Cat. No. W473-E1-16

SYSMAC CJ 시리즈

CJ3M-CPU□□
CJ2H-CPU6□-EIP
CJ2H-CPU6□
CJ2M-CPU□□

CJ2/CJ3 CPU 유닛 소프트웨어

사용자 매뉴얼

⚠ 안내 (For Reference Only)

본 문서는 AI 번역으로 작성된 참고용 국문자료 입니다.
일부 표현은 원문과 다소 다를 수 있으며,
중요한 내용은 원문과 함께 참고해 주시기 바랍니다.

OMRON

주기

모든 권리를 보유합니다. 본 간행물의 어떠한 부분도 오므론의 사전 서면 허가 없이 기계적, 전자적, 복사, 녹음, 그 밖의 어떠한 형태나 수단으로도 복제, 검색 시스템에 저장 또는 전송할 수 없습니다.

본서에 수록된 정보의 사용에 관하여 특허상의 책임은 지지 않습니다. 또한 오므론은 고품질 제품의 개선에 끊임없이 노력하고 있으므로, 본 매뉴얼에 수록된 정보는 예고 없이 변경될 수 있습니다. 본 매뉴얼의 작성에는 만전을 기했습니다. 그럼에도 불구하고 오므론은 오류나 누락에 대해 책임을 지지 않습니다. 또한 본 간행물에 수록된 정보의 사용으로 인한 손해에 대해서도 어떠한 책임도 지지 않습니다.

AI 번역본
-For Reference Only-

상표

・Sysmac 및 SYSMAC은 오므론 팩토리 오토메이션 제품에 대한 일본 및 그 밖의 국가에서의 OMRON Corporation의 상표 또는 등록상표입니다.
・CX-One은 OMRON Corporation이 제작한 프로그래밍 소프트웨어의 등록상표입니다.
・Microsoft, Windows, Excel, Visual Basic 및 Microsoft Edge는 미국 및 그 밖의 국가에서의 Microsoft Corporation의 등록상표 또는 상표입니다.
・EtherCAT®은 독일 Beckhoff Automation GmbH가 라이선스한 등록상표이자 특허 기술입니다.
・ODVA, CIP, CompoNet, DeviceNet 및 EtherNet/IP는 ODVA의 상표입니다.
・SD 및 SDHC 로고는 SD-3C, LLC의 상표입니다.

본 문서에 기재된 그 밖의 회사명 및 제품명은 각 사의 상표 또는 등록상표입니다.

저작권

・Microsoft 제품 화면은 Microsoft의 허가를 받아 사용했습니다.
・본 제품에는 일부 서드파티 소프트웨어가 포함되어 있습니다. 이 소프트웨어와 관련된 라이선스 및 저작권 정보는 https://www.fa.omron.co.jp/product/tool/cj-info/index_en.html 에서 확인할 수 있습니다.

SYSMAC CJ 시리즈
CJ3M-CPU□□
CJ2H-CPU6□-EIP
CJ2H-CPU6□
CJ2M-CPU□□
CJ2/CJ3 CPU 유닛 소프트웨어

사용자 매뉴얼

2026년 10월 개정

AI 번역본
-For Reference Only-

AI 번역본
-For Reference Only-

서문

CJ 시리즈 CJ3M-CPU□□, CJ2H-CPU6□(-EIP) 또는 CJ2M-CPU□□ 프로그래머블 컨트롤러를 구매해 주셔서 감사합니다. 본 매뉴얼에는 CJ3M-CPU□□, CJ2H-CPU6□(-EIP) 또는 CJ2M-CPU□□를 사용하는 데 필요한 정보가 수록되어 있습니다. CJ3M-CPU□□, CJ2H-CPU6□(-EIP) 또는 CJ2M-CPU□□를 사용하기 전에 본 매뉴얼을 충분히 읽고 이해하십시오.

대상 독자

본 매뉴얼은 다음 담당자를 대상으로 하며, 이들은 전기 시스템에 관한 지식(전기 기술자 또는 그에 준하는 지식)을 갖추고 있어야 합니다.

- FA 시스템 설치 담당자
- FA 시스템 설계 담당자.
- FA 시스템 설치 및 접속 담당자
- FA 시스템 및 설비 관리 담당자.

적용 제품

CJ 시리즈 CJ2/CJ3 CPU 유닛

- CJ3M-CPU1□, CJ3M-CPU3□
- CJ2H-CPU6□-EIP, CJ2H-CPU6□
- CJ2M-CPU1□, CJ2M-CPU3□

본 매뉴얼에서는 CPU 유닛을 나타내기 위해 다음 표기를 사용합니다.

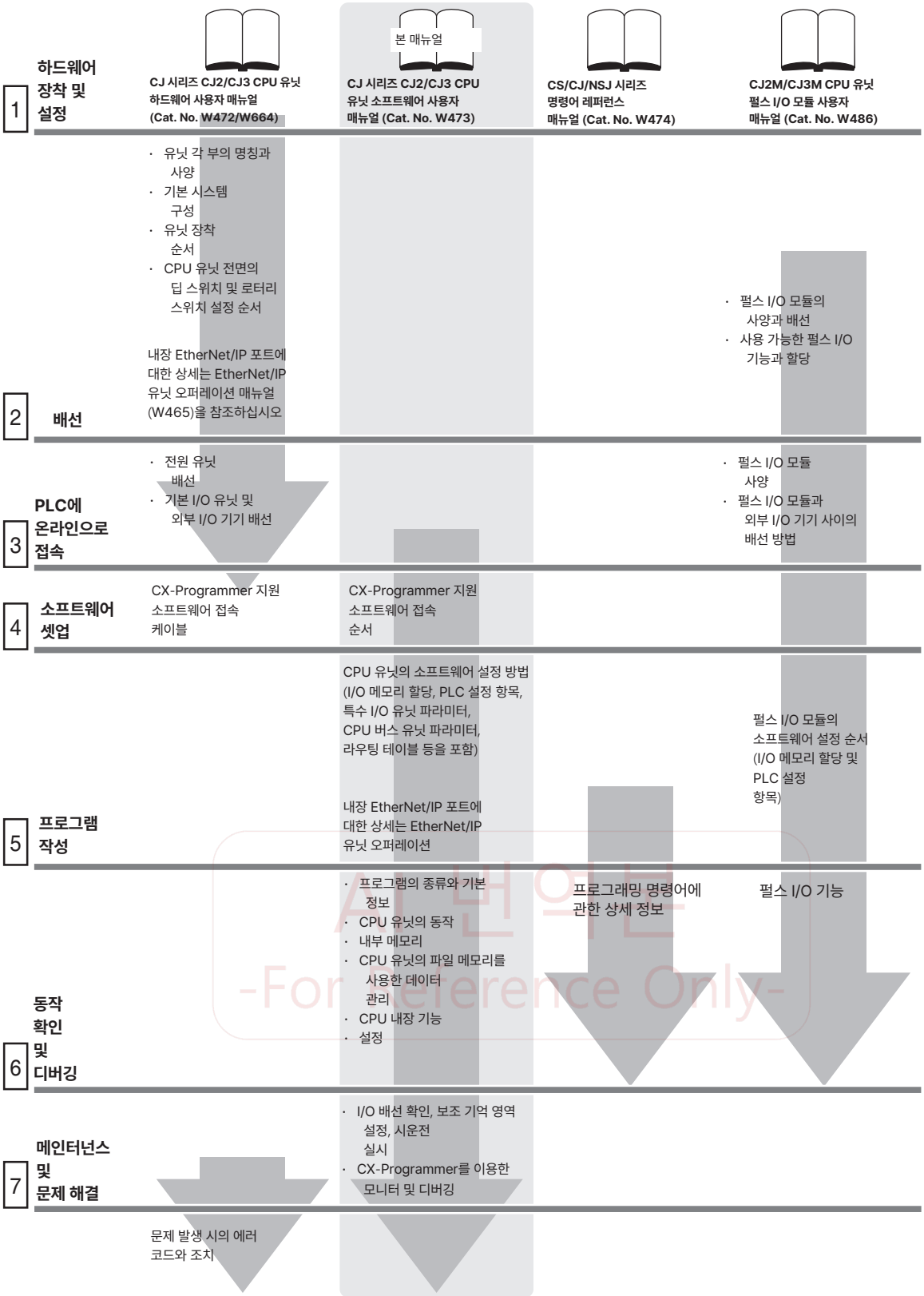
- CJ3M-CPU□□: CJ3M-CPU1□ 및 CJ3M-CPU3□ CPU 유닛을 나타냅니다.
- CJ2H-CPU6□(-EIP): CJ2H-CPU6□-EIP 및 CJ2H-CPU6□ CPU 유닛을 나타냅니다.
- CJ2M-CPU□□: CJ2M-CPU1□ 및 CJ2M-CPU3□ CPU 유닛을 나타냅니다.

옵션인 펄스 I/O 모듈을 사용하면 CJ2M/CJ3M CPU 유닛에서 펄스 I/O를 사용할 수 있습니다.

AI 번역본
-For Reference Only-

CJ2/CJ3 CPU 유닛 매뉴얼

CJ2/CJ3 CPU 유닛에 관한 정보는 다음 매뉴얼에 수록되어 있습니다. 필요한 정보에 따라 해당 매뉴얼을 참조하십시오.



샘플
-For Reference Only-

매뉴얼 구성

CJ2 CPU 매뉴얼은 다음 표에 나열된 섹션으로 구성되어 있습니다. 필요에 따라 매뉴얼의 해당 섹션을 참조하십시오.

CJ3 하드웨어 사용자 매뉴얼 (W664)

섹션	내용
섹션 1 개요	본 섹션에서는 CJ3 CPU 유닛의 개요를 설명하고 특장 및 사양을 설명합니다.
섹션 2 기본 시스템 구성과 디바이스	본 섹션에서는 CJ3 CPU 유닛의 시스템 구성을 설명합니다.
섹션 3 명칭과 기능	본 섹션에서는 CPU 유닛 및 구성 유닛의 각 부 명칭과 기능을 설명합니다.
섹션 4 지원 소프트웨어	본 섹션에서는 프로그래밍 및 디버깅에 사용하는 지원 소프트웨어의 종류와 PLC를 지원 소프트웨어에 접속하는 방법을 설명합니다.
섹션 5 설치	본 섹션에서는 설치 장소와 CPU 유닛 및 구성 유닛의 배선 방법을 설명합니다.
섹션 6 문제 해결	본 섹션에서는 시스템 운전 중에 발생하는 에러의 상태를 확인하는 방법과 그 에러에 대한 조치를 설명합니다.
섹션 7 점검 및 유지보수	본 섹션에서는 정기 점검, 배터리 및 전원 유닛의 수명, 배터리 교체 방법을 설명합니다.
섹션 8 백업 조작	본 섹션에서는 PLC 데이터를 백업하는 순서를 설명합니다.
부록	부록에서는 유닛 외형 치수, 치명 에러 및 비치명 에러의 상세, CPU 유닛의 시리얼 포트 접속에 관한 정보, 컴퓨터에 USB 드라이버를 설치하는 순서, 부하 단락 보호 및 단선 검출에 관한 정보를 제공합니다.

CJ2 하드웨어 사용자 매뉴얼 (W472)

섹션	내용
섹션 1 개요	본 섹션에서는 CJ2 CPU 유닛의 개요를 설명하고 특장 및 사양을 설명합니다.
섹션 2 기본 시스템 구성과 디바이스	본 섹션에서는 CJ2 CPU 유닛의 시스템 구성을 설명합니다.
섹션 3 명칭과 기능	본 섹션에서는 CPU 유닛 및 구성 유닛의 각 부 명칭과 기능을 설명합니다.
섹션 4 지원 소프트웨어	본 섹션에서는 프로그래밍 및 디버깅에 사용하는 지원 소프트웨어의 종류와 PLC를 지원 소프트웨어에 접속하는 방법을 설명합니다.
섹션 5 설치	본 섹션에서는 설치 장소와 CPU 유닛 및 구성 유닛의 배선 방법을 설명합니다.
섹션 6 문제 해결	본 섹션에서는 시스템 운전 중에 발생하는 에러의 상태를 확인하는 방법과 그 에러에 대한 조치를 설명합니다.
섹션 7 점검 및 유지보수	본 섹션에서는 정기 점검, 배터리 및 전원 유닛의 수명, 배터리 교체 방법을 설명합니다.
섹션 8 백업 조작	본 섹션에서는 PLC 데이터를 백업하는 순서를 설명합니다.
부록	부록에서는 유닛 외형 치수, 치명 에러 및 비치명 에러의 상세, CPU 유닛의 시리얼 포트 접속에 관한 정보, 컴퓨터에 USB 드라이버를 설치하는 순서, 부하 단락 보호 및 단선 검출에 관한 정보를 제공합니다.

-For Reference Only-

CJ2/CJ3 소프트웨어 사용자 매뉴얼 (W473) (본 매뉴얼)

섹션	내용
섹션 1 개요	본 섹션에서는 CJ2/CJ3 CPU 유닛의 개요를 설명하고 특장 및 사양을 설명합니다.
섹션 2 CPU 유닛의 내부 메모리	본 섹션에서는 CPU 유닛 내 메모리의 종류와 저장되는 데이터를 설명합니다.
섹션 3 CPU 유닛의 동작	본 섹션에서는 CPU 유닛의 내부 동작을 설명합니다.
섹션 4 CPU 유닛의 초기화	본 섹션에서는 CPU 유닛의 초기 설정을 설명합니다.
섹션 5 프로그래밍의 이해	본 섹션에서는 프로그램의 종류와 심볼, 프로그래밍 명령어 등 프로그래밍의 상세를 설명합니다.
섹션 6 I/O 메모리 영역	본 섹션에서는 CPU 유닛의 I/O 메모리 영역을 설명합니다.
섹션 7 파일 조작	본 섹션에서는 CPU 유닛에 저장할 수 있는 파일, 그 파일의 저장 위치, 파일 조작을 설명합니다.
섹션 8 I/O 할당과 유닛 설정	본 섹션에서는 CPU 유닛과 다른 유닛 사이에서 데이터를 교환하는 데 사용하는 I/O 할당을 설명합니다.
섹션 9 PLC 설정	본 섹션에서는 CPU 유닛의 기본 설정을 수행하는 데 사용하는 PLC 설정 항목의 상세를 설명합니다.
섹션 10 CPU 유닛의 기능	본 섹션에서는 CPU 유닛에 내장된 기능을 설명합니다.
섹션 11 프로그래밍 디바이스와 통신	본 섹션에서는 CJ2 CPU 유닛을 CX-Programmer 등의 지원 소프트웨어 및 다른 기기에 접속하는 순서를 설명합니다.
섹션 12 CPU 유닛의 사이클 타임	본 섹션에서는 사이클 타임을 모니터하고 계산하는 방법을 설명합니다.
부록	부록에서는 프로그래밍 명령어, 실행 시간, 스텝 수, 보조 기억 영역의 워드와 비트, 연속된 PLC 메모리 어드레스의 메모리 맵, 정전 시의 I/O 메모리 동작, CJ 시리즈 PLC와 CS 시리즈 PLC의 비교에 관한 정보를 제공합니다.

명령어 레퍼런스 매뉴얼 (W474)

섹션	내용
섹션 1 명령어의 기본 이해	본 섹션에서는 CS/CJ/NSJ 시리즈 CPU 유닛용 래더 프로그램을 설계하기 위한 기본 정보를 제공합니다.
섹션 2 명령어 목록	본 섹션에서는 CS/CJ/NSJ 시리즈 CPU 유닛에서 사용하는 명령어의 목록을 제공합니다.
섹션 3 명령어	본 섹션에서는 CS/CJ/NSJ 시리즈 CPU 유닛이 지원하는 명령어의 기능, 오퍼랜드, 샘플 프로그램을 설명합니다.
섹션 4 명령어 실행 시간과 스텝 수	본 섹션에서는 CS/CJ/NSJ 시리즈 CPU 유닛의 각 명령어에 대한 명령어 실행 시간을 제공합니다.
부록	부록에서는 CS/CJ/NSJ 시리즈 CPU 유닛의 기능 코드별 · 니모닉별 명령어 목록과 ASCII 표를 제공합니다.

CJ2M/CJ3M 펄스 I/O 모듈 사용자 매뉴얼 (Cat. No. W486)

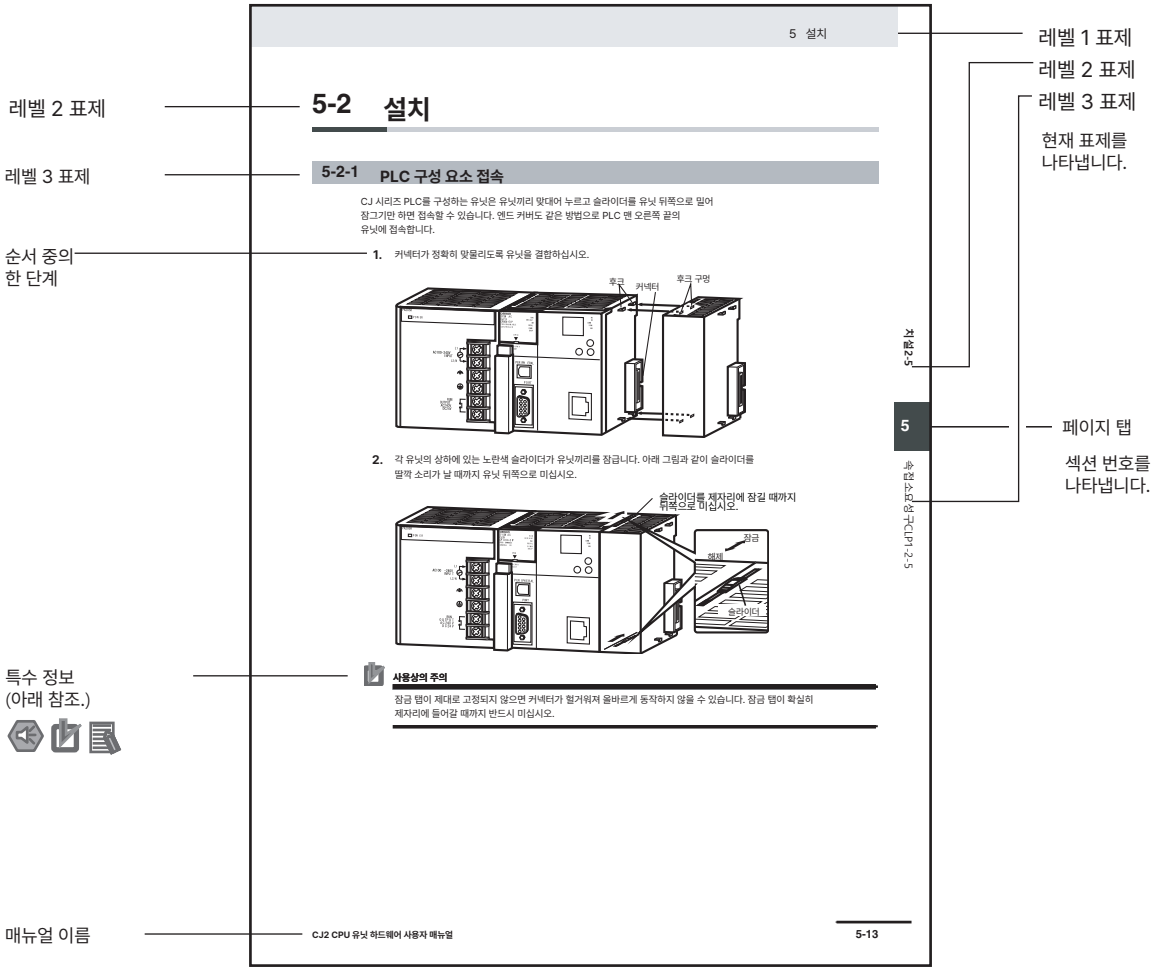
섹션	내용
섹션 1 개요	본 섹션에서는 펄스 I/O 모듈과 CJ2M/CJ3M의 펄스 I/O 기능의 개요를 설명합니다.
섹션 2 I/O 적용 순서와 기능 할당	본 섹션에서는 CJ2M/CJ3M CPU 유닛의 펄스 기능을 나열하고 전체 적용 플로우와 기능 할당을 설명합니다.
섹션 3 펄스 I/O 모듈의 I/O 사양과 배선	본 섹션에서는 펄스 I/O 모듈의 사양을 제공하고 배선을 설명합니다.
섹션 4 범용 I/O	본 섹션에서는 범용 I/O를 설명합니다.
섹션 5 고속 응답 입력	본 섹션에서는 사이클 타임보다 짧은 신호를 입력하는 데 사용할 수 있는 고속 응답 기능을 설명합니다.
섹션 6 인터럽트	본 섹션에서는 인터럽트 입력 기능을 설명합니다.
섹션 7 고속 카운터	본 섹션에서는 고속 카운터 입력과 고속 카운터 인터럽트를 설명합니다.
섹션 8 펄스 출력	본 섹션에서는 사다리꼴 제어, S자 커브 제어, 조깅, 원점 서치 기능 등의 위치 결정 기능을 설명합니다.
섹션 9 PWM 출력	본 섹션에서는 가변 듀티비(PWM) 출력을 설명합니다.
부록	부록에서는 펄스 출력의 플래그 변화표, 다른 기종과의 비교표, 성능표를 제공합니다.

AI 번역본
-For Reference Only-

매뉴얼의 구조

페이지 구성

본 매뉴얼에서는 다음 페이지 구성을 사용합니다.



이 그림은 샘플로 제공된 것이며 본 매뉴얼에 그대로 나타나지 않을 수 있습니다.

샘플
-For Reference Only-

특수 정보

본 매뉴얼의 특수 정보는 다음과 같이 분류됩니다:

안전상의 요점

제품을 안전하게 사용하기 위해 해야 할 일과 하지 말아야 할 일에 관한 주의사항입니다.

사용상의 주의

올바른 동작과 성능을 확보하기 위해 해야 할 일과 하지 말아야 할 일에 관한 주의사항입니다.

참고

이해를 돕거나 조작을 쉽게 하기 위한 참고입니다.

AI 번역본
-For Reference Only-

본 매뉴얼의 섹션



AI 번역본
-For Reference Only-

AI 번역본
-For Reference Only-

목차

서문 1

CJ2/CJ3 CPU 유닛 매뉴얼 2

매뉴얼의 구조 5

본 매뉴얼의 섹션 7

계약 조건 17

안전 주의사항 19

사용 시 주의사항 26

사용 환경 주의사항 31

규격 및 표준 32

CJ2/CJ3 CPU 유닛의 버전 35

관련 매뉴얼 42

섹션 1 개요

1-1 CJ2/CJ3 CPU 유닛 개요 1-2
- 1-1-1 개요 1-2
- 1-1-2 CJ2/CJ3 CPU 유닛의 특징 1-3
- 1-1-3 CJ3M CPU 유닛에서 강화된 기능 1-10

1-2 기본 조작 순서 1-11

섹션 2 CPU 유닛의 내부 메모리

2-1 개요 2-2
- 2-1-1 메모리 구성 2-2
- 2-1-2 메모리 영역과 저장 데이터 2-4
- 2-1-3 프로그래밍 디바이스에서 CPU 유닛으로 데이터 전송 2-5

섹션 3 CPU 유닛의 동작

3-1 CPU 유닛의 내부 동작 3-2
- 3-1-1 개요 3-2
- 3-1-2 사이클 타임 3-4
- 3-1-3 정전 시의 처리 3-7

3-2 CPU 유닛의 동작 모드 3-8
- 3-2-1 동작 모드 3-8
- 3-2-2 동작 모드 확인 3-9
- 3-2-3 동작 모드 변경 3-10
- 3-2-4 동작 모드 상세 3-14

섹션 4 CPU 유닛의 초기화

AI 번역본
-For Reference Only-

4-1 CPU 유닛 초기화 개요 4-2
4-1-1 CPU 유닛의 초기 설정 4-2
4-2 PLC 설정 4-8
4-3 I/O 테이블 작성 4-9
4-3-1 I/O 테이블 4-9
4-3-2 자동 할당 4-10
4-3-3 수동 할당 4-10
4-4 라우팅 테이블 설정 4-11
4-4-1 라우팅 테이블 4-11
4-4-2 라우팅 테이블이 필요한 경우 4-13
4-4-3 라우팅 테이블 설정과 전송 4-14
4-5 특수 I/O 유닛 및 CPU 버스 유닛의 DM 영역 할당 워드 설정 4-15
4-5-1 특수 I/O 유닛 및 CPU 버스 유닛의 DM 영역 할당 워드 설정 4-15
4-5-2 설정 순서 4-15
4-6 CPU 버스 유닛 셋업 영역 4-16
4-6-1 CPU 버스 유닛 셋업 영역 4-16
4-6-2 설정 순서 4-16

섹션 5 프로그래밍의 이해

5-1 프로그래밍 5-3
5-1-1 프로그래밍 개요 5-3
5-1-2 래더 다이어그램의 기본 개념 5-6
5-1-3 ST 언어 5-8
5-1-4 SFC 개요 5-9
5-2 태스크 5-11
5-2-1 태스크 개요 5-11
5-2-2 사이클릭 태스크 5-14
5-2-3 인터럽트 태스크 5-20
5-2-4 태스크 설계 5-30
5-3 섹션 5-40
5-3-1 섹션 개요 5-40
5-4 펑션 블록 5-42
5-4-1 펑션 블록 5-42
5-4-2 펑션 블록의 특징 5-43
5-4-3 펑션 블록 사양 5-44
5-5 심볼 5-48
5-5-1 개요 5-48
5-5-2 심볼의 종류 5-49
5-5-3 글로벌 심볼 5-51
5-5-4 로컬 심볼 5-51
5-5-5 네트워크 심볼 5-52
5-5-6 펑션 블록 내의 변수 5-56
5-5-7 심볼 데이터형 5-57
5-5-8 심볼에 대한 어드레스 자동 할당 5-62
5-6 명령어 5-63
5-6-1 명령어의 기본 이해 5-63
5-6-2 오퍼랜드 지정 5-70
5-6-3 데이터 포맷 5-78
5-6-4 I/O 리프레시 타이밍 5-82
5-7 인덱스 레지스터 5-87
5-7-1 인덱스 레지스터란? 5-87
5-7-2 인덱스 레지스터 사용법 5-87
5-7-3 인덱스 레지스터 관련 처리 5-93
5-7-4 인덱스 레지스터 모니터 5-94
5-7-5 태스크 간 인덱스 레지스터 및 데이터 레지스터 공유 5-95
5-8 어드레스 오프셋 지정 5-97

AI 번역본
-For Reference Only-

5-8-1 개요 ... 5-97
5-8-2 어드레스 오프셋 적용 예 ... 5-99
5-9 프로그램 체크 ... 5-100
5-9-1 CX-Programmer 입력 시의 에러 ... 5-100
5-9-2 CX-Programmer에 의한 프로그램 체크 ... 5-100
5-9-3 시뮬레이터에 의한 디버깅 ... 5-101
5-9-4 프로그램 실행 체크 ... 5-104
5-10 주의사항 ... 5-107
5-10-1 조건 플래그 ... 5-107
5-10-2 특수 프로그램 섹션 ... 5-112

섹션 6 I/O 메모리 영역

6-1 I/O 메모리 영역 ... 6-2
6-1-1 I/O 메모리 영역 개요 ... 6-2
6-1-2 I/O 메모리 영역 구조 ... 6-4
6-1-3 I/O 메모리 값의 유지 ... 6-7
6-2 I/O 영역 ... 6-9
6-2-1 입력 비트 ... 6-9
6-2-2 출력 비트 ... 6-11
6-3 데이터 링크 영역 ... 6-14
6-4 동기 데이터 리프레시 영역 ... 6-15
6-5 CPU 버스 유닛 영역 ... 6-16
6-6 특수 I/O 유닛 영역 ... 6-17
6-7 펄스 I/O 영역 ... 6-18
6-8 시리얼 PLC 링크 영역 ... 6-19
6-9 DeviceNet 영역 ... 6-20
6-10 EtherCAT 리모트 I/O 영역 ... 6-21
6-11 워크 영역 ... 6-22
6-12 유지 영역 ... 6-23
6-13 보조 기억 영역 ... 6-25
6-14 일시 릴레이 영역 ... 6-26
6-15 데이터 메모리 영역 ... 6-27
6-16 확장 데이터 메모리 영역 ... 6-30
6-17 타이머 영역 ... 6-34
6-18 카운터 영역 ... 6-36
6-19 태스크 플래그 ... 6-37
6-20 인덱스 레지스터 ... 6-38
6-21 데이터 레지스터 ... 6-43
6-22 조건 플래그 ... 6-45
6-23 클록 펄스 ... 6-47

AI 번역본
For Reference Only

섹션 7 파일 조작

7-1 파일 메모리 ... 7-2
7-1-1 파일 메모리의 종류 ... 7-2
7-1-2 파일 메모리 초기화 ... 7-3
7-1-3 메모리 카드 주의사항 ... 7-5
7-2 파일 메모리에 저장되는 파일의 종류 ... 7-8

7-2-1 파일 종류 7-8
7-2-2 파일 메모리용 파일의 작성과 저장 7-11

7-3 파일 메모리 조작 7-12
7-3-1 파일 메모리 조작의 종류 7-12
7-3-2 파일 암호화 7-14
7-3-3 파일 메모리 조작 순서와 파일 메모리 파일 7-14
7-3-4 파일 사용상의 제한 7-20
7-3-5 파일 사이즈 7-21
7-3-6 지원 소프트웨어와 파일 메모리 파일의 관계 7-22

섹션 8 I/O 할당과 유닛 설정

8-1 I/O 할당 8-2
8-1-1 I/O 할당 8-2
8-1-2 자동 할당 8-5
8-1-3 수동 할당 8-9
8-1-4 I/O 테이블 에러와 주의사항 8-17

8-2 CPU 버스 유닛 및 특수 I/O 유닛의 설정 8-20
8-2-1 파라미터 설정 8-20
8-2-2 데이터 교환 8-24

섹션 9 PLC 설정

9-1 PLC 설정 개요 9-2

9-2 PLC 설정 항목 목록 9-4

9-3 PLC 설정 항목 설명 9-5
9-3-1 기동 시 동작 설정 9-5
9-3-2 CPU 유닛 설정 9-8
9-3-3 타이밍/동기 설정 9-13
9-3-4 특수 I/O 유닛 사이클릭 리프레시 9-18
9-3-5 기본 I/O 유닛 랙 응답 시간 9-20
9-3-6 시리얼 포트 설정 9-22
9-3-7 주변 서비스 9-29
9-3-8 FINS 보호 9-30
9-3-9 I/O 모듈 9-32

섹션 10 CPU 유닛의 기능

10-1 클록 기능 10-3
10-1-1 클록 기능 10-3
10-1-2 메모리에 저장되는 시각 10-4
10-1-3 프리런 타이머 10-7

10-2 사이클 타임/고속 처리 10-8
10-2-1 최소 사이클 타임 10-8
10-2-2 최대 사이클 타임 10-9
10-2-3 사이클 타임 모니터 10-10
10-2-4 고속 입력 10-10
10-2-5 백그라운드 실행 10-11
10-2-6 고속 인터럽트 기능 10-20

10-3 기동 설정과 메인터넌스 10-23
10-3-1 동작 모드 변경 시 및 기동 시의 유지 설정 10-23
10-3-2 전원 OFF 검출 지연 설정 10-25
10-3-3 전원 OFF 인터럽트 무효화 10-26
10-3-4 RUN 출력 10-27
10-3-5 기동 시 자동 전송 10-28

10-4 유닛 관리 기능 10-37

AI 번역본
For Reference Only

10-4-1 기본 I/O 유닛 관리 ... 10-37
10-4-2 CPU 버스 유닛 플래그/비트 ... 10-39
10-4-3 특수 I/O 유닛 플래그/비트 ... 10-40

10-5 메모리 관리 기능 ... 10-41
10-5-1 자동 백업 ... 10-41
10-5-2 EM 파일 메모리 기능 ... 10-43
10-5-3 코멘트 메모리 ... 10-44
10-5-4 운전 중 전체 프로그램 교체 ... 10-45
10-5-5 전 메모리 클리어 ... 10-52

10-6 보안 기능 ... 10-53
10-6-1 딥 스위치에 의한 쓰기 보호 ... 10-53
10-6-2 패스워드에 의한 읽기 보호 ... 10-53
10-6-3 제조 로트 번호에 의한 프로그램 동작 보호 ... 10-59
10-6-4 FINS 커맨드에 의한 쓰기 보호 ... 10-60
10-6-5 FINS 쓰기 액세스 로그 ... 10-64
10-6-6 조작 로그 기능 ... 10-68
10-6-7 파일 암호화 기능 ... 10-71
10-6-8 PLC 이름 ... 10-77

10-7 디버깅 ... 10-80
10-7-1 강제 세트/리셋 ... 10-80
10-7-2 테스트 입력 ... 10-81
10-7-3 미분 모니터 ... 10-81
10-7-4 온라인 에디팅 ... 10-82
10-7-5 출력 OFF ... 10-84
10-7-6 데이터 트레이스 ... 10-85
10-7-7 에러 시 정지 위치 저장 ... 10-93
10-7-8 고장 알람 명령어 ... 10-94
10-7-9 시스템 에러 시뮬레이션 ... 10-95
10-7-10 고장점 검출 ... 10-96

10-8 동기 유닛 동작 ... 10-98
10-8-1 개요 ... 10-98
10-8-2 동기 유닛 동작 상세 ... 10-102
10-8-3 동기 유닛 동작 사양 ... 10-106
10-8-4 동기 데이터 리프레시 ... 10-107
10-8-5 동기 유닛 동작 사용 시의 제한 ... 10-111
10-8-6 적용 순서 ... 10-113
10-8-7 PLC 설정 ... 10-114
10-8-8 동기 인터럽트 태스크 작성 ... 10-116
10-8-9 동기 유닛 동작의 조정과 문제 해결 ... 10-117

섹션 11 프로그래밍 디바이스와 통신

11-1 CX-Programmer에서 PLC에 액세스 ... 11-2
11-1-1 개요 ... 11-2
11-1-2 액세스 가능한 PLC의 시스템 구성 ... 11-4
11-1-3 CX-Programmer에서 PLC에 액세스 ... 11-9
11-1-4 자동 온라인 접속 ... 11-14

11-2 시리얼 통신 ... 11-18
11-2-1 시리얼 통신 개요 ... 11-18

11-3 통신 네트워크 ... 11-32

섹션 12 CPU 유닛의 사이클 타임

12-1 사이클 타임 모니터 ... 12-2
12-1-1 사이클 타임 모니터 ... 12-2

12-2 사이클 타임 계산 ... 12-4
12-2-1 CPU 유닛 동작 플로우차트 ... 12-4
12-2-2 사이클 타임 개요 ... 12-5

AI 번역본
For Reference Only

12-2-3 유닛별 I/O 유닛 리프레시 시간 12-7
12-2-4 사이클 타임 계산 예 12-11
12-2-5 온라인 에디팅 시 사이클 타임 연장 12-13
12-2-6 I/O 응답 시간 12-13
12-2-7 펄스 I/O 모듈로의 입력 인터럽트 응답 시간 12-14
12-2-8 시리얼 PLC 링크의 응답 성능 12-15

부록

A-1 명령어 기능 A-3
A-1-1 시퀀스 입력 명령어 A-3
A-1-2 시퀀스 출력 명령어 A-5
A-1-3 시퀀스 제어 명령어 A-6
A-1-4 타이머 및 카운터 명령어 A-10
A-1-5 비교 명령어 A-14
A-1-6 데이터 전송 명령어 A-18
A-1-7 데이터 시프트 명령어 A-20
A-1-8 증가/감소 명령어 A-24
A-1-9 심볼 연산 명령어 A-24
A-1-10 변환 명령어 A-29
A-1-11 논리 명령어 A-35
A-1-12 특수 연산 명령어 A-37
A-1-13 부동소수점 연산 명령어 A-38
A-1-14 배정밀도 부동소수점 명령어 A-42
A-1-15 테이블 데이터 처리 명령어 A-45
A-1-16 트래킹 명령어 A-49
A-1-17 데이터 제어 명령어 A-50
A-1-18 서브루틴 명령어 A-54
A-1-19 인터럽트 제어 명령어 A-55
A-1-20 고속 카운터/펄스 출력 명령어 A-56
A-1-21 스텝 명령어 A-58
A-1-22 기본 I/O 유닛 명령어 A-58
A-1-23 시리얼 통신 명령어 A-61
A-1-24 네트워크 명령어 A-63
A-1-25 파일 메모리 명령어 A-65
A-1-26 표시 명령어 A-66
A-1-27 클록 명령어 A-67
A-1-28 디버깅 명령어 A-68
A-1-29 고장 진단 명령어 A-68
A-1-30 기타 명령어 A-69
A-1-31 블록 프로그래밍 명령어 A-70
A-1-32 문자열 처리 명령어 A-74
A-1-33 태스크 제어 명령어 A-77
A-1-34 기종 변환 명령어 A-77
A-1-35 특수 펑션 블록 명령어 A-78

A-2 명령어 실행 시간과 스텝 수 A-80
A-2-1 시퀀스 입력 명령어 A-81
A-2-2 시퀀스 출력 명령어 A-82
A-2-3 시퀀스 제어 명령어 A-82
A-2-4 타이머 및 카운터 명령어 A-83
A-2-5 비교 명령어 A-84
A-2-6 데이터 전송 명령어 A-85
A-2-7 데이터 시프트 명령어 A-86
A-2-8 증가/감소 명령어 A-87
A-2-9 심볼 연산 명령어 A-87
A-2-10 변환 명령어 A-89
A-2-11 논리 명령어 A-91
A-2-12 특수 연산 명령어 A-91
A-2-13 부동소수점 연산 명령어 A-92
A-2-14 배정밀도 부동소수점 명령어 A-93
A-2-15 테이블 데이터 처리 명령어 A-94
A-2-16 트래킹 명령어 A-96

AI 번역본 -Reference Only-

A-2-17 데이터 제어 명령어 A-96
A-2-18 서브루틴 명령어 A-97
A-2-19 인터럽트 제어 명령어 A-98
A-2-20 고속 카운터/펄스 출력 명령어 A-98
A-2-21 스텝 명령어 A-101
A-2-22 기본 I/O 유닛 명령어 A-101
A-2-23 시리얼 통신 명령어 A-102
A-2-24 네트워크 명령어 A-103
A-2-25 파일 메모리 명령어 A-103
A-2-26 표시 명령어 A-104
A-2-27 클록 명령어 A-104
A-2-28 디버깅 명령어 A-104
A-2-29 고장 진단 명령어 A-104
A-2-30 기타 명령어 A-105
A-2-31 블록 프로그래밍 명령어 A-105
A-2-32 문자열 처리 명령어 A-107
A-2-33 태스크 제어 명령어 A-108
A-2-34 기종 변환 명령어 A-108
A-2-35 특수 펑션 블록 명령어 A-109
A-2-36 SFC 명령어 A-109
A-2-37 펑션 블록 인스턴스 실행 시간 A-109

A-3 보조 기억 영역 A-111
A-3-1 읽기 전용 영역(시스템 설정) A-111
A-3-2 읽기/쓰기 영역(사용자 설정) A-153
A-3-3 보조 기억 영역 동작 상세 A-164

A-4 PLC 메모리 어드레스의 메모리 맵 A-167
A-4-1 PLC 메모리 어드레스 A-167
A-4-2 메모리 맵 A-168

A-5 정전 시의 동작 A-169
A-5-1 전원 OFF 시의 동작 A-169
A-5-2 정전 시의 명령어 실행 A-171

A-6 Windows 7 이후 Windows OS에서의 EtherNet/IP 접속 A-174
A-6-1 Windows 방화벽 설정 변경 A-174

A-7 유닛 버전별 지원 기능 A-176
A-7-1 CJ2H CPU 유닛 A-176
A-7-2 CJ2M CPU 유닛 A-179

색인 Index-1

개정 이력 Revision-1

AI 번역본
-For Reference Only-

AI 번역본
-For Reference Only-

계약 조건

보증, 책임의 제한

보증

- **배타적 보증**

  오므론의 배타적 보증은, 오므론의 판매일(또는 오므론이 서면으로 명시한 그 밖의 기간)로부터 12개월 동안 제품에 재료 및 제작상의 결함이 없다는 것입니다. 오므론은 그 밖의 명시적 또는 묵시적 보증을 일체 부인합니다.

- **제한 사항**

  오므론은 제품의 비침해성, 제품성 또는 특정 목적에의 적합성에 관하여 명시적이든 묵시적이든 어떠한 보증이나 진술도 하지 않습니다. 구매자는 제품이 그 의도된 용도의 요구사항을 적합하게 충족할 것임을 구매자 단독으로 판단하였음을 인정합니다.

  오므론은 또한 제품에 의한 지적재산권 침해 또는 그 밖의 침해에 근거한 청구나 비용에 대한 모든 형태의 보증 및 책임을 부인합니다.

- **구매자의 구제 수단**

  오므론이 부담하는 유일한 의무는, 오므론의 선택에 따라 (i) 부적합 제품을 (원래 출하된 형태로, 제거 또는 교체에 따른 인건비는 구매자 부담으로) 교체하거나, (ii) 부적합 제품을 수리하거나, 또는 (iii) 부적합 제품의 구매 가격에 상당하는 금액을 구매자에게 환불 또는 크레딧으로 지급하는 것입니다. 다만 어떠한 경우에도 오므론은, 오므론의 분석 결과 해당 제품이 올바르게 취급, 보관, 설치 및 유지보수되었고 오염, 남용, 오용 또는 부적절한 개조의 대상이 되지 않았음이 확인되지 않는 한, 제품에 관한 보증, 수리, 배상 또는 그 밖의 청구나 비용에 대해 책임을 지지 않습니다. 구매자에 의한 제품 반품은 출하 전에 오므론의 서면 승인을 받아야 합니다. 오므론 관계사는 제품을 그 밖의 전기 또는 전자 부품, 회로, 시스템 어셈블리, 그 밖의 자재나 물질 또는 환경과 조합하여 사용한 경우의 적합성 또는 부적합성이나 그 결과에 대해 책임을 지지 않습니다. 구두 또는 서면으로 제공된 어떠한 조언, 권고 또는 정보도 위 보증에 대한 수정이나 추가로 해석되지 않습니다.

공개된 정보는 http://www.omron.com/global/ 을 참조하거나 오므론 담당자에게 문의하십시오.

AI 번역본
-For Reference Only-

책임의 제한 등

오므론 관계사는 제품과 어떠한 형태로든 관련된 특별 손해, 간접 손해, 부수적 손해 또는 결과적 손해, 이익의 상실, 생산 손실 또는 상업적 손실에 대해, 그 청구가 계약, 보증, 과실 또는 엄격 책임 중 무엇에 근거하든 관계없이 책임을 지지 않습니다.

또한 어떠한 경우에도 오므론 관계사의 책임은 책임이 제기된 해당 제품의 개별 가격을 초과하지 않습니다.

적용상의 고려사항

적합성

오므론 관계사는 구매자의 용도에서의 제품 조합 또는 제품 사용에 적용되는 어떠한 표준, 코드 또는 규정에 대한 적합성에 대해서도 책임을 지지 않습니다. 구매자의 요청이 있는 경우, 오므론은 제품에 적용되는 정격 및 사용 제한 사항을 명시한 해당 제3자 인증 문서를 제공합니다. 이 정보만으로는 최종 제품, 기계, 시스템, 그 밖의 용도나 사용과 조합한 제품의 적합성을 완전히 판단하기에 충분하지 않습니다. 구매자의 용도, 제품 또는 시스템에 대해 해당 제품이 적절한지를 판단하는 것은 전적으로 구매자의 책임입니다. 구매자는 모든 경우에 적용상의 책임을 집니다.

생명이나 재산에 중대한 위험을 수반하는 용도나 대량으로 사용하는 용도에는, 시스템 전체가 그 위험에 대응하도록 설계되어 있고 오므론 제품이 전체 설비 또는 시스템 내에서 의도된 용도에 맞게 올바르게 선정 · 설치되어 있음을 확인하지 않은 채 본 제품을 절대 사용하지 마십시오.

프로그래머블 제품

오므론 관계사는 프로그래머블 제품에 대한 사용자의 프로그래밍이나 그로 인한 결과에 대해 책임을 지지 않습니다.

면책 조항

성능 데이터

오므론 관계사의 웹사이트, 카탈로그, 그 밖의 자료에 제시된 데이터는 사용자가 적합성을 판단하는 데 참고하기 위한 것이며 보증을 구성하지 않습니다. 이는 오므론의 시험 조건에 따른 결과일 수 있으므로 사용자는 실제 적용 요구사항과 대조해야 합니다. 실제 성능은 오므론의 보증 및 책임의 제한을 따릅니다.

사양 변경

제품 사양 및 부속품은 개선 및 그 밖의 사유로 언제든지 변경될 수 있습니다. 공표된 정격이나 기능이 변경되거나 중요한 구조 변경이 있는 경우 형번을 변경하는 것이 당사의 관례입니다. 다만 제품의 일부 사양은 예고 없이 변경될 수 있습니다. 확신이 서지 않는 경우, 귀하의 용도에 필요한 주요 사양을 확정하거나 고정하기 위해 특수 형번이 부여될 수 있습니다. 구매한 제품의 실제 사양을 확인하려면 언제든지 오므론 담당자에게 문의하여 주십시오.

AI 번역부
-For Reference Only-

오류 및 누락

오므론 관계사가 제공하는 정보는 확인을 거쳐 정확하다고 판단되지만, 사무적 오류, 오타 또는 교정상의 오류나 누락에 대해서는 책임을 지지 않습니다.

안전 주의사항

주의 표시의 정의

본 매뉴얼에서는 CJ 시리즈 PLC를 안전하게 사용하는 데 필요한 주의사항을 나타내기 위해 다음 표기를 사용합니다.
제공되는 안전 주의사항은 안전에 매우 중요합니다. 모든 안전 주의사항에 기재된 내용을 반드시 읽고 준수하십시오.

⚠ 경고	피하지 않으면 사망 또는 중상을 초래할 수 있는 잠재적으로 위험한 상황을 나타냅니다. 또한 중대한 물적 손해가 발생할 수 있습니다.
⚠ 주의	피하지 않으면 경상 또는 중간 정도의 부상이나 물적 손해를 초래할 수 있는 잠재적으로 위험한 상황을 나타냅니다.

안전상의 요점
제품을 안전하게 사용하기 위해 해야 할 일과 하지 말아야 할 일에 관한 주의사항을 나타냅니다.

사용상의 주의
올바른 동작과 성능을 확보하기 위해 해야 할 일과 하지 말아야 할 일에 관한 주의사항을 나타냅니다.

AI 번역본
-For Reference Only-

기호



삼각형 기호는 주의사항(경고 포함)을 나타냅니다. 구체적인 내용은 삼각형 안에 표시되고 본문에서 설명합니다. 이 예는 감전에 대한 주의를 나타냅니다.



원과 사선 기호는 해서는 안 되는 조작을 나타냅니다. 구체적인 조작은 원 안에 표시되고 본문에서 설명합니다.



검게 칠한 원 기호는 반드시 해야 하는 조작을 나타냅니다. 구체적인 조작은 원 안에 표시되고 본문에서 설명합니다. 이 예는 반드시 해야 하는 사항에 대한 일반적인 주의를 나타냅니다.



삼각형 기호는 주의사항(경고 포함)을 나타냅니다. 구체적인 내용은 삼각형 안에 표시되고 본문에서 설명합니다. 이 예는 일반적인 주의를 나타냅니다.



삼각형 기호는 주의사항(경고 포함)을 나타냅니다. 구체적인 내용은 삼각형 안에 표시되고 본문에서 설명합니다. 이 예는 고온부에 대한 주의를 나타냅니다.

AI 번역본
-For Reference Only-

⚠ 경고

전원이 공급되는 상태에서 유닛을 분해하거나 유닛 내부를 만지지 마십시오. 감전될 수 있습니다.



전원이 공급되는 상태에서 단자나 단자대를 만지지 마십시오. 감전될 수 있습니다.



프로그래머블 컨트롤러의 오동작이나 프로그래머블 컨트롤러의 동작에 영향을 미치는 그 밖의 외부 요인으로 이상이 발생한 경우에도 시스템의 안전을 확보할 수 있도록, 다음 항목을 포함한 안전 대책을 외부 회로에 (즉, 프로그래머블 컨트롤러 밖에) 마련하십시오. "프로그래머블 컨트롤러"
는 CPU 유닛과 그 밖의 모든 유닛을 가리키며, 본 매뉴얼에서는 "PLC"로 줄여 표기합니다. 그렇게 하지 않으면 중대한 사고가 발생할 수 있습니다.



- PLC는 자기진단 기능이 에러를 검출하거나 중고장 알람(FALS) 명령어가 실행되면 모든 출력을 OFF합니다. 다만 I/O 제어부의 에러, I/O 메모리의 에러, 그 밖에 자기진단 기능으로 검출할 수 없는 에러에 대해서는 예기치 않은 동작이 발생할 수 있습니다. 이러한 모든 에러에 대한 대책으로, 시스템의 안전을 확보하기 위한 외부 안전 대책을 반드시 마련해야 합니다.
- 출력 릴레이의 융착 · 소손이나 출력 트랜지스터의 파손으로 인해 PLC의 출력이 ON 또는 OFF인 상태로 남을 수 있습니다. 이러한 문제에 대한 대책으로, 시스템의 안전을 확보하기 위한 외부 안전 대책을 반드시 마련해야 합니다.
- 데이터 링크 통신이나 리모트 I/O 통신에서 통신 에러나 오동작이 발생하더라도 시스템 전체의 안전을 확보할 수 있도록 컴퓨터 시스템과 프로그래밍에 대책을 마련하십시오.
- 데이터 링크 통신이나 리모트 I/O 통신에서 통신 에러나 오동작이 발생하더라도 시스템 전체의 안전을 확보할 수 있도록 컴퓨터 시스템과 프로그래밍에 대책을 마련하십시오.

사 내 참 고 용
-For Reference Only-

주변 툴을 사용하여 파일 메모리(메모리 카드 또는 EM 파일 메모리)에 저장된 데이터 파일을 CPU 유닛의 I/O 영역(CIO)으로 전송하기 전에 안전을 확인하십시오. 그렇지 않으면 CPU 유닛의 동작 모드와 관계없이 출력 유닛에 접속된 기기가 오동작할 수 있습니다.



신호선 단선, 순간 정전, 그 밖의 원인으로 신호가 잘못되거나 누락되거나 이상해진 경우에도 안전을 확보할 수 있도록 고객 측에서 페일세이프 대책을 강구해야 합니다. 적절한 대책을 마련하지 않으면 이상 동작으로 인해 중대한 사고가 발생할 수 있습니다.



PLC에 규정 범위를 벗어난 전압이나 전류를 인가하지 마십시오.
그렇게 하면 오동작이나 화재의 원인이 될 수 있습니다.



⚠ 경고

바이러스 방지

제어 시스템에 접속된 컴퓨터에는 최신 상용 수준의 안티바이러스 소프트웨어를 설치하고 항상 최신 상태로 유지하십시오.



무단 액세스를 방지하기 위한 보안 대책



당사 제품에 대한 무단 액세스를 방지하기 위해 다음 대책을 강구하십시오.

- 허가된 담당자만 제어 시스템 및 설비에 접근할 수 있도록 물리적 관리 수단을 마련하십시오.
- 신뢰할 수 없는 기기로부터의 액세스를 방지하기 위해 네트워크를 통한 제어 시스템 및 설비로의 접속을 줄이십시오.
- 사용하지 않는 통신 포트를 차단하고 통신 호스트를 제한하며 제어 시스템 및 설비를 IT 네트워크에서 분리하기 위해 방화벽을 설치하십시오.
- 제어 시스템 및 설비에 원격 액세스할 때는 가상 사설망(VPN)을 사용하십시오.
- 제어 시스템 및 설비에 원격 액세스하는 기기에는 다요소 인증을 도입하십시오.
- 강력한 패스워드를 설정하고 자주 변경하십시오.
- USB 드라이브나 그 밖의 외부 저장 장치를 제어 시스템 및 설비에 접속하기 전에 바이러스 검사로 안전을 확인하십시오.

데이터 입출력 보호



제어 시스템 및 설비로의 입출력 데이터가 의도치 않게 변경되는 것에 대응할 수 있도록 백업과 범위를 검증하십시오.

- 데이터 범위 확인
- 백업의 유효성 확인 및 변조 · 이상 발생에 대비한 복원용 데이터 준비
- 데이터 변조 · 이상 발생 시의 비상 정지, 페일소프트 동작 등의 안전 설계

데이터 복구



데이터 손실에 대비하여 정기적으로 데이터를 백업하고 최신 상태로 유지하십시오.

AI 번역본
For Reference Only



글로벌 주소를 통해 인트라넷 환경을 사용하는 경우, SCADA나 HMI 등의 무단 단말기나 무단 서버에 접속하면 스푸핑이나 변조 같은 네트워크 보안 문제가 발생할 수 있습니다.

단말기에 대한 액세스 제한, 시큐어 기능을 갖춘 단말기 사용, 설치 구역 잠금 등 충분한 대책을 고객 측에서 직접 강구해야 합니다.

⚠ 경고

인트라넷을 구축할 때 케이블 단선이나 무단 네트워크 기기의 영향으로
통신 장애가 발생할 수 있습니다.

설치 구역 잠금 등의 수단으로 네트워크 기기에 대한 물리적 접근을 제한하는 등
적절한 대책을 강구하십시오.



SD 메모리 카드 기능을 갖춘 기기를 사용하는 경우, 제3자가 리무버블 미디어를
분리하거나 언마운트하여 리무버블 미디어 내의 파일과 데이터를
취득 · 변경 · 교체할 수 있는 보안 위험이 있습니다.



설치 구역 잠금, 출입 관리 등의 수단으로 컨트롤러에 대한 물리적 접근을 제한하거나
리무버블 미디어에 대한 적절한 관리 대책을 강구하는 등 충분한 대책을 고객 측에서
직접 강구해 주십시오.

AI 번역본
-For Reference Only-

⚠ 주의

사이클 타임이 연장되어도 악영향이 없음을 확인한 후에 온라인 에디팅을 실행하십시오. 그렇지 않으면 입력 신호를 읽지 못할 수 있습니다.



프로그램, PLC 설정, I/O 테이블, I/O 메모리 내용, 파라미터를 다른 노드로 전송하거나 이들 항목의 내용을 변경하기 전에 전송처 노드의 안전을 확인하십시오. 데이터를 전송하거나 변경하면 예기치 않은 시스템 동작이 발생할 수 있습니다.



CJ2 CPU 유닛은 사용자 프로그램과 파라미터 데이터가 CPU 유닛에 쓰일 때 이를 플래시 메모리에 자동으로 백업합니다. 다만 DM, EM, HR, AR 영역을 포함한 I/O 메모리)는 플래시 메모리에 쓰이지 않습니다.

DM, EM, HR, AR 영역은 배터리로 정전 중에도 유지할 수 있습니다. (AR 영역의 일부 어드레스 내용도 유지됩니다.) 배터리 에러가 있으면 정전 후 이들 영역의 내용이 정확하지 않을 수 있습니다. DM, EM, HR, AR 영역의 내용을 외부 출력 제어에 사용하는 경우, 배터리 에러 플래그(A402.04)가 ON일 때는 부적절한 출력이 나가지 않도록 하십시오.



AC 전원 유닛의 단자 나사는 오퍼레이션 매뉴얼에 규정된 토크로 조이십시오. 나사가 헐거우면 소손이나 오동작의 원인이 될 수 있습니다.



전원이 공급되는 중이거나 전원을 OFF한 직후에는 전원 유닛을 만지지 마십시오. 전원 유닛이 뜨거워 화상을 입을 수 있습니다.

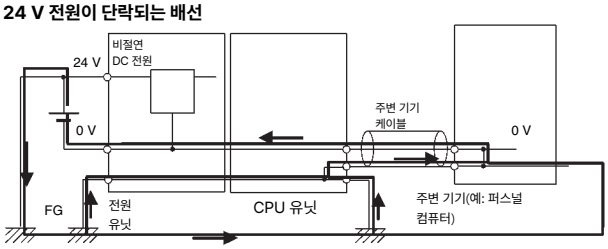


비절연 전원 유닛(CJ1W-PD022)이 장착된 PLC에 퍼스널 컴퓨터나 그 밖의 주변 기기를 접속할 때는, 외부 전원의 0 V 측을 접지하거나 외부 전원을 전혀 접지하지 마십시오 접지. 잘못된 접지 방법을 사용하면 외부 전원에 단락이 발생합니다. 아래 그림과 같이 24 V 측은 절대 접지하지 마십시오.



-For Reference Only-

24 V 전원이 단락되는 배선



⚠ 주의

DC 전원을 배선할 때는 극성(+/-)에 충분히 주의하십시오.
잘못 접속하면 시스템이 오동작할 수 있습니다.



IOWR 명령어를 사용하여 프로그래밍(또는 재프로그래밍)한 후에는, 실제 운전을 시작하기 전에 새 래더 프로그램과 데이터로 올바른 동작이 가능한지 확인하십시오. 이상이 있으면 제품이 동작을 정지하여 기계나 설비가 예기치 않게 동작할 수 있습니다.



AI 번역본
-For Reference Only-

사용 시 주의사항

CJ 시리즈 PLC를 사용할 때는 다음 주의사항을 지키십시오.

● 전원

- 반드시 사용자 매뉴얼에 규정된 전원 전압을 사용하십시오. 전압이 잘못되면 오동작이나 소손의 원인이 될 수 있습니다.
- 전원 유닛의 용량을 초과하면 CPU 유닛이나 그 밖의 유닛이 기동하지 않을 수 있습니다.
- 정격 전압 및 주파수의 규정 전원이 공급되도록 적절한 대책을 강구하십시오. 전원이 불안정한 장소에서는 특히 주의하십시오. 전원이 잘못되면 오동작의 원인이 될 수 있습니다.
- 다음 작업을 하기 전에는 반드시 PLC의 전원을 OFF하십시오. 전원을 OFF하지 않으면 오동작이나 감전의 원인이 될 수 있습니다.
  - 전원 유닛, I/O 유닛, CPU 유닛, 옵션 보드, 펄스 I/O 모듈, 그 밖의 유닛을 장착 또는 분리하는 작업.
  - 유닛을 조립하는 작업.
  - 딥 스위치나 로터리 스위치를 설정하는 작업.
  - 케이블을 접속하거나 시스템을 배선하는 작업.
  - 커넥터를 접속하거나 분리하는 작업.
- 단자를 건너 배선할 때는 모든 단자의 합계 전류가 전선에 흐릅니다. 전선의 허용 전류 용량이 충분한지 확인하십시오.
- 교체 알림 기능을 지원하는 전원 유닛을 사용할 때는 다음 주의사항을 지키십시오.
  - 전원 유닛 전면의 표시가 0.0과 A02를 번갈아 표시하거나 알람 출력이 자동으로 OFF되면 6개월 이내에 전원 유닛을 교체하십시오.
  - 알람 출력 케이블은 전력선 및 고압선과 분리하여 배선하십시오.
  - 알람 출력에 사양을 초과하는 전압을 인가하거나 부하를 접속하지 마십시오.
  - 전원 유닛을 3개월 이상 보관할 때는 교체 알림 기능을 유지하기 위해 −20~30°C, 습도 25%~70%에서 보관하십시오.
  - 전원 유닛을 올바르게 설치하지 않으면 열이 축적되어 교체 알림 신호가 잘못된 시점에 나타나거나 내부 소자가 열화 또는 손상될 수 있습니다. 반드시 표준 설치 방법만 사용하십시오.
- 전원을 OFF한 직후에는 전원 유닛의 단자를 만지지 마십시오. 잔류 전압으로 감전될 수 있습니다.
- 컴퓨터를 PLC에 접속하는 경우 전위차로 인한 고장을 방지하기 위해 다음 주의사항을 지키십시오.
  - 노트북 컴퓨터를 PLC에 접속하기 전에 컴퓨터의 전원 플러그를 AC 콘센트에서 분리하십시오. AC 어댑터의 잔류 전류로 인해 컴퓨터와 PLC 사이에 전위차가 발생할 수 있습니다. 컴퓨터와 PLC를 접속한 후에 AC 어댑터에서 다시 전원을 공급하십시오.
  - 컴퓨터에 FG 단자가 있는 경우, PLC의 FG(GR) 단자와 같은 전위가 되도록 접속하십시오.
- 컴퓨터를 별도의 장소에 접지하는 경우, 접지 조건에 따라 전위차가 발생할 수 있습니다.
- 내전압 시험 중에 스위치로 전전압을 인가하거나 차단하면 그로 인한 임펄스 전압이 전원 유닛을 손상시킬 수 있습니다. 시험 장비의 전압 조절 노브를 사용하여 인가 전압을 서서히 조정하십시오.
- 유닛의 개구부에 이물질이 들어가지 않도록 하십시오. 이물질이 들어가면 유닛의 소손, 감전, 고장의 원인이 될 수 있습니다.

샘플 번역본

● 설치

- 유닛을 만지기 전에 반드시 접지된 금속 물체를 먼저 만져 축적된 정전기를 방전시키십시오. 그렇게 하지 않으면 오동작이나 손상의 원인이 될 수 있습니다.
- 단자대, 커넥터, 메모리 카드, 옵션 보드, 펄스 I/O 모듈, 증설 케이블 등 잠금 장치가 있는 부품은 반드시 제자리에 확실히 잠겼는지 확인하십시오.
- 전원 유닛, CPU 유닛, I/O 유닛, 특수 I/O 유닛, CPU 버스 유닛, 펄스 I/O 모듈의 상하에 있는 슬라이더는 인접 유닛과 접속한 후 (딸깍 소리가 날 때까지) 완전히 잠가야 합니다. 슬라이더가 제자리에 잠기지 않으면 유닛이 올바르게 동작하지 않을 수 있습니다. 슬라이더가 잠기지 않으면 올바른 기능을 얻지 못할 수 있습니다.
- 배선 부스러기, 절삭 가루, 그 밖의 이물질이 유닛 안에 들어가지 않도록 하십시오. 이물질이 들어가면 유닛의 소손, 고장, 오동작이 발생할 수 있습니다. 특히 배선 작업 중에는 유닛을 덮는 등 적절한 대책을 강구하십시오.

● 배선

- 본 매뉴얼의 지시에 따라 올바르게 배선하십시오.
- 전원을 ON하기 전에 모든 배선과 스위치 설정을 재차 확인하십시오. 배선이 잘못되면 소손의 원인이 될 수 있습니다.
- 모든 단자 나사와 케이블 커넥터 나사는 반드시 해당 매뉴얼에 규정된 토크로 조이십시오. 조임 토크가 잘못되면 오동작의 원인이 될 수 있습니다.
- 단자대와 커넥터는 장착 위치를 충분히 확인한 후에 장착하십시오.
- 배선할 때는 유닛에 라벨을 부착한 채로 두십시오. 라벨을 떼면 이물질이 유닛에 들어가 오동작의 원인이 될 수 있습니다.
- 배선이 끝나면 적절한 방열을 위해 라벨을 떼어 내십시오. 라벨을 붙인 채로 두면 오동작의 원인이 될 수 있습니다.
- 배선에는 압착 단자를 사용하십시오. 연선을 벗긴 채로 단자에 직접 접속하지 마십시오. 벗긴 연선을 접속하면 소손의 원인이 될 수 있습니다.
- 입력 유닛에 정격 입력 전압을 초과하는 전압을 인가하지 마십시오. 전압이 초과되면 소손의 원인이 될 수 있습니다.
- 유닛을 설치할 때는 반드시 100 Ω 이하로 접지하십시오. 100 Ω 이하로 접지하지 않으면 감전의 원인이 될 수 있습니다.
  전원 유닛의 GR 단자와 LG 단자를 단락할 때는 반드시 100 Ω 이하의 접지를 해야 합니다.
- 출력 유닛에 최대 개폐 용량을 초과하는 전압을 인가하거나 부하를 접속하지 마십시오. 전압이나 부하가 초과되면 소손의 원인이 될 수 있습니다.
- 케이블을 잡아당기거나 자연스러운 한계를 넘어 구부리지 마십시오. 그렇게 하면 케이블이 끊어질 수 있습니다.
- 케이블이나 그 밖의 배선 위에 물건을 올려놓지 마십시오. 그렇게 하면 케이블이 끊어질 수 있습니다.
- 시판되는 RS-232C 퍼스널 컴퓨터 케이블을 사용하지 마십시오. 반드시 본 매뉴얼에 기재된 전용 케이블을 사용하거나 매뉴얼의 사양에 따라 케이블을 제작하십시오. 시판 케이블을 사용하면 외부 기기나 CPU 유닛이 손상될 수 있습니다.
- CPU 유닛의 RS-232C 포트에 있는 6번 핀(5 V 전원)은 NT-AL001 링크 어댑터, CJ1W-CIF11 컨버터, 프로그래머블 터미널(NV3W-M□20L) 이외의 기기에는 절대 접속하지 마십시오. 외부 기기나 CPU 유닛이 손상될 수 있습니다.

AI 번역본
-For Reference Only-

● 취급

- 절연 내력 시험의 전전압을 스위치로 갑자기 인가하거나 차단하면 전원 유닛이 손상될 수 있습니다. 가변 저항기를 사용하여 전압을 서서히 올리고 내리십시오.
- 내전압 시험이나 절연 저항 시험을 하기 전에 전원 유닛의 라인 그라운드 단자(LG)와 기능 접지 단자(GR)를 분리하십시오. 그렇게 하지 않으면 소손의 원인이 될 수 있습니다.
- 운전을 시작하기 전에 딥 스위치와 DM 영역이 올바르게 설정되었는지 확인하십시오.

· CPU 유닛, 특수 I/O 유닛, CPU 버스 유닛을 교체한 후에는 운전을 재개하기 전에 DM 영역, 유지 영역, 그 밖의 메모리 영역에 필요한 데이터가 새 유닛에 전송되었는지 확인하십시오.

· 다음 작업을 하기 전에 시스템에 악영향이 없음을 확인하십시오. 확인하지 않으면 예기치 않은 동작이 발생할 수 있습니다.
  · PLC의 동작 모드 변경(기동 시 동작 모드 설정 포함).
  · 메모리 내 비트의 강제 세트/강제 리셋.
  · 메모리 내 워드의 현재값이나 설정값 변경.
  · EtherNet/IP 유닛이나 내장 EtherNet/IP 포트의 설정(IP 주소 또는 태그 데이터 링크 설정)을 프로그래밍 디바이스에서 전송하면, 전송된 설정을 유효하게 하기 위해 전송처의 모든 EtherNet/IP 유닛 또는 내장 EtherNet/IP 포트(노드)가 리셋됩니다. 유닛을 재기동해도 시스템에 문제가 없음을 확인한 후에만 EtherNet/IP 유닛이나 내장 EtherNet/IP 포트에 설정을 전송하십시오.

· 유닛을 분해, 수리, 개조하려고 하지 마십시오. 그렇게 하면 오동작, 화재, 감전의 원인이 될 수 있습니다.

· PLC를 떨어뜨리거나 이상한 진동이나 충격을 가하지 마십시오.
· 배터리를 장착하지 않고 전원도 공급하지 않은 상태로 PLC를 일정 기간 방치한 후 전원을 ON하지 않은 채 배터리를 장착하면 배터리 수명이 짧아집니다.

· 배터리 에러가 발생하거나 규정된 배터리 백업 시간이 지나면 즉시 배터리를 교체하십시오. 교체용 배터리는 반드시 배터리 라벨에 표시된 제조일로부터 2년 이내의 것을 장착하십시오.

· 배터리를 교체하기 전에, 교체 순서를 시작하기 전 최소 5분 동안 전원을 ON하고 전원을 OFF한 후 5분 이내에 배터리 교체를 완료하십시오. 메모리 내용이 손상되거나 그렇게 하지
않으면 시계가 멈출 수 있습니다. 이 주의사항을 지키지 않으면.

· 애플리케이션 프로그래밍에 배터리 에러 플래그를 사용하는 경우, 전원이 ON인 상태에서 배터리를 교체하기 전에 시스템이 배터리 에러를 검출하더라도 시스템 안전을 확인하십시오.
· 배터리 단자를 단락시키거나 배터리를 충전, 분해, 가열, 소각하지 마십시오. 배터리에 강한 충격을 가하지 마십시오. 그렇게 하면 배터리의 누액, 파열, 발열, 발화의 원인이 될 수 있습니다. 바닥에 떨어뜨렸거나 그 밖에 과도한 충격을 받은 배터리는 폐기하십시오. 충격을 받은 배터리는 사용하면 누액이 발생할 수 있습니다.

· UL 규격에서는 숙련된 기술자만 배터리를 교체할 수 있도록 요구하고 있습니다. 반드시 숙련된 기술자가 배터리 교체를 담당하도록 하십시오. 본 매뉴얼에 기재된 배터리 교체 순서를 따르십시오.

· 제품과 배터리는 해당 지역의 조례에 따라 폐기하십시오.



廢電池請回收

AI 번역본
For Reference Only

· 과염소산염 함유량이 6 ppb 이상인 리튬 1차 전지를 포함한 제품을 미국 캘리포니아주로 수출하거나 캘리포니아주를 경유하여 출하하는 경우, 모든 제품에 다음 주의 표시를 해야 합니다.
Perchlorate Material–special handling may apply.
http://www.dtsc.ca.gov/hazardouswaste/perchlorate 참조
· 배터리(CJ1W-BAT01)는 과염소산염 함유량이 6 ppb 이상인 리튬 1차 전지를 포함합니다. CJ3M-CPU□□가 포함된 제품을 미국 캘리포니아주로 수출하거나 캘리포니아주를 경유하여 출하하는 경우, 모든 포장 및 출하 용기에 적절히 표시하십시오.
· I/O 홀드 비트를 ON하면 PLC를 RUN 또는 MONITOR 모드에서 PROGRAM 모드로 전환해도 PLC의 출력이 OFF되지 않고 이전 상태를 유지합니다. 이때 외부 부하가 위험한 상태를 만들지 않도록 하십시오. (FALS(007) 명령어로 발생한 것을 포함하여 치명 에러로 운전이 정지한 경우에는 출력 유닛의 모든 출력이 OFF되고 내부 출력 상태만 유지됩니다.)

- 부적절한 데이터 링크 테이블이나 파라미터를 설정하면 예기치 않은 동작이 발생할 수 있습니다. 적절한 데이터 링크 테이블과 파라미터를 설정한 경우에도, 데이터 링크를 개시하거나 정지하기 전에 제어 대상 시스템에 악영향이 없음을 확인하십시오.
- 데이터 링크 통신으로 수신한 데이터는 그 데이터의 송신원 CPU 유닛에 에러가 없는 경우에만 사용되도록 프로그램을 작성하십시오. 송신원 CPU 유닛의 에러는 스테이터스 플래그의 CPU 유닛 에러 정보로 확인하십시오. 송신원 CPU 유닛에 에러가 있으면 잘못된 데이터를 송신할 수 있습니다.
- 프로그래밍 디바이스에서 CPU 유닛으로 라우팅 테이블을 전송하면 모든 CPU 버스 유닛이 재기동됩니다. 새 라우팅 테이블을 읽어 유효하게 하려면 이들 유닛의 재기동이 필요합니다. 라우팅 테이블을 전송하기 전에 시스템에 악영향이 없음을 확인하십시오.
- PLC 운전 중에 태그 데이터 링크 파라미터를 전송하는 동안에는 관련 노드 사이의 태그 데이터 링크가 정지합니다. 태그 데이터 링크 파라미터를 전송하기 전에 시스템에 악영향이 없음을 확인하십시오.
- 네트워크 통신에 방해가 있으면 출력 상태는 사용 중인 기기에 따라 달라집니다. 출력이 있는 기기를 사용할 때는 통신에 방해가 있을 때 발생하는 동작을 확인하고 필요에 따라 안전 대책을 실시하십시오.
- 기동 시 자동으로 데이터를 전송하기 위해 프로그래밍 디바이스(프로그래밍 콘솔 또는 CX-Programmer)에서 AUTOEXEC.IOM 파일을 작성할 때는, 최초 쓰기 어드레스를 D20000으로 설정하고 쓰는 데이터의 사이즈가 DM 영역의 사이즈를 초과하지 않도록 하십시오. 기동 시 메모리 카드에서 데이터 파일을 읽으면, AUTOEXEC.IOM 파일 작성 시 다른 어드레스를 설정했더라도 데이터는 CPU 유닛의 D20000부터 쓰입니다. 또한 DM 영역을 초과하면 (CX-Programmer를 사용하는 경우 발생할 수 있습니다) 나머지 데이터는 EM 영역에 쓰입니다.
- CJ2/CJ3 CPU 유닛의 사용자 프로그램과 파라미터 영역 데이터는 내장 플래시 메모리에 백업됩니다. 백업 동작이 진행 중일 때는 CPU 유닛 전면의 BKUP 표시등이 점등합니다. BKUP 표시등이 점등 중일 때는 CPU 유닛의 전원을 OFF하지 마십시오. 전원을 OFF하면 데이터가 백업되지 않습니다.
- 사용자 프로그램과 유닛 파라미터 설정은 실제로 유닛에서 실행하기 전에 올바르게 실행되는지 확인하십시오. 프로그램과 파라미터 설정을 확인하지 않으면 예기치 않은 동작이 발생할 수 있습니다.
- I/O 테이블에 특수 I/O 유닛이나 CPU 버스 유닛을 설정할 때는 유닛을 재기동하기 전에 접속 대상 기기의 안전을 충분히 확인하십시오.
- 메모리 카드를 읽거나 쓰는 중에는 PLC의 전원을 OFF하지 마십시오. 또한 BUSY 표시등이 점등 중일 때는 메모리 카드를 빼지 마십시오. 그렇게 하면 메모리 카드를 사용할 수 없게 될 수 있습니다.
  메모리 카드를 뺄 때는 먼저 메모리 카드 전원 스위치를 누르고 BUSY 표시등이 꺼진 것을 확인한 후에 메모리 카드를 빼십시오.
- 데이터를 복원할 때는 복원 조작을 실행하기 전에 선택한 데이터가 복원할 올바른 데이터인지 충분히 확인하십시오. 선택한 데이터의 내용에 따라서는 제어 시스템이 예기치 않게 동작할 수 있습니다.
- 일부 특수 I/O 유닛과 CPU 버스 유닛은 CPU 유닛에 저장된 파라미터(예: DM 영역에 할당된 워드, 데이터 링크 테이블, 이더넷 설정)로 동작합니다. 선택한 CPU 버스 유닛이나 특수 I/O 유닛에 제한이 있는 경우, 제한에 관한 정보가 PLC Backup Tool의 정보 영역에 표시됩니다. 제한을 확인한 후, 데이터를 백업하거나 복원할 때는 반드시 CPU 유닛과 해당 CPU 버스 유닛 또는 특수 I/O 유닛을 함께 선택하십시오. 두 유닛을 함께 선택하지 않은 채 백업 또는 복원한 데이터로 설비를 기동하면 제어 시스템이 예기치 않게 동작할 수 있습니다.
- 저장할 데이터에 백업에 제한이 있는 유닛이 포함된 경우, 제한에 관한 정보가 PLC Backup Tool의 정보 영역에 표시됩니다. 제한에 관한 정보를 확인하고 필요한 대책을 강구하십시오. 데이터를 복원한 후 설비를 운전하면 제어 시스템이 예기치 않게 동작할 수 있습니다
- PLC 운전 중에 데이터를 복원하기 전에는 PLC 운전이 정지해도 문제가 없는지 확인하십시오. PLC가 예기치 않은 시점에 정지하면 제어 시스템이 예기치 않게 동작할 수 있습니다.

-For Reference Only-

· 데이터를 복원한 후에는 반드시 PLC 전원을 OFF했다가 다시 ON하십시오. 전원을 재투입하지 않으면 복원한 데이터가 시스템에 반영되지 않아 제어 시스템이 예기치 않게 동작할 수 있습니다.

· 강제 상태 데이터는 백업할 수 있지만 복원할 수는 없습니다. 강제 상태가 포함된 데이터를 복원한 후 운전을 시작하기 전에 필요에 따라 CX-Programmer에서 비트를 강제 세트 또는 강제 리셋하는 순서를 실시하십시오. 강제 상태의 차이에 따라서는 제어 시스템이 예기치 않게 동작할 수 있습니다.

· 래더 또는 ST 프로그래밍에서 배열 변수의 첨자에 심볼이나 메모리 어드레스(ST 프로그래밍에서는 심볼만 허용)를 지정하는 경우, 지정한 요소 번호가 메모리 영역의 최대 범위를 초과하지 않도록 하십시오.
심볼에 지정된 메모리 영역의 최대 범위를 초과하는 요소 번호를 지정하면 다른 메모리 영역의 데이터에 액세스하게 되어 예기치 않은 동작이 발생할 수 있습니다.

· 래더 다이어그램에서 오프셋에 심볼이나 어드레스를 지정하는 경우, 워드 어드레스나 심볼을 사용하여 오프셋을 간접 지정할 때 선두 어드레스의 메모리 영역을 초과하지 않도록 프로그래밍하십시오.
간접 지정으로 어드레스가 선두 어드레스의 영역을 초과하면 시스템이 다른 영역의 데이터에 액세스하여 예기치 않은 동작이 발생할 수 있습니다.

· USB 포트에는 퍼스널 컴퓨터 이외의 기기를 접속하지 마십시오.

· CJ3M CPU 유닛에는 배터리가 없습니다.

  · 사용자 프로그램에서 시계 데이터를 사용하고 전원을 장시간 OFF하는 경우에는 별매의 배터리(CJ1W-BAT01)를 사용하십시오. 공장 출하 시 설정에서는 배터리가 장착되어 있지 않습니다. 시계 데이터는 내장 커패시터로 유지됩니다. 커패시터는 전원이 공급되는 동안 충전되고 전원이 공급되지 않는 동안 방전됩니다. 전원 OFF 시간의 비율이 높으면 시계 데이터가 초기화되어 시계 데이터를 사용하는 프로그램에서 오동작이 발생할 수 있습니다.

  · 배터리를 사용하는 경우에는 CX-Programmer를 접속하여 배터리 관련 에러 검출을 "Detect"로 설정하십시오. 배터리가 내장되어 있지 않으므로 공장 출하 시 설정에서는 배터리 관련 에러 검출이 Do not use로 설정되어 있습니다. 배터리 관련 에러 검출이 "Not Detect"로 설정되어 있으면 배터리가 소모되어도 에러를 검출할 수 없어 시계 데이터를 사용하는 프로그램에서 오동작이 발생할 수 있습니다.

· 래더 프로그램에서 BIT COUNTER (BCNT), BLOCK SET (BSET), BLOCK TRANSFER(XFER)를 사용할 때는 각 명령어에 99 워드를 초과하여 지정하지 마십시오. 99 워드를 초과하여 사용해야 하는 경우에는 명령어를 둘 이상 사용하십시오. 이들 명령어의 실행 중에는 펄스 출력을 할 수 없습니다. 이 중 하나에 99 워드를 초과하여 지정하면 펄스 출력을 예측할 수 없게 되어 일시적으로 정지할 수 있습니다.

· 내장 EtherNet/IP를 리셋하기 전에 리셋을 실시해도 설비에 문제가 발생하지 않는지 확인하십시오.

AI 번역본
-For Reference Only-

● 외부 회로

· 제어 시스템의 전원을 ON하기 전에 반드시 PLC의 전원을 먼저 ON하십시오. 제어 전원보다 나중에 PLC 전원을 ON하면, PLC에 전원이 투입될 때 DC 출력 유닛 등의 출력 단자가 순간적으로 ON되므로 제어 시스템 신호에 일시적인 에러가 발생할 수 있습니다.

· 외부 배선의 단락에 대비하여 외부 브레이커를 설치하는 등 안전 대책을 강구하십시오.
단락에 대한 안전 대책이 불충분하면 소손의 원인이 될 수 있습니다.

● 소프트웨어 라이선스 및 저작권

· 본 제품에는 일부 서드파티 소프트웨어가 포함되어 있습니다. 이 소프트웨어와 관련된 라이선스 및 저작권 정보는 http://www.fa.omron.co.jp/nx_info_e/ 에서 확인할 수 있습니다.

동작 환경 주의사항

- 이 매뉴얼의 지시에 따라 설치를 올바르게 수행하십시오.

- 다음과 같은 장소에서는 제어 시스템을 사용하지 마십시오:
  - 직사광선이 닿는 장소.
  - 사양에 규정된 범위를 벗어나는 온도 또는 습도의 장소.
  - 급격한 온도 변화로 인해 결로가 발생하는 장소.
  - 부식성 가스 또는 가연성 가스가 있는 장소.
  - 먼지(특히 철분)나 염분이 있는 장소.
  - 물, 기름 또는 화학 약품에 노출되는 장소.
  - 충격이나 진동이 있는 장소.

- 다음 장소에 시스템을 설치할 때는 적절하고 충분한 대책을 마련하십시오:
  - 강한 고주파 노이즈가 있는 장소
  - 정전기나 기타 형태의 노이즈가 있는 장소.
  - 강한 전자기장이 있는 장소.
  - 방사능에 노출될 가능성이 있는 장소.
  - 전원 공급 장치와 가까운 장소.

AI 번역본
-For Reference Only-

법규 및 규격

EMC 및 전기 안전 규정 준수

개념

오므론 제품은 다양한 기계 및 제조 설비에 내장되는 산업용 전기 기기입니다. 이 제품은 관련 규격에 적합하므로, 오므론 제품이 내장된 기계 및 설비가 EMC 및 전기 안전 규정을 더 쉽게 준수할 수 있습니다.

당사 제품이 준수하는 최신 규격에 대해서는 오므론 웹사이트(www.ia.omron.com)를 참조하거나 오므론 담당자에게 문의하십시오.

● EMC 규정 준수

본 제품은 PLC 시스템 또는 머신 오토메이션 컨트롤러에 내장될 경우 EMC 규정을 준수합니다.

귀하의 기계 또는 설비가 EMC 규정을 준수하도록 하려면 다음 주의사항을 준수하십시오.

- 본 제품은 패널 내장형 기기로 정의되며, 반드시 제어반 내부에 설치해야 합니다.
- 본 제품은 방출 규격을 준수합니다. 다만 방사 방출 요건의 경우, 사용하는 제어반의 구성, 연결 기기, 배선 방법에 따라 실제 방출량이 달라진다는 점에 유의하십시오. 따라서 EMC 규정에 적합한 기기를 사용하는 경우에도, 고객은 기계 또는 설비 전체가 EMC 규정에 적합한지를 반드시 직접 확인해야 합니다.
- DC 전원 유닛 및 I/O 유닛에 연결되는 DC 전원 공급 장치에는 강화 절연 또는 이중 절연을 사용해야 합니다.
- 오므론 S8VK-S 시리즈 전원 공급 장치를 권장합니다. 이 권장 전원 공급 장치는 EMC 규격 적합성이 확인되었습니다.

주의:

본 기기는 주거 환경에서 사용하도록 의도되지 않았으며, 그러한 환경에서는 무선 수신에 대한 적절한 보호를 제공하지 못할 수 있습니다.

AI 번역본
For Reference Only-

● 전기 안전 규정 준수

본 제품은 EU 지령 및 UKCA 등 특정 법률 및 규정에서 요구하는 전기 안전 규정을 준수합니다.

각 제품에 대한 주의사항은 제품에 동봉된 취급설명서를 참조하십시오.

선급 규격 적합성

본 제품은 다음 선급 규격에 적합합니다. 선급 규격에 대한 적합성은 특정 사용 조건을 전제로 합니다. 일부 장소에서는 제품을 사용할 수 없는 경우가 있습니다. 선박에서 PLC를 사용하기 전에 오므론 담당자에게 문의하십시오.

당사 제품이 적합한 선급 규격에 대해서는 오므론 웹사이트(www.ia.omron.com)를 참조하거나 오므론 담당자에게 문의하십시오.

NK 및 LR 선급 규격의 사용 조건

● 선교 또는 갑판 이외의 용도에 대한 사용 조건

· PLC는 반드시 제어반 내에 설치해야 합니다.
· 제어반 도어의 틈새는 개스킷이나 그 밖의 재료로 완전히 메우거나 덮어야 합니다.

● 선교 및 갑판에 대한 사용 조건 (NK 인증만 해당)

· PLC는 반드시 제어반 내에 설치해야 합니다.
· 제어반 도어의 틈새는 개스킷이나 그 밖의 재료로 완전히 메우거나 덮어야 합니다.
· 다음 노이즈 필터를 전원 라인에 접속해야 합니다.

노이즈 필터

제조사	Cosel Co., Ltd.
형식	TAH-06-683

UL 및 CSA 적합성

CJ 시리즈에는 UL 및 CSA 규격에 적합한 모델이 포함되어 있습니다.
UL 및 CSA 규격에 적합한 모델을 사용하여 귀하의 기계 또는 설비에 이러한 규격을 적용하려면 제품에 동봉된 INSTRUCTION SHEET를 참조하십시오. INSTRUCTION SHEET에는 규격 적용을 위한 사용 조건이 설명되어 있습니다.

KC 규격 적합성

본 제품을 대한민국에서 사용하는 경우 다음 주의사항을 준수하십시오.

사 용 자 안 내 문
이 기기는 업무용 환경에서 사용할 목적으로 적합성평가를 받은 기기로서 가정용 환경에서 사용하는 경우 전파간섭의 우려가 있습니다.

본 제품은 업무용 전자파 적합성 요건을 충족합니다. 본 제품을 가정에서 사용할 경우 전파 간섭이 발생할 위험이 있습니다.

소프트웨어 라이선스 및 저작권

본 제품에는 특정 타사 소프트웨어가 포함되어 있습니다. 이 소프트웨어와 관련된 라이선스 및 저작권 정보는 https://www.fa.omron.co.jp/product/tool/cj-info/index_en.html 에서 확인할 수 있습니다.

・OpenSSL
본 제품에는 OpenSSL Toolkit(http://www.openssl.org/)에서 사용하기 위해 OpenSSL Project가 개발한 소프트웨어가 포함되어 있습니다.
Copyright (C) 1998-2019 The OpenSSL Project. All rights reserved.
Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com) All rights reserved.
본 제품에는 Eric Young이 작성한 암호화 소프트웨어가 포함되어 있습니다

AI 번역본
-For Reference Only-

CJ2/CJ3 CPU 유닛의 버전

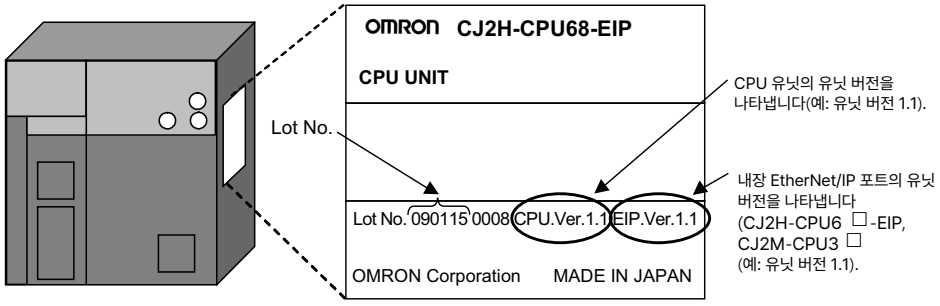
CJ2/CJ3 CPU 유닛의 하드웨어와 소프트웨어는 하드웨어 리비전과 유닛 버전으로 관리됩니다. (하드웨어 리비전은 CJ3 CPU 유닛에서만 지원됩니다.)
하드웨어 또는 소프트웨어 사양이 변경될 때마다 하드웨어 리비전 또는 유닛 버전이 갱신됩니다. 따라서 형식이 같은 유닛이라도 하드웨어 리비전이나 유닛 버전이 다르면 기능이나 성능에 차이가 있습니다.

출하 시점의 버전은 제품의 명판 또는 지원 소프트웨어로 확인할 수 있습니다.

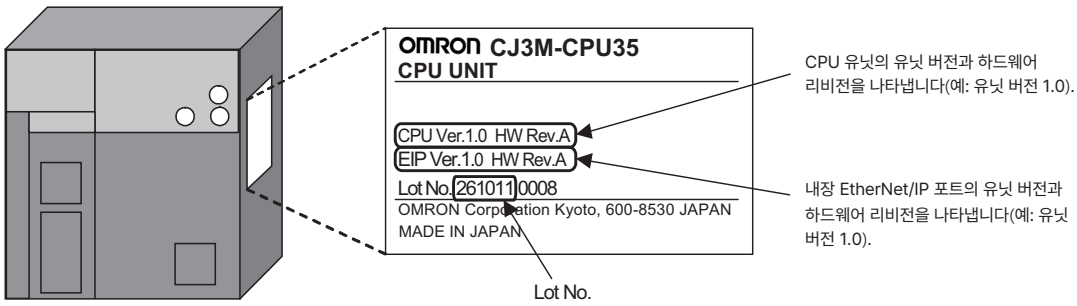
제품의 유닛 버전 표기

유닛 버전이 관리되는 제품의 경우, 아래와 같이 명판의 로트 번호 오른쪽에 유닛 버전이 표기되어 있습니다.

CJ2 CPU 유닛의 경우



CJ3 CPU 유닛의 경우



지원 소프트웨어로 유닛 버전 확인

CX-Programmer에서는 다음 두 가지 방법 중 하나로 유닛 버전을 확인할 수 있습니다.

- **PLC Information 사용**
- Unit Manufacturing Information 사용(이 방법은 특수 I/O 유닛과 CPU 버스 유닛에도 사용할 수 있습니다.)

● PLC Information

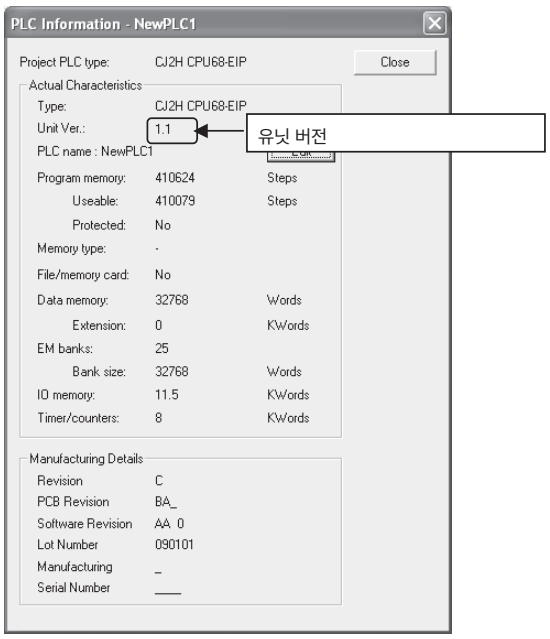
CPU 유닛의 유닛 버전만 확인할 수 있습니다.

1 다음 방법 중 하나로 PLC Information 대화 상자를 표시합니다.

-For Reference Only-

- 디바이스 종별과 CPU 종별을 알고 있는 경우, Change PLC 대화 상자에서 이를 선택하고 온라인으로 전환한 후 메뉴에서 PLC - Edit - Information을 선택합니다.
- 디바이스 종별과 CPU 종별을 모르지만 시리얼 회선으로 CPU 유닛에 직접 접속되어 있는 경우, PLC - Auto Online을 선택하여 온라인으로 전환한 후 메뉴에서 PLC - Edit - Information을 선택합니다.

2 어느 경우든 다음 PLC Information 대화 상자가 표시됩니다.

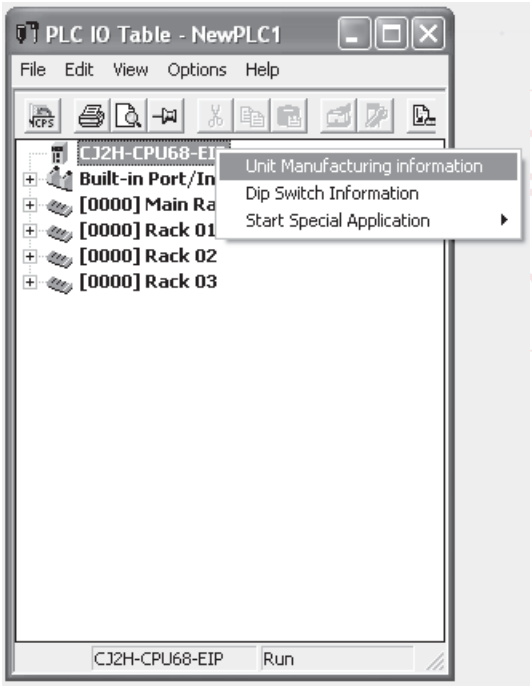


위 화면에서 CPU 유닛의 유닛 버전을 확인하십시오.

● Unit Manufacturing Information

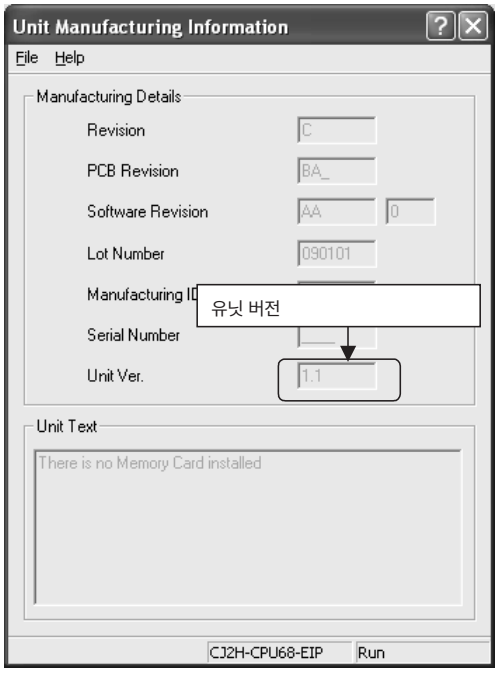
CPU 유닛과 내장 EtherNet/IP 부의 유닛 버전, 그리고 특수 I/O 유닛과 CPU 버스 유닛의 유닛 버전을 확인할 수 있습니다.

1 I/O Table 윈도우에서 마우스 오른쪽 버튼을 클릭하고 Unit Manufacturing information - CPU Unit을 선택합니다.



-열 본
nce Only-

2 다음 Unit Manufacturing information 대화 상자가 표시됩니다.



위 화면에서 온라인으로 접속된 CPU 유닛의 유닛 버전을 확인하십시오.

3 유닛 버전 라벨 사용

CPU 유닛에는 다음과 같은 유닛 버전 라벨이 동봉되어 있습니다.



번역본
-ference Only-

이 라벨을 기존 CPU 유닛의 전면에 부착하면 유닛 버전이 다른 CPU 유닛을 구별할 수 있습니다.

유닛 버전

CJ2 CPU 유닛의 유닛 버전은 다음과 같습니다.
각 유닛 버전이 지원하는 기능에 대해서는 A-7 유닛 버전별 지원 기능을 참조하십시오.

항목	형식	유닛 버전
CJ2H CPU 유닛	CJ2H-CPU6□-EIP	유닛 버전 1.0 (내장 EtherNet/IP 부: 유닛 버전 2.0)
		유닛 버전 1.1 (내장 EtherNet/IP 부: 유닛 버전 2.0)
		유닛 버전 1.2 (내장 EtherNet/IP 부: 유닛 버전 2.0)
		유닛 버전 1.3 (내장 EtherNet/IP 부: 유닛 버전 2.0)
		유닛 버전 1.4 (내장 EtherNet/IP 부: 유닛 버전 2.0)
		유닛 버전 1.5 (내장 EtherNet/IP 부: 유닛 버전 3.0)
		유닛 버전 1.6 (내장 EtherNet/IP 부: 유닛 버전 3.0)
	CJ2H-CPU6□	유닛 버전 1.1
		유닛 버전 1.2
		유닛 버전 1.3
		유닛 버전 1.4
		유닛 버전 1.5
		유닛 버전 1.6
CJ2M CPU 유닛	CJ2M-CPU3□	유닛 버전 1.0 (내장 EtherNet/IP 부: 유닛 버전 2.0) 유닛 버전 2.0 (내장 EtherNet/IP 부: 유닛 버전 2.0) 유닛 버전 2.0 (내장 EtherNet/IP 부: 유닛 버전 2.1) 유닛 버전 2.1 (내장 EtherNet/IP 부: 유닛 버전 2.1) 유닛 버전 2.2 (내장 EtherNet/IP 부: 유닛 버전 2.1)
	CJ2M-CPU1□	유닛 버전 1.0 유닛 버전 2.0 유닛 버전 2.1 유닛 버전 2.2
CJ3M CPU 유닛	CJ3M-CPU3□	유닛 버전 1.0 (내장 EtherNet/IP 부: 유닛 버전 1.1)
	CJ3M-CPU1□	유닛 버전 1.0

AI 번역본
-For Reference Only-

유닛 버전과 프로그래밍 디바이스

다음 표는 유닛 버전과 CX-Programmer 버전의 관계를 나타냅니다. 각 유닛 버전이 지원하는 기능에 대해서는 A-7 유닛 버전별 지원 기능을 참조하십시오.

● 유닛 버전과 프로그래밍 디바이스

· CJ2H의 경우

CPU 유닛	기능		필요한 프로그래밍 디바이스									
			CX-Programmer									프로그래밍 콘솔
			Ver. 7.1 이하	Ver. 8.0	Ver. 8.2 이상	Ver. 9.0	Ver. 9.1	Ver. 9.2	Ver. 9.3	Ver. 9.6 이상	Ver. 9.81 이상	
CJ2H-CPU6□-EIP 유닛 버전 1.0	유닛 버전 1.0의 기능		---	OK	OK	OK	OK	OK	OK	OK	OK	---*3
CJ2H-CPU6□-EIP 유닛 버전 1.1	유닛 버전 1.1에서 추가된 기능	신기능 사용 시	---	---	OK*2	OK	OK	OK	OK	OK	OK	
		신기능 미사용 시	---	OK*1	OK	OK	OK	OK	OK	OK	OK	
CJ2H-CPU6□ 유닛 버전 1.1	유닛 버전 1.1에서 추가된 기능	신기능 사용 시	---	---	OK*2	OK	OK	OK	OK	OK	OK	
		신기능 미사용 시	---	---	OK	OK	OK	OK	OK	OK	OK	
CJ2H-CPU6□-EIP 유닛 버전 1.2	유닛 버전 1.2에서 추가된 기능	신기능 사용 시	---	---	---	OK	OK	OK	OK	OK	OK	
		신기능 미사용 시	---	OK*1	OK*1	OK	OK	OK	OK	OK	OK	
CJ2H-CPU6□ 유닛 버전 1.2	유닛 버전 1.2에서 추가된 기능	신기능 사용 시	---	---	---	OK	OK	OK	OK	OK	OK	
		신기능 미사용 시	---	---	OK*1	OK	OK	OK	OK	OK	OK	
CJ2H-CPU6□-EIP 유닛 버전 1.3	유닛 버전 1.3에서 추가된 기능	신기능 사용 시	---	---	---	---	OK	OK	OK	OK	OK	
		신기능 미사용 시	---	OK*1	OK*1	OK	OK	OK	OK	OK	OK	
CJ2H-CPU6□ 유닛 버전 1.3	유닛 버전 1.3에서 추가된 기능	신기능 사용 시	---	---	---	---	OK	OK	OK	OK	OK	
		신기능 미사용 시	---	---	OK*1	OK	OK	OK	OK	OK	OK	
CJ2H-CPU6□-EIP 유닛 버전 1.4	유닛 버전 1.4에서 추가된 기능	신기능 사용 시	---	---	---	---	---	---	OK	OK	OK	
		신기능 미사용 시	---	OK*1	OK*1	OK	OK	OK	OK	OK	OK	
CJ2H-CPU6□ 유닛 버전 1.4	유닛 버전 1.4에서 추가된 기능	신기능 사용 시	---	---	---	---	---	---	OK	OK	OK	
		신기능 미사용 시	---	---	OK*1	OK	OK	OK	OK	OK	OK	
CJ2H-CPU6□-EIP 유닛 버전 1.5	유닛 버전 1.5에서 추가된 기능	신기능 사용 시	---	---	---	---	---	---	---	OK	OK	
		신기능 미사용 시	---	OK*1	OK*1	OK	OK	OK	OK	OK	OK	
CJ2H-CPU6□ 유닛 버전 1.5	유닛 버전 1.5에서 추가된 기능	신기능 사용 시	---	---	---	---	---	---	---	OK	OK	
		신기능 미사용 시	---	---	OK*1	OK	OK	OK	OK	OK	OK	
CJ2H-CPU6□-EIP 유닛 버전 1.6	---*4	---	---	---	---	---	---	---	---	---	---	
CJ2H-CPU6□ 유닛 버전 1.6	유닛 버전 1.6에서 추가된 기능	신기능 사용 시	---	---	---	---	---	---	---	---	OK	
		신기능 미사용 시	---	---	OK*1	OK	OK	OK	OK	OK	OK	

AI 번역본
For Reference Only

*1 CPU 유닛의 업그레이드로 강화된 기능을 사용하지 않는 경우에는 CX-Programmer의 버전을 업그레이드할 필요가 없습니다.

*2 유닛 버전 1.1에서 추가된 기능을 사용하려면 CX-Programmer 버전 8.2 이상이 필요합니다. 단, 고속 인터럽트 기능과 MONITOR 모드에서의 최소 사이클 타임 설정 변경은 CX-Programmer 버전 8.02에서도 지원됩니다.

*3 프로그래밍 콘솔은 CJ2 CPU 유닛에서 사용할 수 없습니다.

*4 유닛 버전 1.6의 CJ2H-CPU6□-EIP에는 유닛 버전 1.5에 대해 추가된 기능이 없습니다. 따라서 필요한 프로그래밍 디바이스의 버전은 유닛 버전 1.5의 경우와 동일합니다.

· CJ2M의 경우

CPU 유닛	기능		필요한 프로그래밍 디바이스									
			CX-Programmer									프로그래밍 콘솔
			Ver. 7.1 이하	Ver. 8.0	Ver. 8.2 이상	Ver. 9.0	Ver. 9.1	Ver. 9.2	Ver. 9.3	Ver. 9.6 이상	Ver. 9.81 이상	
CJ2M-CPU□□ 유닛 버전 1.0	유닛 버전 1.0의 기능		---	---	---	---	OK	OK	OK	OK	OK	---*3
CJ2M-CPU□□ 유닛 버전 2.0	유닛 버전 2.0에서 추가된 기능	신기능 사용 시	---	---	---	---	---	OK	OK	OK	OK	
		신기능 미사용 시	---	---	---	---	OK*1	OK	OK	OK	OK	
CJ2M-CPU□□ 유닛 버전 2.1	유닛 버전 2.1에서 추가된 기능	신기능 사용 시	---	---	---	---	---	---	---	OK	OK	
		신기능 미사용 시	---	---	---	---	OK*1	OK	OK	OK	OK	
CJ2M-CPU3□ 유닛 버전 2.2	---*4	---	---	---	---	---	---	---	---	---	---	
CJ2M-CPU1□ 유닛 버전 2.2	유닛 버전 2.2에서 추가된 기능	신기능 사용 시	---	---	---	---	---	---	---	---	OK	
		신기능 미사용 시	---	---	---	---	OK*1	OK	OK	OK	OK	

*1 CPU 유닛의 업그레이드로 강화된 기능을 사용하지 않는 경우에는 CX-Programmer의 버전을 업그레이드할 필요가 없습니다.

*2 유닛 버전 1.1에서 추가된 기능을 사용하려면 CX-Programmer 버전 8.2 이상이 필요합니다. 단, 고속 인터럽트 기능과 MONITOR 모드에서의 최소 사이클 타임 설정 변경은 CX-Programmer 버전 8.02에서도 지원됩니다.

*3 프로그래밍 콘솔은 CJ2 CPU 유닛에서 사용할 수 없습니다.

*4 유닛 버전 2.2의 CJ2M-CPU3□에는 유닛 버전 2.1에 대해 추가된 기능이 없습니다. 따라서 필요한 프로그래밍 디바이스의 버전은 유닛 버전 2.1의 경우와 동일합니다.

· CJ3M의 경우

CPU 유닛	기능	필요한 프로그래밍 디바이스	
		CX-Programmer	프로그래밍 콘솔
CJ3M-CPU1□ CJ3M-CPU3□ 유닛 버전 1.0	유닛 버전 1.0의 기능	Ver. 9.9 이상	---

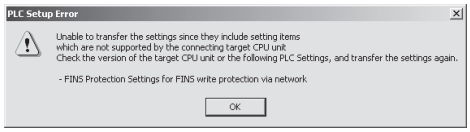
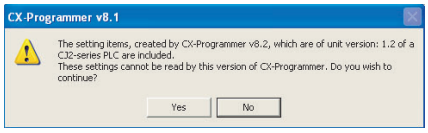
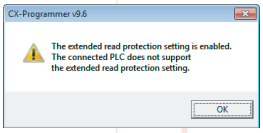
-For Reference Only-

● PLC 기종 풀다운 리스트

CX-Programmer의 Change PLC 대화 상자에 있는 PLC 기종 풀다운 리스트에서는 유닛 버전이 구분되지 않습니다. 유닛 버전에 관계없이 다음 표와 같이 선택하십시오.

시리즈	CPU 유닛	형식	CX-Programmer의 Change PLC 대화 상자의 PLC 기종
CJ 시리즈	CJ2H CPU 유닛	CJ2H-CPU6□-EIP CJ2H-CPU6□	CJ2H
	CJ2M CPU 유닛	CJ2M-CPU3□ CJ2M-CPU1□	CJ2M
	CJ3M CPU 유닛	CJ3M-CPU3□ CJ3M-CPU1□	CJ3M

CX-Programmer에서의 유닛 버전 관련 문제 해결

현상	원인	대책
 위 메시지가 표시된 후 Output 윈도우의 Compile 탭 페이지에 컴파일 에러가 표시됩니다.	이후 유닛 버전 또는 CPU 유닛에서만 지원되는 명령어를 포함한 프로그램을 이전 유닛 버전에 다운로드하려고 했습니다.	프로그램을 확인하거나 이후 유닛 버전의 CPU 유닛으로 교체하십시오.
	이후 유닛 버전 또는 CPU 유닛에서만 지원되는 설정을 포함한 PLC Setup을 이전 유닛 버전에 다운로드하려고 했습니다.	PLC Setup의 설정을 확인하거나 이후 유닛 버전의 CPU 유닛으로 교체하십시오.
PLC에서 CX-Programmer로 전송한 프로그램에 "????"가 표시됩니다.	상위 버전의 CX-Programmer에서만 지원되는 명령어를 포함한 프로그램을 하위 버전의 CX-Programmer로 업로드하려고 했습니다.	신규 명령어는 하위 버전의 CX-Programmer로 업로드할 수 없습니다. 상위 버전의 CX-Programmer를 사용하십시오.
 프로젝트 파일을 읽어 들일 때 위 에러가 표시됩니다.	지원되지 않는 유닛 버전의 프로젝트 파일을 읽어 들이려고 했습니다.	Yes 버튼을 클릭하면 지원되지 않는 설정을 초기화하고 파일을 읽어 들입니다. No 버튼을 클릭하면 프로젝트 파일 읽기를 취소합니다.
 온라인으로 전환할 때 위 경고가 표시됩니다.	상위 버전의 CPU 유닛에서만 지원되는 확장 읽기 보호 설정이 포함된 프로젝트 파일에 대해 하위 버전의 CPU 유닛과 온라인으로 전환하려고 했습니다.	PLC Properties 대화 상자에서 보호 설정을 변경하십시오. 또는 온라인으로 접속해야 하는 CPU 유닛을 상위 버전의 CPU 유닛으로 교체하십시오.

-For Reference Only-

관련 매뉴얼

CJ 시리즈 CJ2/CJ3 CPU 유닛으로 구성한 PLC와 관련된 매뉴얼을 다음 표에 나타냅니다.
이들 매뉴얼을 참조하십시오.

매뉴얼	자료 번호	형식	용도	내용
CJ 시리즈 CJ2/CJ3 CPU 유닛 소프트웨어 사용자 매뉴얼 (본 매뉴얼)	W473	CJ3M-CPU□□ CJ2H-CPU6□-EIP CJ2H-CPU6□ CJ2M-CPU□□	CJ2/CJ3 CPU 유닛의 소프트웨어 사양	CJ2/CJ3 CPU 유닛에 대해 다음 내용을 설명합니다. · CPU 유닛의 동작 · 내부 메모리 · 프로그래밍 · 설정 · CPU 유닛에 내장된 기능 하드웨어 사용자 매뉴얼 (W472)도 함께 참조하십시오.
CJ 시리즈 CJ2 CPU 유닛 하드웨어 사용자 매뉴얼	W472	CJ2H-CPU6□-EIP CJ2H-CPU6□ CJ2M-CPU□□	CJ2/CJ3 CPU 유닛의 하드웨어 사양	CJ2/CJ3 CPU 유닛에 대해 다음 내용을 설명합니다. · 개요와 특장 · 기본 시스템 구성 · 각 부의 명칭과 기능 · 장착 및 설정 순서 · 에러 대책 · 소프트웨어 사용자 매뉴얼 (W473)도 함께 참조하십시오.
CJ 시리즈 CJ3 CPU 유닛 하드웨어 사용자 매뉴얼	W664	CJ3M-CPU□□	CJ3 CPU 유닛의 하드웨어 사양	CJ3 CPU 유닛에 대해 다음 내용을 설명합니다. · 개요와 특장 · 기본 시스템 구성 · 각 부의 명칭과 기능 · 장착 및 설정 순서 · 에러 대책 · 소프트웨어 사용자 매뉴얼 (W473)도 함께 참조하십시오.
CS/CJ 시리즈 EtherNet/IPTM 유닛 오퍼레이션 매뉴얼	W465	CJ3M-CPU3□ CJ2H-CPU6□-EIP CJ2M-CPU3□/4□ CS1W-EIP21/EIP21S CJ1W-EIP21/EIP21S	CJ2/CJ3 CPU 유닛의 내장 EtherNet/IP 포트 사용	내장 EtherNet/IP 포트와 EtherNet/IP 유닛에 대해 설명합니다. 기본 설정, 태그 데이터 링크, FINS 통신 및 기타 기능에 대해 설명합니다.
CJ 시리즈 CJ2M/CJ3M CPU 유닛 펄스 I/O 모듈 사용자 매뉴얼	W486	CJ3M-CPU3□ CJ2M-CPU□□ + CJ2M-MD21□	CJ2M/CJ3M CPU 유닛에서 펄스 I/O를 사용하기 위한 정보	CJ2M/CJ3M CPU 유닛에 대해 다음 정보를 제공합니다. · 사양과 배선 방법 · I/O 기능 · 퀵 리스폰스 입력 · 인터럽트 기능 · 고속 카운터 · 펄스 출력 · PWM 출력 프로그래밍 시에는 이 매뉴얼을 명령어 레퍼런스 매뉴얼(자료 번호 W474)과 함께 사용하십시오.

AI 번역본
-For Reference Only-

매뉴얼	자료 번호	형식	용도	내용
CS/CJ/NSJ 시리즈 명령어 레퍼런스 매뉴얼	W474	CJ3M-CPU□□ CJ2H-CPU6□-EIP CJ2H-CPU6□ CJ2M-CPU□□ CS1G/H-CPU□□H CS1G/H-CPU□□-V1 CS1D-CPU□□H CS1D-CPU□□S CJ1H-CPU□□H-R CJ1G/H-CPU□□H CJ1G-CPU□□P CJ1G-CPU□□ CJ1M-CPU□□ NSJ□-□□□□(B)-G5D NSJ□-□□□□(B)-M3D	명령어에 관한 정보	각 프로그래밍 명령어에 대해 자세히 설명합니다. 프로그래밍 시에는 소프트웨어 사용자 매뉴얼 (W473)도 함께 참조하십시오.
CS/CJ/CP/NSJ 시리즈 통신 커맨드 레퍼런스 매뉴얼	W342	CJ3M-CPU□□ 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/N□□D□-□	CS/CJ/CP 시리즈 CPU 유닛 및 NSJ 시리즈 컨트롤러의 통신에 관한 정보	C 모드 커맨드와 FINS 커맨드에 대해 설명합니다 C 모드 커맨드 또는 FINS 커맨드로 CPU 유닛과 통신할 때의 커맨드에 대한 자세한 설명은 이 매뉴얼을 참조하십시오. 주 이 매뉴얼은 CPU 유닛을 대상으로 하는 통신 커맨드에 대해 설명합니다. 사용되는 통신 경로는 관계없으며, 다음 중 어느 것이든 포함될 수 있습니다: CPU 유닛의 시리얼 포트, 시리얼 통신 유닛/보드의 통신 포트, 통신 유닛. 특수 I/O 유닛 또는 CPU 버스 유닛을 대상으로 하는 통신 커맨드에 대해서는 해당 유닛의 오퍼레이션 매뉴얼을 참조하십시오.
CX-One 셋업 매뉴얼	W463	CXONE-AL□□D-V4/ LT□□□-V4	CX-One에서 소프트웨어 설치	CX-One FA 통합 툴 패키지의 개요를 제공하고 설치 순서를 설명합니다.
CX-Programmer 오퍼레이션 매뉴얼	W446	CXONE-AL□□D-V4	Windows 컴퓨터용 지원 소프트웨어 CX-Programmer 조작 순서	CX-Programmer의 조작 순서를 설명합니다. 프로그래밍 시에는 소프트웨어 사용자 매뉴얼 (W473) 및 CS/CJ/NSJ 시리즈 명령어 레퍼런스 매뉴얼(W474)도 함께 참조하십시오.
CX-Programmer 오퍼레이션 매뉴얼 펑션 블록/구조화 텍스트	W447			
CX-Programmer 오퍼레이션 매뉴얼 SFC 프로그래밍	W469			
CS/CJ/CP/NSJ 시리즈 CX-Simulator 오퍼레이션 매뉴얼	W366		Windows 컴퓨터용 CX-Simulator 시뮬레이션 지원 소프트웨어의 조작 순서 CX-Programmer 버전 6.1 이상에서 CX-Programmer의 시뮬레이션 기능 사용	CX-Simulator의 조작 순서를 설명합니다. 시뮬레이션을 실시할 때는 CX-Programmer 오퍼레이션 매뉴얼(W446), 소프트웨어 사용자 매뉴얼(W473) 및 CS/CJ/NSJ 시리즈 명령어 레퍼런스 매뉴얼(W474)도 함께 참조하십시오.
CS/CJ/CP/NSJ 시리즈 CX-Integrator 네트워크 구성 소프트웨어 오퍼레이션 매뉴얼	W464		네트워크 셋업 및 모니터링	CX-Integrator의 조작 순서를 설명합니다.

AI 번역본
-For Reference Only-

AI 번역본
-For Reference Only-

1

개요

이 절에서는 CJ2/CJ3 CPU 유닛의 개요를 설명합니다.

1-1 CJ2/CJ3 CPU 유닛의 개요 1-2
1-1-1 개요 1-2
1-1-2 CJ2/CJ3 CPU 유닛의 특징 1-3
1-1-3 CJ3M CPU 유닛에서 강화된 기능 1-10
1-2 기본 조작 순서 1-11

AI 번역본
-For Reference Only-

1-1 CJ2/CJ3 CPU 유닛의 개요

1-1-1 개요

SYSMAC CJ2/CJ3 시리즈 CPU 유닛은 다음과 같은 특장을 갖춘 다기능 CPU 유닛입니다.

● 고속 처리와 대용량 메모리

기본 성능이 고속화되고 메모리 용량이 증가하여 기계 제어에 충분한 능력을 제공합니다.

● 내장 EtherNet/IP 포트

CJ2/CJ3 CPU 유닛은 EtherNet/IP 오픈 네트워크를 표준으로 지원합니다. PLC 간 데이터 링크, PLC 간 메시지 통신, FTP 전송 등의 범용 이더넷 통신을 주변 디바이스 접속에서 모두 실현할 수 있습니다.
(CJ2H-CPU6□-EIP, CJ2M-CPU3□, CJ3M-CPU3□ 전용)

● 지원 소프트웨어 인터페이스를 위한 범용 네트워크

USB 및 EtherNet/IP 포트를 통해 시판 케이블로 지원 소프트웨어와 디바이스를 범용 네트워크에 간편하게 접속할 수 있습니다.

● 태그 액세스

CJ2/CJ3 CPU 유닛에는 태그명과 I/O 어드레스를 관리하는 태그명 서버가 있습니다. 이를 통해 I/O 어드레스를 몰라도 태그명으로 외부 디바이스에서 액세스할 수 있습니다.

● 프로그래밍의 간편화

CJ2/CJ3 CPU 유닛은 DM 및 EM 영역의 비트 어드레싱, 어드레스 오프셋 설정, 배열 변수 사용 등의 기능을 갖춘 가독성 높은 프로그래밍 환경을 제공합니다.

● 디버깅 기능 향상

온라인 에디팅과 데이터 트레이스가 개선되어 디버깅 효율이 크게 향상되었습니다.

● 펄스 I/O 용량 증가

CJ2M/CJ3M CPU 유닛에서는 옵션인 펄스 I/O 모듈을 장착하여 최대 4축의 펄스 I/O를 실현할 수 있습니다.

AI 번역본
-For Reference Only-

● 배터리리스 I/O 메모리 백업

정전 시에도 배터리 없이 I/O 메모리가 유지됩니다. 더 이상 배터리 메인터넌스가 필요하지 않습니다. (CJ3M CPU 유닛 전용)

● 보안 기능 강화

다양한 보안 기능이 네트워크 액세스는 물론 디바이스에 저장된 프로그램과 데이터의 보안을 강화하여 고객 자산의 도난이나 변조 위험을 줄입니다. (CJ3M CPU 유닛 전용)

● 동기 유닛 동작 (CJ2H CPU 유닛)

CPU 버스 유닛과 특수 I/O 유닛이 수행하는 처리의 타이밍을 동기화할 수 있습니다.
이를 통해 입력 · 처리에서 출력까지의 타이밍 변동을 최소화하여 애플리케이션 성능을 확보하기 쉬워집니다.

1-1-2 CJ2/CJ3 CPU 유닛의 특징

고속 · 대용량 메모리

● 고속 처리

기본 명령어(CJ2H는 최소 0.016 μs, CJ2M/CJ3M은 최소 0.04 μs), 특수 명령어(CJ2H는 최소 0.05 μs, CJ2M/CJ3M은 최소 0.06 μs), 즉시 리프레시(CJ2H는 최소 0.99 μs, CJ2M/CJ3M은 최소 1.26 μs)의 고속 처리가 가능합니다.

● 대용량 프로그램 용량

최대 400 Ksteps의 대용량 프로그램 용량을 제공하는 모델 라인업이 갖추어져 있습니다.

● 특수 펑션 블록 정의 영역

CJ2M/CJ3M CPU 유닛에서는 펑션 블록 정의를 저장하기 위해 FB 프로그램 영역이라는 특수 영역이 제공됩니다. (CJ3M CPU 유닛: 40K steps, CJ2M CPU 유닛: 20K steps) 이에 따라 펑션 블록을 사용하여 기존 프로그램으로부터 프로그램 부품과 구조를 만들면서 사용자 프로그램 영역의 사용량을 줄일 수 있습니다. 또한 FB 프로그램 영역을 초과하는 펑션 블록 정의는 사용자 프로그램 영역에 저장됩니다.

● 대용량 데이터 메모리 용량

최대 800 Kwords(25뱅크)의 대용량 EM(확장 데이터 메모리) 영역 용량을 제공하는 모델 라인업이 갖추어져 있습니다.

● 최대 128개의 사이클릭 태스크

사용자 프로그램은 최대 128개의 태스크로 분할할 수 있습니다. 태스크 프로그램을 작게 하면 프로그램을 구조화하기 쉬워지고 사이클 타임 단축에도 기여합니다.

● 인터럽트 태스크의 실행 성능 향상

고속 인터럽트 기능을 사용하면 일정한 제약 조건 하에서 CJ2H는 아래와 같이 실행 성능을 향상시킬 수 있습니다.

- 인터럽트 태스크의 오버헤드 타임(인터럽트 태스크 기동 시간 + 사이클릭 태스크 복귀 시간)을 크게 단축합니다.

  예: I/O 인터럽트 태스크의 경우, 통상 동작 시의 시간은 37 μs이지만 다음의 경우 25 μs가 됩니다
  고속 인터럽트 기능을 사용할 때.
- 정주기 인터럽트 태스크의 최소 인터벌 설정 0.1 ms

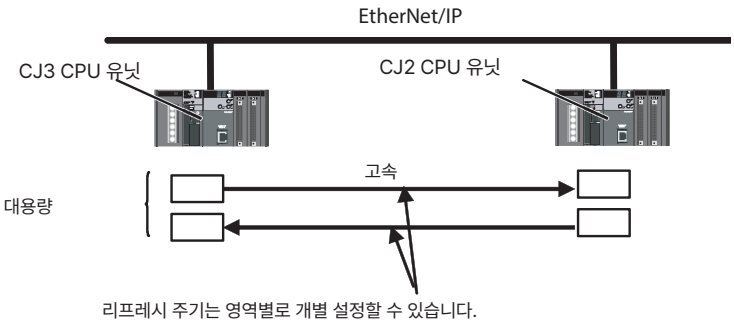
AI 번역본
-For Reference Only-

내장 EtherNet/IP 포트

CJ2/CJ3 CPU 유닛은 이더넷 유닛과 동일한 FINS Ethernet 기능(FTP 서버, 시각 맞춤 등의 기능 포함)과 EtherNet/IP 오픈 네트워크를 지원하는 내장 EtherNet/IP 포트를 표준으로 탑재하고 있습니다.
(CJ2H-CPU6□-EIP, CJ2M-CPU3□, CJ3M-CPU3□ 전용)

● 리프레시 주기를 사용자가 설정할 수 있는 고속 · 대용량 데이터 링크

CJ2/CJ3 CPU 유닛은 고속 · 대용량 데이터 링크를 제공하며, 리프레시 주기는 각 데이터 링크 영역별로 개별 설정할 수 있습니다.



지원 소프트웨어 인터페이스용 범용 네트워크

● 페리페럴 USB 포트 및 EtherNet/IP 포트

시판 USB 케이블을 사용하여 PC의 USB 포트에서 PLC에 직접 접속할 수 있습니다. 또한 CJ2H-CPU6□-EIP 또는 CJ2M-CPU3□, CJ3M-CPU3□에서는 USB를 경유하여 EtherNet/IP 네트워크상의 PLC에 액세스할 수 있습니다.

● 지원 소프트웨어에서 PLC 이름을 사용하여 잘못된 PLC로의 접속 방지

사용자가 설정한 PLC 이름을 기록할 수 있습니다. 지원 소프트웨어로 PLC에 온라인 접속할 때 PLC 이름을 조회함으로써 지원 소프트웨어로부터의 잘못된 접속을 방지합니다.

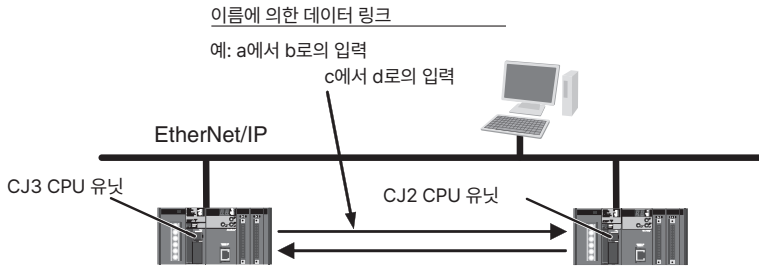
AI 번역본
-For Reference Only-

프로그램 변경에 유연하게 대응하는 네트워크 심볼(태그)

CJ2/CJ3 CPU 유닛은 네트워크 심볼(태그)을 지원합니다. 내부에 태그 네임 서버를 갖추고 있어 CPU 유닛의 심볼 테이블에 태그명과 어드레스를 미리 저장할 수 있습니다.
태그에 의해 다음과 같은 기능이 가능해집니다.

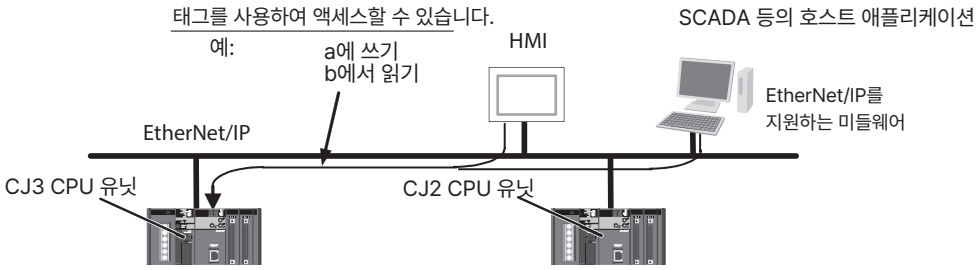
태그를 사용하여 데이터 링크를 설정할 수 있습니다.

EtherNet/IP 네트워크상의 데이터 링크에서는 송신 · 수신 영역을 어드레스가 아니라 심볼로 지정할 수 있습니다. 태그로 설정한 데이터 링크 영역은 그대로 두고, 태그명과 어드레스가 들어 있는 심볼 테이블만 변경하면 되므로 설계 변경에 유연하게 대응할 수 있습니다.



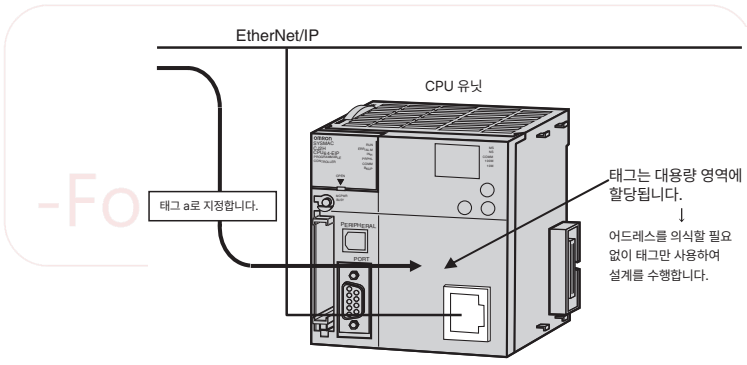
태그를 사용하여 호스트 기기에서 액세스할 수 있습니다.

HMI나 PC 등의 호스트 기기에서 액세스할 때 태그를 사용할 수 있습니다. 이에 따라 HMI 등의 기기에서 호스트 화면과 프로그램을 작성하기가 훨씬 편리해집니다.



태그 자동 할당에 의해 어드레스를 몰라도 됩니다.

CX-Programmer 심볼 테이블의 어드레스 자동 할당을 사용하여 대용량 EM 영역에 태그를 자동 할당함으로써, 어드레스를 의식하지 않고 데이터 링크를 설계하고 호스트 기기에서 액세스할 수 있습니다.



더 쉬워진 프로그래밍

● 어드레스 오프셋을 지정할 수 있습니다.

명령어의 오퍼랜드에 어드레스를 지정할 때, 어드레스 뒤의 대괄호 안에 오프셋을 지정하여 어드레스를 오프셋할 수 있습니다. 예를 들어 대괄호 안에 워드 어드레스를 설정하여 오프셋을 지정하면, 그 워드의 내용에 따라 어드레스를 동적으로 지정할 수 있습니다.

예
W0.00[D0]: W0.00이 선두 어드레스이고 D0의 내용이 오프셋입니다. D0이 &3이면 W0.03이 지정됩니다.

● 배열 변수의 첨자에 심볼을 지정할 수 있습니다.

배열 변수의 첨자에 심볼을 지정하면 심볼의 값에 따라 요소를 동적으로 지정할 수 있습니다.

예
a[b]: 심볼 b의 값이 배열 변수 a[ ]의 요소를 지정합니다.

● 데이터 구조체의 작성과 사용

CX-Programmer에서는 데이터 구조체를 작성하여 사용할 수 있습니다. 이에 따라 CPU 유닛의 I/O 메모리를 데이터베이스나 라이브러리 데이터로 손쉽게 사용할 수 있습니다.

● DM 영역과 EM 영역에서 비트 어드레스를 사용할 수 있습니다.

기존에는 DM 영역과 EM 영역을 워드 단위로만 지정할 수 있었고 비트 어드레스는 지정할 수 없었습니다. DM 영역과 EM 영역에서 비트 어드레스를 사용할 수 있게 되어 비트용 워크 영역을 확장할 수 있습니다.

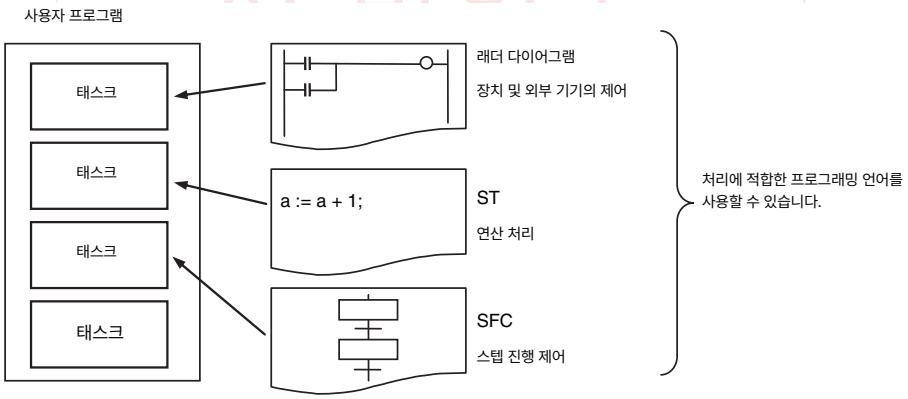
예를 들어 D10.00은 D10의 비트 00을 지정합니다.

● 타이머/카운터 현재값 리프레시 형식은 각 명령어마다 개별적으로 선택할 수 있습니다.

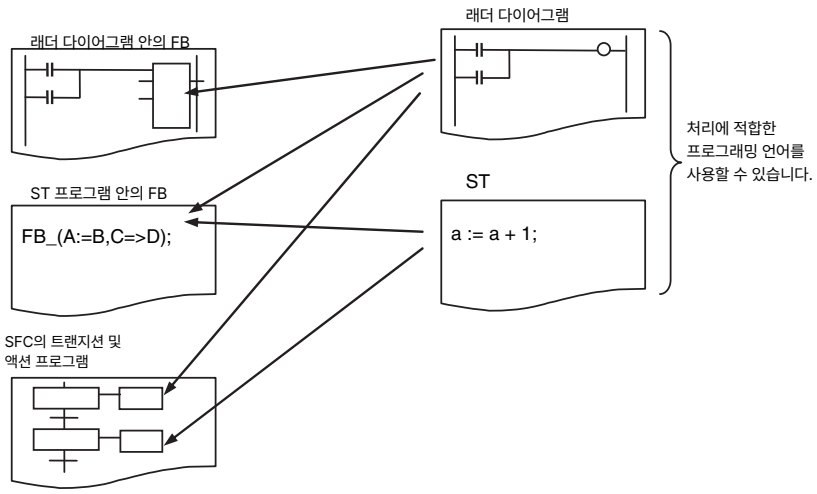
타이머/카운터 현재값 리프레시 형식으로 BCD 또는 바이너리를 각 명령어마다 개별적으로 선택할 수 있습니다. 예를 들어 TIM(BCD) 명령어와 TIMX(바이너리) 명령어를 함께 사용할 수 있습니다.

● 구조화 프로그래밍을 위해 최적의 언어를 조합할 수 있습니다.

사용자 프로그램에서 래더 다이어그램, ST, SFC 언어를 자유롭게 조합할 수 있습니다. 특정 처리에 최적인 언어를 사용할 수 있으므로 프로그래밍에 걸리는 시간을 줄이는 데 도움이 됩니다.



또한 래더 다이어그램과 ST 프로그램 안의 펑션 블록(FB), 그리고 SFC의 트랜지션 · 액션 프로그램에 래더 다이어그램과 ST 언어를 자유롭게 사용할 수 있습니다. 이 기능에 의해 구조화 프로그래밍을 최적화할 수 있습니다.



디버깅 기능 향상

● 온라인 에디팅이 사이클 타임에 미치는 영향 감소

온라인 에디팅에 의한 사이클 타임 증가분이 약 1 ms로 단축되어, 디버깅 중 장치 동작에 미치는 영향을 최소화합니다.

● 고속 · 대용량 데이터 트레이스

CJ3M CPU 유닛에서는 최대 220 Kwords의 대용량 메모리와 최대 6개의 조건을 조합하여 장시간 · 다양한 조건의 데이터 트레이스가 가능합니다.

CJ2H CPU 유닛은 최대 32 Kwords, CJ2M CPU 유닛은 최대 8,000워드까지의 데이터 트레이스를 지원합니다.

● I/O 모듈 보조 기억 영역 선택 기능

트레이스할 데이터를 선택할 때 펄스 I/O 기능의 AR을 목록에서 간단히 선택할 수 있습니다.

● 지정한 EM 영역 뱅크의 비트 강제 세트/리셋

파라미터를 설정하여 지정한 EM 영역 뱅크의 비트를 강제 세트/리셋할 수 있습니다. (CJ2H-CPU 버전 1.1 이전은 지원하지 않습니다.)

사본 번역본
For Reference Only

배터리리스 I/O 메모리 백업 기능

이 기능은 정전 시 배터리 없이 DM 영역(D), EM 영역(E), 유지 영역(H), 카운터(C), 보조 기억 영역(A)의 데이터를 유지합니다. 배터리 메인터넌스가 더 이상 필요하지 않습니다. 시계 기능을 사용하려면 옵션 배터리를 접속하십시오.

보안 기능 강화 (CJ3M CPU 유닛)

다양한 보안 기능이 네트워크 액세스는 물론 기기에 저장된 프로그램과 데이터의 보안을 강화하여, 고객 자산의 도난이나 변조 위험을 줄입니다. 이에 따라 시스템 전체의 보안 성능을 향상시킬 수 있습니다.

● 시큐어 통신 기능 (CJ3M-CPU3□ 내장 EtherNet/IP 포트)

이 기능은 CX-Programmer에서 CPU 유닛의 중요 데이터에 액세스할 때 사용하는 회선의 기밀성과 무결성을 보장합니다.

● 사용자 인증 (CJ3M-CPU3□ 내장 EtherNet/IP 포트)

이 기능은 지정한 사용자만 주변 툴에서 온라인 조작을 수행할 수 있도록 하고, 사용자별로 설정한 권한에 따라 온라인 조작을 추가로 제한합니다.

● 포트 오픈/클로즈 기능 (CJ3M-CPU3□ 내장 EtherNet/IP 포트)

이 기능은 설정 내용에 따라 개별 통신 기능에 할당된 TCP/UDP 포트를 통과하는 통신을 차단하거나 허가할 수 있습니다.

사용하지 않는 통신 기능을 Disabled로 설정하면 외부 해커가 부정하게 액세스할 수 있는 침입 경로의 수가 줄어듭니다.

● IP 패킷 필터 (CJ3M-CPU3□ 내장 EtherNet/IP 포트)

이 기능은 미리 설정한 조건에 따라 EtherNet/IP 포트가 수신한 패킷의 통과를 선택적으로 허가합니다.

● 파일 암호화 기능

이 기능은 CX-Programmer, 백업 툴, SD 메모리 카드, 파일 메모리가 취급하는 파일을 암호화하여 제3자의 부정 액세스를 방지합니다.

● 조작 로그 기능

・CPU 유닛의 조작 로그 기능
이 기능은 CPU 유닛에 대해 수행된 중요한 조작과 그 조작의 결과를 기록합니다.

・내장 EtherNet/IP 포트의 조작 로그 기능
이 기능은 SecureComm을 경유하여 온라인 상태인 온라인 사용자가 수행한 중요한 조작과 그 조작의 결과를 기록합니다.

● 확장 FINS 쓰기 액세스 로그

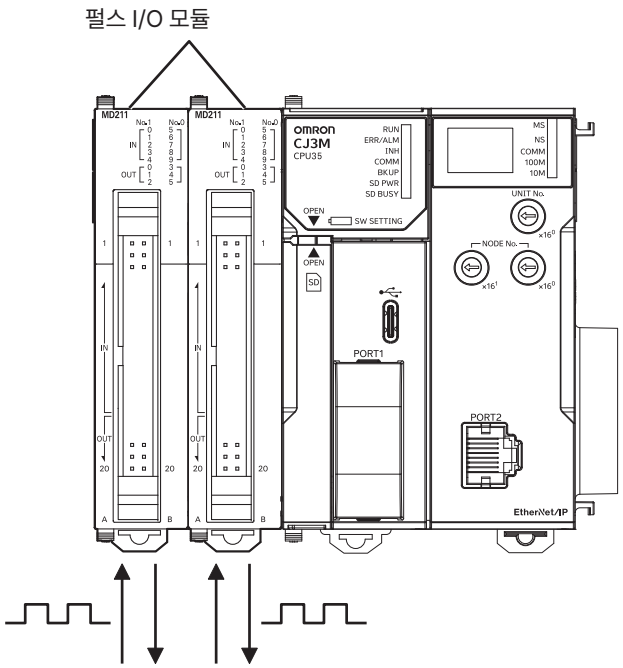
이 기능은 송신원 주소가 자기 노드가 아닐 때 외부에서 CPU 유닛에 쓰기를 요구한 이력을 보존합니다.

CJ3M-CPU3□ 내장 EtherNet/IP 포트의 보안 기능에 대해서는 CS/CJ 시리즈 EtherNet/ IP 유닛 오퍼레이션 매뉴얼 (Cat. No. W465)의 Section 13 보안 기능을
참조하십시오.

AI 번역본
-For Reference Only-

펄스 I/O와 인터럽트 입력의 사용

CJ2M/CJ3M CPU 유닛에는 최대 2대의 펄스 I/O 모듈을 장착할 수 있습니다. 이에 따라 범용 입력, 인터럽트 입력, 고속 카운터 입력, 퀵 리스폰스 입력 등의 입력 기능과 범용 출력, 펄스 출력, 가변 듀티비 펄스 출력 등의 출력 기능을 사용할 수 있습니다. 이를 통해 다양한 종류의 애플리케이션에 대응할 수 있습니다.



- 통상 입력: 20 (펄스 I/O 모듈당 10)
- 인터럽트 입력: 8 (펄스 I/O 모듈당 4)
- 고속 카운터 입력: 4 (펄스 I/O 모듈당 2)
- 퀵 리스폰스 입력: 8 (펄스 I/O 모듈당 4)
- 통상 출력: 12 (펄스 I/O 모듈당 6)
- 펄스 출력: 4 (펄스 I/O 모듈당 2)
- 가변 듀티비 펄스 출력: 4 (펄스 I/O 모듈당 2)

동기 유닛 동작

- **CJ2H CPU 유닛은 CPU 버스 유닛과 특수 I/O 유닛이 수행하는 처리의 개시를 지정한 주기에 동기화할 수 있습니다.**

  CPU 유닛과 동기 유닛* 사이에서 동기 데이터 리프레시를 사용할 수 있으며, 리프레시된 데이터를 동기 인터럽트 태스크의 연산에 사용할 수 있습니다. 이에 따라 입력 · 처리에서 출력까지의 타이밍 변동을 최소화한 다축 제어가 가능합니다. 이를 통해 전자 캠 제어와 같이 정밀한 타이밍이 요구되는 모션 애플리케이션에 대응할 수 있습니다.

사내 배포용
-For Reference Only-

* 동기 유닛이란 동기 유닛 동작 기능을 지원하는 CPU 버스 유닛과 특수 I/O 유닛을 말합니다 (CJ1W-NC□□4, CJ1W-NC□81/□82).

1-1-3 CJ3M CPU 유닛에서 강화된 기능

기능	강화 내용	CJ3M CPU 유닛	CJ2M CPU 유닛	CJ2H CPU 유닛
I/O 용량	최대 5,120점까지 확장되었습니다.	최대 5,120점	최대 2,560점	최대 2,560점
증설 랙 최대 개수	최대 7랙 80유닛까지 확장되었습니다.	최대 7 (80유닛)	최대 3 (40유닛)	최대 3 (40유닛)
접속 가능한 EtherNet/IP 유닛 수	접속 가능한 EtherNet/IP 유닛 수가 8유닛으로 확장되었습니다.	최대 8유닛 (CPU4□는 7유닛) 최대 7유닛 (CPU3□)	최대 2유닛	최대 8유닛 (CPU6□-EIP는 7유닛) 최대 7유닛 (CPU6□-EIP)
사용자 메모리	프로그램 용량과 FB 프로그램 영역이 확장되었습니다.	최대 100K steps FB 프로그램 영역: 40K steps	최대 60K steps FB 프로그램 영역: 20K steps	최대 400K steps
EM 영역	확장 데이터 메모리 영역 (EM 영역)이 확장되었습니다.	최대 10뱅크	최대 4뱅크	최대 25뱅크
메모리 카드	SD 메모리 카드,가 이제 지원됩니다.	SD 메모리 카드, HMC-SD□□□	메모리 카드 HMC-EF□□□	메모리 카드 HMC-EF□□□
배터리리스 I/O 메모리 백업	정전 시 배터리 없이 I/O 메모리가 유지됩니다.	지원	-	-
데이터 트레이스	· 트레이스 메모리가 확장되어 장시간에 걸쳐 데이터를 트레이스할 수 있게 되었습니다. · 여러 비트와 워드를 조합한 트리거 조건을 설정할 수 있습니다.	· 메모리 용량 220,000워드 · 샘플링 수 비트 = 최대 31 워드 데이터: 최대 32 · 트리거 조건 4비트 또는 2워드의 조합	· 메모리 용량 8,000워드 · 샘플링 수 비트 = 최대 31 워드 데이터: 최대 16 · 트리거 조건 1비트 또는 1워드	· 메모리 용량 32,000워드 · 샘플링 수 비트 = 최대 31 워드 데이터: 최대 16 · 트리거 조건 1비트 또는 1워드
보안	조작 로그 기능	지원	-	-
	파일 암호화 기능	지원	-	-
	시큐어 통신 기능	지원 *1	-	-
	사용자 인증	지원 *1	-	-
	포트 오픈/클로즈 기능	지원 *1	-	-
	IP 패킷 필터	지원 *1	-	-
내장 EtherNet/IP	EtherNet/IP 기능이 강화되었습니다.	CJ1W-EIP21S 상당	CJ2M 전용	CJ1W-EIP21 상당
	커넥션 수	256	32	256
	허용 통신 대역	초당 12,000 패킷	초당 3,000 패킷	초당 12,000 패킷
	등록 가능 태그 수	256	32	256
	노드당 최대 링크 데이터 사이즈	184,842워드	6,400워드	184,842워드
	커넥션당 최대 데이터 사이즈	722워드	6,400워드	722워드
	등록 가능 태그 세트 수	256	32	256
	소켓 서비스	지원 (TCP/UDP)	-	-
	Modbus 서버	지원 (TCP/UDP)	-	-
	Modbus 클라이언트	지원 (TCP/UDP)	-	-

*1 CJ3M-CPU3□ 전용.

AI 번역본
For Reference Only-

1-2 기본 조작 순서

일반적으로 다음 순서로 진행합니다.

1. 기기 · 하드웨어 설정

전원 유닛, CPU 유닛, 기타 유닛 및 엔드 커버를 장착합니다. 필요에 따라 딥 스위치와 로터리 스위치를 설정합니다.

CJ2/CJ3 CPU 유닛 하드웨어 사용자 매뉴얼 (Cat. No. W472/W664)의 Section 3 각 부의 명칭과 기능 및 Section 5 설치를 참조하십시오.

2. 배선

전원, I/O, 통신을 배선합니다. 다음 매뉴얼을 참조하십시오.

CJ2/CJ3 CPU 유닛 하드웨어 사용자 매뉴얼 (Cat. No. W472/W664)의 Section 5 설치를 참조하십시오.
CJ2/CJ3 CPU 유닛 소프트웨어 사용자 매뉴얼 (Cat. No. W473)의 Section 11 프로그래밍 디바이스와 통신을 참조하십시오.
CJ2M/CJ3M CPU 유닛 펄스 I/O 모듈 사용자 매뉴얼 (Cat. No. W486)의 Section 3 펄스 I/O 모듈의 I/O 모듈 사양 및 배선을 참조하십시오.

3. PLC에 온라인 접속

PC를 PLC에 온라인으로 접속합니다.

CJ2/CJ3 CPU 유닛 하드웨어 사용자 매뉴얼 (Cat. No. W472/W664)의 Section 4 지원 소프트웨어를 참조하십시오

4. I/O 할당

CX-Programmer를 사용하여 장착된 유닛에 CPU 유닛의 I/O 메모리를 할당합니다.

CJ2/CJ3 CPU 유닛 소프트웨어 사용자 매뉴얼 (Cat. No. W473)의 Section 8 I/O 할당과 유닛 설정을 참조하십시오
CJ2M/CJ3M CPU 유닛 펄스 I/O 모듈 사용자 매뉴얼 (Cat. No. W486)의 Section 2 I/O 적용 순서와 기능 할당을 참조하십시오.

5. 소프트웨어 설정

PLC의 소프트웨어 설정을 수행합니다.

- CPU 유닛의 초기화: CJ2/CJ3 CPU 유닛 소프트웨어 사용자 매뉴얼 (Cat. No. W473)의 Section 4 CPU 유닛의 초기화를 참조하십시오.
- 특수 I/O 유닛과 CPU 버스 유닛의 DM 영역 설정: CJ2/CJ3 CPU 유닛 소프트웨어 사용자 매뉴얼 (Cat. No. W473)의 Section 8 I/O 할당과 유닛 설정을 참조하십시오.
- PLC Setup: CJ2/CJ3 CPU 유닛 소프트웨어 사용자 매뉴얼 (Cat. No. W473)의 Section 9 PLC 설정을 참조하십시오.

AI 번역본
-For Reference Only-

6. 프로그램 작성

CX-Programmer를 사용하여 프로그램을 작성합니다.

CJ2/CJ3 CPU 유닛 소프트웨어 사용자 매뉴얼 (Cat. No. W473)의 Section 5 프로그래밍의 이해를 참조하십시오.

7. 동작 확인

I/O 배선과 AR 영역 설정을 확인하고 시운전을 수행합니다. CX-Programmer를 모니터링과 디버깅에 사용할 수 있습니다.

CJ2/CJ3 CPU 유닛 소프트웨어 사용자 매뉴얼 (Cat. No. W473)의 10-7 디버깅을 참조하십시오.

8. 프로그램 기본 운전

운전 모드를 RUN 모드로 설정하여 운전을 개시합니다.

AI 번역본
-For Reference Only-

2

CPU 유닛의 내부 메모리

이 장에서는 CPU 유닛에 포함된 메모리 영역에 대해 설명합니다.

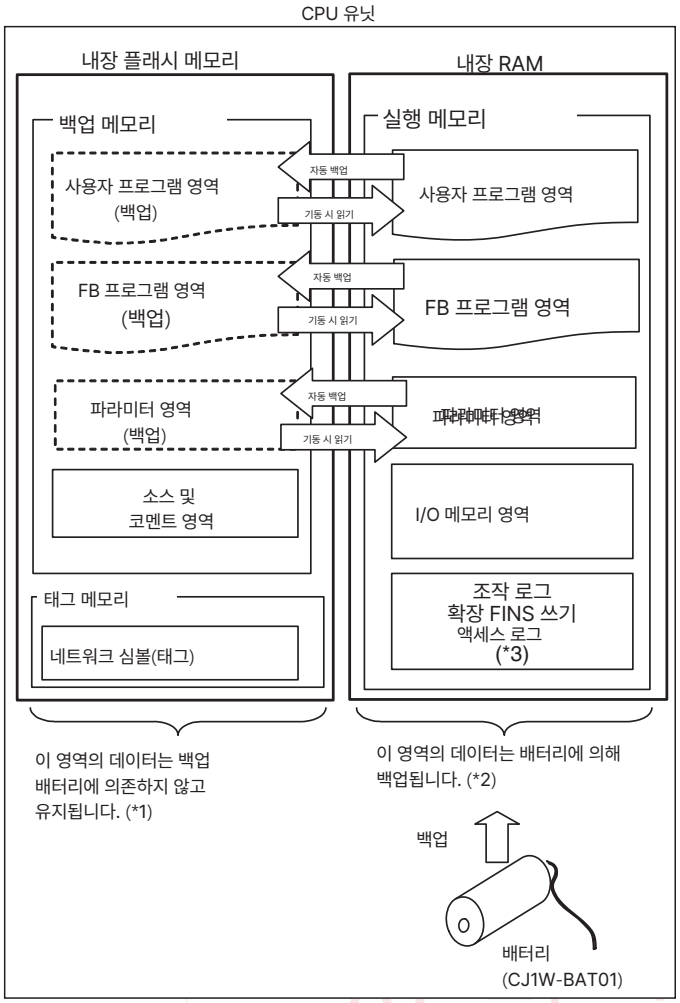
2-1 개요 2-2
2-1-1 메모리 구성 2-2
2-1-2 메모리 영역과 저장 데이터 2-4
2-1-3 프로그래밍 디바이스에서 CPU 유닛으로의 데이터 전송 2-5

AI 번역본
-For Reference Only-

2-1 개요

2-1-1 메모리 구성

다음 그림과 같이 CPU 유닛의 내부 메모리는 내장 플래시 메모리와 내장 RAM으로 구성됩니다. 내장 RAM은 실행 메모리로, 내장 플래시 메모리는 백업 메모리로 사용됩니다.



*1 사용자 프로그램과 파라미터 영역은 내장 플래시 메모리에 실시간으로 백업되므로, 전원이 OFF되거나 배터리가 동작하지 않을 때(배터리 전압이 낮거나 배터리가 접속되어 있지 않은 경우)에도 사용자 프로그램과 파라미터 영역은 플래시 메모리에 유지됩니다.

*2 **CJ2 CPU 유닛**
전원이 OFF되면 I/O 메모리 영역의 데이터는 배터리에 의해 유지됩니다.
CJ3 CPU 유닛
I/O 메모리, 조작 로그, 확장 FINS 쓰기 액세스 로그에 불휘발성 RAM을 사용하므로, 배터리가 장착되어 있지 않아도 데이터가 유지됩니다.

*3 CJ3M CPU 유닛 전용

내장 플래시 메모리

내장 플래시 메모리는 사용자 프로그램과 파라미터를 백업하고, 프로그램 소스, 코멘트, 네트워크 심볼을 저장합니다. 이 영역의 데이터는 배터리에 의존하지 않고도 유지됩니다.

-For Reference Only-

내장 RAM

내장 RAM은 CPU 유닛의 실행 메모리입니다. 사용자 프로그램, 파라미터, I/O 메모리가 내장 RAM에 저장됩니다.

・CJ2 CPU 유닛
전원이 OFF되면 I/O 메모리의 데이터는 배터리에 의해 유지됩니다. 배터리가 동작하지 않는 상태(배터리 전압이 낮거나 배터리가 접속되어 있지 않은 경우)에서 전원이 OFF되면 I/O 메모리의 데이터는 소실됩니다.

・CJ3 CPU 유닛
I/O 메모리, 조작 로그, 확장 FINS 쓰기 액세스 로그의 데이터는 불휘발성 RAM에 저장되므로, 전원이 OFF되고 배터리가 동작하지 않아도 I/O 메모리의 데이터는 유지됩니다.

사용상의 주의

배터리 전압이 낮거나 배터리가 장착되어 있지 않으면 다음과 같은 현상이 발생합니다.

・I/O 메모리의 유지 영역에서 데이터가 소실되거나 값이 불안정해집니다. 에러 로그가 보존되지 않습니다. 출력 OFF 비트가 불안정해집니다. (CJ2 CPU 유닛 전용)

・시계가 정지하고 시계 관련 데이터가 모두 불안정해집니다. (CJ2/CJ3 CPU 유닛)

AI 번역본
-For Reference Only-

2-1-2 메모리 영역과 저장 데이터

다음 표는 CPU 유닛의 메모리 영역과 각 영역에 저장되는 데이터를 나타냅니다.

메모리 영역과 저장 데이터	상세	내장 플래시 메모리	내장 RAM
사용자 프로그램 영역	사용자 프로그램 영역에는 CX-Programmer로 작성한 사용자 프로그램을 실행하기 위한 오브젝트 코드가 저장됩니다.	저장됨.	저장됨.
FB 프로그램 영역*1	FB 프로그램 영역에는 CX-Programmer로 작성한 펑션 블록 정의가 저장됩니다.	저장됨.	저장됨.
파라미터 영역	파라미터 영역에는 PLC의 초기 설정이 저장됩니다.	저장됨.	저장됨.
PLC 이름	CPU 유닛의 이름이 저장되며, CX-Programmer가 잘못된 PLC에 접속하는 것을 방지하기 위해 CX-Programmer가 이를 읽어 조회할 수 있습니다.		
PLC Setup	PLC Setup에서는 소프트웨어 스위치를 사용하여 각종 초기 설정을 수행합니다. Section 9 PLC 설정을 참조하십시오.		
I/O 테이블	I/O 테이블은 사용자가 지정한 유닛의 장착 상태에 관한 정보를 제공합니다. Section 8 I/O 할당과 유닛 설정을 참조하십시오.		
라우팅 테이블	라우팅 테이블은 FINS 통신을 위한 네트워크 파라미터입니다. CX-Integrator를 사용하여 지정합니다. CX-Integrator 오퍼레이션 매뉴얼 (Cat. No. W464)을 참조하십시오.		
CPU 버스 유닛 설정	CPU 버스 유닛 설정에는 특정 CPU 버스 유닛의 초기 설정이 저장됩니다. Ethernet 유닛의 Ethernet 설정, Controller Link 유닛의 데이터 링크 파라미터 등의 설정이 포함됩니다.		
I/O 메모리 영역	I/O 메모리 영역은 사용자 프로그램에서 읽고 쓰는 데 사용됩니다. 용도에 따라 다음 영역으로 구분됩니다. CPU 유닛의 전원이 리셋될 때 데이터가 클리어되는 영역과 데이터가 유지되는 영역입니다. 다른 유닛과 데이터를 교환하는 영역과 내부적으로 사용되는 영역입니다.	---	저장
소스 및 코멘트 영역	소스 및 코멘트 영역은 CX-Programmer로 작성한 프로그램 소스 코드와 코멘트를 저장하는 데 사용됩니다.	저장	---
소스 코드	프로그램의 소스 코드(태스크 및 펑션 블록 내에서 래더, ST, SFC 언어로 작성한 것).		
심볼 테이블	심볼 테이블에는 CX-Programmer로 작성한 심볼(심볼명, 어드레스, I/O 코멘트)이 들어 있습니다.		
코멘트	코멘트는 CX-Programmer로 작성하며, 주석과 행 코멘트가 포함됩니다.		
프로그램 인덱스	프로그램 인덱스는 CX-Programmer로 작성한 프로그램 섹션에 관한 정보와 프로그램 코멘트를 제공합니다.		
네트워크 심볼(태그)*2	글로벌 심볼 테이블의 네트워크 심볼 데이터입니다.	저장	---

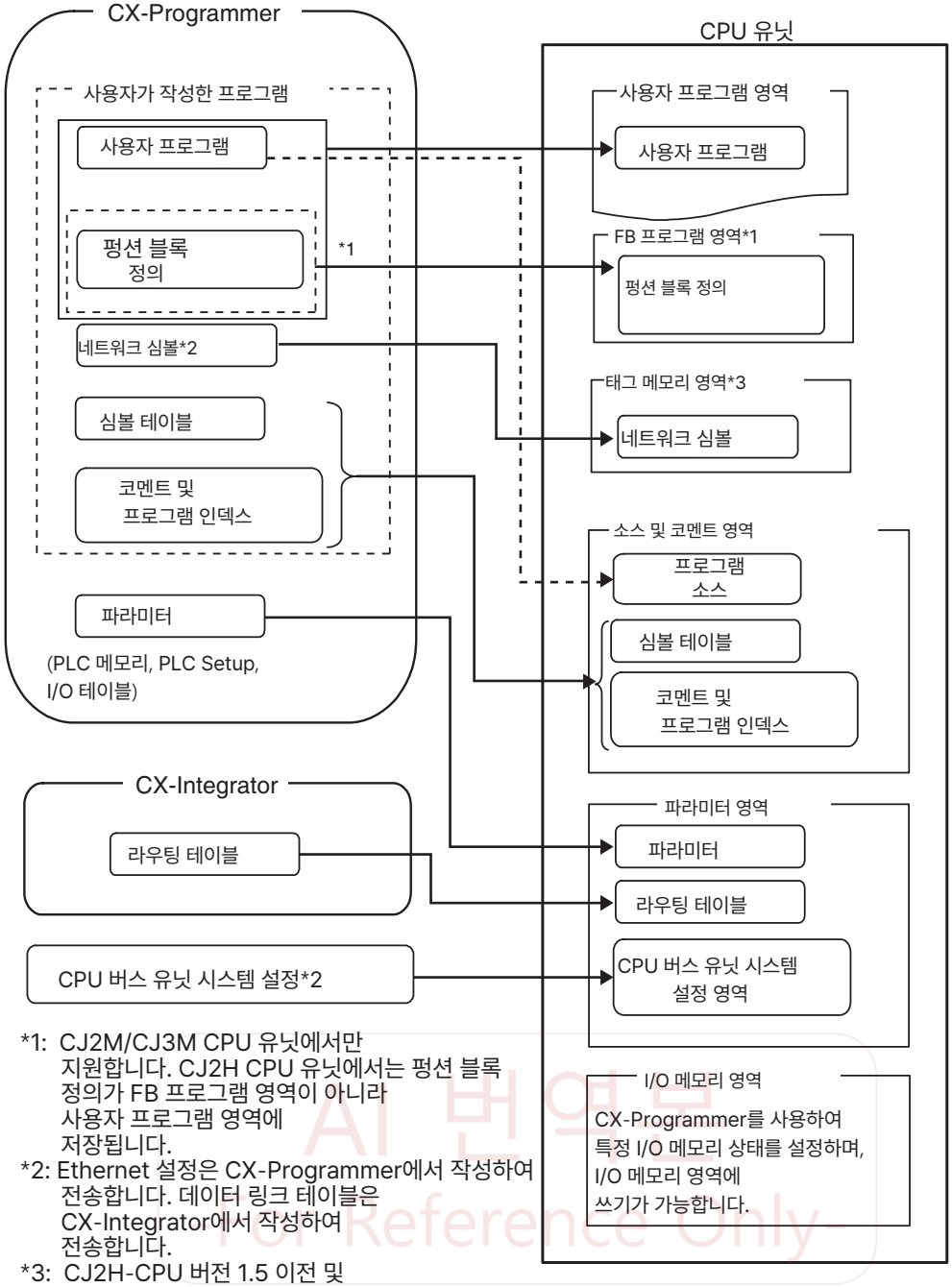
*1 CJ2M/CJ3M CPU 유닛에서만 지원합니다. CJ2H CPU 유닛에서는 펑션 블록 정의가 FB 프로그램 영역이 아니라 사용자 프로그램 영역에 저장됩니다.

*2 CJ2H-CPU□□ (-EIP 없음) 버전 1.5 이전 및 CJ2M-CPU1□ 버전 2.1 이전은 지원하지 않습니다.

사내 열람
-For Reference Only-

2-1-3 프로그래밍 디바이스에서 CPU 유닛으로의 데이터 전송

CX-Programmer 또는 CX-Integrator로 작성한 데이터는 다음 그림과 같이 CPU 유닛의 내부 메모리로 전송됩니다.



*1: CJ2M/CJ3M CPU 유닛에서만 지원합니다. CJ2H CPU 유닛에서는 펑션 블록 정의가 FB 프로그램 영역이 아니라 사용자 프로그램 영역에 저장됩니다.

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

*3: CJ2H-CPU 버전 1.5 이전 및 CJ2M-CPU 버전 2.1 이전은 지원하지 않습니다.

AI 번역본
For Reference Only

AI 번역본
-For Reference Only-

3

CPU 유닛의 동작

이 장에서는 CPU 유닛의 내부 동작과 CPU 유닛에 설정할 수 있는 운전 모드에 대해 설명합니다.

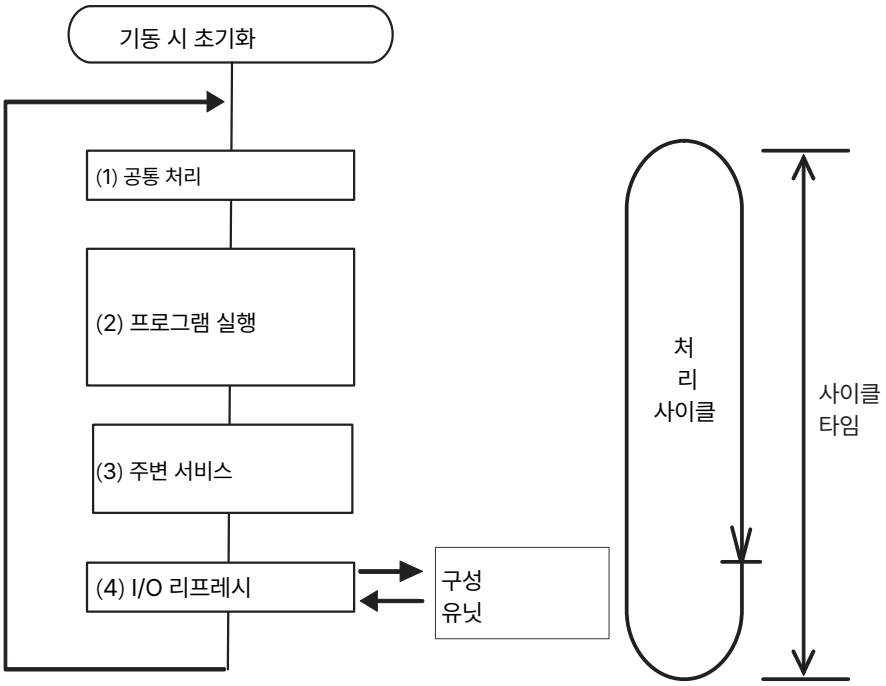
3-1 CPU 유닛의 내부 동작 3-2
3-1-1 개요 3-2
3-1-2 사이클 타임 3-4
3-1-3 정전 시 처리 3-7
3-2 CPU 유닛의 운전 모드 3-8
3-2-1 운전 모드 3-8
3-2-2 운전 모드 확인 3-9
3-2-3 운전 모드 변경 3-10
3-2-4 운전 모드 상세 3-14

AI 번역본
-For Reference Only-

3-1 CPU 유닛의 내부 동작

3-1-1 개요

CPU 유닛은 사용자 프로그램을 사이클릭하게 실행하면서 내부 I/O 메모리 영역에 데이터를 씁니다. I/O 리프레시와 주변 서비스 시에 외부와 데이터를 교환합니다. 다음 그림은 CPU 유닛의 내부 동작을 나타냅니다.



기동 시 초기화

PLC의 전원을 ON할 때마다 다음 처리가 한 번씩 수행됩니다.

- 접속된 유닛의 검출(I/O 할당)
- 등록된 I/O 테이블과 접속된 유닛의 대조
- IOM 홀드 비트의 상태에 따른 I/O 메모리 내 비유지 영역의 클리어
- 강제 상태 유지 비트의 상태에 따른 강제 상태의 클리어
- 메모리 카드가 삽입되어 있는 경우 메모리 카드의 자동 전송 파일을 사용한 오토부팅
- 자기진단(사용자 메모리 체크)의 실행
- 사용자 프로그램의 복원
- PLC Setup의 갱신

사 본 역 본
-For Reference Only-

처리 사이클

CPU 유닛은 기동 시 초기화 후 네 가지 처리(공통 처리, 프로그램 실행, I/O 리프레시, 주변 서비스)를 반복해서 수행합니다. 1사이클을 완료하는 데 걸리는 시간을 사이클 타임이라고 합니다.

- 공통 처리
  이 처리는 사용자 프로그램을 실행하기 위한 준비를 합니다. 처리에 필요한 하드웨어와 소프트웨어를 체크합니다. 공통 처리에 필요한 시간을 오버헤드 타임이라고 합니다.
- 프로그램 실행
  이 처리는 사용자 프로그램의 명령어를 실행합니다. 사용자 프로그램의 명령어는 운전 모드가 RUN 또는 MONITOR 모드일 때 실행됩니다. 명령어의 I/O 처리는 CPU 유닛의 I/O 메모리 영역에 있는 비트와 워드에 대해 수행됩니다.

- 주변 서비스
  이 처리는 프로그래밍 디바이스나 유닛으로부터 메시지가 요구될 때 발생합니다. CX-Programmer에서의 데이터 쓰기나 네트워크 통신 유닛용 명령어 처리가 수행됩니다.
- I/O 리프레시
  이 처리는 I/O 메모리 영역을 사용하여 유닛과 데이터를 교환합니다. 이 처리는 항상 프로그램 실행 후에 수행됩니다. I/O 리프레시는 중단 없이(즉, 타임 슬라이싱 없이) 현재 사이클 내에서 수행됩니다.

AI 번역본
-For Reference Only-

3-1-2 사이클 타임

사이클 타임은 공통 처리, 프로그램 실행, 주변 서비스, I/O 리프레시에 걸리는 시간의 합계입니다. 사이클 타임은 수행되는 처리에 따라 보통 사이클마다 변동합니다. 이들 처리는 각각 사이클 타임 내에 수행됩니다. 사이클 타임 계산 방법에 대해서는 Section 12 CPU 유닛의 사이클 타임을 참조하십시오.



참고

PLC Setup에서 최소 사이클 타임을 설정하면 사이클 타임을 일정하게 만들 수 있습니다. 최소 사이클 타임은 통상의 사이클 타임보다 길어야 합니다. 최소 사이클 타임의 설정 방법에 대해서는 10-2-1 최소 사이클 타임을 참조하십시오.

공통 처리

오버헤드 타임은 매 사이클의 공통 처리에서 발생합니다. 다음 항목이 체크됩니다.

- 배터리 에러
- 메모리 카드
- 딥 스위치
- I/O 버스
- 사용자 프로그램 메모리

프로그램 실행

사용자 프로그램의 명령어 처리 시간입니다.

- 처리 시간에는 READY 상태인 사이클릭 태스크의 명령어와 인터럽트 조건이 충족된 인터럽트 태스크의 명령어가 포함됩니다. 프로그램에서 점프된 명령어는 포함되지 않습니다.

- 처리 시간은 명령어의 개수와 사용되는 명령어의 종류에 따라 달라집니다. 또한 지정되는 오퍼랜드(예: 처리할 데이터의 양)에 따라서도 달라집니다.

- RUN 또는 MONITOR 모드에서는 실행 조건이 OFF인 명령어에도 내부 처리 시간이 필요합니다. 다음의 경우에는 내부 처리가 수행되지 않으므로 처리 시간이 필요하지 않습니다.

  - WAIT 상태인 사이클릭 태스크
  - JMP(004), CJP(510), CJPN(511) 명령어에 의해 점프되는 명령어.

AI 번역본
-For Reference Only-



참고

사이클 타임을 단축해야 하는 경우에는 다음과 같은 프로그래밍을 검토하십시오.

- 프로그램을 사이클릭 태스크로 분할하고, 실행할 필요가 없는 태스크는 WAIT 상태로 둡니다.

- 실행할 필요가 없는 명령어는 JMP(004), CJP(510), CJPN(511) 명령어를 사용하여 점프시킵니다.

주변 서비스

주변 서비스는 외부 기기에 대한 부정기 이벤트를 서비스하는 것입니다. 여기에는 외부 기기로부터의 이벤트와 외부 기기에 대한 서비스 요구가 모두 포함됩니다.

CJ 시리즈 PLC의 주변 서비스는 대부분 FINS 커맨드와 관련됩니다. 시스템에서 지정된 시간이 각 서비스 종류에 할당되어 매 사이클 실행됩니다. 할당된 시간 내에 모든 서비스를 완료할 수 없는 경우, 남은 서비스는 다음 사이클에 수행됩니다. 주변 서비스가 너무 많이 지연되는 경우에는 CX-Programmer를 사용하여 PLC Setup의 Fixed Servicing Time을 10%보다 큰 값으로 늘리십시오.

● 주변 서비스 시간

주변 서비스 시간의 디폴트 설정에서는 직전 사이클 타임의 10%가 전체 주변 서비스에 할당됩니다. 따라서 주변 서비스 시간은 직전 사이클 타임에 따라 달라집니다.

주변 서비스 완료에 너무 많은 사이클이 필요해 서비스가 지연되는 경우에는 CX-Programmer에서 PLC Setup의 Fixed Servicing Time을 사용하여 사이클 타임의 10%보다 긴 시간을 설정하십시오. 서비스 시간을 늘리면 사이클 타임도 늘어난다는 점에 유의하십시오.

● 주변 서비스의 종류

유닛	서비스
특수 I/O 유닛의 이벤트 서비스 CPU 버스 유닛의 이벤트 서비스	· 특수 I/O 유닛 및 CPU 버스 유닛으로부터의 FINS 커맨드에 대한 부정기 서비스(예: 외부 인터럽트 태스크의 기동 요구) · CPU 유닛에서 위 유닛으로 보내는 FINS 커맨드에 대한 부정기 서비스.
페리페럴 USB 포트 서비스 시리얼 포트 서비스	· 프로그래밍 디바이스, PT, 호스트 컴퓨터로부터 페리페럴 USB 포트 또는 시리얼 포트를 경유하여 수신한 FINS 또는 Host Link 커맨드에 대한 부정기 서비스(예: 프로그램 전송, 모니터링, 강제 세트/리셋 조작, 온라인 에디팅의 요구) · CPU 유닛에서 페리페럴 USB 포트 또는 시리얼 포트를 통해 송신하는 부정기 서비스(논솔리시티드 통신)
통신 포트 서비스	· 통신 포트(내부 로직 포트)를 사용하여 SEND(090)/SEND2(491), RECV(098)/RECV2(492), CMND(490)/CMND2(493), PMCR(260)/PMCR2(264) 등의 명령어에 대한 네트워크 통신, 시리얼 통신 또는 파일 메모리 액세스를 실행하는 서비스. · 통신 포트를 사용하여 백그라운드 실행을 수행하는 서비스.
파일 액세스 서비스	메모리 카드 또는 EM 파일 메모리에 대한 파일 읽기/쓰기 조작.
온라인 에디팅 서비스	온라인 에디팅이 처리 중일 때 온라인 에디팅으로 편집한 내용을 내장 플래시 메모리에 백업하는 서비스.
백업 서비스	프로그램이나 파라미터에 변경이 있을 때 변경된 내용을 내장 플래시 메모리에 백업하는 서비스.

-For Reference Only-

참고

서비스 시간은 특수 I/O 유닛, CPU 버스 유닛, 페리페럴 USB 포트, 시리얼 포트, 파일 액세스, 통신 포트에 할당됩니다.
서비스가 여러 사이클에 걸쳐 나뉘어 완료가 지연되는 경우에는 PLC Setup의 실행 시간 설정에서 백분율이 아니라 동일한 할당 시간(모든 서비스에 같은 시간)을 설정하십시오.

참고

사이클 타임이 길어 CPU 유닛으로부터의 응답 시간이 CX-Programmer의 응답 타임아웃 시간을 초과하면 CX-Programmer에 접속하지 못할 수 있습니다. CX-Programmer에 접속할 수 없는 경우에는 CX-Programmer에서 응답 타임아웃 시간의 설정을 늘리십시오. CX-Programmer에서 PLC를 오른쪽 클릭하고 Change PLC를 선택합니다. Change PLC 대화 상자가 표시됩니다. 네트워크 종류의 Setting 버튼을 클릭하고 Network 탭 페이지에서 응답 타임아웃 시간을 늘리십시오.

I/O 리프레시

I/O 유닛의 처리 시간은 PLC에서 사용되는 구성 유닛의 종류와 개수에 따라 달라집니다. 각 유닛의 처리 시간은 일정합니다.

특수 I/O 유닛의 사이클릭 I/O 리프레시를 정지하면 처리 시간을 단축할 수 있습니다. 특수 I/O 유닛의 I/O 리프레시를 정지하려면 CX-Programmer의 SIOU Refresh 탭 페이지에서 파라미터를 설정하십시오. 각 유닛의 I/O 리프레시용 데이터 교환 영역은 다음과 같습니다.

유닛			최대 데이터 교환량	데이터 교환 영역
기본 I/O 유닛			유닛에 따라 다름.	I/O 비트 영역
특수 I/O 유닛	CIO 영역에 할당된 워드		10워드/유닛 (유닛에 따라 다름.)	특수 I/O 유닛 영역
	DM 영역에 할당된 워드		100워드/유닛	특수 I/O 유닛 워드
	유닛 고유 데이터		유닛에 따라 다름.	유닛에 따라 다름.
CPU 버스 유닛	CIO 영역에 할당된 워드		25워드/유닛	CPU 버스 유닛 영역
	DM 영역 할당 워드		100 워드/유닛	CPU 버스 유닛 워드
	유닛 고 유 데이터 (오른쪽을 참조하십시오.)	내장 EtherNet/IP 포트* EtherNet/IP 유닛	사용하는 EtherNet/IP 기능에 따라 다릅니다.	태그 데이터 링크 영역
		Controller Link 유닛 및 SYSMAC LINK 유닛	유닛에 따라 다릅니다.	데이터 링크용으로 설정된 워드(고정 할당 또는 사용자 설정 할당)
		DeviceNet 유닛	유닛에 따라 다릅니다.	리모트 I/O 통신용으로 설정된 워드 (고정 할당 또는 사용자 설정 할당)
		시리얼 통신 유닛	프로토콜 매크로에 따라 다릅니다.	프로토콜 매크로용으로 설정된 통신 데이터
		이더넷 유닛	유닛에 따라 다릅니다.	특정 제어 비트 조작으로 개시되는 소켓 서비스용 통신 데이터.

* CJ2H-CPU6□-EIP, CJ2M-CPU3□, CJ3M-CPU3□에서만 지원합니다.

AI 번역본
-For Reference Only-

참고

통상의 I/O 리프레시 기간이 아니라 명령어가 실행될 때 유닛과 데이터를 교환할 수 있습니다. 이것을 즉시 리프레시라고 합니다. 즉시 리프레시는 일부 명령어에 느낌표(!)를 붙이거나 IORF(097), FIORF(225), DLNK(226) 명령어를 사용하여 실행할 수 있습니다.

다음 명령어를 사용하면 다이렉트 처리에 의해 특정 특수 I/O 유닛 또는 CPU 버스 유닛과의 데이터 교환 속도를 높일 수 있습니다. 자세한 내용은 명령어 레퍼런스 매뉴얼(Cat. No. W474)을 참조하십시오.

- ANALOG INPUT DIRECT CONVERSION (AIDC(216)) (CJ1W-AD042 고속 아날로그 입력 유닛용)*1
- ANALOG OUTPUT DIRECT CONVERSION (AODC(217)) (CJ1W-DA042V 고속 아날로그 출력 유닛용)*1
- DIRECT TRANSMIT VIA SERIAL COMMUNICATIONS UNIT/SERIAL PORT (DTXDU(262)) (CJ1W-SCU□2 전용)*1
- DIRECT RECEIVE VIA SERIAL COMMUNICATIONS UNIT/SERIAL PORT (DRXDU(261)) (CJ1W-SCU□2 전용)*1
- PCU HIGH-SPEED POSITIONING (NCDMV(218)) (CJ1W-NC□□4 및 CJ1W-NC□ 전용)*2
- PCU POSITIONING TRIGGER (NCDTR(219)) (CJ1W-NC□8□ 전용)*2

*1 CJ2H-CPU 버전 1.0 이전은 지원되지 않습니다.

*2 CJ2H-CPU 버전 1.2 이전은 지원되지 않습니다.

3-1-3 정전 시 처리

CPU 유닛이 동작 중일 때 전원이 차단되고 그 차단이 정전으로 확정되면, 현재 실행 중인 명령어가 완료된 후 다음의 정전 처리가 수행됩니다.

정전 시 처리는 전원 OFF 인터럽트 태스크의 설정에 따라 수행됩니다.*1

- 전원 OFF 인터럽트 태스크가 유효한 경우, 해당 태스크가 실행된 후 CPU 유닛이 즉시 리셋됩니다.

- 전원 OFF 인터럽트 태스크가 유효하지 않은 경우, CPU 유닛이 즉시 리셋*2됩니다.

*1 전원 OFF 인터럽트 태스크는 정전으로 인해 CPU 유닛이 리셋되기 직전에 실행됩니다.

*2 동작 중에 전원이 차단된 경우, 지정한 범위의 명령어가 실행된 후에 정전 처리를 수행하도록 할 수 있습니다.

AI 번역본
For Reference Only

AC 전원의 경우 10 ms 미만, DC 전원의 경우 2 ms 미만의 순간 정전에서는 동작이 항상 계속됩니다. 전원 사정이 좋지 않은 경우에는 정전이 검출된 시점부터 정전으로 확정될 때까지의 시간을 길게 할 수 있습니다.

정전 시 처리에 대해서는 A-5 정전 시 동작을 참조하십시오.

3-2 CPU 유닛의 동작 모드

3-2-1 동작 모드

동작 모드를 설정하여 CPU 유닛의 동작 조건과 CPU 유닛에서 설정이 가능한지 여부를 제어할 수 있습니다. 동작 모드에는 세 가지가 있습니다.

- **RUN 모드:**

  RUN 모드는 시스템의 실제 동작에 사용되며 가장 빠른 동작을 제공합니다.
  프로그램이 실행됩니다.
  비트의 강제 세트/리셋, I/O 메모리 값의 변경, 온라인 에디팅은 할 수 없습니다.

- **MONITOR 모드:**

  MONITOR 모드는 시운전 및 조정용입니다.
  프로그램이 실행됩니다.
  비트의 강제 세트/리셋, I/O 메모리 값의 변경, 온라인 에디팅이 가능합니다.

- **PROGRAM 모드:**

  PROGRAM 모드는 프로그램과 PLC 설정의 전송 및 I/O 테이블 작성용입니다.

  프로그램은 실행되지 않습니다.

동작 모드는 CPU 유닛에 접속한 CX-Programmer를 사용하여 변경하십시오.

AI 번역본
-For Reference Only-

3-2-2 동작 모드 확인

CPU 유닛의 전면 표시등

CPU 유닛 전면의 RUN 표시등은 아래와 같이 동작 모드를 나타냅니다.

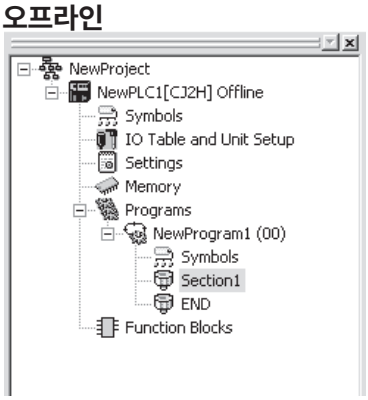
동작 모드	CPU 유닛의 RUN 표시등	비고
PROGRAM 모드	OFF	---
RUN 또는 MONITOR 모드	녹색 점등	RUN 모드인지 MONITOR 모드인지는 CX-Programmer로 확인하십시오.

CX-Programmer

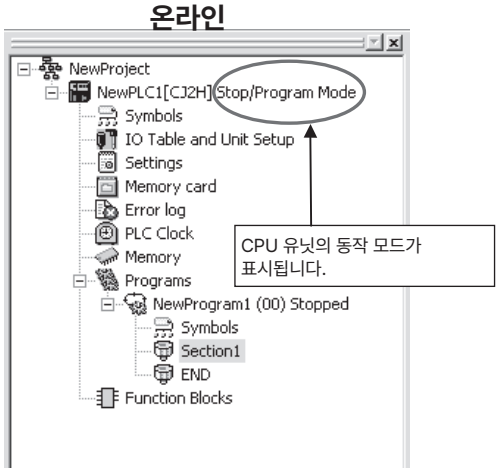
CX-Programmer의 프로젝트 트리 또는 스테이터스 바에서 동작 모드를 확인할 수 있습니다.

● **프로젝트 트리**

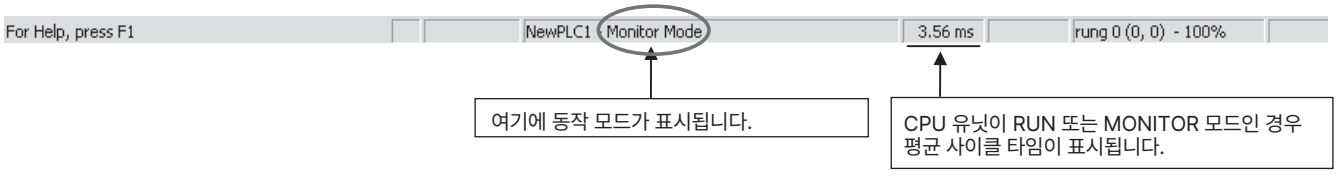
오프라인



온라인



● **스테이터스 바**



참고

-For Reference Only-

CJ1W-PA205R 전원 유닛을 사용하면 외부 RUN 출력을 얻을 수 있습니다. CJ1W-PA205R의 RUN 출력(접점 출력)은 치명 에러가 없는 한 RUN 또는 MONITOR 모드에서 ON됩니다.

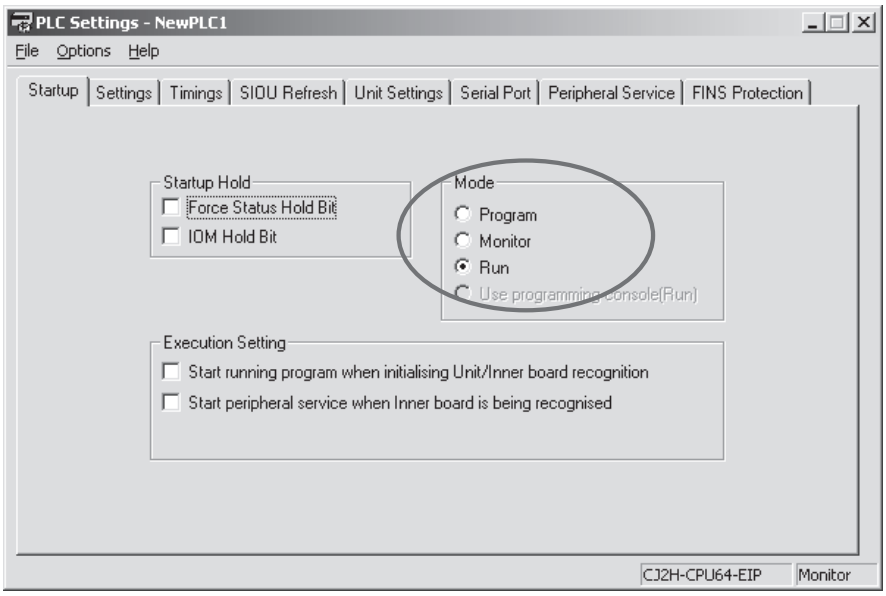
3-2-3 동작 모드 변경

동작 모드 변경

동작 모드는 CX-Programmer에서 변경할 수 있습니다.

● 기동 모드 변경

CPU 유닛을 ON했을 때의 기본 동작 모드는 RUN 모드입니다. 기동 모드를 PROGRAM 또는 MONITOR 모드로 변경하려면 CX-Programmer에서 PLC 설정의 기동 설정에 원하는 모드를 설정하십시오.



● 기동 후 동작 모드 변경

다음 중 한 가지 방법을 사용하십시오.

- 동작 모드 메뉴에서 PROGRAM, MONITOR 또는 RUN을 선택합니다.
- 프로젝트 트리에서 PLC를 오른쪽 클릭한 후 동작 모드 메뉴에서 PROGRAM, MONITOR 또는 RUN을 선택합니다.

AI 번역본
-For Reference Only-

모드 변경 및 정전 후의 메모리 상태

CIO 영역이나 워크 영역 등 I/O 메모리의 일부는 RUN 또는 MONITOR 모드와 PROGRAM 모드 사이에서 동작 모드가 변경될 때 클리어됩니다. 이들을 비유지 영역이라고 합니다. 동작 모드가 변경되어도 이들 영역의 내용을 유지하려면 보조 기억 영역의 IOM 유지 비트(A500.12)를 ON하십시오.

모드 변경	비유지 영역	유지 영역
	· I/O 영역 · 데이터 링크 영역 · CPU 버스 유닛 영역 · 특수 I/O 유닛 영역 · 워크 영역 · 타이머 PV 및 완료 플래그 · 인덱스 레지스터 · 데이터 레지스터 · 태스크 플래그 (보조 기억 영역의 비트/워드는 어드레스에 따라 유지되거나 유지되지 않습니다.)	· HR 영역 · DM 영역 · EM 영역 · 카운터 PV 및 완료 플래그 (보조 기억 영역의 비트/워드는 어드레스에 따라 유지되거나 유지되지 않습니다.)
RUN 또는 MONITOR에서 PROGRAM으로 **PROGRAM에서 RUN 또는 MONITOR로**	클리어*1	유지
RUN에서 MONITOR로 또는 MONITOR에서 RUN으로	유지	유지
정전 복귀*2	클리어*3	유지

*1 IOM 유지 비트가 OFF인 경우 메모리가 클리어됩니다. ON인 경우 데이터는 다음과 같이 유지됩니다:

I/O 메모리 유지 비트 상태 (A500.12)	I/O 메모리			출력 유닛에 할당된 출력 비트		
	모드 변경 PROGRAM과 RUN/MONITOR 간	치명 에러		모드 변경 PROGRAM과 RUN/MONITOR 간	치명 에러	
		FALS 이외의 치명 에러	FALS 실행		FALS 이외의 치명 에러	FALS 실행
OFF	클리어	클리어	유지	OFF	OFF	OFF
ON	유지	유지	유지	유지	OFF: 치명 에러가 발생하면 CPU 유닛의 I/O 메모리 값은 유지되지만, 기본 출력 유닛의 출력은 OFF됩니다.	

AI 번역본
-For Reference Only-

*2 IOM 유지 비트(A500.12)는 정전이 복귀되면 OFF됩니다. 그 상태를 유지하려면 CX-Programmer로 PLC 설정의 기동 시 유지 설정 탭 페이지에서 IOM 유지 비트 체크 박스를 선택한 후 PLC 설정을 전송하십시오.

*3 태스크 플래그는 IOM 유지 비트(A500.12)의 상태와 관계없이 클리어됩니다.

모드 변경 후 기본 출력 유닛의 출력 상태

기본 설정에서 RUN 또는 MONITOR 모드와 PROGRAM 모드 사이에서 동작 모드가 변경되면, 기본 출력 유닛의 출력은 모드가 변경된 시점의 I/O 메모리 상태에 따라 리프레시됩니다.

기본 출력 유닛의 출력 상태는 IOM 유지 비트(A500.12)가 ON인지 OFF인지에 따라 다음과 같이 됩니다.

- A500.12가 OFF이면 PROGRAM 모드로 변경될 때 출력 비트가 클리어되므로, 기본 출력 유닛의 출력은 OFF됩니다.
- A500.12가 ON이면 PROGRAM 모드로 변경될 때 출력 비트의 상태가 유지되므로, 기본 출력 유닛의 출력 상태도 유지됩니다.

사용상의 주의

IOM 유지 비트(A500.12)를 ON으로 설정할 때의 주의사항

RUN 또는 MONITOR 모드에서 PROGRAM 모드로 변경되면 출력의 I/O 메모리 상태는 클리어되지 않고(즉, OFF되지 않고) PROGRAM 모드로 변경되기 전의 상태가 유지됩니다. 따라서 PROGRAM 모드에서 다시 RUN 또는 MONITOR 모드로 변경하면 PROGRAM 모드로 변경되기 전의 I/O 메모리 값이 출력됩니다.

단, 치명 에러(FALS(007) 명령어의 실행 포함)로 인해 동작이 정지한 경우에는 CPU 유닛의 I/O 메모리 상태가 유지되더라도 기본 출력 유닛의 모든 출력이 OFF됩니다.

AI 번역본
-For Reference Only-

PROGRAM 모드에서의 기본 출력 유닛 출력 상태

기본 출력 유닛의 출력은 PROGRAM 모드에서도 I/O 메모리의 출력 비트 상태로 리프레시됩니다. 따라서 CX-Programmer 등의 지원 소프트웨어로 I/O 메모리의 출력 비트를 변경하면 그 상태가 기본 출력 유닛에서 출력됩니다.

⚠ 경고

CPU 유닛은 프로그램이 정지 중일 때에도(즉, PROGRAM 모드에서도) I/O를 리프레시합니다. 기본 I/O 유닛, 특수 I/O 유닛 또는 CPU 버스 유닛에 할당된 메모리의 어느 부분이든 그 상태를 변경하기 전에 반드시 안전을 충분히 확인하십시오. 유닛에 할당된 데이터를 변경하면 해당 유닛에 접속된 부하가 예기치 않게 동작할 수 있습니다. 다음 중 어느 조작이든 메모리 상태를 변경시킬 수 있습니다.

- 프로그래밍 디바이스에서 CPU 유닛으로 I/O 메모리 데이터를 전송
- 프로그래밍 디바이스에서 메모리의 현재값을 변경
- 프로그래밍 디바이스에서 비트를 강제 세트/리셋
- 메모리 카드 또는 EM 파일 메모리에서 CPU 유닛으로 I/O 메모리 파일을 전송
- 호스트 컴퓨터 또는 네트워크상의 다른 PLC에서 I/O 메모리를 전송

기본 출력 유닛의 출력을 확실히 OFF 상태로 유지하려면 출력 OFF 비트(A500.15)를 ON하십시오.(주 참조) 모든 출력이 OFF일 때(즉, 출력 OFF 비트가 ON일 때) CPU 유닛 전면의 INH 표시등이 점등합니다.

주 출력 OFF 비트(A500.15)의 상태는 동작 모드가 변경되어도, 그리고 전원이 OFF · ON되어도 유지됩니다. 즉, 출력은 OFF 상태로 유지됩니다.

AI 번역본
-For Reference Only-

3-2-4 동작 모드 상세

다음 표는 각 동작 모드에서의 상태를 나타냅니다.

<table>
<tr><th colspan="2">동작 모드</th><th>PROGRAM</th><th>MONITOR</th><th>RUN</th></tr>
<tr><td colspan="2">용도</td><td>동작 정지 및 프로그램 전송</td><td>동작 테스트 및 조정</td><td>주 동작</td></tr>
<tr><td colspan="2">프로그램 상태</td><td>정지</td><td>실행</td><td>실행</td></tr>
<tr><td colspan="2">RUN 표시등</td><td>OFF</td><td>ON</td><td>ON</td></tr>
<tr><td rowspan="8">CX-Programmer를 사용한 조작</td><td>I/O 테이블 작성</td><td rowspan="4">가능</td><td colspan="2" rowspan="4">불가능</td></tr>
<tr><td>PLC 설정 전송</td></tr>
<tr><td>프로그램 다운로드</td></tr>
<tr><td>프로그램 체크</td></tr>
<tr><td>온라인 에디팅</td><td colspan="2" rowspan="4">가능</td><td rowspan="4">불가능</td></tr>
<tr><td>강제 세트/리셋에 의한 배선 체크</td></tr>
<tr><td>I/O 메모리 PV 변경</td></tr>
<tr><td>타이머/카운터 PV 및 SV 변경</td></tr>
<tr><td>프로그램 업로드</td><td colspan="3" rowspan="3">가능</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">동작 개시 시 기동하도록 설정된 태스크와 TASK ON 명령어로 기동되는 태스크가 실행됩니다. 그 밖의 태스크는 정지합니다.</td></tr>
<tr><td colspan="2">인터럽트 태스크</td><td>정지</td><td colspan="2">인터럽트 조건이 성립하면 실행됩니다.</td></tr>
<tr><td rowspan="2">기본 출력 유닛에 할당된 출력 비트의 출력*1</td><td>IOM 유지 비트 OFF</td><td>PROGRAM 모드로 변경된 직후 OFF됩니다.*2</td><td rowspan="2">프로그램에 따라 다름</td><td rowspan="2">프로그램에 따라 다름</td></tr>
<tr><td>IOM 유지 비트 ON</td><td>PROGRAM 모드로 변경된 후 상태가 유지됩니다.*2</td></tr>
</table>

*1 출력 OFF 비트(A500.15)가 ON이면 동작 모드와 I/O 메모리 상태에 관계없이 기본 출력 유닛의 출력이 OFF됩니다. 전원을 ON해도 출력은 OFF 상태로 유지됩니다.

*2 출력 유닛의 출력은 PROGRAM 모드에서도 지원 소프트웨어나 PT로 메모리 상태를 변경하면 리프레시됩니다.

사용 안내 For Reference Only-

4

CPU 유닛의 초기화

이 섹션에서는 기동 시 CPU 유닛에 대해 수행되는 초기화 처리에 대해 설명합니다.

4-1 CPU 유닛 초기화 개요 4-2
4-1-1 CPU 유닛의 초기 설정 4-2
4-2 PLC 설정 4-8
4-3 I/O 테이블 작성 4-9
4-3-1 I/O 테이블 4-9
4-3-2 자동 할당 4-10
4-3-3 수동 할당 4-10
4-4 라우팅 테이블 설정 4-11
4-4-1 라우팅 테이블 4-11
4-4-2 라우팅 테이블이 필요한 경우 4-13
4-4-3 라우팅 테이블 설정과 전송 4-14
4-5 특수 I/O 유닛 및 CPU 버스 유닛의 DM 영역 할당 워드 설정 4-15
4-5-1 특수 I/O 유닛 및 CPU 버스 유닛의 DM 영역 할당 워드 설정 4-15
4-5-2 설정 순서 4-15
4-6 CPU 버스 유닛 셋업 영역 4-16
4-6-1 CPU 버스 유닛 셋업 영역 4-16
4-6-2 설정 순서 4-16

AI 번역본
-For Reference Only-

4-1 CPU 유닛 초기화 개요

4-1-1 CPU 유닛의 초기 설정

CPU 유닛에 필요한 초기 설정을 하려면 CPU 유닛 전면의 딥 스위치 및 로터리 스위치에 의한 하드웨어 설정과 CX-Programmer 등의 소프트웨어로 파라미터를 설정하는 소프트웨어 설정을 모두 수행해야 합니다.

하드웨어 설정

● 딥 스위치

용도에 따라 딥 스위치를 설정하십시오.

위치	용도	설정 방법
CPU 유닛 전면의 딥 스위치	주요 용도: · 프로그램 쓰기 방지: 핀 1을 ON합니다. · 기동 시 자동 전송: 핀 2를 ON합니다. · RS-232C에 의한 Toolbus 접속: 핀 5를 ON합니다. · 사용자 정의 핀: 핀 6을 ON합니다. · 간이 백업: 핀 7을 ON합니다.	전원을 OFF한 상태에서 설정을 수행한 후 전원을 ON하십시오.

● 로터리 스위치

CJ2H-CPU6□-EIP, CJ2M-CPU3□ 및 CJ3M-CPU3□의 내장 EtherNet/IP 포트에는 CPU 버스 유닛과 동일한 방식으로 유닛 번호와 노드 주소가 할당됩니다. 유닛 번호 설정에 따라 CIO 영역과 DM 영역의 CPU 버스 유닛 영역에서 내장 EtherNet/IP 포트에 워드가 할당됩니다.

하드웨어 설정에 대한 자세한 내용은 CJ2 CPU 유닛 하드웨어 사용자 매뉴얼 (Cat. No. W472), CJ3 CPU 유닛 하드웨어 사용자 매뉴얼(Cat. No. W664)의 3-1 CPU 유닛을 참조하십시오.

AI 번역본
-For Reference Only-

소프트웨어 설정

다음 표는 해당 유닛의 소프트웨어 설정 용도와 설정 방법을 나타냅니다.

유닛	설정		용도	설정 방법	프로그래밍 디바이스로 작성한 파일	백업 위치
CPU 유닛	파라미터 영역	PLC 이름	CPU 유닛에 대한 오접속 방지	CX-Programmer	CX-Programmer 프로젝트 파일(.CXP)	내장 플래시 메모리
		PLC 설정	기본값 이외의 사양 사용			
		등록 I/O 테이블	오접속 확인과 자동 할당이 필요한 경우, 또는 향후 할당을 위한 워드 예약 및 그 밖의 용도별 설정이 필요한 경우			
		라우팅 테이블	2대 이상의 FINS 네트워크 통신 유닛을 접속할 때 필요합니다 (*1)	CX-Integrator	라우팅 테이블 파일 (.RTG)	
이더넷 유닛		CPU 버스 유닛 셋업 영역 (CPU 버스 유닛 셋업)(*2)	이더넷 유닛 사용 시	CX-Programmer	CX-Programmer 프로젝트 파일(.CXP)	내장 플래시 메모리
Controller Link 유닛(데이터 링크 테이블)			Controller Link에서 사용자 설정 데이터 링크 사용 시	CX-Integrator	데이터 링크 테이블 (.CL2)	
FL-net 유닛			FL-net 유닛 사용 시	CX-FLnet (FL-net 지원 소프트웨어)	FL-net 지원 소프트웨어 설정 파일 (.CSV)	
특수 I/O 유닛 및 CPU 버스 유닛	I/O 메모리	특수 I/O 유닛 또는 CPU 버스 유닛에 대한 DM 영역 워드 할당 (*2)	특수 I/O 유닛 또는 CPU 버스 유닛 사용 시	· CX-Programmer · 사용자 프로그램	CX-Programmer 프로젝트 파일(.CXP)	내장 RAM(배터리 백업)

*1 다음은 FINS 네트워크 통신 유닛으로 분류됩니다.

