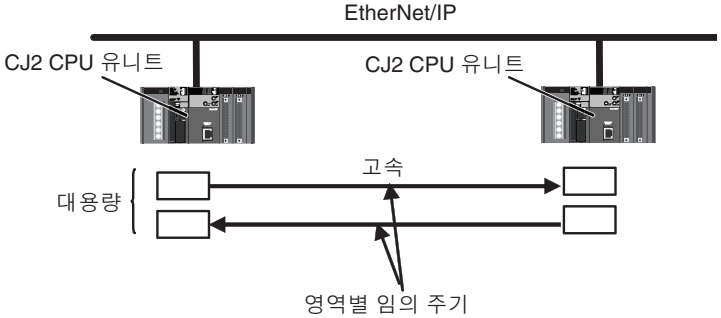
- **사이클릭 태스크 수 최대 128**
  사용자 프로그램을 최대 128개의 태스크로 분할할 수 있습니다. 그 결과, 더 작은 단위로 프로그램을 구조화하기 쉬워지고, 사이클 시간 단축에도 도움이 됩니다.

- **인터럽트 태스크의 실행 성능 향상(CJ2H CPU 유니트 Ver.1.1 이상)**
  CJ2H CPU 유니트 Ver.1.1은 평상시에 사용하는 경우에도 인터럽트 태스크의 오버헤드 시간을 유니트 Ver.1.0에 비해 약 20% 단축합니다. 또한 고속 인터럽트 기능을 사용하면 일정한 제약 조건에서 다음과 같이 인터럽트 태스크의 실행 성능을 최대한 향상시킬 수 있습니다.
  - 인터럽트 태스크의 오버헤드 시간(인터럽트 태스크 기동 시간 + 사이클릭 태스크로 복귀하는 시간) 대폭 단축
    (예: I/O 인터럽트 태스크의 경우, 평상시 사용할 때 37 μs→고속 인터럽트 기능을 사용할 때 25 μs)
  - 예약 인터럽트 태스크의 최소 시간 간격으로 0.1ms 설정 가능
    (유니트 Ver.1.0에서 최소 시간 간격은 0.2ms까지)

EtherNet/IP 포트 장착(CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당)

내장 EtherNet/IP 포트는 Ethernet 유니트와 동등한 FINS Ethernet 기능(FTP 서버, 시계 조정 등을 포함)과 오픈 네트워크 EtherNet/IP에 적합합니다.

- **고속, 대용량에서 갱신 주기를 임의로 설정할 수 있는 데이터 링크 제공**
  고속, 대용량에서 데이터 링크 영역별로 갱신 주기를 임의로 설정할 수 있습니다.



또한 EtherNet/IP 데이터 링크 툴을 사용하면 Controller Link 데이터 링크 자동 설정과 비슷한 인터페이스에서 데이터 링크를 설정할 수 있습니다.

툴 I/F의 범용 네트워크화

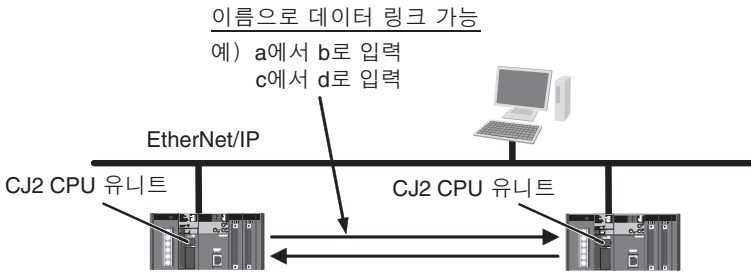
- **USB, EtherNet/IP 포트 내장**
  시판되는 USB 케이블을 사용하여 PC의 USB 포트에서 직접 연결할 수 있습니다.
  또한 CJ2H-CPU6□-EIP, CJ2M-CPU3□의 경우, USB를 경유하여 내장 EtherNet/IP 네트워크상의 PLC에 액세스할 수도 있습니다.

- **PLC 이름 기능을 사용하여 주변 툴을 통한 잘못된 접속 방지**
  CJ2 CPU 유니트는 PLC 이름을 유니트 내에 저장할 수 있습니다.
  주변 툴에서 온라인으로 연결하는 경우 이름을 조회하여 주변 툴에서 잘못 연결되지 않도록 할 수 있습니다.

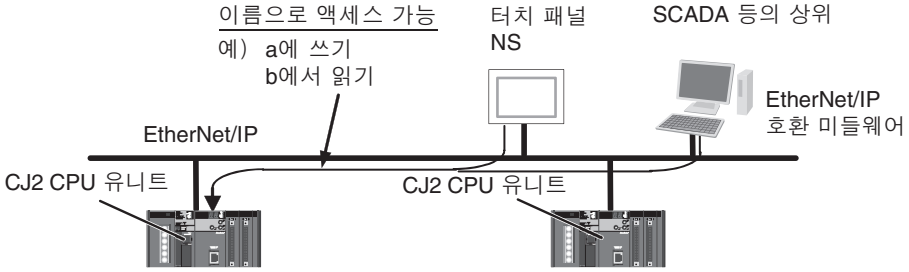
태그 액세스(CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당)

- **네트워크 대응 변수('태그')로 프로그램 변경 시 유연하게 대처 가능**
  CJ2 CPU 유니트는 네트워크 대응 변수('태그')를 지원합니다. CPU 유니트에 태그 이름 서버 기능이 내장되어 태그명과 주소의 대응표를 사전에 CPU 유니트의 변수 테이블에 저장해 둘 수 있습니다. 그 결과, 다음과 같은 작업을 수행할 수 있습니다.

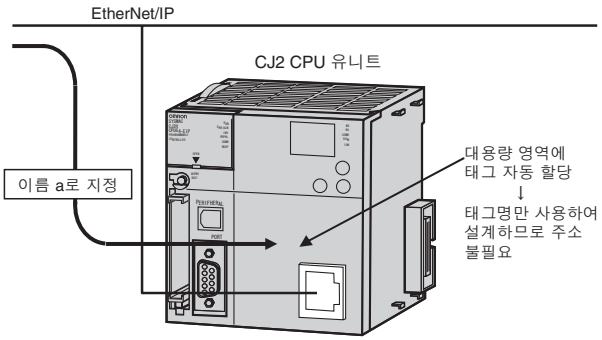
  - 태그명으로 데이터 링크를 설정할 수 있습니다.
    EtherNet/IP 상의 데이터 링크에서 주소가 아닌 변수명으로 송수신 영역을 지정할 수 있습니다. 따라서 태그명으로 설정한 데이터 링크 영역의 설정은 그대로 두고 대응표로 사용되는 변수 테이블만 변경해도 유연한 설계 변경이 가능합니다.



  - 태그명으로 상위 기기에서 액세스할 수 있습니다.
    터치 패널 NS 등의 상위 기기에서 태그명을 사용하여 액세스할 수 있습니다. 그 결과, NS 등의 상위 화면/프로그램 작성 시 재사용 능력이 향상됩니다.



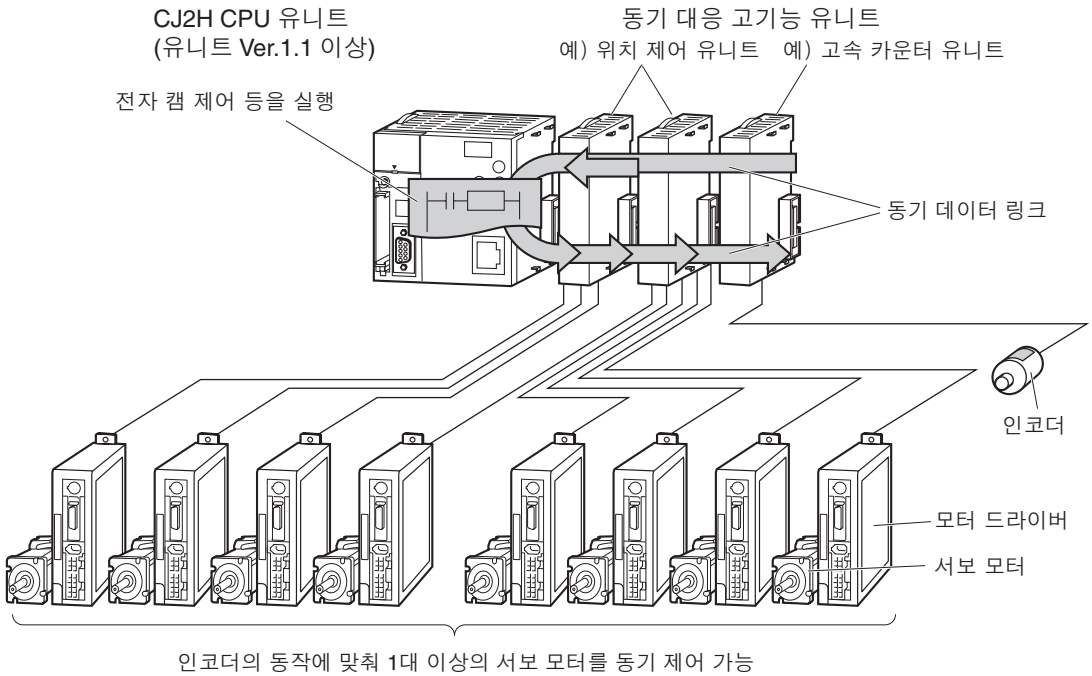
• 태그의 자동 할당 기능으로 주소 불필요
CX-Programmer의 변수 테이블 주소 자동 할당 기능으로 대용량 영역인 EM 영역에 태그를 자동 할당하면, 주소에 상관 없이 데이터 링크 설계 및 상위 기기에서 액세스할 수 있습니다.



유니트 간 동기 제어 기능(CJ2H CPU 유니트 Ver.1.1 이상)

● **고기능 유니트가 처리를 시작하는 타이밍을 CJ2 CPU 유니트에서 일정 간격으로 동기화하는 시스템을 탑재했습니다.**
CPU 유니트와 고기능 유니트(*) 간에 같은 데이터를 공유하기 위한 동기 데이터 링크 또는 해당 데이터의 연산을 처리하기 위한 동기 인터럽트 태스크를 사용할 수 있습니다.
그 결과, 입력→연산→출력까지의 편차를 최소화한 다축 제어를 실현하여, 전자 캠 제어 등과 같이 엄격한 시간 관리가 필요한 모션 응용 프로그램에도 적합합니다.

* 유니트 간 동기 제어 기능에 적합한 고기능 유니트(동기 유니트)가 대상입니다. 2009년 12월 현재, 위치 제어 유니트(CJ1W-NC214/234/414/434)만 해당.



손쉬운 프로그램 작성

- **주소의 오프셋 지정 가능**
  명령의 피연산자로 주소를 지정할 때 지정한 주소를 선두로 하여 그 지점부터의 오프셋을 [ ] 안에 지정하는 방법을 사용할 수 있습니다.
  예를 들어, [ ](오프셋)에 채널 주소를 지정하면 해당 채널 내의 값에 따라 주소를 동적으로 지정할 수 있습니다.
  예: W0.00[D0]: W0.00를 선두로 하여 D0의 값만큼 비트 주소를 이동합니다.
  D0의 값이 &3이면 W0.03이 됩니다.

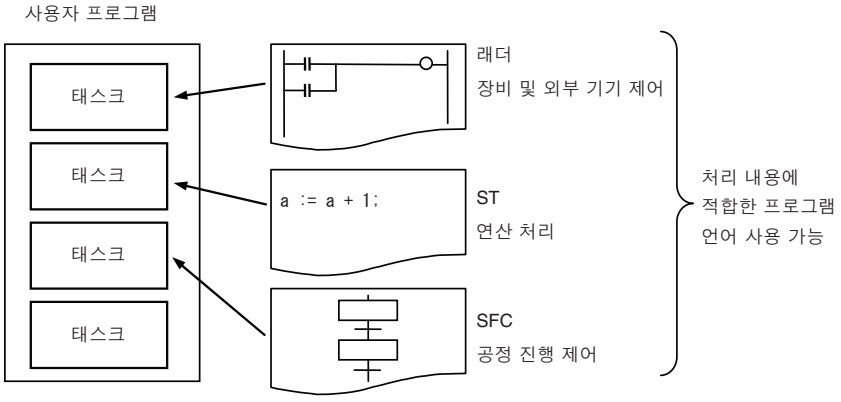
- **배열 변수의 첨자에 변수 지정 가능**
  배열 변수의 첨자에 변수를 지정하면 해당 변수 안의 값에 따라 요소를 동적으로 지정할 수 있습니다.
  예: a[b]: 변수 b의 값이 배열 변수 a[ ]의 요소 번호가 됩니다.

- **구조체 작성/사용 가능**
  CX-Programmer Ver.9.0 이상을 사용하는 경우, 구조체 데이터 형식을 작성하여 사용할 수 있습니다. 그 결과 CPU 유니트의 I/O 메모리를 데이터베이스나 라이브러리 데이터로 취급하기 쉬워집니다.

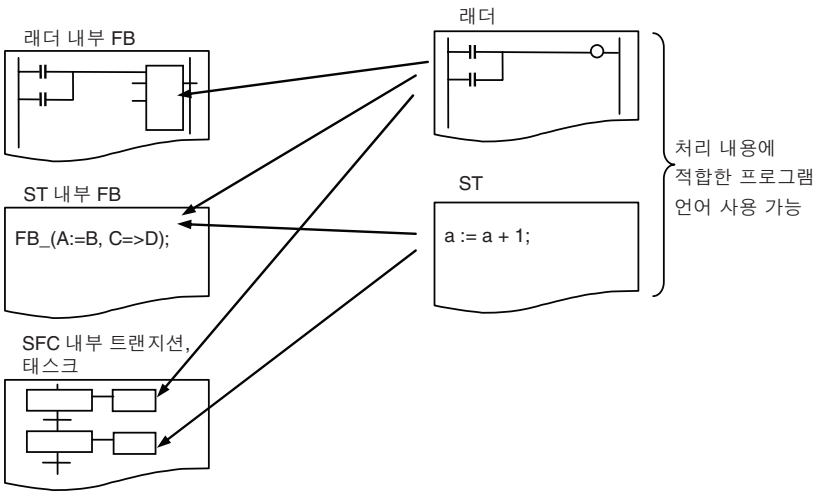
- **DM/EM 비트 지정 가능**
  이전에는 워드 단위로만 액세스할 수 있었던 데이터 메모리 영역(DM, EM 영역)을 비트 단위로도 액세스 가능해졌습니다. 그 결과, 비트 영역의 작업 영역을 확장할 수 있습니다. 또한 워드 단위 데이터를 비트 단위로 각종 처리를 수행할 수도 있습니다.
  예: D10.00은 D10의 비트 00을 지정.

- **타이머/카운터의 현재값 갱신 방식을 명령 단위에서 선택 가능**
  타이머/카운터의 현재값 갱신 방식(BCD 방식/BIN 방식)을 명령 단위에서 선택할 수 있습니다.
  예: TIM 명령(BCD 방식)과 TIMX 명령(BIN 방식)을 조합하여 사용할 수 있습니다.

- **최적의 언어를 조합하여 구조화 프로그래밍 가능**
  래더 언어, ST 언어, SFC 언어를 임의로 조합하여 사용자 프로그램을 기술할 수 있습니다. 처리 내용에 따라 최적의 언어를 사용하면 프로그래밍 작업 시간과 비용을 줄일 수 있습니다.



그리고 래더/ST 프로그램 내의 펑션 블록, SFC 프로그램 내의 트랜지션/작업의 경우 래더 언어, ST 언어를 사용하여 임의로 작성할 수 있습니다. 그 결과, 최적화된 구조화 프로그래밍이 가능합니다.



디버그 기능의 강화

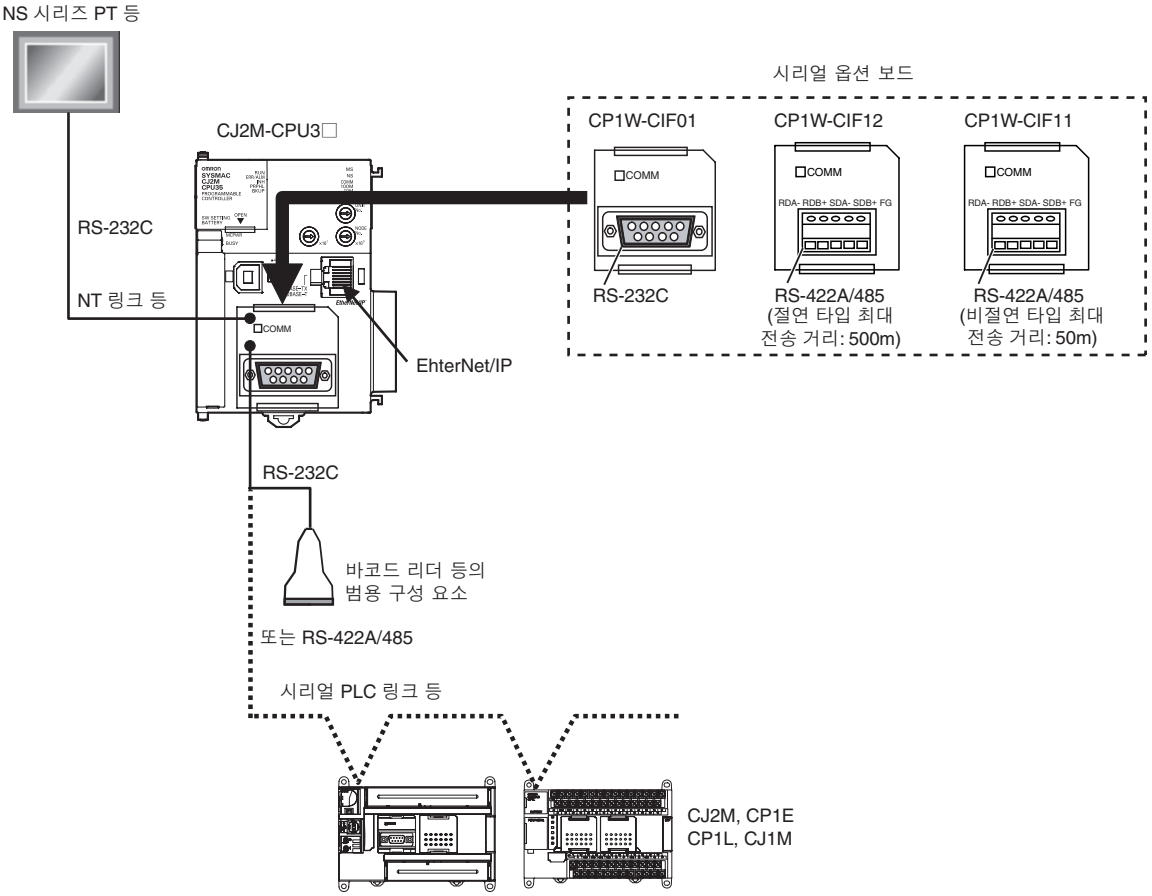
- **온라인 편집이 사이클 시간에 미치는 영향**
  온라인 편집 실행 시 전체 사이클 시간을 약 1ms로 억제합니다.
  디버그 시 장치의 작동에 미치는 영향을 억제합니다.

- **고속, 대용량에 적합한 데이터 트레이스**
  최대 32k워드(기존 대비 약 8배)로 다양한 트리거 조건이 가능하며 장시간 연속 데이터 트레이스도 가능합니다.

- **EM 영역 임의 뱅크에 대한 강제 세트/리셋 가능**
  CX-Programmer Ver.8.3 이상을 사용한 경우, 설정을 통해 EM 영역의 임의 뱅크에 대한 강제 세트/리셋을 수행할 수 있습니다. (CJ2H CPU 유니트의 유니트 Ver.1.2 이상 및 CJ2M CPU 유니트만 해당)

시리얼 옵션 보드 증설 가능(CJ2M-CPU3□만 해당)

CJ2M-CPU3□의 경우, 시리얼 옵션 보드(RS-232C×1 포트 또는 RS-422A/485×1 포트)를 최대 1개 장착할 수 있습니다.
시리얼 포트를 통해 바코드 리더 등의 범용 구성 요소, PT, CJ/CP 시리즈 PLC, 인버터와 간단하게 연결할 수 있습니다.



1-2 기본적인 사용 순서

일반적으로 다음과 같은 순서로 사용합니다.

1. 장착 및 하드웨어 설정

전원 유니트, CPU 유니트, 각 유니트, 엔드 커버 등을 장착합니다.
필요에 따라 딥 스위치 및 로터리 스위치를 설정합니다.

"CJ 시리즈 CJ2 CPU 유니트 사용 설명서 하드웨어편(SBCA-349), 제3장 각 부분의 명칭과 기능, 제5장 부착 및 설정"

2. 배선

전원 배선, I/O 배선, 통신 배선을 실시합니다.

"CJ 시리즈 CJ2 CPU 유니트 사용 설명서 하드웨어편(SBCA-349), 제5장 부착 및 설정"

"CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편(SBCA-350), 제11장 주변 툴/기기와 연결"

3. PLC와 온라인 연결

PC와 PLC를 연결합니다.

"CJ 시리즈 CJ2 CPU 유니트 사용 설명서 하드웨어편(SBCA-349), 제4장 주변 툴"

4. I/O 할당

CX-Programmer에서 장착된 유니트를 CPU 유니트의 I/O 메모리에 할당합니다.

"CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편(SBCA-350), 제8장 I/O 할당과 유니트 설정"

5. 소프트웨어 설정

PLC의 소프트웨어를 설정합니다.

- CPU 유니트의 초기 설정 "CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편(SBCA-350), 제4장 CPU 유니트의 초기 설정"
- PLC 시스템 설정 "CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편(SBCA-350), 제9장 PLC 시스템 설정"
- 고기능 유니트 할당 DM 영역 설정 "CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편(SBCA-350), 제8장 I/O 할당과 유니트 설정"

6. 프로그램 작성

CX-Programmer에서 프로그램을 작성합니다.

"CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편(SBCA-350), 제5장 프로그래밍 지식"

7. 동작 확인

입출력 배선 확인, 특수 보조 릴레이를 설정한 후 시운전을 합니다.
CX-Programmer의 모니터 및 디버그 기능을 사용할 수 있습니다.

"CJ 시리즈 CJ2 CPU 유니트 사용 설명서 소프트웨어편(SBCA-350), 10-7 디버그 기능"

8. 프로그램의 본 운전

동작 모드를 '운전' 모드로 설정하고 본 운전을 시작합니다.

2

CPU 유니트 내부 메모리

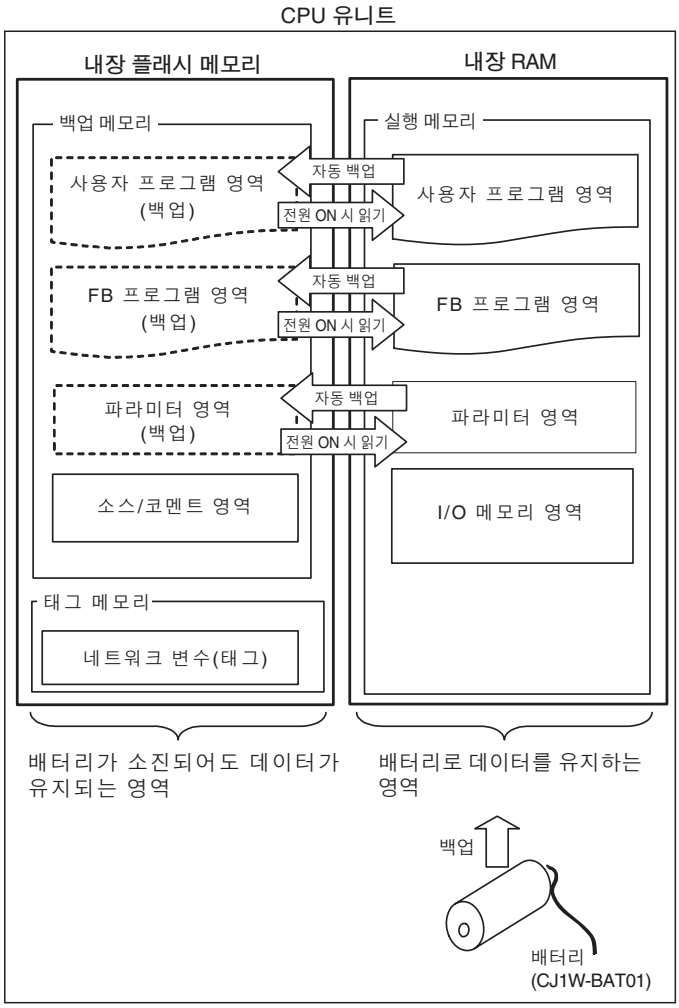
이 장에서는 CPU 내부 메모리의 종류와 저장된 데이터에 대해 설명합니다.

2-1 CPU 유니트의 내부 메모리 2-2
2-1-1 CPU 유니트의 메모리 구성 2-2
2-1-2 메모리 영역과 저장 데이터 2-3
2-1-3 주변 툴에서 데이터 전송 2-4

2-1 CPU 유니트의 내부 메모리

2-1-1 CPU 유니트의 메모리 구성

CPU 유니트의 내부 메모리는 다음과 같이 백업 메모리인 내장 플래시 메모리와 실행용 메모리인 내장 RAM으로 구성되었습니다.



내장 플래시 메모리

사용자 프로그램과 파라미터의 백업용 메모리 및 프로그램 소스와 코멘트, 네트워크 변수의 저장용 메모리로 사용됩니다.
배터리가 소진되어도 데이터가 유지됩니다. 또한 I/O 메모리는 백업되지 않습니다.

내장 RAM

CPU 유니트 실행용 메모리입니다.
사용자 프로그램, 파라미터 영역, I/O 메모리가 저장되어 있습니다.
배터리로 백업됩니다.
배터리 소진 시(배터리 전압 저하 시 또는 배터리 미장착 시), I/O 메모리 데이터가 지워집니다. 사용자 프로그램, 파라미터 영역은 내장 플래시 메모리에 수시로 백업되므로 지워지지 않고 유지됩니다.

사용상의 주의 사항

배터리 소진 시에는 다음과 같은 상황이 발생합니다.

- I/O 메모리 영역은 데이터가 지워지거나 데이터 값이 부정확해집니다.
  여기에는 DM, EM, HR 등과 같이 전원 OFF 복구 시 배터리로 유지되는 영역도 포함됩니다.
- 시계가 정지합니다. 시계와 관련된 모든 정보가 부정확해집니다.
- 이상 이력 정보가 남지 않습니다.
- 부하 차단 플래그가 부정확해집니다.

2-1-2 메모리 영역과 저장 데이터

CPU 유니트의 내부 메모리에서 각 메모리 영역/저장 데이터는 다음과 같습니다.

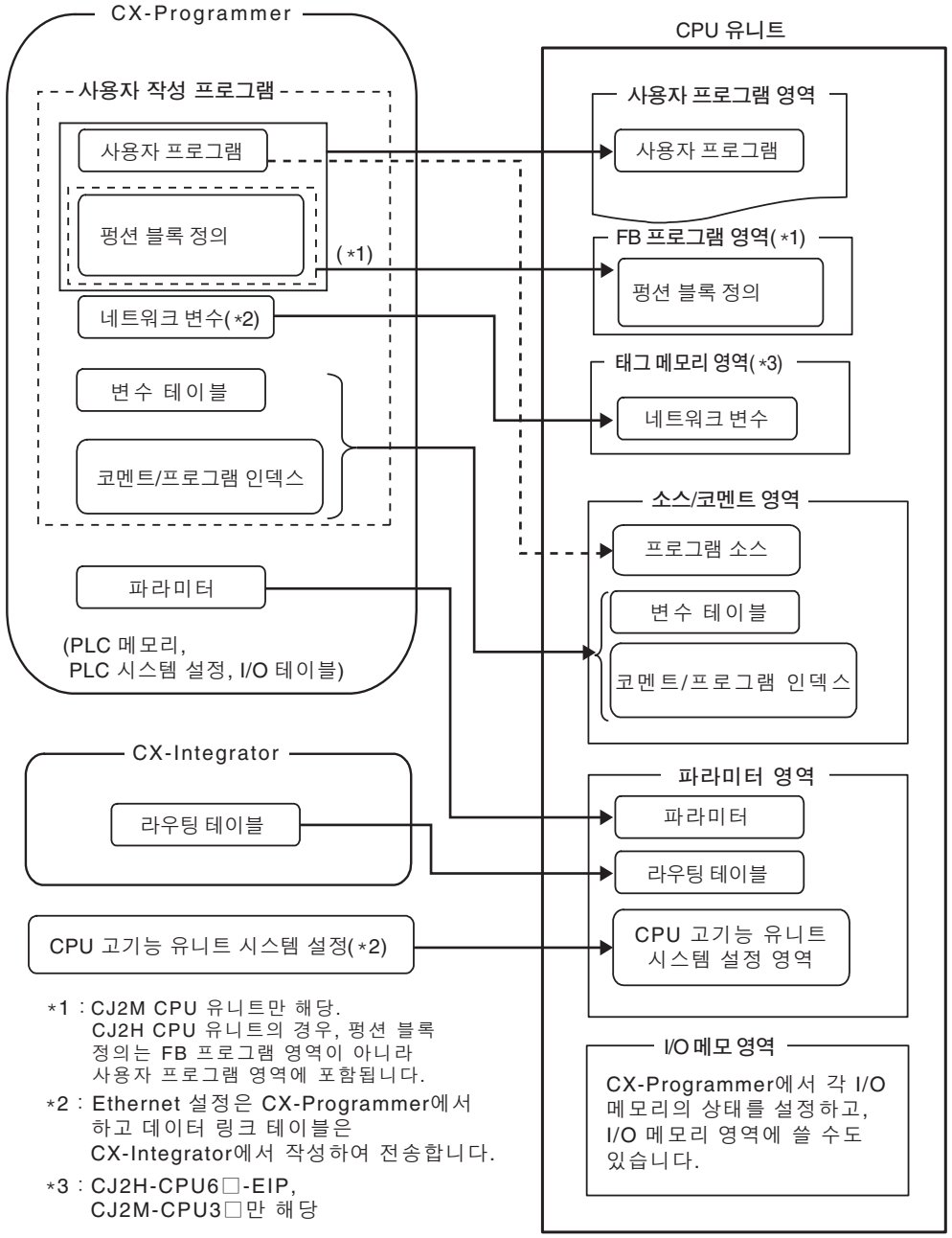
○: 저장 −: 비저장

메모리 영역/저장 데이터	세부 사항	내장 플래시	내장 RAM
사용자 프로그램 영역	CX-Programmer에서 작성한 사용자 프로그램의 실행용(개체) 코드를 저장하는 영역	○	○
FB 프로그램 영역(*1)	CX-Programmer에서 작성한 평션 블록 정의를 저장하는 영역	○	○
파라미터 영역	PLC 초기 설정 영역	○	○
PLC 이름	CX-Programmer로 읽기/조회 가능한 자체 CPU 유니트의 이름 CX-Programmer에서 PLC로 잘못 연결되는 것을 방지할 수 있습니다.		
PLC 시스템 설정	소프트 스위치를 사용한 각종 초기 설정 등 📖 '제9장 PLC 시스템 설정의 설명'을 참조하십시오.		
I/O 테이블	사용자가 지정한 유니트의 장착 상태 정보 📖 '제8장 I/O 할당 및 유니트 설정'을 참조하십시오.		
라우팅 테이블	CX-Integrator로 작성한 FINS 통신용 네트워크 파라미터 📖 "CX-Integrator 조작 설명서"(SBCA-347)를 참조하십시오.		
CPU 고기능 유니트 시스템 설정	특정 CPU 고기능 유니트용 초기 설정 데이터를 CPU 유니트 내부의 메모리에 저장하는 영역 Ethernet 유니트용 Ethernet 설정, Controller Link 유니트용 데이터 링크 파라미터 등이 있습니다.		
I/O 메모리 영역	사용자 프로그램이 읽기/쓰기를 수행하는 메모리 영역 다음과 같은 영역으로 나뉘며, 사용 목적에 따라 구분해서 사용합니다. • CPU 유니트의 전원 OFF 복구 시 지워지는 영역과 유지되는 영역 • 각 유니트와 데이터를 교환하는 영역 및 내부에서 사용하는 영역	—	○
소스/코멘트 영역	CX-Programmer로 작성한 프로그램의 소스 코드 및 코멘트를 저장하는 영역	○	—
프로그램 소스	프로그램(래더/ST/SFC 언어의 태스크/함수 블록 내부)의 소스 코드		
변수 테이블	CX-Programmer로 작성한 변수 테이블(변수명, 주소, I/O 코멘트)		
코멘트	CX-Programmer로 작성한 주석문, 행 코멘트		
프로그램 인덱스	CX-Programmer로 작성한 섹션 정보, 프로그램 코멘트		
네트워크 변수(태그) (*2)	변수 테이블 중의 네트워크 변수 데이터	○	—

*1 CJ2M CPU 유니트만 해당. CJ2H CPU 유니트의 경우, 펑션 블록 정의는 FB 프로그램 영역이 아니라 사용자 프로그램 영역에 저장됩니다.
*2 CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당

2-1-3 주변 툴에서 데이터 전송

주변 툴 CX-Programmer 및 CX-Integrator로 작성한 데이터는 다음과 같이 CPU 유니트의 내부 메모리로 전송됩니다.



*1 : CJ2M CPU 유니트만 해당.
CJ2H CPU 유니트의 경우, 펑션 블록 정의는 FB 프로그램 영역이 아니라 사용자 프로그램 영역에 포함됩니다.

*2 : Ethernet 설정은 CX-Programmer에서 하고 데이터 링크 테이블은 CX-Integrator에서 작성하여 전송합니다.

*3 : CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당

3

CPU 유니트의 동작

이 장에서는 CPU 유니트의 내부 동작에 대해 설명합니다.

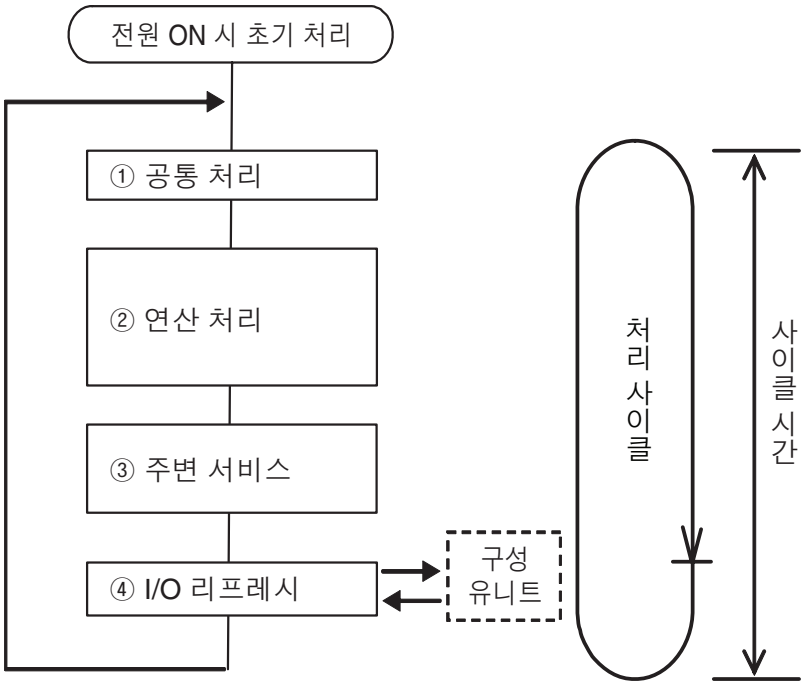
3-1 CPU 유니트의 내부 동작 3-2
3-1-1 내부 동작의 개요 3-2
3-1-2 사이클 시간 3-4
3-1-3 전원 OFF 시의 처리 3-7
3-2 유니트의 동작 모드 3-8
3-2-1 동작 모드 3-8
3-2-2 동작 모드 확인 방법 3-9
3-2-3 동작 모드의 전환 3-10
3-2-4 동작 모드의 자세한 내용 3-14

3-1 CPU 유니트의 내부 동작

3-1-1 내부 동작의 개요

CPU 유니트는 주기적으로 사용자 프로그램을 실행하면서 내부 I/O 메모리 영역을 읽고 씁니다.
또한 I/O 리프레시 및 주변 서비스인 경우에 외부와 데이터를 교환합니다.

CPU 유니트의 내부 동작은 다음 그림과 같습니다.



전원 ON 시 초기 처리

CPU 유니트는 전원이 켜지면 다음과 같은 초기 처리를 수행합니다.

- 장착된 유니트 인식(I/O 할당)
- 등록 I/O 테이블 및 장착되어 있는 유니트 조회
- I/O 메모리 영역 중, 비유지 타입 영역을 IOM(I/O 메모리) 유지 비트 상태에 따라 지우기
- 강제 세트/리셋을 강제 세트/리셋 유지 플래그 상태에 따라 해제
- 메모리 카드가 장착되고 자동 전송용 프로그램이 메모리 카드에 존재하는 경우, 자동 부팅 실행
- 자가 진단(사용자 메모리 확인)
- 사용자 프로그램 복원
- PLC 시스템 설정에 반영 처리

처리 사이클

전원이 켜지고 초기 처리를 수행한 후 CPU 유니트는 4가지 처리(공통 처리, 연산 처리, I/O 리프레시, 주변 서비스)를 반복하여 실행합니다.
이러한 처리를 1사이클 수행하는 데 걸리는 시간을 '사이클 시간'이라고 합니다.

① 공통 처리
사용자 프로그램을 실행할 때의 전처리입니다.
프로그램을 처리할 때 필요한 하드웨어와 소프트웨어를 확인합니다.
공통 처리에 걸리는 시간을 '오버헤드'라고 합니다.

② 연산 처리
사용자 프로그램 내부 명령의 연산 처리입니다.
동작 모드가 '운전' 또는 '모니터' 모드일 때 사용자 프로그램 내부 명령이 실행됩니다.
명령의 입력/출력 처리는 CPU 유니트 내부의 I/O 메모리 영역의 데이터에 대해 실행합니다.

③ 주변 서비스
주변 툴 및 구성 유니트에서 메시지를 요구할 때 발생하는 처리입니다.
CX-Programmer에서 읽기 및 쓰기, 네트워크 통신 유니트의 명령에 의한 읽기 및 쓰기 등이 실행됩니다.

④ I/O 리프레시
구성 유니트와의 데이터 교환을 I/O 메모리 영역에서 실행하는 처리입니다.
연산 처리(명령 실행) 후에 반드시 실행됩니다.

3-1-2 사이클 시간

공통 처리, 연산 처리, I/O 리프레시, 주변 서비스의 합계입니다.
일반적으로 사이클 시간은 처리 내용에 따라 사이클마다 변동됩니다.
사이클 시간 내에 처리하는 내용은 다음과 같습니다.
사이클 시간의 계산 방법은 '제12장 CPU와 사이클 시간'을 참조하십시오.

참고

PLC 시스템 설정의 '사이클 시간 일정화 기능'을 설정하면 사이클 시간을 일정 시간으로 통일할 수 있습니다.
이 설정은 주로 입출력의 응답 시간을 일정하게 하는 경우에 사용합니다.
사이클 시간 일정화 기능의 설정 방법은 '10-2-1 사이클 시간 일정화 기능'을 참조하십시오.

공통 처리

사이클마다 발생하는 오버헤드 시간으로, 다음 사항을 확인합니다.

- 배터리 이상 확인
- 메모리 카드 탈착 확인
- 딥 스위치 감시
- I/O 버스 확인
- 사용자 프로그램의 메모리 확인 등

연산 처리

사용자 프로그램 명령의 처리 시간입니다.

- 사이클릭 태스크 중 실행 가능한 상태의 태스크와 인터럽트 조건이 성립되는 인터럽트 태스크에 할당된(점프된 영역 제외) 프로그램입니다.
- 명령 종류×명령 수, 명령 실행 시의 조건(데이터 처리량 등에 따름)에 좌우됩니다.
- 운전('운전' 모드 또는 '모니터' 모드) 중인 경우 입력 조건이 성립하지 않아 명령이 실행되지 않을 때도 내부적으로는 처리가 수행되며 처리하는 데 시간이 걸리게 됩니다.
  또한 다음과 같은 경우 내부적으로도 처리가 수행되지 않아 명령 처리 시간이 가산되지 않습니다.
  - 프로그램이 할당된 사이클릭 태스크가 대기 상태일 때
  - JMP/CJP/CJPN 명령-JME 명령에 의해 해당 영역을 점프했을 때

참고

사이클 시간을 단축하고 싶은 경우 다음과 같은 프로그래밍을 검토하십시오.

- 사용자 프로그램을 사이클릭 태스크로 분할하고, 실행할 필요가 없을 때는 대기 상태로 만듭니다.
- 프로그램 내부에서 어떤 조건에서 실행할 필요 없는 영역을 JMP/CJP/CJPN 명령-JME 명령으로 점프시킵니다.

주변 서비스

외부에서 부정기적으로 발생하는 서비스 요구에 대한 처리 및 외부에 대한 서비스 요구를 가리킵니다.
CJ 시리즈의 경우에는 주로 FINS 명령을 사용한 서비스에서 발생합니다.
주변 서비스에서는 시스템에서 고정적으로 정한 시간을 각 서비스에 할당하여 사이클마다 실행합니다.
그리고 할당 시간 내에 다 처리하지 못한 경우 나머지는 처리를 실행하지 않고 다음 처리로 이행합니다.

● 주변 서비스의 처리 시간

주변 서비스 처리 시간의 기본값은 직전 사이클 시간의 10%가 할당됩니다.
따라서 직전 사이클 시간에 좌우됩니다.
또한 각 주변 서비스가 어떤 사이클에서도 실행되므로 주변 서비스가 지연될 때는 CX-Programmer의 PLC 시스템 설정 '주변 서비스 일률 시간'에서 사이클 시간의 10%보다 긴 시간으로 설정하십시오.
단, 이런 경우 사이클 시간은 길어집니다.

● 주변 서비스의 종류

서비스 종류	내용
고기능 I/O 유니트의 이벤트 서비스	• 고기능 I/O 유니트, CPU 고기능 유니트의 FINS 명령에 의한 부정기 서비스 요구에 대한 처리(외부 인터럽트 태스크의 기동 요구 등) • CPU 유니트의 위 유니트/보드에 대한 FINS 명령에 의한 부정기 서비스 요구 처리
CPU 고기능 유니트의 이벤트 서비스	
페리페럴(USB) 포트 서비스	• 페리페럴(USB) 포트, 시리얼 포트를 경유하여 주변 툴, PT 또는 상위 PC 등의 FINS 명령, 상위 링크 명령 등에 의한 부정기적인 서비스 요구에 대한 처리(프로그램 전송, 모니터링, 강제 세트/리셋, 온라인 편집 등) • CPU 유니트에서 페리페럴(USB) 포트, 시리얼 포트 서비스를 경유하여 전송하는 부정기적인 서비스 요구 처리(송신 기능)
시리얼 포트 서비스	
통신 포트 서비스	• SEND/SEND2, RECV/RECV2, CMND/CMND2 명령, PMCR/PMCR2 명령에 의해 내부 논리 포트/확장 내부 논리 포트를 사용하여 네트워크 통신, 시리얼 통신, 파일 메모리 액세스를 실행하는 서비스 • 백그라운드 처리 실행 시에는 내부 논리 포트를 사용하여 명령 실행을 백그라운드 처리하는 서비스
파일 액세스 서비스	메모리 카드 또는 EM 파일 메모리에 대한 파일 읽기 및 쓰기 등의 처리
온라인 편집 서비스	온라인 편집 처리 중에 편집 내용을 내장 플래시 메모리에 백업 처리하는 서비스
백업 서비스	프로그램이나 파라미터가 변경된 경우 해당 내용을 내장 플래시 메모리에 백업 처리하는 서비스

참고

고기능 I/O 유니트, CPU 고기능 유니트, 페리페럴(USB) 포트, 시리얼 포트, 파일 액세스, 통신 포트 서비스의 각 서비스별로 서비스 시간을 할당합니다.
각 서비스가 어떤 사이클로 실행되고 서비스가 지연되는 경우, PLC 시스템 설정의 '주변 서비스 시간 일률 설정'에서 비율이 아니라 실제 시간으로 할당 시간을 일률적으로(각 서비스가 같은 시간으로) 설정합니다.

I/O 리프레시

I/O 유니트의 처리 시간은 구성 유니트의 기종과 대수에 좌우됩니다.
대상 유니트별 처리 시간은 고정되어 있습니다.
또한 각 고기능 I/O 유니트에 대해 사이클마다 수행하는 I/O 리프레시를 정지하면 처리 시간을 단축할 수 있습니다.
각 유니트에 대한 I/O 리프레시의 정지는 CX-Programmer의 PLC 시스템 설정 '고기능 I/O 리프레시' 탭으로 설정합니다.
각 유니트의 I/O 리프레시 처리 내용은 다음과 같습니다.

대상 유니트의 종류			최대 데이터 교환량	데이터 교환 영역
기본 I/O 유니트			유니트에 의해 고정	입출력 릴레이 영역
고기능 I/O 유니트	할당 릴레이 영역의 리프레시		10CH/유니트 (유니트 종류에 따라 다름)	고기능 I/O 유니트 릴레이 영역
	유니트 고유의 리프레시	CompoBus/S 마스터 유니트	유니트에 따라 다름	원격 I/O 통신 영역
CPU 고기능 유니트	할당 릴레이 영역의 리프레시		25CH/유니트	CPU 고기능 유니트 릴레이 영역
	할당 DM 영역의 리프레시		100CH/유니트	CPU 고기능 유니트용 DM 영역
	유니트 고유의 리프레시 (예: 오른쪽에 소개)	내장 EtherNet/IP 포트(*)	EtherNet/IP 기능에 따름	태그 데이터 링크 영역
		EtherNet/IP 유니트		
		Controller Link 유니트	유니트에 따라 다름	데이터링크를리프레시하는 영역 (자동 설정, 임의 설정 모두)
		DeviceNet 유니트	유니트에 따라 다름	원격 I/O 통신 수행 영역(고정 할당, 자유 할당 모두)
		시리얼 커뮤니케이션 유니트	프로토콜 매크로 기능에 따라 다름	프로토콜 매크로 기능에 의한 지정 송수신 데이터
		EtherNet 유니트	유니트에 따라 다름	특정 비트 조작에 의한 소켓 서비스의 지정 송수신 데이터

* CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당. CJ2H-CPU6□-EIP의 내장 EtherNet/IP 포트는 EtherNet/IP 유니트(CJ1W-EIP21)와 동등한 성능, 기능을 가지고 있지만 CJ2M-CPU3□의 내장 EtherNet/IP 포트는 성능이 크게 다릅니다. 자세한 내용은 "EtherNet/IP 유니트 사용 설명서"(SBCD-342)의 '10-4 CJ2M 내장 EtherNet/IP 포트의 태그 데이터 링크 성능'을 참조하십시오.

참고

'I/O 리프레시' 시가 아니라 명령 실행 시에 구성 유니트와 교환 처리하는 것도 가능합니다. 이것을 '매회 리프레시'라고 합니다((!)가 붙은 명령어 또는 IORF/FIORF/DLNK 명령이 있습니다). 또한 특정 고기능 유니트 전용으로 데이터 교환을 고속화(직접 처리)한 다음과 같은 명령을 사용할 수 있습니다.

명령에 대한 자세한 내용은 "명령 참조 설명서"(SBCA-351)를 참조하십시오.

- 아날로그 입력 직접 변환(AIDC) 명령(CJ1W-AD042 전용) (*1)
- 아날로그 출력 직접 변환(AODC) 명령(CJ1W-DA042V 전용) (*1)
- 시리얼 커뮤니케이션 유니트 시리얼 포트 직접 송신(DTXDU) 명령 (CJ1W-SCU□2 전용) (*1)
- 시리얼 커뮤니케이션 유니트 시리얼 포트 직접 수신(DRXDU) 명령 (CJ1W-SCU□2 전용) (*1)
- 위치 결정 고속 기동(NCDMV) 명령(CJ1W-NC□□4, NC□81 전용) (*2)
- 위치 결정 시작 트리거(NCDTR) 명령(CJ1W-NC□81 전용) (*2)

*1 CJ2H-CPU6□(-EIP)의 유니트 Ver.1.1 이상 또는 CJ2M CPU 유니트만 해당됩니다.
*2 CJ2H-CPU6□(-EIP)의 유니트 Ver.1.3 이상 또는 CJ2M CPU 유니트만 해당됩니다.

3-1-3 전원 OFF 시의 처리

CPU 유니트는 운전 중에 전원이 꺼지고 '전원 OFF'라고 확정되면 실행 중인 1개 명령의 실행을 완료한 후 전원 OFF 처리를 수행합니다.

전원 OFF 처리는 전원 OFF 인터럽트 태스크(*1) 설정에 의해 다음과 같이 처리됩니다.

- 전원 OFF 인터럽트 태스크가 유효한 경우
  CPU 유니트는 전원 OFF 인터럽트 태스크를 실행한 후 리셋됩니다.
- 전원 OFF 인터럽트 태스크가 무효인 경우
  CPU 유니트가 리셋됩니다(*2).

*1 전원 OFF 인터럽트 태스크는 CPU 유니트가 전원 OFF에 의해 세트되기 직전에 실행하는 태스크입니다.
*2 운전 중에 전원이 꺼져도 일정 영역의 명령군에 대해서는 해당 명령을 실행한 후 전원 OFF 처리를 수행하는 것도 가능합니다.

또한 AC 전원의 경우 10ms 미만, DC 전원의 경우 2ms 미만의 순간 정전에 대해서는 운전을 계속합니다. 전원 사정이 나쁜 경우를 위해 전원 OFF를 검출한 후 '전원 OFF'로 확정될 때까지의 시간을 연장하는 것도 가능합니다.

전원 OFF 시의 처리에 대한 자세한 내용은 '부록-5 전원 OFF 시 PLC의 동작'을 참조하십시오.

3-2 유니트의 동작 모드

3-2-1 동작 모드

'동작 모드'는 CPU 유니트 내부 동작 상태, 설정 가능 상태를 전환하는 기능입니다.
동작 모드에는 다음 3가지가 있습니다.

- **운전 모드**
  프로그램의 본 운전을 하기 위한 모드이며 가장 고속으로 동작합니다.
  프로그램은 실행 상태입니다.
  강제 세트/리셋, I/O 메모리 현재값 변경, 온라인 편집 등은 수행할 수 없습니다.

- **모니터 모드**
  프로그램의 시운전 조정 등을 수행하는 모드입니다.
  프로그램은 실행 상태입니다.
  강제 세트/리셋, I/O 메모리 현재값 변경, 온라인 편집 등을 수행할 수 있습니다.

- **프로그램 모드**
  프로그램, PLC 시스템 설정 등을 전송하고 I/O 테이블을 작성하기 위한 모드입니다.
  프로그램은 정지 상태입니다.

각 동작 모드의 전환은 CPU 유니트에 연결된 CX-Programmer에서 실행합니다.

3-2-2 동작 모드 확인 방법

CPU 유니트 전면 LED에서 확인

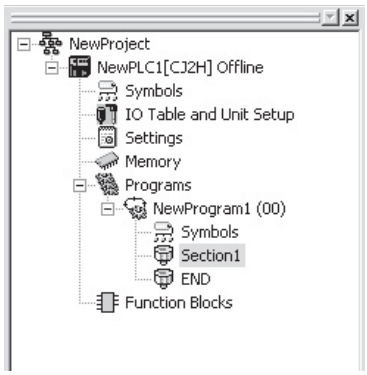
동작 모드에 따라 CPU 유니트 전면 LED(RUN LED)가 다음과 같이 표시됩니다.

동작 모드	CPU 유니트의 RUN LED	비고
'프로그램' 모드일 때	소등	–
'운전' 모드 또는 '모니터' 모드일 때	녹색 점등	'운전' 모드, '모니터' 모드 중 어느 상태인지는 CX-Programmer를 사용하여 확인하십시오.

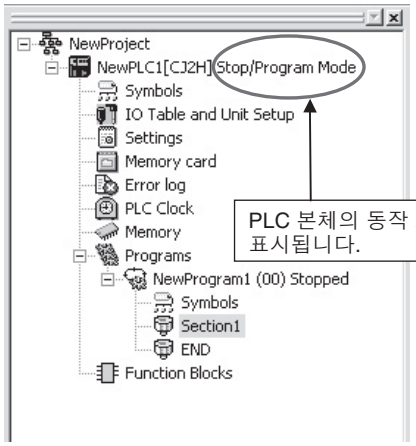
CX-Programmer에서 확인

동작 모드는 CX-Programmer의 프로젝트 트리 또는 상태 표시줄에서 확인하십시오.

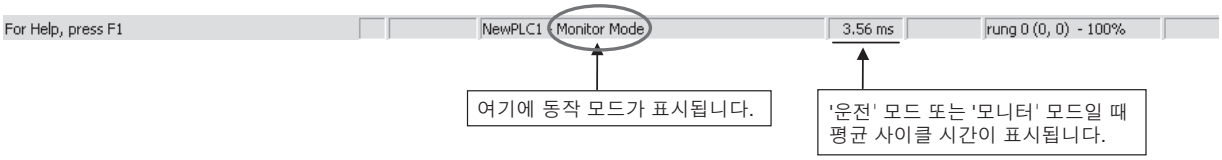
- 프로젝트 트리
  - 오프라인일 때



  - 온라인일 때



- 상태 표시줄



참고

운전 중 외부로 출력하려면 다음과 같이 수행합니다.
전원 유니트 CJ1W-PA205R의 경우 전원 유니트에서 운전 중 출력이 가능합니다.
'운전' 모드 또는 '모니터' 모드에서 운전 정지 이상 이외의 경우에는 운전 중 출력(RUN 접점 출력)이 ON으로 됩니다.

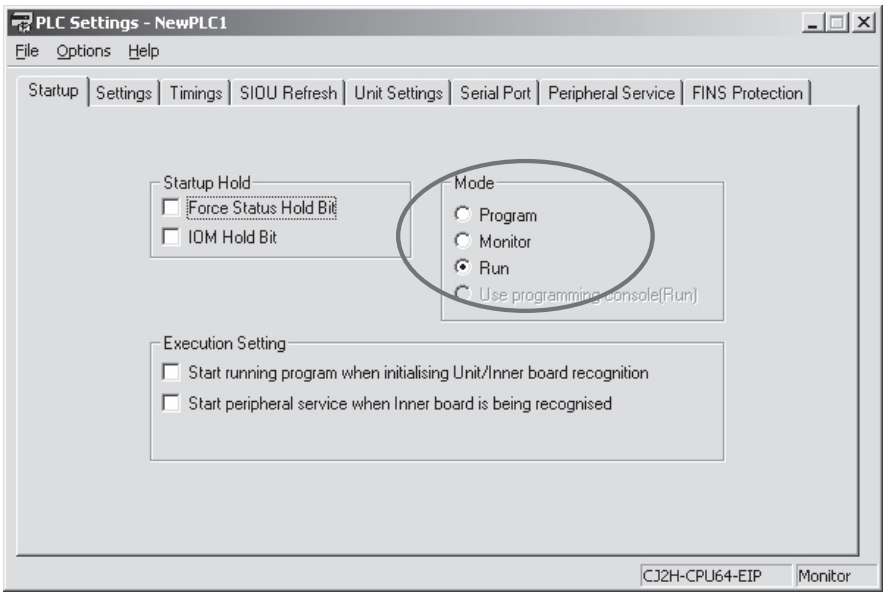
3-2-3 동작 모드의 전환

동작 모드 전환 방법

동작 모드는 CX-Programmer에서 전환할 수 있습니다.

- **전원 ON 시 동작 모드 변경 방법**

  CPU 유니트의 전원 투입 시 동작 모드는 '운전 모드'(초기값)입니다.
  전원 투입 시의 동작 모드를 '프로그램' 또는 '모니터' 모드로 변경할 때는 CX-Programmer에서 [PLC 설정]의 '전원 ON 시 동작 설정'을 통해 변경합니다.



- **전원 투입 후 동작 모드 전환 방법**

  다음 중 한 가지 방법으로 전환합니다.
  - [PLC]의 [작업 모드] 메뉴에서 [프로그램]-[모니터]-[운전]을 선택합니다.
  - 프로젝트 트리에서 PLC를 선택하고 마우스 오른쪽 버튼을 클릭하여 [작업 모드] 메뉴에서 [프로그램]-[모니터]-[운전]을 선택합니다.

동작 모드 전환 시 및 전원 OFF 복구 시의 I/O 메모리 유지 상태

'운전'/'모니터' 모드와 '프로그램' 모드 간에 동작 모드를 전환하면 CIO 영역, WR 영역 등과 같은 I/O 메모리의 일부 영역(비유지 영역)은 지워집니다.
동작 모드 전환 시에도 해당 영역을 유지하고 싶은 경우 특수 보조 릴레이의 ' IOM(I/O 메모리) 유지 비트' (A500.12)를 1(유지)로 설정합니다.

모드 변경	비유지 영역	유지 영역
	입출력 릴레이 데이터 링크 릴레이 CPU 고기능 유니트 릴레이 고기능 I/O 유니트 릴레이 내부 보조 릴레이 타이머 현재값/플래그 인덱스 레지스터 데이터 레지스터 태스크 플래그 (특수 보조 릴레이는 주소에 따라 유지/비유지 영역이 있음)	유지 릴레이 데이터 메모리 확장 데이터 메모리 카운터 현재값/플래그 (특수 보조 릴레이는 주소에 따라 유지/비유지 영역이 있음)
'운전' 또는 '모니터'→ '프로그램'	지우기(*1)	유지
'프로그램'→'운전' 또는 '모니터'		
'운전'←→'모니터'	유지	유지
전원 OFF 복구 시(*2)	지우기(*3)	유지

*1 IOM(I/O 메모리) 유지 비트를 0(OFF)으로 한 경우입니다.
1(ON)로 설정한 경우에는 다음과 같이 유지됩니다.

IOM(I/O 메모리) 유지 비트 (A50012)	I/O 메모리			기본 출력 유니트의 출력 접점		
	동작 모드 변경 프로그램←→ 운전/모니터	운전 정지 이상 발생		동작 모드 변경 프로그램←→ 운전/모니터	운전 정지 이상 발생	
		FALS 명령 실행 이외의 운전 정지 이상	FALS 명령 실행		FALS 명령 실행 이외의 운전 정지 이상	FALS 명령 실행
OFF	지우기	지우기	유지	OFF	OFF	OFF
ON	유지	유지	유지	유지	OFF : 운전 정지 이상 발생 시, CPU 유니트 내의 I/O 메모리 값이 유지되어도 기본 출력 유니트의 출력 접점은 OFF가 됩니다.	

*2 전원 OFF 복구 시에는 IOM(I/O 메모리) 유지 비트 A500.12의 내용도 지워집니다.
내용이 지워지지 않도록 하기 위해서는 CX-Programmer에서 PLC 시스템 설정의 [시작 대기]의 [IOM(I/O 메모리) 유지 비트] 확인란을 선택한 후 전송하십시오.
*3 태스크 플래그는 IOM(I/O 메모리) 유지 비트 A500.12의 내용에 관계 없이 지워집니다.

동작 모드 전환 시 기본 출력 유니트의 출력 접점 상태

'운전'/'모니터' 모드에서 '프로그램' 모드로 동작 모드를 전환하는 경우 기본 출력 유니트의 출력 접점은 해당 시점에 대한 I/O 메모리의 출력 릴레이 상태를 나타냅니다.
'IOM(I/O 메모리) 유지 비트'(A500.12)의 ON/OFF 상태에 따라 기본 출력 유니트의 출력 접점은 다음과 같은 상태가 됩니다.

- OFF(유지 안 함)일 때
  출력 릴레이는 클리어되므로 기본 출력 유니트의 출력 접점은 OFF가 됩니다.
- ON(유지)일 때
  '프로그램' 모드로 전환한 경우에는 출력 릴레이의 상태가 유지되므로 그에 따라 기본 출력 유니트의 출력 접점이 출력됩니다.



사용상의 주의 사항

'IOM(I/O 메모리) 유지 비트'(A500.12)를 1(유지)로 설정한 경우 주의 사항
CPU 유니트가 '운전'/'모니터' 모드에서 '프로그램' 모드로 변경되면 출력 릴레이의 I/O 메모리 값은 클리어(OFF)되지 않고 '프로그램' 모드로 변경되기 전의 상태를 유지합니다.
따라서 다시 '프로그램' 모드에서 '운전'/'모니터' 모드로 변경한 경우 '프로그램' 모드로 변경하기 전의 I/O 메모리 값이 출력되므로 주의하십시오.
단, 운전 정지 이상(FALS 명령 실행 포함)에 의해 운전이 정지됐을 때는 CPU 유니트 내의 I/O 메모리 값이 유지되어도 기본 출력 유니트의 출력 접점은 모두 OFF가 됩니다.

'프로그램' 모드일 때 기본 출력 유니트의 출력 접점 상태

'프로그램' 모드일 때도 기본 출력 유니트의 출력 접점은 I/O 메모리의 출력 릴레이 상태를 항상 나타냅니다. 따라서 CX-Programmer 등에서 I/O 메모리 영역의 출력 릴레이 값을 변경하면 I/O 리프레시가 수행되고 해당 값이 기본 출력 유니트에서 출력됩니다.

⚠ 경고

운전 정지 상태('프로그램' 모드)에도 CPU 유니트는 I/O 리프레시를 실행합니다. 따라서 다음과 같은 조작을 통해 기본 출력 유니트에 할당된 출력 릴레이 영역의 데이터 또는 고기능 I/O 유니트/CPU 고기능 유니트에 할당된 각 릴레이 영역의 데이터를 변경하는 경우에는 안전한지 충분히 확인한 후 수행하십시오.
기본 출력 유니트 또는 고기능 I/O 유니트/CPU 고기능 유니트에 연결된 부하가 예상치 못한 동작을 할 위험이 있습니다.

- 주변 툴(PC 툴)을 사용하여 I/O 메모리의 CPU 유니트로 전송 조작
- 주변 툴을 사용한 현재값 변경 조작
- 주변 툴을 사용한 강제 세트/리셋 조작
- 메모리 카드 또는 EM 파일 메모리에서 CPU 유니트로 I/O 메모리 파일 전송 조작
- 네트워크의 다른 PLC 또는 상위 컴퓨터에서 I/O 메모리 전송 조작

기본 출력 유니트의 출력 접점을 강제로 항상 OFF로 하고 싶을 때(부하를 차단하고 싶은 경우)는 부하 차단 플래그(A500.15)를 1(ON)로 설정하십시오(*).
부하 차단 중에는 CPU 유니트 전면의 INH LED가 점등합니다.

* 부하 차단 플래그(A500.15)는 동작 모드 변경 및 전원 ON/OFF 시에도 유지되며 부하 차단 기능은 유효 상태 그대로입니다.

3-2-4 동작 모드의 자세한 내용

각 동작 모드의 상태에 대한 자세한 내용은 다음과 같습니다.

<table>
<tr><th colspan="2">동작 모드</th><th>'프로그램' 모드</th><th>'모니터' 모드</th><th>'운전' 모드</th></tr>
<tr><td colspan="2">목적</td><td>운전 정지,
프로그램 등의 전송</td><td>시운전 조정</td><td>본 운전</td></tr>
<tr><td colspan="2">프로그램의 상태</td><td>정지</td><td>실행</td><td>실행</td></tr>
<tr><td colspan="2">RUN LED 표시</td><td>소등</td><td>점등</td><td>점등</td></tr>
<tr><td rowspan="10">CX-Programmer의 조작</td><td>I/O 테이블 작성</td><td rowspan="3">가능</td><td colspan="2" rowspan="3">불가능</td></tr>
<tr><td>PLC 시스템 설정의 전송</td></tr>
<tr><td>프로그램 다운로드</td></tr>
<tr><td>프로그램 확인</td><td colspan="2" rowspan="4">가능</td><td rowspan="4">불가능</td></tr>
<tr><td>온라인 편집</td></tr>
<tr><td>강제 세트/리셋에 의한 배선 확인</td></tr>
<tr><td>I/O 메모리의 현재값 변경</td></tr>
<tr><td>타이머/카운터의 현재값 및 설정값 변경</td><td colspan="2" rowspan="4">가능</td></tr>
<tr><td>프로그램 업로드</td></tr>
<tr><td>I/O 메모리 모니터링</td></tr>
<tr><td>프로그램 모니터링</td></tr>
<tr><td colspan="2">I/O 리프레시</td><td colspan="3">실행</td></tr>
<tr><td colspan="2">사이클릭 태스크</td><td>정지</td><td colspan="2">'운전 시작 시 기동'이 세트된 태스크 또는 태스크 기동 명령으로 기동된 태스크는 실행 상태임.
그 밖의 태스크는 정지 상태임</td></tr>
<tr><td colspan="2">인터럽트 태스크</td><td>정지</td><td colspan="2">인터럽트 조건이 성립되면 실행</td></tr>
<tr><td rowspan="2">출력 릴레이 영역의 할당 기본 출력 유니트 접점(*1)</td><td>IOM(I/O 메모리) 유지 비트 OFF 시</td><td>'프로그램' 모드로 전환한 직후에는 OFF(*2)</td><td rowspan="2">프로그램에 따라 다름</td><td rowspan="2">프로그램에 따라 다름</td></tr>
<tr><td>IOM(I/O 메모리) 유지 비트 ON 시</td><td>'프로그램' 모드로 전환한 직후에는 전환 직전의 상태 유지(*2)</td></tr>
</table>

*1 부하 차단 플래그(A500.15)가 1(ON)일 때는 동작 모드나 I/O 메모리의 값에 관계 없이 기본 출력 유니트의 출력 접점이 OFF가 됩니다. 전원 OFF→ON으로 해도 OFF 상태 그대로입니다.

*2 '프로그램' 모드인 동안 주변 툴 또는 PT 등에 의해 값이 변경된 경우 출력 릴레이의 상태가 반영됩니다.

4

CPU 유니트의 초기 설정

이 장에서는 CPU 유니트의 초기 설정 종류에 대해 설명합니다.

4-1 CPU 유니트의 초기 설정에 대한 개요 4-2
4-1-1 CPU 유니트의 초기 설정에 대한 개요 4-2
4-2 PLC 시스템 설정 4-7
4-3 I/O 테이블 작성 4-8
4-3-1 I/O 테이블 4-8
4-3-2 자동 할당 4-9
4-3-3 수동 할당 4-9
4-4 라우팅 테이블 설정 4-10
4-4-1 라우팅 테이블 4-10
4-4-2 라우팅 테이블이 필요한 경우 4-12
4-4-3 라우팅 테이블의 설정/전송 방법 4-13
4-5 고기능 유니트용 할당 DM 영역 설정 4-14
4-5-1 고기능 유니트용 할당 DM 영역 설정 4-14
4-5-2 고기능 유니트용 할당 DM 영역 설정 방법 4-14
4-6 CPU 고기능 유니트 시스템 설정 영역 4-15
4-6-1 CPU 고기능 유니트 시스템 설정 영역 4-15
4-6-2 CPU 고기능 유니트 시스템 설정 영역의 설정 방법 4-15

4-1 CPU 유니트의 초기 설정에 대한 개요

4-1-1 CPU 유니트의 초기 설정에 대한 개요

CPU 유니트에 필요한 초기 설정에는 CPU 전면에 있는 딥 스위치나 로터리 스위치를 이용한 하드웨어 설정과 CX-Programmer 등을 이용한 파라미터 설정 등의 소프트웨어 설정이 있습니다.

하드웨어 설정

● **딥 스위치**

용도에 따라 딥 스위치를 각각 설정합니다.

대상 유니트	용도	설정 방법
CPU 유니트 전면의 딥 스위치	주요 용도: • 프로그램 쓰기 금지 : SW1을 ON • 전원 ON 시 자동 전송 : SW2를 ON • 시리얼 포트에서 툴 버스 연결 : SW5를 ON • 사용자 지정용 스위치 사용 : SW6을 ON • 간이 백업 : SW7을 ON	전원 OFF 상태에서 설정하고 전원을 ON

● **로터리 스위치**

CJ2H-CPU6□-EIP, CJ2M-CPU3□의 내장 EtherNet/IP 포트에 CPU 고기능 유니트와 동일하게 유니트 번호와 노드 주소를 할당합니다.

설정한 유니트 번호에 따라 릴레이 영역과 DM 영역에 전용 영역이 자동으로 할당됩니다.

하드웨어 설정에 대한 자세한 내용은 "CJ 시리즈 CJ2 CPU 유니트 사용 설명서 하드웨어편" (SBCA-349)의 '3-1 CPU 유니트'를 참조하십시오.

소프트웨어 설정

대상 유니트에 대한 각 소프트웨어 설정 용도와 설정 방법은 다음과 같습니다.

대상 유니트	설정 데이터		용도	설정 방법	주변 툴에서 작성한 파일	백업 대상
CPU 유니트용	파라미터 영역	PLC 이름	CPU 유니트에 대한 잘못된 연결 방지	CX-Programmer	CX-Programmer 프로젝트 파일(.CXP)	내장 플래시 메모리
		PLC 시스템 설정	기본값 사양 이외에서 사용할 때			
		등록 I/O 테이블	잘못된 연결의 확인이 필요하며 자동 할당을 수행하고 싶은 경우 또는 빈 채널 확보 등 임의로 할당하고 싶을 때			
		라우팅 테이블	FINS 네트워크 통신 유니트(*1)를 1대 이상 장착할 때 필요	CX-Integrator	라우팅 테이블 파일 (.RTG)	
Ethernet 유니트 설정		CPU 고기능 유니트 시스템 설정 영역 (CPU 고기능 유니트 시스템 설정) (*2)	Ethernet 유니트 사용 시	CX-Programmer	CX-Programmer 프로젝트 파일(.CXP)	내장 플래시 메모리
Controller Link 유니트의 데이터 링크 테이블			Controller Link에서 임의 설정 데이터 링크 사용 시	CX-Integrator	데이터 링크 테이블 (.CL2)	
FL-net 유니트 설정			FL-net 유니트 사용 시	FL-net 설정 툴 CX-FLnet	FL-net 설정 툴 설정 파일(.CSV)	
고기능 유니트용	I/O 메모리	고기능 유니트의 할당 DM 영역(*2)	고기능 유니트를 사용할 때	• CX-Programmer 또는 • 사용자 프로그램	CX-Programmer 프로젝트 파일(.CXP)	내장 RAM (배터리 백업)

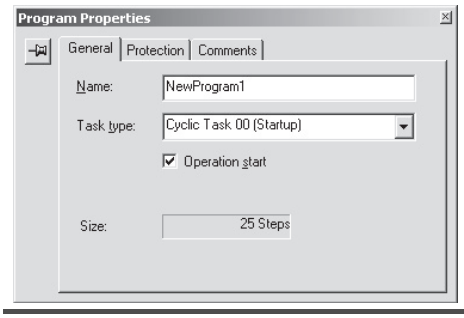
*1 'FINS 네트워크 통신 유니트'란?
- CPU 고기능 유니트의 네트워크 통신 유니트:
  Controller Link 유니트, SYSMAC LINK 유니트, Ethernet 유니트, DeviceNet 유니트, FL-net 유니트
- CJ2H-CPU6□-EIP, CJ2M-CPU3□ 내장 EtherNet/IP 포트를 FINS 네트워크 통신으로 사용하는 경우
- 시리얼 커뮤니케이션 유니트에서 시리얼 게이트웨이 기능을 사용하며 라우팅 테이블을 사용하는 경우
- 또한 고기능 I/O 유니트의 네트워크 통신 유니트(예: CompoNet 마스터 유니트)는 라우팅 테이블의 대상 유니트가 아닙니다.

*2 위 내용은 CPU 유니트에 저장된 고기능 유니트용 초기 설정 데이터입니다.
각 고기능 유니트에 저장된 데이터/프로그램은 별도로 각 고기능 유니트용 주변 툴을 사용하여 작성하며 CPU 유니트를 경유하여 각 고기능 유니트로 전송됩니다.

참고

프로그램의 태스크 설정

프로그램을 어느 태스크에 할당할 것인가에 대한 초기 설정은 CX-Programmer에서 다음의 각 프로그램 속성을 통해 설정합니다.



- **PLC 이름**

  사용자가 설정하는 CPU 유니트의 이름입니다.
  CX-Programmer의 [PLC] 메뉴－[PLC 정보]－[PLC 정보]에서 설정합니다.
  CX-Programmer에서 온라인으로 연결하는 경우 프로젝트에 있는 PLC 이름과 일치하는지 여부를 확인합니다.

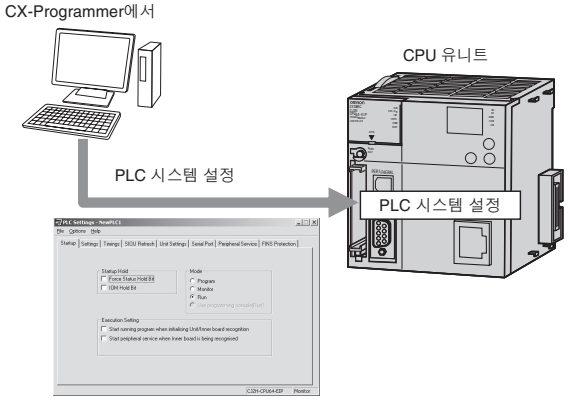
- **PLC 시스템 설정**

  CPU 유니트를 기본값 이외의 사양에서 사용하는 경우 PLC 시스템 설정에서 변경합니다.
  CPU 유니트는 기본적으로 다음과 같이 설정되어 있습니다.

  설정 예)
  전원 ON 시 동작 모드＝'운전' 모드
  시리얼 포트＝상위 링크
  주변 서비스 일률 시간 설정＝사이클 시간의 10%

  이러한 기본값 사양 이외에서 사용할 때는 CX-Programmer에서 PLC 시스템 설정을 변경하고 CPU 유니트로 전송합니다.

  ⇒ CX-Programmer에서 PLC 시스템 설정을 전송합니다.

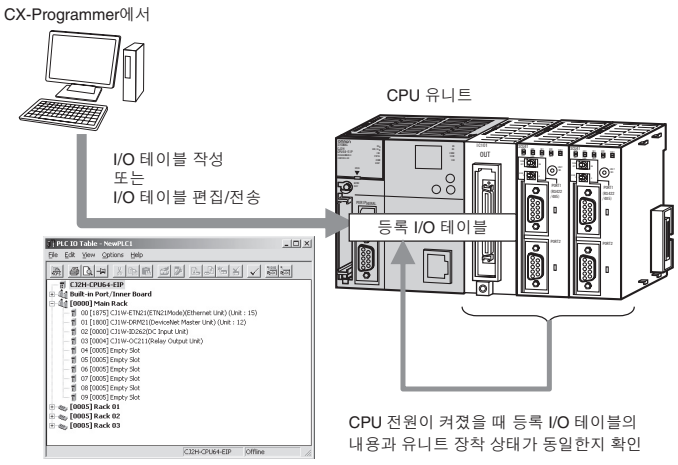


- **I/O 테이블**

  유니트의 장착 위치를 사용자가 임의로 지정하고, 유니트의 잘못된 연결을 검출하는 경우 또는 CPU 유니트의 I/O 할당을 수동 할당하는 경우에는 I/O 테이블을 작성합니다.
  CPU 유니트의 I/O 할당은 기본적으로 전원 투입 시 연결되는 순서로 자동 할당됩니다.
  I/O 테이블은 CX-Programmer에서 다음과 같은 조작을 통해 작성하십시오.

  - 온라인일 때: 장착된 유니트의 'I/O 테이블 작성' 조작
  - 오프라인일 때: 실제 기기 없이 I/O 테이블 편집 후 전송 조작

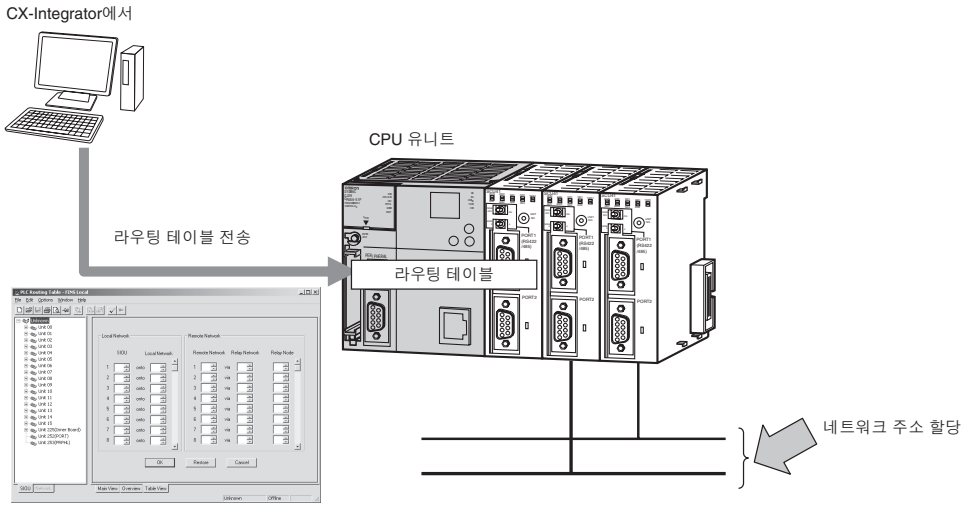
  ⇒ CX-Programmer에서 I/O 테이블을 전송합니다.



- **라우팅 테이블**

  CPU 유니트에 1대 이상의 FINS 네트워크 통신 유니트(CPU 고기능 유니트)를 장착한 경우, CX-Integrator를 사용하여 라우팅 테이블을 작성해야 합니다.
  라우팅 테이블(자체 네트워크 테이블)은 각 FINS 네트워크 통신 유니트에 네트워크 주소를 할당합니다.

  ⇒ CX-Integrator에서 라우팅 테이블(자체 네트워크 테이블)을 설정하고 CPU 유니트로 전송합니다.



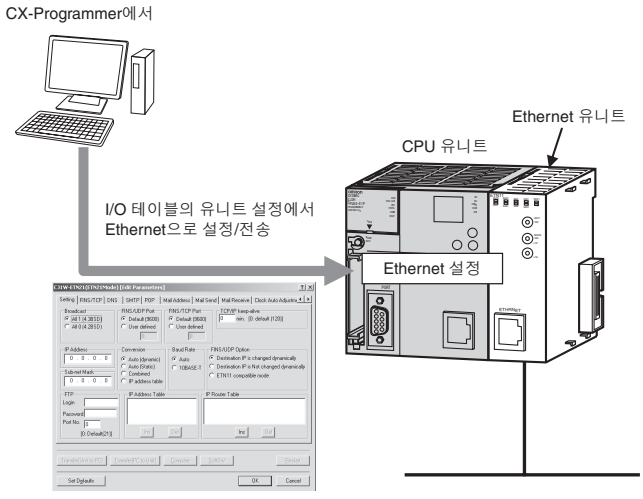
- **CPU 고기능 유니트 시스템 설정 영역**

  Ethernet 유니트, Controller Link 유니트, FL-net 유니트 등의 특정 유니트를 사용하는 경우, 각 유니트에 맞는 설정을 한 후 CPU 유니트로 전송해야 합니다.

  예) Ethernet 유니트를 사용하는 경우

  CX-Programmer를 사용하여 Ethernet을 설정합니다.
  Ethernet 설정은 CX-Programmer의 I/O 테이블에 있는 Ethernet 유니트를 마우스 오른쪽 단추로 클릭한 후 '고기능 유니트 설정 편집'을 선택하여 수행합니다.

  ⇒ CX-Programmer의 I/O 테이블에서 Ethernet 유니트를 설정하여 CPU 유니트로 전송합니다.



참고

Controller Link 유니트를 사용할 때는 임의 설정한 데이터 링크 테이블을 사용하는 경우에 설정을 수행하여 CPU 유니트에 저장합니다.

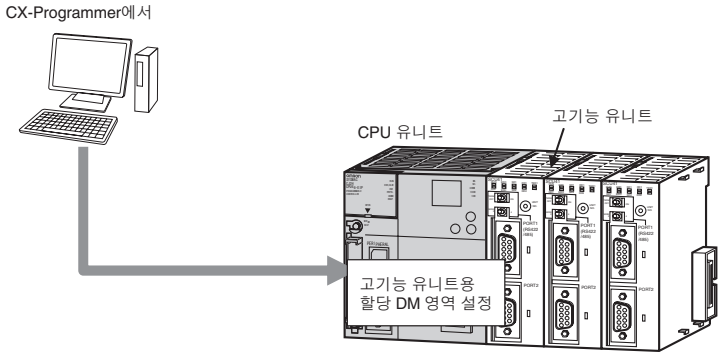
● 고기능 유니트용 할당 DM 영역 설정

고기능 I/O 유니트 또는 CPU 고기능 유니트를 사용하는 경우, 할당 DM 영역을 설정하여 CPU 유니트로 전송합니다.

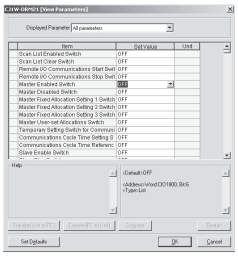
할당 DM 영역은 CX-Programmer에서 다음과 같은 방법으로 설정합니다.

- I/O 테이블에서 각 고기능 유니트를 마우스 오른쪽 단추로 클릭하고 '고기능 유니트 설정 편집'을 선택하여 할당 DM 영역을 설정합니다.
- PLC 메모리에서 할당 DM 영역을 설정합니다.

⇒ CX-Programmer에서 각 고기능 유니트의 할당 DM 영역을 설정하고 CPU 유니트로 전송합니다.

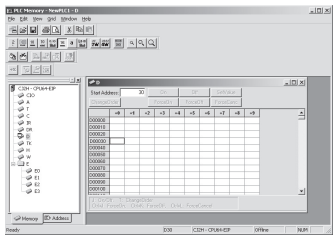


I/O 테이블의 유니트 설정에서
할당 DM 영역을 설정/전송



또는

PLC 메모리의 할당 DM 영역 설정



4-2 PLC 시스템 설정

PLC 시스템 설정은 CPU 유니트의 기본 설정입니다.
기본값(초기값) 이외의 사양에서 CJ2 CPU 유니트를 사용할 때 필요합니다. PLC 시스템 설정은 CX-Programmer에서 조작합니다.

PLC 시스템 설정에 대한 자세한 내용은 📖'제9장 PLC 시스템 설정'을 참조하십시오.

4-3 I/O 테이블 작성

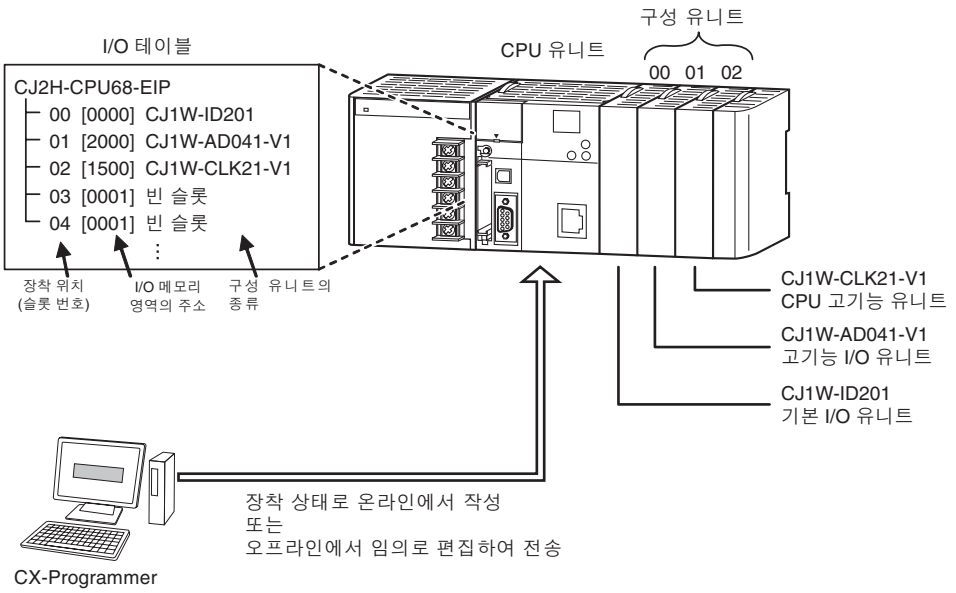
I/O 테이블은 CX-Programmer에서 작성합니다.
I/O 테이블 작성은 필수가 아니며 다음과 같은 경우에만 작성합니다.

- 잘못된 연결을 확인할 필요가 있을 때
- 빈 채널 확보 등을 위해 임의 할당을 해야 할 때

위와 같이 잘못된 연결을 확인할 필요가 없으며 자동 할당으로 문제가 없는 경우에는 I/O 테이블 작성 조작이 필요 없습니다.

4-3-1 I/O 테이블

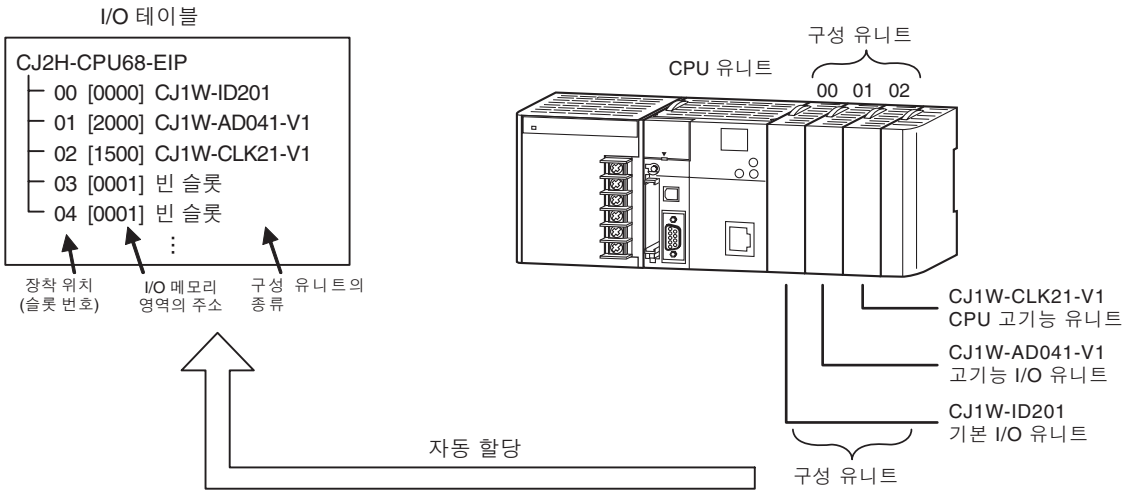
'I/O 테이블'에는 CPU 유니트에 장착된 구성 유니트의 종류와 장착 위치(유니트 구성)가 등록됩니다.
I/O 테이블을 작성하면 CPU 전원을 켰을 때 유니트 장착 상태가 I/O 테이블의 등록 정보와 다르지 않은지 여부를 확인합니다.
I/O 테이블은 CX-Programmer를 사용하여 온라인 장착 상태에서 작성하거나 오프라인에서 임의로 편집하여 전송합니다.



CPU 유니트는 이 정보에 따라 자신에게 장착되어 있는 구성 유니트(실제 I/O 테이블)를 조회하여 자동으로 실행합니다.
일치하지 않는 경우 'I/O 설정 이상'이 되며 운전을 시작할 수 없습니다.

4-3-2 자동 할당

I/O 테이블을 작성하지 않는 방식입니다.
전원을 켤 때마다 장착된 유니트의 위치를 바탕으로 각 유니트의 정보를 I/O 메모리에 할당하여 각 유니트와 데이터를 교환합니다.
I/O 테이블이 없으므로 유니트 구성을 조회하여 확인할 수 없습니다.



4-3-3 수동 할당

I/O 테이블을 작성하는 I/O 할당 방법입니다.
다음과 같은 방법으로 CX-Programmer에서 I/O 테이블을 작성할 수 있습니다.

- 'I/O 테이블 작성' 조작
  CX-Programmer에서 온라인으로 연결하여 장착된 유니트 구성을 바탕으로 I/O 테이블을 자동으로 작성하는 방법입니다.
- 'I/O 테이블 편집/전송' 조작
  오프라인일 때 CX-Programmer를 사용하여 임의 위치에서 I/O 테이블을 수동으로 작성하고 CPU 유니트로 전송하는 방법입니다.

어느 방법이든 CPU 유니트의 전원을 켰을 때 장착되어 있는 유니트 구성과 I/O 테이블의 유니트 구성이 다른 경우, 'I/O 설정 이상'(A401.10이 ON)이 발생하여 운전을 시작할 수 없습니다.
I/O 테이블의 할당 방법에 대한 자세한 내용은 '8-1 I/O 할당'을 참조하고 작성 방법은 '8-2 고기능 유니트 설정'을 참조하십시오.

4-4 라우팅 테이블 설정

PLC에 1대 이상의 FINS 네트워크 통신 유니트를 장착하고 다음과 같은 동작을 수행하는 경우, CX-Integrator를 사용하여 라우팅 테이블을 설정해야 합니다.

- 주변 툴 또는 래더 프로그램의 명령에서 액세스할 네트워크를 전환하는 경우
- 네트워크 간의 계층을 벗어나는 경우

4-4-1 라우팅 테이블

오므론이 제작한 네트워크 통신 유니트(CPU 고기능 유니트) 간에는 전송로에 관계 없이 FINS(핀즈)라는 통신 방식을 사용합니다.
'라우팅 테이블'은 이 FINS 네트워크 통신을 할 때 필요한 네트워크 주소의 할당 설정과 통신 경로의 설정 데이터입니다.

대상 유니트

라우팅 테이블에 따라 네트워크 주소를 할당할 수 있는 대상 유니트는 다음과 같습니다.

- CPU 고기능 유니트의 네트워크 통신 유니트:
  Controller Link 유니트, SYSMAC LINK 유니트, Ethernet 유니트, EtherNet/IP 유니트(FINS 네트워크 통신으로 사용하는 경우), DeviceNet 유니트, FL-net 유니트
- CJ2H-CPU6□-EIP, CJ2M-CPU3□ 내장 EtherNet/IP 포트(FINS 네트워크 통신으로 사용하는 경우)
- 시리얼 커뮤니케이션 유니트의 시리얼 포트
  시리얼 게이트웨이 기능을 사용하며 네트워크의 계층을 벗어나 통신을 수행하는 경우

또한 고기능 I/O 유니트의 네트워크 통신 유니트(예: CompoNet 마스터 유니트)는 라우팅 테이블의 대상 유니트가 아닙니다.

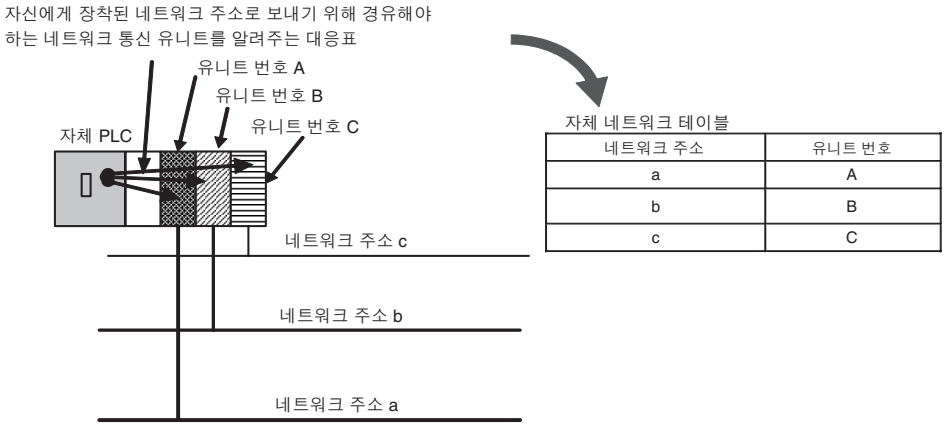
라우팅 테이블을 구성하는 테이블

라우팅 테이블은 다음 2가지 테이블로 구성됩니다.

● 자체 네트워크 테이블

장착되어 있는 네트워크 통신 유니트(CPU 고기능 유니트)에 네트워크 주소를 1~127 범위에서 설정합니다.

PLC에 1대 이상의 네트워크 통신 유니트(CPU 고기능 유니트)를 장착한 경우, CPU 유니트가 장착되어 있는 네트워크 통신 유니트 및 통신 네트워크를 식별하기 위한 테이블입니다.



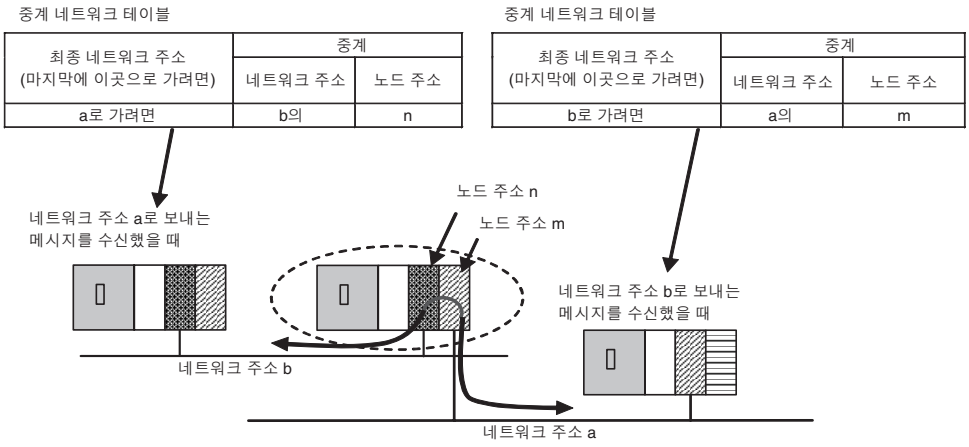
FINS 통신의 경우, 자체 PLC에 장착된 네트워크 통신 유니트(CPU 고기능 유니트)의 유니트 번호와 해당 유니트가 연결되어 있는 FINS 네트워크 주소의 대응표입니다.

● 중계 네트워크 테이블

네트워크의 계층을 벗어나 통신하기 위해 설정합니다.

수신한 메시지를 최종 목적지로 보낼 때까지의 중계 방법을 기재한 테이블입니다.

FINS 통신의 경우, 자체 PLC에는 연결되어 있지 않지만 목표로 하는 네트워크(최종 네트워크)와 거기에 이르는 경로의 최초 중계점(먼저 가야 할 곳)의 네트워크 주소 및 노드 주소의 대응표입니다. 이 중계점을 찾아가는 방식으로 최종 네트워크에 도달하게 됩니다.



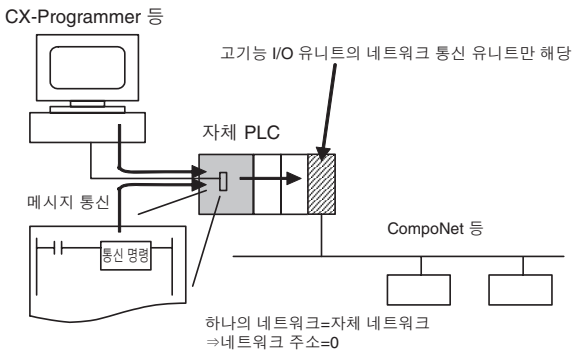
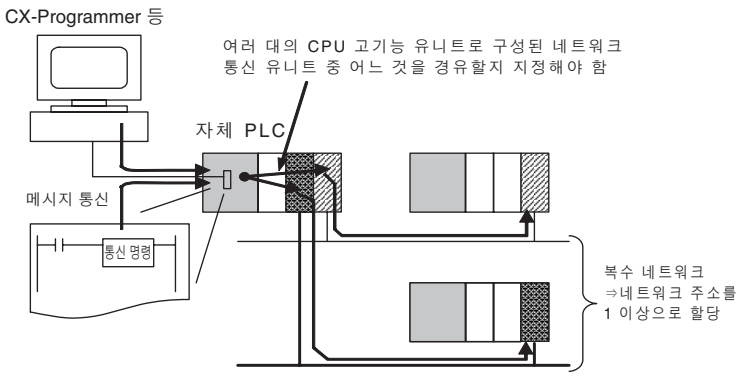
4-4-2 라우팅 테이블이 필요한 경우

라우팅 테이블(자체 네트워크 테이블/중계 네트워크 테이블)은 각각 필요에 따라 PLC에 설정됩니다.

사용상의 주의 사항

자체 네트워크 테이블이 필요한 조건
중계하지 않아도 1대 이상의 네트워크 통신 유니트(CPU 고기능 유니트)를 장착한 경우 네트워크 주소를 할당해야 하므로 자체 네트워크 테이블이 반드시 필요합니다.

네트워크 사용 상태에 따라 라우팅 테이블의 필요 여부는 다음과 같이 달라집니다.

네트워크 사용 상태	시스템 구성	라우팅 테이블	
		자체네트워크 테이블	중계 네트워크
PLC에는 네트워크 통신 유니트를 장착하지 않거나 고기능 I/O 유니트의 네트워크 통신 유니트(CompoNet 등)만 사용 (*1)		필요 없음	필요 없음
PLC에 복수의 통신 유니트가 장착됨		모든 PLC에 필요 (*2)	네트워크 계층을 벗어나 통신하는 경우, 필요 (*3)

*1 주변 툴 등에서 네트워크의 노드에 액세스하는 경우 상대방 네트워크 주소에 0을 설정하십시오.

*2 자체 네트워크 테이블이 없는 상태에서 주변 툴 또는 래더 프로그램의 명령을 통해 네트워크 주소＝0으로 액세스하면 FINS 통신의 경우 유니트 번호가 가장 작은 네트워크 통신 유니트의 네트워크에 자동으로 액세스합니다. 이 점을 이용하여 자체 네트워크 테이블을 설정하지 않고 액세스하려는 네트워크에 대해 네트워크 통신 유니트의 유니트 번호를 가장 작은 번호로 설정하여 해당 네트워크에 액세스할 수도 있습니다.

*3 네트워크 계층을 벗어나 통신하지 않는 경우에는 중계 네트워크 테이블을 설정할 필요가 없습니다.

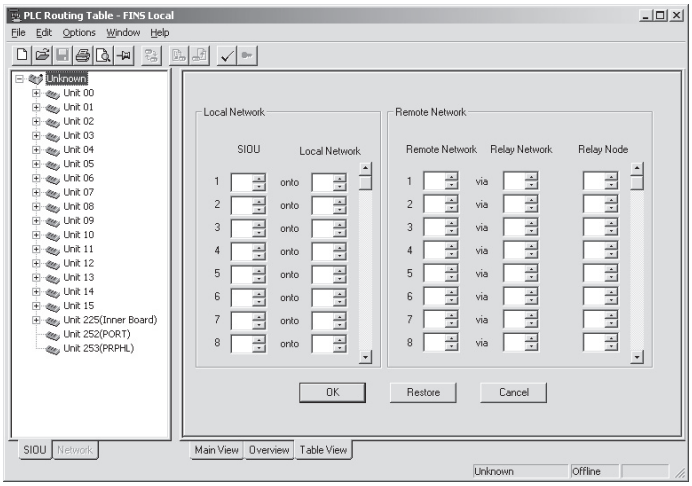
4-4-3 라우팅 테이블의 설정/전송 방법

1 **CX-Integrator를 기동합니다.**

2 **[도구] 메뉴에서 [라우팅 테이블 선택]을 선택합니다.**

3 **[FINS 로컬]을 선택합니다.**

4 **[테이블 보기] 탭에서 라우팅 테이블을 편집합니다.**



5 **CX-Integrator를 온라인으로 연결하여 [옵션] 메뉴에서 [PLC로 전송]을 선택하고 라우팅 테이블을 CPU 유니트로 전송합니다.**

6 **[파일] 메뉴에서 [로컬 라우팅 테이블 파일 저장]을 선택합니다.**

사용상의 주의 사항

라우팅 테이블 데이터의 파일
라우팅 테이블은 CX-Integrator로 작성하는 개별 파일(.rtg)입니다.
CX-Programmer의 프로젝트 파일(.cxp)에는 포함되지 않으므로 주의하십시오.

4-5 고기능 유니트용 할당 DM 영역 설정

고기능 I/O 유니트 또는 CPU 고기능 유니트를 사용하는 경우 설정합니다.
고기능 유니트용 할당 DM 영역 설정은 CX-Programmer에서 조작합니다.

4-5-1 고기능 유니트용 할당 DM 영역 설정

고기능 유니트(고기능 I/O 유니트 및 CPU 고기능 유니트)의 초기 설정은 CPU 유니트의 I/O 메모리에서 할당 DM 영역에 설정합니다.
할당 DM 영역은 전원을 켰을 때 CPU 유니트에서 각 고기능 유니트로 전송됩니다.
할당 DM 영역은 다음과 같은 위치에 할당됩니다.

- **고기능 I/O 유니트의 초기 설정**
  D20000~29599에 유니트 전면의 로터리 스위치로 설정한 기기 번호에 따라 1개 유니트당 100CH가 할당됩니다.

- **CPU 고기능 유니트의 초기 설정**
  D30000~31599에 유니트 전면의 로터리 스위치로 설정한 유니트 번호에 따라 1개 유니트당 100CH가 할당됩니다.

위와 같은 할당 DM 영역은 고기능 유니트별 설정 내용 및 영역별 데이터 전송 타이밍이 다르므로 자세한 내용은 각 고기능 유니트의 사용 설명서를 참조하십시오.

참고

고기능 유니트용 초기 설정을 저장하는 영역에는 그 밖에도 Ethernet 유니트, Controller Link 유니트 등의 특정 유니트용 CPU 고기능 유니트 시스템 설정 영역과 각 고기능 유니트에 저장된 데이터/프로그램(각 고기능 유니트용 주변 툴로 작성)이 있습니다.

4-5-2 고기능 유니트용 할당 DM 영역 설정 방법

다음과 같은 방법으로 설정합니다.

- CX-Programmer의 I/O 테이블 창에서 각 유니트를 선택하여 고기능 유니트를 설정한 후 전송하는 방법
- CX-Programmer의 PLC 메모리 창에서 할당 DM 영역의 각 주소에 데이터를 설정하고 전송하는 방법
- 사용자 프로그램에서 MOV 명령 등을 사용하여 할당 DM 영역의 각 주소에 데이터를 설정하는 방법

자세한 내용은 '8-2 고기능 유니트 설정'을 참조하십시오.

4-6 CPU 고기능 유니트 시스템 설정 영역

4-6-1 CPU 고기능 유니트 시스템 설정 영역

CPU 유니트에 장착된 특정 CPU 고기능 유니트의 설정 정보를 CPU 유니트가 대신 저장하는 영역입니다.
이 영역을 사용하는 특정 CPU 고기능 유니트 및 설정 내용은 다음의 3종류입니다.

- Ethernet 유니트 : Ethernet 설정
- Controller Link 유니트 : 데이터 링크 테이블(임의 설정 사용 시)
- FL-net 유니트 : FL-net 설정

4-6-2 CPU 고기능 유니트 시스템 설정 영역의 설정 방법

각각 다음과 같은 주변 툴로 설정합니다.

데이터	주변 툴
Ethernet 유니트 설정	CX-Programmer의 I/O 테이블에 고기능 유니트 설정 또는 HTML 설정으로 설정
Controller Link 유니트 데이터 링크 테이블	CX-Integrator의 데이터 링크 테이블 설정으로 설정
FL-net 유니트 설정	FL-net 설정 툴 CX-FLnet으로 설정

CPU 고기능 유니트 시스템 설정 영역의 점유 용량

'CPU 고기능 유니트 시스템 설정 영역'의 메모리 제한 용량은 10752바이트입니다.
각 CPU 고기능 유니트의 조합에 따라 'CPU 고기능 유니트 시스템 설정 영역'의 점유 용량이 제한 용량 이내가 되도록 시스템을 설계하십시오.
제한 용량을 초과하면 일부 유니트에서 설정이 기본값 상태에서만 동작하게 되거나 기동되지 않는 경우가 있습니다.

● 'CPU 고기능 유니트 시스템 설정 영역'의 점유 용량 목록

유니트 이름	형식	점유 용량(바이트)
Controller Link 유니트	CJ1W-CLK21-V1	512
Ethernet 유니트	CJ1W-ETN11	412
	CJ1W-ETN21	994
FL-net 유니트	CJ1W-FLN22	988
EtherNet/IP 유니트	CJ1W-EIP21	0(사용 안 함)
시리얼 커뮤니케이션 유니트	CJ1W-SCU21/31/41-V1 CJ1W-SCU22/32/42	
DeviceNet 유니트	CJ1W-DRM21	
위치 제어 유니트	CJ1W-NCF71	
모션 컨트롤 유니트	CJ1W-MCH71	
저장소 및 프로세싱 유니트	CJ1W-SPU01-V2	

점유 용량이 0바이트인 유니트는 CPU 고기능 유니트 시스템 설정 영역을 사용하지 않습니다.

5

프로그램 지식

이 장에서는 프로그램의 종류나 프로그램을 구성하는 변수 또는 명령 등, 프로그램에 대해 자세히 설명합니다.

5-1 프로그램 5-3
5-1-1 프로그램 언어 5-3
5-1-2 래더 언어 5-6
5-1-3 ST 언어 5-8
5-1-4 SFC 언어 5-9

5-2 태스크 5-11
5-2-1 태스크 5-11
5-2-2 사이클릭 태스크 5-14
5-2-3 인터럽트 태스크 5-19
5-2-4 태스크 설계 방법 5-30

5-3 섹션 5-40
5-3-1 섹션이란? 5-40

5-4 펑션 블록(FB) 기능 5-42
5-4-1 펑션 블록이란? 5-42
5-4-2 펑션 블록의 특징 5-43
5-4-3 펑션 블록 관련 사양 5-44

5-5 변수 5-47
5-5-1 변수란? 5-47
5-5-2 변수의 종류 5-48
5-5-3 글로벌 변수 5-50
5-5-4 로컬 변수 5-50
5-5-5 네트워크 변수(CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당) 5-51
5-5-6 펑션 블록 내 변수 5-55
5-5-7 변수의 데이터 형식 5-56
5-5-8 변수의 주소 자동 할당 5-61

5-6 명령 5-62
5-6-1 명령의 기본 지식 5-62
5-6-2 피연산자 지정 방법 5-70
5-6-3 데이터 형식 5-77
5-6-4 I/O 리프레시 타이밍 5-80

5-7 인덱스 레지스터 기능 5-85
5-7-1 인덱스 레지스터란? 5-85
5-7-2 인덱스 레지스터 사용 방법 5-86
5-7-3 인덱스 레지스터 관련 처리 5-89
5-7-4 인덱스 레지스터를 모니터링하는 방법 5-90
5-7-5 인덱스 레지스터/데이터 레지스터의 태스크 간 공통화 5-91

5-8 주소의 오프셋 지정 5-92
5-8-1 주소의 오프셋 지정이란? 5-92
5-8-2 주소의 오프셋 용도의 예 5-95
5-9 작성한 프로그램 확인 5-96
5-9-1 CX-Programmer의 입력 조작 시 등의 확인 5-96
5-9-2 CX-Programmer에서 프로그램 확인 5-96
5-9-3 시뮬레이터 기능을 사용한 디버그 5-97
5-9-4 명령 실행 시 확인 5-100
5-10 프로그램 작성 시 주의 사항 5-103
5-10-1 조건 플래그 사용 시 주의 사항 5-103
5-10-2 특정 프로그램 영역에 관한 주의 사항 5-109

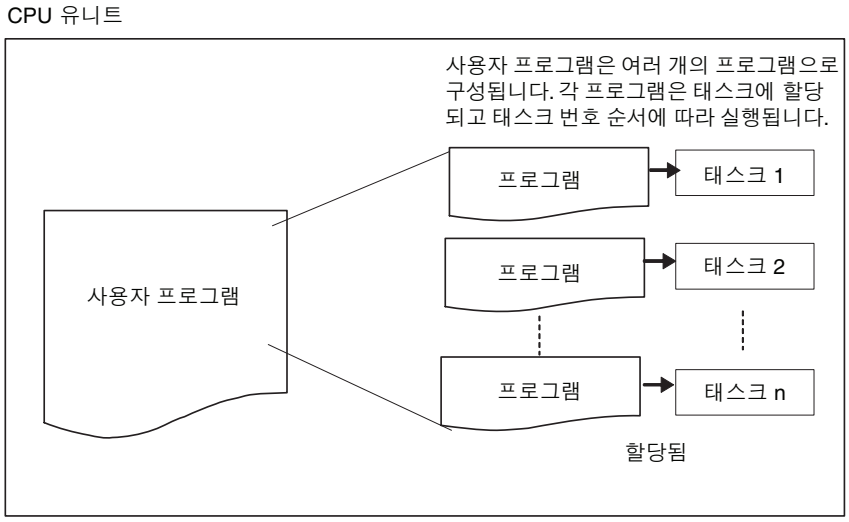
5-1 프로그램

5-1-1 프로그램 언어

사용자 프로그램의 구조

사용자 프로그램은 CX-Programmer로 작성하며 다음과 같이 구성됩니다.

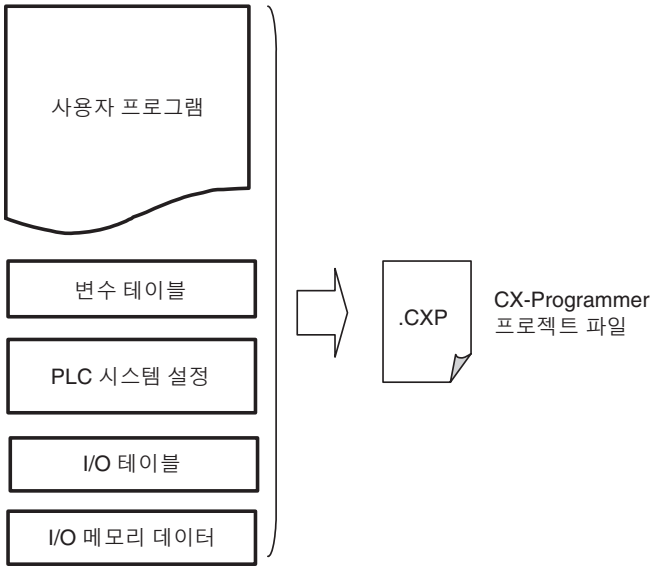
- 프로그램
  전체 사용자 프로그램은 여러 개의 프로그램으로 구성됩니다.
  하나의 프로그램은 END 명령으로 종료됩니다.
- 태스크(실행 단위)
  하나의 프로그램은 각각 태스크로 할당되어 실행됩니다(CX-Programmer에서는 프로그램 속성으로 지정).
  태스크에는 평상시 주기적으로 처리하는 사이클릭 태스크와 인터럽트 조건이 성립할 때 실행하는 인터럽트 태스크, 일정 시간 간격으로 실행하는 예약 인터럽트 태스크, 전원 OFF 시 실행하는 전원 OFF 태스크가 있습니다.



- 섹션
  CX-Programmer에서는 하나의 프로그램을 임의의 개수로 분할하여 작성/표시할 수 있는데, 그 분할 단위를 '섹션'이라고 합니다. 주로 프로그램을 알기 쉽게 하기 위해 분할합니다.
- 서브루틴, 펑션 블록
  하나의 프로그램에 서브루틴 프로그램, 펑션 블록을 기술할 수 있습니다.

사용자 프로그램의 데이터

전체 사용자 프로그램은 CX-Programmer의 프로젝트 파일(.CXP)에 변수 테이블, PLC 시스템 설정, I/O 테이블, I/O 메모리 데이터 등의 다른 설정 데이터와 함께 저장됩니다.

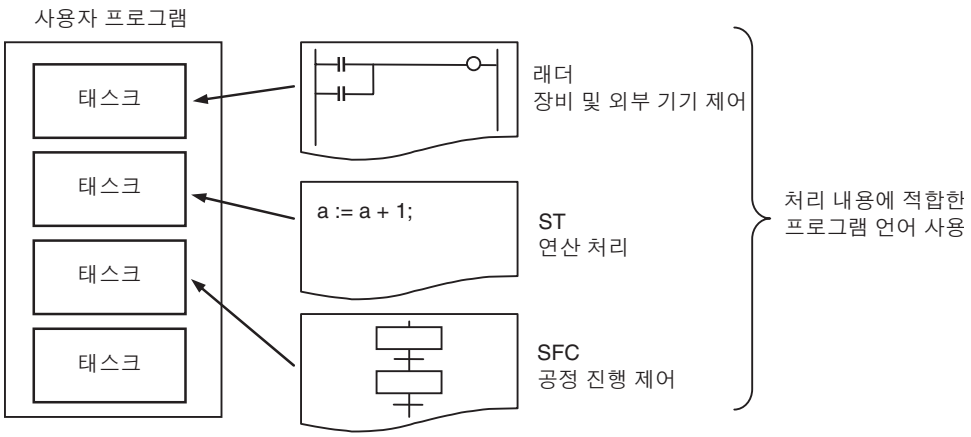


프로그램 언어의 종류

각 프로그램은 다음과 같은 3종류의 언어를 사용하여 임의로 기술할 수 있습니다.

- 래더 언어
- ST(구조적 텍스트) 언어
- SFC(Sequential Function Chart) 언어

실행 단위인 태스크에 래더 언어, ST 언어, SFC 언어로 기술된 각 프로그램을 할당합니다.



이러한 언어를 구분해서 사용하면 장치 제어부를 래더로 기술하고 연산 처리부는 ST로 기술할 수 있게 되며, 처리에 맞춰 최적의 언어를 사용할 수 있습니다.

각 언어는 다음과 같은 단위로 기술할 수 있습니다.

○: 기술 가능, ×: 기술 불가능

기술 가능한 단위 / 언어	태스크 단위의 기술	SFC 차트 내의 작업 프로그램, 트랜지션 프로그램	펑션 블록 정의 내 알고리즘
래더 언어	○	○	○
ST 언어	○	○	○
SFC 언어	○	×	×

또한 각 언어의 주소 표현은 다음과 같습니다.

○: 기술 가능 △: 일부 가능

주소 표현 / 언어	실제 주소	변수명
래더 언어	○	○
ST 언어	△(*1)	○
SFC 언어	△(*2)	△(*2)

*1 변수에 대해 임의의 주소를 할당하여 대체할 수 있습니다.

*2 SFC 차트 내의 논리형(Boolean) 작업/트랜지션은 가능합니다.

프로그램 용량

CJ2 CPU 유니트의 사용자 프로그램(전체 태스크의 프로그램)의 최대 프로그램 용량 합계는 다음과 같습니다. 합계 스텝 수는 최대 프로그램 용량 이하이어야 합니다.

형식	최대 프로그램 용량	입출력 점수
CJ2H-CPU68(-EIP)	400K 스텝	2560
CJ2H-CPU67(-EIP)	250K 스텝	
CJ2H-CPU66(-EIP)	150K 스텝	
CJ2H-CPU65(-EIP)	100K 스텝	
CJ2H-CPU64(-EIP)	50K 스텝	
CJ2M-CPU□5	60K 스텝	
CJ2M-CPU□4	30K 스텝	
CJ2M-CPU□3	20K 스텝	
CJ2M-CPU□2	10K 스텝	
CJ2M-CPU□1	5K 스텝	

프로그램 용량은 CX-Programmer의 [보기] − [메모리 보기]에서 확인할 수 있습니다.

래더 명령의 용량은 명령과 피연산자의 종류에 따라 다릅니다. 자세한 내용은 '부록-2 명령 처리 시간/스텝 수 목록'을 참조하십시오.

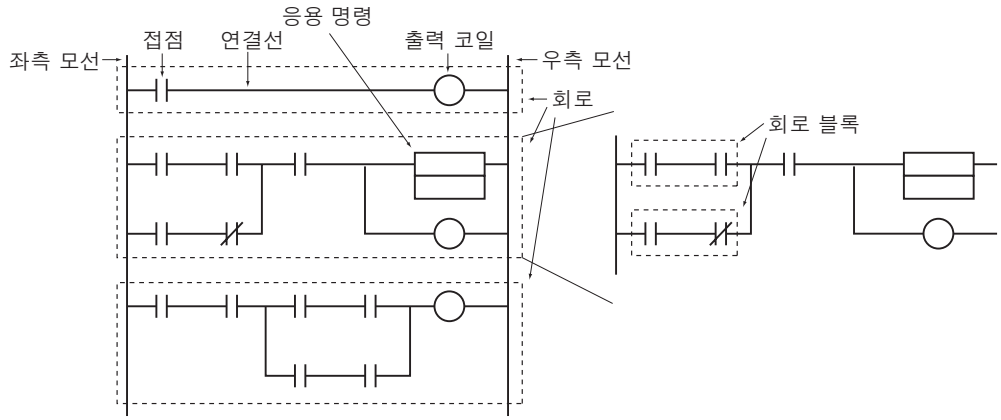
펑션 블록 사용 시의 소비 용량은 '5-4-3 펑션 블록 관련 사양'을 참조하십시오.

5-1-2 래더 언어

래더 언어는 전기 회로의 이미지로 기술한 PLC 기본 언어입니다.
메모리에 저장된 명령어의 순서(니모닉 순서)에 따라 각 명령이 실행되므로 프로그램의 개념, 순서를 정확하게 이해해야 합니다.

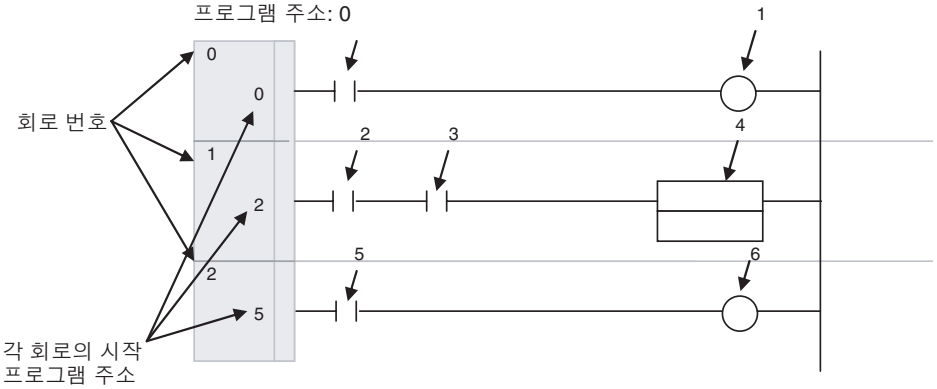
● 래더도의 구성 요소

래더도는 좌우 모선, 연결선, 접점, 출력 코일, 응용 명령으로 이루어집니다.
프로그램은 복수의 회로로 구성됩니다. 회로는 모선을 잘랐을 때 분할할 수 있는 단위(니모닉에서는 LD/LDNOT 명령~LD/LDNOT 명령의 앞에 있는 출력계 명령까지)입니다.
또한 회로는 (논리 시작을 나타내는) LD/LDNOT 명령을 선두로 한 회로 블록으로 구성됩니다.



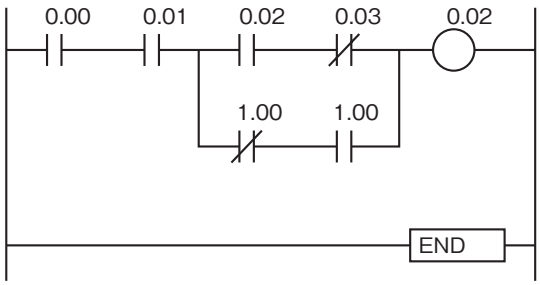
참고

CX-Programmer에서는 각 회로의 좌측에 다음과 같이 회로 번호와 해당 회로의 시작 프로그램 주소가 표시됩니다.



● 니모닉

니모닉은 래더도를 명령어로 기술한 일련의 프로그램입니다.
프로그램 주소에서 1개 프로그램 주소＝1개 명령어를 나타냅니다. 프로그램 주소는 0부터 시작합니다.
예)



프로그램 주소	명령어 (니모닉)	피연산자
0	LD	0.00
1	AND	0.01
2	LD	0.02
3	ANDNOT	0.03
4	LDNOT	1.00
5	AND	1.01
6	OR LD	—
7	AND LD	—
8	OUT	2.00
9	END	—

5-1-3 ST 언어

ST(Structured Text) 언어는 IEC 61131-3에서 규정하는 공업 제어(주로 PLC)용 고급 언어입니다.
준비된 제어문, 연산자, 함수를 사용하여 래더로 기술하기 어려운 수치 연산 처리를 부분적으로 처리하는데 적합합니다(래더 언어로 기술되는 모든 처리를 기술할 수는 없음).
IEC-61131-3에 따라 ST 언어를 지원합니다.



참고

ST 프로그램에 대한 자세한 사양, 기술 내용, 입력 방법 등은 "CX-Programmer 조작 설명서 펑션 블록편/구조적 텍스트편"(SBCA-338)을 참조하십시오.

● ST 프로그램의 특징

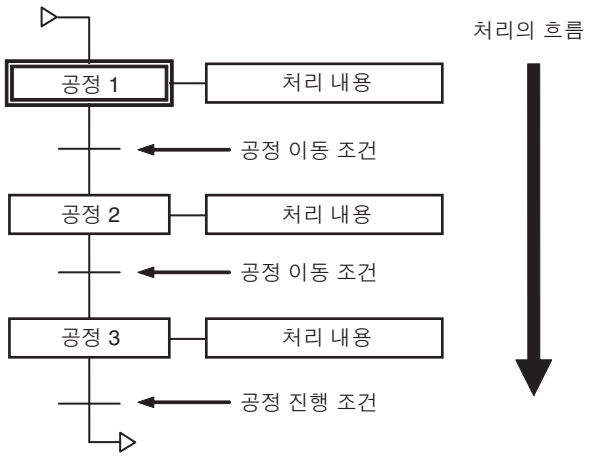
- 루프 문이나 IF THEN ELSE 등의 제어문, 사칙연산, 비교, AND/OR 등의 연산자, 절대값이나 삼각함수 등의 수치 연산 함수, 문자열 추출이나 결합 등의 문자열 함수, 메모리 카드 처리 함수, 문자열 송수신 함수, 각도 변환 함수 등이 있습니다.
- C 언어 등과 같은 고급 언어와 동일한 기술 또는 한자 등의 문자를 포함한 코멘트를 통해 읽기 편한 프로그램을 작성할 수 있습니다.
- ST 프로그램은 일반 프로그램과 동일하게 다운로드/업로드할 수 있습니다. 단, ST 프로그램의 태스크를 태스크 단위로 다운로드/업로드할 수는 없습니다.
- ST 프로그램에서 펑션 블록(래더 언어 또는 ST 언어)을 호출할 수 있습니다.
- 배열형(1차원 배열) 변수를 지원하기 때문에 활용 상황에 따른 데이터 처리를 간단하게 수행할 수 있습니다.

● 사양

항목	내용
ST 프로그램의 단위	• 태스크 단위 • 펑션 블록 정의의 알고리즘 • SFC 차트 내의 작업 프로그램, 트랜지션 프로그램
주소 지정 방법	변수명으로 지정. 변수에 대해 임의의 주소 할당 가능
강제 세트/리셋	가능
온라인 편집	ST 프로그램 편집

5-1-4 SFC 언어

SFC(Sequential Function Chart) 언어는 주로 시퀀스 처리를 제어할 수 있도록 공정 진행 프로그램을 간단하게 기술하기 위해 개발된 그래픽 프로그래밍 언어입니다.
SFC는 공정의 흐름을 그림으로 표시하고 공정 진행 조건이나 각 공정의 처리 내용을 기술하여 시퀀스 처리 제어를 프로그래밍할 수 있습니다.



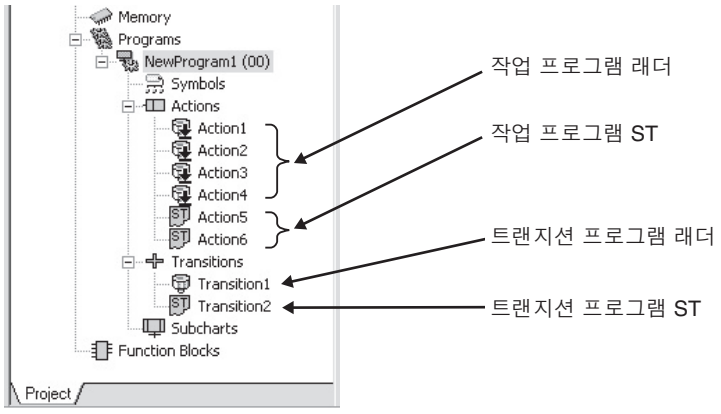
참고

SFC 프로그램의 자세한 사양, 기술 내용, 입력 방법 등은 "CX-Programmer 조작 설명서 SFC 편"(SBCA-348)을 참조하십시오.

- **SFC 프로그램의 특징**

  SFC 프로그램의 일부로 래더/ST 사용 가능

  SFC 프로그램 내 스텝의 작업 및 트랜지션의 이동 조건을 래더 또는 ST를 사용하여 기술할 수 있습니다.



● SFC 차트와 작업 프로그램/트랜지션 프로그램을 동시에 표시/편집 가능 온라인 편집/디버그 가능

온라인 상태에서 SFC 차트를 편집할 수 있습니다. 또한 디버그 시에 작업 블록을 표시하지 않고 진행 상황을 확인할 수 있습니다. 이때 SFC 편집기에 작업 블록이 표시되지 않아도 프로그램 보기에서 작업 프로그램을 확인할 수 있습니다.

● 사양

항목	내용
SFC 프로그램의 단위	태스크 단위(1개 태스크＝1개 SFC 차트)
SFC 프로그램을 할당할 수 있는 태스크	사이클릭 태스크 또는 추가 태스크로 가능
SFC 요소(*)	스텝, 트랜지션, 작업, 점프, 서브 차트
주소 지정 방법	SFC 차트의 논리형 작업/트랜지션: 변수명 또는 실제 주소로 지정 단, ST 프로그램으로 작성한 경우 실제 주소 지정은 불가능
강제 세트/리셋	온라인으로 트랜지션을 강제 세트/리셋하고 1스텝씩 실행 가능
온라인 편집	SFC 차트 편집, 작업 추가/삭제, 트랜지션 추가/삭제 가능

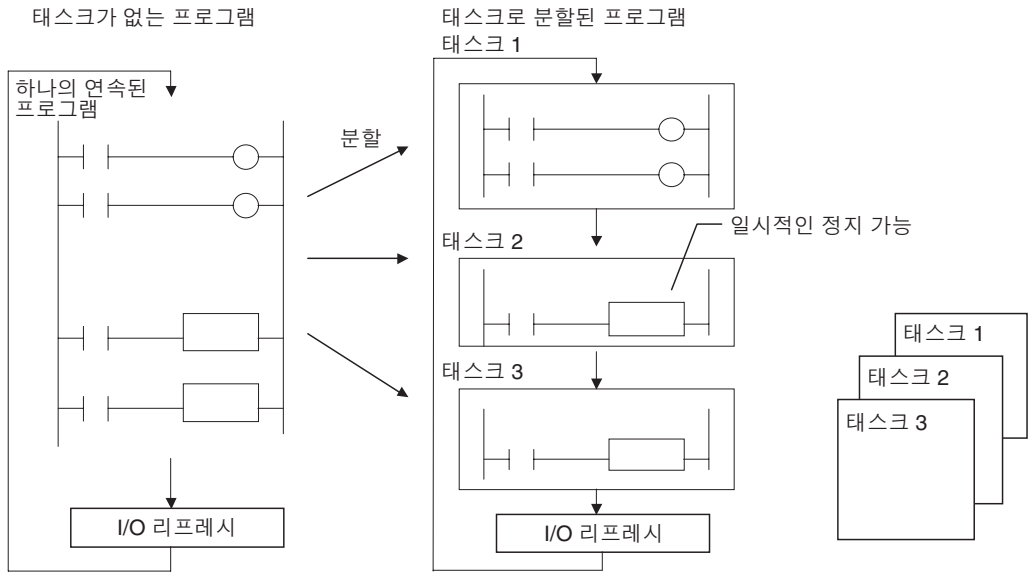
* SFC 요소는 로컬 변수로 자동 등록됩니다.

5-2 태스크

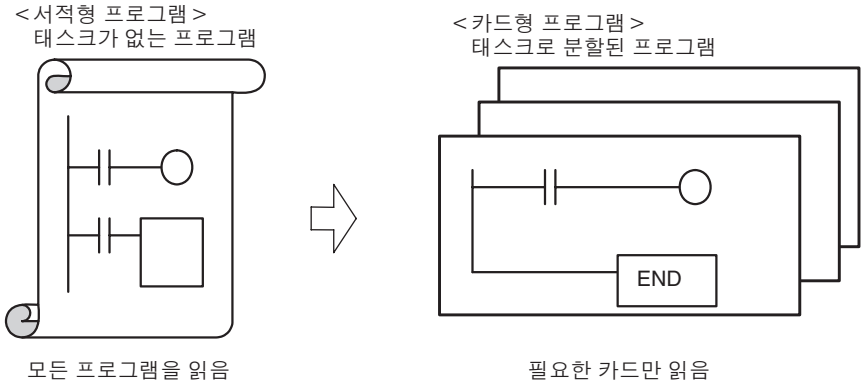
5-2-1 태스크

태스크의 개념

프로그램을 큰 단위로 분할하여 개별 동작을 어떤 순서로 실행할지 지정하는 기능이 태스크입니다.
태스크는 각 태스크 간에 실행 및 정지를 제어할 수 있습니다.
몇 가지 프로그램 부품을 태스크로 조립해 두고 품종이나 공정 등에 맞춰 일정한 태스크만 실행할 수 있습니다.
또한 필요할 때 필요한 태스크만 실행시킬 수 있으므로 성능(사이클 시간) 향상에도 도움이 됩니다.



태스크가 없는 프로그램이 책을 처음부터 모두 읽는 것이라면 태스크가 있는 프로그램은 각 카드를 개별적으로 읽는 것이라고 할 수 있습니다.
각 카드는 서로 실행/정지를 지시할 수 있으며, 정지된 카드는 건너뛰고 읽습니다.

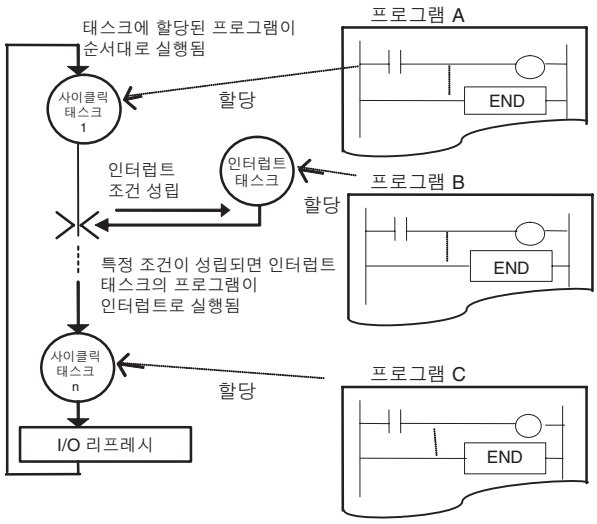


태스크에는 다음과 같이 2종류가 있습니다.

태스크 유형	내용	대상 프로그램 언어	실행 조건
사이클릭 태스크	1사이클에 1회 실행	래더 언어 SFC 언어 ST 언어	다음과 같은 방법으로 태스크를 실행 가능한 상태로 설정하면 실행됩니다. • CX-Programmer를 사용하여 '운전 시작 시 기동'으로 설정 (프로그램의 속성 설정) • 태스크 제어 명령에 의해 기동
인터럽트 태스크(*1, *2)	특정 요인이 발생하면 실행 중인 처리를 중단시키고 인터럽트 실행	래더 언어 ST 언어	다음과 같은 인터럽트 태스크 유형별로 설정된 인터럽트 요인이 발생하면 실행 가능 상태가 됩니다. • 전원 OFF 인터럽트 태스크 • 예약 인터럽트 태스크 • I/O 인터럽트 태스크 • 외부 인터럽트 태스크 • 내장 입력 인터럽트 태스크(CJ2M CPU 유니트만 해당)

*1 태스크 제어 명령으로 인터럽트 태스크를 기동하면 사이클릭 태스크와 동일하게 주기적으로(1사이클에 1회) 실행시킬 수 있습니다('추가 태스크'라고 함).

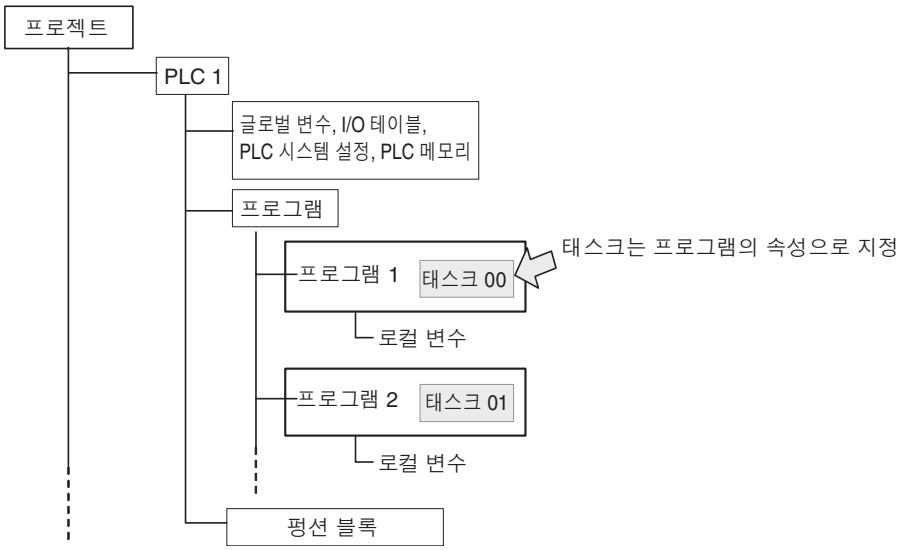
*2 인터럽트 태스크에서는 SFC 프로그램을 사용하지 마십시오. 정상적으로 동작하지 않습니다.
단, 추가 태스크로 설정한 경우는 사용할 수 있습니다.



태스크에 대한 CX-Programmer의 조작

태스크는 CX-Programmer에서 프로그램의 속성으로 설정합니다.

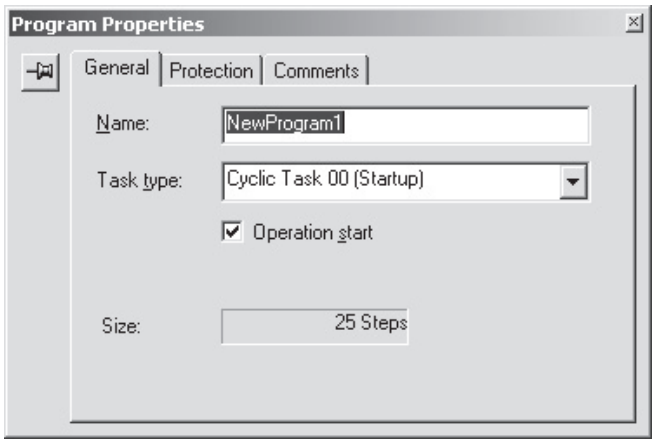
● 태스크 종류 설정



프로그램에서 태스크를 할당하는 방법은 다음과 같습니다.

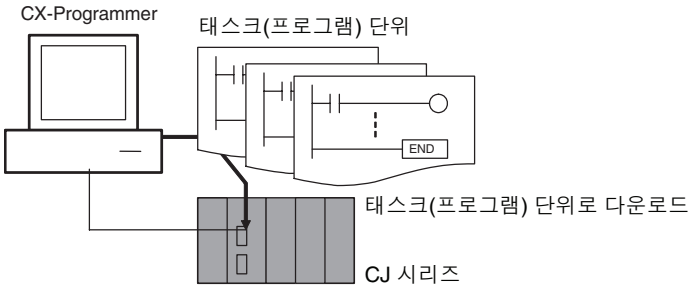
1 **태스크를 할당하는 프로그램을 활성화한 상태에서 [보기] - [속성]을 선택하거나 마우스 오른쪽 단추를 클릭하여 팝업 메뉴에서 [속성]을 선택합니다.**

2 **[일반] 탭을 선택하고 [태스크 종류](태스크 종류와 태스크 번호)를 선택합니다.**
사이클릭 태스크의 경우, 필요에 따라 [작업 시작] 확인란을 선택합니다.



- **태스크 다운로드/업로드**

CX-Programmer에서 태스크 단위로 프로그램을 다운로드/업로드할 수 있습니다.



- **펑션 블록 사용 시의 제한**

펑션 블록을 포함하는 프로그램의 경우, 펑션 블록이 포함된 태스크를 태스크 단위로 다운로드할 수 없습니다(업로드는 가능).

- **SFC 프로그램, ST 프로그램 사용 시의 제한**

SFC 프로그램이나 ST 프로그램을 할당한 태스크는 태스크 단위로 다운로드/업로드할 수 없습니다.

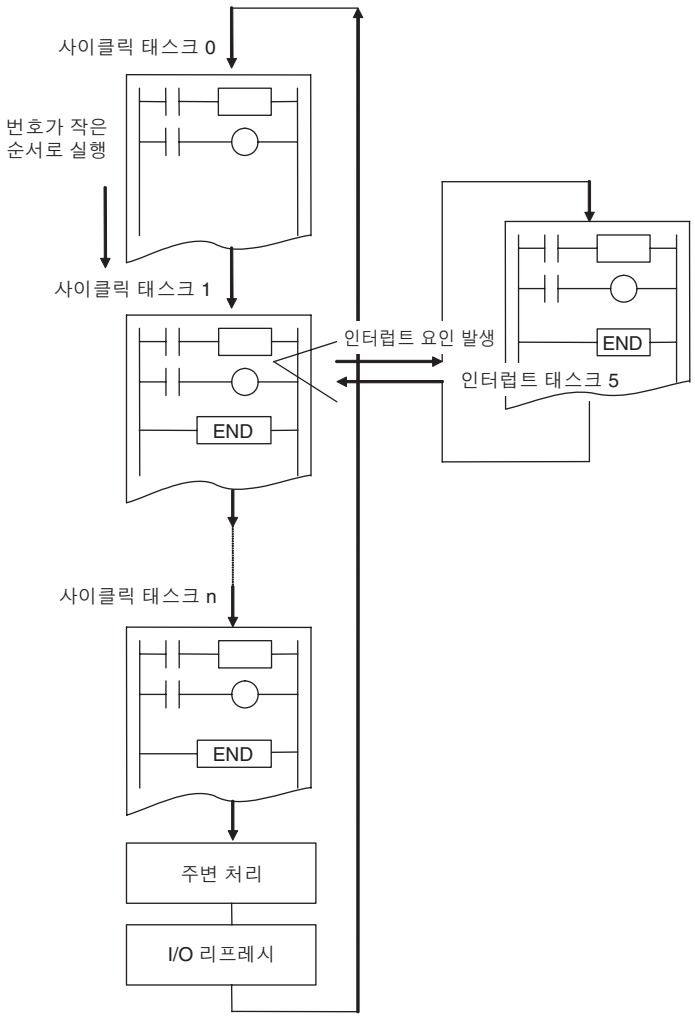
5-2-2 사이클릭 태스크

사이클릭 태스크

1사이클에 1회, 태스크 번호가 작은 순서로 순차적으로 실행됩니다.
최대 128개 태스크가 가능합니다(사이클릭 태스크 번호: 0~127).
CX-Programmer에서 프로그램 속성을 [운전 시작 시 기동]으로 설정하거나 태스크 제어 명령으로 시작할 수 있습니다.
태스크 제어 명령에 대해서는 '부록-1 명령 목록'을 참조하십시오.
추가 태스크를 사용하면 최대 384개 태스크를 처리할 수 있습니다.

사용상의 주의 사항

SFC 프로그램 태스크의 기동/종료에는 태스크 제어 명령의 TKON/TKOF 명령을 사용하지 마십시오. 정상적으로 동작하지 않습니다.



참고

태스크 실행 선두에서 모든 조건 플래그(ER 플래그, CY 플래그, = 플래그, AER 플래그 등)가 클리어됩니다. 태스크 실행 중에는 조건 플래그를 읽을 수 없습니다.
또한 인터록(IL/ILC 명령 등), 점프(JMP/CJP/JME 명령 등) 및 서브루틴(SBS/RET/SBN 명령 등)은 한 태스크 내에서 닫아야 합니다.
예를 들어, 태스크를 벗어나 점프할 수 없습니다. 복수의 태스크에서 공통 서브루틴을 사용하는 경우에는 글로벌 서브루틴(GSBS/GRET/GSBN 명령)을 사용하십시오.

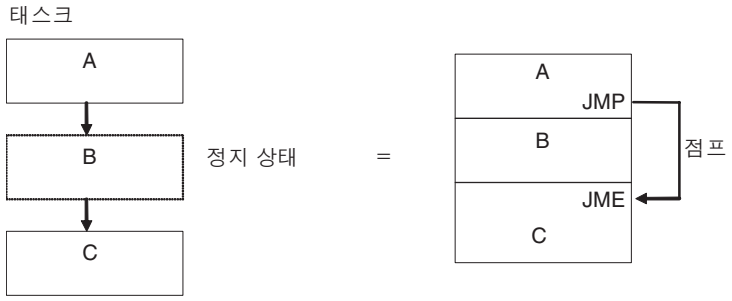
사이클릭 태스크의 상태

● 실행 상태

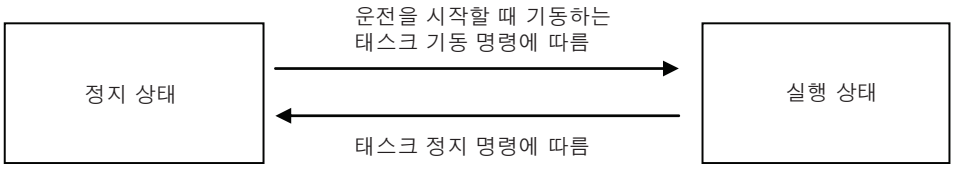
태스크 번호가 작은 순서로 실행합니다.
'프로그램' 모드에서 '운전' 또는 '모니터' 모드로 변경하는 경우 [운전 시작 시 기동]이 선택된 태스크가 실행 상태가 됩니다.
또한 태스크 기동 명령에 따라 정지 상태에서 실행 상태로 이동합니다.
태스크 정지 명령에 따라 자체 태스크를 정지한 경우에는 해당 태스크 정지 명령 이후 실행되지 않습니다.

● 정지 상태

정지 상태의 태스크는 명령을 실행하지 않습니다.
또한 태스크 정지 명령에 따라 실행 상태에서 정지 상태로 이동합니다.
실행 타이밍에 따라 프로그램을 분할하여 태스크 기동/정지를 제어하면 사이클 시간을 단축할 수 있습니다.



● 상태의 이동



참고

정지 상태는 점프(JMP-JME) 명령과 동일한 동작입니다. 정지 상태인 태스크의 출력 상태는 그대로 유지됩니다.

사이클릭 태스크의 사용 방법

● 태스크 기동/정지 명령

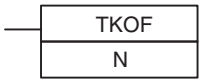
사이클릭 태스크를 프로그램에서 기동/정지하는 경우, 태스크 기동/정지 명령을 사용합니다.
태스크 기동/정지 명령은 기동하는 프로그램 언어에 따라 다음과 같이 달라집니다.

- 래더 프로그램, ST 프로그램 : TKON(태스크 기동 명령), TKOF(태스크 정지 명령)
- SFC 프로그램 : SFCON(태스크 기동 명령), SFCOFF(태스크 정지 명령)

• TKON 명령의 경우



N : 태스크 번호 입력 조건이 ON이면 태스크를 실행 상태로 만듭니다.
사이클릭 태스크인 경우는 동시에 대응하는 태스크 플래그가 ON으로 됩니다.

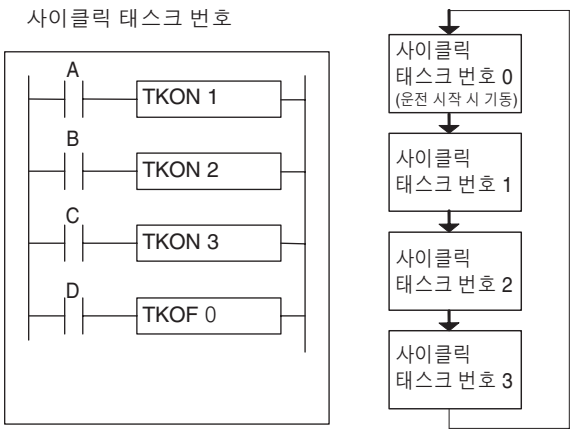


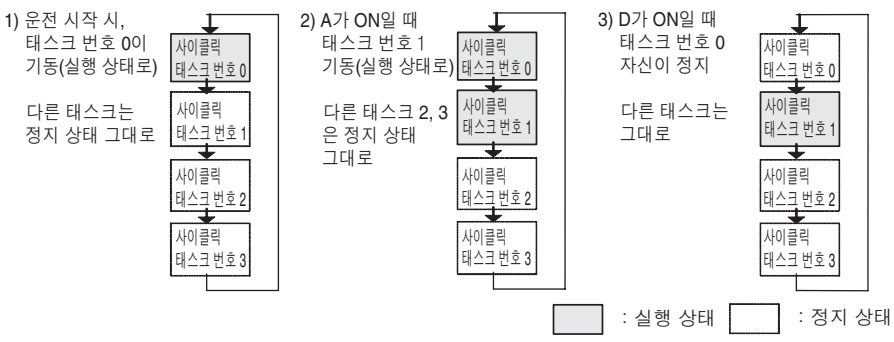
N : 태스크 번호 입력 조건이 ON이면 태스크를 정지합니다.
사이클릭 태스크인 경우는 동시에 대응하는 태스크 플래그가 OFF로 됩니다.

위 명령에 따라 어느 사이클릭 태스크에 대해 어떤 때이든지 실행 가능/정지를 지시할 수 있습니다.
한번 실행 상태가 된 사이클릭 태스크는 다음 사이클에서도 실행 상태 그대로입니다. 또한 한번 정지 상태가 된 사이클릭 태스크는 다음 사이클에도 정지 상태를 유지합니다.
단, 위의 명령은 사이클릭 태스크 내에서만 사용할 수 있으며, 인터럽트 태스크에서는 사용할 수 없습니다.

(주) 사이클에서는 실행 상태인 사이클릭 태스크가 반드시 하나 이상 필요합니다. 실행 상태인 사이클릭 태스크가 하나 이상 존재하지 않으면 태스크 에러 플래그(A295.12)가 ON이 되고 CPU 유니트의 운전은 정지합니다.

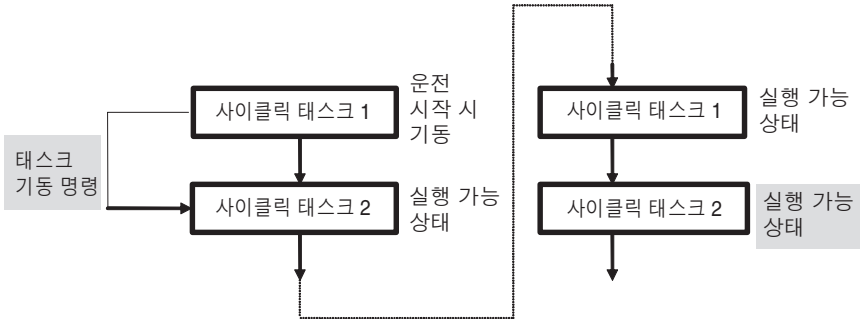
예:



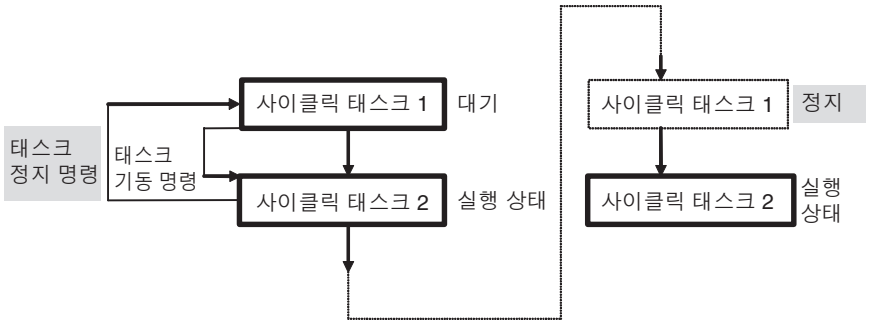


● 사이클 시간과의 관계

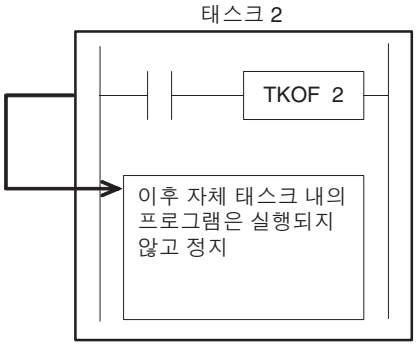
한번 실행 상태가 된 사이클릭 태스크는 다음 사이클에도 실행 상태가 됩니다.



한번 정지 상태가 된 사이클릭 태스크는 다음 사이클에도 정지 상태가 됩니다. 정지 상태에서 실행 상태가 되려면 태스크 기동 명령에 의해 기동되어야 합니다.



자체 태스크에 대해 태스크 정지 명령을 실행한 경우, 해당 시점에서 실행을 정지하고 자체 태스크를 정지 상태로 이동시킵니다.



사이클릭 태스크의 태스크 번호 및 사이클 시간의 관계

- 태스크 m이 태스크 n(m > n)을 기동한 경우 다음 사이클에서 태스크 n이 기동됩니다.
  예: 태스크 5번이 태스크 2번을 기동한 경우 태스크 2번은 다음 사이클에 기동됩니다.
- 태스크 m이 태스크 n(m < n)을 기동한 경우 해당 사이클 내에서 태스크 n이 기동됩니다.
  예: 태스크 2번이 태스크 5번을 기동한 경우 태스크 5번은 해당 사이클에서 기동됩니다.
- 태스크 m이 태스크 n(m > n)을 정지시킨 경우 태스크 n은 다음 사이클에서 정지 상태가 됩니다.
  (해당 사이클에서는 태스크 n이 이미 실행되었습니다.)
  예: 태스크 5번이 태스크 2번을 정지한 경우 태스크 2번은 다음 사이클에 정지됩니다.
- 태스크 m이 태스크 n(m < n)을 정지시킨 경우 태스크 n은 해당 사이클 내에 정지 상태가 됩니다.
  예: 태스크 2번이 태스크 5번을 정지한 경우 태스크 5번은 해당 사이클에서 정지됩니다.

태스크와 I/O 메모리의 관계

- 인덱스 레지스터(IR) 및 데이터 레지스터(DR) 이외에 I/O 메모리는 각 태스크에서 공통으로 사용합니다. 예를 들어, 사이클릭 태스크 1에서 사용하는 접점 10.00과 사이클릭 태스크 2에서 사용하는 접점 10.00은 동일한 접점을 가리킵니다. 인덱스 레지스터(IR) 및 데이터 레지스터(DR) 이외에 I/O 메모리를 태스크 간에 공통으로 사용하는 경우, 한 태스크에서 변경한 값을 다른 태스크에서 사용할 때는 프로그램에서 주의하십시오.
- 인덱스 레지스터(IR)와 데이터 레지스터(DR)에는 2종류의 사용 방법이 있습니다.
  - 각 태스크별로 개별(독립) 사용하는 방법
    이 방법의 경우 사이클릭 태스크 1에서 사용하는 IR0과 사이클릭 태스크 2에서 사용하는 IR0이 다릅니다.
  - 각 태스크에서 공통으로 사용하는 방법
    이 방법의 경우 사이클릭 태스크 1에서 사용하는 IR0과 사이클릭 태스크 2에서 사용하는 IR0이 같습니다.

인덱스 레지스터(IR)와 데이터 레지스터(DR)를 위의 2가지 중 어느 방법으로 사용할지를 설정하려면 CX-Programmer를 사용합니다.

I/O 메모리	태스크와의 관계
CIO, 내부 보조 릴레이, 데이터 메모리 등의 인덱스 레지스터(IR), 데이터 레지스터(DR)를 제외한 모두(*1)	태스크 공통으로 사용
인덱스 레지스터(IR), 데이터 레지스터(DR)만(*2)	설정에 따라 다름

*1 현재 EM 뱅크도 태스크에서 공통으로 사용합니다. 따라서 사이클릭 태스크 1에서 현재 EM 뱅크 번호를 변경한 경우, 사이클릭 태스크 2에서도 변경된 현재 뱅크를 사용할 수 있습니다.

*2 인터럽트 태스크 기동 시, IR 및 DR의 값은 일정하지 않습니다. 따라서 인터럽트 태스크에서 IR 및 DR을 사용하는 경우, 반드시 MOVR/MOVRW(인덱스 레지스터 설정) 명령에 따라 값을 설정한 후 사용하십시오. 인터럽트 태스크 종료 후에는 자동으로 인터럽트 발생 전의 IR 및 DR 값으로 돌아갑니다.

5-2-3 인터럽트 태스크

조건 성립 시(사이클 내 임의의 타이밍) 인터럽트 태스크를 실행할 수 있습니다.
인터럽트 요인이 발생하면 명령 실행 중, I/O 리프레시, 주변 서비스의 사이클 내에서 임의의 타이밍에 실행됩니다.
인터럽트 태스크에는 다음과 같은 종류가 있습니다.

인터럽트 태스크 유형

전원 OFF 인터럽트 태스크
CPU 유니트의 전원이 꺼지기(전원 OFF) 직전에 실행하는 태스크입니다

예약 인터럽트 태스크
정해진 시간마다 가동 상황을 모니터링하는 등, 정시성이 필요한 프로그램에서 사용하는 태스크입니다.

I/O 인터럽트 태스크
CPU 유니트에 연결된 인터럽트 입력 유니트에 입력이 있는 경우 실행하는 태스크입니다.

외부 인터럽트 태스크
고기능 I/O 유니트, CPU 고기능 유니트에서 인터럽트를 요구할 때 실행하는 태스크입니다.

추가 태스크
인터럽트 태스크를 사이클릭 태스크로 취급하는 것입니다. 따라서 다른 인터럽트 태스크처럼 인터럽트 처리가 목적은 아닙니다.
사이클릭 태스크에는 128개 태스크가 있지만 추가 태스크를 사용하면 최대 384개 태스크를 사용할 수 있습니다.

● 인터럽트 태스크 유형 목록

태스크 이름	태스크 번호	실행 조건	관련 설정
전원 OFF 인터럽트 태스크	인터럽트 1	CPU 유니트의 전원 OFF 시	• PLC 시스템 설정의 전원 OFF 인터럽트 유효 설정
예약 인터럽트 태스크 0, 1(*1)	인터럽트 2, 3	CPU 유니트의 내부 타이머에 의해 일정 시간 경과 시마다	• 인터럽트 마스크 세트 명령(MSKS 명령)에 의한 예약 인터럽트 시간 설정(1~9999) • PLC 시스템 설정의 '예약 인터럽트 시간 유니트'(10ms/1.0ms/0.1ms)
I/O 인터럽트 태스크(*2) 0~031	인터럽트 100~131	CPU 장치에 있는 인터럽트 입력 유니트의 입력 접점 기동 시	• 인터럽트 마스크 세트 명령(MSKS 명령)에 의한 지정 접점의 인터럽트 마스크 해제
외부 인터럽트 태스크(*3) 0~255	인터럽트 0~255	CPU 장치의 고기능 I/O 유니트/CPU 고기능 유니트의 사용자 프로그램에 의한 요구 시	없음(항상 유효)
추가 태스크 0~255	인터럽트 0~255	실행 상태(태스크 제어 명령에 의해서만)에서 매 사이클	없음(항상 유효)

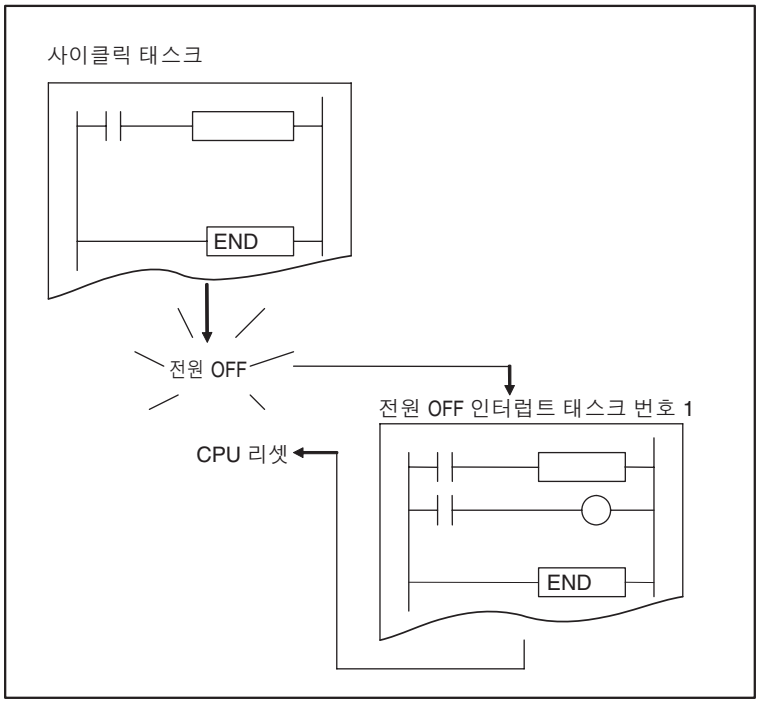
*1 유니트 간 동기 제어 기능을 사용할 때는 예약 인터럽트 태스크를 사용할 수 없습니다.
*2 인터럽트 입력 유니트는 CPU 장치에 장착하십시오. 그 이외의 위치에 장착한 경우, CPU 유니트에 대해 I/O 인터럽트 태스크의 실행을 요구할 수 없습니다.
*3 고기능 I/O 유니트, CPU 고기능 유니트에서 보낸 인터럽트 요구에 따라 외부 인터럽트 태스크를 실행하는 경우, 고기능 I/O 유니트, CPU 고기능 유니트는 CPU 장치에 장착하십시오. 그 이외의 위치에 장착한 경우, CPU 유니트에 대해 외부 인터럽트 태스크의 실행을 요구할 수 없습니다.

사용상의 주의 사항

인터럽트 태스크에서는 SFC 프로그램을 사용하지 마십시오. 정상적으로 동작하지 않습니다.
단, 추가 태스크로 설정한 경우는 사용할 수 있습니다.
(ST 프로그램은 인터럽트 태스크에서 사용 가능)

전원 OFF 인터럽트 태스크

전원OFF(전원 OFF) 시 실행되는 태스크입니다.
전원이 OFF된 경우 전원 유니트 자체에서 10ms 동안 5V 전원을 CPU 유니트에 계속 공급하며, 그 사이에 전원 OFF 인터럽트 태스크를 실행합니다.
(전원 유니트가 CJ1W-PD022인 경우에는 1ms 동안만 전원이 공급되므로 전원 OFF 인터럽트를 사용할 수 없습니다.)



- **전원 OFF 인터럽트 태스크의 인터럽트 오버헤드 시간**
  전원 OFF가 확정된 후 0.1ms 이내에 전원 OFF 인터럽트 태스크가 실행됩니다.

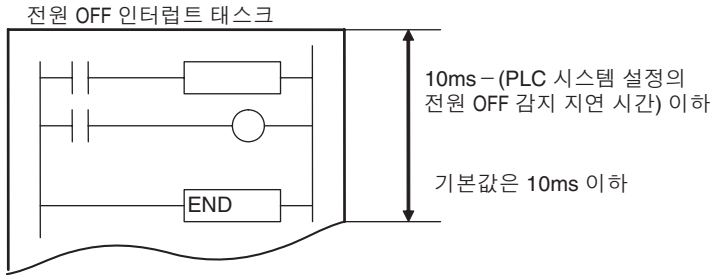
● 전원 OFF 인터럽트 태스크를 실행하기 위한 설정

CX-Programmer의 PLC 시스템 설정(타이머/인터럽트/동기 제어 탭)에서 전원 OFF 인터럽트 태스크를 유효하게 만듭니다.

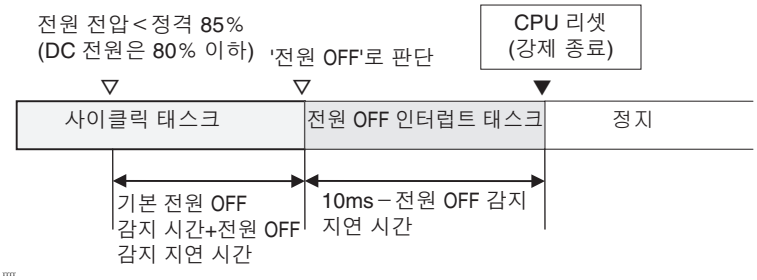
PLC 시스템 설정은 기본적으로 전원 OFF 인터럽트 태스크가 '수신 금지'(전원 OFF 인터럽트 사용 안 함 옵션 선택)로 설정되어 있습니다. 전원 OFF 인터럽트 태스크를 수신할 수 있게 하려면 PLC 시스템 설정에서 전원 OFF 인터럽트 사용 안 함 옵션의 선택을 취소하고 '전원 OFF 인터럽트 사용'으로 설정합니다.

기본적으로 PLC 시스템 설정에서는 전원 OFF 인터럽트 태스크를 10ms 후에 강제로 종료시킵니다. 따라서 이런 경우 전원 OFF 인터럽트 태스크의 실행 시간은 10ms 이하이어야 합니다.

PLC 시스템 설정의 전원 OFF 감지 지연 시간을 설정한 경우 전원 OFF 인터럽트 태스크가 10ms－(PLC 시스템 설정의 전원 OFF 감지 지연 시간) 후에 강제로 종료됩니다. 따라서 실행 시간은 10ms－(PLC 시스템 설정의 전원 OFF 감지 지연 시간) 이하가 되어야 합니다. (예: PLC 시스템 설정의 전원 OFF 감지 지연 시간을 4ms로 설정한 경우, 10-4ms인 6ms 이하가 되어야 합니다.)



(주) 또한 전원 전압이 정격의 85% 이하, DC 전원은 80% 이하가 된 후, '전원 OFF'로 판단하고 실제로 전원 OFF 인터럽트 태스크가 실행될 때까지 걸리는 시간은 기본적으로 전원 OFF 감지 시간(AC 전원: 10~25ms 미확정, DC 전원: 2~5ms 미확정)에 PLC 시스템 설정에서 설정한 전원 OFF 감지 지연 시간(0~10ms)을 더한 시간이 경과한 후입니다. 그 때까지는 사이클릭 태스크가 실행됩니다.



● 전원 OFF 인터럽트 태스크 사용상의 제약 사항

온라인 편집 실행 중 실행 불가능

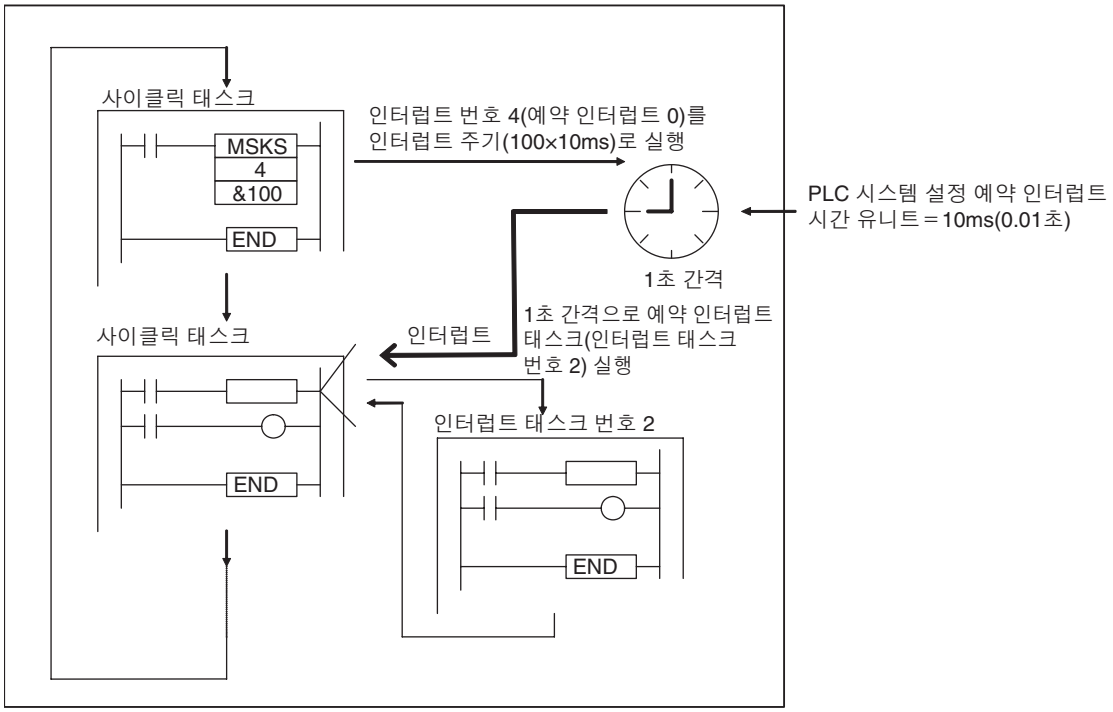
온라인 편집 실행 중에 전원 OFF가 발생한 경우 전원 OFF 인터럽트 태스크는 실행되지 않으므로 주의하십시오.

전원 OFF 인터럽트 태스크 내의 사용 불가 명령

전원 OFF 인터럽트 태스크에는 일반적으로 인터럽트 태스크에서 사용할 수 없는 명령과 더불어 추가로 사용할 수 없는 명령이 있습니다. 자세한 내용은 🕮 '5-2-4 태스크 설계 방법 주의 사항'을 참조하십시오.

예약 인터럽트 태스크

CPU 유니트의 내부 타이머에 의해 일정 시간 간격으로 태스크를 실행합니다.
예약 인터럽트 0, 1의 최대 2개 태스크(인터럽트 태스크 번호: 2, 3)를 사용할 수 있습니다.



● 인터럽트 태스크 번호와 인터럽트 번호의 대응

	인터럽트 태스크 번호	인터럽트 번호 (MSKS 명령)
예약 인터럽트 0	2	4
예약 인터럽트 1	3	5

● 예약 인터럽트 태스크의 실행

예약 인터럽트 태스크는 CPU 유니트가 운전을 시작한 시점에서는 실행 불가능 상태이며, MSKS(인터럽트 마스크 세트) 명령에 의해 실행할 수 있게 됩니다.
예약 인터럽트 태스크의 시간 간격은 PLC 시스템 설정의 '예약 인터럽트 시간 유니트' 값과 MSKS 명령 실행 시에 피연산자로 지정한 '예약 인터럽트 설정 시간' 값에 의해 결정됩니다.

- 예약 인터럽트 시간 유니트 설정(CX-Programmer의 PLC 시스템 설정('타이머/인터럽트/동기' 탭)에 따름)
  MSKS 명령으로 설정하는 인터럽트 시간의 단위를 설정합니다.
  10ms, 1.0ms, 0.1ms 중 하나(초기값: 10ms)
- 예약 인터럽트 설정 시간(MSKS 명령의 실행에 의함)
  예약 인터럽트 태스크를 실행하는 간격을 설정합니다.
  10진수&1~9999(16진수#0001~270F)

예: 예약 인터럽트 시간 유니트 설정이 '0.1ms'이고 예약 인터럽트 설정 시간이 '10진수&5'인 경우, 예약 인터럽트 간격은 0.1×5＝0.5ms입니다.

예약 인터럽트 태스크의 시간 간격으로 설정 가능한 최소값은 다음과 같습니다.
CJ2H CPU 유니트: 0.2ms
CJ2M CPU 유니트: 0.4ms
단, CJ2H CPU 유니트의 유니트 Ver.1.1 이상에서는 고속 인터럽트 기능을 사용하여 예약 인터럽트 0(인터럽트 태스크 번호 2)에 대해서만 0.1ms를 설정할 수 있습니다.
고속 인터럽트 기능에 대해서는 '10-2-6 고속 인터럽트 기능'을 참조하십시오.

사용상의 주의 사항

- 예약 인터럽트의 시간 간격이 짧고 예약 인터럽트 태스크 실행 빈도가 높으면 사이클릭 태스크/추가 태스크 실행 완료까지 시간이 걸려서 사이클 시간이 길어지므로 주의하십시오.
- 유니트 간 동기 제어 기능을 사용할 때는 예약 인터럽트 태스크를 사용할 수 없습니다.

● 예약 인터럽트 태스크의 인터럽트 오버헤드 시간

MSKS 명령을 통해 설정한 일정 시간이 경과한 후 실제로 인터럽트 태스크가 실행될 때까지의 시간을 예약 인터럽트의 '인터럽트 태스크 기동 시간'이라고 합니다.
그리고 인터럽트 태스크 프로그램을 실행한 후 인터럽트 전의 처리로 되돌아갈 때까지의 시간을 '사이클릭 태스크로 복귀하는 시간'이라고 합니다.
인터럽트 태스크 기동 시간과 사이클릭 태스크로 복귀하는 시간을 합하여 '인터럽트 오버헤드 시간'이라고 합니다.

	항목	내용	
		CJ2H CPU 유니트	CJ2M CPU 유니트
예약 인터럽트 태스크의 인터럽트 오버헤드 시간	인터럽트 태스크 기동 시간	22μs 또는 13μs(*) (유니트 Ver.1.0의 경우 27μs)	30μs
	사이클릭 태스크로 복귀하는 시간	11μs 또는 8μs(*) (유니트 Ver.1.0의 경우 15μs)	11μs

* 고속 인터럽트 기능 사용 시

'고속 인터럽트 기능'에 대한 자세한 내용은 '10-2-6 고속 인터럽트 기능'을 참조하십시오.

● MSKS 명령에 의한 리셋 지정

CJ2M CPU 유니트에서는 MSKS 명령에 의한 예약 인터럽트 기동 시, 내부 타이머를 리셋한 후 시작하는 기능(리셋)을 선택할 수 있습니다.
이 기능을 통해 CLI 명령을 사용하지 않고도 첫 번째 인터럽트 시작 시간을 일정하게 할 수 있습니다.
예약 인터럽트의 기동은 MSKS 명령을 통해 예약 인터럽트 시간(인터럽트 및 인터럽트 시간 간격)을 설정하면 실행됩니다. 단, MSKS 명령 실행 후 첫 번째 예약 인터럽트 태스크가 기동되기까지의 시간(첫 번째 인터럽트 시작 시간)은 내부 타이머의 현재값 상태에 따라 달라지므로 별도의 CLI 명령으로 지정하지 않는 한 보증되지 않습니다. 그에 비해 CJ2M CPU 유니트의 경우 내부 타이머를 리셋할 수 있으므로 CLI 명령을 사용하지 않고도 첫 번째 인터럽트 시작 시간을 일정하게 설정할 수 있습니다.

MSKS 명령의 피연산자(예약 인터럽트 지정 시에만)

피연산자	설정값
N(인터럽트 번호)	4: 예약 인터럽트 0 일반 지정(내부 타이머를 리셋하지 않음: 인터럽트 태스크 번호 2) 5: 예약 인터럽트 1 일반 지정(내부 타이머를 리셋하지 않음: 인터럽트 태스크 번호 3) 14: 예약 인터럽트 0 리셋 지정(CJ2M CPU 유니트만 해당: 인터럽트 태스크 번호 2) 15: 예약 인터럽트 1 리셋 지정(CJ2M CPU 유니트만 해당: 인터럽트 태스크 번호 3)

● MSKR 명령에 의한 내부 타이머 현재값 읽기

CJ2M CPU 유니트에서는 예약 인터럽트 시간을 계측하는 내부 타이머의 현재값을 읽을 수 있습니다.
예약 인터럽트 기동 시점 또는 이전의 예약 인터럽트 시점에서 경과 시간을 읽습니다.
내부 타이머의 현재값 읽기는 MSKR 명령으로 실행합니다.
또한 시간의 단위는 예약 인터럽트 시간과 동일하게 PLC 시스템 설정의 '예약 인터럽트 시간 유니트' 설정에 따라 결정됩니다.

MSKR 명령의 피연산자(예약 인터럽트 지정 시에만)

피연산자	설정값
N(인터럽트 번호)	4: 예약 인터럽트 0　예약 인터럽트 시간(설정값) 읽기(인터럽트 태스크 번호 2) 5: 예약 인터럽트 1　예약 인터럽트 시간(설정값) 읽기(인터럽트 태스크 번호 3) 14: 예약 인터럽트 0 내부 타이머 현재값 읽기 (CJ2M CPU 유니트만 해당: 인터럽트 태스크 번호 2) 15: 예약 인터럽트 1 내부 타이머 현재값 읽기 (CJ2M CPU 유니트만 해당: 인터럽트 태스크 번호 3)

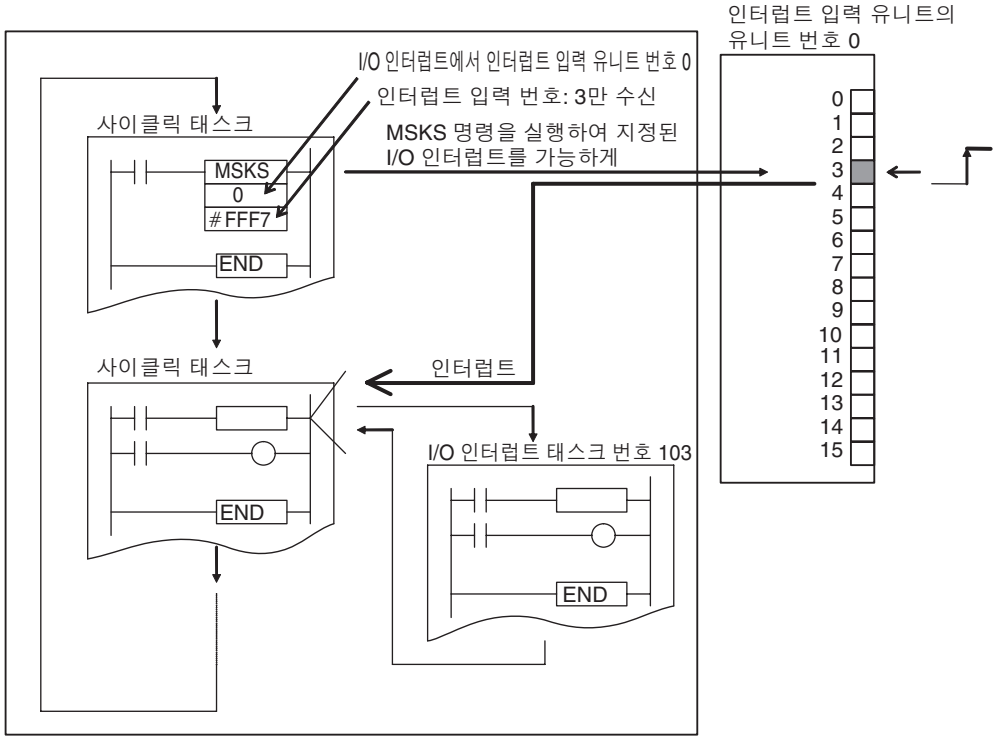
I/O 인터럽트 태스크

인터럽트 입력 유니트의 접점을 기동할 때 실행됩니다. 작성 가능한 태스크의 수는 최대 32개(인터럽트 태스크 번호: 100~131)입니다.

● 인터럽트 태스크 실행

MSKS(인터럽트 마스크 세트) 명령을 사용하여 I/O 인터럽트를 수신 가능하도록 설정합니다.

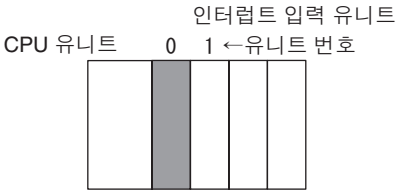
예: 인터럽트 입력 유니트의 유니트 번호 0(왼쪽부터 장착 순서에 따라 0~1의 가장 왼쪽)의 인터럽트 입력 번호 3을 기동하면 I/O 인터럽트 태스크 번호 103이 실행됩니다.



인터럽트 입력 유니트/입력 번호와 I/O 인터럽트 태스크 번호의 대응

인터럽트 입력 유니트 번호(*)	인터럽트 입력 번호	I/O 인터럽트 태스크 번호
0	0~15	100~115
1	0~15	116~131

* 인터럽트 입력 유니트 번호는 CPU 유니트에 가장 가까운 쪽부터 순서대로 0→1입니다.



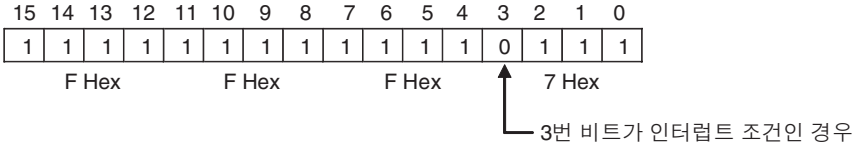
● MSKS(인터럽트 마스크 세트) 명령

I/O 인터럽트 태스크를 수신하려면 사이클릭 태스크의 프로그램에서 MSKS(인터럽트 마스크 세트) 명령을 실행하고, 지정한 인터럽트 입력 유니트의 지정한 인터럽트 번호의 인터럽트를 수신 가능으로 설정합니다.

MSKS 명령의 피연산자 S(제2 피연산자): FFF7Hex의 내용

각 비트가 인터럽트 입력 유니트 번호의 인터럽트 입력 번호에 대응하므로 인터럽트 입력 번호의 0~15는 비트 0~15에 대응합니다.

예)



사용상의 주의 사항

필요 없는 번호의 I/O 인터럽트 태스크는 수신 가능하도록 설정하지 마십시오.
노이즈 등으로 인터럽트 입력이 들어간 경우 적합한 프로그램이 없기 때문에 프로그램 정지 이상(태스크 에러)을 일으킬 수 있습니다.

● I/O 인터럽트 태스크의 인터럽트 오버헤드 시간

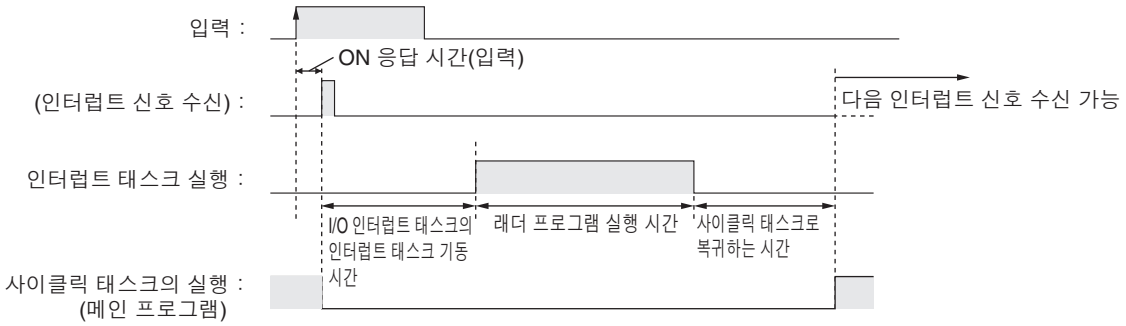
인터럽트 입력 유니트(CJ1W-INT01)의 입력 접점이 OFF→ON(또는 ON→OFF)으로 기동한 후, CPU 유니트가 인터럽트 신호를 수신할 때까지의 시간은 하드웨어의 응답 시간에 따라 달라집니다. 또한 인터럽트 신호를 수신한 후 실제로 I/O 인터럽트 태스크가 실행되기까지의 시간을 I/O 인터럽트 태스크의 '인터럽트 태스크 기동 시간'이라고 합니다.
그리고 인터럽트 태스크 프로그램을 실행한 후 인터럽트 전의 처리로 되돌아갈 때까지의 시간을 '사이클릭 태스크로 복귀하는 시간'이라고 합니다.
인터럽트 태스크 기동 시간과 사이클릭 태스크로 복귀하는 시간을 합하여 '인터럽트 오버헤드 시간'이라고 합니다.

항목		내용	
		CJ2H CPU 유니트	CJ2M CPU 유니트
하드웨어의 응답 시간(CJ1W-INT01)		기동 0.05ms/종료 0.5ms	
인터럽트 오버헤드 시간	인터럽트 태스크 기동 시간	26μs 또는 17μs(*1) (유니트 Ver.1.0의 경우 30μs)	31μs
	사이클릭 태스크로 복귀하는 시간	11μs 또는 8μs(*1) (유니트 Ver.1.0의 경우 15μs)	10μs

*1 고속 인터럽트 기능 사용 시

'고속 인터럽트 기능'에 대한 자세한 내용은 '10-2-6 고속 인터럽트 기능'을 참조하십시오.

(주) I/O 인터럽트 태스크는 사용자 프로그램 실행, I/O 리프레시, 주변 서비스, 공통 처리 중 하나의 타이밍에서도 실행할 수 있습니다(명령 실행 시에도 해당 명령의 실행을 중단하고 실행됩니다).
단, 다른 인터럽트 태스크를 실행하는 중에는 I/O 인터럽트 조건이 성립되어도 즉시 실행되지 않고 실행 중인 인터럽트 태스크의 실행이 종료되기를 기다린 다음(사이클릭 태스크로 복귀하는 시간 경과 후), 인터럽트 태스크 기동 시간 후에 실행됩니다.

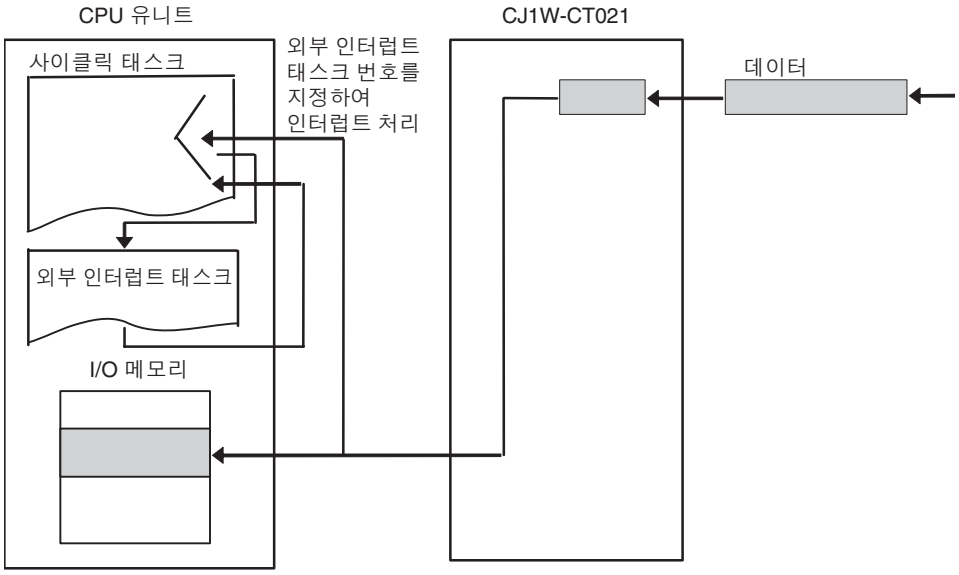


외부 인터럽트 태스크

고기능 I/O 유니트, CPU 고기능 유니트가 보유한 CPU 유니트에 대한 인터럽트 기능에 의해 인터럽트 처리가 실행됩니다.
CPU 유니트에는 외부 인터럽트를 위한 설정 항목이 없습니다.
단, 지정된 번호의 외부 인터럽트 태스크를 사용자 프로그램에 저장해야 합니다.
또한 외부 인터럽트는 CPU 장치에 유니트가 장착되어야 합니다.
외부 인터럽트 태스크는 항상 수신 가능하며, 최대 256개 태스크(인터럽트 태스크 번호: 0~255)를 사용할 수 있습니다.

(주) 외부 인터럽트 태스크 번호가 전원 OFF 인터럽트, 예약 인터럽트, I/O 인터럽트와 동일한 번호인 경우에는 어떤 조건에서나 동작합니다(OR 조건에서 동작). 따라서 번호가 중복되지 않도록 하십시오.

예: 고속 카운터 유니트 CJ1W-CT021에서 수신된 외부 인터럽트인 경우



I/O 리프레시 이외에, 이 유니트의 2점에 대한 외부 컨트롤 입력 및 32점에 대한 출력 상태 정보를 CPU 유니트에 통지하고 싶은 경우 외부 컨트롤 입력 또는 출력이 외부 인터럽트 태스크를 기동시키도록 설정합니다.

사용상의 주의 사항

외부 인터럽트 태스크 번호 0~255, 전원 OFF 인터럽트 태스크 번호 1, 예약 인터럽트 태스크 번호 2와 3, I/O 인터럽트 태스크 번호 100~131이 중복된 경우, 외부 인터럽트의 조건과 기타 인터럽트의 조건 모두에서 동작합니다. 따라서 번호가 중복되지 않도록 하십시오.

● 외부 인터럽트 태스크의 인터럽트 오버헤드 시간

외부 인터럽트 태스크의 경우, CPU 유니트가 인터럽트 신호를 수신할 때까지의 시간은 CPU 유니트에 대해 외부 인터럽트 태스크의 실행을 요구하는 각 유니트/보드(고기능 I/O 유니트, CJ 시리즈 CPU 고기능 유니트)의 종류 및 인터럽트를 요구하는 서비스의 종류에 따라 다릅니다. 자세한 내용은 각 유니트의 설명서를 참조하십시오.
CPU 유니트가 인터럽트 신호를 수신한 이후의 인터럽트 오버헤드 시간은 I/O 인터럽트 태스크의 경우와 같습니다.

추가 태스크

인터럽트 태스크를 사이클릭 태스크와 동일하게 취급할 수 있습니다.
1사이클에 1회, 사이클릭 태스크(사이클릭 태스크 번호 0~127)를 실행한 후, 태스크 상태가 실행 가능한 상태이면 인터럽트 태스크 번호(인터럽트 태스크 번호 0~255)가 작은 순서로 실행됩니다. 최대 256개 태스크(인터럽트 태스크 번호: 0~255).
추가 태스크의 인터럽트 태스크 번호가 전원 OFF 인터럽트, 예약 인터럽트, I/O 인터럽트와 동일한 번호인 경우 어떤 조건에서나 동작합니다(OR 조건에서 동작). 따라서 인터럽트 태스크로 사용하는 태스크 번호를 추가 태스크로 사용하지 마십시오.

(주) 1 추가 태스크 내에서 태스크 실행/태스크 정지 명령을 기술하고 실행할 수도 있습니다.
단, 이 태스크가 인터럽트 태스크로 동작한 경우 태스크 실행/태스크 실행 정지 명령은 실행되지 않습니다.
2 추가 태스크는 다음과 같은 점에서 사이클릭 태스크와 다릅니다.

항목	추가 태스크	사이클릭 태스크
프로그램의 속성 '운전 시작 시 기동'	없음	CX-Programmer로 설정 가능
태스크 플래그	없음	있음 (TK000~TK127이 사이클릭 태스크의 태스크 번호 0~127에 대응)
태스크 첫 회 기동 플래그(A200.15), 태스크 기동 플래그(A200.14)	없음	있음
IR(인덱스 레지스터)/ DR(데이터 레지스터)의 값	인터럽트 태스크와 동일하게 기동 시에는 일정치 않습니다. 각 사이클의 값은 일정치 않습니다. 반드시 값을 설정한 후 사용하십시오. 다음 사이클에서는 이전 사이클에서 설정한 값을 읽을 수 없습니다.	운전 시작 시에는 일정치 않음. 다음 사이클에서는 이전 사이클에서 설정한 값을 읽을 수 있음

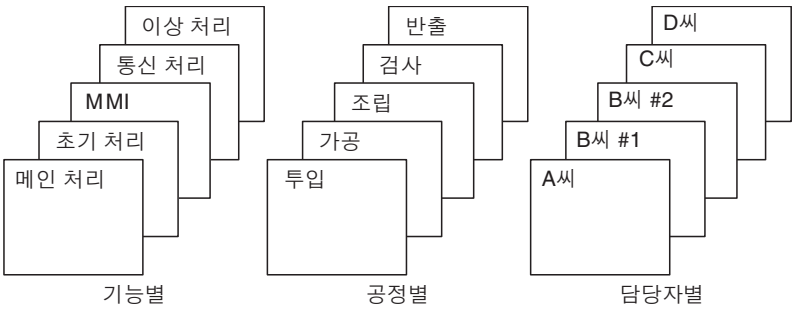
5-2-4 태스크 설계 방법

설계 지침

신뢰성, 보수성이 높은 시스템을 구축하기 위해서는 '어떻게 태스크를 설계할 것인가'가 중요합니다. 다음과 같은 점에 주의할 필요가 있습니다.

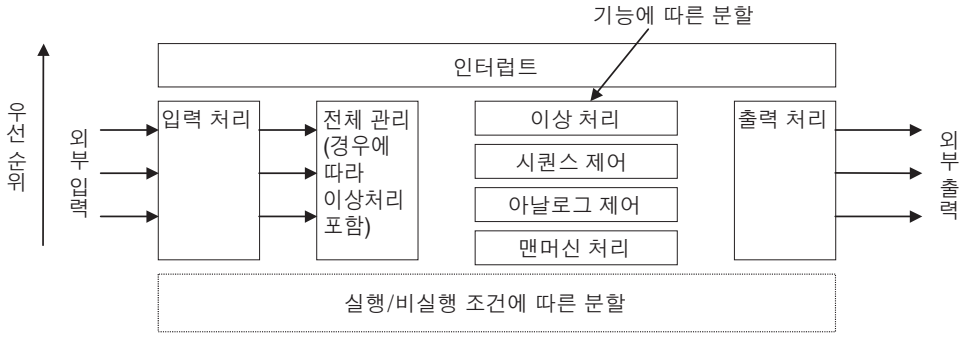
- 태스크 분할은 다음과 같은 기준으로 검토합니다.
  1) 실행/비실행의 조건을 명확히 하고 상태 이동을 이해하고 구분합니다.
  2) 외부 입출력의 유무에 따라 구분합니다.
  3) 기능이나 담당자별로 구분합니다.
  4) 실행 우선 순위로 구분합니다(사이클릭 태스크/인터럽트 태스크로 나눔).
     태스크 중에 우선 순위가 높은 태스크에는 작은 태스크 번호를 할당합니다.

예 1) 기능별, 담당자별로 태스크 설계

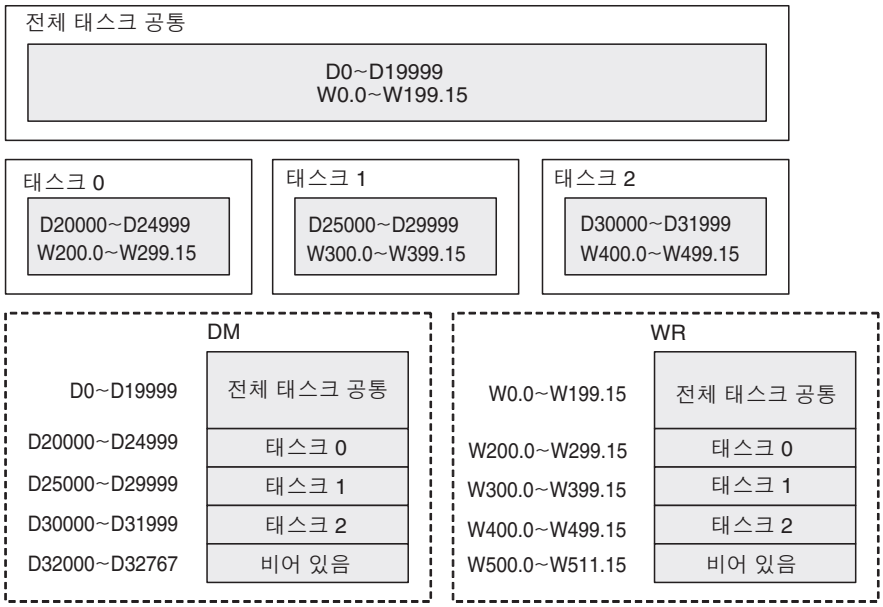


예 2) 실행 우선 순위로 태스크 설계

관리용 태스크에는 각 처리용 태스크보다 작은 태스크 번호를 할당하십시오.



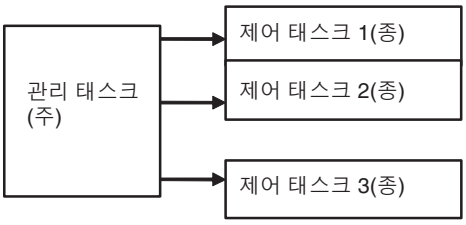
- 태스크는 기본적으로 독립성을 높이고 태스크(프로그램) 간에 데이터를 최소한으로 주고 받도록 분할 설계합니다.
  모든 태스크에서 공통으로 사용하는 I/O 메모리와 각 태스크 내에서만 사용하는 I/O 메모리를 구분하여 각 태스크에서만 사용하는 I/O 메모리의 범위를 태스크별로 구분하여 확보합니다.



- 전체를 관리하는 태스크를 통해 각 태스크의 기동/정지를 제어합니다.

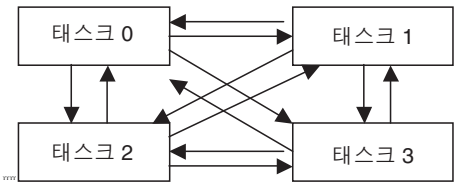
  <좋은 예>

  상하 관계를 명확히 합니다.