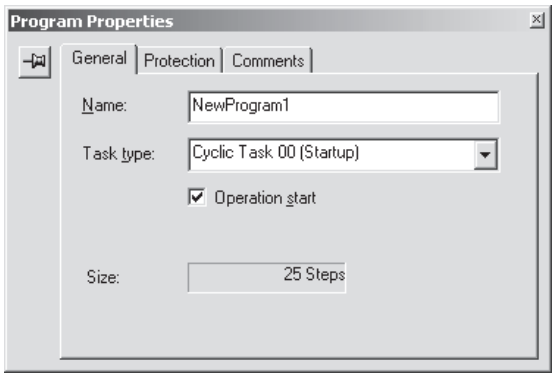
- CPU 버스 유닛으로 분류되는 네트워크 통신 유닛:
  Controller Link 유닛, SYSMAC LINK 유닛, 이더넷 유닛, DeviceNet 유닛, FL-net 유닛
- CJ2H-CPU6□-EIP, CJ2M-CPU3□, CJ3M-CPU3□ 또는 CJ3M-CPU3□의 내장 EtherNet/IP 포트를 FINS 네트워크 통신에 사용하는 경우
- 시리얼 통신 유닛으로 시리얼 게이트웨이를 사용하고 라우팅 테이블을 사용하는 경우.
  라우팅 테이블은 특수 I/O 유닛용 네트워크 통신 유닛(예: CompoNet 마스터 유닛)에는 적용되지 않습니다.

*2 위에 제시된 특수 I/O 유닛 또는 CPU 버스 유닛의 초기 설정은 CPU 유닛에 저장됩니다. 특수 I/O 유닛 및 CPU 버스 유닛에 저장되는 데이터와 프로그램은 해당 유닛 전용 지원 소프트웨어로 별도로 작성한 후 CPU 유닛을 통해 특수 I/O 유닛 및 CPU 버스 유닛으로 전송합니다.

For Reference Only

참고

프로그램 태스크 설정
프로그램을 태스크에 할당하기 위한 초기 설정은 CX-Programmer를 사용하여 다음 프로그램 속성에서 설정합니다.



● PLC 이름

사용자가 CPU 유닛에 대해 설정하는 이름입니다. CX-Programmer의 PLC 메뉴에서 PLC Info – PLC Info를 선택하여 설정하십시오.

CX-Programmer가 온라인일 때 PLC에 등록된 이름과 프로젝트의 PLC 이름이 일치하는지 시스템이 확인합니다.

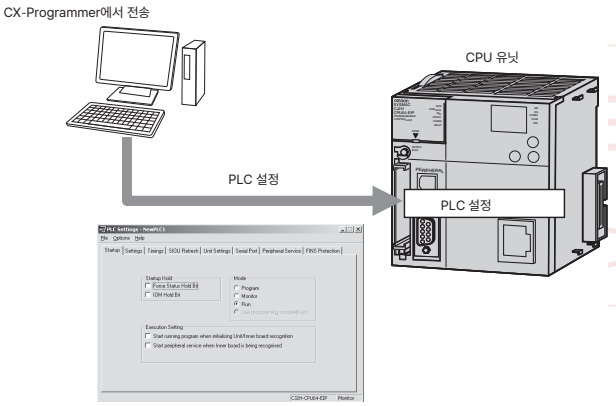
● PLC 설정

PLC 설정은 CPU 유닛을 기본값 이외의 사양으로 사용하기 위한 변경에 사용됩니다. 다음 설정은 CPU 유닛의 기본값 예입니다.

기동 모드: RUN 모드
RS-232C 포트: 호스트 링크
고정 서비스 시간: 사이클 타임의 10%

이들 기본값 이외의 사양을 사용하려면 CX-Programmer로 PLC 설정을 변경하고, PLC 설정을 CPU 유닛으로 전송하십시오.

⇒ CX-Programmer에서 PLC 설정 전송



견본
ce Only-

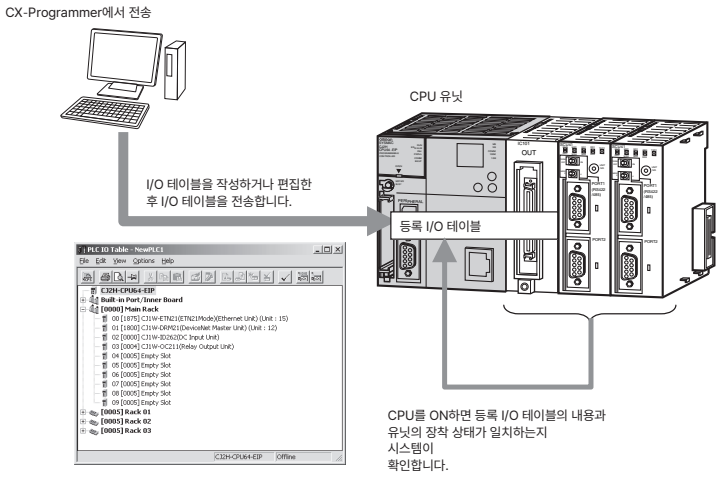
● I/O 테이블

유닛 슬롯을 수동으로 설정할 때 유닛의 오접속을 검출하거나 CPU 유닛에서 I/O를 수동으로 할당하려면 I/O 테이블을 작성하십시오. 기본값에서는 전원을 ON할 때마다 유닛이 접속된 순서대로 CPU 유닛의 I/O가 자동으로 할당됩니다.

CX-Programmer에서 다음 조작 중 하나를 사용하여 I/O 테이블을 작성하십시오.

・온라인: 유닛이 접속된 상태에서 I/O 테이블 작성 순서를 수행합니다.

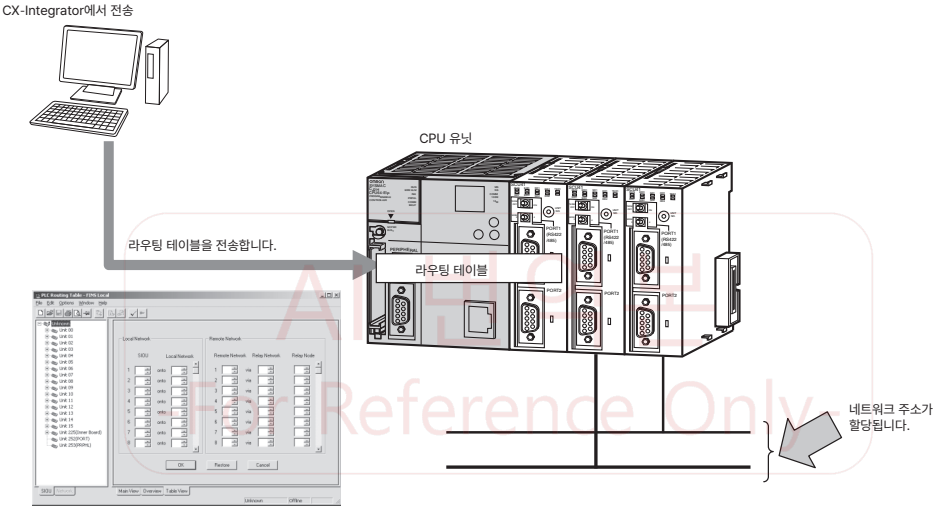
・오프라인: I/O 테이블을 편집한 후(유닛 장착 여부와 관계없이) 해당 테이블을 CPU 유닛으로 전송합니다.



● 라우팅 테이블

CPU 유닛에 FINS 네트워크 통신 유닛(CPU 버스 유닛)이 2대 이상 장착된 경우에는 CX-Integrator로 라우팅 테이블을 작성해야 합니다. 로컬 네트워크 테이블에서는 각 FINS 네트워크 통신 유닛에 네트워크 주소가 할당됩니다.

CX-Integrator로 로컬 네트워크 테이블을 설정하고 CPU 유닛으로 전송하십시오.



For Reference Only

● CPU 버스 유닛 셋업 영역

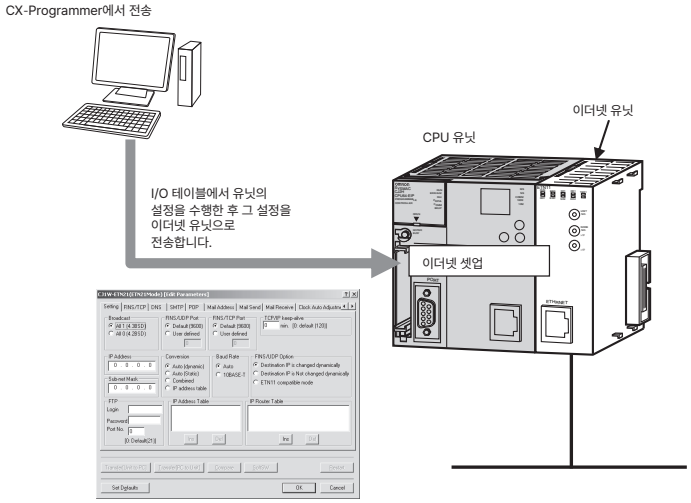
이더넷 유닛, Controller Link 유닛, FL-net 유닛 등 특정 CPU 버스 유닛을 사용하는 경우에는 각 유닛의 고유 설정을 수행하여 CPU 유닛으로 전송해야 합니다.

예: 이더넷 유닛 사용 시

CX-Programmer를 사용하여 이더넷 설정을 수행하십시오.

이더넷 설정을 수행하려면 CX-Programmer의 I/O 테이블에서 이더넷 유닛을 오른쪽 클릭한 후 CPU 버스 유닛 설정 편집을 선택하십시오.

CX-Programmer의 I/O 테이블에서 이더넷 유닛의 설정을 수행한 후 해당 I/O 테이블을 CPU 유닛으로 전송하십시오.



참고

Controller Link 유닛에서 사용자 설정 데이터 링크 테이블을 사용하려면 데이터 링크 테이블을 설정한 후 CPU 유닛에 저장하십시오.

AI 번역본
-For Reference Only-

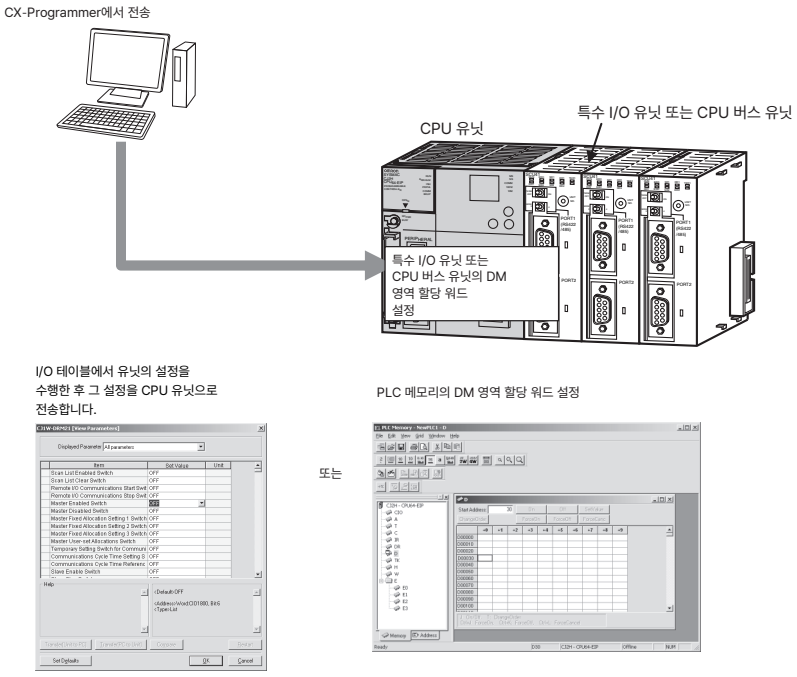
● 특수 I/O 유닛 및 CPU 버스 유닛의 DM 영역 워드 할당

특수 I/O 유닛 또는 CPU 버스 유닛을 사용하는 경우에는 DM 영역에 할당된 워드의 설정을 수행한 후 그 설정을 CPU 유닛으로 전송하십시오.

CX-Programmer로 DM 영역 할당 워드를 설정하려면 다음 방법 중 하나를 사용하십시오.

- I/O 테이블에서 각 특수 I/O 유닛 또는 CPU 버스 유닛을 오른쪽 클릭한 후 CPU 버스 유닛 설정 편집을 선택하여 DM 영역 할당 워드를 설정합니다.
- PLC 메모리에서 DM 영역 할당 워드를 설정합니다.

CX-Programmer로 각 특수 I/O 유닛 또는 CPU 버스 유닛의 DM 영역 할당 워드를 설정한 후 그 설정을 CPU 유닛으로 전송하십시오.



AI 번역본
-For Reference Only-

4-2 PLC 설정

PLC 설정에는 CPU 유닛의 기본 설정이 들어 있습니다. CPU 유닛을 기본값이 아닌 사양으로 사용하려면 PLC 설정의 파라미터를 변경해야 합니다. PLC 설정의 파라미터는 CX-Programmer를 사용하여 설정합니다.

PLC 설정에 대한 자세한 내용은 9장 PLC 설정을 참조하십시오.

AI 번역본
-For Reference Only-

4-3 I/O 테이블 작성

I/O 테이블은 다음의 경우에만 작성해야 합니다. I/O 테이블은 CX-Programmer를 사용하여 작성합니다.

・오접속을 확인하는 경우

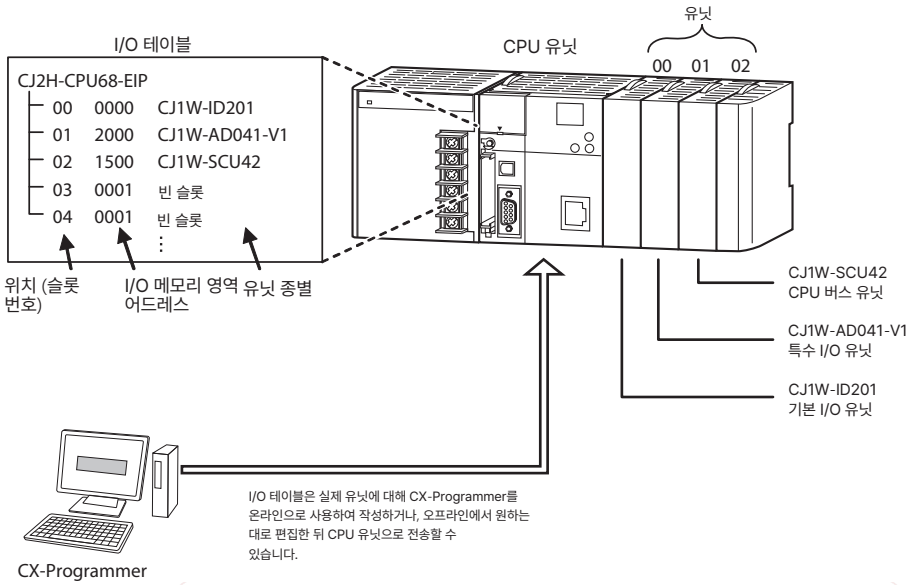
・워드를 예약하는 경우

오접속 확인이 필요하지 않고 자동 할당으로 충분한 경우에는 I/O 테이블을 작성할 필요가 없습니다.

4-3-1 I/O 테이블

CPU 유닛에 접속된 유닛의 종류와 위치가 I/O 테이블에 등록됩니다. I/O 테이블을 작성하면, CPU 유닛이 ON되었을 때 CPU에 실제로 접속된 유닛의 종류와 위치가 I/O 테이블에 등록된 데이터와 일치하는지 시스템이 확인합니다.

I/O 테이블은 CX-Programmer가 온라인일 때 실제로 접속된 유닛을 바탕으로 작성할 수 있으며, CX-Programmer가 오프라인일 때 수동으로 설정한 뒤 CPU 유닛으로 전송할 수도 있습니다.



CPU 유닛은 이 데이터에 따라 자신에게 장착된 접속 유닛(실제 I/O 테이블)과 자동으로 대조합니다.

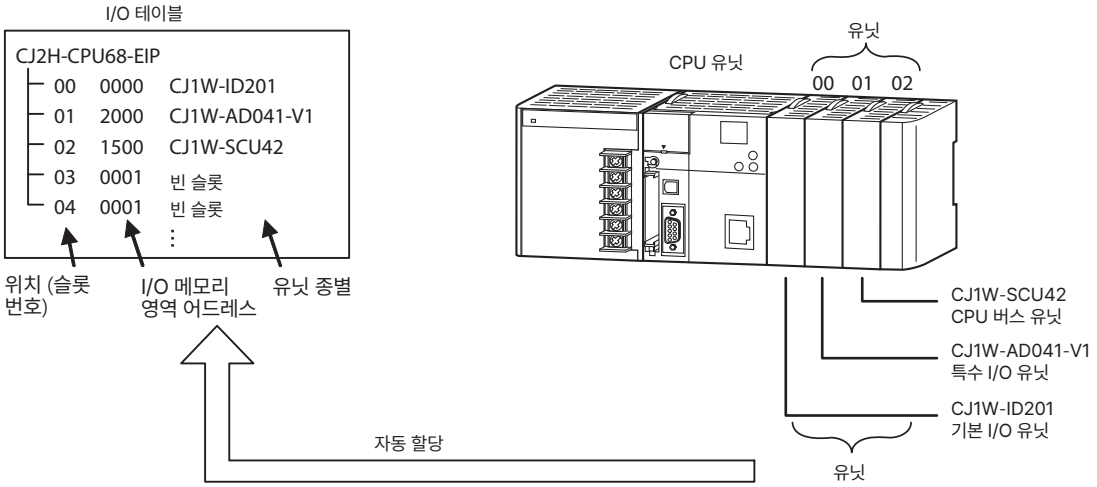
・불일치가 있으면 I/O 설정 에러가 발생하여 동작이 개시되지 않습니다.

AI 번역본
-For Reference Only-

4-3-2 자동 할당

자동 할당에서는 사용자가 I/O 테이블을 작성하지 않습니다. 전원을 투입할 때마다 유닛의 위치를 바탕으로 각 유닛에 I/O 메모리가 할당됩니다. 할당된 메모리는 유닛과의 데이터 교환에 사용됩니다.

등록 I/O 테이블이 없으며, 시스템은 유닛 구성을 체크하지 않습니다.



4-3-3 수동 할당

수동 할당에서는 사용자가 등록 I/O 테이블이라고 하는 I/O 테이블을 작성합니다.

CX-Programmer에서 다음 방법 중 하나로 I/O 테이블을 작성할 수 있습니다.

- 실제 유닛 구성을 바탕으로 I/O 테이블을 작성
  이 방법에서는 CX-Programmer를 온라인 접속한 상태에서 실제로 장착된 유닛의 구성을 바탕으로 I/O 테이블이 자동으로 작성됩니다.
- I/O 테이블을 수동으로 편집하여 CPU 유닛으로 전송
  이 방법에서는 CX-Programmer를 사용하여 각 유닛을 원하는 위치에 등록함으로써 오프라인에서 I/O 테이블을 수동으로 작성하고, 완성된 I/O 테이블을 CPU 유닛으로 전송합니다.

어느 방법을 사용하든, PLC의 전원을 투입했을 때 접속된 유닛의 구성이 I/O 테이블의 유닛 구성과 다르면 I/O 설정 에러가 발생하여(A401.10이 ON) 동작이 개시되지 않습니다.

I/O 할당 방법의 상세는 8-1 I/O 할당을 참조하십시오. I/O 테이블 작성의 상세는 8-2 CPU 버스 유닛과 특수 I/O 유닛의 설정을 참조하십시오.

AI 번역본
-For Reference Only-

4-4 라우팅 테이블 설정

PLC에 FINS 네트워크 통신 유닛이 2대 이상 장착되어 있고 다음 조작을 수행하는 경우에는 CX-Integrator로 라우팅 테이블을 설정해야 합니다.

・액세스할 네트워크를 지원 소프트웨어 또는 래더 프로그램의 명령어로 전환하는 경우

・네트워크 계층을 넘어 통신을 수행하는 경우

4-4-1 라우팅 테이블

오므론 네트워크 통신 유닛(CPU 버스 유닛) 사이에서는 FINS라는 통신 방법이 사용됩니다. 다만 전송 경로는 FINS 프로토콜에 규정되어 있지 않습니다. 라우팅 테이블은 FINS 네트워크 통신을 수행할 때 필요한 네트워크 주소 설정과 통신 경로를 정의합니다.

해당 유닛

다음 유닛의 네트워크 주소를 라우팅 테이블에 설정합니다.

・CPU 버스 유닛에 해당하는 네트워크 통신 유닛:
Controller Link 유닛, SYSMAC LINK 유닛, 이더넷 유닛, FINS 네트워크 통신에 사용하는 EtherNet/IP 유닛, DeviceNet 유닛, FL-net 유닛

・FINS 네트워크 통신에 사용하는 경우의 CJ2H-CPU6□-EIP, CJ2M-CPU3□, CJ3M-CPU3□의 내장 EtherNet/IP 포트

・시리얼 게이트웨이를 사용하여 네트워크 계층을 넘어 통신을 수행하는 경우의 시리얼 통신 유닛의 시리얼 포트

특수 I/O 유닛으로 분류되는 네트워크 통신 유닛(예: CompoNet 마스터 유닛)은 라우팅 테이블에 설정할 필요가 없습니다.

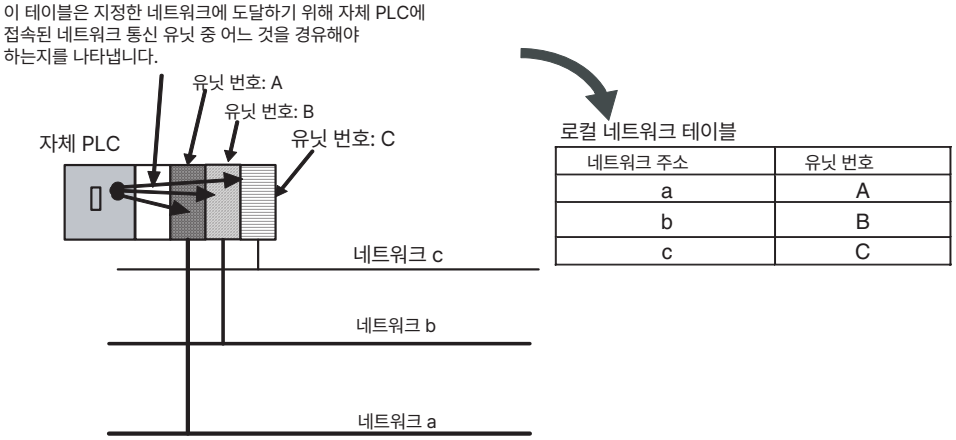
AI 번역본
-For Reference Only-

라우팅 테이블을 구성하는 테이블

라우팅 테이블은 다음 2개의 테이블로 구성됩니다.

● 로컬 네트워크 테이블

자체 PLC에 접속된 네트워크 통신 유닛(CPU 버스 유닛)에 1~127의 네트워크 주소를 설정합니다. 로컬 네트워크 테이블은 PLC에 네트워크 통신 유닛이 2대 이상 장착되어 있을 때 자체 PLC에 장착된 네트워크 통신 유닛과 그에 대응하는 통신 네트워크를 CPU 유닛이 식별하기 위해 사용됩니다.

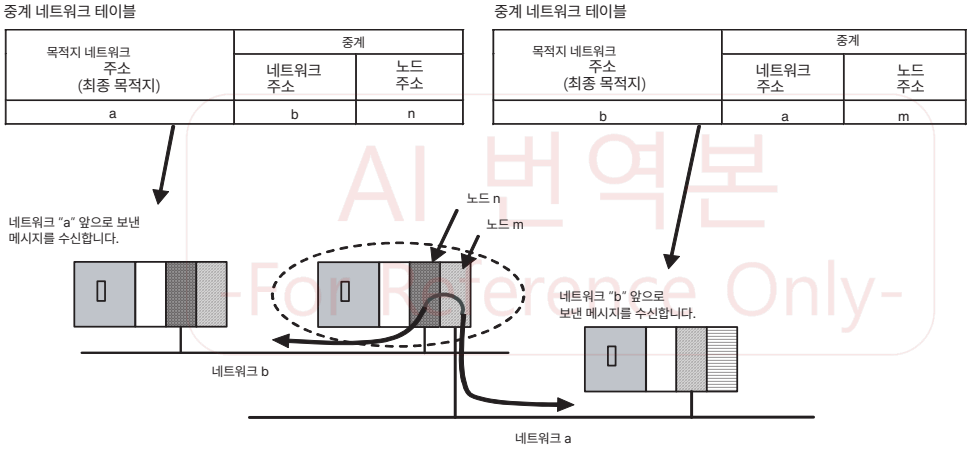


로컬 네트워크 테이블에는 자체 PLC에 접속된 네트워크 통신 유닛의 유닛 번호와, FINS 통신을 위해 해당 유닛에 접속된 FINS 네트워크의 대응 주소가 기재됩니다.

● 중계 네트워크 테이블

중계 네트워크 테이블은 네트워크 계층을 넘어 FINS 통신을 수행하기 위해 설정합니다. 중계 네트워크 테이블은 메시지가 최종 목적지에 도달할 때까지 통신을 어떻게 중계할지를 나타냅니다. 이 테이블에는 자체 PLC가 접속되어 있지 않은 목적지 네트워크(최종 네트워크)로 가는 경로상 첫 번째 중계점(즉 맨 먼저 도달하는 지점)의 네트워크 주소와 노드 주소가 기재됩니다.

중계점을 차례로 거쳐 목적지 네트워크에 도달합니다.



AI 번역본
-For Reference Only-

4-4-2 라우팅 테이블이 필요한 경우

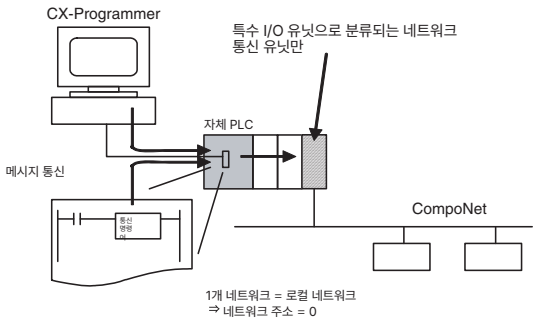
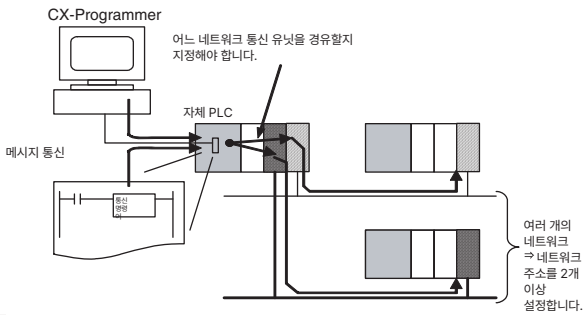
용도에 따라 필요한 라우팅 테이블(로컬 네트워크 테이블과 중계 네트워크 테이블)을 CPU 유닛에 설정하십시오.

사용상의 주의

로컬 네트워크 테이블이 필요한 조건

중계를 수행하지 않는 경우라도 PLC에 네트워크 통신 유닛이 2대 이상 접속되어 있으면 로컬 네트워크 테이블이 필요합니다.

네트워크 사용 조건에 따라 라우팅 테이블이 필요한지 여부를 다음 표에 나타냅니다.

네트워크 사용 조건	시스템 구성	라우팅 테이블	
		로컬 네트워크 테이블	중계 네트워크 테이블
PLC에 네트워크 통신 유닛이 접속되어 있지 않거나, 특수 I/O 유닛으로 분류되는 네트워크 통신 유닛(예: CompoNet)만 접속되어 있는 경우*1		필요 없음	필요 없음
CPU 버스 유닛으로 분류되는 통신 유닛이 PLC에 2대 이상 접속되어 있는 경우.		네트워크 내 모든 PLC에 대해 필요*2	네트워크 계층을 넘는 통신을 수행하는 경우에 필요. *3

*1 프로그래밍 디바이스로 네트워크 내의 노드에 액세스하는 경우에는 상대 네트워크 주소를 0으로 설정하십시오.

*2 로컬 네트워크 테이블이 없는 상태에서 프로그래밍 디바이스나 래더 프로그램의 명령어로 네트워크 주소를 0으로 지정해 액세스하면, FINS 통신에서는 유닛 번호가 가장 작은 네트워크 통신 유닛이 속한 네트워크에 자동으로 액세스합니다. 이와 같이 액세스할 네트워크의 네트워크 통신 유닛에 가장 작은 유닛 번호를 설정하면 로컬 네트워크 테이블을 설정하지 않고도 그 네트워크에 액세스할 수 있습니다.

*3 네트워크 계층을 넘는 통신을 수행하지 않는 경우에는 중계 네트워크 테이블을 설정할 필요가 없습니다.

-For Reference Only-

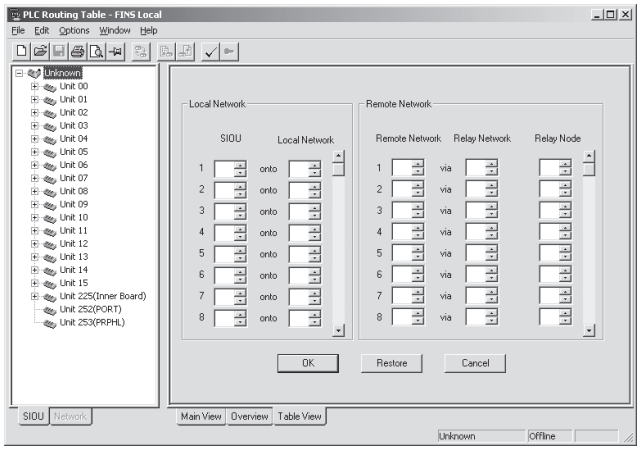
4-4-3 라우팅 테이블의 설정과 전송

1 CX-Integrator를 기동합니다.

2 Tools 메뉴에서 Start Routing Table을 선택합니다.

3 FINS Local을 선택합니다.

4 Table View 탭 페이지에서 라우팅 테이블을 편집합니다.



5 CX-Integrator를 온라인 접속한 뒤 Options 메뉴에서 Transfer to PLC를 선택하여 라우팅 테이블을 CPU 유닛으로 전송합니다.

6 File 메뉴에서 Save Local Routing Table File을 선택합니다.

사용상의 주의

라우팅 테이블 데이터 파일
라우팅 테이블은 CX-Integrator로 작성되는 개별 파일(.rtg)에 저장됩니다. CX-Programmer의 프로젝트 파일(.cxp)에는 포함되지 않습니다.

AI 번역본
-For Reference Only-

4-5 특수 I/O 유닛과 CPU 버스 유닛의 할당 DM 영역 워드 설정

특수 I/O 유닛이나 CPU 버스 유닛을 사용하는 경우에는 이 설정이 필요합니다. 특수 I/O 유닛과 CPU 버스 유닛에 할당된 DM 영역 워드를 CX-Programmer로 설정하십시오.

4-5-1 특수 I/O 유닛과 CPU 버스 유닛의 할당 DM 영역 워드 설정

특수 I/O 유닛과 CPU 버스 유닛의 초기 설정은 CPU 유닛의 I/O 메모리 DM 영역에서 해당 유닛에 할당된 워드에 실시합니다. 할당 DM 영역 워드의 설정은 전원을 투입할 때 CPU 유닛에서 특수 I/O 유닛과 CPU 버스 유닛으로 전송됩니다.

할당 DM 영역 워드는 다음 어드레스부터 할당됩니다.

- **특수 I/O 유닛의 초기 설정**

  유닛 전면의 로터리 스위치로 설정한 유닛 번호에 따라 D20000~D29599 범위에서 유닛당 100워드가 할당됩니다.

- **CPU 버스 유닛의 초기 설정**

  유닛 전면의 로터리 스위치로 설정한 유닛 번호에 따라 D30000~D31599 범위에서 유닛당 100워드가 할당됩니다.

할당 DM 워드의 실제 설정과 전송 타이밍은 특수 I/O 유닛과 CPU 버스 유닛의 종별 및 영역에 따라 다릅니다. 상세는 해당 특수 I/O 유닛 또는 CPU 버스 유닛의 오퍼레이션 매뉴얼을 참조하십시오.

참고

이더넷 유닛이나 Controller Link 유닛과 같은 특정 유닛에서는 특수 I/O 유닛과 CPU 버스 유닛의 설정이 CPU 버스 유닛 셋업 영역에도 들어 있으며, 특수 I/O 유닛과 CPU 버스 유닛 안에도 데이터와 프로그램이 저장됩니다(해당 유닛 전용 지원 소프트웨어로 작성).

AI 번역본
-For Reference Only-

4-5-2 설정 순서

다음 방법 중 하나로 설정하십시오.

- CX-Programmer의 PLC I/O 테이블 윈도우에서 유닛을 선택하고 특수 I/O 유닛과 CPU 버스 유닛의 설정을 실시한 뒤, 그 설정을 CPU 유닛으로 전송합니다.
- CX-Programmer의 PLC 메모리 윈도우에서 할당 DM 영역 워드의 개별 어드레스에 데이터를 설정한 뒤, 그 설정을 CPU 유닛으로 전송합니다.
- 사용자 프로그램에서 MOV(021) 등의 명령어를 사용하여 할당 DM 영역 워드의 개별 어드레스에 데이터를 설정합니다.

CX-Programmer로 CPU 버스 유닛과 특수 I/O 유닛에 DM 영역 워드를 할당하는 순서는 8-2 CPU 버스 유닛과 특수 I/O 유닛의 설정을 참조하십시오.

4-6 CPU 버스 유닛 셋업 영역

4-6-1 CPU 버스 유닛 셋업 영역

셋업 영역에는 CPU 유닛에 장착된 특정 CPU 버스 유닛의 설정이 저장됩니다. 다음 3종류의 유닛과 설정이 이 영역을 사용합니다.

- 이더넷 유닛: Ethernet 설정
- Controller Link 유닛: 데이터 링크 테이블(사용자 설정 테이블을 사용하는 경우)
- FL-net 유닛: FL-net 설정

4-6-2 설정 순서

다음 지원 소프트웨어로 설정하십시오.

데이터	지원 소프트웨어
이더넷 유닛 설정	CX-Programmer의 I/O 테이블에서 특수 I/O 유닛 또는 CPU 버스 유닛의 설정을 실시하거나, HTML을 사용하여 설정합니다.
Controller Link 유닛 데이터 링크 테이블	CX-Integrator의 데이터 링크 테이블로 설정합니다.
FL-net 유닛 설정	FL-net 지원 소프트웨어인 CX-FLnet으로 설정합니다.

CPU 버스 유닛 셋업 영역의 사용 메모리

CPU 버스 유닛 셋업 영역에 사용할 수 있는 최대 메모리 용량은 10,752바이트입니다.
PLC 내 CPU 버스 유닛의 조합에 따라 CPU 버스 유닛 셋업 영역의 사용 메모리가 이 제한 이내가 되도록 시스템을 설계하십시오. 제한을 초과하면 일부 유닛이 디폴트 설정으로만 동작하거나 동작하지 않을 수 있습니다.

● CPU 버스 유닛 셋업 영역의 사용 메모리

유닛	형식	사용 사이즈 (바이트)
Controller Link 유닛	CJ1W-CLK23	512
이더넷 유닛	CJ1W-ETN11	412
	CJ1W-ETN21	994
FL-net 유닛	CJ1W-FLN22	988
EtherNet/IP 유닛	CJ1W-EIP21/EIP21S	0
시리얼 통신 유닛	CJ1W-SCU21/31/41-V1 CJ1W-SCU22/32/42	
DeviceNet 유닛	CJ1W-DRM21	
MECHATROLINK-II 통신 대응 위치 제어 유닛	CJ1W-NC271/471/F71(-MA)	
EtherCAT 대응 위치 제어 유닛	CJ1W-NC281/481/881/F81 CJ1W-NC482/882/F82	
모션 컨트롤 유닛	CJ1W-MCH71	
아날로그 입력 유닛	CJ1W-ADG41	
스토리지 프로세싱 유닛	CJ1W-SPU01-V2	
EtherCAT 슬레이브 유닛	CJ1W-ECT21	

0바이트를 사용하는 유닛은 CPU 버스 유닛 셋업 영역을 사용하지 않습니다.

AI 번역본
-For Reference Only-

5

프로그래밍의 이해

본 장에서는 CJ2/CJ3 CPU 유닛의 프로그래밍 기본을 설명합니다.

5-1 프로그래밍 5-3
5-1-1 프로그래밍 개요 5-3
5-1-2 래더 다이어그램의 기본 개념 5-6
5-1-3 ST 언어 5-8
5-1-4 SFC 개요 5-9

5-2 태스크 5-11
5-2-1 태스크 개요 5-11
5-2-2 사이클릭 태스크 5-14
5-2-3 인터럽트 태스크 5-20
5-2-4 태스크 설계 5-30

5-3 섹션 5-40
5-3-1 섹션 개요 5-40

5-4 펑션 블록 5-42
5-4-1 펑션 블록 5-42
5-4-2 펑션 블록의 특징 5-43
5-4-3 펑션 블록 사양 5-44

5-5 심볼 5-48
5-5-1 개요 5-48
5-5-2 심볼의 종별 5-49
5-5-3 글로벌 심볼 5-51
5-5-4 로컬 심볼 5-51
5-5-5 네트워크 심볼 5-52
5-5-6 펑션 블록의 변수 5-56
5-5-7 심볼 데이터형 5-57
5-5-8 심볼에 대한 자동 어드레스 할당 5-62

5-6 명령어 5-63
5-6-1 명령어의 기본 이해 5-63
5-6-2 오퍼랜드 지정 5-70
5-6-3 데이터 형식 5-78
5-6-4 I/O 리프레시 타이밍 5-82

AI 번역본
-For Reference Only-

5-7 인덱스 레지스터 5-87
5-7-1 인덱스 레지스터란? 5-87
5-7-2 인덱스 레지스터 사용법 5-87
5-7-3 인덱스 레지스터 관련 처리 5-93
5-7-4 인덱스 레지스터 모니터링 5-94
5-7-5 태스크 간 인덱스 레지스터와 데이터 레지스터 공유 5-95
5-8 어드레스 오프셋 지정 5-97
5-8-1 개요 5-97
5-8-2 어드레스 오프셋 응용 예 5-99
5-9 프로그램 체크 5-100
5-9-1 CX-Programmer 입력 시의 에러 5-100
5-9-2 CX-Programmer에 의한 프로그램 체크 5-100
5-9-3 시뮬레이터에 의한 디버그 5-101
5-9-4 프로그램 실행 확인 5-104
5-10 주의사항 5-107
5-10-1 조건 플래그 5-107
5-10-2 특수 프로그램 섹션 5-112

AI 번역본
-For Reference Only-

5-1 프로그래밍

5-1-1 프로그래밍 개요

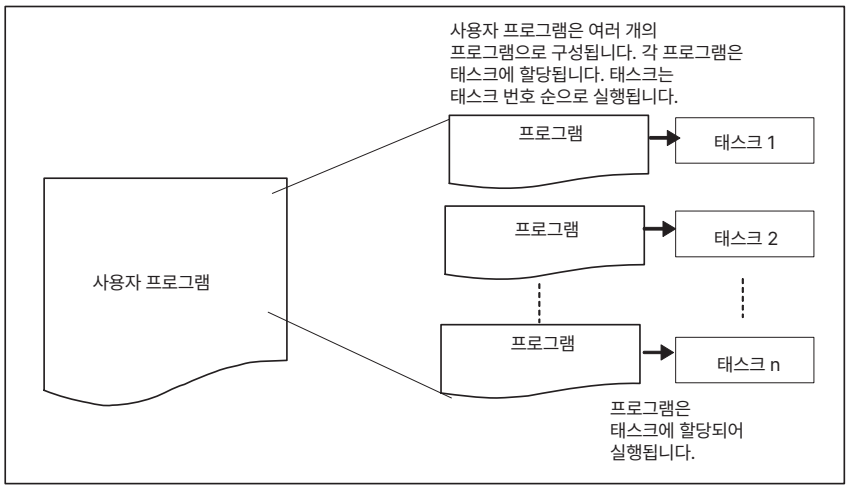
사용자 프로그램의 구성

사용자 프로그램은 CX-programmer를 사용하여 작성합니다. 프로그램은 다음 요소로 구성됩니다.

・프로그램
사용자 프로그램 전체는 여러 개의 프로그램으로 구성됩니다. 각 프로그램은 END(001) 명령어로 끝납니다.

・태스크(실행 가능한 최소 단위)
프로그램은 태스크에 할당되어 실행됩니다. (CX-Programmer에서는 태스크 번호를 프로그램 속성으로서 프로퍼티에서 지정합니다.) 태스크에는 사이클릭 태스크(통상의 사이클 처리로 실행), 인터럽트 태스크(인터럽트 조건이 성립했을 때 실행), 스케줄 인터럽트 태스크(지정 간격마다 실행), 전원 OFF 인터럽트 태스크(정전 시에 실행)가 있습니다.

CPU 유닛



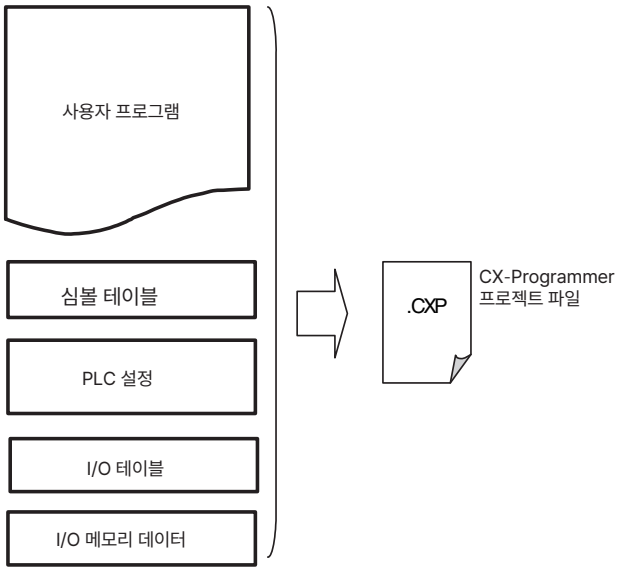
・섹션
CX-Programmer에서 프로그램을 작성 · 표시할 때 하나의 프로그램을 임의의 개수로 나눌 수 있습니다. 나뉜 각 부분을 섹션이라고 합니다. 섹션은 일반적으로 프로그램을 이해하기 쉽게 하기 위해 작성합니다.

・서브루틴과 펑션 블록
하나의 프로그램 안에 서브루틴 프로그램과 펑션 블록을 작성할 수 있습니다.

AI 번역본
-For Reference Only-

사용자 프로그램 데이터

사용자 프로그램 전체는 심볼 테이블, PLC 설정 데이터, I/O 테이블, I/O 메모리 데이터 등의 기타 파라미터와 함께 CX-Programmer 프로젝트 파일(.CXP)에 저장됩니다.

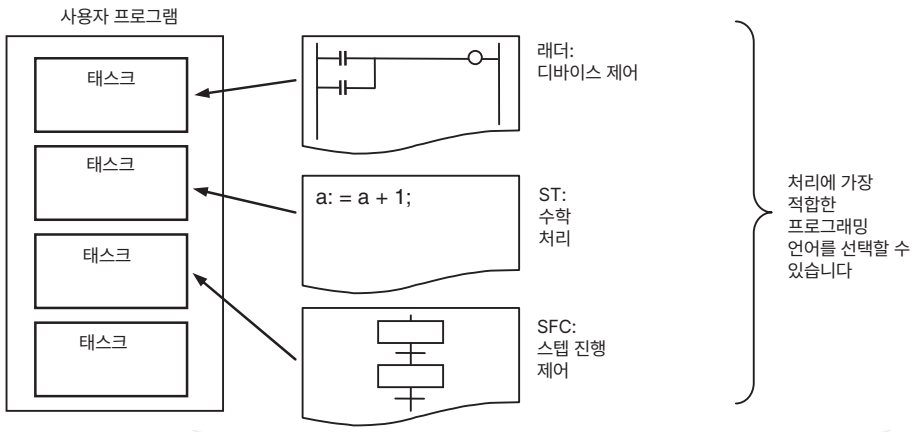


프로그래밍 언어

프로그램은 다음 3가지 프로그래밍 언어 중 하나로 작성할 수 있습니다.

- 래더 다이어그램
- 구조화 텍스트(ST)
- 시퀀셜 펑션 차트(SFC)

래더 다이어그램, ST, SFC로 작성된 프로그램은 태스크에 할당되어 실행됩니다.



디바이스 제어에는 래더 프로그래밍을, 수학 처리에는 ST를 사용하는 것처럼 용도에 따라 언어를 구분해 사용함으로써 각 처리에 가장 적합한 언어를 사용할 수 있습니다.

각 언어는 다음 위치에서 사용할 수 있습니다.

위치 / 언어	태스크	SFC의 액션 프로그램과 트랜지션 프로그램	펑션 블록 정의의 알고리즘
래더 다이어그램	가능	가능	가능
ST	가능	가능	가능
SFC	가능	---	---

각 언어에서 사용할 수 있는 어드레싱 방식은 다음과 같습니다.

어드레싱 방식 / 언어	물리 어드레스	심볼
래더 프로그래밍	가능	가능
ST	조건부*1	가능
SFC	조건부*2	조건부*2

*1 물리 어드레스는 심볼에 할당하여 사용할 수 있습니다.

*2 물리 어드레스와 심볼은 SFC 차트의 불리언 액션과 트랜지션에서 사용할 수 있습니다.

프로그램 용량

CJ2/CJ3 CPU 유닛의 전체 사용자 프로그램에 대한 최대 프로그램 용량(즉, 모든 태스크의 합계 용량)은 다음 표와 같습니다.

· CJ2 CPU 유닛

형식	프로그램 용량	FB 프로그램 영역	I/O 용량
CJ2H-CPU68(-EIP)	400K steps	없음	2,560점
CJ2H-CPU67(-EIP)	250K steps		
CJ2H-CPU66(-EIP)	150K steps		
CJ2H-CPU65(-EIP)	100K steps		
CJ2H-CPU64(-EIP)	50K steps		
CJ2M-CPU15/35	60K steps	20K steps	
CJ2M-CPU14/34	30K steps		
CJ2M-CPU13/33	20K steps		
CJ2M-CPU12/32	10K steps		
CJ2M-CPU11/31	5K steps		

· CJ3 CPU 유닛

형식	프로그램 용량	FB 프로그램 영역	I/O 용량
CJ3M-CPU15/35	100K steps	40K steps	5,120점
CJ3M-CPU14/34	50K steps		
CJ3M-CPU13/33	20K steps		
CJ3M-CPU12/32	10K steps		
CJ3M-CPU11/31	5K steps		

시 버 영 보
-For Reference Only-

CX-programmer에서 View − Memory View를 선택하면 프로그램 용량을 확인할 수 있습니다.

래더 명령어의 크기는 사용하는 명령어와 오퍼랜드에 따라 달라집니다. 자세한 내용은 A-2 명령어 실행 시간과 스텝 수를 참조하십시오.

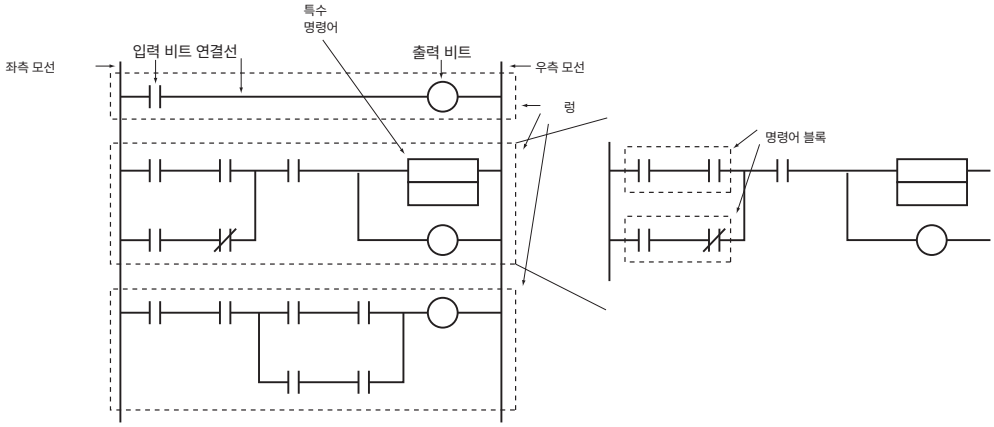
펑션 블록이 사용하는 메모리 용량은 5-4-3 펑션 블록 사양을 참조하십시오.

5-1-2 래더 다이어그램의 기본 개념

래더 다이어그램 로직은 전기 회로와 비슷하게 보이는 형태로 기술하는 PLC의 기본 언어입니다. 명령어는 메모리에 기록된 순서(니모닉 순서)대로 실행됩니다. 실행 순서뿐 아니라 기본적인 프로그래밍 개념도 올바르게 이해하는 것이 중요합니다.

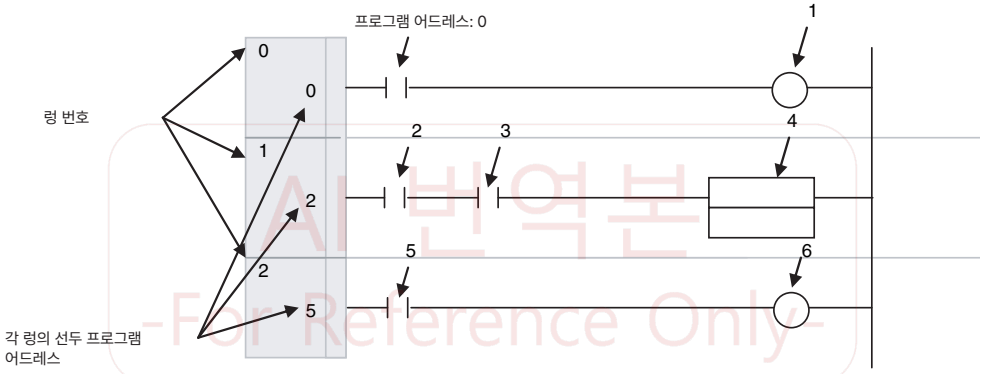
● 래더 다이어그램의 전체 구성

래더 다이어그램은 좌우의 모선, 연결선, 입력 비트, 출력 비트, 특수 명령어로 구성됩니다. 프로그램은 하나 이상의 프로그램 런으로 구성됩니다. 프로그램 렁은 모선을 가로로 분할했을 때 나눌 수 있는 단위입니다. 니모닉 형식에서 렁이란 LD/LD NOT 명령어부터 다음 LD/LD NOT 명령어 직전의 출력 명령어까지의 모든 명령어를 말합니다. 프로그램 렁은 논리 시작을 나타내는 LD/LD NOT 명령어로 시작하는 명령어 블록으로 구성됩니다.



참고

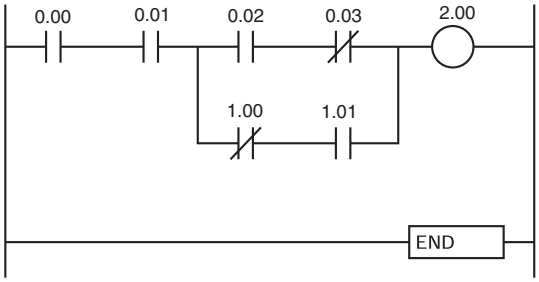
CX-Programmer에서는 각 렁의 왼쪽에 렁 번호와 그 렁의 선두 프로그램 어드레스가 표시됩니다.



AI 번역본
-For Reference Only-

● 니모닉

프로그램 어드레스를 가지며, 1개의 프로그램 어드레스가 1개의 명령어에 해당합니다. 프로그램 어드레스는 0부터 시작하는 6자리입니다.



프로그램 어드레스	명령어 (니모닉)	오퍼랜드
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	

AI 번역본
-For Reference Only-

5-1-3 ST 언어

ST(구조화 텍스트) 언어는 IEC 61131-3 규격으로 정의된 산업용 제어(주로 PLC)를 위한 고급 언어 코드입니다. 표준 제어문, 연산자, 함수를 갖추고 있어, 래더 프로그래밍으로는 기술하기 어려운 수치 연산 처리에 적합합니다. (ST

언어는 래더 다이어그램으로 기술할 수 있는 모든 처리를 지원하지는 않습니다. IEC 61131-3 규격에 준거한 ST 언어를 지원합니다.

참고

ST 프로그래밍의 사양, 표기법, 입력 순서에 대한 자세한 내용은 CX-Programmer 오퍼레이션 매뉴얼: 펑션 블록 및 구조화 텍스트(Cat. No. W447)를 참조하십시오.

● ST 프로그래밍의 특징

- 루프문, IF-THEN-ELSE 문 등 많은 제어문과 산술 연산자, 비교 연산자, AND/OR 연산자 등 많은 연산자를 사용할 수 있으며, 수학 함수, 문자열 추출 · 결합 함수, 메모리 카드 처리 함수, 문자열 전송 함수, 삼각함수 등 많은 함수도 사용할 수 있습니다.
- C 등의 고급 언어와 같은 방식으로 프로그램을 기술할 수 있고, 코멘트를 넣어 프로그램을 읽기 쉽게 만들 수 있습니다.
- ST 프로그램은 일반 프로그램과 마찬가지로 업로드 · 다운로드할 수 있지만, ST 프로그램 태스크는 태스크 단위로 업로드 · 다운로드할 수 없습니다.
- ST 프로그램에서 펑션 블록(래더 또는 ST 언어)을 호출할 수 있습니다.
- 1차원 배열 변수를 지원하므로 애플리케이션에서 데이터를 다루기가 쉬워집니다.

● 사양

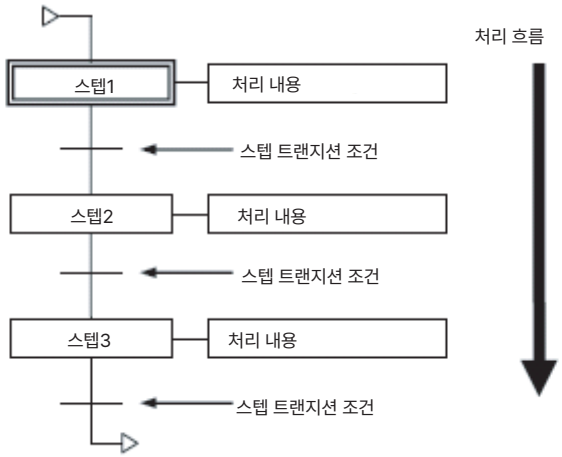
항목	사양
ST 프로그램 단위	· 태스크 · 펑션 블록 정의의 알고리즘 · SFC의 액션 프로그램과 트랜지션 프로그램.
어드레스 지정 방법	어드레스는 심볼을 사용하여 지정합니다. 심볼에는 임의의 어드레스를 할당할 수 있습니다.
강제 세트 및 강제 리셋	지원합니다.
온라인 에디팅	ST 프로그램 편집

AI 번역본
-For Reference Only-

5-1-4 SFC 개요

SFC(Sequential Function Chart) 언어는 주로 시퀀스 처리를 제어하는 스텝 진행형 프로그램을 쉽게 기술할 수 있도록 개발된 그래픽 프로그래밍 언어입니다.

SFC는 스텝의 흐름을 그래픽으로 표현하고 스텝 진행 조건과 각 스텝의 액션을 기술함으로써, 시퀀스 처리의 제어를 프로그래밍할 수 있게 합니다.



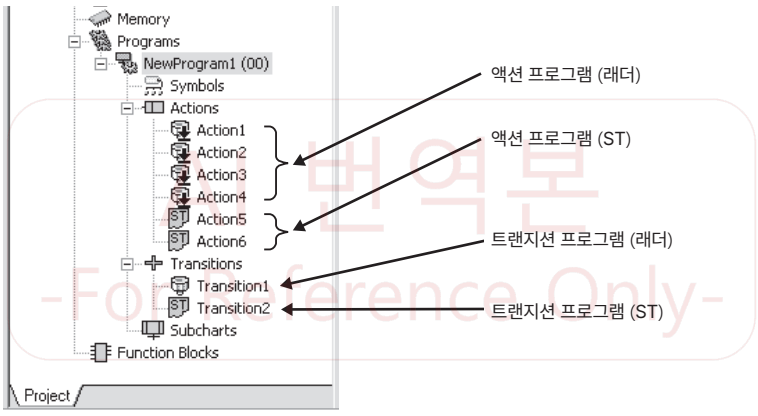
참고

SFC 프로그래밍의 사양, 표기법, 입력 순서에 대한 자세한 내용은 CX-programmer 오퍼레이션 매뉴얼: SFC 프로그래밍(Cat. No. W469)을 참조하십시오.

● SFC 프로그래밍의 특징

SFC 프로그램의 일부로서의 래더 다이어그램과 ST

SFC 프로그램 내의 스텝 액션과 트랜지션 조건은 래더 다이어그램 또는 ST 중 하나로 기술할 수 있습니다.



번역본
-For Reference Only-

● SFC 차트와 액션/트랜지션 프로그램의 동시 표시 · 편집

온라인 에디팅과 디버깅

SFC 차트는 온라인으로 편집할 수 있습니다. 또한 디버깅 중에 액션 블록을 숨기고 스텝 진행 상태를 확인할 수 있습니다. SFC 차트 에디터는 액션 블록을 숨긴 상태에서도 액션 프로그램을 프로그램 뷰에 표시합니다.

● 사양

항목	사양
SFC 프로그램 단위	태스크 (1 태스크 = 1 SFC 차트)
SFC 프로그램을 할당할 수 있는 태스크	사이클릭 태스크 또는 엑스트라 사이클릭 태스크
SFC 요소	스텝, 트랜지션, 액션, 점프, 서브차트 주: SFC 요소는 로컬 변수에 자동으로 등록됩니다.
어드레스 지정 방법	SFC의 불리언 액션 또는 트랜지션에는 심볼 또는 물리 어드레스를 지정할 수 있습니다. 단, ST 프로그래밍을 사용하는 경우에는 물리 어드레스를 지정할 수 없습니다.
강제 세트 및 강제 리셋	트랜지션을 온라인으로 강제 세트/강제 리셋하여 스텝 단위로 실행할 수 있습니다.
온라인 에디팅	SFC 편집, 액션 추가/삭제, 트랜지션 추가/삭제를 할 수 있습니다.

AI 번역본
-For Reference Only-

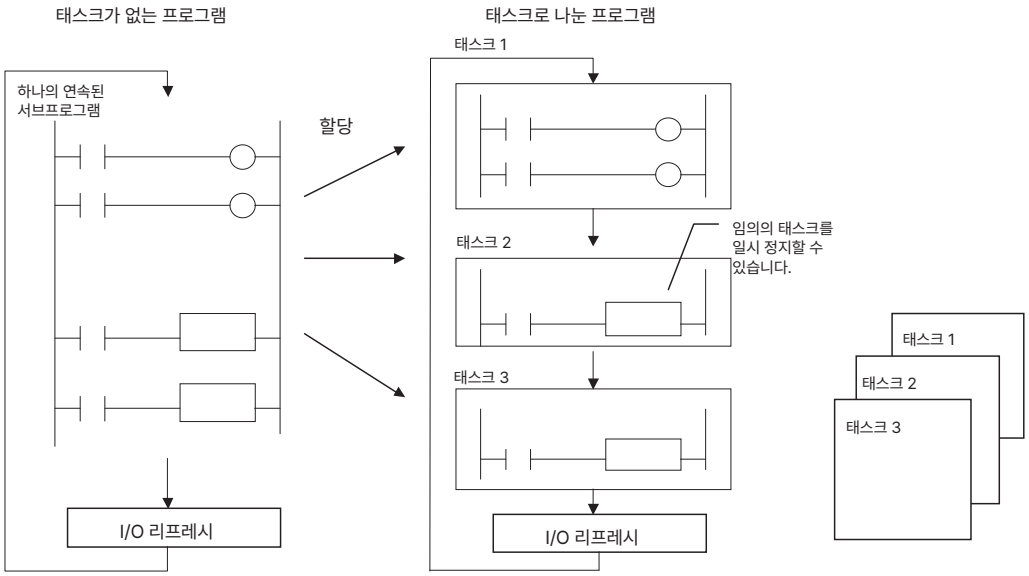
5-2 태스크

5-2-1 태스크 개요

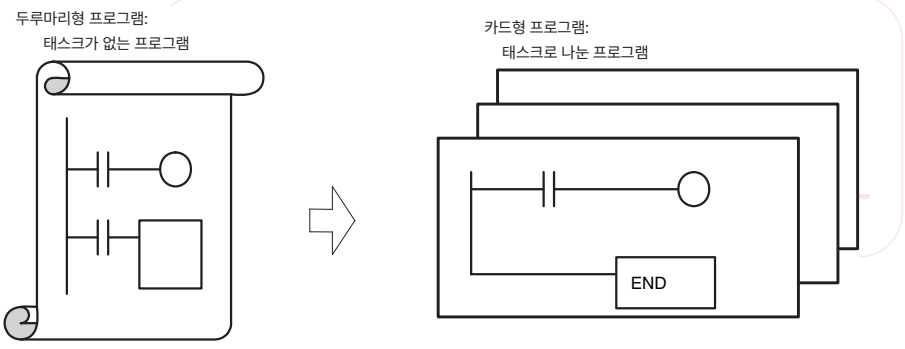
태스크의 개념

태스크는 프로그램을 큰 단위로 나누고 각 단위 또는 프로그래밍을 실행하는 순서를 지정하는 데 사용합니다. 어느 한 태스크의 명령어로 다른 태스크의 실행을 유효 또는 무효로 할 수 있습니다.

즉, 여러 프로그램 구성 요소를 서로 다른 태스크로 조합해 두고, 현재의 제품 기종이나 수행 중인 공정에 필요한 특정 태스크만 실행할 수 있습니다.
이에 따라 공정별로 태스크를 전환할 수 있습니다. 필요한 프로그램만 필요할 때 실행되므로 성능(사이클 타임)이 크게 향상됩니다.



태스크가 없는 프로그램은 두루마리 전체를 처음부터 읽는 것과 같고, 태스크가 있는 프로그램은 카드를 한 장씩 읽는 것과 같습니다. 각 카드는 다른 카드를 유효 또는 무효로 할 수 있습니다. 무효인 카드는 건너뜁니다.



프로그램 전체를 읽습니다.

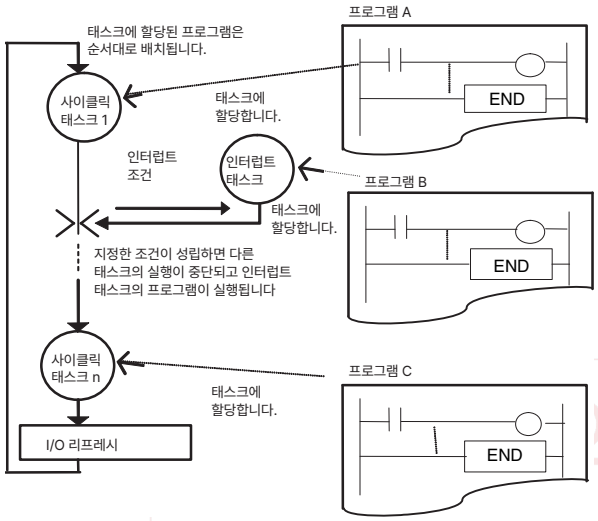
필요한 카드만 읽습니다.

태스크에는 기본적으로 두 종류가 있습니다.

태스크의 종류	설명	적용 가능한 프로그래밍 언어	실행 조건
사이클릭 태스크	사이클마다 1회 실행됩니다	래더 다이어그램 SFC ST	다음 방법 중 하나로 태스크를 READY 상태로 만들었을 때 실행됩니다. ・CX-Programmer로 Activated at the start of operation 속성을 설정한다 (프로그램 속성 설정) ・태스크 제어 명령어로 태스크를 ON한다
인터럽트 태스크*1,*2	특정 조건이 발생했을 때 실행됩니다. 실행 중이던 처리는 중단됩니다.	래더 다이어그램 ST	인터럽트 태스크에 설정된 인터럽트 조건이 발생하면 해당 태스크가 READY 상태가 됩니다. 다음 각 인터럽트 태스크에 대해 조건을 설정할 수 있습니다. ・전원 OFF 인터럽트 태스크 ・정주기 인터럽트 태스크 ・I/O 인터럽트 태스크 ・외부 인터럽트 태스크 ・입력 인터럽트 태스크 (펄스 I/O 모듈) (CJ2M/CJ3M CPU 유닛 전용)

*1 태스크 제어 명령어(TKON)로 인터럽트 태스크를 ON하면, 사이클릭 태스크와 마찬가지로 인터럽트 태스크도 사이클릭 실행(즉, 사이클마다 1회 실행)할 수 있습니다. (이러한 태스크를 엑스트라 사이클릭 태스크라고 합니다.)

*2 인터럽트 태스크에서는 SFC 프로그램을 사용하지 마십시오. 정상적으로 실행되지 않습니다. 단, 인터럽트 태스크를 엑스트라 사이클릭 태스크로 실행하는 경우에는 SFC 프로그램을 사용할 수 있습니다.

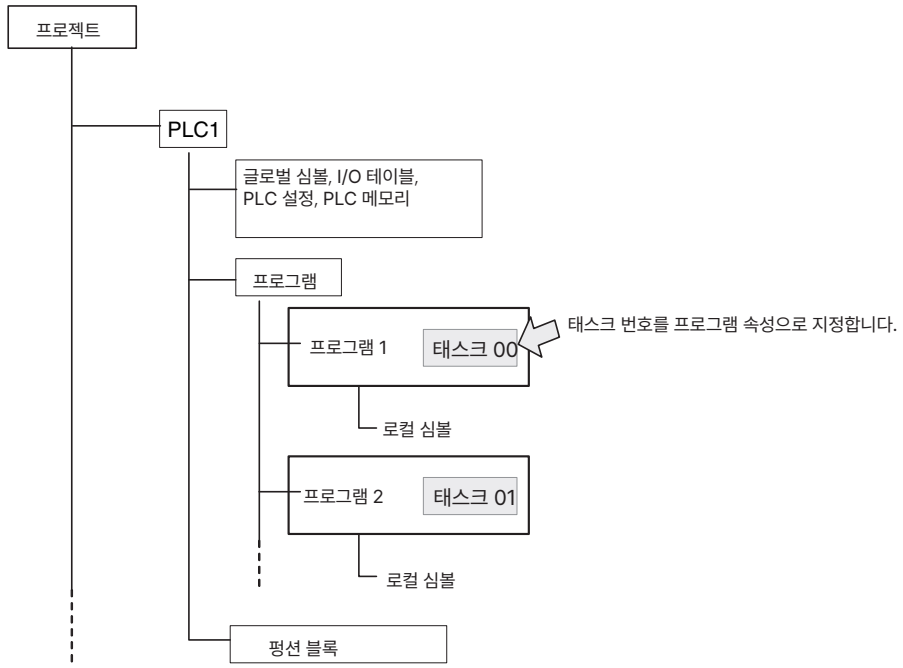


견본
-For Reference Only-

태스크에 관한 CX-Programmer 조작

프로그램의 태스크 번호는 CX-Programmer에서 프로그램 속성으로 설정합니다.

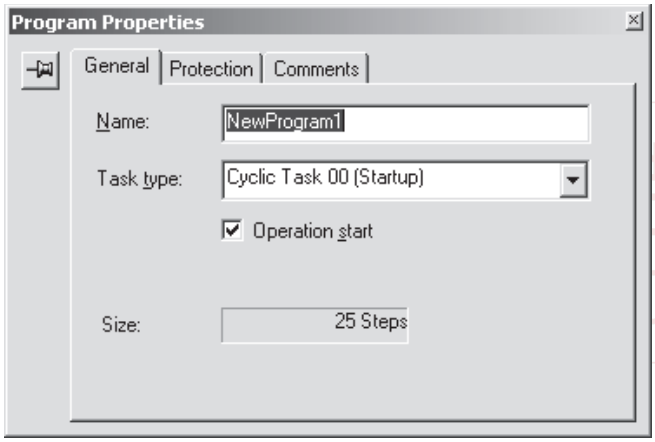
● 태스크 번호 설정



각 프로그램의 속성으로 태스크 종류와 번호를 지정합니다.

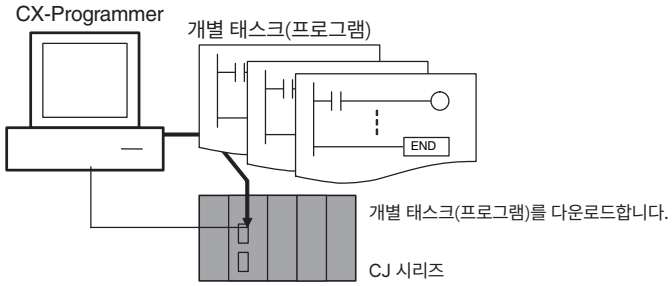
1 View/Properties를 선택하거나 오른쪽 버튼을 클릭해 팝업 메뉴에서 Properties를 선택하여, 태스크를 할당할 프로그램을 표시합니다.

2 General 탭을 선택하고 Task Type과 Task No.를 선택합니다. 사이클릭 태스크의 경우, 운전 개시 시에 태스크 실행을 시작하려면 Operation start 체크 박스를 선택합니다.



● **태스크 단위 다운로드 · 업로드**

CX-Programmer에서 프로그램 태스크를 개별적으로 업로드 · 다운로드할 수 있습니다.



● **펑션 블록 사용 시의 제한**

펑션 블록을 포함하는 프로그램은 태스크 단위로 다운로드할 수 없습니다(업로드는 가능합니다).

● **SFC 프로그램 · ST 프로그램 사용 시의 제한**

SFC 프로그램 또는 ST 프로그램이 할당된 태스크는 개별적으로 업로드 · 다운로드할 수 없습니다.

5-2-2 사이클릭 태스크

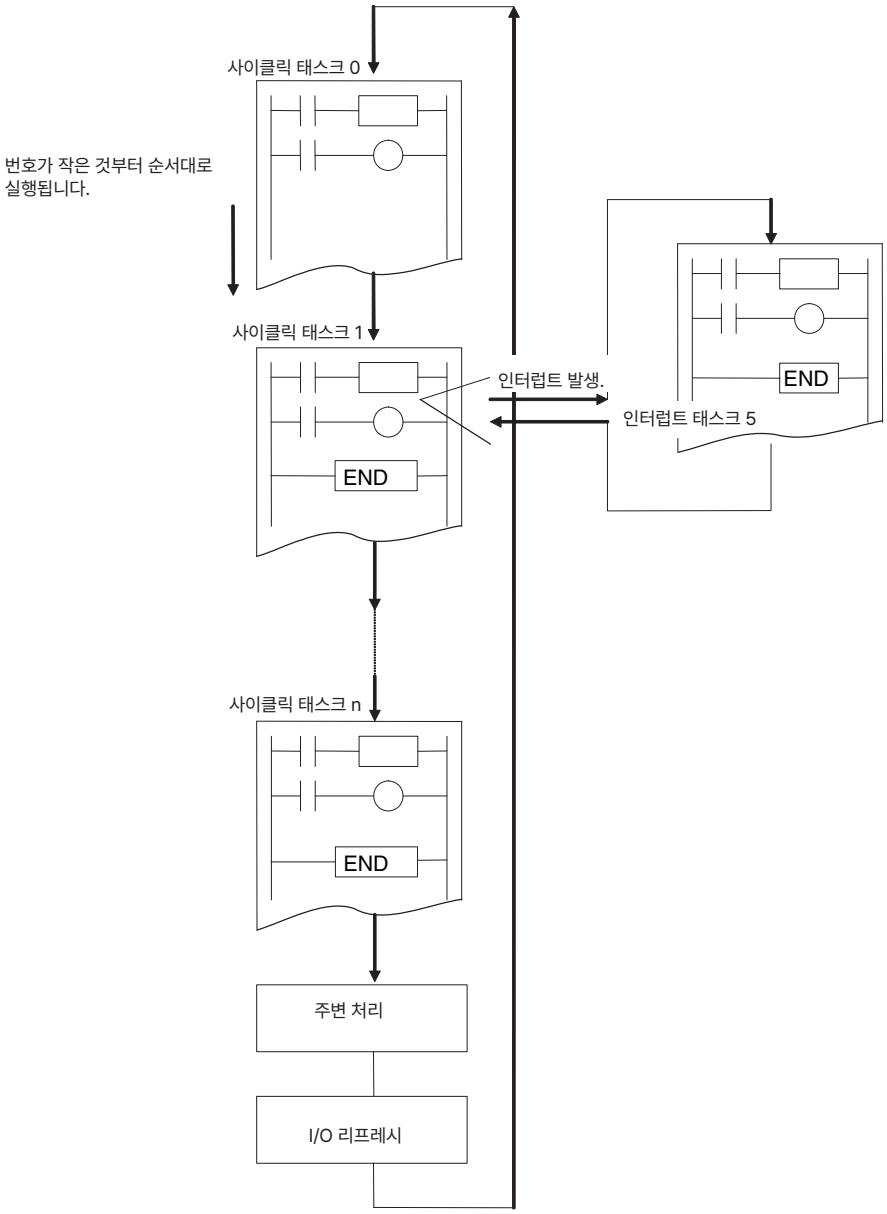
사이클릭 태스크

사이클릭 태스크는 태스크 번호가 작은 것부터 순서대로 사이클마다 1회 실행됩니다. 최대 128개의 태스크를 사용할 수 있습니다(사이클릭 태스크 번호 0~127). 태스크는 CX-Programmer로 Activated at the start of operation 속성을 설정하거나 태스크 제어 명령어를 사용하여 기동할 수 있습니다. 태스크 제어 명령어에 대한 자세한 내용은 A-2 명령어 실행 시간과 스텝 수를 참조하십시오. 엑스트라 사이클릭 태스크를 사용하면 최대 384개의 사이클릭 태스크를 사용할 수 있습니다.

사용상의 주의

SFC 프로그램을 포함하는 태스크를 태스크 제어 명령어(TKON(820), TKOFF(821))로 제어하지 마십시오. 정상적으로 실행되지 않습니다.

AI 번역본
-For Reference Only-



참고

- 모든 조건 플래그(ER, CY, Equals, AER 등)와 명령어 조건은 태스크 시작 시에 클리어됩니다. 따라서 두 태스크 사이에서 조건 플래그를 읽을 수는 없습니다.
- 인터록(IL · ILC 명령어 등), 점프(JMP · CJP · JME 명령어 등), 서브루틴(SBS · RET · SBN 명령어 등)은 각 태스크 안에서 완결되어야 합니다. 예를 들어 어느 태스크에서 다른 태스크로 점프할 수는 없습니다. 여러 태스크에서 서브루틴을 사용하려면 글로벌 서브루틴 (GSBS(750), GRET(752), GSBN(751) 명령어)을 사용하십시오.

사이클릭 태스크의 상태

● READY 상태

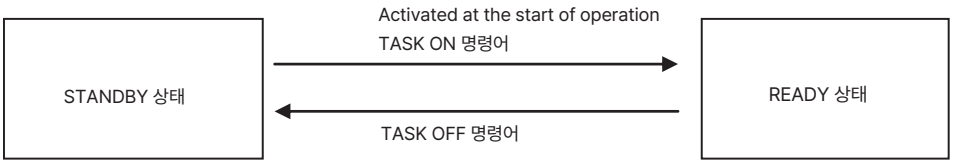
사이클릭 태스크는 태스크 번호가 작은 것부터 순서대로 실행됩니다. Activated at the start of operation 속성이 설정된 태스크는 운전 모드가 PROGRAM 모드에서 RUN 또는 MONITOR 모드로 바뀌었을 때 실행 가능한 상태가 됩니다. TASK ON 명령어를 사용하면 STANDBY 상태에서 READY 상태로 바꿀 수 있습니다. 자기 태스크를 정지시키기 위해 TASK OFF 명령어를 사용하면, 그 태스크는 TASK OFF 명령어 이후로는 실행되지 않습니다.

● STANDBY 상태

STANDBY 상태의 태스크에서는 명령어가 실행되지 않습니다. TASK OFF 명령어를 사용하면 READY 상태에서 STANDBY 상태로 바뀝니다. 사용자 프로그램 전체를 태스크로 나누고 태스크 제어 명령어로 태스크를 기동 · 정지시키면 사이클 타임을 단축할 수 있습니다. 프로그램 전체 중 어느 부분을 언제 실행해야 하는지에 따라 프로그램을 태스크로 나눌 수 있습니다.



● 상태 천이



참고

STANDBY 상태는 점프(JMP-JME)와 완전히 동일하게 동작합니다. STANDBY 태스크의 출력 상태는 유지됩니다.

AI 번역본
-For Reference Only-

사이클릭 태스크의 사용

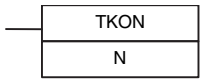
● 태스크 제어 명령어

프로그램에서 사이클릭 태스크를 기동 · 정지시키려면 태스크 제어 명령어를 사용합니다. 태스크의 기동 · 정지에 사용하는 명령어는 프로그래밍 언어에 따라 다음과 같이 다릅니다.

· 래더 프로그래밍 및 ST 프로그래밍: TKON(820)(TASK ON 명령어)와 TKOF(821)(TASK OFF 명령어)

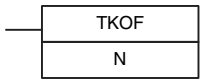
· SFC 프로그래밍: SFCON(789)(SFC ON 명령어)와 SFCOFF(790)(SFC OFF 명령어)

· TASK ON 명령어와 TASK OFF 명령어



N = 태스크 번호

실행 조건이 ON이면 태스크가 READY 상태가 됩니다. 해당 태스크 플래그가 ON됩니다.

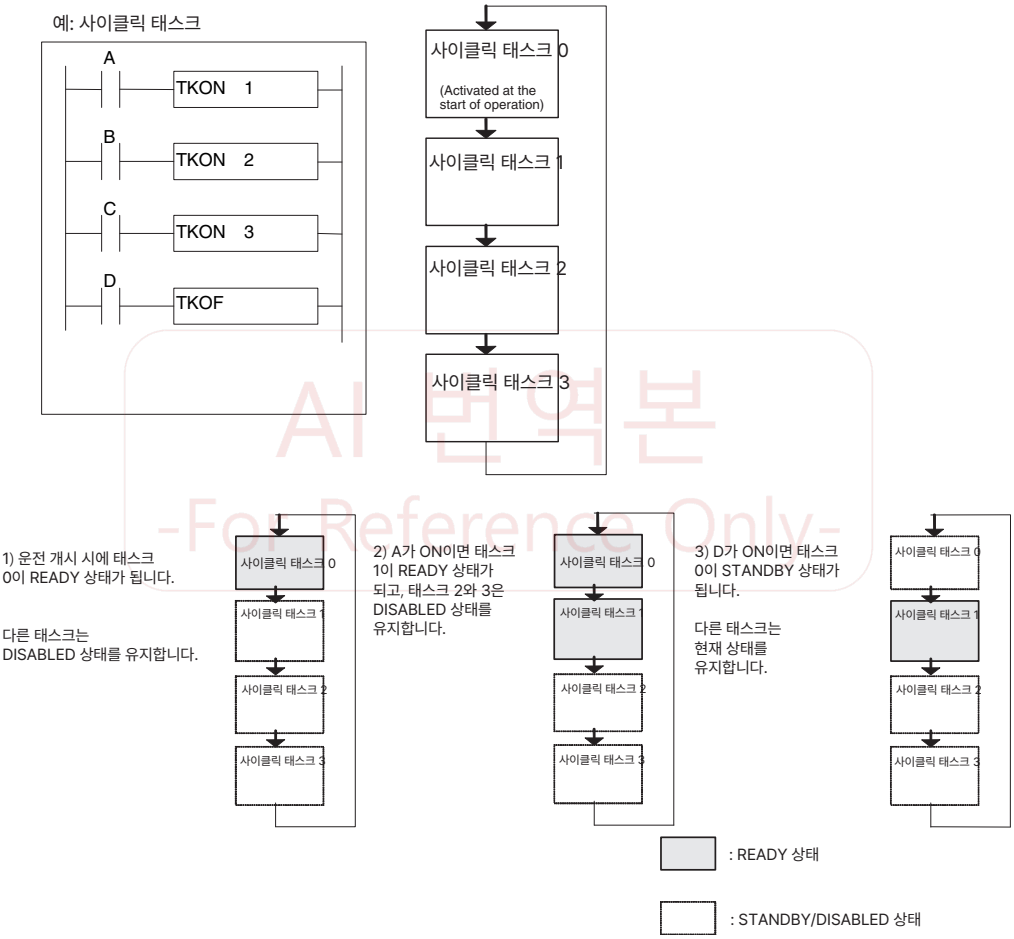


N = 태스크 번호

실행 조건이 ON이면 태스크가 STANDBY 상태가 됩니다. 해당 태스크 플래그가 OFF됩니다.

TASK ON 명령어와 TASK OFF 명령어를 사용하면 임의의 사이클릭 태스크를 임의의 시점에 READY 상태와 STANDBY 상태 사이에서 전환할 수 있습니다. READY 상태의 사이클릭 태스크는 이후 사이클에서도 그 상태를 유지하고, STANDBY 상태의 사이클릭 태스크도 이후 사이클에서 그 상태를 유지합니다. TASK ON 명령어와 TASK OFF 명령어는 사이클릭 태스크에서만 사용할 수 있습니다. 인터럽트 태스크에서는 사용할 수 없습니다.

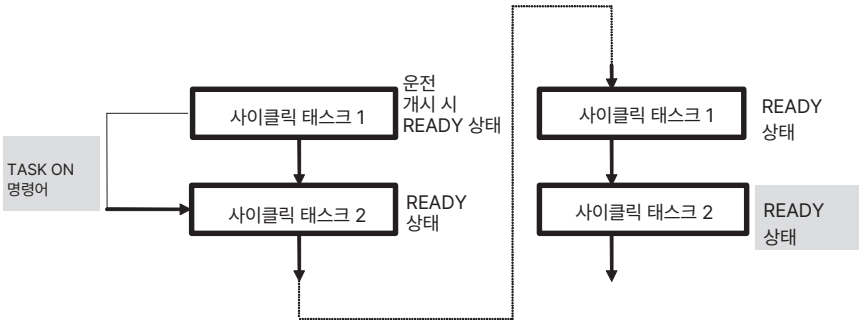
주 각 사이클에서 적어도 1개의 사이클릭 태스크가 READY 상태여야 합니다. READY 상태인 사이클릭 태스크가 하나도 없으면 태스크 에러 플래그(A295.12)가 ON되고 CPU 유닛이 정지합니다.



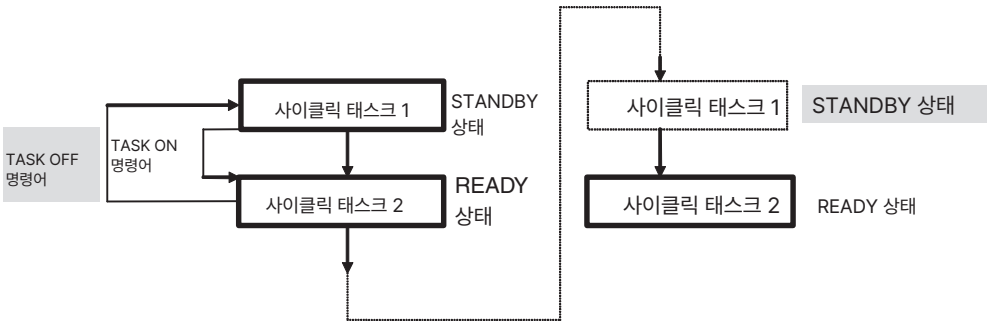
AI 번역본
-For Reference Only-

● 태스크와 실행 사이클

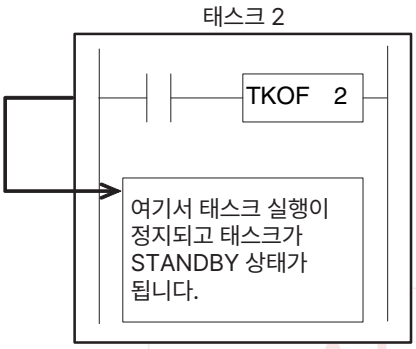
READY 상태인 사이클릭 태스크(엑스트라 사이클릭 태스크 포함)는 이후 사이클에서도 그 상태를 유지합니다.



STANDBY 상태인 사이클릭 태스크는 이후 사이클에서도 그 상태를 유지합니다. STANDBY 상태에서 READY 상태로 전환하려면 TASK ON 명령어로 그 태스크를 ON해야 합니다.



자기 태스크에 대해 TASK OFF 명령어를 실행하면, 그 태스크는 TASK OFF 명령어가 실행된 위치에서 실행이 정지되고 STANDBY 상태가 됩니다.



사이클릭 태스크 번호와 실행 사이클

- 태스크 m이 태스크 n을 ON하고 m > n이면, 태스크 n은 다음 사이클에서 READY 상태가 됩니다.
  예: 태스크 5가 태스크 2를 ON하면, 태스크 2는 다음 사이클에서 READY 상태가 됩니다.
- 태스크 m이 태스크 n을 ON하고 m < n이면, 태스크 n은 같은 사이클에서 READY 상태가 됩니다.
  예: 태스크 2가 태스크 5를 ON하면, 태스크 5는 같은 사이클에서 READY 상태가 됩니다.
- 태스크 m이 태스크 n을 STANDBY 상태로 하고 m > n이면, 다음 사이클에서 STANDBY 상태가 됩니다.
  예: 태스크 5가 태스크 2를 STANDBY 상태로 하면, 태스크 2는 다음 사이클에서 STANDBY 상태가 됩니다.
- 태스크 m이 태스크 n을 STANDBY 상태로 하고 m < n이면, 태스크 n은 같은 사이클에서 STANDBY 상태가 됩니다.
  예: 태스크 2가 태스크 5를 STANDBY 상태로 하면, 태스크 5는 같은 사이클에서 STANDBY 상태가 됩니다.

사 번역본
-For Reference Only-

태스크와 I/O 메모리의 관계

- 인덱스 레지스터(IR)와 데이터 레지스터(DR) 이외의 I/O 메모리는 태스크 간에 공유됩니다. 예를 들어 사이클릭 태스크 1에서 사용하는 CIO 10.00 비트와 사이클릭 태스크 2에서 사용하는 CIO 10.00 비트는 메모리상의 동일한 비트입니다. 인덱스 레지스터(IR)와 데이터 레지스터(DR) 이외의 I/O 메모리를 여러 태스크가 액세스하는 경우, 한 태스크에서 변경한 값을 다른 태스크에서 사용할 때에는 충분히 주의하십시오.

- 인덱스 레지스터(IR)와 데이터 레지스터(DR)의 사용 방법에는 두 가지가 있습니다. 1) 태스크별로 독립, 또는 2) 전체 태스크에서 공유.
  - 독립 레지스터인 경우, 예를 들어 사이클릭 태스크 1에서 사용하는 IR0과 사이클릭 태스크 2에서 사용하는 IR0은 서로 다릅니다.
  - 공유 레지스터인 경우, 예를 들어 사이클릭 태스크 1에서 사용하는 IR0과 사이클릭 태스크 2에서 사용하는 IR0은 동일합니다.

레지스터를 독립으로 할지 공유로 할지의 설정은 CX-Programmer에서 합니다.

I/O 메모리	태스크와의 관계
IR 영역 및 DR 영역을 제외한 CIO, 보조 기억, 데이터 메모리 등 모든 메모리 영역.*1	다른 태스크와 공유합니다.
인덱스 레지스터(IR) 및 데이터 레지스터(DR)*2	태스크마다 개별적으로 사용합니다.

*1 현재 EM 뱅크도 태스크 간에 공유됩니다. 따라서 예를 들어 사이클릭 태스크 1에서 현재 EM 뱅크 번호를 변경하면, 새로운 현재 EM 뱅크 번호는 사이클릭 태스크 2에도 유효합니다.

*2 인터럽트 태스크가 기동될 때 IR과 DR의 값은 설정되지 않습니다. 인터럽트 태스크에서 IR과 DR을 사용하는 경우에는 그 인터럽트 태스크 안에서 MOVR/MOVRW(MOVE TO REGISTER 및 MOVE TIMER/COUNTER PV TO REGISTER) 명령어로 값을 설정해야 합니다. 인터럽트 태스크 실행이 끝나면 IR과 DR은 인터럽트 이전의 값으로 자동으로 되돌아갑니다.

AI 번역본
-For Reference Only-

5-2-3 인터럽트 태스크

다음 조건 중 하나가 성립하면 인터럽트 태스크는 사이클 중 임의의 시점에 실행할 수 있습니다.

인터럽트가 발생하면, CPU 유닛이 사이클릭 태스크의 명령어를 실행 중이든 I/O 리프레시 중이든 주변 서비스 중이든 관계없이 사이클 중의 임의의 시점에 인터럽트 태스크가 실행됩니다.

인터럽트 태스크의 종류

사용할 수 있는 인터럽트 태스크의 종류는 다음과 같습니다.

- 전원 OFF 인터럽트 태스크
  전원 OFF 인터럽트 태스크는 전원이 OFF되기 직전에 실행됩니다.
- 정주기 인터럽트 태스크
  정주기 인터럽트 태스크는 일정 주기로 운전 상태를 모니터링하는 등, 사용자 프로그램의 특정 부분을 정해진 시간마다 실행해야 하는 프로그램에 사용합니다.
- I/O 인터럽트 태스크
  I/O 인터럽트 태스크는 CPU 유닛에 접속된 인터럽트 입력 유닛의 입력이 ON되었을 때 실행됩니다.
- 외부 인터럽트 태스크
  외부 인터럽트 태스크는 특수 I/O 유닛 또는 CPU 버스 유닛이 인터럽트를 요구했을 때 실행됩니다.
- 입력 인터럽트 태스크 (펄스 I/O 모듈) (CJ2M/CJ3M CPU 유닛 전용)
  입력 인터럽트 태스크는 CJ2M/CJ3M CPU 유닛에 접속된 펄스 I/O 모듈의 입력이 ON되었을 때 실행됩니다.
- 엑스트라 사이클릭 태스크
  엑스트라 사이클릭 태스크는 사이클릭 태스크로 취급되는 인터럽트 태스크입니다. 이 경우의 목적은 다른 인터럽트 태스크처럼 인터럽트 처리를 하는 것이 아니라 사이클릭 태스크의 개수를 늘리는 것입니다. 사이클릭 태스크는 최대 128개까지 사용할 수 있지만, 엑스트라 사이클릭 태스크를 사용하면 최대 384개까지 사용할 수 있습니다.

AI 번역본
-For Reference Only-

● 인터럽트 태스크 목록

태스크	태스크 번호	실행 조건	관련 설정
전원 OFF 인터럽트 태스크	1	CPU 유닛을 OFF했을 때	· PLC 설정의 OFF 인터럽트 태스크
정주기 인터럽트 태스크 0과 1*1	2와 3	CPU 유닛의 내부 타이머가 지정 시간을 경과할 때마다	· MSKS(690)(SET INTER-RUPT MASK) 명령어로 인터럽트 주기를 설정합니다. (1~9999) · PLC 설정의 Scheduled Interrupt Time Units를 참조하십시오. (10 ms/1.0 ms/0.1 ms)
I/O 인터럽트 태스크 0 ~31*2	100~131	CPU 랙의 인터럽트 입력 유닛의 입력 비트가 ON될 때	· MSKS(690)(SET INTER-RUPT MASK) 명령어로 CPU 랙의 인터럽트 입력 유닛의 입력을 할당합니다.
외부 인터럽트 0 ~255*3	0~255	CPU 랙의 특수 I/O 유닛 또는 CPU 버스 유닛이 요구했을 때*3	없음 (항상 유효)
입력 인터럽트 태스크 (펄스 I/O 모듈)	140~147	CJ2M/CJ3M CPU 유닛에 접속된 펄스 I/O 모듈의 입력이 ON 또는 OFF될 때, 또는 증분 카운터나 감분 카운터가 카운트 아웃될 때.	· MSKS(690)(SET INTER-RUPT MASK) 명령어로 지정한 입력의 인터럽트 마스크를 해제합니다.
엑스트라 사이클릭 태스크 0 ~255	0~255	태스크가 READY 상태인 동안 매 사이클. (태스크 제어 명령어를 사용해야 합니다.)	태스크 제어 명령어(TKON)로 태스크를 ON한다

*1 동기 유닛 동작을 사용하는 경우에는 정주기 인터럽트 태스크를 사용할 수 없습니다.

*2 인터럽트 입력 유닛은 CPU 랙에 장착해야 합니다. 다른 위치에 장착된 I/O 인터럽트 유닛은 I/O 인터럽트 태스크의 실행 요구에 사용할 수 없습니다.

*3 특수 I/O 유닛 또는 CPU 버스 유닛은 CPU 랙에 장착해야 합니다. 다른 위치에 장착된 유닛은 외부 인터럽트의 발생에 사용할 수 없습니다.

사용상의 주의

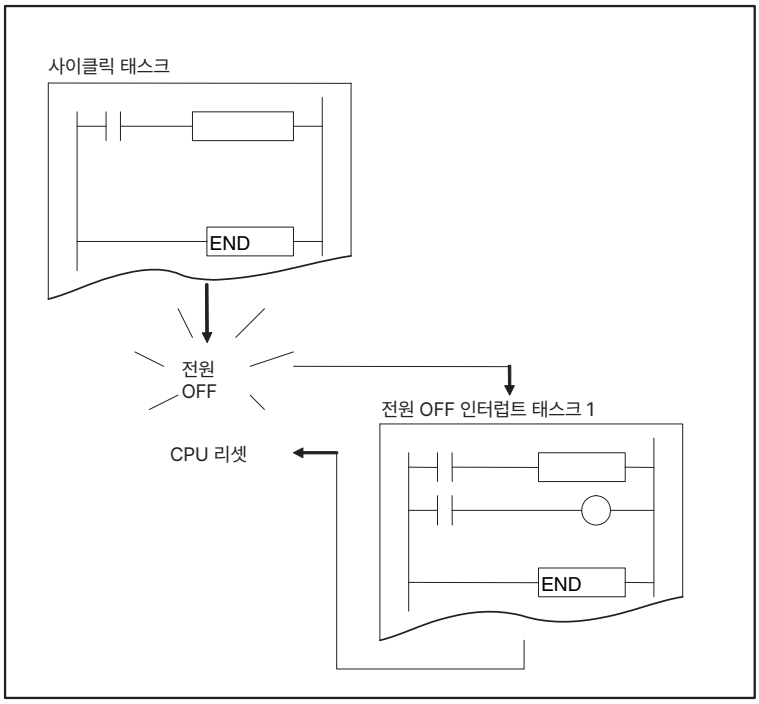
인터럽트 태스크에서는 SFC 프로그램을 사용하지 마십시오. 정상적으로 실행되지 않습니다. 단, 인터럽트 태스크를 엑스트라 사이클릭 태스크로 실행하는 경우에는 SFC 프로그램을 사용할 수 있습니다.

인터럽트 태스크에서 ST 프로그램은 사용할 수 있습니다.

AI 번역본
-For Reference Only-

전원 OFF 인터럽트 태스크

이 태스크는 정전이 발생했을 때 실행됩니다. 정전이 발생하면 전원 유닛은 CPU 유닛에 5 V 전원을 10 ms 동안 계속 공급하며, 그동안 전원 OFF 인터럽트 태스크가 실행됩니다. (CJ1W-PD022 전원 유닛을 사용하는 경우에는 전원이 1 ms 동안만 공급되므로 전원 OFF 인터럽트 태스크를 사용할 수 없습니다.)



● 전원 OFF 인터럽트 태스크의 인터럽트 오버헤드 타임

전원 OFF 인터럽트 태스크는 정전이 확정된 후 0.1 ms 이내에 실행됩니다.

● 전원 OFF 인터럽트 태스크 실행 설정

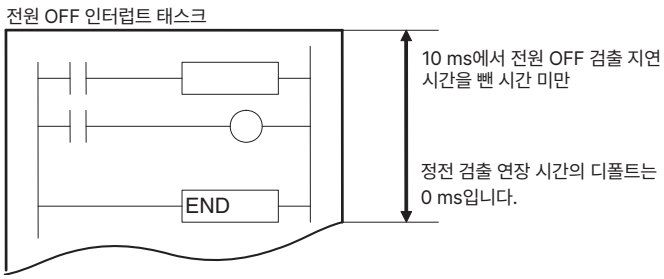
CX-Programmer의 PLC Settings 대화 상자에 있는 Timings/Synchronous 탭 페이지에서 전원 OFF 인터럽트 태스크를 유효하게 설정합니다. 디폴트 설정은 OFF입니다.

디폴트 PLC 설정에서는 전원 OFF 인터럽트 태스크가 10 ms 후에 정지됩니다. 전원 OFF 인터럽트 태스크는 10 ms 이내에 실행을 끝내야 합니다.

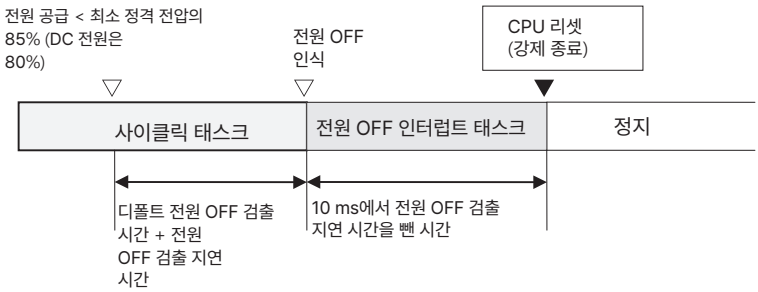
PLC 설정에서 전원 OFF 검출 지연 시간을 설정한 경우, 전원 OFF 인터럽트 태스크는 10 ms에서 PLC 설정의 전원 OFF 검출 지연 시간을 뺀 시간이 지난 후에 정지됩니다. 이 경우 전원 OFF 인터럽트 태스크는 10 ms에서 PLC 설정의 전원 OFF 검출 지연 시간을 뺀 시간 이내에 실행을 끝내야 합니다.

For Reference Only

예: PLC 설정에서 전원 OFF 검출 지연 시간을 4 ms로 설정한 경우, 실행 시간은 10에서 4 ms를 뺀 6 ms 미만이어야 합니다.



주 전원 공급이 최소 정격 전압의 85% 미만으로 떨어지면 정전으로 인식되며 (DC 전원은 80%), 전원 OFF 인터럽트 태스크가 실제로 실행되기까지의 시간은 디폴트 전원 OFF 검출 시간 (AC 전원은 10~25 ms, DC 전원은 2~5 ms)에 PLC 설정의 전원 OFF 검출 지연 시간(0~10 ms)을 더한 시간입니다. 그 시간 동안에는 사이클릭 태스크가 실행됩니다.



- **전원 OFF 인터럽트 태스크 사용 시의 제한**

온라인 에디팅 중에는 실행되지 않습니다

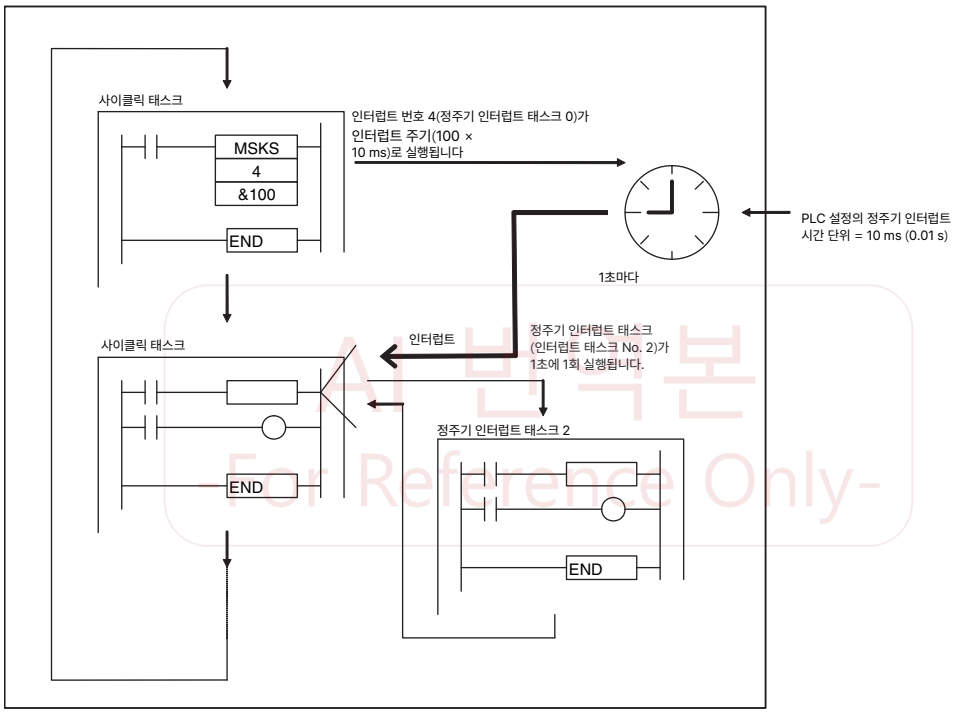
온라인 에디팅 중에 정전이 발생하면 전원 OFF 인터럽트 태스크는 실행되지 않습니다.

전원 OFF 인터럽트 태스크에서 사용할 수 없는 명령어

모든 인터럽트 태스크에서 사용할 수 없는 명령어 외에도, 전원 OFF 인터럽트 태스크에서 사용할 수 없는 명령어가 있습니다. 자세한 내용은 5-2-4 태스크 설계를 참조하십시오.

정주기 인터럽트 태스크

CPU 유닛의 타이머를 사용하여 지정한 시간 간격으로 태스크를 실행합니다. 정주기 인터럽트 0과 1에는 최대 2개의 인터럽트 태스크(인터럽트 태스크 2와 3)를 사용할 수 있습니다.



A 번 국 본
-For Reference Only-

● 인터럽트 태스크 번호와 인터럽트 번호

	인터럽트 태스크 No.	인터럽트 번호 (MSKS(690))
정주기 인터럽트 0	2	4
정주기 인터럽트 1	3	5

● 정주기 인터럽트 태스크의 실행

정주기 인터럽트 태스크는 CPU 유닛의 운전이 시작될 때 무효 상태입니다. MSKS(690)(SET INTERRUPT MASK) 명령어로 유효하게 할 수 있습니다. 정주기 인터럽트 태스크의 시간 간격은 PLC 설정의 정주기 인터럽트 시간 단위 설정과, MSKS(690) 실행 시 오퍼랜드로 지정하는 정주기 인터럽트 시간 설정에 따라 결정됩니다.

・정주기 인터럽트 시간 단위 설정
MSKS(690)으로 설정하는 인터럽트 시간의 단위는 CX-Programmer의 PLC Settings 대화 상자에 있는 Timings/Synchronous 탭 페이지의 "Scheduled Interrupt Interval" 설정에서 설정합니다.

단위는 10 ms, 1.0 ms, 0.1 ms 중에서 설정할 수 있습니다. 디폴트는 10 ms입니다.

・MSKS(690)을 사용한 정주기 인터럽트 시간 설정
정주기 인터럽트 태스크를 실행할 주기를 설정합니다.

단위는 &1~&9999(10진수)(#0001~#270F 16진수) 범위로 설정할 수 있습니다.

예: 정주기 인터럽트 시간 단위를 0.1 ms로 설정하고 정주기 인터럽트 시간을 &5(10진수)로 설정한 경우, 시간 간격은 0.1 x 5 = 0.5 ms입니다.

정주기 인터럽트 태스크에 설정할 수 있는 최소 시간 간격은 다음과 같습니다.

・CJ2H CPU 유닛: 0.2 ms

・CJ2M/CJ3M CPU 유닛: 0.4 ms

단, 유닛 버전 1.1 이후의 CJ2H CPU 유닛에서는 고속 인터럽트 기능을 사용하여 정주기 인터럽트 0(인터럽트 태스크 2)의 인터럽트 주기를 0.1 ms로 설정할 수 있습니다. 이 설정은 다른 인터럽트에는 사용할 수 없습니다. 고속 인터럽트에 대한 자세한 내용은 10-2-6 고속 인터럽트 기능을 참조하십시오.

사용상의 주의

・정주기 인터럽트의 시간 간격을 짧게 하여 정주기 인터럽트 태스크의 실행 빈도를 높이면, 사이클릭 태스크와 엑스트라 사이클릭 태스크의 실행이 완료되기까지의 시간과 전체 사이클 타임이 길어지므로 주의하십시오.

・동기 유닛 동작을 사용하는 경우에는 정주기 인터럽트 태스크를 사용할 수 없습니다.

● 정주기 인터럽트 태스크의 인터럽트 오버헤드 시간

MSKS(690)로 설정한 지정 시간이 경과한 후 인터럽트 태스크가 실제로 실행될 때까지의 시간을 정주기 인터럽트 태스크 기동 시간이라고 합니다. 인터럽트 태스크 프로그램이 실행된 후 중단되었던 처리로 복귀할 때까지 걸리는 시간을 사이클릭 태스크 복귀 시간이라고 합니다. 인터럽트 태스크 기동 시간과 사이클릭 태스크 복귀 시간을 합한 것을 인터럽트 오버헤드 타임이라고 합니다.

	항목	시간	
		CJ2H CPU 유닛	CJ2M/CJ3M CPU 유닛
정주기 인터럽트 태스크의 인터럽트 오버헤드 타임	인터럽트 태스크 기동 시간	22 μs 또는 13 μs*1 (유닛 버전 1.0은 27 μs)	30 μs
	사이클릭 태스크 복귀 시간	11 μs 또는 8 μs*1 (유닛 버전 1.0은 15 μs)	11 μs

*1 고속 인터럽트 기능 사용 시
고속 인터럽트 기능에 대한 자세한 내용은 10-2-6 고속 인터럽트 기능을 참조하십시오.

● MSKS(690)에 의한 리셋과 재기동

CJ2M/CJ3M CPU 유닛에서는 MSKS(690)로 정주기 인터럽트를 기동할 때 내부 타이머를 리셋하도록 지정할 수 있습니다("리셋 스타트"). 이에 따라 CLI(691)를 사용하지 않아도 첫 번째 인터럽트가 기동될 때까지의 시간을 일정하게 할 수 있습니다. 정주기 인터럽트를 기동할 때 정주기 인터럽트 시간(즉, 한 인터럽트에서 다음 인터럽트까지의 간격)은 MSKS(690)로 설정합니다. 그러나 MSKS(690)가 실행된 후 정주기 인터럽트 태스크가 처음 기동될 때까지의 시간은 내부 타이머의 현재값에 따라 달라집니다. 따라서 CLI(691)도 함께 실행하지 않으면 첫 번째 간격까지의 시간이 일정하지 않게 됩니다. 그러나 CJ2M/CJ3M CPU 유닛에서는 기동 시에 내부 타이머를 리셋할 수 있으므로 CLI(691)를 실행하지 않아도 첫 번째 인터럽트까지의 시간이 일정해집니다.

정주기 인터럽트용 MSKS(690) 오퍼랜드

오퍼랜드	설정값
N (인터럽트 식별자)	4: 정주기 인터럽트 0, 통상 동작(내부 타이머 리셋 없음, 인터럽트 태스크 2) 5: 정주기 인터럽트 1, 통상 동작(내부 타이머 리셋 없음, 인터럽트 태스크 3) 14: 정주기 인터럽트 0, 리셋 스타트(CJ2M/CJ3M CPU 유닛 전용, 인터럽트 태스크 2) 15: 정주기 인터럽트 1, 리셋 스타트(CJ2M/CJ3M CPU 유닛 전용, 인터럽트 태스크 3)

● MSKR(692)에 의한 내부 타이머 현재값 읽기

CJ2M/CJ3M CPU 유닛에서는 정주기 인터럽트 시간을 계측하는 데 사용되는 내부 타이머의 현재값을 읽을 수 있습니다. 정주기 인터럽트가 기동될 때까지 경과한 시간 또는 이전 정주기 인터럽트로부터의 시간 중 하나를 읽을 수 있습니다. 내부 타이머의 현재값을 읽는 데에는 MSKR(692)를 사용합니다. 시간 단위는 정주기 인터럽트 시간에 사용되는 단위, 즉 PLC 설정에서 설정한 정주기 인터럽트 인터벌과 동일합니다.

정주기 인터럽트용 MSKR(692) 오퍼랜드

오퍼랜드	설정값
N (인터럽트 식별자)	4: 인터럽트 태스크 0의 정주기 인터럽트 시간(설정값) 읽기(인터럽트 태스크 2) 5: 인터럽트 태스크 1의 정주기 인터럽트 시간(설정값) 읽기(인터럽트 태스크 3) 14: 인터럽트 태스크 0의 내부 타이머 현재값 읽기(CJ2M/CJ3M CPU 유닛 전용, 인터럽트 태스크 2) 15: 인터럽트 태스크 1의 내부 타이머 현재값 읽기(CJ2M/CJ3M CPU 유닛 전용, 인터럽트 태스크 3)

I/O 인터럽트 태스크

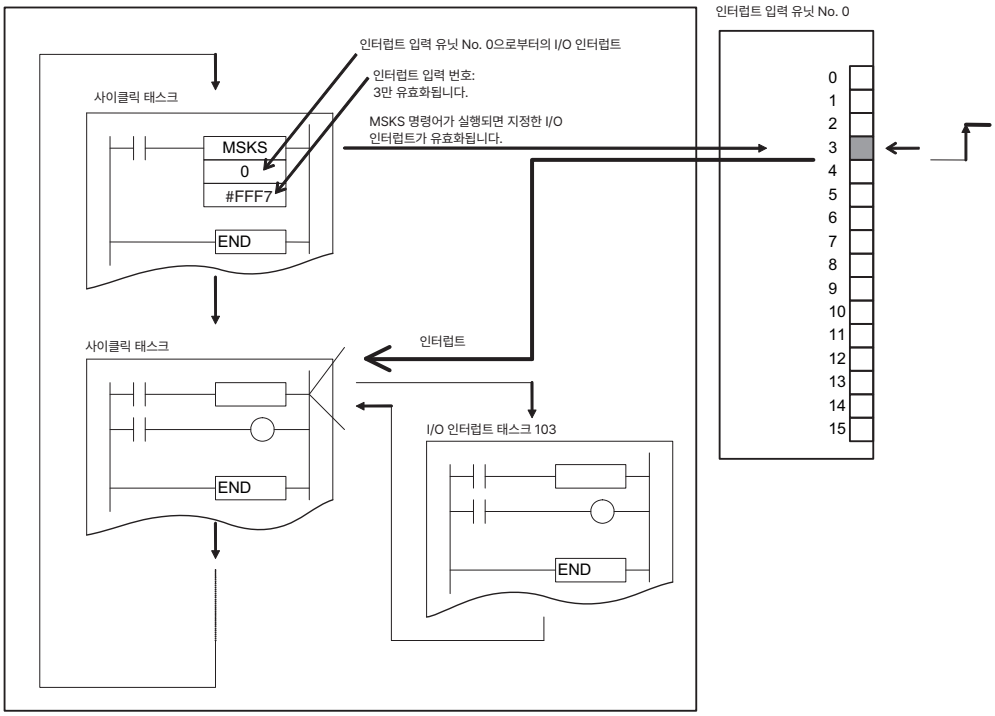
인터럽트 입력 유닛의 입력이 ON 되면 I/O 인터럽트 태스크가 실행됩니다. 작성할 수 있는 태스크의 최대 개수는 32개입니다(인터럽트 태스크 번호 100~131).

AI 번역본
For Reference Only

● 인터럽트 태스크의 실행

I/O 인터럽트를 유효화하려면 MSKS(690)(SET INTERRUPT MASK) 명령어를 실행하십시오.

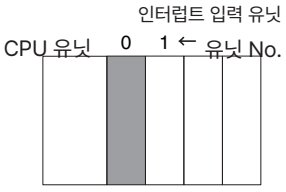
예: 다음 예는 인터럽트 입력 유닛 No. 0(유닛 0과 1 중 가장 왼쪽)의 인터럽트 입력 No. 3이 ON 되었을 때 I/O 인터럽트 태스크 103을 실행하는 예입니다.



인터럽트 입력 유닛 번호, 입력 번호, I/O 인터럽트 태스크 번호

인터럽트 입력 유닛 No.*1	입력 No.	I/O 인터럽트 태스크
0	0~15	100~115
1	0~15	116~131

*1 인터럽트 입력 유닛 번호는 CPU 유닛에서부터 순서대로 0~1입니다.



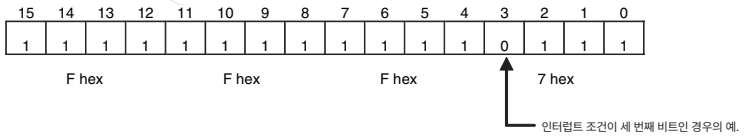
● MSKS(690)

I/O 인터럽트 태스크를 유효화하려면 사이클릭 태스크의 프로그램에서 MSKS(690)(SET INTERRUPT MASK) 명령어를 실행하여 지정한 인터럽트 유닛의 지정한 인터럽트 번호에 대한 인터럽트를 유효화하십시오.

AI 번역본
-For Reference Only-

MSKS(690)의 오퍼랜드 S(제2 오퍼랜드):
FFF7 hex의 각 비트는 인터럽트 입력 유닛의 인터럽트 입력에 대응합니다. 인터럽트 입력 번호 0~15는 비트 0~15에 대응합니다.

예:



사용상의 주의

불필요한 I/O 인터럽트 태스크를 유효화하지 마십시오. 노이즈로 인해 인터럽트 입력이 발생했는데 해당하는 인터럽트 태스크가 없으면 치명 에러(태스크 에러)로 프로그램이 정지합니다.

● I/O 인터럽트 태스크의 인터럽트 오버헤드 시간

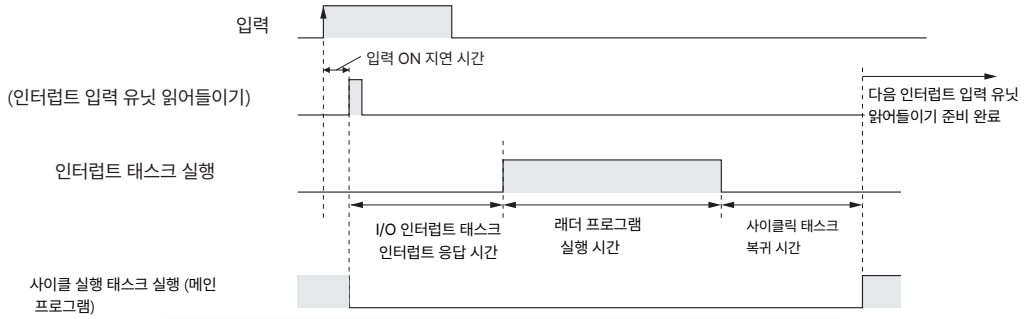
인터럽트 입력 유닛(CJ1W-INT01)의 입력 비트가 ON(또는 OFF) 되고 나서 CPU 유닛이 인터럽트 신호를 수신할 때까지의 시간은 하드웨어 응답 시간에 따라 달라집니다. 인터럽트 신호를 수신하고 나서 I/O 인터럽트 태스크가 실제로 실행될 때까지의 시간을 I/O 인터럽트 태스크의 인터럽트 태스크 기동 시간이라고 합니다. 인터럽트 태스크 프로그램이 실행된 후 중단 전의 처리로 복귀할 때까지 걸리는 시간을 사이클릭 태스크 복귀 시간이라고 합니다. 인터럽트 태스크 기동 시간과 사이클릭 태스크 복귀 시간을 합한 것을 인터럽트 오버헤드 시간이라고 합니다.

인터럽트 입력 유닛 사용 시

항목		시간	
		CJ2H CPU 유닛	CJ2M/CJ3M CPU 유닛
하드웨어 응답(CJ1W-INT01)		상승 미분: 0.05 ms, 하강 미분: 0.5 ms	
인터럽트 오버헤드 시간	인터럽트 태스크 기동 시간	26 μs 또는 17 μs*1 (유닛 버전 1.0은 30 μs)	31 μs
	사이클릭 태스크 복귀 시간	11 μs 또는 8 μs*1 (유닛 버전 1.0은 15 μs)	10 μs

*1 고속 인터럽트 기능 사용 시
고속 인터럽트 기능에 대한 자세한 내용은 10-2-6 고속 인터럽트 기능을 참조하십시오.

주 I/O 인터럽트 태스크는 사용자 프로그램 실행, I/O 리프레시, 주변 서비스 또는 오버헤드 처리 중에 실행될 수 있습니다. (명령어가 실행 중이더라도 I/O 인터럽트 태스크는 실행될 수 있습니다. 그 명령어는 중단됩니다.) 인터럽트 태스크 실행 중에 입력 인터럽트가 발생한 경우에는 즉시 처리되지 않습니다. 현재의 인터럽트 태스크가 먼저 끝까지 실행되고, 그 후 사이클릭 태스크 복귀 시간과 인터럽트 태스크 기동 시간이 경과한 뒤에 새 인터럽트의 실행이 개시됩니다.



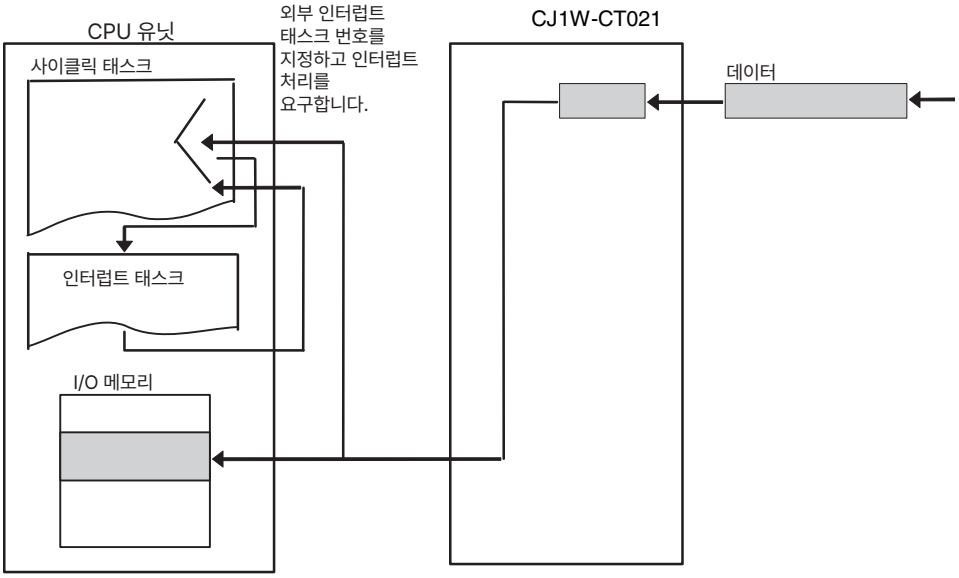
외부 인터럽트

특수 I/O 유닛 또는 CPU 버스 유닛을 포함하는 PLC에서는 인터럽트 처리가 CPU 유닛에서 수행됩니다. 프로그램에 특정 태스크 번호의 외부 인터럽트 태스크가 포함되어 있지 않은 한 CPU 유닛에서 설정할 필요는 없습니다. 외부 인터럽트를 유효화하려면 특수 I/O 유닛 또는 CPU 버스 유닛을 CPU 랙에 접속해야 합니다.

외부 인터럽트 태스크의 수신은 항상 유효합니다. 태스크의 최대 개수는 256개입니다(인터럽트 태스크 번호 0~255).

예: CJ1W-CT021 고속 카운터 유닛으로부터의 외부 인터럽트

시 번역본
for reference only



이 유닛의 외부 제어 입력 2점과 출력 32점의 상태 정보를 I/O 리프레시 이외의 방법으로 CPU 유닛에 전달하려면, 외부 제어 입력 또는 출력이 외부 인터럽트 태스크를 기동하도록 설정하십시오.

사용상의 주의

둘 이상의 인터럽트 태스크에 같은 번호를 사용하지 마십시오.

외부 인터럽트 태스크(0~255)가 전원 OFF 태스크(태스크 1), 정주기 인터럽트 태스크(태스크 2 또는 3) 또는 I/O 인터럽트 태스크(100~131)와 같은 번호를 가지면, 해당 인터럽트 태스크는 어느 쪽 인터럽트 조건(외부 인터럽트 또는 다른 인터럽트 조건)에서도 실행됩니다.

● 펄스 I/O 모듈로부터의 외부 인터럽트

CJ2M/CJ3M CPU 유닛과 함께 펄스 I/O 모듈을 사용하는 경우, 고속 카운터 입력의 비교 결과에 따라 외부 인터럽트 태스크를 실행할 수 있습니다. 목표값 비교에서는 비교 조건(목표값과 계수 방향)을 해당 인터럽트 태스크 번호와 함께 CTBL(882) 비교표에 등록합니다. 고속 카운터의 현재값이 등록한 목표값과 일치하면 지정한 인터럽트 태스크가 실행됩니다.

범위 비교에서는 비교 조건(상한값과 하한값, 범위 진입 또는 이탈)을 해당 인터럽트 태스크 번호와 함께 CTBL(882) 비교표에 등록합니다. 비교 조건이 충족되면 지정한 인터럽트 태스크가 실행됩니다.

AI 번역본

● 외부 인터럽트 태스크의 오버헤드 시간

외부 인터럽트 태스크에서 CPU 유닛이 인터럽트 신호를 수신할 때까지의 시간은, CPU 유닛에 외부 인터럽트 태스크의 실행을 요구하는 유닛 또는 보드의 종류(즉, 특수 I/O 유닛 또는 CJ 시리즈 CPU 버스 유닛)와 인터럽트 요구의 서비스 종류에 따라 달라집니다. 자세한 내용은 해당 유닛의 매뉴얼을 참조하십시오.

CPU 유닛이 인터럽트 신호를 수신한 후의 인터럽트 오버헤드 타임은 I/O 인터럽트 태스크와 동일합니다.

입력 인터럽트 태스크(펄스 I/O 모듈)(CJ2M/CJ3M CPU 유닛 전용)

CJ2M/CJ3M CPU 유닛에 접속된 펄스 I/O 모듈의 입력을 인터럽트 입력으로 사용하는 경우, 입력이 ON 또는 OFF 될 때(Direct Mode라고 합니다) 또는 증가 카운터나 감소 카운터가 카운트 아웃할 때(Counter Mode라고 합니다) 인터럽트 태스크가 실행됩니다. 각 입력에 대해 실행되는 인터럽트 태스크는 고정되어 있어 변경할 수 없습니다(인터럽트 태스크 140~147). 인터럽트를 유효화하고 동작 모드를 Direct Mode 또는 Counter Mode로 설정하는 데에는 SET INTERRUPT MASK 명령어(MSKS(690))를 사용합니다.

추가 사이클릭 태스크

인터럽트 태스크는 일반 사이클릭 태스크와 마찬가지로 매 사이클 실행될 수 있습니다. 추가 사이클릭 태스크(인터럽트 태스크 번호 0~255)는 일반 사이클릭 태스크(사이클릭 태스크 번호 0~127)의 실행이 완료된 후 가장 작은 태스크 번호부터 순서대로 실행됩니다. 추가 사이클릭 태스크의 최대 개수는 256개입니다(인터럽트 태스크 번호: 0~255). 추가 태스크는 태스크 제어 명령어(TKON)를 사용하여 기동하십시오.

사용상의 주의

엑스트라 사이클릭 태스크에 할당된 태스크 번호를 인터럽트 태스크에 사용하지 마십시오. 전원 OFF 인터럽트 태스크, 정주기 인터럽트 태스크 또는 I/O 인터럽트 태스크와 같은 태스크 번호를 가진 엑스트라 사이클릭 태스크는 엑스트라 사이클릭 태스크로서도 인터럽트 태스크로서도 실행됩니다.

주1 TKON(820)과 TKOF(821)은 엑스트라 사이클릭 태스크에 입력하여 실행할 수 있지만, 그 태스크가 인터럽트 태스크로서 실행될 때에는 실행되지 않습니다.

2 일반 사이클릭 태스크와 엑스트라 사이클릭 태스크의 차이는 다음 표와 같습니다.

항목	엑스트라 사이클릭 태스크	일반 사이클릭 태스크
기동 시 활성화	설정할 수 없습니다.	CX-Programmer에서 설정
태스크 플래그	지원하지 않습니다.	지원합니다. (사이클릭 태스크 번호 0~127은 태스크 플래그 TK000~TK127에 대응합니다.)
초기 태스크 실행 플래그 (A200.15)와 태스크 기동 플래그 (A200.14)	지원하지 않습니다.	지원합니다.
인덱스(IR) 및 데이터(DR) 레지스터 값	태스크가 기동될 때 정의되지 않습니다 (일반 인터럽트 태스크와 동일). 각 사이클 시작 시점의 값은 정의되지 않습니다. 사용하기 전에 반드시 값을 설정하십시오. 이전 사이클에서 설정한 값은 읽을 수 없습니다.	동작 시작 시점에는 정의되지 않습니다. 이전 사이클에서 설정한 값을 읽을 수 있습니다.

SAMPLE
-For Reference Only-

5-2-4 태스크 설계

가이드라인

신뢰성이 높고 메인터넌스가 쉬운 시스템을 구축하려면 태스크 설계가 중요합니다. 다음 사항에 주의하십시오.

・다음을 기준으로 프로그래밍을 태스크로 분할합니다.

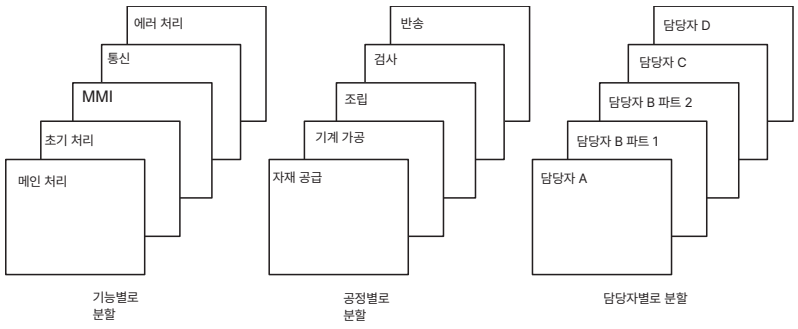
・상태 천이를 이해한 후 실행 조건과 비실행 조건을 구체적으로 검토합니다.

・외부 I/O의 유무를 검토합니다.

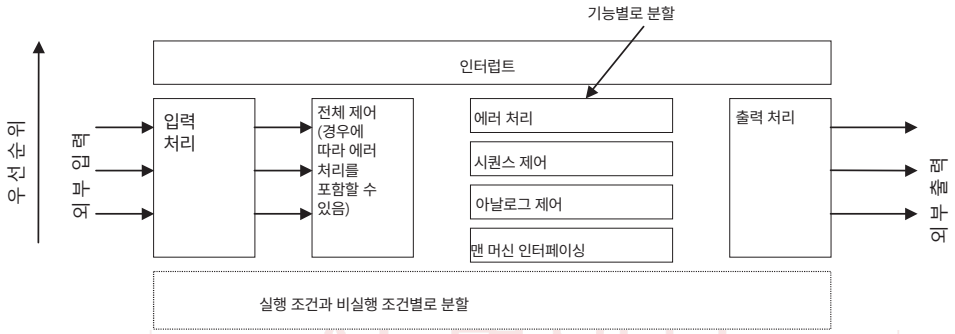
・기능과 담당자를 검토합니다.

・우선순위에 따른 실행을 검토합니다. 처리를 사이클릭 태스크와 인터럽트 태스크로 나눕니다. 우선순위가 가장 높은 태스크에 가장 작은 번호를 할당합니다.

예 1: 기능과 담당자에 따른 태스크 설계.



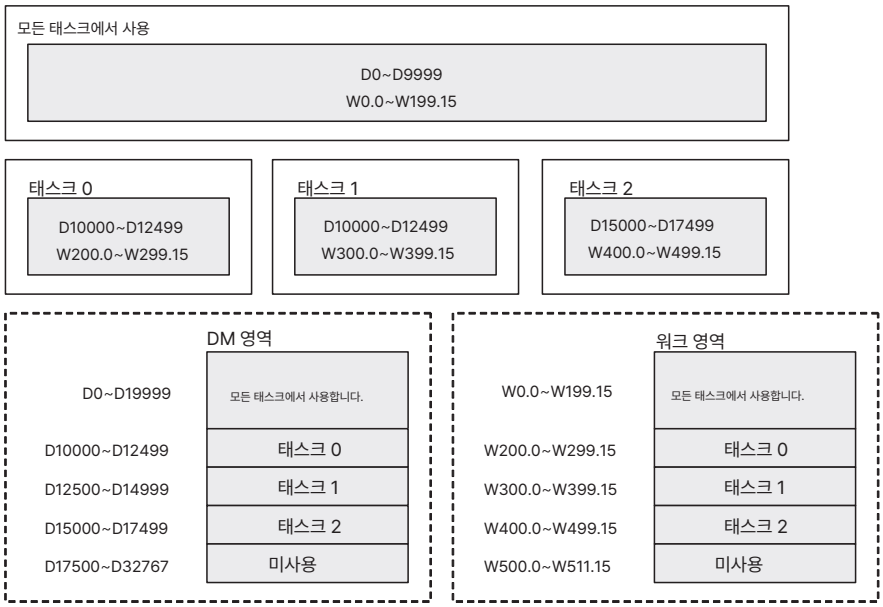
예 2: 실행 우선순위에 따른 태스크 설계
처리 태스크보다 제어 태스크에 더 작은 번호를 할당합니다.



・자립성이 확보되도록 프로그램을 분할하여 설계합니다. 태스크 간에 교환하는 데이터의 양을 최소한으로 억제합니다.

모든 태스크에서 사용하는 I/O 메모리와 개별 태스크에서만 사용하는 I/O 메모리를 정리하십시오. 각 태스크 전용으로 사용할 I/O 메모리 범위를 정리하여 할당하십시오.

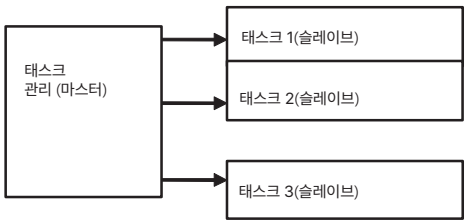
AI 번역본
For Reference Only-



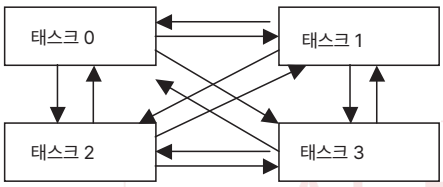
* D20000~D29599와 D30000~D31599는 버스 유닛용 DM 영역으로 사용됩니다. 사용 중인 버스 유닛의 영역이 서로 중복되지 않도록 설계하십시오.

・하나의 태스크로 다른 태스크를 기동 · 정지시켜 사용자 프로그램 전체를 제어하십시오.

좋은 예: 태스크 간에 마스터-슬레이브 관계를 정의합니다.



나쁜 예: 태스크의 기동과 정지를 체계 없이 수행하면 제어가 어려워집니다.



・우선순위가 높은 인터럽트 태스크에 더 작은 번호를 할당하십시오.

・READY 상태의 태스크는 그 태스크 자신 또는 다른 태스크가 STANDBY 상태로 바꾸지 않는 한 이후의 사이클에서도 계속 실행됩니다. 태스크 간에 처리를 분기시키려면 다른 태스크에 대해 반드시 TKOF(821)(TASK OFF) 명령어를 삽입하십시오.

・태스크를 초기화하는 명령어의 실행 조건에는 초기 태스크 실행 플래그(A200.15) 또는 태스크 기동 플래그(A200.14)를 사용하십시오. 초기 태스크 실행 플래그는 각 태스크의 최초 실행 중에 ON 됩니다. 태스크 기동 플래그는 태스크가 READY 상태가 될 때마다.

-For Reference Only-

참고

글로벌 서브루틴

일반 서브루틴 명령어로는 한 태스크의 서브루틴을 다른 태스크에서 호출할 수 없습니다. 글로벌 서브루틴은 인터럽트 태스크 번호 0에 작성할 수 있으며, 이 서브루틴은 임의의 사이클릭 태스크(엑스트라 사이클릭 태스크 포함)에서 호출할 수 있습니다.

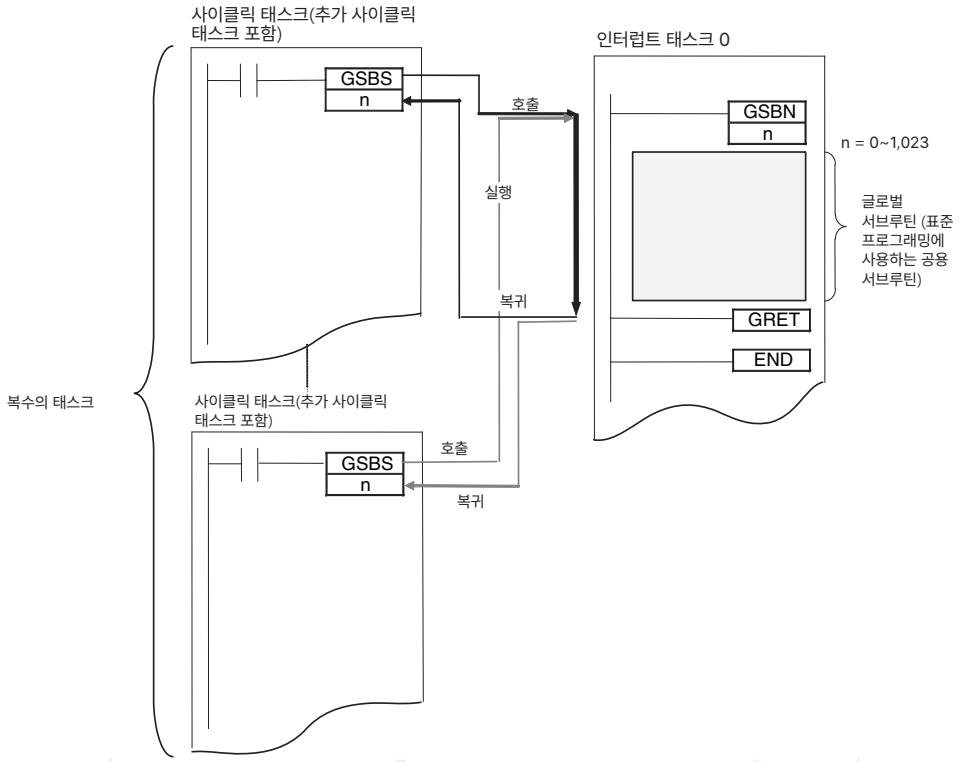
글로벌 서브루틴을 호출하는 데에는 GSBS 명령어를 사용합니다. 서브루틴 번호는 0에서 1,023 사이여야 합니다. 글로벌 서브루틴은 인터럽트 태스크 번호 0의 끝(END(001) 바로 앞)에서 GSBN 명령어와 GRET 명령어 사이에 정의합니다.

글로벌 서브루틴을 사용하면 필요할 때마다 호출할 수 있는 표준 프로그램 섹션의 라이브러리를 작성할 수 있습니다.

글로벌 서브루틴은 인터럽트 태스크 번호 0 안에만 배치되므로, 인터럽트 태스크가 아니라 사이클릭 태스크로서 실행됩니다.

그 때문에 글로벌 서브루틴이 실행 중일 때 인터럽트가 발생하면, 인터럽트 태스크가 글로벌 서브루틴의 실행을 일시적으로 중단하고 해당 태스크가 실행됩니다.

해당 태스크의 실행이 끝나면 처리는 마지막 중단점부터 글로벌 서브루틴의 실행을 재개합니다.



-For Reference Only-

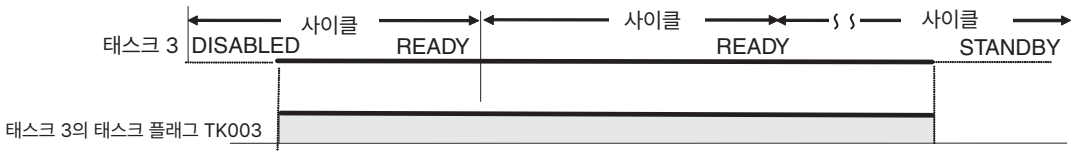
태스크 관련 플래그

● 사이클릭 태스크 관련 플래그

다음 플래그는 일반 사이클릭 태스크에서만 동작합니다. 엑스트라 사이클릭 태스크에서는 동작하지 않습니다.

태스크 플래그(TK000~TK127)

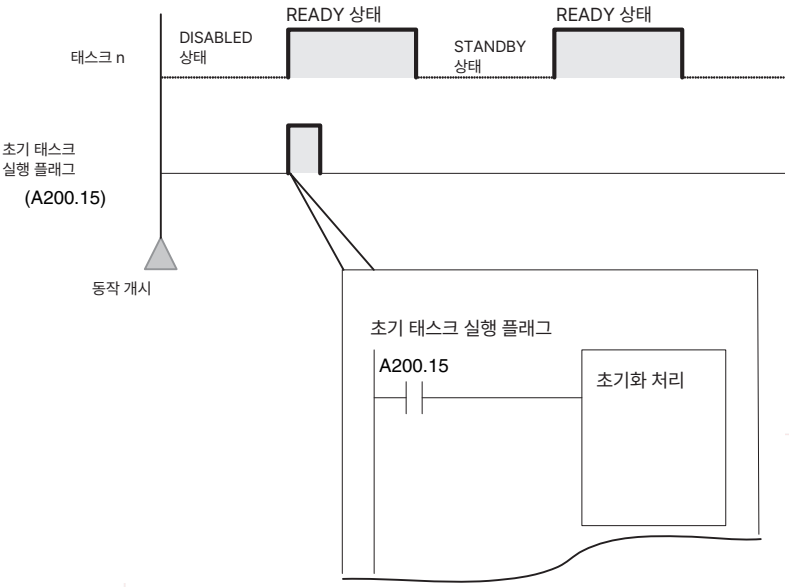
이 플래그는 태스크가 현재 실행 중인지 확인하는 데 사용합니다. 태스크 플래그는 사이클릭 태스크가 READY 상태일 때 ON 되고, 태스크가 DISABLED(INI) 상태 또는 STANDBY(WAIT) 상태일 때 OFF 됩니다. 태스크 번호 00~127은 TK000~TK127에 대응합니다.



주 태스크 플래그는 사이클릭 태스크에서만 사용되며 인터럽트 태스크에서는 사용되지 않습니다. 인터럽트 태스크에는 워드 A440(최대 인터럽트 태스크 처리 시간)이나 워드 A441(최대 처리 시간 인터럽트 태스크)과 같은 인터럽트 태스크 관련 플래그를 사용하십시오. 인터럽트 태스크 관련 플래그는 아래를 참조하십시오.

초기 태스크 실행 플래그(A200.15:P_First_Cycle_Task)

이 플래그는 동작 중 초기 처리를 한 번만 수행하는 데 사용합니다. 태스크가 최초로 실행될 때 ON 되고, 그 태스크의 실행이 완료되면 OFF 됩니다.



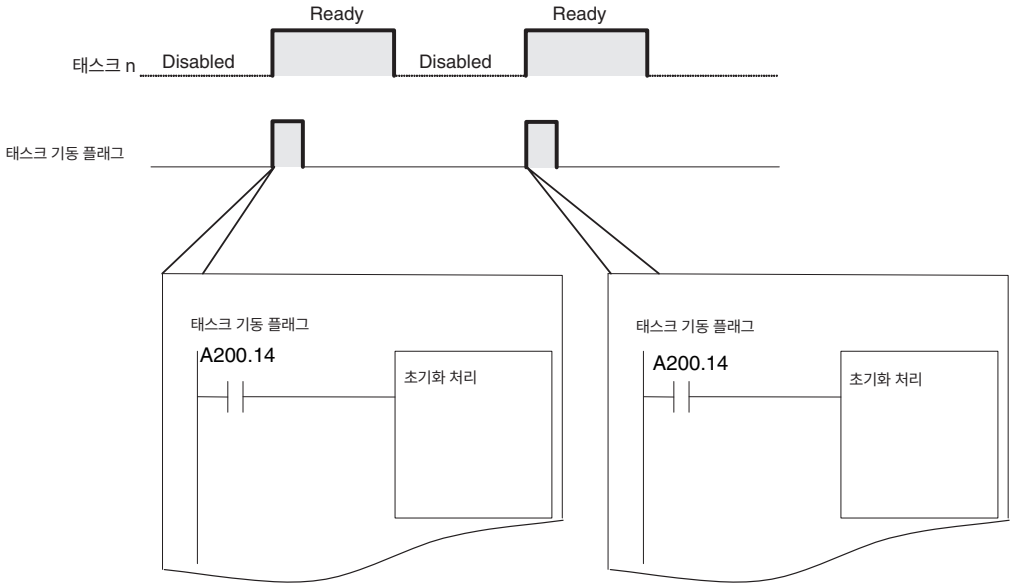
초기 태스크 실행 플래그는 태스크가 최초로 실행될 때 ON 됩니다. 이 플래그를 초기화 처리의 기동 조건으로 사용하십시오.

-For Reference Only-

주 STANDBY 상태의 사이클릭 태스크를 TKON(820) 명령어로 다시 READY 상태로 바꾸더라도, 이는 최초 실행으로 간주되지 않으며 초기 태스크 실행 플래그(20015)는 다시 ON 되지 않습니다. 태스크 기동 플래그(A200.14)를 사용하십시오.

태스크 기동 플래그(A200.14)

태스크 기동 플래그는 사이클릭 태스크의 상태가 STANDBY 상태에서 READY 상태로 바뀔 때마다 ON 됩니다(초기 태스크 실행 플래그(A200.15)는 한 번만 ON 되는 반면, 태스크 기동 플래그는 상태가 STANDBY 상태에서 READY 상태로 바뀔 때마다 ON 됩니다).



태스크 기동 플래그는 태스크가 STANDBY 상태에서 READY 상태로 바뀔 때마다, 즉 STANDBY 상태의 태스크가 TKON(820) 명령어로 유효화될 때마다 초기화 처리를 수행하는 데 사용할 수 있습니다.

● 인터럽트 태스크 관련 플래그와 워드

최대 인터럽트 태스크 처리 시간(A440)*

인터럽트 태스크의 최대 처리 시간이 0.1 ms 단위의 바이너리 데이터로 저장되며, 동작 개시 시에 클리어됩니다.

최대 처리 시간 인터럽트 태스크(A441)*

최대 처리 시간을 가진 인터럽트 태스크 번호가 바이너리 데이터로 저장됩니다. 여기서 8000~ 80FF hex는 태스크 번호 00~FF Hex에 대응합니다.

이 데이터는 동작 개시 시에 클리어됩니다.

중복 리프레시 에러 플래그(비치명 에러)(A402.13)

PLC 설정에서 중복 리프레시 에러 검출이 유효화되어 있으면, 중복 리프레시 에러가 발생했을 때 중복 리프레시 에러 플래그가 ON 됩니다.

* PLC 설정에서 고속 인터럽트 기능이 유효화되어 있으면 워드 A440과 A441의 값은 유효하지 않습니다.

AI 번역본
-For Reference Only-

중복 리프레시 에러 요인 플래그(A426.15)/중복 리프레시 에러를 발생시킨 태스크 번호(A426.00~426.11)

A402.13	중복 리프레시 에러 내용	A426.15	A426.00~426.11
인터럽트 태스크 에러 (PLC 설정에서 인터럽트 태스크 에러 검출이 유효화된 경우)	PLC 설정에서 중복 리프레시 에러 검출이 유효화되어 있는 경우, 동일한 특수 I/O 유닛에 대해 다음 조건이 발생하면 중복 리프레시 에러 플래그가 ON 됩니다. ・인터럽트 태스크에서 실행된 IORF, FIORF, IORD, IOWR 명령어와 사이클릭 태스크에서 실행된 IORF, FIORF, IORD, IOWR 명령어가 경합합니다. ・인터럽트 태스크에서 실행된 IORF, FIORF, IORD, IOWR 명령어와 CPU 유닛의 I/O 리프레시(END 리프레시)가 경합합니다. PLC 설정에서 특수 I/O 유닛의 사이클릭 리프레시가 유효화되어 있는 상태에서 동일한 특수 I/O 유닛에 대해 IORF, FIORF, IORD, IOWR 명령어가 실행되면 중복 리프레시가 발생하여 인터럽트 태스크 에러가 됩니다.	1 (ON)	리프레시 중인 특수 I/O 유닛의 유닛 번호가 12비트 바이너리 데이터로 저장됩니다(유닛 No. 0~95: 000~05F Hex).

프로그램 정지 시 태스크 번호(A294)

프로그램 에러로 태스크의 실행이 정지되었을 때의 태스크 종류와 현재 태스크 번호가 다음과 같이 저장됩니다:

종류	A294
인터럽트 태스크	8000~80FF Hex(인터럽트 태스크 번호 0~255에 대응)
사이클릭 태스크	0000~007F Hex(태스크 번호 0~127에 대응)

● 모든 태스크 관련 플래그

태스크 에러 플래그(A295.12)

다음 태스크 에러 중 하나가 발생하면 태스크 에러 플래그가 ON 됩니다.

・한 사이클 동안 READY 상태인 사이클릭 태스크가 하나도 없습니다.

・사이클릭 태스크(엑스트라 사이클릭 태스크 포함)에 할당된 프로그램이 존재하지 않습니다. (이 상황은 CX-Programmer를 사용할 때에는 발생하지 않습니다.)

・기동된 인터럽트 태스크(엑스트라 사이클릭 태스크 포함)에 프로그램이 할당되어 있지 않습니다.

프로그램 정지 시 태스크 번호(A294)

프로그램 에러로 태스크의 실행이 정지되었을 때의 태스크 종류와 현재 태스크 번호가 다음과 같이 저장됩니다:

종류	A294
사이클릭 태스크	0000~007F Hex(태스크 번호 0~127에 대응)
인터럽트 태스크(추가 사이클릭 태스크 포함)	8000~80FF Hex(인터럽트 태스크 번호 0~255에 대응)

이 정보를 통해 치명 에러가 발생한 위치를 쉽게 파악할 수 있으며, 치명 에러가 해제되면 클리어됩니다. 태스크 동작이 정지된 프로그램 어드레스는 A298(프로그램 어드레스의 하위 비트)과 A299(프로그램 어드레스의 상위 비트)에 저장됩니다.

AI 한국본
For Reference Only

주의사항

● 명령어 사용 제한

동일 태스크 안에 있어야 하는 명령어

다음 명령어는 동일 태스크 안에 배치해야 합니다.

니모닉	명령어
JMP/JME	JUMP/JUMP END
CJP/JME	CONDITIONAL JUMP/JUMP END
CJPN/JME	CONDITIONAL JUMP NOT/CONDITIONAL JUMP END
JMP0/JME0	MULTIPLE JUMP/JUMP END
FOR/NEXT	FOR/NEXT
IL/ILC	INTERLOCK/INTERLOCK CLEAR
SBS/SBN/RET	SUBROUTINE CALL/SUBROUTINE ENTRY/SUBROUTINE RETURN (글로벌 서브루틴 명령어(GSBS(750), GSBN(751), GRET(752))를 사용하면 다른 태스크에서도 서브루틴을 호출할 수 있습니다.)
MCRO/SBN/RET	MACRO/SUBROUTINE ENTRY/SUBROUTINE RETURN
BPRG/BEND	BLOCK PROGRAM BEGIN/BLOCK PROGRAM END
STEP/STEP	STEP DEFINE

인터럽트 태스크에서 사용할 수 없는 명령어

다음 명령어는 인터럽트 태스크에 배치할 수 없습니다. 인터럽트 태스크에서 이들 명령어를 실행하려고 하면 에러 플래그(P_ER)가 ON 되고 그 명령어는 실행되지 않습니다. 인터럽트 태스크를 추가 사이클릭 태스크로 사용하는 경우에는 다음 명령어를 사용할 수 있습니다.

니모닉	명령어
TKON	TASK ON
TKOF	TASK OFF
STEP	STEP DEFINE
SNXT	STEP NEXT
STUP	CHANGE SERIAL PORT SETUP
DI	DISABLE INTERRUPT
EI	ENABLE INTERRUPT
SFCON, SFCOFF, SFCPR, SFCPRN	SFC Task Control Instructions

- 다음 명령어는 인터럽트 태스크에서 동작이 불확정합니다: HUNDRED-MS TIMER: TIM and TIMX(550), TEN-MS TIMER: TIMH(015) and TIMHX(551), ONE-MS TIMER: TMHH(540) and TMHHX(552), TENTH-MS TIMER: TIMU(541) and TIMUX(556), HUNDREDTH-MS TIMER TIMUH(544) and TIMUHX(557), ACCUMULATIVE TIMER: TTIM(087) and TTIMX(555), MULTIPLE OUTPUT TIMER: MTIM(543) and MTIMX(554), LONG TIMER: TIML(542) and TIMLX(553), TIMER WAIT: TIMW(813) and TIMWX(816), HIGH-SPEED TIMER WAIT: TMHW(815) and TMHWX(817), PID CONTROL: PID(190), and FAILURE POINT DETECTION: FPD(269).
- 위에 더해, 다음 명령어는 전원 OFF 인터럽트 태스크에서 사용할 수 없습니다.
  이들 명령어를 전원 OFF 인터럽트 태스크에서 사용해도 에러 플래그(P_ER)는 ON 되지 않습니다. 해당 명령어는 실행되지 않습니다.
  READ DATA FILE: FREAD(700), WRITE DATA FILE: FWRIT(701), WRITE TEXT FILE(TWRIT(704)), NETWORK SEND: SEND/SEND2(090), NETWORK RECEIVE: RECV/RECV2(098), DELIVER COMMAND: CMND/CMND2(490), TRANSMIT: TXD(236), RECEIVE: RXD(235), PROTOCOL MACRO: PMCR(260), EXPLICIT MESSAGE SEND:

EXPLT(720), EXPLICIT GET ATTRIBUTE: EGATR(721), EXPLICIT SET ATTRIBUTE: ESATR(722), EXPLICIT WORD READ: ECHRD(723), EXPLICIT WORD WRITE: ECHWR(724), TRANSMIT VIA SERIAL COMMUNICATIONS UNIT: TXDU/DTXDU(256), RECEIVE VIA SERIAL COMMUNICATIONS UNIT: RXDU/DRXDU(255), INTELLIGENT I/O READ: IORD(222), and INTELLIGENT I/O WRITE: IOWR(223).

・CJ2H CPU 유닛의 PLC 설정에서 고속 인터럽트가 유효화되어 있는 경우, 다음 명령어는 인터럽트 태스크에서 사용할 수 없습니다. 실행하면 에러가 발생합니다.

SELECT EM BANK (EMBC(281)), NETWORK SEND (SEND/SEND2(090)), NETWORK RECEIVE (RECV/RECV2(098)), DELIVER COMMAND (CMND/CMND2(490)), PROTOCOL MACRO (PMCR(260)), TRANSMIT VIA SERIAL COMMUNICATIONS UNIT (TXDU(256)), RECEIVE VIA SERIAL COMMUNICATIONS UNIT (RXDU(255)), EXPLICIT MESSAGE SEND (EXPLT(720)), EXPLICIT GET ATTRIBUTE (EGATR(721)), EXPLICIT SET ATTRIBUTE (ESATR(722)), EXPLICIT WORD READ (ECHRD(723), and EXPLICIT WORD WRITE (ECHWR(724))

PLC 설정에서 백그라운드 처리가 지정되어 있으면 테이블 데이터 처리 명령어, 문자열 처리 명령어 또는 데이터 시프트 명령어에서도 명령어 처리 에러가 발생합니다.

참고

태스크와 타이머 동작의 관계

타이머 명령어는 다음과 같이 동작합니다:

・타이머 번호 0~2047
타이머를 포함하는 태스크가 STANDBY 상태로 바뀌거나 READY 상태로 복귀하더라도 타이머 현재값은 계속 갱신됩니다. 동작 중인 TIM 명령어를 포함하는 태스크가 STANDBY 상태가 되고, 그 태스크가 READY 상태로 복귀했을 때 현재값이 0이면 TIM 명령어가 실행되는 즉시 완료 플래그가 ON 됩니다. (타이머의 완료 플래그는 명령어가 실행될 때만 갱신됩니다.) 현재값은 계속 갱신됩니다.

・타이머 번호 2048~4095
태스크에서 기동된 타이머의 현재값은 그 태스크가 STANDBY 상태인 동안에는 갱신되지 않습니다. 다만 PLC 프로퍼티에 타이머 2048~4095를 타이머 0~2047과 동일하게 동작시킬 수 있는 설정이 있습니다.

태스크와 조건 플래그의 관계

모든 조건 플래그는 각 태스크 실행 전에 클리어됩니다. 따라서 태스크 1이 끝난 시점의 조건 플래그 상태를 태스크 2에서 읽을 수는 없습니다. 프로그램의 다른 부분(예: 다른 태스크)에서 조건 플래그 상태를 읽으려면 CCS(282)와 CCL(283)을 사용할 수 있습니다.

● 인터럽트 태스크에 관한 주의사항

인터럽트 태스크의 우선순위

전원 OFF 인터럽트 태스크를 실행하기 위해 다른 인터럽트 태스크의 실행은 정지됩니다.
CPU 유닛은 리셋되지만, 정지된 인터럽트 태스크는 전원 OFF 인터럽트 태스크의 실행 후에도 실행되지 않습니다.

그 밖의 인터럽트 태스크의 우선순위는 다음 절에서 설명합니다.

복수의 인터럽트가 동시에 발생한 경우

복수의 인터럽트가 동시에 발생하면 전원 OFF 인터럽트 태스크 이외의 인터럽트 태스크는 다음 우선순위에 따라 실행됩니다.

I/O 인터럽트 태스크 > 외부 인터럽트 태스크 > 입력 인터럽트 태스크(펄스 I/O 모듈)* > 고속 카운터 목표값 일치* > 정주기 인터럽트 태스크

AI 번역본
-For Reference Only-

각종 인터럽트 태스크(펄스 I/O 모듈)가 둘 이상 발생하면 번호가 작은 것부터 순서대로 실행됩니다. 복수의 고속 카운터에서 목표값 일치가 발생한 경우,* 인터럽트 태스크는 고속 카운터 번호의 오름차순으로 실행됩니다.

* CJ2M/CJ3M CPU 유닛과 함께 펄스 I/O 모듈을 사용하는 경우에만 지원됩니다.

인터럽트 태스크 실행 중의 인터럽트

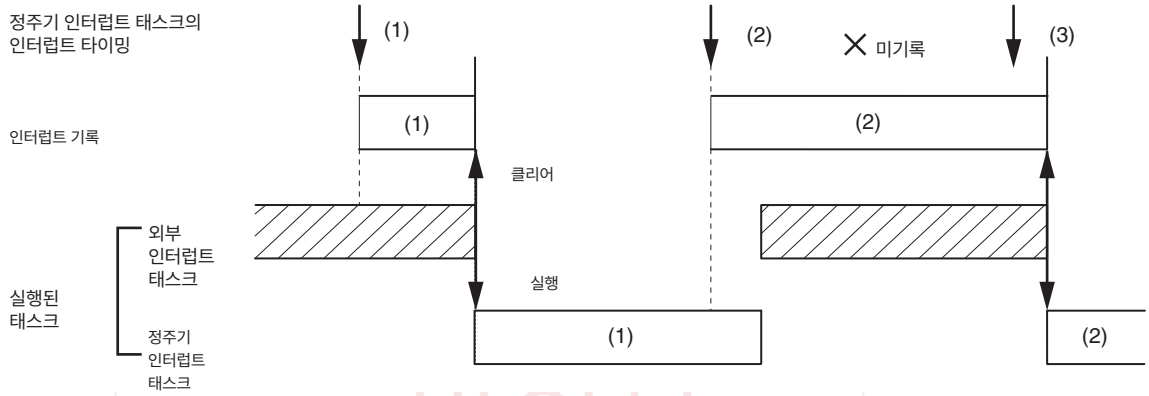
다른 인터럽트 태스크가 실행 중일 때 인터럽트가 발생하면, 그 인터럽트의 태스크는 원래의 인터럽트 실행이 끝날 때까지 실행되지 않습니다.

참고

다른 인터럽트 태스크가 실행 중일 때 발생한 특정 I/O 인터럽트 태스크 번호를 CPU 유닛이 저장했다가 실행하지 않게 하려면, 다른 인터럽트 태스크에서 CLI(CLEAR INTERRUPT) 명령어를 실행하여 내부에 저장된 인터럽트 번호를 클리어하십시오. 정주기 인터럽트와 외부 인터럽트는 취소할 수 없습니다.

참고

각 인터럽트 태스크에 대해 메모리에 기록되는 인터럽트는 하나뿐입니다. 그 때문에 이미 기록된 것과 같은 인터럽트가 다시 발생하면, 새로 발생한 인터럽트는 무시되어 기록되지 않습니다. 기록된 인터럽트는 인터럽트 태스크의 선두가 실행될 때 클리어됩니다. 정주기 인터럽트는 우선순위가 낮고 한 번에 하나의 인터럽트만 기록되므로, 정주기 인터럽트가 건너뛰어질 수 있습니다.



- **사용 시 주의사항**

특수 I/O 유닛에 대한 IORF(097), FIORF(225), IORD(222), IOWR(223)의 실행

특수 I/O 유닛을 사용하면서 인터럽트 태스크에서 IORF(097), FIORF(225), IORD(222), IOWR(223)을 실행하는 경우에는, PLC 설정에서 그 특수 I/O 유닛의 사이클릭 리프레시를 반드시 무효화하십시오.

특수 I/O 유닛이 사이클릭 리프레시 또는 I/O 리프레시 명령어로 리프레시되고 있을 때, 인터럽트 태스크에서 IORF(097)이나 FIORF(225) 명령어로 같은 특수 I/O 유닛을 리프레시하려 하거나 IORD(222)나 IOWR(223) 명령어로 같은 특수 I/O 유닛의 데이터를 읽고 쓰려 하면 중복 리프레시 에러가 발생합니다. 이 경우 IORF(097), FIORF(225), IORD(222), IOWR(223) 명령어는 실행되지 않지만 에러 플래그(P_ER)는 ON 되지 않습니다. 사이클릭 리프레시는 정상적으로 수행됩니다.

중복 리프레시 에러가 발생했을 때 PLC 설정에서 중복 리프레시 에러 검출이 유효화되어 있으면 A402.13(중복 리프레시 에러 플래그)이 ON 되고, 특수 I/O 유닛의 유닛 번호가 A426(중복 리프레시 에러 태스크 번호)에 저장됩니다.

AI 번역본

● 사이클릭 태스크의 특정 처리 중 인터럽트 금지

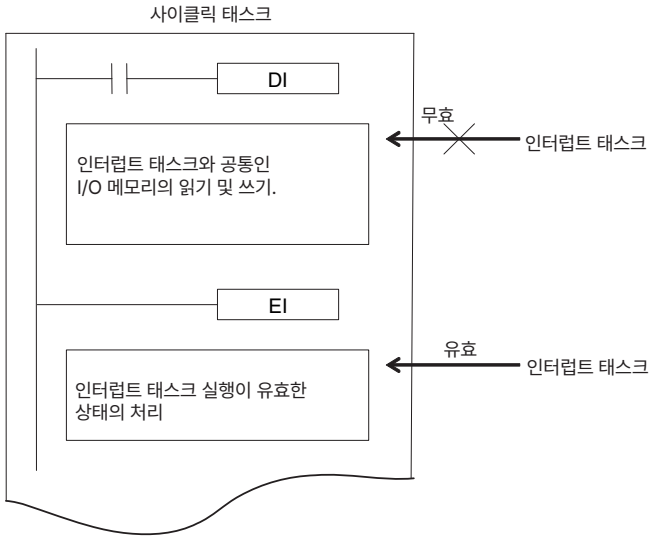
다음의 경우에 인터럽트가 발생하면 처리가 중단되고 인터럽트 태스크가 실행됩니다.

- 명령어가 실행되고 있는 동안
- 기본 I/O 유닛 또는 특수 I/O 유닛의 리프레시 중
- 주변 서비스 중

사이클릭 태스크와 인터럽트 태스크 간의 데이터 동시성 확보

사이클릭 태스크와 인터럽트 태스크가 같은 I/O 메모리 어드레스를 읽고 쓰는 경우 데이터의 동시성이 확보되지 않을 수 있습니다. 사이클릭 태스크 명령어가 메모리에 액세스하는 동안 인터럽트를 무효화하려면 다음 순서를 사용하십시오.

- 사이클릭 태스크 명령어로 읽고 쓰기 직전에 DI(693)(DISABLE INTERRUPT) 명령어를 사용하여 인터럽트 태스크의 실행을 무효화합니다.
- 처리 직후에 EI(694)(ENABLE INTERRUPT) 명령어를 사용하여 인터럽트 태스크의 실행을 유효화합니다.

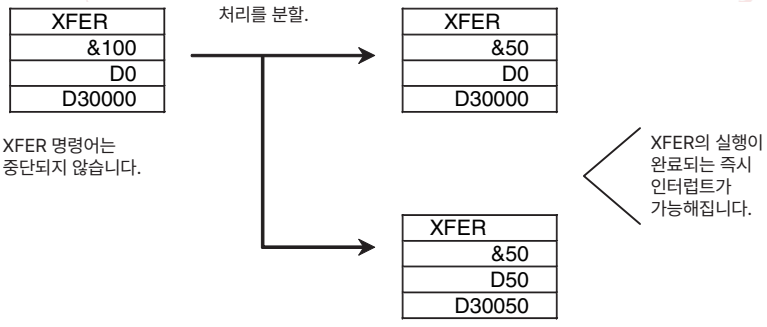


응답의 수신과 처리가 필요한 명령어(네트워크 명령어나 시리얼 통신 명령어 등)의 실행 중에는 DI(693)과 EI(694)로 인터럽트 태스크를 무효화하더라도 데이터 동시성에 문제가 발생할 수 있습니다.

참고

BIT COUNTER(BCNT), BLOCK SET(BSET), BLOCK TRANSFER(XFER) 명령어의 실행은 인터럽트 태스크 실행을 위해 중단되지 않습니다. 즉, 인터럽트 태스크가 실행되기 전에 해당 명령어의 실행이 완료되므로 인터럽트의 응답이 지연됩니다. 이를 방지하려면 아래 XFER의 예와 같이 이들 명령어의 데이터 처리를 여러 개의 명령어로 나누십시오.

-For Reference Only-



5-3 섹션

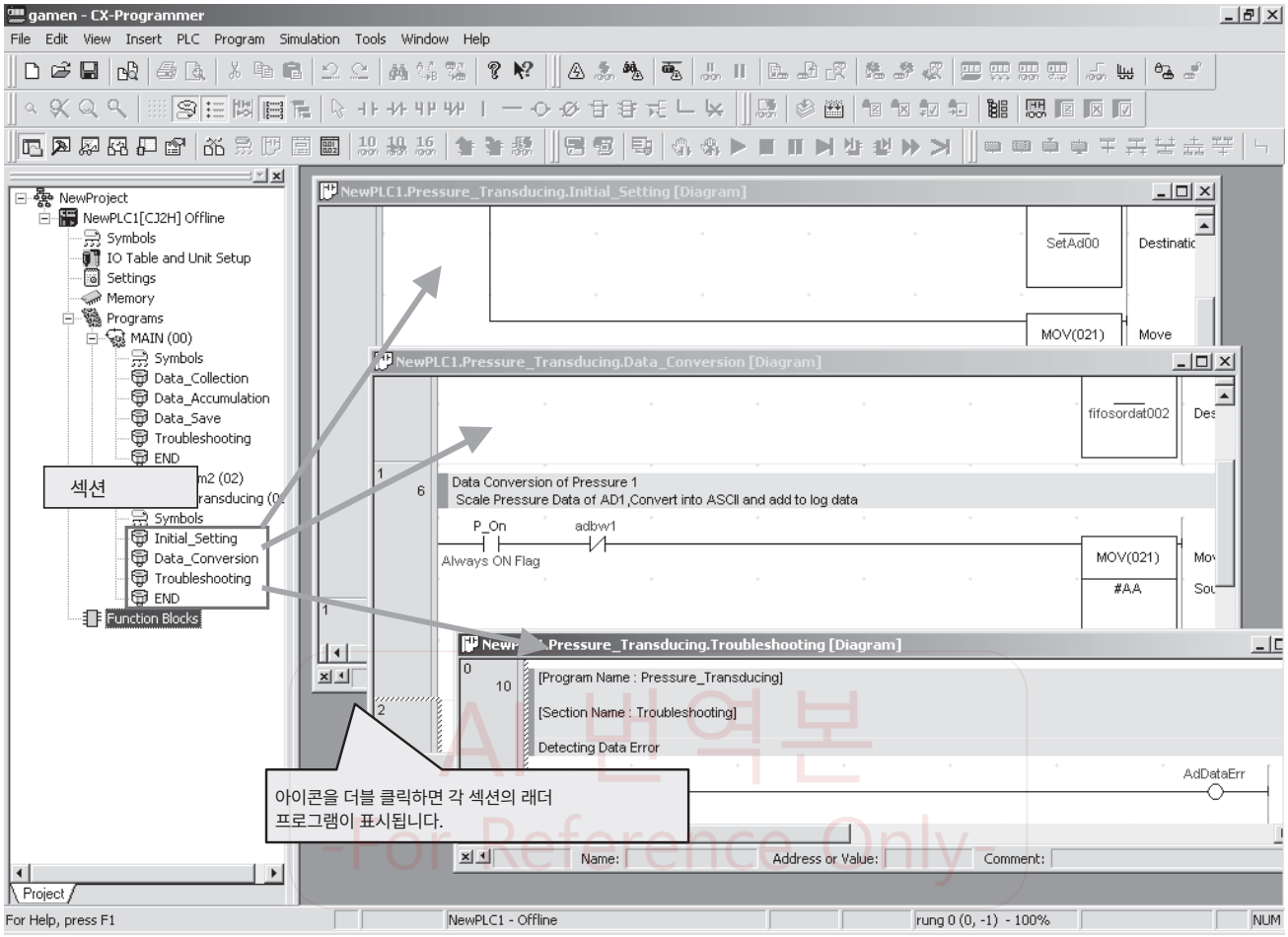
5-3-1 섹션 개요

CX-Programmer에서는 프로그래밍을 기능 단위로 나누기 위해 프로그램을 섹션 단위로 작성하고 표시할 수 있습니다. 태스크 안의 어떤 프로그램이든 섹션으로 분할할 수 있습니다. 섹션을 사용하면 프로그램의 가독성이 좋아지고 편집이 간단해집니다.

・처리가 명확해집니다
섹션에는 그 섹션에서 수행하는 공정이나 제어를 나타내는 임의의 이름을 붙일 수 있습니다

・편집이 쉬워집니다
마우스로 드래그&드롭하여 섹션의 순서를 바꿀 수 있고, 섹션을 복사하여 붙여 넣을 수도 있습니다.

・제한 없이 사용할 수 있습니다
프로그램당 작성할 수 있는 섹션 수에는 제한이 없습니다.



AI 번역본
For Reference Only

● 섹션 수

사용할 수 있는 섹션 수에는 제한이 없습니다. 또한 기존과 마찬가지로 섹션 하나만으로 프로그램을 작성할 수도 있습니다.

● 섹션의 실행 순서

섹션은 프로젝트 트리에 나타나는 순서(즉, 위에서 아래로)대로 실행됩니다. 프로그램 어드레스는 태스크가 실행되는 순서대로 자동으로 연속됩니다. 렁 번호는 각 섹션에서 0부터 시작합니다.

● 섹션 이름

섹션 이름은 "Section□"이 되며, 접미사로 1부터 시작하는 일련번호가 자동으로 할당됩니다. 이름은 사용자 정의로 변경할 수 있습니다.

● 프로그램을 섹션으로 분할하는 이점

프로그래밍을 섹션으로 분할하면 다음과 같은 이점이 있습니다.

- 프로그래밍은 CPU 유닛에서 섹션 단위로 업로드할 수 있습니다. 프로그래밍의 한 섹션을 미리 업로드해 두면 온라인 에디팅을 시작하는 데 걸리는 시간이 단축됩니다. 온라인 에디팅에서는 한 번에 프로그램의 한 섹션만 다운로드할 수 있습니다. 또한 섹션을 하나씩 다운로드하는 것도 불가능합니다.
- 섹션을 이동하여 프로그램 전체의 실행 순서를 바꿀 수 있고, 다른 프로그램으로 이동하거나 프로젝트 트리에서 복사할 수 있습니다.

이를 통해 프로그램 전체보다 작은 프로그래밍 모듈을 표준화할 수 있습니다.

● 섹션 관련 파일 메모리 파일

섹션은 CX-Programmer 프로젝트 파일(.CXP)에 포함됩니다. 하나의 CPU 유닛의 섹션 이름, 섹션 코멘트, 프로그램 코멘트는 프로그램 인덱스 파일(PROGRAMS.IDX)에 저장됩니다.



참고

태스크는 태스크 프로그램의 실행 여부를 제어하는 데 사용됩니다. 한편 섹션은 사용자 정의 프로그램 안에서 더 낮은 수준으로 만든 구분입니다. 섹션은 주로 프로그래밍을 이해하기 쉽게 하기 위해 만들며, 적용 예는 다음과 같습니다.

- 태스크가 실행될 때마다 같은 처리가 필요한 경우, 하나의 태스크 안에서 초기 처리와 메인 처리를 다른 섹션으로 나눌 수 있습니다.
- 통상 프로그램 영역과 서브루틴을 섹션으로 나눌 수 있습니다.

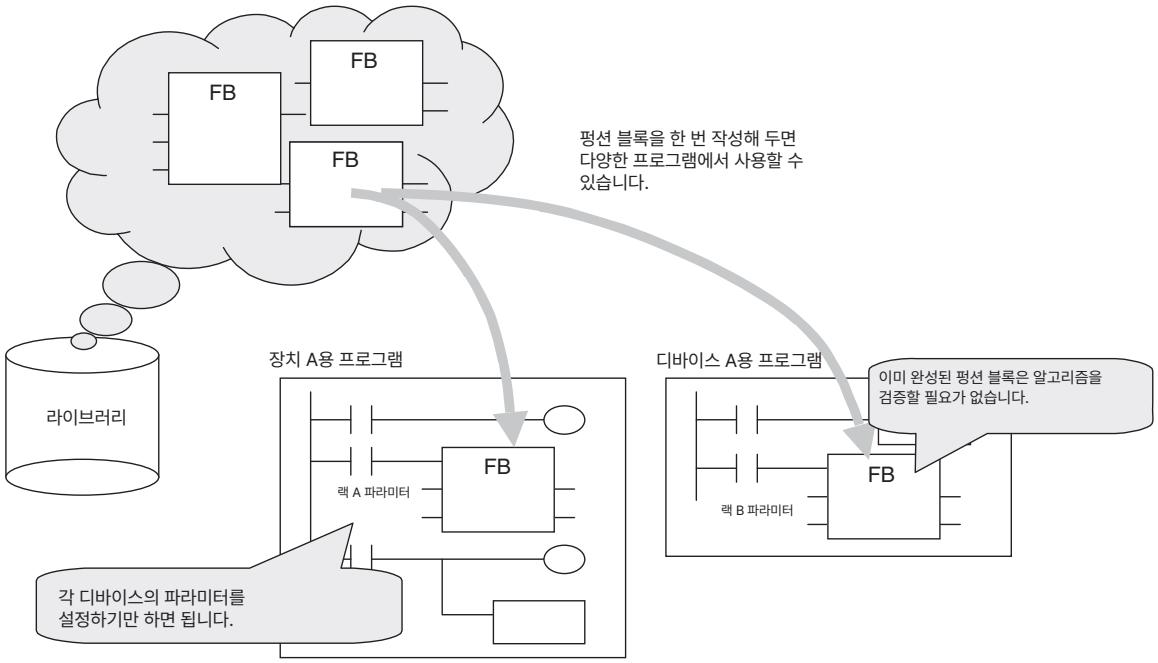
AI 번역본
-For Reference Only-

5-4 펑션 블록

5-4-1 펑션 블록

펑션 블록(FB)은 미리 정의된 표준 처리 기능의 프로그래밍을 담은 기본 프로그램 요소입니다. 펑션 블록을 작성하여 라이브러리에 저장해 두면, 시스템에 필요한 펑션 블록을 선택하여 프로그램에 배치하고 I/O 파라미터를 설정하는 것만으로 프로그래밍에 사용할 수 있습니다. 우수한 펑션 블록은 프로그램 개발의 효율을 크게 향상시킬 수 있습니다.

오므론의 펑션 블록은 IEC 61131-3에 준거합니다.



● 펑션 블록의 특징

・재사용성
펑션 블록은 라이브러리에 저장해 두면 쉽게 재사용할 수 있습니다. 프로그래머가 펑션 블록을 이해하고 나면 파라미터를 설정하는 것만으로 사용할 수 있습니다. 이로써 프로그래밍 효율이 크게 향상됩니다.

・디버깅 작업 감소
이미 디버깅이 끝난 고품질 펑션 블록을 재사용하면 다시 디버깅할 필요가 없어져 디버깅 작업이 줄어듭니다.

・데이터 보호
펑션 블록 내부의 변수는 외부에서 직접 액세스할 수 없으므로 데이터가 보호됩니다. (데이터가 의도치 않게 변경되지 않습니다.)

AI 번역본
-For Reference Only-

5-4-2 펑션 블록의 특징

● 펑션 블록의 어드레스 입력

펑션 블록의 어드레스는 일반적으로 실제 물리 어드레스가 아니라 변수를 사용하여 입력합니다. 프로그래머는 이러한 변수를 사용하여 파라미터(즉, 어드레스 또는 값)를 설정합니다.
각 변수에 사용되는 어드레스는 펑션 블록이 프로그램에 배치될 때마다 CX-Programmer가 자동으로 할당합니다.

● 네스팅

펑션 블록은 다른 펑션 블록 안에 네스팅할 수 있습니다. 첫 번째 펑션 블록의 호출을 포함하여 최대 8단계까지 네스팅할 수 있습니다.

● Smart FB Library

Smart FB Library는 오뭔론 PLC 유닛과 FA 부품 간의 동작을 향상시키는 펑션 블록 모음입니다. 기본적인 유닛 기능과 FA 부품 기능을 사용하는 데 래더 프로그램을 작성할 필요는 없습니다.

● FB 정의의 온라인 에디팅

FB 정의는 운전 중에 변경할 수 있으므로 디버깅 중에도 FB 정의를 신속하게 편집할 수 있습니다. 또한 24시간/일 가동해야 하는 설비에서도 FB를 안심하고 사용할 수 있습니다.

● FB 정의의 보호

프로젝트 파일에 할당된 펑션 블록 정의에 패스워드를 설정하고 그 용도에 따라 정의를 보호 함으로써 프로그램의 부정한 조작, 편집 또는 도용을 방지할 수 있습니다.

● FB 생성 기능

기존 PLC 프로그램을 간단히 FB로 변환하여 재사용할 수 있습니다.

AI 번역본
-For Reference Only-

5-4-3 펑션 블록 사양

항목		사양	
펑션 블록의 정의 및 작성	펑션 블록 정의 개수	CJ2H-CPU6□(-EIP): 최대 2048개 CJ3M-CPU□□: 최대 2048개 CJ2M-CPU□4/□5: 최대 2048개 CJ2M-CPU□1/□2/□3: 최대 256개	
	펑션 블록 이름	최대 64문자	
	변수	변수 이름	최대 15,000문자
		변수 종류	입력 변수(Inputs), 출력 변수(Outputs), 입출력 변수(In Out), 내부 변수(Internals), 외부 변수(Externals)
		펑션 블록에서 사용하는 변수 개수(내부 변수, 외부 변수, EN, EN0은 포함하지 않음)	펑션 블록 정의당 최대 변수 개수 ・입출력 변수: 최대 16개 ・입력 변수 + 입출력 변수: 최대 64개 ・출력 변수 + 입출력 변수: 최대 64개
		변수가 사용하는 어드레스의 할당	자동 할당(할당 범위는 사용자가 설정할 수 있습니다.)
		실제 어드레스 지정	지원
		배열 지정	지원(1차원 배열만, 그리고 내부 변수와 입출력 변수에 대해서만 지원)
	언어	펑션 블록은 래더 프로그래밍 언어 또는 구조화 텍스트로 작성할 수 있습니다.	
인스턴스 작성	인스턴스 개수	CJ2H-CPU6□(-EIP): 최대 2048개 CJ3M-CPU□□: 최대 2048개 CJ2M-CPU□4/□5: 최대 2048개 CJ2M-CPU□1/□2/□3: 최대 256개	
	인스턴스 이름	최대 15,000문자	
FB 프로그램 영역		CJ3M-CPU□□: 40K 스텝 CJ2M-CPU□□: 20K 스텝 CJ2M/CJ3M CPU 유닛에서만 지원합니다. 이 영역은 펑션 블록 정의를 저장하는 데 사용됩니다.	
펑션 블록을 파일로 저장	프로젝트 파일	프로젝트 파일(.cxp/cxt)에는 펑션 블록 정의와 인스턴스가 포함됩니다.	
	프로그램/네트워크 심볼 파일	파일 메모리의 프로그램 파일(*.obj)에는 펑션 블록 정의와 인스턴스가 포함됩니다.	
	펑션 블록 라이브러리 파일	각 펑션 블록 정의는 다른 프로젝트에서 재사용할 수 있도록 하나의 파일(.cxf)로 저장할 수 있습니다.	

펑션 블록이 사용하는 스텝 수

펑션 블록을 사용할 때는 다음 두 항목에 메모리가 필요합니다:

1. 펑션 블록 정의
2. 펑션 블록을 펑션 블록 인스턴스로서 사용자 프로그램에 붙여 넣을 때의 인스턴스 생성 처리

AI 번역 -For Reference Only-

따라서 메모리에서 사용되는 스텝 수는 프로그램에 작성된 펑션 블록의 인스턴스 수에 따라 증가합니다(항목 2).

펑션 블록에 사용되는 메모리 영역

펑션 블록에 사용되는 메모리 영역은 다음 표에 설명된 것처럼 사용하는 CJ2/CJ3 CPU 유닛의 기종에 따라 다릅니다.

CJ2M/CJ3M CPU 유닛에는 펑션 블록 정의를 저장하기 위한 FB 프로그램 영역이라는 전용 영역이 있습니다. CJ2H CPU 유닛에는 이 영역이 없습니다.

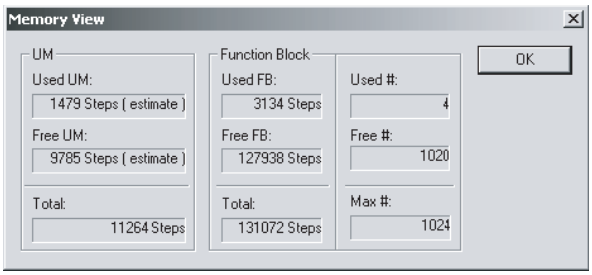
CPU 유닛	기종	펑션 블록에 사용되는 메모리 영역
CJ2H	CJ2H-CPU6□-EIP CJ2H-CPU6□	사용자 프로그램 영역
CJ2M/CJ3M	CJ2M-CPU□□ CJ3M-CPU□□	펑션 블록의 인스턴스를 작성하면(즉, 펑션 블록 정의를 호출하고 파라미터를 전달하면) 사용자 프로그램 메모리를 사용합니다. 펑션 블록 정의는 펑션 블록 전용 영역인 FB 프로그램 영역의 메모리를 사용합니다. FB 프로그램 영역의 용량을 초과하면 사용자 프로그램 영역이 사용됩니다.

펑션 블록 메모리 사용량 확인

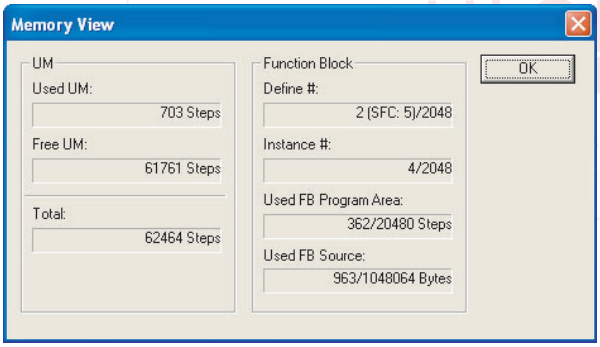
CX-Programmer에서 View – Memory View를 선택하십시오.

다음 Memory View 대화 상자가 표시됩니다.

● PLC 기종(“Device Type”)이 CJ2H인 경우



● PLC 기종(“Device Type”)이 CJ2M/CJ3M인 경우



참고

메모리 뷰 대화 상자에 대한 자세한 내용은 CX-Programmer 오퍼레이션 매뉴얼(Cat. No. W446)을 참조하십시오.

펑션 블록이 사용하는 스텝 수 계산

● CJ2H CPU 유닛

펑션 블록이 사용하는 스텝 수는 다음 계산식을 기준으로 삼으십시오.

사용 스텝 수 = 인스턴스 수 × (호출부 사이즈 m + I/O 파라미터 전송부 사이즈 n × 파라미터 수) + 펑션 블록 정의의 명령어 스텝 수 p(주 참조.)

주 펑션 블록 정의의 명령어 스텝 수(p)에는 메모리가 사용되지 않습니다. 이는 두 번째 이후의 인스턴스에서 동일한 펑션 블록 정의가 여러 위치에 작성된 경우(즉, 여러 인스턴스인 경우)입니다. 따라서 위 계산식에서 펑션 블록 정의의 명령어 스텝 수(p)는 인스턴스 수에 곱하지 않습니다.

항목			스텝 수
m	호출부		52 스텝
n	I/O 파라미터 전송부 데이터형은 괄호 안에 표시되어 있습니다.	1비트(BOOL) 입력 심볼 또는 출력 심볼	6 스텝
		1워드(INT, UINT, WORD) 입력 심볼 또는 출력 심볼	6 스텝
		2워드(DINT, UDINT, DWORD, REAL) 입력 심볼 또는 출력 심볼	6 스텝
		4워드(LINT, ULINT, LWORD, LREAL) 입력 심볼 또는 출력 심볼	18 스텝
		I/O 심볼	6 스텝
p	펑션 블록 정의의 명령어 스텝 수	명령어 스텝 총수(표준 사용자 프로그램과 동일) + 11 스텝.	

예

1워드(INT) 데이터형의 입력 심볼 5개, 1워드(INT) 데이터형의 출력 심볼 5개, 그리고 100스텝의 펑션 블록 정의인 경우:

1 인스턴스의 스텝 수 = 52 + (5 + 5) × 6 스텝 + 100 스텝 + 11 스텝 = 223 스텝

펑션 블록이 표준 텍스트 언어로 작성된 경우에는 실제 스텝 수를 계산할 수 없습니다. 펑션 블록 정의의 명령어 스텝 수는 펑션 블록 정의의 속성에서 확인할 수 있습니다.

참고

펑션 블록의 사양, 표기 방법 및 입력 순서에 대해서는 CX-Programmer 오퍼레이션 매뉴얼: 펑션 블록과 구조화 텍스트(Cat. No. W447)를 참조하십시오.

사용설명서
-For Reference Only-

● CJ2M/CJ3M CPU 유닛

펑션 블록이 사용하는 스텝 수는 다음 계산식을 기준으로 삼으십시오.

사용자 메모리 사용량(사용 스텝 수)
= 인스턴스 수 × (호출부 사이즈 m + I/O 파라미터 전송부 사이즈 n × 파라미터 수)

항목			스텝 수
m	호출부		15 스텝
n	I/O 파라미터 전송부 데이터형은 괄호 안에 표시됩니다.	1비트(BOOL) 입력 심볼 또는 출력 심볼	3 스텝
		1워드(INT, UINT, WORD) 입력 심볼 또는 출력 심볼	3 스텝
		2워드(DINT, UDINT, DWORD, REAL) 입력 심볼 또는 출력 심볼	3 스텝
		4워드(LINT, ULINT, LWORD, LREAL) 입력 심볼 또는 출력 심볼	6 스텝
		I/O 심볼	9 스텝

펑션 블록 프로그램 영역 사용량(사용 스텝 수)
= 인스턴스 수 × (호출부 사이즈 p + I/O 파라미터 전송부 사이즈 q × 파라미터 수) + 펑션 블록 정의의 명령어 스텝 수 r(주 참조.)

항목			스텝 수
p	호출부		37 스텝
q	I/O 파라미터 전송부 데이터형은 괄호 안에 표시됩니다.	1비트(BOOL) 입력 심볼 또는 출력 심볼	3 스텝
		1워드(INT, UINT, WORD) 입력 심볼 또는 출력 심볼	3 스텝
		2워드(DINT, UDINT, DWORD, REAL) 입력 심볼 또는 출력 심볼	3 스텝
		4워드(LINT, ULINT, LWORD, LREAL) 입력 심볼 또는 출력 심볼	6 스텝
		I/O 심볼	9 스텝
r	펑션 블록 정의의 명령어 스텝 수	명령어 스텝 총수(표준 사용자 프로그램과 동일) + 11 스텝.	

주 펑션 블록 정의의 명령어 스텝 수(r)에는 메모리가 사용되지 않습니다. 이는 두 번째 이후의 인스턴스에서 동일한 펑션 블록 정의가 여러 위치에 작성된 경우(즉, 여러 인스턴스인 경우)입니다. 따라서 위 계산식에서 펑션 블록 정의의 명령어 스텝 수(r)는 인스턴스 수에 곱하지 않습니다.

AI 번역본
For Reference Only

예

1워드(INT) 데이터형의 입력 심볼 5개, 1워드(INT) 데이터형의 출력 심볼 5개, 그리고 100스텝의 펑션 블록 정의인 경우:

1 인스턴스의 사용자 메모리 사용량 = 15 + (5 + 5) × 3 스텝 = 45 스텝. 펑션 블록 프로그램 영역의 메모리 사용량 = 37 + (5 + 5) × 3 스텝 + 100 스텝 + 11 스텝 = 178 스텝.

펑션 블록이 표준 텍스트 언어로 작성된 경우에는 실제 스텝 수를 계산할 수 없습니다.

펑션 블록 정의의 명령어 스텝 수는 펑션 블록 정의의 속성에서 확인할 수 있습니다.

5-5 심볼

5-5-1 개요

심볼

● 심볼 테이블

문자열을 심볼로 등록하면 I/O 메모리 영역 어드레스나 상수를 문자열로 지정할 수 있습니다. 심볼은 CX-Programmer의 심볼 테이블에 등록하십시오.
심볼을 사용하면 실제 어드레스를 의식하지 않고 이름으로 프로그래밍할 수 있습니다.

심볼 테이블은 사용자 프로그램이나 I/O 테이블 등의 다른 파라미터와 함께 CX-Programmer 프로젝트 파일(.CXP)에 저장됩니다.

● 심볼 사용 조건

심볼 사용이 필수인지 임의인지는 다음 표와 같이 프로그래밍 언어와, 심볼을 펑션 블록 내부에서 사용하는지 외부에서 사용하는지에 따라 달라집니다.

프로그램 요소	프로그래밍 언어	심볼 지정
펑션 블록 외부의 프로그래밍	래더 다이어그램	임의
	ST	필수
	SFC	임의
펑션 블록 내부의 프로그래밍(펑션 블록 내부에서는 심볼을 "변수"라고 부릅니다.)	래더 다이어그램	필수
	ST	

AI 번역본
-For Reference Only-

5-5-2 심볼의 종류

다음과 같은 종류의 심볼을 사용할 수 있습니다.

프로그램 심볼

● 심볼의 종류

・글로벌 심볼
글로벌 심볼은 CPU 유닛의 모든 태스크에서 액세스할 수 있습니다.

・로컬 심볼
로컬 심볼은 하나의 태스크에서만 액세스할 수 있습니다.

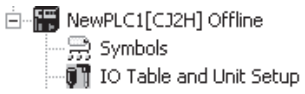
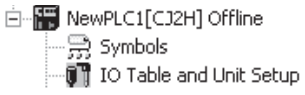
・네트워크 심볼
외부 기기에서 태그를 사용하여 CPU 유닛에 정의된 네트워크 심볼을 통해 로컬 CPU 유닛의 I/O 메모리에 액세스할 수 있습니다. 네트워크 심볼은 로컬 CPU 유닛 내에서도 사용할 수 있습니다.
CJ2H-CPU□□(-EIP 없음) 버전 1.5 이전 및 CJ2M-CPU1□ 버전 2.1 이전에서는 지원하지 않습니다.

● 어드레스 할당

프로그래밍에 사용하는 심볼에는 다음 요인에 따라 어드레스가 할당됩니다.

・사용자 지정

・CX-Programmer를 사용한 자동 할당
자동 할당에 사용되는 메모리 영역은 CX-Programmer의 PLC 메뉴에서 Memory Allocation – Automatic Address Allocation을 선택하여 설정합니다.

심볼의 종류	이름	CX-Programmer 프로젝트 트리	적용 범위			어드레스 및 I/O 코멘트 (심볼 이름 없음)
			네트워크 심볼을 사용한 액세스	다른 태스크에서의 액세스	로컬 태스크에서의 액세스	
프로그램 심볼	글로벌 심볼	PLC 트리 	지원하지 않음	지원	지원	지원
	로컬 심볼	프로그램 트리 		지원하지 않음	지원	지원하지 않음
	네트워크 심볼	PLC 트리의 글로벌 변수 테이블 	지원	지원	지원	지원하지 않음

주 "글로벌"과 "로컬"은 심볼의 적용 범위만을 나타냅니다. 메모리 어드레스의 적용 범위와는 관계가 없습니다. 따라서 다음과 같은 경우에는 에러가 아니라 경고가 발생하며, 사용자 프로그램을 전송할 수 있습니다.

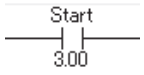
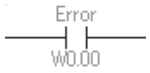
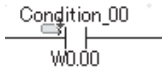
・서로 다른 두 로컬 심볼에 동일한 어드레스가 사용된 경우

・글로벌 심볼과 로컬 심볼에 동일한 어드레스가 사용된 경우

-번역본 For Reference Only-

참고

CX-Programmer 프로그램에서는 글로벌 심볼, 로컬 심볼, 네트워크 심볼을 다음과 같은 문자 색과 심볼을 나타내는 아이콘으로 구별할 수 있습니다.

심볼의 종류	표시 색	예(기본 색)
글로벌 심볼	검정(고정)	
로컬 심볼	파랑(기본값) Tools – Options를 선택하고 Appearance 탭 페이지를 표시한 후 local symbols 항목을 선택하면 색을 변경할 수 있습니다.	
네트워크 심볼	검정(고정) 아이콘( )은 네트워크 심볼을 나타내며 심볼 이름 아래에 표시됩니다.	

펑션 블록 내의 변수

변수는 펑션 블록 내의 알고리즘에서만 액세스할 수 있습니다.

이름	CX-programmer 프로젝트 트리	적용 범위		어드레스 및 I/O 코멘트 (심볼 이름 없음)
		네트워크 심볼을 사용한 액세스	펑션 블록 외부에서의 액세스	
변수(즉, 펑션 블록에서 사용하는 심볼)	펑션 블록 내의 정의 	지원하지 않음 (외부 변수에 네트워크 심볼을 등록할 수 없습니다.)	지원하지 않음	지원하지 않음

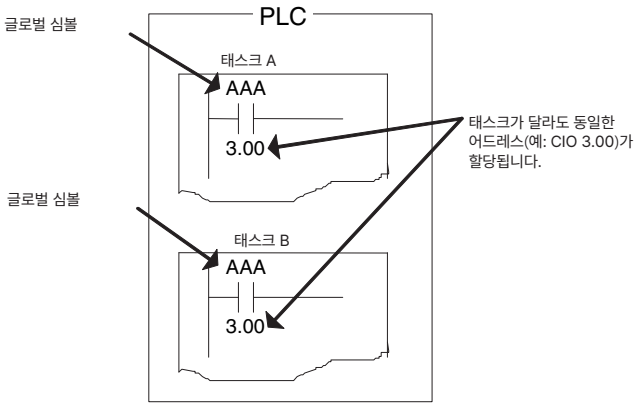
변수의 사양 중 일부는 펑션 블록 외부에서 사용하는 심볼의 사양과 다릅니다. 변수에 사용할 영역은 CX-Programmer의 PLC 메뉴에서 Memory Allocation – Function Block/SFC Memory를 선택하여 설정하십시오.

AI 번역본
-For Reference Only-

5-5-3 글로벌 심볼

글로벌 심볼은 대상 CPU 유닛의 모든 태스크에서 지원되는 심볼입니다. 따라서 예를 들어 "AAA"라는 이름의 심볼은 대상 CPU 유닛의 모든 태스크에서 동일한 어드레스가 됩니다.

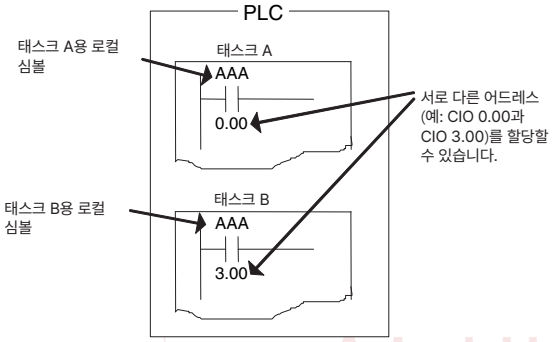
예: "AAA"라는 이름의 심볼을 글로벌 심볼로 설정하면 태스크가 달라도 동일한 어드레스 (예: CIO 3.00)가 할당됩니다.



5-5-4 로컬 심볼

로컬 심볼은 그것이 정의된 태스크에서만 액세스할 수 있습니다. 각 로컬 심볼은 하나의 태스크에서만 지원됩니다. 로컬 심볼은 태스크마다 별도로 지정하므로, 동일한 심볼이라도 태스크가 다르면 다른 심볼로 취급됩니다.

예를 들어 "AAA"라는 이름의 심볼을 로컬 심볼로 설정하면, 동일한 "AAA" 심볼이라도 태스크마다 서로 다른 어드레스(예: CIO 0.00과 CIO 3.00)를 할당할 수 있습니다.



AI 번역본
-For Reference Only-

5-5-5 네트워크 심볼

네트워크 심볼을 지원합니다.

개요

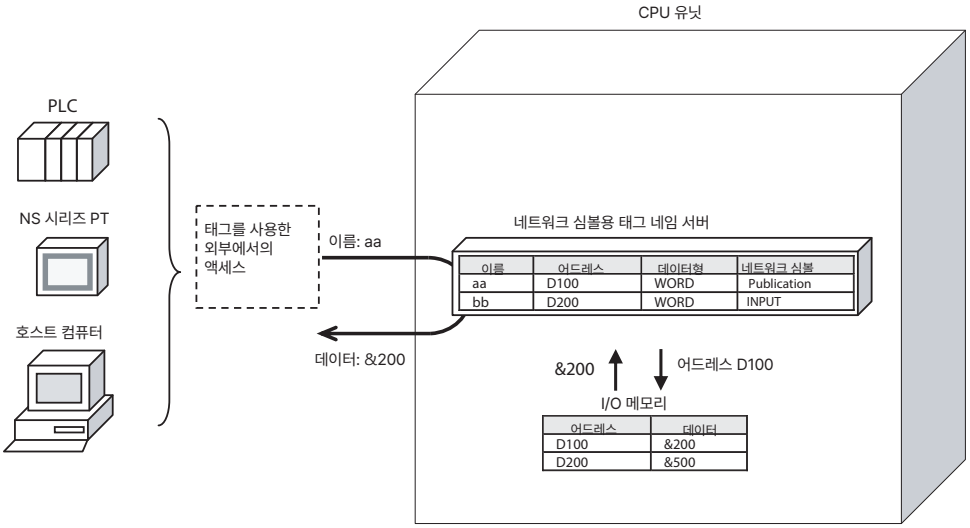
외부 기기에서 태그를 사용하여 CPU 유닛에 정의된 네트워크 심볼을 통해 로컬 CPU 유닛의 I/O 메모리에 액세스할 수 있습니다. CJ2/CJ3 CPU 유닛에는 네트워크 심볼을 실제 I/O 어드레스로 변환할 수 있는 내부 태그 네임 서버가 있으므로, 태그로 네트워크 심볼에 액세스하여 외부에서 CPU 유닛의 I/O 메모리에 액세스할 수 있습니다.

네트워크 심볼은 다음 용도로 사용할 수 있습니다.

・EtherNet/IP 상의 다른 PLC와의 데이터 링크

・호스트나 PT를 사용한 외부에서의 태그 액세스.

네트워크 심볼은 CX-Programmer의 글로벌 심볼 테이블에서 설정하십시오.



참고

태그 데이터 링크를 작성하는 순서에 대해서는 EtherNet/IP 유닛 오퍼레이션 매뉴얼(Cat. No. W465)의 태그 데이터 링크 기능을 참조하십시오.

CJ2H-CPU□□(-EIP 없음) 버전 1.5 이전 및 CJ2M-CPU1□ 버전 2.1 이전에서는 지원하지 않습니다.

AI 번역본
-For Reference Only-

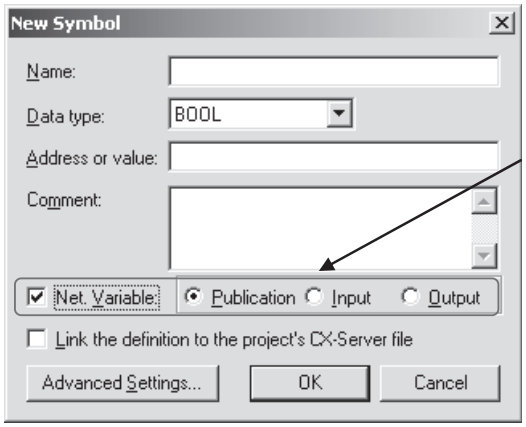
네트워크 심볼 설정

네트워크 심볼은 다음 순서로 설정하십시오.

1 글로벌 변수 테이블에서 마우스 오른쪽 버튼을 클릭하고 Insert Symbol을 선택하십시오.

주 로컬 심볼 테이블에서는 네트워크 심볼을 지정할 수 없습니다.

2 다음 New Symbol 대화 상자가 표시됩니다.
Net, Variable 체크 박스를 선택한 후 Publication, Input 또는 Output 옵션을 선택하고 OK 버튼을 클릭하십시오.

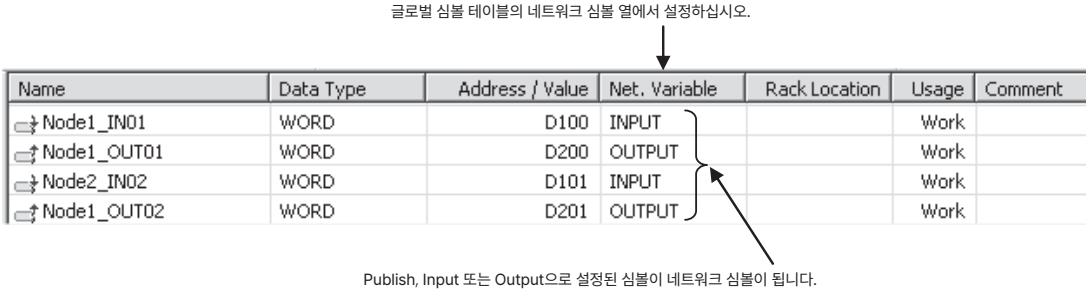


네트워크 심볼 이름에는 최대 48문자를 사용할 수 있습니다. 이름은 대소문자를 구별하지 않습니다.

심볼의 종류	설명	글로벌 심볼 테이블의 네트워크 심볼 열	설명
네트워크 심볼	태그를 사용하여 외부에서 액세스할 수 있는 심볼.	Publication	· CIP 메시지 통신을 사용하여 외부에서 액세스할 수 있는 심볼. · 태그 데이터 링크는 지원하지 않습니다.
		Input	· CIP 메시지 통신 또는 태그 데이터 링크를 사용하여 외부에서 액세스할 수 있는 심볼. · 태그 데이터 링크를 사용할 때 데이터 입력(다른 CPU 유닛에서 로컬 CPU 유닛으로)용 심볼.
		Output	· CIP 메시지 통신 또는 태그 데이터 링크를 사용하여 외부에서 액세스할 수 있는 심볼. · 태그 데이터 링크를 사용할 때 데이터 출력(로컬 CPU 유닛에서 다른 CPU 유닛으로)용 심볼.

AI 번역본
-For Reference Only

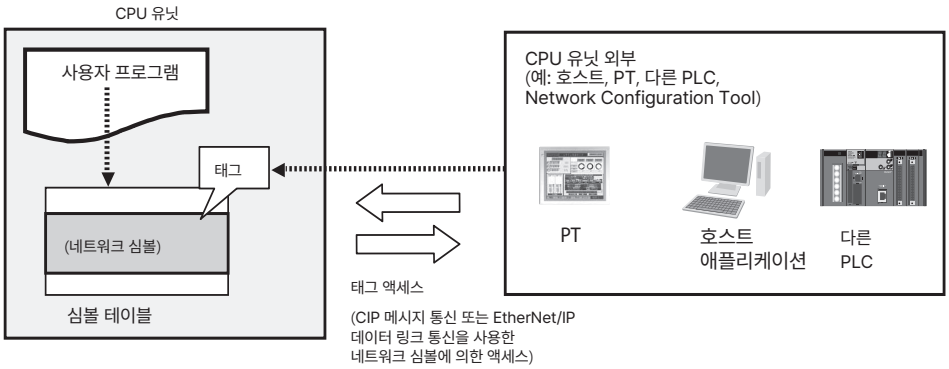
● 글로벌 심볼 테이블의 네트워크 심볼 열



네트워크 심볼의 사용 조건과 취급

● 외부에서의 네트워크 심볼 취급

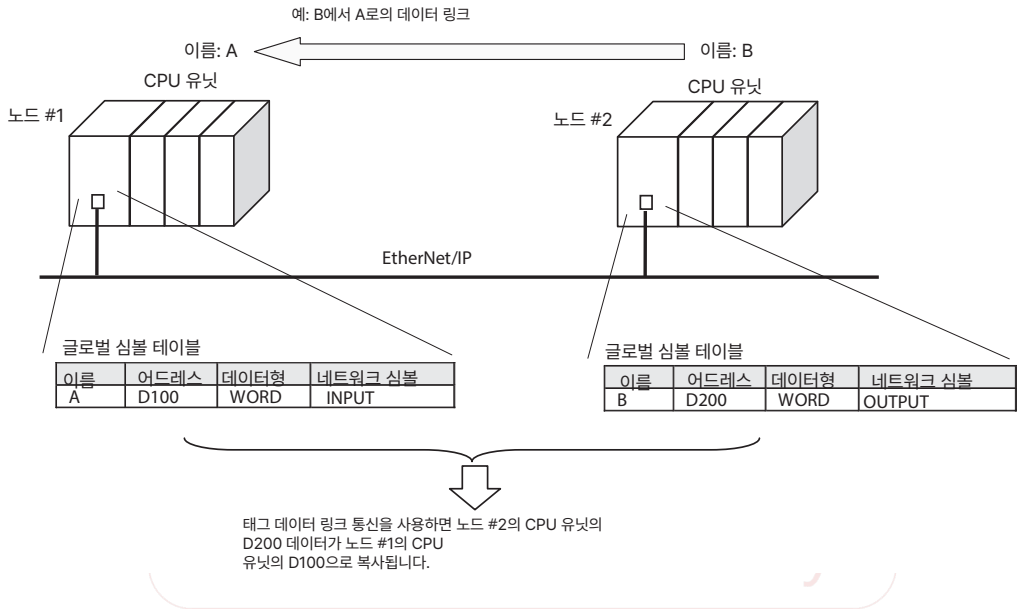
CPU 유닛에서 네트워크 심볼로 설정된 심볼은 PLC 외부(호스트, PT, Network Configuration Tool, 다른 PLC 등)에서 태그로 인식됩니다.



네트워크 심볼의 용도

● 네트워크 심볼을 EtherNet/IP 데이터 링크 태그로 사용

원격 PLC의 데이터는 EtherNet/IP 태그 데이터 링크 통신을 사용하여 태그로 지정할 수 있습니다. 글로벌 심볼 테이블의 네트워크 심볼 열에서 네트워크 심볼을 Input 또는 Output으로 설정하십시오.

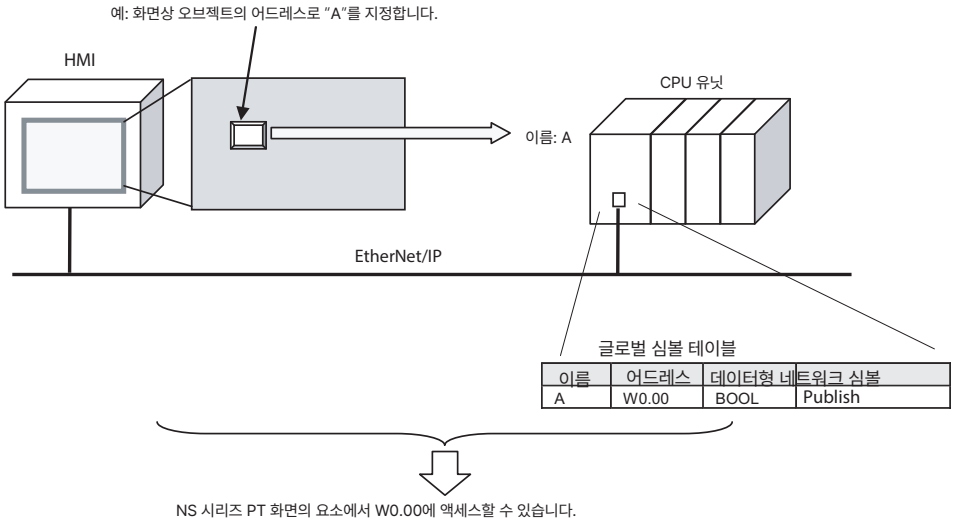


이름	어드레스	데이터형	네트워크 심볼
A	D100	WORD	INPUT

이름	어드레스	데이터형	네트워크 심볼
B	D200	WORD	OUTPUT

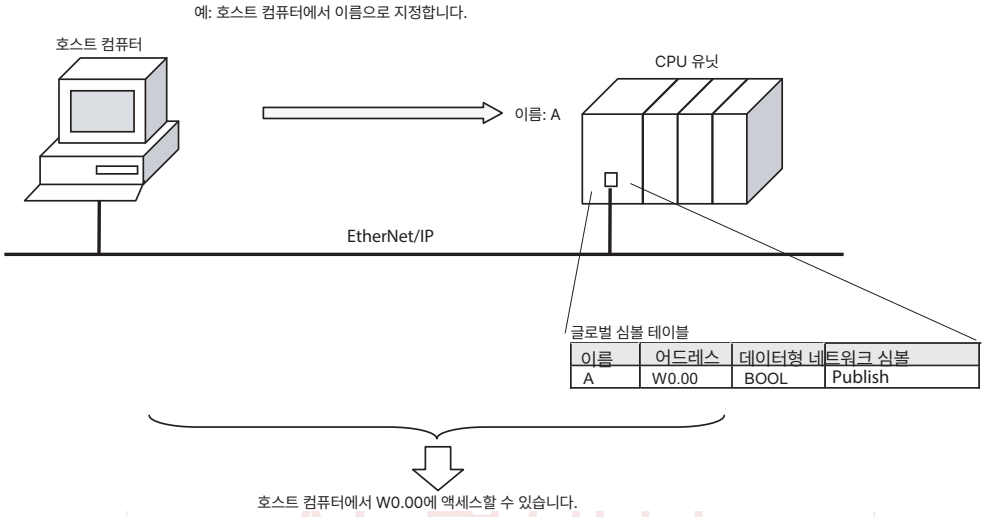
● HMI와의 통신에서 태그 사용

원격 PLC의 데이터는 HMI 화면의 오브젝트를 사용하여 태그로 지정할 수 있습니다. 글로벌 심볼 테이블의 네트워크 심볼 열에서 네트워크 심볼을 Publication으로 설정하십시오.



● 호스트 컴퓨터에서 CIP 메시지 통신으로 태그 사용

원격 PLC의 데이터는 호스트 컴퓨터에서 태그로 지정할 수 있습니다. 글로벌 심볼 테이블에서 네트워크 심볼을 Publish로 설정하십시오.

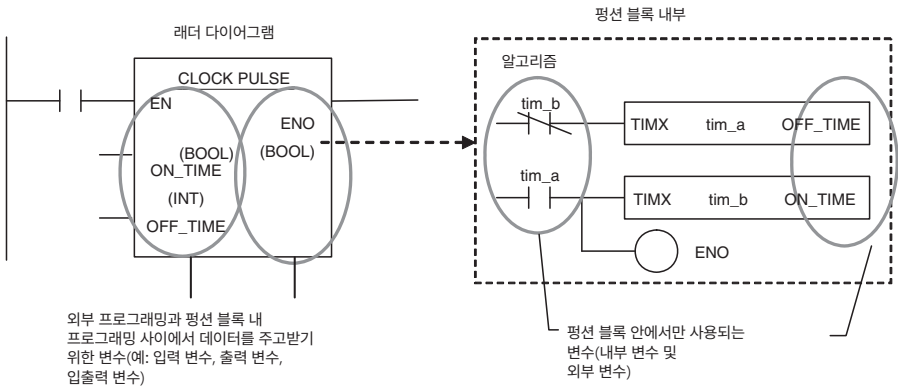


AI 번역본
-For Reference Only-

5-5-6 펑션 블록의 변수

펑션 블록 안의 프로그램은 모두 실제 어드레스가 아니라 변수로 기술됩니다. 펑션 블록 안의 변수는 펑션 블록 밖의 심볼과 변수 종별 및 사양이 다릅니다. 펑션 블록 안의
변수에 대한 자세한 내용은 CX-Programmer 오퍼레이션 매뉴얼: 펑션 블록 및 구조화
텍스트(Cat. No. W447)를 참조하십시오.

● 펑션 블록의 변수 영역



AI 번역본
-For Reference Only-

5-5-7 심볼 데이터형

심볼에 할당된 어드레스에 대해 데이터형을 지정할 수 있습니다.

심볼에 설정할 수 있는 데이터형

펑션 블록 안에서 지정할 수 있는 데이터형은 펑션 블록 밖에서 지정할 수 있는 데이터형과 다를 수 있습니다. 각각에 사용할 수 있는 데이터형은 다음 표에 나타냅니다.

데이터형	프로그래밍의 심볼	펑션 블록의 변수
BOOL (비트)	가능	가능
UINT (1워드 부호 없는 2진)	가능	가능
UDINT (2워드 부호 없는 2진)	가능	가능
ULINT (4워드 부호 없는 2진)	가능	가능
INT (1워드 부호 있는 2진)	가능	가능
DINT (2워드 부호 있는 2진)	가능	가능
LINT (4워드 부호 있는 2진)	가능	가능
UINT BCD (1워드 부호 없는 2진)*1	가능	불가
UDINT BCD (2워드 부호 있는 2진)*1	가능	불가
ULINT BCD (4워드 부호 있는 2진)*1	가능	불가
REAL (2워드 부동소수점)	가능	가능
LREAL (4워드 부동소수점)	가능	가능
CHANNEL (워드)*1	가능	불가
NUMBER (상수 또는 수치)*2	가능	불가
WORD (1워드 16진)	가능	가능
DWORD (2워드 16진)	가능	가능
LWORD (4워드 16진)	가능	가능
STRING (문자열: 1~255 ASCII 문자)	가능	가능
TIMER	불가	가능
COUNTER	불가	가능
사용자 정의 데이터형(데이터 구조체)	가능	가능*3

*1 ST 프로그래밍에서는 데이터형이 다음과 같이 변환됩니다.
- UINT BCD: WORD
- UDINT BCD: DWORD
- ULINT BCD: LWORD
- CHANNEL: WORD

*2 이 데이터형은 ST 프로그래밍에서 사용할 수 없습니다. 프로그램 에러가 발생합니다.

*3 래더 다이어그램으로 작성된 펑션 블록 안의 내부 심볼에 한합니다.

AI 번역본
- For Reference Only -

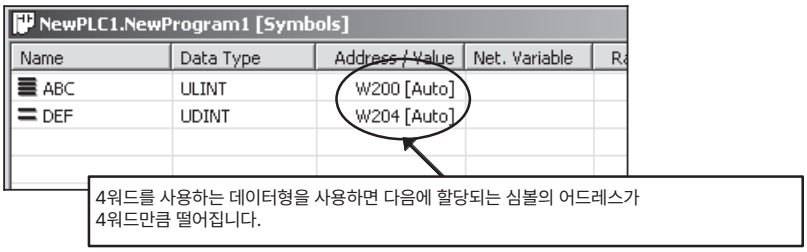
데이터형의 활용

심볼에 설정한 데이터형은 다음 기능에 사용할 수 있습니다.

● 자동 어드레스 할당을 사용한 활용

CX-Programmer에서 자동 어드레스 할당을 사용하면 심볼에 설정한 데이터형에 따라 어드레스를 할당할 수 있습니다.

예: ULINT(4워드 부호 없는 2진) 데이터형 설정



배열 지정

심볼의 속성으로 배열을 지정할 수 있습니다(1차원 배열만). 배열은 STRING과 NUMBER를 제외한 모든 데이터형에 지정할 수 있습니다.

같은 데이터 속성을 가진 여러 심볼을 하나의 그룹으로 관리할 때는 배열 지정을 사용하십시오.
배열 지정을 사용하는 경우에는 다음 순서를 따르십시오.

1 New Symbol 대화 상자 또는 Edit Symbol 대화 상자에서 심볼명을 입력한 후
Advanced Settings 버튼을 클릭하십시오. Advanced Settings 대화 상자가 표시됩니다.

2 Array Settings 필드에서 Array Symbol 옵션을 선택한 후 최대 요소 수를 입력하십시오.

배열 변수를 기술할 때는 변수명 뒤의 대괄호 안에 첨자를 지정합니다.

예: 심볼명이 PV이고 최대 요소 수가 3이면, 명령어의 오퍼랜드에서 배열 변수를 PV[0], PV[1], PV[2]와 같이 지정합니다.

배열의 첨자는 0부터 시작하는 요소 번호로 지정합니다. 요소 번호는 직접 입력할 수도 있고, 심볼이나 메모리 어드레스를 입력하여 간접적으로 지정할 수도 있습니다.

⚠ 주의

래더 또는 ST 프로그래밍에서 요소 번호를 간접적으로 지정하기 위해 배열 변수의 첨자에 심볼이나 메모리 어드레스(ST 프로그래밍에서는 심볼만 사용 가능)를 지정하는 경우, 요소 번호가 메모리 영역의 최대 범위를 넘지 않도록 반드시 확인하십시오.

심볼에 지정된 메모리 영역의 최대 범위를 넘는 요소 번호를 지정하면 다른 메모리 영역의 데이터에 액세스하게 되어, 예기치 않은 동작이 발생할 수 있습니다.



첨자가 워드 어드레스나 심볼인 배열 변수는 즉시 리프레시 형 명령어의 오퍼랜드로 사용할 수 없습니다.

사용자 정의 데이터형(데이터 구조체)

CJ2/CJ3 CPU 유닛에서는 사용자 정의 데이터형으로 구조체를 작성하여 사용할 수 있습니다.

● 데이터 구조체

데이터 구조체는 둘 이상의 데이터형을 하나로 묶은 사용자 정의 데이터형입니다. 데이터형에는 이름을 붙일 수 있습니다. 사용자 정의 데이터형을 사용하는 변수의 이름은 데이터 구조체 안의 변수 중 하나의 이름과 함께 지정합니다. 데이터 구조체 전체를 구조체 변수라고 하고, 데이터 구조체 안의 변수를 멤버라고 합니다.

참고

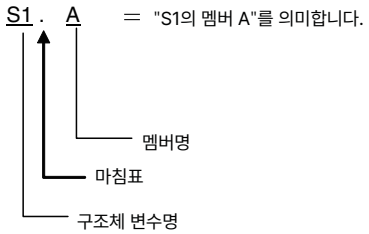
- 여러 개의 데이터를 다루는 데에는 배열도 사용할 수 있습니다. 다만 배열 데이터형은 같은 데이터형의 데이터를 담고 있으며 배열 선두로부터의 오프셋을 지정하여 액세스한다는 점에서 데이터 구조체와 다릅니다. 데이터 구조체에서는 같거나 다른 데이터형의 데이터에 멤버명을 사용하여 액세스합니다. 또한 CX-Programmer에서는 구조체 데이터형에 이름을 붙일 수 있습니다.

- 구조체 변수는 배열에 배치할 수 있습니다.
- 또한 배열 변수를 데이터 구조체의 멤버로 사용할 수 있습니다.

● 구조체 변수의 래더 프로그램 표기와 입력

래더 프로그램에서는 구조체 변수명과 멤버명을 마침표로 구분합니다.

예:

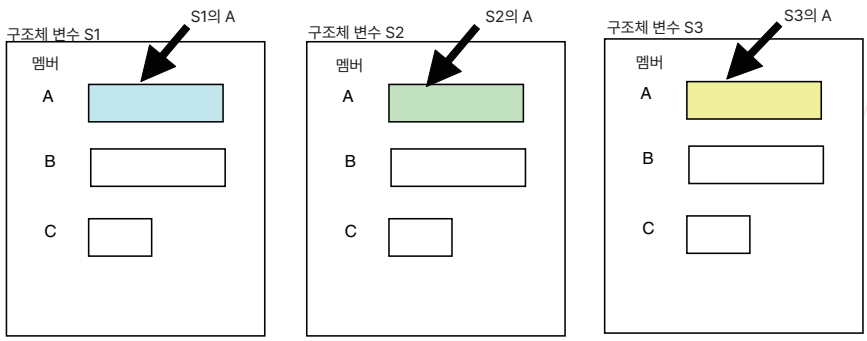


● 데이터 구조체의 활용

묶여 있는 데이터를 지정할 때는 특정 그룹의 멤버를 지정합니다. 즉, "메인 - 서브" 형태의 계층으로 데이터를 지정할 수 있습니다.

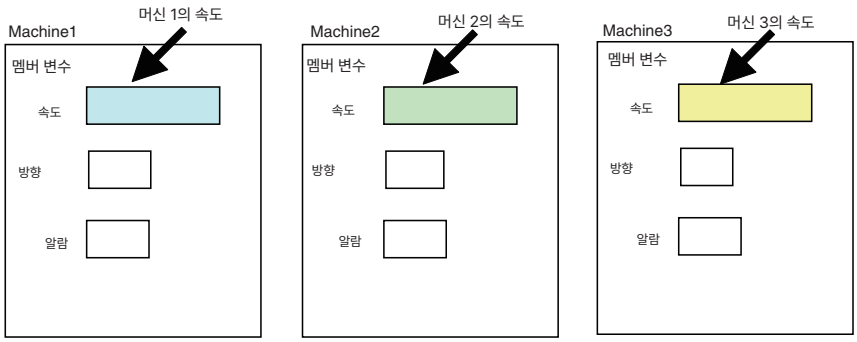
배열을 사용하지 않는 데이터 구조체 사용

멤버 A, B, C를 각각 갖는 구조체 변수 S1, S2, S3의 예



예를 들어 각 머신에 대해 같은 멤버(예: 속도)를 지정할 수 있습니다.

Machine1, Machine2, Machine3이라는 구조체 변수

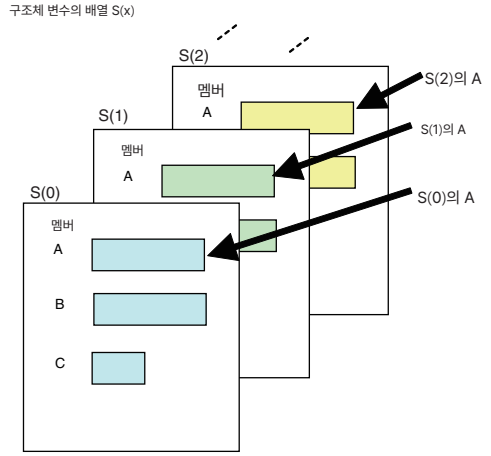


데이터 구조체와 배열의 조합

구조체 변수를 배열에 배치

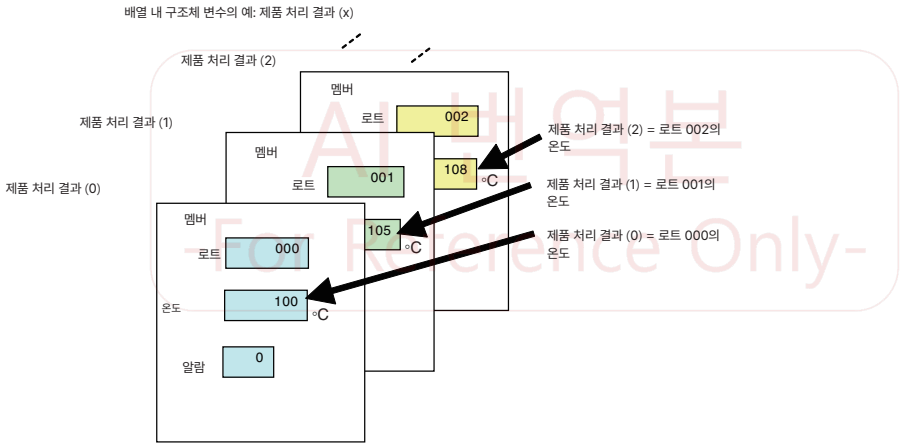
제품별 레시피 데이터처럼 같은 형태의 데이터가 대량으로 있는 경우, 구조체 변수를 배열에 배치할 수 있습니다. 이것은 데이터베이스를 작성할 때 사용합니다. 이 경우 구조체 변수가 하나의 레코드가 되고 각 멤버가 데이터베이스의 필드가 됩니다.

구조체 변수의 배열 S(x)



예를 들어 생산 로트별 처리 결과를 지정하는 데 사용할 수 있습니다.

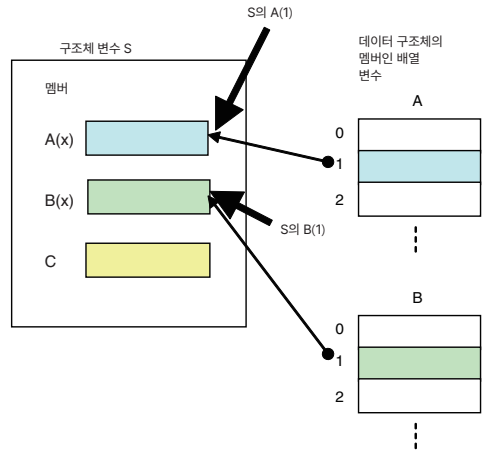
배열 내 구조체 변수의 예: 제품 처리 결과 (x)



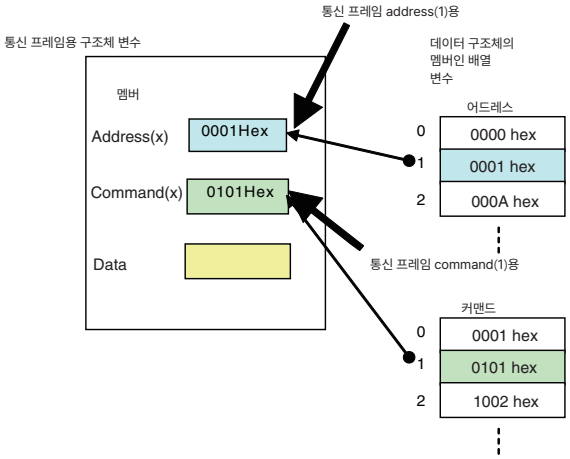
AI 본부
-For Reference Only-

배열 변수를 데이터 구조체의 멤버로 사용

데이터 전체에 특정한 구조가 있고 각 멤버가 라이브러리처럼 여러 요소를 담고 있는 경우, 배열을 데이터 구조체의 멤버로 사용할 수 있습니다. 멤버는 배열에서 자유롭게 지정할 수 있습니다. 이것은 라이브러리로부터 데이터를 구성할 때 사용합니다.



예: 라이브러리에서 요소를 선택하여 통신 프레임을 작성할 수 있습니다.



데이터 구조체의 사양과 작성 순서에 대한 자세한 내용은 CX-Programmer 오퍼레이션 매뉴얼(Cat. No. W446)을 참조하십시오.

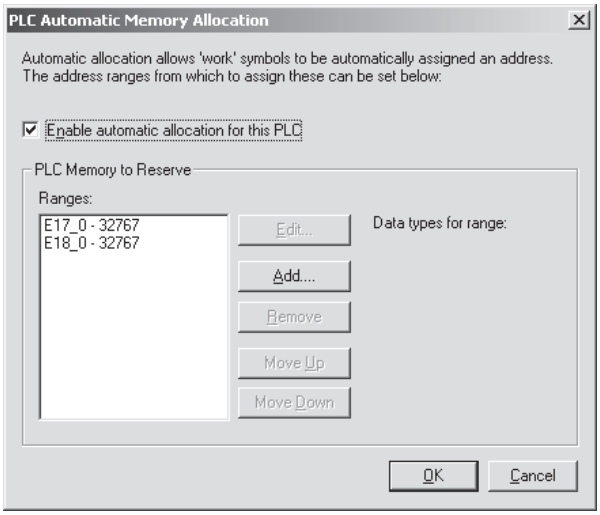
AI 번역본
-For Reference Only-

5-5-8 심볼에 대한 자동 어드레스 할당

프로그래밍 시 심볼에 어드레스를 자동으로 할당할 수 있습니다. CX-Programmer의 PLC 메뉴에서 – Memory Allocation – Automatic Address Allocation을 선택한 후 자동 할당의 범위와 데이터형을 설정하십시오. CJ2/CJ3 CPU 유닛에서는 CIO 영역, 보조 기억 영역, 유지 영역, 워크 영역, DM 영역, EM 영역의 어드레스를 할당할 수 있습니다.

EM 영역에 심볼 어드레스를 자동 할당할 때는 EM 영역의 최상위 뱅크부터 거꾸로 필요한 수만큼의 뱅크를 사용할 것을 권장합니다.

다음 예는 CJ2H-CPU68-EIP CPU 유닛(EM 영역 뱅크가 00~18 hex)의 뱅크 17 및 18 hex에 심볼 어드레스를 자동 할당하는 설정을 나타냅니다.



자동 어드레스 할당의 영역 설정에 대한 자세한 내용은 CX-Programmer 오퍼레이션 매뉴얼(Cat. No. W446)을 참조하십시오.

참고

- CJ2H CPU 유닛에서는 자동 어드레스 할당으로 설정된 EM 영역 뱅크의 비트를 강제 세트/리셋할 수 있습니다. 비트를 강제 세트/리셋할 수 있는 뱅크에 대해서는 6-16 확장 데이터 메모리 영역을 참조하십시오.

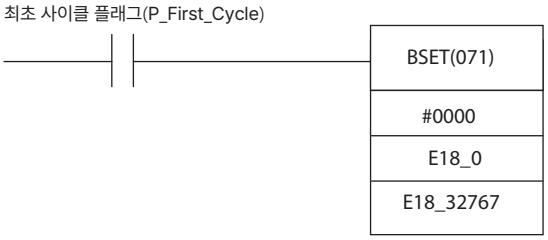
- CX-Programmer 버전 9.2 이상을 사용하는 경우, 구조체 변수에 심볼 어드레스를 자동으로 할당할 수 있습니다.

안전상의 요점

EM 영역의 데이터는 전원이 OFF되거나 동작 모드가 변경될 때 백업됩니다. BOOL 데이터로 지정된 출력 비트를 사용할 때는 주의하십시오. 필요한 경우 메모리를 클리어하는 프로그래밍을 포함시키십시오.

사 본 -For Reference Only-

예: EM 영역 뱅크 18 hex가 자동 어드레스 할당으로 설정되어 있을 때, 전원이 ON되면 다음 명령어로 뱅크 18 hex를 모두 0으로 클리어할 수 있습니다.

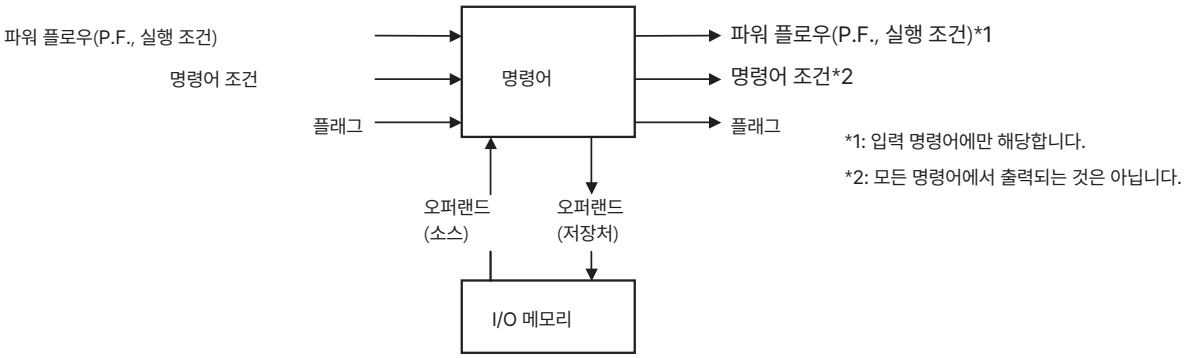


5-6 명령어

5-6-1 명령어의 기본 이해

명령어의 구조

프로그램은 명령어로 구성됩니다. 명령어에 대한 입력과 명령어로부터의 출력에 대한 개념적 구조를 다음 그림에 나타냅니다.

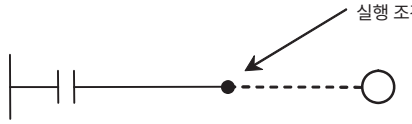


- **파워 플로우**

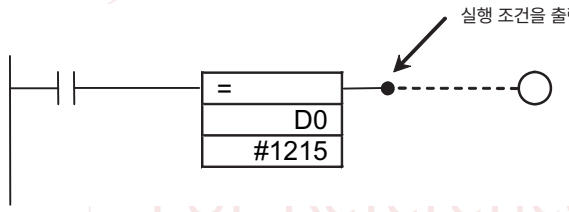
  파워 플로우는 프로그램이 정상적으로 실행될 때 명령어의 실행을 제어하는 데 사용되는 실행 조건입니다. 래더 프로그램에서 파워 플로우는 실행 조건의 상태를 나타냅니다.

입력 명령어

· 로드 명령어는 논리 시작을 나타내며 실행 조건을 출력합니다.



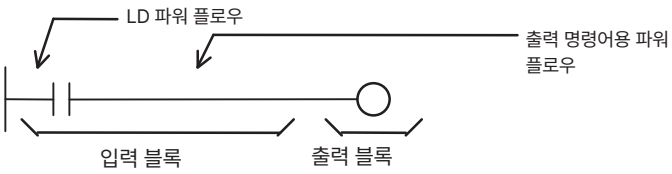
· 중간 명령어는 파워 플로우를 실행 조건으로 입력하고, 중간 명령어 또는 출력 명령어로 파워 플로우를 출력합니다.



For Reference Only

출력 명령어

출력 명령어는 파워 플로우를 실행 조건으로 하여 모든 기능을 실행합니다.



● 명령어 조건

명령어 조건은 명령어 실행 전반에 관계되는 특수한 조건으로, 다음 명령어에 의해 출력됩니다. 명령어를 실행할지 여부를 결정할 때 명령어 조건은 파워 플로우(P.F.)보다 우선순위가 높습니다. 명령어 조건에 따라 명령어가 실행되지 않거나 다르게 동작할 수 있습니다. 명령어 조건은 각 태스크 시작 시에 리셋(해제)됩니다. 즉, 태스크가 바뀔 때 리셋됩니다.

다음 명령어는 특정 명령어 조건을 설정하고 해제하기 위해 쌍으로 사용됩니다. 쌍을 이루는 이 명령어들은 같은 태스크 안에 있어야 합니다.

명령어 조건	설명	설정 명령어	해제 명령어
인터록 중	인터록은 프로그램의 일부를 OFF합니다. 출력 비트를 OFF하고, 타이머를 리셋하며, 카운터를 유지하는 등의 특수한 조건이 적용됩니다.	IL(002)	ILC(003)
BREAK(514) 실행	실행 중인 FOR(512) - NEXT(513) 루프를 종료합니다. (NEXT(513) 명령어까지 모든 명령어의 실행을 막습니다.)	BREAK(514)	NEXT(513)
	JMP0(515)에서 JME0(516)으로의 점프를 실행합니다.	JMP0(515)	JME0(516)
블록 프로그램 실행	BPRG(096)부터 BEND(801)까지의 프로그램 블록을 실행합니다.	BPRG(096)	BEND(801)

●플래그

여기서 플래그란 명령어 사이의 인터페이스 역할을 하는 비트를 말합니다.

입력 플래그		출력 플래그	
플래그	설명	플래그	설명
캐리(CY) 플래그	캐리 플래그는 데이터 시프트 명령어와 가산/감산 명령어에서 지정되지 않은 오퍼랜드로 사용됩니다.	조건 플래그	조건 플래그에는 상시 ON/OFF 플래그와, 명령어 실행 결과에 의해 갱신되는 플래그가 포함됩니다. 사용자 프로그램에서는 이 플래그들을 어드레스가 아니라 P_On, P_Off, P_ER, P_CY, P_EQ 같은 라벨로 지정할 수 있습니다.
특수 명령어용 플래그	FPD(269) 명령어의 티칭 플래그와 네트워크 통신 가능 플래그 등이 여기에 해당합니다.	특수 명령어용 플래그	메모리 카드 명령어 플래그와 MSG(046) 실행 완료 플래그 등이 여기에 해당합니다.

AI 번역본
-For Reference Only-

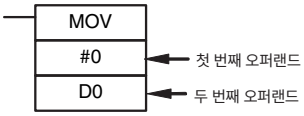
● 오퍼랜드

오퍼랜드는 I/O 메모리 영역의 내용이나 상수를 지정하는 데 사용되는, 미리 정해진 명령어 파라미터(래더 다이어그램의 상자)를 지정합니다. 오퍼랜드로 어드레스나 상수를 입력하여 명령어를 실행할 수 있습니다. 오퍼랜드는 소스, 저장처, 번호 오퍼랜드로 분류됩니다.



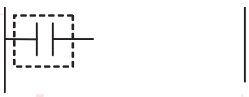
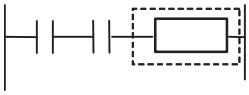
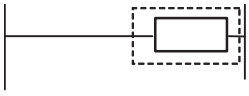
오퍼랜드 종별		오퍼랜드 기호	설명	
소스	읽어 들일 데이터의 어드레스 또는 상수를 지정합니다.	S	소스 오퍼랜드	제어 데이터(C) 이외의 소스 오퍼랜드
		C	제어 데이터	비트 상태에 따라 뜻이 달라지는 소스 오퍼랜드 안의 복합 데이터.
저장처 (결과)	데이터가 기록될 어드레스를 지정합니다.	D	---	
번호	점프 번호나 서브루틴 번호처럼 명령어에서 사용되는 특정 번호를 지정합니다.	N	---	

주 오퍼랜드는 명령어의 위에서부터 차례로 첫 번째 오퍼랜드, 두 번째 오퍼랜드와 같이 부르기도 합니다.



명령어의 위치와 실행 조건

다음 표는 명령어가 놓일 수 있는 위치를 나타냅니다. 명령어는 실행 조건이 필요한 것과 필요하지 않은 것으로 나뉩니다. 명령어에 대한 자세한 내용은 A-1 명령어 기능을 참조하십시오.

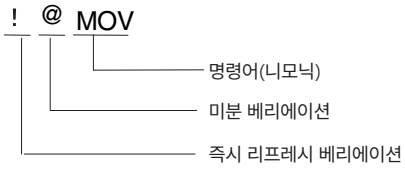
명령어		위치	실행 조건	다이어그램	예
입력 명령어	논리 시작 (로드 명령어)	좌측 모선에 직접 접속되거나 명령어 블록의 선두에 있습니다.	필요 없음.		LD, LD TST(350), LD > (및 기타 기호 비교 명령어)
	중간 명령어	논리 시작과 출력 명령어 사이입니다.	필요		AND, OR, AND TEST(350), AND > (및 기타 ADD 기호 비교 명령어), UP(521), DOWN(522), NOT(520) 등.
출력 명령어		우측 모선에 직접 접속됩니다.	필요		OUT과 MOV(021)을 포함한 대부분의 명령어.
			필요 없음.		END(001), JME(005), FOR(512), ILC(003), 등.

AI 번역본
-For Reference Only

명령어 베리에이션

명령어에는 실행 조건을 미분하거나 명령어 실행 시 데이터를 리프레시(즉시 리프레시)하기 위한 다음과 같은 베리에이션이 있습니다.

베리에이션		기호	설명
미분	ON	@	실행 조건이 ON될 때 미분하는 명령어입니다.
	OFF	%	실행 조건이 OFF될 때 미분하는 명령어입니다.
즉시 리프레시		!	명령어 실행 시 오퍼랜드로 지정된 I/O 영역 또는 특수 I/O 유닛 워드의 데이터를 리프레시합니다.



실행 조건

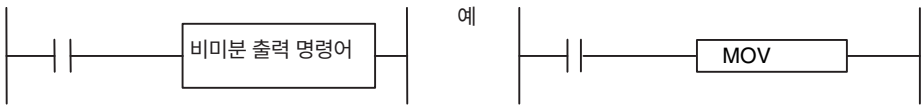
기본 명령어와 특수 명령어에는 다음 두 종류를 사용할 수 있습니다.

・비미분 명령어: 매 사이클 실행됩니다

・미분 명령어: 한 번만 실행됩니다

● 비미분 명령어

출력 명령어(입력 조건이 필요한 명령어):
실행 조건이 성립해 있는(ON 또는 OFF) 동안 매 사이클 한 번씩 실행됩니다.



입력 명령어(논리 시작 및 중간 명령어):
매 사이클 비트 상태를 읽고, 비교하고, 비트를 테스트하는 등의 처리를 수행합니다. 결과가 ON이면 파워 플로우가 출력됩니다(즉, 실행 조건이 ON됩니다).

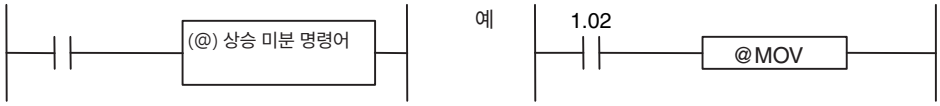


AI 번역본
-For Reference Only-

● 입력 미분 명령어

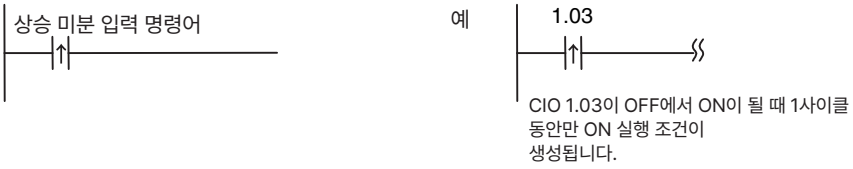
상승 미분 명령어(명령어 앞에 @가 붙는 것)

출력 명령어:
실행 조건이 OFF에서 ON으로 바뀐 사이클에서만 실행되며, 이후 사이클에서는 실행되지 않습니다.



CIO 1.02가 OFF → ON될 때 MOV 명령어를 한 번 실행합니다.

입력 명령어(논리 시작 및 중간 명령어):
이 명령어는 매 사이클 비트 상태를 읽고, 비교하고, 비트를 테스트하는 등의 처리를 수행하며, 결과가 OFF에서 ON으로 바뀔 때 ON 실행 조건(파워 플로우)을 출력합니다. 실행 조건은 다음 사이클에 OFF됩니다.

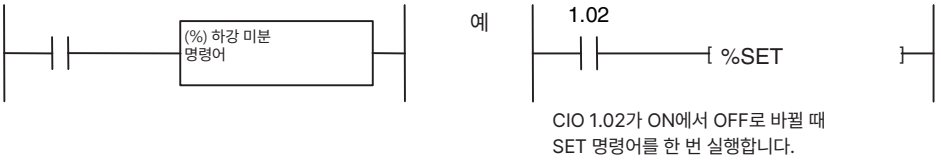


입력 명령어(논리 시작 및 중간 명령어):
이 명령어는 매 사이클 비트 상태를 읽고, 비교하고, 비트를 테스트하는 등의 처리를 수행하며, 결과가 OFF에서 ON으로 바뀔 때 OFF 실행 조건(파워 플로우 정지)을 출력합니다. 실행 조건은 다음 사이클에 ON됩니다.

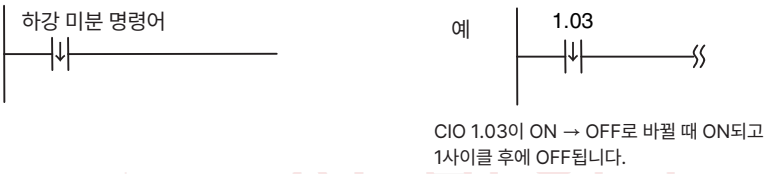


하강 미분 명령어(명령어 앞에 %가 붙는 것)

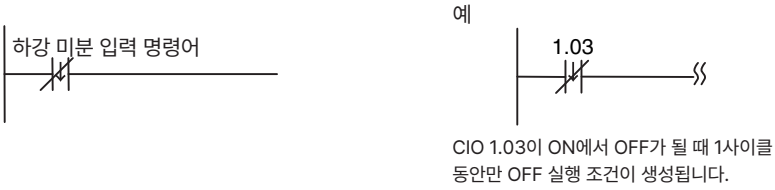
출력 명령어:
실행 조건이 ON에서 OFF로 바뀐 사이클에서만 실행되며, 이후 사이클에서는 실행되지 않습니다.



입력 명령어(논리 시작 및 중간 명령어):
이 명령어는 매 사이클 비트 상태를 읽고, 비교하고, 비트를 테스트하는 등의 처리를 수행하며, 결과가 ON에서 OFF로 바뀔 때 실행 조건(파워 플로우)을 출력합니다. 실행 조건은 다음 사이클에 OFF됩니다.



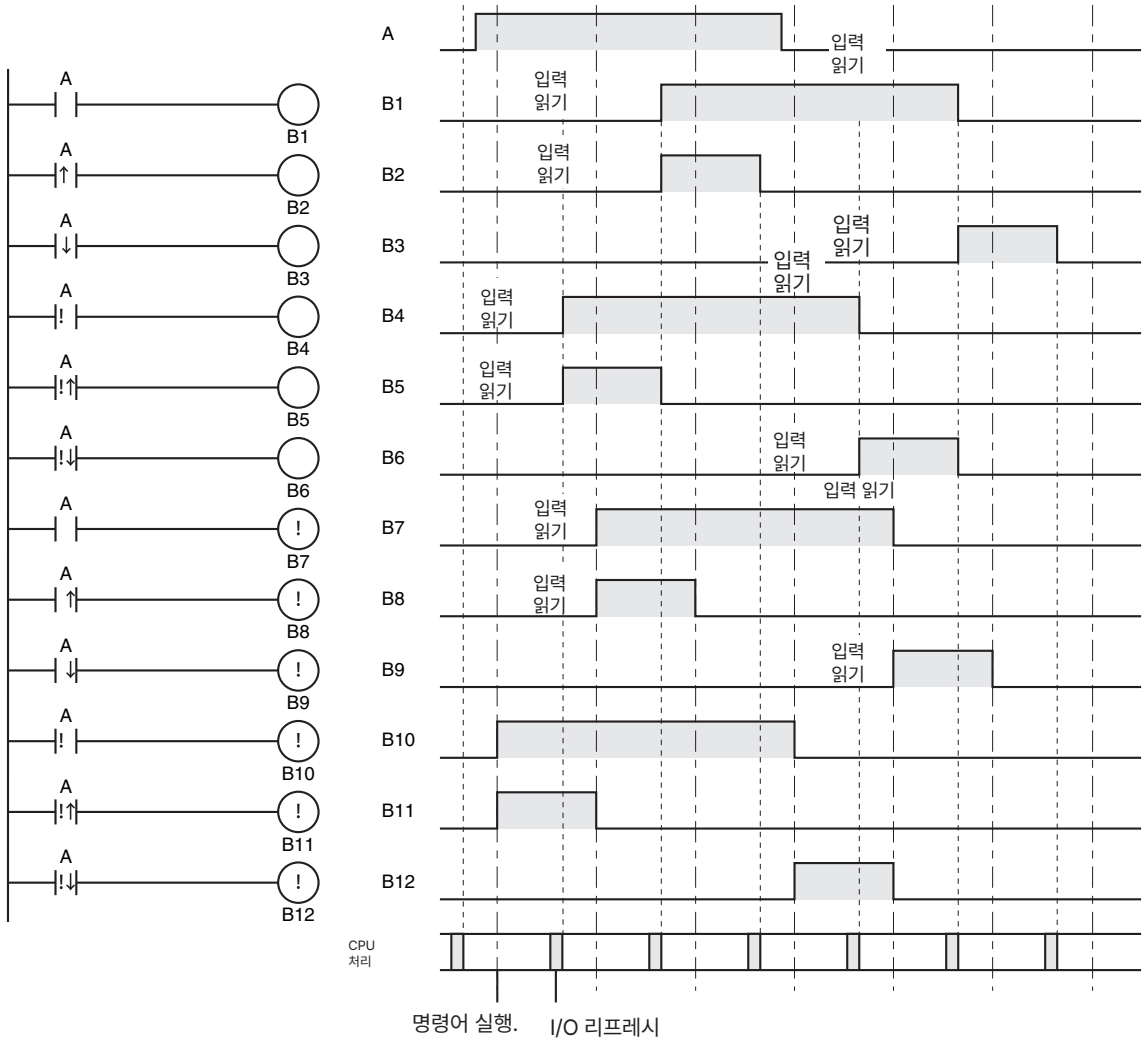
입력 명령어(논리 시작 및 중간 명령어):
이 명령어는 매 사이클 비트 상태를 읽고, 비교하고, 비트를 테스트하는 등의 처리를 수행하며, 결과가 ON에서 OFF로 바뀔 때 OFF 실행 조건(파워 플로우 정지)을 출력합니다. 실행 조건은 다음 사이클에 ON됩니다.



-For Reference Only-

I/O 명령어 타이밍

다음 타이밍 차트는 LD 명령어와 OUT 명령어만으로 구성된 프로그램을 사용하여 각 명령어의 서로 다른 동작 타이밍을 나타냅니다.



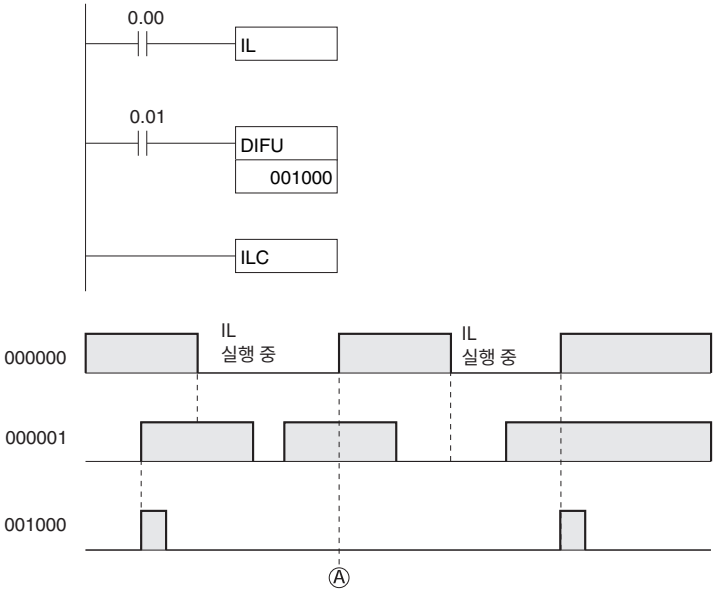
● 미분 명령어

- 미분 명령어에는 이전 값이 ON인지 OFF인지를 나타내는 내부 플래그가 있습니다. 운전 개시 시 상승 미분 명령어(DIFU 및 @ 명령어)의 이전 값 플래그는 ON으로 설정되고, 하강 미분 명령어(DIFD 및 % 명령어)의 이전 값 플래그는 OFF로 설정됩니다. 이에 따라 운전 개시 시 미분 출력이 예기치 않게 출력되는 것을 방지합니다.
- 상승 미분 명령어(DIFU 또는 @ 명령어)는 실행 조건이 ON이고 이전 값 플래그가 OFF일 때만 ON을 출력합니다.

For Reference Only

인터록(IL - ILC 명령어)에서 미분 명령어 사용

다음 예에서는 인터록이 적용되어 있는 동안 값이 갱신되지 않으므로, 미분 명령어의 이전 값 플래그가 인터록 직전의 값을 유지하며 A 지점에서 미분 출력을 출력하지 않습니다.



점프(JMP(004) - JME(005) 명령어)에서 미분 명령어 사용

인터록과 마찬가지로, 명령어가 점프되었을 때 미분 명령어의 이전 값 플래그는 변경되지 않습니다. 즉, 이전 값이 유지됩니다.

- 하강 미분 명령어(DIFD(014) 또는 %가 붙은 명령어)에서는 이전 값 플래그가 ON일 때만 입력이 OFF될 때 출력이 ON됩니다.
- 상승 미분 명령어와 하강 미분 명령어 모두, 출력은 다음 사이클에 OFF됩니다.

사용상의 주의

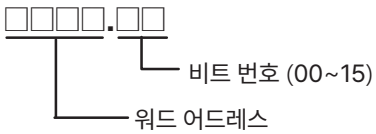
상승 미분 명령어의 입력 비트로 상시 ON 플래그 P_On 또는 A200.11(최초 사이클 플래그)을 사용하지 마십시오. 하강 미분 명령어의 입력 비트로 상시 OFF 플래그 P_Off를 사용하지 마십시오. 어느 쪽이든 사용하면 그 명령어는 절대 실행되지 않습니다.

AI 번역본
-For Reference Only-

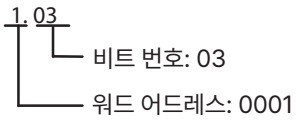
5-6-2 오퍼랜드 지정

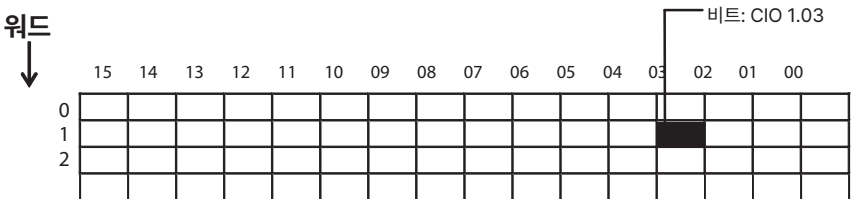
I/O 메모리 영역의 어드레싱

● **비트 어드레스**



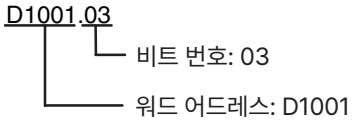
예: CIO 영역의 워드 1에 있는 비트 03의 어드레스는 아래와 같습니다.

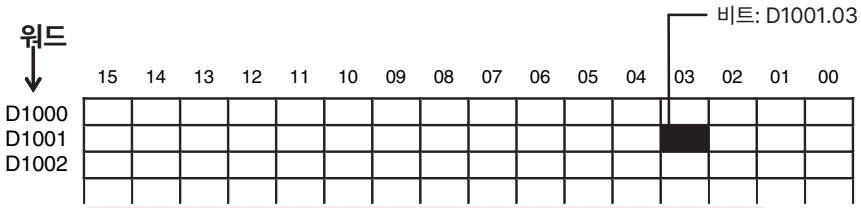




CJ2/CJ3 CPU 유닛에서는 DM 영역과 EM 영역에서 비트 어드레스를 지정할 수 있습니다.

예: DM 영역





AI 번역본
-For Reference Only-

● 워드 어드레스

□□□□
└── 워드 어드레스를 나타냅니다

예: I/O 영역

10
└── 워드 어드레스

DM 영역과 EM 영역의 어드레스에는 어드레스 D200에 대해 아래에 나타낸 것처럼 "D" 또는 "E" 접두어가 붙습니다.

예: DM 영역

D200
└── 워드 어드레스

예: EM 영역

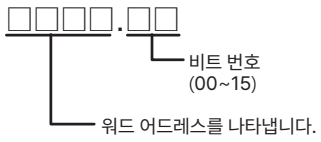
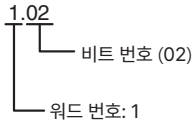
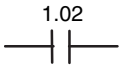
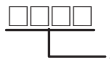
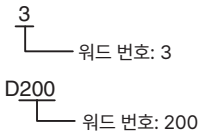
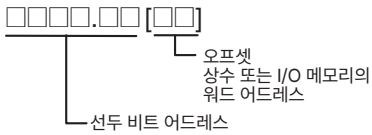
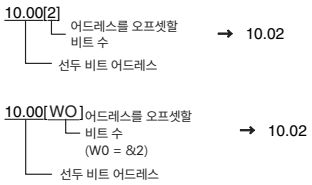
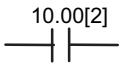
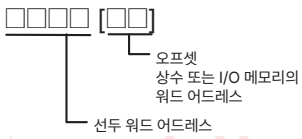
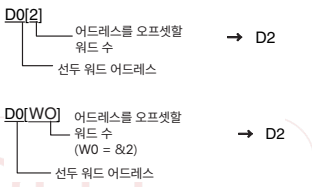
E200
└── 워드 어드레스

예: EM 영역 뱅크 1

E01_200
├── 워드 어드레스
└── 뱅크 번호

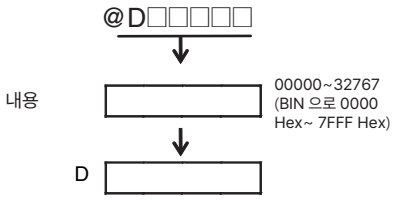
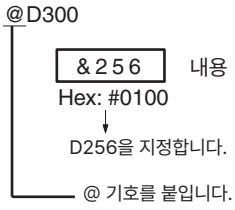
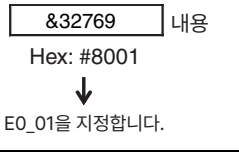
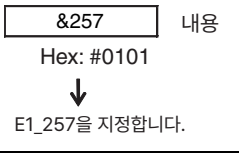
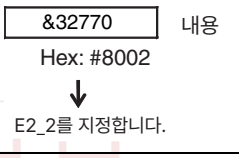
AI 번역본
-For Reference Only-

오퍼랜드 지정

오퍼랜드	설명	표기	적용 예
비트 어드레스 지정	워드 번호와 비트 번호를 직접 지정하여 비트(입력 비트)를 지정합니다.  *1	 	
워드 어드레스 지정	워드 번호를 직접 지정하여 16비트 워드를 지정합니다.  	 	MOV 3 D200
비트 어드레스의 오프셋 지정	대괄호 안에, 지정한 선두 비트 어드레스를 오프셋할 비트 수를 지정합니다.  선두 비트 어드레스에는 배열 변수도 지정할 수 있습니다. 물리 어드레스를 사용하든 배열 변수를 사용하든 CIO, 유지, 워크, DM, EM 영역의 어드레스만 사용할 수 있습니다. 오프셋에는 상수 또는 I/O 메모리의 워드 어드레스를 사용할 수 있습니다. 워드 어드레스를 지정하면 그 워드의 내용이 그대로 오프셋으로 사용됩니다.	 	
워드 어드레스의 오프셋 지정	대괄호 안에, 지정한 선두 비트 어드레스를 오프셋할 비트 수를 지정합니다.  선두 워드 어드레스에는 배열 변수도 지정할 수 있습니다. 물리 어드레스를 사용하든 배열 변수를 사용하든 CIO, 유지, 워크, DM, EM 영역의 어드레스만 사용할 수 있습니다. 오프셋에는 상수 또는 I/O 메모리의 워드 어드레스를 사용할 수 있습니다. 워드 어드레스를 지정하면 그 워드의 내용이 그대로 오프셋으로 사용됩니다.	 	MOV 3 D0[200]

*1 타이머/카운터의 완료 플래그와 현재값에는 동일한 어드레스가 사용됩니다. 태스크 플래그의 어드레스도 하나뿐입니다.

-AI 번역본 Reference Only-

오퍼랜드	설명	표기	적용 예
바이너리 모드에서의 DM/EM 간접 어드레스 지정*2	영역 선두로부터의 오프셋을 지정합니다. 그 어드레스의 내용은 바이너리 데이터(00000~32767)로 취급되어 DM 영역 또는 EM 영역의 워드 어드레스를 지정합니다. 바이너리 모드로 간접 어드레스를 지정하려면 앞에 @ 기호를 붙입니다. 		
	@D□□□□□의 내용이 16진수 0000~7FFF (10진수 0~32767)이면 D0~D32767이 지정됩니다.		MOV #0001 @D300
	@D□□□□□의 내용이 16진수 8000~FFFF(10진수 32768~65535)이면 그때 EM 영역 뱅크 0의 E0_0~E0 _32767이 지정됩니다.		
	@E□_□□□□□의 내용이 16진수 0000~7FFF (10진수 0~32767)이면 지정한 뱅크의 E□_0~E□_32767이 지정됩니다.		MOV #0001@E1_200
	참고 @E□_□□□□□ 내용이 16진수 8000~FFFF(10진수 32768~65535)이면 그때 지정 뱅크 @ 다음 뱅크의 E(□+1)_0~E(□+1)_32767이 지정됩니다.		

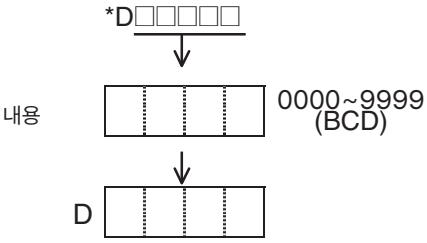
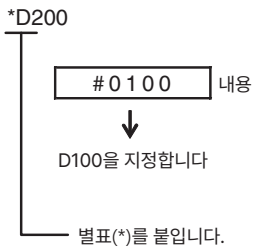
*2 바이너리 모드로 간접 어드레스를 지정할 때는 DM 영역과 EM 영역 (뱅크 0~18 hex)을 하나로 이어진 어드레스로 취급합니다. @ 기호가 붙은 어드레스의 내용이 32767을 넘으면, 그 어드레스는 EM 영역 뱅크 0의 0부터 이어지는 어드레스로 간주됩니다.

예: DM 영역 워드의 내용이 32768이면 EM 영역 뱅크 0의 E0_0이 지정됩니다.

EM 영역의 뱅크 번호를 "n"으로 지정하고 그 워드의 내용이 32767을 넘으면, 그 어드레스는 EM 영역 뱅크 n+1의 0부터 이어지는 어드레스로 간주됩니다.

예: EM 영역 뱅크 2의 내용이 32768이면 EM 영역 뱅크 3의 E3_0이 지정됩니다.

-For Reference Only-

오퍼랜드	설명	표기	적용 예
BCD 모드에서의 DM/EM 간접 어드레스 지정	영역 선두로부터의 오프셋을 지정합니다. 그 어드레스의 내용은 BCD 데이터(0000~9999)로 취급되어 DM 영역 또는 EM 영역의 워드 어드레스를 지정합니다. BCD 모드로 간접 어드레스를 지정하려면 앞에 별표(*)를 붙입니다. 		MOV #0001 *D200

AI 번역본
-For Reference Only-

오퍼랜드	설명		표기	적용 예
레지스터를 직접 지정*	IR□(□: 0~15) 또는 DR□(□: 0~15)를 지정하여 인덱스 레지스터(IR) 또는 데이터 레지스터(DR)를 직접 지정합니다.		IR0 IR1	MOVR 1.02 IR0 CIO 0010의 PLC 메모리 어드레스를 IR0에 저장합니다. MOVR 10 IR1 CIO 0010의 PLC 메모리 어드레스를 IR1에 저장합니다.
레지스터를 사용한 간접 어드레스 지정*	간접 어드레스 (오프셋 없음)	IR@에 들어 있는 PLC 메모리 어드레스의 비트 또는 워드가 지정됩니다. 명령어 오퍼랜드의 비트와 워드를 지정하려면 ,IR□을 지정합니다.	,IR0 ,IR1	LD ,IR0 IR0에 있는 PLC 메모리 어드레스의 비트를 로드합니다. MOV #0001 ,IR1 IR1에 있는 PLC 메모리의 워드에 #0001을 저장합니다.
	상수 오프셋	IR□에 있는 PLC 메모리 어드레스 + 또는 − 상수의 비트 또는 워드가 지정됩니다. +/− 상수 ,IR□을 지정합니다. 상수 오프셋의 범위는 −2048~+2047 (10진수)입니다. 오프셋은 명령어가 실행될 때 바이너리 데이터로 변환됩니다.	+5 ,IR0 31 ,IR1	LD +5 ,IR0 IR0 + 5의 PLC 메모리 어드레스에 있는 비트를 로드합니다. MOV #0001 +31 ,IR1 IR1 + 31의 PLC 메모리 어드레스에 있는 워드에 #0001을 저장합니다
	DR 오프셋	IR□에 있는 PLC 메모리 어드레스 + DR□의 내용에 해당하는 비트 또는 워드가 지정됩니다. DR□ ,IR□을 지정합니다. DR(데이터 레지스터)의 내용은 부호 있는 2진 데이터로 취급됩니다. 부호 있는 2진 값이 음수이면 IR□의 내용에 음의 오프셋이 주어집니다.	DR0 ,IR0 DR0 ,IR1	LD DR0 ,IR0 IR0 + DR0의 값에 해당하는 PLC 메모리 어드레스의 비트를 로드합니다. MOV #0001 DR0 ,IR1 IR1 + DR0의 값에 해당하는 PLC 메모리 어드레스의 워드에 #0001을 저장합니다.
	오토 인크리먼트	값을 PLC 메모리 어드레스로 참조한 후 IR□의 내용이 +1 또는 +2만큼 증가합니다. +1: ,IR□+ 를 지정합니다 +2: ,IR□ + + 를 지정합니다	,IR0++ ,IR1+	LD ,IR0 ++ IR0의 PLC 메모리 어드레스에 있는 비트를 로드한 후 IR0의 내용을 2만큼 증가시킵니다. MOV #0001 ,IR1 + IR1의 PLC 메모리 어드레스에 있는 워드에 #0001을 저장한 후 IR1의 내용을 1만큼 증가시킵니다.
	오토 디크리먼트	값을 PLC 메모리 어드레스로 참조한 후 IR□의 내용이 −1 또는 −2만큼 감소합니다. −1: ,−IR□ 을 지정합니다 −2: ,− −IR□ 을 지정합니다	,--IR0 ,IR1	LD ,− −IR0 IR0의 내용을 2만큼 감소시킨 후 IR0의 PLC 메모리 어드레스에 있는 비트를 로드합니다. MOV #0001 ,−IR1 IR1의 내용을 1만큼 감소시킨 후 IR1의 PLC 메모리 어드레스에 있는 워드에 #0001을 저장합니다.

* 자세한 사용 방법은 5-7 인덱스 레지스터를 참조하십시오.

AI 번역본
For Reference Only

데이터	오퍼랜드	데이터 형	기호	범위	적용 예
16비트 상수	모든 바이너리 데이터 또는 제한된 범위의 바이너리 데이터	부호 없는 2진	#	#0000~#FFFF	MOV #0100 D0 #0100 hex(10진수 &256)를 D0에 저장합니다. +#0009 #0001 D1 #000A hex(10진수 &10)를 D1에 저장합니다.
		부호 있는 10진	±	−32768~+32767	MOV −100 D0 10진수 −100(#FF9C hex)을 D0에 저장합니다. +−9 -1 D1 10진수 −10(#FFF6 hex)을 D1에 저장합니다.
		부호 없는 10진	&	&0~&65535	MOV &256 D0 10진수 −256(#0100 hex)을 D0에 저장합니다. +&9 &1 D1 10진수 −10(#000A hex)을 D1에 저장합니다.
	모든 BCD 데이터 또는 제한된 범위의 BCD 데이터	BCD	#	#0000~#9999	MOV #0100 D0 #0100(BCD)을 D0에 저장합니다. +B #0009 #0001 D1 #0010(BCD)을 D1에 저장합니다.
32비트 상수	모든 바이너리 데이터 또는 제한된 범위의 바이너리 데이터	부호 없는 2진	#	#00000000~ #FFFFFFFF	MOVL #12345678 D0 #12345678 hex를 D0과 D1에 저장합니다. D1 D0 1234 5678
		부호 있는 2진	+	−2147483648~ +2147483647	MOVL −12345678 D0 10진수 −12345678을 D0과 D1에 저장합니다.
		부호 없는 10진	&	&0~ &4294967295	MOVL &12345678 D0 10진수 &12345678을 D0과 D1에 저장합니다.
	모든 BCD 데이터 또는 제한된 범위의 BCD 데이터	BCD	#	#00000000~ #99999999	MOVL #12345678 D0 #12345678(BCD)을 D0과 D1에 저장합니다

AI 번역본
-For Reference Only-

데이터	설명	기호	예	---
문자열	문자열 데이터는 ASCII(특수 문자를 제외하고 1바이트)로 저장되며, 왼쪽 바이트에서 오른쪽 바이트 순으로, 그리고 오른쪽(가장 작은) 워드에서 왼쪽 워드 순으로 저장됩니다. 문자 수가 홀수이면 마지막 워드의 오른쪽 바이트에 00 hex (NUL 코드)가 저장됩니다. 문자 수가 짝수이면 마지막 워드 + 1의 왼쪽과 오른쪽 빈 바이트에 0000 hex (NUL 코드 2개)가 저장됩니다.	---	↓ ABCDE 'A' 'B' 'C' 'D' 'E' NUL ‖ 41 42 43 44 45 00 ABCD 'A' 'B' 'C' 'D' NUL NUL ‖ 41 42 43 44 00 00	MOV$ D100 D200 D100 41 42 D101 43 44 D102 45 00 ↓ D200 41 42 D201 43 44 D202 45 00

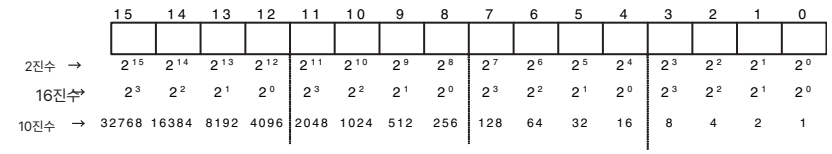
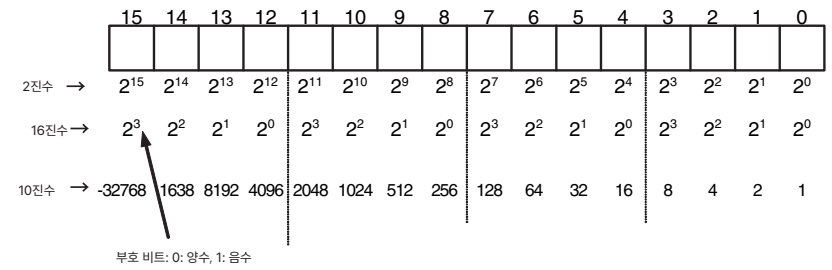
문자열에 사용할 수 있는 ASCII 문자에는 영숫자, 가타카나, 기호(특수 문자 제외)가 있습니다. 사용할 수 있는 문자는 다음 표와 같습니다.

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

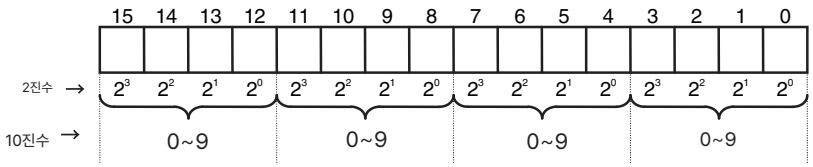
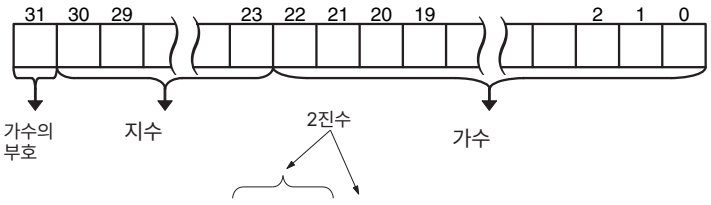
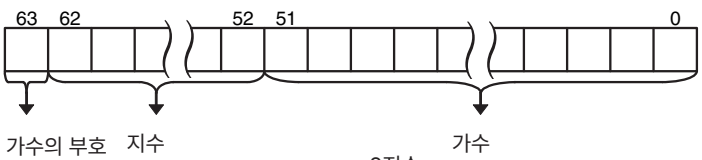
-For Reference Only-

5-6-3 데이터 형식

CJ 시리즈가 취급할 수 있는 데이터 형식은 다음 표와 같습니다.

데이터 종류	데이터 형식	10진수	4자리 16진수
부호 없는 2진		&0~ &65535	#0000~#FFFF
부호 있는 2진	 데이터는 최상위 비트를 부호 비트로 사용하는 부호 있는 2진 16비트 데이터로 취급됩니다. 값은 4자리 16진수로 표현됩니다. 양수: 최상위 비트가 OFF이면 음이 아닌 값을 나타냅니다. 4자리 16진수로는 값이 0000~7FFF hex가 됩니다. 음수: 최상위 비트가 ON이면 음의 값을 나타냅니다. 4자리 16진수로는 값이 8000~FFFF hex가 되며, 그 음수의 절대값(10진수)에 대한 2의 보수로 표현됩니다.	0~−32768 0~+32767	음수: #8000~#FFFF 양수: #0000~#7FFF

AI 번역본
-For Reference Only-

데이터 종류	데이터 형식	10진수	4자리 16진수
BCD (2진화 10진수)		#0~#9999	#0000~#9999
단정밀도 부동소수점 10진수	 값 = (−1)부호 × 1.[가수] × 2지수 부호 (비트 31) 1: 음수 또는 0: 양수 가수 비트 00부터 비트 22까지 23비트에는 가수, 즉 1.□□□□.....에서 소수점 아래 부분이 2진수로 들어갑니다. 지수 비트 23부터 비트 30까지 8비트에는 지수가 들어갑니다. 지수는 2n의 n에 127을 더한 값으로 2진수로 표현됩니다. 이 형식은 단정밀도 부동소수점 데이터에 관한 IEEE754 규격을 따르며, 부동소수점 데이터를 변환하거나 연산하는 명령어에서만 사용됩니다. CX-Programmer의 I/O 메모리 편집 · 모니터 화면에서 설정하거나 모니터할 수 있습니다. 따라서 사용자는 이 형식을 알 필요는 없지만, 이 형식이 2워드를 차지한다는 점은 알아 두어야 합니다.	---	---
배정밀도 부동소수점 10진수	 값 = (−1)부호 × 1.[가수] × 2지수 부호 (비트 63) 1: 음수 또는 0: 양수 가수 비트 00부터 비트 51까지 52비트에는 가수, 즉 1.□□□□.....에서 소수점 아래 부분이 2진수로 들어갑니다. 지수 비트 52부터 비트 62까지 11비트에는 지수가 들어갑니다 지수는 2n의 n에 1023을 더한 값으로 2진수로 표현됩니다. 이 형식은 배정밀도 부동소수점 데이터에 관한 IEEE 754 규격을 따르며, 부동소수점 데이터를 변환하거나 연산하는 명령어에서만 사용됩니다. CX-Programmer의 I/O 메모리 편집 · 모니터 화면에서 설정하거나 모니터할 수 있습니다. 따라서 사용자는 이 형식을 알 필요는 없지만, 이 형식이 4워드를 차지한다는 점은 알아 두어야 합니다.	---	---

주 부동소수점 데이터는 근사 표현이므로, 데이터와 실제 값 사이에 오차가 있을 수 있습니다. 또한 값의 유효 범위에도 제한이 있습니다. 따라서 산술 연산에서 생기는 불일치는 반올림, 오버플로, 언더플로 처리로 처리됩니다.

사용상의 주의

일반적으로 프로세서 등의 하드웨어가 다르면 실수형의 연산 결과가 달라질 수 있습니다. 기종이 다른 CPU 유닛 간에 프로그램이나 라이브러리를 재사용할 때에는 실수형의 연산 결과를 충분히 확인하십시오.

참고

· 보수
일반적으로 x 진법의 보수란, 주어진 수의 모든 자리를 x-1에서 뺀 다음 최하위 자리에 1을 더해 얻는 수를 말합니다. (예: 7556의 10의 보수는 9999 − 7556 + 1 = 2444입니다.) 보수는 뺄셈 등의 연산을 덧셈으로 표현하는 데 사용됩니다.

예: 8954 − 7556 = 1398이고, 8954 + (7556의 10의 보수) = 8954 + 2444 = 11398입니다. 최상위 비트를 무시하면 뺄셈 결과 1398을 얻습니다.

· 2의 보수
2의 보수는 2 진법의 보수입니다. 여기서는 모든 자리를 1(2 − 1 = 1)에서 빼고 1을 더합니다.

예: 2진수 1101의 2의 보수는 1111(F Hex) − 1101(D Hex) + 1 (1 Hex) = 0011(3 Hex)입니다. 이 값을 4자리 16진수로 표현하면 다음과 같습니다.
a Hex의 2의 보수 b Hex는 FFFF Hex − a Hex + 0001 Hex = b Hex입니다. "a Hex"의 2의 보수 b Hex를 구하려면 b Hex = 10000 Hex − a Hex를 사용합니다.

예: 3039 Hex의 2의 보수를 구하려면 10000 Hex − 3039 Hex = CFC7 Hex를 사용합니다.
마찬가지로 2의 보수 b Hex에서 값 a Hex를 구하려면 a Hex = 10000 Hex − b Hex를 사용합니다.

예: 2의 보수 CFC7 Hex에서 실제 값을 구하려면 10000 Hex
− CFC7 Hex = 3039 Hex를 사용합니다.
CJ 시리즈에는 참값에서 2의 보수를 구하거나 2의 보수에서 참값을 구하는 데 사용할 수 있는 NEG(160)(2의 보수)과 NEGL(161)(2의 보수(더블))의 두 가지 명령어가 있습니다.

AI 번역본
-For Reference Only-

● 1워드 데이터로 표현되는 값

값 (10진수)	2진 표현			BCD 표현 (10진수)
	10진 표현		16진 표현	
	부호 없음	부호 있음		
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	해당 없음.
	:	:	:	
32767	&32767	+32767	#7FFF	
32768	&32768	해당 없음.	#8000	
:	:		:	
65535	&65535		#FFFF	
–1	해당 없음.	–1	#FFFF	해당 없음.
:		:	:	
–32768		–32768	#8000	
–32769		해당 없음.	해당 없음.	

AI 번역본
-For Reference Only-

5-6-4 I/O 리프레시 타이밍

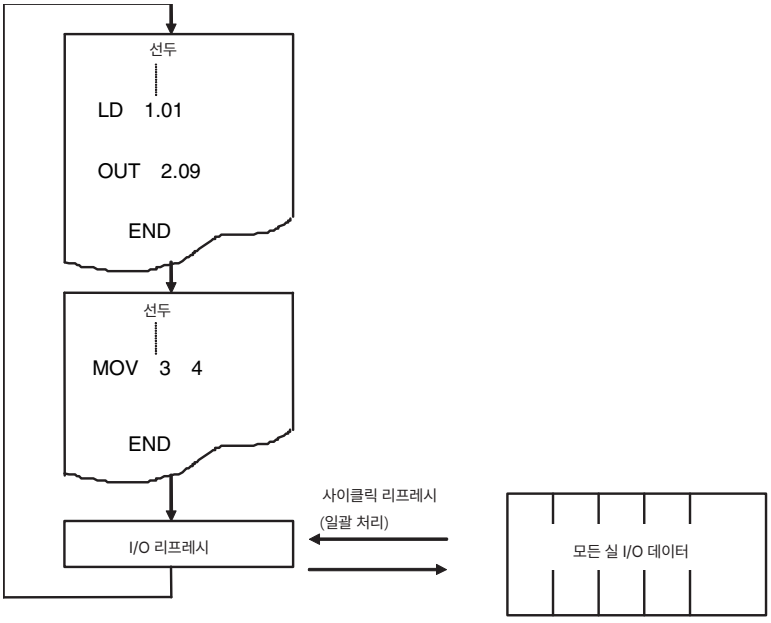
외부 I/O를 리프레시하는 방법으로는 다음이 사용됩니다.

・사이클릭 리프레시

・즉시 리프레시(! 지정이 붙은 명령어, IORF(097), FIORF(225), DLNK(226))

사이클릭 리프레시

사이클릭 리프레시에서는 READY 상태인 모든 사이클릭 태스크의 실행이 완료된 후 I/O 리프레시가 일괄 수행됩니다.



태스크 안에서 I/O 리프레시가 필요한 경우에는 END(001) 명령어 앞에서 필요한 모든 워드에 대해 IORF(097) 명령어를 실행하십시오.

AI 번역본
-For Reference Only-

즉시 리프레시

즉시 리프레시에서는 프로그램에서 명령어가 실행될 때마다 리프레시가 수행됩니다.

● 즉시 리프레시의 종류

즉시 리프레시를 수행하는 방법에는 두 가지가 있습니다. 명령어에 즉시 리프레시 변형(!)을 지정하는 방법과, 즉시 리프레시 전용 명령어를 사용하는 방법입니다.

리프레시 변형(!)이 붙은 명령어

명령어의 즉시 리프레시 변형(!)을 사용하면 명령어가 실행될 때(즉, 사이클 도중에) 리프레시를 수행할 수 있습니다.

즉시 리프레시 전용 명령어

메모리를 리프레시할 유닛에 따라 다음 세 가지 명령어를 사용할 수 있습니다.

IORF(097) 명령어: 기본 I/O 유닛 및 특수 I/O 유닛(CIO 영역에 할당된 워드만)

FIORF(225) 명령어: 특수 I/O 유닛

DLNK(226) 명령어: CPU 버스 유닛

어느 명령어로 어느 유닛을 리프레시할 수 있는지는 다음 표와 같습니다.

유닛	즉시 리프레시 변형(!)이 붙은 명령어	IORF(097) 명령어	FIORF(225) 명령어	DLNK(226) 명령어
기본 I/O 유닛	가능	가능	불가	불가
CIO 영역에 할당된 워드				
특수 I/O 유닛	불가	가능*2	가능*2	불가
CIO 영역에 할당된 워드				
DM 영역에 할당된 워드		불가		
CPU 버스 유닛	불가	불가	불가	가능
CIO 영역에 할당된 워드				
DM 영역에 할당된 워드				
유닛 고유 메모리*1				
CJ2M/CJ3M CPU 유닛용 펄스 I/O 모듈	가능*3	불가	불가	불가
CIO 영역에 할당된 워드				

*1 EtherNet/IP 유닛, Controller Link 유닛, SYSMAC LINK 유닛의 데이터 링크, DeviceNet 리모트 I/O 통신.

*2 특수 I/O 유닛에 CIO 영역으로 할당된 워드는 IORF(097) 명령어 또는 FIORF(225) 명령어 중 어느 쪽으로도 리프레시할 수 있습니다. 명령어 실행 시간은 IORF(097) 명령어보다 FIORF(225) 명령어 쪽이 짧습니다. 또한 오퍼랜드에 특수 I/O 유닛의 유닛 번호를 지정하기만 하면 되므로 사용하기도 쉽습니다.

*3 즉시 리프레시는 통상 입력 또는 퀵 리스폰스 입력으로 설정된 입력과 통상 출력으로 설정된 출력에서만 가능합니다.

사용상의 주의

즉시 리프레시 변형의 실행 시간은 명령어의 통상 변형보다 길어지므로, 사이클 타임이 길어진다는 점에 주의하십시오.

AI 번역본
For Reference Only

● 리프레시 변형(!)이 붙은 명령어

즉시 리프레시를 지정하려면 명령어 앞에 느낌표(!)를 붙입니다.

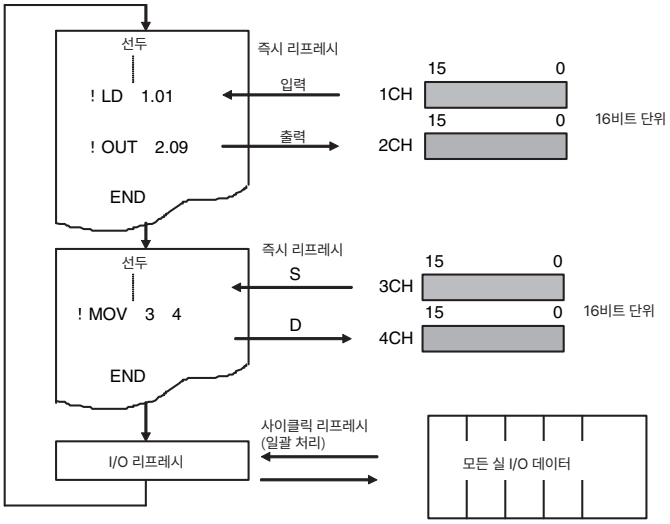
・오퍼랜드에 실 I/O 비트를 지정하면 명령어가 실행될 때 아래와 같이 I/O가 리프레시됩니다.

유닛	리프레시되는 데이터
기본 I/O 유닛	해당 비트를 포함하는 16비트에 대해 I/O가 리프레시됩니다.

・명령어에 워드 오퍼랜드를 지정하면 지정한 16비트에 대해 I/O가 리프레시됩니다.

・입력 또는 소스 오퍼랜드의 입력은 명령어가 실행되기 직전에 리프레시됩니다.

・출력 또는 저장처(D) 오퍼랜드의 출력은 명령어가 실행된 직후에 리프레시됩니다.



● IORF(097): I/O 리프레시

IORF(097)을 사용하면 사이클 도중에 기본 I/O 유닛의 실 I/O 데이터와 특수 I/O 유닛에 할당된 CIO 영역 워드의 데이터를 모두 리프레시할 수 있습니다.

IORF	D1: 선두 워드
D1	D2: 최종 워드
D2	IORF(097)이 실행되면 D1부터 D2까지 모든 워드의 데이터가 리프레시됩니다.

예:

IORF	IORF(097)이 실행되면 CIO 10부터 CIO 16까지 7워드가 리프레시됩니다.
10	
16	

기본 I/O 유닛의 입력 데이터를 사용하는 연산이나 기본 I/O 유닛으로 데이터를 출력하는 연산에 고속 응답이 필요한 경우에는, 연산 명령어의 직전과 직후에 IORF(097)을 사용하십시오.

사용상의 주의

IORF(097)은 명령어 실행 시간이 비교적 길고, 그 실행 시간은 리프레시하는 워드 수에 비례해 늘어나므로 사이클 타임을 크게 늘릴 수 있습니다. 사이클 타임이 너무 길어지지 않도록 주의하십시오.

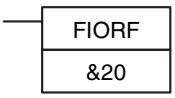
● FIORF(225): 특수 I/O 유닛 I/O 리프레시

FIORF(225)를 사용하면 지정한 유닛 번호의 특수 I/O 유닛에서 다음 데이터를 필요할 때만 리프레시할 수 있습니다.



N: 특수 I/O 유닛의 유닛 번호

예:



유닛 번호 &20인 특수 I/O 유닛에 할당된 CIO 워드와 DM 워드를 리프레시합니다.

PLC 설정에서 특수 I/O 유닛의 통상 사이클릭 I/O 리프레시를 무효화하고(해당 유닛의 특수 I/O 유닛 사이클릭 리프레시 무효 비트를 ON), FIORF(225)를 실행하여 필요할 때만 그 유닛의 I/O 리프레시를 수행할 수 있습니다. 이 기능을 사용하면 PLC에 특수 I/O 유닛을 접속했을 때 PLC의 사이클 타임이 늘어나는 것을 방지할 수 있습니다.

● DLNK(226) 사용: CPU 버스 유닛 I/O 리프레시

DLNK(226)을 사용하면 CIO 영역과 DM 영역에서 CPU 버스 유닛에 할당된 워드와 CPU 버스 유닛 고유의 데이터에 대한 리프레시 빈도를 높일 수 있습니다.*1

*1 CPU 버스 유닛 고유의 데이터에는 Controller Link 유닛이나 SYSMAC LINK 유닛의 데이터 링크, DeviceNet 유닛의 리모트 I/O 등이 있습니다.



N: CPU 버스 유닛의 유닛 번호

예:



왼쪽 명령어는 유닛 번호 1인 CPU 버스 유닛*1에 대해 CIO 영역과 DM 영역에 할당된 워드와 그 CPU 버스 유닛 고유의 데이터를 리프레시합니다.

*1 유닛 번호 1인 CPU 버스 유닛이 Controller Link 유닛이면 데이터 링크가 리프레시됩니다.

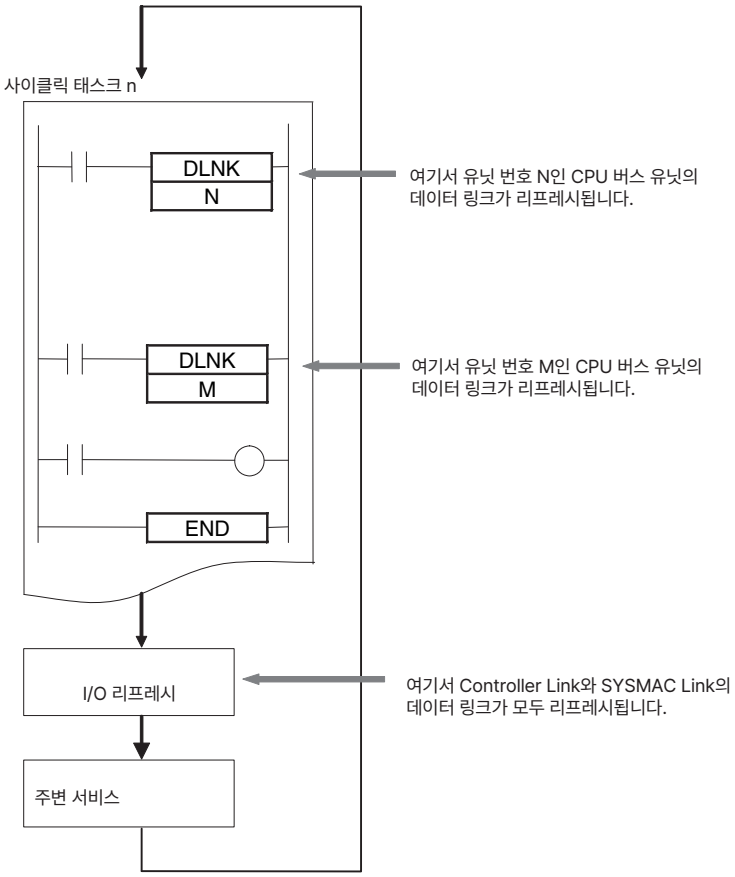
적용 예: 사이클 타임이 길면 Controller Link 데이터 링크의 리프레시 간격이 매우 길어질 수 있습니다. Controller Link 유닛에 대해 DLNK(226)을 실행하여 데이터 링크 리프레시 빈도를 높이면 이 간격을 줄일 수 있습니다.

AI 번역본
-For Reference Only-

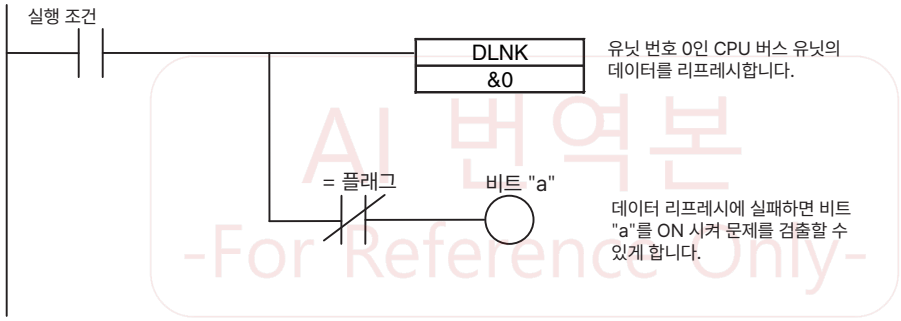
참고

DLNK(226) 명령어의 적용 예를 다음 그림에 나타냅니다.

예: 데이터 링크



데이터를 리프레시하는 중인 CPU 버스 유닛에 대해 DLNK(226)을 실행하면 데이터는 리프레시되지 않고 이퀄 플래그가 OFF 됩니다. 통상적으로는 리프레시가 정상적으로 완료되었는지 확인할 수 있도록 아래와 같이 이퀄 플래그를 프로그래밍하십시오.



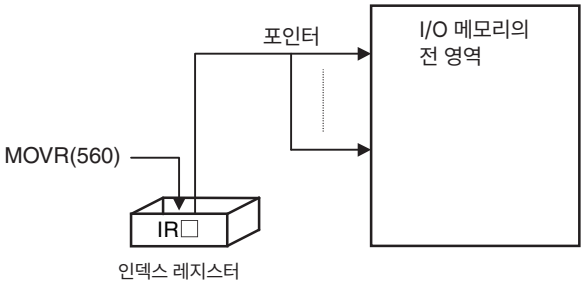
AI 번역본
-For Reference Only-

5-7 인덱스 레지스터

5-7-1 인덱스 레지스터란?

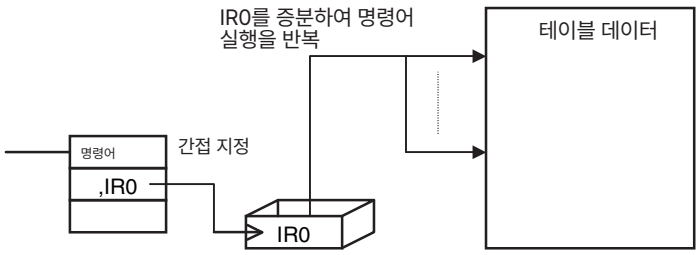
인덱스 레지스터는 PLC 메모리 어드레스를 지정하는 포인터로 기능하며, 이 PLC 메모리 어드레스는 I/O 메모리의 절대 메모리 어드레스입니다. MOVR(560) 또는 MOVRW(561)로 PLC 메모리 어드레스를 인덱스 레지스터에 저장한 후, 다른 명령어의 오퍼랜드로 그 인덱스 레지스터를 입력하면 저장된 PLC 메모리 어드레스를 간접 지정할 수 있습니다.

인덱스 레지스터의 장점은 타이머와 카운터의 현재값을 포함하여 I/O 메모리의 임의의 비트나 워드를 지정할 수 있다는 점입니다.



5-7-2 인덱스 레지스터 사용법

인덱스 레지스터는 FOR-NEXT 루프와 같은 루프와 조합하면 강력한 도구가 됩니다. 인덱스 레지스터의 내용은 증분 · 감분 · 오프셋을 매우 쉽게 할 수 있으므로, 루프 안의 몇 개 명령어만으로 연속된 데이터 테이블을 매우 효율적으로 처리할 수 있습니다.



기본적으로 인덱스 레지스터는 다음 순서로 사용합니다.

1 MOVR(560)으로 원하는 비트 또는 워드의 PLC 메모리 어드레스를 인덱스 레지스터에 저장합니다.

2 거의 모든 명령어에서 인덱스 레지스터를 오퍼랜드로 지정하여 원하는 비트 또는 워드를 간접 지정합니다.

3 원래의 PLC 메모리 어드레스를 오프셋하거나 증분(아래 참조)하여 포인터를 다른 어드레스로 전환합니다.

4 순서 2와 3을 반복하여 임의 개수의 어드레스에 대해 명령어를 실행합니다.

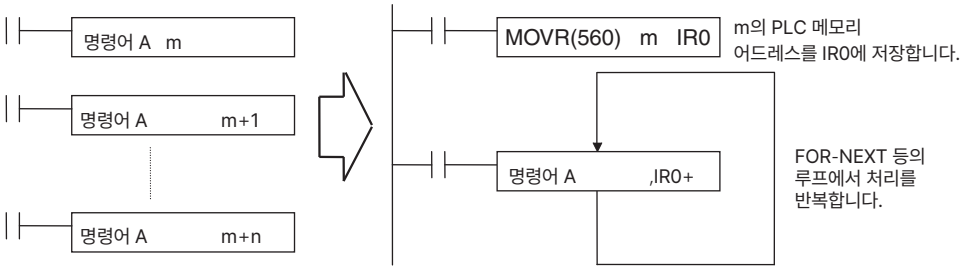
주 인덱스 레지스터를 오프셋하거나 증분하려면 다음 방법 중 하나를 사용하십시오.

· 인덱스 레지스터의 간접 지정:
오토 인크리먼트 부가 간접 지정(,IR□+ 또는 ,IR□++), 오토 디크리먼트 부가 간접 지정(,−IR□ 또는 ,−−IR□), 정수 오프셋 부가 간접 참조(constant,IR□), DR 오프셋 부가 간접 지정(DR□,IR□)

AI 번역본
-For Reference Only-

· 인덱스 레지스터의 직접 지정 명령어:
BIN 가산(+L), BIN 감산(−L), BIN 더블 인크리먼트(++L), BIN 더블 디크리먼트(−−L)

예:



위에 제시한 예는 프로그램 루프 안의 인덱스 레지스터가 긴 명령어 나열을 어떻게 대체할 수 있는지를 보여 줍니다. 이 경우 명령어 A가 n+1회 반복되어 값의 테이블을 읽고 비교하는 등의 처리를 수행합니다.

사용상의 주의

다음 명령어는 입력 조건이 OFF일 때에도 실행됩니다. 따라서 이들 명령어의 오퍼랜드에 오토 인크리먼트 또는 오토 디크리먼트(,IR+ 또는 ,IR-)를 사용하여 간접 메모리 어드레스를 지정하면, 인덱스 레지스터(IR)의 값이 입력 조건과 관계없이 매 사이클 갱신됩니다(매 사이클 1씩 증가 또는 감소).
프로그램을 작성할 때 이 점을 고려해야 합니다.

분류	명령어
시퀀스 입력 명령어	LD, LD NOT, AND, AND NOT, OR, OR NOT, LD TST(350), LD TSTN(351), AND TST(350), AND TSTN(351), OR TST(350), OR TSTN(351)
시퀀스 출력 명령어	OUT, OUT NOT, DIFU(013), DIFD(014)
시퀀스 제어 명령어	JMP(004), FOR(512)
타이머 · 카운터 명령어	TIM/TIMX(550), TIMH(015)/TIMHX(551), TMHH(540)/TMHHX(552), TIMU(541)/TIMUX(556), TMUH(544)/TMUHX(557), TTIM(087)/TTIMX(555), TIML(542)/TIMLX(553), MTIM(533)/MTIMX(554), CNT/CNTX(546), CNTR(012)/CNTRX(548)
비교 명령어	심볼 비교 명령어(LD, AND, OR = 등(펑션 코드: 300, 305, 310, 320, 325))
단정밀도 부동소수점 연산 명령어	단정밀도 부동소수점 데이터 비교(LD, AND, OR = F 등(펑션 코드: 329~334))
배정밀도 부동소수점 연산 명령어	배정밀도 부동소수점 데이터 비교(LD, AND, OR = D 등(펑션 코드: 335~340))
블록 프로그래밍 명령어	BPPS(811), BPRS(812), EXIT(806), EXIT(806) NOT, IF(802), IF(802) NOT, WAIT(805), WAIT(805) NOT, TIMW(813)/TIMWX(816), CNTW(814)/CNTWX(818), TMHW(815)/TMHWX(817), LEND(810), LEND(810) NOT
문자열 처리 명령어	문자열 비교(LD, AND, OR = $ 등(펑션 코드: 670~675))

AI 사용불
-For Reference Only-

다음 래더 프로그래밍 예는 인덱스 레지스터가 어떻게 취급되는지를 보여 줍니다.

예 1

래더 프로그램:
LD P_Off
OUT, IR0+

동작: PLC 메모리 어드레스 CIO 0.13이 IR0에 저장되어 있는 경우.

입력 조건이 OFF이므로(P_Off는 상시 OFF 플래그), OUT 명령어는 IR0로 간접 지정된 CIO 0.13을 OFF로 합니다. OUT 명령어가 실행되므로 IR0는 증분됩니다. 그 결과 IR0에서 +1 증분된 PLC 메모리 어드레스 CIO 0.14가 저장됩니다. 따라서 다음 사이클에서 OUT 명령어는 CIO 0.14를 OFF로 합니다.

예 2

래더 프로그램:

```
LD P_Off
SET, IR0+
```

동작: PLC 메모리 어드레스 CIO 0.13이 IR0에 저장되어 있는 경우.

입력 조건이 OFF이므로(P_Off는 상시 OFF 플래그) SET 명령어는 실행되지 않습니다. 따라서 IR0는 증분되지 않고 IR0에 저장된 값은 PLC 메모리 어드레스 CIO 0.13 그대로 유지됩니다.

· 다음 명령어는 인터록이 유효한 경우에도 실행됩니다. 따라서 이들 명령어의 오퍼랜드에 오토 인크리먼트 또는 오토 디크리먼트(,IR+ 또는 ,IR-)를 사용하여 간접 메모리 어드레스를 지정하면, 인덱스 레지스터(IR)의 값이 입력 조건과 관계없이 매 사이클 갱신됩니다(매 사이클 1씩 증가 또는 감소). 프로그램을 작성할 때 이 점을 고려해야 합니다.

명령어 그룹	명령어
시퀀스 출력 명령어	OUT, OUTB(534), OUT, NOT(520)
타이머 · 카운터 명령어	TIM/TIMX(550), TIMH(015)/TIMHX(551), TMHH(540)/TMHHX(552), TIMU(541)/TIMUX(556), TMUH(544)/TMUHX(557), TIML(542)/TIMLX(553)

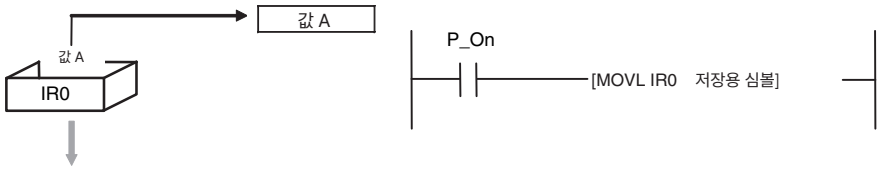
AI 번역본
-For Reference Only-

사용상의 주의

・인덱스 레지스터 IR0~IR15를 펑션 블록 안에서 사용할 때, 동일한 인덱스 레지스터를 다른 펑션 블록 안이나 펑션 블록 외부의 프로그램에서 사용하면 두 인스턴스 사이에 경합이 발생하여 프로그램이 정상적으로 실행되지 않습니다. 따라서 인덱스 레지스터(IR0~IR15)를 사용할 때에는 펑션 블록이 시작되는 시점(또는 인덱스 레지스터를 사용하기 전)에서 반드시 인덱스 레지스터의 값을 저장하고, 펑션 블록이 완료된 시점(또는 인덱스 레지스터를 사용한 후)에서 인덱스 레지스터를 저장된 값으로 되돌리는 처리를 프로그램에 포함시키십시오.

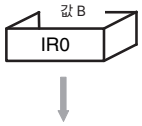
예: 펑션 블록 시작 시(또는 인덱스 레지스터 사용 전):

1. IR의 값을 저장합니다(예: A).



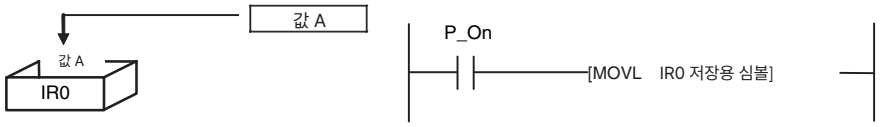
펑션 블록 내:

2. IR을 사용합니다.



펑션 블록 시작 시(또는 인덱스 레지스터 사용 전):

3. IR을 저장된 값으로 되돌립니다(예: A).



・인덱스 레지스터는 사용하기 전에 반드시 값을 설정하십시오. 값을 설정하지 않은 채 인덱스 레지스터를 사용하면 동작이 안정되지 않습니다.

사용상의 주의

인덱스 레지스터에 의한 레지스터 간접 지정 시의 주의

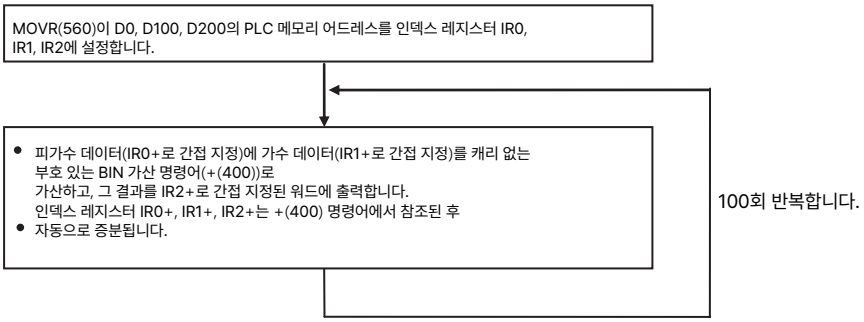
I/O 메모리 외의 영역이나 시스템이 예약한 영역에 액세스하기 위해 레지스터 간접 지정을 사용하지 마십시오. 액세스를 시도하면 부정 액세스 에러가 발생합니다.

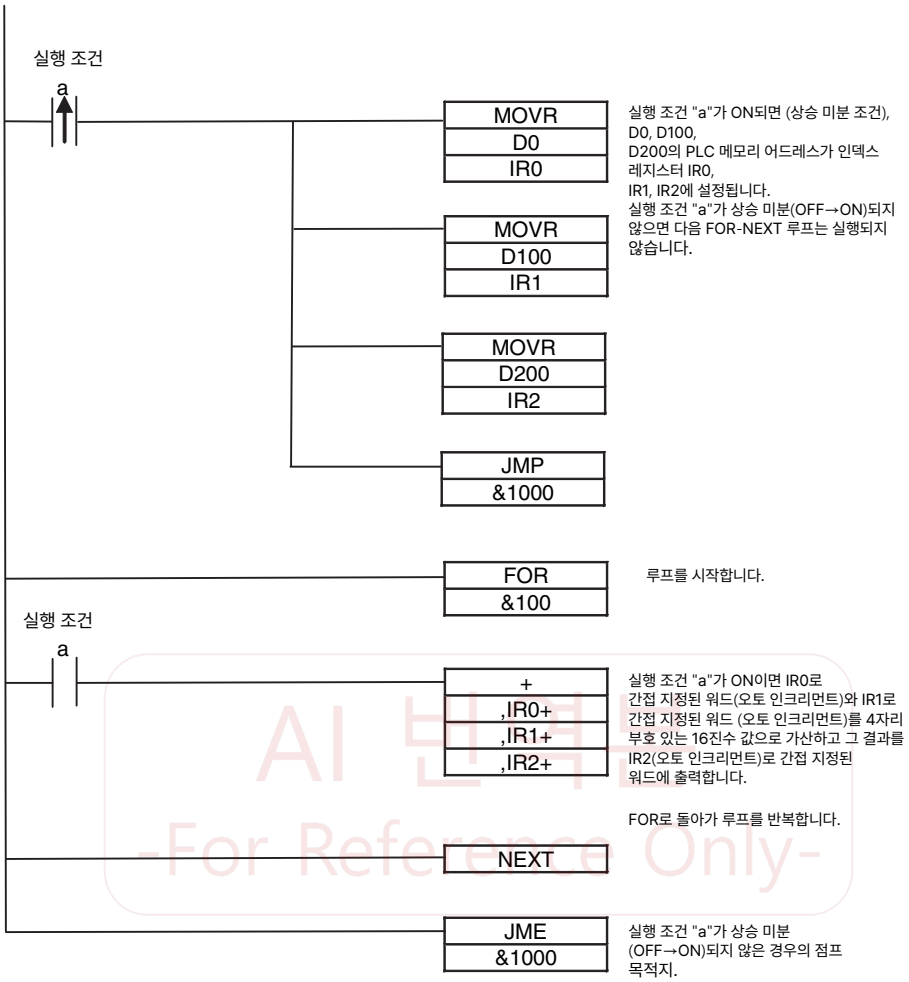
PLC 메모리 어드레스의 한계에 대한 자세한 내용은 A-4 PLC 메모리 어드레스의 메모리 맵을 참조하십시오.

AI 번역본
-For Reference Only-

인덱스 레지스터의 응용 예

D0~D99의 데이터(피가수 데이터)에 D100~D199의 데이터(가수 데이터)를 가산하고 그 가산 결과를 D200~D299에 출력합니다. 하나의 가산 명령어의 오퍼랜드를 인덱스 레지스터로 지정하고, 인덱스 레지스터를 증분하면서 가산 명령어를 반복 실행함으로써 가산 연산을 수행합니다.





AI 번역
-For Reference Only-

참고

인덱스 레지스터를 직접 지정할 수 있는 명령어는 다음 표에 나타낸 것뿐입니다.

● 인덱스 레지스터의 직접 지정

인덱스 레지스터의 크기는 인덱스 레지스터 IR0~IR15에 대해 레지스터당 2워드이므로 더블 워드 명령어(니모닉에 "L"이 붙은 것)를 사용하십시오.

명령어 그룹	명령어 이름	니모닉	주요 기능
데이터 전송 명령어	레지스터로 전송	MOVR(560)	비트 또는 워드의 PLC 메모리 어드레스를 인덱스 레지스터에 저장합니다.
	타이머/카운터 현재값을 레지스터로 전송	MOVRW(561)	
테이블 데이터 처리 명령어	레코드 위치 설정	SETR(635)	
	레코드 번호 취득	GETR(636)	인덱스 레지스터에 저장된 PLC 메모리 어드레스를 출력합니다.
트래킹 명령어	레코드 검색 명령어	RSRCH□ (펑션 코드: 360~364, 370~374, 380~384)	일치한 레코드의 선두 PLC 메모리 어드레스를 인덱스 레지스터에 출력합니다.
	레코드 정렬 명령어	RSORT□ (펑션 코드: 203 205)	정렬된 범위의 마지막 어드레스의 PLC 메모리 어드레스보다 1 큰 값을 인덱스 레지스터에 출력합니다.
데이터 전송 명령어	더블 전송	MOVL(498)	인덱스 레지스터 간에 전송합니다. 교환과 비교에 사용합니다.
	더블 데이터 교환	XCGL(562)	
비교 명령어	더블 등가	=L(301)	
	더블 부등가	< >L(306)	
	더블 미만	< L(311)	
	더블 이하	< =L(316)	
	더블 초과	>L(321)	
	더블 이상	>=L(326)	
	더블 비교	CMPL(060)	
인크리먼트/디크리먼트 명령어	BIN 더블 인크리먼트	++L(591)	인덱스 레지스터의 내용을 증분 · 감분 · 오프셋하여 레지스터 안의 PLC 메모리 어드레스를 변경합니다.
	BIN 더블 디크리먼트	—L(593)	
심볼 연산 명령어	캐리 없는 부호 있는 BIN 가산(더블)	+L(401)	
	캐리 없는 부호 있는 BIN 감산(더블)	–L(411)	
특수 명령어	CV로부터의 어드레스 변환	FRMCV(284)	CVM1/CV 시리즈와 CJ 시리즈 어드레스 간에 실제 PLC 메모리 어드레스를 변환합니다.
	CV로의 어드레스 변환	TOCV(285)	

삼성 번역본 -For Reference Only

5-7-3 인덱스 레지스터 관련 처리

CJ 시리즈 CPU 유닛의 테이블 데이터 처리 명령어는 인덱스 레지스터의 기능을 보완합니다. 테이블 데이터 처리 명령어는 크게 스택 처리 명령어와 테이블 처리 명령어로 나눌 수 있습니다.

처리			목적	명령어
스택 처리			FIFO(선입선출) 또는 LIFO(후입선출) 데이터 테이블을 조작하고, 데이터 테이블의 데이터 엔트리를 읽기 · 쓰기 · 삽입 · 삭제하거나 개수를 셉니다.	SSET(630), PUSH(632), FIFO(633), LIFO(634), SREAD(639), SWRIT(640), SINS(641), SDEL(642), SNUM(638)
테이블 처리	1워드 레코드 테이블 (범위 명령어)	기본 처리	최댓값, 최솟값, 체크섬 등의 정수 또는 부동소수점 소수 값을 구합니다.	MIN(183), MAX(182), MINL(175), MAXL(174), MINF(177), MAXDF(176), MIND(179), MAXD(178), SRCH(181), SUM(184), FCS(180)
		특수 처리	비교나 정렬 등 그 밖의 다양한 테이블 처리를 수행합니다.	인덱스 레지스터를 SRCH(181), MAX(182), MIN(183) 및 비교 명령어 등과 조합합니다.
	멀티 워드 레코드 테이블		사용자가 등록한 지정 영역(워드 수와 범위) 안의 레코드를 검색하고 정렬합니다.	RSRCH <(360), RSRCH <=(361), RSRCH =,(362), RSRCH >=(363), RSRCH >(364), RSRCH2 <(370), RSRCH2 <=(371), RSRCH2 =(372), RSRCH2 >=(373), RSRCH2 >(374), RSRCH4 <(380), RSRCH4 <=(381), RSRCH4 =(382), RSRCH4 >=(383), RSRCH4 >(384), RSORT(204), RSORT2(204), RSORT4(205)
			사용자가 등록한 지정 영역(워드 수와 범위) 안의 레코드에 대해 연산을 수행합니다.	다음 명령어를 인덱스 레지스터와 조합합니다. · DIM(631), SETR(635), GETR(636) 명령어 · 그 밖의 명령어(예: 비교 명령어)

AI 번역본
-For Reference Only-

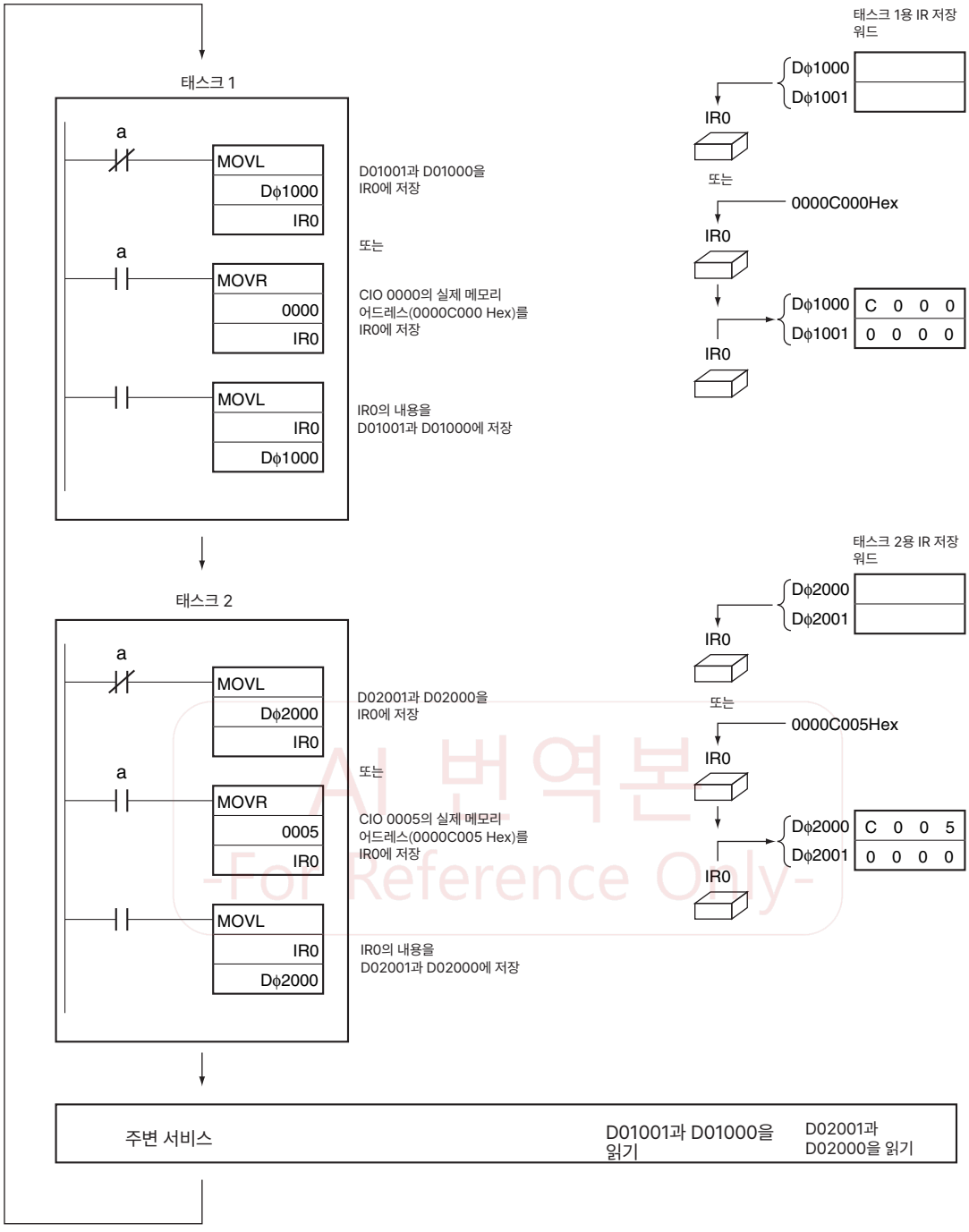
5-7-4 인덱스 레지스터 모니터링

인덱스 레지스터는 다음과 같이 모니터링할 수 있습니다.

- CX-Programmer로 각 태스크의 최종 인덱스 레지스터 값을 모니터링합니다.
- Host Link 커맨드나 FINS 커맨드로 인덱스 레지스터 값을 모니터링하려면, 각 태스크의 끝에서 인덱스 레지스터 값을 다른 영역(예: DM 영역)에 저장하고 각 태스크의 시작에서 그 저장 워드 (예: DM 영역)로부터 인덱스 레지스터 값을 읽어 들이는 프로그램을 작성하십시오. 다른 영역 (예: DM 영역)에 태스크별로 저장된 값은 CX-Programmer, Host Link 커맨드 또는 FINS 커맨드로 편집할 수 있습니다.

● 예

주 인덱스 레지스터에는 반드시 PLC 메모리 어드레스를 사용하십시오.



AI 번역본
-For Reference Only-

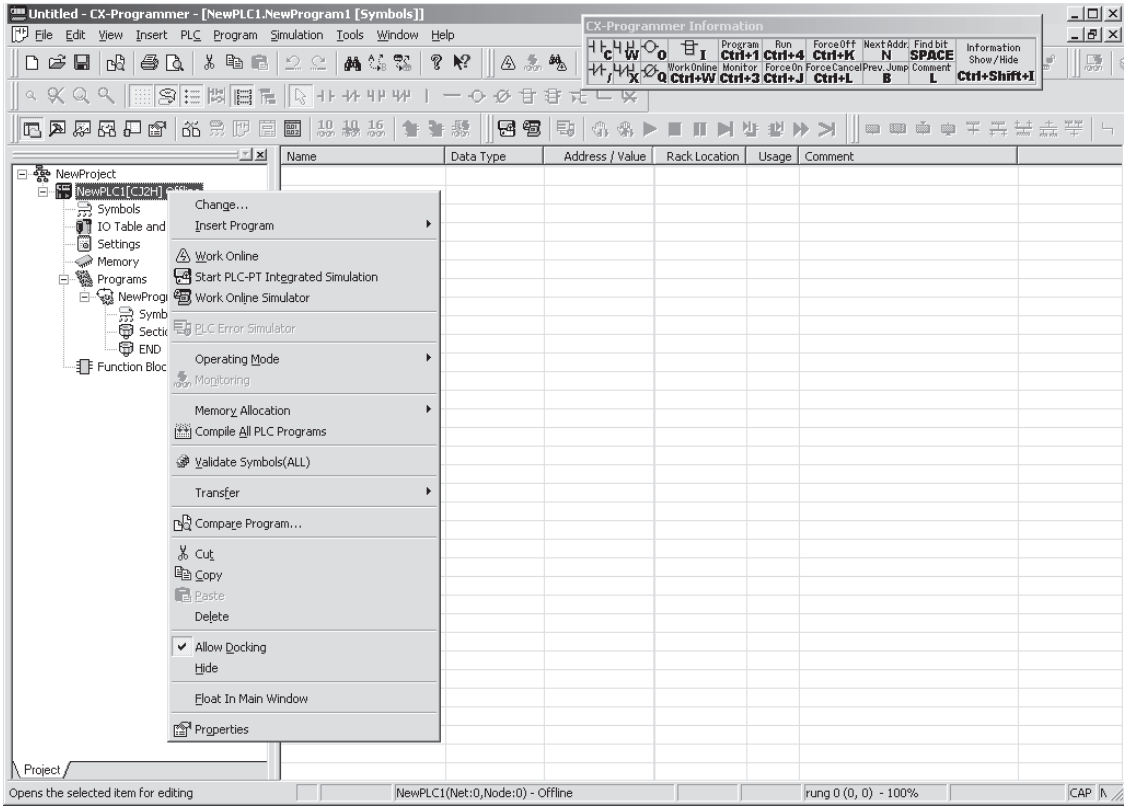
5-7-5 태스크 간 인덱스 레지스터와 데이터 레지스터 공유

인덱스 레지스터와 데이터 레지스터(IR/DR)는 태스크 간에 공유할 수 있습니다. 통상 설정은 태스크마다 별도의 레지스터를 사용하는 것입니다. 현재 설정은 A99.14에서 확인할 수 있습니다.

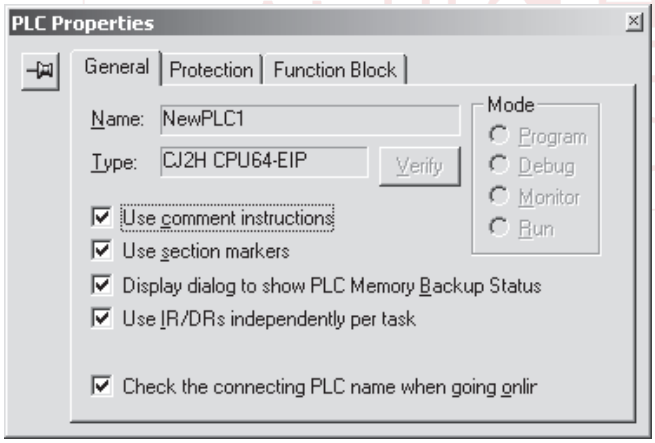
● 설정 방법

CX-Programmer를 사용하여 인덱스 레지스터와 데이터 레지스터의 공유를 설정합니다. 이 설정은 프로그래밍 콘솔에서는 할 수 없습니다.

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



2 Properties를 선택합니다. 다음 PLC Properties 대화 상자가 표시됩니다.



태스크마다 별도의 인덱스 레지스터와 데이터 레지스터가 필요한 경우에는 IR/DR을 태스크별로 독립 사용하는 체크 표시를 그대로 두십시오. 모든 태스크에서 인덱스 레지스터와 데이터 레지스터를 공유하려면 체크 표시를 해제하십시오.

● 보조 기억 영역 플래그와 워드

이름	어드레스	설명
태스크 간 IR/DR 동작	A99.14	인덱스 레지스터와 데이터 레지스터를 태스크 간에 공유할지 여부를 나타냅니다. 0: 태스크마다 별도의 레지스터(기본값) 1: 모든 태스크에서 레지스터 공유



참고

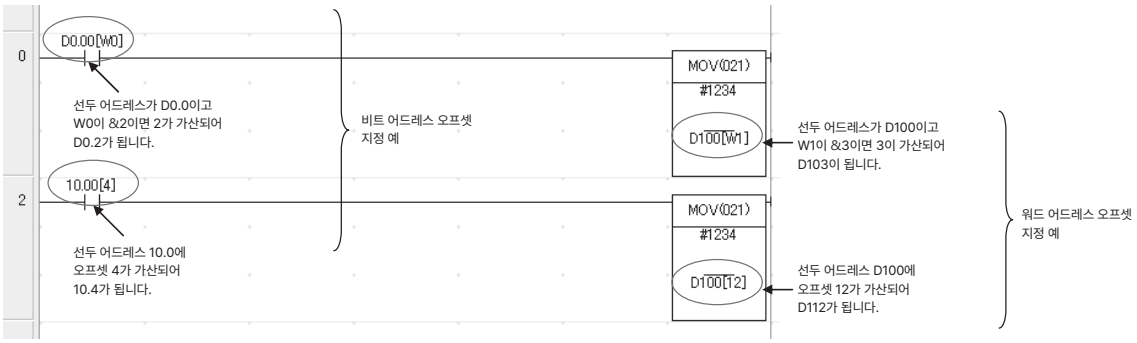
- 둘 이상의 태스크에서 동일한 내용이 필요한 경우, 인덱스 레지스터와 데이터 레지스터를 공유하면 태스크 간에 레지스터 내용을 저장 · 로드할 필요가 없어집니다. 인덱스 레지스터 내용의 저장 · 로드에 대해서는 6-20 인덱스 레지스터를 참조하십시오.
- 인덱스 레지스터와 데이터 레지스터를 공유하면 태스크 간 전환 시간이 다소 빨라집니다. 레지스터를 사용하지 않는 경우나 태스크마다 별도의 레지스터가 특별히 필요하지 않은 경우에는 레지스터 공유 설정을 권장합니다.

AI 번역본
-For Reference Only-

5-8 어드레스 오프셋 지정

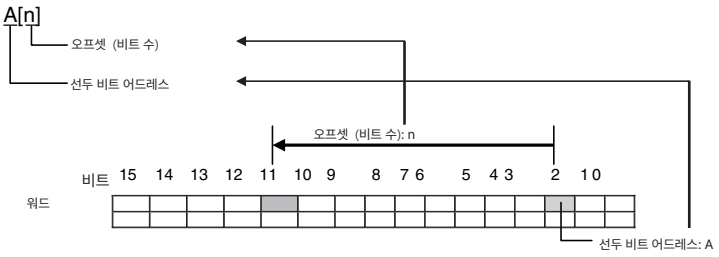
5-8-1 개요

명령어의 오퍼랜드에 어드레스를 지정할 때, 지정한 어드레스 뒤의 대괄호 안에 오프셋을 지정하면 지정된 어드레스를 변경할 수 있습니다.



● 비트 어드레스

비트 어드레스는 A(선두 비트 어드레스)로부터 n(비트 수)으로 지정한 양만큼 오프셋됩니다.



선두 비트 어드레스

선두 비트 어드레스는 비트 어드레스 또는 배열 변수(STRING 및 NUMBER 데이터형 제외)로 지정할 수 있습니다. 오프셋은 CIO, H, W, DM, EM 영역의 어드레스에 대해서만 가능합니다. 그 밖의 영역의 어드레스를 지정하면 동작이 보장되지 않습니다. I/O 코멘트는 이 선두 비트 어드레스의 I/O 코멘트를 나타냅니다.

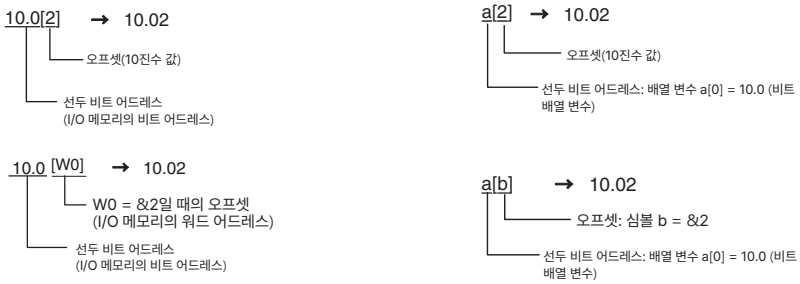
오프셋

오프셋은 10진 정수, 워드 어드레스(단, CIO 영역의 어드레스는 지정할 수 없습니다) 또는 1워드 심볼(즉, INT, UINT, WORD, CHANNEL 데이터형의 심볼)로 지정할 수 있습니다. 워드 어드레스를 지정하면 지정한 워드의 내용이 오프셋으로 사용됩니다.

오프셋이 지정한 워드의 비트 15를 넘으면 다음 워드의 비트 00부터 오프셋이 계속됩니다.
오프셋을 간접 지정하는 경우에는 입력 비교 명령어 등을 사용하여 최종 비트 어드레스가 상한을 넘지 않도록 하십시오.

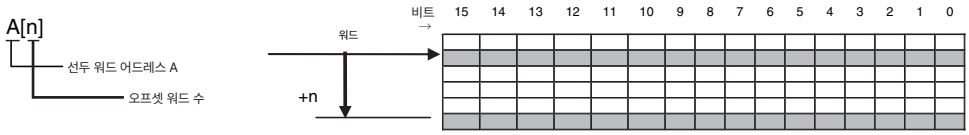
오프셋 비트 수가 선두 비트 어드레스의 메모리 영역을 넘으면, 최종 비트 어드레스는 실제 PLC 메모리 어드레스로 결정되는 순서에 따라 다음 메모리 영역에 위치합니다. (메모리 맵의 영역 배치에 대해서는 A-4 PLC 메모리 어드레스의 메모리 맵을 참조하십시오.

예:



● 워드 어드레스

워드 어드레스는 A(선두 워드 어드레스)로부터 n(오프셋 워드 수)으로 지정한 양만큼 오프셋됩니다.



선두 워드 어드레스

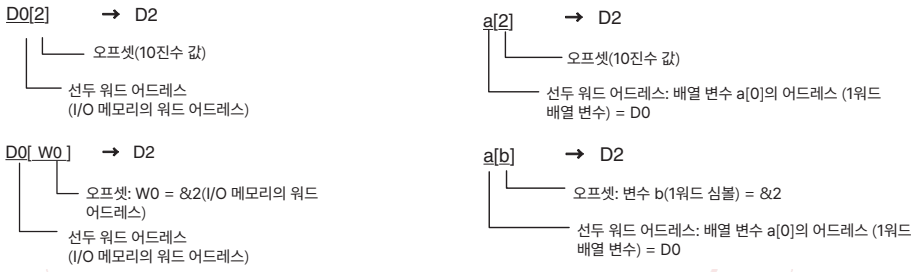
선두 워드 어드레스는 워드 어드레스 또는 배열 변수(STRING 및 NUMBER 데이터형 제외)로 지정할 수 있습니다. 오프셋은 CIO, H, W, DM, EM 영역의 어드레스에 대해서만 가능합니다. 그 밖의 영역의 어드레스를 지정하면 동작이 보장되지 않습니다. I/O 코멘트는 이 선두 워드 어드레스의 I/O 코멘트를 나타냅니다.

오프셋

오프셋은 10진 정수, 워드 어드레스(단, CIO 영역의 어드레스는 지정할 수 없습니다) 또는 1워드 심볼(즉, INT, UINT, WORD, CHANNEL 데이터형의 심볼)로 지정할 수 있습니다. 워드 어드레스를 지정하면 지정한 워드의 내용이 오프셋으로 사용됩니다.

오프셋 워드 수가 선두 워드 어드레스의 메모리 영역을 넘으면, 최종 워드 어드레스는 실제 PLC 메모리 어드레스로 결정되는 순서에 따라 다음 메모리 영역에 위치합니다. (메모리 맵의 영역 배치에 대해서는 A-4 PLC 메모리 어드레스의 메모리 맵을 참조하십시오.

예:



⚠ 주의

래더 프로그램에서 심볼 또는 어드레스를 사용하여 오프셋을 직접 지정할 때에는 선두 어드레스의 메모리 영역을 넘지 않도록 프로그래밍하십시오.

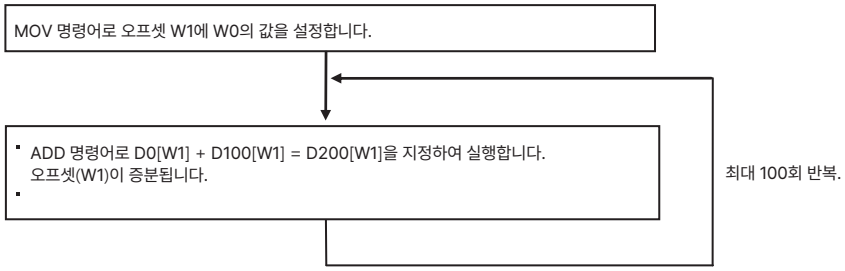
간접 지정에 의해 어드레스가 선두 어드레스의 메모리 영역을 넘으면, 시스템이 다음 영역의 데이터에 액세스하여 예기치 않은 동작이 발생할 수 있습니다.

5-8-2 어드레스 오프셋 응용 예

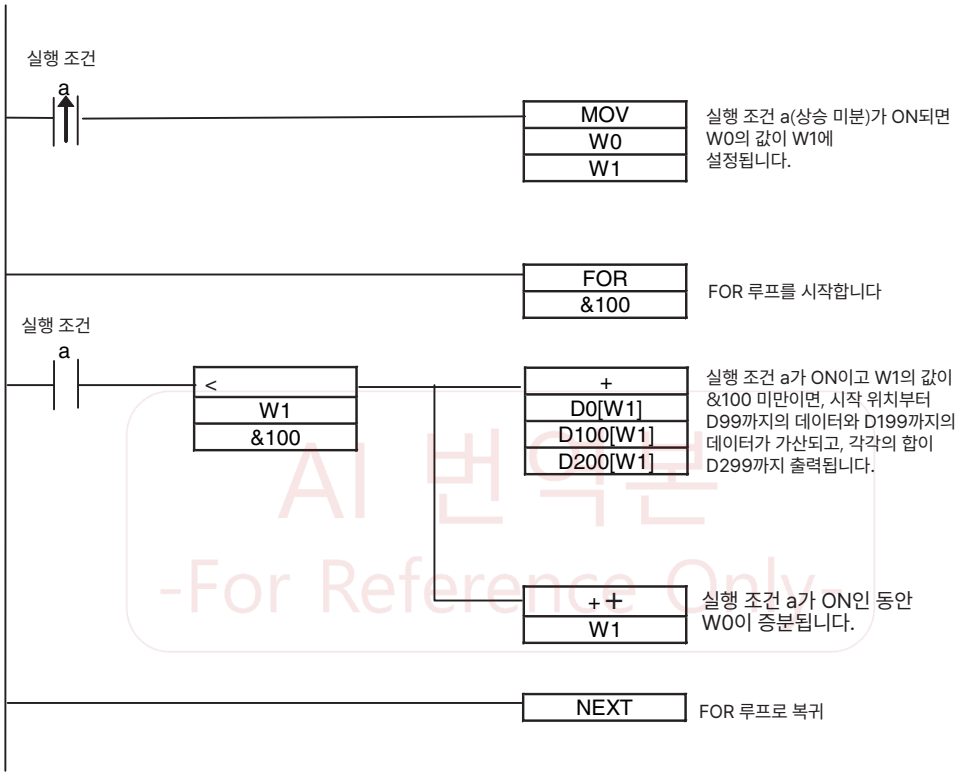
오프셋을 넣는 대괄호 안에 I/O 메모리의 워드 어드레스를 지정하면 오프셋을 동적으로 지정할 수 있습니다. 지정한 워드 어드레스의 내용이 오프셋으로 사용됩니다. 예를 들어 대괄호 안의 값을 증분하면서 명령어 하나만으로 어드레스를 늘려 가며 실행할 수 있습니다.

● 래더 프로그래밍 예

이 예에서는 D0~D99와 D100~D199의 두 연속 데이터 영역을 사용합니다. 지정한 시작점 W0부터 영역의 끝까지 대응하는 워드의 내용을 가산하고, 그 합을 D200에서 지정한 오프셋만큼 떨어진 위치부터 D200~D299에 출력합니다. 예를 들어 W0이 30이면 D30~D99와 D130~D199의 대응 워드가 가산되고, 그 합이 D230~D299에 출력됩니다.



각 처리는 W1이 &100을 넘지 않도록 하는 입력 비교 명령어(<)를 실행 조건으로 하여 수행되며, 이로써 간접 지정 범위의 상한을 넘지 않도록 합니다.



AI 번역본
-For Reference Only-

5-9 프로그램 체크

CJ 시리즈 프로그램은 다음 네 단계에서 체크할 수 있습니다.

- CX-Programmer 입력 조작 시의 입력 체크
- CX-Programmer에 의한 프로그램 체크
- 실행 중의 명령어 체크
- 실행 중의 치명 에러 체크(프로그램 에러)

5-9-1 CX-Programmer 입력 시의 에러

프로그램은 다음 시점에 CX-Programmer에 의해 자동으로 체크됩니다.

시점	체크 내용
래더 다이어그램 입력 시	명령어 입력, 오퍼랜드 입력, 프로그래밍 패턴
파일 로드 시	전 명령어의 전 오퍼랜드와 전 프로그래밍 패턴
파일 다운로드 시	CJ 시리즈가 지원하는 형식과 전 명령어의 전 오퍼랜드
온라인 에디팅 중	용량 등

체크 결과는 Output 윈도우의 텍스트 탭에 출력됩니다. 또한 부정한 프로그램 부분의 좌측 모선이 래더 뷰에서 빨간색으로 표시됩니다.

5-9-2 CX-Programmer에 의한 프로그램 체크

사용자 프로그램은 CX-Programmer에서 체크할 수 있습니다. 프로그램을 체크할 때 사용자는 A, B 또는 C(에러의 심각도 순)나 커스텀 체크 레벨 중 네 가지 레벨 중 하나로 프로그램 체크를 지정할 수 있습니다.

CX-Programmer는 명령어에서 간접 지정된 오퍼랜드의 범위 에러를 체크하지 않습니다. 명령어의 오퍼랜드 데이터가 부정한 경우, 다음 절에서 설명하는 프로그램 실행 체크 중에 ER 플래그가 ON됩니다. 자세한 내용은 CS/CJ/NSJ 시리즈 명령어 레퍼런스 매뉴얼(Cat. No. W474)을 참조하십시오.

CX-Programmer의 체크에 대한 자세한 내용은 CX-Programmer 오퍼레이션 매뉴얼(Cat. No. W446)을 참조하십시오.

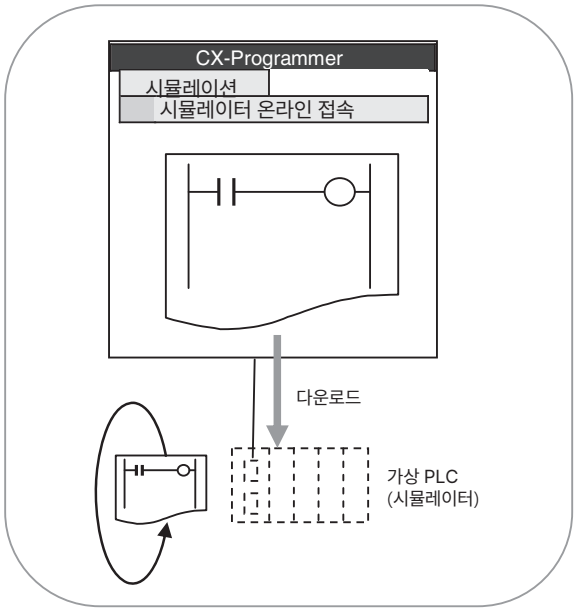
AI 번역본
-For Reference Only-

5-9-3 시뮬레이터에 의한 디버그

컴퓨터에서 CPU 유닛의 동작을 시뮬레이션하여 실제 PLC에 접속하지 않고 프로그래밍을 디버그할 수 있습니다.

래더 프로그램 동작 확인

작성한 프로그래밍은 CX-Programmer에서 CX-Simulator의 시뮬레이터를 기동하여 가상 PLC로 체크 · 디버그할 수 있습니다.



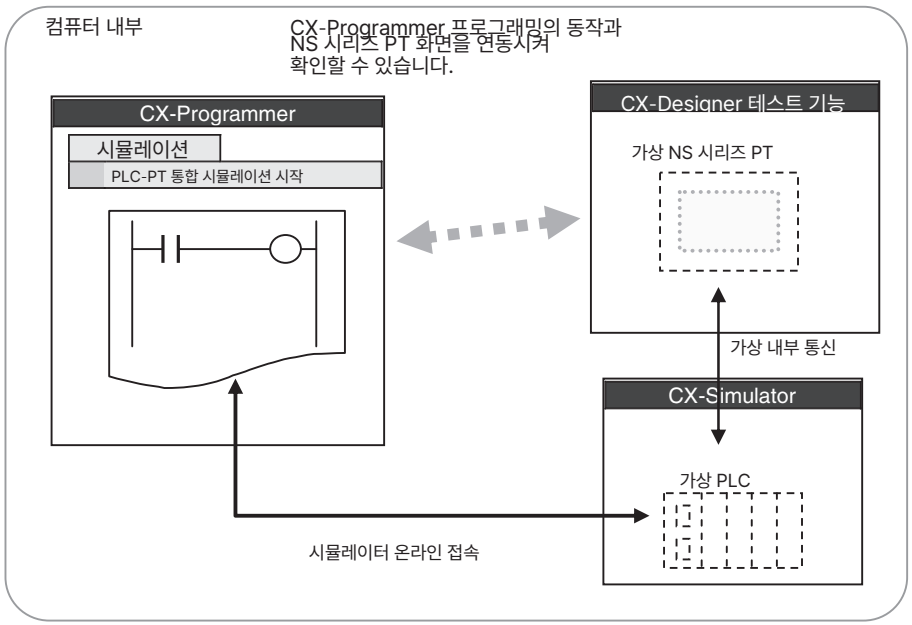
프로그램 전송과 모니터링 외에도 시뮬레이터에서 다음 기능을 사용할 수 있습니다. 디버그 순서에 대한 자세한 내용은 CX-Programmer 오퍼레이션 매뉴얼(Cat. No. W446)을 참조하십시오.

- 스텝 실행, 연속 스텝 실행 또는 스캔 실행.
- 중단점, 시작점, I/O 중단 조건의 지정.
- 태스크별 실행 횟수와 실행 시간의 확인.
- 인터럽트 태스크 실행의 시뮬레이션.
- 비트의 강제 세트 및 강제 리셋.

AI 번역본
-For Reference Only-

PT와 래더 프로그래밍 사이의 동작 디버그: 통합 시뮬레이션

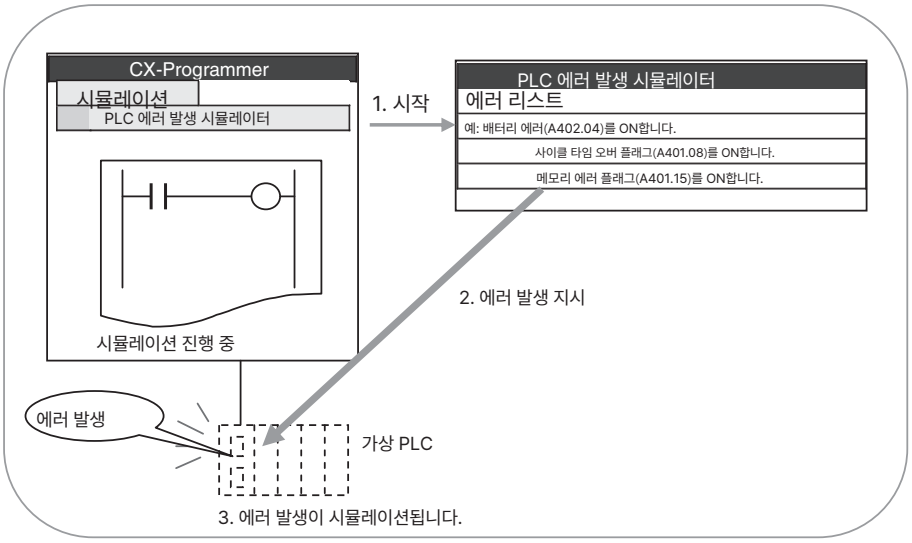
PLC-PT 통합 시뮬레이션을 사용하면 NS 시리즈 PT의 화면 데이터와 CJ 시리즈 PLC 프로그램 사이의 동작을 디버그할 수 있습니다. 이는 CX-Programmer의 시뮬레이터 기능과 PT 화면 작성 소프트웨어 CX-Designer의 오프라인 테스트 기능을 연동시켜 실현합니다. 이에 따라 컴퓨터와 PT를 케이블로 접속하지 않고 컴퓨터만으로 래더 프로그래밍에서 화면과 화면 제어를 디버그할 수 있습니다. PLC-PT 통합 시뮬레이션은 CX-Programmer에서도 기동할 수 있습니다. 디버그 순서에 대한 자세한 내용은 CX-Programmer 오퍼레이션 매뉴얼(Cat. No. W446)을 참조하십시오.



AI 번역본
-For Reference Only-

에러 시뮬레이션 기능

CX-Programmer에서는 래더 프로그래밍 시뮬레이션 중에 가상 PLC에서 시스템 에러를 발생시킬 수 있습니다. 전용 조작 창에서 원하는 치명 에러 또는 비치명 에러를 발생시켜 PLC 시스템 에러가 발생했을 때의 래더 프로그래밍과 NS 시리즈 PT의 동작을 손쉽게 확인할 수 있습니다. 디버그 순서에 대한 자세한 내용은 CX-Programmer 오퍼레이션 매뉴얼(Cat. No. W446)을 참조하십시오.



주 실제 에러와 달리, PLC 에러 발생 시뮬레이션 기능으로 치명 에러를 발생시켜도 래더 실행은 정지하지 않습니다.

참고

FAL(006) 또는 FALS(007) 명령어를 사용해서도 PLC에서 시스템 에러를 발생시킬 수 있습니다.

AI 번역본
-For Reference Only-

5-9-4 프로그램 실행 확인

프로그래밍이 실행 중일 때 CPU 유닛에 의해 다음 체크가 수행됩니다.

명령어 실행 중에 다음 네 가지 체크가 수행됩니다.

에러 종류	에러 시 ON되는 플래그	운전 정지/계속
명령어 처리 에러	ER 플래그 에러 발생 시 운전 정지가 지정되어 있으면 명령어 처리 에러 플래그(A295.08)도 ON됩니다.	**PLC 설정의 Stop CPU on Instruction** Error 체크 박스에서 명령어 처리 에러 시 운전을 정지할지 계속할지를 지정할 수 있습니다. 기본값은 운전 계속입니다. 운전 정지가 지정된 경우에만 프로그램 에러가 발생하고 운전이 정지됩니다.
액세스 에러	AER 플래그 에러 발생 시 운전 정지가 지정되어 있으면 액세스 에러 플래그(A295.10)가 ON됩니다.	**PLC 설정의 Stop CPU on Instruction** Error 체크 박스에서 명령어 처리 에러 시 운전을 정지할지 계속할지를 지정할 수 있습니다. 기본값은 운전 계속입니다. 운전 정지가 지정된 경우에만 프로그램 에러가 발생하고 운전이 정지됩니다.
부정 명령어 에러	부정 명령어 에러 플래그 (A295.14)	치명(프로그램 에러)
사용자 프로그램 영역 오버플로 에러	사용자 프로그램 영역 오버플로 에러 플래그(A295.15)	치명(프로그램 에러)

명령어 처리 에러(P_ER 플래그 ON 에러)

- 명령어 실행 시에 부정한 데이터가 주어졌거나 태스크 외부에서 명령어를 실행하려고 한 경우 명령어 처리 에러가 발생합니다. 여기서는 명령어 처리의 시작 시점에 필요한 데이터가 체크되고, 그 결과 명령어가 실행되지 않으면 P_ER 플래그(에러 플래그)가 ON되며 P_EQ와 P_N 플래그는 명령어에 따라 유지되거나 OFF됩니다. 명령어(입력 명령어 제외)가 정상적으로 종료되면 P_ER 플래그(에러 플래그)는 OFF됩니다. P_ER 플래그를 ON시키는 조건은 명령어마다 다릅니다. 자세한 내용은 CS/CJ/NSJ 시리즈 명령어 레퍼런스 매뉴얼(Cat. No. W474)의 개별 명령어 설명을 참조하십시오.
- PLC 설정에서 명령어 에러가 운전 정지로 설정되어 있으면, 명령어 처리 에러가 발생하여 P_ER 플래그가 ON될 때 운전이 정지되고(치명 에러) 명령어 처리 에러 플래그 (A295.08)가 ON됩니다.

AI 번역본
For Reference Only

부정 액세스 에러(P_AER 플래그 ON 에러)

- 부정 액세스 에러는 명령어 오퍼랜드를 지정하는 어드레스에 액세스했을 때 다음 중 하나의 방식으로 잘못된 영역에 액세스했음을 나타냅니다.
  - 파라미터 영역에 대해 읽기 또는 쓰기가 실행되었습니다.
  - 시스템이 예약한 영역에 대해 읽기 또는 쓰기가 실행되었습니다.
  - 실장되어 있지 않은 메모리 영역에 쓰기가 실행되었습니다*1
  - EM 파일 메모리로 지정된 EM 영역 뱅크에 쓰기가 실행되었습니다.
  - 읽기 전용 영역에 쓰기가 실행되었습니다.
  - BCD 모드의 간접 DM/EM 어드레스에 지정된 값이 BCD가 아니었습니다(예: *D1에 #A000이 들어 있음).
- 액세스 에러가 발생해도 명령어 처리는 계속되고 에러 플래그(ER 플래그)는 ON되지 않지만, 액세스 에러 플래그(P_AER 플래그)가 ON됩니다.

*1 다음의 경우에 액세스 에러가 발생합니다:

- 현재 뱅크에 대해 지정한 EM 어드레스가 32767을 초과한 경우(예: E32768).
- BIN 모드의 간접 EM 어드레스로 최종 뱅크가 지정되어 있고 지정한 워드에 8000~ FFFF Hex가 저장되어 있는 경우(예: @EC_00001에 #8000이 저장됨).
- BIN 모드의 간접 EM 어드레스로 현재 뱅크가 지정되어 있고 지정한 워드에 8000~ FFFF Hex가 저장되어 있는 경우(예: @EC_00001에 #8000이 저장됨)
- 비트의 내부 메모리 어드레스가 저장된 IR 레지스터를 워드 어드레스로 사용하거나, 워드의 내부 메모리 어드레스가 저장된 IR을 비트 어드레스로 사용한 경우.

- PLC 설정에서 명령어 에러가 동작 정지로 설정되어 있는 경우, 부정 액세스 에러가 발생하여 AER 플래그가 ON되면 동작이 정지하고(치명 에러) "부정 액세스 에러 플래그"(A295.10)가 ON됩니다.

참고

액세스 에러 플래그(P_AER 플래그)는 태스크 실행 후에도 클리어되지 않습니다. 명령어 에러가 동작 계속으로 설정되어 있으면, END(001) 명령어 직전까지 이 플래그를 모니터링하여 태스크 프로그램에서 부정 액세스 에러가 발생했는지 확인할 수 있습니다. (CX-Programmer에서 AER 플래그를 모니터링하면 사용자 프로그램 전체가 실행된 후의 최종 P_AER 플래그 상태를 모니터링하게 됩니다.)

기타 에러

● 부정 명령어 에러

부정 명령어 에러는 시스템에 정의된 것 이외의 명령어 데이터를 실행하려고 시도했음을 나타냅니다. 이 에러는 프로그램을 CX-Programmer로 작성하는 한 통상적으로는 발생하지 않습니다.

드물게 이 에러가 발생한 경우에는 프로그램 에러로 취급되어 동작이 정지하고(치명 에러) 부정 명령어 플래그(A295.14)가 ON됩니다.

● 사용자 프로그램 영역 오버플로 에러

사용자 프로그램 영역 오버플로 에러는 프로그램 저장 영역으로 정의된 사용자 프로그램 영역의 최종 어드레스를 넘어서 저장된 명령어 데이터를 실행하려고 시도했음을 나타냅니다. 이 에러는 프로그램을 CX-Programmer로 작성하는 한 통상적으로는 발생하지 않습니다.

드물게 이 에러가 발생한 경우에는 프로그램 에러로 취급되어 동작이 정지하고(치명 에러) UM 오버플로 플래그(A295.15)가 ON됩니다.

참고

에러 플래그(P_ER) 또는 부정 액세스 에러 플래그(P_AER)가 ON되면 프로그램 에러로 취급되며, 이를 사용하여 CPU 유닛의 운전을 정지시킬 수 있습니다. 프로그램 에러 시의 동작은 PLC 설정에서 지정합니다.

사내 배포용
-For Reference Only-

● 프로그램 에러

프로그램 에러	설명	관련 플래그
END 명령어 없음	프로그램에 END 명령어가 존재하지 않습니다.	END 없음 플래그(A295.11)가 ON 됩니다.
태스크 실행 중 에러	사이클 내에 READY 상태인 태스크가 없습니다. 태스크에 프로그램이 할당되어 있지 않습니다. 인터럽트 태스크의 실행 조건이 충족되었 음에도 해당하는 인터럽트 태스크 번호가 존재하지 않습니다.	태스크 에러 플래그(295.12)가 ON 됩니다.
명령어 처리 에러(P_ER 플래그 ON) 및 PLC 설정의 명령어 에러가 동작 정지로 설정되어 있는 경우	명령어를 실행하려고 했을 때 오퍼랜드에 잘못된 데이터 값이 지정되어 있었습니다.	ER 플래그가 ON되고, PLC 설정의 명령어 에러가 동작 정지로 설정되어 있으면 명령어 처리 에러 플래그 (A295.08)도 ON 됩니다.
부정 액세스 에러(P_AER 플래그 ON)이고 PLC 설정의 명령어 에러가 동작 정지로 설정되어 있는 경우	파라미터 영역에 대해 읽기 또는 쓰기가 실행되었습니다. 시스템용으로 예약된 영역에 대해 읽기 또는 쓰기가 실행되었습니다. 실장되어 있지 않은 메모리 영역에 대해 읽기 또는 쓰기가 실행되었습니다. EM 파일 메모리로 지정된 EM 영역 뱅크에 대해 읽기 또는 쓰기가 실행되었습니다. 읽기 전용 영역에 대해 쓰기가 실행되었습니다. BCD 모드의 간접 DM/EM 어드레스에 지정된 값이 BCD가 아니었습니다.	AER 플래그가 ON되고, PLC 설정의 명령어 에러가 동작 정지로 설정되어 있으면 부정 액세스 에러 플래그 (A295.10)도 ON됩니다
간접 DM/EM BCD 에러이고 PLC 설정의 명령어 에러가 동작 정지로 설정되어 있는 경우	BCD 모드의 간접 DM/EM 어드레스에 지정된 값이 BCD가 아닙니다.	액세스 에러 플래그가 ON 됩니다. PLC 설정에서 Stop CPU on Instruction Error 체크 박스가 선택되어 있으면 간접 DM/EM BCD 에러 플래그(A295.09)도 ON됩니다.
미분 어드레스 오버플로 에러	온라인 에디팅 중에 131,071개를 초과하는 미분 명령어가 삽입 또는 삭제되었습니다.	미분 오버플로 에러 플래그 (A295.13)가 ON됩니다.
부정 명령어 에러	실행할 수 없는 명령어를 실행하려고 시도했습니다.	부정 명령어 플래그 (A295.14)가 ON됩니다.
UM(사용자 메모리) 오버플로 에러	프로그램 저장 영역으로 정의된 사용자 메모리(UM)의 최종 어드레스를 넘어서 저장된 명령어 데이터를 실행하려고 시도했습니다.	UM(사용자 메모리) 오버플로 플래그 (A295.15)가 ON됩니다.

AI 번역본
-For Reference Only-

5-10 주의사항

5-10-1 조건 플래그

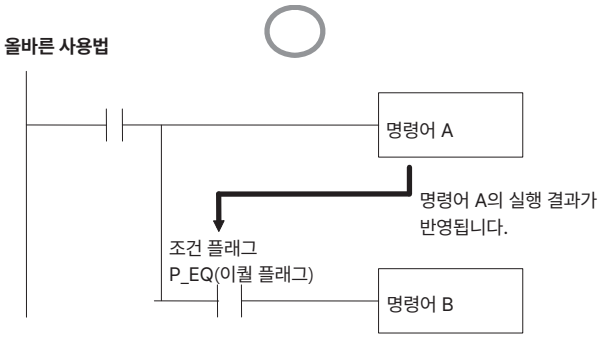
조건 플래그의 사용

조건 플래그는 모든 명령어가 공유하며, 개별 명령어의 실행 결과에 따라 1사이클 중에 변화합니다. 따라서 명령어 실행 결과를 반영하려면 반드시 그 명령어 바로 뒤에 동일한 실행 조건으로 분기시킨 출력에서 조건 플래그를 사용하십시오. 조건 플래그를 모선에 직접 접속하면 다른 명령어의 실행 결과가 반영되므로 절대로 직접 접속하지 마십시오.

예: 명령어 A의 실행 결과 사용

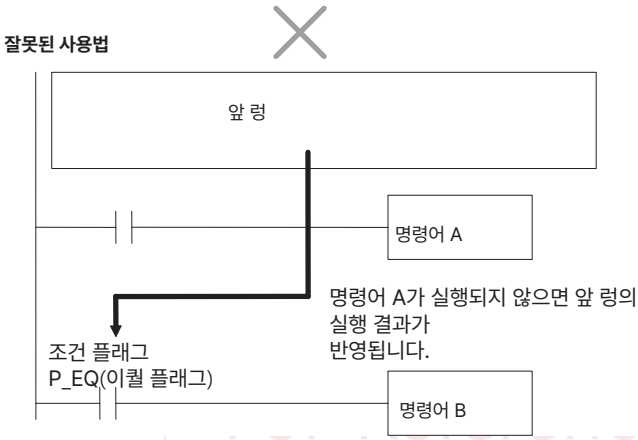
니모닉

올바른 사용법



명령어 A의 실행 결과에 따라 명령어 B를 실행하기 위해 명령어 A와 B에 동일한 실행 조건 (a)을 사용합니다. 이 경우 명령어 B는 명령어 A가 실행된 경우에만 조건 플래그에 따라 실행됩니다.

잘못된 사용법

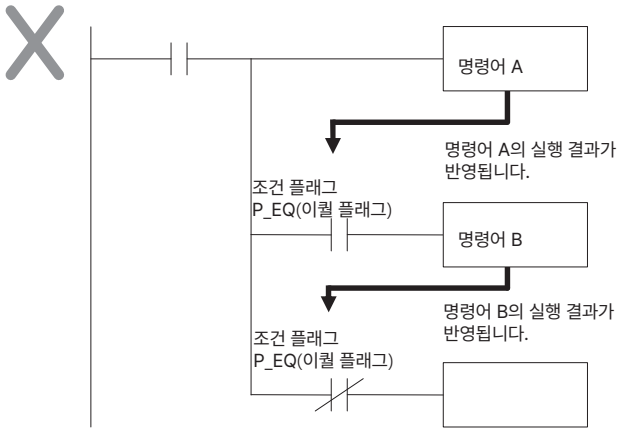


조건 플래그를 좌측 모선에 직접 접속하면, 명령어 A가 실행되지 않은 경우 명령어 B는 앞 렁의 실행 결과에 따라 실행됩니다.

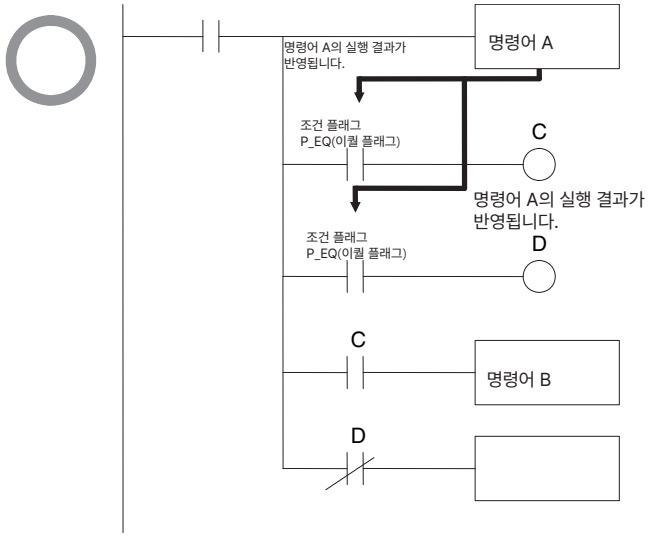
인터럽트 태스크를 사용하는 경우, 사이클릭 태스크 실행 중이라도 기동 조건이 충족되면 인터럽트 태스크가 동작합니다. 이 경우 인터럽트 태스크에서 조건 플래그가 ON/OFF되었 더라도, 처리가 인터럽트 태스크에서 사이클릭 태스크로 돌아올 때 조건 플래그는 원래 상태로 복귀합니다.

● N.C. 입력과 N.O. 입력에서의 실행 결과 사용

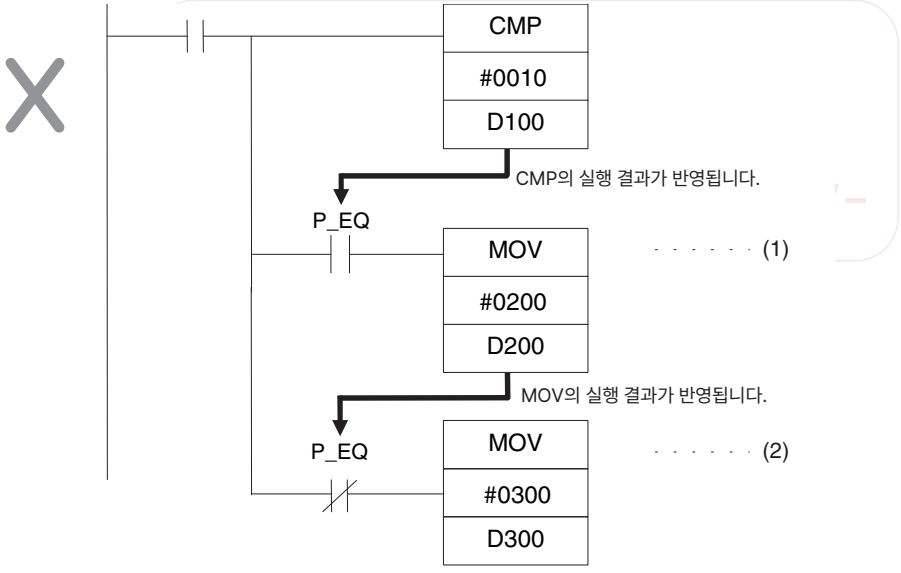
N.C. 입력 비트와 N.O. 입력 비트가 동일한 출력 분기에서 실행되더라도, 아래 예와 같이 조건 플래그는 명령어 B의 실행 결과를 받아들입니다.



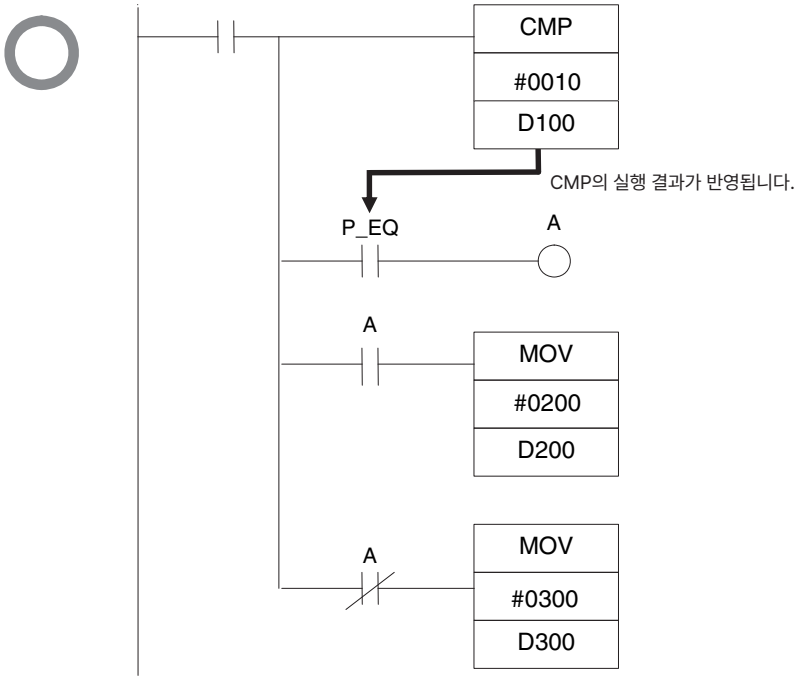
명령어 B의 실행 결과가 받아들여지지 않도록, 각 결과를 OUTPUT 명령어로 한 번씩 받아들이도록 하십시오.



예: 다음 예에서는 D100에 #0010이 저장되어 있으면 #0200을 D200으로 전송하고, D100에 #0010이 저장되어 있지 않으면 #0300을 D300으로 전송합니다.



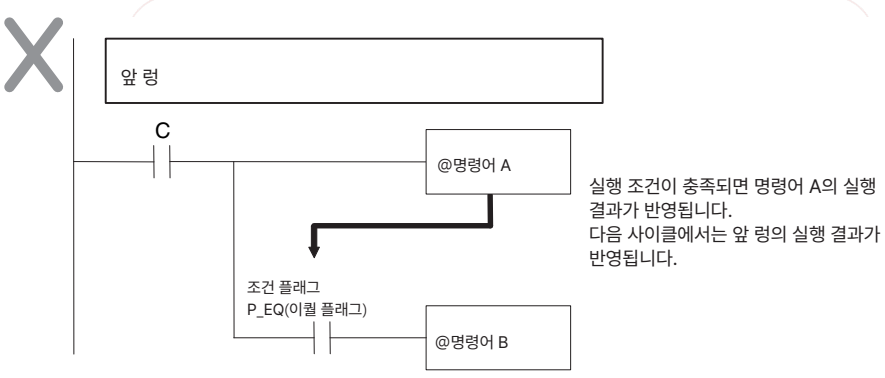
위 렁에서 D100에 #0010이 저장되어 있으면 이퀄 플래그가 ON됩니다. 명령어 (1)에서 #0200이 D200으로 전송되지만, 전송원 데이터 #0200이 0000 Hex가 아니므로 이퀄 플래그는 OFF됩니다. 그러면 (2)의 MOV 명령어가 실행되어 #0300이 D300으로 전송됩니다.
따라서 첫 번째 MOV 명령어의 실행 결과가 받아들여지지 않도록 아래와 같이 렁을 삽입해야 합니다.



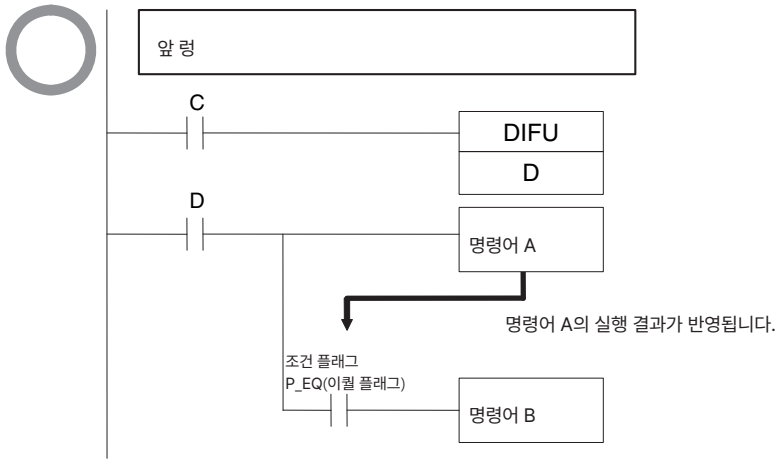
● 미분 명령어의 실행 결과 사용

미분 명령어에서는 실행 조건이 충족되었을 때만 명령어의 실행 결과가 조건 플래그에 반영되며, 다음 사이클에서는 (미분 명령어의 실행 결과가 아니라) 앞 렁의 결과가 조건 플래그에 반영됩니다. 따라서 미분 명령어의 실행 결과를 사용하는 경우에는 다음 사이클에서 조건 플래그가 어떻게 되는지에 주의해야 합니다.

예를 들어 다음에서 명령어 A와 B는 실행 조건 C가 충족된 경우에만 실행되지만, 명령어 B가 명령어 A의 실행 결과를 받아들일 때 다음과 같은 문제가 발생합니다. 명령어 A가 실행된
다음 사이클에서도 실행 조건 C가 ON인 상태로 유지되면, 앞 렁에서 반영된 결과로 인해 조건 플래그가 OFF에서 ON으로 바뀔 때 명령어 B가 (실행 조건에 의해) 의도치 않게 실행됩니다.



이 경우에는 명령어 A와 B를 미분 명령어로 하지 않고, 아래와 같이 대신 DIFU(of DIFD) 명령어를 사용하여 명령어 A와 B를 모두 상승(또는 하강) 미분시켜 1사이클만 실행합니다.



참고

조건 플래그 세이브 및 조건 플래그 로드(CCS(282)와 CCL(283)) 명령어를 사용하면 조건 플래그 상태를 세이브하고 로드할 수 있습니다. 이를 사용하여 태스크 내의 다른 위치나 다른 태스크에서 조건 플래그 상태에 액세스할 수 있습니다.

조건 플래그를 ON시키는 주요 조건

● 에러 플래그(P_ER)

에러 플래그는 명령어의 오퍼랜드 데이터가 올바르지 않은 경우 등 특수한 조건에서 ON됩니다. 에러 플래그가 ON되면 해당 명령어는 실행되지 않습니다.

에러 플래그가 ON일 때 <, >, OF, UF 플래그 등 다른 조건 플래그의 상태는 변하지 않으며, = 플래그와 N 플래그의 상태는 명령어에 따라 달라집니다.

에러 플래그가 ON되는 조건에 대해서는 CS/CJ/NSJ 시리즈 명령어 레퍼런스 매뉴얼(Cat. No. W474)의 각 명령어 설명을 참조하십시오. 일부 명령어는 조건에 관계없이 에러 플래그를 OFF시키므로 주의가 필요합니다.

참고

에러 플래그가 ON되었을 때 동작을 정지할지 여부는 명령어 에러 발생 시에 대한 PLC 설정 설정값에 따라 결정됩니다. 디폴트 설정에서는 에러 플래그가 ON되어도 동작이 계속됩니다. 에러 플래그가 ON되었을 때 동작 정지가 지정되어 동작이 정지하면(프로그램 에러로 취급), 동작이 정지한 시점의 프로그램 어드레스가 A298~A299에 저장됩니다. 동시에 A295.08이 ON됩니다.

-For Reference Only-

● 이퀄 플래그(P_EQ)

이퀄 플래그는 비교 결과가 같은(=) 경우를 제외하고 모든 명령어에 대해 일시적인 플래그입니다. 시스템에 의해 자동으로 세트되며 변화합니다. 앞의 명령어가 이퀄 플래그를 ON(OFF)시킨 뒤에 다른 명령어가 이를 OFF(ON)시킬 수 있습니다. 예를 들어 이퀄 플래그는 MOV 등의 전송 명령어가 전송원 데이터로 0000 Hex를 전송할 때 ON되고, 그 밖의 경우에는 OFF입니다. 어떤 명령어가 이퀄 플래그를 ON시키더라도 전송 명령어는 즉시 실행되며, 전송 명령어의 전송원 데이터가 0000 Hex인지 여부에 따라 이퀄 플래그가 ON 또는 OFF됩니다.

● 캐리 플래그(P_CY)

CY 플래그는 시프트 명령어, 캐리 입력이 있는 가산 · 감산 명령어, 가산 · 감산 명령어의 빌림과 올림, 그리고 특수 I/O 유닛 명령어, PID 명령어, FPD 명령어에서 사용됩니다. 다음 주의사항에 유의하십시오.

- CY 플래그는 시프트 명령어, 캐리 입력이 있는 가산 · 감산 명령어, 가산 · 감산 명령어의 빌림과 올림, 그리고 특수 I/O 유닛 명령어, PID 명령어, FPD 명령어에서 사용됩니다. 다음 주의사항에 유의하십시오.
- CY 플래그는 어떤 명령어의 실행 결과에 의해 ON(OFF)되고 다른 명령어에 의해 OFF(ON)될 수 있습니다. 캐리 플래그를 사용할 때는 올바른 결과가 반영되어 있는지 확인하십시오.

● 미만 플래그와 초과 플래그(P_LT, P_GT)

> 플래그와 < 플래그는 비교 명령어를 비롯해 LMT, BAND, ZONE, PID 등의 명령어에서 사용됩니다.

> 또는 < 플래그는 어떤 명령어의 실행 결과에 의해 ON(OFF)되더라도 다른 명령어에 의해 OFF(ON)될 수 있습니다.

● 네거티브 플래그(P_N)

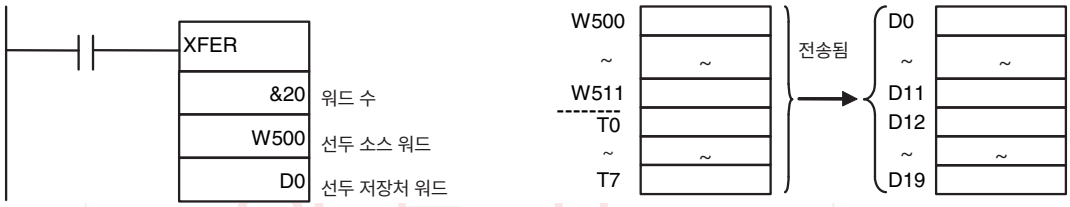
N 플래그는 일부 명령어에서 명령어 실행 결과 워드의 최좌측 비트가 "1"일 때 ON되며, 다른 명령어에서는 무조건 OFF됩니다.

● 여러 워드에 걸친 오퍼랜드 지정

여러 워드가 필요한 오퍼랜드를 지정할 때 그 오퍼랜드의 모든 워드가 동일한 영역에 있지 않더라도 명령어는 기술된 대로 실행됩니다. 이 경우 워드는 PLC 메모리 어드레스 순서대로 취급됩니다. 에러 플래그는 ON되지 않습니다.

예를 들어 W500부터 20워드를 전송하도록 지정하여 XFER(070)으로 블록 전송을 실행한 결과를 생각해 봅니다. 여기서는 W511에서 끝나는 워크 영역을 초과하게 되지만, 에러 플래그가 ON되지 않고 명령어가 실행됩니다. PLC 메모리 어드레스에서는 워크 영역 뒤에 타이머의 현재값이 메모리에 유지되므로, 다음 명령어에서는 W500~W511이 D0~D11로 전송되고 T0~T7의 현재값이 D12~D19로 전송됩니다.

주 구체적인 PLC 메모리 어드레스에 대해서는 부록 PLC 메모리 어드레스의 메모리 맵을 참조하십시오.



-For Reference Only-

5-10-2 특수 프로그램 섹션

CJ 시리즈 프로그램에는 명령어 조건을 제어하는 특수 프로그램 섹션이 있습니다. 다음과 같은 특수 프로그램 섹션을 사용할 수 있습니다

프로그램 섹션	명령어	명령어 조건	상태
서브루틴	SBS, SBN, RET 명령어 GSBS, GSBN, GRET 명령어	서브루틴 프로그램이 실행됩니다.	SBN 명령어와 RET 명령어 사이의 서브루틴 프로그램 섹션이 실행됩니다.
IL - ILC 섹션	IL 명령어와 ILC 명령어	섹션이 인터록됩니다	출력 비트가 OFF되고 타이머가 리셋됩니다. 그 밖의 명령어는 실행되지 않으며 이전 상태가 유지됩니다.
스텝 래더 섹션	STEP 명령어		
FOR-NEXT 루프	FOR 명령어와 NEXT 명령어	중단 중.	루프 중
JMP0 - JME0 섹션	JMP0 명령어와 JME0 명령어		점프
블록 프로그램 섹션	BPRG 명령어와 BEND 명령어	블록 프로그램이 실행 중입니다.	BPRG 명령어와 BEND 명령어 사이에 니모닉으로 기술된 블록 프로그램이 실행됩니다.

명령어 조합

다음 표는 특수 명령어 중 어느 것을 다른 프로그램 섹션 안에서 사용할 수 있는지 나타냅니다.

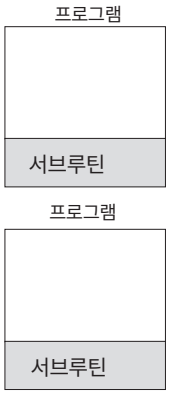
	서브루틴	IL - ILC 섹션	스텝 래더 섹션	FOR - NEXT 루프	JMP0 - JME0 섹션	블록 프로그램 섹션
서브루틴	불가능.	불가능.	불가능.	불가능.	불가능.	불가능.
IL - ILC	OK	불가능.	불가능.	OK	OK	불가능.
스텝 래더 섹션	불가능.	OK	불가능.	불가능.	OK	불가능.
FOR - NEXT 루프	OK	OK	불가능.	OK	OK	불가능.
JMP0 - JME0	OK	OK	불가능.	불가능.	불가능.	불가능.
블록 프로그램 섹션	OK	OK	OK	불가능.	OK	불가능.

주 프로그램 영역을 지정하는 명령어는 다른 태스크의 프로그램에는 사용할 수 없습니다. 자세한 내용은 5-2-4 태스크 설계를 참조하십시오.

AI 번역본
-For Reference Only-

서브루틴

모든 프로그램에서 서브루틴은 서브루틴 이외의 프로그래밍 뒤이면서 END(001) 명령어 바로 앞에 한데 모아 배치하십시오. (따라서 서브루틴은 스텝 래더, 블록 프로그램, FOR - NEXT, JMP0 - JME0 섹션에는 배치할 수 없습니다.) 서브루틴 프로그램(SBN~RET) 뒤에 서브루틴 이외의 프로그램을 배치하면 그 프로그램은 실행되지 않습니다.



서브루틴에서 사용할 수 없는 명령어

다음 명령어는 서브루틴에 배치할 수 없습니다.

기능	니모닉	명령어
공정 스텝 제어	STEP(008)	스텝 래더 섹션 정의
	SNXT(009)	스텝 래더 스텝 진행

주 서브루틴에는 블록 프로그램 섹션을 포함할 수 있습니다. 단, 서브루틴에서 메인 프로그램으로 실행이 복귀할 때 블록 프로그램이 WAIT 상태이면, 다음에 호출될 때에도 블록 프로그램 섹션은 WAIT 상태로 유지됩니다.

AI 번역본
-For Reference Only-

스텝 래더 프로그램 섹션에서 사용할 수 없는 명령어

다음 명령어는 스텝 래더 프로그램 섹션에서 사용할 수 없습니다.

기능	니모닉	명령어
시퀀스 제어	FOR(512), NEXT(513), BREAK(514)	FOR, NEXT, BREAK LOOP
	END(001)	END
	IL(002)과 ILC(003)	인터록과 인터록 해제
	JMP(004)와 JME(005)	점프와 점프 종료
	CJP(510)와 CJPN(511)	조건부 점프와 조건부 점프 NOT
	JMP0(515)와 JME0(516)	다중 점프와 다중 점프 종료
서브루틴	SBN(092), RET(093), GSBN (751), GRET(752)	서브루틴 엔트리와 서브루틴 리턴
블록 프로그램	IF(802) (NOT), ELSE(803), IEND(804)	조건부 블록 분기 (NOT), 조건부 블록 분기 ELSE, 조건부 블록 분기 종료
	BPRG(096)과 BEND(801)	블록 프로그램 개시/종료
	EXIT(806) (NOT)	조건부 블록 탈출 (NOT)
	LOOP(809)와 LEND(810) (NOT)	루프 블록과 루프 블록 종료 (NOT)
	WAIT(805) (NOT)	1 사이클 대기 (NOT)
	TIMW(813)	타이머 대기
	TMHW(815)	고속 타이머 대기
	CNTW(814)	카운터 대기
	BPPS(811)과 BPRS(812)	블록 프로그램 일시 정지와 재개

주1 스텝 래더 프로그램 섹션은 인터록 섹션(IL과 ILC 사이)에서 사용할 수 있습니다. 인터록이 ON이면 스텝 래더 섹션은 완전히 리셋됩니다.

2 스텝 래더 프로그램 섹션은 다중 점프(JMP0)와 다중 점프 종료(JME0) 사이에서 사용할 수 있습니다.

AI 번역본
-For Reference Only-

블록 프로그램 섹션에서 지원하지 않는 명령어

다음 명령어는 블록 프로그램 섹션에 배치할 수 없습니다

기능별 분류	니모닉	명령어
시퀀스 제어	FOR(512), NEXT(513), BREAK(514)	FOR, NEXT, BREAK LOOP
	IL(002)과 ILC(003)	인터록과 인터록 해제
	JMP0(515)와 JME0(516)	다중 점프와 다중 점프 종료
	END(001)	END
시퀀스 입력	UP(521)	조건 ON
	DOWN(522)	조건 OFF
시퀀스 출력	DIFU	미분 업
	DIFD	미분 다운
	KEEP	KEEP
	OUT	출력
	OUT NOT	출력 NOT
타이머/카운터	TIM	100ms 타이머
	TIMH	10ms 타이머
	TMHH(540)	1ms 타이머
	TIMU	0.1ms 타이머
	TMUH	0.01ms 타이머
	TTIM(087)	적산 타이머
	TIML(542)	롱 타이머
	MTIM(543)	다출력 타이머
	CNT	카운터
	CNTR	리버시블 카운터
서브루틴	SBN(092)과 RET(093)	서브루틴 엔트리와 서브루틴 리턴
데이터 시프트	SFT	시프트
래더 스텝 제어	STEP(008)과 SNXT(009)	스텝 정의와 스텝 개시
데이터 제어	PID	PID 제어
블록 프로그램	BPRG(096)	블록 프로그램 개시
장애 진단	FPD(269)	고장점 검출

주1 블록 프로그램은 스텝 래더 프로그램 섹션에서 사용할 수 있습니다.

2 블록 프로그램은 인터록 섹션(IL과 ILC 사이)에서 사용할 수 있습니다. 인터록이 ON이면 블록 프로그램 섹션은 실행되지 않습니다.

3 블록 프로그램 섹션은 다중 점프(JMP0)와 다중 점프 종료(JME0) 사이에서 사용할 수 있습니다.

4 점프 명령어(JMP)와 조건부 점프 명령어(CJP/CJPN)는 블록 프로그램 섹션에서 사용할 수 있습니다. 점프(JMP)와 점프 종료(JME) 명령어, 그리고 조건부 점프(CJP/CJPN)와 점프 종료(JME) 명령어는 짝을 이루어 사용하지 않는 한 블록 프로그램 섹션에서 사용할 수 없습니다. 이들 명령어가 짝을 이루지 않으면 프로그램이 올바르게 실행되지 않습니다.

AI 번역본 For Reference Only

AI 번역본
-For Reference Only-

6

I/O 메모리 영역

본 장에서는 CPU 유닛의 I/O 메모리 영역에 대해 설명합니다. I/O 메모리는 CPU 유닛에 내장된 메모리의 한 종류입니다.

6-1 I/O 메모리 영역 6-2
6-1-1 I/O 메모리 영역 개요 6-2
6-1-2 I/O 메모리 영역 구성 6-4
6-1-3 I/O 메모리 값의 유지 6-7
6-2 I/O 영역 6-9
6-2-1 입력 비트 6-9
6-2-2 출력 비트 6-11
6-3 데이터 링크 영역 6-14
6-4 동기 데이터 리프레시 영역 6-15
6-5 CPU 버스 유닛 영역 6-16
6-6 특수 I/O 유닛 영역 6-17
6-7 펄스 I/O 영역 6-18
6-8 시리얼 PLC 링크 영역 6-19
6-9 DeviceNet 영역 6-20
6-10 EtherCAT 리모트 I/O 영역 6-21
6-11 워크 영역 6-22
6-12 유지 영역 6-23
6-13 보조 기억 영역 6-25
6-14 일시 릴레이 영역 6-26
6-15 데이터 메모리 영역 6-27
6-16 확장 데이터 메모리 영역 6-30
6-17 타이머 영역 6-34
6-18 카운터 영역 6-36
6-19 태스크 플래그 6-37
6-20 인덱스 레지스터 6-38
6-21 데이터 레지스터 6-43
6-22 조건 플래그 6-45
6-23 클록 펄스 6-47

AI 번역본
-For Reference Only-

6-1 I/O 메모리 영역

6-1-1 I/O 메모리 영역 개요

I/O 메모리 영역은 명령어의 오퍼랜드를 사용하여 액세스할 수 있습니다. 다음 표에 I/O 메모리의 영역을 나타냅니다.

영역명	설명	참조
CIO 영역(Core I/O 영역)	CIO 영역의 워드는 각종 유닛과의 I/O 리프레시 등 데이터 교환에 사용됩니다. 유닛에 할당되지 않은 워드는 프로그램에서 워크 워드나 워크 비트로 사용할 수 있습니다. CIO 영역의 어드레스를 지정할 때는 "CIO" 접두어를 입력할 필요가 없습니다.	---
I/O 영역	I/O 영역의 워드는 기본 I/O 유닛의 외부 I/O 단자에 할당됩니다.	6-9
데이터 링크 영역	데이터 링크 영역의 워드는 Controller Link 유닛을 통해 네트워크상의 다른 PLC 유닛과 데이터 링크를 하는 데 사용됩니다.	6-14
동기 데이터 리프레시 영역	동기 데이터 리프레시 영역의 워드는 동기 유닛 동작을 사용할 때 동기 리프레시 데이터로 할당됩니다.	6-15
CPU 버스 유닛 영역	CPU 버스 유닛 영역의 워드는 스테이터스 정보를 전송하기 위해 CPU 버스 유닛에 할당됩니다.	6-16
특수 I/O 유닛 영역	특수 I/O 유닛 영역의 워드는 스테이터스 정보를 전송하기 위해 특수 I/O 유닛에 할당됩니다.	6-17
펄스 I/O 영역	펄스 I/O 영역의 워드는 펄스 I/O 모듈이 접속되어 있을 때 펄스 I/O 기능에 할당됩니다. 이 영역은 CJ2M/CJ3M CPU 유닛에서만 지원됩니다.	6-18
시리얼 PLC 링크 영역	이 영역은 시리얼 PLC 링크에 사용됩니다. 통신 프로그래밍 없이 시리얼 포트를 사용하여 CPU 유닛 간에 데이터를 교환합니다. 이 영역은 CJ2M/CJ3M CPU 유닛에서만 지원됩니다.	6-19
DeviceNet 영역	DeviceNet 영역의 워드는 DeviceNet 리모트 I/O 통신을 위해 슬레이브에 할당됩니다. 이 영역의 할당은 고정되어 있어 변경할 수 없습니다.	6-20
EtherCAT 리모트 I/O 영역	EtherCAT 리모트 I/O 영역의 워드는 EtherCAT 리모트 I/O 통신을 위해 슬레이브에 할당됩니다. 이 영역의 할당은 고정되어 있어 변경할 수 없습니다. 이 영역은 EtherCAT 리모트 I/O 통신 스테이터스를 저장하는 데에도 사용됩니다.	6-21
내부 I/O 영역 *1	이 워드는 프로그램 내에서만 사용할 수 있습니다. 외부 I/O 단자와의 I/O 교환에는 사용할 수 없습니다.	---
워크 영역(W) *1	워크 영역의 워드는 프로그램 내에서만 사용할 수 있습니다.	6-22
유지 영역(H)	유지 영역의 워드는 프로그램 내에서만 사용할 수 있습니다. 이 워드는 PLC의 전원이 ON되거나 동작 모드가 PROGRAM 모드와 RUN 또는 MONITOR 모드 사이에서 전환되어도 내용을 유지합니다.	6-23
보조 기억 영역(A)	보조 기억 영역에는 PLC의 동작을 모니터링하고 제어하는 데 사용되는 플래그와 제어 비트가 있습니다.	6-25
일시 릴레이 영역 (TR) (TR 영역)	TR 영역에는 프로그램 분기의 ON/OFF 상태를 기록하는 비트가 있습니다. TR 비트는 니모닉에서만 사용됩니다.	6-26
데이터 메모리 영역(D) (DM 영역)	DM 영역은 다목적 데이터 영역입니다. DM 영역은 워드 단위 또는 비트 단위로 읽고 쓸 수 있습니다. DM 영역의 워드는 PLC의 전원이 ON되거나 동작 모드가 PROGRAM 모드와 RUN 또는 MONITOR 모드 사이에서 전환되어도 내용을 유지합니다. DM 영역의 일부 워드는 특수 I/O 유닛과 CPU 버스 유닛의 초기 설정을 저장하는 데 사용됩니다.	6-27

-Reference Only-

영역명		설명	참조
확장 데이터 메모리 영역(E) (EM 영역)		EM 영역은 다목적 데이터 영역입니다. EM 영역은 워드 단위 또는 비트 단위로 읽고 쓸 수 있습니다. EM 영역의 워드는 PLC의 전원이 ON되거나 동작 모드가 PROGRAM 모드와 RUN 또는 MONITOR 모드 사이에서 전환되어도 내용을 유지합니다. EM 영역은 뱅크로 나뉘어 있습니다.	6-30
타이머 영역	타이머 완료 플래그 영역 (T)	타이머의 설정 시간이 경과하면 완료 플래그가 ON됩니다.	6-34
	타이머 PV 영역 (T)	타이머의 PV는 타이머가 동작함에 따라 증가 또는 감소합니다.	
카운터 영역	카운터 완료 플래그 영역 (C)	설정값에 도달하거나 0까지 카운트다운되면 완료 플래그가 ON됩니다.	6-36
	카운터 PV 영역 (C)	카운터의 PV는 카운터가 동작함에 따라 증가 또는 감소합니다.	
태스크 플래그 영역(TK)		해당 사이클릭 태스크가 READY 상태이면 태스크 플래그가 ON됩니다.	6-37
인덱스 레지스터(IR)		인덱스 레지스터는 I/O 메모리 워드를 간접 지정하기 위해 PLC 메모리 어드레스(RAM의 절대 메모리 어드레스)를 저장하는 데 사용됩니다.	6-38
데이터 레지스터(DR)		데이터 레지스터는 워드를 간접 지정할 때 인덱스 레지스터의 PLC 메모리 어드레스를 오프셋하는 데 사용됩니다.	6-43
조건 플래그		조건 플래그는 명령어의 실행 결과를 나타내는 에러 플래그, 캐리 플래그 등의 특수 플래그입니다.	6-45
클록 펄스		클록 펄스는 일정한 간격으로 ON/OFF되는 특수 플래그입니다.	6-47

*1 워크 비트를 제공하는 영역은 CIO 영역 내의 내부 I/O 영역과 워크 영역 두 가지입니다. 워크 영역의 워드 비트를 먼저 사용하십시오.

AI 번역본
-For Reference Only-

6-1-2 I/O 메모리 영역 구성

영역		사이즈	범위	외부 I/O 할당	비트 액세스	워드 액세스	액세스		프로그래밍 디바이스에서의 변경	기동 시 또는 모드 변경 시의 상태	강제 비트 상태	참조
							읽기	쓰기				
CIO 영역	I/O 영역	5,120비트 (320워드)	CIO 0~ CIO 319*1	기본 I/O 유닛	OK	OK	OK	OK	OK	클리어*2	OK	6-9
	데이터 링크 영역	3,200비트 (200워드)	CIO 1000~ CIO 1199	데이터 링크 또는 PLC 링크 (조건부)	OK	OK	OK	OK	OK		OK	6-14
	동기 데이터 리프레시 영역	1,536비트 (96워드)	CIO 1200~ CIO 1295	동기 유닛*3	OK	OK	OK	OK	OK		OK	6-15
	CPU 버스 유닛 영역	6,400비트 (400워드)	CIO 1500~ CIO 1899	CPU 버스 유닛 (조건부)	OK	OK	OK	OK	OK		OK	6-16
	특수 I/O 유닛 영역	15,360비트 (960워드)	CIO 2000~ CIO 2959	특수 I/O 유닛 (조건부)	OK	OK	OK	OK	OK		OK	6-17
	펄스 I/O 영역	32비트 (4 워드)	CIO 2960~ CIO 2963	펄스 I/O 모듈*4	OK	OK	OK	OK	OK		OK	6-18
	시리얼 PLC 링크 영역	1,440비트 (90 워드)	CIO 3100~ CIO 3189	링크된 PLC*4	OK	OK	OK	OK	OK		OK	6-19
	DeviceNet 영역	9,600비트 (600워드)	CIO 3200~ CIO 3799	DeviceNet 마스터 (고정 할당) (조건부)	OK	OK	OK	OK	OK		OK	6-20
	EtherCAT 리모트 I/O 유닛용 출력 영역 (노드당 1워드)	1,024점 (64워드)	CIO 3800~ CIO 3863	---	OK	OK	OK	OK	OK		OK	6-21
	EtherCAT 리모트 I/O 유닛용 입력 영역 (노드당 1워드)	1,024점 (64워드)	CIO 3900~ CIO 3963	---	OK	OK	OK	OK	OK		OK	6-21
	EtherCAT 리모트 I/O 통신 스테이터스용 영역 (20워드)	320점 (20워드)	CIO 3980~ CIO 3999	---	OK	OK	OK	OK	OK		OK	6-21
	내부 I/O 영역	3,200비트 (200워드) 34,304비트 (2,144워드)	CIO 1300~ CIO 1499 CIO 4000~ CIO 6143	---	OK	OK	OK	OK	OK		OK	6-22
워크 영역		8,192비트 (512워드)	W000~W511	---	OK	OK	OK	OK	OK	클리어 *2	OK	6-22
유지 영역*5		8,192비트 (512워드)	H000~H511	---	OK	OK	OK	OK	OK	유지	OK	6-23
보조 기억 영역		48,128비트 (3,008워드)	A000~A447	---	OK	OK	OK	No	No	어드레스에 따라 다름	No	6-25
			A448~A959	---	OK	OK	OK	OK	OK			
			A960~ A1471*6	---	OK	OK	OK	OK	OK			
			A10000~ A11535*6	---	OK	OK	OK	No	No			
TR 영역		16비트	TR0~TR15	---	OK	---	OK	OK	No	클리어	No	6-26
DM 영역		32,768워드	D00000~ D32767	---	OK*6	OK	OK	OK	OK	유지	No	6-27
EM 영역		뱅크당 32,768 워드, 최대 25 뱅크 (0~18 hex)*6	E00_0~ E18_32767*8	---	OK*6	OK	OK	OK	OK	유지	설정으로 유효화할 수 있습니다.*8	6-30

AI 번역본 -For Reference Only-

영역	사이즈	범위	외부 I/O 할당	비트 액세스	워드 액세스	액세스		프로그래밍 디바이스에서의 변경	기동 시 또는 모드 변경 시의 상태	비트 상태 강제	참조처
						읽기	쓰기				
타이머 완료 플래그	4,096 비트	T0~T4095	---	가능	---	가능	가능	가능	클리어	가능	6-34
카운터 완료 플래그	4,096 비트	C0~C4095	---	가능	---	가능	가능	가능	유지	가능	6-36
타이머 현재값	4,096 워드	T0~T4095	---	---	가능	가능	가능	가능	클리어	불가*9	6-34
카운터 현재값	4,096 워드	C0~C4095	---	---	가능	가능	가능	가능	유지	불가*10	6-36
태스크 플래그 영역	128 비트	TK000~TK127	---	가능	---	가능	불가	불가	클리어	불가	6-37
인덱스 레지스터*7	16 레지스터	IR0~IR15	---	가능	가능	간접 지정만 가능	특정 명령어만 가능	불가	클리어	불가	6-38
데이터 레지스터*7	16 레지스터	DR0~DR15	---	불가	가능	가능	가능	불가	클리어	불가	6-43
조건 플래그	예: 항상 ON 플래그	CX-Programmer의 글로벌 심볼 테이블에 있는 시스템 심볼 (예: P_On)	---	가능	---	가능	불가	불가	클리어	불가	6-45
펄스 비트	예: 1s 클록 펄스	CX-Programmer의 글로벌 심볼 테이블에 있는 시스템 심볼 (예: P_1s)	---	가능	---	가능	불가	불가	클리어	불가	6-47

*1 CJ3 CPU 유닛은 5,120점(0~319워드), CJ2 CPU 유닛은 2,560점(0~159워드)을 지원합니다.
각 랙의 선두 워드를 변경하여 0~999워드를 외부에 할당해 I/O 영역으로 사용할 수 있습니다. 단, I/O 영역을 확장하더라도 외부 I/O에 할당할 수 있는 최대 점수는 CJ3 CPU 유닛에서 5,120점(320워드), CJ2 CPU 유닛에서 2,560점(160워드)으로 변함이 없습니다.
(선두 워드 설정은 CX-Programmer로 I/O 테이블의 선두 워드를 설정하여 수행할 수 있습니다. 선두 워드의 설정 범위는 CIO 0~CIO 900입니다.)

*2 I/O 메모리 유지 플래그(A500.12)가 ON이면 동작 모드가 변경되어도 메모리 값이 유지됩니다. 또한 PLC 설정에서 기동 시 I/O 메모리 유지 플래그의 상태를 유지하도록(IOM 유지 비트 파라미터) 설정되어 있으면, 전원이 ON될 때에도 메모리 값이 유지됩니다.

*3 이 영역은 CJ2H CPU 유닛에서만 지원됩니다. "동기 유닛"은 동기 유닛 동작을 지원하는 CPU 버스 유닛과 특수 I/O 유닛입니다.

*4 이 영역은 CJ2M/CJ3M CPU 유닛에서만 지원됩니다.

*5 H512~H1535는 펑션 블록 메모리 또는 SFC 메모리 전용으로만 사용하도록 설정할 수 있습니다.

*6 A960~A1471 워드, A10000~A11535 워드, 확장 데이터 메모리의 뱅크 D~18 hex는 CJ2/CJ3 CPU 유닛에서 확장된 영역입니다. 따라서 이 영역과 그 기능은 CJ2/CJ3 CPU 유닛을 지원하지 않는 CPU 버스 유닛, 특수 I/O 유닛, PT, 소프트웨어에서는 사용할 수 없습니다.
CJ2/CJ3 CPU 유닛을 명시적으로 지원하는 CPU 버스 유닛과 특수 I/O 유닛은 다음뿐입니다.

- EtherNet/IP 유닛: CJ1W-EIP21/EIP21S
- 위치 제어 유닛: CJ1W-NC□□4/□81/□82
- 아날로그 입력 유닛: CJ1W-AD042
- 아날로그 출력 유닛: CJ1W-DA042V
- 시리얼 통신 유닛: CJ1W-SCU22, CJ1W-SCU32, CJ1W-SCU42

*7 인덱스 레지스터와 데이터 레지스터는 태스크별로 개별 사용하거나 전 태스크에서 공유하여 사용할 수 있습니다.

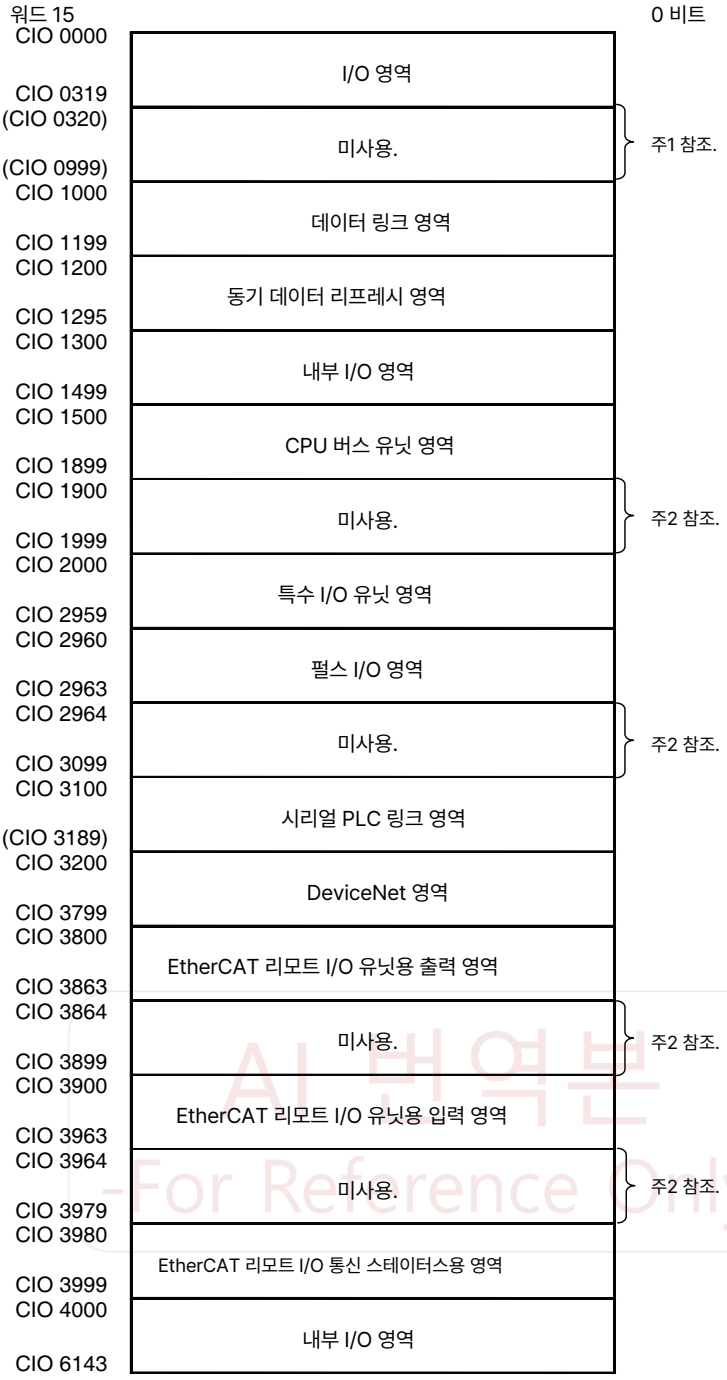
*8 지정한 뱅크와 그 이후의 모든 뱅크에 있는 비트를 강제 세트/리셋할 수 있습니다. (이를 EM 영역 강제 세트/리셋 기능이라고 합니다.)
CJ2H CPU 유닛에서는 자동 어드레스 할당으로 설정된 다음 EM 영역 뱅크의 비트를 강제 세트/리셋할 수 있습니다.
CJ2H-CPU64/65(-EIP): E03_0~E03_32767
CJ2H-CPU65(-EIP): E06_0~E09_32767
CJ2H-CPU67(-EIP): E07_0~E0E_32767
CJ2H-CPU68(-EIP): E11_0~E18_32767

사내 참조용 -For Reference Only-

*9 타이머 현재값은 타이머 완료 플래그를 강제 세트/리셋하여 간접적으로 리프레시할 수 있습니다.
*10 카운터 현재값은 카운터 완료 플래그를 강제 세트/리셋하여 간접적으로 리프레시할 수 있습니다.