  <좋지 않은 예>

  태스크 간에 기동/정지를 복잡하게 지시하면 관리하기 어려워집니다.



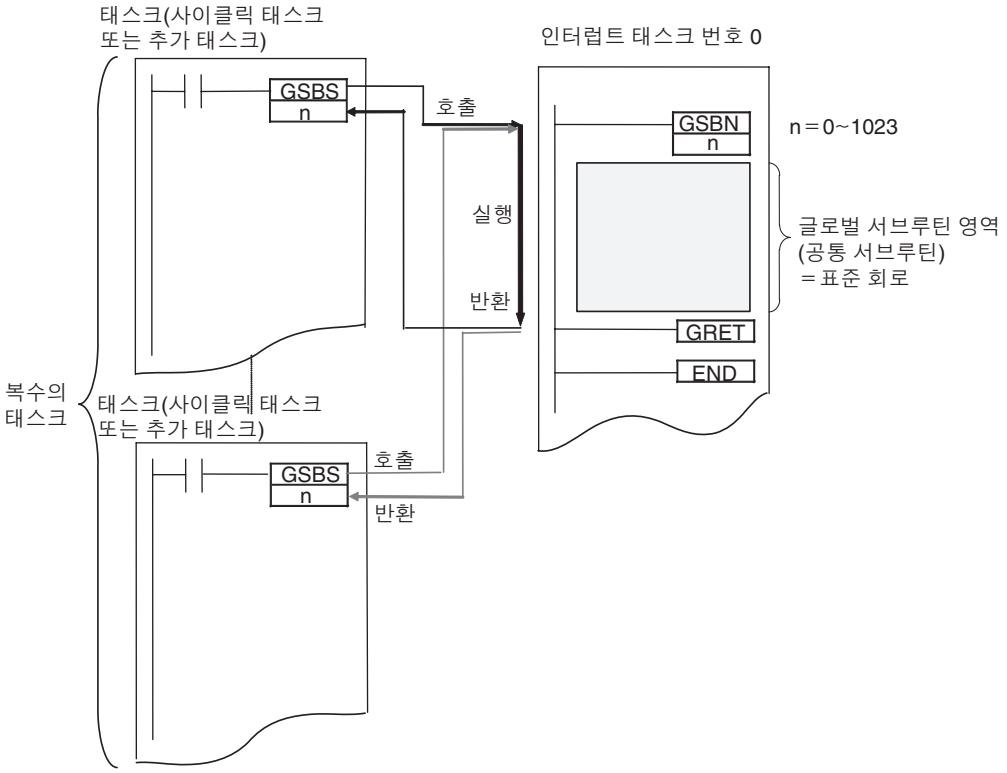
- 우선 순위가 높은 태스크에는 작은 인터럽트 태스크 번호를 할당하십시오.
- 태스크는 한번 기동하면 자신 또는 다른 태스크에 의해 정지되지 않는 한 다음 사이클 이후에도 실행 상태가 됩니다. 조건에 의한 태스크 분기를 처리할 때는 다른 태스크에 대한 태스크 정지 명령을 잊지 말고 삽입하십시오.
- 태스크 실행 시의 초기화 처리를 수행할 때는 다음을 사용하십시오.
  - 운전 중에 한번만 초기화 처리를 수행하는 경우: 태스크 첫 회 기동 플래그(A200.15)
  - 태스크를 기동할 때마다 초기화 처리를 수행하는 경우: 태스크 기동 플래그(A200.14)

참고

글로벌 서브루틴 기능

서브루틴 명령의 경우 한 태스크에서 다른 태스크에 있는 서브루틴을 호출할 수 없습니다.
글로벌 서브루틴 명령을 사용하면 하나 이상의 태스크(사이클릭 태스크 또는 추가 태스크)에서 특정 태스크(인터럽트 태스크 번호 0만 해당) 내의 서브루틴을 호출할 수 있습니다.

글로벌 서브루틴의 호출은 GSBS 명령을 사용하여 수행합니다.
지정 가능한 서브루틴 번호는 0~1023입니다.
글로벌 서브루틴은 인터럽트 태스크 번호 0인 프로그램 끝부분(END 명령 직전)에서 GSBN 명령~GRET 명령으로 정의합니다.
이렇게 하면 공통으로 호출하는 서브루틴을 표준 회로로 라이브러리화할 수 있습니다.



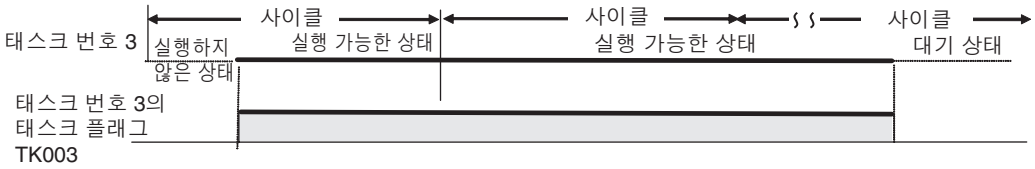
관련 플래그

● 사이클릭 태스크 관련 플래그

다음 플래그는 사이클릭 태스크 관련 플래그며, 추가 태스크와는 관계 없습니다.

태스크 플래그(TK000~TK127)

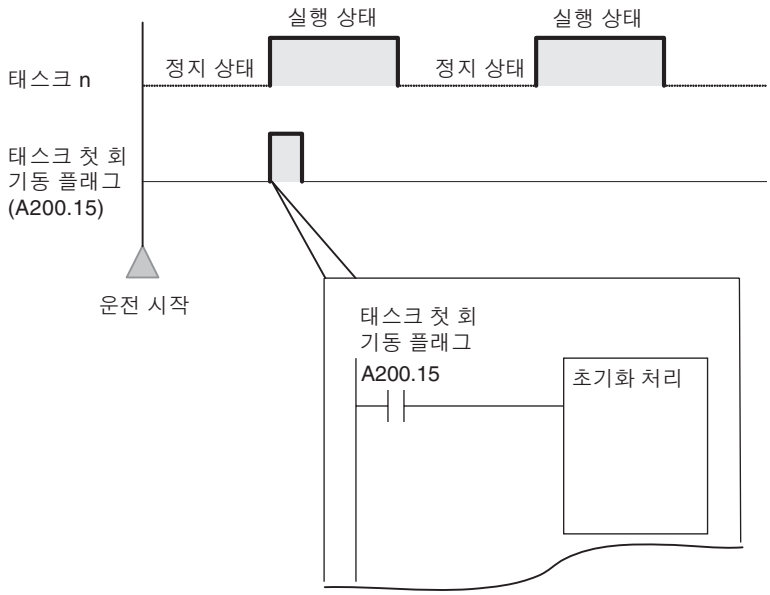
이 플래그는 현재 태스크가 실행되고 있는지 여부를 확인할 때 사용합니다. 사이클릭 태스크가 실행 가능 상태(READY)이면 1(ON)이 되고 실행되지 않는 상태(INI) 또는 정지 상태(WAIT)이면 0(OFF)이 됩니다. 또한 태스크 번호 00~127이 TK000~127에 대응합니다.



(주) 이 플래그는 사이클릭 태스크에 적합하며 인터럽트 태스크에는 적합하지 않습니다. 인터럽트 태스크는 인터럽트 태스크 관련 플래그를 사용하십시오.
인터럽트 태스크에는 A440CH(인터럽트 태스크 최대값 처리 시간), A441CH(최대 처리 시간 인터럽트 태스크 번호) 등의 인터럽트 태스크 관련 플래그를 사용하십시오.
인터럽트 태스크 관련 플래그는 다음 페이지를 참조하십시오.

태스크 첫 회 기동 플래그(A200.15)

운전 중에 한번만 초기화 처리를 수행하는 경우 사용합니다.
사이클릭 태스크가 처음 실행될 때 ON이 됩니다. 그리고 실제로 실행이 종료되었을 때 OFF가 됩니다.

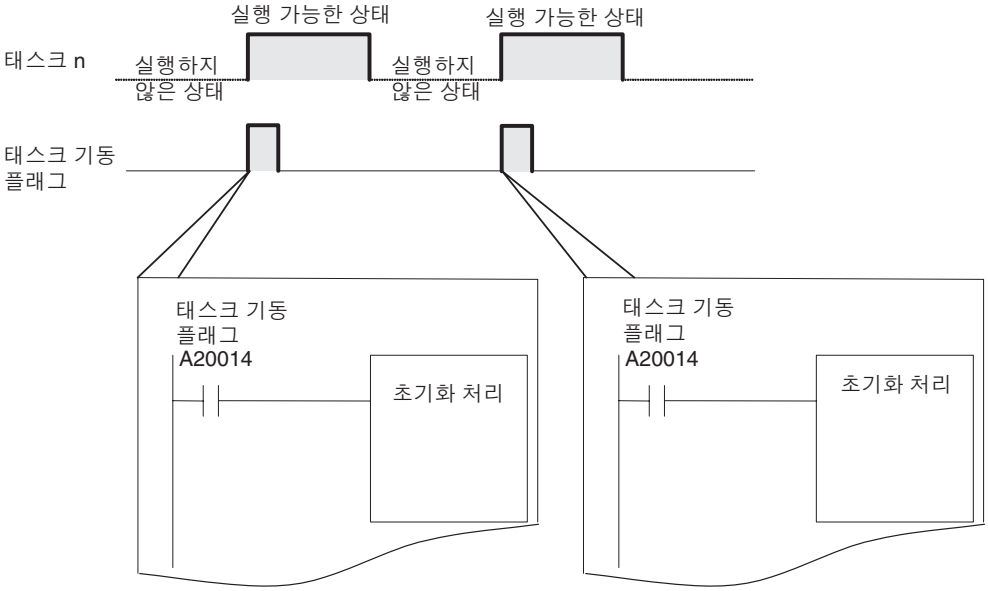


사이클릭 태스크는 이 플래그를 통해 자신의 실행이 첫 회인지 여부를 판단하며, 첫 회인 경우에는 초기화 처리 등을 수행합니다.

(주) 한번 실행한 후 정지한 사이클릭 태스크는 태스크 기동 명령을 통해 다시 실행해도 첫 회 기동으로 인식되지 않으므로 태스크 첫 회 기동 플래그(A200.15)는 ON이 되지 않습니다. 이 경우 태스크 기동 플래그(A200.14)를 사용하십시오.

태스크 기동 플래그(A200.14)

태스크를 기동할 때마다 초기화 처리를 수행하는 경우 사용합니다.
사이클릭 태스크가 정지 상태에서 실행 상태(RUN)로 되었을 때 ON이 됩니다.
이 플래그는 태스크 첫 회 기동 플래그(A200.15)와 달리, 정지 상태에서 기동할 때마다 ON이 됩니다.



이 플래그를 입력 조건으로 하면 태스크 기동 시(한번 정지 상태가 된 사이클릭 태스크를 TKON 명령으로 실행 상태로 했을 때)의 초기화 처리를 실행할 수 있습니다.

- **인터럽트 태스크 관련 플래그**

인터럽트 태스크 최대값 처리 시간(A440CH)*

인터럽트 태스크의 최대 처리 시간이 0.1ms 단위의 BIN 데이터로 저장되며, 운전을 시작할 때 클리어됩니다.

최대 처리 시간 인터럽트 태스크 번호(A441CH)*

최대 처리 시간의 인터럽트 태스크 번호가 BIN 데이터로 저장됩니다. 또한 태스크 번호 00~FFHex에 대응하여 8000Hex~80FFHex가 됩니다.
운전을 시작할 때 클리어됩니다.

다중 리프레시 이상 플래그(운전 계속 이상) (A402.13)

PLC 시스템 설정의 '다중 리프레시 이상 검출 유무 설정'에서 '검출'을 설정한 경우, 다중 리프레시 이상이 발생하면 1(ON)이 됩니다.

* PLC 시스템 설정에서 '고속 인터럽트 기능 사용'을 설정한 경우, A440CH, A441CH는 무효가 됩니다.

다중 리프레시 이상 요인 플래그(A426.15)/다중 리프레시 이상 발생 태스크 번호 (A426.00~426.11)

A402.13이 1(ON)인 경우, A426.15 및 A426.00~426.11에 다음과 같은 정보가 저장됩니다.

A402.13	다중 리프레시 이상 내용	A426.15	A426.00~426.11
다중 리프레시 이상 발생 (PLC 시스템 설정의 '다중 리프레시 이상 검출 유무 설정'에서 '검출'로 설정한 경우)	PLC 시스템 설정의 '다중 리프레시 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(*)	1(ON)	다중 리프레시 대상인 고기능 I/O 유니트의 기기 번호가 BIN 12비트로 저장됩니다. (기기 번호 0~95: 000~05FHex)

* PLC 시스템 설정에서 '고기능 I/O 유니트와 주기적으로 리프레시'로 설정한 상태에서 인터럽트 태스크 중에 동일한 고기능 I/O 유니트에 대해 IORF/FIORF/IORD/IOWR 명령을 실행하면 다중 리프레시 상태가 되어 다중 리프레시 이상이 됩니다.

프로그램 정지 시 태스크 번호(A294CH)

프로그램 에러에 의해 실행이 정지된 경우, 정지 위치의 태스크 종류 및 태스크 번호가 다음 값으로 저장됩니다.

태스크 유형	A294CH
인터럽트 태스크	8000~80FFHex(인터럽트 태스크 번호 0~255에 대응)
사이클릭 태스크	0000~007FHex(태스크 번호 0~127에 대응)

● 태스크 관련 공통 플래그

태스크 에러 플래그(A295.12)

태스크 에러가 발생했을 때는 1(ON)이 됩니다. 태스크 에러는 다음과 같은 내용을 가리킵니다.

- 실행 가능 상태(READY)인 사이클릭 태스크가 해당 사이클 내에서 하나도 존재하지 않는 경우
- 사이클릭 태스크에 할당된 프로그램이 존재하지 않는 경우
  (주변 툴 CX-Programmer를 사용한 경우에는 이 상태가 발생하지 않습니다.)
- 기동된 인터럽트 태스크(추가 태스크 포함)에 할당된 프로그램이 존재하지 않는 경우

프로그램 정지 시 태스크 번호(A294CH)

프로그램 에러에 의해 실행이 정지된 경우, 정지 위치의 태스크 종류/태스크 번호가 다음 값으로 저장됩니다.

태스크 유형	A294CH
사이클릭 태스크	0000~007FHex(태스크 번호 0~127에 대응)
인터럽트 태스크(추가 태스크 포함)	8000~80FFHex(인터럽트 태스크 번호 0~255에 대응)

이 플래그를 통해 운전 정지 이상이 어느 태스크에서 발생했는지 확인할 수 있습니다. 운전 정지 이상이 해제되면 클리어됩니다.
또한 해당 태스크의 프로그램 중 어느 프로그램 주소에서 정지했는지에 대한 정보는 A298CH(하위 프로그램 주소), A299CH(상위 프로그램 주소)에 저장됩니다.

주의 사항

● 명령의 사용 제한

태스크를 벗어나 사용할 수 없는 명령

다음 명령은 세트로 사용하므로 동일 태스크 내에 배치해야 합니다.

니모닉	명령어
JMP/JME	점프/점프 끝
CJP/JME	조건부 점프/점프 끝
CJPN/JME	조건부 부정 점프/점프 끝
JMP0/JME0	복수 점프/복수 점프 끝
FOR/NEXT	반복 시작/종료
IL/ILC	인터록/인터록 클리어
SBS/SBN/RET	서브루틴 호출/서브루틴 진입/서브루틴 반환(*)
MCRO/SBN/RET	매크로/서브루틴 진입/서브루틴 반환
BPRG/BEND	블록 프로그램/블록 프로그램 끝
STEP/STEP	스텝 래더 영역 명령

* 글로벌 서브루틴 명령(GSBS/GSBN/GRET)을 사용하면 다른 태스크에서도 호출할 수 있습니다.

인터럽트 태스크 내에서 사용할 수 없는 명령

다음 명령은 인터럽트 태스크 내에서 실행할 수 없습니다. 인터럽트 태스크 내에서 실행한 경우 에러 플래그(P_ER)가 ON으로 되어 명령이 실행되지 않습니다.
단, 인터럽트 태스크를 추가 태스크로 사용하는 경우에는 다음 명령을 사용할 수 있습니다.

니모닉	명령어
TKON	태스크 실행 기동
TKOF	태스크 실행 정지
STEP	스텝 래더 영역
SNXT	스텝 래더 진행
STUP	시리얼 포트 통신 설정 변경
DI	인터럽트 태스크 실행 금지
EI	인터럽트 태스크 실행 금지 해제
SFCON, SFCOFF, SFCPR, SFCPRN	SFC 태스크 제어 명령

- 인터럽트 태스크 내에서는 다음 명령의 동작을 보증할 수 없습니다.
  100ms 타이머(TIM/TIMX), 10ms 타이머(TIMH/TIMHX), 1ms 타이머(TMHH/TMHHX), 0.1ms 타이머(TIMU/TIMUX), 0.01ms 타이머(TMUH/TMUHX), 적산 타이머(TTIM/TTIMX), 멀티 출력 타이머(MTIM/MTIMX), 장시간 타이머(TIML/TIMLX), 블록 프로그램의 타이머 대기(TIMW/TIMWX), 고속 타이머 대기(TMHW/TMHWX)
  PID 제어 명령(PID/PIDAT), 고장 지점 검출 명령(FPD)

- 전원 OFF 인터럽트 태스크 내에서는 위 명령 이외에 다음 명령도 사용할 수 없습니다.
  전원 OFF 인터럽트 태스크 내에서 사용한 경우 에러 플래그(P_ER)가 ON으로 되지 않습니다. 또한 명령은 실행되지 않습니다.
  데이터 파일 읽기(FREAD), 데이터 파일 쓰기(FWRIT), 텍스트 파일 쓰기(TWRIT), 네트워크 송신(SEND), 네트워크 수신(RECV), 명령 송신(CMND), 프로토콜 매크로(PMCR), 시리얼 포트 출력(TXD), 시리얼 포트 입력(RXD), 범용 Explicit 메시지 송신 명령(EXPLT), Explicit 읽기 명령(EGATR), Explicit 쓰기 명령(ESATR), Explicit CPU 유니트 데이터 읽기 명령(ECHRD), Explicit CPU 유니트 데이터 쓰기 명령(ECHWR), 시리얼 커뮤니케이션 유니트 시리얼 포트 출력(TXDU), 시리얼 커뮤니케이션 유니트 시리얼 포트 입력(RXDU)

- CJ2H CPU 유니트에서 PLC 시스템 설정의 '고속 인터럽트 기능 사용'을 설정한 경우 다음과 같은 명령은 인터럽트 태스크 내에서 사용할 수 없으며, 실행하면 명령 처리 에러가 발생합니다.
  확장 DM 뱅크 전환(EMBC), 네트워크 송신(SEND), 네트워크 수신(RECV)
  명령 송신(CMND), 프로토콜 매크로(PMCR), 시리얼 커뮤니케이션 유니트 시리얼 포트 출력(TXDU), 시리얼 커뮤니케이션 유니트 시리얼 포트 입력(RXDU), 범용 Explicit 메시지 송신(EXPLT), Explicit 읽기 명령(EGATR), Explicit 쓰기 명령(ESATR), Explicit CPU 유니트 데이터 읽기 명령(ECHRD), Explicit CPU 유니트 데이터 쓰기 명령(ECHWR)
  또한 PLC 시스템 설정에서 백그라운드 처리를 지정한 명령(테이블 데이터 처리 명령, 문자열 처리 명령, 데이터 이동 명령)도 명령 처리 에러가 됩니다.

참고

태스크와 타이머 동작의 관계

타이머 명령은 다음과 같이 동작합니다.

- 타이머 번호가 0~2047인 경우
  타이머를 기동한 태스크가 정지 상태가 되거나 재기동되어도 타이머의 현재값 갱신은 그대로 계속됩니다.
  따라서, 예를 들어 TIM 명령을 기동한 태스크가 일단 정지되었다가 잠시 후 재기동된 경우, 현재값이 0이면 TIM 명령 실행 시 시간 종료(Time Up) 플래그가 ON이 됩니다(타이머의 업(Up) 플래그 갱신은 모두 명령이 실행될 때만 가능하기 때문). 현재값이 갱신되는 중이면 그대로 갱신을 계속합니다.
- 타이머 번호가 2048~4095인 경우
  타이머를 기동한 태스크가 정지 상태이면 타이머 현재값은 갱신되지 않고 유지됩니다.
  단, PLC 속성에서 'T2048-4095의 TIM 실행을 다른 TIM과 동일하게'를 선택한 경우, 타이머 번호 2048~4095는 타이머 번호0~2047과 동일한 동작이 됩니다.

태스크와 조건 플래그의 관계

조건 플래그는 각 태스크를 실행하기 직전에 모두 클리어됩니다.
따라서 예를 들어 태스크 번호 1에 대한 마지막 조건 플래그의 상태를 태스크 번호 2에서 읽을 수 없습니다. 읽는 경우는 CCS/CCL 명령을 사용하십시오.

● 인터럽트 태스크의 주의 사항

인터럽트 태스크의 우선 순위

전원 OFF 인터럽트 태스크는 다른 인터럽트 태스크가 실행되는 중이어도 그 인터럽트 태스크의 실행을 강제로 중단시키고 실행됩니다. 전원 OFF 인터럽트 태스크 실행 후, 중단된 인터럽트 태스크는 실행되지 않으며 CPU는 리셋됩니다.
그 밖의 인터럽트 태스크 우선 순위는 다음과 같습니다.

복수의 인터럽트 요인이 동시에 발생한 경우

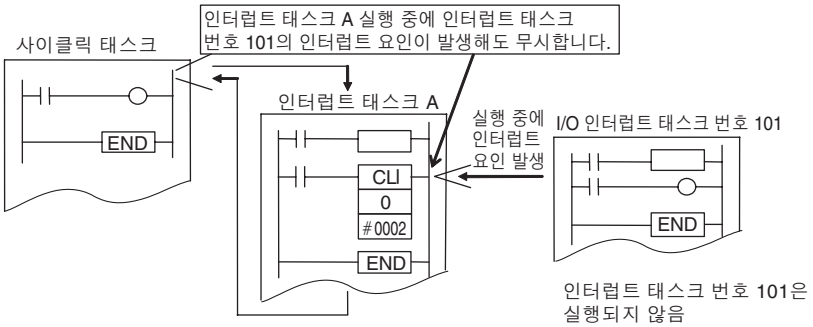
전원 OFF 인터럽트 태스크 이외의 인터럽트 요인이 동시에 발생했을 때의 인터럽트 태스크 실행 순서는 I/O 인터럽트 태스크 > 외부 인터럽트 태스크 > 예약 인터럽트 태스크가 됩니다.
그리고 I/O 인터럽트 태스크, 외부 인터럽트 태스크, 예약 인터럽트 태스크에서 각각 여러 요인이 동시에 발생한 경우에는 태스크 번호가 작은 순서로 실행됩니다.

인터럽트 태스크 실행 중에 다른 인터럽트 요인이 발생한 경우

한 인터럽트 태스크가 실행되는 중에 다른 인터럽트 태스크의 인터럽트 요인이 발생한 경우, 실행 중인 인터럽트 태스크의 실행이 종료되기를 기다린 후 다른 인터럽트 태스크를 실행합니다.

참고

번호가 지정된 I/O 인터럽트 태스크인 경우 다른 인터럽트 태스크 실행 중에 인터럽트 요인이 발생해도 인터럽트 요인을 저장하지 않고 실행하지 않으려면, 그(다른) 인터럽트 태스크 내에서 CLI(인터럽트 해제) 명령을 실행하여 내부에 저장된 해당 번호의 인터럽트 요인을 해제(무효화)합니다. 또한 예약 인터럽트 및 외부 인터럽트의 인터럽트 요인을 해제할 수는 없습니다.



참고

동일한 인터럽트 요인은 하나만 저장됩니다. 또한 실행 중인 인터럽트 태스크와 동일한 인터럽트 요인은 저장되지 않습니다. 따라서 복수의 인터럽트 요인이 존재하는 경우, 우선 순위가 낮은 예약 인터럽트 태스크는 마지막에 실행됩니다. 그 결과, 다음 인터럽트 요인을 빠뜨릴 가능성이 있습니다.

● 인터럽트 태스크의 사용상 주의 사항

고기능 I/O 유니트에 대해 IORF/FIORF/IORD/IOWR 명령을 실행하는 경우

고기능 I/O 유니트 사용 시, 인터럽트 태스크에서 IORF/FIORF/IORD/IOWR 명령을 실행하고 싶으면 PLC 시스템 설정에서 고기능 I/O 유니트의 주기적인 리프레시 유무 지정을 통해 지정 기기 번호와 주기적인 리프레시를 '없음'으로 설정하십시오.

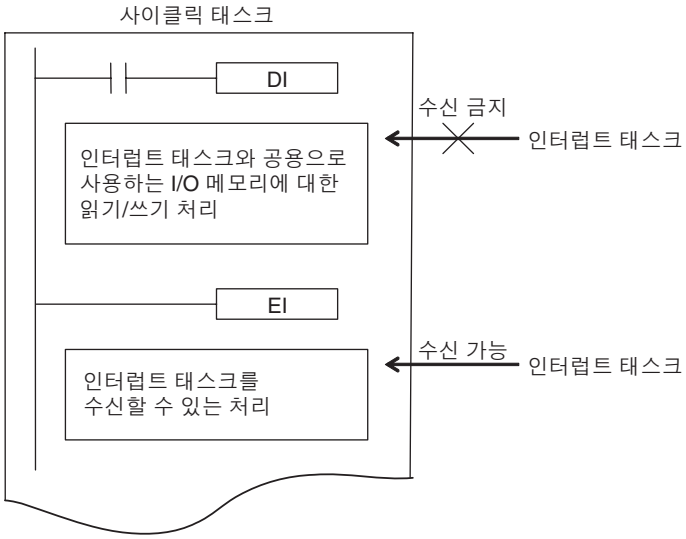
고기능 I/O 유니트를 I/O 관련 명령에 의해 리프레시하거나 주기적으로 리프레시하는 중에, 인터럽트 태스크 내에서 동일한 유니트에 대해 IORF/FIORF 명령으로 I/O 리프레시를 실행하거나 동일한 유니트에 대해 IORD/IOWR 명령으로 읽기/쓰기를 수행하려고 하면 '다중 리프레시 이상'이 발생합니다. 이때, 해당 유니트에 대한 IORF/FIORF/IORD/IOWR 명령은 실행되지 않습니다(에러 플래그(P_ER)는 ON으로 되지 않음). 단, 주기적인 I/O 리프레시는 평상시와 마찬가지로 실행됩니다.

다중 리프레시 이상이 발생한 경우 PLC 시스템 설정(CPU 유니트 설정 탭)의 '다중 리프레시 이상 검출 유무 설정'이 '검출'로 설정되어 있으면 특수 보조 릴레이의 A402.13(다중 리프레시 이상 플래그)이 ON으로 됩니다. 이와 동시에 다중 리프레시 대상인 고기능 I/O 유니트의 기기 번호가 A426CH(다중 리프레시 이상 발생 태스크 번호)에 저장됩니다.

● 사이클릭 태스크에서 특정 처리 중에 인터럽트를 금지하고 싶은 경우

다음과 같은 경우에도 강제로 처리를 중단하고 인터럽트 태스크를 실행합니다.

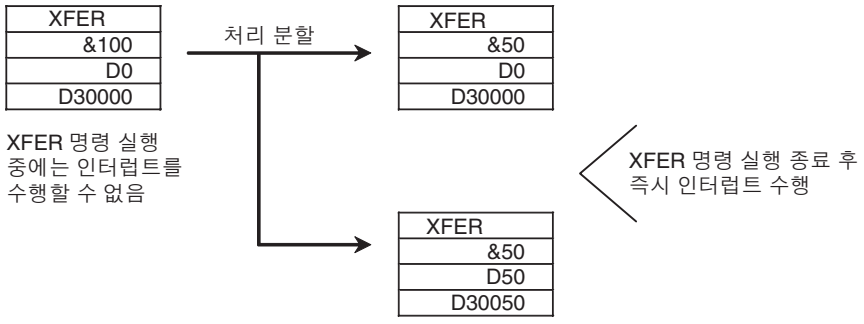
- 1개 명령 실행 중
- 기본 I/O 유니트, 고기능 I/O 유니트 리프레시 중
- 주변 서비스 중
- 사이클릭 태스크와 인터럽트 태스크 사이에 데이터의 동시성 확보가 필요한 경우
  따라서 사이클릭 태스크와 인터럽트 태스크가 동일한 I/O 메모리 주소에 액세스하는 경우, 사이클릭 태스크 내의 명령에 의한 읽기 또는 쓰기 중에 인터럽트 태스크 내의 명령에 의한 읽기 또는 쓰기 처리가 실행되면 서로 간의 데이터 동시성이 보증되지 않습니다.
  그러므로 다음과 같은 방법을 사용하여 사이클릭 태스크의 명령에 의한 액세스 중에는 인터럽트가 들어오지 않도록 하십시오.
  - 사이클릭 태스크 내의 명령에 의한 읽기 또는 쓰기 처리 직전에 DI(인터럽트 태스크 실행 금지) 명령을 통해 인터럽트 태스크의 실행을 금지합니다.
  - 해당 처리 직후 EI(인터럽트 태스크 실행 금지 해제) 명령을 통해 인터럽트 태스크 실행의 금지를 해제합니다.



단, 사이클릭 태스크 내의 특정 명령(네트워크 통신 명령, 시리얼 통신 명령 등과 같이 응답 수신 처리가 필요한 명령) 실행 중에 인터럽트가 수신되지 않도록 하려는 경우, 위와 같이 해도 데이터의 동시성이 보증되지 않으므로 주의하십시오.

참고

BCNT(비트 카운트) 명령, BSET(블록 설정) 및 XFER(블록 전송) 명령의 경우 BCNT/BSET/XFER 명령 실행 중에 인터럽트 태스크의 실행 조건이 성립되면 해당 명령의 실행이 완료되기를 기다렸다가 인터럽트 태스크가 기동됩니다. 따라서 이들 명령의 실행 중에는 인터럽트의 응답성이 저하됩니다. 이를 방지하기 위해 해당 명령이 1개 명령에서 처리하는 데이터 범위를 분할하여 몇 개의 명령으로 처리하도록 하십시오.

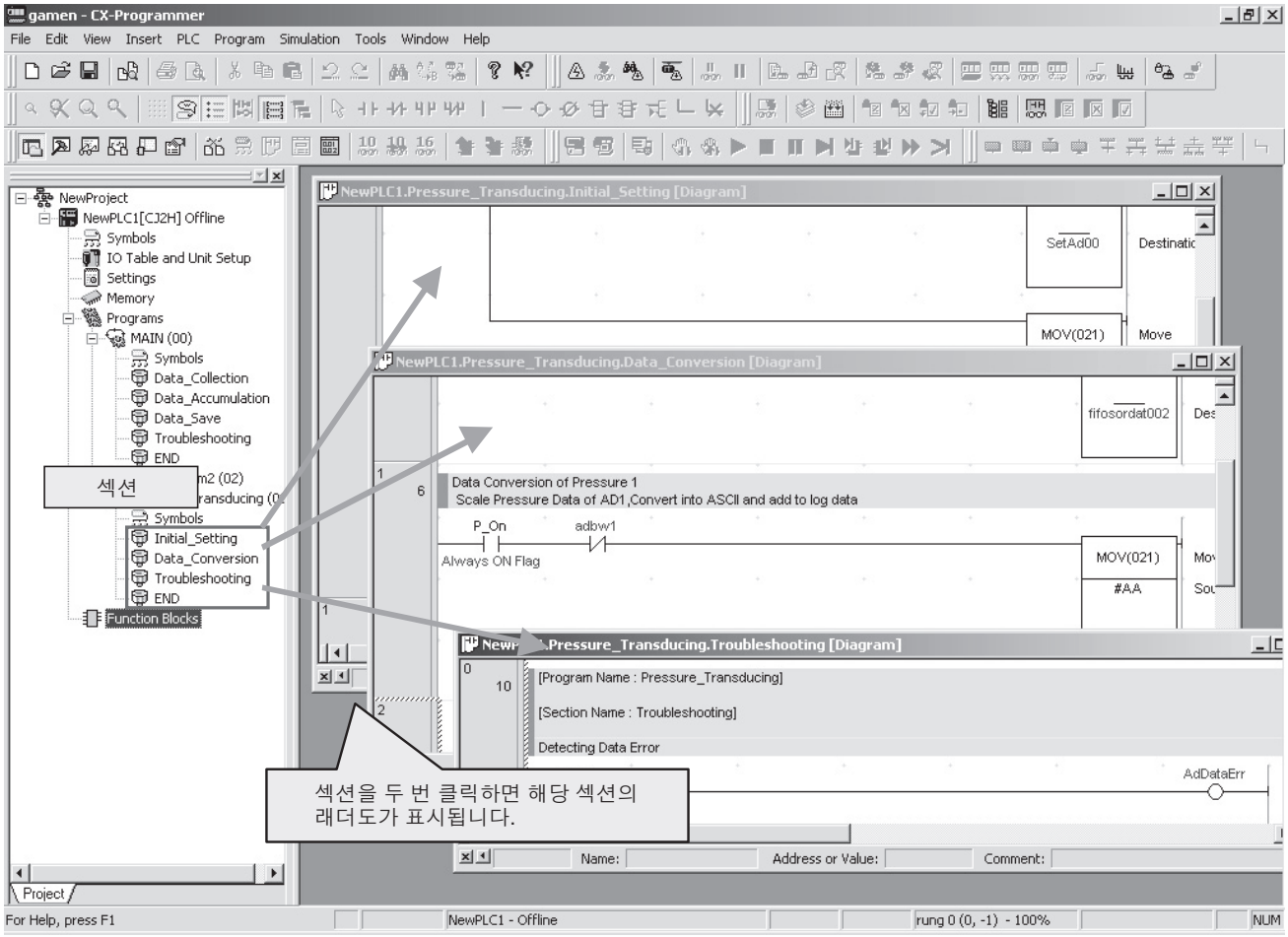


5-3 섹션

5-3-1 섹션이란?

섹션은 기능별로 분할된 프로그램 "블록"을 작성/표시하는 CX-Programmer의 기능입니다.
태스크에 할당한 프로그램은 '섹션'으로 분할할 수 있습니다.
섹션을 이용하면 프로그램의 시인성이 높아지고 편집도 간단해집니다.

- 처리 내용의 이해
  처리나 제어 내용을 나타내는 임의의 이름을 섹션에 붙일 수 있습니다.
- 간단한 편집
  마우스의 끌어다 놓기 기능을 통해 순서를 변경하고 복사하여 붙여 넣을 수 있습니다.
- 자유로운 사용
  1개 프로그램당 작성 가능한 섹션 수에는 제한이 없습니다.



● 섹션 수

섹션 수에 제한이 없습니다.
기존과 마찬가지로 1개 프로그램을 1개 섹션으로 작성하는 것도 가능합니다.

● 섹션 실행 순서

프로젝트 트리에 표시되는 순서(위→아래)가 섹션 실행 순서입니다.
따라서 프로그램 주소는 태스크(프로그램 내)에서 자동적으로 실행 순서의 번호가 됩니다. 회로 번호는 섹션 별로 0부터 시작합니다.

● 섹션명

섹션명은 '섹션□'과 같이 끝에 1부터 자동으로 번호가 붙습니다. 이 섹션명을 사용자 정의하여 변경할 수도 있습니다.

● 섹션 분할의 장점

프로그램을 섹션으로 분할했을 때의 장점은 다음과 같습니다.

- PLC에서 프로그램을 섹션 단위로 업로드(1섹션의 부분 업로드)할 수 있습니다. 프로그램의 일부분을 업로드한 후 온라인으로 편집하면 온라인 편집을 시작할 때까지의 준비 시간을 단축할 수 있습니다. 여러 섹션을 업로드할 수는 없습니다. 또한 섹션 단위의 다운로드는 불가능합니다.
- 섹션 단위의 프로젝트 트리에서 이동(실행 순서의 변경(Move Up, Down), 다른 프로그램으로 이동), 복사는 수행할 수 있습니다.

그 결과, 프로그램보다 작은 단위의 실행 모듈을 표준화할 수 있습니다.

● 섹션 관련 파일 메모리용 파일

섹션은 CX-Programmer 프로젝트(.CXP)에 포함됩니다.
또한 1개 PLC의 섹션명, 섹션 코멘트, 프로그램 코멘트가 프로그램 인덱스 파일(PROGRAMS.IDX)에 저장됩니다.



참고

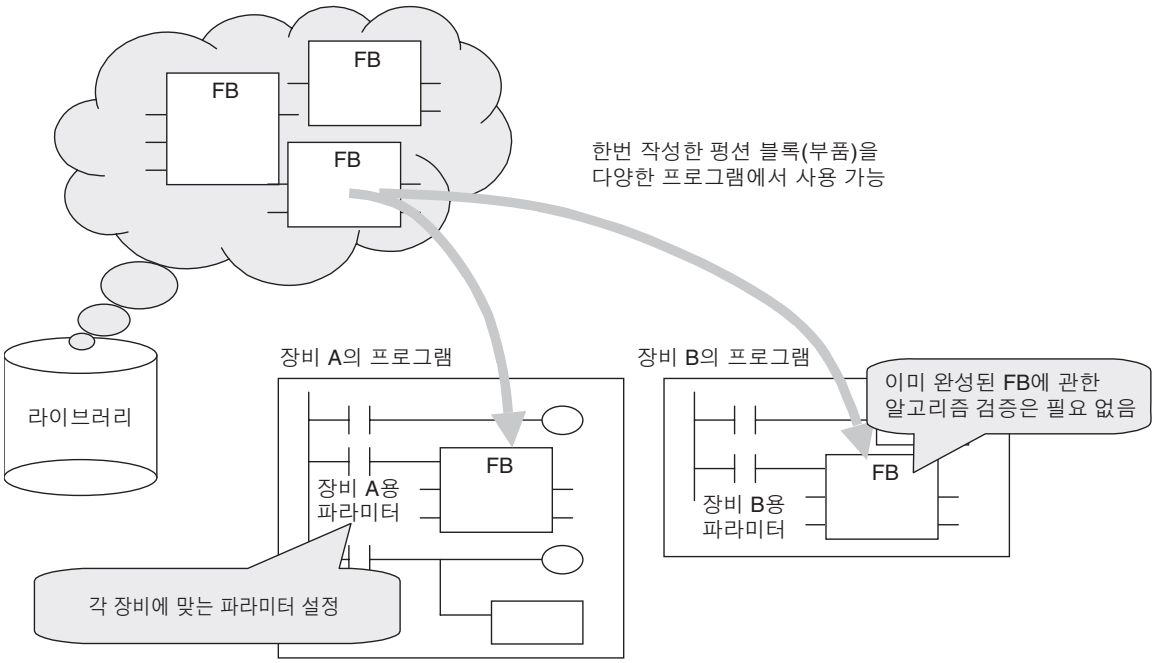
'태스크'가 프로그램의 실행/비실행을 제어하는 실행 단위인데 비해 섹션은 그 하위 단위로, 사용자 정의 프로그램 작성 시 사용하는 분할 단위입니다. 주로 프로그램을 알기 쉽게 하기 위해 분할합니다.
예를 들어, 다음과 같습니다.

- 실행/비실행에 대해 동일 처리를 수행하는 1개 태스크(프로그램) 내에서 초기 처리와 메인 처리를 섹션으로 분할합니다.
- 일반적으로 프로그램 영역과 서브루틴 영역을 섹션으로 분할합니다.

5-4 펑션 블록(FB) 기능

5-4-1 펑션 블록이란?

펑션 블록(FB)은 어떠한 정형성을 지닌 처리(기능)를 하나의 블록으로 작성한 부품입니다.
한번 작성한 부품을 라이브러리로 저장하고, 시스템에 맞춰 필요한 부품을 선택하고, 프로그램에 배치하여 입출력 파라미터를 설정하는 것만으로 사용할 수 있습니다.
뛰어난 펑션 블록은 프로그램 개발 효율을 크게 개선할 수 있습니다.
오므론의 펑션 블록은 IEC61131-3을 준수합니다.



- **펑션 블록의 장점**

부품화하여 재사용 가능

펑션 블록을 하나의 부품으로 라이브러리에 저장할 수 있습니다. 사용자는 펑션 블록의 기능을 이해하고 해당 파라미터를 설정하는 것만으로도 이용할 수 있습니다. 그 결과, 프로그램 작성 효율이 높아집니다.

디버그 시간과 비용 절감

이미 디버그를 마친 질 높은 펑션 블록 부품을 재사용하면 다시 디버그할 필요가 없으므로 디버그에 드는 시간과 비용을 줄일 수 있습니다.

데이터 보호 가능

외부에서 펑션 블록 내 변수에 직접 액세스할 수 없으므로 데이터를 보호할 수 있습니다. 따라서 예기치 못한 데이터 변경이 발생하지 않습니다.

5-4-2 펑션 블록의 특징

● 펑션 블록 내의 주소 기재

펑션 블록 내의 주소는 기본적으로 실제 주소가 아니라 '변수'로 기술되어 있습니다. 사용자는 해당 '변수'에 파라미터(주소 또는 값)를 설정하여 사용합니다. 또한 프로그램에 배치하면 '변수' 자체의 사용 주소가 CX-Programmer에서 자동으로 할당됩니다.

● 중첩 구조

펑션 블록에서 다른 펑션 블록을 호출할 수 있습니다.
최대 중첩 수는 8단계(프로그램에서의 펑션 블록 호출 포함)입니다.

● Smart FB 라이브러리

Smart FB 라이브러리는 오므론이 제작한 PLC용 유니트, FA 구성 요소의 연결 친화성 향상을 위한 펑션 블록 형식의 부품집입니다. PLC용 유니트, FA 구성 요소의 기본적인 기능을 사용하기 위한 래더 프로그램을 작성할 필요가 없습니다.

● FB 정의의 온라인 편집

운전 중에 FB 정의의 내용을 변경할 수 있기 때문에 디버그 시 단시간에 FB 정의를 수정할 수 있습니다. 또한 24시간 가동되는 설비에서도 안심하고 FB를 사용할 수 있습니다.

● FB 정의의 보호

프로젝트 파일에 할당된 펑션 블록 정의에 비밀번호를 설정하여 목적에 맞게 FB 정의를 보호하면 프로그램 노하우의 유출, 예기치 못한 변경, 개조 등에 대한 방지 대책을 세울 수 있습니다.

● FB 생성 기능

기존 PLC 프로그램을 재사용하여 간단하게 FB화할 수 있습니다.

5-4-3 펑션 블록 관련 사양

항목		사양	
펑션 블록 정의 작성	펑션 블록 정의 개수	CJ2H-CPU6□(-EIP), CJ2M-CPU□5/□4: 최대 2048 CJ2M-CPU□3/□2/□1: 최대 256	
	펑션 블록 정의명	반각 또는 전각 문자 최대 전각 32자	
	변수	변수명	반각 또는 전각 문자 최대 전각 15000자
		변수의 종류	입력 변수(Inputs), 출력 변수(Outputs), 입출력 변수(In Out), 내부 변수(Internals), 외부 변수(Externals)
		펑션 블록 정의에서 사용할 수 있는 변수(내부 변수, 외부 변수, EN, ENO 제외)의 수	펑션 블록 1개당 • 입력 변수＋입출력 변수: 64개 • 출력 변수＋입출력 변수: 64개 • 입출력 변수: 16개
		변수의 사용 주소 할당	자동 할당(할당 범위를 사용자가 설정 가능)
		실제 주소(AT) 지정	가능
		배열 지정	가능(단, 내부 변수 또는 입출력 변수로 1차원 배열만 해당)
	언어	래더 또는 ST(구조적 텍스트)로 작성 가능	
인스턴스 생성	인스턴스 수	CJ2H-CPU6□(-EIP), CJ2M-CPU□5/□4: 최대 2048 CJ2M-CPU□3/□2/□1: 최대 256	
	인스턴스명	반각 또는 전각 문자 최대 전각 15000자	
FB 프로그램 영역		20K 스텝 CJ2M CPU 유니트만 해당. 펑션 블록 정의 전용 저장 영역	
파일 저장 기능	프로젝트 파일	프로젝트 파일(.cxp/cxt)에 펑션 블록 정의 및 인스턴스 포함 가능	
	프로그램/네트워크 변수 파일	파일 메모리용 프로그램/네트워크 변수 파일(*.obj)에 펑션 블록 정의 및 인스턴스 포함 가능	
	펑션 블록 및 라이브러리 파일	1개 펑션 블록 정의＝1개 파일(.cxf)로 저장. 다른 프로젝트에서 재사용 가능	

펑션 블록의 소비 용량(스텝 수)

펑션 블록을 사용하면 다음 2가지 용량(스텝 수)을 소비합니다.
① 펑션 블록 정의의 용량
② 펑션 블록을 사용자 프로그램에서 인스턴스로 삽입할 때의 인스턴스 생성 처리 용량

따라서 ②의 삽입 수(인스턴스 수)가 많을수록 용량(스텝 수)을 많이 소비하는 점에 주의하십시오.

펑션 블록의 소비 영역

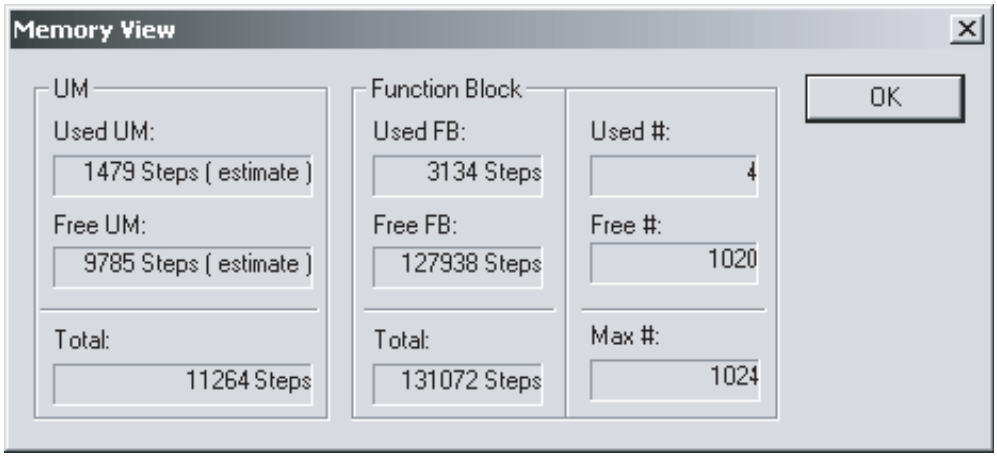
CJ2 CPU 유니트의 경우 펑션 블록 사용 시의 소비 영역이 기종에 따라 다음 표와 같이 다릅니다.
CJ2H CPU 유니트와 달리, CJ2M CPU 유니트에는 펑션 블록 정의 전용 영역으로 'FB 프로그램 영역'이 있습니다.

기종	형식	펑션 블록 사용 시의 소비 영역
CJ2H	CJ2H-CPU6□-EIP CJ2H-CPU6□	사용자 프로그램의 프로그램 용량을 소비합니다.
CJ2M	CJ2M-CPU3□ CJ2M-CPU1□	• 인스턴스 생성 처리(펑션 블록 정의의 호출과 파라미터 전송)는 사용자 프로그램 용량을 소비합니다. • 펑션 블록 정의는 펑션 블록 전용 FB 프로그램 영역을 소비합니다. 또한 초과했을 때는 사용자 프로그램의 프로그램 용량을 소비합니다.

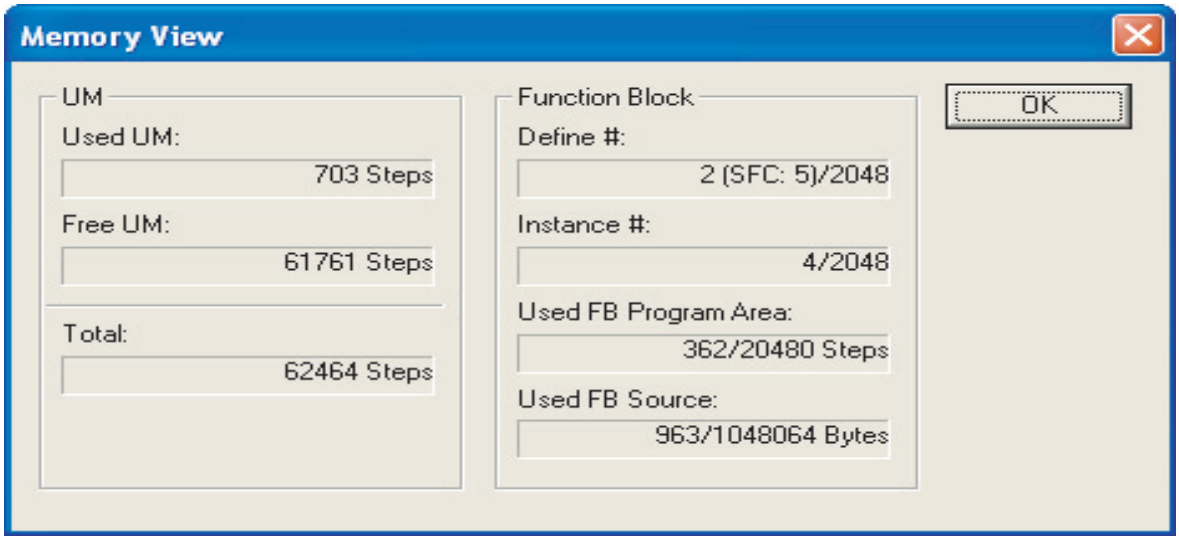
펑션 블록의 사용량 확인 방법

CX-Programmer Ver.9.0 이상의 [보기] 메뉴에서 [메모리 보기]를 선택합니다.
다음과 같은 [메모리 보기] 대화 상자가 표시됩니다.

- PLC 기종이 'CJ2M' 이외의 CS/CJ 시리즈(유니트 Ver.3.0 이상)일 때



- PLC 기종이 'CJ2M'일 때



참고

메모리 보기 대화 상자에 대한 자세한 내용은 "CX-Programmer 조작 설명서"(SBCA-337)의 '4-8 메모리 보기 기능'을 참조하십시오.

펑션 블록의 소비 용량(스텝 수) 계산 방법

펑션 블록 사용 시의 소비 용량(스텝 수)은 다음을 기준으로 계산하십시오.

소비 용량(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			사용 스텝 수
m	호출부		57스텝
n	입출력 파라미터 전송부 괄호 안은 데이터 형식	1비트(BOOL)의 입력 변수, 출력 변수인 경우	6스텝
		1워드(INT, UINT, WORD)의 입력 변수, 출력 변수인 경우	6스텝
		2워드(DINT, UDINT, DWORD, REAL)의 입력 변수, 출력 변수인 경우	6스텝
		4워드(LINT, ULINT, LWORD, LREAL)의 입력 변수, 출력 변수인 경우	12스텝
		입출력 변수인 경우	18스텝
p	펑션 블록 정의 내의 명령 스텝 수	명령 스텝 수의 합계(일반적인 사용자 프로그램과 동일)에 27스텝을 더하여 계산한 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.
또한 ST 언어로 기술된 경우 실제 스텝 수를 계산할 수 없습니다.
펑션 블록 정의 내의 명령 스텝 수는 펑션 블록 정의의 속성을 통해 확인할 수 있습니다.



참고

펑션 블록의 자세한 사양, 기술 내용, 입력 방법 등은 "CX-Programmer 조작 설명서 펑션 블록 편/구조적 텍스트편"(SBCA-338)을 참조하십시오.

5-5 변수

5-5-1 변수란?

변수

- **변수(변수 테이블)**

  I/O 메모리 영역의 주소 또는 상수를 변수로 등록하여 임의의 문자열로 지정할 수 있습니다.
  변수는 CX-Programmer의 변수 테이블에 등록됩니다.
  변수명(문자열)을 사용하여 프로그래밍하면 주소에 상관 없이 실제 데이터의 이름으로 프로그래밍할 수 있습니다.
  변수 테이블은 사용자 프로그램, I/O 테이블 등의 다른 설정 데이터와 함께 CX-Programmer의 프로젝트 파일(.CXP)에 저장됩니다.

- **변수 지정 필요 여부**

  변수 사용은 펑션 블록 내부/외부 및 프로그램 언어에 따라 임의로 결정해도 되는 경우(사용하지 않아도 되는 경우)와 반드시 필요한 경우가 있습니다.

프로그램 요소	프로그램 언어	변수 지정 필요 여부
프로그램측 (펑션 블록 외부)	래더 언어	임의(사용하지 않아도 됨)
	ST 언어	필수
	SFC 언어	임의(사용하지 않아도 됨)
펑션 블록 내부	래더 언어	필수
	ST 언어	

5-5-2 변수의 종류

변수에는 다음과 같은 종류가 있습니다.

프로그램측 변수

● 변수의 종류

- 글로벌 변수

  대상 PLC 내의 모든 프로그램(태스크에 할당하는 단위)에서 액세스할 수 있는 변수입니다.
- 로컬 변수

  하나의 프로그램(태스크에 할당하는 단위)에서만 액세스할 수 있는 변수입니다.
- 네트워크 변수(CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당)

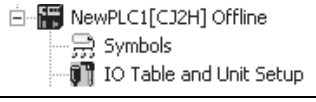
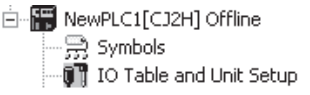
  자체 CPU 유니트의 외부에서도 변수의 이름을 사용하여 CPU 유니트 내의 I/O 메모리에 액세스할 수 있는 변수입니다. (자체 CPU 유니트에서도 액세스할 수 있습니다.)

● 주소 할당

프로그램측 변수의 주소 할당은 다음과 같은 방법으로 수행합니다.

- 사용자 개별 지정
- CX-Programmer로 자동 할당

  CX-Programmer의 [PLC] 메뉴에서 [메모리 할당]−[주소 자동 할당]을 선택하여 사용 영역을 설정합니다.

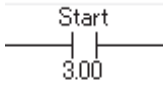
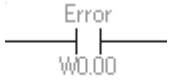
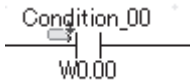
종류	명칭	CX-Programmer 프로젝트 트리에서	유효 범위			변수명이 없는 주소 + I/O 코멘트
			네트워크에서 변수명으로 액세스	자체 프로그램 이외에서 액세스	자체 프로그램에서 액세스	
프로그램측 변수	글로벌 변수	PLC의 트리 	불가능	가능	가능	등록 가능
	로컬 변수	프로그램의 트리 		불가능	가능	등록 불가능
	네트워크 변수	PLC의 트리: 글로벌 변수 테이블 	가능	가능	가능	등록 불가능

(주) 글로벌/로컬은 단순히 '변수명'의 적용 범위를 나타내며, 주소 적용 범위와는 관계 없습니다.
따라서 다음의 경우에도 에러가 아니라 주의가 되므로 사용자 프로그램을 전송할 수 있습니다.

- 로컬 변수 간에 동일한 주소
- 글로벌 변수와 로컬 변수 간에 동일한 주소

참고

CX-Programmer 프로그램에서는 글로벌 변수, 로컬 변수, 네트워크 변수를 다음과 같이 문자색과 태그를 표시하는 아이콘으로 식별할 수 있습니다.

종류	표시색	예(기본색)
글로벌 변수	흑색(고정)	
로컬 변수	청색(기본색) [도구]−[옵션]의 [모양] 탭에서 [로컬 심볼] 항목을 선택하여 표시색을 변경할 수 있습니다.	
네트워크 변수	흑색(고정) 이름 아래에 태그를 나타내는 아이콘이 표시됩니다.	

펑션 블록 내 변수

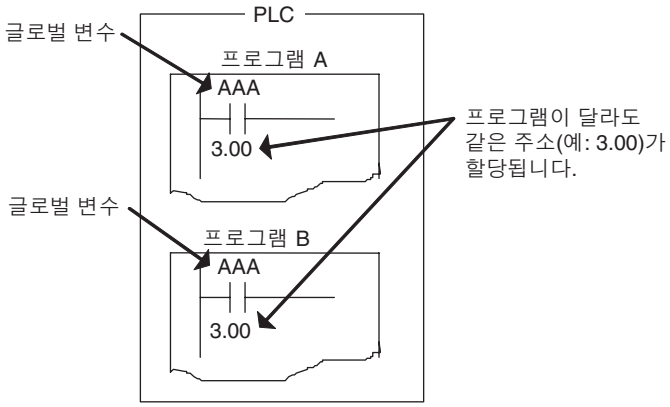
펑션 블록 내 알고리즘에서만 액세스할 수 있는 변수입니다.

명칭	CX-Programmer 프로젝트 트리에서	유효 범위		변수명이 없는 주소 + I/O 코멘트
		네트워크에서 변수명으로 액세스	펑션 블록 외부에서 액세스	
펑션 블록 내에서 사용하는 변수	펑션 블록 내의 정의 	불가능 (네트워크 변수는 외부 변수에 등록할 수 없음)	불가능	등록 불가능

펑션 블록 외부에서 사용하는 변수와 일부 사양이 다릅니다.
주소에 대한 할당은 CX-Programmer에서 자동 할당을 통해 수행됩니다. CX-Programmer의 [PLC] 메뉴에서 [메모리 할당]−[펑션 블록/SFC 메모리]를 선택하여 사용 영역을 설정합니다.

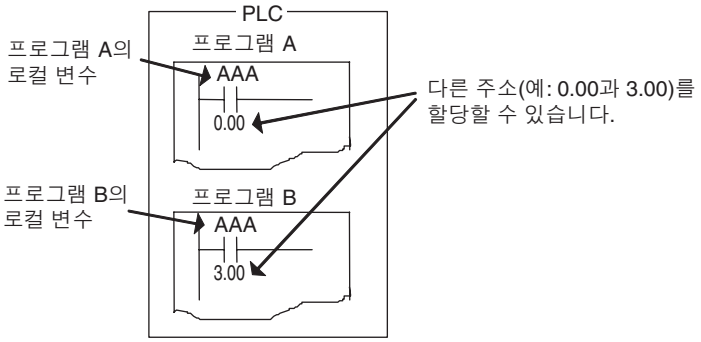
5-5-3 글로벌 변수

글로벌 변수는 대상 PLC 내의 모든 프로그램에서 사용 가능한 변수(문자)입니다. 따라서 예를 들어 AAA라는 변수는 대상 PLC 내 모든 프로그램에서 동일한 주소가 됩니다.
예: "AAA"라는 변수를 글로벌 변수로 설정하면 프로그램이 달라도 동일한 주소(예: 3.00)가 할당됩니다.



5-5-4 로컬 변수

해당 프로그램(태스크에 할당하는 단위)에서만 액세스할 수 있는 변수입니다.
로컬 변수는 하나의 프로그램 내에서만 사용할 수 있는 변수(문자)입니다.
로컬 변수는 각 프로그램별로 개별적으로 지정하므로 프로그램이 다르면 같은 변수라도 다른 변수로 취급됩니다.
예를 들어 AAA라는 로컬 변수는 해당 프로그램에서만 동일한 주소이며, 다른 프로그램에서는 다른 주소를 할당할 수 있습니다.
예: "AAA"라는 변수를 로컬 변수로 설정하면 같은 "AAA" 변수라도 프로그램이 다른 경우 다른 주소(예: 0.00과 3.00)를 할당할 수 있습니다.



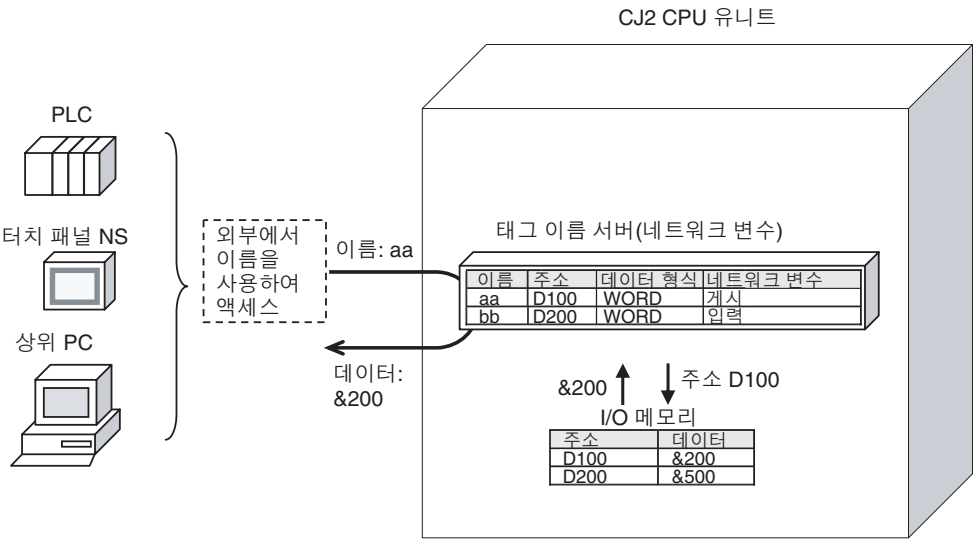
5-5-5 네트워크 변수(CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당)

개요

외부에서 변수명을 사용하여 CPU 유니트의 I/O 메모리에 액세스할 수 있는 변수입니다.
CJ2 CPU 유니트에는 네트워크 변수를 실제 I/O 주소로 변환할 수 있는 '태그 이름 서버'가 내부에 있으므로 외부에서 변수명으로 CPU 유니트의 I/O 메모리에 액세스할 수 있습니다.
다음 용도로 사용할 수 있습니다.

- EtherNet/IP의 다른 PLC와 태그 데이터 링크
- 상위, 표시기 등과 같은 외부에서 태그 액세스

네트워크 변수는 CX-Programmer의 글로벌 변수 테이블에서 설정합니다.



참고

태그 데이터 링크 실현 방법은 "EtherNet/IP 유니트 사용 설명서"(SBCD-342)를 참조하십시오.

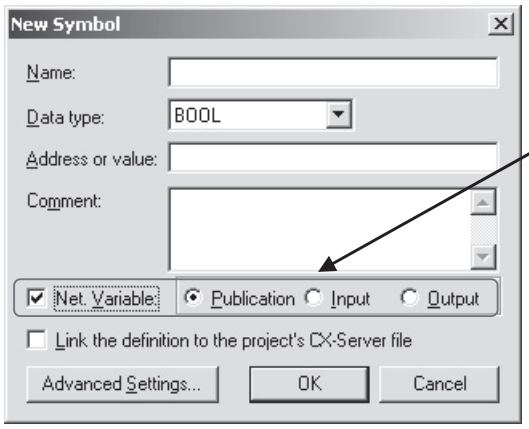
네트워크 변수 설정 방법

네트워크 변수는 다음을 수행하여 설정합니다.

1 **글로벌 변수 테이블에서 마우스 오른쪽 단추를 클릭하여 [심볼 삽입]을 선택합니다.**
다음과 같은 [새 심볼] 대화 상자가 표시됩니다.

(주) 네트워크 변수는 로컬 변수 테이블에서 지정할 수 없습니다.

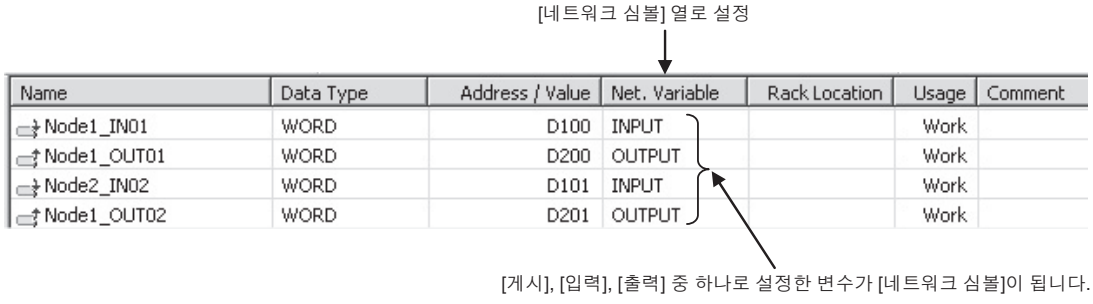
2 **[네트워크 변수] 확인란을 선택한 후 [게시], [입력], [출력] 중 하나를 선택하고 [확인] 버튼을 클릭합니다.**



네트워크 변수명의 경우 문자 수는 최대 반각 48자까지 사용할 수 있습니다. 또한 대문자와 소문자를 구별하지 않습니다.

분류	내용	글로벌 변수 테이블의 [네트워크 심볼] 열	내용
네트워크 변수	외부에서 변수명으로 액세스 가능한 변수	공개만	• CIP 메시지 통신을 통해 외부에서 액세스 가능한 변수 • 태그 데이터 링크는 불가능
		입력	• CIP 메시지 통신 또는 태그 데이터 링크를 통해 외부에서 액세스 가능한 변수 • 태그 데이터 링크 시에는 데이터 입력용(다른 CPU 유니트→자체 CPU 유니트) 변수가 됨
		출력	• CIP 메시지 통신 또는 태그 데이터 링크를 통해 외부에서 액세스 가능한 변수 • 태그 데이터 링크 시에는 데이터 출력용(자체 CPU 유니트→다른 CPU 유니트) 변수가 됨

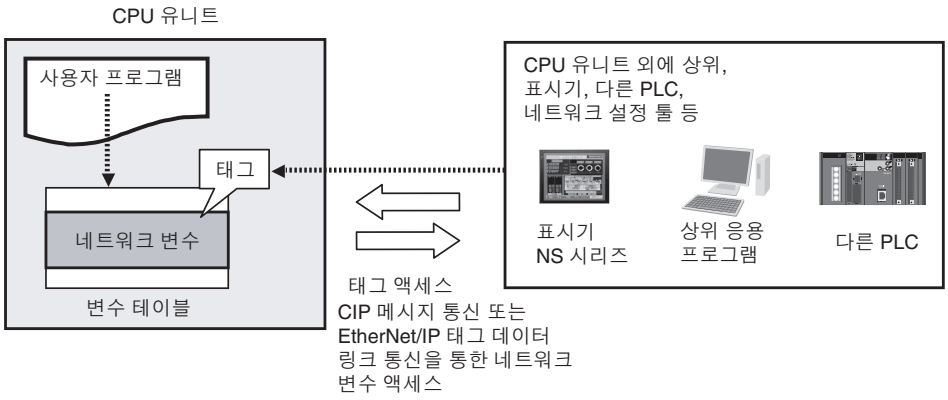
● 글로벌 변수 테이블의 [네트워크 심볼] 열



네트워크 변수의 사용 조건 및 사용 방법

● 네트워크 변수의 외부에서 사용하는 방법

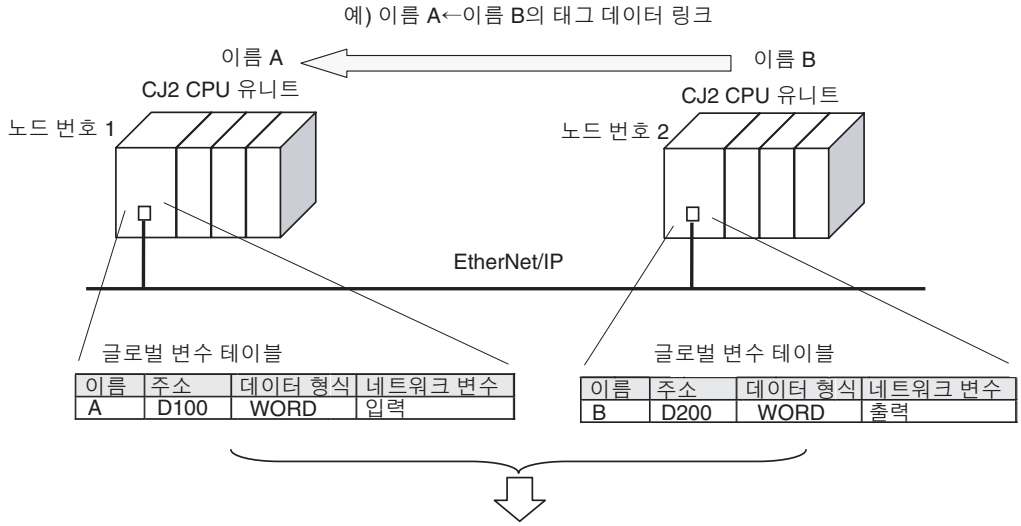
CPU 유니트 내에서 '네트워크 변수'로 설정한 변수는 PLC 외부(상위, 표시기, 네트워크 설정 툴 또는 다른 PLC 등)에서 '태그'로 인식됩니다.



네트워크 변수의 용도

● EtherNet/IP 태그 데이터 링크에서 태그로 사용

EtherNet/IP 태그 데이터 링크 통신에서 상대 PLC의 데이터를 이름으로 지정할 수 있습니다.
글로벌 변수 테이블의 [네트워크 심볼] 열에서 [입력] 또는 [출력]을 설정합니다.

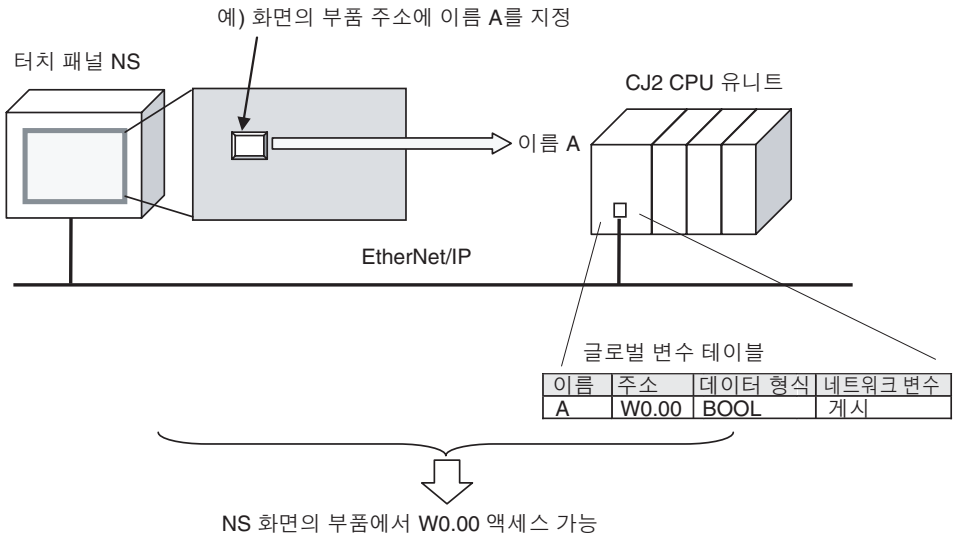


이름	주소	데이터 형식	네트워크 변수
A	D100	WORD	입력

이름	주소	데이터 형식	네트워크 변수
B	D200	WORD	출력

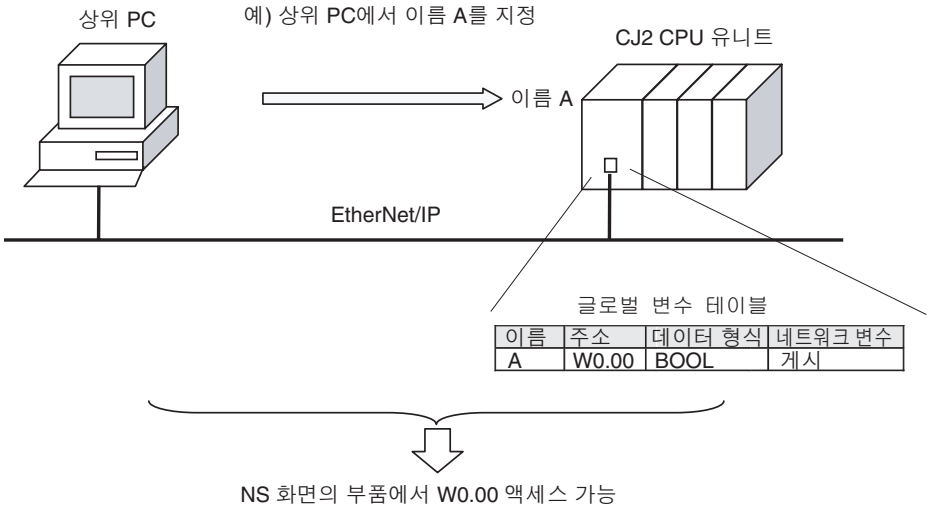
● 터치 패널 NS와의 통신에서 태그로 사용

터치 패널 NS 화면에서 부품으로 상대 PLC의 데이터를 이름으로 지정할 수 있습니다.
글로벌 변수 테이블의 [네트워크 심볼]에서 [게시]를 설정합니다.



● 상위 PC의 CIP 메시지 통신에서 태그로 사용

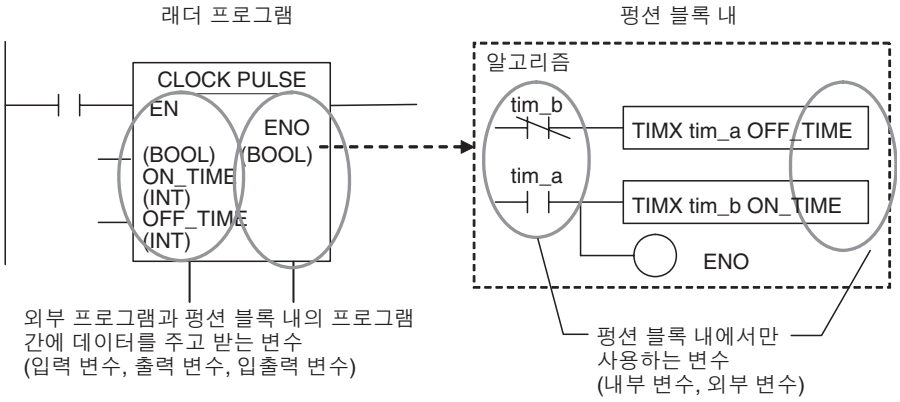
상위 PC에서 상대 PLC의 데이터를 이름으로 지정할 수 있습니다.
글로벌 변수 테이블의 [네트워크 심볼] 열에서 [공개만]을 설정합니다.



5-5-6 펑션 블록 내 변수

펑션 블록 내의 프로그램은 실제 주소가 아니라 변수명으로 모두 기술합니다.
펑션 블록 내 변수는 변수의 종류나 사양이 펑션 블록 외부의 변수와 다릅니다.
펑션 블록용 변수에 대한 자세한 내용은 "CX-Programmer 조작 설명서 펑션 블록편/구조적 텍스트편" (SBCA-338)을 참조하십시오.

[펑션 블록의 변수 예]



5-5-7 변수의 데이터 형식

변수의 경우 확보(할당)된 주소 내의 데이터 형식을 지정할 수 있습니다.

변수에 설정할 수 있는 데이터 형식의 종류

펑션 블록 외부와 펑션 블록 내에 지정할 수 있는 데이터 형식이 서로 다릅니다.
각각 다음과 같은 데이터 형식을 지정할 수 있습니다.

○: 있음, ×: 없음

데이터 형식의 종류	프로그램측 변수	펑션 블록 내 변수
BOOL(비트)	○	○
UINT(부호 없는 1채널 BIN)	○	○
UDINT(부호 없는 2채널 BIN)	○	○
ULINT(부호 없는 4채널 BIN)	○	○
INT(부호 있는 1채널 BIN)	○	○
DINT(부호 있는 2채널 BIN)	○	○
LINT(부호 있는 4채널 BIN)	○	○
UINT BCD(부호 없는 1채널 BCD) (*1)	○	×
UDINT BCD(부호 없는 2채널 BCD) (*1)	○	×
ULINT BCD(부호 없는 4채널 BCD) (*1)	○	×
REAL(부동 소수점 2채널)	○	○
LREAL(부동 소수점 4채널)	○	○
CHANNEL(채널) (*1)	○	×
NUMBER(상수 또는 번호) (*2)	○	×
WORD(16진수 1채널)	○	○
DWORD(16진수 2채널)	○	○
LWORD(16진수 4채널)	○	○
STRING(문자열: 문자 수 1~255의 ASCII 코드)	○	○
TIMER(타이머)	×	○
COUNTER(카운터)	×	○
사용자 정의 형식(구조체)	○	○(*3)

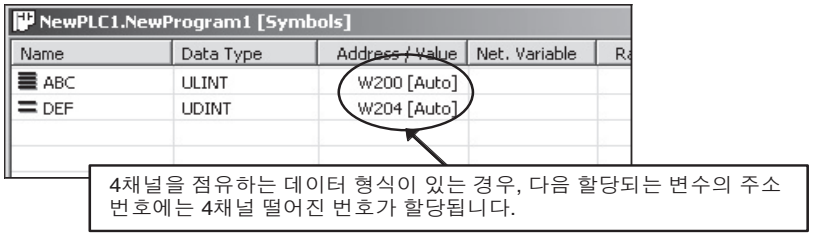
*1 ST 프로그램에서 사용하는 경우 각각 다음과 같은 데이터 형식으로 인식됩니다.
- UINT BCD → WORD로 인식됩니다.
- UDINT BCD → DWORD로 인식됩니다.
- ULINT BCD → LWORD로 인식됩니다.
- CHANNEL → WORD로 인식됩니다.

*2 ST 프로그램에서는 사용할 수 없으며, 프로그램 에러가 발생합니다.
*3 래더 언어의 펑션 블록 내 변수만 해당

데이터 형식의 용도

변수에 설정한 데이터 형식은 다음과 같은 기능으로 사용됩니다.

- **주소 자동 할당 기능 사용 시**
  CX-Programmer의 주소 자동 할당 기능을 사용하면 변수에 설정된 데이터 형식에 맞게 주소 번호가 할당됩니다.
  예: ULINT(부호 없는 4채널 BIN) 데이터 형식을 설정한 경우



배열 지정

변수 속성으로 배열(1차원 배열만)을 지정할 수 있습니다.
STRING 형식과 NUMBER 형식을 제외한 모든 데이터 형식으로 지정할 수 있습니다.

동일한 데이터 속성을 가진 여러 개의 변수를 사용하는 경우 이들 변수를 하나의 그룹으로 관리하고 싶을 때 배열을 사용합니다.
배열은 다음과 같이 지정합니다.

1 **[새 심볼] 또는 [심볼 편집] 대화 상자에서 변수명을 입력한 후 [고급 설정] 버튼을 클릭합니다.**
[고급 설정] 대화 상자가 표시됩니다.

2 **배열 설정 필드의 배열 변수를 확인하고 최대 요소 수를 입력합니다.**

배열 변수를 기술할 때는 변수명 다음에 첨자를 반각 [ ]로 감싸서 지정합니다.
예) 변수명이 PV이고 최대 요소 수＝3인 경우 명령의 피연산자 내에서 PV[0], PV[1], PV[2]라고 지정합니다.
첨자는 0부터 시작하는 요소 번호로 직접 지정합니다.
요소 번호를 직접 입력하는 방법 이외에 변수명 또는 주소를 입력하여 요소 번호를 간접 지정할 수도 있습니다.

⚠ 주 의

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.



첨자에 채널 주소 또는 변수명을 지정한 배열 변수는 매회 리프레시를 지정한 명령어의 피연산자에 사용할 수 없습니다.

사용자 정의 형식(구조체)

CX-Programmer Ver.9.0 이상을 사용하면 CJ2 CPU 유니트에서 사용자 정의 데이터 형식으로 사용자가 구조체를 작성하여 사용할 수 있습니다.

● 구조체란?

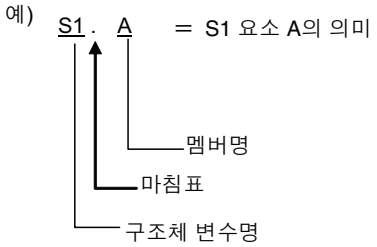
구조체는 서로 다른 데이터 형식을 하나의 그룹으로 만들어 사용자가 정의한 데이터 형식입니다. 이 데이터 형식에는 임의의 이름을 붙일 수 있습니다.
작성한 사용자 정의 데이터 형식을 사용하는 변수('구조체 변수') 이름과 해당 그룹 내의 변수('멤버') 이름을 하나 지정합니다.

참고

- 마찬가지로 여러 데이터를 묶어서 사용하는 방법으로 배열이 있습니다.
  단, 배열은 같은 형식의 데이터를 선두부터의 오프셋 위치에서 사용한다는 점에서 구조체와 다릅니다. 구조체에서는 서로 다른 형식(같은 형식도 관계 없음)의 데이터를 멤버명으로 지정합니다. 또한 CX-Programmer의 경우 구조체의 데이터 형식에 임의의 이름을 붙일 수 있습니다.
- 구조체 변수 자체를 배열로 만들 수 있습니다.
- 멤버에 배열 변수를 사용할 수도 있습니다.

● 래더 프로그램에서 구조체 변수의 표기 및 입력

래더 프로그램에서는 구조체 변수명과 멤버명 사이에 마침표(.)를 넣어 표기합니다.

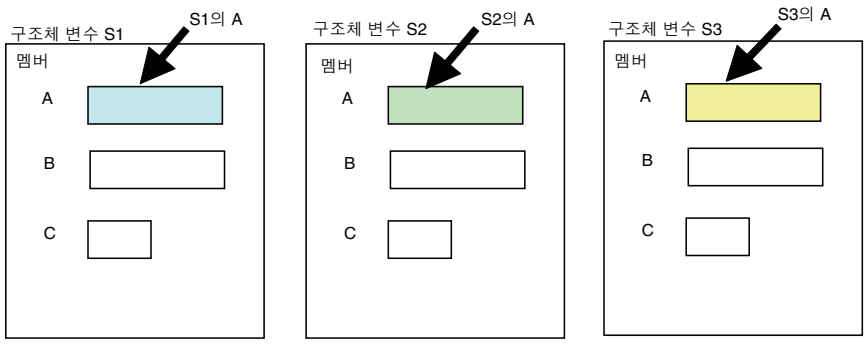


● 구조체의 용도

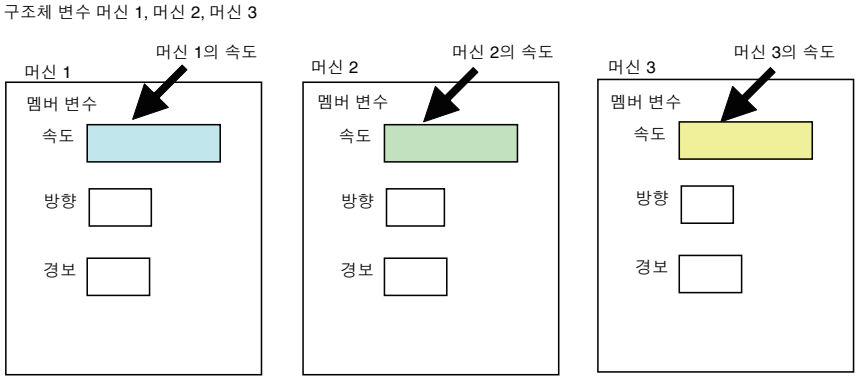
데이터가 그룹화된 경우 특정 데이터를 지정할 때, 어떤 그룹의 요소라는 형태로 지정할 수 있습니다. 즉, 메인-서브와 같은 형태의 계층적인 데이터 지정이 가능합니다.

• 배열과 조합하지 않는 경우

예) 구조체 변수 S1, S2, S3의 멤버(요소)가 A, B, C인 경우



예: 머신별 요소(예: 속도)를 지정하는 경우 등에 사용합니다.

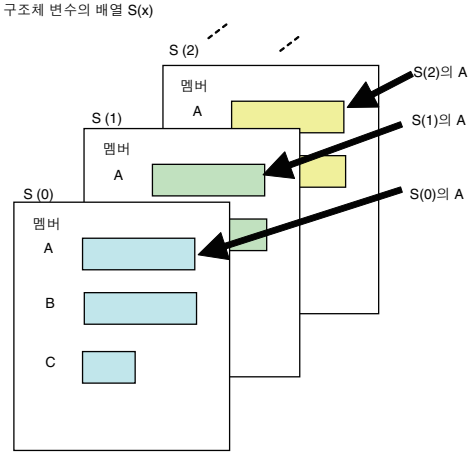


- **배열 지정과 조합하는 경우**

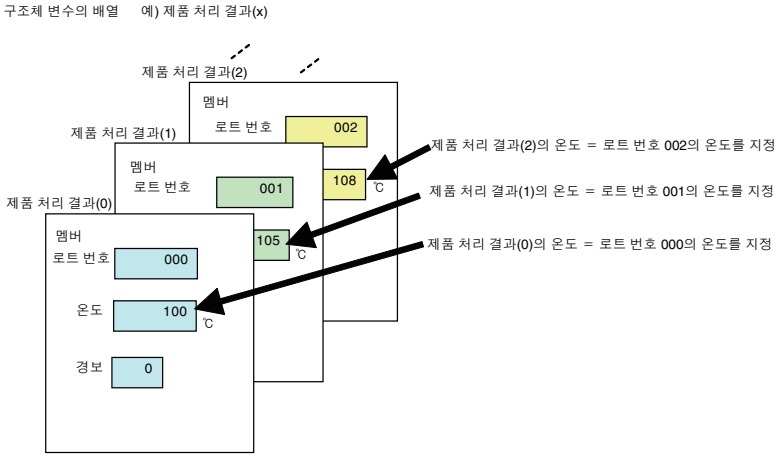
구조체 변수를 배열로 지정하는 경우

제품별 레시피 데이터 등과 같이 동일 형태 데이터가 많은 경우 구조체 변수 자체를 배열로 지정합니다.

데이터베이스를 작성할 때 사용합니다. 이 경우, 구조체 변수가 하나의 레코드가 되고 각 멤버는 필드가 됩니다.

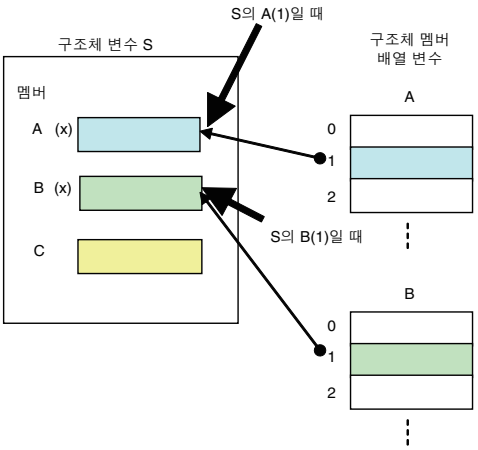


예: 로트 번호별 처리 결과를 지정하는 경우 등에 사용합니다.

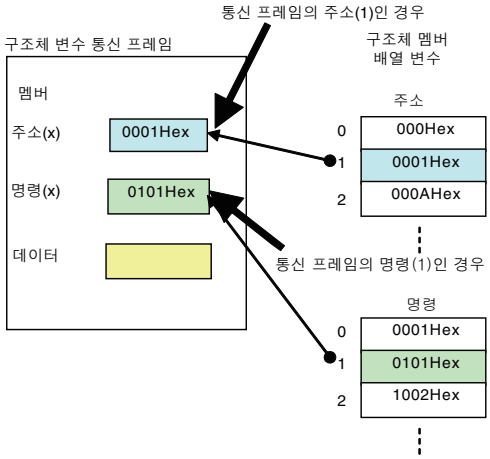


멤버에 배열 변수를 사용하는 경우

일정한 구조로 구성된 데이터이고 그 요소가 라이브러리처럼 여러 개 존재하는 경우, 구조체로 만들어 멤버를 배열로 지정할 수 있습니다. 이때 각 멤버는 배열에서 임의로 선택합니다.
데이터를 라이브러리에서 가져올 때 사용합니다.



예: 통신 프레임을 라이브러리에서 선택하여 작성하는 경우 등에 사용합니다.



구조체의 사양 및 작성 방법에 대한 자세한 내용은 📖 "CX-Programmer 조작 설명서 펑션 블록편/구조적 텍스트편"(SBCA-338)을 참조하십시오.

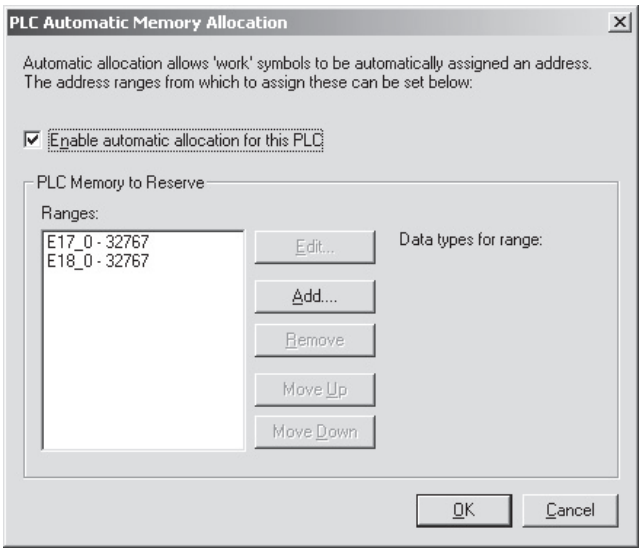
5-5-8 변수의 주소 자동 할당

프로그램을 작성할 때 변수를 자동으로 주소에 할당할 수 있습니다.

CX-Programmer의 [PLC]－[메모리 할당]－[주소 자동 할당] 메뉴에서 자동 할당할 데이터의 범위와 형식을 설정할 수 있습니다. CJ2 CPU 유니트에서는 'CIO', 'AR', 'HR', 'WR', 'DM', 'EM'으로 주소를 자동 할당할 수 있습니다.

변수의 주소를 자동 할당하는 경우에는 EM 뱅크 번호가 큰 뱅크(최하위 뱅크)에서 필요한 뱅크 수만큼 할당 영역으로 사용할 것을 권장합니다.

예) CPU68(E00~E18)의 E17~E18을 주소 자동 할당 영역으로 설정하는 경우



자동 할당 영역 설정에 대한 자세한 내용은 "CX-Programmer 조작 설명서"(SBCA-337)를 참조하십시오.

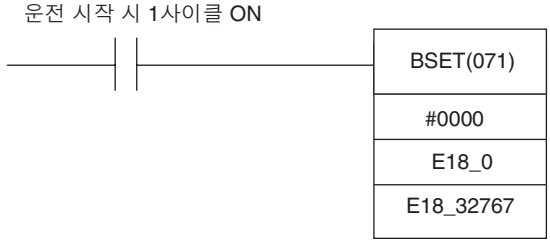
참고

- CJ2H CPU 유니트의 경우 주소 자동 할당 영역으로 EM 뱅크를 설정하면 특정 뱅크에서 강제 세트/리셋을 사용할 수 있습니다. 강제 세트/리셋이 가능한 EM 뱅크 번호는 '6-14 확장 데이터 메모리'를 참조하십시오.
- 구조체 변수는 자동으로 주소가 할당되지 않습니다.

안전상의 중요 사항

EM 영역은 전원 재투입 또는 동작 모드 변경 시에도 데이터를 유지하므로 BOOL 형식 등으로 지정하여 출력 접점에 사용하는 경우는 주의하십시오. 또한 필요에 따라 전원 투입 시 영역을 0으로 클리어하는 프로그램을 사용하십시오.

예) EM 뱅크 18이 주소 자동 할당 영역으로 설정되어 있고 전원 투입 시 뱅크 18을 0으로 클리어하는 경우

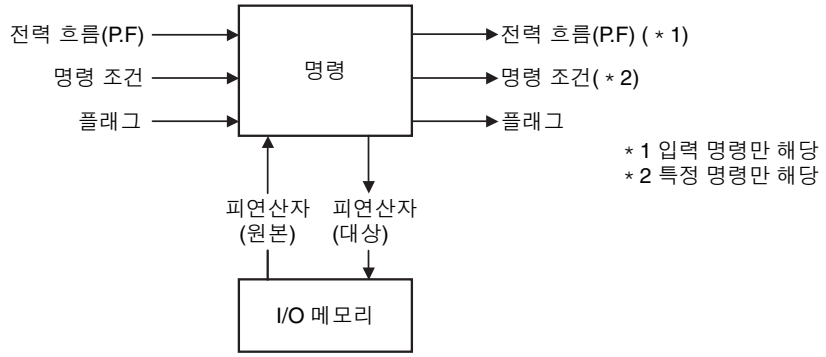


5-6 명령

5-6-1 명령의 기본 지식

명령의 구조

프로그램은 명령의 조합으로 구성됩니다.
명령은 다음과 같은 입력 및 출력의 구조로 이루어집니다.

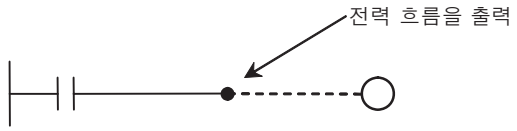


- **전원 흐름(P.F.)**

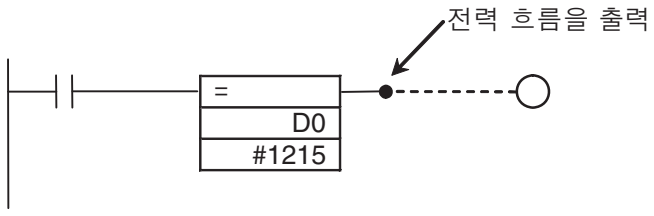
  전원 흐름이란 평상시에 프로그램을 실행할 때 명령 기능의 실행/비실행을 제어 하기 위한 정보입니다.
  래더 프로그램에서는 모선에서의 신호 흐름을 나타냅니다.

a) 입력부를 구성하는 입력계 명령의 경우

- 논리 시작을 나타내는 LD형 명령: 전원 흐름을 출력합니다.

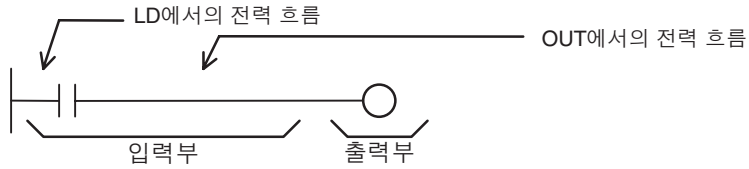


- 다음 단계 연결형 명령: 입력 조건으로 전원 흐름을 입력하고, 다음 단계 연결형 명령 또는 출력 명령에 전원 흐름을 출력합니다.



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	설정하는 명령	해제하는 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 비실행으로 함)	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.

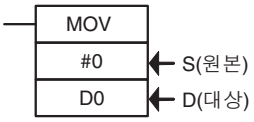
입력 플래그		출력 플래그	
플래그명	내용	플래그명	내용
캐리(CY) 플래그	데이터 이동 명령, 가감산 명령에서 명시적이지 않은 피연산자로 사용되는 조건 플래그의 일종입니다.	조건 플래그	명령 실행 결과에 따라 갱신되는 플래그 및 상시 ON/상시 OFF 플래그의 총칭입니다. 사용자 프로그램에서는 주소 대신 P_On, P_Off, P_ER, P_CY, P_EQ 등의 라벨을 사용하여 지정합니다.
명령 전용 입력 플래그	FPD 명령용 티칭 플래그, 네트워크 통신 실행 가능 플래그 등	명령 전용 출력 플래그	메모리 카드 명령 실행 중 플래그,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

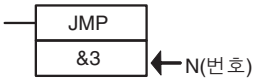
● 피연산자

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.

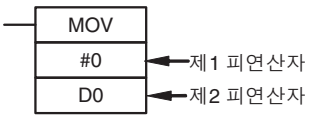
예:





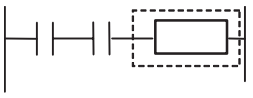
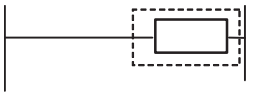
피연산자 분류		피연산자 기호	내용	
원본 피연산자	데이터를 읽어올 주소 또는 데이터 자신을 저장합니다.	S	원본 피연산자	제어 데이터(C) 이외의 원본 피연산자
		C	제어 데이터	원본 피연산자 중에 특히 1비트 또는 복수의 비트에 의해 여러 가지 의미를 갖는 데이터의 복합체
대상 피연산자	데이터를 기술할 주소를 저장합니다.	D	—	
번호	점프, 서브루틴 등, 각각의 명령에 특정 번호가 지정된 경우 해당번호를 저장합니다.	N	—	

(주) 기술된 위치에 따라 위에서부터 제1피연산자, 제2피연산자 등으로 부릅니다.



명령의 배치 위치와 입력 조건

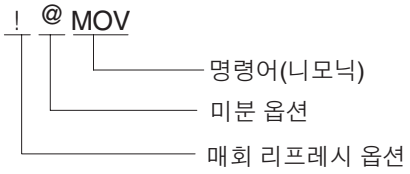
명령에는 다음과 같이 배치 가능한 위치가 있습니다.
또한 명령에는 입력 조건이 필요한 명령과 불필요한 명령이 있습니다.
명령별 내용에 대해서는 '부록-1 명령 목록'을 참조하십시오.

분류		배치 가능 위치	입력 조건 필요/필요 없음	그림	명령 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 등의 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	@	명령을 입력 기동 미분형 명령으로 수행합니다.
	종료	%	명령을 입력 종료 미분형 명령으로 수행합니다.
매회 리프레시 옵션		!	명령 실행 시, 피연산자로 지정한 입출력 영역의 데이터를 I/O 리프레시합니다.



실행 조건

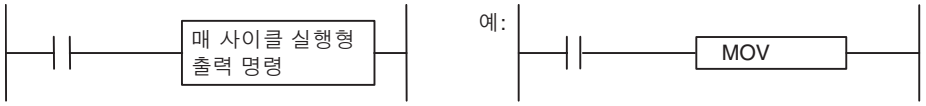
기본 명령이나 응용 명령에는 다음의 두 타입이 있습니다.

- 매 사이클 실행형
- 입력 미분형(1사이클 실행)

● 매 사이클 실행형 명령(옵션이 없는 일반 명령)

출력계 명령(입력 조건이 필요한 명령만 해당)의 경우

입력 조건이 성립(ON 또는 OFF)하는 동안 사이클마다 실행하는 명령입니다.



입력계 명령(논리 시작형 및 다음 단계 연결형 명령)인 경우

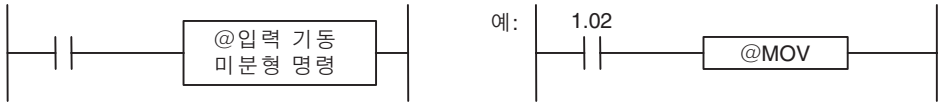
사이클마다 접점 읽기/비교/비트 테스트 등의 처리를 실행하고, 결과가 ON이면 입력 조건(전원 흐름)을 출력하는 명령입니다.



● 입력 미분형 명령

입력 기동 미분형 명령(명령어 앞에 '@' 옵션이 붙는 명령)

- 출력계 명령인 경우
  입력 조건의 기동(OFF→ON) 시에만 실행되는 명령이며 다음 사이클에서는 실행되지 않습니다.



1.02가 OFF→ON일 때 1회만 MOV 명령을 실행합니다.

- 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 번 사이클에서 OFF가 됩니다.



1.03의 입력 접점이 OFF→ON일 때 입력 조건은 ON이 되고, 1사이클 후에 OFF가 됩니다.

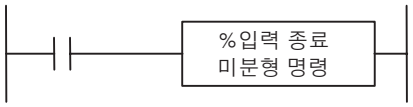
- 입력계 명령(논리 시작형 및 다음 단계 연결형 명령)인 경우
  사이클마다 접점 읽기/비교/비트 테스트 등의 처리를 실행하며, 결과가 OFF→ON으로 바뀌는 경우를 제외하고 입력 조건(전원 흐름)을 출력하는 명령입니다.

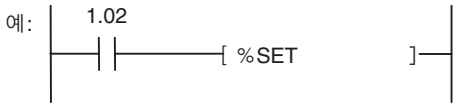


1.03의 입력 접점이 OFF→ON일 때 OFF가 되고, 1사이클 후에 ON이 됩니다.

입력 종료 미분형 명령(명령어 앞에 '%' 옵션이 붙는 명령)

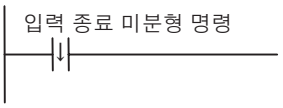
- 출력계 명령인 경우
  입력 조건의 종료(OFF→ON) 시에만 실행되는 명령이며, 다음 사이클에서는 실행되지 않습니다.

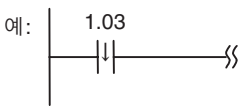




1.02가 ON→OFF일 때 1회만 SET 명령을
실행합니다.

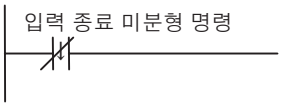
- 입력계 명령(논리 시작형 및 다음 단계 연결형 명령)인 경우
  사이클마다 접점 읽기/비교/비트 테스트 등의 처리를 실행하고, 결과가 ON→OFF로 변하면 입력 조건(전원 흐름)을 출력하는 명령입니다.
  출력하는 입력 조건은(1사이클 ON한 후) 다음 번 사이클에서 OFF가 됩니다.

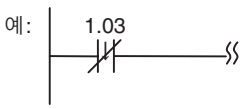




1.03의 입력 접점이 ON→OFF일 때 ON이 되고,
1사이클 후에 OFF가 됩니다.

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

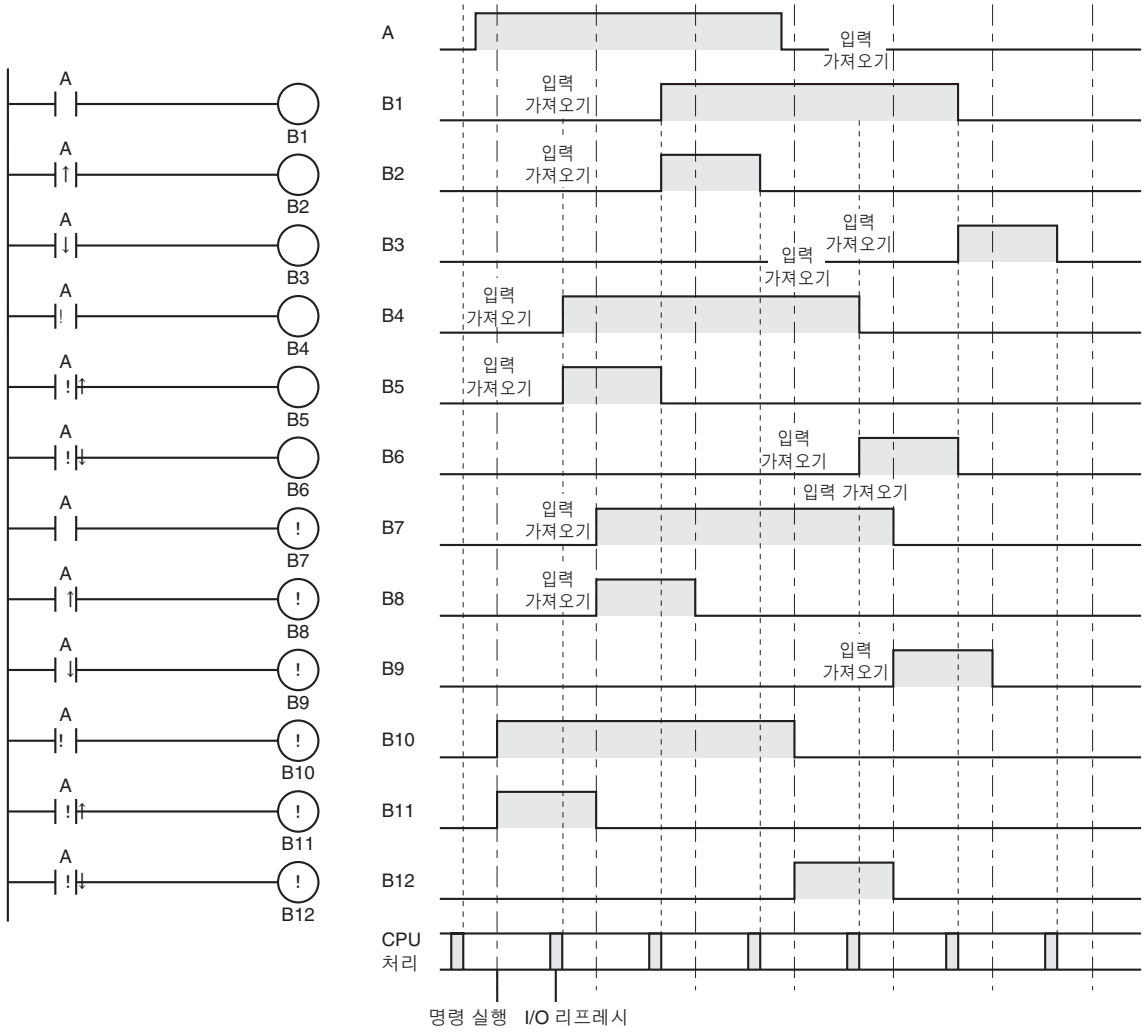




1.03의 입력 접점이 ON→OFF일 때 OFF가 되고,
1사이클 후에 입력 조건이 ON이 됩니다.

각 입출력 명령의 동작 타이밍

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

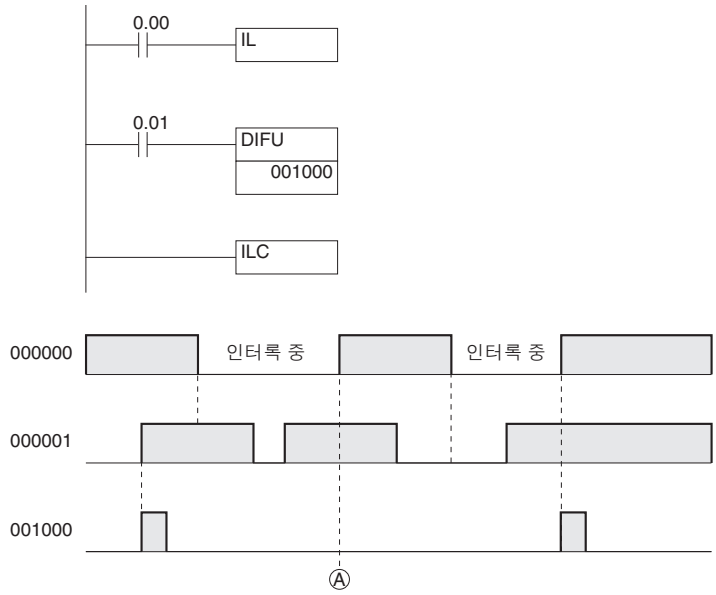


● 미분 명령

- 미분 명령은 이전 값이 ON인지, OFF인지의 플래그를 내부에 가지고 있습니다.
  운전 시작 시에는 기동 미분 명령(DIFU/@ 표시가 있는 명령)의 이전 값 플래그는 항상 ON에 설정되고, 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.

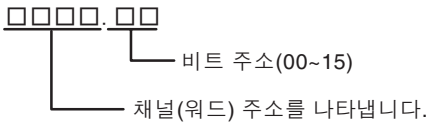
사용상의 주의 사항

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.
어느 경우든 명령이 실행되지 않습니다.

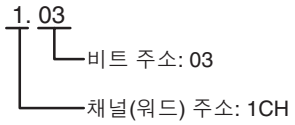
5-6-2 피연산자 지정 방법

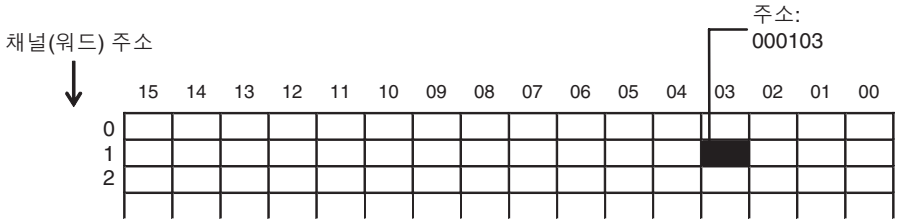
I/O 메모리 영역 지정 방법

● **비트 주소 지정 방법**



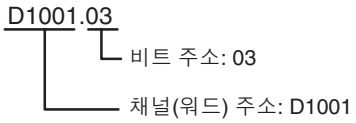
예: 입출력 릴레이
1CH의 비트 03 표시 방법

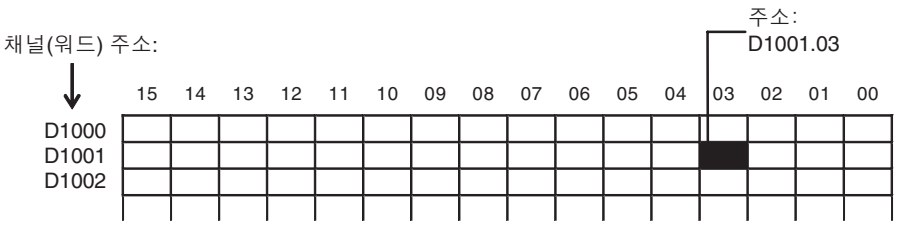




CJ2 CPU에서는 데이터 메모리(DM) 및 확장 데이터 메모리(EM)의 비트 주소를 지정할 수 있습니다.

예: DM 영역
D1001의 비트 03 표시 방법





● 채널(워드) 주소 지정 방법

□□□□CH
└──채널(워드) 주소를 나타냅니다.

예: 입출력 릴레이
10CH(비트 00~15) 표시 방법

10CH
└──채널(워드) 주소

데이터 메모리의 경우 D□□□□□ 또는 E□□□□□로 나타냅니다.

예: 데이터 메모리(DM)
D200 표시 방법

D200
└──채널(워드) 주소

예: 확장 데이터 메모리(EM)
현재 뱅크의 200CH 표시 방법

E200
└──채널(워드) 주소

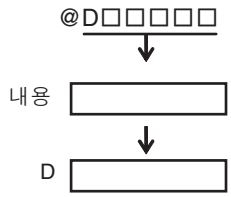
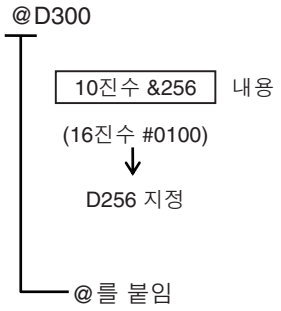
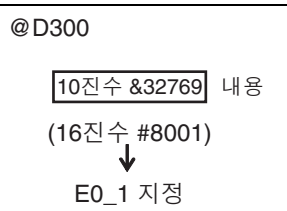
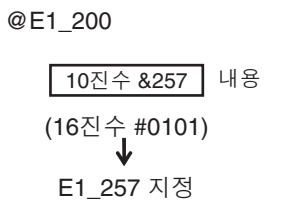
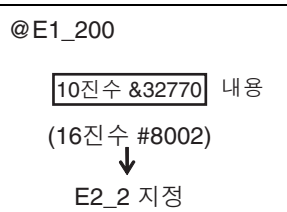
뱅크 번호 01의 200CH 표시 방법

E01_200
└──채널(워드) 주소
└──뱅크 번호

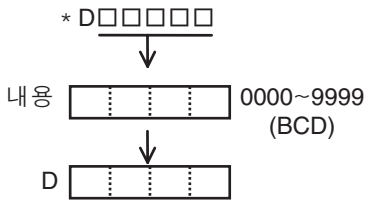
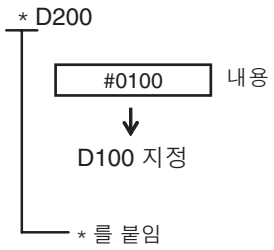
데이터 지정 방법

데이터 지정 방법	내용	예	명령에서 사용 예
비트 주소 직접 지정	1비트(접점)를 지정하기 위해 채널 주소+비트 주소로 직접 지정합니다. □□□□. □□ 비트 주소 00~15 채널 주소 (*1)	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	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[W0] 오프셋 비트 수 W0＝&2일 때 →10.02 시작 비트 주소	10.00[2]
채널 주소의 오프셋 지정	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＝&2일 때 →D2 시작 채널 주소	MOV 3 D0[200]

*1 타임업 플래그, 카운트 업 플래그, 태스크 플래그는 채널 주소/비트 주소를 구별하지 않습니다.

데이터 지정 방법	내용	예	명령에서 사용 예
DM/EM 간접 지정 BIN 모드 (*2)	데이터 메모리(DM) 또는 확장 데이터 메모리(EM)의 채널 주소를 지정하기 위해 내용을 BIN 값(00000~32767)으로 읽어서 각 영역의 선두부터의 오프셋으로 간접 지정합니다. '@' 표시를 맨 앞에 붙입니다. 		
	1. @D□□□□□의 내용이 10진수 &0~32767(16진수 #0000~7FFF)인 경우 D00000~32767을 지정합니다.		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.		—
	3. @E□_□□□□□의 내용이 10진수 &0~32767(16진수 #0000~7FFF)인 경우 지정된 뱅크 번호 □에서 E□_0~E□_32767을 지정합니다.		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.		

*2 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 0에서 E0_0을 지정합니다.
확장 데이터 메모리(EM)의 뱅크 번호가 n일 때 뱅크 번호(n+1)의 0을 시작점으로 간접 지정합니다.
예: 확장 데이터 메모리(EM)에서 뱅크 2의 내용이 32768인 경우 확장 데이터 메모리(EM)의 뱅크 3에서 E3_0을 지정합니다.

데이터 지정 방법	내용		예	명령에서 사용 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. '*' 표시를 맨 앞에 붙입니다. 			MOV #0001 * D200
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.

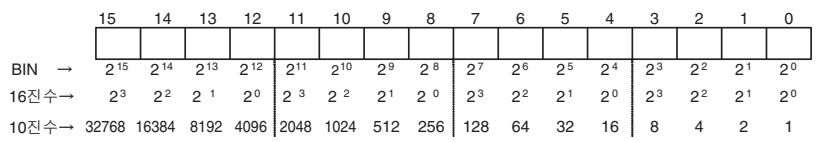
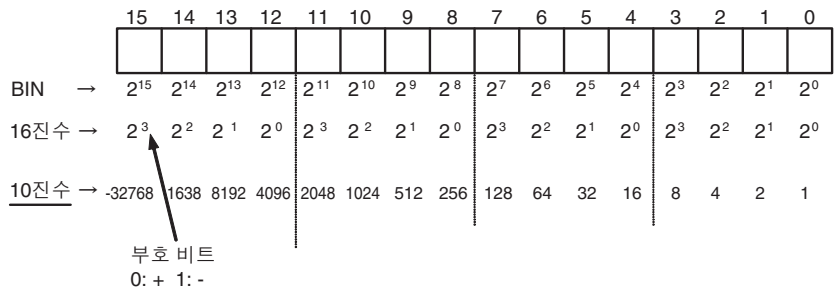
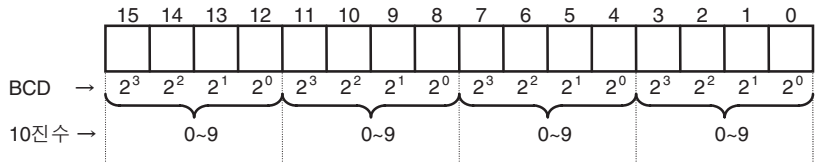
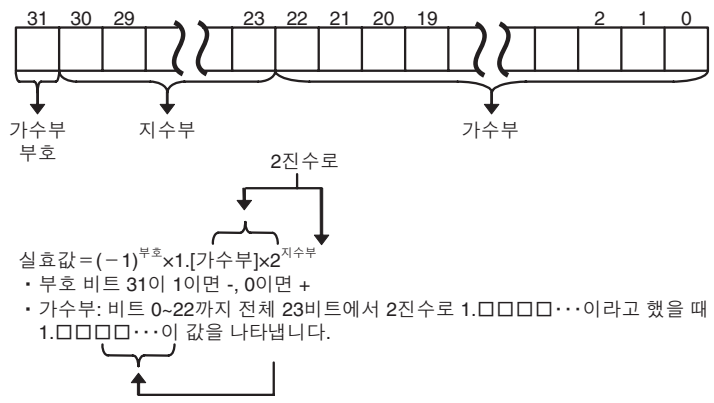
데이터 지정 방법	내용		예	명령에서 사용 예
레지스터 간접 지정	자동 감소	IR□의 내용을 1 또는 2씩 감소시킨 후 그 값을 I/O 메모리 실제 주소로 참조합니다. 1씩 감소시키는 경우: ,－IR□ 2씩 감소시키는 경우: ,－－IR□로 지정합니다.	,－－IR0 ,－IR1	LD ,－－IR0 IR0의 내용을 2 감소시킨 후 그 값을 I/O 메모리 실제 주소로 하는 접점을 로드합니다. MOV #0001 ,－IR1 IR1의 내용을 1 감소시킨 후 그 값을 I/O 메모리 실제 주소로 하는 채널에 #0001을 저장합니다.

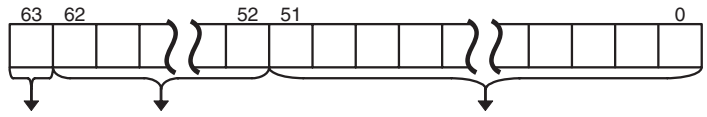
데이터 지정 방법	사용하는 피연산자	데이터 형식	기호	범위	명령에서 사용 예
상수 (16비트 데이터)	모든 BIN 데이터, 또는 범위가 있는 BIN 데이터	부호 없는 BIN	#	#0000~#FFFF	MOV #0100 D0 D0에 16진수 #0100(10진수 &256)을 저장합니다. + #0009 #0001 D1 D1에 16진수 #000A(10진수 &10)를 저장합니다.
		부호 있는 10진수	±	－32768~＋32767	MOV －100 D0 D0에 10진수 －100(16진수 #FF9C)을 저장합니다. + －9 －1 D1 D1에 10진수 －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><th>데이터
지정 방법</th><th>내용</th><th>기호</th><th>예</th><th>—</th></tr>
<tr>
<td>문자열
데이터</td>
<td>ASCII 코드(1바이트, 특수 문자는 제외)로 상위 바이트→하위 바이트의 순서로 하위(낮은 숫자) 채널→상위 채널의 순서로 저장됩니다.
문자 수가 홀수인 경우 최종 채널의 상위 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’
<table>
<tr><td>‘A’</td><td>‘B’</td></tr>
<tr><td>‘C’</td><td>‘D’</td></tr>
<tr><td>‘E’</td><td>NUL</td></tr>
</table>
‖
<table>
<tr><td>41</td><td>42</td></tr>
<tr><td>43</td><td>44</td></tr>
<tr><td>45</td><td>00</td></tr>
</table>
‘ABCD’
<table>
<tr><td>‘A’</td><td>‘B’</td></tr>
<tr><td>‘C’</td><td>‘D’</td></tr>
<tr><td>NUL</td><td>NUL</td></tr>
</table>
‖
<table>
<tr><td>41</td><td>42</td></tr>
<tr><td>43</td><td>44</td></tr>
<tr><td>00</td><td>00</td></tr>
</table>
</td>
<td>MOV$ D100 D200
<table>
<tr><td>D100</td><td>41</td><td>42</td></tr>
<tr><td>D101</td><td>43</td><td>44</td></tr>
<tr><td>D102</td><td>45</td><td>00</td></tr>
<tr><td>↓</td><td></td><td></td></tr>
<tr><td>D200</td><td>41</td><td>42</td></tr>
<tr><td>D201</td><td>43</td><td>44</td></tr>
<tr><td>D202</td><td>45</td><td>00</td></tr>
</table>
</td>
</tr>
<tr>
<td colspan="5">문자열로 취급되는 ASCII 코드는 다음과 같이 특수 문자를 제외한 영문자, 숫자, 가타카나, 기호입니다.
<table>
<tr><th></th><th></th><th colspan="16">상위 4비트</th></tr>
<tr><th></th><th></th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th></tr>
<tr><td rowspan="16">하위 4비트</td><td>0</td><td></td><td></td><td>sp</td><td>0</td><td>@</td><td>P</td><td></td><td>p</td><td></td><td></td><td></td><td>ー</td><td>タ</td><td>ミ</td><td></td><td></td></tr>
<tr><td>1</td><td></td><td></td><td>!</td><td>1</td><td>A</td><td>Q</td><td>a</td><td>q</td><td></td><td></td><td>。</td><td>ア</td><td>チ</td><td>ム</td><td></td><td></td></tr>
<tr><td>2</td><td></td><td></td><td>”</td><td>2</td><td>B</td><td>R</td><td>b</td><td>r</td><td></td><td></td><td>「</td><td>イ</td><td>ツ</td><td>メ</td><td></td><td></td></tr>
<tr><td>3</td><td></td><td></td><td>#</td><td>3</td><td>C</td><td>S</td><td>c</td><td>s</td><td></td><td></td><td>」</td><td>ウ</td><td>テ</td><td>モ</td><td></td><td></td></tr>
<tr><td>4</td><td></td><td></td><td>$</td><td>4</td><td>D</td><td>T</td><td>d</td><td>t</td><td></td><td></td><td>、</td><td>エ</td><td>ト</td><td>ヤ</td><td></td><td></td></tr>
<tr><td>5</td><td></td><td></td><td>%</td><td>5</td><td>E</td><td>U</td><td>e</td><td>u</td><td></td><td></td><td>・</td><td>オ</td><td>ナ</td><td>ユ</td><td></td><td></td></tr>
<tr><td>6</td><td></td><td></td><td>&</td><td>6</td><td>F</td><td>V</td><td>f</td><td>v</td><td></td><td></td><td>ヲ</td><td>カ</td><td>ニ</td><td>ヨ</td><td></td><td></td></tr>
<tr><td>7</td><td></td><td></td><td>’</td><td>7</td><td>G</td><td>W</td><td>g</td><td>w</td><td></td><td></td><td>ァ</td><td>キ</td><td>ヌ</td><td>ラ</td><td></td><td></td></tr>
<tr><td>8</td><td></td><td></td><td>(</td><td>8</td><td>H</td><td>X</td><td>h</td><td>x</td><td></td><td></td><td>ィ</td><td>ク</td><td>ネ</td><td>リ</td><td></td><td></td></tr>
<tr><td>9</td><td></td><td></td><td>)</td><td>9</td><td>I</td><td>Y</td><td>i</td><td>y</td><td></td><td></td><td>ゥ</td><td>ケ</td><td>ノ</td><td>ル</td><td></td><td></td></tr>
<tr><td>A</td><td></td><td></td><td>*</td><td>:</td><td>J</td><td>Z</td><td>j</td><td>z</td><td></td><td></td><td>ェ</td><td>コ</td><td>ハ</td><td>レ</td><td></td><td></td></tr>
<tr><td>B</td><td></td><td></td><td>+</td><td>;</td><td>K</td><td>[</td><td>k</td><td>{</td><td></td><td></td><td>ォ</td><td>サ</td><td>ヒ</td><td>ロ</td><td></td><td></td></tr>
<tr><td>C</td><td></td><td></td><td>,</td><td><</td><td>L</td><td>¥</td><td>l</td><td>|</td><td></td><td></td><td>ャ</td><td>シ</td><td>フ</td><td>ワ</td><td></td><td></td></tr>
<tr><td>D</td><td></td><td></td><td>ー</td><td>=</td><td>M</td><td>]</td><td>m</td><td>}</td><td></td><td></td><td>ュ</td><td>ス</td><td>ヘ</td><td>ン</td><td></td><td></td></tr>
<tr><td>E</td><td></td><td></td><td>.</td><td>></td><td>N</td><td>^</td><td>n</td><td>~</td><td></td><td></td><td>ョ</td><td>セ</td><td>ホ゛</td><td></td><td></td><td></td></tr>
<tr><td>F</td><td></td><td></td><td>／</td><td>?</td><td>O</td><td>_</td><td>o</td><td></td><td></td><td></td><td>ッ</td><td>ソ</td><td>マ゜</td><td></td><td></td><td></td></tr>
</table>
</td>
</tr>
</table>

5-6-3 데이터 형식

CJ 시리즈에서 사용 가능한 데이터의 형식은 다음과 같습니다.

종류	데이터 형식	10진수	16진수 4자리
부호 없는 BIN 데이터		&0 ~ 65535	#0000~FFFF
부호 있는 BIN 데이터	 최상위 비트를 부호 비트로 사용하는 2진수 16비트 데이터로 처리하지 않고, 그 값을 16진수 4자리로 표현합니다. 양수: 최상위 비트가 0(OFF)이면 양의 값 또는 0을 의미합니다. 16진수 4자리 표현은 0000~7FFFHex가 됩니다. 음수: 최상위 비트가 1(ON)이면 음의 값을 의미합니다. 16진수 4자리 표현은 8000~FFFFHex가 됩니다. 이때 음의 값(10진수)의 절대값은 2의 보수로 표현됩니다.	0~ −32768 0~ +32767	음수: #8000~FFFF 양수: #0000~7FFF
BCD 데이터 (2진수화 10진수)		#0~9999	#0000~9999
단정도 부동 소수점 데이터	 ・부호 비트 31이 1이면 -, 0이면 + ・가수부: 비트 0~22까지 전체 23비트에서 2진수로 1.□□□□···이라고 했을 때 1.□□□□···이 값을 나타냅니다. ・지수부 비트 23~30까지 전체 8비트에서 2^{n}의 n을 BIN 값으로 나타냅니다. 실제 값은 2^{n-127}이 됩니다. IEEE754 규격의 단정도를 준수합니다. 부동소수점 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.	—	—

종류	데이터 형식	10진수	16진수 4자리
배정도 부동 소수점 데이터	 가수부 부호 지수부 가수부 2진수로 실효값=(−1)부호×1.[가수부]×2지수부 ・부호 비트 63이 1이면 -, 0이면 + ・가수부: 비트 0~51까지 전체 52비트에서 2진수로 1.□□□□···이라고 했을 때 1.□□□□···이 값을 나타냅니다. ・지수부 비트 52~62까지 전체 11비트에서 2^n의 n을 BIN 값으로 나타냅니다. 실제 값은 2^{n-1023}이 됩니다. IEEE754 규격의 배정도를 준수합니다. 부동소수점 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.	—	—

참고

- 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+(7556에 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	BIN 표기			BCD 표기 (BCD 데이터)
	10진수 표기		16진수 표기	
(10진수)	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	

5-6-4 I/O 리프레시 타이밍

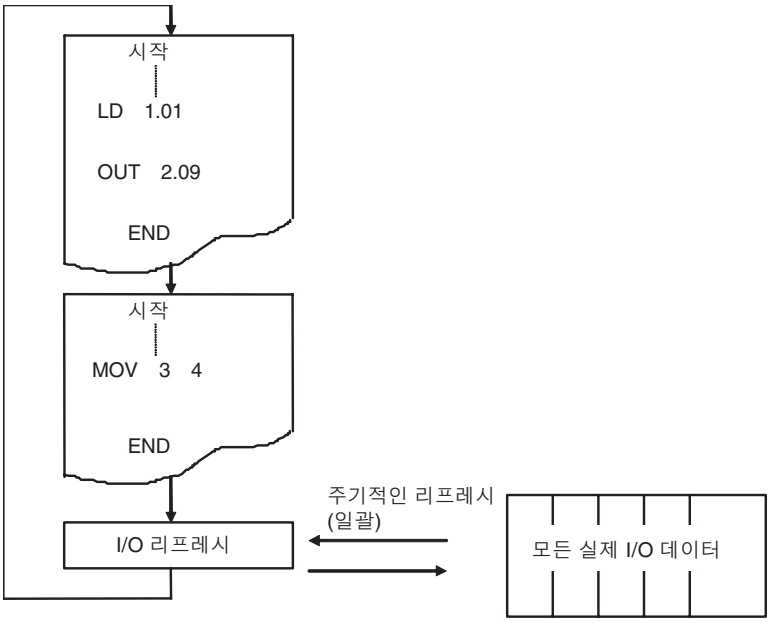
실제 I/O의 리프레시 타이밍에는 다음 2가지 방식이 있습니다.

- 주기적 리프레시
- 매회 리프레시(! 지정 명령 및 IORF/FIORF/DLNK 명령)

주기적 리프레시

실행 가능한 상태의 사이클릭 태스크를 모두 실행한 후 I/O 리프레시를 일괄 처리하는 리프레시 방식입니다.

I/O 리프레시 타이밍은 실행 상태에 있는 각 태스크를 실행한 후가 됩니다.



각 사이클릭 태스크를 실행한 후 I/O 리프레시를 수행하고 싶은 경우, END 명령 전에 실제 I/O의 모든 데이터 또는 지정 범위에 대해 IORF 명령을 실행하십시오.

매회 리프레시

프로그램에서 명령이 실행된 타이밍에 실행하는 리프레시 방식입니다.

● 매회 리프레시의 종류

매회 리프레시는 매회 리프레시 옵션(!)을 지정한 명령에 의한 방법과 매회 리프레시용 응용 명령을 사용하는 방법이 있습니다.

매회 리프레시 옵션 지정(!) 명령

명령에 매회 리프레시 옵션(!)을 지정하면 해당 명령 실행 시(사이클 도중) 리프레시합니다.

매회 리프레시용 응용 명령

리프레시 대상 유니트에 따라 다음과 같은 3종류의 명령이 있습니다.

IORF 명령 : 기본 I/O 유니트, 고기능 I/O 유니트(할당 릴레이 영역만 해당)
FIORF 명령 : 고기능 I/O 유니트
DLNK 명령 : CPU 고기능 유니트

각 명령별로 대응하는 리프레시 대상 유니트는 다음과 같습니다.

리프레시 가능 여부 ○: 가능 ×: 불가능

유니트 종류	매회 리프레시 옵션 지정(!) 명령	IORF 명령	FIORF 명령	DLNK 명령
기본 I/O 유니트	○	○	×	×
할당 릴레이 영역				
고기능 I/O 유니트	×	○(*2)	○(*2)	×
할당 릴레이 영역				
할당 DM 영역		×		
CPU 고기능 유니트	×	×	×	○
• 할당 릴레이 영역 　• 할당 DM 영역 　• 유니트 고유의 영역(*1)				

*1 EtherNet/IP 유니트/Controller Link 유니트/SYSMAC LINK 유니트의 데이터 링크, DeviceNet 원격 I/O 통신

*2 고기능 I/O 유니트의 할당 릴레이 영역에 대한 매회 리프레시는 IORF 명령 및 FIORF 명령 둘 모두에서 실행할 수 있습니다. FIORF 명령은 IORF 명령보다 명령 실행 시간이 빠르며 피연산자에 고기능 I/O 유니트의 기기 번호를 지정하기만 하면 되므로 간단합니다.

사용상의 주의 사항

매회 리프레시를 실행하면 명령 실행 처리 시간이 길어지므로 사이클 시간이 길어지는 경우가 있으므로 주의하십시오.

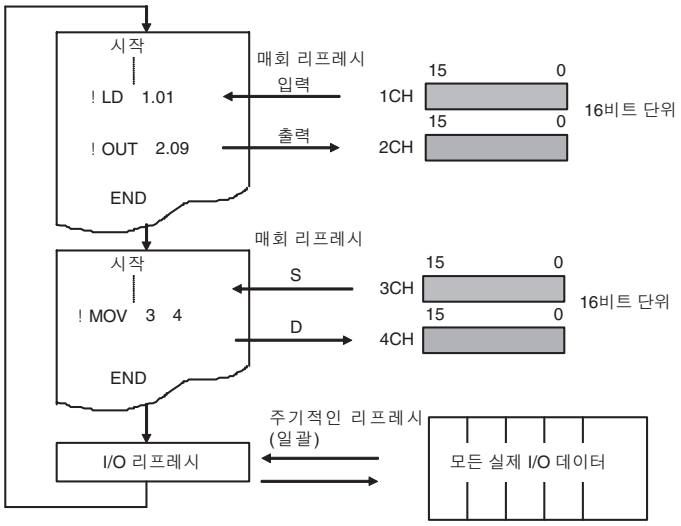
● 매회 리프레시 옵션 지정(!) 명령에 의한 경우

명령어 앞에 '!'(매회 리프레시 옵션)를 붙입니다.
피연산자로 실제 I/O주소(입출력 릴레이 영역)를 지정한 경우 명령을 실행하면 다음과 같이 리프레시를 수행합니다.

- 비트 지정 명령의 경우:

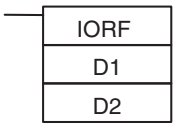
유니트 종류	리프레시 데이터
기본 I/O 유니트	해당 비트를 포함하여 16비트를 I/O 리프레시합니다.

- 채널 지정 명령의 경우: 지정된 16비트를 I/O 리프레시합니다.
- 입력 또는 원본(S) 피연산자의 경우: 명령 실행 직전에 IN 리프레시합니다.
- 출력 또는 대상(D) 피연산자의 경우: 명령 실행 직후에 OUT 리프레시합니다.



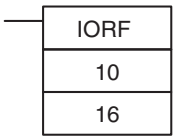
● IORF(I/O 리프레시) 명령

IORF(I/O 리프레시) 명령을 사용하면 기본 I/O 유니트에 있는 실제 I/O의 모든 데이터 또는 고기능 I/O 유니트에 있는 할당 릴레이 영역의 데이터를 사이클 도중에 리프레시할 수 있습니다.



D1: 시작 CH 번호
D2: 종료 CH 번호
D1에서 D2 범위의 I/O 채널 데이터를 리프레시합니다.

예



10CH~16CH의 7CH를 I/O 리프레시합니다.

용도: 기본 I/O 유니트에 대해 입력에서 출력까지 고속 응답이 요구되는 연산 처리를 수행하고 싶은 경우, 연산 전과 후에 이 IORF 명령을 실행합니다.

사용상의 주의 사항

IORF 명령은 비교적 실행 시간이 길며, I/O 리프레시하는 채널 수가 많아질수록 실행 시간이 길어져서 전체 사이클 시간이 길어지므로 주의하십시오.

● FIORF(고기능 I/O 유니트 매회 리프레시) 명령

FIORF(고기능 I/O 유니트 매회 리프레시) 명령을 사용하면, 지정된 기기 번호를 가진 고기능 I/O 유니트의 할당 릴레이 영역 및 할당 DM 영역의 데이터를 필요할 때만 리프레시할 수 있습니다.



N: 고기능 I/O 유니트의 기기 번호

예



기기 번호 &20의 고기능 I/O 유니트에 대해
할당 릴레이, 할당 DM의 리프레시를 실행합니다.

용도: PLC 시스템 설정에서 해당 기기 번호를 가진 고기능 I/O 유니트의 주기적 리프레시 유무 지정을 '없음'으로 설정한 후 이 FIORF 명령을 실행하면, 필요할 때만 해당 기기 번호의 고기능 I/O 유니트에 대해 I/O 리프레시가 실행됩니다.
그 결과, 고기능 I/O 유니트 장착 시 사이클 시간의 증가를 막을 수 있습니다.

● DLNK(CPU 고기능 유니트 매회 I/O 리프레시) 명령

DLNK(CPU 고기능 유니트 매회 I/O 리프레시) 명령을 사용하면, 지정된 유니트 번호를 가진 CPU 고기능 유니트의 할당 릴레이/할당 DM 영역, CPU 고기능 유니트 고유의 리프레시(*) 데이터의 리프레시 타이밍을 늘릴 수 있습니다.

* CPU 고기능 유니트 고유의 리프레시는 다음 예와 같은 서비스를 가리킵니다.
예: Controller Link 또는 SYSMAC LINK의 데이터 링크, DeviceNet 원격 I/O 통신



N: CPU 고기능 유니트의 유니트 번호

예

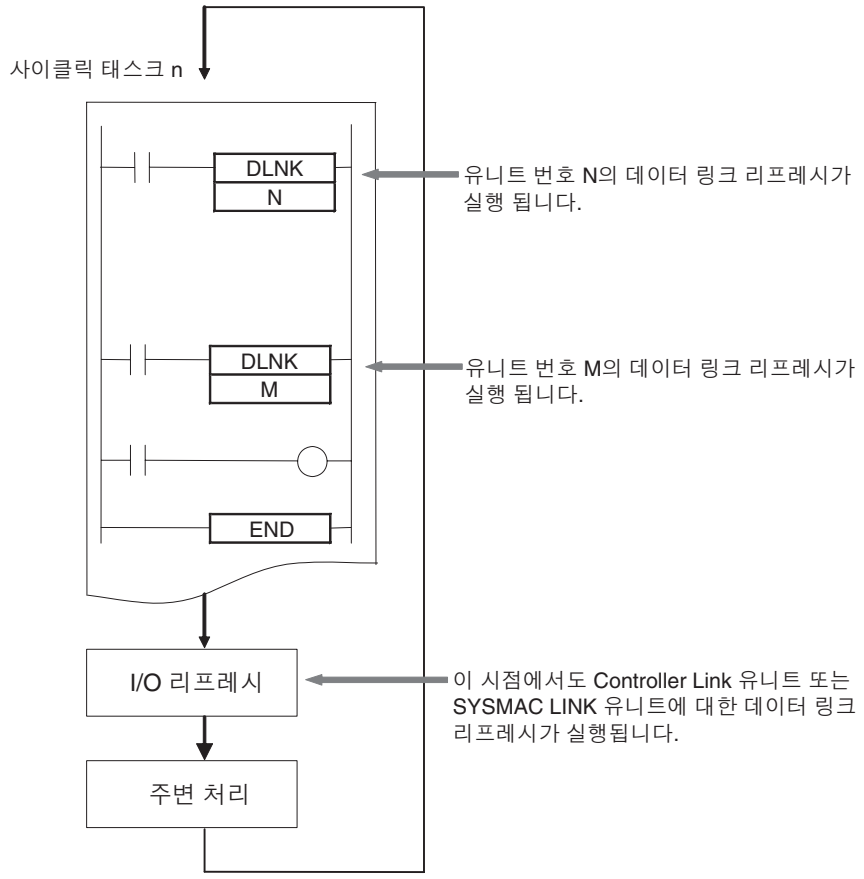


유니트 번호 #1의 CPU 고기능 유니트에 대해 할당 릴레이, 할당 DM의 리프레시 및 고유 리프레시를 실행합니다.
또한 여기서 유니트 번호 1의 CPU 고기능 유니트가 Controller Link 유니트인 경우, 고유 리프레시가 아니라 데이터 링크 리프레시를 실행합니다.

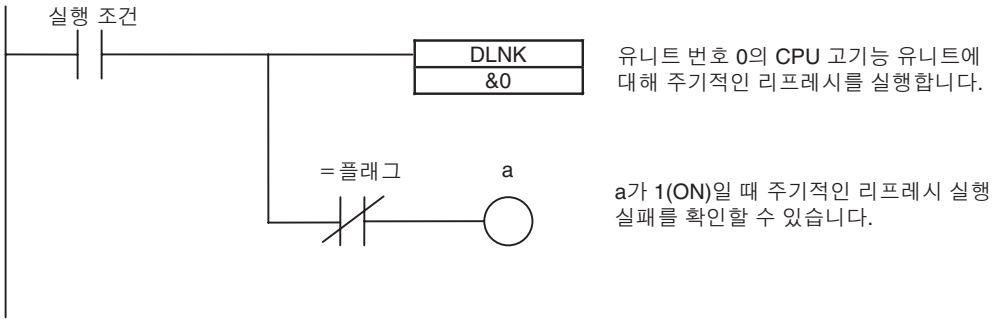
용도: 사이클 시간이 길기 때문에 Controller Link 데이터 링크의 리프레시 간격이 길어지는 경우 등에 Controller Link 유니트에 대한 DLNK 명령을 실행하면 데이터 링크 실행 타이밍을 늘려서 데이터 링크 실행 간격을 줄일 수 있습니다.

참고

DLNK 명령의 사용 예는 다음과 같습니다.
예) 데이터 링크의 경우



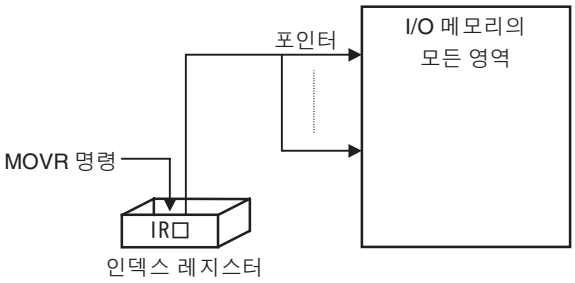
DLNK 명령으로 지정한 유니트 번호의 CPU 고기능 유니트가 데이터 갱신 중에 바쁜 상태인 경우, 매회 I/O 리프레시가 실행되지 않고 = 플래그가 0(OFF)이 됩니다. 따라서 DLNK 명령은 기본적으로 다음과 같이 = 플래그를 사용하여 실행 완료를 확인하십시오.



5-7 인덱스 레지스터 기능

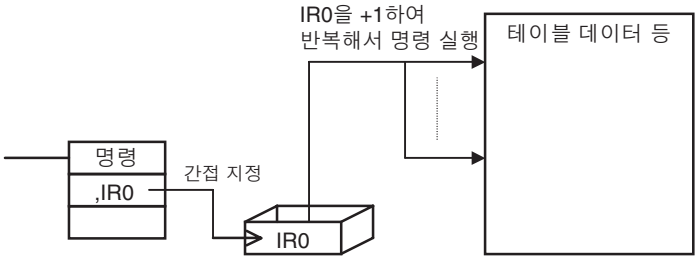
5-7-1 인덱스 레지스터란?

인덱스 레지스터는 I/O 메모리의 주소를 지정하기 위한 포인터의 역할을 합니다.
이 인덱스 레지스터에 I/O 메모리 실제 주소를 전용 명령(MOVR 명령 등)을 사용하여 저장한 후, 일반 명령으로 인덱스 레지스터를 간접 참조하여 I/O 메모리를 지정합니다.
인덱스 레지스터는 접점, 타이머/카운터 등, I/O 메모리의 모든 영역을 지정할 수 있는 점이 특징입니다.



5-7-2 인덱스 레지스터 사용 방법

인덱스 레지스터를 통한 간접 지정과 반복 처리(FOR－NEXT 명령 등)를 조합하면 테이블 데이터 등의 연속 주소 등에 대한 복수의(동일) 명령 처리를 하나의 명령으로 묶을 수 있으므로 프로그램을 간소화할 수 있습니다.



기본적으로 다음과 같은 순서로 실행됩니다.

1 MOVR 명령을 사용하여 인덱스 레지스터에 있는 주소의 I/O 메모리 실제 주소를 저장합니다.

2 그런 다음, 명령 A의 피연산자에 인덱스 레지스터 간접 지정을 사용하여 실행합니다.

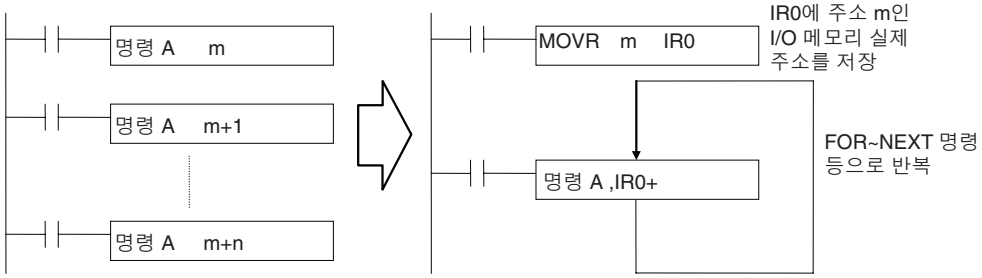
3 인덱스 레지스터에 가산/감산/증가/감소 등의 처리(*)를 해서 주소를 이동합니다.

4 조건이 성립할 때까지 2, 3의 처리를 반복합니다.

* 인덱스 레지스터의 가산/감산/증가/감소는 다음 방법으로 수행됩니다.

- 인덱스 레지스터 간접 지정의 유형:
  인덱스 레지스터의 자동 증가(,IR□+ 또는 ,IR□++), 자동 감소(,-IR□ 또는 ,--IR□), 상수 오프셋(상수 ,IR□), DR 오프셋(DR□ ,IR□)
- 인덱스 레지스터 직접 지정 명령:
  BIN 2워드 가산(+L), BIN 2워드 감산(-L), BIN 2워드 증가(++L),
  BIN 2워드 감소(--L)

예)



예를 들어, 위의 명령 A가 비교 명령인 경우 테이블 데이터의 하위에서 상위 순서로 데이터를 읽어서 어떤 값과 비교하는 등의 처리를 수행할 수 있습니다.

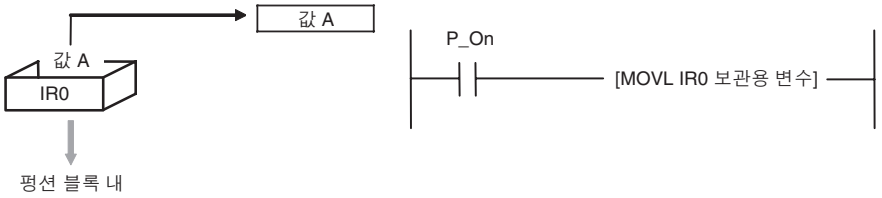
이와 같이, 인덱스 레지스터는 응용 방법에 따라 사용자 정의 명령 처리 블록을 자유롭게 작성할 수 있습니다.

사용상의 주의 사항

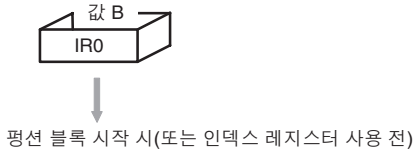
- 펑션 블록 내에서 인덱스 레지스터(IR0~15)를 사용하는 경우, 다른 펑션 블록 내 또는 프로그램에서 동일한 인덱스 레지스터를 사용하면 서로 경합하여 프로그램이 정상적으로 동작하지 않습니다.
  따라서 인덱스 레지스터(IR0~15)를 사용하는 경우, 반드시 펑션 블록을 시작할 때(또는 인덱스 레지스터 사용 전에) 해당 시점의 인덱스 레지스터의 값을 다른 곳에 보관해 두었다가 펑션을 종료할 때(또는 인덱스 레지스터 사용 후) 사용한 인덱스 레지스터의 값을 보관해 둔 값으로 되돌리도록 프로그램을 작성하십시오.

  예: 펑션 블록 시작 시(또는 인덱스 레지스터 사용 전)

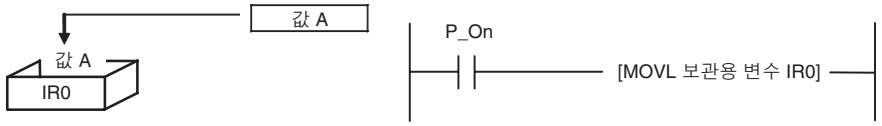
  1 IR의 값(예: A)을 다른 곳에 보관



  2 IR 사용



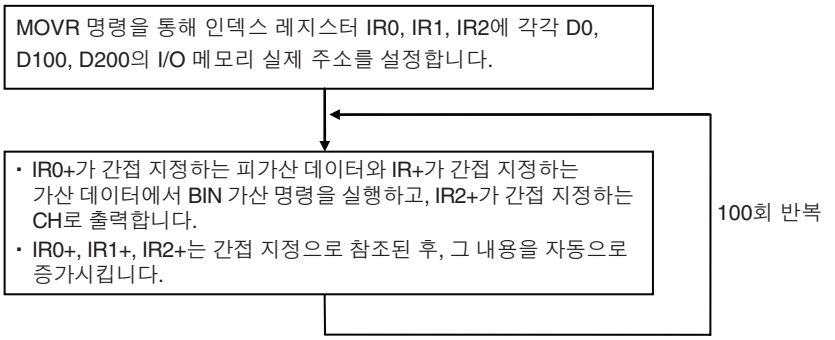
  3 보관해 둔 값(예: A)을 IR에 되돌림

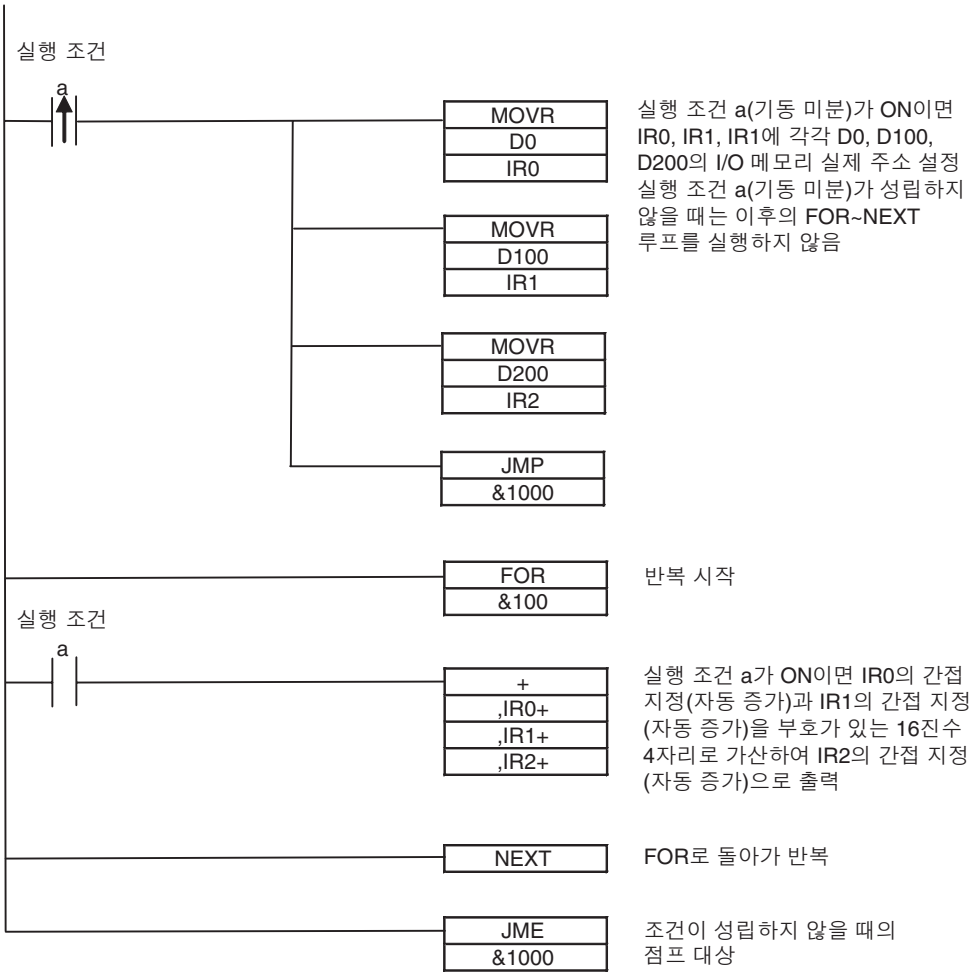


- 인덱스 레지스터를 사용하는 경우 반드시 값을 설정한 후 사용하십시오. 값이 설정되지 않은 상태에서 사용하면 동작이 일정치 않습니다.

인덱스 레지스터의 사용 예

예: D0~D99까지의 데이터(피가산 데이터)와 D100~D199의 데이터(가산 데이터)를 더하여, 각 연산 결과를 D200~D299에 출력합니다. 하나의 가산 명령의 피연산자에 인덱스 레지스터로 지정하고 그것을 증가시키면서 반복하면 이 동작을 구현할 수 있습니다.





참고

인덱스 레지스터를 간접 지정하는 일반 명령과 다음의 인덱스 레지스터 직접 지정 명령을 조합하여 사용합니다.

● 인덱스 레지스터 직접 지정 명령

인덱스 레지스터 IR0~15, 1레지스터당 2CH 크기이므로 2워드 길이 명령(L 표시 있음)을 사용합니다.

명령 종류	명령어	니모닉 (*1)	주요 용도
데이터 전송	인덱스 레지스터 설정	MOVR	인덱스 레지스터에 특정 주소의 I/O 메모리 실제 주소 저장
	인덱스 레지스터 설정 (타이머/카운터의 현재값용)	MOVRW	
테이블 데이터 처리 명령	레코드 위치 설정	SETR	
	레코드 위치 읽기	GETR	인덱스 레지스터 내의 I/O 메모리 실제 주소를 바탕으로 출력
추적 명령	레코드 검색	RSRCH	일치하는 레코드 선두의 I/O 메모리 실제 주소를 인덱스 레지스터에 저장
	레코드 정렬	RSORT	정렬 대상 범위의 최종 I/O 메모리 실제 주소+1을 인덱스 레지스터에 저장
데이터 전송	2워드 전송	MOVL	인덱스 레지스터 간의 전송. 교환, 비교에 사용
	2워드 데이터 교환	XCGL	
데이터 비교 명령	2워드 및 일치	=L	
	2워드 및 불일치	<>L	
	2워드 및 미만	<L	
	2워드 및 이하	<=L	
	2워드 및 초과	>L	
	2워드 및 이상	>=L	
	부호 없는 2워드 비교	CMPL	
증가 및 감소 명령	BIN 2워드 증가	++L	인덱스 레지스터에 가산/증가 등의 처리를 하여 주소를 이동시킴
	BIN 2워드 감소	--L	
사칙연산 명령	부호 있음 및 CY 없음 BIN 2워드 가산	+L	
	부호 있음 및 CY 없음 BIN 2워드 감산	-L	
특수 명령	CV→CS 주소 변환	FRMCV	CVM1/CV 시리즈의 I/O 메모리 실제 주소와 CJ 시리즈의 I/O 메모리 실제 주소 간에 변환
	CS→CV 주소 변환	TOCV	

5-7-3 인덱스 레지스터 관련 처리

CJ 시리즈에는 인덱스 레지스터와 관련된 처리로 테이블 데이터 처리가 있습니다.
테이블 데이터 처리는 크게 스택 처리와 테이블 처리로 구분됩니다.

처리			용도	사용 명령
1. 스택 처리			선입 선출, 후입 선출, 읽기(참조), 변경(갱신), 삽입, 삭제, 카운트 등의 처리	SSET, PUSH, FIFO, LIFO, SREAD, SWRIT, SINS, SDEL, SNUM 명령
2. 테이블 처리	1개 레코드가 1워드인 테이블 처리	기본적인 처리	정수 또는 부동 소수점 데이터의 최대값/최소값 검색, SUM 계산 등의 처리	MIN, MAX, MINL, MAXL, MINF, MAXDF, MIND, MAXD, SRCH, SUM, FCS 명령
		응용 처리	테이블 데이터 내의 기타 각종 처리(비교, 재배열 등)	MIN, MAX, SRCH 명령이나 비교 명령 등과 인덱스 레지스터의 조합
	1개 레코드가 복수의 워드인 테이블 처리		사용자가 등록한 레코드 영역(워드 수, 범위)에 대한 검색, 정렬 처리	RSRCH <, RSRCH <=, RSRCH =, RSRCH >=, RSRCH >, RSRCH2 <, RSRCH2 <=, RSRCH2 =, RSRCH2 >=, RSRCH2 >, RSRCH4 <, RSRCH4 <=, RSRCH4 =, RSRCH4 >=, RSRCH4 >, RSORT, RSORT2, RSORT4 명령
			사용자가 등록한 레코드 영역(워드 수, 범위)에 대해 각종 명령을 사용한 연산 처리	다음의 여러 명령과 인덱스 레지스터를 조합 • DIM, SETR, GETR 명령 • 각종 명령(비교 명령 등)

5-7-4 인덱스 레지스터를 모니터링하는 방법

다음과 같은 방식으로 모니터링할 수 있습니다.

- CX-Programmer에서 각 태스크의 마지막 IR 값을 모니터링하려는 경우

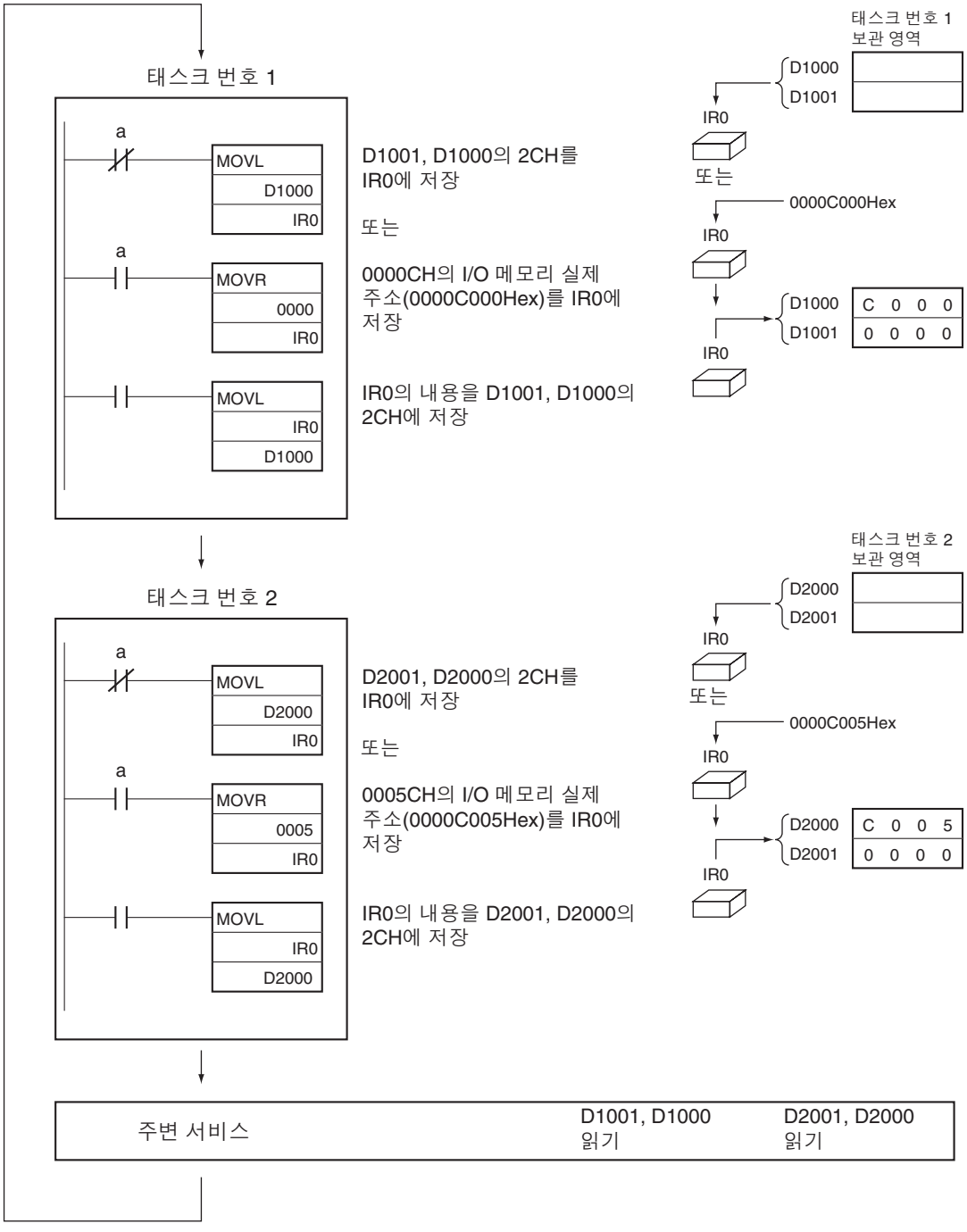
또는

- 상위 링크 명령/FINS 명령을 통해 IR 값을 모니터링하려는 경우

  태스크 종료 시 IR의 데이터를 각 태스크별로 다른 영역(DM 등)에 보관하고, 각 태스크 기동 시 해당 보관 영역(DM 등)에서 데이터를 가져오도록 프로그램을 작성하십시오.
  그런 다음 해당 태스크별로 보관한 영역(DM 등)에 대해 CX-Programmer, 상위 링크 명령/FINS 명령을 사용하여 읽기/쓰기를 수행하십시오.

예:

(주) IR에는 반드시 정확한 I/O 메모리 실제 주소를 저장하십시오.



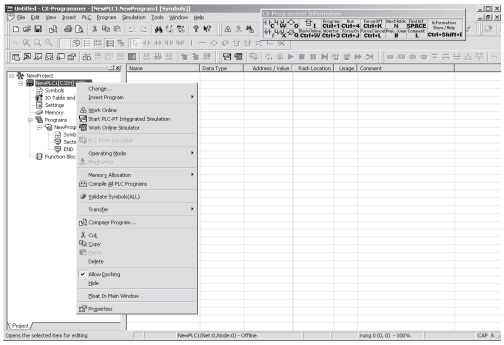
5-7-5 인덱스 레지스터/데이터 레지스터의 태스크 간 공통화

IR(인덱스 레지스터) 및 DR(데이터 레지스터)을 태스크 간에 공통으로 사용할 수 있습니다. 하지만 평상시에는 태스크별로 독립되어 있습니다.
현재 어떤 설정에서 동작하고 있는지는 특수 보조 릴레이(A99.14)로 확인할 수 있습니다.

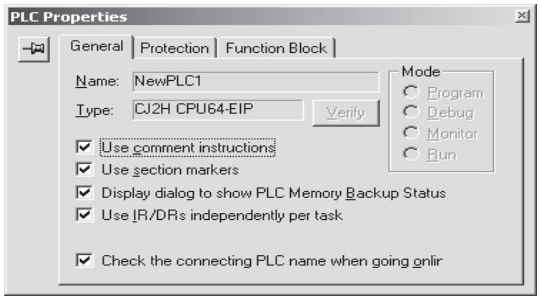
● 설정 방법

CX-Programmer를 사용하여 설정하며 설정 방법은 다음과 같습니다.

1 **CX-Programmer의 프로젝트 트리에서 PLC를 선택하고 마우스 오른쪽 단추로 클릭합니다.**



2 **[속성]을 선택합니다.**
다음과 같은 [PLC 속성] 대화 상자가 표시됩니다.



IR/DR을 태스크별로 독립시키려면 '태스크별로 개별 IR/DR 사용'을 선택한 상태로 둡니다(기본적으로 선택됨).
IR/DR을 태스크 간에 공통으로 사용하는 경우에는 '태스크별로 개별 IR/DR 사용'의 선택을 취소합니다.

● 관련 특수 보조 릴레이

명칭	주소	내용
IR/DR의 태스크 간 동작 표시	A99.14	IR(인덱스 레지스터) 및 DR(데이터 레지스터)을 태스크 간에 공통으로 사용하는지 여부를 표시합니다. 0: 태스크별로 독립(초기값) 1: 태스크 간 공통

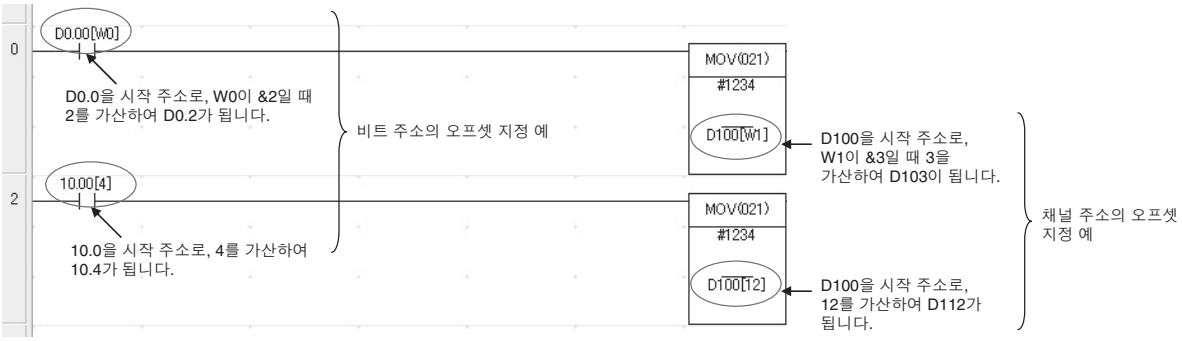
참고

- IR/DR을 태스크 간에 공통으로 사용하여 IR/DR의 값을 공유하면 래더 프로그램에서 별도로 보관하는 처리를 작성할 필요가 없게 됩니다.
  보관 처리란, 태스크 종료 시 IR/DR의 값을 특정 영역에 저장해 두었다가 태스크 기동 시 해당 영역에서 가져오는 처리를 말합니다.
  래더 프로그램을 사용한 보관 처리에 대해서는 '6-18 인덱스 레지스터'(6-32페이지)를 참조하십시오.
- IR/DR을 태스크 간에 공통으로 사용하는 경우 태스크 전환 시간이 약간 빨라집니다.
  IR/DR을 사용하지 않는 경우 또는 IR/DR을 태스크별로 독립시킬 필요가 없는 경우 이 설정을 사용하십시오.

5-8 주소의 오프셋 지정

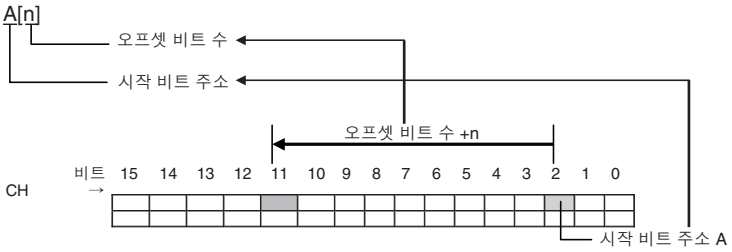
5-8-1 주소의 오프셋 지정이란?

명령의 피연산자에 주소를 지정하는 경우 지정한 주소에 [ ]를 붙이고 그 안에 오프셋 값을 기술하면 지정할 주소를 변경할 수 있습니다.



● 비트 주소의 경우

A(시작 비트 주소)에 n(오프셋 비트 수)으로 지정한 숫자만큼 비트 주소를 가산합니다.



시작 비트 주소

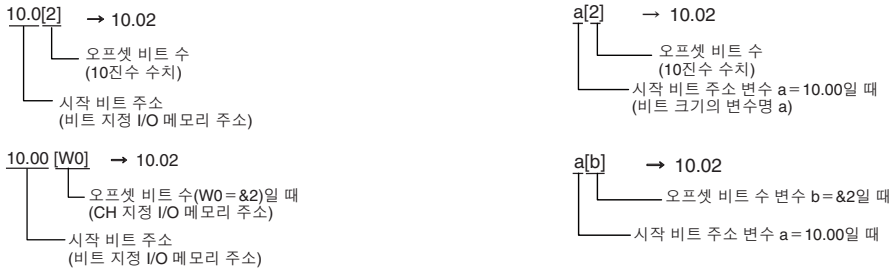
비트 지정 주소 또는 변수 지정(STRING 형식, NUMBER 형식 제외)이 가능하며, 영역 종류는 H/W/DM/EM 영역만 사용할 수 있습니다.
I/O 코멘트는 이 시작 비트 주소의 I/O 코멘트를 나타냅니다.

오프셋 비트 수

10진수 숫자, CH 지정 주소(CIO는 지정 불가능) 또는 1CH 크기(INT 형식, UINT 형식, WORD 형식, CHANNEL 형식)의 변수를 지정할 수 있습니다.
CH 지정 주소의 경우 해당 주소 내의 값을 간접 지정합니다.
오프셋 후의 비트 주소가 비트 15를 초과하면 다음의 +1CH의 비트 00에 이어서 오프셋 지정합니다.
오프셋을 간접 지정하는 경우, 심볼 비교 명령 등으로 오프셋 후의 값이 상한을 초과하지 않도록 처리의 입력 조건을 기술하십시오.

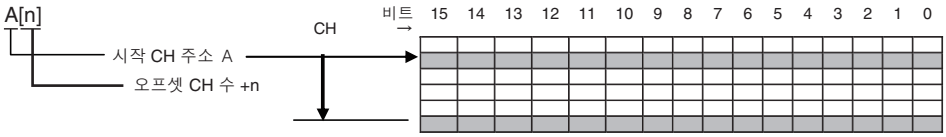
시작 비트 주소가 속하는 영역 종류에 대해 오프셋 비트 수가 초과한 경우, I/O 메모리 실제 주소에서 다음의 영역 종류를 지정합니다(메모리 맵에서 영역 종류별 배치 순서는 '부록-4 I/O 메모리 실제 주소'를 참조하십시오).

예



● 채널 주소의 경우

A(시작 CH 주소)에 n(오프셋 CH 수)으로 지정한 숫자만큼 CH 주소를 가산합니다.



시작 CH 주소

채널 지정 주소 또는 변수 지정(STRING 형식, NUMBER 형식 제외)이 가능하며, 영역 종류는 H/W/DM/EM 영역만 사용할 수 있습니다.
I/O 코멘트는 이 시작 CH 주소의 I/O 코멘트를 표시합니다.

오프셋 CH 수

10진수 숫자, CH 지정 주소(CIO는 지정 불가능) 또는 1CH 크기(INT 형식, UINT 형식, WORD 형식, CHANNEL 형식)의 변수를 지정할 수 있습니다.
CH 지정 주소, 변수의 경우 해당 주소 내의 값을 간접 지정합니다.

오프셋 후의 비트 주소가 비트 15를 초과하면 다음의 +1CH의 비트 00에 이어서 오프셋 지정합니다.
오프셋을 간접 지정하는 경우, 심볼 비교 명령 등으로 오프셋 후의 값이 상한을 초과하지 않도록 처리의 입력 조건을 기술하십시오.

시작 채널 주소가 속하는 영역 종류에 대해 오프셋 CH 수가 초과한 경우, I/O 메모리 실제 주소에서 다음의 영역 종류를 지정합니다(메모리 맵에서 영역 종류별 배치 순서는 '부록-4 I/O 메모리 실제 주소'를 참조하십시오).

예



⚠ 주 의

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.

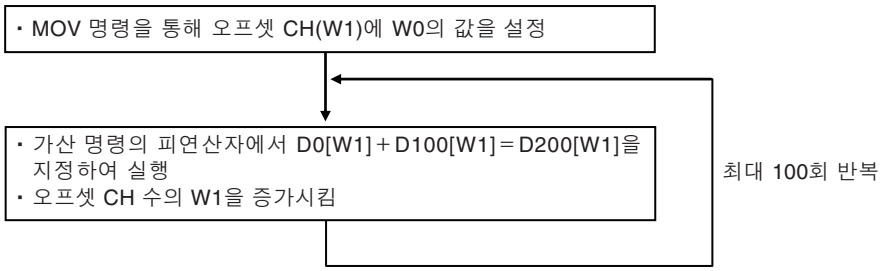


5-8-2 주소의 오프셋 용도의 예

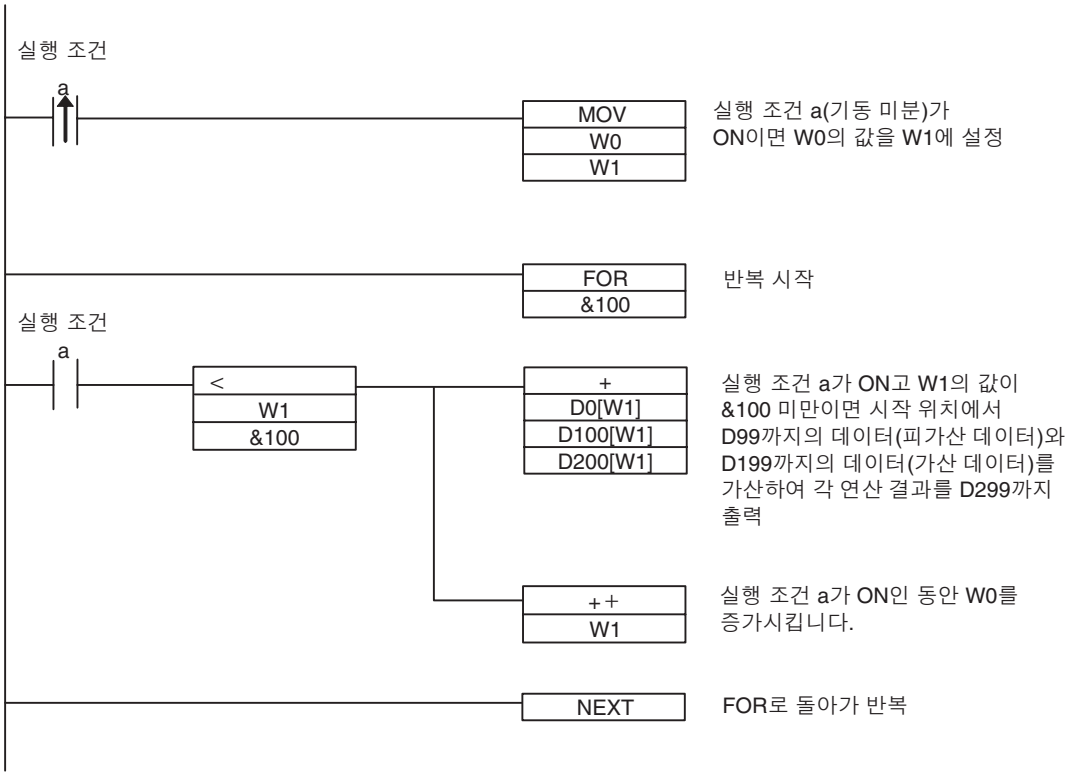
[ ] 안의 오프셋에 CH 지정 I/O 메모리를 지정하면 해당 채널 내의 값에 따라 주소를 동적으로 지정할 수 있습니다.
예를 들어, [ ] 안의 값을 증가시켜 하나의 명령에서 주소를 +1 가산해서 변경시킨 후 실행할 수 있습니다.

● 래더 프로그램 예

예) D0~D99와 D100~D199의 연속된 데이터가 2개 있습니다.
W0으로 지정한 시작 위치에서 마지막 데이터까지 가산하여 D200의 도중부터 D299에 출력합니다.
예를 들어, W0이 30인 경우 D30~D99와 D130~D199의 값을 더하여 D230~D299에 출력합니다.



또한 간접 지정 범위가 상한을 초과하지 않도록 심볼 비교 명령(<)을 입력 조건으로 설정하여 W1이 &100을 초과하지 않도록 처리를 실행합니다.



5-9 작성한 프로그램 확인

CJ 시리즈에서는 작성한 프로그램을 다음의 4단계로 확인합니다.

- CX-Programmer의 입력 조작 시
- CX-Programmer로 프로그램 확인
- 명령 실행 시 확인
- 운전을 정지하는 에러(프로그램 에러) 확인

5-9-1 CX-Programmer의 입력 조작 시 등의 확인

CX-Programmer에서는 다음과 같은 상황에 자동으로 프로그램을 확인합니다.

프로그램 확인 타이밍	확인 내용
래더 프로그램 입력 시	명령 입력, 피연산자 입력, 회로 패턴
파일 로드 시	모든 명령의 모든 피연산자에서 확인, 회로 패턴으로 확인
다운로드 시	CJ 시리즈 지원 기종, 모든 명령의 모든 피연산자
온라인 편집 시	용량 초과 등

확인 결과는 출력 창의 텍스트 탭에 표시됩니다. 또한 래더 보기에서 에러가 있는 회로는 해당 좌측 모선이 적색으로 표시됩니다.

5-9-2 CX-Programmer에서 프로그램 확인

CX-Programmer에서는 사용자가 프로그램을 확인할 수 있습니다. 이때, 프로그램 확인 수준을 A, B, C(중요도가 A, B, C 순서로 낮아짐) 및 사용자 지정 수준의 4개로 설정할 수 있습니다.

CX-Programmer에서는 명령의 간접 지정 피연산자의 범위 에러가 확인되지 않습니다. 명령의 피연산자에서 데이터가 잘못된 경우 다음 항목에서 설명하는 명령 실행 시의 확인에서 ER 플래그가 ON이 됩니다. 자세한 내용은 "명령 참조 설명서"(SBCA-351)를 참조하십시오.

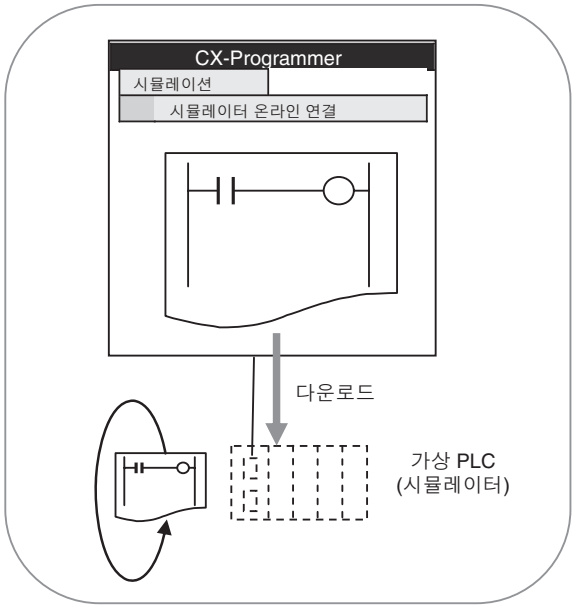
CX-Programmer 확인 내용에 대한 자세한 사항은 "CX-Programmer 조작 설명서"(SBCA-337)를 참조하십시오.

5-9-3 시뮬레이터 기능을 사용한 디버그

PC에서 CPU 유니트의 동작을 시뮬레이션하면 작성한 프로그램의 디버그 작업을 실제 기기의 PLC에 연결하지 않고도 PC에서 처리할 수 있습니다.

래더 프로그램의 동작 확인

CX-Programmer에서 시뮬레이터(CX-Simulator)를 기동하면 작성된 래더 프로그램의 동작을 가상 PLC에서 확인 및 디버그할 수 있습니다.



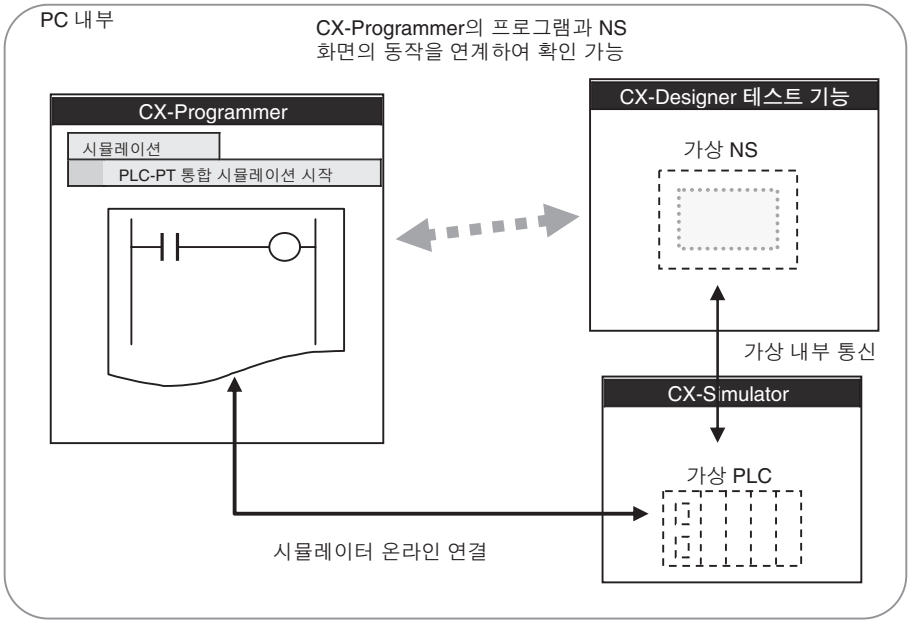
시뮬레이터에서 프로그램의 전송, 모니터링 이외에 다음과 같은 기능을 사용할 수 있습니다.

디버그 조작에 대한 자세한 내용은 "CX-Programmer 조작 설명서"(SBCA-337)를 참조하십시오.

- 연속 스캔/스텝 실행, 1스캔/스텝 실행 등의 다양한 실행
- 중단점, 시작점, I/O 중단 조건 지정
- 각 태스크의 실행 횟수/실행 시간 확인
- 의사 인터럽트 태스크의 기동
- 접점의 강제 세트/리셋

터치 패널과의 공유 동작에 의한 디버그(통합 시뮬레이션)

CX-Programmer의 시뮬레이션 기능과 터치 패널 화면 툴인 CX-Designer의 오프라인 테스트 기능을 연결하여, CJ 시리즈의 PLC 프로그램과 NS 시리즈의 터치 패널(프로그래밍 가능한 터미널, 이하 PT) 화면 데이터 간의 공유 동작을 디버그할 수 있습니다(PLC-PT 통합 시뮬레이션 기능).
기존 PC와 표시기를 케이블로 연결한 화면 및 화면 제어 래더 프로그램의 디버그 작업을 PC 1대로 처리할 수 있습니다.
CX-Programmer에서도 'PLC-PT 통합 시뮬레이션'을 기동할 수 있습니다.
디버그 조작에 대한 자세한 내용은 "CX-Programmer 조작 설명서"(SBCA-337)를 참조하십시오.

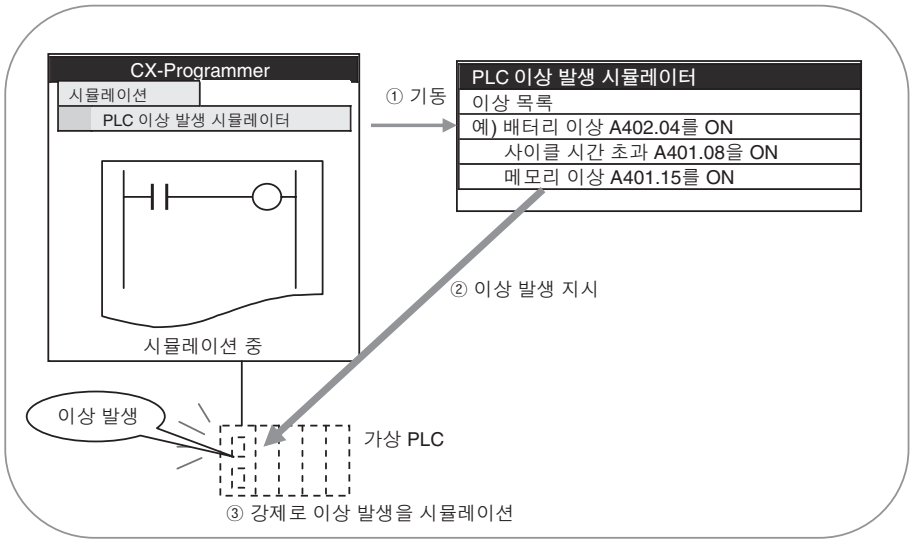


이상 발생 시뮬레이션 기능

CX-Programmer에서는 래더 프로그램 시뮬레이션 중에 가상 PLC의 시스템 이상을 발생시킬 수 있습니다.

임의의 시스템 이상(운전 정지 이상 또는 운전 계속 이상)을 전용 조작 창을 사용하여 발생시키면 PLC의 시스템 이상 발생 시 래더 프로그램 및 NS 터치 패널의 동작을 간단하게 확인할 수 있습니다.

디버그 조작에 대한 자세한 내용은 "CX-Programmer 조작 설명서"(SBCA-337)를 참조하십시오.



(주) 실제 이상과 달리 CX-Simulator에서 이 PLC 이상 발생 시뮬레이션 기능으로 운전 정지 이상을 발생시켜도 동작 중인 래더 실행은 정지하지 않습니다.

참고

FAL/FALS 명령을 사용하여 PLC에서 시스템 이상을 강제로 발생시킬 수도 있습니다.

5-9-4 명령 실행 시 확인

CX-Programmer에서는 명령을 실행할 때 다음과 같은 사항을 확인합니다.

명령을 실행할 때 발생하는 에러에는 다음 4가지가 있습니다.

명령을 실행할 때 발생하는 에러의 종류	발생 시 ON이 되는 플래그	발생 시 운전 정지/계속
1. 명령 처리 에러	에러 플래그(*1)	PLC 시스템 설정에서 정지/계속을 지정합니다 (기본값: 운전 계속). 정지로 지정한 경우만 프로그램 에러가 되어 운전을 정지합니다.
2. 잘못된 영역 액세스 에러	액세스 에러 플래그(*2)	PLC 시스템 설정에서 정지/계속을 지정합니다 (기본값: 운전 계속). 정지로 지정한 경우만 프로그램 에러가 되어 운전을 정지합니다.
3. 잘못된 명령 에러	잘못된 명령 플래그(A295.14)	운전 정지(프로그램 에러)
4. 사용자 프로그램 영역 초과 에러	사용자 프로그램 영역 초과 플래그 (A295.15)	운전 정지(프로그램 에러)

*1 발생 시 운전 정지를 지정한 경우는 명령 처리 에러 플래그(A295.08)가 ON이 됩니다.

*2 발생 시 운전 정지를 지정한 경우는 잘못된 영역 액세스 에러 플래그(A295.10)가 ON이 됩니다.

명령 처리 에러(P_ER 플래그 ON 에러)

- 명령을 실행하려고 했을 때 지정된 데이터 값이 잘못되어, 태스크를 초과하여 명령을 실행하려고 하는 등의 요인에 의해 명령 동작을 중단한 경우를 가리킵니다. 명령 처리의 선두에서 필요한 데이터를 확인하므로 명령을 실행하지 않고 에러 플래그(P_ER)만 ON이 됩니다. 이 경우, N 플래그(P_N), = 플래그(P_EQ)의 동작은 명령에 따라 달라집니다.
  명령(입력계 명령 제외)이 정상 종료되면 에러 플래그(P_ER)는 OFF가 됩니다.
  에러 플래그(P_ER) 성립 조건은 각 명령에 따라 달라집니다. 각 명령에 대해서는 "명령 참조 설명서"(SBCA-351)를 참조하십시오.

- PLC 시스템 설정에서 명령 에러 발생 시 동작을 정지로 설정하면 명령 처리 에러 발생(에러 플래그 ON) 시, 운전을 정지하는(운전 정지 이상) 동시에 '명령 처리 에러 플래그'(A295.08)가 1(ON)이 됩니다.

잘못된 영역 액세스 에러(P_AER 플래그 ON 에러)

- 명령의 피연산자를 지정한 주소에 액세스했을 때 다음과 같이 잘못된 영역을 액세스한 경우를 가리킵니다.
  - 파라미터 영역에 대한 읽기 또는 쓰기 실행
  - 메모리가 장착되지 않은 영역에 대한 쓰기 실행(*)
  - 'EM 파일 메모리'로 지정된 EM 영역에 대한 쓰기 실행
  - 읽기 전용 영역에 대한 쓰기 실행
  - DM/EM 간접 지정(BCD 모드)에서 내용이 BCD 값이 아님(예: *D1의 내용이 #A000일 때)

- 잘못된 영역 액세스 에러가 발생한 경우, 명령 처리가 계속되고 에러 플래그(ER 플래그)는 ON이 되지 않습니다. '액세스 에러 플래그'(P_AER)가 ON이 됩니다.

  * • 현재 EM 뱅크 지정에서 32767을 초과한 주소를 지정한 경우(예: E32768)
    • EM 간접 지정(BIN 모드)에서 최종 뱅크를 지정했을 때 내용이 8000~FFFFHex인 경우 (예: @EC_00001의 내용이 #8000)
    • EM 간접 지정(BIN 모드)에서 현재 뱅크를 지정했을 때 내용이 8000~FFFFHex인 경우 (예: @E1의 내용이 #8000)
    • 인덱스 레지스터(IR) 간접 지정에서 접점의 I/O 메모리 실행 주소를 저장한 IR을 CH로 사용하거나 CH의 I/O 메모리 실행 주소를 저장한 IR을 접점으로 사용

- PLC 시스템 설정에서 명령 에러 발생 시 동작 설정을 정지로 설정하면 잘못된 영역 액세스 에러(P_AER 플래그 ON 에러)가 발생했을 때 운전을 정지(운전 정지 이상)하는 동시에 '잘못된 영역 액세스 에러 플래그'(A295.10)가 1(ON)이 됩니다.

참고

'액세스 에러 플래그'(P_AER 플래그)는 태스크 실행 후 클리어되지 않습니다. 명령 에러 발생 시 동작 설정을 계속으로 설정한 경우, END 명령 직전에 이 플래그를 모니터링하면 해당 태스크의 프로그램 내에서 잘못된 영역 액세스 에러가 발생했는지 여부를 확인할 수 있습니다. (CX-Programmer에서 P_AER 플래그를 모니터링한 경우, 사용자 프로그램 전체를 실행한 후의 마지막 P_AER 플래그의 상태를 모니터링할 수 있습니다.)

기타 에러

● 잘못된 명령 에러

시스템에서 정의한 것 이외의 명령 데이터를 실행하려고 한 경우를 가리킵니다.
CX-Programmer에서 프로그램을 작성하는 한, 일반적으로 이 에러는 발생하지 않습니다.
하지만 이 에러가 발생한 경우 '프로그램 에러'가 되며, 운전을 정지(운전 시 정지 이상)하는 동시에 '잘못된 명령 플래그'(A295.14)가 1(ON)이 됩니다.

● 사용자 프로그램 영역 초과 에러

프로그램 저장 영역으로 정의된 사용자 프로그램 영역 내의 최종 주소를 초과한 위치에 저장된 명령 데이터를 실행하려고 한 경우를 가리킵니다.
CX-Programmer에서 프로그램을 작성하는 한, 일반적으로 이 에러는 발생하지 않습니다.
하지만 이 에러가 발생한 경우 '프로그램 에러'가 되며, 운전을 정지(운전 시 정지 이상)하는 동시에 '사용자 프로그램 영역 초과 플래그'(A295.15)가 1(ON)이 됩니다.



참고

에러 플래그(P_ER) 또는 잘못된 영역 액세스 에러 플래그(P_AER)가 ON인 경우 '프로그램 에러'로 취급하여 CPU 유니트의 운전을 정지시킬 수 있습니다. PLC 시스템 설정으로 지정합니다.

● 프로그램 에러

프로그램 에러의 종류	내용	관련 플래그
END 명령 없음	프로그램 내에 END 명령이 존재하지 않음	END가 없는 플래그(A295.11)가 ON
태스크 실행 시의 에러	해당 사이클 내에 실행 가능 상태인 태스크가 하나도 존재하지 않는 경우 태스크에 할당된 프로그램이 하나도 존재하지 않는 경우 인터럽트 태스크 실행 조건이 성립되었지만 해당하는 번호의 인터럽트 태스크가 존재하지 않는 경우	태스크 에러 플래그(A295.12)가 ON
명령 처리 에러(P_ER 플래그 ON 에러) 및 PLC 시스템 설정에서 명령 에러 발생 시 동작 설정이 1: 정지	명령을 실행하려고 했을 때 피연산자 내의 해당 데이터 값이 잘못된 경우	에러 플래그가 ON이 되고, PLC 시스템 설정에서 명령 에러 발생 시 동작 설정이 1: 정지일 때 명령 처리 에러 플래그(A295.08)가 ON
잘못된 영역 액세스 에러(P_AER 플래그 ON 에러) 및 PLC 시스템 설정에서 명령 에러 발생 시 동작 설정이 1: 정지	파라미터 영역에 대한 읽기 또는 쓰기를 실행한 경우 메모리가 장착되지 않은 영역에 대한 읽기 또는 쓰기를 실행한 경우 EM 파일 메모리로 지정된 EM 영역에 대해 읽기 또는 쓰기를 실행한 경우 읽기 전용 영역에 대해 쓰기를 실행한 경우 DM/EM 간접 지정(BCD 모드)에서 내용이 BCD 값이 아닌 경우	액세스 에러 플래그가 ON이 되고, PLC 시스템 설정에서 명령 에러 발생 시 동작 설정이 1: 정지일 때 잘못된 영역 액세스 에러 플래그(A295.10)가 ON
간접 DM/EM BCD 에러 및 PLC 시스템 설정에서 명령 에러 발생 시 동작 설정이 1: 정지	DM/EM 간접 지정(BCD 모드)에서 내용이 BCD 값이 아닌 경우	액세스 에러 플래그가 ON이 되고, PLC 시스템 설정에서 명령 에러 발생 시 동작 설정이 1: 정지일 때 DM/EM 간접 BCD 에러 플래그(A295.09)가 ON
미분 주소 영역 초과	온라인 편집에서 미분 명령의 삽입/삭제를 계속 반복한 경우(131072회 이상)	미분 초과 플래그(A295.13)가 ON
잘못된 명령 에러	실행할 수 없는 명령을 실행하려고 한 경우	잘못된 명령 플래그(A295.14)가 ON
사용자 프로그램 영역 초과 에러	프로그램 저장 영역으로 정의된 사용자 프로그램 영역 내의 최종 주소를 초과한 위치의 명령을 실행하려고 한 경우	사용자 프로그램 영역 초과 플래그(A295.15)가 ON

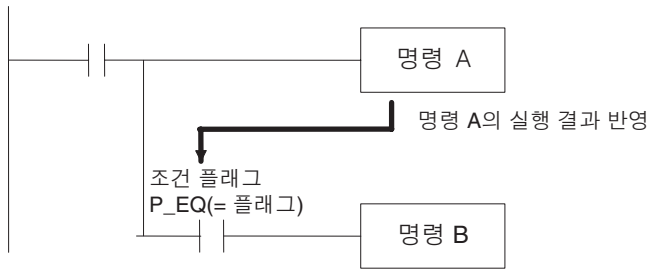
5-10 프로그램 작성 시 주의 사항

5-10-1 조건 플래그 사용 시 주의 사항

조건 플래그 사용 방법

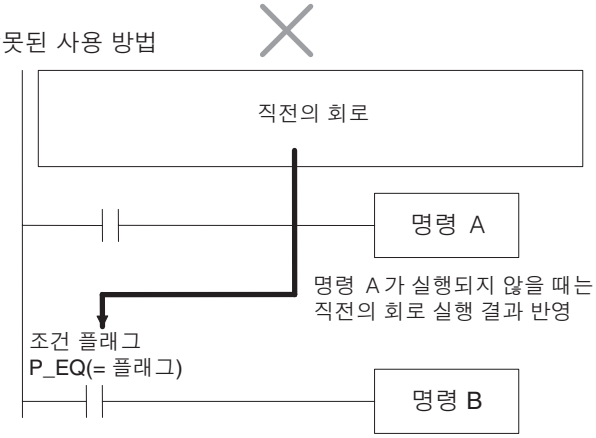
조건 플래그는 명령에서 공통으로 사용되기 때문에 각 명령 실행별로 해당 실행 결과에 따라 1사이클 내에서 바뀝니다.
따라서 한 명령의 실행 결과를 반영할 때는 해당 명령 바로 뒤에서 동일한 실행 조건의 출력 분기를 통해 사용하십시오.
조건 플래그를 모선에 직접 연결하면 다른 명령의 실행 결과를 반영하므로 모선에 직접 연결해서 사용하지 마십시오.
예: 명령 A의 실행 결과를 수신하는 경우
니모닉

올바른 사용 방법



명령 A의 실행 결과에 따라 명령 B를 실행하려고 할 때는 명령 A와 명령 B의 실행 조건(a)을 동일하게 합니다. 그러면 명령 A가 실행되었을 때만 조건 플래그의 상태에 맞춰 명령 B가 실행됩니다.