● CIO 영역

CIO 영역의 어드레스를 지정할 때 "CIO"라는 약칭을 입력할 필요는 없습니다. CIO 영역은 일반적으로 각종 유닛과의 I/O 리프레시 등 데이터 교환에 사용됩니다. 유닛에 할당되지 않은 워드는 프로그램 내에서 워크 워드 · 워크 비트로만 사용할 수 있습니다.



주1 랙의 선두 워드를 적절히 설정하면 CIO 0320~CIO 0999(CJ2 CPU 유닛은 CIO 160~CIO 0999)를 I/O 워드로 사용할 수 있습니다. 랙의 선두 워드 설정은 CX-Programmer로 I/O 테이블의 랙 선두 어드레스를 설정하여 수행할 수 있습니다. 랙 선두 어드레스의 설정 범위는 CIO 0000~CIO 0900입니다.

2 CIO 영역 중 "미사용"으로 표시된 부분은 프로그래밍에서 워크 비트로 사용할 수 있습니다. 다만 앞으로 기능을 확장할 때 미사용 CIO 영역 비트가 사용될 수 있습니다. 반드시 워크 영역 비트를 먼저 사용하십시오.

AI 번역본
-For Reference Only-

6-1-3 I/O 메모리 값의 유지

CPU 유닛 동작 상태 변화에 따른 I/O 메모리 클리어

CPU 유닛의 동작 상태가 바뀌면 I/O 메모리 값(일부 I/O 메모리 영역 제외)이클리어됩니다.

● I/O 메모리가 클리어되는 변화

- PROGRAM 모드와 RUN 또는 MONITOR 모드 사이의 동작 모드 변경
- 치명 에러로 인한 운전 정지(단, FALS(007) 명령어 실행으로 인한 운전 정지는 제외
- 전원 재투입(전원을 OFF한 후 다시 ON)

● I/O 메모리를 유지하기 위한 설정

CPU 유닛의 동작 상태가 바뀌어도 I/O 메모리를 유지하도록 설정할 수 있습니다.

- 동작 모드 변경 및 치명 에러 시의 I/O 메모리 유지:
  보조 기억 영역의 IOM 유지 비트(A500.12)를 ON하십시오.
- 전원 재투입 시의 I/O 메모리 유지
  보조 기억 영역의 IOM 유지 비트(A500.12)를 ON하고, PLC 설정의 Startup 탭 페이지에 있는 기동 시 유지 영역에서 IOM 유지 비트 체크 박스를 선택하십시오.

안전상의 요점

I/O 영역을 유지하도록 설정하는 경우

CPU 유닛의 동작 모드가 RUN 또는 MONITOR 모드에서 PROGRAM 모드로 바뀌어도 I/O 메모리의 출력 비트는 클리어되지 않습니다(즉, OFF되지 않습니다). PROGRAM 모드로 바뀌기 전의 상태가 유지됩니다. 그 후 PROGRAM 모드에서 RUN 또는 MONITOR 모드로 바뀌면 이전의 I/O 메모리 값이 출력됩니다.

치명 에러(FALS(007) 명령어 실행 포함)로 운전이 정지되면 CPU 유닛의 I/O 메모리 값은 유지되지만, 출력 유닛으로부터의 출력은 모두 OFF됩니다.

AI 번역본
-For Reference Only-

● I/O 메모리 영역별 I/O 메모리 유지 상태

다음 표는 각 I/O 메모리 영역의 유지 상태를 나타냅니다.

영역		CPU 유닛의 동작 모드 변경	치명 에러		전원 투입
			FALS(007) 실행	기타 치명 에러	
CIO 영역 (Core I/O Area)	I/O 영역	유지 가능*2 (설정에 따라 다릅니다.)	유지	유지 가능*2 (설정에 따라 다릅니다.)	유지 가능*3 (설정에 따라 다릅니다.)
	데이터 링크 영역				
	동기 데이터 리프레시 영역*1				
	CPU 버스 유닛 영역				
	특수 I/O 유닛 영역				
	펄스 I/O 영역*4				
	시리얼 PLC 링크 영역*4				
	DeviceNet 영역				
	EtherCAT 리모트 I/O 영역				
	내부 I/O 영역				
워크 영역 (W)		유지 가능*2 (설정에 따라 다릅니다.)	유지	유지 가능*2 (설정에 따라 다릅니다.)	유지 가능*3 (설정에 따라 다릅니다.)
유지 영역 (H)		유지			
보조 기억 영역 (A)		어드레스에 따라 유지 또는 클리어됩니다.			
데이터 메모리 영역 (D)		유지			
확장 데이터 메모리 영역 (E)		유지			
타이머 완료 플래그 (T)		유지 가능*2 (설정에 따라 다릅니다.)	유지	유지 가능*2 (설정에 따라 다릅니다.)	유지 가능*3 (설정에 따라 다릅니다.)
타이머 현재값 (T)		유지 가능*2 (설정에 따라 다릅니다.)	유지	유지 가능*2 (설정에 따라 다릅니다.)	유지 가능*3 (설정에 따라 다릅니다.)
카운터 완료 플래그 (C)		유지			
카운터 현재값 (C)		유지			
태스크 플래그 영역 (TK)		클리어	유지	클리어	클리어
인덱스 레지스터 (IR)		클리어	유지	클리어	클리어
데이터 레지스터 (DR)		클리어	유지	클리어	클리어

AI 번역본 -For Reference Only-

*1 이 영역은 CJ2H CPU 유닛에서만 지원합니다.

*2 이 영역들을 유지하려면 보조 기억 영역의 IOM 유지 비트(A500.12)를 ON하십시오.

*3 보조 기억 영역의 IOM 유지 비트(A500.12)를 ON하고, PLC 설정의 Startup 탭 페이지에 있는 기동 시 유지 영역에서 IOM 유지 비트 체크 박스를 선택하십시오.

*4 이 영역은 CJ2M/CJ3M CPU 유닛에서만 지원합니다.

6-2 I/O 영역

I/O 영역의 어드레스는 워드가 CIO 0~CIO 319, 비트가 CIO 0.00~CIO 319.15입니다.

I/O 영역의 워드는 기본 I/O 유닛의 I/O 단자에 할당됩니다. 워드는 슬롯 위치(좌측에서 우측)와 필요한 워드 수에 따라 기본 I/O 유닛에 할당됩니다. 워드는 연속으로 할당되며 빈 슬롯은 건너뜁니다. 기본 I/O 유닛에 할당되지 않은 I/O 영역의 워드는 프로그램 내에서만 사용할 수 있습니다.

I/O 영역의 비트는 강제 세트 · 강제 리셋할 수 있습니다.

주 각 랙의 선두 워드를 변경하면

0~999워드를 외부에 할당하여 I/O 영역으로 사용할 수 있습니다.
단, I/O 영역을 확장하더라도 외부 I/O에 할당할 수 있는 최대 점수는 CJ3 CPU 유닛이 5,120점(320 워드), CJ2 CPU 유닛이 2,560점(160워드)임에 유의하십시오.

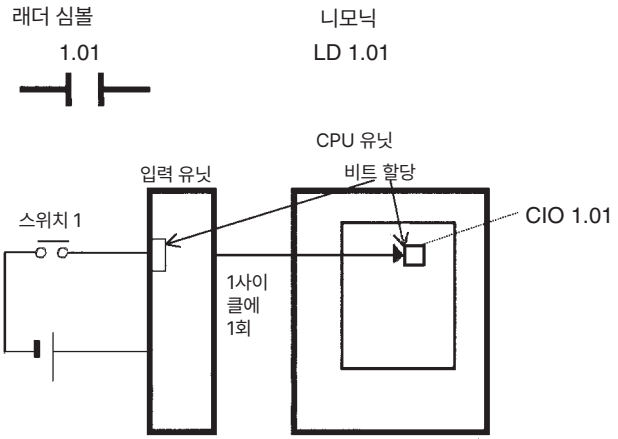
6-2-1 입력 비트

I/O 영역의 비트가 입력 유닛에 할당되면 그 비트를 입력 비트라고 합니다. 입력 비트는 푸시버튼 스위치, 리밋 스위치, 광전 스위치 등 기기의 ON/OFF 상태를 반영합니다.

입력 점의 상태가 PLC에서 리프레시되는 방식은 세 가지입니다: 통상 I/O 리프레시, 즉시 리프레시, IORF(097) 리프레시.

통상 I/O 리프레시

외부 기기의 입력 점 상태는 프로그램 실행 후 1사이클에 1회 읽힙니다. 다음 예에서는 입력 유닛의 입력 단자에 접속된 외부 스위치인 스위치 1에 CIO 1.01이 할당되어 있습니다. 스위치 1의 ON/OFF 상태는 1사이클에 1회 CIO 1.01에 저장됩니다.



-For Reference Only-

즉시 리프레시

명령어 바로 앞에 느낌표를 입력하여 명령어의 즉시 리프레시 변형을 지정하고 그 명령어의 오퍼랜드가 입력 비트 또는 워드인 경우, 아래와 같이 리프레시가 수행됩니다. 이 즉시 리프레시는 1사이클에 1회 수행되는 통상 I/O 리프레시에 더해 수행됩니다.

● 비트 오퍼랜드

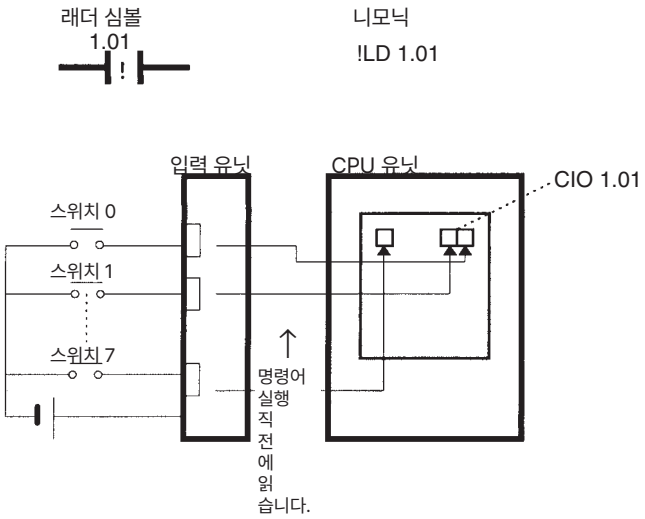
명령어가 실행되기 직전에, 지정한 비트가 포함된 워드에 할당된 16개 I/O 점의 ON/OFF 상태가 PLC로 읽힙니다.

● 워드 오퍼랜드

명령어가 실행되기 직전에, 지정한 워드에 할당된 16개 I/O 점의 ON/OFF 상태가 PLC로 읽힙니다.

● 예

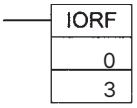
다음 예에서는 입력 유닛의 입력 단자에 접속된 외부 스위치인 스위치 1에 CIO 1.01이 할당되어 있습니다. 스위치 1의 ON/OFF 상태는 !LD 1.01이 실행되기 직전에 읽혀 CIO 1.01에 반영됩니다.



IORF(097) 리프레시

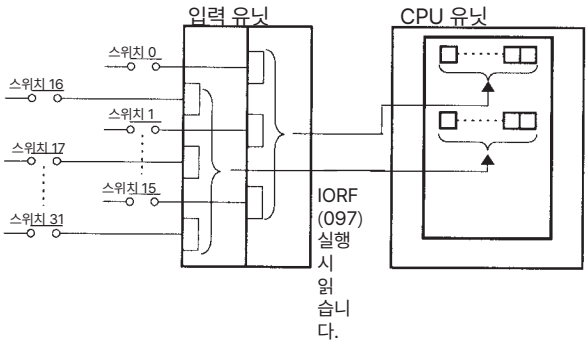
IORF(097)(I/O 리프레시) 명령어가 실행되면 지정한 워드 범위의 입력 비트가 리프레시됩니다. 이 I/O 리프레시는 1사이클에 1회 수행되는 통상 I/O 리프레시에 더해 수행됩니다.

다음 IORF(097) 명령어는 I/O 영역 워드 CIO 0~CIO 3의 모든 I/O 점 상태를 리프레시합니다. 입력 점의 상태는 입력 유닛에서 읽히고, 출력 비트의 상태는 출력 유닛에 쓰입니다.



AI 번역본
-For Reference Only-

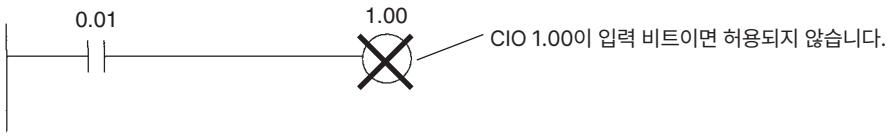
다음 예에서는 CIO 0과 CIO 1에 할당된 입력 점의 상태를 입력 유닛에서 읽습니다. (CIO 2와 CIO 3은 출력 유닛에 할당되어 있습니다.)



입력 비트의 제한 사항

프로그램에서 입력 비트를 a접점 및 b접점 조건으로 사용할 수 있는 횟수에는 제한이 없습니다. 어드레스는 어떤 순서로든 프로그래밍할 수 있습니다.

입력 비트는 OUTPUT 명령어의 오퍼랜드로 사용할 수 없습니다.



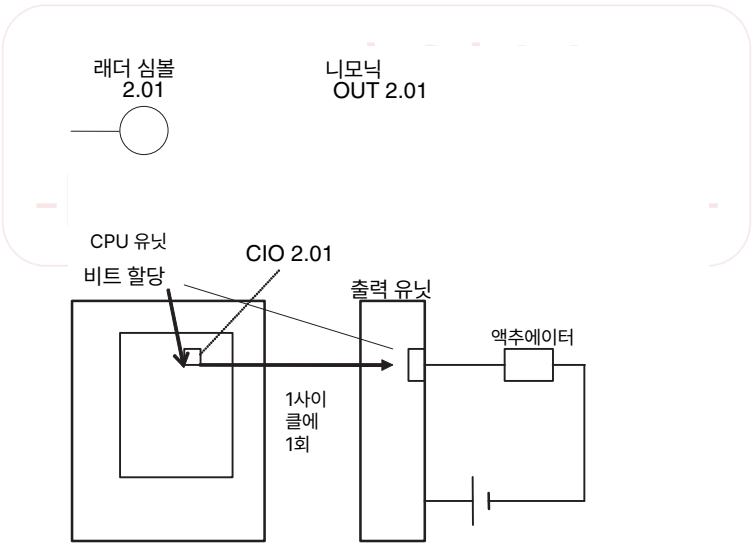
6-2-2 출력 비트

I/O 영역의 비트가 출력 유닛에 할당되면 그 비트를 출력 비트라고 합니다. 출력 비트의 ON/OFF 상태는 액추에이터 등의 기기로 출력됩니다. 출력 비트의 상태가 출력 유닛으로 리프레시되는 방식은 세 가지입니다: 통상 I/O 리프레시, 즉시 리프레시, IORF(097) 리프레시.

통상 I/O 리프레시

출력 비트의 상태는 프로그램 실행 후 1사이클에 1회 외부 기기로 출력됩니다.

다음 예에서는 출력 유닛의 출력 단자에 접속된 외부 기기인 액추에이터에 CIO 2.01이 할당되어 있습니다. CIO 2.01의 ON/OFF 상태는 1사이클에 1회 그 액추에이터로 출력됩니다.



즉시 리프레시

명령어 바로 앞에 느낌표를 입력하여 명령어의 즉시 리프레시 변형을 지정하고, 그 명령어의 오퍼랜드가 출력 비트 또는 워드인 경우, 아래와 같이 리프레시가 수행됩니다. 이 즉시 리프레시는 1사이클에 1회 수행되는 통상 I/O 리프레시에 더해 수행됩니다.

● 비트 오퍼랜드

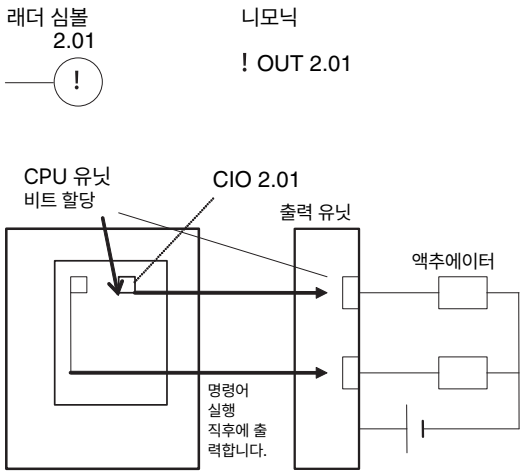
명령어가 실행된 직후에, 지정한 비트가 포함된 워드에 할당된 16개 I/O 점의 ON/OFF 상태가 출력 기기로 출력됩니다.

● 워드 오퍼랜드

명령어가 실행된 직후에, 지정한 워드에 할당된 16개 I/O 점의 ON/OFF 상태가 출력 기기로 출력됩니다.

● 예

다음 예에서는 출력 유닛의 출력 단자에 접속된 외부 기기인 액추에이터에 CIO 2.01이 할당되어 있습니다. CIO 2.01의 ON/OFF 상태는 !OUT 2.01이 실행된 직후에 액추에이터로 출력됩니다.

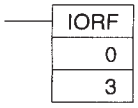


IORF(097) 리프레시

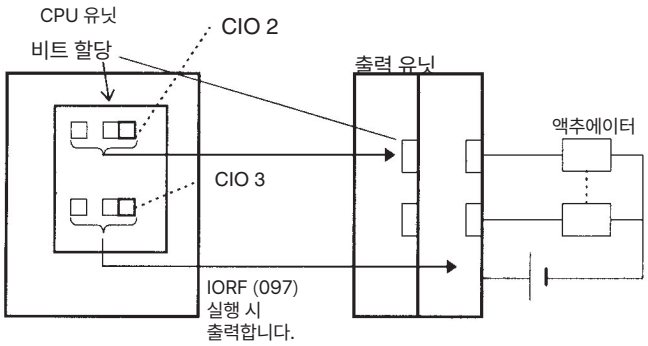
IORF(097)(I/O 리프레시)가 실행되면 지정한 워드 범위에 있는 출력 비트의 ON/OFF 상태가 해당 외부 기기로 출력됩니다. 이 I/O 리프레시는 1사이클에 1회 수행되는 통상 I/O 리프레시에 더해 수행됩니다.

AI 번역부
-For Reference Only-

다음 IORF(097) 명령어는 I/O 영역 워드 CIO 0~CIO 3의 모든 I/O 점 상태를 리프레시합니다. 입력 점의 상태는 입력 유닛에서 읽히고, 출력 비트의 상태는 출력 유닛에 쓰입니다.



이 예에서는 CIO 2와 CIO 3에 할당된 입력 점의 상태가 출력 유닛으로 출력됩니다. (CIO 0과 CIO 1은 입력 유닛에 할당되어 있습니다.)

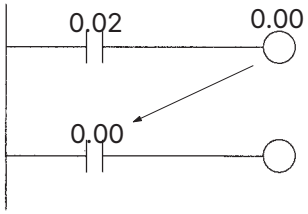


사용상의 주의

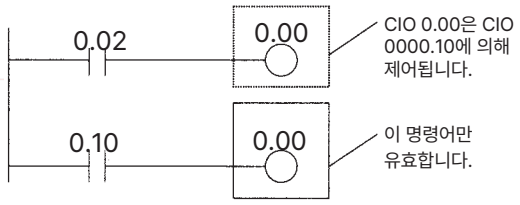
출력 OFF 비트(A500.15)를 ON하면 기본 I/O 유닛과 특수 I/O 유닛의 모든 출력을 OFF할 수 있습니다. 실제 출력이 OFF되더라도 출력 비트의 상태는 영향을 받지 않습니다.

출력 비트의 제한 사항

출력 비트는 어떤 순서로든 프로그래밍할 수 있습니다. 출력 비트는 입력 명령어의 오퍼랜드로 사용할 수 있습니다. 프로그램에서 출력 비트를 a접점 및 b접점 조건으로 사용할 수 있는 횟수에는 제한이 없습니다.



출력 비트는 그 상태를 제어하는 명령어 하나에서만 사용할 수 있습니다. 출력 비트의 상태를 둘 이상의 명령어가 제어하는 경우, 마지막 명령어만 유효합니다.



-For Reference Only-

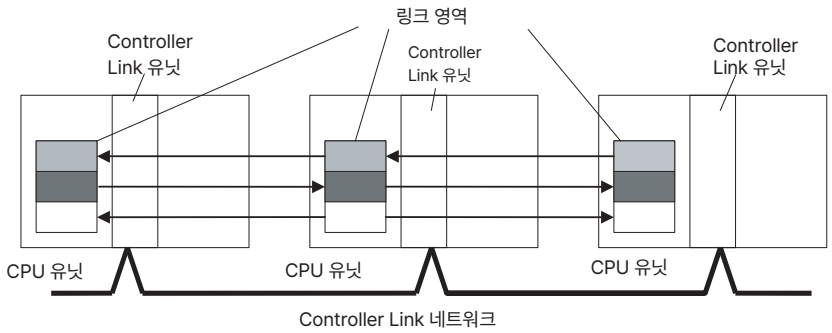
6-3 데이터 링크 영역

데이터 링크 영역의 어드레스는 워드가 CIO 1000~CIO 1199, 비트가 CIO 1000.00~ CIO 1199.15입니다. 데이터 링크 영역의 워드는 Controller Link 네트워크의 데이터 링크 영역으로 LR이 설정된 경우 데이터 링크에 사용됩니다.

데이터 링크는 PLC에 장착된 Controller Link 유닛을 통해 네트워크상의 다른 CPU 유닛의 데이터 링크 영역과 자동으로(프로그램과는 무관하게) 데이터를 공유합니다.

데이터 링크는 자동(각 노드에 동일한 워드 수 사용) 또는 수동으로 생성할 수 있습니다. 사용자가 데이터 링크를 수동으로 정의하면 각 노드에 임의의 워드 수를 할당할 수 있고, 노드를 수신 전용 또는 송신 전용으로 할 수 있습니다. 자세한 내용은 Controller Link 유닛 오퍼레이션 매뉴얼 (Cat. No. W309)을 참조하십시오.

데이터 링크 영역의 워드는 Controller Link 네트워크의 데이터 링크 영역으로 LR이 설정되지 않은 경우, 사용되지 않는 경우에 프로그램에서 사용할 수 있습니다.



데이터 링크 영역의 비트는 강제 세트 · 강제 리셋할 수 있습니다.

AI 번역본
-For Reference Only-

6-4 동기 데이터 리프레시 영역

동기 데이터 리프레시 영역의 어드레스는 워드가 CIO 1200~CIO 1295, 비트가 CIO 1200.00~CIO 1295.15입니다.

동기 데이터 리프레시 영역은 CJ2H CPU 유닛에서 동기 유닛 동작을 사용할 때 CPU 유닛과 동기 유닛 사이에서 데이터를 교환하는 데 사용됩니다. 이 영역은 CJ2H CPU 유닛에서만 지원합니다. 자세한 내용은 10-8-4 동기 데이터 리프레시를 참조하십시오.

AI 번역본
-For Reference Only-

6-5 CPU 버스 유닛 영역

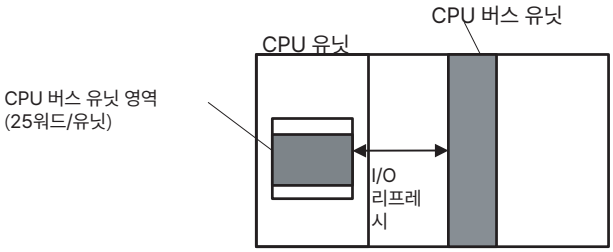
CPU 버스 유닛 영역의 어드레스는 워드가 CIO 1500~CIO 1899, 비트가 CIO 1500.00~ CIO 1899.15입니다.

CPU 버스 유닛 영역의 워드는 유닛의 운전 상태 등 데이터를 전송하기 위해 CPU 버스 유닛에 할당됩니다. 각 유닛에는 유닛 번호 설정에 따라 25워드가 할당됩니다.

CPU 버스 유닛과의 데이터 교환은 다음 시점에 수행됩니다.

・I/O 리프레시 기간 중

・DLNK(226))가 실행될 때



각 CPU 버스 유닛에는 다음 표와 같이 유닛 번호에 따라 25워드가 할당됩니다.

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

AI 번역본
For Reference Only

25워드의 기능은 사용하는 CPU 버스 유닛에 따라 다릅니다. 자세한 내용은 해당 유닛의 오퍼레이션 매뉴얼을 참조하십시오.

CPU 버스 유닛 영역의 워드 중 CPU 버스 유닛에 할당되지 않은 워드는 프로그램 내에서만 사용할 수 있습니다.

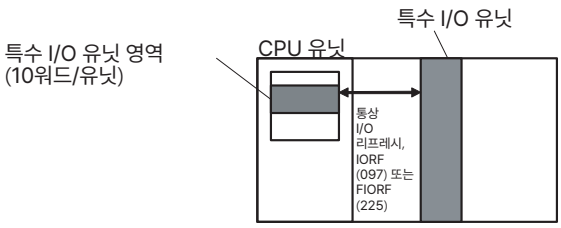
6-6 특수 I/O 유닛 영역

특수 I/O 유닛 영역의 어드레스는 워드가 CIO 2000~CIO 2959, 비트가 CIO 2000.00~ CIO 2959.15입니다.

특수 I/O 유닛 영역의 워드는 각 유닛의 운전 상태 등 데이터를 위해 특수 I/O 유닛에 할당됩니다. 각 유닛에는 유닛 번호 설정에 따라 10워드가 할당됩니다. 유닛 번호 0~95로 최대 96대의 유닛을 사용할 수 있습니다.

특수 I/O 유닛 영역의 워드는 다음 시점에 리프레시됩니다:

- 통상 I/O 리프레시
- IORF (097)가 실행될 때
- FIORF (225)가 실행될 때



각 특수 I/O 유닛에는 다음 표와 같이 유닛 번호에 따라 25워드가 할당됩니다.

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

샘플 번역본
-For Reference Only-

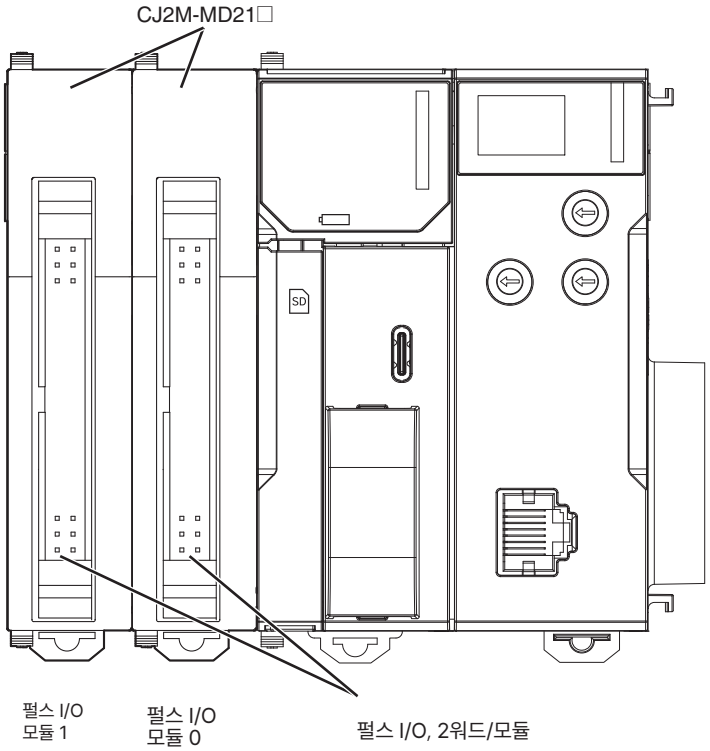
유닛에 할당된 10워드의 기능은 사용하는 특수 I/O 유닛에 따라 다릅니다. 자세한 내용은 해당 유닛의 오퍼레이션 매뉴얼을 참조하십시오.

특수 I/O 유닛 영역의 워드 중 특수 I/O 유닛에 할당되지 않은 워드는 프로그램 내에서만 사용할 수 있습니다.

6-7 펄스 I/O 영역

펄스 I/O 영역의 어드레스는 워드가 CIO 2960~CIO 2963, 비트가 CIO 2960.00~CIO 2963.15입니다.

펄스 I/O 영역의 워드는 CJ2M/CJ3M CPU 유닛에 펄스 I/O 모듈이 접속되었을 때 펄스 I/O 기능에 할당됩니다. 각 펄스 I/O 모듈에는 모듈 번호에 따라 2워드가 할당됩니다. 합계 4워드를 할당할 수 있습니다(펄스 I/O 모듈 0과 펄스 I/O 모듈 1).



할당은 다음과 같습니다:

워드 어드레스	모듈 번호
CIO 2960과 CIO 2961	0
CIO 2962와 CIO 2963	1

펄스 I/O 모듈에 할당되는 워드에 관한 자세한 내용은 CJ2M/CJ3M CPU 유닛 펄스 I/O 모듈 사용자 매뉴얼(Cat. No. W486)을 참조하십시오.

AI 번역본
-For Reference Only-

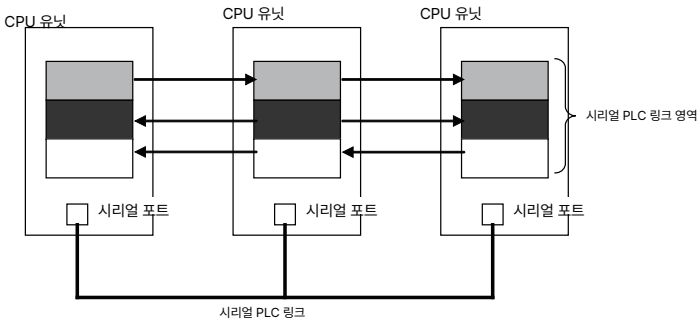
6-8 시리얼 PLC 링크 영역

시리얼 PLC 링크 영역의 어드레스는 워드가 CIO 3100~CIO 3189, 비트가 CIO 3100.00~CIO 3189.15입니다.

시리얼 PLC 링크 영역은 시리얼 PLC 링크에 사용됩니다. 다른 PLC와의 데이터 링크에 사용할 수 있습니다.
시리얼 PLC 링크에서는 통신 프로그래밍 없이 시리얼 포트를 사용하여 CPU 유닛 사이에서 데이터를 교환합니다.
(CJ2M/CJ3M CPU 유닛에서 사용합니다.)

시리얼 PLC 링크 영역의 워드는 폴링 PLC의 PLC 설정에서 수행한 설정에 따라 자동으로 할당됩니다.

- 시리얼 PLC 링크 모드
- 시리얼 PLC 링크 송신 워드 수
- 시리얼 PLC 링크 최대 유닛 번호



시리얼 PLC 링크 영역의 워드 중 시리얼 PLC 링크에 할당되지 않은 워드는 프로그램 내에서 워크 비트로만 사용할 수 있습니다. 시리얼 PLC 링크 영역의 비트는 강제 세트 · 강제 리셋할 수 있습니다.

이 영역의 내용은 다음 경우에 클리어됩니다:

- 동작 모드가 PROGRAM 모드에서 RUN/MONITOR 모드로, 또는 그 반대로 변경될 때.
- PLC 전원을 재투입할 때
- CX-Programmer에서 이 영역을 클리어할 때
- 치명 에러로 운전이 정지될 때(단, FALS(007) 명령어 실행으로 운전이 정지되는 경우에는 이 영역이 클리어되지 않습니다)

● IOM 유지 비트(A500.12)를 ON으로 설정한 경우

IOM 유지 비트(A500.12)가 ON이면 치명 에러가 발생하거나 동작 모드가 PROGRAM 모드에서 RUN 또는 MONITOR 모드로, 또는 그 반대로 변경되어도 이 영역의 내용이 유지됩니다. 다만 전원을 재투입하면 내용이 클리어됩니다.

● IOM 유지 비트(A500.12)를 ON으로 설정하고 IOM 유지 비트 자체를 PLC 설정에서 보호한 경우

전원을 재투입해도 이 영역의 내용이 유지됩니다.

AI 번역본
For Reference Only-

6-9 DeviceNet 영역

DeviceNet 영역의 어드레스는 워드가 CIO 3200~CIO 3799, 비트가 CIO 3200.00~CIO 3799.15입니다.

DeviceNet 영역의 워드는 DeviceNet 리모트 I/O 통신을 위해 슬레이브에 할당됩니다. 데이터는 DeviceNet 유닛을 통해 네트워크상의 슬레이브와 정기적으로(프로그램과는 무관하게) 교환됩니다.

워드는 고정 할당 설정 1, 2, 3에 따라 고정 할당으로 슬레이브에 할당됩니다.
이 고정 영역 중 하나를 선택합니다.

영역	출력 영역 (마스터에서 슬레이브로)	입력 영역 (슬레이브에서 마스터로)
고정 할당 영역 1	CIO 3200~CIO 3263	CIO 3300~CIO 3363
고정 할당 영역 2	CIO 3400~CIO 3463	CIO 3500~CIO 3563
고정 할당 영역 3	CIO 3600~CIO 3663	CIO 3700~CIO 3763

고정 할당으로 리모트 I/O 슬레이브 기능을 사용할 때 DeviceNet 유닛에 다음 워드가 할당됩니다.

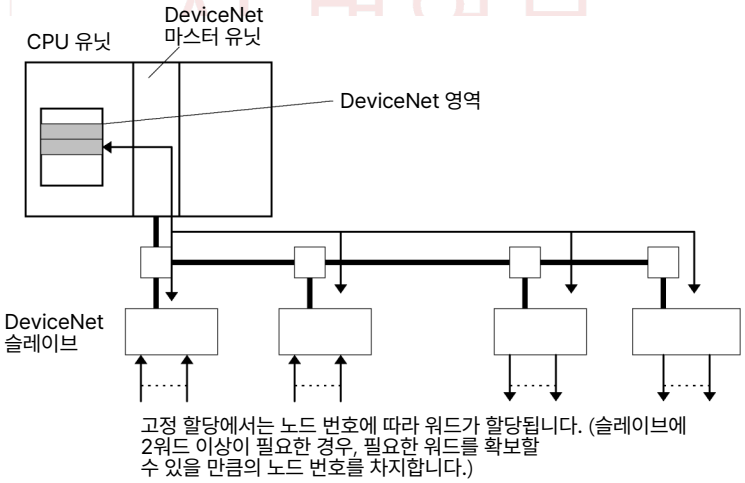
영역	출력 영역 (마스터에서 슬레이브로)	입력 영역 (슬레이브에서 마스터로)
고정 할당 영역 1	CIO 3370	CIO 3270
고정 할당 영역 2	CIO 3570	CIO 3470
고정 할당 영역 3	CIO 3770	CIO 3670

참고

DeviceNet 네트워크에서 I/O를 할당하는 방법은 두 가지입니다: 노드 주소에 따른 고정 할당과 사용자 설정 할당.

- 고정 할당에서는 지정한 고정 할당 영역(1~3)에서 노드 주소에 따라 슬레이브에 워드가 자동으로 할당됩니다.
- 사용자 설정 할당에서는 사용자가 다음 워드에서 슬레이브에 워드를 할당할 수 있습니다.
  CIO 0~CIO 6143
  W0~W511
  H0~H511
  D0~D32767
  E00_0~E0C_32767, 뱅크: 0~18 hex (EM 뱅크 D hex 이후는 할당할 수 없습니다.)

워드 할당에 관한 자세한 내용은 DeviceNet 오퍼레이션 매뉴얼(Cat. No. W267)을 참조하십시오.



6-10 EtherCAT 리모트 I/O 영역

EtherCAT 리모트 I/O 영역의 어드레스 범위는 다음과 같습니다:

・워드는 CIO 3800~CIO 3863, CIO 3900~CIO 3963, CIO 3980~CIO 3999,

・비트는 CIO 3800.00~CIO 3863.15, CIO 3900.00~CIO 3963.15, CIO 3980.00~CIO 3999.15입니다.

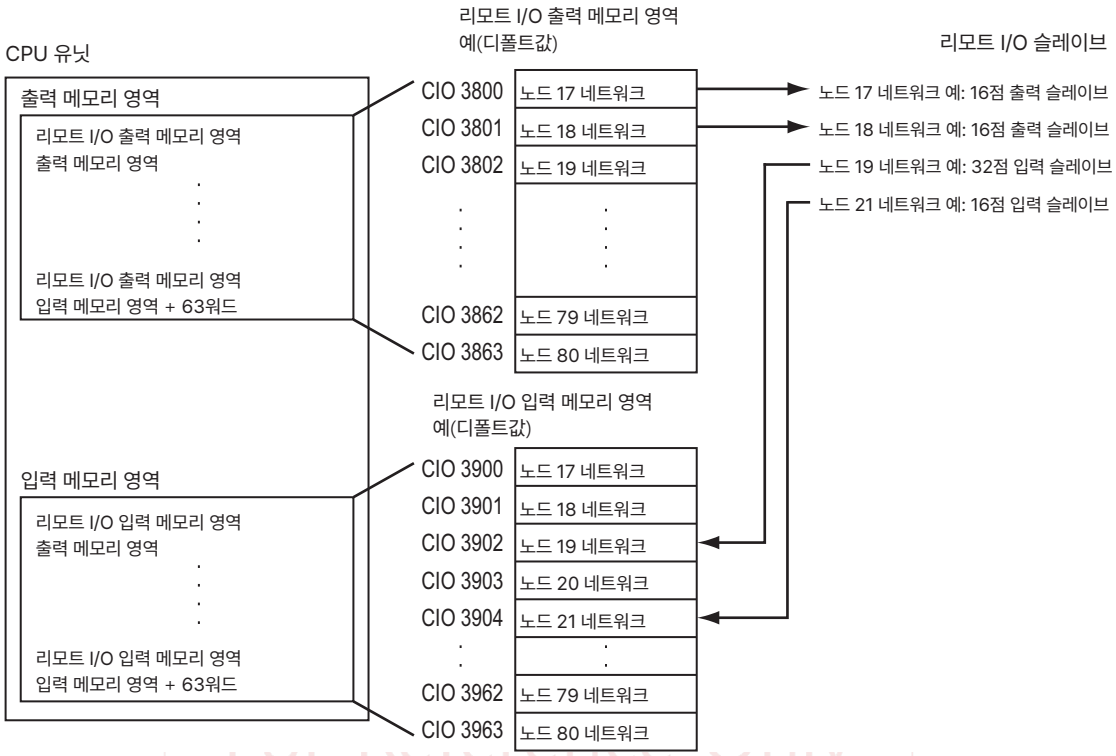
이 영역은 CJ1W-NC□82를 사용하여 리모트 I/O 통신을 수행할 때 다음 두 가지 목적으로 사용됩니다:

・CPU 유닛과 CPU 버스 유닛 사이의 데이터 교환,

・리모트 I/O 통신 스테이터스의 저장.

CJ1W-NC□82의 공통 파라미터 설정으로 사용할 어드레스 범위를 변경할 수 있습니다.

다음은「 고정 할당 」을 사용할 때 리모트 I/O 출력 영역과 리모트 I/O 입력 영역을 사용하는 예입니다. 자세한 내용은 위치 제어 유닛 오퍼레이션 매뉴얼(Cat. No. W487)의 설정 데이터를 참조하십시오.



For Reference Only

리모트 I/O 통신 스테이터스 영역은 디폴트로 CIO 3980~CIO 3999(20워드)의 어드레스에 할당됩니다. 자세한 내용은 위치 제어 유닛 오퍼레이션 매뉴얼 (Cat. No. W487)의 설정 데이터를 참조하십시오.

6-11 워크 영역

워크 영역은 512워드로 구성되며, 어드레스는 워드가 W0~W511, 비트가 W0.00~ W511.15입니다. 이 영역의 워드는 외부 I/O 단자와의 I/O에는 사용할 수 없습니다. 프로그램 내부에서 사용할 수 있습니다.

CIO 영역의 내부 I/O 영역(CIO 1300~CIO 1499 및 CIO 4000~CIO 6143)에도 프로그램에서 사용할 수 있는 미사용 워드가 있습니다. 다만 CIO 영역의 미사용 워드는 앞으로의 CPU 유닛 버전에서 새 기능에 할당될 수 있습니다. 내부 I/O 영역의 워드를 사용하기 전에 반드시 워크 영역의 사용 가능한 워드를 먼저 사용하십시오.

AI 번역본
-For Reference Only-

6-12 유지 영역

유지 영역의 어드레스는 워드가 H000~H511, 비트가 H000.00~H511.15입니다.
이 워드는 프로그램 내에서만 사용할 수 있습니다. 이 영역의 비트는 전원이 OFF되거나 동작 모드가 PROGRAM 모드에서 RUN 또는 MONITOR 모드로, 또는 그 반대로 변경되어도 상태를 유지합니다.

유지 영역 비트는 프로그램에서 어떤 순서로든 사용할 수 있으며, a접점 또는 b접점 조건으로 필요한 만큼 사용할 수 있습니다.

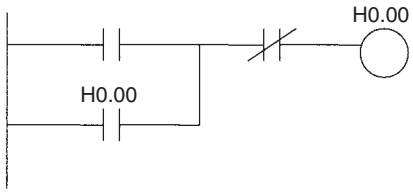
● 유지 영역의 초기화

유지 영역의 데이터는 PLC 전원을 재투입하거나 CPU 유닛의 동작 모드가 PROGRAM 모드에서 RUN 또는 MONITOR 모드로, 또는 그 반대로 변경되어도 클리어되지 않습니다.

유지 영역 비트가 IL(002)와 ILC(003) 사이에 프로그래밍되어 있고 IL(002)의 실행 조건이 OFF이면 그 비트는 클리어됩니다. IL(002)의 실행 조건이 OFF일 때도 비트를 ON으로 유지하려면 IL(002) 직전에 SET 명령어로 그 비트를 ON하십시오.

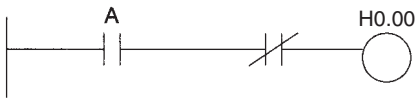
● 자기 유지 비트

자기 유지 비트를 유지 영역 비트로 프로그래밍하면 전원을 재투입해도 그 자기 유지 비트는 클리어되지 않습니다.



자기 유지 비트에 유지 영역 비트를 사용하지 않으면 전원 재투입 시 그 비트가 OFF되고 자기 유지 비트가 클리어됩니다.

다음 그림과 같이 유지 영역 비트를 사용하되 자기 유지 비트로 프로그래밍하지 않으면, 전원 재투입 시 실행 조건 A에 의해 그 비트가 OFF됩니다.

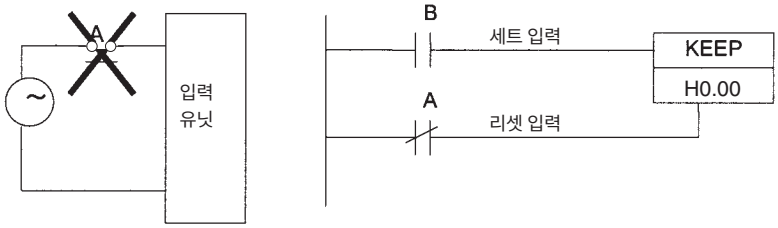


유지 영역의 H512~H1535 워드는 펑션 블록 또는 SFC 프로그램용으로 설정할 수 있습니다.이 워드는 사용자 프로그램에서 명령어 오퍼랜드로 지정할 수 없습니다.

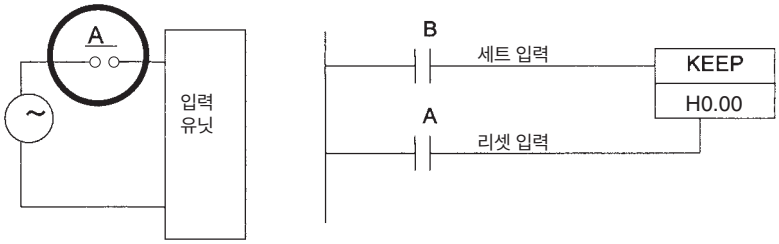
● 주의사항

유지 영역 비트를 KEEP(011) 명령어에 사용하는 경우, 입력 기기가 AC 전원을 사용한다면 리셋 입력에 b접점 조건을 절대 사용하지 마십시오. 전원이 OFF되거나 일시적으로 차단되면 PLC의 내부 전원보다 먼저 입력이 OFF되어 유지 영역 비트가 리셋됩니다.

AI 번역본
For Reference Only



대신 아래와 같은 구성을 사용하십시오.



비트 어드레스를 사용하는 순서나 프로그래밍할 수 있는 N.C. 또는 N.O. 조건의 수에는 제한이 없습니다.

AI 번역본
-For Reference Only-

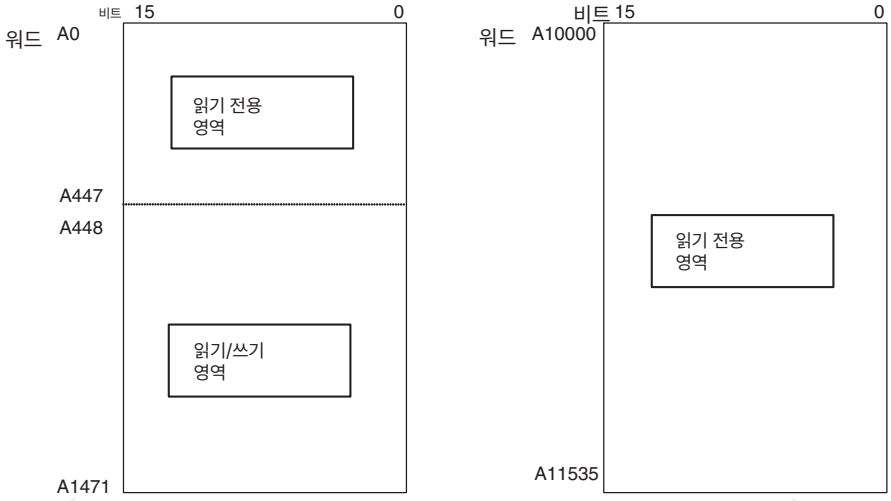
6-13 보조 기억 영역

보조 기억 영역의 어드레스는 워드가 A0~A1471 및 A1000~A11535, 비트가 A0.00~ A1471.15 및 A1000.00~A11535.15입니다. 이 워드들은 동작을 모니터 · 제어하기 위한 플래그와 제어 비트로 미리 할당되어 있습니다.

일부 워드나 비트는 시스템이 자동으로 설정하고, 나머지는 사용자가 설정 · 조작합니다. 보조 기억 영역에는 자기진단으로 설정되는 에러 플래그, 초기 설정, 제어 비트, 스테이터스 데이터가 포함됩니다. (보조 기억 영역 비트 · 워드의 기능에 관한 정보는 A-3 보조 기억 영역을 참조하십시오.)

A0~A447 및 A10000~A11535는 읽기 전용이지만, A448~A1471은 프로그램이나 CX-Programmer에서 읽고 쓸 수 있습니다.
A960~A1471과 A10000~A11535는 CJ2/CJ3 CPU 유닛에서 보조 기억 영역을 확장하기 위해 추가되었습니다. 이 워드들은 CJ2/CJ3 CPU 유닛을 명시적으로 지원하지 않는 CPU 버스 유닛, 특수 I/O 유닛, PT, 지원 소프트웨어에서는 액세스할 수 없습니다.
CJ2/CJ3 CPU 유닛을 명시적으로 지원하는 CPU 버스 유닛과 특수 I/O 유닛은 다음뿐입니다.

- EtherNet/IP 유닛: CJ1W-EIP21/EIP21S
- 위치 제어 유닛: CJ1W-NC□□4/□81/□82
- 아날로그 입력 유닛: CJ1W-AD042
- 아날로그 출력 유닛: CJ1W-DA042V
- 시리얼 통신 유닛: CJ1W-SCU22, CJ1W-SCU32, CJ1W-SCU42



보조 기억 영역에 데이터를 쓰기 위해 CX-Programmer에서 다음 조작을 수행할 수 있습니다:
프로그래밍 어드레스를 모니터할 때 현재값 변경(설정값 대화 상자), 또는 PLC 데이터 테이블을 편집한 후 PLC로 데이터 전송. 또한 보조 기억 영역의 비트에는 온라인 세트/리셋 조작을 사용할 수 없습니다. CX-Programmer 오퍼레이션 매뉴얼(Cat. No. W414)을 참조하십시오.

OMRON
For Reference Only-

사용상의 주의

앞으로의 CPU 유닛 업그레이드에서 보조 기억 영역의 미정의 워드나 비트에 기능이 할당될 가능성이 있습니다. 보조 기억 영역의 미정의 워드나 비트를 사용자 프로그램의 워크 워드나 워크 비트로 사용하지 마십시오.

6-14 일시 릴레이 영역

TR 영역은 16비트로 구성되며 어드레스는 TR0~TR15입니다. TR 비트는 출력 분기가 여러 개 있고 인터록을 사용할 수 없을 때 유용합니다.

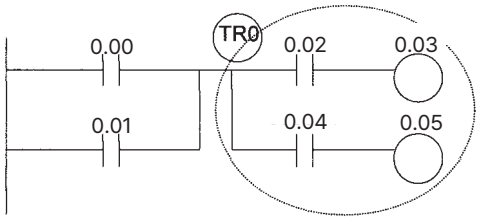
TR 비트는 같은 명령어 블록에서 동일한 TR 비트를 두 번 사용하지 않는 한, 필요한 횟수만큼 필요한 순서로 사용할 수 있습니다.

TR 비트는 OUT 명령어와 LD 명령어에서만 사용할 수 있습니다. OUT 명령어(OUT TR0~OUT TR15)는 분기점의 ON OFF 상태를 저장하고, LD 명령어는 저장된 분기점의 ON OFF 상태를 불러옵니다.

TR 비트의 상태는 CX-Programmer로 변경할 수 없습니다.

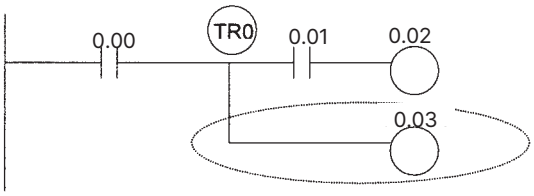
● 적용 예

이 예에서는 두 개의 출력이 분기점에 직접 접속되어 있을 때 TR 비트를 사용합니다.



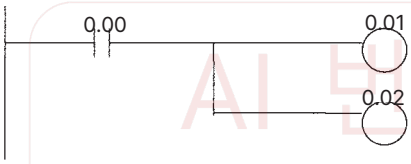
명령어	오퍼랜드
LD	0.00
OR	0.01
OUT	TR 0
AND	0.02
OUT	0.03
LD	TR 0
AND	0.04
OUT	0.05

이 예에서는 별도의 실행 조건 없이 출력이 분기점에 접속되어 있을 때 TR 비트를 사용합니다.

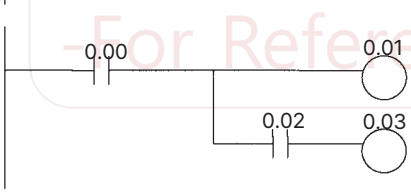


명령어	오퍼랜드
LD	0.00
OUT	TR 0
AND	0.01
OUT	0.02
LD	TR 0
OUT	0.03

분기점 이후에 실행 조건이 없거나 명령어 블록의 마지막 줄에만 실행 조건이 있는 경우에는 TR 비트가 필요하지 않습니다.



명령어	오퍼랜드
LD	0.00
OUT	0.01
OUT	0.02



명령어	오퍼랜드
LD	0.00
OUT	0.01
AND	0.02
OUT	0.03

AI 번역본
-For Reference Only-

6-15 데이터 메모리 영역

DM 영역의 어드레스는 워드가 D0~D32767입니다. 이 데이터 영역은 일반적인 데이터의 저장과 조작에 사용되며 워드 단위 또는 비트 단위로 액세스할 수 있습니다.

DM 영역의 데이터는 PLC 전원을 재투입하거나 CPU 유닛의 동작 모드가 PROGRAM 모드에서 RUN/MONITOR 모드로, 또는 그 반대로 변경되어도 유지됩니다.

DM 영역의 비트는 강제 세트 · 강제 리셋할 수 없습니다.

간접 지정

DM 영역의 워드는 두 가지 방법으로 간접 지정할 수 있습니다: 바이너리 모드와 BCD 모드.

● 바이너리 모드 지정 (@D)

DM 어드레스 앞에 "@" 문자를 입력하면 그 DM 워드의 내용이 바이너리로 취급되어, 명령어는 그 바이너리 어드레스의 DM 워드를 대상으로 동작합니다. DM 영역 전체(D0~ D32767)를 16진수 0~7FFF로 간접 지정할 수 있습니다.



● BCD 모드 지정 (*D)

DM 어드레스 앞에 "*" 문자를 입력하면 그 DM 워드의 내용이 BCD로 취급되어, 명령어는 그 BCD 어드레스의 DM 워드를 대상으로 동작합니다. DM 영역의 일부(D0~ D09999)만 BCD 값 0~9999로 간접 지정할 수 있습니다.

예: [MOV #1234 *D100]



AI 번역본
-For Reference Only-

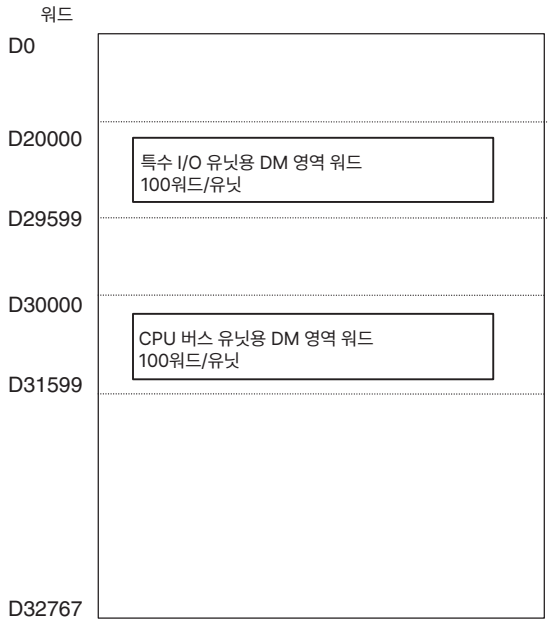
특수 I/O 유닛과 CPU 버스 유닛에 대한 DM 영역 할당

DM 영역의 일부는 유닛 초기 설정 등의 기능을 위해 특수 I/O 유닛과 CPU 버스 유닛에 할당됩니다. 해당 유닛을 PLC에서 사용하지 않는 경우 이 워드는 일반적인 데이터 저장에 사용할 수 있습니다.

데이터 전송 타이밍은 이 유닛들마다 다르지만, 다음 세 시점 중 어느 하나에 수행될 수 있습니다.

- PLC 전원이 ON되거나 유닛이 재기동될 때 데이터 전송
- 1사이클에 1회 데이터 전송
- 필요할 때 데이터 전송

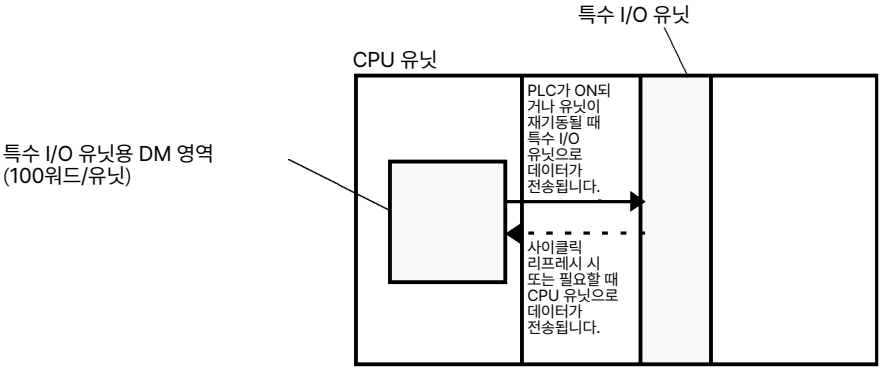
데이터 전송 타이밍에 관한 자세한 내용은 해당 유닛의 오퍼레이션 매뉴얼을 참조하십시오.



AI 번역본
-For Reference Only-

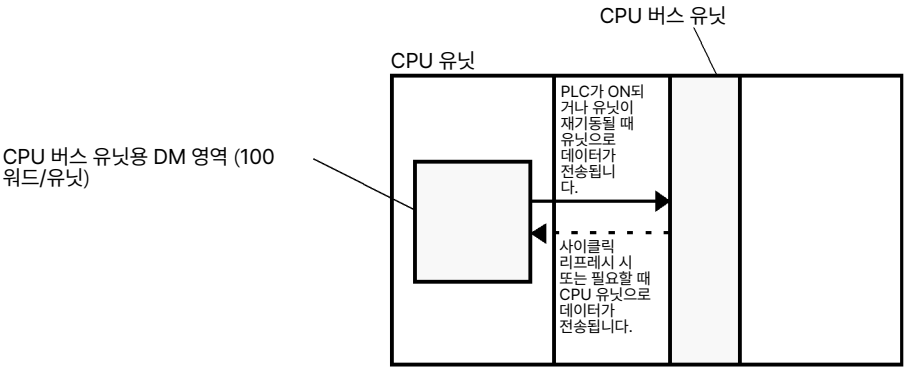
● 특수 I/O 유닛 (D20000~D29599)

각 특수 I/O 유닛에는 100워드가 할당됩니다(유닛 번호 0~95 기준). 이 워드들의 기능에 관한 자세한 내용은 해당 유닛의 오퍼레이션 매뉴얼을 참조하십시오.



● CPU 버스 유닛 (D30000~D31599)

각 CPU 버스 유닛에는 100워드가 할당됩니다(유닛 번호 0~F 기준). 이 워드들의 기능에 관한 자세한 내용은 해당 유닛의 오퍼레이션 매뉴얼을 참조하십시오. 시리얼 통신 유닛 등 일부 CPU 버스 유닛에서는 초기 설정을 CPU 유닛의 파라미터 영역에도 등록해야 하며, 이 데이터는 프로그래밍 디바이스로 등록할 수 있습니다.



AI 번역본
-For Reference Only-

6-16 확장 데이터 메모리 영역

EM 영역의 어드레스는 워드가 E0_0~E18_32767입니다. EM 영역은 24개 뱅크(0~18 hex)로 나뉩니다. 어드레스에서 "E" 뒤의 16진수가 뱅크를 나타냅니다.

EM 영역은 일반적인 데이터의 저장과 조작에 사용되며 워드 단위 또는 비트 단위로 액세스할 수 있습니다.

EM 영역의 데이터는 PLC 전원을 재투입하거나 CPU 유닛의 동작 모드가 PROGRAM 모드에서 RUN/MONITOR 모드로, 또는 그 반대로 변경되어도 유지됩니다.

일반 데이터 영역 외에, EM 영역을 자동 어드레스 할당, 트레이스 메모리, 파일 메모리로 사용하도록 설정할 수 있습니다.
EM 영역의 뱅크 D~18 hex(E0D_0~E18_32767)는 CJ2/CJ3 CPU 유닛에서 EM 영역을 확장하기 위해 추가되었습니다. 이 뱅크들은 CJ2/CJ3 CPU 유닛을 명시적으로 지원하지 않는 CPU 버스 유닛, 특수 I/O 유닛, PT, 지원 소프트웨어에서는 액세스할 수 없습니다.
CJ2/CJ3 CPU 유닛을 명시적으로 지원하는 CPU 버스 유닛과 특수 I/O 유닛은 다음뿐입니다.

- EtherNet/IP 유닛: CJ1W-EIP21/EIP21S
- 위치 제어 유닛: CJ1W-NC□□4/□81/□82
- 아날로그 입력 유닛: CJ1W-AD042
- 아날로그 출력 유닛: CJ1W-DA042V
- 시리얼 통신 유닛: CJ1W-SCU22, CJ1W-SCU32, CJ1W-SCU42

자동 어드레스 할당

심볼에 어드레스를 자동으로 할당하는 데에는 EM 영역을 사용할 것을 권장합니다. CJ2H CPU 유닛에서는 자동 어드레스 할당을 사용하는 경우 다음 EM 영역 뱅크의 비트를 강제 세트/리셋할 수 있습니다.

형식	자동 어드레스 할당 사용 시 비트를 강제 세트/리셋할 수 있는 워드
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)	E11_0~E18_32767

자동 어드레스 할당을 사용하는 순서는 5-5-8 심볼에 대한 자동 어드레스 할당을 참조하십시오.

AI 번역본
-For Reference Only-

참고

유닛 버전 1.2 이후의 CPU 유닛에서는 모든 EM 영역 뱅크의 비트를 강제 세트 · 강제 리셋할 수 있습니다.

CJ2M/CJ3M CPU 유닛에서는 자동 어드레스 할당을 사용하지 않고 EM 영역 강제 세트/리셋 기능용으로 지정한 뱅크에 대해서만 EM 영역의 비트를 강제 세트/리셋할 수 있습니다.

형식	강제 세트/리셋 가능 뱅크
CJ2M-CPU□1/□2/□3	E0
CJ2M-CPU□4/□5	E0~E3
CJ3M-CPU□1/□2/□3	E0
CJ3M-CPU□4/□5	E0~E09

EM 영역 강제 세트/리셋 기능

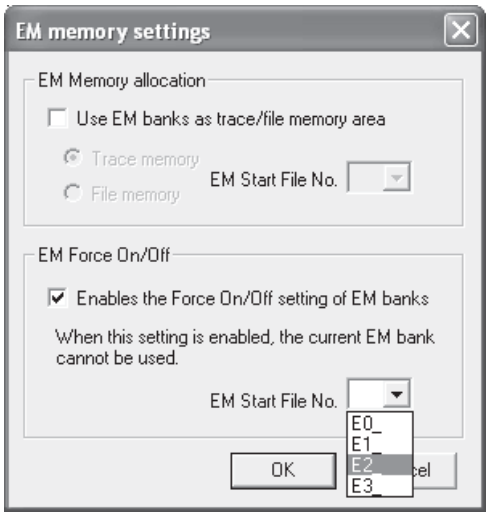
CX-Programmer에서 파라미터를 설정하면 지정한 EM 영역 뱅크와 그 이후의 모든 EM 영역 뱅크에서 비트를 강제 세트/리셋할 수 있습니다. (이 기능은 디폴트 설정에서는 무효입니다.)
이에 따라 자동 어드레스 할당 영역을 사용하지 않아도 EM 영역의 비트를 강제 세트/리셋할 수 있습니다. 이 기능은 모든 EM 영역 뱅크에 설정할 수 있습니다.

● 설정 순서

1. **CX-Programmer에서 PLC - Memory Allocate - EM Memory Settings를 선택합니다.***

   EM Memory Settings 대화 상자가 표시됩니다.

2. 체크 박스를 선택하여 EM 영역 강제 세트/리셋 기능을 유효로 하고 선두 EM 영역 뱅크를 설정합니다.



3. CPU 유닛에 온라인으로 접속하고 사용자 프로그램을 전송합니다.

* EM 영역 강제 세트/리셋 기능을 사용하려면 CX-Programmer 버전 8.3 이상이 필요합니다.

사용상의 주의

- EM 영역 강제 세트/리셋 기능이 유효한 경우, EM 영역 강제 세트/리셋 기능용으로 지정한 뱅크는 현재 EM 영역 뱅크로 사용할 수 없습니다.
- 트레이스 메모리 또는 파일 메모리로 사용 중인 EM 영역 뱅크에는 EM 영역 강제 세트/리셋 기능을 사용할 수 없습니다.
- 강제 세트 및 강제 리셋은 MONITOR 또는 PROGRAM 모드에서 가능합니다. RUN 모드에서는 실행할 수 없습니다.
  비트를 강제 세트/리셋하는 순서는 10-7-1 강제 세트/리셋을 참조하십시오.

AI 번역본
For Reference Only

파일 메모리

파일 메모리는 CPU 유닛이 사용하는 파일을 저장하는 데 사용할 수 있습니다. 자세한 내용은 Section 7 파일 조작을 참조하십시오.

트레이스 메모리

트레이스 메모리는 데이터 트레이스를 위해 샘플링한 데이터에 사용됩니다. 자세한 내용은 10-7-6 데이터 트레이스를 참조하십시오.

EM 어드레스의 직접 지정

EM 어드레스를 지정하는 방법은 두 가지입니다: 뱅크와 어드레스를 동시에 지정하는 방법과 현재 뱅크의 어드레스를 지정하는 방법입니다. 일반적으로 뱅크와 어드레스를 동시에 지정할 것을 권장합니다.

● 뱅크 · 어드레스 지정

이 방법에서는 어드레스 바로 앞에 뱅크 번호를 지정합니다. 예를 들어 E2_10은 뱅크 2의 어드레스 E10을 지정합니다.

● 현재 뱅크 어드레스 지정

이 방법에서는 어드레스만 지정합니다. 예를 들어 E10은 현재 뱅크의 어드레스 E10을 지정합니다.

- IOM 유지 비트(A500.12)가 ON이 아닌 한, 동작 모드가 PROGRAM 모드에서 RUN/MONITOR 모드로 변경되면 현재 뱅크는 0으로 리셋됩니다.
- 다른 뱅크의 데이터에 액세스하려면 EMBC(281)로 현재 뱅크를 변경할 수 있습니다.
- A301에는 현재 EM 뱅크 번호가 저장됩니다.
- 프로그램이 사이클릭 태스크를 진행해도 현재 뱅크는 변경되지 않습니다. 예를 들어 태스크 1에서 현재 EM 뱅크를 뱅크 2로 변경하면 태스크 2에서도 현재 EM 뱅크는 뱅크 2입니다.
  인터럽트 태스크에서 현재 뱅크가 변경된 경우에는 (호출한 사이클릭 태스크의) 원래 값으로 되돌아갑니다.

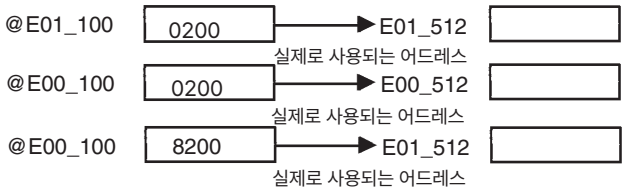
AI 번역본
-For Reference Only-

EM 어드레스의 간접 지정

EM 영역의 워드는 두 가지 방법으로 간접 지정할 수 있습니다: 바이너리 모드와 BCD 모드.

● 바이너리 모드 지정 (@E)

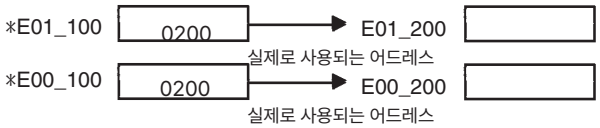
EM 어드레스 앞에 "@" 문자를 입력하면 그 EM 워드의 내용이 다른 EM 영역 어드레스를 나타내는 바이너리 값으로 취급됩니다. 명령어는 같은 뱅크 또는 다음 뱅크에서 그 바이너리 어드레스의 EM 워드를 대상으로 동작합니다. 같은 EM 뱅크의 모든 워드(E0~ E32767)는 16진수 0~7FFF로, 다음 EM 뱅크의 워드(E0~E32767)는 16진수 8000~FFFF로 간접 지정할 수 있습니다.



● BCD 모드 지정 (*E)

EM 어드레스 앞에 "*" 문자를 입력하면 그 EM 워드의 내용이 다른 EM 어드레스를 나타내는 BCD 값으로 취급됩니다.

BCD 값이 0~9999이면 최종 어드레스는 같은 뱅크 내에 있습니다.



● EM 영역을 파일 메모리 또는 트레이스 메모리로 변환

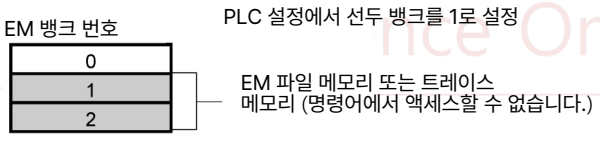
CX-Programmer로 EM 영역의 뱅크를 지정하면 지정한 뱅크부터 EM 영역 끝까지의 모든 뱅크를 파일 메모리 또는 트레이스 메모리로 변환할 수 있습니다.

파일 메모리:
파일 메모리는 메모리 카드에 파일을 저장하는 것처럼 파일을 저장하는 데 사용할 수 있습니다. 최대 13개 뱅크를 파일 메모리로 변환할 수 있습니다.

트레이스 메모리:
데이터 트레이스의 결과를 트레이스 메모리에 저장할 수 있으며, 이로써 1회 트레이스에서 저장할 수 있는 결과의 양이 늘어납니다.

EM 영역 뱅크를 파일 메모리 또는 트레이스 메모리로 변환하면 그 뱅크는 사용자 프로그램의 명령어에서 액세스할 수 없습니다. 액세스를 시도하면 부정 액세스 에러가 발생합니다.

AI 번역본
-Reference Only-



6-17 타이머 영역

타이머 번호 T0~T4095로 최대 4,096개의 타이머를 사용할 수 있습니다. 타이머 데이터 영역은 두 가지입니다: 타이머 완료 플래그 영역과 타이머 현재값(PV) 영역.

・타이머 완료 플래그 (T)
완료 플래그에 액세스할 때는 타이머 번호를 사용합니다. 완료 플래그는 타이머의 설정 시간이 경과하면 ON됩니다.

・타이머 현재값 (T)
타이머의 현재값(16비트 워드)을 읽고 쓸 때도 타이머 번호를 사용합니다. 현재값은 타이머 동작에 따라 증가하거나 감소합니다.

비트 데이터가 필요한 오퍼랜드에 타이머 번호를 사용하면 그 타이머 번호는 완료 플래그에 액세스합니다. 워드 데이터가 필요한 오퍼랜드에 타이머 번호를 사용하면 그 타이머 번호는 현재값에 액세스합니다.

다음 명령어는 모두 동일한 타이머 번호를 사용합니다: 100ms 타이머 (TIM/TIMX(550)), 10ms 타이머 (TIMH(015)/TIMHX(551)), 1ms 타이머 (TMHH(540)/TMHHX(552)), 0.1ms 타이머 (TIMU(541)/TIMUX(556)), 0.01ms 타이머 (TMUH(544)/TMUHX(557)), 적산 타이머 (TTIM(087)/TTIMX(555)), 타이머 대기 (TIMW(813)/TIMWX(816)), 고속 타이머 대기 (TMHW(815)/TMHWX(817)).

사용상의 주의

두 개의 타이머 명령어에 동일한 타이머 번호를 사용하는 것은 권장하지 않습니다. 동시에 시간을 계측하면 타이머가 올바르게 동작하지 않기 때문입니다. (둘 이상의 타이머 명령어가 동일한 타이머 번호를 사용하면 프로그램 체크 시 에러가 발생하지만, 그 명령어들이 같은 사이클에서 실행되지 않는 한 타이머는 동작합니다.)

AI 번역본
-For Reference Only-

다음 표는 타이머 현재값과 완료 플래그가 리셋되는 시점을 나타냅니다.

명령어 이름*1	현재값과 완료 플래그에 대한 영향			점프 및 인터록 시의 동작	
	모드 변경 (PROGRAM에서 RUN 또는 MONITOR로, 또는 그 반대)*2	PLC 기동*3	TRSET(549)	점프 (JMP-JME) 또는 WAIT 상태의 태스크*4	인터록 (IL-ILC)
100ms 타이머: TIM/TIMX(550)	현재값 → 0 플래그 → OFF	현재값 → 0 플래그 → OFF	현재값 → 유지 플래그 → OFF	동작 중인 타이머의 현재값이 리프레시됩니다.	현재값 → 설정값 (설정값으로 리셋.) 플래그 → OFF
10ms 타이머: TIMH(015)/TIMHX(551)					
1ms 타이머: TMHH(540)/TMHHX(552)					
0.1ms 타이머: TIMU(541)/TIMUX(556)*1					
0.01ms 타이머: TMUH(544)/TMUHX(557)*1					
적산 타이머: TTIM(087)/TTIMX(555)				현재값 유지.	현재값 유지.
타이머 대기: TIMW(813)TIMWX(816)				동작 중인 타이머의 현재값이 리프레시됩니다.	---
고속 타이머 대기: TMHW(815)/TMHWX(817)					---

*1 TIML(542), TIMLX(553), MTIM(543), MTIMX(554) 명령어는 타이머 번호를 사용하지 않으며, 다른 조건에서 리셋됩니다. (TIML(542)와 TIMLX(553)은 설정값으로 리셋되고, MTIM(543)과 MTIMX(554)는 0으로 리셋됩니다.) 자세한 내용은 각 명령어의 설명을 참조하십시오.

*2 IOM 유지 비트(A500.12)가 ON이면 치명 에러가 발생하거나 동작 모드가 PROGRAM 모드에서 RUN 또는 MONITOR 모드로, 또는 그 반대로 변경되어도 현재값과 완료 플래그가 유지됩니다. 전원을 재투입하면 현재값과 완료 플래그는 클리어됩니다. 자세한 내용은 CS/CJ/NSJ 시리즈 명령어 레퍼런스 매뉴얼(Cat. No. W474)을 참조하십시오.

*3 IOM 유지 비트(A500.12)가 ON이고 PLC 설정의 Startup 탭 페이지에 있는 기동 시 유지 영역에서 IOM 유지 비트 체크 박스가 선택되어 있으면, PLC 전원을 재투입해도 현재값과 완료 플래그가 유지됩니다. 다만 전원을 재투입하면 타이머를 연속으로 동작시킬 수 없습니다(기동 시 타이머 입력이 ON되어도 타이머는 연속으로 동작하지 않습니다).

*4 타이머 번호 T0~T2047로 프로그래밍된 100ms 타이머(TIM/TIMX(550)), 10ms 타이머(TIMH(015)/TIMHX(551)), 1ms 타이머(TMHH(540)/TMHHX(552)), 타이머 대기(TIMW(813)/TIMWX(816), 고속 타이머 대기(TMHW(815)/TMHWX(817)) 타이머의 현재값은 JMP와 JME 명령어 사이에서 점프되거나 WAIT 상태의 태스크에 있을 때도 갱신됩니다. 타이머 번호 T2048~T4095로 프로그래밍된 타이머의 현재값은 점프되거나 WAIT 상태의 태스크에 있을 때 유지됩니다.

AT 전용 -For Reference Only-

- 타이머 완료 플래그는 강제 세트 · 강제 리셋할 수 있습니다.
- 타이머 번호를 사용하는 순서나 프로그래밍할 수 있는 N.C. 또는 N.O. 조건의 수에는 제한이 없습니다.
- 0.1ms 타이머(TIMU(541)/TIMUX(556))와 0.01ms 타이머(TMUH(544)/TMUHX(557))를 제외한 모든 타이머의 현재 데이터는 워드 데이터로 읽을 수 있습니다.

6-18 카운터 영역

카운터 번호 C0~C4095로 최대 4,096개의 카운터를 사용할 수 있습니다. 카운터 데이터 영역은 두 가지입니다: 카운터 완료 플래그 영역과 카운터 현재값(PV) 영역.

・카운터 완료 플래그 (C)
완료 플래그에 액세스할 때는 카운터 번호를 사용합니다. 완료 플래그는 카운터의 설정값에 도달하면 ON됩니다.

・카운터 현재값 (C)
카운터의 현재값(16비트)을 읽고 쓸 때도 카운터 번호를 사용합니다. 현재값은 카운터 동작에 따라 증가하거나 감소합니다.

비트 데이터가 필요한 오퍼랜드에 카운터 번호를 사용하면 그 카운터 번호는 카운터의 완료 플래그에 액세스합니다.
워드 데이터가 필요한 오퍼랜드에 카운터 번호를 사용하면 그 카운터 번호는 카운터의 현재값에 액세스합니다.

다음 명령어는 모두 동일한 타이머 번호를 사용합니다 CNT, CNTX(546), CNTR(012), CNTRX(548), CNTW(814), CNTWX(818).

사용상의 주의

두 개의 카운터 명령어에 동일한 카운터 번호를 사용하는 것은 권장하지 않습니다. 동시에 계수하면 카운터가 올바르게 동작하지 않기 때문입니다. 둘 이상의 카운터 명령어가 동일한 카운터 번호를 사용하면 프로그램 체크 시 에러가 발생하지만, 그 명령어들이 같은 사이클에서 실행되지 않는 한 카운터는 동작합니다.

다음 표는 카운터 현재값과 완료 플래그가 리셋되는 시점을 나타냅니다.

명령어 이름	현재값과 완료 플래그에 대한 영향					
	리셋	모드 변경	PLC 기동	리셋 입력	CNR(545)/CNRX(547)	인터록 (IL-ILC)
카운터: CNT/CNTX(546)	현재값 → 0000 플래그 → OFF	유지	유지	리셋	리셋	유지
리버시블 카운터: CNTR(012)/CNTRX(548)						
카운터 대기: CNTW(814)/CNTWX(818)						

・카운터 완료 플래그는 강제 세트 · 강제 리셋할 수 있습니다.
・카운터 번호를 사용하는 순서나 프로그래밍할 수 있는 N.C. 또는 N.O. 조건의 수에는 제한이 없습니다.
・카운터 현재값은 워드 데이터로 읽어 프로그래밍에 사용할 수 있습니다.

AI 번역본
-For Reference Only-

6-19 태스크 플래그

태스크 플래그는 TK0~TK127이며 사이클릭 태스크 0~127에 대응합니다.

태스크 플래그는 대응하는 사이클릭 태스크가 READY 또는 RUN 상태일 때 ON되고, 그 사이클릭 태스크가 INI 또는 WAIT 상태일 때 OFF됩니다.

주 이 플래그들은 사이클릭 태스크의 상태만 나타내며, 엑스트라 사이클릭 태스크나 인터럽트 태스크의 상태는 반영하지 않습니다.

태스크 플래그는 IOM 유지 비트(A500.12)의 상태와 관계없이 다음 경우에 클리어됩니다.

- 동작 모드가 PROGRAM 모드에서 RUN/MONITOR 모드로, 또는 그 반대로 변경될 때.
- PLC 전원을 재투입할 때.

태스크 플래그는 강제 세트 · 강제 리셋할 수 없습니다.

AI 번역본
-For Reference Only-

6-20 인덱스 레지스터

16개의 인덱스 레지스터(IR0~IR15)는 간접 지정에 사용됩니다. 각 인덱스 레지스터는 PLC 메모리 어드레스 하나를 보유할 수 있으며, 이는 I/O 메모리에 있는 워드의 절대 메모리 어드레스입니다.
이것은 CIO 영역, DM 영역 등의 I/O 메모리 영역 어드레스와는 다릅니다. 연속된 RAM 어드레스입니다.

인덱스 레지스터는 각 태스크에서 독립적으로 사용하거나 전 태스크에서 공유하여 사용할 수 있습니다.

사용자가 인덱스 레지스터에 PLC 메모리 어드레스를 직접 입력할 수는 없습니다. MOVR(560)을 사용하여 일반 데이터 영역 어드레스를 그에 해당하는 PLC 메모리 어드레스로 변환하고 그 값을 지정한 인덱스 레지스터에 쓰십시오. (타이머/카운터 현재값의 PLC 메모리 어드레스를 인덱스 레지스터에 설정하려면 MOVRW(561)을 사용하십시오.)

참고

PLC 메모리 어드레스에 관한 자세한 내용은 A-4 PLC 메모리 어드레스의 메모리 맵을 참조하십시오.

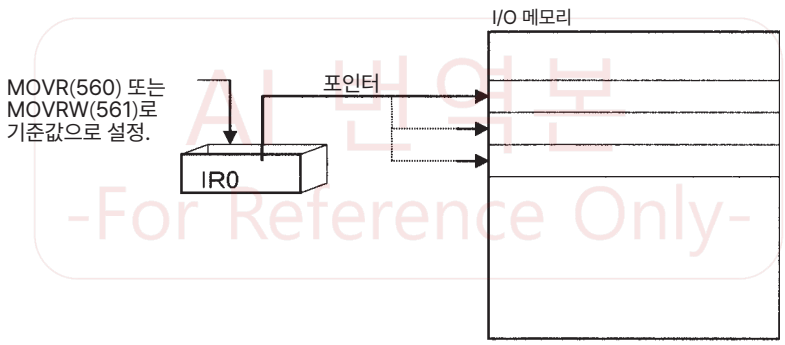
● 간접 지정

인덱스 레지스터를 "," 접두사와 함께 오퍼랜드로 사용하면 명령어는 인덱스 레지스터 자체가 아니라 인덱스 레지스터 내 PLC 메모리 어드레스가 가리키는 워드를 대상으로 동작합니다. 기본적으로 인덱스 레지스터는 I/O 메모리 포인터입니다.

- I/O 메모리의 모든 어드레스(인덱스 레지스터, 데이터 레지스터, 조건 플래그 제외)는 PLC 메모리 어드레스로 끊김 없이 지정할 수 있습니다. 데이터 영역을 지정할 필요는 없습니다.
- 기본 간접 지정 외에, 인덱스 레지스터 내의 PLC 메모리 어드레스를 상수나 데이터 레지스터로 오프셋하거나 오토 인크리먼트 · 오토 디크리먼트할 수 있습니다. 이 기능들은 명령어가 실행될 때마다 어드레스를 하나씩 증가 · 감소시키면서 데이터를 읽거나 쓰는 루프에 사용할 수 있습니다.

오프셋 및 인크리먼트/디크리먼트 변형에서는 MOVR(560) 또는 MOVRW(561)로 인덱스 레지스터를 기준값으로 설정한 후, 명령어가 실행될 때마다 포인터로서 변경할 수 있습니다.

포인터 동작에는 인덱스 레지스터를 직접 지정할 수 있는 특수 명령어(MOVR(560), MOVRW(561), 인크리먼트 명령어, 디크리먼트 명령어, 연산 명령어)를 사용하거나, 간접 오프셋, 오토 인크리먼트, 오토 디크리먼트를 사용합니다.



AI 번역본
-For Reference Only-

인덱스 레지스터를 각 태스크에서 독립적으로 사용할지 전 태스크에서 공유할지를 PLC에 설정할 수 있습니다.

인덱스 레지스터는 다음 시점에 클리어됩니다:

- PROGRAM 또는 MONITOR 모드와 RUN 모드 사이에서 동작 모드가 변경될 때
- PLC에 전원이 투입될 때
- 치명 에러가 발생할 때(FALS(007)로 발생시킨 치명 에러는 제외

사용상의 주의

인덱스 레지스터를 사용한 간접 레지스터 지정 시의 주의

I/O 메모리 외의 영역이나 시스템이 예약한 영역에 액세스하기 위해 레지스터 간접 지정을 사용하지 마십시오. 액세스를 시도하면 부정 액세스 에러가 발생합니다. PLC 메모리 어드레스의 한계에 대한 자세한 내용은 A-4 PLC 메모리 어드레스의 메모리 맵을 참조하십시오.

다음 표는 인덱스 레지스터로 I/O 메모리를 간접 지정할 때 사용할 수 있는 베리에이션을 나타냅니다.

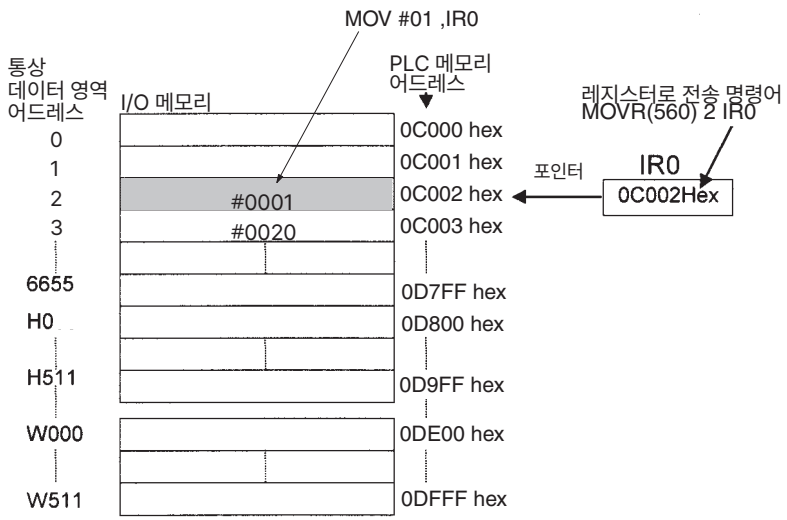
베리에이션	기능	구문	예	
간접 지정	IR□의 내용이 비트 또는 워드의 PLC 메모리 어드레스로 취급됩니다.	,IR□	LD , IR0	IR0에 들어 있는 PLC 메모리 어드레스의 비트를 로드합니다.
정수 오프셋 부가 간접 지정	정수 접두사가 IR□의 내용에 가산되고 그 결과가 비트 또는 워드의 PLC 메모리 어드레스로 취급됩니다. 정수는 −2,048~2,047 범위의 −임의의 정수를 사용할 수 있습니다.	정수 ,IR□ (정수에 + 또는 −를 포함합니다.)	LD +5, IR0	IR0의 내용에 5를 가산하고 그 PLC 메모리 어드레스의 비트를 로드합니다.
DR 오프셋 부가 간접 지정	데이터 레지스터의 내용이 IR□의 내용에 가산되고 그 결과가 비트 또는 워드의 PLC 메모리 어드레스로 취급됩니다.	DR□ , IR□	LD DR0 , IR0	DR0의 내용을 IR0의 내용에 가산하고 그 PLC 메모리 어드레스의 비트를 로드합니다.
오토 인크리먼트 부가 간접 지정	IR□의 내용을 비트 또는 워드의 PLC 메모리 어드레스로 참조한 후, 그 내용이 1 또는 2만큼 증가합니다.	1씩 증가: , IR□+ 2씩 증가: , IR□++	LD , IR0++	IR0에 들어 있는 PLC 메모리 어드레스의 비트를 로드한 후 IR0의 내용을 2만큼 증가시킵니다.
오토 디크리먼트 부가 간접 지정	IR□의 내용이 1 또는 2만큼 감소하고 그 결과가 비트 또는 워드의 PLC 메모리 어드레스로 취급됩니다.	1씩 감소: ,− IR□ 2씩 감소: ,− − IR□	LD , − −IR0	IR0의 내용을 2만큼 감소시킨 후 그 PLC 메모리 어드레스의 비트를 로드합니다.

주 IR□은 IR0~IR15의 인덱스 레지스터를 나타냅니다.

● **예**

이 예는 워드(CIO 2)의 PLC 메모리 어드레스를 인덱스 레지스터(IR0)에 저장하고, 그 인덱스 레지스터를 명령어에서 사용하며, 오토 인크리먼트 베리에이션을 사용하는 방법을 나타냅니다.

```
MOVR(560) 2 IR0          CIO 2의 PLC 메모리 어드레스를 IR0에 저장합니다.
MOV(021) #1 ,IR0         #1을 IR0에 들어 있는 PLC 메모리 어드레스, 즉 CIO 2에 씁니다.
MOV(021) #20 +1, IR0     IR0를 읽어 1을 가산하고 그 PLC 메모리 어드레스, 즉
                         CIO 3에 #20을 씁니다.
```



주 위 그림에 PLC 메모리 어드레스가 기재되어 있지만, 인덱스 레지스터를 사용할 때 PLC 메모리 어드레스를 알 필요는 없습니다.

일부 오퍼랜드는 워드 데이터로 취급되고 다른 일부는 비트 데이터로 취급되므로, 인덱스 레지스터 안의 데이터가 갖는 의미는 사용되는 오퍼랜드에 따라 달라집니다.

・워드 오퍼랜드:
MOVR(560) 0 IR2
MOV(021) D0 , IR2

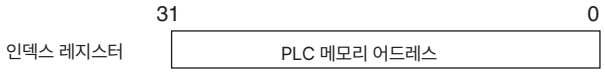
오퍼랜드가 워드로 취급되는 경우, 인덱스 레지스터의 내용이 워드의 PLC 메모리 어드레스로 "그대로" 사용됩니다. 이 예에서는 MOVR(560)이 CIO 2의 PLC 메모리 어드레스를 IR2에 설정하고, MOV(021) 명령어가 D0의 내용을 CIO 2에 복사합니다.

・비트 오퍼랜드:
MOVR(560) 0.13 ,IR2
SET +5 , IR2

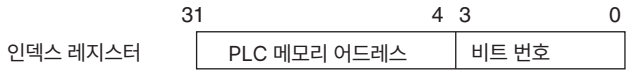
위의 SET 명령어처럼 인덱스 레지스터를 사용하여 비트를 지정할 수도 있습니다. 이 예에서는 MOVR(560)이 CIO 0.13의 PLC 메모리 비트 어드레스를 IR2에 설정합니다. SET 명령어는 이 PLC 메모리 어드레스에 비트 13부터 +5를 가산하므로 CIO 1.02 비트를 ON합니다.

참고

MOVR(560)을 사용하여 인덱스 레지스터에 워드 어드레스를 설정하면 어드레스는 다음과 같이 저장됩니다:



MOVR(560)을 사용하여 인덱스 레지스터에 비트 어드레스를 설정하면 어드레스는 다음과 같이 저장됩니다:



사용상의 주의

・인덱스 레지스터(IR)를 사용하기 전에 반드시 그 값을 설정하십시오. 값을 먼저 설정하지 않고 인덱스 레지스터를 사용하면 동작을 신뢰할 수 없습니다.

・인터럽트 태스크가 시작된 후에는 인덱스 레지스터의 값을 신뢰할 수 없습니다. 인터럽트 태스크에서 인덱스 레지스터를 사용할 때는 인덱스 레지스터를 사용하기 전에 MOVR(560) 또는 MOVRW(561)로 값을 설정하십시오. (MOVRW(561)은 타이머/카운터 현재값에, MOVR(560)은 그 밖의 값에 사용합니다.)

AI 번역본

참고

・인덱스 레지스터를 "," 접두사 없이 오퍼랜드로 사용하면, 명령어는 인덱스 레지스터 자체의 내용(2워드, 즉 "더블" 값)에 대해 동작합니다. 인덱스 레지스터를 직접 지정할 수 있는 것은 다음 표에 나오는 명령어뿐입니다. 인덱스 레지스터를 포인터로 조작하려면 이 명령어를 사용하십시오.

・인덱스 레지스터는 그 밖의 명령어에서는 직접 지정할 수 없지만, 대개 간접 지정에는 사용할 수 있습니다.

● 직접 지정

명령어 그룹	명령어 이름	니모닉
데이터 전송 명령어	레지스터로 전송	MOVR(560)
	타이머/카운터 현재값을 레지스터로 전송	MOVRW(561)
	더블 전송	MOVL(498)
	더블 데이터 교환	XCGL(562)
테이블 데이터 처리 명령어	레코드 위치 설정	SETR(635)
	레코드 번호 취득	GETR(636)
트래킹 명령어	부호 없는 1워드 레코드 검색 명령어	RSRCH (360~364)
	부호 없는 1워드 레코드 정렬	RSORT(203)
인크리먼트/디크리먼트 명령어	BIN 더블 인크리먼트	++L(591)
	BIN 더블 디크리먼트	– –L(593)
비교 명령어	더블 등가	=L(301)
	더블 부등가	< >L(306)
	더블 미만	< L(311)
	더블 이하	< =L(316)
	더블 초과	> L(321)
	더블 이상	> =L(326)
	더블 비교	CMPL(060)
심볼 연산 명령어	캐리 없는 부호 있는 BIN 가산(더블)	+L(401)
	캐리 없는 부호 있는 BIN 감산 (더블)	–L(411)

SRCH(181), MAX(182), MIN(183) 명령어는 원하는 값(서치 값, 최댓값, 최솟값)을 가진 워드의 PLC 메모리 어드레스를 IR0에 출력할 수 있습니다. 이 경우 이후의 명령어에서 IR0를 사용하여 그 워드의 내용에 액세스할 수 있습니다.

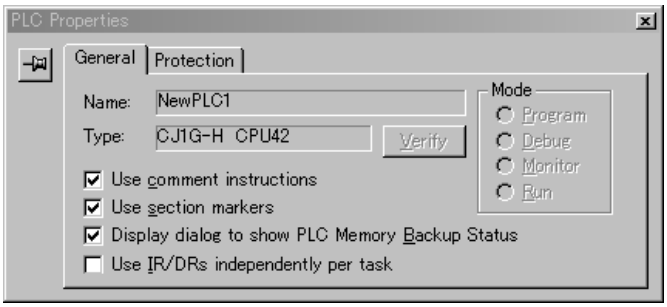
● 주의사항

- PLC 메모리 어드레스가 레지스터에 설정되기 전에는 인덱스 레지스터를 사용하지 마십시오. 값을 설정하지 않고 레지스터를 사용하면 포인터 동작을 신뢰할 수 없습니다.
- 인터럽트 태스크 시작 시점의 인덱스 레지스터 값은 예측할 수 없습니다. 인터럽트 태스크에서 인덱스 레지스터를 사용할 때는 그 태스크에서 레지스터를 사용하기 전에 반드시 MOVR(560) 또는 MOVRW(561)로 인덱스 레지스터에 PLC 메모리 어드레스를 설정하십시오.
- 인덱스 레지스터는 각 태스크에서 독립적으로 처리되므로 서로 영향을 주지 않습니다. 예를 들어 태스크 1에서 사용하는 IR0와 태스크 2에서 사용하는 IR0는 다릅니다. 따라서 각 인덱스 레지스터 태스크는 16개의 인덱스 레지스터를 가집니다.
- CX-Programmer에서는 사이클 내에서 마지막으로 실행된 태스크의 인덱스 레지스터만 읽을 수 있습니다. 같은 번호의 인덱스 레지스터를 여러 태스크에서 사용하는 경우, 여러 태스크 중 사이클 내에서 마지막으로 실행된 태스크의 인덱스 레지스터 값만 CX-Programmer에서 읽을 수 있습니다. 또한 CX-Programmer에서 인덱스 레지스터 값을 쓸 수도 없습니다.

● 인덱스 레지스터 공유

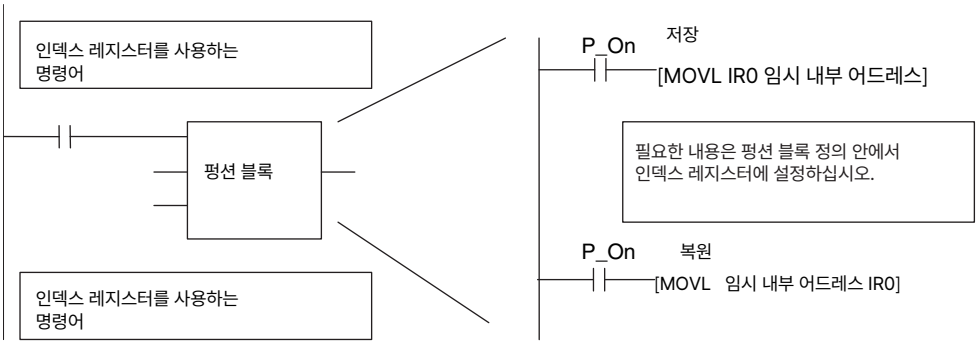
CX-Programmer의 PLC Properties 대화 상자에서 다음 설정을 하여 태스크 간의 인덱스 레지스터와 데이터 레지스터 공유를 제어할 수 있습니다.

모든 태스크 간에 인덱스 레지스터를 공유하려면 PLC Properties 대화 상자에서 Use IR/DRs independently per task 체크 박스의 선택을 해제하십시오.



참고

펑션 블록 안에서 사용하는 인덱스 레지스터의 내용은 펑션 블록이 호출될 때 손상될 수 있습니다. 펑션 블록을 호출하기 전에 반드시 인덱스 레지스터의 내용을 저장하고, 펑션 블록에서 빠져나온 후에 그 내용을 복원하십시오. 필요한 내용은 펑션 블록 안에서 인덱스 레지스터에 설정하십시오.



AI 번역본
-For Reference Only-

6-21 데이터 레지스터

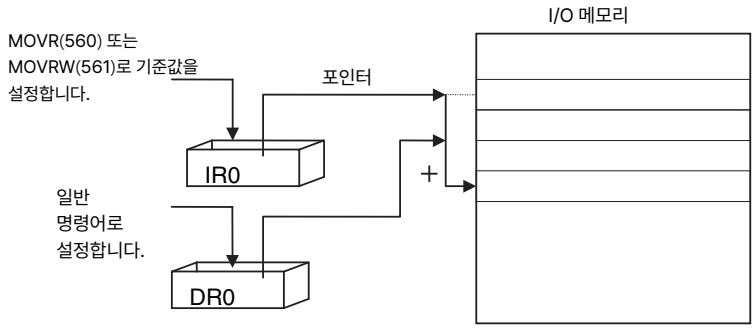
16개의 데이터 레지스터(DR0~DR15)는 워드를 간접 지정할 때 인덱스 레지스터의 PLC 메모리 어드레스를 오프셋하는 데 사용합니다. 데이터 레지스터는 워드를 간접 지정할 때 인덱스 레지스터에 가산할 오프셋을 지정하는 데 사용할 수 있습니다.

데이터 레지스터는 태스크마다 독립적으로 사용하거나 태스크 간에 공유할 수 있습니다.

데이터 레지스터의 값을 인덱스 레지스터의 PLC 메모리 어드레스에 가산하여 I/O 메모리 내 비트나 워드의 절대 메모리 어드레스를 지정할 수 있습니다. 데이터 레지스터는 부호 있는 2진 데이터를 저장하므로 인덱스 레지스터의 내용을 낮은 어드레스나 높은 어드레스로 오프셋할 수 있습니다.

일반 명령어를 사용하여 데이터 레지스터에 데이터를 저장할 수 있습니다.

데이터 레지스터는 강제 세트 또는 강제 리셋할 수 없습니다.



● 예

다음 예는 데이터 레지스터를 사용하여 인덱스 레지스터의 PLC 메모리 어드레스를 오프셋하는 방법을 나타냅니다.

LD DR0 ,IR0
DR0의 내용을 IR0의 내용에 가산하고 그 PLC 메모리 어드레스의 비트를 로드합니다.

MOV(021) #1 DR0 ,IR1
DR0의 내용을 IR1의 내용에 가산하고 그 PLC 메모리 어드레스에 #1을 씁니다.

● 값의 범위

데이터 레지스터의 내용은 부호 있는 바이너리 데이터로 취급되므로 그 범위는 –32,768~ 32,767입니다. (음수값은 2의 보수로 주어집니다.)

16진수 내용	10진수 환산값
8000~FFFF	–32,768~–1
0000~7FFF	0~32,767

사본 열람본
For Reference Only

● 데이터 레지스터의 초기화

데이터 레지스터는 다음의 경우에 클리어됩니다:

- 동작 모드가 PROGRAM 모드에서 RUN/MONITOR 모드로, 또는 그 반대로 변경되고 IOM 유지 비트가 OFF인 경우.
- PLC의 전원이 재투입되고 IOM 유지 비트가 OFF이거나 PLC 설정에서 유지하도록 설정되어 있지 않은 경우.
- 치명 에러가 발생한 경우(FALS(007)로 발생시킨 것은 제외).

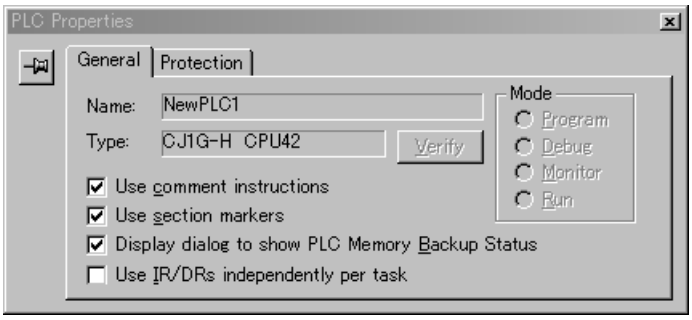
● IOM 유지 비트의 동작

IOM 유지 비트(A500.12)가 ON이면 FALS 에러가 발생하거나 동작 모드가 PROGRAM 모드에서 RUN/MONITOR 모드로, 또는 그 반대로 변경되어도 데이터 레지스터는 클리어되지 않습니다.

IOM 유지 비트(A500.12)가 ON이고 PLC 설정의 기동 탭 페이지에 있는 기동 시 유지 영역에서 IOM 유지 비트 체크 박스가 선택되어 있으면, PLC의 전원이 재투입되어도 데이터 레지스터는 클리어되지 않습니다.

● 데이터 레지스터 공유

CX-Programmer의 PLC Properties 대화 상자에서 다음 설정을 하여 태스크 간의 인덱스 레지스터와 데이터 레지스터 공유를 제어할 수 있습니다.



참고

프로그램에서 인덱스 레지스터와 데이터 레지스터를 사용하지 않는 경우에는 태스크 간에 인덱스 레지스터와 데이터 레지스터를 공유하도록 PLC 프로퍼티를 설정할 것을 권장합니다.

AI 번역본
-For Reference Only-

6-22 조건 플래그

조건 플래그에는 명령어 실행 결과를 나타내는 에러 플래그, 캐리 플래그 등의 플래그와 상시 ON 플래그 및 상시 OFF 플래그가 있습니다. 이전 PLC에서는 이 플래그들이 보조 기억 영역에 있었습니다.

조건 플래그는 어드레스가 아니라 P_CY, P_ER 같은 글로벌 심볼로 지정합니다. 이 플래그는 명령어나 CX-Programmer에서 직접 쓸 수 없습니다.

모든 조건 플래그는 프로그램이 태스크를 전환할 때 클리어되므로, 에러 플래그, 액세스 에러 플래그 등의 상태는 에러가 발생한 태스크 안에서만 유지됩니다.

조건 플래그는 강제 세트 및 강제 리셋할 수 없습니다.

● 조건 플래그 목록

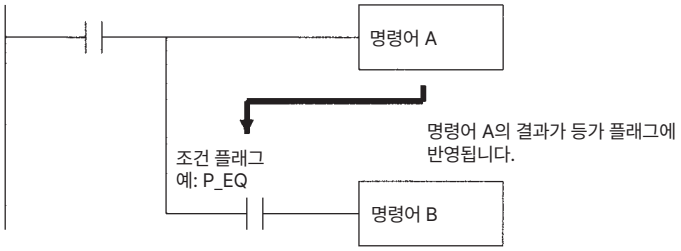
다음 표는 조건 플래그의 기능을 정리한 것입니다.

명칭	심볼	라벨	기능
에러 플래그	P_ER	ER	명령어의 오퍼랜드 데이터가 올바르지 않을 때(명령어 처리 에러) ON되어 명령어가 에러로 종료되었음을 나타냅니다. PLC 설정에서 명령어 에러 시 동작 정지(Instruction Error Operation)로 설정되어 있으면, 에러 플래그가 ON될 때 프로그램 실행이 정지되고 명령어 처리 에러 플래그(A295.08)가 ON됩니다.
액세스 에러 플래그	P_AER	AER	부정 액세스 에러가 발생했을 때 ON됩니다. 부정 액세스 에러는 액세스해서는 안 되는 메모리 영역에 명령어가 액세스하려 했음을 나타냅니다. PLC 설정에서 명령어 에러 시 동작 정지(Instruction Error Operation)로 설정되어 있으면, 액세스 에러 플래그가 ON될 때 프로그램 실행이 정지되고 명령어 처리 에러 플래그(A295.10)가 ON됩니다.
캐리 플래그	P_CY	CY	연산 결과에 캐리가 발생하거나 데이터 시프트 명령어에 의해 "1"이 캐리 플래그로 시프트되었을 때 ON됩니다. 캐리 플래그는 일부 데이터 시프트 명령어와 심볼 연산 명령어의 결과의 일부입니다.
초과 플래그	P_GT	>	비교 명령어의 첫 번째 오퍼랜드가 두 번째 오퍼랜드보다 크거나 값이 지정 범위를 초과했을 때 ON됩니다.
이퀄 플래그	P_EQ	=	비교 명령어의 두 오퍼랜드가 같을 때 연산 결과가 0일 때 ON됩니다.
미만 플래그	P_LT	<	비교 명령어의 첫 번째 오퍼랜드가 두 번째 오퍼랜드보다 작거나 값이 지정 범위를 밑돌 때 ON됩니다.
네거티브 플래그	P_N	N	결과의 최상위 비트(부호 비트)가 ON일 때 ON됩니다.
오버플로 플래그	P_OF	OF	연산 결과가 결과 워드의 용량을 오버플로했을 때 ON됩니다.
언더플로 플래그	P_UF	UF	연산 결과가 결과 워드의 용량을 언더플로했을 때 ON됩니다.
이상 플래그	P_GE	>=	비교 명령어의 첫 번째 오퍼랜드가 두 번째 오퍼랜드보다 크거나 같을 때 ON됩니다.
부등가 플래그	P_NE	< >	비교 명령어의 두 오퍼랜드가 같지 않을 때 ON됩니다.
이하 플래그	P_LE	< =	비교 명령어의 첫 번째 오퍼랜드가 두 번째 오퍼랜드보다 작거나 같을 때 ON됩니다.
상시 ON 플래그	P_On	ON	상시 ON입니다.
상시 OFF 플래그	P_Off	OFF	상시 OFF입니다.

AI 번역본
For Reference Only

● 조건 플래그 사용

조건 플래그는 모든 명령어가 공유하므로 1사이클 안에서도 상태가 자주 바뀔 수 있습니다. 조건 플래그는 반드시 명령어 실행 직후에, 가급적 같은 실행 조건에서 분기한 자리에서 읽으십시오.



명령어	오퍼랜드
LD	
명령어 A	
AND	P_EQ
명령어 B	

사용상의 주의

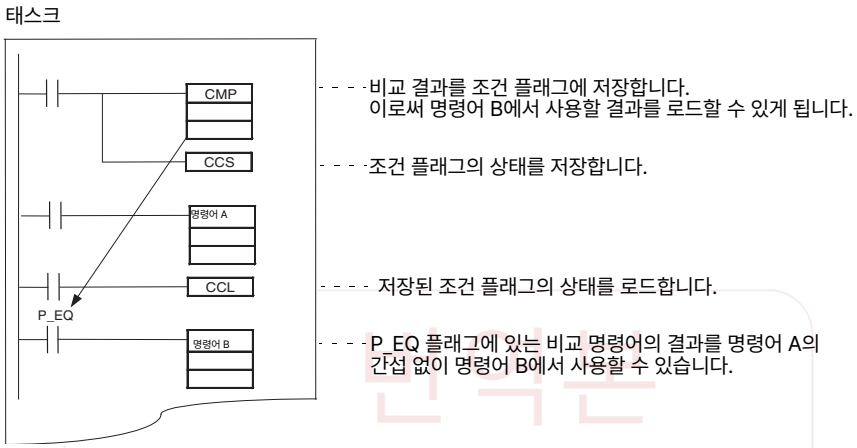
- 조건 플래그는 모든 명령어가 공유하므로, 하나의 태스크가 인터럽트되는 것만으로도 프로그램 동작이 예상한 경로에서 벗어날 수 있습니다. 프로그램을 작성할 때는 반드시 인터럽트의 영향을 고려하십시오.

- 조건 플래그는 프로그램이 태스크를 전환할 때 클리어되므로 조건 플래그의 상태를 다른 태스크로 전달할 수 없습니다. 예를 들어 태스크 1의 플래그 상태를 태스크 2에서 읽을 수 없습니다.

● 조건 플래그 상태의 저장과 로드

조건 플래그 상태 명령어(CCS(282)와 CCL(283))를 사용하여 하나의 태스크(프로그램) 안의 서로 다른 위치 사이, 또는 서로 다른 태스크나 사이클 사이에서 조건 플래그의 상태를 저장하고 로드할 수 있습니다.

다음 예는 같은 태스크 안의 다른 위치에서 이퀄 플래그를 사용하는 방법을 나타냅니다.



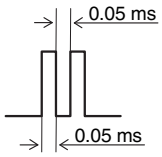
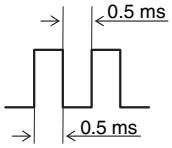
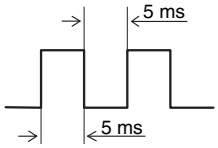
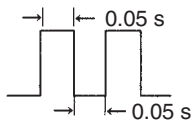
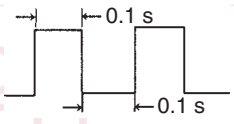
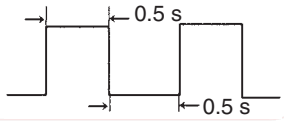
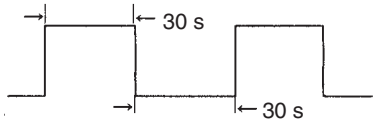
-For Reference Only-

6-23 클록 펄스

클록 펄스는 시스템에 의해 일정 간격으로 ON/OFF되는 플래그입니다.

클록 펄스는 읽기 전용이며 명령어나 CX-Programmer에서 덮어쓸 수 없습니다.
운전 개시 시에 클리어됩니다.

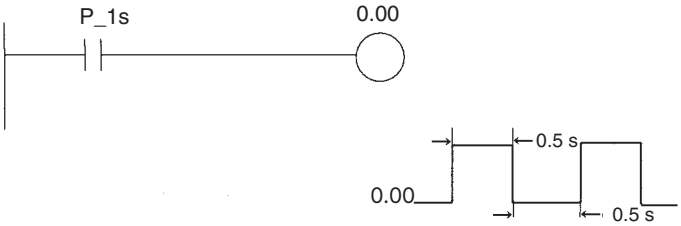
클록 펄스는 P_0_1ms, P_1ms 같은 글로벌 심볼을 사용하여 지정합니다.

명칭	심볼	프로그래밍 콘솔에서의 명칭	동작	
0.1 ms 클록 펄스	P_0_1ms	0.1 ms		0.05 ms 동안 ON 0.05 ms 동안 OFF
1 ms 클록 펄스	P_1ms	1 ms		0.5 ms 동안 ON 0.5 ms 동안 OFF
0.01 s 클록 펄스	P_0_01s	0.01 s		5 ms 동안 ON 5 ms 동안 OFF
0.02 s 클록 펄스	P_0_02_s	0.02 s		0.01 s 동안 ON 0.01 s 동안 OFF
0.1 s 클록 펄스	P_0_1s	0.1 s		0.05 s 동안 ON 0.05 s 동안 OFF
0.2 s 클록 펄스	P_0_2s	0.2 s		0.1 s 동안 ON 0.1 s 동안 OFF
1 s 클록 펄스	P_1s	1 s		0.5 s 동안 ON 0.5 s 동안 OFF
1 min 클록 펄스	P_1min	1 min		30 s 동안 ON 30 s 동안 OFF

AI 번역
-For Referen

● 클록 펄스 사용

다음 예는 CIO 0.00을 0.5 s 간격으로 ON/OFF합니다.



명령어	오퍼랜드
LD	P_1s
OUT	0.00

● 클록 펄스의 리프레시

클록 펄스는 프로그램 실행 중에도 리프레시됩니다. 프로그램의 시작과 끝에서 ON/OFF 상태가 같지 않을 수 있습니다.

● 클록 펄스의 에러

클록 펄스의 최대 오차는 0.01%(25°C에서)입니다. 장시간에 걸친 시간 기반 제어에는 클록 펄스 대신 내부 클록을 사용할 것을 권장합니다. 반드시 내부 클록의 오차를 고려하십시오.

AI 번역본
-For Reference Only-

AI 번역본
-For Reference Only-

AI 번역본
-For Reference Only-

7

파일 조작

이 섹션에서는 CJ2/CJ3 CPU 유닛에서 수행할 수 있는 파일 조작을 설명합니다.

7-1 파일 메모리 7-2
- 7-1-1 파일 메모리의 종류 7-2
- 7-1-2 파일 메모리 초기화 7-3
- 7-1-3 메모리 카드 주의사항 7-5

7-2 파일 메모리에 저장되는 파일의 종류 7-8
- 7-2-1 파일 종류 7-8
- 7-2-2 파일 메모리용 파일의 작성과 저장 7-11

7-3 파일 메모리 조작 7-12
- 7-3-1 파일 메모리 조작의 종류 7-12
- 7-3-3 파일 메모리 조작 순서와 파일 메모리 파일 7-14
- 7-3-4 파일 사용상의 제한 7-20
- 7-3-5 파일 사이즈 7-21
- 7-3-6 지원 소프트웨어와 파일 메모리 파일의 관계 7-22

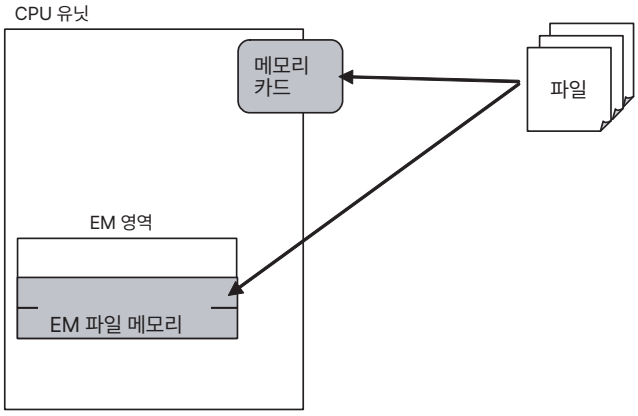
AI 번역본
-For Reference Only-

7-1 파일 메모리

7-1-1 파일 메모리의 종류

CJ 시리즈 PLC에서는 파일 메모리를 사용하여 파일을 저장할 수 있습니다. 다음 두 종류의 파일 메모리를 사용합니다.

- 메모리 카드
- EM 파일 메모리라고 하는 EM 영역의 지정 범위



CJ 시리즈 PLC에서는 메모리 카드와 EM 영역의 지정 범위를 파일 저장용 메모리로 사용할 수 있습니다. 어느 쪽이든 사용자 프로그램 전체, I/O 메모리, 파라미터 영역을 파일로 저장할 수 있습니다.

분류	메모리 종류	메모리 용량	저장 가능한 파일 종류
메모리 카드	플래시 메모리	CJ3 CPU 유닛용 HMC-SD29□: 2 Gbytes HMC-SD49□: 4 Gbytes HMC-SD1A□: 16 Gbytes	· 프로그램 파일 · 코멘트 파일 · 프로그램 인덱스 파일 · 심볼 테이블 파일 · 파라미터 파일 · 유닛 및 보드 백업 파일 (메모리 카드 전용)
		CJ2 CPU 유닛용 HMC-EF183: 128 Mbytes HMC-EF283: 256 Mbytes HMC-EF583: 512 Mbytes	
내장 RAM (EM 파일 메모리)	RAM	EM 영역의 EM 영역 용량에서 EM 영역의 마지막 뱅크까지	

AI 번역본
-For Reference Only-

7-1-2 파일 메모리 초기화

메모리 카드

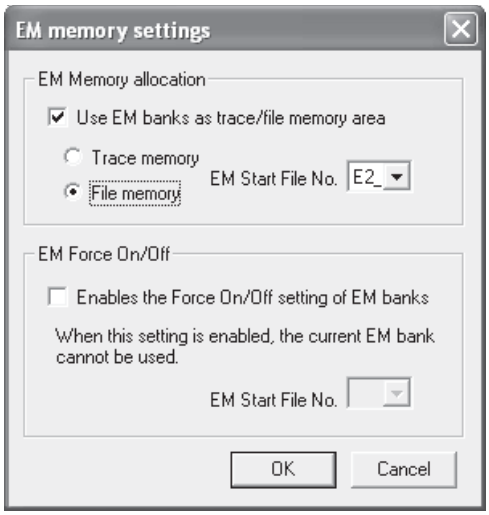
메모리 카드를 초기화하려면 CX-Programmer의 Memory Card 대화 상자를 사용하십시오. 다만 HMC-EF□□□ 및 HMC-SD□□□ 메모리 카드는 디폴트로 이미 초기화되어 있으므로 처음 사용할 때 초기화할 필요가 없습니다.

EM 파일 메모리

EM 파일 메모리로 사용할 EM 파일 메모리의 선두 뱅크를 지정하고 EM 파일 메모리를 초기화하려면 다음 순서를 사용하십시오.

1 **CX-Programmer에서 PLC – Memory Allocate – EM Memory Settings를 선택합니다.**

2 EM Memory Settings 대화 상자에서 File Memory를 선택한 다음 EM start File No.를 설정합니다.



참고

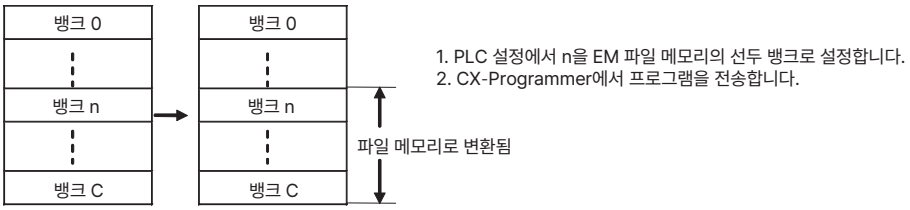
- EM 영역은 파일 메모리와 트레이스 메모리로 동시에 사용할 수 없습니다. 이 기능 중 하나만 사용하십시오.
- 트레이스/파일 메모리로 사용되는 EM 뱅크에서는 강제 세트/리셋을 할 수 없습니다.
- EM 파일 메모리의 루트 디렉터리에는 최대 64개의 파일을 저장할 수 있습니다.
- EM 파일 메모리의 할당 단위 사이즈는 512바이트입니다.

AI 번역본
-For Reference Only-

3 CPU 유닛에 온라인으로 접속하여 사용자 프로그램을 전송합니다. 그러면 EM 파일 메모리를 사용할 수 있게 됩니다.

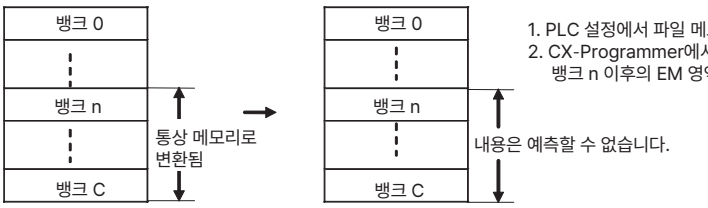
● EM 파일 메모리 설정 변경

다음 그림은 EM 영역을 지정한 뱅크부터 마지막 뱅크까지 파일 메모리로 변환하는 것을 나타냅니다.



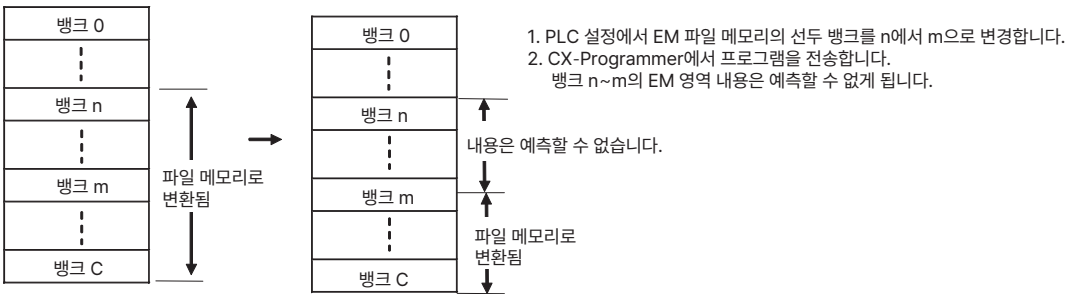
1. PLC 설정에서 n을 EM 파일 메모리의 선두 뱅크로 설정합니다.
2. CX-Programmer에서 프로그램을 전송합니다.

다음 그림은 파일 메모리로 사용되던 EM 영역 뱅크를 통상의 EM 영역 메모리로 되돌리는 것을 나타냅니다.



1. PLC 설정에서 파일 메모리 변환을 무효화합니다.
2. CX-Programmer에서 프로그램을 전송합니다.
   뱅크 n 이후의 EM 영역 내용은 예측할 수 없게 됩니다.

파일 메모리의 선두 뱅크 번호는 변경할 수 있습니다.



1. PLC 설정에서 EM 파일 메모리의 선두 뱅크를 n에서 m으로 변경합니다.
2. CX-Programmer에서 프로그램을 전송합니다.
   뱅크 n~m의 EM 영역 내용은 예측할 수 없게 됩니다.

AI 번역본
-For Reference Only-

7-1-3 메모리 카드 주의사항

메모리 카드를 사용하기 전에 다음 항목을 확인하십시오.

메모리 카드 사양

· CJ2 CPU 유닛용

분류	메모리 종류	메모리 용량	데이터 포맷	재기록 횟수 (1블록당)	쓰기 방지
HMC-EF183	메모리 카드	128 Mbytes	FAT16	100,000	없음
HMC-EF283		256 Mbytes			
HMC-EF583		512 Mbytes			

· CJ3 CPU 유닛용

분류	메모리 종류	메모리 용량	데이터 포맷	재기록 횟수 (1블록당)	쓰기 방지
HMC-SD292/293	SD 카드	2 Gbytes	FAT16	50,000	SD 메모리 카드의 하드웨어 스위치로 설정 가능
HMC-SD492/493	SDHC 카드	4 Gbytes	FAT32	100,000	
HMC-SD1A2/1A3		16 Gbytes			

포맷

· HMC-EF□83용(CJ2 CPU 유닛용 메모리 카드)
메모리 카드는 출하 전에 포맷되어 있습니다. 구입 후에 포맷할 필요는 없습니다. 사용한 뒤에 포맷할 때는 반드시 CX-Programmer를 사용하여 CPU 유닛에서 포맷하십시오. 메모리 카드를 노트북 컴퓨터나 그 밖의 컴퓨터에서 직접 포맷하면 CPU 유닛이 메모리 카드를 인식하지 못할 수 있습니다. 이 경우 CPU 유닛에서 다시 포맷하더라도 그 메모리 카드를 사용할 수 없게 됩니다.

· HMC-SD□□□용(CJ3 CPU 유닛용 메모리 카드)
메모리 카드는 출하 전에 포맷되어 있습니다. 구입 후에 포맷할 필요는 없습니다. 사용한 뒤에 포맷할 때는 반드시 CX-Programmer 또는 컴퓨터를 사용하여 CPU 유닛에서 포맷하십시오.

루트 디렉터리의 파일 수

메모리 카드의 루트 디렉터리 또는 서브디렉터리에 둘 수 있는 파일 수에는 제한이 있습니다. 둘 수 있는 최대 파일 수는 다음과 같습니다.

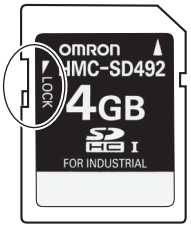
분류	데이터 포맷	디렉터리 계층	둘 수 있는 최대 파일 수
HMC-EF183 HMC-EF283 HMC-EF583 HMC-SD292/293	FAT16	루트 디렉터리	511
		서브디렉터리	65533
HMC-SD492/493 HMC-SD1A2/1A3	FAT32	루트 디렉터리	65533
		서브디렉터리	65533

서브디렉터리는 컴퓨터에서 작성하거나 CMND(490) 명령어를 사용하여 작성할 수 있습니다.

사 본 영 본

쓰기 방지

SD 메모리 카드(HMC-SD□□□)는 쓰기 방지를 할 수 있습니다. 탭을 LOCK 위치로 설정하면 카드에 대한 쓰기가 무효화됩니다. 데이터가 덮어써지는 것을 방지하려면 쓰기 방지를 사용하십시오. (HMC-EF□□□에는 쓰기 방지 기능이 없습니다.)



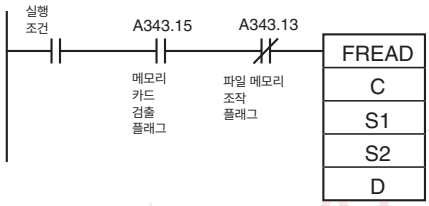
메모리 카드 인식 시간

CPU 유닛이 메모리 카드를 삽입한 후 인식하기까지는 보통 수 초가 걸립니다.
필요한 시간은 PLC의 사이클 타임, 메모리 카드 용량, 메모리 카드에 저장된 파일 수 등에 따라 달라집니다.

메모리 카드 액세스

- SD BUSY/BUSY 표시등이 점등 중일 때(즉, 메모리 카드에 액세스 중임을 나타내는 표시등이 점등 중일 때)는 절대로 CPU 유닛의 전원을 OFF하지 마십시오. 그렇게 하면 메모리 카드를 사용할 수 없게 될 수 있습니다.

- SD BUSY/BUSY 표시등이 점등 중일 때는 절대로 CPU 유닛에서 메모리 카드를 빼지 마십시오. 메모리 카드를 빼기 전에 메모리 카드 전원 OFF 버튼을 누르고 SD BUSY/BUSY 표시등이 소등될 때까지 기다리십시오. 그렇게 하지 않으면 메모리 카드를 사용할 수 없게 될 수 있습니다.
- CPU 유닛이 메모리 카드를 삽입한 후 인식하기까지는 수 초가 걸립니다.
  전원을 ON한 직후나 메모리 카드를 삽입한 직후에 메모리 카드에 액세스할 때는, 아래와 같이 메모리 카드 인식 플래그(A343.15)의 NC 조건을 입력 조건으로 하여 프로그래밍하십시오.



AI 번역본
For Reference Only

파일 메모리 액세스 중의 정전

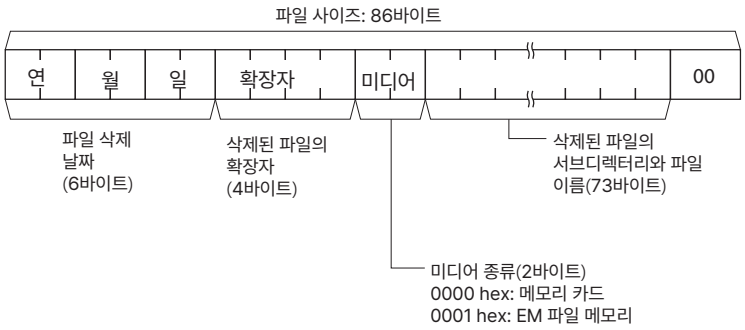
CPU가 파일 메모리(메모리 카드 또는 EM 파일 메모리)에 액세스하는 중에 정전이 발생하면 메모리 카드의 내용이 정확하지 않을 수 있습니다. 갱신 중이던 파일이 올바르게 덮어써지지 않을 수 있으며, 경우에 따라서는 메모리 카드 자체가 손상될 수 있습니다.

- CJ2 CPU 유닛

해당 파일은 다음에 전원을 ON할 때 시스템에 의해 자동으로 삭제됩니다. 그에 대응하는 파일 삭제 통지 플래그(메모리 카드는 A395.06, EM 파일 메모리는 A395.07)가 ON됩니다. 이 플래그는 다음에 전원을 OFF할 때 OFF됩니다.

파일이 삭제되면 메모리 카드 또는 EM 파일 메모리의 루트 디렉터리에 삭제 로그 파일 (DEL_FILE.TMP)이 작성됩니다. 삭제 로그 파일(DEL_FILE.TMP)은 바이너리 에디터로 읽어 다음 정보를 확인할 수 있습니다: 파일이 삭제된 날짜, 존재했던 파일 메모리의 종류(미디어), 서브디렉터리, 파일 이름, 확장자. 필요한 경우 삭제된 파일을 다시 작성하거나 다시 복사하십시오.

다음 그림은 삭제 로그 파일(DEL_FILE.TMP)의 구조를 나타냅니다.



・CJ3 CPU 유닛

해당 파일은 유지됩니다. 파일은 삭제되지 않으며 삭제 로그 파일도 작성되지 않습니다. 보조 기억 영역의 파일 쓰기 중 정전 발생 플래그(A937.00)는 다음에 전원을 ON할 때 자동으로 1(ON)로 바뀝니다. 파일 쓰기 중 정전 발생 플래그가 ON되면 파일이 손상되지 않았는지 확인하십시오. 파일 쓰기 중 정전 발생 플래그는 자동으로 0(OFF)이 되지 않습니다. CPU 유닛이 배터리 없이 동작할 때는 다음 데이터를 유지할 수 있습니다. 파일이 손상되지 않았음을 확인한 후 수동으로 플래그를 OFF하십시오.

파일 메모리 용량이 부족한 상태에서 파일 메모리에 저장할 때의 동작

파일 메모리(메모리 카드 또는 EM 파일 메모리)에 데이터를 저장할 때 파일 메모리에 사용 가능한 용량이 부족하면 다음과 같이 동작합니다.

● CX-Programmer에서의 동작

(1) 파일 메모리의 사용 가능 용량은 CPU 유닛 데이터(사용자 프로그램, I/O 메모리, 파라미터 영역 데이터)를 파일 메모리로 전송하기 전에 시스템이 확인합니다. 파일 메모리에 사용 가능한 용량이 부족하면 파일이 작성되지 않습니다.

(2) 컴퓨터의 파일을 파일 메모리로 전송하는 경우에는 시스템이 사용 가능 용량을 사전에 확인하지 않습니다. 파일 메모리에 사용 가능한 용량이 부족하면 사용 가능한 파일 메모리 용량만큼의 파일이 작성됩니다. (예: 1메가바이트 파일을 전송했는데 파일 메모리에 사용 가능한 용량이 500킬로바이트뿐이면 500킬로바이트 파일이 작성됩니다.) 또한 파일 메모리에 이미 존재하는 파일 이름으로 파일을 덮어쓸 때, 파일 메모리의 용량이 부족하면 그 파일은 삭제됩니다.

● FWRIT(701) 명령어 사용

시스템은 데이터를 쓰기 전에 파일 메모리 용량을 확인합니다. 사용 가능한 파일 메모리 용량이 부족하면 파일이 작성되지 않고 파일 쓰기 에러 플래그(A343.09)가 ON됩니다.

● 백업 동작

간이 백업 동작 중에 파일 메모리의 사용 가능 용량이 부족하면 백업 동작은 에러로 종료됩니다. (SD PWR/MCPWR 표시등이 계속 점멸합니다.)

시 범 영 본
-For Reference Only-

7-2 파일 메모리에 저장되는 파일의 종류

파일 메모리는 다음 용도로 사용할 수 있습니다:

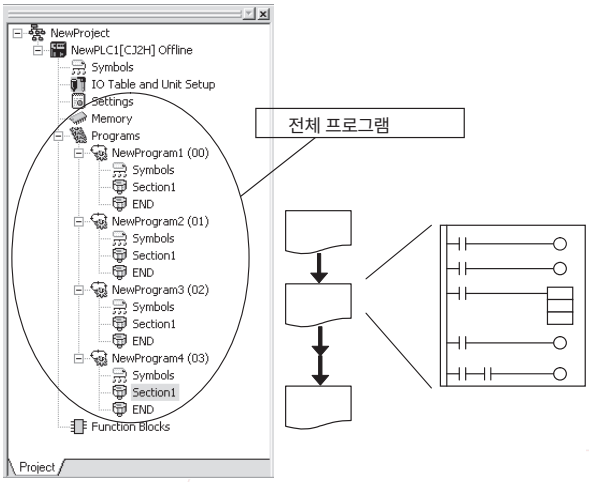
- 프로그램/네트워크 심볼 파일
- 파라미터 파일
- 데이터 파일
- 심볼 테이블 파일(네트워크 심볼 제외)
- 코멘트 파일
- 프로그램 인덱스 파일
- 유닛 백업 파일

7-2-1 파일 종류

프로그램/네트워크 심볼 파일

프로그램/네트워크 심볼 파일에는 CPU 유닛의 사용자 프로그램(사이클릭 태스크와 인터럽트 태스크의 프로그램)과 네트워크 심볼(즉, 글로벌 심볼 테이블 안의 네트워크 심볼)*이 들어 있습니다.

이 파일에는 각 프로그램의 프로퍼티도 들어 있습니다.

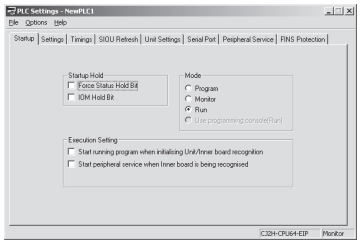


AI 번역본
-For Reference Only-

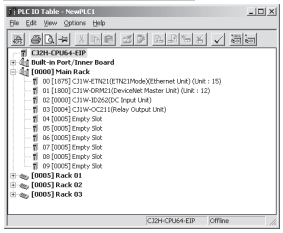
파라미터 파일

파라미터 파일에는 CPU 유닛 내부의 파라미터 영역 데이터가 들어 있습니다. 파라미터 영역 데이터에는 PLC 이름, PLC 설정, I/O 테이블, CPU 버스 유닛 설정(데이터 링크 파라미터 포함), 라우팅 테이블이 포함됩니다.

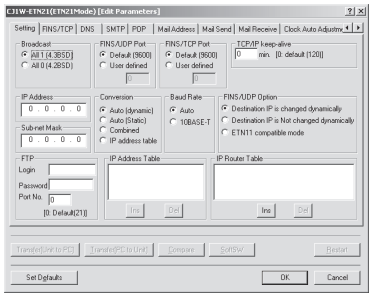
PLC 설정



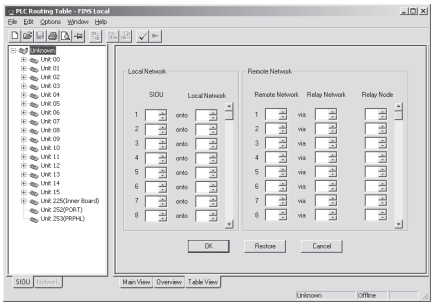
I/O 테이블



CPU 버스 유닛 설정

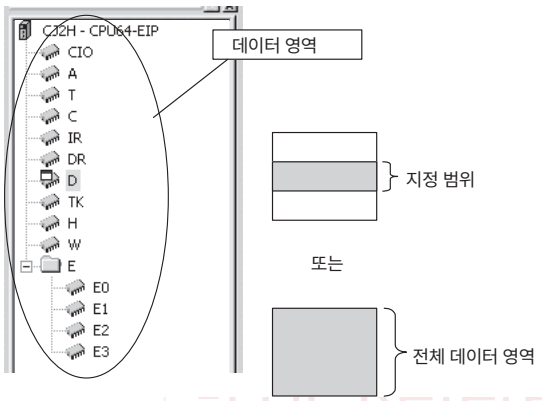


라우팅 테이블



데이터 파일

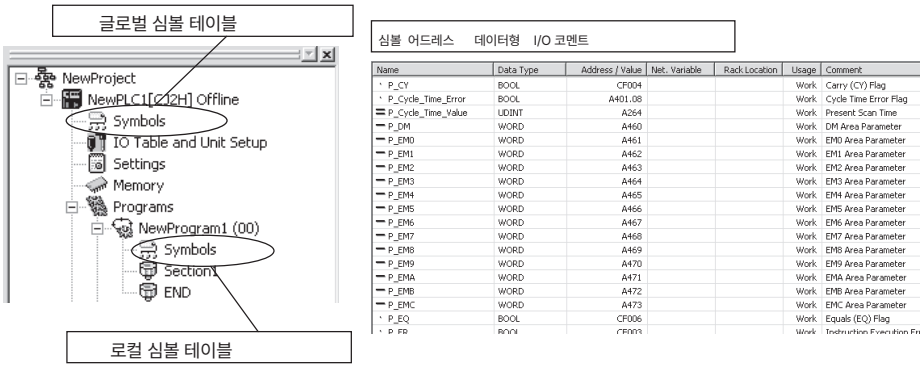
데이터 파일에는 하나의 I/O 메모리 데이터 영역의 데이터가 워드(16비트) 단위로 들어 있습니다. 데이터 영역의 전체 데이터를 저장할 수도 있고 지정한 어드레스 범위만 저장할 수도 있습니다. 다음 6개 데이터 영역 중 어느 하나를 저장할 수 있습니다: CIO 영역, 유지 영역, 워크 영역, 보조 기억 영역, DM 영역, EM 영역.



역본
-For Reference Only-

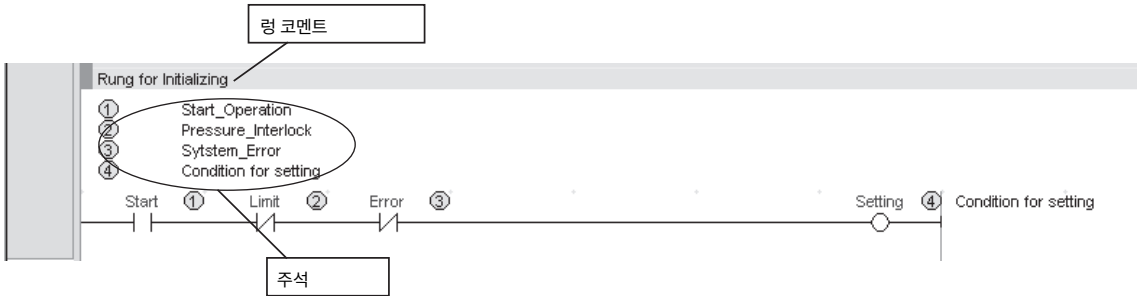
심볼 테이블 파일(네트워크 심볼 제외)

심볼 테이블 파일(네트워크 심볼 제외)에는 CX-Programmer에서 사용하는 심볼 테이블 정보와 자동으로 할당된 PLC 어드레스가 들어 있습니다.



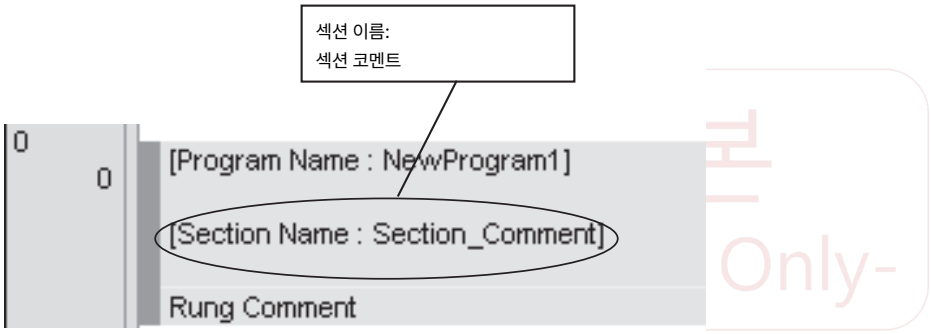
코멘트 파일

코멘트 파일에는 CX-Programmer에서 사용하는 코멘트 정보가 들어 있습니다.



프로그램 인덱스 파일

프로그램 인덱스 파일에는 CX-Programmer에서 사용하는 섹션 정보가 들어 있습니다.



유닛 백업 파일

유닛 백업 파일에는 간이 백업 기능에서 사용하는 PLC 유닛의 내부 데이터가 들어 있습니다. 이 파일은 간이 백업 조작이 실행될 때 작성됩니다. 내부 데이터는 유닛마다 저장됩니다.

예:

DeviceNet 유닛: 디바이스 파라미터

시리얼 통신 유닛: 프로토콜 매크로 데이터

7-2-2 파일 메모리용 파일의 작성과 저장

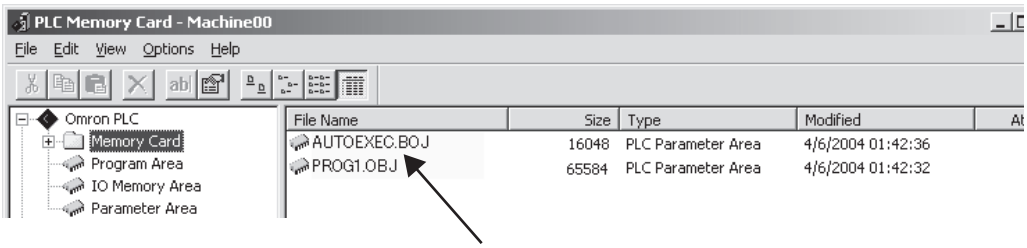
사용자 프로그램, I/O 메모리, 파라미터 영역 데이터

CX-Programmer의 PLC Memory Card 대화 상자를 사용하여 데이터를 전송합니다.

1 이미 초기화된 메모리 카드를 CPU 유닛에 장착하거나 EM 파일 메모리를 초기화합니다. EM 파일 메모리 초기화에 대해서는 7-1-2 파일 메모리 초기화를 참조하십시오.

2 CX-Programmer를 CPU 유닛과 온라인으로 접속합니다.

3 PLC Memory Card 대화 상자에서 전송처(즉, 메모리 카드 또는 EM 파일 메모리)를 선택한 다음 File 메뉴에서 Transfer를 선택합니다.
CPU 유닛의 데이터(즉, 사용자 프로그램, I/O 메모리, 파라미터 영역)가 메모리 카드 또는 EM 파일 메모리에 파일로 저장됩니다.



기동 시 자동 전송되는 프로그램 파일

심볼 테이블, 코멘트, 프로그램 인덱스

CX-Programmer를 사용하여 데이터를 전송합니다.

1 포맷된 메모리 카드를 CPU 유닛에 삽입하거나 EM 파일 메모리를 포맷합니다. EM 파일 메모리의 초기화에 대해서는 7-1-2 파일 메모리 초기화를 참조하십시오.

2 CX-Programmer를 CPU 유닛과 온라인으로 접속합니다.

3 PLC 메뉴에서 Transfer를 선택한 다음 To PLC 또는 From PLC를 선택합니다.

4 전송할 데이터로 Symbols 또는 Comments 중 하나를 선택합니다.

주 CPU 유닛에 메모리 카드가 장착되어 있으면 메모리 카드로만 데이터를 전송할 수 있습니다. (EM 파일 메모리로는 전송할 수 없습니다.)

AI 번역본
-For Reference Only-

7-3 파일 메모리 조작

7-3-1 파일 메모리 조작의 종류

파일 메모리를 사용하기 위해 다음을 수행할 수 있습니다:

- CX-Programmer에서의 조작 순서
- 기동 시 자동 전송
- 간이 백업 기능
- FREAD(700) 및 FWRIT(701) 명령어
- 보조 기억 영역 제어 비트를 사용한 전체 프로그램 교체
- FINS 커맨드

CX-Programmer에서의 조작 순서

다음 조작 순서에 대한 자세한 내용은 CX-Programmer 오퍼레이션 매뉴얼을 참조하십시오.

● CX-Programmer의 Memory Card 대화 상자에서의 데이터 전송

다음 파일을 작성, 읽기, 쓰기 할 수 있습니다.

- 기동 시 자동 전송되는 파일(프로그램/네트워크 심볼 파일, 데이터 파일, 파라미터 파일)
- 범용 파일(프로그램/네트워크 심볼 파일, 데이터 파일, 파라미터 파일)
- CX-Programmer 파일(심볼 테이블, 코멘트 파일, 프로그램 인덱스 파일)

● 온라인/오프라인 데이터 전송

범용 파일(즉, 프로그램/네트워크 심볼 파일, 데이터 파일, 파라미터 파일)은 온라인으로 PLC에 전송하거나 오프라인으로 컴퓨터 메모리 디바이스에 전송할 수 있습니다.

참고

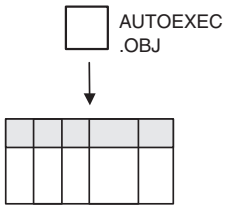
CX-Programmer의 Memory Card 대화 상자에서 파일 조작을 수행할 때는 다음 사항에 주의하십시오.

- 메모리 카드에 파일이 많이 존재하면 Memory Card 대화 상자에서 파일을 확인하지 못할 수 있습니다. 기준으로, 메모리 카드의 루트 디렉터리와 각 서브디렉터리에 각각 1,000개를 넘지 않게 파일을 저장하십시오.
- Memory Card 대화 상자에서 PC와 PLC 사이에 많은 파일을 전송하면 에러가 발생할 수 있습니다. 기준으로, 한 번에 1,000개를 넘지 않게 파일을 전송하십시오.
- Memory Card 대화 상자에서 PC와 PLC 사이에 큰 파일을 전송하지 마십시오. 10 MB를 넘으면 전송 시간이 10분 이상, 100 MB를 넘으면 1시간 이상 걸립니다.

AI 번역본 -For Reference Only-

기동 시 자동 전송되는 파일

전환용 프로그램, 파라미터, I/O 메모리를 메모리 카드에 저장하고 기동 시 메모리 카드에서 읽어 들일 수 있습니다.



자세한 내용은 10-3-5 기동 시 자동 전송을 참조하십시오.

간이 백업 조작

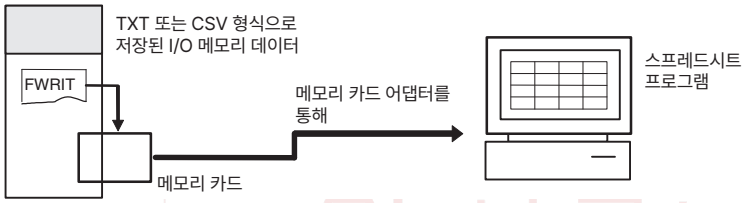
간이 백업 조작을 사용하면 프로그래밍 디바이스 없이 CPU 유닛, 특수 I/O 유닛, CPU 버스 유닛의 데이터를 포함한 PLC의 모든 데이터를 메모리 카드에 백업하고 메모리 카드에서 복원할 수 있습니다. (자세한 내용은 CJ2 CPU 유닛 하드웨어 사용자 매뉴얼 (Cat. No. W472) 또는 CJ3 CPU 유닛 하드웨어 사용자 매뉴얼 (Cat. No. W664)을 참조하십시오.)

백업 • 복원 조작

PLC Backup Tool을 사용하면 CPU 유닛, 특수 I/O 유닛, CPU 버스 유닛의 데이터를 포함한 PLC의 모든 데이터를 컴퓨터에 백업하고 복원할 수 있습니다. (자세한 내용은 CJ2 CPU 유닛 하드웨어 사용자 매뉴얼 (Cat. No. W472) 또는 CJ3 CPU 유닛 하드웨어 사용자 매뉴얼 (Cat. No. W664)의 Section 8 백업 조작을 참조하십시오.)

FWRIT(701)/FREAD(700) 명령어

사용자 프로그램에서 FWRIT(701) 명령어를 사용하면 I/O 메모리 데이터를 텍스트 또는 CSV 형식으로 메모리 카드나 EM 파일 메모리에 저장할 수 있습니다. 그 후 메모리 카드 어댑터를 통해 컴퓨터로 전송하여 스프레드시트 프로그램으로 편집할 수 있습니다.

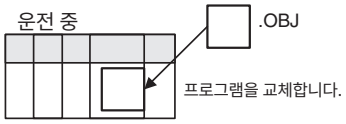


반대로, 특수 I/O 유닛 설정 등의 데이터를 스프레드시트 프로그램에서 텍스트 또는 CSV 형식으로 작성하여 메모리 카드에 저장하고, FREAD(700) 명령어로 CPU 유닛에 읽어 들일 수 있습니다.
자세한 내용은 CS/CJ/NSJ 시리즈 명령어 레퍼런스 매뉴얼 (Cat. No. W474)을 참조하십시오

For Reference Only

보조 기억 영역 제어 비트를 사용한 운전 중 사용자 프로그램 교체

보조 기억 영역 제어 비트를 사용하면 프로그래밍 디바이스 없이 프로그램 자체에서 운전 중에 사용자 프로그램 전체를 교체할 수 있습니다. (자세한 내용은 10-5-4 운전 중 전체 프로그램 교체를 참조하십시오.)



FINS 커맨드

파일 메모리용 FINS 커맨드를 사용하여 파일 메모리를 조작할 수 있습니다. FINS 커맨드는 호스트 링크 프로토콜로 접속된 컴퓨터에서 보낼 수 있습니다. 네트워크상의 다른 PLC가 CMND(490)으로 FINS 커맨드를 보내거나, 자체 PLC가 사용자 프로그램에서 CMND(490)으로 FINS 커맨드를 보낼 수 있습니다. (자세한 내용은 통신 커맨드 레퍼런스 매뉴얼 (Cat. No. W474) 또는 커맨드 레퍼런스 매뉴얼 (Cat. No. W342)을 참조하십시오.)

7-3-2 파일 암호화

파일을 암호화할 수 있습니다. 파일을 암호화하면 안전하게 주고받을 수 있습니다.

자세한 내용은 10-6-7 파일 암호화 기능을 참조하십시오.

7-3-3 파일 메모리 조작 순서와 파일 메모리 파일

다음 표는 파일 메모리 조작의 종류별로 조작할 수 있는 파일 메모리 파일을 정리한 것입니다.

읽기: 파일 메모리에서 CPU 유닛으로 파일을 전송합니다.

쓰기: CPU 유닛에서 파일 메모리로 파일을 전송합니다.

OK: 가능, ---: 불가능

조작 순서	매체	설명	파일 종류				
			프로그램/ 네트워크 심볼 파일	데이터 파일	파라미터 영역 데이터	심볼 테이블, 코멘트 파일, 프로그램 인덱스 파일	유닛 백업 파일
CX-Programmer에서의 조작 순서	메모리 카드, EM 파일 메모리 또는 컴퓨터 메모리 디바이스	작성	OK (범용 파일) OK (기동 시 자동 전송되는 파일)			OK (CX-Programmer 파일)	---
		읽기					
		쓰기					
		그 밖의 조작					
기동 시 자동 전송	메모리 카드	읽기	OK (기동 시 자동 전송되는 파일)			---	
		쓰기	---				
간이 백업 기능	메모리 카드	읽기	OK (간이 백업 파일)				
		쓰기					
PLC Backup Tool을 사용한 백업 또는 복원	컴퓨터 메모리 디바이스만	읽기	OK (PLC 백업 파일)				
		쓰기					
FREAD(700) 및 FWRIT(701) 명령어	메모리 카드 또는 EM 파일 메모리	하나의 파일에서 데이터를 읽습니다.	---	OK (범용 파일)	---	---	
		하나의 파일에 데이터를 씁니다.					
보조 기억 영역 제어 비트 조작으로 운전 중 전체 프로그램을 교체합니다.	메모리 카드	읽기	OK (범용 파일)	---			
FINS 커맨드	메모리 카드 또는 EM 파일 메모리	읽기	OK (범용 파일)			---	
		쓰기					
		그 밖의 조작					

AI 번역본
-For Reference Only-

기동 시 자동 전송되는 파일

기동 시 사용자 프로그램/네트워크 심볼, 파라미터 데이터, I/O 메모리를 일괄로 읽어 들이기 위한 파일입니다. 파일명은 AUTOEXEC 또는 REPLACE로 고정되어 있습니다. 자세한 내용은 10-3-5 기동 시 자동 전송을 참조하십시오.

간이 백업 파일

이 파일들은 메모리 카드에 저장되며, CPU 유닛 전면의 딥 스위치 또는 메모리 카드의 전원 스위치를 사용하여 PLC의 모든 데이터를 백업합니다. 파일명은 BACKUP□□로 고정되어 있습니다. 자세한 내용은 CJ2 CPU 유닛 하드웨어 사용자 매뉴얼 (Cat. No. W472) 또는 CJ3 CPU 유닛 하드웨어 사용자 매뉴얼 (Cat. No. W664)의 8-3 간이 백업을 참조하십시오.

PLC Backup Tool 파일

이 파일들은 PLC의 모든 데이터를 백업하며, PLC Backup Tool의 백업・복원 조작으로 작성됩니다. 자세한 내용은 CJ2 CPU 유닛 하드웨어 사용자 매뉴얼 (Cat. No. W472) 또는 CJ3 CPU 유닛 하드웨어 사용자 매뉴얼 (Cat. No. W664)의 8-2 컴퓨터를 사용한 데이터 백업을 참조하십시오.

CX-Programmer 파일

이 파일들은 CX-Programmer가 작성합니다. 파일 메모리에 작성되는 파일명은 고정되어 있습니다. 프로젝트를 전송할 때 이들 CX-Programmer 파일의 전송처로 다음 메모리 중 하나를 선택할 수 있습니다.

・코멘트 메모리(CPU 유닛의 플래시 메모리 내)

・메모리 카드

・EM 파일 메모리

종류	이름	내용	설명
심볼 테이블 파일 (네트워크 심볼 제외)	SYMBOLS.SYM SYMBOLS.SBS*1	글로벌 심볼 테이블(네트워크 심볼 제외)과 로컬 심볼 테이블	・CX-Programmer의 글로벌/로컬 심볼 테이블에 있는 변수, 어드레스, 데이터형, I/O 코멘트 정보가 포함됩니다. 네트워크 심볼은 포함되지 않습니다. ・CX-Programmer의 PLC 어드레스 자동 할당에서 설정한 데이터도 포함됩니다.
코멘트 파일	COMMENTS.CMT COM-MENTS.CMS *1	렁 코멘트와 코멘트(주석)	CX-Programmer의 렁 코멘트 및 코멘트 정보가 들어 있습니다.
프로그램 인덱스 파일	PROGRAM.IDX PROGRAM.IDS *1	섹션 이름, 섹션 코멘트	CX-Programmer의 섹션 구분자 정보가 들어 있습니다. 다만 구분자 위치는 프로그램의 섹션 구분 명령어에 따라 달라집니다.

*1 파일 암호화 기능이 유효한 경우에는 파일 확장자가 다릅니다.

주 CX-Programmer에서는 위 표의 심볼 테이블 파일과 코멘트 파일(네트워크 심볼 제외)을 CX-Programmer와 퍼스널 컴퓨터 RAM 사이, 그리고 퍼스널 컴퓨터 RAM과 메모리 기억 디바이스 사이에서 온라인으로 전송할 수 있습니다.

CX-Programmer에서 CPU 유닛으로 프로젝트를 전송하면 심볼 테이블, 코멘트 파일, 프로그램 인덱스 파일을 CPU 유닛에 내장된 플래시 메모리에 저장할 수 있습니다. 간이 백업 조작을 수행하면 코멘트 메모리에 저장된 심볼 테이블, 코멘트 파일, 프로그램 인덱스 파일이 자동으로 백업 파일로 작성되어 메모리 카드에 저장됩니다.

사용 금지
-For Reference Only

범용 파일

이 파일들은 CX-Programmer, FINS 커맨드, 명령어, 보조 기억 영역 비트를 사용하여 읽기, 쓰기 및 그 밖의 조작을 수행할 수 있습니다.

파일명은 사용자가 정의합니다.

파일 종류	파일명*1	확장자	내용	설명	
프로그램/네트워크 심볼 파일	********	.OBJ .OBS*4	· 사용자 프로그램 · 네트워크 심볼*2	CPU 유닛의 사이클릭 태스크와 인터럽트 태스크에 있는 프로그램과 네트워크 심볼.	
파라미터 파일	********	.STD .STS*4	· PLC 설정 · 등록 I/O 테이블 · 라우팅 테이블 · CPU 버스 유닛 셋업 영역	· CPU 유닛의 파라미터 영역 데이터 · 사용자가 파일 내의 데이터를 구별할 필요는 없습니다.	
데이터 파일	********	.IOM .IOS*4	지정된 범위의 I/O 메모리	· 하나의 영역에서 선두 워드부터 종료 워드까지 워드 (16비트) 단위의 데이터. · 영역은 CIO, HR(유지 영역), WR(워크 영역), AR(보조 기억 영역), DM(데이터 메모리), EM(확장 메모리)입니다.	바이너리 데이터 형식
		.TXT*5			텍스트 형식(구분자나 탭 없음)*3
		.CSV*5			CSV 형식(콤마 구분)*3

*1 ********에는 ASCII 문자 8자 이하를 설정하십시오.

*2 CJ2H-CPU□□(-EIP 없음) 버전 1.5 이전 및 CJ2M-CPU1□ 버전 2.1 이전은 지원되지 않습니다.

*3 텍스트 및 CSV 데이터 파일은 FWRIT(701)과 FREAD(700) 명령어로만 읽고 쓸 수 있습니다. CX-Programmer에서는 읽거나 쓸 수 없습니다.

*4 파일 암호화 기능이 유효한 경우에는 파일 확장자가 다릅니다.

*5 파일 암호화 기능이 유효한 경우에는 FWRIT/FREAD 명령어에서 TXT 및 CSV 형식 파일을 사용할 수 없습니다.

AI 번역본
-For Reference Only-

● 범용 데이터 파일

데이터 파일의 종류

범용 데이터 파일에는 파일 확장자가 IOM/.IOS, TXT, CSV인 세 종류가 있습니다.

확장자	내용	용도
.IOM .IOS*1	바이너리 형식 CS/CJ 시리즈 데이터 형식	I/O 메모리 백업
.TXT*1,2	이 데이터 형식에서는 I/O 메모리의 1워드 또는 2워드 필드가 ASCII 데이터로 변환됩니다*2. 레코드는 캐리지 리턴으로 구분할 수 있습니다.	스프레드시트 소프트웨어와 데이터 교환
.CSV*2		

*1 파일 암호화 기능이 유효한 경우에는 파일 확장자가 다릅니다.

*2 TXT 및 CSV 데이터 파일의 읽기와 쓰기:
TXT 및 CSV 데이터 파일은 FREAD(700)과 FWRIT(701)으로만 읽고 쓸 수 있습니다.
파일 암호화 기능이 유효한 경우에는 FWRIT/FREAD 명령어에서 TXT 및 CSV 형식 파일을 사용할 수 없습니다.

텍스트 및 CSV 파일에는 다음 6가지 데이터 형식이 사용됩니다.

확장자	데이터 형식	내용	
		필드당 I/O 메모리 사이즈	구분자
.TXT*1	구분 없는 워드	1워드	없음
	구분 없는 더블 워드	2워드	없음
	탭 구분 워드	1워드	탭 코드
	탭 구분 더블 워드	2워드	탭 코드
.CSV*1	콤마 구분 워드	1워드	콤마
	콤마 구분 더블 워드	2워드	콤마

*1 문자에 관한 주의사항:
TXT 또는 CSV 파일에 16진수 문자(0~9, A~F, a~f) 이외의 문자가 들어 있으면 I/O 메모리에 데이터를 올바르게 쓸 수 없습니다.
파일 암호화 기능이 유효한 경우에는 FWRIT/FREAD 명령어에서 TXT 및 CSV 형식 파일을 사용할 수 없습니다.

- 필드 사이즈에 관한 주의사항:
  워드를 사용하는 경우, TXT 또는 CSV 파일에 4자리 16진수가 아닌 필드가 들어 있으면 I/O 메모리에 데이터를 올바르게 쓸 수 없습니다. 마찬가지로 더블 워드를 사용하는 경우, 8자리 16진수가 아닌 필드가 들어 있으면 올바르게 쓸 수 없습니다.
- 저장 순서:
  워드를 사용하는 경우, I/O 메모리 데이터는 ASCII로 변환되어 I/O 메모리 어드레스가 낮은 쪽에서 높은 쪽 순서로 1워드 필드에 저장됩니다.
  더블 워드를 사용하는 경우, I/O 메모리 데이터는 ASCII로 변환되어 I/O 메모리 어드레스가 낮은 쪽에서 높은 쪽 순서로 2워드 필드에 저장됩니다. (2워드 필드 안에서는 상위 어드레스 워드가 먼저, 하위 어드레스 워드가 나중에 저장됩니다.)
- 구분자:
  구분자가 없으면 필드가 연속으로 채워져 저장됩니다. 콤마로 구분하면 필드 사이에 콤마가 삽입되어 저장됩니다. 탭으로 구분하면 필드 사이에 탭 코드가 삽입되어 저장됩니다.
  FREAD(700)에서 구분자(콤마 또는 탭)를 지정하면, 데이터는 1워드 구분자(콤마 또는 탭)로 구분된 데이터로 읽어 들여집니다.
- 캐리지 리턴:
  캐리지 리턴을 사용하지 않으면 데이터가 연속으로 채워집니다.
  캐리지 리턴을 사용하면 지정한 필드 수 뒤에 캐리지 리턴 코드가 삽입됩니다.
  파일에 캐리지 리턴을 사용한 경우에는 FREAD(700)/FWRIT(701) 명령어에서 파일 선두부터의 오프셋(읽기 시작 워드 또는 쓰기 시작 워드)을 지정할 수 없습니다.

시 내 영 업 부
For Reference Only

・필드 수:
파일 전체의 데이터량은 FWRIT(701) 명령어에서 지정한 필드 수(쓰기 항목 수)와 필드당 워드 수에 따라 달라집니다.
사이즈는 필드 수로 지정합니다.
워드 데이터의 경우, I/O 메모리의 1워드 = 1필드
롱 워드 데이터의 경우, I/O 메모리의 2워드 = 1필드.

데이터 파일명

데이터 파일에는 어떤 데이터가 저장되어 있는지, 즉 어느 메모리 영역이 저장되어 있는지를 나타내는 정보가 들어 있지 않습니다. 파일 관리를 돕기 위해 아래 예와 같이 내용을 알 수 있는 파일명을 지정하십시오.
예: D00100.IOM, CIO0020.IOM
데이터 파일(IOM, TXT, CSV)에 원래 쓰여 있던 데이터가 같은 영역의 것이 아니더라도, 파일 선두부터의 데이터는 I/O 메모리에서 지정한 어드레스부터 쓰입니다. 예를 들어 파일 안의 CIO 데이터를 프로그래밍 디바이스에서 DM 영역에 쓰면, 영역이 다르다는 표시 없이 CPU 유닛의 DM 영역으로 읽어 들여집니다.

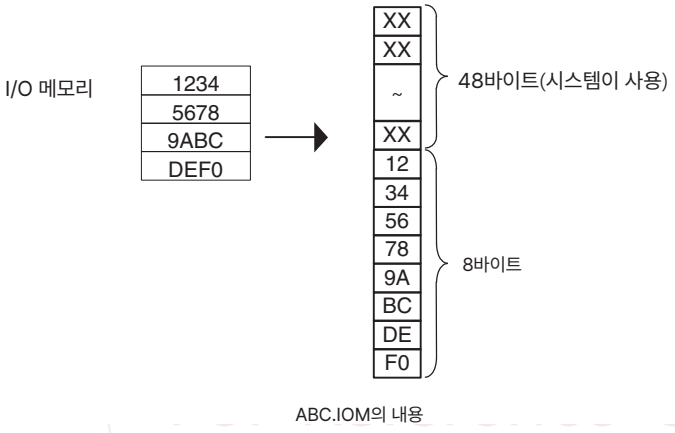
참고

TXT 및 CSV 형식의 데이터 파일에는 16진수(0~9, A~F) 데이터가 들어 있어, I/O 메모리의 수치 데이터를 스프레드시트 프로그램과 주고받을 수 있습니다.

I/O 메모리의 ASCII 데이터는 TWRIT(704)(텍스트 파일 쓰기) 래더 프로그래밍 명령어 또는 ST 언어의 메모리 카드 처리 함수 WRITE_TEXT ( )를 사용하여 텍스트 파일로 저장할 수 있습니다.

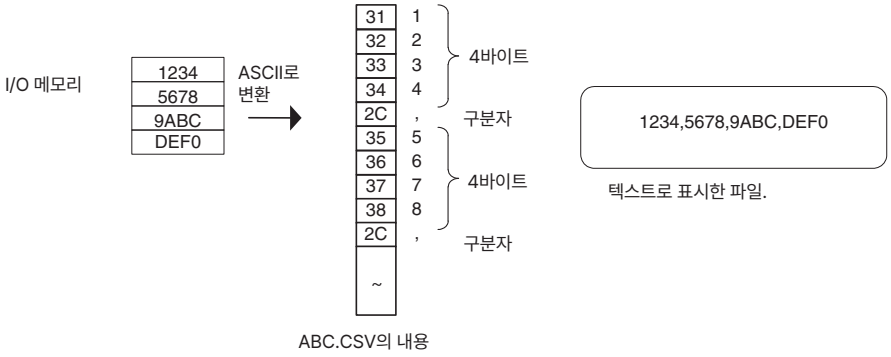
참고

다음 그림은 I/O 메모리의 4워드(1234 hex, 5678 hex, 9ABC hex, DEF0 hex)가 들어 있는 데이터 파일(ABC.IOM)의 바이너리 데이터 구조를 나타냅니다. 다만 통상 조작에서는 사용자가 데이터 형식을 의식할 필요가 없습니다.
암호화된 파일(.IOS 확장자)의 데이터는 액세스하거나 열람할 수 없습니다.



・1워드 필드를 사용하는 CSV 및 TXT 데이터 파일의 구조

다음 그림은 I/O 메모리의 4워드(1234 hex, 5678 hex, 9ABC hex, DEF0 hex)가 들어 있는 1워드 필드 CSV 데이터 파일(ABC.CSV)의 데이터 구조를 나타냅니다.



・2워드 필드를 사용하는 CSV 및 TXT 데이터 파일의 구조

다음 그림은 I/O 메모리의 4워드(1234 Hex, 5678 Hex, 9ABC Hex, DEF0 Hex)가 들어 있는 2워드 필드 CSV 데이터 파일(ABC.CSV)의 데이터 구조를 나타냅니다.



참고

스프레드시트 소프트웨어로 데이터 파일 작성

Microsoft Excel 등의 스프레드시트 소프트웨어로 TXT 및 CSV 데이터 파일을 작성하려면 다음 순서를 따르십시오.

・셀의 내용을 문자로 설정합니다.

・1워드 필드를 사용하는 경우에는 각 셀에 4문자를, 2워드 필드를 사용하는 경우에는 8문자를 입력합니다. 예를 들어 1워드 필드를 사용하는 경우에는 A만이 아니라 000A를 입력합니다.

・셀에는 반드시 16진수 문자(0~9, A~F, a~f)만 입력하십시오. 그 밖의 문자와 코드는 사용할 수 없습니다.

I/O 메모리에 16진수를 저장하려면 스프레드시트의 10진수 입력을 16진수로 변환하는 것이 편리합니다. 16진수로 변환하려면 다음 순서를 따르십시오.

AI 번역본
-For Reference Only-

(1) Tools 메뉴에서 Add-Ins...를 선택합니다.

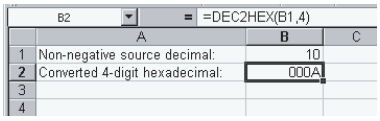
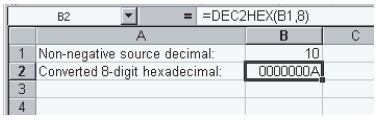
(2) Add-Ins 메뉴에서 Analysis ToolPak을 선택합니다.

(3) 함수를 사용할 셀에서 Insert 메뉴의 Function을 선택합니다.

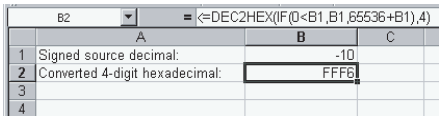
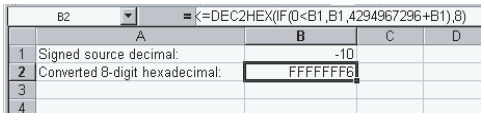
(4) Category 필드의 Engineering에서 DEC2HEX (number, digits)를 선택합니다.

(5) 4자리 16진수로 변환하는 경우, number 변수에 다음을 입력합니다:
IF(0<=cell location,cell location,65535+cell location)
8자리 16진수로 변환하는 경우, number 변수에 다음을 입력합니다:
IF(0<=cell location,cell location,4294967296+cell location)

예 1: 음수가 아닌 10진수 값 입력

항목	부호 없는 10진수를 4자리 16진수로 변환	부호 없는 10진수를 8자리 16진수로 변환
사용 함수	DEC2HEX(*cell_location,4*)	DEC2HEX(*cell_location,8*)
예	10진수로 10을 입력하여 4자리 16진수 000A로 변환합니다. 	10진수로 10을 입력하여 8자리 16진수 0000000A로 변환합니다. 

예 2: 부호 있는 10진수 값 입력

항목	부호 없는 10진수를 4자리 16진수로 변환	부호 없는 10진수를 8자리 16진수로 변환
사용 함수	DEC2HEX(IF(0<=*cell_location,cell_location,65536+cell_location*),4)	DEC2HEX(IF(0<=*cell_location,cell_location, 4294967296+cell_location*),8)
예	10진수로 -10을 입력하여 4자리 16진수 FFF6으로 변환합니다. 	10진수로 -10을 입력하여 8자리 16진수 FFFFFFF6으로 변환합니다. 

7-3-4 파일 사용 시의 제한

파일은 DOS 형식으로 포맷되므로 Windows 컴퓨터에서 통상적인 파일로 사용할 수 있습니다.

파일 형식: 8문자 파일명과 3문자 확장자
□□□□□□□□ □□□

● 파일명

파일은 다음 표와 같이 파일명과 확장자로 식별됩니다. 파일명에는 다음 문자를 사용합니다: 영문자 a~z, A~Z, 숫자 0~9, !, &, $, #,', {, }, -, ^, (,), _

파일명에는 다음 문자를 사용할 수 없습니다: ,, ., /, \, ?, *, ", :, ;, <, >, =, +, 공백, 2바이트 문자.

파일 확장자는 저장되는 파일의 종류에 따라 달라집니다.

AI 번역본

● 디렉터리

CS/CJ 시리즈 CPU 유닛은 서브디렉터리에 있는 파일에 액세스할 수 있습니다. 파일이 저장되어 있는 파일 메모리 내의 디렉터리 위치를 지정하십시오. 프로그래밍 콘솔을 사용하지 않는 한, 디렉터리는 (루트 디렉터리를 포함하여) 최대 5계층까지 지정할 수 있습니다. 디렉터리 경로의 최대 길이는 65문자입니다. Windows 등의 운영체제로 메모리 카드의 서브디렉터리를 작성할 때는 최대 서브디렉터리 계층(5계층)을 넘지 마십시오.

7-3-5 파일 사이즈

파일의 사이즈(바이트)는 다음 표의 계산식으로 구할 수 있습니다.

파일 종류	파일 사이즈
데이터 파일(.IOM)	(워드 수 × 2) + 48바이트 예: DM 영역 전체(D0~D32767) (32,768워드 × 2) + 48 = 65,584바이트
데이터 파일(.TXT 또는 .CSV)	파일 사이즈는 사용하는 구분자와 캐리지 리턴의 수에 따라 달라집니다. 구분자 코드는 1바이트, 캐리지 리턴 코드는 2바이트입니다. 예 1: 구분 없는 워드, 캐리지 리턴 없음 123456789ABCDEF012345678은 24바이트를 차지합니다. 예 2: 구분된 워드, 2필드마다 캐리지 리턴 1234,5678↵ 9ABC,DEF0↵ 1234,5678↵은 33바이트를 차지합니다. 예 3: 구분된 더블 워드, 2필드마다 캐리지 리턴 56781234,DEF01234↵ 56781234↵은 29바이트를 차지합니다.
프로그램 파일(.OBJ)	(사용 스텝 수 × 4) + 48바이트*1
파라미터 파일(.STD)	16,048바이트

*1 프로그램 파일의 스텝 수는 전체 UM 스텝에서 사용 가능한 UM 스텝을 빼서 계산합니다. 이 값들은 CX-Programmer의 Cross-Reference Report에 표시됩니다. 자세한 내용은 CX-Programmer 오퍼레이션 매뉴얼을 참조하십시오.

주 파일이 암호화된 경우에는 사이즈가 위 계산식과 달라집니다.

AI 번역본
-For Reference Only-

7-3-6 지원 소프트웨어와 파일 메모리 파일의 관계

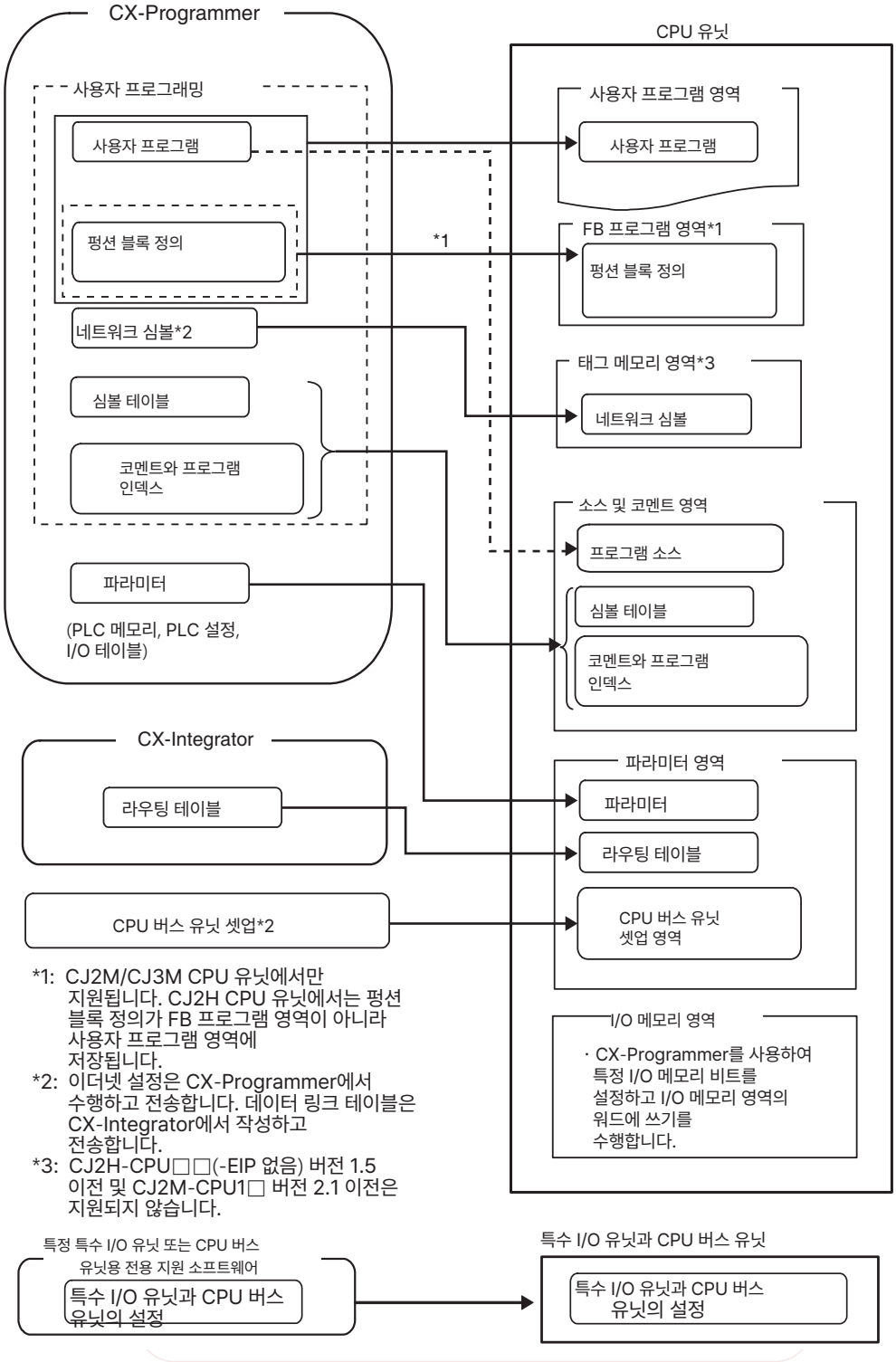
다음 표는 각 지원 소프트웨어 패키지의 파일과 파일 메모리 파일의 관계를 나타냅니다.

지원 소프트웨어	지원 소프트웨어 파일	내용	파일 메모리 파일	저장 영역
CX-Programmer	CX-Programmer 프로젝트 파일 (확장자:.cxp)	사용자 프로그램/네트워크 심볼	프로그램/네트워크 심볼 파일(확장자: OBJ) (확장자: .OBS*2)	CPU 유닛
		펑션 블록		
		심볼 테이블(심볼 이름과 어드레스 대응, I/O 코멘트 포함) 주: 네트워크 심볼은 제외합니다.	심볼 테이블(네트워크 심볼 제외) (SYMBOLS.SYM) (SYMBOLS.SBS *2)	
		코멘트(I/O 코멘트를 제외한 행 코멘트와 주석)	코멘트 파일 (COMMENTS.CMT) (COMMENTS.CMS *2)	
		PLC 이름	파라미터 파일(확장자: STD)*1 (확장자: STS *2)	
		PLC 설정		
		등록 I/O 테이블		
		이더넷 유닛 설정		
	데이터 링크 테이블 파일 (.cl2)	Controller Link 데이터 링크 테이블		
	CX-Programmer 프로젝트 파일 (확장자: .cxp) 유닛 설정(XML 파일): CPS 내용	PLC 메모리(I/O 메모리) (할당 I/O 메모리와 할당 DM 영역 워드가 이어집니다.)	데이터 파일(확장자: IOM) (확장자: IOS *2)	
		할당 I/O 메모리와 할당 DM 영역 워드		
CX-Integrator	라우팅 테이블 파일 (확장자: .rtg)	라우팅 테이블	파라미터 파일 *1 (확장자: STD) (확장자: STS *2)	
L-net 지원 소프트웨어	전용 파일(.csv)	FL-net 설정		
특정 특수 I/O 유닛 또는 CPU 버스 유닛용 전용 지원 소프트웨어	특수 I/O 유닛과 CPU 버스 유닛용 전용 지원 소프트웨어가 작성한 파일	위 파일에 포함되지 않는 데이터	유닛 백업 파일 (BACKUP□□.PRM)	특수 I/O 유닛과 CPU 버스 유닛

*1 지원 소프트웨어 애플리케이션으로 작성한 파일은 파일 메모리에 하나의 파일로 작성됩니다.

*2 파일 암호화 기능이 유효한 경우에는 파일 확장자가 다릅니다.

AI 번역본
-For Reference Only-


CX-Programmer
사용자 프로그래밍
사용자 프로그램
펑션 블록 정의
네트워크 심볼*2
심볼 테이블
코멘트와 프로그램 인덱스
파라미터
(PLC 메모리, PLC 설정, I/O 테이블)
CX-Integrator
라우팅 테이블
CPU 버스 유닛 셋업*2
*1
CPU 유닛
사용자 프로그램 영역
사용자 프로그램
FB 프로그램 영역*1
펑션 블록 정의
태그 메모리 영역*3
네트워크 심볼
소스 및 코멘트 영역
프로그램 소스
심볼 테이블
코멘트와 프로그램 인덱스
파라미터 영역
파라미터
라우팅 테이블
CPU 버스 유닛 셋업 영역
I/O 메모리 영역
· CX-Programmer를 사용하여 특정 I/O 메모리 비트를 설정하고 I/O 메모리 영역의 워드에 쓰기를 수행합니다.
*1: CJ2M/CJ3M CPU 유닛에서만 지원됩니다. CJ2H CPU 유닛에서는 펑션 블록 정의가 FB 프로그램 영역이 아니라 사용자 프로그램 영역에 저장됩니다.
*2: 이더넷 설정은 CX-Programmer에서 수행하고 전송합니다. 데이터 링크 테이블은 CX-Integrator에서 작성하고 전송합니다.
*3: CJ2H-CPU□□(-EIP 없음) 버전 1.5 이전 및 CJ2M-CPU1□ 버전 2.1 이전은 지원되지 않습니다.
특정 특수 I/O 유닛 또는 CPU 버스 유닛용 전용 지원 소프트웨어
특수 I/O 유닛과 CPU 버스 유닛의 설정
특수 I/O 유닛과 CPU 버스 유닛
특수 I/O 유닛과 CPU 버스 유닛의 설정

AI 번역본
-For Reference Only-

8

I/O 할당과 유닛 설정

이 섹션에서는 기본 I/O 유닛, 특수 I/O 유닛, CPU 버스 유닛에 메모리를 할당하는 방법과 특수 I/O 유닛 및 CPU 버스 유닛을 설정하는 방법을 설명합니다.

8-1 I/O 할당 8-2

8-1-1 I/O 할당 8-2

8-1-2 자동 할당 8-5

8-1-3 수동 할당 8-9

8-1-4 I/O 테이블 에러와 주의사항 8-17

8-2 CPU 버스 유닛과 특수 I/O 유닛 설정 8-20

8-2-1 파라미터 설정 8-20

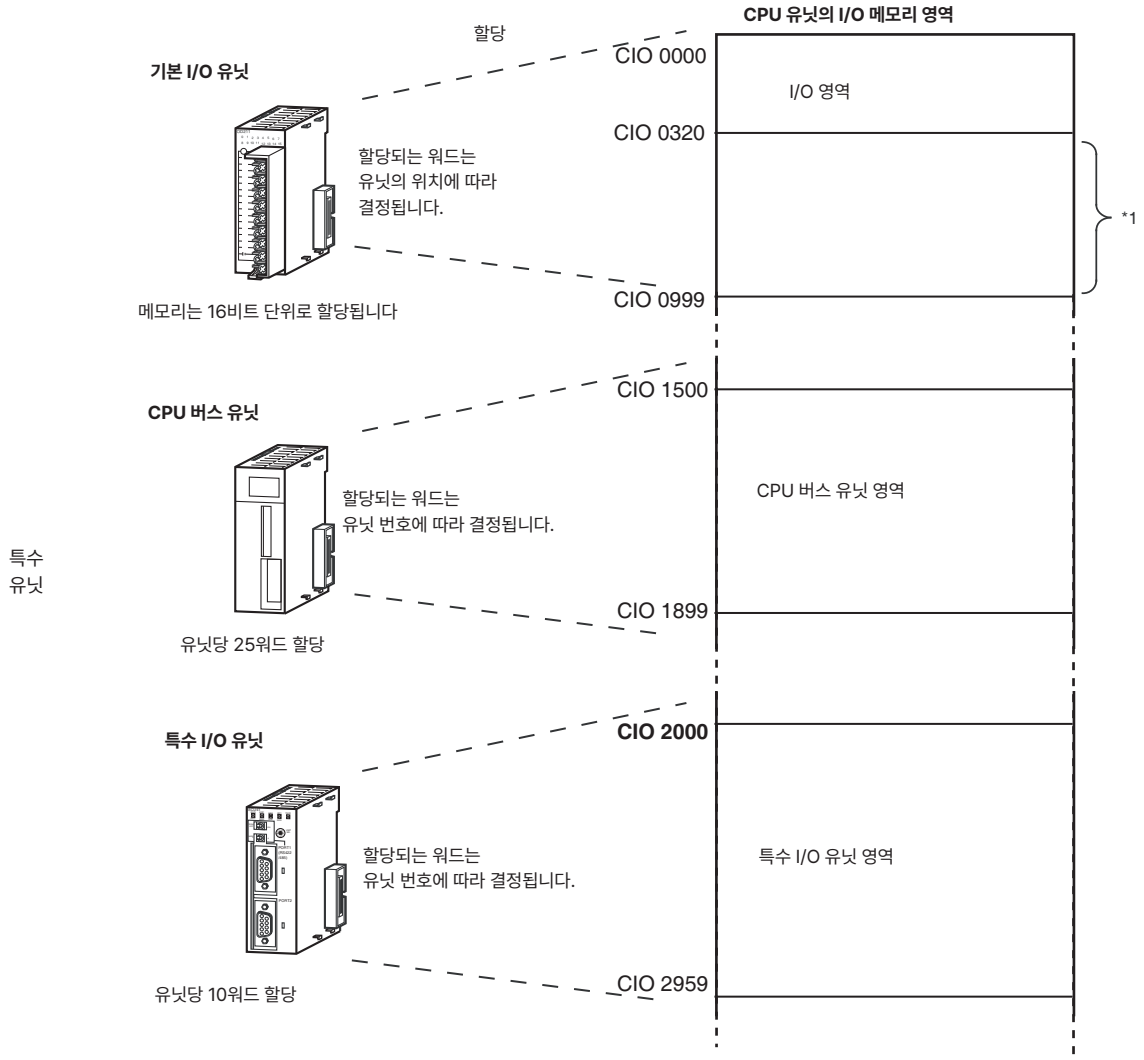
8-2-2 데이터 교환 8-24

AI 번역본
-For Reference Only-

8-1 I/O 할당

8-1-1 I/O 할당

CPU 유닛과 다른 유닛 사이에서 데이터를 주고받기 위해 CPU 유닛의 I/O 메모리에 워드를 할당하는 것을 I/O 할당이라고 합니다. 기본 I/O 유닛, 특수 I/O 유닛, CPU 버스 유닛에는 각각 다른 방식으로 메모리가 할당됩니다.



*1 각 랙의 선두 워드는 CX-Programmer에서 CIO 0000~CIO 0999 범위의 어드레스로 설정할 수 있습니다(디폴트 설정: CIO 0000부터 연속 할당).

I/O 할당 방법

I/O를 할당하려면 다음 두 가지 방법 중 하나를 사용합니다.

・자동 할당(디폴트)
유닛의 전원을 ON하면 접속된 유닛의 상태(슬롯 위치, 유닛 번호 등)에 따라 I/O가 할당됩니다.

・수동 할당(I/O 테이블 작성)
CX-Programmer로 I/O 테이블을 작성하여 원하는 할당을 지정할 수 있습니다. I/O 테이블을 작성한 경우, CPU 유닛의 전원을 ON했을 때 I/O 테이블이 접속된 유닛의 상태와 일치하지 않으면 에러가 발생합니다.

다음 표는 이 두 방법의 차이를 나타냅니다. 시스템의 목적과 필요에 따라 방법을 선택하십시오.

할당 방법 항목	자동 할당	수동 할당	
		유닛을 접속한 상태에서 I/O 테이블 작성	사용자 설정 할당으로 I/O 테이블 작성
용도	・유닛의 오접속 확인이 필요하지 않은 경우에 이 방법을 사용합니다. ・사용자 설정 할당이 필요하지 않은 경우에 이 방법을 사용합니다 (미사용 워드 예약 포함).	유닛의 오접속을 확인할 때 이 방법을 사용합니다.	・사용자 설정 할당을 작성할 때 이 방법을 사용합니다(미사용 워드 예약 포함). ・실제 I/O 유닛 없이 (즉, 오프라인으로) I/O를 할당할 때 이 방법을 사용합니다.
I/O 테이블 작성	필요 없음.	필요.	
접속된 유닛과 I/O 테이블의 대조	유닛을 대조하지 않음.	유닛을 대조함. (접속된 유닛이 I/O 테이블과 일치하지 않으면 운전을 개시할 수 없습니다.)	
I/O 할당에 사용하는 조작	자동으로 할당됩니다.	수동으로 할당됩니다.	
CX-Programmer	필요 없음.	필요	
I/O 테이블 작성에 접속된 유닛이 필요한지 여부	필요.	필요	필요 없음.
할당 방법	자동 할당(CPU 유닛부터 순서대로)		수동 할당(그룹마다 선두 어드레스를 지정할 수 있습니다.)

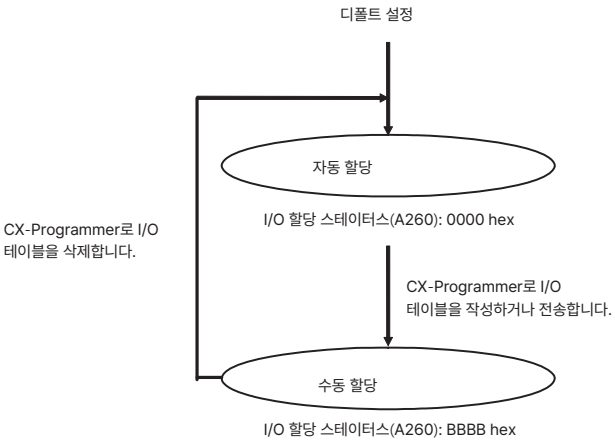
● I/O 할당 방법 확인

I/O 할당 스테이터스 워드(A260)를 사용하여 I/O 할당이 수동으로 설정되었는지 자동으로 설정되었는지 확인할 수 있습니다.

어드레스	이름	내용	
A260	I/O 할당 스테이터스	0000 hex	I/O 자동 할당
		BBBB hex	수동 할당

● I/O 할당 스테이터스의 천이

I/O 할당 방법은 다음과 같이 변화합니다.



CX-Programmer로 I/O 테이블을 삭제하면, I/O 할당 스테이터스가 자동 할당으로 돌아감과 동시에 각 랙의 선두 워드가 클리어됩니다. 또한 CPU 유닛의 시스템 셋업 영역이 초기화됩니다.

AI 번역본
-For Reference Only-

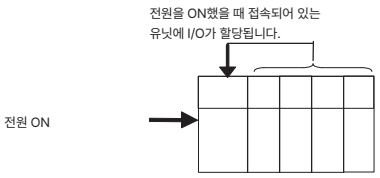
8-1-2 자동 할당

자동 할당(디폴트)

접속된 유닛의 상태에 따라 I/O를 할당하려면 이 방법을 사용합니다. I/O가 자동으로 할당되므로 프로그래밍 디바이스에서의 조작은 필요하지 않습니다.

자동 할당을 사용하면 PLC의 전원을 ON할 때마다 I/O 할당이 갱신됩니다.

수동 할당에서 자동 할당으로 바꾸려면 프로그래밍 디바이스로 I/O 테이블을 삭제하십시오.



참고

수동 할당과 달리 자동 할당에서는 I/O 테이블을 작성하지 않으므로 접속된 유닛을 대조하지 않습니다.

할당 방법

자동 할당에서는 다음 표와 같이 접속된 유닛에 I/O가 할당됩니다. 사용자의 조작은 필요하지 않습니다.

유닛 분류	할당 방법
기본 I/O 유닛	위치에 따라 순서대로 할당됩니다.
특수 I/O 유닛	전면 로터리 스위치 설정(즉, 유닛 번호)에 따라 할당됩니다.
CPU 버스 유닛	

AI 번역본
-For Reference Only-

유닛 분류별 I/O 할당 방법

이 섹션에서는 유닛 분류별(기본 I/O 유닛, 특수 I/O 유닛, CPU 버스 유닛)의 I/O 할당 방법을 설명합니다.

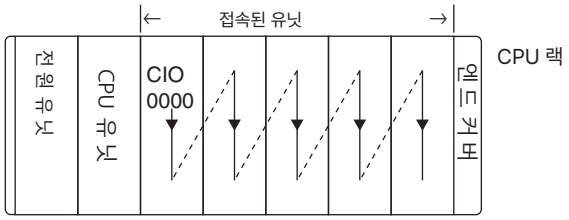
● 기본 I/O 유닛에 대한 I/O 할당

CJ 시리즈 기본 I/O 유닛에는 I/O 영역(CIO 0000~CIO 0319)의 워드가 할당됩니다. CPU 랙 또는 증설 랙에 접속할 수 있습니다. 다음 방법이 사용됩니다.

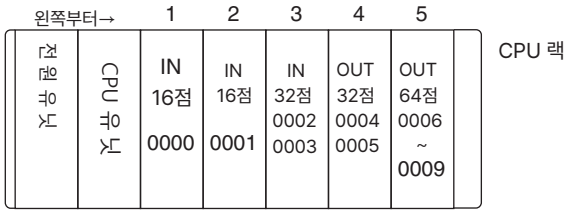
CPU 랙의 기본 I/O 유닛

CPU 랙의 기본 I/O 유닛에는 CPU 유닛에 가장 가까운 유닛부터 왼쪽에서 오른쪽으로 워드가 할당됩니다. 각 유닛에는 필요한 만큼의 워드가 할당됩니다.

주 I/O 1~16점의 유닛에는 16비트가, I/O 17~32점의 유닛에는 32비트가 할당됩니다.



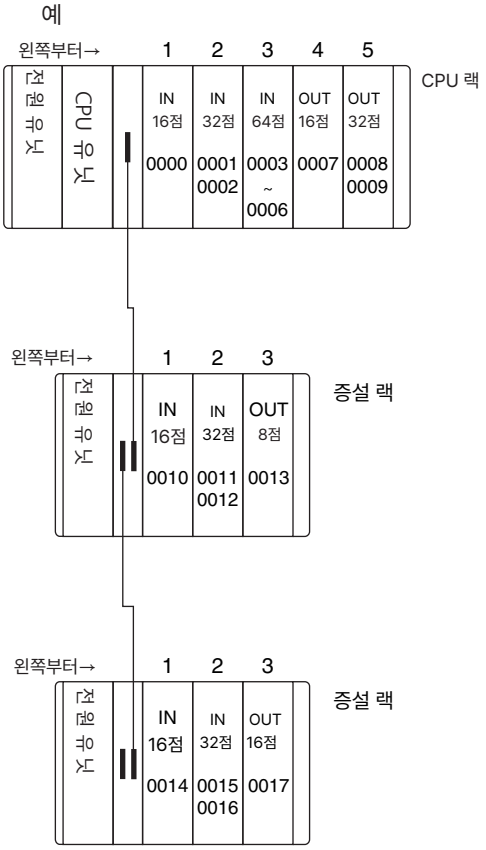
예



AI 번역본
-For Reference Only-

증설 랙의 기본 I/O 유닛

기본 I/O 유닛에 대한 I/O 할당은 CPU 랙에서 CPU 랙에 직접 접속된 증설 랙으로 이어집니다. CPU 랙의 유닛과 마찬가지로 왼쪽에서 오른쪽으로 워드가 할당되고, 각 유닛에는 필요한 만큼의 워드가 할당됩니다.



● 특수 I/O 유닛에 대한 I/O 할당

CJ 시리즈 특수 I/O 유닛에는 유닛에 설정한 유닛 번호에 따라 특수 I/O 유닛 영역 (CIO 2000~CIO 2959)의 10워드가 할당됩니다. 특수 I/O 유닛은 CPU 랙 또는 증설 랙에 장착할 수 있습니다. 다음 표는 특수 I/O 유닛 영역의 어느 워드가 각 유닛에 할당되는지를 나타냅니다.

유닛 번호	워드 수	할당 워드
0	10	CIO 2000~CIO 2009
1	10	CIO 2010~CIO 2019
2	10	CIO 2020~CIO 2029
:	:	:
15	10	CIO 2150~CIO 2159
:	:	:
:	:	:
95	10	CIO 2950~CIO 2959

기본 I/O 유닛에 대한 I/O 할당 시 특수 I/O 유닛은 무시됩니다. 특수 I/O 유닛이 있는 위치에는 I/O 영역의 워드가 할당되지 않습니다.

AI 번역본
-For Reference Only-

● CPU 버스 유닛에 대한 I/O 할당

CJ 시리즈 CPU 버스 유닛에는 유닛에 설정한 유닛 번호에 따라 CPU 버스 유닛 영역 (CIO 1500~CIO 1899)의 25워드가 할당됩니다. CPU 버스 유닛은 CPU 랙 또는 증설 랙에 장착할 수 있습니다. 다음 표는 CPU 버스 유닛 영역의 어느 워드가 각 유닛에 할당되는지를 나타냅니다.

유닛 번호	워드 수	할당 워드
0	25	CIO 1500~CIO 1524
1	25	CIO 1525~CIO 1549
2	25	CIO 1550~CIO 1574
:	:	:
15	25	CIO 1875~CIO 1899

기본 I/O 유닛에 대한 I/O 할당 시 CPU 버스 유닛은 무시됩니다. CPU 버스 유닛이 있는 위치에는 I/O 영역의 워드가 할당되지 않습니다. CPU 버스 유닛의 유닛 번호는 특수 I/O 유닛의 유닛 번호와 별개입니다. CPU 버스 유닛과 특수 I/O 유닛에 같은 유닛 번호를 사용해도 유닛 번호의 중복 사용이 되지 않습니다.

슬롯	유닛	형식	필요 워드 수	할당 워드	유닛 번호	그룹
0	16점 DC 입력 유닛	CJ1W-ID211	1	CIO 0000	---	기본 I/O 유닛
1	아날로그 입력 유닛	CJ1W-AD081-V1	10	CIO 2000~ CIO 2009	0	특수 I/O 유닛
2	시리얼 통신 유닛	CJ1W-SCU42	25	CIO 1525~ CIO 1549	1	CPU 버스 유닛
3	16점 트랜지스터 출력 유닛	CJ1W-OD211	1	CIO 0001	---	기본 I/O 유닛
4	Controller Link 유닛	CJ1W-CLK23	25	CIO 1550~ CIO 1574	2	CPU 버스 유닛
---	CPU 유닛의 내장 EtherNet/IP 포트 *1	CJ3M-CPU35	25	CIO 1500~ CIO 1524	0	CPU 유닛(이 포트는 CPU 버스 유닛으로 취급됩니다.)

*1 CJ2H-CPU6□-EIP 및 CJ2M-CPU3□, CJ3M-CPU3□에서는 내장 EtherNet/IP 포트에 워드가 할당됩니다. 다른 CPU 버스 유닛과 마찬가지로, CJ2H-CPU6□-EIP 또는 CJ2M-CPU3□, CJ3M-CPU3□ 전면의 로터리 스위치로 설정한 유닛 번호에 따라 워드가 할당됩니다.