잘못된 사용 방법

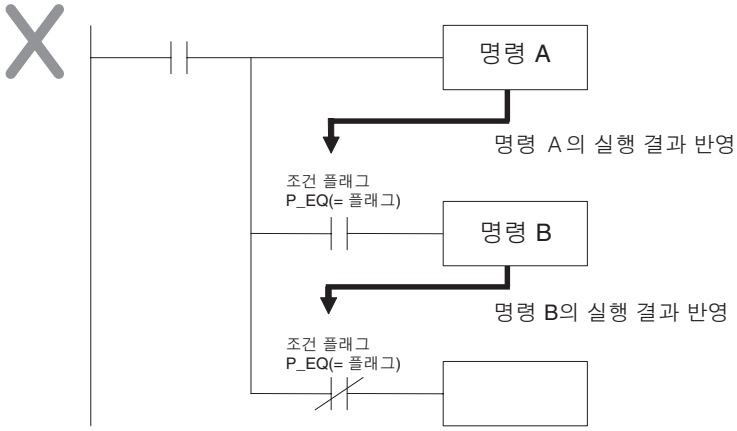


조건 플래그를 좌측 모선에 직접 연결하면 명령 A가 실행되지 않을 때도 이전 회로의 실행 결과에 따라 명령 B가 실행됩니다.

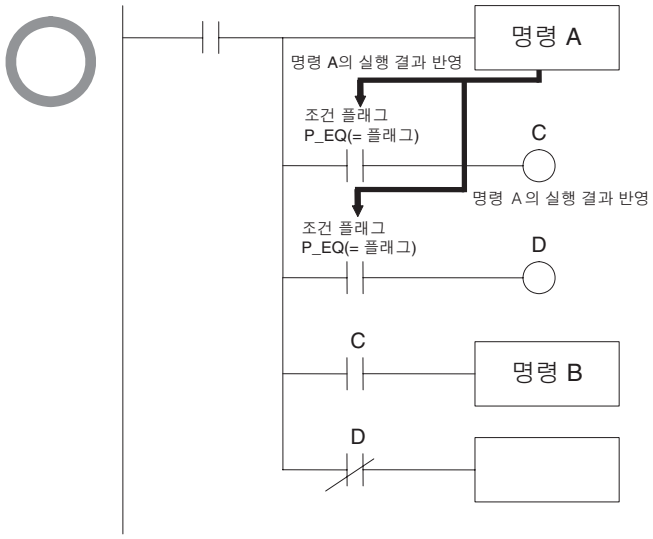
인터럽트 태스크를 사용한 경우 사이클릭 태스크의 명령 실행 중에도 기동 조건에 따라 인터럽트 태스크가 동작합니다. 이때, 인터럽트 태스크 내에서 조건 플래그를 ON/OFF해도 인터럽트 태스크에서 사이클릭 태스크로 복귀했을 때는 조건 플래그도 원래 상태로 돌아와 있습니다.
조건 플래그는 각 명령에서 공통으로 사용하므로 하나의 프로그램 내에서 간섭하지 않도록 주의해야 합니다.

- **명령 A의 실행 결과를 a 접점과 b 접점에서 수신할 때**

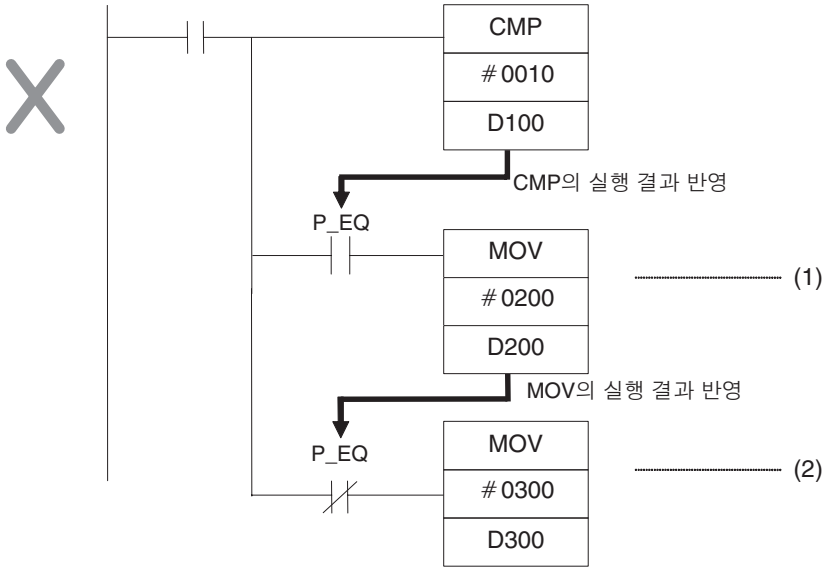
  a 접점과 b 접점을 동일한 출력 분기에서 수신해도 다음 예와 같이 조건 플래그는 명령 B의 실행 결과를 수신합니다.



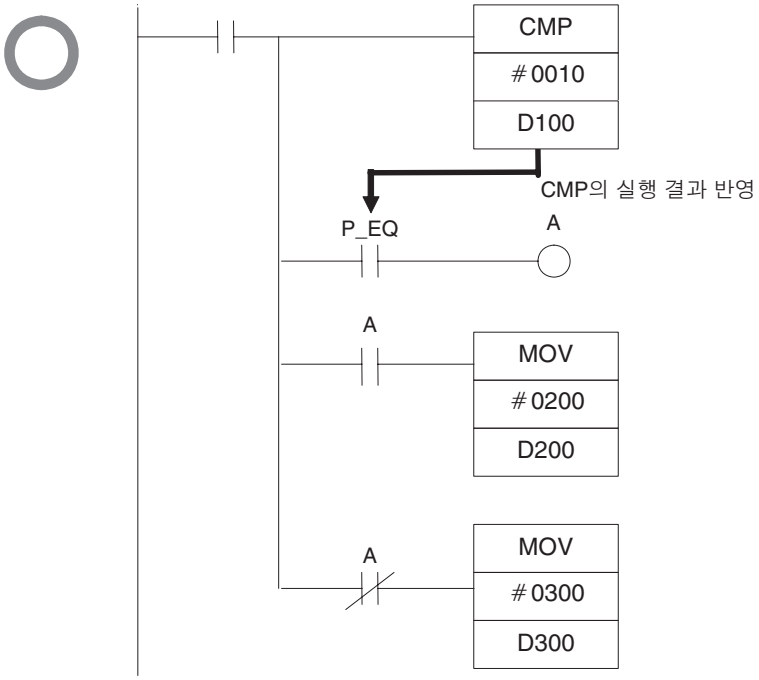
  따라서 명령 B의 실행 결과를 수신하지 않도록 하려면 다음과 같이 각각을 OUT 명령으로 수신하십시오.



예: D100의 내용이 #0010일 때 D200에 #0200 전송.
D100의 내용이 #0010이 아닐 때 D300에 #0300 전송.

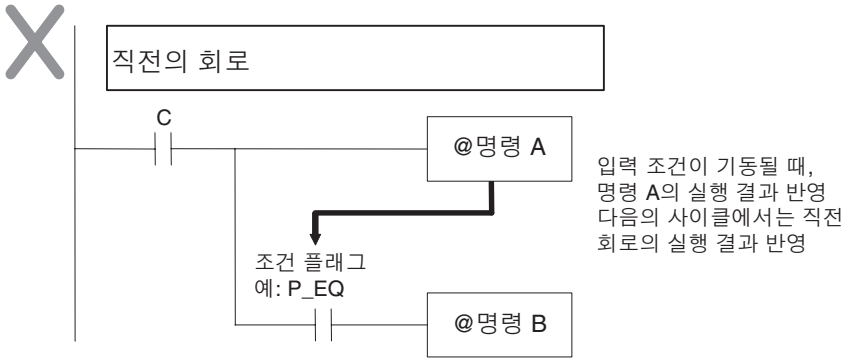


위와 같은 회로에서는 D100의 내용이 #0010일 때 = 플래그가 ON이 되고 (1)에 의해 D200에 #0200이 전송되지만, 그 후 전송 원본 데이터 #0200이 0000Hex가 아니므로 = 플래그가 OFF가 되며, 그 결과 (2)의 MOV 명령이 실행되어 D300에 #0300이 전송됩니다.
따라서 다음과 같이 MOV 명령의 실행 결과를 수신하지 않도록 회로를 구성해야 합니다.

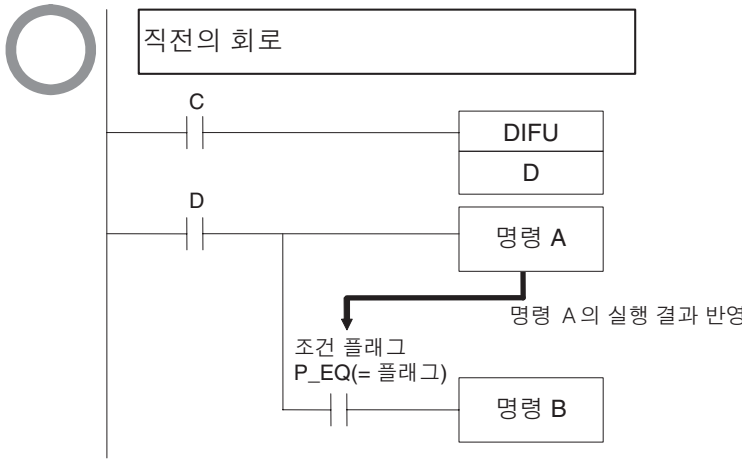


● 입력 미분형 명령의 실행 결과를 수신하는 경우

입력 미분형 명령의 경우, 명령의 실행 결과는 입력 조건의 기동 시에만 조건 플래그에 반영되고 다음 사이클에서는 직전 회로의 결과(해당 명령의 실행 결과가 아님)가 조건 플래그에 반영됩니다.
따라서 입력 미분형 명령의 실행 결과를 수신할 때는 조건 플래그가 다음 사이클에서 어떻게 될지 주의해야 합니다.
예를 들어, 다음의 예에서 입력 조건 C가 기동할 때만 명령 A 및 B를 실행하고, 명령 A의 실행 결과를 명령 B에서 수신하도록 한 경우 다음과 같은 문제가 있습니다.
입력 조건의 기동을 통해 명령 A를 실행한 후 다음 사이클에서도 입력 조건 C가 ON 그대로인 경우, 직전 회로의 반영 결과에 의해 조건 플래그가 OFF→ON으로 되면 명령 B는 입력 조건이 기동되지 않아도 실행됩니다.



따라서 이런 경우에는 다음과 같이 명령 A 및 B를 입력 미분형 명령으로 하지 않고 DIFU(또는 DIFD) 명령으로 수신하여 명령 A와 B를 동시에 기동(또는 종료)시켜 1사이클만 실행합니다.



참고

CCS(조건 플래그 저장) 명령/CCL(조건 플래그 로드) 명령을 사용하면 조건 플래그를 태스크(프로그램) 내의 다른 위치, 태스크간 또는 이후 사이클에서 보관/복구시킬 수 있습니다.

각 조건 플래그의 주요 성립 조건

● 에러 플래그(P_ER)

에러 플래그는 각 명령의 피연산자 데이터가 잘못된 경우 등에서 ON이 됩니다.
에러 플래그가 ON일 때는 해당 명령이 실행되지 않습니다.

에러 플래그가 ON이면 <, >, OF, UF와 같은 다른 조건 플래그는 변화하지 않습니다. =와 N은 각 명령에 따라 동작이 달라집니다.

에러 플래그의 성립 조건에 대해서는 "명령 참조 설명서"(SBCA-351)를 참조하십시오.
명령 실행에 따라 에러 플래그를 무조건 OFF로 하는 명령이 있으므로 주의하십시오.

참고

에러 플래그가 ON이 되었을 때 운전을 정지할지 여부는 PLC 시스템 설정의 '명령 에러 발생 시 동작 설정'으로 지정합니다. 기본적으로 에러 플래그 ON 시 운전을 계속하도록 설정되어 있습니다. 에러 플래그 ON 시 '운전 정지'로 지정하는 경우 프로그램 에러로 취급하여 운전이 정지하면 정지 위치의 프로그램 주소가 특수 보조 릴레이의 A298~299CH에 저장됩니다. 이와 동시에 특수 보조 릴레이의 A295.08이 ON됩니다.

● = 플래그: 등호 플래그(P_EQ)

= 플래그는 비교 결과의 등호(=) 조건 성립 이외에 각종 명령의 임시(일시 저장) 플래그로 시스템에 의해 자동 설정되어 변경됩니다.
일단, 한 명령의 실행 결과에 의해 = 플래그가 ON(OFF)으로 되어도 다른 명령에 의해 OFF(ON)로 된다는 점에 주의해야 합니다.
예를 들어, MOV 등의 전송 명령은 전송 원본 데이터가 0000Hex일 때 = 플래그를 ON으로 하고 그 이외의 경우에는 = 플래그를 OFF로 합니다. 따라서 어떤 명령에 의해 = 플래그를 ON으로 해도 전송 명령을 즉시 실행하면 전송 명령의 전송 원본 데이터가 0000Hex인지 여부에 따라 = 플래그는 ON/OFF됩니다.
명령에 따라 명령 실행만으로 OFF로 하는 경우가 있으므로 주의하십시오.

● CY 플래그: 캐리 플래그(P_CY)

CY 플래그는 이동 명령으로 사용하는 것 이외에 CY 표시가 있는 가산 및 감산 명령의 입력, 가산 및 감산 명령의 오버 플로, 자리 빌림이나 PID 명령, FPD 명령 등에서 사용됩니다.
다음과 같은 경우는 주의해야 합니다.

- 한 명령의 실행 결과에 의해 CY 플래그를 ON(OFF)으로 한 상태로 CY 플래그를 입력으로 사용하는 다른 명령(CY 표시가 있는 가산 및 감산 명령, 이동 명령)을 실행하는 경우
- 한 명령의 실행 결과에 의해 CY 플래그를 ON(OFF)으로 해도 다른 명령에서 OFF(ON)가 되는 경우

● >, < 플래그(P_GT, P_LT)

>, < 플래그는 비교 명령에서 사용하는 것 이외에 LMT, BAND, ZONE, PID 명령 등의 데이터 제어 명령에서 사용됩니다.
일단, 한 명령의 실행 결과에 의해 >, < 플래그가 ON(OFF)으로 되어도 다른 명령에 의해 OFF(ON)로 된다는 점에 주의해야 합니다.

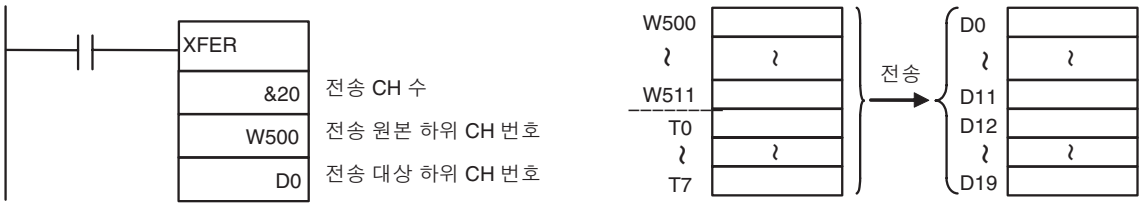
● N 플래그: 부정 플래그(P_N)

N 플래그는 명령 실행 결과에서 CH의 최상위 비트가 "1"이 된 경우 이외에 명령에 따라 명령 실행만으로 무조건 OFF로 하는 경우가 있으므로 주의하십시오.

● 명령의 피연산자에 의한 여러 채널 지정 시 주의 사항

명령의 피연산자로 지정한 여러 채널이 동일 영역 종류가 아닌 경우에도 I/O 메모리 실제 주소의 나열 순서에 따라 실행됩니다. 이때, 에러 플래그가 ON이 되지 않으므로 주의하십시오.
예를 들어, 다음과 같이 XFER(블록 전송) 명령을 실행하는 경우, 전송 원본 데이터 W500CH를 선두로 한 20CH는 내부 보조 릴레이 WR 영역의 최대 주소(W511CH)를 초과하지만 명령을 실행할 때 에러 플래그는 시작되지 않고 XFER 명령은 평상시와 같이 실행됩니다. 이 경우, I/O 메모리 실제 주소에서 WR 영역의 다음 주소는 타이머의 현재값 영역이 되므로 W500~W511CH가 D0~D11에 전송되면 동시에 T0~T7의 현재값이 D12~D19로 전송됩니다.

(주) I/O 메모리 실제 주소의 메모리 맵에 대해서는 '부록-4 I/O 메모리 실제 주소'를 참조하십시오.



5-10-2 특정 프로그램 영역에 관한 주의 사항

일반적인 프로그램 영역 이외에 명령 조건을 성립시키는 특정 프로그램 영역이 있습니다.

구체적으로 다음과 같은 프로그램 영역이 있습니다.

프로그램 영역	명령	성립시키는 명령 조건	상태의 내용
서브루틴 프로그램 영역	SBS 명령/SBN 명령/ RET 명령 GSBS 명령/GSBN 명령/GRET 명령	서브루틴 프로그램 실행 중	SBN－RET 명령 간의 서브루틴 프로그램 영역을 실행하는 상태
IL-ILC 영역	IL 명령/ILC 명령	인터록 중	출력 릴레이 OFF, 타이머 리셋 그 밖의 명령은 실행하지 않고 이전 상태 유지
스텝 래더 영역	STEP 명령		
FOR-NEXT 영역	FOR 명령/NEXT 명령	BREAK 중	반복을 실행하는 상태
JMP0-JME0 영역	JMP0 명령/JME0 명령		점프를 실행하는 상태
블록 프로그램 영역	BPRG 명령/BEND 명령	블록 프로그램 실행 중	BPRG 명령－BEND 명령 간에 니모닉으로 기술한 블록 프로그램 명령 실행

각 특정 프로그램 영역에 대한 명령의 조합 가능 여부

각 특정 프로그램 영역 내의 명령 사용에는 다음과 같은 제한이 있습니다.

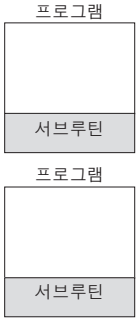
×: 사용 불가능, ○: 사용 가능

	서브루틴 영역 내	IL-ILC 영역 내	스텝 래더 영역 내	FOR-NEXT 영역 내	JMP0-JME0 영역 내	블록 프로그램 영역 내
서브루틴 영역	×	×	×	×	×	×
IL-ILC	○	×	×	○	○	×
스텝 래더 영역	×	○	×	×	○	×
FOR-NEXT 루프	○	○	×	○	○	×
JMP0-JME0	○	○	×	×	×	×
블록 프로그램 영역	○	○	○	×	○	×

(주) 프로그램 영역을 나타내는 명령은 태스크를 초과하여 사용할 수 없습니다. 자세한 내용은 '5-2-4 태스크 설계 방법'(5-30페이지)을 참조하십시오.

서브루틴 영역

서브루틴 영역은 각 프로그램 내의 END 직전에(서브루틴이 여러 개 있는 경우는 묶어서), 서브루틴 이외의 프로그램 다음에 배치하십시오. (따라서, 서브루틴 영역은 스텝 래더 영역, 블록 프로그램 영역, FOR-NEXT 영역, JMP0-JME0 영역 중 어느 영역에도 배치할 수 없습니다.)
서브루틴 영역(SBN~RET) 이후에 서브루틴 이외의 프로그램을 배치한 경우 해당하는 일반 프로그램은 실행되지 않고 무효가 됩니다.



서브루틴 영역 내에서 사용할 수 없는 명령

서브루틴 정의 영역 내에서 사용할 수 없는 명령은 다음과 같습니다.

기능별 분류	니모닉	명령어
공정 진행 제어	STEP	스텝 래더 영역 정의
	SNXT	스텝 래더 진행

(주) 서브루틴 영역 내에 블록 프로그램 영역을 구성할 수 있습니다. 그러나 해당 서브루틴을 호출했을 때 블록 프로그램이 WAIT 상태가 되어 반환된 경우 블록 프로그램의 상태를 유지하기 위해 다음 번 서브루틴 호출 시에는 WAIT 상태에서 실행을 재개합니다.

스텝 래더 프로그램 영역에서 사용할 수 없는 명령

스텝 래더 프로그램 영역 내에서 사용할 수 없는 명령은 다음과 같습니다.

기능별 분류	니모닉	명령어
시퀀스 제어	FOR/NEXT/BREAK	반복 시작/반복 종료/루프 중단
	END	종료
	IL/ILC	인터록/인터록 클리어
	JMP/JME	점프/점프 끝
	CJPL/CJPN	조건부 점프/조건부 부정 점프
	JMP0/JME0	복수 점프/복수 점프 종료
서브루틴	SBN/RET/GSBN/GRET	서브루틴 진입/서브루틴 반환
블록 프로그램	IF(IF NOT)/ELSE/IEND	조건부 분기 블록, 조건부 분기 블록(부정)/조건부 분기 거짓 블록/조건부 분기 블록 끝
	BPRG/BEND	블록 프로그램/블록 프로그램 끝
	EXIT, EXIT NOT	조건부 종료, 조건부 종료(부정)
	LOOP/LEND(LEND NOT)	반복 블록/반복 블록 종료, 반복 블록 종료(부정)
	WAIT(WAIT NOT)	1스캔 조건 대기, 1스캔 조건 대기(부정)
	TIMW	타이머 대기
	TMHW	고속 타이머 대기
	CNTW	카운터 대기
	BPRS/BPPS	블록 프로그램 재기동/블록 프로그램 일시 정지

(주) 1 인터록 영역(IL~ILC) 간에 스텝 래더 프로그램을 사용할 수 있습니다. 인터록을 수행하면 스텝 래더 영역은 모두 리셋됩니다.

2 복수 점프(JMP0)~복수 점프 종료(JME0) 간에 스텝 래더 프로그램 영역을 사용할 수 있습니다.

블록 프로그램 영역에서 사용할 수 없는 명령

블록 프로그램 영역에서 사용할 수 없는 명령은 다음과 같습니다.

기능별 분류	니모닉	명령어
시퀀스 제어	FOR/NEXT/BREAK	반복 시작/반복 종료/루프 중단
	IL/ILC	인터록/인터록 클리어
	JMP0/JME0	복수 점프/복수 점프 종료
	END	종료
시퀀스 입력	UP	전원 흐름. 기동 미분
	DOWN	전원 흐름. 종료 미분
시퀀스 출력	DIFU	기동 미분
	DIFD	종료 미분
	KEEP	유지
	OUT	출력
	OUT NOT	부정 출력
타이머/카운터	TIM	100ms 타이머
	TIMH	10ms 타이머
	TMHH	1ms 타이머
	TIMU	0.1ms 타이머
	TMUH	0.01ms 타이머
	TTIM	적산 타이머
	TIML	장시간 타이머
	MTIM	멀티 출력 타이머
	CNT	카운터
	CNTR	가역 카운터
서브루틴	SBN/RET	서브루틴 진입/서브루틴 반환
데이터 이동	SFT	이동 레지스터
공정 진행 제어	STEP/SNXT	스텝 래더 영역 정의/스텝 래더 진행
데이터 제어	PID	PID 연산
블록 프로그램	BPRG	블록 프로그램
고장 진단	FPD	고장 지점 검출

(주) 1 스텝 래더 프로그램 영역 내에서 블록 프로그램을 사용할 수 있습니다.

2 인터록 영역(IL~ILC 간)에서 블록 프로그램 영역을 사용할 수 있습니다. 단, 인터록 수행 중에는 블록 프로그램 영역이 동작하지 않습니다.

3 복수 점프(JMP0)~복수 점프 끝(JME0) 간에 블록 프로그램 영역을 사용할 수 있습니다.

4 점프 명령(JMP)/조건부 점프 명령(CJP)을 블록 프로그램 영역 내에서 사용할 수 있습니다. 단, 블록 프로그램 영역 내에서 점프 명령(JMP)~점프 끝(JME)/조건부 점프 명령(CJP/CJPN)~점프 끝(JME)이 서로 대응하지 않는 경우 오동작하므로 사용할 수 없습니다.

6

I/O 메모리

이 장에서는 CPU 유니트의 내부 메모리의 일종인 I/O 메모리 내 각 영역의 종류와 자세한 내용에 대해 설명합니다.

6-1 I/O 메모리 영역의 개요 6-2
6-1-1 I/O 메모리 영역이란? 6-2
6-1-2 I/O 메모리 영역 목록 6-3
6-1-3 I/O 메모리 값의 유지 6-5
6-2 입출력 릴레이 6-7
6-3 데이터 링크 릴레이 6-11
6-4 동기 데이터 링크 릴레이 6-12
6-5 CPU 고기능 유니트 릴레이 6-13
6-6 고기능 I/O 유니트 릴레이 6-14
6-7 시리얼 PLC 링크 릴레이 6-15
6-8 DeviceNet 릴레이 6-16
6-9 내부 보조 릴레이 6-17
6-10 유지 릴레이 6-18
6-11 특수 보조 릴레이 6-19
6-12 일시 기억 릴레이 6-20
6-13 데이터 메모리 6-21
6-14 확장 데이터 메모리 6-24
6-15 타이머 6-28
6-16 카운터 6-30
6-17 태스크 플래그 6-31
6-18 인덱스 레지스터 6-32
6-19 데이터 레지스터 6-37
6-20 조건 플래그 6-39
6-21 클록 펄스 6-41

6-1 I/O 메모리 영역의 개요

6-1-1 I/O 메모리 영역이란?

명령의 피연산자를 사용하여 액세스할 수 있는 영역입니다.
I/O 메모리 영역을 구성하는 영역은 다음과 같습니다.

영역 명칭		내용	참조 페이지
CIO(채널 I/O) 영역		주소를 지정할 때 앞에 영문자 기호가 붙지 않는 영역 각 유니트와 I/O 리프레시 등의 데이터 교환이 가능하며, 각 유니트에 할당되지 않는 영역은 내부 보조 릴레이로도 사용 가능	—
	입출력 릴레이	기본 I/O 유니트의 유니트 외부 입출력 단자를 할당하기 위한 릴레이 영역	6-7
	데이터 링크 릴레이	Controller Link 유니트를 통해 네트워크상의 다른 PLC 유니트와 데이터 링크를 수행하는 영역	6-11
	동기 데이터 링크 릴레이	유니트 간 동기 제어 기능을 사용할 때 동기 데이터를 할당하기 위한 릴레이 영역 CJ2H CPU 유니트만 사용 가능	6-12
	CPU 고기능 유니트 릴레이	CPU 고기능 유니트의 상태 정보를 할당하기 위한 릴레이 영역	6-13
	고기능 I/O 유니트 릴레이	고기능 I/O 유니트의 상태 정보를 할당하기 위한 릴레이 영역	6-14
	시리얼 PLC 링크 릴레이	시리얼 PLC 링크에서 사용하는 릴레이 영역 시리얼 포트에서 CPU 유니트 간의 데이터 교환(프로그램 없이 상시 데이터 교환) 실행 CJ2M CPU 유니트만 사용 가능	6-15
	DeviceNet 릴레이	DeviceNet의 원격 I/O 통신(고정 할당)에 의해 각 슬레이브가 할당되는 영역	6-16
	내부 보조 릴레이(*)	프로그램에서만 사용할 수 있는 릴레이 영역	–
내부 보조 릴레이(WR) (*)		프로그램에서만 사용할 수 있는 릴레이 영역	6-17
유지 릴레이(HR)		프로그램에서만 사용할 수 있는 릴레이 영역 전원 OFF 복구(OFF→ON) 또는 모드 전환(프로그램 모드←→운전 모드/모니터 모드 간의 전환) 시에도 ON/OFF 상태 유지	6-18
특수 보조 릴레이(AR)		시스템에서 특정 기능이 할당된 릴레이	6-19
일시 기억 릴레이(TR)		회로의 분기점에서 ON/OFF 상태를 일시적으로 기억하는 릴레이 니모닉에서만 사용 가능	6-20
데이터 메모리(DM)		범용 데이터 영역이며, 비트 단위로 읽기/쓰기 가능 전원 OFF 복구(OFF→ON) 또는 모드 전환(프로그램 모드←→운전 모드/모니터 모드 간의 전환) 시에도 데이터 유지 고기능 유니트의 초기 설정 데이터 할당에도 사용	6-21
확장 데이터 메모리(EM)		범용 데이터 영역이며, 비트 단위로 읽기/쓰기 가능 설정에 따라 지정된 뱅크 이후의 강제 세트/리셋 가능(EM 강제 세트/리셋 기능) 전원 OFF 복구(OFF→ON) 또는 모드 전환(프로그램 모드←→운전 모드/모니터 모드 간의 전환) 시에도 데이터 유지(뱅크라는 여러 개의 영역으로 나뉘어져 있음)	6-24
타이머	타임업 플래그(T)	타이머가 설정 시간을 경과했을 때 ON이 되는 플래그	6-28
	타이머 현재값(T)	타이머 계산용으로 카운트 다운/업하는 영역	
카운터	카운트 업 플래그(C)	카운터가 설정 값을 경과했을 때 ON이 되는 플래그	6-30
	카운터 현재값(C)	카운터 계산용으로 카운트 다운/업하는 영역	
태스크 플래그(TK)		사이클릭 태스크를 실행할 수 있는 상태가 되면 1(ON)이 되는 플래그	6-31
인덱스 레지스터(IR)		레지스터 간접 지정을 위해 I/O 메모리 실제 주소를 저장하는 전용 레지스터 이 레지스터에서 I/O 메모리를 간접 지정합니다.	6-32
데이터 레지스터(DR)		레지스터 간접 지정의 일종으로, 'DR(데이터 레지스터) 오프셋 지정' 시에 사용하는 전용 레지스터	6-37
조건 플래그		에러 플래그 또는 캐리 플래그 등과 같은 각 명령의 실행 결과가 반영되는 전용 플래그 영역	6-39
클록 펄스		정해진 시간마다 시스템이 ON/OFF하여 클록 펄스를 발생시키는 전용 플래그 영역	6-41

* 내부 보조 릴레이에는 CIO(채널 I/O) 영역 내의 내부 보조 릴레이와 내부 보조 릴레이(WR)의 2종류가 있지만 내부 보조 릴레이(WR) 영역을 우선적으로 사용하십시오.

6-1-2 I/O 메모리 영역 목록

○: 가능 △: 조건부 ×: 불가능

명칭		점수	채널 번호	외부 I/O 할당	비트 사용	채널(워드) 사용	읽기/쓰기		주변 툴에서 변경	전원 OFF→ON, 모드 전환 시	강제 세트/리셋	참조 페이지
							읽기	쓰기				
CIO(채널 I/O) 영역	입출력 릴레이	2560점 (160CH)	0~159 (*1)	기본 I/O 유니트	○	○	○	○	○	클리어 (*2)	○	6-7
	데이터 링크 릴레이	3200점 (200CH)	1000~1199	링크 대상 PLC	○	○	○	○	○		○	6-11
	동기 데이터 링크 릴레이	1536점 (96CH)	1200~1295	동기 유니트(*3)	○	○	○	○	○		○	6-12
	CPU 고기능 유니트 릴레이	6400점 (400CH)	1500~1899	CPU 고기능 유니트	○	○	○	○	○		○	6-13
	고기능 I/O 유니트 릴레이	15360점 (960CH)	2000~2959	고기능 I/O 유니트	○	○	○	○	○		○	6-14
	시리얼 PLC 링크 릴레이	1440점 (90CH)	3100~3189	(링크 대상 PLC)	○	○	○	○	○		○	6-15
	DeviceNet 릴레이	9600점 (600CH)	3200~3799	DeviceNet 마스터 (고정 할당 시)	○	○	○	○	○		○	6-16
	내부 보조 릴레이	3200점 (200CH) 37504점 (2344CH)	1300~1499 3800~6143	–	○	○	○	○	○		○	–
내부 보조 릴레이		8192점 (512CH)	W0~W511	–	○	○	○	○	○	클리어 (*2)	○	6-17
유지 릴레이(*4)		8192점 (512CH)	H0~H511	–	○	○	○	○	○	유지	○	6-18
특수 보조 릴레이		48128점 (3008CH)	A0~A447	–	○	○	○	×	×	주소에 따라 클리어/ 유지	×	6-19
			A448~A959	–	○	○	○	○	○			
			A960~ A1471(*5)	–	○	○	○	○	○			
			A10000~ A11535(*5)	–	○	○	○	×	×			

*1 160CH 이상(0160~0999CH)도 랙의 시작 채널로 설정하면 입출력 릴레이로 사용할 수 있습니다.
(랙의 시작 채널은 주변 툴(CX-Programmer)의 I/O 테이블 편집 조작에서 '랙 시작 주소 설정'을 통해서만 설정할 수 있습니다. 랙별로 설정할 수 있는 시작 채널 번호의 설정 범위는 0~900CH입니다.)

*2 IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정한 경우 동작 모드를 변경해도 값이 유지됩니다. 그리고 PLC 시스템 설정에서 '전원 ON 시 IOM(I/O 메모리) 유지 비트 설정'을 유지하도록 설정한 경우에는 전원을 OFF→ON으로 해도 값이 유지됩니다.

*3 CJ2H CPU 유니트만 사용 가능. 유니트 간 동기 제어 기능에 적합한 CPU 고기능 유니트 및 고기능 I/O 유니트

*4 H512~H1535CH는 펑션 블록/SFC 메모리에만 설정할 수 있습니다.

*5 A960~A1471CH, A10000~A11535CH는 CJ2 CPU 유니트에서 확대된 영역이므로 CJ2 CPU 유니트에 적합하지 않은 CPU 고기능 유니트, 고기능 I/O 유니트, 표시기, 소프트웨어 등에서는 액세스할 수 없습니다.
CJ2 CPU 유니트에 적합한 CPU 고기능 유니트, 고기능 I/O 유니트는 다음과 같습니다.

- EtherNet/IP 유니트 CJ1W-EIP21
- 위치 제어 유니트 CJ1W-NC214/234/281/414/434/481/881
- 아날로그 입력 유니트 CJ1W-AD042
- 아날로그 출력 유니트 CJ1W-DA042V
- 시리얼 커뮤니케이션 유니트 CJ1W-SCU22/32/42

○: 가능 △: 조건부 X: 불가능

명칭	점수	채널 번호	외부 I/O 할당	비트 사용	채널 (워드) 사용	읽기/쓰기		주변 툴에서 변경	전원 OFF →ON, 모드 전환 시	강제 세트/ 리셋	참조 페이지
						읽기	쓰기				
일시 기억 릴레이	16개	TR0~TR15	–	○	–	○	○	×	클리어	×	6-20
데이터 메모리	32768CH	D0~D32767	–	○ (*2)	○	○	○	○	유지	×	6-21
확장 데이터 메모리	32768CH (1뱅크당) 최대 25뱅크 (0~18Hex) (*2)	E00_0~E18_ 32767	–	○ (*2)	○	○	○	○	유지	△ (*3)	6-24
타임업 플래그	4096점	T0~T4095	–	○	–	○	○	○	클리어	○	6-28
카운트 업 플래그	4096점	C0~C4095	–	○	–	○	○	○	유지	○	6-30
타이머 현재값	4096CH	T0~T4095	–	–	○	○	○	○	클리어	× (*4)	6-28
카운터 현재값	4096CH	C0~C4095	–	–	○	○	○	○	유지	× (*5)	6-30
태스크 플래그	128점	TK000~TK127	–	○	–	○	×	×	클리어	×	6-31
인덱스 레지스터(*1)	16개	IR0~IR15	–	○	○	△ (간접 지정만 가능)	△ (지정 명령만 해당)	×	클리어	×	6-32
데이터 레지스터 (*1)	16개	DR0~DR15	–	×	○	○	○	×	클리어	×	6-37
조건 플래그	상시 ON 플래그 등	CX-Program-mer의 글로벌 변수 테이블의 시스템 정의 변수 (P_On 등)	–	○	–	○	×	×	클리어	×	6-39
클록 펄스	1초 클록 등	CX-Program-mer의 글로벌 변수 테이블의 시스템 정의 변수 (P_1s 등)	–	○	–	○	×	×	클리어	×	6-41

*1 IR/DR은 태스크에서 개별적으로 또는 태스크 간에 공통으로 사용할 수 있습니다.

*2 확장 데이터 메모리의 뱅크 D~18Hex는 CJ2 CPU 유니트에서 확대된 영역입니다. 또한 데이터 메모리, 확장 데이터 메모리에 대한 비트 사용은 CJ2 CPU 유니트에서 새로 추가된 기능이며, 이 기능을 사용하면 비트 주소 지정이 가능합니다. 그러므로 CJ2 CPU 유니트에 적합하지 않은 CPU 고기능 유니트, 고기능 I/O 유니트, 표시기, 소프트웨어 등에서는 이러한 영역 및 기능을 사용할 수 없습니다.
CJ2 CPU 유니트에 적합한 CPU 고기능 유니트, 고기능 I/O 유니트는 다음과 같습니다.
- EtherNet/IP 유니트 CJ1W-EIP21
- 위치 제어 유니트 CJ1W-NC214/234/281/414/434/481/881
- 아날로그 입력 유니트 CJ1W-AD042
- 아날로그 출력 유니트 CJ1W-DA042V
- 시리얼 커뮤니케이션 유니트 CJ1W-SCU22/32/42

*3 설정에 따라 지정된 뱅크 이후의 강제 세트/리셋이 가능합니다(EM 강제 세트/리셋 기능).
또한 CJ2H CPU 유니트에서는 주소 자동 할당 영역으로 다음과 같은 EM 영역을 사용한 경우에도 해당 뱅크에서 강제 세트/리셋이 가능해집니다.
CJ2H-CPU64/65(-EIP) : E03_0~E03_32767
CJ2H-CPU65(-EIP) : E06_0~E09_32767
CJ2H-CPU67(-EIP) : E07_0~E0E_32767
CJ2H-CPU68(-EIP) : E11_0~E18_32767

*4 타임업 플래그의 강제 세트/리셋을 통해 간접적으로 갱신합니다.

*5 카운트 업 플래그의 강제 세트/리셋을 통해 간접적으로 갱신합니다.

6-1-3 I/O 메모리 값의 유지

CPU 유니트의 동작 상태 변경에 의한 I/O 메모리 값 클리어

I/O 메모리(일부 I/O 메모리 종류 제외)의 값은 CPU 유니트의 동작 상태 변경에 의해 클리어됩니다.

● I/O 메모리 값이 클리어되는 CPU 유니트의 동작 상태

- 동작 모드 변경(프로그램←→운전 또는 모니터 모드) 시
- 운전 정지 이상 발생에 의한 운전 정지 시(FALS 명령 실행에 의한 운전 정지 시는 제외)
- 전원 OFF 복구(ON→OFF→ON) 시

● I/O 메모리 값을 유지하는 설정

CPU 유니트의 동작 상태가 변경되어도 설정에 의해 I/O 메모리 값을 유지할 수 있습니다.

- 동작 모드 변경 시, 운전 정지 이상 발생에 따른 운전 정지 시의 I/O 메모리 값 유지
  특수 보조 릴레이의 IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정합니다.
- 전원 OFF 복구 시의 I/O 메모리 값 유지
  PLC 시스템 설정의 전원 ON 시 설정에서 'IOM(I/O 메모리) 유지 비트 유지'를 선택한 상태로 특수 보조 릴레이의 IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정합니다.

사용상의 주의 사항

입출력 릴레이의 I/O 메모리 값을 유지하는 것으로 설정했을 때의 주의 사항

CPU 유니트가 '운전'/'모니터' 모드에서 '프로그램' 모드로 변경되면 출력 릴레이의 I/O 메모리 값은 클리어(OFF)되지 않고 '프로그램' 모드로 변경되기 전의 상태를 유지합니다.
따라서 다시 '프로그램' 모드에서 '운전'/'모니터' 모드로 변경한 경우 '프로그램' 모드로 변경하기 전의 I/O 메모리 값이 출력되므로 주의하십시오.

단, 운전 정지 이상(FALS 명령 실행 포함)에 의해 운전이 정지됐을 때는 CPU 유니트 내부의 I/O 메모리 값은 유지되지만 출력 유니트의 출력 접점은 모두 OFF가 됩니다.

● I/O 메모리 종류에 따른 I/O 메모리 값의 유지 상태

I/O 메모리 종류별 I/O 메모리 값의 유지 상태는 다음과 같습니다.

◎: 유지 ○: 설정에 따라 유지 가능 ×: 클리어

명칭		CPU 유니트 동작 모드 변경 시	운전 정지 이상 시		전원 ON 시
			FALS 명령 실행	기타 운전 정지	
CIO (채널 I/O)	입출력 릴레이	○(*2) (설정에 따름)	◎	○(*2) (설정에 따름)	○(*3) (설정에 따름)
	데이터 링크 릴레이				
	동기 데이터 링크 릴레이 (*1)				
	CPU 고기능 유니트 릴레이				
	고기능 I/O 유니트 릴레이				
	시리얼 PLC 링크 릴레이 (*4)				
	DeviceNet 릴레이				
	내부 보조 릴레이				
내부 보조 릴레이(W)		○(*2) (설정에 따름)	◎	○(*2) (설정에 따름)	○(*3) (설정에 따름)
유지 릴레이(H)		◎			
특수 보조 릴레이(A)		주소에 따라 유지/클리어			
데이터 메모리(D)		◎			
확장 데이터 메모리(E)		◎			
타이머 플래그(T)		○(*2) (설정에 따름)	◎	○(*2) (설정에 따름)	○(*3) (설정에 따름)
타이머 현재값(T)		○(*2) (설정에 따름)	◎	○(*2) (설정에 따름)	○(*3) (설정에 따름)
카운터 플래그(C)		◎			
카운터 현재값(C)		◎			
태스크 플래그(TK)		×	◎	×	×
인덱스 레지스터(IR)		×	◎	×	×
데이터 레지스터(DR)		×	◎	×	×

*1 CJ2H CPU 유니트만 사용 가능.

*2 특수 보조 릴레이의 IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정합니다.

*3 PLC 시스템 설정의 전원 ON 시 설정에서 'IOM(I/O 메모리) 유지 비트 유지'를 선택한 상태로 특수 보조 릴레이의 IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정합니다.

*4 CJ2M CPU 유니트만 사용 가능.

6-2 입출력 릴레이

0.00~159.15(0~159CH)

기본 I/O 유니트의 입출력 단자를 할당할 수 있는 릴레이 영역입니다.
연결 위치에 따라(CPU 유니트에서 가장 가까운 위치부터, 왼쪽에서 오른쪽으로) 각 기본 I/O 유니트의 점유 채널 수만큼 연속 번호로 16점 단위로 할당할 수 있습니다.
할당할 수 없는 영역은 내부 보조 릴레이와 동일하게 프로그램에서만 사용 가능합니다.
또한 강제 세트/리셋이 가능합니다.

(주) 랙 시작 채널을 변경하면 0~999CH를 입출력 릴레이로 외부 할당하여 사용할 수 있습니다. 단, 그 경우에도 외부에 할당할 수 있는 점수는 최대 2560점(160CH)입니다.

입력 릴레이(기본 입력 유니트에 할당된 경우)

푸시 버튼 스위치, 리미트 스위치, 광전 스위치 등과 같은 외부 기기의 ON/OFF 상태가 반영됩니다.
반영 타이밍에는 다음 3가지 방법이 있습니다.

● 평상시(I/O 리프레시 시)

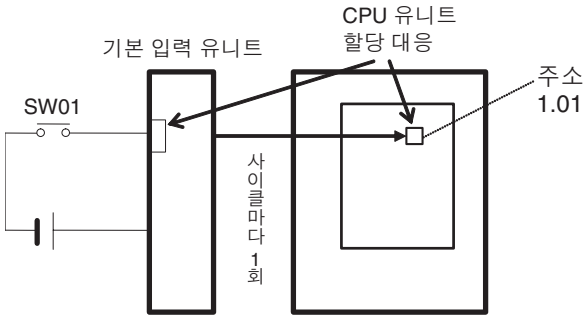
사이클마다 1회, 사용자 프로그램을 실행한 후 할당된 외부 기기의 ON/OFF 상태를 일괄적으로 전송합니다. 그 외의 타이밍에서는 바뀌지 않습니다.

예:

래더	니모닉
1.01	LD 1.01



사이클마다 1회, 1.01에 할당된 기본 입력 유니트의 입력 단자에 연결된 외부 기기 SW1의 ON/OFF 상태가 반영됩니다.



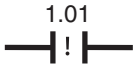
● 매회 리프레시 시

명령의 피연산자로 입력 릴레이를 지정한 경우, 명령어 앞에 ! 기호를 붙이면 다음과 같이 리프레시됩니다.

- 비트 지정 명령의 경우
  명령 실행 직전에 해당 비트를 포함한 16비트의 할당된 외부 기기 ON/OFF 상태를 전송합니다.
- 채널 지정 명령의 경우
  명령 실행 직전에 지정된 채널에 할당된 외부 기기의 ON/OFF 상태를 전송합니다.

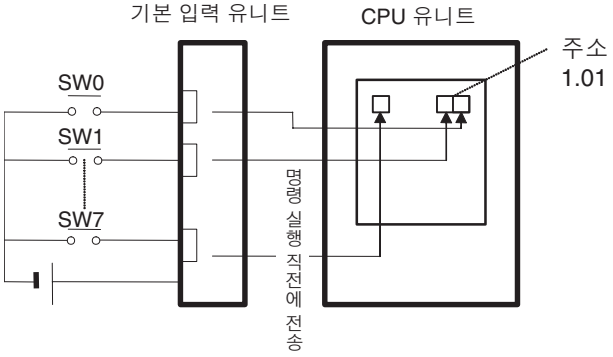
이것을 매회 리프레시라고 합니다.
이렇게 매회 리프레시한 입력 릴레이는 매 사이클의 I/O 리프레시 시에도 할당된 외부 기기의 상태가 반영됩니다.

래더 니모닉



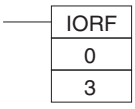
! LD 1.01

LD 1.01을 실행하기 직전에 할당된 기본 입력 유니트의 입력 단자에 연결된 외부 기기 SW1의 ON/OFF 상태가 전송되어 반영됩니다.

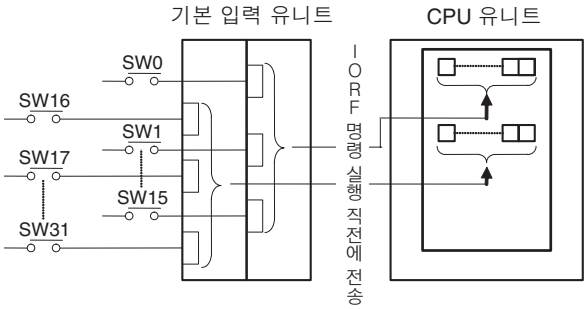


● IORF 명령 실행 시

IORF(I/O 리프레시) 명령을 실행할 때는 지정된 범위의 입력 릴레이에 할당된 외부 기기의 ON/OFF 상태를 전송합니다.
예:

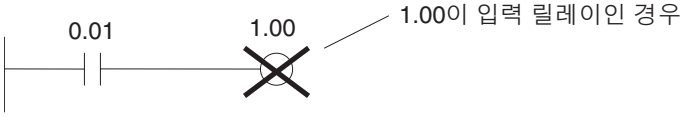


IORF 명령 실행 시 입력 릴레이의 경우, 0CH ~ 3CH까지 4CH 데이터를 입력합니다. 출력 릴레이의 경우는 출력합니다.



주소 0CH ~ 1CH를 외부에서 전송합니다. (2CH ~ 3CH는 출력 릴레이인 경우)

- 프로그램상의 릴레이 번호 사용 순서나 a 접점, b 접점의 사용 횟수에는 제한이 없습니다.
- 출력 명령의 피연산자에는 사용할 수 없습니다.



출력 릴레이(기본 출력 유니트에 할당된 경우)

이 ON/OFF 상태가 액츄에이터 등의 외부 기기로 출력됩니다.
출력 타이밍에는 다음 3가지 방법이 있습니다.

● 평상시(I/O 리프레시 시)

사이클마다 1회, 사용자 프로그램을 실행한 후 ON/OFF 상태가 할당된 외부 기기에 일괄적으로 출력됩니다. 그 외의 타이밍에서는 출력되지 않습니다.

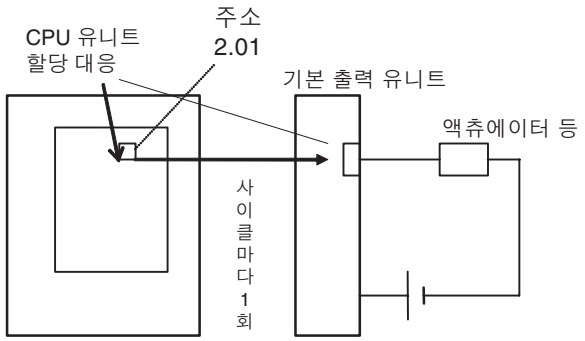
예:

래더 니모닉

2.01 OUT 2.01



사이클마다 1회, 2.01에 할당된 기본 출력 유니트의 출력 단자에 연결된 외부 기기에 출력됩니다.



● 매회 리프레시 시

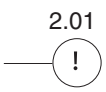
명령의 피연산자로 출력 릴레이를 지정한 경우, 명령어 앞에 ! 기호를 붙이면 다음과 같이 출력됩니다.

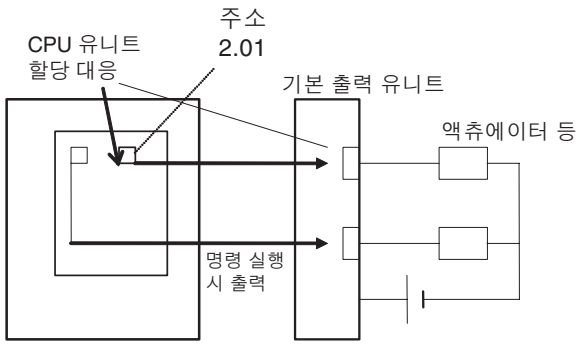
- 비트 지정 명령의 경우
  명령 실행 직후에 해당 비트가 포함되는 16비트 ON/OFF 상태를 할당된 외부 기기에 출력합니다.
- 채널 지정 명령의 경우
  명령 실행 직후 지정된 채널의 ON/OFF 상태를 할당된 외부 기기에 출력합니다.

이를 매회 리프레시이라고 합니다(또한 이렇게 매회 리프레시한 출력 릴레이는 매 사이클의 I/O 리프레시 시에도 할당된 외부 기기에 출력됩니다).

래더 니모닉

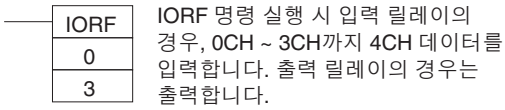
2.01 ! OUT 2.01

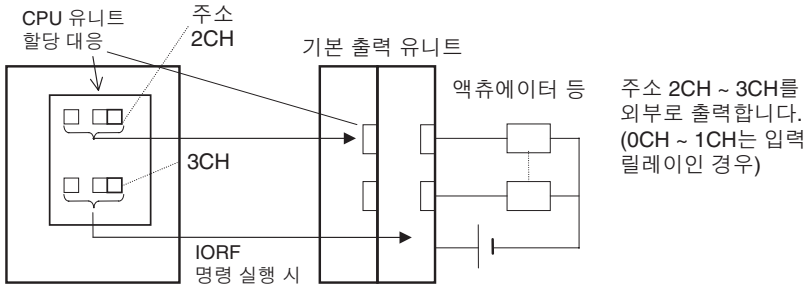




● IORF 명령 실행 시

IORF(I/O 리프레시) 명령을 실행할 때는 지정된 범위의 출력 릴레이에 할당된 외부 기기의 ON/OFF 상태를 출력합니다.
예:



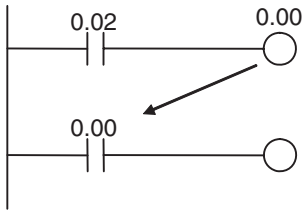


사용상의 주의 사항

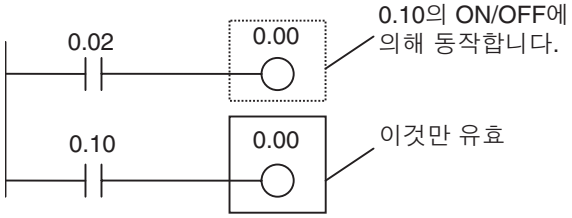
부하 차단 플래그(A500.15)를 1(ON)로 설정한 경우, 할당된 기본 I/O 유니트, 고기능 I/O 유니트의 실제 출력은 모두 OFF가 됩니다. 단, I/O 메모리의 출력 릴레이는 클리어되지 않습니다.

입력 릴레이/출력 릴레이 공통

- 프로그램상에서 릴레이 번호의 사용 순서에는 제한이 없습니다.
- 입력 명령의 피연산자에 사용할 수 있으며 a 접점, b 접점의 사용 횟수에는 제한이 없습니다.



- 출력 명령의 피연산자에 동일 주소를 중복해서 사용할 수 없습니다. 중복 사용한 경우에는 마지막 회로에 의한 출력 동작만 유효하게 됩니다.



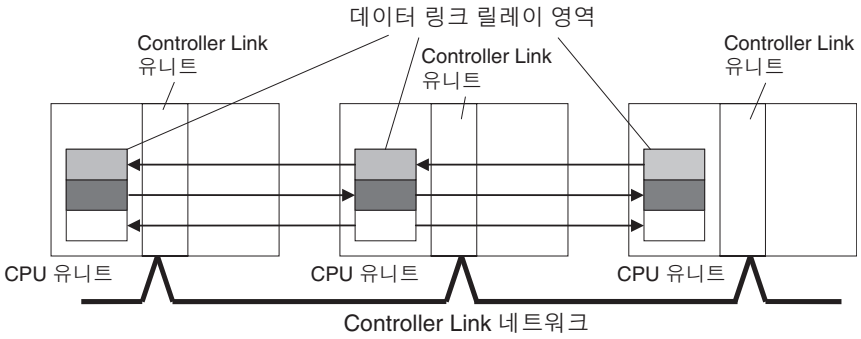
6-3 데이터 링크 릴레이

1000.00~1199.15(1000~1199CH)

Controller Link 자동 설정 데이터 링크에서 영역 종류를 '링크 릴레이'(LR)로 설정했을 때 사용하는 릴레이 영역입니다.

Controller Link 유니트를 통해 네트워크의 다른 CPU 유니트의 데이터 링크 영역과 데이터를 공유(프로그램 없이 상시 데이터 교환)합니다.
데이터 링크 영역의 할당 방법에는 자동 설정(1노드당 송신 채널 수는 동일)과 임의 설정(노드별로 할당 영역을 자유롭게 설정, 1노드당 송신 채널 수는 임의, 송신만 또는 수신만 하도록 설정 가능)이 있습니다.
이 데이터 링크 릴레이는 자동 설정인 경우 데이터 링크 영역으로 자동 할당됩니다. (임의 설정인 경우에는 사용자 정의하여 이 릴레이를 할당하는 것도 가능합니다.)
자세한 내용은 "Controller Link 유니트 사용 설명서"(SCCC-326)를 참조하십시오.

Controller Link 네트워크의 데이터 링크에서 영역 종류를 링크 릴레이로 지정하지 않는 경우, 내부 보조 릴레이와 동일하게 프로그램상에서만 사용할 수 있습니다.



또한 강제 세트/리셋이 가능합니다.

6-4 동기 데이터 링크 릴레이

1200~1295CH

CJ2H CPU 유니트에서 유니트 간 동기 제어 기능을 사용할 때 CPU 유니트와 각 동기 유니트 사이에서 데이터를 주고받기 위한 릴레이 영역입니다.
CJ2H CPU 유니트만 사용할 수 있습니다.
자세한 내용은 '10-8-4 동기 데이터 링크 릴레이'를 참조하십시오.

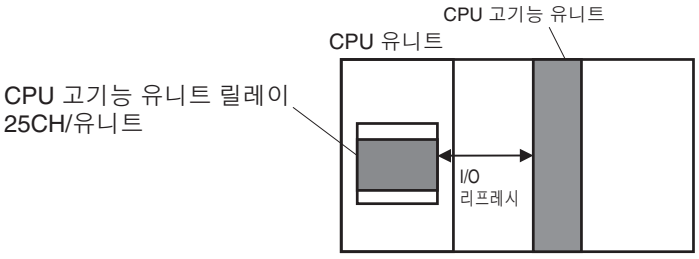
6-5 CPU 고기능 유니트 릴레이

1500~1899CH

CPU 고기능 유니트의 상태 정보 등을 할당할 수 있는 릴레이 영역입니다. 1유니트당 25CH가 유니트 번호에 따라 할당됩니다.

CPU 고기능 유니트와의 데이터 교환 타이밍은 다음 2가지 방법이 있습니다.

- 평상시(I/O 리프레시 시)
- DLNK 명령 실행 시



할당 영역은 다음과 같습니다.

채널 번호	유니트 번호
1500~1524CH	0
1525~1549CH	1
1550~1574CH	2
1575~1599CH	3
1600~1624CH	4
1625~1649CH	5
1650~1674CH	6
1675~1699CH	7
1700~1724CH	8
1725~1749CH	9
1750~1774CH	A
1775~1799CH	B
1800~1824CH	C
1825~1849CH	D
1850~1874CH	E
1875~1899CH	F

CPU 고기능 유니트의 할당 영역 내용은 각 유니트에 따라 다릅니다. 그러므로 각 유니트의 사용 설명서를 참조하십시오.

CPU 고기능 유니트의 할당 영역으로 사용하지 않는 경우에는 내부 보조 릴레이와 동일하게 프로그램상에서만 사용할 수 있습니다.

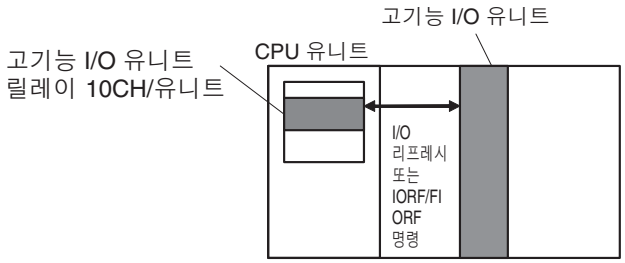
6-6 고기능 I/O 유니트 릴레이

2000~2959CH

고기능 I/O 유니트의 상태 정보 등을 할당할 수 있는 릴레이 영역입니다. 1유니트당 10CH로, 총 96유니트(0~95호기) 분량이 기기 번호에 따라 할당됩니다.

고기능 I/O 유니트와의 데이터 교환 타이밍에는 다음 3가지 방법이 있습니다.

- 평상시(I/O 리프레시 시)
- IORF 명령 실행 시
- FIORF 명령 실행 시



할당 영역은 다음과 같습니다.

채널 번호	기기 번호(Hex)
2000~2009CH	0호기
2010~2019CH	1호기
2020~2029CH	2호기
2030~2039CH	3호기
2040~2049CH	4호기
2050~2059CH	5호기
2060~2069CH	6호기
2070~2079CH	7호기
2080~2089CH	8호기
2090~2099CH	9호기
2100~2109CH	10호기
2110~2119CH	11호기
2120~2129CH	12호기
2130~2139CH	13호기
2140~2149CH	14호기
2150~2159CH	15호기
2160~2169CH	16호기
2170~2179CH	17호기
⋮	⋮
2950~2959CH	95호기

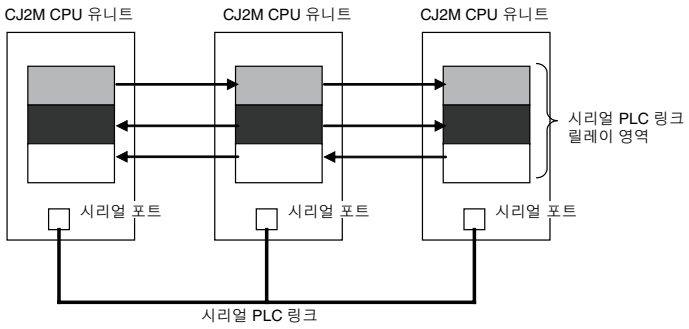
고기능 I/O 유니트의 할당 영역으로 사용하지 않는 경우에는 내부 보조 릴레이와 동일하게 프로그램상에서만 사용할 수 있습니다.

6-7 시리얼 PLC 링크 릴레이

3100~3189CH

시리얼 PLC 링크에서 사용하는 릴레이 영역으로, 다른 PLC와의 데이터 링크에 사용합니다.
시리얼 PLC 링크는 시리얼 포트에서 CPU 유니트 간의 데이터 교환(프로그램 없이 상시 데이터 교환)을 수행하는 기능입니다.

시리얼 PLC 링크 영역의 할당은 친국에 설정된 다음과 같은 PLC 시스템 설정에 따라 자동으로 설정됩니다.
- 시리얼 PLC 링크 모드
- 시리얼 PLC 링크 송신 채널 수
- 시리얼 PLC 링크의 최대 기기 번호



시리얼 PLC 링크에서 사용하지 않는 릴레이 번호는 '내부 보조 릴레이'와 동일하게 프로그램상에서만 사용할 수 있습니다.
또한 강제 세트/리셋이 가능합니다.

다음과 같은 경우 클리어됩니다.
- 동작 모드 변경(프로그램←→운전 또는 모니터 모드) 시
- 전원 OFF 복구(ON→OFF→ON) 시
- 주변 툴(CX-Programmer)에 의한 클리어 조작 시
- FALS 명령 실행 이외의 운전 정지 이상 발생에 의한 운전 정지 시(FALS 명령 실행에 의한 운전 정지 시에는 유지됩니다.)

● IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정한 경우

동작 모드 변경(프로그램←→운전 또는 모니터 모드) 시 또는 운전 정지 이상(FALS 명령 실행 포함)에 의한 운전 정지 시에도 값이 유지됩니다. 단, 전원 OFF 복구 시에는 클리어됩니다.

● 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 경우

전원 OFF 복구 시에도 유지됩니다.

6-8 DeviceNet 릴레이

3200~3799CH

DeviceNet 유니트의 원격 I/O 마스터 기능을 사용할 때 각 슬레이브가 할당되는 릴레이 영역입니다(고정 할당 시).

할당 릴레이 영역의 소프트 스위치 내 고정 할당 영역 설정 1/2/3 스위치를 사용하여 다음과 같은 고정 할당 영역 1/2/3 중 하나를 선택할 수 있습니다.

영역	마스터→슬레이브 출력(OUT) 영역	슬레이브→마스터 입력(IN) 영역
고정 할당 영역 1	3200~3263CH	3300~3363CH
고정 할당 영역 2	3400~3463CH	3500~3563CH
고정 할당 영역 3	3600~3663CH	3700~3763CH

DeviceNet 유니트의 원격 I/O 슬레이브 기능을 고정 할당으로 사용하는 경우에도 이 DeviceNet 릴레이에 할당됩니다.

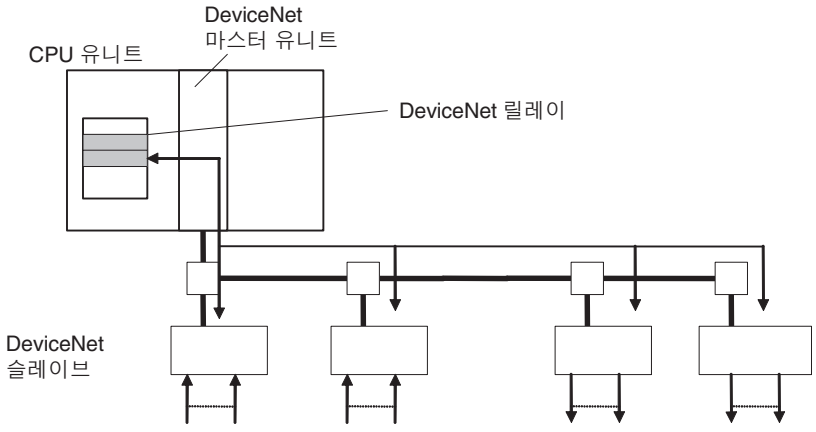
영역	마스터→슬레이브 출력(OUT) 영역	슬레이브→마스터 입력(IN) 영역
고정 할당 영역 1	3370CH	3270CH
고정 할당 영역 2	3570CH	3470CH
고정 할당 영역 3	3770CH	3670CH

DeviceNet 유니트를 통해 네트워크상의 각 슬레이브와 프로그램 없이 상시 데이터 교환을 실행합니다.

참고

각 슬레이브의 할당 방법에는 고정 할당(노드 주소에 의해 할당 영역 고정)과 자유 할당(할당 영역을 자유롭게 설정)이 있습니다.

- 고정 할당인 경우, 각 슬레이브가 자동으로 노드 주소 순서에 따라 이 DeviceNet 릴레이의 고정 할당 1~3 중 하나에 할당됩니다.
- 자유 할당인 경우에는 다음과 같은 범위에서 각 슬레이브의 주소를 할당할 수 있습니다.
  CIO: 0~6143CH
  WR: W0~511CH
  HR: H0~H511CH
  DM: D0~D32767
  EM: E00_0~EC_32767(0~18Hex(16진수) 가능)
  EM 뱅크 D Hex 이후는 지정할 수 없습니다.
  자세한 내용은 "DeviceNet 유니트 사용 설명서"(SBCD-314)를 참조하십시오.



고정 할당인 경우 노드 주소 순서로 할당됩니다. (단, 2CH 이상의 슬레이브는 CH 수만큼 노드 주소 수를 점유합니다.)

6-9 내부 보조 릴레이

W0~W511CH의 512CH

프로그램에서만 사용할 수 있는 릴레이입니다. 외부 입출력 단자를 통한 입출력은 수행할 수 없습니다.
내부 보조 릴레이는 기본적으로 이 영역부터 사용하십시오.
내부 보조 릴레이에는 다음의 2종류가 있습니다.

- W0~W511CH
- 1300~1499CH, 3800~6143CH

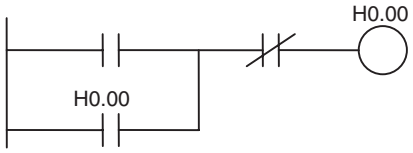
두 영역은 다음과 같은 차이가 있습니다.
W0~W511CH는 이후 CPU 유니트의 기능을 확장해도 다른 특정 용도로 할당되지 않는 영역입니다. 그에 비해 1300~1499CH, 3800~6143CH의 영역은 기능을 확장하면 다른 특정 용도로 할당할 수 있습니다. 따라서 기본적으로 내부 보조 릴레이에는 이 W0~W511CH를 우선적으로 사용할 것을 권장합니다.

6-10 유지 릴레이

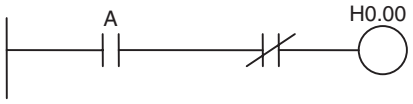
H0~H511CH

프로그램에서만 사용할 수 있으며, 전원 OFF 복구(ON→OFF→ON) 또는 동작 모드 변경(프로그램 모드←→운전 또는 모니터 모드) 시에도 전원 OFF 직전 또는 모드 변경 직전의 ON/OFF 상태를 유지하는 릴레이입니다.
IL-ILC 내에서 IL 조건이 OFF일 때 클리어됩니다. IL 조건이 OFF일 때도 ON으로 유지하고 싶은 영역은 IL 명령 전에 SET 명령을 사용하여 ON으로 설정하십시오.

유지 릴레이를 사용하여 자체 유지 회로를 구성하면 전원 OFF 복구 시에도 자체 유지 기능이 해제되지 않습니다.



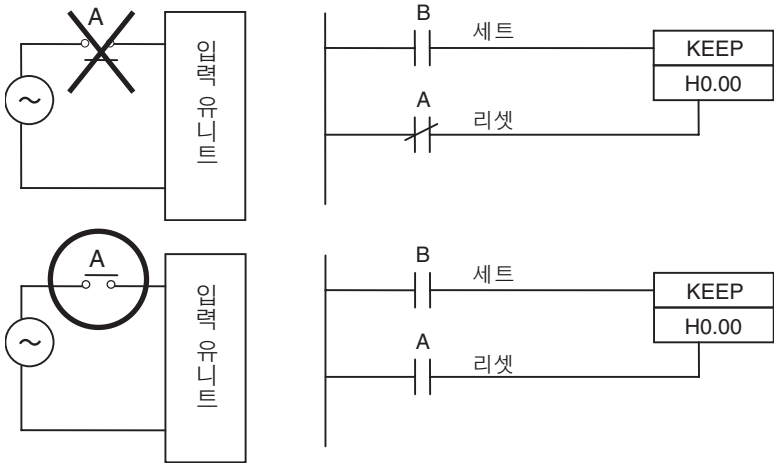
유지 릴레이를 사용하지 않는 경우에는 자체 유지 회로를 구성해도 전원 OFF 복구 시 릴레이가 OFF로 되어 자체 유지가 해제됩니다.
또한 유지 릴레이를 사용하고 있어도 자체 유지 회로를 구성하지 않은 경우는 유지 릴레이에서 다음 입력 조건 A에 의해 전원 OFF 복구 시 OFF가 됩니다.



H512~H1535CH는 펑션 블록/SFC 메모리에만 설정할 수 있습니다.
사용자 프로그램의 명령에 대한 피연산자로 지정할 수는 없습니다.

사용상의 주의 사항

유지 릴레이를 KEEP 명령에서 사용할 때 리셋 입력의 접점을 b 접점으로 설정하지 마십시오. 전원 OFF 시 또는 순간 정전 시, PLC 본체의 내부 전원에서 입력만 먼저 전원이 끊어져서 유지 릴레이가 리셋됩니다.



프로그램상의 릴레이 번호 사용 순서나 a, b 접점의 사용 횟수에는 제한이 없습니다.

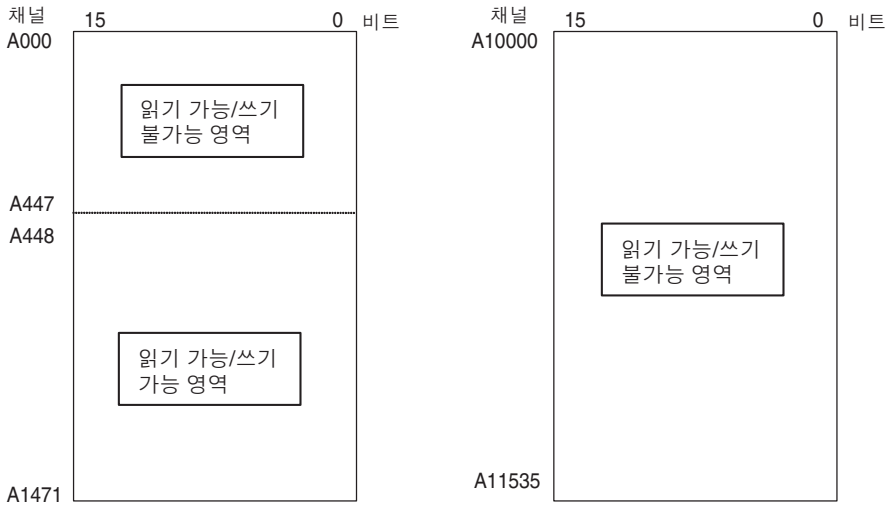
6-11 특수 보조 릴레이

A0~A1471CH
A10000~A11535CH

시스템에서 특정 기능이 할당된 릴레이입니다.
시스템이 자동으로 설정하는 것과 사용자가 설정하여 조작하는 것이 있습니다.
자가 진단에 의한 이상 플래그, 초기 설정 플래그, 조작 플래그, 운전 상태 모니터 데이터 등으로 구성됩니다.
특수 보조 릴레이의 각 플래그에 대한 자세한 내용은 '부록-3 특수 보조 릴레이 목록'을 참조하십시오.

프로그램 또는 CX-Programmer에서 이 릴레이 영역을 읽고 쓰는 데 사용합니다. 또한 이 릴레이 영역에는 읽기 전용 영역(A000~A447, A10000~A11535CH)과 읽기/쓰기가 가능한 영역(A448~A1471CH)이 있습니다.
A960~A1471CH, A10000~A11535CH는 CJ2 CPU 유니트에서 확대된 영역이므로 CJ2 CPU 유니트에 적합하지 않은 CPU 고기능 유니트, 고기능 I/O 유니트, 표시기, 소프트웨어 등에서는 액세스할 수 없습니다.
CJ2 CPU 유니트에 적합한 CPU 고기능 유니트, 고기능 I/O 유니트는 다음과 같습니다.

- EtherNet/IP 유니트 CJ1W-EIP21
- 위치 제어 유니트 CJ1W-NC214/234/281/414/434/481/881
- 아날로그 입력 유니트 CJ1W-AD042
- 아날로그 출력 유니트 CJ1W-DA042V
- 시리얼 커뮤니케이션 유니트 CJ1W-SCU22/32/42



CX-Programmer에서 쓰는 경우 다음과 같이 조작하십시오.

- 온라인 연결 중의 세트/리셋 조작, [주소 모니터]에서 현재값 변경([값 설정] 대화 상자) 또는 PLC 데이터 테이블 편집 후 PLC로 전송 조작(자세한 내용은 "CX-Programmer 조작 설명서"(SBCA-337)를 참조하십시오.).

사용상의 주의 사항

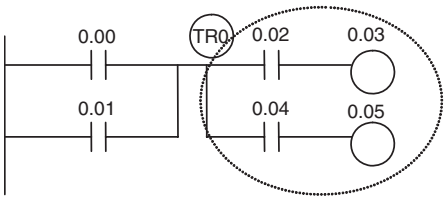
특수 보조 릴레이(AR)의 정의되지 않은 영역은 앞으로 CPU 유니트의 버전 업그레이드에 의해 기능이 할당될 가능성이 있습니다. 그러므로 사용자 프로그램에서 내부 보조 릴레이로 사용하지 마십시오.

6-12 일시 기억 릴레이

TR0~TR15

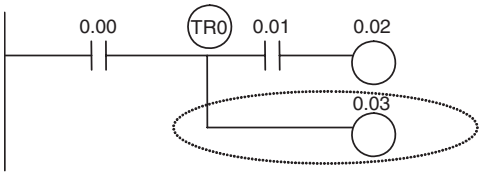
회로의 분기점에서 ON/OFF 상태를 일시적으로 기억하는 릴레이로, 니모닉에서만 사용합니다.
TR0~TR15의 사용 순서나 사용 횟수에는 제한이 없습니다.
OUT 명령과 LD 명령만으로 사용할 수 있습니다. OUT 명령(OUT TR0~15)은 분기점의 ON/OFF 상태를 기억합니다. LD 명령(LD TR0~15)은 기억한 분기점의 ON/OFF 상태를 읽어옵니다.
동일 회로 내에서 일시 기억 릴레이 번호를 중복 사용할 수 없습니다(단, 회로 간에는 중복 사용이 가능합니다).
CX-Programmer에서 변경할 수 없습니다.
일시 기억 릴레이는 다음과 같은 경우에 사용합니다.

최종 분기점 이후이고 접점이 직접 연결된 출력이 2개 이상 있을 때



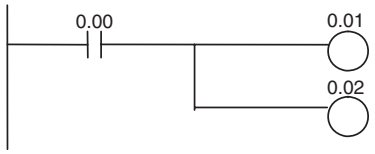
명령	데이터
LD	0.00
OR	0.01
OUT	TR 0
AND	0.02
OUT	0.03
LD	TR 0
AND	0.04
OUT	0.05

접점이 직접 연결된 출력의 후단에 접점이 없는 출력이 있을 때

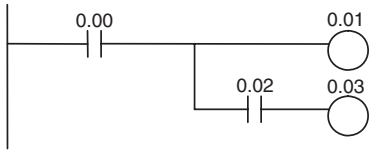


명령	데이터
LD	0.00
OUT	TR 0
AND	0.01
OUT	0.02
LD	TR 0
OUT	0.03

다음과 같이 분기점 이후의 어느 단에도 접점이 없는 경우 및 최종단에만 접점이 있는 경우 일시 기억 릴레이 TR은 필요하지 않습니다.



명령	데이터
LD	0.00
OR	0.01
OUT	0.02



명령	데이터
LD	0.00
OR	0.01
AND	0.02
OUT	0.03

6-13 데이터 메모리

D0~D32767

워드(16비트) 단위로 읽기/쓰기를 실행하는 범용 데이터 영역입니다.
또한 주소에 비트 번호를 부여하면 비트 단위의 지정(읽기/쓰기)도 가능합니다.
전원 OFF 복구(ON→OFF→ON) 또는 동작 모드 변경(프로그램 모드←→운전 또는 모니터 모드) 시에도 해당 전원 OFF 직전 또는 모드 변경 직전의 데이터를 유지합니다.
강제 세트/리셋할 수는 없습니다.

데이터 메모리는 간접 지정으로 지정할 수 있습니다.
지정 방법은 다음 2가지입니다.

- **BIN 모드(@ 표시가 있는 D)**

  지정한 데이터 메모리의 내용을 BIN 값으로 처리하여 그 값을 주소로 한 데이터 메모리를 지정합니다.
  D 앞에 @를 붙입니다.
  BIN 값으로 0Hex~7FFFHex를 저장한 경우 D0~32767의 모든 영역을 지정할 수 있습니다.

  예: @D100 [0100Hex] → (참조) D256 [ ]

- **BCD 모드(* 표시가 있는 D)**

  지정한 데이터 메모리의 내용을 BCD 값으로 처리하여 그 값을 주소로 한 데이터 메모리를 지정합니다. D 앞에 *를 붙입니다.
  BCD 값으로 0~9999를 저장한 경우 D0~09999를 지정할 수 있습니다.

  예: [MOV #1234 *D100]

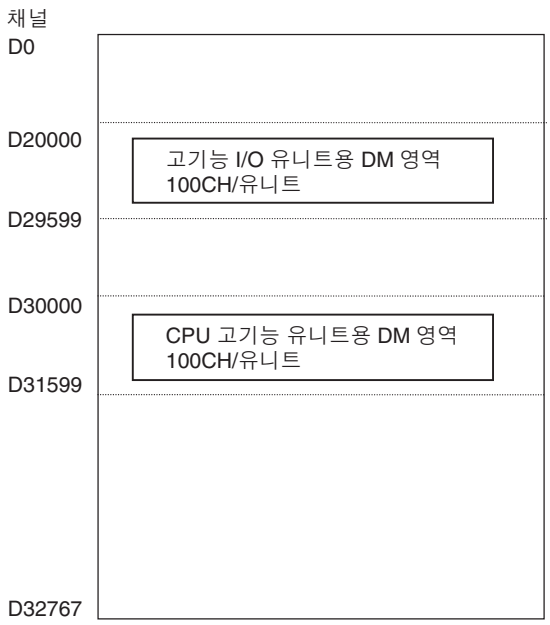
  *D100 [0100Hex] → (참조) D100 [1234Hex]

고기능 유니트용 DM 영역

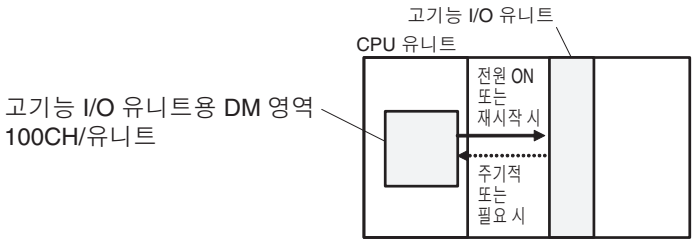
데이터 메모리 영역 중 다음 영역은 고기능 유니트의 초기 설정 데이터 등의 할당에 사용됩니다. 데이터 전송 타이밍은 다음과 같으며 유니트 기종에 따라 다릅니다.

- 전원 ON 시 또는 유니트 재시작 시
- 주기적 데이터 교환
- 필요 시 데이터 교환

전송 타이밍은 각 고기능 유니트의 사용 설명서를 참조하십시오.
고기능 유니트를 사용하지 않는 경우, 범용 데이터 메모리 영역으로 사용할 수 있습니다.

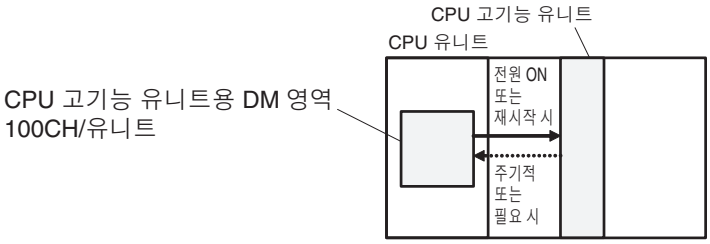


- **D20000~29599: 고기능 I/O 유니트의 시스템 설정 등에 사용 (100CH×96호기 분량)**



이 영역의 사용 방법에 대한 자세한 내용은 각 고기능 I/O 유니트의 사용 설명서를 참조하십시오.

- **D30000~31599: CPU 고기능 유니트의 시스템 설정 등에 사용 (100CH×16유니트 번호 분량)**



이 영역에 대한 자세한 사용 방법은 각 CPU 고기능 유니트의 사용 설명서를 참조하십시오.
또한 CPU 고기능 유니트 중에는 이 DM 영역 이외에 CPU 유니트의 '파라미터 영역' 내의 'CPU 고기능 유니트 시스템 설정 영역'에 초기 설정 데이터를 등록해야 하는 기종이 있습니다(예: Ethernet 유니트). 등록 방법은 주변 툴에 따라 달라집니다.

6-14 확장 데이터 메모리

E0_0~E18_32767

워드(16비트) 단위로 읽기/쓰기를 실행하는 범용 데이터 영역입니다.
또한 주소에 비트 번호를 부여하면 비트 단위의 지정(읽기/쓰기)도 가능합니다.
전원 OFF 복구(ON→OFF→ON) 또는 동작 모드 변경('프로그램' 모드←→'운전' 또는 '모니터' 모드) 시에도 해당 전원 OFF 직전 또는 모드 변경 직전의 데이터를 유지합니다.

확장 데이터 메모리 영역은 범용 데이터 영역 외에, 설정에 따라 '주소 자동 할당 영역', '트레이스 메모리', '파일 메모리'로 사용할 수 있습니다.
확장 데이터 메모리의 뱅크 D~18(E0D_0~E18_32767)은 CJ2 CPU 유니트에서 확대된 영역이므로 CJ2 CPU 유니트에 적합하지 않은 CPU 고기능 유니트, 고기능 I/O 유니트, 표시기, 소프트웨어 등에서는 액세스할 수 없습니다.
CJ2 CPU 유니트에 적합한 CPU 고기능 유니트, 고기능 I/O 유니트는 다음과 같습니다.

- EtherNet/IP 유니트 CJ1W-EIP21
- 위치 제어 유니트 CJ1W-NC214/234/281/414/434/481/881
- 아날로그 입력 유니트 CJ1W-AD042
- 아날로그 출력 유니트 CJ1W-DA042V
- 시리얼 커뮤니케이션 유니트 CJ1W-SCU22/32/42

주소 자동 할당

변수의 주소 할당 영역으로 확장 데이터 메모리를 사용하는 것이 좋습니다.
CJ2H CPU 유니트의 경우 주소 자동 할당 영역으로 확장 데이터 메모리 영역을 사용하면 다음과 같은 영역에서 강제 세트/리셋할 수 있습니다.

형식	주소 자동 할당 시의 강제 세트/리셋 가능 영역
CJ2H-CPU64(-EIP)	E03_0~E03_32767
CJ2H-CPU65(-EIP)	
CJ2H-CPU66(-EIP)	E06_0~E09_32767
CJ2H-CPU67(-EIP)	E07_0~E0E_32767
CJ2H-CPU68(-EIP)	E011_0~E18_32767

주소 자동 할당 방법은 '5-5-8 변수의 주소 자동 할당'을 참조하십시오.

참고

위와 같은 영역 이외의 EM 뱅크에 대해서도 EM 강제 세트/리셋 기능을 사용하면 강제 세트/리셋이 가능해집니다(CJ2H CPU 유니트의 유니트 Ver.1.2 이상).
CJ2M CPU 유니트에서는 주소 자동 할당 영역의 설정에 따르지 않으며 'EM 강제 세트/리셋 기능'을 설정한 경우에만 EM 영역의 강제 세트/리셋이 가능해집니다.

형식	강제 세트/리셋 가능 영역
CJ2M-CPU□2	E0
CJ2M-CPU□2	
CJ2M-CPU□3	
CJ2M-CPU□4	E0~E3
CJ2M-CPU□5	

EM 강제 세트/리셋 기능

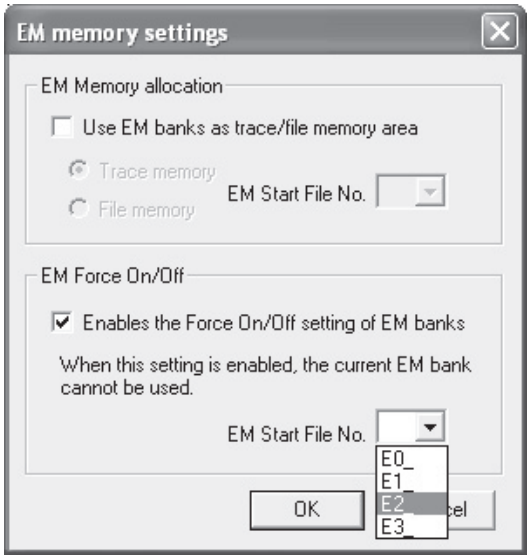
CX-Programmer의 설정에 따라 지정한 EM 뱅크 이후 EM 영역의 비트를 강제 세트/리셋할 수 있습니다 (초기 설정에서는 EM 영역의 강제 세트/리셋 불가능).
주소 자동 할당 영역으로 사용하지 않는 EM 영역에서도 강제 세트/리셋이 가능해집니다.
이 기능의 설정 대상이 되는 EM 뱅크는 전체 EM 뱅크입니다.

● 설정 방법

1. CX-Programmer(*)의 [PLC]-[메모리 할당]-[EM 메모리 설정]을 선택합니다.

[EM 메모리 설정] 대화 상자가 표시됩니다.

2. [EM 영역 강제 설정 가능]을 선택하고 [EM 영역 뱅크]를 설정합니다.



3. 온라인으로 사용자 프로그램을 전송합니다.

* CX-Programmer는 EM 강제 세트/리셋 기능에 적합한 CX-Programmer V8.3 이상을 사용하십시오.

사용상의 주의 사항

- EM 강제 세트/리셋 기능을 설정한 경우 현재 EM 뱅크를 사용할 수 없습니다.
- 트레이스 메모리/파일 메모리로 사용하는 EM 뱅크는 강제 세트/리셋할 수 없습니다.
- 강제 세트/리셋은 모니터 또는 프로그램 모드에서 실행할 수 있습니다. 운전 모드에서는 실행할 수 없습니다.
  강제 세트/리셋 방법은 '10-7-1 강제 세트/리셋'을 참조하십시오.

파일 메모리

CPU 유니트가 처리하는 파일을 저장하는 메모리 영역으로 사용할 수 있습니다.
자세한 내용은 '제7장 파일 조작'을 참조하십시오.

트레이스 메모리

데이터 트레이스 시 샘플링 데이터를 저장하는 영역으로 사용할 수 있습니다.
자세한 내용은 '10-7-6 데이터 트레이스'를 참조하십시오.

EM 영역의 직접 지정

EM 영역은 뱅크라는 여러 개의 영역으로 나뉘어 있으며 지정 방법은 다음 2가지입니다. 기본적으로 아래의 '● 뱅크 번호와 주소를 동시에 지정하는 방법'으로 지정할 것을 권장합니다.

● 뱅크 번호와 주소를 동시에 지정하는 방법

예: E2_10 뱅크 번호 2의 주소 10에 해당하는 확장 데이터 메모리

● 현재 뱅크를 설정한 후 해당 뱅크 내의 주소만 지정하는 방법

예: E10 현재 뱅크의 주소 10에 해당하는 확장 데이터 메모리

- 현재 뱅크는 모드 변경(프로그램←→운전 또는 모니터 모드) 시, 뱅크 0이 됩니다.
- 현재 뱅크를 전환(지정)하기 위해서는 프로그램에서 EMBC(확장 DM 뱅크 전환) 명령을 사용합니다.
- 현재 뱅크 번호가 특수 보조 릴레이 A301CH에 저장됩니다.
  전환한 뱅크 번호를 모드 변경 시에 유지하고 싶을 때는 IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정합니다.
- 현재 뱅크는 태스크를 전환할 때도 변경되지 않습니다.
  따라서 예를 들어 태스크 번호 1에서 현재 EM 뱅크를 2로 설정한 경우 태스크 번호 2에서도 현재 EM 뱅크는 2입니다.
  단, 인터럽트 태스크 내에서 현재 뱅크를 전환하면 사이클릭 태스크로 복구하는 경우 원래의 현재 뱅크로 전환됩니다.
- EM 강제 세트/리셋 기능을 설정하면 현재 EM 뱅크를 사용할 수 없습니다.

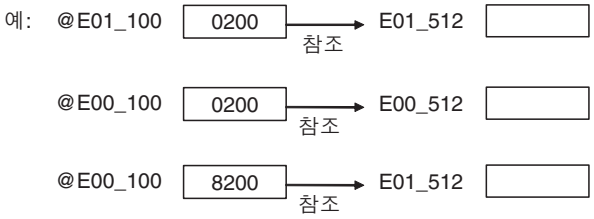
EM 영역의 간접 지정

확장 데이터 메모리는 간접 지정으로 번호를 지정할 수 있습니다.
지정 방법은 다음 2가지입니다.

● BIN 모드(@ 표시가 있는 E)

지정한 확장 데이터 메모리의 내용을 BIN 값으로 처리하여 그 값을 주소로 한 데이터 메모리를 지정합니다. E 앞에 @를 붙입니다.

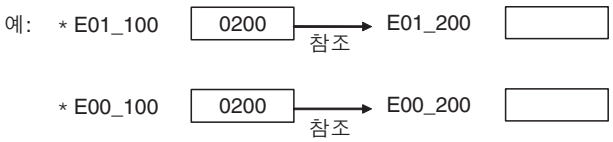
BIN 값으로 0Hex~7FFFHex를 저장한 경우 해당 뱅크 번호에서 0~32767의 모든 영역을 지정합니다. 8000Hex~FFFFHex를 지정한 경우는 다음 뱅크 번호의 0~32767을 지정합니다.



● BCD 모드(* 표시가 있는 E)

지정한 확장 데이터 메모리의 내용을 BCD 값으로 처리하여 그 값을 주소로 한 데이터 메모리를 지정합니다. E 앞에 *를 붙입니다.

BCD 값으로 0~9999를 저장한 경우 해당 뱅크 번호의 0~09999를 지정합니다.



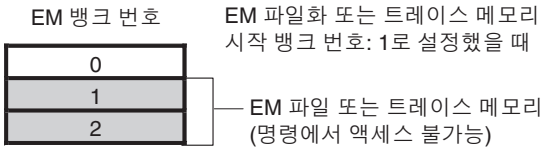
EM 영역의 특수 기능

CX-Programmer를 사용하여 뱅크를 지정하면 해당 뱅크 이후부터 마지막 뱅크까지의 영역을 파일 메모리 또는 트레이스 메모리로 사용할 수 있습니다.

파일 메모리 : 메모리 카드와 같이 파일을 저장할 수 있는 영역.
최대 13뱅크를 사용할 수 있습니다.

트레이스 메모리: 데이터 트레이스 결과를 여기에 저장하면 1회의 트레이스 결과를 증가시킬 수 있습니다.

EM 영역을 파일 메모리 또는 트레이스 메모리로 사용한 경우 명령에서 액세스(읽기/쓰기)할 수 없습니다. 액세스를 시도하면 '잘못된 영역 에러'가 발생합니다.



6-15 타이머

T00~T4095

타이머에는 타임업 플래그와 타이머 현재값의 2종류가 있습니다.
T00~4095까지 4096개의 타이머를 사용할 수 있습니다.

- 타임업 플래그(T)
  접점(1비트) 단위로 읽어오는 영역입니다.
  타이머가 설정 시간을 경과했을 때 ON이 됩니다.
- 타이머 현재값(T)
  워드(16비트) 단위로 읽고 쓰는 영역입니다.
  타이머 계산용으로 카운트 다운/업합니다.

타이머 번호는 100ms 타이머(TIM/TIMX) 명령, 10ms 타이머(TIMH/TIMHX) 명령, 1ms 타이머(TMHH/TMHHX) 명령, 0.1ms 타이머(TIMU/TIMUX) 명령, 0.01ms 타이머(TMUH/TMUHX) 명령, 적산 타이머(TTIM/TTIMX) 명령, 블록 프로그램의 타이머 대기(TIMW/TIMWX) 명령, 고속 타이머 대기(TMHW/TMHWX) 명령에서 공용으로 사용합니다.

명령의 피연산자가 비트(접점) 지정인 경우 타임업 플래그를 가리키고, 채널 지정인 경우 타이머 현재값을 가리킵니다.

사용상의 주의 사항

위 명령에서 같은 번호의 타이머를 동시에 동작시키면 오동작하므로 번호를 중복해서 사용하지 마십시오. 중복 사용하면 프로그램 확인 시, '코일 이중 사용'이 표시됩니다. (단, 동일 사이클에서 같은 번호의 타이머 명령을 동시에 동작시키지 않는 경우에는 같은 번호를 중복해서 사용할 수 있습니다. 하지만 이 경우에도 프로그램 확인 시, '코일 이중 사용'은 표시됩니다.)

타이머는 각 조건에서 다음과 같이 리셋 또는 유지됩니다.

명령(*1)	명령 이름	동작 모드 변경 (프로그램←→운전 또는 모니터 모드) 시 (*2)	전원 OFF 복구 (ON→OFF→ON) 시 (*3)	TRSET (타이머 리셋) 명령	JMP-JME 명령에 의한 점프 시 또는 태스크 대기 중일 때 (*4)	IL-ILC 명령 내에서 IL 조건 OFF 시
TIM/TIMX	100ms 타이머	현재값=0 업 플래그=OFF	현재값=0 업 플래그=OFF	현재값=유지 업 플래그=OFF	기동 중인 타이머는 현재값 갱신	리셋 (현재값=설정값, 업 플래그=OFF)
TIMH/TIMHX	10ms 타이머					
TMHH/TMHHX	1ms 타이머					
TIMU/TIMUX (*1)	0.1ms 타이머					
TMUH/TMUHX (*1)	0.01ms 타이머					
TTIM/TTIMX	적산 타이머				유지	유지
TIMW/TIMWX	타이머 대기				기동 중인 타이머는 현재값 갱신	–
TMHW/TMHWX	고속 타이머 대기					–

*1 TIML/TIMLX(장시간 타이머), MTIM/MTIMX(멀티 출력 타이머)는 타이머 번호가 없으며, 위 조건과는 다른 조건에서 리셋됩니다. 또한 TIML/TIMLX(장시간 타이머) 명령은 설정값으로 리셋되고 MTIM/MTIMX(멀티 출력 타이머) 명령은 0000으로 리셋됩니다. 자세한 내용은 "명령 참조 설명서"(SBCA-351)를 참조하십시오.

*2 IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정한 경우
동작 모드 변경(프로그램←→운전 또는 모니터 모드) 시 또는 운전 정지 이상(FALS 명령 실행 포함)에 의한 운전 정지 시에도 값이 유지됩니다. (전원 OFF 복구 시에는 클리어됩니다.)

*3 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 경우 전원 OFF 복구 시에도 값이 유지됩니다.

*4 TIM/TIMX(100ms 타이머), TIMH/TIMHX(10ms 타이머), TMHH/TMHHX(1ms 타이머), TIMW/TIMWX(타이머 대기), TIHW/TIHWX(고속 타이머 대기)는 타이머 번호가 0~2047인 경우, JMP-JME 명령에 의해 점프된 경우 또는 태스크 대기 중에도 현재값을 갱신합니다. 타이머 번호가 2048~4095이면 점프한 경우 또는 태스크가 대기 중인 경우 현재값을 유지합니다.

- 타임업 플래그는 강제 세트/리셋할 수 있습니다.
- 타이머 번호의 사용 순서나 a 접점, b 접점 사용 횟수에는 제한이 없습니다.
- TIMU/TIMUX(0.1ms 타이머)와 TMUH/TMUHX(0.01ms 타이머)를 제외하고, 타이머의 현재값을 채널 데이터로 읽을 수도 있습니다.

6-16 카운터

C0~C4095

카운터에는 카운트 업 플래그와 카운터 현재값의 2종류가 있습니다.
C0~4095까지 4096개의 카운터를 사용할 수 있습니다.

- 카운트 업 플래그(C)
  접점(1비트) 단위로 읽어오는 영역입니다.
  카운터가 설정값을 경과했을 때 ON이 됩니다.
- 카운터 현재값(C)
  워드(16비트) 단위로 읽고 쓰는 영역입니다.
  카운터 계산용으로 카운트 다운/업합니다.

카운터 번호는 카운터(CNT/CNTX) 명령, 가역 카운터(CNTR/CNTRX) 명령, 블록 프로그램의 카운터 대기(CNTW/CNTWX) 명령에서 공용으로 사용합니다.
명령의 피연산자가 비트(접점) 지정인 경우 카운트 업 플래그를 가리키고, 채널 지정인 경우 카운터 현재값을 가리킵니다.

사용상의 주의 사항

위 명령에서 같은 번호의 카운터를 동시에 동작시키면 오동작하므로 동시에 사용하지 마십시오. 동시에 사용하면 프로그램 확인 시, '코일 이중 사용'이 표시됩니다. (단, 동일 사이클에서 같은 번호의 카운터 명령을 동시에 동작시키지 않는 경우에는 같은 번호를 중복해서 사용할 수 있습니다. 하지만 이 경우에도 프로그램 확인 시, '코일 이중 사용'은 표시됩니다.)

카운터는 다음과 같은 경우에 리셋/유지됩니다.

명령	명령 이름	리셋 시의 현재값/카운트 업 플래그	동작 모드 변경 (프로그램←→운전 또는 모니터 모드) 시	전원 OFF 복구 (ON→OFF→ON) 시	리셋 입력 시	CNR/CNRX (타이머/카운터 리셋) 명령	IL-ILC 명령 내에서 IL 조건 OFF 시
CNT/CNTX	카운터	0000/OFF	유지	유지	리셋	리셋	유지
CNTR/CNTRX	가역 카운터						
CNTW/CNTWX	카운터 대기						

- 카운트 업 플래그는 강제 세트/리셋할 수 있습니다.
- 카운터 번호의 사용 순서나 a 접점, b 접점 사용 횟수에는 제한이 없습니다.
- 카운터 현재값을 채널 데이터로 읽는 용도로 사용할 수도 있습니다.

6-17 태스크 플래그

TK000~TK127

사이클릭 태스크가 실행 가능 상태(RUN)이면 1(ON)이 되고 실행되지 않은 상태(INI) 또는 대기 상태(WAIT)이면 0(OFF)이 되는 플래그입니다.
TK000~127이 사이클릭 태스크 번호 00~127에 대응합니다.
(주) 이 플래그는 사이클릭 태스크에만 적합하며 추가 태스크 및 인터럽트 태스크에는 적합하지 않습니다.

다음과 같은 경우 태스크 플래그가 클리어됩니다.
- 동작 모드 변경(프로그램←→운전 또는 모니터 모드) 시
- 전원 OFF 복구(ON→OFF→ON) 시

IOM(I/O 메모리) 유지 비트(A500.12)의 값에 관계 없이 클리어됩니다.
강제 세트/리셋할 수는 없습니다.

6-18 인덱스 레지스터

IR0~IR15

레지스터 간접 지정을 위해 I/O 메모리 실제 주소를 저장하는 전용 레지스터입니다.
태스크별로 개별 사용하는 경우와 각 태스크에서 공통으로 사용하는 경우가 있습니다.
인덱스 레지스터에 저장되는 I/O 메모리 실제 주소는 사용자가 지정하는 영역 종류(입출력 릴레이, 데이터 메모리 등)별 주소가 아니라 내부적인 RAM의 연속 주소를 말합니다.
사용자는 이 I/O 메모리 실제 주소를 직접 값으로 이 인덱스 레지스터에 입력할 수 없습니다.
인덱스 레지스터 설정(MOVR, 타이머 및 카운터 현재값은 MOVRW) 명령을 통해 값을 입력합니다.

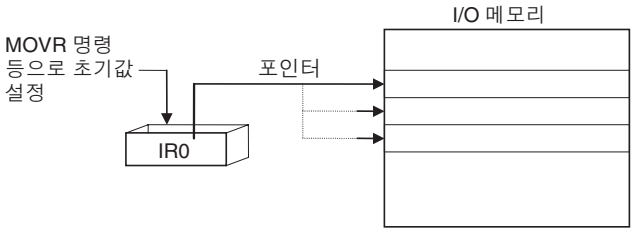
참고

I/O 메모리 실제 주소에 대한 자세한 내용은 '부록-4 I/O 메모리 실제 주소'를 참조하십시오.

인덱스 레지스터 간접 지정이란 이 레지스터 IR0~IR15의 내용을 I/O 메모리 실제 주소로 하는 I/O 메모리를 간접적으로 지정하는 주소 지정 방법입니다.
I/O 메모리에 대한 포인터로, 명령에서 공통으로 사용할 수 있습니다.

- I/O 메모리의 전체 영역을 연속적으로 지정할 수 있습니다(영역 종류별로 지정할 필요는 없습니다). 단, IR, DR, 조건 플래그의 I/O 메모리 실제 주소는 저장할 수 없습니다.
- 일반적인 레지스터 간접 지정에 추가하여 상수 오프셋, DR(데이터 레지스터) 오프셋, 자동 증가, 자동 감소 지정이 가능합니다.
  그 결과, 일정한 주소 간격으로 데이터를 읽고 쓰거나 명령 실행별로 주소를 1 증가 또는 1 감소시키는 것이 간단해집니다.

레지스터 IR0~IR15에는 인덱스 레지스터 설정(MOVR) 명령 등으로 초기값을 설정하고, 이후에는 이 값을 포인터로 삼아 명령 실행별로 해당 포인터를 조작하는 형태로 주소를 지정합니다.
포인터 조작에는 인덱스 레지스터를 직접 지정할 수 있는 특정 명령(인덱스 레지스터 설정 명령, 증가 및 감소 명령, 사칙연산 명령 등)을 사용하거나 간접 지정인 오프셋 지정, 자동 증가/감소 지정을 사용합니다.



인덱스 레지스터를 태스크 간에 독립적으로 사용할지, 공유하여 사용할지 선택할 수 있습니다.
인덱스 레지스터의 값은 다음과 같은 경우 클리어됩니다.

- 동작 모드 변경('프로그램'←→'운전' 또는 '모니터' 모드) 시
- 전원 OFF 복구(ON→OFF→ON) 시
- 운전 정지 이상 발생 시(FALS 명령 실행 시는 제외)

사용상의 주의 사항

인덱스 레지스터를 조작하면 I/O 메모리 영역 이외의 영역을 지정하게 될 가능성이 있습니다. 이런 경우 인덱스 레지스터 간접 지정을 실행하면 '잘못된 영역 액세스 에러'가 발생하므로 주의하십시오. I/O 메모리 영역의 실제 주소에 대해서는 '부록-4 I/O 메모리 실제 주소'를 참조하십시오.

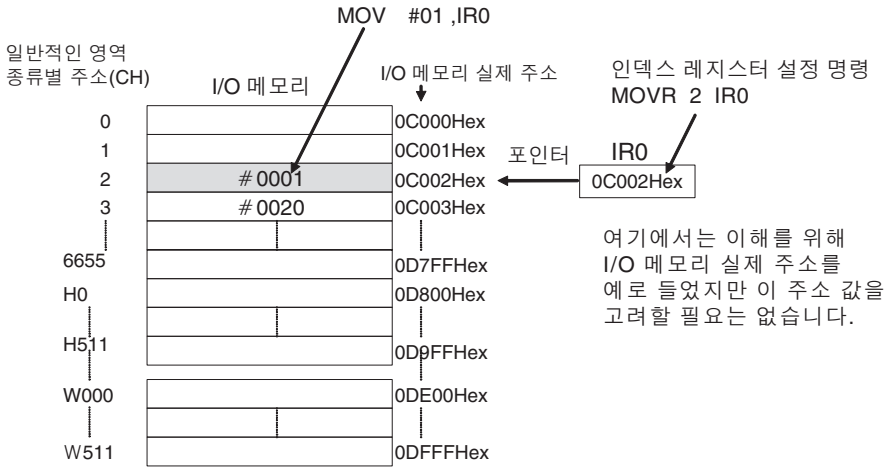
● 레지스터 간접 지정의 종류

종류	내용	지정 방법	예	의미
간접 지정 (오프셋 없음)	IR□의 내용을 I/O 메모리 실제 주소로 하는 접점 또는 채널을 지정합니다.	,IR□	LD , IR0	IR0의 내용을 I/O 메모리 실제 주소로 하는 접점을 로드합니다.
상수 오프셋 지정	IR□의 내용에 지정된 상수를 더하거나 뺀 값을 I/O 메모리 실제 주소로 하는 접점 또는 채널을 지정합니다. 상수 오프셋 지정 범위는 -2048~2047입니다.	상수 ,IR□ 상수에 +, −를 붙임	LD＋5 ,IR0	IR0의 내용+5를 I/O 메모리 실제 주소로 하는 접점을 로드합니다.
DR(데이터 레지스터) 오프셋 지정	IR□의 내용에 DR□의 내용을 더한(+) 값을 I/O 메모리 실제 주소로 하는 접점 또는 채널을 지정합니다.	DR□ ,IR□	LD DR0 ,IR0	IR0의 내용＋DR0의 내용을 I/O 메모리 실제 주소로 하는 접점을 로드합니다.
자동 증가	IR□의 내용을 I/O 메모리 실제 주소로 참조한 후 IR□의 내용을 1 또는 2 증가시킵니다.	1 증가시키는 경우 ,IR□＋ 2 증가시키는 경우 ,IR□＋＋	LD ,IR0＋＋	IR0의 내용을 I/O 메모리 실제 주소로 하는 접점을 로드한 후 IR0의 내용을 2 증가시킵니다.
자동 감소	IR□의 내용을 1 또는 2 감소시키고 그 값을 I/O 메모리 실제 주소로 참조합니다.	1 감소시키는 경우 ,−IR□ 2 감소시키는 경우 ,−−IR□	LD ,−−IR0	IR0의 내용을 2 감소시키고 그 값을 I/O 메모리 실제 주소로 하는 접점을 로드합니다.

(주) □는 0~15를 가리킵니다.

예:

MOVR 2 IR0	2CH의 I/O 메모리 실제 주소를 IR0에 저장합니다.
MOV #0001 ,IR0	IR0의 내용을 I/O 메모리 실제 주소로 하는 채널에 #01을 전송합니다. 그 결과, 2CH에 #0001이 설정됩니다.
MOV #0020 +1 ,IR0	IR0의 내용을 I/O 메모리 실제 주소로 하는 채널+1에 #20을 전송합니다. 그 결과, 3CH에 #0020이 설정됩니다.



명령에는 피연산자를 채널로 취급하는 명령과 접점으로 취급하는 명령이 있습니다. 따라서 인덱스 레지스터의 내용을 I/O 메모리 실제 주소로 사용하는 경우, 명령에 따라 인덱스 레지스터 내용의 의미가 달라집니다.

채널 데이터인 경우

——— [MOVR 0 ,IR2]
——— [MOV D0 ,IR2]

MOV 명령처럼 피연산자가 채널로 취급되는 경우, 인덱스 레지스터의 내용이 그대로 채널 주소로 취급됩니다.
이 예에서는 먼저 MOVR(인덱스 레지스터 설정) 명령을 사용하여 0CH의 I/O 메모리 실제 주소를 IR2에 설정합니다. 그에 따라 IR2의 간접 지정은 0CH를 가리키므로 MOV 명령에 의해 0CH에 D0의 내용이 전송됩니다.

접점 데이터인 경우

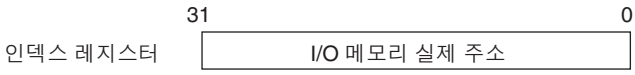
——— [MOVR 0.13, IR2]
——— [SET +5, IR2]

SET 명령처럼 피연산자가 접점으로 취급되는 경우, 인덱스 레지스터의 내용이 비트 주소로 취급됩니다.
이 예에서는 먼저 MOVR(인덱스 레지스터 설정) 명령을 사용하여 0CH의 비트 13의 비트 주소를 IR2에 설정합니다. 그에 따라 SET 명령을 사용하여 0CH의 비트 13에서 5비트를 더한 장소, 즉 1CH의 비트 2를 "1"로 설정합니다.

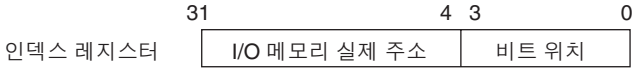


참고

채널을 MOVR 명령으로 설정한 경우 인덱스 레지스터에는 다음과 같이 데이터가 저장됩니다.



비트를 MOVR 명령으로 설정한 경우 인덱스 레지스터에는 다음과 같이 데이터가 저장됩니다.





사용상의 주의 사항

- IR(인덱스 레지스터)은 반드시 값을 설정한 후 사용하십시오. 값이 설정되지 않은 상태에서 사용하면 동작이 정확하지 않게 됩니다.
- 인터럽트 태스크 기동 시, IR(인덱스 레지스터)의 값은 일정하지 않습니다. 인터럽트 태스크 내에서 IR을 사용하는 경우에는 반드시 MOVR(타이머/카운터 현재값 이외의 모두)/MOVRW(타이머/카운터 현재값 설정용) 명령을 사용하여 값을 설정하십시오.

참고

- 테이블 데이터 처리 명령의 데이터 검색(SRCH)/최대값 검색(MAX)/최소값 검색(MIN) 명령은 지정된 테이블 내의 지정된 데이터/최대값/최소값을 검색하고, 그 I/O 메모리 실제 주소를 인덱스 레지스터 IR0에 저장합니다. 따라서 그 후 IR0의 간접 지정을 사용하여 해당 주소의 데이터에서 데이터 읽기 등의 처리를 수행합니다.
- IR(인덱스 레지스터)을 직접 지정할 수 있는 명령은 다음과 같으며, 포인터로 IR(인덱스 레지스터)의 조작에 사용합니다.
  그 외의 명령에서는 직접 지정할 수 없습니다(앞에서 설명한 간접 지정은 가능합니다.).

- 인덱스 레지스터를 직접 지정할 수 있는 명령

명령 종류	명령어	니모닉
데이터 전송	인덱스 레지스터 설정	MOVR
	인덱스 레지스터 설정 (타이머/카운터 현재값용)	MOVRW
	2워드 전송	MOVL
	데이터 2워드 교환	XCGL
테이블 데이터 처리 명령	레코드 위치 설정	SETR
	레코드 위치 읽기	GETR
추적 명령	레코드 검색	RSRCH
	레코드 정렬	RSORT
증가 및 감소 명령	BIN 2워드 증가	++L
	BIN 2워드 감소	--L
데이터 비교 명령	2워드, 일치	=L
	2워드, 불일치	<>L
	2워드, 미만	<L
	2워드, 이하	<=L
	2워드, 초과	>L
	2워드, 이상	>=L
	부호 없는 2워드 비교	CMPL
사칙연산 명령	부호 있음, CY 없는 BIN 2워드 가산	+L
	부호 있음, CY 없는 BIN 2워드 감산	-L

● 인덱스 레지스터의 사용 주의 사항

IR(인덱스 레지스터)은 각 태스크별로 독립되어 있으므로 서로 영향을 주지 않습니다. 예를 들어, 태스크 번호 1에서 사용하는 IR0과 태스크 번호 2에서 사용하는 IR0은 다릅니다. 따라서 IR은 각 태스크별로 16개 있는 것이 됩니다.

단, 다음과 같은 제한이 있습니다.

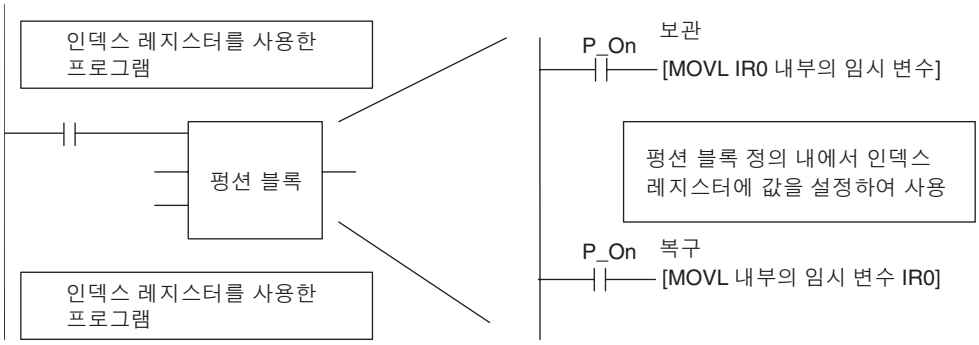
- CX-Programmer에서는 1사이클의 마지막에 실행된 태스크의 IR만 읽을 수 있습니다.
  같은 번호의 IR을 여러 태스크에서 사용한 경우 CX-Programmer에서는 이러한 여러 태스크 중에서 사이클 마지막에 실행된 태스크의 IR 값만 읽을 수 있습니다. (또한 CX-Programmer에서 IR 값을 쓸 수 없습니다.)

● IR(인덱스 레지스터)을 태스크 간에 공유하는 방법

IR(인덱스 레지스터)을 태스크 간에 공유할 수 있습니다.
CX-Programmer를 사용하여 설정합니다.
PLC 속성 대화 상자에서 '태스크별로 개별 IR/DR 사용'의 선택을 취소하십시오.

참고

펑션 블록 내에서 인덱스 레지스터를 사용하는 경우, 펑션 블록을 호출하기 전에 사용 중인 인덱스 레지스터의 값을 펑션 블록 내에서 손상시킬 가능성이 있습니다.
따라서 펑션 블록 내에서 인덱스 레지스터의 사용 전후, 인덱스 레지스터의 보관/복구를 수행할 것을 권장합니다.



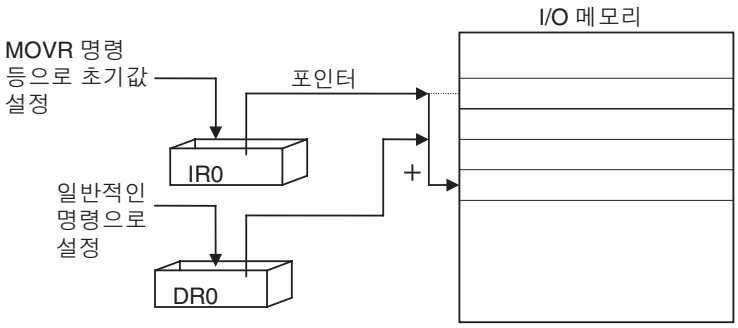
6-19 데이터 레지스터

DR0~DR15

레지스터 간접 지정의 일종으로, 'DR(데이터 레지스터) 오프셋 지정' 시에 사용하는 전용 레지스터입니다.
인덱스 레지스터에 의한 간접 지정의 일종으로, 인덱스 레지스터에 이 데이터 레지스터의 내용을 가산한 값을 지정(오프셋 지정)할 수 있습니다.
태스크별로 개별 사용하는 경우와 각 태스크에서 공통으로 사용하는 경우가 있습니다.

- DR(데이터 레지스터) 오프셋 지정: IR□의 내용에 DR□의 내용을 더한(+) 값을 I/O 메모리 실제 주소로 하는 접점 또는 채널을 지정합니다.
  (주) DR□의 내용이 부호 있는 BIN 값으로 음수인 경우, IR□의 내용에 빼기(-)를 수행하여 이동합니다.

- 강제 세트/리셋은 불가능합니다.
  일반적인 명령으로 DR(데이터 레지스터)에 값을 저장할 수 있습니다.



예 1: LD DR0 ,IR0 — IR0의 내용+DR0의 내용을 I/O 메모리 실제 주소로 하는 접점을 로드합니다.

예 2: MOV #1 DR0 ,IR1 — IR1의 내용+DR0의 내용을 I/O 메모리 실제 주소로 하는 채널에 #1을 전송합니다.

- 데이터 레지스터의 값은 부호 있는 16진수로 취급합니다. 따라서 "－"는 "2의 보수" 표현이 됩니다.

16진수	7FFF	⟷	0, FFFF	⟷	8000
10진수	32767	⟷	0, －1	⟷	－32768

- DR(데이터 레지스터)을 태스크 간에 독립적으로 사용할지, 공유해서 사용할지 선택할 수 있습니다.
- DR(데이터 레지스터)의 값은 CX-Programmer에서 읽고 쓸 수 없습니다.

● 데이터 레지스터 값의 유지/비유지

DR(데이터 레지스터)의 값은 다음과 같은 경우 클리어됩니다.
- 동작 모드 변경('프로그램'←→'운전' 또는 '모니터' 모드) 시
- 전원 OFF 복구(ON→OFF→ON) 시
- 운전 정지 이상 발생 시(FALS 명령 실행 시는 제외)

사용상의 주의 사항

- DR(데이터 레지스터)은 반드시 값을 설정한 후 사용하십시오. 값이 설정되지 않은 상태에서 사용하면 동작이 정확하지 않게 됩니다.
- 인터럽트 태스크 기동 시, DR(데이터 레지스터)의 값은 일정하지 않습니다. 인터럽트 태스크 내에서 DR을 사용하는 경우, 반드시 값을 설정한 후 사용하십시오.

● DR을 태스크 간에 공유하는 방법

DR을 태스크 간에 공유할 수 있습니다.
CX-Programmer를 사용하여 설정합니다.
다음의 PC 속성 대화 상자에서 '태스크별로 개별 IR/DR 사용'의 선택을 취소하십시오.

참고

프로그램 내에서 인덱스 레지스터/데이터 레지스터를 사용하지 않는 경우에는 IR/DR을 태스크 간에 공유하는 것이 좋습니다.

6-20 조건 플래그

에러 플래그 또는 캐리 플래그 등, 각 명령의 실행 결과가 반영되는 전용 플래그 영역 및 상시 ON 또는 OFF 플래그입니다.
(기존에는 이 플래그가 특수 보조 릴레이에 포함되어 있었습니다.)

P_CY, P_ER 등의 변수명(글로벌 변수)으로 지정합니다.
명령 및 CX-Programmer에서 쓸 수 없습니다.

태스크를 전환하면 모두 클리어됩니다. 따라서 에러 플래그 또는 액세스 에러 플래그의 경우, 사이클 내에서 해당 에러가 발생한 태스크(프로그램) 내에서만 그 상태가 유지됩니다.
강제 세트/리셋은 불가능합니다.

조건 플래그의 개요는 다음과 같습니다.

명칭		의미
에러 플래그	P_ER	각 명령의 피연산자 데이터가 잘못된 경우(명령 처리 에러 발생 시) ON이 됩니다. 명령의 잘못된 종료를 나타냅니다. (*1)
액세스 에러 플래그	P_AER	'잘못된 영역 액세스 에러'가 발생했을 때 ON이 됩니다. 잘못된 영역 액세스 에러는 원래 명령으로 액세스하지 않아야 할 영역에 액세스한 경우를 가리킵니다. (*2)
캐리 플래그	P_CY	연산 결과, 자리올림 또는 자리내림이 있는 경우, 비트가 이동한 경우 등에서 ON이 됩니다. (*3)
> 플래그	P_GT	두 데이터의 비교 결과가 ">"인 경우, 어떤 데이터가 지정된 범위를 상회한 경우 등에 ON이 됩니다.
= 플래그	P_EQ	두 데이터의 비교 결과가 "="인 경우, 연산 결과가 0이 된 경우 등에 ON이 됩니다.
< 플래그	P_LT	두 데이터의 비교 결과가 "<"인 경우, 어떤 데이터가 지정 범위를 하회한 경우 등에 ON이 됩니다.
부정 플래그	P_N	연산 결과의 최상위 비트가 1이 된 경우 등에 ON이 됩니다.
오버플로 플래그	P_OF	연산 결과가 오버플로된 경우 ON이 됩니다.
언더플로 플래그	P_UF	연산 결과가 언더플로된 경우 ON이 됩니다.
플래그	P_GE	두 데이터의 비교 결과가 " "인 경우 ON이 됩니다.
≒ 플래그	P_NE	두 데이터의 비교 결과가 "≒"인 경우 ON이 됩니다.
플래그	P_LE	두 데이터의 비교 결과가 " "인 경우 ON이 됩니다.
상시 ON 플래그	P_On	항상 ON 상태인 플래그입니다.
상시 OFF 플래그	P_Off	항상 OFF 상태인 플래그입니다.

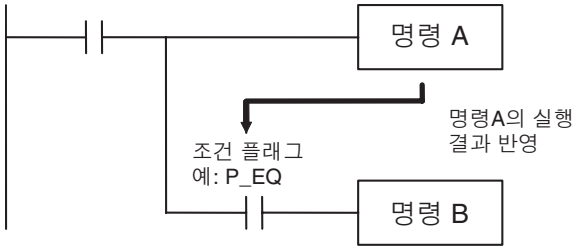
*1 PLC 시스템 설정에서 '명령 에러 발생 시 동작 설정'을 '정지'로 설정하면 이 에러(ER) 플래그가 ON일 때 운전을 정지하며 동시에 명령 처리 에러 플래그(A295.08)가 ON이 됩니다.

*2 PLC 시스템 설정에서 '명령 에러 발생 시 동작 설정'을 '정지'로 설정하면 이 액세스 에러(AER) 플래그가 ON일 때 운전을 정지하며 동시에 잘못된 영역 액세스 에러 플래그(A4295.10)가 ON이 됩니다.

*3 데이터 이동 명령, 사칙연산(CY가 있는 가산 및 감산) 명령에서는 연산 대상의 일종이 됩니다.

● 사용 방법

조건 플래그는 명령에서 공통으로 사용되기 때문에 각 명령 실행별로 해당 실행 결과에 따라 1사이클 내에서 변경됩니다.
따라서 한 명령의 실행 결과를 반영할 때는 해당 명령 바로 뒤에 같은 입력 조건의 출력 분기에서 사용하십시오.
예: 명령 A의 실행 결과를 수신한 경우



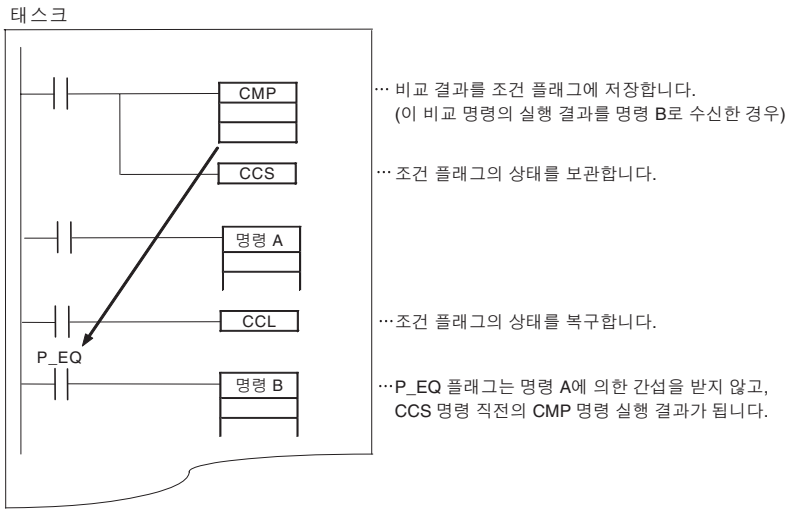
명령	데이터
LD	—
명령 A	—
AND	P_EQ
명령 B	—

사용상의 주의 사항

- 조건 플래그는 각 명령에서 공통으로 사용되므로 한 프로그램(태스크) 내에서 간섭이 발생하고, 그로 인해 프로그램이 원래 동작과 다르게 동작하는 경우가 있습니다. 따라서 조건 플래그 사용 방법에 주의하십시오.
- 조건 플래그는 태스크를 전환하면 클리어됩니다. 예를 들어 태스크 번호 1의 마지막 조건 플래그 상태를 태스크 번호 2에서 읽을 수 없으므로 주의하십시오.

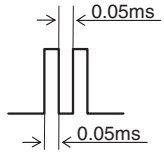
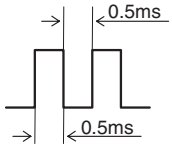
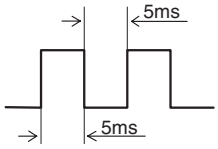
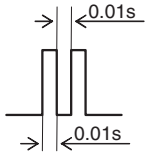
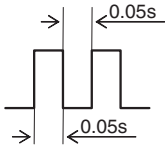
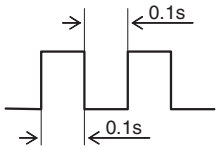
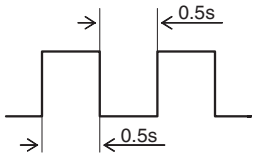
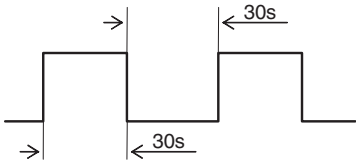
참고

CCS(조건 플래그 저장) 명령 및 CCL(조건 플래그 로드) 명령을 사용하면 조건 플래그를 태스크(프로그램) 내의 다른 위치, 태스크 간 또는 이후 사이클에서 보관/복구시킬 수 있습니다.
예) 태스크 내 다른 위치에 보관/복구하는 경우



6-21 클록 펄스

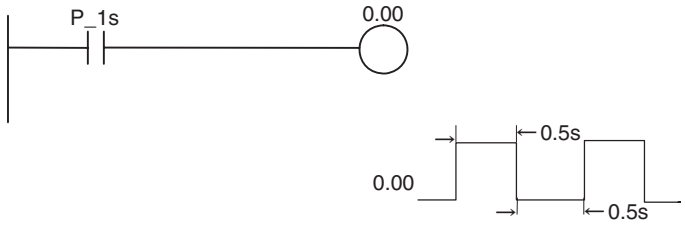
일정 시간별로 시스템이 ON/OFF하여 클록 펄스를 발생시키는 전용 플래그 영역입니다.
P_0_1ms, P_1ms 등의 변수명(글로벌 변수)으로 지정합니다.

명칭		내용	
0.1밀리초 클록 펄스	P_0_1ms		0.05밀리초 ON, 0.05밀리초 OFF의 반복 펄스
1밀리초 클록 펄스	P_1ms		0.5밀리초 ON, 0.5밀리초 OFF의 반복 펄스
0.01초 클록 펄스	P_0_01s		5밀리초 ON, 5밀리초 OFF의 반복 펄스
0.02초 클록 펄스	P_0_02s		0.01초 ON, 0.01초 OFF의 반복 펄스
0.1초 클록 펄스	P_0_1s		0.05초 ON, 0.05초 OFF의 반복 펄스
0.2초 클록 펄스	P_0_2s		0.1초 ON, 0.1초 OFF의 반복 펄스
1초 클록 펄스	P_1s		0.5초 ON, 0.5초 OFF의 반복 펄스
1분 클록 펄스	P_1min		30초 ON, 30초 OFF의 반복 펄스

클록 펄스는 읽기 전용이며, 피연산자에 지정하여 쓸 수 없습니다.
이 값은 운전을 시작할 때 클리어됩니다.

● 사용 방법

0.5초 간격으로 ON/OFF 데이터 출력



명령	데이터
LD	P_1s
OUT	0.00

● 클록 펄스의 갱신

클록 펄스는 프로그램 실행 중에도 갱신됩니다.
따라서 프로그램의 선두와 마지막에 ON/OFF가 다른 경우가 있습니다.

● 오차

클록 펄스는 최대 0.01%(25℃)의 오차가 있습니다.
장시간 시간을 제어하는 경우에는 클록 펄스보다 내부 시계를 기준으로 제어하는 것이 좋습니다. 단, 이 경우에도 내부 시계 자체의 오차는 발생합니다.

7

파일 조작

이 장에서는 CPU 유니트에 저장된 파일의 종류와 저장 위치, 파일 조작에 대해 설명합니다.

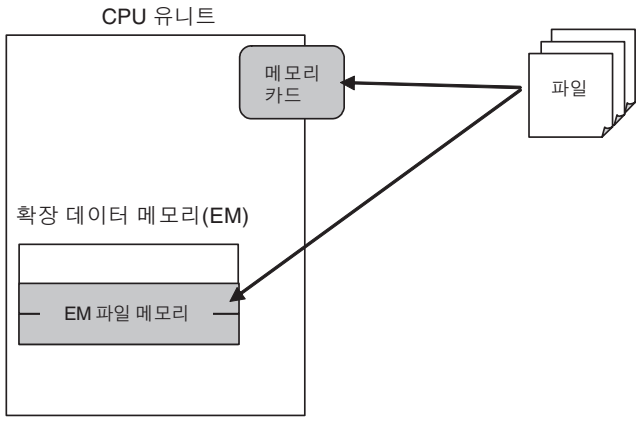
7-1 파일 메모리란? 7-2
7-1-1 파일 메모리의 매체 7-2
7-1-2 매체의 초기화 7-3
7-1-3 메모리 카드를 사용하는 경우 주의 사항 7-5
7-2 파일 메모리용 파일 종류 7-8
7-2-1 파일 종류 7-8
7-2-2 파일 작성 및 매체에 저장하는 순서 7-11
7-3 파일 메모리 조작 방법 7-12
7-3-1 파일 메모리 조작의 종류 7-12
7-3-2 파일 메모리 조작 및 파일 7-14
7-3-3 파일의 사용 제한 7-21
7-3-4 파일 크기 7-21
7-3-5 주변 툴과 파일 메모리의 관계 7-22

7-1 파일 메모리란?

7-1-1 파일 메모리의 매체

파일 메모리는 파일 저장용 메모리를 말합니다. 파일 메모리에는 다음의 2종류가 있습니다.

- 메모리 카드
- EM 파일 메모리(EM 영역 지정 범위)



CJ 시리즈에서는 메모리 카드 및 EM 영역의 지정 범위를 파일 저장용 메모리로 사용할 수 있습니다. 어느 경우든 사용자 프로그램 전체, I/O 메모리, 파라미터 영역을 파일로 저장할 수 있습니다.

파일 메모리의 종류	메모리 유형	메모리 용량	저장할 수 있는 파일
메모리 카드	플래시 메모리	HMC-EF183 128MB HMC-EF283 256MB HMC-EF583 512MB	• 프로그램 • 코멘트 파일 • 프로그램 인덱스 • 변수 테이블 • 파라미터 파일 • 유니트 백업 파일 (메모리 카드만 해당)
내장 RAM (EM 파일 메모리)	RAM	EM 영역의 최대 용량 (I/O 메모리에서 EM 영역의 지정 뱅크 이후 마지막 뱅크까지)	

7-1-2 매체의 초기화

메모리 카드의 경우

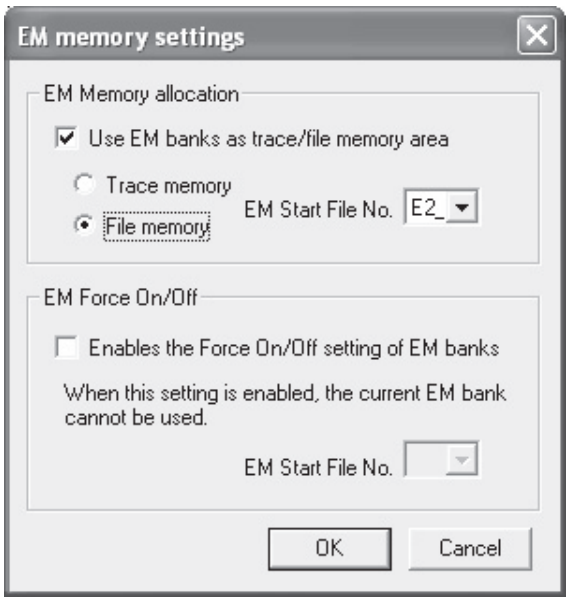
메모리 카드는 CX-Programmer의 [메모리 카드] 창에서 초기화(포맷)합니다.
단, 메모리 카드 HMC-EF□□□는 공장 출하 시에 이미 초기화되어 있으므로 처음 사용하는 경우 초기화할 필요가 없습니다.

EM 파일 메모리의 경우

EM 파일 메모리는 반드시 다음과 같은 조작을 통해 파일 메모리로 하는 시작 뱅크 번호를 지정하여 초기화하십시오.

1 **CX-Programmer의 [PLC] – [메모리 할당] – [EM 메모리 설정]을 선택합니다.**
[EM 메모리 설정] 대화 상자가 표시됩니다.

2 **[파일 메모리]를 선택한 후 [EM 시작 파일 번호]를 선택합니다.**



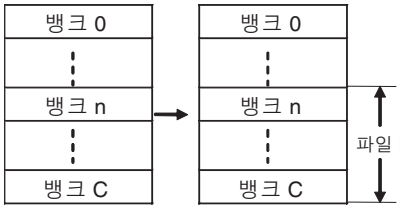
참고

- EM 영역을 파일 메모리와 트레이스 메모리로 동시에 사용할 수 없습니다.
  두 기능 중 하나를 선택하십시오.
- 트레이스 메모리/파일 메모리로 사용하는 EM 뱅크는 EM 강제 세트/리셋을 설정할 수 없습니다.

3 **온라인으로 사용자 프로그램을 전송합니다.**
EM 파일 메모리를 사용할 수 있게 됩니다.

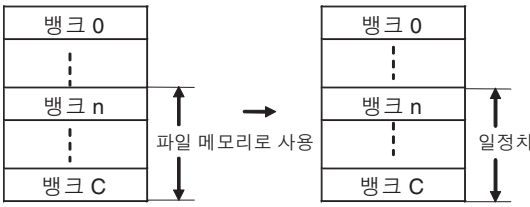
● EM 파일 초기화의 각종 사례

평상시의 EM 상태에서 지정된 EM 뱅크 이후를 파일 메모리로 사용하는 경우



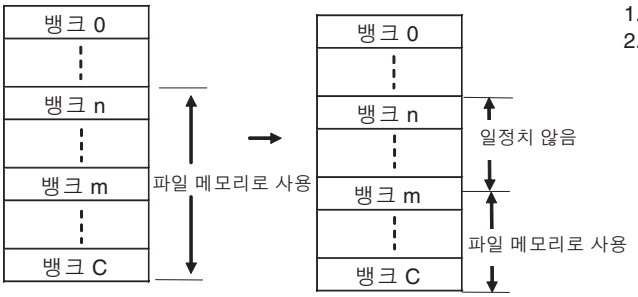
1. EM 메모리 설정에서 n 지정
2. CX-Programmer에서 프로그램 전송

파일 메모리로 설정된 EM을 평상시의 EM 상태로 되돌리는 경우



1. EM 메모리 설정에서 파일 메모리 없음 지정
2. CX-Programmer에서 프로그램 전송. 이때, 뱅크 n 이후의 EM 영역은 일정치 않습니다.

파일 메모리의 시작 뱅크 번호를 변경하는 경우



1. EM 메모리 설정에서 n을 m으로 변경
2. CX-Programmer에서 프로그램 전송. 이때, 뱅크 n~m의 EM 영역은 일정치 않습니다.

7-1-3 메모리 카드를 사용하는 경우 주의 사항

메모리 카드를 사용하기 전에 다음 사항을 확인하십시오.

초기화

메모리 카드는 초기화하여 출하되므로 구입한 후 초기화할 필요가 없습니다.
사용한 후 초기화하는 경우는 반드시 CX-Programmer를 사용하여 CPU 유니트에서 초기화하십시오.
노트북 컴퓨터 등으로 메모리 카드를 직접 초기화한 경우, CPU 유니트가 메모리 카드를 인식하지 못할 수 있습니다. 이런 경우에는 다시 CPU 유니트를 사용하여 메모리 카드를 초기화해도 사용할 수 없으므로 주의하십시오.

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.

쓰기 횟수

일반적으로 플래시 메모리에는 쓰기 횟수의 제한이 있습니다.
보장할 수 있는 메모리 카드의 쓰기 횟수 제한은 10만 회입니다.
가령 10분 간격으로 끊임없이 쓰기를 실시하면 2년 이내에 10만 회에 도달한다는 계산이 됩니다.

1개 파일의 최소 크기

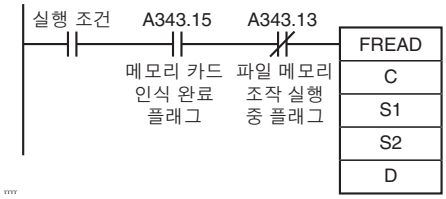
DM 영역의 수 CH 분량의 데이터와 같이 작은 파일을 여러 개 저장하면 카드 용량의 상한까지 저장할 수 없습니다.
예를 들어, 할당 단위 크기가 4096바이트인 메모리 카드인 경우 내부 구조 관계로 한 파일의 데이터량이 아무리 작아도 4096바이트 크기를 사용하게 됩니다.
따라서 DM 영역의 10CH 분량의 데이터를 메모리 카드에 저장하면 파일 크기는 68바이트이지만 할당 단위 크기가 4096바이트인 메모리 카드에서는 68바이트의 파일이 실질적으로 4096바이트 용량을 사용합니다.
이와 같이 한 파일의 크기가 작은 경우는 카드의 사용 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.
할당 단위 크기에 대한 자세한 내용은 일반적인 PC 관련 서적을 참조하십시오.

메모리 카드의 인식 시간

메모리 카드를 장착하고 CPU 유니트가 인식을 완료할 때까지 몇 초에서 몇 분까지 걸리는 경우가 있습니다. 이러한 시간은 PLC의 사이클 시간이나 메모리 카드의 용량, 저장된 파일 수 등에 따라 달라집니다.
예를 들어, 사이클 시간이 0.4ms이고 PLC 시스템 설정이 모두 초기값인 경우 메모리 카드(HMC-EF183)의 인식 시간은 8초(기준)가 됩니다.

메모리 카드에 대한 액세스

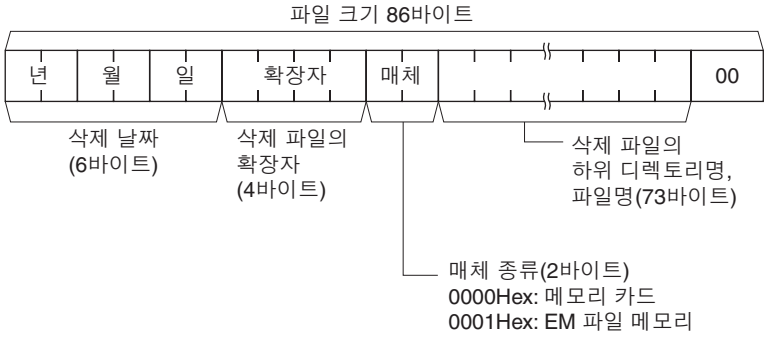
- BUSY LED(메모리 카드 액세스 중 LED)가 점등 중일 때는 절대로 PLC 본체의 전원을 끄지 마십시오. 메모리 카드를 사용할 수 없게 되는 경우가 있습니다.
- BUSY LED가 점등되어 있는 상태에서 그대로 메모리 카드를 빼지 마십시오.
  메모리 카드를 사용할 수 없게 되는 경우가 있습니다.
  메모리 카드를 뺄 때는 메모리 카드 전원 공급 정지 버튼을 누르고 BUSY LED가 꺼졌는지 확인한 후 빼십시오.
- 메모리 카드를 장착한 후 CPU 유니트가 인식을 완료할 때까지는 몇 초 정도 걸립니다. 따라서 전원 ON 시 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.



파일 메모리 액세스 중에 전원 OFF가 발생한 경우의 동작

파일 메모리(메모리 카드 또는 EM 파일 메모리)에 액세스하는 중에 전원 OFF(전원 OFF)가 발생한 경우, 메모리 카드의 내용을 보장할 수 없으며 갱신 도중의 파일이 제대로 기입되지 않거나 경우에 따라 카드 자체가 손상될 수 있습니다.
이 경우, 해당 파일은 다음 번에 전원을 켰을 때 시스템에서 자동으로 삭제합니다.
삭제와 동시에 CPU 유니트의 특수 보조 릴레이의 파일 삭제 통지 플래그(메모리 카드의 파일인 경우: A395.06, EM 파일 메모리의 파일인 경우: A395.07)가 자동으로 1(ON)이 됩니다. (파일 삭제 통지 플래그는 다음 번 전원 OFF 시 0(OFF)이 됩니다.)
또한 동시에 삭제 로그 파일(파일명: DELETE.TMP)이 메모리 카드 또는 EM 파일 메모리의 루트 디렉토리(￥)에 자동으로 작성됩니다.
바이너리 편집기를 사용하여 삭제 로그 파일(파일명: DELETE.TMP)을 검토하면 다음과 같은 정보를 확인할 수 있습니다.
삭제된 파일의 삭제 날짜, 존재했던 파일 메모리(매체) 종류, 하위 디렉토리, 파일명, 확장자의 필요에 따라 삭제된 파일을 다시 작성하십시오.

- 삭제 로그 파일(￥DELETE.TMP)



7-2 파일 메모리용 파일 종류

파일 메모리에 저장할 수 있는 파일 종류는 다음과 같습니다.

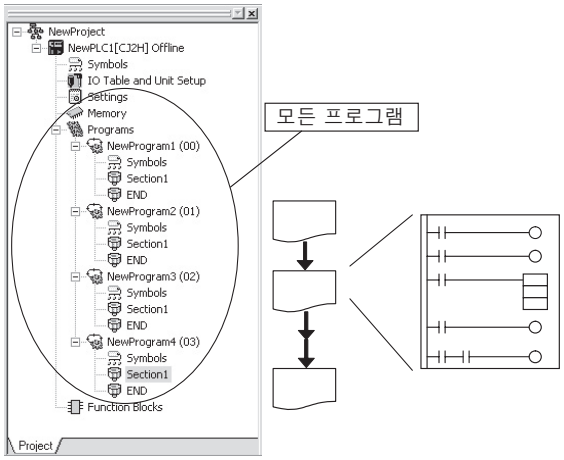
- 프로그램/네트워크 변수 파일
- 파라미터 파일
- 데이터 파일
- 변수 테이블(네트워크 변수 제외)
- 코멘트 파일
- 프로그램 인덱스 파일
- 유니트 백업 파일

7-2-1 파일 종류

프로그램/네트워크 변수 파일

프로그램/네트워크 변수 파일은 CPU 유니트의 사용자 프로그램(사이클릭 태스크 및 인터럽트 태스크 내의 프로그램) 및 네트워크 변수(글로벌 변수 테이블 내의 네트워크 변수) (*)를 기록한 파일입니다. 또한 각 프로그램의 속성 정보도 포함됩니다.

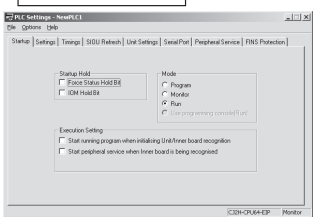
*CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당



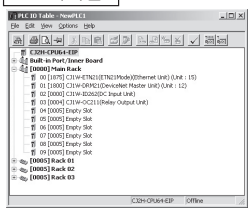
파라미터 파일

파라미터 파일은 CPU 유니트 내 파라미터 영역의 데이터를 기록한 파일입니다. 파라미터 영역에는 PLC 이름, PLC 시스템 설정, I/O 테이블, CPU 고기능 유니트 시스템 설정(데이터 링크 파라미터 포함), 라우팅 테이블과 같은 데이터가 포함됩니다.

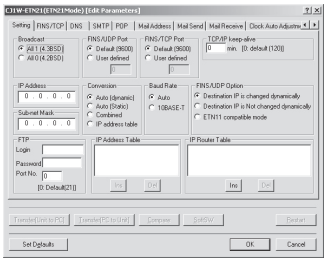
PLC 시스템 설정



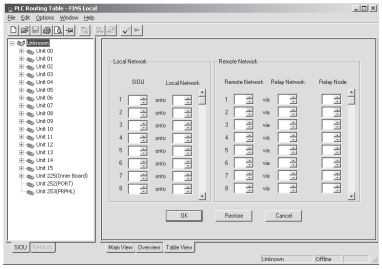
I/O 테이블



CPU 고기능 유니트 시스템 설정

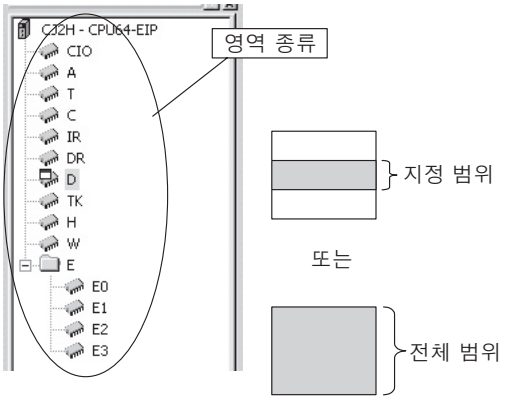


라우팅 테이블



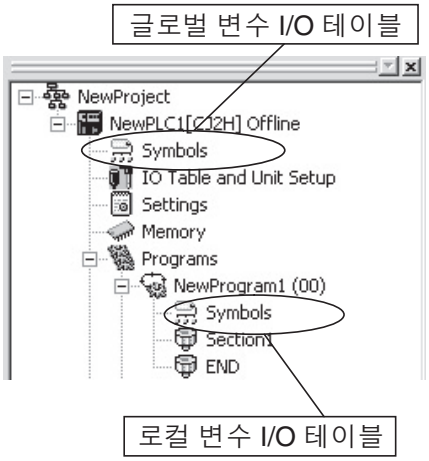
데이터 파일

데이터 파일은 I/O 메모리의 1영역 종류별 데이터를 채널(16비트) 단위로 기록한 파일입니다. 1영역 종류 내의 지정 범위 또는 모든 범위의 데이터를 기록할 수 있습니다. 영역 종류에는 CIO, HR(유지 릴레이), WR(내부 보조 릴레이), AR(특수 보조 릴레이), DM(데이터 메모리), EM(확장 데이터 메모리)의 6개 영역이 있습니다.



변수 테이블 파일

변수 테이블 파일(네트워크 변수 제외)은 CX-Programmer에서 사용하는 변수 테이블의 정보, PLC 주소 자동 할당 데이터를 기록한 파일입니다.



변수 주소 데이터 형식 I/O 코멘트

Name	Data Type	Address / Value	Net. Variable	Rack Location	Usage	Comment
P_CY	BOOL	CF004			Work	Carry (CY) Flag
P_Cycle_Time_Error	BOOL	A401.08			Work	Cycle Time Error Flag
P_Cycle_Time_Value	UDINT	A264			Work	Present Scan Time
P_DM	WORD	A460			Work	DM Area Parameter
P_EM0	WORD	A461			Work	EM0 Area Parameter
P_EM1	WORD	A462			Work	EM1 Area Parameter
P_EM2	WORD	A463			Work	EM2 Area Parameter
P_EM3	WORD	A464			Work	EM3 Area Parameter
P_EM4	WORD	A465			Work	EM4 Area Parameter
P_EM5	WORD	A466			Work	EM5 Area Parameter
P_EM6	WORD	A467			Work	EM6 Area Parameter
P_EM7	WORD	A468			Work	EM7 Area Parameter
P_EM8	WORD	A469			Work	EM8 Area Parameter
P_EM9	WORD	A470			Work	EM9 Area Parameter
P_EMA	WORD	A471			Work	EMA Area Parameter
P_EMB	WORD	A472			Work	EMB Area Parameter
P_EMC	WORD	A473			Work	EMC Area Parameter
P_EQ	BOOL	CF006			Work	Equals (EQ) Flag
P_ER	BOOL	CF003			Work	Instruction Execution Er

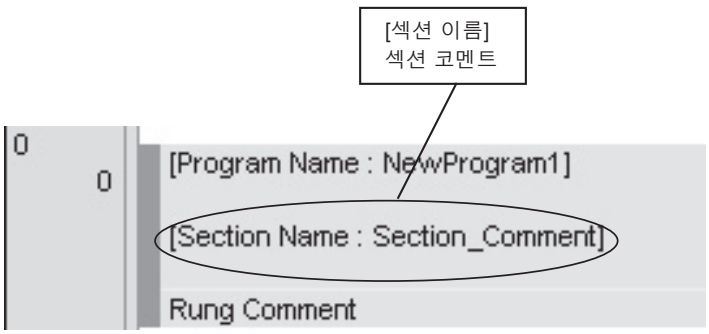
코멘트 파일

코멘트 파일은 CX-Programmer에서 사용하는 코멘트용 정보를 기록한 파일입니다.



프로그램 인덱스 파일

프로그램 인덱스 파일은 CX-Programmer에서 사용하는 섹션용 정보를 기록한 파일입니다.



유니트 백업 파일

유니트 백업 파일은 간이 백업 기능에 대응한 유니트의 내부 데이터를 기록한 파일이며, 간이 백업 조작을 수행한 경우에 작성됩니다. 내부 데이터의 내용은 개별 유니트에 따라 달라집니다.
예:

- DeviceNet 유니트: 장치 파라미터
- 시리얼 커뮤니케이션 유니트: 프로토콜 매크로 데이터

7-2-2 파일 작성 및 매체에 저장하는 순서

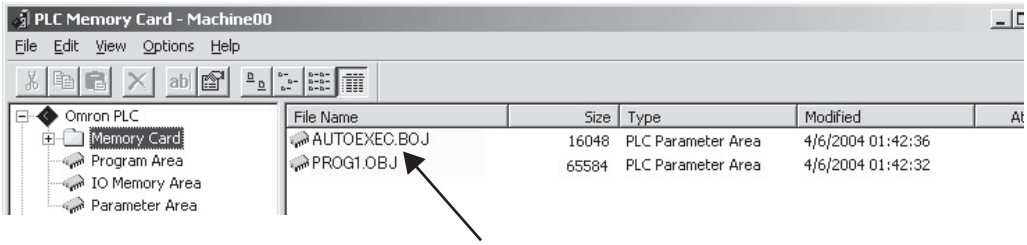
사용자 프로그램, I/O 메모리, 파라미터 영역 데이터의 경우

CX-Programmer의 [PLC 메모리 카드] 창에서 전송 기능을 사용합니다.

1 **CPU 유니트에 이미 초기화된 메모리 카드를 장착하거나 EM 파일 메모리를 초기화합니다.**
EM 파일 메모리의 초기화에 대해서는 '7-1-2 매체의 초기화'를 참조하십시오.

2 **CX-Programmer를 온라인으로 연결합니다.**

3 **[PLC 메모리 카드] 창에서 전송 대상(메모리 카드/EM 파일 메모리)을 선택하고, [파일] 메뉴에서 [전송]을 선택합니다.**
CPU 유니트 내의 데이터(사용자 프로그램, I/O 메모리, 파라미터 영역)가 파일 메모리용 파일로 메모리 카드 또는 EM 파일 메모리에 저장됩니다.



전원 ON 시 자동 전송되는 프로그램 파일

변수 테이블, 코멘트, 프로그램 인덱스의 경우

CX-Programmer의 전송 기능을 사용합니다.

1 **CPU 유니트에 이미 초기화된 메모리 카드를 장착하거나 EM 파일 메모리를 초기화합니다.**
EM 파일 메모리의 초기화에 대해서는 '7-1-2 매체의 초기화'를 참조하십시오.

2 **CX-Programmer를 온라인으로 연결합니다.**

3 **[PLC] 메뉴에서 [전송] – [전송[PC→PLC]] 또는 [전송[PLC→PC]]를 선택합니다.**

4 **전송할 데이터로 [심볼] 또는 [코멘트]를 선택합니다.**

- CPU 유니트에 메모리 카드가 장착된 경우 메모리 카드로만 전송할 수 있습니다(EM 파일 메모리는 불가능).

7-3 파일 메모리 조작 방법

7-3-1 파일 메모리 조작의 종류

파일 메모리를 사용하는 조작 방법은 다음과 같습니다.

- CX-Programmer에서 조작
- 전원 ON 시 자동 전송
- 간이 백업 조작
- FWRIT/FREAD 명령
- 특수 보조 릴레이를 통해 운전 중에 전체 사용자 프로그램 교체
- FINS 명령

CX-Programmer에서 조작

● CX-Programmer의 메모리 카드 창에서 전송 조작

다음과 같은 파일의 작성, 읽기/쓰기 작업을 수행할 수 있습니다.

- 전원 ON 시 자동 전송 파일(프로그램/네트워크 변수 파일, 데이터 파일, 파라미터 파일)
- 범용 파일(프로그램/네트워크 변수 파일, 데이터 파일, 파라미터 파일)
- CX-Programmer 전용 파일(변수 테이블, 코멘트 파일, 프로그램 인덱스 파일)

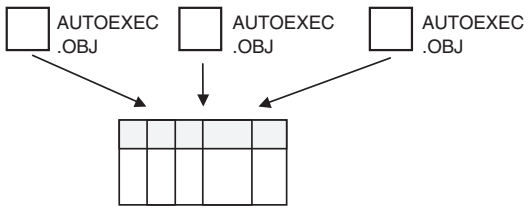
● CX-Programmer에서 온라인/오프라인 전송 조작

범용 파일(프로그램/네트워크 변수 파일, 데이터 파일, 파라미터 파일)을 온라인으로 실제 기기 PLC에 또는 오프라인으로 PC 기억 장치에 전송할 수 있습니다.

자세한 내용은 "CX-Programmer 조작 설명서"(SBCA-337)를 참조하십시오.

전원 ON 시 자동 전송

작업 전환용 프로그램, 파라미터, I/O 메모리를 메모리 카드에 저장해 두고 전원을 켤 때 메모리 카드에서 읽을 수 있습니다.



자세한 내용은 "제10장 CPU 내장 기능"을 참조하십시오.

간이 백업 조작

PLC의 모든 데이터(CPU 유니트 내의 모든 데이터 및 각 고기능 유니트 내의 데이터)를 주변 툴 없이 메모리 카드에 백업/복원할 수 있습니다.

자세한 내용은 "CJ 시리즈 CJ2 CPU 유니트 사용 설명서 하드웨어편"(SBCA-349)을 참조하십시오.

백업 복원 조작

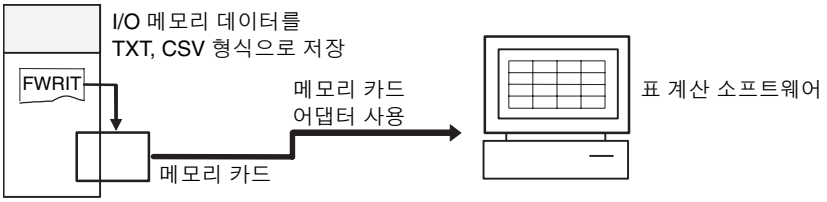
PLC 전체 데이터(CPU 유니트 내의 전체 데이터 및 각 고기능 유니트 내의 데이터)를 PLC 백업 툴을 사용하여 PC에 백업/복원할 수 있습니다.

자세한 내용은 "CJ 시리즈 CJ2 CPU 유니트 사용 설명서 하드웨어편"(SBCA-349)을 참조하십시오.

FWRIT/FREAD 명령

사용자 프로그램의 FWRIT 명령을 통해, I/O 메모리 데이터를 메모리 카드 또는 EM 파일 메모리에 TXT 또는 CSV 형식으로 저장할 수 있습니다.

그리고 메모리 카드 어댑터를 통해 데이터를 PC의 표 계산 소프트웨어로 읽어서 편집할 수 있습니다.

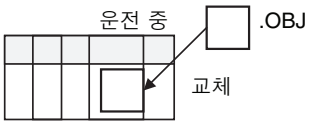


또한 이와 반대로 표 계산 소프트웨어에서 작성한 설정 데이터 등을 TXT 또는 CSV 형식으로 메모리 카드에 저장한 후 FREAD 명령을 사용하여 CPU 유니트로 읽을 수도 있습니다.

자세한 내용은 "명령 참조 설명서"(SBCA-351)를 참조하십시오.

특수 보조 릴레이를 통해 운전 중에 사용자 프로그램 교체

운전 중에 래더 프로그램에서(주변 툴 없이) 사용자 프로그램을 교체할 수 있습니다.



자세한 내용은 "제10장 CPU 내장 기능"을 참조하십시오.

FINS 명령

파일 메모리를 조작하는 FINS 명령을 사용하여 각종 파일 메모리를 조작할 수 있습니다.

FINS 명령은 상위 링크 연결된 상위 컴퓨터에서, 네트워크에 있는 다른 PLC의 CMND 명령에서 또는 자체 사용자 프로그램의 CMND 명령에서 전송합니다.

자세한 내용은 "통신 명령 참조 설명서"(SBCA-304)를 참조하십시오.

7-3-2 파일 메모리 조작 및 파일

파일 메모리용 조작 종류별로 사용할 수 있는 파일 메모리에 대해서는 다음에 정리되어 있습니다.

읽기: 파일 메모리→CPU 유니트로 파일 전송
쓰기: CPU 유니트→파일 메모리로 파일 전송

○: 조작 가능, −: 조작 불가능

조작 방법	매체	내용	파일 종류				
			프로그램/ 네트워크 변수 파일	데이터 파일	파라미터 파일	변수 테이블, 코멘트 파일, 프로그램 인덱스 파일	유니트 백업 파일
CX-Programmer	메모리 카드, EM 파일 메모리, PC 기억 장치	작성	○(범용 파일), ○(전원 ON 시 자동 전송용 파일)			○ (CX-Programmer 전용 파일)	—
		읽기					
		쓰기					
		기타 조작					
전원 ON 시 자동 전송	메모리 카드만 해당	읽기	○(전원 ON 시 자동 전송용 파일)			—	
		쓰기	—				
간이 백업 조작	메모리 카드만 해당	읽기	○(간이 백업용 파일)				
		쓰기					
PLC 백업 툴 조작	PC 기억 장치만 해당	읽기	○(PLC 백업 툴 전용 파일)				
		쓰기					
FREAD/FWRIT 명령	메모리 카드, EM 파일 메모리	한 파일 내의 데이터 읽기	—	○(범용 파일)	—	—	
		한 파일 내의 데이터 쓰기					
특수 보조 릴레이를 통해 운전 중에 전체 사용자 프로그램 교체	메모리 카드만 해당	읽기	○(범용 파일)	—			
FINS 명령	메모리 카드, EM 파일 메모리	읽기	○(범용 파일)			—	
		쓰기					
		기타 조작					

전원 ON 시 자동 전송용 파일

전원을 켰을 때 사용자 프로그램/네트워크 변수, 파라미터 데이터, I/O 메모리를 일괄적으로 메모리 카드에서 CPU 유니트로 읽어오기 위한 파일입니다.
파일명은 AUTOEXEC 또는 REPLACE로 정해져 있습니다.
자세한 내용은 '10-3-5 전원 ON 시 자동 전송 기능'을 참조하십시오.

간이 백업용 파일

CPU 유니트의 전면 딥 스위치 및 메모리 카드 전원 공급 정지 버튼 스위치를 사용한 간이 백업 조작을 통해 메모리 카드에 저장되는 PLC 전체의 백업 파일입니다. 파일명은 BACKUP□□로 정해져 있습니다.
자세한 내용은 "CJ 시리즈 CJ2 CPU 유니트 사용 설명서 하드웨어편"의 '8-3 간이 백업'을 참조하십시오.

PLC 백업 툴 전용 파일

PLC 백업 툴의 백업 목록 조작에 의해 작성되는 PLC 전체의 백업 파일입니다.

자세한 내용은 "CJ 시리즈 CJ2 CPU 유니트 사용 설명서 하드웨어편"의 '8-2 PC 조작 백업'을 참조하십시오.

CX-Programmer 전용 파일

CX-Programmer에서 작성하는 파일입니다. 파일 메모리에 작성되는 파일명은 정해져 있습니다. 프로젝트 전송 조작 시, 이러한 전용 파일의 전송 대상을 다음 중에서 선택할 수 있습니다.

- 코멘트 메모리(CPU 유니트 내장 파일 메모리 내)
- 메모리 카드
- EM 파일 메모리

파일 종류	파일명	내용	설명
변수 테이블 파일 (네트워크 변수 제외)	SYMBOLS.SYM	글로벌 변수 테이블 (네트워크 변수 제외) 및 로컬 변수 테이블	• CX-Programmer의 글로벌/로컬 변수 테이블 내의 변수, 주소, 데이터 형식, I/O 코멘트 정보가 포함됩니다. 단, 네트워크 변수는 포함되지 않습니다. • CX-Programmer의 [PC 주소 자동 할당]에서 설정한 데이터도 포함됩니다.
코멘트 파일	COMMENTS.CMT	행 코멘트, 주석문(메모)	• CX-Programmer의 행 코멘트, 주석문 정보
프로그램 인덱스 파일	PROGRAM.IDX	섹션명, 섹션 코멘트	• CX-Programmer의 섹션 구분 정보 (단, 구분 위치는 프로그램 내의 섹션 구분 명령에 따라 다름)

(주) CX-Programmer에서는 오프라인에서 위의 변수 테이블 파일(네트워크 변수 제외) 및 코멘트 파일을 CX-Programmer←→PC의 RAM←→기억 장치 간에서 전송할 수 있습니다.

CX-Programmer에서 CPU 유니트로 프로젝트 전송 시, 변수 테이블, 코멘트 파일, 프로그램 인덱스 파일을 CPU 유니트 내장 플래시 메모리 내의 코멘트 메모리에 저장할 수 있습니다. 또한 간이 백업 조작 시, 코멘트 메모리에 저장된 변수 테이블, 코멘트 파일, 프로그램 인덱스 파일이 각각 백업 파일로 자동 작성되어 메모리 카드에 저장됩니다.

범용 파일

CX-Programmer, FINS 명령, 명령 또는 특수 보조 릴레이를 통해 조작(읽기/쓰기 등) 가능한 파일입니다. 파일명은 사용자가 임의로 정의할 수 있습니다.

파일 종류	파일명(*1)	확장자	내용	설명	
프로그램/네트워크 변수 파일	＊＊＊＊＊＊＊＊	.OBJ	• 사용자 프로그램 • 네트워크 변수(*2)	CPU 유니트의 사이클릭 태스크와 인터럽트 태스크 내의 프로그램 및 네트워크 변수입니다.	
파라미터 파일	＊＊＊＊＊＊＊＊	.STD	• PLC 시스템 설정 • 등록 I/O 테이블 • 라우팅 테이블 • CPU 고기능 유니트 시스템 설정 영역 등	• CPU 유니트 내부의 파라미터 영역의 데이터입니다. • 파일 내의 각 정보 종류별로 데이터를 식별할 필요는 없습니다.	
데이터 파일	＊＊＊＊＊＊＊＊	.IOM	I/O 메모리의 지정 범위	• 1개 영역 종류 내의 시작 CH~최종 CH까지의 CH(16비트) 단위 데이터입니다. • 영역 종류는 CIO, HR(유지 릴레이), WR(내부 보조 릴레이), AR(특수 보조 릴레이), DM(데이터 메모리), EM(확장 데이터 메모리) 중 하나를 사용할 수 있습니다.	BIN 데이터 형식
		.TXT			TXT 형식(구분 없음 또는 탭으로 구분) (*3)
		.CSV			CSV 형식(쉼표로 구분)

*1 ＊＊＊＊＊＊＊＊은 8자 이내의 문자(ASCII 코드)로 설정합니다.

*2 CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당

*3 데이터 파일 중에 TXT 형식, CSV 형식의 파일은 FWRIT/FREAD 명령을 통해서만 읽고 쓸 수 있습니다. CX-Programmer에서는 읽고 쓸 수 없습니다.

● 범용 데이터 파일

데이터 파일의 종류

범용 데이터 파일의 확장자에는 .IOM, .TXT, .CSV의 3종류가 있습니다.

확장자	내용	용도
.IOM	BIN 형식 CS/CJ 시리즈의 독자적인 데이터 형식	I/O 메모리의 백업
.TXT(*) .CSV	I/O 메모리의 1CH 또는 2CH가 필드(항목)로서 ASCII 코드로 변환된 데이터 형식 레코드 사이를 행바꿈 코드로 구분 가능	표 계산 소프트웨어와 데이터 교환

* TXT, CSV 형식의 데이터 파일을 읽고 쓰는 방법
FWRIT/FREAD 명령을 통해서만 읽고 쓸 수 있습니다.

TXT, CSV의 데이터 형식에는 다음의 6종류가 있습니다.

확장자	데이터 형식	내용	
		1필드(구분 요소)당 I/O 메모리 크기	구분 코드
.TXT(*)	구분 없는 워드	1CH	없음
	구분 없는 LONG 워드	2CH	없음
	탭 구분 워드	1CH	탭 코드
	탭 구분 LONG 워드	2CH	탭 코드
.CSV(*)	쉼표 구분 워드	1CH	쉼표
	쉼표 구분 LONG 워드	2CH	쉼표

* TXT, CSV 형식의 문자 코드 제한
데이터 파일에 반각 16진수 값(0~9, A~F, a~f) 이외의 문자가 포함된 경우, I/O 메모리로 정확하게 읽어올 수 없습니다.

- 필드 크기의 제한
  데이터 파일 내 필드 크기는 워드 형식인 경우, 16진수 값으로 반각 4자리(4바이트)이어야 합니다.
  LONG 워드 형식의 경우, 16진수 값으로 반각 8자리(8바이트)이어야 합니다. 그 밖의 경우, I/O 메모리로 정확하게 읽어올 수 없습니다.

- 저장 순서
  워드 형식의 경우, I/O 메모리의 데이터를 I/O 메모리상의 하위 CH→상위 CH의 순서로 1CH마다(필드로) 구분하여 ASCII 코드로 변환해서 저장합니다.
  LONG 워드 형식의 경우, I/O 메모리의 하위 CH→상위 CH 순서로 2CH마다(2CH 이내는 상위 CH→하위 CH 순서) (필드로) 구분하여 ASCII 코드로 변환해서 저장합니다.

- 구분 코드
  구분하지 않는 경우 필드 사이를 채워서 연속으로 저장합니다.
  쉼표로 구분하면 필드 사이에 쉼표 코드를 삽입해서 저장합니다.
  탭으로 구분하면 필드 사이에 탭 코드를 삽입해서 저장합니다.
  단, 읽을 때는 FREAD 명령에서 구분 있음(쉼표 또는 탭)으로 지정한 경우, 1바이트의 코드(쉼표/탭으로 한정되지 않음)로 구분된 데이터로 읽어옵니다.

- 행바꿈 지정
  행바꿈이 있는 경우, 지정된 필드 수에 따라 행바꿈 코드를 삽입해서 저장합니다.
  또한 행바꿈이 있는 파일은 FWRIT/FREAD 명령에서 읽기 시작 위치/쓰기 시작 위치(파일 메모리 선두에서의 오프셋 위치)를 지정할 수 없습니다.
  행바꿈이 없는 경우에는 연속으로 저장합니다.

- 저장 크기
  1파일당 저장 크기는 FWRIT 명령의 피연산자를 사용하여 지정하며, 크기는 필드 수로 지정합니다. 워드 형식인 경우 I/O 메모리의 1CH＝1필드에 해당합니다.
  LONG 워드 형식인 경우 I/O 메모리의 2CH＝1필드에 해당합니다.

데이터 파일의 파일명

파일 내에는 저장되어 있는 영역 종류와 데이터에 대해 알려주는 정보가 없습니다. 따라서 파일명에 다음과 같이 식별용 문자(시작 주소 등)를 넣는 방법 등을 통해 사용자가 관리하십시오.

예: D00100.IOM, CIO0020.TXT
전송 원본 데이터 파일(.IOM/.TXT/.CSV) 내의 영역 종류와 전송 대상 CPU 유니트의 영역 종류가 달라도 CX-Programmer에서 파일→CPU 유니트로 전송을 수행하면, 특별히 식별하지 않고 파일 선두부터 CH 데이터를 I/O 메모리의 지정된 주소 이후로 읽어옵니다. (예: CX-Programmer에서 파일 내의 CIO 데이터를 DM 영역에 읽어오도록 조작하면 그대로 CPU 유니트의 DM 영역으로 읽어옵니다.)



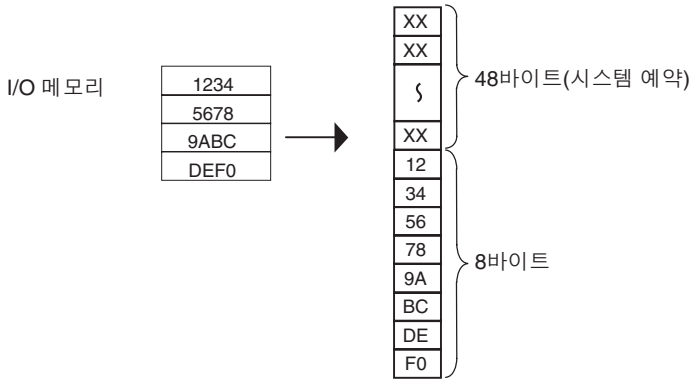
참고

- TXT, CSV 형식의 데이터 파일은 I/O 메모리에 있는 16진수 값(0~9, A~F)을 표 계산 소프트웨어와 교환하기 위한 데이터입니다.
- 래더 언어의 TWRIT(텍스트 파일 쓰기) 명령 또는 ST 언어의 메모리 카드 처리 함수 WRITE_TEXT( )를 사용하여 I/O 메모리의 ASCII 코드 데이터를 텍스트 파일로 저장할 수 있습니다.

참고

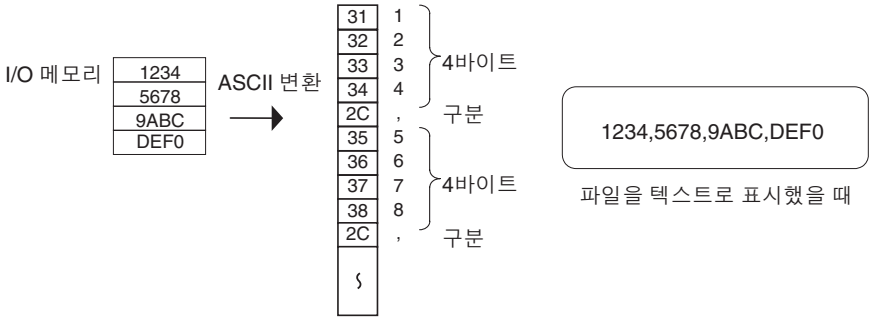
데이터 파일 내의 데이터 구조는 다음과 같습니다.

- BIN 형식(.IOM)의 경우
  예: I/O 메모리의 4채널 데이터가 1234Hex, 5678Hex, 9ABCHex, DEF0Hex를 ABC.IOM이라는 파일에 저장한 경우
  단, 평상시 조작에서는 이 데이터 형식을 고려할 필요가 없습니다.



ABC.IOM 파일의 내용

- 워드 CSV 형식(.CSV), 워드 TXT 형식(.TXT)의 경우
  예: I/O 메모리의 4채널 데이터가 1234Hex, 5678Hex, 9ABCHex, DEF0Hex를 ABC.CSV라는 파일에 저장한 경우



ABC.CSV 파일의 내용

- LONG 워드 CSV 형식(.CSV), LONG 워드 TXT 형식(.TXT)의 경우
  예: I/O 메모리의 4채널 데이터가 1234Hex, 5678Hex, 9ABCHex, DEF0Hex를 ABC.CSV라는 파일에 저장한 경우



ABC.CSV 파일의 내용

참고

표 계산 소프트웨어를 사용하여 데이터 파일(TXT, CSV 형식)을 작성하는 방법

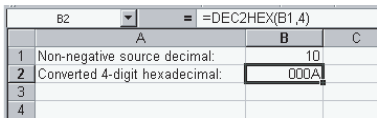
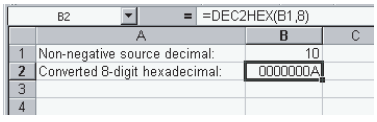
표 계산 소프트웨어(예: Microsoft Excel)를 사용하여 CSV, TXT 형식의 데이터 파일을 작성하는 방법은 다음과 같습니다.

- 셀의 표시 형식
  문자열
- 셀 내 문자 수
  워드 형식인 경우 반드시 반각 4문자로 입력하십시오. LONG 워드 형식의 경우에는 반드시 반각 8문자로 입력하십시오.
  예) 워드 형식인 경우 000A를 A로만 입력하지 마십시오.
- 셀 내의 문자 종류
  반드시 16진수 값(0~9, A~F, a~f)의 문자를 사용하십시오. 가타카나, 영문자, 히라가나, 한자, 기호는 사용할 수 없습니다

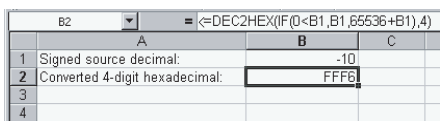
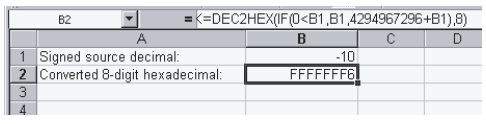
- I/O 메모리에 16진수 숫자를 저장하려는 경우 표 계산 소프트웨어에서 10진수 입력을 16진수로 변환하면 편리합니다. 방법은 다음과 같습니다.
  (1) [도구] 메뉴에서 [추가 기능 등록 및 관리]를 선택합니다.
  (2) [추가 기능 등록 및 관리] 대화 상자에서 [분석 도구]를 선택합니다.
  (3) 함수를 사용할 셀 위치에서 [삽입] 메뉴의 [펑션]을 선택합니다.
  (4) [범주] 필드의 [엔지니어링] 내에서 DEC2HEX(숫자,자릿수)를 선택합니다.
  (5) 16진수 4자리로 변환하는 경우, 숫자 위치에 IF(0<= 참조 셀 위치,참조 셀 위치,65535+참조 셀 위치)를 입력합니다.
  16진수 8자리로 변환하는 경우에는 숫자 위치에 IF(0<= 참조 셀 위치,참조 셀 위치,4294967296+참조 셀 위치)의 조건을 입력합니다.

예

・0 또는 양의 10진수 값을 입력하는 경우

	16진수 4자리로 변환하는 경우	16진수 8자리로 변환하는 경우
사용 함수	DEC2HEX(참조 셀 위치,4)	DEC2HEX(참조 셀 위치,8)
예	10진수로 10을 입력하고 16진수 4자리수인 000AHex로 변환 	10진수로 10을 입력하고, 16진수 8자리수인 0000000AHex로 변환 

・부호 있는 10진수 값을 입력하는 경우

	16진수 4자리로 변환하는 경우	16진수 8자리로 변환하는 경우
사용 함수	DEC2HEX(IF(0<=참조 셀 위치,참조 셀 위치,65536+참조 셀 위치),4)	DEC2HEX(IF(0<=참조 셀 위치,참조 셀 위치 ,4294967296+참조 셀 위치),8)
예	10진수로 -10을 입력하고 16진수 4자리수인 FFF6Hex로 변환 	10진수로 -10을 입력하고, 16진수 8자리수인 FFFFFFF6Hex로 변환 

7-3-3 파일의 사용 제한

파일 메모리로 사용하는 파일 형식은 MS-DOS 형식입니다.
따라서 Windows PC에서 일반적인 파일처럼 처리할 수 있습니다.

파일명

파일에 파일명과 확장자를 붙여서 식별합니다.
파일명에는 다음과 같은 문자를 사용할 수 있습니다.
a~z, A~Z의 영문자, 0~9의 숫자, !, &, $, #, ', {, }, -, ^, (,), _
파일명에는 다음과 같은 문자(영어 및 수학 기호)를 사용할 수 없습니다.
,, ., /, \, ?, *, ", :, ;, <, >, =, +, 공백, 한자 등의 전각 문자
확장자는 다음의 '파일 분류' 표와 같이 파일 종류에 따라 사전에 정해져 있습니다.

디렉토리

CS/CJ 시리즈는 하위 디렉토리에 있는 파일에 액세스할 수 있습니다. 파일 메모리 내의 파일 저장 위치에 디렉토리를 지정하고, 디렉토리를 최대(루트 디렉토리 포함) 5계층까지 작성할 수 있습니다.
또한 전체 디렉토리 길이는 최대 65문자까지입니다.
Windows 등에서 메모리 카드의 하위 디렉토리를 작성하는 경우 하위 디렉토리의 깊이에 주의하십시오.

7-3-4 파일 크기

파일 크기(바이트)는 다음과 같이 계산할 수 있습니다.

파일 종류	크기
데이터 파일(.IOM)	(CH 수×2+48)바이트 예) DM 전체 범위(D0~32767)의 경우, 32768×2+48=65584바이트
데이터 파일(.TXT 또는 .CSV)	구분 형식, 행바꿈 위치에 따라 파일 크기가 달라집니다. (*1) 예 1) 구분 없는 워드, 행바꿈이 없는 경우 12345689ABCDEF012345678일 때, 24바이트 예 2) 구분 있는 워드, 2필드마다 행바꿈하는 경우 1234,5678 ↵ 9ABC,DEF0 ↵ 1234,5678 ↵일 때, 33바이트 예 3) 구분 있는 LONG 워드, 2필드마다 행바꿈하는 경우 56781234,DEF01234 ↵ 56781234 ↵일 때, 29바이트
프로그램/네트워크 변수 파일(.OBJ)	(사용 스텝 수(*2)×4+48)바이트
파라미터 파일(.STD)	16048바이트

*1 구분 코드는 1바이트이고 행바꿈 코드는 2바이트입니다.

*2 프로그램/네트워크 변수 파일의 사용 스텝 수는 CX-Programmer의 상호 참조 보고서 화면에 표시되는 'UM 합계 스텝 수'에서 'UM 빈 스텝 수'를 빼서 구합니다(상호 참조 보고서 화면은 [보기]−[상호 참조 보고서]−[보고서 작성]을 수행하여 표시합니다).

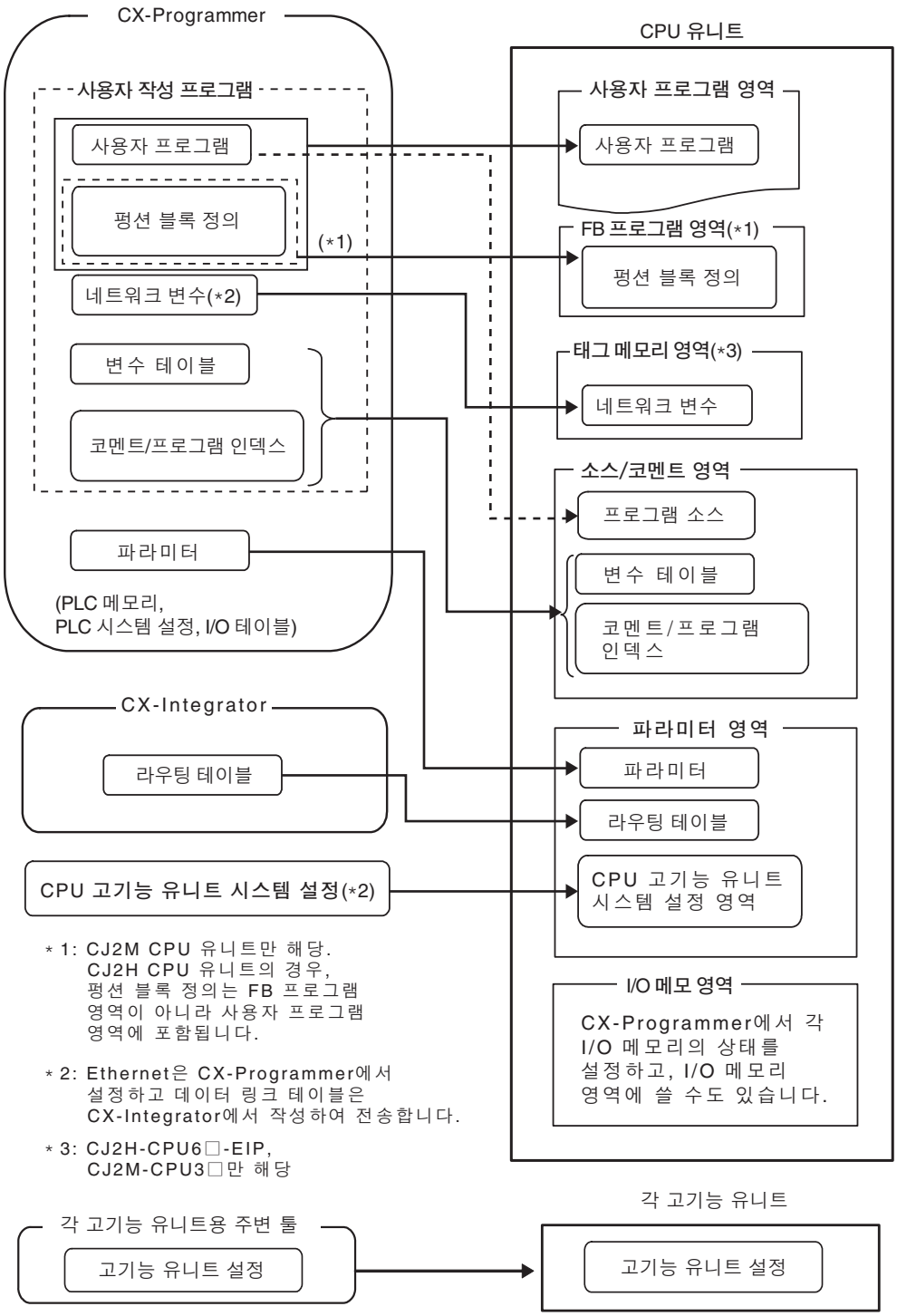
7-3-5 주변 툴과 파일 메모리의 관계

각 주변 툴의 파일과 파일 메모리 파일의 관계는 다음과 같습니다.

주변 툴	주변 툴 파일	데이터 내용	파일 메모리의 파일	저장 영역
CX-Programmer	CX-Programmer의 프로젝트 파일(확장자: .cxp)	사용자 프로그램/네트워크 변수	프로그램/네트워크 변수 파일 (확장자: .OBJ)	CPU 유니트
		펑션 블록		
		변수 테이블 (변수명－주소 대응, I/O 코멘트 포함) (*1)	변수 테이블 파일 (네트워크 변수 제외) (SYMBOLS.SYM)	
		코멘트 (I/O 코멘트 이외의 행 코멘트, 주석문)	코멘트 파일 (COMMENTS.CMT)	
		PLC 이름	파라미터 파일 (확장자: .STD) (*1)	
		PLC 시스템 설정		
		등록 I/O 테이블		
		Ethernet 유니트 설정		
	데이터 링크 테이블 파일(.cl2)	Controller Link 데이터 링크 테이블		
	CX-Programmer의 프로젝트 파일(확장자: .cxp), 유니트 설정(XML 파일)＝CPS의 내용 데이터	PLC 메모리(I/O 메모리) (할당 릴레이, 할당 DM은 아래에 해당)	데이터 파일 (확장자: IOM)	
		할당 릴레이, 할당 DM		
CX-Integrator	라우팅 테이블 파일 (확장자: .rtg)	라우팅 테이블	파라미터 파일 (확장자: .STD) (*2)	
FL-net 설정 툴	특정 파일(.csv)	FL-net 설정		
고기능 유니트전용 주변 툴	각 고기능 유니트 전용 주변 툴에서 작성한 파일	위 데이터 이외	유니트 백업 파일 (BACKUP□□.PRM)	고기능 유니트

*1 네트워크 변수는 제외됩니다.

*2 개별 주변 툴로 작성한 파일이지만 파일 메모리에서는 하나의 파일로 작성됩니다.


CX-Programmer
사용자 작성 프로그램
사용자 프로그램
펑션 블록 정의
(*1)
네트워크 변수(*2)
변수 테이블
코멘트/프로그램 인덱스
파라미터
(PLC 메모리,
PLC 시스템 설정, I/O 테이블)
CX-Integrator
라우팅 테이블
CPU 고기능 유니트 시스템 설정(*2)
* 1: CJ2M CPU 유니트만 해당.
CJ2H CPU 유니트의 경우,
펑션 블록 정의는 FB 프로그램
영역이 아니라 사용자 프로그램
영역에 포함됩니다.
* 2: Ethernet은 CX-Programmer에서
설정하고 데이터 링크 테이블은
CX-Integrator에서 작성하여 전송합니다.
* 3: CJ2H-CPU6□-EIP,
CJ2M-CPU3□만 해당
각 고기능 유니트용 주변 툴
고기능 유니트 설정
CPU 유니트
사용자 프로그램 영역
사용자 프로그램
FB 프로그램 영역(*1)
펑션 블록 정의
태그 메모리 영역(*3)
네트워크 변수
소스/코멘트 영역
프로그램 소스
변수 테이블
코멘트/프로그램
인덱스
파라미터 영역
파라미터
라우팅 테이블
CPU 고기능 유니트
시스템 설정 영역
I/O 메모 영역
CX-Programmer에서 각
I/O 메모리의 상태를
설정하고, I/O 메모리
영역에 쓸 수도 있습니다.
각 고기능 유니트
고기능 유니트 설정

8

I/O 할당 및 유니트 설정

이 장에서는 CPU 유니트 및 유니트와 데이터를 교환하는 I/O 할당에 대한 자세한 내용과 더불어 각 유니트별 설정에 대해 설명합니다.

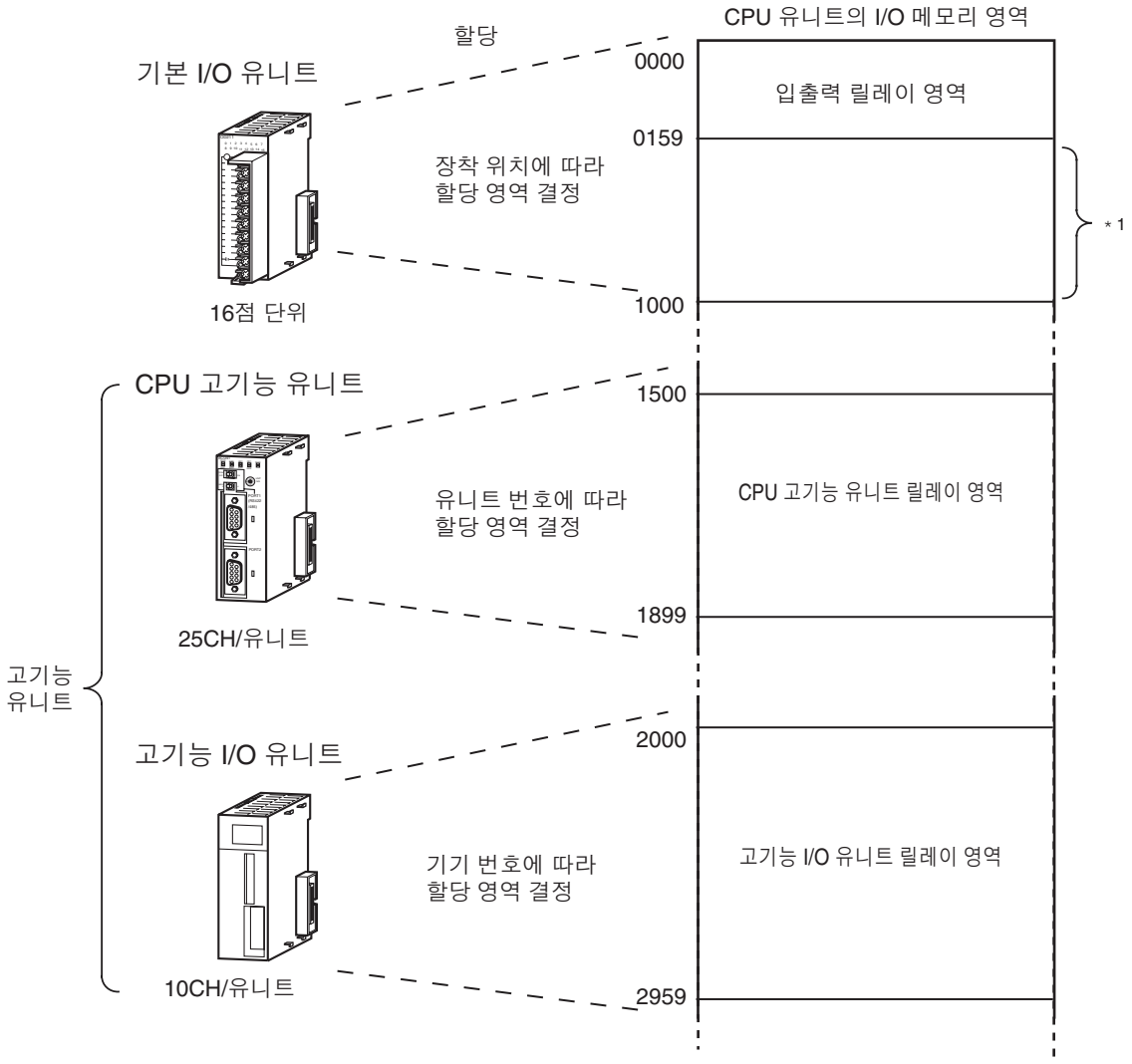
8-1 I/O 할당 8-2
8-1-1 I/O 할당이란? 8-2
8-1-2 자동 할당 8-5
8-1-3 수동 할당 8-9
8-1-4 I/O 테이블 관련 이상과 주의 사항 8-17
8-2 고기능 유니트 설정 8-20
8-2-1 파라미터 설정 8-20
8-2-2 CPU 유니트와 데이터 교환 8-24

8-1 I/O 할당

8-1-1 I/O 할당이란?

CPU 유니트와 장착된 유니트가 데이터를 교환하기 위한 영역을 CPU의 I/O 메모리 영역에 할당하는 것을 I/O 할당이라고 합니다.
유니트는 다음의 3개 그룹으로 나뉘며 그룹별로 할당 방법이 다릅니다.

- CJ 시리즈 기본 I/O 유니트
- CJ 시리즈 고기능 I/O 유니트
- CJ 시리즈 CPU 고기능 유니트



*1 랙 시작 채널 설정을 변경하면(기본값은 0000CH부터 채우기), 0000~0999CH에 할당할 수 있습니다.
랙 시작 채널은 CX-Programmer에서 설정합니다.

I/O 할당 방법

I/O 할당 방법은 다음 2가지입니다.

- 자동 할당(기본값)
  유니트 전원을 켰을 때, 장착되어 있는 유니트의 상태(장착 위치나 기기 번호)에 따라 I/O를 할당합니다.
- 수동 할당(I/O 테이블 작성)
  사용자가 임의로 할당 위치를 지정한 것을 CX-Programmer를 사용하여 I/O 테이블로 작성합니다.
  I/O 테이블을 작성하면 CPU 유니트의 전원을 켰을 때 I/O 테이블과 장착되어 있는 유니트의 상태가 일치하지 않는 경우 에러가 발생합니다.

I/O 할당 종류에 따른 차이는 다음과 같습니다.
I/O 할당의 종류는 목적이나 I/O 할당 동작에 맞게 선택하십시오.

I/O 할당 방법 / 항목	자동 할당	수동 할당	
		장착 상태에서 I/O 테이블 작성	임의의 할당으로 I/O 테이블 작성
목적	• 유니트의 장착 에러 확인 필요 없음 • 임의의 할당(빈 채널 확보 포함)을 하지 않는 경우	유니트의 장착 에러를 확인하고 싶은 경우	• 임의의 할당(빈 채널 확보 포함)을 하고 싶은 경우 • 장착 유니트를 준비할 수 없는 상태(오프라인)에서 I/O 할당을 실행하고 싶은 경우
I/O 테이블 작성	작성 안 함	작성	
I/O 테이블과 장착 상태 조회	조회 안 함	조회 (일치하지 않는 경우, 운전 시작 불가능)	
I/O 할당 조작	자동 생성	수동 조작	
CX-Programmer	필요 없음	필요	
I/O 테이블 작성 시 장착 유니트	필요	필요	필요 없음
할당 방법	자동 할당(CPU 유니트에서의 장착 순서)		수동 할당(그룹별로 시작 주소 지정 가능)

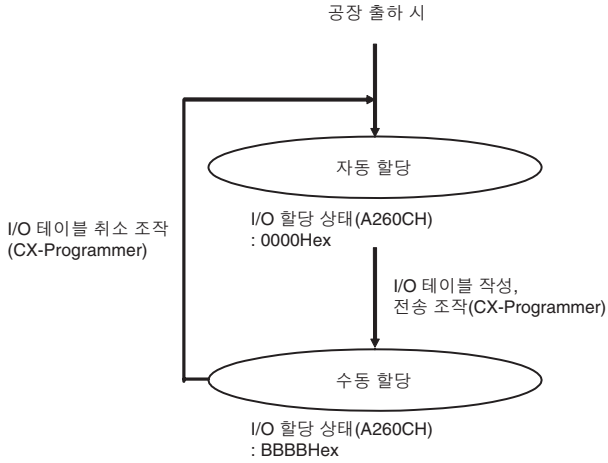
● I/O 할당 상태 확인 방법

I/O 할당이 자동 할당인지, 수동 할당인지를 I/O 할당 상태 플래그(A260CH)에서 확인할 수 있습니다.

주소	명칭	내용	
A260CH	I/O 할당 상태 플래그	0000Hex	자동 할당
		BBBBHex	수동 할당

● I/O 할당 상태의 이동

I/O 할당 방법은 다음과 같이 이동합니다.

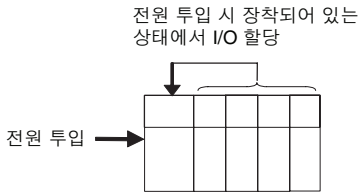


- CX-Programmer에서 'I/O 테이블 삭제' 조작을 실행하면 I/O 할당 상태가 자동 할당으로 되돌아가는 동시에 랙별 시작 채널 설정도 클리어됩니다. 또한 CPU 유니트 시스템 설정 영역도 초기화됩니다.

8-1-2 자동 할당

자동 할당(기본값)

전원을 켤 때마다 장착 유니트의 장착 상태에 따라 I/O 할당을 하는 방식(상태)입니다.
자동으로 I/O 할당이 되므로 주변 툴의 조작은 필요하지 않습니다.
I/O 할당의 할당 내용은 CPU 전원을 켤 때마다 갱신됩니다.
수동 설정에서 자동 설정으로 변경하려면 주변 툴에서 I/O 테이블의 취소 조작을 실행하십시오.



참고

I/O 테이블을 작성하지 않으므로 수동 할당과 달리 장착 유니트와의 조회 확인을 수행할 수 없습니다.

할당 방법

I/O 할당을 자동 할당으로 수행하면 장착 상태에서 다음과 같이 자동 할당됩니다.
I/O 할당 조작은 하지 않습니다.

유니트의 종류	할당 방법
기본 I/O 유니트	장착 위치에 따라 순서대로 할당됩니다.
고기능 I/O 유니트	전면 로터리 스위치(기기 번호 및 유니트 번호)에 의해 할당됩니다.
CPU 고기능 유니트	

유니트 종류별 I/O 할당 방법

유니트 종류(기본 I/O 유니트, 고기능 I/O 유니트, CPU 고기능 유니트)별 I/O 할당 방법을 설명합니다.

- **기본 I/O 유니트의 I/O 할당**

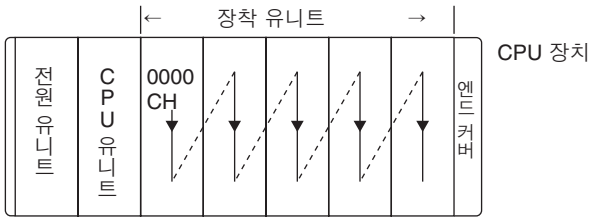
CJ 시리즈 기본 I/O 유니트
할당 영역: 입출력 릴레이 영역(0000~0159CH)
연결 위치: CJ 시리즈 CPU 장치, 증설 장치

할당 방법:

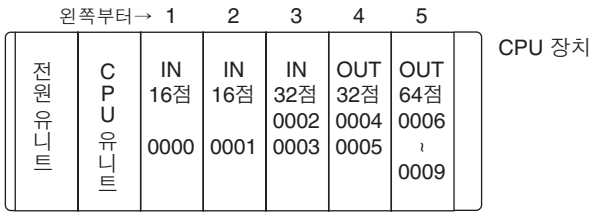
- CPU 장치에서 할당하는 경우

  CPU 장치의 가장 왼쪽(CPU 유니트에서 가장 가까운 위치)에 연결된 유니트를 0000CH로 하여, 왼쪽에서 오른쪽으로 각 기본 I/O 유니트의 점유 채널 수만큼 채널(16점) 단위로 할당됩니다.

  (주) 1~16점의 유니트는 1CH(16점), 17~32점의 유니트는 2CH(32점)를 점유합니다.



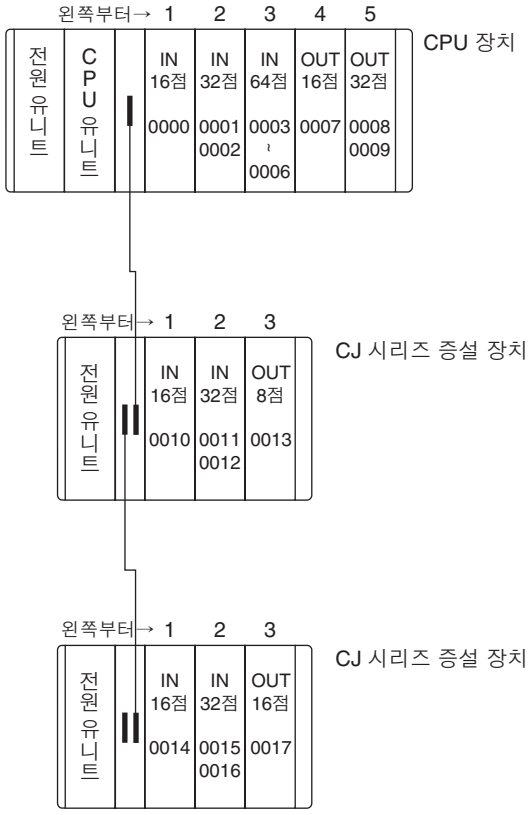
예:



- CJ 시리즈 증설 장치

  CPU 장치의 다음 증설 장치에서 순서대로 할당됩니다. 증설 장치에서는 CPU 장치에서와 마찬가지로 왼쪽에서 오른쪽으로 채널(16점) 단위로 할당됩니다.

예:



● 고기능 I/O 유니트의 I/O 할당

CJ 시리즈 고기능 I/O 유니트
할당 영역: 고기능 I/O 유니트 영역(2000~2959CH)
연결 위치: CJ 시리즈 CPU 장치, CJ 시리즈 증설 장치

할당 방법: 각 유니트 전면의 기기 번호 설정에 따라, 아래와 같이 미리 정해진 영역에 1유니트당 10CH가 할당됩니다.

기기 번호	할당 CH 수	할당 영역
0	10CH	2000~2009CH
1	10CH	2010~2019CH
2	10CH	2020~2029CH
:	:	:
15	10CH	2150~2159CH
⋮	⋮	⋮
95	10CH	2950~2959CH

고기능 I/O 유니트의 연결은 기본 I/O 유니트의 입출력 릴레이 영역에 대한 할당에 영향을 주지 않습니다. 기본 I/O 유니트는 고기능 I/O 유니트를 건너뛰고(없는 것으로 생각하고) 입출력 릴레이 영역에 할당됩니다.

● CPU 고기능 유니트의 I/O 할당

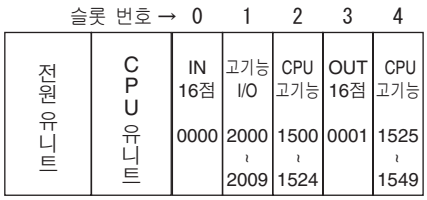
CJ 시리즈 CPU 고기능 유니트
할당 영역: CPU 고기능 유니트 릴레이 영역(1500~1899CH)
연결 위치: CJ 시리즈 CPU 장치, 증설 장치

할당 방법: 각 유니트 전면의 유니트 번호 설정에 따라, 아래와 같이 미리 정해진 영역에 1유니트당 25CH가 할당됩니다.

유니트 번호	할당 CH 수	할당 영역
0	25CH	1500~1524CH
1	25CH	1525~1549CH
2	25CH	1550~1574CH
:	:	:
15	25CH	1875~1899CH

CPU 고기능 유니트의 연결은 기본 I/O 유니트의 입출력 릴레이 영역에 대한 할당에 영향을 주지 않습니다. 또한 고기능 I/O 유니트의 기기 번호와 중복되어도 괜찮습니다.

예:



슬롯 번호	사양	형식	할당 CH 수	할당 영역	기기 번호 또는 유니트 번호	유니트의 그룹
0	DC 16점 입력	CJ1W-ID211	1CH	0000CH	—	기본 I/O 유니트
1	아날로그 유니트	CJ1W-AD081	10CH	2000~ 2009CH	0	고기능 I/O 유니트
2	시리얼 커뮤니케이션 유니트	CJ1W-SCU41	25CH	1525~ 1549CH	1	CPU 고기능 유니트
3	트랜지스터 16점 출력	CJ1W-OD211	1CH	0001CH	—	기본 I/O 유니트
4	Controller Link 유니트	CJ1W-CLK21	25CH	1550~ 1574CH	2	CPU 고기능 유니트
—	CPU 유니트(내장 EtherNet/IP 포트) (*)	CJ2H-CPU68-EIP	25CH	1500~ 1524CH	0	CPU 유니트(CPU 고기능 유니트 취급)

* CJ2H-CPU6□-EIP, CJ2M-CPU3□에서는 내장 EtherNet/IP 포트에서 네트워크 통신 상태 등을 저장하는 영역으로 CPU 고기능 유니트용 할당 영역이 사용됩니다. 할당 영역은 CPU 고기능 유니트와 동일하게 CJ2H-CPU6□-EIP, CJ2M-CPU3□ 전면의 로터리 스위치를 통해 유니트 번호 설정에 따라 할당됩니다.

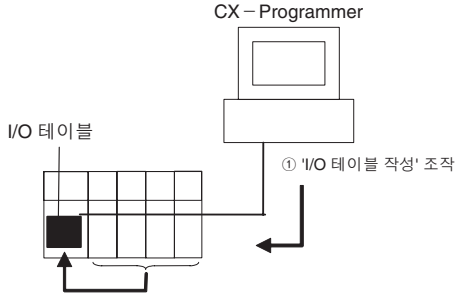
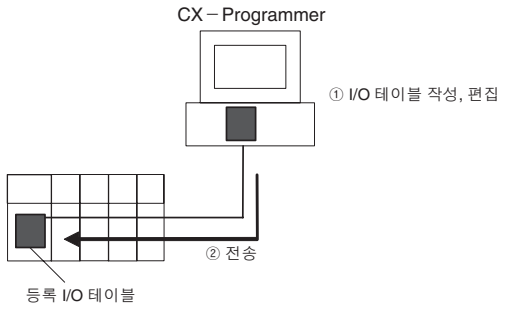
8-1-3 수동 할당

수동 할당(I/O 테이블 작성)

사용자가 임의로 작성한 I/O 테이블을 바탕으로 I/O 할당을 수행하는 방식(상태)입니다.
전원을 켜면 장착 유니트와의 조회 확인을 수행합니다.

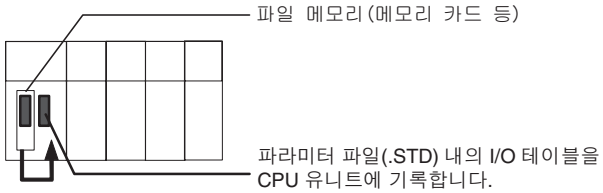
● I/O 테이블 작성 방법

I/O 테이블은 다음 2가지 방법으로 작성합니다.

장착 상태에서 I/O 테이블 작성	임의의 할당으로 I/O 테이블 작성
온라인 상태일 때 장착 상태에서 I/O 테이블을 작성합니다.  	오프라인일 때 CX-Programmer에서 I/O 테이블을 임의로 작성하고 CPU 유니트에 전송합니다. 

참고

전원을 켜면 메모리 카드에 저장된 I/O 테이블을 CPU 유니트에 자동으로 전송하는 조작도 가능합니다.



● 유니트의 잘못된 연결 확인 유무

수동 할당으로 I/O 테이블을 작성하면 I/O 테이블과 유니트 장착 상태를 조회하여 확인합니다. 일치하지 않는 경우에는 운전을 시작할 수 없습니다.
등록한 I/O 테이블과 장착 유니트의 장착 상태가 일치하지 않는 경우, I/O 설정 이상(A401CH 비트 10이 ON)이 됩니다(운전 정지 이상으로 운전 시작 불가능).

장착 상태에서 I/O 테이블을 작성하는 경우

● 할당 방법

I/O 할당되는 상태는 자동 할당과 동일합니다.
할당 방법에 대한 자세한 내용은 '8-1-2 자동 할당'을 참조하십시오.

● I/O 할당 조작

온라인 상태에서 다음과 같은 조작을 통해 장착 상태에서 I/O 테이블을 작성합니다.

1 **메인 창의 프로젝트 트리에서 [I/O 테이블]을 두 번 클릭합니다.**
I/O 테이블 창이 표시됩니다.

2 **[옵션] - [만들기]를 선택합니다.**
장착된 유니트의 종류와 연결 위치가 CPU 유니트에 I/O 테이블로 기록됩니다.

● I/O 테이블 조회

온라인 상태에서 다음과 같은 조작을 통해 장착 상태와 I/O 테이블을 조회하여 확인합니다.

1 **메인 창의 프로젝트 트리에서 [I/O 테이블]을 두 번 클릭합니다.**
I/O 테이블 창이 표시됩니다.

2 **[옵션] - [검증]을 선택합니다.**
장착된 유니트의 종류와 연결 위치, 등록 I/O 테이블을 검증하고, 그 결과가 표시됩니다.

임의의 할당으로 I/O 테이블을 작성하는 경우

● 할당 방법

장착 상태에서 자동 할당하지 않고 임의로 I/O 테이블을 작성하면 다음과 같은 수동 할당이 가능합니다.

- 임의의 장착 위치에 있는 유니트에 채널 번호 할당
- 빈 채널 확보
- 랙 시작 채널 지정
- 장착 유니트가 준비되지 않은 상태에서 I/O 할당

다음과 같이 할당할 수 있습니다.

유니트의 종류	할당 방법
기본 I/O 유니트	완전히 임의는 아니지만 임의의 랙의 임의의 장착 위치에 있는 유니트에 시작 채널 번호를 할당하여, 이후 장착 순서로 이용할 수 있습니다. 63개의 임의의 할당 그룹을 작성할 수 있습니다.
고기능 I/O 유니트	전면 로터리 스위치(기기 번호 및 유니트 번호)에 의해 할당됩니다.
CPU 고기능 유니트	

● I/O 할당 조작

CX-Programmer에서 I/O 테이블을 작성하고 PLC에 전송합니다.
I/O 테이블을 작성할 때는 오프라인에서, PLC로 전송할 때는 온라인에서 실행합니다.

1 **오프라인에서 메인 창의 프로젝트 트리에 있는 [I/O 테이블]을 두 번 클릭합니다.**
I/O 테이블 창이 표시됩니다.

2 **편집할 랙을 두 번 클릭합니다.**
선택한 랙의 슬롯이 표시됩니다.

3 **유니트를 할당할 슬롯을 마우스 오른쪽 버튼으로 클릭하고, 풀다운 메뉴에서 할당할 유니트를 선택합니다.**

4 **온라인 상태에서 [옵션] – [PLC로 전송]을 선택합니다.**
I/O 테이블이 전송됩니다.

● 임의의 장착 위치에 있는 유니트에 시작 채널 번호를 할당하거나 빈 채널을 확보할 때

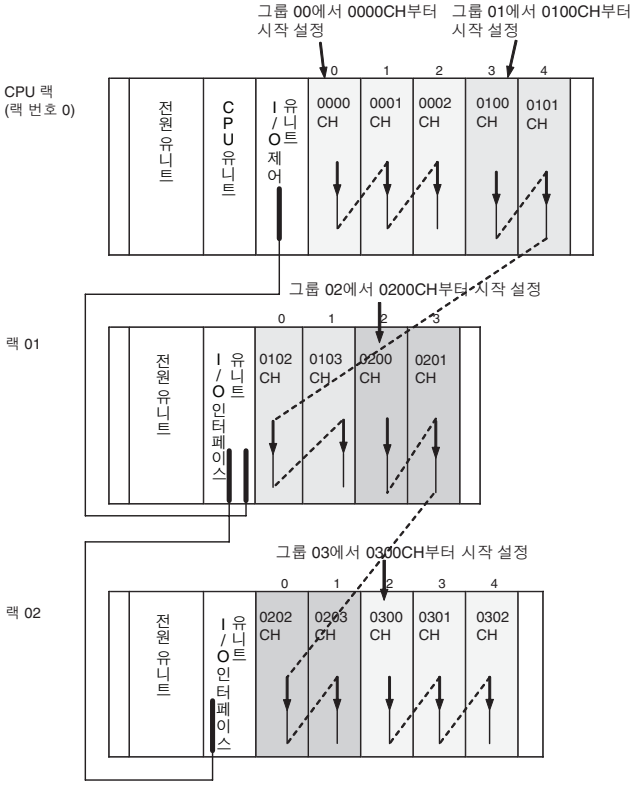
CX-Programmer에서 임의의 랙의 임의의 장착 위치에 있는 유니트에 시작 채널 번호를 할당하여, 이후 장착 순서로 이용할 수 있습니다. 시작 채널 번호를 설정하여 장착 순서로 이용할 수 있는 그룹은 63개 작성할 수 있습니다.

할당 방법

시작 채널 번호를 설정한 랙에서는 장착 위치에 있는 유니트(이후 '슬롯')의 왼쪽에서 오른쪽으로, 설정된 채널 번호를 선두로 하여 할당됩니다.

- 같은 랙에 시작 채널이 설정된 슬롯이 있는 경우 바로 앞의 슬롯까지 같은 그룹이 됩니다.
- 같은 랙에 시작 채널이 설정된 슬롯이 없는 경우에는 다음 랙의 시작 채널이 설정된 슬롯 직전까지 같은 그룹이 됩니다.
- 그룹 00은 CPU 랙의 슬롯 00의 시작 채널 번호를 설정합니다.
  그룹 01~63은 임의의 랙, 슬롯에 대해 시작 채널 번호를 설정할 수 있습니다.

예) 각 랙의 중간부터 랙 및 슬롯 시작 CH를 설정하는 경우
모두 1유니트＝16점으로 한 경우의 예입니다.



랙 및 슬롯 시작 CH 설정

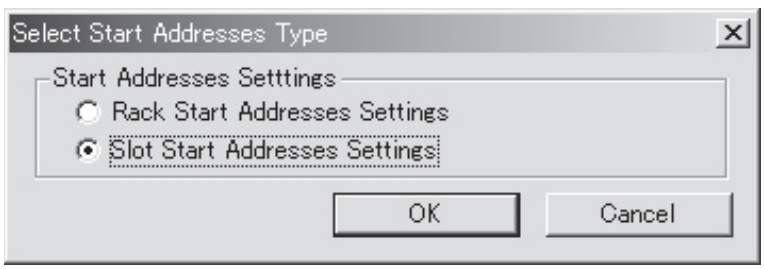
그룹 번호	랙 번호	슬롯 번호	시작 CH
00(*)	CPU 랙	00	0000
01	CPU 랙	03	0100
02	랙 1	02	0200
03	랙 2	02	0300

* 그룹 00은 CPU 랙의 슬롯 00을 설정합니다. 시작 채널 번호는 임의로 설정됩니다. 그룹 01에서 63까지는 랙이고 슬롯은 임의로 설정합니다.

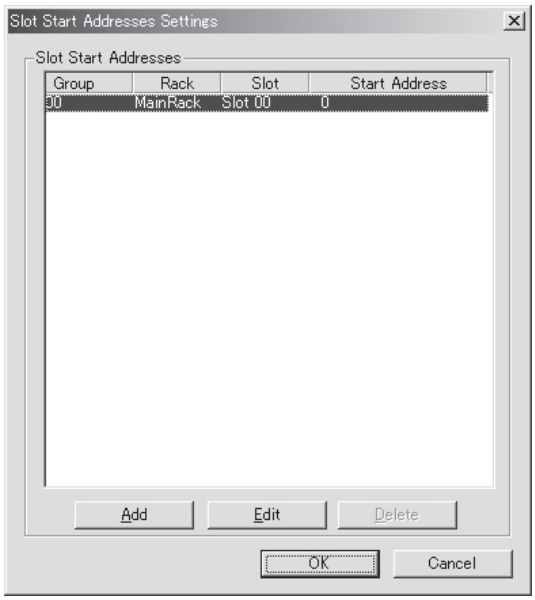
CX-Programmer에서 조작 방법

1 **CX-Programmer의 I/O 테이블 창에서 [옵션] – [랙/슬롯 시작 주소]를 선택합니다.**

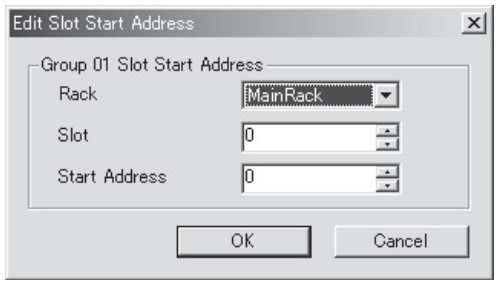
2 **[슬롯 시작 주소 설정]을 선택하고 [확인]을 클릭합니다.**
[슬롯 시작 주소 설정] 대화 상자가 표시됩니다.



3 **그룹 00은 CPU 랙의 슬롯 00의 시작 채널 번호를 설정합니다.**



4 **시작 주소(채널 번호)를 0000CH 이외로 설정하려면 [편집] 버튼을 클릭합니다.**
다음과 같은 [슬롯 시작 주소 편집] 대화 상자가 표시됩니다.



5 **[추가] 버튼을 클릭하여 그룹 01 이후의 랙, 슬롯, 시작 주소(채널 번호)를 설정하십시오.**

설정 항목	설정 범위	초기값	비고
그룹 번호	00~63	00	편집 대상 그룹 표시 설정 순서대로 자동 할당됩니다.
랙 번호	CPU 랙, 랙 01~랙 07	CPU 랙	그룹 00의 경우 CPU 랙의 슬롯 00을 설정합니다.
슬롯 번호	00~03	0	
시작 주소	0~999	0	—

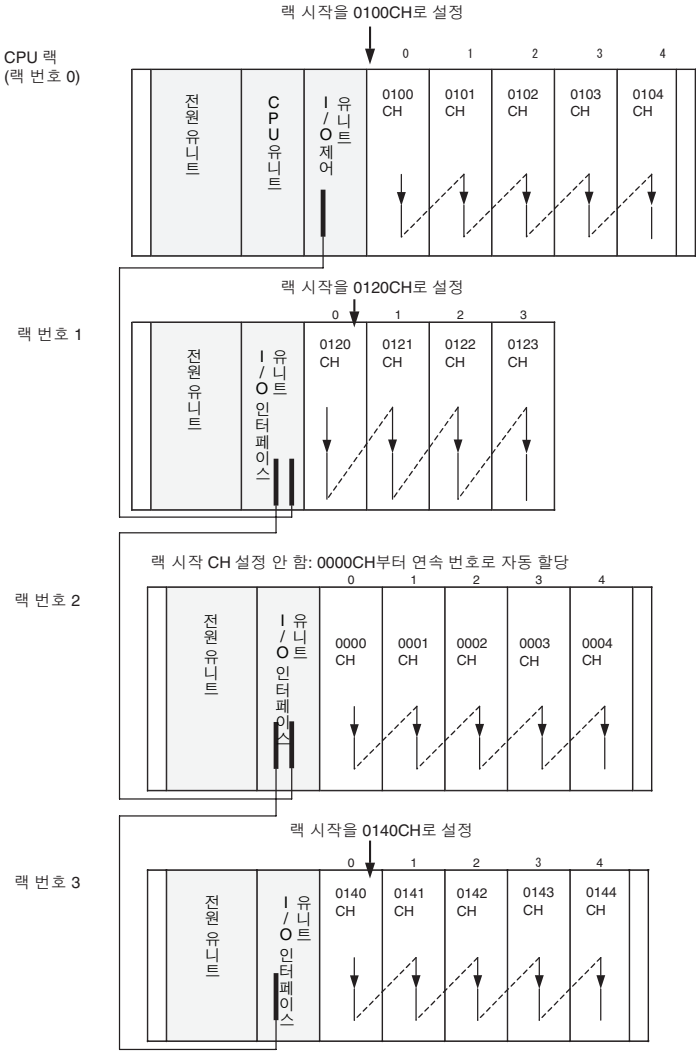
랙별로 시작 채널 번호를 할당할 때

CX-Programmer에서 랙별로 시작 채널 번호를 할당할 수 있습니다. 예를 들어, CPU 랙은 0000CH부터, 다음의 증설 랙은 0100CH부터, 그리고 그 다음의 증설 랙은 0200CH부터 시작하는 방식으로 할당할 수 있습니다.
유니트의 채널 번호를 확인하는 경우 CPU 장치에서 순서대로 계산할 필요가 없으므로 유지 보수가 간편합니다.

● 할당 방법

시작 채널 번호를 설정한 랙은 장착된 유니트에 대해 설정한 채널 번호에 따라 순서대로 왼쪽→오른쪽으로 채널 번호가 할당됩니다.
시작 채널 번호를 설정하지 않은 랙의 경우, 랙 번호는 작은 번호에서 큰 번호 순서로 0000CH부터 채워서 채널 번호가 할당됩니다.

예: 랙 번호의 시작 채널을 설정한 경우



랙 시작 CH 설정

랙 번호	시작 CH
CPU 랙	0100
랙 1	0120
랙 2	0000
랙 3	0140

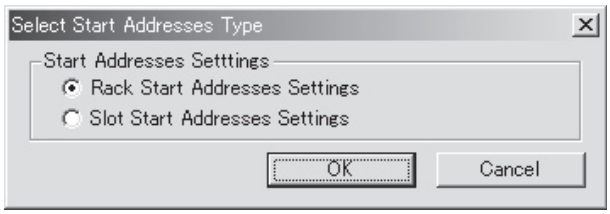
(주) 랙 번호 '0'~'3'은 실제로 I/O 연결 케이블로 연결된 순서에 따라 고정됩니다(CPU 장치의 랙 번호는 '0'이고 이후 순서대로 '1'~'3'이 됩니다). 연결 순서와 다른 순서로 랙 번호를 설정할 수는 없습니다.

CX-Programmer에서 조작 방법

오프라인으로 조작합니다.

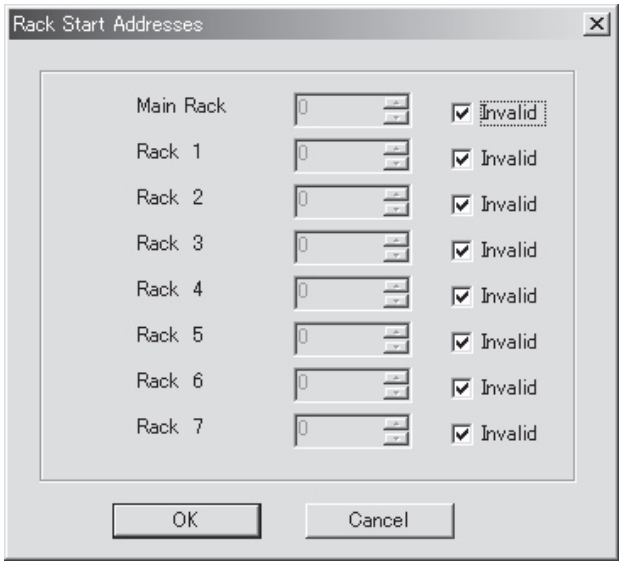
1 **CX-Programmer의 I/O 테이블 창에서 [옵션] – [랙/슬롯 시작 주소]를 선택합니다.**

다음과 같은 대화 상자가 표시됩니다.



2 **[랙 시작 주소 설정]을 선택하고 [확인]을 클릭합니다.**

3 **[랙 시작 주소] 대화 상자에서 각 랙 번호의 시작 주소(슬롯 00의 채널 번호)를 설정합니다.**



설정 항목	설정 범위	초기값	비고
랙 시작 주소	0~900	0	모든 랙 공통
랙 시작 주소 유효/무효	유효/무효	무효	

(주)1 PLC 기종/CPU 유니트 형식에 관계 없이 3개의 랙을 설정할 수 있습니다.

2 CX-Programmer 화면에는 랙 7까지 표시되지만 CJ2 CPU에서 설정할 수 있는 것은 랙 3까지입니다.

사용상의 주의 사항

- 각 랙의 할당 채널이 중복되지 않도록 랙별 시작 채널을 설정하십시오. 설정할 수 있는 시작 채널 번호의 설정 범위는 0000~0900CH입니다.
  각 랙의 할당 채널이 중복되는 경우 또는 각 랙별 시작 채널 설정이 0901CH 이상인 경우, 증설 장치 중복 랙 번호 플래그(A409.00~409.03: 랙 번호 '0'~'3'에 대응)가 1(ON)이 됩니다. 동시에, 번호 이중 사용 플래그(A401.13)가 1(ON)이 됩니다.
- 랙별 채널 할당을 설정한 후에는 반드시 CX-Programmer에서 'I/O 테이블 작성 조작'(I/O 테이블 창의 [옵션]-[만들기])을 수행하십시오.
  'I/O 테이블 작성 조작'을 수행하면 할당된 입출력 채널이 등록됩니다.
- 'I/O 테이블 작성 조작' 후, 등록된 채널 점유 수가 다른 유니트 또는 I/O 종류가 다른 유니트에 대한 변경 또는 유니트의 추가, 삭제를 수행하면, 등록 시의 상태와 다르기 때문에 'I/O 설정 이상(A401.10)', 'CPU 고기능 설정 이상(A402.03)' 또는 '고기능 I/O 설정 이상(A402.02)'이 발생합니다.
  유니트 삭제의 경우 'I/O 테이블 변경' 조작을 통해 빈 채널을 확보합니다. 유니트 변경 및 추가의 경우는 변경 및 추가 이후 프로그램상의 채널을 모두 변경하고, 다시 'I/O 테이블 작성 조작'을 수행하십시오.
- CX-Programmer에서 'I/O 테이블 삭제' 조작을 실행한 경우는 랙별 시작 채널 설정이 클리어됩니다.

8-1-4 I/O 테이블 관련 이상과 주의 사항

I/O 테이블 관련 이상

● I/O 설정 이상 및 번호 이중 사용 에러

I/O 테이블 작성 조작 또는 I/O 테이블 편집/전송 조작을 수행했을 때는(사용자 조작에 의한 I/O 할당 방식) 다음과 같은 경우, I/O 테이블을 작성할 수 없습니다.

이런 경우	이상 이름	내용	대책
장착 상태와 I/O 테이블의 조회 이상일 때	I/O 설정 이상 (운전 정지 이상)	실제로 기본 시스템(CPU 장치 및 증설 장치)에 연결된 유니트의 종류와 연결 위치 정보, 등록 I/O 테이블의 정보가 일치하지 않을 때(조회 이상), 'I/O 설정 이상'(A401.10이 ON)이 됩니다. 이 경우, 운전할 수 없습니다. CPU 유니트의 전면 RUN LED가 꺼지고 ERR/ALM이 적색으로 켜집니다.	• 시작할 때 유니트 수가 다르지 않은지 확인하고, 다른 경우 일단 전원을 끈 후 유니트를 정확하게 다시 장착하십시오. • 유니트 수에 문제가 없으면 CX-Programmer에서 'I/O 테이블 조회' 조작을 통해 I/O 테이블과 다른 유니트를 확인한 후, 일단 전원을 끄고 올바른 유니트를 장착하십시오. • I/O 테이블에 에러가 있으면 CX-Programmer에서 'I/O 테이블 작성' 또는 'I/O 테이블 편집/전송 조작'을 사용하여 I/O 테이블을 정확하게 다시 작성하십시오.
고기능 유니트의 유니트 번호(기기 번호)가 중복될 때	번호 이중 사용 에러(운전 정지 이상)	다음 중 하나에 해당하는 경우 '번호 이중 사용 에러'(A401.13이 ON)가 됩니다. • CPU 고기능 유니트의 기기 번호가 중복되는 경우 • 고기능 I/O 유니트의 기기 번호가 중복되는 경우 • 기본 I/O 유니트의 할당 채널이 중복되는 경우 • 증설 장치의 랙 번호가 중복되는 경우 이 경우, 운전할 수 없습니다. CPU 유니트의 전면 RUN LED가 꺼지고 ERR/ALM이 적색으로 켜집니다.	CPU 고기능 유니트의 유니트 번호 또는 고기능 I/O 유니트의 기기 번호를 확인하고, 중복되지 않도록 수정한 후 장치의 전원을 ON→OFF→ON하십시오. A409.00~A409.03 중에서 1(ON)이 된 대응 랙 번호의 유니트 할당 채널을 확인하고 다른 랙의 유니트 할당 채널과 중복되지 않도록 수정한 후 장치의 전원을 ON→OFF→ON하십시오. A409.00~A409.03 중에서 1(ON)이 된 대응 랙 번호의 시작 CH 설정을 확인하고 CX-Programmer에서 0900CH 이하를 다시 설정하십시오.

그 밖에 I/O 지점 수 초과(A401.11이 ON), I/O 버스 이상(A401.14가 ON)일 때도 I/O 테이블을 작성할 수 없습니다.

● I/O 테이블 작성 에러 상세 정보

CX-Programmer에서 'I/O 테이블 작성 조작'을 실행했지만 I/O 테이블 작성 조작 에러가 발생한 경우, I/O 테이블을 작성하지 못한 요인을 특수 보조 릴레이 A261CH('I/O 테이블 작성 에러 상세 정보')에 저장합니다.
따라서 어느 유니트가 I/O 테이블 작성의 에러 요인인지 쉽게 파악할 수 있습니다.

명칭		주소		내용
		CH	비트	
I/O 테이블 작성에러 상세 정보	CPU 고기능 시스템 설정 영역 초기화 이상에 의한 I/O 테이블 작성 에러	A261CH	00	CPU 고기능 시스템 설정 영역 초기화 이상이 발생한 경우, 1(ON)이 됩니다. I/O 테이블 작성 조작이 정상적으로 종료되면 0(OFF)으로 되돌아갑니다.
	I/O 지점 수 초과에 의한 I/O 테이블 작성 에러		02	I/O 지점 수가 초과되는 유니트 구성인 경우, 1(ON)이 됩니다. I/O 테이블 작성 조작이 정상적으로 종료되면 0(OFF)으로 되돌아갑니다.
	번호 이중 사용 에러에 의한 I/O 테이블 작성 에러		03	유니트 구성에서 번호 이중 사용 에러가 발생하는 경우, 1(ON)이 됩니다. I/O 테이블 작성 조작이 정상적으로 종료되면 0(OFF)으로 되돌아갑니다.
	I/O 버스 이상에 의한 I/O 테이블 작성 에러		04	I/O 버스 이상이 발생한 경우, 1(ON)이 됩니다. I/O 테이블 작성 조작이 정상적으로 종료되면 0(OFF)으로 되돌아갑니다
	고기능 유니트 이상에 의한 I/O 테이블 작성 에러		07	고기능 유니트 이상이 발생한 경우, 1(ON)이 됩니다. I/O 테이블 작성 조작이 정상적으로 종료되면 0(OFF)으로 되돌아갑니다.
	I/O 인식 미완료에 의한 I/O 테이블 작성 에러		09	I/O 인식이 완료되지 못한 경우, 1(ON)이 됩니다. I/O 테이블 작성 조작이 정상적으로 종료되면 0(OFF)으로 되돌아갑니다.

I/O 테이블 설정 주의 사항

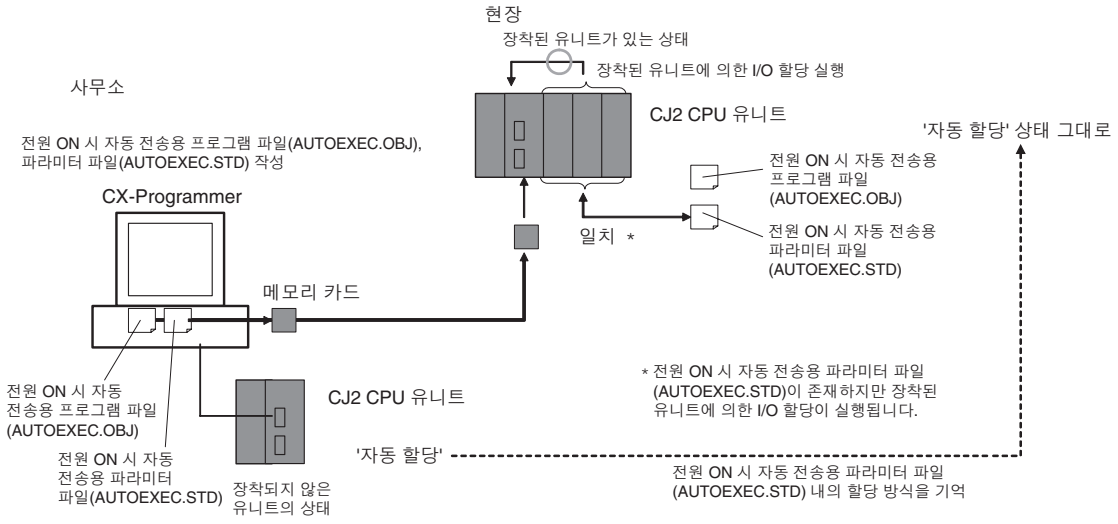
I/O 테이블을 설정한 후, 유니트 변경 등을 하는 경우 채널 번호가 중복되지 않도록 주의하십시오.
(예를 들어, 1CH 분을 점유하는 유니트를 2CH 분을 점유하는 유니트와 교환하면 다음 채널 번호를 점유하고 있던 다른 유니트와 채널 번호가 중복되는 경우가 생길 수 있습니다.) 유니트 변경 후 다시 전원을 켜면 CPU 유니트가 I/O 테이블과 실제 I/O(장착한 유니트의 I/O 상태)가 다르지 않은지 확인하고, 채널 번호가 중복되면 에러를 발생시킵니다.
에러가 발생하면 이후의 I/O 테이블 변경을 수행할 수 없게 되므로 새로 I/O 테이블을 편집해서 다시 전송해야 합니다.
I/O 테이블을 편집할 때는 임의의 시작 채널 번호 설정에 의한 채널 번호 중복이나 설정 범위의 에러에 대해 CX-Programmer가 확인하기 때문에 문제는 없습니다.

메모리 카드 사용 시의 주의 사항

전원 ON 시 자동 전송용 파라미터 파일(AUTOEXEC.STD)에 CPU 유니트의 I/O 테이블이 어떤 방식으로 작성되었는지 기록됩니다. 메모리 카드에서 전원 ON 시 자동 전송을 실행하면 기록된 방식에 따라 전원을 켰을 때 자동으로 I/O 할당을 할지 또는 I/O 테이블과 조회할지 판단합니다.

전원 ON 시 자동 전송용 파라미터 파일(AUTOEXEC.STD)을 작성한 CPU 유니트의 I/O 테이블 작성 방식에 따라 다음과 같이 달라집니다.

- 자동 할당(*)의 경우
  메모리 카드 내 전원 ON 시 자동 전송용 파라미터 파일의 I/O 테이블은 무효가 되며, 장착 유니트에 의해 I/O 할당됩니다.
  * I/O 테이블을 작성하지 않기 때문에 전원을 켤 때마다 CPU 유니트가 장착된 기본 I/O 유니트에 자동으로 채널 번호를 할당하는 방식을 말합니다.
- 수동 할당의 경우
  메모리 카드 내 전원 ON 시 자동 전송용 파라미터 파일의 I/O 테이블이 유효화되며, CPU 유니트는 I/O 테이블과 장착된 유니트를 조회합니다.



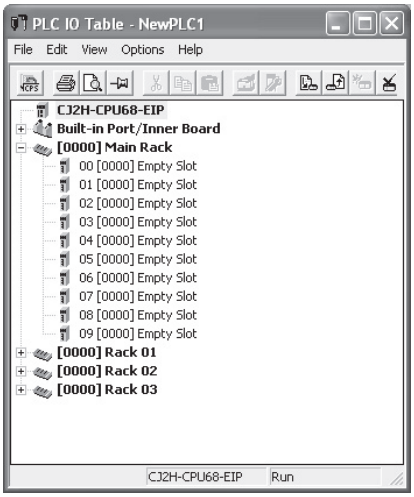
8-2 고기능 유니트 설정

8-2-1 파라미터 설정

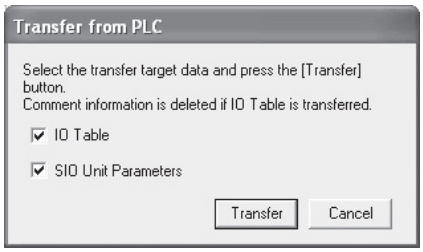
CX-Programmer의 I/O 테이블 창에서 CJ 시리즈 고기능 유니트(고기능 I/O 유니트, CPU 고기능 유니트)의 할당 DM, 할당 릴레이, CPU 고기능 유니트 시스템 설정(Ethernet 유니트만 해당)의 파라미터를 설정할 수 있습니다.
온라인/오프라인 양쪽에서 설정할 수 있지만 전송은 온라인에서만 가능합니다.

여기서는 'I/O 테이블 작성 조작' 후, 'I/O 테이블'과 '고기능 유니트 설정 데이터'를 PLC에서 I/O 테이블 창으로 전송한 다음 'I/O 테이블'과 '고기능 유니트 설정 데이터'를 편집하는 경우를 예로 들어 조작 방법을 설명합니다.
또한 오프라인에서 I/O 테이블에 고기능 유니트를 등록하고 '고기능 유니트 설정 데이터'를 편집하는 경우도 조작 방법은 동일합니다.

1 **CX-Programmer를 온라인으로 연결하여 I/O 테이블 창을 실행합니다.**



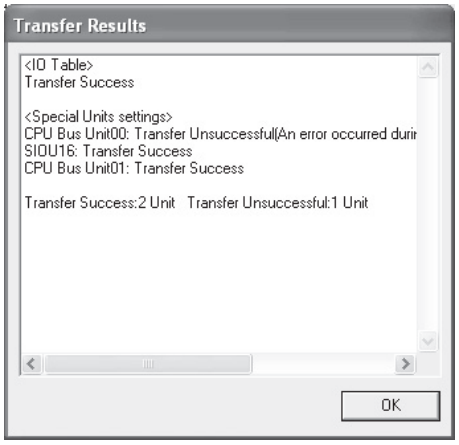
2 **I/O 테이블 창의 [옵션] 메뉴에서 [PLC로부터 전송]을 선택합니다.**
다음과 같은 [PLC로부터 전송] 대화 상자가 표시됩니다.



- 'I/O 테이블'과 '고기능 유니트 설정 데이터'를 선택할 수 있습니다.

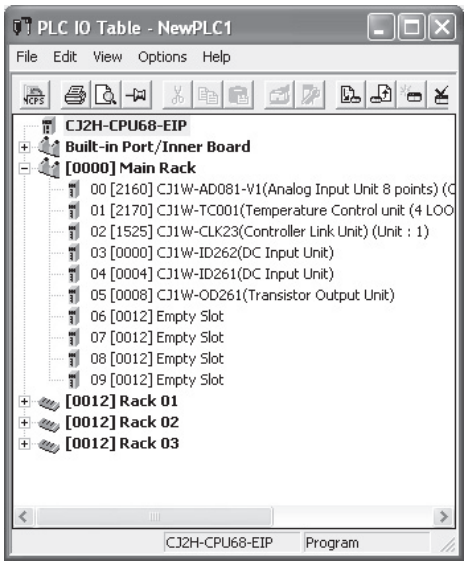
3 **전송할 데이터를 확인란에서 선택한 후 [전송] 버튼을 클릭합니다. 여기에서는 둘 다 선택되어 있습니다.**

'I/O 테이블'과 '고기능 유니트 설정 데이터'가 PLC에서 전송되고 전송 결과가 표시됩니다.



4 **전송 결과를 확인하고 [확인]을 클릭합니다.**

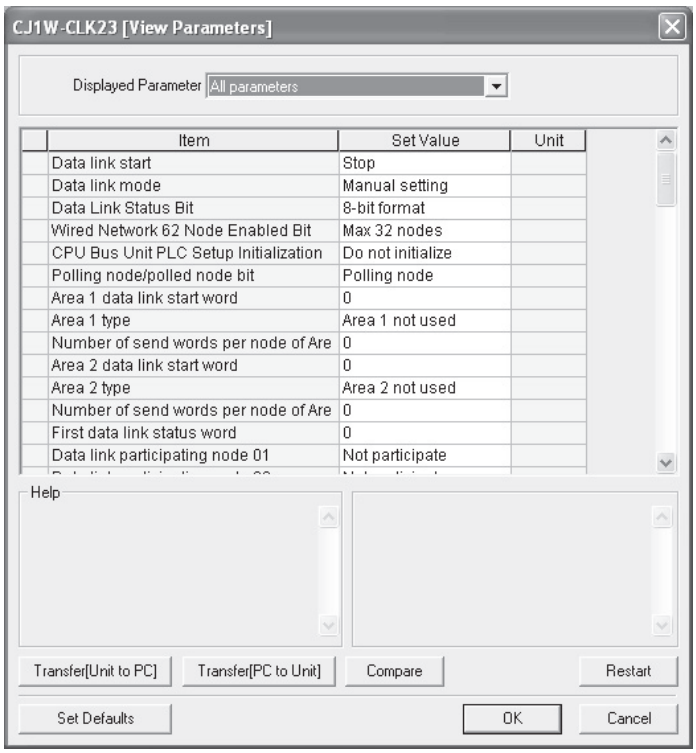
PLC에서 전송된 I/O 테이블이 표시됩니다.



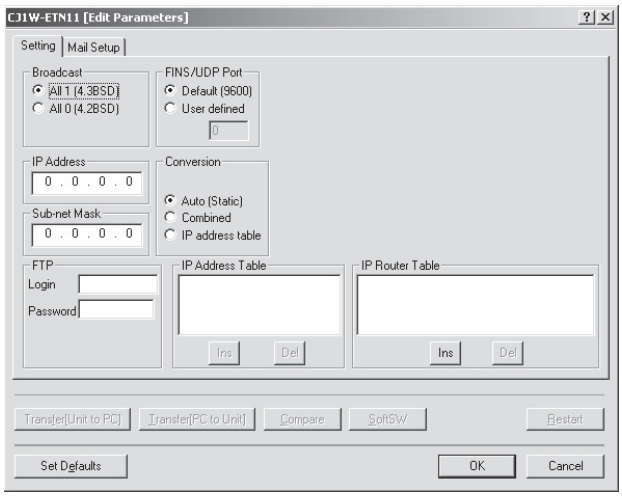
5 고기능 유니트 설정을 수행할 유니트를 선택하고 두 번 클릭하거나 [편집] 메뉴에서 [SIO 유니트 파라미터]를 선택합니다.

선택한 유니트의 [파라미터 편집] 대화 상자가 표시됩니다.

- Controller Link 유니트의 [파라미터 편집] 대화 상자
  (CPU 고기능 유니트 시스템 설정의 편집)



- Ethernet 유니트의 [파라미터 편집] 대화 상자
  (CPU 고기능 유니트 시스템 설정의 편집)

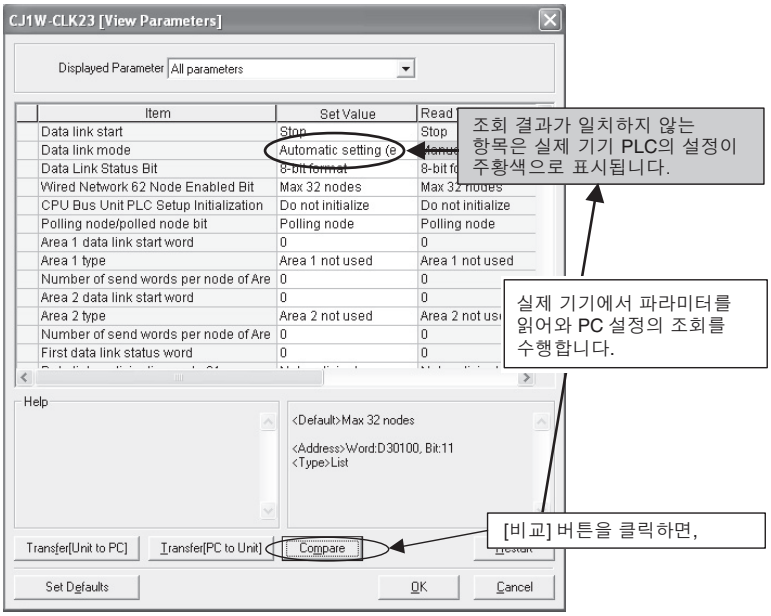


6 **파라미터를 편집하고 필요에 따라 [비교] 버튼을 클릭합니다.**

온라인 상태에서 [비교] 버튼을 클릭하면 I/O 테이블에서 설정한 내용과 실제 기기 PLC(CPU 유니트)의 유니트 설정(해당 할당 DM/릴레이 영역 등) 내용이 일치하는지 여부를 매회 확인할 수 있습니다.

비교 결과가 일치하지 않는 항목은 다음과 같이 실제 기기에서 읽은 내용이 주황색으로 표시됩니다.

예) Controller Link 유니트의 고기능 설정에 대한 비교 실행 결과



7 **설정한 파라미터를 유니트 데이터별로 CPU 유니트에 다운로드한 경우 [전송[PC→ 유니트]] 버튼을 클릭합니다.**

사용상의 주의 사항

I/O 테이블에서 고기능 유니트 설정을 수행하는 경우, [파라미터 편집] 대화 상자의 [전송[PC→ 유니트]] 버튼을 클릭하고 고기능 유니트 설정(각 유니트의 파라미터 중 할당 DM 영역 및 CPU 고기능 유니트 시스템 설정 영역)을 실제 기기 PLC에 전송하는 것만으로는 자동으로 해당 유니트가 다시 시작되지 않습니다(즉, 파라미터의 값이 유효화되지 않음). 파라미터의 값을 유효화하려면 [초기화] 버튼을 클릭하십시오.

참고

파라미터 설정 데이터 파일의 저장

- [파라미터 편집] 대화 상자에서 설정된 데이터는 파라미터 설정 데이터 파일(확장자는 xml)로 저장할 수 있습니다(I/O 테이블의 고기능 유니트를 마우스 오른쪽 버튼으로 클릭하여 [파라미터 저장] 선택).
- 저장한 파라미터 설정 데이터 파일(확장자는 xml)에서 파라미터를 읽을 수도 있습니다(단, 동일 형식의 고기능 유니트이어야 함).

8-2-2 CPU 유니트와 데이터 교환

고기능 유니트(고기능 I/O 유니트, CPU 고기능 유니트)와 CPU 유니트는 다음과 같이 데이터를 교환할 수 있습니다.

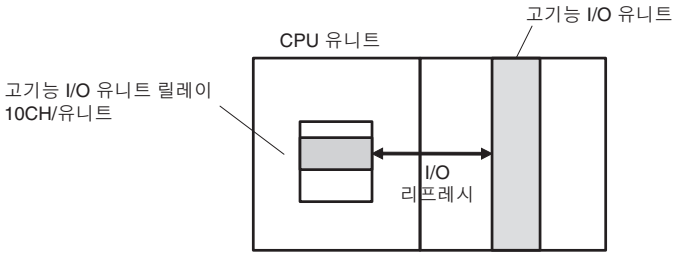
고기능 I/O 유니트의 경우

- **데이터 교환 방법**

I/O 리프레시

고기능 I/O 유니트 릴레이 영역을 I/O 리프레시하면 사이클마다 데이터를 교환합니다. 기본적으로 1유니트당 10CH가 기기 번호에 따라 할당됩니다. 할당에 대한 자세한 내용은 각 고기능 I/O 유니트의 사용 설명서를 참조하십시오.

고기능 I/O 유니트 릴레이 영역: 2000~2959CH(10CH×96호기)



할당 DM 영역의 전송

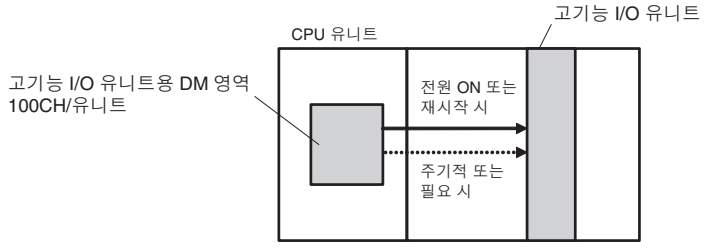
기본적으로 1유니트당 100CH가 기기 번호에 따라 할당됩니다.
데이터 전송 타이밍은 다음과 같으며 기종에 따라 다릅니다.

- 전원 ON 시 또는 재시작 시
- 주기적 데이터 교환
- 필요 시 데이터 교환

전송 방향에 대해서도 CPU 유니트→고기능 I/O 유니트의 방향 이외에, 기종에 따라 고기능 I/O 유니트→CPU 유니트의 방향도 가능합니다.
전송 타이밍 및 방향은 각 고기능 I/O 유니트의 사용 설명서를 참조하십시오.

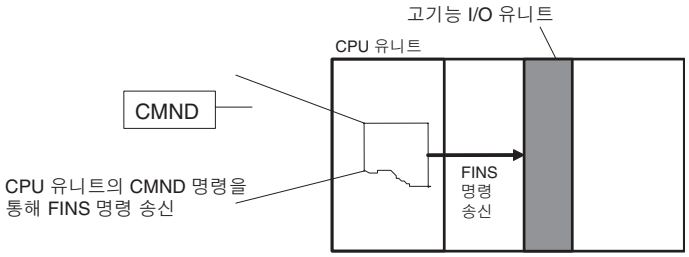
고기능 I/O 유니트용 DM 영역: D20000~D29599(100CH×96호기)

이 DM 영역으로의 전송은 고기능 I/O 유니트의 초기 설정 등에 사용됩니다. 단계 전환 등을 위해 프로그램의 명령을 통해 이 영역을 변경한 경우, 고기능 I/O 유니트 재시작 플래그(A502.00~507.15)를 OFF→ON으로 기동해야 합니다.

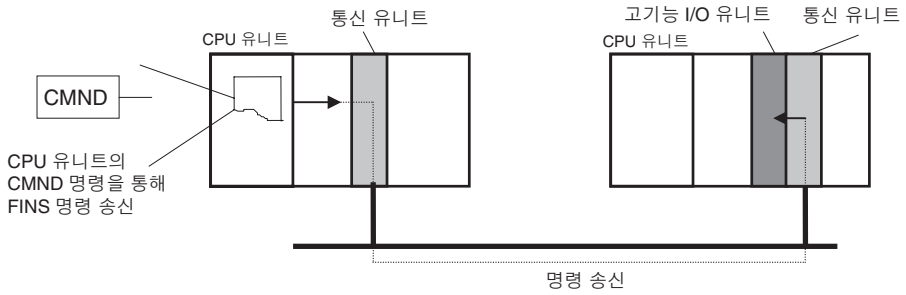


● FINS 명령 수신

CPU 유니트의 래더 프로그램의 CMND 명령을 사용하여 고기능 I/O 유니트에 대한 FINS 명령을 고기능 I/O 유니트로 송신(전송)할 수 있습니다.



(주) FINS 명령은 자체 PLC 본체(CPU 유니트)에서 고기능 I/O 유니트로 전송하는 것뿐만 아니라, 네트워크상의 PLC 본체(CPU 유니트)에서 고기능 I/O 유니트로 전송하는 것도 가능합니다.



● 고기능 I/O 유니트의 기동

고기능 I/O 유니트는 전원 ON 또는 고기능 I/O 유니트 재시작 플래그(A502.00~507.15)가 기동되어야만 사용할 수 있게 됩니다. 전원 ON 또는 재시작에 의해 고기능 I/O 유니트에 대한 초기 처리가 실행됩니다. 그 동안, 고기능 I/O 유니트 초기화 중 플래그(A330.00~335.15)가 ON 상태를 유지합니다. 고기능 I/O 유니트 초기화 중 플래그가 ON 상태일 때는 고기능 I/O 유니트와의 I/O 리프레시(주기적 리프레시 및 IORF 명령 또는 FIORF 명령을 통한 리프레시)가 실행되지 않습니다.

● 고기능 I/O 유니트의 주기적인 리프레시 유무 지정

고기능 I/O 유니트는 고기능 I/O 유니트 릴레이 영역(2000~2959CH)에 유니트당 10CH로 유니트 전면의 유니트 번호에 따라 할당됩니다. 이 할당 영역의 CPU 유니트에 대한 리프레시는 END 명령 직후의 I/O 리프레시(주기적인 리프레시) 시에 실행됩니다.
그러나 고기능 I/O 유니트를 여러 개 사용하면 I/O 리프레시 시간이 연결 대수만큼 가산되므로 사이클 시간이 길어집니다. 이를 줄이기 위해 PLC 시스템 설정을 통해, 지정한 고기능 I/O 유니트와의 END 명령 직후 I/O 리프레시(주기적인 리프레시)를 중지(금지)할 수 있습니다.

또한 고기능 I/O 유니트는 리프레시 간격이 너무 짧으면 자체 내부 처리가 따라가지 못하여 고기능 I/O 유니트 이상 플래그(A402.06)가 ON이 되거나 고기능 I/O 유니트가 정상적으로 동작하지 않는 경우가 있습니다.
이런 경우, 사이클 시간 일정화 기능으로 사이클 시간을 연장하거나 이 PLC 시스템 설정을 통해 END 명령 직후 고기능 I/O 유니트와의 I/O 리프레시를 금지합니다.

END 리프레시를 하지 않는 경우에는 래더 프로그램의 IORF(I/O 리프레시) 명령 또는 FIORF(고기능 I/O 유니트 매회 리프레시) 명령을 사용하여 고기능 I/O 유니트와의 리프레시를 수행합니다.

사용상의 주의 사항

인터럽트 태스크 중에 고기능 I/O 유니트에 대해 IORF/FIORF/IORD/IOWR 명령을 사용할 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(ON: 주기적 리프레시 안 함)로 설정하십시오.
고기능 I/O 유니트와 주기적으로 리프레시(OFF: 주기적 리프레시 수행)하도록 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/FIORF 명령을 통해 동일한 고기능 I/O 유니트에 대해 I/O 리프레시를 실행했을 때
- IORD/IOWR 명령을 통해 동일한 고기능 I/O 유니트의 메모리 영역에 대해 읽기 또는 쓰기를 실행했을 때

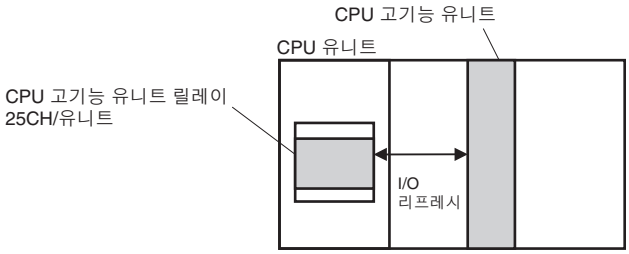
고기능 I/O 유니트에 대한 주기적인 리프레시를 금지하는 경우, 운전 시작 이후 11초 이하의 간격으로 해당 유니트에 대한 IORF 명령 또는 FIORF 명령이 실행되도록 하십시오.
주기적인 리프레시를 금지한 고기능 I/O 유니트에 대해 11초 이상 IORF 명령 또는 FIORF 명령이 실행되지 않는 경우에는 고기능 I/O 유니트에서 CPU 유니트 감시 이상이 발생합니다(ERH LED 또는 RUN LED 점등).

CPU 고기능 유니트의 경우

● I/O 리프레시

CPU 고기능 유니트 릴레이 영역을 I/O 리프레시하면 사이클마다 데이터를 교환합니다. 1유니트당 25CH가 유니트 번호에 따라 할당됩니다.

CPU 고기능 유니트 릴레이 영역: 1500~1899CH(25CH×16유니트)



(주) CPU 유니트는 래더 프로그램에서 DLNK 명령(CPU 고기능 유니트 매회 I/O 리프레시 명령)을 실행하여, 지정된 유니트 번호의 CPU 고기능 유니트와 이 릴레이 영역을 매회 리프레시할 수 있습니다.

● 할당 DM 영역의 전송

1유니트당 100CH가 유니트 번호에 따라 할당됩니다.
(주) 단, CPU 고기능 유니트의 기종에 따라 이 영역을 사용하지 못할 수 있습니다.

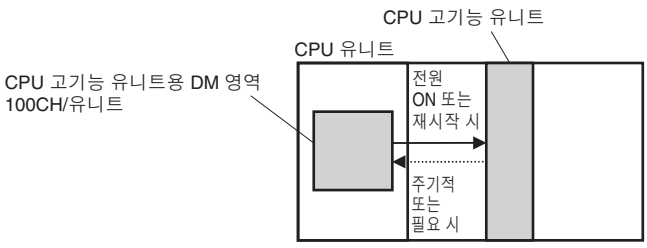
CPU 고기능 유니트용 DM 영역: D30000~D31599(100CH×16유니트)
데이터 전송 타이밍은 다음과 같으며 기종에 따라 다릅니다.

- 전원 ON 시 또는 재시작 시
- 주기적 데이터 교환
- 필요 시 데이터 교환

(주) CPU 유니트는 래더 프로그램에서 DLNK 명령(CPU 고기능 유니트 매회 I/O 리프레시 명령)을 실행하여, 지정된 유니트 번호의 CPU 고기능 유니트와 이 DM 영역을 매회 리프레시할 수 있습니다.

전송 방향에 대해서도 CPU 유니트→CPU 고기능 유니트의 방향 이외에, 기종에 따라 CPU 고기능 유니트→CPU 유니트의 방향도 가능합니다.
전송 타이밍 및 방향에 대해서는 각 CPU 고기능 유니트의 사용 설명서를 참조하십시오.

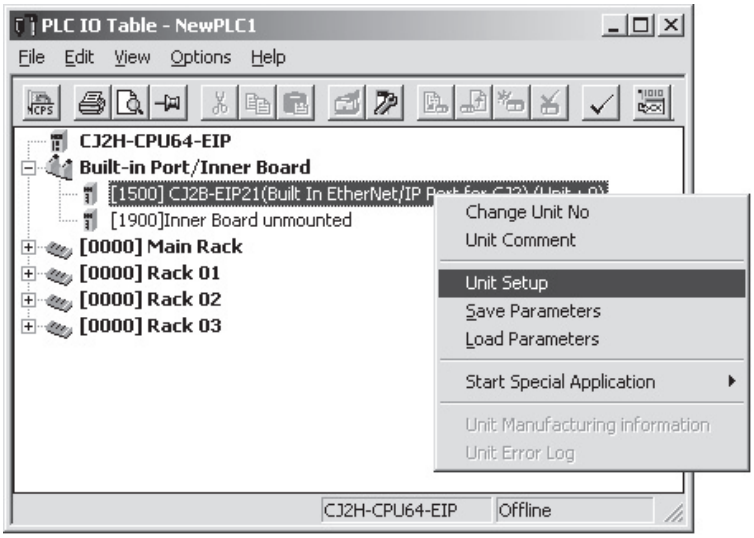
이 DM 영역으로의 전송은 CPU 고기능 유니트의 초기 설정 등에 사용됩니다. 단계 전환 등을 위해 프로그램의 명령을 통해 이 영역을 변경한 경우, CPU 고기능 유니트 재시작 플래그(A501.00~501.15)를 OFF→ON으로 기동해야 합니다.



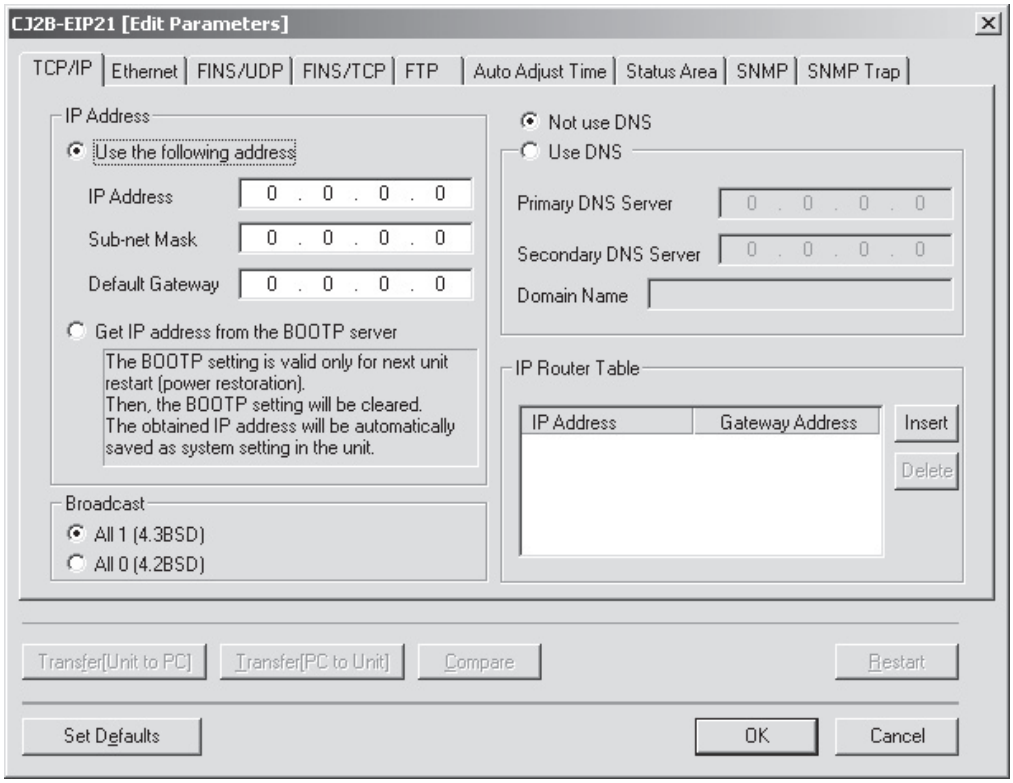
참고

CJ2H-CPU6□-EIP, CJ2M-CPU3□의 내장 EtherNet/IP 포트는 로터리 스위치 설정에 따라 CPU 고기능 유니트로 할당됩니다. 통신 파라미터는 다음과 같이 설정하십시오

(1) PLC의 I/O 테이블에서 내장 포트/내부 보드에서 CJ2B-EIP21(CJ2H용 내장 EtherNet/IP 포트) 또는 CJ2M-EIP21(CJ2M용 내장 EtherNet/IP 포트)을 선택합니다. 마우스 오른쪽 버튼으로 클릭하여 [유니트 설정]을 선택합니다.



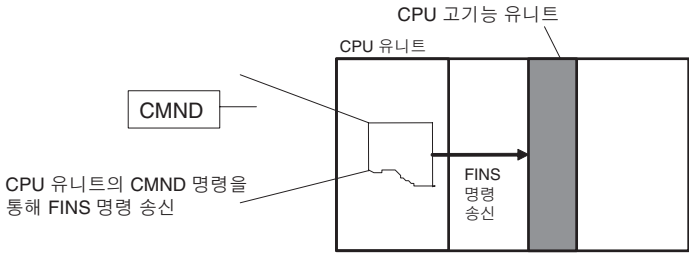
(2) 파라미터를 편집하여 각 파라미터를 설정합니다.



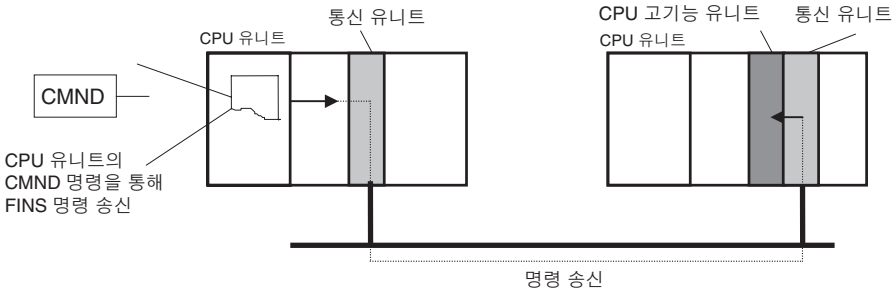
내장 EtherNet/IP 포트의 파라미터에 대한 자세한 내용은 "EtherNet/IP 유니트 사용 설명서" (SBCD-342)를 참조하십시오.

● FINS 명령 수신

CPU 유니트의 래더 프로그램의 CMND 명령을 사용하여 CPU 고기능 유니트에 대한 FINS 명령을 송신(전송)할 수 있습니다.



(주) FINS 명령은 자체 PLC 본체(CPU 유니트)에서 CPU 고기능 유니트로 전송하는 것뿐만 아니라, 네트워크상의 PLC 본체(CPU 유니트)에서 CPU 고기능 유니트로 전송하는 것도 가능합니다.



● CPU 고기능 유니트의 기동

CPU 고기능 유니트는 전원 ON 또는 CPU 고기능 유니트 재시작 플래그(A501.00~501.15)가 기동되어야만 사용할 수 있게 됩니다. 전원 ON 또는 재시작에 의해 CPU 고기능 유니트에 대한 초기 처리가 실행됩니다. 그 동안, CPU 고기능 유니트 초기화 중 플래그(A302.00~A302.15)가 ON 상태를 유지합니다.

CPU 고기능 유니트 초기화 중 플래그가 ON 상태일 때는 CPU 고기능 유니트와의 I/O 리프레시(주기적 리프레시)가 실행되지 않습니다.

9

PLC 시스템 설정

이 장에서는 CPU 유니트의 기본 설정을 수행하는 PLC 시스템 설정에 대한 자세한 내용을 설명합니다.

9-1 PLC 시스템 설정의 개요 9-2

9-2 PLC 시스템 설정 목록 9-3

9-3 PLC 시스템 설정의 설명 9-4

9-3-1 전원 ON 시 동작 설정 9-4
9-3-2 CPU 유니트 설정 9-7
9-3-3 타이머/인터럽트/동기 제어 9-12
9-3-4 고기능 I/O 리프레시 9-17
9-3-5 유니트 설정 9-19
9-3-6 시리얼 포트 9-21
9-3-7 주변 서비스 9-27
9-3-8 FINS 금지 9-28

9-1 PLC 시스템 설정의 개요

PLC 시스템 설정이란 CPU 유니트의 기본 사양을 소프트웨어적으로 정의할 수 있는 기능입니다. CPU 유니트의 각종 설정은 이 PLC 시스템 설정을 통해 수행합니다.

다음과 같은 경우 주변 툴에서 PLC 시스템 설정을 수행합니다.

이런 경우(기본값 이외의 값으로 사용하는 경우)	설정 항목
CJ 시리즈 기본 I/O 유니트에서 • 채터링 또는 노이즈의 영향이 있을 때 • 사이클 시간보다 긴 조건에서 짧은 펄스 입력을 수신하고 싶을 때	기본 I/O 유니트의 입력 응답 시간 설정
전원 OFF→ON 시에도 I/O 메모리의 비유지 영역(CIO, 내부 보조 릴레이, 타이머 현재값/플래그, 태스크 플래그, 인덱스 레지스터, 데이터 레지스터)을 유지하고 싶을 때	'IOM(I/O 메모리) 유지 비트'의 전원 ON 시 유지/비유지 설정
전원 OFF→ON 시에도 주변 툴에서 설정한 접점의 강제 세트/리셋 상태를 유지하고 싶을 때	'강제 세트/리셋 유지 플래그'의 전원 ON 시 유지/비유지 설정
• 프로그램을 디버그할 때 • 전원 ON 시의 동작 모드를 '프로그램' 모드 또는 '모니터' 모드로 변경하고 싶을 때	전원 ON 시 동작 모드 지정 방법
배터리 없이 사용할 때(CPU 유니트의 배터리 이상을 검출하고 싶지 않을 때)	배터리 이상 검출 유무 설정
다중 리프레시 이상을 검출하고 싶지 않을 때	다중 리프레시 이상 검출 유무 설정
시리얼 포트를 CX-Programmer(툴 버스)의 통신 속도 자동 인식 이외의 설정에서 사용할 때와 더불어 상위 링크 모드의 9600비트/s 등과 같은 기본 통신 속도 이외에서 사용할 때(*1)	시리얼 포트 설정
NT 링크를 통한 PT와의 통신 속도를 고속화하고 싶을 때	시리얼 포트 설정의 통신 속도를 고속 NT 링크에 설정
예약 인터럽트의 태스크 설정 단위 시간을 단축하고 싶을 때(10ms→1.0ms, 0.1ms)	예약 인터럽트 단위 시간 설정
I/O 인터럽트 태스크, 예약 인터럽트 태스크에 의한 인터럽트 처리를 고속으로 기동하고 싶을 때	고속 인터럽트 기능 사용 유무 지정
유니트 간 동기 제어 기능을 사용하고 싶을 때	유니트 간 동기 설정
프로그램 디버그 중에 명령 에러 발생 여부를 파악하고 싶을 때	명령 에러 발생 시 동작 설정
I/O 리프레시 간격 또는 사이클 시간을 일정하게 유지하고 싶을 때	고정 사이클 시간
모니터 사이클 시간을 1초 이외(10~40000ms)로 설정하고 싶을 때	모니터 사이클 시간
전원 OFF 시에 특정 처리를 수행하고 싶을 때	전원 OFF 인터럽트 태스크 유효/무효 설정(*2)
순간 정전 시간을 연장하고 싶을 때	전원 OFF 감지 지연 시간 설정(*1)
• 인터럽트 태스크 내에서 IORF 명령 또는 FIORF 명령을 실행할 때 • 고기능 I/O 유니트를 다수 사용한 경우, 평균적인 사이클 시간을 단축하고 싶을 때 • 고기능 I/O 유니트에 대한 I/O 리프레시 간격을 연장하고 싶을 때	고기능 I/O 유니트의 주기적인 리프레시 유무 지정
FAL 명령 또는 FPD 명령 실행에 따른 사용자 정의 이상을 이상 내역에 등록하고 싶지 않을 때	FAL 이상에 대한 이상 내역 등록 유무 지정
데이터 처리 시의 사이클 시간 편차를 방지하고 싶을 때	테이블 데이터 처리 명령/문자열 처리 명령/데이터 이동 명령의 백그라운드 처리 유무
• 전원을 켠 후 기동에 시간이 걸리는 유니트가 있을 때 • 대기하지 않고 CPU 유니트의 운전을 시작하고 싶을 때	유니트 기동 처리 미완료 시 운전 시작 유무

*1 CPU 유니트의 전면 딥 스위치 SW5를 OFF(PLC 시스템 설정)로 설정해야 함

*2 CJ1W-PD022를 장착한 경우는 사용 불가

● 관련 특수 보조 릴레이

명칭	주소	설명	읽기/쓰기
PLC 시스템 설정 이상 플래그(운전 계속 이상)	A402.10	PLC 시스템 설정에 설정값 이상이 발생한 경우 1(ON)이 됩니다.	읽기 전용

9-2 PLC 시스템 설정 목록

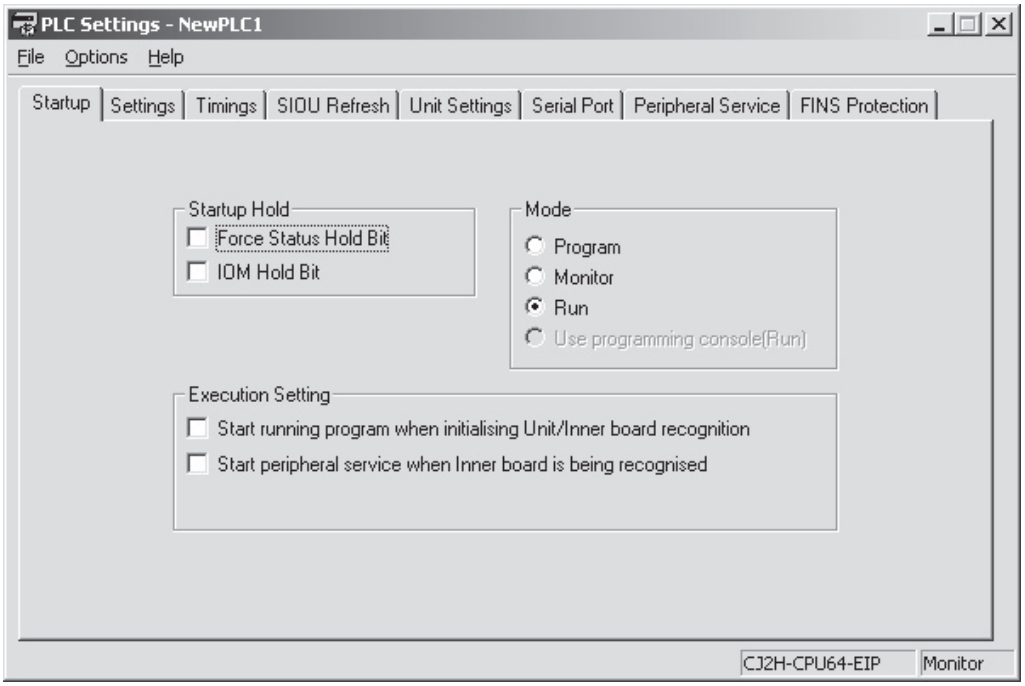
기본값(초기값)은 다음과 같은 설정 상태가 됩니다.
기본값의 설정을 변경하고 싶을 때는 CX-Programmer에서 PLC 시스템 설정을 변경하여 CPU 유니트로 전송하십시오.

CX-Programmer PLC 시스템 설정 탭	설정 항목		호환 기종 CJ2H CPU 유니트	호환 기종 CJ2M CPU 유니트	기본값	참조 페이지
전원 ON 시	전원 ON 시 유지 설정	강제 상태 유지 플래그	○	○	전원 ON 시 유지 안 함	9-4
		IOM(I/O 메모리) 유지 비트	○	○	전원 ON 시 유지 안 함	9-5
	동작 모드		○	○	'운전' 모드	9-6
	운전 시작 조건 설정	특정 유니트의 기동을 기다리지 않고 CPU 운전 시작	○	○	운전 시작 안 함	9-6
CPU 유니트 설정	CPU 실행 처리	배터리 이상 검출 설정	○	○	검출	9-7
		다중 리프레시 이상 검출 설정	○	○	검출	9-9
		명령 에러 발생 시 동작 설정	○	○	정지 안 함	9-9
		FAL 이상에 대한 이상 내역 등록 유무 지정	○	○	이상 내역에 등록	9-9
	백그라운드 처리		○	○	백그라운드 처리 안 함	9-10
	FB 내 통신 명령 설정	재송신 횟수	○	○	0회	9-11
		응답 감시 시간 설정	○	○	2s	9-11
타이머/인터럽트/동기 제어	모니터 사이클 시간		○	○	1000ms(1초)	9-12
	고정 사이클 시간 설정		○	○	일정하지 않음: 가변	9-13
	예약 인터럽트 단위 시간 설정		○	○	10ms	9-14
	전원 OFF 감지 지연 시간 설정		○	○	0ms	9-14
	전원 OFF 인터럽트 태스크 무효 설정		○	○	무효	9-15
	고속 인터럽트 기능 사용		×	○	고속 인터럽트 기능 사용 안 함	9-16
	동기 제어 사용		×	○	사용 안 함	9-16
고기능 I/O 리프레시	고기능 I/O 리프레시 금지 설정		○	○	금지 설정 없음	9-17
유니트 설정	각 기본 I/O 유니트 입력 응답 시간 설정		○	○	8ms	9-19
시리얼 포트	모드 (CPU 유니트의 딥 스위치 SW5가 OFF(기본값)인 경우만 해당)		○ (*)	○	상위 링크	9-21
주변 서비스	CPU 실행 처리 모드 설정		○	○	통상	9-27
	주변 서비스 일률 시간		○	○	사이클 시간의 10%	9-27
FINS 금지	네트워크를 경유하는 FINS 쓰기 금지 설정		○	○	FINS 쓰기 금지 기능을 유효화 안 함	9-28

* 시리얼 PLC 링크 선택 불가

9-3 PLC 시스템 설정의 설명

9-3-1 전원 ON 시 동작 설정



전원 ON 시 유지 설정

- **'강제 상태 유지 플래그'**

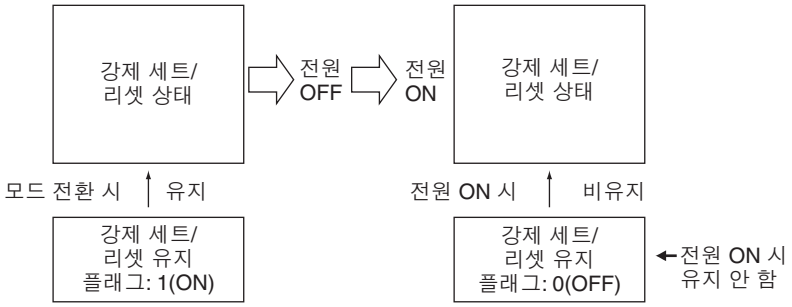
  특수 보조 릴레이의 '강제 세트/리셋 유지 플래그'(A500.13)의 전원 ON 시 유지/비유지 설정입니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
강제 상태 유지 플래그	OFF: 전원 ON 시 유지 안 함 ON: 전원 ON 시 유지	OFF (유지 안 함)	특수 보조 릴레이의 '강제 세트/리셋 유지 플래그'(A500.13)를 전원 ON 시 유지할지 여부를 설정합니다.	A500.13 (강제 세트/리셋 유지 플래그)

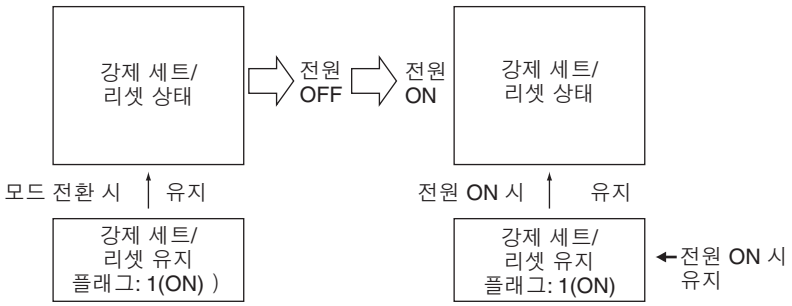
'강제 세트/리셋 유지 플래그'와 이 설정의 관계

특수 보조 릴레이의 '강제 세트/리셋 유지 플래그'(A500.13)는 CPU 유니트의 모드 전환(모니터←→프로그램) 시, 강제 세트/리셋 전환 전의 상태를 유지하는 플래그입니다.
그러나 이 플래그는 전원을 켰을 때 클리어되어 작동하지 않습니다.
따라서 전원을 켰을 때도 강제 세트/리셋 상태를 유지하려면 전원 ON 시 이 플래그를 유지하도록 해야 합니다.
이 경우, 이 설정(ON: 전원 ON 시 유지)을 사용하여 '강제 세트/리셋 유지 플래그' 설정 내용을 유지하십시오.
배터리 전압 저하 등으로 인해 메모리가 유지되지 않을 때는 이 설정에 관계 없이 강제 세트/리셋 데이터 유지 플래그가 클리어됩니다.

강제 상태 유지 플래그 OFF: '전원 ON 시 유지 안 함'일 때



강제 상태 유지 플래그 ON: '전원 ON 시 유지'일 때



- **'IOM(I/O 메모리) 유지 비트'**

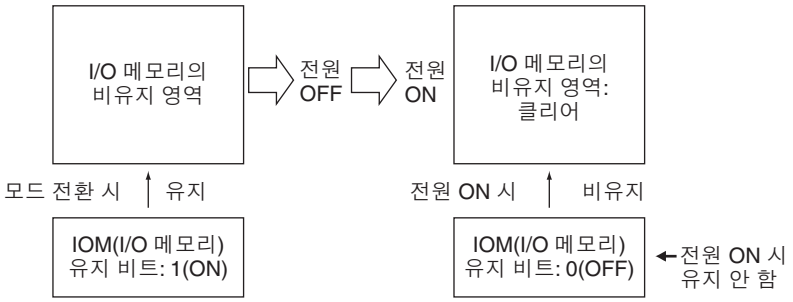
  특수 보조 릴레이의 'IOM(I/O 메모리) 유지 비트'(A500.12)의 전원 ON 시 유지/비유지 설정입니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
I/O 메모리 유지 플래그	OFF: 전원 ON 시 (유지 안 함) ON: 전원 ON 시 유지	OFF (유지 안 함)	특수 보조 릴레이의 'IOM(I/O 메모리) 유지 비트'(A500.12)를 전원 ON 시 유지할지 여부를 설정합니다. 전원을 켰을 때도 I/O 메모리의 비유지 영역을 유지할 경우, 이 설정을 ON(전원 ON 시 유지)으로 설정하고 특수 보조 릴레이의 'IOM(I/O 메모리) 유지 비트'(A500.12)를 1(ON)로 설정합니다.	A500.12 (I/O 메모리 유지 플래그

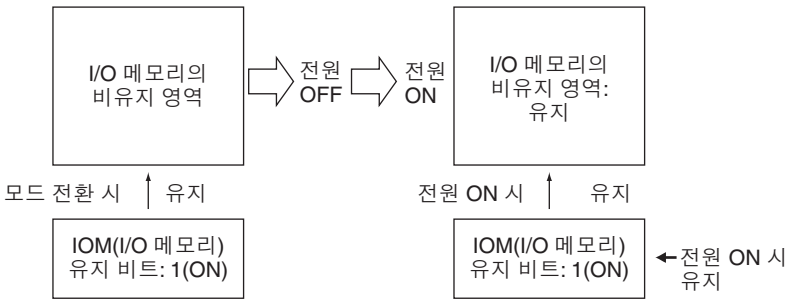
'IOM(I/O 메모리) 유지 비트'와 이 설정의 관계

특수 보조 릴레이의 'IOM(I/O 메모리) 유지 비트'(A500.12)는 CPU 유니트의 모드 전환(운전/모니터←→프로그램) 시, I/O 메모리 전환 전의 내용을 유지하는 플래그입니다. 그러나 이 플래그는 전원을 켰을 때 클리어되어 작동하지 않습니다. 따라서 전원을 켰을 때도 I/O 메모리의 내용을 유지하기 위해서는 전원 ON 시 이 플래그를 유지하도록 해야 합니다. 따라서 이 설정(ON: 전원 ON 시 유지)을 통해 'IOM(I/O 메모리) 유지 비트'의 설정 내용을 유지하십시오.
또한 'IOM(I/O 메모리) 유지 비트' 자체도 1(ON)로 설정해야 합니다.
배터리 전압 저하 등으로 인해 메모리가 유지되지 않을 때는 이 설정에 관계 없이 'IOM(I/O 메모리) 유지 비트'가 클리어됩니다.

IOM(I/O 메모리) 유지 비트 OFF: '전원 ON 시 유지 안 함'일 때



IOM(I/O 메모리) 유지 비트 ON: '전원 ON 시 유지'일 때



동작 모드

● '프로그램/모니터/운전'

전원 ON 시의 동작 모드를 설정합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
동작 모드	• 프로그램: '프로그램' 모드 • 모니터: '모니터' 모드 • 운전: '운전' 모드	'운전' 모드	전원 ON 시의 CPU 유니트의 동작 모드를 설정합니다.	—

운전 시작 조건

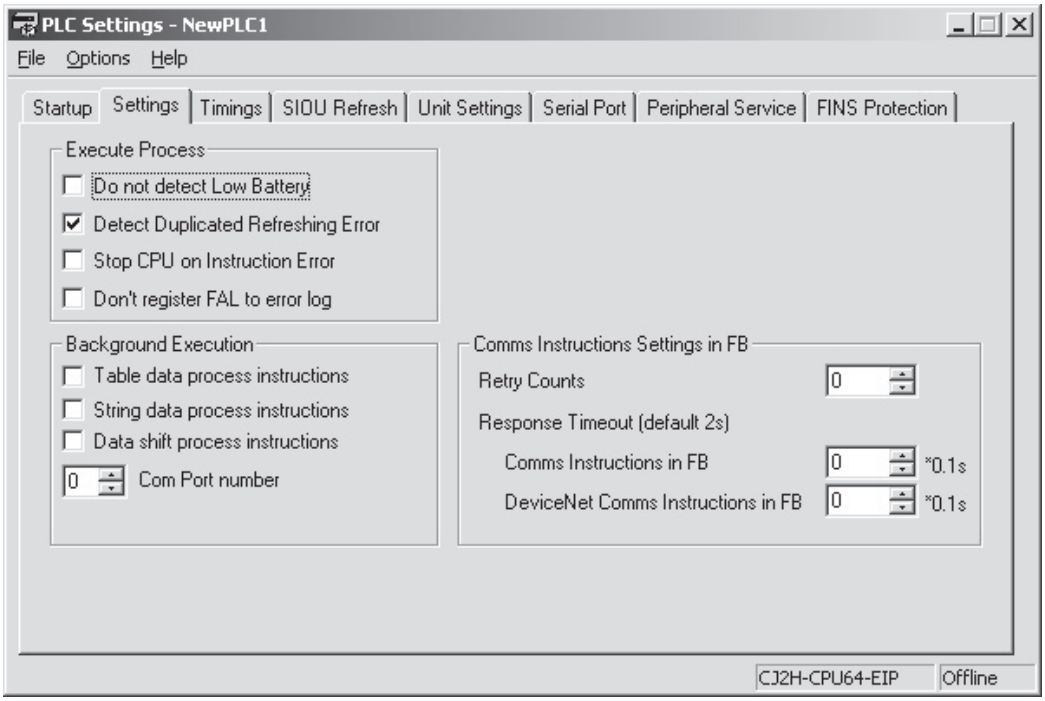
● '특정 유니트/INNER 보드의 기동을 기다리지 않고 CPU 운전 시작'

전원을 켰을 때 특정 유니트의 기동 처리를 기다릴지 여부를 설정합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
운전 시작 조건	OFF: 무효(운전 시작 안 함) ON: 유효(운전 시작)	OFF (운전 시작 안 함)	전원을 켠 후 '운전' 모드 또는 '모니터' 모드일 때, 특정 유니트(*)의 기동 처리가 완료되지 않아도 기다리지 않고 CPU 유니트의 운전을 시작하고 싶은 경우 ON(유효)으로 설정합니다. 평상시와 같이 모든 유니트의 기동 처리가 완료될 때까지 CPU 유니트의 운전을 시작하지 않는 경우는 OFF(무효)로 설정합니다.	—

* 운전 시작 조건 설정을 '유효'로 한 경우, 모든 유니트의 기동 처리가 완료되기를 기다리지 않고 CPU가 운전을 시작하는 것은 아닙니다. 특정 유니트의 기동 처리가 완료되지 않았을 때만 시작하지 않습니다.

9-3-2 CPU 유니트 설정



CPU 실행 처리

- **'배터리 이상을 검출하지 않음(배터리 없이 운전)'**

  배터리 이상을 검출할지 여부를 설정합니다(초기값: 검출).
  배터리 없이 운전을 하는 경우 '검출 안 함'으로 설정하십시오.
  (자세한 내용은 "CJ 시리즈 CJ2 CPU 유니트 사용 설명서 하드웨어편"(SBCA-349)을 참조하십시오.)

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
배터리 이상 검출 안 함 (배터리 없이 운전)	OFF: 검출 ON: 검출 안 함	OFF (검출)	CPU 유니트의 배터리 이상을 검출할지 여부를 지정합니다. OFF(검출)로 설정한 경우, 이상 검출 시 CPU 유니트 전면 ERR/ALM LED가 점멸하고 특수 보조 릴레이의 '배터리 이상'(A402.04)이 1(ON)로 됩니다. 이때, CPU 유니트의 운전은 계속됩니다.	A402.04 (배터리 이상 플래그)

참고

배터리 없이 운전

• 배터리 없이 운전할 때의 동작

PLC가 배터리 없이 운전하는 경우 유지할 수 있는 데이터는 다음과 같습니다.

- 사용자 프로그램 및 네트워크 변수
- 파라미터(PLC 시스템 설정, 등록 I/O 테이블, 라우팅 테이블, CPU 고기능 유니트 설정, PLC 이름)
- 고기능 유니트 내 비휘발성 메모리에 저장된 데이터 (예: SCU 유니트의 프로토콜 매크로 데이터)

다음 데이터는 유지되지 않습니다. 따라서 값이 불안정하므로 주의하십시오.

- I/O 메모리(HR/DM/EM/AR 포함)
- CPU 유니트 내장 시계

• 배터리 없는 운전 설정 방법

- PLC 시스템 설정
  배터리 없이 운전 시, PLC 시스템 설정을 다음과 같이 설정하여 사용하십시오.
  ① 배터리가 없으면 I/O 메모리의 상태가 불안정해지므로 'IOM(I/O 메모리) 유지 비트' 설정은 "전원 ON 시 유지 안 함"으로 설정하십시오.
  ② 배터리가 없으면 강제 세트/리셋 상태가 불안정해지므로 '강제 세트/리셋 유지 플래그' 설정은 "전원 ON 시 유지 안 함"으로 설정하십시오.
  ③ '배터리 이상 검출 유무 설정'은 '검출 안 함'으로 설정하십시오.
- 프로그램
  배터리가 없으면 부하 차단 플래그(A500.15)의 동작이 불안정해지므로, ON하지 않도록 다음과 같은 프로그램을 삽입하십시오.

```
LD P_Off
OUT A500.15
```

• 배터리 없이 운전 시 주의 사항

배터리 없이 운전을 할 때는 다음 사항에 주의하십시오.

- 데이터의 초기화
  DM/EM을 비롯한 I/O 메모리가 불안정한 값이 됩니다.
  반드시 프로그램에서 초기값을 설정하십시오.

```
(예)
 LD A200.11
 MOV #0918 D0// 운전 시작 시 D0 값 초기화:
```

• 시계가 불안정해짐

CPU 유니트의 내장 시계가 동작하지 않고 불안정한 값이 됩니다.
따라서 이상 내역 등에 기록된 날짜 정보가 정확하게 표시되지 않습니다.
또한 메모리 카드 등에 파일을 작성했을 때 해당 파일 작성 날짜도 정확하지 않게 됩니다.

사용상의 주의 사항

CPU 유니트 내의 I/O 메모리 영역(DM/EM/HR/AR 포함)은 비휘발성 메모리(내장 플래시 메모리)에 기록되지 않습니다. 전원 OFF 복구 시의 유지 영역인 DM/EM/HR/AR 영역은 배터리에 의해 유지됩니다. 따라서 배터리에 이상이 발생한 경우 정확한 값을 유지하지 못할 가능성이 있습니다. 그러므로 필요에 따라 배터리 이상 플래그(A402.04)가 1(ON)일 때는 래더 프로그램에서 DM/EM/HR/AR 영역의 내용을 재설정하는 등의 대책을 세우십시오.

● '다중 리프레시 이상 검출'(다중 리프레시 이상 검출 유무 설정)

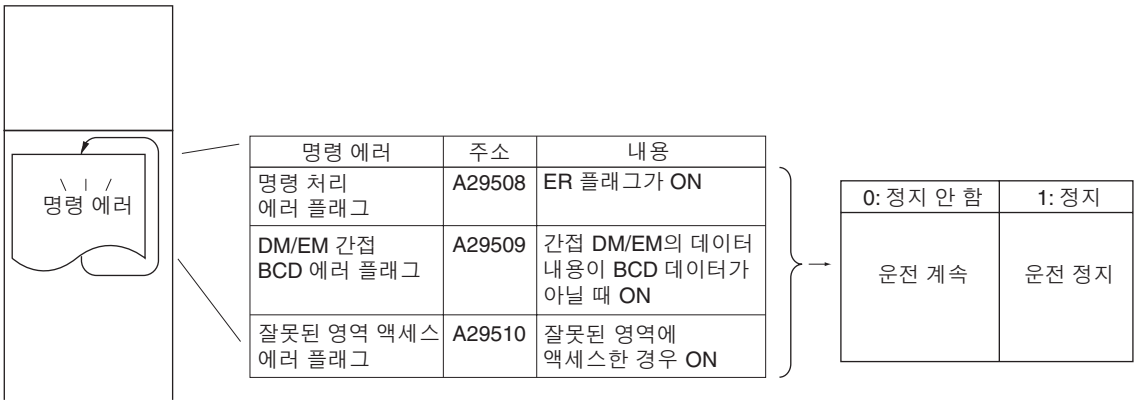
다중 리프레시 이상을 검출할지 여부를 설정합니다(초기값: 검출).
초기값인 'ON(검출)'으로 설정하면 다음과 같은 경우에 다중 리프레시 이상이 검출됩니다.

- 고기능 I/O 유니트와의 I/O 리프레시 수행 시, 인터럽트 태스크 내에서 동일 유니트에 대해 IORF 명령 또는 FIORF 명령을 실행한 경우

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
다중 리프레시 이상 검출	OFF: 검출 안 함 ON: 검출	ON (검출)	다중 리프레시 이상을 검출할지 여부를 지정합니다. ON(검출)으로 설정한 경우, 이상을 검출하면 CPU 유니트 전면 ERR/ALM LED가 점멸하고 특수 보조 릴레이의 '다중 리프레시 이상 플래그'(A402.13)가 1(ON)이 됩니다. 이때, CPU 유니트의 운전은 계속됩니다.	A402.13 (다중 리프레시 이상 플래그)

● '명령 에러 발생 시 실행 정지'

명령 에러 발생 시 운전 정지 여부를 설정합니다(초기값: 정지 안 함).
프로그램을 디버그할 때 사용합니다.
'정지(ON)'로 설정하면 다음 3가지 플래그 중 하나가 ON일 때 명령 에러가 되어 프로그램 에러를 발생시킵니다.



설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이(*)
명령 에러 발생 시 실행 정지	OFF: 정지 안 함 ON: 정지	OFF (정지 안 함)	명령 실행 시 명령 에러(명령 처리 에러 또는 잘못된 영역 액세스 에러)가 발생했을 때 그것을 '프로그램 에러'로 취급하여 CPU 유니트의 운전 정지 여부를 설정합니다.	A295.08(명령 처리 에러 플래그), A295.09(DM/EM 간접 BCD 에러 플래그) A295.10(잘못된 영역 액세스 플래그)

* 이 설정을 OFF(정지 안 함)로 설정한 경우는 명령 에러가 발생해도 ON으로 되지 않습니다.

● '이상 내역에 FAL 등록 안 함'

사용자 정의 FAL 이상이 발생했을 때 이상 내역에 등록할지 여부를 설정합니다(초기값: 등록).

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
이상 내역에 FAL 등록 안 함	OFF: 이상 내역에 등록 ON: 이상 내역에 등록 안 함	OFF (등록)	FAL 명령을 실행했을 때 또는 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 때 설정합니다.	—

백그라운드 처리

다음 명령은 데이터 양에 따라 처리 시간이 길어져 사이클 시간에 큰 영향을 줍니다. 이 설정을 사용하여 명령의 처리를 분할하고 사이클 시간에 대한 영향을 줄입니다. 자세한 내용은 '10-2-5 백그라운드 처리 실행 기능'을 참조하십시오.

● '테이블 데이터 처리 명령'

테이블 데이터 처리 명령에서 백그라운드 처리 여부를 설정합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
테이블 데이터 처리 명령	OFF: 백그라운드 처리 안 함 ON: 백그라운드 처리	OFF (백그라운드 처리 안 함)	테이블 데이터 처리 명령에서 해당 명령 실행을 여러 사이클에 걸쳐 시간 분할 처리(백그라운드 처리)할 때 설정합니다.	—

● '문자열 처리 명령'의 백그라운드 처리 유무

문자열 처리 명령에서 백그라운드 처리 여부를 설정합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
문자열 처리 명령	OFF: 백그라운드 처리 안 함 ON: 백그라운드 처리	OFF (백그라운드 처리 안 함)	문자열 처리 명령에서 해당 명령 실행을 여러 사이클에 걸쳐 시간 분할 처리(백그라운드 처리)할 때 설정합니다.	—

● '데이터 이동 명령'의 백그라운드 처리 유무

데이터 이동 명령에서 백그라운드 처리 여부를 설정합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
데이터 이동 명령	OFF: 백그라운드 처리 안 함 ON: 백그라운드 처리	OFF (백그라운드 처리 안 함)	데이터 이동 명령에서 해당 명령 실행을 여러 사이클에 걸쳐 시간 분할 처리(백그라운드 처리)할 때 설정합니다	—

● '통신 포트 번호'

백그라운드 처리에서 사용하는 통신 포트 번호를 설정합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
통신 포트 번호	0~7: 통신 포트(내부 논리 포트) 번호 0~7	0 (번호 0)	백그라운드 처리를 실행하기 위해 사용하는 통신 포트 번호(내부 논리 포트 번호)를 설정합니다. 각 명령 종류에서 공통입니다.	—



사용상의 주의 사항

PLC 시스템 설정에서 '고속 인터럽트 기능 사용'을 유효화한 경우, 인터럽트 태스크 내에서는 백그라운드 처리를 사용할 수 없습니다.

FB 내 통신 명령 설정(Smart FB 라이브러리 전용 PLC 시스템 설정)

오므론이 제공하는 펑션 블록 라이브러리 'Smart FB 라이브러리'를 사용할 때만 사용하는 PLC 시스템 설정 영역입니다. 그 이외의 용도로는 사용할 수 없습니다.

오므론이 제공하는 'Smart FB 라이브러리' 중에서 특히, FINS 메시지 통신 또는 DeviceNet Explicit 메시지 통신을 실행하는 Smart FB를 사용하는 경우, 이 PLC 시스템 설정 영역에 재시도 횟수, 응답 감시 시간을 설정해야 합니다.
여기서 설정한 값은 위의 관련 특수 보조 릴레이 A580~582CH에 자동으로 저장되고 'Smart FB 라이브러리'는 그 값을 참조하여 동작합니다.

● '재송신 횟수'

FB 내 통신 명령을 실행할 때의 재시도 횟수를 설정합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
재송신 횟수	0~15	0	FB 내에서 FINS 메시지 통신 명령 또는 DeviceNet Explicit 메시지 통신 명령을 실행하는 경우 명령 송신을 재시도하는 횟수를 설정합니다.	A580.00~ A580.03

● '응답 감시 시간' - 'FB 내 통신 명령'

FB 내 통신 명령을 실행할 때의 응답 감시 시간을 설정합니다.

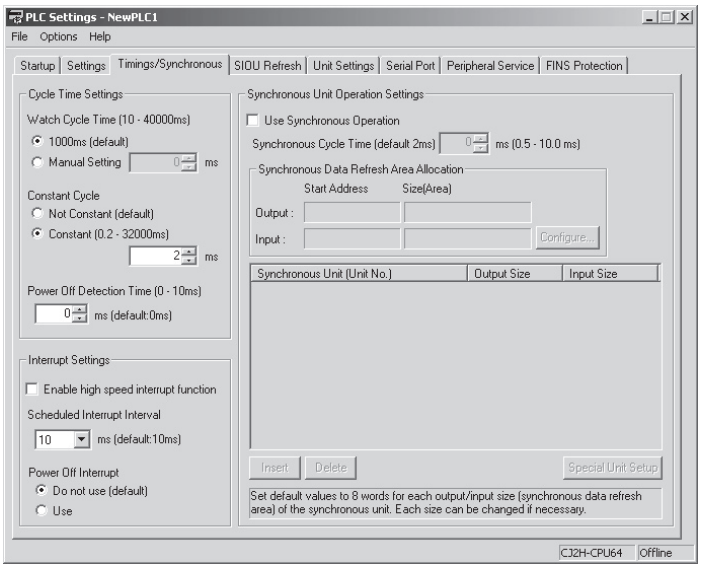
설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
FB 내 통신 명령	1~65535 (단위 0.1초, 0.1~6553.5초) 0: 2초	0 (2초)	FB 내에서 FINS 메시지 통신 명령을 실행한 경우 이 시간을 초과하여 응답이 돌아오지 않으면 응답 시간 초과로 간주합니다.	A581CH

● '응답 감시 시간' - 'FB 내 DeviceNet 통신 명령'

FB 내 DeviceNet 통신 명령을 실행할 때의 응답 감시 시간을 설정합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
FB 내 DeviceNet 통신 명령	1~65535 (단위 0.1초, 0.1~6553.5초) 0: 2초	0 (2초)	FB 내에서 DeviceNet Explicit 메시지 통신 명령을 실행한 경우 이 시간을 초과하여 응답이 돌아오지 않으면 응답 시간 초과로 간주합니다.	A582CH

9-3-3 타이머/인터럽트/동기 제어



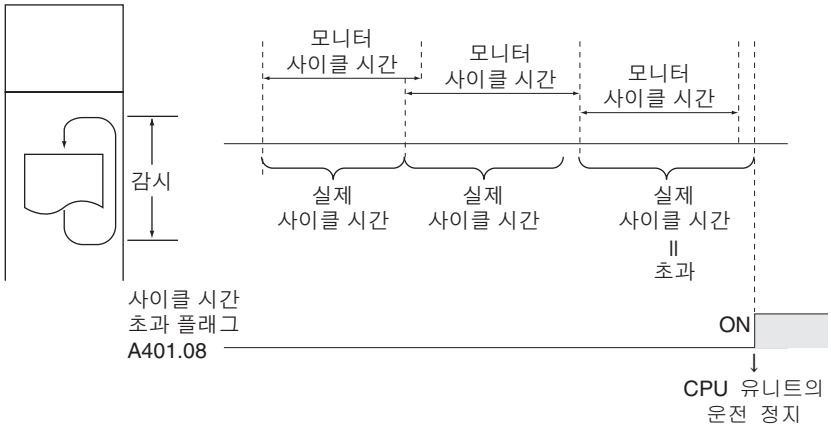
- **'모니터 사이클 시간'**

  모니터 사이클 시간을 초기값(1000ms) 이외로 설정할 때 사용합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
사이클 시간 감시 시간	OFF: 임의 설정 ON: 초기값 (1000ms: 1초)	ON (1000ms: 1초)	모니터 사이클 시간을 초기값인 1초 이외로 설정하는 경우 'OFF: 임의 설정'으로 설정합니다.	A401.08 (사이클 시간 초과 플래그)
	10~40,000ms (10ms 단위)	0	임의 설정 유무를 OFF(임의 설정)로 한 경우에만 설정합니다. 사이클 시간이 설정된 감시 시간을 초과했을 때, 특수 보조 릴레이 A401.08(사이클 시간 초과 플래그)을 1(ON)로 설정합니다.	A264~265CH (사이클 시간 현재값)

모니터 사이클 시간을 변경해야 하는 경우

사이클 시간이 설정된 감시 시간을 초과했을 때, 특수 보조 릴레이 A401.08(사이클 시간 초과 플래그)을 1(ON)로 설정하고 운전을 정지합니다. 따라서 평상시의 사이클 시간이 초기값인 1초를 초과하는 경우는 반드시 이 값을 설정해야 합니다.



주: 모니터 사이클 시간의 기본값은 1초(1000ms)입니다.

● **'고정 사이클 시간'**

사이클 시간 일정화 기능을 사용하는 경우 해당 사이클 시간 일정화 시간을 설정합니다(초기값: 사이클 시간 가변).

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
사이클 시간 일정 시간	OFF: 일정화 ON: 가변	ON (가변)	사이클 시간을 일정하게 하는 경우 OFF로 설정합니다. 일정하게 하는 경우는 시간을 설정해야 합니다.	—
	0.2~32000.0ms (0.1ms 단위)	—	사이클 시간을 일정하게 하는 경우 0.2~32000.0의 값을 설정합니다(단, 실제 사이클 시간이 이 값보다 짧은 경우만 해당). 일정하게 하지 않는 경우에는 0을 그대로 둡니다.	—

참고

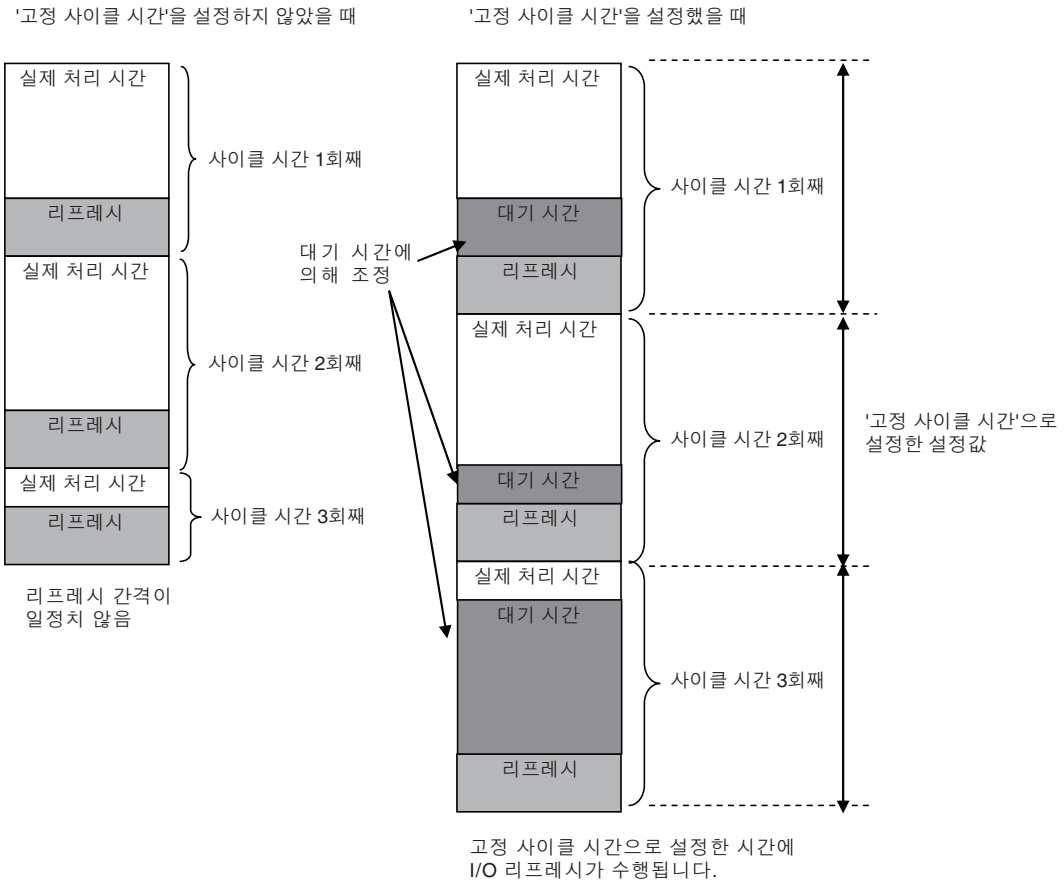
CPU 유니트를 운전하는 중(동작 모드가 '모니터' 모드일 때)에 CX-Programmer의 사이클 모니터 화면에서 고정 사이클 시간을 변경할 수 있습니다(유니트 Ver.1.1 이상).

사용상의 주의 사항

사이클 시간을 길게 설정하면 툴 서비스 간격도 길어지기 때문에 툴의 온라인 조작 응답성이 나빠지거나 온라인으로 연결하기 어려워지는 경우가 있습니다.

사이클 시간 일정화 기능을 유효화하는 조건

사이클 시간을 일정하게 하여 입출력 응답의 편차를 없애고 싶은 경우 이 값을 0000Hex 이외의 값으로 설정합니다. 단, 이 기능이 유효화되는 것은 실제 사이클 시간이 이 값보다 짧은 경우만 해당합니다. 실제 사이클 시간이 이 값보다 길면 실제 사이클 시간이 그대로 사용됩니다.



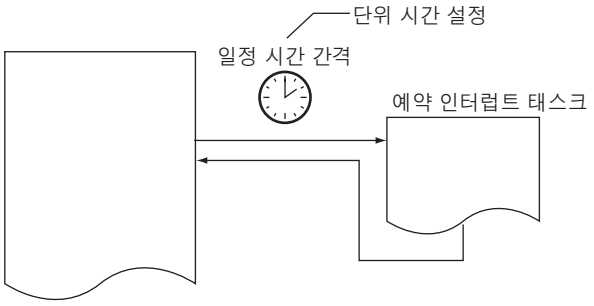
● '예약 인터럽트 단위 시간 설정'

예약 인터럽트 단위 시간을 초기값인 10ms 이외에 1ms 또는 0.1ms로 할 때 설정합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
예약 인터럽트 단위 시간 설정	• 10ms • 1.0ms • 0.1ms	10ms	예약 인터럽트 태스크의 단위 시간을 설정합니다.	—

예약 인터럽트 단위 시간 설정

일정 시간 간격으로 인터럽트 태스크 프로그램을 실행하는 예약 인터럽트인 경우, 단위 시간을 설정합니다.
예약 인터럽트 시간 설정은 MSKS(인터럽트 마스크 세트) 명령을 통해 지정합니다.



● '전원 OFF 감지 지연 시간 설정'

전원 OFF 감지 지연 시간을 초기값인 0ms 이외로 할 때 설정합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
전원 OFF 감지 지연 시간 설정	0~10ms (1ms 단위)	0ms	전원 전압이 최저 정격인 AC 전원은 85% 이하(DC 전원은 80% 이하)가 된 다음 전원 OFF를 검출(AC 전원은 10~25ms, DC 전원은 2~5ms: 비확정)한 후, 실제로 '전원 OFF'를 확정하기까지의 시간('전원 OFF' 확정 연장 시간)을 설정합니다. 기본값은 0ms입니다. '전원 OFF'를 확정하면 전원 OFF 인터럽트 태스크가 유효인 경우 전원 OFF 인터럽트 태스크가 실행됩니다. 무효인 경우는 CPU가 리셋되고 운전을 정지합니다.	—

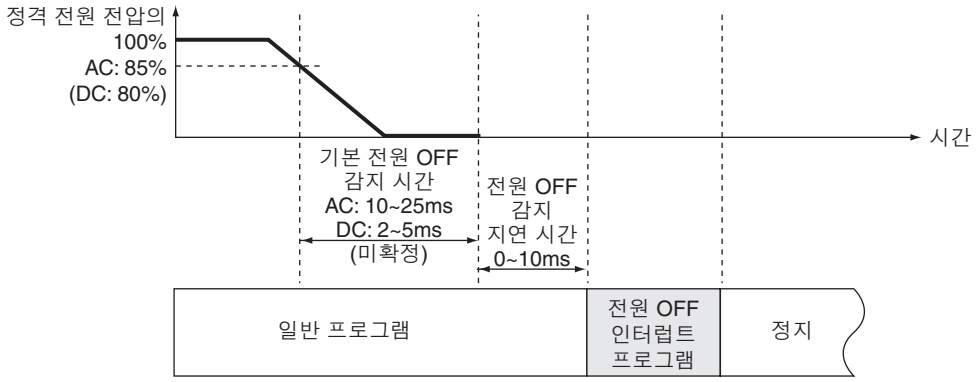
(주) 전원 유니트 CJ1W-PD022를 장착한 경우에는 이 설정을 사용할 수 없습니다.

전원 OFF 감지 지연 시간 설정

전원 OFF가 발생한 후 전원 OFF를 확정하고 통상 프로그램을 정지시킬 때까지의 시간을 초기 전원 OFF 검출 시간 값(*)+몇 ms의 시간으로 할지 여기에서 지정합니다(0~10ms). 전원 ON 시 또는 운전 시작 시, 이 설정이 유효가 됩니다.
전원 사정이 나쁘고 순간 정전에 의한 전원 OFF가 발생할 때는 이 시간을 길게 설정하십시오.
또한 초기값으로 지정된 전원 OFF 검출 시간(*) 후, 내부 5V 전원이 0V가 될 때까지의 시간('전원 유지 시간' = 전원 OFF 검출 시간 연장 설정 + 전원 OFF 확정 처리 시간의 합계)은 최대 10ms입니다.

* 전원 OFF 검출 시간은 AC 전원의 경우 10~25ms 비확정, DC 전원의 경우 PD025는 2~5ms 비확정, PD022는 2~10ms 비확정이 됩니다.

(주) 전원 유니트 CJ1W-PD022를 장착했을 때는 설정할 수 없습니다(🕮 '부록-5 전원 OFF 시 PLC의 동작').



(주) 전원 OFF 인터럽트 태스크 내의 프로그램 실행 시간은
10ms－(전원 OFF 감시 지연 시간) 범위 이내이어야 합니다.

전원 OFF 인터럽트 태스크는 🕮 '5-2-3 인터럽트 태스크'를 참조하고 전원 OFF 시의 동작에 대해서는 🕮 '부록-5 전원 OFF 시 PLC의 동작'을 참조하십시오.

● '전원 OFF 인터럽트 태스크 무효 설정'의 전원 OFF 인터럽트 태스크 유효/무효 설정

전원 OFF 인터럽트 태스크를 유효화할 때 설정합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
전원 OFF 인터럽트 태스크 무효 설정	OFF: 유효 ON: 무효	ON(무효)	전원 OFF 인터럽트 태스크를 OFF(유효)로 설정하는 경우, 전원 OFF가 발생하면 전원 OFF 인터럽트 태스크가 실행됩니다.	—

(주) 전원 유니트 CJ1W-PD022를 장착했을 때는 설정할 수 없습니다(🕮 '부록-5 전원 OFF 시 PLC의 동작' 참조).

전원 OFF 인터럽트 태스크

전원 OFF 확정 후 전원 OFF 인터럽트 태스크가 실행됩니다. (전원 OFF 인터럽트 태스크를 'ON: 무효'로 설정하는 경우, 전원 OFF 확정 시점에서 통상 프로그램이 종료됩니다.)
단, 전원 OFF 인터럽트 태스크는 '전원 유지 시간'(=전원 OFF 감지 지연 시간 설정+전원 OFF 확정 처리 시간의 합계)이 종료한 시점에 강제적으로 종료됩니다. '전원 유지 시간'은 최대 10ms입니다.
따라서 필요에 따라 전원 OFF 감지 지연 시간을 설정하고 전원 OFF 인터럽트 태스크 실행 시간이 '10ms－전원 OFF 감지 지연 시간'(= 전원 OFF 확정 후 처리 시간) 이하가 되도록 전원 OFF 인터럽트 태스크를 작성하십시오.

(주) 전원 유니트 CJ1W-PD022를 장착했을 때는 설정할 수 없습니다(🕮 '부록-5 전원 OFF 시 PLC의 동작' 참조).

● '고속 인터럽트 기능 사용'(CJ2H CPU 유니트만 해당)

고속 인터럽트 기능을 유효화할 때 설정합니다.

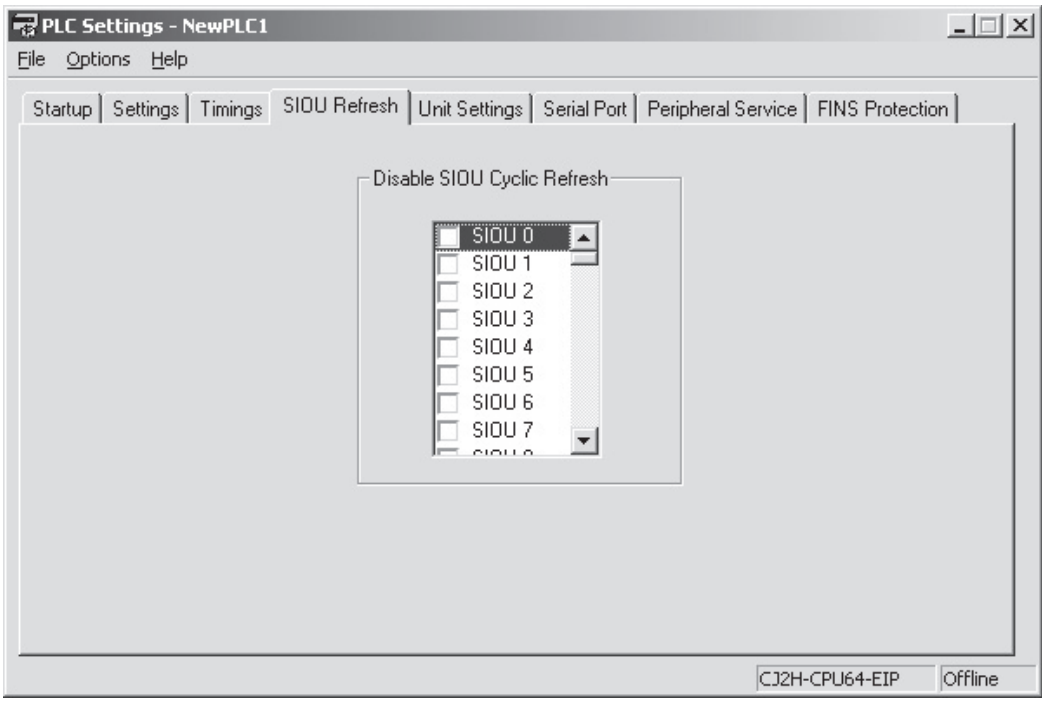
설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
고속 인터럽트 기능 사용	OFF: 고속 인터럽트 기능을 무효로 설정 ON: 고속 인터럽트 기능을 유효로 설정	OFF(고속 인터럽트 기능을무효로 설정)	'고속 인터럽트 기능 사용'을 ON으로 설정하면 고속 인터럽트 기능이 유효화됩니다.	—

고속 인터럽트 기능은 일정한 제약 조건에서 인터럽트 태스크의 실행 성능을 향상시키는 기능입니다.
자세한 내용은 📖 '10-2-6 고속 인터럽트 기능'을 참조하십시오.
PLC 시스템 설정에서 '동기 제어 사용'을 ON으로 하면 '고속 인터럽트 기능 사용'이 자동으로 ON이 됩니다.

● '유니트 간 동기 설정'(CJ2H CPU 유니트만 해당)

유니트 간 동기 제어 기능을 사용할 때 필요한 설정입니다.
자세한 내용은 📖 '10-8-7 PLC 시스템 설정'을 참조하십시오.

9-3-4 고기능 I/O 리프레시



고기능 I/O의 주기적 리프레시 금지 설정

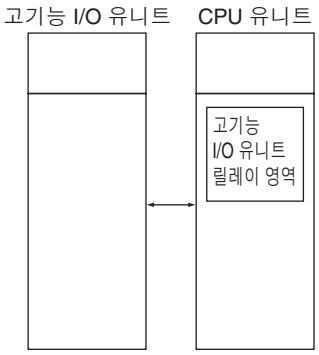
고기능 I/O 유니트와의 주기적 리프레시를 무효화할 때 설정합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
고기능 I/O 유니트 0~15호기와의 주기적 리프레시	OFF: 금지 안 함 ON: 금지	OFF (금지 안 함)	고기능 I/O 유니트와 주기적으로 리프레시할 때, 지정한 기기와 고기능 I/O 유니트 릴레이 영역(10CH/유니트)의 데이터 교환 여부를 설정합니다.	—
고기능 I/O 유니트 16~31호기와의 주기적 리프레시	OFF: 금지 안 함 ON: 금지	OFF (금지 안 함)		
고기능 I/O 유니트 32~47호기와의 주기적 리프레시	OFF: 금지 안 함 ON: 금지	OFF (금지 안 함)		
고기능 I/O 유니트 48~63호기와의 주기적 리프레시	OFF: 금지 안 함 ON: 금지	OFF (금지 안 함)		
고기능 I/O 유니트 64~79호기와의 주기적 리프레시	OFF: 금지 안 함 ON: 금지	OFF (금지 안 함)		
고기능 I/O 유니트 80~95호기와의 주기적 리프레시	OFF: 금지 안 함 ON: 금지	OFF (금지 안 함)		

● 고기능 I/O 유니트의 주기적 리프레시 금지로 설정해야 하는 경우

인터럽트 태스크 내에서 고기능 I/O 유니트에 대해 IORF/FIORF/IORD/IOWR 명령을 사용할 때는 반드시 이 고기능 I/O 유니트의 주기적인 리프레시를 'ON: 금지'로 설정하십시오.
고기능 I/O 유니트와의 주기적 리프레시(OFF: 금지 안 함)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/FIORF 명령을 통해 동일한 고기능 I/O 유니트에 대해 I/O 리프레시를 실행했을 때
- IORD/IOWR 명령에 의해 동일한 고기능 I/O 유니트의 메모리 영역에 대해 읽기 또는 쓰기를 실행했을 때

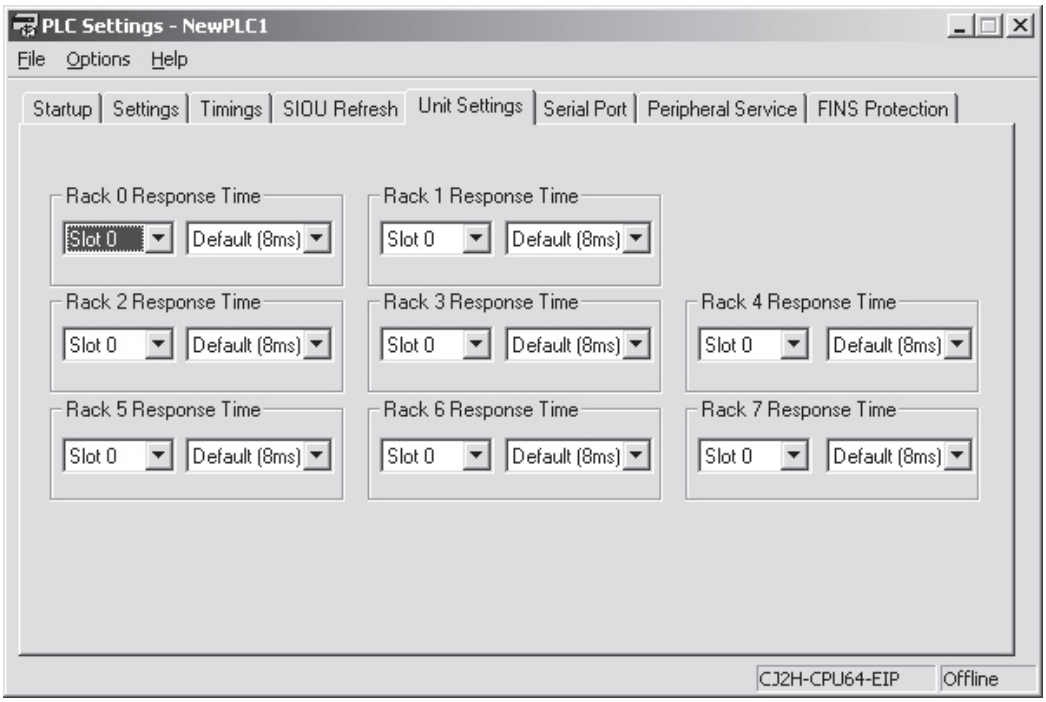


CPU 유니트가 지정된 고기능 I/O 유니트와 주기적으로 리프레시를 수행할 때 고기능 I/O 유니트 릴레이(2000CH~2959CH) 영역(10CH/유니트)을 데이터 변환할지 여부를 설정합니다.

사용상의 주의 사항

고기능 I/O 유니트와의 주기적인 리프레시를 하지 않는 것으로 설정한 경우, I/O 리프레시 명령(IORF) 또는 고기능 I/O 유니트 매회 리프레시 명령(FIORF)을 사용하여 사용자 프로그램에서 I/O 리프레시를 실행하도록 하십시오.
고기능 I/O 유니트의 경우 CPU 유니트에서 정기적(11초 이내)인 I/O 리프레시가 실행되지 않은 경우 [CPU 유니트 감시 이상]이 발생합니다(고기능 I/O 유니트의 ERH LED, RUN LED 점등).

9-3-5 유니트 설정



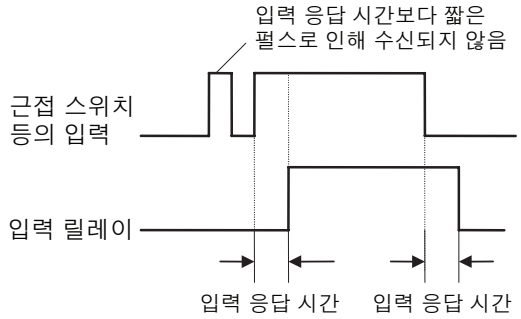
랙 0~7 입력 응답 시간

기본 I/O 유니트의 입력 응답 시간을 초기값(8ms) 이외의 값으로 설정할 때 사용합니다.
PLC 시스템 설정을 CPU 유니트로 전송한 후 PLC 전원을 다시 공급해야 합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
0 랙 0~9 슬롯 1 랙 0~9 슬롯 2 랙 0~9 슬롯 3 랙 0~9 슬롯 4 랙 0~9 슬롯 5 랙 0~9 슬롯 6 랙 0~9 슬롯 7 랙 0~9 슬롯	초기값(8ms) 필터 없음 0.5ms 1ms 2ms 4ms 8ms 16ms 32ms	8ms	기본 I/O 유니트(CJ 시리즈 기본 입력 유니트만 해당)의 입력 응답 시간(ON 응답 시간＝OFF 응답 시간)을 설정합니다. 필터 없음(0ms)으로 설정했을 때는 유니트 내부 소자의 지연에 의해 ON 및 OFF 시의 응답에 지연이 발생합니다. 유니트별 ON/OFF 응답 지연에 대해서는 "CJ 시리즈 CJ2 CPU 유니트 사용 설명서 하드웨어편"(SBCA-349)의 '부록-1 기본 I/O 유니트 사양'을 참조하십시오.	A220~A259CH (기본 I/O 유니트의 입력 응답 시간: 실제 유니트의 값)

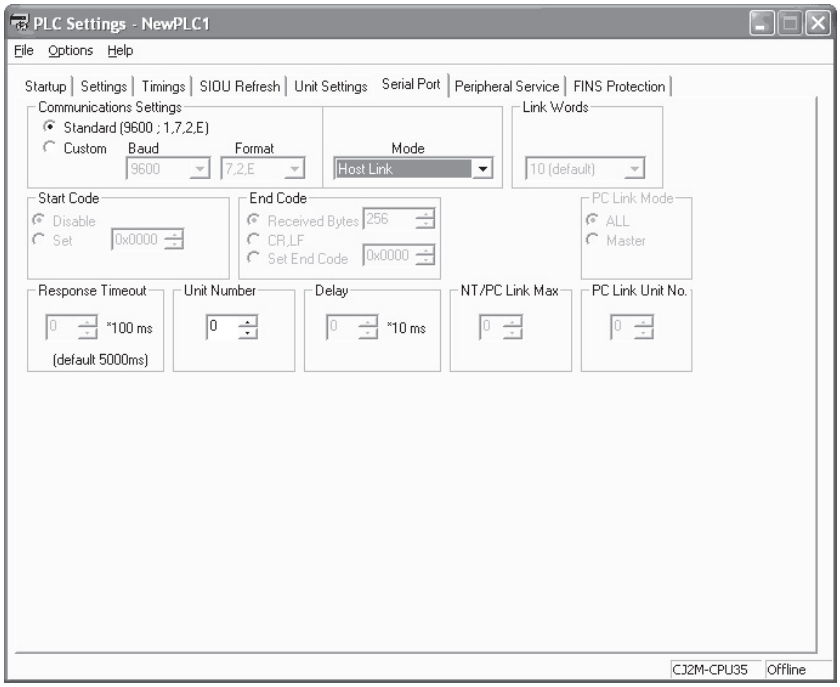
● 기본 I/O 유니트의 입력 응답 시간 설정 변경

입력 응답 시간을 길게 설정하면 채터링 또는 외부 노이즈의 영향을 줄일 수 있습니다. 반대로 짧게 설정하면(ON 또는 OFF 시간이 사이클 시간보다 긴 조건에서) 짧은 펄스 입력을 수신할 수 있습니다.



전원 ON 시, 설정된 입력 응답 시간을 각 기본 I/O 유니트로 전송합니다.
변경하면 각 값이 특수 보조 릴레이의 기본 I/O 유니트의 입력 응답 시간 설정값(A220~A259CH)에 저장됩니다. 프로그램 모드에서 이 PLC 시스템 설정을 변경한 경우, 실제로 기본 I/O 유니트에 설정된 값과 이 설정 값이 달라집니다. 이런 경우, 이 특수 보조 릴레이의 값을 통해 실제 기본 I/O 유니트의 입력 응답 시간을 모니터링합니다.

9-3-6 시리얼 포트



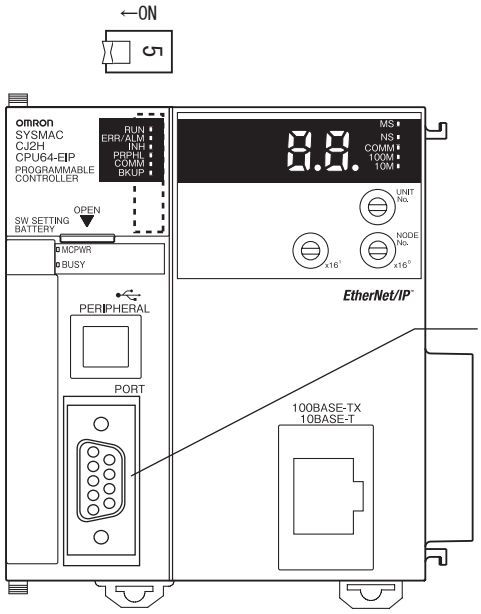
이 설정은 CPU 유니트의 딥 스위치 SW5가 OFF(기본값)로 설정된 경우만 유효화됩니다.

통신 설정

설정 항목	선택 데이터	초기값	설명	관련 특수 보조 릴레이
통신 설정	• 표준(9600: 1,7,2,E): 표준 • 사용자 설정: 임의 설정	표준 (9600: 1,7,2,E)	표준(9600: 1,7,2,E)은 다음과 같습니다. • 모드: 상위 링크 • 통신 속도: 9600비트/s • 시작 비트: 1비트 • 데이터 길이: 7비트 • 패리티: 짝수 • 정지 비트: 2비트 이러한 설정 이외로 사용할 때는 '사용자 설정'을 선택하여 '모드' 등을 설정합니다.	A619.02 (COMM 포트 설정 변경 중 플래그)

시리얼 포트 설정

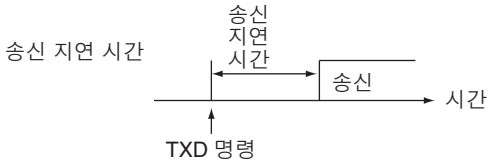
초기값과 임의 설정 중 어느 설정에서 시리얼 포트를 사용할지 선택합니다.
초기값 이외로 사용하는 경우 임의의 값을 설정합니다. 특히, 무수순 모드로 설정한 경우에는 해당 프레임 형식을 지정합니다.
이 설정은 STUP(RS-232C 통신 설정 변경) 명령에서도 변경할 수 있습니다.
이 경우, 명령이 실행된 후 설정 변경이 완료될 때까지 특수 보조 릴레이의 'COMM 포트 설정 변경 중 플래그'(A619.02)가 ON이 됩니다.
CPU 유니트의 딥 스위치 SW5를 ON으로 설정한 경우, 시리얼 포트는 연결된 주변 툴의 통신 조건을 자동 인식합니다. 이런 경우, 자동 인식된 통신 조건은 이 PLC 시스템 설정 영역에 저장되지 않습니다.



시리얼 포트
통신 조건
SW5를 OFF로 하고,

- 이 시리얼 포트 설정을 설정하지 않은 경우
  기본값은 상위 링크 모드에서
  시작 비트: 1비트, 데이터 수: 7비트
  패리티: 짝수　정지 비트: 2비트
  통신 속도: 9600비트/s
- 위 이외의 조건에서 사용하고 싶은 경우 시리얼 통신 모드:
  NT 링크, 무수순(주), 툴 버스, 상위 링크(다른 조건일 때)
  포트 통신 속도 등을 설정합니다.

무수순 모드일 때 다음의 데이터 설정



무수순 모드에서 송수신하는 메시지

		종료 코드		
		없음	있음	CR+LF
시작 코드	없음	데이터	데이터 ED	데이터 CR+LF
	있음	ST 데이터	ST 데이터 ED	ST 데이터 CR+LF
수신 데이터 수		데이터: 1~256바이트 길이 가능		

● [표준(9600: 1,7,2,E)] 선택 시

[표준(9600: 1,7,2,E)]을 선택한 경우 무조건 다음 통신 설정이 됩니다.

- 통신 속도: 9600비트/s
- 시작 비트: 1비트
- 데이터 길이: 7비트
- 정지 비트: 2비트
- 패리티: 짝수
- 모드(시리얼 통신 모드): 상위 링크

(주) '모드' 등의 다른 설정은 무효가 됩니다

[표준] 이외의 통신 설정을 사용하고 싶은 경우 다음 순서로 설정하십시오.

1 **'통신 설정'에서 [사용자 지정 옵션]을 선택합니다.**

2 **'모드'에서 시리얼 통신 모드를 선택합니다.**

3 **기타 항목을 설정합니다.**

● '사용자 설정' 선택 시

'사용자 설정'을 선택하면 통신 속도, 파라미터, 모드를 선택할 수 있게 됩니다.

● 통신 속도('사용자 설정' 선택 시에만 해당)

선택 데이터	설정	선택 데이터	설정	관련 특수 보조 릴레이
115200	115200비트/s	4800	4800비트/s/s	A619.02 (COMM 포트 설정 변경 중 플래그)
57600	57600비트/s	2400	2400비트/s	
38400	38400비트/s	1200	1200비트/s	
19200	19200비트/s	600	600비트/s	
9600(초기값)	9600비트/s	300	300비트/s	

● 파라미터('사용자 설정' 선택 시에만 해당)

선택 데이터	데이터 길이	정지 비트	패리티	관련 특수 보조 릴레이
7,1,E	7	1비트	짝수	A619.02 (COMM 포트 설정 변경 중 플래그)
7,1,O			홀수	
7,1,N			패리티 없음	
7,2,E(초기값)		2비트	짝수	
7,2,O			홀수	
7,2,N			패리티 없음	
8,1,E	8	1비트	짝수	
8,1,O			홀수	
8,1,N			패리티 없음	
8,2,E		2비트	짝수	
8,2,O			홀수	
8,2,N			패리티 없음	

● 모드('사용자 설정' 선택 시에만 해당)

시리얼 포트의 시리얼 통신 모드를 선택합니다.

설정 항목	선택 데이터	초기값	설명	관련 특수 보조 릴레이
모드	• 상위 링크 • NT 링크(1:N) • RS-232C(무수순) • 툴 버스 • 시리얼 게이트웨이 • 시리얼 PLC 링크(친국) • 시리얼 PLC 링크(자국)	상위 링크 (*1)	시리얼 포트를 사용할 시리얼 통신 모드를 지정합니다. (*2)	A619.02 (COMM 포트 설정 변경 중 플래그)

*1 상위 링크(SYSMAC WAY)란 일반적인 상위 PC와의 연결용 통신 모드입니다.
*2 NT 링크(1:1 모드)의 PT와는 통신할 수 없으므로 주의하십시오.

선택 모드별 관련 설정

모드	관련 설정
상위 링크(초기값)	통신 속도, 파라미터, 기기 번호
NT 링크(1:N)	통신 속도, NT/PLC 링크 최대 번호
RS-232C(무수순)	통신 속도, 파라미터, 시작 코드, 종료 코드, 딜레이
툴 버스	통신 속도
시리얼 게이트웨이	통신 속도, 파라미터, 응답 감시 시간
시리얼 PLC 링크(친국)	통신 설정, 모드, 링크 채널, PLC 링크의 연결 방식, NT/PLC 링크 최대 번호
시리얼 PLC 링크(자국)	통신 설정, 모드, PLC 링크의 자국 기기 번호

● '상위 링크'를 선택했을 때

상위 PC 등과 상위 링크에서 통신할 때, 그리고 통신 속도를 9600비트/s 이외로 설정할 때 선택합니다.

기기 번호

PLC의 자국으로서 기기 번호를 설정합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이	CPU 유니트 반영 타이밍
기기 번호	0~31	0	상위 링크에서 1:N 연결(N=2~32대)하는 경우, CPU 유니트의 상위 링크 기기 번호를 설정합니다.	A619.02 (COMM 포트 설정 변경 중 플래그)	매 사이클

● 'NT 링크(1 : N)'를 선택했을 때

오므론이 제작한 프로그래머블 터미널(NS 시리즈 또는 NT 시리즈)과 통신할 때 선택합니다.

통신 속도

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
통신 속도	115200비트/s, 38400비트/s	115200 비트/s	CX-Programmer에서 설정하는 경우는 '115200비트/s'를 권장합니다.	A619.02 (COMM 포트 설정 변경 중 플래그)

NT/PLC 링크 최대 번호

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
NT/PLC 링크 최대 번호	0~7	1	PLC에 연결하는 프로그래머블 터미널의 최대 기기 번호를 지정합니다.	A619.02 (COMM 포트 설정 변경 중 플래그)

● 'RS-232C(무수순)'를 선택했을 때

바코드 등과 무수순으로 통신할 때 선택합니다.

시작 코드

설정 항목	선택 데이터	초기값	설명	관련 특수 보조 릴레이
시작 코드	없음, 설정	없음	무수순 모드로 송수신하는 데이터 프레임 형식의 시작 코드 유무를 지정합니다.	—
	0x0000~00FF (0000~00FF Hex: 0x는 Hex와 같은 16진수의 의미)	0x 0000 (00Hex)	시작 코드를 '설정'으로 한 경우 해당 시작 코드를 16진수로 설정합니다.	—

종료 코드

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
종료 코드	• 수신 바이트 (종료 코드 없음) • CR,LF • 종료 코드 지정	없음 (수신 데이터 수 지정)	무수순 모드로 송수신하는 데이터 프레임 형식의 종료 코드를 지정합니다.	—
수신 바이트일 때	256 또는 1~255	256	종료 코드를 '수신 바이트'로 설정한 경우 무수순 모드로 송수신하는 데이터의 길이를 바이트 단위로 설정합니다. 또한 데이터 길이에 종료 코드, 시작 코드의 데이터 길이는 포함되지 않습니다. 1회의 TXD/RXD 명령으로 송수신할 수 있는 데이터 길이의 초기값은 최대 256 바이트입니다.	—
종료 코드 지정한 경우	0x0000~00FF (0000~00FF Hex: 0x는 Hex와 같은 16진수의 의미)	0x 00 (00Hex)	종료 코드를 '종료 코드 지정'으로 한 경우 종료 코드를 16진수로 설정합니다.	—

딜레이(송신 딜레이 시간)

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
딜레이	0~99990ms (10ms 단위)	0ms	TXD 명령 실행 시, 여기서 지정한 송신 딜레이 시간 경과 후 시리얼 포트에서 데이터가 송신됩니다.	—

● '툴 버스'를 선택했을 때

주변 툴(CX-Programmer 등)과 통신할 때 선택합니다.

통신 속도

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
통신 속도	115200, 38400, 57600, 19200, 9600비트/s	115200비트/s	툴 버스를 지정하는 경우	A619.02 (COMM 포트 설정 변경 중 플래그)

CPU 유니트의 딥 스위치 SW5를 ON으로 설정하면 CX-Programmer의 통신 속도를 자동으로 인식해서 툴 버스와 연결할 수 있습니다.

● '시리얼 게이트웨이'를 선택했을 때

CompoWay/F에서 오므론이 제작한 구성 요소와 통신할 때 선택합니다.

응답 감시 시간

설정 항목	설정 데이터	초기값	설명(*)	관련 특수 보조 릴레이
응답 감시 시간	5000ms, 100~25500ms (100ms 단위)	0 (5000ms)	시리얼 게이트웨이 기능을 통해 각 프로토콜로 변환한 메시지를 송신한 후 응답을 수신하기까지의 시간을 감시합니다. 기본값: 5초 임의 설정: 0.1~25.5초	A619.02 (COMM 포트 설정 변경 중 플래그)

* 시간 초과 시, FINS 전송지에 FINS 종료 코드(0205Hex(응답 시간 초과))를 반환합니다.

● '시리얼 PLC 링크(친국)'를 선택했을 때(CJ2M CPU 유니트만 해당)

CJ2M CPU 유니트끼리 또는 CJ2M CPU 유니트와 CJ1M/CP1H/CP1L/CP1E 유니트 사이에서 친국으로서 프로그램 없이 데이터를 교환할 때 선택합니다.

통신 설정

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
통신 속도	115200비트/s, 38400비트/s	115200 비트/s	시리얼 PLC 링크(친국)를 지정하는 경우	A619.02 (COMM 포트 설정 변경 중 플래그)

링크 채널

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
링크 채널	1~10CH	10CH	친국에서만 설정합니다. 시리얼 PLC 링크 릴레이 중, 1국당 몇 채널을 시리얼 PLC 링크에 사용할지 지정합니다.	A619.02 (COMM 포트 설정 변경 중 플래그)

PLC 링크의 연결 방식

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
연결 방식	전체 국 연결 또는 친국 연결	전체 국 연결 방식	PLC 링크에서 1:N으로 연결하는 경우 친국에 연결하는 자국의 최대 기기 번호를 지정합니다. 친국에서만 설정합니다.	A619.02 (COMM 포트 설정 변경 중 플래그)

NT/PLC 링크 최대 번호

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
NT/PLC 링크 최대 번호	0~7	전체 국 연결 방식	PLC 링크에서 1:N으로 연결하는 경우 친국에 연결하는 자국의 최대 기기 번호를 지정합니다. 친국에서만 설정합니다.	A619.02 (COMM 포트 설정 변경 중 플래그)

● '시리얼 PLC 링크(자국)'를 선택했을 때(CJ2M CPU 유니트만 해당)

CJ2M CPU 유니트끼리 또는 CJ2M CPU 유니트와 CJ1M/CP1H/CP1L/CP1E 유니트 사이에서 자국으로서 프로그램 없이 데이터를 교환할 때 선택합니다.

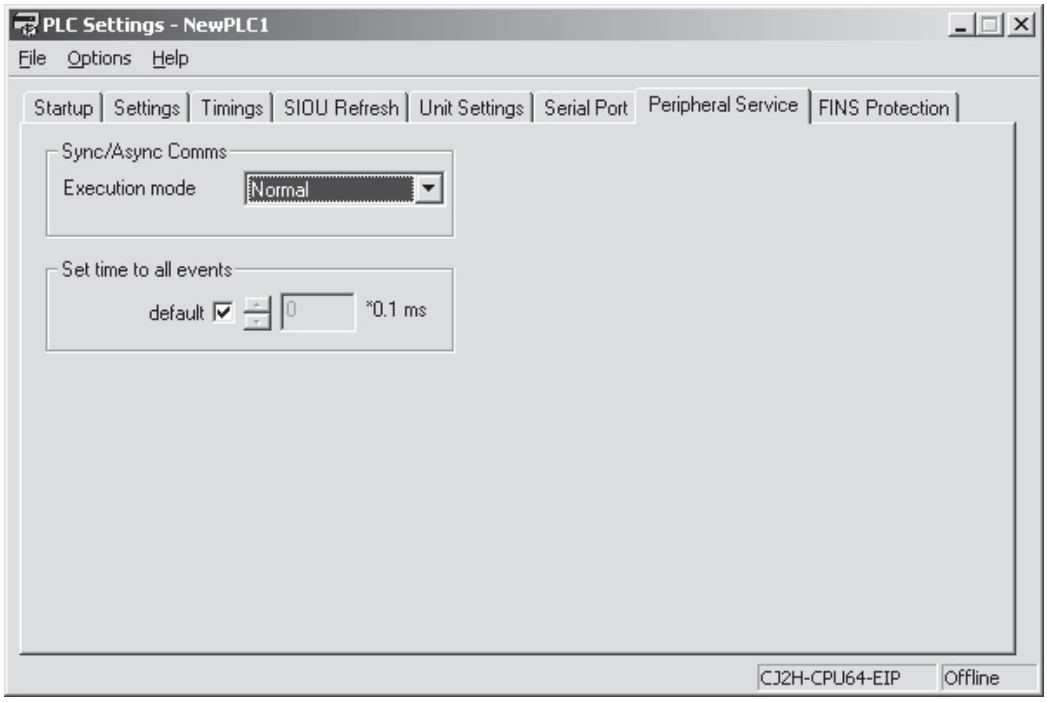
통신 설정

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
통신 속도	115200비트/s, 38400비트/s	115200 비트/s	시리얼 PLC 링크(자국)를 지정하는 경우	A619.02 (COMM 포트 설정 변경 중 플래그)

PLC 링크의 자국 기기 번호

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
PLC 링크의 자국 기기 번호	0~7	0	PLC 링크에서 1:N으로 연결하는 경우 친국에 연결하는 자국의 기기 번호를 지정합니다.	A619.02 (COMM 포트 설정 변경 중 플래그)

9-3-7 주변 서비스



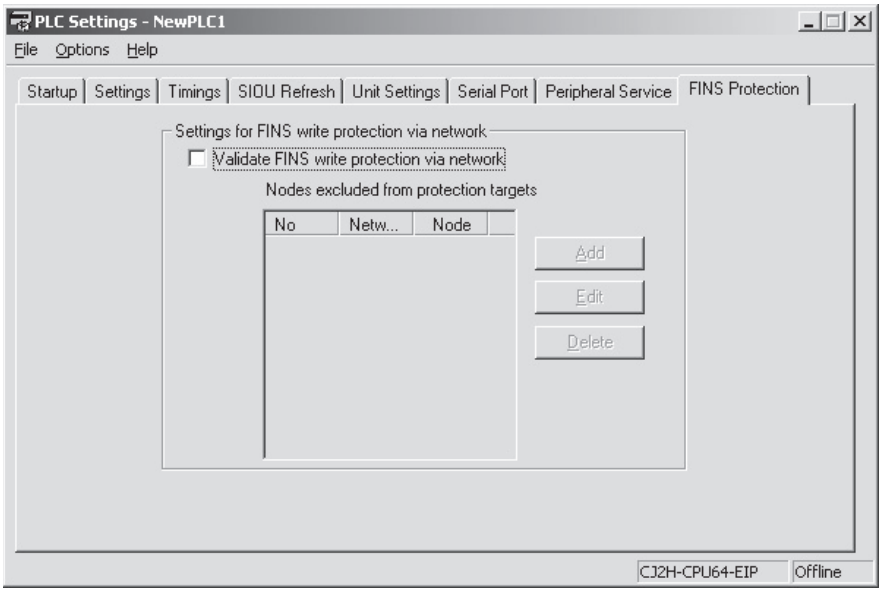
● CPU 실행 처리 모드 설정

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
실행 모드	통상 모드	통상	CJ2 CPU 유니트의 경우, 통상 모드만 선택할 수 있습니다.	–

● 주변 서비스 일률 시간

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
주변 서비스 일률 시간	초기값 (사이클 시간의 10%) 0.1~3276.7ms	사이클 시간의 10% (단, 사이클 시간의 10%가 0.1ms 이하인 경우는 0.1ms)	주변 서비스 시간(각 서비스의 합계)의 최대 시간을 설정합니다. 0.1ms 이상을 설정할 수 있습니다.	–

9-3-8 FINS 금지



네트워크를 경유하는 FINS 쓰기 금지 설정

- **'네트워크를 경유하는 FINS 쓰기 금지 기능 유효화'**

  네트워크를 경유하는 FINS 명령을 사용한 쓰기를 금지할지 여부를 설정합니다.
  시리얼 연결 이외의 네트워크를 경유했을 때 실수로 쓰기가 실행되는 것을 막기 위해 사용합니다.
  '금지 대상 외 노드 지정'에서 설정한 노드 이외에서의 쓰기가 금지됩니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
네트워크를 경유하는 FINS 쓰기 금지 기능 유효화	OFF: FINS 쓰기 금지 기능을 유효화 안 함 ON: FINS 쓰기 금지 기능을 유효로 설정	OFF	네트워크를 경유(직접 시리얼 연결 이외)하여 FINS 명령을 사용한 CPU 유니트에 대한 쓰기 금지 기능을 유효화할지 여부를 지정합니다.	—

- **'금지 대상 외 노드 지정'**

  쓰기 금지를 실행할 경우 쓰기를 허가할 노드를 설정합니다.
  어느 네트워크 주소의 어느 노드에서 FINS 쓰기를 허가할지(쓰기 금지 대상 외) 설정합니다. 이와 동시에, 결과적으로 쓰기를 허가할(금지 대상에서 제외) 노드의 수를 설정하게 됩니다.
  네트워크와 노드를 설정합니다. 최대 32노드를 설정할 수 있습니다.
  다음과 같이 설정하지 않으면(금지 대상에서 제외되는 노드 수＝0으로 간주), 로컬 노드 이외의 모든 노드에서 쓰기 처리를 금지합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
금지 대상 외 노드 지정 (*1)	0~127	—	네트워크: FINS 송신처 네트워크 주소	—
	1~255(*2)	—	노드: FINS 송신처 노드 주소	—

*1 FINS 쓰기 금지 기능이 유효할 때만 유효.
*2 255는 노드 주소를 지정하지 않습니다.

10

CPU 내장 기능

이 장에서는 CPU 유니트 본체에서 사용할 수 있는 기능에 대해 설명합니다.

10-1 시계 기능 10-3
10-1-1 시계 기능 10-3
10-1-2 시각 기록 기능 10-4
10-1-3 프리 러닝 타이머 기능 10-6

10-2 사이클 시간 관련/고속 처리 기능 10-7
10-2-1 사이클 시간 일정화 기능 10-7
10-2-2 사이클 시간 감시 기능 10-8
10-2-3 사이클 시간 모니터링 기능 10-9
10-2-4 펄스 캐치 기능 10-9
10-2-5 백그라운드 처리 실행 기능 10-10
10-2-6 고속 인터럽트 기능 10-18

10-3 운전 시작 시 동작 설정 및 보수 기능 10-20
10-3-1 동작 모드 변경 및 전원 ON 시의 유지 설정 기능 10-20
10-3-2 전원 OFF 감지 지연 시간 설정 10-22
10-3-3 전원 OFF 금지 구간 설정 10-23
10-3-4 운전 중 출력 기능 10-24
10-3-5 전원 ON 시 자동 전송 기능 10-25

10-4 유니트 관련 관리 기능 10-32
10-4-1 기본 I/O 유니트 관련 10-32
10-4-2 CPU 고기능 유니트 관련 10-33
10-4-3 고기능 I/O 유니트 관련 10-34

10-5 메모리 관리 기능 10-35
10-5-1 자동 백업 기능 10-35
10-5-2 EM 파일 메모리 기능 10-38
10-5-3 코멘트 기억 기능 10-38
10-5-4 프로그램 교체 기능 10-40

10-6 보호 기능 10-46
10-6-1 딥 스위치에 의한 쓰기 방지 기능 10-46
10-6-2 비밀번호를 사용한 읽기 방지 기능 10-47
10-6-3 제조 로트 번호에 따른 운전 금지 기능 10-53
10-6-4 FINS 쓰기 방지 기능 10-54
10-6-5 PLC 이름 기능 10-58

10-7 디버그 기능 10-61
10-7-1 강제 세트/리셋 10-61
10-7-2 테스트 입력 10-62
10-7-3 미분 모니터 10-62
10-7-4 온라인 편집 10-63
10-7-5 부하 차단 10-65
10-7-6 데이터 트레이스 10-66
10-7-7 에러 발생 시 정지 위치 저장 기능 10-70
10-7-8 고장 진단 명령(FAL 명령, FALS 명령) 기능 10-71
10-7-9 시스템 이상을 고의로 발생시키는 기능 10-72
10-7-10 고장 지점 검출 명령(FPD 명령) 기능 10-73
10-8 유니트 간 동기 제어 기능 10-75
10-8-1 개요 10-75
10-8-2 유니트 간 동기 제어의 자세한 구조 10-78
10-8-3 유니트 간 동기 제어 기능의 사양 10-81
10-8-4 동기 데이터 링크 릴레이 10-82
10-8-5 유니트 간 동기 제어 기능의 사용상 제한 사항 10-86
10-8-6 사용 순서 10-88
10-8-7 PLC 시스템 설정 10-89
10-8-8 동기 인터럽트 태스크의 작성 10-91
10-8-9 유니트 간 동기 제어 동작의 조정/문제 해결 10-92

10-1 시계 기능

10-1-1 시계 기능

시계 데이터

CJ 시리즈는 CPU 유니트에 시계가 내장돼 있습니다.
다음 특수 보조 릴레이에 CPU 유니트의 내부 시계 데이터가 BCD로 저장됩니다.

명칭	주소	설명
시계 데이터	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

배터리를 장착하지 않았거나 배터리가 소모된 경우, 위에서 설명한 특수 보조 릴레이의 '시계 데이터' (A351~354CH) 값이 불안정해집니다. 따라서 그런 경우 시계 기능을 사용할 수 없습니다.

시계 데이터의 설정

CX-Programmer에서 온라인으로 연결하여 프로젝트 트리의 [PLC 시계] 아이콘을 두 번 클릭하는 방법과 다음 명령을 실행하는 방법을 사용하여 설정할 수 있습니다. 그리고 시계 데이터의 가산/감산, 시분초와 초 사이의 변환을 수행할 수 있습니다.

● 시계 기능 관련 명령

명령 이름	명령어	내용
시계 보정	DATE	지정한 시계 데이터에 따라 내부 시계의 값을 설정
시분초→초 변환	SEC	시분초 데이터를 초 데이터로 변환
초→시분초 변환	HMS	초 데이터를 시분초 데이터로 변환
달력 가산	CADD	시각 데이터에 시간 데이터를 가산
달력 감산	CSUB	시각 데이터에서 시간 데이터를 감산

10-1-2 시각 기록 기능

전원을 투입한 시각 및 내역, 전원을 OFF한 시각, 그 횟수, 통전 시간, 사용자 메모리(프로그램, 파라미터) 재기입 시각, 운전 시작/정지 시각이 특수 보조 릴레이에 자동으로 저장됩니다.
배터리를 장착하지 않은 경우 또는 배터리가 소모된 경우에는 '시각 기록 기능'을 사용할 수 없습니다.

전원 투입 시각

PLC 본체에 전원을 투입한 연월일 및 시각의 내역이 다음의 특수 보조 릴레이에 저장됩니다.

명칭	주소	설명
전원 ON 시각 이력 데이터 #1	A720CH~ A722CH	BCD 데이터로 저장됩니다. A720.00~A720.07: 초(00~59) A720.08~A720.15: 분(00~59) A721.00~A721.07: 시(00~23) A721.08~A721.15: 일(01~31) A722.00~A722.07: 월(01~12) A722.08~A722.15: 년(00~99)
전원 ON 시각 이력 데이터 #2	A723CH~ A725CH	1회 이전의 전원 ON 시각 데이터이며, BCD 데이터로 저장됩니다. 저장 형식은 A720CH~A722CH와 같습니다.
전원 ON 시각 이력 데이터 #3	A726CH~ A728CH	2회 이전의 전원 ON 시각 데이터이며, BCD 데이터로 저장됩니다. 저장 형식은 A720CH~A722CH와 같습니다.
전원 ON 시각 이력 데이터 #4	A729CH~ A731CH	3회 이전의 전원 ON 시각 데이터이며, BCD 데이터로 저장됩니다. 저장 형식은 A720CH~A722CH와 같습니다.
전원 ON 시각 이력 데이터 #5	A732CH~ A734CH	4회 이전의 전원 ON 시각 데이터이며, BCD 데이터로 저장됩니다. 저장 형식은 A720CH~A722CH와 같습니다.
전원 ON 시각 이력 데이터 #6	A735CH~ A737CH	5회 이전의 전원 ON 시각 데이터이며, BCD 데이터로 저장됩니다. 저장 형식은 A720CH~A722CH와 같습니다.
전원 ON 시각 이력 데이터 #7	A738CH~ A740CH	6회 이전의 전원 ON 시각 데이터이며, BCD 데이터로 저장됩니다. 저장 형식은 A720CH~A722CH와 같습니다.
전원 ON 시각 이력 데이터 #8	A741CH~ A743CH	7회 이전의 전원 ON 시각 데이터이며, BCD 데이터로 저장됩니다. 저장 형식은 A720CH~A722CH와 같습니다.
전원 ON 시각 이력 데이터 #9	A744CH~ A746CH	8회 이전의 전원 ON 시각 데이터이며, BCD 데이터로 저장됩니다. 저장 형식은 A720CH~A722CH와 같습니다.
전원 ON 시각 이력 데이터 #10	A747CH~ A749CH	9회 이전의 전원 ON 시각 데이터이며, BCD 데이터로 저장됩니다. 저장 형식은 A720CH~A722CH와 같습니다.

전원 OFF 시각

PLC 본체의 전원을 OFF한 날짜와 시각이 다음의 특수 보조 릴레이에 저장됩니다.
또한 전원 OFF 누적 횟수도 다음의 특수 보조 릴레이에 저장됩니다.

명칭	주소	설명
전원 OFF 시각	A512CH~ A513CH	전원 OFF 시의 시각이 저장됩니다. 전원 OFF가 발생할 때마다 갱신되며 BCD 데이터로 저장됩니다. A512.00~A512.07: 초(00~59) A512.08~A512.15: 분(00~59) A513.00~A513.07: 시(00~23) A513.08~A513.15: 일(01~31)
전원 OFF 발생 횟수	A514CH	PLC 본체에 최초로 전원을 투입한 이후 전원 OFF가 발생할 때마다 누적(+1)됩니다. BIN으로 저장되며, 리셋하는 경우는 0000Hex를 저장하십시오.

통전 시간

PLC 본체에 전원을 투입한 후 통전된 시간이 다음의 특수 보조 릴레이에 저장됩니다.

명칭	주소	설명
통전 시간	A523CH	PLC 본체의 통전 시간을 BIN 16비트, 10시간 단위로 저장합니다. 리셋하는 경우는 0000Hex를 저장하십시오.

사용자 메모리 재기입 시각

사용자 프로그램 및 파라미터 영역(PLC 시스템 설정, I/O 테이블, 라우팅 테이블, CPU 고기능 유니트 시스템 설정)이 재기입됐을 때의 연월일 및 시각이 다음의 특수 보조 릴레이에 저장됩니다.

명칭	주소	설명
사용자 프로그램 재기입 일시	A090CH~ A093CH	사용자 프로그램 재기입 시, 해당 시각(연월일, 시분초, 요일)이 BCD 데이터로 저장됩니다. A090.00~A090.07: 초(00~59) A090.08~A090.15: 분(00~59) A091.00~A091.07: 시(00~23) A091.08~A091.15: 일(01~31) A092.00~A092.07: 월(01~12) A092.08~A092.15: 년(00~99) A093.00~A093.07: 요일(00~06) 00: 일요일, 01: 월요일, 02: 화요일, 03: 수요일, 04: 목요일, 05: 금요일, 06: 토요일
파라미터 영역 재기입 일시	A094CH~ A097CH	파라미터 영역 재기입 시, 해당 시각(연월일, 시분초, 요일)이 BCD 데이터로 저장됩니다. A094.00~A094.07: 초(00~59) A094.08~A094.15: 분(00~59) A095.00~A095.07: 시(00~23) A095.08~A095.15: 일(01~31) A096.00~A096.07: 월(01~12) A096.08~A096.15: 년(00~99) A097.00~A097.07: 요일(00~06) 00: 일요일, 01: 월요일, 02: 화요일, 03: 수요일, 04: 목요일, 05: 금요일, 06: 토요일

운전 시작 시각, 운전 정지 시각

PLC의 운전을 시작한 연월일과 시각 및 운전을 정지한 연월일과 시각이 다음의 특수 보조 릴레이에 자동으로 저장됩니다.

명칭	주소	설명
운전 시작 시각	A515CH~ A517CH	동작 모드를 '운전' 또는 '모니터' 모드로 변경하면 운전을 시작한 시각이 BCD 데이터로 저장됩니다.(*1) A515.00~A515.07: 초(00~59) A515.08~A515.15: 분(00~59) A516.00~A516.07: 시(00~23) A516.08~A516.15: 일(01~31) A517.00~A517.07: 월(01~12) A517.08~A517.15: 년(00~99)
운전 정지 시각	A518CH~ A520CH	운전 정지 이상이 발생하거나 동작 모드를 '프로그램' 모드로 변경한 경우, 운전을 정지한 시각이 BCD 데이터로 저장됩니다.(*2) A518.00~A518.07: 초(00~59) A518.08~A518.15: 분(00~59) A519.00~A519.07: 시(00~23) A519.08~A519.15: 일(01~31) A520.00~A520.07: 월(01~12) A520.08~A520.15: 년(00~99)

*1 전원을 켰을 때부터 운전 시작 전까지는 지난 번의 운전 시작 시각이 저장되어 있습니다.

*2 이상 발생 시에는 이상 발생 시각이 저장됩니다. 그 후 '프로그램' 모드로 변경하면 그 시각으로 덮어 쓰게 됩니다.

10-1-3 프리 러닝 타이머 기능

전원 투입 후에 사용되는 시스템 타이머는 특수 보조 릴레이(A0CH, A1CH)에 포함되어 있습니다. 따라서 타이머 명령을 사용하지 않고 시간 간격을 계산할 수 있습니다.
다음의 특수 보조 릴레이에 전원 투입 후의 시스템 타이머가 저장됩니다.

명칭	주소	설명
10ms 가산 프리 러닝 타이머	A0CH	전원을 켜면 0000Hex가 되고 10ms마다 1이 자동으로 가산됩니다. FFFFHex(655,350ms)까지 도달하면 0000Hex로 돌아가서 10ms마다 1씩 자동 가산이 반복됩니다.
100ms 가산 프리 러닝 타이머	A1CH	전원을 켜면 0000Hex가 되고 100ms마다 1이 자동으로 가산됩니다. FFFFHex(6,553,500ms)까지 도달하면 0000Hex로 돌아가서 100ms마다 1씩 자동 가산이 반복됩니다.
1s 가산 프리 러닝 타이머	A2CH	전원을 켜면 0000Hex가 되고 1s마다 1이 자동으로 가산됩니다. FFFFHex(65,535s)까지 도달하면 0000Hex로 돌아가서 1s마다 1씩 자동 가산이 반복됩니다.

(주) 동작 모드를 운전 모드로 변경했을 때도 1이 자동으로 가산되는 동작은 계속됩니다.

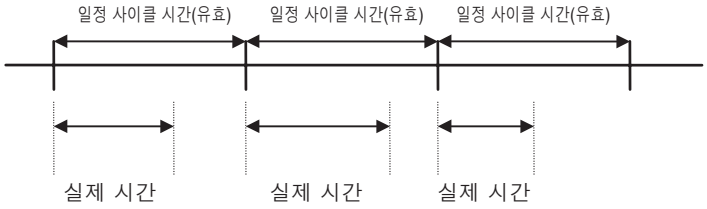
예) 처리 A일 때 A0CH 값과 다음의 처리 B일 때 A0CH 값의 차를 계산하면 타이머 명령을 사용하지 않고 처리 A에서 처리 B까지의 시간 간격(10ms 단위)을 계산할 수 있습니다.

10-2 사이클 시간 관련/고속 처리 기능

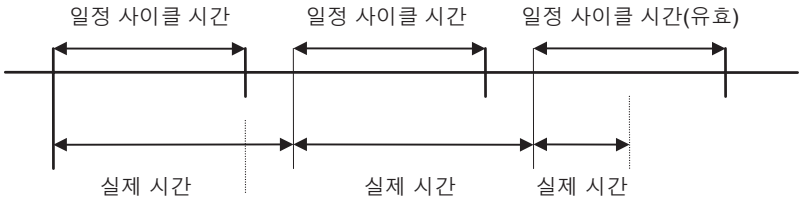
10-2-1 사이클 시간 일정화 기능

사이클 시간을 일정하게 할 수 있습니다. 그 결과, 일정 간격으로 프로그램을 반복 실행하면 입출력 응답의 편차를 없앨 수 있습니다.

PLC 시스템 설정에서 고정 사이클 시간을 설정하여 구현할 수 있습니다(0.2~32,000ms 범위, 0.1ms 단위).

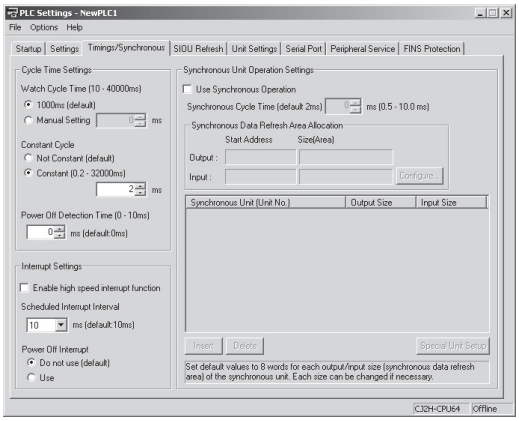


실제 사이클 시간이 설정된 사이클 시간보다 큰 경우, 사이클 시간 일정화 기능은 무시되고 실제 사이클 시간으로 동작합니다.



- **PLC 시스템 설정**

  CX-Programmer의 '타이밍/동기' 탭에 있는 '고정 사이클 시간'으로 설정합니다.



참고

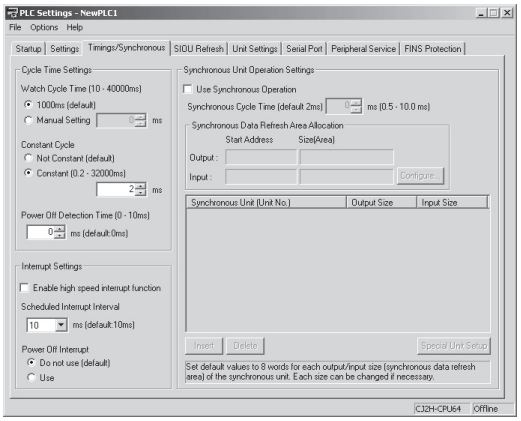
CPU 유니트의 운전 중(동작 모드가 '모니터' 모드일 때), CX-Programmer의 사이클 모니터 화면에서 고정 사이클 시간을 변경할 수 있습니다(CJ2H CPU 유니트 Ver.1.1 이상 및 CJ2M CPU 유니트).
자세한 내용은 '12-1 사이클 시간의 모니터링'을 참조하십시오.

10-2-2 사이클 시간 감시 기능

설정된 모니터 사이클 시간보다 실제 사이클 시간이 긴 경우, CPU 유니트는 운전을 정지합니다. 이때, '사이클 시간 초과'가 됩니다. 초기값은 1초입니다.

● PLC 시스템 설정

CX-Programmer의 '타이밍/동기' 탭에 있는 '모니터 사이클 시간'으로 설정합니다.



● 관련 특수 보조 릴레이

명칭	주소	내용
사이클 시간 초과 플래그	A401.08	사이클 시간 현재값이 PLC 시스템 설정의 '모니터 사이클 시간'을 초과했을 때 1(ON)이 됩니다.

10-2-3 사이클 시간 모니터링 기능

특수 보조 릴레이의 사이클 시간 최대값(A262~263CH), 현재값(A264~265CH), 266~267CH가 특수 보조 릴레이에 사이클마다 저장됩니다.

● 관련 특수 보조 릴레이

명칭	주소	내용
사이클 시간 최대값 (0.1ms 단위)	A262~263CH	0.1ms 단위로 사이클 시간의 최대값이 BIN 32비트로 사이클마다 저장됩니다. 범위는 다음과 같습니다. 0~429496729.5ms(0~FFFFFFFFms), 0.1ms 단위 또한 A262CH: 하위 데이터, A263CH: 상위 데이터가 저장됩니다.
사이클 시간 현재값 (0.1ms 단위)	A264~265CH	0.1ms 단위로 사이클 시간의 현재값이 BIN 32비트로 사이클마다 저장됩니다. 범위는 다음과 같습니다. 0~429496729.5ms(0~FFFFFFFFms), 0.1ms 단위 또한 A264CH: 하위 데이터, A265CH: 상위 데이터가 저장됩니다. 0~429496729.5ms, 0.1ms 단위 (0~FFFFFFFFms)
사이클 시간 현재값 (0.01ms 단위)	A266~267CH	0.01ms 단위로 사이클 시간의 현재값이 BIN 32비트로 사이클마다 저장됩니다. 범위는 다음과 같습니다. 0~42949672.95ms(0~FFFFFFFFms), 0.01ms 단위 또한 A266CH: 하위 데이터, A267CH: 상위 데이터가 저장됩니다. 0~42949672.95ms, 0.1ms 단위 (0~FFFFFFFFms)

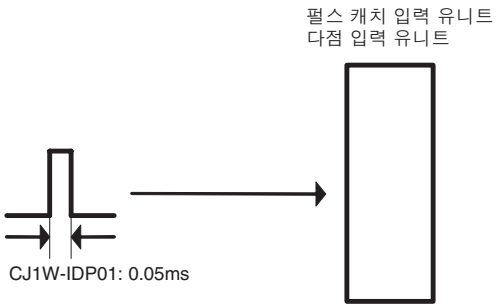
또한 CX-Programmer에서 과거 8스캔 분량의 사이클 시간에 대한 평균값을 읽어올 수 있습니다.

참고

사이클 시간을 단축하려면 다음과 같은 방법을 사용할 수 있습니다.
- 실행하지 않는 태스크는 정지합니다.
- 실행하지 않는 프로그램 영역은 JMP/CJP/CJPN 명령－JME 명령을 통해 건너뜁니다.

10-2-4 펄스 캐치 기능

사이클 시간보다 짧은 펄스를 수신한 경우는 펄스 캐치 입력 유니트(CJ 시리즈: CJ1W-IDP01)의 펄스 캐치 기능을 사용합니다.
펄스 캐치 기능으로 전송할 수 있는 펄스 폭(ON 시간)은 CJ1W-IDP01에서 0.05ms입니다.



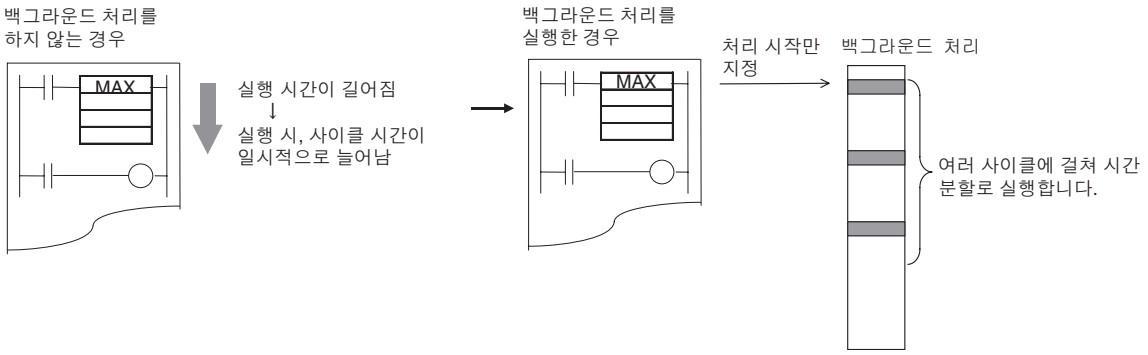
내부 회로에 입력된 내용은 입력을 리프레시할 때 클리어됩니다.

10-2-5 백그라운드 처리 실행 기능

데이터 검색 등의 테이블 데이터 처리, 문자열 검색 등의 문자열 처리를 수행하는 경우, 해당 명령을 실행하면 명령을 실행하는 시간이 걸리기 때문에 사이클 시간이 일시적으로 늘어나고, 그 결과 사이클 시간의 편차가 발생합니다.
다음과 같은 명령 종류에서 명령 실행을 여러 사이클에 걸쳐 시간 분할 처리(백그라운드 처리)하면 사이클 시간의 편차를 줄일 수 있습니다. PLC 시스템 설정에서 명령 종류별로 백그라운드 처리를 실행/비실행으로 지정합니다.

- 테이블 데이터 처리 명령
- 문자열 처리 명령
- 데이터 이동 명령(비동기 이동 레지스터 명령만 해당)

이렇게 하면 명령을 실행했을 때 일시적으로 사이클 시간이 늘어나는 것을 방지할 수 있습니다.



사용상의 주의 사항

CJ2H CPU 유니트에서 PLC 시스템 설정을 통해 '고속 인터럽트 기능 사용'을 유효화한 경우, 인터럽트 태스크 내에서는 백그라운드 처리를 사용할 수 없습니다. 명령 처리 에러가 됩니다.

● 대상 명령

이 기능은 펑션 블록 정의 내에서는 동작하지 않고 일반적인 명령으로 동작합니다.

• 테이블 데이터 처리 명령

명령 이름	니모닉
데이터 검색	SRCH
바이트 스와핑	SWAP
최대값 검색	MAX
	MAXL
	MAXF
	MAXD
최소값 검색	MIN
	MINL
	MINF
	MIND
합계 값 산출	SUM
FCS 값 산출	FCS

• 문자열 처리 명령

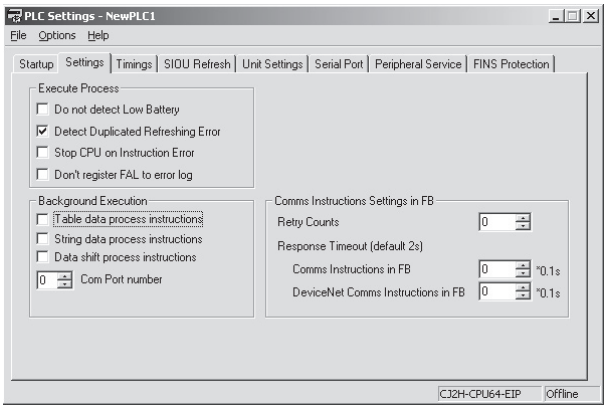
명령 이름	니모닉
문자열, 전송	MOV$
문자열, 연결	+$
문자열, 왼쪽에서 꺼냄	LEFT$
문자열, 오른쪽에서 꺼냄	RGHT$
문자열, 임의 위치에서 꺼냄	MID$
문자열, 검색	FIND$
문자열, 길이 검출	LEN$
문자열, 치환	RPLC$
문자열, 삭제	DEL$
문자열, 교환	XCHG$
문자열, 클리어	CLR$
문자열, 삽입	INS$

• 데이터 이동 명령

명령 이름	니모닉
비동기 이동 레지스터	ASFT

● **조작 순서**

1 **명령 종류에 따라 백그라운드 처리의 실행/비실행을 CX-Programmer의 PLC 시스템 설정에서 '설정' 탭의 [백그라운드 실행]을 사용하여 설정합니다.**



2 **백그라운드 처리를 실행하기 위해 사용하는 내부 논리 포트 번호는 PLC 시스템 설정에서 지정합니다(각 명령 종류 공통).**

하나의 내부 논리 포트를 백그라운드 처리를 위해 공통으로 사용합니다. 따라서 어떤 명령에서 지정된 내부 논리 포트를 사용하여 백그라운드 처리를 실행하는 중에는 다른 명령에서 백그라운드 처리를 실행할 수 없습니다. 사용하는 내부 논리 포트 번호에 대응하는 통신 명령 실행 가능 플래그를 입력 조건으로 해서 여러 백그라운드 처리 명령 사이에서 배타적 제어를 수행해야 합니다.

3 **백그라운드 처리를 실행하도록 설정한 경우, 명령 실행 조건이 성립했을 때 백그라운드 처리 명령은 (해당 사이클 내에서 처리를 완료하지 않고) 처리의 시작만 지시합니다.**

4 **처리가 시작되면 사용하는 내부 논리 포트 번호에 대응하는 통신 명령 실행 가능 플래그가 0(OFF)이 됩니다.**

5 **몇 사이클에 걸쳐(백그라운드로) 처리가 실행됩니다.**

6 **처리가 종료되면 사용하는 내부 논리 포트 번호에 대응하는 통신 명령 실행 가능 플래그가 1(ON)이 됩니다.**

그 결과, 다른 백그라운드 처리 명령을 실행할 수 있게 됩니다.

● 백그라운드 실행 시, 일반적인 명령과의 차이점

백그라운드로 실행하는 경우, 일반적인 명령과 다음과 같은 점이 다릅니다.

인덱스 레지스터(IR) 출력을 지정하는 경우

최대값 검색(MAX/MAXL/MAXF/MAXD) 명령 또는 최소값 검색(MIN/MINL/MINF/MIND) 명령에서 인덱스 레지스터(IR)에 최대값 또는 최소값이 있는 CH 번호의 I/O 메모리 실제 주소를 출력 지정한 경우, 그 CH 번호의 I/O 메모리 실제 주소를 인덱스 레지스터(IR)에는 출력하지 않는 대신 특수 보조 릴레이 A595CH/A596CH에 출력합니다.
따라서 백그라운드로 실행하고 싶지 않을 때와 마찬가지로, 인덱스 레지스터(IR)에 CH 번호를 저장하고 싶은 경우는 일단 A595CH/A596CH에 저장된 CH 번호를 MOVL 명령 등을 사용하여 인덱스 레지스터(IR)에 전송하는 과정이 필요합니다.

조건 플래그

명령의 실행이 완료될 때 조건 플래그는 갱신되지 않습니다. 조건 플래그를 참조하고 싶은 경우, 조건 플래그의 동작과 동일한 다른 명령이 있으면 해당 명령을 사용하여 조건 플래그를 조작하고 그 결과를 참조하십시오.

인덱스 레지스터(IR00)에 출력하는 경우

SRCH(데이터 검색) 명령에서 일치하는 데이터가 있는 경우, 데이터가 존재하는 CH(여러 개 있는 경우는 하위 CH)의 I/O 메모리 실제 주소를 인덱스 레지스터(IR00)에 출력하지 않는 대신 특수 보조 릴레이 A595/596CH에 출력합니다.

데이터 레지스터(DR) 출력을 지정하는 경우

SRCH(데이터 검색) 명령에서 데이터 레지스터(DR)에 일치하는 수의 데이터를 출력하도록 지정한 경우, 일치하는 수의 데이터를 데이터 레지스터(DR)에 출력하지 않고 대신 특수 보조 릴레이 A597CH에 출력합니다.

일치하는 데이터가 존재하는 경우

SRCH(데이터 검색) 명령에서 일치하는 데이터가 존재하는 경우 = 플래그를 1(ON)로 하지 않고 대신 특수 보조 릴레이 A598.01을 1(ON)로 합니다.

백그라운드 처리 중 명령 에러 발생 시의 동작

백그라운드 처리 중에 명령 처리 에러 또는 잘못된 영역 액세스 에러가 발생한 경우, ER/AER 플래그가 1(ON)이 되지 않는 대신 특수 보조 릴레이 A395.10이 1(ON)로 됩니다. 다음과 같은 백그라운드 처리를 실행하기 시작할 때까지 1(ON)이 그대로 유지됩니다.

출력 대상 CH 번호에 데이터 레지스터(DR0~15)를 지정한 경우

최대값 검색(MAX/MAXL/MAXF/MAXD) 명령 또는 최소값 검색(MIN/MINL/MINF/MIND) 명령에서 최대값 또는 최소값 출력 CH 번호에 데이터 레지스터(DR0~15)를 지정할 수 없습니다. 지정하는 경우에는 명령 처리 에러가 발생하고 ER 플래그는 1(ON)이 됩니다.

● 관련 특수 보조 릴레이

명칭	주소	내용
네트워크 통신 명령 실행 가능 플래그	A202.00~A202.07	네트워크 통신 명령 또는 데이터 백그라운드 처리를 통해 해당 내부 논리 포트 번호를 사용할 수 있을 때 1(ON)이 됩니다. 각 비트가 내부 논리 포트 번호에 대응하며, 비트 00~07: 내부 논리 포트 번호 0~7입니다. (*1)
네트워크 통신 명령 실행 에러 플래그	A219.00~A219.07	네트워크 통신 명령 실행 시 에러가 발생하면 1(ON)이 됩니다. 정상적으로 종료되면 0(OFF)으로 되돌아갑니다. 각 비트는 내부 논리 포트 번호에 대응하며, 비트 00~07: 내부 논리 포트 번호 0~7입니다. 즉, 비트 00~07이 내부 논리 포트 번호 0~7에 대응합니다. (*2)
네트워크 통신 종료 코드	A203~A210CH	네트워크 통신 명령이 실행된 경우 응답 코드가 저장되며, 백그라운드 처리가 종료되면 클리어(0000Hex 저장)됩니다. 각 CH가 포트 번호에 대응하며 A203~210CH: 내부 논리 포트 번호 0~7입니다. (*3)
백그라운드 처리 중 ER/AER 플래그	A395.10	백그라운드 처리 중에 명령 에러 또는 잘못된 영역 액세스 에러가 발생한 경우, 이 플래그가 1(ON)이 됩니다. 전원 ON 시 또는 운전을 시작하면 0(OFF)으로 되돌아갑니다.
백그라운드 처리 시 IR00 출력	A595~A596CH	인덱스 레지스터 출력을 지정한 명령을 백그라운드 처리로 실행한 경우, 결과를 이 영역에 출력하며 IR00에는 출력하지 않습니다. 00000000~FFFFFFFFHex 하위 4자리: A595CH, 상위 4자리: A596CH
백그라운드 처리 시 DR00 출력	A597CH	데이터 레지스터 출력을 지정한 명령을 백그라운드 처리로 실행한 경우, 결과를 이 영역에 출력하며 DR00에는 출력하지 않습니다. 0000~FFFFHex
백그라운드 처리 시 = 플래그 출력	A59801	SRCH 명령을 백그라운드 처리로 실행한 경우 데이터를 검색한 결과, 일치하는 데이터가 있을 때 이 플래그가 1(ON)이 됩니다.

*1 간이 백업 조작으로 메모리 카드에 쓰거나 조회할 때는 빈 내부 논리 포트 번호가 자동으로 사용되며, 이 플래그에 대응하는 비트가 실행 시 0(OFF)이 되고 종료 시 1(ON)이 됩니다.

*2 간이 백업 조작으로 메모리 카드에 쓰거나 조회할 때는 빈 내부 논리 포트 번호가 자동으로 사용되며, 대응하는 비트가 이상 종료 시 1(ON)이 되고 정상 종료 시 0(OFF)이 됩니다.

*3 간이 백업 조작으로 메모리 카드에 쓰거나 조회할 때는 빈 내부 논리 포트 번호가 자동으로 사용되며, 대응하는 CH에 응답 코드가 저장됩니다.

참고

이 백그라운드 처리는 다음과 같은 명령에서도 사용 중인 CPU 유니트 내의 내부 논리 포트를 사용합니다.

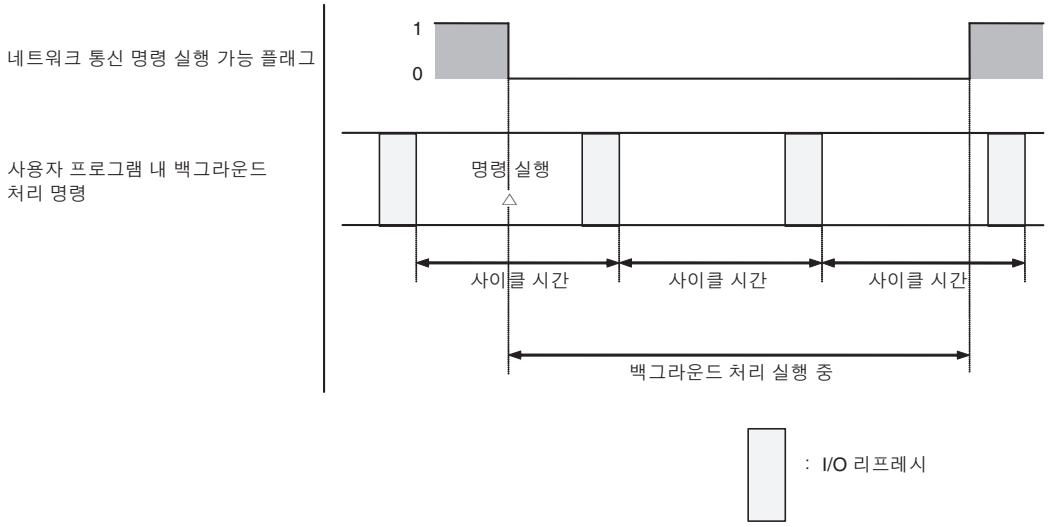
- 네트워크 통신 명령(SEND/RECV/CMND 명령 등)
- 프로토콜 매크로 명령(PMCR 명령 등)
- 시리얼 커뮤니케이션 유니트를 사용한 무수순 명령(TXDU/RXDU 명령)

따라서 백그라운드 처리에서 사용하는 내부 논리 포트 번호가 이들 명령에서 지정하는 내부 논리 포트 번호와 중복되는 경우, 백그라운드 처리 명령과 이들 명령은 동시에 동작할 수 없습니다. 따라서 네트워크 통신 명령 실행 가능 플래그를 사용하여 배타적으로 제어해야 합니다.

(주) 네트워크 통신 명령 실행 가능 플래그가 0(OFF)일 때 백그라운드 처리 명령을 실행하려고 하면 ER 플래그가 1(ON)이 됩니다. 이런 경우, NOP 처리가 되고 해당 명령은 실행되지 않습니다.

● 네트워크 통신 명령 실행 가능 플래그의 동작

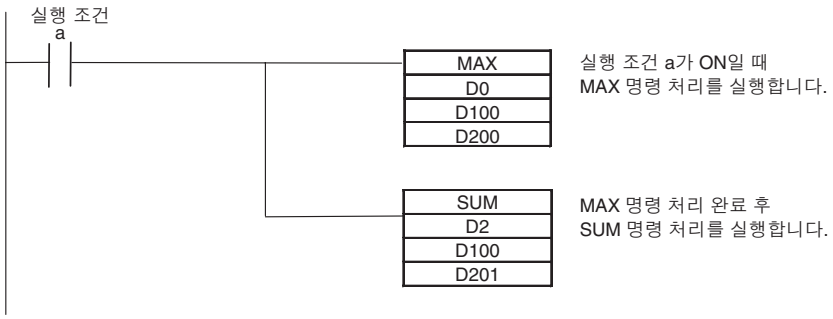
네트워크 통신 명령 실행 가능 플래그는 평상시(처리를 실행하지 않을 때)는 1(ON), 처리 실행 중에는 0(OFF)이 됩니다.



래더 프로그램 예 1

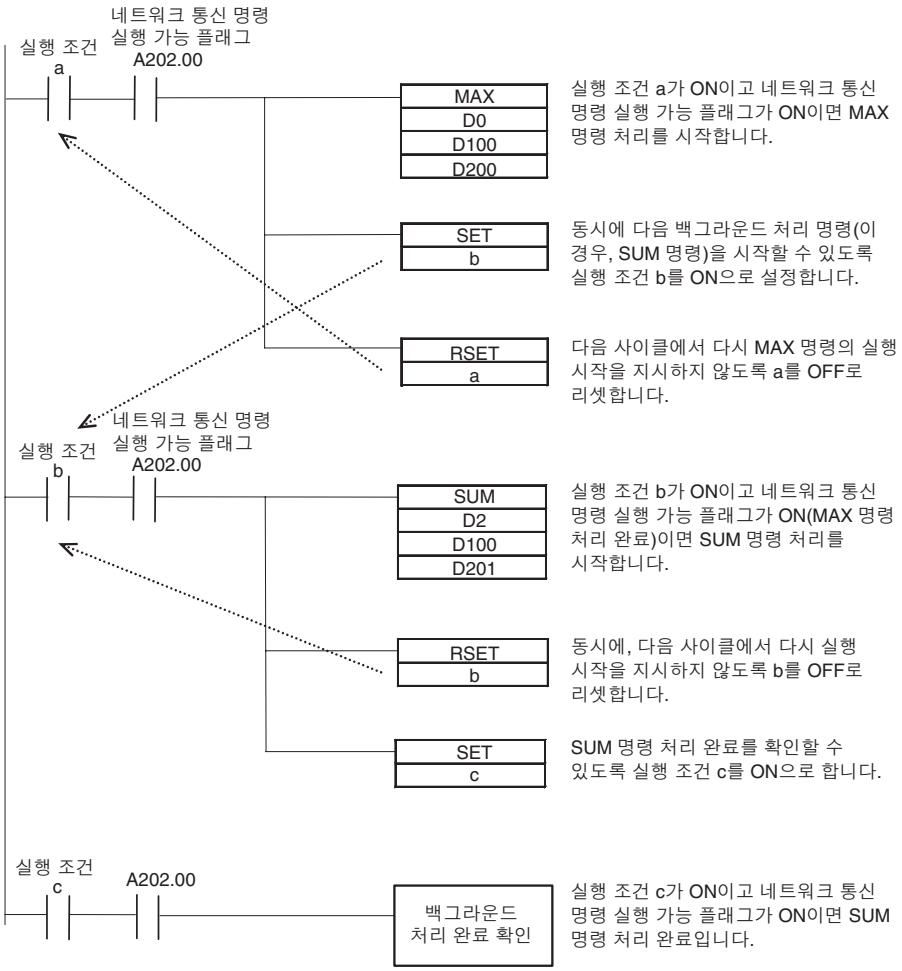
● 백그라운드로 처리하지 않는 경우

명령을 실행하면 처리가 완료됩니다.