AI 번역본
-For Reference Only-

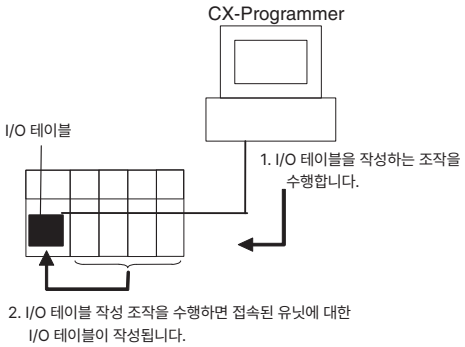
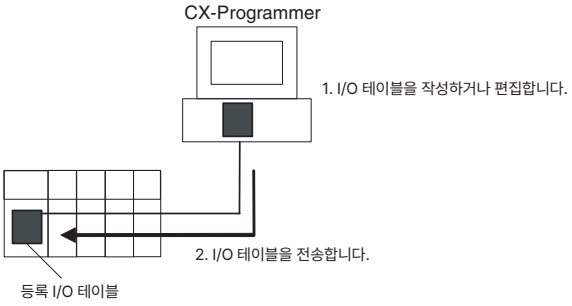
8-1-3 수동 할당

수동 할당

사용자가 작성한 I/O 테이블에 따라 I/O를 할당하려면 수동 할당을 사용합니다. I/O 테이블을 작성하면 전원을 ON했을 때 접속된 유닛이 I/O 테이블과 대조됩니다.

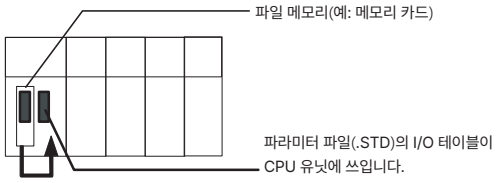
● I/O 테이블 작성

I/O 테이블을 작성하려면 다음 방법 중 하나를 사용합니다.

유닛을 접속한 상태에서 I/O 테이블 작성	사용자 설정 할당으로 I/O 테이블 작성
유닛을 접속하고 CX-Programmer를 온라인으로 접속한 상태에서 I/O 테이블을 작성합니다.	CX-Programmer를 오프라인으로 한 상태에서 CX-Programmer로 필요한 I/O 테이블을 작성하여 CPU 유닛에 전송합니다.
	

참고

전원을 ON했을 때 메모리 카드에 저장된 I/O 테이블을 CPU 유닛으로 자동 전송하도록 PLC를 설정할 수 있습니다.



● 유닛 대조

I/O 테이블을 작성하면 기동 시 등록 I/O 테이블과 실제 I/O가 대조됩니다. 일치하지 않으면 A401.10이 ON되어 I/O 설정 에러를 나타내며 운전할 수 없습니다.

AI 번역본
-For Reference Only-

유닛을 접속한 상태에서 I/O 테이블 작성

● 할당 방법

I/O는 자동 할당과 같은 방식으로 할당됩니다. 할당 방법에 대해서는 8-1-2 자동 할당을 참조하십시오.

● I/O 할당 순서

CX-Programmer를 온라인으로 한 상태에서 다음 순서로 접속된 유닛에 대한 I/O 테이블을 작성합니다.

1 메인 윈도우의 프로젝트 트리에서 IO Table을 더블 클릭합니다. I/O Table 윈도우가 표시됩니다.

2 Options를 선택한 다음 Create를 선택합니다. 랙에 장착된 유닛의 형식과 위치가 등록 I/O 테이블로 CPU 유닛에 쓰입니다.

● I/O 테이블 대조 순서

CX-Programmer가 온라인인 상태에서 다음 순서로 I/O 테이블과 실제로 접속된 유닛을 대조합니다.

1 메인 윈도우의 프로젝트 트리에서 IO Table을 더블 클릭합니다. I/O Table 윈도우가 표시됩니다.

2 Options를 선택한 다음 Verify를 선택합니다. 랙에 장착된 유닛의 형식과 위치가 등록 I/O 테이블과 대조됩니다. 결과가 표시되면 확인하십시오.

사용자 지정 할당으로 I/O 테이블 작성

● 할당 방법

접속된 유닛에 I/O를 자동으로 할당하는 대신, 다음 방법으로 원하는 I/O 테이블을 작성하여 I/O를 수동으로 할당할 수 있습니다.

- 랙의 임의 위치에 있는 유닛에 워드 할당
- 워드 예약
- 랙의 선두 워드 지정
- 실제 유닛 없이 I/O 할당

워드는 다음과 같이 할당됩니다:

유닛 분류	할당 방법
기본 I/O 유닛	설정에 일부 제한이 있지만 다음이 가능합니다: 임의의 랙에 있는 임의의 유닛에 할당되는 워드를 지정할 수 있습니다. 이어지는 유닛에는 접속된 순서대로 연속된 워드가 할당됩니다. 이렇게 하여 사용자 설정 할당 그룹 63개를 작성할 수 있습니다.
특수 I/O 유닛	전면 로터리 스위치 설정(즉, 유닛 번호)에 따라 워드가 할당됩니다.
CPU 버스 유닛	
펄스 I/O 모듈	CPU 유닛에 가장 가까운 펄스 I/O 모듈이 펄스 I/O 모듈 0이고, 다른 하나가 펄스 I/O 모듈 1입니다.

AI 번역본

● I/O 할당 순서

CX-Programmer로 I/O 테이블을 작성한 다음 PLC에 전송합니다. 오프라인으로 I/O 테이블을 작성한 후 온라인으로 PLC에 전송하십시오.

1 메인 윈도우의 프로젝트 트리에서 IO Table을 더블 클릭합니다. I/O Table 윈도우가 표시됩니다.

2 편집할 랙을 더블 클릭합니다. 선택한 랙의 슬롯이 표시됩니다.

3 유닛을 할당할 슬롯을 오른쪽 클릭하고 풀다운 메뉴에서 유닛을 선택합니다.

4 I/O 테이블을 편집한 후 Options - Transfer to PLC를 선택하여 CPU 유닛에 **전송합니다.**

● 랙 위치에 선두 워드 할당 및 미사용 워드 예약

CX-Programmer를 사용하여 임의의 랙의 임의의 위치에 있는 유닛에 선두 워드를 설정할 수 있습니다. 그 뒤로는 지정한 선두 워드 다음의 연속된 워드가 유닛이 접속된 순서대로 다음 유닛에 할당됩니다. 각 그룹의 첫 번째 유닛에 선두 워드를 설정하여 최대 63개의 그룹을 만들 수 있습니다.

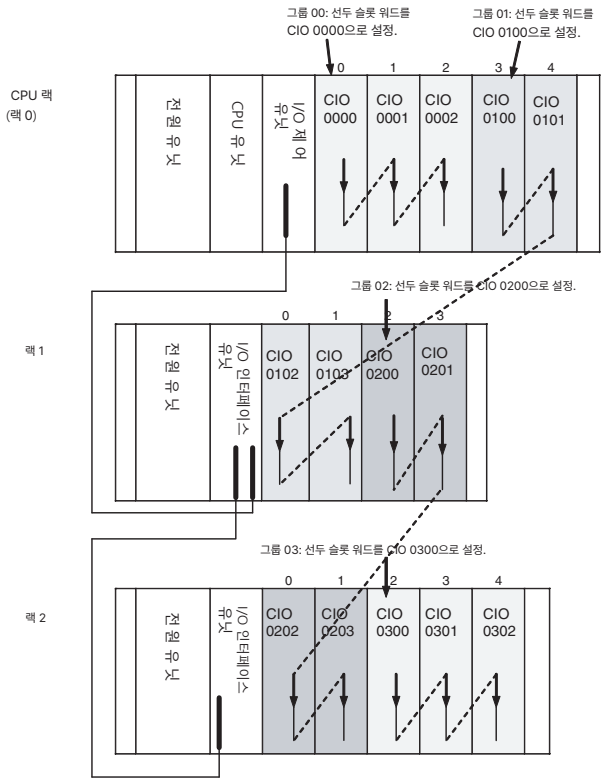
할당 방법

랙 위치(「슬롯」이라고 합니다)에 선두 워드를 설정할 때는 CPU 랙의 슬롯 00에 선두 워드를 반드시 설정해야 합니다. 그다음 다른 슬롯 최대 63개에 대해 임의의 랙의 임의의 슬롯에 선두 워드를 설정할 수 있습니다.

- 슬롯에 설정한 각 선두 워드는 그 슬롯부터 시작하는 그룹을 만듭니다.
- 워드는 지정한 워드부터 그룹의 첫 번째 유닛에 할당되고, 다음 그룹까지(즉, 선두 슬롯 워드가 설정된 다음 유닛까지) 왼쪽에서 오른쪽으로 각 유닛에 연속된 워드가 할당됩니다. 다음 그룹은 같은 랙에서 시작할 수도 있고 그 뒤의 랙에서 시작할 수도 있습니다.
- 그룹 00에서는 CPU 랙의 슬롯 00에 선두 워드를 설정합니다. 그룹 01~63에서는 임의의 랙의 임의의 슬롯에 선두 워드를 설정할 수 있습니다.

AI 번역본
-For Reference Only-

예를 들어 각 랙의 중간에 선두 슬롯 워드를 설정한 경우입니다. 16점 유닛만 사용했습니다.



선두 슬롯 워드 설정

그룹	랙	슬롯	워드
00 *	CPU 랙	00	CIO 0000
01	CPU 랙	03	CIO 0100
02	랙 1	02	CIO 0200
03	랙 2	02	CIO 0300

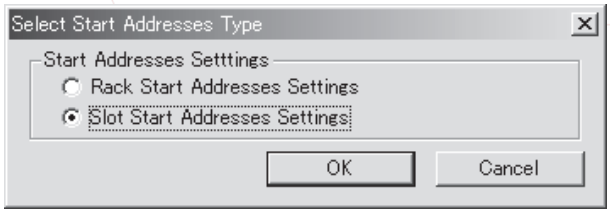
* 그룹 00은 반드시 CPU 랙의 슬롯 00에서 시작해야 합니다. 워드는 임의로 설정할 수 있습니다. 그룹 01~63은 임의의 랙의 임의의 슬롯에 설정할 수 있습니다.

CX-Programmer에서 선두 슬롯 워드 설정

다음 순서로 랙의 선두 워드를 설정합니다.

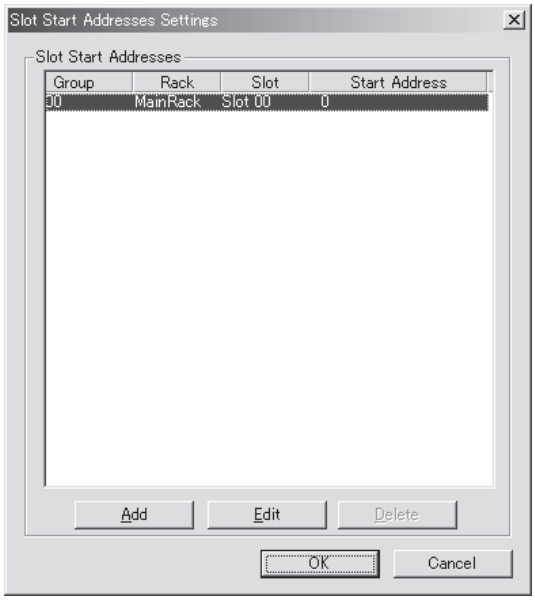
1 I/O Table 윈도우의 Option 메뉴에서 Rack/Slot Start Addresses를 선택합니다. 다음 대화 상자가 표시됩니다.

2 Slot Start Addresses Settings 옵션을 선택하고 OK 버튼을 클릭합니다.

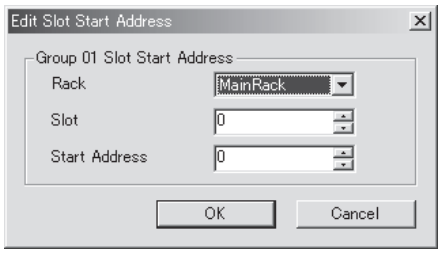


AI 번역본
For Reference Only

3 표시되는 대화 상자에서 CPU 랙의 슬롯 00에 대한 선두 워드를 설정합니다.



4 설정을 CIO 0000에서 변경하려면 Edit 버튼을 클릭합니다. 다음 대화 상자가 표시됩니다.



5 원하는 워드를 설정하고 OK 버튼을 클릭합니다.

6 다른 그룹의 슬롯 선두 워드를 설정하려면 Add 버튼을 클릭하고 랙, 슬롯, 워드를 적절히 설정합니다.

설정	설정 범위	디폴트	비고
그룹	00~63	00	그룹 번호는 그룹이 표시되고 설정되는 순서대로 자동으로 할당됩니다.
랙	CPU 랙(「Main Rack」) 랙 1~7	CPU 랙	그룹 00은 항상 CPU 랙의 슬롯 00에서 시작합니다.
슬롯	00~03	0	
선두 워드	0~999	0	---

AI 번역본
-For Reference Only-

랙에 선두 워드 할당

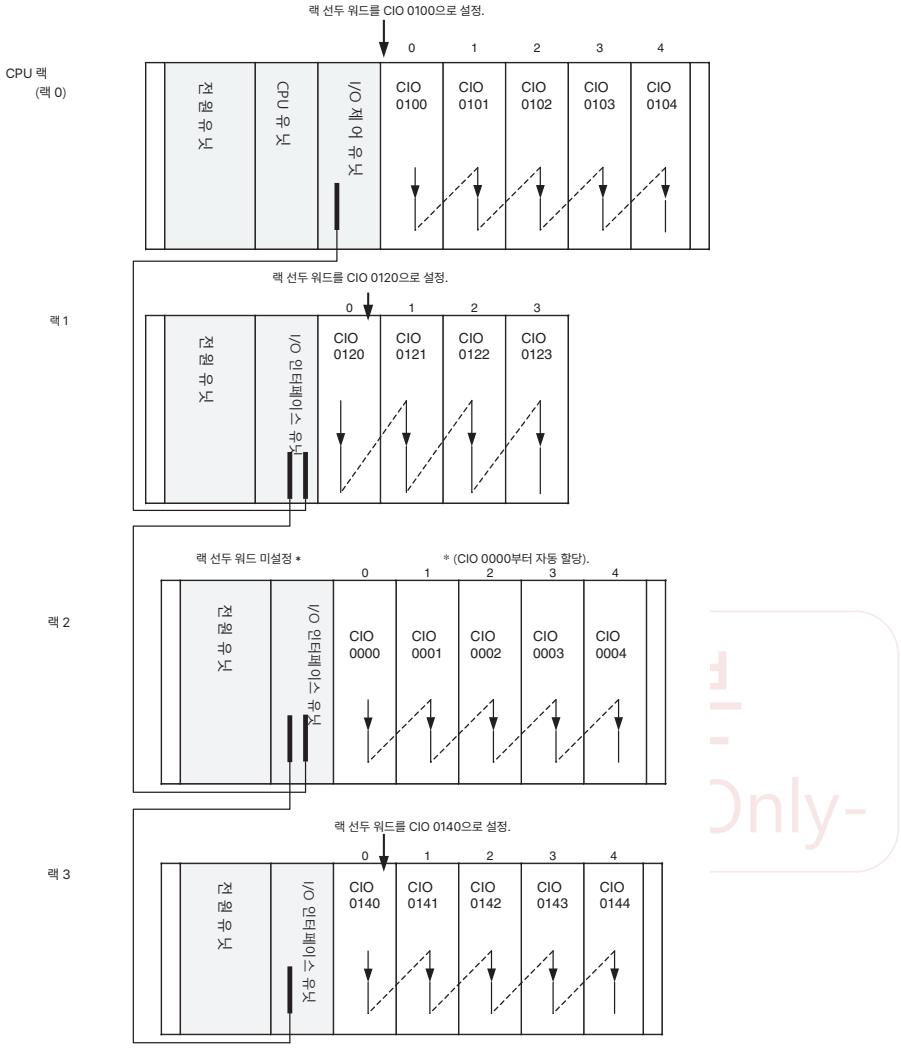
CJ 시리즈 PLC에서는 CX-Programmer의 I/O 테이블 편집 조작으로 각 랙에 할당되는 선두 워드를 설정할 수 있습니다. 예를 들어 CPU 랙은 CIO 0000부터, 다음 랙은 CIO 0100부터, 그다음 랙은 CIO 0200부터 워드가 할당되도록 설정할 수 있습니다.
이렇게 하면 CPU 랙부터 전부 계산하지 않아도 유닛에 대한 워드 할당을 쉽게 확인할 수 있습니다.

● 워드 할당

선두 워드 어드레스가 설정된 랙에서는 지정한 선두 워드부터 시작하여 유닛이 장착된 순서대로(왼쪽에서 오른쪽으로) 유닛에 워드가 할당됩니다. 빈 슬롯에는 워드가 할당되지 않습니다.

선두 워드 어드레스가 설정되지 않은 랙에서는 이전 랙에 할당된 마지막 워드에 이어서 랙 번호 순서로(작은 번호부터 큰 번호로) 워드가 할당되며, 선두 워드가 설정되지 않은 첫 번째 랙에서는 CIO 0000부터 시작합니다.

예: 랙의 선두 워드 설정



랙 선두 워드 설정

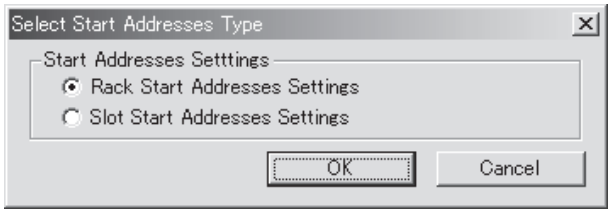
랙	선두 워드
CPU 랙	CIO 0100
랙 1	CIO 0120
랙 2	0000
랙 3	0140

주 랙 번호(0~3)는 랙이 케이블로 물리적으로 접속되는 순서에 따라 고정됩니다.
CPU 랙은 항상 랙 0이고 나머지 랙은 순서대로 랙 1~3이 됩니다. 이 번호는 변경할 수 없습니다.

CX-Programmer에서 랙 선두 워드 설정

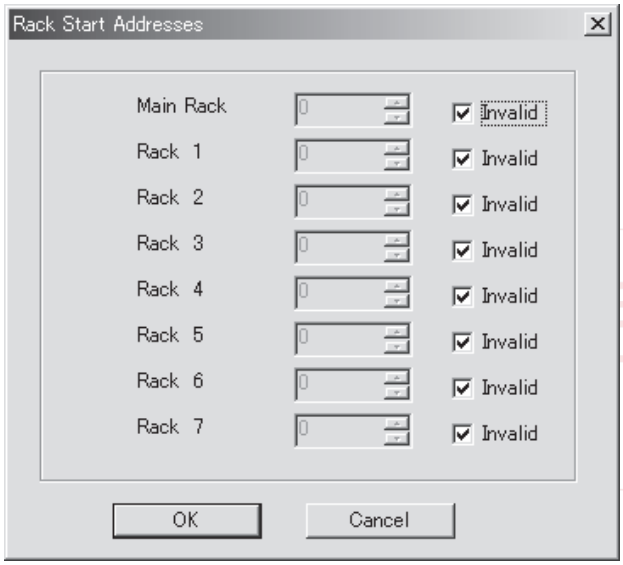
랙 선두 워드를 설정하려면 다음 순서를 사용하십시오.

1 I/O Table 윈도우의 Option 메뉴에서 Rack/Slot Start Addresses를 선택합니다. 다음 대화 상자가 표시됩니다.



2 Rack Start Addresses Settings 옵션을 선택하고 OK 버튼을 클릭합니다.

3 표시되는 대화 상자에서 랙 선두 워드 설정을 무효화하는 설정의 체크 표시를 지우고, CPU 랙과 증설 랙(1~7)의 선두 워드 어드레스를 설정합니다.



설정	설정 범위	디폴트	비고
Rack Start Address	0~900	0	모든 랙에서 동일
Invalid	선택 또는 해제	선택됨(무효)	

주1 어느 CPU 유닛 기종에서도 최대 3개의 랙을 설정할 수 있습니다.
2 CJ2 CPU 유닛과 CJ3 CPU 유닛에서는 각각 최대 3개의 랙과 7개의 랙을 설정할 수 있습니다

사용상의 주의

- 할당되는 워드가 중복되지 않도록 반드시 선두 워드를 설정하십시오. 랙의 선두 워드 설정은 CIO 0000~CIO 0900 범위의 어드레스로 할 수 있습니다. 같은 워드가 두 랙에 할당되거나 두 랙에 설정된 선두 워드가 CIO 0901 이상이면 I/O 테이블을 작성할 수 없으며 증설 랙 번호 중복 플래그(A409.00~A409.07: 랙 번호 0~7)가 ON됩니다. 중복 에러 플래그(치명 에러)(A401.13)도 ON됩니다.

- 랙의 선두 워드 할당을 설정한 후에는 반드시 I/O 테이블을 등록하십시오. I/O 테이블을 등록하려면 CX-Programmer의 I/O Table 윈도우에서 Options를 선택한 다음 Create를 선택합니다. I/O 테이블 등록 조작은 랙에 할당된 I/O 워드를 등록합니다.
- I/O 테이블을 등록한 후 실제 시스템 구성이 변경되어 워드 수나 유닛 기종이 I/O 테이블의 내용과 일치하지 않으면 I/O 대조 에러(A402.09) 또는 I/O 설정 에러(A401.10)가 발생합니다. CPU 버스 유닛 설정 에러(A402.03)나 특수 I/O 유닛 설정 에러(A402.02)가 발생할 수도 있습니다. 유닛을 분리한 경우에는 I/O 테이블 변경 조작으로 없는 유닛의 워드를 예약할 수 있습니다. 유닛을 교체하거나 추가한 경우에는 그 유닛에 할당된 워드 이후의 프로그램 내 모든 워드가 변경되므로 I/O 테이블 등록 조작을 다시 수행해야 합니다.

- CX-Programmer로 I/O 테이블을 삭제하면 랙의 선두 워드 설정도 함께 삭제됩니다.

AI 번역본
-For Reference Only-

8-1-4 I/O 테이블 에러와 주의사항

I/O 테이블 에러

● I/O 설정 에러와 번호 중복 에러

I/O 테이블을 작성 · 편집 · 전송하는 순서(즉, 사용자 설정 I/O 할당 사용)를 수행했을 때 다음의 경우에는 I/O 테이블을 작성할 수 없습니다.

현상	에러	설명	대책
접속된 유닛과 I/O 테이블 사이의 대조 에러	I/O SET ERR (치명 에러)	대조 에러가 있는 경우, 즉 등록된 I/O 테이블이 기본 시스템(즉, CPU 랙과 증설 랙)에 실제로 접속된 유닛의 기종 및 위치와 일치하지 않는 경우 I/O 설정 에러 플래그(치명 에러)(A401.10)가 ON됩니다. 운전할 수 없게 됩니다. CPU 유닛 전면의 RUN 표시등이 OFF되고 ERR/ALARM 표시등이 적색으로 점멸합니다.	· 유닛 개수가 올바르지 않으면 전원을 OFF하고 올바른 유닛을 제대로 접속하십시오. · 유닛 개수가 올바르면 CX-Programmer로 I/O 테이블을 비교하여 유닛의 불일치를 확인한 뒤 전원을 OFF하고 유닛 접속을 수정하십시오. · I/O 테이블에 잘못이 있으면 CX-Programmer로 I/O 테이블을 다시 작성하거나 편집하여 수정하십시오.
특수 I/O 유닛 또는 CPU 버스 유닛에 같은 유닛 번호가 두 번 설정되어 있습니다.	중복 에러 (치명 에러)	다음 중 어느 하나에 해당하면 중복 에러 플래그(A401.13)가 ON됩니다. · 둘 이상의 CPU 버스 유닛에 같은 유닛 번호가 설정되어 있다. · 둘 이상의 특수 I/O 유닛에 같은 유닛 번호가 설정되어 있다 · 둘 이상의 기본 I/O 유닛에 같은 워드가 할당되어 있다. · 둘 이상의 증설 랙에 같은 랙 번호가 설정되어 있다. 운전할 수 없게 됩니다. CPU 유닛 전면의 RUN 표시등이 OFF되고 ERR/ALARM 표시등이 적색으로 점멸합니다.	CPU 버스 유닛 또는 특수 I/O 유닛의 유닛 번호를 확인하여 중복을 해소한 뒤 랙의 전원을 OFF했다가 다시 ON하십시오. A409.00~A409.07에서 비트가 ON인 랙 번호의 유닛에 대한 할당을 확인하십시오. 다른 랙의 유닛을 포함하여 같은 워드가 두 번 이상 할당되지 않도록 할당을 수정한 뒤 랙의 전원을 OFF했다가 다시 ON하십시오. A409.00~A409.07에 표시된 랙의 선두 워드 설정을 확인하고 CX-Programmer로 CIO 0900 이하의 유효한 워드 어드레스로 변경하십시오.

또한 I/O 점수 초과 플래그(치명 에러)(A401.11)가 ON이거나 I/O 버스 에러 플래그(치명 에러)(A401.14)일 때는 I/O 테이블을 사용할 수 없습니다.

AI 번역본
-For Reference Only-

● I/O 테이블 작성 에러 상세 정보

A261(I/O 테이블 작성 에러 상세)의 내용은 CX-Programmer에서 I/O 테이블을 작성할 때 에러가 발생한 경우 그 에러를 일으킨 유닛에 관한 정보를 제공합니다. 이 정보를 사용하면 I/O 테이블 문제 해결 시 문제를 일으킨 유닛을 더 쉽게 찾을 수 있습니다.

명칭		어드레스		내용
		워드	비트	
I/O 테이블 작성 에러 상세 정보	CPU 버스 유닛 셋업 영역 초기화 에러 플래그	A261	00	ON: CPU 버스 유닛 셋업에 에러가 있습니다 I/O 테이블이 정상적으로 생성되면 OFF됩니다.
	I/O 오버플로 플래그		02	ON: 최대 I/O 점수를 초과했습니다. I/O 테이블이 정상적으로 생성되면 OFF됩니다.
	중복 에러 플래그		03	ON: 같은 유닛 번호가 두 번 이상 사용되었습니다. I/O 테이블이 정상적으로 생성되면 OFF됩니다.
	I/O 버스 에러 플래그		04	ON: I/O 버스 에러 I/O 테이블이 정상적으로 생성되면 OFF됩니다.
	특수 I/O 유닛 에러 플래그		07	ON: 특수 I/O 유닛에 에러가 있습니다 I/O 테이블이 정상적으로 생성되면 OFF됩니다.
	I/O 미확정 에러 플래그		09	ON: I/O 검출이 완료되지 않았습니다. I/O 테이블이 정상적으로 생성되면 OFF됩니다.

I/O 테이블 설정 시의 주의사항

I/O 테이블을 설정한 후 유닛을 교체할 예정이라면 워드 할당에 중복이 없는지 확인하십시오. 다만 I/O 테이블을 등록한 뒤에, 예를 들어 1워드 유닛을 2워드 유닛으로 교체한 결과로 워드 할당에 중복이 생길 수도 있습니다. 이 경우 새 유닛에 필요한 추가 워드가 다음 유닛에도 여전히 할당된 상태가 됩니다.

유닛을 교체한 후 PLC의 전원을 ON하면 CPU 유닛은 등록된 I/O 테이블을 PLC에 실제로 접속된 유닛과 대조합니다. 중복이 있으면 에러가 발생하여 I/O 테이블을 더 이상 편집할 수 없게 됩니다. 이 경우에는 I/O 테이블을 편집하여 다시 전송해야 합니다.

I/O 테이블을 편집하면 CX-Programmer가 선두 워드 설정의 중복과 설정 범위 에러를 확인하여 문제를 해소하는 데 도움을 줍니다.

AI 번역본
-For Reference Only-

메모리 카드 사용 시의 주의사항

CPU 유닛의 I/O 테이블을 작성하는 데 사용한 I/O 할당 방법(기동 시 I/O 자동 할당 또는 사용자 설정 I/O 할당)은 기동 시 자동 전송용 파라미터 파일(AUTOEXEC.STD)에 기록됩니다. 메모리 카드에서 기동 시 자동 전송이 실행되면 기록된 방법이 자동으로 검출되어 기동 시 I/O를 자동으로 할당하거나 I/O 테이블을 대조하는 데 사용됩니다.

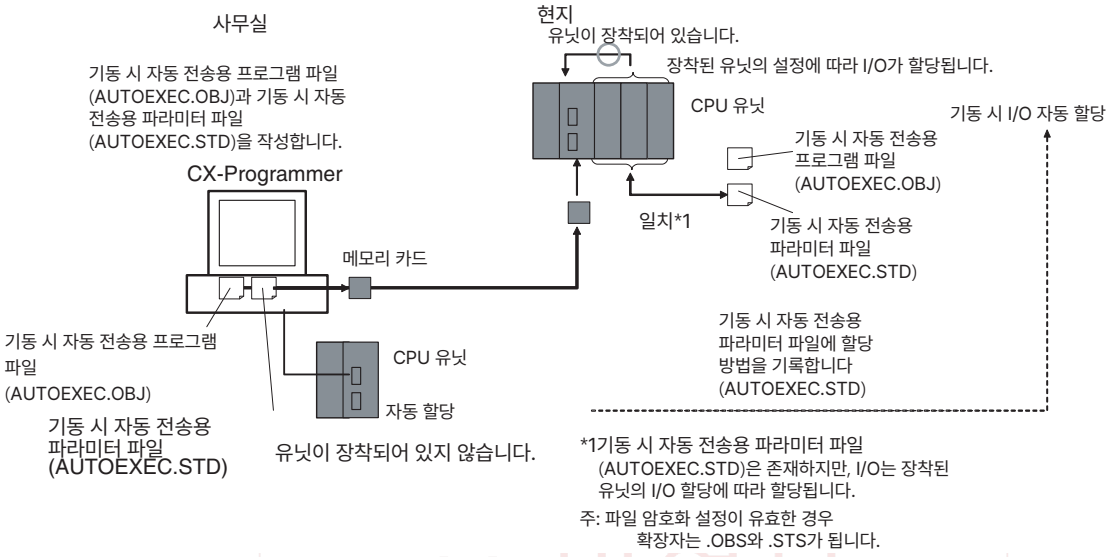
아래에서는 기동 시 자동 전송용 파라미터 파일(AUTOEXEC.STD)을 작성하여 CPU 유닛에 I/O 테이블을 작성하는 두 가지 방법을 설명합니다.

・자동 할당*1
메모리 카드의 기동 시 자동 전송용 파라미터 파일에 있는 I/O 테이블이 무효화되고, PLC에 실제로 접속된 유닛을 기준으로 기동 시 I/O 자동 할당을 사용하여 I/O가 할당됩니다.

*1 기동 시 I/O 자동 할당에서는 I/O 테이블을 미리 작성하지 않으며, 전원을 ON할 때마다 실제로 접속된 기본 I/O 유닛에 I/O가 자동으로 할당됩니다.

・수동 할당
메모리 카드의 기동 시 자동 전송용 파라미터 파일에 있는 I/O 테이블이 유효화되고, CPU 유닛이 PLC에 실제로 접속된 유닛과 I/O 테이블을 대조합니다.

자동 할당의 예를 아래에 나타냅니다.
현지에 기본 I/O 유닛만 접속되어 있는 경우, 접속된 유닛의 상태에 따라 I/O가 할당됩니다.
현지에 특수 I/O 유닛이나 CPU 버스 유닛이 접속되어 있는 경우에는 주의가 필요합니다. 유닛이 접속되지 않은 사무실에서 작성한 기동 시 자동 전송용 파일에는 유닛 설정이 포함되지 않으므로, 현지에서 유닛이 올바르게 동작하지 않을 수 있습니다.



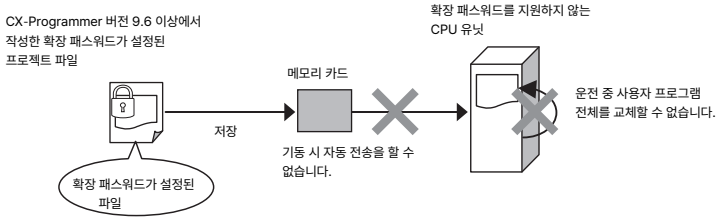
사용상의 주의

확장 패스워드가 설정된 프로그램 파일은 확장 패스워드를 지원하지 않는 CPU 유닛에서는 기동 시 자동 전송을 할 수 없으며, 운전 중 사용자 프로그램 전체를 교체하는 데에도 사용할 수 없습니다.

AI 번역본
-For Reference Only-

● 확장 패스워드를 지원하지 않는 CPU 유닛

・CJ2H CPU 유닛, 유닛 버전 1.4 이전
・CJ2M CPU 유닛, 유닛 버전 2.0 이전



8-2 CPU 버스 유닛과 특수 I/O 유닛 설정

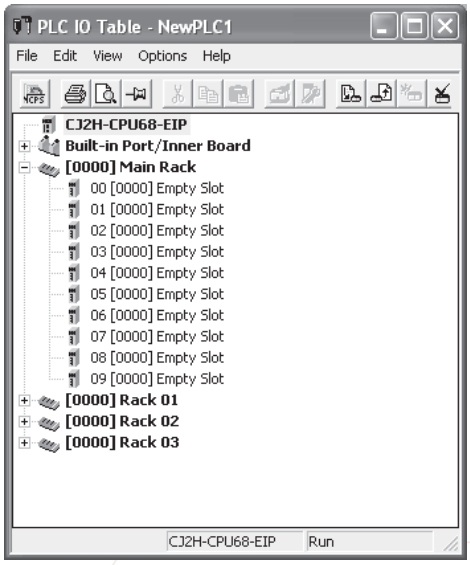
8-2-1 파라미터 설정

CX-Programmer를 사용하여 CJ 시리즈 CPU 버스 유닛과 특수 I/O 유닛에 할당된 DM 영역 및 CIO 영역 워드의 파라미터를 설정하고, 이더넷 유닛의 CPU 버스 유닛 셋업을 설정할 수 있습니다.
CX-Programmer에서는 I/O Table 윈도우를 사용합니다.

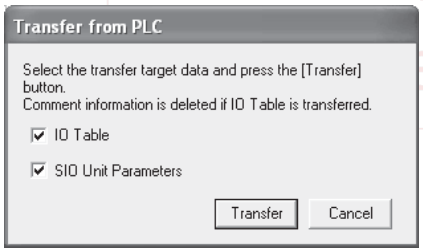
설정은 온라인 또는 오프라인으로 할 수 있습니다. 오프라인으로 설정한 경우에는 온라인으로 전환하여 전송해야 합니다.

이 절에서는 I/O 테이블을 작성하여 CPU 유닛에서 I/O Table 윈도우로 전송한 후 I/O 테이블을 편집하고 CPU 버스 유닛과 특수 I/O 유닛을 설정하는 예를 설명합니다. 이 순서는 CPU 버스 유닛과 특수 I/O 유닛을 오프라인으로 I/O 테이블에 등록한 다음 CPU 버스 유닛과 특수 I/O 유닛의 설정을 편집하는 경우와 동일합니다.

1 CX-Programmer를 온라인으로 접속하고 I/O Table 윈도우를 엽니다.



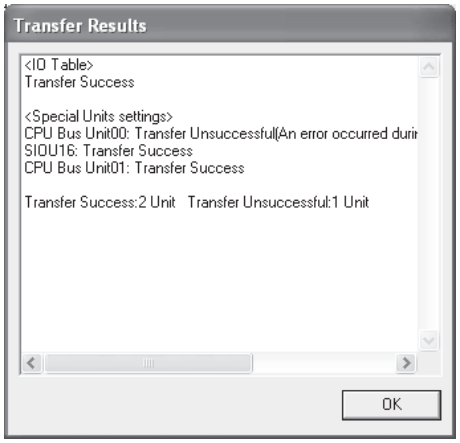
2 I/O Table 윈도우 메뉴에서 Options – Transfer from the PLC를 선택합니다. 다음 그림과 같이 Transfer from the PLC 대화 상자가 표시됩니다.



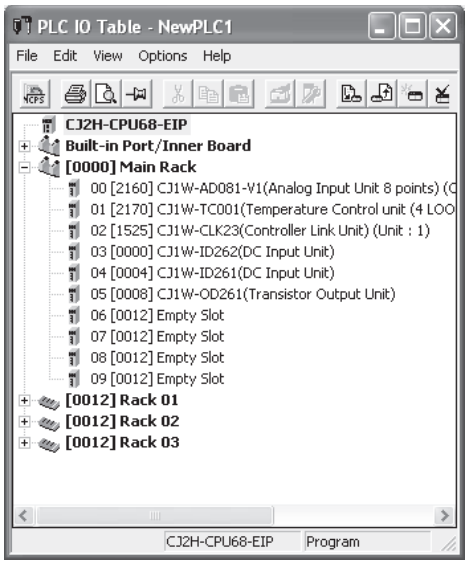
이 시점에서 I/O Table 및 SIO Unit Parameters 옵션을 선택할 수 있습니다.

사 - 번역본
-erence Only-

3 전송하려는 데이터의 옵션을 선택하고(박스를 체크) Transfer 버튼을 클릭합니다. 이 예에서는 두 옵션을 모두 선택했습니다. PLC에서 I/O 테이블과 유닛 파라미터 데이터가 전송되고 전송 결과가 표시됩니다.



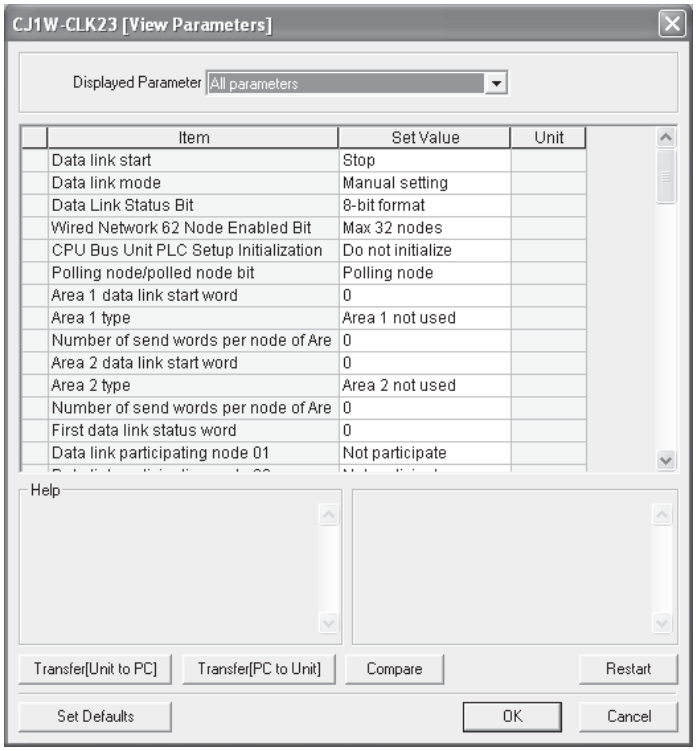
4 전송 결과를 확인하고 OK 버튼을 클릭합니다. PLC에서 전송된 I/O 테이블이 표시됩니다.



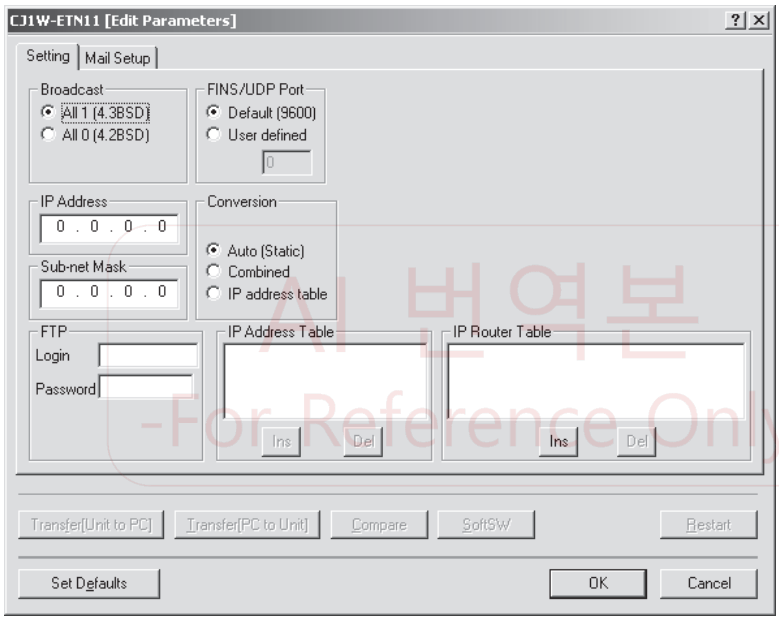
AI 번역본
-For Reference Only-

5 특수 I/O 유닛이나 CPU 버스 유닛의 파라미터를 편집하려면 해당 유닛을 선택한 뒤 더블 클릭하거나 메뉴에서 Edit – SIO Unit Parameters를 선택합니다. 선택한 유닛의 Edit Parameters 대화 상자가 표시됩니다.

- 이 예에서는 Controller Link 유닛의 CPU 버스 유닛 할당 DM 설정을 편집하기 위해 그 유닛의 Edit Parameters 대화 상자를 표시했습니다.



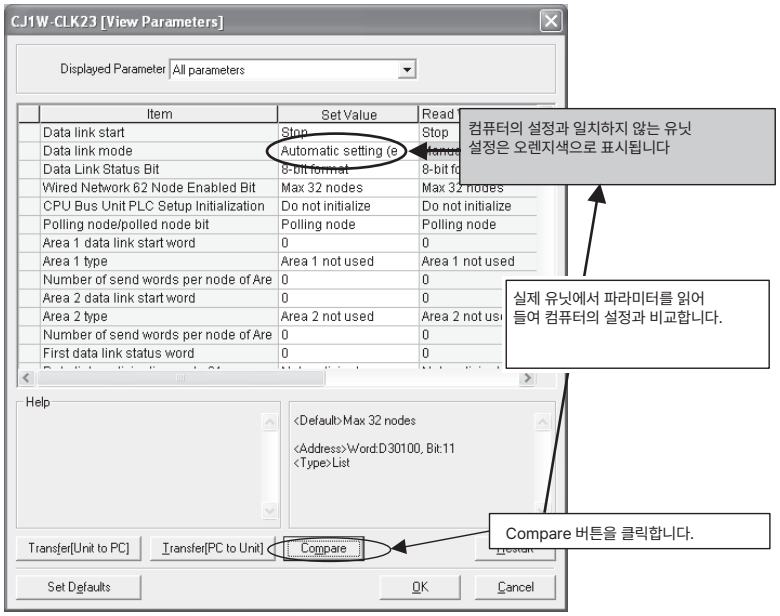
- 이 예에서는 DM 영역에 할당된 워드의 CPU 버스 유닛 설정을 편집하기 위해 이더넷 유닛의 Edit Parameters 대화 상자를 표시했습니다.



AI 번역본
-For Reference Only-

6 파라미터를 편집하고 필요하면 Compare 버튼을 클릭합니다. PLC가 온라인 상태에서 Compare 버튼을 클릭하면, I/O 테이블 설정이 CPU 유닛의 유닛 설정(예: DM 영역과 CIO 영역에 할당된 워드)과 일치하는지 소프트웨어가 즉시 확인합니다. 내용이 일치하지 않으면
실제 유닛에서 읽어 들인 설정이 오렌지색으로 표시됩니다.

다음 예는 Controller Link 유닛의 설정을 비교한 결과를 나타냅니다.



7 각 유닛에 설정한 파라미터를 CPU 유닛으로 다운로드하려면 Download to Unit 버튼을 클릭합니다.

사용상의 주의

I/O Table 윈도우에서 특수 I/O 유닛이나 CPU 버스 유닛의 설정을 편집한 경우, 그 유닛의 DM 영역 또는 CPU 버스 유닛 셋업 영역에 할당된 파라미터는 Edit Parameters 대화 상자에서 Transfer PC to Unit 버튼을 클릭했을 때에만 실제 PLC로 전송됩니다.
전송된 파라미터는 자동으로 유효화되지 않으며, 유효화하지 않으면 사용되지 않습니다.
전송된 파라미터 설정을 유효화하려면 Reset 버튼을 클릭하십시오.

참고

파라미터 설정 데이터 파일의 저장

- Edit Parameters 대화 상자에서 설정한 데이터는 확장자가 xml인 파라미터 설정 파일로 저장할 수 있습니다. 파라미터 설정을 저장하려면 I/O 테이블에서 원하는 특수 I/O 유닛이나 CPU 버스 유닛을 오른쪽 클릭하고 팝업 메뉴에서 Save Parameters를 선택하십시오.
- 저장한 파라미터 설정은 파라미터 설정 파일(확장자 xml)에서 읽어 들일 수도 있지만, 같은 기종의 특수 I/O 유닛 또는 CPU 버스 유닛에 대한 설정이어야 합니다.

신 번역본
Version Only

8-2-2 데이터 교환

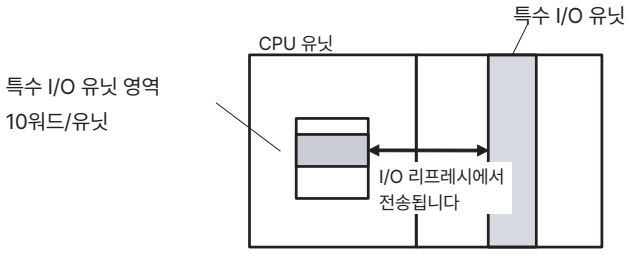
이 절에서는 특수 I/O 유닛 또는 CPU 버스 유닛과 CPU 유닛 사이에서 데이터를 교환하는 방법을 설명합니다.

특수 I/O 유닛

특수 I/O 유닛 영역(I/O 리프레시)

특수 I/O 유닛 영역의 I/O 리프레시 중에 매 사이클 데이터가 교환됩니다. 기본적으로 유닛 번호 설정에 따라 각 특수 I/O 유닛에 10워드가 할당됩니다. 자세한 내용은 각 특수 I/O 유닛의 오퍼레이션 매뉴얼을 참조하십시오.

특수 I/O 유닛 영역의 범위는 CIO 2000~CIO 2959입니다(10워드 × 96유닛).



DM 영역에 할당된 워드의 전송

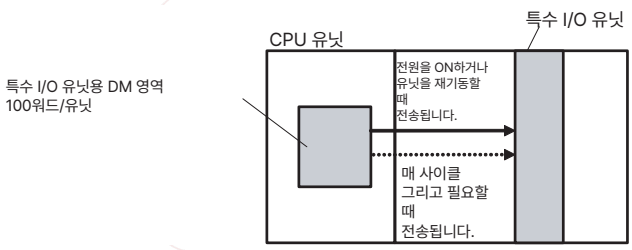
각 유닛에 할당된 워드를 통해 데이터가 전송되는 시점은 세 가지입니다. 데이터 전송 타이밍은 사용하는 기종에 따라 달라집니다.

- PLC의 전원을 ON했을 때 전송되는 데이터.
- 유닛을 재기동했을 때 전송되는 데이터.
- 필요할 때 전송되는 데이터.

일부 기종은 DM 영역에서 유닛으로, 그리고 유닛에서 DM 영역으로 양방향으로 데이터를 전송합니다. 데이터 전송에 대한 자세한 내용은 해당 유닛의 오퍼레이션 매뉴얼을 참조하십시오.

DM 영역의 특수 I/O 유닛 워드: D20000~D29599(100워드 × 96유닛)

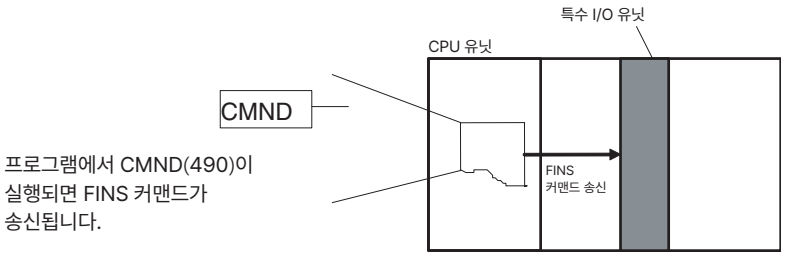
특수 I/O 유닛의 초기 설정을 위해 DM 영역의 이 워드로 데이터가 전송됩니다. 시스템을 변경하기 위해 프로그램에서 이 할당 워드의 내용을 변경한 경우에는 유닛을 재기동하도록 특수 I/O 유닛 재기동 비트(A502.00~A507.15)를 ON해야 합니다.



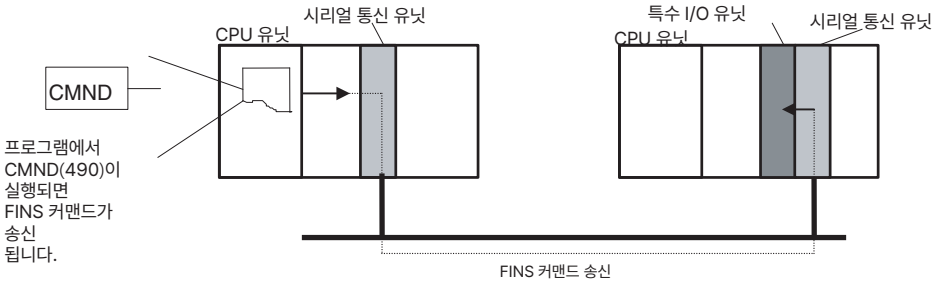
본
- Only-

● FINS 커맨드

래더 프로그램에 CMND(490) 명령어를 추가하여 특수 I/O 유닛으로 FINS 커맨드를 보낼 수 있습니다.



주 FINS 커맨드는 자체 PLC뿐 아니라 네트워크상의 다른 PLC에 있는 특수 I/O 유닛으로도 송신할 수 있습니다.



● 특수 I/O 유닛의 초기화

특수 I/O 유닛은 PLC의 전원이 ON되거나 유닛의 재기동 비트(A502.00~A507.15)가 ON될 때 초기화됩니다. 유닛이 초기화되는 동안에는 그 유닛의 특수 I/O 유닛 초기화 플래그(A330.00~ A335.15)가 ON됩니다.

초기화 플래그가 ON인 동안에는 그 특수 I/O 유닛에 대해 I/O 리프레시(사이클릭 I/O 리프레시 또는 IORF(097) · FIORF(225)에 의한 리프레시가 수행되지 않습니다.

● 특수 I/O 유닛 사이클릭 리프레시 무효화

각 유닛 전면에 설정된 유닛 번호에 따라 특수 I/O 유닛 영역(CIO 2000~CIO 2959)에서 각 특수 I/O 유닛에 10워드가 할당됩니다. 특수 I/O 유닛 영역의 데이터는 I/O 리프레시 중(END(001) 명령어 실행 직후)에 CPU 유닛에서 매 사이클 리프레시됩니다.

특수 I/O 유닛을 너무 많이 장착하면 I/O 리프레시에 시간이 지나치게 걸릴 수 있습니다. I/O 리프레시에 시간이 너무 많이 걸리는 경우에는 특정 특수 I/O 유닛의 사이클릭 리프레시를 무효화하도록 PLC 설정을 할 수 있습니다. (특수 I/O 유닛 사이클릭 리프레시 무효화 비트는 PLC 설정 어드레스 226~231에 있습니다.)

I/O 리프레시 시간이 너무 짧으면 유닛의 내부 처리가 따라가지 못해 특수 I/O 유닛 에러 플래그(A402.06)가 ON되고 특수 I/O 유닛이 올바르게 동작하지 않을 수 있습니다. 이 경우 PLC 설정에서 최소 사이클 타임을 설정하여 사이클 타임을 늘리거나, 그 특수 I/O 유닛과의 사이클릭 I/O 리프레시를 무효화할 수 있습니다.

그러면 사이클릭 리프레시가 무효화된 경우, 프로그램 실행 중에 IORF(097) 또는 FIORF(225)로 특수 I/O 유닛의 데이터를 리프레시할 수 있습니다.

-For Reference Only-

사용상의 주의

IORF(097), FIORF(225), IORD (222), IOWR(223)은 인터럽트 태스크에서 특수 I/O 유닛에 대해 실행할 수 있습니다. 그 경우에는 반드시 PLC 설정에서 그 특수 I/O 유닛의 사이클릭 리프레시를 무효화하십시오. 사이클릭 리프레시를 무효화하지 않은 상태에서 인터럽트 태스크에서 다음 처리 중 하나가 실행되면 비치명 에러가 발생하고 중복 리프레시 에러 플래그(A402.13)가 ON됩니다.

・같은 특수 I/O 유닛에 대해 IORF(097) 또는 FIORF(225)로 I/O 리프레시가 실행된 경우.
・같은 특수 I/O 유닛에 대해 IORD (222) 또는 IOWR(223)으로 메모리 영역에서 데이터를 읽거나 메모리 영역에 데이터를 쓴 경우.

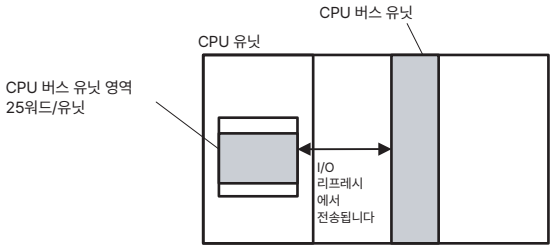
특수 I/O 유닛의 사이클릭 리프레시를 무효화하는 경우에는, 운전 중 프로그램에서 최소 11초마다 IORF(097) 또는 FIORF(225)로 그 유닛의 I/O가 리프레시되도록 반드시 하십시오. 11초마다 리프레시되지 않으면 그 특수 I/O 유닛에서 CPU 유닛 서비스 감시 에러가 발생합니다.

CPU 버스 유닛

● CPU 버스 유닛 영역(I/O 리프레시)

CPU 버스 유닛 영역의 I/O 리프레시 중에 매 사이클 데이터가 교환됩니다. 기본적으로 유닛 번호 설정에 따라 각 CPU 버스 유닛에 25워드가 할당됩니다. CPU 버스 유닛이 실제로 사용하는 워드 수는 기종에 따라 다릅니다.

특수 I/O 유닛 영역의 범위는 CIO 1500~CIO 1899입니다(25워드 × 16유닛).



주 래더 프로그램에서 CPU 버스 I/O 리프레시 명령어(DLNK(226))를 실행하여 지정한 유닛 번호의 CPU 버스 유닛에 할당된 CIO 영역 워드를 리프레시할 수 있습니다.

● DM 영역에 할당된 워드의 전송

유닛 번호에 따라 각 유닛에 100워드가 할당됩니다.

주 이 워드는 모든 기종의 CPU 버스 유닛에서 사용되는 것은 아닙니다.

CPU 버스 유닛용 DM 영역 워드: D30000~D31599(100워드 × 16유닛)

이 워드의 내용은 다음 시점에 전송됩니다. 실제로 어떤 전송이 이루어지는지는 유닛의 기종에 따라 달라집니다.

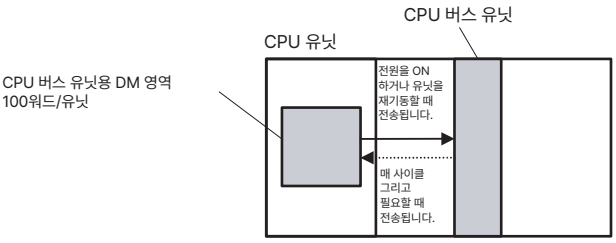
・PLC의 전원을 ON했을 때 전송되는 데이터.
・매 사이클 전송되는 데이터.
・필요할 때 전송되는 데이터.

주 래더 프로그램에서 CPU 버스 I/O 리프레시 명령어(DLNK(226))를 실행하여 지정한 유닛 번호의 CPU 버스 유닛에 할당된 DM 영역 워드를 리프레시할 수 있습니다.

일부 기종은 DM 영역에서 유닛으로, 그리고 유닛에서 DM 영역으로 양방향으로 데이터를 전송합니다. 데이터 전송에 대한 자세한 내용은 해당 유닛의 오퍼레이션 매뉴얼을 참조하십시오.

AI 번역본
-For Reference Only-

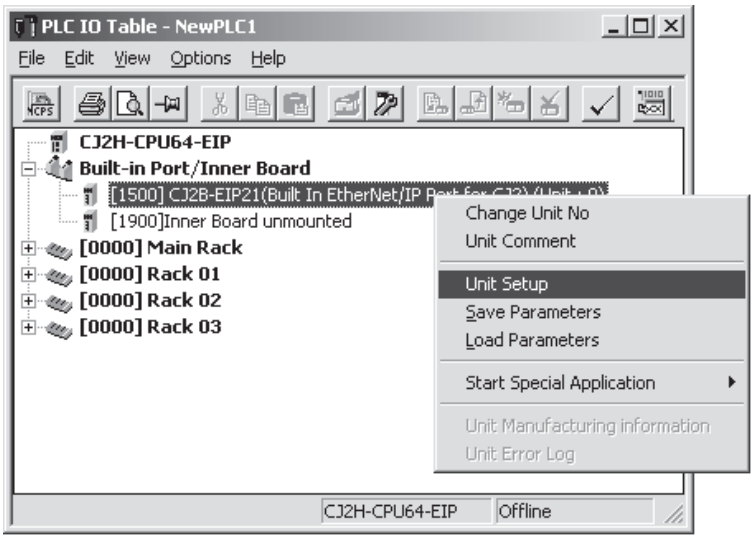
이 100워드는 일반적으로 CPU 버스 유닛의 초기 설정을 유지하는 데 사용됩니다. 시스템 변경을 반영하기 위해 프로그램에서 이 영역의 내용을 변경한 경우에는 해당 유닛을 재기동하도록 그 유닛의 재기동 비트(A501.00~A501.15)를 ON해야 합니다.



참고

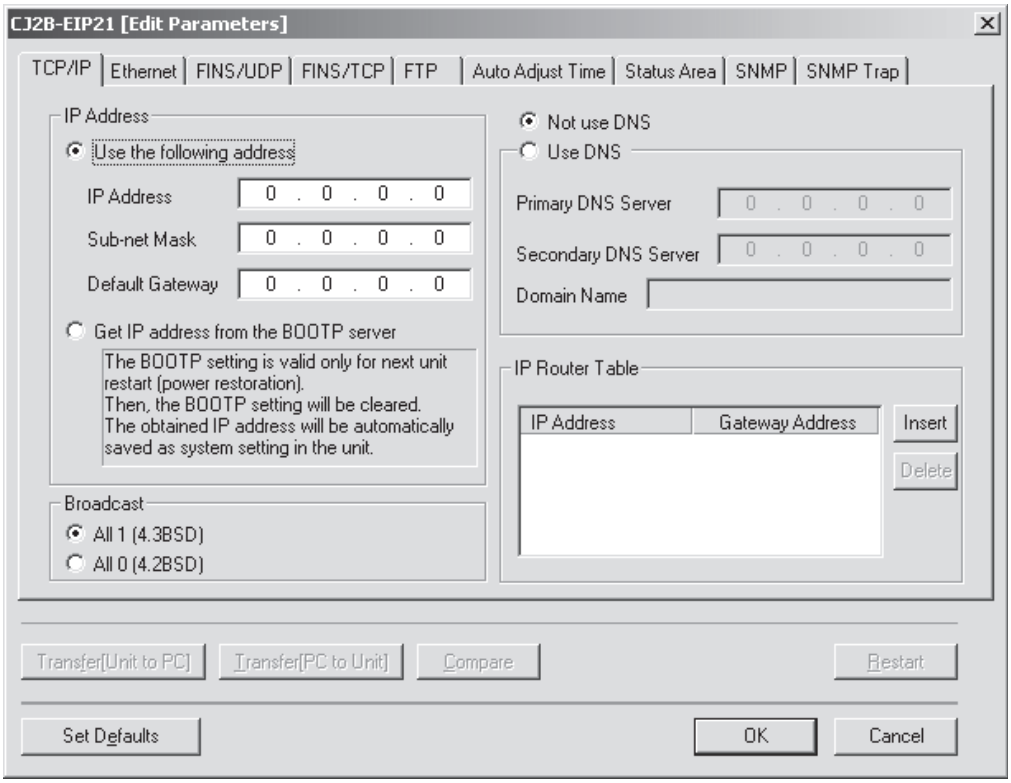
CJ2H-CPU6□-EIP 또는 CJ2M-CPU3□, CJ3M-CPU3□의 내장 EtherNet/IP 포트는 로터리 스위치 설정에 따라 CPU 버스 유닛으로 취급됩니다. 통신 파라미터를 설정하려면 다음 순서를 사용하십시오.

(1) CJ2B-EIP21(CJ2H-CPU6□-EIP용 내장 EtherNet/IP 포트), CJ3M-EIP21S (CJ3M-CPU3□용 내장 EtherNet/IP 포트) 또는 CJ2M-EIP21(CJ2M-CPU3□용 내장 EtherNet/IP 포트)을 PLC의 I/O 테이블에서 Built-in Port/Inner Board 아래에서 선택합니다. 오른쪽 클릭하여 Unit Setup을 선택합니다.



AI 번역본
-For Reference Only-

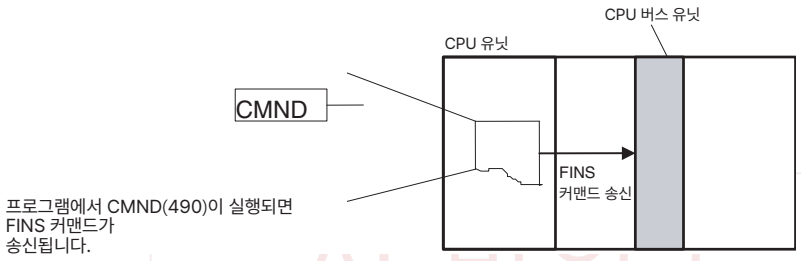
(2) 파라미터를 설정합니다.



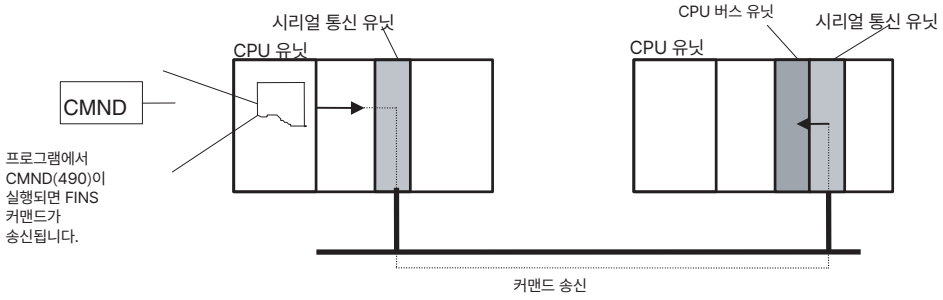
내장 EtherNet/IP 포트의 파라미터에 대해서는 EtherNet/IP 유닛 오퍼레이션 매뉴얼 (Cat. No. W465)을 참조하십시오.

● FINS 커맨드

래더 프로그램에 CMND/CMND2(490) 명령어를 추가하여 CPU 버스 유닛으로 FINS 커맨드를 발행할 수 있습니다.



주 FINS 커맨드는 자체 PLC뿐 아니라 네트워크상의 다른 PLC에 있는 CPU 버스 유닛으로도 송신할 수 있습니다.



● CPU 버스 유닛의 초기화

CPU 버스 유닛은 PLC의 전원이 ON되거나 유닛의 재기동 비트(A501.00~A501.15)가 ON될 때 초기화됩니다. 유닛이 초기화되는 동안에는 그 유닛의 CPU 버스 유닛 초기화 플래그 (A302.00~A302.15)가 ON됩니다. 초기화 플래그가 ON인 동안에는 그 CPU 버스 유닛에 대해 사이클릭 I/O 리프레시가 수행되지 않습니다.

9

PLC 설정

이 섹션에서는 PLC 운전의 옵션을 설정하는 데 사용하는 PLC 설정의 파라미터를 설명합니다.

9-1 PLC 설정 개요 9-2

9-2 PLC 설정 항목 목록 9-4

9-3 PLC 설정 항목 설명 9-5

9-3-1 기동 시 동작 설정 9-5
9-3-2 CPU 유닛 설정 9-8
9-3-3 타이밍/동기 설정 9-13
9-3-4 특수 I/O 유닛 사이클릭 리프레시 9-18
9-3-5 기본 I/O 유닛 랙 응답 시간 9-20
9-3-6 시리얼 포트 설정 9-22
9-3-7 주변 서비스 9-29
9-3-8 FINS 보호 9-30
9-3-9 I/O 모듈 9-32

AI 번역본
-For Reference Only-

9-1 PLC 설정 개요

PLC 설정에는 PLC 운전을 사용자에 맞게 조정하기 위해 변경할 수 있는 CPU 유닛의 기본 소프트웨어 파라미터 설정이 들어 있습니다. 이 설정은 프로그래밍 콘솔이나 그 밖의 프로그래밍 디바이스에서 변경할 수 있습니다. 다음의 경우에 PLC 설정을 변경하십시오.

용도	파라미터
다음의 경우에 CJ 시리즈 기본 I/O 유닛의 입력 응답 시간 설정을 변경해야 합니다: ・기본 I/O 유닛에서 채터링이나 노이즈가 발생한다. ・사이클 타임보다 긴 간격으로 짧은 펄스 입력을 받고 있다.	기본 I/O 유닛 랙 응답 시간
PLC의 전원을 ON했을 때 I/O 메모리의 모든 영역(CIO 영역, 워크 영역, 타이머 플래그 및 현재값, 태스크 플래그, 인덱스 레지스터, 데이터 레지스터 포함)의 데이터를 유지해야 한다.	기동 시 IOM 유지 비트 상태
PLC의 전원을 ON했을 때 프로그래밍 디바이스에서 강제 세트 또는 강제 리셋한 비트의 상태를 유지해야 한다.	기동 시 강제 상태 유지 비트 상태
・PLC를 디버깅한다, ・기동 모드를 PROGRAM 또는 MONITOR 모드로 변경한다.	기동 모드
배터리리스 운전을 사용할 때 배터리 전압 저하 에러 검출이 필요하지 않다.	배터리 전압 저하 검출
중복 리프레시 검출을 무효화한다.	중복 리프레시 에러 검출
RS-232C 포트를 프로그래밍 콘솔이나 CX-Programmer (페리페럴 버스)의 통신 속도 자동 검출과 함께 사용하지 않으며, 9,600 bps 같은 디폴트 호스트 링크 통신 설정도 사용하지 않는다.*1	RS-232C 포트 설정
NT 링크를 통한 PT와의 통신을 고속화하고 싶다.	페리페럴 포트 또는 RS-232C 포트의 통신 포트 보레이트를 "high-speed NT Link"로 설정합니다.
정주기 인터럽트 기능의 주기를 10 ms가 아니라 1 ms (또는 0.1 ms) 단위로 설정하고 싶다.	정주기 인터럽트 시간 단위
I/O 인터럽트 태스크나 정주기 인터럽트 태스크에서 고속 인터럽트 기능을 사용하고 싶다.	고속 인터럽트 기능 유효화
동기 유닛 동작 기능을 사용하고 싶다.	동기 동작 사용
디버깅 시 명령어 에러를 찾는다.	명령어 에러 시 CPU 정지
일정한 I/O 리프레시 주기를 만들기 위해 최소 사이클 타임을 설정하고 싶다.	최소 사이클 타임
1초 이외의 최대 사이클 타임을 설정하고 싶다 (10 ms~40,000 ms).	감시 사이클 타임
정전 시에 특수한 처리를 수행한다.	전원 OFF 인터럽트 태스크*2
정전 검출을 지연시키고 싶다.	전원 OFF 검출 지연 시간*1
・인터럽트 태스크에서 IORF나 FIORF를 실행하고 싶다. ・특수 I/O 유닛을 많이 사용할 때 평균 사이클 타임을 단축하고 싶다. ・특수 I/O 유닛의 I/O 리프레시 주기를 늘리고 싶다.	특수 I/O 유닛 사이클릭 리프레시

AI 번역
For Reference Only

용도	파라미터
FAL(006)과 FPD(269)의 사용자 정의 에러를 에러 로그에 기록하고 싶지 않다.	FAL 에러 로그 등록
문자열 처리로 인한 사이클 타임의 변동을 줄이고 싶다	테이블 데이터 · 문자열 · 데이터 시프트 명령어의 백그라운드 실행
· 전원을 ON했을 때 기동에 시간이 걸리는 유닛이 있다. · CPU 유닛의 운전을 개시하기 위해 유닛의 기동 처리 완료를 기다리고 싶지 않다.	실행 설정

*1 PLC 설정 항목을 변경하려면 CPU 유닛 전면의 딥 스위치 핀 5가 OFF여야 합니다.

*2 CJ1W-PD022가 장착되어 있으면 이 설정을 사용할 수 없습니다.

● 보조 기억 영역의 관련 비트 · 워드

명칭	어드레스	설명	액세스
PLC 설정 에러 플래그 (비치명 에러)	A40210	PLC 설정에 설정 에러가 있을 때 ON됩니다.	읽기 전용

AI 번역본
-For Reference Only-

9-2 PLC 설정 항목 목록

다음 표는 PLC 설정의 디폴트 설정을 나타냅니다. 설정을 변경하려면 CX-Programmer로 PLC 설정을 편집한 다음 PLC 설정을 CPU 유닛으로 전송하십시오.

CX-Programmer PLC 설정 탭 페이지	설정 이름		적용 CPU 유닛		디폴트	페이지
			CJ2H CPU 유닛	CJ3M/ CJ2M CPU 유닛		
기동	기동 시 유지 설정	강제 상태 유지 비트	가능	가능	전원을 ON할 때 유지되지 않음.	9-5
		IOM 유지 비트	가능	가능	전원을 ON할 때 유지되지 않음.	9-6
	동작 모드		가능	가능	RUN 모드	9-7
	실행 설정	유닛/이너 보드 인식을 초기화하기 전에 프로그램 실행 시작	가능	가능	시작하지 않음.	9-7
CPU 유닛 설정	실행 처리 설정	배터리 전압 저하 검출하지 않음	가능	가능	검출함.*1	9-8
		중복 리프레시 에러 검출	가능	가능	검출함.	9-10
		명령어 에러 시 CPU 정지	가능	가능	정지하지 않음.	9-10
		FAL 에러 로그 등록	가능	가능	에러 로그에 등록함.	9-11
	백그라운드 실행 설정		가능	가능	백그라운드에서 실행하지 않음.	9-11
	FB 내 통신 명령어 설정	리트라이 횟수	가능	가능	0회	9-12
		응답 타임아웃	가능	가능	2 s	9-12
타이밍/ 동기 설정	감시 사이클 타임		가능	가능	1,000 ms (1 s)	9-13
	일정 사이클 타임(최소 사이클 타임)		가능	가능	일정하지 않음	9-14
	정주기 인터럽트 주기		가능	가능	10 ms	9-15
	전원 OFF 검출 시간		가능	가능	0 ms	9-16
	전원 OFF 인터럽트		가능	가능	사용하지 않음.	9-17
	고속 인터럽트 기능 유효화		가능	불가	유효화하지 않음.	9-17
	동기 동작 사용		가능	불가	사용하지 않음.	9-17
특수 I/O 유닛 사이클릭 리프레시	SIOU 사이클 리프레시 무효화		가능	가능	무효화되지 않음.	9-18
유닛 설정	기본 I/O 유닛의 입력 응답 시간		가능	가능	8 ms	9-18
시리얼 포트	모드 (모드를 설정하려면 CPU 유닛의 딥 스위치 핀 5가 OFF(디폴트)여야 합니다.)		가능	가능	Host Link	9-22
페리페럴 서비스	실행 모드		가능	가능	Normal	9-29
	모든 이벤트에 시간 설정		가능	가능	사이클 타임의 10%	9-29
FINS 보호	네트워크를 통한 FINS 쓰기 방지 설정		가능	가능	FINS 쓰기 방지 무효.	9-30
	확장 FINS 쓰기 액세스 로그		불가	가능 *2	확장 FINS 쓰기 액세스 로그 사용	
I/O 모듈	펄스 I/O 모듈의 기능 할당과 상세 설정.		불가	가능	---	9-32

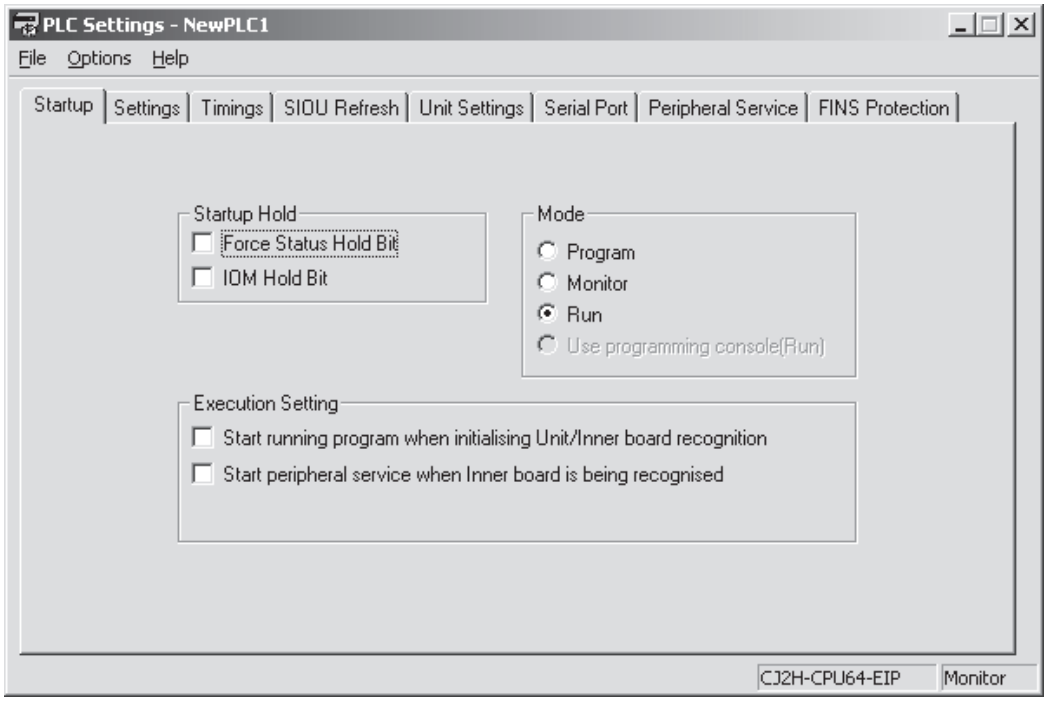
*1 CJ3 CPU의 디폴트는 "검출하지 않음"입니다.

*2 FINS 액세스 로그 설정은 CJ3 CPU 유닛에서만 지원됩니다.

AI 번역본 For Reference Only

9-3 PLC 설정 항목 설명

9-3-1 기동 시 동작 설정



기동 시 유지 설정

● 강제 상태 유지 비트 기동 시 유지 설정

이 파라미터는 기동 시에 보조 기억 영역의 강제 상태 유지 비트(A500.13)를 유지할지 여부를 설정하는 데 사용합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
강제 상태 유지 비트 기동 시 유지 설정	OFF: 클리어 ON: 유지	OFF	이 설정은 기동 시에 강제 상태 유지 비트(A500.13)의 상태를 유지할지 여부를 정합니다.	A500.13 (강제 상태 유지 비트)

AI 번역본 -For Reference Only-

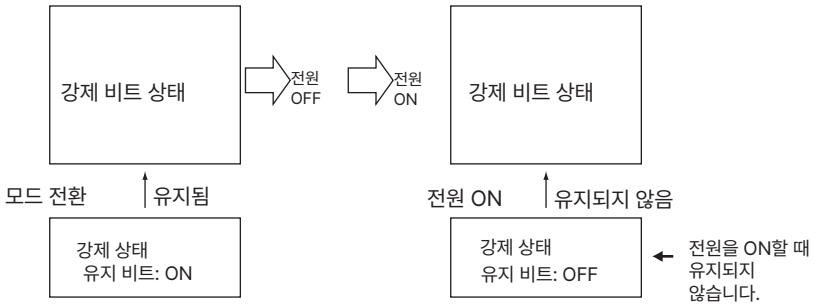
・기동 시 강제 상태 유지 비트와 이 파라미터의 관계

강제 상태 유지 비트(A500.13)를 ON하면 CPU 유닛의 동작 모드가 PROGRAM 모드와 MONITOR 모드 사이에서 전환될 때 강제 세트 또는 강제 리셋된 모든 비트의 강제 상태를 유지할 수 있습니다. PLC의 전원을 ON하면, 이 PLC 설정 파라미터로 보호하지 않는 한 강제 상태 유지 비트 자체가 OFF됩니다.

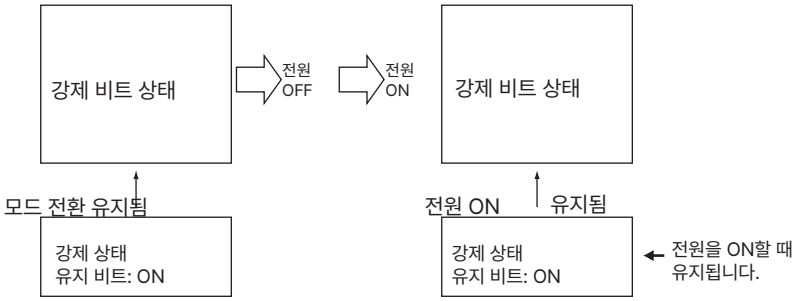
강제 상태 유지 비트 기동 시 유지 파라미터가 ON으로 설정되어 있으면 PLC의 전원을 ON할 때 강제 상태 유지 비트의 상태가 보호됩니다. 이 파라미터가 ON으로 설정되어 있고 강제 상태 유지 비트 자체도 ON이면, PLC의 전원을 ON할 때 강제 세트 및 강제 리셋된 모든 비트가 강제 상태를 유지합니다.

CJ2 CPU 유닛에서는 배터리 전압 저하 등의 이유로 메모리가 유지되지 않는 경우, 이 설정과 관계없이 강제 상태 유지 비트가 클리어됩니다.

OFF: 기동 시 강제 상태 유지 비트 클리어



ON: 기동 시 강제 상태 유지 비트 보호



● IOM 유지 비트 기동 시 유지 설정

이 파라미터는 기동 시에 보조 기억 영역의 IOM 유지 비트 상태를 유지할지 여부를 설정하는 데 사용합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
IOM 유지 비트 기동 시 유지 설정	OFF: 클리어 ON: 유지	OFF	이 설정은 기동 시에 IOM 유지 비트(A500.12)의 상태를 유지할지 여부를 정합니다. 전원을 ON할 때 I/O 메모리의 모든 데이터를 유지하려면 IOM 유지 비트를 ON하고 이 설정을 ON으로 설정하십시오.	A500.12 (IOM 유지 비트)

· IOM 유지 비트와 이 파라미터의 관계

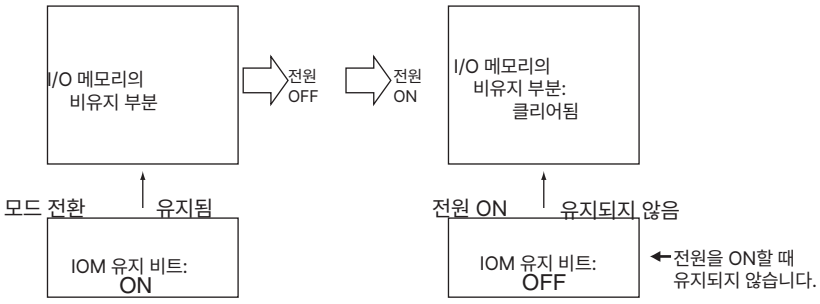
IOM 유지 비트(A500.12)를 ON하면 CPU 유닛의 동작 모드가 PROGRAM 모드에서 RUN 또는 MONITOR 모드로, 또는 그 반대로 전환될 때 I/O 메모리의 모든 데이터를 유지할 수 있습니다. PLC의 전원을 ON하면, 이 PLC 설정으로 보호하지 않는 한 IOM 유지 비트 자체가 OFF됩니다.

기동 시 IOM 유지 비트 상태 설정이 ON이면 PLC의 전원을 ON할 때 IOM 유지 비트의 상태가 보호됩니다. 이 파라미터가 ON으로 설정되어 있고 IOM 유지 비트 자체도 ON이면, PLC의 전원을 ON할 때 I/O 메모리의 모든 데이터가 유지됩니다.

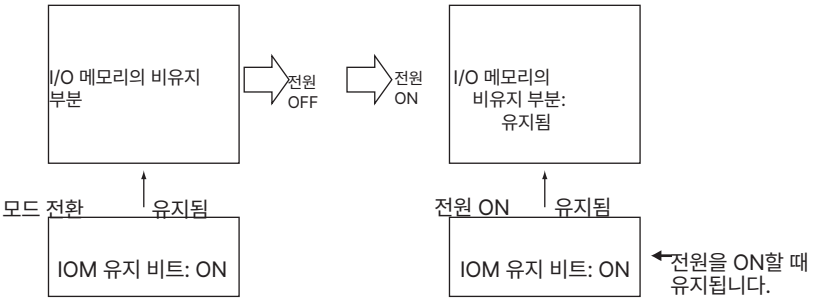
CJ2 CPU 유닛에서는 배터리 전압 저하 등의 이유로 메모리가 유지되지 않는 경우, 이 파라미터가 ON이든 OFF든 관계없이 IOM 유지 비트가 클리어됩니다.

-For Reference Only-

OFF: 기동 시 IOM 유지 비트 클리어



ON: 기동 시 IOM 유지 비트 보호



모드

● PROGRAM, MONITOR 또는 RUN

기동 시에 사용할 동작 모드를 설정합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
모드	· Program: PROGRAM 모드 · Monitor: MONITOR 모드 · Run: RUN 모드	RUN 모드	기동 시 CPU 유닛의 동작 모드를 설정합니다.	---

실행 설정

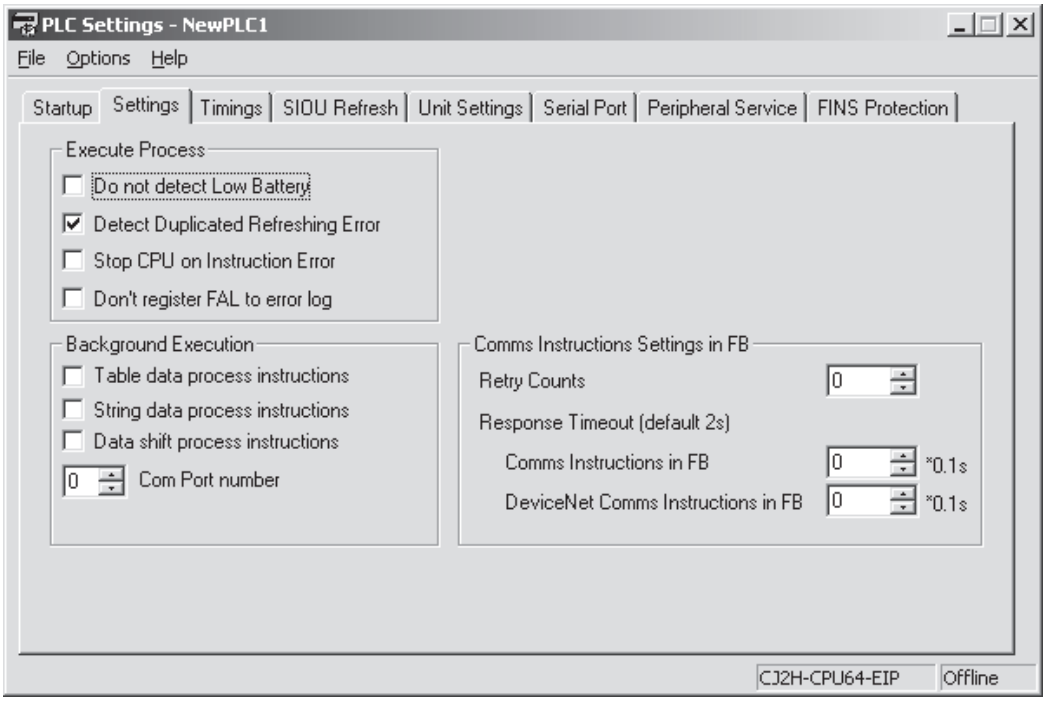
● 유닛/이너 보드 인식 초기화 시 프로그램 실행 시작

기동 시에 특정 유닛의 기동 처리를 기다릴지 여부를 설정합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
유닛/이너 보드 인식을 초기화할 때 프로그램 실행 시작	OFF: 유닛을 기다립니다. ON: 기다리지 않습니다.	OFF	기동 처리를 완료하지 않은 유닛*1이 하나 이상 있어도 CPU 유닛을 RUN 또는 MONITOR 모드로 기동하려면 이 설정을 ON (유닛을 기다리지 않음)으로 설정하십시오. 모든 유닛이 기동 처리를 마칠 때까지 기다리려면 이 설정을 OFF(유닛을 기다림)로 설정하십시오.	---

*1 이 설정은 특정 유닛에만 적용됩니다. "기다리지 않음"으로 설정하면 CPU 유닛은 그 특정 유닛만 기다리지 않으며, 그 밖의 모든 유닛의 기동은 계속 기다립니다.

9-3-2 CPU 유닛 설정



실행 처리

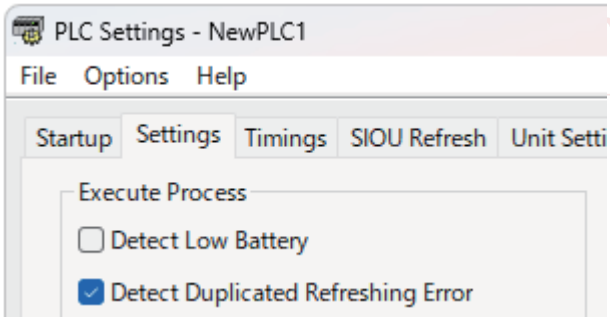
- **배터리 전압 저하 검출하지 않음(배터리리스 운전)(CJ2 CPU 유닛)**

  배터리 에러를 검출할지 여부를 설정합니다.

  배터리 없이 운전하려면 "ON"(검출하지 않음)으로 설정하십시오.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
배터리 전압 저하 검출하지 않음	OFF: 검출함 ON: 검출하지 않음	OFF (검출함)	이 설정은 CPU 유닛의 배터리 에러를 검출할지 여부를 정합니다. 이 설정을 OFF (검출)로 설정한 상태에서 배터리 에러가 검출되면 CPU 유닛의 ERR/ALM 표시등이 점멸하고 배터리 에러 플래그(A402.04)가 ON되지만, CPU 유닛의 운전은 계속됩니다.	A402.04 (배터리 에러 플래그)

- **배터리 전압 저하 검출(CJ3 CPU 유닛)**



배터리 에러를 검출할지 여부를 설정합니다.

AI 번역본
-Reference Only-

CJ3 CPU 유닛에서 클록 기능을 사용하기 위해 배터리를 접속한 경우에는 "ON"(검출)으로 설정하십시오.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
배터리 저전압 검출	OFF: 검출 안 함 ON: 검출	OFF (검출 안 함)	이 설정은 CPU 유닛의 배터리 에러를 검출할지 여부를 정합니다. 이 설정을 ON (검출)으로 설정한 상태에서 배터리 에러가 검출되면 CPU 유닛의 ERR/ALM 표시등이 점멸하고 배터리 에러 플래그(A402.04)가 ON되지만, CPU 유닛의 운전은 계속됩니다.	A402.04 (배터리 에러 플래그)

참고

배터리리스 운전

・CJ2 CPU 유닛에서의 배터리리스 운전

CPU 유닛이 배터리 없이 동작할 때 다음 데이터를 유지할 수 있습니다.

	유지되는 I/O 메모리 부분*	클록	사용자 프로그램 또는 파라미터
CJ2 CPU 유닛	유지 안 됨	유지 안 됨	유지됨

* DM 영역, EM 영역, 유지 영역, 보조 기억 영역, 카운터

・설정 방법

PLC 설정

PLC 설정을 아래와 같이 설정하십시오.

(1) 배터리가 없으면 I/O 메모리가 불안정해지므로, IOM 유지 비트 체크 박스를 해제하여 기동 시에 IOM 유지 비트가 클리어되도록 하십시오.

(2) 배터리가 없으면 강제 상태가 불안정해지므로, 강제 상태 유지 비트 체크 박스를 해제하여 기동 시에 강제 상태 유지 비트가 클리어되도록 하십시오.

(3) 배터리 저전압을 검출하지 않음 체크 박스를 선택하십시오.

프로그래밍

배터리가 없으면 출력 OFF 비트의 동작이 불안정해지므로, 출력 OFF 비트가 ON되지 않도록 다음 명령어를 삽입하십시오.

```
LD P_Off
OUT A500.15
```

・주의사항

배터리 없이 운전할 때는 다음 사항에 주의하십시오.

・데이터 초기화
DM 영역, EM 영역 등 I/O 메모리 영역의 값이 불안정해집니다. 반드시 프로그램에서 초기값을 설정하십시오.

AI 번역본
For Reference Only-

・CJ3 CPU 유닛에서의 배터리리스 운전

CJ3 CPU 유닛은 배터리리스 운전이 디폴트 설정입니다.
I/O 메모리는 배터리 없이도 유지되므로 설정할 필요가 없습니다.
CPU 유닛이 배터리 없이 동작할 때 다음 데이터를 유지할 수 있습니다.

	유지되는 I/O 메모리 부분*	클록	사용자 프로그램 또는 파라미터
CJ3 CPU 유닛	유지됨	유지 안 됨	유지됨

사용상의 주의

CJ2 CPU 유닛의 DM, EM, HR, AR 영역의 내용은 전원이 OFF되거나 정전되었을 때 배터리에 의해 유지됩니다. (AR 영역의 일부 어드레스 내용도 유지됩니다.) 배터리 에러가 있으면 이 데이터가 소실될 수 있습니다. 배터리 에러 플래그(A402.04)를 사용하여 데이터를 다시 초기화하는 대책을 프로그램에 마련하십시오.

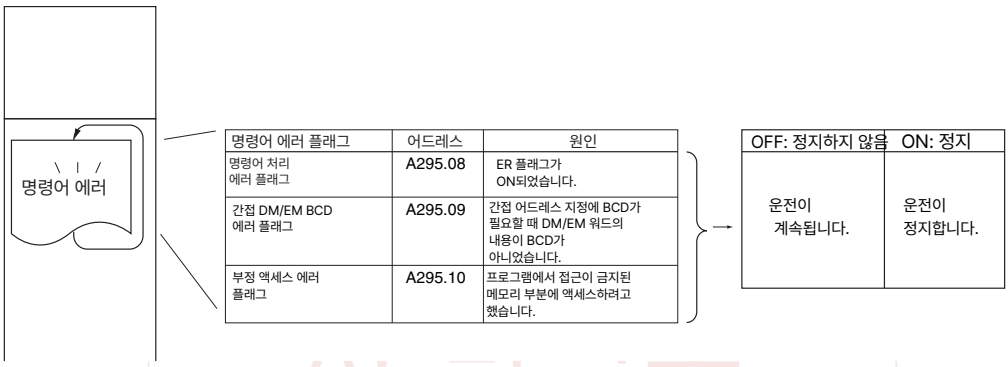
● 중복 리프레시 에러 검출 (중복 리프레시 에러를 검출할지 여부 설정)

이 설정은 중복 리프레시 에러를 검출할지 여부를 정합니다(디폴트: 검출). 이 파라미터가 디폴트(ON: 검출)로 설정되어 있으면 다음의 경우에 중복 리프레시 에러가 검출됩니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
중복 리프레시 에러 검출	OFF: 검출하지 않음 ON: 검출	ON	이 설정은 중복 리프레시 에러를 검출할지 여부를 정합니다. 이 설정을 ON(검출)으로 설정한 상태에서 에러가 검출되면 CPU 유닛의 ERR/ALM 표시등이 점멸하고 중복 리프레시 에러 플래그(A402.13)가 ON됩니다. CPU 유닛의 운전은 계속됩니다.	A402.13 (중복 리프레시 에러 플래그)

● 명령어 에러 시 CPU 정지

이 설정은 명령어 실행 에러가 발생했을 때 운전을 정지할지 여부를 정합니다 (디폴트: 정지하지 않음). 명령어 에러 시 CPU 유닛을 정지시키는 기능은 프로그램을 디버깅할 때 사용합니다. 명령어 에러 시 CPU 정지가 CPU 유닛을 정지시키도록 설정되어 있으면 다음 플래그 중 하나가 ON될 때 명령어 에러로서 프로그램 에러가 발생합니다.



파라미터	설정	디폴트	기능	관련 플래그 및 워드
명령어 에러 시 CPU 정지	OFF: 계속 ON: 정지	OFF	이 설정은 명령어 에러(명령어 처리 에러(ER)와 불법 액세스 에러(AER))를 비치명 에러로 취급할지 치명 에러로 취급할지 정합니다.	A295.08(명령어 처리 에러 플래그), A295.09(간접 DM/EM BCD 에러 플래그, A295.10(부정 액세스 에러 플래그) (이 설정을 OFF로 설정하면 명령어 에러가 발생해도 이 플래그들은 ON되지 않습니다.)

● 에러 로그에 FAL을 등록하지 않음

이 파라미터는 사용자가 프로그래밍한 FAL 에러가 발생했을 때 그 에러를 에러 로그에 등록할지 여부를 정합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
에러 로그에 FAL을 등록 안 함	OFF: 사용자 정의 FAL 에러를 에러 로그에 기록합니다. ON: 사용자 정의 FAL 에러를 에러 로그에 기록하지 않습니다.	OFF	이 설정은 FAL(006)으로 생성되는 사용자 정의 FAL 에러와 FPD(269)의 시간 감시로 생성되는 에러를 에러 로그(A100~A199)에 기록할지 여부를 정합니다. 이 에러들이 기록되지 않도록 하려면 ON으로 설정하십시오.	---

백그라운드 실행 설정

다음 명령어는 그 명령어가 다루는 데이터 양에 따라 사이클 타임에 큰 영향을 미칩니다. 백그라운드 실행 설정을 사용하면 명령어의 처리를 여러 사이클에 걸쳐 나누어 사이클 타임에 미치는 영향을 줄일 수 있습니다. 자세한 내용은 10-2-5 백그라운드 실행을 참조하십시오.

● 테이블 데이터 처리 명령어

이 파라미터는 테이블 데이터 명령어를 백그라운드에서 처리할지 여부를 정합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
테이블 데이터 처리 명령어	OFF: 백그라운드에서 실행하지 않음. ON: 백그라운드에서 실행함.	OFF	이 설정은 테이블 데이터 명령어를 여러 사이클 타임에 걸쳐 처리할지(즉, 백그라운드에서 처리할지) 여부를 정합니다.	---

● 문자열 데이터 처리 명령어

이 파라미터는 문자열 데이터 명령어를 백그라운드에서 처리할지 여부를 정합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
문자열 데이터 처리 명령어	OFF: 백그라운드에서 실행하지 않음. ON: 백그라운드에서 실행함.	OFF	이 설정은 문자열 데이터 명령어를 여러 사이클 타임에 걸쳐 처리할지(즉, 백그라운드에서 처리할지) 여부를 정합니다.	---

AP 문서
-For Reference Only-

● 데이터 시프트 처리 명령어

이 파라미터는 데이터 시프트 명령어를 백그라운드에서 처리할지 여부를 정합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
데이터 시프트 처리 명령어	OFF: 백그라운드에서 실행하지 않음. ON: 백그라운드에서 실행함.	OFF	이 설정은 데이터 시프트 명령어를 여러 사이클 타임에 걸쳐 처리할지(즉, 백그라운드에서 처리할지) 여부를 정합니다.	---

● 통신 포트 번호

이 파라미터는 백그라운드 처리에 사용하는 통신 포트 번호를 설정합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
통신 포트 번호	0~7: 통신 포트 0~7 (내부 논리 포트)	0 (No. 0)	백그라운드 실행에 사용할 통신 포트 번호(내부 논리 포트)를 나타냅니다.	---

사용상의 주의

PLC 설정에서 고속 인터럽트 기능이 유효한 경우 인터럽트 태스크에서는 백그라운드 처리를 사용할 수 없습니다. 명령어 처리 에러가 발생합니다.

FB 내 통신 명령어 설정(FB 라이브러리용)

다음 파라미터는 오므론 펑션 블록에만 사용됩니다. 그 밖의 용도로는 사용되지 않습니다.

특히 오므론 FB 라이브러리의 펑션 블록을 사용하여 FINS 메시지나 DeviceNet 익스플리싯 메시지 통신을 실행할 때는, PLC 설정의 FB 통신 명령어 설정에서 재송신 횟수와 응답 감시 시간을 사용자가 설정해야 합니다. 오므론 FB 라이브러리용으로 이 PLC 설정에서 설정한 값은 관련 보조 기억 영역 워드 A580~A582에 자동으로 저장되어 오므론 FB 라이브러리의 펑션 블록에서 사용됩니다.

● 리트라이 횟수

이 파라미터는 펑션 블록 내에서 통신 명령어를 실행할 때의 리트라이 횟수를 정합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
리트라이 횟수	0~15	0	펑션 블록 내에서 DeviceNet 익스플리싯 메시지나 FINS 메시지를 실행할 때 커맨드를 송신하는 리트라이 횟수를 설정합니다.	A580.00~ A580.03

● FB 내 통신 명령어의 응답 타임아웃

이 파라미터는 펑션 블록에서 통신 명령어를 실행할 때의 응답 감시 시간을 정합니다.

사본 번역본

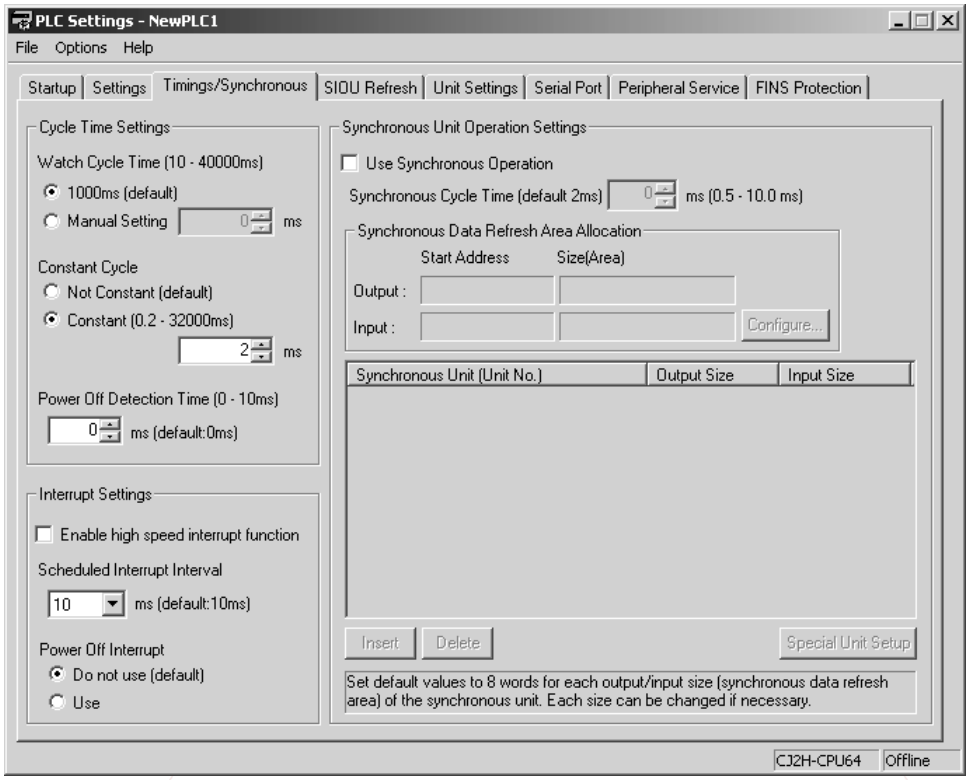
파라미터	설정	디폴트	기능	관련 플래그 및 워드
FB 내 통신 명령어 설정	0001~65535 (단위: 0.1 s, 0.1~6553.5) 0000: 2 s	0000: 2 s	펑션 블록 내에서 실행된 FINS 커맨드에 대해 여기서 설정한 시간 내에 응답이 반환되지 않으면 응답 타임아웃이 발생합니다.	A581

● FB 내 DeviceNet 통신 명령어의 응답 타임아웃

이 파라미터는 펑션 블록에서 DeviceNet 통신 명령어를 실행할 때의 응답 감시 시간을 정합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
FB 내 DeviceNet 통신 명령어	0001~FFFF (단위: 0.1 s, 0.1~6553.5) 0000: 2 s	0000: 2 s	펑션 블록 내에서 실행된 익스플리싯 메시지 커맨드에 대해 여기서 설정한 시간 내에 응답이 반환되지 않으면 응답 타임아웃이 발생합니다.	A582

9-3-3 타이밍/동기 설정



● 감시 사이클 타임

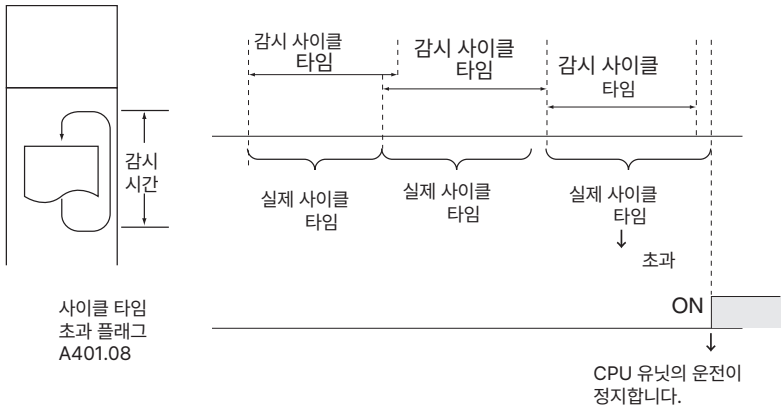
이 파라미터는 감시 사이클 타임을 디폴트(1000 ms) 이외의 값으로 설정할 때 사용합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
감시 사이클 타임	ON: 디폴트 (1,000 ms: 1 s) OFF: 수동 설정	ON (1,000 ms: 1 s)	디폴트 값인 1 s 이외의 감시 사이클 타임을 설정하려면 OFF로 설정하십시오.	A401.08 (사이클 타임 초과 플래그)
	10~40,000 ms (10 ms 단위)	0	이 설정은 위 파라미터가 OFF(수동 설정)로 설정되어 있을 때만 유효합니다. 사이클 타임이 감시 사이클 타임으로 설정한 값을 초과하면 A401.08(사이클 타임 과대 플래그)이 ON됩니다.	A264 및 A265 (현재 사이클 타임)

AI 번역본

· 감시 사이클 타임을 변경해야 할 때

사이클 타임이 감시(최대) 사이클 타임 설정을 초과하면 사이클 타임 초과 플래그 (A401.08)가 ON되고 PLC 운전이 정지합니다. 통상의 사이클 타임이 디폴트 감시 사이클 타임 설정인 1 s를 초과하는 경우에는 이 파라미터를 변경해야 합니다.



주 감시 사이클 타임의 디폴트 값은 1 s(1,000 ms)입니다.

● 사이클 타임

이 파라미터는 최소 사이클 타임 기능을 사용할 때의 최소 사이클 타임을 설정합니다 (디폴트: 가변 사이클 타임).

파라미터	설정	디폴트	기능	관련 플래그 및 워드
일정 사이클 타임	OFF: 최소 사이클 타임 ON: 가변	ON (가변)	최소 사이클 타임을 사용하려면 이 설정을 OFF로 설정하십시오. 최소 사이클 타임을 사용하는 경우 사이클 타임을 설정해야 합니다.	---
	0.2~32,000 ms (0.1 ms 단위)	---	최소 사이클 타임을 지정하려면 0.2~32000.0으로 설정하십시오. 사이클 타임이 이 설정값보다 짧으면 이 시간이 경과할 때까지 연장됩니다. 가변 사이클 타임으로 하려면 이 설정을 0으로 두십시오.	---

참고

CPU 유닛이 MONITOR 모드일 때 사이클 타임을 모니터하는 CX-Programmer 화면에서 최소 사이클 타임을 변경할 수 있습니다(유닛 버전 1.1 이후만 해당).

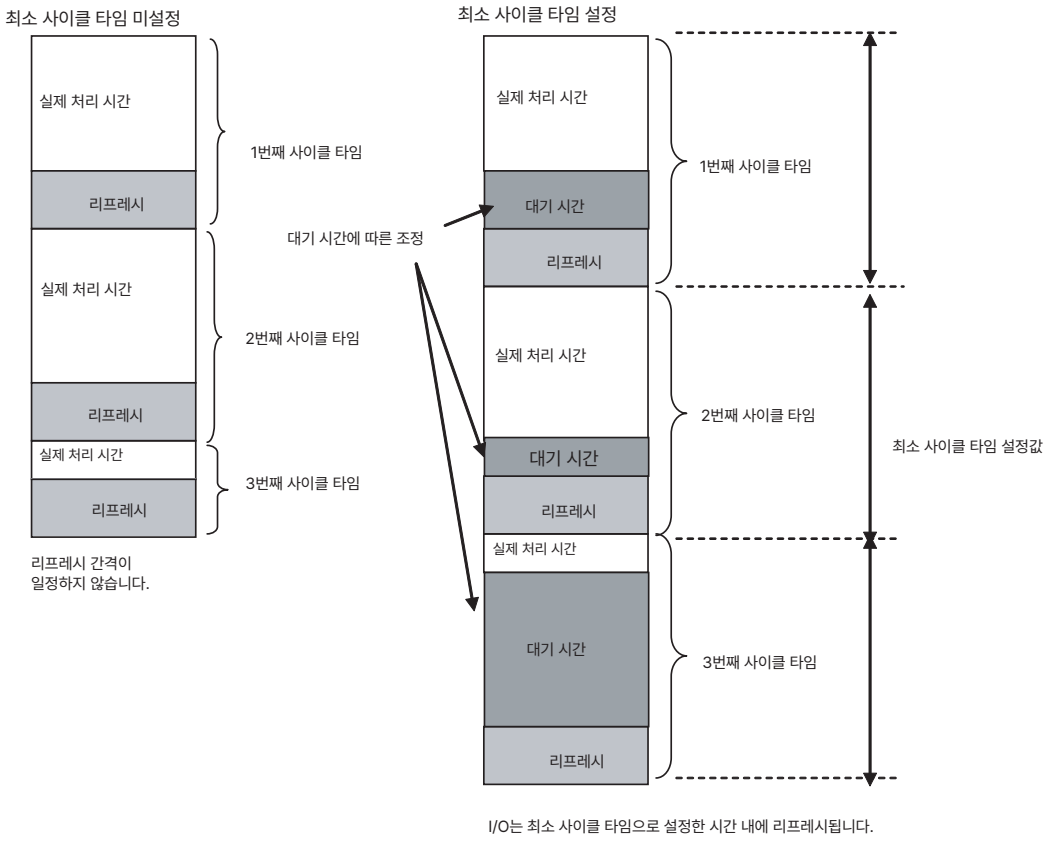
사용상의 주의

사이클 타임을 길게 설정하면 지원 소프트웨어의 서비스 간격도 길어져, 온라인 조작의 응답이 나빠지거나 온라인 접속이 어려워질 수 있습니다.

AI 번역본
-For Reference Only-

· 최소 사이클 타임이 유효해지는 조건

I/O 응답의 불균일을 없애려면 최소 사이클 타임을 0 이외의 값으로 설정하십시오. 이 파라미터는 실제 사이클 타임이 최소 사이클 타임 설정보다 짧을 때만 유효합니다. 실제 사이클 타임이 최소 사이클 타임 설정보다 길면 실제 사이클 타임은 변경되지 않고 그대로 유지됩니다.



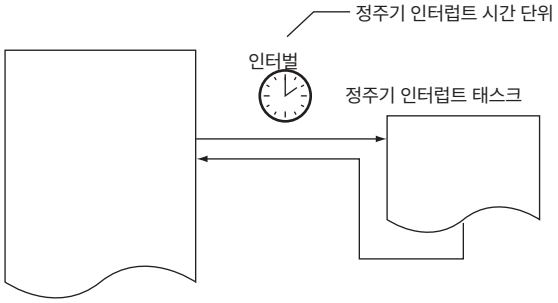
● 정주기 인터럽트 인터벌

이 파라미터는 정주기 인터럽트 인터벌에 사용하는 시간 단위를 1 ms, 0.1 또는 디폴트인 10 ms로 설정합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
정주기 인터럽트 인터벌	· 10 ms · 1.0 ms · 0.1 ms	10 ms	이 설정은 정주기 인터럽트 인터벌 설정에 사용하는 시간 단위를 정합니다.	---

· 정주기 인터럽트 시간 단위

이 파라미터는 정주기 인터럽트 인터벌 설정의 시간 단위를 설정합니다. 정주기 인터럽트 인터벌은 프로그램에서 MSKS(690)으로 설정하십시오.



AI 번역본
-For Reference Only-

● 전원 OFF 검출 지연 시간

이 파라미터는 전원 OFF 검출 지연 시간을 0 ms 이외의 값으로 설정할 때 사용합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
전원 OFF 검출 시간	0~10 ms (1 ms 단위)	0 ms	이 설정은 정전 검출(전원 전압이 정격값의 85% 미만으로 떨어진 후 AC 전원은 약 10~25 ms, DC 전원은 약 2~5 ms)부터 정전 확정까지 어느 정도의 지연을 둘지 정합니다. 디폴트 설정은 0 ms입니다. 전원 OFF 인터럽트 태스크가 유효한 경우, 정전이 확정되면 그 태스크가 실행됩니다. 전원 OFF 인터럽트 태스크가 무효인 경우에는 CPU가 리셋되고 운전이 정지합니다.	---

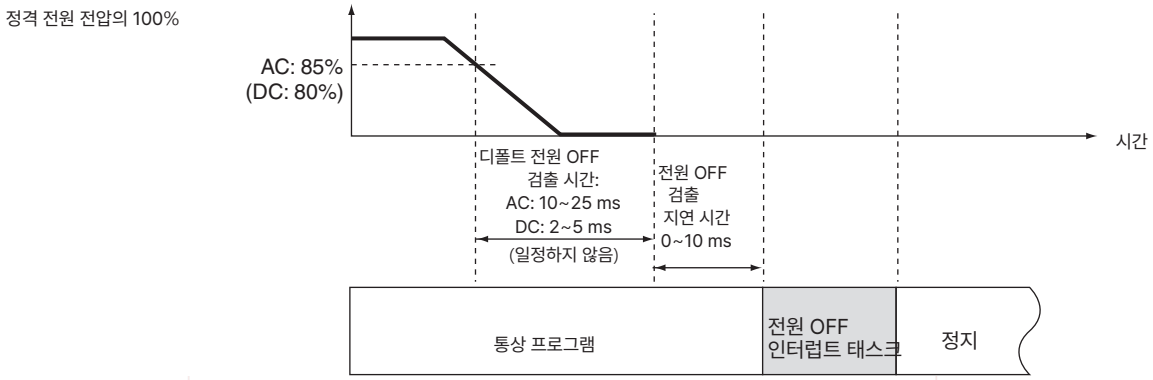
주 CJ1W-PD022 전원 유닛은 전원 OFF 검출 지연을 지원하지 않습니다. 이 설정을 사용하지 마십시오.

전원 OFF 검출 지연 시간

이 파라미터는 정전을 검출한 시점부터 정전이 확정되어 통상 프로그램이 정지할 때까지 어느 정도의 지연을 둘지 정합니다. 설정은 0~10 ms 사이로 할 수 있습니다.

전원 상태가 나빠 순시 정전으로 PLC 운전이 정지하는 경우에는 정전을 검출하기까지의 시간을 연장하십시오.
최초의 정전 검출 시간이 지난 후 내부 5 VDC 전원이 0 VDC까지
떨어지는 데 최대 10 ms가 걸립니다. 이 10 ms가 전원 유지 시간입니다. 이는 전원 OFF 검출 시간에 정전을 확정하는 데 필요한 처리 시간을 더한 값입니다. 전원 OFF 지연 시간은 AC 전원의 경우 10~25 ms(일정하지 않음), CJ1W-PD025 DC 전원 유닛의 경우 2~5 ms, CJ1W-PD022 DC 전원 유닛의 경우 2~10 ms입니다.

주 CJ1W-PD022 전원 유닛을 장착한 경우 이 파라미터는 지원되지 않습니다. (A-5 정전 시의 동작을 참조하십시오.)



주 전원 OFF 인터럽트 태스크 내 프로그램의 실행 시간은 10 ms에서 전원 OFF 검출 지연 시간을 뺀 시간보다 짧아야 합니다. 전원 OFF 인터럽트 태스크에 대해서는 5-2-3 인터럽트 태스크를 참조하십시오. 전원 OFF 시의 동작에 대해서는 A-5 정전 시의 동작을 참조하십시오.

Preliminary Only

● 전원 OFF 인터럽트

이 파라미터는 전원 OFF 인터럽트 태스크를 유효하게 할지 여부를 정합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
전원 OFF 인터럽트	ON: 무효 OFF: 유효	ON	이 설정을 OFF(유효)로 설정하면 정전 시에 전원 OFF 인터럽트 태스크가 실행됩니다.	---

주 CJ1W-PD022 전원 유닛을 장착한 경우 이 파라미터는 지원되지 않습니다.

● 전원 OFF 인터럽트 태스크

이 파라미터는 정전이 검출되었을 때 전원 OFF 인터럽트 태스크를 실행할지 여부를 정합니다. (이 파라미터를 ON으로 설정하면 정전이 검출되었을 때 통상 프로그램이 정지하기만 합니다.)

전원 유지 시간(정전 후 처리 시간 + 전원 OFF 검출 지연 시간)이 경과하면 전원 OFF 인터럽트 태스크가 정지합니다. 전원 유지 시간의 최대값은 10 ms입니다.

전원 OFF 검출 지연 시간을 설정해야 하는 경우에는, 사용할 수 있는 시간(10 ms에서 전원 OFF 검출 지연 시간을 뺀 시간) 내에 전원 OFF 인터럽트 태스크를 실행할 수 있도록 하십시오.

주 CJ1W-PD022 전원 유닛을 장착한 경우 이 파라미터는 지원되지 않습니다. (A-5 정전 시의 동작을 참조하십시오.)

● 고속 인터럽트 기능 유효화 (CJ2H CPU 유닛 전용)

이 파라미터는 고속 인터럽트 기능을 유효 또는 무효로 할 때 사용합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
고속 인터럽트 기능 유효화	OFF: 고속 인터럽트 기능을 무효로 합니다. ON: 고속 인터럽트 기능을 유효로 합니다.	OFF (고속 인터럽트 무효.)	고속 인터럽트 기능 유효화 옵션을 선택하면 고속 인터럽트 기능이 유효해집니다.	---

고속 인터럽트 기능은 일정한 제약 조건 하에서 인터럽트 태스크의 실행을 향상시킵니다. 자세한 내용은 10-2-6 고속 인터럽트 기능을 참조하십시오.

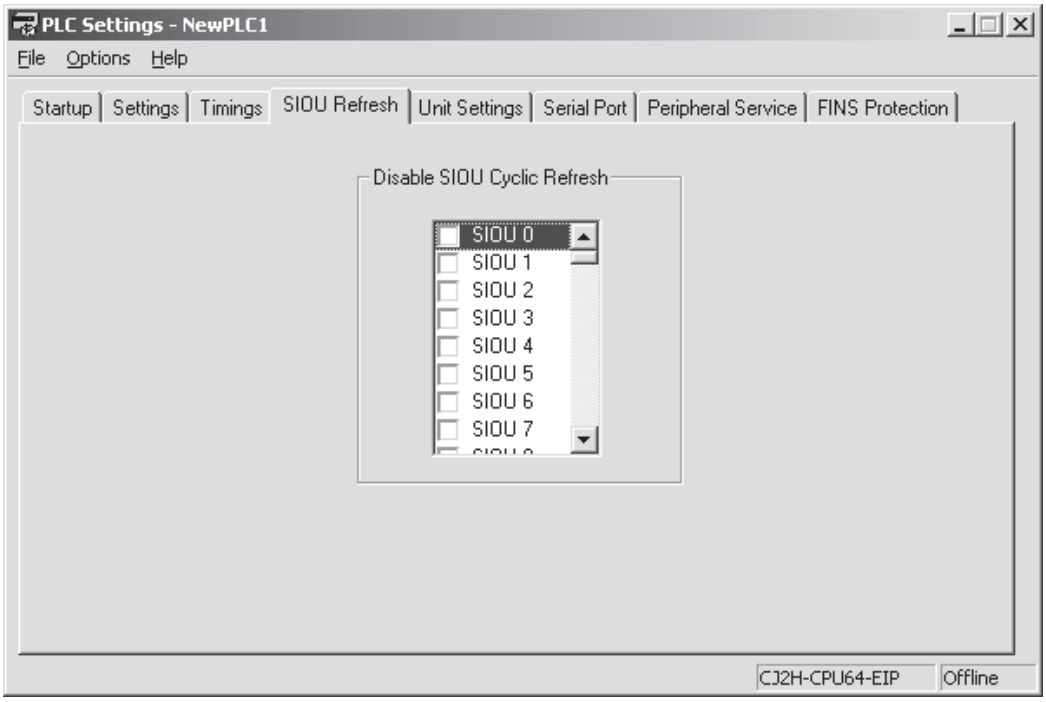
PLC 설정에서 동기 동작 사용 체크 박스를 선택하면 고속 인터럽트 기능 유효화 체크 박스가 자동으로 선택됩니다.

AI 번역본
-For Reference Only-

● 동기 유닛 동작 설정 (CJ2H CPU 유닛 전용)

이 설정은 동기 유닛 동작 기능을 사용하는 데 필요합니다. 자세한 내용은 10-8-7 PLC 설정을 참조하십시오.

9-3-4 특수 I/O 유닛 사이클릭 리프레시



SIOU 사이클릭 리프레시 무효

이 파라미터는 특수 I/O 유닛(SIOU)의 사이클릭 리프레시를 무효로 할지 여부를 지정합니다.

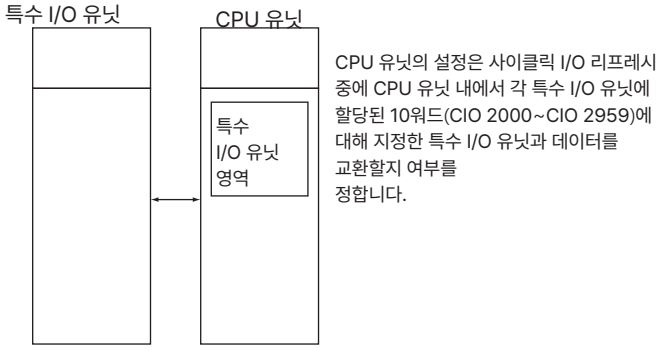
파라미터	설정	디폴트	기능	관련 플래그 및 워드
유닛 0~95의 사이클릭 리프레시 유닛 15대마다 파라미터가 하나씩 있습니다	OFF: 유효 ON: 무효	OFF (무효)	이 설정은 특수 I/O 유닛의 사이클릭 리프레시 중에 지정한 유닛과 특수 I/O 유닛의 할당 워드 (유닛당 10워드) 사이에서 데이터를 교환할지 여부를 정합니다.	---

AI 번역본
-For Reference Only-

- **특수 I/O 유닛의 사이클릭 리프레시 금지:**

  인터럽트 태스크에서 IORF(097), FIORF(225), IORD(222) 또는 IOWR(223) 명령어로 특수 I/O 유닛을 리프레시하는 경우에는 반드시 특수 I/O 유닛의 사이클릭 리프레시를 무효로 하십시오. 특수 I/O 유닛의 사이클릭 리프레시가 유효한 상태에서 인터럽트 태스크에서 다음 중 하나가 실행되면 중복 리프레시 에러(비치명)가 발생하고 중복 에러 리프레시 플래그(A402.13)가 ON됩니다.

  · IORF(097)/FIORF(225) 명령어로 동일한 특수 I/O 유닛의 I/O를 리프레시하는 경우
  · IORD(222)/IOWR(223) 명령어로 동일한 특수 I/O 유닛의 메모리 영역 데이터를 읽거나 쓰는 경우

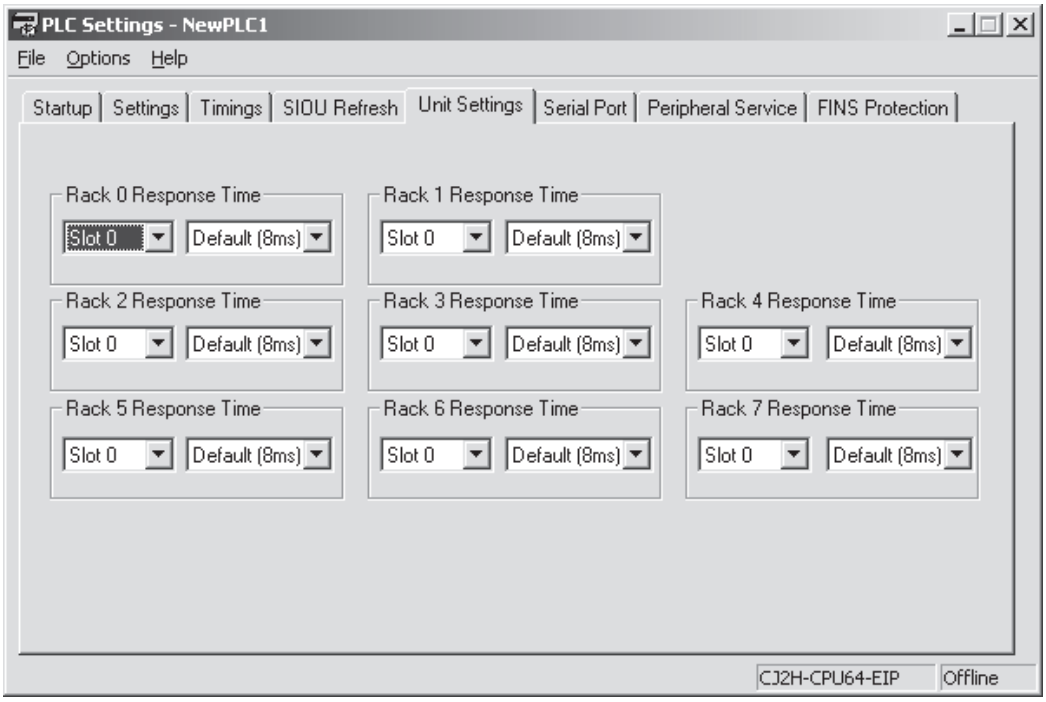


사용상의 주의

특수 I/O 유닛의 사이클릭 리프레시를 무효로 하는 경우에는, 운전 중에 프로그램에서 최소 11초마다 IORF(097) 또는 FIORF(225)로 그 유닛의 I/O를 리프레시하도록 하십시오. 11초마다 리프레시하지 않으면 특수 I/O 유닛에서 CPU 유닛 서비스 감시 에러가 발생합니다. (특수 I/O 유닛의 ERH 표시등과 RUN 표시등이 점등합니다.)

AI 번역본
-For Reference Only-

9-3-5 기본 I/O 유닛 랙 응답 시간



랙 0~7의 랙 응답 시간

이 파라미터는 기본 I/O 유닛의 입력 응답 시간을 디폴트인 8 ms 이외의 값으로 설정할 때 사용합니다.

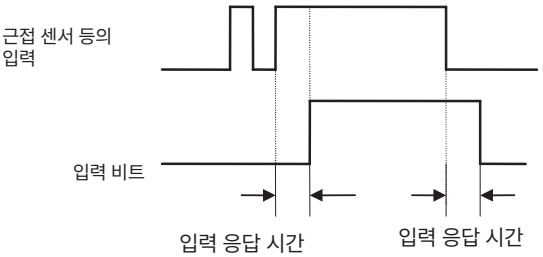
PLC 설정을 CPU 유닛에 전송한 후에는 PLC의 전원을 OFF했다가 다시 ON해야 합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
랙 0, 슬롯 0~9 랙 1, 슬롯 0~9 랙 2, 슬롯 0~9 랙 3, 슬롯 0~9 랙 4, 슬롯 0~9 랙 5, 슬롯 0~9 랙 6, 슬롯 0~9 랙 7, 슬롯 0~9	디폴트: 8 ms 필터 없음 0.5 ms 1 ms 2 ms 4 ms 8 ms 16 ms 32 ms	8 ms	CJ 시리즈 기본 I/O 유닛의 입력 응답 시간 (ON 응답 시간 = OFF 응답 시간)을 설정합니다. 필터를 설정하지 않으면(0 ms) 유닛 내부 소자의 지연으로 인해 ON 또는 OFF 응답 시간에 지연이 발생합니다. 각 유닛의 ON/OFF 응답 시간 지연에 대한 자세한 내용은 CJ2 CPU 유닛 하드웨어 사용자 매뉴얼(Cat. No. W472) 또는 CJ3 CPU 유닛 하드웨어 사용자 매뉴얼(Cat. No. W664)의 A-1 기본 I/O 유닛의 사양을 참조하십시오.	A220~A259: 기본 I/O 유닛의 실제 입력 응답 시간

AI 번역 본
-For Reference Only-

● 기본 I/O 유닛 랙 응답 시간의 변경

기본 I/O 유닛의 랙 응답 시간은 랙과 슬롯별로 설정할 수 있습니다. 설정값을 크게 하면 채터링과 노이즈의 영향이 줄어듭니다. 설정값을 작게 하면 더 짧은 입력 펄스를 수신할 수 있습니다. ON 응답 시간이나 OFF 응답 시간을 사이클 타임보다 짧게 설정하지 마십시오.
입력 응답 시간은 필터로서의 기준값입니다. 설정값보다 펄스 폭이 짧은 입력이 수신될 수도 있습니다. 실제 펄스를 사용하여 필터의 영향에 따라 설정값을 조정하십시오.

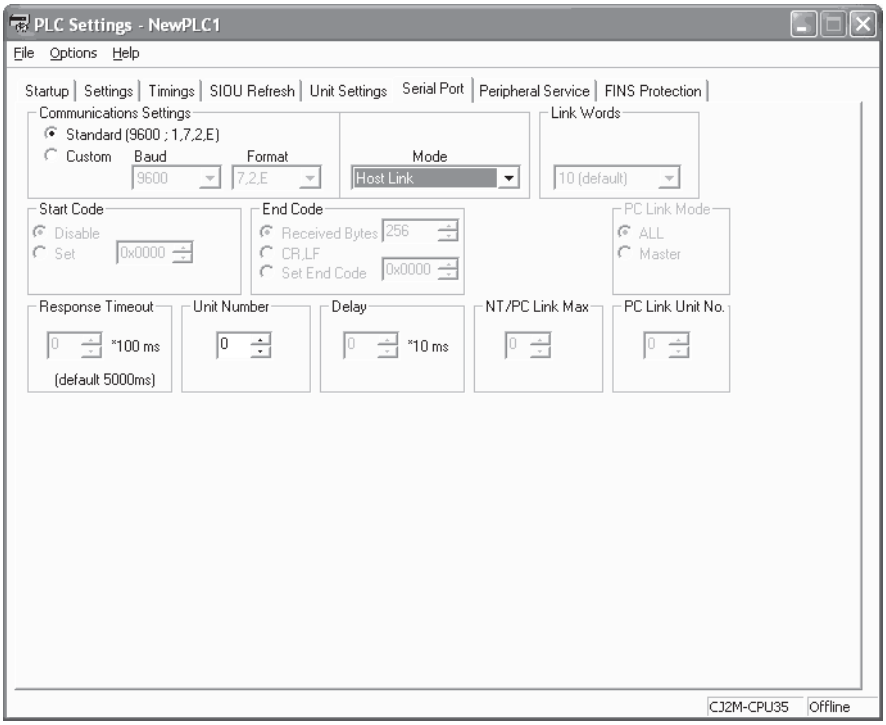


랙 응답 시간 설정은 PLC가 ON될 때 기본 I/O 유닛에 전송됩니다.

유닛의 설정이 변경되면 그 설정은 A220~A259(기본 I/O 유닛의 실제 입력 응답 시간)에 저장됩니다. PLC가 PROGRAM 모드인 상태에서 PLC 설정을 변경한 경우, PLC 설정의 설정값과 유닛의 실제 설정값이 달라집니다. 이 경우 A220~A259의 값을 확인하면 유닛에 실제로 설정된 입력 응답 시간을 확인할 수 있습니다.

AI 번역본
-For Reference Only-

9-3-6 시리얼 포트 설정



다음 파라미터는 CPU 유닛의 딥 스위치 핀 5가 OFF(디폴트)일 때 유효합니다.

통신 설정

파라미터	설정	디폴트	기능	관련 플래그 및 워드
통신 설정	· Standard (9600: 1, 7, 2, E): 표준 설정 · Custom: 임의 설정	Standard	표준 설정(9600: 1, 7, 2, E)은 호스트 링크 모드, 스타트 비트 1, 데이터 비트 7, 짝수 패리티, 스톱 비트 2, 보 레이트 9,600 bps를 뜻합니다. 그 밖의 통신 설정을 사용하려면 Custom 설정을 지정하십시오.	A619.02 (시리얼 포트 설정 변경 플래그)

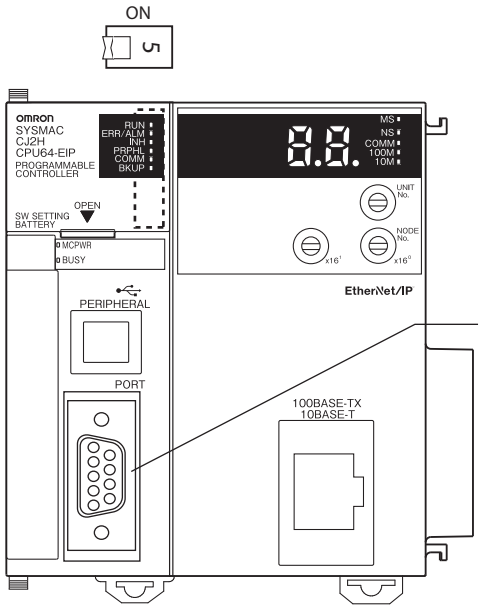
· 시리얼 포트 설정

이 설정을 디폴트에서 변경해야 하는 경우에는 PLC 설정에서 시리얼 포트 설정을 설정하십시오. 무순서 모드를 선택한 경우에는 프레임 포맷을 지정하십시오.

포트 설정은 STUP(237)로도 변경할 수 있습니다. STUP(237)이 실행되면 시리얼 포트 설정 변경 플래그(A619.02)가 ON되고, 포트 설정 변경이 완료되면 OFF됩니다.

CPU 유닛 전면의 딥 스위치 핀 5가 ON이면 CPU 유닛은 RS-232C 포트에 접속된 프로그래밍 디바이스(프로그래밍 콘솔 포함)의 통신 파라미터를 자동으로 검출합니다. 자동으로 검출된 이 파라미터는 PLC 설정에 저장되지 않습니다.

AI 번역본 -For Reference Only-

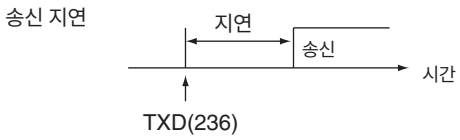


딥 스위치 핀 5가 OFF일 때의 시리얼 포트 통신 설정:

디폴트 설정:
호스트 링크 모드, 스타트 비트 1, 데이터 비트 7, 짝수 패리티, 스톱 비트 2, 보 레이트 9,600 bps

Custom 설정:
통신 모드(호스트 링크, NT 링크, 무순서, 페리페럴 버스)와 보 레이트 등 그 밖의 설정을 설정합니다.

무순서 모드에서는 송신 지연을 설정할 수 있습니다. 이 지연의 동작은 다음 그림과 같습니다.



다음 표는 무순서 모드에서 송신과 수신에 설정할 수 있는 메시지 포맷을 나타냅니다.

		엔드 코드 설정		
		없음	있음	CR+LF
스타트 코드 설정	없음	DATA	DATA | ED	DATA | CR+LF
	있음	ST | DATA	ST | DATA | ED	ST | DATA | CR+LF
수신 바이트 수		데이터: 1~256바이트		

AI 번역본
-For Reference Only-

Standard (9600: 1,7,2,E) 선택 시

Standard (9600: 1,7,2,E)를 선택하면 다음 통신 설정이 무조건 사용됩니다.

- 보 레이트: 9,600 bps
- 스타트 비트: 1비트
- 데이터 길이: 7비트
- 스톱 비트: 2비트
- 패리티: 짝수
- 모드(시리얼 통신 모드): 호스트 링크

주 모드와 그 밖의 설정은 무효가 됩니다.

통신을 Custom 설정으로 설정하려면 다음 순서를 따르십시오.

1 통신 설정에서 Custom 옵션을 선택합니다.

2 Mode 필드에서 시리얼 통신 모드를 선택합니다.

3 그 밖의 설정을 합니다.

Custom 선택 시

Custom 설정을 선택하면 다음 보 레이트, 파라미터, 모드를 선택할 수 있습니다.

보 레이트(Custom 설정 선택 시)

설정	의미	설정	의미	관련 플래그 및 워드
115200	11,5200 bps	4800	4,800 bps	A619.02 (시리얼 포트 설정 변경 플래그)
57600	57,600 bps	2400	2,400 bps	
38400	38,400 bps	1200	1,200 bps	
19200	19,200 bps	600	600 bps	
9600 (디폴트)	9,600 bps	300	300 bps	

● 포맷(Custom 설정 선택 시)

선택 데이터	데이터 길이	스톱 비트	패리티	관련 플래그 및 워드
7,1,E	7	1비트	짝수	A619.02 (시리얼 포트 설정 변경 플래그)
7,1,O			홀수	
7,1,N			없음	
7,2,E (디폴트)		2비트	짝수	
7,2,O			홀수	
7,2,N			없음	
8,1,E	8	1비트	짝수	
8,1,O			홀수	
8,1,N			없음	
8,2,E		2비트	짝수	
8,2,O			홀수	
8,2,N			없음	

-For Reference Only-

모드(Custom 설정 선택 시)

내장 RS-232C 포트의 시리얼 통신 모드를 선택합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
모드	· 호스트 링크 · NT 링크 (1:N) · RS-232C (무순서) · 페리페럴 버스 (Toolbus) · 시리얼 게이트웨이 · 시리얼 PLC 링크 (폴링 유닛)*3 · 시리얼 PLC 링크 (폴드 유닛)*3	호스트 링크*1	이 설정은 시리얼 포트를 호스트 링크 모드로 동작시킬지 다른 시리얼 통신 모드로 동작시킬지 정합니다.*2	A619.02 (시리얼 포트 설정 변경 플래그)

*1 호스트 링크(SYSMAC WAY)는 일반적인 호스트 컴퓨터와 접속하기 위한 통신 모드입니다.

*2 1:1 NT 링크로 설정된 PT와는 통신할 수 없습니다.

*3 CJ2H-CPU6□(-EIP)에서는 시리얼 PLC 링크를 선택할 수 없습니다.

● 선택 모드별 관련 설정

모드	관련 설정
호스트 링크 (디폴트)	보 레이트, 파라미터, 유닛 번호
NT 링크 (1:N)	보 레이트, NT/PC link max.
RS-232C (무순서)	보 레이트, 파라미터, 스타트 코드, 엔드 코드, 지연
페리페럴 버스 (Tool Bus)	보 레이트
시리얼 게이트웨이	보 레이트, 파라미터, 응답 감시 시간
시리얼 PLC 링크 (폴링 유닛)	통신 설정, 모드, 링크 워드 수, 링크 방식, NT 링크 또는 시리얼 PLC 링크의 최대 유닛 번호 (No. NT/PC Link Max.)
시리얼 PLC 링크 (폴드 유닛)	통신 설정, 모드, 시리얼 PLC 링크 폴드 유닛의 유닛 번호

● 호스트 링크 설정

컴퓨터 등의 호스트 기기와 호스트 링크로 통신하는 경우, 그리고 9,600 bps 이외의 보 레이트를 사용하는 경우에는 호스트 링크 모드를 사용하십시오.

· 유닛 번호(호스트 링크 모드의 CPU 유닛용)

PLC 슬레이브의 유닛 번호를 설정합니다.

AI 번역본 For Reference Only-

파라미터	설정	디폴트	기능	관련 플래그 및 워드	CPU 유닛에서의 갱신 타이밍
유닛 번호	0~31	0	이 설정은 1:N(N=2~32) 호스트 링크로 접속했을 때의 CPU 유닛의 유닛 번호를 정합니다.	A619.02 (시리얼 포트 설정 변경 플래그)	매 사이클

● NT 링크 설정

오므론 프로그래머블 터미널(NS 시리즈 또는 NT 시리즈)과 통신하는 경우에는 NT 링크 모드를 사용하십시오.

보 레이트(bps)

파라미터	설정	디폴트	기능	관련 플래그 및 워드
보 레이트	115,200, 38,400 bps	115,200 bps	CX-Programmer에서 이 값을 설정할 때는 115,200 bps로 설정할 것을 권장합니다.	A619.02 (시리얼 포트 설정 변경 플래그)

NT/PC Link Max.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
NT/PC Link Max.	0~7	1	이 설정은 PLC에 접속할 수 있는 PT의 최대 유닛 번호를 정합니다.	A619.02 (시리얼 포트 설정 변경 플래그)

● RS-232C (무순서) 설정

바코드 리더 등의 기기와 무순서 통신을 수행하는 경우에는 무순서 모드를 사용하십시오.

스타트 코드

파라미터	설정	디폴트	기능	관련 플래그 및 워드
스타트 코드	무효 또는 설정	무효	무순서 통신으로 송수신하는 데이터의 프레임 포맷에 스타트 코드를 사용할지 여부를 설정합니다.	---
	0x0000~00FF (0000~00FF hex: "0x"는 hex와 같은 뜻입니다.)	0x0000 (00 hex)	스타트 코드를 16진수 값으로 설정합니다.	

엔드 코드

파라미터	설정	디폴트	기능	관련 플래그 및 워드
엔드 코드	· 수신 바이트 수(엔드 코드 없음) · CR LF · 엔드 코드 설정	수신 바이트 수	무순서 통신으로 송수신하는 데이터의 프레임 포맷에 사용할 엔드 코드의 종별을 설정합니다	---
수신 바이트 수	256 또는 1~255	256	엔드 코드로 수신 바이트 수를 선택한 경우, 무순서 통신으로 송수신하는 데이터 길이가 바이트 단위로 설정됩니다. 엔드 코드와 스타트 코드의 길이는 데이터 길이에 포함되지 않습니다. 하나의 TXD(236)/RXD(235) 명령어로 송신 또는 수신할 수 있는 최대 데이터 길이는 디폴트로 256 바이트입니다.	
엔드 코드 설정	0x0000~00FF (0000~00FF hex: "0x"는 hex와 같은 뜻입니다.)	0x00 (00 hex)	엔드 코드로 "엔드 코드 설정"을 지정한 경우, 여기에 16진수 값으로 설정합니다.	

지연

파라미터	설정	디폴트	기능	관련 플래그 및 워드
지연	0~99990 ms (10 ms 단위)	0 ms	TXD(236) 명령어가 실행되면 여기서 설정한 송신 지연이 경과한 후에 시리얼 포트에서 데이터가 송신됩니다.	---

AI 번역본 -For Reference Only-

● 페리페럴 버스 (Toolbus) 설정

CX-Programmer 등의 프로그래밍 디바이스와 통신하는 경우에는 페리페럴 버스 모드를 선택하십시오.

보 레이트

파라미터	설정	디폴트	기능	관련 플래그 및 워드
보 레이트	9600, 19200, 38400, 57600, 115200 bps	115200 bps	페리페럴 버스(툴 버스)를 지정합니다.	A619.02 (시리얼 포트 설정 변경 플래그)

CPU 유닛의 딥 스위치 핀 5를 ON으로 설정하면 CX-Programmer에서 보 레이트를 자동으로 검출하여 페리페럴 버스로 접속할 수 있습니다.

● 시리얼 게이트웨이 설정

CompoWay/F를 사용하여 오므론 부품과 통신하는 경우에는 시리얼 게이트웨이 프로토콜을 선택합니다.

응답 감시 시간

파라미터	설정	디폴트	기능*1	관련 플래그 및 워드
응답 감시 시간	5000 ms, 100~25500 ms (100 ms 단위)	0 (50000 ms)	시리얼 게이트웨이로 지정한 프로토콜로 변환된 FINS 커맨드를 송신한 시점부터 응답을 수신할 때까지의 시간을 감시합니다. 디폴트: 5 s, PLC 설정: 0.1~25.5 s	A619.02 (시리얼 포트 설정 변경 플래그)

*1 타임아웃이 발생하면 FINS 송신원에 FINS 엔드 코드 0205 hex(응답 타임아웃)가 반환됩니다.

● 시리얼 PLC 링크 폴링 유닛 설정

시리얼 PLC 링크 폴링 유닛을 선택하면 특별한 프로그래밍 없이 CJ3M/CJ2M/CJ1M/CP1H/CP1L/CP1E/CP2E CPU 유닛 사이에서 데이터를 교환할 수 있습니다.
(CJ2H CPU 유닛은 사용할 수 없습니다.)

통신 설정

파라미터	설정	디폴트	기능	관련 플래그 및 워드
보 레이트	38400, 115200 bps	115200 bps	시리얼 PLC 링크 폴링 유닛을 지정할 때의 보 레이트를 선택합니다.	A619.02 (시리얼 포트 설정 변경 플래그)

링크 워드 수

파라미터	설정	디폴트	기능	관련 플래그 및 워드
링크 워드	1~10워드	10워드	이 파라미터는 폴링 유닛에서만 설정합니다. 시리얼 PLC 링크 영역에서 노드당 사용하는 워드 수를 설정합니다.	A619.02 (시리얼 포트 설정 변경 플래그)

AI 번역본

PLC 링크 방식

파라미터	설정	디폴트	기능	관련 플래그 및 워드
링크 방식	완전 링크 방식 또는 폴링 유닛 링크 방식	완전 링크	이 설정은 시리얼 PLC 링크의 링크 방식을 지정합니다. 이 파라미터는 폴링 유닛에서만 설정합니다.	A619.02 (시리얼 포트 설정 변경 플래그)

NT/시리얼 PLC 링크의 최대 유닛 번호

파라미터	설정	디폴트	기능	관련 플래그 및 워드
No. NT/PC Link Max.	0~7	0	이 설정은 시리얼 PLC 링크에서 1:N 접속을 사용할 때 폴링 유닛에 접속된 폴드 유닛의 최대 유닛 번호를 정합니다. 이 파라미터는 폴링 유닛에서만 설정합니다.	A619.02 (시리얼 포트 설정 변경 플래그)

● 시리얼 PLC 링크 폴드 유닛 설정

시리얼 PLC 링크 폴드 유닛을 선택하면 특별한 프로그래밍 없이 CJ3M/CJ2M/CJ1M/CP1H/CP1L/CP1E/CP2E CPU 유닛 사이에서 데이터를 교환할 수 있습니다.
(CJ2H CPU 유닛은 사용할 수 없습니다.)

통신 설정

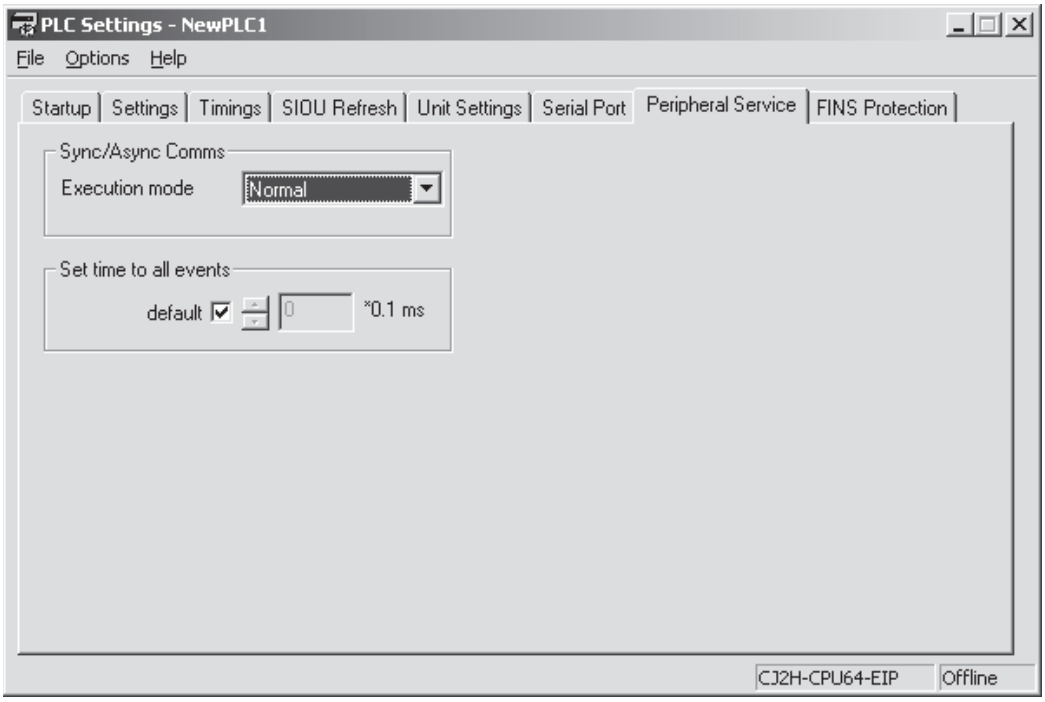
파라미터	설정	디폴트	기능	관련 플래그 및 워드
보 레이트	38400, 115200 bps	115200 bps	시리얼 PLC 링크 폴드 유닛을 지정할 때의 보 레이트를 선택합니다.	A619.02 (시리얼 포트 설정 변경 플래그)

시리얼 PLC 링크 폴드 유닛의 유닛 번호

파라미터	설정	디폴트	기능	관련 플래그 및 워드
시리얼 PLC 링크 유닛 번호	0~7	0	시리얼 PLC 링크에서 1:N 접속을 사용할 때 폴링 유닛에 접속된 폴드 유닛의 유닛 번호를 설정합니다.	A619.02 (시리얼 포트 설정 변경 플래그)

AI 번역본
-For Reference Only-

9-3-7 주변 서비스



● 동기/비동기 통신(CPU 처리 모드)

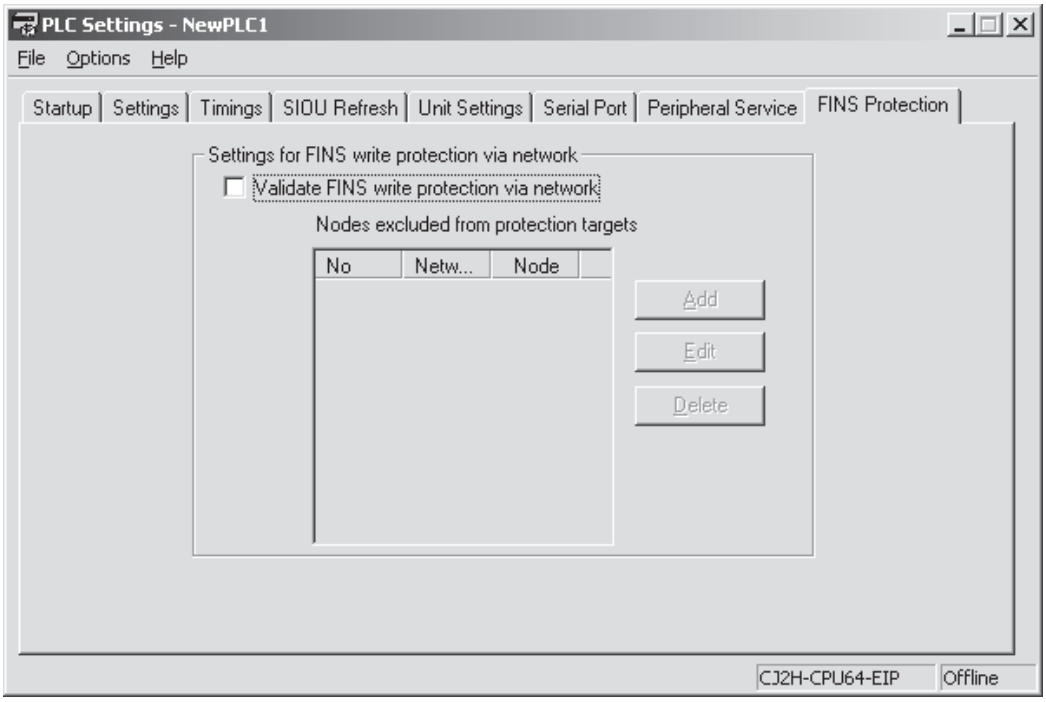
파라미터	설정	디폴트	기능	관련 플래그 및 워드
실행 모드	통상 모드	통상 모드	CJ2/CJ3 CPU 유닛은 통상 모드만 지원합니다.	---

● 모든 이벤트에 대한 시간 설정

파라미터	설정	디폴트	기능	관련 플래그 및 워드
모든 이벤트에 대한 시간 설정	디폴트: 사이클 타임의 10% 0.1~3,276.7 ms	사이클 타임의 10% (사이클 타임의 10%가 0.1 ms 미만인 경우에는 0.1 ms)	전체 주변 서비스에 사용되는 최대 시간을 설정합니다. 최대 시간은 0.1 ms 이상으로 설정할 수 있습니다.	---

AI 번역본
-For Reference Only-

9-3-8 FINS 보호



네트워크를 경유한 FINS 쓰기 보호 설정

● 네트워크를 경유한 FINS 쓰기 보호 유효화

이 파라미터는 네트워크를 경유한 FINS 커맨드와 CIP 메시지 통신에 대해 쓰기 보호를 사용할지 여부를 정합니다.

이 파라미터는 시리얼 접속 이외의 접속으로 연결된 네트워크를 통해 메모리에 잘못 쓰는 것을 방지하는 데 사용합니다. 이 파라미터를 유효하게 하면 보호 대상에서 제외하는 노드 파라미터에 설정한 노드에서만 쓰기가 가능해집니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
네트워크를 경유한 FINS 쓰기 보호 유효화	ON: FINS 쓰기 보호 무효 OFF: FINS 쓰기 보호 유효	OFF	네트워크(즉, 시리얼 접속을 제외한 모든 접속)를 통해 송신된 FINS 커맨드 또는 CIP 메시지 통신에 대한 CPU 유닛의 쓰기 보호를 유효 또는 무효로 합니다.	---

● 보호 대상에서 제외하는 노드

보호가 유효한 상태에서도 쓰기가 가능하도록 할 노드를 설정합니다.

보호가 유효한 상태에서도 FINS 커맨드에 의한 쓰기 조작이 가능하도록 할 노드와 네트워크를 설정합니다. 쓰기 보호 대상에서 제외하도록 설정한 노드의 총수는 자동으로 설정됩니다.

최대 32개의 노드를 설정할 수 있습니다. 이 설정을 하지 않으면(즉, 노드의 총수가 0이면) 자기 노드를 제외한 모든 노드에서 FINS 커맨드에 의한 쓰기 조작이 무효가 됩니다

다만 CIP 메시지 통신에 의한 쓰기 조작에 대해서는 보호 대상에서 제외하는 노드를 설정할 수 없습니다.

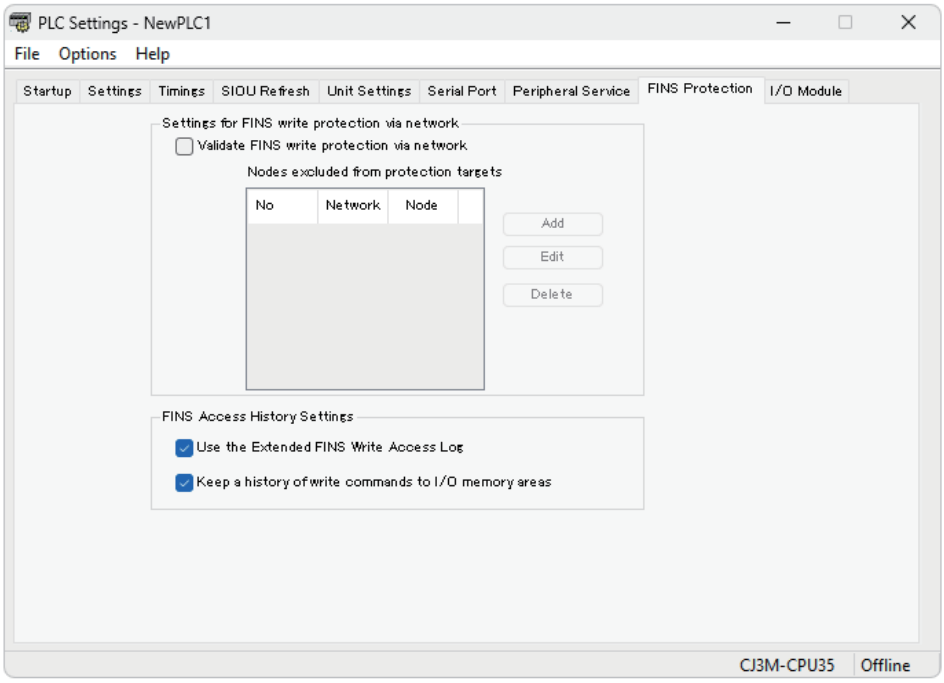
파라미터	설정	디폴트	기능	관련 플래그 및 워드
보호 대상에서 제외하는 노드*1	0~127	---	FINS 커맨드 송신원 네트워크 주소	---
	1~255*2	---	FINS 커맨드 송신원 노드 주소	---

*1 이 설정은 FINS 쓰기 보호가 유효한 경우에만 유효합니다.

*2 지정한 네트워크의 모든 노드를 포함하려면 255(FF hex)를 설정할 수 있습니다.

AI 번역본 For Reference Only

FINS 쓰기 액세스 이력 설정 (CJ3 CPU 유닛 전용)



● 확장 FINS 쓰기 액세스 로그 기능 사용

확장 FINS 쓰기 액세스 로그 기능을 사용할지 FINS 쓰기 액세스 로그 기능을 사용할지 설정합니다. FINS 쓰기 보호 기능이 유효한 경우에도 액세스 로그는 유지됩니다.

자세한 내용은 10-6-5 FINS 쓰기 액세스 로그를 참조하십시오.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
기존 기능과의 호환성 설정	· 확장 FINS 쓰기 액세스 로그 사용 · 기존 FINS 쓰기 액세스 로그 사용	확장 FINS 쓰기 액세스 로그 사용	확장 FINS 쓰기 액세스 로그를 사용할지 이전 FINS 쓰기 액세스 로그를 사용할지 설정합니다.	---

● I/O 메모리 영역에 대한 쓰기 커맨드 이력 유지

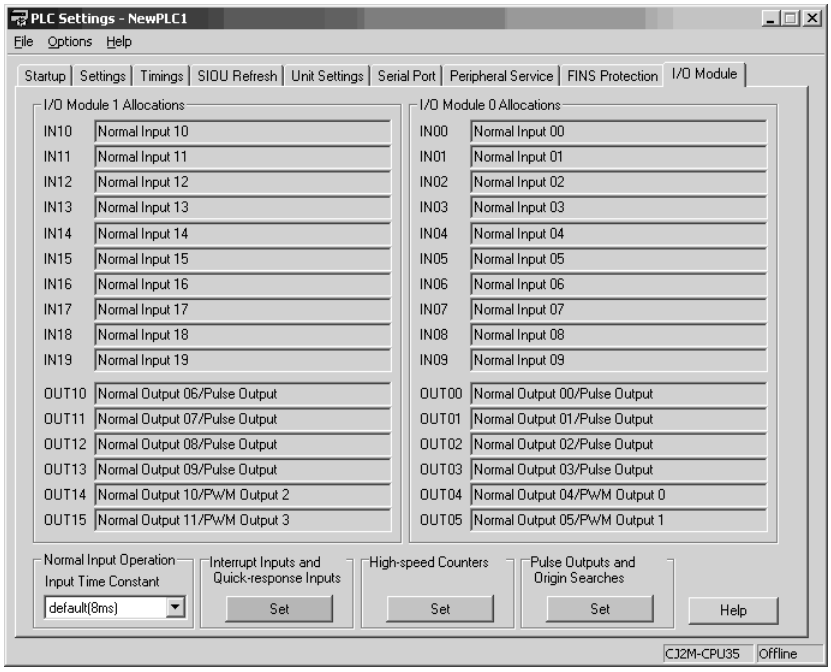
I/O 메모리 쓰기 커맨드를 사용할 때 FINS 쓰기 액세스 로그에 로그를 유지할지 여부를 설정합니다.
이 설정은 확장 FINS 쓰기 액세스 로그와 FINS 쓰기 액세스 로그 양쪽에 유효합니다.

파라미터	설정	디폴트	기능	관련 플래그 및 워드
I/O 메모리 영역 쓰기 커맨드 로그 등록 ON/OFF	· 로그 유지 · 로그 유지 안 함	로그 유지	I/O 메모리 영역에 대한 쓰기 로그를 유지할지 여부를 설정합니다.*	---

* 다음 FINS 커맨드 코드에서는 로그가 유지되지 않습니다.
  - · MEMORY AREA WRITE: 0102 hex
  - · MEMORY AREA FILL: 0103 hex
  - · MEMORY AREA TRANSFER: 0105 hex

9-3-9 I/O 모듈

펄스 I/O 모듈은 CJ2M/CJ3M CPU 유닛에서 사용할 수 있습니다. 자세한 내용은 CJ2M/CJ3M CPU 유닛 펄스 I/O 모듈 사용자 매뉴얼(Cat. No. W486)을 참조하십시오.



AI 번역본
-For Reference Only-

10

CPU 유닛의 기능

본 섹션에서는 CPU 유닛에 내장된 기능을 설명합니다.
CJ2M/CJ3M CPU 유닛에 CJ2M-MD21□ 펄스 I/O 모듈을 접속했을 때 사용할 수 있는 기능(고속 카운터, 펄스 출력 등)에 대한 자세한 내용은 CJ2M/CJ3M CPU 유닛 펄스 I/O 모듈 사용자 매뉴얼(Cat. No. W486)을 참조하십시오.

10-1 클록 기능 10-3
10-1-1 클록 기능 10-3
10-1-2 메모리에 저장되는 시각 10-4
10-1-3 프리런 타이머 10-7

10-2 사이클 타임/고속 처리 10-8
10-2-1 최소 사이클 타임 10-8
10-2-2 최대 사이클 타임 10-9
10-2-3 사이클 타임 모니터 10-10
10-2-4 고속 입력 10-10
10-2-5 백그라운드 실행 10-11
10-2-6 고속 인터럽트 기능 10-20

10-3 기동 설정과 메인터넌스 10-23
10-3-1 동작 모드 변경 시 및 기동 시의 유지 설정 10-23
10-3-2 전원 OFF 검출 지연 설정 10-25
10-3-3 전원 OFF 인터럽트 무효화 10-26
10-3-4 RUN 출력 10-27
10-3-5 기동 시 자동 전송 10-28

10-4 유닛 관리 기능 10-37
10-4-1 기본 I/O 유닛 관리 10-37
10-4-2 CPU 버스 유닛 플래그/비트 10-39
10-4-3 특수 I/O 유닛 플래그/비트 10-40

10-5 메모리 관리 기능 10-41
10-5-1 자동 백업 10-41
10-5-2 EM 파일 메모리 기능 10-43
10-5-3 코멘트 메모리 10-44
10-5-4 운전 중 전체 프로그램 교체 10-45
10-5-5 전 메모리 클리어 10-52

사본 견본
-For Reference Only-

10-6 보안 기능 10-53
10-6-1 딥 스위치에 의한 쓰기 보호 10-53
10-6-2 패스워드에 의한 읽기 보호 10-53
10-6-3 제조 로트 번호에 의한 프로그램 동작 보호 10-59
10-6-4 FINS 커맨드에 의한 쓰기 보호 10-60
10-6-5 FINS 쓰기 액세스 로그 10-64
10-6-6 조작 로그 기능 10-68
10-6-7 파일 암호화 기능 10-71
10-6-8 PLC 이름 10-77
10-7 디버깅 10-80
10-7-1 강제 세트/리셋 10-80
10-7-2 테스트 입력 10-81
10-7-3 미분 모니터 10-81
10-7-4 온라인 에디팅 10-82
10-7-5 출력 OFF 10-84
10-7-6 데이터 트레이스 10-85
10-7-7 에러 시 정지 위치 저장 10-93
10-7-8 고장 알람 명령어 10-94
10-7-9 시스템 에러 시뮬레이션 10-95
10-7-10 고장점 검출 10-96
10-8 동기 유닛 동작 10-98
10-8-1 개요 10-98
10-8-2 동기 유닛 동작 상세 10-102
10-8-3 동기 유닛 동작 사양 10-106
10-8-4 동기 데이터 리프레시 10-107
10-8-5 동기 유닛 동작 사용 시의 제한 10-111
10-8-6 적용 순서 10-113
10-8-7 PLC 설정 10-114
10-8-8 동기 인터럽트 태스크 작성 10-116
10-8-9 동기 유닛 동작의 조정과 문제 해결 10-117

AI 번역본
-For Reference Only-

10-1 클록 기능

10-1-1 클록 기능

클록 데이터

CJ2/CJ3 CPU 유닛에는 클록이 내장되어 있습니다.
CPU 유닛의 클록에서 얻은 클록 데이터는 보조 기억 영역의 다음 비트에 BCD로 저장됩니다.

명칭	어드레스	설명
클록 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

・배터리가 접속되어 있지 않거나 배터리 전압이 낮을 때의 클록 동작
아래 표는 배터리가 접속되어 있지 않거나 배터리 전압이 낮을 때의 클록 동작을 정리한 것입니다.

배터리 상태	클록 동작
전원을 ON했을 때 다음 중 어느 하나인 경우 ・배터리가 접속되어 있지 않음 ・접속된 배터리의 전압이 낮음	・클록이 정지한 경우(내장 커패시터가 방전된 경우)* 클록이 2000년 1월 1일 0시 0분 0초부터 시작합니다 ・클록이 정지하지 않은 경우(내장 커패시터로 클록 데이터가 유지된 경우)* 클록이 계속됩니다
전원을 ON한 후 다음 중 어느 하나의 상태인 경우 ・배터리를 분리함 ・접속된 배터리의 전압이 낮음	클록이 계속됩니다

* 전원이 OFF일 때 클록 데이터는 아래와 같이 내장 커패시터에 의해 유지됩니다.
CJ2 CPU 유닛: 최소 5분
CJ3 CPU 유닛: 40°C(주위 온도)에서 10일

CJ3 CPU 유닛에서는 클록 발진의 정지를 검출할 수 있습니다.

명칭	어드레스	기능
클록 정지 플래그	A10008.01	클록이 한 번이라도 정지하면 1(ON)로 바뀝니다. 클록 데이터를 설정하면 플래그가 클리어(OFF)됩니다. 플래그의 1(ON) 상태는 클록 데이터가 설정될 때까지 유지됩니다. 0: 클록 정지 없음 1: 클록 정지 있음

사용상의 주의

・배터리가 접속되어 있지 않거나 배터리 전압이 낮으면 A351~A354 워드의 클록 데이터가 불안정해집니다. 정상적인 배터리가 접속되어 있지 않으면 클록 데이터를 사용하지 마십시오.

・표준 사양에서는 CJ3 CPU 유닛에 배터리가 접속되어 있지 않습니다. 클록 기능을 사용하려면 배터리(옵션 CJ1W-BAT01)를 접속하십시오. 배터리를 접속한 후 PLC 설정에서 "검출"을 ON으로 설정하십시오. 디폴트 설정은 "검출하지 않음"입니다.

AI 번역본

클록 데이터 설정

이 데이터는 CX-Programmer를 온라인으로 접속하고 프로젝트 트리에서 PLC Clock 아이콘을 더블 클릭하거나 DATE(735) 명령어를 실행하여 설정할 수 있습니다. 또한 클록 데이터는 가산 · 감산하거나 초와 시간 사이에서 변환할 수 있습니다.

● 클록 명령어

명령어	니모닉	기능
클록 조정	DATE(735)	내부 클록 설정을 지정한 소스 워드의 설정값으로 변경합니다.
시분초 → 초 변환	SEC(065)	시/분/초 형식의 시간 데이터를 초 단위만의 시간으로 변환합니다.
초 → 시분초 변환	HMS(066)	초 데이터를 시/분/초 형식의 시간으로 변환합니다.
캘린더 가산	CADD(730)	지정한 워드의 캘린더 데이터에 시간을 가산합니다.
캘린더 감산	CSUB(731)	지정한 워드의 캘린더 데이터에서 시간을 감산합니다.

10-1-2 메모리에 저장되는 시각

보조 기억 영역의 워드에는 전원이 ON된 시각, 전원이 OFF된 시각, 정전 횟수, 전원 ON 총 시간, 사용자 메모리(프로그램과 파라미터)가 쓰인 시각, 운전이 개시 · 정지된 시각이 자동으로 저장됩니다. 배터리가 접속되어 있지 않거나 배터리 전압이 낮은 경우에는 메모리에 저장된 시각을 사용하지 마십시오.

전원 ON 클록 데이터

PLC가 ON된 연, 월, 일, 시각은 다음 보조 기억 영역 워드에 저장됩니다.

명칭	어드레스	설명
전원 ON 클록 데이터 1	A720~ A722	데이터는 BCD입니다. 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 클록 데이터 2	A723~ A725	이 워드에는 2회 전에 전원이 ON되었을 때의 기동 일시가 저장됩니다. 데이터는 BCD이며 저장 형식은 A720~A722 워드와 같습니다.
전원 ON 클록 데이터 3	A726~ A728	이 워드에는 3회 전에 전원이 ON되었을 때의 기동 일시가 저장됩니다. 데이터는 BCD이며 저장 형식은 A720~A722 워드와 같습니다.
전원 ON 클록 데이터 4	A729~ A731	이 워드에는 4회 전에 전원이 ON되었을 때의 기동 일시가 저장됩니다. 데이터는 BCD이며 저장 형식은 A720~A722 워드와 같습니다.
전원 ON 클록 데이터 5	A732~ A734	이 워드에는 5회 전에 전원이 ON되었을 때의 기동 일시가 저장됩니다. 데이터는 BCD이며 저장 형식은 A720~A722 워드와 같습니다.
전원 ON 클록 데이터 6	A735~ A737	이 워드에는 6회 전에 전원이 ON되었을 때의 기동 일시가 저장됩니다. 데이터는 BCD이며 저장 형식은 A720~A722 워드와 같습니다.

For Reference Only

이름	어드레스	설명
전원 ON 클록 데이터 7	A738~ A740	전원이 ON된 횟수 중 7회 전의 기동 시각/날짜가 저장됩니다. 데이터는 BCD이며 저장 형식은 워드 A720~A722와 동일합니다.
전원 ON 클록 데이터 8	A741~ A743	전원이 ON된 횟수 중 8회 전의 기동 시각/날짜가 저장됩니다. 데이터는 BCD이며 저장 형식은 워드 A720~A722와 동일합니다.
전원 ON 클록 데이터 9	A744~ A746	전원이 ON된 횟수 중 9회 전의 기동 시각/날짜가 저장됩니다. 데이터는 BCD이며 저장 형식은 워드 A720~A722와 동일합니다.
전원 ON 클록 데이터 10	A747~ A749	전원이 ON된 횟수 중 10회 전의 기동 시각/날짜가 저장됩니다. 데이터는 BCD이며 저장 형식은 워드 A720~A722와 동일합니다.

정전 시각

PLC가 OFF된 날짜와 시각은 다음 보조 기억 영역 워드에 저장됩니다. 정전 횟수도 다음 보조 기억 영역 워드에 저장됩니다.

이름	어드레스	설명
정전 시각	A512 및 A513	전원이 차단된 시각이 BCD로 저장됩니다. 내용은 전원이 차단될 때마다 갱신됩니다. 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)
정전 발생 횟수	A514	전원을 처음 ON한 이후 전원이 차단된 횟수가 바이너리로 저장됩니다. 이 값을 리셋하려면 현재값을 0000으로 덮어쓰십시오.

총 전원 ON 시간

PLC가 ON되어 있던 총 시간은 다음 보조 기억 영역 워드에 저장됩니다.

이름	어드레스	설명
총 전원 ON 시간	A523	PLC가 ON되어 있던 총 시간이 10시간 단위로 16비트 바이너리로 저장됩니다. 이 값을 리셋하려면 현재값을 0000 hex로 덮어쓰십시오.

AI 번역본
-For Reference Only-

사용자 프로그램 및 파라미터 변경 시각

이 보조 기억 영역 워드에는 사용자 프로그램 또는 파라미터(즉, PLC 설정, I/O 테이블, 라우팅 테이블, CPU 버스 유닛 셋업)에 데이터를 쓴 날짜와 시각이 저장됩니다.

이름	어드레스	설명
사용자 프로그램 날짜	A090~ A093	사용자 프로그램이 마지막으로 덮어쓰기된 날짜와 시각이 BCD로 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 날짜	A094~ A097	파라미터가 마지막으로 덮어쓰기된 날짜와 시각이 BCD로 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 개시/종료 시각

이 보조 기억 영역 워드에는 PLC 운전이 개시된 날짜 · 시각과 정지된 날짜 · 시각이 자동으로 저장됩니다.

이름	어드레스	설명
운전 개시 시각	A515~ A517	동작 모드를 RUN 또는 MONITOR 모드로 변경하여 운전이 개시된 시각이 BCD로 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 종료 시각	A518~ A520	동작 모드를 PROGRAM 모드로 변경하여 운전이 정지된 시각이 BCD로 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 ON한 후 운전이 개시될 때까지는 이전 개시 시각이 저장됩니다.

*2 운전 중에 에러가 발생하면 그 에러의 발생 시각이 저장됩니다. 그 후 동작 모드를 PROGRAM 모드로 변경하면 PROGRAM 모드로 전환된 시각이 저장됩니다.

사본
-For Reference Only-

10-1-3 프리런 타이머

전원을 ON한 후 사용되는 시스템 타이머의 시간이 보조 기억 영역에 저장됩니다.
이 타이머를 사용하면 타이머 명령어를 쓰지 않고 시간 간격을 계산할 수 있습니다. 이 타이머를 사용하면 타이머 명령어를 쓰지 않고 시간 간격을 계산할 수 있습니다.

이름	어드레스	설명
10 ms 증가 프리런 타이머	A0	전원이 ON되면 0000 hex가 설정되고 이 값은 10 ms마다 자동으로 1씩 증가합니다. FFFF hex(655,350 ms)에 도달하면 값은 0000 hex로 돌아가며, 그 후에도 10 ms마다 자동으로 1씩 계속 증가합니다.
100 ms 증가 프리런 타이머	A1	전원이 ON되면 0000 hex가 설정되고 이 값은 100 ms마다 자동으로 1씩 증가합니다. FFFF hex(6,553,500 ms)에 도달하면 값은 0000 hex로 돌아가며, 그 후에도 100 ms마다 자동으로 1씩 계속 증가합니다.
1 s 증가 프리런 타이머	A2	전원이 ON되면 0000 hex가 설정되고 이 값은 1초마다 자동으로 1씩 증가합니다. FFFF hex(65,535 s)에 도달하면 값은 0000 hex로 돌아가며, 그 후에도 1초마다 자동으로 1씩 계속 증가합니다.

주 동작 모드를 RUN 모드로 변경해도 1씩 자동 증가는 계속됩니다.

예: 처리 A와 처리 B 사이의 시간 간격은 타이머 명령어를 사용하지 않고 처리 시각 A일 때의 A0 값과 처리 시각 B일 때의 A0 값의 차를 계산하여 10 ms 단위로 구할 수 있습니다.

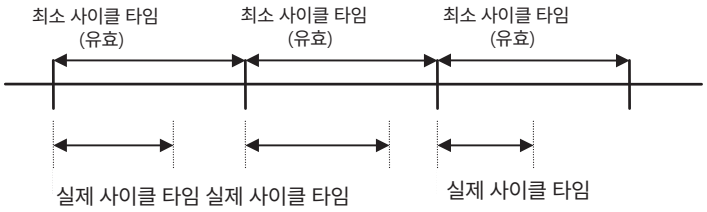
AI 번역본
-For Reference Only-

10-2 사이클 타임/고속 처리

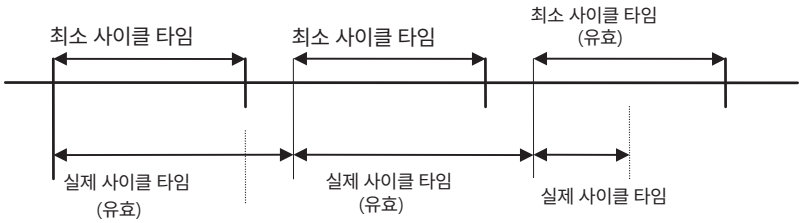
10-2-1 최소 사이클 타임

PLC에는 최소(고정) 사이클 타임을 설정할 수 있습니다. 일정한 사이클 타임으로 프로그램을 반복 실행하면 I/O 응답 시간의 편차를 없앨 수 있습니다.

최소 사이클 타임(0.1~32,000 ms)은 PLC 설정에서 0.1 ms 단위로 지정합니다.

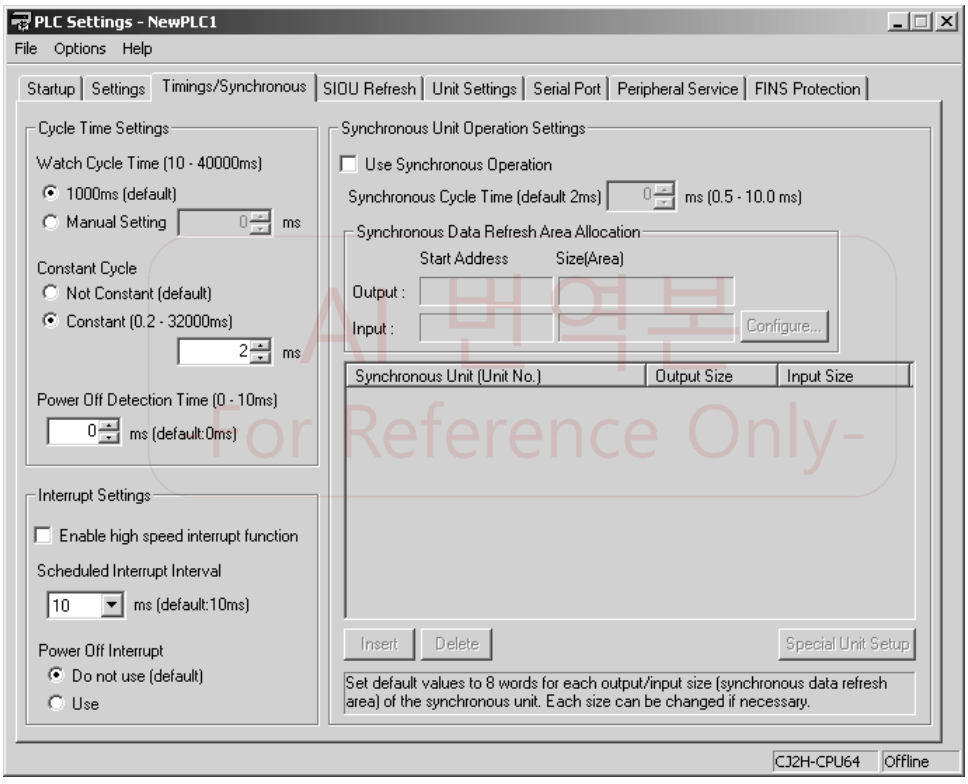


실제 사이클 타임이 최소 사이클 타임보다 길면 최소 사이클 타임 기능은 무효가 되고 사이클 타임은 사이클마다 변동합니다.



- **PLC 설정**

CX-Programmer를 사용하는 경우 Timings/Synchronous 탭 페이지에서 설정합니다.



AI 번역본
For Reference Only

참고

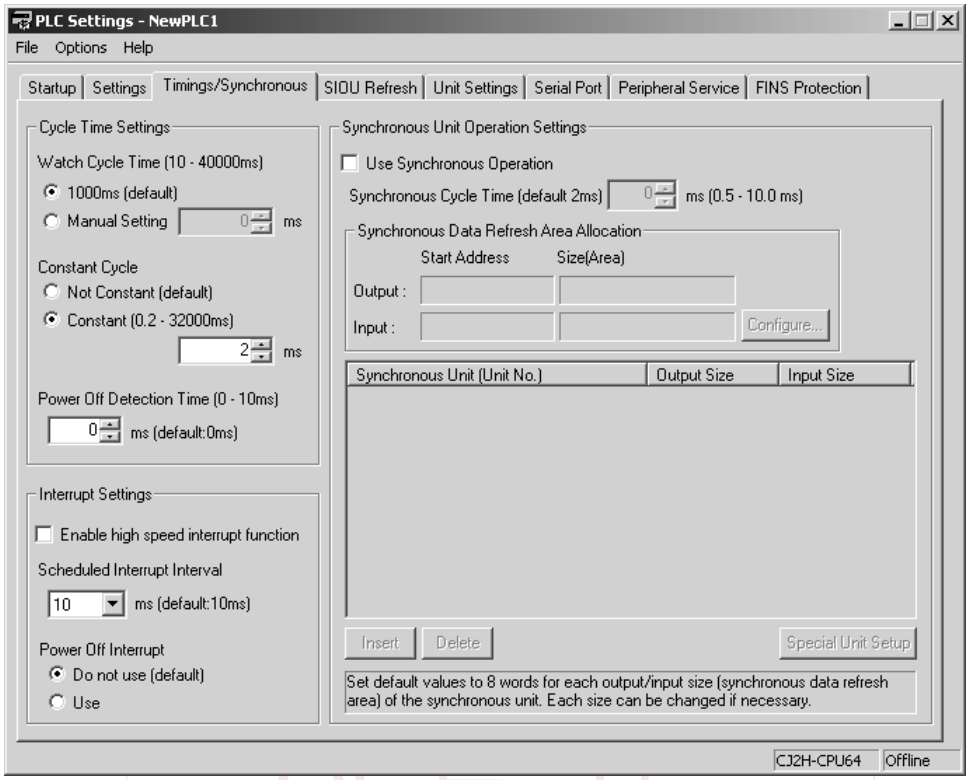
CPU 유닛이 MONITOR 모드로 동작 중일 때는 CX-Programmer의 PLC Cycle Time 대화 상자에서 최소 사이클 타임(일정 주기 사이클 타임)을 변경할 수 있습니다. (이 기능은 유닛 버전 1.1 이후의 CJ2H CPU 유닛과 CJ2M/CJ3M CPU 유닛에서만 지원됩니다.) 자세한 내용은 12-1 사이클 타임 모니터를 참조하십시오.

10-2-2 최대 사이클 타임

사이클 타임이 최대 사이클 타임 설정값을 초과하면 CPU 유닛은 운전을 정지합니다. 사이클 타임 오버 플래그(A401.08)가 ON됩니다. 최대 사이클 타임의 디폴트값은 1 s입니다.

● PLC 설정

CX-Programmer를 사용하는 경우 Timings/Synchronous 탭 페이지의 Watch Cycle Time 필드에서 최대 사이클 타임을 설정합니다.



● 보조 기억 영역 플래그와 워드

이름	어드레스	설명
사이클 타임 오버 플래그	A401.08	사이클 타임이 감시 사이클 타임 설정값을 초과하면 A401.08이 ON됩니다.

For Reference Only

10-2-3 사이클 타임 모니터

보조 기억 영역에는 사이클마다 최대 사이클 타임이 A262~A263에, 현재 사이클 타임이 A264~A265 및 A266~A267에 저장됩니다.

● 보조 기억 영역 플래그와 워드

이름	어드레스	설명
최대 사이클 타임 (0.1 ms 단위)	A262 및 A263	0.1 ms 단위의 최대 사이클 타임이 사이클마다 8자리 16진수로 다음 범위에서 저장됩니다: 0~429,496,729.5 ms (0~FFFF FFFF) 하위 자릿수는 A262에, 상위 자릿수는 A263에 저장됩니다.
현재 사이클 타임 (0.1 ms 단위)	A264 및 A265	0.1 ms 단위의 현재 사이클 타임이 사이클마다 8자리 16진수로 다음 범위에서 저장됩니다: 0~429,496,729.5 ms (0~FFFF FFFF) 하위 자릿수는 A264에, 상위 자릿수는 A265에 저장됩니다.
현재 사이클 타임 (0.01 ms 단위)	A266 및 A267	0.01 ms 단위의 현재 사이클 타임 값이 사이클마다 8자리 16진수로 다음 범위에서 저장됩니다: 0~42,949,672.95 ms (0~FFFF FFFF) 하위 자릿수는 A266에, 상위 자릿수는 A267에 저장됩니다.

CX-Programmer를 사용하면 최근 8사이클의 평균 사이클 타임을 읽을 수 있습니다.

참고

CJ 시리즈 PLC에서 사이클 타임을 단축하는 데에는 다음 방법이 효과적입니다:

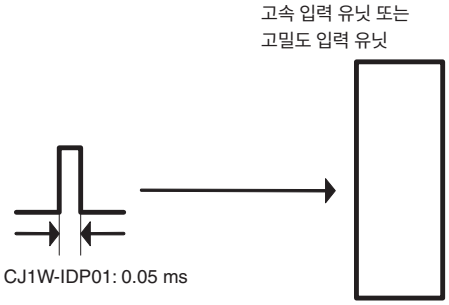
- 실행하지 않는 태스크를 WAIT 상태로 둡니다.
- JMP(004), CJP (510) 또는 CJPN (511)을 JME(005)와 함께 사용하여 실행할 필요가 없는 프로그램 부분을 점프합니다.

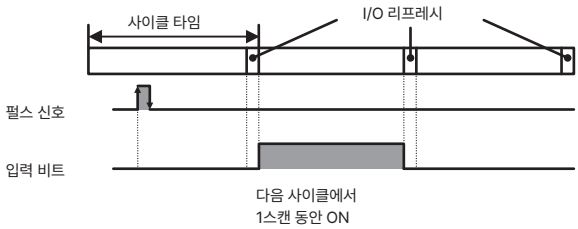
10-2-4 고속 입력

사이클 타임보다 짧은 펄스를 받으려면 CJ1W-IDP01 고속 입력 유닛을 사용하십시오.

CJ1W-IDP01 고속 입력 유닛의 고속 입력은 펄스 폭(ON 시간)이 0.05 ms인 펄스를 받을 수 있습니다.

AI 번역본





- 펄스 폭(ON 시간)이 0.05 ms 이상이면 펄스 신호를 받습니다.
- 1사이클 내에 여러 개의 펄스 신호가 입력되어도 1개의 입력만 받습니다.
- 입력을 받으면 입력 비트는 펄스 신호 다음 사이클에서 1스캔 동안 ON되고, I/O 리프레시 시에 자동으로 OFF됩니다.

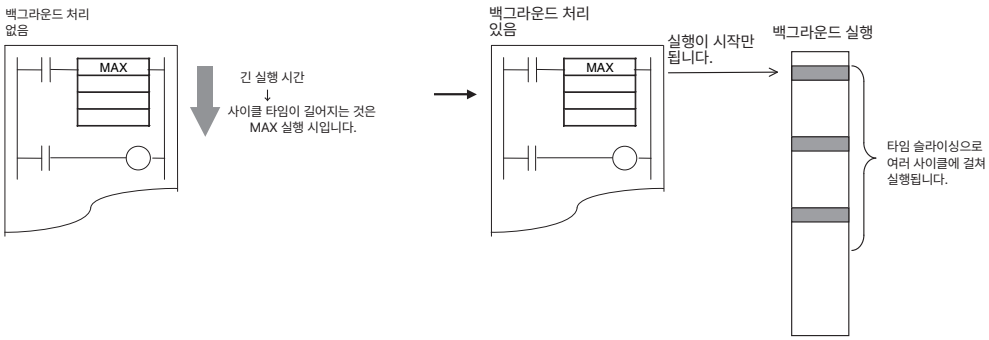
10-2-5 백그라운드 실행

테이블 데이터 처리(데이터 검색 등)와 문자열 처리(문자열 검색 등)는 실행에 시간이 걸리며, 실행에 필요한 시간이 길어지기 때문에 사이클 타임에 큰 변동을 일으킬 수 있습니다.

백그라운드 실행(타임 슬라이싱)을 사용하면 다음 명령어를 여러 사이클에 걸쳐 실행하여 사이클 타임의 변동을 억제할 수 있습니다. PLC 설정에서 명령어 종류별로 백그라운드 실행을 설정할 수 있습니다.

・테이블 데이터 처리 명령어
・문자열 처리 명령어
・데이터 시프트 명령어(비동기 시프트 레지스터만)

위 명령어에 백그라운드 실행을 설정하면 사이클 타임의 일시적인 증가를 억제할 수 있습니다.



사용상의 주의

CJ2H CPU 유닛에서 PLC 설정으로 고속 인터럽트를 유효하게 한 경우, 인터럽트 태스크에서는 백그라운드 처리가 수행되지 않습니다. 명령어 처리 에러가 발생합니다.

AI 번역본
-For Reference Only-

● 적용 가능한 명령어

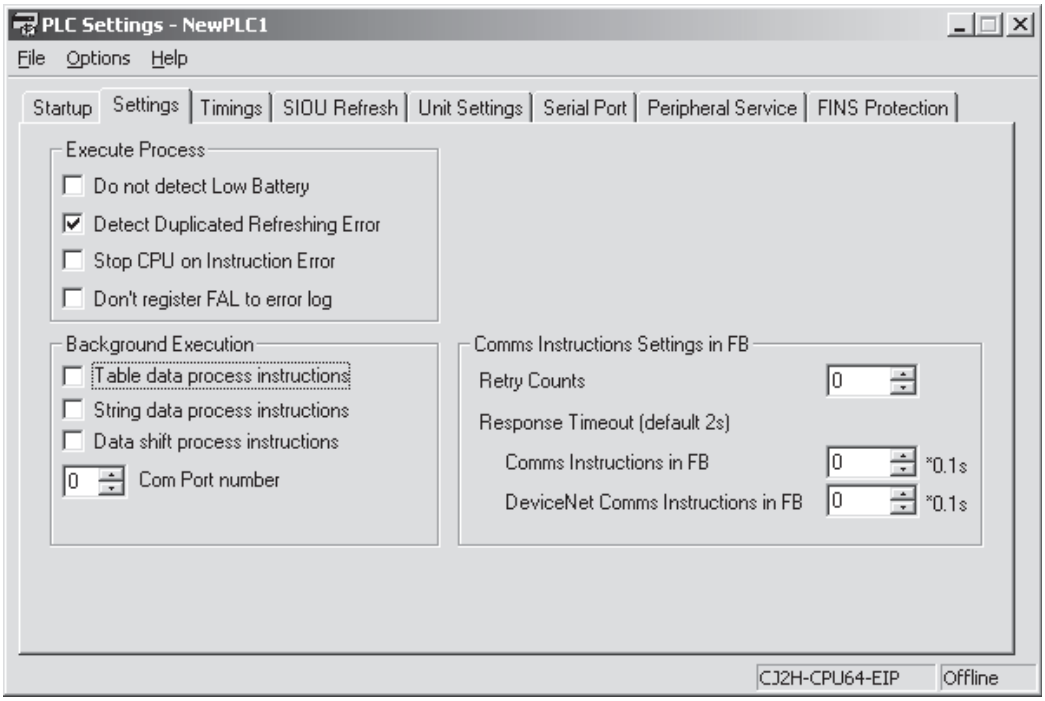
다음 명령어를 펑션 블록에서 사용하는 경우에는 백그라운드 처리가 수행되지 않습니다. 통상 처리로 실행됩니다.

그룹	명령어	니모닉
테이블 데이터 처리 명령어	데이터 검색	SRCH
	바이트 교체	SWAP
	최대값 검색 명령어	MAX
		MAXL
		MAXF
		MAXD
	최소값 검색 명령어	MIN
		MINL
		MINF
		MIND
	합계	SUM
	프레임 체크섬	FCS
데이터 시프트 명령어	비동기 시프트 레지스터	ASFT
문자열 처리 명령어	문자열 전송	MOV$
	문자열 결합	+$
	문자열 좌측 취득	LEFT$
	문자열 우측 취득	RGHT$
	문자열 중간 취득	MID$
	문자열 내 검색	FIND$
	문자열 길이	LEN$
	문자열 치환	RPLC$
	문자열 삭제	DEL$
	문자열 교환	XCHG$
	문자열 클리어	CLR$
	문자열 삽입	INS$

AI 번역본
-For Reference Only-

● **순서**

1 필요한 명령어에 대해 백그라운드 실행을 유효하게 하려면 CX-Programmer에서 PLC Settings 대화 상자의 Settings 탭 페이지에서 백그라운드 실행을 설정합니다.



2 PLC 설정에서 백그라운드 실행에 사용할 논리 포트 번호를 설정합니다. 이 포트 번호는 백그라운드에서 처리되는 모든 명령어에 사용됩니다.
모든 백그라운드 실행에 하나의 포트가 사용됩니다. 따라서 다른 명령어가 이미 백그라운드 실행 중이면 그 명령어의 백그라운드 실행은 시작할 수 없습니다. 통신 포트 사용 가능 플래그를 사용하여 백그라운드 실행으로 지정한 명령어를 제어함으로써 동시에 2개 이상의 명령어가 실행되지 않도록 하십시오.

3 백그라운드 실행이 지정된 명령어를 실행하면, 실행 조건이 성립한 사이클에서 실행이 시작될 뿐이며 같은 사이클 안에서 실행이 완료되지는 않습니다.

4 백그라운드 실행이 시작되면 해당 포트의 통신 포트 사용 가능 플래그가 OFF됩니다.

AI 번역본
-For Reference Only-

5 백그라운드 실행은 여러 사이클에 걸쳐 계속됩니다.

6 처리가 완료되면 해당 포트의 통신 포트 사용 가능 플래그가 ON됩니다. 이로써 다른 명령어를 백그라운드에서 실행할 수 있게 됩니다.

● 통상 실행되는 명령어와 백그라운드에서 실행되는 명령어의 차이

통상적인 명령어 실행과 백그라운드 실행의 차이는 아래에 나열되어 있습니다.

인덱스 레지스터(IR)로의 출력

MAX(182), MAXL(174), MAXF(176), MAXD(178), MIN(183), MINL(175), MINF(177) 또는 MIND(179)를 실행하여 최소값 또는 최대값이 들어 있는 워드의 I/O 메모리 맵 어드레스를 인덱스 레지스터로 출력하는 경우, 그 어드레스는 인덱스 레지스터로 출력되지 않고 대신 A595 및 A596에 출력됩니다.

통상 실행과 동일하게 어드레스를 인덱스 레지스터에 저장하려면 데이터 전송 명령어 (예: MOVL(498))로 A595 및 A596의 어드레스를 인덱스 레지스터에 복사하십시오.

조건 플래그

백그라운드에서 처리되는 명령어를 실행한 후에는 조건 플래그가 갱신되지 않습니다. 조건 플래그의 상태를 액세스하려면 다음 예와 같이 조건 플래그에 같은 방식으로 영향을 주는 명령어를 실행한 다음 조건 플래그를 액세스하십시오.

인덱스 레지스터 IR00으로의 출력

SRCH(181)을 실행하여 일치하는 값이 들어 있는 워드(여러 개인 경우 선두 워드)의 I/O 메모리 맵 어드레스를 인덱스 레지스터로 출력하는 경우, 그 어드레스는 인덱스 레지스터로 출력되지 않고 대신 A595 및 A596에 출력됩니다.

SRCH(181)의 데이터 레지스터로의 출력

SRCH(181)을 실행하여 일치하는 데이터를 데이터 레지스터로 출력하는 경우, 그 데이터는 데이터 레지스터로 출력되지 않고 대신 A597에 출력됩니다.

일치하는 문자열

SRCH(181)이 일치하는 데이터를 찾은 경우, 이퀄 플래그가 ON되지 않고 대신 A598.01이 ON됩니다.

명령어 에러

백그라운드에서 처리 중인 명령어에 명령어 실행 에러 또는 불법 액세스 에러가 발생한 경우, ER 또는 AER 플래그는 ON되지 않고 대신 A395.10이 ON됩니다.
A395.10은 다음에 명령어가 백그라운드에서 처리될 때까지 ON 상태를 유지합니다.

MAX(182) 또는 MIN(183)의 데이터 레지스터로의 출력

MAX(182), MAXL(174), MAXF(176), MAXD(178), MIN(183), MINL(175), MINF(177) 또는 MIND(179)를 최소값 또는 최대값의 출력 워드로 데이터 레지스터(DR0~DR15)를 지정하여 실행하면 명령어 실행 에러가 발생하고 ER 플래그가 ON됩니다.

-For Reference Only-

● 보조 기억 영역 플래그와 워드

이름	어드레스	설명
통신 포트 사용 가능 플래그	A202.00~ A202.07	해당 포트 번호로 네트워크 명령어를 실행할 수 있거나 해당 포트 번호로 백그라운드 실행을 수행할 수 있을 때 ON됩니다. 비트 00~07은 통신 포트 0~7에 대응합니다. 간이 백업 기능으로 메모리 카드에 대한 쓰기 또는 비교 동작을 수행하는 경우, 통신 포트가 자동으로 할당되며 해당 플래그는 동작 중에 ON되고 동작이 완료되면 OFF됩니다.
통신 포트 에러 플래그	A219.00~ A219.07	네트워크 명령어 실행 중에 에러가 발생하면 ON됩니다. 정상 완료 시에는 OFF됩니다. 비트 00~07은 통신 포트 0~7에 대응합니다. 간이 백업 기능으로 메모리 카드에 대한 쓰기 또는 비교 동작을 수행하는 경우, 통신 포트가 자동으로 할당됩니다. 에러가 발생하면 해당 플래그가 ON되고 간이 백업 동작이 정상적으로 종료되면 OFF됩니다.
통신 포트 완료 코드	A203~A210	네트워크 명령어를 실행했을 때 해당 포트 번호의 완료 코드가 저장됩니다. 백그라운드 실행이 완료되면 그 내용은 0000 hex로 클리어됩니다. 워드 A203~A210은 통신 포트 0~7에 대응합니다. 간이 백업 기능으로 메모리 카드에 대한 쓰기 또는 비교 동작을 수행하는 경우, 통신 포트가 자동으로 할당되며 해당 워드에 완료 코드가 저장됩니다.
백그라운드 실행 ER/AER 플래그	A395.10	백그라운드에서 실행 중인 명령어에 명령어 실행 에러 또는 불법 액세스 에러가 발생하면 ON됩니다. 전원이 ON되거나 운전이 개시되면 OFF됩니다.
백그라운드 실행 IR00 출력	A595 및 A596	백그라운드에서 실행된 명령어의 출력처로 인덱스 레지스터가 지정된 경우 이 워드가 출력을 받습니다. IR00으로는 출력되지 않습니다. 범위: 0000 0000~FFFF FFFF hex 하위 4자리: A595, 상위 4자리: A596
백그라운드 실행 DR00 출력	A597	백그라운드에서 실행된 명령어의 출력처로 데이터 레지스터가 지정된 경우 이 워드가 출력을 받습니다. DR00으로는 출력되지 않습니다. 범위: 0000~FFFF hex
백그라운드 실행 이퀄 플래그 출력	A598.01	백그라운드에서 실행된 SRCH(181)에서 일치하는 데이터가 발견되면 이 플래그가 ON됩니다.

AI 번역중
-For Reference Only-

참고

CPU 유닛의 내부 논리 포트는 백그라운드 실행과 다음 명령어 양쪽에 사용됩니다:

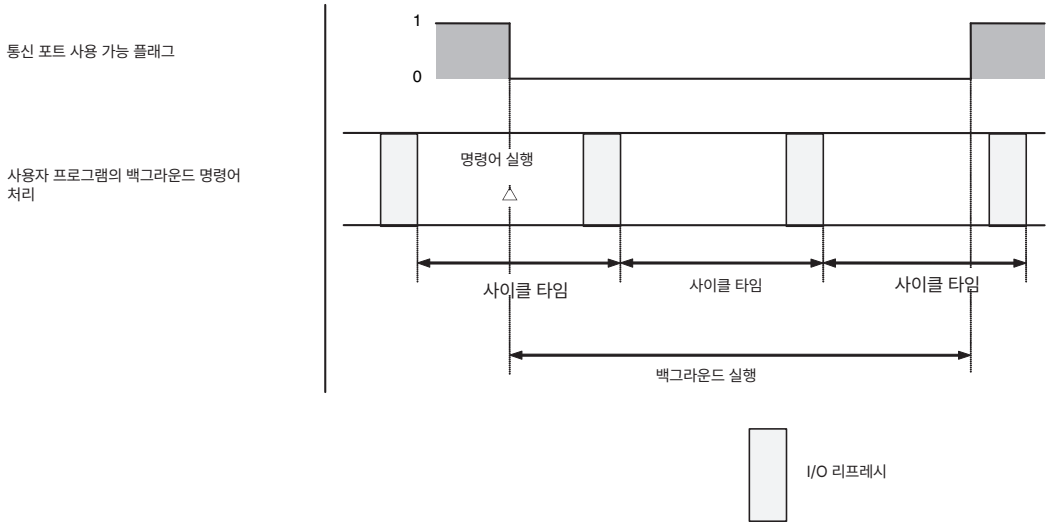
- SEND(090), RECV(098), CMND(490) 등(네트워크 명령어)
- PMCR(260)(프로토콜 매크로)
- TXDU(256) 및 RXDU(255)(시리얼 통신 유닛에서 사용하는 무순서 통신 명령어)
- EXPLT, EGATR, ESATR, ECHRD, ECHWR 등(익스플리시트 명령어)

백그라운드 명령어와 위 명령어는 같은 포트에서 동시에 실행할 수 없습니다. 통신 포트 사용 가능 플래그를 사용하여 각 포트에서 한 번에 하나의 명령어만 실행되도록 하십시오.

주 통신 포트 사용 가능 플래그가 OFF인 포트에 대해 백그라운드 실행을 지정한 명령어를 실행하면 ER 플래그가 ON되고 그 백그라운드 명령어는 실행되지 않습니다.

● 통신 포트 사용 가능 플래그

통신 포트 사용 가능 플래그는 포트가 사용되지 않을 때 ON, 포트에서 처리가 수행되는 중에는 OFF입니다.