● 백그라운드로 처리하는 경우

사용하는 내부 논리 포트 번호에 대응한 네트워크 통신 실행 가능 플래그가 ON일 때(다른 백그라운드 처리 및 네트워크 관련 처리가 실행되지 않았을 때) 백그라운드 처리를 시작합니다.
또한 백그라운드 처리 명령 간의 실행 순서를 보증하기 위해 입력 조건을 SET/RSET 명령 등으로 제어합니다.
(백그라운드 처리에서 사용하는 내부 논리 포트 번호를 0으로 한 경우)

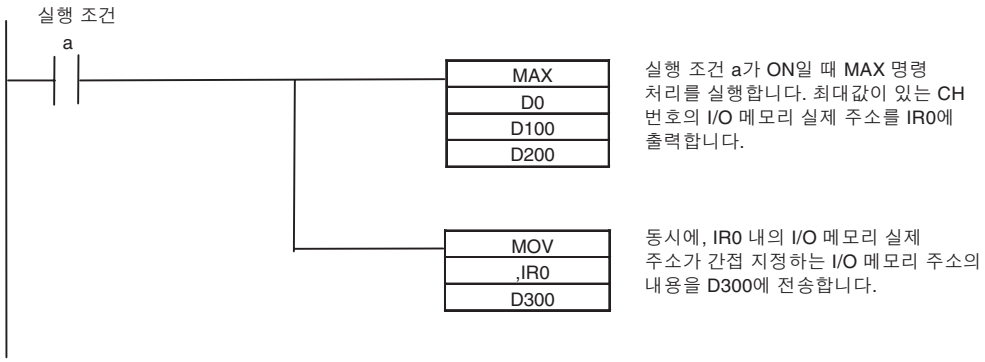


래더 프로그램 예 2

특히, 인덱스 레지스터 출력을 지정한 경우(MAX, MIN 또는 SRCH 명령만 해당)

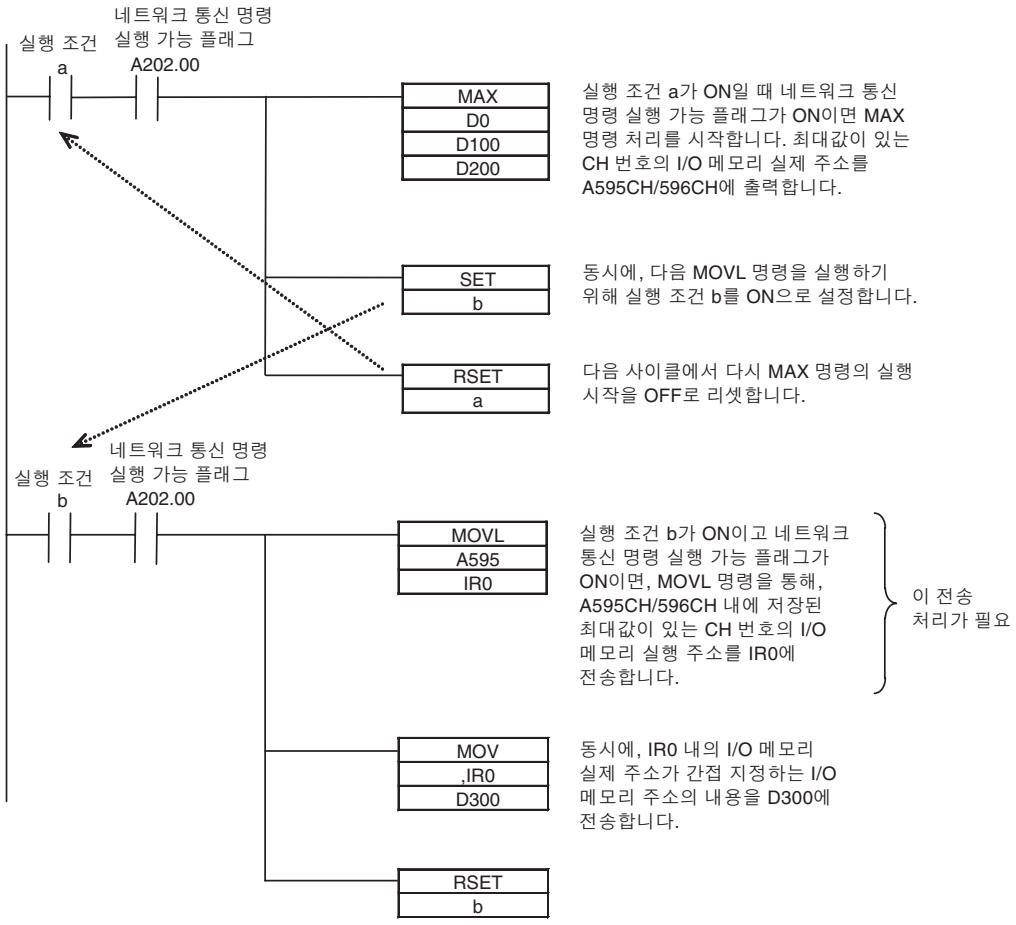
● 백그라운드로 처리하지 않는 경우

다음과 같이 인덱스 레지스터(IR)에 최대값 또는 최소값이 있는 CH 번호의 I/O 메모리 실제 주소를 저장합니다.



● 백그라운드로 처리하는 경우

특수 보조 릴레이 A595CH/596CH에 최대값 또는 최소값이 있는 CH 번호의 I/O 메모리 실제 주소를 저장합니다. 일단 A595CH/A596CH에 저장된 CH 번호를 MOVL 명령 등을 사용하여 인덱스 레지스터(IR)에 전송하는 과정이 필요합니다.

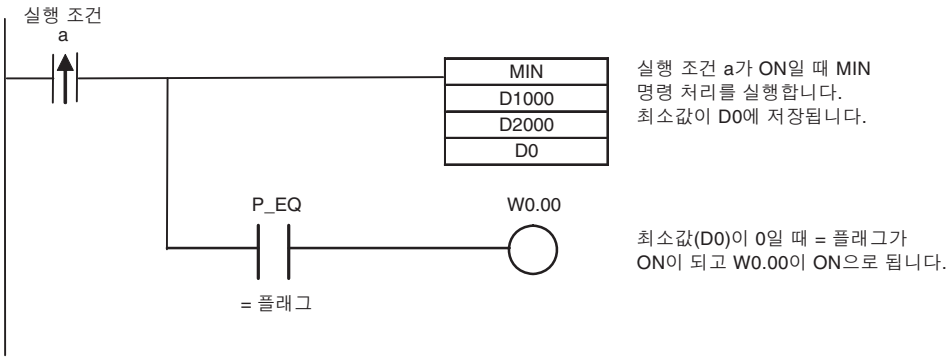


래더 프로그램 예 3

특히, 조건 플래그를 참조하는 경우

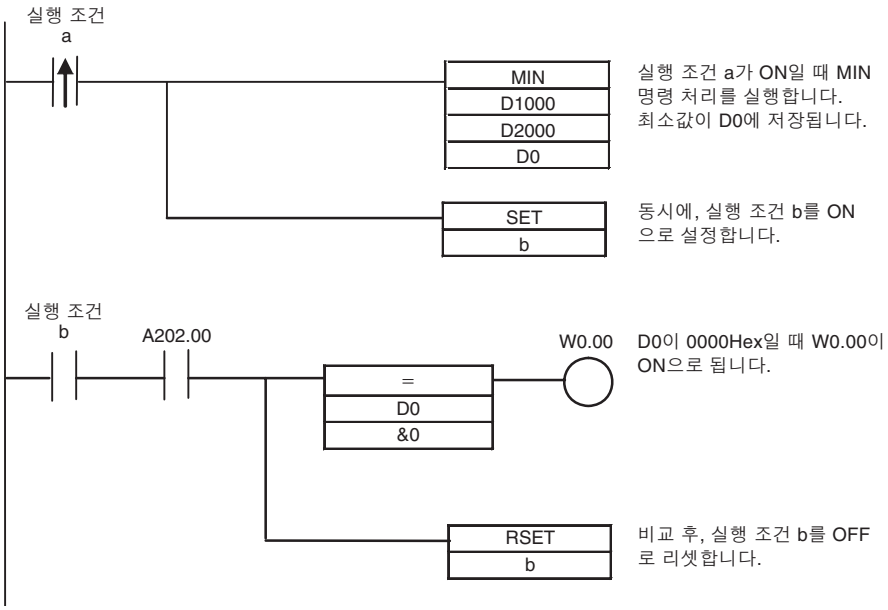
● 백그라운드로 처리하지 않는 경우

MIN 명령으로 검색한 최소값이 0인지 여부를 조사하고 싶으면 다음과 같이 MIN 명령 직후에 = 플래그를 참조합니다.



● 백그라운드로 처리하는 경우

MIN 명령으로 검색한 최소값이 0인지 여부를 조사하고 싶으면 다음과 같이 비교(=) 명령을 사용합니다.



10-2-6 고속 인터럽트 기능

고속 인터럽트 기능은 일정한 제약 조건에서 인터럽트 태스크의 실행 성능을 향상시키는 기능입니다.
CJ2H CPU 유니트의 유니트 Ver. 1.1 이상에서 사용 가능합니다.

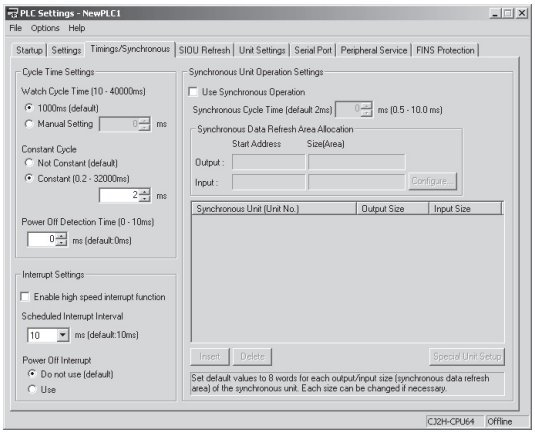
인터럽트 오버헤드 시간 단축

I/O 인터럽트 태스크/외부 인터럽트 태스크, 예약 인터럽트 태스크를 실행할 때의 인터럽트 오버헤드 시간(인터럽트 태스크 기동 시간+사이클릭 태스크로의 복귀 시간)을 다음과 같이 단축합니다.

인터럽트 태스크 유형	인터럽트 오버헤드 시간	
	고속 인터럽트 기능을 사용할 때	고속 인터럽트 기능을 사용하지 않을 때(평상시)
I/O 인터럽트 태스크/외부 인터럽트 태스크	25 μs (인터럽트 태스크 기동 시간 17 μs+ 사이클릭 태스크로 복귀하는 시간 8 μs)	37 μs (인터럽트 태스크 기동 시간 26 μs+ 사이클릭 태스크로 복귀하는 시간 11 μs)
예약 인터럽트 태스크	21 μs (인터럽트 태스크 기동 시간 13 μs+ 사이클릭 태스크로 복귀하는 시간 8 μs)	33 μs (인터럽트 태스크 기동 시간 22 μs+ 사이클릭 태스크로 복귀하는 시간 11 μs)

- **PLC 시스템 설정**

  CX-Programmer에서 '타이밍/동기' 탭의 '고속 인터럽트 기능 사용'으로 설정합니다.



● 고속 인터럽트 기능 사용 시의 제한 사항

'고속 인터럽트 기능'을 사용할 때는 다음과 같은 제한이 있습니다.

- 다음과 같은 명령(네트워크 통신용 명령 등)을 인터럽트 태스크 내에서 사용할 수 없습니다.

니모닉	명령 이름
EMBC	확장 DM 뱅크 전환
SEND (*)	네트워크 송신
RECV (*)	네트워크 수신
CMND (*)	명령 송신
PMCR (*)	프로토콜 매크로
TXDU	시리얼 커뮤니케이션 유니트의 시리얼 포트 출력
RXDU	시리얼 커뮤니케이션 유니트의 시리얼 포트 입력
EXPLT, EGATR, ESATR, ECHRD, ECHWR	범용 Explicit 메시지 송신, Explicit 읽기 명령(EGATR), Explicit 쓰기 명령(ESATR), Explicit CPU 유니트 데이터 읽기 명령(ECHRD), Explicit CPU 유니트 데이터 쓰기 명령(ECHWR)

* SEND2, RECV2, CMND2, PMCR2 명령으로 대체 가능

- 백그라운드 처리 설정을 유효화한 명령은 인터럽트 태스크 내에서 사용할 수 없습니다(테이블 데이터 처리 명령, 문자열 처리 명령, 데이터 이동 명령).
- 다음과 같은 특수 보조 릴레이가 무효가 됩니다.
  A440CH(인터럽트 태스크 최대값 처리 시간), A441CH(최대 처리 시간 인터럽트 태스크 번호)
- 고속 인터럽트 기능을 사용하면 다음과 같이 내장 EtherNet/IP 및 EtherNet/IP 유니트와의 리프레시 성능이 저하됩니다.

	고속 인터럽트 기능을 사용한 경우	평상시
공통 처리 시간에 대한 가산 시간 (데이터 링크 사용 시)	100 μs+전송 CH×0.87 μs	100 μs+전송 CH×0.33 μs
EtherNet/IP에 대한 DLNK 명령 실행 시간 (CPU 고기능 유니트 매회 I/O 리프레시)	910 μs/1000CH	430 μs/1000CH

예약 인터럽트 태스크의 최소 시간 간격

예약 인터럽트 0(인터럽트 태스크 번호 2)에 대해서만 예약 인터럽트의 최소 시간 간격으로 0.1ms를 설정(MSKS 명령 사용)할 수 있습니다.

● 예약 인터럽트 0에 최소 시간 간격 0.1ms를 설정하는 경우의 조건

1 **고속 인터럽트 기능을 사용(PLC 시스템 설정의 '고속 인터럽트 기능 사용' 유효)합니다.**

고속 인터럽트 기능을 사용하지 않는 경우 MSKS 명령에서 0.1ms를 지정하면 명령 처리 에러가 됩니다.

2 **CPU 유니트의 페리페럴(USB) 포트, 시리얼 포트에 기기를 직접 연결하지 않습니다.**

주변 툴(CX-Programmer)이나 터치 패널 NS 등을 CPU 유니트의 페리페럴(USB) 포트, 시리얼 포트에 직접 연결하면 0.1ms의 시간 간격이 보장되지 않습니다. 연결해야 하는 경우에는 내장 EtherNet/IP 포트 또는 EtherNet/IP 유니트, 시리얼 커뮤니케이션 유니트의 통신 포트를 경유하여 연결하십시오.

사용상의 주의 사항

고속 인터럽트 기능은 일정한 제약 조건에서 인터럽트 태스크의 실행 성능을 향상시키는 기능입니다. 사용 가능한 조건을 충분히 확인한 후 사용하십시오. 특히, 예약 인터럽트의 최소 시간 간격을 0.1ms로 설정하여 동작시키는 경우, CPU 유니트의 페리페럴(USB) 포트, 시리얼 포트에 주변 툴(CX-Programmer)이나 터치 패널 NS 등을 직접 연결하지 않아야 합니다. 직접 연결하면 0.1ms의 시간 간격으로 동작하지 않을 가능성이 있습니다. 보수 목적 등으로 주변 툴(CX-Programmer)을 직접 CPU 유니트에 연결하는 경우는 0.1ms의 시간 간격으로 동작하지 않아도 설비에 영향이 없다는 것을 확인한 후 실시하십시오.

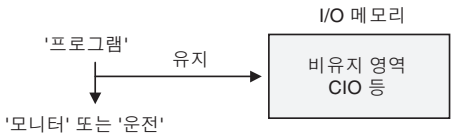
10-3 운전 시작 시 동작 설정 및 보수 기능

10-3-1 동작 모드 변경 및 전원 ON 시의 유지 설정 기능

동작 모드 변경 시

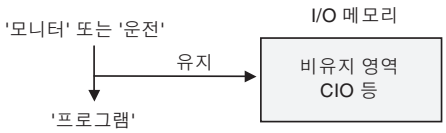
- **운전 시작 시**

운전을 시작할 때('프로그램'→'모니터' 또는 '운전' 모드) I/O 메모리의 비유지 영역 데이터를 그대로 유지한 상태로 프로그램을 시작하고 싶은 경우, 특수 보조 릴레이의 IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정합니다.



- **운전 정지 시**

운전을 정지할 때('모니터' 또는 '운전' 모드→'프로그램'으로 모드 변경 시)도 I/O 메모리의 비유지 영역 데이터를 유지하고 싶은 경우, 특수 보조 릴레이의 IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정합니다.



I/O 메모리의 비유지 영역: CIO(입출력 릴레이, 데이터 링크 릴레이, CPU 고기능 유니트 릴레이, 고기능 I/O 유니트 릴레이, DeviceNet 릴레이, 내부 보조 릴레이), 내부 보조 릴레이(WR), 타이머 플래그, 타이머 현재값

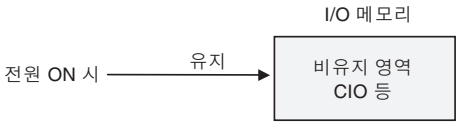
- **관련 특수 보조 릴레이**

명칭	주소	내용
IOM(I/O 메모리) 유지 비트	A500.12	CPU 유니트의 동작 모드 전환(프로그램←→운전/모니터) 또는 전원 ON 시, I/O 메모리 영역의 유지 여부를 지정합니다. 1: 운전 모드 변경(프로그램←→모니터 또는 운전 모드) 시, I/O 메모리 유지 0: 운전 모드 변경 시, I/O 메모리 클리어

따라서 이 플래그를 1(ON)로 설정하면 운전을 정지했을 때 직전 출력 유니트의 출력 상태를 유지 및 기억합니다.
그리고 운전을 시작할 때 직전의 출력 상태를 유지하고 명령을 실행합니다. (이 플래그를 0(OFF)으로 설정하면 일단 명령 실행 전에 출력을 클리어한 후 명령을 실행합니다.)

전원 ON 시

전원 투입(OFF→ON) 시에도 I/O 메모리의 비유지 영역 데이터를 유지하고 싶으면 특수 보조 릴레이의 IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 하고, PLC 시스템 설정의 전원 ON 시 IOM(I/O 메모리) 유지 비트를 유지로 설정합니다.

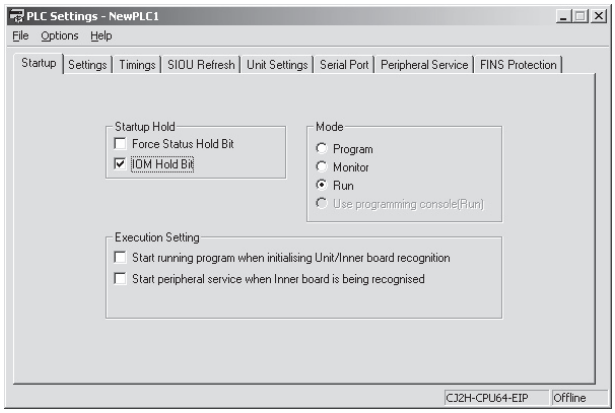


● 관련 특수 보조 릴레이

명칭	주소	내용
IOM(I/O 메모리) 유지 비트	A500.12	CPU 유니트의 동작 모드 전환(프로그램←→운전/모니터) 또는 전원 ON 시, I/O 메모리 영역의 유지 여부를 지정합니다. 1: 운전 모드 변경(프로그램←→모니터 또는 운전 모드) 시, I/O 메모리 유지 0: 운전 모드 변경 시, I/O 메모리 클리어

● PLC 시스템 설정

CX-Programmer의 '시작' 탭에 있는 '시작 대기'에서 'IOM(I/O 메모리) 유지 비트'를 선택하십시오.

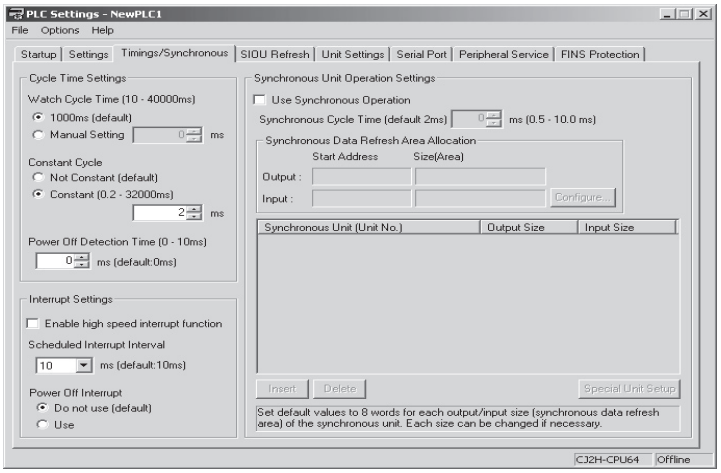


10-3-2 전원 OFF 감지 지연 시간 설정

전원 전압이 최저 정격 전압의 85% 이하(DC 전원은 80% 이하)가 되면 전원 OFF를 확정할 때까지의 시간을 지연시킬 수 있습니다.
기본 전원 OFF 감지 지연 시간 0ms에서는 10ms 이상(DC 전원은 2ms 이상)의 전원 OFF에 대해 정격 전압의 85%(DC 전원은 80%)가 된 후 10~25ms(DC 전원은 2~5ms) 후에 운전을 정지하며, 전원 OFF 인터럽트 태스크가 있을 때는 전원 OFF 인터럽트 태스크를 실행하기 시작합니다. 이 타이밍을 전원 OFF 감지 지연 시간 설정에서 그리고 지정된 시간으로 연장할 수 있습니다.

● PLC 시스템 설정

CX-Programmer에서 '타이밍/동기' 탭의 '전원 OFF 감지 지연 시간'에서 설정하십시오.



참고

전원 OFF 인터럽트 태스크를 사용하지 않을 때는 순간 정전 대책으로 전원 OFF 감지 지연 시간을 최대값인 10ms로 설정하기 바랍니다. 단, 전원 유니트 CJ1W-PD022를 사용하는 경우는 연장을 설정할 수 없으므로 초기값 0ms를 설정하십시오.

10-3-3 전원 OFF 금지 구간 설정

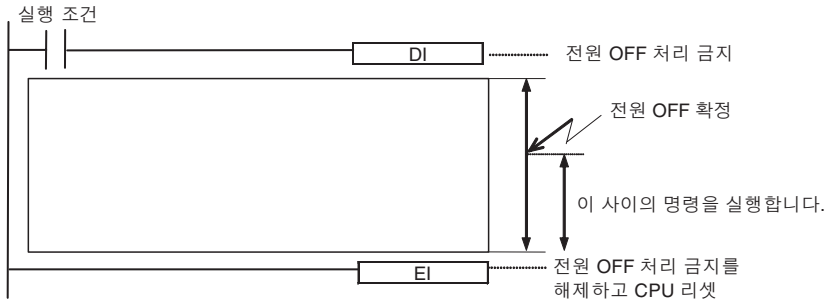
전원 OFF가 확정되어도 특정 영역의 명령(DI 명령－EI 명령 간의 명령)은 모두 실행한 후 CPU 유니트가 리셋되도록 지정할 수 있습니다.
특정 명령군에서 데이터 처리를 하는 경우, 해당 명령군을 실행하는 도중에 처리를 중단했다가 다음에 전원을 켰을 때 중간에 저장한 데이터부터 시작하고 싶지 않은 경우 등에 이 전원 OFF 금지 구간 설정을 사용합니다.

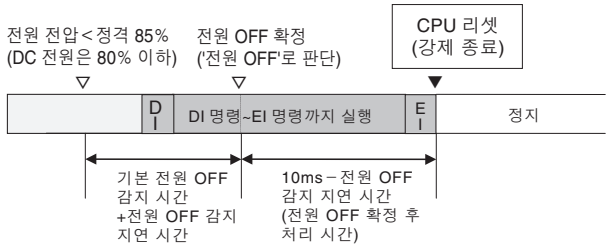
사용 방법

1 **A530CH(전원 OFF 금지 구간 설정)에 A5A5Hex(전원 OFF 금지 구간 설정 유효)를 설정합니다.**

2 **PLC 시스템 설정에서 전원 OFF 금지 구간 설정 무효(초기값)로 설정합니다.**

3 **처리를 중단하고 싶지 않은 명령군 앞에서 DI 명령을 실행하고 전원 OFF 처리를 금지합니다. 또한 처리 완료 후 EI 명령을 실행하고 전원 OFF 처리 금지를 해제합니다.**
이렇게 하면 DI 명령－EI 명령 간에 전원 OFF가 확정되어도 해당 영역 내의 명령을 모두 실행한 후 CPU 유니트를 리셋합니다.





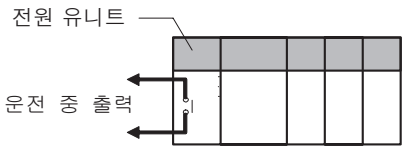
● 관련 특수 보조 릴레이

명칭	주소	내용
전원 OFF 금지 구간 설정	A530CH	DI－EI 명령 사이는 전원 OFF 처리(전원 OFF 인터럽트 태스크 실행은 제외)를 금지하고 EI 명령까지 실행합니다. A5A5Hex: 전원 OFF 금지 구간 설정 유효 A5A5Hex 이외: 전원 OFF 금지 구간 설정 무효

10-3-4 운전 중 출력 기능

운전 중 출력 기능

CJ 시리즈에서 사용할 수 있는 전원 유니트에는 운전 중 출력 기능이 있는 경우가 있습니다(CJ1W-PA205R). 이 출력 접점은 CPU 유니트가 운전 또는 모니터 모드일 때 ON이 됩니다.



명칭	동작
운전 중 출력	모니터 또는 운전 모드: ON(폐쇄) 프로그램 모드: OFF(개방)

PLC의 외부 회로를 구성하는 경우, 예를 들어, 페일 세이프 대책으로 운전 중 출력을 사용하고 이 접점이 ON인 경우만 출력 유니트의 외부 전원이 들어오도록 기동 및 정지 회로를 구성하십시오.

(주) 운전 중 출력 접점이 없는 전원 유니트를 사용하는 경우는 상시 ON 플래그를 입력 조건으로 한 출력 접점을 출력 유니트에서 확보하십시오.

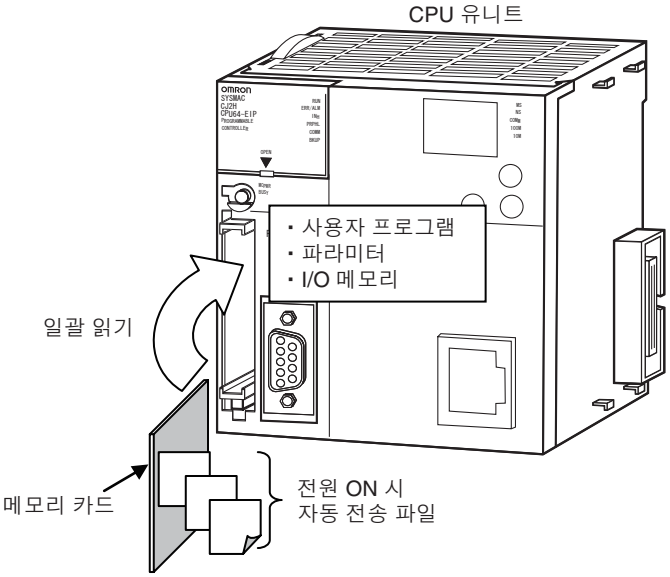
안전상의 중요 사항

출력 유니트의 외부 전원을 켠 후 PLC 본체 전원을 ON으로 하면 PLC 본체 전원을 켰을 때 출력 유니트가 순간적으로 오동작할 수 있습니다. 이를 피하려면 PLC 본체 전원을 ON으로 한 후 출력 유니트의 외부 전원을 ON으로 해야 합니다. 또한 PLC가 운전 중이 아닐 때 외부 부하가 작동하지 않도록 하기 위해 위의 운전 중 출력을 사용하고, PLC가 운전 또는 모니터 모드로 이동했을 때 출력 유니트의 외부 전원을 ON으로 하는 페일 세이프 회로를 구성하십시오.

10-3-5 전원 ON 시 자동 전송 기능

개요

전원을 켰을 때 메모리 카드에 저장된 사용자 프로그램, 파라미터 데이터, I/O 메모리를 CPU 유니트로 일괄적으로 읽어올 수 있습니다.
또한 전원 ON 시 자동 기동 전송용 파일은 CX-Programmer의 메모리 카드 창에서 작성하여 메모리 카드에 저장할 수 있습니다.



(주) EM 파일 메모리에서는 읽어올 수 없습니다.

참고

배터리 없이 운전할 때는 I/O 메모리가 불안정해지므로 이 기능을 사용하여 I/O 메모리의 값을 고정시킬 수 있습니다.

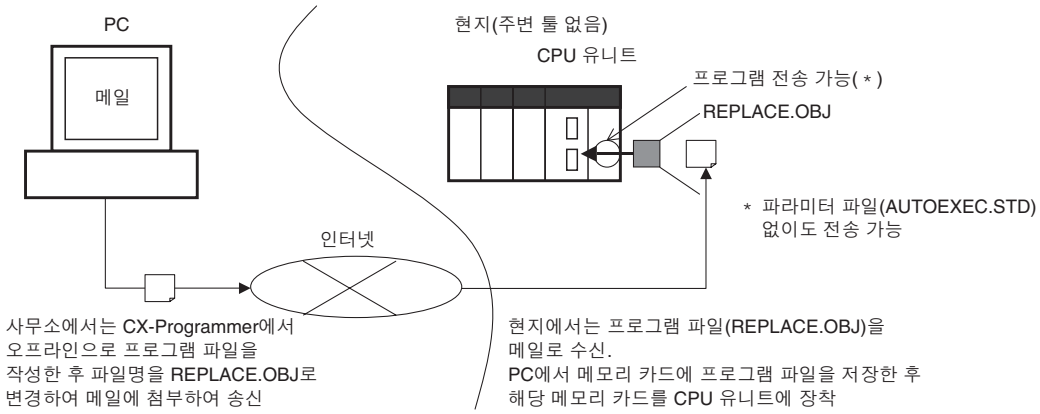
전원 ON 시 자동 전송의 종류

전원 ON 시 자동 전송에는 파라미터 파일(AUTOEXEC.STD)을 첨부하여 전송하는 경우와 파라미터 파일(AUTOEXEC.STD) 없이 전송하는 경우의 2종류가 있습니다.
이 두 종류는 각각 작성해야 하는 파일명이 다릅니다.

전원 ON 시 자동 전송의 종류	파일명	용도
파라미터 파일을 첨부하여 전송	AUTOEXEC 또는 ATEXEC□□로 고정	프로그램 및 네트워크 설정을 변경하여 교체
파라미터 파일 없이 전송	REPLACE 또는 REPLC□□로 고정	프로그램을 변경하여 교체

● 파라미터 파일 없이 전송하고 싶은 경우의 사용 예

오프라인에서(실제 기기 없이) 작성한 프로그램/네트워크 변수 파일(.OBJ)만 있고 파라미터 파일(.STD)이 없을 때
예를 들어, 사무소에는 프로그램/네트워크 변수 파일(.OBJ)만 첨부 메일로 현지에 송신하고, 현지에서는 주변 툴 없이 메모리 카드에 프로그램/네트워크 변수 파일을 저장하여 CPU 유니트에 해당 프로그램을 전송합니다.



순서

● 전원 ON 시 자동 전송용 파일 작성 및 전송

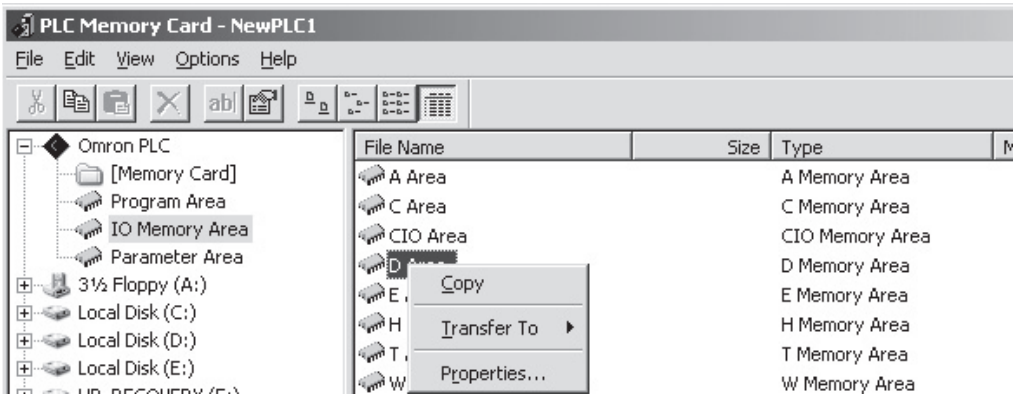
CX-Programmer에서 전원 ON 시 자동 전송 파일을 작성하여 CPU 유니트에 장착된 메모리 카드로 전송합니다.

프로그램 영역, 파라미터 영역:

마우스 오른쪽 버튼을 클릭하여 [전송 대상]−[메모리 카드]를 선택하고, 전원 ON 시 자동 전송용 파일명을 입력하여 작성 및 전송합니다.

I/O 메모리 영역:

오른쪽 보기에 표시된 D 영역 또는 E 영역을 마우스 오른쪽 버튼으로 클릭하고 전원 ON 시 자동 전송용 파일명을 입력하여 작성 및 전송합니다.

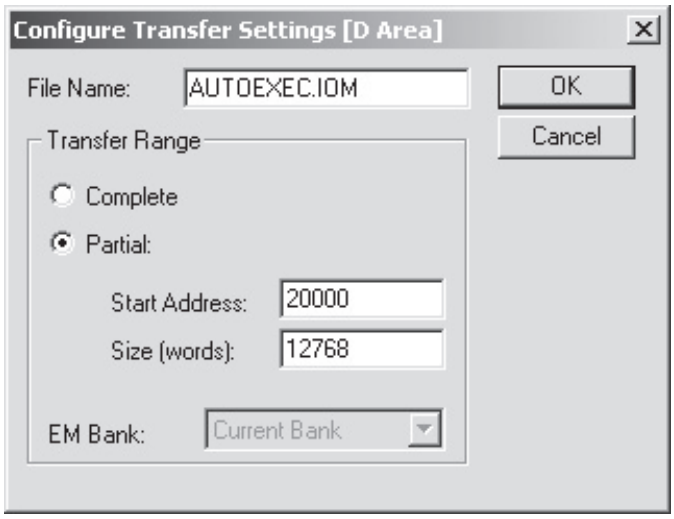


전원 ON 시 자동 전송용 파일명

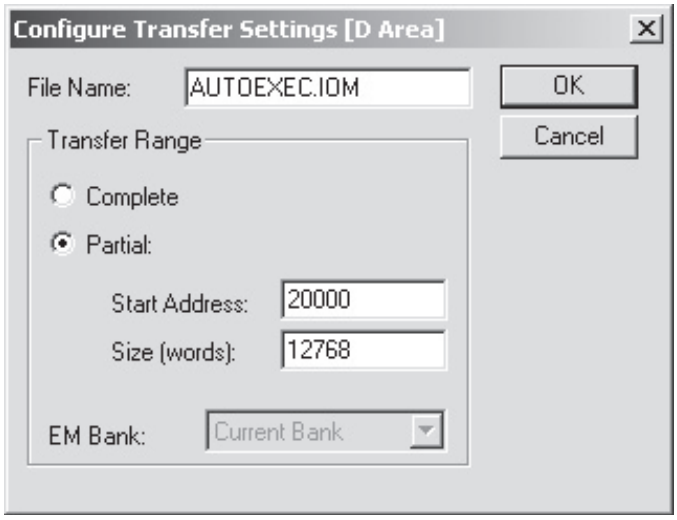
파일 종류	내용	파라미터 파일 첨부 전원 ON 시 자동 전송	필수	파라미터 파일 없이 전원 ON 시 자동 전송	필수
프로그램/ 네트워크 변수 파일	사용자 프로그램	AUTOEXEC.OBJ	○	REPLACE.OBJ	○
파라미터 파일	• PLC 이름 • PLC 시스템 설정 • 등록 I/O 테이블 • 라우팅 테이블 • CPU 고기능 유니트 시스템 설정 영역 등	AUTOEXEC.STD	○	없음	—
데이터 파일	고기능 I/O 유니트/CPU 고기능 유니트의 할당 DM 영역 (D20000~D32767)	AUTOEXEC.IOM	—	REPLACE.IOM	—
	DM(D0~D19999)	ATEXECDM.IOM	—	REPLACDM.IOM	—
	EM	AUTOEM□□.IOM (□□: 00~18)	—	RPLCEM□□.IOM (□□: 00~18)	—

예) D 영역의 경우 다음과 같이 시작 주소와 크기를 지정합니다.

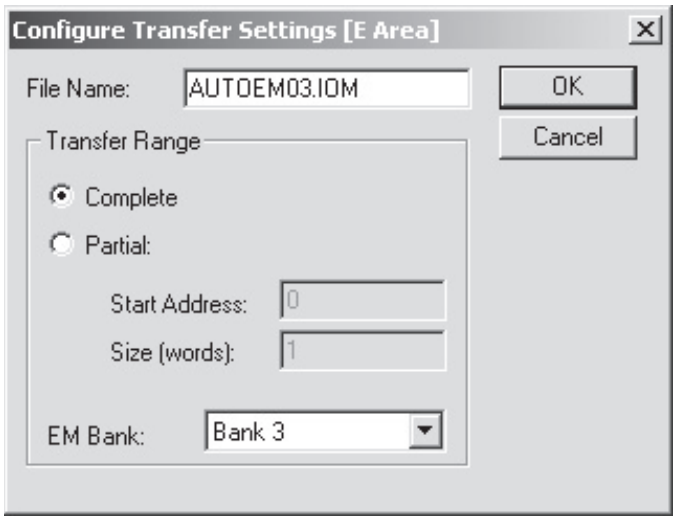
- 고기능 I/O 유니트/CPU 고기능 유니트의 할당 DM 영역:



- DM:



- EM:



● 전원 ON 시 자동 전송 실행

1 **PLC 본체의 전원을 OFF로 합니다.**

2 **CPU 유니트 전면의 딥 스위치 SW2를 ON으로 합니다.**
딥 스위치의 SW7이 ON인 경우 간이 백업 기능이 동작하므로 OFF로 설정하십시오.

3 **필요한 파일이 미리 저장된 메모리 카드를 CPU 유니트에 장착합니다.**

4 **PLC 본체의 전원을 ON으로 합니다.**

5 **전원이 켜지면 자동 전송이 시작되고 BUSY LED가 황색으로 점멸합니다. BUSY LED가 꺼진 후 종료합니다.**
전원 ON 시 자동 전송에 실패하면 카드 전송 이상이 발생하고 CPU 유니트의 운전이 정지됩니다.

CPU 유니트 전면 딥 스위치

스위치 번호	명칭	설정
SW2	전원 ON 시 자동 전송 실행/비실행	ON: 전원 ON 시 자동 전송 실행 OFF: 실행 안 함

관련 특수 보조 릴레이

명칭	주소	내용
카드 전송 이상 플래그 (운전 정지 이상)	A401.03	메모리 카드에서 전원 ON 시 자동 전송에 실패한 경우 1(ON)이 됩니다. CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다.

전원 ON 시 자동 전송용 데이터 파일

전원 ON 시 자동 전송용 데이터 파일에는 다음과 같이 6종류가 있습니다.

전원 ON 시 자동 전송용 데이터 파일		내용	범위	시작 주소	크기
파라미터 파일 첨부 전원 ON 시 자동 전송	파라미터 파일 없이 전원 ON 시 자동 전송				
AUTOEXEC.IOM	REPLACE.IOM	고기능 I/O 유니트/CPU 고기능 유니트의 할당 DM 영역 (D20000~D32767)	일부	20000	12768
ATEXECDM.IOM	REPLACDM.IOM	DM(D00000~D19999)	일부	0	20000
AUTOEM□□.IOM (□□: 00~18)	RPLCEM□□.IOM (□□: 00~18)	EM	모두	—	—

각 전원 ON 시 자동 전송용 데이터 파일을 작성하는 경우, 반드시 각각 위의 영역 종류 및 시작 주소를 선두로 하십시오. 저장 크기는 해당 영역 종류의 최종 주소까지의 크기 이하로 설정하십시오.
전원을 켜면 메모리 카드 내에 있는 데이터 파일의 모든 데이터가 각각 D20000 이후, D0 이후, E□_0 이후에 데이터 수만큼 전송됩니다.

파라미터 파일을 첨부하여 전원 ON 시 자동 전송하는 경우

다음과 같은 파일명을 지정합니다.
아래 표의 오른쪽 필수란에 ○ 표시된 파일은 전원 ON 시 자동 전송 기능을 실행하는 경우, 반드시 메모리 카드 내에 있어야 합니다.

파일 종류	파일명	확장자	내용	설명	필수
프로그램/네트워크 변수 파일	AUTOEXEC	.OBJ	사용자 프로그램 네트워크 변수(*1)	• CPU 유니트의 사이클릭 태스크와 인터럽트 태스크 내의 프로그램입니다. • 전원 ON 시 자동 전송 기능을 실행하는 경우 메모리 카드에 반드시 있어야 합니다. • 메모리 카드에 파라미터 파일(AUTOEXEC.STD)이 없으면 전송되지 않습니다.	○
파라미터 파일	AUTOEXEC	.STD	• PLC 이름 • PLC 시스템 설정 • I/O 테이블 • 라우팅 테이블 • CPU 고기능 유니트 시스템 설정 영역 등	• CPU 유니트 내부의 파라미터 영역의 데이터입니다. • 전원을 켜면 CPU 유니트의 특정 위치에 초기 설정 데이터가 자동으로 저장됩니다. • 사용자는 파일 내의 각 정보 종류별로 데이터를 식별할 필요가 없습니다. • 전원 ON 시 자동 전송 기능을 실행하는 경우 프로그램/네트워크 변수 파일이 AUTOEXEC.OBJ이면 메모리 카드에 반드시 있어야 합니다.	○
데이터 파일	AUTOEXEC	.IOM	I/O 메모리의 데이터 메모리(D20000을 선두로 하는 지정된 CH 수의 데이터 저장)	• 파일명이 AUTOEXEC이고 확장자가 .IOM인 파일에는 D20000을 선두로 한 DM 영역의 데이터를 저장하십시오. • 전원을 켜면 파일 내의 선두부터 모든 데이터가 D20000 이후에 데이터 수만큼 전송됩니다. • 전원 ON 시 자동 전송 기능을 실행하는 경우 메모리 카드에 없어도 됩니다.	—
	ATEXECDM	.IOM	I/O 메모리의 데이터 메모리(D0을 선두로 하는 지정된 CH 수의 데이터 저장)	• 파일명이 ATEXECDM이고 확장자가 .IOM인 파일에는 D0을 선두로 한 DM 영역의 데이터를 저장하십시오. • 전원을 켜면 파일 내의 선두부터 모든 데이터가 D0 이후에 데이터 수만큼 전송됩니다. • 전원 ON 시 자동 전송 기능을 실행하는 경우 메모리 카드에 없어도 됩니다. (*2)	—
	AUTOEM□□	.IOM	I/O 메모리의 뱅크 번호 □□의 확장 데이터 메모리(E□□_0을 선두로 하는 지정된 CH 수의 데이터 저장)	• 파일명이 AUTOEM□□이고 확장자가 .IOM인 파일에는 E□_0을 선두로 한 뱅크 번호 □의 EM 영역 데이터를 저장하십시오. (□: 뱅크 번호 0~C, 최대 뱅크 번호는 CPU 유니트 형식에 따라 다름.) • 전원을 켜면 파일 내의 선두부터 E□_0 이후에 데이터 수만큼 전송됩니다. • 전원 ON 시 자동 전송 기능을 실행하는 경우 메모리 카드에 없어도 됩니다.	—

*1 네트워크 변수는 CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당. CJ2H-CPU6□, CJ2M-CPU3□에 네트워크 변수가 포함된 변수 파일(.OBJ)이 전송되게 하면 A401.03(카드 전송 이상)이 ON이 됩니다.
*2 AUTOEXEC.IOM 파일과 DM 영역이 중복된 경우는 이 파일이 우선됩니다.

- AUTOEXEC.IOM 파일과 ATEXECDM.IOM 파일 영역이 중복된 경우(파일 읽기 순서가 AUTOEXEC.IOM 파일→ATEXECDM.IOM 파일 순서이기 때문에), ATEXECDM.IOM 파일이 우선됩니다(덮어 씀).

참고

전원 ON 시 자동 전송 기능과 특수 보조 릴레이에 의한 프로그램 전체 교체 기능을 조합해서 사용할 수 있습니다. 전원 ON 시 자동 전송을 통해 읽어온 프로그램 내에서 특수 보조 릴레이의 조작을 통해 다른 프로그램으로 교체할 수 있습니다.

파라미터 파일 없이 전원 ON 시 자동 전송하는 경우

다음과 같은 파일명을 지정합니다.
아래 표의 오른쪽 필수란에 ○ 표시된 파일은 전원 ON 시 자동 전송 기능을 실행하는 경우, 반드시 메모리 카드 내에 있어야 합니다.

파일 종류	파일명	확장자	내용	설명	필수
프로그램/네트워크 변수 파일	REPLACE	.OBJ	사용자 프로그램	• AUTOEXEC.OBJ와 내용이 동일합니다. • 메모리 카드에 파라미터 파일(AUTOEXEC.STD)이 없어도 전원을 켜면 전송됩니다.	○
파라미터 파일	필요 없음	—	—	• 메모리 카드에서 어떤 파일명을 사용하든 상관 없이 전송되지 않습니다.	—
데이터 파일	REPLACE	.IOM	I/O 메모리의 데이터 메모리(D20000을 선두로 하는 지정된 CH 수의 데이터 저장)	• AUTOEXEC.IOM과 내용이 동일합니다. • 메모리 카드에 프로그램/네트워크 변수 파일(REPLACE.OBJ)이 있으면 전원을 켤 때 전송됩니다.	—
	REPLCDM	.IOM	I/O 메모리의 데이터 메모리(D0을 선두로 하는 지정된 CH 수의 데이터 저장)	• ATEXECDM.IOM과 내용이 동일합니다. • 메모리 카드에 프로그램/네트워크 변수 파일(REPLACE.OBJ)이 있으면 전원을 켤 때 전송됩니다.	—
	RPLCEM□□	.IOM	I/O 메모리의 뱅크 번호 □의 확장 데이터 메모리(E□_0을 선두로 하는 지정된 CH 수의 데이터 저장)	• AUTOEM□□.IOM과 내용이 동일합니다. • 메모리 카드에 프로그램/네트워크 변수 파일(REPLACE.OBJ)이 있으면 전원을 켤 때 전송됩니다.	—

참고

- 고기능 I/O 유니트용 할당 DM 영역(D20000~29599), CPU 고기능 유니트용 할당 DM 영역(D30000~31599)의 데이터를 AUTOEXEC.IOM 파일로 메모리 카드에 저장해 두면 메모리 카드를 교체해서 각 고기능 I/O 유니트/CPU 고기능 유니트의 시스템 설정 데이터를 전원 ON 시 읽어올 수 있습니다. 이렇게 하면 메모리 카드를 사용해서 장치 및 머신별 고기능 I/O 유니트/CPU 고기능 유니트의 시스템 데이터를 라이브러리로 만들어 관리할 수 있습니다.

전원 ON 시 자동 전송에 필요한 시간의 기준

사용자 프로그램과 태그의 크기	자동 전송에 필요한 시간 (전원 ON부터 운전 시작까지의 시간)
사용자 프로그램 120K 스텝	9초
사용자 프로그램 120K 스텝＋태그 데이터 20,000개	59초

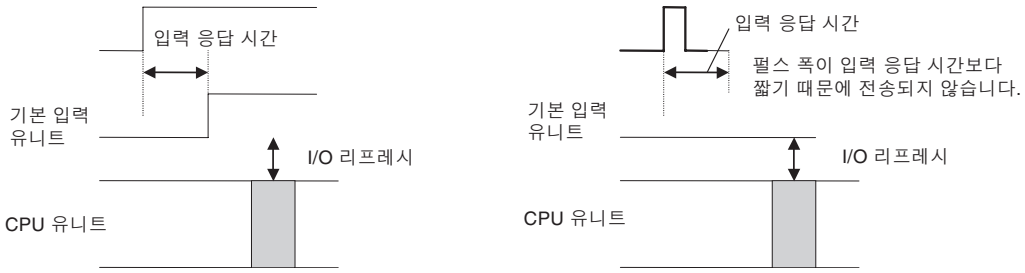
10-4 유니트 관련 관리 기능

10-4-1 기본 I/O 유니트 관련

● 기본 I/O 유니트의 입력 응답 시간 설정

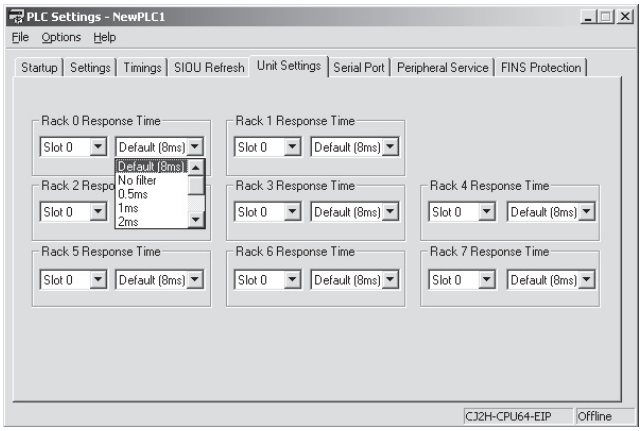
PLC 시스템 설정을 통해 기본 I/O 유니트의 입력 응답 시간을 랙 및 슬롯별로 설정할 수 있습니다. 이 응답 시간을 길게 설정하면 입력 접점의 채터링 또는 노이즈의 영향을 줄일 수 있습니다. 또한 반대로 응답 시간을 짧게 하면 짧은 펄스 입력을 전송할 수 있습니다(단, 펄스 폭이 사이클 시간보다 길어야 함).

(주) 사이클 시간보다 짧은 펄스를 수신하고 싶은 경우는 펄스 캐치 기능이 있는 입력 유니트를 사용하십시오. 자세한 내용은 '10-2-4 펄스 캐치 기능'을 참조하십시오.



PLC 시스템 설정

CX-Programmer에서 '유니트 설정' 탭의 '랙 0~7 응답 시간'에 각각 설정하십시오.



관련 특수 보조 릴레이

명칭	주소	설명
기본 I/O 유니트의 입력 응답 시간	A220.00~A259.15	실제 기본 I/O 유니트의 입력 응답 시간 값을 읽어서 여기에 저장합니다.

● 기본 I/O 유니트 이상 관련

기본 I/O 유니트 관련 이상은 다음과 같습니다.

관련 특수 보조 릴레이

명칭	주소	설명
기본 I/O 유니트 이상 플래그 (운전 계속 이상)	A402.12	기본 I/O 유니트에 이상이 있을 때 1(ON)이 됩니다.
기본 I/O 유니트 이상 랙 번호	A408.08~ A408.15	기본 I/O 유니트에 이상이 있을 때(A40212가 ON) 이상이 발생한 유니트가 장착된 랙 번호가 BIN 데이터로 저장됩니다.
기본 I/O 유니트 이상 슬롯 번호	A408.00~ A408.07	기본 I/O 유니트에 이상이 있을 때(A40212가 ON) 이상이 발생한 슬롯 번호가 BIN 데이터로 저장됩니다.
기본 I/O 유니트 정보 영역 (0랙 0슬롯~3랙 9슬롯)	A050.00~ A069.15	기본 I/O 유니트(부하 단락 보호 기능 탑재)의 알람 정보를 표시합니다.
인터럽트 입력 유니트 장착 위치 부정 플래그	A405.08	인터럽트 입력 유니트가 CPU 장치의 슬롯 번호 0~3(CPU 유니트의 우측 4 유니트분) 이외에 장착됐을 때 1(ON)이 됩니다.

10-4-2 CPU 고기능 유니트 관련

● CPU 고기능 유니트 재시작/초기화 중

CPU 고기능 유니트는 다음의 재시작 플래그를 기동시켜서 PLC의 전원을 끄지 않고 CPU 유니트의 할당 DM 영역 등의 초기 설정 변경을 유효화합니다. 평상시에는 사용자 프로그램에서 조작합니다.

관련 특수 보조 릴레이

명칭	주소	설명
CPU 고기능 유니트 재시작 플래그	A501.00~ A501.15	기동(0→1) 시 CPU 고기능 유니트가 재시작되며 각 비트가 유니트 번호에 대응합니다. 비트 00~15: 0~F(유니트 번호)
CPU 고기능 유니트 초기화 중 플래그	A302.00~ A302.15	CPU 고기능 유니트 재시작 플래그(A501.00~A501.15)가 기동되거나 전원을 켰을 때, CPU 고기능 유니트가 초기 처리(초기화)에 들어가며 그 동안은 1(ON)이 됩니다. 각 비트가 유니트 번호에 대응합니다. 비트 00~15: 0~15(유니트 번호)

● CPU 고기능 I/O 유니트 관련 이상

CPU 고기능 유니트 관련 이상은 다음과 같습니다.

관련 특수 보조 릴레이

명칭	주소	설명
CPU 고기능 유니트 중복 유니트 번호 플래그	A410.00~ A410.15	CPU 고기능 유니트의 유니트 번호가 중복 설정됐을 때(A401.13이 ON 일 때) 중복된 대응 유니트 번호가 1(ON)로 설정되며 각 비트가 유니트 번호에 대응합니다. 비트 00~15: 0~F(유니트 번호)
CPU 고기능 유니트 설정 이상 플래그 (운전 계속 이상)	A402.03	I/O 테이블에 등록된 CPU 고기능 유니트의 기종과 장착된 CPU 고기능 유니트의 기종이 다를 때 1(ON)이 됩니다.
CPU 고기능 유니트 설정 이상 유니트 번호 플래그	A427.00~ A427.15	CPU 고기능 유니트 설정 이상 발생 시(A402.03이 ON일 때), 설정 이상이 발생한 유니트 번호에 해당하는 비트가 1(ON)이 되며 각 비트가 유니트 번호에 대응합니다. 비트 00~15: 0~F(유니트 번호)
CPU 고기능 유니트 이상 플래그 (운전 계속 이상)	A402.07	CPU 유니트←→CPU 고기능 유니트 간의 데이터 교환에 이상이 있는 경우(CPU 고기능 유니트 자체의 이상 포함) 1(ON)로 설정됩니다.
CPU 고기능 유니트 이상 유니트 번호 플래그	A417.00~ A417.15	CPU 유니트←→CPU 고기능 유니트 간의 데이터 교환에 이상이 있는 경우(A402.07이 ON일 때) 이상이 발생한 유니트 번호에 해당하는 비트가 1(ON)이 됩니다. 각 비트가 유니트 번호에 대응합니다. 비트 00~15: 0~F(유니트 번호)

10-4-3 고기능 I/O 유니트 관련

● 고기능 I/O 유니트 재시작/초기화 중

고기능 I/O 유니트는 다음의 재시작 플래그를 기동시켜서 PLC의 전원을 끄지 않고 CPU 유니트의 할당 DM 영역 등의 초기 설정 변경을 유효화합니다. 평상시에는 사용자 프로그램에서 조작합니다.

관련 특수 보조 릴레이

명칭	주소	설명
고기능 I/O 유니트 재시작 플래그	A502.00~ A507.15	기동(0→1) 시 고기능 I/O 유니트가 재시작되며(초기 처리에 들어감) 각 비트가 기기 번호에 대응합니다. A502.00~A507.15: 0~95(기기 번호)
고기능 I/O 유니트 초기화 중 플래그	A330.00~ A335.15	고기능 I/O 유니트 재시작 플래그(A502.00~507.15)가 기동되거나 전원을 켰을 때, 고기능 I/O 유니트가 초기 처리(초기화)에 들어가며 그 동안은 1(ON)이 됩니다. 각 비트가 기기 번호에 대응합니다. A330.00~A335.15: 0~95(기기 번호)

● 고기능 I/O 유니트 관련 이상

고기능 I/O 유니트 관련 이상은 다음과 같습니다.

관련 특수 보조 릴레이

명칭	주소	설명
고기능 I/O 유니트 중복 기기 번호 플래그	A411.00~ A416.15	고기능 I/O 유니트의 기기 번호가 중복 설정된 경우(A401.13이 ON일 때) 중복된 대응 기기 번호가 1(ON)로 설정되며 각 비트가 기기 번호에 대응합니다. A411.00~A416.15: 0~95(기기 번호)
고기능 I/O 유니트 설정 이상 플래그 (운전 계속 이상)	A402.02	I/O 테이블에 등록된 고기능 I/O 유니트의 기종과 장착된 고기능 I/O 유니트의 기종이 다를 때 1(ON)이 됩니다.
고기능 I/O 유니트 설정 이상 기기 번호 플래그	A428.00~ A433.15	고기능 I/O 유니트 설정 이상 발생 시(A402.02가 ON일 때), 설정 이상이 발생한 기기 번호에 해당하는 비트가 1(ON)이 되며 각 비트가 기기 번호에 대응합니다. A428의 비트 00~A433의 비트 15: 0~95(기기 번호)
고기능 I/O 유니트 이상 플래그 (운전 계속 이상)	A402.06	CPU 유니트←→고기능 I/O 유니트 간의 데이터 교환에 이상이 있는 경우(고기능 I/O 유니트 자체의 이상 포함) 1(ON)로 설정됩니다.
고기능 I/O 유니트 이상 기기 번호 플래그	A418.00~ A423.15	CPU 유니트←→고기능 I/O 유니트 간의 데이터 교환에 이상이 있는 경우(A402.06이 ON일 때) 1(ON)로 설정됩니다.

10-5 메모리 관리 기능

10-5-1 자동 백업 기능

사용자 프로그램 또는 파라미터를 CPU 유니트에 기록하면 자동으로 백업 메모리에 백업됩니다.

대상 데이터

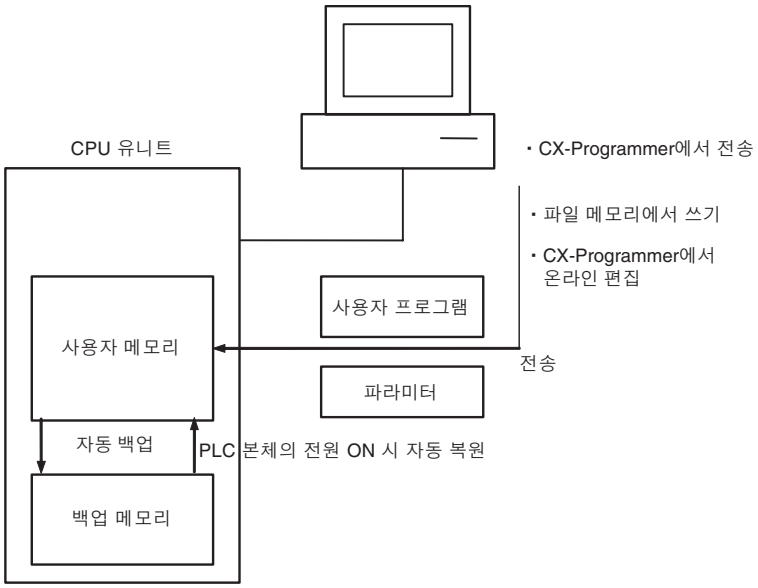
사용자 프로그램, 파라미터(PLC 이름, PLC 시스템 설정, I/O 테이블, 라우팅 테이블, 데이터 링크 테이블 등의 CPU 고기능 유니트 시스템 설정)

백업 처리가 실행될 때

CPU 유니트의 사용자 프로그램 또는 파라미터 영역에 기록했을 때(CX-Programmer에서의 전송 조작, 온라인 편집 조작, 메모리 카드 또는 EM 파일 메모리에서의 전송 조작 등)

복원 처리가 실행될 때

백업 메모리에 기록된 데이터(사용자 프로그램, 파라미터)는 PLC 본체의 전원을 켰을 때 자동으로 사용자 메모리에 읽기 처리가 실행됩니다.



- 백업 메모리에 백업하는 중에는 CPU 유니트 전면의 BKUP LED가 점등됩니다. CX-Programmer 또는 파일 메모리에서 사용자 프로그램이나 파라미터 영역으로 데이터를 전송하는 경우 또는 온라인 편집을 수행하는 경우, 백업 완료가 표시될 때까지(BKUP LED가 소등될 때까지) CPU 유니트의 전원을 끄지 마십시오. 전원을 끄면 다음과 같이 동작합니다.

● 자동 백업 시, 전원 OFF 발생 후의 동작

자동 백업 타이밍	대상 데이터	전원 OFF 발생 후의 동작
프로그램 전송 시	사용자 프로그램	• 데이터가 삭제됩니다. • 다음 번에 전원을 투입했을 때 메모리 이상이 발생합니다.
온라인 편집 시	사용자 프로그램 • 래더 태스크부	• 다음 번에 전원을 투입했을 때 온라인 편집 변경이 반영된 상태가 됩니다. (단, 배터리가 없는 경우에는 변경이 반영되지 않고 메모리 이상이 발생합니다.)
	사용자 프로그램 • ST 태스크부 • SFC 태스크부 • FB 정의 • SFC 작업/트랜지션부	• 온라인 편집 내용을 잃게 됩니다.
PLC 시스템 설정 전송 시 I/O 테이블 전송 시 I/O 테이블 작성/취소 시 라우팅 테이블 전송 시 고기능 유니트 설정 전송 시	파라미터	• 데이터가 삭제됩니다. • 다음 번에 전원을 투입했을 때 메모리 이상이 발생합니다.

- CPU 유니트에 배터리가 있는 경우 래더 태스크부 온라인 편집에 한하여 백업 완료 전에 전원을 꺼도 다음 번에 전원을 켜면 전원 OFF 직전의 상태로 복구되지만, 다음과 같은 점에 주의하십시오.
  - CPU 유니트의 기동 시간이 길어집니다.
  - 배터리가 있어도 장시간 전원을 투입하지 않은 경우에는 백업 완료를 확인한 후 전원을 끄십시오.

• 사용자 프로그램 용량에 맞는 백업 처리 시간(BKUP LED 점등 시간)의 기준은 다음과 같습니다.

사용자 프로그램 용량	백업 처리 시간	
	'モニタ' 모드 시	'プログラム' 모드 시
100K 스텝	20초 (사이클 시간 3ms)	3초
250K 스텝	47초 (사이클 시간 7ms)	6초
400K 스텝	75초 (사이클 시간 11ms)	10초

(주) 전원을 켜면 CPU 유니트의 BKUP LED가 점등됩니다.

사용상의 주의 사항

- 자동 백업 기능에서는 I/O 메모리 영역(DM/EM/HR/AR 포함)이 백업 메모리에 기록되지 않습니다. 특히, 전원 OFF 복구 시 유지 영역인 DM/EM/HR 영역은 배터리에 의해 유지됩니다. 따라서 배터리에 이상이 발생한 경우 정확한 값을 유지하지 못할 가능성이 있습니다.
  그러므로 프로그램에서 DM/EM/HR/AR 영역의 내용을 이용하여 외부로 출력하는 경우, 배터리 이상 플래그(A402.04)가 1(ON)이면 외부로 출력하지 않는 등의 대책을 마련하십시오.
- CPU 유니트에서 백업 메모리 이상이 검출되면 백업 메모리 이상(운전 계속 이상)이 발생합니다. 이와 같은 이상이 발생한 경우, 다음 번 전원 투입 시에 데이터가 복원되지 않고 메모리 이상이 될 가능성이 있습니다.
  따라서 전원 OFF 전에 데이터를 보관하시기 바랍니다.

참고

CX-Programmer에서는 평상시의 전송 조작([PLC]－[전송] 선택) 이외에 의한 전송 조작의 경우, [메모리 백업 상태] 창에 CPU 유니트 내 백업 메모리로 프로그램 또는 파라미터 영역 데이터를 백업한 상태(쓰기 상황)가 표시됩니다. (PLC의 속성에서 '백업 중인 상태의 대화 상자 표시'를 선택하고, [보기] 메뉴에서 [창]－[PLC 백업 모니터 상태]를 선택합니다.)
평상시의 전송 조작([PLC]－[전송] 선택)의 경우, [전송[PC→PLC]] 대화 상자 내에서 프로그램 등의 전송에 이어 백업 메모리의 백업 상태(쓰기 상황)가 표시됩니다.

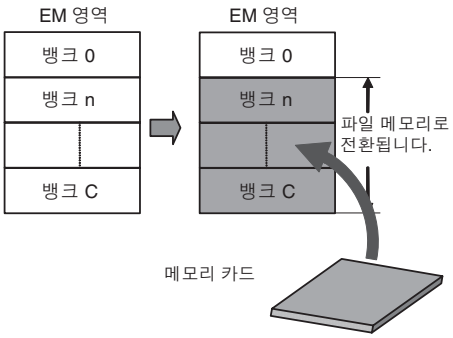
관련 특수 보조 릴레이

명칭	주소	내용
백업 메모리 이상 플래그	A315.15	백업 메모리가 고장났을 때 1(ON)이 됩니다.

10-5-2 EM 파일 메모리 기능

메모리 카드 대신 EM 영역을 CPU 유니트 내의 파일 저장 영역으로 사용할 수 있습니다. 지정된 뱅크 번호 이후를 파일 메모리로 사용합니다.

EM 영역을 파일 메모리로 사용하는 방법에 대해서는 '7-1-2 매체의 초기화'를 참조하십시오.



1. CX-Programmer의 EM 메모리 설정에서 시작 뱅크 번호를 지정합니다.
2. CX-Programmer에서 프로그램을 전송하면 EM 파일 메모리를 사용할 수 있습니다.

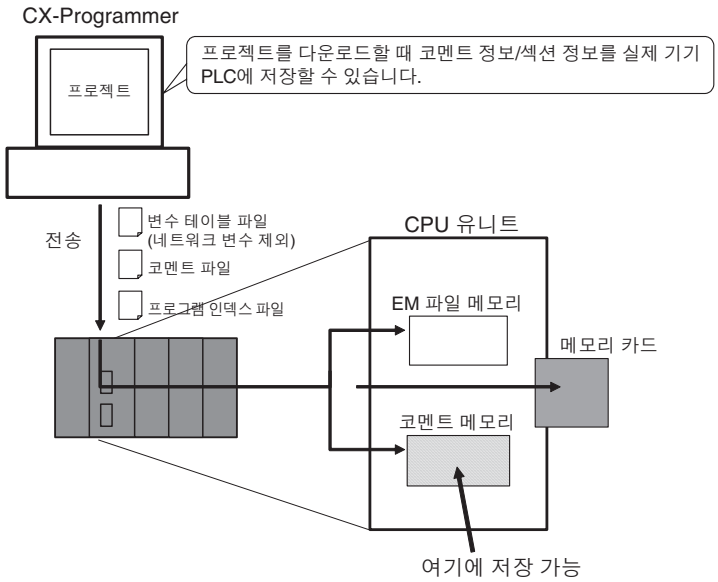
EM 영역은 파일 메모리나 트레이스 메모리 중 하나를 선택하여 할당합니다.

10-5-3 코멘트 기억 기능

CPU 유니트는 백업 메모리 내에 코멘트 메모리가 있습니다.

따라서 메모리 카드나 EM 파일 메모리가 모두 없는 경우라도 이 코멘트 메모리에 다음과 같은 코멘트 정보/섹션 정보를 저장해서 읽어올 수 있습니다.

- 변수 테이블 파일(네트워크 변수 제외, CX-Programmer의 변수명, I/O 코멘트 포함)
- 코멘트 파일(CX-Programmer의 행 코멘트, 주석문)
- 프로그램 인덱스 파일(CX-Programmer의 섹션명, 섹션 코멘트, 프로그램 코멘트)

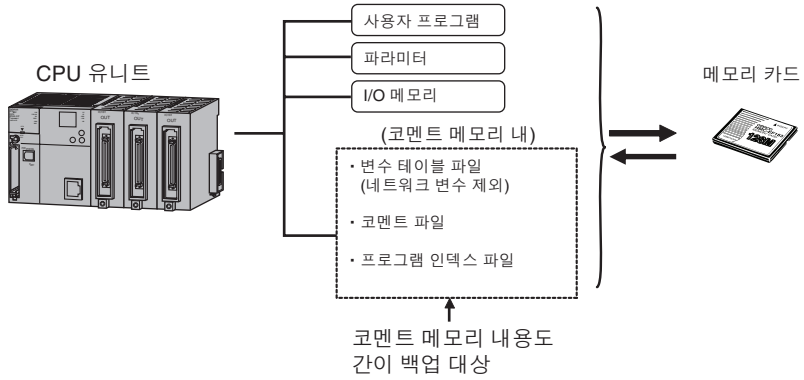


프로젝트를 다운로드할 때는 코멘트 정보/섹션 정보의 전송 대상을 다음에서 선택할 수 있습니다.

- 코멘트 메모리(CPU 유니트 백업 메모리 내)
- 메모리 카드
- EM 파일 메모리

간이 백업 조작을 실행하면 코멘트 메모리에 있는 파일이 메모리 카드에 백업되거나 반대로 메모리 카드에서 코멘트 메모리에 복원됩니다.

- 변수 테이블 파일(네트워크 변수 제외) (CX-Programmer의 변수명, I/O 코멘트 포함)
- 코멘트 파일(CX-Programmer의 행 코멘트, 주석문)
- 프로그램 인덱스 파일(CX-Programmer의 섹션명, 섹션 코멘트, 프로그램 코멘트)

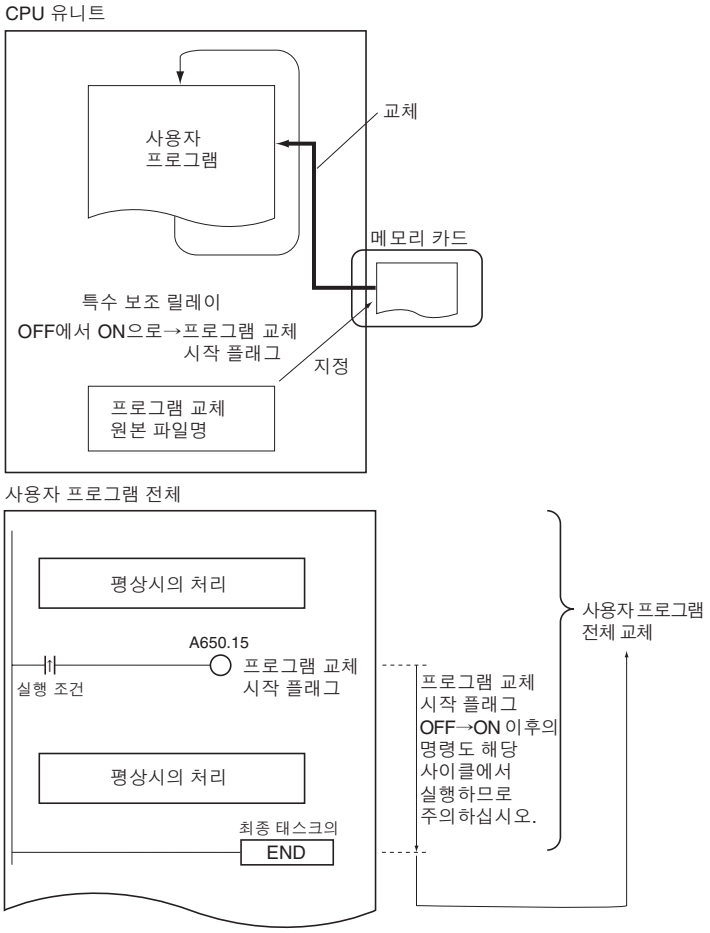


따라서 문제 발생 시 또는 동일 사양의 CPU 유니트 증설 시 등에 I/O 코멘트 등을 포함한 CPU 유니트 내의 모든 데이터를 CX-Programmer 없이 간단하게 백업/복원할 수 있습니다.

10-5-4 프로그램 교체 기능

개요

사용자 프로그램 실행 중('운전' 모드 또는 '모니터' 모드)에 특수 보조 릴레이 '프로그램 교체 시작 플래그'(A650.15)를 0(OFF)→1(ON)로 설정하면 미리 특수 보조 릴레이에 지정해 둔 파일명의 프로그램/네트워크 변수 파일(.OBJ)을 메모리 카드에서 읽어오고, 해당 사이클의 마지막에 그 사용자 프로그램에 실행 프로그램을 교체합니다.



또한 사용자 프로그램 정지 중('프로그램' 모드 시)에도 교체가 가능합니다. (이 경우, '프로그램 교체 시작 플래그'를 CX-Programmer에서 조작합니다.)

- EM 파일 메모리에서 읽어올 수는 없습니다.
- 사용자 프로그램에서 '프로그램 교체 시작 플래그'(A650.15)를 0(OFF)→1(ON)로 설정하는 위치는 임의의 위치(프로그램 주소)에서 가능합니다.
- '프로그램 교체 시작 플래그'를 0(OFF)→1(ON)로 한 후 해당 사이클에서 그 이후의 명령이 실행됩니다.
- 프로그램 교체 중에는 프로그램이 실행되지 않습니다. 프로그램 교체 종료 후, 평상시의 '프로그램' 모드에서 '운전' 또는 '모니터' 모드로의 이동과 동일한 동작에서 운전이 시작됩니다.
- 프로그램의 교체 타이밍은 '프로그램 교체 시작 플래그'를 0(OFF)→1(ON)로 한 사이클의 마지막(최종 태스크의 END 명령 실행 후)입니다.

프로그램 교체 중의 동작

프로그램의 교체 중에 CPU 유니트의 동작은 다음과 같습니다.

- 프로그램의 실행: 정지합니다.
- 사이클 시간의 감시: 감시되지 않습니다.

● 프로그램의 교체 중 및 교체 후에도 계속되는 동작

- IOM(I/O 메모리) 유지 비트(A500.12)가 1(ON)인 경우 다음 I/O 메모리 영역이 유지됩니다. 단, IOM(I/O 메모리) 유지 비트를 1(ON)로 한 경우에는 교체 전에 출력한 부하를 교체 후의 프로그램에서 OFF로 하는 처리가 없으면 부하는 출력된 상태를 유지하므로 주의하십시오.
  유지되는 영역: CIO, 내부 보조 릴레이(WR), 타이머 업 플래그, 인덱스 레지스터(IR), 데이터 레지스터(DR), 현재 EM 뱅크 번호
  (주) 타이머 현재값은 클리어됩니다.
- 강제 세트/리셋 유지 플래그(A500.13)가 1(ON)인 경우 강제 세트/리셋 상태가 유지됩니다.
- 인터럽트 허가 중인 경우: 초기화되고 인터럽트는 금지됩니다.
- 데이터 트레이스 실행: 트레이스를 중단합니다.
- 미분 조건 플래그: 초기화됩니다(I/O 메모리의 유지, 비유지 어느 경우이든).

● 프로그램 교체 후의 동작

- 사이클릭 태스크 상태: 프로그램의 속성 내 '운전 시 태스크 기동'에 좌우됩니다('프로그램' 모드에서 '운전' 또는 '모니터' 모드로 이동할 때와 동일).
- 운전을 시작할 때 1사이클 ON 플래그(A200.11)의 상태: 교체 후의 프로그램 실행 1사이클째만 ON이 됩니다('프로그램' 모드에서 '운전' 또는 '모니터' 모드로 이동할 때와 동일).

순서

1 **프로그램 파일이 미리 저장된 메모리 카드를 CPU 유니트에 장착합니다.**

2 **특수 보조 릴레이 '프로그램 교체 원본 파일명'(A654~657CH), '프로그램 교체 비밀번호'(A651CH)를 설정하고 '교체 시작 플래그'(A650.15)를 프로그램에서 OFF→ON합니다.**



참고

프로그램 교체 전후에 I/O 메모리를 유지하려면 IOM(I/O 메모리) 유지 비트(A500.12)를 1(ON)로 설정합니다.
프로그램 교체 전후에 강제 세트/리셋 상태를 유지하려면 강제 세트/리셋 유지 플래그(A500.13)를 1(ON)로 설정합니다.



안전상의 중요 사항

프로그램 교체 전에 IOM(I/O 메모리) 유지 비트(A500.12)가 1(ON)이 된 경우, 프로그램 교체 후에도 I/O 메모리가 유지되므로 외부 부하의 상태에 주의하십시오.
마찬가지로 프로그램 교체 전에 강제 세트/리셋 유지 플래그(A500.13)가 1(ON)이 된 경우, 프로그램의 교체 후에도 강제 세트/리셋 상태가 유지되므로 외부 부하의 상태에 주의하십시오.

교체 대상 파일

다음과 같은 프로그램/네트워크 변수 파일을 메모리 카드에서 읽어옵니다.

파일	파일명/확장자	특수 보조 릴레이 '프로그램 교체 시작 플래그' (A65015) OFF→ON 시	메모리 카드 내의 프로그램/ 네트워크 변수 파일명 ＊＊＊＊＊＊＊＊에 대한 지정
프로그램/네트워크 변수 파일	＊＊＊＊＊＊＊＊.OBJ	CPU 유니트의 사용자 프로그램을 교체합니다.	특수 보조 릴레이 '프로그램 교체 원본 파일명'(A654~A657CH) 사용

프로그램 교체가 가능한 조건

프로그램 교체가 가능한 조건은 다음과 같습니다.

- 메모리 카드 내에 특수 보조 릴레이 '프로그램 교체 원본 파일명'(A654~A657CH)에서 지정한 프로그램/네트워크 변수 파일이 루트 디렉토리에 존재할 것
- 메모리 카드가 인식될 것(A343.15가 ON)
- 운전 정지 이상이 발생하지 않을 것
- 파일 메모리 관련 조작이 실행되지 않을 것(A343.13이 OFF)
- 프로그램 영역에 쓰기가 실행되지 않을 것
- 액세스 권한이 해제되어 있을 것(예: CX-Programmer에서 PLC로 전송이 실행되지 않음)

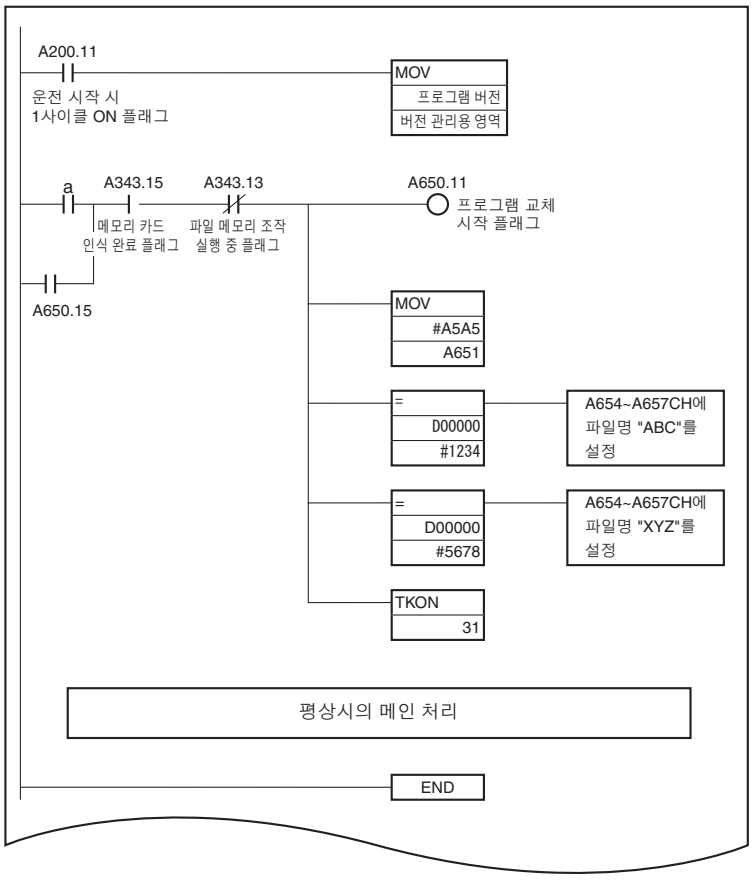
(주) CPU 유니트의 동작 모드는 어떤 모드라도 가능합니다.

프로그램 예

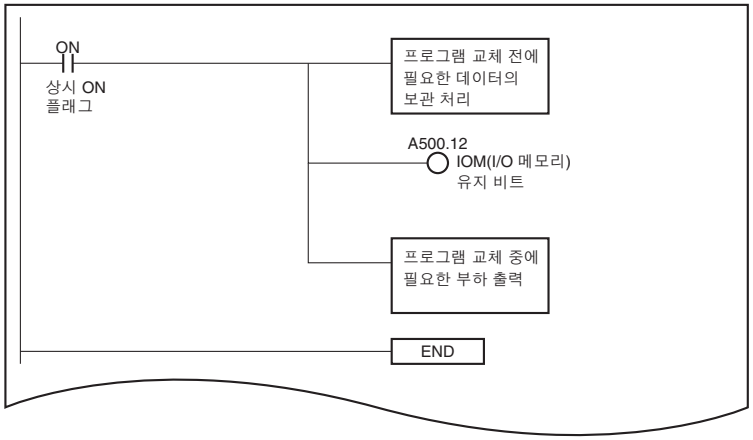
● 프로그램 예 1

메모리 카드 내에 프로그램/네트워크 변수 파일 ABC.OBJ와 XYZ.OBJ를 저장해 두고 실행 조건이 OFF→ON으로 변경될 때 D0 내의 값에 맞춰 프로그램을 선택해서 교체합니다.
프로그램/네트워크 변수 파일 ABC.OBJ를 선택한 경우 D0 내에는 #1234를 저장해 둡니다. 프로그램/네트워크 변수 파일 XYZ.OBJ를 선택한 경우에는 #5678을 저장해 둡니다.
또한 프로그램 교체 전에 필요한 처리 및 필요한 출력의 유지 처리를 교체 중에 다른 태스크를 기동하여 실행합니다.

메인 태스크(사이클릭 태스크 번호 0)

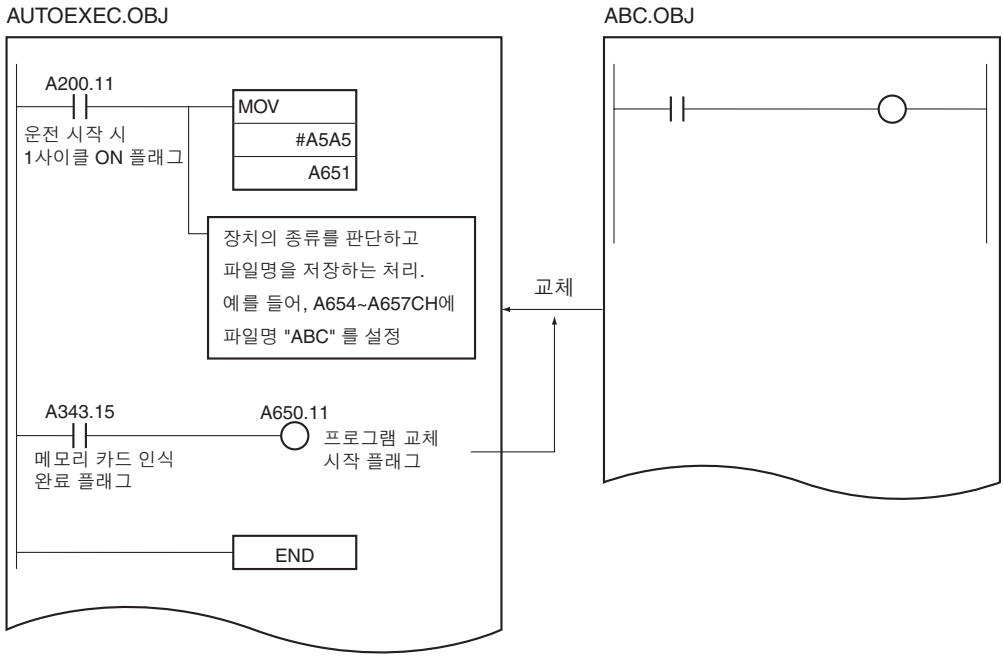


프로그램 교체용 데이터 보관 태스크(사이클릭 태스크 번호 31)<운전 시작 시 기동 안 함>

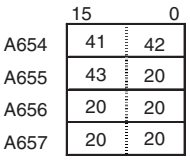


● 프로그램 예 2

1장의 메모리 카드에 여러 장치의 프로그램/네트워크 변수 파일(.OBJ) 및 전원 ON 시 자동 전송 프로그램/네트워크 변수 파일(AUTOEXEC.OBJ 또는 REPLACE.OBJ)을 저장해 둡니다. 전원 OFF→ON 시에 먼저 전원 ON 시 자동 전송 프로그램을 읽고 그 프로그램에서 장치 종류에 맞춰 프로그램/네트워크 변수 파일(.OBJ)을 선택하여 교체합니다.



관련 특수 보조 릴레이

명칭	주소	내용
파일 메모리 조작 실행 중 플래그	A343.13	다음 중 하나의 파일 메모리 관련 조작을 실행 중일 때 1(ON)이 됩니다. 아무 것도 실행하지 않을 때는 0(OFF)이 됩니다. • 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 인식할 수 있으면 1(ON)이 됩니다. 메모리 카드를 인식할 수 없으면 0(OFF)이 됩니다.
IOM(I/O 메모리) 유지 비트	A500.12	CPU 유니트의 동작 모드 전환(프로그램←→운전/모니터) 또는 전원 ON 시, I/O 메모리 영역의 유지 여부를 지정합니다. 1: 운전 모드 변경(프로그램←→모니터 또는 운전 모드) 시, I/O 메모리 유지 0: 운전 모드 변경 시, I/O 메모리 클리어
강제 세트/리셋 유지 플래그	A500.13	CPU 유니트의 동작 모드 전환(프로그램←→모니터) 시 또는 전원 ON 시, 강제 세트/리셋 상태의 유지 여부를 지정합니다.
프로그램 교체 종료 코드	A650.00~ A650.07	• 정상 종료 시(A650.14가 0(OFF)일 때) 01Hex: 프로그램/네트워크 변수 파일(.OBJ)을 교체했습니다. • 이상 종료 시(A650.14가 1(ON)일 때) 00Hex: 정지 이상이 발생했습니다. 01Hex: 메모리 이상이 발생했습니다. 11Hex: 쓰기가 금지되어 있습니다. 12Hex: 프로그램 교체 비밀번호 이상입니다. 21Hex: 메모리 카드가 존재하지 않습니다. 22Hex: 지정된 파일이 존재하지 않습니다. 23Hex: 지정된 파일의 크기가 기종의 용량을 초과했습니다(메모리 이상). 31Hex: 다음 중 하나의 다른 처리를 실행 중입니다. ・ 파일 메모리 관련 처리 실행 중 ・ 사용자 프로그램 쓰기 중 ・ 동작 모드 변경 중
프로그램 교체 이상 종료 플래그	A650.14	프로그램 교체 시작 플래그(A650.15)를 0(OFF)→1(ON)로 해서 프로그램을 교체하려고 했지만 이상 종료가 발생한 경우 1(ON)이 됩니다. 다시 프로그램 교체 시작 플래그가 0(OFF)→1(ON)이 되면 0(OFF)이 됩니다.
프로그램 교체 시작 플래그	A650.15	프로그램 교체 비밀번호(A651CH)가 유효한 상태(A5A5Hex)에서 이 플래그를 0(OFF)→1(ON)로 하면 프로그램 교체를 시작합니다. 프로그램 교체 중에는 0(OFF)→1(ON)로 하지 마십시오. 전원 ON 시 또는 프로그램 교체 종료 시에는 정상 종료/이상 종료에 관계 없이 0(OFF)이 됩니다. CX-Programmer, PT, 상위 PC 등에서 이 플래그를 읽으면 프로그램을 교체 중인지 여부를 알 수 있습니다.
프로그램 교체 비밀번호	A651CH	프로그램을 교체하는 데 필요한 비밀번호를 입력합니다. A5A5Hex: 프로그램 교체 시작 플래그(A650.15) 유효 위 이외: 프로그램 교체 시작 플래그(A650.15) 무효 전원 ON 시 또는 프로그램 교체 종료 시에는 정상 종료/이상 종료에 관계 없이 0(OFF)이 됩니다.
프로그램 교체 원본 파일명	A654~ A657CH	프로그램 교체를 시작하면 교체 원본 파일명을 ASCII 코드로 저장해 둡니다. 파일명은 확장자를 제외하고 영숫자로 8문자 이내에서 지정합니다. A654CH에서 A657CH 순서(하위 CH→상위 CH)로, 상위 바이트→하위 바이트 순으로 저장합니다. 8문자에 도달하지 않는 경우는 나머지 하위 바이트 및 상위 CH를 20Hex(공백)로 채웁니다. 파일명 중간에 NUL이나 공백을 삽입할 수 없습니다. 예) 파일명이 ABC.OBJ인 경우 

10-6 보호 기능

CJ 시리즈는 사용자 프로그램에 대해 다음의 보호 작업을 수행할 수 있습니다.

10-6-1 딥 스위치에 의한 쓰기 방지 기능

CX-Programmer에 의한 사용자 프로그램 및 파라미터 영역의 데이터(PLC 시스템 설정, I/O 테이블 등)의 재기입을 금지할 수 있습니다. 이렇게 하면 예상치 못한 프로그램 변경을 방지할 수 있습니다.
CPU 유니트의 딥 스위치 SW1을 ON/OFF시키면 유효/무효를 전환할 수 있습니다.
또한 쓰기 방지를 사용하는 경우에도 CX-Programmer에서 프로그램을 읽고 표시할 수는 있습니다.

● CPU 유니트 전면 딥 스위치

스위치 번호	명칭	설정
SW1	사용자 프로그램 메모리 쓰기 불가	1: 불가 0: 가능

10-6-2 비밀번호를 사용한 읽기 방지 기능

비밀번호를 사용하여 읽기 방지를 수행할 수 있습니다.
보호된 프로그램의 경우 CX-Programmer 조작자가 비밀번호를 입력하지 않으면 사용자 프로그램이나 특정 태스크, 펑션 블록을 표시/편집할 수 없습니다.

보호의 종류

비밀번호를 사용한 읽기 방지에는 다음과 같은 종류가 있습니다.

● UM 읽기 방지

CPU 유니트에서 사용자 프로그램을 읽을 수 없습니다.

● 태스크 읽기 방지

하나 이상의 태스크를 CX-Programmer에서 표시할 수 없게 됩니다.
CPU 유니트에서 사용자 프로그램을 읽을 수 있지만 보호 대상 태스크는 CX-Programmer에 표시되지 않습니다.

참고

- 읽기 방지된 태스크(프로그램)를 포함한 사용자 프로그램 전체를 다른 CPU 유니트로 전송할 수 있습니다. 단, 태스크(프로그램)에 설정된 비밀번호를 사용한 읽기 방지는 유효한 상태입니다.
- CX-Programmer에서 PC 메모리 내의 사용자 프로그램과 CPU 유니트 내의 사용자 프로그램을 비교하는 경우 비밀번호로 읽기 방지된 태스크(프로그램)도 함께 비교합니다.

● 펑션 블록 보호

임의의 펑션 블록에 대해 다음과 같이 보호할 수 있습니다.

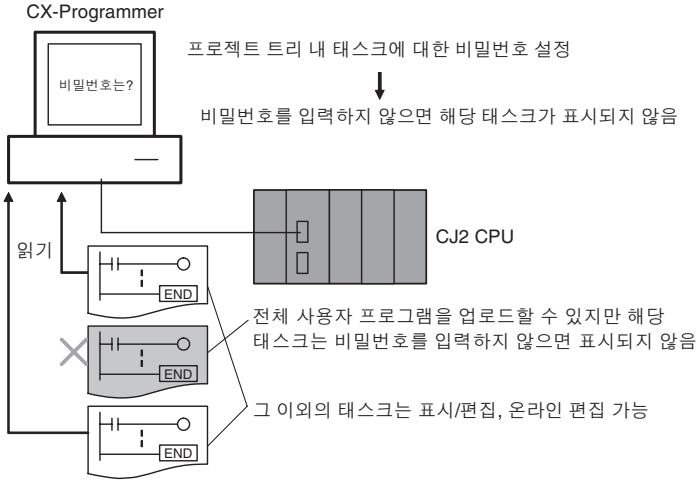
- 쓰기 및 표시 금지:
  CX-Programmer에서 표시할 수 없습니다. 따라서 변경할 수도 없습니다.
- 쓰기 금지:
  CX-Programmer에서 표시할 수 있지만 변경할 수는 없습니다.

참고

펑션 블록을 포함한 사용자 프로그램의 경우, 태스크 읽기 방지를 설정해도 펑션 블록 정의 자체는 읽을 수 있습니다.
펑션 블록 자체를 읽지 못하게 하려면 펑션 블록 보호 기능을 사용하십시오.

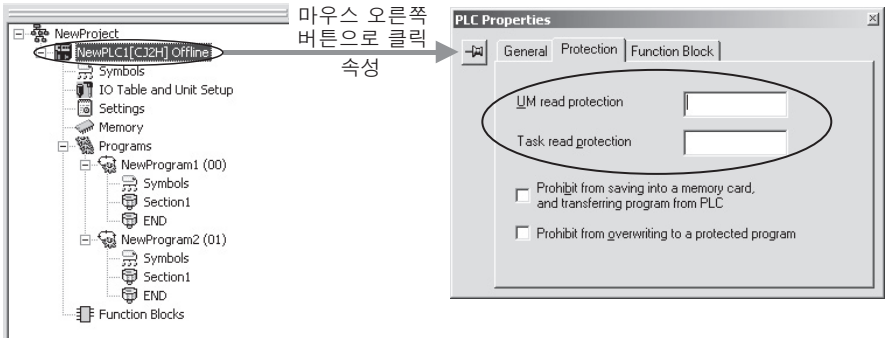
참고

UM 읽기 방지 중에는 태스크 읽기 방지를 수행할 수 없습니다.
단, 태스크 읽기 방지 중에 UM 읽기 방지를 수행할 수는 있습니다.



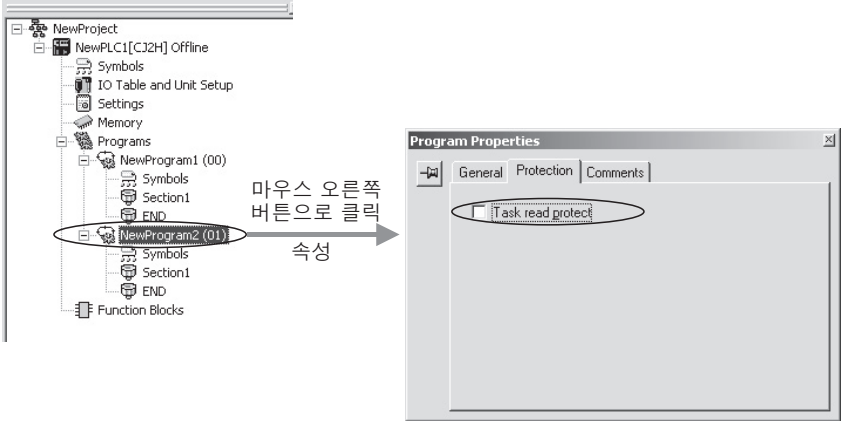
조작 방법

1 CX-Programmer의 PLC 속성의 '보호' 탭에서 다음의 보호용 비밀번호를 등록합니다.

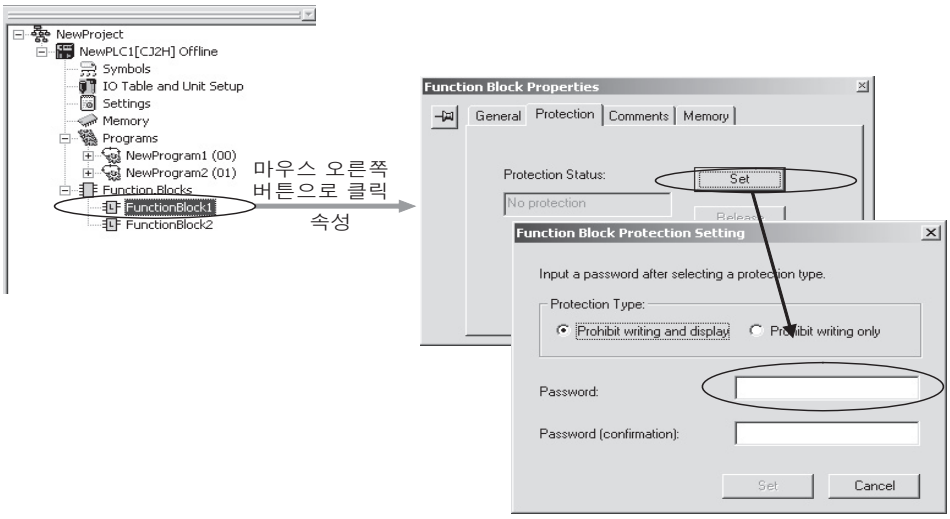


- 'UM 읽기 보호'
  UM 읽기 방지용 비밀번호를 입력합니다.
- '태스크 읽기 보호'
  태스크 읽기 방지용 비밀번호를 입력합니다.

2 태스크에 읽기 방지를 설정하려면 보호 대상 태스크(프로그램)를 선택하고 프로그램의 속성에서 '태스크 읽기 보호'를 선택합니다.



3 **펑션 블록(FB) 정의에 대해 읽기 방지를 실행하려면 보호 대상 펑션 블록을 선택하고 펑션 블록의 속성에서 [설정] 버튼을 클릭한 후 '펑션 블록 보호 설정'에서 비밀번호를 입력합니다.**



4 **온라인으로 연결한 후 다음의 a) 또는 b) 중 하나를 실행합니다.**

a) 프로그램 전송과 함께 보호를 실행하는 경우:

[PLC]−[전송]−[PLC로 전송]을 통해 프로그램을 전송합니다.
위의 순서 2에서 대상으로 지정된 태스크가 보호됩니다.

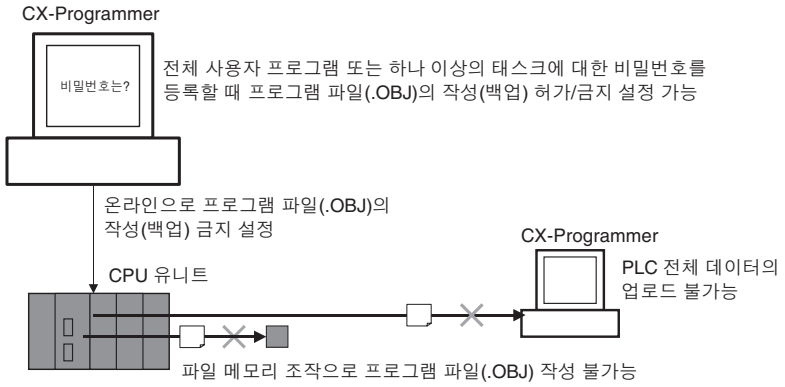
b) 프로그램을 전송하지 않고 보호를 실행하는 경우:

[PLC]−[보호]−[암호 설정]을 선택하고 [확인] 버튼을 클릭합니다.
위의 순서 2에서 대상으로 지정된 태스크가 보호됩니다.

메모리 카드에 저장 금지/허가 설정

● 개요

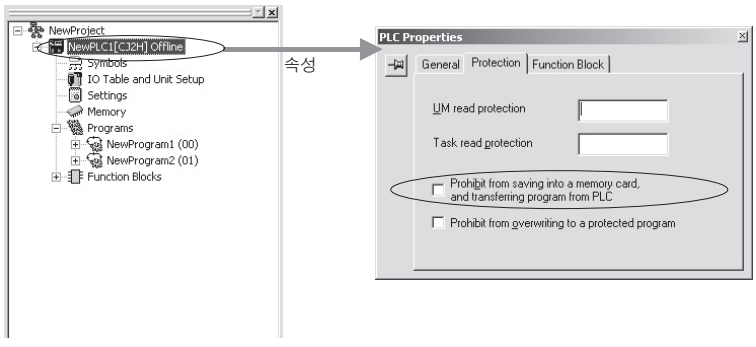
CX-Programmer에서 사용자 프로그램 전체 또는 하나 이상의 태스크에 대한 비밀번호 등록 시, 옵션으로 프로그램/네트워크 변수 파일(.OBJ)의 작성(백업) 금지/허가를 동시에 설정할 수 있습니다.
프로그램/네트워크 변수 파일(.OBJ) 작성(백업)을 금지하도록 설정한 경우, 파일 메모리 조작을 통해 프로그램/네트워크 변수 파일(.OBJ)을 작성할 수 없게 됩니다(온라인에서 메모리 카드/EM 파일 메모리로 전송, CX-Programmer로 PLC 전체 데이터의 업로드, 오프라인에서 기억 장치에 저장 등이 모두 불가능).
이렇게 하면 사용자 프로그램의 불법 복사를 방지할 수 있습니다.



- 프로그램/네트워크 변수 파일(.OBJ)의 작성(복제)을 금지하도록 설정한 경우도 간이 백업 조작은 가능합니다. 단, 간이 백업용 프로그램/네트워크 변수 파일(BACKUP.OBJ)은 작성되지 않습니다.
- 읽기 방지에 의한 설정이 아닌 경우 복제 가능합니다.
- 프로그램/네트워크 변수 파일의 작성 금지/허가 설정은 프로그램을 전송하지 않으면 반영되지 않습니다. 설정을 변경한 후에는 반드시 프로그램을 전송하십시오.

● 조작 방법

1 **CX-Programmer의 PLC 속성의 '보호' 탭에서 'UM 읽기 보호' 또는 '태스크 읽기 보호'에 비밀번호를 등록하는 경우 '메모리 카드에 저장 및 PLC에서 프로그램 전송 금지'를 선택합니다.**



2 **온라인으로 연결한 후 [PLC] – [전송] – [PLC로 전송]을 통해 프로그램을 전송하거나 [PLC] – [보호] – [암호 설정]을 선택하고 [확인] 버튼을 클릭합니다.**

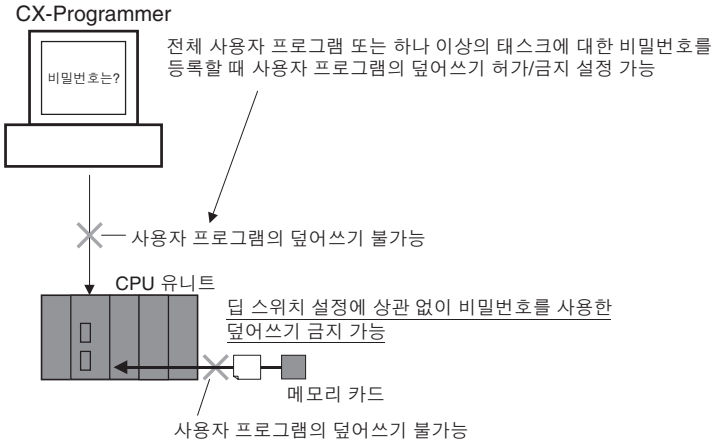
● 용도

이렇게 하면 비밀번호를 사용하여 프로그램의 외부 유출을 방지할 수 있습니다.

프로그램의 덮어쓰기 금지/허가 설정

● 개요

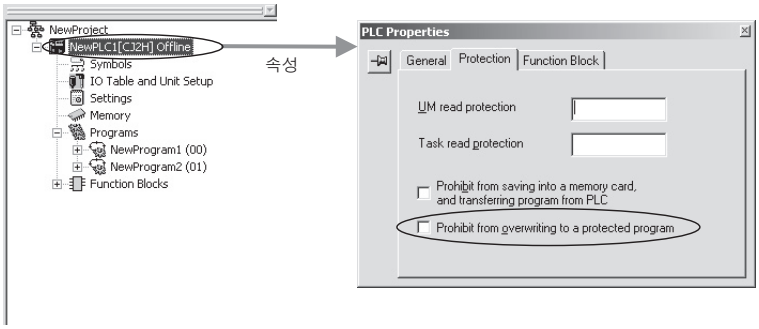
위의 딥 스위치를 사용한 방법 이외에 CX-Programmer에서 사용자 프로그램 전체 또는 하나 이상의 태스크에 대해 비밀번호를 등록할 때 옵션으로 프로그램의 덮어쓰기 허가/금지를 동시에 설정할 수도 있습니다.
이렇게 하면 제삼자에 의한 프로그램의 덮어쓰기 또는 예상치 못한 덮어쓰기를 막을 수 있습니다.



- 태스크 읽기 방지에서 옵션인 '프로그램의 덮어쓰기 금지'를 설정한 경우, 금지 대상 태스크(프로그램)만 덮어 쓸 수 없습니다. 그 이외의 태스크(프로그램)는 온라인 편집, 태스크 다운로드 등을 통해 덮어 쓸 수 있습니다.
- 읽기 방지를 설정하지 않은 경우는 모든 태스크(프로그램)에 덮어쓰기가 가능합니다.
- 프로그램의 덮어쓰기 금지/허가 설정은 프로그램을 전송하지 않으면 반영되지 않습니다. 설정을 변경한 후에는 반드시 프로그램을 전송하십시오.

● 조작 방법

1 **CX-Programmer의 PLC 속성의 '보호' 탭에서 'UM 읽기 보호' 또는 '태스크 읽기 보호'용 비밀번호를 등록하고 '보호된 프로그램에 덮어쓰지 못하도록 금지'를 선택합니다.**



2 **온라인으로 연결한 후 [PLC] – [전송] – [PLC로 전송]을 통해 프로그램을 전송하거나 [PLC] – [보호] – [암호 설정]을 선택하고 [확인] 버튼을 클릭합니다.**

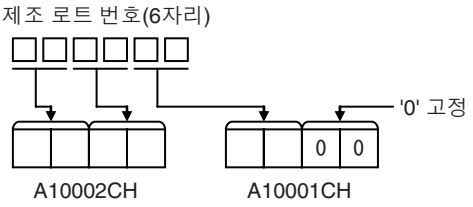
비밀번호를 사용한 읽기 방지 관련 특수 보조 릴레이

명칭	주소	내용
UM 읽기 방지 상태	A99.00	PLC(사용자 프로그램 전체) 단위로 읽기 방지가 설정되어 있는지 여부를 표시합니다. 0: UM 읽기 방지 설정 안 함 1: UM 읽기 방지 설정
태스크 읽기 방지 상태	A99.01	태스크 단위로 읽기 방지가 설정되어 있는지 여부를 표시합니다. 0: 태스크 읽기 방지 설정 안 함 1: 태스크 읽기 방지 설정
읽기 방지 설정 시, 프로그램 덮어쓰기 금지/허가 설정 상태	A99.02	프로그램의 덮어쓰기 허가/금지 상태를 표시합니다. 0: 허가 중 1: 금지 중
읽기 방지 설정 시, 메모리 카드에 프로그램 백업 허가/금지 상태	A99.03	프로그램/네트워크 변수 파일(.OBJ)의 작성(백업) 허가/금지 상태가 표시됩니다. 0: 허가 중 1: 금지 중

10-6-3 제조 로트 번호에 따른 운전 금지 기능

특수 보조 릴레이 영역(A10001CH, A10002CH)에 저장된 제조 로트 번호를 사용하여 프로그램의 운전을 방지할 수 있습니다.
이 제조 로트 번호는 사용자가 변경할 수 없습니다.

- 제조 로트 번호는 6자리이므로 상위 4자리가 A10002CH에 하위 2자리가 A10001CH에 저장됩니다.

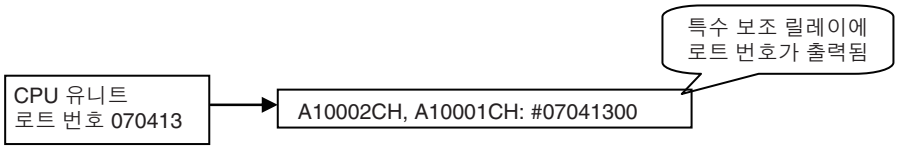


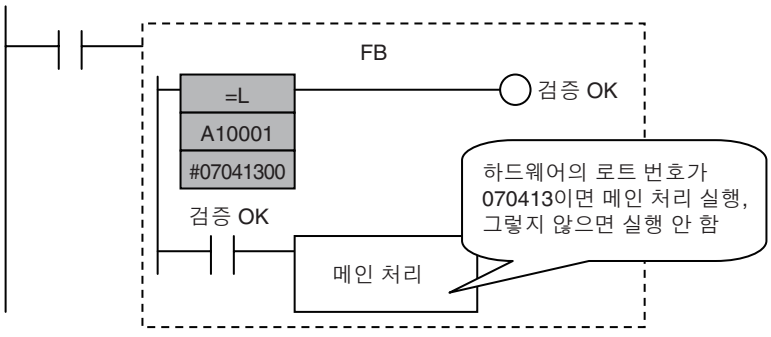
제조 로트 번호의 저장 예

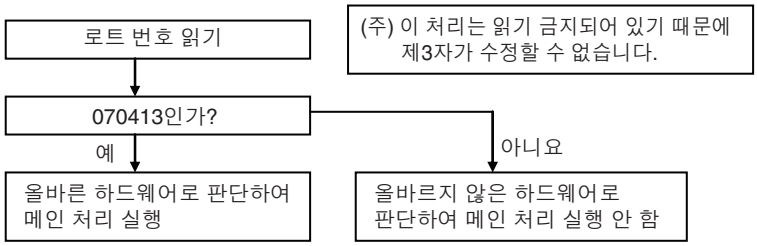
제조일	제조 로트 번호	A10002CH	A10001CH
2008년 6월 20일	080620	0806	2000

● 응용 예

특정 제조 로트 번호의 CPU 유니트에서만 프로그램을 동작시키는 경우

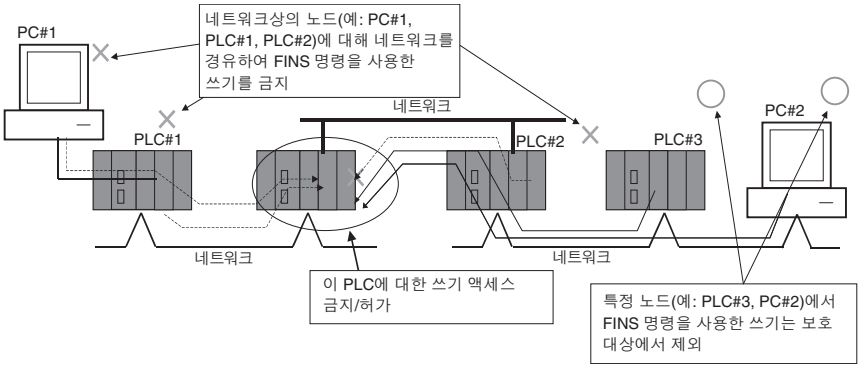






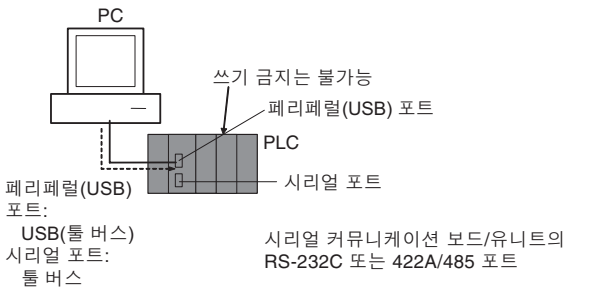
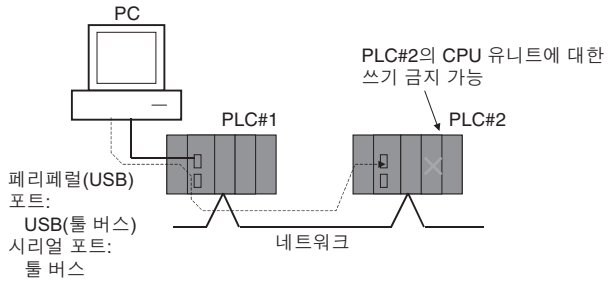
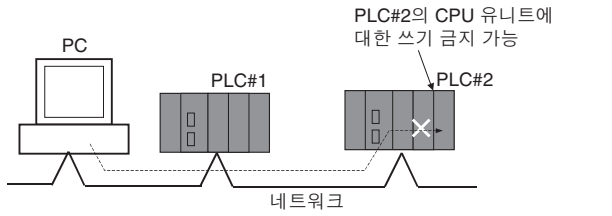
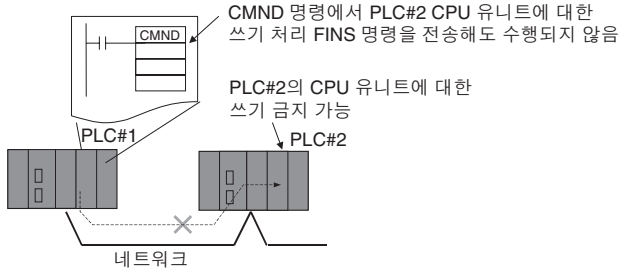
10-6-4 FINS 쓰기 방지 기능

네트워크를 경유(직접 시리얼 연결 이외)하여 쓰기 관련 FINS 명령을 사용해서 CPU 유니트에 대한 쓰기 처리/조작 처리를 하려고 하는 경우(CX-Programmer/CX-Protocol/CX-Process, 기타 FinsGateway를 사용한 응용 프로그램에서 쓰기 포함), 이를 금지할 수 있습니다(읽기 처리는 가능).
사용자 프로그램, PLC 시스템 설정, I/O 메모리 등의 모든 다운로드 및 동작 모드 변경, 온라인 편집 등과 같은 쓰기 처리를 불가능하도록 설정할 수 있습니다.
단, 특정 노드에서의 쓰기는 보호 대상에서 제외(보호 해제)할 수도 있습니다.
또한 네트워크를 경유한 쓰기 처리는 CPU 유니트 내에 쓰기 처리 내역이 자동으로 기록되며, 해당 읽기용 FINS 명령을 통해 읽을 수 있습니다.
예)



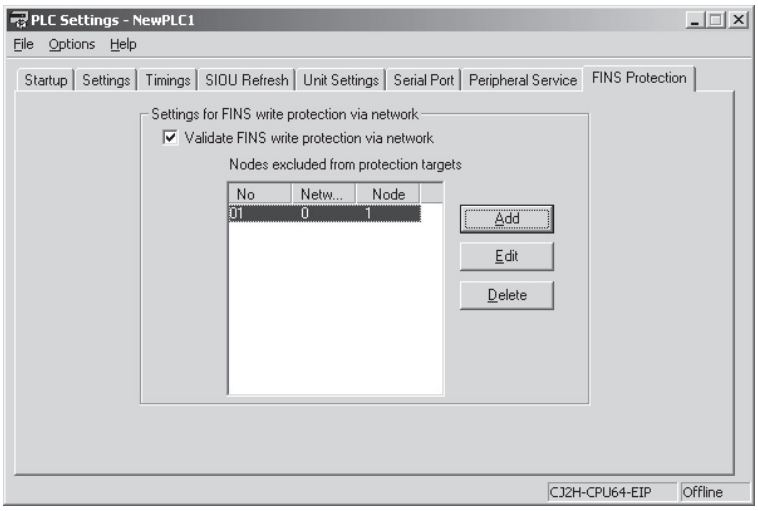
(주) 이 기능은 FINS 명령을 통한 쓰기 방지 기능입니다. 따라서 FINS 명령 이외의 데이터 링크 등에 의한 쓰기 처리와는 관계 없습니다.

● 각 패턴별 보호 여부

연결 패턴		그림(예)	쓰기 방지
직접시리얼 연결한 PC에서	직접 연결한 PLC의 경우		불가능
	게이트웨이(시리얼→네트워크)를 통해 연결한 PLC의 경우		가능
직접 네트워크 연결한 PC에서			가능
네트워크상의 다른 PLC에서			가능

● 조작

CX-Programmer에서 PLC 시스템 설정의 'FINS 보호' 탭에서 '네트워크를 통한 FINS 쓰기 방지 유효성 검사'를 선택합니다. 이 경우, 해당 CPU 유니트에 대해 네트워크를 경유(직접 시리얼 연결 이외)하여 쓰기 관련 FINS 명령을 사용한 쓰기 처리를 수행할 수 없게 됩니다.
또한 특정 노드에서의 쓰기는 허가하려는 경우는 해당 노드의 네트워크 주소 및 노드 주소, 그리고 PLC 시스템 설정의 '보호 대상에서 노드 제외'를 설정합니다(최대 32노드를 보호 해제 대상으로 설정 가능).



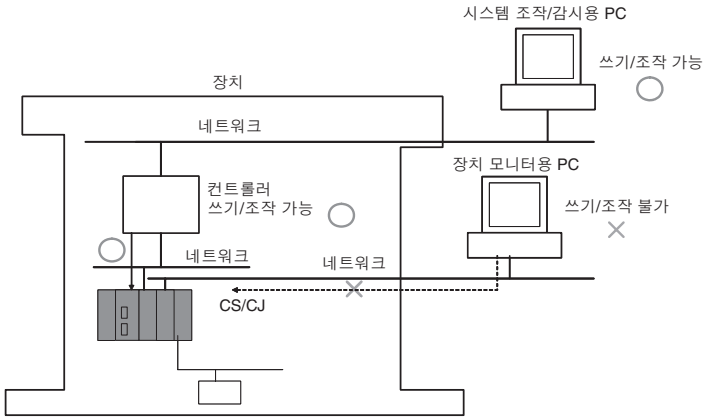
● PLC 시스템 설정

항목	설명	내용	초기값
네트워크를 통한 FINS 쓰기 방지 유효성 검사	네트워크를 경유(직접 시리얼 연결 이외)하여 FINS 명령을 사용한 CPU 유니트에 대한 쓰기 처리 금지 유효/무효 설정	0(OFF): 쓰기 방지 무효 1(ON): 쓰기 방지 유효	0(OFF): 쓰기 방지 무효
보호 대상에서 노드 제외	어느 네트워크 주소의 어느 노드에서의 FINS 쓰기를 허가할지(쓰기 방지 대상에서 제외) 설정합니다. 최대 32개 노드를 지정할 수 있습니다.(*)	—	—
	네트워크 주소: FINS 송신 원본 네트워크 주소	00~7FHex	—
	노드: FINS 송신 원본 노드 주소	01~FEHex, FFHex (FFHex는 노드 주소를 지정하지 않음)	—
보호 대상에서 제외된 노드 수 (CX-Programmer에서는 자동 계산되므로 설정하지 않음)	보호 대상에서 제외되는 노드로 지정할 노드의 수를 설정합니다. 0을 지정하면(CX-Programmer에서 위의 보호 대상에서 제외되는 노드를 하나도 지정하지 않는 경우) 로컬 노드 이외의 모든 노드에서 FINS 쓰기 처리가 금지됩니다. (*)	0~32 (00~20Hex) (0: 모든 노드가 보호 대상)	0(모든 노드가 보호 대상)

* 쓰기 방지 기능을 설정한 경우에만 유효.

● 용도

장치 내의 PLC에 대해 특정 노드에서만 쓰기를 실행할 수 있도록 시스템을 구성할 수 있습니다(예: 장치 내 컨트롤러, 사용자 측 조작/감시 PC에서만 쓰기가 가능하게 하고 싶은 경우).
이렇게 하면 네트워크를 경유하여 데이터를 모니터링하면서 PLC에 대한 예상치 못한 쓰기를 방지할 수 있으므로 시스템에 문제가 발생하는 것을 막을 수 있습니다.



● 네트워크를 경유한 쓰기 방지 대상이 되는 처리

쓰기 계통 FINS 명령

다음과 같은 CPU 유니트 대상 FINS 명령이 네트워크를 경유한 쓰기 방지 대상입니다.

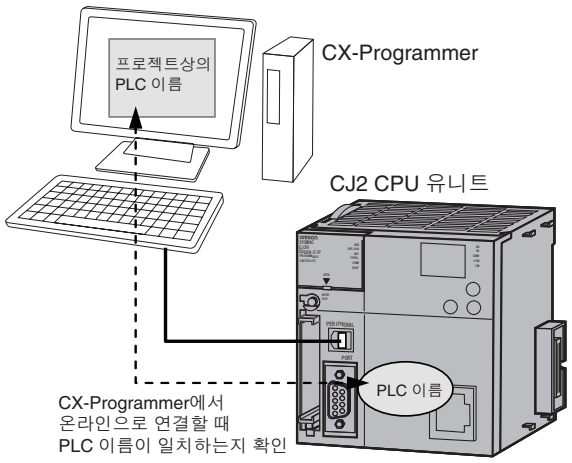
명령	명령 이름
0102Hex	I/O 메모리 영역의 쓰기
0103Hex	I/O 메모리 영역의 일괄 쓰기
0105Hex	I/O 메모리 영역의 전송
0202Hex	파라미터 영역의 쓰기
0203Hex	파라미터 영역의 일괄 쓰기(클리어)
0307Hex	프로그램 영역의 쓰기
0308Hex	프로그램 영역의 클리어
0401Hex	동작 모드 변경(운전 시작)
0402Hex	동작 모드 변경(운전 정지)
0702Hex	시간 정보 쓰기
0C01Hex	액세스 권한 획득

명령	명령 이름
2101Hex	이상 해제
2103Hex	이상 내역 포인터의 클리어
2203Hex	1개 파일 쓰기
2204Hex	파일 메모리 포맷
2205Hex	파일 삭제
2207Hex	파일 복사
2208Hex	파일명 변경
220AHex	I/O 메모리 영역←→파일 메모리 전송
220BHex	파라미터 영역←→파일 메모리 전송
220CHex	프로그램 영역←→파일 메모리 전송
2215Hex	하위 디렉토리 작성/삭제
2301Hex	강제 세트/리셋
2302Hex	전 지점에서 강제 세트/리셋 일괄 해제

10-6-5 PLC 이름 기능

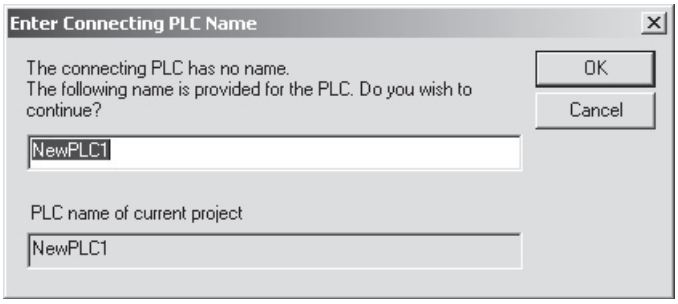
PLC 이름 기능이란?

CPU 유니트 내에 PLC 이름을 기억(등록)시킬 수 있습니다.
CX-Programmer에서 온라인으로 연결할 때 프로젝트상의 PLC 이름과 연결 대상 CPU 유니트의 PLC 이름이 일치하는지 확인할 수 있습니다.
따라서 CX-Programmer에서 잘못된 연결되는 것을 방지할 수 있습니다.



PLC 이름 등록

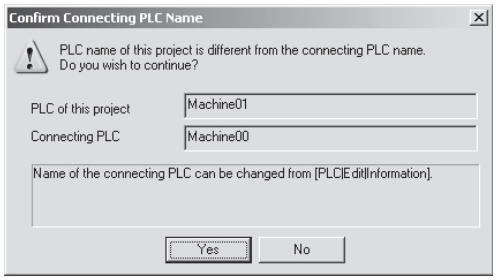
공장 출하 시 또는 메모리를 모두 삭제한 CJ2 CPU 유니트에 온라인으로 연결하면 [연결 PLC 이름 입력] 대화 상자가 표시됩니다.



온라인으로 연결된 CJ2 CPU에 등록할 PLC 이름을 입력하고 [확인] 버튼을 클릭합니다.
PLC 이름을 입력하지 않으려면 [취소] 버튼을 클릭합니다.

PLC 이름이 일치하지 않는 경우

CPU 유니트에 PLC 이름을 저장한 경우, 연결 대상 CPU 유니트 내의 PLC 이름과 프로젝트상의 PLC 이름이 일치하는지 여부를 온라인 연결 시 자동으로 확인합니다.
연결 대상 CPU 유니트 내에 저장된 PLC 이름과 프로젝트의 PLC 이름이 다르면 다음의 경고가 표시되므로 그대로 온라인 연결을 계속 유지할지 여부를 선택합니다.



- [예] 버튼
  PLC 이름이 다른 상태로 온라인 연결을 계속할 때 클릭합니다.
- [아니오] 버튼
  오프라인 상태로 변경합니다.

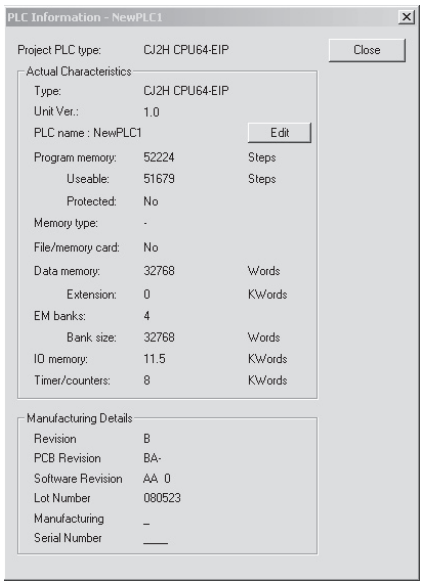
참고

PLC 이름이 일치하는지 확인하지 않으려면 PLC 속성에서 '온라인 전환 시 연결하는 PLC 이름을 확인합니다.'를 선택 취소하십시오.

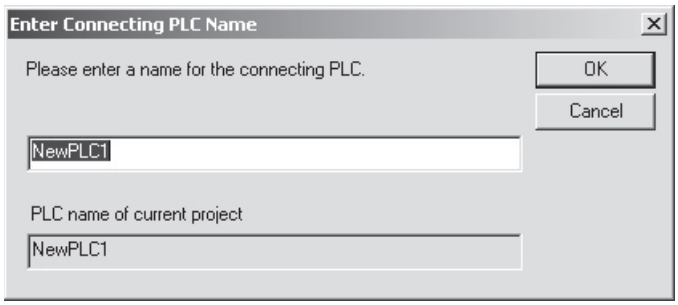
PLC 이름 변경

CJ2 CPU 유니트에 저장된 PLC 이름을 변경하려면 다음을 수행합니다.

1 **CX-Programmer에서 온라인으로 연결한 후 [PLC] 메뉴에서 [편집] – [정보]를 선택합니다.**
다음과 같은 대화 상자가 표시됩니다.



2 **[PLC 이름] 오른쪽의 [편집] 버튼을 클릭합니다.**
다음과 같은 [연결 PLC 이름 입력] 대화 상자가 표시됩니다.



3 **연결 대상 PLC에 지정할 PLC 이름을 입력한 후 [확인] 버튼을 클릭합니다.**

10-7 디버그 기능

10-7-1 강제 세트/리셋

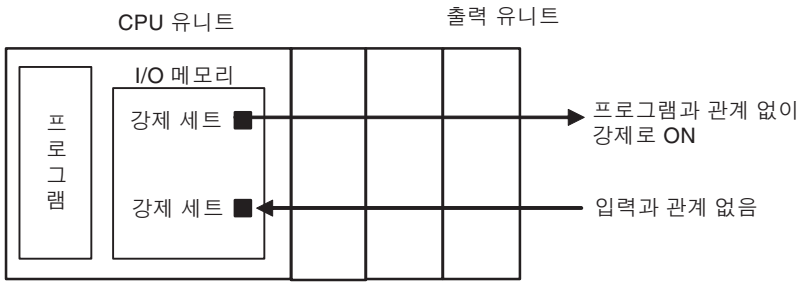
CX-Programmer에서 특정 비트(CIO 영역, 내부 보조 릴레이, 유지 릴레이, 타이머 플래그, 카운터 플래그)를 강제로(프로그램 또는 외부 입력과는 관계 없음) 세트(ON) 또는 리셋(OFF)시키는 기능입니다. 명령을 통해 이 상태를 변경할 수는 없습니다. 이 상태는 CX-Programmer에서 강제 상태 해제를 지시할 때까지 유지됩니다.

시운전할 때 입력 조건이나 출력 조건을 강제로 성립시키거나, 디버그 시 특정 조건을 성립시키고 싶은 경우 사용합니다.

모니터 또는 프로그램 모드에서 실행할 수 있으며 운전 모드에서는 실행할 수 없습니다.

(주) 동작 모드 변경(모니터←→프로그램) 시, 강제 세트 또는 리셋시킨 비트의 상태를 유지하고 싶은 경우에는 특수 보조 릴레이의 강제 세트/리셋 유지 플래그(A500.13) 및 IOM(I/O 메모리) 유지 비트(A500.12)를 동시에 ON으로 설정합니다.

전원 OFF 시, 강제 세트 또는 리셋시킨 비트의 상태를 유지하고 싶은 경우에는 특수 보조 릴레이의 강제 세트/리셋 유지 플래그(A500.13) 및 IOM(I/O 메모리) 유지 비트(A500.12)를 ON으로 설정하고, PLC 시스템 설정에서 전원 ON 시 강제 세트/리셋 데이터 유지 플래그로 설정합니다.



강제 세트/리셋 대상 영역은 다음과 같습니다.

CIO(입출력 릴레이, 데이터 링크 릴레이, CPU 고기능 유니트 릴레이, 고기능 I/O 릴레이, 내부 보조 릴레이), 내부 보조 릴레이(WR), 타이머 플래그, 유지 릴레이(HR), 카운터 플래그, EM 강제 세트/리셋 기능으로 지정한 뱅크 이후의 EM 영역(*1) (*2)

*1 EM 강제 세트/리셋 기능은 CJ2H CPU 유니트의 유니트 Ver.1.2 이상 및 CJ2M CPU 유니트와 CX-Programmer Ver.8.3 이상을 조합하여 사용할 수 있습니다.

*2 CJ2H CPU 유니트에서는 주소 자동 할당 영역으로 다음의 EM 영역을 설정한 경우도 해당 EM 뱅크에서 강제 세트/리셋이 가능해집니다.
CJ2H-CPU64/65(-EIP): EM 뱅크 3 Hex
CJ2H-CPU66(-EIP): EM 뱅크 6 ~ 9 Hex
CJ2H-CPU67(-EIP): EM 뱅크 7 ~ E Hex
CJ2H-CPU68(-EIP): EM 뱅크 11 ~ 18 Hex

● CX-Programmer의 조작

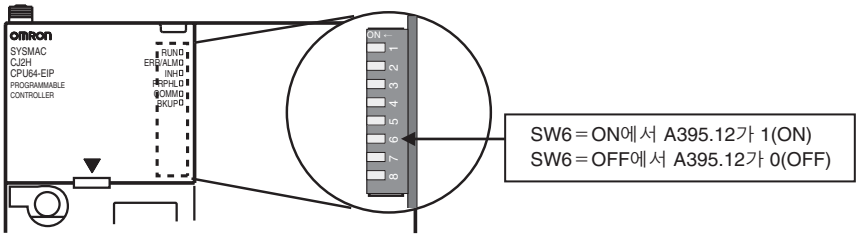
- 강제 세트/리셋 대상 접점 선택
- 강제 상태의 세트 또는 리셋 선택
- 강제 상태 해제 또는 일괄 해제 선택

● 강제 세트/리셋 유지 플래그

명칭	주소	설명	읽기/쓰기
강제 세트/리셋 유지 플래그	A500.13	CPU 유니트의 동작 모드 전환(프로그램←→모니터) 시 또는 전원 ON 시, 강제 세트/리셋 상태의 유지 여부를 지정합니다.	읽기/쓰기 가능

10-7-2 테스트 입력

CPU 유니트 전면의 딥 스위치 SW6의 ON/OFF 상태는 특수 보조 릴레이 '딥 스위치 6 상태 플래그'(A395.12)에 저장됩니다.
디버그 등을 수행할 때 입력 유니트를 사용하지 않고 수동으로 테스트용 접점을 입력할 수 있습니다.

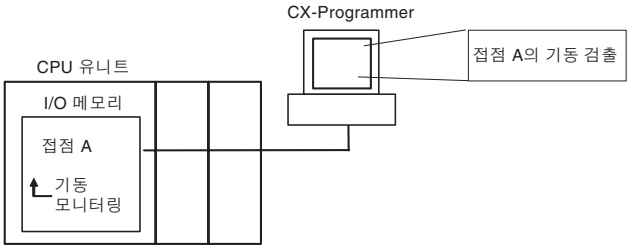


- 관련 특수 보조 릴레이

명칭	주소	설명	읽기/쓰기
딥 스위치 6 상태 플래그	A39512	CPU 유니트 전면의 딥 스위치 SW6의 ON/OFF 상태가 사이클마다 저장됩니다.	읽기 전용

10-7-3 미분 모니터

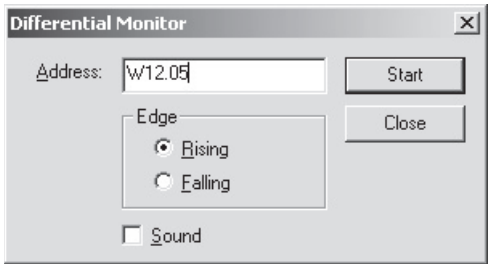
CX-Programmer에서 설정한 1접점에 대해 CPU 유니트가 접점의 기동(OFF→ON) 또는 종료(ON→OFF)를 검출하고, 결과를 특수 보조 릴레이의 미분 모니터 실행 완료 플래그(A508.09)에 반영합니다(미분 모니터 조건 성립 시, 1(ON)로 설정).
CX-Programmer는 그 결과를 감시하여 화면에 표시합니다.



● 미분 모니터의 조작(CX-Programmer 사용)

1 **미분 모니터를 실시할 접점을 마우스 오른쪽 버튼으로 클릭합니다.**

2 **[펄스 모니터]를 선택합니다.**
[펄스 모니터] 대화 상자가 표시됩니다.



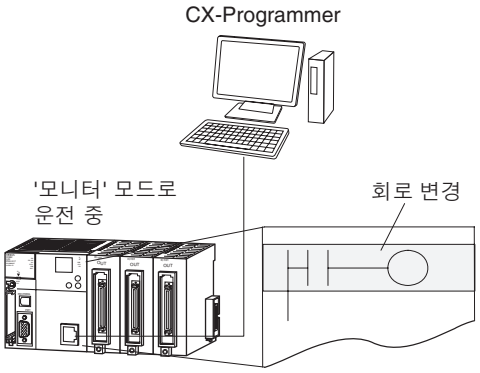
3 **[상승] 또는 [하강]을 클릭합니다.**

4 **[시작] 버튼을 클릭합니다.**
지정한 미분(기동 또는 종료)을 검출하면 부저가 울리고 카운트가 증가합니다.

5 **[정지] 버튼을 클릭합니다.**
미분 모니터가 정지합니다.

10-7-4 온라인 편집

CPU 유니트가 모니터 모드 또는 프로그램 모드일 때 CX-Programmer를 사용하여 직접 CPU 유니트 내의 프로그램 일부를 변경 및 추가하는 기능입니다. CX-Programmer에서는 여러 개의 회로 단위로 변경 및 추가합니다. CPU 유니트의 운전을 정지하지 않고 프로그램을 약간만 변경하고 싶을 때 사용합니다.
또한 서로 다른 태스크를 여러 개의 CX-Programmer에서 동시에 온라인 편집할 수도 있습니다.



모니터 모드일 때 온라인 편집을 수행하면 CPU 유니트의 프로그램을 재기입했을 때 CPU 유니트의 사이클 시간이 1사이클~여러 사이클에 걸쳐 늘어납니다.

또한 온라인 편집 완료 후 실행 결과를 백업 메모리에 기록하기 때문에 사이클 시간이 늘어납니다. 백업이 진행되는 동안 BKUP LED가 점등하고 CX-Programmer에 진행 상황이 표시됩니다.
1사이클당 전체 사이클 시간은 1ms를 기준으로 하십시오.

또한 백업 완료를 기다리지 않고 연속으로 온라인 편집을 수행하는 횟수에 제한이 있습니다. 제한 횟수는 온라인 편집의 내용에 따라 다르지만 400회를 기준으로 하십시오.
온라인 편집 제한 횟수를 초과하면 CX-Programmer에 메시지가 표시됩니다. 이런 경우, CPU 유니트가 백업 처리를 완료할 때까지 온라인 편집을 할 수 없습니다. 백업 처리 완료 후 다시 온라인 편집을 계속할 수 있습니다.

사용상의 주의 사항

'모니터' 모드에서 온라인 편집을 통한 프로그램 재기입 시, 사이클 시간이 늘어나며, 그 결과 PLC 시스템 설정에서 설정한 '사이클 시간 감시 시간'을 초과하는 일이 없도록 주의하십시오. 초과하면 '사이클 시간 초과' 에러가 발생하고 CPU 유니트의 운전이 정지됩니다. 다시 운전하는 경우, '프로그램' 모드로 변경한 후 '운전' 또는 '모니터' 모드로 돌아가십시오.

(주) 1 온라인 편집의 대상으로 한 회로에 포함된 기동 미분 명령(DIFU/@ 표시가 있는 명령) 및 종료 미분 명령(DIFD/% 표시가 있는 명령)의 내부 상태(미분 플래그)는 초기화됩니다.
2 온라인 편집을 실행한 태스크에 블록 프로그램아 포함된 경우,
WAIT/TIMW/TIMWX/CNTWE/CNTWX/TMHW/TMHWX 명령에 의한 대기 상태, BPPS 명령에 의한 일시 정지 상태는 해제되고 다음 번 블록 프로그램은 선두부터 실행됩니다.

● CX-Programmer를 통한 온라인 편집 조작

1 **수정할 회로를 화면에 표시합니다.**

2 **수정할 회로를 하나 이상 선택합니다.**

3 **[프로그램] – [온라인 편집] – [시작]을 선택합니다.**

4 **회로를 편집합니다.**

5 **[프로그램] – [온라인 편집] – [변경 사항 전송]을 선택합니다.**
프로그램을 확인하고 프로그램에 에러가 없으면 변경된 회로가 PLC에 전송됩니다. (이때 CPU 유니트의 프로그램을 덮어 쓰게 되고 사이클 시간이 연장됩니다.)

⚠ 주 의

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



참고

온라인 편집 금지 기능

기계 제어의 응답 성능을 위해 특정 타이밍에서의 응답 성능을 유지하고 싶은 경우, 특정 사이클에서만 온라인 편집을 금지 상태로 만들 수 있습니다.
온라인 편집을 금지 상태로 설정하면 제삼자의 온라인 편집으로 인해 예기치 않게 사이클 시간이 연장되는 것을 막을 수 있습니다.

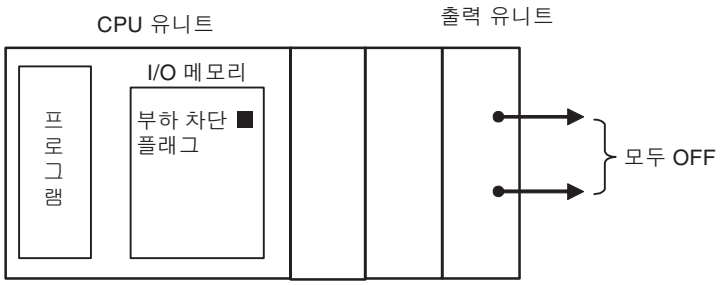
- 온라인 편집을 금지하는 설정
  온라인 편집을 금지하려면 다음 2종류의 설정을 수행합니다.
  (1) '온라인 편집 금지 중 플래그 액세스 중 비밀번호'
  (A527.00~527.07)에 5AHex 설정
  (2) '온라인 편집 금지 중 플래그'(A527.09)를 1(ON)로 설정
- 온라인 편집을 금지할 때의 동작
  CX-Programmer에서 온라인 편집 실행 요구가 있는 경우 온라인 편집은 실행되지 않고 대기 상태가 됩니다. 이때, '온라인 편집 대기 중 플래그'(A201.10)가 1(ON)이 됩니다.
  그 후, 사용자가 '온라인 편집 금지 중 플래그'(A527.09)를 0(OFF)으로 하면 그 시점에서 온라인 편집 처리가 실행됩니다.
  또한 이미 온라인 편집이 대기 중인 상태에서 온라인 편집 대상을 추가하기 위해 온라인 편집을 조작해도 수행되지 않습니다.

● 관련 특수 보조 릴레이

명칭	주소	내용
온라인 편집 금지 중 플래그 액세스용 비밀번호	A527.00~527.07	온라인 편집 금지 중 플래그(A527.09)가 유효인지, 무효인지 지정합니다. 5AHex: 온라인 편집 금지 중 플래그 유효 5AHex 이외: 온라인 편집 금지 중 플래그 무효
온라인 편집 금지 중 플래그	A527.09	온라인 편집을 금지하는 경우 이 플래그를 1(ON)로 설정하십시오. 이 플래그의 유효/무효는 A527.00~A527.07(온라인 편집 금지 중 플래그 액세스용 비밀번호 영역)의 설정에 좌우됩니다.
온라인 편집 대기 중 플래그	A201.10	온라인 편집 처리가 대기 중일 때 1(ON)이 됩니다.

10-7-5 부하 차단

운전('운전' 또는 '모니터' 모드) 상태에서 이상 시 긴급 처리로 I/O 메모리상의 출력 릴레이의 상태와 관계없이 출력 유니트의 모든 출력을 OFF로 하고 싶은 경우, 특수 보조 릴레이의 부하 차단 플래그(A500.15)를 ON으로 설정합니다.
이때, CPU 유니트 전면의 INH LED가 점등합니다.
부하 차단 플래그는 배터리가 있는 상태에서 전원 ON→OFF→ON 시에도 유지됩니다.



참고

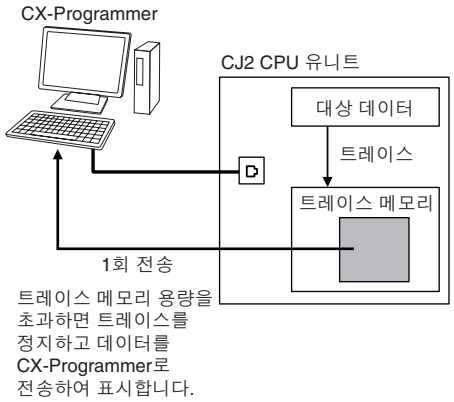
평상시('IOM(I/O 메모리) 유지 비트'(A500.12)를 OFF로 설정한 경우), 동작 모드를 운전('운전' 모드 또는 '모니터' 모드)에서 정지('프로그램' 모드)로 변경하면 출력 유니트의 모든 출력이 클리어(OFF)됩니다. 단, 사용자 조작을 통해 I/O 메모리의 출력 릴레이를 1(ON)로 설정하면 I/O 리프레시되고 출력 접점은 ON이 됩니다.
동작 모드에 관계 없이 출력 유니트의 출력을 모두 OFF로 할 경우 이 부하 차단 기능을 사용합니다.

10-7-6 데이터 트레이스

지정된 I/O 메모리의 데이터를 특정 조건에서 샘플링하고 트레이스 메모리에 저장해둔 후, CX-Programmer에서 읽어서 확인하고 파일로 저장하는 기능입니다.
트레이스 방법은 다음 2종류가 가능합니다.

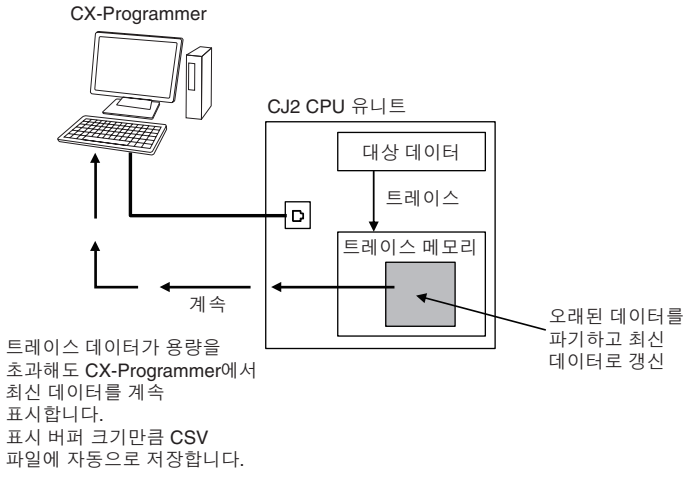
트레이스

주로 단시간에 고속 데이터 수집을 수행할 때(디버그 시 등) 사용합니다.
트레이스 메모리가 용량을 초과하면 트레이스를 정지하고, 트레이스 메모리 분량의 트레이스 결과 데이터를 표시하는 동시에 CSV 파일로 저장합니다.



트레이스(연속)

주로 장시간에 걸쳐 고속 데이터 수집을 수행할 때(가동 중 등) 사용합니다.
트레이스 메모리가 용량을 초과해도 계속 트레이스를 수행하며, 동시에 CX-Programmer에 트레이스 데이터를 계속 읽어옵니다.
표시 버퍼 크기에 도달하면 자동으로 CSV 파일에 저장합니다.
그 결과, 트레이스를 시작한 후 정지할 때까지 장기간의 트레이스 결과 데이터를 하나 이상의 CSV 파일로 저장할 수 있습니다.



● 샘플링 가능한 대상과 타이밍

데이터 개수

샘플링 대상으로 지정할 수 있는 I/O 메모리의 데이터 개수는 최대 31접점, 16채널입니다.

데이터 형식

샘플링 대상으로 지정할 수 있는 데이터 형식은 다음과 같습니다.

- BOOL(비트, 접점)
- UINT(부호 없는 1채널 BIN)
- UDINT(부호 없는 2채널 BIN)
- ULINT(부호 없는 4채널 BIN)
- INT(부호 있는 1채널 BIN)
- DINT(부호 있는 2채널 BIN)
- LINT(부호 있는 4채널 BIN)
- UINT BCD(부호 없는 1채널 BCD)
- UDINT BCD(부호 없는 2채널 BCD)
- ULINT BCD(부호 없는 4채널 BCD)
- REAL(부동 소수점 2채널)
- LREAL(부동 소수점 4채널)
- CHANNEL(채널)
- WORD(16진수 1채널)
- DWORD(16진수 2채널)
- LWORD(16진수 4채널)

타이밍

I/O 메모리의 데이터는 다음 타이밍에 샘플링됩니다.

- 지정 샘플링 시간(1~2550ms, 1ms 단위)
- 1사이클에 1회
- 트레이스 메모리 샘플링 명령(TRSM 명령) 실행 시
- 동기 제어 주기마다(0.5~10.0ms, 0.1ms 단위) (*)

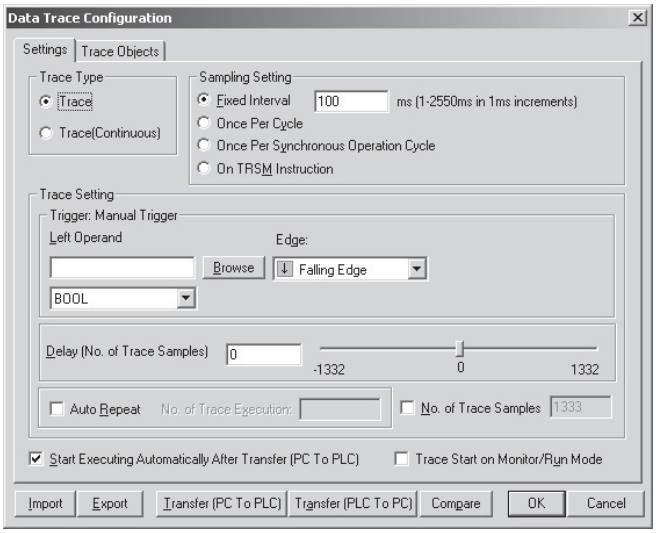
  * CJ2H CPU 유니트만 해당하며 동기 제어 주기가 3.0ms 미만인 경우는 트레이스(연속)를 사용할 수 없습니다.

● 데이터 트레이스 실행 순서

데이터 트레이스의 실행 순서는 다음과 같습니다.

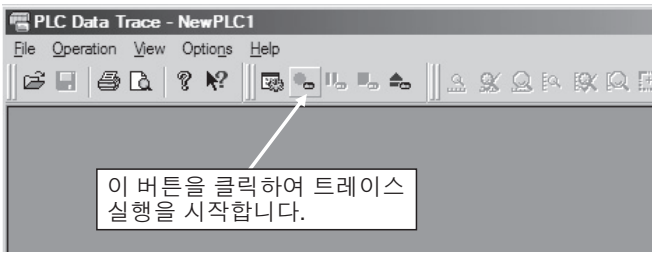
1 CX-Programmer에서 트레이스 파라미터를 설정합니다.

([PLC]－[데이터 트레이스] 또는 [시간 차트 모니터]를 선택하여 데이터 트레이스 창을 연 후 [실행]－[설정]으로 설정)

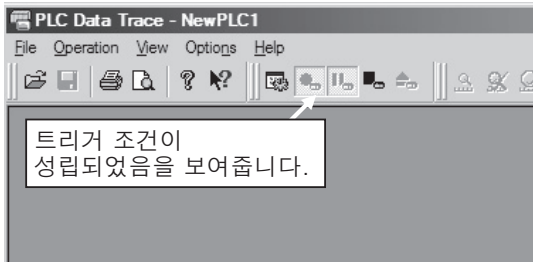


- '트레이스'를 실행할 때
  [트레이스 형식]에서 '트레이스'를 선택합니다.
  [트레이스 개체] 탭에서 샘플링 대상 데이터의 주소를 설정합니다. 변수로 지정할 수도 있습니다.
  [설정] 탭의 트레이스 설정에서 트리거 조건, 딜레이값 등을 설정합니다.
- '트레이스(연속)'를 실행할 때
  [트레이스 형식]에서 '트레이스(연속)'를 선택합니다.
  [트레이스 개체] 탭에서 샘플링 대상 데이터의 주소를 설정합니다. 변수로 지정할 수도 있습니다.
  [트레이스 설정]에서 화면에 표시할 버퍼 크기, 트레이스 시간, 수집한 데이터를 CSV 파일로 저장할 폴더를 지정합니다.

2 트레이스 시작 플래그 A508.15를 ON으로 설정하거나 아래의 버튼을 클릭하여 트레이스를 시작합니다.

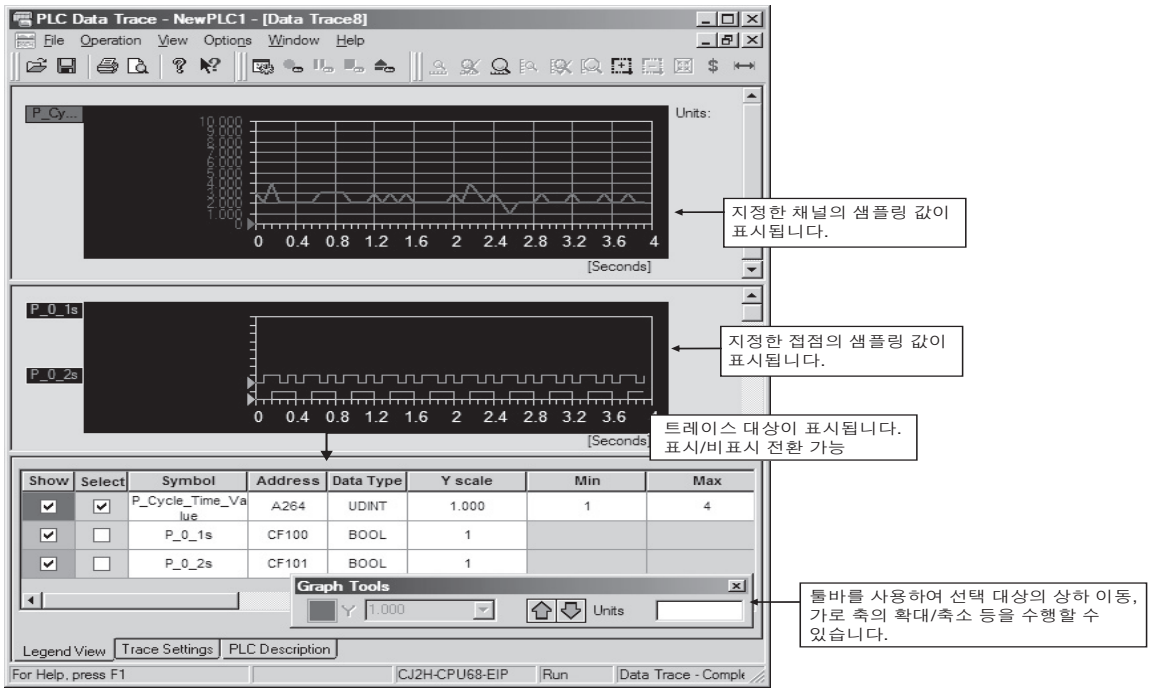


트레이스 트리거 조건이 성립되면 다음과 같이 표시됩니다.



● 데이터 트레이스 창

데이터 트레이스 창의 개요는 다음과 같습니다.



조작 방법 및 설정에 대한 자세한 내용은 "CX-Programmer 조작 설명서"(SBCA-337)를 참조하십시오.

● CJ2 CPU 유니트의 데이터 트레이스 사양

CJ1 CPU 유니트와 비교한 내용을 다음과 같이 정리했습니다.

항목	CJ2H CPU 유니트	CJ2M CPU 유니트	CJ1 CPU 유니트
메모리 용량	CPU64(-EIP)/65(-EIP): 8000워드 CPU66(-EIP): 16000워드 CPU67(-EIP)/68(-EIP): 32000워드 데이터 트레이스용 영역 대신 EM 영역을 사용하는 것도 가능 (최대 32K 워드×25뱅크까지)	8000워드 데이터 트레이스용 영역 대신 EM 영역을 사용하는 것도 가능 (최대 32K 워드×4뱅크까지)	4000워드
샘플링 수	샘플링 접점 수: 31접점, 샘플링 채널 수: 최대 16점 1CH 길이 데이터=16점 2CH 길이 데이터=8점 4CH 길이 데이터＝4점	샘플링 접점 수=31접점, 샘플링 채널 수: 최대 16점 1CH 데이터 길이: 16점 2CH 데이터 길이: 8점 4CH 데이터 길이: 4점	샘플링 접점 수: 31접점, 샘플링 채널 수: 최대 6점
주기 (고정 주기인 경우)	1~2550ms(1ms 단위)	1~2550ms(1ms 단위)	10~2550ms(10ms 단위)
운전 시작 시의 자동 시작	가능	가능	없음
연속 트레이스	가능 트레이스 메모리가 용량을 초과하기 전에 트레이스 데이터를 CX-Programmer에 읽어오면 트레이스를 계속할 수 있음	가능 트레이스 메모리가 용량을 초과하기 전에 트레이스 데이터를 CX-Programmer에 읽어오면 트레이스를 계속할 수 있음	없음
트리거 조건을 설정할 수 있는 데이터 길이	1CH 길이/2CH 길이/4CH 길이 데이터	1CH 길이/2CH 길이/4CH 길이 데이터	1CH 길이 데이터
트리거 성립 조건	=, >, , <, , ≒ 사용 가능	=, >, , <, , ≒ 사용 가능	'='만 사용 가능
딜레이값	－32768 ~＋32767	－32768 ~＋32767	－1999 ~＋2000

10-7-7 에러 발생 시 정지 위치 저장 기능

래더 프로그램의 명령은 입력된 파라미터 등이 범위 밖일 때 에러(P_ER) 플래그를 1(ON)로 합니다. 명령 실행 시 이 플래그를 참조하여 명령 에러 발생 여부를 확인할 수 있습니다.

PLC 시스템 설정에서 [명령 오류 시 CPU 정지]를 선택한 경우, 다음과 같은 명령 에러가 발생했을 때 '프로그램 에러'를 발생시켜 운전을 정지시킬 수 있습니다.
이 기능을 이용하여 명령 에러 발생 유무를 확인할 수 있습니다.

프로그램 에러 발생 요인이 되는 명령 에러(A295)

명칭	주소	내용
명령 처리 에러 플래그	A295.08	PLC 시스템 설정의 [명령 오류 시 CPU 정지]를 설정하여 에러(P_ER) 플래그를 1(ON)로 한 경우 1(ON)이 됩니다.
DM/EM 간접 BCD 에러 플래그	A295.09	DM/EM 간접(BCD 모드)으로 지정한 데이터가 BCD가 아닌 경우 1(ON)이 됩니다.
잘못된 영역 액세스 에러 플래그	A295.10	잘못된 영역 액세스(시스템 영역, EM 파일로 전환된 영역에 대한 R/W 액세스, 쓰기 방지 영역에 대한 쓰기 액세스, 간접 DM/EM(BCD 모드)에서 BCD 에러)가 발생한 경우 1(ON)이 됩니다.

'프로그램 에러'에 의해 운전이 정지되면 정지 위치의 태스크가 다음과 같이 저장됩니다.

프로그램 정지 시 태스크 번호(A294CH)

명칭	주소	내용
프로그램 정지 시 태스크 번호	A294CH	프로그램 에러에 의해 정지된 경우 정지 위치의 태스크 유형, 태스크 번호가 저장됩니다. 0000~007FHex(태스크 번호 0~127에 대응) 8000~80FFHex(인터럽트 태스크 번호 0~255에 대응) (추가 태스크 포함)

운전 정지 이상이 어느 태스크에서 발생했는지 확인할 수 있습니다. 운전 정지 이상이 해제되면 클리어됩니다.

프로그램 에러 발생 위치(A299: 프로그램 주소 상위, A298: 프로그램 주소 하위)

프로그램 에러에 의해 실행이 정지된 경우 정지 위치에 대해 다음과 같은 프로그램 주소가 저장됩니다.

(주) 래더 언어로 기재했을 때만 유효하며 ST 언어나 SFC 언어를 사용한 경우는 사용할 수 없습니다.

- 래더 프로그램에서 발생한 경우:
  프로그램 선두부터 계산한 프로그램의 시작 프로그램 주소
- 래더 작업 프로그램에서 발생한 경우:
  래더 작업 선두부터 계산한 프로그램의 시작 프로그램 주소
- 트랜지션 프로그램에서 발생한 경우:
  트랜지션 래더 선두부터 계산한 프로그램의 시작 프로그램 주소
- 펑션 블록에서 발생한 경우:
  펑션 블록 선두부터 계산한 프로그램의 시작 프로그램 주소

10-7-8 고장 진단 명령(FAL 명령, FALS 명령) 기능

사용자 정의 이상을 검출하는 명령입니다. 이상 발생 시, CPU 유니트의 운전을 계속하는 명령(FAL 명령)과 운전을 정지하는 명령(FALS 명령)이 있습니다.
사용자가 '대상 시스템의 이상(고장)'이라고 판단하는 입력 조건이 성립했을 때 고장 진단 명령(FAL 명령 또는 FALS 명령)을 실행하고 다음과 같이 처리합니다.

- 특수 보조 릴레이(FAL 이상 플래그/FALS 이상 플래그) 설정
- 동시에 이상(고장)에 대응하는 사용자 정의의 고장 코드를 특수 보조 릴레이에 설정
- 동시에 이상 내역 영역에 고장 코드와 발생 시각을 저장
- 동시에 CPU 유니트 전면의 LED 점등/점멸

위의 처리를 실행 후, FAL 명령은 이상(고장) 발생 시 CPU 유니트의 운전을 계속합니다.
FALS 명령은 CPU 유니트의 운전을 정지시킵니다.

FAL(운전 계속 고장 진단) 명령



A가 ON이 되면 FAL 번호 002의 고장이 발생한 것으로 판단하여 특수 보조 릴레이 A402.15를 ON하고 A360.02(FAL 번호 002에 대응)를 ON합니다. 프로그램의 운전은 계속됩니다.
FAL 명령 실행에 의한 이상을 해제하기 위해서는 FAL00(FAL 번호 00) 명령을 실행하거나 CX-Programmer에서 이상 읽기/해제 조작을 합니다.

FALS(운전 정지 고장 진단) 명령



B가 ON이 되면 FALS 번호 003의 고장이 발생한 것으로 판단하여 특수 보조 릴레이 A401.06을 ON합니다. 프로그램의 운전은 정지됩니다.
FALS 명령 실행에 의한 이상을 해제하기 위해서는 이상 원인 제거 후 CX-Programmer에서 이상 읽기/해제 조작을 합니다.

10-7-9 시스템 이상을 고의로 발생시키는 기능

FAL 명령 또는 FALS 명령 실행에 의해 시스템에서 운전 계속 이상 또는 운전 정지 이상을 고의로 발생시킬 수 있습니다.
사용자 응용 프로그램의 디버그 시, 고의로 이상을 발생시켜 PT(프로그래머블 터미널) 등의 HMI에 에러 메시지가 정확하게 표시되는지 여부를 확인할 때 등에 사용합니다.

다음 방법으로 확인합니다.

1 **특수 보조 릴레이 A529CH에 이 기능용으로 사용할 FAL/FALS 번호를 설정합니다.**
A529CH에 저장하는 번호는 FAL 명령과 FALS 명령에서 공통입니다.

2 **이 FAL/FALS 번호를 FAL 명령 또는 FALS 명령의 제1 피연산자 N으로 지정합니다.**

3 **발생시킬 고장 코드, 이상 내용을 FAL 명령 또는 FALS 명령의 제2 피연산자 S, S+1로 지정합니다.**
FAL 명령에서는 운전 계속 이상의 고장 코드와 이상 내용을 지정합니다. FALS 명령에서는 운전 정지 이상의 고장 코드와 이상 내용을 지정합니다.

또한 복수의 시스템 이상을 고의로 발생시키는 경우, A529CH 내의 같은 값을 제1 피연산자에 지정하고 다른 S, S+1의 내용으로 여러 개의 FAL/FALS 명령을 실행합니다.

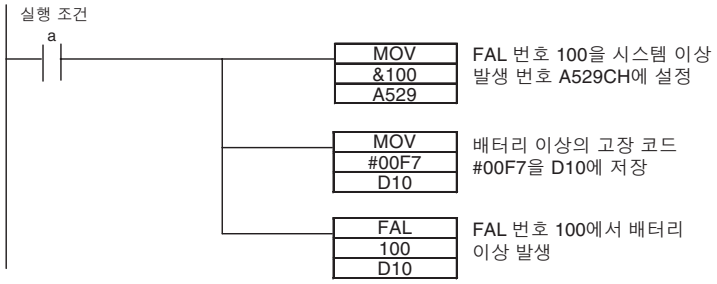
사용상의 주의 사항

이 기능은 CPU 유니트에서 검출할 이상을 강제로 발생시켜서 응용 프로그램 내에서 검출 처리가 정확하게 동작하는지 확인하기 위한 기능입니다.
시스템 가동 시에는 이 기능의 FAL 명령 또는 FALS 명령을 삭제하거나 기동 조건에 상시 OFF(P_Off) 신호를 입력하는 등의 작업을 수행하여 이 기능이 동작하지 않도록 하십시오.

● 관련 특수 보조 릴레이

명칭	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. 0001~01FFHex: 1~511 중 하나 0000, 0200~FFFF Hex: 시스템 이상 발생의 FAL/FALS 번호 설정 안 함

● 예: 배터리 이상을 발생시키는 경우



사용상의 주의 사항

FAL 명령 또는 FALS 명령을 실행하여 고의로 발생시킨 시스템 운전 계속 이상 또는 시스템 운전 정지 이상을 해제하는 조작은 해당하는 이상이 실제로 발생한 경우와 동일한 조작이어야 한다는 점에 주의하십시오.
이상 해제 방법에 대해서는 "CJ 시리즈 CJ2 CPU 유니트 사용 설명서 하드웨어편"(SCBA-349)의 '제6장 문제 해결'을 참조하십시오.
또한 전원을 껐다가 다시 켜면 FAL 명령 또는 FALS 명령을 통해 고의로 발생시킨 모든 시스템 이상을 해제할 수 있습니다.

10-7-10 고장 지점 검출 명령(FPD 명령) 기능

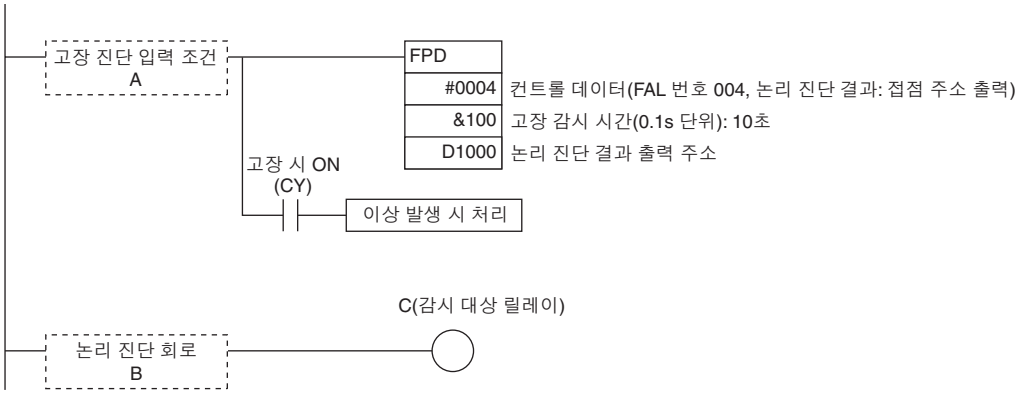
하나의 회로에 대해 시간 진단(지정한 출력 접점이 설정 시간 내에 ON되는지 여부를 감시)과 논리 진단(지정한 출력 접점이 ON되지 않는 원인이 된 입력 접점을 추출)을 수행합니다.

시간 진단

어떤 조건이 성립한 후 지정한 출력 접점이 ON이 될 때까지의 시간을 감시하고, 지정한 시간 내에 ON이 되지 않는 경우는 CY 플래그를 ON으로 합니다. 이 CY 플래그를 입력 조건으로 해서 이상 발생 시 처리의 회로를 실행할 수 있습니다. 또한 지정한 대로 FAL 이상을 발생시킬 수 있습니다.
FAL 이상을 발생시킨 경우, FAL 명령 실행과 동시에 이상 메시지를 등록하고 CX-Programmer에 표시되도록 할 수 있습니다. 이때, 논리 진단의 출력 방식이 메시지 텍스트 출력으로 지정되어 있으면 원인이 된 입력 접점의 주소를 메시지 내용에 저장할 수도 있습니다.
또한 지정한 출력 접점이 ON으로 되기까지의 실제 시간을 계측하고 그 값을 바탕으로 감시 시간을 자동 설정할 수도 있습니다(티칭 동작).

논리 진단

지정한 출력 접점이 ON이 되지 않는 원인이 된 입력 접점을 추출하여 결과를 출력합니다. 출력 방식은 접점 주소 출력 또는 메시지 텍스트 출력 중에 선택할 수 있습니다.
접점 주소 출력을 선택하면 원인이 된 접점 주소를 인덱스 레지스터에 전송하여 별도로 사용할 수 있습니다.
메시지 텍스트 출력을 선택하면 시간 감시에서 FAL 이상을 발생시킬 때 그와 동시에 CX-Programmer에 이상 메시지를 표시하도록 할 수 있습니다.



- **시간 감시**

  A가 ON으로 설정된 후 10초 이내에 C가 ON이 되는지 감시합니다. ON으로 되지 않으면 고장이라고 판단하고 CY 플래그가 ON이 됩니다. 그 결과 이상 발생 시 처리를 실행합니다. 또한 FAL 번호 004의 FAL 이상(운전 계속 이상)을 발생시킵니다.

- **논리 진단**

  C가 ON으로 되지 않는 요인을 B 중에서 찾아서 요인이 된 접점 주소를 D1000에 저장합니다.

● 관련 특수 보조 릴레이

명칭	주소	내용
고장 코드	A400	이상 발생 시, 고장 코드 저장
FAL 이상 플래그	A402.15	FAL 명령이 실행되었을 때 ON
FALS 이상 플래그	A401.06	FALS 명령이 실행되었을 때 ON
FAL 번호	A360~391	FAL 명령이 실행되면 FAL 번호에 대응하는 해당 비트가 ON
이상 내역 저장 영역	A100~A199	이상 발생 시마다 이상 내역 정보가 저장되며 최신 20개 저장
이상 내역 저장 개수	A300	이상 발생 시마다 1씩 증가
이상 내역 정보 클리어	A500.14	이 플래그가 ON이 되면 이상 내역 저장 개수가 클리어됨
FPD 명령용 티칭 릴레이	A598.00	ON 시, FPD 명령의 이상 감시 시간 자동 설정 동작 실행

10-8 유니트 간 동기 제어 기능

10-8-1 개요

유니트 간 동기 제어 기능이란?

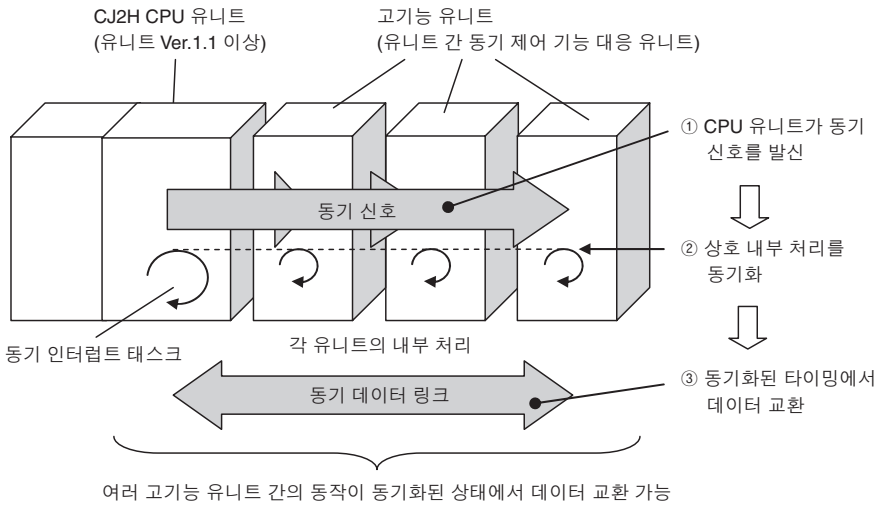
CPU 유니트가 정해진 주기로 발신하는 동기 신호에 의해 CPU 유니트와 복수의 고기능 유니트 간에 각 내부 처리의 시작 타이밍 및 데이터 교환을 동기화하는 기능입니다.
CJ2H CPU 유니트의 유니트 Ver.1.1 이상에서 사용할 수 있습니다.

- **내부 처리 시작 동기화**

  CPU 유니트는 고기능 유니트와의 데이터 교환과 동기하여 인터럽트 태스크를 실행할 수 있습니다. 복수의 고기능 유니트가 CPU 유니트의 동기 신호에 맞춰 상호 내부 처리 시작 타이밍을 동기화합니다.

- **데이터 교환 동기화**

  CPU 유니트와 복수의 고기능 유니트 사이 및 고기능 유니트끼리 동기 신호 타이밍에 데이터를 교환합니다.



사용상의 주의 사항

유니트 간 동기 제어 기능을 사용하기 위해서는 다음과 같은 조합이 되어야 합니다.

- CJ2H CPU 유니트 CJ2H-CPU6□(-EIP)의 유니트 Ver.1.1 이상
- 유니트 간 동기 제어 기능에 대응하는 고기능 유니트(이하 '동기 유니트')

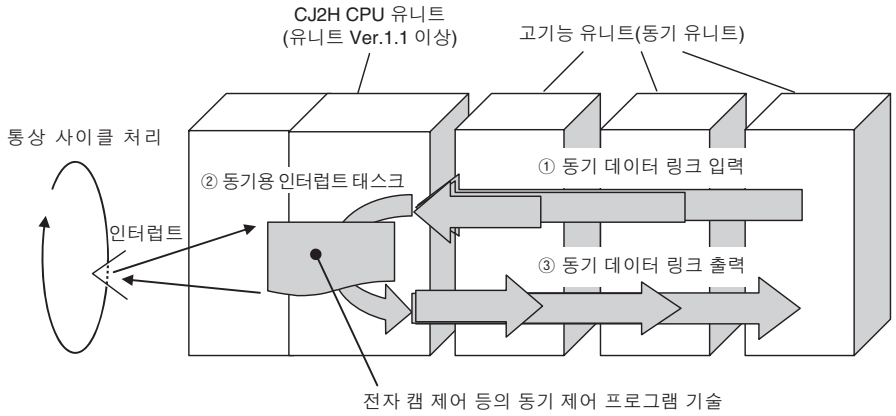
유니트 간 동기 제어의 대략적인 흐름

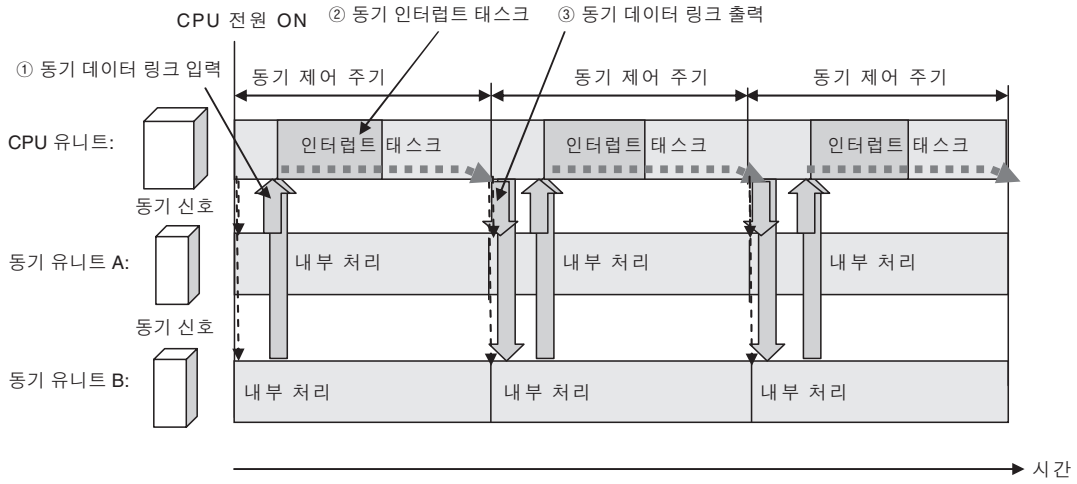
유니트 간 동기 제어 기능은 다음과 같은 흐름으로 실행됩니다.

1 각 동기 유니트는 상호 내부 처리를 동기화시킨 상태에서 정해진 주기('동기 제어 주기')로 일제히 CPU 유니트에 동기 데이터를 보냅니다('동기 데이터 링크' CPU 유니트로 입력) (①).

2 CPU 유니트는 동기 데이터를 입력하고 동기용 인터럽트 태스크('동기 인터럽트 태스크')를 실행합니다(②).

3 CPU 유니트는 그 결과를 각 동기 유니트에 일제히 출력합니다('동기 데이터 링크' CPU 유니트에서 출력) (③).





항목		설명
동기 데이터 링크		CPU 유니트의 사이클 시간과 관계 없이 정해진 주기(동기 제어 주기)에 따라 CPU 유니트와 동기 유니트 간에 교환되는 데이터 링크. 동기 데이터 링크용으로 할당된 특정 영역(동기 데이터 링크 영역)의 데이터가 동기 데이터로 교환됩니다.
	동기 데이터 링크 입력	동기 유니트에서 CPU 유니트로 보내는 동기 데이터
	동기 데이터 링크 출력	CPU 유니트에서 동기 유니트로 보내는 동기 데이터
동기 제어 주기		동기 신호가 발신되고 각 동기 유니트의 내부 처리를 시작하는 타이밍이 동기화되는 주기. 동기 데이터 링크의 갱신 주기이기도 합니다.
동기 인터럽트 태스크		동기 인터럽트 태스크는 동기 유니트에서 CPU 유니트로 동기 데이터 링크의 입력 리프레시 후에 실행되는 인터럽트 태스크입니다. 인터럽트 태스크 번호 2(예약 인터럽트 0)를 사용하여 설정. 주. 유니트 간 동기 제어 기능은 동기 인터럽트 태스크 없이도 사용 가능.

유니트 간 동기 제어 기능의 설정 항목 개요

유니트 간 동기 제어 기능을 사용할 때는 다음 항목을 설정합니다.

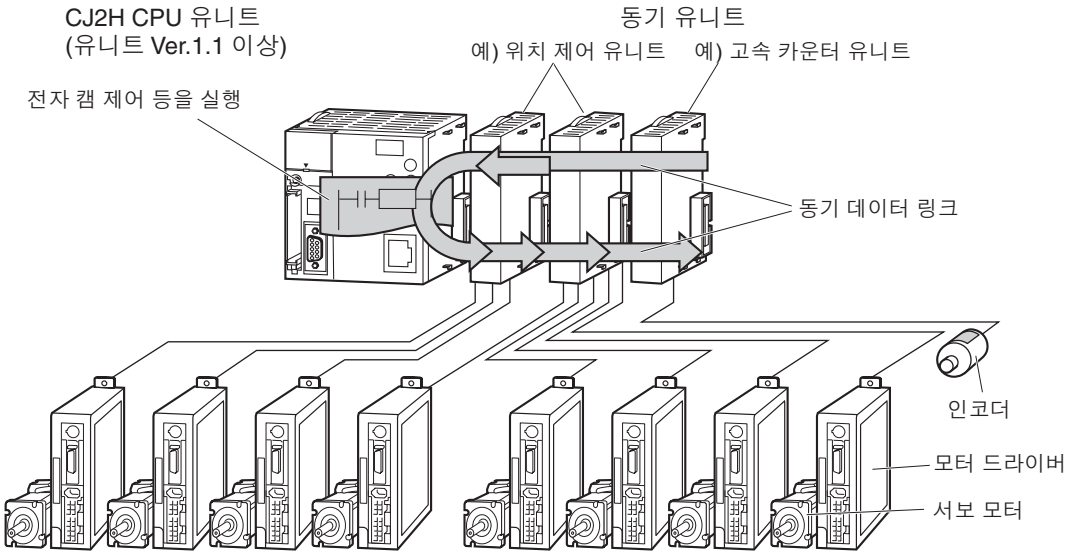
설정 항목	설정 내용	설정 방법
유니트 간 동기 제어 기능	사용 여부	PLC 시스템 설정
동기 데이터 링크	• 동기 유니트로 등록할 고기능 유니트 • 할당 영역 지정	PLC 시스템 설정
	각 동기 유니트와의 동기 데이터 링크 내용 설정	PLC 시스템 설정으로 기동된 각 동기 유니트의 동기 데이터 링크 내용 설정
동기 제어 주기	주기	PLC 시스템 설정
동기 인터럽트 태스크	실행되는 프로그램	프로그램 속성의 태스크 유형에서 '인터럽트 태스크 번호 2(예약 인터럽트 0)'를 선택 주. 유니트 간 동기 제어 기능은 동기 인터럽트 태스크 없이도 사용 가능

용도 예

● 복수의 서보 모터 동기 제어

인코더의 동작에 따라 복수의 서보 모터를 동기 제어합니다.

예) 고속 카운터 유니트에서 입력한 동기 데이터에 대해 전자 캠 등의 동기 제어 프로그램을 동기 인터럽트 태스크로 실행하고, 그 처리 결과를 동기 데이터로 복수의 위치 제어 유니트에 출력하여 위치를 결정합니다.



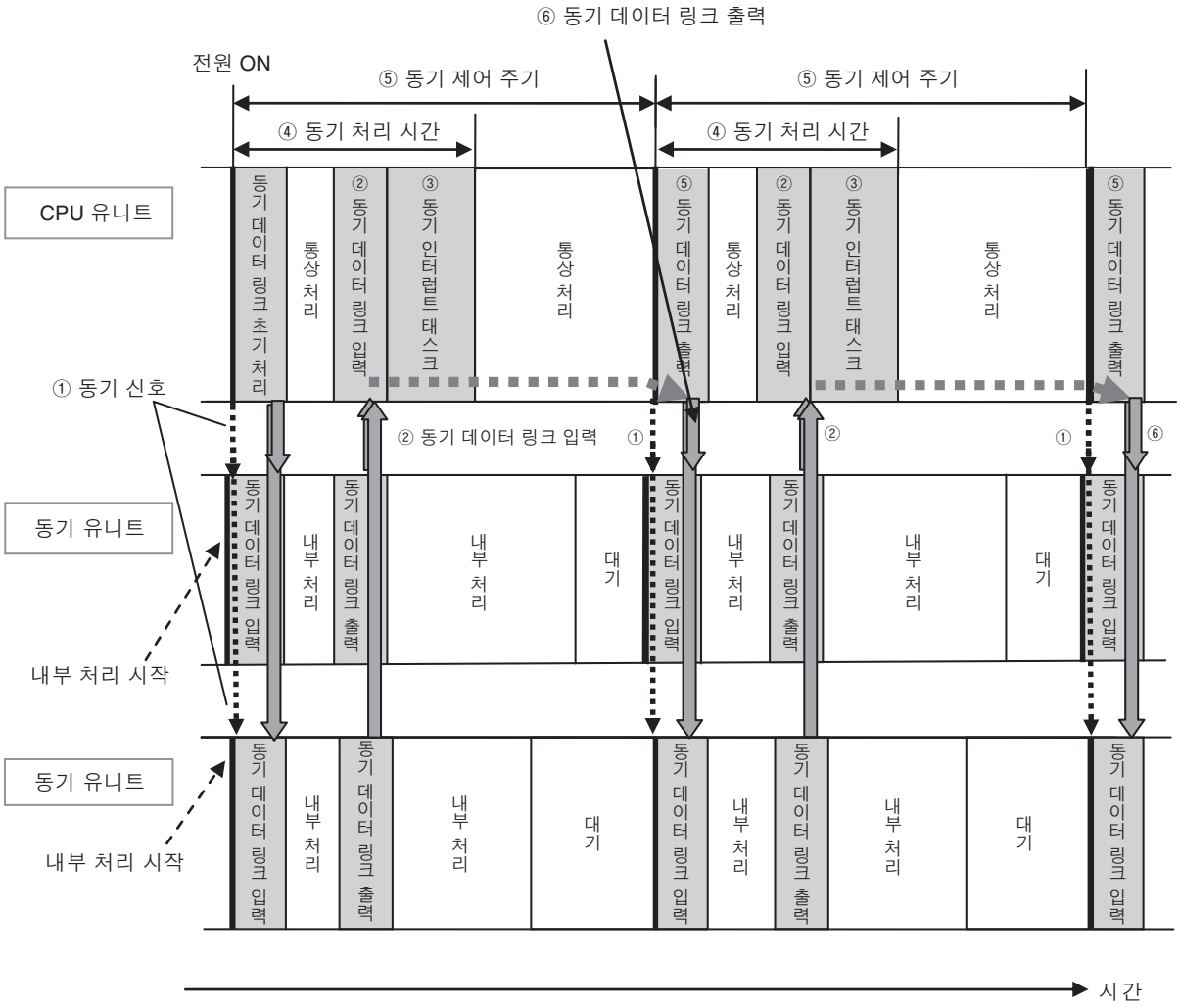
참고

I/O 리프레시도 CPU 유니트와 각 유니트 간에 데이터를 교환하는 기능입니다. 유니트 간 동기 제어와의 차이점은 다음과 같습니다.

항목	유니트 간 동기 제어	I/O 리프레시
데이터 교환 타이밍	• 동기 제어 주기마다(CPU 유니트의 통상 사이클에 인터럽트 실행) • 동기 유니트의 내부 처리 타이밍과 동기	• CPU 유니트의 통상 사이클마다 (사용자 프로그램 실행 후) • 동기 유니트의 내부 처리 타이밍과 관계 없이
데이터 교환 주기	일정(설정된 동기 제어 주기)	일정치 않거나 사이클 시간 일정화 기능으로 일정화(단, 실제 사이클 시간 이상일 것)
동기 유니트 간의 내부 처리 시작 타이밍	동기	비동기

10-8-2 유니트 간 동기 제어의 자세한 구조

시간적인 순서로 본 경우, 유니트 간 동기 제어의 동작은 다음과 같습니다.



1 동기 신호

전원 ON 시, CPU 유니트가 동기 신호를 발신하고 그 후 동기 데이터 링크를 시작하기 위한 초기 처리를 시작합니다.

동시에 CPU 유니트와 각 동기 유니트는 자신의 처리를 시작합니다.

2 동기 데이터 링크(동기 유니트→CPU 유니트)

동기 유니트에서 CPU 유니트로, 동기 데이터 링크를 통해 데이터를 입력합니다.

3 동기 인터럽트 태스크

CPU 유니트에서 동기 인터럽트 태스크를 실행합니다.

주. 유니트 간 동기 제어 기능은 동기 인터럽트 태스크 없이도 사용할 수 있습니다.

4 동기 처리 시간

동기 제어 주기 1사이클 내에서 동기 제어 처리에 걸리는 일련의 시간(동기 신호 발신부터 동기 인터럽트 태스크가 종료될 때까지)입니다.

이 동기 처리 시간 경과 후, 설정한 동기 제어 주기가 지날 때까지 통상 처리를 수행합니다.

동기 처리 시간은 동기 제어 주기 이하이어야 합니다. (동기 처리 시간의 경우 CX-Programmer의 [동기 작업 상태] 대화 상자에서 최대값과 현재값을 모니터링할 수 있습니다.)

5 동기 제어 주기

동기 제어 주기가 지나면 동기 신호(①)를 다시 발신합니다.

6 동기 데이터 링크(CPU 유니트→동기 유니트)

CPU 유니트에서 동기 유니트로, 동기 데이터 링크를 통해 데이터를 출력합니다.

7 이후에는 ②부터 반복합니다.

사용상의 주의 사항

CPU 유니트는 동기 데이터 링크 입력, 동기 인터럽트 태스크 실행, 동기 데이터 링크 출력을 수행하는 동안, 통상 사이클 처리를 중단합니다. 따라서, 통상 사이클 내에서 중단 시간의 합계만큼 CPU 유니트의 사이클 시간이 늘어납니다.

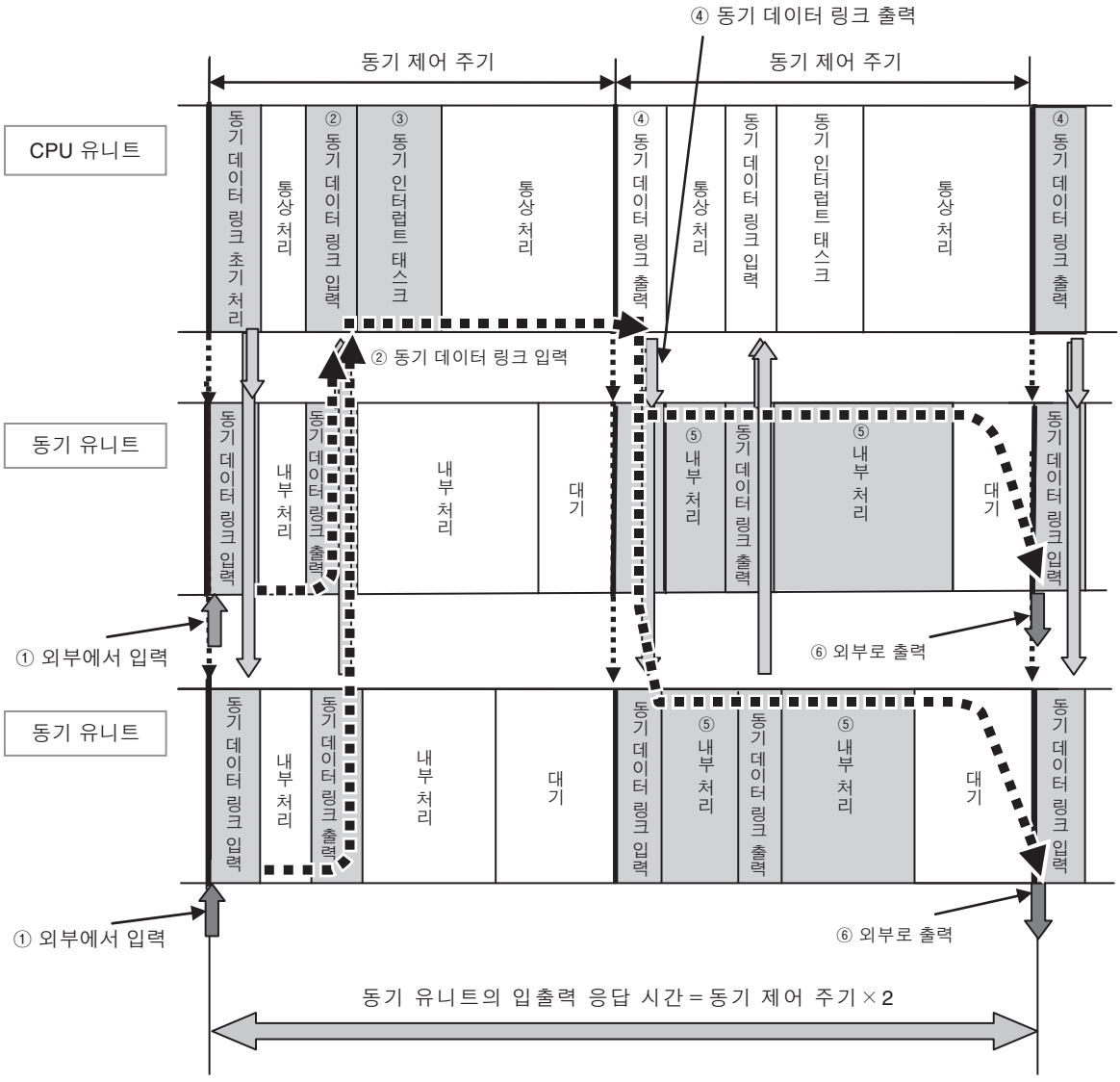
사이클 시간이 늘어나도 제어 시스템에 영향이 없는지 확인한 후 유니트 간 동기 제어 기능을 사용하십시오.

● 동기 유니트의 입출력 응답 시간

동기 유니트의 입출력 응답 시간은 다음 ①~⑥의 합계 시간이 됩니다.

① 외부에서 동기 유니트로의 입력
② 동기 데이터 링크(동기 유니트→CPU 유니트)
③ 동기 인터럽트 태스크의 실행
④ 동기 데이터 링크(CPU 유니트→동기 유니트)
⑤ 동기 유니트의 내부 처리
⑥ 동기 유니트에서 외부로의 출력

따라서 입출력 응답 시간의 최대 시간은 동기 제어 주기의 2사이클 길이가 됩니다.



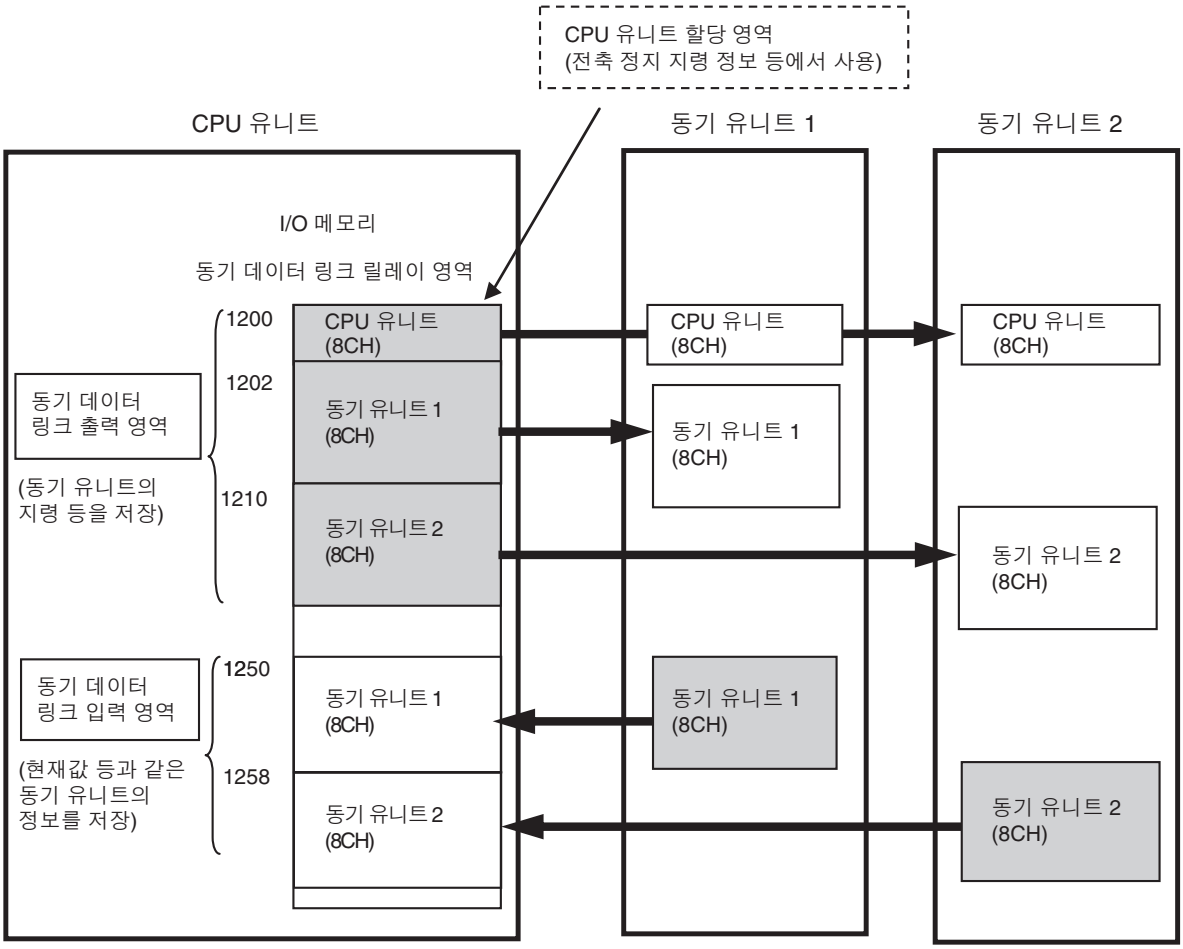
예) 위치 제어 유니트 CJ1W-NC□□4의 경우
내부 처리의 연산 시간이 1ms이므로 동기 제어 주기는 최소한 1ms로 설정해야 합니다.
그러므로 위치 제어 유니트의 입출력 응답 시간은 2ms로 고정됩니다.