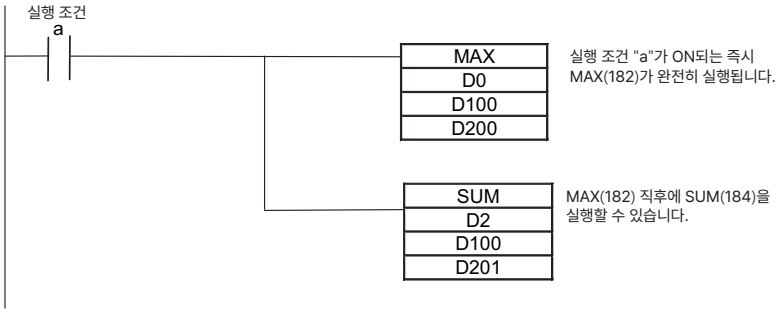


AI 번역본
-For Reference Only-

프로그래밍 예 1

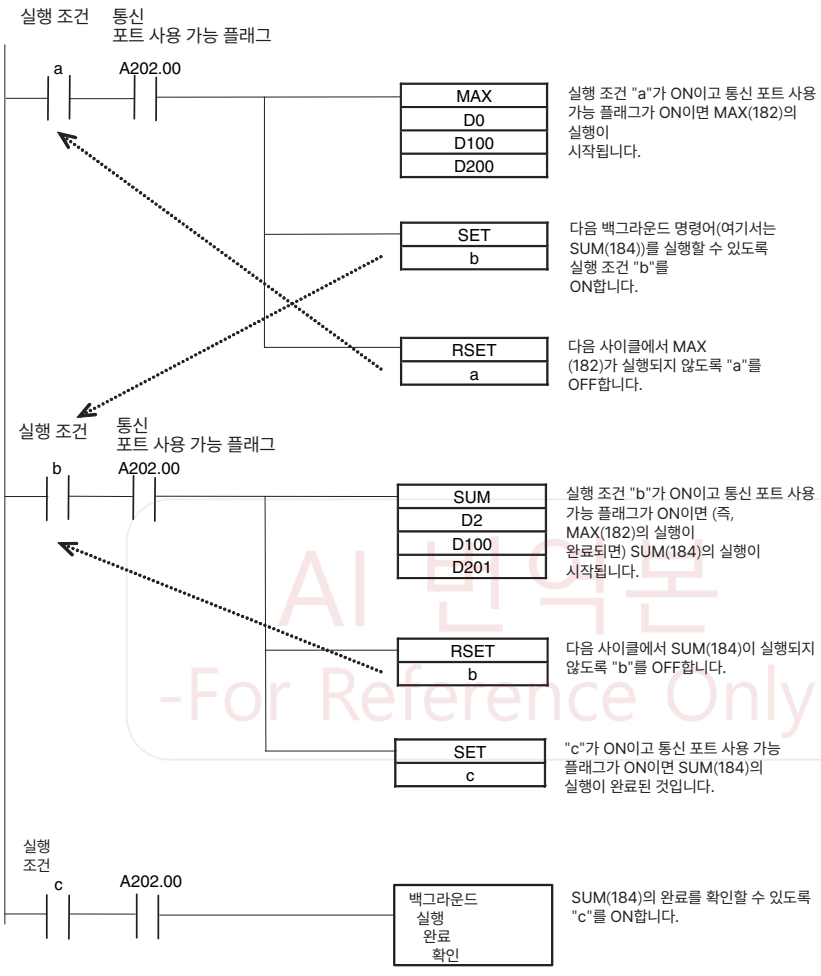
● 백그라운드 실행을 사용하지 않는 프로그래밍

아래와 같이 명령어를 실행하면 처리가 완료됩니다.



● 백그라운드 실행을 사용하는 프로그래밍

백그라운드 실행을 사용하는 경우, 지정한 통신 포트 사용 가능 플래그가 ON일 때만 (즉, 그 포트가 백그라운드 실행이나 네트워크 통신에 이미 사용되고 있지 않을 때만) MAX(182)가 실행되도록 프로그램을 변경합니다. 또한 SET 및 RSET 명령어로 입력 조건을 제어하여 처리가 올바른 순서로 수행되도록 합니다. (다음 예에서는 백그라운드 실행에 통신 포트 0을 사용합니다.)



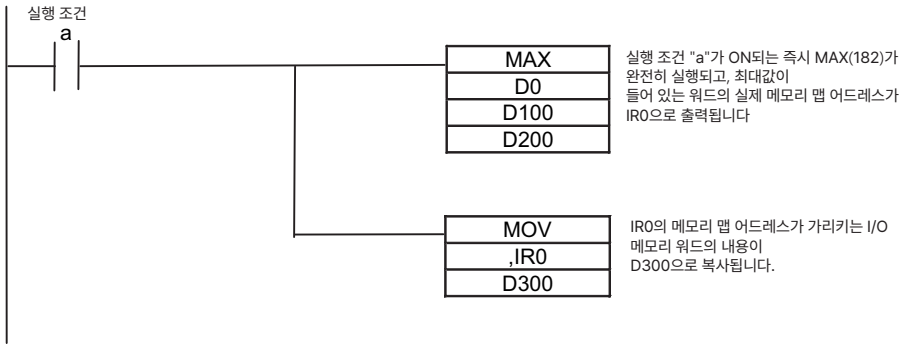
AI 번역본
-For Reference Only-

프로그래밍 예 2

이 예는 MAX(182), MIN(183), SRCH(181)에서 가능한 인덱스 레지스터 출력을 지정한 경우의 백그라운드 실행을 나타냅니다.

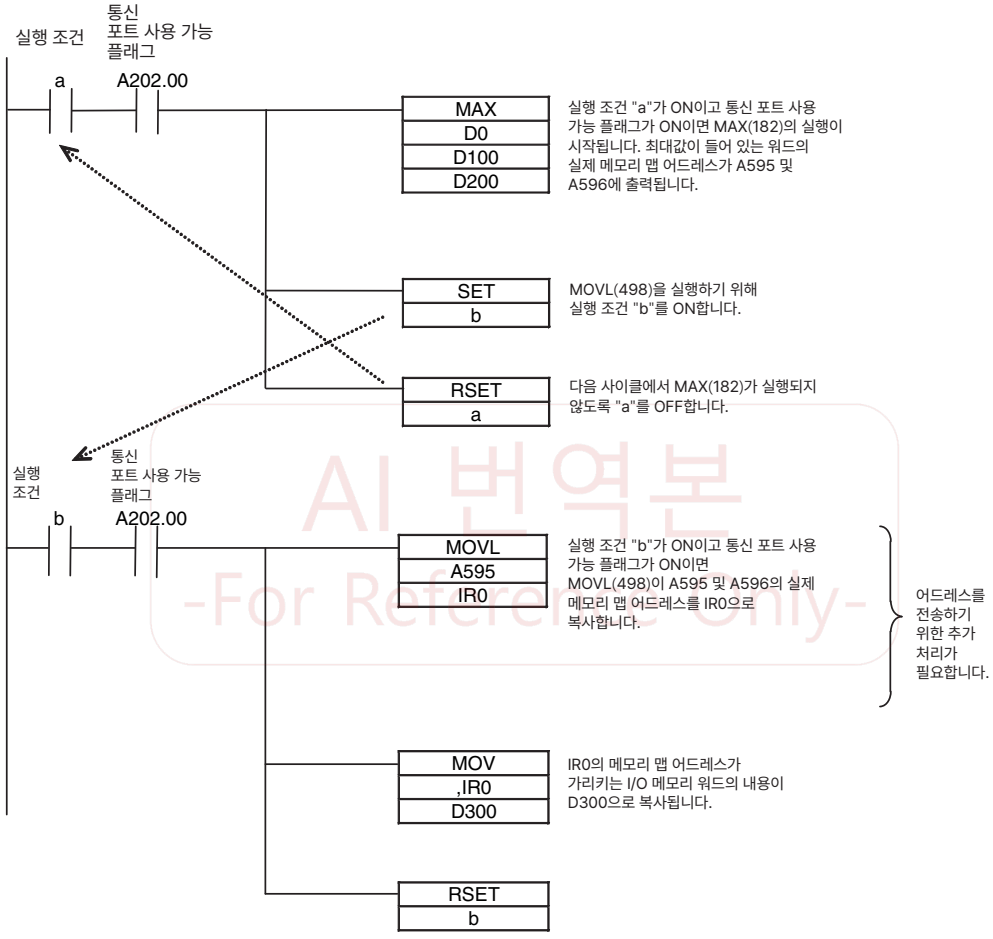
● 백그라운드 실행을 사용하지 않는 프로그래밍

아래와 같이 최대값 또는 최소값이 들어 있는 워드의 PLC 메모리 맵 어드레스가 인덱스 레지스터로 출력됩니다.



● 백그라운드 실행을 사용하는 프로그래밍

백그라운드 실행에서는 최대값 또는 최소값이 들어 있는 워드의 PLC 메모리 맵 어드레스가 A595 및 A596에 출력됩니다. 그 후 MOVL(498)로 그 PLC 메모리 맵 어드레스를 인덱스 레지스터로 전송합니다.



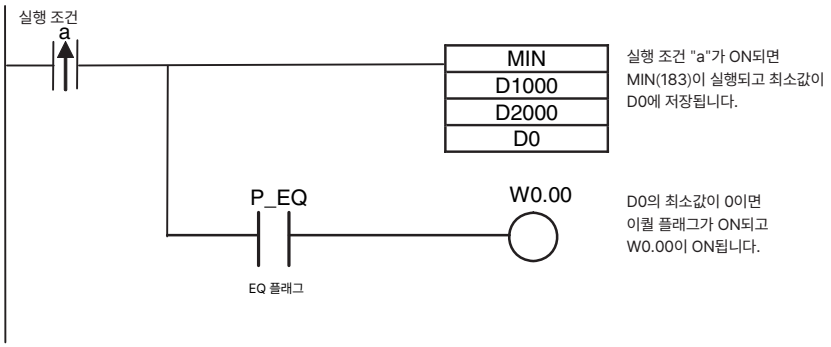
AI 번역본
-For Reference Only-

프로그래밍 예 3

이 예는 조건 플래그를 참조하는 경우의 백그라운드 실행을 나타냅니다.

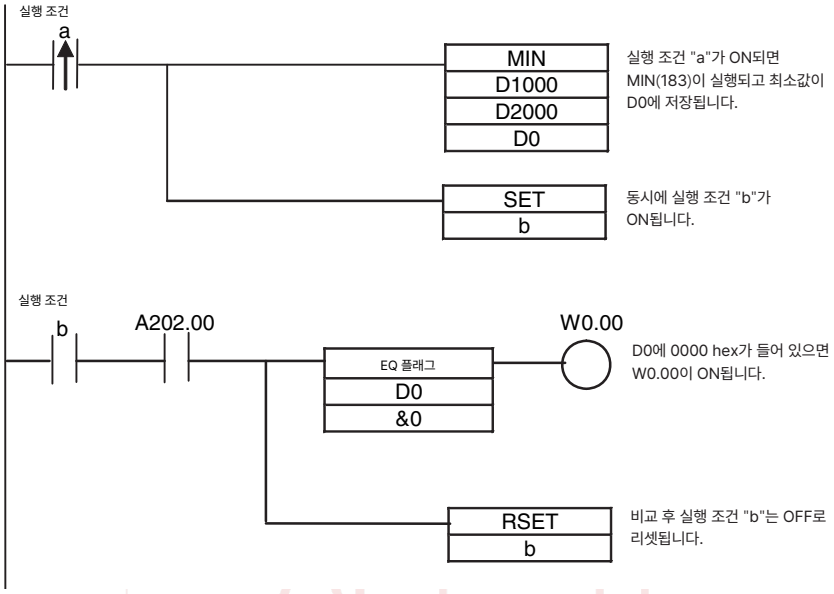
● 백그라운드 실행을 사용하지 않는 프로그래밍

MIN(183)이 찾은 최소값이 0인지 확인하기 위해 MIN(183) 실행 직후에 이퀄 플래그의 상태를 확인합니다.



● 백그라운드 실행을 사용하는 프로그래밍

다음 그림과 같이 AND =(300) 명령어를 사용하여 MIN(183)이 찾은 최소값이 0인지 확인합니다.



-For Reference Only-

10-2-6 고속 인터럽트 기능

고속 인터럽트 기능은 일정한 제약 조건 하에서 인터럽트 태스크의 실행을 개선합니다(유닛 버전 1.1 이후).

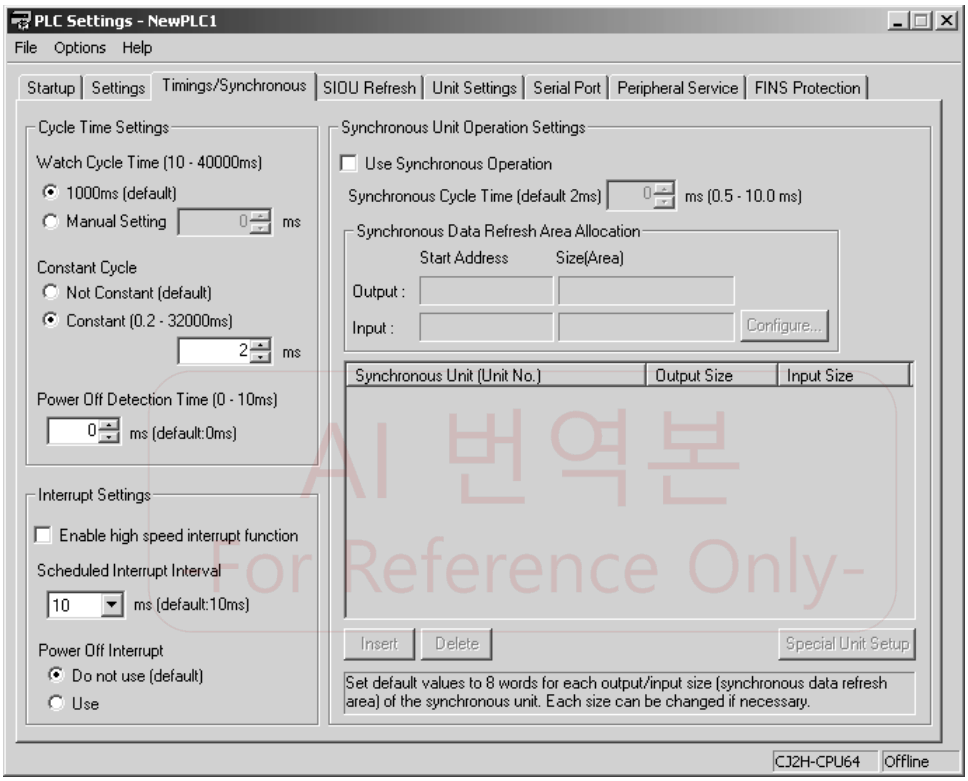
인터럽트 오버헤드 타임의 단축

I/O 인터럽트 태스크, 외부 인터럽트 태스크 또는 정주기 인터럽트 태스크를 실행할 때 오버헤드 타임(즉, 인터럽트 태스크 기동 시간 + 사이클릭 태스크 복귀 시간)을 단축하려면 다음 순서를 사용하십시오.

인터럽트 태스크 종별	인터럽트 오버헤드 타임	
	고속 인터럽트 기능 유효	고속 인터럽트 기능 무효 (디폴트)
I/O 인터럽트 태스크 및 외부 인터럽트 태스크	25 µs (인터럽트 태스크 기동 시간 17 µs + 사이클릭 태스크 복귀 시간 8 µs)	37 µs (인터럽트 태스크 기동 시간 26 µs + 사이클릭 태스크 복귀 시간 11 µs)
정주기 인터럽트 태스크	21 µs (인터럽트 태스크 기동 시간 13 µs + 사이클릭 태스크 복귀 시간 8 µs)	33 µs (인터럽트 태스크 기동 시간 22 µs + 사이클릭 태스크 복귀 시간 11 µs)

- **PLC 설정**

CX-Programmer를 사용하는 경우 Timings/Synchronous 탭 페이지에서 Enable high-speed interrupt function 체크 박스를 선택합니다.



AI 번역본
For Reference Only-

● 고속 인터럽트 기능을 유효하게 한 경우의 제약

고속 인터럽트 기능을 유효하게 하면 다음 제약이 적용됩니다.

・다음 명령어(예: 네트워크 통신 명령어)는 인터럽트 태스크에서 사용할 수 없습니다.

니모닉	명령어 이름
EMBC(281)	EM 뱅크 선택
SEND(090)*	네트워크 송신
RECV(098)*	네트워크 수신
CMND(490)*	커맨드 송신
PMCR(260)*	프로토콜 매크로
TXDU(256)*	시리얼 통신 유닛 경유 송신
RXDU(255)*	시리얼 통신 유닛 경유 수신
EXPLT(720), EGATR(721), ESATR(722), ECHRD(723), ECHWR(724)	익스플리시트 메시지 송신(EXPLT(720)), 익스플리시트 어트리뷰트 취득(EGATR(721)), 익스플리시트 어트리뷰트 설정 (ESATR(722)), 익스플리시트 워드 읽기(ECHRD(723), 익스플리시트 워드 쓰기(ECHWR(724))

* SEND2(491), RECV2a(492), CMND2(493), PMCR2(264), DTXDU(262), DRXDU(261)은 고속 인터럽트 기능을 유효하게 한 경우 인터럽트 태스크에서 사용할 수 있습니다.

・백그라운드에서 실행되는 명령어(즉, 테이블 데이터 처리 명령어, 문자열 처리 명령어, 데이터 시프트 명령어)는 인터럽트 태스크에서 사용할 수 없습니다.

・다음 보조 기억 영역 워드의 데이터는 유효하지 않게 됩니다: A440(최대 인터럽트 태스크 처리 시간)과 A441(처리 시간이 최대인 인터럽트 태스크).

・고속 인터럽트 기능을 유효하게 하면 내장 EtherNet/IP 포트 및 EtherNet/IP 유닛의 리프레시 성능이 다음 표와 같이 저하됩니다.

	고속 인터럽트 기능 유효	통상 동작
오버헤드 처리 시간(데이터 링크 사용 시의 추가 시간)	100 µs + 전송 워드 수 × 0.87 µs	100 µs + 전송 워드 수 × 0.33 µs
EtherNet/IP용 DLNK(226) 명령어의 실행 시간(CPU 버스 유닛 즉시 I/O 리프레시)	910 µs/1,000 워드	430 µs/1,000 워드

정주기 인터럽트 태스크의 최소 시간 간격

MSKS(690)을 사용하여 정주기 인터럽트 0(인터럽트 태스크 2)에 최소 시간 간격 0.1 ms를 설정할 수 있습니다. 정주기 인터럽트 1에서는 이것이 불가능합니다.

AI 번역본
-For Reference Only-

● 정주기 인터럽트 0의 최소 시간 간격을 0.1 ms로 설정하기 위한 조건

1 고속 인터럽트 기능이 유효해야 합니다. (PLC 설정에서 Enable high-speed interrupt function 옵션을 선택하십시오.)

고속 인터럽트 기능이 무효인 경우, MSKS(690)에서 0.1 ms를 지정하면 명령어 처리 에러가 발생합니다.

2 CPU 유닛의 페리페럴 포트(USB)나 시리얼 포트에 기기를 직접 접속해서는 안 됩니다. 지원 소프트웨어(예: CX-Programmer)나 HMI를 CPU 유닛의 페리페럴 포트(USB)나 시리얼 포트에 직접 접속하면 0.1 ms의 시간 간격이 안정되지 않을 수 있습니다. 접속이 필요한 경우에는 내장 EtherNet/IP 포트를 사용하거나 EtherNet/IP 유닛 또는 시리얼 통신 유닛의 통신 포트를 통해 접속하십시오.

사용상의 주의

고속 인터럽트 기능은 일정한 제약 조건 하에서 인터럽트 태스크의 실행을 개선합니다. 운전하기 전에 사용 가능한 조건을 반드시 확인하십시오. 특히, 정주기 인터럽트의 최소 시간 간격을 0.1 ms로 설정한 운전은 지원 소프트웨어(예: CX-Programmer)나 HMI가 CPU 유닛의 페리페럴 포트(USB)나 시리얼 포트에 직접 접속되어 있지 않다는 조건에서만 가능합니다. 직접 접속한 경우에는 0.1 ms의 시간 간격으로 운전되지 않을 수 있습니다. 메인터넌스 등 다른 이유로 지원 소프트웨어를 CPU 유닛에 직접 접속해야 하는 경우에는, 0.1 ms의 시간 간격으로 정주기 인터럽트가 실행되지 않아도 설비에 영향이 없음을 확인한 후에 접속을 하십시오.

AI 번역본
-For Reference Only-

10-3 기동 설정과 메인터넌스

10-3-1 동작 모드 변경 시 및 기동 시의 유지 설정

동작 모드 변경

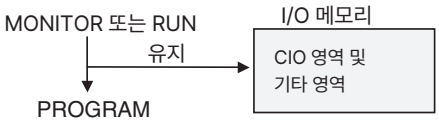
● 프로그램 실행 개시

프로그램 실행을 개시하기 위해 CPU 유닛을 PROGRAM 모드에서 RUN/MONITOR 모드로 전환할 때 I/O 메모리의 모든 데이터를 유지하려면 IOM 유지 비트(A500.12)를 ON하십시오.



● 프로그램 실행 정지

IOM 유지 비트(A500.12)가 ON이면 프로그램 실행을 정지하기 위해 CPU 유닛을 RUN/MONITOR 모드에서 PROGRAM 모드로 전환할 때도 I/O 메모리의 모든 데이터가 유지됩니다.



IOM 유지 비트에 의해 유지되는 영역은 다음과 같습니다: CIO 영역(I/O 영역, 데이터 링크 영역, CPU 버스 유닛 영역, 특수 I/O 유닛 영역, DeviceNet 영역, EtherCAT 리모트 I/O 영역, 내부 I/O 영역), 워크 영역, 타이머 완료 플래그, 타이머 현재값.

● 보조 기억 영역 플래그와 워드

이름	어드레스	설명
IOM 유지 비트	A500.12	이 비트가 ON이면 동작 모드가 PROGRAM과 RUN/MONITOR 모드 사이에서 변경되거나 전원이 ON될 때 I/O 메모리 전체가 유지됩니다. ON: 동작 모드가 변경될 때 I/O 메모리가 유지됩니다. OFF: 동작 모드가 변경될 때 I/O 메모리가 클리어됩니다.

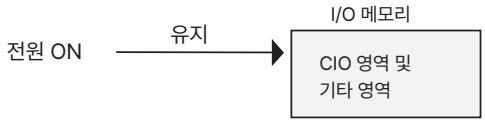
IOM 유지 비트가 ON이면 프로그램 실행이 정지될 때 출력 유닛의 모든 출력이 유지됩니다. 프로그램이 다시 개시되면 출력은 프로그램이 정지되기 전과 동일한 상태가 됩니다.

(IOM 유지 비트가 OFF인 경우, 출력이 클리어된 후에 명령어가 실행됩니다.)

-For Reference Only-

PLC 전원 ON

PLC가 ON될 때 I/O 메모리의 모든 데이터가 유지되도록 하려면 IOM 유지 비트 (A500.12)가 ON이어야 하며 PLC 설정에서 그것이 보호되어야 합니다.

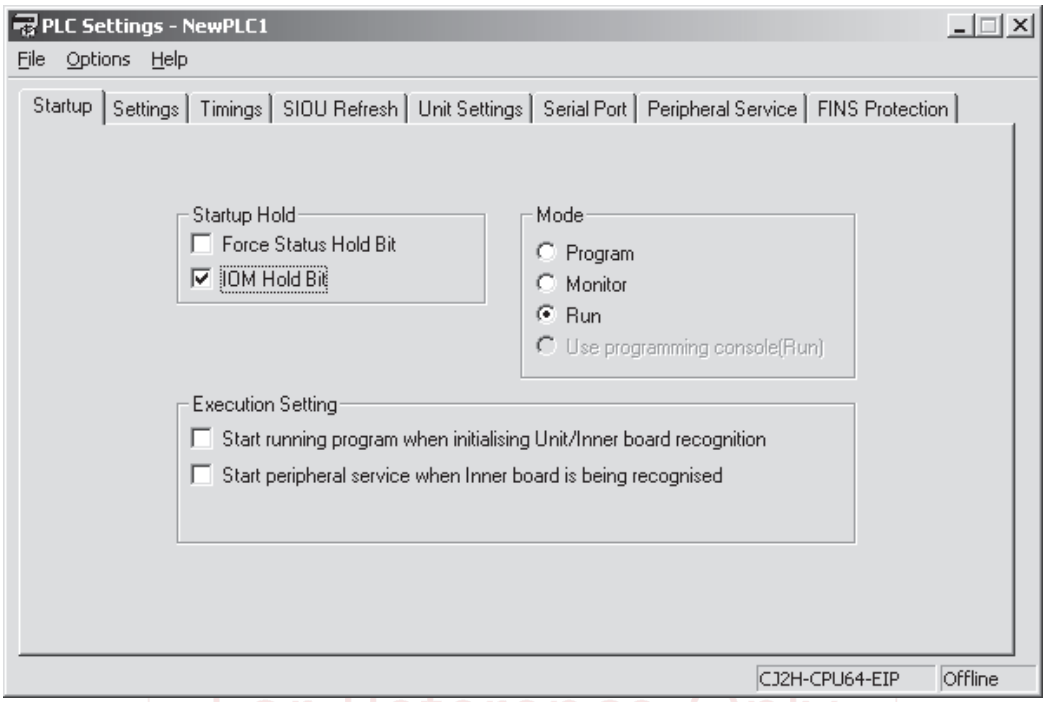


● 보조 기억 영역 플래그와 워드

이름	어드레스	설명
IOM 유지 비트	A500.12	이 비트가 ON이면 동작 모드가 PROGRAM과 RUN/MONITOR 사이에서 변경되거나 전원이 ON될 때 I/O 메모리 전체가 유지됩니다. ON: 동작 모드가 변경될 때 I/O 메모리가 유지됩니다. OFF: 동작 모드가 변경될 때 I/O 메모리가 클리어됩니다.

● PLC 설정

CX-Programmer를 사용하는 경우 Startup 탭 페이지의 기동 시 유지 영역에서 IOM Hold Bit 체크 박스를 선택하여 설정합니다.



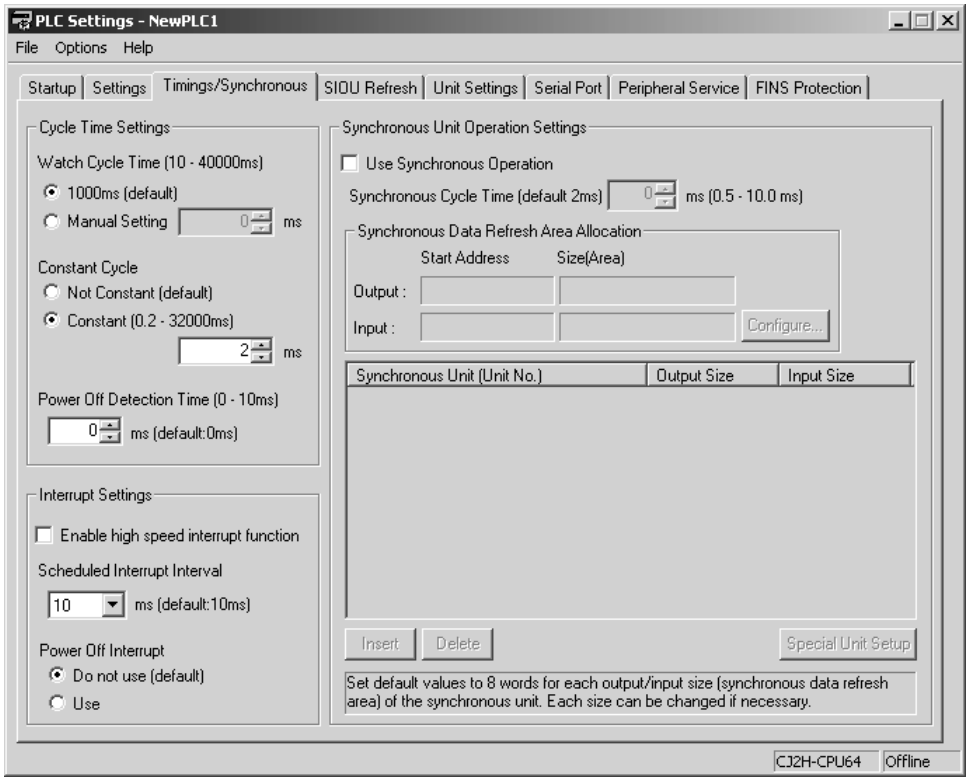
-For Reference Only-

10-3-2 전원 OFF 검출 지연 설정

디폴트에서는 10 ms 이상의 AC 정전(DC 정전의 경우 2 ms)이 전원 전압이 최소 정격값의 85%(DC 전원의 경우 80%) 아래로 떨어진 후 약 10~25 ms(DC 전원의 경우 2~5 ms) 이내에 검출됩니다. PLC 설정에는 이 시간을 연장할 수 있는 설정이 있습니다. 전원 OFF 인터럽트 태스크가 유효한 경우, 정전이 확정되면 그것이 실행되고, 그렇지 않으면 CPU 유닛이 리셋되어 운전이 정지됩니다.

PLC 설정

CX-Programmer를 사용하는 경우 Timings/Synchronous 탭 페이지의 Power Off detection time 필드에서 설정합니다.



참고

전원 OFF 인터럽트 태스크를 사용하지 않는 경우, 정전 대책으로 전원 OFF 검출 시간을 10 ms 이하로 설정하십시오. 다만 CJ1W-PD022 전원 유닛을 사용하는 경우에는 지연을 설정할 수 없으므로 디폴트 설정인 0 ms를 사용하십시오.

-For Reference Only-

10-3-3 전원 OFF 인터럽트 무효화

프로그램의 특정 부분을 전원 OFF 인터럽트로부터 보호하여, 전원이 차단되더라도 CPU 유닛이 리셋되기 전에 실행되도록 할 수 있습니다. 이것은 인터럽트 금지(DI(693))와 인터럽트 허가(EI(694)) 명령어를 사용하여 실현합니다.

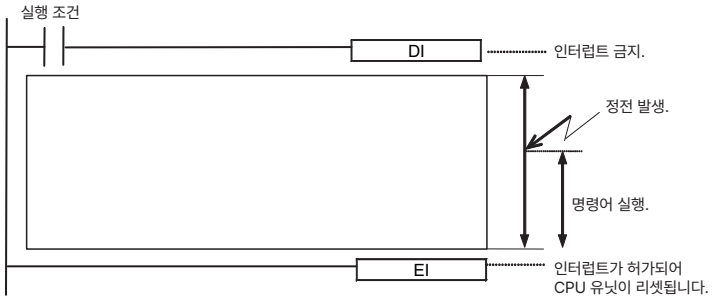
이 기능은 한 묶음으로 실행되어야 하는 명령어 집합에 사용할 수 있습니다. 예를 들어 다음에 전원을 ON했을 때 중간 저장 데이터로 실행이 시작되지 않도록 하는 경우입니다.

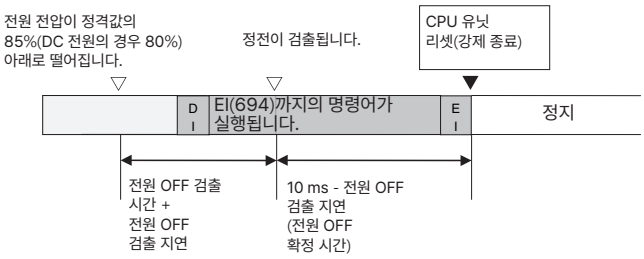
순서

1 전원 OFF 인터럽트를 금지할 수 있도록 A530의 전원 OFF 인터럽트 금지 설정을 A5A5 hex로 설정합니다.

2 PLC 설정에서 전원 OFF 인터럽트 무효 설정을 무효로 합니다(이것이 디폴트 설정입니다).

3 보호할 프로그램 부분 앞에서 DI(693)으로 인터럽트를 금지하고, 그 부분 뒤에서 EI(694)로 인터럽트를 허가합니다. DI(693)과 EI(694) 사이의 명령어를 실행하는 중에 정전이 발생하더라도 DI(693)과 EI(694) 사이의 모든 명령어는 전원 OFF 인터럽트가 실행되기 전에 완료됩니다.





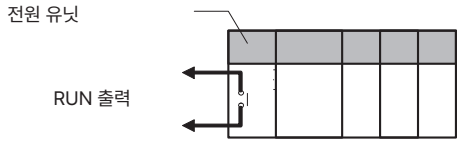
- **보조 기억 영역 플래그와 워드**

이름	어드레스	의미
전원 OFF 인터럽트 금지 설정	A530	EI(694)가 실행될 때까지 DI(693)으로 전원 OFF 인터럽트 처리(전원 OFF 인터럽트 태스크의 실행은 제외)를 금지할 수 있게 합니다. A5A5 hex: DI(693)으로 전원 OFF 인터럽트 처리를 금지할 수 있게 합니다 그 밖의 값: DI(693)으로 전원 OFF 인터럽트 처리를 금지할 수 없게 합니다

10-3-4 RUN 출력

RUN 출력

CJ1W-PA205R 전원 유닛에는 RUN 출력이 탑재되어 있습니다. 이 출력점은 CPU 유닛이 RUN 또는 MONITOR 모드로 동작 중일 때 ON(닫힘)이 됩니다.



이름	동작
RUN 출력	MONITOR 또는 RUN 모드: ON(닫힘) PROGRAM 모드: OFF(열림)

이 RUN 출력은 PLC가 ON이 아니면 출력 유닛의 외부 전원이 전력을 공급하지 못하게 하는 비상 정지 회로 등 외부 안전 회로를 구성하는 데 사용할 수 있습니다.

주 RUN 출력이 없는 전원 유닛을 사용하는 경우, 상시 ON 플래그(A1)를 출력 유닛의 출력점에 대한 실행 조건으로 프로그래밍하여 동등한 출력을 만들 수 있습니다.

안전상의 요점

출력 유닛의 외부 전원이 PLC의 전원보다 먼저 ON되면, PLC가 처음 ON될 때 출력 유닛이 순간적으로 오동작할 수 있습니다. 오동작을 방지하려면 출력 유닛의 외부 전원이 PLC 자체의 전원보다 먼저 ON되지 않도록 하는 외부 회로를 추가하십시오. PLC가 RUN 또는 MONITOR 모드로 동작 중일 때만 외부 전원에서 전력이 공급되도록, 위에서 설명한 것과 같은 페일세이프 회로를 구성하십시오.

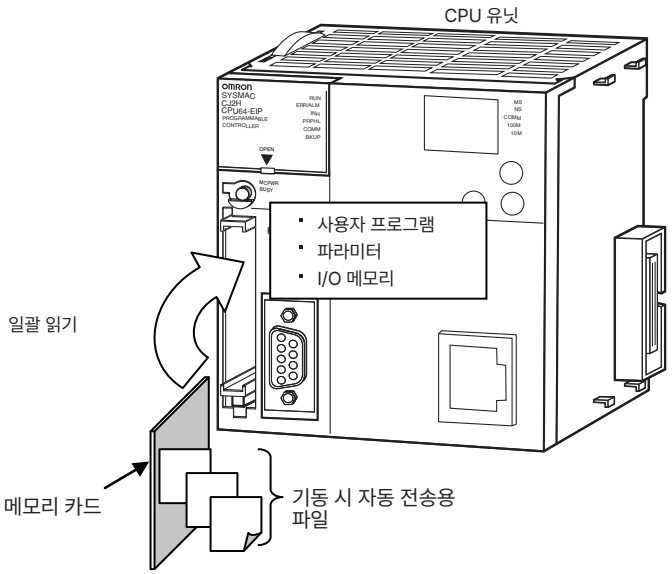
AI 번역본
-For Reference Only-

10-3-5 기동 시 자동 전송

개요

기동 시 자동 전송은 전원이 ON될 때 메모리 카드에서 CPU 유닛으로 사용자 프로그램, 파라미터, I/O 메모리 데이터를 읽어 들이는 데 사용합니다.

기동 시 자동 전송용 파일은 CX-Programmer의 Memory Card 윈도우에서 작성하여 메모리 카드에 저장할 수 있습니다.



이 기능은 EM 파일 메모리를 읽어 들이는 데는 사용할 수 없습니다.

참고

CJ2 CPU 유닛에서는 배터리리스 운전 중에 I/O 메모리가 불안정해지므로, 이 기능을 사용하여 I/O 메모리 값을 고정할 수 있습니다.

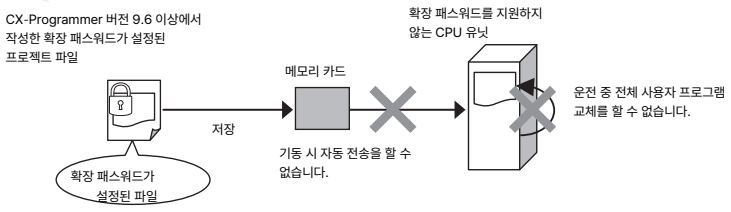
사용상의 주의

확장 패스워드가 설정된 프로그램 파일은 확장 패스워드를 지원하지 않는 CPU 유닛에서는 기동 시 자동 전송을 할 수 없으며, 운전 중 전체 사용자 프로그램 교체에도 사용할 수 없습니다.

AI 번역본
-For Reference Only-

- **확장 패스워드를 지원하지 않는 CPU 유닛**

・CJ2H CPU 유닛, 유닛 버전 1.4 이전
・CJ2M CPU 유닛, 유닛 버전 2.0 이전



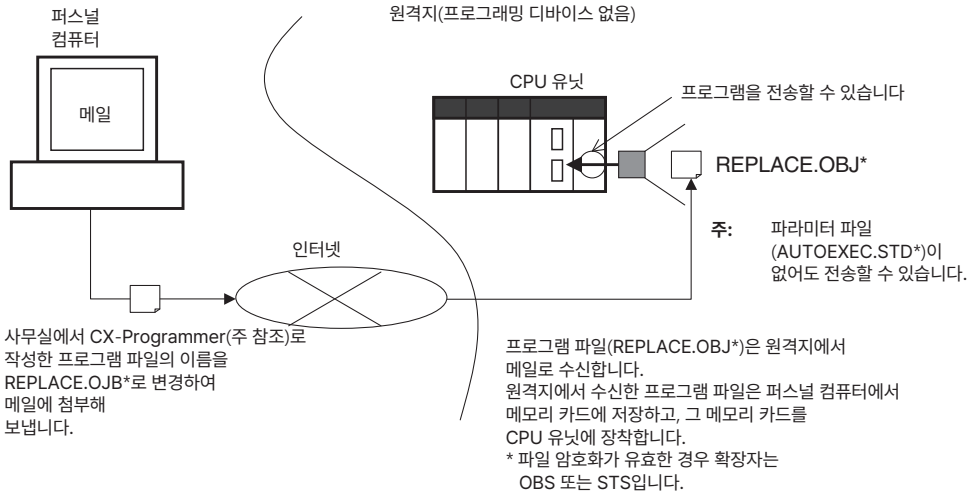
기동 시 자동 전송의 종류

기동 시 자동 전송에는 두 가지 방법이 있습니다: 파라미터 파일(AUTOEXEC.STD/AUTOEXEC.STS)을 사용하는 전송과 사용하지 않는 전송입니다. 작성할 파일 이름은 두 전송 방식 중 어느 것을 사용하는지에 따라 달라집니다.

기동 시 자동 전송 종류	파일 이름	용도
파라미터 파일 있는 전송	AUTOEXEC 또는 ATEXEC□□	프로그램과 네트워크 설정의 덮어쓰기
파라미터 파일 없는 전송	REPLACE 또는 REPLC□□	프로그램의 덮어쓰기

● 파라미터 파일 없이 파일을 자동 전송하는 적용 예

프로그램/네트워크 심볼 파일(.OBJ/.OBS)은 실제 기기 없이 사무실에서 오프라인으로 작성하여 파라미터 파일(.STD/.STS) 없이 원격지로 전송할 수 있습니다. 프로그램/네트워크 심볼 파일은 프로그래밍 디바이스를 사용하지 않고 원격지에서 메모리 카드에 저장할 수 있으며, 그 메모리 카드를 사용하여 기동 시에 프로그램을 CPU 유닛으로 자동 전송할 수 있습니다.



AI 번역본
-For Reference Only-

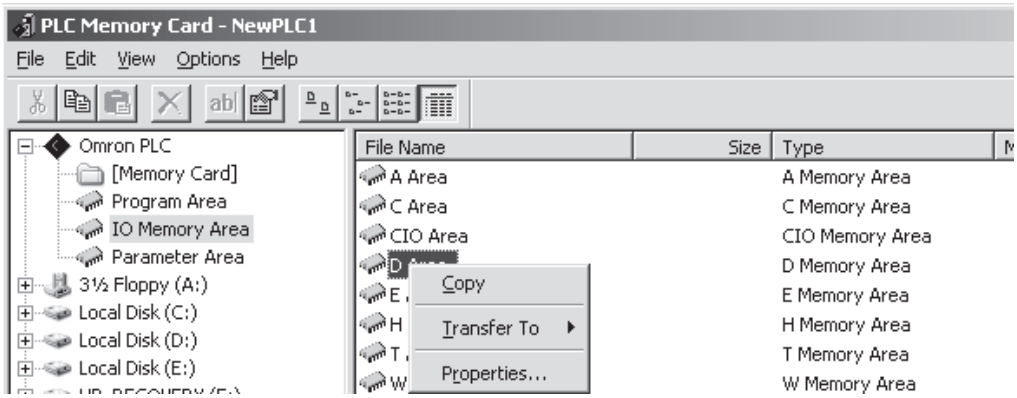
순서

● 기동 시 자동 전송용 파일의 작성과 전송

CX-Programmer를 사용하여 기동 시 자동 전송용 파일을 작성하고, CPU 유닛에 장착된 메모리 카드로 전송합니다.

・프로그램 영역 및 파라미터 영역:
오른쪽 클릭하고 Transfer to – Memory Card를 선택합니다. 기동 시 자동 전송용 파일의 이름을 입력한 후 작성하여 전송합니다.

・I/O 메모리 영역:
오른쪽 창에서 D 영역 또는 E 영역을 오른쪽 클릭합니다. 기동 시 자동 전송용 파일의 이름을 입력한 후 작성하여 전송합니다.



기동 시 자동 전송에는 다음 파일 이름이 사용됩니다.

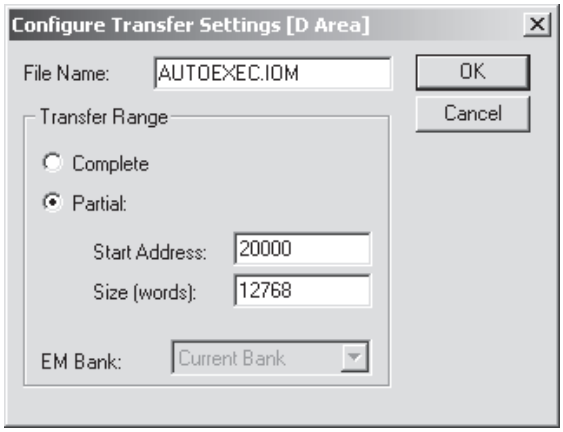
파일 종별	내용	파라미터 파일 있는 기동 시 자동 전송	필수?	파라미터 파일 없는 기동 시 자동 전송	필수?
프로그램/네트워크 심볼 파일	사용자 프로그램	AUTOEXEC.OBJ AUTOEXEC.OBS *	예	REPLACE.OBJ REPLACE.OBS *	예
파라미터 파일	・PLC 이름 ・PLC 설정 ・등록 I/O 테이블 ・라우팅 테이블 ・CPU 버스 유닛 셋업 영역	AUTOEXEC.STD AUTOEXEC.STS *	예	없음	---
데이터 파일	특수 I/O 유닛 및 CPU 버스 유닛 할당 DM 영역 워드 (D20000~ D32767)	AUTOEXEC.IOM AUTOEXEC.IOS *	---	REPLACE.IOM REPLACE.IOS *	---
	DM 영역(D0~D19999)	ATEXECDM.IOM ATEXECDM.IOS *	---	REPLACDM.IOM REPLACDM.IOS *	---
	EM 영역	AUTOEM□□.IOM AUTOEM□□.IOS * (□□: 00~18 hex)	---	RPLCEM□□.IOM RPLCEM□□.IOS * (□□:00~18 hex)	---

AI 번역본 -For Reference Only

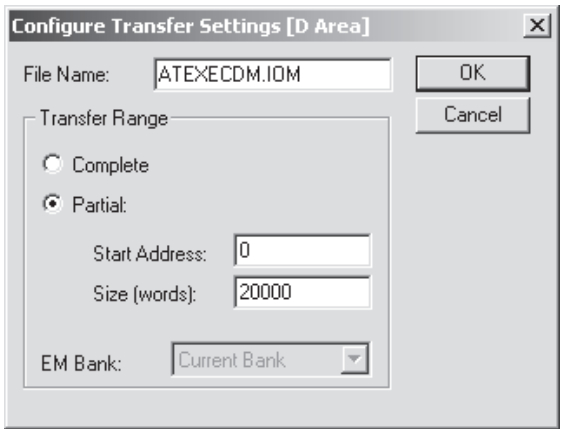
* 파일 암호화가 유효한 경우 암호화된 파일이 작성됩니다.

예를 들어 DM 영역에는 다음 선두 어드레스와 사이즈를 지정합니다.

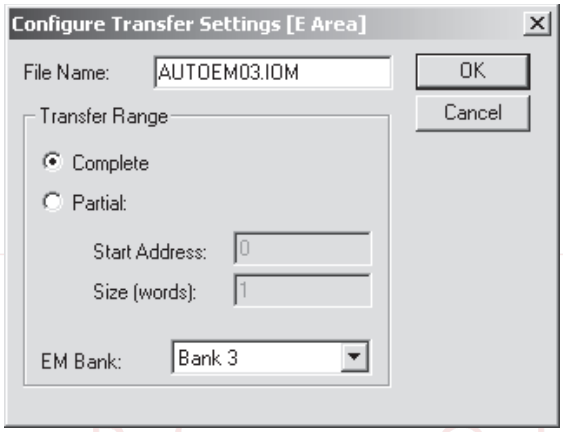
· 특수 I/O 유닛 및 CPU 버스 유닛 할당 DM 영역 워드



· DM 영역:



· EM 영역:



-For Reference Only-

● 기동 시 자동 전송의 실행

1 PLC를 OFF합니다.

2 CPU 유닛 전면의 딥 스위치 핀 2를 ON합니다.

핀 7이 ON이면 간이 백업이 동작합니다. 반드시 OFF로 설정하십시오.

3 필요한 파일이 저장된 메모리 카드를 CPU 유닛에 장착합니다.

4 PLC를 ON합니다.

기동 시 자동 전송이 시작되고 SD BUSY/BUSY 표시등이 황색으로 점멸합니다. 전송이 완료되면 SD BUSY/BUSY 표시등이 소등됩니다. 기동 시 자동 전송에 실패하면 카드 전송 에러가 발생하고 CPU 유닛의 운전이 정지합니다.

● CPU 유닛 전면의 딥 스위치

스위치 핀 번호	이름	설정
2	기동 시 자동 전송	ON: 기동 시 자동 전송을 실행합니다 OFF: 기동 시 자동 전송을 실행하지 않습니다

● 보조 기억 영역 플래그와 워드

이름	어드레스	설명
카드 전송 에러 플래그 (치명 에러)	A401.03	메모리 카드로부터의 기동 시 자동 전송에 실패하면 ON됩니다. CPU 유닛의 운전이 정지하고 CPU 유닛 전면의 ERR/ALM 표시등이 점등합니다.

기동 시 자동 전송용 데이터 파일

기동 시 자동 전송에는 다음 여섯 종류의 데이터 파일이 사용됩니다.

기동 시 자동 전송용 데이터 파일		내용	범위	선두 어드레스	사이즈
파라미터 파일 있는 기동 시 자동 전송	파라미터 파일 없는 기동 시 자동 전송				
AUTOEXEC.IOM AUTOEXEC.IOS (*)	REPLACE.IOM REPLACE.IOS (*)	특수 I/O 유닛 및 CPU 버스 유닛 할당 DM 영역 워드(D20000~D32767)	일부	20000	12,768 워드
ATEXECDM.IOM ATEXECDM.IOS (*)	REPLACDM.IOM REPLACDM.IOS (*)	DM 영역(D00000~D19999)	영역 전체	0	20,000 워드
AUTOEM□□.IOM AUTOEM□□.IOS (*) (□□: 00~18)	AUTOEM□□.IOM RPLCEM□□.IOS (*) (□□: 00~18)	EM	영역 전체	---	---

* CPU 유닛의 파일 암호화 설정이 유효한 경우에는 암호화된 파일을 저장하십시오.

기동 시 자동 전송용 데이터 파일을 작성할 때는 반드시 위의 각 영역 종별과 선두 어드레스를 먼저 설정하십시오. 저장 사이즈는 해당 영역 종별의 마지막 어드레스까지의 사이즈로 설정합니다.

기동 시 메모리 카드의 데이터 파일에 있는 모든 데이터가 D20000, D0, E□_0부터 전송됩니다. 영역 전체에 대해 워드 수가 부족하면 그 영역의 나머지 워드는 변경되지 않습니다.

AI 번역본 -For Reference Only-

파일 암호화가 유효한 CPU 유닛으로의 전송

파일 암호화 설정이 유효한 CPU 유닛에 기동 시 자동 전송을 수행하는 경우, 기동 시 자동 전송용 데이터 파일을 암호화된 파일로 작성하십시오. 암호화된 파일을 작성할 때는 기동 시 자동 전송을 수행하는 CPU 유닛과 동일한 패스워드로 암호화하십시오.

암호화된 파일의 작성 순서

・파일 암호화 설정이 유효한 CPU 유닛에서 파일을 작성합니다.

・CX-Programmer에서 파일을 작성합니다. (프로그램/네트워크 심볼 파일만 작성할 수 있습니다.)

기동 시 자동 전송을 수행하는 CPU 유닛의 파일 암호화 설정과 자동 전송할 파일의 대응 관계는 다음과 같습니다. 패스워드가 일치하지 않으면 기동 시 자동 전송은 실행되지 않습니다.

		기동 시 자동 전송을 수행하는 CPU 유닛의 파일 암호화 설정	
		암호화 무효	암호화 유효
기동 시 자동 전송용 파일	암호화되지 않은 파일	지원	지원하지 않음
	암호화된 파일	지원하지 않음	지원 (*)

* 기동 시 자동 전송은 파일 암호화 설정의 패스워드가 일치하는 경우에만 가능합니다.

AI 번역본
-For Reference Only-

파라미터 파일 있는 기동 시 자동 전송

다음 파일 이름을 사용하십시오. 기동 시 자동 전송을 수행하려면 오른쪽 열에 필수로 표시된 파일이 메모리 카드에 있어야 합니다.

파일 종별	파일 이름	확장자	내용	설명	필수?
프로그램/네트워크 심볼 파일	AUTOEXEC	.OBJ .OBS(*3)	· 사용자 프로그램 · 네트워크 심볼*1	· 사이클릭 태스크와 인터럽트 태스크의 프로그램. · 기동 시 자동 전송을 수행하려면 이 파일이 메모리 카드에 있어야 합니다. · 기동 시 자동 전송을 수행하려면 이 파라미터 파일 (AUTOEXEC.STD)도 메모리 카드에 있어야 합니다.	예
파라미터 파일	AUTOEXEC	.STD .STS(*3)	· PLC 이름 · PLC 설정 · I/O 테이블 · 라우팅 테이블 · CPU 버스 유닛 셋업 영역	· CPU 유닛의 파라미터 영역 데이터 · 전원이 ON되면 파라미터는 CPU 유닛 내의 지정된 위치에 자동으로 저장됩니다. · 사용자가 파일 안에서 데이터를 개별적으로 지정할 필요는 없습니다. · 프로그램/네트워크 심볼 파일이 AUTOEXEC.OBJ인 경우, 기동 시 자동 전송을 수행하려면 파라미터 파일이 메모리 카드에 있어야 합니다.	예
데이터 파일	AUTOEXEC	.IOM .IOS(*3)	DM 영역 데이터 (D20000부터 시작하는 지정된 워드 수의 데이터가 들어 있습니다.)	· D20000부터 시작하는 DM 영역 데이터를 AUTOEXEC.IOM이라는 이름의 파일에 저장합니다. · 기동 시 파일 선두부터의 모든 데이터가 D20000부터 시작하여 전송됩니다. 영역 전체에 대해 워드 수가 부족한 경우, 영역의 나머지 워드는 변경되지 않습니다. · 이 파일은 기동 시 자동 전송을 수행하는 데 필수는 아닙니다.	---
	ATEXECDM		DM 영역 데이터 (D0부터 시작하는 지정된 워드 수의 데이터가 들어 있습니다.)	· D0부터 시작하는 DM 영역 데이터를 ATEXECDM.IOM이라는 이름의 파일에 저장합니다. · 기동 시 파일 선두부터의 모든 데이터가 D0부터 시작하여 전송됩니다. 영역 전체에 대해 워드 수가 부족한 경우, 영역의 나머지 워드는 변경되지 않습니다. · 이 파일은 기동 시 자동 전송을 수행하는 데 필수는 아닙니다.*2	---

*1 CJ2H-CPU□□(-EIP 없음) 버전 1.5 이전 및 CJ2M-CPU1□ 버전 2.1 이전은 지원하지 않습니다. 네트워크 심볼을 포함한 심볼 파일(.OBJ/OBS)을 네트워크 심볼을 지원하지 않는 CJ2/CJ3 CPU 유닛으로 전송하면 A401.03(카드 전송 에러 플래그)이 ON됩니다.

*2 ATEXECDM.IOM 파일의 내용이 AUTOEXEC.IOM 파일의 내용과 중복되는 경우 ATEXECDM.IOM 파일이 우선됩니다.

*3 파일 암호화 기능이 유효한 경우의 확장자입니다.

AI 번역 -For Reference Only-

파일 종별	파일 이름	확장자	내용	설명	필수?
데이터 파일 (계속)	AUTOEM□□	.IOM .IOS(*3)	뱅크 No.□□ 의 EM 영역 데이터 (E□□_0부터 시작하는 지정된 워드 수의 데이터가 들어 있습니다.)	· 뱅크 No. □의 EM 영역 데이터를 E□_0부터 시작하여 AUTOEM□□.IOM이라는 이름의 파일에 저장합니다. (상자는 뱅크 No. 0~C를 가리킵니다. 뱅크의 최대 뱅크 번호는 CPU 유닛의 기종에 따라 다릅니다.) · 기동 시 파일 선두부터의 모든 데이터가 E□_0부터 시작하여 전송됩니다. 영역 전체에 대해 워드 수가 부족한 경우, 영역의 나머지 워드는 변경되지 않습니다. · 이 파일은 기동 시 자동 전송을 수행하는 데 필수는 아닙니다.*2	---

AUTOEXEC.IOM 파일과 ATEXECDM.IOM 파일의 내용이 중복되는 경우, 쓰기에서는 ATEXECDM.IOM 파일이 우선됩니다. (파일이 AUTOEXEC.IOM, ATEXECDM.IOM 순으로 로드되기 때문입니다.)

참고

기동 시 자동 전송과 보조 기억 영역 제어 비트를 사용한 전체 프로그램 교체는 조합하여 사용할 수 있습니다. 기동 시 자동 전송으로 프로그램을 다른 프로그램으로 교체한 다음, 보조 기억 영역 제어 비트를 사용하여 운전 중에 프로그램을 다시 교체하는 것도 가능합니다.

AI 번역본
-For Reference Only-

파라미터 파일 없는 기동 시 자동 전송

다음 파일 이름을 사용하십시오. 기동 시 자동 전송을 수행하려면 오른쪽 열에 필수로 표시된 파일이 메모리 카드에 있어야 합니다.

파일 종별	파일 이름	확장자	내용	설명	필수?
프로그램/네트워크 심볼 파일	REPLACE	.OBJ .OBS(*)	사용자 프로그램	· 내용은 AUTOEXEC.OBJ와 동일합니다. · 파라미터 파일 (AUTOEXEC.STD)이 메모리 카드에 없어도 프로그램/네트워크 심볼 파일은 기동 시에 전송됩니다.	예
파라미터 파일	필요 없음	---	---	· 메모리 카드에서 어떤 파일 이름을 사용하더라도 파라미터 파일은 전송되지 않습니다.	---
데이터 파일	REPLACE	.IOM .IOS(*)	DM 영역 데이터 (D20000부터 시작하는 지정된 워드 수의 데이터가 들어 있습니다.)	· 내용은 AUTOEXEC.IOM과 동일합니다. · 프로그램/네트워크 심볼 파일 (REPLACE.OBJ)이 메모리 카드에 있으면 이 파일의 데이터가 기동 시에 전송됩니다.	---
	REPLCDM		DM 영역 데이터 (D0부터 시작하는 지정된 워드 수의 데이터가 들어 있습니다.	· 내용은 ATEXECDM.IOM과 동일합니다. · 프로그램/네트워크 심볼 파일 (REPLACE.OBJ)이 메모리 카드에 있으면 이 파일의 데이터가 기동 시에 전송됩니다.	---
	RPLCEM□□		뱅크 No. 의 EM 영역 데이터 □□ (E□□_0부터 시작하는 지정된 워드 수의 데이터가 들어 있습니다.)	· 내용은 AUTOEM□□.IOM과 동일합니다. · 프로그램/네트워크 심볼 파일 (REPLACE.OBJ)이 메모리 카드에 있으면 이 파일의 데이터가 기동 시에 전송됩니다.	---

* 파일 암호화 기능이 유효한 경우의 확장자입니다.

참고

특수 I/O 유닛 할당 DM 영역 워드(D20000~D29599)와 CPU 버스 유닛 할당 DM 영역 워드(D30000~D31599)의 데이터를 AUTOEXEC.IOM 파일로 메모리 카드에 저장하면, 기동 시에 메모리 카드에서 특수 I/O 유닛과 CPU 버스 유닛의 셋업 데이터를 읽어 들일 수 있습니다. 이런 식으로 메모리 카드를 사용하면 설비별로 특수 I/O 유닛과 CPU 버스 유닛의 시스템 데이터 라이브러리를 관리할 수 있습니다.

기동 시 자동 전송에 필요한 대략적인 시간

사용자 프로그램과 네트워크 심볼의 사이즈	자동 전송에 필요한 시간 (전원 ON부터 운전 개시까지)
사용자 프로그램: 120K steps	9 s
사용자 프로그램: 120K steps, 네트워크 심볼: 20,000	59 s

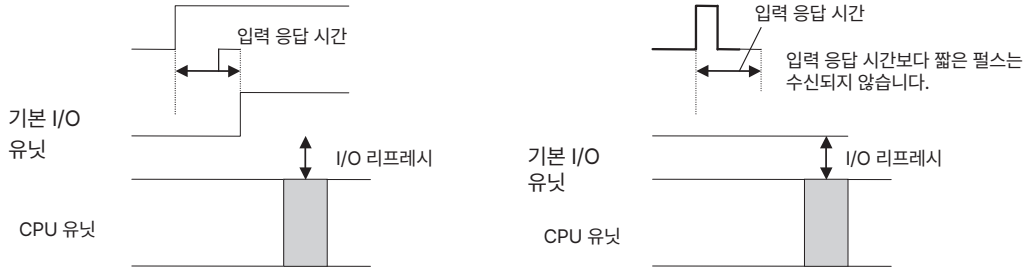
10-4 유닛 관리 기능

10-4-1 기본 I/O 유닛 관리

● 기본 I/O 유닛의 입력 응답 시간 설정

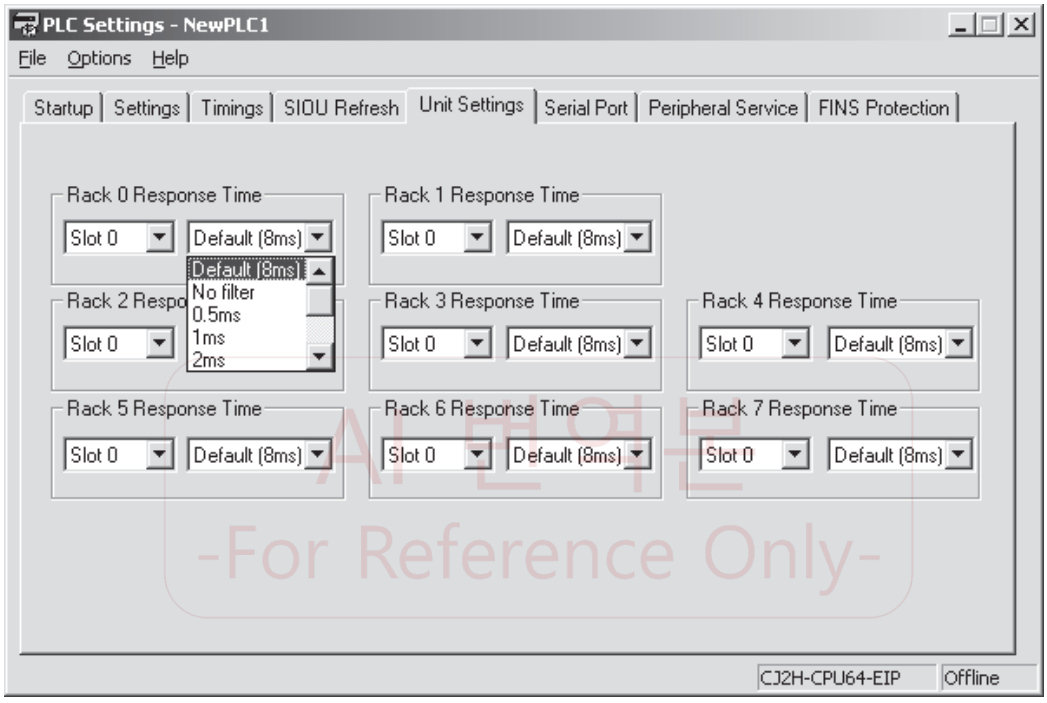
기본 I/O 유닛의 입력 응답 시간은 랙 번호와 슬롯 번호별로 설정할 수 있습니다. 입력 응답 시간을 늘리면 채터링과 노이즈의 영향이 줄어듭니다. 입력 응답 시간을 줄이면(단, 펄스 폭은 사이클 타임보다 길게 유지) 더 짧은 입력 펄스를 받을 수 있습니다.

주 사이클 타임보다 짧은 펄스는 고속 입력 유닛으로 입력할 수 있습니다. 자세한 내용은 10-2-4 고속 입력을 참조하십시오.



PLC 설정

CX-Programmer를 사용하는 경우 Unit Settings 탭 페이지에서 랙 0~7의 응답 시간을 지정하는 영역에 설정합니다.



AI 번역본
-For Reference Only-

보조 기억 영역 플래그와 워드

이름	어드레스	설명
기본 I/O 유닛의 I/O 응답 시간	A220.00~ A259.15	CJ 시리즈 기본 I/O 유닛의 현재 I/O 응답 시간이 저장됩니다.

● 기본 I/O 유닛 에러 관리

다음 표는 기본 I/O 유닛과 관련된 에러 정보를 나타냅니다.

보조 기억 영역 플래그와 워드

이름	어드레스	설명
기본 I/O 유닛 에러 플래그 (비치명 에러)	A402.12	기본 I/O 유닛에 에러가 발생하면 ON됩니다.
기본 I/O 유닛 에러 랙 번호	A408.08~ A408.15	기본 I/O 유닛에 에러가 발생했을 때(A402.12가 ON일 때) 에러가 발생한 랙 번호가 바이너리로 저장됩니다.
기본 I/O 유닛 에러 슬롯 번호	A408.00~ A408.07	기본 I/O 유닛에 에러가 발생했을 때(A402.12가 ON일 때) 에러가 발생한 슬롯 번호가 바이너리로 저장됩니다.
기본 I/O 유닛 스테이터스 영역(랙 0 슬롯 0~랙 7 슬롯 9)	A050.00~ A89.15	기본 I/O 유닛(*1)의 알람 상태(부하 단락 보호)를 나타냅니다.
인터럽트 입력 유닛 위치 에러 플래그	A405.08	CPU 랙(*2)에서 CPU 유닛 옆의 다섯 위치(슬롯 0~4) 중 하나에 인터럽트 입력 유닛이 접속되어 있지 않으면 ON됩니다.

*1 CJ2 CPU 유닛에서는 랙 0~3(A50.00~A69.15)이 지원됩니다.

*2 CJ2H-CPU6□(-EIP)에서는 슬롯 번호 0~3(CPU 유닛 오른쪽의 4유닛)이 사용됩니다.

AI 번역본
-For Reference Only-

10-4-2 CPU 버스 유닛 플래그/비트

● CPU 버스 유닛의 리스타트와 초기화

CPU 버스 유닛에서는 PLC를 OFF하지 않고 다음 리스타트 비트를 ON함으로써 CPU 유닛의 할당 DM 영역 워드에 대한 초기 설정 변경을 유효하게 할 수 있습니다. 통상적으로 이 조작은 사용자 프로그램에서 수행합니다.

보조 기억 영역 플래그와 워드

이름	어드레스	설명
CPU 버스 유닛 리스타트 비트	A501.00~ A501.15	이 비트는 CPU 버스 유닛 00~15(유닛 번호 0~F)에 대응합니다. 비트를 ON하면 해당 유닛이 리스타트됩니다.
CPU 버스 유닛 초기화 플래그	A302.00~ A302.15	이 플래그는 CPU 버스 유닛 00~15(유닛 번호 0~F)에 대응합니다. 전원이 ON되거나 해당 유닛의 리스타트 비트(A501.00~A501.15)가 ON된 후 그 유닛이 초기화되는 동안 플래그가 ON이 됩니다.

● CPU 버스 유닛 에러

다음 표는 CPU 버스 유닛과 관련된 에러 정보를 나타냅니다.

보조 기억 영역 플래그와 워드

이름	어드레스	설명
CPU 버스 유닛 번호 중복 플래그	A410.00~ A410.15	CPU 버스 유닛의 유닛 번호가 중복되면 중복 에러 플래그(A401.13)와 A410의 해당 플래그가 ON됩니다. 비트 00~15는 유닛 번호 0~F에 대응합니다.
CPU 버스 유닛 설정 에러 플래그 (비치명 에러)	A402.03	장착된 CPU 버스 유닛이 I/O 테이블에 등록된 CPU 버스 유닛과 일치하지 않으면 ON됩니다.
CPU 버스 유닛 설정 에러 유닛 번호 플래그	A427.00~ A427.15	CPU 버스 유닛 설정 에러가 발생하면 A402.03과 A427의 해당 플래그가 ON됩니다. 비트 00~15는 유닛 번호 0~F에 대응합니다.
CPU 버스 유닛 에러 플래그 (비치명 에러)	A402.07	CPU 유닛과 CPU 버스 유닛 사이의 데이터 교환에 에러가 발생하면(CPU 버스 유닛 자체의 에러를 포함) ON됩니다.
CPU 버스 유닛 에러 유닛 번호 플래그	A417.00~ A417.15	CPU 유닛과 CPU 버스 유닛 사이의 데이터 교환에 에러가 발생하면 CPU 버스 유닛 에러 플래그(A402.07)와 A417의 해당 플래그가 ON됩니다. 비트 00~15는 유닛 번호 0~F에 대응합니다.

AI 번역본
-For Reference Only-

10-4-3 특수 I/O 유닛 플래그/비트

● 특수 I/O 유닛의 리스타트와 초기화

특수 I/O 유닛에서는 PLC를 OFF하지 않고 다음 리스타트 비트를 ON함으로써 CPU 유닛의 할당 DM 영역 워드에 대한 초기 설정 변경을 유효하게 할 수 있습니다. 통상적으로 이 조작은 사용자 프로그램에서 수행합니다.

보조 기억 영역 플래그와 워드

이름	어드레스	설명
특수 I/O 유닛 리스타트 비트	A502.00~ A507.15	비트 A502.00~A507.15는 특수 I/O 유닛 0~95에 대응합니다. 비트를 ON하면 해당 유닛이 리스타트됩니다.
특수 I/O 유닛 초기화 플래그	A330.00~ A335.15	이 플래그는 특수 I/O 유닛 0~95에 대응합니다. 전원이 ON되거나 해당 유닛의 리스타트 비트(A502.00~A507.15)가 ON된 후 그 유닛이 초기화되는 동안 플래그가 ON이 됩니다. 비트 A330.00~A335.15는 유닛 번호 0~95에 대응합니다.

● 특수 I/O 유닛 에러

다음 표는 특수 I/O 유닛과 관련된 에러 정보를 나타냅니다.

보조 기억 영역 플래그와 워드

이름	어드레스	설명
특수 I/O 유닛 번호 중복 플래그	A411.00~ A416.15	특수 I/O 유닛의 유닛 번호가 중복되면 중복 에러 플래그(A401.13)와 A411~A416의 해당 플래그가 ON됩니다. 비트 A411.00~A416.15는 유닛 번호 0~95에 대응합니다.
특수 I/O 유닛 설정 에러 플래그 (비치명 에러)	A402.02	장착된 특수 I/O 유닛이 I/O 테이블에 등록된 특수 I/O 유닛과 일치하지 않으면 ON됩니다.
특수 I/O 유닛 설정 에러 유닛 번호 플래그	A428.00~ A433.15	특수 I/O 유닛 설정 에러가 발생하면 A402.02와 이들 워드의 해당 플래그가 ON됩니다. 비트 A428.00~A433.15는 유닛 번호 0~95에 대응합니다.
특수 I/O 유닛 에러 플래그 (비치명 에러)	A402.06	CPU 유닛과 특수 I/O 유닛 사이의 데이터 교환에 에러가 발생하면(특수 I/O 유닛 자체의 에러를 포함) ON됩니다.
특수 I/O 유닛 에러 유닛 번호 플래그	A418.00~ A423.15	CPU 유닛과 특수 I/O 유닛 사이의 데이터 교환에 에러가 발생하면 특수 I/O 유닛 에러 플래그(A402.06)와 이들 워드의 해당 플래그가 ON됩니다.

-For Reference Only-

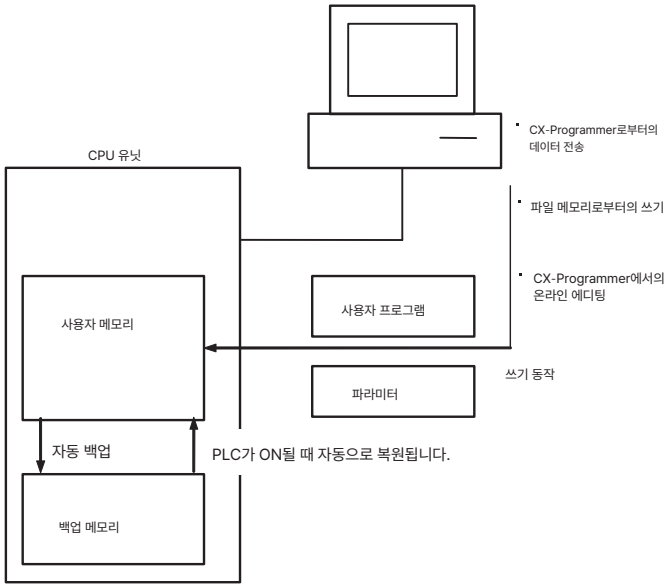
10-5 메모리 관리 기능

10-5-1 자동 백업

사용자 프로그램과 파라미터는 CPU 유닛에 쓰이거나 변경될 때마다 백업 메모리에 자동으로 백업됩니다.

- 자동으로 백업되는 데이터는 다음과 같습니다: 사용자 프로그램, 파라미터(PLC 이름, PLC 설정, I/O 테이블, 라우팅 테이블, 데이터 링크 테이블 등의 CPU 버스 유닛 데이터 포함).
- CX-Programmer로부터의 데이터 전송 조작, 온라인 에디팅, 메모리 카드 또는 EM 파일 메모리로부터의 데이터 전송 등 사용자 프로그램이나 파라미터가 CPU 유닛에 쓰일 때마다 데이터가 자동으로 백업됩니다.

- 백업 메모리에 쓰인 사용자 프로그램과 파라미터 데이터는 기동 시에 CPU 유닛의 사용자 메모리로 자동 전송됩니다.



- 백업 메모리에 데이터를 쓰는 동안에는 CPU 유닛 전면의 BKUP 표시등이 점등하고, CX-Programmer의 스테이터스 바 왼쪽에 있는 백업 상태 표시가 "Backup"이 됩니다. CX-Programmer에서 사용자 프로그램을 전송하거나 파일 메모리에서 파라미터 영역으로 데이터를 전송할 때는 백업 동작이 완료될 때까지(즉, BKUP 표시등이 소등될 때까지) CPU 유닛의 전원을 OFF하지 마십시오. 백업이 완료되기 전에 전원을 OFF한 경우에 수행되는 동작을 다음 표에 나타냅니다.

AI 번역본
-For Reference Only-

● 자동 백업 중 정전이 발생한 후의 동작

자동 백업 타이밍	해당 데이터	정전 후의 동작
프로그램을 전송하는 중	사용자 프로그램	· 데이터가 소실됩니다. · 다음에 전원을 ON했을 때 메모리 에러가 발생합니다.
온라인 에디팅 중	사용자 프로그램: · 래더 태스크	· 온라인 에디팅 중에 수행한 변경이 다음에 전원을 ON했을 때 반영됩니다. (단, 배터리가 없는 경우에는 변경이 반영되지 않고 메모리 에러가 발생합니다.)
	사용자 프로그램: · ST 태스크 · SFC 태스크 · FB 정의 · SFC 액션과 트랜지션	· 온라인 에디팅의 내용이 소실됩니다.
PLC 설정을 전송하는 중, I/O 테이블을 전송하는 중, I/O 테이블을 작성하거나 클리어하는 중, 라우팅 테이블을 전송하는 중, 또는 특수 I/O 유닛 설정을 전송하는 중.	파라미터	· 데이터가 소실됩니다. · 다음에 전원을 ON했을 때 메모리 에러가 발생합니다.

· CPU 유닛에 배터리가 장착되어 있고 래더 태스크만 온라인 에디팅 중일 때 전원을 OFF하면, 다음에 전원을 ON했을 때 정전 이전의 상태가 복원됩니다. 다만 다음 사항에 주의하십시오.

  · CPU 유닛의 기동에 더 많은 시간이 걸립니다.
  · CPU 유닛에 배터리가 있더라도, CPU 유닛을 장시간 전원이 꺼진 상태로 둘 경우에는 전원을 OFF하기 전에 백업 동작이 완료되었는지 반드시 확인하십시오.

· 데이터 백업에 필요한 시간(BKUP 표시등이 점등하는 시간)은 다음 표와 같이 사용자 프로그램의 사이즈에 따라 달라집니다.

사용자 프로그램 사이즈	백업 처리 시간	
	MONITOR 모드	PROGRAM 모드
100 Ksteps	20 s (사이클 타임 3 ms)	3 s
250 Ksteps	47 s (사이클 타임 7 ms)	6 s
400 Ksteps	75 s (사이클 타임 11 ms)	10 s

주 CPU 유닛에 전원이 공급되면 BKUP 표시등이 점등합니다.

사용 설명서
-For Reference Only-

사용상의 주의

・사용자 프로그램과 파라미터 데이터는 CPU 유닛에 쓰일 때 플래시 메모리에 자동으로 백업됩니다. 다만 I/O 메모리(DM, EM, HR, AR 영역 포함)는 플래시 메모리에 쓰이지 않습니다. (AR 영역의 일부 어드레스 내용도 유지됩니다.)
DM, EM, HR, AR, 카운터 영역은 배터리에 의해 정전 중에도 유지할 수 있습니다. 배터리 에러가 있으면 정전 후 이들 영역의 내용이 정확하지 않을 수 있습니다. DM, EM, 유지, 보조 기억 영역의 내용을 외부 출력 제어에 사용하는 경우, 배터리 에러 플래그(A402.04)가 ON일 때 부적절한 출력이 나가지 않도록 방지하십시오.

・CJ3 CPU 유닛에서는 정전이 발생해도 DM, EM, HR, AR, 카운터 영역이 배터리 없이 유지됩니다.

・CPU 유닛의 백업 메모리에 에러가 검출되면 백업 메모리 에러(비치명)가 발생합니다. 이 에러가 발생하면 다음에 전원을 ON했을 때 데이터가 복원되지 않고 메모리 에러가 발생할 수 있습니다. 따라서 전원을 OFF하기 전에 데이터를 저장할 것을 권장합니다.

참고

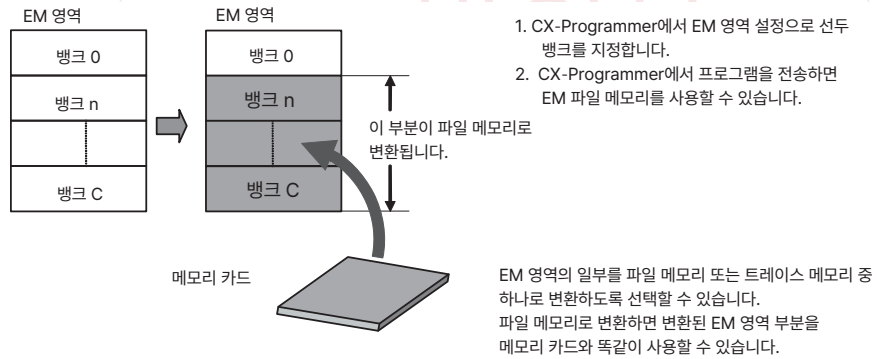
통상적인 데이터 전송(PLC - Transfer) 이외의 전송 조작으로 CX-Programmer에서 데이터를 백업하는 경우, CX-Programmer가 Memory Backup Status 윈도우에 백업 상태를 표시합니다. 이 윈도우를 표시하려면 PLC Properties에서 백업 상태 대화 상자를 표시하는 설정을 체크하고 View 메뉴에서 그 윈도우를 선택해야 합니다.
통상적인 전송 조작에서는 프로그램과 그 밖의 데이터의 전송 상태 뒤에 전송 윈도우에 백업 상태가 표시됩니다.

● 보조 기억 영역 플래그

이름	어드레스	의미
백업 메모리 에러 플래그	A315.15	백업 메모리에 이상이 발생하면 ON됩니다.

10-5-2 EM 파일 메모리 기능

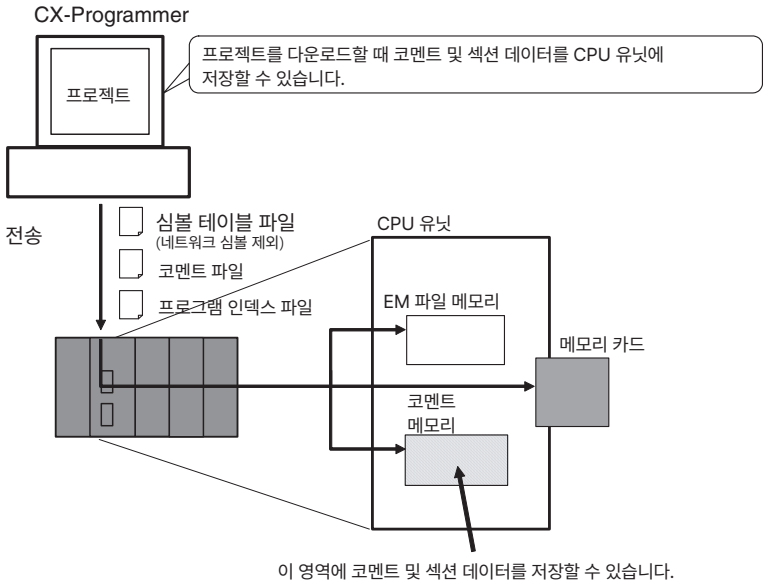
CPU 유닛에서 파일을 저장할 때 메모리 카드 대신 EM 영역을 사용할 수 있습니다. 지정한 선두 뱅크 이후의 뱅크가 파일 메모리로 사용됩니다. EM 영역 뱅크를 파일 메모리로 변환하는 방법은 7-1-2 파일 메모리 초기화를 참조하십시오.



10-5-3 코멘트 메모리

CPU 유닛 내부의 백업 메모리에는 코멘트 메모리가 마련되어 있습니다. 메모리 카드와 EM 파일 메모리를 모두 사용할 수 없는 경우에도 다음 코멘트/섹션 데이터를 코멘트 메모리에 저장하고 읽어 들일 수 있습니다.

- 심볼 테이블 파일(CX-Programmer 심볼과 I/O 코멘트 포함, 네트워크 심볼은 제외)
- 코멘트 파일(CX-Programmer 렁 코멘트와 그 밖의 코멘트)
- 프로그램 인덱스 파일(CX-Programmer 섹션 이름, 섹션 코멘트, 프로그램 코멘트)

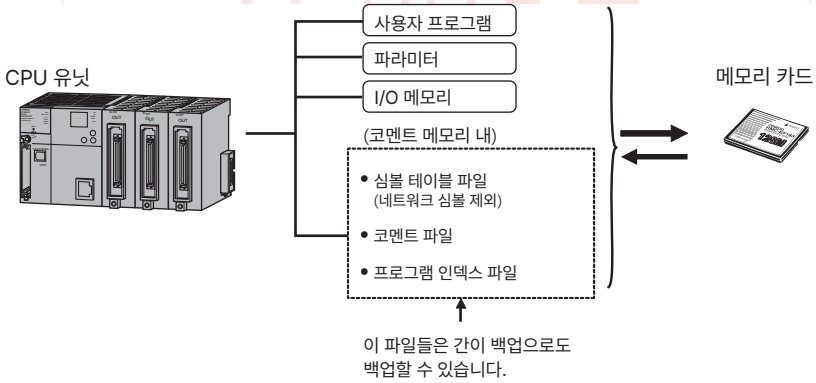


프로젝트를 다운로드할 때 코멘트 데이터와 섹션 데이터의 전송처로 다음 저장 위치 중 하나를 선택할 수 있습니다.

- 코멘트 메모리(CPU 유닛의 백업 플래시 메모리 내)
- 메모리 카드
- EM 파일 메모리

간이 백업 조작을 실행하면 코멘트 메모리에 저장된 다음 파일을 메모리 카드에 백업하거나, 메모리 카드에서 코멘트 메모리로 복원할 수 있습니다.

- 심볼 테이블 파일(CX-Programmer 심볼과 I/O 코멘트 포함, 네트워크 심볼은 제외)
- 코멘트 파일(CX-Programmer 렁 코멘트와 그 밖의 코멘트)
- 프로그램 인덱스 파일(CX-Programmer 섹션 이름, 섹션 코멘트, 프로그램 코멘트)

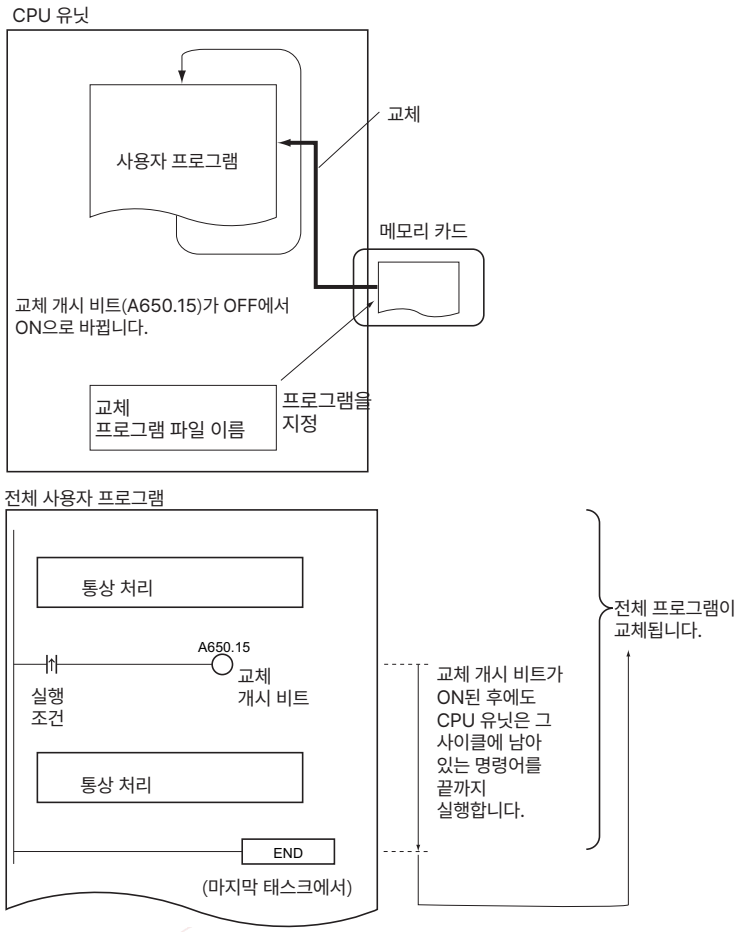


이로써 에러가 발생한 경우나 동일 사양의 CPU 유닛을 추가할 때 CX-Programmer 없이 I/O 코멘트를 포함한 CPU 유닛 내의 모든 데이터를 백업 · 복원할 수 있습니다.

10-5-4 운전 중 전체 프로그램 교체

개요

교체 개시 비트(A650.15)를 ON하면 RUN 또는 MONITOR 모드의 운전 중에 전체 태스크 프로그램을 교체할 수 있습니다. 보조 기억 영역에 미리 지정한 파일 이름의 프로그램/네트워크 심볼 파일(.OBJ/.OBS)을 메모리 카드에서 읽어 들여, 현재 사이클이 끝날 때 프로그램을 교체합니다.



프로그램 실행이 정지된 상태(PROGRAM 모드)에서도 CX-Programmer에서 교체 개시 비트를 ON하여 프로그램을 교체할 수 있습니다.

- 교체용 프로그램 파일은 EM 파일 메모리에서 읽어 들일 수 없습니다.
- 교체 개시 비트(A650.15)는 프로그램 내의 어느 위치(프로그램 어드레스)에서든 ON할 수 있습니다.
- 교체 개시 비트가 OFF에서 ON으로 바뀐 후에도 CPU 유닛은 그 사이클에 남아 있는 명령어를 실행합니다.
- 프로그램을 교체하는 동안에는 프로그램이 실행되지 않습니다. 프로그램 교체가 끝나면 CPU 유닛을 PROGRAM 모드에서 RUN 또는 MONITOR 모드로 전환했을 때와 똑같이 운전이 다시 시작됩니다.
- 프로그램은 교체 개시 비트가 OFF에서 ON으로 바뀐 사이클의 마지막, 즉 프로그램의 마지막 태스크에서 END(001)가 실행된 후에 교체됩니다.

사내 번역부
For Reference Only

프로그램 교체 중 CPU 동작

프로그램 교체 중 CPU 유닛의 동작은 다음과 같습니다:

- 프로그램 실행: 정지
- 사이클 타임 감시: 감시하지 않음

● 프로그램 교체 중 및 교체 후에 계속되는 동작

- IOM 유지 비트(A500.12)가 ON이면 다음 메모리 영역의 데이터가 유지됩니다: CIO 영역, 워크 영역(W), 타이머 완료 플래그(T), 인덱스 레지스터(IR), 데이터 레지스터(DR), 현재 EM 뱅크 번호.

주 타이머 현재값은 프로그램 교체 중에 클리어됩니다.

- 프로그램을 전송할 때 IOM 유지 비트가 ON이면, 프로그램 교체 전에 출력되고 있던 부하는 교체 후에도 계속 출력됩니다. 프로그램 교체 후에 외부 부하가 올바르게 동작하는지 반드시 확인하십시오.
- 강제 상태 유지 비트(A500.13)가 ON이면 강제 세트 · 강제 리셋된 비트의 상태가 프로그램 교체 중에도 유지됩니다.
- 인터럽트는 마스크됩니다.
- 데이터 트레이스가 수행 중이면 정지됩니다.
- 미분 플래그는 IOM 유지 비트가 ON이든 OFF이든 초기화됩니다.

● 프로그램 교체 후의 동작

- 사이클릭 태스크의 상태는 그 운전 개시 속성에 따라 달라집니다. (PLC를 PROGRAM 모드에서 RUN/MONITOR 모드로 전환했을 때와 같은 상태가 됩니다.)
- 프로그램 실행이 재개된 후 1사이클 동안 최초 사이클 플래그(A200.11)가 ON이 됩니다. (PLC를 PROGRAM 모드에서 RUN/MONITOR 모드로 전환했을 때와 같은 상태입니다.)

순서

1 프로그램 파일이 들어 있는 메모리 카드를 CPU 유닛에 장착합니다.

2 보조 기억 영역에 프로그램 파일 이름(A654~A657)과 프로그램 패스워드(A651)를 설정한 후, 교체 개시 비트(A650.15)를 ON합니다.



참고

프로그램 교체 중에도 I/O 메모리 데이터의 상태를 유지하려면 IOM 유지 비트(A500.12)를 ON하십시오.

프로그램 교체 중에도 강제 세트 · 강제 리셋된 비트의 상태를 유지하려면 강제 상태 유지 비트(A500.13)를 ON하십시오.

AI 번역본
-For Reference Only-



안전상의 요점

프로그램 교체 전에 IOM 유지 비트(A500.12)가 ON이면 I/O 메모리의 비트 상태가 프로그램 교체 후에도 유지됩니다. 동일한 I/O 메모리 데이터로 외부 부하가 올바르게 동작하는지 반드시 확인하십시오.

프로그램 교체 전에 강제 상태 유지 비트(A500.13)가 ON이면 I/O 메모리의 강제 상태가 프로그램 교체 후에도 유지됩니다. 동일한 강제 상태로 외부 부하가 올바르게 동작하는지 반드시 확인하십시오.

교체 파일

지정한 프로그램/네트워크 심볼 파일을 메모리 카드에서 읽어 들입니다.

파일	파일 이름과 확장자	보조 기억 영역의 교체 개시 비트(A650.15)를 ON하기	교체 파일 이름 지정 (********)
프로그램/네트워크 심볼 파일	********.OBJ ********.OBS(*)	CPU 유닛의 사용자 프로그램 교체	프로그램 교체 전에 교체 프로그램 파일 이름을 A654~A657에 씁니다.

* 파일 암호화 기능이 유효한 경우의 확장자입니다.

프로그램 교체에 필요한 조건

운전 중에 프로그램을 교체하려면 다음 조건이 필요합니다.

- 프로그램 파일 이름 워드(A654~A657)에 지정한 프로그램/네트워크 심볼 파일이 메모리 카드의 루트 디렉터리에 존재할 것.
- CPU 유닛이 메모리 카드를 검출했을 것. (A343.15가 ON이어야 합니다.)
- 치명 에러가 발생하지 않았을 것.
- 파일 메모리 조작이 실행 중이 아닐 것. (A343.13이 OFF여야 합니다.)
- 프로그램 영역에 데이터를 쓰는 중이 아닐 것.
- 액세스 권한을 사용할 수 있을 것. (예를 들어 CX-Programmer에서 PLC로 데이터를 전송하는 중이 아닐 것.)

주 프로그램은 어느 동작 모드에서든 전송할 수 있습니다.

- 교체할 프로그램의 파일 확장자 대응은 다음과 같습니다.
  파일 암호화 설정과 패스워드가 일치하지 않으면 프로그램 교체는 실행되지 않습니다.

		교체 대상 CPU 유닛의 파일 암호화 설정	
		무효	유효
교체할 프로그램의 파일 확장자	.OBJ(암호화되지 않은 파일)	지원	지원하지 않음
	.OBS(암호화된 파일)	지원하지 않음	지원 (*)

* 기동 시 자동 전송은 파일 암호화 설정의 패스워드가 일치하는 경우에만 가능합니다.

AI 번역본
-For Reference Only-

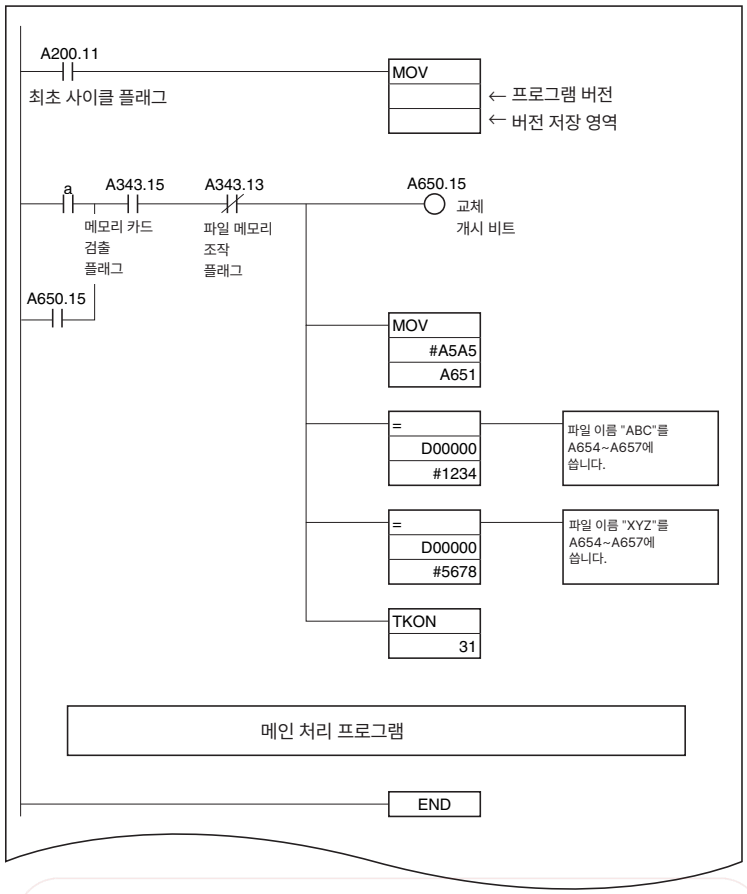
프로그램 예

● 프로그램 예 1

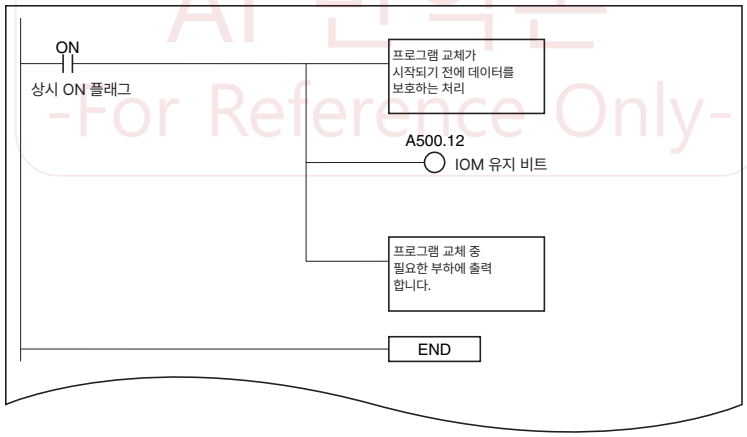
다음 예에서는 프로그램/네트워크 심볼 파일 ABC.OBJ와 XYZ.OBJ가 메모리 카드에 저장되어 있고, D0의 값에 따라 둘 중 하나의 프로그램이 선택됩니다. ABC.OBJ를 선택하려면 D0을 #1234로, XYZ.OBJ를 선택하려면 #5678로 설정합니다.

IOM 유지 비트 설정을 포함하여 프로그램 교체 전에 필요한 처리를 수행하기 위해 별도의 태스크가 기동됩니다.

메인 태스크(사이클릭 태스크 0)



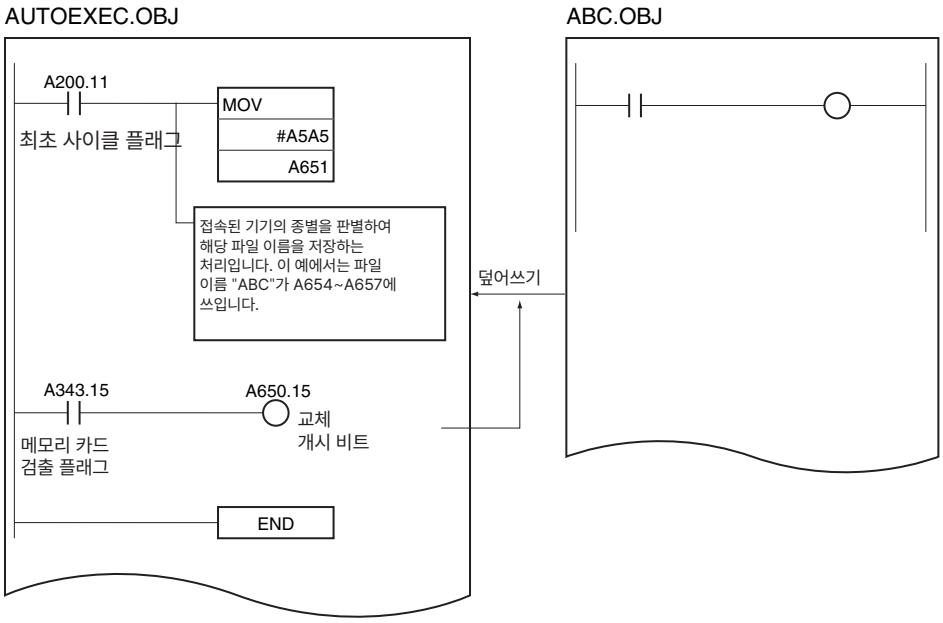
프로그램 교체 중 데이터를 보호하는 태스크
(사이클릭 태스크 31, 기동 시 스탠바이 상태)



AT 번역본
-For Reference Only-

● 프로그램 예 2

이 예에서는 여러 기기용 프로그램/네트워크 심볼 파일과 기동 시 자동 전송용 프로그램/네트워크 심볼 파일(AUTOEXEC.OBJ 또는 REPLACE.OBJ)이 메모리 카드에 저장되어 있습니다. PLC를 ON하면 기동 시 자동 전송 파일이 읽히고, 그 후 그 프로그램이 다른 기기용 프로그램 파일로 교체됩니다.



AI 번역본
-For Reference Only-

보조 기억 영역 플래그와 워드

이름	어드레스	동작
파일 메모리 조작 플래그	A343.13	다음 조작 중 하나라도 실행 중이면 ON입니다. 어느 것도 실행 중이 아니면 OFF입니다. ・메모리 카드 검출 ・CPU 유닛이 CMND(490)을 사용하여 자기 자신에게 FINS 커맨드를 송신한 경우. ・FREAD(700) 또는 FWRIT(701)이 실행 중. ・보조 기억 영역 제어 비트(A650.15)를 사용하여 프로그램을 덮어쓰는 중. ・간이 백업 조작이 수행 중.
메모리 카드 검출 플래그	A343.15	메모리 카드가 검출되면 ON입니다. 메모리 카드가 검출되지 않으면 OFF입니다.
IOM 유지 비트	A500.12	CPU 유닛을 PROGRAM 모드와 RUN 또는 MONITOR 모드 사이에서 전환하거나 전원을 ON할 때 I/O 메모리의 상태를 유지하려면 이 비트를 ON하십시오. ON: 동작 모드를 변경할 때 I/O 메모리 상태가 유지됩니다. OFF: 동작 모드를 변경할 때 I/O 메모리 상태가 클리어됩니다.
강제 상태 유지 비트	A500.13	CPU 유닛을 PROGRAM 모드와 RUN 또는 MONITOR 모드 사이에서 전환하거나 전원을 ON할 때 강제 세트 또는 강제 리셋된 비트의 상태를 유지하려면 이 비트를 ON하십시오.
교체 완료 코드	A650.00~ A650.07	・프로그램 교체가 정상 완료된 경우의 코드(A650.14 OFF): 01 Hex: 프로그램/네트워크 심볼 파일(.OBJ)이 프로그램을 교체했습니다. ・프로그램 교체가 완료되지 않은 경우의 코드(A650.14 ON): 00 hex: 치명 에러가 발생했습니다. 01 hex: 메모리 에러가 발생했습니다. 11 hex: 프로그램이 쓰기 보호되어 있습니다. 12 hex: A651의 프로그램 패스워드가 올바르지 않습니다. 21 hex: 메모리 카드가 장착되어 있지 않습니다. 22 hex: 지정한 파일이 존재하지 않습니다. 23 hex: 지정한 파일이 너무 큽니다(메모리 에러). 31 hex: 다음 조작 중 하나가 수행되고 있었습니다: - 파일 메모리 조작이 수행되고 있었습니다. - 프로그램을 쓰는 중이었습니다. - 동작 모드를 변경하는 중이었습니다.
교체 에러 플래그	A650.14	A650.15가 OFF에서 ON으로 바뀐 후 프로그램을 교체하려 할 때 에러가 발생하면 ON됩니다. 다음에 A650.15가 다시 OFF에서 ON으로 바뀔 때 OFF됩니다.

AI 번역
-For Reference Only-

이름	어드레스	동작
교체 개시 비트	A650.15	프로그램 패스워드(A651)를 A5A5 hex로 설정하여 이 비트가 유효해진 경우, 이 비트가 OFF에서 ON으로 바뀌면 프로그램 교체가 시작됩니다. 프로그램 교체 중에 이 비트를 OFF에서 ON으로 다시 바꾸지 마십시오. 이 비트는 프로그램 교체가 완료되거나(정상 또는 에러) 전원이 ON될 때 자동으로 OFF됩니다. 이 비트의 상태를 CX-Programmer, PT 또는 호스트 컴퓨터에서 읽어 프로그램 교체가 완료되었는지 여부를 판단할 수 있습니다.
프로그램 패스워드	A651	프로그램 교체를 유효하게 하려면 이 워드에 패스워드를 씁니다. A5A5 hex: 교체 개시 비트(A650.15)를 유효하게 합니다. 그 밖의 값: 교체 개시 비트(A650.15)를 무효로 합니다. 이 비트는 프로그램 교체가 완료되거나(정상 또는 에러) 전원이 ON될 때 자동으로 OFF됩니다.
프로그램 파일 이름	A654~A657	프로그램 교체를 시작하기 전에 교체 프로그램 파일의 이름을 이 워드에 ASCII로 씁니다. 8문자의 파일 이름만 쓰면 되며, OBJ 확장자는 자동으로 붙습니다. 문자는 A654부터 순서대로 (최상위 바이트부터) 씁니다. 파일 이름이 8문자보다 짧은 경우, 남은 바이트는 스페이스 코드(20 hex)로 채웁니다. 파일 이름 자체에는 NULL 문자나 스페이스를 포함하지 마십시오. 다음 예는 프로그램 파일 ABC.OBJ의 데이터를 나타냅니다: 15 0 A654 41 42 A655 43 20 A656 20 20 A657 20 20

AI 번역본
-For Reference Only-

10-5-5 전 메모리 클리어

이 기능은 CPU 유닛의 내부 메모리를 초기화합니다.

다음 내부 메모리를 디폴트 설정으로 초기화할 수 있습니다.

- 프로그램 영역
- I/O 메모리 영역
- 파라미터 영역
  PLC 설정 영역, 주변 기기 영역, I/O 테이블 영역, CPU 버스 유닛 영역
- 파일 암호화 설정과 패스워드

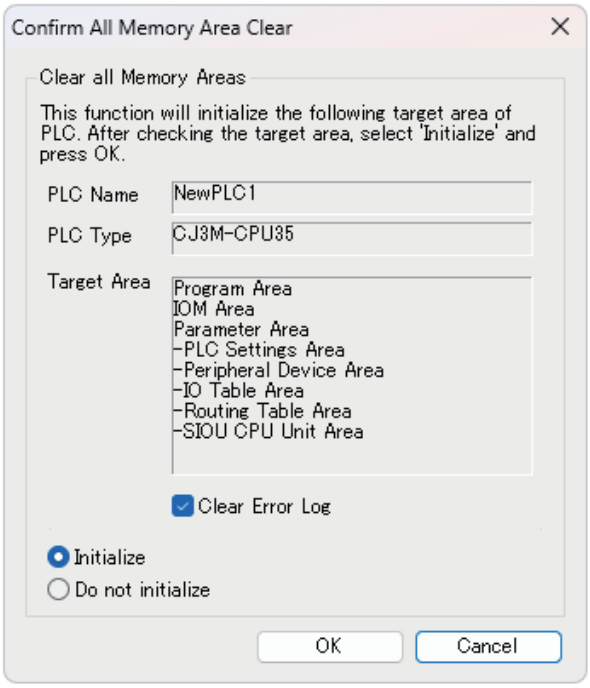
주 1 딥 스위치 1이 ON으로 설정된 경우(사용자 프로그램 쓰기 금지), 전 메모리 클리어를 실행해도 전체 사용자 프로그램과 파라미터 영역의 데이터는 클리어되지 않습니다.

2 확장 FINS 쓰기 액세스 로그와 조작 로그는 전 메모리 클리어로 클리어되지 않습니다. 확장 FINS 쓰기 액세스 로그를 클리어하는 방법은 10-6-5 FINS 쓰기 액세스 로그를, 조작 로그를 클리어하는 방법은 10-6-6 조작 로그 기능을 참조하십시오.

3 내장 EtherNet/IP의 설정은 CPU 유닛의 전 메모리 클리어로 초기화되지 않습니다. EtherNet/IP 유닛의 내장 메모리를 초기화하는 순서는 SYSMAC CS 및 CJ 시리즈 EtherNet/IPTM 유닛 오퍼레이션 매뉴얼(Cat. No. W465)의 17-4 유닛 초기화를 참조하십시오.

순서

1 PLC 메뉴에서 Clear All Memory를 선택합니다.



2 Execute Initialization 옵션과 Clear Error Log 체크 박스를 선택하고 OK 버튼을 클릭합니다.

10-6 보안 기능

CJ 시리즈 사용자 프로그램에는 다음 보호를 적용할 수 있습니다.

10-6-1 딥 스위치에 의한 쓰기 보호

사용자 프로그램과 파라미터 영역의 데이터(예: PLC 설정, I/O 테이블)를 CX-Programmer에서 덮어쓸 수 없도록 쓰기 보호를 설정할 수 있습니다. 이 기능으로 프로그램이 부주의하게 덮어쓰이는 것을 방지할 수 있습니다.

쓰기 보호를 설정하려면 CPU 유닛의 딥 스위치 핀 1을 ON하십시오. 이 기능이 유효한 경우에도 CX-Programmer로 프로그램을 읽고 표시할 수는 있습니다.

● CPU 유닛 전면의 딥 스위치

핀 No.	이름	설정
1	사용자 프로그램 메모리 쓰기 보호	ON: 보호 ON OFF: 보호 OFF

10-6-2 패스워드에 의한 읽기 보호

패스워드에 의한 읽기 보호를 설정할 수 있습니다.

보호된 프로그램의 사용자 프로그램, 특정 태스크, 펑션 블록은 CX-Programmer에서 패스워드를 입력하지 않으면 표시하거나 편집할 수 없습니다.

● 확장 패스워드에 의한 읽기 보호

UM 읽기 보호와 태스크 읽기 보호의 패스워드가 강화되어 사용자의 설계 자산을 견고하게 보호합니다.

또한 잘못된 패스워드를 연속으로 5회 입력하면, 그 후 2시간 이상 전원이 계속 공급되지 않는 한 패스워드 입력이 다시 접수되지 않습니다.

주 이 기능은 CJ2H CPU 유닛 버전 1.4 이전 및 CJ2M CPU 유닛 버전 2.0 이전에서는 사용할 수 없습니다.

AI 번역본
-For Reference Only-

보호의 종류

패스워드에 의한 읽기 보호에는 다음 종류가 있습니다.

● UM 읽기 보호

CPU 유닛에서 사용자 프로그램을 읽는 것이 금지됩니다.

● 태스크 읽기 보호

하나 이상의 임의의 태스크에 대해 CX-Programmer에서의 태스크 표시가 금지됩니다. CPU 유닛에서 사용자 프로그램을 읽을 수는 있지만, 보호된 태스크는 CX-Programmer에 표시되지 않습니다.



참고

・프로그램 내의 개별 태스크가 읽기 보호되어 있어도 전체 프로그램을 다른 CPU 유닛으로 전송할 수 있습니다. 보호된 태스크에 대해서는 태스크 읽기 보호가 계속 유효합니다.

・CX-Programmer로 컴퓨터 메모리의 사용자 프로그램과 CPU 유닛의 사용자 프로그램을 비교할 때는 보호된 태스크도 비교됩니다.

● 펑션 블록 보호

원하는 펑션 블록에 다음 보호를 적용할 수 있습니다.

・쓰기/표시 보호
CX-Programmer에서 펑션 블록의 표시가 금지되므로 변경도 할 수 없습니다.

・쓰기 보호
CX-Programmer에서 펑션 블록을 표시할 수는 있지만 변경은 할 수 없습니다.



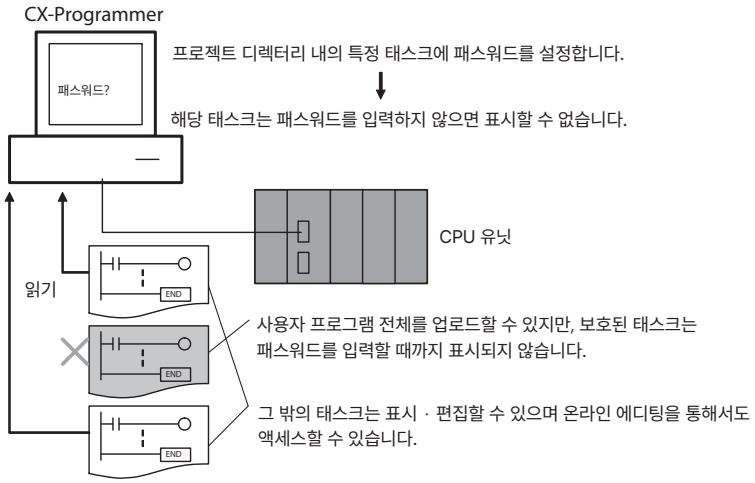
참고

태스크 읽기 보호를 적용하더라도 펑션 블록을 포함한 사용자 프로그램을 사용하면 펑션 블록 정의를 읽을 수 있습니다. 펑션 블록을 읽기 보호하려면 펑션 블록 보호를 사용하십시오.

AI 번역본
-For Reference Only-

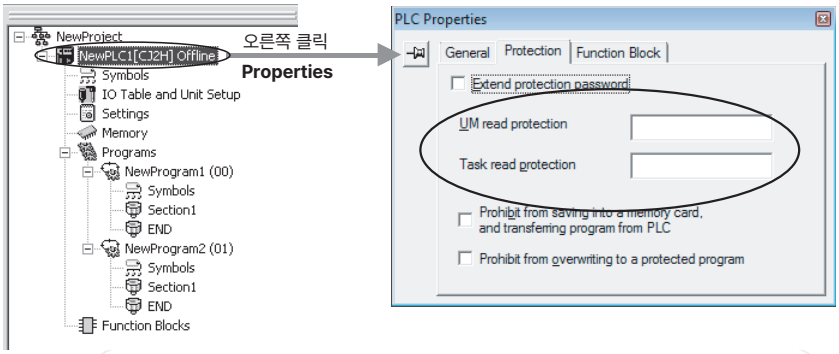
참고

UM 읽기 보호가 이미 설정되어 있으면 태스크 읽기 보호를 설정할 수 없습니다. 다만 태스크 읽기 보호를 설정한 후에 UM 읽기 보호를 설정하는 것은 가능합니다.



조작 순서

1 PLC Properties 윈도우의 Protection 탭 페이지를 표시하고 보호용 패스워드를 등록합니다.



· UM read protection 필드
UM 읽기 보호용 패스워드를 입력합니다.

· Task read protection 필드
태스크 읽기 보호용 패스워드를 입력합니다.

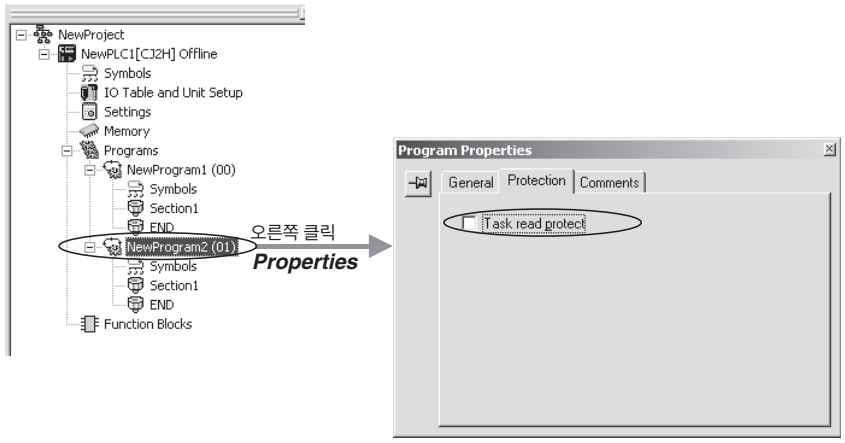
· Extend Protection Passwords
패스워드에 입력할 수 있는 문자 수의 상한이 최대 8자에서 최대 16자로 변경됩니다.
PLC Properties 대화 상자에서 Protection 탭을 클릭하고 Extend protection password 체크 박스를 선택한 후 패스워드를 입력합니다.
패스워드 문자열 길이의 상한은 다음 표와 같습니다.

Extend Protection Passwords	입력 문자 수	입력 문자 종류
선택함*	최대 8자	영숫자 문자
선택 안 함*	최대 16자	

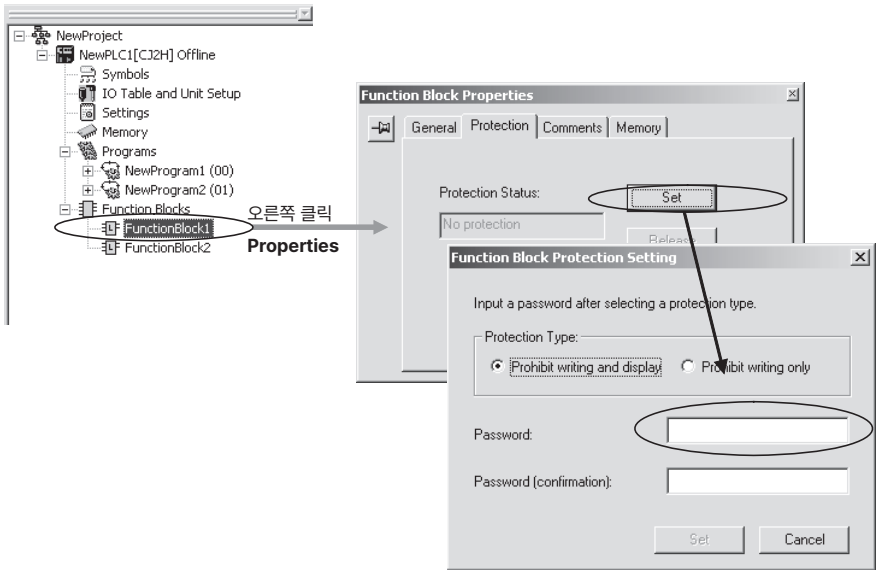
* "Extend Protection Passwords"는 CJ2H CPU 유닛 버전 1.4 이전 및 CJ2M CPU 유닛 버전 2.0 이전에서는 사용할 수 없습니다.

OMRON 번역본
Reference Only

2 태스크에 읽기 보호를 설정하려면 태스크를 선택한 후 Program Properties 대화 상자에서 Task read protect 체크 박스를 선택합니다.



3 펑션 블록(FB) 정의에 읽기 보호를 적용하려면 보호할 펑션 블록을 선택하고, 펑션 블록 프로퍼티에서 Set 버튼을 클릭한 후 Function Block Protections Setting 필드에 패스워드를 입력합니다.



4 온라인으로 접속하고 아래의 a 또는 b 중 한쪽 순서를 실행합니다.

(a) 프로그램 전송과 패스워드 보호 설정:

PLC - Transfer - To PLC을 선택하여 프로그램을 전송합니다. 순서 2에서 등록한 태스크가 패스워드로 보호됩니다.

(b) 프로그램을 전송하지 않고 패스워드 보호를 설정:

PLC - Protection - Set Password를 선택하고 OK 버튼을 클릭합니다. 순서 2에서 등록한 태스크가 패스워드로 보호됩니다.

AI 번역본 For Reference Only

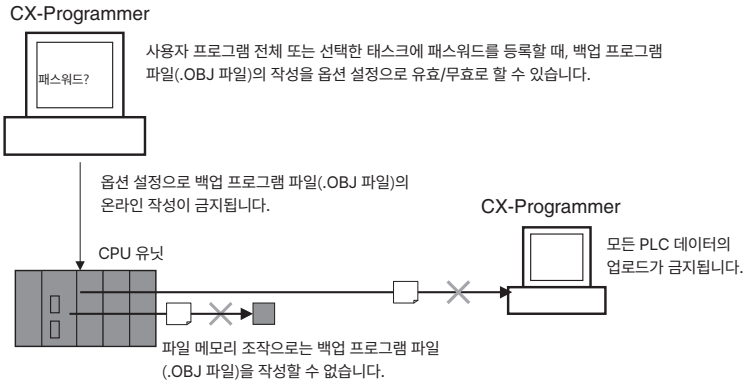
5 패스워드로 읽기 보호를 해제하려면 CX-Programmer의 Release Password 대화 상자에 패스워드를 입력합니다. 패스워드를 연속 5회 잘못 입력하면, 그 후 2시간 이상 전원을 공급하지 않는 한 패스워드 입력이 다시 접수되지 않습니다. (이 기능은 CJ2H CPU 유닛 버전 1.4 이전 및 CJ2M CPU 유닛 버전 2.0 이전에서는 사용할 수 없습니다).

메모리 카드 저장의 유효/무효 설정

● 개요

CX-Programmer에서 프로그램 전체 또는 개별 태스크에 읽기 보호를 설정한 경우, .OBJ 프로그램/네트워크 심볼 파일의 작성 및 백업을 유효/무효로 하는 옵션을 설정할 수 있습니다. 이 설정으로 프로그램/네트워크 심볼 파일의 작성 및 백업을 금지하면, 파일 메모리 조작으로 프로그램/네트워크 심볼 파일(.OBJ 파일)을 작성할 수 없게 됩니다. (이 설정은 메모리 카드/EM 파일 메모리로의 온라인 전송과, CX-Programmer로 업로드한 PLC 데이터의 오프라인 저장을 모두 금지합니다.)

파일 메모리 프로그램 파일의 작성을 무효로 하면 사용자 프로그램의 부정 복사를 방지하는 데 도움이 됩니다.



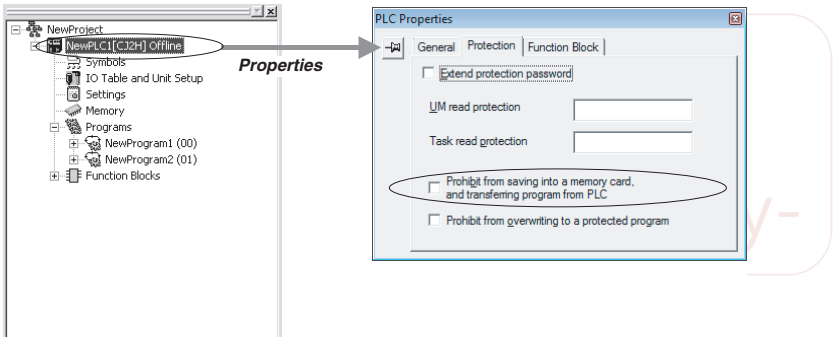
・프로그램/네트워크 심볼 파일의 작성이 금지되어 있어도 간이 백업 조작은 실행할 수 있지만, 백업 프로그램/네트워크 심볼 파일(BACKUP.OBJ)은 작성되지 않습니다.

・프로그램 읽기 보호가 유효하지 않은 경우에는 프로그램을 복사할 수 있습니다.

・프로그램/네트워크 심볼 파일 작성의 유효/무효 설정은 프로그램을 CPU 유닛으로 전송하지 않으면 반영되지 않습니다. 이 설정을 변경한 후에는 반드시 프로그램을 전송하십시오.

● 조작 순서

1 UM read protection 박스 또는 Task read protection 박스에 패스워드를 등록하려면 CX-Programmer의 PLC Properties 대화 상자에 있는 Protection 탭 페이지에서 Prohibit from saving into a memory card, and transferring program from PLC 체크 박스를 선택합니다.



2 PLC - Transfer - To PLC을 선택하여 프로그램을 전송하거나, PLC - Protection - Set Password를 선택하고 OK 버튼을 클릭합니다.

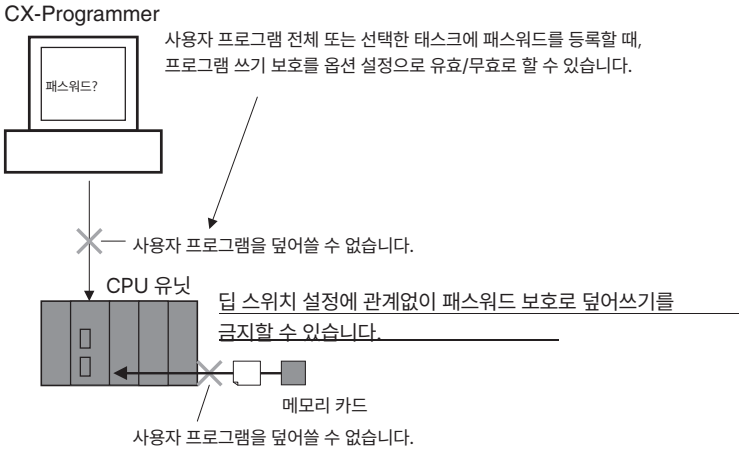
● 용도

・이 옵션을 사용하면 패스워드를 이용해 프로그램이 PLC 밖으로 전송되는 것을 방지할 수 있습니다.

프로그램 덮어쓰기의 유효/무효 설정

● 개요

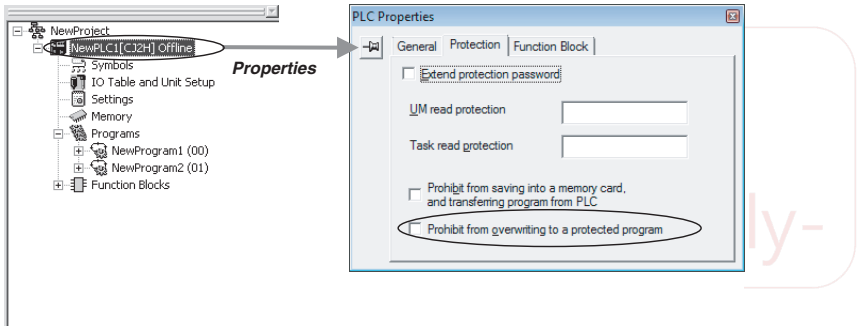
앞에서 설명한 딥 스위치를 사용하는 방법 외에도, 프로그램 전체 또는 선택한 태스크에 패스워드를 등록할 때 CX-Programmer에서 쓰기 보호 옵션을 선택하면 프로그램(또는 선택한 태스크)에 쓰기 보호를 걸 수 있습니다. 쓰기 보호 설정은 프로그램의 무단 또는 실수에 의한 덮어쓰기를 방지할 수 있습니다.



- 패스워드 등록 시 이 옵션을 선택하여 선택한 태스크 또는 프로그램에 쓰기 보호를 걸면, 패스워드로 보호된 태스크(프로그램)만 덮어쓰기로부터 보호됩니다. 그 밖의 태스크나 프로그램은 온라인 에디팅, 태스크 다운로드 등의 조작으로 여전히 덮어쓸 수 있습니다.
- 프로그램 읽기 보호가 유효하지 않은 경우에는 모든 태스크(프로그램)를 덮어쓸 수 있습니다.
- 프로그램 파일 작성의 유효/무효 설정은 프로그램을 CPU 유닛으로 전송하지 않으면 반영되지 않습니다. 이 설정을 변경한 후에는 반드시 프로그램을 전송하십시오.

● 조작 순서

1 UM read protection 박스 또는 Task read protection 박스에 패스워드를 등록하려면 CX-Programmer의 PLC Properties 대화 상자에 있는 Protection 탭 페이지에서 Prohibit from overwriting to a protected program 옵션을 선택합니다.



2 PLC - Transfer - To PLC을 선택하여 프로그램을 전송하거나, PLC - Protection - Set Password를 선택하고 OK 버튼을 클릭합니다.

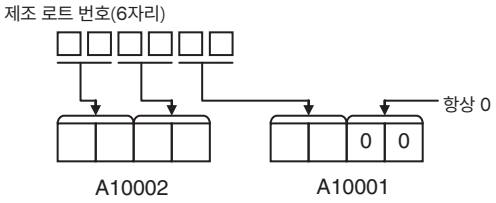
패스워드 보호 관련 보조 기억 영역 플래그 및 비트

이름	비트 어드레스	설명
UM 읽기 보호 플래그	A99.00	PLC(사용자 프로그램 전체)에 읽기 보호가 걸려 있는지 여부를 나타냅니다. OFF: UM 읽기 보호가 설정되어 있지 않습니다. ON: UM 읽기 보호가 설정되어 있습니다.
태스크 읽기 보호 플래그	A99.01	선택한 프로그램 태스크에 읽기 보호가 걸려 있는지 여부를 나타냅니다. OFF: 태스크 읽기 보호가 설정되어 있지 않습니다. ON: 태스크 읽기 보호가 설정되어 있습니다.
읽기 보호 대상의 프로그램 쓰기 보호	A99.02	프로그램의 덮어쓰기를 방지하기 위해 쓰기 보호 옵션이 선택되어 있는지 여부를 나타냅니다. OFF: 덮어쓰기 허가 ON: 덮어쓰기 금지(쓰기 보호)
프로그램 백업 유효/무효 비트	A99.03	UM 읽기 보호 또는 태스크 읽기 보호가 설정되어 있을 때 백업 프로그램/네트워크 심볼 파일(.OBJ 파일)을 작성할 수 있는지 여부를 나타냅니다. OFF: 백업 프로그램 파일 작성 허가 ON: 백업 프로그램 파일 작성 금지

10-6-3 제조 로트 번호에 의한 프로그램 동작 보호

보조 기억 영역의 A10001 및 A10002 워드에 저장된 제조 로트 번호를 사용하여 프로그램의 동작을 보호할 수 있습니다. 제조 로트 번호는 사용자가 변경할 수 없습니다.

・제조 로트 번호는 6자리입니다. 왼쪽 4자리는 A10002에, 오른쪽 2자리는 A10001에 저장됩니다.

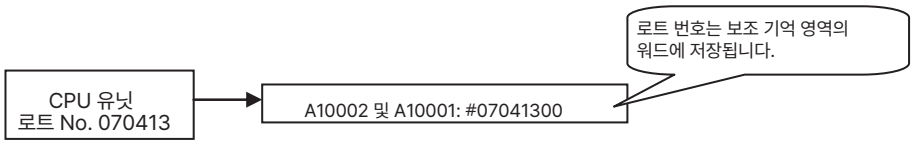


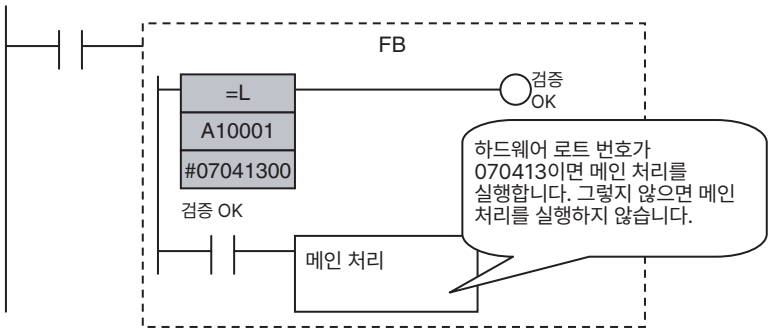
제조 로트 번호 저장 예

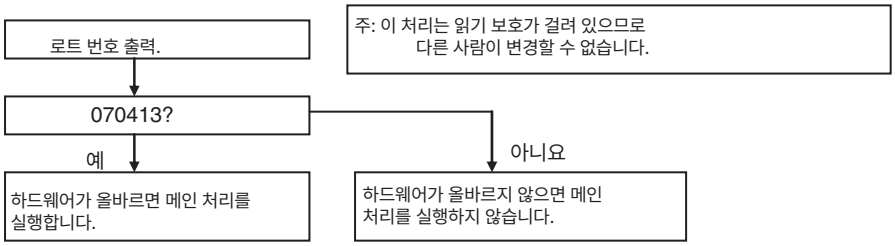
제조 일자	제조 로트 번호	A10002	A10001
2008년 6월 20일	080620	0806	2000

-For Reference Only-

적용 예: 특정 제조 로트 번호를 가진 CPU 유닛에서만 프로그램을 동작시키기







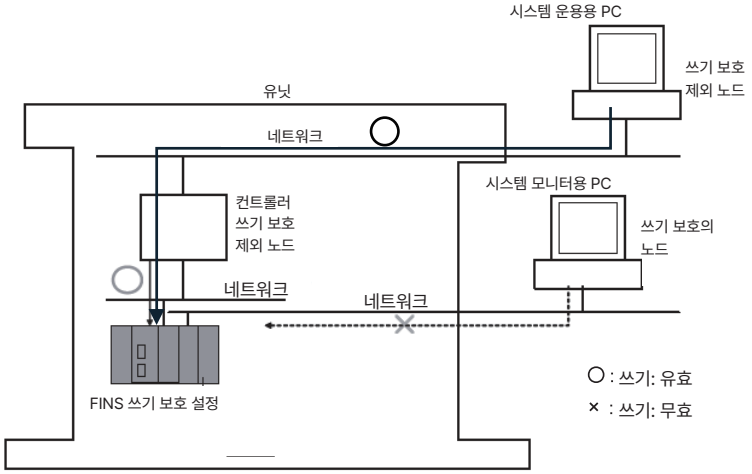
10-6-4 FINS 커맨드에 의한 쓰기 보호

FINS 커맨드에 의한 쓰기 보호

네트워크를 통해 FINS 쓰기 커맨드와 CIP 메시지 통신으로 CPU 유닛에 쓰기나 그 밖의 편집 조작을 수행할 때, 미리 설정한 특정 노드 이외의 노드에서 이루어지는 쓰기나 그 밖의 편집 조작을 금지할 수 있습니다. 그 결과, 네트워크를 통해 데이터 모니터를 실행하면서도 CPU 유닛에 대한 무단 쓰기로 인한 시스템 문제를 방지할 수 있습니다.

AI 번역본

예: 장치 내의 컨트롤러와 시스템 운용용 PC에서의 쓰기만 유효하게 하는 경우



FINS 쓰기 보호 설정 상세

네트워크를 통한 CPU 유닛으로의 쓰기를 보호할 수 있습니다(USB 또는 직접 시리얼 접속은 제외).

CPU 유닛으로의 쓰기에는 다음이 포함됩니다.

・툴에서 사용자 프로그램, PLC 설정, I/O 메모리를 다운로드 · 쓰기, 그리고 동작 모드 변경이나 온라인 에디팅 실행 등의 조작*1

・호스트 PC 또는 주변 기기에서 FINS 커맨드나 CIP 메시지에 의한 쓰기

FINS 커맨드를 사용한 쓰기 조작에서는 특정 노드를 쓰기 보호 대상에서 제외할 수 있습니다. (다만 CIP 메시지 통신을 사용한 쓰기 조작에는 쓰기 보호 제외를 적용할 수 없습니다.)

*1 CX-Programmer, CX-Protocol, CX-Process 등.

주 이 기능은 FINS 커맨드와 CIP 메시지 통신에 의한 쓰기만 금지하므로, 태그 데이터 링크나 데이터 링크 등 FINS 커맨드 · CIP 메시지 통신 이외의 기능에 의한 쓰기 조작에는 영향을 주지 않습니다.

● 보호 대상 FINS 쓰기 커맨드 목록

네트워크를 통해 CPU 유닛으로 보내진 경우, 다음 FINS 커맨드가 FINS 쓰기 보호에 의해 제한됩니다.

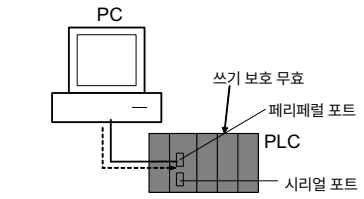
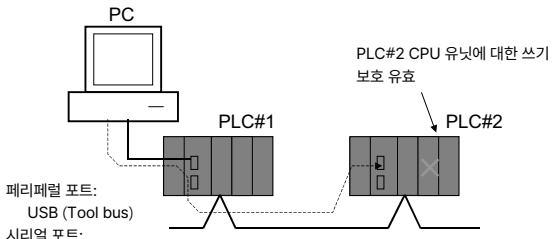
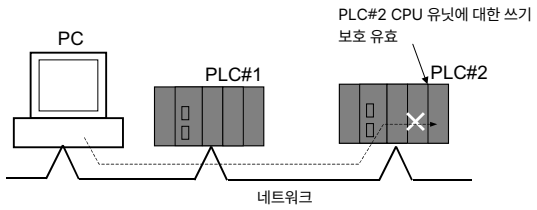
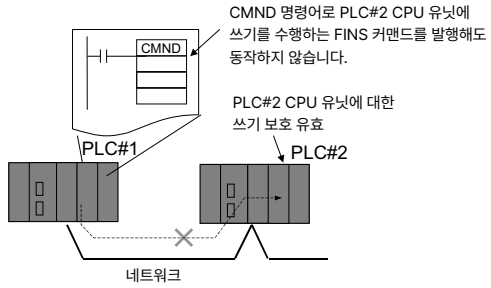
코드	커맨드명
0102 hex	메모리 영역 쓰기
0103 hex	메모리 영역 채우기
0105 hex	메모리 영역 전송
0202 hex	파라미터 영역 쓰기
0203 hex	파라미터 영역 채우기(클리어)
0307 hex	프로그램 영역 쓰기
0308 hex	프로그램 영역 클리어
0401 hex	RUN
0402 hex	STOP
0702 hex	클록 쓰기
0C01 hex	액세스 권한 취득
2101 hex	에러 클리어
2103 hex	에러 로그 포인터 클리어

코드	커맨드명
2203 hex	싱글 파일 쓰기
2204 hex	파일 메모리 포맷
2205 hex	파일 삭제
2207 hex	파일 복사
2208 hex	파일명 변경
220A hex	메모리 영역-파일 전송
220B hex	파라미터 영역-파일 전송
220C hex	프로그램 영역-파일 전송
2215 hex	디렉터리 작성/삭제
2301 hex	강제 세트/리셋
2302 hex	강제 세트/리셋 해제
2141 hex	FINS 쓰기 액세스 로그 클리어
2143 hex	확장 FINS 쓰기 액세스 로그 클리어

● 접속 패턴별 쓰기 보호 유효 여부

네트워크를 통한 보호 대상 접속 패턴은 다음과 같습니다.

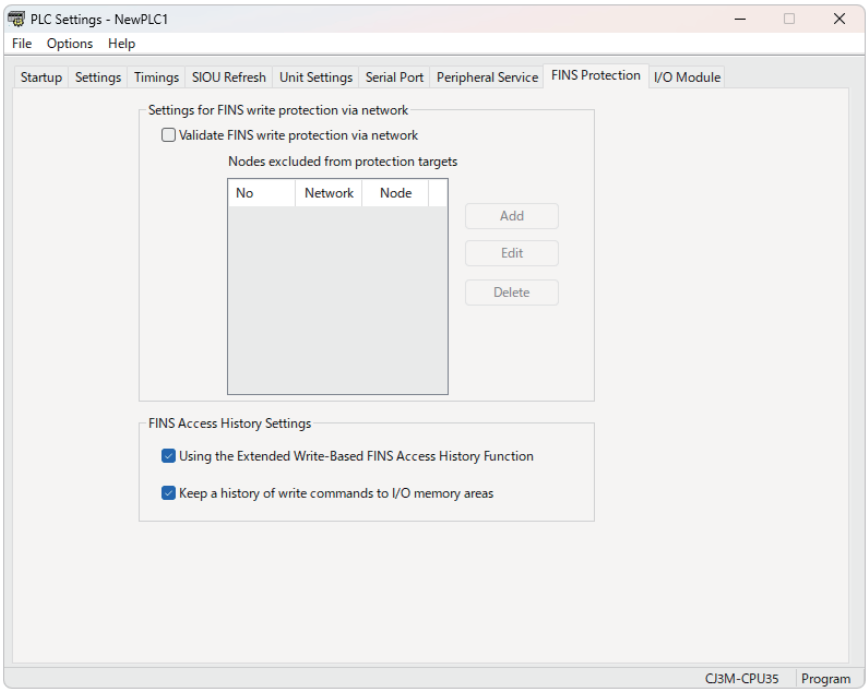
AI 번역본
-For Reference Only-

접속 패턴		그림(예)	쓰기 보호
컴퓨터에서 직접 시리얼 접속으로 접속	PLC에 직접 접속	 페리페럴 포트: USB (Tool bus) 시리얼 포트: Tool bus 시리얼 통신 유닛의 RS-232 또는 422A/485 포트	적용할 수 없습니다.
	PLC에 게이트웨이 접속(시리얼-네트워크)	 페리페럴 포트: USB (Tool bus) 시리얼 포트: Tool bus	적용할 수 있습니다.
컴퓨터에서 직접 네트워크 접속으로 접속하는 경우			적용할 수 있습니다.
네트워크 내의 다른 PLC에서			적용할 수 있습니다.

FINS 쓰기 보호 설정 순서

CX-Programmer에서 PLC 설정의 FINS Protection 탭을 열고 Validate FINS write protection via network 옵션을 선택합니다.

AI 번역본
-For Reference Only-



● 네트워크를 통한 FINS 쓰기 보호 유효화

네트워크를 통해 보내진 FINS 커맨드나 CIP 메시지 통신에 의한 쓰기에 대해 보호를 적용할지 여부를 설정합니다.

항목	설정값	디폴트 설정	설명
네트워크를 통한 FINS 쓰기 보호 유효화	OFF: 무효 ON: 유효	OFF	네트워크를 통해 보내진 FINS 커맨드나 CIP 메시지 통신(즉, 시리얼 접속을 제외한 모든 접속)에 의한 CPU 유닛으로의 쓰기 보호를 유효 또는 무효로 설정합니다.

● 쓰기 보호 제외 노드

이 영역에는 FINS 커맨드에 대한 쓰기 보호로 제한되지 않는 네트워크 내의 노드가 표시됩니다. 노드와 네트워크를 설정하십시오. 최대 32개의 노드를 지정할 수 있습니다.

쓰기 보호 제외 노드가 없으면, 자기 노드 이외의 모든 노드에서 FINS 커맨드에 의한 쓰기 조작이 금지됩니다. 다만 CIP 메시지 통신을 사용한 쓰기 조작에 대해서는 보호 대상 제외 노드를 설정할 수 없습니다.

AI 번역본
-For Reference Only-

항목	설정값	디폴트 설정	설명
쓰기 보호 제외 노드 *1	0~127 hex*1	-	Network: FINS 커맨드 송신원의 네트워크 주소
	1~255 hex*2	-	Node: FINS 커맨드 송신원의 노드 주소

*1 이 설정은 FINS 커맨드에 대한 쓰기 보호가 유효한 경우에만 적용됩니다.

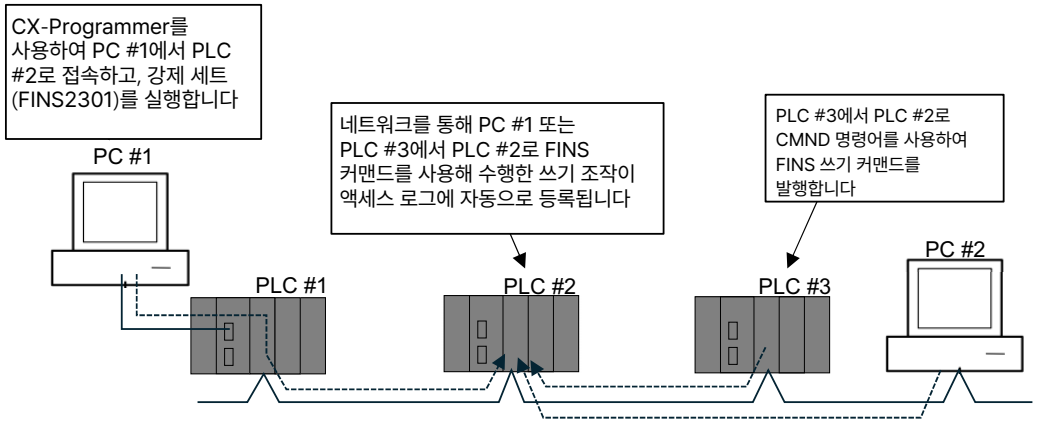
*2 255 hex: 노드 주소 미지정

10-6-5 FINS 쓰기 액세스 로그

FINS 쓰기 액세스 로그 개요

네트워크를 통해 FINS 쓰기 커맨드로 CPU 유닛에 쓰기나 그 밖의 편집 조작을 수행하면, 그 조작의 로그가 자동으로 등록됩니다. 이를 통해 어디서, 언제, 어떤 조작이 이루어졌는지 확인할 수 있으며, 문제가 발생했을 때 부인을 방지하는 데 사용할 수 있습니다.

FINS 쓰기 액세스 로그 기능을 사용하기 위한 설정은 필요하지 않습니다.



FINS 쓰기 액세스 로그 상세

액세스 로그에는 확장 FINS 쓰기 액세스 로그와 FINS 쓰기 액세스 로그의 두 종류가 있습니다.

다음 표에 두 종류의 차이를 정리했습니다.

항목	확장 FINS 쓰기 액세스 로그	FINS 쓰기 액세스 로그
해당 CPU 유닛	CJ3M CPU 유닛 *1	CJ2H/CJ2M/CJ3M CPU 유닛
CX-Programmer에서의 조작	지원 읽기, 클리어(*2), 익스포트	지원하지 않음
FINS 커맨드에 의한 조작 (*3)	지원 읽기, 클리어(*2)	지원 읽기, 클리어
로그에 기록되는 데이터	· 네트워크 주소 · 노드 주소 · 유닛 번호 · FINS 커맨드 코드 · FINS 종료 코드 · 액세스 발생 일시 · 연속 액세스 횟수	· 네트워크 주소 · 노드 주소 · 유닛 번호 · FINS 커맨드 코드 · 액세스 발생 일시
로그 상한	최대 1,024건	최대 20건
저장 규칙	로그는 항상 새 로그로 추가됩니다. 로그 상한을 초과하면 가장 오래된 로그가 파기되고 새 로그가 추가됩니다.	동일한 송신원의 로그가 있으면 그 로그가 새 로그로 덮어쓰기됩니다. 동일한 송신원의 로그가 없으면 새 로그로 추가됩니다. 로그 상한을 초과하면 가장 오래된 로그가 파기되고 새 로그가 추가됩니다.

*1 CJ3M CPU 유닛에서는 확장 FINS 쓰기 액세스 로그가 디폴트입니다.

*2 확장 FINS 쓰기 액세스 로그를 클리어하려면 딥 스위치를 설정해야 합니다.

*3 FINS 커맨드의 상세는 통신 커맨드 레퍼런스 매뉴얼 (Cat. No. W342)의 FINS 쓰기 액세스 로그 읽기/FINS 쓰기 액세스 로그 클리어를 참조하십시오.

AI 번역
-For Reference Only-

● FINS 쓰기 커맨드 목록

네트워크를 통해 CPU 유닛으로 보내진 다음 FINS 커맨드가 FINS 쓰기 액세스 로그에 등록됩니다.

코드	커맨드명
0102 hex	메모리 영역 쓰기
0103 hex	메모리 영역 채우기
0105 hex	메모리 영역 전송
0202 hex	파라미터 영역 쓰기
0203 hex	파라미터 영역 채우기(클리어)
0307 hex	프로그램 영역 쓰기
0308 hex	프로그램 영역 클리어
0401 hex	RUN
0402 hex	STOP
0702 hex	클록 쓰기
0C01 hex	액세스 권한 취득
2101 hex	에러 클리어
2103 hex	에러 로그 포인터 클리어

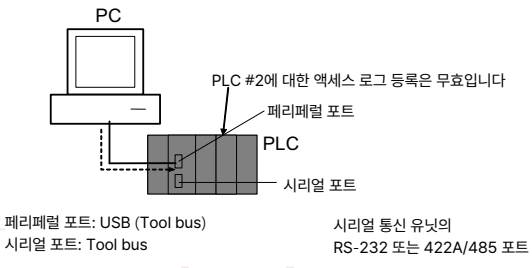
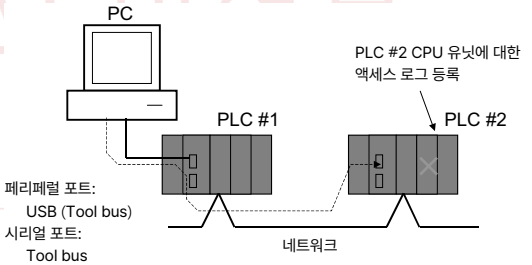
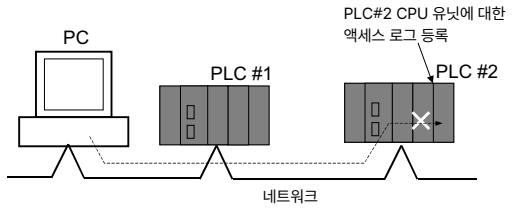
코드	커맨드명
2203 hex	싱글 파일 쓰기
2204 hex	파일 메모리 포맷
2205 hex	파일 삭제
2207 hex	파일 복사
2208 hex	파일명 변경
220A hex	메모리 영역-파일 전송
220B hex	파라미터 영역-파일 전송
220C hex	프로그램 영역-파일 전송
2215 hex	디렉터리 작성/삭제
2301 hex	강제 세트/리셋
2302 hex	강제 세트/리셋 해제
2141 hex	FINS 쓰기 액세스 로그 클리어
2143 hex	확장 FINS 쓰기 액세스 로그 클리어

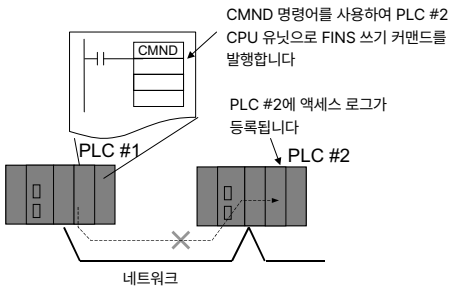
주 1 CX-Programmer 접속 시 등의 내부 처리용 FINS 커맨드가 로그에 등록되는 경우가 있습니다.

2 등록되는 FINS 커맨드는 확장 FINS 쓰기 커맨드 액세스 로그 기능과 FINS 쓰기 커맨드 액세스 로그 기능 모두 동일합니다.

● 접속 패턴별 액세스 로그 등록 유효 여부

FINS 쓰기 액세스 로그가 등록되는 네트워크 접속 패턴은 다음과 같습니다.

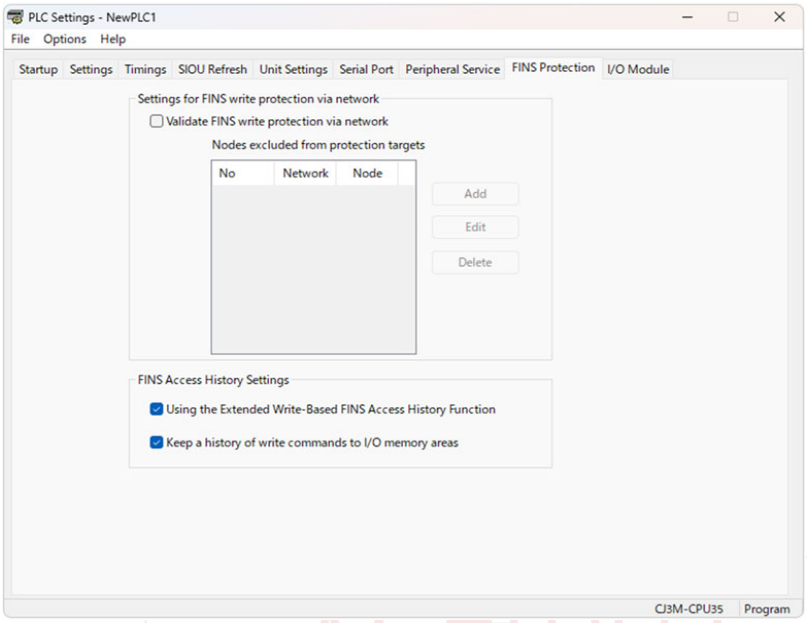
접속 패턴		그림(예)	액세스 로그 등록
컴퓨터에서 직접 시리얼 접속으로 접속	PLC에 직접 접속		지원하지 않음
	PLC에 게이트웨이 접속(시리얼-네트워크)		적용할 수 있습니다.
컴퓨터에서 직접 네트워크 접속으로 접속하는 경우			적용할 수 있습니다.

접속 패턴	그림(예)	액세스 로그 등록
네트워크 내의 다른 PLC에서		적용할 수 있습니다.

FINS 쓰기 액세스 로그 기능 설정 방법

● CJ3M CPU 유닛의 경우

CX-Programmer에서 PLC 설정의 FINS Protection 탭을 열고 Set FINS Access Log 옵션을 선택합니다.



FINS 쓰기 액세스 로그 설정

항목	설정값	디폴트 설정	설명
확장 쓰기 기반 FINS 액세스 이력 기능 사용	· ON: 확장 FINS 쓰기 액세스 로그 사용 · OFF: 기존 FINS 쓰기 액세스 로그 사용	· ON	확장 FINS 쓰기 액세스 로그를 사용할지 기존 FINS 쓰기 액세스 로그를 사용할지 설정합니다.
I/O 메모리 영역에 대한 쓰기 커맨드의 이력 보존 (*)	· ON: 로그 보존 · OFF: 로그 보존하지 않음	· ON	I/O 메모리 영역에 대한 쓰기 로그를 보존할지 여부를 설정합니다.

*대상이 되는 FINS 커맨드 코드는 다음과 같습니다:

- · 메모리 영역 쓰기: 0102 hex
- · 메모리 영역 채우기: 0103 hex
- · 메모리 영역 전송: 0105 hex

● **CJ2H/CJ2M CPU 유닛의 경우**

CJ2H/CJ2M CPU 유닛에는 FINS 쓰기 액세스 로그 기능의 설정이 없습니다.
I/O 메모리 영역에 대한 쓰기 커맨드는 자동으로 로그에 보존됩니다.

FINS 쓰기 액세스 로그 액세스 방법

● **확장 FINS 쓰기 액세스 로그 액세스 방법 (CJ3M CPU 유닛 전용)**

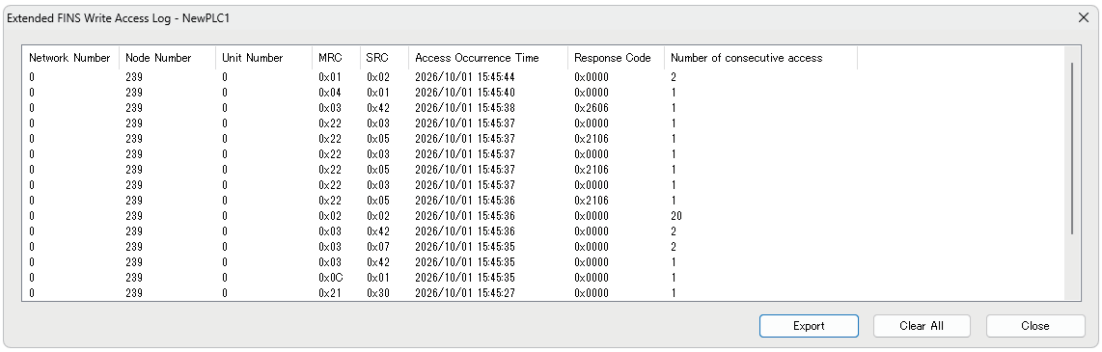
확장 FINS 쓰기 액세스 로그 읽기 방법

확장 FINS 쓰기 액세스 로그는 CX-Programmer로 읽기, 클리어, 익스포트할 수 있습니다.

1 CX-Programmer와 CPU 유닛을 온라인으로 접속합니다.

2 **PLC - PLC Information - Extended FINS Write Access Log를 선택합니다.**

액세스 로그 윈도우가 표시됩니다.



항목	설명
네트워크 주소	액세스 송신원 기기가 접속된 네트워크 주소를 표시합니다.
노드 번호	액세스 송신원 기기의 노드 주소를 표시합니다.
유닛 번호	액세스 송신원 기기의 유닛 어드레스를 표시합니다. (*)
MRC	FINS 커맨드 코드의 상위 1바이트(16진수 2자리)를 표시합니다.
SRC	FINS 커맨드 코드의 하위 1바이트(16진수 2자리)를 표시합니다.
액세스 발생 일시	액세스 일시(년/월/일/시/분)를 표시합니다.
종료 코드	CPU 유닛에서 돌아온 FINS 응답 코드(16진수 4자리)를 표시합니다.
연속 액세스 횟수	동일한 네트워크 번호, 노드 번호, FINS 커맨드 코드로 연속해서 액세스가 이루어지면 액세스 로그의 최신 항목이 덮어쓰기됩니다. (액세스 발생 시각, 응답 코드, 연속 액세스 횟수가 갱신됩니다.) 덮어쓰기가 연속해서 발생하면 연속 덮어쓰기 횟수가 표시됩니다. 새로운 액세스가 액세스 로그에 등록되면 "1"이 등록되고, 이 횟수는 최대 65,535회까지 갱신됩니다. 연속 액세스 횟수가 이 상한을 초과하면 표시되는 횟수는 65,535에서 변하지 않습니다.

-For Reference Only-

* 유닛 번호는 다음과 같이 표시됩니다:

- CPU 유닛: 16진수: #00
- CPU 버스 유닛: 16진수#10+유닛 번호(16진수)
- 이너 보드: 16진수#E1 (CS 시리즈 전용)

시리얼 포트의 유닛 번호

- 시리얼 통신 유닛
  포트 1: 16진수#80 + 16진수#04 × 유닛 번호(16진수)
  포트 2: 16진수#81 + 16진수#04 × 유닛 번호(16진수)
- 시리얼 통신 보드
  포트 1: 16진수: #E4
  포트 2: 16진수: #E5

· CPU 유닛
페리페럴 포트: 16진수: #FD
RS-232C 포트: 16진수: #FC

확장 FINS 쓰기 액세스 로그 클리어 방법

1 딥 스위치를 다음과 같이 설정합니다:

SW1	SW2	SW3	SW4	SW5	SW6	SW7	SW8
- (*)	- (*)	OFF	ON	- (*)	- (*)	- (*)	ON

* "-" 상태는 임의입니다.

2 All Clear 버튼을 클릭합니다.

주 액세스 로그는 전 메모리 클리어 조작으로는 클리어되지 않습니다.

확장 FINS 쓰기 액세스 로그 익스포트

1 Export 버튼을 클릭하고 파일을 저장할 디렉터리를 선택합니다.
액세스 로그가 csv 파일로 출력됩니다.

사용상의 주의

CJ3M-CPU3□의 내장 EtherNet/IP 또는 CJ1W-EIP21S의 EtherNet/IP 포트에서 시큐어 통신 접속을 확립한 경우, 확장 FINS 쓰기 액세스 로그는 관리자 권한을 가진 사용자만 액세스할 수 있습니다. 다른 수단으로 접속한 경우(예: USB 포트 경유 또는 시리얼 포트 경유)에는 액세스가 제한되지 않습니다.

- **FINS 쓰기 액세스 로그 액세스 방법**

이 액세스 로그는 CX-Programmer로 읽을 수 없습니다. FINS 커맨드의 상세는 통신 커맨드 레퍼런스 매뉴얼 (Cat. No. W342)의 FINS 쓰기 액세스 로그 읽기/FINS 쓰기 액세스 로그 클리어를 참조하십시오.

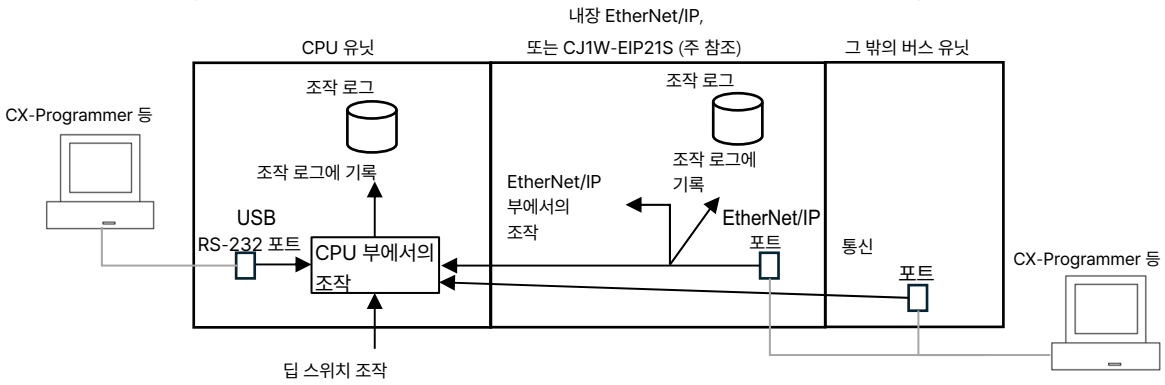
10-6-6 조작 로그 기능

조작 로그 기능은 CJ3M CPU 유닛에서만 지원됩니다.

조작 로그 기능이란?

이 기능은 CPU 유닛에서 수행된 중요한 조작과 그 조작의 결과를 일시 정보와 함께 기록합니다. 이를 통해 언제 어떤 조작이 이루어졌는지 확인할 수 있으며, 문제가 발생했을 때 부인을 방지하는 데 사용할 수 있습니다.

AI 번역본
-For Reference Only-



주 이 항목에서 설명하는 조작 로그는 CPU 유닛의 기능입니다.
CJ3M-CPU3의 내장 EtherNet/IP 부 또는 EtherNet/IP 유닛 CJ1W-EIP21S의 조작 로그 기능에 대해서는 EtherNet/ IP 유닛 오퍼레이션 매뉴얼 (Cat. No. W465)의 13-6 조작 로그 기능을 참조하십시오.

조작 로그 기능 상세

아래에 설명하는 대상 조작이 CPU 유닛에서 실행되면, 그 조작의 결과가 조작 로그로 기록됩니다.

● 조작 로그 대상 조작

- 딥 스위치 변경
- 조작 로그 클리어
- 확장 FINS 쓰기 액세스 로그 클리어
- 특수 I/O 유닛 및 CPU 버스 유닛 재기동

● 조작 로그 최대 건수

CPU 유닛의 불휘발성 메모리에 기록할 수 있는 조작 로그의 최대 건수는 256건입니다. 조작 로그 건수가 상한인 256건을 초과하면 가장 오래된 조작 로그가 파기되고 새로운 조작 로그가 기록됩니다.

사용상의 주의

조작 로그에 사용되는 일시는 CPU 유닛 내의 시계 데이터입니다.
배터리가 소모되었거나 배터리리스 운전 중에는 내부 커패시터가 방전된 후 시계 데이터가 불안정해집니다.
시계 데이터가 필요한 경우에는 배터리를 장착하고, CX-Programmer 또는 EtherNet/IP 유닛이 제공하는 "Automatic Clock Adjustment" 기능으로 CPU 유닛에 올바른 시계 정보를 설정하십시오.

AI 번역본
-For Reference Only-

조작 로그 액세스 방법

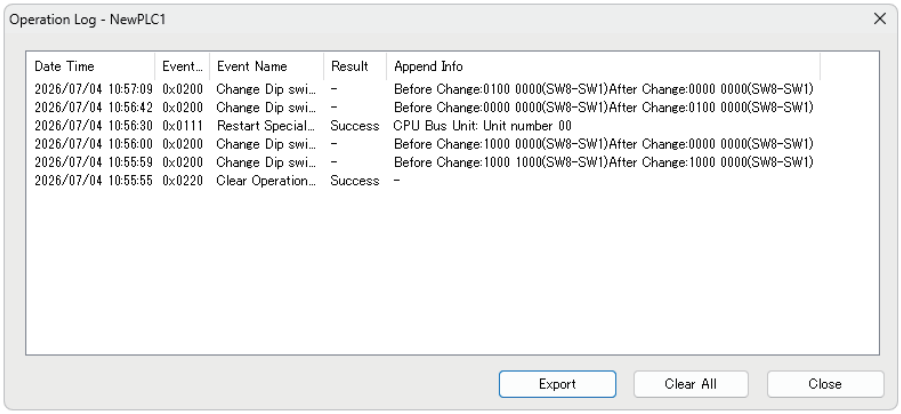
조작 로그는 읽기, 클리어, 익스포트할 수 있습니다.

● 조작 로그 읽기 방법

1 CX-Programmer와 PLC를 온라인으로 접속합니다.

2 **PLC - PLC Information - Operation Logs를 선택합니다.**

조작 로그 윈도우가 표시됩니다.



● 조작 로그 클리어 방법

1 딥 스위치를 다음과 같이 설정합니다:

SW1	SW2	SW3	SW4	SW5	SW6	SW7	SW8
- (*)	- (*)	OFF	ON	- (*)	- (*)	- (*)	ON

* "-" 상태는 임의입니다.

2 조작 로그 익스포트

주 Export 버튼을 클릭하고 파일을 저장할 디렉터리를 선택합니다.

AI 번역본
For Reference Only-

● 조작 로그 익스포트

1 Export 버튼을 클릭하고 파일을 저장할 디렉터리를 선택합니다.
조작 로그가 csv 파일로 출력됩니다.

사용상의 주의

시큐어 통신으로 접속했을 때의 액세스 제한입니다. CJ3M-CPU3□의 내장 EtherNet/IP 포트 또는 CJ1W-EIP21S의 EtherNet/IP 포트에 시큐어 통신으로 접속한 경우, 조작 로그는 관리자 권한을 가진 사용자만 액세스할 수 있습니다. 다른 수단으로 접속한 경우 (예: USB 포트 경유 또는 시리얼 포트 경유)에는 액세스가 제한되지 않습니다.

조작 로그 목록

다음 표에 기록되는 조작 코드, 조작명, 조작 결과, 추가 정보를 정리했습니다.

조작 코드	조작명 조작 상세	조작 결과	추가 정보
0111 hex	특수 I/O 유닛 및 CPU 버스 유닛 재기동 버스 유닛이 재기동되면 재기동 결과가 기록됩니다.	재기동 성공 또는 실패	유닛 종별과 유닛의 유닛 번호 예: "CPU 버스 유닛: 유닛 번호 1" "특수 I/O 유닛: 유닛 번호 10"
0200 hex	딥 스위치 변경 딥 스위치 설정이 변경되면 변경 전과 변경 후의 딥 스위치 상태가 기록됩니다. ・전원 ON 시의 딥 스위치 설정이 직전에 전원을 OFF했을 때의 설정과 다른 경우 ・유닛의 전원이 ON인 상태에서 딥 스위치 설정이 변경된 경우	없음	변경 전과 변경 후의 딥 스위치 상태를 표시합니다. 각 스위치의 상태는 0(OFF) 또는 1(ON)로 표시됩니다. 예: "변경 전: 0000 0000 (SW8-SW1)" 모두 OFF "변경 후: 0010 0000 (SW8-SW1)" 스위치 6이 ON
0220 hex	조작 로그 클리어 조작 로그가 클리어되면 클리어 조작의 결과가 기록됩니다.	클리어 성공 또는 실패	없음
0240 hex	확장 FINS 쓰기 액세스 로그 클리어 FINS 쓰기 액세스 로그가 클리어되면 클리어 조작의 결과가 기록됩니다.	클리어 성공 또는 실패	없음

주 위 정보 외에 메인터넌스용 시스템 동작 정보가 기록되는 경우가 있습니다.

참고

[0220 조작 로그 클리어]와 [0200 딥 스위치 변경]은 공장 출하 시에 기록됩니다.

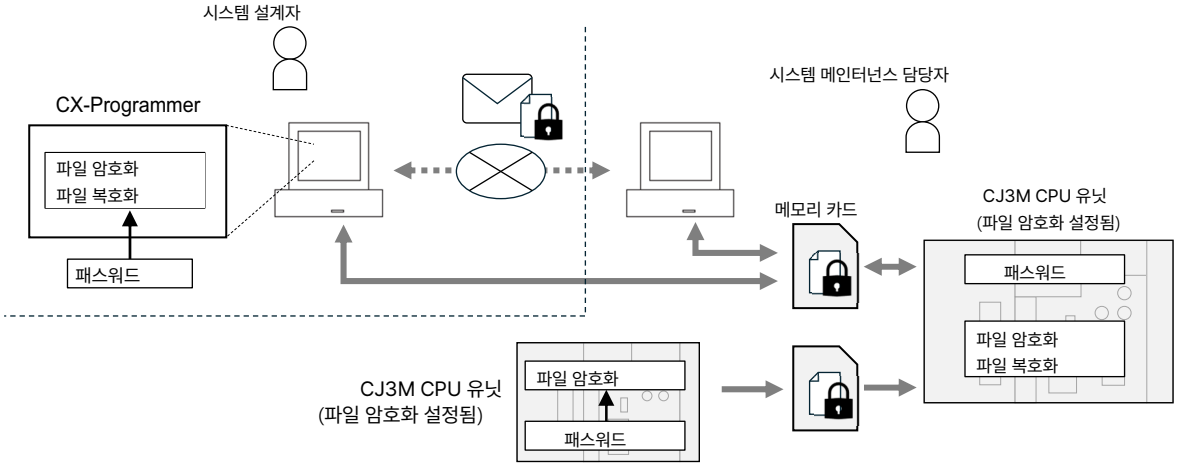
10-6-7 파일 암호화 기능

파일 암호화 기능은 CJ3M-CPU1□/3□에서만 지원됩니다.

AI 번역본
For Reference Only

파일 암호화 기능이란?

이 기능은 CX-Programmer에서 작성한 프로그램이나 CPU 유닛에서 백업한 프로그램 등 CPU 유닛 내부 데이터와 관련된 파일을 암호화하고, 암호화된 파일을 읽을 때 변조를 검출합니다. 그 결과, 제3자의 부정 액세스로부터 파일을 보호할 수 있습니다.
파일을 암호화하면 파일을 안전하게 주고받을 수 있습니다.



적용 예

- **시스템 내의 CPU 유닛에서 파일을 출력하여 CX-Programmer로 설계 공정에서 읽기**

1 시스템 메인터넌스 담당자가 파일 암호화가 유효한 CPU 유닛에 메모리 카드를 장착하고, CPU 유닛에 저장된 파일을 "간이 백업 기능"으로 암호화된 파일로서 메모리 카드에 전송합니다. <246></246>

2 시스템 메인터넌스 담당자는 메모리 카드의 암호화된 파일을 전자 메일 또는 메모리 카드 실물로 시스템 설계자에게 보냅니다. 시스템 설계자는 CX-Programmer가 설치된 PC에 암호화된 파일을 저장합니다.

3 시스템 설계자는 CX-Programmer의 [Read from File] 기능을 사용하여 암호화된 파일을 읽습니다.

- **설계 공정에서 CX-Programmer로 파일을 작성하여 시스템 내의 CPU 유닛으로 다운로드하기**

1 시스템 설계자는 CX-Programmer의 Write from File 기능을 사용하여 패스워드를 설정하고 암호화된 파일을 작성합니다.

주 이때 파일 암호화가 유효한 CPU 유닛의 패스워드와 동일한 패스워드를 설정합니다.

AI 번역본
-For Reference Only-

2 시스템 설계자 또는 시스템 메인터넌스 담당자가 암호화된 파일을 메모리 카드에 저장합니다.

3 시스템 메인터넌스 담당자가 파일 암호화가 유효한 CPU 유닛에 메모리 카드를 장착하고, "기동 시 자동 전송" 또는 "운전 중 전체 프로그램 교체"로 암호화된 파일을 메모리 카드에서 CPU 유닛으로 전송합니다.

주 사전에 시스템 내의 CPU 유닛에서 파일 암호화를 유효로 설정해야 합니다.

● CPU 유닛에서 메모리 카드를 경유해 다른 CPU 유닛으로 프로그램 복사하기 (CPU 유닛 교체)

1 시스템 메인터넌스 담당자가 전송원인 파일 암호화가

유효한 CPU 유닛에서 "간이 백업 기능"으로 프로그램을 백업하고, CPU 유닛에서 메모리 카드로 암호화된 파일을 전송합니다.

2 시스템 메인터넌스 담당자가 백업 파일이 들어 있는 메모리 카드를 전송처인 파일 암호화가 유효한 CPU 유닛에 장착하고, "간이 백업 조작에 의한 메모리 카드에서 PLC로의 복원"으로 암호화된 파일을 CPU 유닛으로 전송합니다.

주 사전에 시스템 내의 CPU 유닛에서 파일 암호화를 유효로 설정해야 합니다.

파일 암호화 기능 상세

● 파일 암호화 기능의 대상 파일

대상 파일의 종류와 암호화된 파일의 확장자는 다음과 같습니다.

파일 종류	내용	암호화되지 않은 경우의 확장자	암호화된 경우의 확장자
프로그램/네트워크 심볼 파일	사용자 프로그램, 펑션 블록, 네트워크 심볼 등	.OBJ	.OBS
파라미터 영역 파일	PLC 이름, PLC 설정, 등록 I/O 테이블, 라우팅 테이블, 데이터 링크 테이블 등	.STD	.STS
데이터 파일	I/O 메모리(데이터 메모리, 확장 데이터 메모리)	.IOM	.IOS
	I/O 메모리(CIO, 워크 영역, 유지 영역, 타이머, 카운터, 보조 기억 영역)	.IOR	.IRS
심볼 테이블 파일	심볼 테이블(심볼명, I/O 코멘트)	.SYM	.SBS
코멘트 파일	코멘트(행 코멘트, I/O 코멘트 이외의 주석)	.CMT	.CMS
프로그램 인덱스 파일	CX-Programmer의 섹션명, 섹션 코멘트, 프로그램 코멘트	.IDX	.IDS
유닛 백업 파일	각 버스 유닛 전용 주변 툴의 파일	.PRM	.---(*)

* 유닛 백업 파일은 암호화되지 않습니다.

● 파일 조작 기능의 대상 파일

암호화가 유효한 경우에는 암호화된 파일만 사용할 수 있습니다. 암호화되지 않은 파일은 읽기나 쓰기를 할 수 없습니다. 또한 변조된 암호화 파일은 읽을 수 없습니다.

파일 종류	간이 백업 기능 *1	기동 시 자동 전송 *2	운전 중 전체 프로그램 교체 *3	파일 메모리 명령어 *4	CX-Programmer 파일 메모리 전송 기능 (온라인)	CX-Programmer에 의한 파일 읽기/쓰기 (오프라인)	FINS 커맨드에 의한 전송
	CPU 유닛 → 메모리 카드 메모리 카드 → CPU 유닛	메모리 카드 → CPU 유닛	메모리 카드 → CPU 유닛	메모리 카드 → CPU 유닛 CPU 유닛 → 메모리 카드	메모리 카드 → CPU 유닛 CPU 유닛 → 메모리 카드	CX-Programmer → 파일 파일 → CX-Programmer	CPU 유닛 → PC PC → CPU 유닛
프로그램/네트워크 심볼 파일	가능	가능	가능	---	가능	가능	가능
파라미터 영역 파일	가능	가능	---	---	가능	---	가능
데이터 파일	가능	가능	---	가능	가능	---	가능
심볼 테이블 파일	가능	---	---	---	---	가능	---

파일 종류	간이 백업 기능 *1	기동 시 자동 전송 *2	운전 중 전체 프로그램 교체 *3	파일 메모리 명령어 *4	CX-Programmer 파일 메모리 전송 기능 (온라인)	CX-Programmer에 의한 파일 읽기/쓰기 (오프라인)	FINS 커맨드에 의한 전송
	CPU 유닛 → 메모리 카드 메모리 카드 → CPU 유닛	메모리 카드 → CPU 유닛	메모리 카드 → CPU 유닛	메모리 카드 → CPU 유닛 CPU 유닛 → 메모리 카드	메모리 카드 → CPU 유닛 CPU 유닛 → 메모리 카드	CX-Programmer → 파일 파일 → CX-Programmer	CPU 유닛 → PC PC → CPU 유닛
코멘트 파일	가능	---	---	---	---	가능	---
프로그램 인덱스 파일	가능	---	---	---	---	가능	---
유닛 백업 파일 *5	가능	---	---	---	---	---	---

가능: 조작할 수 있는 대상 파일, ---: 조작할 수 없는 파일

*1 파일 암호화가 유효한 경우, 암호화된 파일이 메모리 카드에 저장됩니다. 간이 백업 기능의 상세는 CJ3 CPU 유닛 하드웨어 사용자 매뉴얼 (Cat. No. W472)의 간이 백업을 참조하십시오.

*2 파일 암호화가 유효한 경우, 암호화된 파일을 메모리 카드에 저장하십시오. 기동 시 자동 전송의 상세는 10-3-5 기동 시 자동 전송을 참조하십시오.

*3 파일 암호화가 유효한 경우, 암호화된 파일을 메모리 카드에 저장하십시오. 기동 시 자동 전송의 상세는 10-5-4 운전 중 전체 프로그램 교체를 참조하십시오.

*4 파일 암호화가 유효한 경우, 각 명령어 사용에 다음과 같은 제한이 적용됩니다. 상세는 CS/CJ/NSJ 시리즈 명령어 레퍼런스 매뉴얼 (W474)의 FREAD(700)/FWRIT(701)/TWRIT(704)를 참조하십시오.
제한 사항

- FREAD: 명령어에서 사용할 수 있는 데이터 형식은 BIN(확장자: .IOS)뿐입니다.
- FWRIT: 명령어에서 사용할 수 있는 데이터 형식은 BIN(확장자: .IOS)뿐입니다.
  암호화가 유효한 경우에는 새 파일 작성만 가능하며, 덮어쓰기나 추가 쓰기는 할 수 없습니다.
- TWRIT: 암호화가 유효한 경우에는 이 명령어를 사용할 수 없습니다.

*5 유닛 백업 파일은 암호화 대상이 아닙니다.

파일 암호화 기능의 유효/무효 설정 및 상태 확인 방법

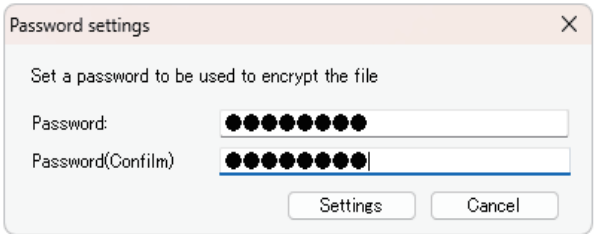
● 파일 암호화 유효화

CPU 유닛에 온라인으로 접속한 상태에서 CX-Programmer에서 조작하여 파일 암호화를 설정합니다.

1 **CX-Programmer에서 PLC - Transfer - File Encryption을 선택합니다.**

2 패스워드를 설정합니다.

패스워드는 최대 16자까지 설정할 수 있습니다.



AI 번역본
-For Reference Only-

● 파일 암호화 무효화

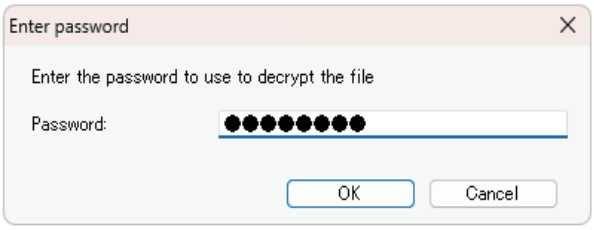
CPU 유닛에 온라인으로 접속한 상태에서 CX-Programmer에서 조작하여 파일 암호화를 설정합니다.

1 **CX-Programmer에서 PLC - Transfer - File Encryption을 선택합니다.**

암호화 상태가 표시됩니다.

2 암호화를 유효로 했을 때 설정한 패스워드를 입력합니다.

패스워드를 연속 5회 잘못 입력하면 2시간 동안 패스워드를 다시 입력할 수 없게 됩니다.



참고

CPU 유닛에 설정한 패스워드를 잊어버린 경우, 파일 암호화 설정을 무효로 하려면 전 메모리 클리어 조작이 필요합니다. 전 메모리 클리어 조작을 실행하면 파일 암호화 설정이 무효가 됩니다. 또한 파일에 설정한 패스워드를 잊어버리면 CX-Programmer 등의 소프트웨어로 CPU 유닛의 데이터를 읽을 수 없게 됩니다. 패스워드는 적절하게 관리하십시오.

AI 번역본
-For Reference Only-

● 파일 암호화의 유효/무효 상태 확인 방법

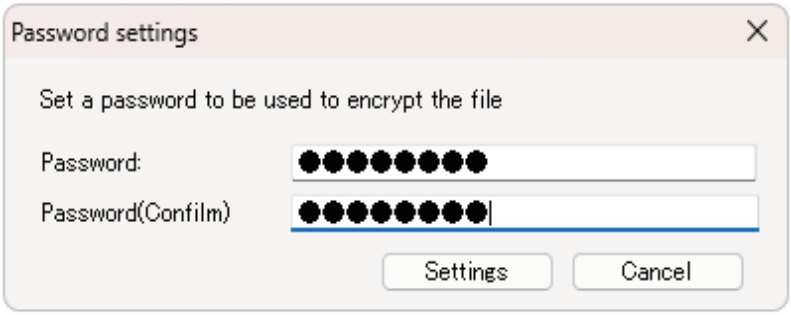
관련 보조 기억 영역

이름	어드레스	설명
파일 암호화 유효 플래그	A343.04	파일 암호화 유효 설정의 상태가 저장됩니다. 0: 암호화 무효 1: 암호화 유효

CX-Programmer를 사용한 파일 읽기/쓰기 방법 (오프라인 조작)

● 암호화된 파일 쓰기

CX-Programmer에서 PLC - Transfer - Write to File을 선택하고, write to file 화면에서 Encrypt File 체크 박스를 선택한 후 파일을 저장할 디렉터리를 지정합니다.

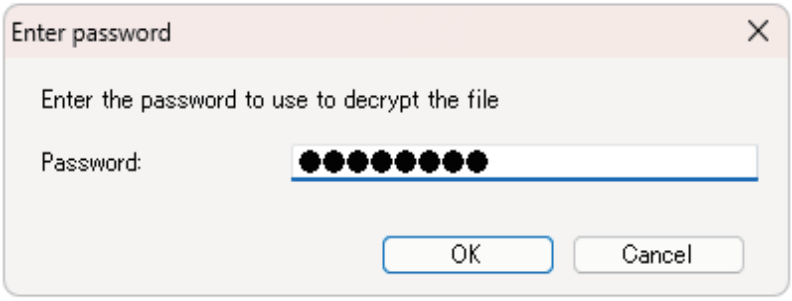


● 암호화된 파일 읽기

CX-Programmer에서 PLC - Transfer - Read from File을 선택하고 파일을 선택합니다.

패스워드를 입력하고 데이터를 읽습니다.

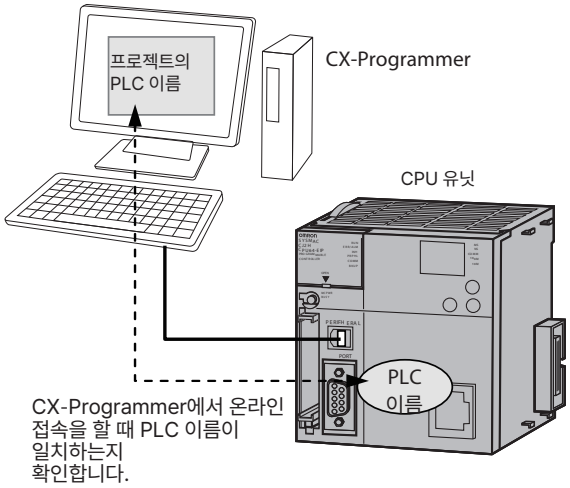
(암호화된 파일을 읽으면 패스워드 입력 화면이 표시됩니다.)



10-6-8 PLC 이름

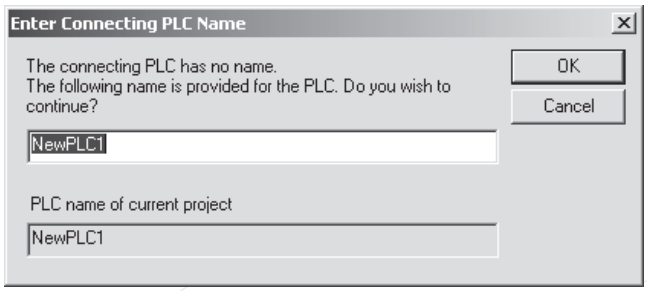
PLC 이름

CX-Programmer에서 온라인 접속을 할 때, 프로젝트의 PLC 이름과 접속 대상 CPU 유닛의 PLC 이름이 일치하는지 확인할 수 있습니다. 이를 통해 CX-Programmer에서의 잘못된 접속을 방지할 수 있습니다.



PLC 이름 등록

디폴트 설정을 사용 중이거나 전 메모리 클리어를 실행한 CPU 유닛에 온라인 접속을 하면 Enter Connecting PLC Name 대화 상자가 표시됩니다.



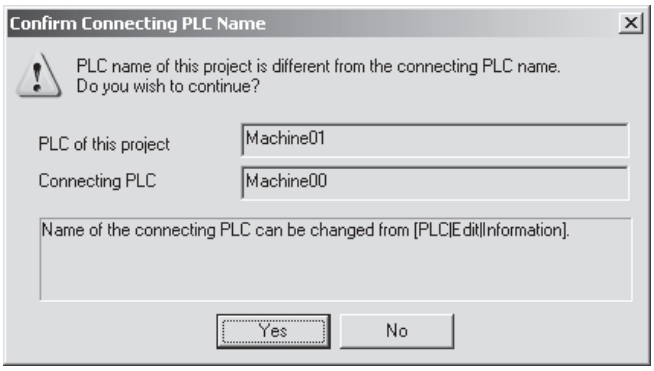
온라인으로 접속된 CPU 유닛에 등록할 PLC 이름을 입력하고 OK 버튼을 클릭합니다. PLC 이름을 입력하지 않을 경우에는 Cancel 버튼을 클릭합니다.

AI 번역본
-For Reference Only-

PLC 이름 불일치

CPU 유닛에 PLC 이름이 저장되어 있으면, 온라인 접속 시 접속 대상 CPU 유닛의 PLC 이름과 프로젝트의 PLC 이름이 일치하는지 시스템이 자동으로 확인합니다.

접속 대상 CJ2/CJ3 CPU 유닛에 저장된 PLC 이름이 프로젝트의 PLC 이름과 다르면 다음 경고가 표시됩니다. 온라인 접속을 계속할지 여부를 선택하십시오.



· Yes 버튼
이름이 다른 상태로 온라인 접속을 계속하려면 Yes 버튼을 클릭합니다.

· No 버튼
오프라인으로 전환하려면 No 버튼을 클릭합니다.

참고

PLC 이름의 일치 여부를 자동으로 확인하지 않으려면 PLC Properties 대화 상자에서 Check the connecting PLC name when going online 체크 박스의 선택을 해제하십시오.

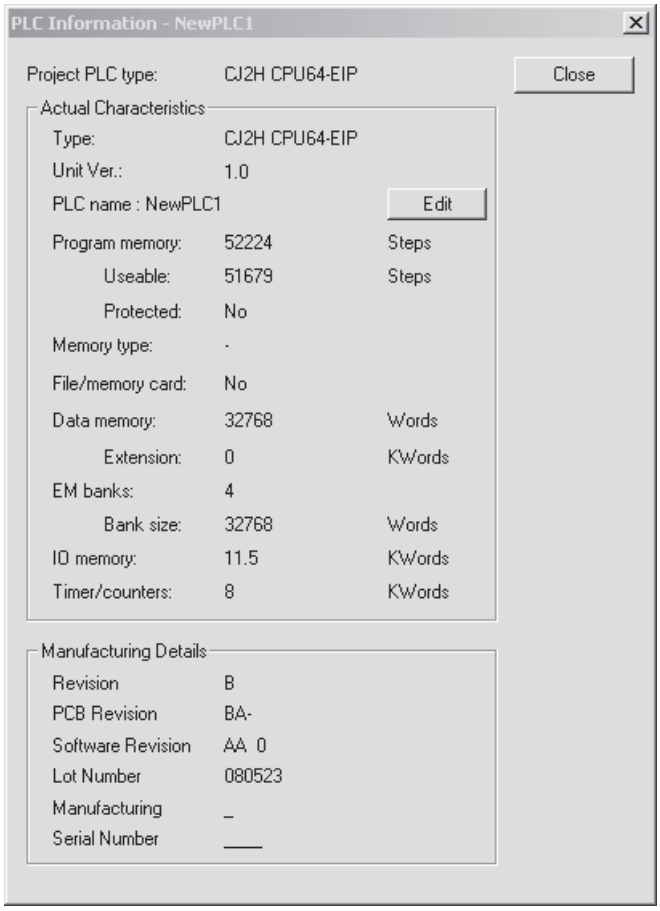
AI 번역본
-For Reference Only-

PLC 이름 변경

CPU 유닛에 저장된 PLC 이름을 변경하려면 다음 순서를 따르십시오.

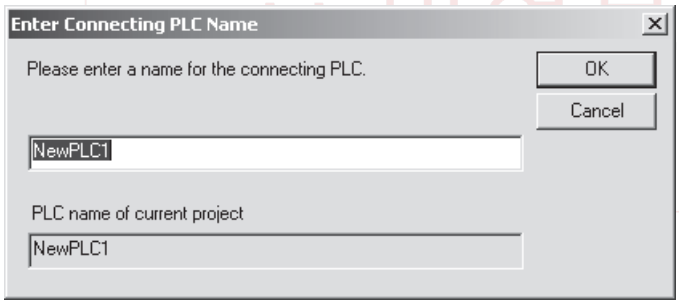
1 CX-Programmer로 온라인 접속하고 PLC 메뉴에서 Edit – Information을 선택합니다.

다음 대화 상자가 표시됩니다.



2 PLC Name 영역 오른쪽의 Edit 버튼을 클릭합니다.

다음 Enter Connecting PLC Name 대화 상자가 표시됩니다.



3 접속 대상 PLC에 등록할 PLC 이름을 입력하고 OK 버튼을 클릭합니다.

10-7 디버깅

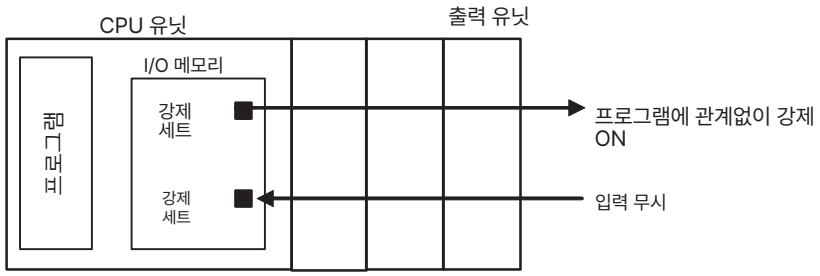
10-7-1 강제 세트/리셋

CX-Programmer는 지정한 비트(CIO 영역, 보조 기억 영역, HR 영역, 타이머/카운터 완료 플래그)를 강제 세트(ON)하거나 강제 리셋(OFF)할 수 있습니다. 강제 상태는 프로그램이나 I/O 리프레시에서 출력되는 상태보다 우선합니다. 이 상태는 명령어로 덮어쓸 수 없으며, CX-Programmer에서 클리어할 때까지 프로그램이나 외부 입력의 상태에 관계없이 유지됩니다.

강제 세트/리셋 조작은 시운전 중에 입력과 출력을 강제하거나 디버깅 중에 특정 조건을 강제하는 데 사용합니다.

강제 세트/리셋 조작은 MONITOR 또는 PROGRAM 모드에서는 실행할 수 있지만 RUN 모드에서는 실행할 수 없습니다.

주 강제 상태 유지 비트(A500.13)와 IOM 유지 비트(A500.12)를 동시에 ON하면 동작 모드를 전환할 때 강제 세트 또는 리셋된 비트의 상태를 유지할 수 있습니다.
기동 시 강제 상태 유지 비트의 상태를 유지하려면 강제 상태 유지 비트(A500.13)와 IOM 유지 비트(A500.12)를 ON하고, PLC 설정에서 Forced Status Hold Bit at Startup 설정을 지정하십시오. 이렇게 하면 전원을 ON했을 때 강제 상태를 유지할 수 있습니다.



다음 영역의 비트를 강제 세트 및 리셋할 수 있습니다.

CIO 영역(I/O 영역 비트, 데이터 링크 영역 비트, CPU 버스 유닛 영역 비트, 특수 I/O 유닛 영역 비트, 내부 I/O 영역 비트), 워크 영역, 타이머 완료 플래그, 유지 영역, 카운터 완료 플래그, 그리고 EM 영역 강제 세트/리셋 기능 사용이 지정된 EM 영역의
뱅크*1, *2

*1 EM 강제 세트/리셋 기능은 CJ2H CPU 유닛 버전 1.1 이전에서는 지원되지 않습니다.

*2 CJ2H CPU 유닛의 경우, 자동 어드레스 할당이 지정된 다음 뱅크에서도 EM 영역의 비트를 강제 세트/리셋할 수 있습니다.
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

AI 번역본
-For Reference Only-

● CX-Programmer 조작

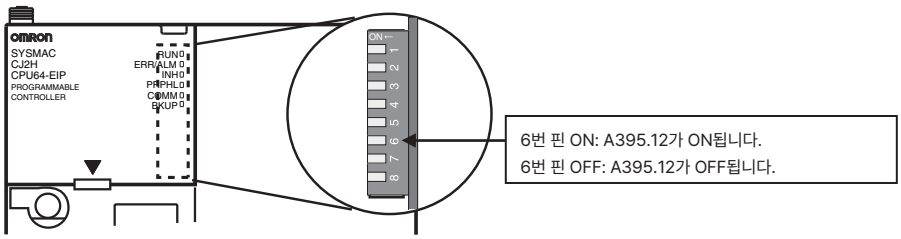
- 강제 세트/리셋할 비트를 선택합니다.
- 강제 세트 또는 강제 리셋을 선택합니다.
- 강제 상태를 클리어합니다(모든 강제 상태를 동시에 클리어하는 것도 포함).

● 강제 상태 유지 비트

이름	어드레스	기능	액세스
강제 상태 유지 비트	A500.13	PROGRAM 모드와 RUN 또는 MONITOR 모드 사이를 전환할 때, 또는 전원을 ON할 때 강제 세트 또는 강제 리셋된 비트의 상태를 유지하려면 이 비트를 ON합니다.	읽기/쓰기

10-7-2 테스트 입력

CPU 유닛 전면의 딥 스위치 6번 핀의 ON/OFF 상태는 보조 기억 영역의 딥 스위치 핀 상태 플래그(A395.12)에 저장됩니다. 디버깅 등의 목적으로 입력 유닛을 사용하지 않고 테스트로 입력을 수동으로 조작할 수 있습니다.

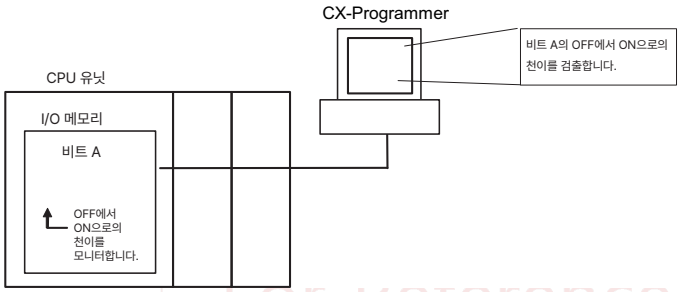


● 보조 기억 영역 플래그 및 워드

이름	어드레스	기능	액세스
딥 스위치 핀 상태 플래그	A395.12	CPU 유닛의 딥 스위치 6번 핀에 설정된 상태를 저장합니다. (매 사이클 리프레시됩니다.)	읽기 전용

10-7-3 미분 모니터

CX-Programmer로 설정한 비트가 OFF에서 ON으로, 또는 ON에서 OFF로 바뀐 것을 CPU 유닛이 검출하면 그 결과가 미분 모니터 완료 플래그(A508.09)에 표시됩니다. 이 플래그는 미분 모니터에 설정한 조건이 충족되면 ON됩니다. CX-Programmer는 이 결과를 모니터하여 화면에 표시할 수 있습니다.

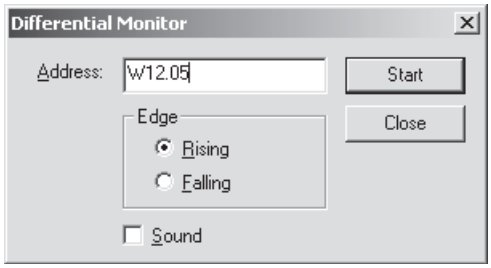


-For Reference Only-

- **CX-Programmer에서의 조작**

1 미분 모니터할 비트를 오른쪽 클릭합니다.

2 PLC 메뉴에서 Differential Monitor를 클릭합니다. Differential Monitor 대화 상자가 표시됩니다.



3 **Rising 또는 Falling을 클릭합니다.**

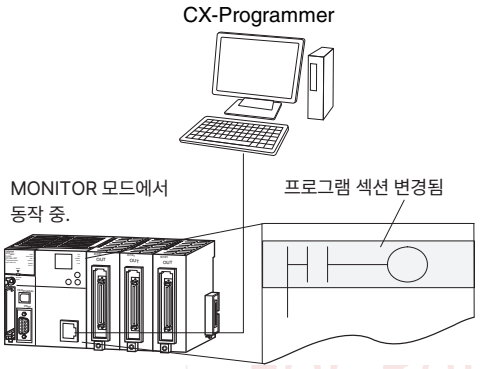
4 Start 버튼을 클릭합니다. 지정한 변화가 검출되면 부저가 울리고 카운트가 증가합니다.

5 Stop 버튼을 클릭합니다. 미분 모니터가 정지합니다.

10-7-4 온라인 에디팅

온라인 에디팅 기능은 CPU 유닛이 MONITOR 또는 PROGRAM 모드일 때 CX-Programmer에서 직접 CPU 유닛 내 프로그램의 일부를 추가하거나 변경하는 데 사용합니다. CX-Programmer에서 한 번에 하나 이상의 프로그램 섹션을 추가하거나 변경합니다. 따라서 이 기능은 CPU 유닛을 정지시키지 않고 프로그램을 소폭 변경하기 위한 것입니다.

서로 다른 태스크를 편집하는 한, CX-Programmer를 실행하는 여러 대의 컴퓨터에서 동시에 온라인 에디팅을 할 수 있습니다.



MONITOR 모드에서 CPU 유닛 내 프로그램을 온라인으로 편집하면 사이클 타임이 1사이클에서 수 사이클분 늘어납니다.

또한 온라인 에디팅 후 플래시 메모리에 데이터를 백업하기 위해 사이클 타임이 늘어납니다. 이 기간 동안 BKUP 표시등이 점등합니다. 백업 진행 상황은 CX-Programmer에 표시됩니다. 사이클당 사이클 타임 지연은 약 1 ms입니다.

백업 완료를 기다리지 않고 연속해서 편집할 수 있는 횟수에는 제한이 있습니다. 실제 횟수는 수행하는 편집의 종류에 따라 다르지만, 400회를 기준으로 삼을 수 있습니다.

제한을 초과하면 CX-Programmer에 메시지가 표시되며, CPU 유닛이 데이터 백업을 완료할 때까지 더 이상 편집할 수 없습니다. 백업이 완료되면 온라인 에디팅을 다시 계속할 수 있습니다.

-역본
-Experience Only-

사용상의 주의

MONITOR 모드에서 온라인 에디팅으로 프로그램을 덮어쓰면 사이클 타임이 평소보다 길어지므로, 늘어나는 시간이 PLC 설정에서 설정한 사이클 감시 시간을 초과하지 않도록 확인하십시오. 감시 시간을 초과하면 사이클 타임 오버 에러가 발생하고 CPU 유닛이 정지합니다. 먼저 PROGRAM 모드를 선택한 후 RUN 또는 MONITOR 모드로 변경하여 CPU 유닛을 재기동하십시오.

온라인 에디팅 후에는 백업 메모리로의 백업이 완료되어 BKUP 표시등이 OFF될 때까지 CPU 유닛의 전원을 OFF하지 마십시오.

주 1 온라인으로 편집하는 프로그램에 포함된 상승 미분 명령어(DIFU 또는 @가 붙은 명령어)와 하강 미분 명령어(DIFD 또는 %가 붙은 명령어)의 내부 상태(미분 플래그)는 초기화됩니다.

2 온라인으로 편집하는 태스크에 블록 프로그램이 포함되어 있으면, WAIT(805), TIMW(813), TIMWX(816), CNTW(814), CNTWX(817), TMHW(815), TMHWX(818)로 만들어진 WAIT 상태가 BPPS(811)에 의해 클리어되고, 다음 실행은 처음부터 이루어집니다.

● CX-Programmer에서의 온라인 에디팅

1 편집할 프로그램 섹션을 표시합니다.

2 편집할 명령어를 선택합니다.

3 **Program, Online Edit, Begin을 차례로 선택합니다.**

4 명령어를 편집합니다.

5 Program, Online Edit, Send Changes를 차례로 선택합니다. 명령어가 체크되고, 에러가 없으면 CPU 유닛으로 전송됩니다. 이때 CPU 유닛 내의 명령어가 덮어쓰기되고 사이클 타임이 늘어납니다.

⚠ 주의

늘어난 사이클 타임이 동작에 영향을 주지 않는지 확인한 후에 온라인 에디팅을 진행하십시오. 사이클 타임이 너무 길면 입력 신호가 입력되지 않을 수 있습니다.

AI 번역본
For Reference Only-

참고

온라인 에디팅의 일시적 금지

해당 사이클의 기계 제어 응답 특성을 확보하기 위해 특정 시간 동안 온라인 에디팅을 금지할 수 있습니다. 온라인 에디팅을 금지하면 누군가가 온라인 에디팅을 수행하여 사이클 타임이 의도치 않게 늘어나는 것을 방지할 수 있습니다.

- 온라인 에디팅 금지 설정
  다음 두 가지 설정 중 하나로 온라인 에디팅을 금지할 수 있습니다.
  온라인 에디팅 금지 비트 유효화(A527.00~A527.07)를 5A hex로 설정합니다.
  온라인 에디팅 금지 비트(A527.09)를 ON합니다.
- 온라인 에디팅이 금지된 상태에서의 동작
  CX-Programmer에서 온라인 에디팅 요구가 있으면 시스템은 대기 상태가 되고 온라인 에디팅을 수행하지 않습니다. 온라인 에디팅 대기 플래그(A201.10)가 ON됩니다.
  그 후 사용자가 온라인 에디팅 금지 비트(A527.09)를 OFF하면 온라인 에디팅이 수행됩니다. 이미 온라인 에디팅이 대기 중이면 온라인 에디팅 조작은 무시됩니다.

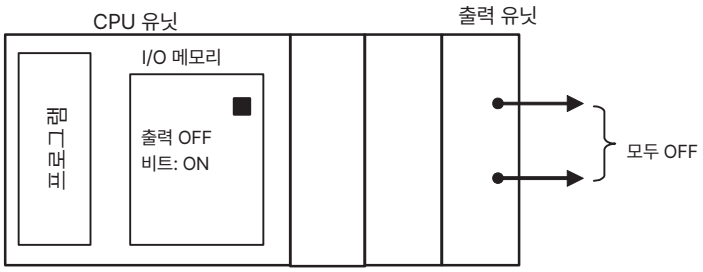
● 관련 보조 기억 비트/워드

이름	어드레스	설명
온라인 에디팅 금지 비트 유효화	A527.00~ A527.07	온라인 에디팅 금지 비트(A527.09)를 유효하게 합니다. 5A hex: 온라인 에디팅 금지 비트 유효 5A hex 이외: 온라인 에디팅 금지 비트 무효
온라인 에디팅 금지 비트	A527.09	온라인 에디팅을 금지하려면 이 비트를 ON하고 온라인 에디팅 금지 비트 유효화(A527.00~A527.07)를 5A로 설정합니다.
온라인 에디팅 대기 플래그	A201.10	온라인 에디팅이 금지되어 온라인 에디팅 처리가 대기 중일 때 ON됩니다.

10-7-5 출력 OFF

출력 OFF 비트(A500.15)를 ON하면 I/O 메모리의 출력 비트 상태에 관계없이 모든 출력 유닛의 출력이 OFF됩니다. 이는 유닛이 RUN 또는 MONITOR 모드로 동작 중일 때 긴급 에러 처리에 사용할 수 있습니다. CPU 유닛 전면의 INH 표시등이 황색으로 점등합니다.

배터리가 있으면 전원을 OFF했다가 ON해도 출력 OFF 비트의 상태가 유지됩니다.



참고

디폴트 상태(즉, IOM 유지 비트(A500.12)가 OFF인 경우)에서는 동작 모드가 RUN 또는 MONITOR 모드에서 PROGRAM 모드로 전환되면 모든 출력이 클리어(즉, OFF)됩니다. 사용자 프로그램에 의해 I/O 메모리의 출력 비트가 ON되면 I/O가 리프레시될 때 출력이 ON됩니다. 동작 모드에 관계없이 출력 유닛의 모든 출력을 OFF하려면 출력 OFF 비트를 사용하십시오.

AI 번역본
-For Reference Only-

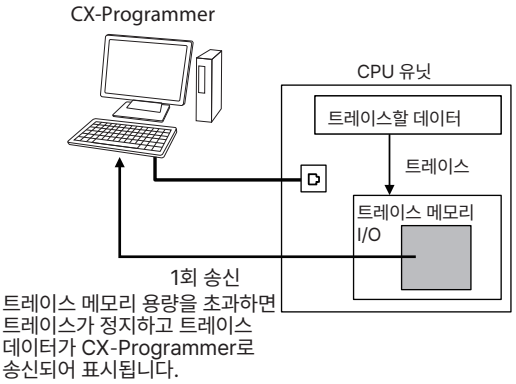
10-7-6 데이터 트레이스

데이터 트레이스 기능은 다음 타이밍 방식 중 하나를 사용하여 지정한 조건에서 지정한 I/O 메모리 데이터를 샘플링하고, 샘플링한 데이터를 트레이스 메모리에 저장합니다. 저장된 데이터는 나중에 CX-Programmer에서 읽어 확인할 수 있으며 파일로 저장할 수도 있습니다.

다음 두 가지 트레이스 방식을 사용할 수 있습니다.

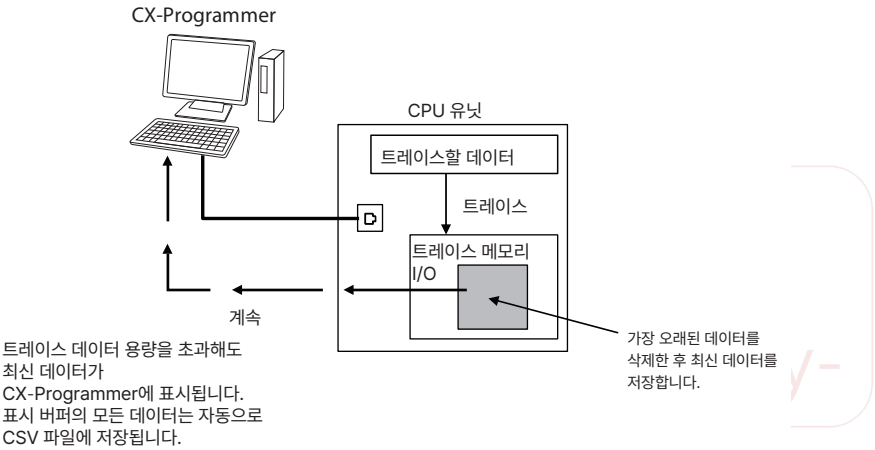
● 일반 트레이스

디버깅 등과 같이 단시간에 고속으로 데이터를 수집할 때 주로 이 방식을 사용합니다. 트레이스 메모리 용량을 초과하면 트레이스가 정지하고, 트레이스 메모리에 있는 현재 트레이스 결과가 표시되며, 동시에 데이터가 CSV 파일에 저장됩니다.



● 연속 트레이스

연속 트레이스는 주로 장시간에 걸쳐 고속으로 데이터를 수집할 때 사용합니다. 트레이스 메모리 용량을 초과해도 트레이스가 계속되며, 트레이스 데이터는 계속해서 CX-Programmer로 읽힙니다. 표시 버퍼가 가득 찰 때마다 데이터가 자동으로 CSV 파일에 저장됩니다. 이를 통해 트레이스 개시부터 종료까지의 장기 트레이스 결과 데이터를 여러 개의 CSV 파일에 저장할 수 있습니다.



● 데이터 트레이스 사양

항목		CJ3M CPU 유닛	CJ2H CPU 유닛	CJ2M CPU 유닛
메모리 용량		220000워드 데이터 트레이스 영역 대신 EM 영역의 일부를 사용할 수 있습니다.	CPU67/68(-EIP) 32000워드 CPU66(-EIP) 16000워드 CPU64/65(-EIP) 8000워드 데이터 트레이스 영역 대신 EM 영역의 일부를 사용할 수 있습니다.	8000워드 데이터 트레이스 영역 대신 EM 영역의 일부를 사용할 수 있습니다.
샘플링 사이즈	샘플링 비트	최대 31	최대 31	최대 31
	샘플링 워드	최대 32 1워드 데이터 = 최대 32 2워드 데이터 = 최대 16 4워드 데이터 = 최대 8	최대 16 1워드 데이터 = 최대 16 2워드 데이터 = 최대 8 4워드 데이터 = 최대 4	최대 16 1워드 데이터 = 최대 16 2워드 데이터 = 최대 8 4워드 데이터 = 최대 4
사이클		· 지정 샘플링 시간 1~2,550 ms (1 ms 단위) · 1사이클에 1회 · TRSM(045) 명령어 실행 시	· 지정 샘플링 시간 1~2,550 ms (1 ms 단위) · 1사이클에 1회 · TRSM(045) 명령어 실행 시 · 지정 샘플링 시간 0.5~10.0 ms (1 ms 단위)	· 지정 샘플링 시간 1~2,550 ms (1 ms 단위) · 1사이클에 1회 · TRSM(045) 명령어 실행 시
동기 동작 사이클에 1회		지원	지원	지원
연속 트레이스		지원 트레이스 메모리 용량을 초과하기 전에 트레이스 데이터를 CX-Programmer로 읽어 들이면 트레이스를 계속할 수 있습니다.	지원 트레이스 메모리 용량을 초과하기 전에 트레이스 데이터를 CX-Programmer로 읽어 들이면 트레이스를 계속할 수 있습니다.	지원 트레이스 메모리 용량을 초과하기 전에 트레이스 데이터를 CX-Programmer로 읽어 들이면 트레이스를 계속할 수 있습니다.
트리거 길이 설정용 데이터 길이		비트, 1워드, 2워드, 4워드의 데이터 길이	비트, 1워드, 2워드, 4워드의 데이터 길이	비트, 1워드, 2워드, 4워드의 데이터 길이
트리거 조건		· 비트 상승/하강 미분, ON, OFF 최대 4비트의 AND/OR 조건 조합 · 워드 =, >, ≥, <, ≤, ≠ 최대 2워드의 AND/OR 조건 조합 · 비트 조건과 워드 조건의 AND/OR 조건 조합	· 비트 상승/하강 미분 · 워드 =, >, ≥, <, ≤, ≠	· 비트 상승/하강 미분 · 워드 =, >, ≥, <, ≤, ≠

AI 번역본
-For Reference Only-

샘플링에는 다음 데이터형을 지정할 수 있습니다.

- BOOL (비트)
- UINT (1워드 부호 없는 2진)
- UDINT (2워드 부호 없는 2진)
- ULINT (4워드 부호 없는 2진)
- INT (1워드 부호 있는 2진)
- DINT (2워드 부호 있는 2진)
- LINT (4워드 부호 있는 2진)
- UINT BCD (1워드 부호 없는 2진)
- UDINT BCD (2워드 부호 없는 2진)
- ULINT BCD (4워드 부호 없는 2진)
- REAL (2워드 부동소수점)
- LREAL (4워드 부동소수점)
- CHANNEL (워드)
- WORD (1워드 16진)
- DWORD (2워드 16진)
- LWORD (4워드 16진)

샘플링 시간 기준

최대 31비트와 최대 32워드로 트레이스를 수행하는 경우

샘플링 시간		CJ3M CPU 유닛
	메모리 용량	220000워드
	샘플링 수	약 5,800
1 ms		5.8 s
10 ms		58 s
100 ms		580 s (약 9.6분)

최대 31비트와 최대 16워드로 트레이스를 수행하는 경우

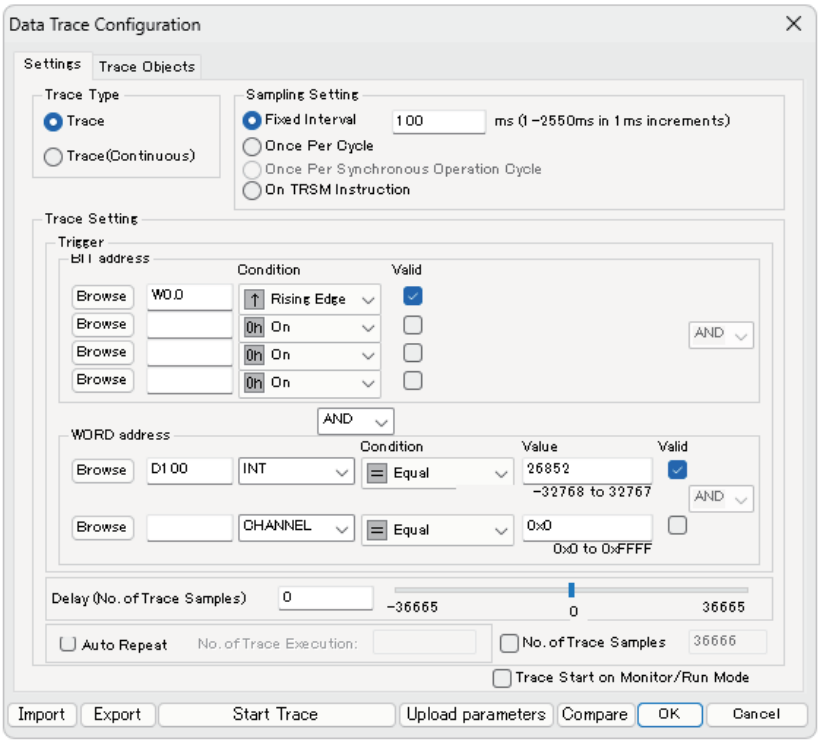
샘플링 시간		CJ3M CPU 유닛	CJ2H-CPU68(-EIP)	CJ2M CPU 유닛
	메모리 용량	220000워드	32000워드	8000워드
	샘플링 수	약 10,000	약 1,450	약 360
1 ms		10 s	1.45 s	0.36 s
10 ms		100 s	14.5 s	3.6 s
100 ms		1,000 s (약 16.6 분)	145 s (약 2.4분)	36 s

AI 번역본
-For Reference Only-

● 데이터 트레이스 순서 개요

트레이스를 실행하려면 다음 순서를 따르십시오.

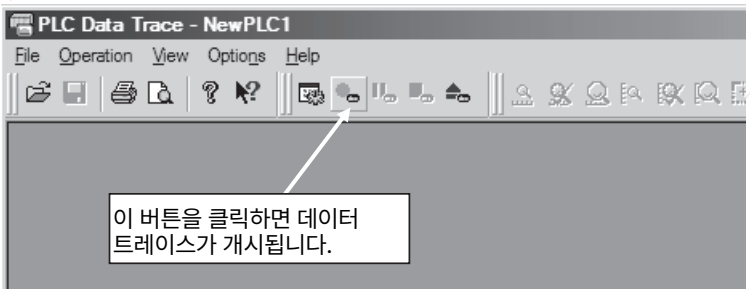
1 CX-Programmer로 트레이스 파라미터를 설정합니다. (PLC - Data Trace 또는 PLC - Time Chart Monitor를 선택하여 Data Trace Configuration 대화 상자를 열고, Execute - **Set을 선택합니다).**



· 일반 트레이스 실행
Trace Type 영역에서 Trace를 선택합니다. Data Trace Configuration 탭 페이지에서 샘플링할 데이터의 어드레스를 설정합니다. 여러 개의 어드레스를 지정할 수 있습니다. Settings 탭 페이지의 Trace Setting 영역에서 트리거 조건과 딜레이 값을 설정합니다.

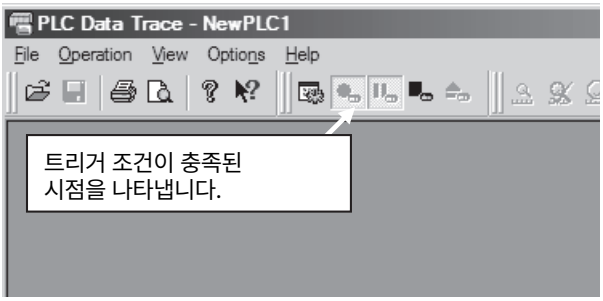
· 연속 트레이스 실행
Trace Type 영역에서 Trace (continuous)를 선택합니다. Data Trace Configuration 탭 페이지에서 샘플링할 데이터의 어드레스를 설정합니다. 여러 개의 어드레스를 지정할 수 있습니다. Settings 탭 페이지의 Trace Setting 영역에서 화면 표시 버퍼 크기, 트레이스 시간,
수집 데이터의 CSV 파일을 저장할 폴더를 지정합니다.

2 트레이스 개시 비트(A508.15)를 ON하거나 다음 버튼을 누릅니다. 트레이스가 개시됩니다.



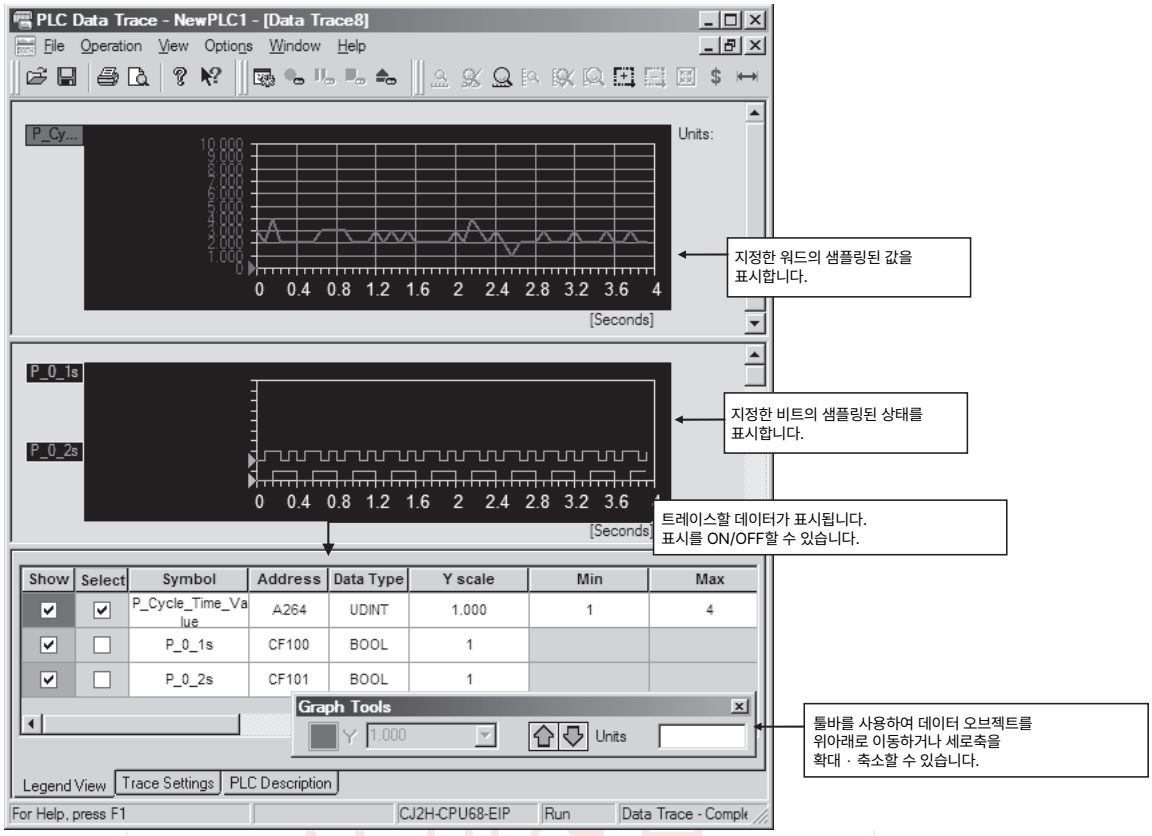
AI 번역본

트리거 조건이 충족되면 다음 표시가 나타납니다.



● Data Trace 윈도우

다음 그림은 Data Trace 윈도우의 개요를 나타냅니다.

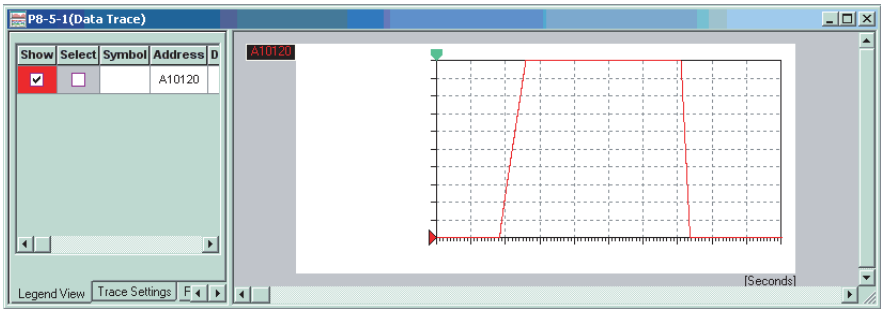


순서와 설정의 상세는 CX-Programmer 오퍼레이션 매뉴얼 (Cat. No. W446)을 참조하십시오.

AI 번역본
-For Reference Only-

I/O 모듈 보조 기억 영역 선택 기능

이 기능을 사용하면 CJ2M/CJ3M CPU 유닛에서 펄스 I/O 모듈을 사용할 때 펄스 I/O 모듈과 관련된 현재값과 상태를 간단히 선택할 수 있습니다. 트레이스할 현재값과 상태 플래그는 Data Trace Configuration 탭 페이지에서 I/O Module AR Select 버튼을 클릭하면 표시되는 목록에서 선택할 수 있습니다.



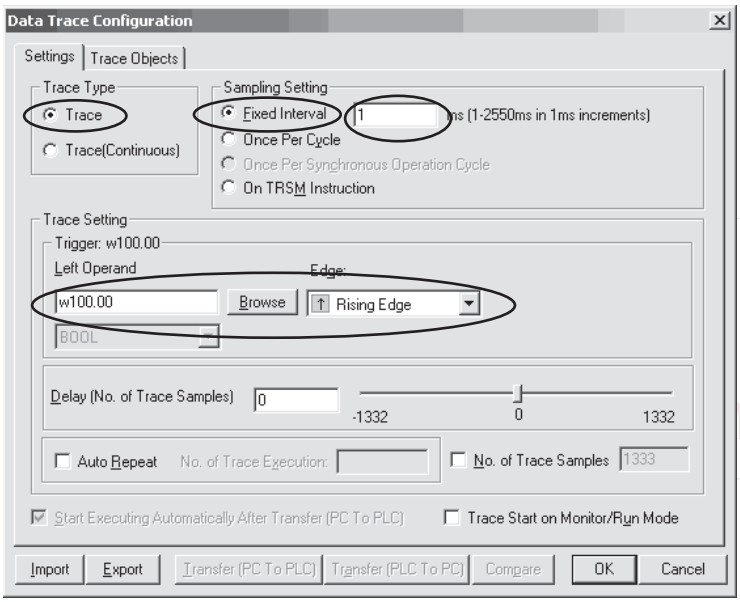
I/O 모듈 AR 선택 사용 순서

I/O 모듈 보조 기억 영역 선택 기능을 사용하려면 다음 순서를 따르십시오.

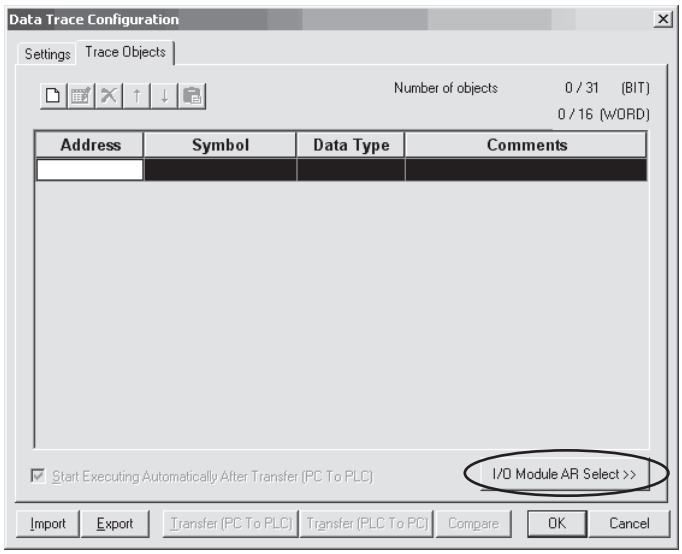
1 CX-Programmer로 트레이스 파라미터를 설정합니다.

(PLC - Data Trace 또는 PLC - Time Chart Monitor를 선택하여 Data Trace Configuration 윈도우를 열고 Execute - Set을 선택합니다.) 이 예는 1 ms 사이클로 펄스 출력 주파수를 트레이스하는 것을 보여 줍니다. 다음 설정은 Settings 탭에서 수행합니다.

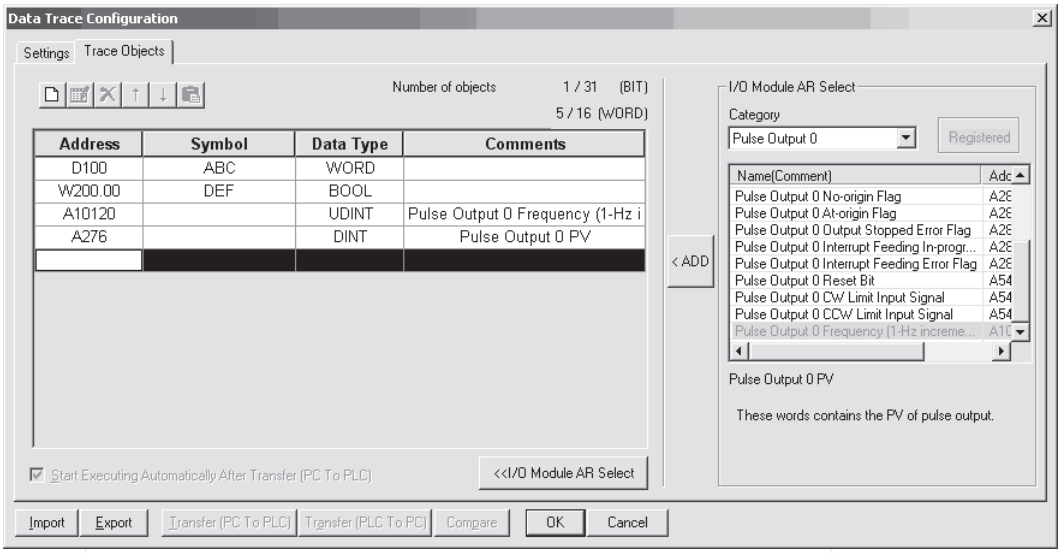
파라미터	CJ2 CPU 유닛
Trace Type	Trace
Sampling Setting	Fixed Interval: 1 ms
Trace Trigger	트레이스를 개시할 조건을 입력합니다.



2 Data Trace Configuration 탭 페이지에서 I/O Module AR Select 버튼을 클릭합니다.



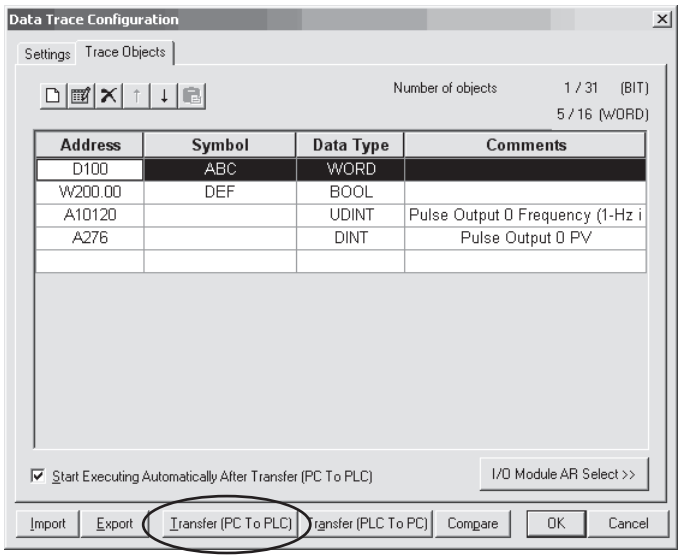
3 Data Trace Configuration 탭 페이지에서 트레이스할 펄스 출력을 선택하고 Add 버튼을 클릭합니다. 트레이스를 취소하려면 취소할 펄스 출력을 선택하고 Delete 버튼을 클릭합니다.



4 설정을 완료하고 OK 버튼을 클릭합니다.

AI 번역본
-For Reference Only-

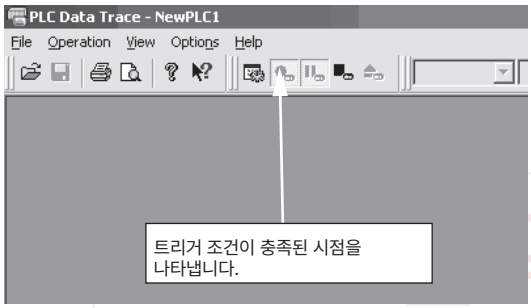
5 Transfer (PC to PLC) 버튼을 선택하여 설정을 CPU 유닛으로 전송합니다.



6 트레이스 개시 비트(A508.15)를 ON하거나 다음 버튼을 누릅니다.



트리거 조건이 충족되면 다음 표시가 나타납니다.



-For Reference Only-

10-7-7 에러 시 정지 위치 저장

래더 프로그램의 명령어가 입력 파라미터의 범위를 벗어나면 에러 플래그(P_ER)가 ON됩니다. 이 플래그를 참조하면 명령어 실행 시의 명령어 에러를 확인할 수 있습니다.

PLC 설정에서 Stop CPU on Instruction Error 체크 박스를 선택하면, 다음 명령어 에러 중 하나가 발생했을 때 프로그램 에러가 발생하고 동작이 정지합니다. 이 기능으로 명령어 에러가 발생했는지 확인할 수 있습니다.

● 프로그램 에러를 일으키는 명령어 에러 (A295)

이름	어드레스	설명
명령어 처리 에러 플래그	A295.08	명령어 처리 에러가 발생하고 PLC 설정이 명령어 에러 시 동작을 정지하도록 설정되어 있으면 이 플래그와 에러 플래그(P_ER)가 ON됩니다.
간접 DM/EM BCD 에러 플래그	A295.09	간접 DM/EM BCD 에러가 발생하고 PLC 설정이 간접 DM/EM BCD 에러 시 동작을 정지하도록 설정되어 있으면 이 플래그가 ON됩니다.
부정 액세스 에러 플래그	A295.10	부정 액세스 에러가 발생하면 이 플래그가 ON됩니다(예: I/O 메모리 영역 밖의 영역에 대한 액세스, 시스템이 예약한 영역에 대한 액세스, EM 파일로 변환된 영역에 대한 R/W 액세스, 쓰기 보호된 영역에 대한 쓰기 액세스, 간접 DM/EM BCD 에러).

프로그램 에러로 동작이 정지하면 정지 위치가 아래와 같이 저장됩니다.

● 프로그램 에러 태스크 (A294)

이름	어드레스	설명
프로그램 에러 태스크	A294	프로그램 에러로 프로그램 실행이 정지했을 때 실행 중이던 태스크의 종별과 번호를 나타냅니다. 0000~007F hex (태스크 0~127에 대응) 8000~80FF hex (엑스트라 사이클릭 태스크를 포함한 인터럽트 태스크 0~255에 대응)

치명 에러가 어느 태스크에서 발생했는지 확인할 수 있습니다. 치명 에러가 클리어되면 프로그램 에러 태스크도 클리어됩니다.

● 프로그램 에러 위치 (A299: 상위 바이트, A298: 하위 바이트)

프로그램 에러로 실행이 정지하면 정지 위치의 다음 프로그램 어드레스가 저장됩니다.

AI 번역본
-For Reference Only-

주 프로그램 어드레스는 래더 프로그램에서만 유효합니다. ST나 SFC 언어 프로그램에서는 사용할 수 없습니다.

- 래더 프로그램에서의 에러:
  래더 프로그램의 선두부터 센 어드레스
- 래더 액션 프로그램에서의 에러:
  래더 액션 프로그램의 선두부터 센 어드레스
- 트랜지션 프로그램에서의 에러:
  트랜지션 프로그램의 선두부터 센 어드레스
- 펑션 블록에서의 에러
  펑션 블록 프로그램의 선두부터 센 어드레스

10-7-8 고장 알람 명령어

FAL(006)과 FALS(007) 명령어는 사용자 정의 에러를 발생시킵니다. FAL(006)은 비치명 에러를, FALS(007)은 프로그램 실행을 정지시키는 치명 에러를 발생시킵니다.

사용자 정의 에러 조건(FAL(006) 또는 FAL(007)의 실행 조건)이 충족되면 고장 알람 명령어가 실행되고 다음 처리가 수행됩니다.

- FAL 에러 플래그(A402.15) 또는 FALS 에러 플래그(A401.06)가 ON됩니다.
- 해당 에러 코드가 보조 기억 영역의 A400에 쓰입니다.
- 에러 코드와 발생 시각이 에러 로그에 저장됩니다.
- CPU 유닛 전면의 에러 표시등이 점멸 또는 점등합니다.

FAL(006)이 실행된 경우 CPU 유닛은 계속 동작합니다.
FALS(007)이 실행된 경우 CPU 유닛은 동작을 정지합니다. (프로그램 실행이 정지합니다.)

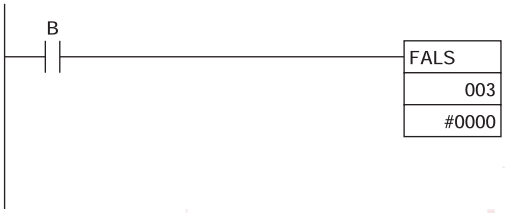
FAL(006)의 동작



실행 조건 A가 ON되면 FAL 번호 002의 에러가 발생하고 A402.15(FAL 에러 플래그)가 ON되며, A360.02(FAL 번호 002 플래그)가 ON됩니다. 프로그램 실행은 계속됩니다.

FAL(006)으로 발생한 에러는 FAL 번호 00으로 FAL(006)을 실행하거나 CX-Programmer에서 에러 읽기/클리어 조작을 수행하여 클리어할 수 있습니다.

FALS(007)의 동작



실행 조건 B가 ON되면 FALS 번호 003의 에러가 발생하고 A401.06(FALS 에러 플래그)이 ON됩니다. 프로그램 실행이 정지합니다.

FAL(006)으로 발생한 에러는 에러의 원인을 제거하고 CX-Programmer에서 에러 읽기/클리어 조작을 수행하여 클리어할 수 있습니다.

AI 번역본
-For Reference Only-

10-7-9 시스템 에러 시뮬레이션

FAL(006)과 FALS(007)을 사용하면 치명 시스템 에러와 비치명 시스템 에러를 의도적으로 만들 수 있습니다. 이는 시스템 디버깅에서 프로그래머블 터미널(PT)이나 그 밖의 오퍼레이터 인터페이스의 표시 메시지를 테스트하는 데 사용할 수 있습니다.

다음 순서를 따르십시오.

1 시뮬레이션에 사용할 FAL 또는 FALS 번호를 A529에 설정합니다. (A529는 FAL(006)과 FALS(007)의 에러를 시뮬레이션할 때 사용합니다.)

2 시뮬레이션에 사용할 FAL 또는 FALS 번호를 FAL(006) 또는 FALS(007)의 첫 번째 오퍼랜드로 설정합니다.

3 시뮬레이션할 에러 코드와 에러를 FAL(006) 또는 FALS(007)의 두 번째 오퍼랜드(S 및 S+1)로 설정합니다. FAL(006)에는 비치명 에러를, FALS(007)에는 치명 에러를 지정합니다.

여러 개의 시스템 에러를 시뮬레이션하려면 첫 번째 오퍼랜드에 A529와 동일한 값을 지정하고, 두 번째 오퍼랜드가 다른 FAL(006) 또는 FALS(007) 명령어를 여러 개 사용합니다.

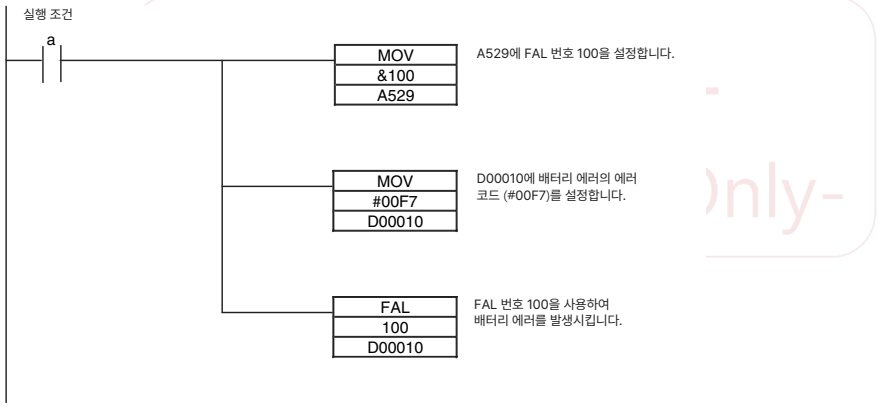
사용상의 주의

이 기능은 CPU 유닛에서 검출할 에러를 시뮬레이션하여 애플리케이션에서 검출이 올바르게 동작하는지 확인하는 데 사용합니다. 시스템을 운용할 때는 FAL(006) 또는 FALS(007) 명령어를 삭제하거나 입력 조건으로 항상 OFF(P_Off) 신호를 입력하여 이 기능을 무효로 하십시오.

● 보조 기억 영역 플래그 및 워드

이름	어드레스	동작
시스템 에러 시뮬레이션용 FAL/FALS 번호	A529	시스템 에러를 시뮬레이션하는 데 사용할 더미 FAL/FALS 번호를 설정합니다. 0001~01FF hex: FAL/FALS 번호 1~511 0000 또는 0200~FFFF hex: 시스템 에러 시뮬레이션용 FAL/FALS 번호 없음.

● 배터리 에러의 예



사용상의 주의

FAL(006)과 FALS(007) 명령어의 실행으로 시뮬레이션한 치명 시스템 에러와 비치명 시스템 에러를 클리어하려면 실제 시스템 에러와 동일한 방법을 사용하십시오. 에러 클리어 방법에 대해서는 CJ2 CPU 유닛 하드웨어 사용자 매뉴얼 (Cat. No. W472) 또는 CJ3 CPU 유닛 하드웨어 사용자 매뉴얼 (Cat. No. W664)의 Section 6 문제 해결을 참조하십시오. FAL(006)과 FALS(007)로 시뮬레이션한 모든 시스템 에러는 전원을 재투입하여 클리어할 수 있습니다.

10-7-10 고장점 검출

FPD(269) 명령어는 시간 감시와 논리 진단을 수행합니다. 시간 감시 기능은 지정한 감시 시간 안에 진단 출력이 ON되지 않으면 비치명 에러를 발생시킵니다. 논리 진단 기능은 어느 입력이 진단 출력이 ON되는 것을 막고 있는지 나타냅니다.

시간 감시 기능

FPD(269)는 실행되면 계시를 시작하고, 지정한 감시 시간 안에 진단 출력이 ON되지 않으면 캐리 플래그를 ON합니다. 캐리 플래그는 에러 처리 블록의 실행 조건으로 프로그래밍할 수 있습니다. 또한 FPD(269)는 원하는 FAL 번호로 비치명 FAL 에러를 발생시키도록 프로그래밍할 수 있습니다.

FAL 에러가 발생하면 미리 설정한 메시지가 등록되어 CX-Programmer에 표시할 수 있습니다. FPD(269)는 논리 진단 결과(진단 출력이 ON되는 것을 막고 있는 비트의 어드레스)를 메시지 바로 앞에 출력하도록 설정할 수 있습니다.

티칭 기능을 사용하면 진단 출력이 ON되기까지 실제로 걸리는 시간을 자동으로 구해 감시 시간을 설정할 수 있습니다.

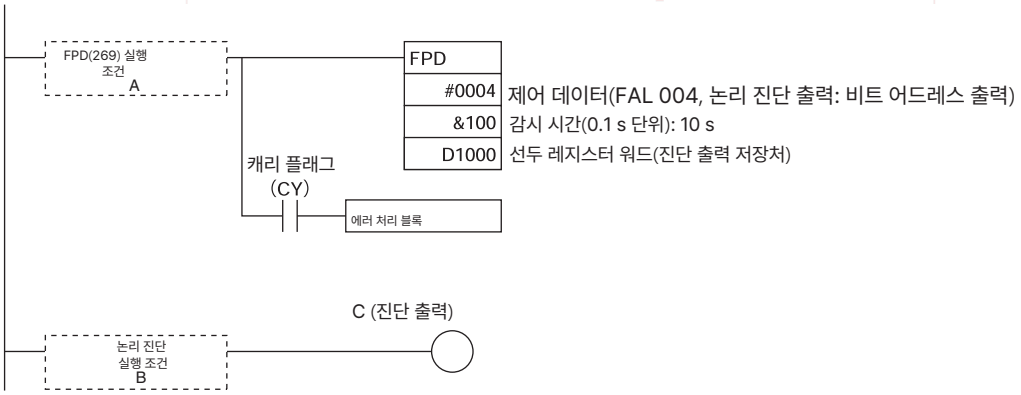
논리 진단 기능

FPD(269)는 어느 입력 비트가 진단 출력을 OFF 상태로 두고 있는지 판별하여 그 비트의 어드레스를 출력합니다. 출력은 비트 어드레스 출력(PLC 메모리 어드레스) 또는 메시지 출력(ASCII)으로 설정할 수 있습니다.

비트 어드레스 출력을 선택하면 그 비트의 PLC 메모리 어드레스를 인덱스 레지스터로 전송할 수 있고, 이후 처리에서 인덱스 레지스터를 간접 어드레스 지정할 수 있습니다.

메시지 출력을 선택하면 비트 어드레스가 ASCII 메시지에 등록되어 CX-Programmer에 표시할 수 있습니다.

AI 번역본



- **시간 감시:**

  입력 A 이후 10초 이내에 출력 C가 ON되는지 감시합니다. 10초 이내에 C가 ON되지 않으면 고장이 검출되고 캐리 플래그가 ON됩니다. 캐리 플래그가 에러 처리 블록을 실행합니다. 또한 FAL 번호 004의 FAL 에러(비치명 에러)가 발생합니다.

- **논리 진단:**

  FPD(269)는 블록 B의 어느 입력 비트가 출력 C의 ON을 막고 있는지 판별합니다. 그 비트 어드레스는 D1000에 출력됩니다.

- **보조 기억 영역 플래그 및 워드**

이름	어드레스	동작
에러 코드	A400	에러가 발생하면 그 에러 코드가 A400에 저장됩니다.
FAL 에러 플래그	A402.15	FAL(006)이 실행되면 ON됩니다.
FALS 에러 플래그	A401.06	FALS(007)이 실행되면 ON됩니다.
실행된 FAL 번호 플래그	A360~ A391	FAL(006) 또는 FALS(007) 에러가 발생하면 해당 플래그가 ON됩니다.
에러 로그 영역	A100~ A199	에러 로그 영역에는 최근 20건의 에러 정보가 들어 있습니다.
에러 로그 포인터	A300	에러가 발생하면 에러 로그 포인터가 1씩 증가하여, 다음 에러 기록이 어디에 기록될지를 에러 로그 영역(A100)의 선두부터의 오프셋으로 나타냅니다.
에러 로그 포인터 리셋 비트	A500.14	에러 로그 포인터(A300)를 00으로 리셋하려면 이 비트를 ON합니다.
FPD 티칭 비트	A598.00	FPD(269) 실행 시 감시 시간을 자동으로 설정하려면 이 비트를 ON합니다.

AI 번역본
-For Reference Only-

10-8 동기 유닛 동작

10-8-1 개요

동기 유닛 동작 기능

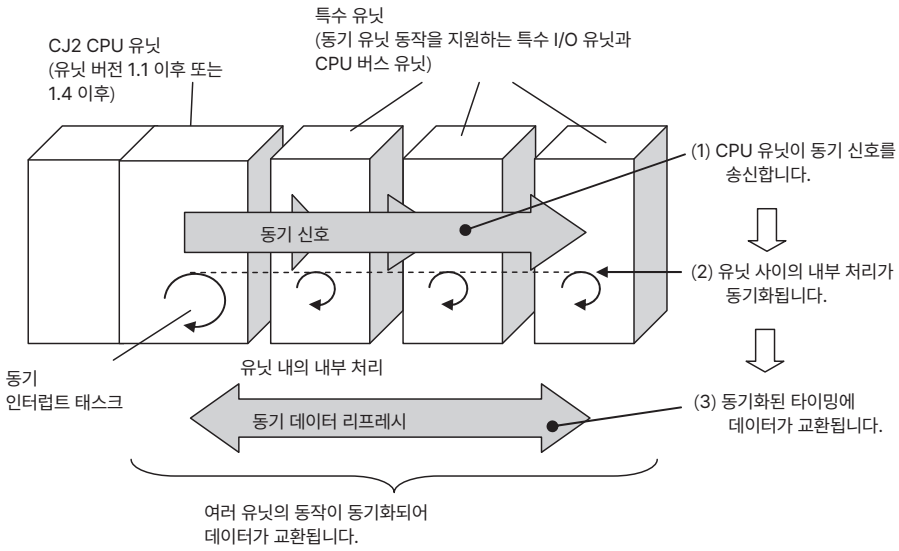
동기 유닛 동작 기능은 CPU 유닛이 지정한 사이클로 발생시키는 동기 신호를 사용하여 CPU 유닛과 여러 동기 유닛 사이의 처리 개시를 동기화하고, 이들 유닛 사이의 데이터 교환을 동기화합니다. 동기 유닛이란 동기 유닛 동작을 지원하는 CPU 버스 유닛과 특수 I/O 유닛입니다.

내부 처리 동기화

CPU 유닛은 동기 유닛과의 데이터 교환에 동기하여 인터럽트 태스크를 실행할 수 있습니다. 동기 유닛은 CPU 유닛에서 오는 동기 신호를 사용하여 모든 유닛 사이의 내부 처리 개시를 동기화할 수 있습니다.

데이터 교환 동기화

동기 신호가 송신되면 CPU 유닛과 여러 동기 유닛 사이 또는 동기 유닛끼리 데이터를 교환할 수 있습니다.



AI 번역본
-Japanese Only-

사용상의 주의

동기 유닛 동작 기능을 사용하려면 다음 요건을 모두 충족해야 합니다.

- CJ2H-CPU6□(-EIP) CPU 유닛은 특수 I/O 유닛의 경우 유닛 버전 1.1 이후, CPU 버스 유닛의 경우 유닛 버전 1.4 이후여야 합니다.
- 특수 I/O 유닛과 CPU 버스 유닛이 동기 유닛 동작을 지원해야 합니다. (이들을 동기 유닛이라고 합니다.)
- CJ1W-NC□82 위치 제어 유닛도 사용할 수 있지만 내부 처리의 개시는 동기화되지 않습니다. 유닛 사이의 데이터 교환을 동기화하는 기능만 사용할 수 있습니다. 유닛 버전 1.2 이후의 CJ1W-NC□82 위치 제어 유닛은 동기 유닛으로 등록할 수 있습니다.

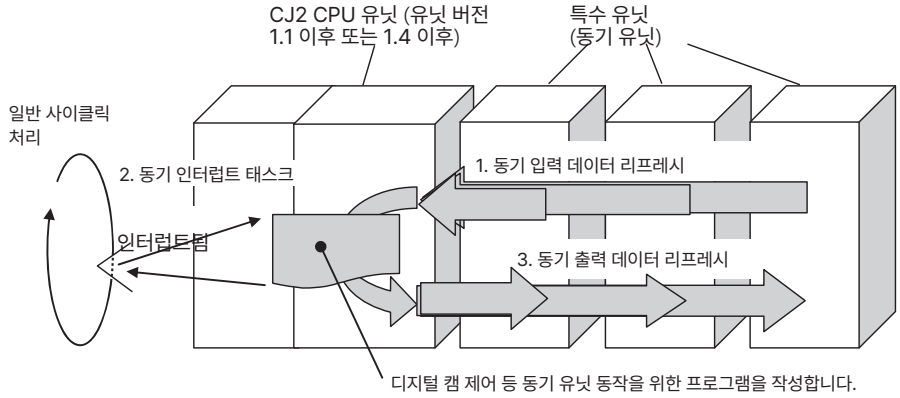
동기 유닛 동작의 전체 흐름

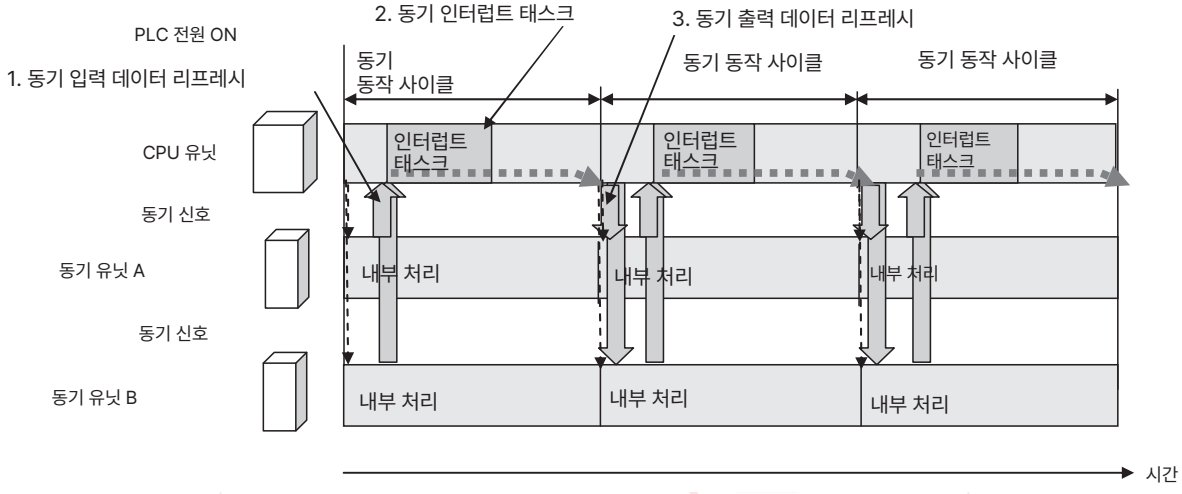
동기 유닛 동작은 다음 흐름으로 수행됩니다.

1 동기 유닛끼리 내부 처리를 동기화하고, 동기 동작 사이클마다 1회씩 동기 입력 데이터를 CPU 유닛으로 송신합니다.

2 CPU 유닛은 동기 데이터를 메모리에 저장하고 동기 유닛 동작용 인터럽트 태스크를 실행합니다. 이 인터럽트 태스크를 동기 인터럽트 태스크라고 합니다.

3 CPU 유닛은 동기 인터럽트 태스크의 결과를 동기 출력 데이터로서 모든 동기 유닛으로 송신합니다.





항목		설명
동기 데이터 리프레시		동기 데이터는 CPU 유닛의 일반 사이클 타임에 영향을 받지 않는 동기 동작 사이클로 CPU 유닛과 동기 유닛 사이에서 리프레시됩니다. 동기 데이터 리프레시를 위해 동기 데이터 리프레시 영역이라는 특수 영역이 할당됩니다. 이 영역의 데이터는 동기 데이터로서 유닛 사이에서 교환됩니다.
	동기 입력 데이터 리프레시	동기 데이터 리프레시의 입력 데이터는 동기 유닛에서 CPU 유닛으로 송신됩니다.
	동기 출력 데이터 리프레시	동기 데이터 리프레시의 출력 데이터는 CPU 유닛에서 동기 유닛으로 송신됩니다.
동기 동작 사이클		동기 동작 사이클은 동기 신호를 발생시키는 데 사용되며, 이 신호는 동기 유닛 내부 처리의 타이밍을 동기화하는 데 사용됩니다. 이 사이클은 동기 데이터를 리프레시하는 데에도 사용됩니다.

항목	설명
동기 인터럽트 태스크	동기 인터럽트 태스크는 CPU 유닛이 동기 데이터 리프레시의 입력 데이터를 받은 후에 실행됩니다. 동기 인터럽트 태스크에는 인터럽트 태스크 2(정주기 인터럽트 0)가 사용됩니다. 동기 인터럽트 태스크를 사용하지 않고도 동기 유닛 동작을 사용할 수 있습니다.

동기 유닛 동작 설정 개요

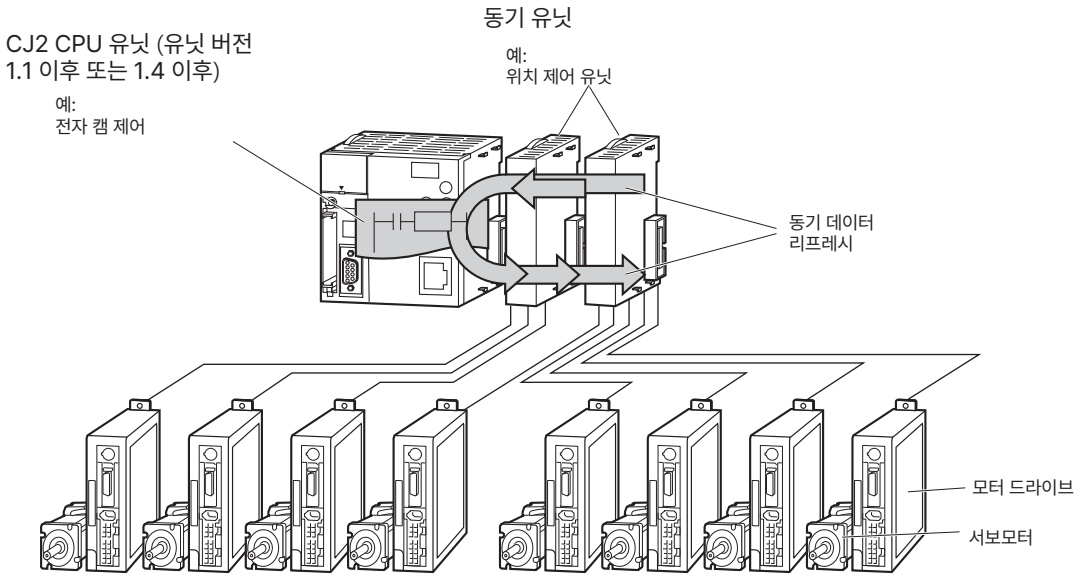
동기 유닛 동작에는 다음 설정을 사용합니다.

항목	설명	설정 방법
동기 유닛 동작	동기 유닛 동작을 유효/무효로 할 수 있습니다.	PLC 설정
동기 데이터 리프레시	・동기 유닛 동작에 사용할 동기 유닛을 등록해야 합니다. ・동기 데이터 리프레시 영역을 할당해야 합니다.	PLC 설정
	각 동기 유닛의 동기 데이터를 설정해야 합니다.	각 동기 유닛의 동기 데이터 설정은 PLC 설정에서 액세스합니다
동기 동작 사이클	동기 사이클 타임을 설정합니다.	PLC 설정
동기 인터럽트 태스크	이 태스크의 프로그램이 실행됩니다.	프로그램 프로퍼티에서 인터럽트 태스크 2(정주기 인터럽트 0)를 선택하십시오. 동기 인터럽트 태스크를 사용하지 않고도 동기 유닛 동작을 사용할 수 있습니다.

적용 예

● 서보모터 사이의 동작 동기화

이 애플리케이션에서는 엔코더의 동작에 따라 여러 서보모터의 동작을 동기화합니다. 여기서는 고속 카운터 유닛에서 CPU 유닛으로 입력된 동기 데이터를 동기 인터럽트 태스크의 동기 동작 프로그램(예: 디지털 캠 프로그램)에서 처리합니다. 그 결과는 위치 제어에 사용할 동기 데이터로서 CPU 유닛에서 여러 위치 제어 유닛으로 출력됩니다.



참고

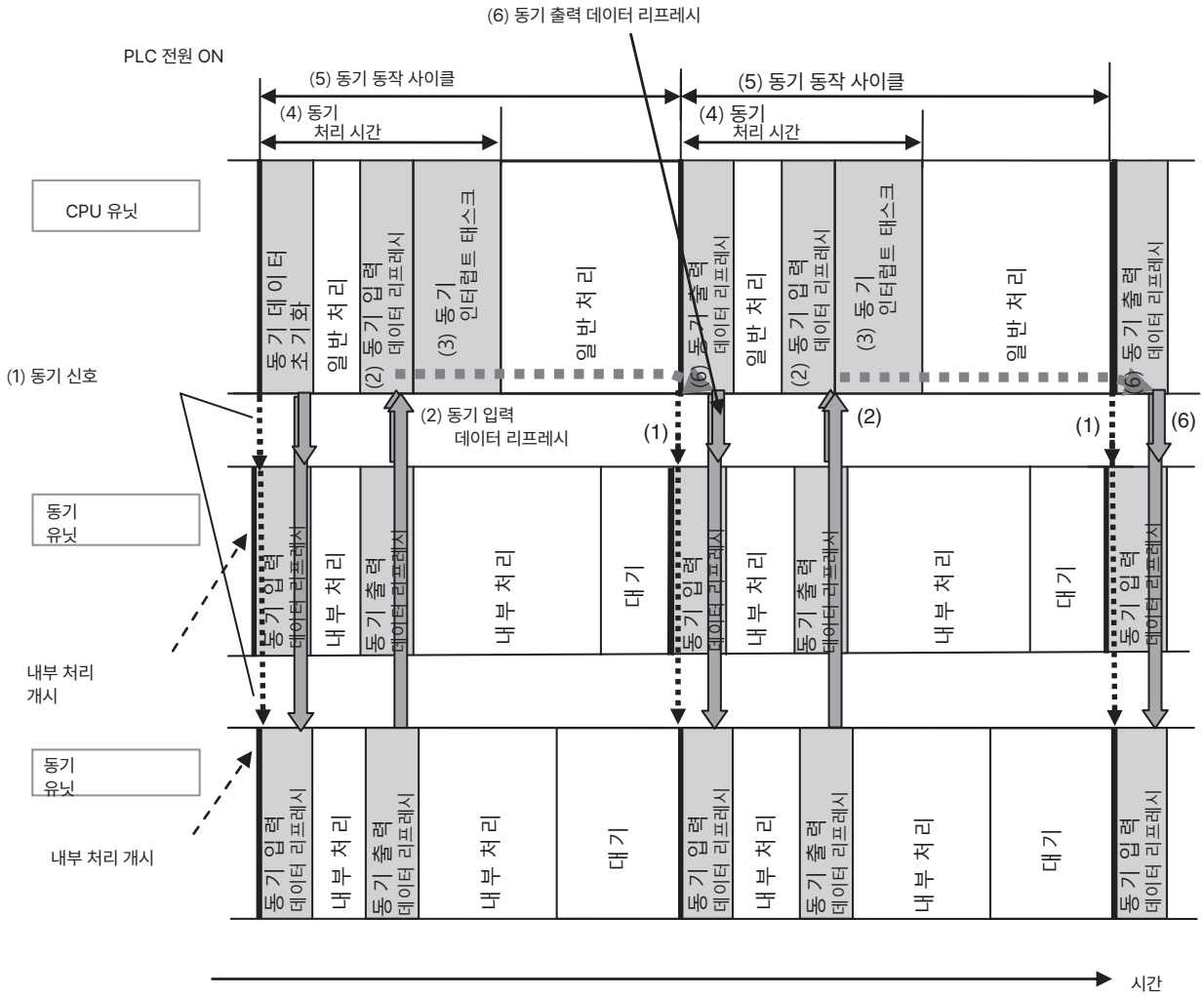
CPU 유닛과 동기 유닛 사이에서는 일반 I/O 리프레시도 수행됩니다. 일반 I/O 리프레시와 동기 데이터 리프레시의 차이는 다음 표와 같습니다.

항목	동기 데이터 리프레시	일반 I/O 리프레시
데이터 교환 타이밍	· 동기 동작 사이클마다. (동기 데이터 리프레시를 수행하기 위해 CPU 유닛의 일반 사이클이 인터럽트됩니다.) · 데이터 교환은 동기 유닛의 내부 처리와 동기화됩니다. CJ1W-NC□82도 사용할 수 있지만 내부 처리의 개시는 동기화되지 않습니다. 유닛 사이의 데이터 교환을 동기화하는 기능만 사용할 수 있습니다.	· CPU 유닛의 일반 사이클마다 (사용자 프로그램 실행 후). · 일반 I/O 리프레시의 타이밍은 동기 유닛 내부 처리의 타이밍과 관계가 없습니다.
데이터 교환 사이클	일정(항상 지정한 동기 동작 사이클).	일정하지 않음(최소 사이클 타임이 설정되어 있고 사이클이 설정 시간을 넘지 않으면 최소 사이클 타임).
동기 유닛의 내부 처리 개시	동기화됩니다.	동기화되지 않습니다.

AI 번역본
-For Reference Only-

10-8-2 동기 유닛 동작 상세

동기 유닛 동작의 타임 시퀀스를 다음 그림에 나타냅니다.



1 동기 신호

PLC의 전원이 ON되면 CPU 유닛이 동기 신호를 발생시킨 후 동기 데이터의 초기화를 시작합니다. 이때 CPU 유닛과 동기 유닛도 내부 처리를 시작합니다.

AI 번역본
For Reference Only-

2 동기 입력 데이터 리프레시 (동기 유닛 → CPU 유닛)

동기 유닛이 CPU 유닛으로 동기 데이터를 송신했습니다.

3 동기 인터럽트 태스크

CPU 유닛이 동기 인터럽트 태스크를 실행했습니다. (동기 인터럽트 태스크를 사용하지 않고도 동기 유닛 동작을 사용할 수 있습니다.)

사용상의 주의

CJ1W-NC□82 위치 제어 유닛은 내부 처리의 개시가 동기화되지 않습니다. 따라서 위에 나타낸 CPU 유닛으로부터의 동기 입력 데이터 리프레시는 해당 유닛의 제어 사이클 중에 발생하며, 그때 동기 I/O 데이터 리프레시가 처리됩니다.

4 동기 처리 시간

동기 처리 시간은 하나의 동기 동작 사이클에서 일련의 동기 동작 처리에 필요한 시간입니다. 동기 신호가 발생한 때부터 동기 인터럽트 태스크의 실행이 완료될 때까지의 시간입니다. 동기 처리 시간 이후에는 동기 제어 사이클이 만료될 때까지 일반 처리가 수행됩니다.

동기 처리 시간은 동기 동작 사이클보다 짧아야 합니다. (동기 처리 시간의 최대값과 현재값은 CX-Programmer의 Synchronous Operation Status 대화 상자에서 모니터할 수 있습니다.)

5 동기 동작 사이클

동기 동작 사이클 타임이 만료되면 동기 신호가 다시 발생합니다.

6 동기 출력 데이터 리프레시 (CPU 유닛 → 동기 유닛)

CPU 유닛이 동기 유닛으로 동기 데이터를 송신합니다.

7 전체 처리가 순서 2부터 반복됩니다.

사용상의 주의

CPU 유닛은 동기 입력 데이터 리프레시, 동기 인터럽트 태스크 실행, 동기 출력 데이터 리프레시 중에 일반 처리를 인터럽트합니다. CPU 유닛의 일반 사이클 타임은 이러한 처리에 필요한 시간만큼 길어집니다. 동기 유닛 동작을 사용하기 전에, 사이클 타임이 길어져도 제어 시스템에 악영향이 없는지 반드시 확인하십시오.

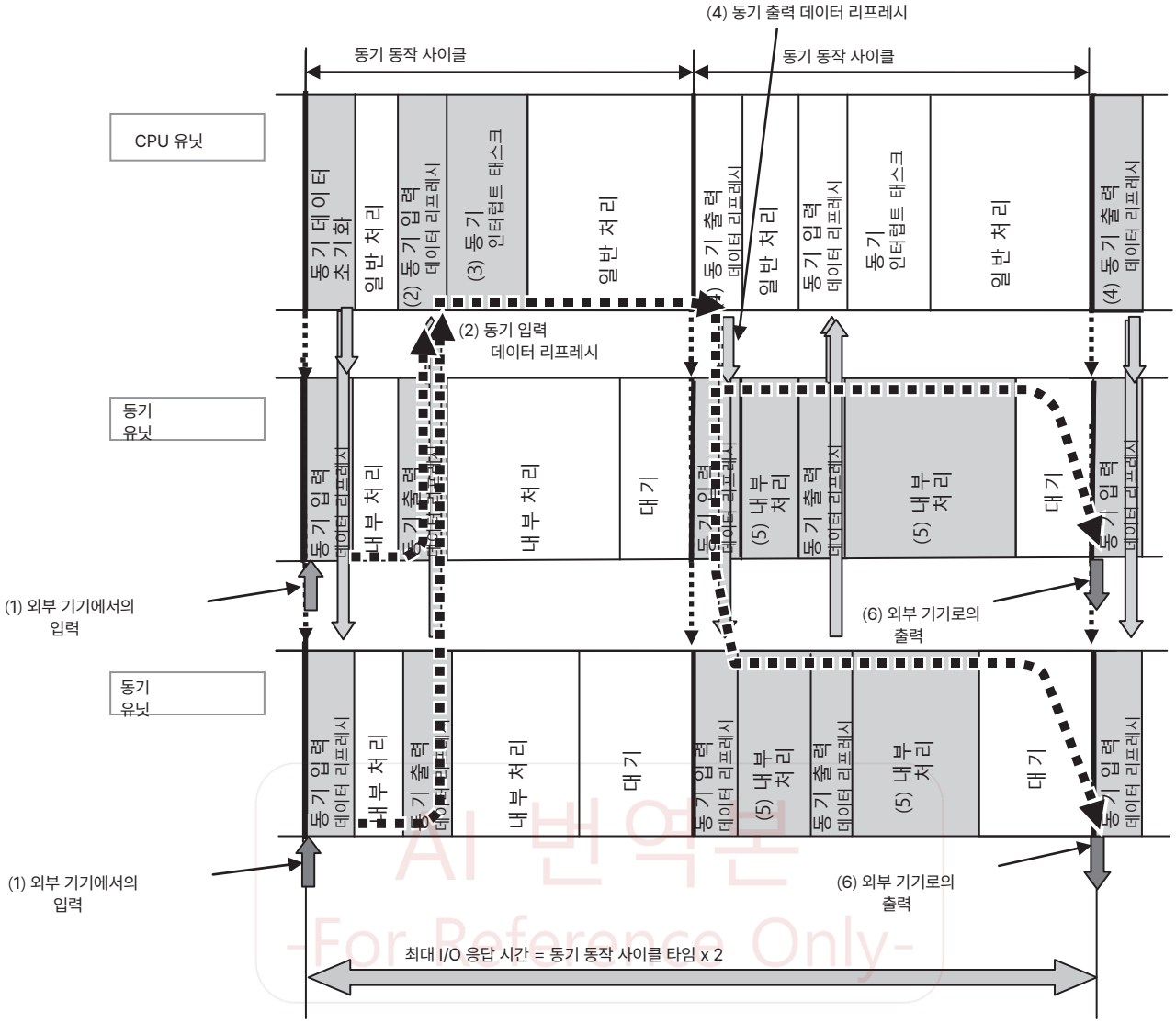
AI 번역본
-For Reference Only-

● 동기 유닛의 I/O 응답 시간

동기 유닛의 I/O 응답 시간은 다음 처리에 필요한 시간의 합계입니다.

(1) 외부 기기에서 동기 유닛으로의 입력
(2) 동기 입력 데이터 리프레시 (동기 유닛 → CPU 유닛)
(3) 동기 인터럽트 태스크 실행
(4) 동기 출력 데이터 리프레시 (CPU 유닛 → 동기 유닛)
(5) 동기 유닛 내의 내부 처리
(6) 동기 유닛에서 외부 기기로의 출력

따라서 최대 I/O 응답 시간은 동기 동작 사이클 타임의 2배입니다.



AI 번역본
-For Reference Only-

CJ1W-NC□□4 위치 제어 유닛의 예

이 유닛의 내부 처리 시간은 1 ms이므로 동기 동작 사이클 타임은 최소 1 ms여야 합니다. 따라서 이 위치 제어 유닛의 I/O 응답 시간은 항상 2 ms가 됩니다.

사용상의 주의

CJ1W-NC□82 위치 제어 유닛은 내부 처리의 개시가 동기화되지 않습니다.
따라서 위에 나타낸 CPU 유닛으로부터의 동기 입력 데이터 리프레시는 해당 유닛의 제어 사이클 중에 발생하며, 동기 I/O 데이터 리프레시도 그때 처리됩니다.
동기 동작 사이클과 유닛의 제어 사이클을 모두 1 ms로 설정하면 외부 기기로의 출력이 위에 나타낸 1 동기 동작 사이클보다 더 지연될 수 있습니다. I/O 응답 시간은 2~3 ms가 됩니다. 통신 네트워크를 사용하는 CJ1W-NC□82에서는 (1) 외부 기기로부터의 입력과
(6) 외부 기기로의 출력이 유닛이 통신 네트워크 데이터를 송수신하는 것을 가리킵니다.

AI 번역본
-For Reference Only-

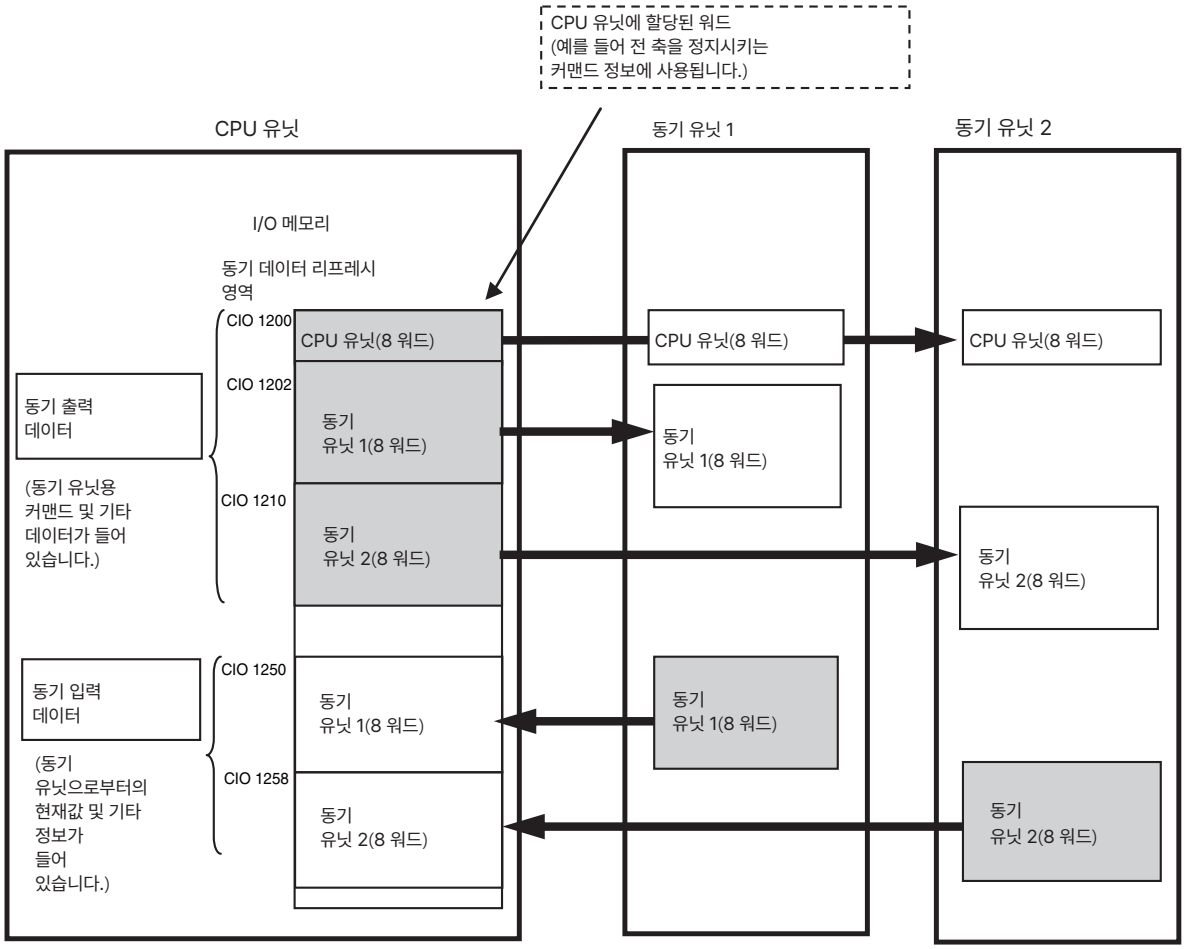
10-8-3 동기 유닛 동작 사양

항목	사양
적용 CPU 유닛	CJ2H-CPU6□(-EIP) CPU 유닛은 특수 I/O 유닛의 경우 유닛 버전 1.1 이상, CPU 버스 유닛의 경우 유닛 버전 1.4 이상이어야 합니다.
동기 유닛으로 사용할 수 있는 특수 I/O 유닛 및 CPU 버스 유닛	CJ1W-NC□□4 위치 제어 유닛 * CJ1W-NC□82 위치 제어 유닛도 사용할 수 있지만, 내부 처리의 개시는 동기화되지 않습니다. 유닛 간의 데이터 교환을 동기화하는 기능만 사용할 수 있습니다. 유닛 버전 1.2 이상의 CJ1W-NC□82 위치 제어 유닛은 동기 유닛으로 등록할 수 있습니다.
장착 가능한 동기 유닛 수	최대 10 유닛 CJ1W-NC□□4 위치 제어 유닛의 경우 최대 5 유닛
동기 유닛의 장착 위치	동기 유닛은 CPU 랙에 장착해야 합니다. (증설 랙에 장착하면 동작하지 않습니다.)
동기 동작 사이클 타임	0.5~10 ms(0.1 ms 단위) PLC 설정에서 설정합니다. 다음 조건을 만족해야 합니다: ・동기 처리 시간이 동기 동작 사이클 타임보다 짧아야 합니다. ・동기 유닛의 내부 처리 시간이 동기 동작 사이클 타임보다 짧아야 합니다. * 동기 유닛에 CJ1W-NC□82가 포함된 경우, 가장 긴 유닛 제어 사이클이 동기 동작 사이클보다 짧은지 확인하십시오.
동기 동작 사이클 타임의 최대 변동	10 μs
CPU 유닛의 동기 데이터 교환용 I/O 메모리 영역	동기 데이터 리프레시 영역은 두 부분으로 나뉘며, 하나는 동기 유닛에서 CPU 유닛으로의 입력 데이터용이고 다른 하나는 CPU 유닛에서 동기 유닛으로의 출력 데이터용입니다. 동기 데이터 리프레시 영역은 CIO 1200~CIO 1295입니다. 내용은 동기 유닛에 따라 달라집니다.
동기 인터럽트 태스크의 태스크 번호	인터럽트 태스크 번호 2를 사용해야 합니다. 동기 인터럽트 태스크를 사용하지 않고도 동기 유닛 동작을 사용할 수 있습니다.

AI 번역본
-For Reference Only-

10-8-4 동기 데이터 리프레시

CPU 유닛의 CIO 영역에 있는 동기 데이터 리프레시 영역은 항상 CIO 1200에서 시작합니다. CPU 유닛에서 동기 유닛으로 보내는 출력 데이터가 먼저 오고, 이어서 동기 유닛에서 CPU 유닛으로 보내는 입력 데이터가 옵니다.



다음 설정은 PLC 설정의 Timings/Synchronous 탭 페이지에 있는 동기 유닛 동작 설정 영역에서 수행합니다.

- 출력 데이터의 총 사이즈와 입력 데이터의 총 사이즈
- 각 유닛의 출력 사이즈와 입력 사이즈
- 입력 영역의 선두 어드레스

AI 번역본
-For Reference Only-

동기 데이터 리프레시 영역의 어드레스는 CIO 1200~CIO 1295입니다. 이 영역의 선두 워드는 항상 CIO 1200입니다. 그 밖의 어드레스는 설정할 수 있습니다.

I/O	항목	설명
출력 데이터	방향	CPU 유닛에서 동기 유닛으로
	선두 워드	CIO 1200(고정)
	할당	PLC 설정에서 지정한 데이터 사이즈가 먼저 CPU 유닛에 할당되고, 이어서 동기 유닛이 등록된 순서대로 동기 유닛에 할당됩니다. CPU 유닛에는 2 워드 이상을 할당할 수 있고, 각 동기 유닛에는 0 워드 이상을 할당할 수 있습니다.
입력 데이터	방향	동기 유닛에서 CPU 유닛으로
	선두 워드	선두 워드의 어드레스는 PLC 설정에서 설정합니다. CIO 1202~CIO 1294 범위에서 설정할 수 있습니다.
	할당	PLC 설정에서 지정한 데이터 사이즈가 동기 유닛이 등록된 순서대로 동기 유닛에 할당됩니다. 특수 I/O 유닛의 경우 동기 유닛마다 0~16 워드를 할당할 수 있습니다. CPU 버스 유닛의 경우 동기 유닛마다 0~64 워드를 할당할 수 있습니다.

● 할당 예

동기 데이터의 입력 데이터 및 출력 데이터 워드 할당 예

동기 데이터 리프레시를 위해 선두 어드레스와 데이터 사이즈를 설정합니다. (이 예에서는 유닛 번호 0과 1을 사용합니다.)

어드레스	항목	I/O
CIO 1200	CPU 유닛에 할당된 워드(데이터 사이즈 설정이 2 워드인 경우)	CPU 유닛에서 동기 유닛으로의 출력
CIO 1201		
CIO 1202	동기 유닛 0에 할당된 워드(데이터 사이즈 설정이 2 워드인 경우)	
CIO 1203		
CIO 1204	동기 유닛 1에 할당된 워드(데이터 사이즈 설정이 4 워드인 경우)	
CIO 1205		
CIO 1206		
CIO 1207		
:	:	
CIO 1250	동기 유닛 0에 할당된 워드(데이터 사이즈 설정이 4 워드인 경우)	동기 유닛에서 CPU 유닛으로의 입력
CIO 1251		
CIO 1252		
CIO 1253		
CIO 1254	동기 유닛 1에 할당된 워드(데이터 사이즈 설정이 2 워드인 경우)	
CIO 1255		
:	:	
CIO 1295	...	

AI 번역본
-For Reference Only-

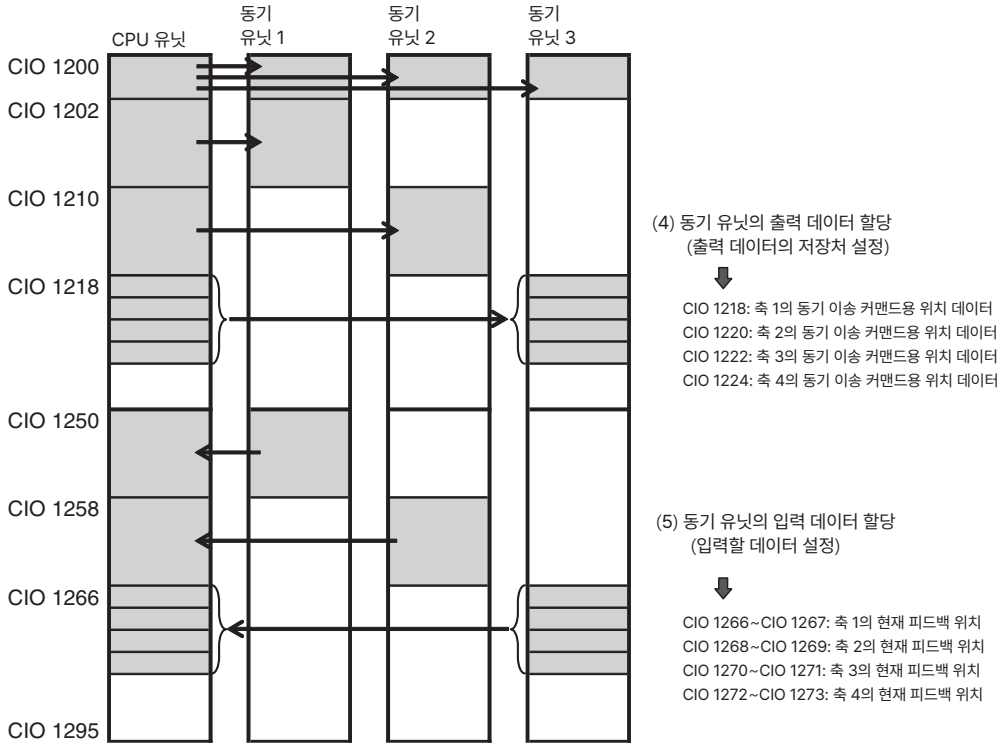
위치 제어 유닛을 사용한 동기 데이터 리프레시의 I/O 데이터 설정 예

할당된 I/O 데이터의 용도를 설정해야 합니다.

・출력 데이터
위치 제어 유닛에서는 동기 이송 커맨드의 위치 데이터에 사용되는 워드를 축마다 설정합니다.

・입력 데이터
위치 제어 유닛에서는 축마다 현재 지령값 또는 현재 피드백 위치를 설정합니다.

설정 예

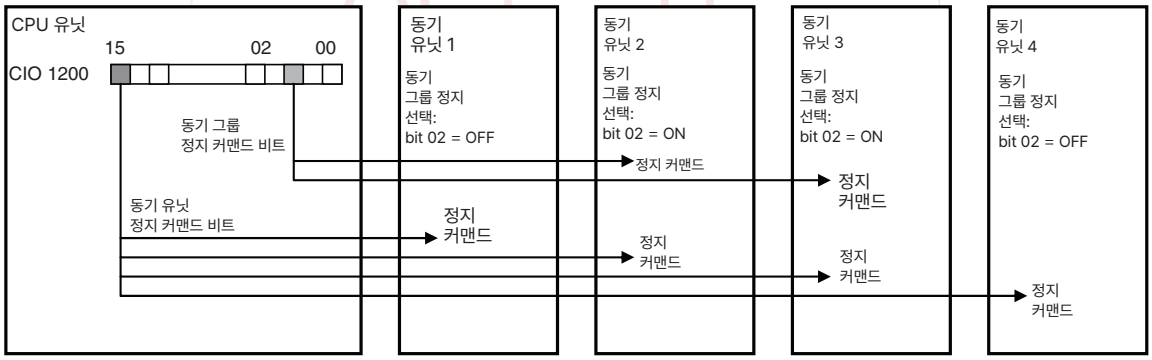


● 적용 예

위치 제어 유닛과 동기 유닛을 사용하는 경우 I/O 데이터는 다음 표와 같이 사용됩니다.

데이터	적용 예
CPU 유닛에 할당된 워드	동기 그룹 정지 커맨드 비트, 동기 유닛 정지 커맨드 비트* 등.
동기 데이터 리프레시의 출력 데이터	목표 위치 데이터(X, Y, Z, U 축)
동기 데이터 리프레시의 입력 데이터	현재 피드백 데이터(X, Y, Z, U 축)

* 동기 그룹 정지 커맨드 비트와 동기 유닛 정지 커맨드 비트
동기 유닛 동작을 수행 중인 동기 유닛에 대해 현재 실행 중인 기능을 정지시키기 위해 CPU 유닛에서 커맨드 비트를 사용할 수 있습니다. 커맨드 비트는 두 가지입니다. 동기 유닛 정지 커맨드 비트는 모든 동기 유닛에 적용됩니다. 동기 그룹 정지 커맨드 비트는 미리 설정된 동기 유닛 그룹에 적용됩니다.



동기 유닛 정지 커맨드 비트

동기 데이터 리프레시 영역의 CIO 1200 비트 15는 동기 유닛 정지 커맨드 비트입니다. 이 비트를 ON 하면 모든 동기 유닛이 현재 수행 중인 동작을 정지시키는 커맨드가 송신됩니다.

동기 그룹 정지 커맨드 비트

동기 데이터 리프레시 영역의 CIO 1200 비트 00~14는 동기 그룹 정지 커맨드 비트입니다. 이들 비트 중 하나를 ON 하면, 동기 유닛의 동기 그룹 정지 선택 파라미터 설정에서 같은 비트가 ON 되어 있는 모든 동기 유닛이 현재 수행 중인 동작을 정지시키는 커맨드가 송신됩니다.

위치 제어 유닛의 축 파라미터에 있는 동기 그룹 정지 선택 파라미터 설정은 다음 표와 같습니다.

명칭	비트	기능	설정	디폴트
동기 그룹 정지 선택	00	이들 비트는 동기 데이터 리프레시 영역의 CIO 1200의 각 비트에 대한 동기 유닛의 동작을 지정합니다. OFF: 비트 00을 무시합니다. ON: 비트 00이 ON 되면 정지합니다.	OFF, ON	OFF
	01	OFF: 비트 01을 무시합니다. ON: 비트 01이 ON 되면 정지합니다.	OFF, ON	OFF
	02	OFF: 비트 02를 무시합니다. ON: 비트 02가 ON 되면 정지합니다.	OFF, ON	OFF
	:	:		
	14	OFF: 비트 14를 무시합니다. ON: 비트 14가 ON 되면 정지합니다.	OFF, ON	OFF
	15	---	---	---

AI 번역본
-For Reference Only-

10-8-5 동기 유닛 동작 사용 시의 제한 사항

● 동기 동작 사이클 타임에 대한 제한

다음 두 조건을 만족하도록 동기 동작 사이클 타임을 설정하십시오.

1. 동기 처리 시간은 동기 동작 사이클 타임보다 짧아야 합니다.

* 일반적으로 동기 동작 사이클 타임은 동기 처리 시간의 1.5배 이상으로 설정하십시오.

2. 동기 유닛의 내부 처리 시간은 동기 사이클 타임보다 짧아야 합니다.

* 동기 유닛에 CJ1W-NC□82가 포함된 경우, 가장 긴 유닛 제어 사이클이 동기 동작 사이클보다 짧은지 확인하십시오.

1. 동기 처리 시간

동기 처리 시간은 다음 두 시간의 합계입니다.

(1) 동기 신호가 발생한 시점부터 동기 인터럽트 태스크의 실행이 완료될 때까지의 시간(즉, 동기 출력 데이터 리프레시 시간, 일반 처리 시간, 동기 입력 데이터 리프레시 시간, 동기 입력 태스크 실행 시간의 합계)

(2) I/O 인터럽트 태스크나 외부 인터럽트 태스크를 실행하는 데 필요한 시간. 이들 태스크는 위 (1)에서 설명한 처리 중에 발생합니다. (동기 처리 시간의 최댓값과 현재값은 CX-Programmer의 Synchronous Operation Status 대화 상자에서 모니터링할 수 있습니다.)

2. 동기 유닛의 내부 처리 시간

CJ1W-NC□□4 위치 제어 유닛의 내부 처리 시간은 1 ms이므로, 동기 동작 사이클 타임은 최소 1 ms 이상이어야 합니다.

3. 위 조건을 만족하지 못하는 경우

동기 동작 사이클 타임이 동기 처리 시간보다 짧은 경우

동기 인터럽트 태스크, I/O 인터럽트 태스크 또는 외부 인터럽트 태스크의 실행 중에 다음 동기 동작 사이클에 들어갑니다. 이로 인해 CPU 유닛에서 동기 처리 시간 오버 에러가 발생합니다. 그 동기 동작 사이클에서는 동기 인터럽트 태스크와 동기 데이터 리프레시가 한 번 실행되지 않습니다.

동기 동작 사이클 타임이 동기 유닛의 내부 처리 시간보다 짧은 경우

동기 유닛이 내부 처리를 완료하기 전에 다음 동기 동작 사이클에 들어갑니다. 이로 인해 동기 유닛에서 동기 처리 시간 오버 에러가 발생합니다. 동기 유닛에서는 처리가 계속됩니다(예: 위치 제어 유닛에서는 축 동작이 계속됩니다).

-For Reference Only-

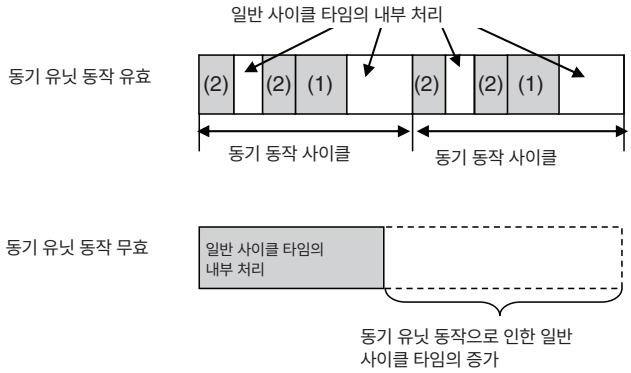
● 일반 사이클 타임의 증가

동기 유닛 동작을 사용하면 일반 사이클 타임이 다음 시간만큼 증가합니다.

(1) 동기 인터럽트 시간의 실행 시간(이는 일반 정주기 인터럽트의 경우와 동일합니다.)

(2) 동기 입력 및 출력 데이터 리프레시 시간

주 위 시간은 동기 처리 시간에서 그동안 수행된 일반 처리 시간을 뺀 것과 같습니다.



● 정주기 인터럽트 태스크 0과 1의 사용 불가

동기 유닛 동작을 사용하는 경우 정주기 인터럽트 0과 1로 인터럽트 태스크를 실행할 수 없습니다. 따라서 정주기 인터럽트 태스크 0(인터럽트 태스크 2)과 정주기 인터럽트 태스크 1(인터럽트 태스크 3)은 사용할 수 없습니다. 다만 인터럽트 태스크 2는 동기 인터럽트 태스크로 사용됩니다.

● 인터럽트 제어 명령어에 대한 제한

동기 인터럽트 태스크에서는 MSKS, MSKR, CLI 명령어를 사용할 수 없습니다. DI와 EI 명령어는 사용할 수 있습니다.

● 타이머 명령어에 대한 제한

다음 제한은 100 ms 타이머(TIM/TIMX(550)), 10 ms 타이머(TIMH(015)/TIMHX(551)), 1 ms 타이머(TMHH(540)/TMHHX(552)) 명령어 사용에 적용됩니다.

- 타이머 현재값 정밀도에 최대 1 사이클 타임의 오차가 발생합니다.
- 사이클 타임이 100 ms를 초과하면 타이머가 올바르게 동작하지 않습니다.
- 위 명령어 중 하나가 정지된 태스크에 있거나 JMP(004), CJMP(510), CJPN(511) 명령어로 점프되어 실행되지 않는 경우, 타이머는 올바르게 동작하지 않습니다.

● 고속 인터럽트 기능이 유효할 때 인터럽트 태스크에서의 명령어 사용 제한

동기 유닛 동작 기능의 사용이 유효화되면 PLC 설정에서 고속 인터럽트 기능이 무조건 유효화됩니다. 따라서 네트워크 통신 명령어 등 일부 명령어는 동기 인터럽트 태스크를 포함한 어떤 인터럽트 태스크에서도 사용할 수 없습니다.
자세한 내용은 10-2-6 고속 인터럽트 기능을 참조하십시오.

AI 번역본
-For Reference Only-

● 고속 인터럽트 기능이 유효할 때의 그 밖의 제한

- 인터럽트 태스크에서는 백그라운드 처리를 사용할 수 없습니다.
- 다음 보조 기억 영역 워드의 데이터는 유효하지 않습니다: A440(최대 인터럽트 태스크 처리 시간)과 A441(최대 처리 시간의 인터럽트 태스크).
- EtherNet/IP 태그 데이터 링크 사용으로 인한 사이클 타임 증가가 더욱 길어집니다.

● 동기 유닛의 장착 위치

동기 유닛은 CPU 랙에 장착해야 합니다. 증설 랙에는 장착할 수 없습니다.

10-8-6 적용 순서

동기 유닛 동작을 사용하는 순서는 다음과 같습니다.

1 초기 설정(10-8-7 PLC 설정을 참조하십시오.)

CX-Programmer에서 PLC 설정의 Timings/Synchronous 탭 페이지에서 동기 유닛 동작 설정을 수행합니다.

(1) 동기 유닛 동작을 유효화합니다. Use Synchronous Operation 체크 박스를 선택합니다.

(2) 동기 동작 사이클 타임을 설정합니다.

(3) 동기 데이터 리프레시로 데이터를 교환할 워드를 설정합니다.

(a) 동기 데이터 리프레시 영역 할당 영역에서 입력 데이터의 선두 어드레스를 설정합니다.

(b) 동기 유닛을 등록합니다.

(c) 각 동기 유닛의 입력 및 출력 데이터에 대한 선두 어드레스와 데이터 사이즈를 설정합니다.

2 동기 유닛의 I/O 데이터 할당

각 동기 유닛에 대해 I/O 데이터를 어떻게 사용할지 할당합니다. CX-Programmer에서 PLC 설정의 Timings/Synchronous 탭 페이지에 있는 동기 유닛 동작 설정 영역에서 Special Unit Setup 버튼을 클릭하여 각 동기 유닛에 대한 설정을 수행합니다.
구체적인 설정에 대해서는 각 동기 유닛의 매뉴얼을 참조하십시오.

3 필요한 경우 동기 인터럽트 태스크 작성. (10-8-8 동기 인터럽트 태스크 작성을 참조하십시오.)

CX-Programmer에서 동기 인터럽트 태스크의 프로그램을 인터럽트 태스크 2에 작성합니다.

4 동기 유닛 동작 디버깅(10-8-9 동기 유닛 동작의 조정과 문제 해결을 참조하십시오.)

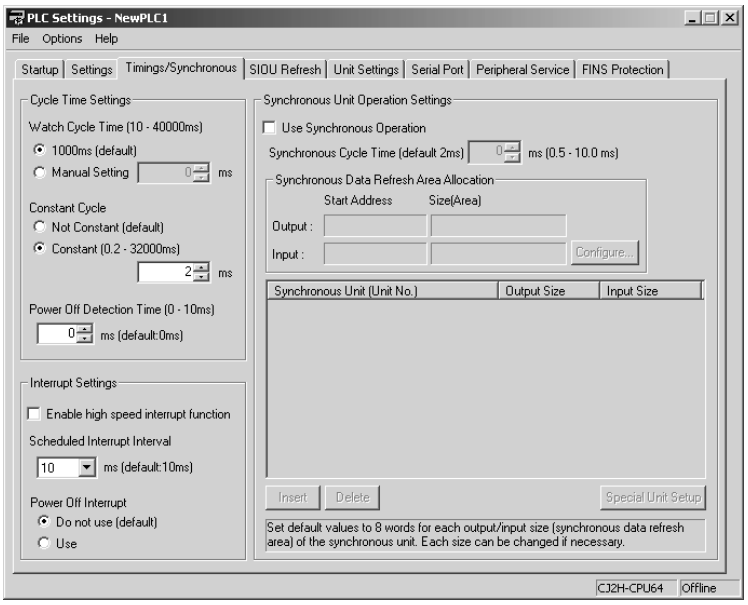
CX-Programmer의 Synchronous Operation Status 대화 상자에서 다음 값을 모니터링하면서 동기 동작 사이클 타임을 조정합니다.

- 동기 처리 시간의 현재값과 최댓값
- 동기 동작 사이클 타임의 설정값
- 일반 사이클 타임의 현재값과 최댓값
- 동기 유닛과의 데이터 교환 에러

AI 번역본
-For Reference Only-

10-8-7 PLC 설정

동기 유닛 동작 기능의 설정은 CX-Programmer의 Timings/Synchronous 탭 페이지에 있는 동기 유닛 동작 설정 영역에서 수행합니다.



● Use Synchronous Operation

이 체크 박스는 동기 유닛 동작 기능을 유효/무효로 설정하는 데 사용합니다.

파라미터	설정	디폴트	설명	관련 보조 기억 영역 비트 및 워드
Use Synchro-nous Operation	해제: 사용하지 않음. 선택: 사용함.	해제(사용하지 않음)	동기 유닛 동작을 사용하려면 체크 박스를 선택합니다. 동기 유닛 동작 기능의 사용이 유효화되면 고속 인터럽트 기능이 무조건 유효화됩니다.	---

● Synchronous Cycle Time

동기 동작 사이클 타임을 설정합니다.

파라미터	설정	디폴트	설명	관련 보조 기억 영역 비트 및 워드
Synchronous Cycle Time	0.5~10.0 ms (0.1 ms 단위)	2.0 ms	동기 동작 사이클 타임을 설정합니다.	A10102(동기 동작 사이클 타임)

사용상의 주의

동기 동작 사이클 타임은 다음 조건을 만족해야 합니다.

- 동기 처리 시간은 동기 동작 사이클 타임보다 짧아야 합니다.
- 동기 유닛의 내부 처리 시간은 동기 사이클 타임보다 짧아야 합니다.

일반적으로 동기 동작 사이클 타임은 동기 처리 시간의 1.5배 이상으로 설정하십시오.

● 동기 데이터 리프레시 영역 할당

1 동기 유닛의 출력 및 입력 데이터 설정

동기 데이터 리프레시의 데이터 사이즈를 설정합니다.

파라미터	설정	설정	디폴트	설명	관련 보조 기억 영역 비트 및 워드
출력	선두 어드레스	CIO 1200(고정)	CIO 1200	이것은 동기 출력 데이터 리프레시의 선두 워드 어드레스입니다.	---
	사이즈(할당 영역)	2~96 워드*	50 워드	이것은 동기 출력 데이터 리프레시의 사이즈입니다.	---
입력	선두 어드레스	CIO 1202~CIO 1294 또는 "미사용"	CIO 1250	이것은 동기 입력 데이터 리프레시의 선두 워드 어드레스입니다.	---
	사이즈(할당 영역)	2~94 워드 또는 설정 없음*	46 워드	이것은 동기 입력 데이터 리프레시의 사이즈입니다.	---

* 선두 어드레스를 설정하면 사이즈가 자동으로 계산됩니다.

2 사이즈(영역)

출력 데이터
동기 출력 데이터 리프레시의 데이터 사이즈를 설정합니다. 선두 어드레스는 자동으로 계산됩니다.
출력 사이즈는 동기 유닛마다 개별적으로 설정합니다.

파라미터	설정	디폴트	설명	관련 보조 기억 영역 비트 및 워드
출력 사이즈	CPU 유닛: 2 워드 이상(최대 출력 사이즈까지) 동기 유닛: 0 워드 이상(최대 출력 사이즈까지)	CPU 유닛: 2 워드 동기 유닛: 8 워드	동기 출력 데이터 리프레시의 데이터 사이즈를 설정합니다.	---

OMRON 번역본 -For Reference Only-

입력 데이터
동기 입력 데이터 리프레시의 데이터 사이즈를 설정합니다. 선두 어드레스는 자동으로 계산됩니다.
출력 사이즈는 동기 유닛마다 개별적으로 설정합니다.

파라미터	설정	디폴트	설명	관련 보조 기억 영역 비트 및 워드
입력 사이즈	동기 유닛: 0~16 워드	동기 유닛: 8 워드	동기 입력 데이터 리프레시의 데이터 사이즈를 설정합니다.	---

● 동기 유닛(유닛 No.)

동기 유닛 동작에 사용할 동기 유닛을 PLC에 등록합니다. 일반적으로 최대 10 유닛까지 등록할 수 있지만, 유닛에 따라서는 등록할 수 있는 유닛 수가 더 적을 수 있습니다.
자세한 내용은 각 유닛의 오퍼레이션 매뉴얼을 참조하십시오.

파라미터	설정	디폴트	설명	관련 보조 기억 영역 비트 및 워드
동기 유닛	동기 유닛 동작을 지원하는 CPU 유닛 및 기타 유닛의 명칭	CPU 유닛만	빈 행을 클릭하면 동기 유닛 목록이 표시됩니다. 목록에서 유닛을 선택합니다.	---

10-8-8 동기 인터럽트 태스크 작성

동기 인터럽트 태스크에서 실행할 프로그램을 작성하고, 프로그램 프로퍼티에서 태스크 종별을 인터럽트 태스크 2(정주기 인터럽트 0)로 설정합니다. 이 태스크에는 디지털 캠 제어 등의 동기 제어를 위한 프로그래밍이 포함됩니다. 예를 들어 디지털 캠 제어의 경우, 이 태스크에는 실축 또는 가상축을 기준으로 캠 곡선을 제어하는 근사 연산(APR) 명령어나 표준 펑션 블록이 포함됩니다.

● 관련 보조 기억 영역 비트 및 워드

명칭	비트/워드	설명	읽기/쓰기
동기 유닛 동작 서비스 플래그	A10100.00	이 플래그는 CPU 유닛이 동기 유닛 동작 기능을 위해 동기 유닛을 서비스할 수 있을 때 두 번째 동기 동작 사이클부터 ON 됩니다. PLC를 기동하거나 재기동한 후의 첫 번째 동기 동작 사이클을 포함하여 그 밖의 경우에는 OFF입니다. 동기 유닛 동작과 관련된 프로그램 실행을 제어하기 위해 사용자 프로그래밍에서 이 플래그를 사용하십시오.	읽기
동기 입력 데이터 리프레시 에러 코드	A10101	CPU 유닛이 동기 유닛으로부터 동기 입력 데이터를 수신하지 못하면 이 워드에 0001 hex가 저장됩니다. 동기 입력 데이터를 정상적으로 수신한 후에는 0000 hex가 저장됩니다. 이 워드는 동기 동작 사이클마다 갱신됩니다. 동기 입력 데이터를 사용하기 위한 조건으로 사용자 프로그래밍에서 이 워드를 사용하십시오.	읽기
동기 동작 사이클 타임	A10102	이 워드에는 설정한 사이클 타임이 유효해진 시점부터 PLC 설정에서 설정한 동기 동작 사이클 타임이 0.1 ms 단위로 저장됩니다. 펑션 블록 등의 사용자 프로그래밍에서 설정된 동기 동작 사이클 타임을 읽으려면 이 워드를 사용하십시오.	읽기

AI 번역본
-For Reference Only

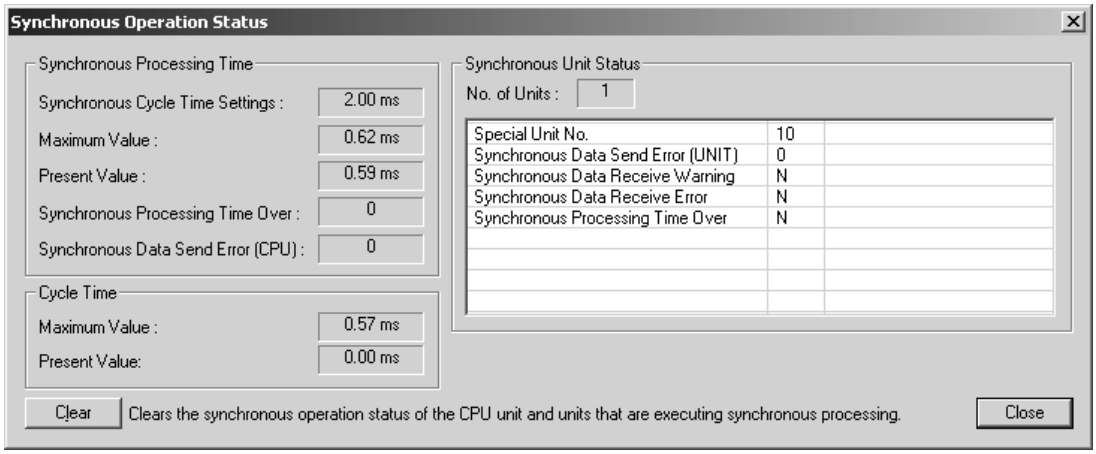
10-8-9 동기 유닛 동작의 조정과 문제 해결

동기 유닛 동작 상태 확인

동기 유닛 동작의 상태는 CX-Programmer의 Synchronous Operation Status 대화 상자 또는 동기 유닛의 SYNC 표시등에서 확인할 수 있습니다.

● CX-Programmer

동기 유닛 동작을 사용하는 PLC와 CX-Programmer를 온라인으로 접속하고, PLC Information 메뉴에서 PLC - Edit - Synchronous Operation Status를 선택합니다. 아래와 같은 Synchronous Operation Status 대화 상자가 표시됩니다.



항목		설명
동기 처리 시간	동기 사이클 타임 설정	동기 동작 사이클에 설정된 값입니다.
	최댓값과 현재값	현재까지의 최대 동기 동작 사이클 타임과 현재의 동기 동작 사이클 타임입니다.
	동기 처리 시간 오버	CPU 유닛의 동기 처리 시간이 설정된 동기 동작 사이클 타임을 초과한 횟수입니다. 이 값은 PLC가 기동될 때 클리어됩니다.
	동기 데이터 송신 에러(CPU)	CPU 유닛이 동기 유닛으로 동기 데이터를 송신하지 못한 횟수입니다. 이 값은 PLC가 기동될 때 클리어됩니다.
사이클 타임	최댓값과 현재값	일반 사이클 타임의 최댓값과 현재값입니다.

AI 번역본
-For Reference Only-

항목		설명
동기 유닛 상태	유닛 수	동기 유닛 동작으로 설정된 유닛의 수입니다.
	특수 유닛 No.	특수 I/O 유닛 또는 CPU 버스 유닛으로서의 동기 유닛의 유닛 번호입니다.
	동기 데이터 송신 에러(UNIT)	CPU 유닛이 동기 유닛이 송신한 데이터를 수신하지 못한 횟수입니다. 이 값은 PLC가 기동될 때 클리어됩니다.
	동기 데이터 수신 경고	동기 유닛이 CPU 유닛에서 송신한 동기 데이터를 한 번이라도 수신하지 못한 경우(CPU 유닛이 동기 출력 데이터를 송신하지 못한 경우 포함) 이 상태는 Yes가 됩니다. CPU 버스 유닛 영역 또는 특수 I/O 유닛 영역의 유닛 경고 리셋 비트를 ON 하면 이 상태가 클리어됩니다.
	동기 데이터 수신 에러	동기 유닛이 CPU 유닛에서 송신한 동기 데이터를 연속으로 2회 이상 수신하지 못한 경우 이 상태는 Yes가 됩니다. CPU 버스 유닛 영역 또는 특수 I/O 유닛 영역의 유닛 에러 리셋 비트를 ON 하면 이 상태가 클리어됩니다.
	동기 처리 시간 오버	동기 유닛의 내부 처리 시간이 동기 동작 사이클 타임을 1회 이상 초과한 경우 이 상태는 Yes가 됩니다. CPU 버스 유닛 영역 또는 특수 I/O 유닛 영역의 유닛 경고 리셋 비트를 ON 하면 이 상태가 클리어됩니다.
Clear 버튼		이 버튼은 CPU 유닛과 동기 유닛 양쪽의 동기 유닛 동작 상태 및 에러/경고 값을 클리어합니다.

동기 동작 사이클 타임 조정

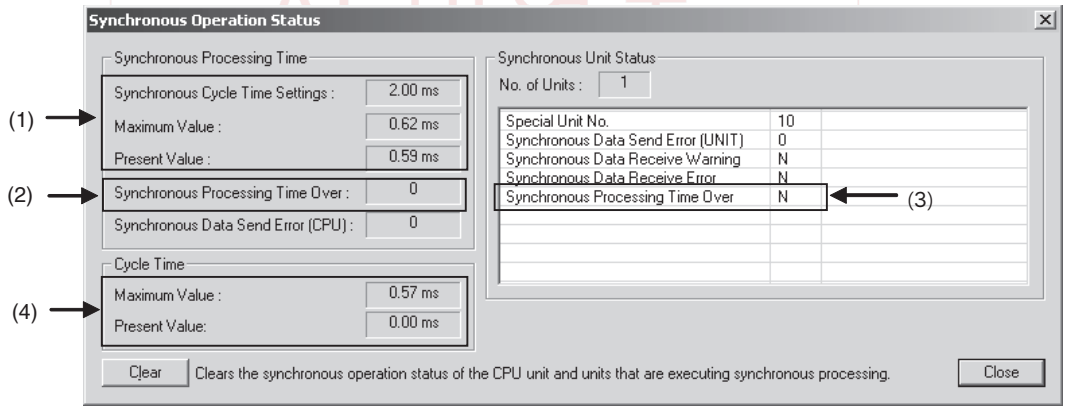
다음 순서로 설정된 동기 동작 사이클 타임이 적절한지 확인하고, 필요한 경우 조정할 수 있습니다.

(1) 동기 동작 사이클의 최댓값과 현재값이 설정값을 초과하는지 확인합니다.

(2) CPU 유닛에서 동기 처리 시간 오버 에러가 빈번하게 발생하는지 확인합니다.

(3) 동기 유닛에서 동기 처리 시간 오버 경고가 빈번하게 발생하는지 확인합니다.

(4) 일반 사이클 타임의 최댓값과 현재값을 확인하여, 동기 유닛 동작으로 인한 증가로 일반 사이클 타임이 허용 범위를 초과했는지 확인합니다.



● 동기 유닛 전면의 SYNC 표시등 확인

동기 유닛이 동기 유닛 동작 모드일 때 동기 유닛 전면의 SYNC 표시등이 녹색으로 점등합니다.
CJ1W-NC□82에서는 전면 패널의 7세그먼트 표시의 첫 번째 자릿수의 점이 점등합니다. 동기 출력 데이터 수신 에러 등의 에러가 발생하면 이 표시등은 소등합니다.



참고

다음 비트가 특수 I/O 유닛 영역 또는 CPU 버스 유닛 영역에서 동기 유닛에 할당됩니다.
이 비트는 CPU 유닛의 사용자 프로그래밍에서 동기 유닛의 상태를 확인하고 제어하는 데 사용할 수 있습니다.

I/O 구분	사이즈	명칭	기능
특수 I/O 유닛 영역 또는 CPU 버스 유닛 영역에 할당되는 입력 비트 (CPU 유닛으로의 입력)	1 비트	동기 유닛 동작 모드 플래그	이 플래그는 동기 유닛이 동기 유닛 동작 모드로 동작하는 동안 ON 됩니다. (이 플래그는 동기 유닛의 SYNC 표시등과 동일한 상태를 나타냅니다.)
	1 비트	동기 데이터 수신 경고 플래그	동기 유닛이 CPU 유닛에서 송신한 동기 데이터를 한 번이라도 수신하지 못하면 이 플래그가 ON 됩니다.
	1 비트	동기 데이터 수신 에러 플래그	동기 유닛이 CPU 유닛에서 송신한 동기 데이터를 연속으로 2회 이상 수신하지 못하면 이 플래그가 ON 됩니다.
	1 비트	동기 처리 시간 오버 플래그	동기 유닛의 내부 처리 시간이 동기 동작 사이클 타임을 1회 이상 초과하면 이 플래그가 ON 됩니다.
특수 I/O 유닛 영역 또는 CPU 버스 유닛 영역에 할당되는 출력 비트 (CPU 유닛에서의 출력)	1 비트	유닛 에러 리셋 비트	동기 유닛의 동기 데이터 수신 에러 등의 에러를 리셋하려면 이 비트를 ON 하십시오.
	1 비트	유닛 경고 리셋 비트	동기 유닛의 동기 데이터 수신 경고 및 동기 처리 시간 초과 경고 등의 경고를 리셋하려면 이 비트를 ON 하십시오.

동기 유닛 동작의 정지

동기 신호 감시 에러

정상 운전 중에 CPU 유닛에서 동기 유닛으로의 동기 신호가 100 ms를 넘도록(또는 CJ1W-NC□82의 유닛 제어 사이클의 4배를 넘도록) 발생하지 않거나 기동 시 11초 동안 발생하지 않으면, 모든 동기 유닛과 모든 동기 유닛 동작이 정지합니다.

I/O 버스 에러 또는 동기 유닛 에러

동기 유닛 중 하나라도 I/O 버스 에러나 다음 유닛 에러가 발생하면, 모든 동기 유닛과 모든 동기 유닛 동작이 정지합니다.

- I/O 설정 에러
- 유닛 번호 중복 에러
- 특수 I/O 유닛 설정 에러
- CPU 버스 유닛 설정 에러
- 특수 I/O 유닛 에러
- CPU 버스 유닛 에러

CX-Programmer에 에러를 나타내는 대화 상자가 표시됩니다.

AI 번역본
For Reference Only

참고

동기 유닛 동작 중에 동기 유닛 중 하나라도 재기동되면 동기 동작 사이클이 정지하고 모든 동기 유닛이 재기동됩니다. 동기 유닛 동작은 모든 동기 유닛이 정상적으로 재기동된 후에만 재개됩니다.

조정과 문제 해결

동기 동작 사이클 타임 변경과 같은 조정은 다음 표에 기재된 현상에 대해 필요합니다.

CX-Programmer의 동기 동작 상태	현상	원인	대책
동기 처리 시간 영역	동기 처리 시간 오버 에러가 빈번하게 발생합니다.	동기 동작 처리(동기 데이터 리프레시와 동기 인터럽트 태스크 실행)가 동기 동작 사이클 내에 완료되지 못했습니다.	동기 동작 사이클 타임을 동기 동작 처리 시간(동기 데이터 리프레시 시간과 동기 인터럽트 태스크 실행 시간의 합)보다 길게 설정하십시오.
	동기 데이터 송신 에러(CPU)가 발생합니다.	CPU 유닛에서 하드웨어 에러가 발생했습니다.	CPU 유닛을 교체하십시오.
동기 유닛 상태 영역	동기 데이터 송신 에러(UNIT)가 빈번하게 발생합니다.	동기 유닛에서 CPU 유닛으로 송신하는 동기 입력 데이터를 동기 입력 데이터 리프레시 기간 전에 준비하지 못했습니다.	동기 동작 사이클 타임을 동기 유닛의 내부 처리 시간보다 길게 설정하십시오.
	동기 처리 시간 오버 경고가 발생합니다	동기 유닛의 내부 처리가 동기 동작 사이클이 끝나기 전에 완료되지 못했습니다.	동기 동작 사이클 타임을 동기 유닛의 내부 처리 시간보다 길게 설정하십시오.

AI 번역본
-For Reference Only-

11

프로그래밍 디바이스와 통신

이 섹션에서는 CX-Programmer에서 PLC에 액세스하는 방법을 설명합니다. 또한 시리얼 통신과 네트워크를 넘나드는 통신에 대해서도 설명합니다.

11-1 CX-Programmer에서 PLC에 액세스 11-2
11-1-1 개요 11-2
11-1-2 액세스 가능한 PLC의 시스템 구성 11-4
11-1-3 CX-Programmer에서 PLC에 액세스 11-9
11-1-4 자동 온라인 접속 11-14
11-2 시리얼 통신 11-18
11-2-1 시리얼 통신 개요 11-18
11-3 통신 네트워크 11-32

AI 번역본
-For Reference Only-

11-1 CX-Programmer에서 PLC에 액세스

11-1-1 개요

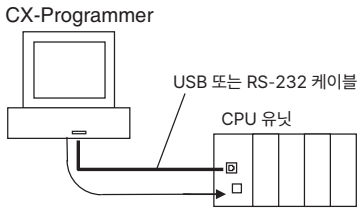
이 섹션에서는 CX-Programmer에서 PLC에 액세스하는 방법의 개요를 설명합니다.

PLC에 접속

CX-Programmer에서 PLC에 액세스하려면 다음 두 가지 방법 중 하나를 사용할 수 있습니다.

- **PLC에 직접 접속**

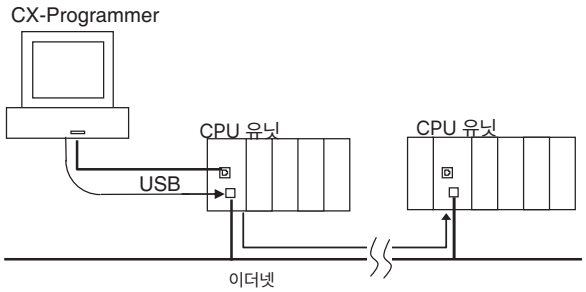
  USB 포트 또는 시리얼 포트로 CX-Programmer에 직접 접속된 PLC에 온라인으로 접속할 수 있습니다.



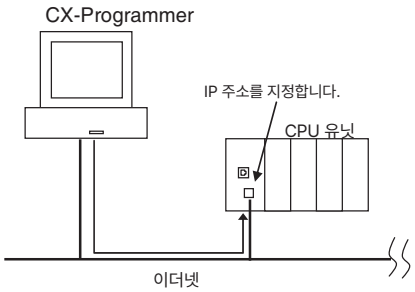
- **네트워크상의 PLC에 액세스**

  네트워크상의 PLC에 액세스하려면 다음 두 가지 방법 중 하나를 사용할 수 있습니다.

CX-Programmer에 시리얼 포트로 직접 접속된 PLC를 경유하여 네트워크상의 PLC에 온라인으로 접속합니다.



네트워크에 직접 접속된 퍼스널 컴퓨터에서 네트워크상의 PLC에 온라인으로 접속합니다.



-For Reference Only-

PLC에 온라인으로 접속하는 순서

CX-Programmer에서 PLC에 온라인으로 접속하려면 다음 두 가지 순서 중 하나를 사용할 수 있습니다.

● Change PLC 대화 상자

CX-Programmer 프로젝트에서 대상 PLC에 액세스하는 방법을 설정합니다. 이 설정은 프로젝트를 작성할 때 수행하지만, 프로젝트를 작성한 후에도 변경할 수 있습니다.

● 자동 온라인 접속

자동 온라인 접속은 접속이 확립된 PLC에 액세스할 때 사용합니다. CX-Programmer 프로젝트를 작성하지 않고도 사용할 수 있습니다.

다이렉트 접속

다이렉트 접속은 USB 포트 또는 시리얼 포트로 CX-Programmer에 직접 접속된 PLC에 온라인으로 접속할 때 사용합니다.

EtherNet/IP 노드 접속

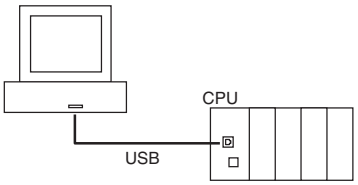
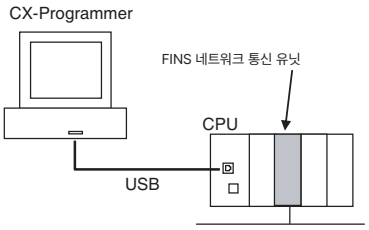
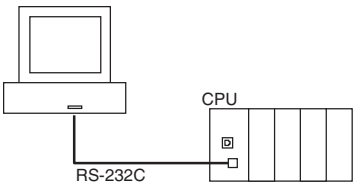
EtherNet/IP 노드 접속은 EtherNet/IP를 통해 접속이 확립된 PLC에 액세스할 때 사용합니다. 대상 PLC의 IP 주소를 모르더라도 검색하여 해당 PLC에 액세스할 수 있습니다.

AI 번역본
-For Reference Only-

11-1-2 액세스 가능한 PLC의 시스템 구성

PLC와의 USB 또는 시리얼 다이렉트 접속

다음 표와 같이 네트워크 종별을 선택합니다.

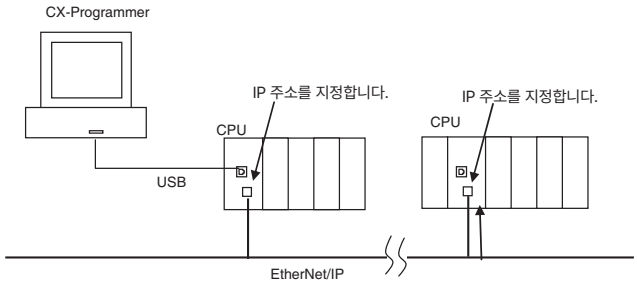
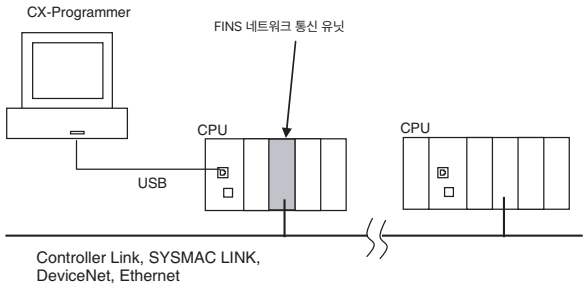
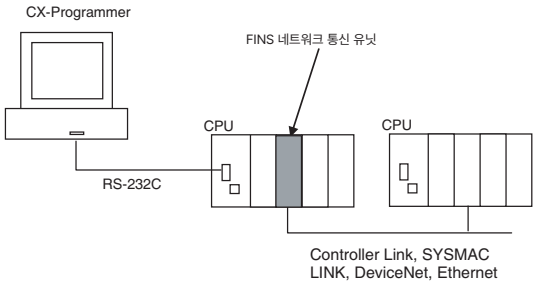
접속 케이블	시스템 구성	Change PLC 대화 상자 (네트워크 종별)	자동 온라인 접속	
			다이렉트 접속	EtherNet/IP 접속
USB 케이블	CX-Programmer  	USB	액세스 가능	액세스 불가
	FINS 네트워크에 액세스 CX-Programmer  * 로컬 네트워크 테이블을 설정해야 합니다.	Toolbus(USB 포트)	Toolbus (USB 포트)	액세스 불가
RS-232C 케이블	CX-Programmer  	Toolbus	SYSMAC WAY, 액세스 가능	액세스 불가

AI 번역본
-For Reference Only-

네트워크상의 PLC에 액세스

● 시리얼 포트로 접속된 PLC를 경유하여 액세스

다음 표와 같이 네트워크 종별을 선택합니다.

접속 케이블	시스템 구성	Change PLC 대화 상자 (네트워크 종별)	자동 온라인 접속	
			다이렉트 접속	Ether-Net/IP 접속
USB 케이블	USB를 경유하여 EtherNet/IP 네트워크에 액세스 (*1) 	USB	액세스 불가	액세스 가능
	USB를 경유하여 FINS 네트워크에 액세스 (*2)  * 로컬 네트워크 테이블을 설정해야 합니다.	Toolbus(USB 포트)	액세스 불가	액세스 불가
RS-232C 케이블	RS-232C를 경유하여 FINS 네트워크에 액세스 (*2)  * 로컬 네트워크 테이블을 설정해야 합니다.	SYSMAC WAY, Toolbus	액세스 불가	액세스 불가

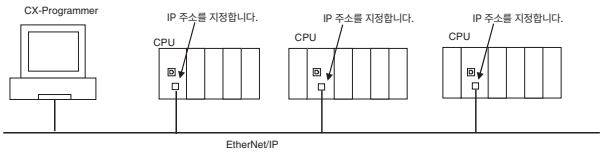
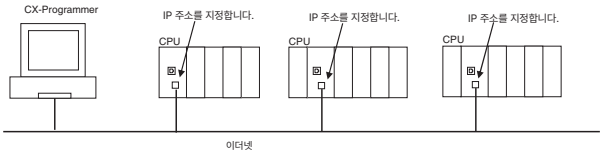
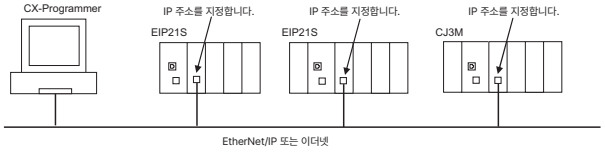
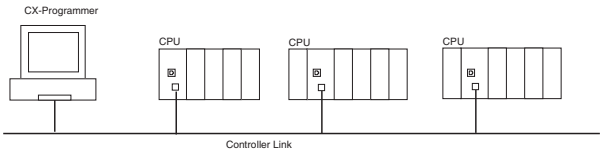
*1 PLC 기종 설정을 변경하는 대화 상자에서 네트워크 종별을 설정한 경우, EtherNet/IP로의 접속은 로컬 네트워크에 대해서만 가능합니다. 네트워크 계층을 넘는 접속은 할 수 없습니다. 자동 온라인 접속으로 EtherNet/IP 접속을 수행하면 네트워크 계층을 넘는 접속이 가능합니다.

*2 FINS 네트워크 간에는 계층을 넘는 접속을 사용할 수 있습니다.

견본
For Reference Only

● 네트워크를 경유하여 접속

다음 표와 같이 네트워크 종별을 선택합니다.

접속 케이블	시스템 구성	Change PLC 대화 상자 (네트워크 종별)	자동 온라인 접속	
			다이렉트 접속	Ether-Net/IP 접속
EtherNet/IP	EtherNet/IP 네트워크에 액세스 (*1)  접속 대상의 IP 주소를 설정해야 합니다.	EtherNet/IP	액세스 불가	액세스 가능
Ethernet	Ethernet을 FINS 네트워크로서 액세스 (*2)  접속처의 FINS 네트워크 주소, 노드 주소, IP 주소를 설정해야 합니다.	Ethernet, Ethernet (FINS/TCP)	액세스 불가	액세스 불가
EtherNet/IP 또는 이더넷	네트워크 보안 위험 저감 (*3)  접속 대상의 IP 주소를 설정해야 합니다.	Secure Comm	액세스 불가	액세스 불가
Controller Link		Controller Link	액세스 불가	액세스 불가

*1 PLC 기종 설정을 변경하는 대화 상자에서 네트워크 종별을 설정한 경우, EtherNet/IP로의 접속은 로컬 네트워크에 대해서만 가능합니다. 네트워크 계층을 넘는 접속은 할 수 없습니다. 자동 온라인 접속으로 EtherNet/IP 접속을 수행하면 네트워크 계층을 넘는 접속이 가능합니다.

*2 FINS 네트워크 간에는 계층을 넘는 접속을 사용할 수 있습니다.

*3 이것은 CJ1W-EIP21S, CJ3M-CPU3□ EtherNet/IP 유닛을 경유하여 접속하는 경우에만 해당됩니다.

-For Reference Only-

사용상의 주의

CJ2H-CPU6□-EIP 또는 CJ3M-CPU3□의 내장 EtherNet/IP 포트를 경유하여 지원 소프트웨어로 접속하는 경우, 다음 에러 중 하나가 발생하면 접속할 수 없습니다.

I/O 버스 에러(CJ2M-CPU3□에서는 접속 가능합니다.)

I/O 점수 초과 에러

유닛 번호 중복 에러

랙 번호 중복 에러

I/O 설정 에러

위 에러 중 하나가 발생하여 지원 소프트웨어로 접속할 수 없는 경우에는, CPU 유닛의 USB 포트를 경유하여 지원 소프트웨어로 접속하십시오.

사용상의 주의

EtherNet/IP 접속 사용 전의 주의사항

Windows XP 이후 운영체제의 방화벽 보안이 강화되어 이더넷 포트의 데이터 통신에 대한 제한이 늘어났습니다. 컴퓨터의 이더넷 포트에서 다음 PLC 중 하나에 EtherNet/IP 접속*1을 사용하는 경우, CX-Programmer 통신을 사용할 수 있도록 Windows 방화벽 설정을 변경해야 합니다.

*1: EtherNet/IP 접속에는 다음 경우가 포함됩니다:

・네트워크 종별을 EtherNet/IP로 설정한 온라인 접속

・PLC 메뉴에서 Auto Online - EtherNet/IP Node Online을 선택했을 때 EtherNet/IP 네트워크상의 PLC에 대한 자동 온라인 접속.

조작 순서는 A-6 Windows 7 이후 Windows OS에서의 EtherNet/IP 접속을 참조하십시오.

사용상의 주의

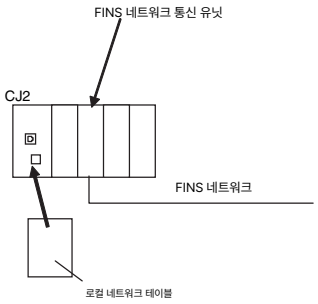
Windows 10보다 이전 운영체제가 설치된 컴퓨터에 CX-Programmer를 설치한 경우, Network Type에서 Secure Comm을 선택할 수 없습니다. 또한 1803보다 이전 버전의 Windows 10이 설치된 컴퓨터에 CX-Programmer를 설치한 경우, 시큐어 통신(Secure Comm)으로 온라인 접속을 할 수 없습니다.

AI 번역본
-For Reference Only-

참고

다음의 경우 라우팅 테이블을 작성하여 전송해야 합니다.

CPU 유닛에 FINS 네트워크 통신 유닛*이 하나 이상 접속된 경우, 지원 소프트웨어에서 내장 EtherNet/IP 네트워크(CJ2H-CPU6□-EIP, CJ2M-CPU3□ 또는 CJ3M-CPU3□)와 FINS 네트워크 양쪽에 액세스할 수 있도록 CX-Integrator로 라우팅 테이블의 로컬 네트워크 테이블을 작성하여 전송해야 합니다.



*「FINS 네트워크 통신 유닛」이란 Controller Link 유닛, SYSMAC LINK 유닛, 이더넷 유닛, DeviceNet 유닛 또는 FL-net 유닛을 가리킵니다. CompoNet 유닛은 포함되지 않습니다. 또한 시리얼 통신용 시리얼 게이트웨이를 라우팅 테이블과 함께 사용하는 경우도 가리킵니다.

참고

CJ2H-CPU6□-EIP, CJ2M-CPU3□ 또는 CJ3M-CPU3□에 FINS 네트워크 통신 유닛이 하나 이상 접속된 경우, 로컬 네트워크 테이블을 작성하여 전송하십시오. 로컬 네트워크 테이블을 작성하여 전송하지 않고 네트워크 주소를 0(즉, 디폴트)으로 설정한 채 지원 소프트웨어에서 액세스를 시도하면, 다음의 경우 내장 EtherNet/IP 포트의 노드 주소에 액세스하게 됩니다.

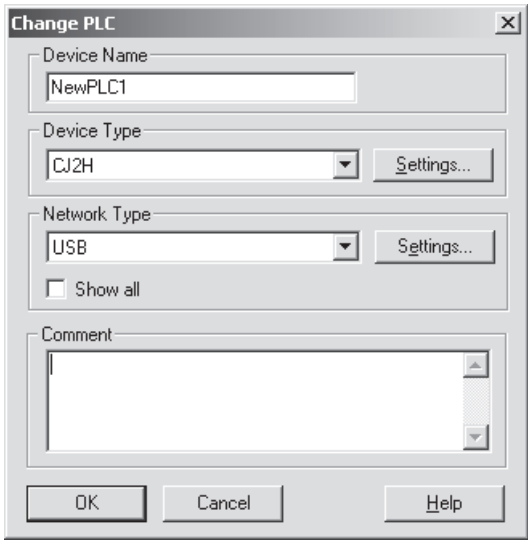
내장 EtherNet/IP 포트의 유닛 번호(디폴트: 0)가 FINS 통신 유닛의 유닛 번호보다 작은 경우.

AI 번역본
-For Reference Only-

11-1-3 CX-Programmer에서 PLC에 액세스

Change PLC 대화 상자에서의 순서

새 프로젝트를 작성할 때는 다음 순서로 Change PLC 대화 상자에서 PLC에 접속하는 방법을 선택합니다. 이 예는 CJ2H CPU 유닛의 경우입니다.



1 Device Type 필드에서 CJ2H를 선택합니다.

2 Network Type 필드에서 PLC에 접속하는 방법을 선택합니다. 디폴트로는 다음 방법 중 하나를 선택할 수 있습니다.

- USB
- EtherNet/IP
- Secure Comm

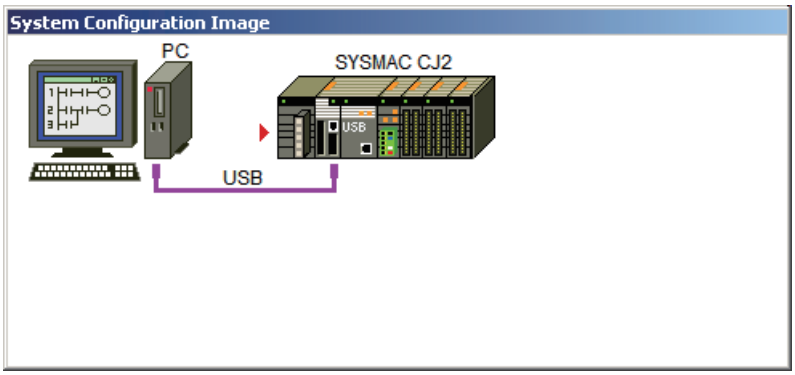
필요한 경우 Show All 버튼을 클릭하면 다른 네트워크 종별을 표시할 수 있습니다. 다음 표에 나타낸 두 가지 조건에 따라 네트워크 종별을 선택하십시오.

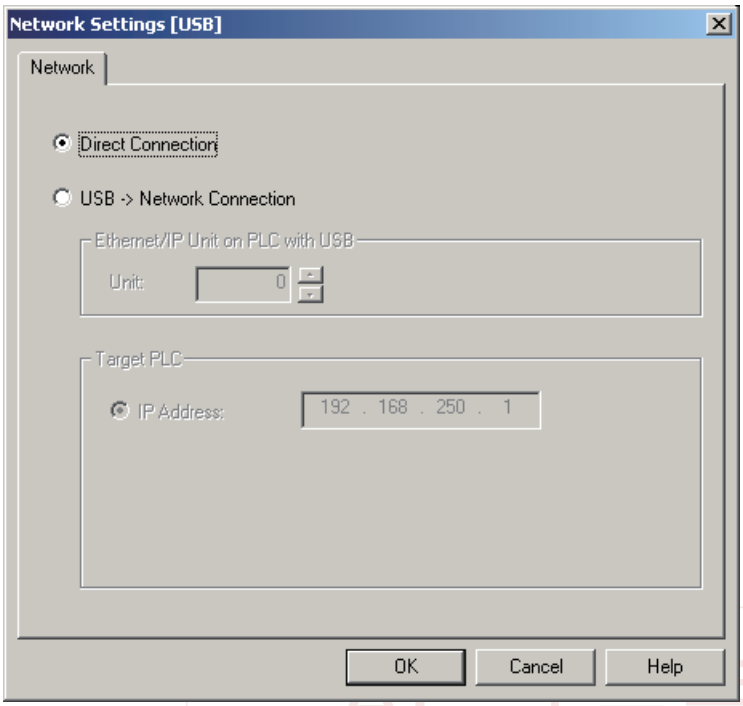
조건 1: PLC에 접속하는 케이블의 종류	조건 2	네트워크 종별	
USB	---	USB	디폴트로 표시됩니다.
Ethernet	EtherNet/IP 네트워크에 액세스	EtherNet/IP	
	네트워크 보안 위험 저감	Secure Comm	
USB	USB를 경유하여 FINS 네트워크에 액세스	Toolbus(USB 포트)	Show All 버튼을 클릭하면 **표시됩니다.**
Ethernet	이더넷 네트워크를 FINS 네트워크로서 액세스	Ethernet	
		Ethernet (FINS/TCP)	
RS-232C	---	Toolbus	
		SYSMAC WAY	
Controller Link(Controller Link 보드로 접속.)	---	Controller Link	
Fins Gateway	---	Fins Gateway	

AI 번역본
-For Reference Only-

3 Settings 버튼을 클릭합니다. 선택한 네트워크 종별에 대한 Network Settings 대화 상자가 표시됩니다.
이 예에서는 디폴트로 표시되는 네트워크 종별, 즉 USB, EtherNet/IP,
Secure Comm에 대해 설명합니다. 다른 네트워크 종별의 설정에 대해서는 CX-Programmer
오퍼레이션 매뉴얼 (Cat. No. W446)을 참조하십시오.

● 네트워크 종별: USB





Direct Connection:

USB로 직접 접속된 PLC에 액세스하려면 이 옵션을 선택합니다.

USB -> Network Connection:

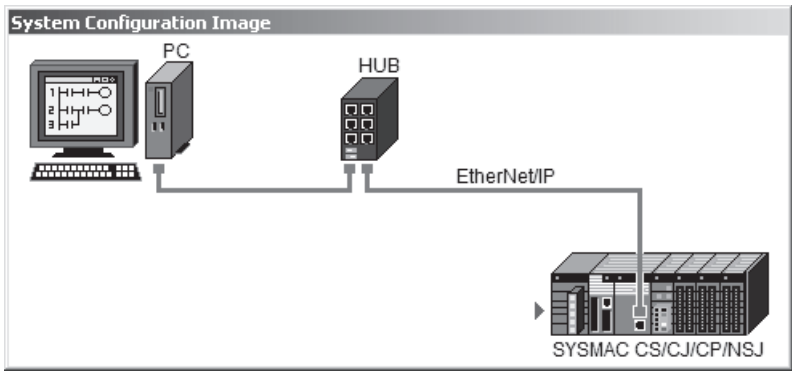
USB 포트를 경유하여 EtherNet/IP 네트워크(단일 네트워크)상의 PLC에 액세스하려면 이 옵션을 선택합니다.
다음 설정을 수행하십시오:

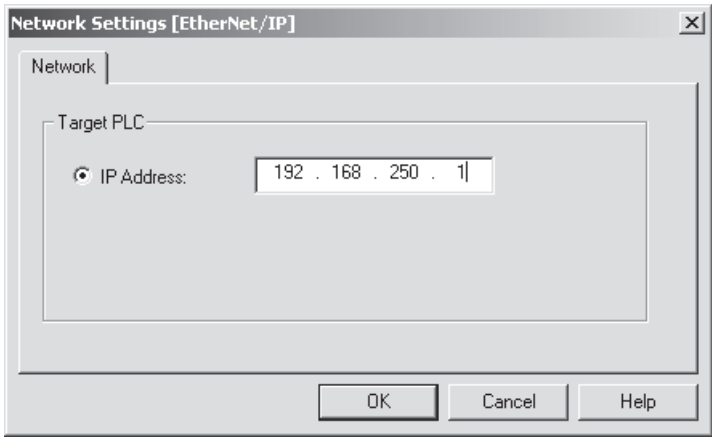
EtherNet/IP Unit on PLC connected with USB:
CPU 유닛의 내장 EtherNet/IP 포트에 접속하는 경우에는 유닛을 0으로 설정합니다.

IP Address of Target PLC:
CPU 유닛의 내장 EtherNet/IP 포트 또는 EtherNet/IP 유닛의 IP 주소를 설정합니다.
CPU 유닛의 내장 EtherNet/IP 포트의 디폴트 IP 주소는 다음과 같습니다:
192.168.250.node_address.

-For Reference Only-

● 네트워크 종별: EtherNet/IP





IP Address of *Target PLC*:

CPU 유닛의 내장 EtherNet/IP 포트 또는 EtherNet/IP 유닛의 IP 주소를 설정합니다.
CPU 유닛의 내장 EtherNet/IP 포트의 디폴트 IP 주소는 다음과 같습니다:
192.168.250.node_address.

● 퍼스널 컴퓨터의 IP 주소 설정

EtherNet/IP를 경유하여 PLC에 접속하는 경우, 퍼스널 컴퓨터의 IP 주소를 CPU 유닛의 내장 EtherNet/IP 포트 또는 EtherNet/IP 유닛의 IP 주소에 맞게 설정해야 합니다. 예를 들어 CPU 유닛의 내장 EtherNet/IP 포트의 IP 주소가 디폴트로 설정되어 있는 경우, 퍼스널 컴퓨터에 다음 IP 주소를 설정합니다.

• IP 주소

예: 192.168.250.2

- 사용자 설정값(1~254)
- 내장 EtherNet/IP 포트의 IP 주소와 동일한 세그먼트의 값

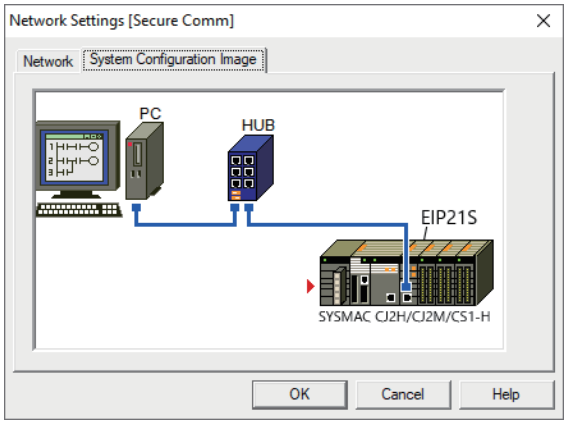
• 서브넷 마스크

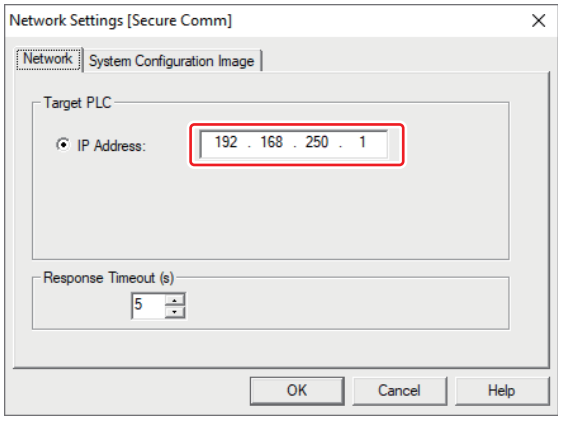
예: 255.255.255.0

- 사용자 설정값(0~252)
- 클래스 C 마스크

-For Reference Only-

● 네트워크 종별: Secure Comm





IP Address of *Target PLC*:

CJ3M-CPU3□의 내장 EtherNet/IP 포트 또는 CJ1W-EIP21S EtherNet/IP 유닛의 IP 주소를 설정합니다. IP 주소에 대한 자세한 내용은 CS/CJ 시리즈 EtherNet/IP 유닛 오퍼레이션 매뉴얼 (Cat. No. W465)을 참조하십시오.

● 퍼스널 컴퓨터의 IP 주소 설정

EtherNet/IP를 경유하여 PLC에 접속하는 경우, 퍼스널 컴퓨터의 IP 주소를 CJ3M-CPU3□ 또는 CJ1W-EIP21S의 IP 주소에 맞게 설정해야 합니다.
예를 들어 IP 주소가 디폴트로 설정되어 있는 경우, 퍼스널 컴퓨터에 다음 IP 주소를 설정합니다.

• IP 주소

예: 192.168.250.2

- 사용자 설정값(1~254)
- 내장 EtherNet/IP 포트의 IP 주소와 동일한 세그먼트의 값

• 서브넷 마스크

예: 255.255.255.0

- 사용자 설정값(0~252)
- 클래스 C 마스크

사용상의 주의

- Windows 10보다 이전 운영체제가 설치된 컴퓨터에 CX-Programmer를 설치한 경우, Network Type에서 Secure Comm을 선택할 수 없습니다. 또한 1803보다 이전 버전의 Windows 10이 설치된 컴퓨터에 CX-Programmer를 설치한 경우, 시큐어 통신(Secure Comm)으로 온라인 접속을 할 수 없습니다.
- Secure Comm으로 온라인 접속할 때 사용자 이름과 패스워드가 일치하지 않으면 인증이 실패하여 CPU 유닛에 온라인으로 접속할 수 없습니다. 사용자 이름과 패스워드를 기록해 두십시오. 잊어버린 경우에는 문제 해결 순서에 설명된 대로 인증 설정을 클리어하십시오. 이때 설정이 디폴트로 리셋되는 점에 주의하십시오.

참고

Secure Comm을 선택한 경우, 위에 설명한 설정 외에 CS1W/CJ1W-EIP21S, CJ3M-CPU3□ EtherNet/IP 유닛에 사용자 인증 설정을 저장해야 합니다.

사용자 인증 설정은 EIP21S User Management Tool로 수행합니다. 사용자 인증 설정에 대한 자세한 내용은 CS/CJ 시리즈 EtherNet/IP 유닛 오퍼레이션 매뉴얼 (Cat. No. W465)의 User Authentication을 참조하십시오.

AI 번역본
-For Reference Only-

11-1-4 자동 온라인 접속

대상 PLC의 프로젝트가 퍼스널 컴퓨터에 없는 경우에도, 온라인으로 접속하여 PLC에서 프로그램을 업로드할 수 있습니다.

시리얼 다이렉트 접속

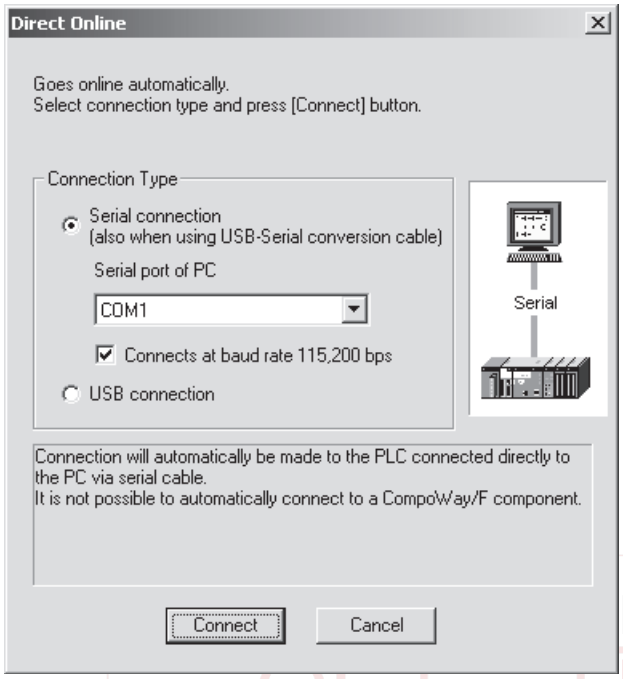
자동 온라인 접속을 실행하면 사용 가능한 퍼스널 컴퓨터의 시리얼 포트를 자동으로 검색합니다. 해당하는 시리얼 포트가 발견되면 CX-Programmer가 그 시리얼 포트에 접속된 PLC에 자동으로 접속됩니다.

1 PLC 메뉴에서 Auto Online – Direct Online을 선택합니다. 아래와 같이 Direct Online 대화 상자가 표시됩니다.

2 퍼스널 컴퓨터를 PLC에 접속하는 다음 방법 중 하나를 선택합니다.

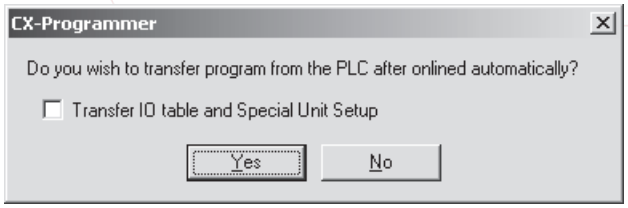
・시리얼 접속(퍼스널 컴퓨터의 USB 접속을 PLC의 RS-232C 접속으로 변환하는 경우 포함)

・USB 접속



퍼스널 컴퓨터의 USB 포트에서 CPU 유닛의 페리페럴 USB 포트에 접속한 경우, USB 접속을 선택하고 Connect 버튼을 클릭합니다. 다음 대화 상자가 표시됩니다.

-For Reference Only-



자동으로 온라인 접속한 후 프로그램, PLC 설정, I/O 테이블을 퍼스널 컴퓨터로 업로드하려면 Yes 버튼을 클릭합니다. CX-Programmer가 PLC에 자동으로 접속되고, 이들 항목이 PLC에서 업로드됩니다.

EtherNet/IP 노드 접속

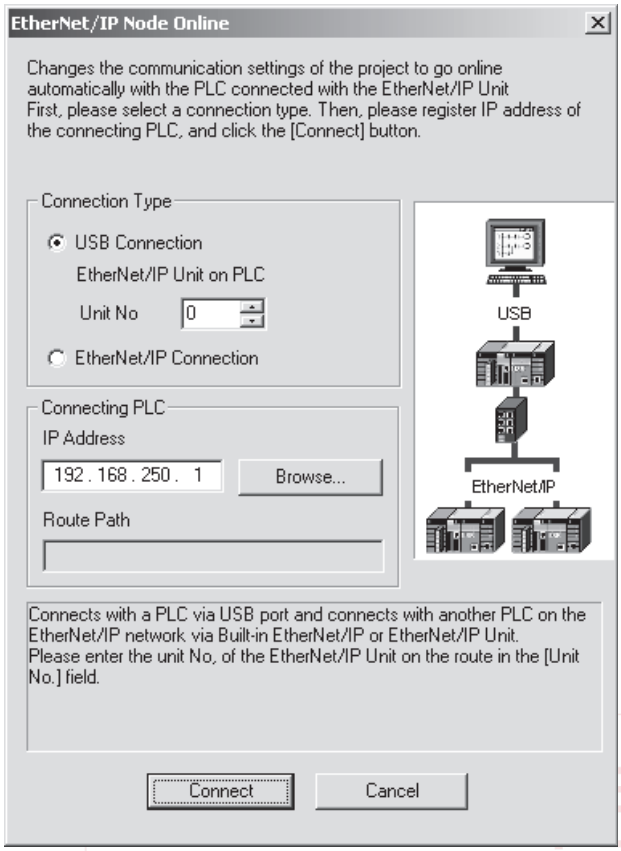
EtherNet/IP 네트워크에 접속된 PLC에는 EtherNet/IP 네트워크를 경유하여 접속할 수 있습니다. 대상 PLC의 IP 주소를 모르는 경우에도 검색하여 해당 PLC에 액세스할 수 있습니다.

사용상의 주의

자동 온라인 접속에서는 CJ3M-CPU3□, CJ1W-EIP21S를 Secure Comm으로 접속할 수 없습니다.

1 **PLC 메뉴에서 Auto online – EtherNet/IP Node Online을 선택합니다.**

2 EtherNet/IP Node Online 대화 상자가 표시됩니다.



3 접속 종별에서 대상 PLC에 접속하는 방법을 선택합니다.

・USB 접속
USB 포트를 통해 PLC에 물리적으로 직접 접속한 다음, 내장 EtherNet/IP 포트 또는 EtherNet/IP 유닛을 경유하여 EtherNet/IP 네트워크(단일 네트워크)상의 PLC에 온라인으로 접속하려면 USB 접속을 선택합니다.

・EtherNet/IP 접속
퍼스널 컴퓨터를 EtherNet/IP 네트워크에 물리적으로 접속하고 EtherNet/IP를 경유하여 네트워크(단일 네트워크)상의 PLC에 온라인으로 접속하려면 EtherNet/IP 접속을 선택합니다.

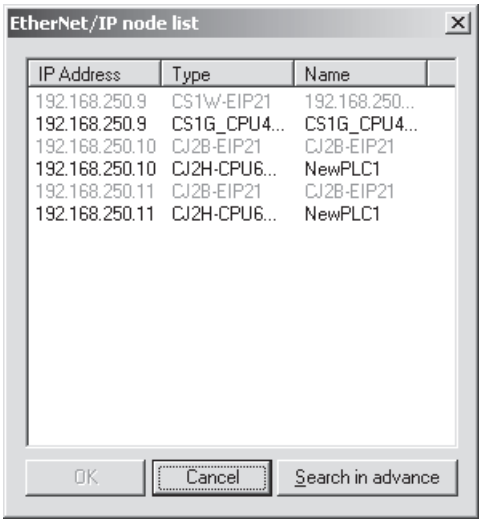
-For Reference Only-

4 접속할 PLC에 대해 대상 PLC의 IP 주소를 설정하는 방법을 선택합니다.

- IP Address
  대상 PLC의 IP 주소를 직접 입력합니다.
- Browse
  네트워크에 온라인으로 접속된 PLC를 검색하여 대상 PLC를 선택합니다. 대상 PLC의 IP 주소를 모르는 경우에 이 방법을 사용합니다.

접속된 노드는 목록으로 표시하거나 네트워크 계층별로 표시할 수 있습니다.

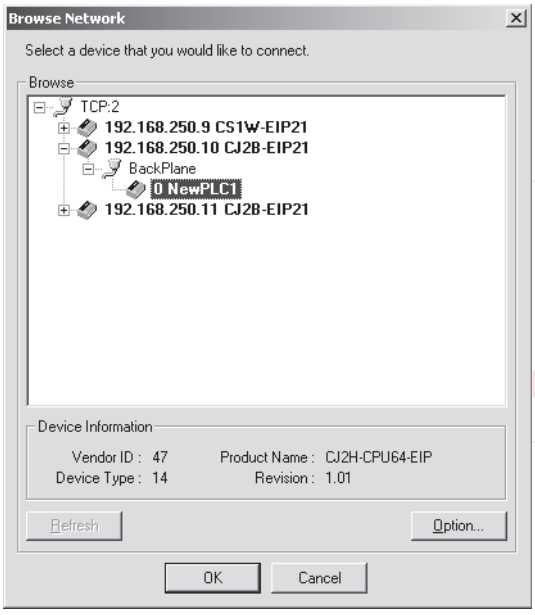
노드 목록(Browse 시 표시)



- IP 주소, 디바이스 종별, 이름이 표시됩니다. IP 주소를 선택하고 OK 버튼을 클릭합니다.

Search in Advance

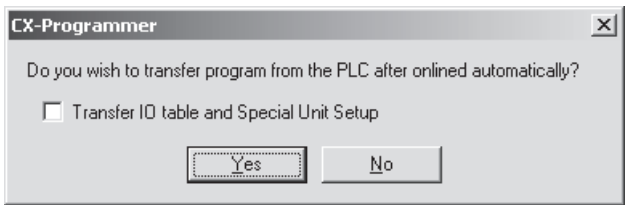
EtherNet/IP node list 대화 상자에서 Search in advance 버튼을 클릭하면 다음 대화 상자가 표시됩니다.



- 플러스(+) 아이콘을 클릭하면 해당 항목 아래의 모든 계층이 표시됩니다. 접속할 PLC를 선택하고 OK 버튼을 클릭합니다.

본
e Only-

5 Connect 버튼을 클릭하여 온라인으로 접속합니다. 다음 대화 상자가 표시됩니다.



자동으로 온라인 접속한 후 프로그램, PLC 설정, I/O 테이블을 퍼스널 컴퓨터로 업로드하려면 Yes 버튼을 클릭합니다. 컴퓨터가 PLC에 자동으로 접속되고, 이들 항목이 PLC에서 업로드됩니다.

AI 번역본
-For Reference Only-

11-2 시리얼 통신

11-2-1 시리얼 통신 개요

시리얼 통신 포트 모드(프로토콜)는 CPU 유닛의 PLC 설정에서 전환할 수 있습니다.
선택한 프로토콜에 따라 다음 시스템을 구성할 수 있습니다.

프로토콜

시리얼 통신은 다음 프로토콜을 지원합니다.

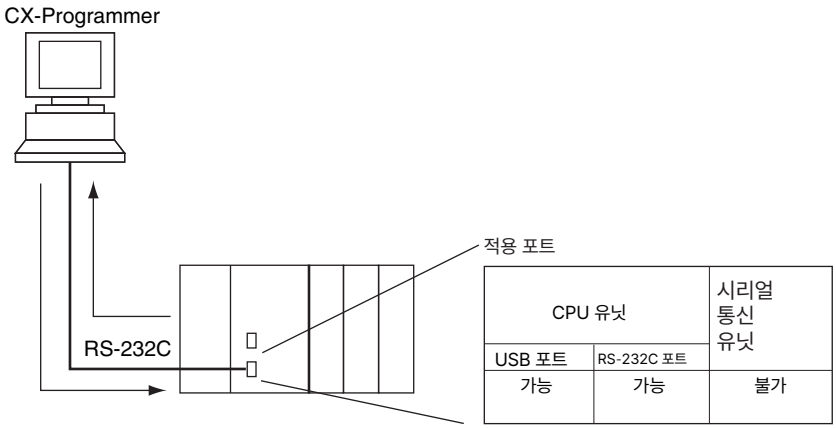
프로토콜 (시리얼 통신 모드)	주요 접속 대상	용도	적용 커맨드, 통신 명령어
페리페럴 버스(toolbus) (CPU 유닛의 RS-232C 포트에 한함. 시리얼 통신 유닛에서는 지원하지 않습니다.)	CX-Programmer	컴퓨터에서 프로그래밍 디바이스와 PLC 사이의 통신.	없음
호스트 링크(SYSMAC WAY) 슬레이브	퍼스널 컴퓨터 프로그래머블 터미널	호스트 컴퓨터와 PLC 사이의 통신. PLC에서 컴퓨터로 커맨드를 송신할 수 있습니다.	Host Link 커맨드/FINS 커맨드. PLC에서 컴퓨터로 커맨드를 송신할 수 있습니다.
무순서*1	범용 외부 기기	범용 기기와의 무순서 통신.	TXD(236) 명령어, RXD(235) 명령어, TXDU(256) 명령어, RXDU(255) 명령어 DTXDU (262) 명령어 DRXDU (261) 명령어
NT 링크(1: N)	오므론 프로그래머블 터미널	다이렉트 액세스를 통한 프로그래머블 터미널과의 고속 통신.	없음
시리얼 게이트웨이	오므론 부품 PLC	수신한 FINS 커맨드를 CompoWay/F, Modbus 또는 Host Link 프로토콜로 변환하여 변환된 커맨드를 시리얼 회선으로 송신합니다.	---
CompoWay/F 마스터*1	CompoWay/F 슬레이브	시리얼 포트에서 수신한 FINS 커맨드(캡슐화된 CompoWay/F 커맨드)를 CompoWay/F 커맨드로 변환합니다.	FINS 커맨드 2803 hex 수신 (CMND(490)을 사용한 FINS 커맨드 송신 포함)
Modbus 마스터*1 (시리얼 통신 유닛에 한함. CPU 유닛의 RS-232C 포트에서는 지원하지 않습니다.)	Modbus 슬레이브	시리얼 포트에서 수신한 FINS 커맨드(캡슐화된 Modbus 커맨드)를 Modbus 커맨드로 변환합니다.	FINS 커맨드 2804 hex 또는 2805 hex 수신 (CMND(490)을 사용한 FINS 커맨드 송신 포함)
Host Link FINS (SYSWAY) 마스터*1 (시리얼 통신 유닛에 한함. CPU 유닛의 RS-232C 포트에서는 지원하지 않습니다.)	Host Link FINS (SYSWAY) 슬레이브(PLC)	FINS 커맨드를 Host Link에 캡슐화된 FINS 커맨드로 변환합니다	시리얼 포트로 송신되는 것을 제외한 수신된 모든 FINS 커맨드 (CMND(490)을 사용한 FINS 커맨드 송신 포함)
프로토콜 매크로 (시리얼 통신 유닛에 한함. CPU 유닛의 RS-232C 포트에서는 지원하지 않습니다.)	범용 외부 기기	외부 기기의 통신 사양에 따라 메시지(통신 프레임)를 송수신합니다. (SYSMAC-PST를 사용하여 각종 파라미터를 설정함으로써 프로토콜을 작성합니다.)	PMCR/PMCR2(260) 명령어
시리얼 PLC 링크(CJ2M/CJ3M CPU 유닛에 한함)	오므론 PLC (CJ3M,CJ2M, CJ1M, CP1H, CP1L, CP1E, CP2E CPU 유닛)	최대 9대의 CPU 유닛(폴링 유닛 1대와 최대 8대의 폴드 유닛) 사이에서 유닛당 최대 10 워드의 데이터를 공유할 수 있습니다.	없음

*1 유닛 버전 1.2 이상의 시리얼 통신 유닛에 한합니다.

AI 번역본 -For Reference Only-

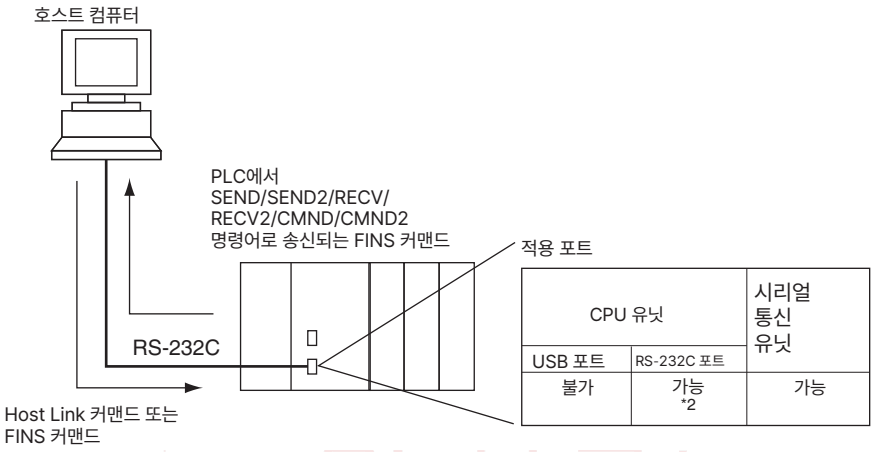
페리페럴 버스 (Toolbus)

CX-Programmer 등의 CX-One 지원 소프트웨어



호스트 링크 시스템 슬레이브(SYSMAC WAY 모드 1:N)

호스트 링크 시스템에서는 헤더와 터미네이터가 붙은 Host Link 커맨드 또는 FINS 커맨드를 실행하여 호스트 컴퓨터(퍼스널 컴퓨터*1 또는 프로그래머블 터미널)에서 PLC의 I/O 메모리를 읽고 쓰거나 운전 모드를 변경할 수 있습니다. 또한 PLC에서 네트워크 통신 명령어 (SEND/SEND2(090)/RECV/RECV2(098)/CMND/CMND2(490))를 실행하여, 호스트 링크 시스템을 경유하여 접속된 컴퓨터로 FINS 커맨드(헤더와 터미네이터가 붙은 것)를 송신할 수도 있습니다.



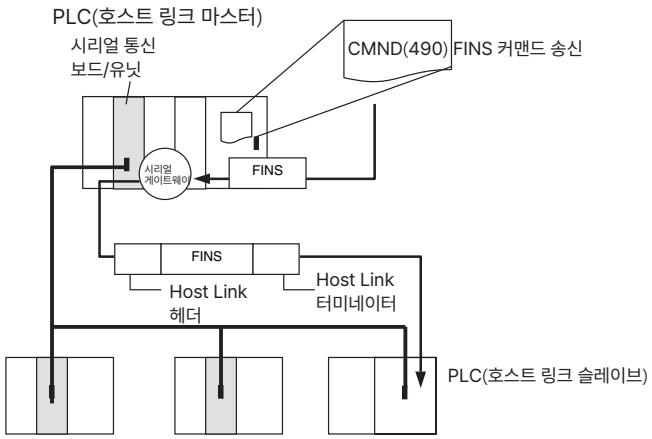
*1 CX-One 지원 소프트웨어는 제외합니다.

*2 CPU 유닛 전면 패널의 딥 스위치 핀 5를 OFF로 설정하고, PLC 설정의 시리얼 통신 모드를 호스트 링크로 설정하십시오.

-For Reference Only-

참고

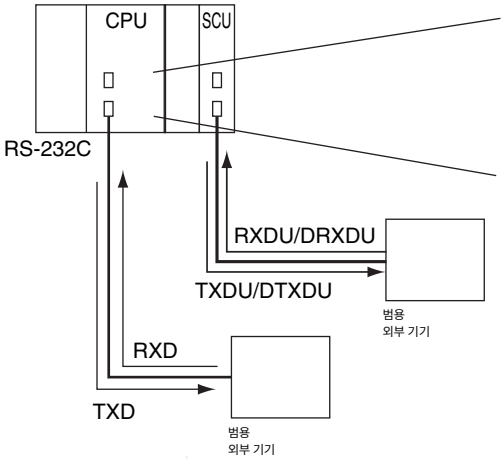
유닛 버전 1.2 이상의 시리얼 통신 유닛을 사용하는 경우, 시리얼 게이트웨이를 경유하여 CMND/CMND2(490) 명령어를 송신함으로써 호스트 링크 마스터 기능을 수행할 수 있습니다.



무순서 통신 시스템

무순서 통신에서는 통신 포트 I/O 명령어를 사용하여 바코드 데이터 입력, 프린터 데이터 출력과 같은 간단한 데이터 전송을 할 수 있습니다. 무순서 통신에서는 개시 코드와 종료 코드를 설정할 수 있고, RS · CS 신호 제어도 가능합니다.

다음 그림은 사용하는 통신 포트와 데이터 전송 방향(송신 또는 수신)에 따른 각 통신 포트 I/O 명령어의 사용법을 나타냅니다.



적용 포트

CPU 유닛		시리얼 통신 유닛
USB 포트	RS-232C 포트	
불가	가능 *1	가능 *2

*1 CPU 유닛 전면 패널의 딥 스위치 핀 5를 OFF로 설정하고, PLC 설정의 시리얼 통신 모드를 무순서 통신으로 설정하십시오.

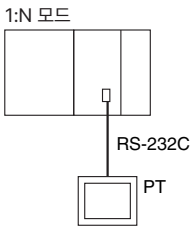
*2 무순서 통신은 유닛 버전 1.2 이상의 시리얼 통신 유닛에서만 지원됩니다.

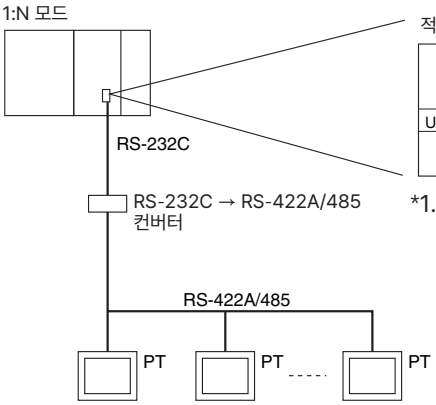
-For Reference Only-

NT 링크 시스템(1:N 모드)

PLC와 프로그래머블 터미널(PT)을 RS-232C 포트로 접속하면, PT의 스테이터스 제어 영역, 스테이터스 통지 영역, 터치 스위치 · 표시등 등의 오브젝트, 메모리 맵의 할당을 PLC의 I/O 메모리에 할당할 수 있습니다. NT 링크 시스템에서는 PLC가 PT를 제어할 수 있고, PT는 PLC의 스테이터스 제어 영역에서 주기적으로 데이터를 읽어, 그 영역에 변화가 있으면 필요한 동작을 수행합니다. PT는 PT에서 PLC의 스테이터스 통지 영역에 데이터를 씀으로써 PLC와 통신할 수 있습니다. NT 링크 시스템에서는 PLC 래더 프로그램을 사용하지 않고 PT 상태를 제어 · 모니터링할 수 있습니다. PLC와 PT의 비율은 1:N이며, 여기서 N은 1 이상입니다.

PT의 통신 설정을 1:N NT 링크로 설정하십시오. 각 PLC에는 PT를 1대 이상 접속할 수 있습니다.





적용 포트

CPU 유닛		시리얼 통신
USB 포트	RS-232C	유닛
불가	가능 *1	가능

*1. CPU 유닛 전면의 딥 스위치 핀 5를 OFF로 하고, PLC 설정의 시리얼 통신 모드를 NT 링크로 설정하십시오.

사용상의 주의

- PLC는 1:N NT 링크를 지원하는 모든 PT 포트에 접속할 수 있습니다. NT30 또는 NT30C의 RS-232C 포트는 1:1 NT 링크만 지원하므로 접속할 수 없습니다.
- CPU 유닛의 사이클 타임이 800 ms 이상이면 NT20S, NT600S, NT30, NT30C, NT620S, NT620C, NT625C는 사용할 수 없습니다(이들 PT를 1대만 접속한 경우에도 마찬가지입니다).
- 동일한 PLC에 여러 대의 PT를 접속하는 경우, 각 PT에 고유한 유닛 번호가 할당되었는지 반드시 확인하십시오. 동일한 유닛 번호가 2대 이상의 PT에 설정되면 오동작이 발생합니다.

For Reference Only

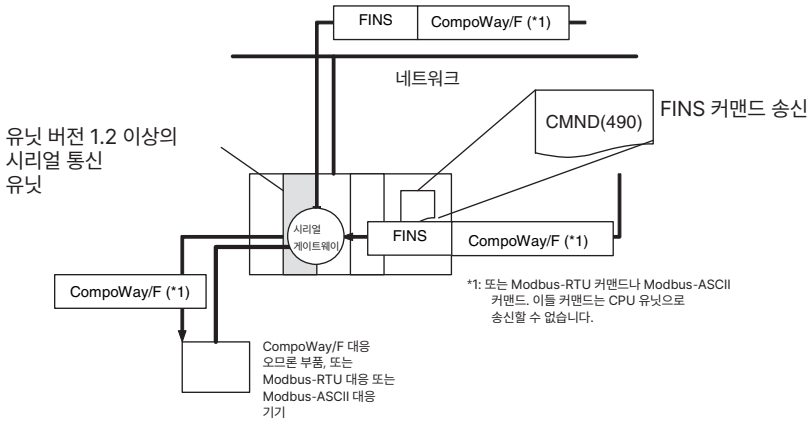
참고

1:1 NT 링크 프로토콜과 1:N NT 링크 프로토콜은 서로 호환되지 않습니다. 즉, 별개의 시리얼 통신 프로토콜입니다.

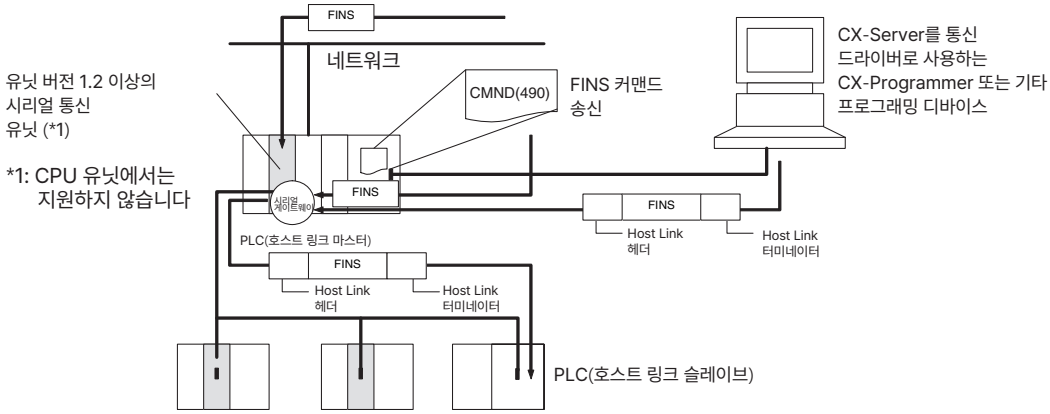
시리얼 게이트웨이 모드

수신한 FINS 메시지는 메시지에 따라 자동으로 CompoWay/F로 변환됩니다.
유닛 버전 1.2 이상의 시리얼 통신 유닛을 사용하는 경우, 수신한 FINS 메시지는 메시지에 따라 CompoWay/F, Modbus-RTU, Modbus-ASCII 또는 Host Link FINS 중 하나로 자동 변환됩니다.

● CompoWay/F, Modbus-RTU, Modbus-ASCII



● Host Link FINS

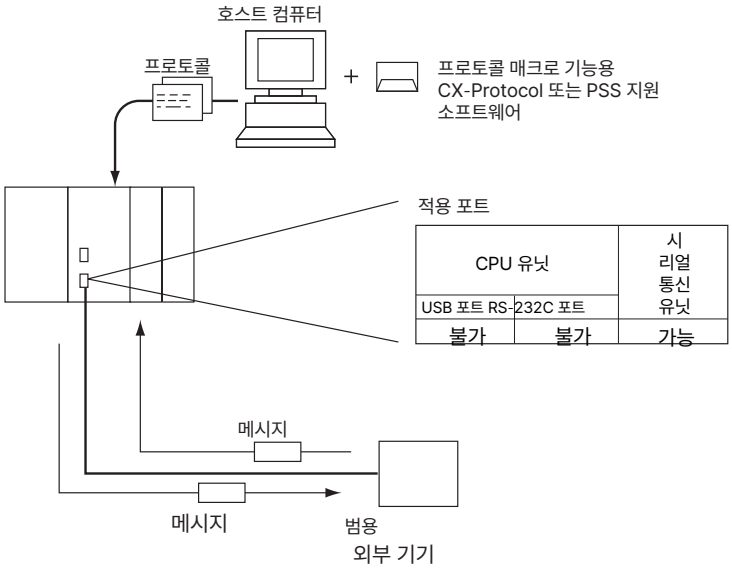


AI 번역본
-For Reference Only-

프로토콜 매크로(시리얼 통신 유닛 전용)

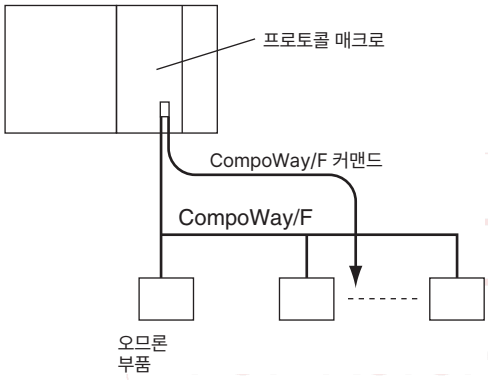
CX-Protocol은 범용 외부 기기의 통신 사양(반이중 또는 전이중, 비동기)에 따라 범용 외부 기기용 데이터 전송 순서(프로토콜)를 작성하는 데 사용합니다. 작성된 프로토콜은 시리얼 통신 유닛에 기록되며, CPU 유닛에서 PMCR/PMCR2(260) 명령어를 실행하는 것만으로 외부 기기와 데이터를 송수신할 수 있습니다. 온도 조절기, 인텔리전트 신호 처리기, 바코드 리더, 모뎀 등의 오므론 기기와 데이터 통신을 하기 위한 프로토콜이 표준 프로토콜로 제공됩니다. (*1) 사용자는 애플리케이션에 따라 이들 프로토콜을 변경할 수도 있습니다.

*1 표준 프로토콜은 CX-Protocol과 시리얼 통신 유닛에 부속되어 제공됩니다.



주 CompoWay/F(호스트 기능)

CJ 시리즈 CPU 유닛은 호스트로 동작하여 시스템에 접속된 오므론 부품으로 CompoWay/F 커맨드를 송신할 수 있습니다. CompoWay/F 커맨드는 프로토콜 매크로 기능의 표준 프로토콜에 있는 CompoWay/F 송수신 시퀀스를 사용하여 실행합니다.



역본
ence Only-

시리얼 PLC 링크

시리얼 PLC 링크는 CJ2M/CJ3M CPU 유닛에서만 지원됩니다. 시리얼 PLC 링크를 사용하면 특별한 프로그래밍 없이 CJ3M/CJ2M/CJ1M/CP1H/CP1L/CP1E/CP2E CPU 유닛 간에 데이터를 교환할 수 있습니다. (CJ2H CPU 유닛에서는 사용할 수 없습니다.)

시리얼 PLC 링크 영역(CIO 3100~CIO 3199)을 사용합니다. CPU 유닛은 RS-232C 또는 RS-422A/485로 접속합니다. CJ2M/CJ3M CPU 유닛의 RS-232C 시리얼 포트를 RS-422A/485로 변환하는 데에는 CJ1W-CIF11 RS-422A 컨버터를 사용합니다.

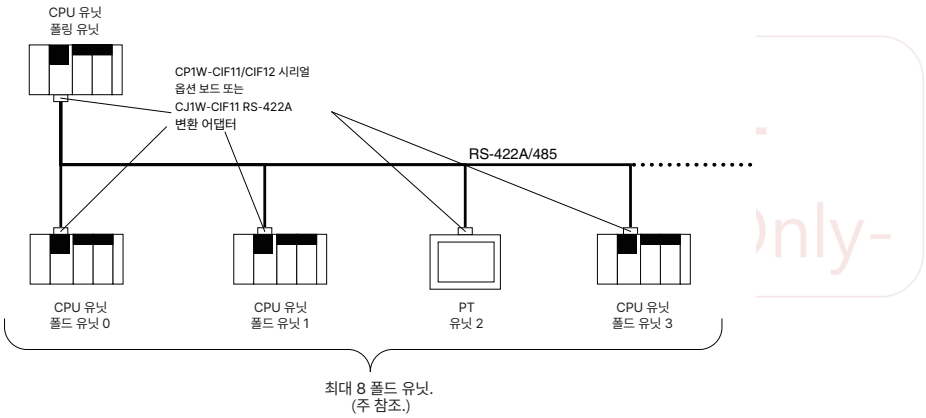
1:N NT 링크 프로토콜로 설정된 PT를 네트워크에 포함시킬 수 있습니다. 폴드 유닛으로 설정된 PT는 네트워크를 사용하여 1:N NT 링크 프로토콜로 폴링 유닛으로 설정된 CPU 유닛과 통신합니다. 다만 PT에 접속하는 경우, 그 PT에 대한 시리얼 PLC 링크 영역의 워드 내용은 안정되지 않습니다.

● 동작 사양

파라미터	설정
적용 PLC	CJ3M, CJ2M, CJ1M, CP1H, CP1L, CP1E, CP2E
보 레이트	115,200 또는 38,400 bps
적용 시리얼 포트	CPU 유닛의 시리얼 포트(CJ2M-CPU3□에는 시리얼 옵션 보드가 필요합니다.)
접속 방법	RS-232C 또는 RS-422A/485
할당 워드	시리얼 PLC 링크 영역 워드: CIO 3100~CIO 3199(CPU 유닛마다 최대 10 워드를 할당할 수 있습니다.) 주: CP1E/CP2E CPU 유닛에서는 시리얼 PLC 링크에 CIO 200~CIO 289가 할당됩니다.
최대 유닛 수	폴링 유닛 1대와 폴드 유닛 8대를 포함하여 최대 9 유닛. 동일 회선에서 PT가 1:N NT 링크 프로토콜로 설정된 경우, 최대 9 유닛에는 폴링 대상 PT와 CPU 유닛이 포함되어야 합니다.
링크 방식(데이터 리프레시 방식)	완전 링크 방식 또는 폴링 유닛 링크 방식

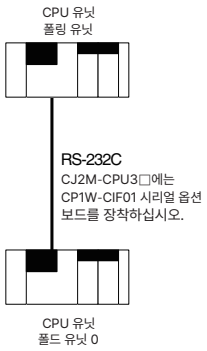
● 시스템 구성

CJ2M, CP1L, CP1H, CP1E 또는 CJ1M CPU 유닛을 1:N으로 접속(최대 8 노드)



주: 동일 회선에서 PT가 시리얼 PLC 링크 프로토콜로 설정된 경우, 최대 8 유닛에는 폴링 대상 PT와 CPU 유닛이 포함되어야 합니다.

CJ2M, CP1L, CP1H, CP1E 또는 CJ1M CPU 유닛을 1:1로 접속



사용상의 주의

CJ1M-CIF11과 CP1W-CIF11은 절연되어 있지 않습니다. 따라서 최대 전송 거리는 50 m입니다. 거리가 50 m를 초과하는 경우에는 NT-AL001(절연형)을 사용하고, 전송 경로의 어느 곳에도 CJ1W-CIF11이나 CP1W-CIF11을 사용하지 마십시오. NT-AL001만 사용하는 경우 최대 총 전송 거리는 500 m가 됩니다.

● 순서

시리얼 PLC 링크는 폴링 유닛과 폴드 유닛의 PLC 설정에서 다음 설정에 따라 동작합니다.

폴링 유닛 설정

1. RS-232C 포트의 시리얼 통신 모드를 시리얼 PLC 링크, 폴링 유닛으로 설정합니다.
2. 링크 방식을 All 또는 Polling Unit으로 설정합니다.
3. 링크할 워드 수를 설정합니다. (1~10)
4. 시리얼 PLC 링크에 사용할 최대 유닛 번호를 설정합니다. (0~7)

폴드 유닛 설정

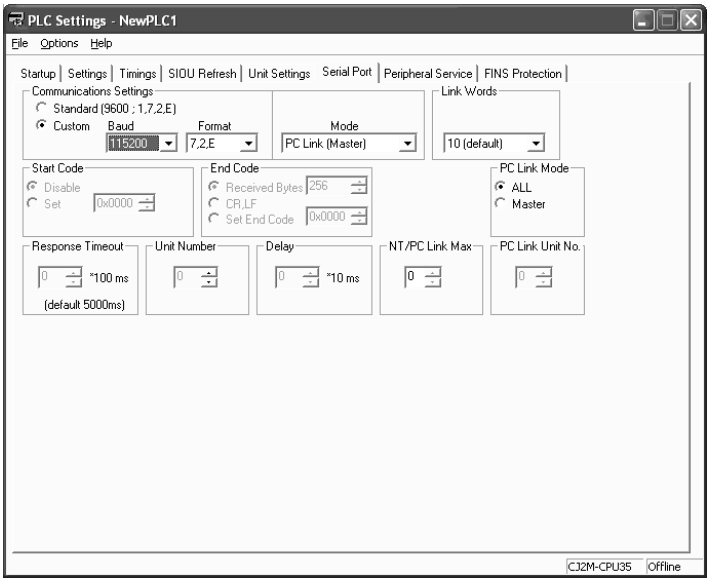
1. RS-232C 포트의 시리얼 통신 모드를 시리얼 PLC 링크, 폴드 유닛으로 설정합니다.
2. 시리얼 PLC 링크에서 폴드 유닛의 유닛 번호를 설정합니다.

AI 번역본
-For Reference Only-

● PLC 설정

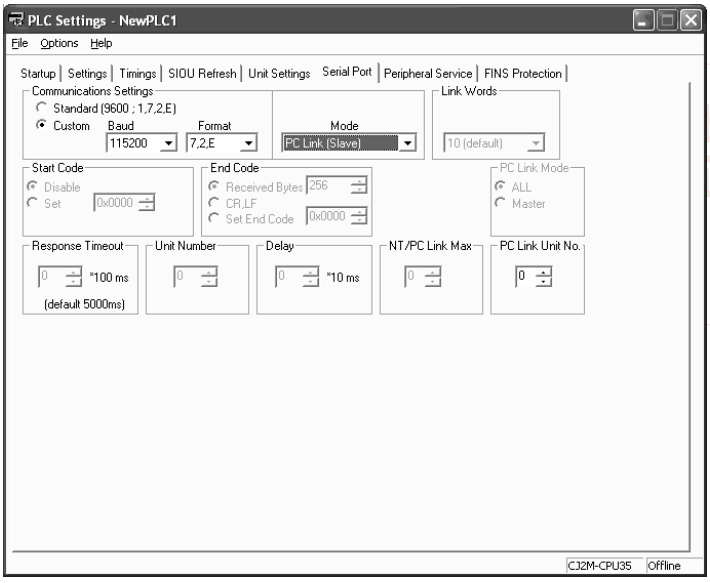
폴링 유닛에서의 설정

Serial Port 탭 페이지



파라미터	설정
통신 설정	접속된 PLC와 일치하도록 통신 설정을 설정합니다. 접속된 PLC가 115,200 bps로 설정되어 있는 경우, Custom 옵션을 선택하고 보레이트를 115,200으로 설정합니다. Parameter 설정에서 무엇을 선택하든 상관없습니다.
모드	PC Link Polling Unit을 선택합니다.
링크 워드 수	이 파라미터는 폴링 유닛에서만 설정합니다. 10 워드(디폴트)
PC 링크 방식	All 또는 Polling Unit을 선택합니다.
NT/시리얼 PLC 링크의 최대 유닛 번호(No. NT/PC Link Max.)	접속된 폴드 유닛의 최대 유닛 번호를 설정합니다.

폴드 유닛에서의 설정



-Only-

파라미터	설정
통신 설정	접속된 PLC와 일치하도록 통신 설정을 설정합니다. 접속된 PLC가 115,200 bps로 설정되어 있는 경우, Custom 옵션을 선택하고 보레이트를 115,200으로 설정합니다. Parameter 설정에서 무엇을 선택하든 상관없습니다.
모드	PC Link Polled Unit을 선택합니다.
시리얼 PLC 링크 유닛 번호	유닛 번호를 설정합니다(0~7).

● 링크 방식(데이터 리프레시 방식)

데이터 리프레시에는 다음 두 가지 방식을 사용할 수 있습니다.

・완전 링크 방식

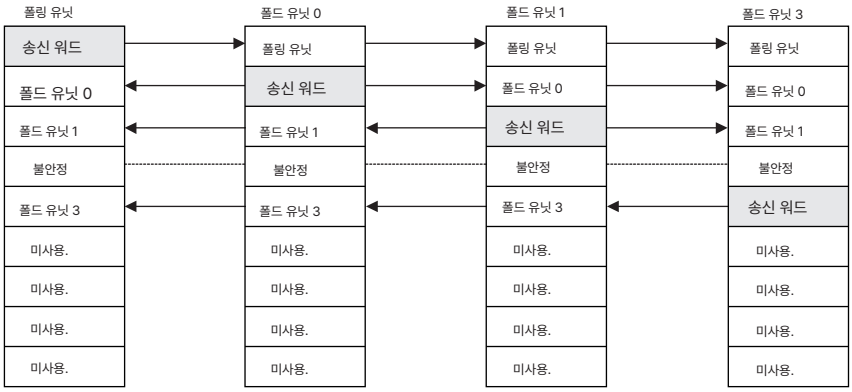
・폴링 유닛 링크 방식

완전 링크 방식

시리얼 PLC 링크의 모든 노드의 데이터가 폴링 유닛과 폴드 유닛 양쪽에서 갱신됩니다. 예외는 네트워크에 존재하지 않는 폴드 유닛의 데이터와 접속된 PT의 데이터뿐입니다. 이들 데이터 영역은 모든 노드에서 불안정합니다.

예: 완전 링크 방식, 최대 유닛 번호: 3

다음 그림에서 폴드 유닛 2는 네트워크에 존재하지 않는 유닛이거나 PT이므로, 폴드 유닛 2에 할당된 워드는 모든 노드에서 불안정합니다.



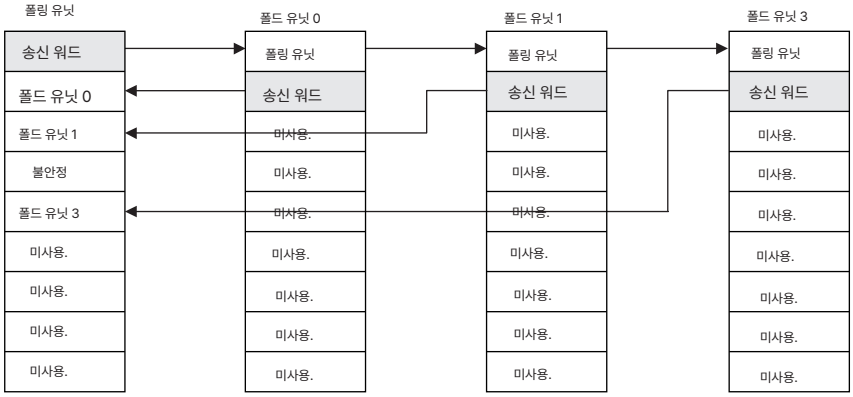
폴링 유닛 링크 방식

시리얼 PLC 링크의 모든 폴드 유닛의 데이터는 폴링 유닛에서만 갱신되고, 각 폴드 유닛은 폴링 유닛의 데이터만 갱신합니다. 폴링 유닛 링크 방식의 이점은 자체 폴드 유닛 데이터에 할당되는 어드레스가 각 폴드 유닛에서 동일하여 공통 래더 프로그래밍으로 데이터에 액세스할 수 있다는 점입니다. 네트워크에 존재하지 않는 폴드 유닛의 데이터와 접속된 PT의 데이터는 모든 노드에서 불안정합니다.

-For Reference Only-

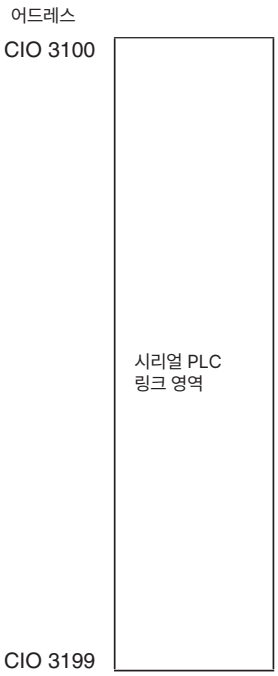
예: 폴링 유닛 링크 방식, 최대 유닛 번호: 3

다음 그림에서 폴드 유닛 2는 네트워크에 존재하지 않는 유닛이거나 PT이므로, 이 데이터는 폴링 유닛에서 불안정합니다.



● 할당 워드

완전 링크 방식

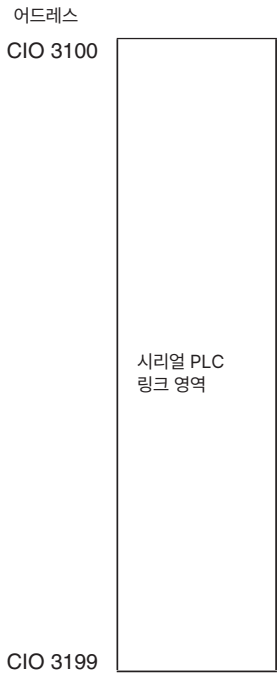


링크 워드 수	1 워드	2 워드	3 워드	~	10 워드
폴링 유닛	CIO 3100	CIO 3100~ CIO 3101	CIO 3100~ CIO 3102		CIO 3100~ CIO 3109
폴드 유닛 0	CIO 3101	CIO 3102~ CIO 3103	CIO 3103~ CIO 3105		CIO 3110~ CIO 3119
폴드 유닛 1	CIO 3102	CIO 3104~ CIO 3105	CIO 3106~ CIO 3108		CIO 3120~ CIO 3129
폴드 유닛 2	CIO 3103	CIO 3106~ CIO 3107	CIO 3109~ CIO 3111		CIO 3130~ CIO 3139
폴드 유닛 3	CIO 3104	CIO 3108~ CIO 3109	CIO 3112~ CIO 3114		CIO 3140~ CIO 3149
폴드 유닛 4	CIO 3105	CIO 3110~ CIO 3111	CIO 3115~ CIO 3117		CIO 3150~ CIO 3159
폴드 유닛 5	CIO 3106	CIO 3112~ CIO 3113	CIO 3118~ CIO 3120		CIO 3160~ CIO 3169
폴드 유닛 6	CIO 3107	CIO 3114~ CIO 3115	CIO 3121~ CIO 3123		CIO 3170~ CIO 3179
폴드 유닛 7	CIO 3108	CIO 3116~ CIO 3117	CIO 3124~ CIO 3126		CIO 3180~ CIO 3189
미사용.	CIO 3109~ CIO 3199	CIO 3118~ CIO 3199	CIO 3127~ CIO 3129		CIO 3190~ CIO 3199

AI 번역본
-For Reference Only-

폴링 유닛 링크 방식

마스터 할당



링크 워드 수	1 워드	2 워드	3 워드	~	10 워드
폴링 유닛 (출력)	CIO 3100	CIO 3100~ CIO 3101	CIO 3100~ CIO 3102		CIO 3100~ CIO 3109
폴드 유닛 0 (입력)	CIO 3101	CIO 3102~ CIO 3103	CIO 3103~ CIO 3105		CIO 3110~ CIO 3119
폴드 유닛 1 (입력)	CIO 3102	CIO 3104~ CIO 3105	CIO 3106~ CIO 3108		CIO 3120~ CIO 3129
폴드 유닛 2 (입력)	CIO 3103	CIO 3106~ CIO 3107	CIO 3109~ CIO 3111		CIO 3130~ CIO 3139
폴드 유닛 3 (입력)	CIO 3104	CIO 3108~ CIO 3109	CIO 3112~ CIO 3114		CIO 3140~ CIO 3149
폴드 유닛 4 (입력)	CIO 3105	CIO 3110~ CIO 3111	CIO 3115~ CIO 3117		CIO 3150~ CIO 3159
폴드 유닛 5 (입력)	CIO 3106	CIO 3112~ CIO 3113	CIO 3118~ CIO 3120		CIO 3160~ CIO 3169
폴드 유닛 6 (입력)	CIO 3107	CIO 3114~ CIO 3115	CIO 3121~ CIO 3123		CIO 3170~ CIO 3179
폴드 유닛 7 (입력)	CIO 3108	CIO 3116~ CIO 3117	CIO 3124~ CIO 3126		CIO 3180~ CIO 3189
미사용.	CIO 3109~ CIO 3199	CIO 3118~ CIO 3199	CIO 3127~ CIO 3129		CIO 3190~ CIO 3199

슬레이브 할당(유닛 0의 예)

링크 워드 수	1 워드	2 워드	3 워드	~	10 워드
폴링 유닛 (입력)	CIO 3100	CIO 3100~ CIO 3101	CIO 3100~ CIO 3102		CIO 3100~ CIO 3109
폴드 유닛 0 (출력)	CIO 3101	CIO 3102~ CIO 3103	CIO 3103~ CIO 3105		CIO 3110~ CIO 3119

AI 번역본
-For Reference Only-

● 관련 보조 기억 영역 플래그 및 워드

명칭	어드레스	기능	읽기/쓰기	리프레시 타이밍
시리얼 포트 PT 통신 플래그*	A393.00~ A393.07	시리얼 포트가 NT 링크 모드 또는 시리얼 PLC 링크 모드로 통신 중일 때 해당 비트가 ON 됩니다. 비트 0~7은 유닛 0~7에 대응합니다. ON: 통신 중 OFF: 통신하지 않음	읽기	· 전원을 ON 하면 클리어됩니다. · 시리얼 포트가 NT 링크 또는 시리얼 PLC 링크 모드일 때 갱신됩니다. · 비트 0~7은 유닛 0~7에 대응합니다.
시리얼 포트 재기동 비트	A526.00	시리얼 포트를 재기동하려면 ON 합니다.	읽기/쓰기	· 전원을 ON 하면 클리어됩니다. · Toolbus 모드를 제외한 모든 모드에서 시리얼 포트를 재기동하려면 이 비트를 ON 합니다. 주: 재기동 처리가 완료되면 이 비트는 자동으로 OFF 됩니다.
시리얼 포트 에러 플래그	A528.00~ A528.07	이들 플래그는 시리얼 포트에서 어떤 종류의 에러가 발생했는지를 나타냅니다. 비트 0: 미사용. 비트 1: 미사용. 비트 2: 패리티 에러 비트 3: 프레이밍 에러 비트 4: 오버런 에러 비트 5: 타임아웃 에러 비트 6: 미사용. 비트 7: 미사용.	읽기/쓰기	· 전원을 ON 하면 클리어됩니다. · 이들 플래그는 시리얼 포트에서 어떤 종류의 에러가 발생했는지를 나타냅니다. · 시리얼 포트가 재기동되면 시스템에 의해 자동으로 OFF 됩니다. · 시리얼 PLC 링크 모드에서는 다음 비트만 유효합니다. 폴링 유닛: 비트 5: 타임아웃 에러 시 ON. 폴드 유닛: 비트 3: 프레이밍 에러 시 ON. 비트 4: 오버런 에러 시 ON. 비트 5: 타임아웃 에러 시 ON.
시리얼 포트 설정 변경 중 플래그	A619.02	시리얼 포트의 통신 파라미터가 변경되는 중일 때 ON 됩니다. ON: 변경 중 OFF: 변경 중 아님	읽기/쓰기	· 전원을 ON 하면 클리어됩니다. · 시리얼 포트의 통신 파라미터가 변경되는 중일 때 ON 됩니다. · 시리얼 포트 설정 변경 (STUP(237)) 명령어가 실행될 때 ON 됩니다. · 파라미터 변경이 정상적으로 완료되면 OFF 됩니다.

* 기존 1:N NT 링크와 마찬가지로, 시리얼 PLC 링크에서 폴드 유닛의 상태(통신 중/통신하지 않음)는 폴링 유닛(CPU 유닛)에서 시리얼 포트 PT 통신 플래그(유닛 번호 0~7에 대해 A393.00~A393.07)를 읽어 확인할 수 있습니다.

AI 번역본
-For Reference Only-

유닛/프로토콜 대응

유닛	형식	포트	프로토콜(시리얼 통신 모드)						
			페리 페럴 버스 (toolbus)	호스트 링크 (CX-One 제외)	무순서 통 신	프로토콜 매크로	NT 링크 (1:N 모드)	시리얼 게이트 웨이*2	시리얼 PLC 링크
CPU 유닛	CJ2H-CPU6□ (-EIP)	RS-232C	가능	가능	가능	---	가능	가능	불가
	CJ2M-CPU3□	RS-232C 또는 RS-422A/485*1	가능	가능	가능	---	가능	가능	가능
	CJ2M-CPU1□ CJ2M-CPU3□ CJ3M-CPU3□	RS-232C	가능	가능	가능	---	가능	가능	가능
시리얼 통신 유닛	CJ1W-SCU41-V1 CJ1W-SCU31-V1 CJ1W-SCU21-V1 CJ1W-SCU22 CJ1W-SCU32 CJ1W-SCU42	RS422/485	---	가능	가능*2	가능	가능	가능	불가
		RS-232C	---	가능	가능*2	가능	가능	가능	불가

*1 시리얼 옵션 보드를 장착하면 RS-232C 또는 RS-422A/485 통신을 할 수 있습니다.

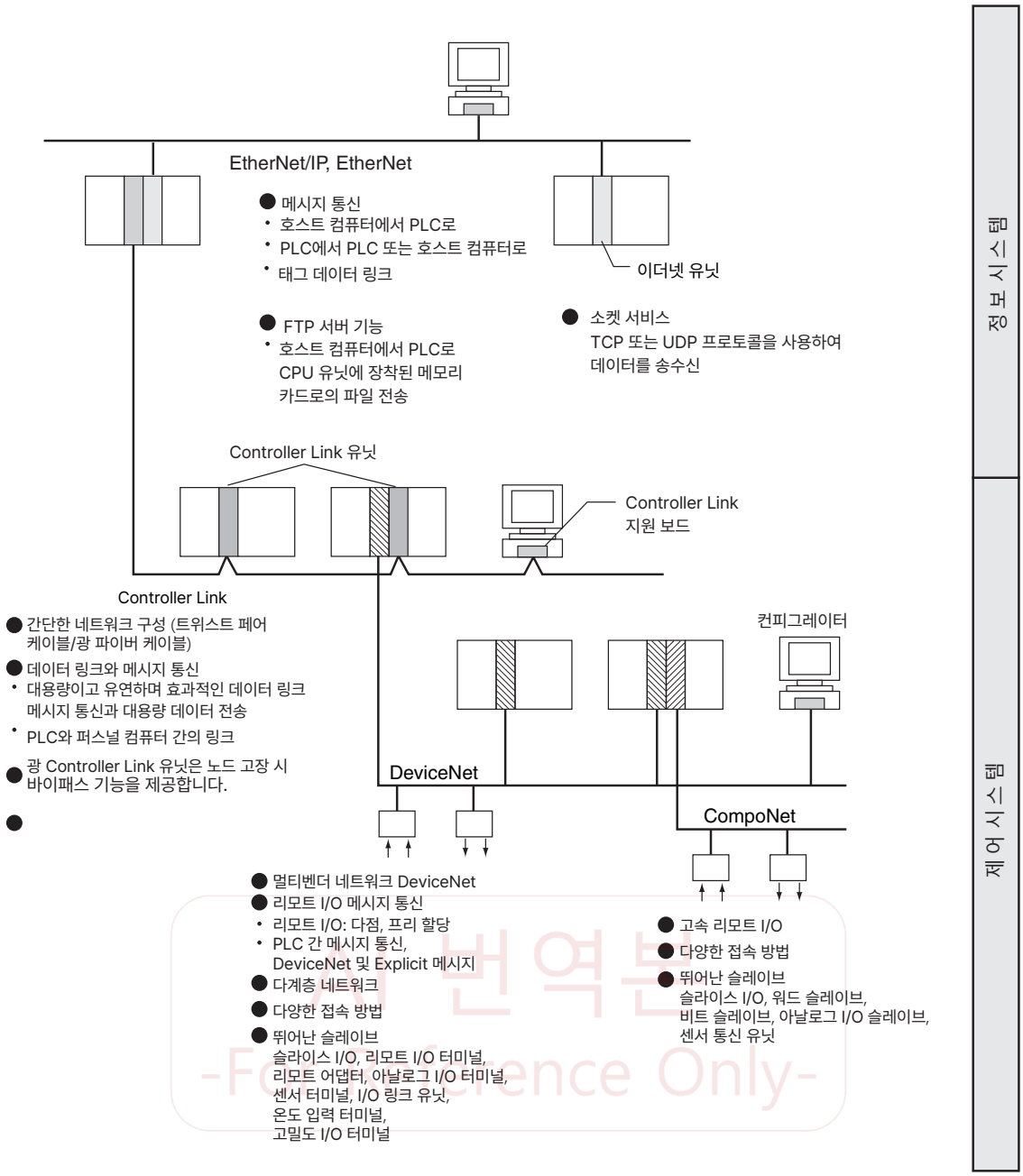
*2 유닛 버전 1.2 이상의 시리얼 통신 유닛에서만 지원됩니다. 다만 CPU 유닛에서는 시리얼 게이트웨이 프로토콜에 대해 CompoWay/F로의 자동 변환만 가능합니다.