10-8-3 유니트 간 동기 제어 기능의 사양

항목	내용
사용 가능한 CPU 유니트	CJ2H CPU 유니트 CJ2H-CPU6□(-EIP)의 유니트 Ver.1.1 이상
동기 유니트로 사용 가능한 고기능 유니트	위치 제어 유니트 CJ1W-NC□□4
동기 유니트의 장착 가능 대수	10대 주. 위치 제어 유니트 CJ1W-NC□□4의 경우, 최대 5대
동기 유니트의 장착 위치	CPU 장치상에 있어야 함(증설 장치 장치상에서는 불가)
동기 제어 주기	0.5~10ms(0.1ms 단위) PLC 시스템 설정을 통해 설정 단, 다음 조건을 동시에 만족해야 함 1) 동기 처리 시간 < 동기 제어 주기 2) 동기 유니트의 내부 처리 연산 시간 < 동기 제어 주기
동기 제어 주기의 변동 최대값	10μs
동기화하여 데이터 교환되는 CPU 유니트의 I/O 메모리 영역	동기 데이터 링크 영역(CIO 1200~1295CH) 출력(CPU→유니트), 입력(유니트→CPU)의 2개 영역으로 나뉩니다. 내용은 각 동기 유니트에 따라 달라집니다.
동기 인터럽트 태스크의 태스크 번호	인터럽트 태스크 번호 2 고정 주. 유니트 간 동기 제어 기능은 동기 인터럽트 태스크 없이도 사용할 수 있습니다.

10-8-4 동기 데이터 링크 릴레이

동기 데이터 링크 영역은 CPU 유니트에서 CIO 영역의 1200CH(고정)부터 시작하며 출력 영역(CPU 유니트→동기 유니트)⇒입력 영역(동기 유니트→CPU 유니트) 순서가 됩니다.



다음 내용을 PLC 시스템 설정의 [타이밍/동기] 탭의 '동기 유니트 작동 설정'에서 설정합니다.

- 전체 출력 영역의 크기, 전체 입력 영역의 크기
- 출력 영역 내 1유니트의 할당 크기, 입력 영역 내 1유니트의 할당 크기
- 입력 영역의 시작 주소

동기 데이터 링크 릴레이의 주소는 1200~1295CH입니다.
선두는 1200CH로 고정되며 그 밖에는 임의로 설정할 수 있습니다.

입력/출력	항목	내용
출력	전송 방향	CPU 유니트→동기 유니트
	시작 주소	1200CH 고정
	할당	처음에는 CPU 유니트부터 시작하며 그 후에는 동기 유니트의 등록 순서에 따라 PLC 시스템 설정에서 설정된 크기만큼 할당됩니다. CPU 유니트는 2CH 이상, 동기 유니트는 1대당 0CH 이상 할당할 수 있습니다.
입력	전송 방향	동기 유니트→CPU 유니트
	시작 주소	PLC 시스템 설정을 통해 설정한 값입니다. 1202~1294 범위로 설정 가능합니다.
	할당	동기 유니트 등록 순서에 따라 PLC 시스템 설정에서 설정된 크기만큼 할당됩니다. 동기 유니트 1대당 0~16CH를 할당할 수 있습니다.

- **할당 예**

동기 데이터 링크의 입출력 영역 할당 예 (유니트 번호 0과 1의 동기 유니트를 사용하는 경우)

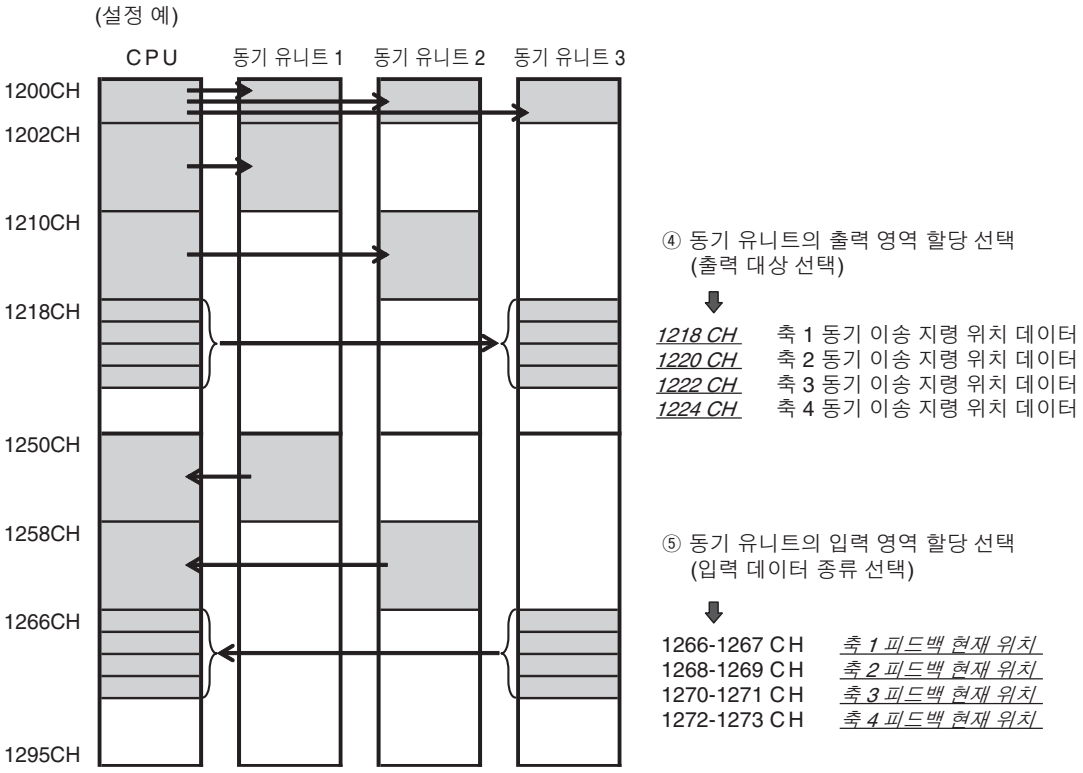
동기 데이터 링크를 수행하는 입출력 영역의 시작 위치, 영역의 크기를 설정합니다.

채널 번호	항목	입력/출력의 방향
1200CH	CPU 유니트 할당 영역 (영역 크기 2CH로 설정한 경우)	출력(CPU 유니트→동기 유니트)
1201CH		
1202CH	동기 유니트 번호 0 할당 영역 (영역 크기 2CH로 설정한 경우)	
1203CH		
1204CH	동기 유니트 번호 1 할당 영역 (영역 크기 4CH로 설정한 경우)	
1205CH		
1206CH		
1207CH		
:		
1250CH	동기 유니트 번호 0 할당 영역 (영역 크기 4CH로 설정한 경우)	입력(동기 유니트→CPU 유니트)
1251CH		
1252CH		
1253CH		
1254CH	동기 유니트 번호 1 할당 영역 (영역 크기 2CH로 설정한 경우)	
1255CH		
:		
1295CH		

동기 데이터 링크의 입출력 데이터 종류 설정 예 (위치 제어 유니트를 사용하는 경우)

할당한 입출력 영역을 어떻게 사용할지 설정합니다.

- 출력 영역 할당 선택
  위치 제어 유니트의 경우 각 축의 동기 이송 지령 위치 데이터로 어느 채널의 데이터를 사용할지 선택합니다.
- 입력 영역 할당 선택
  위치 제어 유니트의 경우, 각 축의 지령 현재 위치, 피드백 현재 위치를 선택합니다.

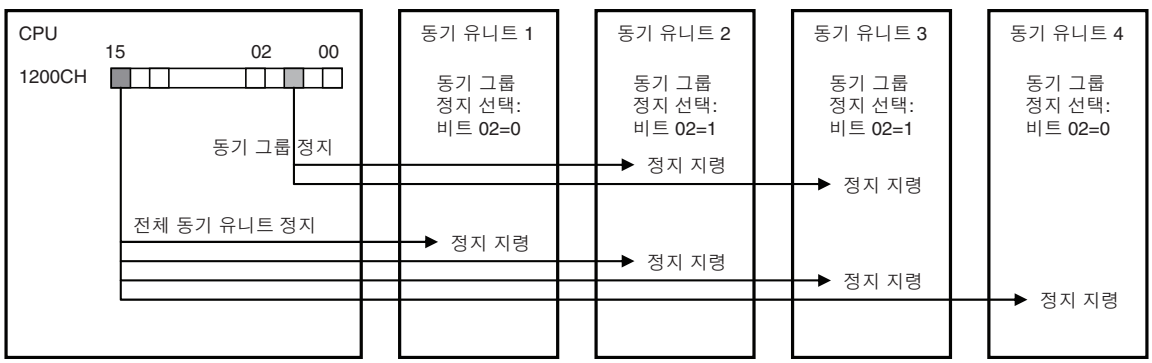


● 용도 예

동기 유니트로 위치 제어 유니트, 고속 카운터 유니트를 사용하는 경우, 다음과 같은 용도로 사용됩니다.

영역	용도 예
CPU 유니트 할당 영역	그룹 축 정지 기능, 전체 동기 대상 유니트 축 정지 기능(*1) 등의 설정
동기 데이터 링크 출력 영역	목표 위치 데이터(X, Y, Z, U 축)
동기 데이터 링크 입력 영역	피드백 현재 위치(X, Y, Z, U 축)

*1 그룹 축 정지 기능, 전체 동기 대상 유니트 축 정지 기능
유니트 간 동기 제어를 실행하고 있는 동기 유니트에 대해 실행 중인 기능을 정지하도록 지령을 내릴 수 있습니다.
모든 동기 유니트에 대해 정지 지령을 내리는 전체 동기 유니트 정지 기능과 미리 설정한 조합의 유니트에 대해 정지 지령을 내리는 동기 그룹 정지 기능이 있습니다.



- **전체 동기 유니트 정지 기능**
  동기 데이터 링크 영역에 있는 1200CH의 비트 15는 전체 동기 유니트 정지 릴레이로 사용되고 있습니다. 이 릴레이를 ON으로 하면 전체 동기 유니트에 대해 실행 중인 동작을 정지하는 지령을 일제히 내릴 수 있습니다.
- **동기 그룹 정지 기능**
  동기 데이터 링크 영역에 있는 1200CH의 비트 00~14는 동기 그룹 정지 릴레이로 사용되고 있습니다. 이 릴레이를 ON으로 하면 동기 유니트 측의 동기 그룹 정지 선택 파라미터에서 해당하는 비트가 선택된 동기 유니트에 대해 실행 중인 동작을 정지하는 지령을 내릴 수 있습니다.

동기 그룹 정지 선택 파라미터(위치 제어 유니트 CJ1W-NC□□4의 축 파라미터)

명칭	비트	기능	설정 범위	초기값
동기 그룹 정지 선택	00	동기 데이터 링크 영역에 있는 1200CH의 각 비트의 ON에 대한 동작을 선택합니다. 0: 비트 00의 ON을 무시 1: 비트 00이 ON이 된 경우 정지 실행	0.1	0
	01	0: 비트 01의 ON을 무시 1: 비트 01이 ON이 된 경우 정지 실행	0.1	0
	02	0: 비트 02의 ON을 무시 1: 비트 02가 ON이 된 경우 정지 실행	0.1	0
	:	:		
	14	0: 비트 14의 ON을 무시 1: 비트 14가 ON이 된 경우 정지 실행	0.1	0
	15	—	—	—

10-8-5 유니트 간 동기 제어 기능의 사용상 제한 사항

● 동기 제어 주기의 설정 시간 제약

동기 제어 주기는 다음 두 조건에서 설정하십시오.

(1) 동기 처리 시간 < 동기 제어 주기
※ 일반적으로 동기 제어 주기는 동기 처리 시간의 1.5배 이상으로 설정하십시오.

(2) 동기 유니트의 내부 처리 연산 시간 < 동기 제어 주기

동기 처리 시간이란?

다음 ①과 ②의 합계 시간입니다.

① 동기 신호를 발신한 후 동기 인터럽트 태스크가 종료될 때까지의 시간
(동기 데이터 링크 출력 시간+통상 처리 시간+동기 데이터 링크 입력 시간+동기 인터럽트 태스크 실행 시간의 합계)

② 위의 ① 동안의 통상 처리 중에 조건이 성립한 I/O 인터럽트 태스크 또는 외부 인터럽트 태스크의 실행 시간
동기 처리 시간의 최대값과 현재값은 CX-Programmer의 [동기 작업 상태] 대화 상자에서 모니터링할 수 있습니다.

동기 유니트의 내부 처리 연산 시간

위치 제어 유니트 CJ1W-NC□□4를 예로 들면, 내부 처리의 연산 시간은 1ms이므로 동기 제어 주기는 1ms 이상이어야 합니다.

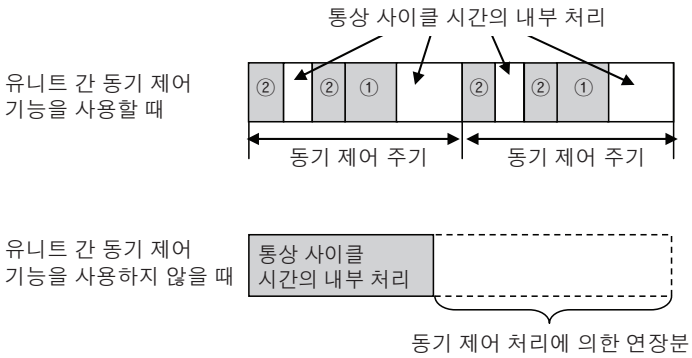
동기 제어 주기가 위의 두 조건에 해당하지 않는 경우

- **동기 제어 주기가 동기 처리 시간보다 짧은 경우**
  동기 인터럽트 태스크 또는 다른 인터럽트 태스크(I/O 인터럽트, 외부 인터럽트) 실행 중에 다음 동기 사이클로 들어가서 에러(CPU 유니트 측에서 동기 처리 시간 초과)가 발생합니다.
  동기 인터럽트 태스크와 동기 데이터 링크 리프레시는 해당 동기 사이클 내에서 한번에 실행되지 않습니다.
- **동기 제어 주기가 동기 유니트의 내부 처리 연산 시간보다 짧은 경우**
  동기 유니트의 내부 처리가 완료되지 않은 상태에서 다음 동기 사이클로 들어가서 에러(동기 유니트 측에서 동기 처리 시간 초과)가 발생합니다.
  동기 유니트 측의 처리는 계속합니다. (위치 제어 유니트의 경우, 축 동작을 계속합니다.)

● 통상 사이클 시간 연장

유니트 간 동기 제어 기능을 사용하면 일반적으로 사이클 시간이 다음과 같이 늘어납니다.
① 동기 인터럽트 태스크의 실행 시간(일반적인 예약 인터럽트 태스크와 동일)
② 동기 데이터 링크의 입력 리프레시 시간+출력 리프레시 시간
주. 위의 연장 시간은 동기 처리 시간에서 통상 처리 시간을 뺀 시간입니다.

예) 평상시의 1사이클에 대해 동기 제어 처리가 동기 제어 주기 2회에 해당하는 인터럽트를 수행한 경우



● 예약 인터럽트 태스크 0 및 1 사용 불가

유니트 간 동기 제어 기능을 사용하면 예약 인터럽트 0, 1을 인터럽트 요인으로 사용할 수 없게 됩니다. 따라서 예약 인터럽트 태스크 0(인터럽트 태스크 번호 2), 예약 인터럽트 태스크 1(인터럽트 태스크 번호 3)은 사용할 수 없습니다. 또한 인터럽트 태스크 번호 2는 동기 인터럽트 태스크로 사용합니다.

● 인터럽트 제어 명령의 사용 제한

동기 인터럽트 태스크에 대해 MSKS 명령, MSKR 명령, CLI 명령은 사용할 수 없습니다.
또한 DI 명령, EI 명령은 사용 가능합니다.

● 타이머 계통 명령의 사용 제한

- TIM/TIMX 명령(100ms 타이머), TIMH/TIMHX 명령(10ms 타이머), TMHH/TMHHX 명령(1ms 타이머)에는 다음과 같은 사용 제한이 있습니다.
- 통상 사이클 시간이 100ms를 초과하는 경우 타이머가 정상적으로 동작하지 않습니다.
- 위의 타이머 명령이 정지한 태스크 내에 있거나 JMP/CJMP/CJPN-JME 명령으로 점프되어서 실행되지 않은 경우, 타이머 정도에 다음과 같은 오차가 발생합니다.
  TIM/TIMX 명령(100ms 타이머): 최대 −10ms의 오차
  TIMH/TIMHX 명령(10ms 타이머): 최대 −10ms의 오차
  TMHH/TMHHX 명령(1ms 타이머): 최대 −1ms의 오차
- 현재값은 참조할 수 없는데, 참조하는 경우에는 잘못된 값을 읽어옵니다.

● 인터럽트 태스크 내에서 일반 명령어 사용 불가('고속 인터럽트 기능 사용' 유효 시의 제한)

유니트 간 동기 제어 기능을 사용하면 무조건 PLC 시스템 설정의 '고속 인터럽트 기능 사용'이 유효화됩니다. 따라서 동기 인터럽트 태스크를 포함한 모든 인터럽트 태스크 내에서 네트워크 통신 명령 등 일부 명령어를 사용할 수 없습니다. 자세한 내용은 '10-2-6 고속 인터럽트 기능'을 참조하십시오.

● 기타 제한('고속 인터럽트 기능 사용' 유효 시의 제한)

- 인터럽트 태스크 내에서 명령어의 백그라운드 처리를 사용할 수 없습니다.
- 특수 보조 릴레이의 인터럽트 태스크 최대값 처리 시간(A440CH), 최대 처리 시간 인터럽트 태스크 번호(A441CH)가 무효화됩니다.
- EtherNet/IP 태그 데이터 링크 사용 시 사이클 시간에 영향을 주는 시간이 길어집니다.

● 동기 유니트의 장착 위치

동기 유니트는 CPU 장치에 장착하십시오.
증설 장치에서는 사용할 수 없습니다.

10-8-6 사용 순서

유니트 간 동기 제어를 사용하기 위한 사용 순서는 다음과 같습니다.

1 **초기 설정 ('10-8-7 PLC 시스템 설정' 참조)**

CX-Programmer의 PLC 시스템 설정에서 [타이밍/동기] 탭의 '동기 유니트 작동 설정'을 통해 설정합니다.

(1) 유니트 간 동기 제어를 유효화합니다.
'동기 작동 사용'을 선택합니다.

(2) '동기 사이클 시간'을 설정합니다.

(3) 동기 데이터 링크에서 데이터를 교환할 할당 영역을 설정합니다.

(a) '동기 데이터 갱신 영역 할당'에서 입력 영역의 선두를 설정합니다.

(b) 동기 유니트를 등록합니다.

(c) 출력 영역 및 입력 영역의 크기와 시작 주소를 유니트마다 설정합니다.

2 **동기 유니트마다 입출력 영역 할당**

동기 유니트마다 할당한 입출력 영역에 대해 사용 방법을 설정합니다.

CX-Programmer의 PLC 시스템 설정에서 [타이밍/동기] 탭의 '동기 유니트 작동 설정'에 있는 [특별 유니트 설정] 버튼을 클릭하여 동기 유니트마다 설정합니다.

각 동기 유니트의 할당 내용에 대해서는 각 동기 유니트의 설명서를 참조하십시오.

3 **동기 인터럽트 태스크의 작성(필요한 경우만) ('10-8-8 동기 인터럽트 태스크의 작성' 참조)**

CX-Programmer에서 동기 인터럽트 태스크로 실행하고 싶은 프로그램을 인터럽트 태스크 번호 2에 작성합니다.

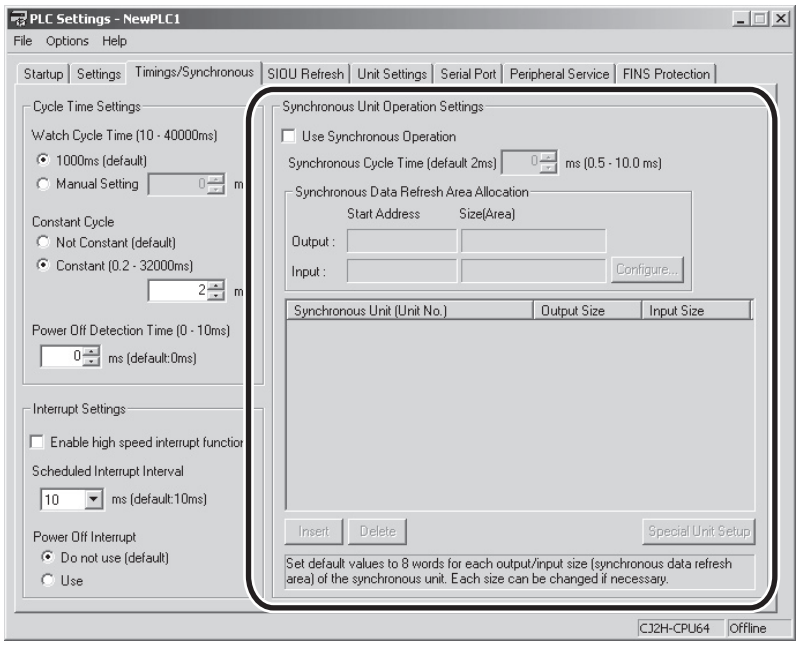
4 **유니트 간 동기 제어 동작의 디버그**
('10-8-9 유니트 간 동기 제어 동작의 조정/문제 해결' 참조)

CX-Programmer의 [동기 작업 상태]에서 다음 내용을 확인하면서 동기 제어 주기를 조정합니다.

- 동기 처리 시간의 현재값, 최대값
- 동기 제어 주기의 설정 시간
- 통상 사이클 시간의 현재값, 최대값
- 각 동기 유니트와의 데이터 교환 발생 에러

10-8-7 PLC 시스템 설정

유니트 간 동기 제어 기능의 초기 설정은 모두 CX-Programmer에서 PLC 시스템 설정의 [타이밍/동기] 탭에 있는 '동기 유니트 작동 설정'에서 설정합니다.



● '동기 작동 사용'

유니트 간 동기 제어 기능을 사용할 때의 설정입니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
동기 작동 사용	OFF: 사용 안 함 ON: 사용	OFF (사용 안 함)	유니트 간 동기 제어를 실행합니다. 주. 이 설정을 ON으로 하면 인터럽트 설정의 '고속 인터럽트 기능 사용'도 자동으로 ON이 됩니다.	–

● '동기 사이클 시간'

동기 제어 주기의 설정입니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
동기 제어 주기	0.5~10.0ms (0.1ms 단위)	2.0ms	동기 제어 주기를 설정합니다.	A10102.00~A10102.15 (동기 제어 주기 설정 시간)

사용상의 주의 사항

동기 제어 주기는 다음과 같은 조건을 만족해야 합니다.

- 동기 처리 시간 < 동기 제어 주기
- 동기 유니트의 내부 처리 연산 시간 < 동기 제어 주기

일반적으로 동기 제어 주기는 동기 처리 시간의 1.5배 이상으로 설정하십시오.

● '동기 데이터 갱신 영역 할당'

'동기 데이터 갱신 영역 할당 설정'

동기 데이터 링크 릴레이의 영역 할당 설정입니다.

설정 항목		설정 데이터	초기값	설명	관련 특수 보조 릴레이
출력 영역	시작 주소	1200 고정	CIO1200	전체 동기 데이터 링크 릴레이 (출력)의 시작 주소입니다.	–
	크기 (할당 영역)	2~96채널 (*)	50채널	전체 동기 데이터 링크 릴레이 (출력)의 크기입니다.	–
입력 영역	시작 주소	1202~1294 또는 [사용 안 함]	CIO1250	전체 동기 데이터 링크 릴레이 (입력)의 시작 주소입니다.	–
	크기 (할당 영역)	2~94채널 (*) 또는 설정 안 함	46채널	전체 동기 데이터 링크 릴레이 (입력)의 크기입니다.	–

* 입력 영역의 시작 주소 설정에 의해 자동 계산됩니다.

'크기 할당 영역'

출력 영역:

동기 데이터 링크 릴레이의 출력 영역 크기에 대한 설정입니다.

시작 주소는 자동 계산됩니다.

각 동기 유니트에 대해 개별적으로 설정합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
출력 크기	CPU 유니트: 2채널 ~ 출력 가능한 크기 동기 유니트: 0채널 ~ 출력 가능한 크기	CPU 유니트: 2채널 동기 유니트: 8채널	동기 데이터 링크 릴레이(출력)의 크기를 설정합니다.	–

입력 영역:

동기 데이터 링크 릴레이의 입력 영역 크기에 대한 설정입니다.

시작 주소는 자동 계산됩니다.

각 동기 유니트에 대해 개별적으로 설정합니다.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
입력 크기	동기 유니트: 0 ~ 16채널	동기 유니트: 8채널	동기 데이터 링크 릴레이(입력)의 크기를 설정합니다.	–

● '동기 유니트(유니트 번호)'

연결되어 있는 동기 유니트 중에 어느 것을 유니트 간 동기 제어에 사용할지 설정합니다.

동기 유니트는 10대까지 설정할 수 있지만 경우에 따라 10대 미만으로만 설정할 수 있는 유니트가 있습니다. 자세한 내용은 각 유니트의 설명서를 참조하십시오.

설정 항목	설정 데이터	초기값	설명	관련 특수 보조 릴레이
동기 유니트	CPU 유니트 및 유니트 간 동기 제어가 가능한 유니트명	CPU 유니트만 해당	빈 행을 클릭하면 유니트 간 동기 제어가 가능한 유니트명 목록이 표시되므로 이 목록에서 사용할 유니트명을 선택합니다.	–

10-8-8 동기 인터럽트 태스크의 작성

CX-Programmer에서 동기 인터럽트 태스크로 실행하고 싶은 프로그램을 작성하고 프로그램의 속성에서 태스크 유형: '인터럽트 태스크 02(예약 인터럽트 0)'를 지정합니다.
동기 인터럽트 태스크의 프로그램에는 전자 캠 제어 등을 기술합니다.
예) 전자 캠 제어를 하는 경우 실제 축 또는 가상 축에 따라 변화하는 캠 곡선을 제어하는 APR(숫자 변환) 명령 또는 표준 펑션 블록을 기술할 수 있습니다.

● 관련 특수 보조 릴레이

명칭	주소	설명	읽기/쓰기
유니트 간 동기 제어 기능 서비스 중 플래그	A10100.00	CPU 유니트가 동기 유니트에 대해 유니트 간 동기 제어 기능을 지원할 수 있는 상태일 때, 동기 사이클 2회째 이후에 1(ON)이 됩니다. 그 이외에는 0(OFF) (전원 ON 시, 유니트 재시작 후의 동기 사이클 1회째도 OFF)입니다. 사용자 프로그램에서 동기 제어용 프로그램의 실행 조건으로 사용합니다.	읽기
동기 데이터 전송 에러 (UNIT)	A10101	CPU 유니트가 동기 유니트에서 보낸 동기 데이터의 수신에 실패한 경우 0001Hex가 되고 수신에 성공한 경우에는 0000Hex가 됩니다. 동기 사이클마다 반영됩니다. 사용자 프로그램에서 동기 데이터 입력의 사용 가능 조건으로 사용합니다.	읽기
동기 제어 주기 설정 시간	A10102.00~ A10102.15	PLC 시스템 설정의 동기 제어 주기 설정 시간(0.1ms 단위)이 유효화된 시점에서 이 영역에 반영됩니다. 펑션 블록 등에서 동기 제어 주기 설정을 읽어오고 싶을 때 사용합니다.	읽기

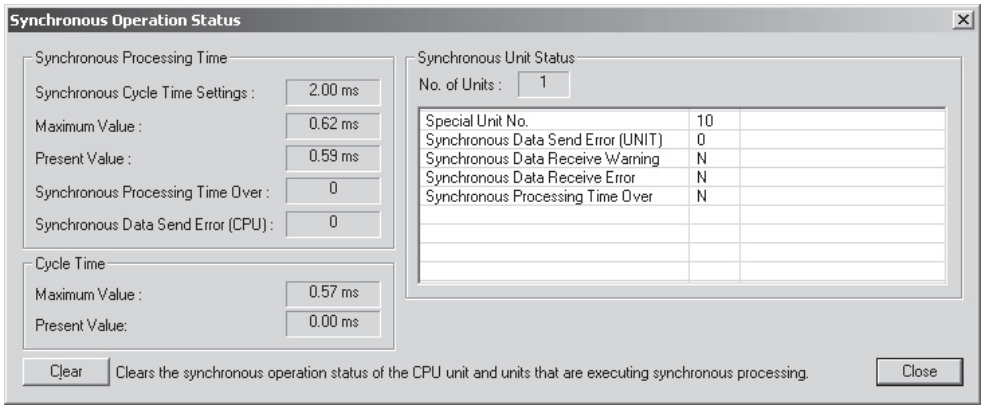
10-8-9 유니트 간 동기 제어 동작의 조정/문제 해결

동기 제어 상태의 확인

유니트 간 동기 제어 기능이 정상적으로 동작하고 있는지 여부는 CX-Programmer의 [동기 작업 상태] 대화 상자 또는 동기 유니트 전면의 SYNC LED로 확인할 수 있습니다.

- **CX-Programmer에서 확인**

  CX-Programmer를 유니트 간 동기 제어 기능을 실행 중인 PLC에 온라인으로 연결하고, CX-Programmer의 [PLC] 메뉴에서 [편집] | [동기 작업 상태]를 선택합니다.
  다음과 같이 [동기 작업 상태] 대화 상자가 표시됩니다.

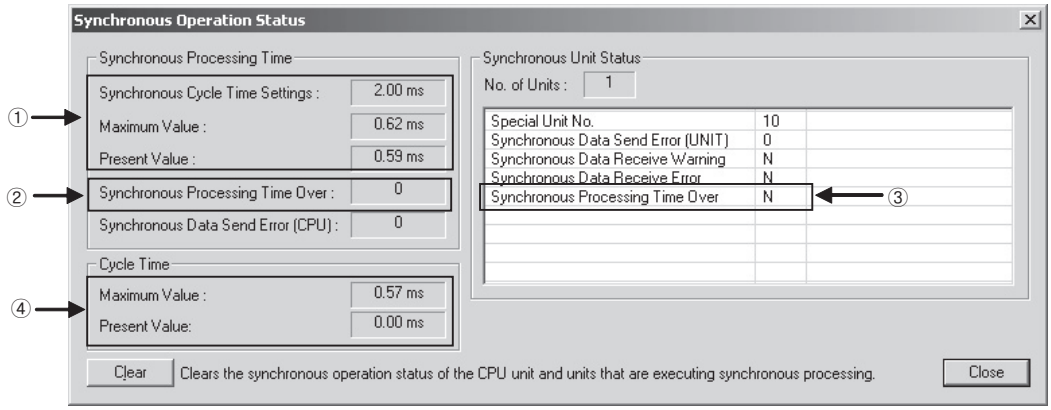


표시 항목		설명
동기 처리 시간	동기 사이클 시간 설정값	동기 제어 주기의 설정값
	최대값, 현재값	지금까지 동기 처리 시간의 최대값과 현재값
	동기 처리 시간 초과	CPU 유니트의 동기 처리 시간이 동기 제어 주기보다 커진 횟수. 동기 제어 주기 이내에 종료해야 하는 동기 처리가 종료되지 않은 횟수를 말합니다. 전원 ON 시 0으로 클리어됩니다.
	동기 데이터 전송 에러(CPU)	CPU 유니트에서 동기 유니트로 동기 데이터 송신이 실패한 횟수. 전원 ON 시 0으로 클리어됩니다.
사이클 시간	최대값, 현재값	통상 사이클 시간의 최대값과 현재값
동기 유니트의 상태	유니트 수	동기 유니트로 설정된 유니트의 수
	고기능 유니트 번호	각 동기 유니트의 유니트 번호
	동기 데이터 전송 에러(UNIT)	CPU 유니트가 동기 유니트에서 보낸 동기 데이터의 수신에 실패한 횟수. 전원 ON 시 0으로 클리어됩니다.
	동기 데이터 링크 수신 경고	CPU 유니트에서 송신되는 동기 데이터를 동기 유니트가 수신하지 못한 경우가 한번이라도 있으면 '있음'이 되는 경고입니다. 동기 데이터 전송 에러(CPU)가 발생하면 '있음'이 됩니다. 고기능 유니트용 할당 릴레이 영역(유니트 경고 리셋)이 ON으로 되면 클리어됩니다.
	동기 데이터 링크 수신 에러	CPU 유니트에서 송신되는 동기 데이터를 동기 유니트가 2회 연속으로 수신하지 못하면 '있음'이 되는 경고입니다. 고기능 유니트용 할당 릴레이 영역(유니트 에러 리셋)이 ON으로 되면 클리어됩니다.
	동기 처리 시간 초과	동기 유니트의 내부 처리 연산 시간이 동기 제어 주기보다 크지 않은 경우가 한번 이상 있으면 '있음'이 되는 경고입니다. 고기능 유니트용 할당 릴레이 영역(유니트 경고 리셋)이 ON으로 되면 클리어됩니다.
[지우기] 버튼		CPU 측과 동기 유니트 측의 동기 제어 상태나 에러 및 경고의 값을 클리어합니다.

동기 제어 주기 조정 방법

동기 제어 주기의 설정이 적절한지 여부를 다음 방법으로 확인하여 조정하십시오.

① 동기 제어 주기의 최대값이나 현재값이 동기 제어 주기 설정값을 초과하지 않았는지 여부를 확인합니다.
② CPU 유니트 측의 동기 처리 시간 초과가 빈번하게 발생하지 않는지 여부를 확인합니다.
③ 동기 유니트 측의 동기 처리 시간 초과가 '있음'이 되지 않았는지 여부를 확인합니다.
④ 유니트 간 동기 제어에 의해 통상 사이클의 연장이 허용 범위인지 여부를 사이클 시간의 최대값, 현재값으로 확인합니다.



● 동기 유니트 전면의 SYNC LED에서 확인

각 동기 유니트가 유니트 간 동기 제어 모드이면 각 동기 유니트 전면의 SYNC LED가 녹색으로 점등됩니다.
그리고 동기 데이터 링크 수신 에러 등이 발생하면 소등됩니다.

참고

동기 유니트와의 고기능 I/O 유니트용 할당 릴레이 영역 또는 CPU 고기능 유니트용 할당 릴레이 영역에는 다음과 같은 정보가 할당됩니다.
CPU 유니트의 사용자 프로그램에서 다음 비트를 사용하여 동기 유니트의 상태를 확인할 수 있습니다.

	크기	명칭	기능
고기능 유니트용 할당 릴레이 영역 (동기 유니트→CPU 유니트)	1비트	유니트 간 동기 제어 모드 중 플래그	동기 유니트가 유니트 간 동기 제어 모드에서 동작 중이면 1(ON)이 됩니다. (고기능 유니트의 SYNC LED의 점등/소등 타이밍과 공통입니다.)
	1비트	동기 데이터 링크 수신 경고	CPU 유니트에서 송신되는 동기 데이터를 동기 유니트가 수신하지 못한 경우가 한번이라도 있으면 1(ON)이 됩니다.
	1비트	동기 데이터 링크 수신 에러	CPU 유니트에서 송신되는 동기 데이터를 동기 유니트가 2회 연속으로 수신하지 못한 경우가 있으면 1(ON)이 됩니다.
	1비트	동기 처리 시간 초과	동기 유니트의 내부 처리 연산 시간이 동기 제어 주기보다 크지 않은 경우가 한번 이상 있으면 1(ON)이 됩니다.
고기능 유니트용 할당 릴레이 영역 (CPU 유니트→동기 유니트)	1비트	유니트 에러 리셋	기동하면 동기 유니트의 에러 리셋을 시작합니다. 동기 데이터 링크 수신 에러 등을 리셋합니다.
	1비트	유니트 경고 리셋	기동하면 동기 유니트의 경고 리셋을 시작합니다. 동기 데이터 링크 수신 경고, 동기 처리 시간 초과(경고) 등을 리셋합니다.

유니트 간 동기 제어 기능의 정지

● 동기 신호 감시 이상

CPU 유니트에서 동기 유니트에 대한 동기 신호가 100ms를 초과해도 발신되지 않을 때(CPU 유니트 기동 시는 11s) 모든 동기 유니트와의 유니트 간 동기 제어 기능이 정지합니다.

● I/O 버스 또는 동기 유니트의 이상

동기 유니트 중 1대라도 I/O 버스 이상 또는 다음과 같은 유니트 이상이 있는 경우 모든 동기 유니트와의 유니트 간 동기 제어 기능이 정지합니다.

- I/O 설정 이상, 유니트 번호 이중 사용 이상, 고기능 I/O 설정 이상
- CPU 고기능 설정 이상, 고기능 I/O 이상, CPU 고기능 이상

이때, CX-Programmer에 이상을 표시하는 대화 상자가 나타납니다.

참고

유니트 간 동기 제어 기능에서는 동기 유니트 중 1대라도 유니트 재시작이 실행된 경우, 동기 사이클을 정지하고 다른 모든 동기 유니트에 대해서도 유니트 재시작을 실행합니다.
모든 동기 유니트의 재시작이 정상적으로 종료된 단계에서 유니트 간 동기 제어를 재개합니다.

조정/문제 해결 방법

다음과 같은 경우 동기 제어 주기 등을 조정해야 합니다.

CX-Programmer의 [동기 작업 상태]	현상	요인	대처 방법
'동기 처리 시간' 필드	'동기 처리 시간 초과'가 빈번하게 발생할 때	CPU 유니트의 동기 처리(동기 리프레시+동기 인터럽트 태스크 실행)가 동기 제어 주기 이내에 종료되지 않았습니다.	동기 제어 주기를 동기 처리 시간(동기 리프레시 시간+동기 인터럽트 태스크 실행 시간의 합계)보다 길게 설정하십시오.
	'동기 데이터 송신 에러(CPU)'가 1회 이상일 때	CPU 유니트의 하드웨어 이상입니다.	CPU 유니트를 교환하십시오.
'동기 유니트 상태' 필드	'동기 데이터 전송 에러(UNIT)'가 빈번하게 발생할 때	동기 유니트에서 CPU 유니트로 입력되는 동기 데이터가 동기 데이터 송신까지 준비되지 않았습니다.	동기 제어 주기를 동기 유니트의 내부 처리 연산 시간보다 길게 설정하십시오.
	'동기 처리 시간 초과'가 '있음'일 때	동기 유니트의 내부 처리를 동기 제어 주기 이내에 종료하지 못했습니다.	동기 제어 주기를 동기 유니트의 내부 처리 연산 시간보다 길게 설정하십시오.

11

주변 툴/기기와 연결

이 장에서는 CJ2 CPU 유니트와 CX-Programmer 등의 주변 툴 또는 기타 기기와 연결하는 방법을 설명합니다.

11-1 CX-Programmer에서 PLC로 연결하는 방법 11-2
11-1-1 개요 11-2
11-1-2 CX-Programmer에서 액세스 가능한 시스템 구성 11-4
11-1-3 CX-Programmer에서 PLC에 대한 액세스 조작 11-8
11-1-4 자동 온라인 연결 11-11
11-2 시리얼 통신 11-15
11-2-1 시리얼 통신 11-15
11-3 통신 네트워크 11-28

11-1 CX-Programmer에서 PLC로 연결하는 방법

11-1-1 개요

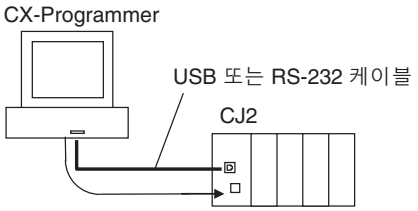
CX-Programmer에서 PLC에 액세스하는 경우에 대한 개요는 다음과 같습니다.

PLC에 연결하는 방법

CX-Programmer에서 PLC에 액세스하는 방법에는 다음의 2종류가 있습니다.

- **PLC에 직접 연결하여 액세스**

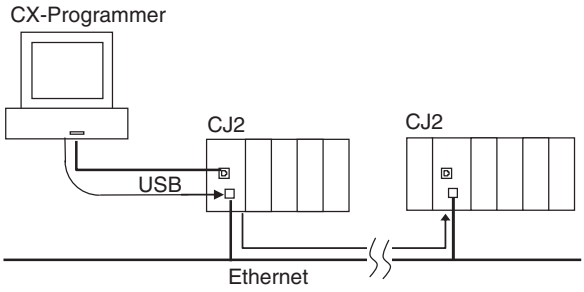
  시리얼 통신으로 직접 연결된 PLC에 연결합니다.



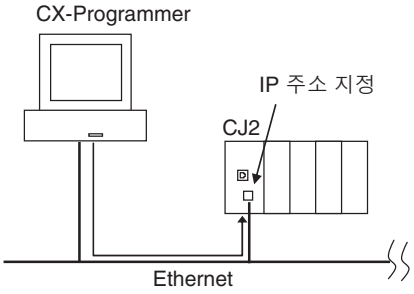
- **네트워크상의 PLC에 액세스**

  네트워크상의 PLC에 액세스하는 방법은 다음의 2종류가 있습니다.

  시리얼 통신으로 직접 연결된 PLC를 통해 네트워크상의 PLC에 연결



  네트워크에 연결된 PC에서 네트워크상의 PLC에 연결



PLC에 대한 연결 조작

CX-Programmer에서 PLC에 연결하는 조작에는 다음의 2종류가 있습니다.

- **PLC 기종 변경 화면**

  CX-Programmer 프로젝트에서 대상 PLC에 대한 액세스 방법을 설정합니다.
  프로젝트를 작성할 때 설정합니다. 프로젝트를 새로 작성한 후에도 변경 가능합니다.

- **자동 온라인 연결**

  온라인 상태의 PLC에 액세스하는 조작입니다.
  CX-Programmer에서 프로젝트를 작성하지 않았을 때 사용할 수 있습니다.

  직접 연결

  직접 시리얼 연결된 PLC에 온라인 연결하는 조작입니다.

  EtherNet/IP 노드 연결

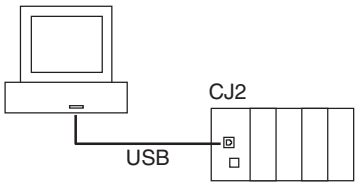
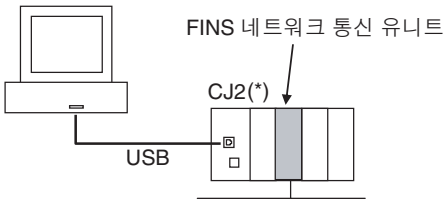
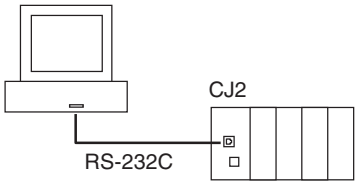
  EtherNet/IP상에서 온라인 상태의 PLC에 액세스하는 조작입니다.
  대상 PLC의 IP 주소를 모르는 경우에도 검색하여 액세스할 수 있습니다.

11-1-2 CX-Programmer에서 액세스 가능한 시스템 구성

PLC와 직접 시리얼 연결하는 경우

다음과 같이 네트워크 종류를 선택하십시오.

○: 연결 가능 ×: 연결 불가능

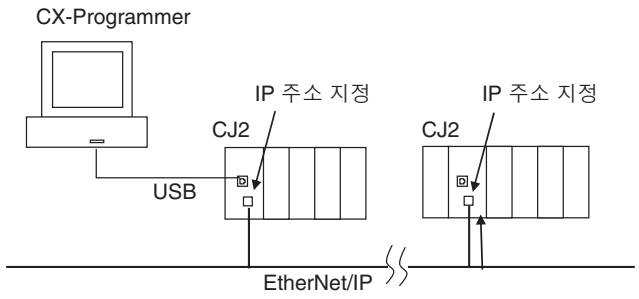
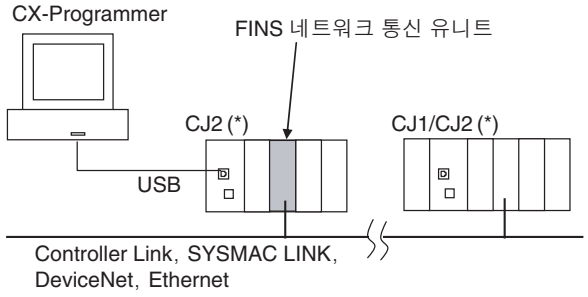
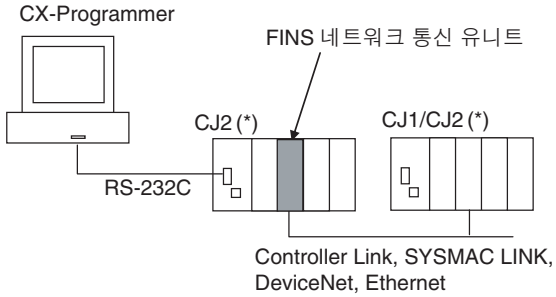
연결 케이블	시스템 구성	PLC 기종 변경 화면 (네트워크 종류)	자동 온라인 연결	
			직접 연결	EtherNet/IP 연결
USB 케이블	CX-Programmer  	USB	○	×
	FINS 네트워크에 액세스하는 경우 CX-Programmer  * 자체 네트워크 테이블을 설정해야 함	Toolbus USB 포트	○	×
RS-232C 케이블	CX-Programmer  	Toolbus	○	×

네트워크상의 PLC와 연결하는 경우

● 직접 시리얼 연결된 PLC를 경유하여 연결하는 방법

다음과 같이 네트워크 종류를 선택하십시오.

○: 연결 가능 ×: 연결 불가능

연결 케이블	시스템 구성	PLC 기종 변경 화면 (네트워크 종류)	자동 온라인 연결	
			직접 연결	EtherNet/IP 연결
USB 케이블	USB를 경유하여 EtherNet/IP 네트워크에 액세스하는 경우(*1) 	USB	×	○
	USB를 경유하여 FINS 네트워크에 액세스하는 경우(*2)  * 자체 네트워크 테이블을 설정해야 함	Toolbus (USB 포트)	×	×
RS-232C 케이블	RS-232C를 경유하여 FINS 네트워크에 액세스하는 경우(*2)  * 자체 네트워크 테이블을 설정해야 함	Toolbus	×	×

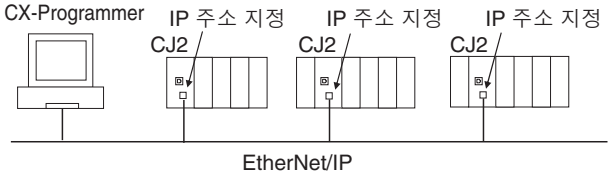
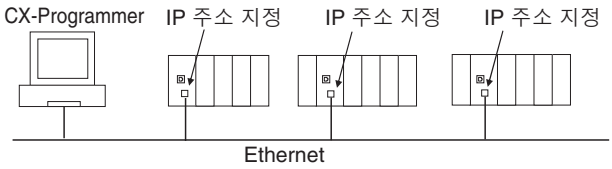
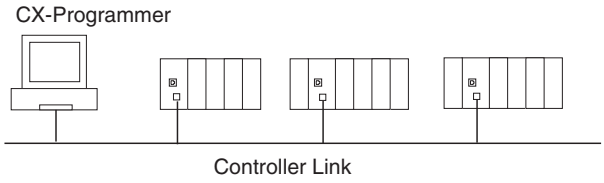
*1 PLC 기종 설정 변경 화면에서 네트워크 종류를 설정한 경우, EtherNet/IP에 대한 연결은 자체 네트워크만 가능합니다. 네트워크 계층을 벗어나서 연결할 수 없습니다. 자동 온라인 연결로 EtherNet/IP와 연결한 경우에는 네트워크의 계층을 벗어나서 연결할 수 있습니다.

*2 FINS 네트워크 간의 계층을 벗어나서 연결할 수 있습니다.

● 네트워크를 경유하여 연결하는 방법

다음과 같이 네트워크 종류를 선택하십시오.

○: 연결 가능 ×: 연결 불가능

연결할 네트워크 종류	시스템 구성	PLC 기종 변경 화면 (네트워크 종류)	자동 온라인 연결	
			직접 연결	EtherNet/IP 연결
EtherNet/IP	EtherNet/IP 네트워크에 액세스하는 경우(*1)  연결 대상 IP 주소를 설정해야 합니다.	EtherNet/IP	×	○
Ethernet	FINS 네트워크로서 Ethernet에 액세스하는 경우(*2)  연결 대상 FINS 네트워크 주소, 노드 주소, IP 주소를 설정해야 합니다.	Ethernet, Ethernet (FINS/TCP)	×	×
Controller Link	 	Controller Link	×	×

*1 PLC 기종 설정 변경 화면에서 네트워크 종류를 설정한 경우, EtherNet/IP에 대한 연결은 자체 네트워크만 가능합니다. 네트워크 계층을 벗어나서 연결할 수 없습니다. 자동 온라인 연결로 EtherNet/IP와 연결한 경우에는 네트워크의 계층을 벗어나서 연결할 수 있습니다.

*2 FINS 네트워크 간의 계층을 벗어나서 연결할 수 있습니다.

사용상의 주의 사항

CJ2H-CPU6□-EIP, CJ2M-CPU3□의 내장 EtherNet/IP 포트를 경유하여 주변 툴과 연결하는 경우, 다음과 같은 이상이 발생했을 때는 연결할 수 없습니다.

I/O 버스 이상(CJ2M-CPU3□에서는 연결 가능)
I/O 점수 초과
유니트 번호 이중 사용 에러
증설 번호 이중 사용 에러
I/O 설정 이상

위와 같은 이상이 발생하고 주변 툴과 연결할 수 없는 경우, CPU 유니트의 페리페럴(USB) 포트를 경유하여 주변 툴과 연결하십시오.

사용상의 주의 사항

EtherNet/IP 연결 전에

Windows XP(SP2 이상)/Vista에서 방화벽을 이용한 보안을 강화하여, Ethernet 포트의 데이터 송수신 제한이 강화되었습니다. 따라서 PC의 Ethernet 포트에서 CJ2 CPU 유니트에 다음과 같은 방법으로 EtherNet/IP 연결을 하는 경우는 Windows 방화벽 설정을 변경하여 CX-Programmer의 통신을 사용할 수 있도록 해야 합니다.

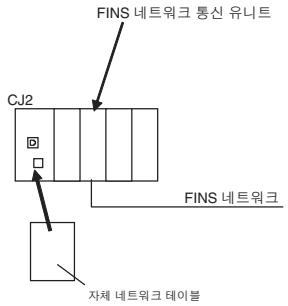
- '네트워크 종류'로 'EtherNet/IP'를 선택하여 온라인으로 연결할 때
- [PLC] 메뉴에서 [자동 온라인]－[이더넷/IP 노드 온라인]을 선택하여 자동으로 EtherNet/IP상의 PLC와 온라인으로 연결할 때

구체적인 조작 방법은 '부록-6 Windows XP(SP2 이상)/Vista에서 EtherNet/IP를 연결하기 전에'를 참조하십시오.

참고

라우팅 테이블을 작성/전송해야 하는 경우

CJ2 CPU 유니트에 FINS 네트워크 통신 유니트(*)가 1 대라도 장착된 경우, 연결한 지원 툴 등에서 CJ2H-CPU6□-EIP, CJ2M-CPU3□의 내장 EtherNet/IP 네트워크, FINS 네트워크의 쌍방에 액세스할 수 있도록 하려면 CX-Integrator에서 라우팅 테이블의 자체 네트워크 테이블을 작성/전송해야 합니다.



* 'FINS 네트워크 통신 유니트'는 Controller Link 유니트, SYSMAC LINK 유니트, Ethernet 유니트, DeviceNet 유니트, FL-net 유니트를 가리킵니다. 또한 시리얼 통신에서 시리얼 게이트웨이 기능과 라우팅 테이블을 사용하는 경우를 포함합니다.
CompoNet 유니트는 포함되지 않습니다.

참고

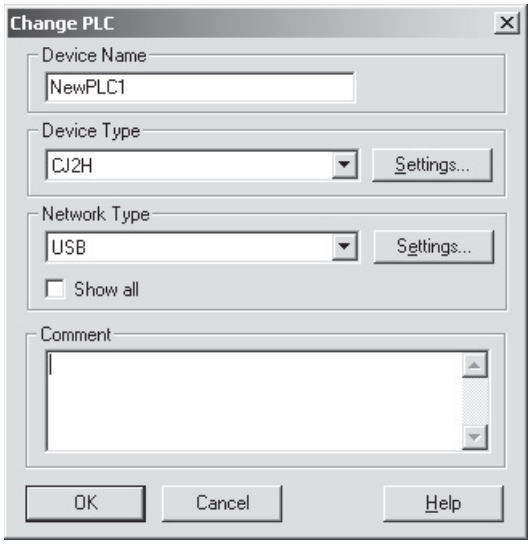
CJ2H-CPU6□-EIP, CJ2M-CPU3□에 FINS 네트워크 통신 유니트가 1대라도 장착된 경우는 반드시 자체 네트워크 테이블을 작성/전송하십시오.
자체 네트워크 테이블을 작성/전송하지 않고 주변 툴 등에서 네트워크 주소＝0(기본값)으로 액세스하면 다음과 같은 경우, 내장 EtherNet/IP 포트의 노드 주소에 액세스합니다.

내장 EtherNet/IP 포트의 유니트 번호(기본값 0)가 장착된 FINS 네트워크 통신 유니트의 유니트 번호보다 작을 때

11-1-3 CX-Programmer에서 PLC에 대한 액세스 조작

PLC 변경 화면 조작

프로젝트를 새로 작성할 때 [PLC 변경] 대화 상자에서 다음과 같은 순서로 PLC에 대한 연결 방법을 선택합니다. 여기에서는 CJ2H CPU 유니트를 사용하는 경우를 예로 들어 설명하겠습니다.



1 **[장치 종류]에서 'CJ2H'를 선택합니다.**

2 **[네트워크 종류]에서 PLC에 대한 연결 방법을 선택합니다.**

기본값으로 다음 중 하나를 선택할 수 있습니다.

- USB
- EtherNet/IP

'전체 보기' 확인란을 선택하면 위의 항목 이외의 네트워크 종류도 표시되므로 필요에 따라 선택하십시오.

네트워크 종류는 두 가지 조건에 따라 다음 표와 같이 선택하십시오.

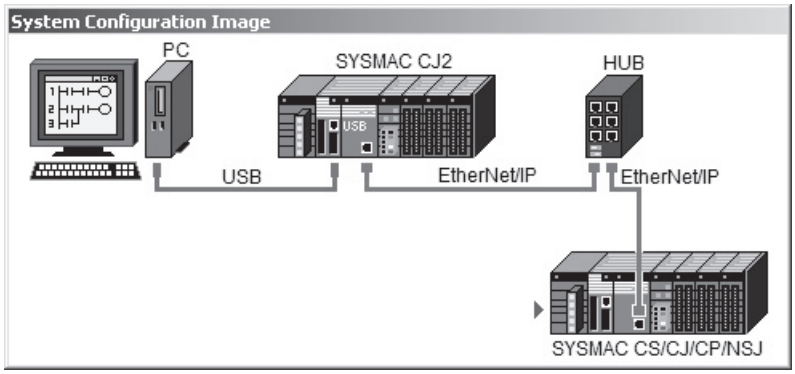
조건 1: PLC와 연결하는 케이블의 종류	조건 2:	선택할 네트워크 종류	
USB	–	USB	기본값으로 표시
Ethernet	EtherNet/IP 네트워크에 액세스하는 경우	EtherNet/IP	
USB	USB를 경유하여 FINS 네트워크에 액세스하는 경우	Toolbus(USB 포트)	'전체 보기'를 선택하여 표시
Ethernet	FINS 네트워크로 Ethernet에 액세스하는 경우	Ethernet	
		Ethernet(FINS/TCP)	
RS-232C	–	Toolbus	
		SYSMAC WAY	
Controller Link (Controller Link 보드로 연결)	–	Controller Link	
FinsGateway	–	FinsGateway	

3 **[설정] 버튼을 클릭하면 선택한 네트워크 종류에 따라 [네트워크 설정] 화면이 표시됩니다.**

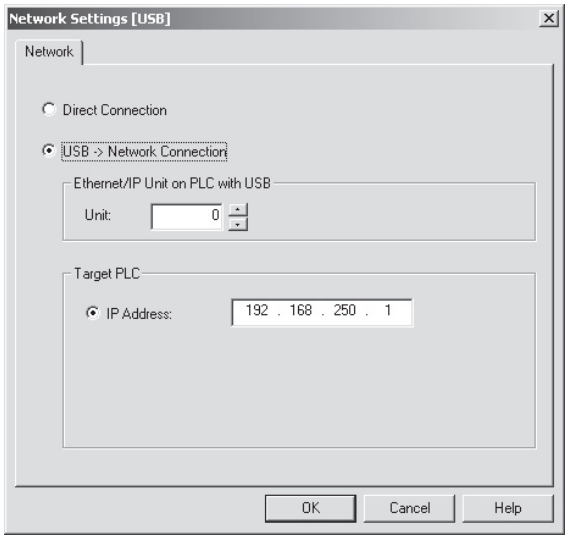
여기에서는 [네트워크 종류]에 기본값으로 표시된 'USB'와 'EtherNet/IP'의 설정에 대해 설명합니다.

다른 네트워크 종류에 대한 설정은 "CX-Programmer 조작 설명서"(SBCA-337)를 참조하십시오.

네트워크 종류: 'USB' 선택 시

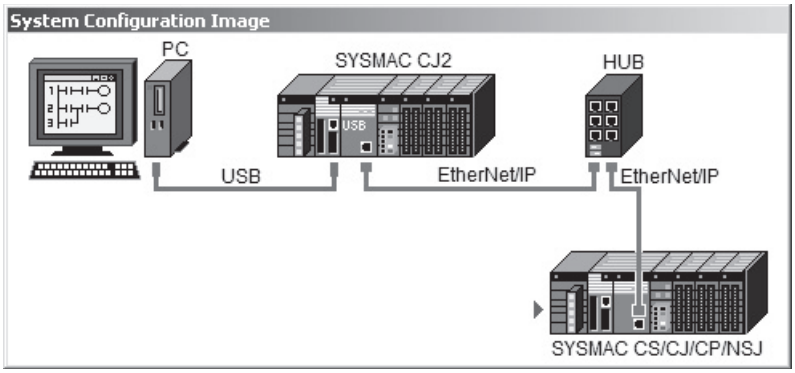


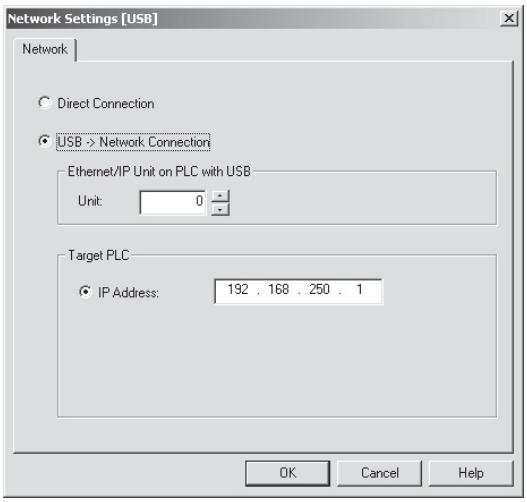
[네트워크] 탭에서 'USB－>네트워크 연결' 선택 시



- '직접 연결'
  USB를 통해 직접 연결된 PLC에 액세스할 때 선택합니다.
- 'USB->네트워크 연결'
  USB를 경유하여 EtherNet/IP 네트워크(단일 네트워크)상의 PLC에 액세스할 때 선택합니다.
  선택한 후 다음과 같이 설정합니다.
  - 'USB로 연결된 PLC의 EtherNet/IP 유니트'
    CJ2H-CPU6□-EIP, CJ2M-CPU3□의 내장 EtherNet/IP 포트를 경유하여 연결하는 경우, 유니트: '0'으로 설정합니다.
  - '연결 대상 PLC'의 'IP 주소'
    CJ2H-CPU6□-EIP, CJ2M-CPU3□의 내장 EtherNet/IP 포트 또는 EtherNet/IP 유니트의 IP 주소를 설정합니다.
    CJ2H-CPU6□-EIP, CJ2M-CPU3□의 내장 EtherNet/IP 포트의 IP 주소는 기본적으로 다음과 같이 설정되어 있습니다.
    '192.168.250.노드 주소'

네트워크 종류: 'EtherNet/IP' 선택 시:





- '연결 대상 PLC'의 'IP 주소'
  CJ2H-CPU6□-EIP, CJ2M-CPU3□의 내장 EtherNet/IP 포트 또는 EtherNet/IP 유니트의 IP 주소를 설정합니다.
  CJ2H-CPU6□-EIP, CJ2M-CPU3□의 내장 EtherNet/IP 포트의 IP 주소는 기본적으로 다음과 같이 설정되어 있습니다.
- '192.168.250.노드 주소'

- PC 측의 IP 주소 설정
  EtherNet/IP를 경유하여 PLC와 연결하는 경우, PC 측의 IP 주소도 CJ2H-CPU6□-EIP, CJ2M-CPU3□의 내장 EtherNet/IP 포트 또는 EtherNet/IP 유니트의 IP 주소와 일치하도록 설정해야 합니다.
  CJ2H-6□-EIP, CJ2M-CPU3□의 내장 EtherNet/IP 포트에 기본적으로 설정된 IP 주소와 일치시키는 경우, PC 측의 IP 주소는 다음과 같이 설정하십시오.
  설정 예:
  - IP 주소

192.168.250. 2

- 2: 임의(1 ~254)의 값
- 192.168.250.: 내장 EtherNet/IP 포트의 IP 주소와 같은 세그먼트의 값

  - 서브넷 마스크

255.255.255. 0. 200

- 200: 임의(0~252)의 값
- 255.255.255.: 클래스 C의 마스크 상태

11-1-4 자동 온라인 연결

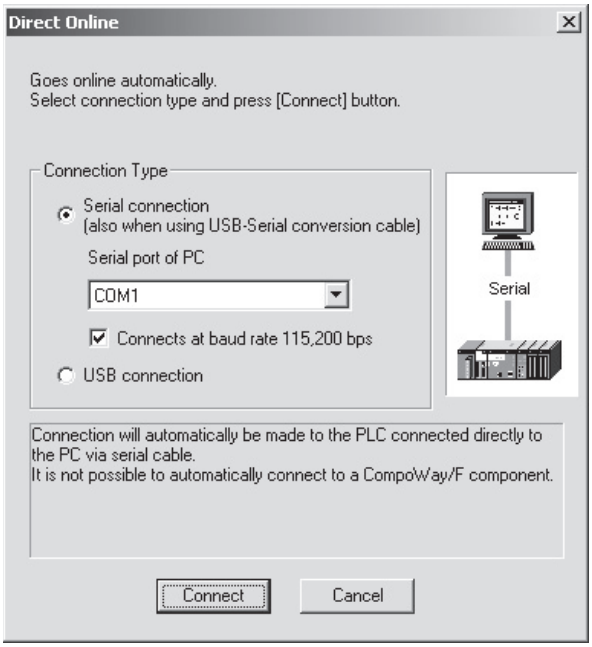
PC에 대상 PLC의 프로젝트가 없으면 온라인으로 연결하여 프로그램을 업로드할 수 있습니다.

직접 연결

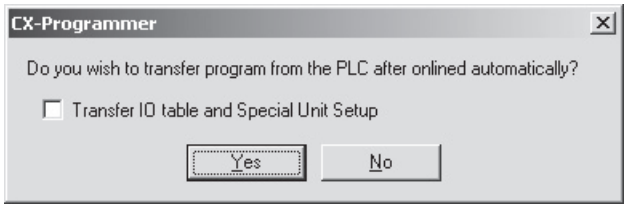
자동 온라인 연결을 실행하면 사용 가능한 PC의 시리얼 포트를 자동으로 검색하고, 온라인 연결이 가능한 시리얼 포트를 찾은 경우 검색된 시리얼 포트에서 온라인으로 자동 연결합니다.

1 **[PLC] 메뉴에서 [자동 온라인] – [다이렉트 온라인]을 선택합니다.**
다음과 같이 [다이렉트 온라인] 대화 상자가 표시됩니다.

2 **PC와 PLC의 연결 방법을 다음 중에서 선택합니다.**
- '시리얼 연결'(PC 측 USB – PLC 측 RS-232C 변환을 사용하는 경우 포함)
- 'USB 연결'



PC의 USB 포트에서 CJ2 CPU 유니트의 페리페럴(USB) 포트에 연결한 경우는 'USB 연결'을 선택한 후 [연결] 버튼을 클릭합니다.
다음과 같은 대화 상자가 표시됩니다.



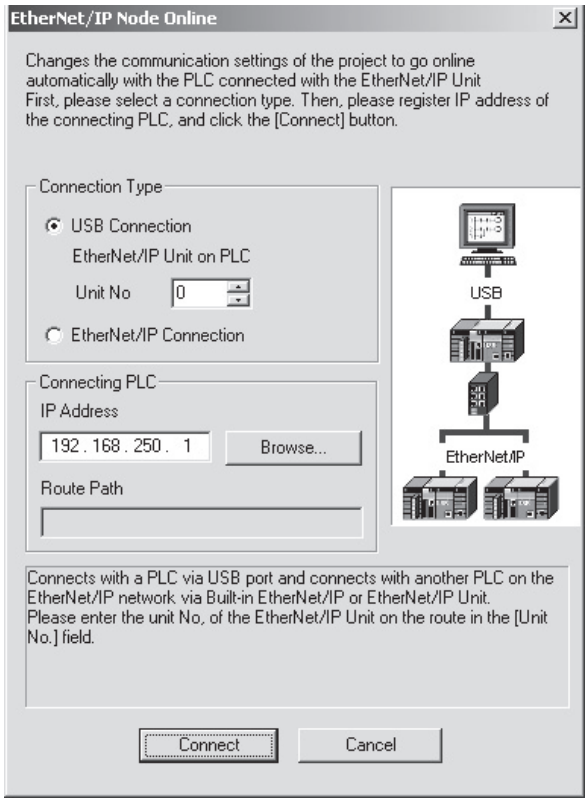
자동 온라인 연결 후 프로그램, PLC 시스템 설정 및 I/O 테이블을 PC에 업로드하려면 [예] 버튼을 클릭합니다.
PLC와의 자동 온라인 연결이 실행되고 PLC에서 업로드됩니다.

EtherNet/IP 노드 연결

EtherNet/IP를 경유하여 EtherNet/IP상의 PLC에 온라인으로 연결합니다.
연결 대상 PLC의 IP 주소를 모르는 경우도 검색해서 액세스할 수 있습니다.

1 **[PLC] 메뉴에서 [자동 온라인] - [이더넷/IP 노드 온라인]을 선택합니다.**

다음과 같이 [이더넷/IP 노드 온라인] 대화 상자가 표시됩니다.



2 **[연결 종류]에서 연결 대상 PLC와 연결하는 방법을 선택합니다.**

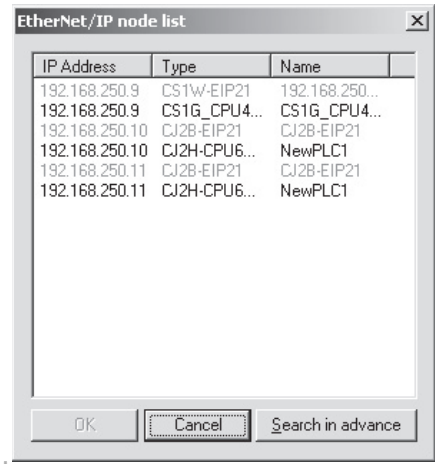
- 'USB 연결'
  PLC와 USB로 직접 연결하고 내장 EtherNet/IP 포트 또는 EtherNet/IP 유니트를 경유하여 EtherNet/IP 네트워크(단일 네트워크)의 PLC와 연결하는 경우에 선택합니다.
- 'EtherNet/IP 연결'
  PC를 EtherNet/IP 네트워크에 연결하고 EtherNet/IP 네트워크를 경유해서 네트워크(단일 네트워크)의 PLC와 연결하는 경우에 선택합니다.

3 **[연결 대상 PLC]에서 연결 대상 PLC를 설정하는 방법에는 다음의 2종류가 있습니다.**

- IP 주소
  연결 대상 IP 주소를 직접 입력합니다.
- 노드 목록
  네트워크에 연결된 온라인 상태의 PLC를 검색하여 연결 대상 PLC를 선택합니다.
  연결 대상 PLC의 IP 주소를 모를 때 선택하십시오.

연결된 노드는 노드 목록 또는 네트워크의 계층 별 표시 중 하나로 표시할 수 있습니다.

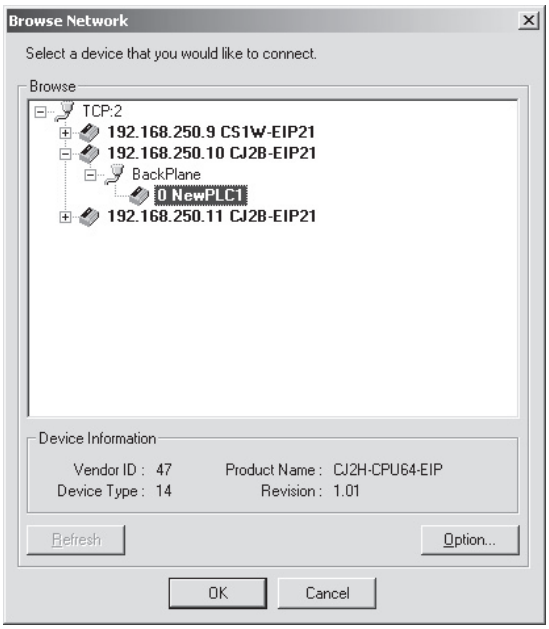
(1) 노드 목록



- IP 주소, PLC 형식 및 이름 정보가 표시됩니다.
  IP 주소를 선택하고 [확인] 버튼을 클릭합니다.

(2) 네트워크 참조

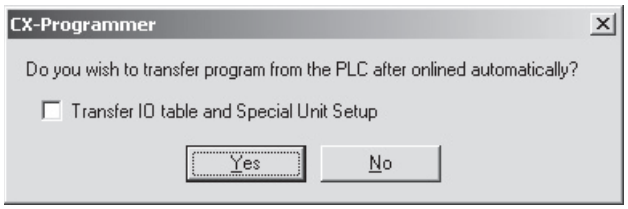
위와 같이 [이더넷/IP 노드 목록] 대화 상자의 '우선 검색' 버튼을 클릭하면 다음 화면이 표시됩니다.



- 탐색용 '+' 표시를 클릭하여 계층적으로 구성을 표시합니다.
  연결하는 PLC를 선택하고 [확인] 버튼을 클릭합니다.

4 **[연결] 버튼을 클릭하면 온라인으로 연결됩니다.**

다음과 같은 대화 상자가 표시됩니다.



자동 온라인 연결 후 프로그램, PLC 시스템 설정 및 I/O 테이블을 PC에 업로드하려면 [예] 버튼을 클릭합니다.

PLC와의 자동 온라인 연결이 실행되고 PLC에서 업로드됩니다.

11-2 시리얼 통신

11-2-1 시리얼 통신

시리얼 포트의 모드(프로토콜)는 CPU 유니트의 PLC 시스템 설정에서 전환할 수 있습니다.
시리얼 포트의 모드 전환을 통해 다음과 같은 각종 시스템을 구축할 수 있습니다.

각 프로토콜 목록

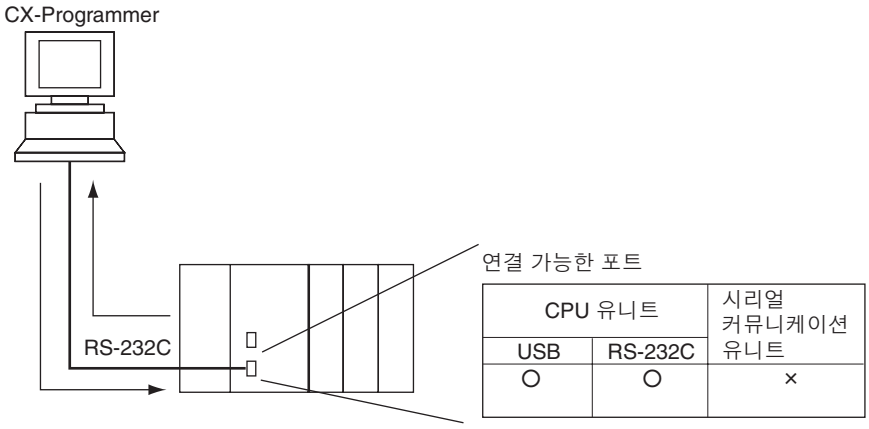
시리얼 통신에서 지원하는 프로토콜은 다음과 같습니다.

프로토콜 (시리얼 통신 모드)	주요 연결 대상	개요	대응 명령/통신 명령
툴 버스 (CPU 유니트의 시리얼 포트만 가능하며 시리얼 커뮤니케이션 유니트에서는 불가능)	CX-Programmer	PC의 각종 주변 툴과 PLC 간의 통신용	없음
상위 링크(SYSWAY) 자국	PC 오므론이 제작한 PT (프로그래머블 터미널)	상위 PC와 PLC 간의 통신용. PLC에서의 강제도 가능	상위 링크 명령/FINS 명령 (강제 가능)
무수순(*)	범용 외부 기기	범용 외부 기기와의 무수순 통신용	TXD 명령, RXD 명령 TXDU 명령, RXDU 명령 DTXDU 명령, DRXDU 명령
NT 링크 (1:N 모드)	오므론이 제작한 PT (프로그래머블 터미널)	직접 액세스를 사용한 PT와의 고속 통신용	없음
시리얼 게이트웨이	오므론이 제작한 구성 요소 PLC	FINS 명령을 수신하여 CompoWay/F, Modbus 또는 상위 링크로 프로토콜 변환해서 시리얼 통신상에 전송합니다.	–
CompoWay/F 친국(*)	CompoWay/F 자국	시리얼 포트(CompoWay/F 캡슐화)에서 FINS 명령을 수신하여 CompoWay/F 명령으로 변환 가능	FINS 명령 2803Hex 수신 (CMND 명령에 의한 FINS 명령 전송 포함)
Modbus 친국(*) (시리얼 커뮤니케이션 유니트만 가능하며 CPU 유니트의 시리얼 포트에서는 불가능)	Modbus 자국	시리얼 포트(Modbus 캡슐화)에서 FINS 명령을 수신하여 Modbus 명령으로 변환 가능	FINS 명령 2804Hex 수신 또는 2805Hex 수신(CMND 명령에 의한 FINS 명령 전송 포함)
상위 링크 FINS(SYSWAY) 친국(*) (시리얼 커뮤니케이션 유니트만 가능하며 CPU 유니트의 시리얼 포트에서는 불가능)	상위 링크 FINS (SYSWAY) 자국(PLC)	FINS 명령을 수신하여 상위 링크에서 캡슐화한 FINS 명령으로 변환 가능	시리얼 포트 이외의 임의의 FINS 명령 수신(CMND 명령에 의한 FINS 명령 전송 포함)
프로토콜 매크로 (시리얼 커뮤니케이션 유니트만 가능하며 CPU 유니트의 시리얼 포트에서는 불가능)	범용 외부 기기	범용 외부 기기의 통신 사양에 따른 메시지(통신 프레임) 송수신용 (CX-Protocol을 사용하여 프로토콜을 파라미터 형식으로 작성)	PMCR 명령
시리얼 PLC 링크 (CJ2M CPU 유니트만 해당)	오므론이 제작한 PLC(CJ2M/CJ1M/CP1H/CP1L/CP1E CPU 유니트)	최대 9대(친국 1대, 자국 최대 8대)의 CPU 유니트 간에 1유니트당 최대 10CH의 데이터 공유 가능	없음

* 시리얼 커뮤니케이션 유니트는 유니트 Ver.1.2 이상만 해당

툴 버스

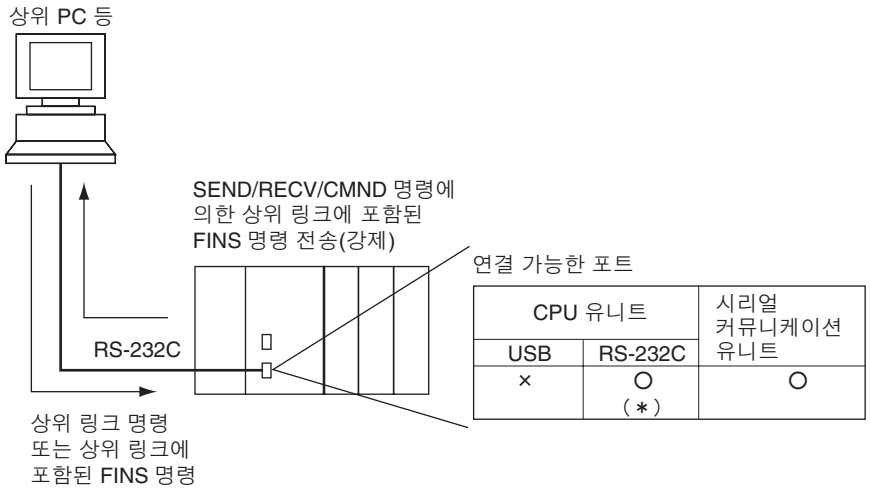
CX-Programmer 등의 주변 툴 CX-One과 연결합니다.



CPU 유니트		시리얼 커뮤니케이션 유니트
USB	RS-232C	
○	○	×

상위 링크 시스템(SYSWAY 1:N 모드) (자국)

상위 컴퓨터(PC)와 PT(프로그래머블 터미널)에서 상위 링크 명령 또는 상위 링크의 헤더/터미네이터 등으로 감싼 FINS 명령을 전송하고 PLC의 I/O 메모리나 동작 모드 등을 읽고 씁니다.
또한 PLC 측에서 상위 컴퓨터에 대해 상위 링크의 헤더/터미네이터 등으로 감싼 FINS 명령을 자발적으로(SEND/RECV/CMND 명령을 통해) 전송하는 것도 가능(송신 기능)합니다.

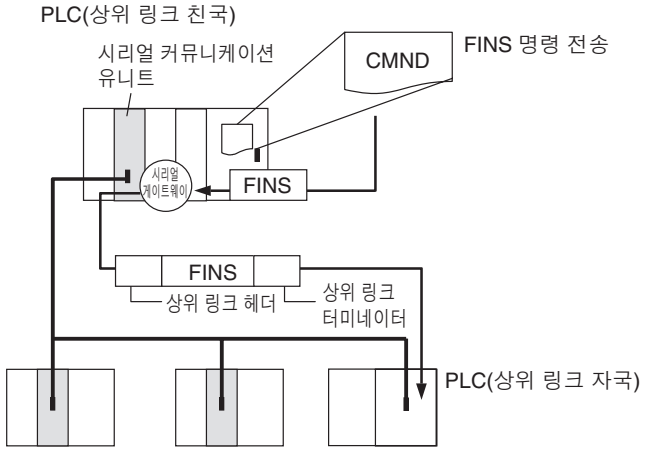


CPU 유니트		시리얼 커뮤니케이션 유니트
USB	RS-232C	
×	○ (*)	○

* CPU 유니트 전면의 딥 스위치 SW5를 OFF로 하고 PLC 시스템 설정의 시리얼 통신 모드를 '상위 링크'로 설정하십시오.

참고

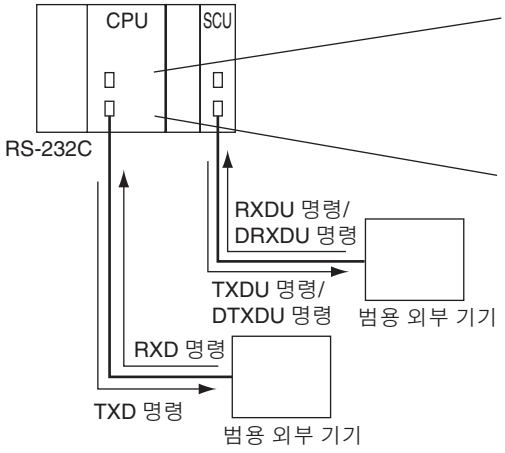
시리얼 커뮤니케이션 유니트의 유니트 Ver.1.2 이상인 경우, 시리얼 게이트웨이 기능을 사용하여 CMND 명령을 통한 상위 링크의 친국 기능을 구현할 수 있습니다.



무수순 시스템

통신 포트 입출력 명령을 사용하여 바코드의 데이터 입력 또는 프린터의 데이터 출력 등과 같은 간단한 (무수순) 송수신을 구현할 수 있습니다. 시작/종료 코드의 유무 지정이나 RS, CS 등의 제어 신호를 컨트롤할 수도 있습니다.

통신 포트 입출력 명령은 사용하는 통신 포트 또는 송수신에 따라 다음과 같이 구분해서 사용하십시오.



연결 가능한 포트

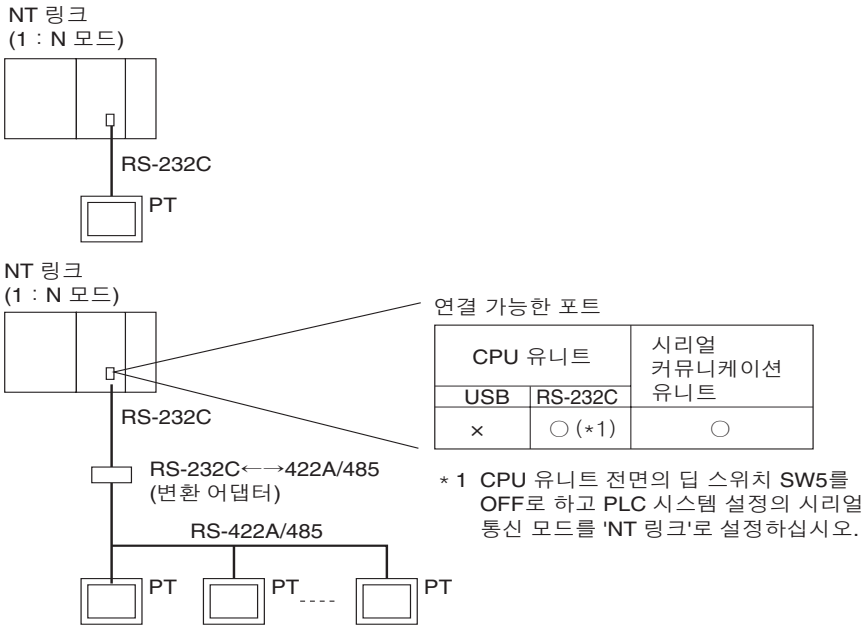
CPU 유니트		시리얼 커뮤니케이션 유니트
USB	RS-232C	
×	○(*1)	○(*2)

*1 CPU 유니트 전면의 딥 스위치 SW5를 OFF로 하고 PLC 시스템 설정의 시리얼 통신 모드를 '무수순'으로 설정하십시오.

*2 시리얼 커뮤니케이션 유니트의 유니트 Ver.1.2 이상만 무수순이 가능합니다.

NT 링크 시스템(1:N 모드)

PLC와 PT(프로그래머블 터미널) 사이를 시리얼 포트로 연결하고, PLC의 I/O 메모리 내에 PT에 대한 상태 제어 영역과 상태 통지 영역 및 각 터치 스위치, 램프, 메모리 테이블 등의 개체를 할당합니다.
PLC에서 PT에 대한 제어는 PT에서 PLC에 대해 정기적으로 상태 제어 영역의 데이터를 읽어온 후 변경되었으면 적절한 동작을 수행합니다. PT에서 PLC에 대한 통지는 PT에서 PLC의 상태 통지 영역에 대해 쓰기를 실행합니다.
그 결과, PLC 측의 래더 프로그램 없이 PT의 상태를 제어 및 모니터링할 수 있습니다. PLC:PT＝1:n(n 1)이 가능합니다.
PT 측의 통신 방식을 NT 링크(1:N 모드)로 설정하십시오. 또한 NT 링크(1:N 모드)는 PLC:PT가 1:1 연결, 1:다 연결 중 어느 경우든 가능합니다.



사용상의 주의 사항

- NT 링크 시스템에서는 NT 링크(1:N 모드)를 지원하는 PT 측 포트와 연결하여 통신할 수 있습니다. NT 링크(1:1 모드)만 지원하는 PT 측 포트(NT30/30C의 RS-232C 포트)에는 연결해도 통신되지 않으므로 주의하십시오.
- NT20S, NT600S, NT30/30C, NT620S/620C/625C를 사용하는 경우, CPU 유니트의 사이클 시간이 800ms 이상에서는 사용할 수 없으므로 주의하십시오. 이 중 한 PT를 1:N에서 1대라도 포함하는 경우도 동일합니다.
- PLC:PT＝1:N으로 연결하는 경우 PT의 기기 번호가 중복되지 않도록 하십시오. 중복되면 오동작합니다.

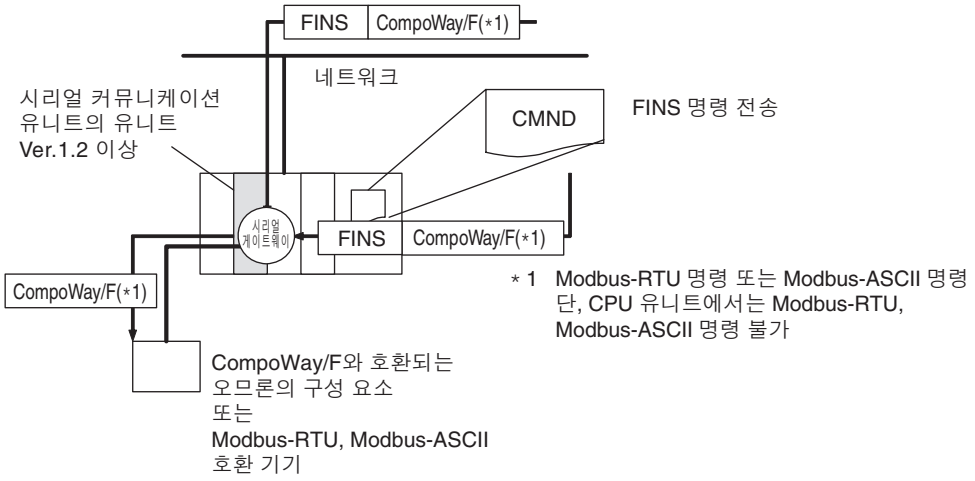
참고

NT 링크에는 1:1 모드와 1:N 모드가 있지만 통신 모드로서의 호환성은 없는 별도의 시리얼 통신입니다.

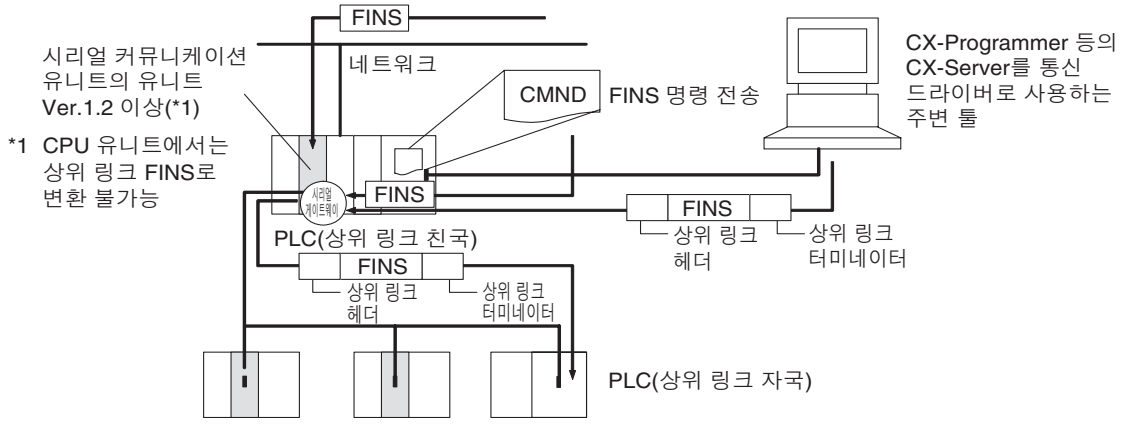
시리얼 게이트웨이 모드

수신한 FINS 메시지를 해당 메시지에 따라 CompoWay/F로 자동 변환합니다.
시리얼 커뮤니케이션 유니트의 유니트 Ver.1.2 이상인 경우, 수신한 FINS 메시지를 해당 메시지에 따라 CompoWay/F, Modbus-RTU, Modbus-ASCII, 상위 링크 FINS로 자동 변환합니다.

● CompoWay/F, Modbus-RTU, Modbus -ASCII



● 상위 링크 FINS

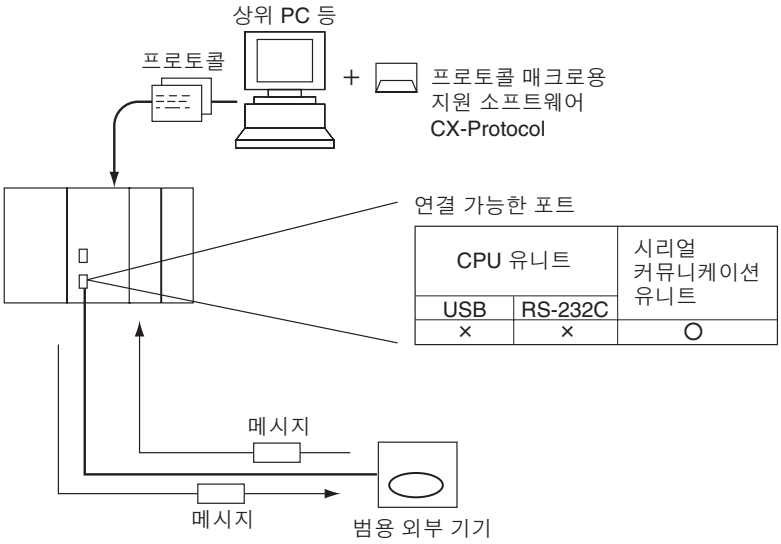


프로토콜 매크로 기능(시리얼 커뮤니케이션 유니트만 가능)

범용 외부 기기의 통신 사양(반이중 또는 전이중, 비동기식이 조건)에 따라 범용 외부 기기와의 데이터 송수신 순서(프로토콜)를 CX-Protocol로 작성할 수 있습니다. 해당 프로토콜을 시리얼 커뮤니케이션 유니트에 저장하고 CPU 유니트의 PMCR 명령을 실행하는 것만으로도 범용 외부 기기와 데이터 송수신이 가능합니다.

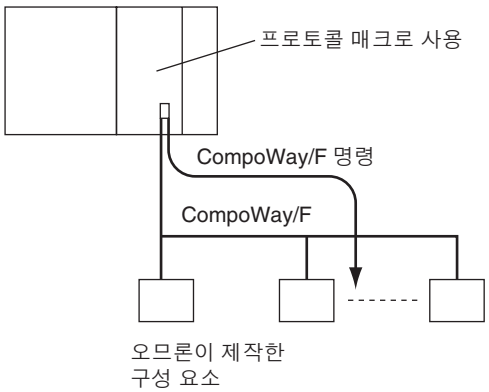
그리고 오므론이 제작한 구성 요소(온도 조절기, 패널 미터, 바코드 리더, 모뎀 등)와의 데이터 교환용 프로토콜을 표준 시스템 프로토콜로 제공하고 있습니다(*). 또한 이것을 사용자의 필요에 따라 변경해서 사용할 수도 있습니다.

* CX-Protocol, 시리얼 커뮤니케이션 유니트에 표준 탑재되어 있습니다.



(주) CompoWay/F(친국 기능)

CJ 시리즈가 호스트(친국)가 되어 오므론이 제작한 구성 요소에 대해 CompoWay/F 명령을 전송할 수 있습니다. 프로토콜 매크로의 표준 시스템 프로토콜 내에서 CompoWay/F용 송수신 시퀀스를 사용하여 구현합니다.



시리얼 PLC 링크 기능(CJ2M CPU 유니트만 해당)

이 기능은 CJ2M CPU 유니트에서만 사용할 수 있습니다.
시리얼 포트에서 CJ2M CPU 유니트끼리 또는 CJ1M/CP1H/CP1L/CP1E CPU 유니트 간에 프로그램 없이 데이터를 교환하는 기능입니다.
시리얼 PLC 링크 릴레이(3100~3199CH)를 사용합니다.
CPU 유니트 사이는 RS-232C 또는 RS-422A/485로 연결합니다. CJ2M-CPU1□의 시리얼 포트(RS-232C)를 RS-422A/485로 변환하려면 RS-422A 변환 어댑터(CJ1W-CIF11)를 사용합니다.

NT 링크(1:N)에 설정한 PT와 회선상에서 혼재도 가능합니다. 자국의 PT는 이 회선을 사용하여 친국의 CPU 유니트와 NT 링크(1:N) 통신을 수행합니다.
단, PT가 연결된 경우, PT의 자국 번호에 해당하는 시리얼 PLC 링크 릴레이의 영역은 불안정해집니다.

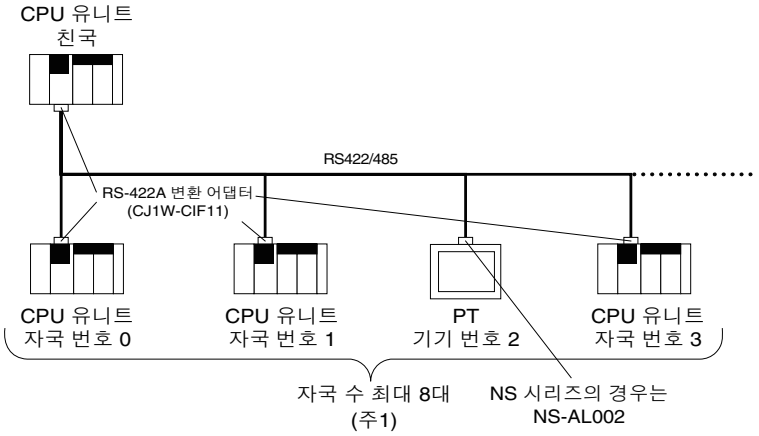
● 동작 사양

설정 항목	설정 내용
연결 PLC 기종	CJ2M, CJ1M, CP1H, CP1L, CP1E
통신 속도	115200비트/s, 38400비트/s
대상 시리얼 포트	CPU 유니트의 시리얼 포트(CJ2M-CPU3□의 경우 시리얼 옵션 보드 장착 시)
연결 방식	RS-232C 또는 RS-422A/485 연결
할당 릴레이 영역	시리얼 PLC 링크 릴레이: 3100~3199CH(CPU 유니트 1대당 최대 10CH) (*)
최대 연결 수	9대(친국 1대, 자국 8대) 단, NT 링크(1:N)로 설정한 PT를 회선상에 혼재시키는 경우는 자국과 PT를 합하여 8대까지
연결 방식 (데이터 갱신 방식)	전체 국 연결 방식 또는 친국 연결 방식

* CP1E CPU 유니트의 시리얼 PLC 링크 릴레이는 200~289CH에 할당됩니다.

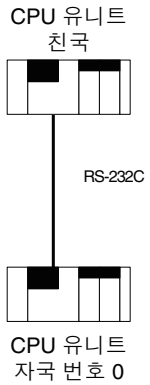
● 시스템 구성

친국:자국 = 1:N(최대 8) 연결 시



주: 단, 시리얼 PLC 링크로 설정한 PT를 회선상에 혼재시키는 경우는 자국과 PT를 합하여 8대까지

친국:자국 = 1:1 연결 시



주: 단, 시리얼 PLC 링크로 설정한 PT를 회선상에 혼재시키는 경우는 자국과 PT를 합하여 8대까지

사용상의 주의 사항

CJ1W-CIF11은 비절연 타입이며 전송 경로 전체의 총 전송 거리는 50m까지입니다. 총 전송 거리가 50m를 초과하는 경우, 절연 타입의 NT-AL001을 사용하고 전송 경로에 CJ1W-CIF11을 사용하지 마십시오. NT-AL001만 사용하는 경우는 전송 경로 전체의 총 전송 거리는 500m까지가 됩니다.

● 사용 순서

시리얼 PLC 링크는 친국/자국이 되는 각 CPU 유니트에 대해 다음과 같은 PLC 시스템 설정을 수행하여 동작합니다.

• 친국 측 설정

1 **'RS-232C 통신 포트 – 시리얼 통신 모드'를 '시리얼 PLC 링크(친국)'로 설정합니다.**

2 **'연결 방식'을 '전체 국 연결 방식' 또는 '친국 연결 방식'으로 설정합니다.**

3 **'링크 CH 수'를 설정합니다(1~10).**

4 **'시리얼 PLC 링크 최대 기기 번호'를 설정합니다(0~7).**

• 자국 측 설정

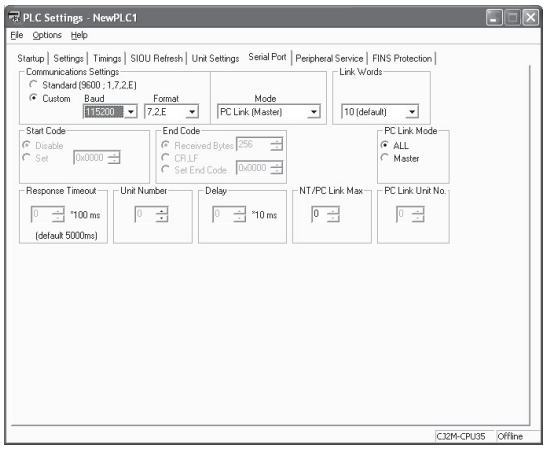
1 **'RS-232C 통신 포트 – 시리얼 통신 모드'를 '시리얼 PLC 링크(자국)'로 설정합니다.**

2 **'시리얼 PLC 링크 자국 기기 번호'를 설정합니다.**

● PLC 시스템 설정

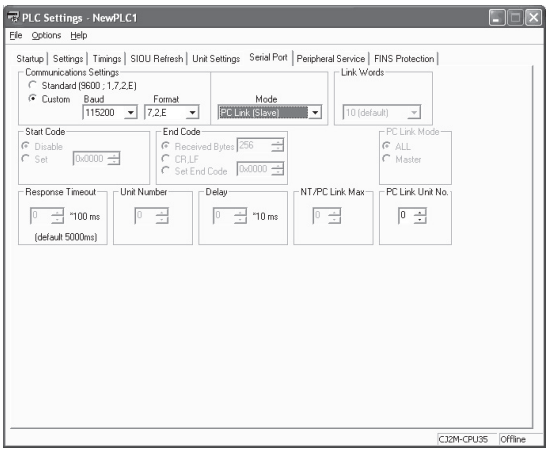
친국 측

[시리얼 포트] 탭



설정 항목	설정 내용
통신 설정	연결 대상 PLC에서 통신 설정을 일치시킵니다. 예: 115200비트/s의 경우, '사용자 지정'을 선택하고 '전송 속도'에서 '115200'을 선택합니다. '파라미터'는 어느 것을 선택해도 상관 없습니다.
모드	'PLC 링크(친국)'를 선택합니다.
링크 채널	친국만으로 설정합니다. 10CH(초기값)
PLC 링크 연결 방식	'전체 국 연결' 또는 '친국 연결'을 선택합니다.
NT/PLC 링크 최대 번호	친국에는 연결 대상 자국의 최대 기기 번호를 설정합니다.

자국 측



설정 항목	설정 내용
통신 설정	연결 대상 PLC에서 통신 설정을 일치시킵니다. 예: 115200비트/s의 경우, '사용자 지정'을 선택하고 '전송 속도'에서 '115200'을 선택합니다. '파라미터'는 어느 것을 선택해도 상관 없습니다.
모드	'PLC 링크(자국)'를 선택합니다.
PLC 링크 자국 기기 번호	각 기기 번호(0 ~ 7)를 설정합니다.

● 연결 방식(데이터 갱신 방식)

데이터 갱신 방식은 다음 두 가지 중에서 선택할 수 있습니다.

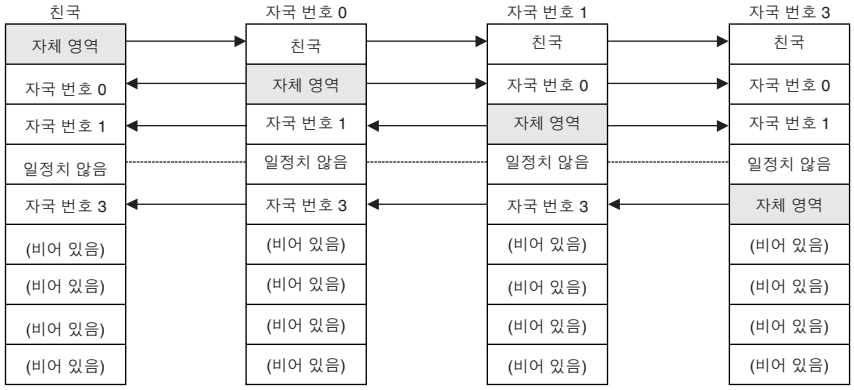
- 전체 국 연결 방식
- 친국 연결 방식

전체 국 연결 방식

친국, 자국과 더불어 다른 모든 국의 데이터를 반영하는 방식입니다.
단, PT가 연결된 기기 번호의 영역 또는 회선상에 존재하지 않는 자국의 영역에 대해서는 국 전체의 데이터가 불안정해집니다.

예: 전체 국 연결 방식, 최대 기기 번호 3인 경우

다음 그림에서 자국 번호 2가 PT 또는 존재하지 않기 때문에 국 전체가 불안정해졌습니다.

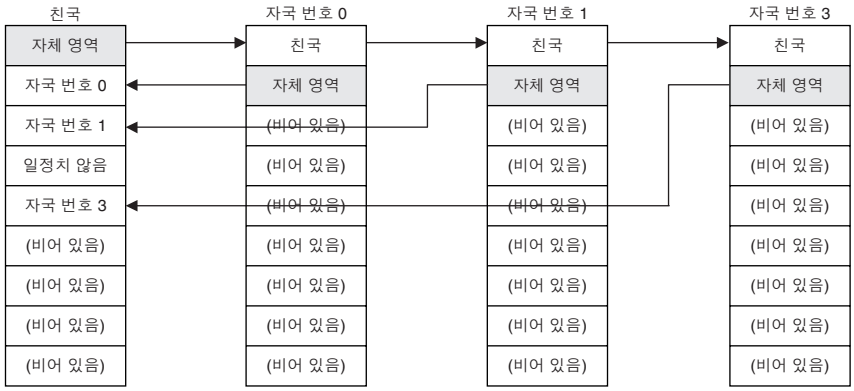


친국 연결 방식

친국은 자국의 모든 데이터를 반영하고 자국은 친국의 데이터만 반영하는 방식입니다.
자국 영역의 주소가 모두 동일해지므로 데이터를 참조하는 래더 프로그램을 공통화할 수 있는 장점이 있습니다.
PT가 연결된 기기 번호의 영역 또는 회선상에 존재하지 않는 자국의 영역에 대해서는 친국에서만 데이터가 불안정해집니다.

예: 친국 연결 방식, 최대 기기 번호 3인 경우

다음 그림에서 자국 번호 2가 PT 또는 존재하지 않기 때문에 친국이 불안정해졌습니다.



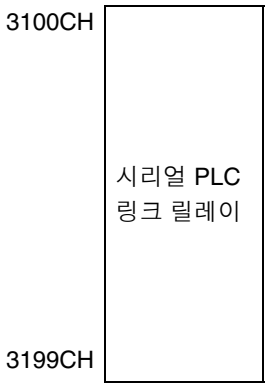
● **할당 릴레이 영역**

전체 국 연결 방식의 경우



링크 CH 수	1CH	2CH	3CH	...	10CH
친국	3100	3100~3101	3100~3102		3100~3109
자국 번호 0	3101	3102~3103	3103~3105		3110~3119
자국 번호 1	3102	3104~3105	3106~3108		3120~3129
자국 번호 2	3103	3106~3107	3109~3111		3130~3139
자국 번호 3	3104	3108~3109	3112~3114		3140~3149
자국 번호 4	3105	3110~3111	3115~3117		3150~3159
자국 번호 5	3106	3112~3113	3118~3120		3160~3169
자국 번호 6	3107	3114~3115	3121~3123		3170~3179
자국 번호 7	3108	3116~3117	3124~3126		3180~3189
빈 영역	3109~3199	3118~3199	3127~3199		3190~3199

친국 연결 방식의 경우



링크 CH 수	1CH	2CH	3CH	...	10CH
친국	3100	3100~3101	3100~3102		3100~3109
자국 번호 0	3101	3102~3103	3103~3105		3110~3119
자국 번호 1	3102	3104~3105	3106~3108		3120~3129
자국 번호 2	3103	3106~3107	3109~3111		3130~3139
자국 번호 3	3104	3108~3109	3112~3114		3140~3149
자국 번호 4	3105	3110~3111	3115~3117		3150~3159
자국 번호 5	3106	3112~3113	3118~3120		3160~3169
자국 번호 6	3107	3114~3115	3121~3123		3170~3179
자국 번호 7	3108	3116~3117	3124~3126		3180~3189
빈 영역	3109~3199	3118~3199	3127~3199		3190~3199

● 관련 특수 보조 릴레이

명칭	주소	설명	읽기/쓰기	반영 타이밍
시리얼 포트에서 PT와의 통신 실행 중 플래그 (*)	A393.00~ A393.07	시리얼 포트가 NT 링크 모드일 때 또는 시리얼 PLC 링크 모드일 때, 통신을 실행하고 있는 해당 비트가 1(ON)이 됩니다. 비트 0~7이 기기 번호 0~7에 대응합니다. 1: 통신 실행 중 0: 통신 비실행 중	읽기	• 전원을 켜면 클리어됩니다. • 시리얼 포트가 NT 링크 모드 또는 시리얼 PLC 링크 모드일 때. • 비트 0~7이 기기 번호 0~7에 대응합니다.
시리얼 포트 재시작 플래그	A526.00	시리얼 포트에 대해 포트를 재시작하는 경우 0→1로 기동하십시오.	읽기/쓰기	• 전원을 켜면 클리어됩니다. • 시리얼 포트에 대해 포트를 재시작하는 경우 0→1로 기동하십시오(툴 버스 모드에서 통신 할 때는 제외). (주) 재시작 처리 종료 후 시스템에 의해 자동으로 0(OFF)이 됩니다.
시리얼 포트 에러 플래그	A528.00~ A528.07	시리얼 포트에서 에러가 발생하면 해당 에러 코드가 저장됩니다. 비트 0: 사용 안 함 비트 1: 사용 안 함 비트 2: 패리티 에러 비트 3: 프레이밍 에러 비트 4: 오버런 에러 비트 5: 시간 초과 에러 비트 6: 사용 안 함 비트 7: 사용 안 함	읽기/쓰기	• 전원을 켜면 클리어됩니다. • 시리얼 포트에서 에러가 발생하면 해당 에러 코드가 저장됩니다. • 시리얼 포트를 재시작 처리할 때 시스템에 의해 클리어됩니다. • 시리얼 PLC 링크 모드일 때는, 친국의 경우: 비트 05: 시간 초과 에러 자국의 경우: 비트 05: 시간 초과 에러 비트 04: 오버런 에러 비트 03: 프레이밍 에러만 유효화됩니다.
시리얼 포트 설정 변경 중 플래그	A619.02	시리얼 포트의 통신 조건 설정을 변경하는 중에 1(ON)이 됩니다. 1: 변경 중 0: 변경 중 아님	읽기/쓰기	• 전원을 켜면 클리어됩니다. • 시리얼 포트의 통신 조건 설정을 변경하는 동안 1(ON)이 됩니다. • 시리얼 포트 설정 변경 명령(STUP 명령)을 실행할 때 1(ON)이 됩니다. • 설정 변경이 완료되면 0(OFF)으로 되돌아갑니다.

* 기존 NT 링크(1:N)와 동일하게 친국의 CPU 유니트에서는 '시리얼 포트에서 PT와 통신 실행 중 플래그'(A393 CH 비트 00~07: 기기 번호 0~7에 대응)를 참조하면 시리얼 PLC 링크상의 자국의 상태(통신 실행 중/비실행 중)를 알 수 있습니다.

유니트/프로토콜 관련 목록

모델명	형식	인터페이스	프로토콜(시리얼 통신 모드)						
			툴 버스	상위 링크	무수순	프로토콜 매크로	NT 링크 (1:N 모드)	시리얼 게이트웨이 (*2)	시리얼 PLC 링크 (*1)
CPU 유니트	CJ2H-CPU68(-EIP) CJ2H-CPU67(-EIP) CJ2H-CPU66(-EIP) CJ2H-CPU65(-EIP) CJ2H-CPU64(-EIP)	RS-232C	○	○	○	–	○	○	×
	CJ2M-CPU35 CJ2M-CPU34 CJ2M-CPU33 CJ2M-CPU32 CJ2M-CPU31	RS-232C 또는 RS-422A/485 (*1)	○	○	○	–	○	○	○
	CJ2M-CPU15 CJ2M-CPU14 CJ2M-CPU13 CJ2M-CPU12 CJ2M-CPU11	RS-232C	○	○	○	–	○	○	○
시리얼 커뮤니케이션 유니트	CJ1W-SCU41-V1 CJ1W-SCU31-V1 CJ1W-SCU21-V1 CJ1W-SCU22 CJ1W-SCU32 CJ1W-SCU42	RS-422A/485	–	○	○ (*2)	○	○	○	×
		RS-232C	–	○	○ (*2)	○	○	○	×

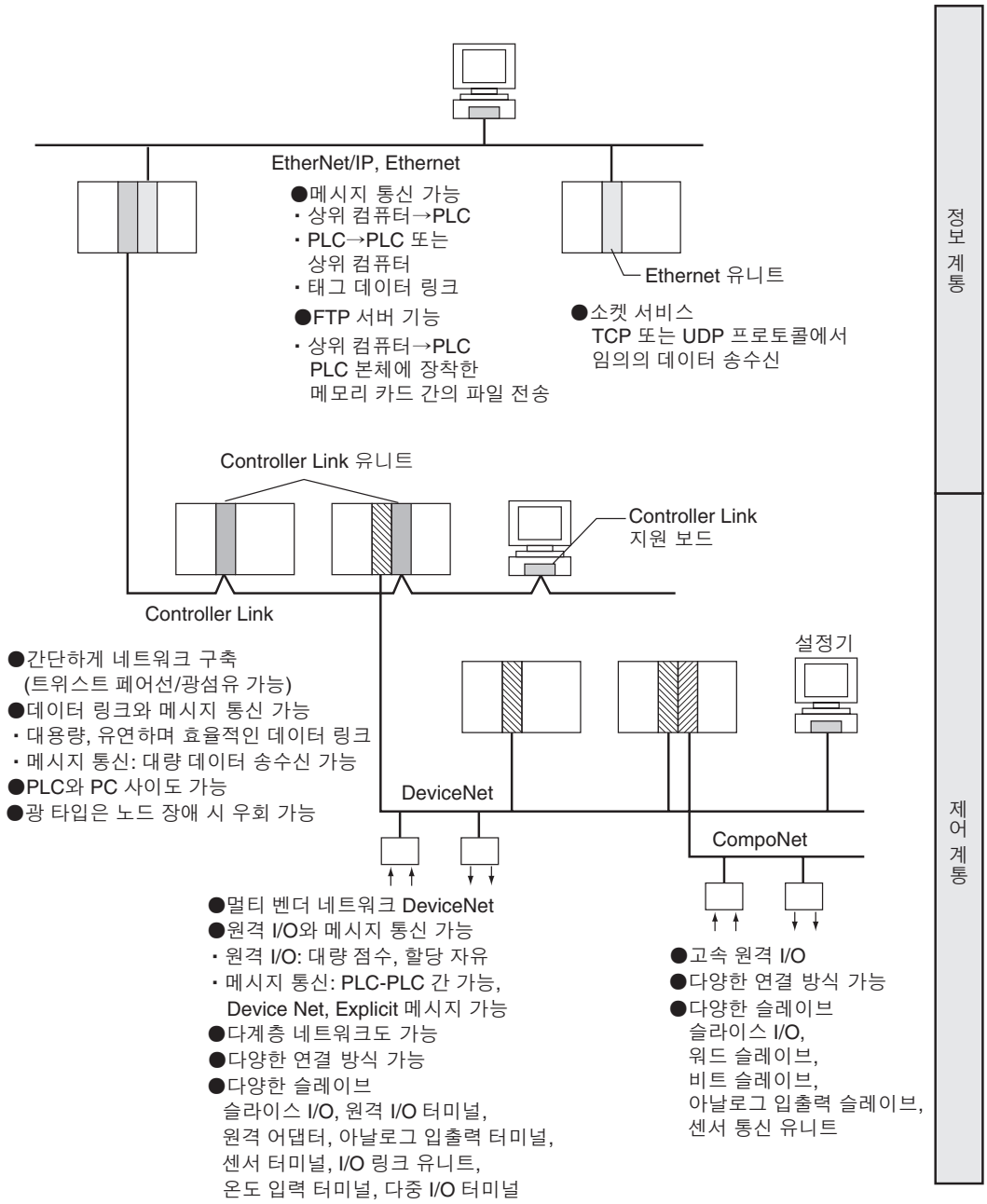
*1 시리얼 옵션 보드를 장착하면 RS-232C 통신 또는 RS-422A/485 통신이 가능해집니다.

*2 시리얼 커뮤니케이션 유니트의 유니트 Ver.1.2 이상만 해당하며, CPU 유니트의 경우 시리얼 게이트웨이는 CompoWay/F로 자동 변환만 가능합니다.

11-3 통신 네트워크

통신 네트워크의 체계

CJ 시리즈에는 다음과 같은 통신 네트워크 체계가 있습니다.

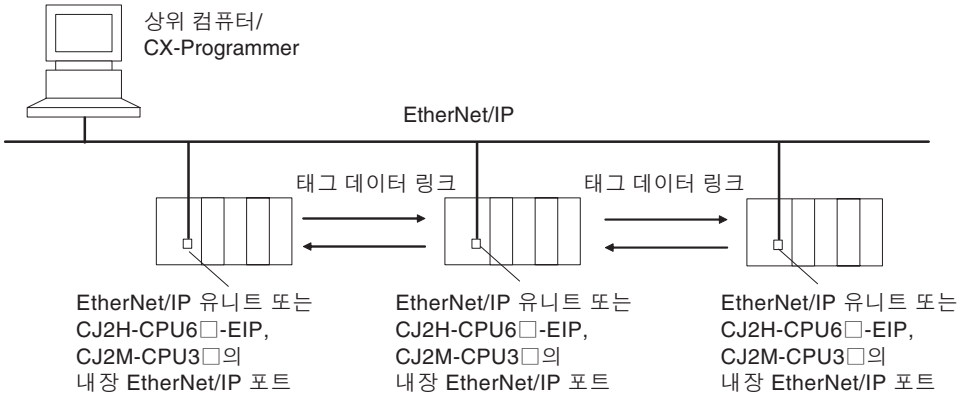


● EtherNet/IP, Ethernet

태그 데이터 링크 및 FINS 네트워크 통신을 Ethernet 케이블에서 구현할 수 있습니다.
또한 Ethernet상의 상위 컴퓨터에서 PLC 본체에 대해 FTP 명령을 전송하면 CPU 유니트에 연결된 메모리 카드의 파일을 읽고 쓰는(전송) 것도 가능합니다. UDP 또는 TCP 프로토콜에서 임의의 데이터를 송수신할 수도 있습니다.
그 결과, 상위 정보 계통 네트워크와의 친화성이 향상됩니다.

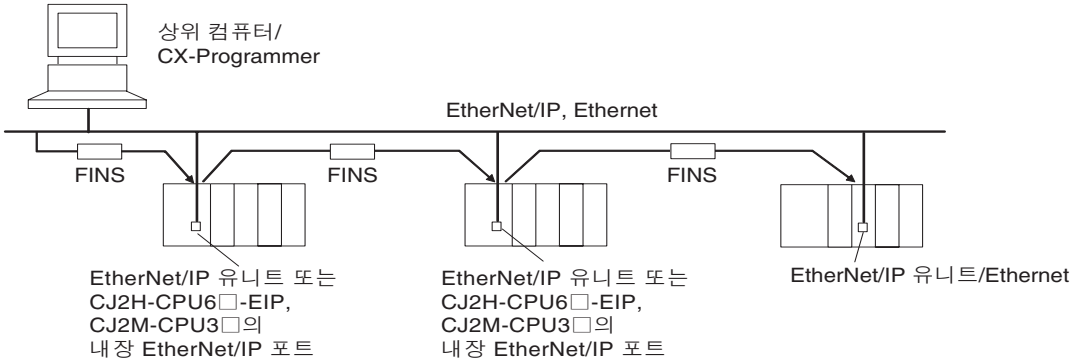
태그 데이터 링크

고속, 대용량, 영역별 임의의 갱신 주기로 PLC－PLC 간의 데이터 링크를 실행할 수 있습니다.



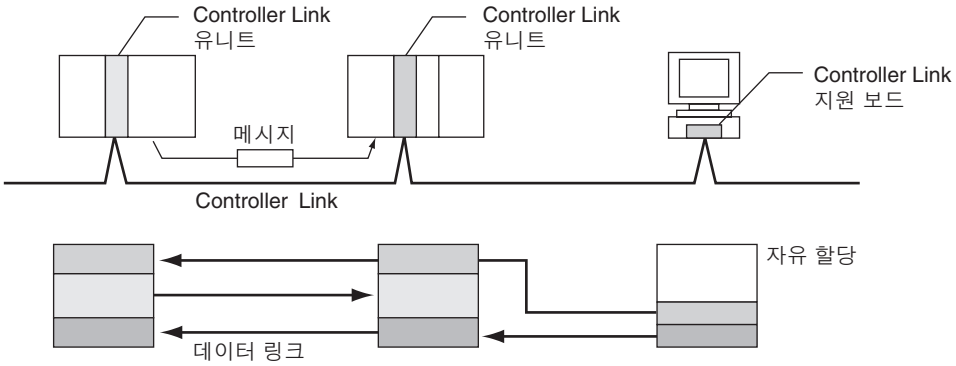
FINS 네트워크 통신

내장 EtherNet/IP 포트 또는 EtherNet/IP 유니트/Ethernet 유니트를 장착한 CPU 유니트와 연결하면 Ethernet상의 상위 컴퓨터와 PLC 간 또는 PLC－PLC 간에 메시지(FINS) 통신이 가능합니다.



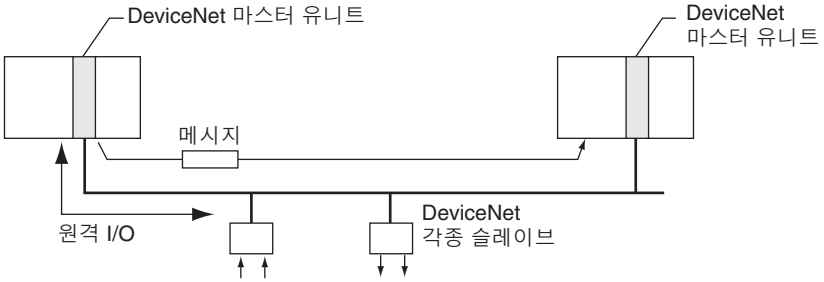
● Controller Link

오므론 PLC의 전용 FA 네트워크입니다. Controller Link 유니트를 연결하면 PLC－PLC 간의 데이터 링크(프로그램 없이 데이터 공유) 및 PLC－PLC 간의 메시지(FINS) 통신(필요 시 데이터 송수신, 각종 제어)을 구현할 수 있습니다. 트위스트 페어 또는 광섬유로 시스템을 구축할 수 있습니다. 그리고 PLC와 PC 간의 데이터 링크, 메시지 통신도 구현할 수 있습니다.
데이터 링크는 대용량으로 자유 할당이 가능합니다. 메시지(FINS) 통신도 대용량 데이터 송수신이 가능합니다.



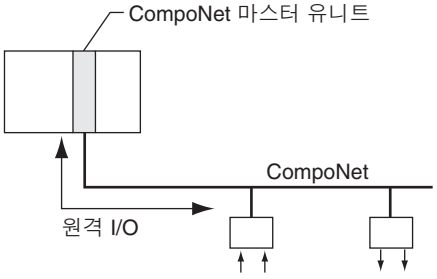
● DeviceNet

제어와 정보가 혼재된 다중 비트 계통의 멀티 벤더 네트워크입니다. 오픈 필드 네트워크 'DeviceNet'에 준거합니다. DeviceNet 마스터 유니트를 장착하면 PLC－ 각종 슬레이브 간의 원격 I/O 통신을 구현할 수 있습니다. 원격 I/O 통신은 대량 점수, 자유 할당이 가능합니다. 슬레이브에 아날로그 입출력 터미널 등이 준비되어 있으므로 PLC－PLC 간 또는 PLC－타사 DeviceNet 기기와의 메시지 통신도 가능합니다.



● CompoNet

고속, 다점, 다중 노드이며, 주로 I/O 제어용 멀티 벤더 네트워크입니다. CompoNet 마스터 유니트를 장착하면 PLC－각종 슬레이브 간의 원격 I/O 통신을 구현할 수 있습니다. 1024점에서 통신 사이클 시간 1ms 이하의 고속 통신을 수행합니다. 최대 2,560점, 최대 384노드를 연결할 수 있습니다.



통신 네트워크의 개요

계통	네트워크 종류	기능	통신	통신 기기
정보 계통	EtherNet/IP	상위 컴퓨터 − PLC 간	메시지(FINS) 통신	• CJ2H-CPU6□-EIP, CJ2M-CPU3□의 내장 EtherNet/IP 포트 • EtherNet/IP 유니트
		PLC − PLC 간		
		상위 컴퓨터 − CPU 유니트 연결 메모리 카드 간	FTP 서버 기능	
	Controller Link	네트워크에 직접 연결된 PC와 PLC 간	메시지(FINS) 통신	Controller Link 지원 보드와 Controller Link 유니트
			데이터 링크(오프셋 지정, 간이 설정 가능)	
	RS-232C→ Controller Link	상위 링크 PC와 네트워크상의 PLC 간	상위 링크 명령 기능 및 게이트웨이 기능	RS-232C 케이블과 Controller Link 유니트
제어 계통	EtherNet/IP	PLC − PLC 간	태그 데이터 링크	• CJ2H-CPU6□-EIP, CJ2M-CPU3□의 내장 EtherNet/IP 포트 • EtherNet/IP 유니트
	Controller Link	PLC − PLC 간	메시지 통신(FINS)	Controller Link 유니트
			데이터 링크 (오프셋 지정, 간이 설정 가능)	
	DeviceNet		오픈 네트워크에서 메시지 통신(FINS)	DeviceNet 마스터 유니트와 설정기
	DeviceNet	PLC − 구성 요소(슬레이브) 기기 간	오픈 네트워크에서 대용량 원격 I/O (고정 또는 자유 할당)	DeviceNet 마스터 유니트와 설정기
	CompoNet		오픈 네트워크에서 고속, 다점, 다중 노드 연결의 원격 I/O	CompoNet 마스터 유니트

12

CPU와 사이클 시간

이 장에서는 CPU 유니트의 내부 처리를 수행하는 사이클 시간의 모니터링 방법과 계산 방법에 대해 설명합니다.

12-1 사이클 시간의 모니터링 12-2
12-1-1 사이클 시간의 모니터링 방법 12-2
12-2 사이클 시간 계산 방법 12-4
12-2-1 CPU 유니트의 동작 흐름 12-4
12-2-2 사이클 시간 계산의 개요 12-5
12-2-3 각 유니트의 I/O 리프레시 시간 12-7
12-2-4 사이클 시간 계산 예 12-10
12-2-5 온라인 편집 시의 사이클 시간 연장 시간 12-11
12-2-6 입출력 응답 시간 12-12
12-2-7 시리얼 PLC 링크의 응답 성능 12-13

12-1 사이클 시간의 모니터링

12-1-1 사이클 시간의 모니터링 방법

CX-Programmer에서는 온라인으로 연결한 상태에서 사이클 시간의 평균값, 최대값, 최소값을 모니터링 할 수 있습니다.

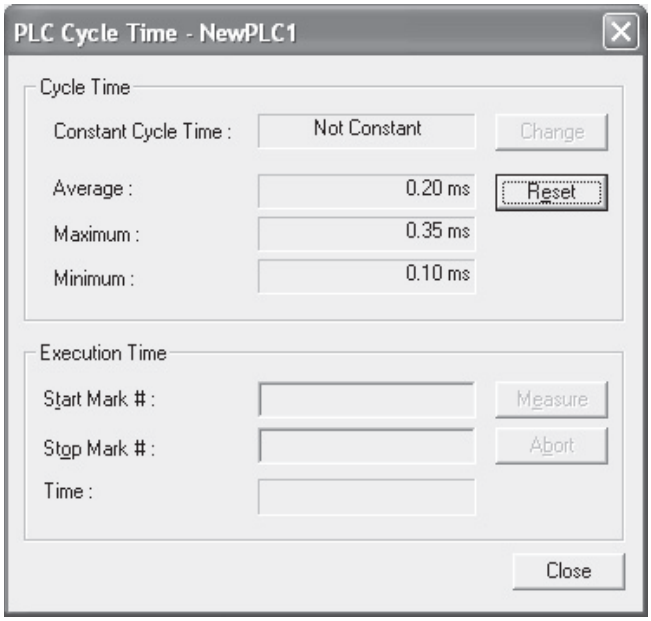
평균값 모니터링

온라인에서 CPU 유니트가 '프로그램' 모드 이외일 때 상태 표시줄에 사이클 시간의 평균값이 표시됩니다.



최대값, 최소값 모니터링

[PLC] 메뉴에서 [편집]－[사이클 시간]을 선택합니다.
다음과 같이 [PLC 사이클 시간] 대화 상자가 표시됩니다.



위부터 사이클 시간의 평균값, 최대값, 최소값이 각각 표시됩니다.
[초기화] 버튼을 클릭하면 사이클 시간이 다시 계측되어 표시됩니다.

참고

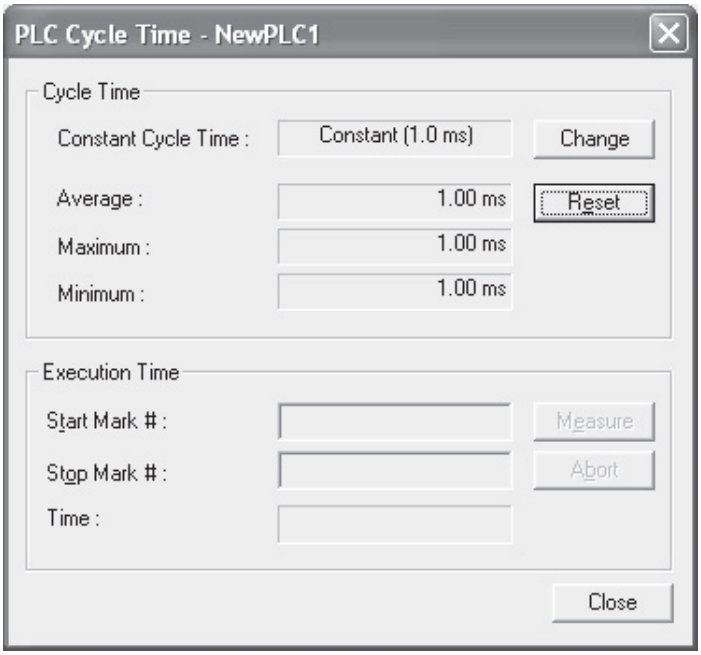
사이클 시간 평균값(현재값), 사이클 시간 최대값은 다음과 같은 특수 보조 릴레이에 저장됩니다.

- 사이클 시간 평균값(현재값) (0.1ms 단위): A264/A265CH(A264: 하위, A265: 상위)
- 사이클 시간 평균값(현재값) (0.01ms 단위): A266/A267CH(A266: 하위, A267: 상위)
- 사이클 시간 최대값(0.01ms 단위): A262/A263CH(A262: 하위, A263: 상위)

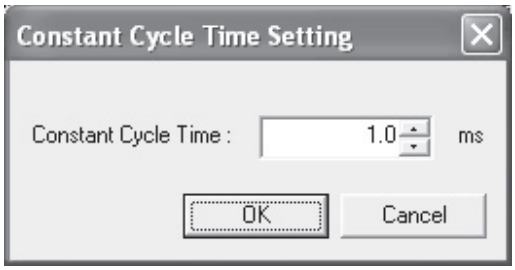
고정 사이클 시간의 변경

사이클 시간 일정화 기능을 사용할 때 PLC의 동작 모드가 모니터 모드이면 고정 사이클 시간을 변경할 수 있습니다.

1 **[PLC] 메뉴에서 [PLC 정보]-[사이클 시간]을 선택합니다.**
다음과 같이 [PLC 사이클 시간] 대화 상자가 표시됩니다.



2 **'변경' 버튼을 클릭합니다.**
[고정 사이클 시간 설정] 대화 상자가 표시됩니다.

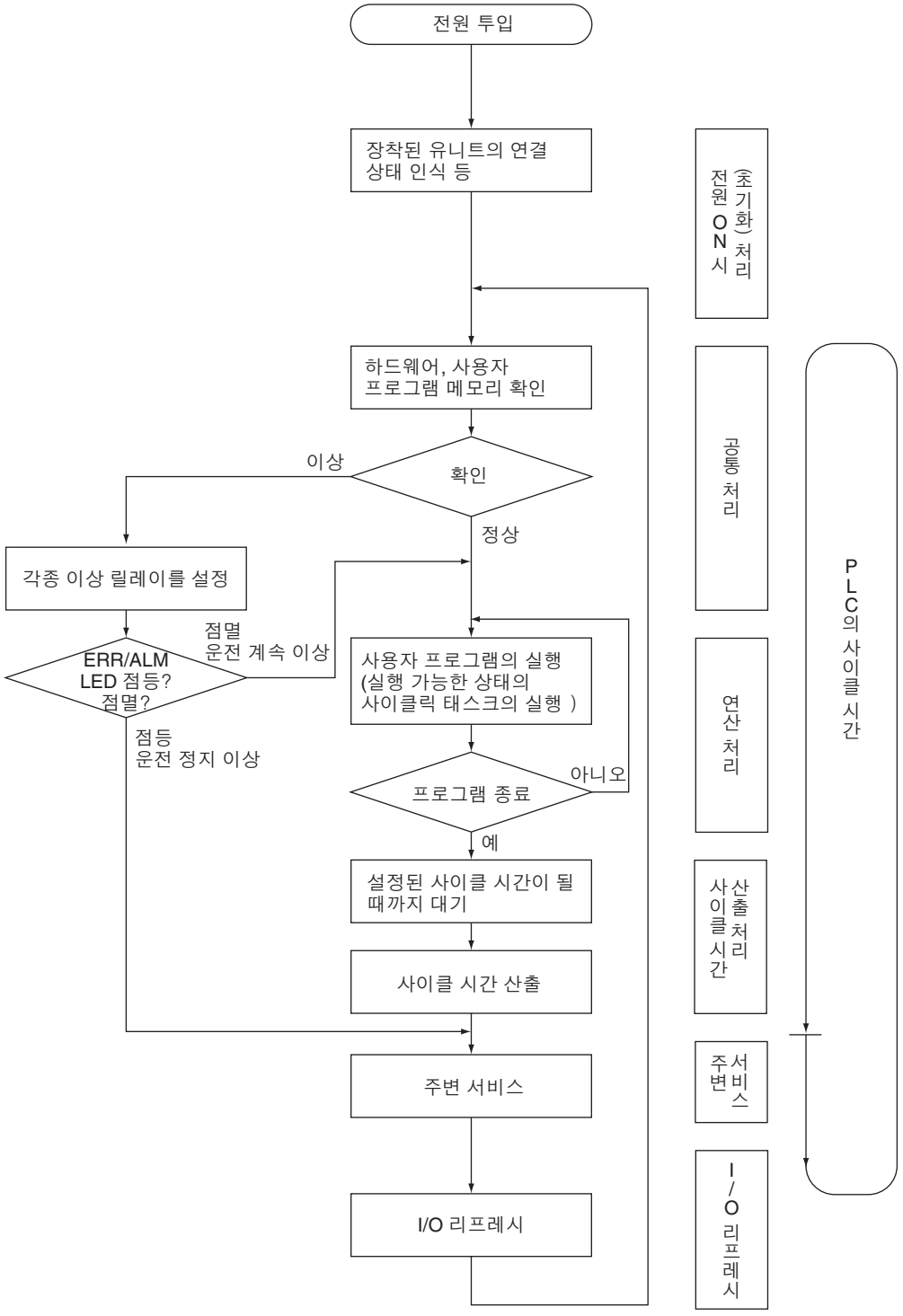


3 **'고정 사이클 시간'의 시간을 변경하고 '확인' 버튼을 클릭합니다.**
고정 사이클 시간이 변경됩니다.

12-2 사이클 시간 계산 방법

12-2-1 CPU 유니트의 동작 흐름

CJ2 CPU 유니트는 공통 처리~주변 서비스까지의 처리를 주기적으로 반복합니다.



12-2-2 사이클 시간 계산의 개요

CJ2 CPU 유니트의 사이클 시간은 다음과 같은 요소에 의해 변화합니다.

- 사용자 프로그램 전체(해당 사이클 내에서 실행 가능한 상태인 모든 사이클릭 태스크/추가 태스크 및 실행 조건이 성립된 인터럽트 태스크 내)의 명령 종류와 수
- 기본 I/O 유니트의 종류와 수
- 고기능 I/O 유니트, CPU 고기능 유니트의 종류와 수
- 다음과 같은 고기능 유니트/보드 고유의 서비스
  - EtherNet/IP의 태그 데이터 링크 리프레시 유무 및 태그 데이터 링크 전송(교환) CH 수
  - Controller Link의 데이터 링크 리프레시 유무 및 데이터 링크 전송(교환) CH 수
  - DeviceNet 원격 I/O 통신의 유무 및 데이터 점유(교환) CH 수
  - 프로토콜 매크로 기능 사용 유무 및 송수신 메시지 내의 송수신 데이터 최대 CH 수
  - Ethernet 유니트의 특정 비트 조작에 의한 소켓 서비스의 유무 및 그에 따른 송수신 데이터 CH 수
- PLC 시스템 설정에 의한 '고정 사이클 시간 설정'
- 파일 메모리에 대한 파일 액세스의 유무, 전송 데이터량
- 고기능 I/O 유니트, CPU 고기능 유니트와의 이벤트 서비스, 통신 포트를 사용한 이벤트 서비스의 종류 및 빈도
- 페리페럴(USB) 포트, 시리얼 포트의 사용 유무
- PLC 시스템 설정에 의한 '주변 서비스 일률 시간 설정'

(주) 1 사용자 프로그램 전체를 구성하는 태스크 수는 사이클 시간에 영향을 주지 않습니다. 사이클릭 태스크 내의 해당 사이클 내에서 실행 가능한 상태인 태스크가 영향을 줍니다.

2 동작 모드를 '모니터' 모드에서 '운전' 모드로 전환한 경우 사이클 시간은 10ms 정도 연장됩니다. (단, 사이클 시간을 초과하지는 않습니다.)

3 Ethernet/IP의 태그 데이터 링크의 데이터 처리는 CJ2H-CPU6□-EIP의 내장 EtherNet/IP 포트와 CJ2M-CPU3□의 내장 EtherNet/IP 포트에서 성능이 크게 달라집니다. 자세한 내용은 "EtherNet/IP 유니트 사용 설명서"(SBCD-342)의 '10-4 CJ2M 내장 EtherNet/IP 포트의 태그 데이터 링크 성능'을 참조하십시오.

사이클 시간은 다음과 같이 각 처리의 처리 시간의 합계가 됩니다.
사이클 시간 = (1) + (2) + (3) + (4) + (5)

(1) 공통 처리

처리 내용	처리 시간과 변동 요인
I/O 버스 확인 사용자 프로그램 메모리 확인, 배터리 이상 확인 등	CJ2H-CPU6□-EIP: 0.2ms CJ2H-CPU6□: 0.1ms CJ2M-CPU1□: 0.16ms CJ2M-CPU3□: 0.27ms

(2) 연산 처리

처리 내용	처리 시간과 변동 요인
사용자 프로그램(명령 실행)을 처리합니다. 명령어 실행 시간의 합계입니다.	명령 실행 시간의 합계

(3) 사이클 시간 일정화, 산출 처리

처리 내용	처리 시간과 변동 요인
PLC 시스템 설정에서 사이클 시간 일정화를 지정하면 사이클 시간이 될 때까지 기다립니다. 사이클 시간을 산출합니다.	−일정화 처리를 하지 않을 때의 시간은 0에 가깝습니다. −일정화 처리를 위한 추가 시간 = 일정화 지정 사이클 시간 −실제 사이클 시간((1)+(2)+(4)+(5))

(4) I/O 리프레시

처리 내용			처리 시간과 변동 요인
기본 I/O 유니트	기본 I/O 유니트의 리프레시 처리를 수행합니다. 유니트별로 OUT(CPU 유니트→유니트)한 뒤에 IN (유니트→CPU 유니트) 순서로 리프레시됩니다.		각 유니트의 I/O 리프레시 시간×사용 대수
고기능 I/O 유니트	할당 릴레이 영역의 리프레시		각 유니트의 I/O 리프레시 시간×사용 대수
	고기능 I/O 유니트 고유의 리프레시	예) CompoBus/S 원격 I/O 통신 처리	
CPU 고기능 유니트	할당 릴레이 영역 및 할당 DM 영역의 리프레시		각 유니트의 I/O 리프레시 시간×사용 대수
	CPU 고기능 유니트 고유의 리프레시	예) • EtherNet/IP의 태그 데이터 링크 리프레시 처리 • Controller Link의 데이터 링크 리프레시 처리, • DeviceNet 원격 I/O 통신 처리, • 프로토콜 매크로의 송수신 데이터 리프레시 처리, • 이더넷의 특정 비트 조작에 의한 소켓 서비스 처리	

(5) 주변 서비스

주변 서비스에는 다음과 같은 서비스가 있습니다.

내부 논리 포트를 사용한 서비스

래더 프로그램 내에서 통신 명령이나 백그라운드 처리, 간이 백업 기능 등을 사용했을 때 동작하는 처리입니다.

고기능 I/O 유니트/CPU 고기능 유니트와의 이벤트 서비스

유니트가 장착되면 동작합니다. 또한 통신 유니트를 경유하여 주변 툴을 사용하는 경우 등에도 동작하는 처리입니다.

페리페럴(USB) 포트 서비스

페리페럴(USB) 포트를 사용할 때 동작하는 처리입니다.

시리얼 포트 서비스

시리얼 포트를 사용할 때 동작하는 처리입니다.

파일 액세스 서비스

메모리 카드 또는 EM 파일 메모리에 액세스하는 경우 등에 동작합니다.

온라인 편집 서비스

온라인 편집을 실행할 때 동작하는 처리입니다.

백업 서비스

프로그램이나 파라미터 변경이 발생했을 때 동작하는 처리입니다.

주변 서비스를 실행하지 않을 때는 처리 시간이 걸리지 않습니다.
주변 서비스 처리 시간의 최소값과 최대값은 PLC 시스템 설정의 '주변 서비스 시간 일률 설정'의 내용에 따라 다음과 같습니다.

		주변 서비스 처리 시간	
		최소	최대
주변 서비스 시간 일률 설정의 임의 설정	없음	0ms (서비스를 실행하지 않은 경우)	사이클 시간의 10% 단, 사이클 시간의 10%가 0.1ms 이하인 경우는 0.1ms
	있음		설정 시간(0.1~3276.7ms)

12-2-3 각 유니트의 I/O 리프레시 시간

● 대표적인 기본 I/O 유니트의 리프레시 시간

유니트 이름	형식	1유니트당 I/O 리프레시 시간	
		CJ2H	CJ2M
DC 입력 유니트(8/16점)	CJ1W-ID201/211/212	0.0014ms	0.0039ms
DC 입력 유니트(32점)	CJ1W-ID231/232/233	0.0023ms	0.0054ms
DC 입력 유니트(64점)	CJ1W-ID261/262	0.0041ms	0.0089ms
AC 입력 유니트(8점/16점)	CJ1W-IA201/111	0.0014ms	0.0039ms
인터럽트 입력 유니트(16점)	CJ1W-INT01	0.0014ms	0.0039ms
펄스 캐치 입력 유니트(16점)	CJ1W-IDP01	0.0014ms	0.0039ms
릴레이 접점 출력 유니트(8점/16점)	CJ1W-OC201/211	0.0014ms	0.0039ms
트라이액 출력 유니트(8점)	CJ1W-OA201	0.0014ms	0.0039ms
트랜지스터 출력 유니트(8점/16점)	CJ1W-OD201/202/203/204/211/212/213	0.0014ms	0.0039ms
트랜지스터 출력 유니트(32점)	CJ1W-OD231/232/233/234	0.0023ms	0.0054ms
트랜지스터 출력 유니트(64점)	CJ1W-OD261/262/263	0.0041ms	0.0089ms
DC 24V 입력/트랜지스터 출력 유니트 (16점 입력/16점 출력)	CJ1W-MD231/232/233	0.0023ms	0.0054ms
DC 24V 입력/트랜지스터 출력 유니트 (32점 입력/32점 출력)	CJ1W-MD261/263	0.0041ms	0.0089ms
TTL 입출력 유니트(16점 입력/16점 출력)	CJ1W-MD563	0.0041ms	0.0089ms
B7A 인터페이스 유니트(64점 입력)	CJ1W-B7A14	0.0041ms	0.0089ms
B7A 인터페이스 유니트(64점 출력)	CJ1W-B7A04	0.0041ms	0.0089ms
B7A 인터페이스 유니트(32점 입력/32점 출력)	CJ1W-B7A22	0.0041ms	0.0089ms

● 대표적인 고기능 I/O 유니트의 리프레시 시간

유니트 이름	형식		1유니트당 I/O 리프레시 시간	
			CJ2H	CJ2M
아날로그 입력 유니트	CJ1W-AD041/081(V1)		0.05ms	0.07ms
	CJ1W-AD042		0.05ms	0.07ms
아날로그 출력 유니트	CJ1W-DA021/041/08V		0.05ms	0.07ms
	CJ1W-DA042V		0.05ms	0.07ms
아날로그 입출력 유니트	CJ1W-MAD42		0.05ms	0.07ms
온도 조절 유니트	CJ1W-TC□□□		0.05ms	0.25ms
위치 제어 유니트	CJ1W-NC113/133		0.12ms	0.13ms
			IOWR/IORD 명령을 데이터 전송에 사용할 때 1개당+0.8ms	
	CJ1W-NC213/233		0.14ms	0.15ms
			IOWR/IORD 명령을 데이터 전송에 사용할 때 1개당+0.8ms	
	CJ1W-NC413/433		0.17ms	0.18ms
			IOWR/IORD 명령을 데이터 전송에 사용할 때 1개당+0.7ms	
	CJ1W-NC214/234 (2축 2태스크 사용 시)		0.20ms	0.355ms
			IOWR/IORD 명령을 데이터 전송에 사용할 때 1개당+0.1ms	
	CJ1W-NC414/434 (4축 4태스크 사용 시)		0.30ms	0.511ms
			IOWR/IORD 명령을 데이터 전송에 사용할 때 1개당+0.1ms	
ID 센서 유니트	CJ1W-V600C11		0.12ms	0.13ms
	CJ1W-V600C12		0.14ms	0.15ms
고속 카운터 유니트	CJ1W-CT021		0.10ms	0.11ms
CompoNet 마스터 유니트	CJ1W-CRM21	통신 모드 번호 0	0.13ms	0.14ms
		통신 모드 번호 1	0.14ms	0.15ms
		통신 모드 번호 2	0.17ms	0.17ms
		통신 모드 번호 3	0.21ms	0.22ms
		통신 모드 번호 8(*)	0.081+(0.0012× 점유 CH 수)ms	0.083+(0.0009× 점유 CH 수)ms
CompoBus/S 마스터 유니트	CJ1W-SRM21	1호기 점유 시	0.08ms	0.08ms
		2호기 점유 시	0.08ms	0.10ms

* 점유 CH 수는 연결된 모든 슬레이브가 실제로 점유하는 I/O 메모리 영역의 CH 수(워드 수)입니다.

● CPU 고기능 유니트의 사이클 시간에 대한 영향 시간

유니트 이름	형식	영향 시간		비고
		CJ2H	CJ2M	
Controller Link 유니트	CJ1W-CLK21(V1)	0.1ms	0.2ms	데이터 링크를 실행하는 경우는 다시 다음 값이 가산됩니다. • CJ2H CPU 유니트의 경우: 0.1ms+전송 CH 수×0.7μs • CJ2M CPU 유니트의 경우: 0.1ms+전송 CH 수×1.2μs 메시지 서비스를 실행하는 경우는 이벤트 실행 시간이 별도로 다시 가산됩니다.
시리얼 커뮤니케이션 유니트	CJ1W-SCU41-V1 CJ1W-SCU31-V1 CJ1W-SCU21-V1 CJ1W-SCU42 CJ1W-SCU32 CJ1W-SCU22	0.20ms		프로토콜 매크로를 실행하는 경우는 최대에서 다음 값이 다시 가산됩니다. • CJ2H CPU 유니트의 경우: 0.4μs×데이터 송수신 최대 CH 수(0~500CH) • CJ2M CPU 유니트의 경우: 0.6μs×데이터 송수신 최대 CH 수(0~500CH) 상위 링크, NT 링크(1 : N 모드)를 실행하는 경우는 이벤트 실행 시간이 별도로 다시 가산됩니다.
Ethernet 유니트	CJ1W-ETN11/21	0.1ms	0.9ms	특정 비트 조작에 의한 소켓 서비스를 실행하는 경우는 다시 다음 값이 가산됩니다. • CJ2H CPU 유니트의 경우: 1.4μs×(송신 바이트 수 또는 수신 바이트 수) • CJ2M CPU 유니트의 경우: 0.17μs×(송신 바이트 수 또는 수신 바이트 수) FINS 통신 서비스, CMND 명령에 의한 소켓 서비스, FTP 서비스를 실행하는 경우는 이벤트 실행 시간이 별도로 다시 가산됩니다.
EtherNet/IP 유니트	CJ1W-EIP21	0.1ms		태그 데이터 링크를 실행하는 경우는 다시 다음 값이 가산됩니다. • CJ2H CPU 유니트의 경우: 0.1ms＋전송 CH 수×0.33μs(*) • CJ2M CPU 유니트의 경우: 0.1ms＋전송 CH 수×0.70μs 메시지 서비스를 실행하는 경우는 이벤트 실행 시간이 별도로 다시 가산됩니다.
FL-net 유니트	CJ1W-FLN22	0.25ms+ 데이터 링크 CH 수×1.6μs	0.25ms+ 데이터 링크 CH 수×1.7μs	'데이터 링크 CH 수'는 해당 노드가 송수신하는 데이터 CH 수입니다. 메시지 서비스를 실행하는 경우는 이벤트 실행 시간이 별도로 가산됩니다.
DeviceNet 유니트	CJ1W-DRM21	0.4ms+0.7μs ×점유 CH 수	0.4ms+0.4μs ×점유 CH 수	점유 CH 수는 모든 슬레이브가 점유하는 I/O 영역의 CH 수입니다. 도중에 있는 빈 영역도 포함됩니다. 메시지 통신을 수행하는 경우는 메시지 처리가 되었을 때만 왼쪽과 같은 점유 CH 수에 메시지 통신 CH 수를 가산해서 계산한 시간이 됩니다.
위치 제어 유니트 (EtherCAT 통신 호환 타입)	CJ1W-NC881 (8축 4 태스크 사용 시)	0.3ms	0.7ms	—
	CJ1W-NC481 (4축 4 태스크 사용 시	0.3ms	0.6ms	
	CJ1W-NC281 (2축 2 태스크 사용 시)	0.2ms	0.5ms	
위치 제어 유니트 (MECHATROLINK-II 통신 호환 타입)	CJ1W-NCF71	위치 제어 유니트에 연결한 서보 드라이버의 축 수에 의해 증가합니다. 축 수 1개: 0.2ms, 축 수 3개: 0.3ms, 축 수 6개: 0.4ms, 축 수 16개: 0.8ms		—
모션 컨트롤 유니트 (MECHATROLINK-II 통신 호환 타입)	CJ1W-MCH71	0.2＋(모션 태스크 수×4＋축 수×5＋범용 입출력의 할당 CH 수)×0.001(ms)	0.2＋(모션 태스크 수×6＋축 수×1＋범용 입출력의 할당 CH 수)×0.001(ms)	—

유니트 이름	형식	영향 시간		비고
		CJ2H	CJ2M	
SYSMAC SPU 유니트	CJ1W-SPU01-V2	0.2ms+샘플링 CH 수×0.8μs	0.2ms+샘플링 CH 수×1.1μs	—

* CJ2H CPU 유니트의 유니트 Ver.1.1 이상에서 고속 인터럽트 기능을 사용하면 다음과 같이 값이 증가합니다.
0.1ms+전송 CH 수×0.87μs

12-2-4 사이클 시간 계산 예

● (예 1) 기본 명령과 기본 I/O 유니트를 중심으로 한 활용 사례

CJ2H-CPU6□-EIP를 사용하고 기본 명령/전송 명령으로 이루어진 프로그램이 20K 스텝에서 기본 I/O 유니트만 장착된 경우의 사이클 시간의 계산 방법은 다음과 같습니다.

(조건)

항목	내용	
CPU 유니트	CJ2H-CPU6□-EIP	
CJ 시리즈 CPU 장치	64점 입력 유니트 CJ1W-ID261	2대
	64점 출력 유니트 CJ1W-OD261	2대
사용자 프로그램	20K 스텝	LD 명령 10K 스텝, MOV 명령 10K 스텝 (*)
페리페럴(USB) 포트 연결	있음/없음	
사이클 시간 일정화 처리	없음	
시리얼 포트 연결	없음	
다른 주변 서비스 (고기능 I/O, CPU 고기능, 파일 액세스)	없음	

* LD 명령은 1명령=1스텝
MOV 명령은 1명령=3스텝

(계산 예)

처리 이름	계산식	처리 시간	
		주변 툴을 페리페럴(USB) 포트에 연결한 경우	주변 툴을 페리페럴(USB) 포트에 연결하지 않은 경우
(1) 공통 처리	—	0.20ms	0.20ms
(2) 연산 처리	0.016μs×10000+0.14μs/3스텝×10000	0.63ms	0.63ms
(3) 사이클 시간 일정화, 산출 처리	(사이클 시간 일정화 처리 없음)	0ms	0ms
(4) I/O 리프레시	0.0039ms×2+0.0039ms×2	0.0164ms	0.0164ms
(5) 주변 서비스	(페리페럴(USB) 포트만 연결)	0.1ms	0ms
사이클 시간	(1)+(2)+(3)+(4)+(5)	0.95ms	0.85ms

● (예 2) 연산 명령과 고기능 I/O 유니트를 포함한 활용 사례

CJ2H-CPU6□-EIP를 사용하고 기본 명령/전송 명령/부동 소수점 연산 명령으로 이루어진 프로그램이 20K 스텝에서 기본 I/O 유니트, 고기능 I/O 유니트를 장착한 경우의 사이클 시간 계산 방법은 다음과 같습니다.

(조건)

항목	내용	
CPU 유니트	CJ2H-CPU6□-EIP	
CJ 시리즈 CPU 장치	64점 입력 유니트 CJ1W-ID261	2대
	64점 출력 유니트 CJ1W-OD261	2대
	아날로그 입력 유니트 CJ1W-AD081	2대
	위치 제어 유니트 CJ1W-NC413	2대
사용자 프로그램	20K 스텝	LD 명령 12K 스텝, MOV 명령 6K 스텝 +F 명령 2K 스텝(*)
페리페럴(USB) 포트 연결	있음/없음	
사이클 시간 일정화 처리	없음	
시리얼 포트 연결	없음	
다른 주변 서비스 (고기능 I/O, CPU 고기능, 파일 액세스)	없음	

* LD 명령은 1명령=1스텝
MOV 명령은 1명령=3스텝
+F 명령은 1명령=4스텝

(계산 예)

처리 이름	계산식	처리 시간	
		주변 툴을 페리페럴(USB) 포트에 연결한 경우	주변 툴을 페리페럴(USB) 포트에 연결하지 않은 경우
(1) 공통 처리	—	0.20ms	0.20ms
(2) 연산 처리	0.016μs×12000+0.14μs/3스텝×6000+0.24μs/4스텝×2000	0.59ms	0.59ms
(3) 사이클 시간 일정화, 산출 처리	(사이클 시간 일정화 처리 없음)	0ms	0ms
(4) I/O 리프레시	0.0041ms×2+0.0041ms×2+0.05ms×2+0.17ms×2	0.4564ms	0.4564ms
(5) 주변 서비스	(페리페럴(USB) 포트만 연결)	0.1ms	0ms
사이클 시간	(1)+(2)+(3)+(4)+(5)	1.35ms	1.25ms

12-2-5 온라인 편집 시의 사이클 시간 연장 시간

'모니터' 모드인 경우 CX-Programmer에서 온라인으로 편집할 수 있습니다. 이때, CPU 유니트의 프로그램 쓰기 처리에 의해 사이클 시간이 약 1ms 연장됩니다.
특정 타이밍에 사이클 시간을 연장하고 싶지 않은 경우는 온라인 편집 금지(대기) 기능을 사용하십시오.
자세한 내용은 '10-7-4 온라인 편집'을 참조하십시오.

12-2-6 입출력 응답 시간

입력 유니트에 연결된 입력 접점이 ON이 되고 해당 데이터를 CJ 시리즈 CPU 유니트가 인식해서 사용자 프로그램을 실행한 후, 결과를 출력 유니트의 출력 접점에 출력할 때까지의 시간을 입출력 응답 시간이라고 합니다.

이 시간은 다음과 같은 조건에 의해 변화합니다.

- 입력 접점이 ON으로 되는 타이밍
- 사이클 시간
- 입력 유니트 및 출력 유니트의 연결 장치(CPU 장치/증설 장치, 증설 장치의 경우는 그 종류)

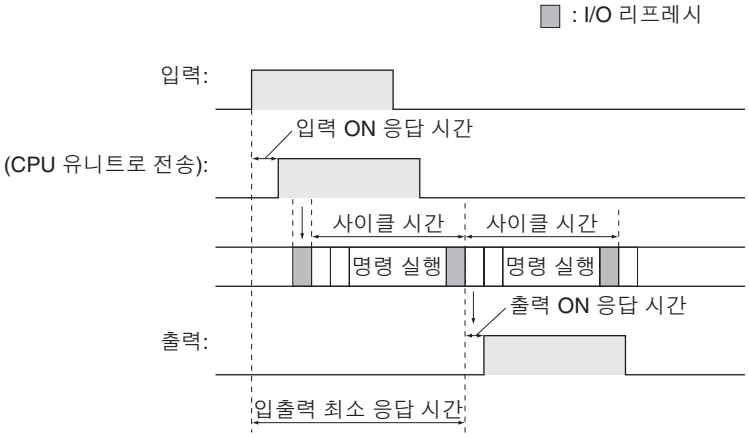
CJ 시리즈 기본 I/O 유니트의 입출력 응답 시간

● 입출력 최소 응답 시간

CPU 유니트가 I/O 리프레시 직전에 데이터를 전송하는 경우, 입출력 응답 시간이 최소가 됩니다. 이 경우의 응답 시간은 다음과 같습니다.

입출력 최소 응답 시간 = 입력 ON 응답 시간 + 사이클 시간 + 출력 ON 응답 시간

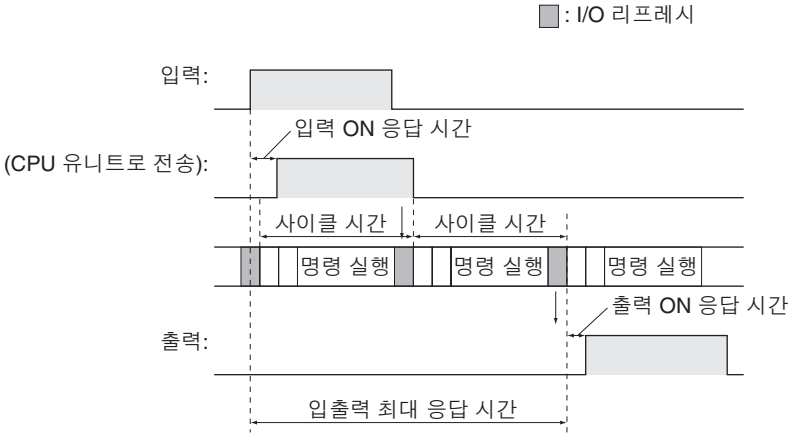
(주) 입력 ON 응답 시간 및 출력 응답 시간은 유니트 형식에 따라 달라집니다.



● 입출력 최대 응답 시간

입력 유니트가 I/O 리프레시 직후에 데이터를 전송하는 경우, 입출력 응답 시간이 최대가 됩니다. 이 경우의 응답 시간은 다음과 같습니다.

입출력 최대 응답 시간 = 입력 ON 응답 시간 + (사이클 시간 × 2) + 출력 ON 응답 시간



● **계산 예**

조건: 입력 ON 응답 시간 1.5ms
출력 ON 응답 시간 0.2ms
사이클 시간 20ms
입출력 최소 응답 시간 = 1.5ms + 20ms + 0.2ms = 21.7ms
입출력 최대 응답 시간 = 1.5ms + (20ms × 2) + 0.2ms = 41.7ms

12-2-7 시리얼 PLC 링크의 응답 성능

시리얼 PLC 링크로 연결된 CPU 유니트 간(친국→자국 또는 자국→친국)의 입출력 응답 성능은 다음 식으로 구할 수 있습니다.

- 입출력 최대 응답 시간(하드 딜레이 포함 안 함) [단위: ms]
  친국 사이클 시간×2+통신 사이클 시간+자국 사이클 시간×2+자국 통신 시간+4
- 입출력 최소 응답 시간(하드 딜레이 포함 안 함) [단위: ms]
  자국 통신 시간+0.54

가입 자국 대수	친국에 설정한 최대 기기 번호의 범위 내에서 링크가 확립된 자국의 대수
이탈 자국 대수	친국에 설정한 최대 기기 번호의 범위 내에서 링크로부터 이탈한 자국의 대수
통신 사이클 시간 [단위: ms]	• 자국이 2대 이상인 경우 자국 통신 시간×가입 자국 대수+10×이탈 자국 대수 • 자국이 1대인 경우 자국 통신 시간+10ms
자국 통신 시간 [단위: ms]	• 통신 속도가 '표준'인 경우 자국 사이클 시간+0.286×[(자국 대수+1)×링크 CH 수×2+12] • 통신 속도가 '고속'인 경우 자국 사이클 시간+0.0955×[(자국 대수+1)×링크 CH 수×2+12]

부록

부록-1 명령 목록 부록-2

부록-2 명령 처리 시간/스텝 수 목록 부록-75

부록-3 특수 보조 릴레이 목록 부록-96

부록-3-1 읽기 전용 영역 부록-96
부록-3-2 읽기/쓰기 가능 영역 부록-114
부록-3-3 특수 보조 릴레이 보충 설명 부록-121

부록-4 I/O 메모리 실제 주소 부록-124

부록-4-1 I/O 메모리 실제 주소 부록-124
부록-4-2 I/O 메모리 실제 주소 목록 부록-125

부록-5 전원 OFF 시 PLC의 동작 부록-126

부록-5-1 전원 OFF 시의 동작 부록-126
부록-5-2 전원 OFF 시의 명령 실행 부록-128

부록-6 Windows XP(SP2 이상)/Vista에서 EtherNet/IP를 연결하기 전에 부록-130

부록-6-1 Windows 방화벽 설정 변경 순서 부록-130

부록-7 CJ/CS 시리즈 비교 표 부록-132

부록-8 유니트 버전에 따른 지원 기능 목록 부록-136

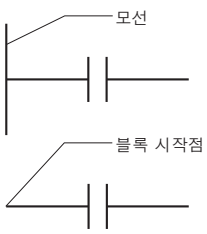
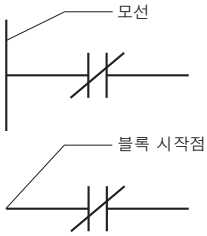
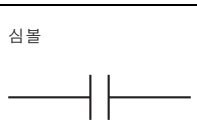
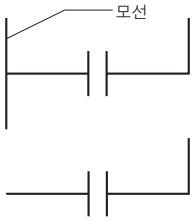
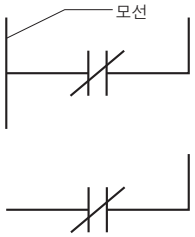
색인 부록-139

부록-1 명령 목록

CJ2 CPU 유니트가 지원하는 명령 목록입니다.
자세한 내용은 "명령 참조 설명서"(SBCA-351)를 참조하십시오.

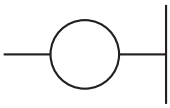
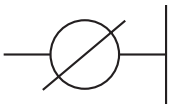
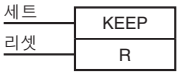
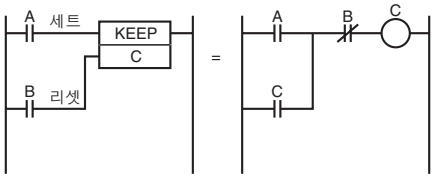
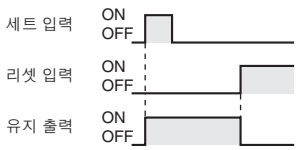
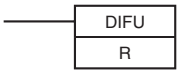
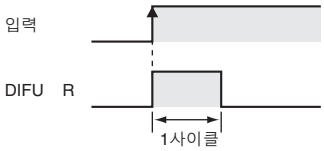
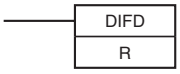
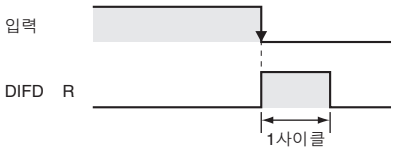
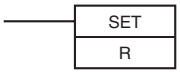
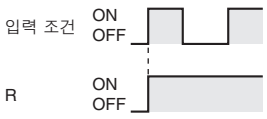
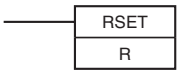
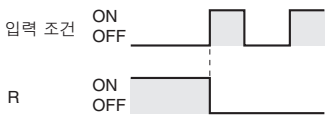
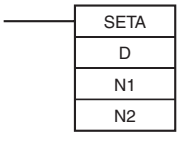
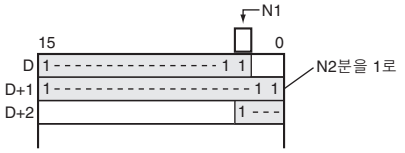
부록-1-1 시퀀스 입력 명령 부록-3
부록-1-2 시퀀스 출력 명령 부록-5
부록-1-3 시퀀스 제어 명령 부록-7
부록-1-4 타이머/카운터 명령 부록-10
부록-1-5 데이터 비교 명령 부록-14
부록-1-6 데이터 전송 명령 부록-17
부록-1-7 데이터 이동 명령 부록-19
부록-1-8 증가/감소 명령 부록-22
부록-1-9 사칙연산 명령 부록-23
부록-1-10 데이터 변환 명령 부록-28
부록-1-11 논리 연산 명령 부록-34
부록-1-12 특수 연산 명령 부록-35
부록-1-13 부동 소수점 변환 및 연산 명령 부록-36
부록-1-14 배정도 부동 소수점 변환 및 연산 명령 부록-40
부록-1-15 테이블 데이터 처리 명령 부록-43
부록-1-16 추적 명령 부록-47
부록-1-17 데이터 제어 명령 부록-49
부록-1-18 서브루틴 명령 부록-52
부록-1-19 인터럽트 제어 명령 부록-53
부록-1-20 공정 진행 제어 명령 부록-54
부록-1-21 I/O 유니트용 명령 부록-55
부록-1-22 시리얼 통신 명령 부록-58
부록-1-23 네트워크 통신용 명령 부록-60
부록-1-24 파일 메모리용 명령 부록-62
부록-1-25 표시 기능용 명령 부록-63
부록-1-26 시계 기능 명령 부록-63
부록-1-27 디버그 처리 명령 부록-64
부록-1-28 고장 진단 명령 부록-64
부록-1-29 특수 명령 부록-65
부록-1-30 블록 프로그램 명령 부록-66
부록-1-31 문자열 처리 명령 부록-70
부록-1-32 태스크 제어 명령 부록-72
부록-1-33 기종 변환용 명령 부록-73
부록-1-34 펑션 블록용 특수 명령 부록-74

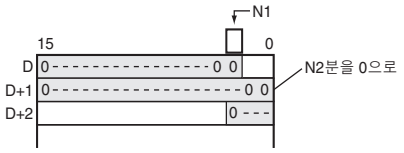
부록-1-1 시퀀스 입력 명령

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
로드	LD @LD %LD !LD !@LD !%LD	–		논리 시작을 나타냅니다. 접점의 ON/OFF 내용을 전송합니다.	논리 시작	필요 없음
로드 부정	LD NOT @LD NOT %LD NOT !LD NOT !@LD NOT !%LD NOT	–		논리 시작을 나타냅니다. 접점의 ON/OFF 내용을 반전하여 전송합니다.	논리 시작	필요 없음
논리곱	AND @AND %AND !AND !@AND !%AND	–		접점의 ON/OFF 내용과 이전 입력 조건의 논리곱을 계산합니다.	다음 단계 연결형	필요
부정 논리곱	AND NOT @AND NOT %AND NOT !AND NOT !@AND NOT !%AND NOT	–		접점의 ON/OFF 내용을 반전하여 이전 입력 조건과의 논리곱을 계산합니다.	다음 단계 연결형	필요
논리합	OR @OR %OR !OR !@OR !%OR	–		접점의 ON/OFF 내용과 이전 입력 조건과의 논리합을 계산합니다.	다음 단계 연결형	필요
부정 논리합	OR NOT @OR NOT %OR NOT !OR NOT !@OR NOT !%OR NOT	–		접점의 ON/OFF 내용을 반전하여 이전 입력 조건과의 논리합을 계산합니다.	다음 단계 연결형	필요
로드 논리곱	AND LD	–		회로 블록 간의 논리곱을 계산합니다. LD ⟆ 회로 블록 A LD ⟆ 회로 블록 B AND LD……회로 블록 A와 회로 블록 B의 직렬 연결	다음 단계 연결형	필요

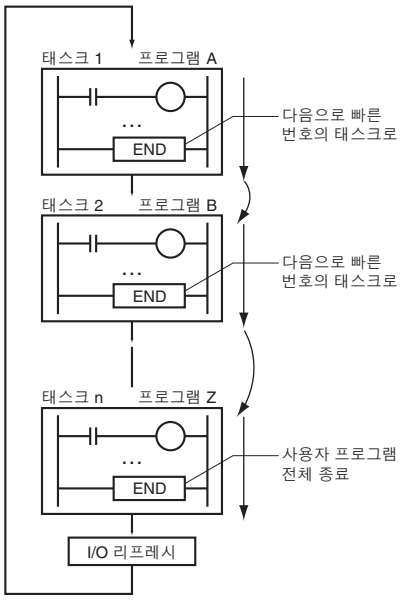
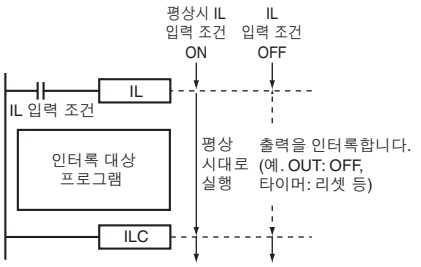
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
로드 논리합	OR LD	–	회로 블록 회로 블록	회로 블록 간의 논리합을 계산합니다. LD 회로 블록 A LD 회로 블록 B OR LD……회로 블록 A와 회로 블록 B의 병렬 연결	다음 단계 연결형	필요
부정	NOT	520		입력 조건을 반전합니다.	다음 단계 연결형	필요
P.F. 기동 미분	UP	521	UP	입력 신호가 기동될 때(OFF→ON) 1사이클 ON하고 다음 단계에 연결합니다.	다음 단계 연결형	필요
P.F. 종료 미분	DOWN	522	DOWN	입력 신호가 종료될 때(ON→OFF) 1사이클 ON하고 다음 단계 이후로 연결합니다.	다음 단계 연결형	필요
LD 비트 테스트	LD TST	350	TST S N S: 테스트 데이터 채널 번호 N: 비트 위치 데이터	지정 비트가 1일 때 다음 단계에 LD 연결합니다.	논리 시작	필요 없음
LD 비트 테스트 부정	LD TSTN	351	TSTN S N S: 테스트 데이터 채널 번호 N: 비트 위치 데이터	지정 비트가 0일 때 다음 단계에 LD 연결합니다.	논리 시작	필요 없음
AND 비트 테스트	AND TST	350	AND TST S N S: 테스트 데이터 채널 번호 N: 비트 위치 데이터	지정 비트가 1일 때 다음 단계에 AND 연결합니다.	다음 단계 연결형	필요
AND 비트 테스트 부정	AND TSTN	351	AND TSTN S N S: 테스트 데이터 채널 번호 N: 비트 위치 데이터	지정 비트가 0일 때 다음 단계에 AND 연결합니다.	다음 단계 연결형	필요
OR 비트 테스트	OR TST	350	TST S N S: 테스트 데이터 채널 번호 N: 비트 위치 데이터	지정 비트가 1일 때 다음 단계에 OR 연결합니다.	다음 단계 연결형	필요
OR 비트 테스트 부정	OR TSTN	351	TSTN S N S: 테스트 데이터 채널 번호 N: 비트 위치 데이터	지정 비트가 0일 때 다음 단계에 OR 연결합니다.	다음 단계 연결형	필요

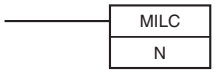
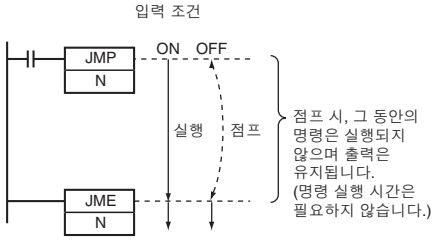
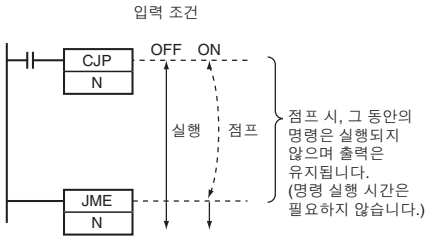
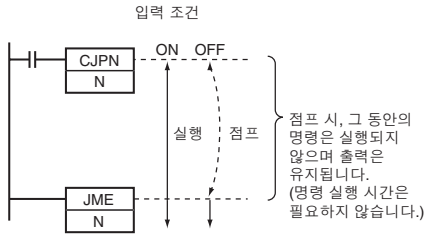
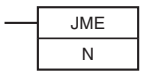
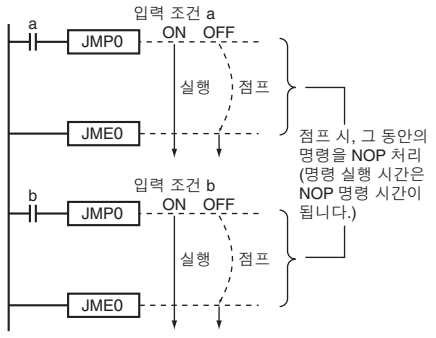
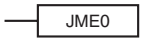
부록-1-2 시퀀스 출력 명령

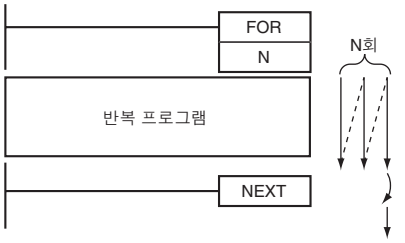
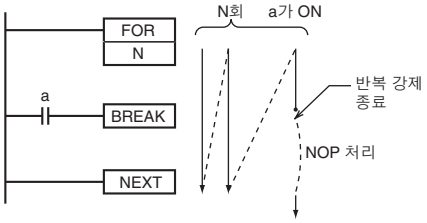
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
출력	OUT !OUT	–		논리 연산 처리 결과를 릴레이 출력합니다.	출력 계통	필요
부정 출력	OUT NOT !OUT NOT	–		논리 연산 처리 결과를 반전하여 릴레이 출력합니다.	출력 계통	필요
유지	KEEP !KEEP	011	 R: 릴레이 번호	래치 릴레이(자체 유지)의 동작을 수행합니다.  	출력 계통	필요
기동 미분	DIFU !DIFU	013	 R: 릴레이 번호	입력 신호가 기동될 때(OFF→ON) 지정한 접점을 1사이클 ON합니다. 	출력 계통	필요
종료 미분	DIFD !DIFD	014	 R: 릴레이 번호	입력 신호가 종료될 때(ON→OFF) 지정한 접점을 1사이클 ON합니다. 	출력 계통	필요
세트	SET @SET %SET !SET !@SET !%SET	–	 R: 릴레이 번호	입력 조건이 ON일 때 지정된 접점을 ON으로 설정합니다. 이후, 입력 조건의 ON/OFF에 관계 없이 지정된 접점은 ON 상태를 그대로 유지합니다. 	출력 계통	필요
리셋	RSET @RSET %RSET !RSET !@RSET !%RSET	–	 R: 릴레이 번호	입력 조건이 ON일 때 지정된 접점을 OFF로 리셋합니다. 이후, 입력 조건의 ON/OFF에 관계 없이 지정된 접점은 OFF 상태를 그대로 유지합니다. 	출력 계통	필요
다중 비트 세트	SETA @SETA	530	 D: 세트 하위 CH 번호 N1: 세트 시작 비트 위치 N2: 비트 수	연속으로 지정된 비트 수 분량의 비트를 ON합니다. 	출력 계통	필요

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
다중 비트 리셋	RSTA @RSTA	531	RSTA D N1 N2 D: 리셋 하위 CH 번호 N1: 리셋 시작 비트 위치 N2: 비트 수	연속으로 지정 비트 수 분량의 비트를 OFF합니다. 	출력 계통	필요
1비트 세트	SETB @SETB !SETB !@SETB	532	SETB D N D: CH 주소 N: 비트 위치	입력 조건이 ON일 때 지정된 CH의 지정된 접점을 ON으로 설정합니다. RST 명령과 달리 DM 또는 EM 영역의 지정 비트를 리셋 대상으로 할 수 있습니다.	1비트 세트	SETB
1비트 리셋	RSTB @RSTB !RSTB !@RSTB	533	RSTB D N D: CH 주소 N: 비트 위치	입력 조건이 ON일 때 지정된 CH의 지정된 접점을 OFF하고 리셋합니다. RST 명령과 달리 DM 또는 EM 영역의 지정 비트를 리셋 대상으로 할 수 있습니다.	출력 계통	필요
1비트 출력	OUTB @OUTB !OUTB	534	OUTB D N D: CH 주소 N: 비트 위치	논리 연산 처리 결과(입력 조건)를 지정된 CH의 지정된 비트에 출력합니다. OUT 명령과 달리 DM 또는 EM 영역의 지정 비트를 대상으로 할 수 있습니다.	출력 계통	필요

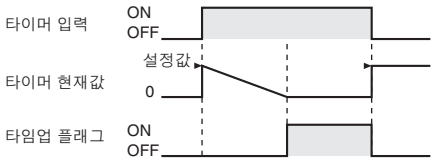
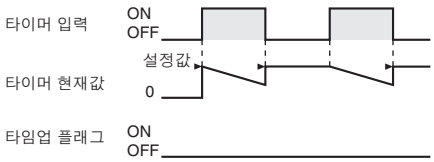
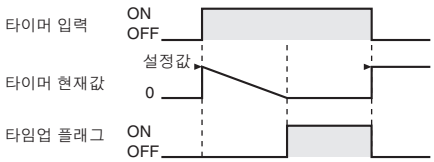
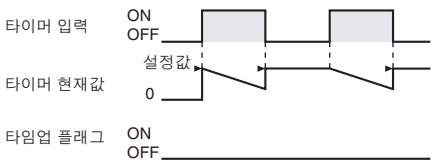
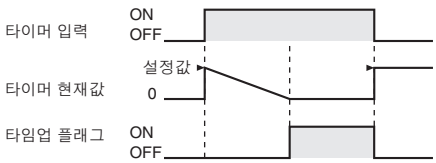
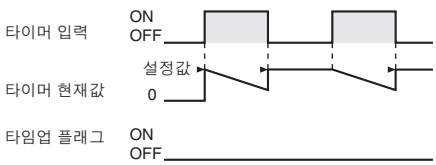
부록-1-3 시퀀스 제어 명령

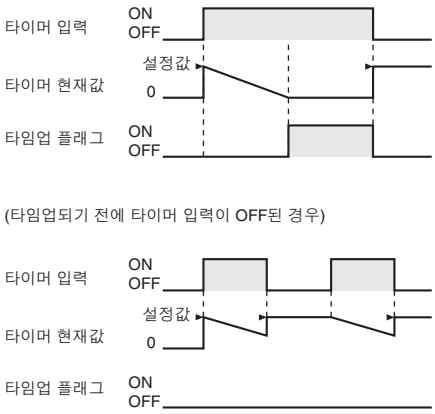
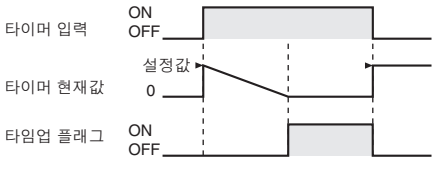
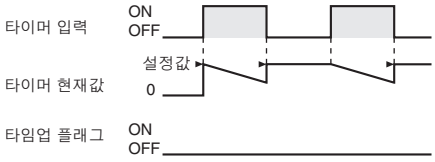
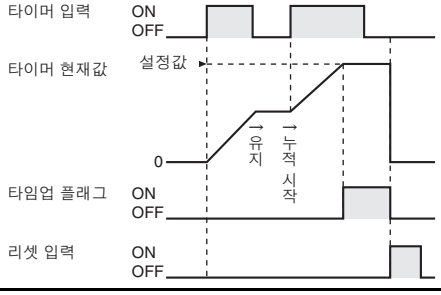
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
종료	END	001	END	한 프로그램의 종료를 나타냅니다. 사용자 프로그램 전체를 실행하는 경우에는 해당 사이클 내에서 기동된 사이클릭 태스크 중 다음으로 빠른 번호의 사이클릭 태스크 프로그램으로 실행 제어를 이동합니다. 이 명령이 기재된 프로그램의 사이클릭 태스크가 기동되어 있는 사이클릭 태스크 중에 가장 큰 태스크 번호인 경우, 이 명령은 사용자 프로그램 전체의 종료를 나타냅니다. 	출력 계통	필요 없음
기능 없음	NOP	000		아무 기능도 없는 명령입니다. 니모닉 표시 시에만 사용됩니다.	출력 계통	필요 없음
인터록	IL	002	IL	입력 조건이 OFF이면 이 명령 이후 ILC 명령까지의 출력을 인터록합니다. 입력 조건이 ON일 때는 ILC 명령까지의 각 명령을 평상시 대로 실행합니다. 	출력 계통	필요
인터록 클리어	ILC	003	ILC	인터록의 범위의 종료를 나타냅니다.	출력 계통	필요 없음
멀티 인터록 (미분 플래그 유지형)	MILH	517	MILH N D N: 인터록 번호 D: 인터록 상태 출력 비트	MILH 명령의 입력 조건이 OFF이면 MILH 명령 이후 MILC 명령까지의 출력을 인터록합니다. MILH 명령과 MILC 명령은 한 쌍으로 사용합니다. MILH/MILC 명령은 중첩할 수 있습니다(예: MILH－MILH－MILC－MILC). 또한 MILH~MILC 명령까지의 각 명령 중에 미분 명령(DIFU/DIFD/@/% 표시가 있는 명령)이 있는 경우, 인터록 시작과 인터록 해제 사이에서 미분 조건이 성립하면 인터록 해제 시에 미분 명령을 실행합니다.	출력 계통	필요
멀티 인터록 (미분 플래그 비유지형)	MILR	518	MILR N D N: 인터록 번호 D: 인터록 상태 출력 비트	MILR 명령의 입력 조건이 OFF이면 MILR 명령 이후 MILC 명령까지의 출력을 인터록합니다. MILR 명령과 MILC 명령은 한 쌍으로 사용합니다. MILR/MILC 명령은 중첩할 수 있습니다(예: MILR－MILR－MILC－MILC). 또한 MILR~MILC 명령까지의 각 명령 중에 미분 명령(DIFU/DIFD/@/% 표시가 있는 명령)이 있는 경우, 인터록 중에 미분 조건이 성립해도 해당 조건의 성립은 취소됩니다.	출력 계통	필요

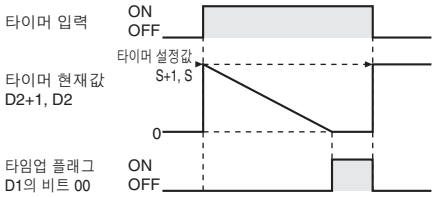
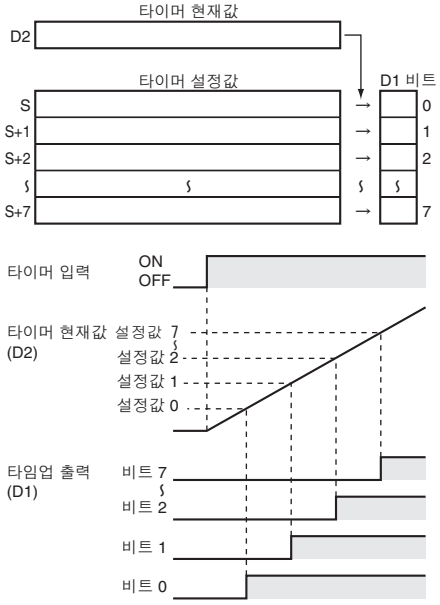
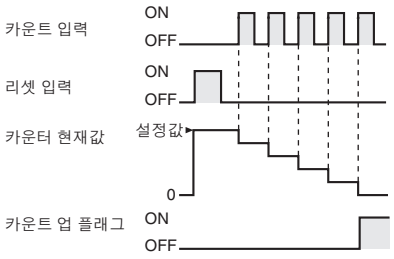
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
멀티 인터록 클리어	MILC	519	 N: 인터록 번호	인터록 번호가 동일한 MILH 또는 MILR 명령에 의한 멀티 인터록 범위의 종료를 나타냅니다.	출력 계통	필요 없음
점프	JMP	004	 N: 점프 번호	JMP 명령의 입력 조건이 OFF일 때 JME 명령까지 직접 점프합니다. 입력 조건이 ON이면 다음 명령 이후를 평상시 대로 실행합니다. 	출력 계통	필요
조건부 점프	CJP	510	 N: 점프 번호	점프 조건이 ON일 때 JME 명령까지 직접 점프합니다. 입력 조건이 OFF이면 다음 명령 이후를 평상시 대로 실행합니다. 	출력 계통	필요
조건부 부정 점프	CJPN	511	 N: 점프 번호	점프 조건이 OFF일 때 JME 명령까지 직접 점프합니다. 입력 조건이 ON이면 다음 명령 이후를 평상시 대로 실행합니다. 	출력 계통	필요 없음
점프 종료	JME	005	 N: 점프 번호	JMP 또는 CJP 명령에 의한 점프의 종료 위치를 나타냅니다.	출력 계통	필요 없음
복수 점프	JMP0	515		JMP0 명령의 입력 조건이 OFF일 때 JMP0의 다음 명령에서 JME0 명령까지 NOP 처리합니다. 입력 조건이 ON이면 다음 명령 이후를 평상시 대로 실행합니다. 	출력 계통	필요
복수 점프 종료	JME0	516		JMP0 명령에 의한 점프의 종료 위치를 나타냅니다.	출력 계통	필요 없음

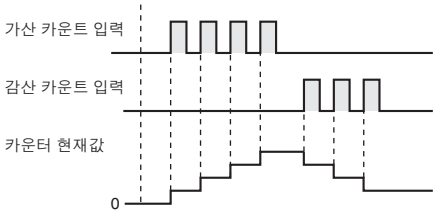
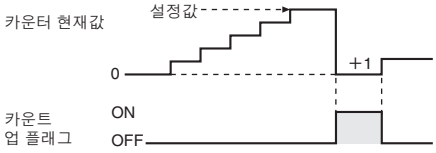
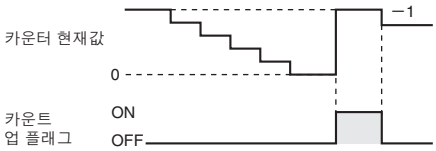
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
반복 시작	FOR	512	FOR N N: 루프 횟수	FOR~NEXT 사이의 프로그램을 무조건 지정된 횟수만큼 반복 실행한 후 NEXT 명령 이후의 명령을 실행합니다. 	출력 계통	필요 없음
루프 중단	BREAK	514	BREAK	FOR-NEXT 루프를 중단하고 NEXT 명령까지 NOP 처리합니다. 	출력 계통	필요
반복 종료	NEXT	513	NEXT	FOR-NEXT 루프의 종료를 나타냅니다.	출력 계통	필요 없음

부록-1-4 타이머/카운터 명령

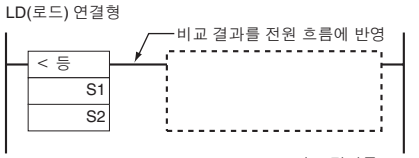
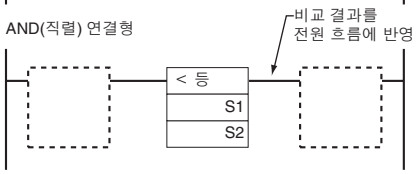
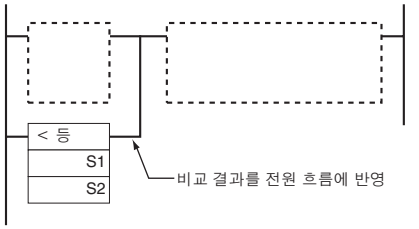
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
100ms 타이머	TIM (BCD 방식)	—	TIM N S N: 타이머 번호 S: 타이머 설정값	감산식 ON 딜레이 0.1초 단위의 타이머 동작을 합니다. 설정값: BCD 방식일 때: 0~999.9초(BCD) BIN 방식일 때: 0~6553.5초(10진수 또는 16진수)  (타임업되기 전에 타이머 입력이 OFF된 경우) 	출력 계통	필요
	TIMX (BIN 방식)	550	TIMX N S N: 타이머 번호 S: 타이머 설정값			
10ms 타이머	TIMH (BCD 방식)	015	TIMH N S N: 타이머 번호 S: 타이머 설정값	감산식 ON 딜레이 10ms(0.01초) 단위의 타이머 동작을 합니다. 설정값: BCD 방식일 때: 0~99.99초(BCD) BIN 방식일 때: 0~655.35초(10진수 또는 16진수)  (타임업되기 전에 타이머 입력이 OFF된 경우) 	출력 계통	필요
	TIMHX (BIN 방식)	551	TIMHX N S N: 타이머 번호 S: 타이머 설정값			
1ms 타이머	TMHH (BCD 방식)	540	TMHH N S N: 타이머 번호 S: 타이머 설정값	감산식 고속 ON 딜레이 1ms 단위의 타이머의 동작을 나타냅니다. 설정값: BCD 방식일 때: 0~9.999초(BCD) BIN 방식일 때: 0~65.535초(10진수 또는 16진수)  (타임업되기 전에 타이머 입력이 OFF된 경우) 	출력 계통	필요
	TMHHX (BIN 방식)	552	TMHHX N S N: 타이머 번호 S: 타이머 설정값			

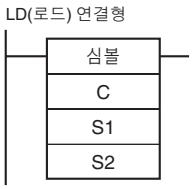
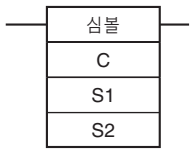
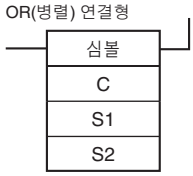
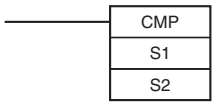
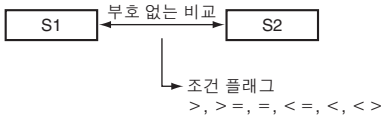
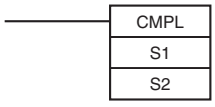
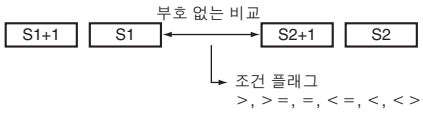
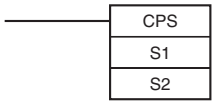
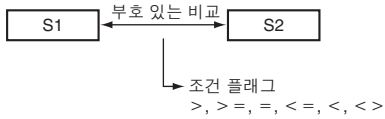
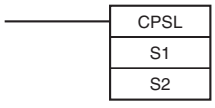
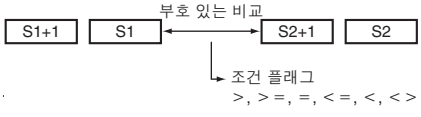
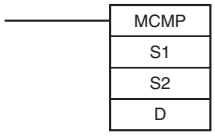
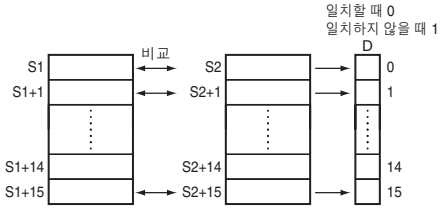
명령 이름	니모닉	평선 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
0.1ms 타이머	TIMU (BCD 방식)	541	TIMU N S N: 타이머 번호 S: 타이머 설정값	감산식 고속 ON 딜레이 0.1ms 단위의 타이머의 동작을 나타냅니다. 설정값: BCD 방식일 때: 0~0.9999초(BCD) BIN 방식일 때: 0~6.5535초(10진수 또는 16진수)  (타임업되기 전에 타이머 입력이 OFF된 경우) 	출력 계통	필요
	TIMUX (BIN 방식)	556	TIMUX N S N: 타이머 번호 S: 타이머 설정값			
0.01ms 타이머	TMUH (BCD 방식)	544	TMUH N S N: 타이머 번호 S: 타이머 설정값	감산식 고속 ON 딜레이 0.01ms 단위의 타이머의 동작을 나타냅니다. 설정값: BCD 방식일 때: 0~0.09999초(BCD) BIN 방식일 때: 0~0.65535초(10진수 또는 16진수)  (타임업되기 전에 타이머 입력이 OFF된 경우) 	출력 계통	필요
	TMUHX (BIN 방식)	556	TMUHX N S N: 타이머 번호 S: 타이머 설정값			
적산 타이머	TTIM (BCD 방식)	087	타이머 입력 TTIM N 리셋 입력 S N: 타이머 번호 S: 타이머 설정값	적산식 ON 딜레이 0.1초 단위의 타이머 동작을 합니다. 설정 시간: BCD 방식일 때: 0~999.9초(BCD) BIN 방식일 때: 0~6553.5초(10진수 또는 16진수) 	출력 계통	필요
	TTIMX (BIN 방식)	555	타이머 입력 TTIMX N 리셋 입력 S N: 타이머 번호 S: 타이머 설정값			

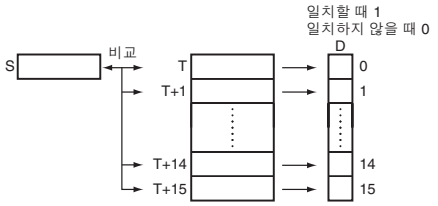
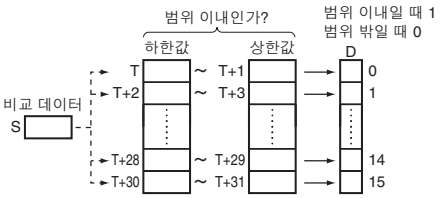
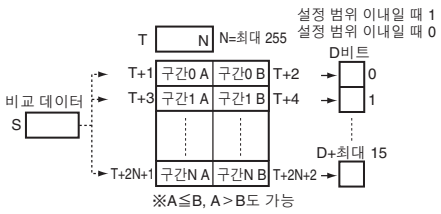
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
장시간 타이머	TIML (BCD 방식)	542	TIML D1 D2 S D1: 타임업 플래그 CH 번호 D2: 현재값 출력 하위 CH 번호 S: 타이머 설정값 하위 CH 번호	감산식 ON 딜레이 100ms 타이머입니다. 설정값(초 단위): BCD 방식일 때: 115일(BCD) BIN 방식일 때: 49710일(10진수 또는 16진수) 	출력 계통	필요
	TIMLX (BIN 방식)	553	TIMLX D1 D2 S D1: 타임업 플래그 CH 번호 D2: 현재값 출력 하위 CH 번호 S: 타이머 설정값 하위 CH 번호			
멀티 출력 타이머	MTIM (BCD 방식)	543	MTIM D1 D2 S D1: 결과 출력 CH 번호 D2: 현재값 출력 CH 번호 S: 설정값 하위 CH 번호	8점의 임의 타임업 플래그 값을 얻을 수 있는 누적식 ON 딜레이 0.1초 단위의 타이머입니다. 설정 시간: BCD 방식일 때: 0~999.9초(BCD) BIN 방식일 때: 0~6553.5초(10진수 또는 16진수) 	출력 계통	필요
	MTIMX (BIN 방식)	554	MTIMX D1 D2 S D1: 결과 출력 CH 번호 D2: 현재값 출력 CH 번호 S: 설정값 하위 CH 번호			
타이머 리셋	TRSET	549	TRSET N N: 타이머 번호	지정한 타이머를 리셋합니다.	출력 계통	필요
카운터	CNT (BCD 방식)	—	카운트 입력 CNT N 리셋 입력 S N: 카운터 번호 S: 카운터 설정값	감산 카운터의 동작을 수행합니다. 설정값: BCD 방식일 때: 0~9999회(BCD) BIN 방식일 때: 0~65535회(10진수 또는 16진수) 	출력 계통	필요
	CNTX (BIN 방식)	546	카운트 입력 CNTX N 리셋 입력 S N: 카운터 번호 S: 카운터 설정값			

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
가역 카운터	CNTR (BCD 방식)	012	가산 카운트 감산 카운트 리셋 입력 CNTR N S N: 카운터 번호 S: 카운터 설정값	가감산 카운터 동작을 합니다. 가산 카운트 입력의 기동으로 가산을 실행하고 감산 카운트 입력의 기동으로 감산을 실행합니다.  카운트업 플래그의 동작 ・가산을 사용할 때  ・감산을 사용할 때 	출력 계통	필요
	CNTRX (BIN 방식)	548	가산 카운트 감산 카운트 리셋 입력 CNTRX N S N: 카운터 번호 S: 카운터 설정값			
타이머/카운터 리셋	CNR @CNR (BCD 방식)	545	CNR D1 D2 D1: 타이머/카운터 번호 1 D2: 타이머/카운터 번호 2	지정 범위의 타이머/카운터의 업 플래그를 리셋합니다. 동시에, 현재값을 최대값(BCD 방식일 때: #9999, BIN 방식일 때: #FFFF)으로 설정합니다.	출력 계통	필요
	CNRX @CNRX (BIN 방식)	547	CNRX D1 D2 D1: 타이머/카운터 번호 1 D2: 타이머/카운터 번호 2			

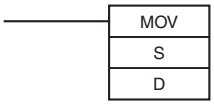
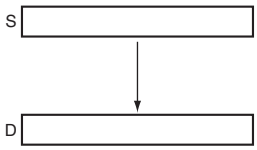
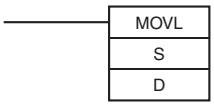
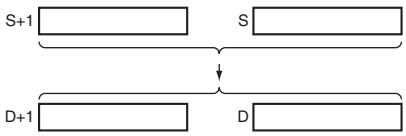
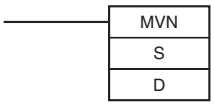
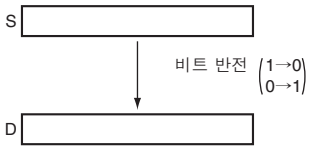
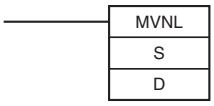
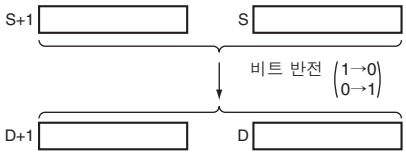
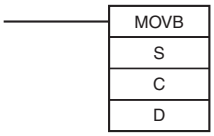
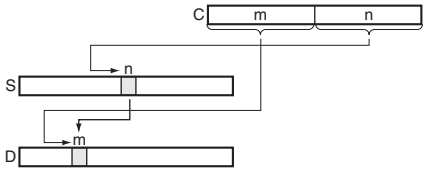
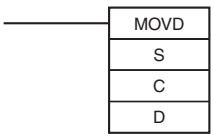
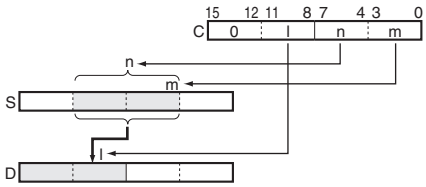
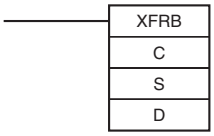
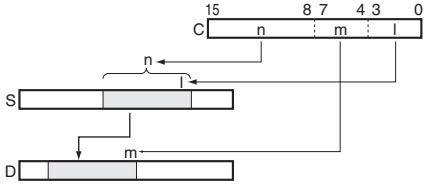
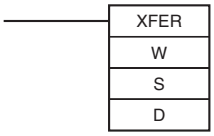
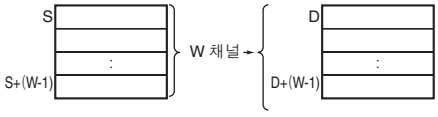
부록-1-5 데이터 비교 명령

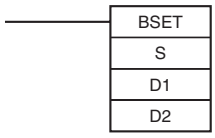
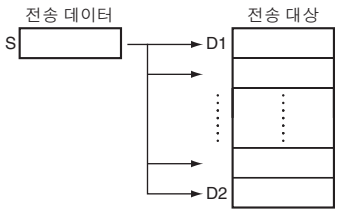
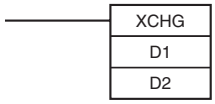
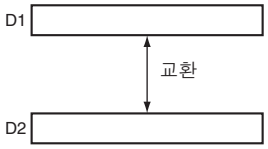
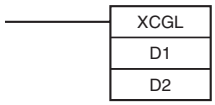
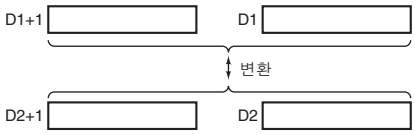
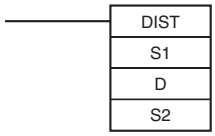
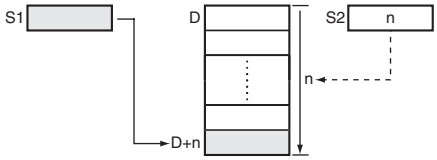
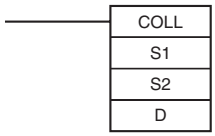
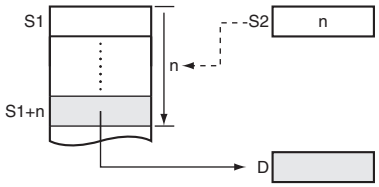
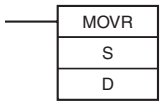
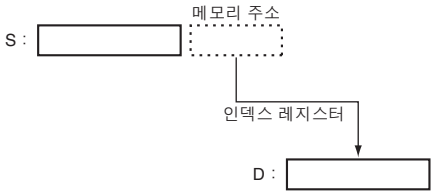
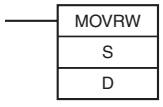
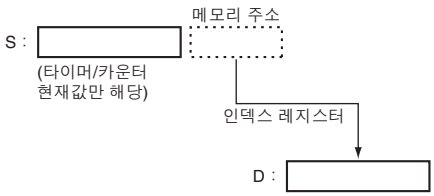
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
심볼 비교(부호 없음)	LD,AND, OR + =, < >, <, < =, >, >=	300 (=) 305 (< >) 310 (<) 315 (<=) 320 (>) 325 (>=)	LD(로드) 연결형  AND(직렬) 연결형  OR(병렬) 연결형  S1: 비교 데이터 1 S2: 비교 데이터 2	CH 데이터 또는 상수를 부호 없는 BIN 16비트(16진수 4자리)로 비교(=, < >, <, < =, >, > = 중 하나)하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다. 연결 형식에는 LD(로드) 연결, AND(직렬) 연결, OR(병렬) 연결의 3종류가 있습니다. LD(로드) 연결형  AND(직렬) 연결형  OR(병렬) 연결형  	LD형: 논리 시작 AND, OR형: 다음 단계 연결형	LD형 : 필요 없음 AND, OR형 : 필요
심볼 비교(2워드, 부호 없음)	LD,AND,OR + =, < >, <, < =, >, > = + L	301 (=) 306 (< >) 311 (<) 316 (<=) 321 (>) 326 (>=)	S1: 비교 데이터 1 하위 CH 번호 S2: 비교 데이터 2 하위 CH 번호	2CH분의 CH 데이터 또는 상수를 부호 없는 BIN 32비트(16진수 8자리)로 비교(=, < >, <, < =, >, > = 중 하나)하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다. 연결 형식에는 LD(로드) 연결, AND(직렬) 연결, OR(병렬) 연결의 3종류가 있습니다.	LD형: 논리 시작 AND, OR형: 다음 단계 연결형	LD형 : 필요 없음 AND, OR형 : 필요
심볼 비교(부호 있음)	LD,AND,OR + =, < >, <, < =, >, > = + S	302 (=) 307 (< >) 312 (<) 317 (< =) 322 (>) 327 (>=)	S1: 비교 데이터 1 S2: 비교 데이터 2	CH 데이터 또는 상수를 부호 있는 BIN 16비트(16진수 4자리)로 비교(=, < >, <, < =, >, > = 중 하나)하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다. 연결 형식에는 LD(로드) 연결, AND(직렬) 연결, OR(병렬) 연결의 3종류가 있습니다.	LD형: 논리 시작 AND, OR형: 다음 단계 연결형	LD형 : 필요 없음 AND, OR형 : 필요
심볼 비교(2워드, 부호 있음)	LD,AND, OR + =, < >, <, =, >, > = + SL	303 (=) 308 (< >) 313 (<) 318 (< =) 323 (>) 328 (>=)	S1: 비교 데이터 1 하위 CH 번호 S2: 비교 데이터 2 하위 CH 번호	2CH분의 CH 데이터 또는 상수를 부호 있는 BIN 32비트(16진수 8자리)로 비교(=, < >, <, < =, >, > = 중 하나)하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다. 연결 형식에는 LD(로드) 연결, AND(직렬) 연결, OR(병렬) 연결의 3종류가 있습니다.	LD형: 논리 시작 AND, OR형: 다음 단계 연결형	LD형 : 필요 없음 AND, OR형 : 필요

명령 이름	니모닉	평선 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
시각 비교	LD, AND, OR + =DT, < >DT, < DT, < =DT, > DT, > =DT	341 (=DT) 342 (< > DT) 343 (< DT) 344 (< = DT) 345 (> DT) 346 (> = DT)	LD(로드) 연결형  AND(직렬) 연결형  OR(병렬) 연결형  C: 제어 데이터 S1: 현재 시각 데이터 하위 CH 번호 S2: 비교 시각 데이터 하위 CH 번호	2개의 시각 데이터(BCD 데이터)를 비교하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다. • 연결 형식에는 LD(로드) 연결, AND(직렬) 연결, OR(병렬) 연결의 3종류가 있습니다. • 비교 대상 데이터를 한정할 수 있으므로 캘린더 타이머 기능을 구현할 수 있습니다.	LD형: 논리 시작 AND, OR형: 다음 단계 연결형	LD형 : 필요 없음 AND, OR형 : 필요
부호 없는 비교	CMP !CMP	020	 S1: 비교 데이터 1 S2: 비교 데이터 2	CH 데이터 또는 상수를 부호 없는 BIN 16비트(16진수 4자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.  	출력 계통	필요
부호 없는 2워드 비교	CMPL	060	 S1: 비교 데이터 1 하위 CH 번호 S2: 비교 데이터 2 하위 CH 번호	2CH분의 CH 데이터 또는 상수를 부호 없는 BIN 32비트(16진수 8자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.  	출력 계통	필요
부호 있는 BIN 비교	CPS !CPS	114	 S1: 비교 데이터 1 S2: 비교 데이터 2	CH 데이터 또는 상수를 부호 있는 BIN 16비트(최상위 비트를 부호 비트로 하는 16진수 4자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.  	출력 계통	필요
부호 있는 BIN 2워드 비교	CPSL	115	 S1: 비교 데이터 1 하위 CH 번호 S2: 비교 데이터 2 하위 CH 번호	2CH분의 CH 데이터 또는 상수를 부호 있는 2워드 BIN 32비트(최상위 비트를 부호 비트로 하는 16진수 8자리)로 비교하여 비교 결과를 조건 플래그에 반영합니다.  	출력 계통	필요
다채널 비교	MCMP @MCMP	019	 S1: 비교 데이터 1 하위 CH 번호 S2: 비교 데이터 2 하위 CH 번호 D: 비교 결과 출력 CH 번호	16CH분의 데이터를 CH 단위로 비교하여 지정된 CH의 해당 비트에 일치하는지 여부의 비교 결과를 출력합니다.  	출력 계통	필요

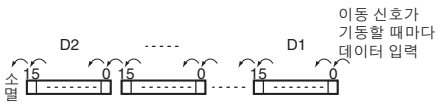
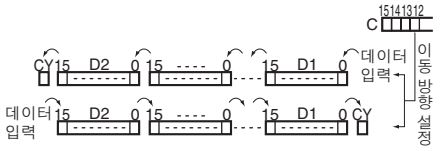
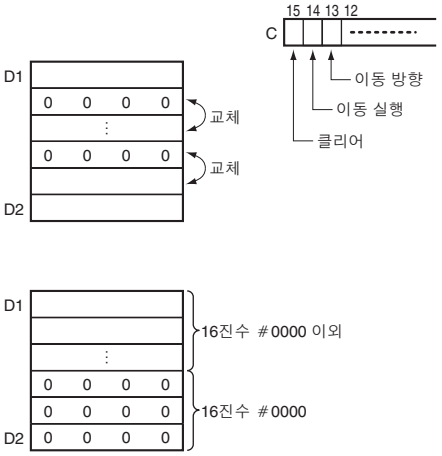
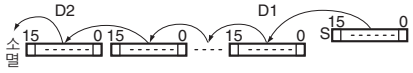
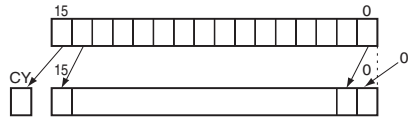
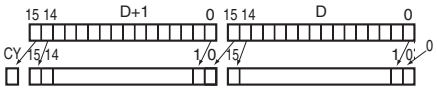
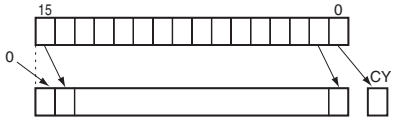
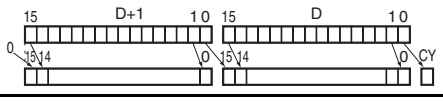
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
테이블 일치	TCMP @TCMP	085	TCMP S T D S: 비교 데이터 T: 비교 테이블 하위 CH 번호 D: 비교 결과 출력 CH 번호	비교 데이터 1CH를 비교 테이블 16CH분의 데이터와 각각 일치하는지 비교하여 출력 CH의 해당 비트에 1을 출력합니다. 	출력 계통	필요
부호 없는 테이블 간 비교	BCMP @BCMP	068	BCMP S T D S: 비교 데이터 T: 비교 테이블 하위 CH 번호 D: 비교 결과 출력 CH 번호	비교 데이터의 내용이 16쌍의 비교 데이터의 상하한 범위 안에 있는지 여부를 판별하여 범위 이내이면 출력 CH의 해당 비트에 1을 출력합니다. 	출력 계통	필요
확장 테이블 간 비교	BCMP2 @BCMP2	502	BCMP2 S T D S: 비교 데이터 T: 설정 테이블 하위 CH 번호 D: 비교 결과 출력 CH 번호	비교 데이터의 내용이 최대 256개인 비교 데이터 상하한 범위 안에 있는지 여부를 판별하여 범위 이내이면 D~D+최대 15CH의 해당 비트에 1을 출력합니다. 	출력 계통	필요
영역 비교	ZCP	088	ZCP S T1 T2 S: 비교 데이터 (1CH 데이터) T1: 하한값 T2: 상한값	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.	출력 계통	필요
2워드 영역 비교	ZCPL	116	ZCPL S T1 T2 S: 비교 데이터 (2CH 데이터) T1: 하한값 하위 CH 번호 T2: 상한값 하위 CH 번호	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.	출력 계통	필요
부호 있는 영역 비교	ZCPS	117	ZCPS S T1 T2 S: 비교 데이터 (1CH 데이터) T1: 하한값 T2: 상한값	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.	출력 계통	필요
부호 있는 2워드 영역 비교	ZCPSL	118	ZCPSL S T1 T2 S: 비교 데이터 (2CH 데이터) T1: 하한값 하위 CH 번호 T2: 상한값 하위 CH 번호	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.	출력 계통	필요

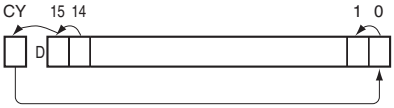
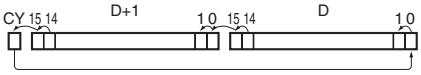
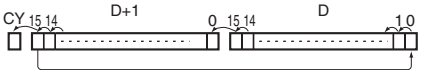
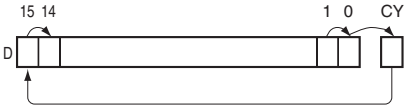
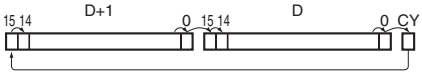
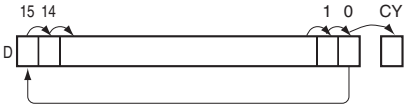
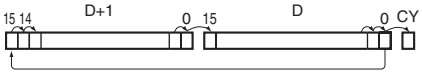
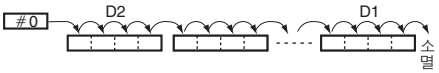
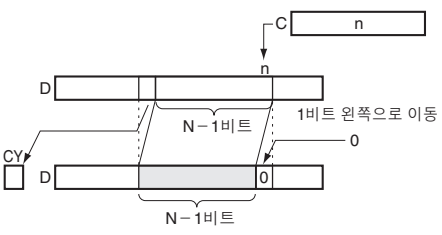
부록-1-6 데이터 전송 명령

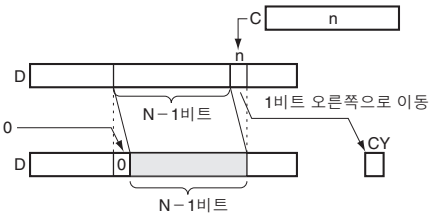
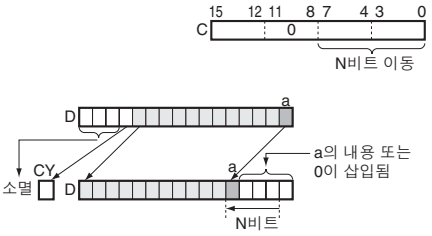
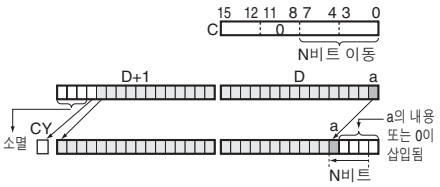
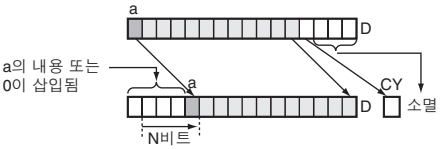
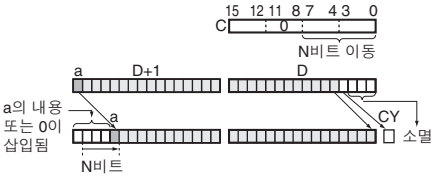
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
전송	MOV @MOV !MOV !@MOV	021	 S: 전송 데이터 D: 전송 대상 CH 번호	CH 데이터 또는 상수를 지정된 CH로 전송합니다. 	출력 계통	필요
2워드 전송	MOVL @MOVL	498	 S: 전송 데이터 하위 CH 번호 D: 전송 대상 하위 CH 번호	2CH분의 CH 데이터 또는 상수를 32비트 단위로 전송 대상 CH에 출력합니다. 	출력 계통	필요
부정 전송	MVN @MVN	022	 S: 전송 데이터 D: 전송 대상 CH 번호	CH 데이터 또는 상수의 비트 반전 데이터를 지정된 CH로 전송합니다. 	출력 계통	필요
부정 2워드 전송	MVNL @MVNL	499	 S: 전송 데이터 하위 CH 번호 D: 전송 대상 하위 CH 번호	2CH분의 CH 데이터 또는 상수의 비트 반전 데이터를 32비트 단위로 지정된 CH에 출력합니다. 	출력 계통	필요
비트 전송	MOVB @MOVB	082	 S: 전송 대상 CH 번호 C: 제어 데이터 D: 전송 대상 CH 번호	지정한 비트를 전송합니다. 	출력 계통	필요
자릿수 전송	MOVD @MOVD	083	 S: 전송 대상 CH 번호 C: 제어 데이터 D: 전송 대상 CH 번호	자리(4비트) 단위로 전송하며 여러 자리의 전송도 가능합니다. 	출력 계통	필요
다중 비트 전송	XFRB @XFRB	062	 C: 제어 데이터 S: 전송 원본 하위 CH 번호 D: 전송 대상 하위 CH 번호	지정된 CH의 지정된 복수의 비트를 전송합니다. 	출력 계통	필요
블록 전송	XFER @XFER	070	 W: 전송 CH 수 S: 전송 원본 하위 CH 번호 D: 전송 대상 하위 CH 번호	연속된 여러 CH 데이터를 일괄 전송합니다. 	출력 계통	필요

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
블록 설정	BSET @BSET	071	 S: 전송 데이터 D1: 전송 대상 하위 CH 번호 D2: 전송 대상 상위 CH 번호	연속된 복수의 CH 전체에 동일한 데이터를 설정합니다. 	출력 계통	필요
데이터 교환	XCHG @XCHG	073	 D1: 교환 CH 번호 1 D2: 교환 CH 번호 2	CH 간에 데이터를 교환합니다. 	출력 계통	필요
데이터 2워드 교환	XCGL @XCGL	562	 D1: 교환 하위 CH 번호 1 D2: 교환 하위 CH 번호 2	2CH의 CH 데이터끼리 32비트 단위로 교환합니다. 	출력 계통	필요
데이터 분배	DIST @DIST	080	 S1: 전송 데이터 D: 전송 대상 기준 CH 번호 S2: 오프셋 데이터	전송 대상을 기준으로 오프셋 지정된 채널로 전송 데이터를 전송합니다. 	출력 계통	필요
데이터 추출	COLL @COLL	081	 S1: 전송 원본 기준 CH 번호 S2: 오프셋 데이터 D: 전송 대상 CH 번호	전송 원본을 기준으로 오프셋 지정된 채널의 내용을 지정 채널로 전송합니다. 	출력 계통	필요
인덱스 레지스터 설정	MOVR @MOVR	560	 S: 지정 CH 번호/접점 번호 D: 전송 대상 인덱스 레지스터 번호	지정된 CH 번호, 접점 번호의 I/O 메모리 실제 주소를 지정된 인덱스 레지스터에 설정합니다. 	출력 계통	필요
인덱스 레지스터 설정	MOVRW @MOVRW	561	 S: 지정 타이머/카운터 번호 D: 전송 대상 인덱스 레지스터 번호	지정된 타이머/카운터 현재값의 I/O 메모리 실제 주소를 지정된 인덱스 레지스터에 설정합니다. 	출력 계통	필요

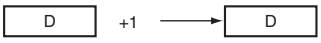
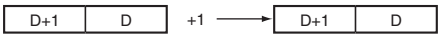
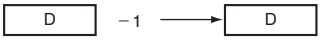
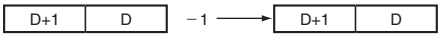
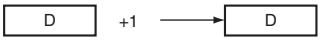
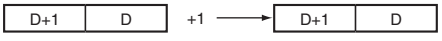
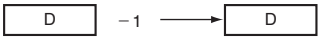
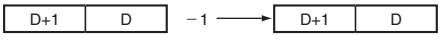
부록-1-7 데이터 이동 명령

명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
이동 레지스터	SFT	010	데이터 입력 이동 신호 입력 리셋 입력 SFT D1 D2 D1: 이동 하위 CH 번호 D2: 이동 상위 CH 번호	이동 레지스터 동작을 수행합니다. 	출력 계통	필요
좌우 이동 레지스터	SFTR @SFTR	084	SFTR C D1 D2 C: 제어 데이터 D1: 이동 하위 CH 번호 D2: 이동 상위 CH 번호	이동 방향을 전환할 수 있는 이동 레지스터 동작을 수행합니다. 	출력 계통	필요
비동기 이동 레지스터	ASFT @ASFT	017	ASFT C D1 D2 C: 제어 데이터 D1: 이동 하위 CH 번호 D2: 이동 상위 CH 번호	지정 CH 범위의 채널 데이터 중 #0000 이외의 채널 데이터에 대해 앞 또는 뒤 채우기를 실행하고 #0000의 채널 데이터와 위치를 교체합니다. 	출력 계통	필요
워드 이동	WSFT @WSFT	016	WSFT S D1 D2 S: 이동 데이터 번호 D1: 이동 하위 CH 번호 D2: 이동 상위 CH 번호	CH 데이터 단위의 이동 동작을 나타냅니다. 	출력 계통	필요
1비트 좌측 이동	ASL @ASL	025	ASL D D: 이동 CH 번호	CH 데이터를 왼쪽으로 1비트 이동합니다. 	출력 계통	필요
1비트 좌측 이동(2워드)	ASLL @ASLL	570	ASLL D D: 이동 하위 CH 번호	2CH분의 데이터를 왼쪽으로 1비트 이동합니다. 	출력 계통	필요
1비트 우측 이동	ASR @ASR	026	ASR D D: 이동 CH 번호	CH 데이터를 오른쪽으로 1비트 이동합니다. 	출력 계통	필요
1비트 우측 이동(2워드)	ASRL @ASRL	571	ASRL D D: 이동 하위 CH 번호	2CH분의 데이터를 오른쪽으로 1비트 이동합니다. 	출력 계통	필요

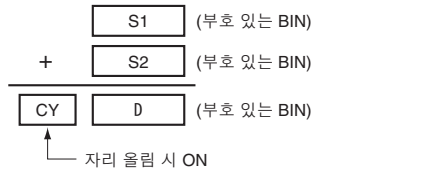
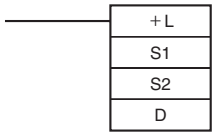
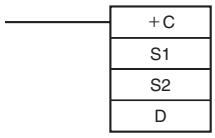
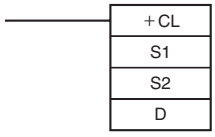
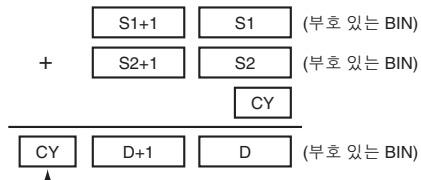
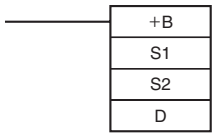
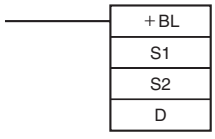
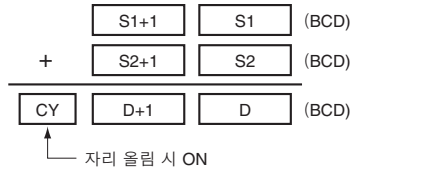
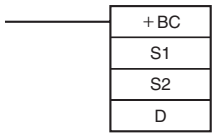
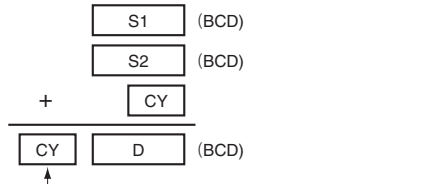
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
CY가 있는 1비트 좌측 회전	ROL @ROL	027	ROL D D: 이동 CH 번호	16비트의 채널 데이터를 캐리(CY) 플래그를 포함하여 1비트 좌측으로 회전합니다. 	출력 계통	필요
CY가 있는 1비트 좌측 회전(2워드)	ROLL @ROLL	572	ROLL D D: 이동 하위 CH 번호	32비트의 채널 데이터를 캐리(CY) 플래그를 포함하여 1비트 좌측으로 회전합니다. 	출력 계통	필요
CY가 없는 1비트 좌측 회전	RLNC @RLNC	574	RLNC D D: 이동 CH 번호	16비트의 채널 데이터를 캐리(CY) 플래그를 포함하지 않고 1비트 좌측으로 회전합니다. 	출력 계통	필요
CY가 없는 1비트 좌측 회전(2워드)	RLNL @RLNL	576	RLNL D D: 이동 하위 CH 번호	32비트의 채널 데이터를 캐리(CY) 플래그를 포함하지 않고 1비트 좌측으로 회전합니다. 	출력 계통	필요
CY가 있는 1비트 우측 회전	ROR @ROR	028	ROR D D: 이동 CH 번호	16비트의 채널 데이터를 캐리(CY) 플래그를 포함하여 1비트 우측으로 회전합니다. 	출력 계통	필요
CY가 있는 1비트 우측 회전(2워드)	RORL @RORL	573	RORL D D: 이동 하위 CH 번호	32비트의 채널 데이터를 캐리(CY) 플래그를 포함하여 1비트 우측으로 회전합니다. 	출력 계통	필요
CY가 없는 1비트 우측 회전	RRNC @RRNC	575	RRNC D D: 이동 CH 번호	16비트의 채널 데이터를 캐리(CY) 플래그를 포함하지 않고 1비트 우측으로 회전합니다. 	출력 계통	필요
CY가 없는 1비트 우측 회전(2워드)	RRNL @RRNL	577	RRNL D D: 이동 하위 CH 번호	32비트의 채널 데이터를 캐리(CY) 플래그를 포함하지 않고 1비트 우측으로 회전합니다. 	출력 계통	필요
1자리 좌측 이동	SLD @SLD	074	SLD D1 D2 D1: 이동 하위 CH 번호 D2: 이동 상위 CH 번호	연속된 채널의 데이터 내용을 1자리(4비트) 단위로 좌측 이동합니다. 	출력 계통	필요
1자리 우측 이동	SRD @SRD	075	SRD D1 D2 D1: 이동 하위 CH 번호 D2: 이동 상위 CH 번호	연속된 채널의 데이터 내용을 1자리(4비트) 단위로 우측 이동합니다. 	출력 계통	필요
N비트 데이터 좌측 이동	NSFL @NSFL	578	NSFL D C N D: 이동 하위 CH 번호 C: 이동 시작 비트 N: 이동 데이터 길이	지정된 비트 수 분량의 비트 데이터를 좌측으로 1비트 이동합니다. 	출력 계통	필요

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
N비트 데이터 우측 이동	NSFR @NSFR	579	NSFR D C N D: 이동 하위 CH 번호 C: 이동 시작 비트 N: 이동 데이터 길이	지정된 비트 수 분량의 비트 데이터를 우측으로 1비트 이동합니다. 	출력 계통	필요
N비트 좌측 이동	NASL @NASL	580	NASL D C D: 이동 CH 번호 C: 제어 데이터	16비트의 채널 데이터를 지정 비트 수만큼 좌측으로 이동합니다. 	출력 계통	필요
N비트 2워드 좌측 이동	NSLL @NSLL	582	NSLL D C D: 이동 하위 CH 번호 C: 제어 데이터	32비트의 채널 데이터를 지정 비트 수만큼 좌측으로 이동합니다. 	출력 계통	필요
N비트 우측 이동	NASR @NASR	581	NASR D C D: 이동 CH 번호 C: 제어 데이터	16비트의 채널 데이터를 지정 비트 수만큼 우측으로 이동합니다. 	출력 계통	필요
N비트 2워드 우측 이동	NSRL @NSRL	583	NSRL D C D: 이동 하위 CH 번호 C: 제어 데이터	32비트의 채널 데이터를 지정 비트 수만큼 우측으로 이동합니다. 	출력 계통	필요

부록-1-8 증가/감소 명령

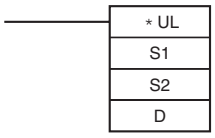
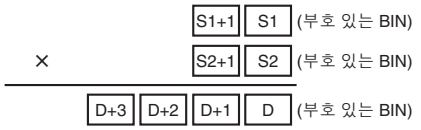
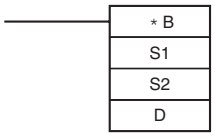
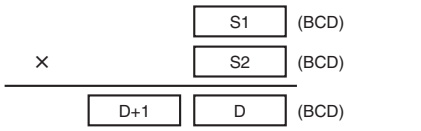
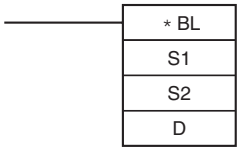
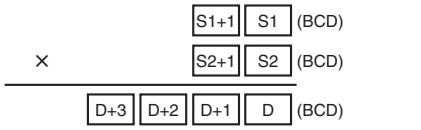
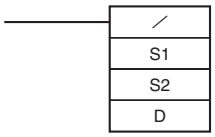
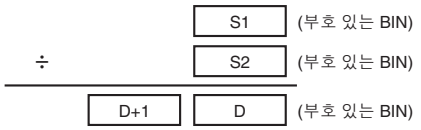
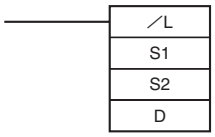
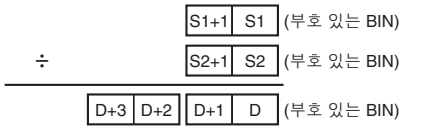
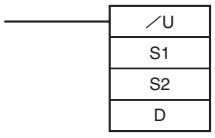
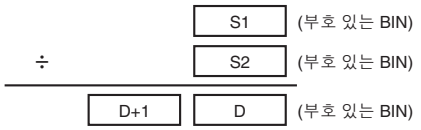
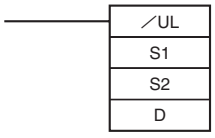
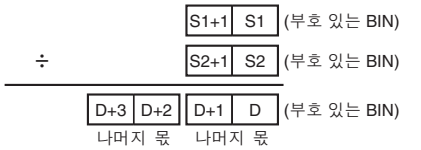
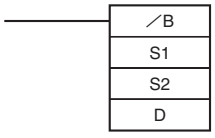
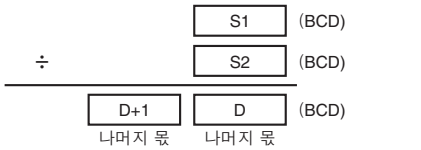
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
BIN 증가	++ @++	590	++ D D: 데이터 CH 번호	16진수 4자리의 1CH 데이터에 1을 더합니다. 	출력 계통	필요
BIN 2워드 증가	++L @++L	591	++L D D: 데이터 하위 CH 번호	16진수 8자리의 2CH 데이터에 1을 더합니다. 	출력 계통	필요
BIN 감소	−− @−−	592	−− D D: 데이터 CH 번호	16진수 4자리의 1CH 데이터에서 1을 뺍니다. 	출력 계통	필요
BIN 2워드 감소	−−L @−−L	593	−−L D D: 데이터 하위 CH 번호	16진수 8자리의 2CH 데이터에서 1을 뺍니다. 	출력 계통	필요
BCD 증가	++B @++B	594	++B D D: 데이터 CH 번호	BCD 4자리의 1CH 데이터에 1을 더합니다. 	출력 계통	필요
BCD 2워드 증가	++BL @++BL	595	++BL D D: 데이터 하위 CH 번호	BCD 8자리의 2CH 데이터에 1을 더합니다. 	출력 계통	필요
BCD 감소	−−B @−−B	596	−−B D D: 데이터 CH 번호	BCD 4자리의 1CH 데이터에서 1을 뺍니다. 	출력 계통	필요
BCD 2워드 감소	−−BL @−−BL	597	−−BL D D: 데이터 하위 CH 번호	BCD 8자리의 2CH 데이터에서 1을 뺍니다. 	출력 계통	필요

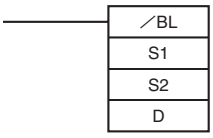
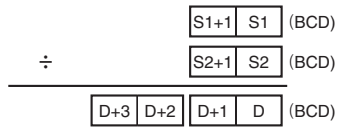
부록-1-9 사칙연산 명령

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
부호 있음, CY 없는 BIN 가산	+ @ +	400	 S1: 피가산 데이터 S2: 가산 데이터 D: 연산 결과 출력 CH 번호	채널 데이터나 상수를 부호 있는 16진수 4자리로 가산합니다. 	출력 계통	필요
부호 있음, CY 없는 BIN 2워드 가산	+L @ +L	401	 S1: 피가산 데이터 하위 CH 번호 S2: 가산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	2CH 데이터나 상수를 부호 있는 16진수 8자리로 가산합니다. 	출력 계통	필요
부호 및 CY 있는 BIN 가산	+C @ +C	402	 S1: 피가산 데이터 S2: 가산 데이터 D: 연산 결과 출력 CH 번호	데이터나 상수를 캐리(CY) 플래그를 포함하여 부호 있는 16진수 4자리로 가산합니다. 	출력 계통	필요
부호 및 CY 있는 BIN 2워드 가산	+CL @ +CL	403	 S1: 피가산 데이터 하위 CH 번호 S2: 가산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	2CH분의 데이터나 상수를 캐리(CY) 플래그를 포함하여 부호 있는 16진수 8자리로 가산합니다. 	출력 계통	필요
CY 없는 BCD 가산	+B @ +B	404	 S1: 피가산 데이터 S2: 가산 데이터 D: 연산 결과 출력 CH 번호	채널 데이터나 상수를 BCD 4자리로 가산합니다. 	출력 계통	필요
CY 없는 BCD 2워드 가산	+BL @ +BL	405	 S1: 피가산 데이터 하위 CH 번호 S2: 가산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	2CH분의 채널 데이터나 상수를 BCD 8자리로 가산합니다. 	출력 계통	필요
CY 있는 BCD 가산	+BC @ +BC	406	 S1: 피가산 데이터 S2: 가산 데이터 D: 연산 결과 출력 CH 번호	채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 부호 있는 BCD 4자리로 가산합니다. 	출력 계통	필요

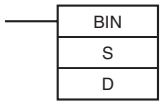
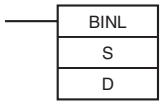
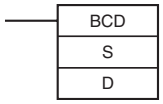
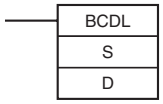
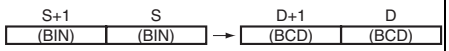
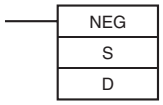
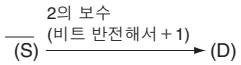
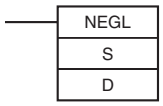
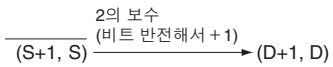
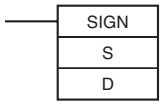
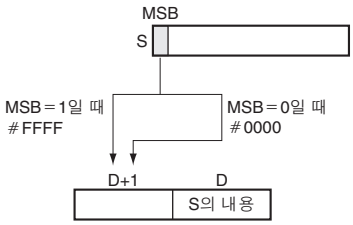
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
CY 있는 BCD 2 워드 가산	+BCL @+BCL	407	+BCL S1 S2 D S1: 피가산 데이터 S2: 가산 데이터 D: 연산 결과 출력 하위 CH 번호	2CH분의 채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 BCD 8자리로 가산합니다. S1+1 S1 (BCD) + S2+1 S2 (BCD) CY CY D+1 D (BCD) 자리 올림 시 ON	출력 계통	필요
부호 있음, CY 없는 BIN 감산	− @−	410	− S1 S2 D S1: 피감산 데이터 S2: 감산 데이터 D: 연산 결과 출력 CH 번호	채널 데이터나 상수를 부호 있는 16진수 4자리로 감산합니다. S1 (BIN) − S2 (BIN) CY D (BIN) 자리 내림 시 ON	출력 계통	필요
부호 있음, CY 없는 BIN 2워드 감산	−L @−L	411	−L S1 S2 D S1: 피감산 데이터 하위 CH 번호 S2: 감산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	2CH분의 채널 데이터나 상수를 부호 있는 16진수 8자리로 감산합니다. S1+1 S1 (BIN) − S2+1 S2 (BIN) CY D+1 D (BIN) 자리 내림 시 ON	출력 계통	필요
부호 및 CY 있는 BIN 감산	−C @−C	412	−C S1 S2 D S1: 피감산 데이터 S2: 감산 데이터 D: 연산 결과 출력 CH 번호	채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 부호 있는 16진수 4자리로 감산합니다. S1 (BIN) S2 (BIN) − CY CY D (BIN) 자리 내림 시 ON	출력 계통	필요
부호 및 CY 있는 BIN 2워드 감산	−CL @−CL	413	−CL S1 S2 D S1: 피감산 데이터 하위 CH 번호 S2: 감산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	2CH분의 채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 부호 있는 16진수 8자리로 감산합니다. S1+1 S1 (BIN) S2+1 S2 (BIN) − CY CY D+1 D (BIN) 자리 내림 시 ON	출력 계통	필요
CY 없는 BCD 감산	−B @−B	414	−B S1 S2 D S1: 피감산 데이터 S2: 감산 데이터 D: 연산 결과 출력 CH 번호	채널 데이터나 상수를 BCD 4자리로 감산합니다. S1 (BCD) − S2 (BCD) CY D (BCD) 자리 내림 시 ON	출력 계통	필요
CY 없는 BCD 2 워드 감산	−BL @−BL	415	−BL S1 S2 D S1: 피감산 데이터 하위 CH 번호 S2: 감산 데이터 CH 번호 D: 연산 결과 출력 하위 CH 번호	2CH분의 채널 데이터나 상수를 BCD 8자리로 감산합니다. S1+1 S1 (BCD) − S2+1 S2 (BCD) CY D+1 D (BCD) 자리 내림 시 ON	출력 계통	필요

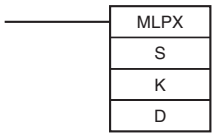
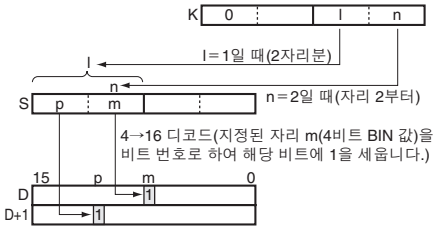
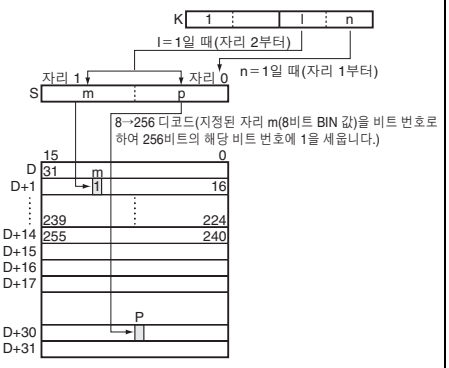
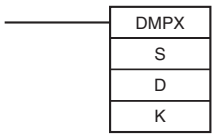
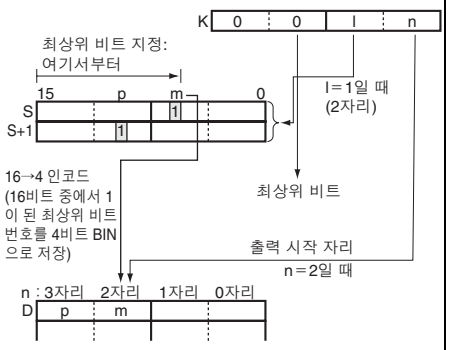
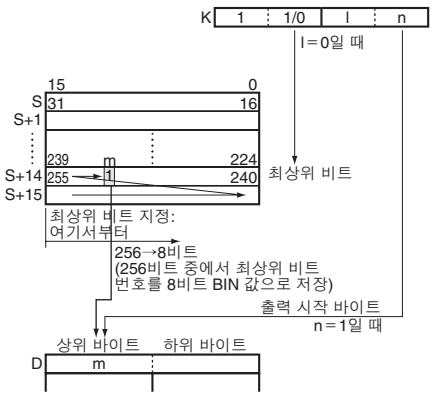
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
CY 있는 BCD 감산	−BC @−BC	416	−BC S1 S2 D S1: 피감산 데이터 S2: 감산 데이터 D: 연산 결과 출력 CH 번호	채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 BCD 4자리로 감산합니다. S1 (BCD) S2 (BCD) − CY CY D (BCD) 자리 내림 시 ON	출력 계통	필요
CY 있는 BCD 2 워드 감산	−BCL @−BCL	417	−BCL S1 S2 D S1: 피감산 데이터 하위 CH 번호 S2: 감산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	2CH분의 채널 데이터나 상수를 캐리(CY) 플래그를 포함하여 BCD 8자리로 감산합니다. S1+1 S1 (BCD) S2+1 S2 (BCD) − CY CY D+1 D (BCD) 자리 내림 시 ON	출력 계통	필요
부호 있는 BIN 승산	* @*	420	* S1 S2 D S1: 피승산 데이터 S2: 승산 데이터 D: 연산 결과 출력 하위 CH 번호	채널 데이터나 상수를 부호 있는 16진수 4자리로 곱합니다. S1 (부호 있는 BIN) × S2 (부호 있는 BIN) D+1 D (부호 있는 BIN)	출력 계통	필요
부호 있는 BIN 2워드 승산	*L @*L	421	*L S1 S2 D S1: 피승산 데이터 하위 CH 번호 S2: 승산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	2CH분의 채널 데이터나 상수를 부호 있는 16진수 8자리로 곱합니다. S1+1 S1 (부호 있는 BIN) × S2+1 S2 (부호 있는 BIN) D+3 D+2 D+1 D (부호 있는 BIN)	출력 계통	필요
부호 없는 BIN 승산	*U @*U	422	*U S1 S2 D S1: 피승산 데이터 S2: 승산 데이터 D: 연산 결과 출력 하위 CH 번호	채널 데이터나 상수를 부호 없는 16진수 4자리로 곱합니다. S1 (부호 있는 BIN) × S2 (부호 있는 BIN) D+1 D (부호 있는 BIN)	출력 계통	필요
부호 있는 BIN 승산	* @*	420	* S1 S2 D S1: 피승산 데이터 S2: 승산 데이터 D: 연산 결과 출력 하위 CH 번호	채널 데이터나 상수를 부호 있는 16진수 4자리로 곱합니다. S1 (부호 있는 BIN) × S2 (부호 있는 BIN) D+1 D (부호 있는 BIN)	출력 계통	필요
부호 있는 BIN 2워드 승산	*L @*L	421	*L S1 S2 D S1: 피승산 데이터 하위 CH 번호 S2: 승산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	2CH분의 채널 데이터나 상수를 부호 있는 16진수 8자리로 곱합니다. S1+1 S1 (부호 있는 BIN) × S2+1 S2 (부호 있는 BIN) D+3 D+2 D+1 D (부호 있는 BIN)	출력 계통	필요
부호 없는 BIN 승산	*U @*U	422	*U S1 S2 D S1: 피승산 데이터 S2: 승산 데이터 D: 연산 결과 출력 하위 CH 번호	채널 데이터나 상수를 부호 없는 16진수 4자리로 곱합니다. S1 (부호 있는 BIN) × S2 (부호 있는 BIN) D+1 D (부호 있는 BIN)	출력 계통	필요

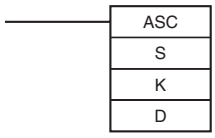
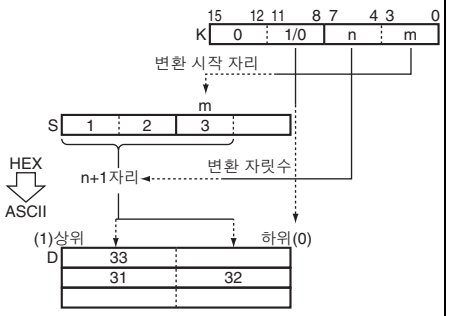
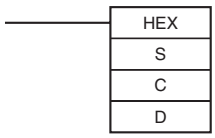
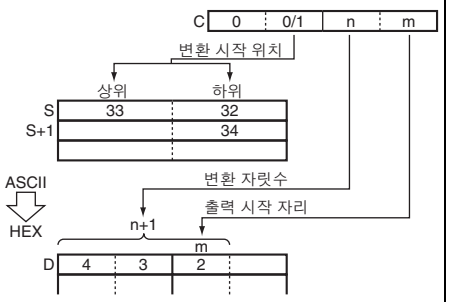
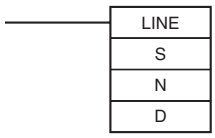
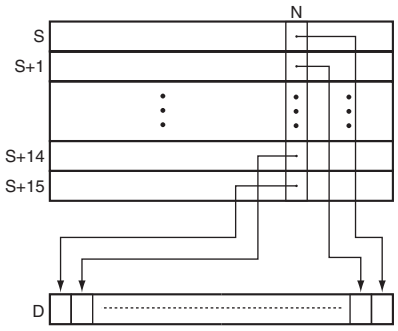
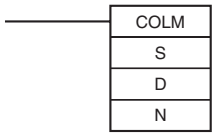
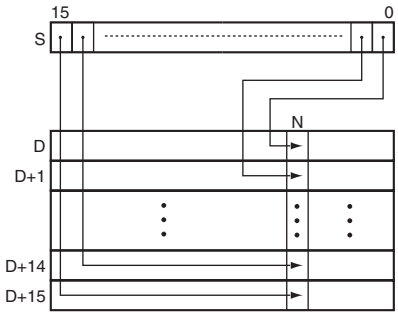
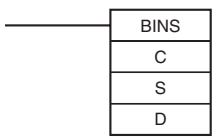
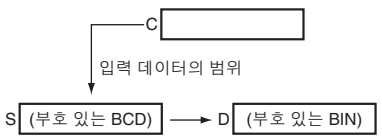
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
부호 없는 BIN 2워드 승산	* UL @ * UL	423	 S1: 피승산 데이터 하위 CH 번호 S2: 승산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	2CH분의 채널 데이터나 상수를 부호 없는 16진수 8자리로 곱합니다. 	출력 계통	필요
BCD 승산	* B @ * B	424	 S1: 피승산 데이터 S2: 승산 데이터 D: 연산 결과 출력 하위 CH 번호	채널 데이터나 상수를 BCD 4자리로 곱합니다. 	출력 계통	필요
BCD 2워드 승산	* BL @* BL	425	 S1: 피승산 데이터 하위 CH 번호 S2: 승산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	2CH분의 채널 데이터나 상수를 BCD 8자리로 곱합니다. 	출력 계통	필요
부호 있는 BIN 제산	/ @/	430	 S1: 피제산 데이터 S2: 제산 데이터 D: 연산 결과 출력 하위 CH 번호	채널 데이터나 상수를 부호 있는 16진수 4자리로 나눕니다. 	출력 계통	필요
부호 있는 BIN 2워드 제산	/ L @/ L	431	 S1: 피제산 데이터 하위 CH 번호 S2: 제산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	2CH분의 채널 데이터나 상수를 부호 있는 16진수 8자리로 나눕니다. 	출력 계통	필요
부호 없는 BIN 제산	/ U @/ U	432	 S1: 피제산 데이터 S2: 제산 데이터 D: 연산 결과 출력 CH 번호	채널 데이터나 상수를 부호 없는 16진수 4자리로 나눕니다. 	출력 계통	필요
부호 없는 BIN 2워드 제산	/ UL @/ UL	433	 S1: 피제산 데이터 하위 CH 번호 S2: 제산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	2CH분의 채널 데이터나 상수를 부호 없는 16진수 8자리로 나눕니다. 	출력 계통	필요
BCD 제산	/ B @/ B	434	 S1: 피제산 데이터 S2: 제산 데이터 D: 연산 결과 출력 하위 CH 번호	채널 데이터나 상수를 BCD 4자리로 나눕니다. 	출력 계통	필요

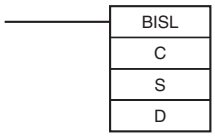
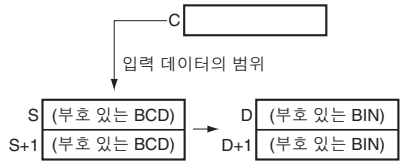
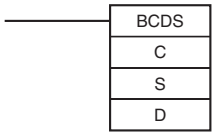
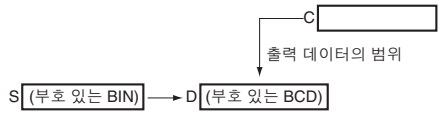
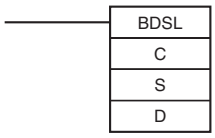
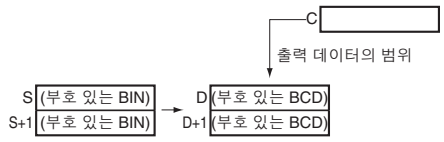
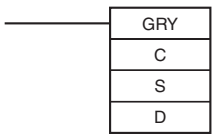
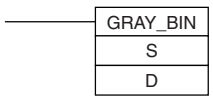
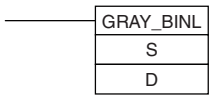
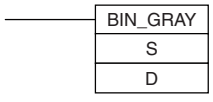
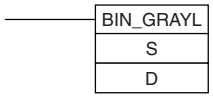
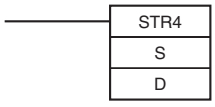
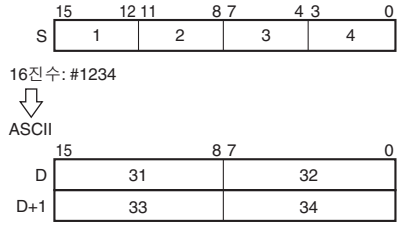
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
BCD 2워드 제산	/ BL @/ BL	435	 S1: 피제산 데이터 하위 CH 번호 S2: 제산 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	2CH분의 채널 데이터나 상수를 BCD 8자리로 나눕니다. 	출력 계통	필요

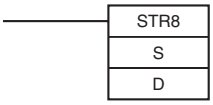
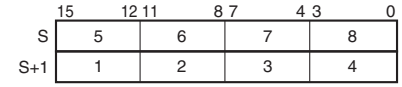
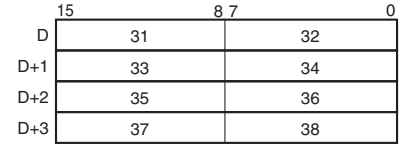
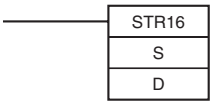
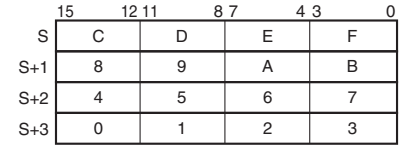
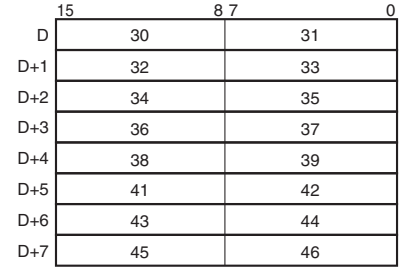
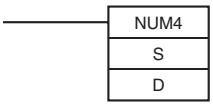
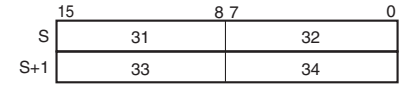
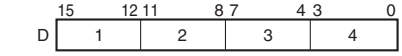
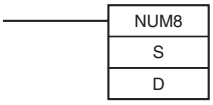
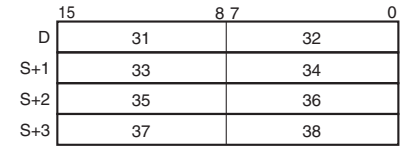
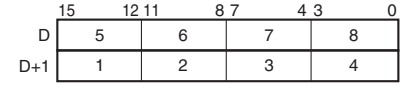
부록-1-10 데이터 변환 명령

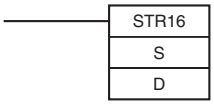
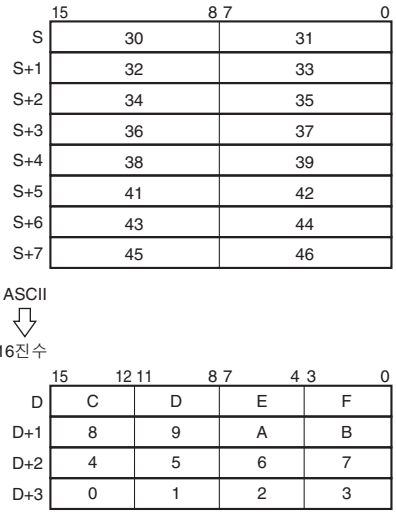
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
BCD→BIN 변환	BIN @BIN	023	 S: 변환 데이터 CH 번호 D: 변환 결과 출력 CH 번호	BCD 데이터를 BIN 데이터로 변환합니다. 	출력 계통	필요
BCD→BIN 2워드 변환	BINL @BINL	058	 S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	2워드 BCD 데이터를 BIN 데이터로 변환합니다. 	출력 계통	필요
BIN→BCD 변환	BCD @BCD	024	 S: 변환 데이터 CH 번호 D: 변환 결과 출력 CH 번호	BIN 데이터를 BCD 데이터로 변환합니다. 	출력 계통	필요
BIN→BCD 2워드 변환	BCDL @BCDL	059	 S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	2워드 BIN 데이터를 BCD 데이터로 변환합니다. 	출력 계통	필요
2의 보수 변환	NEG @NEG	160	 S: 변환 데이터 CH 번호 D: 변환 결과 출력 CH 번호	BIN 16비트 데이터의 2의 보수를 계산합니다. 	출력 계통	필요
2의 보수 변환 (2워드)	NEGL @NEGL	161	 S: 변환 데이터 하위 CH 번호 D: 변환 결과 하위 CH 번호	BIN 32비트 데이터의 2의 보수를 계산합니다. 	출력 계통	필요
부호 확장	SIGN @SIGN	600	 S: 확장 데이터 CH 번호 D: 결과 출력 하위 CH 번호	지정 CH의 데이터를 1CH의 부호 있는 BIN 데이터로 2CH에 부호 확장합니다. 	출력 계통	필요

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
4→16/8→ 256 디코더	MLPX @MLPX	076	 S: 변환 데이터 CH 번호 K: 제어 데이터(자리 지정) D: 변환 결과 출력 하위 CH 번호	지정된 CH의 지정된 자리 또는 바이트를 읽어서(4→16 디코드 또는 8→256 디코드) 지정된 CH의 해당 비트에 1을 출력하고 다른 비트에는 0을 출력합니다. 4→16 디코더 지정 시  8→256 디코더 지정 시 	출력 계통	필요
16→4/256→ 8 인코더	DMPX @DMPX	077	 S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 CH 번호 K: 제어 데이터(자리 지정)	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. 16→4 인코더 지정 시  256→8 인코더 지정 시 	출력 계통	필요

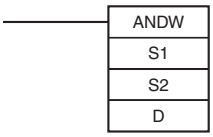
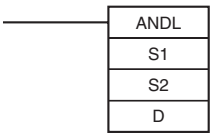
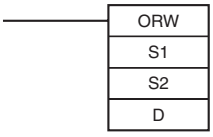
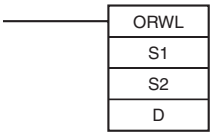
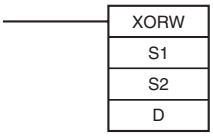
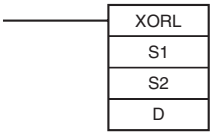
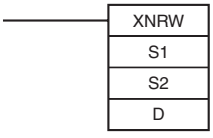
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
ASCII 코드 변환	ASC @ASC	086	 S: 변환 데이터 CH 번호 K: 자리 지정 데이터 D: 변환 결과 출력 하위 CH 번호	지정된 16비트 데이터 자리를 ASCII 코드로 변환합니다. 	출력 계통	필요
ASCII→HEX 변환	HEX @HEX	162	 S: 변환 데이터 하위 CH 번호 C: 제어 데이터 D: 변환 결과 출력 CH 번호	지정된 CH 데이터 자리의 내용을 8비트 ASCII 데이터로 처리하여 지정된 CH의 지정된 자리에 대응하는 문자로 출력합니다. 	출력 계통	필요
비트 열→ 비트 행 변환	LINE @LINE	063	 S: 변환 데이터 하위 CH 번호 N: 비트 위치 지정 데이터 D: 변환 결과 출력 CH 번호	지정된 16CH분의 데이터 중 지정된 비트의 데이터만 지정된 다른 CH의 해당 비트로 설정합니다. 	출력 계통	필요
비트 행→ 비트 열 변환	COLM @COLM	064	 S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호 N: 비트 위치 지정 데이터	지정된 CH의 각 비트 데이터를 지정된 다른 CH에서 16CH의 지정된 비트로 설정합니다. 	출력 계통	필요
부호 있는 BCD →BIN 변환	BINS @BINS	470	 C: 데이터 형식 지정 번호 S: 변환 데이터 CH 번호 D: 변환 결과 출력 CH 번호	부호 있는 BCD 데이터를 부호 있는 BIN 데이터로 변환합니다. 	출력 계통	필요

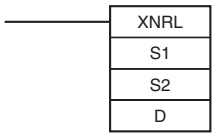
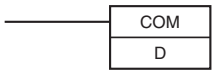
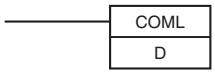
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
부호 있는 BCD →BIN 2워드 변환	BISL @BISL	472	 C: 데이터 형식 지정 번호 S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	부호 있는 2워드 BCD 데이터를 부호 있는 2워드 BIN 데이터로 변환합니다. 	출력 계통	필요
부호 있는 BIN →BCD 변환	BCDS @BCDS	471	 C: 데이터 형식 지정 번호 S: 변환 데이터 CH 번호 D: 변환 결과 출력 CH 번호	부호 있는 BIN 데이터를 부호 있는 BCD 데이터로 변환합니다. 	출력 계통	필요
부호 있는 BIN →BCD 2워드 변환	BDSL @BDSL	473	 C: 데이터 형식 지정 번호 S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	부호 있는 2워드 BIN 데이터를 부호 있는 2워드 BCD 데이터로 변환합니다. 	출력 계통	필요
그레이 코드 변환	GRY	474	 C: 제어 데이터 S: 그레이 2진수 코드 데이터 D: 변환 결과 출력 하위 CH 번호	지정 CH 내의 그레이 2진수 코드 데이터를 지정된 분해능에 따라 BIN 데이터, BCD 데이터 또는 각도(°) 데이터 중 하나로 변환합니다.	출력 계통	필요
그레이 코드→ BIN 변환	GRAY_BIN @GRAY_BIN	478	 S: 그레이 코드 D: BIN 데이터	S로 지정한 그레이 코드(1워드)를 BIN 코드(1워드)로 변환하여 D에 출력합니다.	출력 계통	필요
그레이 코드→ BIN 2워드 변환	GRAY_BINL @GRAY_BINL	479	 S: 그레이 코드 D: BIN 데이터	S로 지정한 그레이 코드(2워드)를 BIN 코드(2워드)로 변환하여 D에 출력합니다.	출력 계통	필요
BIN→그레이 코드 변환	BIN_GRAY @BIN_GRAY	480	 S: BIN 데이터 D: 그레이 코드	S로 지정한 BIN 코드(1워드)를 그레이 코드(1워드)로 변환하여 D에 출력합니다.	출력 계통	필요
BIN→그레이 코드 2워드 변환	BIN_GRAYL @BIN_GRAYL	481	 S: BIN 데이터 D: 그레이 코드	S로 지정한 BIN 코드(2워드)를 그레이 코드(2워드)로 변환하여 D에 출력합니다.	출력 계통	필요
숫자 4자리→ ASCII 데이터 변환	STR4	601	 S: 변환 데이터 CH 번호 D: 변환 결과 출력 하위 CH 번호	숫자 데이터(16진수 4자리)를 ASCII 데이터(4문자)로 변환합니다. 	출력 계통	필요

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
숫자 8자리→ ASCII 데이터 변환	STR8	602	 S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	숫자 데이터(16진수 8자리)를 ASCII 데이터(8문자)로 변환합니다.  16진수: #12345678 ⇩ ASCII 	출력 계통	필요
숫자 16자리→ ASCII 데이터 변환	STR16	603	 S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	숫자 데이터(16진수 16자리)를 ASCII 데이터(16문자)로 변환합니다.  16진수: #0123456789ABCDEF ⇩ ASCII 	출력 계통	필요
ASCII 데이터→ 숫자 4자리 변환	NUM4	604	 S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 CH 번호	ASCII 데이터(4문자)를 숫자 데이터(16진수 4자리)로 변환합니다.  ASCII ⇩ 16진수 	출력 계통	필요
ASCII 데이터→ 숫자 8자리 변환	NUM8	605	 S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	ASCII 데이터(8문자)를 숫자 데이터(16진수 8자리)로 변환합니다.  ASCII ⇩ 16진수 	출력 계통	필요

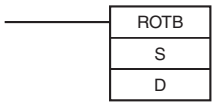
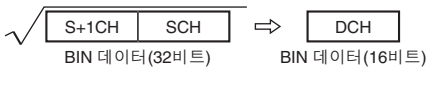
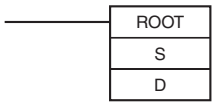
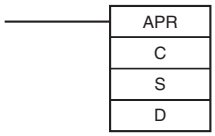
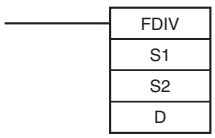
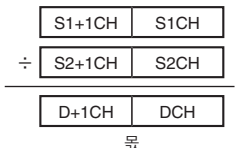
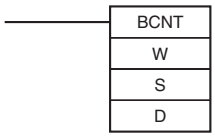
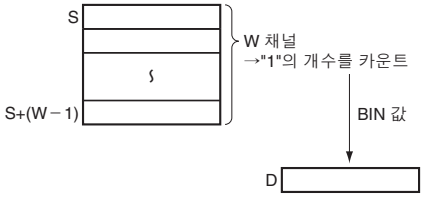
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
ASCII 데이터→숫자 16자리 변환	NUM16	606	 S: 변환 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	ASCII 데이터(16문자)를 숫자 데이터(16진수 16자리)로 변환합니다. 	출력 계통	필요

부록-1-11 논리 연산 명령

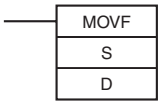
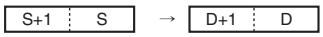
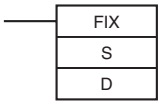
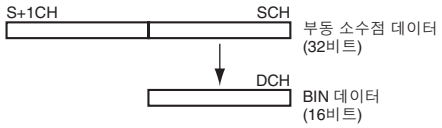
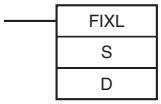
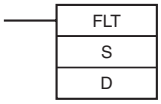
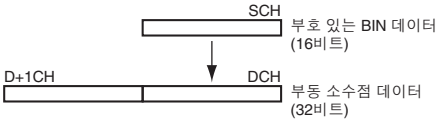
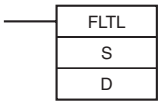
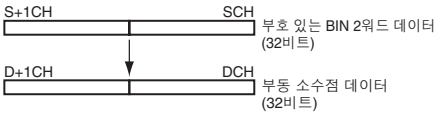
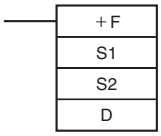
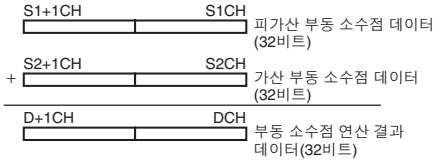
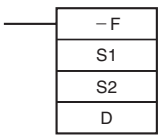
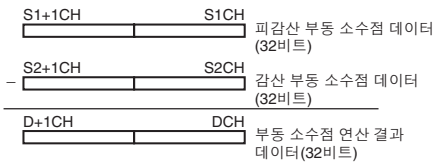
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
워드 논리곱	ANDW @ANDW	034	 S1: 연산 데이터 1 S2: 연산 데이터 2 D: 연산 결과 출력 CH 번호	CH 데이터 단위에서 CH 데이터와 CH 데이터, 상수의 논리곱을 계산합니다. S1 · S2→D S1 | S2 | D 1 | 1 | 1 1 | 0 | 0 0 | 1 | 0 0 | 0 | 0	출력 계통	필요
워드 논리곱 (2워드)	ANDL @ANDL	610	 S1: 연산 데이터 1 하위 CH 번호 S2: 연산 데이터 2 하위 CH 번호 D: 연산 결과 출력 CH 번호	2CH분의 CH 데이터와 CH 데이터, 상수의 논리곱을 계산합니다. (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	출력 계통	필요
워드 논리합	ORW @ORW	035	 S1: 연산 데이터 1 S2: 연산 데이터 2 D: 연산 결과 출력 CH 번호	CH 데이터 단위에서 CH 데이터와 CH 데이터, 상수의 논리합을 계산합니다. S1+S2→D S1 | S2 | D 1 | 1 | 1 1 | 0 | 1 0 | 1 | 1 0 | 0 | 0	출력 계통	필요
워드 논리합 (2워드)	ORWL @ORWL	611	 S1: 연산 데이터 1 하위 CH 번호 S2: 연산 데이터 2 하위 CH 번호 D: 연산 결과 출력 CH 번호	2CH분의 CH 데이터와 CH 데이터, 상수의 논리합을 계산합니다. (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	출력 계통	필요
워드 배타적 논리합	XORW @XORW	036	 S1: 연산 데이터 1 S2: 연산 데이터 2 D: 연산 결과 출력 CH 번호	CH 데이터 단위에서 CH 데이터와 CH 데이터, 상수의 배타적 논리합을 계산합니다. S1 · $\overline{S2}$+$\overline{S1}$ · S2→D S1 | S2 | D 1 | 1 | 0 1 | 0 | 1 0 | 1 | 1 0 | 0 | 0	출력 계통	필요
워드 배타적 논리합(2워드)	XORL @XORL	612	 S1: 연산 데이터 1 하위 CH 번호 S2: 연산 데이터 2 하위 CH 번호 D: 연산결과 출력 하위 CH 번호	2CH의 CH 데이터에 대해 CH 데이터, 상수의 배타적 논리합을 계산합니다. (S1+1, S1) · $\overline{(S2+1, S2)}$+$\overline{(S1+1, S1)}$ · (S2+1, S2)→(D+1, D) S1+1, S1 | S2+1, S2 | D+1, D 1 | 1 | 0 1 | 0 | 1 0 | 1 | 1 0 | 0 | 0	출력 계통	필요
워드 배타적 논리합 부정	XNRW @XNRW	037	 S1: 연산 데이터 1 S2: 연산 데이터 2 D: 연산 결과 출력 CH 번호	CH 데이터 단위에서 CH 데이터와 CH 데이터, 상수의 배타적 논리합 부정을 계산합니다. S1 · S2+$\overline{S1}$ · $\overline{S2}$→D S1 | S2 | D 1 | 1 | 1 1 | 0 | 0 0 | 1 | 0 0 | 0 | 1	출력 계통	필요

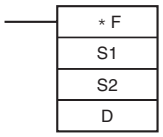
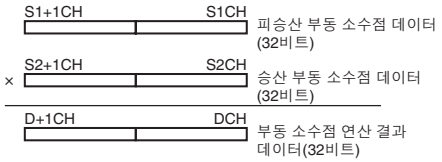
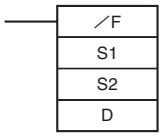
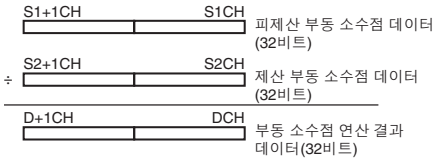
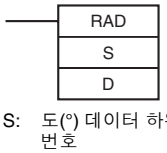
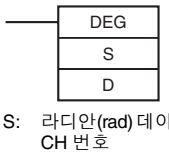
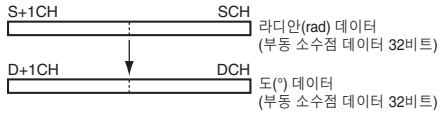
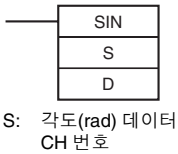
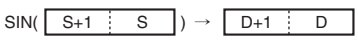
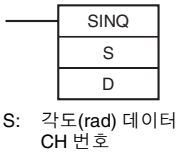
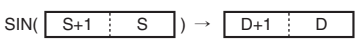
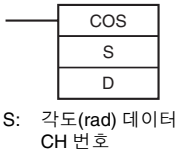
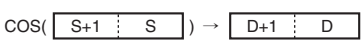
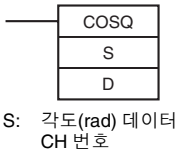
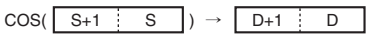
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
워드 배타적 논리합 부정 (2워드)	XNRL @XNRL	613	 S1: 연산 데이터 1 하위 CH 번호 S2: 연산 데이터 2 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	2CH분의 CH 데이터와 CH 데이터, 상수의 배타적 논리합을 계산합니다. (S1+1, S1) · (S2+1, S2)+$\overline{(S1+1, S1)}$ · $\overline{(S2+1, S2)}$→(D+1,D) S1+1, S1 | S2+1, S2 | D+1, D 1 | 1 | 1 1 | 0 | 0 0 | 1 | 0 0 | 0 | 1	출력 계통	필요
비트 반전	COM @COM	029	 D: 반전 데이터 CH 번호	CH 데이터를 비트 반전합니다. $\overline{D}$→D (1→0, 0→1)	출력 계통	필요
비트 2워드 반전	COML @COML	614	 D: 반전 데이터 하위 CH 번호	2CH분의 CH 데이터에서 각 비트를 반전 출력합니다. $\overline{(D+1, D)}$→(D+1, D)	출력 계통	필요

부록-1-12 특수 연산 명령

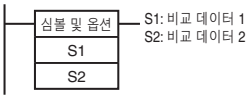
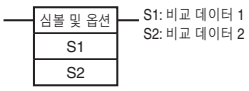
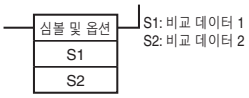
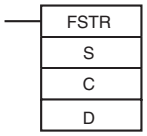
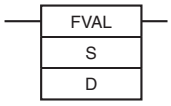
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
BIN 제곱근 연산	ROTB @ROTB	620	 S: 입력 데이터 하위 CH 번호 D: 연산 결과 출력 CH 번호	지정된 BIN 데이터(32비트)를 제곱근 연산하고 결과의 정수부를 출력합니다. 	출력 계통	필요
BCD 제곱근 연산	ROOT @ROOT	072	 S: 연산 데이터 하위 CH 번호 D: 연산 결과 출력 CH 번호	지정된 데이터를 BCD의 2워드 데이터로 제곱근 연산하고 결과의 정수부를 BCD 데이터로 출력합니다. 	출력 계통	필요
숫자 변환	APR @APR	069	 C: 제어 데이터 S: 숫자 변환 입력 데이터 D: 숫자 변환 출력 CH 번호	SIN, COS, 꺾은선 근사 계산을 수행합니다. SIN/COS 계산 지정 시: 각도 데이터(0° ~ 90°)에 대한 SIN 값/COS 값을 계산하고 결과를 소수점 이하 4자리를 나타내는 BCD 데이터로 출력합니다. 꺾은선 근사 계산 시: 지정된 입력 데이터를 지정된 꺾은선 데이터(BIN)에 기초하여 근사값을 계산해서 출력합니다.	출력 계통	필요
부동 소수점 제산(BCD)	FDIV @FDIV	079	 S1: 피제산 데이터 하위 CH 번호 S2: 제산 데이터 하위 CH 번호 D: 제산 결과 출력 하위 CH 번호	가수 7자리와 지수 1자리의 2CH(BCD 8자리) 부동 소수점을 나눕니다. 	출력 계통	필요
비트 카운터	BCNT @BCNT	067	 W: 카운트 CH 수 S: 카운트 하위 CH 번호 D: 카운트 결과 출력 CH 번호	CH 데이터 내에서 "1"인 비트의 총 수를 계산합니다. 	출력 계통	필요

부록-1-13 부동 소수점 변환 및 연산 명령

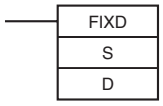
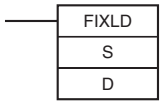
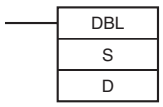
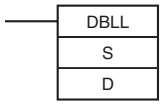
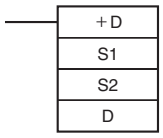
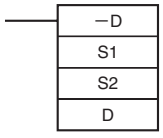
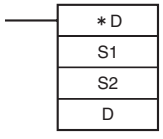
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
부동 소수점 전송<단>	MOVF	469	 S: 전송 데이터 하위 CH 번호 D: 전송 대상 하위 CH 번호	지정된 부동 소수점 데이터(32비트)를 지정 채널로 전송합니다. 	출력 계통	필요
부동 소수점 →16비트 BIN 변환	FIX @FIX	450	 S: 부동 소수점 데이터 하위 CH 번호 D: 변환 결과 출력 CH 번호	지정된 부동 소수점 32비트 데이터를 BIN 16비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다. 	출력 계통	필요
부동 소수점 →32비트 BIN 변환	FIXL @FIXL	451	 S: 부동 소수점 데이터 하위 CH 번호 D: 변환 결과 출력 CH 번호	지정된 부동 소수점 32비트 데이터를 부호 있는 BIN 32비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다. 	출력 계통	필요
16비트 BIN→ 부동 소수점 변환	FLT @FLT	452	 S: BIN 데이터 D: 변환 결과 출력 하위 CH 번호	지정된 부호 있는 BIN 16비트 데이터를 부동 소수점 32비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다. 	출력 계통	필요
32비트 BIN→ 부동 소수점 변환	FLTL @FLTL	453	 S: BIN 2워드 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	지정된 부호 있는 BIN 32비트 데이터를 부동 소수점 32비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다. 	출력 계통	필요
부동 소수점 가산	+F @+F	454	 S1: 피가산 부동 소수점 데이터 하위 CH 번호 S2: 가산 부동 소수점 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)를 가산하여 결과를 지정된 채널에 출력합니다. 	출력 계통	필요
부동 소수점 감산	−F @−F	455	 S1: 피감산 부동 소수점 데이터 하위 CH 번호 S2: 감산 부동 소수점 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)를 감산하여 결과를 지정된 채널에 출력합니다. 	출력 계통	필요

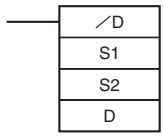
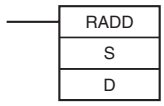
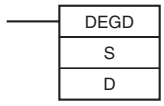
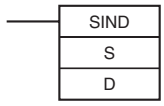
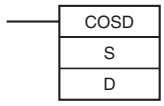
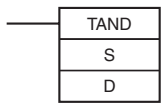
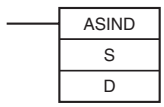
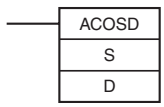
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
부동 소수점 승산	* F @ * F	456	 S1: 피승산 부동 소수점 데이터 하위 CH 번호 S2: 승산 부동 소수점 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)를 곱하여 결과를 지정된 채널에 출력합니다. 	출력 계통	필요
부동 소수점 제산	/F @/F	457	 S1: 피제산 부동 소수점 데이터 하위 CH 번호 S2: 제산 부동 소수점 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)를 나눠서 결과를 지정된 채널에 출력합니다. 	출력 계통	필요
각도→라디안 변환	RAD @RAD	458	 S: 도(°) 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)를 도(°) 단위에서 라디안(rad) 단위로 변환하여 결과를 지정된 채널에 출력합니다. 	출력 계통	필요
라디안→각도 변환	DEG @DEG	459	 S: 라디안(rad) 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)를 라디안(rad) 단위에서 도(°) 단위로 변환하여 결과를 지정된 채널에 출력합니다. 	출력 계통	필요
SIN 연산	SIN @SIN	460	 S: 각도(rad) 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 SIN(사인) 값을 연산하여 결과를 지정된 채널에 출력합니다. 	출력 계통	필요
고속 SIN 연산	SINQ @SINQ	475	 S: 각도(rad) 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 SIN(사인) 값을 연산하여 결과를 고속으로 지정된 채널에 출력합니다. 	출력 계통	필요
COS 연산	COS @COS	461	 S: 각도(rad) 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 COS(코사인) 값을 연산하여 결과를 지정된 채널에 출력합니다. 	출력 계통	필요
고속 COS 연산	COSQ @COSQ	476	 S: 각도(rad) 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 COS(코사인) 값을 연산하여 결과를 고속으로 지정된 채널에 출력합니다. 	출력 계통	필요

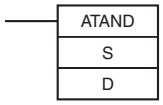
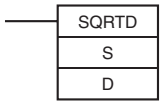
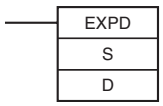
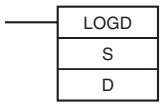
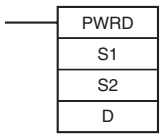
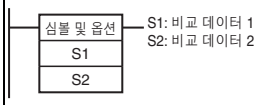
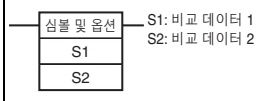
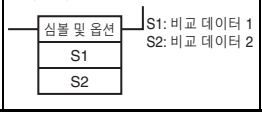
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
TAN 연산	TAN @TAN	462	TAN S D S: 각도(rad) 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 TAN(탄젠트) 값을 연산하여 결과를 지정된 채널에 출력합니다. TAN(S+1 S) → D+1 D	출력 계통	필요
고속 TAN 연산	TANQ @TANQ	477	TANQ S D S: 각도(rad) 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)로 표시되는 각도(라디안 단위)의 TAN(탄젠트) 값을 연산하여 결과를 고속으로 지정된 채널에 출력합니다. TAN(S+1 S) → D+1 D	출력 계통	필요
SIN^{-1} 연산	ASIN @ASIN	463	ASIN S D S: SIN(사인) 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)로 표시되는 SIN(사인) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다. SIN^{-1}(S+1 S) → D+1 D	출력 계통	필요
COS^{-1} 연산	ACOS @ACOS	464	ACOS S D S: COS(코사인) 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)로 표시되는 COS(코사인) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다. COS^{-1}(S+1 S) → D+1 D	출력 계통	필요
TAN^{-1} 연산	ATAN @ATAN	465	ATAN S D S: TAN(탄젠트) 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)로 표시되는 TAN(탄젠트) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다. TAN^{-1}(S+1 S) → D+1 D	출력 계통	필요
제곱근 연산	SQRT @SQRT	466	SQRT S D S: 입력 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)의 제곱근(√)을 연산하여 결과를 지정된 채널에 출력합니다. √ S+1 S → D+1 D	출력 계통	필요
지수 연산	EXP @EXP	467	EXP S D S: 입력 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)의 지수(e를 밑수로 했을 때의 지수)를 연산하여 결과를 지정된 채널에 출력합니다. e S+1 S → D+1 D	출력 계통	필요
로그 연산	LOG @LOG	468	LOG S D S: 입력 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)의 자연로그(e를 밑수로 했을 때의 로그)를 연산하여 결과를 지정된 채널에 출력합니다. log_e S+1 S → D+1 D	출력 계통	필요
거듭제곱 연산	PWR @PWR	840	PWR S1 S2 D S1: 피거듭제곱 데이터 하위 CH 번호 S2: 거듭제곱 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 부동 소수점 데이터(32비트)의 거듭제곱 연산을 수행하여 결과를 지정된 채널에 출력합니다. 거듭제곱 데이터 S2+1 S2 S1+1 S → D+1 D 피거듭제곱 데이터	출력 계통	필요

명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
단정도 부동 소수점 데이터 비교	LD, AND, OR + =F, <>F, <F, <=F, >F, >=F	329 (=F) 330 (<>F) 331 (<F) 332 (<=F) 333 (>F) 334 (>=F)	LD(로드) 연결형  AND(직렬) 연결형  OR(병렬) 연결형  	지정된 단정도 부동 소수점 데이터(32비트) 또는 상수를 비교하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다. 연결 형식에는 LD(로드) 연결, AND(직렬) 연결, OR(병렬) 연결의 3종류가 있습니다.	LD형: 논리 시작 ADD, OR형: 다음 단계 연결형	LD형 : 필요 없음 AND, OR형 : 필요
부동 소수점 <단>→문자열 변환	FSTR @FSTR	448	 S: 부동 소수점 데이터 하위 CH 번호 C: 제어 데이터 D: 변환 결과 출력 대상 CH 번호	지정된 단정도 부동 소수점 데이터(32비트)를 소수점 형식 또는 지수 형식으로 표시하고 그것을 문자열 데이터(ASCII 코드)로 변환합니다.	출력 계통	필요
문자열→부동 소수점<단> 변환	FVAL @FVAL	449	 S: 문자열 데이터 저장 채널 번호 D: 변환 결과 출력 채널 번호	지정된 소수점 형식 또는 지수 형식으로 나타낸 문자열(ASCII 코드)을 단정도 부동 소수점 데이터(32비트)로 변환하여 지정된 채널에 결과를 출력합니다.	출력 계통	필요

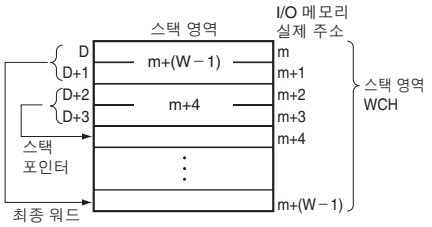
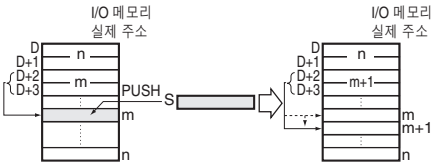
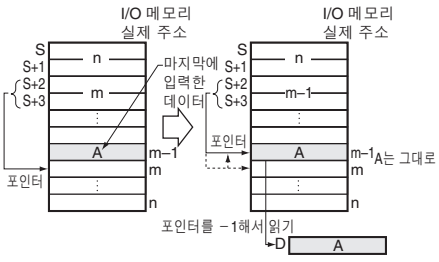
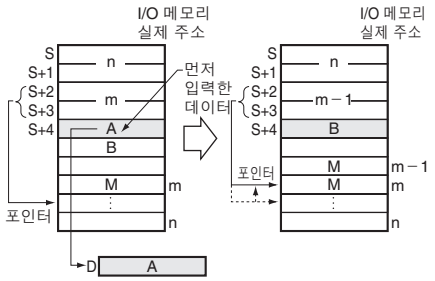
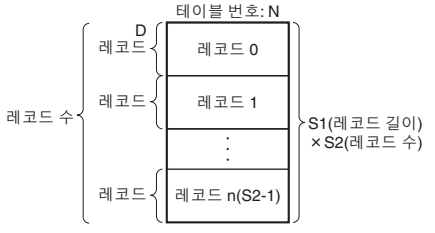
부록-1-14 배정도 부동 소수점 변환 및 연산 명령

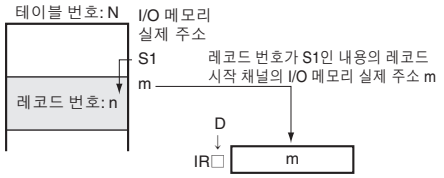
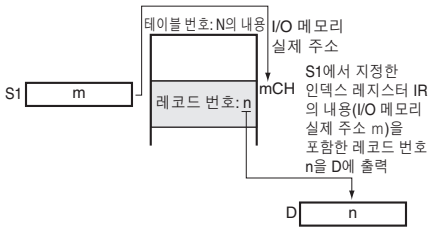
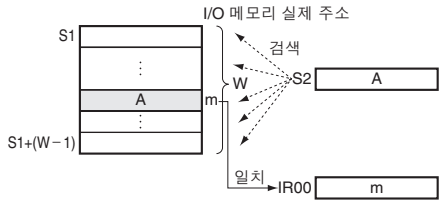
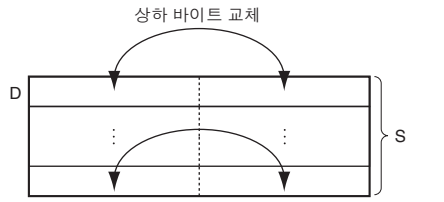
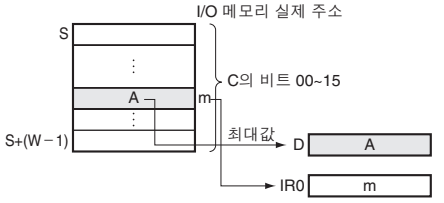
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
부동 소수점 →16비트 BIN 변환 <배>	FIXD @FIXD	841	 S: 부동 소수점 데이터 하위 CH 번호 D: 변환 결과 출력 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)를 부호 있는 BIN 16비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요
부동 소수점 →32비트 BIN 변환 <배>	FIXLD @FIXLD	842	 S: 부동 소수점 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)를 부호 있는 BIN 32비트 데이터로 변환하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요
16비트 BIN→ 부동 소수점 변환 <배>	DBL @DBL	843	 S: BIN 데이터 D: 변환 결과 출력 하위 CH 번호	지정된 부호 있는 BIN 16비트 데이터를 배정도 부동 소수점 데이터(64비트)로 변환하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요
32비트 BIN→ 부동 소수점 변환 <배>	DBLL @DBLL	844	 S: BIN 2워드 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	지정된 부호 있는 BIN 32비트 데이터를 배정도 부동 소수점 데이터(64비트)로 변환하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요
부동 소수점 가산<배>	+D @+D	845	 S1: 피가산 부동 소수점 데이터 하위 CH 번호 S2: 가산 부동 소수점 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)를 더하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요
부동 소수점 감산<배>	−D @−D	846	 S1: 피감산 부동 소수점 데이터 하위 CH 번호 S2: 감산 부동 소수점 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)를 빼서 결과를 지정된 채널에 출력합니다.	출력 계통	필요
부동 소수점 승산<배>	* D @ *D	847	 S1: 피승산 부동 소수점 데이터 하위 CH 번호 S2: 승산 부동 소수점 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)를 곱해서 결과를 지정된 채널에 출력합니다.	출력 계통	필요

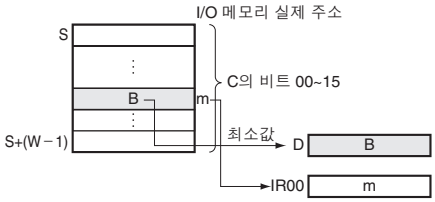
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
부동 소수점 제산<배>	/D @/D	848	 S1: 피제산 부동 소수점 데이터 하위 CH 번호 S2: 제산 부동 소수점 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)를 나눠서 결과를 지정된 채널에 출력합니다.	출력 계통	필요
각도→라디안 변환<배>	RADD @RADD	849	 S: 도(°) 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)를 도(°) 단위에서 라디안(rad) 단위로 변환하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요
라디안→각도 변환<배>	DEGD @DEGD	850	 S: 라디안(rad) 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)를 라디안(rad) 단위에서 도(°) 단위로 변환하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요
SIN 연산 <배>	SIND @SIND	851	 S: 각도(rad) 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 각도(라디안 단위)의 SIN(사인) 값을 연산하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요
COS 연산 <배>	COSD @COSD	852	 S: 각도(rad) 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 각도(라디안 단위)의 COS(코사인) 값을 연산하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요
TAN 연산 <배>	TAND @TAND	853	 S: 각도(rad) 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 각도(라디안 단위)의 TAN(탄젠트) 값을 연산하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요
SIN^{-1} 연산 <배>	ASIND @ASIND	854	 S: SIN(사인) 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 SIN(사인) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요
COS^{-1} 연산 <배>	ACOSD @ACOSD	855	 S: COS(코사인) 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 COS(코사인) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요

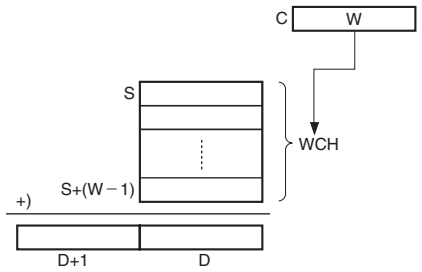
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
TAN^{-1} 연산 <배>	ATAND @ATAND	856	 S: TAN(탄젠트) 데이터 하위 CH 번호 D: 변환 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)로 표시되는 TAN(탄젠트) 값에서 각도(라디안 단위)를 연산하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요
제곱근 연산 <배>	SQRTD @SQRTD	857	 S: 입력 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)의 제곱근($\sqrt{}$)을 연산하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요
지수 연산 <배>	EXPD @EXPD	858	 S: 입력 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)의 지수(e를 밑수로 했을 때의 지수)를 연산하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요
로그 연산 <배>	LOGD @LOGD	859	 S: 입력 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)의 자연로그(e를 밑수로 했을 때의 로그)를 연산하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요
거듭제곱 연산 <배>	PWRD @PWRD	860	 S1: 피거듭제곱 데이터 하위 CH 번호 S2: 거듭제곱 데이터 하위 CH 번호 D: 연산 결과 출력 하위 CH 번호	지정된 배정도 부동 소수점 데이터(64비트)의 거듭제곱 연산을 수행하여 결과를 지정된 채널에 출력합니다.	출력 계통	필요
배정도 부동 소수점 데이터 비교	LD, AND, OR + =D, <>D, <D, <=D, >D, >=D	335 (=D) 336 (<>D) 337 (<D) 338 (<=D) 339 (>D) 340 (>=D)	LD(로드) 연결형  AND(직렬) 연결형  OR(병렬) 연결형 	지정된 배정도 부동 소수점 데이터(64비트)를 비교하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다. • 연결 형식에는 LD(로드) 연결, AND(직렬) 연결, OR(병렬) 연결의 3종류가 있습니다. • 상수와는 비교할 수 없습니다.	LD형: 논리 시작 AND, OR형: 다음 단계 연결형	LD형 : 필요 없음 AND, OR형 : 필요

부록-1-15 테이블 데이터 처리 명령

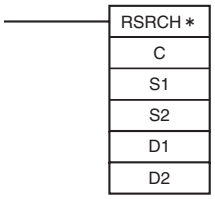
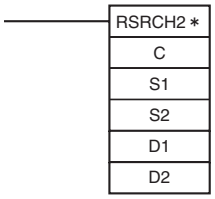
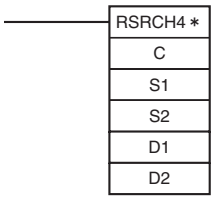
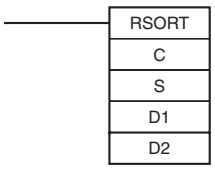
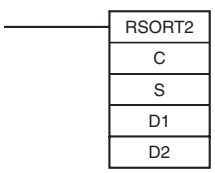
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
스택 영역 설정	SSET @SSET	630	SSET D W D: 스택 영역 하위 CH 번호 W: 영역 CH 번호	지정된 CH에서 지정된 CH 수의 스택 영역을 설정하고, 동시에 데이터 저장 영역을 0으로 클리어합니다. 	출력 계통	필요
스택 데이터 저장	PUSH @PUSH	632	PUSH D S D: 스택 영역 하위 CH 번호 S: 저장 데이터 CH 번호	지정된 스택 영역에 데이터를 저장합니다. 동시에 스택 포인터의 내용을 1 증가시킵니다. 	출력 계통	필요
후입선출	LIFO @LIFO	634	LIFO S D S: 스택 영역 하위 CH 번호 D: 출력 대상 CH 번호	지정된 스택 영역의 스택 포인터의 데이터를 1 감소시키고, 그 주소에서 데이터(마지막에 저장한 데이터)를 읽습니다. 	출력 계통	필요
선입선출	FIFO @FIFO	633	FIFO S D S: 스택 영역 하위 CH 번호 D: 출력 대상 CH 번호	지정된 스택 영역에서 처음 저장된 데이터를 읽습니다. 동시에 스택 포인터의 내용을 1 감소시키고 데이터 저장 영역 선두+1~스택 포인터가 가리키는 위치의 데이터를 하위 CH 측으로 1CH 이동합니다. 	출력 계통	필요
테이블 영역 선언	DIM @DIM	631	DIM N S1 S2 D N: 테이블 번호 S1: 레코드 길이 S2: 레코드 수 D: 테이블 영역 하위 CH 번호	테이블 영역을 등록합니다. 1개의 테이블은 레코드 길이×레코드 수로 이루어지며 최대 16테이블입니다. 	출력 계통	필요

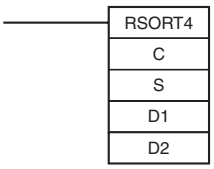
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
레코드 위치 설정	SETR @SETR	635	SETR N S1 D N: 테이블 번호 S1: 레코드 번호 D: 출력 대상 인덱스 레지스터 번호	지정된 레코드의 위치(레코드 선두의 I/O 메모리 실제 주소)를 지정된 인덱스 레지스터에 저장합니다. 	출력 계통	필요
레코드 위치 읽기	GETR @GETR	636	GETR N S1 D S1: 테이블 번호 S2: 인덱스 레지스터 번호 D: 레코드 위치 출력 CH 번호	지정된 인덱스 레지스터의 값(I/O 메모리 실제 주소)이 포함된 레코드의 번호를 출력합니다. 	출력 계통	필요
데이터 검색	SRCH @SRCH	181	SRCH W S1 S2 W: 테이블 길이 지정 데이터 S1: 테이블 하위 CH 번호 S2: 검색 데이터	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. 	출력 계통	필요
바이트 스와핑	SWAP @SWAP	637	SWAP S D S: 테이블 길이 지정 데이터 D: 테이블 하위 CH 번호	지정된 테이블 데이터의 지정된 CH 수의 상위 바이트와 하위 바이트를 교체합니다. 	출력 계통	필요
최대값 검색	MAX @MAX	182	MAX C S D C: 제어 데이터 S: 테이블 하위 CH 번호 D: 최대값 출력 대상 CH 번호	지정된 테이블 내의 최대값을 검색합니다. 	출력 계통	필요

명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
2워드 데이터 최대값 검색	MAXL @MAXL	174	MAXL C S D C: 제어 데이터 S: 테이블 하위 CH 번호 D: 최대값 출력 대상 하위 CH 번호	S로 지정된 테이블 하위 CH 번호부터 C로 지정된 데이터 수만큼 2CH 데이터의 테이블 데이터로 간주하고, 그 중에서 최대값을 검색하여 D+1,D에 출력합니다.	출력 계통	필요
단정도 부동 소수점 데이터 최대값 검색	MAXF @MAXF	176	MAXF C S D C: 제어 데이터 S: 테이블 하위 CH 번호 D: 최대값 출력 대상 하위 CH 번호	S로 지정된 테이블 하위 CH 번호부터 C로 지정된 데이터 수만큼 2CH 데이터의 단정도 부동 소수점 데이터로 간주하고, 그 중에서 최대값을 검색하여 D+1,D에 출력합니다.	출력 계통	필요
배정도 부동 소수점 데이터 최대값 검색	MAXD @MAXD	178	MAXD C S D C: 제어 데이터 S: 테이블 하위 CH 번호 D: 최대값 출력 대상 하위 CH 번호	S로 지정된 테이블 하위 CH 번호부터 C로 지정된 데이터 수만큼 2CH 데이터의 배정도 부동 소수점 데이터로 간주하고, 그 중에서 최대값을 검색하여 D+1,D에 출력합니다.	출력 계통	필요
최소값 검색	MIN @MIN	183	MIN C S D C: 제어 데이터 S: 테이블 하위 CH 번호 D: 최소값 출력 대상 CH 번호	지정된 테이블 내의 최소값을 검색합니다. 	출력 계통	필요
2워드 데이터 최소값 검색	MINL @MINL	175	MINL C S D C: 제어 데이터 S: 테이블 하위 CH 번호 D: 최소값 출력 대상 하위 CH 번호	S로 지정된 테이블 하위 CH 번호부터 C로 지정된 데이터 수만큼 2CH 데이터의 테이블 데이터로 간주하고, 그 중에서 최소값을 검색하여 D+1,D에 출력합니다.	출력 계통	필요
단정도 부동 소수점 데이터 최소값 검색	MINF @MINF	177	MINF C S D C: 제어 데이터 S: 테이블 하위 CH 번호 D: 최소값 출력 대상 하위 CH 번호	S로 지정된 테이블 하위 CH 번호부터 C로 지정된 데이터 수만큼 2CH 데이터의 단정도 부동 소수점 데이터로 간주하고, 그 중에서 최소값을 검색하여 D+1,D에 출력합니다.	출력 계통	필요
배정도 부동 소수점 데이터 최소값 검색	MIND @MIND	179	MIND C S D C: 제어 데이터 S: 테이블 하위 CH 번호 D: 최소값 출력 대상 하위 CH 번호	S로 지정된 테이블 하위 CH 번호부터 C로 지정된 데이터 수만큼 2CH 데이터의 배정도 부동 소수점 데이터로 간주하고, 그 중에서 최소값을 검색하여 D+1,D에 출력합니다.	출력 계통	필요

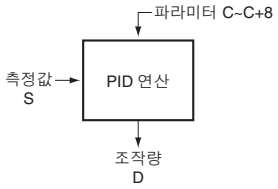
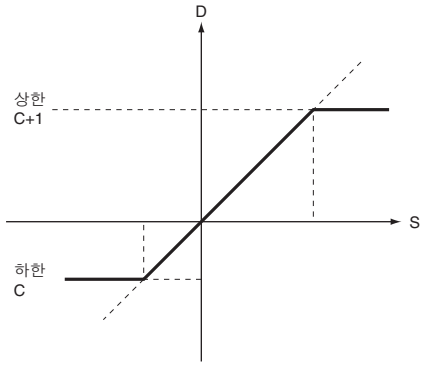
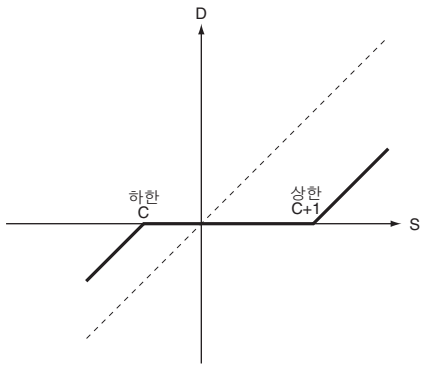
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
SUM 값 산출	SUM @SUM	184	SUM C S D C: 제어 데이터 S: 테이블 하위 CH 번호 D: SUM 값 출력 대상 하위 CH 번호	지정된 테이블 내의 SUM 값을 계산하고 2CH로 출력합니다. 	출력 계통	필요
스택 데이터 수 출력	SNUM @SNUM	638	SNUM S D S: 스택 영역 하위 CH 번호 D: 출력 대상 CH 번호	지정된 스택 영역 내의 스택 데이터 수(CH 수)를 계산합니다.	출력 계통	필요
스택 데이터 참조	SREAD @SREAD	639	SREAD S C D S: 스택 영역 하위 CH 번호 C: 참조 위치(오프셋 값 α) D: 출력 대상 CH 번호	지정된 스택 영역에서 지정된 데이터(현재 스택 포인터가 있는 위치에서 지정된 개수 만큼 앞에 있는 데이터)를 읽습니다.	출력 계통	필요
스택 데이터 갱신	SWRIT @SWRIT	640	SWRIT D C S D: 스택 영역 하위 CH 번호 C: 갱신 위치(오프셋 값 α) S: 쓰기 데이터	지정된 스택 영역에서 지정된 데이터(현재 스택 포인터가 있는 위치에서 지정된 개수 만큼 앞에 있는 데이터)의 위치에 다른 데이터를 덮어 씁니다.	출력 계통	필요
스택 데이터 삽입	SINS @SINS	641	SINS D C S D: 스택 영역 하위 CH 번호 C: 삽입 위치(오프셋 값 α) S: 삽입 데이터	지정된 스택 영역에서 지정된 데이터 위치(현재 스택 포인터가 있는 위치에서 지정된 개수 만큼 앞에 있는 위치)에 다른 데이터를 삽입합니다.	출력 계통	필요
스택 데이터 삭제	SDEL @SDEL	642	SDEL S C D S: 스택 영역 하위 CH 번호 C: 삭제 위치(오프셋 값 α) D: 조작량 출력 CH 번호	지정된 스택 영역에서 지정된 데이터 위치(현재 스택 포인터가 있는 위치에서 지정된 개수 만큼 앞에 있는 위치)의 데이터를 삭제합니다.	출력 계통	필요
FCS 값 산출	FCS @FCS	180	FCS C S D C: 제어 데이터 S: 테이블 하위 CH 번호 D: FCS 값 저장 시작 CH 번호	지정된 테이블 내의 FCS 값을 계산하고 ASCII 코드 데이터로 변환하여 출력합니다.	출력 계통	필요

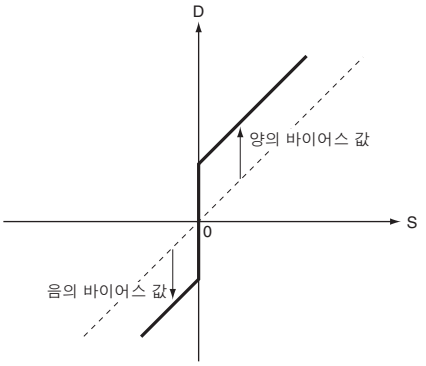
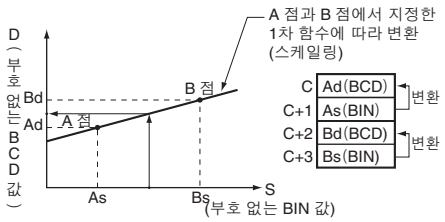
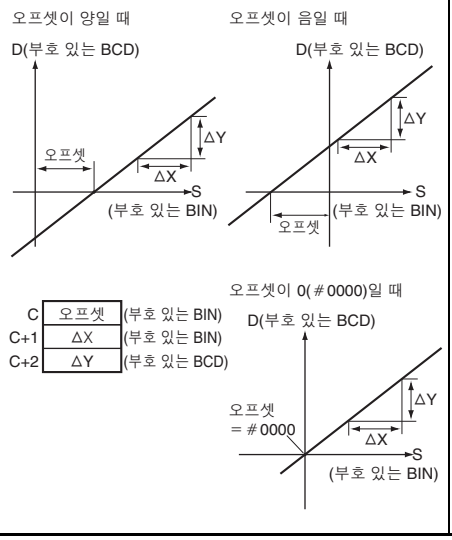
부록-1-16 추적 명령

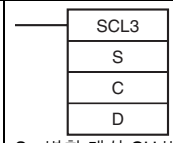
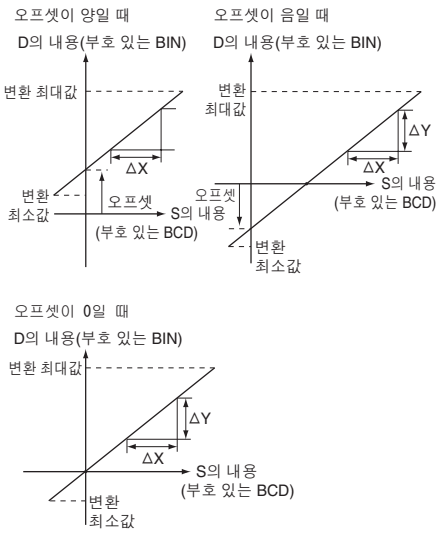
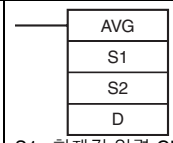
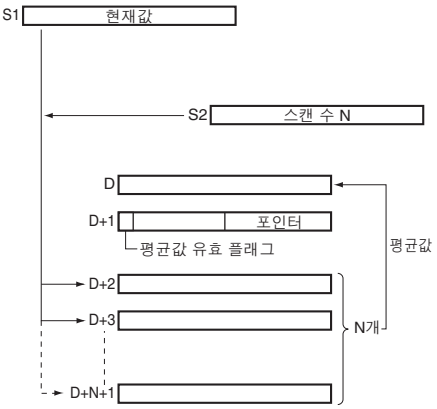
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
부호 없는 1W 레코드 검색	RSRCH + <, < =, =, >, > = @RSRCH + <, < =, =, >, > =	360 (<) 361 (< =) 362 (=) 363 (>) 364 (>=)	 C: 제어 데이터 S1: 테이블 데이터 S2: 검색 데이터 D1: 출력 대상 D2: 전송 대상 인덱스 레지스터	S1로 지정한 테이블에서 S2로 지정한 데이터(1워드)를 검색하고, 일치하는 레코드가 있으면 D1 이후에 레코드 번호와 데이터를 출력합니다. D2에 인덱스 레지스터를 지정한 경우는 일치하는 레코드의 주소를 인덱스 레지스터에 출력합니다. 인덱스 레지스터 출력을 사용하지 않는 경우는 D2에 #00000000을 설정하십시오.	출력 계통	필요
부호 없는 2W 레코드 검색	RSRCH2 + <, < =, =, >, > = @RSRCH2 + <, < =, =, >, > =	370 (<) 371 (< =) 372 (=) 373 (>) 374 (>=)	 C: 제어 데이터 S1: 테이블 데이터 S2: 검색 데이터 D1: 출력 대상 D2: 전송 대상 인덱스 레지스터	S1로 지정한 테이블에서 S2로 지정한 데이터(2워드)를 검색하고, 일치하는 레코드가 있으면 D1 이후에 레코드 번호와 데이터를 출력합니다. D2에 인덱스 레지스터를 지정한 경우는 일치하는 레코드의 주소를 인덱스 레지스터에 출력합니다. 인덱스 레지스터 출력을 사용하지 않는 경우는 D2에 #00000000을 설정하십시오.	출력 계통	필요
부호 없는 4W 레코드 검색	RSRCH4 + <, < =, =, >, > = @RSRCH4 + <, < =, =, >, > =	380 (<) 381 (< =) 382 (=) 383 (>) 384 (>=)	 C: 제어 데이터 S1: 테이블 데이터 S2: 검색 데이터 D1: 출력 대상 D2: 전송 대상 인덱스 레지스터	S1로 지정한 테이블에서 S2로 지정한 데이터(4워드)를 검색하고, 일치하는 레코드가 있으면 D1 이후에 레코드 번호와 데이터를 출력합니다. D2에 인덱스 레지스터를 지정한 경우는 일치하는 레코드의 주소를 인덱스 레지스터에 출력합니다. 인덱스 레지스터 출력을 사용하지 않는 경우는 D2에 #00000000을 설정하십시오.	출력 계통	필요
부호 없는 1W 레코드 정렬	RSORT @RSORT	203	 C: 제어 데이터 S: 테이블 데이터 D1: 결과 출력 대상 D2: 전송 대상 인덱스 레지스터	S로 지정한 테이블 내에 있는 레코드(1워드)를 제어 데이터에 따라 정렬합니다.	출력 계통	필요
부호 없는 2W 레코드 정렬	RSORT2 @RSORT2	204	 C: 제어 데이터 S: 테이블 데이터 D1: 결과 출력 대상 D2: 전송 대상 인덱스 레지스터	S로 지정한 테이블 내에 있는 레코드(2워드)를 제어 데이터에 따라 정렬합니다.	출력 계통	필요

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
부호 없는 4W 레코드 정렬	RSORT4 @RSORT4	205	 C: 제어 데이터 S: 테이블 데이터 D1: 결과 출력 대상 D2: 전송 대상 인덱스 레지스터	S로 지정한 테이블 내에 있는 레코드(4워드)를 제어 데이터에 따라 정렬합니다.	출력 계통	필요

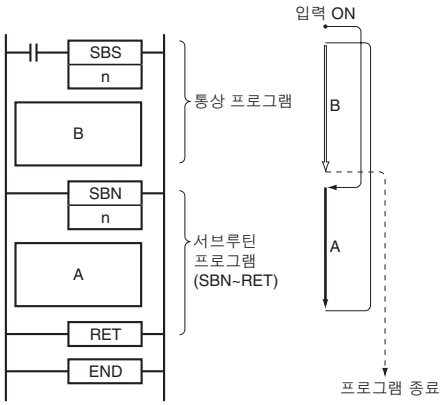
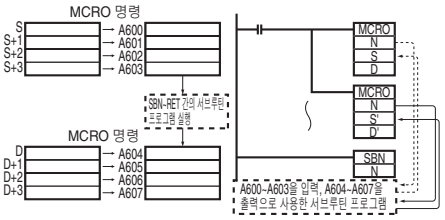
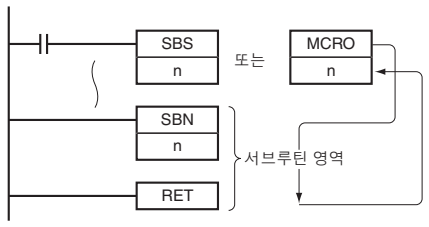
부록-1-17 데이터 제어 명령

명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
PID 연산	PID	190	PID S C D S: 측정값 입력 CH 번호 C: PID 파라미터 저장 하위 CH 번호 D: 조작량 출력 CH 번호	지정된 파라미터에 따라 PID 연산(목표값 필터형 2 자유도 PID)을 수행하여 조작량을 출력합니다. 	출력 계통	필요
자동 조정 가능한 PID 연산	PIDAT	191	PIDAT S C D S: 측정값 입력 CH 번호 C: PID 파라미터 저장 하위 CH 번호 D: 조작량 출력 CH 번호	지정된 파라미터에 따라 PID 연산을 수행합니다. PID 상수의 자동 조정(AT)을 실행할 수 있습니다.	출력 계통	필요
상하한 리미트 제어	LMT @LMT	680	LMT S C D S: 입력 CH 번호 C: 리미트 데이터 하위 CH 번호 D: 출력 CH 번호	입력 데이터가 상하한 리미트 데이터의 범위 안인지 여부에 따라 출력 데이터를 제어합니다. 	출력 계통	필요
데드 밴드 제어	BAND @BAND	681	BAND S C D S: 입력 CH 번호 C: 상하한 데이터 하위 CH 번호 D: 출력 CH 번호	입력 데이터가 불감대(Dead band)의 범위 안인지 여부에 따라 출력 데이터를 제어합니다. 	출력 계통	필요

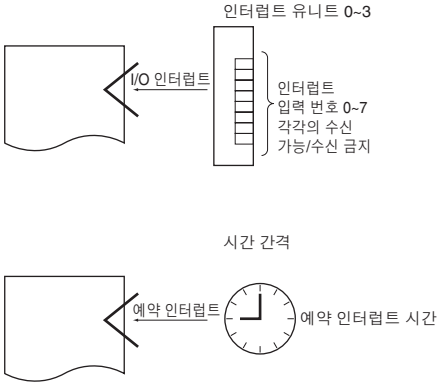
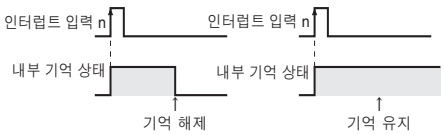
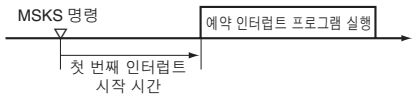
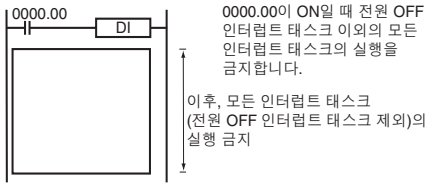
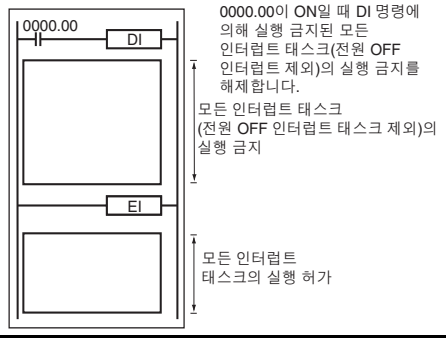
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
데드 존 제어	ZONE @ZONE	682	ZONE S C D S: 입력 CH 번호 C: 바이어스 데이터 하위 CH 번호 D: 출력 CH 번호	지정한 바이어스 값을 입력 데이터에 더하여 출력합니다. 	출력 계통	필요
시분할 비례 출력	TPO	685	TPO S C D S: 입력 듀티비 또는 조작량 저장 채널 번호 C: 파라미터 저장 하위 채널 번호 R: 펄스 출력 릴레이 번호	지정된 채널 내의 듀티비 또는 조작량을 입력하고 지정된 파라미터에 따라 듀티비를 시분할 비례 출력으로 변환하여 지정한 접점에 펄스 출력합니다.	출력 계통	필요
스케일링	SCL @SCL	194	SCL S C D S: 변환 대상 CH 번호 C: 파라미터 저장 하위 CH 번호 D: 변환 결과 저장 CH 번호	지정한 1차 함수에 따라 부호 없는 BIN 데이터를 부호 없는 BCD 데이터로 스케일링(변환)합니다. 	출력 계통	필요
스케일링 2	SCL2 @SCL2	486	SCL2 S C D S: 변환 대상 CH 번호 C: 파라미터 저장 하위 CH 번호 D: 변환 결과 저장 CH 번호	지정한 오프셋이 있는 1차 함수에 따라 부호 있는 BIN 데이터를 부호 있는 BCD 데이터로 스케일링(변환)합니다. 	출력 계통	필요

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
스케일링 3	SCL3 @SCL3	487	 S: 변환 대상 CH 번호 C: 파라미터 저장 하위 CH 번호 D: 변환 결과 저장 CH 번호	지정한 오프셋이 있는 1차 함수에 따라 부호 있는 BCD 데이터를 부호 있는 BIN 데이터로 스케일링(변환)합니다. 	출력 계통	필요
데이터 평균화	AVG	195	 S1: 현재값 입력 CH 번호 대상 CH S2: 평균값 계산 사이클 횟수 D: 평균값 저장 하위 CH 번호	CH 데이터의 지정한 사이클 횟수 분량의 평균값을 BIN으로 계산합니다. 	출력 계통	필요

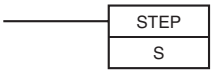
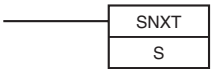
부록-1-18 서브루틴 명령

명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
서브루틴 호출	SBS @SBS	091	SBS N N: 서브루틴 번호	지정된 번호의 서브루틴을 호출하는 프로그램을 실행합니다. 	출력 계통	필요
매크로	MCRO @MCRO	099	MCRO N S D N: 서브루틴 번호 S: 인수 데이터 하위 CH 번호 D: 반환값 데이터 하위 CH 번호	인수가 있는 서브루틴 호출입니다. SBS 명령과 달리, S로 지정한 인수 데이터와 D로 지정한 반환값 데이터를 사용하여 서브루틴 영역 프로그램과 데이터를 주고받을 수 있습니다. 	출력 계통	필요
서브루틴 엔트리	SBN	092	SBN N N: 서브루틴 번호	지정된 번호의 서브루틴의 선두를 나타냅니다. 	출력 계통	필요 없음
서브루틴 반환	RET	093	RET	서브루틴의 종료를 나타냅니다.	출력 계통	필요 없음
글로벌 서브루틴 호출	GSBS @GSBS	750	GSBS N N: 글로벌 서브루틴 번호	지정된 번호의 글로벌 서브루틴을 호출하고 프로그램을 실행합니다.	출력 계통	필요
글로벌 서브루틴 엔트리	GSBN	751	GSBN N N: 글로벌 서브루틴 번호	지정된 번호의 글로벌 서브루틴의 선두를 나타냅니다.	출력 계통	필요 없음
글로벌 서브루틴 반환	GRET	752	GRET	글로벌 서브루틴의 종료를 나타냅니다.	출력 계통	필요 없음

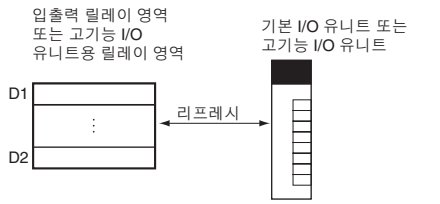
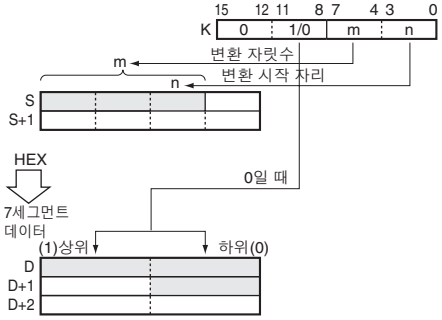
부록-1-19 인터럽트 제어 명령

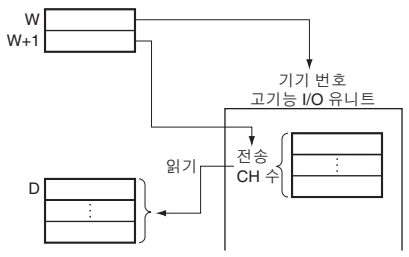
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
인터럽트 마스크 세트	MSKS @MSKS	690	MSKS N C N: 인터럽트 번호 C: 제어 데이터	I/O 인터럽트 처리 또는 예약 인터럽트 처리를 지정합니다. I/O 인터럽트 태스크 및 예약 인터럽트 태스크의 실행은 초기 상태(전원 투입 시)에서 마스크(금지)가 실행됩니다. 이 명령을 실행하면 I/O 인터럽트 태스크의 수신 가능/금지의 제어 또는 예약 인터럽트 태스크의 예약 인터럽트 시간 설정이 가능해집니다. 	출력 계통	필요
인터럽트 마스크 읽기	MSKR @MSKR	692	MSKR N D N: 인터럽트 번호 D: 읽은 결과 출력 CH 번호	MSKS 명령에 의해 지정된 인터럽트 제어 상태를 읽습니다.	출력 계통	필요
인터럽트 해제	CLI @CLI	691	CLI N C N: 인터럽트 번호 C: 제어 데이터	I/O 인터럽트 입력 기억 해제/유지 지정 또는 예약 인터럽트의 첫 번째 인터럽트 시작 시간 지정을 수행합니다. N=0~3일 때  N=4~5일 때 	출력 계통	필요
인터럽트 태스크 실행 금지	DI @DI	693	DI	모든 인터럽트 태스크(전원 OFF 인터럽트 태스크 제외)의 실행을 금지합니다. 	출력 계통	필요
인터럽트 태스크 실행 금지 해제	EI	694	EI	모든 인터럽트 태스크(전원 OFF 인터럽트 태스크 제외)의 실행 금지를 해제합니다. 	출력 계통	필요 없음

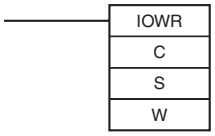
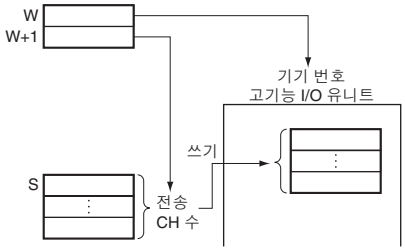
부록-1-20 공정 진행 제어 명령

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
스텝 래더 영역 정의	STEP	008	 S: 공정 번호	배치 위치 및 공정 번호 지정 유무에 따라 다음 2가지 동작을 수행합니다. 1) 지정된 공정의 시작 2) 스텝 래더 영역 전체 종료 스텝 래더 영역은 STEP 명령(공정 번호 지정)~STEP 명령(공정 번호 없음)까지의 영역입니다.	출력 계통	필요
스텝 래더 영역 진행	SNXT	009	 S: 공정 번호	배치 위치에 따라 다음 3가지 동작을 수행합니다. 1) 스텝 래더 영역으로 진행 2) 다음 공정 번호로 진행 3) 스텝 래더 영역의 종료로 진행	출력 계통	필요

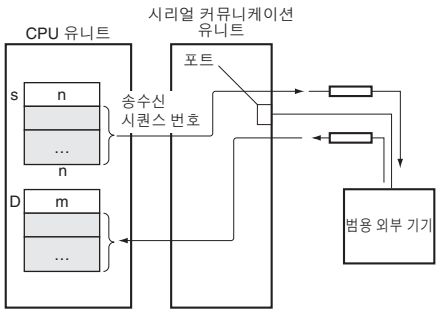
부록-1-21 I/O 유니트용 명령

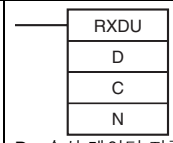
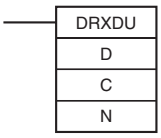
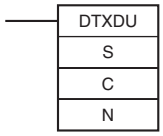
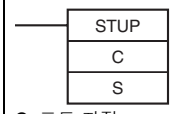
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
I/O 리프레시	IORF @IORF	097	IORF D1 D2 D1: 리프레시 하위 CH 번호 D2: 리프레시 상위 CH 번호	지정된 리프레시 하위 CH 번호부터 지정된 리프레시 상위 CH 번호까지의 I/O 채널 데이터를 리프레시합니다. 리프레시 대상은 CPU 장치 및 증설 장치에 장착된 기본 I/O 유니트와 고기능 I/O 유니트입니다. 	출력 계통	필요
고기능 I/O 유니트 매회 리프레시	FIORF @FIORF	225	FIORF N N: 기기 번호	지정된 고기능 I/O 유니트와의 I/O 리프레시를 매회 실행합니다.	출력 계통	필요
CPU 고기능 유니트 매회 I/O 리프레시	DLNK @DLNK	226	DLNK N N: 유니트 번호	지정된 CPU 고기능 유니트와의 I/O 리프레시를 매회 실행합니다.	출력 계통	필요
7세그먼트 디코더	SDEC @SDEC	078	SDEC S K D S: 변환 데이터 CC 번호 K: 자리 지정 데이터 D: 변환 결과 출력 하위 CH 번호	채널 데이터에서 지정된 자리의 각 4비트 내용(0~F)을 8비트의 7세그먼트 데이터로 변환하여 지정된 CH 이후의 상위 또는 하위의 각 8비트에 출력합니다. 	출력 계통	필요
디지털 스위치	DSW	210	DSW I O D C1 C2 I: 데이터 라인 입력 (D0~3) CH 번호 O: 제어 신호(CS/RD) 출력 CH 번호 D: 데이터 저장 시작 CH 번호 C1: 자릿수 지정 시작 CH번호 C2: 작업 영역 시작 CH 번호	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.	출력 계통	필요
10키 입력	TKY	211	TKY I D1 D2 I: 데이터 라인 입력 CH 번호 D1: 데이터 저장 시작 CH 번호 D2: 키 입력 정보 저장 CH 번호	입력 유니트에 연결된 외부의 10키패드에서 숫자를 순서대로 읽어 지정된 CH 이후에 최대 8자리의 숫자(BCD 데이터)로 저장합니다.	출력 계통	필요
16키 입력	HKY	212	HKY I O D C I: 데이터 입력 CH 번호 O: 선택 제어 신호 출력 CH 번호 D: 데이터 저장 시작 CH 번호 C: 작업 영역 시작 CH 번호	입력 유니트 및 출력 유니트에 연결된 외부의 16키보드에서 숫자를 순서대로 읽어 지정된 CH 이후에 최대 8자리의 숫자(BCD 데이터)로 저장합니다.	출력 계통	필요

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
행렬 입력	MTR	213	MTR I O D C I: 데이터 입력 채널 번호 O: 데이터 선택 신호 출력 CH 번호 D: 입력 데이터 저장 시작 CH 번호 C: 작업 영역 시작 CH 번호	입력 유니트 및 출력 유니트에 연결된 외부의 8점×8열의 접점(행렬)에서 64점을 순서대로 읽어 지정된 CH 이후에 64점(4CH) 데이터로 저장합니다.	출력 계통	필요
7세그먼트 표시	7SEG	214	7SEG S O C D S: 표시 데이터 저장 시작 CH 번호 O: 데이터 출력/래치 출력 저장 CH 번호 C: 표시 자릿수 및 출력 논리 선택 데이터 D: 작업 영역 시작 CH 번호	4자리 또는 8자리의 숫자(BCD 데이터)를 7세그먼트 표시기용 데이터로 변환하여 지정된 CH 이후에 출력합니다.	출력 계통	필요
아날로그 입력 직접 변환 [CJ1W-AD042 전용]	AIDC/ @AIDC	216	AIDC W C W: 기기 번호 C: 아날로그 입력 번호	직접 변환 모드의 아날로그 입력 유니트 CJ1W-AD042에서 지정된 아날로그 입력 번호의 입력 변환값을 읽습니다.	출력 계통	필요
아날로그 출력 직접 변환 [CJ1W-DA042V 전용]	AODC/ @AODC	217	AODC W C W: 기기 번호 C: 아날로그 출력 번호	직접 변환 모드의 아날로그 출력 유니트 CJ1W-DA042V에 대해 지정된 아날로그 출력 번호의 출력 설정값을 출력합니다.	출력 계통	필요
NC 유니트 위치 결정 고속 기동 [CJ1W-NC□□4, CJ1W-NC□81 전용]	NCDMV @NCDMV	218	NCDMV C S C: 제어 데이터 S: 직접 운전 지령 릴레이 영역 시작 CH 번호	위치 제어 유니트(CJ1W-NC□□4, CJ1W-NC□81)의 지정된 축 번호에 대해 상대 이동 또는 절대 이동의 고속 PTP 기동을 합니다.	출력 계통	필요
NC 유니트 위치 결정 시작 트리거 [CJ1W-NC□81 전용]	NCDTR @NCDTR	219	NCDTR C C: 제어 데이터	위치 제어 유니트(CJ1W-NC□81)의 메모리 운전 기능에서 시퀀스의 시작 조건이 이 명령에서 지령 대기가 된 경우 해당 시퀀스 실행을 시작합니다.	출력 계통	필요
인텔리전트 I/O 읽기	IORD @IORD	222	IORD C W D C: 제어 데이터 W: 전송 원본 기기 번호 및 전송 CH 수 D: 전송 대상 하위 CH 번호	고기능 I/O 유니트 또는 CPU 고기능 유니트의 메모리 영역의 내용을 읽습니다. 	출력 계통	필요

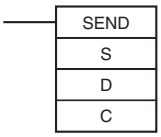
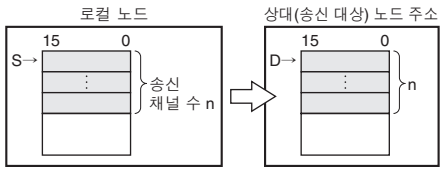
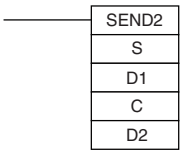
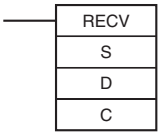
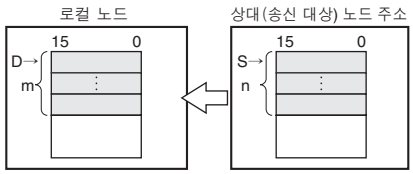
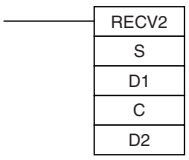
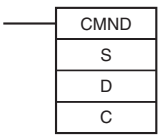
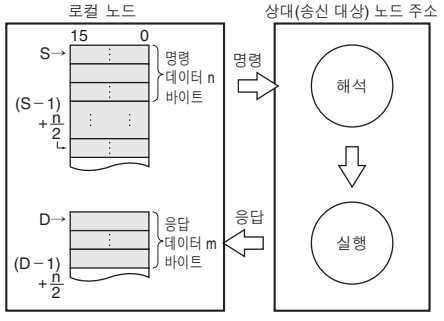
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
인텔리전트 I/O 쓰기	IOWR @IOWR	223	 C: 제어 데이터 S: 전송 원본 하위 CH 번호 W: 전송 대상 기기 번호 및 전송 CH 수	CPU 유니트의 I/O 메모리 영역의 내용을 고기능 I/O 유니트 또는 CPU 고기능 유니트에 출력합니다. 	출력 계통	필요

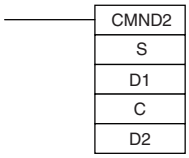
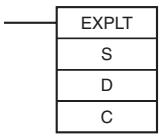
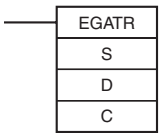
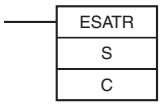
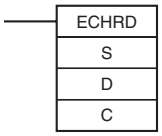
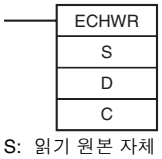
부록-1-22 시리얼 통신 명령

명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
프로토콜 매크로	PMCR @PMCR	260	PMCR C1 C2 S D 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	시리얼 커뮤니케이션 유니트에 등록된 송수신 시퀀스 (프로토콜 데이터)를 호출하여 실행합니다. 	출력 계통	필요
프로토콜 매크로 송신 2	PMCR2 @PMCR2	264	PMCR2 C1 C2 S D1 D2 C1: 제어 데이터 1 C2: 제어 데이터 2 S: 송신 데이터 시작 채널 D1: 수신 데이터 저장 대상 시작 채널 D2: 통신 정보	동시에 최대 64개를 사용할 수 있는 PMCR 명령입니다.	출력 계통	필요
시리얼 포트 출력	TXD @TXD	236	TXD S C N S: 송신 데이터 시작 CH 번호 C: 제어 데이터 N: 송신 바이트 수 10진수 &0~256(16진수 #0000~0100)	지정된 바이트 수 분량의 데이터를 그 상태 그대로 변환하지 않고 PLC 시스템 설정의 무수순 모드일 때의 시작 코드/종료 코드 지정에 따라 CPU 유니트의 시리얼 포트(무수순 모드)에서 송신합니다.	출력 계통	필요
시리얼 포트 입력	RXD @RXD	235	RXD D C N D: 수신 데이터 저장 시작 CH 번호 C: 제어 데이터 N: 저장 바이트 수 10진수 &0~256(16진수 #0000~0100)	CPU 유니트의 시리얼 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.	출력 계통	필요
시리얼커뮤니케이션 유니트 시리얼 포트 출력	TXDU @TXDU	256	TXDU S C N S: 송신 데이터 시작 CH 번호 C: 제어 데이터 시작CH 번호 N: 송신 바이트 수 10진수 &0~256(16진수 #0000~0100)	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.	출력 계통	필요

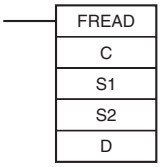
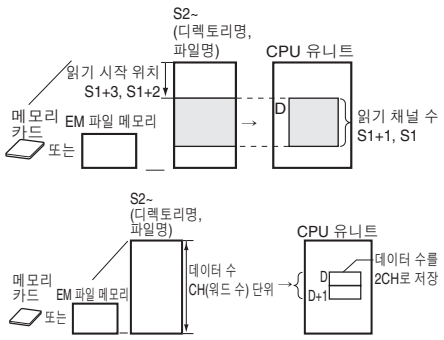
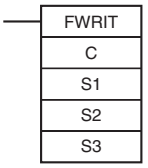
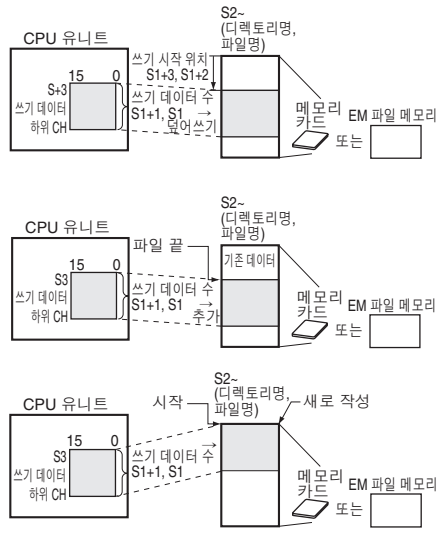
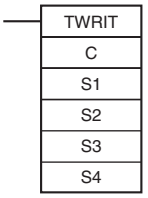
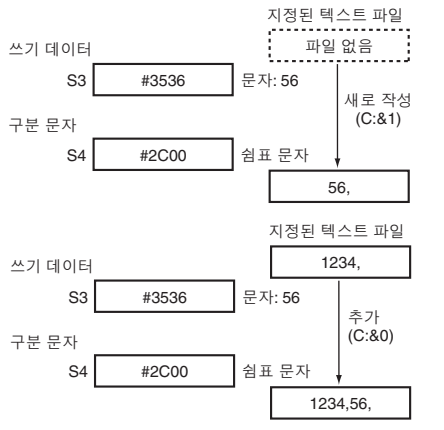
명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
시리얼 커뮤니케이션 유니트 시리얼 포트 입력	RXDU @RXDU	255	 D: 수신 데이터 저장 시작 CH 번호 C: 제어 데이터 시작 CH 번호 N: 저장 바이트 수 10진수 &0~256(16진수 #0000~0100)	시리얼 커뮤니케이션 유니트의 유니트 Ver.1.2 이상의 시리얼 통신 포트에서 할당 DM 영역 설정의 무수순 모드일 때의 시작 코드/종료 코드 지정에 따라 수신한 데이터를 지정된 시작 CH 번호부터 지정된 바이트 수만큼 출력합니다.	출력 계통	필요
시리얼 커뮤니케이션 유니트 시리얼 포트 직접 수신 [CJ1W-SCU 22/32/42 전용]	DRXDU @DRXDU	261	 D: 수신 데이터 시작 CH 번호 C: 제어 데이터 시작 CH 번호 N: 저장 바이트 수 10진수 &0~256(16진수 #0000~0100)	지정된 바이트 수 분량의 데이터를 그 상태 그대로 변환하지 않고 PLC 시스템 설정의 무수순 모드일 때의 시작 코드/종료 코드 지정에 따라 시리얼 커뮤니케이션 유니트 CJ1W-SCU22/32/42의 시리얼 통신 포트(무수순 중점 모드)에서 송신합니다. 이 명령은 명령 실행 시에 시리얼 커뮤니케이션 유니트에서 CPU 유니트로 수신 데이터를 전송하기 때문에 고속 데이터 전송이 가능합니다.	출력 계통	필요
시리얼 커뮤니케이션 유니트 시리얼 포트 직접 송신 [CJ1W-SCU 22/32/42 전용]	DTXDU @DTXDU	262	 S: 송신 데이터 시작 CH 번호 C: 제어 데이터 시작CH 번호 N: 송신 바이트 수 10진수 &0~256(16진수 #0000~0100)	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 CPU 유니트에서 시리얼 커뮤니케이션 유니트로 송신 데이터를 전송하기 때문에 고속 데이터 송신이 가능합니다.	출력 계통	필요
시리얼 포트 통신 설정 변경	STUP @STUP	237	 C: 포트 지정 S: 설정 데이터 시작 CH 번호	CPU 유니트, 시리얼 커뮤니케이션 유니트 또는 시리얼 커뮤니케이션 보드의 시리얼 포트 통신 설정을 변경합니다.	출력 계통	필요

부록-1-23 네트워크 통신용 명령

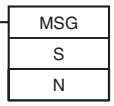
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
네트워크 송신	SEND @SEND	090	 S: 송신 원본(로컬 노드)의 송신 시작 CH 번호 D: 송신 대상(상대 노드)의 수신 시작 CH 번호 C: 제어 데이터 하위 CH 번호	네트워크상의 노드에 데이터를 송신합니다. 	출력 계통	필요
네트워크 송신 2	SEND2 @SEND2	491	 S: 송신 요구 대상의 송신 시작 채널 D1: 송신 요구 원본의 수신 시작 채널 C: 제어 데이터 D2: 통신 정보	동시에 최대 64개를 사용할 수 있는 SEND 명령입니다.	출력 계통	필요
네트워크 수신	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	네트워크상의 노드에 대해 송신을 요구하고 데이터를 수신합니다. 	출력 계통	필요
네트워크 수신 2	RECV2 @RECV2	492	 S: 송신 요구 대상의 송신 시작 채널 D1: 송신 요구 원본의 수신 시작 채널 C: 제어 데이터 D2: 통신 정보	동시에 최대 64개를 사용할 수 있는 RECV 명령입니다.	출력 계통	필요
명령 송신	CMND @CMND	490	 S: 명령 저장 시작 CH 번호 D: 응답 저장 시작 CH 번호 C: 제어 데이터 하위 CH 번호	임의의 FINS 명령을 전송하고 응답을 수신합니다. 	출력 계통	필요

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
명령 송신 2	CMND2 @CMND2	493	 S: 송신 명령 시작 채널 D1: 응답 저장 대상의 시작 채널 C: 제어 데이터 D2: 통신 정보	동시에 최대 64개를 사용할 수 있는 CMND 명령입니다.	출력 계통	필요
범용 Explicit 메시지 송신 명령	EXPLT	720	 S: 송신 메시지 저장 시작 CH 번호 D: 수신 메시지 저장 시작 CH 번호 C: 제어 데이터	임의의 ServiceCode의 Explicit 메시지를 송신합니다.	출력 계통	필요
Explicit 읽기 명령	EGATR	721	 S: 송신 메시지 저장 시작 CH 번호 D: 수신 메시지 저장 시작 CH 번호 C: 제어 데이터	Explicit 메시지를 통해 정보/상태 읽기(단일 속성 가져오기, 서비스 코드: 0EHex)를 실행합니다.	출력 계통	필요
Explicit 쓰기 명령	ESATR	722	 S: 송신 메시지 저장 시작 CH 번호 C: 제어 데이터	Explicit 메시지를 통해 정보 쓰기(단일 속성 설정, 서비스 코드: 10Hex)를 실행합니다.	출력 계통	필요
Explicit CPU 유니트 데이터 읽기 명령	ECHRD	723	 S: 상대 CPU 유니트 읽기 시작 CH 번호 D: 자체 CPU 유니트 수신 시작 CH 번호 C: 제어 데이터	Explicit 메시지를 지원하는 네트워크상의 CPU 유니트 데이터를 자체 CPU 유니트로 읽어옵니다.	출력 계통	필요
Explicit CPU 유니트 데이터 쓰기 명령	ECHWR	724	 S: 읽기 원본 자체 CPU 유니트의 시작 CH 번호 D: 쓰기 대상 상대 CPU 유니트의 시작 CH 번호 C: 제어 데이터	자체 CPU 유니트에서 Explicit 메시지를 지원하는 네트워크상의 CPU 유니트에 데이터를 씁니다.	출력 계통	필요

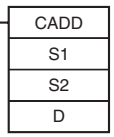
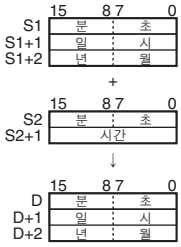
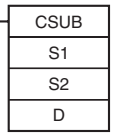
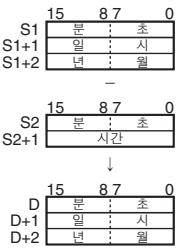
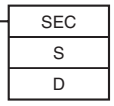
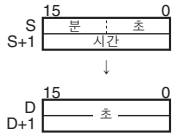
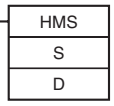
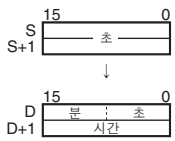
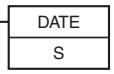
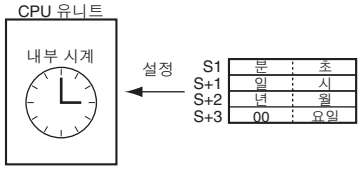
부록-1-24 파일 메모리용 명령

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
데이터 파일 읽기	FREAD @FREAD	700	 C: 제어 데이터 S1: 읽기 채널 수 및 읽기 시작 위치 S2: 디렉토리명 및 파일명 저장 CH 번호 D: 읽기 데이터 출력 하위 CH 번호	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. 	출력 계통	필요
데이터 파일 쓰기	FWRIT @FWRIT	701	 C: 제어 데이터 S1: 쓰기 채널 수 및 쓰기 시작 위치 S2: 디렉토리명 및 파일명 저장 CH 번호 S3: 읽기 원본 데이터 출력 하위 CH 번호	지정된 I/O 메모리 영역의 데이터를 지정된 데이터 파일(I/O 메모리 파일)에 씁니다. 	출력 계통	필요
텍스트 파일 쓰기	TWRIT @TWRIT	704	 C: 제어 데이터 S1: 쓰기 바이트 수 S2: 디렉토리명 및 파일명 저장 시작 CH 번호 S3: 쓰기 데이터 저장 시작 CH 번호 S4: 구분 문자	I/O 메모리 영역에 있는 ASCII 데이터를 메모리 카드에 텍스트 파일로 저장합니다(새로 쓰기 또는 추가로 쓰기). 저장하는 데이터 형식은 TXT 형식입니다. 	출력 계통	필요

부록-1-25 표시 기능용 명령

명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
메시지 표시	MSG @MSG	046	 S: 메시지 번호 N: 메시지 저장 하위 CH 번호	지정된 CH에서 16채널(워드)분의 메시지(ASCII 코드)를 읽어서 프로그래밍 콘솔에 표시합니다.	출력 계통	필요

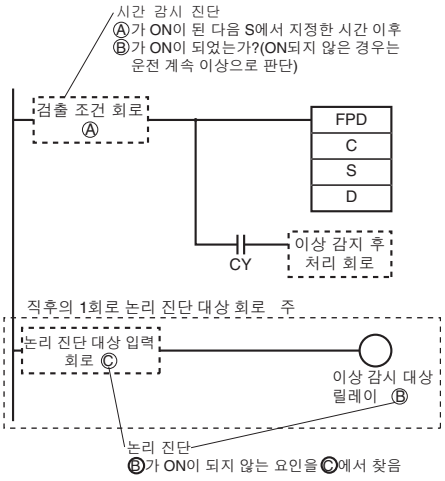
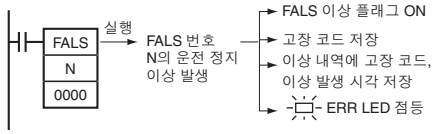
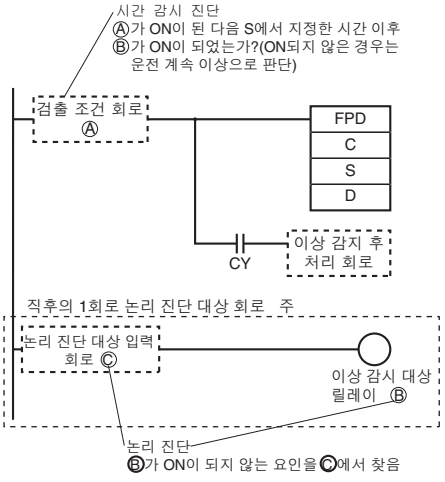
부록-1-26 시계 기능 명령

명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
캘린더 가산	CADD @CADD	730	 S1: 피가산 데이터(시각) 하위 CH 번호 S2: 가산 데이터(시간) 하위 CH 번호 D: 연산 결과(시각) 출력 하위 CH 번호	시각 데이터에 시간 데이터를 가산합니다. 	출력 계통	필요
캘린더 감산	CSUB @CSUB	731	 S1: 피감산 데이터(시각) 하위 CH 번호 S2: 감산 데이터(시간) 하위 CH 번호 D: 연산 결과(시각) 출력 하위 CH 번호	시각 데이터에서 시간 데이터를 감산합니다. 	출력 계통	필요
시분초→초 변환	SEC @SEC	065	 S: 변환 원본 데이터(시분초) 하위 CH 번호 D: 변환 결과(초) 출력 하위 CH 번호	시분초 데이터를 초 데이터로 변환합니다. 	출력 계통	필요
초→시분초 변환	HMS @HMS	066	 S: 변환 원본 데이터(초) 하위 CH 번호 D: 변환 결과(시분초) 출력 하위 CH 번호	초 데이터를 시분초 데이터로 변환합니다. 	출력 계통	필요
시계 보정	DATE @DATE	735	 S: 시계 데이터 하위 CH 번호	지정된 시계 데이터에 따라 내부 시계의 값을 변경합니다. 	출력 계통	필요

부록-1-27 디버그 처리 명령

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
트레이스 메모리 샘플링	TRSM	045	TRSM	이 명령을 실행하면 접점 또는 채널 데이터를 샘플링하여 트레이스 메모리에 저장합니다. 프로그램에 여러 개를 배치할 수 있습니다.	출력 계통	필요 없음

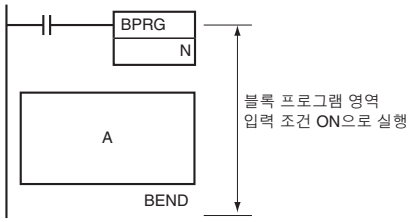
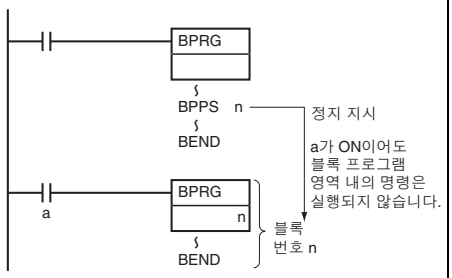
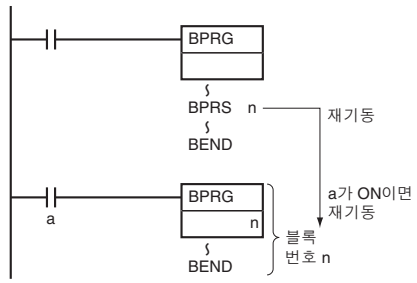
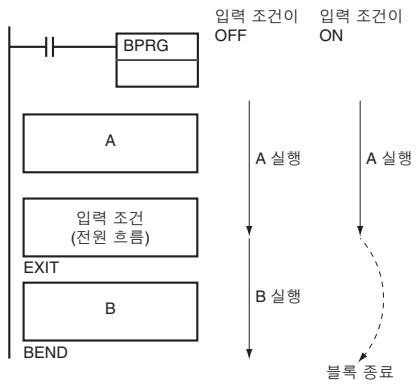
부록-1-28 고장 진단 명령

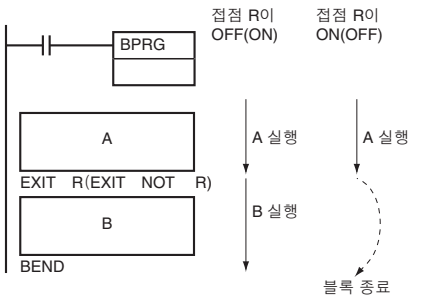
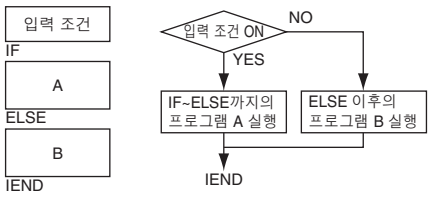
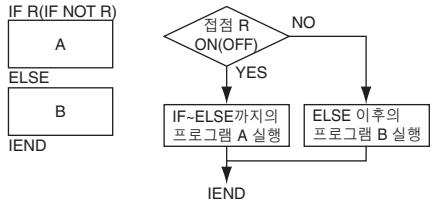
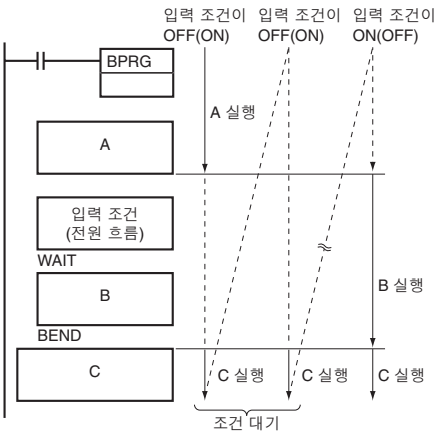
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
운전 계속 고장 진단	FAL @FAL	006	FAL N S N: FAL 번호 S: 메시지 저장 장소 하위 CH 번호 또는 메시지가 없는 경우는 상수(#0000~FFFF 중 하나) 또는 발생시키는 고장 코드	사용자 정의 운전 계속 이상을 등록 또는 해제합니다. 또는 시스템에 의한 운전 계속 이상을 고의로 발생시킵니다. 	출력 계통	필요
운전 정지 고장 진단	FALS	007	FALS N S N: FALS 번호 S: 메시지 저장 장소 하위 CH 번호 또는 메시지가 없는 경우는 상수(#0000~FFFF 중 하나) 또는 발생시키는 고장 코드	사용자 정의 운전 정지 이상을 등록합니다. 또는 시스템에 의한 운전 정지 이상을 고의로 발생시킵니다. 	출력 계통	필요
고장 지점 검출	FPD	269	FPD C S D C: 제어 데이터 S: 이상 감시 시간 설정값 D: 논리 진단 결과 출력 및 메시지 저장 하위 CH 번호	지정한 1회로에 대해 시간 감시 진단 및 논리 진단을 수행합니다. 	출력 계통	필요

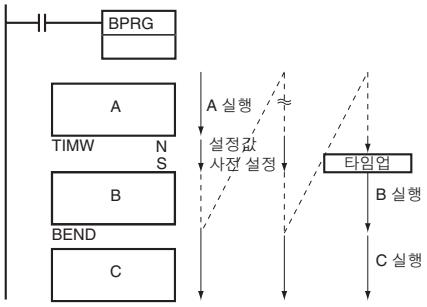
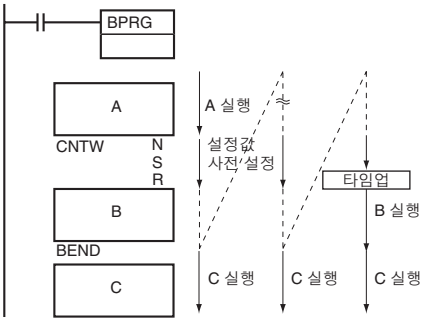
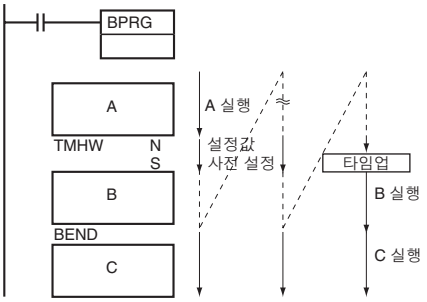
부록-1-29 특수 명령

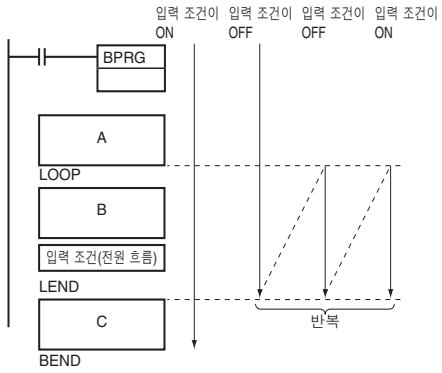
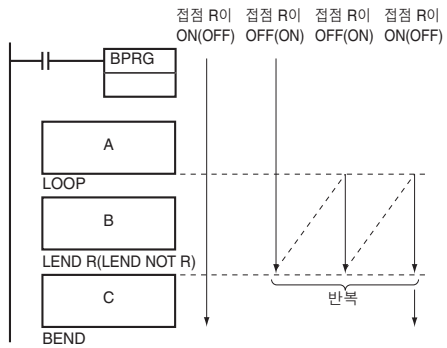
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
세트 캐리	STC @STC	040	STC	CY 플래그를 ON합니다.	출력 계통	필요
클리어 캐리	CLC @CLC	041	CLC	CY 플래그를 OFF합니다.	출력 계통	필요
확장 DM 뱅크 전환	EMBC @EMBC	281	EMBC N N: 뱅크 번호	EM(확장 DM) 영역의 현재 뱅크를 전환합니다.	출력 계통	필요
사이클 시간 감시 시간 설정	WDT @WDT	094	WDT S S: 사이클 시간 감시 시간 연장 데이터	이 명령을 실행한 사이클만 사이클 시간 감시 시간을 연장합니다.	출력 계통	필요
조건 플래그 저장	CCS @CCS	282	CCS	조건 플래그의 상태를 보관합니다.	출력 계통	필요
조건 플래그 로드	CCL @CCL	283	CCL	보관된 조건 플래그의 상태를 읽어옵니다.	출력 계통	필요
CV→CS 주소 변환	FRMCV @FRMCV	284	FRMCV S D S: CVM1/CV 시리즈의 메모리 주소 저장 대상 채널 번호 D: 변환 결과 출력 대상 IR	CVM1/CV 시리즈의 I/O 메모리 실제 주소를 CS/CJ 시리즈의 I/O 메모리 실제 주소로 변환합니다.	출력 계통	필요
CS→CV 주소 변환	TOCV @TOCV	285	TOCV S D S: CS 시리즈의 I/O 메모리 실제 주소 저장 대상 IR D: 변환 결과 출력 대상	CS/CJ 시리즈의 I/O 메모리 실제 주소를 CVM1/CV 시리즈의 I/O 메모리 실제 주소로 변환합니다.	출력 계통	필요

부록-1-30 블록 프로그램 명령

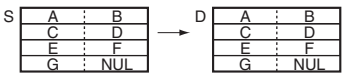
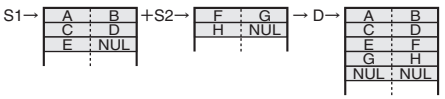
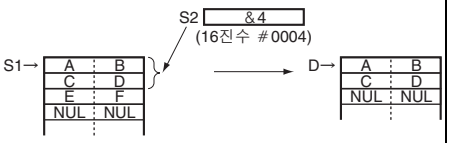
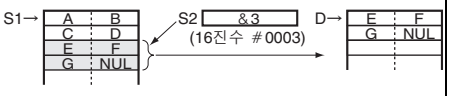
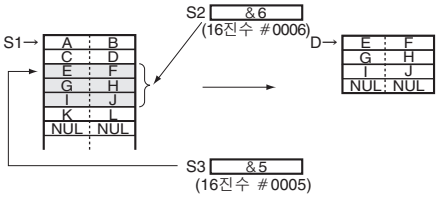
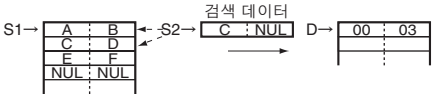
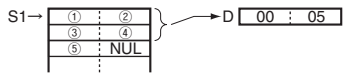
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
블록 프로그램	BPRG	096	BPRG N N: 블록 번호	지정된 번호의 블록 프로그램 영역의 선두를 나타냅니다. 	출력 계통	필요
블록 프로그램 종료	BEND	801		블록 프로그램 영역의 종료를 나타냅니다.	블록 프로그램	필요
블록 프로그램 일시 정지	BPPS	811	BPPS N N: 블록 번호	실행 중인 지정된 블록 프로그램을 일시 정지시킵니다. 	블록 프로그램	필요
블록 프로그램 재기동	BPRS	812	BPRS N N: 블록 번호	정지 중인 지정된 블록 프로그램 번호의 블록 프로그램을 재기동시켜서 실행합니다. 	블록 프로그램	필요
조건부 종료	입력 조건 EXIT	806	입력 조건 EXIT	입력 조건이 ON일 때 EXIT 명령부터 BEND 명령까지의 프로그램을 실행시키지 않고 블록 프로그램을 종료합니다. 	블록 프로그램	필요

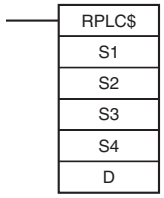
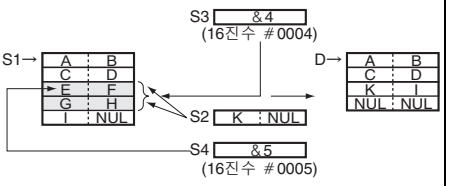
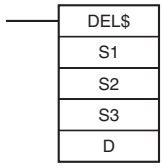
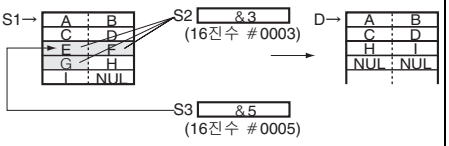
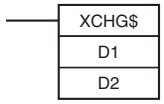
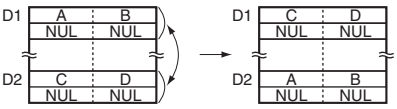
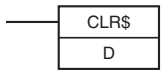
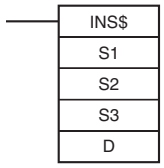
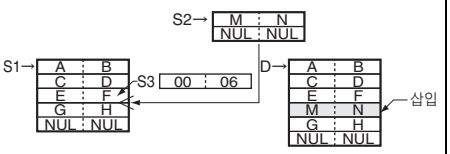
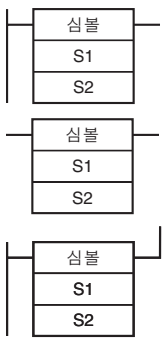
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
조건부 종료	EXIT 릴레이 번호	806	EXIT 릴레이 번호 R R: 릴레이 번호	지정된 접점이 ON일 때 EXIT 명령부터 BEND 명령까지의 프로그램을 실행시키지 않고 블록 프로그램을 종료합니다. 	블록 프로그램	필요
조건부 종료 부정	EXIT NOT 릴레이 번호	806	EXIT NOT 릴레이 번호 R R: 릴레이 번호	지정된 접점이 OFF일 때 EXIT 명령부터 BEND 명령까지의 프로그램을 실행시키지 않고 블록 프로그램을 종료합니다.	블록 프로그램	필요
조건 분기 블록	입력 조건 IF	802	입력 조건 IF	입력 조건이 OFF이면 다음 명령 이후를 실행하고 ON이면 ELSE 명령까지의 명령을 무시합니다. 	블록 프로그램	필요
조건 분기 블록	IF 릴레이 번호	802	IF 릴레이 번호 R R: 릴레이 번호	지정된 접점이 OFF이면 다음 명령 이후를 실행하고 ON이면 ELSE 명령까지의 명령을 무시합니다. 	블록 프로그램	필요
조건 분기 블록 (부정)	IF NOT 릴레이 번호	802	IF NOT 릴레이 번호 R R: 릴레이 번호	지정된 접점이 ON이면 다음 명령 이후를 실행하고 OFF이면 ELSE 명령까지의 명령을 무시합니다.	블록 프로그램	필요
조건 분기 거짓 블록	ELSE	803		IF 조건이 거짓일 때 실행되는 블록의 선두를 나타냅니다.	블록 프로그램	필요
조건 분기 블록 종료	IEND	804		조건 분기 블록 영역의 종료를 나타냅니다.	블록 프로그램	필요
1스캔 조건 대기	입력 조건 WAIT	805	입력 조건 WAIT	입력 조건이 ON이 될 때까지 WAIT 명령부터 BEND 명령까지의 프로그램을 실행하지 않습니다. 	블록 프로그램	필요
1스캔 조건 대기	WAIT 릴레이 번호	805	WAIT 릴레이 번호 R R: 릴레이 번호	지정된 접점이 ON이 될 때까지 WAIT 명령부터 BEND 명령까지의 프로그램을 실행하지 않습니다.	블록 프로그램	필요

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
1스캔 조건 대기(부정)	WAIT NOT 릴레이 번호	805	WAIT NOT 릴레이 번호 R R: 릴레이 번호	지정된 접점이 OFF가 될 때까지 WAIT 명령부터 BEND 명령까지의 프로그램을 실행하지 않습니다.	블록 프로그램	필요
타이머 대기	TIMW (BCD 방식)	813	TIMW N S N: 타이머 번호 S: 타이머 설정값	지정된 타이머가 타임업될 때까지 다음 명령부터 BEND 명령까지의 프로그램을 실행하지 않습니다. 설정값: BCD 방식일 때: 0~999.9초 BIN 방식일 때: 0~6553.5초 	블록 프로그램	필요
	TIMWX (BIN 방식)	816	TIMWX N S N: 타이머 번호 S: 타이머 설정값			
카운터 대기	CNTW (BCD 방식)	814	CNTW N S R N: 카운터 번호 S: 카운터 설정값 R: 카운트 입력	지정된 카운터가 카운트 업될 때까지 다음 명령부터 BEND 명령까지의 프로그램을 실행하지 않습니다. 설정값: BCD 방식일 때: 0~9999회 BIN 방식일 때: 0~65535회 	블록 프로그램	필요
	CNTWX (BIN 방식)	818	CNTWX N S R N: 카운터 번호 S: 카운터 설정값 R: 카운트 입력			
고속 타이머 대기	TMHW (BCD 방식)	815	TMHW N S N: 타이머 번호 S: 타이머 설정값	지정된 타이머가 타임업될 때까지 다음 명령부터 BEND 명령까지의 프로그램을 실행하지 않습니다. 설정값: BCD 방식일 때: 0~99.99초 BIN 방식일 때: 0~655.35초 	블록 프로그램	필요
	TMHWX (BIN 방식)	817	TMHWX N S N: 타이머 번호 S: 타이머 설정값			

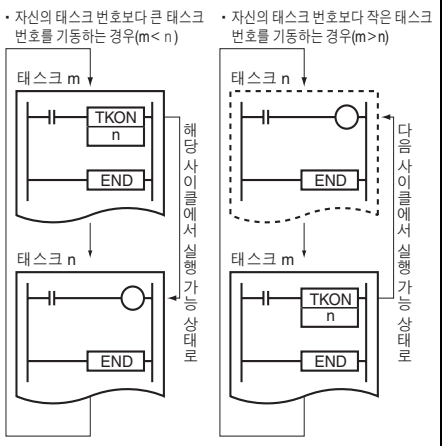
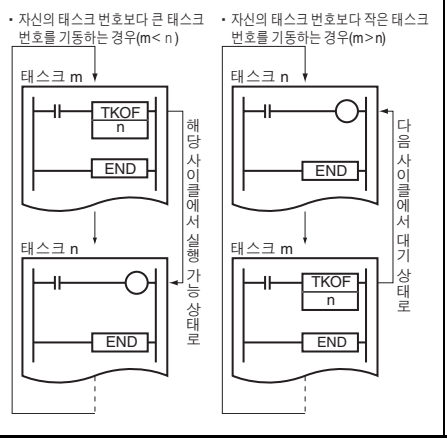
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
반복 블록	LOOP	809		반복 블록 영역(LOOP~LEND)의 선두를 나타냅니다. 	블록 프로그램	필요
반복 블록 종료	입력 조건 LEND	810	입력 조건 LEND	반복 블록 영역(LOOP~LEND)의 종료를 나타냅니다. 입력 조건이 OFF이면 LOOP 명령의 다음 명령으로 되돌아가 반복합니다. 입력 조건이 ON이면 반복 블록을 종료하고 다음 명령 이후를 실행합니다.	블록 프로그램	필요
반복 블록 종료	LEND 릴레이 번호	810	LEND 릴레이 번호 R R: 릴레이 번호	반복 블록 영역(LOOP~LEND)의 종료를 나타냅니다. 지정된 접점이 OFF이면 LOOP 명령의 다음 명령으로 되돌아가 반복합니다.  (주) 괄호 안은 LEND NOT 명령인 경우입니다. 지정된 접점이 ON이면 반복 블록을 종료하고 다음 명령 이후를 실행합니다.	블록 프로그램	필요
반복 블록 종료 (부정)	LEND NOT 릴레이 번호	810	LEND NOT 릴레이 번호 R R: 릴레이 번호	반복 블록 영역(LOOP~LEND NOT)의 종료를 나타냅니다. 지정된 접점이 ON이면 LOOP 명령의 다음 명령으로 되돌아가 반복합니다. 지정된 접점이 OFF이면 반복 블록을 종료하고 다음 명령 이후를 실행합니다.	블록 프로그램	필요

부록-1-31 문자열 처리 명령

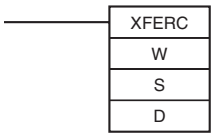
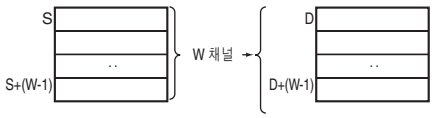
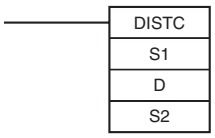
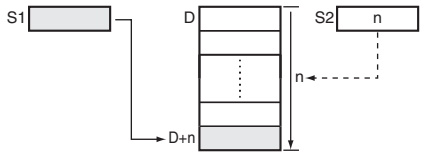
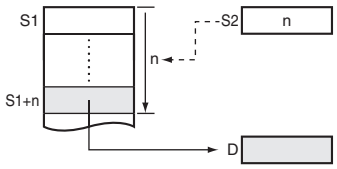
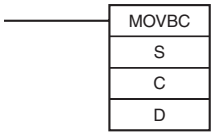
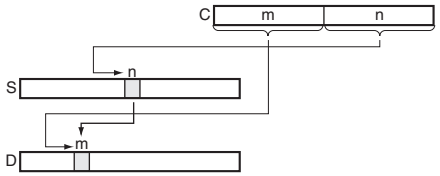
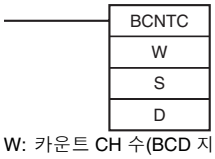
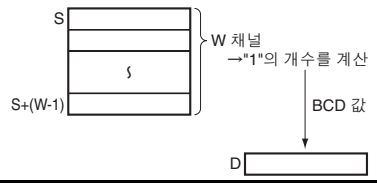
명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
문자열 전송	MOV$ @MOV$	664	MOV$ S D S: 전송 문자열 데이터 하위 CH 번호 D: 전송 대상 하위 CH 번호	문자열을 전송합니다. 	출력 계통	필요
문자열 연결	+$ @+$	656	+$ S1 S2 D S1: 문자열 데이터 하위 CH 번호 1 S2: 문자열 데이터 하위 CH 번호 2 D: 연결 문자열 출력 하위 CH 번호	문자열과 문자열을 연결합니다. 	출력 계통	필요
왼쪽에서 문자열 가져오기	LEFT$ @LEFT$	652	LEFT$ S1 S2 D S1: 문자열 데이터 하위 CH 번호 S2: 가져올 문자 수 데이터 D: 문자열 출력 하위 CH 번호	문자열의 왼쪽(선두)부터 지정된 문자 수만큼 가져옵니다. 	출력 계통	필요
오른쪽에서 문자열 가져오기	RGHT$ @RGHT$	653	RGHT$ S1 S2 D S1: 문자열 데이터 하위 CH 번호 S2: 가져올 문자 수 데이터 D: 문자열 출력 하위 CH 번호	문자열의 오른쪽(끝)부터 지정된 문자 수만큼 가져옵니다. 	출력 계통	필요
임의 위치에서 문자열 가져오기	MID$ @MID$	654	MID$ S1 S2 S3 D S1: 문자열 데이터 하위 CH 번호 S2: 가져올 문자 수 데이터 S3: 가져올 시작 위치 데이터 D: 문자열 출력 하위 CH 번호	문자열의 임의의 위치부터 지정된 수만큼 가져옵니다. 	출력 계통	필요
문자열 검색	FIND$ @FIND$	660	FIND$ S1 S2 D S1: 피검색 문자열 데이터 하위 CH 번호 S2: 검색 문자열 데이터 하위 CH 번호 D: 검색 결과 출력 CH 번호	문자열에서 지정된 문자열을 검색하고 그 위치를 출력합니다. 	출력 계통	필요
문자열 길이 검출	LEN$ @LEN$	650	LEN$ S D S: 문자열 데이터 하위 CH 번호 D: 연산 결과 출력 CH 번호	문자열의 길이를 계산합니다. 	출력 계통	필요

명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
문자열 치환	RPLC$ @RPLC$	661	 S1: 문자열 데이터 하위 CH 번호 S2: 치환 문자열 데이터 하위 CH 번호 S3: 문자 수 데이터 S4: 치환 시작 위치 데이터 D: 치환 결과 출력 하위 CH 번호	지정된 위치의 문자열을 지정된 문자로 치환합니다. 	출력 계통	필요
문자열 삭제	DEL$ @DEL$	658	 S1: 문자열 데이터 하위 CH 번호 S2: 문자 수 데이터 S3: 삭제 시작 위치 데이터 D: 삭제 결과 출력 하위 CH 번호	문자열에서 지정된 문자열을 삭제합니다. 	출력 계통	필요
문자열 교환	XCHG$ @XCHG$	665	 D1: 교환 CH 번호 1 D2: 교환 CH 번호 2	문자열과 문자열을 교환합니다. 	출력 계통	필요
문자열 클리어	CLR$ @CLR$	666	 D: 문자열 데이터 하위 CH 번호	문자열을 모두 NUL(00Hex)로 클리어합니다. 	출력 계통	필요
문자열 삽입	INS$ @INS$	657	 S1: 피삽입 문자열 데이터 하위 CH 번호 S2: 삽입 문자열 데이터 하위 CH 번호 S3: 삽입 시작 위치 데이터 CH 번호 D: 삽입 결과 출력 하위 CH 번호	지정된 문자열 위치에 지정된 문자열을 삽입합니다. 	출력 계통	필요
문자열 비교	LD,AND, OR + =$, <>$, <$, <=$, >$, >=$	670 (= $) 671 (< > $) 672 (< $) 673 (< = $) 674 (> $) 675 (> = $)	 S1: 문자열 데이터 하위 CH 번호 1 S2: 문자열 데이터 하위 CH 번호 2	2개의 문자열 데이터를 ASCII 코드의 크기로 선두 문자부터 비교(=, < >, <, < =, >, > = 중 하나)하여 비교 결과가 참일 때 다음 단계 이후로 연결합니다. 연결 형식에는 LD(로드) 연결, AND(직렬) 연결, OR(병렬) 연결의 3종류가 있습니다.	LD형: 논리 시작 AND, OR형: 다음 단계 연결형	LD형 :필요 없음 AND, OR형 : 필요

부록-1-32 태스크 제어 명령

명령 이름	니모닉	평션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
태스크 실행 기동	TKON @TKON	820	TKON N N: 태스크 번호	태스크를 실행 가능 상태로 설정합니다. 	출력 계통	필요
태스크 실행 대기	TKOF @TKOF	821	TKOF N N: 태스크 번호	태스크를 대기 상태로 설정합니다. 	출력 계통	필요

부록-1-33 기종 변환용 명령

명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
블록 전송	XFERC @ XFERC	565	 W: 전송 CH 수(BCD 지정) S: 전송 원본 하위 CH 번호 D: 전송 대상 하위 CH 번호	연속된 여러 CH 데이터를 일괄 전송합니다. 	출력 계통	필요
데이터 분배	DISTC @DISTC	566	 S1: 전송 데이터 D: 전송 대상 기준 CH 번호 S2: 오프셋 데이터(BCD 지정)	전송 대상을 기준으로 오프셋 지정된 채널로 전송 데이터를 전송합니다.  데이터 분배 동작 이외에 스택 PUSH 동작이 가능	출력 계통	필요
데이터 추출	COLLC @COLLC	567	 S1: 전송 원본 기준 CH 번호 S2: 오프셋 데이터(BCD 지정) D: 전송 대상 CH 번호	전송 원본을 기준으로 오프셋 지정된 채널의 내용을 지정 채널로 전송합니다.  데이터 추출 동작 이외에 스택 읽기 동작이 가능	출력 계통	필요
비트 전송	MOVBC @MOVBC	568	 S: 전송 원본 CH 번호 C: 제어 데이터(BCD 지정) D: 전송 대상 CH 번호	지정한 비트를 전송합니다. 	출력 계통	필요
비트 카운터	BCNTC @BCNTC	621	 W: 카운트 CH 수(BCD 지정) S: 카운트 하위 CH 번호 D: 카운트 결과 출력 CH 번호	CH 데이터 내에서 "1"인 비트의 총 수를 계산합니다. 	출력 계통	필요

부록-1-34 펑션 블록용 특수 명령

명령 이름	니모닉	펑션 번호	심볼/피연산자	기능	명령 배치 위치에 따른 분류	입력 조건
변수 종류 취득	GETID @ GETID	286	GETID S D1 D2 S: 대상 변수 또는 주소 D1: 변수 종류(I/O 메모리 영역 종류) 코드 D2: CH 번호	지정된 변수(또는 주소)의 FINS 명령의 변수 종류(I/O 메모리 영역 종류) 코드 및 CH 번호를 출력합니다. 주로 펑션 블록 내 변수의 할당 주소를 취득할 때 사용합니다.	출력 계통	필요
스텝 활성화 명령	SA @SA	784	SA D	지정된 스텝 및 서브 차트를 활성화 상태로 하고 작업의 실행을 시작합니다.	출력 계통	필요
스텝 비활성화 명령	SE @SE	785	SE D	지정된 스텝 및 서브 차트를 비활성화 상태로 하고 작업의 실행을 정지시킵니다.	출력 계통	필요
스텝 타이머 현재값 읽기 명령	TSR @TSR	780	TSR S D	S로 지정한 스텝의 스텝 타이머 현재값을 D에 저장합니다.	출력 계통	필요
스텝 타이머 현재값 변경 명령	TSW @TSW	781	TSW S D	D로 지정한 스텝의 스텝 타이머 값을 S로 지정한 값으로 변경합니다.	출력 계통	필요
SFC 태스크 실행 명령	SFCON	789	SFCON N N: SFC 태스크 번호	SFC 태스크 제어 명령에 의해 종료 또는 일시 정지된 SFC 태스크의 실행을 재개합니다.	출력 계통	필요
SFC 태스크 종료 명령	SFCOFF	790	SFCOFF N N: SFC 태스크 번호	SFC 프로그램의 실행 상태를 종료시킵니다. 이때, 출력은 모두 유지된 상태에서 종료합니다. 종료한 SFC 태스크의 실행을 시작하면 초기화 스텝부터 처리가 시작됩니다.	출력 계통	필요
SFC 태스크 정지 명령 (리셋 있음)	SFCPR	793	SFCPR N N: SFC 태스크 번호	N으로 지정한 SFC 태스크를 일시 정지합니다. 이때, 정지한 SFC 태스크의 출력은 모두 리셋됩니다.	출력 계통	필요
SFC 태스크 정지 명령 (리셋 없음)	SFCPRN	791	SFCPRN N N: SFC 태스크 번호	N으로 지정한 SFC 태스크를 일시 정지합니다. 이때, 정지한 SFC 태스크의 출력은 모두 유지됩니다.	출력 계통	필요

부록-2 명령 처리 시간/스텝 수 목록

부록-2-1 시퀀스 입력 명령 부록-76
부록-2-2 시퀀스 출력 명령 부록-77
부록-2-3 시퀀스 제어 명령 부록-77
부록-2-4 타이머/카운터 명령 부록-78
부록-2-5 데이터 비교 명령 부록-79
부록-2-6 데이터 전송 명령 부록-80
부록-2-7 데이터 이동 명령 부록-81
부록-2-8 증가/감소 명령 부록-81
부록-2-9 사칙연산 명령 부록-82
부록-2-10 데이터 변환 명령 부록-83
부록-2-11 논리 연산 명령 부록-84
부록-2-12 특수 연산 명령 부록-84
부록-2-13 부동 소수점 변환 및 연산 명령 부록-85
부록-2-14 배정도 부동 소수점 변환 및 연산 명령 부록-86
부록-2-15 테이블 데이터 처리 명령 부록-87
부록-2-16 추적 명령 부록-88
부록-2-17 데이터 제어 명령 부록-88
부록-2-18 서브루틴 명령 부록-88
부록-2-19 인터럽트 제어 명령 부록-89
부록-2-20 공정 진행 제어 명령 부록-89
부록-2-21 I/O 유니트용 명령 부록-89
부록-2-22 시리얼 통신 명령 부록-90
부록-2-23 네트워크 통신 명령 부록-90
부록-2-24 파일 메모리용 명령 부록-90
부록-2-25 표시 기능용 명령 부록-90
부록-2-26 시계 기능 명령 부록-91
부록-2-27 디버그 처리 명령 부록-91
부록-2-28 고장 진단 명령 부록-91
부록-2-29 특수 명령 부록-91
부록-2-30 블록 프로그램 명령 부록-92
부록-2-31 문자열 처리 명령 부록-93
부록-2-32 태스크 제어 명령 부록-93
부록-2-33 기종 변환용 명령 부록-93
부록-2-34 펑션 블록용 특수 명령 부록-94
부록-2-35 SFC 제어 명령 부록-94
부록-2-36 펑션 블록 사용 시의 인스턴스 실행 시간 부록-94

각 명령어의 처리 시간을 나타냅니다.
하나의 사용자 프로그램 전체(*)에서 각 명령어의 처리 시간을 합한 시간이 사이클 시간 중 '연산 처리 시간'입니다.

* 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: μ초)

조건	CJ2 CPU 유니트	
	CJ2H-CPU6□(-EIP)	CJ2M-CPU□□
미분(↑ ↓)을 지원하는 경우	+0.24	+0.32
미분(@%)을 지원하는 경우	+0.24	+0.32

2 명령 비실행 시의 1스텝당 처리 시간의 기준은 다음과 같습니다.

(단위: μ초)

CJ2 CPU 유니트	
CJ2H-CPU6□(-EIP)	CJ2M-CPU□□
0.016	0.020

명령 실행 순서, 실행 유무에 따라 처리 시간이 약간 다릅니다. CJ2 CPU 유니트의 경우, 최악의 값은 다음과 같이 계산됩니다.
CJ2H CPU 유니트: 명령 스텝 수 × 0.016(μs)
CJ2M CPU 유니트: 명령 스텝 수 × 0.020(μs)

부록-2-1 시퀀스 입력 명령

명령 이름	니모닉	펑션 번호	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	520	1	0.016	0.040	—
P.F. 기동 미분	UP	521	3	0.26	0.36	—
P.F. 종료 미분	DOWN	522	4	0.27	0.40	—
LD 비트 테스트	LD TST	350	4	0.11	0.16	—
LD 비트 테스트 부정	LD TSTN	351	4	0.11	0.16	—
AND 비트 테스트	AND TST	351	4	0.11	0.16	—
AND 비트 테스트 부정	AND TSTN	351	4	0.11	0.16	—
OR 비트 테스트	OR TST	350	4	0.11	0.16	—
OR 비트 테스트 부정	OR TSTN	351	4	0.11	0.16	—

부록-2-2 시퀀스 출력 명령

명령 이름	니모닉	펑션 번호	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	011	1~2	0.048	0.060	—
	!KEEP	011	16	0.99	1.340	—
기동 미분	DIFU	013	2~2	0.28	0.30	—
종료 미분	DIFD	014	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	530	4	3.68	4.12	1비트 세트
				15.5	24.4	1000비트 세트
다중 비트 리셋	RSTA	531	4	3.7	4.1	1비트 리셋
				15.5	24.4	1000비트 리셋
1비트 세트	SETB	532	2	0.19	0.280	—
	!SETB	—	16	0.99	1.120	—
1비트 리셋	RSTB	534	2	0.19	0.280	—
	!RSTB	—	16	0.99	1.120	—
1비트 출력	OUTB	534	2	0.19	0.280	—
	!OUTB	—	16	0.99	1.180	—

부록-2-3 시퀀스 제어 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CP U6□(-EIP)	CJ2M-CPU □□	
종료	END	001	1	2.6	3.5	—
기능 없음	NOP	000	1	0.016	0.040	—
인터록	IL	002	1	0.048	0.060	—
인터록 클리어	ILC	003	1	0.048	0.060	—
멀티 인터록 (미분 플래그 유지형)	MILH	517	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	518	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	519	2	1.2	1.7	인터록 중이 아닐 때
				1.6	2.2	인터록 중일 때
점프	JMP	004	2	0.31	0.34	—
점프 종료	JME	005	2	—	—	—
조건부 점프	CJP	510	2	0.31	0.34	점프 조건이 성립할 때 (입력 조건 ON)
조건부 부정 점프	CJPN	511	2	0.31	0.34	점프 조건이 성립할 때 (입력 조건 OFF)
복수 점프	JMP0	515	1	0.048	0.060	—
복수 점프 종료	JME0	516	1	0.048	0.060	—
반복 시작	FOR	512	2	0.27	0.42	상수 지정
루프 중단	BREAK	514	1	0.048	0.060	—
반복 종료	NEXT	513	1	0.14	0.16	루프를 계속할 때
				0.18	0.18	루프를 종료할 때

부록-2-4 타이머/카운터 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
100ms 타이머	TIM	—	3	0.67	0.84	—
	TIMX	550		0.67	0.84	—
10ms 타이머	TIMH	015	3	0.67	0.84	—
	TIMHX	551		0.67	0.84	—
1ms 타이머	TMHH	540	3	0.67	0.84	—
	TMHHX	552		0.67	0.84	—
0.1ms 타이머	TIMU	541	3	0.67	0.84	—
	TIMUX	556		0.67	0.84	—
00.1ms 타이머	TMUH	544	3	0.67	0.84	—
	TMUHX	557		0.67	0.84	—
적산 타이머	TTIM	087	3	9.2	12.1	—
				6.9	8.4	리셋할 때
				5.0	6.5	인터록할 때
	TTIMX	555	3	8.8	11.7	—
				6.8	8.5	리셋할 때
				5.0	6.5	인터록할 때
장시간 타이머	TIML	542	4~5	5.8	7.0	—
				3.9	4.1	인터록할 때
	TIMLX	553	4~5	5.7	7.0	—
				3.6	3.7	인터록할 때
멀티 출력 타이머	MTIM	543	4	6.4	7.2	—
				3.7	4.3	리셋할 때
	MTIMX	554	4	5.5	6.4	—
				3.4	3.8	리셋할 때
타이머 리셋	TRSET	549	2	0.58	0.8	—
카운터	CNT	—	3	0.51	0.58	—
	CNTX	546		0.51	0.58	—
가역 카운터	CNTR	012	3	9.1	11.8	—
	CNTRX	548		8.0	10.3	—
타이머/카운터 리셋	CNR	545	3	4.8	5.4	1CH 리셋
				2839	2555	1000CH 리셋
	CNRX	547	3	4.7	5.5	1CH 리셋
				2839	2555	1000CH 리셋

부록-2-5 데이터 비교 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
심볼 비교(부호 없음)	=	300	4	0.08	0.16	—
	< >	305				
	<	310				
	< =	315				
	>	320				
	> =	325				
심볼 비교(2워드, 부호 없음)	= L	301	4~7	0.08	0.24	—
	< > L	306				
	< L	311				
	< = L	316				
	> L	321				
	> = L	326				
심볼 비교(부호 있음)	= S	302	4	0.08	0.16	—
	< > S	307				
	< S	312				
	< =	317				
	> S	322				
	> = S	327				
심볼 비교(2워드, 부호 있음)	= SL	303	4~7	0.08	0.24	—
	< > SL	308				
	< SL	313				
	< = SL	318				
	> SL	323				
	> = SL	328				
시각 비교	= DT	341	4	16.300	27.9	—
	< > DT	342				—
	< DT	343				—
	< = DT	344				—
	> DT	345				—
	> = DT	346				—
비교	CMP	020	3	0.06	0.080	—
	!CMP	020	30	2.06	2.6	—
2워드 비교	CMPL	060	3~5	0.064	0.120	—
부호 있는 BIN 비교	CPS	114	3~5	0.064	0.080	—
	!CPS	114	30	2.06	2.6	—
부호 있는 BIN 2워드 비교	CPSL	115	3~5	0.064	0.120	—
테이블 일치	TCMP	085	4	10.3	12.5	—
다채널 비교	MCMP	019	4	15.2	20.3	—
부호 없는 테이블 간 비교	BCMP	068	4	16.3	20.5	—
확장 테이블 간 비교	BCMP2	502	4	5.0	5.1	데이터 수 1
				217.2	278	데이터 수 255
영역 비교	ZCP	088	3	0.14	0.400	—
2워드 영역 비교	ZCPL	116	3~5	0.14	0.640	—
부호 있는 영역 비교	ZCPS	117	3	0.14	0.400	
부호 있는 2워드 영역 비교	ZCPSL	118	3~5	0.14	0.640	

부록-2-6 데이터 전송 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
전송	MOV	021	3	0.05	0.12	—
	!MOV	021	30	1.98	2.6	—
2워드 전송	MOVL	498	3~4	0.05	0.20	—
부정 전송	MVN	022	3	0.05	0.12	—
부정 2워드 전송	MVNL	499	3~4	0.05	0.20	—
비트 전송	MOVB	082	4	0.19	0.32	—
자릿수 전송	MOVD	083	4	0.19	0.32	—
다중 비트 전송	XFRB	062	4	6.6	9.4	1비트 전송
				85.8	119	255비트 전송
블록 전송	XFER	070	4	0.29	0.28	1CH 전송
				240.1	220	1000CH 전송
블록 설정	BSET	071	4	0.21	0.20	1CH 세트
				142.2	140	1000CH 세트
데이터 교환	XCHG	073	3	0.32	0.48	—
데이터 2워드 교환	XCGL	562	3~4	0.12	0.29	—
데이터 분배	DIST	080	4	4.5	4.7	—
데이터 추출	COLL	081	4	4.6	4.7	—
인덱스 레지스터 설정	MOVR	560	3	0.064	0.200	—
인덱스 레지스터 설정	MOVRW	561	3	0.064	0.200	—

부록-2-7 데이터 이동 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6 □(-EIP)	CJ2M-CPU □□	
이동 레지스터	SFT	010	3	2.86	3.47	1CH 이동
				315	422	1000CH 이동
좌우 이동 레지스터	SFTR	084	4	6.22	6.38	1CH 이동
				319	422	1000CH 이동
비동기 이동 레지스터	ASFT	017	4	5.3	6.3	1CH 이동
				948	1285	1000CH 이동(*)
워드 이동	WSFT	016	4	2.3	3.1	1CH 이동
				233	187	1000CH 이동
1비트 좌측 이동	ASL	025	2	0.18	0.260	—
1비트 좌측 이동(2워드)	ASLL	570	2	0.32	0.420	—
1비트 우측 이동	ASR	026	2	0.18	0.260	—
1비트 우측 이동(2워드)	ASRL	571	2	0.32	0.420	—
CY가 있는 1비트 좌측 회전	ROL	027	2	0.18	0.260	—
CY가 있는 1비트 좌측 회전(2워드)	ROLL	572	2	0.32	0.420	—
CY가 없는 1비트 좌측 회전	RLNC	574	2	0.18	0.260	—
CY가 없는 1비트 좌측 회전(2워드)	RLNL	576	2	0.32	0.420	—
CY가 있는 1비트 우측 회전	ROR	028	2	0.18	0.260	—
CY가 있는 1비트 우측 회전(2워드)	RORL	573	2	0.32	0.420	—
CY가 없는 1비트 우측 회전	RRNC	575	2	0.18	0.260	—
CY가 없는 1비트 우측 회전(2워드)	RRNL	577	2	0.32	0.420	—
1자리 좌측 이동	SLD	074	3	3.7	4.4	1CH 이동
				317.8	429	1000CH 이동
1자리 우측 이동	SRD	075	3	4.5	5.4	1CH 이동
				479.5	656	1000CH 이동
N비트 데이터 좌측 이동	NSFL	578	4	4.6	5.2	1비트 이동
				31.5	36.1	1000비트 이동
N비트 데이터 우측 이동	NSFR	579	4	4.5	5.2	1비트 이동
				39.0	50.2	1000비트 이동
N비트 좌측 이동	NASL	580	3	0.18	0.38	—
N비트 2워드 좌측 이동	NSLL	582	3	0.32	0.54	—
N비트 우측 이동	NASR	581	3	0.18	0.38	—
N비트 2워드 우측 이동	NSRL	583	3	0.32	0.54	—

* 데이터량에 따라 명령 실행 시간이 크게 변화하며 사이클 시간에 영향을 줍니다. 사이클 시간에 대한 영향을 줄이기 위해 '백그라운드 실행'을 수행할 수 있습니다. 자세한 내용은 '10-2-5 백그라운드 처리 실행 기능'을 참조하십시오.

부록-2-8 증가/감소 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6 □(-EIP)	CJ2M-CPU □□	
BIN 증가	++	590	2	0.18	0.24	—
BIN 2워드 증가	++L	591	2	0.18	0.24	—
BIN 감소	--	592	2	0.18	0.24	—
BIN 2워드 감소	--L	593	2	0.18	0.24	—
BCD 증가	++B	594	2	3.0	3.4	—
BCD 2워드 증가	++BL	595	2	3.2	3.5	—
BCD 감소	--B	596	2	3.0	3.5	—
BCD 2워드 감소	--BL	597	2	3.2	3.5	—

부록-2-9 사칙연산 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
부호 있음, CY 없는 BIN 가산	+	400	4	0.18	0.34	—
부호 있음, CY 없는 BIN 2워드 가산	+L	401	4~6	0.18	0.24	—
부호 및 CY 있는 BIN 가산	+C	402	4	0.18	0.34	—
부호 및 CY 있는 BIN 2워드 가산	+CL	403	4~6	0.18	0.24	—
CY 없는 BCD 가산	+B	404	4	4.0	4.8	—
CY 없는 BCD 2워드 가산	+BL	405	4~6	4.9	6.0	—
CY 있는 BCD 가산	+BC	406	4	4.4	5.2	—
CY 있는 BCD 2워드 가산	+BCL	407	4~6	5.2	6.6	—
부호 있음, CY 없는 BIN 감산	-	410	4	0.18	0.340	—
부호 있음, CY 없는 BIN 2워드 감산	-L	411	4~6	0.18	0.24	—
부호 및 CY 있는 BIN 감산	-C	412	4	0.18	0.340	—
부호 및 CY 있는 BIN 2워드 감산	-CL	413	4~6	0.18	0.24	—
CY 없는 BCD 감산	-B	414	4	4.1	4.9	—
CY 없는 BCD 2워드 감산	-BL	415	4~6	4.9	5.9	—
CY 있는 BCD 감산	-BC	416	4	4.5	5.2	—
CY 있는 BCD 2워드 감산	-BCL	417	4~6	5.2	6.3	—
부호 있는 BIN 승산	*	420	4	0.26	0.520	—
부호 있는 BIN 2워드 승산	* L	421	4~6	3.6	3.9	—
부호 없는 BIN 승산	* U	422	4	0.26	0.26	—
부호 없는 BIN 2워드 승산	* UL	423	4~6	3.6	3.9	—
BCD 승산	* B	424	4	3.6	4.6	—
BCD 2워드 승산	* BL	425	4~6	4.9	6.2	—
부호 있는 BIN 제산	/	430	4	0.29	0.540	—
부호 있는 BIN 2워드 제산	/L	431	4~6	4.2	4.8	—
부호 없는 BIN 제산	/U	432	4	0.29	0.540	—
부호 없는 BIN 2워드 제산	/UL	433	4~6	3.8	4.2	—
BCD 제산	/B	434	4	5.0	5.9	—
BCD 2워드 제산	/BL	435	4~6	4.8	5.9	—

부록-2-10 데이터 변환 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6 □(-EIP)	CJ2M-CPU □□	
BCD→BIN 변환	BIN	023	3	0.18	0.280	—
BCD→BIN 2워드 변환	BINL	058	3~4	3.3	3.5	—
BIN→BCD 변환	BCD	024	3	0.19	0.300	—
BIN→BCD 2워드 변환	BCDL	059	3~4	3.3	3.7	—
2의 보수 변환	NEG	160	3	0.14	0.240	—
2의 보수 변환(2워드)	NEGL	161	3~4	0.26	0.440	—
부호 확장	SIGN	600	3	0.26	0.340	—
4→16/8→256 디코더	MLPX	076	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	077	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	086	4	4.0	4.5	1자리 ASCII 코드 변환
				4.6	5.2	4자리 ASCII 코드 변환
ASCII→HEX 변환	HEX	162	4	3.3	3.8	1자리 변환
비트 열→비트 행 변환	LINE	063	4	10.5	13.1	—
비트 행→비트 열 변환	COLM	064	4	13.8	17.6	—
부호 있는 BCD→BIN 변환	BINS	470	4	3.6	4.0	데이터 형식 지정 번호 0
				3.6	4.0	데이터 형식 지정 번호 1
				3.6	4.0	데이터 형식 지정 번호 2
				3.6	4.0	데이터 형식 지정 번호 3
부호 있는 BCD→BIN 2워드 변환	BISL	472	4~5	3.7	4.1	데이터 형식 지정 번호 0
				3.6	4.1	데이터 형식 지정 번호 1
				3.7	4.2	데이터 형식 지정 번호 2
				3.7	4.2	데이터 형식 지정 번호 3
부호 있는 BIN→BCD 변환	BCDS	471	4	3.7	4.0	데이터 형식 지정 번호 0
				3.7	4.1	데이터 형식 지정 번호 1
				3.7	4.2	데이터 형식 지정 번호 2
				3.7	4.2	데이터 형식 지정 번호 3
부호 있는 BIN→BCD 2워드 변환	BDSL	473	4~5	4.0	4.5	데이터 형식 지정 번호 0
				4.0	4.6	데이터 형식 지정 번호 1
				4.0	4.6	데이터 형식 지정 번호 2
				4.1	4.6	데이터 형식 지정 번호 3
그레이 코드 변환	GRY	474	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	478	3	0.1	0.3	—
그레이 코드→BIN 2워드 변환	GRAY_BINL	479	3~4	0.1	0.4	—
BIN→그레이 코드 변환	BIN_GRAY	480	3	0.1	0.3	—
BIN→그레이 코드 2워드 변환	BIN_GRAYL	481	3~4	0.1	0.4	—
숫자 4자리→ASCII 변환	STR4	601	3	8.4	14.2	—
숫자 8자리→ASCII 변환	STR8	602	3~4	10.2	16.4	—
숫자 16자리→ASCII 변환	STR16	603	3	15.8	28.2	—
ASCII→숫자 4자리 변환	NUM4	604	3~4	10.5	18.5	—
ASCII→숫자 8자리 변환	NUM8	605	3	14.8	27.1	—
ASCII→숫자 16자리 변환	NUM16	606	3	27.4	52.0	—

부록-2-11 논리 연산 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
워드 논리곱	ANDW	034	4	0.14	0.340	—
워드 논리곱(2워드)	ANDL	610	4~6	0.26	0.640	—
워드 논리합	ORW	035	4	0.18	0.340	—
워드 논리합(2워드)	ORWL	611	4~6	0.26	0.640	—
워드 배타적 논리합	XORW	036	4	0.18	0.340	—
워드 배타적 논리합(2워드)	XORL	612	4~6	0.26	0.640	—
워드 배타적 논리합 부정	XNRW	037	4	0.18	0.340	—
워드 배타적 논리합 부정(2워드)	XNRL	613	4~6	0.26	0.640	—
비트 반전	COM	029	2	0.18	0.240	—
비트 2워드 반전	COML	614	2	0.32	0.440	—

부록-2-12 특수 연산 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
BIN 제곱근 연산	ROTB	620	3	15.4	24.2	—
BCD 제곱근 연산	ROOT	072	3	17.1	25.3	—
숫자 변환	APR	069	4	4.6	5.3	SIN, COS 계산 지정
				5.7	6.9	꺾은선 근사 계산 지정
부동 소수점 제산(BCD)	FDIV	079	4	76	149	—
비트 카운터	BCNT	067	4	0.24	0.360	1CH 카운트

부록-2-13 부동 소수점 변환 및 연산 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU 6□(-EIP)	CJ2M-CPU □□	
부동 소수점→16비트 BIN 변환	FIX	450	3~4	0.13	0.24	—
부동 소수점→32비트 BIN 변환	FIXL	451	3~4	0.13	0.32	—
16비트 BIN→부동 소수점 변환	FLT	452	3~4	0.13	0.30	—
32비트 BIN→부동 소수점 변환	FLTL	453	3~4	0.13	0.32	—
부동 소수점 가산	+F	454	4~6	0.24	0.66	—
부동 소수점 감산	-F	455	4~6	0.24	0.66	—
부동 소수점 제산	/F	457	4~6	0.4	0.9	—
부동 소수점 승산	* F	456	4~6	0.24	0.66	—
각도→라디안 변환	RAD	458	3~4	2.7	3.3	—
라디안→각도 변환	DEG	459	3~4	3.0	3.2	—
SIN 연산	SIN	460	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	475	8~9	0.59	0.86	0, 45, 90도 지정 시
COS 연산	COS	461	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	476	8~9	0.59	0.86	0, 45, 90도 지정 시
TAN 연산	TAN	462	3~4	3.9	4.5	0도 지정 시
				6.1	8.2	45도 지정 시
고속 TAN 연산	TANQ	477	15~16	1.2	1.7	0, 45, 90도 지정 시
SIN-1 연산	ASIN	463	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	464	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	465	3~4	4.0	5.0	0도 지정 시
				5.6	7.0	45도 지정 시
제곱근 연산	SQRT	466	3~4	0.42	0.66	—
지수 연산	EXP	467	3~4	3.8	4.5	—
로그 연산	LOG	468	3~4	5.8	6.5	—
거듭제곱 연산	PWR	840	4~6	35.7	56.6	—
단정도 부동 소수점 데이터 비교	= F	329	3~5	0.13	0.26	—
	< > F	330				
	< F	331				
	< = F	332				
	> F	333				
	> = F	334				
부동 소수점<단>→문자열 변환	FSTR	448	4~5	15.6	23.9	—
문자열→부동 소수점<단> 변환	FVAL	449	3	21.2	31.4	—
부동 소수점 전송<단>	MOVF	469	3~4	0.18	0.20	—

부록-2-14 배정도 부동 소수점 변환 및 연산 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
배정도 부동 소수점 데이터 비교	= D	335	3	5.1	6.7	—
	< > D	336				
	< D	337				
	< = D	338				
	> D	339				
	> = D	340				
부동 소수점→16비트 BIN 변환<배>	FIXD	841	3	5.1	5.4	—
부동 소수점→32비트 BIN 변환<배>	FIXLD	842	3	5.1	5.4	—
16비트 BIN→부동 소수점 변환<배>	DBL	843	3	3.5	4.3	—
32비트 BIN→부동 소수점 변환<배>	DBLL	844	3	3.5	4.3	—
부동 소수점 가산<배>	＋D	845	4	6.0	7.1	—
부동 소수점 감산<배>	-D	846	4	6.1	7.1	—
부동 소수점 승산<배>	* D	847	4	6.1	7.1	—
부동 소수점 제산<배>	/D	848	4	6.4	7.5	—
각도→라디안 변환<배>	RADD	849	3	6.1	6.5	—
라디안→각도 변환<배>	DEGD	850	3	6.0	6.4	—
SIN 연산<배>	SIND	851	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	852	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	853	3	7.3	9.4	0도 지정 시
				27.4	50.3	45도 지정 시
SIN-1 연산<배>	ASIND	854	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	855	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	856	3	6.1	7.4	0도 지정 시
				29.7	36.5	45도 지정 시
제곱근 연산<배>	SQRTD	857	3	16.6	23.4	—
지수 연산<배>	EXPD	858	3	39.7	58.4	—
로그 연산<배>	LOGD	859	3	35.5	52.2	—
거듭제곱 연산<배>	PWRD	860	4	66	99	—

부록-2-15 테이블 데이터 처리 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6 □(-EIP)	CJ2M-CPU □□	
스택 영역 설정	SSET	630	3	7.6	9.4	스택 영역에 5CH 설정
				107	65	스택 영역에 1000CH 설정
스택 데이터 저장	PUSH	632	3	4.9	5.9	—
선입선출	FIFO	633	3	4.8	5.0	스택 영역에 5CH 설정
				231	167	스택 영역에 1000CH 설정
후입선출	LIFO	634	3	5.3	7.1	—
테이블 영역 선언	DIM	631	5	11.1	19.7	—
레코드 위치 설정	SETR	635	4	3.8	5.5	—
레코드 위치 읽기	GETR	636	4	4.6	7.9	—
데이터 검색	SRCH	181	4	13.9	25.0	1CH 검색
				1940	3257	1000CH 검색(*)
바이트 스와핑	SWAP	637	3	10.1	17.5	1CH 교체
				1421	2098	1000CH 교체(*)
최대값 검색	MAX	182	4~5	4.8	5.8	데이터 수 1
				465	672	데이터 수 1000(*)
2워드 데이터 최대값 검색	MAXL	174	4~5	4.8	5.8	데이터 수 1
				465	773	데이터 수 1000(*)
단정도 부동 소수점 데이터 최대값 검색	MAXF	176	4~5	5.2	6.5	데이터 수 1
				682	1090	데이터 수 1000(*)
배정도 부동 소수점 데이터 최대값 검색	MAXD	178	4~5	5.4	6.4	데이터 수 1
				1435	2333	데이터 수 1000(*)
최소값 검색	MIN	183	4~5	4.8	5.8	데이터 수 1
				465	677	데이터 수 1000(*)
2워드 데이터 최소값 검색	MINL	175	4~5	4.8	5.9	데이터 수 1
				189	774	데이터 수 1000(*)
단정도 부동 소수점 데이터 최소값 검색	MINF	177	4~5	5.2	6.5	데이터 수 1
				683	1091	데이터 수 1000(*)
배정도 부동 소수점 데이터 최소값 검색	MIND	179	4~5	5.2	6.4	데이터 수 1
				1402	2303	데이터 수 1000(*)
SUM 값 산출	SUM	184	4	17.5	31.3	1CH 산출
				900	1696	1000CH 산출(*)
FCS 값 산출	FCS	180	4	14.1	25.2	테이블 길이 1
				1235	2089	테이블 길이 1000(*)
스택 데이터 수 출력	SNUM	638	3	4.5	5.3	—
스택 데이터 참조	SREAD	639	4	4.6	5.4	—
스택 데이터 갱신	SWRIT	640	4	4.3	5.0	—
스택 데이터 삽입	SINS	641	4	8.2	9.3	—
				275	256	테이블 길이 1000
스택 데이터 삭제	SDEL	642	4	6.1	7.8	—
				247	180	테이블 길이 1000

* 데이터량에 따라 명령 실행 시간이 크게 변화하며 사이클 시간에 영향을 줍니다. 사이클 시간에 대한 영향을 줄이기 위해 '백그라운드 실행'을 수행할 수 있습니다. 자세한 내용은 '10-2-5 백그라운드 처리 실행 기능'을 참조하십시오.

부록-2-16 추적 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6 □(-EIP)	CJ2M-CPU □□	
부호 없는 1W 레코드 검색	RSRCH	360~364	6	13.9	15.9	레코드 수 1
				504	585	레코드 수 1000
부호 없는 2W 레코드 검색	RSRCH2	370~374	6	14.7	17.6	레코드 수 1
				838	932	레코드 수 1000
부호 없는 4W 레코드 검색	RSRCH4	380~384	6	17.0	19.1	레코드 수 1
				1544	1684	레코드 수 1000
부호 없는 1W 레코드 정렬	RSORT	203	5	149	156	100레코드 분할 정렬 설정 무효 99,98,97,..,0의 데이터를 0,1,2,...,99로 정렬(최악값)
부호 없는 2W 레코드 정렬	RSORT2	204	5	250	249	
부호 없는 4W 레코드 정렬	RSORT4	205	5	457	440	

부록-2-17 데이터 제어 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6 □(-EIP)	CJ2M-CPU □□	
PID 연산	PID	190	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	191	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	680	4~5	10.8	18.3	—
데드 밴드 제어	BAND	681	4~5	11.2	19.2	—
데드 존 제어	ZONE	682	4~5	10.9	17.7	—
시분할 비례 출력	TPO	685	4	6.9	10.2	PF OFF
				37	65	PF ON 듀티 지정 또는 출력 리미트 무효
스케일링	SCL	194	4	7.6	9.3	—
스케일링 2	SCL2	486	4	6.8	9.2	—
스케일링 3	SCL3	487	4	7.8	9.9	—
데이터 평균화	AVG	195	4	22	40	1회 평균
				212	351	64회 평균

부록-2-18 서브루틴 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6 □(-EIP)	CJ2M-CPU □□	
서브루틴 호출	SBS	091	2	0.90	2.8	—
서브루틴 엔트리	SBN	092	2	2.8	4.1	—
서브루틴 반환	RET	093	1	0.43	2.0	—
매크로	MCRO	099	4	16.8	21.7	—
글로벌 서브루틴 호출	GSBS	750	2	0.90	2.8	—
글로벌 서브루틴 엔트리	GSBN	751	2	2.7	3.6	—
글로벌 서브루틴 반환	GRET	752	1	0.43	2.0	—

부록-2-19 인터럽트 제어 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
인터럽트 마스크 세트	MSKS	690	3	10.6	22.1	—
인터럽트 마스크 읽기	MSKR	692	3	9.6	14.8	—
인터럽트 해제	CLI	691	3	10.1	21.6	—
인터럽트 태스크 실행 금지	DI	693	1	10.3	20.4	—
인터럽트 태스크 실행 금지 해제	EI	694	1	9.3	16.0	—

부록-2-20 공정 진행 제어 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
스텝 래더 영역 정의	STEP	008	2	8.7	10.6	스텝 릴레이가 ON일 때
				8.7	9.8	스텝 릴레이가 OFF일 때
스텝 래더 영역 진행	SNXT	009	2	2.2	2.8	—

부록-2-21 I/O 유니트용 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
I/O 리프레시	IORF	097	3	10.1	12.2	1CH IN
				10.5	13.0	1CH OUT
고기능 I/O 유니트 매회 리프레시	FIORF	225	2	*	*	—
CPU 고기능 유니트 매회 I/O 리프레시	DLNK	226	4	234	256	1CH 점유
7세그먼트 디코더	SDEC	078	4	2.5	3.3	—
디지털 스위치	DSW	210	6	24.8	39.6	4자리 지정, 데이터 입력값: 0
				24.8	40.2	8자리 지정, 데이터 입력값: 00
10키 입력	TKY	211	4	7.2	9.7	데이터 입력값: 00
				6.5	8.6	데이터 입력값: FF
16키 입력	HYK	212	5	25.9	40.8	데이터 입력값: 00
				25.9	41.0	데이터 입력값: FF
행렬 입력	MTR	213	5	25.0	38.5	데이터 입력값: 00
				25.0	38.5	데이터 입력값: FF
7세그먼트 표시	7SEG	214	5	31.4	51.9	4자리 지정 시
				34.6	59.4	8자리 지정 시
아날로그 입력 직접 변환 [CJ1W-AD042 전용]	AIDC	216	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 [CJ1W-DA042V 전용]	AODC	217	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 [CJ1W-NC□□4, CJ1W-NC□81 전용]	NCDMV	218	4	81.7	95.3	—
NC 유니트 위치 결정 시작 트리거 [CJ1W-NC□81 전용]	NCDTR	219	3	22.9	25.5	—
인텔리전트 I/O 읽기	IORD	222	4	*	*	—
인텔리전트 I/O 쓰기	IOWR	223	4	*	*	—

* IORD, IORW, FIORF 명령은 읽기 상대의 고기능 I/O 유니트에 따라 다릅니다.

부록-2-22 시리얼 통신 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
프로토콜 매크로	PMCR	260	5	57.8	97.8	직접 지정
				77	132	피연산자 지정 송신 1CH, 수신 1CH
프로토콜 매크로 송신 2	PMCR2	264	6	49.5	96.0	직접 지정
				69	129	피연산자 지정 송신 1CH, 수신 1CH
시리얼 포트 출력	TXD	236	4	57.5	93.8	송신 바이트 수 1바이트
				517	947	송신 바이트 수 256바이트
시리얼 포트 입력	RXD	235	4	79	128	저장 바이트 수 1바이트
				570	1033	저장 바이트 수 256바이트
시리얼 커뮤니케이션 유니트 시리얼 포트 출력 [CJ1W-SCU22/32/42 전용]	TXDU	256	4	75	130	송신 바이트 수 1바이트
시리얼 커뮤니케이션 유니트 시리얼 포트 입력 [CJ1W-SCU22/32/42 전용]	RXDU	255	4	74	128	저장 바이트 수 1바이트
시리얼 커뮤니케이션 유니트 시리얼 포트 직접 송신	DTXDU	262	4	25.8	37.0	송신 바이트 수 1바이트
				179	203	송신 바이트 수 256바이트
시리얼 커뮤니케이션 유니트 시리얼 포트 직접 수신	DRXDU	261	4	27.8	39.7	저장 바이트 수 1바이트
				188	205	저장 바이트 수 256바이트
시리얼 통신 설정 변경	STUP	237	3	233	276	CPU 유니트의 COM 포트 대상

부록-2-23 네트워크 통신 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
네트워크 송신	SEND	090	4	44.3	79.4	—
네트워크 송신 2	SEND2	491	5	43.4	82.8	—
네트워크 수신	RECV	098	4	43.9	79.9	—
네트워크 수신 2	RECV2	492	5	44.5	82.8	—
명령 송신	CMND	490	4	52.7	95.1	—
명령 송신 2	CMND2	493	5	53.0	98.1	—
범용 Explicit 메시지 송신	EXPLT	720	4	78	134	—
Explicit 읽기	EGATR	721	4	74	127	—
Explicit 쓰기	ESATR	722	3	69	117	—
Explicit CPU 유니트 데이터 읽기	ECHRD	723	4	65	110	—
Explicit CPU 유니트 데이터 쓰기	ECHWR	724	4	64	110	—

부록-2-24 파일 메모리용 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
데이터 파일 읽기	FREAD	700	5	217	372	BIN 형식
데이터 파일 쓰기	FWRIT	701	5	216	366	BIN 형식
텍스트 파일 쓰기	TWRIT	704	5	205	370	—

부록-2-25 표시 기능용 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
메시지 표시	MSG	046	3	6.9	10.5	메시지 표시
				6.6	9.5	메시지 표시 해제

부록-2-26 시계 기능 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6 □(-EIP)	CJ2M-CPU □□	
캘린더 가산	CADD	730	4	15.6	22.5	—
캘린더 감산	CSUB	731	4	16.4	24.9	—
시분초→초 변환	SEC	065	3	3.6	4.1	—
초→시분초 변환	HMS	066	3	3.5	4.0	—
시계 보정	DATE	735	2	29.6	53.2	—

부록-2-27 디버그 처리 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6 □(-EIP)	CJ2M-CPU □□	
트레이스 메모리 샘플링	TRSM	045	1	8.9	12.6	1점+0CH 샘플링
				31.6	33.1	31점+6CH 샘플링
				38.8	39.2	31점+16CH 샘플링 지정 시

부록-2-28 고장 진단 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6 □(-EIP)	CJ2M-CPU □□	
운전 계속 고장 진단	FAL	006	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	007	3	—	—	—
고장 지점 검출	FPD	269	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

부록-2-29 특수 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6 □(-EIP)	CJ2M-CPU □□	
세트 캐리	STC	040	1	0.048	0.060	—
클리어 캐리	CLC	041	1	0.048	0.060	—
확장 DM 뱅크 전환	EMBC	281	2	7.6	14.6	—
사이클 시간 감시 시간 설정	WDT	094	2	7.6	17.1	—
조건 플래그 저장	CCS	282	1	5.8	8.3	—
조건 플래그 로드	CCL	283	1	6.4	9.9	—
CV→CS 주소 변환	FRMCV	284	3	9.4	15.7	—
CS→CV 주소 변환	TOCV	285	3~4	10.3	18.2	—
주변 서비스 금지	IOSP	287	1	—	—	—
주변 서비스 금지 해제	IORS	288	1	—	—	—

부록-2-30 블록 프로그램 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
블록 프로그램	BPRG	096	2	7.8	14.1	—
블록 프로그램 종료	BEND	801	1	8.8	13.4	—
블록 프로그램 일시 정지	BPPS	811	2	5.4	8.4	—
블록 프로그램 재기동	BPRS	812	2	3.6	4.8	—
조건부 종료	입력 조건 EXIT	806	1	8.6	13.2	EXIT 실행(입력 조건 ON)
				2.0	2.6	EXIT 실행 안 함 (입력 조건 OFF)
	EXIT 릴레이	806	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	806	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	802	1	1.9	2.4	IF 참(입력 조건 ON)
				3.8	6.4	IF 거짓(입력 조건 OFF)
	IF 릴레이	802	2	3.2	4.0	IF 참(릴레이 ON)
				5.1	8.0	IF 거짓(릴레이 OFF)
조건 분기 블록(부정)	IF NOT 릴레이	802	2	5.1	8.2	IF 참(릴레이 OFF)
				3.2	4.1	IF 거짓(릴레이 ON)
조건 분기 거짓 블록	ELSE	803	1	3.5	5.7	IF 참
				5.3	7.3	IF 거짓
조건 분기 블록 종료	IEND	804	1	5.3	8.5	IF 참
				2.0	2.4	IF 거짓
1스캔 조건 대기	입력 조건 WAIT	805	1	10.0	15.9	WAIT 실행 안 함 (입력 조건 ON)
				1.4	1.9	WAIT 실행(입력 조건 OFF)
	WAIT 릴레이	805	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	805	2	9.2	13.5	WAIT 실행 안 함(릴레이 OFF)
				2.8	3.7	WAIT 실행(릴레이 ON)
100ms 타이머 대기	TIMW	813	3	15.6	22.9	처음 기동할 때
				16.0	23.2	처음 이후
	TIMWX	816	3	15.1	21.7	처음 기동할 때
				16.0	23.2	처음 이후
10ms 타이머 대기	TMHW	815	3	15.7	22.6	처음 기동할 때
				17.5	24.9	처음 이후
	TMHWX	817	3	15.2	22.1	처음 기동할 때
				16.4	23.4	처음 이후
카운터 대기	CNTW	814	4	13.7	20.5	처음 기동할 때
				13.4	19.8	처음 이후
	CNTWX	818	4	13.1	19.5	처음 기동할 때
				13.5	19.7	처음 이후
반복 블록	LOOP	809	1	4.6	9.1	—
반복 블록 종료	입력 조건 LEND	810	1	4.2	8.6	LOOP 실행 안 함 (입력 조건 ON)
				3.9	6.5	LOOP 실행(입력 조건 OFF)
	LEND 릴레이	810	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	810	2	6.7	10.9	LOOP 실행 안 함(릴레이 OFF)
				6.6	8.2	LOOP 실행 안 함(릴레이 ON)

부록-2-31 문자열 처리 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
문자열 전송	MOV$	664	3	31.5	58.3	1문자 전송(*)
문자열 연결	+$	656	4	56	104	1문자+1문자(*)
왼쪽에서 문자열 가져오기	LEFT$	652	4	33.5	62.2	2문자에서 1문자 가져오기(*)
오른쪽에서 문자열 가져오기	RGHT$	653	4	33.4	62.1	2문자에서 1문자 가져오기(*)
임의 위치에서 문자열 가져오기	MID$	654	5	32.3	60.8	3문자 중에서 1문자 가져오기(*)
문자열 검색	FIND$	660	4	30.3	56.3	2문자에서 1문자 검색(*)
문자열 길이 검출	LEN$	650	3	14.0	24.9	1문자 검출(*)
문자열 치환	RPLC$	661	6	110	213	2문자의 첫 번째 문자를 1문자 치환(*)
문자열 삭제	DEL$	658	5	45.6	86.8	2문자의 선두 1문자 삭제(*)
문자열 교환	XCHG$	665	3	40.3	75.4	1문자와 1문자 교환(*)
문자열 클리어	CLR$	666	2	15.9	28.3	1문자 클리어(*)
문자열 삽입	INS$	657	5	85	162	2문자의 첫 번째 문자 뒤에 1문자 삽입(*)
문자열 비교	=$	670	4	27.0	50.9	1문자와 1문자 비교
	<>$	671				
	<$	672				
	<=$	673				
	>$	674				
	>=$	675				

* 데이터량에 따라 명령 실행 시간이 크게 변화하며 사이클 시간에 영향을 줍니다. 사이클 시간에 대한 영향을 줄이기 위해 '백그라운드 실행'을 수행할 수 있습니다. 자세한 내용은 '10-2-5 백그라운드 처리 실행 기능'을 참조하십시오.

부록-2-32 태스크 제어 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
태스크 실행 기동	TKON	820	2	12.5	19.0	사이클릭 태스크 지정 시
				13.6	22.7	추가 태스크 지정 시
태스크 실행 대기	TKOF	821	2	240	393	사이클릭 태스크 지정 시
				15.5	25.8	추가 태스크 지정 시

부록-2-33 기종 변환용 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
블록 전송	XFERC	565	4	6.7	8.2	1CH 전송
				362	409	1000CH 전송
데이터 분배	DISTC	566	4	4.6	5.3	데이터 분배 동작 시
				6.0	7.3	스택 동작 시
데이터 추출	COLLC	567	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	568	4	4.9	5.7	—
비트 카운터	BCNTC	621	4	5.5	6.4	1CH 카운트
				873	974	1000CH 카운트

부록-2-34 펑션 블록용 특수 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
변수 종류 취득	GETID	286	4	7.6	12.5	—

부록-2-35 SFC 제어 명령

명령 이름	니모닉	펑션 번호	명령 길이 (스텝)	실행 시간(μs)		실행 조건
				CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	
스텝 활성화	SA	784	2	9.1	11.8	—
스텝 비활성화	SE	785	2	9.1	11.9	—
스텝 타이머 현재값 읽기	TSR	780	3	3.9	4.2	—
스텝 타이머 현재값 변경	TSW	781	3	5.7	8.2	—
SFC 실행 시작	SFCON	789	2	14.0	20.4	—
SFC 실행 종료	SFCOFF	790	2	249	402	—
SFC 태스크 정지(리셋 있음)	SFCPR	793	2	249	405	—
SFC 태스크 정지(리셋 없음)	SFCPRN	791	2	249	405	—

부록-2-36 펑션 블록 사용 시의 인스턴스 실행 시간

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	
			CJ2H-CPU6□(-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-CPU67-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.



참고

펑션 블록 사용 시의 프로그램 스텝 수

펑션 블록 정의를 작성하고 사용자 프로그램에 인스턴스를 첨부했을 때 프로그램 스텝 수는 다음과 같습니다.

인스턴스가 사이클 시간에 미치는 영향 시간

스텝 수
= 인스턴스 수×(호출부 용량 m＋입출력 파라미터 전송부 용량 n×파라미터 수)
　＋펑션 블록 정의 내 명령 스텝 수 p(*)

* 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.

부록-3 특수 보조 릴레이 목록

특수 보조 릴레이의 주소 순서별 목록은 다음과 같습니다.

A000~A447CH: 읽기 전용　　A448~A1000CH: 읽기/쓰기 가능

부록-3-1 읽기 전용 영역

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A0CH	—	10ms 가산 프리 러닝 타이머	전원 투입 후의 시스템 타이머입니다. 전원을 켜면 0000Hex가 되고 10ms마다 1이 자동으로 가산됩니다. FFFFHex(655,350ms)까지 도달하면 0000Hex로 돌아가서 10ms마다 1씩 자동 가산이 반복됩니다. (*1) 예) 처리 A일 때의 A000CH 값과 다음 처리 B일 때의 A000CH 값의 차를 계산하면 타이머 명령을 사용하지 않고 처리 A에서 처리 B까지의 시간 간격(10ms 단위)을 계산할 수 있습니다.	유지	클리어	전원 ON 후 10ms마다
A1CH	—	100ms 가산 프리 러닝 타이머	전원 투입 후의 시스템 타이머입니다. 전원을 켜면 0000Hex가 되고 100ms마다 1이 자동으로 가산됩니다. FFFFHex(6,553,500ms)까지 도달하면 0000Hex로 돌아가서 100ms마다 1씩 자동 가산이 반복됩니다. 동작 모드를 운전 모드로 변경했을 때도 계속됩니다.	유지	클리어	전원 ON 후 100ms마다
A2CH	—	1s 가산 프리 러닝 타이머	전원 투입 후의 시스템 타이머입니다. 전원을 켜면 0000Hex가 되고 1s마다 1이 자동으로 가산됩니다. FFFFHex(65,535s)까지 도달하면 0000Hex로 돌아가서 1s마다 1씩 자동 가산이 반복됩니다. 동작 모드를 운전 모드로 변경했을 때도 계속됩니다.	유지	클리어	전원 ON 후 1s마다
A50CH	A50.00~A50.07	기본 I/O 유니트 정보 영역(0랙 0슬롯)	기본 I/O 유니트(부하 단락 보호 기능 탑재)의 알람 출력을 표시합니다. CJ1W-OD202에서는 최하위 4비트만 사용할 수 있습니다(2점 단위). CJ1W-OD212, OD204, MD232에서는 최하위 1비트, CJ1W-OD232에서는 최하위 2비트만 사용합니다. (공통 단위) 1: 부하 단락 있음 0: 부하 단락 없음	—	—	사이클마다
	A50.08~A50.15	기본 I/O 유니트 정보 영역(0랙 1슬롯)		—	—	
A51~A69CH	A51.00~A69.15	기본 I/O 유니트 정보 영역(0랙 2슬롯~3랙 9슬롯)		—	—	
A90~A93CH	—	사용자 프로그램 재기입 일시	사용자 프로그램을 재기입한 시각(연월일, 시분초, 요일)이 BCD로 저장됩니다. A90.00~90.07: 초(00~59) A90.08~90.15: 분(00~59) A91.00~91.07: 시(00~23) A91.08~91.15: 일(01~31) A92.00~92.07: 월(01~12) A92.08~92.15: 년(00~99) A93.00~93.07: 요일(00~06) 00: 일요일, 01: 월요일, 02: 화요일, 03: 수요일, 04: 목요일, 05: 금요일, 06: 토요일	유지	유지	—
A94~A97CH	—	파라미터 영역 재기입 일시	파라미터 영역을 재기입한 시각(연월일, 시분초, 요일)이 저장됩니다. 저장 형식은 위의 사용자 프로그램과 동일합니다.	유지	유지	—
A99CH	00	UM 읽기 금지 상태	PLC(사용자 프로그램 전체) 단위로 읽기 금지가 설정되어 있는지 여부를 표시합니다. 0: UM 읽기 금지 설정 없음 1: UM 읽기 금지 설정 있음	유지	유지	보호 설정/해제 시
	01	태스크 읽기 금지 상태	태스크 단위로 읽기 금지가 설정되어 있는지 여부를 표시합니다. 0: 태스크 읽기 금지 설정 안됨 1: 태스크 읽기 금지 설정됨			
	02	읽기 금지 설정 시, 프로그램 덮어쓰기 금지/허가 설정 상태	프로그램의 덮어쓰기 허가/금지 상태를 표시합니다. 0: 허가 중 1: 금지 중			
	03	읽기 금지 설정 시, 메모리 카드에 프로그램 백업 허가/금지 상태	프로그램 파일(.OBJ)의 작성(백업) 허가/금지 상태가 표시됩니다. 0: 허가 중 1: 금지 중			
	14	IR/DR의 태스크 간 동작 표시	IR(인덱스 레지스터) 및 DR(데이터 레지스터)을 태스크 간에 공통으로 사용하는지 여부를 표시합니다. 0: 태스크별로 독립 1: 태스크 간 공통(초기값)	유지	유지	—

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A100~ A199CH	—	이상 내역 저장 영역	이상이 발생하면 이상 내역 정보(고장 코드, 이상 내용 릴레이의 내용, 이상 발생 시각)가 저장됩니다. 최신 20개의 이상 내역 정보를 저장할 수 있습니다. 이상 내역 정보는 5CH분을 점유하며 다음과 같은 내용으로 구성됩니다. 시작 CH: 고장 코드(0~15비트) 시작+1CH: 이상 내용(0~15비트) 이상 릴레이의 내용: 특수 보조 릴레이 관련 채널(관련 채널이 없는 경우는 0000Hex 저장) 시작+2CH: 분(상위 바이트), 초(하위 바이트) 분: 00~59, BCD 초: 00~59, BCD 시작+3CH: 일(상위 바이트), 시(하위 바이트) 일: 01~31, BCD 시: 00~23, BCD 시작+4CH: 년(상위 바이트), 월(하위 바이트) 년: 00~99, BCD 월: 00~12, BCD FALS, FAL 명령 실행에 의한 이상이 발생했을 때도 이 이상 내역에 저장됩니다. 이상 내역 정보는 CX-Programmer에서 조작하여 클리어해야 합니다. 이상이 20개를 초과하면 가장 오래된(A100~104) 이상 데이터가 파기되고 최신 이상 데이터가 A195~199CH에 저장됩니다.	유지	유지	이상 발생 시 A50014 A300CH A400CH
A200CH	A200.11	운전 시작 시 1사이클 ON 플래그	운전을 시작할 때('프로그램' 모드에서 '운전' 또는 '모니터' 모드로 이동 시) 1사이클 1(ON)이 됩니다. 운전 시작 시 1사이클 1(ON)	—	—	—
	A200.12	스텝 래더 1사이클 ON 플래그	STEP 명령에 의한 각 공정의 시작 시에 1회, 1사이클 동안만 1(ON)이 됩니다. 이 플래그를 사용하여 각 공정에 들어갔을 때의 초기화 처리 등을 수행합니다. STEP 명령에 의한 각 공정 시작 시 1사이클 1(ON)	클리어	—	—
	A200.14	태스크 기동 플래그	사이클릭 태스크가 대기 상태(WAIT) 또는 실행되지 않은 상태(INI)에서 실행(RUN)되었을 때 해당 태스크 내 1사이클에서만 ON이 됩니다. 1: WAIT/INI 상태에서 RUN 상태가 된 첫 번째 사이클 0: 기타 태스크 첫회 기동 플래그(A200.15)가 실행되지 않은 상태(INI)에서 실행 상태(RUN)가 됐을 때 ON이 되는데 비해 이 플래그는 대기 상태(WAIT)에서 실행 상태(RUN)가 됐을 때 ON이 됩니다.	클리어	클리어	—
	A200.15	태스크 첫회 기동 플래그	처음으로 '실행 가능'한 상태가 된 태스크가 실제로 실행권을 획득하여 '실행 상태'가 된 동안 1(ON)이 됩니다. 태스크별 초기화 처리를 위해 자체 태스크가 첫회 실행인지 여부를 판단할 때 사용합니다. 1: 첫회 실행 상태 0: 첫회 이외의 실행 가능한 상태 또는 실행하지 않은 상태	클리어	—	—
A201CH	A201.10	온라인 편집 대기 중 플래그	온라인 편집 처리가 대기 중일 때 1(ON)이 됩니다. 1: 온라인 편집 대기 중 0: 온라인 편집 대기 중 아님 온라인 편집 대기 중에 다른 온라인 편집 지시를 받은 경우, 다른 온라인 편집은 기억되지 않고 에러가 발생합니다.	클리어	클리어	A527CH
	A201.11	온라인 편집 처리 중 플래그	온라인 편집 처리가 실행되는 동안 1(ON)이 됩니다. 1: 온라인 편집 실행 중 0: 온라인 편집 비실행 중	클리어	클리어	A527CH
A202CH	A202.00~ A202.07	네트워크 통신 명령 실행 가능 플래그	해당 통신 포트 번호를 사용하여 통신 명령(SEND, RECV, CMND, PMCR, TXDU, RXDU 명령 중 하나) 또는 데이터 백그라운드 처리를 실행할 수 있을 때 1(ON)이 됩니다. 각 비트가 통신 포트 번호에 대응하며 비트 00~07: 통신 포트 번호 0~7입니다. 1: 네트워크 통신 비실행 중 0: 네트워크 통신 실행 중 동일한 통신 포트 번호를 사용하여 여러 개의 네트워크 통신 명령을 실행하는 경우, 이 플래그를 사용하여 배타적 제어를 구성하십시오. 명령을 실행하면 클리어됩니다.	클리어	—	—
	A202.08	CJ2 전용 네트워크 통신 명령 사용 가능 플래그	CJ2 전용 네트워크 통신 명령 사용 가능 플래그이며 초기값=1입니다. 통신 명령 SEND2, CMND2, PMCR2, RECV2만 사용할 수 있습니다.	—	내부 상태 반영 (클리어)	—
	A202.15	네트워크 통신 자동 할당 가능 플래그	통신 포트의 자동 할당 기능을 사용하여 통신 명령(SEND, RECV, CMND, PMCR, TXDU, RXDU 명령 중 하나)을 실행할 수 있을 때 1(ON)이 됩니다. 1: 빈 통신 포트 있음 0: 빈 통신 포트 없음 통신 포트의 자동 할당 기능을 사용하여 동시에 9개 이상의 통신 명령을 실행할 때는 이 플래그를 사용하여 배타적 제어를 구성하십시오.	클리어	—	—

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A203~ A210CH	—	네트워크 통신 종료 코드	네트워크 통신 명령(SEND, RECV, CMND, PMCR 명령 중 하나)이 실행된 경우 응답 코드가 저장됩니다. 백그라운드 처리가 종료되면 클리어(0000Hex 저장)됩니다. 각 CH가 포트 번호에 대응하며 A203~210CH: 통신 포트 번호 0~7입니다. 0000Hex 이외: 에러 코드 0000Hex: 정상 코드 Explicit 메시지 통신 명령(EXPLT, EGATR, ESATR, ECHRD, ECHWR)을 실행하면 다음 코드가 저장됩니다. • Explicit 통신 실행 에러 플래그가 0(OFF)일 때: 0000Hex가 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 코드가 저장됩니다. 통신 실행 중에는 0000Hex가 되고 통신 명령 실행이 종료되면 반영됩니다. 운전 시작 시에 클리어됩니다. 명령을 실행하면 클리어됩니다.	클리어	—	—
A211 CH	—	사용 가능한 CJ2 전용 네트워크 통신 명령 수	사용 가능한 CJ2 전용 네트워크 통신 명령 수가 0이 되면 A202.08이 0(OFF)이 됩니다. 통신 명령 SEND2, CMND2, PMCR2, RECV2만 사용할 수 있습니다. 이 CH를 참조하면 통신 부하를 확인할 수 있습니다.	—	내부 상태 반영 (클리어)	—
A213CH	A213.00~ A213.07	Explicit 통신 실행 에러 플래그	Explicit 메시지 통신 명령(EXPLT, EGATR, ESATR, ECHRD, ECHWR) 실행 시에 에러(이상)가 발생하면 1(ON)이 됩니다. Explicit 메시지 자체가 전송되지 않은 경우, 전송되었지만 Explicit 이상 응답이 발생한 경우에 1(ON)이 됩니다. 각 비트가 통신 포트(내부 논리 포트) 번호에 대응합니다. 비트 00~07: 통신 포트 번호 00~07 1: 이상 종료 0: 정상 종료 다음 Explicit 메시지 통신 명령 실행까지 상태를 유지합니다. 이상 종료가 발생해도 다음 Explicit 메시지 통신 명령 실행에서 0(OFF)이 됩니다.	클리어	—	A219.00~ A219.07 A203CH~ A210CH
A214CH	A214.00~ A214.07	네트워크 통신 완료 시 1사이클 ON 플래그	통신 포트 자동 할당으로 통신 명령을 실행한 경우 통신 완료 시 1사이클만 1(ON)이 됩니다. 각 비트 00~07이 통신 포트 번호 0~7에 대응합니다. 이 플래그는 통신 명령을 실행한 다음 사이클부터 유효가 되므로 통신 명령을 실행한 사이클에서 참조하지 않도록 래더 프로그램에서 1사이클 대기한 후 참조해야 합니다. 1: 통신 완료 시 1사이클만 0: 그 이외 참고: A218CH(사용 통신 포트 번호)에 저장된 사용 통신 포트 번호를 바탕으로 비트 00~07 중 하나를 지정하여 참조합니다.	유지	클리어	—
A215CH	A215.00~ A215.07	네트워크 통신 완료 시 1사이클 ON 에러 플래그	통신 포트 자동 할당으로 통신 명령을 실행한 경우 통신 완료 시에 에러이면 1사이클만 1(ON)이 됩니다. 각 비트 00~07이 통신 포트 번호 0~7에 대응합니다. 이 플래그가 1(ON)이 된 경우, 네트워크 통신 응답 코드(A203~A210CH)를 참조하여 에러 요인을 확인하십시오. 이 플래그는 통신 명령을 실행한 다음 사이클부터 유효가 되므로 통신 명령을 실행한 사이클에서 참조하지 않도록 래더 프로그램에서 1사이클 대기한 후 참조해야 합니다. 1: 통신 완료 에러 시 1사이클만 0: 그 이외 참고: A218CH(사용 통신 포트 번호)에 저장된 사용 통신 포트 번호를 바탕으로 비트 00~07 중 하나를 지정하여 참조합니다.	유지	클리어	—
A216~ A217CH	—	네트워크 통신 응답 코드 저장 주소	통신 포트 자동 할당으로 통신 명령을 실행한 경우, 보조 릴레이의 임의의 채널 주소에 통신 응답 코드가 자동으로 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.	유지	클리어	—
A218CH	—	사용 통신 포트 번호	통신 포트 자동 할당으로 통신 명령을 실행한 경우, 사용한 통신 포트 번호가 여기에 저장됩니다. 0000~0007Hex: 통신 포트 번호 0~7	유지	클리어	—

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A219CH	A219.00~ A219.07	네트워크 통신 실행 에러 플래그	네트워크 통신 명령(SEND, RECV, CMND, PMCR 명령 중 하나) 실행 시에 에러가 발생한 경우 1(ON)이 됩니다. 각 비트가 포트 번호에 대응하며 비트 00~07 : 통신 포트 번호 0~7입니다. 1: 이상(에러) 종료 0: 정상 종료	유지	—	—
A220~ A259CH	A220.00~ A259.15	기본 I/O 유니트의 입력 응답 시간	실제 기본 I/O 유니트의 입력 응답 시간 값을 읽어서 여기에 저장합니다. 0~17Hex 프로그램 모드에서 PLC 시스템 설정의 기본 I/O 유니트 입력 응답 시간 설정을 변경한 경우, 전원을 OFF→ON으로 하지 않는 한 실제 기본 I/O 유니트 내의 값과 PLC 시스템 설정의 값이 다릅니다. 이런 경우, 이 값을 사용하여 실제 값을 모니터링할 수 있습니다.	유지	설명란에 기재	PLC 시스템 설정의 '기본 I/O 유니트의 입력 응답 시간 설정'
A260CH	—	I/O 할당 상태 플래그	현재 자동 할당(상태)인지, 수동 할당(상태)인지 확인할 수 있습니다. 0000Hex: 자동 할당(상태) BBBBHex: 수동 할당(상태)	유지	유지	—
A261CH	A261.00	I/O 테이블 작성 에러 상세 정보	CPU 고기능 시스템 설정 영역 초기화 이상에 의해 I/O 테이블 작성 에러가 발생했을 때 1(ON)이 됩니다. 1: CPU 고기능 시스템 설정 영역 초기화 이상이 발생한 경우 1→0: I/O 테이블 작성 조작 정상 종료 시	유지	클리어	I/O 테이블 작성 조작 시
	A261.02		I/O 지점 수 초과에 의해 I/O 테이블 작성 에러가 발생했을 때 1(ON)이 됩니다. 1: I/O 지점 수를 초과하는 유니트 구성인 경우 1→0: I/O 테이블 작성 조작 정상 종료 시			A401.11(I/O 지점 수 초과 플래그)
	A261.03		번호 이중 사용 에러에 의해 I/O 테이블 작성 에러가 발생했을 때 1(ON)이 됩니다. 1: 번호 이중 사용 에러가 되는 구성인 경우 1→0: I/O 테이블 작성 조작 정상 종료 시			A401.13(번호 이중 사용 에러 플래그)
	A261.04		I/O 버스 이상에 의해 I/O 테이블 작성 에러가 발생했을 때 1(ON)이 됩니다. 1: I/O 버스 이상이 발생한 경우 1→0: I/O 테이블 작성 조작 정상 종료 시			A401.14(I/O 버스 이상 플래그)
	A261.06		잘못된 SYSBUS 인식으로 인해 I/O 테이블 작성 에러가 발생했을 때 1(ON)이 됩니다. 1: SYSBUS 인식이 잘못되어 종료한 경우 1→0: I/O 테이블 작성 조작 정상 종료 시			—
	A261.07		고기능 유니트 이상에 의해 I/O 테이블 작성 에러가 발생했을 때 1(ON)이 됩니다. 1: 고기능 유니트 이상이 발생한 경우 1→0: I/O 테이블 작성 조작 정상 종료 시			—
	A261.09		I/O 인식이 완료되지 않아 I/O 테이블 작성 에러가 발생했을 때 1(ON)이 됩니다. 1: I/O 인식이 완료되지 않은 경우 1→0: I/O 테이블 작성 조작 정상 종료 시			—
A262~ A263CH	—	사이클 시간 최대값 (0.1ms 단위)	사이클 시간의 0.1ms 단위 최대값이 BIN 32비트로 사이클마다 저장됩니다(A262CH: 하위 데이터, A263CH: 상위 데이터). 0~FFFFFFFFHex: 0~429496729.5ms(0.1ms 단위)	클리어	클리어	—
A264~ A265CH	—	사이클 시간 현재값 (0.1ms 단위)	사이클 시간의 0.1ms 단위 현재값이 BIN 32비트로 사이클마다 저장됩니다(A264CH: 하위 데이터, A265CH: 상위 데이터). 0~FFFFFFFFHex: 0~429496729.5ms(0.1ms 단위)	클리어	클리어	—
A266~ A267CH	—	사이클 시간 현재값 (0.01ms 단위)	사이클 시간의 0.01ms 단위 현재값이 BIN 32비트로 사이클마다 저장됩니다(A266CH: 하위 데이터, A267CH: 상위 데이터). 0~FFFFFFFFHex: 0~42949672.95ms(0.01ms 단위)	클리어	클리어	—
A293CH	—	버전 이상 상세 정보	전송된 사용자 프로그램에 CPU 유니트의 유니트 버전에서 사용할 수 없는 기능이 포함된 경우 값이 설정됩니다. 0000 Hex: 이상 없음 0001 Hex: 이상 있음	클리어	클리어	운전 시작 시
A294CH	—	프로그램 정지 시 태스크 번호	프로그램 에러에 의해 실행 정지된 경우 정지 위치의 태스크 유형 및 태스크 번호가 저장됩니다. 사이클릭 태스크: 0000~007FHex(태스크 번호 0~127) 인터럽트 태스크: 8000~80FFHex(태스크 번호 0~255) 해당 태스크의 프로그램 중 어느 프로그램 주소에서 정지했는지는 A298CH(프로그램 주소 하위), A299CH(프로그램 주소 상위)에 저장됩니다.	클리어	클리어	A298/A299CH

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A295CH	A295.08	명령 처리 에러 플래그	PLC 시스템 설정의 명령 에러 발생 시 동작을 정지로 설정해서 명령 처리 에러가 발생한 경우, 에러 플래그(ER 플래그) ON과 동시에 1(ON)이 됩니다. 이 에러 플래그가 ON이면 CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED가 점등됩니다. 1: ER 플래그 ON 0: ER 플래그 OFF	클리어	클리어	• PLC 시스템 설정의 '명령 에러 발생 시 동작 설정' • 정지한 태스크 번호와 프로그램 주소는 A294CH 및 A298/299CH에 저장됩니다.
	A295.09	DM/EM 간접 지정 BCD 에러 플래그	PLC 시스템 설정의 명령 에러 발생 시 동작을 정지로 설정한 경우 DM/EM BCD 간접 지정에서 데이터가 BCD가 아닐 때, 액세스 에러 플래그(AER 플래그) ON과 동시에 1(ON)이 됩니다. 이 에러 플래그가 ON이면 CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED가 점등됩니다. 1: BCD 데이터가 아님 0: 정상			
	A295.10	잘못된 영역 액세스 에러 플래그	PLC 시스템 설정의 명령 에러 발생 시 동작을 정지로 설정한 경우 잘못된 영역을 액세스했을 때, 액세스 에러 플래그(AER 플래그) ON과 동시에 1(ON)이 됩니다. 이 에러 플래그가 ON이면 CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED가 점등됩니다. 잘못된 영역 액세스로 인식되는 조건은 다음과 같습니다. 시스템 사용 영역에 대한 읽기/쓰기 EM 파일 메모리화 영역에 대한 읽기/쓰기 쓰기 방지 영역에 대한 쓰기 DM/EM 간접 지정(BCD 모드)에서 BCD 에러입니다. 1: 잘못된 영역에 대한 액세스 발생 0: 정상			
	A295.11	END 없음 플래그	태스크 내의 각 프로그램에 END 명령이 존재하지 않을 때 1(ON)이 됩니다. CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 1: END 없음 0: END 있음			정지한 태스크 번호와 프로그램 주소는 A294CH 및 A298/299CH에 저장됩니다.
	A295.12	태스크 에러 플래그	태스크 에러가 발생했을 때 1(ON)이 됩니다. 태스크 에러는 다음과 같은 내용을 가리킵니다. 실행 가능한 상태(기동 중)인 사이클릭 태스크가 하나도 존재하지 않는 경우 태스크에 할당된 프로그램이 존재하지 않는 경우 1: 이상 0: 정상			
	A295.13	미분 초과 플래그	미분 명령에 대응하는 미분 플래그 번호가 규정값을 초과했을 때 1(ON)이 됩니다. CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 1: 이상 0: 정상			
	A295.14	잘못된 명령 플래그	실행할 수 없는 프로그램이 저장된 경우 1(ON)이 됩니다. CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 1: 이상 0: 정상			
	A295.15	UM 초과 플래그	UM(사용자 메모리)의 최종 주소를 초과해서 명령을 실행하려고 한 경우 1(ON)이 됩니다. CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 1: 이상 0: 정상			
A298~ A299CH	—	프로그램 정지 시 명령 프로그램 주소	프로그램 에러에 의해 실행 정지된 경우 해당 정지 위치의 명령에 대한 프로그램 주소를 BIN으로 저장합니다.	클리어	클리어	정지한 태스크 번호는 A294CH에 저장됩니다.
A300CH	—	이상 내역 저장 포인터	이상 내역 저장 영역(A100~199CH)에 대해 이상 발생 시 이상 내역 정보가 저장될 때마다 가산(+1)하며, 다음에 저장하는 위치를 이상 내역 저장 영역의 선두 위치에서의 오프셋 데이터로서 BIN 데이터로 나타냅니다. 00~14Hex A500.14(이상 내역 정보 클리어 플래그)가 ON 기동하면 클리어됩니다. 이 값이 20(14Hex)이 된 경우, 다음 이상 발생 시에 이상 내역 정보가 저장되는 위치의 선두는 195CH가 됩니다.	유지	유지	이상 발생 시 A500.14
A301CH	—	현재 EM 뱅크	현재 EM 뱅크 번호가 BIN으로 설정됩니다. 현재 EM 뱅크 번호는 EMBC(확장 DM 뱅크 전환) 명령에 따라 전환됩니다. 0000~0018Hex	클리어	클리어	—

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A302CH	A302.00~ A302.15	CPU 고기능 유니트 초기화 중 플래그	CPU 고기능 유니트 재시작 플래그(A501.00~A501.15)가 기동되거나 전원을 켰을 때, CPU 고기능 유니트가 초기 처리(초기화)에 들어가며 그 동안은 1(ON)이 됩니다. 각 비트가 유니트 번호에 대응합니다. 비트 00~15: 0~15(유니트 번호) 초기(초기화) 처리 중에는 해당 CPU 고기능 유니트의 리프레시 데이터를 사용하지 않도록 프로그램을 작성하십시오. 또한 초기(초기화) 처리 중에는 IORF 명령 및 FIORF 명령은 실행되지 않습니다. 0: 초기 처리 중이 아님 1: 초기 처리 중(초기 처리 종료 후, 시스템에 의해 자동으로 0) 초기(초기화) 처리 종료 후 시스템에 의해 자동으로 0(OFF)이 됩니다.	유지	클리어	초기(초기화) 처리 시 A50100
A315 CH	A315.12	태그 메모리 이상 플래그(운전 계속 이상) CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당	네트워크 변수가 저장된 태그 메모리에 이상이 발생했을 때 1(ON)이 됩니다.	유지	클리어	—
	A315.13	옵션 보드 이상 발생 중 플래그	장착된 옵션 보드가 통전 중에 빠진 경우 또는 지원 대상이 아닌 옵션 보드가 장착된 경우 1(ON)이 됩니다. 0: 정상 1: 이상	클리어	클리어	• 전원 ON 시 • 사이클마다(공통 처리)
	A315.15	백업 메모리 이상 플래그 (운전 계속 이상)	내장 플래시 메모리 내의 백업 데이터 영역, 소스/코멘트 영역에 대한 쓰기에 실패했을 때 1(ON)이 됩니다. 해당하는 이상이 해제되면 0(OFF)이 됩니다.	유지	클리어	—
A330~ A335CH	A330.00~ A335.15	고기능 I/O 유니트 초기화 중 플래그	고기능 I/O 유니트 재시작 플래그(A502.00~507.15)가 기동되거나 전원을 켰을 때, 고기능 I/O 유니트가 초기 처리(초기화)에 들어가며 그 동안은 1(ON)이 됩니다. 각 비트가 기기 번호에 대응합니다. A330.00~330.15: 0~15호기 A331.00~331.15: 16~31호기 .... A335.00~335.15: 80~95호기 초기(초기화) 처리 중에는 해당 고기능 I/O 유니트의 리프레시 데이터를 사용하지 않도록 프로그램을 작성하십시오. 또한 초기(초기화) 처리 중에는 IORF 명령 및 FIORF 명령은 실행되지 않습니다. 0: 초기 처리 중이 아님 1: 초기 처리 중(초기 처리 종료 후, 시스템에 의해 자동으로 0) 초기(초기화) 처리 종료 후 시스템에 의해 자동으로 0(OFF)이 됩니다.	유지	클리어	A502.00~ A507.15
A336CH	A336.00~ A336.15	전원 ON 시 인식 완료 유니트 수 (0~3랙)	전원 ON 시 CPU 유니트가 인식한 유니트의 대수가 16진수 1자리로 랙별로 저장됩니다. (0~AHex) 0랙: A336.00~A336.03 1랙: A336.04~A336.07 2랙: A336.08~A336.11 3랙: A336.12~A336.15 예: 0랙 1대, 1랙 4대, 2랙 8대, 3랙 10대를 인식한 경우 다음과 같이 저장됩니다. A336CH | A | 8 | 4 | 1 |	유지	클리어	—
A339~ A340CH	—	미분 플래그 번호 최대값	미분 명령에서 사용된 미분 플래그 번호의 최대값이 저장됩니다.	설명란에 기재	클리어	운전 시작 시 A29513

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A343CH	A34300~ A34302	메모리 카드 종류	장착된 메모리 카드의 종류가 저장됩니다. 0: 없음 4: 플래시 ROM	유지	설명란에 기재	전원 ON 시 또는 메모리 카드 급전 스위치 기동 시
	A343.06	EM 파일 메모리화 영역의 포맷 이상 플래그	EM 파일 메모리화 시작 뱅크 번호의 영역이 포맷 이상일 때 1(ON)이 됩니다. 1: 포맷 이상 있음 0: 포맷 이상 없음 초기(초기화) 처리 종료 후 시스템에 의해 자동으로 0(OFF)이 됩니다.)	유지	클리어	—
	A343.07	메모리 카드 포맷 이상 플래그	메모리 카드 장착 시, 메모리 카드가 포맷되지 않았을 때 또는 포맷 이상일 때 1(ON)이 됩니다. 1: 메모리 카드가 포맷되지 않았거나 포맷에 이상 있음 0: 맷 이상 없음(포맷 정상) 포맷이 정상적으로 종료되면 0(OFF)으로 되돌아갑니다. 초기(초기화) 처리 종료 후 시스템에 의해 자동으로 0(OFF)이 됩니다.	유지	설명란에 기재	전원 ON 시 또는 메모리 카드 급전 스위치 기동 시
	A343.08	파일 쓰기 이상 플래그	파일 쓰기에서 이상이 발생했을 때 1(ON)이 됩니다. 1: 이상 있음 0: 이상 없음	유지	클리어	파일 쓰기 실행 시
	A343.09	파일 쓰기 불가 플래그	파일 쓰기 실행 시, 쓰기 방지되어 있거나 용량 초과 때문에 쓰기 작업을 수행할 수 없는 경우 1(ON)이 됩니다. 1: 쓰기 불가 0: 쓰기 가능 또는 쓰기 처리 실행 시	유지	클리어	파일 쓰기 실행 시
	A343.10	파일 읽기 이상 플래그	파일을 읽을 때 이상이 발생한 경우(파일이 손상되거나 파일을 읽을 때 잘못된 데이터가 있는 경우) 1(ON)이 됩니다. 1: 읽기 불가 0: 읽기 가능 또는 읽기 처리 실행 시	유지	클리어	파일 쓰기 실행 시
	A343.11	파일 없음 플래그	파일을 읽을 때 읽어올 파일이 존재하지 않는 경우 또는 파일을 쓸 때 지정한 디렉토리가 존재하지 않는 경우 1(ON)이 됩니다. 1: 읽어올 파일이 없거나 쓰기를 실행할 디렉토리가 없음 0: 읽어올 파일이 있거나, 쓰기를 실행할 디렉토리가 있거나, 읽기 처리 실행 중	유지	클리어	파일 읽기 실행 시
	A343.13	파일 메모리 조작 실행 중 플래그	다음 중 하나의 파일 메모리 관련 조작 실행 중에 1(ON)이 됩니다. 아무 것도 실행하지 않을 때는 0(OFF)이 됩니다. 자체 CPU 유니트 대상 CMND 명령 실행 중 파일 메모리 조작 명령 실행 특수 보조 릴레이에 의한 프로그램 교체 중 간이 백업 조작 실행 중 1: 실행 중 0: 비실행 중	유지	클리어	파일 메모리 관련 명령 실행 시
	A343.14	파일 액세스 중 플래그	파일 데이터를 액세스하는 중에 1(ON)이 됩니다. 1: 액세스 중 0: 액세스 중이 아님 파일 메모리 관련 명령은 동시에 실행할 수 없으므로 이 플래그를 사용하여 배타적으로 제어해야 합니다.	유지	클리어	—
	A343.15	메모리 카드 인식 완료 플래그	메모리 카드를 인식할 수 있으면 1(ON)이 됩니다. 메모리 카드를 인식할 수 없으면 0(OFF)이 됩니다. 1: 메모리 카드 인식 완료 0: 메모리 카드 인식 못함	유지	클리어	메모리 카드를 장착했을 때 또는 전원 ON 시
A344CH	A344.00~ A344.07	EM 파일 메모리화 또는 트레이스 메모리화 시작 뱅크 번호	해당 시점의 EM 파일 메모리 영역 또는 EM 트레이스 메모리 영역의 시작 뱅크 번호가 저장됩니다. 또한 EM 영역은 시작 뱅크 번호 이후 최종 뱅크 번호까지 파일 메모리 또는 트레이스 메모리로 사용됩니다. 파일 메모리 또는 트레이스 메모리로 사용되지 않은 경우는 FFFFHex가 됩니다. EM 영역을 파일 메모리 또는 트레이스 메모리로 사용하려면 CX-Programmer에서 [PLC]－[메모리 할당]－[EM 메모리 설정]에서 '파일' 또는 '트레이스'를 선택합니다.	유지	유지	CX-Programmer의 [PLC]－[메모리 할당]－[EM 메모리 설정]
	A344.14	EM 트레이스 메모리 지정 플래그	A344CH가 FFFFHex 이외일 때, 이 플래그가 1(ON)일 때, 하위 바이트(A344.00~344.07)의 뱅크 번호 이후가 EM 트레이스 메모리로 사용됨을 나타냅니다.	유지	유지	
	A344.15	EM 파일 메모리 지정 플래그	A344CH가 FFFFHex 이외일 때, 이 플래그가 1(ON)일 때, 하위 바이트(A344.00~344.07)의 뱅크 번호 이후가 EM 파일 메모리로 사용됨을 나타냅니다.	유지	유지	

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A345CH	A345.00	FB 프로그램 소스 정보 유무 플래그	소스/코멘트 메모리 내에 FB 프로그램 소스 정보가 있을 때 1(ON)이 됩니다. 1: 있음 0: 없음	유지	내부 상태 반영	CX-Programmer에서 전송 시
	A345.01	변수 테이블 정보 유무 플래그	소스/코멘트 메모리 내에 변수 테이블 정보가 있을 때 1(ON)이 됩니다. 1: 있음 0: 없음	유지	내부 상태 반영	
	A345.02	코멘트 정보 유무 플래그	소스/코멘트 메모리 내에 코멘트 정보가 있을 때 1(ON)이 됩니다. 1: 있음 0: 없음	유지	내부 상태 반영	
	A345.03	프로그램 인덱스 정보 유무 플래그	소스/코멘트 메모리 내에 프로그램 인덱스 정보가 있을 때 1(ON)이 됩니다. 1: 있음 0: 없음	유지	내부 상태 반영	
	A345.05	SFC 프로그램 소스 정보 유무 플래그	소스/코멘트 메모리 내에 SFC 프로그램 소스 정보가 있을 때 1(ON)이 됩니다. 1: 있음 0: 없음	유지	내부 상태 반영	
	A345.07	네트워크 변수(태그) 정보 유무 플래그	태그 메모리 내에 네트워크 변수(태그) 정보가 있을 때 1(ON)이 됩니다. 1: 있음 0: 없음	유지	내부 상태 반영	
A346~ A347CH	—	남은 데이터 파일 전송 데이터 수	데이터 파일 전송 상황을 알리기 위한 2워드 채널(32비트)입니다. 데이터 파일 읽기(FREAD) 명령 또는 데이터 파일 쓰기(FWRIT) 명령을 실행하기 시작할 때, 지정된 파일 메모리에서 지정된 파일명의 데이터 파일에 대해 전송(읽기 또는 쓰기)해야 하는 명령 피연산자에 의한 설정 채널 수가 저장됩니다. A346CH: 하위, A347CH: 상위, 32비트 BIN 값입니다. 이 영역의 값을 확인하면 지정된 채널 수의 전송이 실제로 성공했는지 여부를 확인할 수 있습니다.	유지	클리어	FREAD 명령 또는 FWRIT 명령 실행 시 저장 실제 전송 시 감산
A351~ A354CH	—	시계 데이터	CPU 유니트의 내부 시계 데이터가 BCD로 저장됩니다. CX-Programmer에 의한 설정, DATE(시계 보정) 명령 실행, FINS 명령의 시간 정보 쓰기(0702Hex) 중 하나를 통해 설정합니다. 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	유지	유지	사이클 처리
A360~ A391CH	A360.01~ A391.15	실행 FAL 번호	FAL 명령이 실행되면 해당 비트가 1(ON)이 되며 A36001~39115의 각 비트가 FAL001~511에 대응합니다. 1: 해당 FAL 명령 실행 0: 해당 FAL 명령 비실행 이상 해제를 수행하면 0(OFF)이 됩니다.	유지	클리어	이상 발생 시 A40215
A392CH	A392.04	시리얼 포트 통신 이상 플래그	시리얼 포트에서 통신 이상이 발생했을 때 1(ON)이 됩니다. 툴 버스 모드, NT 링크(1 : N) 모드 또는 시리얼 PLC 링크 친국/자국 모드일 때는 참조하지 마십시오. 1: 이상 0: 정상	유지	클리어	이상 발생 시
	A392.05	시리얼 포트 송신 준비 플래그 (무수순 모드)	무수순 모드에서 시리얼 포트가 송신 가능한 상태일 때 1(ON)이 됩니다. 1: 송신 가능 0: 송신 불가	유지	클리어	송신 완료 시
	A392.06	시리얼 포트 수신 완료 플래그 (무수순 모드)	무수순 모드에서 시리얼 포트가 수신을 완료했을 때 1(ON)이 됩니다. 수신 바이트 수 지정 시: 지정한 바이트 수를 수신하면 ON 종료 코드 지정 시: 종료 코드를 수신하면 ON 또는 종료 코드를 수신하지 않은 경우 256바이트를 수신하면 ON	유지	클리어	수신 완료 시
	A392.07	시리얼 포트 수신 초과 플래그 (무수순 모드)	무수순 모드에서 시리얼 포트가 수신 데이터 수를 초과하여 수신했을 때 1(ON)이 됩니다. 수신 바이트 수 지정 시: 수신 완료 후 RXD 명령을 실행하기 전에 데이터를 수신하면 ON 종료 코드 지정 시: 종료 코드를 수신한 경우 RXD 명령을 실행하기 전에 데이터를 수신하면 ON. 종료 코드를 수신하지 않은 경우(256바이트 데이터 수신 시), 257바이트째에 종료 코드 이외의 데이터를 수신했을 때 ON 1: 수신 데이터 수 초과 0: 수신 데이터 수 초과 안 함	유지	클리어	

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A393CH	A393.00~ A393.07	시리얼 포트에서 PT와의 통신 실행 중 플래그	시리얼 포트가 NT 링크 모드일 때 또는 시리얼 PLC 링크 모드일 때, 통신을 실행하고 있는 해당 비트가 1(ON)이 됩니다. 비트 0~7이 기기 번호 0~7에 대응합니다. 1: 통신 실행 중 0: 통신 비실행 중	유지	클리어	토큰에 대한 정상 응답이 있을 때
	A393.08~ A393.15	시리얼 포트에서 PT 우선 등록 중 플래그	시리얼 포트가 NT 링크 모드일 때 우선 등록된 PT의 해당 비트가 1(ON)이 됩니다. 비트 0~7이 기기 번호 0~7에 대응합니다. 1: 우선 등록 있음 0: 우선 등록 없음	유지	클리어	우선 등록 명령 수신 시
	A393.00~ A393.15	시리얼 포트 수신 카운터(무수순 모드만 해당)	시리얼 포트가 무수순 모드일 때 수신한 데이터의 바이트 수를 바이너리로 표시합니다.	유지	클리어	데이터 수신 시
A395CH	A395.06	파일 삭제 통지 플래그	메모리 카드의 파일을 갱신 중일 때 전원 OFF가 발생한 경우, 다음에 전원이 켜지면 시스템에서 갱신 중이던 파일을 자동으로 삭제합니다. 이때, 이 플래그가 1(ON)이 됩니다. 1: 삭제 실행 0: 삭제 비실행	클리어	클리어	시스템에 의한 파일 삭제 실행 시
	A395.07		EM 파일 메모리의 파일을 갱신 중일 때 전원 OFF가 발생한 경우, 다음에 전원이 켜지면 시스템에서 갱신 중이던 파일을 자동으로 삭제합니다. 이때, 이 플래그가 1(ON)이 됩니다. 1: 삭제 실행 0: 삭제 비실행	클리어	클리어	시스템에 의한 파일 삭제 실행 시
	A395.10	백그라운드 처리 시 ER/AER 플래그	백그라운드 처리 중에 명령 처리 에러/잘못된 영역 액세스 에러가 발생한 경우, 이 플래그가 1(ON)이 됩니다. 0: 에러 없음(백그라운드 처리 시작 시 0) 1: 에러 있음	클리어	클리어	—
	A395.11	메모리 손상 통지 플래그	전원이 켜졌을 때 메모리 손상이 발생했다고 인식되는 경우 1(ON)이 됩니다. 1: 메모리 손상 발생 0: 정상	유지	설명란에 기재	전원 ON 시
	A395.12	딥 스위치 6 상태 플래그	CPU 유니트 전면의 딥 스위치 SW6의 ON/OFF 상태가 사이클마다 저장됩니다. 1: 딥 스위치 SW6 ON 0: 딥 스위치 SW6 OFF	유지	설명란에 기재	사이클마다
A400CH	—	고장 코드	운전 계속 이상 발생 시(사용자 정의 FAL 명령 실행 또는 시스템의 운전 계속 이상 발생 시) 또는 운전 정지 이상 발생 시(사용자 정의 FALS 명령 실행 또는 시스템의 운전 정지 이상 발생 시), 고장 코드(🕮 '부록-3-3 특수 보조 릴레이 보충 설명')가 저장됩니다. 여러 개가 동시에 발생한 경우에는 가장 중요도가 높은 고장 코드가 저장됩니다.	클리어	클리어	이상 발생 시

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A401CH	A401.03	카드 전송 이상 플래그	• 전원을 켰을 때 메모리 카드의 내용을 CPU 유니트에 정상적으로 읽어올 수 없는 경우, 1(ON)이 됩니다. • 전원을 켰을 때 네트워크 변수가 포함되는 사용자 프로그램(.obj 파일)을 전송하려고 한 경우 1(ON)이 됩니다 (CJ2H-CPU6□, CJ2M-CPU1□만 해당).	이상 해제에 의해 클리어	클리어	—
	A401.05	버전 이상 플래그	전송된 사용자 프로그램에 CPU 유니트의 유니트 버전에서 사용할 수 없는 기능이 포함된 경우 1(ON)이 됩니다.	이상 해제에 의해 클리어	클리어	—
	A401.06	FALS 이상 플래그 (운전 정지 이상)	FALS 명령(운전 정지 이상)이 실행되었을 때 1(ON)이 됩니다. CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 실행된 FALS 번호 001~511에 대해 고장 코드 C101~C2FF가 A400CH에 저장됩니다. 1: FALS 명령 실행 0: FALS 명령 비실행 이상 해제를 수행하면 0(OFF)이 됩니다.	클리어	클리어	이상 발생 시 A400CH
	A401.08	사이클 시간 초과 플래그 (운전 정지 이상)	사이클 시간 현재값이 PLC 시스템 설정의 '사이클 시간 감시 시간'을 초과했을 때 1(ON)이 됩니다. CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 0: 사이클 시간 현재값사이클 시간 감시 시간 1: 사이클 시간 현재값 > 사이클 시간 감시 시간 이상 해제를 수행하면 0(OFF)이 됩니다.	클리어	클리어	사이클 시간이 초과되었을 때 PLC 시스템 설정 [사이클 시간 감시 시간]
	A401.09	프로그램 에러 플래그 (운전 정지 이상)	프로그램 내용이 잘못되었을 때 1(ON)이 됩니다. CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 동시에 A298/299CH에 실행 정지 시의 명령 위치(프로그램 주소)가 저장됩니다. 발생한 프로그램 에러의 종류는 A295CH의 비트 8~15에서 확인 가능합니다. 프로그램 에러의 내용은 A295CH 또는 '5-9 작성한 프로그램 확인'을 참조하십시오. 프로그램 에러에 의해 실행을 정지한 정지 위치의 태스크 유형/번호는 A294CH에 저장됩니다. 1: 이상 0: 정상 이상 해제를 수행하면 0(OFF)이 됩니다.	클리어	클리어	A294, A295, A298~A299CH

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A401CH	A401.10	I/O 설정 이상 플래그 (운전 정지 이상)	• 등록 I/O 테이블의 유니트 기종과 실제 I/O 테이블(실제로 연결됨)의 유니트 기종이 다를 때 1(ON)이 됩니다. • 인터럽트 입력 유니트가 다음의 범위 밖인 CPU 장치의 슬롯에 장착될 때 1(ON)이 됩니다. • CJ2H-CPU6□-EIP: CPU 장치의 슬롯 번호 0~3 • CJ2H-CPU6□, CJ2M-CPU□□: CPU 장치의 슬롯 번호 0~4 • CJ2H-CPU6□-EIP, CJ2M-CPU3□용 등록 I/O 테이블이 CJ2H-CPU6□, CJ2M-CPU1□에 다운로드된 경우 또는 CJ2H-CPU6□, CJ2M-CPU1□용 등록 I/O 테이블이 CJ2H-CPU6□-EIP, CJ2M-CPU3□에 다운로드된 경우 1(ON)이 됩니다. 1(ON)이 되면 CPU 유니트의 운전은 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 1: 이상 0: 정상 이상 해제를 수행하면 0(OFF)이 됩니다.	클리어	클리어	A405.08
	A401.11	I/O 지점 수 초과 플래그 (운전 정지 이상)	등록 I/O의 I/O 지점 수가 초과된 경우(사용하는 기본 I/O 유니트의 I/O 총 채널 수가 기종에 따라 규정된 최대값을 초과했을 때) 또는 1랙(장치)에 11대 이상의 유니트가 연결된 경우 1(ON)이 됩니다. CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 1: 이상 0: 정상 이상 해제를 수행하면 0(OFF)이 됩니다.	클리어	클리어	A407CH
	A401.13	유니트 번호 이중 사용 에러 플래그/ 증설 번호 이중 사용 에러 플래그 (운전 정지 이상)	다음과 같은 경우 1(ON)로 설정됩니다. • 유니트 번호 이중 사용 에러 CPU 고기능 유니트의 기기 번호가 중복되는 경우 고기능 I/O 유니트의 기기 번호가 중복되는 경우 • 증설 번호 이중 사용 에러 기본 I/O 유니트의 할당 채널이 중복되는 경우 또한 위와 같은 경우 CPU 유니트의 운전은 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 어느 기기 번호가 중복되어 있는지는 A409~A416에 반영됩니다. 1: 중복됨 0: 중복 안됨 이상 해제를 수행하면 0(OFF)이 됩니다.	클리어	클리어	A409~A416CH
	A401.14	I/O 버스 이상 플래그 (운전 정지 이상)	CPU 유니트와 각 슬롯에 장착된 유니트 간의 데이터 전송에 이상이 발생한 경우 또는 엔드 커버가 CPU 장치 또는 증설 장치의 우측에 연결되지 않은 경우, 1(ON)이 됩니다. CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 1: 이상 0: 정상 이상 해제를 수행하면 0(OFF)이 됩니다. A404.00~A404.07에 I/O 버스 이상 슬롯 번호(00~09)가 BIN으로 저장됩니다. 엔드 커버가 연결되지 않은 경우는 0EHex가 저장됩니다. A404.08~A404.15에 I/O 버스 이상의 랙 번호(00~07)가 BIN으로 저장됩니다. CJ2H-CPU6□-EIP, CJ2M-CPU3□의 내장 네트워크에서 버스 이상이 발생한 경우에는 0BHex가 저장됩니다. 엔드 커버가 연결되지 않은 경우는 0EHex가 저장됩니다.	클리어	클리어	A404CH
	A401.15	메모리 이상 플래그 (운전 정지 이상)	메모리에 이상이 발생한 경우 또는 메모리 카드에서 전원 ON 시 자동 전송에 실패한 경우 1(ON)이 됩니다. CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 1: 이상 0: 정상 메모리 이상 발생 위치가 A403.00~403.08에 저장됩니다. 전원 ON 시 자동 전송에 실패한 경우는 A403.09가 1(ON)이 됩니다. 이상 해제를 수행하면 0(OFF)이 됩니다. 전원 ON 시 자동 전송 이상이 발생한 경우는 이상을 해제할 수 없습니다.	클리어	클리어	A403CH

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A402CH	A402.00	기타 운전 계속 이상 (시스템 작업 메모리 이상)	• A402.01~A402.15에 할당된 운전 계속 이상 이외의 운전 계속 이상이 발생한 경우, 1(ON)이 됩니다. 예) 온라인 편집용 메모리에 이상 발생 • 그 밖에 계속 이상에 대한 자세한 내용은 A315CH에 할당됩니다.	이상 해제에 의해 클리어	클리어	
	A402.02	고기능 I/O 유니트 설정 이상 플래그 (운전 계속 이상)	등록 I/O 테이블에 등록된 고기능 I/O 유니트의 기종과 장착된 고기능 I/O 유니트의 기종이 다를 때 1(ON)이 됩니다. CPU 유니트의 운전은 계속됩니다. 전면 ERR/ALM LED가 점멸합니다. 1: 이상 0: 정상 이상 해제를 수행하면 0(OFF)이 됩니다. 어느 기기 번호에서 설정 이상이 발생했는지 A428~433CH에 반영됩니다. 유니트 간 동기 제어 설정으로 지정한 동기 유니트가 다음 조건에 해당하는 경우 1(ON)이 됩니다. · 해당 유니트가 동기 제어에 대응하지 않는 경우 · 해당 유니트가 실제로 연결되지 않은 경우 · 해당 유니트가 CPU 랙에 존재하지 않는 경우 (증설 랙에 연결됨) CPU 유니트의 운전은 계속됩니다. 전면 ERR/ALM LED가 점멸합니다. 1: 이상 0: 정상 이상을 해제하고 PLC 시스템 설정 내용을 유니트에 전송한 후 0(OFF)이 됩니다.	클리어	클리어	A428~433CH
A402CH	A402.03	CPU 고기능 유니트 설정 이상 플래그 (운전 계속 이상)	등록 I/O 테이블에 등록된 CPU 고기능 유니트의 기종과 장착된 CPU 고기능 유니트의 기종이 다를 때 1(ON)이 됩니다. CPU 유니트의 운전은 계속됩니다. 전면 ERR/ALM LED가 점멸합니다. 1: 이상 0: 정상 이상 해제를 수행하면 0(OFF)이 됩니다. 어느 기기 번호에서 설정 이상이 발생했는지 A427CH에 반영됩니다.	클리어	클리어	A427CH

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A402CH	A402.04	배터리 이상 플래그 (운전 계속 이상)	PLC 시스템 설정의 '배터리 이상 검출'에서 CPU 유니트의 배터리 이상을 '검출'로 설정한 경우, CPU 유니트의 배터리 이상(연결되지 않았거나 배터리 전압 저하)이 발생하면 1(ON)이 됩니다. CPU 유니트의 운전은 계속됩니다. 전면 ERR/ALM LED가 점멸합니다. 1: 이상 0: 정상 이상 해제를 수행하면 0(OFF)이 됩니다. 배터리 이상 플래그를 경보등 등을 이용하여 외부에 알릴 때 사용합니다.	클리어	클리어	PLC 시스템 설정의 '배터리 이상 검출'
	A402.06	고기능 I/O 유니트 이상 플래그 (운전 계속 이상)	CPU 유니트←→어떤 고기능 I/O 유니트 간의 데이터 교환에 이상이 있는 경우(고기능 I/O 유니트 자체의 이상 포함) 1(ON)로 설정됩니다. CPU 유니트의 운전은 계속됩니다. 전면 ERR/ALM LED가 점멸합니다. 이상이 발생한 유니트는 동작을 정지합니다. 1: 이상 0: 정상 이상 해제를 수행하면 0(OFF)이 됩니다. 어느 기기 번호와 CPU 유니트 간의 데이터 교환에 이상이 발생했는지 A418~423CH에 반영됩니다.	클리어	클리어	A418~A423CH
	A402.07	CPU 고기능 유니트 이상 플래그 (운전 계속 이상)	CPU 유니트←→CPU 고기능 유니트 간의 데이터 교환에 이상이 있는 경우(CPU 고기능 유니트 자체의 이상 포함) 1(ON)로 설정됩니다. CPU 유니트의 운전은 계속됩니다. 전면 ERR/ALM LED가 점멸합니다. 이상이 발생한 유니트는 동작을 정지합니다. 1: 이상 0: 정상 이상 해제를 수행하면 0(OFF)이 됩니다. 어느 유니트 번호와 CPU 유니트 간의 데이터 교환에 이상이 발생했는지 A417CH에 반영됩니다.	클리어	클리어	A417CH
	A402.10	PLC 시스템 설정 이상 플래그 (운전 계속 이상)	PLC 시스템 설정에 설정값 이상이 발생한 경우 1(ON)이 됩니다. CPU 유니트의 운전은 계속됩니다. 전면 ERR/ALM LED가 점멸합니다. 이상 발생 위치는 A406CH에 저장됩니다. 1: 이상 0: 정상 이상 해제를 수행하면 0(OFF)이 됩니다. 단, 유니트 간 동기 설정의 설정값 이상인 경우는 정상적인 설정값을 동기 유니트에 전송한 후 0(OFF)이 됩니다.	클리어	클리어	A406CH
	A402.12	기본 I/O 유니트 이상 플래그 (운전 계속 이상)	기본 I/O 유니트에 이상이 있을 때 1(ON)이 됩니다. CPU 유니트의 운전은 계속됩니다. 전면 ERR/ALM LED가 점멸합니다. 1: 이상 0: 정상 이상 해제를 수행하면 0(OFF)이 됩니다. 이상 발생 위치는 A408CH에 저장됩니다.	클리어	클리어	A408CH
	A402.13	다중 리프레시 이상 플래그 (운전 계속 이상)	다음과 같은 경우 1(ON)이 됩니다. PLC 시스템 설정의 '다중 갱신 이상 검출 유무 설정'을 '검출'로 설정한 경우, 동일한 고기능 I/O 유니트에 대해 다음과 같은 상태가 된 경우 • 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 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 다중 리프레시 이상이 됩니다.	클리어	클리어	A426CH PLC 시스템 설정의 '다중 리프레시 이상 검출 유무 설정'
	A402.15	FAL 이상 플래그 (운전 계속 이상)	FAL 명령(운전 계속 이상)이 실행되었을 때 1(ON)이 됩니다. CPU 유니트의 운전은 계속됩니다. 전면 ERR/ALM LED가 점멸합니다. 실행된 FAL 번호 001~511은 A360~391CH에 저장됩니다. 또한 실행된 FAL 번호 001~511(101Hex~2FFHex)의 선두에 4가 부가되고, 고장 코드 4101~42FF로 A400CH에 저장됩니다. 1: FAL 명령 실행 0: FAL 명령 비실행 이상 해제를 수행하면 0(OFF)이 됩니다.	클리어	클리어	A360~391CH A400CH

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A403CH	A403.00~ A403.08	메모리 이상 발생 위치	메모리 이상이 발생했을 때(A401.15가 ON) 메모리 이상 발생 위치에 따라 다음 비트가 1(ON)이 됩니다. 1: 이상 0: 정상 A403.00: 사용자 프로그램 A403.04: PLC 시스템 설정 A403.05: 등록 I/O 테이블 A403.07: 라우팅 테이블 A403.08: CPU 고기능 유니트 시스템 설정 메모리 이상이 발생하면 CPU 유니트의 운전은 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 이상 해제를 수행하면 0(OFF)이 됩니다.	클리어	클리어	A401.15
A404CH	A404.00~ A404.07	I/O 버스 이상 슬롯 번호	I/O 버스 이상의 슬롯 번호(00~09)가 BIN으로 저장됩니다. CJ2H-CPU6□-EIP, CJ2M-CPU3□의 내장 EtherNet/IP부에서 I/O 버스 이상이 발생한 경우 OBHex가 저장됩니다. 엔드 커버가 연결되지 않은 경우는 0EHex가 저장됩니다. 확인할 수 없는 경우는 0FHex가 저장됩니다. CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 00~09 Hex(슬롯 번호 0~9) A401.14(I/O 버스 이상 플래그)가 1(ON)이 됩니다. 이상 해제를 수행하면 0(OFF)이 됩니다.	클리어	클리어	A401.14
	A404.08~ A404.15	I/O 버스 이상 랙 번호	I/O 버스 이상인 랙 번호(00~03)가 BIN으로 저장됩니다. CJ2H-CPU6□-EIP, CJ2M-CPU3□의 내장 EtherNet/IP부에서 I/O 버스 이상이 발생한 경우 OBHex가 저장됩니다. 엔드 커버가 연결되지 않은 경우는 0EHex가 저장됩니다. 확인할 수 없는 경우는 0FHex가 저장됩니다. CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 00~03 Hex(랙 번호 0~3) A401.14(I/O 버스 이상 플래그)가 1(ON)이 됩니다. 이상 해제를 수행하면 0(OFF)이 됩니다.	클리어	클리어	A401.14
A405CH	A405.08	인터럽트 입력 유니트 장착 위치 부정 플래그	인터럽트 입력 유니트가 다음 범위 이외에 장착되었을 때 1(ON)이 됩니다. 이때, 'I/O 설정 이상(운전 정지 이상)'이 되고 운전은 정지됩니다. • CJ2H-CPU6□-EIP의 경우: CPU 장치의 슬롯 번호 0~3(CPU 유니트의 우측 4개 유니트분) • CJ2H-CPU6 □, CJ2M-CPU□□의 경우: CPU 장치의 슬롯 번호 0~4(CPU 유니트의 우측 5개 유니트분) 또한 물리적으로 위의 범위 안에 장착된 경우도 더미 유니트의 등록에 의해 I/O 테이블 에서 범위 밖으로 할당되면 이상이 발생합니다.	클리어	클리어	A401.10
	A405.15	주변 처리 계통의 사이클 시간 초과	주변 처리 계통의 사이클 시간이 2.0s를 초과했을 때 1(ON)이 됩니다. 이때, 사이클 시간이 초과되어 운전은 정지됩니다. 1: 주변 처리 계통의 사이클 시간 초과 0: 주변 처리 계통의 사이클 시간 초과 없음	클리어	클리어	A268CH
A407CH	A407.00~ A407.12	I/O 지점 수 초과 상세 정보 1	다음 중 하나의 값에 초과 또는 중복 이상이 발생했을 때 해당 값이 BIN 데이터로 저장됩니다. CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 발생 요인은 'I/O 지점 수 초과 상세 정보 1'(A407CH 비트 13~15)을 참조하십시오. 1. 등록 I/O 테이블에 설정된 I/O 총 지점 수(원격 I/O 자국 장치 제외)가 CPU 유니트의 최대 규정값을 초과했을 때의 I/O 지점 수 2. 인터럽트 입력 점수가 32점을 초과했을 때의 인터럽트 입력 점수 3. 증설 장치 장착 대수가 초과했을 때의 대수 이 중 어떤 이상이 발생하면 A407.00~A407.12에 그 값이 저장됩니다. 이상 해제 시에 클리어됩니다.	클리어	클리어	A401.11, A407.13~ A407.15
	A407.13~ A407.15	I/O 지점 수 초과 상세 정보 2	I/O 지점 수 초과 이상 발생 요인을 3비트로 저장합니다. A407CH의 비트 0~12에 저장된 데이터의 의미를 표시합니다. 000: I/O 총 지점 수 초과 001: 인터럽트 입력 점수 초과 101: 장착 가능 증설 장치 수 초과 111: 1랙(장치)의 유니트 수 초과 이상 해제 시에 클리어됩니다.	클리어	클리어	—

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A408CH	A408.00~A408.07	기본 I/O 유니트 이상 슬롯 번호	기본 I/O 유니트에 이상이 있을 때(A402.12가 ON일 때), 이상이 발생한 슬롯 번호가 BIN 데이터로 저장됩니다. CPU 유니트의 운전은 계속됩니다. 전면 ERR/ALM LED가 점멸합니다. 00~09 Hex(슬롯 번호 0~9) 이상 해제 시에 클리어됩니다.	클리어	클리어	A402.12
	A408.08~A408.15	기본 I/O 유니트 이상 랙 번호	기본 I/O 유니트에 이상이 있을 때(A402.12가 ON일 때), 이상이 발생한 유니트가 장착된 랙 번호가 BIN 데이터로 저장됩니다. 00~03 Hex(랙 번호 0~3) 이상 해제 시에 클리어됩니다.	클리어	클리어	A402.12
A409CH	A409.00~A409.03	증설 장치 중복 랙 번호 플래그	CX-Programmer에 의한 증설 장치의 시작 CH 설정에서 랙 간의 CH 할당이 중복된 경우 또는 각 랙의 시작 CH 설정이 0901CH 이상인 경우, 해당하는 랙 번호가 1(ON)이 됩니다. 비트 00~03: 랙 번호 0~3에 대응 1: 랙 간 CH 할당 중복 또는 랙 시작 CH 번호 초과 0: 정상 이상 해제 시에 클리어됩니다.	클리어	클리어	—
A410CH	A410.00~A410.15	CPU 고기능 유니트 중복 유니트 번호 플래그	CPU 고기능 유니트의 유니트 번호가 중복 설정된 경우(A401.13이 ON일 때) 중복된 대응 유니트 번호가 1(ON)로 설정됩니다. 각 비트가 유니트 번호에 대응합니다. 비트 00~15: 0~F(유니트 번호) CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 1: 중복됨 0: 중복 안됨	클리어	클리어	A401.13
A411~A416CH	A411.00~A416.15	고기능 I/O 유니트 중복 기기 번호 플래그	고기능 I/O 유니트의 기기 번호가 중복 설정된 경우(A401.13이 ON일 때) 중복된 대응 기기 번호가 1(ON)로 설정됩니다. 각 비트가 기기 번호에 대응합니다. A411CH의 비트 00~A416CH의 비트 15: 0~95(기기 번호) CPU 유니트의 운전이 정지됩니다. 이때, CPU 유니트 전면의 ERR/ALM LED는 점등됩니다. 증설 랙 시작 CH 설정에 따라 기본 I/O 유니트의 채널이 중복된 경우 1(ON)이 저장됩니다. 1: 중복됨 0: 중복 안됨	클리어	클리어	A401.13
A417CH	A417.00~A417.15	CPU 고기능 유니트 이상 유니트 번호 플래그	CPU 유니트←→CPU 고기능 유니트 간의 데이터 교환에 이상이 있는 경우(A402.07이 ON일 때), 이상이 발생한 유니트 번호에 해당하는 비트가 1(ON)이 됩니다. 각 비트가 유니트 번호에 대응합니다. 비트 00~15: 0~F(유니트 번호) CPU 유니트의 운전은 계속됩니다. 전면 ERR/ALM LED가 점멸합니다. 1: 이상 있음 0: 이상 없음	클리어	클리어	A402.07
A418~A423CH	A418.00~A423.15	고기능 I/O 유니트 이상 기기 번호 플래그	CPU 유니트←→고기능 I/O 유니트 간의 데이터 교환에 이상이 있는 경우(A402.06이 ON일 때) 1(ON)로 설정됩니다. 각 비트가 기기 번호에 대응합니다. A418CH의 비트 00~A423CH의 비트 15: 0~95(기기 번호). CPU 유니트의 운전은 계속됩니다. 전면 ERR/ALM LED가 점멸합니다. 1: 이상 있음 0: 이상 없음 이상 해제 시에 클리어됩니다. 이상이 발생한 기기 번호를 확인할 수 없을 때는 어느 비트도 ON이 되지 않습니다.	클리어	클리어	A402.06
A426CH	A426.00~A426.11	다중 리프레시 이상 발생 기기 번호	A402.13이 ON일 때 다중 리프레시 대상 고기능 I/O 유니트의 기기 번호가 BIN 12비트로 저장됩니다. 다중 리프레시 대상 고기능 I/O 유니트의 기기 번호: 000~05FHex(0~95호기) 이상 해제 시에 클리어됩니다.	클리어	클리어	A402.13, A426.15
	A426.15	다중 리프레시 이상 요인 플래그	A402.13(다중 리프레시 이상 플래그)이 ON일 때의 이상 요인을 저장합니다. CPU 유니트의 운전은 계속됩니다. 전면 ERR/ALM LED가 점멸합니다. 인터럽트 태스크 중에 중복해서 고기능 I/O 유니트에 대해 리프레시를 실행한 경우 1(ON)이 됩니다. 1: 다중 리프레시	클리어	클리어	A402.13, A426.00~A426.11

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A427CH	A427.00~ A427.15	CPU 고기능 유니트 설정 이상 유니트 번호 플래그	CPU 고기능 유니트 설정 이상 발생 시(A402.03이 ON일 때), 설정 이상이 발생한 유니트 번호에 해당하는 비트가 1(ON)이 됩니다. CPU 유니트의 운전은 계속됩니다. 전면 ERR/ALM LED가 점멸합니다. 각 비트가 유니트 번호에 대응합니다. 비트 00~15: 0~F(유니트 번호) 1: 설정 이상 있음 0: 설정 이상 없음	클리어	클리어	전원 ON 시 또는 I/O 인식 시 A402.03
A428~ A433CH	A428.00~ A433.15	고기능 I/O 유니트 설정 이상 기기 번호 플래그	고기능 I/O 유니트 설정 이상 발생 시(A402.02가 ON일 때), 설정 이상이 발생한 기기 번호에 해당하는 비트가 1(ON)이 됩니다. CPU 유니트의 운전은 계속됩니다. 전면 ERR/ALM LED가 점멸합니다. 각 비트가 기기 번호에 대응합니다. 비트 00~15: 0~F(기기 번호) 1: 설정 이상 있음 0: 설정 이상 없음	클리어	클리어	전원 ON 시 또는 I/O 인식 시 A402.02
A440CH	—	인터럽트 태스크 최대값 처리 시간	인터럽트 태스크 최대 처리 시간이 0.1ms 단위의 BIN 데이터로 저장됩니다. 운전을 시작할 때 클리어됩니다. PLC 시스템 설정에 따라 '고속 인터럽트'가 유효화되어 있는 경우 무효가 됩니다.	클리어	클리어	최대 처리 시간의 인터럽트 태스크를 실행한 후
A441CH)	—	최대 처리 시간 인터럽트 태스크 번호	최대 처리 시간의 인터럽트 태스크 번호가 BIN 데이터로 저장됩니다(8000Hex~80FFHex). 인터럽트가 발생한 경우 비트 15가 1(ON)이 됩니다. Hex 하위 2자리가 태스크 번호 00~FFHex에 대응합니다. 운전을 시작할 때 클리어됩니다. PLC 시스템 설정에 따라 '고속 인터럽트'가 유효화되어 있는 경우 무효가 됩니다.	클리어	클리어	최대 처리 시간의 인터럽트 태스크를 실행한 후
A446CH	A446.00~ 446.07	보호 해제 조작 횟수	• 보호가 실행된 상태에서 보호 해제에 실패했을 때(보호 해제 비밀번호 입력 시) 카운트 업됩니다. • UM 금지와 태스크 금지의 해제 조작 횟수의 합계가 표시됩니다. • 255(10진수)까지 카운트 업되면 255에서 정지합니다. • 보호가 모두 해제되면 00Hex가 됩니다.	—	—	—
A450CH	—	CIO 영역 지정	평션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A451CH	—	WR 영역 지정	평션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A452CH	—	HR 영역 지정	평션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A459CH	—	IR 영역 지정	평션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A460CH	—	DM 영역 지정	평션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A461CH	—	EM0 영역 지정	평션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A462CH	—	EM1 영역 지정	평션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A463CH	—	EM2 영역 지정	평션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A464CH	—	EM3 영역 지정	평션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A465CH	—	EM4 영역 지정	평션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A466CH	—	EM5 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A467CH	—	EM6 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A468CH	—	EM7 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A469CH	—	EM8 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A470CH	—	EM9 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A471CH	—	EMA 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A472CH	—	EMB 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A473CH	—	EMC 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A474CH	—	EMD 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A475CH	—	EME 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A476CH	—	EMF 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A477CH	—	EM10 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A478CH	—	EM11 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A479CH	—	EM12 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A480CH	—	EM13 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A481CH	—	EM14 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A482CH	—	EM15 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A483CH	—	EM16 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A484CH	—	EM17 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—

주소		명칭	설명	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트					
A485CH	—	EM18 영역 지정	펑션 블록의 파라미터로 입출력 변수(VER_IN_OUT)를 사용하지 않고 주소를 지정하는 경우, 영역을 지정하기 위한 코드(고정값)를 저장합니다. Smart FB 라이브러리만 사용합니다. 이 값을 변경하지 마십시오.	고정값 설정	고정값 설정	—
A10001~ A10003CH	—	제조 로트 번호	CPU 유니트의 제조 로트 번호가 BCD로 저장됩니다. 이 영역을 참조하면 하드웨어를 확인할 수 있습니다. A10000.00~07: 00Hex A10000.08~15: 00Hex A10001.00~07: 00Hex A10001.08~15: 일(01~31) A10002.00~07: 월(01~12) A10002.08~15: 년(00~09) A10003.00~07: 00Hex A10003.08~15: 00Hex	—	—	—
A10100CH	A10100.00	동기 제어 실행 중 플래그	유니트 간 동기 제어 실행 중에 1(ON)이 됩니다. 1: 유니트 간 동기 제어 실행 중 0: 유니트 간 동기 제어 실행 중이 아님	클리어	클리어	전원 ON 또는 유니트 재시작 후 2회째 동기 신호가 발신되었을 때
A10101CH	—	동기 데이터 링크 송신 에러(유니트)	규정 시간 내에 동기 유니트에서 데이터를 송신하지 않아서 동기 데이터 링크 릴레이의 입력에 실패한 경우 1(ON)이 됩니다. 0001Hex: 에러 발생 0000Hex: 정상	클리어	클리어	동기 리프레시 처리 실행 시
A10102CH	—	동기 제어 주기 설정 시간	PLC 시스템 설정에서 설정한 동기 제어 주기	유지	유지	전원 ON 또는 유니트 재시작 시

(주) 1 CJ 시리즈에서 다음의 플래그는 특수 보조 릴레이가 아닌 '조건 플래그' 및 '클록 펄스'로 별도의 읽기 전용 플래그 영역에 할당되어 있습니다. 어느 경우든 라벨로 지정합니다. 자세한 내용은 '6-20 조건 플래그' 및 '6-21 클록 펄스'를 참조하십시오.

에러 플래그	조건 플래그
액세스 에러 플래그	
캐리 플래그	
> 플래그	
= 플래그	
< 플래그	
부정 플래그	
오버 플로 플래그	
언더 플로 플래그	
플래그	
≠ 플래그	
플래그	
상시 ON 플래그	
상시 OFF 플래그	
0.02초 클록 펄스	클록 펄스
0.1초 클록 펄스	
0.2초 클록 펄스	
1초 클록 펄스	
1분 클록 펄스	

부록-3-2 읽기/쓰기 가능 영역

주소		명칭	설명	설정 내용	운전모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트						
A500CH	A500.12	IOM(I/O 메모리) 유지 비트	CPU 유니트의 동작 모드 전환(프로그램←→운전/모니터) 또는 전원 ON 시, I/O 메모리 영역의 유지 여부를 지정합니다. 전원을 켰을 때 이 플래그를 유지하며, 전원이 켜졌을 때 I/O 메모리 영역을 유지하려면 PLC 시스템 설정의 IOM(I/O 메모리) 유지 비트를 유지로 설정해야 합니다.)	1: 유지 0: 유지 안 함	유지	비유지 (PLC 시스템 설정으로 유지 가능)	전원 ON 시 참조 PLC 시스템 설정 'IOM(I/O 메모리) 유지 비트 유지'
	A500.13	강제 세트/리셋 유지 플래그	CPU 유니트의 동작 모드 전환(프로그램←→모니터) 시 또는 전원 ON 시, 강제 세트/리셋 상태의 유지 여부를 지정합니다. 반드시 IOM(I/O 메모리) 유지 비트(A500.12)와 한 쌍으로 사용하고, 이 플래그의 ON과 동시에 A500.12를 ON으로 설정하십시오. 전원을 켰을 때 이 플래그를 유지하며, 전원이 켜졌을 때 강제 세트/리셋 상태를 유지하려면 PLC 시스템 설정의 강제 세트/리셋 유지 플래그를 유지로 설정해야 합니다.)	1: 유지 0: 유지 안 함	유지	비유지 (PLC 시스템 설정으로 유지 가능)	전원 ON 시 참조 PLC 시스템 설정 '전원 ON 시 '강제 세트/리셋 유지 플래그' 유지'
	A500.14	이상 내역 저장 포인터 리셋 플래그	기동(0→1) 시에 이상 내역 저장 포인터(A300CH)를 리셋(00Hex)합니다. 이상 내역 저장 영역(A100~199CH)은 클리어하지 않습니다. 클리어 실행 후 시스템에 의해 자동으로 0(OFF)으로 되돌아갑니다.	0→1: 클리어	유지	클리어	A100~199CH, A300CH
	A500.15	부하 차단 플래그	1(ON)로 하면 I/O 메모리의 출력 릴레이 상태와 관계 없이(운전 정지 이상 시와 동일하게) 모든 기본 I/O 유니트 및 출력 유니트를 0(OFF)으로 합니다. 이때, CPU 유니트 전면의 INH LED가 점등합니다. (주) 전원 OFF 시에도 유지합니다. 고기능 I/O 유니트의 할당 릴레이 영역 출력(외부로 아날로그 출력)은 각 고기능 유니트의 기능에 의해 좌우됩니다.	—	유지	유지	—
A501CH	A501.00~A501.15	CPU 고기능 유니트 재시작 플래그	기동(0→1) 시 CPU 고기능 유니트가 재시작됩니다. (초기 처리에 들어갑니다.) 각 비트가 유니트 번호에 대응합니다. 비트 00~15: 0~F(유니트 번호) 재시작 후 CPU 고기능 유니트 초기화 중 플래그(A302.00~302.15)가 1(ON)이 되고, 초기화 처리 후 자동으로 시스템에 의해 0(OFF)이 됩니다.	0→1: 재시작 1→0: 재시작 종료 후 시스템에 의해 자동으로 0	유지	클리어	A302.00~A302.15
A502~A507CH	A502.00~A507.15	고기능 I/O 유니트 재시작 플래그	기동(0→1) 시 고기능 I/O 유니트가 재시작됩니다(초기 처리에 들어갑니다). 각 비트가 기기 번호에 대응합니다. A502CH의 비트 00~A507CH의 비트 15: 0~95(기기 번호) 재시작 후 고기능 I/O 유니트 초기화 중 플래그(A330.00~335.15)가 1(ON)이 되고, 초기화 처리 후 자동으로 시스템에 의해 0(OFF)이 됩니다.	0→1: 재시작 1→0: 재시작 종료 후 시스템에 의해 자동으로 0	유지	클리어	A330.00~A335.15
A508CH	A508.09	미분 모니터 실행 완료 플래그	미분 모니터 실행 시, 미분 모니터의 조건이 성립했을 때 1(ON)이 됩니다. 미분 모니터를 시작하면 클리어됩니다.	1: 미분 모니터 조건 성립 0: 조건 성립 안 함	유지	클리어	–
	A508.11	트레이스 트리거 모니터 플래그	트레이스 트리거 플래그(A508.14)에 의해 트레이스 트리거 조건이 성립했을 때 1(ON)이 되고, 데이터 트레이스 시작 플래그에서 다음 트레이스의 샘플링이 시작되면 0(OFF)이 됩니다.	1: 트레이스 트리거 조건 성립 0: 트레이스 실행 중에 트리거 조건이 성립하지 않거나 트레이스가 실행되지 않음	유지	클리어	–
	A508.12	트레이스 완료 시 ON 플래그	트레이스 실행 시 트레이스 메모리 용량 분량의 샘플링이 완료되었을 때 1(ON)이 됩니다. 다음으로 트레이스에서 데이터 트레이스 시작 플래그가 OFF→ON이 되면 0(OFF)이 됩니다.	1: 트레이스 완료 0: 트레이스를 실행하지 않았거나 트레이스 실행 중에 완료되지 않음	유지	클리어	–

주소		명칭	설명	설정 내용	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트						
A508CH	A508.13	트레이스 실행 중 ON 플래그	데이터 트레이스 시작 플래그가 0(OFF)→1(ON)이 되면 1(ON)이 되고, 트레이스가 완료되면 0(OFF)이 됩니다.	1: 데이터 트레이스 실행(샘플링) 중 0: 데이터 트레이스 비실행(비샘플링) 중	유지	클리어	—
	A508.14	트레이스 트리거 플래그	0(OFF)→1(ON)에서 트레이스의 트리거 조건이 성립합니다. 딜레이값(횟수)만큼 뒤 또는 앞의 데이터가 유효가 됩니다.	1: 트레이스 트리거 조건 성립 0: 트레이스 트리거 조건 성립 안 함	유지	클리어	—
	A508.15	데이터 트레이스 시작 플래그	CX-Programmer에서 데이터 트레이스를 시작할 때 0(OFF)→1(ON)로 하면 그 이후 다음 3가지 중 하나의 타이밍에서 실제로 트레이스 메모리에 데이터가 저장되기 시작합니다. 1. 정해진 주기에 의한 트레이스(일정 주기 10~2550ms) 2. TRSM 명령에 의한 트레이스(명령 실행 시에 트레이스) 3. 1사이클마다 트레이스(1사이클의 마지막에 트레이스) 이 플래그의 조작은 CX-Programmer에서만 실행합니다.	0→1: 데이터 트레이스 (샘플링) 시작 (주변 툴에 의함)	유지	클리어	—
A510~ A511CH	—	전원 ON 시각	전원이 켜진 시각이 저장됩니다. 전원이 켜질 때마다 갱신되며 BCD 데이터로 저장됩니다. A510.00~A510.07: 초(00~59) A510.08~A510.15: 분(00~59) A511.00~A511.07: 시(00~23) A511.08~A511.15: 일(01~31)	왼쪽 설명 참조	유지	설명란에 기재	전원 ON 시
A512~ A513CH	—	전원 OFF 시각	전원 OFF 시의 시각이 저장됩니다. 전원을 끌 때마다 갱신되며 BCD 데이터로 저장됩니다. A512.00~A512.07: 초(00~59) A512.08~A512.15: 분(00~59) A513.00~A513.07: 시(00~23) A513.08~A513.15: 일(01~31) 전원을 켰을 때 클리어되지 않습니다.	왼쪽 설명 참조	유지	유지	전원 OFF 시
A514CH	—	전원 OFF 발생 횟수	PLC 본체에 최초로 전원을 투입한 이후 전원 OFF가 발생할 때마다 누적(+1)됩니다. BIN으로 저장되며, 리셋하는 경우는 0000Hex를 저장하십시오. 전원을 켰을 때 클리어되지 않습니다. 메모리 손상 통지 플래그(A395.11)가 ON일 때 클리어됩니다.	0000~FFFFHex	유지	유지	전원 ON 시 A395.11
A515~ A517CH	—	운전 시작 시각	동작 모드를 '운전' 또는 '모니터' 모드로 변경하면 운전을 시작한 시각이 BCD 데이터로 저장됩니다. A515.00~A515.07: 초(00~59) A515.08~A515.15: 분(00~59) A516.00~A516.07: 시(00~23) A516.08~A516.15: 일(01~31) A517.00~A517.07: 월(01~12) A517.08~A517.15: 년(00~99) 전원을 켰을 때부터 운전 시작 전까지는 지난번의 운전 시작 시각이 저장되어 있습니다.	왼쪽 설명 참조	유지	유지	왼쪽 설명 참조
A518~ A520CH	—	운전 정지 시각	운전 정지 이상이 발생하거나 동작 모드를 '프로그램' 모드로 변경한 경우, 운전을 정지한 시각이 BCD 데이터로 저장됩니다. A518.00~A518.07: 초(00~59) A518.08~A518.15: 분(00~59) A519.00~A519.07: 시(00~23) A519.08~A519.15: 일(01~31) A520.00~A520.07: 월(01~12) A520.08~A520.15: 년(00~99) 이상 발생 시에는 이상 발생 시각이 저장됩니다. 그 후 '프로그램' 모드로 변경하면 그 시각으로 덮어 쓰게 됩니다.	왼쪽 설명 참조	유지	유지	왼쪽 설명 참조

주소		명칭	설명	설정 내용	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트						
A523CH	—	통전 시간	PLC 본체의 통전 시간을 BIN 16비트, 10시간 단위로 저장합니다. 리셋하는 경우는 0000Hex를 저장하십시오. FFFFHex가 되면 더 이상 갱신되지 않습니다. 통전 10시간 단위로 갱신됩니다. 전원을 켰을 때 클리어되지 않습니다. 메모리 손상 통지 플래그(A395.11)가 ON일 때 클리어(0000Hex)됩니다.	0000~FFFFHex	유지	유지	—
A526CH	A526.00	시리얼 포트 재시작 플래그	시리얼 포트에 대해 포트를 재시작하는 경우 0→1로 기동하십시오. (툴 버스 모드로 통신할 때는 제외) 재시작 처리 종료 후 시스템에 의해 자동으로 0(OFF)이 됩니다.	0→1: 재시작	유지	클리어	—
A527CH	A527.00~ A527.07	온라인 편집 금지 중 플래그 액세스용 비밀번호	온라인 편집 금지 중 플래그(A527.09)가 유효인지, 무효인지 지정합니다. CX-Programmer에서의 온라인 편집('모니터' 모드에서 운전 중인 프로그램의 일부 변경 및 추가) 조작을 금지하는 경우, 이 8비트를 5AHex로 하고, A527.09 플래그를 1(ON)로 설정합니다.	5A 이외: 온라인 편집 금지 중 플래그 무효 5A: 온라인 편집 금지 중 플래그 유효	유지	클리어	A527.09
	A527.09	온라인 편집 금지 중 플래그	온라인 편집을 금지하는 경우 이 플래그를 1(ON)로 설정하십시오. 이 플래그의 유효/무효는 A527.00~A527.07(온라인 편집 금지 중 플래그 액세스용 비밀번호)의 설정에 좌우됩니다.	1: 금지 0: 금지 안 함	유지	클리어	A527.00~ A527.07
A528CH	A528.00~ A528.07	시리얼 포트 에러 플래그	시리얼 포트에서 에러가 발생하면 해당 에러 코드가 저장됩니다. • 시리얼 포트를 재시작 처리할 때 시스템에 의해 클리어됩니다. • 툴 버스 모드일 때는 무효가 됩니다. • NT 링크 모드일 때는 '비트 5: 시간 초과 에러'만 유효가 됩니다. • 시리얼 PLC 링크 모드일 때 친국의 경우: 비트 05: 시간 초과 에러 자국의 경우: 비트 05: 시간 초과 에러 비트 04: 오버런 에러 비트 03: 프레이밍 에러 만 유효가 됩니다. CX-Programmer에서 클리어할 수 있습니다.	비트 0: 사용 안 함 비트 1: 사용 안 함 비트 2: 패리티 에러 비트 3: 프레이밍 에러 비트 4: 오버런 에러 비트 5: 시간 초과 에러 비트 6: 사용 안 함 비트 7: 사용 안 함	유지	클리어	—
A529CH	—	시스템 이상 발생 FAL/FALS 번호	FAL 명령 또는 FALS 명령 실행을 사용하여 시스템 이상을 고의로 발생시키는 경우, 더미로 사용할 FAL/FALS 번호를 설정합니다. FAL 명령 또는 FALS 명령 실행 시에 명령이 설정값을 참조하고, 명령의 피연산자 N의 값이 여기에서 설정한 값인 경우는 일반적인 사용자 정의 운전 계속/정지 이상이 아니라 피연산자 S(고장 코드), S+1(이상 내용)로 정의된 시스템 이상을 고의로 발생시킵니다.	0001~01FFHex: 1~511 중 하나 0000, 0200~FFFFHex: 시스템 이상 발생의 FAL/FALS 번호 설정 없음(시스템 이상을 고의로 발생시키지 않음)	유지	클리어	—
A530CH	—	전원 OFF 금지 구간 설정	DI 명령—EI 명령 간 또는 DI 명령—END 명령 간의 전원 OFF 처리를 마스크할 수 있습니다(단, 전원 OFF 인터럽트 태스크가 무효인 경우만 가능).	A5A5Hex: 전원 OFF 처리 마스크 유효 A5A5Hex 이외: 전원 OFF 처리 마스크 무효	클리어	클리어	—

주소		명칭	설명	설정 내용	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트						
A580CH	A580.00~ A580.03	FB 내 통신 명령 재송신 횟수	PLC 시스템 설정에서 설정한 FB 내 통신 명령 설정의 재송신 횟수가 자동 저장됩니다. 사용자가 사용하는 특수 보조 릴레이는 아닙니다. 오므론이 제공하는 'Smart FB 라이브러리' 중에서 특히, FINS 메시지 통신 또는 DeviceNet Explicit 메시지 통신을 실행하는 Smart FB를 사용하는 경우, 사용자는 PLC 시스템 설정의 'FB 내 통신 명령 설정'에 재시도 횟수, 응답 감시 시간을 설정해야 합니다. 거기에서 설정한 값이 특수 보조 릴레이 A580~582CH에 자동으로 저장됩니다. 'Smart FB 라이브러리'는 그 값을 참조하여 동작합니다.	0~FHex	PLC 시스템 설정값 설정	클리어	운전 시작 시
A581CH	—	FB 내 통신 명령 응답 감시 시간	PLC 시스템 설정에서 설정한 FB 내 통신 명령 응답 감시 시간이 자동 저장됩니다. 사용자가 사용하는 특수 보조 릴레이는 아닙니다. 오므론이 제공하는 'Smart FB 라이브러리' 중에서 특히, FINS 메시지 통신 또는 DeviceNet Explicit 메시지 통신을 실행하는 Smart FB를 사용하는 경우, 사용자는 PLC 시스템 설정의 'FB 내 통신 명령 설정'에 재시도 횟수, 응답 감시 시간을 설정해야 합니다. 거기에서 설정한 값이 특수 보조 릴레이 A580~582CH에 자동으로 저장됩니다. 'Smart FB 라이브러리'는 그 값을 참조하여 동작합니다.	0001~FFFFHex (단위 0.1초, 0.1~6553.5) 0000Hex: 2초	PLC 시스템 설정값 설정	클리어	운전 시작 시
A582CH	—	FB 내 DeviceNet 통신 명령 응답 감시 시간	PLC 시스템 설정에서 설정한 FB 내 DeviceNet 통신 명령 응답 감시 시간이 자동 저장됩니다. FAL 명령 또는 FALS 명령 실행 시에 명령이 설정값을 참조하고, 명령의 피연산자 N의 값이 여기에서 설정한 값인 경우는 일반적인 사용자 정의 운전 계속/정지 이상이 아니라 피연산자 S(고장 코드), S+1(이상 내용)로 정의된 시스템 이상을 고의로 발생시킵니다.	0001~FFFFHex (단위 0.1초, 0.1~6553.5) 0000Hex: 2초	PLC 시스템 설정값 설정	클리어	운전 시작 시
A595~ A596CH	—	백그라운드 처리 시 IR00 출력	인덱스 레지스터 출력을 지정한 명령을 백그라운드 처리로 실행한 경우, 결과를 이 영역에 출력하며 IR00에는 출력하지 않습니다.	00000000~FFFFFFFFHex 하위 4 자리: A595CH, 상위 4자리: A596CH	클리어	클리어	—
A597CH	—	백그라운드 처리 시 DR00 출력	데이터 레지스터 출력을 지정한 명령을 백그라운드 처리로 실행한 경우, 결과를 이 영역에 출력하며 DR00에는 출력하지 않습니다.	0000~FFFFHex	클리어	클리어	—
A598CH	A598.00	FPD 명령용 티칭 플래그	FPD 명령 실행 시에 이상 감시 시간을 자동 설정하여 동작(티칭 동작)시킬 때 1(ON)로 설정합니다. 이 플래그가 ON인 동안에 검출 회로 조건이 ON이 된 후, 이상 감시 릴레이가 ON이 될 때까지의 시간을 계측합니다. 계측한 시간이 설정값을 초과하면 1.5배의 값이 이상 감시 시간 설정값의 피연산자에 저장됩니다. 이상 감시 시간 설정값을 CH로 지정할 때만 가능합니다.	1: 이상 감시 시간 티칭 0: 이상 감시 시간 사용자 설정	클리어	클리어	—
	A598.01	백그라운드 처리 시 = 플래그 출력	SRCH 명령을 백그라운드 처리로 실행한 경우 데이터를 검색한 결과, 일치하는 데이터가 있을 때 이 플래그를 1(ON)로 설정합니다.	1: 검색 결과, S2의 데이터와 동일한 데이터가 테이블 내에 있을 때 0: 그 이외	클리어	클리어	—
A600~ A603CH	—	MCRO 명령용 인수 영역	MCRO 명령 실행 시에 인수 데이터 하위 CH 번호에서 4CH의 인수 데이터가 여기에 복사됩니다. 복사된 인수 4CH는 서브루틴으로 전달됩니다.	인수 데이터: 4CH	클리어	클리어	—
A604~ A607CH	—	MCRO 명령용 반환 값 영역	MCRO 명령의 서브루틴 실행 후 서브루틴에서 반환한 결과 4CH가 전달됩니다. 전달된 반환 결과는 MCRO 명령의 파라미터로 반환 데이터의 하위 CH 번호부터 4CH에 저장됩니다.	반환 데이터: 4CH	클리어	클리어	—

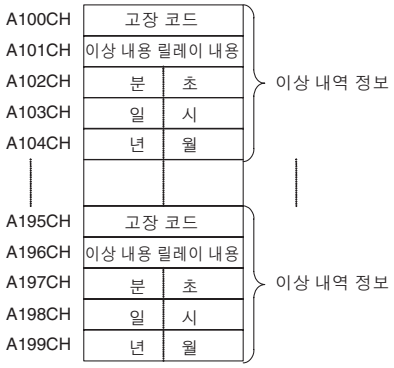
주소		명칭	설명	설정 내용	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트						
A619CH	A619.02	시리얼 포트 설정 변경 중 플래그	시리얼 포트의 통신 조건 설정 변경 중에 1(ON)이 됩니다. 시리얼 포트 설정 변경 명령(STUP 명령)을 실행할 때 1(ON)이 됩니다. 설정 변경이 완료되면 0(OFF)으로 되돌아갑니다.	1: 변경 중 0: 변경 중 아님	유지	클리어	—
A620CH	A620.01	시리얼커뮤니케이션 유니트 0호기 포트 1용 시리얼 통신 포트 설정 변경 중 플래그	해당 포트의 설정 변경 중에 1(ON)이 됩니다. STUP(시리얼 포트 설정 변경 명령)을 실행할 때 1(ON)이 됩니다. 설정 변경이 완료되면 자동으로 0(OFF)으로 되돌아갑니다. STUP 명령이 아닌 수동 조작으로 이 비트를 1(ON)로 설정하면 시리얼 포트 설정 변경을 통지하는 것도 가능합니다.	1: 변경 중 0: 변경 중 아님	유지	클리어	—
	A620.02	시리얼커뮤니케이션 유니트 0호기 포트 2의 시리얼 통신 포트 설정 변경 중 플래그					
	A620.03	시리얼커뮤니케이션 유니트 0호기 포트 3의 시리얼 통신 포트 설정 변경 중 플래그					
	A620.04	시리얼커뮤니케이션 유니트 0호기 포트 4의 시리얼 통신 포트 설정 변경 중 플래그					
A621~ A635CH	A621.00~ A635.04	시리얼커뮤니케이션 유니트 1~15호기 포트 1~4의 시리얼 통신 포트 설정 변경 중 플래그					
A650CH	A650.00~ A650.07	프로그램 교체 종료 플래그	• 정상 종료 시(A650.14가 0(OFF)일 때) O1Hex: 프로그램 파일(.obj)을 교체했습니다. • 이상 종료 시(A650.14가 1(ON)일 때) 00Hex: 정지 이상이 발생했습니다. 01Hex: 메모리 이상이 발생했습니다. 11Hex: 쓰기 방지되어 있습니다. 12Hex: 프로그램 교체 비밀번호 이상입니다. 21Hex: 메모리 카드가 존재하지 않습니다. 22Hex: 지정된 파일이 존재하지 않습니다. 23Hex: 지정된 파일의 크기가 기종의 용량을 초과했습니다(메모리 이상). 31Hex: 다음 중 하나의 다른 처리를 실행 중입니다. • 파일 메모리 관련 처리 실행 중 • 사용자 프로그램 쓰기 중 • 동작 모드 변경 중	—	유지	클리어	—
	A650.14	프로그램 교체 이상 종료 플래그	프로그램 교체 시작 플래그(A650.15)를 0(OFF)→1(ON)로 해서 프로그램을 교체하려고 했지만 이상 종료인 경우, 1(ON)이 됩니다. 다시 프로그램 교체 시작 플래그가 0(OFF)→1(ON)이 되면 0(OFF)이 됩니다.	1: 교체 이상 종료 0: 교체 정상 종료 또는 A650.15를 0→1	유지	클리어	—
	A650.15	프로그램 교체 시작 플래그	프로그램 교체 비밀번호(A651CH)가 유효한 상태(A5A5Hex)에서 이 플래그를 0(OFF)→1(ON)로 하면 프로그램의 교체를 시작합니다. 프로그램 교체 중에는 0(OFF)→1(ON)로 하지 마십시오. 전원 ON 시 또는 프로그램 교체 종료 시에는 정상 종료/이상 종료에 관계 없이 0(OFF)이 됩니다. CX-Programmer, PT, 상위 PC 등에서 이 플래그를 읽으면 프로그램을 교체 중인지 여부를 알 수 있습니다.	0→1: 프로그램 교체 0: 교체 정상/이상 종료 또는 전원 ON	유지	클리어	—

주소		명칭	설명	설정 내용	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트						
A651CH	—	프로그램 교체 비밀번호	프로그램을 교체하는 데 필요한 비밀번호를 입력합니다. A5A5Hex: 프로그램 교체 시작 플래그(A650.15) 유효 위 이외: 프로그램 교체 시작 플래그(A650.15) 무효 전원 ON 시 또는 프로그램 교체 종료 시에는 정상 종료/이상 종료에 관계 없이 0(OFF)이 됩니다.	—	유지	클리어	—
A654~ A657CH	—	프로그램 교체 원본 파일명	프로그램 교체를 시작하면 교체하는 파일명을 ASCII 코드로 저장해 둡니다. 파일명은 확장자를 제외하고 영숫자로 8문자 이내에서 지정합니다. A654CH에서 A657CH 순서(하위 CH→상위 CH)로 상위 바이트→하위 바이트 순서로 저장합니다. 8문자에 도달하지 않는 경우는 나머지 하위 바이트 및 상위 CH를 20Hex(공백)로 채웁니다. 파일명 중간에 NUL이나 공백을 삽입할 수 없습니다. 예) 파일명이 ABC.OBJ인 경우 15 0 A654 41 42 A655 43 20 A656 20 20 A657 20 20	—	유지	클리어	—
A720~ A722CH	—	전원 ON 시각 이력 데이터 #1	A510~511CH의 전원 ON 시각과 같은 시각의 데이터입니다(연, 월 정보를 추가한 데이터). A720.00~A720.07: 초(00~59) A720.08~A720.15: 분(00~59) A721.00~A721.07: 시(00~23) A721.08~A721.15: 일(01~31) A722.00~A722.07: 월(01~12) A722.08~A722.15: 년(00~99)	왼쪽 설명 참조	유지	유지	전원 ON 시
A723~ A725CH	—	전원 ON 시각 이력 데이터 #2	A510~511CH의 전원 ON 시각의 1회 이전의 전원 ON 시각 데이터입니다. A723.00~A723.07: 초(00~59) A723.08~A723.15: 분(00~59) A724.00~A724.07: 시(00~23) A724.08~A724.15: 일(01~31) A725.00~A725.07: 월(01~12) A725.08~A725.15: 년(00~99)	왼쪽 설명 참조	유지	유지	전원 ON 시
A726~ A728CH	—	전원 ON 시각 이력 데이터 #3	A510~511CH의 전원 ON 시각의 2회 이전의 전원 ON 시각 데이터입니다. A726.00~A726.07: 초(00~59) A726.08~A726.15: 분(00~59) A727.00~A727.07: 시(00~23) A727.08~A727.15: 일(01~31) A728.00~A727.07: 월(01~12) A728.08~A728.15: 년(00~99)	왼쪽 설명 참조	유지	유지	전원 ON 시
A729~ A731CH	—	전원 ON 시각 이력 데이터 #4	A510~511CH의 전원 ON 시각의 3회 이전의 전원 ON 시각 데이터입니다. A729.00~A729.07: 초(00~59) A729.08~A729.15: 분(00~59) A730.00~A730.07: 시(00~23) A730.08~A730.15: 일(01~31) A731.00~A731.07: 월(01~12) A731.08~A731.15: 년(00~99)	왼쪽 설명 참조	유지	유지	전원 ON 시
A732~ A734CH	—	전원 ON 시각 이력 데이터 #5	A510~511CH의 전원 ON 시각의 4회 이전의 전원 ON 시각 데이터입니다. A732.00~A732.07: 초(00~59) A732.08~A732.15: 분(00~59) A733.00~A733.07: 시(00~23) A733.08~A733.15: 일(01~31) A734.00~A734.07: 월(01~12) A734.08~A734.15: 년(00~99)	왼쪽 설명 참조	유지	유지	전원 ON 시

주소		명칭	설명	설정 내용	운전 모드 변경 시	전원 ON 시	설정되는 타이밍/관련 영역
채널	비트						
A735~ A737CH	—	전원 ON 시각 이력 데이터 #6	A510~511CH의 전원 ON 시각의 5회 이전의 전원 ON 시각 데이터입니다. A735.00~A735.07: 초(00~59) A735.08~A735.15: 분(00~59) A736.00~A736.07: 시(00~23) A736.08~A736.15: 일(01~31) A737.00~A737.07: 월(01~12) A737.08~A737.15: 년(00~99)	왼쪽 설명 참조	유지	유지	전원 ON 시
A738~ A740CH	—	전원 ON 시각 이력 데이터 #7	A510~511CH의 전원 ON 시각의 6회 이전의 전원 ON 시각 데이터입니다. A738.00~A738.07: 초(00~59) A738.08~A738.15: 분(00~59) A739.00~A739.07: 시(00~23) A739.08~A739.15: 일(01~31) A740.00~A740.07: 월(01~12) A740.08~A740.15: 년(00~99)	왼쪽 설명 참조	유지	유지	전원 ON 시
A741~ A743CH	—	전원 ON 시각 이력 데이터 #8	A510~511CH의 전원 ON 시각의 7회 이전의 전원 ON 시각 데이터입니다. A741.00~A741.07: 초(00~59) A741.08~A741.15: 분(00~59) A742.00~A742.07: 시(00~23) A742.08~A742.15: 일(01~31) A743.00~A743.07: 월(01~12) A743.08~A743.15: 년(00~99)	왼쪽 설명 참조	유지	유지	전원 ON 시
A744~ A746CH	—	전원 ON 시각 이력 데이터 #9	A510~511CH의 전원 ON 시각의 8회 이전의 전원 ON 시각 데이터입니다. A744.00~A744.07: 초(00~59) A744.08~A744.15: 분(00~59) A745.00~A745.07: 시(00~23) A745.08~A745.15: 일(01~31) A746.00~A746.07: 월(01~12) A746.08~A746.15: 년(00~99)	왼쪽 설명 참조	유지	유지	전원 ON 시
A747~ A749CH	—	전원 ON 시각 이력 데이터 #10	A510~511CH의 전원 ON 시각의 9회 이전의 전원 ON 시각 데이터입니다. A747.00~A747.07: 초(00~59) A747.08~A747.15: 분(00~59) A748.00~A748.07: 시(00~23) A748.08~A748.15: 일(01~31) A749.00~A749.07: 월(01~12) A749.08~A749.15: 년(00~99)	왼쪽 설명 참조	유지	유지	전원 ON 시

부록-3-3 특수 보조 릴레이 보충 설명

A100~199CH ≅ ≅ 이상 내역 저장 영역



예: 메모리 이상(고장 코드: 80F1)이 1998년 4월 1일 17시 10분 30초에 발생. 메모리 이상 발생 위치는 PLC 시스템 설정(04Hex)

80	F1
00	04
10	30
01	17
98	04

예: FALS 번호 001이 1997년 5월 2일 8시 30분 15초에 발생

C1	01
00	00
30	15
02	08
97	05

고장 코드 목록/이상 내용 릴레이

종류	고장 코드	이상 명칭	이상 내용 릴레이 주소
시스템에 의한 운전 정지 이상	0x809F	사이클 시간 초과	—
	80C0~80C7, 80CE, 80CF	I/O 버스 이상	A404CH
	80E0	I/O 설정 이상	—
	80E1	I/O 점수 초과	A407CH
	80E9	유니트 번호 이중 사용 에러	A410CH, A411~A416CH(*1)
	80EA	증설 번호 이중 사용 에러	A409CH
	80F0	프로그램 에러	A295~A299CH(*2)
	80F1	메모리 이상	A403CH
	80F2	버전 이상	—
	80F4	카드 전송 이상	—
사용자 정의 운전 정지 이상	C101~C2FF	FALS 명령 이상(*3)	—
시스템에 의한 운전 계속 이상	008B	다중 리프레시 이상	A426CH
	009A	기본 I/O 유니트 이상	A408CH
	009B	PLC 시스템 설정 이상	A406CH
	00D1	옵션 보드 이상	—
	00F1	백업 메모리 이상	A315.15
	00F2	태그 메모리 이상(*4)	A315.12
	00F3	시스템 작업 이상	A402.00
	00F7	배터리 이상	—
	0200~020F	CPU 고기능 유니트 이상	A417CH
	0300~035F	고기능 I/O 유니트 이상	A418~A423CH(*5)
	0400~040F	CPU 고기능 유니트 설정 이상	A427CH
	0500~055F	고기능 I/O 유니트 설정 이상	A428~A433CH(*5)
사용자 정의 운전 계속 이상	4101~42FF	FAL 명령 이상(*6)	—

*1 번호 이중 사용 에러의 이상 내용 릴레이 내용은 다음과 같습니다.
비트 0~7: 기기 번호(BIN 데이터) 고기능 I/O 유니트의 경우: 00~5FHex
CPU 고기능 유니트의 경우: 00~0FHex
비트 8~14: 모두 0
비트 15: 유니트 종류 0: CPU 고기능 유니트
1: 고기능 I/O 유니트

*2 프로그램 에러의 이상 내용 릴레이의 내용은 295CH의 내용만 저장됩니다.

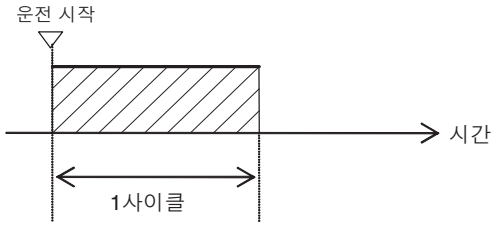
*3 FALS 번호 001~511에 대해 C101~C2FF가 저장됩니다.

*4 CJ2H-CPU6□-EIP, CJ2M-CPU3□만 해당

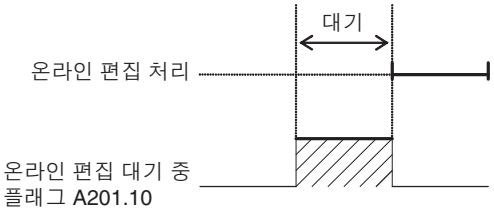
*5 이상 내용 릴레이의 내용은 0000Hex가 저장됩니다.

*6 FAL 번호 001~511에 대해 4101~42FF가 저장됩니다.

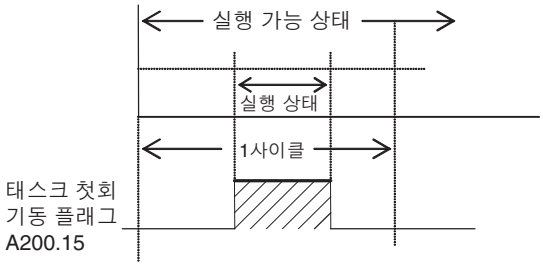
A200.11≅ 운전 시작 시 1사이클 ON 플래그



A201.10≅ 온라인 편집 대기 중 플래그

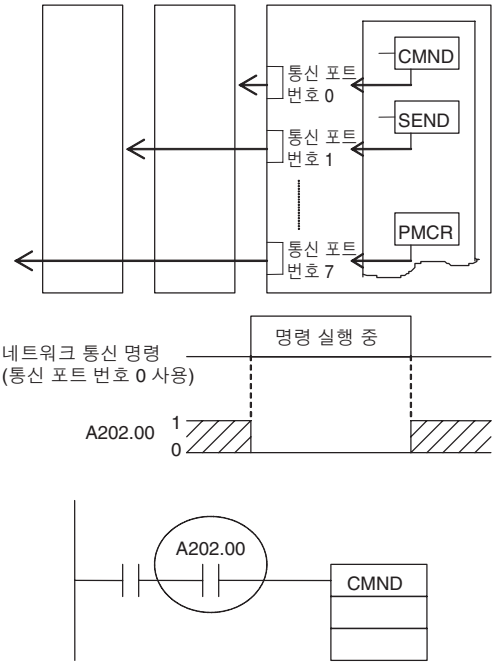


A200.15≅ 태스크 첫회 기동 플래그



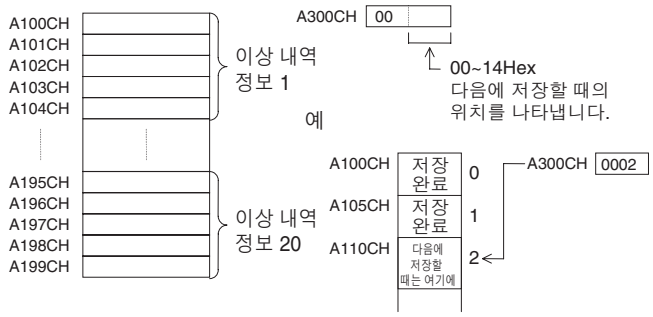
해당 태스크가 처음으로 실행 가능 상태가 되고, 실제로 태스크의 실행권을 획득하여 실행 상태가 된 동안에만 ON이 됩니다. 다음 사이클부터는 OFF가 됩니다.

A202.00~A202.07≅ 네트워크 통신 명령 실행 가능 플래그

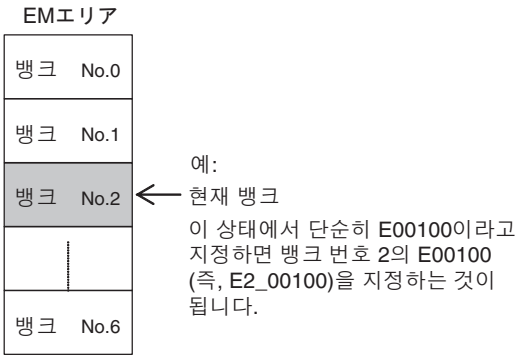


네트워크 통신 명령 실행 가능 플래그
A202.00이 ON일 때 CMND 명령을 실행하십시오.

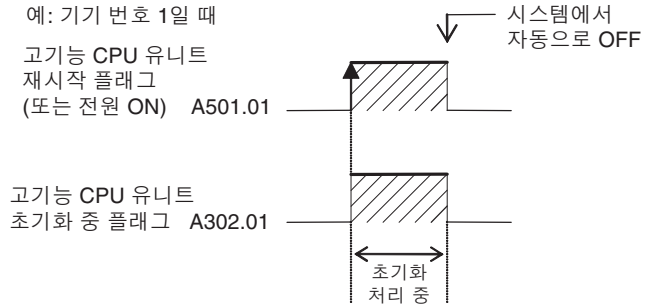
A300CH≅ 이상 내역 저장 포인터



A301CH≅ 현재 EM 뱅크



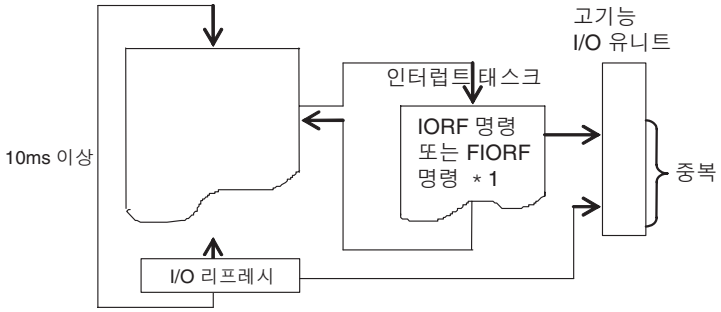
A501.01~A501.15≅ CPU 고기능 유니트 재시작 플래그



A401.09≅ 프로그램 에러 플래그

	내용	주소
프로그램 에러 A40109가 1(ON)	UM 초과 플래그	A295.15
	잘못된 명령 플래그	A295.14
	미분 초과 플래그	A295.13
	태스크 에러 플래그	A259.12
	END 없음 플래그	A295.11
	잘못된 영역 액세스 플래그	A295.10
	DM/EM 간접 BCD 에러 플래그	A295.09

A426.15≅ 다중 리프레시 이상 요인 플래그



부록-4 I/O 메모리 실제 주소

부록-4-1 I/O 메모리 실제 주소

I/O 메모리 실제 주소는 인덱스 레지스터(IR0~15)에 설정하여 I/O 메모리를 간접 지정하기 위한 주소입니다.
이 I/O 메모리 실제 주소를 인덱스 레지스터(IR0~15)에 설정하려면 기본적으로 인덱스 레지스터 설정(MOVR, MOVRW) 명령을 사용하십시오.
또한 일부 명령(테이블 데이터 처리 명령인 MAX, MIN, SRCH 명령)에서는 출력으로 반드시 인덱스 레지스터에 값(검색 결과 등의 I/O 메모리 실제 주소)이 저장됩니다.

그리고 이들 명령을 통해 인덱스 레지스터에 저장된 I/O 메모리 실제 주소에 대해 일부 명령(*)에서 인덱스 레지스터 직접 지정으로 조작하는 것도 가능합니다.

* 전송(MOVL), 비교(=L, <>L, <L, >L, <=L, >=L, CMPL), 교환(XCGL), 증가/감소(++L/--L), 가산(+L)/감산(-L), 레코드 위치 설정(SETR), 레코드 위치 읽기(GETR)

단, I/O 메모리 실제 주소는 사용자가 지정하는 I/O 메모리 영역에 대한 연속 주소이므로 각 영역 종류에 따라 순서 및 경계를 고려해야 할 수도 있습니다.
참고로 다음 페이지에 I/O 메모리 실제 주소에 따른 메모리 맵이 정리되어 있습니다.

사용상의 주의 사항

이 메모리 맵을 바탕으로 I/O 메모리 실제 주소의 값을 직접 프로그램에 설정하여 사용하면 CPU 유니트의 기종 추가/영역 배치 변경 등에 따라 프로그램 자체의 유용성이 손상될 수 있으므로 피하십시오.

사용상의 주의 사항

주소 오프셋 기능, 배열 변수 사용 시의 주의 사항

다음 기능을 사용하는 경우, 오프셋 값에 따라 I/O 메모리 실제 주소에서 다음 영역 종류를 지정할 위험이 있습니다.

- 주소 오프셋 기능으로 오프셋을 간접 지정할 때
- 배열 변수로 요소 번호를 간접 지정할 때

영역 종류를 벗어나지 않도록 프로그래밍하거나 이 메모리 맵상의 영역 종류별 배치 순서를 생각하여 프로그래밍하십시오.

부록-4-2 I/O 메모리 실제 주소 목록

I/O 메모리 영역의 I/O 메모리 실제 주소는 다음과 같습니다.
I/O 메모리 영역은 배터리에 의해 백업됩니다. 따라서 배터리가 없으면 일정치 않습니다.
(주) 시스템 예약 영역에는 액세스하지 마십시오.

분류	I/O 메모리 실제 주소(Hex)	주소	영역 종류
I/O 메모리 영역	00000 ∫ 0B7FF	—	시스템 예약
	0B800 ∫ 0B801	TK0000 ∫ TK0031	태스크 플래그
	0B802 ∫ 0B83F	—	시스템 예약
	0B840 ∫ 0B9FF	A000CH ∫ A447CH	특수 보조 릴레이(읽기 전용)
	0BA00 ∫ 0BDFF	A448CH ∫ A1471CH	특수 보조 릴레이(읽기/쓰기)
		A10000CH ∫ A11535CH	특수 보조 릴레이(읽기 전용)
	0BE00 ∫ 0BEFF	T0000 ∫ T4095	타이머 플래그
	0BF00 ∫ 0BFFF	C0000 ∫ C4095	카운터 플래그
	0C000 ∫ 0D7FF	CIO: 0000CH ∫ CIO: 6143CH	채널 I/O(CIO)
	0D800 ∫ 0D9FF	H0000CH ∫ H0511CH	유지 릴레이
	0DA00 ∫ 0DDFF	H0512CH ∫ H1535CH	유지 릴레이 이 영역은 펑션 블록 전용 영역입니다.
	0DE00 ∫ 0DFFF	W000CH ∫ W511CH	내부 보조 릴레이
	0E000 ∫ 0EFFF	T0000 ∫ T4095	타이머 현재값
	0F000 ∫ 0FFFF	C0000 ∫ C4095	카운터 현재값
	10000 ∫ 17FFF	D00000 ∫ D32767	데이터 메모리
	18000 ∫ 1FFFF	E0_00000 ∫ E0_32767	확장 데이터 메모리(EM) 뱅크 0
	20000 ∫ 27FFF	E1_00000 ∫ E1_32767	확장 데이터 메모리(EM) 뱅크 1Hex
	:	:	:
	D8000 ∫ DFFFF	E18_00000 ∫ E18_32767	확장 데이터 메모리(EM) 뱅크 18Hex
	:	:	
	F8000 ∫ FFFFF	E_00000 ∫ E_32767	확장 데이터 메모리(EM) 현재 뱅크 (*)

* 현재 프로그램에서 지정한 확장 데이터 메모리(EM)의 뱅크 데이터가 저장됩니다. 예를 들어, 뱅크 1을 지정한 경우 현재 뱅크에는 뱅크 1(20000~27FFF)과 동일한 내용이 저장됩니다.

부록-5 전원 OFF 시 PLC의 동작

부록-5-1 전원 OFF 시의 동작

CJ 시리즈 CPU 유니트는 전원 OFF 시, 다음과 같이 동작합니다.
운전 중('운전' 또는 '모니터' 모드에서 동작 중)에 전원 전압이 최저 정격 전압의 85% 이하(DC 전원 유니트 CJ1W-PD025에서는 80% 이하, CJ1W-PD022에서는 90% 이하)로 낮아지면

- CPU 유니트가 운전을 정지합니다.
- 동시에 출력 유니트의 모든 출력을 OFF합니다.

(주) 1 IOM(I/O 메모리) 유지 비트 및 전원 ON 시 'IOM(I/O 메모리) 유지 비트' 설정(PLC 시스템 설정)에 관계 없이 OFF로 합니다.

2 정격 전압의 85%(AC 전원):
AC 전원 유니트가 AC100V일 때: AC85V
AC 전원 유니트가 AC200V일 때: AC170V
AC 전원 유니트가 AC100~240V(광범위)일 때: AC85V

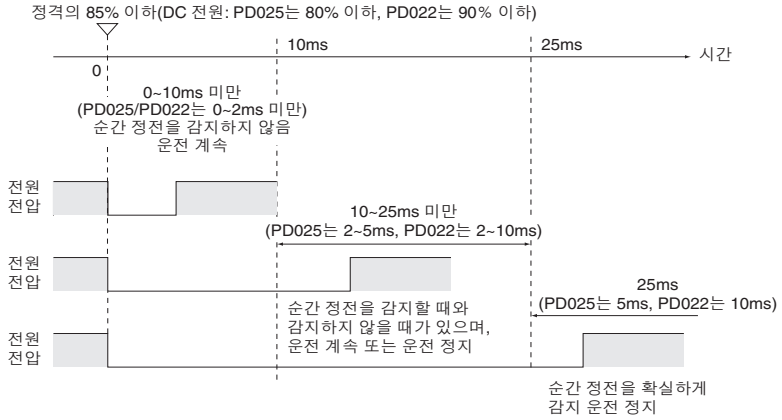
3 정격 전압의 80/90%(DC 전원):
DC 전원 유니트(CJ1W-PD025): DC19.2V
DC 전원 유니트(CJ1W-PD022): DC21.6V

● 순간 정전 시의 동작

짧은 시간의 전압 저하(순간 정전)에 대해서는 다음과 같이 동작합니다.

- 10ms 미만(DC 전원은 2ms 미만)의 순간 정지(*1)에 대해서는 운전을 계속합니다.
  *1 최저 정격 전압의 85% 이하가 된 후 동일하게 85% 이상으로 복귀할 때까지의 시간＜10ms
  DC 전원의 경우 CJ1W-PD025에서는 80% 이하에서 80% 이상으로, CJ1W-PD022에서는 90% 이하에서 90% 이상으로 복귀할 때까지의 시간＜2ms
- 10ms 이상 25ms 미만(DC 전원은 CJ1W-PD025에서 2ms 이상 5ms 미만, CJ1W-PD022에서 2ms 이상 10ms 미만)의 순간 정전인 경우는 불확정 영역으로 순간 정전을 수신할 때와 수신하지 않을 때가 있습니다.
- 25ms 이상(DC 전원의 경우 CJ1W-PD025는 5ms 이상, CJ1W-PD022는 10ms 이상)의 순간 정전에 대해서는 운전을 정지합니다.

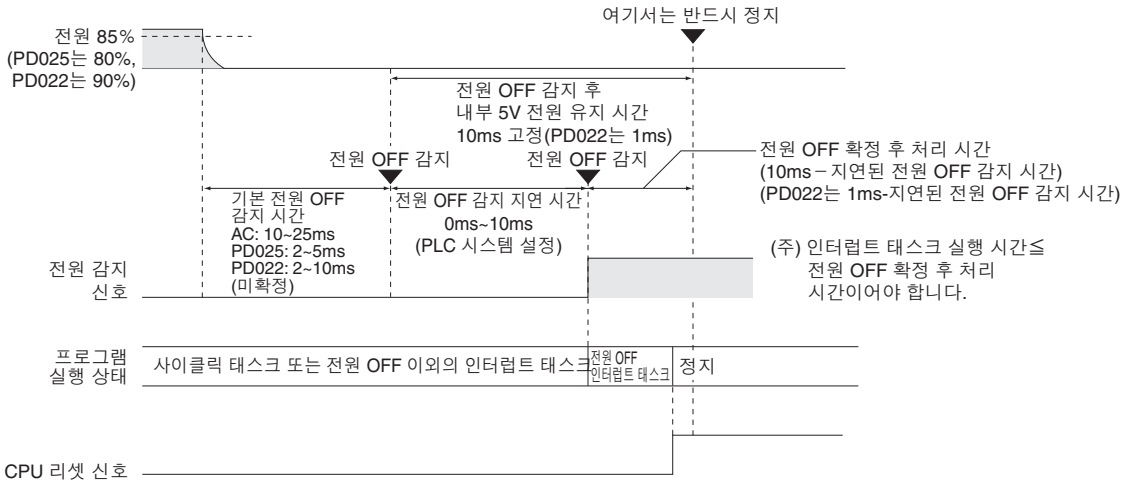
결국, 전원 OFF를 감지할 때까지의 시간은 10~25ms(DC 전원은 PD025에서 2~5ms, PD022에서 2~10ms) 사이가 됩니다. 이 시간을 PLC 시스템 설정의 전원 OFF 감지 지연 시간(0~10ms)을 설정하여 연장할 수 있습니다.
단, CJ1W-PD022에서는 전원 OFF 시의 DC5V 출력 유지 시간이 1ms로 짧기 때문에 PLC 시스템 설정의 '전원 OFF 인터럽트 태스크 유효/무효 설정'(223CH 비트 15) 및 '전원 OFF 감지 지연 시간 설정' (225CH 비트 0~7)은 설정할 수 없습니다.



(주) 위 그림은 전원 OFF 감지 지연 시간을 0ms로 한 경우의 예입니다.

전원 OFF 시의 타이밍 차트

전원 ON/OFF 시 자세한 CPU 유니트의 타이밍 차트는 다음과 같습니다.



전원 OFF 감지 시간: 전원 전압이 최저 정격 전압의 85% 이하(DC 전원 유니트 CJ1W-PD025에서는 80% 이하, CJ1W-PD022에서는 90% 이하)가 된 후 전원 OFF가 감지될 때까지의 시간

전원 OFF 감지 지연 시간: 전원 OFF 감지 후 실제로 '전원 OFF'를 확정할 때까지 전원 OFF 확정을 연장하는 시간입니다.
PLC 시스템 설정을 통해 0부터 10ms 범위에서 설정할 수 있습니다.
기본값은 0ms입니다.
전원 사정이 나쁘고 순간 정전에 의한 전원 OFF가 발생하는 경우, 이 시간을 길게(최대 10ms까지) 설정하십시오.
단, DC 전원 유니트 CJ1W-PD022에서는 1ms까지만 유지할 수 있으므로 설정할 수 없습니다.

전원 유지 시간: 전원 OFF 감지 후 내부 5V 전압을 유지하는 최대 시간을 10ms로 고정합니다.
전원 OFF 인터럽트 태스크의 실행 시간은 이 시간 10ms에서 전원 OFF 감지 지연 시간을 뺀 시간(전원 OFF 확정 후 처리 시간) 이하이어야 합니다. 이 시간을 초과하면 그 시점에서 전원 OFF 인터럽트 태스크가 강제 종료됩니다.
단, DC 전원 유니트 CJ1W-PD022에서는 1ms까지만 유지할 수 있으므로 설정할 수 없습니다.

동작 설명

AC100~240V의 전원이 최저 정격 전압의 85% 이하(DC 전원 유니트 CJ1W-PD025에서는 80% 이하, CJ1W-PD022에서는 90% 이하)로 낮아지면 전원 OFF 감지 시간(AC 전원 유니트: 10~25ms 비확정, DC 전원 유니트 CJ1W-PD025에서는 2~5ms, CJ1W-PD022에서는 2~10ms 비확정) 이후 전원 OFF가 감지됩니다.

PLC 시스템 설정에서 전원 OFF 감지 지연 시간이 설정(0ms~10ms 가능)되면 해당 시간 이후 다음 동작을 합니다.

- 전원 OFF 인터럽트 태스크가 무효인 경우(PLC 시스템 설정의 기본값)
  내부 전원이 유지되는 동안 CPU 리셋 신호가 ON이 되고 CPU 유니트는 리셋됩니다.
- 전원 OFF 인터럽트 태스크가 유효인 경우(PLC 시스템 설정에 따름)
  전원 OFF 인터럽트 태스크(프로그램) 실행 후 CPU 리셋 신호가 ON이 되고 CPU 유니트는 리셋됩니다.

또한 전원 OFF 감지 후 내부 5V 전원이 유지되는 시간은 10ms이므로 10ms－전원 OFF 감지 지연 시간＝전원 OFF 확정 후 처리 시간 이내에 전원 OFF 인터럽트 태스크(프로그램)가 실행을 종료하도록 하십시오.
단, DC 전원 유니트 CJ1W-PD022에서는 전원 OFF 감지 후 내부 5V 전원이 유지되는 시간이 1ms이므로 전원 OFF 감지 시간 연장 기능, 전원 OFF 인터럽트 태스크 기능은 사용할 수 없습니다.

부록-5-2 전원 OFF 시의 명령 실행

CPU 유니트는 운전 중('운전' 모드 또는 '모니터' 모드)에 전원 OFF가 발생하여 전원 OFF를 확정하면 실행 중인 1개 명령에 대한 실행을 완료한 후(*1) 다음과 같이 전원 OFF 처리를 실행합니다.

- 전원 OFF 인터럽트 태스크가 무효인 경우: CPU 유니트가 리셋됩니다.
- 전원 OFF 인터럽트 태스크가 유효인 경우: 전원 OFF 인터럽트 태스크를 실행한 후 CPU 유니트가 리셋됩니다.

전원 OFF 인터럽트의 유효/무효는 PLC 시스템 설정에서 설정합니다.

*1 전원 OFF가 발생한 후 1개 명령 실행의 남은 시간이 전원 OFF 확정 후 처리 시간(10ms－전원 OFF 확정 지연 시간) 이내이어야 합니다. 초과하는 경우는 전원 OFF 확정 후 처리 시간을 초과한 시점에서 명령 실행 도중이라도 위의 전원 OFF 처리를 실행합니다.

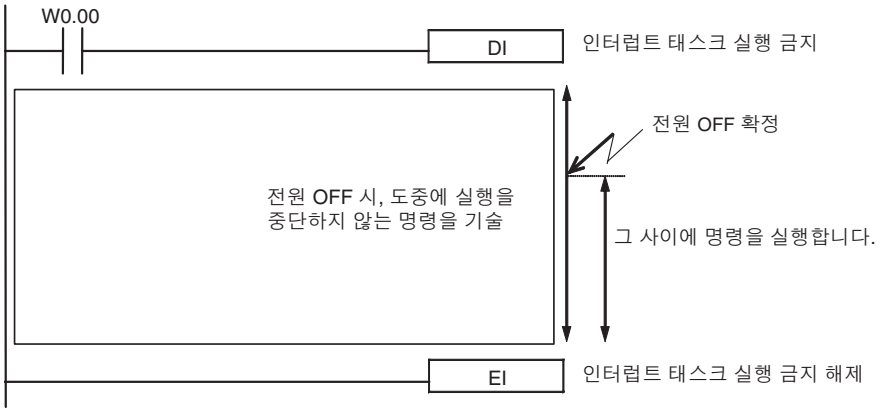
(주) CJ1W-PD022를 장착한 경우는 전원 OFF 확정 후의 처리 시간이 1ms가 됩니다.

전원 OFF 금지 구간 설정

전원 OFF 인터럽트 태스크가 무효인 경우, 운전 중에 전원 OFF가 발생해도 특정 영역(DI 명령－EI 명령 간)의 명령군에 대해서는 해당 명령군을 모두 실행한 후 전원 OFF 처리를 실행하도록 할 수 있습니다.

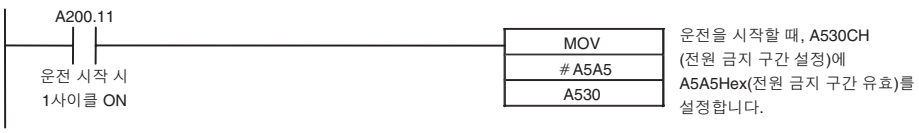
구체적으로 다음과 같이 처리합니다.

1 전원 OFF 시, 도중에 명령의 실행을 중단하고 싶지 않은 영역의 선두에 DI 명령을 기술합니다. 그리고 해당 영역의 마지막에 EI 명령을 기술합니다.



2 특수 보조 릴레이 A530CH(전원 OFF 금지 구간 설정)에 A5A5Hex(전원 OFF 금지 구간 설정 유효)를 설정합니다.

(주) A530CH(전원 OFF 금지 구간 설정)는 전원 ON 시 클리어됩니다.
따라서 PLC 시스템 설정의 '전원 ON 시 IOM(I/O 메모리) 유지 비트'를 '유지'로 설정하고, 'IOM(I/O 메모리) 유지 비트' (A500.12)를 1(ON)로 설정해서 A530CH 등의 비유지 영역을 '전원 ON 시 유지'로 설정해야 합니다. 비유지 영역을 '전원 ON 시 유지'로 설정하지 않은 경우는 다음과 같이 래더 프로그램을 통해 MOV 명령 등으로 운전을 시작할 때 A5A5Hex를 매회 설정해야 합니다.



3 PLC 시스템 설정에서 전원 OFF 인터럽트 태스크를 무효로 설정합니다.

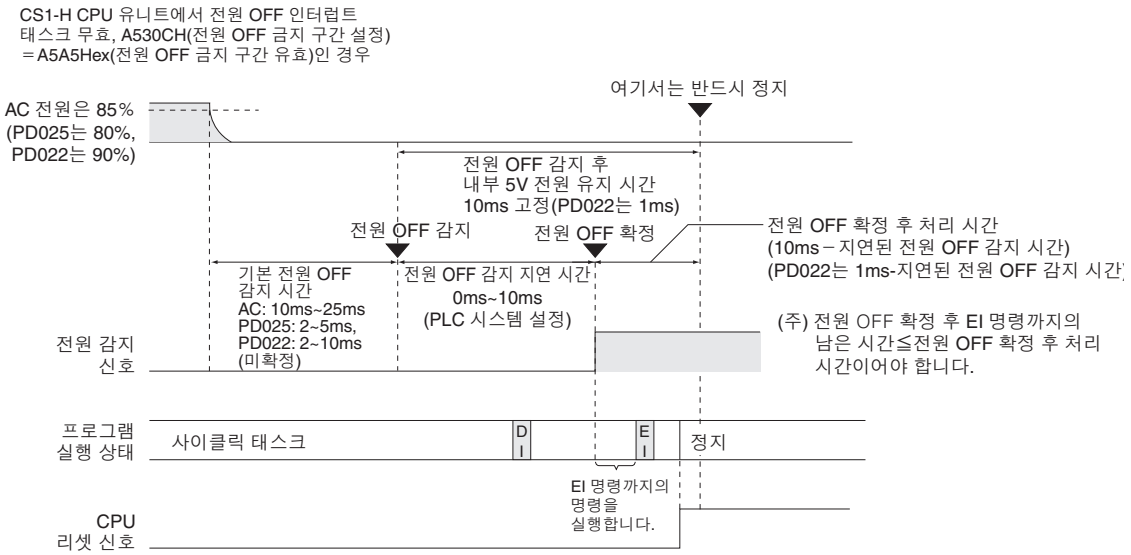
이 경우, DI 명령－EI 명령 간의 명령군 실행 중에 전원 OFF를 확정하면 DI 명령－EI 명령 간의 명령군을 반드시 실행한 후(*1) CPU 유니트가 리셋됩니다.

*1 전원 OFF가 발생한 후 EI 명령 또는 END 명령까지의 남은 시간이 전원 OFF 확정 후 처리 시간(10ms－전원 OFF 확정 지연 시간) 이내이어야 합니다. 초과하는 경우는 명령 실행 도중이라도 전원 OFF 확정 후 처리 시간을 초과한 시점에서 CPU 유니트가 리셋됩니다.

(주) 1 CJ1W-PD022를 장착한 경우는 전원 OFF 확정 후의 처리 시간이 1ms가 됩니다.

2 전원 OFF 인터럽트 태스크가 유효인 경우는 전원 OFF를 확정한 시점에서(EI 명령 또는 END 명령까지의 명령은 실행되지 않음) 즉시 전원 OFF 인터럽트 태스크를 실행한 후 CPU가 리셋됩니다.

3 DI 명령 실행 중에 전원 OFF를 확정한 경우 EI 명령 또는 END 명령까지의 명령은 실행하지 않고 그 시점에서 CPU가 리셋됩니다.



특수 보조 릴레이 A530CH(전원 OFF 금지 구간 설정)가 A5A5Hex 이외(전원 OFF 금지 구간 무효)인 경우, 평상시 대로 실행 중인 1개 명령의 실행을 완료한 후 전원 OFF 처리를 실행합니다.

참고

A530CH의 설정과 PLC 시스템 설정의 관계에 따라 전원 OFF 확정 시의 처리는 다음과 같습니다.

전원 OFF 확정 시의 처리		A530CH(전원 OFF 금지 구간 설정)	
		A5A5Hex (전원 OFF 금지 구간 설정 유효)	A5A5Hex 이외 (전원 OFF 금지 구간 설정 무효)
PLC 시스템 설정 (전원 OFF 인터럽트 태스크 유무 설정)	무효	DI－EI 간(또는 DI－END 간)의 명령군 실행 후 CPU 리셋	실행 중인 1개 명령 실행 후 CPU 리셋
	유효	실행 중인 1개 명령의 실행을 중단하고 전원 OFF 인터럽트 태스크 실행 후 CPU 리셋	

부록-6 Windows XP(SP2 이상)/Vista에서 EtherNet/IP를 연결하기 전에

Windows XP(SP2 이상)/Vista에서는 방화벽을 이용한 보안이 강화되어, Ethernet 포트의 데이터 송수신 제한이 강화됐습니다. 따라서 PC의 Ethernet 포트에서 CJ2 CPU 유니트에 EtherNet/IP 연결을 수행하는 경우, 이 항목에 기재된 방법으로 Windows 방화벽 설정을 변경하여 CX-Programmer의 통신을 사용할 수 있게 하십시오.

사용상의 주의 사항

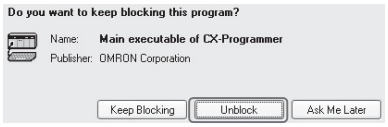
방화벽 기능은 주로 외부(인터넷 등)에서의 부정 액세스를 막기 위한 것입니다. 본 순서의 방화벽 설정 변경은 CX-Programmer에서 EtherNet/IP 연결을 사용할 수 있게 하기 위한 것이지만, PC를 사내 네트워크 등에서 사용하는 경우는 설정을 변경하기 전에 보안에 대한 영향이 없는지 확인하십시오.
또한 Ethernet 포트를 사용하여 EtherNet/IP 연결을 수행하지 않는 경우(예: 페리페럴(USB) 포트를 사용한 EtherNet/IP 연결 등)는 Windows 방화벽 설정을 변경할 필요가 없습니다.

부록-6-1 Windows 방화벽 설정 변경 순서

Windows XP의 경우

1 **CX-Programmer에서 Ethernet 포트를 통해 EtherNet/IP 네트워크의 PLC에 연결하는 경우 [Windows 보안 경고] 대화 상자가 표시됩니다.**

2 **[차단 해제]를 선택합니다.**

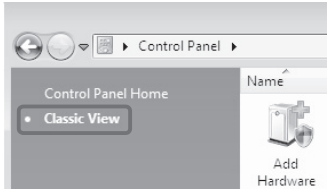


CX-Programmer의 EtherNet/IP 연결이 허가된 상태가 되며, 이후 EtherNet/IP 연결이 가능해집니다.

Windows Vista의 경우

CX-Programmer에서 연결하기 전에 다음 순서에 따라 설정합니다. 도중에 [사용자 계정 컨트롤] 대화 상자가 표시되는 경우가 있습니다. [계속]을 선택하여 설정을 계속하십시오.

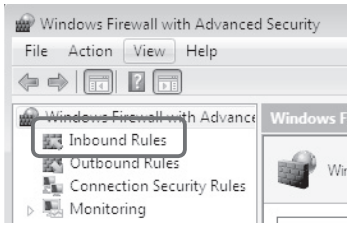
1 **시작 메뉴에서 [제어판]을 선택하고 '클래식 보기'를 선택합니다.**



2 **'관리 도구'를 선택하고 표시되는 대화 상자에서 '고급 보안이 설정된 Windows 방화벽'을 선택합니다.**



3 **'고급 보안이 설정된 Windows 방화벽' 대화 상자의 좌측에 있는 [고급 보안이 설정된 Windows 방화벽(로컬 컴퓨터)] - [인바운드 규칙]을 클릭합니다.**



4 **대화 상자의 오른쪽에 있는 [작업 영역] — [인바운드 규칙] — [새 규칙]을 클릭합니다.**



5 **[새 인바운드 규칙 마법사] 대화 상자에서 단계에 따라 각 항목을 설정합니다. 각 항목을 다음과 같이 선택하고 [다음] 버튼을 클릭합니다.**

'규칙의 종류'	[사용자 지정] 선택
'프로그램'	[모든 프로그램] 선택
'프로토콜 및 지원'	프로토콜 종류로 'ICMPv4' 선택 
'범위'	모두 [모든 IP 주소] 선택
'작업'	[연결 허용] 선택
'프로파일'	[도메인], [개인], [공용]을 모두 선택
'이름'	임의의 이름(예: Omron_EIP) 입력

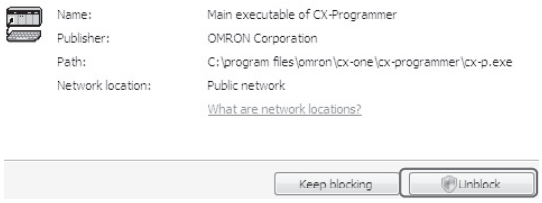
6 **이름을 입력하고 [마침] 버튼을 클릭하면 작성된 규칙(예: Omron_EIP)이 '인바운드 규칙'에 등록됩니다.**

[고급 보안이 설정된 Windows 방화벽] 대화 상자를 닫습니다.



7 **CX-Programmer에서 Ethernet 포트를 통해 EtherNet/IP 네트워크의 PLC에 연결하는 경우 [Windows 보안 경고] 대화 상자가 표시됩니다.**

8 **[차단 해제]를 선택합니다.**



CX-Programmer의 EtherNet/IP 연결이 허가된 상태가 되며, 이후 EtherNet/IP 연결이 가능해집니다.

부록-7 CJ/CS 시리즈 비교 표

CJ2 CPU 유니트와 다른 CS/CJ 시리즈의 주요 차이점은 다음과 같습니다.

기능 비교

항목		CJ 시리즈					CS 시리즈
		CJ2H CPU 유니트	CJ2M CPU 유니트	CJ1-H-R CPU 유니트	CJ1-H CPU 유니트	CJ1-M CPU 유니트	CS1-H CPU 유니트
일반 사양	외형 규격 높이×깊이×폭	CJ2H-CPU6□-EIP: 90×65×80mm CJ2H-CPU6□: 90×65×49mm	CJ2H-CPU1□: 90×75×31mm CJ2M-CPU3□: 90×75×62mm	90×65×62mm		90×65×31mm 90×65×49mm	130×101×71mm
	질량	CJ2H-CPU6□-EIP: 280g 이하 CJ2H-CPU6□: 190g 이하	CJ2M-CPU1□: 130g 이하 CJ2M-CPU3□: 190g 이하 (시리얼 옵션 보드가 없는 상태)	200g 이하		120g 이하 170g 이하	350g 이하
	소비 전류 DC5V	CJ2H-CPU6□-EIP: 0.82A CJ2H-CPU6□: 0.42A	CJ2M-CPU1□: 0.5A CJ2M-CPU3□: 0.7A	0.99A		0.58A 0.64A	0.82A
	내진동	JIS C60068-2-6 준수	JIS C60068-2-6 준수	JIS C0040 준수			
	내충격	JIS C60068-2-27 준수	JIS C60068-2-27 준수	JIS C0041 준수			

항목		CJ 시리즈					CS 시리즈
		CJ2H CPU 유니트	CJ2M CPU 유니트	CJ1-H-R CPU 유니트	CJ1-H CPU 유니트	CJ1-M CPU 유니트	CS1-H CPU 유니트
성능 사양	최대 프로그램 용량	400K 스텝	60K 스텝	250K 스텝			
	최대 입출력 점수	2560점	2560점				5120점
	공통 처리 시간	CJ2H-CPU6□-EIP: 200μs (EtherNet/IP 포트의 오버헤드 포함) CJ2H-CPU6□: 100μs	CJ2M-CPU3□: 270μs CJ2M-CPU1□: 160μs (EtherNet/IP 태그 데이터 링크를 사용하는 경우, 위 내용에 100μs +전송 CH 수×1.8μs 를 가산)	130μs	300μs	500μs	300μs
	기본 명령 처리 시간 LD	0.016μs	0.04μs		0.02μs	0.1μs	0.02μs
	응용 명령 처리 시간 MOV	0.048μs	0.06μs		0.18μs	0.3μs	0.18μs
	매회 리프레시 명령 시간 !LD	0.99μs	1.26μs	21μs		24μs, 28μs	21μs
	인터럽트 태스크 기동 시간	26μs (유니트 Ver.1.0에서는 30μs)	31μs	40μs	124μs	169μs	124μs
	기본 I/O 리프레시 (16점 유니트)	1.4μs	3.9μs		3μs		
	고기능 I/O 리프레시 (아날로그 8점 입력)	50μs	70μs		120μs	160μs	120μs
	CPU 고기능 리프레시 (EtherNet/IP 유니트에서 태그 데이터 링크 사용 시)	100μs+전송 CH 수 ×0.33μs	100μs+전송 CH 수 ×0.7μs	200μs+전송 CH 수 ×0.7μs		270μs+ 전송 CH 수× 0.7μs	200μs+ 전송 CH 수× 0.7μs
	최대 확장 데이터 메모리	32K 워드×25뱅크	32K 워드×4뱅크	32K 워드×13뱅크			
	소스/코멘트 메모리	3.5MB	1024KB	2MB			
	FB 정의 수/작업 수/ 트랜지션 수	2048개	CJ2M-CPU□1/□2/ □3: 256개 CJ2M-CPU□4/ □5: 2048개	1024개			
	사용 가능한 메모리 카드	30MB 64MB 128MB 256MB 512MB	128MB 256MB 512MB	30MB 64MB 128MB			
	사이클릭 태스크 수	128	128	32			
	내부 논리 포트	8+64개	8+64개	8개			
	온라인 편집에 의한 사이클 지연 시간	약 1ms	약 1ms	약 8ms		약 14ms	약 8ms
	I/O 증설 장치 연결 대수	3랙	3랙				7랙
	인터럽트 입력 유니트 장착 수	최대 2대	최대 2대				최대 4대
	CPU 고기능 유니트 장착 수	15대	CJ2M-CPU1□: 16대 CJ2M-CPU3□: 15대	16대			
	INNER 보드	사용 불가능	사용 불가능				사용 가능

항목		CJ 시리즈					CS 시리즈
		CJ2H CPU 유니트	CJ2M CPU 유니트	CJ1-H-R CPU 유니트	CJ1-H CPU 유니트	CJ1-M CPU 유니트	CS1-H CPU 유니트
기능 사양	**실행 모드**	통상 모드	통상 모드	통상 모드 병렬 처리 모드			
	유니트 간 동기 기능	있음(유니트 Ver.1.1 이상만 해당)	없음	없음			
	특정 고기능 유니트 전용 명령	가능(유니트 Ver.1.1 이상만 해당)	가능	없음			
	사이클 시간 일정화	0.2~32000ms 0.1ms 단위 운전 중에 고정 사이클 시간 변경 가능(유니트 Ver.1.1 이상)	0.2~32000ms 0.1ms 단위 운전 중에 고정 사이클 시간 변경 가능		1~32000ms 1ms 단위		
	사이클 시간 모니터 단위	0.01ms 단위	0.01ms 단위	0.1ms 단위			
	예약 인터럽트 시간 단위	0.1ms 1ms 10ms	0.1ms 1ms 10ms	1ms 10ms		0.1ms 1ms 10ms	1ms 10ms
	통신 포트	USB RS-232C EtherNet/IP (CJ2H-CPU6□-EIP만 해당)	USB RS-232C RS-422/485 (CJ2M-CPU3□의 경우 시리얼 옵션 보드 장착 시) EtherNet/IP (CJ2M-CPU3□만 해당)	페리페럴 RS-232C			
	시리얼 PLC 링크	불가능	가능			가능	불가능
	태그 이름 서버 기능	CJ2H-CPU6□-EIP: 있음 CJ2H-CPU6□: 없음	CJ2M-CPU3□: 있음 CJ2M-CPU1□: 없음	없음			
	외부 인터럽트 대상 슬롯	CJ2H-CPU6□-EIP: 0~3슬롯 CJ2H-CPU6□: 0~4슬롯	0~4슬롯	0~4슬롯			0~9슬롯
	데이터 트레이스	최대 32K 워드 샘플링 주기 1~2550ms 연속 트레이스 가능	최대 8K 워드 샘플링 주기 1~2550ms 연속 트레이스 가능	최대 4K 워드 샘플링 주기 10~2550ms			
	제조 로트 번호에 의한 쓰기 방지 기능	있음	있음	없음			
	PLC 이름 기능	있음	있음	없음			
	DM/EM의 비트 참조	가능	가능	불가능			
	EM의 강제 세트/리셋	주소 자동 할당을 실행한 경우 가능 또는 유니트 Ver.1.2 이상에서는 EM 강제 세트/리셋 기능에 의해 가능	EM 강제 세트/리셋 기능에 의해 가능	불가능			
	EM의 트레이스 메모리화	가능	가능	불가능			
	주소의 오프셋 지정	가능	가능	불가능			
	타이머 명령: BCD/BIN의 동일 프로그램 내 공존	가능	가능	불가능			
	라우팅 테이블의 중계 네트워크 테이블 등록 수	최대 128	최대 128		최대 32		
	클록 펄스	0.1ms, 1ms, 0.01s, 0.02s, 0.1s, 0.2s, 1s, 1분	0.1ms, 1ms, 0.01s, 0.02s, 0.1s, 0.2s, 1s, 1분		0.02s, 0.1s, 0.2s, 1s, 1분		

항목		CJ 시리즈					CS 시리즈
		CJ2H CPU 유니트	CJ2M CPU 유니트	CJ1-H-R CPU 유니트	CJ1-H CPU 유니트	CJ1-M CPU 유니트	CS1-H CPU 유니트
기능 사양	펄스 입출력 기능 구현 방법	CPU 유니트만으로는 불가능 트랜지스터 NPN 출력만 가능		CPU 유니트만으로는 불가능 트랜지스터 NPN 출력만 가능		CPU 유니트 내장 입출력타입만 가능	CPU 유니트만으로는 불가능 트랜지스터 NPN 출력만 가능
	범용 입력	없음		없음		10점	없음
	인터럽트 입력	없음		없음		최대 4점 입력 신호의 최소 펄스 폭: 30μs	없음
	펄스 캐치 입력	없음		없음		4점 (입력 접점은펄스 캐치, 고속 카운터, 범용 입력과 겸용)	없음
	고속 카운터 입력	없음		없음		2점(고속 카운터 0/1)	없음
	범용 출력	없음		없음		6점	없음
	펄스 출력	없음		없음		2점	없음
	PWM 출력	없음		없음		CJ1M-CPU22/23: 2점 CJ1M-CPU21: 1점	없음
	원점 검색/복귀 기능	없음		없음		2점	없음

부록-8 유니트 버전에 따른 지원 기능 목록

유니트 Ver.1.3 이상 지원 기능 목록

유니트 Ver.1.3에서 추가된 기능을 사용할 때는 CX-Programmer의 Ver.9.1 이상을 사용해야 합니다.

○: 지원, ×: 지원 안 함

기종		CJ2H CPU 유니트			
형식		CJ2H-CPU6□-EIP CJ2H-CPU6□			
기능 \ 유니트 버전		유니트 Ver.1.3 이상	유니트 Ver.1.2	유니트 Ver.1.1	유니트 Ver.1.0
특정 고기능 유니트 전용 명령	위치 제어 유니트 CJ1W-NC281/481/881 전용: 위치 결정 고속 기동(NCDMV) 명령	○	×	×	×
	위치 제어 유니트 CJ1W-NC281/481/881 전용: 위치 결정 시작 트리거(NCDTR) 명령	○	×	×	×
신규 추가 응용 명령	부호 있는 영역 비교(ZCPS) 명령	○	×	×	×
	부호 있는 2워드 영역 비교 (ZCPSL) 명령	○	×	×	×

유니트 Ver.1.2 이상 지원 기능 목록

유니트 Ver.1.2에서 추가된 기능을 사용할 때는 CX-Programmer Ver.8.3 이상을 사용해야 합니다.

○: 지원, ×: 지원 안 함

기종	CJ2H CPU 유니트	
형식	CJ2H-CPU6□-EIP CJ2H-CPU6□	
기능 \ 유니트 버전	유니트 Ver.1.2 이상	왼쪽 유니트 이외
EM 강제 세트/리셋 기능	○	—

(주) CJ2H CPU 유니트의 유니트 Ver.1.2 이상의 기능을 사용한 사용자 프로그램은 CJ2H CPU 유니트의 유니트 Ver.1.1 이하에서는 사용할 수 없습니다.
이러한 기능을 사용한 프로그램을 CX-Programmer에서 유니트 Ver.1.1 이하의 CPU 유니트로 전송하려고 하면 에러가 표시되며 CPU 유니트에 다운로드할 수 없습니다.

유니트 Ver.1.1 이상 지원 기능 목록

유니트 Ver.1.1에서 추가된 기능을 사용할 때는 CX-Programmer Ver.8.1(*) 이상을 사용해야 합니다.

○: 지원, −: 지원 안 함

기종	CJ2H CPU 유니트	
형식	CJ2H-CPU6□-EIP CJ2H-CPU6□	
기능 \ 유니트 버전	유니트 Ver.1.1 이상	유니트 Ver.1.0
고속 인터럽트 기능 인터럽트 태스크의 오버헤드 시간 단축 예약 인터럽트 태스크의 최소 간격 0.1ms 설정	○	−
고정 사이클 시간의 운전 중 변경 기능	○	−
유니트 간 동기 제어 기능	○	−
특정 고기능 유니트 전용 매회 리프레시 명령 추가 CJ1W-AD042 전용: 아날로그 입력 직접 변환(AIDC) 명령 CJ1W-DA042V 전용: 아날로그 출력 직접 변환(AODC) 명령	○	−
고속 시리얼 커뮤니케이션 유니트 CJ1W-SCU22/32/42 전용: 시리얼 커뮤니케이션 유니트 시리얼 포트 직접 수신(DRXDU) 명령 시리얼 커뮤니케이션 유니트 시리얼 포트 직접 송신(DTXDU) 명령	○	−

* 고속 인터럽트 기능과 고정 사이클 시간의 운전 중 변경 기능에 대해서만 CX-Programmer Ver.8.02에서도 지원합니다.

(주) CJ2H CPU 유니트의 유니트 Ver.1.1 이상의 기능을 사용한 사용자 프로그램은 CJ2H CPU 유니트의 유니트 Ver.1.0 이하에서는 사용할 수 없습니다.
이러한 기능을 사용한 프로그램을 CX-Programmer에서 유니트 Ver.1.0의 CPU 유니트로 전송하려고 하면 에러가 표시되며 CPU 유니트에 다운로드할 수 없습니다.
또한 이러한 기능을 사용한 프로그램 파일(확장자: .OBJ)을 유니트 Ver.1.0의 CPU 유니트에 전송하면 운전 시작 또는 기능 동작 시에 프로그램 에러가 발생하고 CPU 유니트는 운전을 정지합니다.

색인

가

간이 백업 7-12, 7-14
강제 상태 유지 플래그 9-4
강제 세트/리셋 10-61
고기능 I/O 리프레시 9-17
고기능 I/O 유니트 릴레이 6-2, 6-14
고기능 I/O의 주기적 리프레시 금지 설정 9-17
고기능 유니트 설정 8-20
고기능 유니트용 할당 DM 영역 4-14
고속 인터럽트 기능 10-18
고속 인터럽트 기능 사용 9-16
고장 지점 검출 명령(FPD 명령) 기능 10-73
고장 진단 명령(FAL 명령, FALS 명령) 기능 10-71
고정 사이클 시간 9-13
공통 처리 3-4
구조체 5-58
글로벌 변수 5-48, 5-50

나

내부 메모리 2-2
내부 보조 릴레이 6-2, 6-17
내장 RAM 2-3
내장 플래시 메모리 2-2
네트워크 변수(태그) 2-3, 5-48, 5-51
네트워크를 경유하는 FINS 금지 설정 9-28

다

데이터 레지스터 6-2, 6-37
데이터 링크 릴레이 6-2, 6-11
데이터 메모리 6-2, 6-21
데이터 트레이스 10-66
데이터 파일 7-9
데이터 형식 5-51, 5-56
동기 데이터 링크 릴레이 10-82
동기 제어 주기 10-76, 10-89
동작 모드 3-8, 9-6
동작 모드 변경 및 전원 ON 시의 유지 설정 기능 10-20
디버그 기능 10-61
딥 스위치 4-2
딥 스위치에 의한 쓰기 방지 기능 10-46

라

라우팅 테이블 2-3, 4-5, 4-10
래더 언어 5-6
랙 시작 주소 8-14
랙 입력 응답 시간 9-19
로컬 변수 5-48, 5-50
로터리 스위치 4-2

마

매개 변수 영역 2-3
매회 갱신 5-81
메모리 관리 기능 10-35
메모리 카드 7-2, 7-5
메모리 카드에 저장 금지/허가 설정 10-50
메모리 카드의 인식 시간 7-6
명령
  데이터 지정 방법 5-72
  데이터 형식 5-77
  동작 옵션 5-65
  명령 조건 5-63
  실행 조건 5-66
  전원 흐름 5-62
  플래그 5-63
  피연산자 5-64
  피연산자 지정 방법 5-70
명령 목록 ït-2
명령 처리 시간 ït-75
명령 처리 오류 5-100
모니터 모드 3-8
모니터 사이클 시간 9-12
무수순 시스템 11-17
미분 모니터 10-62

바

배열 지정 5-57
배터리 없이 운전 9-8
백그라운드 처리 9-10
백그라운드 처리 실행 기능 10-10
백업 메모리 2-2, 10-35
백업 복원 7-13
범용 파일 7-16
변수 5-47
변수 테이블 2-3, 5-47
변수 테이블 파일 7-9
보호 기능 10-46
  FINS 쓰기 방지 기능 10-54
  딥 스위치에 의한 쓰기 방지 기능 10-46
  비밀번호를 사용한 읽기 방지 기능 10-47
  제조 로트 번호에 따른 금지 기능 10-53
부하 차단 10-65
블록 프로그램 영역 5-109
비밀번호를 사용한 읽기 방지 기능 10-47

사

사용자 메모리 재기입 시각 10-5
사용자 정의 형식 5-58
사용자 프로그램 영역 2-3
사용자 프로그램 영역 초과 오류 5-102
사이클 시간 3-4, 12-2
사이클 시간 감시 기능 10-8
사이클 시간 계산 방법 12-4

사이클 시간 모니터링 기능 10-9
사이클 시간 연장 시간 12-11
사이클 시간 일정화 기능 10-7
사이클릭 태스크 5-14
상위 링크 시스템 11-16
서브루틴 프로그램 영역 5-109
설명 2-3
섹션 5-40
소스 2-3
소스/설명 영역 2-3
스텝 래더 영역 5-109
스텝 수 ït-75
슬롯 시작 주소 8-12
시각 기록 기능 10-4
시계 기능 10-3
시리얼 PLC 링크 11-21
시리얼 PLC 링크 릴레이 6-2, 6-15
시리얼 PLC 링크의 응답 성능 12-13
시리얼 게이트웨이 11-19
시리얼 포트 9-21
시뮬레이터 기능 5-97
시스템 이상을 고의로 발생시키는 기능 10-72

아

에러 발생 시 정지 위치 저장 기능 10-70
연산 처리 3-4
예약 인터럽트 단위 시간 설정 9-14
예약 인터럽트 태스크 5-19, 5-22
오류
  P_ARR 플래그 ON 오류 5-101
  명령 처리 오류 5-100
  사용자 프로그램 영역 초과 오류 5-102
  오류 플래그 ON 오류 5-100
  잘못된 명령 오류 5-102
  잘못된 영역 액세스 오류 5-101
  프로그램 오류 5-102
오류 플래그 ON 오류 5-100
온라인 편집 10-63
외부 인터럽트 태스크 5-19, 5-28
운전 계속 고장 진단 명령 10-71
운전 모드 3-8
운전 시작 시 동작 설정 및 보수 기능 10-20
운전 시작 시각 10-5
운전 시작 조건 9-6
운전 정지 고장 진단 명령 10-71
운전 정지 시각 10-5
운전 중 출력 기능 10-24
유니트 간 동기 제어 기능 10-75
유니트 관련 관리 기능 10-32
유니트 백업 파일 7-10
유니트 설정 9-19
유지 릴레이 6-2, 6-18
이상 발생 시뮬레이션 5-99
인덱스 레지스터 5-85, 6-2, 6-32
인터럽트 오버헤드 시간 5-20, 5-23, 5-27
인터럽트 태스크 5-19
일시 기억 릴레이 6-2, 6-20
입출력 릴레이 6-2, 6-7
입출력 응답 시간 12-12

자

자동 백업 기능 10-35
자동 온라인 연결 11-3, 11-11
자체 네트워크 테이블 4-11
잘못된 명령 오류 5-102
잘못된 영역 액세스 오류 5-101
전원 OFF 감지 지연 시간 설정 9-14
전원 OFF 금지 구간 설정 10-23
전원 OFF 시 PLC의 동작 ït-126
전원 OFF 시각 10-4
전원 OFF 시의 처리 3-7, ït-126
전원 OFF 인터럽트 태스크 5-19
전원 OFF 인터럽트 태스크 무효 설정 9-15
전원 ON 시 동작 설정 9-4
전원 ON 시 유지 설정 9-4
전원 ON 시 자동 전송 7-12
전원 ON 시 자동 전송 기능 10-25
전원 ON 시 자동 전송용 파일 7-14
전원 ON 시 초기 처리 3-2
전원 끔 감지 지연 시간 설정 10-22
전원 끔 인터럽트 태스크 5-20
전원 투입 시각 10-4
제조 로트 번호에 따른 금지 기능 10-53
조건 플래그 5-103, 6-2, 6-39
주기적 갱신 5-80
주변 서비스 3-5, 9-27
  CPU 고기능 유니트의 이벤트 서비스 3-5
  RS-232C 포트 서비스 3-5
  고기능 I/O 유니트의 이벤트 서비스 3-5
  내장 USB 포트 서비스 3-5
  백업 서비스 3-5
  온라인 편집 서비스 3-5
  통신 포트 서비스 3-5
  파일 액세스 서비스 3-5
주변 서비스 일률 시간 9-27
주변 툴/기기와 연결 11-1
주소 자동 할당 5-61, 6-24
주소의 오프셋 지정 5-92
중계 네트워크 테이블 4-11

차

초기화 7-3
추가 태스크 5-19, 5-29

카

카운터 6-2, 6-30
카운터 현재값 6-2, 6-30
카운트 업 플래그 6-2, 6-30
코멘트 기억 기능 10-38
코멘트 파일 7-10
클록 펄스 6-2, 6-41

타

타이머 ... 6-2, 6-28
타이머 인터럽트 ... 9-12
타이머 현재값 ... 6-2, 6-28
타임업 플래그 ... 6-2, 6-28
태그 ... 5-51
태그 메모리 ... 2-2
태스크 ... 5-11
태스크 관련 플래그 ... 5-33
태스크 기동/정지 ... 5-16
태스크 플래그 ... 6-2, 6-31
테스트 입력 ... 10-62
통신 설정 ... 9-21
통전 시간 ... 10-4
통합 시뮬레이션 ... 5-98
트레이스 메모리 ... 6-26
특수 보조 릴레이 ... 6-2, 6-19
특수 보조 릴레이 목록 ... ït-96
특정 프로그램 영역 ... 5-109

파

파라미터 파일 ... 7-8
파일 메모리 ... 6-25, 7-2
펄스 캐치 기능 ... 10-9
프로그램 교체 기능 ... 10-40
프로그램 모드 ... 3-8
프로그램 언어 ... 5-3
프로그램 영역
  FOR-NEXT 영역 ... 5-109
  IL-ILC 영역 ... 5-109
  JMP0-JME0 영역 ... 5-109
  블록 프로그램 영역 ... 5-109
  서브루틴 프로그램 영역 ... 5-109
  스텝 래더 영역 ... 5-109
프로그램 오류 ... 5-102
프로그램 용량 ... 5-5
프로그램 인덱스 ... 2-3
프로그램 인덱스 파일 ... 7-10
프로그램 확인 ... 5-96
프로그램/네트워크 변수 파일 ... 7-8
프로그램의 덮어쓰기 금지/허가 설정 ... 10-51
프로토콜 매크로 ... 11-20
프리 러닝 타이머 기능 ... 10-6

하

함수 블록 ... 5-42
현재 뱅크 ... 6-26
확장 데이터 메모리 ... 6-2, 6-24

A

AR 특수 보조 릴레이 ... 6-2, 6-19

C

C 카운트 업 플래그 ... 6-2, 6-30
CIO(채널 I/O) 영역 ... 6-2
CPU 고기능 시스템 설정 ... 2-3
CPU 고기능 유니트 릴레이 ... 6-2, 6-13
CPU 고기능 유니트 시스템 설정 ... 4-5, 4-15
CPU 실행 처리 ... 9-7
CPU 실행 처리 모드 설정 ... 9-27
CPU 유니트 설정 ... 9-7

D

DeviceNet 릴레이 ... 6-2, 6-16
DM 데이터 메모리 ... 6-2, 6-21
DR 데이터 레지스터 ... 6-2, 6-37

E

EM 파일 메모리 ... 7-2, 10-38
EM 파일 메모리 기능 ... 10-38
EM 확장 데이터 메모리 ... 6-2, 6-24

F

FAL 명령(고장 진단 명령) ... 3-11
FAL(운전 계속 고장 진단) 명령 ... 10-71
FALS 명령(고장 진단 명령) ... 3-11
FALS(운전 정지 고장 진단) 명령 ... 10-71
FB 내 통신 명령 처리 ... 9-11
FB 프로그램 영역 ... 2-3
FINS 금지 ... 9-28
FINS 메시지 서비스 ... 11-29
FINS 쓰기 방지 기능 ... 10-54
FOR-NEXT 영역 ... 5-109
FPD 명령 기능 ... 10-73

H

HR 유지 릴레이 ... 6-2, 6-18

I

I/O 갱신 ... 3-6, 5-80
I/O 갱신 타이밍 ... 5-80
I/O 리프레시 시간 ... 12-7
I/O 메모리 ... 6-2
I/O 메모리 실제 주소 ... ït-124
I/O 메모리 영역 ... 2-3
I/O 인터럽트 태스크 ... 5-19, 5-25
I/O 테이블 ... 2-3, 4-4, 8-3
  수동 할당 ... 4-9, 8-9
  자동 할당 ... 4-9, 8-5
I/O 테이블 조회 ... 8-10
I/O 할당 ... 8-2
  수동 할당 ... 8-9
  자동 할당 ... 8-5
IL-ILC 영역 ... 5-109
IOM(I/O 메모리) 유지 비트 ... 9-5
IR 인덱스 레지스터 ... 6-2, 6-32

J

JMP0-JME0 영역 5-109

N

NT 링크 시스템 11-18

P

P_AER 플래그 ON 오류 5-101
PLC 백업 7-13, 7-15
PLC 시스템 설정 2-3, 4-4, 4-7, 9-2
CPU 실행 처리 9-7
CPU 실행 처리 모드 설정 9-27
CPU 유니트 설정 9-7
FB 내 통신 명령 처리 9-11
FINS 금지 9-28
IOM(I/O 메모리) 유지 비트 9-5
강제 상태 유지 플래그 9-4
고기능 I/O 리프레시 9-17
고기능 I/O의 주기적 리프레시 금지 설정 9-17
고속 인터럽트 기능 사용 유무 설정 9-16
고정 사이클 시간 9-13
네트워크를 경유하는 FINS 금지 설정 9-28
동작 모드 9-6
랙 입력 응답 시간 9-19
모니터 사이클 시간 9-12
배터리 없이 운전 9-8
백그라운드 처리 9-10
시리얼 포트 9-21
예약 인터럽트 단위 시간 설정 9-14
운전 시작 조건 9-6
유니트 설정 9-19
전원 OFF 감지 지연 시간 설정 9-14
전원 OFF 인터럽트 태스크 무효 설정 9-15
전원 ON 시 동작 설정 9-4
전원 ON 시 유지 설정 9-4
주변 서비스 9-27
주변 서비스 일률 시간 9-27
타이머 인터럽트 9-12
통신 설정 9-21
PLC 이름 2-3, 4-4, 10-58

S

SFC 언어 5-9
SmartFB 라이브러리 전용 PLC 시스템 설정 9-11
ST 언어 5-8

T

T 타임업 플래그 6-2, 6-28
TK 태스크 플래그 6-2, 6-31
TR 일시 기억 릴레이 6-2, 6-20

W

WR 내부 보조 릴레이 6-2, 6-17

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
A	2008년 6월	초판
B	2008년 12월	・CJ2H-CPU6□ 추가 ・CJ2 CPU 유니트의 유니트 Ver.1.1 추가 ・기본 I/O 유니트 CJ1W-ID212, CJ1W-ID233, CJ1W-OD213, CJ1W-OD234 추가
C	2009년 2월	・유니트 간 동기 제어 기능에 관한 내용 추가 ・위치 제어 유니트 CJ1W-NC214/234/414/434 추가
D	2009년 7월	・아날로그 입력 유니트 CJ1W-AD042 추가 ・아날로그 출력 유니트 CJ1W-DA042V 추가
E	2009년 10월	・시리얼 커뮤니케이션 유니트 CJ1W-SCU22/32/42 추가 ・DTXDU, DRXDU 명령 추가 ・유니트 Ver.1.2의 추가 기능, EM 강제 세트/리셋 기능에 관한 내용 추가
F	2010년 2월	・CJ2M-CPU□□ 추가

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-350F 2010년 8월 현재

© OMRON Corporation 2008-2010 All Rights Reserved.
예고없이 사양 등을 변경하는 경우가 있으므로 양해 부탁드립니다.