AI 번역본
-For Reference Only-

11-3 통신 네트워크

통신 네트워크 구성

CJ 시리즈 PLC를 사용하면 다음 네트워크를 구성할 수 있습니다.



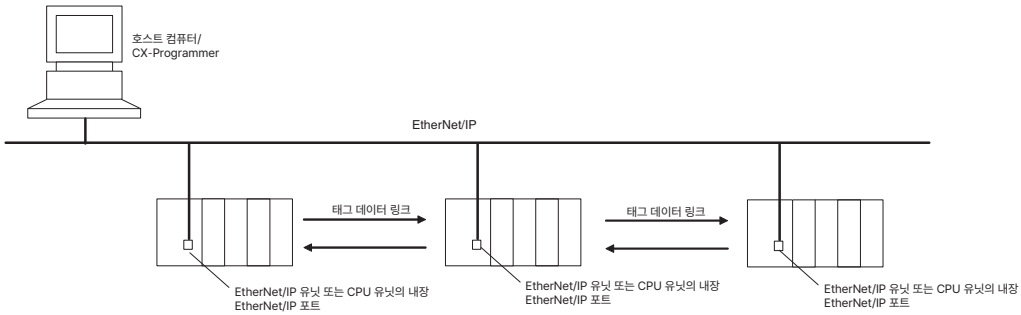
참고 번역본
-For Reference Only-

● EtherNet/IP와 이더넷

이더넷 케이블을 통해 태그 데이터 링크와 FINS 네트워크 통신을 할 수 있습니다. 또한 이더넷에 접속된 호스트 컴퓨터에서 PLC에 대해 FTP 커맨드를 실행하면, CPU 유닛에 장착된 메모리 카드의 파일 내용을 읽거나 쓸 수 있습니다. UDP와 TCP 프로토콜을 사용하여 데이터를 송수신할 수 있습니다. 이들 기능으로 호스트 정보 네트워크와의 호환성을 높일 수 있습니다.

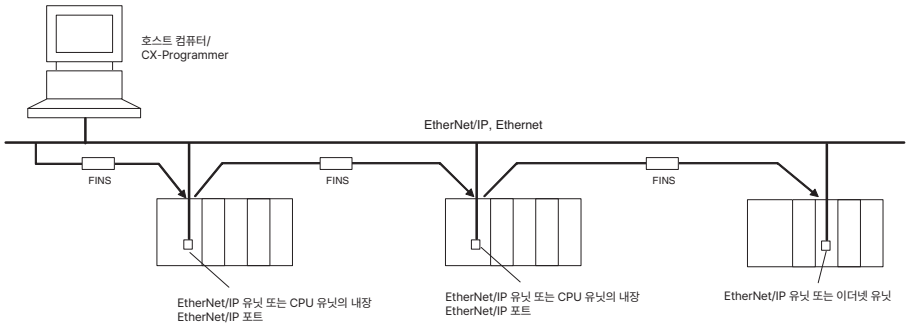
태그 데이터 링크

두 PLC 간에 데이터 링크를 실행하여 영역마다 사용자가 설정한 리프레시 주기로 고속 · 대용량 데이터 전송을 할 수 있습니다.



FINS 네트워크 통신

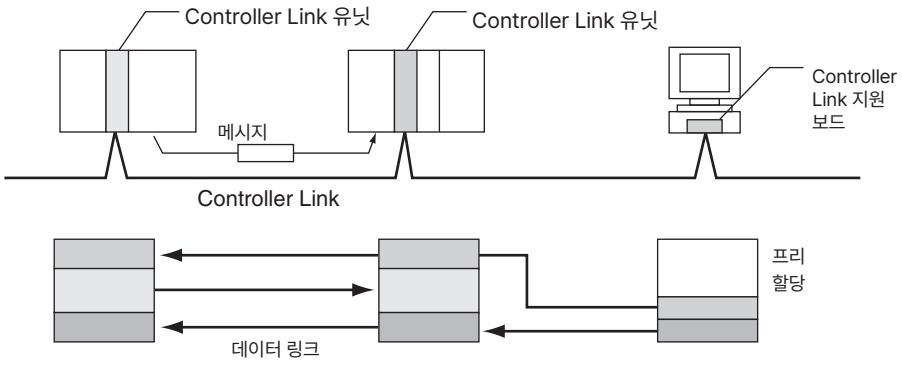
내장 EtherNet/IP 포트에 접속하거나 EtherNet/IP 유닛 또는 이더넷 유닛이 접속된 CPU 유닛에 접속하면, 이더넷 네트워크상의 다른 PLC와, 또는 PLC와 호스트 컴퓨터 사이에 FINS 메시지 통신을 할 수 있습니다.



● Controller Link

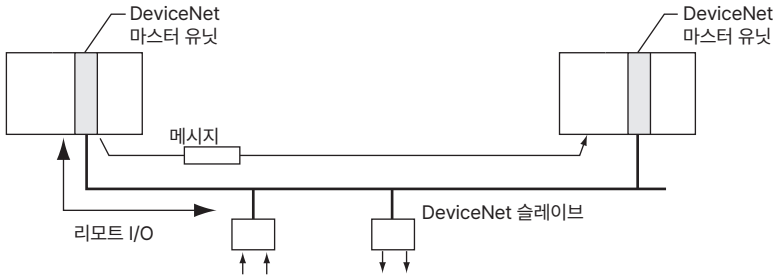
Controller Link 네트워크는 오므론 PLC FA 전용 네트워크입니다. Controller Link 유닛을 네트워크에 접속하면 PLC 간 데이터 링크가 가능해져 프로그래밍 없이 데이터를 공유할 수 있고, PLC 간 FINS 메시지 통신이 가능해져 필요할 때 개별 제어와 데이터 전송을 할 수 있습니다. Controller Link 네트워크 접속에는 트위스트 페어 케이블 또는 광 파이버 케이블을 사용합니다. PLC와 퍼스널 컴퓨터 간에도 데이터 링크와 메시지 통신이 가능합니다. 데이터 링크는 대용량과 프리 할당이 가능합니다. FINS 메시지 통신도 대용량 데이터 전송이 가능합니다.

-For Reference Only-



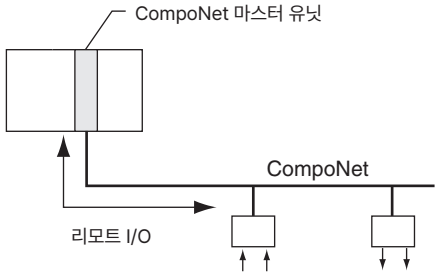
DeviceNet

DeviceNet은 멀티비트 제어 · 정보 시스템으로 구성되며 Open Field DeviceNet 사양에 준거한 멀티벤더 네트워크입니다. DeviceNet 마스터 유닛을 네트워크에 접속하면 PLC와 네트워크상의 슬레이브 간에 리모트 I/O 통신을 할 수 있습니다. 리모트 I/O 통신은 대용량 I/O와 사용자 설정 할당이 가능합니다. 슬레이브로는 아날로그 I/O 터미널이 사용됩니다. PLC 간, 그리고 PLC와 타사제 DeviceNet 기기 간에 메시지 통신이 가능합니다.



CompoNet

CompoNet은 주로 I/O 제어에 사용되는 고속 · 다점 · 다노드 멀티벤더 네트워크입니다.
PLC에 CompoNet 마스터 유닛을 접속하면 PLC와 CompoNet 슬레이브 간에 리모트 I/O 통신을 할 수 있습니다. 1,024점을 최대 1 ms의 사이클 타임으로 고속 통신합니다. 최대 2,560점, 384 노드를 접속할 수 있습니다.



AI 번역본
-For Reference Only-

통신 네트워크 개요

시스템	네트워크	기능	통신	통신 기기
정보 네트워크	EtherNet/IP	호스트 컴퓨터와 PLC 간.	FINS 메시지 통신	CPU 유닛 내장 Ether-Net/IP 포트 EtherNet/IP 유닛
		PLC 간.		
		호스트 컴퓨터와 CPU 유닛에 장착된 메모리 카드 간.	FTP 서버	
	Controller Link	PLC 와 네트워크에 직접 접속된 퍼스널 컴퓨터 간.	FINS 메시지 통신	Controller Link 서포트 보드 또는 Controller Link 유닛
			데이터 링크(오프셋, 간이 설정)	
	RS-232C → Controller Link	호스트 링크 컴퓨터와 네트워크상의 PLC 간.	호스트 링크 커맨드 및 게이트웨이.	RS-232C 케이블과 Con-troller Link 유닛
제어 네트워크	EtherNet/IP	PLC 간.	태그 데이터 링크	CPU 유닛 내장 Ether-Net/IP 포트 EtherNet/IP 유닛
	Controller Link	PLC 간.	FINS 메시지 통신	Controller Link 유닛
			데이터 링크(오프셋, 간이 설정)	
	DeviceNet		오픈 네트워크에서의 FINS 메시지 통신.	DeviceNet 마스터 유닛 및 컴피그레이터
	DeviceNet	PLC 와 네트워크 기기 (슬레이브).	오픈 네트워크에서의 대용량 리모트 I/O(고정 할당 또는 프리 할당)	DeviceNet 마스터 유닛 및 컴피그레이터
	CompoNet		오픈 네트워크에서의 고속 · 다점 · 다노드 리모트 I/O	CompoNet 마스터 유닛

AI 번역본
-For Reference Only-

AI 번역본
-For Reference Only-

12

CPU 유닛의 사이클 타임

이 섹션에서는 CPU 유닛의 처리에 사용되는 사이클 타임에 대해 설명합니다.

12-1 사이클 타임 모니터링 12-2
12-1-1 사이클 타임 모니터링 12-2
12-2 사이클 타임 계산 12-4
12-2-1 CPU 유닛 동작 플로우차트 12-4
12-2-2 사이클 타임 개요 12-5
12-2-3 유닛별 I/O 유닛 리프레시 시간 12-7
12-2-4 사이클 타임 계산 예 12-11
12-2-5 온라인 에디팅에 의한 사이클 타임 연장 12-13
12-2-6 I/O 응답 시간 12-13
12-2-7 펄스 I/O 모듈로의 입력 인터럽트 응답 시간 12-14
12-2-8 시리얼 PLC 링크의 응답 성능 12-15

AI 번역본
-For Reference Only-

12-1 사이클 타임 모니터링

12-1-1 사이클 타임 모니터링

CX-Programmer 를 CPU 유닛에 온라인으로 접속하면 사이클 타임의 평균값 · 최대값 · 최소값을 모니터링할 수 있습니다.

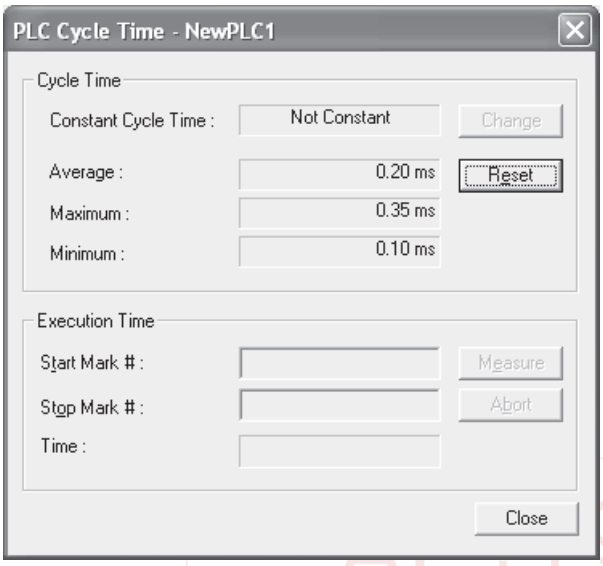
평균값 모니터링

PLC 에 온라인으로 접속되어 있을 때 CPU 유닛이 PROGRAM 모드 이외의 모드이면 스테이터스 바에 평균 사이클 타임이 표시됩니다.



최대값 · 최소값 모니터링

PLC 메뉴에서 Edit – Cycle Time 을 선택합니다. 다음의 PLC Cycle Time 대화 상자가 표시됩니다.



평균값, 최대값, 최소값의 사이클 타임이 위에서부터 차례로 표시됩니다. Reset 버튼을 클릭하면 사이클 타임 값을 다시 계산해 표시합니다.

-For Reference Only-

참고

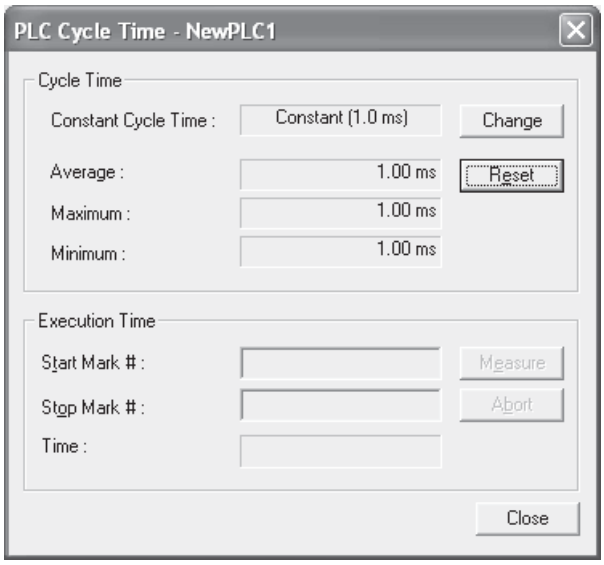
사이클 타임의 평균값(= 현재값)과 최대값은 다음의 보조 기억 영역 워드에 저장됩니다.

- 현재 사이클 타임(= 현재값) (단위: 0.1 ms)
  A264(하위 바이트) 및 A265(상위 바이트)
- 현재 사이클 타임(= 현재값) (단위: 0.01 ms)
  A266(하위 바이트) 및 A267(상위 바이트)
- 최대 사이클 타임(단위: 0.01 ms)
  A262(하위 바이트) 및 A263(상위 바이트)

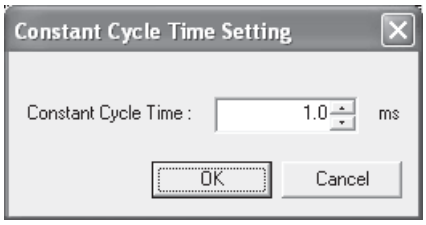
최소 사이클 타임 변경

최소 사이클 타임이 설정되어 있는 경우, CPU 유닛이 MONITOR 모드일 때 최소 사이클 타임을 변경할 수 있습니다.

1 PLC 메뉴에서 PLC Info - Cycle Time 을 선택합니다. 아래와 같이 PLC Cycle Time 대화 상자가 표시됩니다.



2 Change 버튼을 클릭합니다. Constant Cycle Time Setting 대화 상자가 표시됩니다.



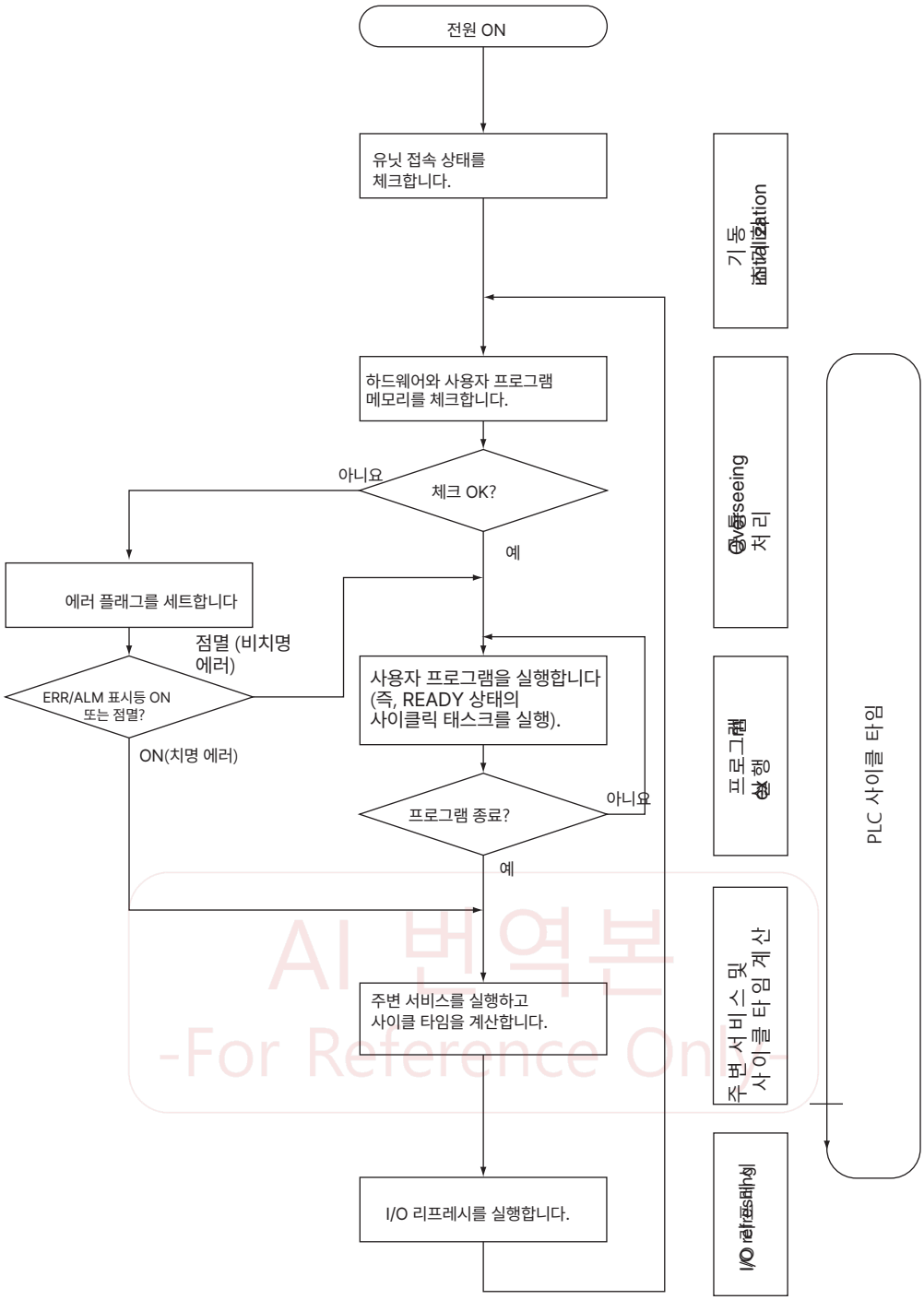
3 Constant Cycle Time 필드에서 시간을 변경한 후 OK 버튼을 클릭합니다. 최소 사이클 타임이 변경됩니다.

AI 번역본
-For Reference Only-

12-2 사이클 타임 계산

12-2-1 CPU 유닛 동작 플로우차트

CJ 시리즈 CPU 유닛은 다음 그림과 같이 공통 처리부터 I/O 리프레시까지를 반복하는 사이클로 데이터를 처리합니다.



AI 번역본
-For Reference Only-

12-2-2 사이클 타임 개요

사이클 타임은 다음 조건에 따라 달라집니다.

- 사용자 프로그램 내 명령어의 종류와 개수(1 사이클 중에 실행되는 모든 사이클릭 태스크 및 실행 조건이 성립한 인터럽트 태스크 내).
- 기본 I/O 유닛의 종류와 대수
- 특수 I/O 유닛 및 CPU 버스 유닛의 종류와 대수.
- 다음 유닛/보드에 대한 고유 서비스
  - EtherNet/IP 태그 데이터 링크 리프레시와 교환되는 태그 데이터 링크 워드 수
  - Controller Link 의 데이터 링크 리프레시와 데이터 링크 워드 수
  - DeviceNet 유닛의 리모트 I/O 와 리모트 I/O 워드 수
  - 프로토콜 매크로의 사용과 최대 통신 메시지
  - 이더넷 유닛의 특정 제어 비트에 대한 소켓 서비스와 송수신 워드 수
- PLC 설정의 고정 사이클 타임 설정
- 파일 메모리의 파일 액세스 및 파일 메모리와 주고받는 데이터 양
- 특수 I/O 유닛, CPU 버스 유닛 및 통신 포트에 대한 이벤트 서비스
- 주변 USB 포트 및 시리얼 포트의 사용
- PLC 설정의 고정 주변 서비스 실행 시간

주1 사이클 타임은 사용자 프로그램에서 사용하는 태스크 수의 영향을 받지 않습니다. 사이클 타임에 영향을 주는 태스크는 그 사이클에서 READY 상태인 사이클릭 태스크입니다.

2 MONITOR 모드에서 RUN 모드로 전환하면 사이클 타임이 10 ms 연장됩니다 (단, 이로 인해 사이클 타임이 제한을 초과하지는 않습니다).

사이클 타임은 PLC 가 다음 표에 나타낸 5 가지 처리를 수행하는 데 필요한 시간의 합계입니다.

사이클 타임 = (1) + (2) + (3) + (4) + (5)

1. 공통 처리

상세	처리 시간과 변동 요인
I/O 버스와 사용자 프로그램 메모리를 체크하고, 배터리 에러를 체크하며 시계를 갱신합니다.	CJ2H-CPU6□-EIP: 0.2 ms CJ2H-CPU6□: 0.1 ms CJ2M-CPU1□: 0.16 ms CJ2M-CPU3□: 0.27 ms CJ3M-CPU1□: 0.16 ms CJ3M-CPU3□: 0.26 ms

2. 프로그램 실행

상세	처리 시간과 변동 요인
사용자 프로그램을 실행하고, 명령어가 프로그램을 실행하는 데 걸린 합계 시간을 계산합니다.	명령어 실행 시간의 합계

3. 사이클 타임 계산

상세	처리 시간과 변동 요인
PLC 설정에서 최소(고정) 사이클 타임이 설정되어 있는 경우, 지정한 사이클 타임이 경과할 때까지 대기합니다. 사이클 타임을 계산합니다.	사이클 타임이 고정되어 있지 않은 경우, 3 의 시간은 거의 0 입니다. 사이클 타임이 고정되어 있는 경우, 3 의 시간은 설정된 고정 사이클 타임에서 실제 사이클 타임((1) + (2) + (4) + (5))을 뺀 값입니다.

AI 번역본 - For Reference Only -

4. I/O 리프레시

상세			처리 시간과 변동 요인
기본 I/O 유닛	기본 I/O 유닛을 리프레시합니다. 유닛마다 먼저 CPU 유닛에서 출력 유닛으로의 출력을 리프레시하고, 이어서 입력 유닛에서 CPU 유닛으로의 입력을 리프레시합니다.		유닛별 I/O 리프레시 시간에 사용 유닛 대수를 곱한 값.
특수 I/O 유닛	CIO 영역에 할당된 워드		유닛별 I/O 리프레시 시간에 사용 유닛 대수를 곱한 값.
	유닛 고유 데이터	예: CompoBus/S 리모트 I/O	
CPU 버스 유닛	CIO 영역 및 DM 영역에 할당된 워드		유닛별 I/O 리프레시 시간에 사용 유닛 대수를 곱한 값.
	유닛 고유 데이터	예: · EtherNet/IP 태그 데이터 링크 리프레시 · Controller Link 데이터 링크 리프레시 · DeviceNet 리모트 I/O 통신 · 프로토콜 매크로의 송수신 데이터 · 이더넷 유닛의 특정 제어 비트에 대한 소켓 서비스	

5. 주변 서비스

다음의 주변 기기 서비스가 지원됩니다.

· 통신 포트(내부 로직 포트) 서비스
래더 프로그램의 통신 명령어나 백그라운드 실행, 간이 백업 등의 기능을 사용할 때 통신 포트 서비스가 실행됩니다.

· 특수 I/O 유닛 또는 CPU 버스 유닛에 대한 이벤트 서비스
PLC 에 유닛이 접속되어 있으면 이벤트 서비스가 실행됩니다. 통신 유닛을 경유해 지원 소프트웨어를 사용하는 경우 등에도 실행됩니다.
특수 I/O 유닛과 CPU 버스 유닛의 이벤트 서비스 실행에는 우선순위가 없습니다.
이벤트 서비스의 실행 순서는 유닛 번호나 유닛 장착 위치와는 관계가 없습니다.

· USB 포트 서비스
USB 포트를 사용 중일 때 USB 포트 서비스가 실행됩니다.

· RS-232C 포트 서비스
RS-232C 포트를 사용 중일 때 RS-232C 포트 서비스가 실행됩니다.

· 파일 액세스 서비스
메모리 카드 또는 EM 파일 메모리에 액세스할 때 파일 액세스 서비스가 실행됩니다.

· 온라인 에디트 서비스
온라인 에디팅을 실행할 때 온라인 에디트 서비스가 실행됩니다.

· 백업 서비스
프로그램이나 파라미터를 변경할 때 백업 서비스가 실행됩니다.

주변 서비스의 실행에는 우선순위가 없습니다. 모든 서비스는 타임 슬라이싱으로 병렬 실행됩니다.

AI 번역본
-For Reference Only-

주변 기기 서비스가 실행되지 않을 때는 처리에 시간이 걸리지 않습니다. 주변 기기 서비스의 최소 · 최대 처리 시간은 다음 표와 같이 PLC 설정의 Set Time to All Events 파라미터에 따라 달라집니다.

		주변 서비스 실행 시간	
		최소	최대
Set Time to All Events 파라미터	**Disable**	0 ms(서비스를 실행하지 않는 경우)	사이클 타임의 10%(사이클 타임의 10% 가 0.1 ms 미만인 경우에는 0.1 ms)
	Enabled		설정 시간(0.1~3,276.7 ms)

위 표의 최대 실행 시간은 Set Time to All Events 파라미터를 무효로 한 상태에서 온라인 에디팅을 하면 1 ms 더 길어집니다.

12-2-3 유닛별 I/O 유닛 리프레시 시간

● 대표적인 기본 I/O 유닛 리프레시 시간

명칭	형식	유닛당 I/O 리프레시 시간	
		CJ2H	CJ2M/CJ3M
8/16점 DC 입력 유닛	CJ1W-ID201/211/212	0.0014 ms	0.0039 ms
32점 DC 입력 유닛	CJ1W-ID231/232/233	0.0023 ms	0.0054 ms
64점 DC 입력 유닛	CJ1W-ID261/262	0.0041 ms	0.0089 ms
8/16점 AC 입력 유닛	CJ1W-IA201/111	0.0014 ms	0.0039 ms
16점 인터럽트 입력 유닛	CJ1W-INT01	0.0014 ms	0.0039 ms
16점 퀵 리스폰스 입력 유닛	CJ1W-IDP01	0.0014 ms	0.0039 ms
8/16점 릴레이 출력 유닛	CJ1W-OC201/211	0.0014 ms	0.0039 ms
8점 트라이액 출력 유닛	CJ1W-OA201	0.0014 ms	0.0039 ms
8/16점 트랜지스터 출력 유닛	CJ1W-OD201/202/203/204/211/212/213	0.0014 ms	0.0039 ms
32점 트랜지스터 출력 유닛	CJ1W-OD231/232/233/234	0.0023 ms	0.0054 ms
64점 트랜지스터 출력 유닛	CJ1W-OD261/262/263	0.0041 ms	0.0089 ms
24 VDC 입력/트랜지스터 출력 유닛 (입력 16점/출력 16점)	CJ1W-MD231/232/233	0.0023 ms	0.0054 ms
24 VDC 입력/트랜지스터 출력 유닛 (입력 32점/출력 32점)	CJ1W-MD261/263	0.0041 ms	0.0089 ms
TTL 입력/TTL 출력 유닛 (입력 16점/출력 16점)	CJ1W-MD563	0.0041 ms	0.0089 ms
B7A 인터페이스 유닛(입력 64점)	CJ1W-B7A14	0.0041 ms	0.0089 ms
B7A 인터페이스 유닛(출력 64점)	CJ1W-B7A04	0.0041 ms	0.0089 ms
B7A 인터페이스 유닛 (입력 32점/출력 32점)	CJ1W-B7A22	0.0041 ms	0.0089 ms
펄스 I/O 모듈	CJ2M-MD211/212	---	0.010 ms

● 대표적인 특수 I/O 유닛 리프레시 시간

명칭	형식	유닛당 I/O 리프레시 시간	
		CJ2H	CJ2M/CJ3M
아날로그 입력 유닛	CJ1W-AD041/081-V1	0.05 ms	0.07 ms
	CJ1W-AD042	0.05 ms	0.07 ms
아날로그 출력 유닛	CJ1W-DA021/041/08V/08C	0.05 ms	0.07 ms
	CJ1W-DA042V	0.05 ms	0.07 ms
아날로그 I/O 유닛	CJ1W-MAD42	0.05 ms	0.07 ms
온도 조절 유닛	CJ1W-TC□□□	0.05 ms	0.25 ms

AI 번역중
- For Reference Only -

명칭	형식		유닛당 I/O 리프레시 시간	
			CJ2H	CJ2M/CJ3M
위치 제어 유닛	CJ1W-NC113/133		0.12 ms	0.13 ms
			+ 데이터 전송에 사용하는 명령어(IOWR/IORD) 1 개당 0.8 ms.	
	CJ1W-NC213/233		0.14 ms	0.15 ms
			+ 데이터 전송에 사용하는 명령어(IOWR/IORD) 1 개당 0.8 ms.	
	CJ1W-NC413/433		0.17 ms	0.18 ms
			+ 데이터 전송에 사용하는 명령어(IOWR/ IORD) 1 개당 0.7 ms.	
	CJ1W-NC214/234 (2 축 2 태스크)		0.20 ms	0.355 ms
			+ 데이터 전송에 사용하는 명령어(IOWR/IORD) 1 개당 0.1 ms.	
	CJ1W-NC414/434 (4 축 4 태스크)		0.30 ms	0.511 ms
			+ 데이터 전송에 사용하는 명령어(IOWR/IORD) 1 개당 0.1 ms.	
ID 센서 유닛	CJ1W-V600C11		0.12 ms	0.13 ms
	CJ1W-V600C12		0.14 ms	0.15 ms
고속 카운터 유닛	CJ1W-CT021		0.10 ms	0.11 ms
CompoNet 마스터 유닛	CJ1W-CRM21	통신 모드 No. 0	0.13 ms	0.14 ms
		통신 모드 No. 1	0.14 ms	0.15 ms
		통신 모드 No. 2	0.17 ms	0.17 ms
		통신 모드 No. 3	0.21 ms	0.22 ms
		통신 모드 No. 8 (*)	0.081 + (0.0012 × 할당 워드 수) ms	0.083 + (0.009 × 할당 워드 수) ms
CompoBus/S 마스터 유닛	CJ1W-SRM21	유닛 번호 1 개 할당	0.08 ms	0.08 ms
		유닛 번호 2 개 할당	0.08 ms	0.10 ms
프로세스 I/O 유닛	CJ1W-PH41U		0.12 ms	0.13 ms

* 할당 워드 수는 접속된 슬레이브에 실제로 할당되어 있는 I/O 메모리의 실제 워드 수입니다.

AI 번역본
-For Reference Only-

● CPU 버스 유닛에 의한 사이클 타임 증가

명칭	형식	증가량		비고
		CJ2H	CJ2M/CJ3M	
Controller Link 유닛	CJ1W-CLK23	0.1 ms	0.2 ms	데이터 링크를 사용하는 경우 다음 시간이 추가로 필요합니다. ・CJ2H CPU 유닛: 0.1 ms + 0.7 μs × 데이터 링크 워드 수 ・CJ2M/CJ3M CPU 유닛: 0.1 ms + 1.2 μs × 데이터 링크 워드 수 메시지 서비스를 사용하는 경우 이벤트 실행 시간이 추가로 증가합니다.
시리얼 통신 유닛	CJ1W-SCU41-V1 CJ1W-SCU21-V1 CJ1W-SCU31-V1 CJ1W-SCU42 CJ1W-SCU32 CJ1W-SCU22	0.20 ms		프로토콜 매크로를 실행하는 경우 최대 다음 시간만큼 증가합니다: ・CJ2H CPU 유닛: 0.4 μs × 송신 또는 수신하는 최대 데이터 워드 수(0~500 워드) ・CJ2M/CJ3M CPU 유닛: 0.6 μs × 송신 또는 수신하는 최대 데이터 워드 수(0 워드 이상) 호스트 링크 또는 1:N NT 링크를 사용하는 경우 이벤트 실행 시간이 증가합니다.
이더넷 유닛	CJ1W-ENT11/21	0.1 ms	0.9 ms	소프트웨어 스위치로 소켓 서비스를 실행하는 경우 최대 다음 시간만큼 증가합니다 ・CJ2H CPU 유닛: 1.4 μs × 송수신 바이트 수 ・CJ2M/CJ3M CPU 유닛: 0.17 μs × 송수신 바이트 수 FINS 통신 서비스, CMND 명령어에 의한 소켓 서비스 또는 FTP 서비스를 실행하는 경우 이벤트 실행 시간이 증가합니다.
EtherNet/IP 유닛	CJ1W-EIP21/EIP21S	CPU 랙에 장착: 0.1 ms 증설 랙에 장착: 0.13 ms	CPU 랙에 장착: 0.14 ms 증설 랙에 장착: 0.16 ms	태그 데이터 링크를 사용하는 경우 다음 값을 더합니다. CJ2H CPU 유닛 0.1 ms + 전송 워드 수 × 0.33 μs (CPU 랙) 0.1 ms + 전송 워드 수 × 0.45 μs (증설 랙)* CJ2M/CJ3M CPU 유닛 0.02 ms + 전송 워드 수 × 0.78 μs (CPU 랙) 0.02 ms + 전송 워드 수 × 0.92 μs (증설 랙)* 메시지 서비스를 사용하는 경우 이벤트 실행 시간도 더해야 합니다.
FL-net 유닛	CJ1W-FLN22	0.25 ms + 데이터 링크 워드 수 × 1.6 μs	0.25 ms + 데이터 링크 워드 수 × 1.7 μs	데이터 링크 워드 수는 1 개 노드가 송수신하는 데이터의 워드 수입니다. 메시지 서비스를 사용하는 경우 이벤트 실행 시간을 별도로 더해야 합니다.
DeviceNet 유닛	CJ1W-DRM21	0.4 ms + 할당 워드 1 개당 0.7 μs	0.4 ms + 할당 워드 1 개당 0.4 μs	사용하지 않는 것을 포함해 슬레이브에 할당된 모든 워드를 포함합니다. 메시지 통신의 경우 왼쪽과 같이 통신 워드 수를 계산에 더합니다.
아날로그 입력 유닛	CJ1W-ADG41	0.12 ms	0.15 ms	---

AI 번역본
-For Reference Only-

명칭	형식	증가량		비고
		CJ2H	CJ2M/CJ3M	
EtherCAT 통신 기능이 있는 위치 제어 유닛	CJ1W-NC881 8 축 4 태스크 사용 시	0.3 ms	0.5 ms	---
	CJ1W-NC481 4 축 4 태스크 사용 시	0.3 ms	0.4 ms	
	CJ1W-NC281 2 축 2 태스크 사용 시	0.2 ms	0.3 ms	
	CJ1W-NCF81 16 축 4 태스크 사용 시	0.5 ms	0.7 ms	
	CJ1W-NC482 4 축 4 태스크 사용 시	0.3 ms	0.4 ms	리모트 입력: 20 워드, 리모트 출력: 20 워드
		0.7 ms	1.0 ms	리모트 입력: 640 워드, 리모트 출력: 640 워드
	CJ1W-NC882 8 축 4 태스크 사용 시	0.3 ms	0.5 ms	리모트 입력: 20 워드, 리모트 출력: 20 워드
		0.8 ms	1.1 ms	리모트 입력: 640 워드, 리모트 출력: 640 워드
MECHA-TROLINK-II 통신 기능이 있는 위치 제어 유닛	CJ1W-NCF71(-MA) CJ1W-NC471 CJ1W-NC271	유닛에 접속된 서보 드라이버의 축 수에 따릅니다. 1 축: 0.2 ms, 3 축: 0.3 ms, 6 축: 0.4 ms, 16 축: 0.8 ms		---
MECHA-TROLINK-II 통신 기능이 있는 모션 컨트롤 유닛	CJ1W-MCH71	0.2 + (모션 태스크 수× 4 + 축 수 × 5 + 범용 I/O 할당 워드 수) × 0.001 (ms)	0.2 + (모션 태스크 수 × 6 + 축 수 × 1 + 범용 I/O 할당 워드 수) × 0.001 (ms)	---
SYSMAC SPU 유닛	CJ1W-SPU01-V2	0.2 ms + 샘플링 워드 수 × 0.8 µs	0.2 ms + 샘플링 워드 수 × 1.1 ms	---
EtherCAT 슬레이브 유닛	CJ1W-ETC21	0.3 ms + I/O 데이터 영역 사이즈 × 0.0024 ms		I/O 데이터 영역 사이즈 = 입력 데이터 영역 사이즈 + 출력 데이터 영역 사이즈

* 유닛 버전 1.1 이후의 CJ2H CPU 유닛에서 고속 인터럽트를 사용하는 경우 다음 값을 더해야 합니다.

0.1 ms + 전송 워드 수 × 0.87 µs

AI 번역본
-For Reference Only-

12-2-4 사이클 타임 계산 예

● 예 1: 기본 명령어와 기본 I/O 유닛을 사용하는 애플리케이션

다음 예는 PLC 에 기본 I/O 유닛만 접속되어 있고 프로그램이 기본 명령어와 데이터 전송 명령어 20K 스텝으로 구성되어 있는 경우의 사이클 타임 계산 방법을 나타냅니다. 여기서는 CJ2H-CPU6□-EIP CPU 유닛을 사용합니다.

조건

항목	상세	
CPU 유닛	CJ2H-CPU6□-EIP	
CJ 시리즈 CPU 랙	CJ1W-ID261 64점 입력 유닛	2 대
	CJ1W-OD261 64점 출력 유닛	2 대
사용자 프로그램	20 Ksteps	LD 명령어: 10 Ksteps MOV 명령어: 10 Ksteps (LD: 명령어 1 개당 1 스텝. MOV: 명령어 1 개당 3 스텝.)
주변 USB 포트 접속	접속 시와 미접속 시	
고정 사이클 타임 처리	없음	
시리얼 포트 접속	없음	
기타 기기와의 주변 서비스 (특수 I/O 유닛, CPU 버스 유닛, 파일 액세스)	없음	

계산 예

처리 명칭	계산식	처리 시간	
		프로그래밍 디바이스를 주변 USB 포트에 접속한 경우	프로그래밍 디바이스를 주변 USB 포트에 접속하지 않은 경우
(1) 공통 처리	---	0.20 ms	0.20 ms
(2) 프로그램 실행	0.016 μs × 10,000 + 0.14 μs/3 스텝 × 10,000	0.63 ms	0.63 ms
(3) 사이클 타임 계산 (최소 사이클 타임용)	(고정 사이클 타임 미설정)	0 ms	0 ms
(4) I/O 리프레시	0.0039 ms × 2 + 0.0039 ms × 2	0.0164 ms	0.0164 ms
(5) 주변 서비스	(주변 USB 포트 접속 시에만)	0.1 ms	0 ms
사이클 타임	(1) + (2) + (3) + (4) + (5)	0.95 ms	0.85 ms

AI 번역본
-For Reference Only-

● 예 2: 연산과 특수 I/O 유닛을 포함하는 애플리케이션

다음 예는 PLC 에 기본 I/O 유닛과 특수 I/O 유닛이 접속되어 있고 프로그램이 기본 명령어, 데이터 전송 명령어, 부동소수점 연산 명령어 20K 스텝으로 구성되어 있는 경우의 사이클 타임 계산 방법을 나타냅니다. 여기서는 CJ2H-CPU6□-EIP CPU 유닛을 사용합니다.

조건

항목	상세	
CPU 유닛	CJ2H-CPU6□-EIP	
CJ 시리즈 CPU 랙	CJ1W-ID261 64점 입력 유닛	2 대
	CJ1W-OD261 64점 출력 유닛	2 대
	CJ1W-AD081 아날로그 입력 유닛	2 대
	CJ1W-NC413 위치 제어 유닛	2 대
사용자 프로그램	20 Ksteps	LD 명령어: 12 Ksteps MOV 명령어: 6 Ksteps +F 명령어: 2K 스텝 (LD: 명령어 1 개당 1 스텝. MOV: 명령어 1 개당 3 스텝. +F: 명령어 1 개당 4 스텝)
주변 USB 포트 접속	접속 시와 미접속 시	
고정 사이클 타임 처리	없음	
시리얼 포트 접속	없음	
기타 기기와의 주변 서비스 (특수 I/O 유닛, CPU 버스 유닛, 파일 액세스)	없음	

계산 예

처리 명칭	계산식	처리 시간	
		프로그래밍 디바이스를 내장 USB 포트에 접속한 경우	프로그래밍 디바이스를 내장 USB 포트에 접속하지 않은 경우
(1) 공통 처리	---	0.20 ms	0.20 ms
(2) 프로그램 실행	0.016 ms × 12,000 + 0.05 µs × 6,000 + 0.24 µs × 2,000	0.972 ms	0.972 ms
(3) 사이클 타임 계산 (최소 사이클 타임용)	(고정 사이클 타임 미설정)	0 ms	0 ms
(4) I/O 리프레시	0.0041 ms × 2 + 0.0041 ms × 2 + 0.05 ms × 2 + 0.17 ms × 2	0.4564 ms	0.4564 ms
(5) 주변 서비스	(주변 USB 포트 접속 시에만)	0.1 ms	0 ms
사이클 타임	(1) + (2) + (3) + (4) + (5)	1.73 ms	1.73 ms

AI 번역본 For Reference Only

12-2-5 온라인 에디팅에 의한 사이클 타임 연장

CPU 유닛이 MONITOR 모드일 때 CX-Programmer 에서 온라인 에디팅을 실행할 수 있습니다.
CPU 유닛 프로그램의 쓰기 처리로 인해 사이클 타임이 약 1 ms 연장됩니다. 사이클 타임이 특정 값만큼 연장되는 것을 피하려면 온라인 에디팅 금지(스탠바이) 기능을 사용하십시오. 자세한 내용은 10-7-4 온라인 에디팅을 참조하십시오.

12-2-6 I/O 응답 시간

I/O 응답 시간이란 입력 유닛의 입력이 ON 되고, 그 데이터가 CJ 시리즈 CPU 유닛에 인식되어 사용자 프로그램이 실행된 후, 그 결과가 출력 유닛의 출력 단자로 출력되기까지 걸리는 시간입니다.

I/O 응답 시간의 길이는 다음 조건에 따라 달라집니다.

- 입력 비트가 ON 되는 타이밍.
- 사이클 타임.
- 입력 유닛과 출력 유닛이 장착된 랙의 종류(CPU 랙 또는 증설 랙).

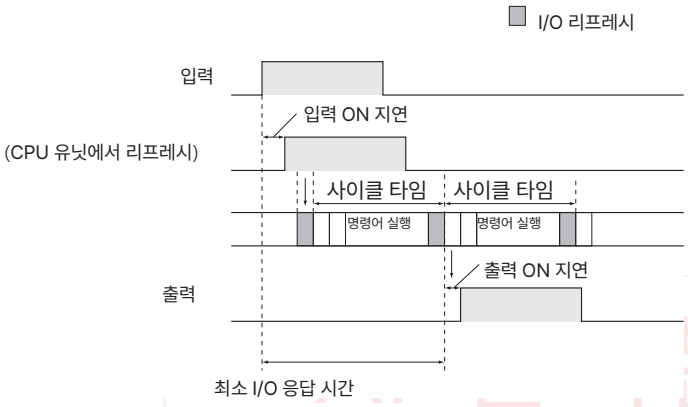
CJ 시리즈 기본 I/O 유닛의 응답 시간

● 최소 I/O 응답 시간

CPU 유닛의 I/O 리프레시 직전에 데이터를 가져올 때 I/O 응답 시간이 가장 짧아집니다.

최소 I/O 응답 시간은 입력 ON 지연, 사이클 타임, 출력 ON 지연의 합계입니다.

주 입력 및 출력의 ON 지연은 사용하는 유닛에 따라 다릅니다.

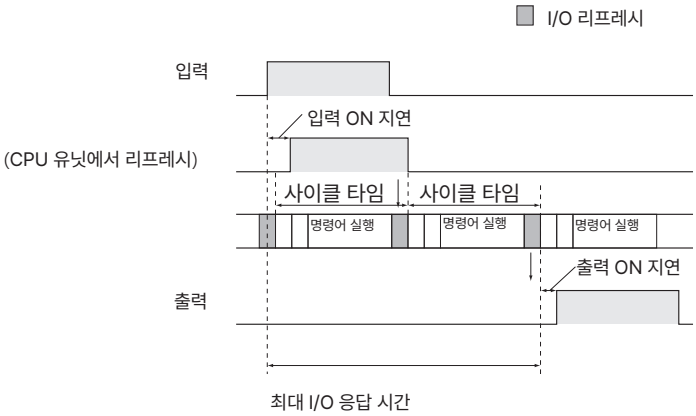


-For Reference Only-

● 최대 I/O 응답 시간

입력 유닛의 I/O 리프레시 직후에 데이터를 가져올 때 I/O 응답 시간이 가장 길어집니다.

최대 I/O 응답 시간은 입력 ON 지연, (사이클 타임 × 2), 출력 ON 지연의 합계입니다.



● 계산 예

조건:

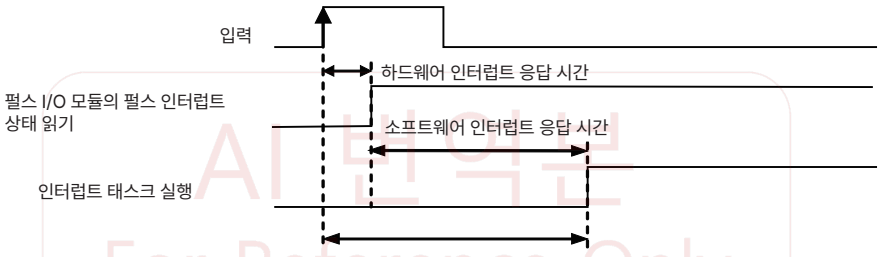
입력 ON 지연: 1.5 ms
출력 ON 지연: 0.2 ms
사이클 타임: 20.0 ms

최소 I/O 응답 시간 = 1.5 ms + 20 ms + 0.2 ms = 21.7 ms

최대 I/O 응답 시간 = 1.5 ms + (20 ms × 2) + 0.2 ms = 41.7 ms

12-2-7 펄스 I/O 모듈로의 입력 인터럽트 응답 시간

입력 인터럽트 태스크의 응답 시간이란 CJ2M/CJ3M CPU 유닛에 접속된 펄스 I/O 모듈로의 펄스 입력이 ON 또는 OFF 된 후 I/O 인터럽트 태스크가 실행되기까지의 시간입니다. 입력 인터럽트 태스크의 인터럽트 응답 시간의 길이는 하드웨어 인터럽트 응답 시간과 소프트웨어 인터럽트 응답 시간의 합계에 따라 달라집니다.



AI 번역본
-For Reference Only-

펄스 I/O 모듈로의 입력 인터럽트 응답 시간 = 하드웨어 인터럽트 응답 시간 + 소프트웨어 인터럽트 응답 시간

하드웨어 인터럽트 응답 시간

상승/하강 미분	인터럽트 응답 시간
상승	30 μs
하강	150 μs

소프트웨어 인터럽트 응답 시간

인터럽트 종별	인터럽트 응답 시간
다이렉트 모드의 입력 인터럽트	최소: 33 μs
카운터 모드의 입력 인터럽트	최소: 34 μs

12-2-8 시리얼 PLC 링크의 응답 성능

시리얼 PLC 링크로 접속된 CPU 유닛 간(폴링 유닛에서 폴드 유닛으로, 또는 폴드 유닛에서 폴링 유닛으로)의 응답 시간은 다음과 같이 계산할 수 있습니다.

이 계산은 주변 서비스 시간이 필요한 통신 시간을 처리하기에 충분히 길다는 것을 전제로 합니다.

- 최대 I/O 응답 시간(하드웨어 지연 제외) [ms] =
  폴링 유닛 사이클 타임 × 2 + 통신 사이클 타임 + 폴드 유닛 사이클 타임 × 2 + 폴드 유닛 통신 시간 + 4 ms
- 최소 I/O 응답 시간(하드웨어 지연 제외) [ms] =
  폴드 유닛 통신 시간 + 0.54 ms

참가 중인 폴드 유닛 노드 수	폴링 유닛에 설정된 최대 유닛 번호 범위 내에서 링크가 확립된 폴드 유닛의 수.
미참가 폴드 유닛 노드 수	폴링 유닛에 설정된 최대 유닛 번호 범위 내에서 링크에 참가하지 않은 폴드 유닛의 수.
통신 사이클 타임 [ms]	· 폴드 유닛이 2 대 이상인 경우 폴드 유닛 통신 시간 × 참가 중인 폴드 유닛 노드 수 + 10 × 미참가 폴드 유닛 노드 수 · 폴드 유닛이 1 대인 경우 폴드 유닛 통신 시간 + 10 ms
폴드 유닛 통신 시간 [ms]	· 보 레이트를 Standard 로 설정한 경우: 폴링 유닛 사이클 타임 + 0.286 × [(폴드 유닛 수 + 1) × 링크 워드 수 × 2 + 12] · 보 레이트를 Fast 로 설정한 경우: 폴링 유닛 사이클 타임 + 0.0955 × [(폴드 유닛 수 + 1) × 링크 워드 수 × 2 + 12]

AI 번역본
-For Reference Only-

AI 번역본
-For Reference Only-

App

부록

부록에서는 정전 시의 CPU 유닛 동작, CPU 유닛이 지원하는 명령어, 보조 기억 영역의 워드와 비트, 명령어 실행 시간, 연속된 PLC 메모리 어드레스의 메모리 맵, CJ 시리즈 PLC 와 CS 시리즈 PLC 의 비교에 대해 설명합니다.

App

A-1 명령어 기능 A-3

A-1-1 시퀀스 입력 명령어 A-3
A-1-2 시퀀스 출력 명령어 A-5
A-1-3 시퀀스 제어 명령어 A-6
A-1-4 타이머 · 카운터 명령어 A-10
A-1-5 비교 명령어 A-14
A-1-6 데이터 전송 명령어 A-18
A-1-7 데이터 시프트 명령어 A-20
A-1-8 증가/감소 명령어 A-24
A-1-9 심볼 연산 명령어 A-24
A-1-10 변환 명령어 A-29
A-1-11 논리 명령어 A-35
A-1-12 특수 연산 명령어 A-37
A-1-13 부동소수점 연산 명령어 A-38
A-1-14 배정밀도 부동소수점 명령어 A-42
A-1-15 테이블 데이터 처리 명령어 A-45
A-1-16 트래킹 명령어 A-49
A-1-17 데이터 제어 명령어 A-50
A-1-18 서브루틴 명령어 A-54
A-1-19 인터럽트 제어 명령어 A-55
A-1-20 고속 카운터/펄스 출력 명령어 A-56
A-1-21 스텝 명령어 A-58
A-1-22 기본 I/O 유닛 명령어 A-58
A-1-23 시리얼 통신 명령어 A-61
A-1-24 네트워크 명령어 A-63
A-1-25 파일 메모리 명령어 A-65
A-1-26 표시 명령어 A-66
A-1-27 시계 명령어 A-67
A-1-28 디버깅 명령어 A-68
A-1-29 고장 진단 명령어 A-68
A-1-30 기타 명령어 A-69
A-1-31 블록 프로그래밍 명령어 A-70
A-1-32 문자열 처리 명령어 A-74
A-1-33 태스크 제어 명령어 A-77

사용 번역본
-For Reference Only-

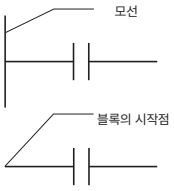
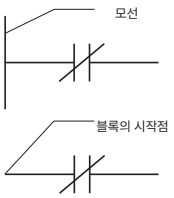
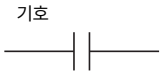
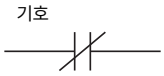
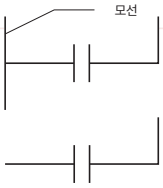
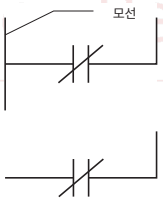
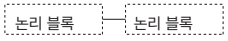
A-1-34 기종 변환 명령어 A-77
A-1-35 특수 펑션 블록 명령어 A-78
A-2 명령어 실행 시간과 스텝 수 A-80
A-2-1 시퀀스 입력 명령어 A-81
A-2-2 시퀀스 출력 명령어 A-82
A-2-3 시퀀스 제어 명령어 A-82
A-2-4 타이머 · 카운터 명령어 A-83
A-2-5 비교 명령어 A-84
A-2-6 데이터 전송 명령어 A-85
A-2-7 데이터 시프트 명령어 A-86
A-2-8 증가/감소 명령어 A-87
A-2-9 심볼 연산 명령어 A-87
A-2-10 변환 명령어 A-89
A-2-11 논리 명령어 A-91
A-2-12 특수 연산 명령어 A-91
A-2-13 부동소수점 연산 명령어 A-92
A-2-14 배정밀도 부동소수점 명령어 A-93
A-2-15 테이블 데이터 처리 명령어 A-94
A-2-16 트래킹 명령어 A-96
A-2-17 데이터 제어 명령어 A-96
A-2-18 서브루틴 명령어 A-97
A-2-19 인터럽트 제어 명령어 A-98
A-2-20 고속 카운터/펄스 출력 명령어 A-98
A-2-21 스텝 명령어 A-101
A-2-22 기본 I/O 유닛 명령어 A-101
A-2-23 시리얼 통신 명령어 A-102
A-2-24 네트워크 명령어 A-103
A-2-25 파일 메모리 명령어 A-103
A-2-26 표시 명령어 A-104
A-2-27 시계 명령어 A-104
A-2-28 디버깅 명령어 A-104
A-2-29 고장 진단 명령어 A-104
A-2-30 기타 명령어 A-105
A-2-31 블록 프로그래밍 명령어 A-105
A-2-32 문자열 처리 명령어 A-107
A-2-33 태스크 제어 명령어 A-108
A-2-34 기종 변환 명령어 A-108
A-2-35 특수 펑션 블록 명령어 A-109
A-2-36 SFC 명령어 A-109
A-2-37 펑션 블록 인스턴스 실행 시간 A-109
A-3 보조 기억 영역 A-111
A-3-1 읽기 전용 영역(시스템 설정) A-111
A-3-2 읽기/쓰기 영역(사용자 설정) A-153
A-3-3 보조 기억 영역 동작의 상세 A-164
A-4 PLC 메모리 어드레스의 메모리 맵 A-167
A-4-1 PLC 메모리 어드레스 A-167
A-4-2 메모리 맵 A-168
A-5 정전 시의 동작 A-169
A-5-1 전원 OFF 시의 동작 A-169
A-5-2 정전 시의 명령어 실행 A-171
A-6 Windows 7 이후 Windows OS 에서의 EtherNet/IP 접속 A-174
A-6-1 Windows 방화벽 설정 변경 A-174
A-7 유닛 버전별 지원 기능 A-176
A-7-1 CJ2H CPU 유닛 A-176
A-7-2 CJ2M CPU 유닛 A-179

참고용 - For Reference Only -

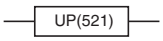
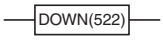
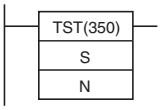
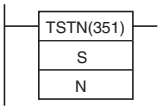
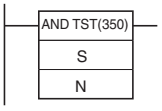
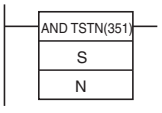
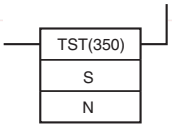
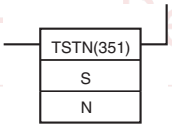
A-1 명령어 기능

CJ2/CJ3 CPU 유닛은 다음 명령어를 지원합니다.
자세한 내용은 CS/CJ/NSJ 시리즈 명령어 레퍼런스 매뉴얼 (Cat. No. W474)을 참조하십시오.

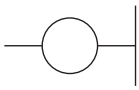
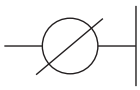
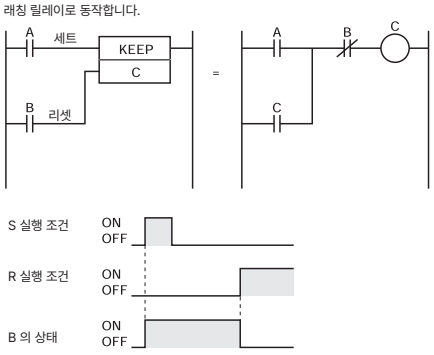
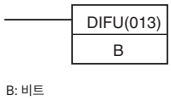
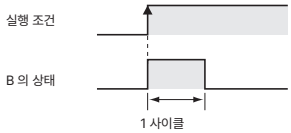
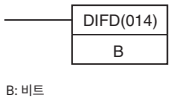
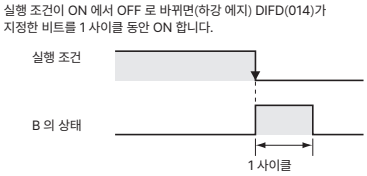
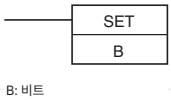
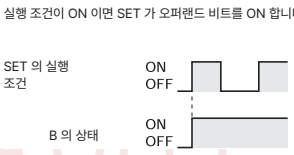
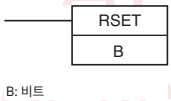
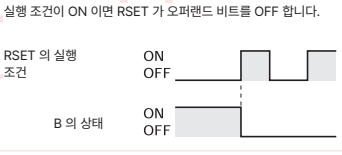
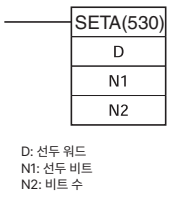
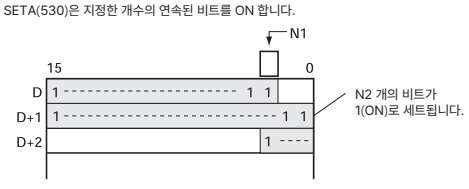
A-1-1 시퀀스 입력 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
로드	LD @LD %LD !LD !@LD !%LD	---		논리 시작을 나타내며, 지정한 오퍼랜드 비트의 ON/OFF 상태에 따라 ON/OFF 실행 조건을 만듭니다.	논리 시작	필요 없음
로드 NOT	LD NOT @LD NOT*2 %LD NOT !LD NOT !@LD NOT !%LD NOT	---		논리 시작을 나타내며, 지정한 오퍼랜드 비트의 ON/OFF 상태를 반전한 값에 따라 ON/OFF 실행 조건을 만듭니다.	논리 시작	필요 없음
AND	AND @AND %AND !AND !@AND !%AND	---		지정한 오퍼랜드 비트의 상태와 현재 실행 조건의 논리곱(AND)을 취합니다.	렁에 연속	필요
AND NOT	AND NOT @AND NOT %AND NOT !AND NOT !@AND NOT !%AND NOT	---		지정한 오퍼랜드 비트의 상태를 반전하고, 현재 실행 조건과의 논리곱(AND)을 취합니다.	렁에 연속	필요
OR	OR @OR %OR !OR !@OR !%OR	---		지정한 오퍼랜드 비트의 ON/OFF 상태와 현재 실행 조건의 논리합(OR)을 취합니다.	렁에 연속	필요
OR NOT	OR NOT @OR NOT %OR NOT !OR NOT !@OR NOT !%OR NOT	---		지정한 비트의 상태를 반전하고, 현재 실행 조건과의 논리합(OR)을 취합니다	렁에 연속	필요
AND 로드	AND LD	---		논리 블록 간의 논리곱(AND)을 취합니다. LD ~ 논리 블록 A LD ~ 논리 블록 B AND LD ····· 논리 블록 A 와 논리 블록 B 의 직렬 접속.	렁에 연속	필요

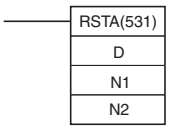
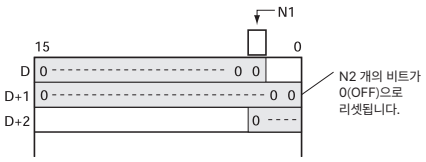
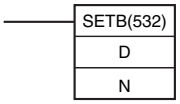
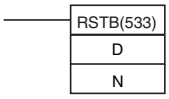
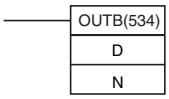
번역본
-For Reference Only-

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
OR 로드	OR LD	---		논리 블록 간의 논리합(OR)을 취합니다. LD ~ 논리 블록 A LD ~ 논리 블록 B OR LD ····· 논리 블록 A 와 논리 블록 B 의 병렬 접속.	렁에 연속	필요
NOT	NOT	520		실행 조건을 반전합니다.	렁에 연속	필요
조건 ON	UP	521		실행 조건이 OFF 에서 ON 으로 바뀌면 UP(521)이 1 사이클 동안 실행 조건을 ON 합니다.	렁에 연속	필요
조건 OFF	DOWN	522		실행 조건이 ON 에서 OFF 로 바뀌면 DOWN(522)이 1 사이클 동안 실행 조건을 ON 합니다.	렁에 연속	필요
비트 테스트	LD TST	350	 S: 소스 워드 N: 비트 번호	LD TST(350), AND TST(350), OR TST(350)은 프로그램에서 LD, AND, OR 와 같이 사용합니다. 지정한 워드 내의 지정한 비트가 ON 이면 실행 조건이 ON 이 되고, 그 비트가 OFF 이면 OFF 가 됩니다.	논리 시작	필요 없음
비트 테스트	LD TSTN	351	 S: 소스 워드 N: 비트 번호	LD TSTN(351), AND TSTN(351), OR TSTN(351)은 프로그램에서 LD NOT, AND NOT, OR NOT 와 같이 사용합니다. 지정한 워드 내의 지정한 비트가 ON 이면 실행 조건이 OFF 가 되고, 그 비트가 OFF 이면 ON 이 됩니다.	논리 시작	필요 없음
비트 테스트	AND TST	350	 S: 소스 워드 N: 비트 번호	LD TST(350), AND TST(350), OR TST(350)은 프로그램에서 LD, AND, OR 와 같이 사용합니다. 지정한 워드 내의 지정한 비트가 ON 이면 실행 조건이 ON 이 되고, 그 비트가 OFF 이면 OFF 가 됩니다.	렁에 연속	필요
비트 테스트	AND TSTN	351	 S: 소스 워드 N: 비트 번호	LD TSTN(351), AND TSTN(351), OR TSTN(351)은 프로그램에서 LD NOT, AND NOT, OR NOT 와 같이 사용합니다. 지정한 워드 내의 지정한 비트가 ON 이면 실행 조건이 OFF 가 되고, 그 비트가 OFF 이면 ON 이 됩니다.	렁에 연속	필요
비트 테스트	OR TST	350	 S: 소스 워드 N: 비트 번호	LD TST(350), AND TST(350), OR TST(350)은 프로그램에서 LD, AND, OR 와 같이 사용합니다. 지정한 워드 내의 지정한 비트가 ON 이면 실행 조건이 ON 이 되고, 그 비트가 OFF 이면 OFF 가 됩니다.	렁에 연속	필요
비트 테스트	OR TSTN	351	 S: 소스 워드 N: 비트 번호	LD TSTN(351), AND TSTN(351), OR TSTN(351)은 프로그램에서 LD NOT, AND NOT, OR NOT 와 같이 사용합니다. 지정한 워드 내의 지정한 비트가 ON 이면 실행 조건이 OFF 가 되고, 그 비트가 OFF 이면 ON 이 됩니다.	렁에 연속	필요

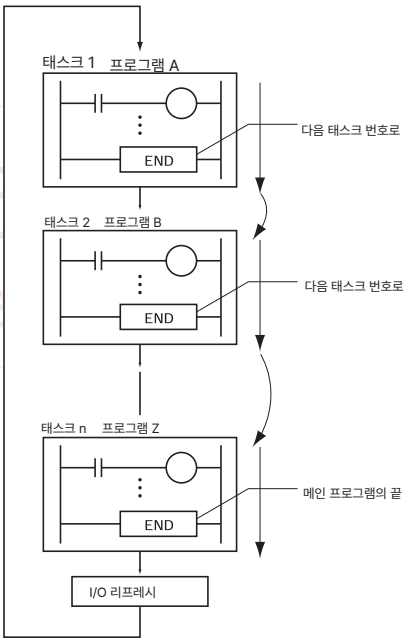
A-1-2 시퀀스 출력 명령어

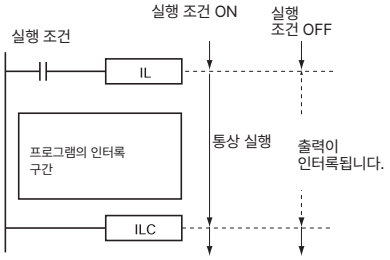
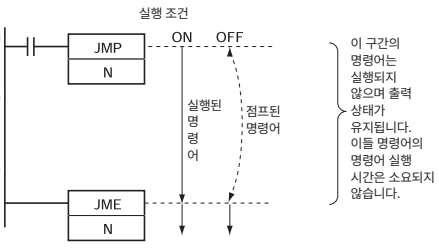
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
출력	OUT !OUT	---		논리 처리의 결과(실행 조건)를 지정한 비트로 출력합니다.	출력	필요
출력 NOT	OUT NOT !OUT NOT	---		논리 처리의 결과(실행 조건)를 반전해 지정한 비트로 출력합니다.	출력	필요
KEEP	KEEP !KEEP	011			출력	필요
상승 미분	DIFU !DIFU	013			출력	필요
하강 미분	DIFD !DIFD	014			출력	필요
SET	SET @SET %SET !SET !@SET !%SET	---			출력	필요
리셋	RSET @RSET %RSET !RSET !@RSET !%RSET	---			출력	필요
복수 비트 세트	SETA @SETA	530			출력	필요

사 본 입 니 다
-For Refe

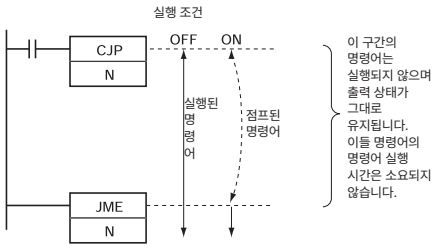
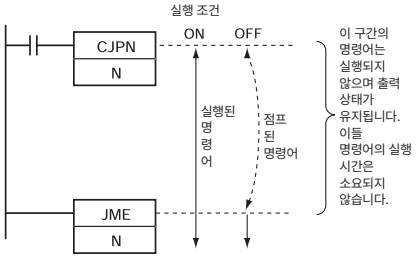
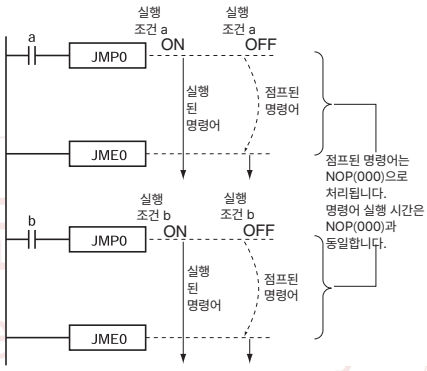
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
복수 비트 리셋	RSTA @RSTA	531	 D: 선두 워드 N1: 선두 비트 N2: 비트 수	RSTA(531)은 지정한 개수의 연속된 비트를 OFF 합니다. 	출력	필요
싱글 비트 세트	SETB @SETB !SETB !@SETB	532	 D: 워드 어드레스 N: 비트 번호	실행 조건이 ON 이면 SETB(532)가 지정한 워드 내의 지정한 비트를 ON 합니다. SET 명령어와 달리 SETB(532)는 DM 또는 EM 워드 내의 비트를 세트할 수 있습니다.	싱글 비트 세트	SETB
싱글 비트 리셋	RSTB @RSTB !RSTB !@RSTB	533	 D: 워드 어드레스 N: 비트 번호	실행 조건이 ON 이면 RSTB(533)가 지정한 워드 내의 지정한 비트를 OFF 합니다. RSET 명령어와 달리 RSTB(533)는 DM 또는 EM 워드 내의 비트를 리셋할 수 있습니다.	출력	필요
싱글 비트 출력	OUTB @OUTB !OUTB	534	 D: 워드 어드레스 N: 비트 번호	OUTB(534)는 논리 처리의 결과(실행 조건)를 지정한 비트로 출력합니다. OUT 명령어와 달리 OUTB(534)는 DM 또는 EM 워드 내의 비트를 제어할 수 있습니다.	출력	필요

A-1-3 시퀀스 제어 명령어

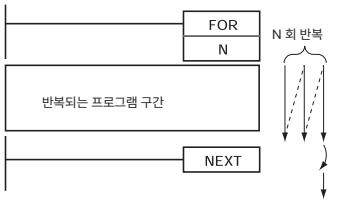
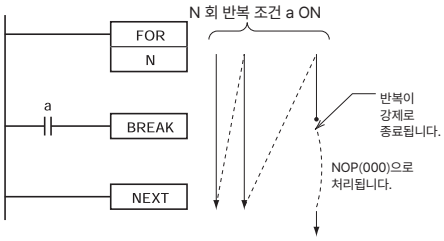
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
END	END	001		프로그램의 끝을 나타냅니다. END(001)은 그 사이클의 프로그램 실행을 완료합니다. END(001) 뒤에 기술된 명령어는 실행되지 않습니다. 실행은 다음 태스크 번호의 프로그램으로 진행됩니다. 실행 중인 프로그램이 프로그램 내에서 가장 큰 태스크 번호를 가진 경우, END(001)은 메인 프로그램 전체의 끝을 나타냅니다. 	출력	필요 없음
노 오퍼레이션	NOP	000	---	이 명령어는 기능이 없습니다. (NOP(000)에서는 아무 처리도 하지 않습니다.)	출력	필요 없음

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
인터록	IL	002	IL(002)	IL(002)의 실행 조건이 OFF 이면 IL(002)과 ILC(003) 사이의 모든 출력을 인터록합니다. IL(002)과 ILC(003)은 보통 쌍으로 사용합니다. 	출력	필요
인터록 클리어	ILC	003	ILC(003)	IL(002)의 실행 조건이 OFF 이면 IL(002)과 ILC(003) 사이의 모든 출력이 인터록됩니다. IL(002)과 ILC(003)은 보통 쌍으로 사용합니다.	출력	필요 없음
복수 인터록 미분 홀드	MILH	517	MILH (517) N D N: 인터록 번호 D: 인터록 상태 비트	MILH(517)의 실행 조건이 OFF 이면, 해당 MILH(517) 명령어와 다음 MILC(519) 명령어 사이의 모든 명령어의 출력이 인터록됩니다. MILH(517)과 MILC(519)는 쌍으로 사용합니다. MILH(517)/MILC(519) 인터록은 네스팅할 수 있습니다(예: MILH(517)-MILH(517)-MILC(519)-MILC(519)). MILH(517)과 대응하는 MILC(519) 사이에 미분 명령어 (DIFU, DIFD 또는 @ · % 접두가 붙은 명령어)가 있는 경우, 그 명령어의 미분 조건이 성립했다면 인터록이 해제된 후 그 명령어가 실행됩니다.	출력	필요
복수 인터록 미분 릴리스	MILR	518	MILR (518) N D N: 인터록 번호 D: 인터록 상태 비트	MILR(518)의 실행 조건이 OFF 이면, 해당 MILR(518) 명령어와 다음 MILC(519) 명령어 사이의 모든 명령어의 출력이 인터록됩니다.MILR(518)과 MILC(519)는 쌍으로 사용합니다. MILR(518)/MILC(519) 인터록은 네스팅할 수 있습니다(예: MILR(518)-MILR(518)-MILC(519)-MILC(519)). MILR(518)과 대응하는 MILC(519) 사이에 미분 명령어 (DIFU, DIFD 또는 @ · % 접두가 붙은 명령어)가 있는 경우, 그 명령어의 미분 조건이 성립했더라도 인터록이 해제된 후 그 명령어는 실행되지 않습니다.	출력	필요
복수 인터록 클리어	MILC	519	MILC (519) N N: 인터록 번호	같은 인터록 번호를 가진 MILH(517) 또는 MILR(518)로 시작된 인터록을 해제합니다. MILH(517)/MILR(518)의 실행 조건이 OFF 이면, MILH(517)/MILR(518)과 같은 인터록 번호를 가진 대응 MILC(519) 사이의 모든 출력이 인터록됩니다.	출력	필요 없음
점프	JMP	004	JMP(004) N N: 점프 번호	JMP(004)의 실행 조건이 OFF 이면, 프로그램 실행은 같은 점프 번호를 가진 프로그램 내 최초의 JME(005)로 직접 점프합니다. JMP(004)와 JME(005)는 쌍으로 사용합니다. 	출력	필요

-For Refer

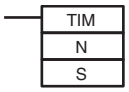
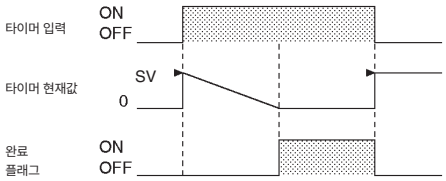
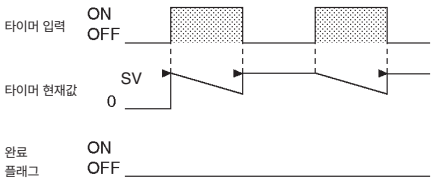
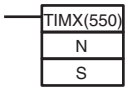
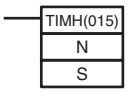
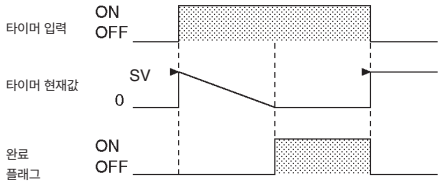
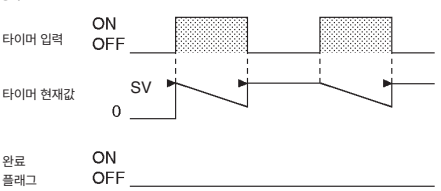
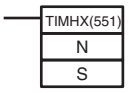
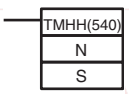
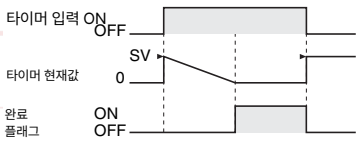
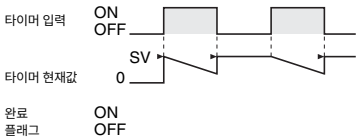
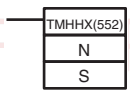
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
조건부 점프	CJP	510	CJP(510) N N: 점프 번호	CJP(510)의 동작은 기본적으로 JMP(004)와 반대입니다. CJP(510)의 실행 조건이 ON 이면, 프로그램 실행은 같은 점프 번호를 가진 프로그램 내 최초의 JME(005)로 직접 점프합니다. CJP(510)과 JME(005)는 쌍으로 사용합니다. 	출력	필요
조건부 점프	CJPN	511	CJPN(511) N N: 점프 번호	CJPN(511)의 동작은 JMP(004)와 거의 동일합니다. CJP(004)의 실행 조건이 OFF 이면, 프로그램 실행은 같은 점프 번호를 가진 프로그램 내 최초의 JME(005)로 직접 점프합니다. CJPN(511)과 JME(005)는 쌍으로 사용합니다. 	출력	필요 없음
점프 엔드	JME	005	JME(005) N N: 점프 번호	점프 명령어의 점프처를 나타냅니다.	출력	필요 없음
복수 점프	JMP0	515	JMP0(515)	JMP0(515)의 실행 조건이 OFF 이면, 프로그램 내 JMP0(515)부터 다음 JME0(516)까지의 모든 명령어가 NOP(000)으로 처리됩니다. JMP0(515)과 JME0(516)은 쌍으로 사용하십시오. 프로그램에서 사용할 수 있는 쌍의 수에는 제한이 없습니다. 	출력	필요
복수 점프 엔드	JME0	516	JME0(516)	JMP0(515)의 실행 조건이 OFF 이면, 프로그램 내 JMP0(515)부터 다음 JME0(516)까지의 모든 명령어가 NOP(000)으로 처리됩니다. JMP0(515)과 JME0(516)은 쌍으로 사용하십시오. 프로그램에서 사용할 수 있는 쌍의 수에는 제한이 없습니다.	출력	필요 없음

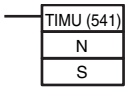
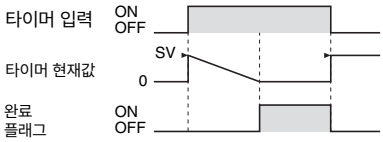
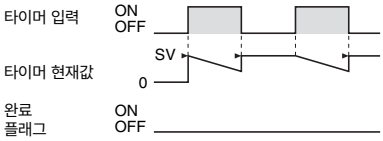
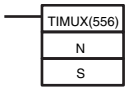
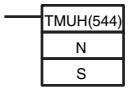
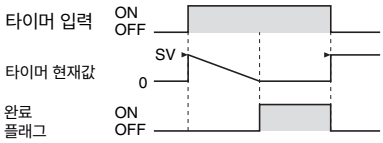
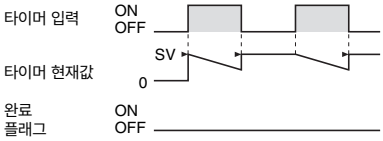
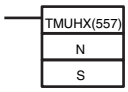
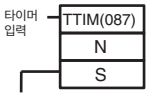
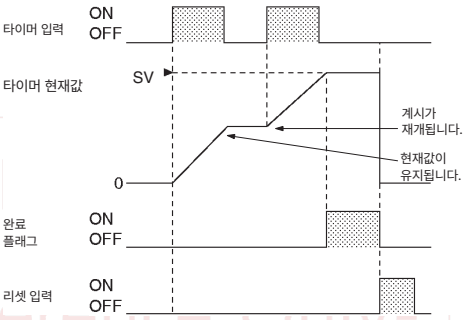
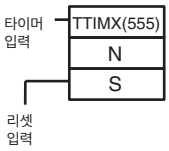
AI
-For Refe

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
FOR-NEXT LOOPS	FOR	512	FOR(512) N N: 반복 횟수	FOR(512)와 NEXT(513) 사이의 명령어를 지정한 횟수만큼 반복합니다. FOR(512)와 NEXT(513)은 쌍으로 사용합니다. 	출력	필요 없음
BREAK LOOP	BREAK	514	BREAK(514)	FOR-NEXT 루프 안에 프로그래밍해, 지정한 실행 조건에서 루프 실행을 취소합니다. 루프 내의 남은 명령어는 NOP(000) 명령어로 처리됩니다. 	출력	필요
FOR-NEXT LOOPS	NEXT	513	NEXT(513)	FOR(512)와 NEXT(513) 사이의 명령어를 지정한 횟수만큼 반복합니다. FOR(512)와 NEXT(513)은 쌍으로 사용합니다.	출력	필요 없음

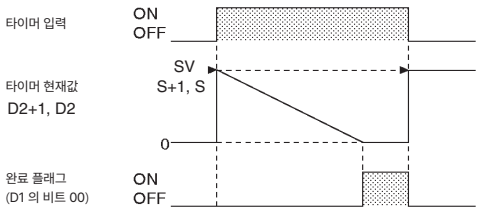
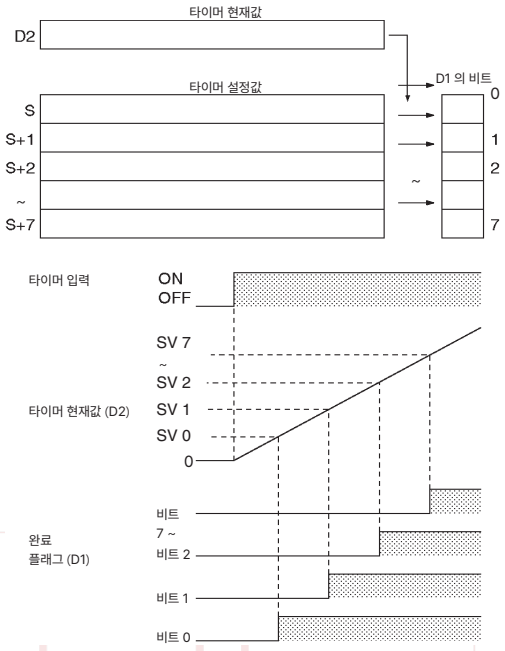
AI 번역본
-For Reference Only-

A-1-4 타이머 · 카운터 명령어

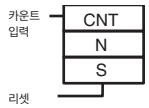
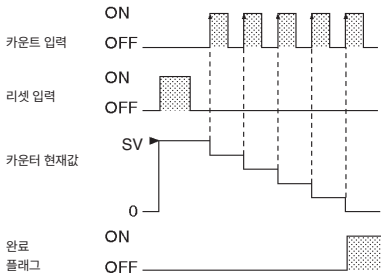
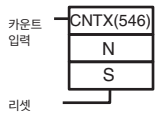
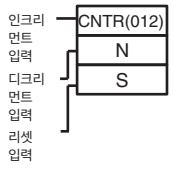
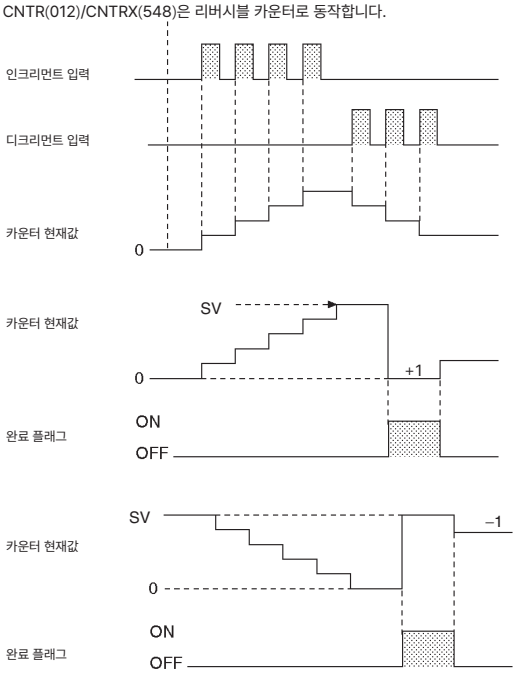
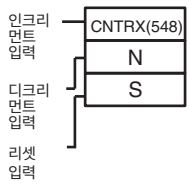
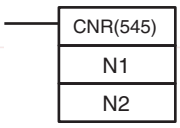
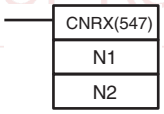
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
100ms 타이머	TIM (BCD)	---	 N: 타이머 번호 S: 설정값	TIM/TIMX(550)은 0.1 s 단위의 감산 타이머로 동작합니다. 설정값(SV)의 설정 범위는 BCD 에서는 0~999.9 s, 2진(10진 또는 16진)에서는 0~6,553.5 s 입니다.  완료 플래그가 ON 되기 전에 타이머 입력이 OFF 된 경우 	출력	필요
	TIMX (Binary)	550	 N: 타이머 번호 S: 설정값			
10ms 타이머	TIMH (BCD)	015	 N: 타이머 번호 S: 설정값	TIMH(015)/TIMHX(551)은 10 ms 단위의 감산 타이머로 동작합니다. 설정값(SV)의 설정 범위는 BCD 에서는 0~99.99 s, 2진(10진 또는 16진)에서는 0~655.35 s 입니다.  완료 플래그가 ON 되기 전에 타이머 입력이 OFF 된 경우 	출력	필요
	TIMHX (Binary)	551	 N: 타이머 번호 S: 설정값			
1ms 타이머	TMHH (BCD)	540	 N: 타이머 번호 S: 설정값	TMHH(540)/TMHHX(552)는 1 ms 단위의 감산 타이머로 동작합니다. 설정값(SV)의 설정 범위는 BCD 에서는 0~9.999 s, 2진(10진 또는 16진)에서는 0~65.535 s 입니다. TMHH(540)과 TMHHX(552)의 타이밍 차트는 위의 TIMH(015)와 동일합니다.  완료 플래그가 ON 되기 전에 타이머 입력이 OFF 된 경우 	출력	필요
	TMHHX (BCD)	552	 N: 타이머 번호 S: 설정값			

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
0.1ms 타이머	TIMU (BCD)	541	 N: 타이머 번호 S: 설정값	TIMU(541)/TIMUX(556)은 0.1 ms 단위의 감산 타이머로 동작합니다. 설정값(SV)의 설정 범위는 BCD 에서는 0~0.9999 s, 2진(10진 또는 16진)에서는 0~6.5535 s 입니다.  완료 플래그가 ON 되기 전에 타이머 입력이 OFF 된 경우 	출력	필요
	TIMUX (BIN)	556	 N: 타이머 번호 S: 설정값			
0.01ms 타이머	TMUH (BCD)	544	 N: 타이머 번호 S: 설정값	TMUH(544)/TMUHX(557)은 0.01 ms 단위의 감산 타이머로 동작합니다. 설정값(SV)의 설정 범위는 BCD 에서는 0~0.09999 s, 2진(10진 또는 16진)에서는 0~0.65535 s 입니다.  완료 플래그가 ON 되기 전에 타이머 입력이 OFF 된 경우 	출력	필요
	TIMUX (BIN)	556	 N: 타이머 번호 S: 설정값			
적산 타이머	TTIM (BCD)	087	 N: 타이머 번호 S: 설정값	TTIM(087)/TTIMX(555)는 0.1 s 단위의 가산 타이머로 동작합니다. 설정값(SV)의 설정 범위는 BCD 에서는 0~999.9 s, 2진(10진 또는 16진)에서는 0~6,553.5 s 입니다. 	출력	필요
	TTIMX (Binary)	555	 N: 타이머 번호 S: 설정값			

-Reference Only-

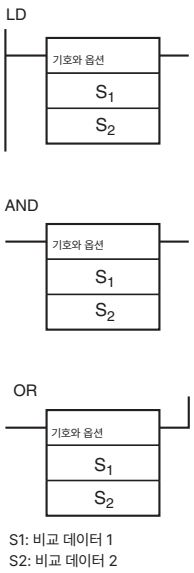
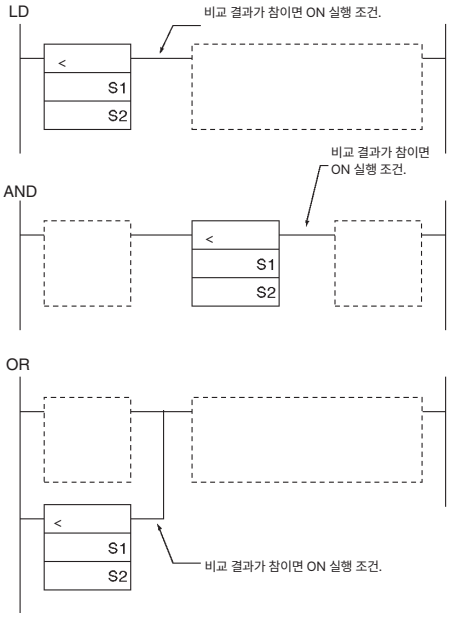
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
롱 타이머	TIML (BCD)	542	TIML(542) D1 D2 S D1: 완료 플래그 D2: 현재값 워드 S: 설정값 워드	TIML(542)/TIMLX(553)은 0.1 s 단위의 감산 타이머로 동작하며, BCD 에서는 약 115 일, 2진(10진 또는 16진)에서는 49,710 일까지 계시할 수 있습니다. 	출력	필요
	TIMLX (Binary)	553	TIMLX(553) D1 D2 S D1: 완료 플래그 D2: 현재값 워드 S: 설정값 워드			
복수 출력 타이머	MTIM (BCD)	543	MTIM(543) D1 D2 S D1: 완료 플래그 D2: 현재값 워드 S: 선두 설정값 워드	MTIM(543)/MTIMX(554)는 8 개의 독립된 설정값과 완료 플래그를 갖는 0.1 s 가산 타이머로 동작합니다. 설정값(SV)의 설정 범위는 BCD 에서는 0~999.9 s, 2진(10진 또는 16진)에서는 0~6,553.5 s 입니다. 	출력	필요
	MTIMX (Binary)	554	MTIMX(554) D1 D2 S D1: 완료 플래그 D2: 현재값 워드 S: 선두 설정값 워드			
타이머 리셋	TRSET	549	TRSET(549) N N: 타이머 번호	지정한 타이머를 리셋합니다.	출력	필요

AI -for Reference Only-

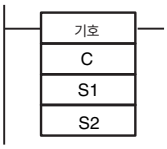
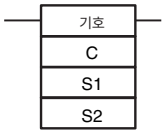
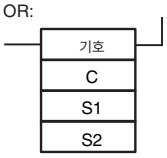
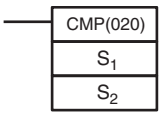
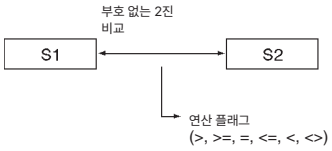
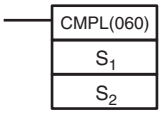
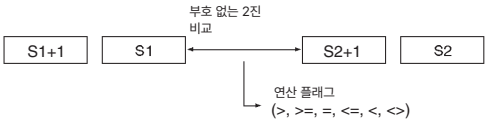
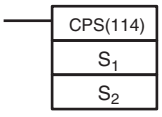
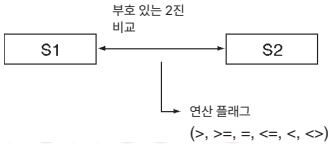
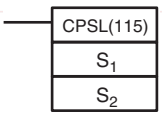
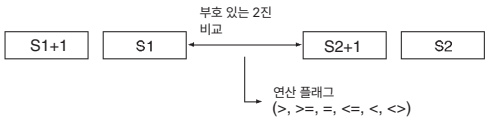
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
카운터	CNT (BCD)	---	 N: 카운터 번호 S: 설정값	CNT/CNTX(546)은 감산 카운터로 동작합니다. 설정값(SV)의 설정 범위는 BCD 에서는 0~9,999, 2진(10진 또는 16진)에서는 0~65,535 입니다. 	출력	필요
	CNTX (Binary)	546	 N: 카운터 번호 S: 설정값			
리버시블 카운터	CNTR (BCD)	012	 N: 카운터 번호 S: 설정값	CNTR(012)/CNTRX(548)은 리버시블 카운터로 동작합니다. 	출력	필요
	CNTRX (Binary)	548	 N: 카운터 번호 S: 설정값			
타이머/카운터 리셋	CNR @CNR (BCD)	545	 N1: 범위의 선두 번호 N2: 범위의 최종 번호	CNR(545)/CNRX(547)은 지정한 타이머 번호 또는 카운터 번호 범위 내의 타이머 또는 카운터를 리셋합니다. 설정값(SV)을 BCD 에서는 최대값 #9999, 2진에서는 #FFFF 로 설정합니다.	출력	필요
	CNRX @CNRX (Binary)	547	 N1: 범위의 선두 번호 N2: 범위의 최종 번호			

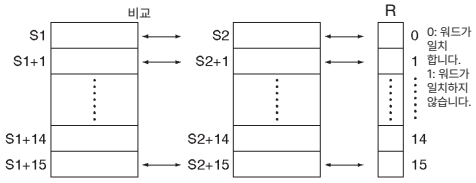
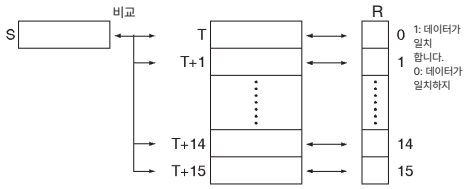
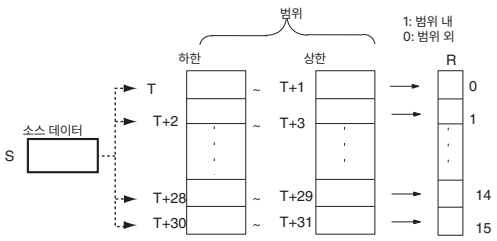
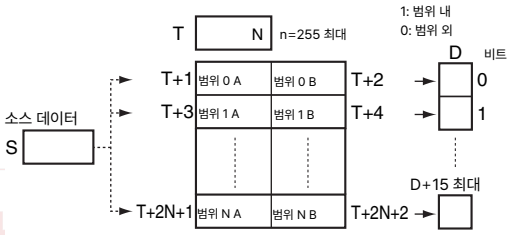
번역본
-For Reference Only-

A-1-5 비교 명령어

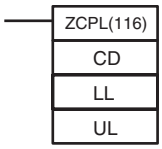
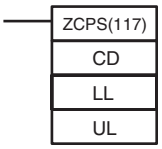
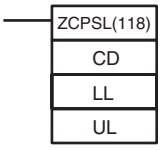
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
심볼 비교 (부호 없음)	LD, AND, OR + =, <>, <, <=, >, >=	300 (=) 305 (<>) 310 (<) 315 (<=) 320 (>) 325(>=)	 S1: 비교 데이터 1 S2: 비교 데이터 2	심볼 비교 명령어(부호 없음)는 두 값(상수 및/또는 지정한 워드의 내용)을 16 비트 바이너리 데이터로 비교하고, 비교 조건이 참이면 ON 실행 조건을 만듭니다. 심볼 비교 명령어에는 LD (LOAD), AND, OR 의 세 종류가 있습니다. 	LD: 논리 시작 AND, OR: 렁에 연속	LD: 필요 없음 AND, OR: 필요
심볼 비교 (더블 워드, 부호 없음)	LD, AND, OR + =, <>, <, <=, >, >= + L	301 (=) 306 (<>) 311 (<) 316 (<=) 321 (>) 326 (>=)	S1: 비교 데이터 1 S2: 비교 데이터 2	심볼 비교 명령어(더블 워드, 부호 없음)는 두 값(상수 및/또는 지정한 더블 워드 데이터의 내용)을 부호 없는 32 비트 2진 데이터로 비교하고, 비교 조건이 참이면 ON 실행 조건을 만듭니다. 심볼 비교 명령어에는 LD (LOAD), AND, OR 의 세 종류가 있습니다.	LD: 논리 시작 AND, OR: 렁에 연속	LD: 필요 없음 AND, OR: 필요
심볼 비교 (부호 있음)	LD, AND, OR + =, <>, <, <=, >, >= +S	302 (=) 307 (<>) 312 (<) 317 (<=) 322 (>) 327 (>=)	S1: 비교 데이터 1 S2: 비교 데이터 2	심볼 비교 명령어(부호 있음)는 두 값(상수 및/또는 지정한 워드의 내용)을 부호 있는 16 비트 2진(4 자리 16진)으로 비교하고, 비교 조건이 참이면 ON 실행 조건을 만듭니다. 심볼 비교 명령어에는 LD (LOAD), AND, OR 의 세 종류가 있습니다.	LD: 논리 시작 AND, OR: 렁에 연속	LD: 필요 없음 AND, OR: 필요
심볼 비교 (더블 워드, 부호 있음)	LD, AND, OR + =, <>, <, <=, >, >= +SL	303 (=) 308 (<>) 313 (<) 318 (<=) 323 (>) 328 (>=)	S1: 비교 데이터 1 S2: 비교 데이터 2	심볼 비교 명령어(더블 워드, 부호 있음)는 두 값(상수 및/또는 지정한 더블 워드 데이터의 내용)을 부호 있는 32 비트 2진 (8 자리 16진)으로 비교하고, 비교 조건이 참이면 ON 실행 조건을 만듭니다. 심볼 비교 명령어에는 LD (LOAD), AND, OR 의 세 종류가 있습니다.	LD: 논리 시작 AND, OR: 렁에 연속	LD: 필요 없음 AND, OR: 필요

AI 번역본
For Reference Only

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 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 (LOAD):  AND:  OR:  C: 제어 워드 S1: 현재 시각의 선두 워드 S2: 비교 시각의 선두 워드	시각 비교 명령어는 두 BCD 시각 값을 비교하고, 비교 조건이 참이면 ON 실행 조건을 만듭니다. 시각 비교 명령어에는 LD (LOAD), AND, OR 의 세 종류가 있습니다. 비교에서 시각 값(년, 월, 일, 시, 분, 초)을 마스크/마스크 해제할 수 있으므로 캘린더 타이머 기능을 쉽게 만들 수 있습니다.	LD: 논리 시작 AND, OR: 렁에 연속	LD: 필요 없음 AND, OR: 필요
부호 없는 비교	CMP !CMP	020	 S1: 비교 데이터 1 S2: 비교	두 부호 없는 2진 값(상수 및/또는 지정한 워드의 내용)을 비교하고, 그 결과를 보조 기억 영역의 연산 플래그로 출력합니다. 	출력	필요
더블 부호 없는 비교	CMPL	060	 S1: 비교 데이터 1 S2: 비교 데이터 2	두 더블 부호 없는 2진 값(상수 및/또는 지정한 워드의 내용) 을 비교하고, 그 결과를 보조 기억 영역의 연산 플래그로 출력합니다. 	출력	필요
부호 있는 BIN 비교	CPS !CPS	114	 S1: 비교 데이터 1 S2: 비교 데이터 2	두 부호 있는 2진 값(상수 및/또는 지정한 워드의 내용)을 비교하고, 그 결과를 보조 기억 영역의 연산 플래그로 출력합니다. 	출력	필요
더블 부호 있는 BIN 비교	CPSL	115	 S1: 비교 데이터 1 S2: 비교	두 더블 부호 있는 2진 값(상수 및/또는 지정한 워드의 내용) 을 비교하고, 그 결과를 보조 기억 영역의 연산 플래그로 출력합니다. 	출력	필요

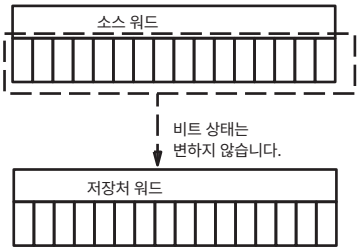
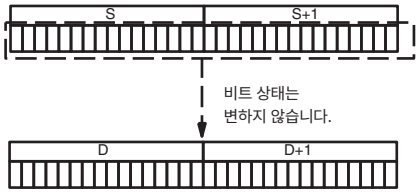
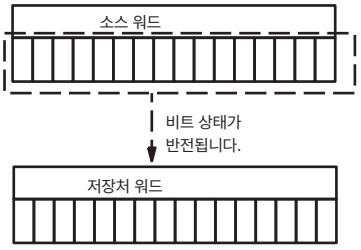
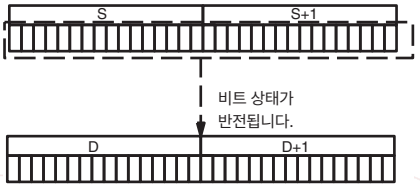
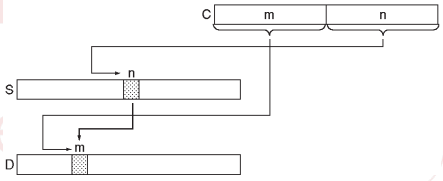
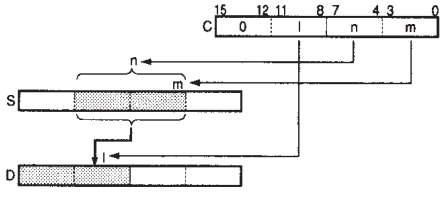
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
복수 비교	MCMP @MCMP	019	MCMP(019) S_1 S_2 R S1: 세트 1 의 선두 워드 S2: 세트 2 의 선두 워드 R: 결과 워드	연속된 16 워드를 또 다른 연속된 16 워드와 비교하고, 워드의 내용이 일치하지 않는 자리의 결과 워드 해당 비트를 ON 합니다. 	출력	필요
테이블 비교	TCMP @TCMP	085	TCMP(085) S T R S: 소스 데이터 T: 테이블의 선두 워드 R: 결과 워드	소스 데이터를 16 워드의 내용과 비교하고, 내용이 일치하면 결과 워드의 해당 비트를 ON 합니다. 	출력	필요
부호 없는 블록 비교	BCMP @BCMP	068	BCMP(068) S T R S: 소스 데이터 T: 테이블의 선두 워드 R: 결과 워드	소스 데이터를 16 개의 범위(16 개의 하한과 16 개의 상한으로 정의)와 비교하고, 소스 데이터가 범위 내에 있으면 결과 워드의 해당 비트를 ON 합니다. 	출력	필요
확장 블록 비교	BCMP2 @BCMP2	502	BCMP2(502) S T R S: 소스 데이터 T: 블록의 선두 워드 R: 결과 워드	소스 데이터를 최대 256 개의 범위(상한과 하한으로 정의)와 비교하고, 소스 데이터가 범위 내에 있으면 결과 워드의 해당 비트를 ON 합니다.  주: A 는 B 보다 작거나 같을 수도, B 보다 클 수도 있습니다.	출력	필요
영역 범위 비교	ZCP	088	ZCP(088) CD LL UL CD: 비교 데이터(1 워드) LL: 범위의 하한 UL: 범위의 상한	CD 의 16 비트 부호 없는 2진 값(워드 내용 또는 상수)을 LL 과 UL 로 정의된 범위와 비교하고, 그 결과를 보조 기억 영역의 연산 플래그로 출력합니다.	출력	필요

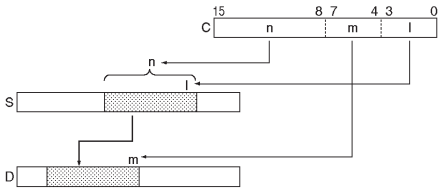
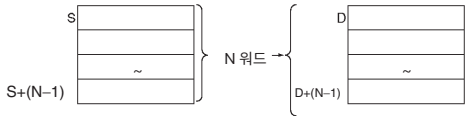
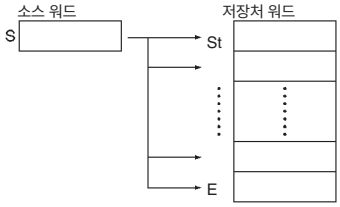
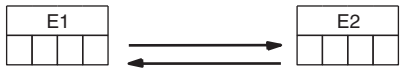
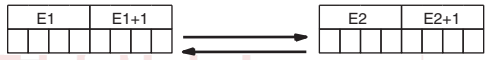
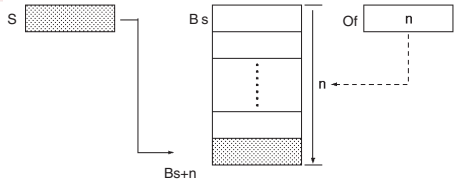
AI
-For Reference Only

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
더블 영역 범위 비교	ZCPL	116	 CD: 비교 데이터(2 워드) LL: 하한의 선두 워드 UL: 상한의 선두 워드	CD 와 CD+1 의 32 비트 부호 없는 2진 값(워드 내용 또는 상수)을 LL 과 UL 로 정의된 범위와 비교하고, 그 결과를 보조 기억 영역의 연산 플래그로 출력합니다.	출력	필요
부호 있는 영역 범위 비교	ZCPS	117	 CD: 비교 데이터(1 워드) LL: 범위의 하한 UL: 범위의 상한	CD 의 16 비트 부호 있는 2진 값(워드 내용 또는 상수)을 LL 과 UL 로 정의된 범위와 비교하고, 그 결과를 보조 기억 영역의 연산 플래그로 출력합니다.	출력	필요
더블 부호 있는 영역 범위 비교	ZCPSL	118	 CD: 비교 데이터(2 워드) LL: 하한의 선두 워드 UL: 상한의 선두 워드	CD 와 CD+1 의 32 비트 부호 있는 2진 값(워드 내용 또는 상수)을 LL 과 UL 로 정의된 범위와 비교하고, 그 결과를 보조 기억 영역의 연산 플래그로 출력합니다.	출력	필요

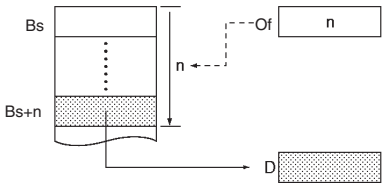
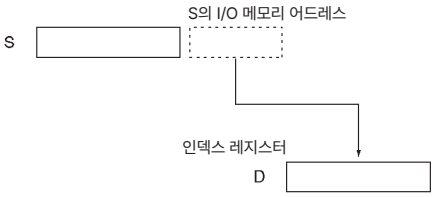
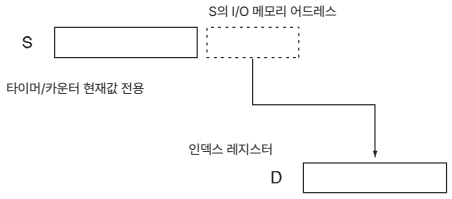
AI 번역본
-For Reference Only-

A-1-6 데이터 전송 명령어

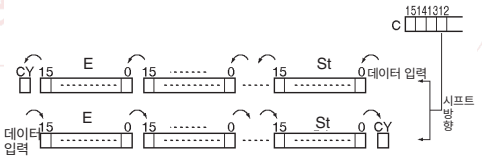
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
전송	MOV @MOV !MOV !@MOV	021	MOV(021) S D S: 소스 D: 저장처	1 워드의 데이터를 지정한 워드로 전송합니다. 	출력	필요
더블 전송	MOVL @MOVL	498	MOVL(498) S D S: 선두 소스 워드 D: 선두 저장처 워드	2 워드의 데이터를 지정한 워드로 전송합니다. 	출력	필요
전송 NOT	MVN @MVN	022	MVN(022) S D S: 소스 D: 저장처	1 워드 데이터의 보수를 지정한 워드로 전송합니다. 	출력	필요
더블 전송 NOT	MVNL @MVNL	499	MVNL(499) S D S: 선두 소스 워드 D: 선두 저장처 워드	2 워드 데이터의 보수를 지정한 워드로 전송합니다. 	출력	필요
비트 전송	MOVB @MOVB	082	MOVB(082) S C D S: 소스 워드 또는 데이터 C: 제어 워드 D: 저장처 워드	지정한 비트를 전송합니다. 	출력	필요
디지트 전송	MOVD @MOVD	083	MOVD(083) S C D S: 소스 워드 또는 데이터 C: 제어 워드 D: 저장처 워드	지정한 1 자리 또는 여러 자리를 전송합니다. (1 자리는 4 비트로 구성됩니다.) 	출력	필요

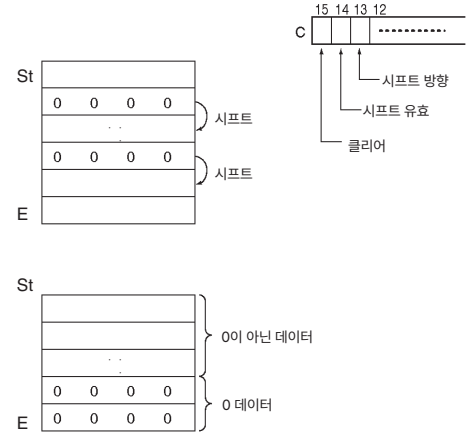
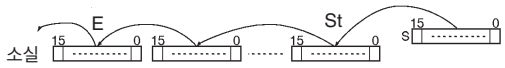
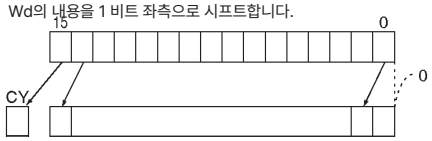
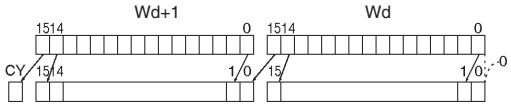
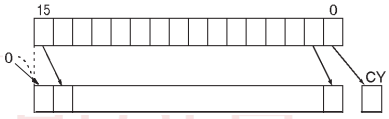
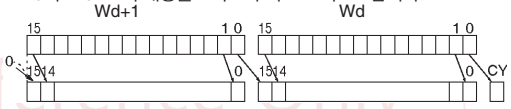
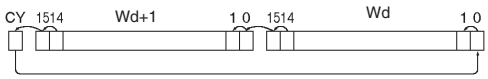
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
복수 비트 전송	XFRB @XFRB	062	XFRB(062) C S D C: 제어 워드 S: 선두 소스 워드 D: 선두 저장처 워드	지정한 개수의 연속된 비트를 전송합니다. 	출력	필요
블록 전송	XFER @XFER	070	XFER(070) N S D N: 워드 수 S: 선두 소스 워드 D: 선두 저장처 워드	지정한 개수의 연속된 워드를 전송합니다. 	출력	필요
블록 설정	BSET @BSET	071	BSET(071) S St E S: 소스 워드 St: 선두 워드 E: 최종 워드	동일한 워드를 연속된 워드 범위에 복사합니다. 	출력	필요
데이터 교환	XCHG @XCHG	073	XCHG(073) E1 E2 E1: 교환 워드 1 E2: 교환 워드 2	지정한 두 워드의 내용을 교환합니다. 	출력	필요
더블 데이터 교환	XCGL @XCGL	562	XCGL(562) E1 E2 E1: 교환 워드 1 E2: 두 번째 교환 워드	연속된 두 워드의 내용을 다른 연속된 두 워드와 교환합니다. 	출력	필요
싱글 워드 분배	DIST @DIST	080	DIST(080) S Bs Of S: 소스 워드 Bs: 저장처 베이스 어드레스 Of: 오프셋	소스 워드를, 베이스 어드레스에 오프셋 값을 더해 산출한 저장처 워드로 전송합니다. 	출력	필요

번역본
For Reference Only

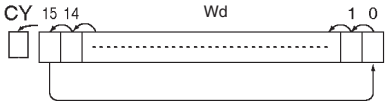
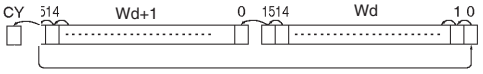
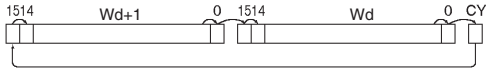
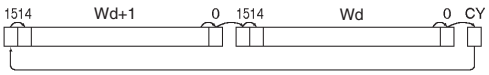
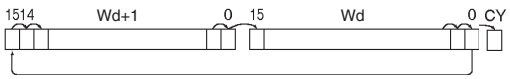
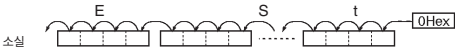
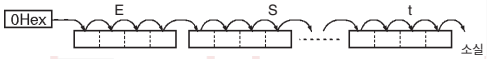
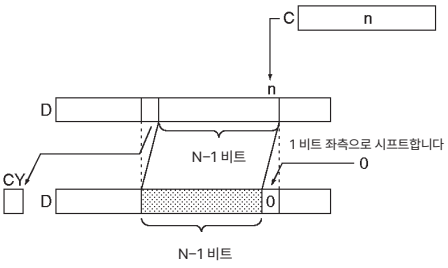
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
데이터 수집	COLL @COLL	081	COLL(081) Bs Of D Bs: 소스 베이스 어드레스 Of: 오프셋 D: 저장처 워드	소스 워드(베이스 어드레스에 오프셋 값을 더해 산출)를 저장처 워드로 전송합니다. 	출력	필요
레지스터로 전송	MOVR @MOVR	560	MOVR(560) S D S: 소스 (대상 워드 또는 비트) D: 저장처	지정한 워드, 비트 또는 타이머/카운터 완료 플래그의 내부 I/O 메모리 어드레스를 지정한 인덱스 레지스터에 설정합니다. (타이머/카운터 현재값의 내부 I/O 메모리 어드레스를 인덱스 레지스터에 설정할 때는 MOVRW(561)을 사용하십시오.) 	출력	필요
타이머/ 카운터 현재값을 레지스터로 전송	MOVRW @MOVRW	561	MOVRW(561) S D S: 소스 (대상 TC 번호) D: 저장처 (인덱스 레지스터)	지정한 타이머 또는 카운터의 현재값의 내부 I/O 메모리 어드레스를 지정한 인덱스 레지스터에 설정합니다. (워드, 비트 또는 타이머/카운터 완료 플래그의 내부 I/O 메모리 어드레스를 인덱스 레지스터에 설정할 때는 MOVR(560)을 사용하십시오.) 	출력	필요

A-1-7 데이터 시프트 명령어

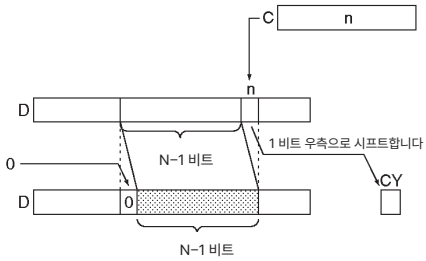
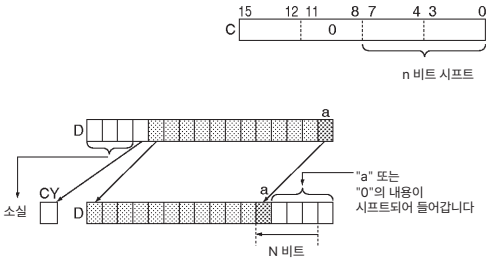
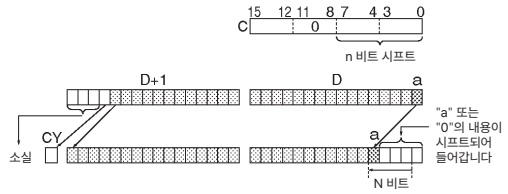
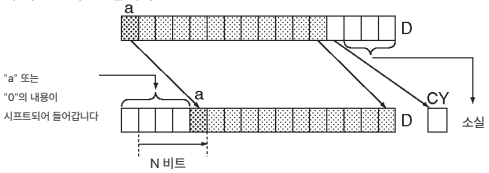
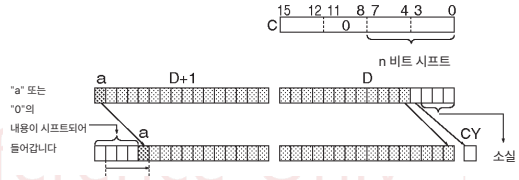
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
시프트 레지스터	SFT	010	데이터 입력 시프트 입력 리셋 입력 SFT(010) St E St: 선두 워드 E: 최종 워드	시프트 레지스터를 동작시킵니다. 	출력	필요
리버시블 시프트 레지스터	SFTR @SFTR	084	SFTR(084) C St E C: 제어 워드 St: 선두 워드 E: 최종 워드	데이터를 우측 또는 좌측으로 시프트하는 시프트 레지스터를 작성합니다. 	출력	필요

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
비동기 시프트 레지스터	ASFT @ASFT	017	ASFT(017) C St E C: 제어 워드 St: 선두 워드 E: 최종 워드	지정한 워드 범위 내의 0이 아닌 모든 워드 데이터를 St 방향 또는 E 방향으로 시프트하여 0000Hex 워드 데이터를 치환합니다. 	출력	필요
워드 시프트	WSFT @WSFT	016	WSFT(016) S St E S: 소스 워드 St: 선두 워드 E: 최종 워드	St부터 E까지의 데이터를 워드 단위로 시프트합니다. 	출력	필요
산술 좌 시프트	ASL @ASL	025	ASL(025) Wd Wd: 워드	Wd의 내용을 1 비트 좌측으로 시프트합니다. 	출력	필요
더블 좌 시프트	ASLL @ASLL	570	ASLL(570) Wd Wd: 워드	Wd와 Wd +1의 내용을 1 비트 좌측으로 시프트합니다. 	출력	필요
산술 우 시프트	ASR @ASR	026	ASR(026) Wd Wd: 워드	Wd의 내용을 1 비트 우측으로 시프트합니다. 	출력	필요
더블 우 시프트	ASRL @ASRL	571	ASRL(571) Wd Wd: 워드	Wd와 Wd +1의 내용을 1 비트 우측으로 시프트합니다. 	출력	필요
좌 로테이트	ROL @ROL	027	ROL(027) Wd Wd: 워드	캐리 플래그(CY)를 포함하여 Wd의 모든 비트를 1 비트 좌측으로 시프트합니다. 	출력	필요
더블 좌 로테이트	ROLL @ROLL	572	ROLL(572) Wd Wd: 워드	캐리 플래그(CY)를 포함하여 Wd와 Wd +1의 모든 비트를 1 비트 좌측으로 시프트합니다. 	출력	필요

사용 금지
-For Reference Only-

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
캐리 없는 좌 로테이트	RLNC @RLNC	574	RLNC(574) Wd Wd: 워드	캐리 플래그(CY)를 포함하지 않고 Wd의 모든 비트를 1 비트 좌측으로 시프트합니다. 	출력	필요
캐리 없는 더블 좌 로테이트	RLNL @RLNL	576	RLNL(576) Wd Wd: 워드	캐리 플래그(CY)를 포함하지 않고 Wd와 Wd +1의 모든 비트를 1 비트 좌측으로 시프트합니다. 	출력	필요
우 로테이트	ROR @ROR	028	ROR(028) Wd Wd: 워드	캐리 플래그(CY)를 포함하여 Wd의 모든 비트를 1 비트 우측으로 시프트합니다. 	출력	필요
더블 우 로테이트	RORL @RORL	573	RORL(573) Wd Wd: 워드	캐리 플래그(CY)를 포함하여 Wd와 Wd +1의 모든 비트를 1 비트 우측으로 시프트합니다. 	출력	필요
캐리 없는 우 로테이트	RRNC @RRNC	575	RRNC(575) Wd Wd: 워드	캐리 플래그(CY)를 포함하지 않고 Wd의 모든 비트를 1 비트 우측으로 시프트합니다. Wd의 최우측 비트의 내용이 최좌측 비트와 캐리 플래그(CY)로 시프트됩니다. 	출력	필요
캐리 없는 더블 우 로테이트	RRNL @RRNL	577	RRNL(577) Wd Wd: 워드	캐리 플래그(CY)를 포함하지 않고 Wd와 Wd +1의 모든 비트를 1 비트 우측으로 시프트합니다. Wd +1의 최우측 비트의 내용이 Wd의 최좌측 비트와 캐리 플래그(CY)로 시프트됩니다. 	출력	필요
1 자리 좌 시프트	SLD @SLD	074	SLD(074) St E St: 선두 워드 E: 최종 워드	데이터를 1 자리(4 비트) 좌측으로 시프트합니다. 	출력	필요
1 자리 우 시프트	SRD @SRD	075	SRD(075) St E St: 선두 워드 E: 최종 워드	데이터를 1 자리(4 비트) 우측으로 시프트합니다. 	출력	필요
N 비트 데이터 좌 시프트	NSFL @NSFL	578	NSFL(578) D C N D: 시프트 선두 워드 C: 선두 비트 N: 시프트 데이터 길이	지정한 비트 수만큼 좌측으로 시프트합니다. 	출력	필요

-For Reference Only-

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
N 비트 데이터 우 시프트	NSFR @NSFR	579	NSFR(579) D C N D: 시프트 선두 워드 C: 선두 비트 N: 시프트 데이터 길이	지정한 비트 수만큼 우측으로 시프트합니다. 	출력	필요
N 비트 좌 시프트	NASL @NASL	580	NASL(580) D C D: 시프트 워드 C: 제어 워드	지정한 16 비트의 워드 데이터를 지정한 비트 수만큼 좌측으로 시프트합니다. 	출력	필요
더블 N 비트 좌 시프트	NSLL @NSLL	582	NSLL(582) D C D: 시프트 워드 C: 제어 워드	지정한 32 비트의 워드 데이터를 지정한 비트 수만큼 좌측으로 시프트합니다. 	출력	필요
N 비트 우 시프트	NASR @NASR	581	NASR(581) D C D: 시프트 워드 C: 제어 워드	지정한 16 비트의 워드 데이터를 지정한 비트 수만큼 우측으로 시프트합니다. 	출력	필요
더블 N 비트 우 시프트	NSRL @NSRL	583	NSRL(583) D C D: 시프트 워드 C: 제어 워드	지정한 32 비트의 워드 데이터를 지정한 비트 수만큼 우측으로 시프트합니다. 	출력	필요

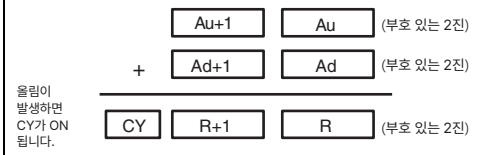
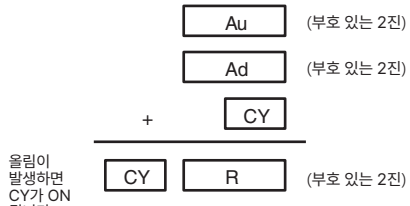
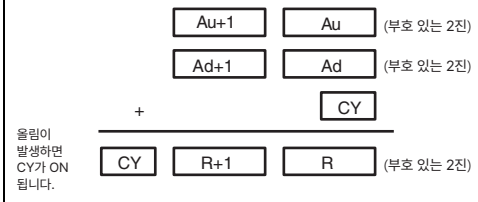
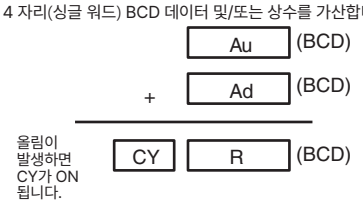
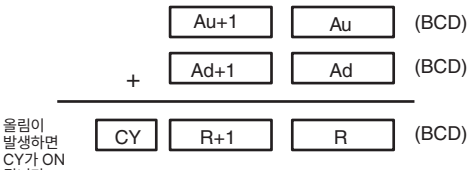
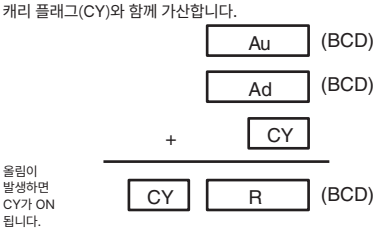
-For Reference Only-

A-1-8 증가/감소 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
BIN 인크리먼트	++ @++	590	++(590) Wd Wd: 워드	지정한 워드의 4 자리 16진 내용을 1 증가시킵니다. Wd +1 → Wd	출력	필요
BIN 더블 인크리먼트	++L @++L	591	++L(591) Wd Wd: 워드	지정한 워드의 8 자리 16진 내용을 1 증가시킵니다. Wd+1 Wd +1 → Wd+1 Wd	출력	필요
BIN 디크리먼트	– – @– –	592	– – (592) Wd Wd: 워드	지정한 워드의 4 자리 16진 내용을 1 감소시킵니다. Wd –1 → Wd	출력	필요
BIN 더블 디크리먼트	– –L @– –L	593	– – L(593) Wd Wd: 선두 워드	지정한 워드의 8 자리 16진 내용을 1 감소시킵니다. Wd+1 Wd –1 → Wd+1 Wd	출력	필요
BCD 인크리먼트	++B @++B	594	++B(594) Wd Wd: 워드	지정한 워드의 4 자리 BCD 내용을 1 증가시킵니다. Wd +1 → Wd	출력	필요
BCD 더블 인크리먼트	++BL @++BL	595	++BL(595) Wd Wd: 선두 워드	지정한 워드의 8 자리 BCD 내용을 1 증가시킵니다. Wd+1 Wd +1 → Wd+1 Wd	출력	필요
BCD 디크리먼트	– –B @– –B	596	– – B(596) Wd Wd: 워드	지정한 워드의 4 자리 BCD 내용을 1 감소시킵니다. Wd –1 → Wd	출력	필요
BCD 더블 디크리먼트	– –BL @– –BL	597	– – BL(597) Wd Wd: 선두 워드	지정한 워드의 8 자리 BCD 내용을 1 감소시킵니다. Wd+1 Wd –1 → Wd+1 Wd	출력	필요

A-1-9 심볼 연산 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
캐리 없는 부호 있는 BIN 가산	+ @+	400	+(400) Au Ad R Au: 피가수 워드 Ad: 가수 워드 R: 선두 결과 워드	4 자리(싱글 워드) 16진 데이터 및/또는 상수를 가산합니다. Au (부호 있는 2진) + Ad (부호 있는 2진) 올림이 발생하면 CY가 ON 됩니다. CY R (부호 있는 2진)	출력	필요

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
캐리 없는 부호 있는 BIN 가산 (더블)	+L @+L	401	+L(401) Au Ad R Au: 선두 피가수 워드 Ad: 선두 가수 워드 R: 선두 결과 워드	8 자리(더블 워드) 16진 데이터 및/또는 상수를 가산합니다. 	출력	필요
캐리 있는 부호 있는 BIN 가산	+C @+C	402	+C(402) Au Ad R Au: 피가수 워드 Ad: 가수 워드 R: 결과 워드	4 자리(싱글 워드) 16진 데이터 및/또는 상수를 캐리 플래그(CY)와 함께 가산합니다. 	출력	필요
캐리 있는 부호 있는 BIN 가산 (더블)	+CL @+CL	403	+CL(403) Au Ad R Au: 선두 피가수 워드 Ad: 선두 가수 워드 R: 선두 결과 워드	8 자리(더블 워드) 16진 데이터 및/또는 상수를 캐리 플래그(CY)와 함께 가산합니다. 	출력	필요
캐리 없는 BCD 가산	+B @+B	404	+B(404) Au Ad R Au: 피가수 워드 Ad: 가수 워드 R: 결과 워드	4 자리(싱글 워드) BCD 데이터 및/또는 상수를 가산합니다. 	출력	필요
캐리 없는 BCD 가산 (더블)	+BL @+BL	405	+BL(405) Au Ad R Au: 선두 피가수 워드 Ad: 선두 가수 워드 R: 선두 결과 워드	8 자리(더블 워드) BCD 데이터 및/또는 상수를 가산합니다. 	출력	필요
캐리 있는 BCD 가산	+BC @+BC	406	+BC(406) Au Ad R Au: 피가수 워드 Ad: 가수 워드 R: 결과 워드	4 자리(싱글 워드) BCD 데이터 및/또는 상수를 캐리 플래그(CY)와 함께 가산합니다. 	출력	필요

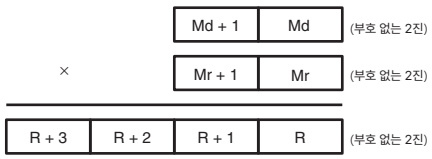
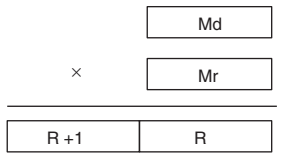
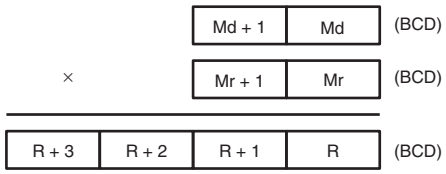
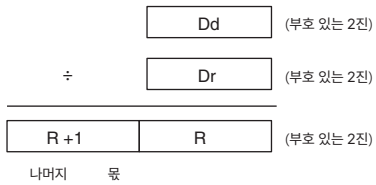
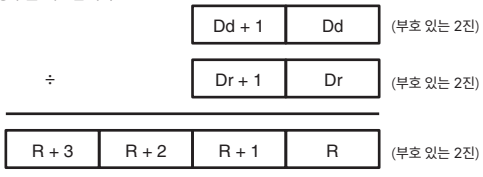
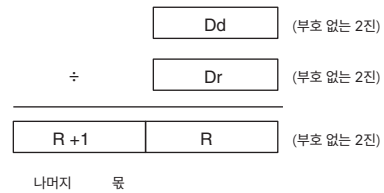
-Reference Only-

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
캐리 있는 BCD 가산 (더블)	+BCL @+BCL	407	+BCL(407) Au Ad R Au: 선두 피가수 워드 Ad: 선두 가수 워드 R: 선두 결과 워드	8 자리(더블 워드) BCD 데이터 및/또는 상수를 캐리 플래그(CY)와 함께 가산합니다. Au+1 Au (BCD) Ad+1 Ad (BCD) + CY 올림이 발생하면 CY가 ON 됩니다. CY R+1 R (BCD)	출력	필요
캐리 없는 부호 있는 BIN 감산	– @–	410	– (410) Mi Su R Mi: 피감수 워드 Su: 감수 워드	4 자리(싱글 워드) 16진 데이터 및/또는 상수를 감산합니다. Mi (부호 있는 2진) – Su (부호 있는 2진) 빌림이 발생하면 CY가 ON 됩니다. CY R (부호 있는 2진)	출력	필요
캐리 없는 부호 있는 BIN 감산 (더블)	– L @– L	411	–L(411) Mi Su R Mi: 피감수 워드 Su: 감수 워드 R: 결과 워드	8 자리(더블 워드) 16진 데이터 및/또는 상수를 감산합니다. Mi+1 Mi (부호 있는 2진) – Su+1 Su (부호 있는 2진) 빌림이 발생하면 CY가 ON 됩니다. CY R+1 R (부호 있는 2진)	출력	필요
캐리 있는 부호 있는 BIN 감산	– C @– C	412	–C(412) Mi Su R Mi: 피감수 워드 Su: 감수 워드 R: 결과 워드	4 자리(싱글 워드) 16진 데이터 및/또는 상수를 캐리 플래그(CY)와 함께 감산합니다. Mi (부호 있는 2진) Su (부호 있는 2진) – CY 빌림이 발생하면 CY가 ON 됩니다. CY R (부호 있는 2진)	출력	필요
캐리 있는 부호 있는 BIN (더블)	– CL @–- CL	413	–CL(413) Mi Su R Mi: 피감수 워드 Su: 감수 워드 R: 결과 워드	8 자리(더블 워드) 16진 데이터 및/또는 상수를 캐리 플래그(CY)와 함께 감산합니다. Mi+1 Mi (부호 있는 2진) Su+1 Su (부호 있는 2진) – CY 빌림이 발생하면 CY가 ON 됩니다. CY R+1 R (부호 있는 2진)	출력	필요
캐리 없는 BCD 감산	– B @– B	414	–B(414) Mi Su R Mi: 피감수 워드 Su: 감수 워드 R: 결과 워드	4 자리(싱글 워드) BCD 데이터 및/또는 상수를 감산합니다. Mi (BCD) – Su (BCD) 올림이 발생하면 CY가 ON 됩니다. CY R (BCD)	출력	필요

For Reference Only

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
캐리 없는 BCD 감산 (더블)	– BL @– BL	415	–BL(415) Mi Su R Mi: 선두 피감수 워드 Su: 선두 감수 워드 R: 선두 결과 워드	8 자리(더블 워드) BCD 데이터 및/또는 상수를 감산합니다. Mi +1 Mi (BCD) – Su+1 Su (BCD) 빌림이 발생하면 CY가 ON 됩니다. CY R+1 R (BCD)	출력	필요
캐리 있는 BCD 감산	– BC @– BC	416	–BC(416) Mi Su R Mi: 피감수 워드 Su: 감수 워드 R: 결과 워드	4 자리(싱글 워드) BCD 데이터 및/또는 상수를 캐리 플래그(CY)와 함께 감산합니다. Mi (BCD) Su (BCD) – CY 빌림이 발생하면 CY가 ON 됩니다. CY R (BCD)	출력	필요
캐리 있는 BCD 감산 (더블)	– BCL @– BCL	417	–BCL(417) Mi Su R Mi: 선두 피감수 워드 Su: 선두 감수 워드 R: 선두 결과 워드	8 자리(더블 워드) BCD 데이터 및/또는 상수를 캐리 플래그(CY)와 함께 감산합니다. Mi +1 Mi (BCD) Su+1 Su (BCD) – CY 빌림이 발생하면 CY가 ON 됩니다. CY R+1 R (BCD)	출력	필요
부호 있는 BIN 승산	* @*	420	*(420) Md Mr R Md: 피승수 워드 Mr: 승수 워드 R: 결과 워드	4 자리 부호 있는 16진 데이터 및/또는 상수를 승산합니다. Md (부호 있는 2진) × Mr (부호 있는 2진) R +1 R (부호 있는 2진)	출력	필요
부호 있는 BIN 승산 (더블)	*L @*L	421	*L(421) Md Mr R Md: 선두 피승수 워드 Mr: 선두 승수 워드 R: 선두 결과 워드	8 자리 부호 있는 16진 데이터 및/또는 상수를 승산합니다. Md + 1 Md (부호 있는 2진) × Mr + 1 Mr (부호 있는 2진) R + 3 R + 2 R + 1 R (부호 있는 2진)	출력	필요
부호 없는 BIN 승산	*U @*U	422	*U(422) Md Mr R Md: 피승수 워드 Mr: 승수 워드 R: 결과 워드	4 자리 부호 없는 16진 데이터 및/또는 상수를 승산합니다. Md (부호 없는 2진) × Mr (부호 없는 2진) R +1 R (부호 없는 2진)	출력	필요

-Reference Only-

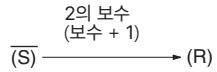
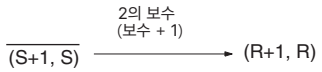
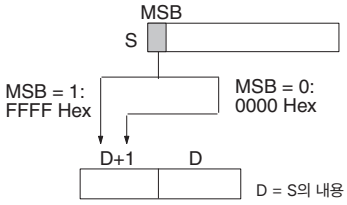
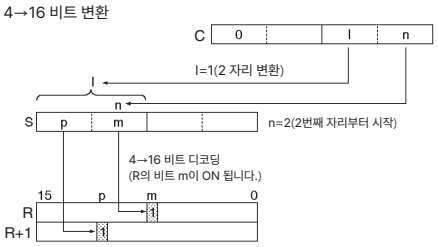
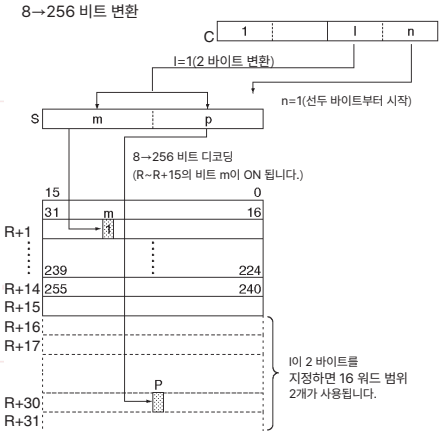
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
더블 부호 없는 BIN 승산	*UL @*UL	423	*UL(423) Md Mr R Md: 선두 피승수 워드 Mr: 선두 승수 워드 R: 선두 결과 워드	8 자리 부호 없는 16진 데이터 및/또는 상수를 승산합니다. 	출력	필요
BCD 승산	*B @*B	424	*B(424) Md Mr R Md: 피승수 워드 Mr: 승수 워드 R: 결과 워드	4 자리(싱글 워드) BCD 데이터 및/또는 상수를 승산합니다. 	출력	필요
BCD 승산 (더블)	*BL @*BL	425	*BL(425) Md Mr R Md: 선두 피승수 워드 Mr: 선두 승수 워드 R: 선두 결과 워드	8 자리(더블 워드) BCD 데이터 및/또는 상수를 승산합니다. 	출력	필요
부호 있는 BIN 제산	/ @/	430	/(430) Dd Dr R Dd: 피제수 워드 Dr: 제수 워드 R: 결과 워드	4 자리(싱글 워드) 부호 있는 16진 데이터 및/또는 상수를 제산합니다. 	출력	필요
부호 있는 BIN 제산 (더블)	/L @/L	431	/L(431) Dd Dr R Dd: 선두 피제수 워드 Dr: 선두 제수 워드 R: 선두 결과 워드	8 자리(더블 워드) 부호 있는 16진 데이터 및/또는 상수를 제산합니다. 	출력	필요
부호 없는 BIN 제산	/U @/U	432	/U(432) Dd Dr R Dd: 피제수 워드 Dr: 제수 워드 R: 결과 워드	4 자리(싱글 워드) 부호 없는 16진 데이터 및/또는 상수를 제산합니다. 	출력	필요

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
부호 없는 BIN 제산 (더블)	/UL @/UL	433	/UL(433) Dd Dr R Dd: 선두 피제수 워드 Dr: 선두 제수 워드 R: 선두 결과 워드	8 자리(더블 워드) 부호 없는 16진 데이터 및/또는 상수를 제산합니다. Dd + 1 | Dd (부호 없는 2진) ÷ Dr + 1 | Dr (부호 없는 2진) R + 3 | R + 2 | R + 1 | R (부호 없는 2진) 나머지 몫	출력	필요
BCD 제산	/B @/B	434	/B(434) Dd Dr R Dd: 피제수 워드 Dr: 제수 워드 R: 결과 워드	4 자리(싱글 워드) BCD 데이터 및/또는 상수를 제산합니다. Dd (BCD) ÷ Dr (BCD) R +1 | R (BCD) 나머지 몫	출력	필요
BCD 제산 (더블)	/BL @/BL	435	/BL(435) Dd Dr R Dd: 선두 피제수 워드 Dr: 선두 제수 워드 R: 선두 결과 워드	8 자리(더블 워드) BCD 데이터 및/또는 상수를 제산합니다. Dd + 1 | Dd (BCD) ÷ Dr + 1 | Dr (BCD) R + 3 | R + 2 | R + 1 | R (BCD) 나머지 몫	출력	필요

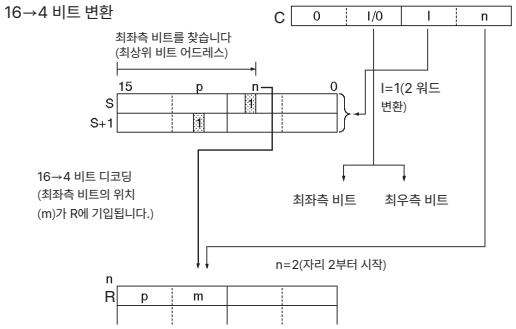
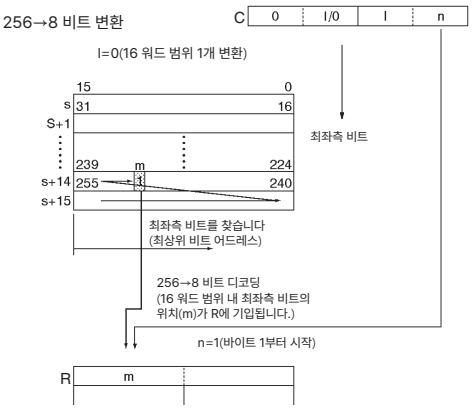
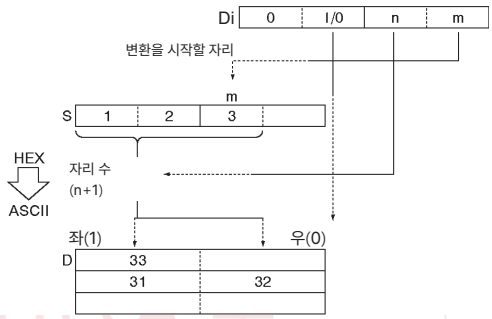
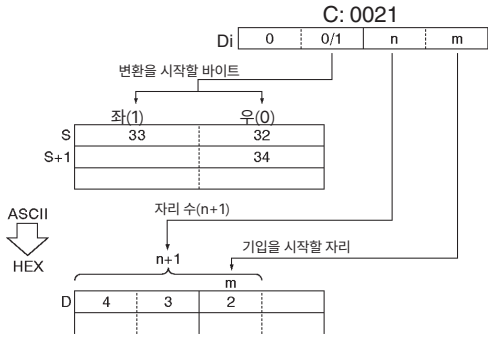
A-1-10 변환 명령어

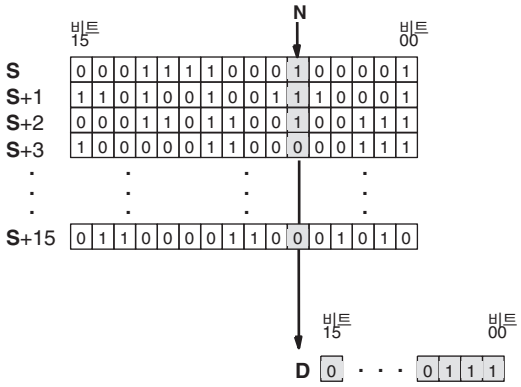
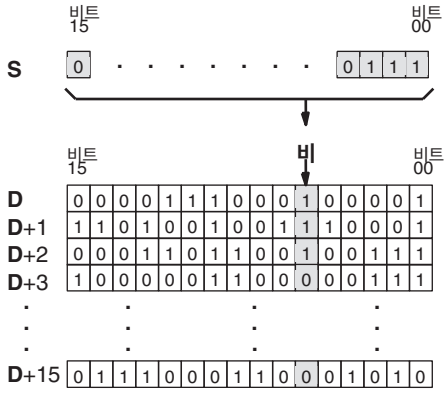
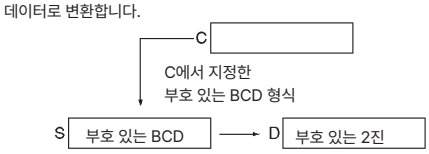
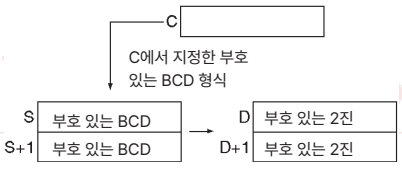
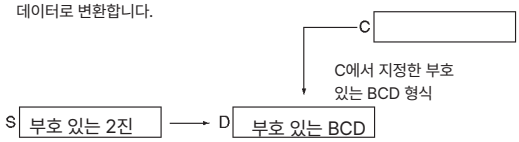
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
BCD→ BIN 변환	BIN @BIN	023	BIN(023) S R S: 소스 워드 R: 결과 워드	BCD 데이터를 바이너리 데이터로 변환합니다. S (BCD) → R (BIN)	출력	필요
BCD→ BIN 변환 (더블)	BINL @BINL	058	BINL(058) S R S: 선두 소스 워드 R: 선두 결과 워드	8 자리 BCD 데이터를 8 자리 16진(32 비트 2진) 데이터로 변환합니다. S (BCD) → R (BIN) S+1 (BCD) → R+1 (BIN)	출력	필요
BIN→ BCD 변환	BCD @BCD	024	BCD(024) S R S: 소스 워드 R: 결과 워드	1 워드의 바이너리 데이터를 1 워드의 BCD 데이터로 변환합니다. S (BIN) → R (BCD)	출력	필요
BIN→ BCD 변환 (더블)	BCDL @BCDL	059	BCDL(059) S R S: 선두 소스 워드 R: 선두 결과 워드	8 자리 16진(32 비트 2진) 데이터를 8 자리 BCD 데이터로 변환합니다. S (BIN) → R (BCD) S+1 (BIN) → R+1 (BCD)	출력	필요

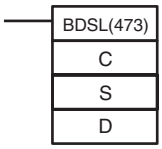
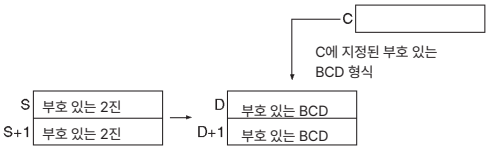
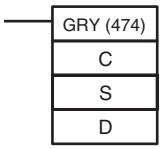
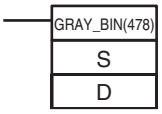
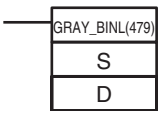
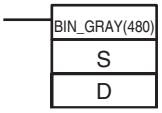
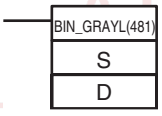
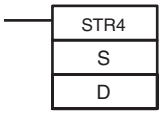
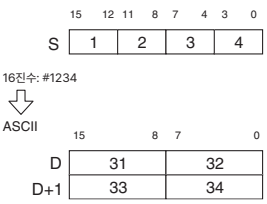
-For Reference Only-

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
2의 보수	NEG @NEG	160	NEG(160) S R S: 소스 워드 R: 결과 워드	1 워드 16진 데이터의 2의 보수를 산출합니다. 	출력	필요
2의 보수 (더블)	NEGL @NEGL	161	NEGL(161) S R S: 선두 소스 워드 R: 선두 결과 워드	2 워드 16진 데이터의 2의 보수를 산출합니다. 	출력	필요
16 비트 → 32 비트 부호 있는 BIN 변환	SIGN @SIGN	600	SIGN(600) S R S: 소스 워드 R: 선두 결과 워드	16 비트 부호 있는 2진 값을 32 비트 상당 값으로 확장합니다. 	출력	필요
데이터 디코더	MLPX @MLPX	076	MLPX(076) S C R S: 소스 워드 C: 제어 워드 R: 선두 결과 워드	소스 워드의 지정한 자리(또는 바이트)의 수치를 읽어, 결과 워드(또는 16 워드 범위)의 해당 비트를 ON 하고, 결과 워드(또는 16 워드 범위)의 그 밖의 모든 비트를 OFF 합니다. 4→16 비트 변환  8→256 비트 변환 	출력	필요

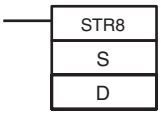
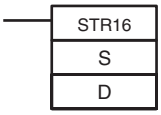
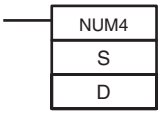
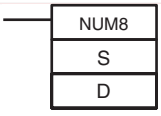
AI
-For Re

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
데이터 인코더	DMPX @DMPX	077	DMPX(077) S R C S: 선두 소스 워드 R: 결과 워드 C: 제어 워드	소스 워드(또는 16 워드 범위) 내에서 최초 또는 최후의 ON 비트 위치를 찾아, 그 값을 결과 워드의 지정한 자리(또는 바이트)에 기입합니다. 16→4 비트 변환  256→8 비트 변환  	출력	필요
ASCII 변환	ASC @ASC	086	ASC(086) S Di D S: 소스 워드 Di: 자리 지정 D: 선두 저장처 워드	소스 워드의 4 비트 16진 자리를 8 비트 ASCII 상당 값으로 변환합니다.  	출력	필요
ASCII→HEX 변환	HEX @HEX	162	HEX(162) S Di D S: 선두 소스 워드 Di: 자릿수 지정자 D: 저장처 워드	소스 워드의 최대 4 바이트의 ASCII 데이터를 16진 상당 값으로 변환하여, 그 자리들을 지정한 저장처 워드에 기입합니다.  	출력	필요

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
열 → 행 변환	LINE @LINE	063	LINE(063) S N D S: 선두 소스 워드 N: 비트 번호 D: 저장처 워드	16 워드 범위의 비트 열(연속된 16 워드의 동일한 비트 번호)을 저장처 워드의 16 비트로 변환합니다. 	출력	필요
행 → 열 변환	COLM @COLM	064	COLM(064) S D N S: 소스 워드 D: 선두 저장처 워드 N: 비트 번호	소스 워드의 16 비트를 16 워드 범위의 저장처 워드의 비트 열(연속된 16 워드의 동일한 비트 번호)로 변환합니다. 	출력	필요
부호 있는 BCD→ BIN 변환	BINS @BINS	470	BINS(470) C S D C: 제어 워드 S: 소스 워드 D: 저장처 워드	1 워드의 부호 있는 BCD 데이터를 1 워드의 부호 있는 바이너리 데이터로 변환합니다. 	출력	필요
부호 있는 BCD→ BIN 변환 (더블)	BISL @BISL	472	BISL(472) C S D C: 제어 워드 S: 선두 소스 워드 D: 선두 저장처 워드	더블 부호 있는 BCD 데이터를 더블 부호 있는 바이너리 데이터로 변환합니다. 	출력	필요
부호 있는 BIN→ BCD 변환	BCDS @BCDS	471	BCDS(471) C S D C: 제어 워드 S: 소스 워드 D: 저장처 워드	1 워드의 부호 있는 바이너리 데이터를 1 워드의 부호 있는 BCD 데이터로 변환합니다. 	출력	필요

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
더블 부호 있는 BIN → BCD 변환	BDSL @BDSL	473	 C: 제어 워드 S: 선두 소스 워드 D: 선두 저장처 워드	더블 부호 있는 바이너리 데이터를 더블 부호 있는 BCD 데이터로 변환합니다. 	출력	필요
그레이 코드 변환	GRY	474	 C: 제어 워드 S: 소스 워드 D: 선두 저장처 워드	지정한 워드의 그레이 코드 데이터를 지정한 분해능으로 바이너리, BCD 또는 각도(?) 데이터로 변환합니다.	출력	필요
그레이 코드 → BIN 변환	GRAY _BIN @GRAY_ BIN	478	 S: 소스 (그레이 코드) D: 저장처 (바이너리 데이터)	S로 지정한 그레이 코드 워드를 1워드의 바이너리 데이터로 변환하여 D에 출력합니다.	출력	필요
더블 그레이 코드 → BIN 변환	GRAY _BINL @GRAY_ BINL	479	 S: 선두 소스 워드(그레이 코드) D: 선두 저장처 워드(바이너리)	S로 지정한 2워드의 그레이 코드를 2워드의 바이너리 데이터로 변환하여 D에 출력합니다.	출력	필요
BIN → 그레이 코드 변환	BIN _GRAY @BIN_ GRAY	480	 S: 소스 워드 (바이너리 데이터) D: 저장처 워드(그레이 코드)	S로 지정한 바이너리 데이터 워드를 1워드의 그레이 코드로 변환하여 D에 출력합니다.	출력	필요
더블 BIN → 그레이 코드 변환	BIN _GRAYL @BIN_ GRAYL	481	 S: 선두 소스 워드(바이너리 데이터) D: 선두 저장처 워드(그레이 코드)	S로 지정한 2워드의 바이너리 데이터를 2워드의 그레이 코드로 변환하여 D에 출력합니다.	출력	필요
4자리 수치 → ASCII 변환	STR4 @STR4	601	 S: 수치 D: ASCII 문자열	4자리 16진수(#0000~#FFFF)를 ASCII 데이터(4문자)로 변환합니다. 	출력	필요

-For Reference Only-

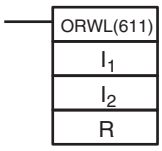
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
8자리 수치 → ASCII 변환	STR8 @STR8	602	 S: 수치 D: ASCII 문자열	8자리 16진수(#0000 0000~ #FFFF FFFF)를 ASCII 데이터(8문자)로 변환합니다. 15 12 11 8 7 4 3 0 S: 5 6 7 8 S+1: 1 2 3 4 16진수: #12345678 ⇩ ASCII 15 8 7 0 D: 31 32 D+1: 33 34 D+2: 35 36 D+3: 37 38	출력	필요
16자리 수치 → ASCII 변환	STR16 @STR16	603	 S: 수치 D: ASCII 문자열	16자리 16진수(#0000 0000 0000 0000~#FFFF FFFF FFFF FFFF)를 ASCII 데이터(16문자)로 변환합니다. 15 12 11 8 7 4 3 0 S: C D E F S+1: 8 9 A B S+2: 4 5 6 7 S+3: 0 1 2 3 16진수: #1234567890ABCDEF ⇩ ASCII 15 8 7 0 D: 30 31 D+1: 32 33 D+2: 34 35 D+3: 36 37 D+4: 38 39 D+5: 41 42 D+6: 43 44 D+7: 45 46	출력	필요
ASCII → 4자리 수치 변환	NUM4 @NUM4	604	 S: ASCII 문자열 D: 수치	ASCII 데이터 4문자를 4자리 16진수로 변환합니다. 15 8 7 0 S: 31 32 S+1: 33 34 ASCII ⇩ 16진수 15 12 11 8 7 4 3 0 D: 1 2 3 4	출력	필요
ASCII → 8자리 수치 변환	NUM8 @NUM8	605	 S: ASCII 문자열 D: 수치	ASCII 데이터 8문자를 8자리 16진수로 변환합니다. 15 8 7 0 S: 31 32 S+1: 33 34 S+2: 35 36 S+3: 37 38 ASCII ⇩ 16진수 15 12 11 8 7 4 3 0 D: 5 6 7 8 D+1: 1 2 3 4	출력	필요

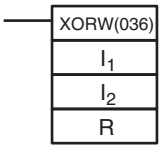
-For Re Only-

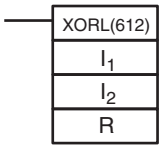
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
ASCII → 16자리 수치 변환	NUM16 @NUM16	606	STR16(606) S D S: ASCII 문자열 D: 수치	ASCII 데이터 16문자를 16자리 16진수로 변환합니다. 15 8 7 0 S 30 31 S+1 32 33 S+2 34 35 S+3 36 37 S+4 38 39 S+5 41 42 S+6 43 44 S+7 45 46 ASCII ⇩ 16진수 15 12 11 8 7 4 3 0 D C D E F D+1 8 9 A B D+2 4 5 6 7 D+3 0 1 2 3	출력	필요

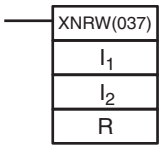
A-1-11 논리 명령어

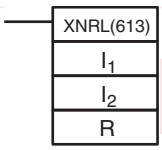
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
논리곱	ANDW @ANDW	034	ANDW(034) I_1 I_2 R I1: 입력 1 I2: 입력 2 R: 결과 워드	워드 데이터 및/또는 상수의 싱글 워드에서 대응하는 비트의 논리곱을 취합니다. **$I_1 \cdot I_2 \rightarrow R$** I_1 I_2 R 1 1 1 1 0 0 0 1 0 0 0 0	출력	필요
더블 논리곱	ANDL @ANDL	610	ANDL(610) I_1 I_2 R I1: 입력 1 I2: 입력 2 R: 결과 워드	워드 데이터 및/또는 상수의 더블 워드에서 대응하는 비트의 논리곱을 취합니다. **$(I_1,I_1+1) \cdot (I_2,I_2+1) \rightarrow (R, R+1)$** I_1,I_1+1 I_2,I_2+1 R, R+1 1 1 1 1 0 0 0 1 0 0 0 0	출력	필요
논리합	ORW @ORW	035	ORW(035) I_1 I_2 R I1: 입력 1 I2: 입력 2 R: 결과 워드	워드 데이터 및/또는 상수의 싱글 워드에서 대응하는 비트의 논리합을 취합니다. **$I_1 + I_2 \rightarrow R$** I_1 I_2 R 1 1 1 1 0 1 0 1 1 0 0 0	출력	필요

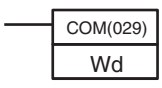
<table>
<tr><th>명령어</th><th>니모닉</th><th>코드</th><th>기호 · 오퍼랜드</th><th>기능</th><th>위치</th><th>실행 조건</th></tr>
<tr><td>더블
논리합</td><td>ORWL
@ORWL</td><td>611</td><td> 

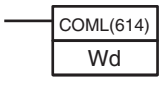
I1: 입력 1
I2: 입력 2
R: 결과 워드</td><td>워드 데이터 및/또는 상수의 더블 워드에서 대응하는 비트의 논리합을 취합니다.
(I1, I1+1) + (I2, I2+1) → (R, R+1)
<table>
<tr><th>I1, I1+1</th><th>I2, I2+1</th><th>R, R+1</th></tr>
<tr><td>1</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
</table></td><td>출력</td><td>필요</td></tr>
<tr><td>배타적
논리합</td><td>XORW
@XORW</td><td>036</td><td> 

I1: 입력 1
I2: 입력 2
R: 결과 워드</td><td>워드 데이터 및/또는 상수의 싱글 워드에서 대응하는 비트의 배타적 논리합을 취합니다.
I1. I2 + I1. I2 →R
<table>
<tr><th>I1</th><th>I2</th><th>R</th></tr>
<tr><td>1</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
</table></td><td>출력</td><td>필요</td></tr>
<tr><td>더블
배타적
논리합</td><td>XORL
@XORL</td><td>612</td><td> 

I1: 입력 1
I2: 입력 2
R: 결과 워드</td><td>워드 데이터 및/또는 상수의 더블 워드에서 대응하는 비트의 배타적 논리합을 취합니다.
(I1, I1+1). (<u>I2, I2+1</u>) + (<u>I1, I1+1</u>). (I2, I2+1) → (R, R+1)
<table>
<tr><th>I1, I1+1</th><th>I2, I2+1</th><th>R, R+1</th></tr>
<tr><td>1</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
</table></td><td>출력</td><td>필요</td></tr>
<tr><td>배타적
부정 논리합</td><td>XNRW
@XNRW</td><td>037</td><td> 

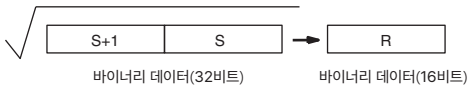
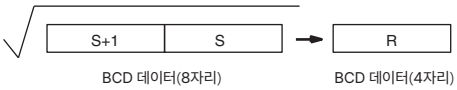
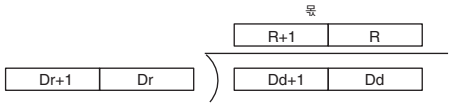
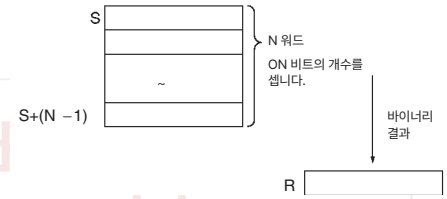
I1: 입력 1
I2: 입력 2
R: 결과 워드</td><td>워드 데이터 및/또는 상수의 대응하는 싱글 워드의 배타적 부정 논리합을 취합니다.
I1. I2 + I1. I2 →R
<table>
<tr><th>I1</th><th>I2</th><th>R</th></tr>
<tr><td>1</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>1</td></tr>
</table></td><td>출력</td><td>필요</td></tr>
<tr><td>더블
배타적
부정 논리합</td><td>XNRL
@XNRL</td><td>613</td><td> 

I1: 입력 1
I2: 입력 2
R: 선두 결과 워드</td><td>워드 데이터 및/또는 상수의 더블 워드에서 대응하는 비트의 배타적 부정 논리합을 취합니다.
(I1, I1+1). (I2, I2+1) + (<u>I1, I1+1</u>). (<u>I2, I2+1</u>) → (R, R+1)
<table>
<tr><th>I1, I1+1</th><th>I2, I2+1</th><th>R, R+1</th></tr>
<tr><td>1</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>1</td></tr>
</table></td><td>출력</td><td>필요</td></tr>
<tr><td>보수</td><td>COM
@COM</td><td>029</td><td> 

Wd: 워드</td><td>Wd 안의 ON 비트를 모두 OFF하고 OFF 비트를 모두 ON합니다.
<u>Wd</u> → Wd: 1 → 0 및 0 → 1</td><td>출력</td><td>필요</td></tr>
<tr><td>더블
보수</td><td>COML
@COML</td><td>614</td><td> 

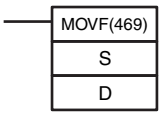
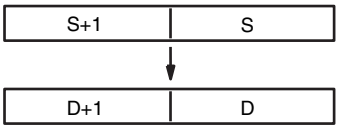
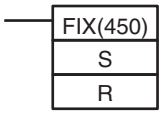
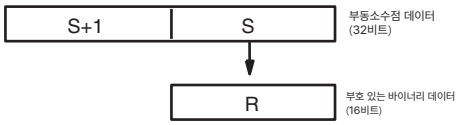
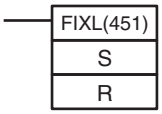
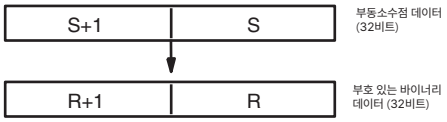
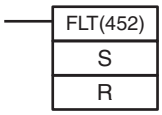
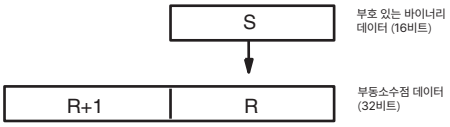
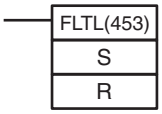
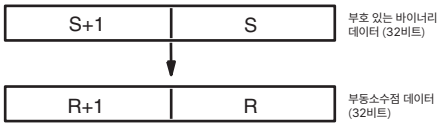
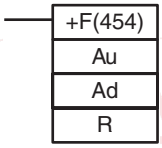
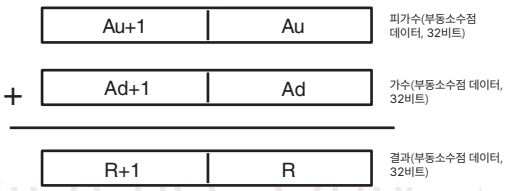
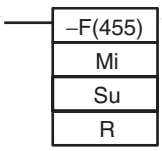
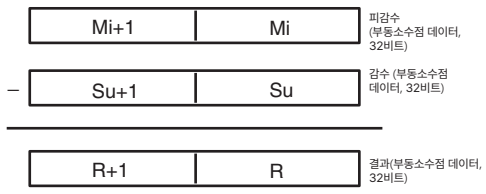
Wd: 워드</td><td>Wd와 Wd+1 안의 ON 비트를 모두 OFF하고 OFF 비트를 모두 ON합니다.
<u>(Wd+1, Wd)</u> → (Wd+1, Wd)</td><td>출력</td><td>필요</td></tr>
</table>

A-1-12 특수 연산 명령어

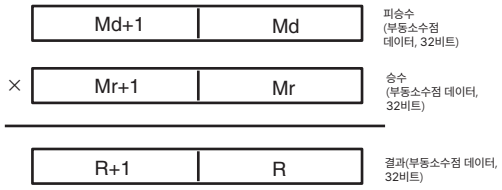
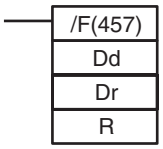
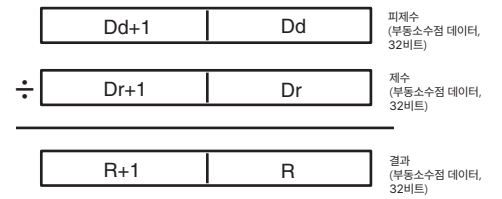
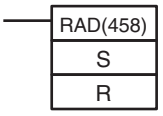
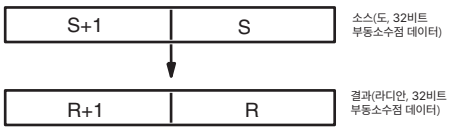
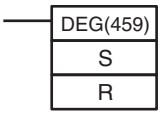
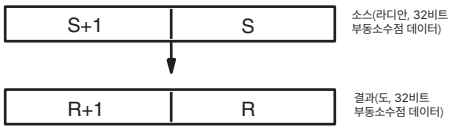
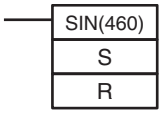
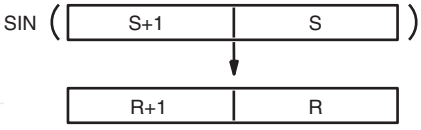
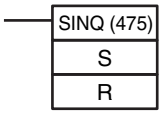
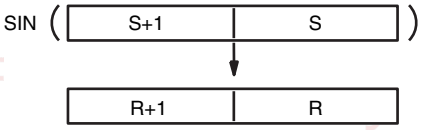
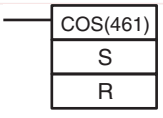
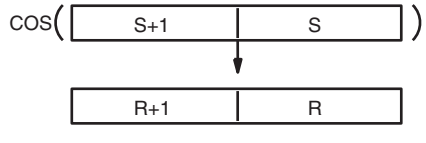
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
BIN 제곱근	ROTB @ROTB	620	ROTB(620) S R S: 선두 소스 워드 R: 결과 워드	지정한 워드의 32비트 바이너리 내용의 제곱근을 구하여 그 결과의 정수 부분을 지정한 결과 워드에 출력합니다. 	출력	필요
BCD 제곱근	ROOT @ROOT	072	ROOT(072) S R S: 선두 소스 워드 R: 결과 워드	8자리 BCD 수치의 제곱근을 구하여 그 결과의 정수 부분을 지정한 결과 워드에 출력합니다. 	출력	필요
산술 연산	APR @APR	069	APR(069) C S R C: 제어 워드 S: 소스 데이터 R: 결과 워드	SIN, COS 또는 선형 외삽을 연산합니다. SIN 또는 COS 연산: 각도 데이터(0°~90°)로부터 SIN 또는 COS를 연산하여 그 결과를 소수점 이하 4자리까지 BCD로 출력합니다. 선형 외삽: 지정한 입력 데이터로부터 선형 외삽을 바이너리로 연산하여 출력합니다.	출력	필요
부동소수점 제산	FDIV @FDIV	079	FDIV(079) Dd Dr R Dd: 선두 피제수 워드 Dr: 선두 제수 워드 R: 선두 결과 워드	7자리 부동소수점 수를 다른 부동소수점 수로 나눕니다. 부동소수점 수는 과학적 표기법(7자리 가수와 1자리 지수)으로 표현됩니다. 	출력	필요
비트 카운터	BCNT @BCNT	067	BCNT(067) N S R N: 워드 수 S: 선두 소스 워드 R: 결과 워드	지정한 워드의 ON 비트 총 개수를 셉니다. 	출력	필요

-For Reference Only-

A-1-13 부동소수점 연산 명령어

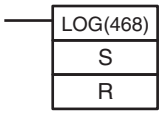
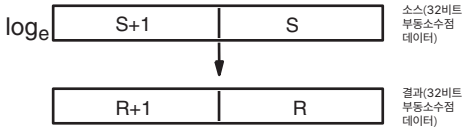
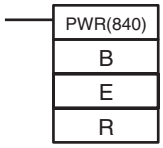
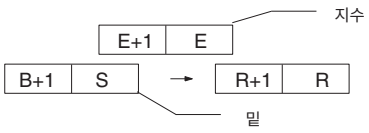
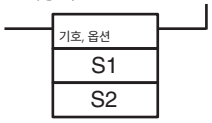
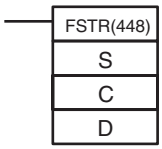
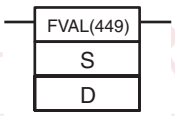
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
부동 소수점 전송 (단정밀도)	MOVF	469	 S: 선두 소스 워드 D: 선두 저장처 워드	지정한 32비트 부동소수점 수를 지정한 저장처 워드로 전송합니다. 	출력	필요
부동소수점 → 16비트 변환	FIX @FIX	450	 S: 선두 소스 워드 R: 결과 워드	32비트 부동소수점 값을 16비트 부호 있는 2진 데이터로 변환하여 그 결과를 지정한 결과 워드에 저장합니다. 	출력	필요
부동소수점 → 32비트 변환	FIXL @FIXL	451	 S: 선두 소스 워드 R: 선두 결과 워드	32비트 부동소수점 값을 32비트 부호 있는 2진 데이터로 변환하여 그 결과를 지정한 결과 워드에 저장합니다. 	출력	필요
16비트 → 부동소수점 변환	FLT @FLT	452	 S: 소스 워드 R: 선두 결과 워드	16비트 부호 있는 2진 값을 32비트 부동소수점 데이터로 변환하여 그 결과를 지정한 결과 워드에 저장합니다. 	출력	필요
32비트 → 부동소수점 변환	FLTL @FLTL	453	 S: 선두 소스 워드 R: 선두 결과 워드	32비트 부호 있는 2진 값을 32비트 부동소수점 데이터로 변환하여 그 결과를 지정한 결과 워드에 저장합니다. 	출력	필요
부동 소수점 가산	+F @+F	454	 Au: 선두 피가수 워드 AD: 선두 가수 워드 R: 선두 결과 워드	두 개의 32비트 부동소수점 수를 가산하여 그 결과를 지정한 결과 워드에 저장합니다. 	출력	필요
부동 소수점 감산	–F @–F	455	 Mi: 선두 피감수 워드 Su: 선두 감수 워드 R: 선두 결과 워드	32비트 부동소수점 수에서 다른 32비트 부동소수점 수를 감산하여 그 결과를 지정한 결과 워드에 저장합니다. 	출력	필요

Reference Only

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
부동 소수 점 승산	*F @*F	456	 Md: 선두 피승수 워드 Mr: 선두 승수 워드 R: 선두 결과 워드	두 개의 32비트 부동소수점 수를 승산하여 그 결과를 지정한 결과 워드에 저장합니다. 	출력	필요
부동 소수 점 제산	/F @/F	457	 Dd: 선두 피제수 워드 Dr: 선두 제수 워드 R: 선두 결과 워드	32비트 부동소수점 수를 다른 32비트 부동소수점 수로 나누어 그 결과를 지정한 결과 워드에 저장합니다. 	출력	필요
도 → 라디안 변환	RAD @RAD	458	 S: 선두 소스 워드 R: 선두 결과 워드	32비트 부동소수점 수를 도에서 라디안으로 변환하여 그 결과를 지정한 결과 워드에 저장합니다. 	출력	필요
라디안 → 도 변환	DEG @DEG	459	 S: 선두 소스 워드 R: 선두 결과 워드	32비트 부동소수점 수를 라디안에서 도로 변환하여 그 결과를 지정한 결과 워드에 저장합니다. 	출력	필요
사인	SIN @SIN	460	 S: 선두 소스 워드 R: 선두 결과 워드	32비트 부동소수점 수(라디안)의 사인을 연산하여 그 결과를 지정한 결과 워드에 저장합니다. 	출력	필요
고속 사인	SINQ @SINQ	475	 S: 선두 소스 워드 R: 선두 결과 워드	32비트 부동소수점 수(라디안)의 사인을 연산하여 그 결과를 지정한 결과 워드에 저장합니다. 	출력	필요
코사인	COS @COS	461	 S: 선두 소스 워드 R: 선두 결과 워드	32비트 부동소수점 수(라디안)의 코사인을 연산하여 그 결과를 지정한 결과 워드에 저장합니다. 	출력	필요

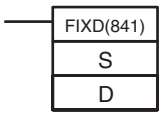
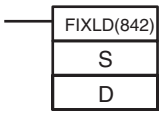
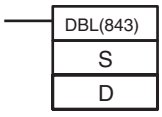
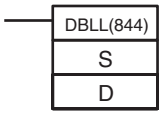
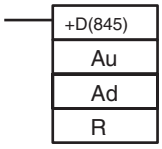
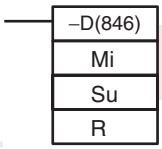
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
고속 코사인	COSQ @COSQ	476	COSQ(476) S R S: 선두 소스 워드 R: 선두 결과 워드	32비트 부동소수점 수(라디안)의 코사인을 연산하여 그 결과를 지정한 결과 워드에 저장합니다. COS(S+1 | S) R+1 | R	출력	필요
탄젠트	TAN @TAN	462	TAN(462) S R S: 선두 소스 워드 R: 선두 결과 워드	32비트 부동소수점 수(라디안)의 탄젠트를 연산하여 그 결과를 지정한 결과 워드에 저장합니다. TAN (S+1 | S) R+1 | R	출력	필요
고속 탄젠트	TANQ @TANQ	477	TANQ(477) S R S: 선두 소스 워드 R: 선두 결과 워드	32비트 부동소수점 수(라디안)의 탄젠트를 연산하여 그 결과를 지정한 결과 워드에 저장합니다. TAN (S+1 | S) R+1 | R	출력	필요
아크 사인	ASIN @ASIN	463	ASIN(463) S R S: 선두 소스 워드 R: 선두 결과 워드	32비트 부동소수점 수의 아크 사인을 연산하여 그 결과를 지정한 결과 워드에 저장합니다. (아크 사인 함수는 사인 함수의 역함수로, −1에서 1 사이의 주어진 사인 값을 만드는 각도를 반환합니다.) SIN^{-1}(S+1 | S) R+1 | R	출력	필요
아크 코사인	ACOS @ACOS	464	ACOS(464) S R S: 선두 소스 워드 R: 선두 결과 워드	32비트 부동소수점 수의 아크 코사인을 연산하여 그 결과를 지정한 결과 워드에 저장합니다. (아크 코사인 함수는 코사인 함수의 역함수로, −1에서 1 사이의 주어진 코사인 값을 만드는 각도를 반환합니다.) COS^{-1}(S+1 | S) R+1 | R	출력	필요
아크 탄젠트	ATAN @ATAN	465	ATAN(465) S R S: 선두 소스 워드 R: 선두 결과 워드	32비트 부동소수점 수의 아크 탄젠트를 연산하여 그 결과를 지정한 결과 워드에 저장합니다. (아크 탄젠트 함수는 탄젠트 함수의 역함수로, 주어진 탄젠트 값을 만드는 각도를 반환합니다.) TAN^{-1}(S+1 | S) R+1 | R	출력	필요
제곱근	SQRT @SQRT	466	SQRT(466) S R S: 선두 소스 워드 R: 선두 결과 워드	32비트 부동소수점 수의 제곱근을 연산하여 그 결과를 지정한 결과 워드에 저장합니다. $\sqrt{}$ S+1 | S R+1 | R	출력	필요
지수	EXP @EXP	467	EXP(467) S R S: 선두 소스 워드 R: 선두 결과 워드	32비트 부동소수점 수의 자연(밑 e) 지수를 연산하여 그 결과를 지정한 결과 워드에 저장합니다. S+1 | S (소스(32비트 부동소수점 데이터)) e R+1 | R (결과(32비트 부동소수점 데이터))	출력	필요

-For Reference Only-

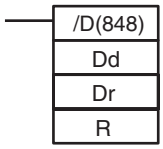
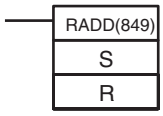
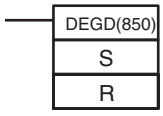
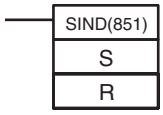
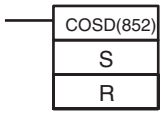
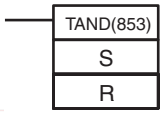
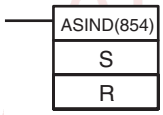
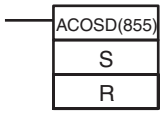
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
로그	LOG @LOG	468	 S: 선두 소스 워드 R: 선두 결과 워드	32비트 부동소수점 수의 자연(밑 e) 로그를 연산하여 그 결과를 지정한 결과 워드에 저장합니다. 	출력	필요
지수 거듭 제곱	PWR @PWR	840	 B: 선두 밑 워드 E: 선두 지수 워드 R: 선두 결과 워드	32비트 부동소수점 수를 다른 32비트 부동소수점 수의 거듭제곱으로 계산합니다. 	출력	필요
부동소수점 기호 비교	LD, AND. 또는 OR + =F, <>F, <F, <=F, >F, 또는 >=F	329 (=F), 330 (<>F), 331 (<F), 332 (<=F), 333 (>F), 334 (>=F)	LD 사용 시:  AND 사용 시:  OR 사용 시:  S1: 비교 데이터 1 S2: 비교 데이터 2	지정한 단정밀도 데이터(32비트) 또는 상수를 비교하여 비교 결과가 참이면 ON 실행 조건을 만듭니다. 부동소수점 기호 비교 명령어에는 LD(Load), AND, OR의 세 가지 기호를 사용할 수 있습니다.	LD: 논리 시작. AND 또는 OR: 렁에서 계속 이어짐	LD: 필요 없음 AND 또는 OR: 필요
부동 소수점 → ASCII 변환	FSTR @FSTR	448	 S: 선두 소스 워드 C: 제어 워드 D: 저장처 워드	지정한 단정밀도 부동소수점 데이터(32비트 소수점 형식 또는 지수 형식)를 문자열 데이터(ASCII)로 변환하여 그 결과를 저장처 워드에 출력합니다.	출력	필요
ASCII → 부동소수점 변환	FVAL @FVAL	449	 S: 소스 워드 D: 선두 저장처 워드	지정한 단정밀도 부동소수점 데이터(소수점 형식 또는 지수 형식)의 문자열(ASCII) 표현을 32비트 단정밀도 부동소수점 데이터로 변환하여 그 결과를 저장처 워드에 출력합니다.	출력	필요

- 시 번역본 -
- Reference Only -

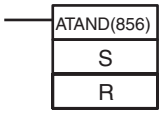
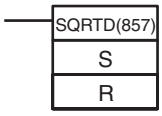
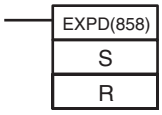
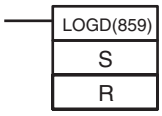
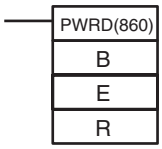
A-1-14 배정밀도 부동소수점 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
부동소수점 → 16비트 BIN 변환(배정밀도)	FIXD @FIXD	841	 S: 선두 소스 워드 D: 저장처 워드	지정한 배정밀도 부동소수점 데이터(64비트)를 16비트 부호 있는 바이너리 데이터로 변환하여 그 결과를 저장처 워드에 출력합니다.	출력	필요
부동소수점 → 32비트 BIN 변환(배정밀도)	FIXLD @FIXLD	842	 S: 선두 소스 워드 D: 선두 저장처 워드	지정한 배정밀도 부동소수점 데이터(64비트)를 32비트 부호 있는 바이너리 데이터로 변환하여 그 결과를 저장처 워드에 출력합니다.	출력	필요
16비트 BIN → 부동소수점 변환(배정밀도)	DBL @DBL	843	 S: 소스 워드 D: 선두 저장처 워드	지정한 16비트 부호 있는 바이너리 데이터를 배정밀도 부동소수점 데이터(64비트)로 변환하여 그 결과를 저장처 워드에 출력합니다.	출력	필요
32비트 BIN → 부동소수점 변환(배정밀도)	DBLL @DBLL	844	 S: 선두 소스 워드 D: 선두 저장처 워드	지정한 32비트 부호 있는 바이너리 데이터를 배정밀도 부동소수점 데이터(64비트)로 변환하여 그 결과를 저장처 워드에 출력합니다.	출력	필요
부동소수점 가산 (배정밀도)	+D @+D	845	 Au: 선두 피가수 워드 Ad: 선두 가수 워드 R: 선두 결과 워드	지정한 배정밀도 부동소수점 값(각 64비트)을 가산하여 그 결과를 결과 워드에 출력합니다.	출력	필요
부동 소수점 감산 (배정밀도)	–D @–D	846	 Mi: 선두 피감수 워드 Su: 선두 감수 워드 R: 선두 결과 워드	지정한 배정밀도 부동소수점 값(각 64비트)을 감산하여 그 결과를 결과 워드에 출력합니다.	출력	필요
부동 소수점 승산 (배정밀도)	*D @*D	847	 Md: 선두 피승수 워드 Mr: 선두 승수 워드 R: 선두 결과 워드	지정한 배정밀도 부동소수점 값(각 64비트)을 승산하여 그 결과를 결과 워드에 출력합니다.	출력	필요

번역본
-Reference Only-

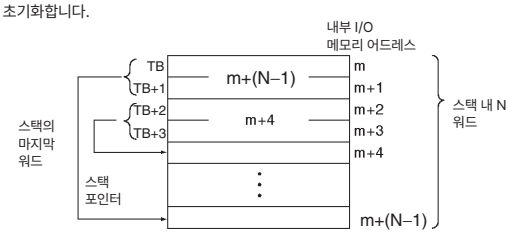
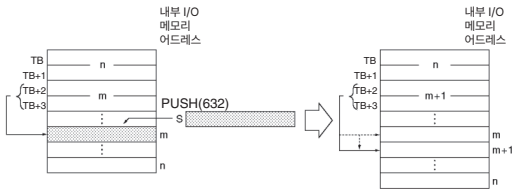
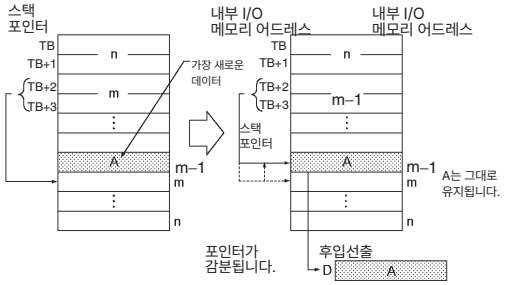
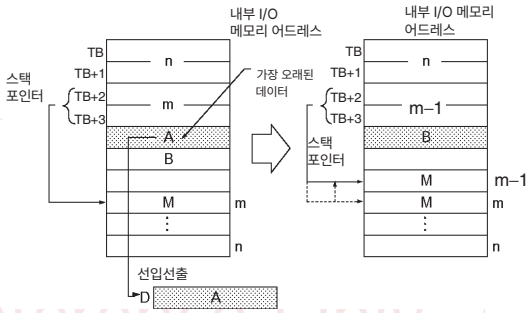
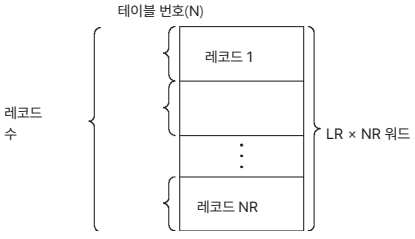
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
부동소수점 제산 (배정밀도)	/D @/D	848	 Dd: 선두 피제수 워드 Dr: 선두 제수 워드 R: 선두 결과 워드	지정한 배정밀도 부동소수점 값(각 64비트)을 제산하여 그 결과를 결과 워드에 출력합니다.	출력	필요
도 → 라디안 변환 (배정밀도)	RADD @RADD	849	 S: 선두 소스 워드 R: 선두 결과 워드	지정한 배정밀도 부동소수점 데이터(64비트)를 도에서 라디안으로 변환하여 그 결과를 결과 워드에 출력합니다.	출력	필요
라디안 → 도 변환 (배정밀도)	DEGD @DEGD	850	 S: 선두 소스 워드 R: 선두 결과 워드	지정한 배정밀도 부동소수점 데이터(64비트)를 라디안에서 도로 변환하여 그 결과를 결과 워드에 출력합니다.	출력	필요
사인(배정밀도)	SIND @SIND	851	 S: 선두 소스 워드 R: 선두 결과 워드	지정한 배정밀도 부동소수점 데이터(64비트)의 각도 (라디안)에 대한 사인을 연산하여 그 결과를 결과 워드에 출력합니다.	출력	필요
코사인 (배정밀도)	COSD @COSD	852	 S: 선두 소스 워드 R: 선두 결과 워드	지정한 배정밀도 부동소수점 데이터(64비트)의 각도 (라디안)에 대한 코사인을 연산하여 그 결과를 결과 워드에 출력합니다.	출력	필요
탄젠트 (배정밀도)	TAND @TAND	853	 S: 선두 소스 워드 R: 선두 결과 워드	지정한 배정밀도 부동소수점 데이터(64비트)의 각도 (라디안)에 대한 탄젠트를 연산하여 그 결과를 결과 워드에 출력합니다.	출력	필요
아크 사인 (배정밀도)	ASIND @ASIND	854	 S: 선두 소스 워드 R: 선두 결과 워드	지정한 배정밀도 부동소수점 데이터(64비트)의 사인 값으로부터 각도(라디안)를 연산하여 그 결과를 결과 워드에 출력합니다. (아크 사인 함수는 사인 함수의 역함수로, −1에서 1 사이의 주어진 사인 값을 만드는 각도를 반환합니다.)	출력	필요
아크 코사인 (배정밀도)	ACOSD @ACOSD	855	 S: 선두 소스 워드 R: 선두 결과 워드	지정한 배정밀도 부동소수점 데이터(64비트)의 코사인 값으로부터 각도(라디안)를 연산하여 그 결과를 결과 워드에 출력합니다. (아크 코사인 함수는 코사인 함수의 역함수로, −1에서 1 사이의 주어진 코사인 값을 만드는 각도를 반환합니다.)	출력	필요

-Reference Only-

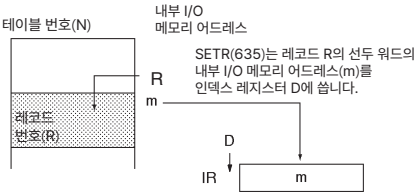
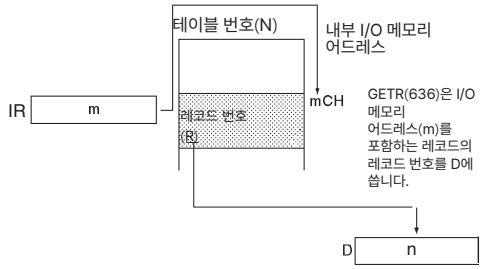
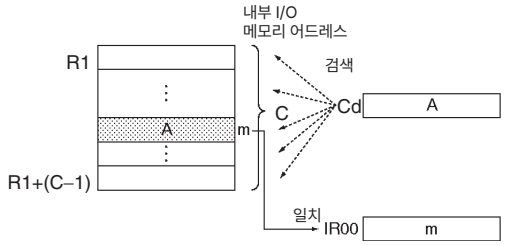
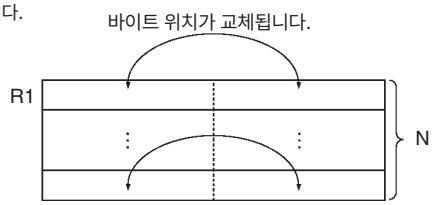
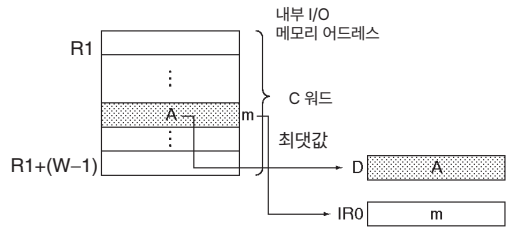
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
아크 탄젠트 (배정밀도)	ATAND @ATAND	856	 S: 선두 소스 워드 R: 선두 결과 워드	지정한 배정밀도 부동소수점 데이터(64비트)의 탄젠트 값으로부터 각도(라디안)를 연산하여 그 결과를 결과 워드에 출력합니다. (아크 탄젠트 함수는 탄젠트 함수의 역함수로, 주어진 탄젠트 값을 만드는 각도를 반환합니다.)	출력	필요
제곱근 (배정밀도)	SQRTD @SQRTD	857	 S: 선두 소스 워드 R: 선두 결과 워드	지정한 배정밀도 부동소수점 데이터(64비트)의 제곱근을 연산하여 그 결과를 결과 워드에 출력합니다.	출력	필요
지수 (배정밀도)	EXPD @EXPD	858	 S: 선두 소스 워드 R: 선두 결과 워드	지정한 배정밀도 부동소수점 데이터(64비트)의 자연 (밑 e) 지수를 연산하여 그 결과를 결과 워드에 출력합니다.	출력	필요
로그 (배정밀도)	LOGD @LOGD	859	 S: 선두 소스 워드 R: 선두 결과 워드	지정한 배정밀도 부동소수점 데이터(64비트)의 자연 (밑 e) 로그를 연산하여 그 결과를 결과 워드에 출력합니다.	출력	필요
지수 거듭제곱 (배정밀도)	PWRD @PWRD	860	 B: 선두 밑 워드 E: 선두 지수 워드 R: 선두 결과 워드	배정밀도 부동소수점 수(64비트)를 다른 배정밀도 부동소수점 수의 거듭제곱으로 계산하여 그 결과를 결과 워드에 출력합니다.	출력	필요
기호 비교 (배정밀도)	LD, AND. 또는 OR + =D, <>D, <D, <=D, >D, 또는 >=D	LD, AND. 또는 OR + 335 (=D), 336 (<>D), 337 (<D), 338 (<=D), 339 (>D), 340 (>=D)	LD 사용 시:  AND 사용 시:  OR 사용 시:  S1: 비교 데이터 1 S2: 비교 데이터 2	지정한 배정밀도 데이터(64비트)를 비교하여 비교 결과가 참이면 ON 실행 조건을 만듭니다. 부동소수점 기호 비교 명령어에는 LD(Load), AND, OR의 세 가지 기호를 사용할 수 있습니다.	LD: 논리 시작. AND 또는 OR: 렁에서 계속 이어짐	LD: 필요 없음 AND 또는 OR: 필요

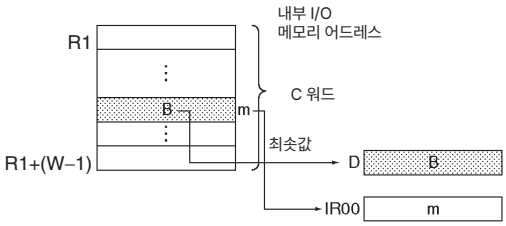
번역본
-Reference Only-

A-1-15 테이블 데이터 처리 명령어

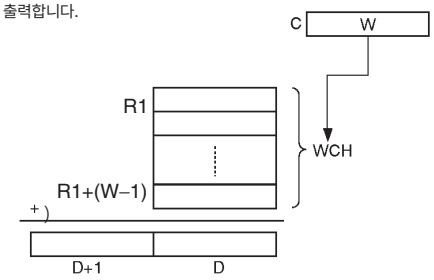
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
스택 설정	SSET @SSET	630	SSET(630) TB N TB: 선두 스택 어드레스 N: 워드 수	지정한 워드에서 시작하는 지정한 길이의 스택을 정의하고 데이터 영역의 워드를 모두 0으로 초기화합니다. 	출력	필요
스택에 푸시	PUSH @PUSH	632	PUSH(632) TB S TB: 선두 스택 어드레스 S: 소스 워드	지정한 스택에 1워드의 데이터를 씁니다. 	출력	필요
후입 선출	LIFO @LIFO	634	LIFO(634) TB D TB: 선두 스택 어드레스 D: 저장처 워드	지정한 스택에 쓰인 마지막 워드의 데이터(스택 내 가장 새로운 데이터)를 읽습니다. 	출력	필요
선입 선출	FIFO @FIFO	633	FIFO(633) TB D TB: 선두 스택 어드레스 D: 저장처 워드	지정한 스택에 쓰인 첫 워드의 데이터(스택 내 가장 오래된 데이터)를 읽습니다. 	출력	필요
레코드 테이블 정의	DIM @DIM	631	DIM(631) N LR NR TB N: 테이블 번호 LR: 각 레코드의 길이 NR: 레코드 수 TB: 선두 테이블 워드	각 레코드의 길이와 레코드 수를 선언하여 레코드 테이블을 정의합니다. 최대 16개의 레코드 테이블을 정의할 수 있습니다. 	출력	필요

AI -For Reference Only-

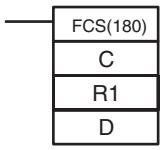
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
레코드 위치 설정	SETR @SETR	635	SETR(635) N R D N: 테이블 번호 R: 레코드 번호 D: 저장처 인덱스 레지스터	지정한 레코드의 위치(레코드 선두의 내부 I/O 메모리 어드레스)를 지정한 인덱스 레지스터에 씁니다. 	출력	필요
레코드 번호 취득	GETR @GETR	636	GETR(636) N IR D N: 테이블 번호 IR: 인덱스 레지스터 D: 저장처 워드	지정한 인덱스 레지스터에 들어 있는 내부 I/O 메모리 어드레스에 있는 레코드의 레코드 번호를 반환합니다. 	출력	필요
데이터 검색	SRCH @SRCH	181	SRCH(181) C R1 Cd C: 선두 제어 워드 R1: 범위의 선두 워드 Cd: 비교 데이터	워드 범위 내에서 1워드의 데이터를 검색합니다. 	출력	필요
바이트 스왑	SWAP BYTES	637	SWAP(637) N R1 N: 워드 수 R1: 범위의 선두 워드	범위 내 모든 워드에서 최상위 바이트와 최하위 바이트를 서로 바꿉니다. 	출력	필요
최댓값 검색	MAX @MAX	182	MAX(182) C R1 D C: 선두 제어 워드 R1: 범위의 선두 워드 D: 저장처 워드	범위 내 최댓값을 구합니다. 	출력	필요
더블 최댓값 검색	MAXL @MAXL	174	MAXL(174) C R1 D C: 선두 제어 워드 R1: 범위의 선두 워드 D: 선두 저장처 워드	C로 지정한 개수의 데이터 항목을 R1으로 지정한 범위의 선두 워드부터 시작하는 더블 워드 테이블 데이터로 취급하여 그 테이블의 최댓값을 D+1과 D에 출력합니다.	출력	필요

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
부동소수점 최댓값 검색	MAXF @MAXF	176	MAXF(176) C R1 D C: 선두 제어 워드 R1: 범위의 선두 워드 D: 선두 저장처 워드	C로 지정한 개수의 데이터 항목을 R1으로 지정한 범위의 선두 워드부터 시작하는 단정밀도 부동소수점 데이터(더블 워드 데이터) 테이블로 취급하여 그 테이블의 최댓값을 D+1과 D에 출력합니다.	출력	필요
배정밀도 부동소수점 최댓값 검색	MAXD @MAXD	178	MAXD(178) C R1 D C: 선두 제어 워드 R1: 범위의 선두 워드 D: 선두 저장처 워드	C로 지정한 개수의 데이터 항목을 R1으로 지정한 범위의 선두 워드부터 시작하는 배정밀도 부동소수점 데이터(더블 워드 데이터) 테이블로 취급하여 그 테이블의 최댓값을 D+1과 D에 출력합니다.	출력	필요
최솟값 검색	MIN @MIN	183	MIN(183) C R1 D C: 선두 제어 워드 R1: 범위의 선두 워드 D: 저장처 워드	범위 내 최솟값을 구합니다.  	출력	필요
더블 최솟값 검색	MINL @MINL	175	MINL(175) C R1 D C: 선두 제어 워드 R1: 범위의 선두 워드 D: 선두 저장처 워드	C로 지정한 개수의 데이터 항목을 R1으로 지정한 범위의 선두 워드부터 시작하는 더블 워드 테이블 데이터로 취급하여 그 테이블의 최솟값을 D+1과 D에 출력합니다.	출력	필요
부동소수점 최솟값 검색	MINF @MINF	177	MINF(177) C R1 D C: 선두 제어 워드 R1: 범위의 선두 워드 D: 선두 저장처 워드	C로 지정한 개수의 데이터 항목을 R1으로 지정한 범위의 선두 워드부터 시작하는 단정밀도 부동소수점 데이터(더블 워드 데이터) 테이블로 취급하여 그 테이블의 최솟값을 D+1과 D에 출력합니다.	출력	필요
배정밀도 부동소수점 최솟값 검색	MIND @MIND	179	MIND(179) C R1 D C: 선두 제어 워드 R1: 범위의 선두 워드 D: 선두 저장처 워드	C로 지정한 개수의 데이터 항목을 R1으로 지정한 범위의 선두 워드부터 시작하는 배정밀도 부동소수점 데이터(더블 워드 데이터) 테이블로 취급하여 그 테이블의 최솟값을 D+1과 D에 출력합니다.	출력	필요

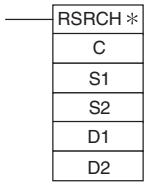
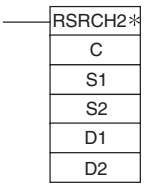
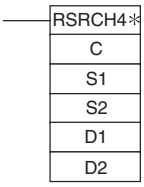
번역본
-Reference Only-

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
합계	SUM @SUM	184	SUM(184) C R1 D C: 선두 제어 워드 R1: 범위의 선두 워드 D: 선두 저장처 워드	범위 내 바이트 또는 워드를 가산하여 그 결과를 2워드로 출력합니다. 	출력	필요
스택 사이즈 읽기	SNUM @SNUM	638	SNUM(638) TB D TB: 선두 스택 어드레스 D: 저장처 워드	지정한 스택의 스택 데이터 양(워드 수)을 셉니다.	출력	필요
스택 데이터 읽기	SREAD @SREAD	639	SREAD(639) TB C D TB: 선두 스택 어드레스 C: 오프셋 값 D: 저장처 워드	스택 내 지정한 데이터 요소에서 데이터를 읽습니다. 오프셋 값은 원하는 데이터 요소의 위치(현재 포인터 위치보다 몇 개 앞의 데이터 요소인지)를 나타냅니다.	출력	필요
스택 데이터 덮어쓰기	SWRIT @SWRIT	640	SWRIT(640) TB C S TB: 선두 스택 어드레스 C: 오프셋 값 S: 소스 데이터	스택 내 지정한 데이터 요소에 소스 데이터를 씁니다(기존 데이터를 덮어씁니다). 오프셋 값은 원하는 데이터 요소의 위치(현재 포인터 위치보다 몇 개 앞의 데이터 요소인지)를 나타냅니다.	출력	필요
스택 데이터 삽입	SINS @SINS	641	SINS(641) TB C S TB: 선두 스택 어드레스 C: 오프셋 값 S: 소스 데이터	스택 내 지정한 위치에 소스 데이터를 삽입하고 스택 내 나머지 데이터를 아래쪽으로 시프트합니다. 오프셋 값은 삽입 위치(현재 포인터 위치보다 몇 개 앞의 데이터 요소인지)를 나타냅니다.	출력	필요
스택 데이터 삭제	SDEL @SDEL	642	SDEL(642) TB C D TB: 선두 스택 어드레스 C: 오프셋 값 D: 저장처 워드	스택 내 지정한 위치의 데이터 요소를 삭제하고 스택 내 나머지 데이터를 위쪽으로 시프트합니다. 오프셋 값은 삭제 위치(현재 포인터 위치보다 몇 개 앞의 데이터 요소인지)를 나타냅니다.	출력	필요

AI 번역본
-For Reference Only-

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
프레임 체크섬	FCS @FCS	180	 C: 선두 제어 워드 R1: 범위의 선두 워드 D: 선두 저장처 워드	지정한 범위의 FCS 값을 연산하여 그 값을 ASCII로 출력합니다.	출력	필요

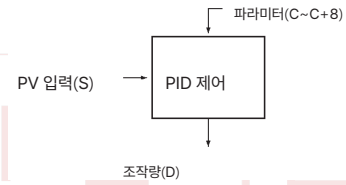
A-1-16 트래킹 명령어

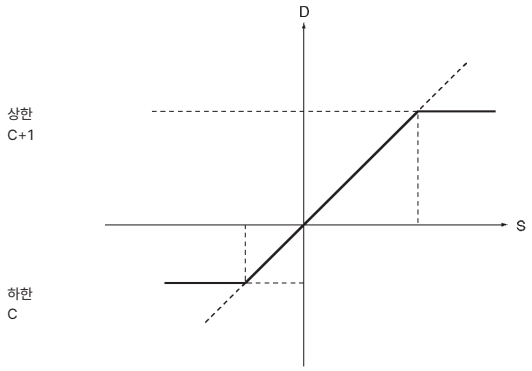
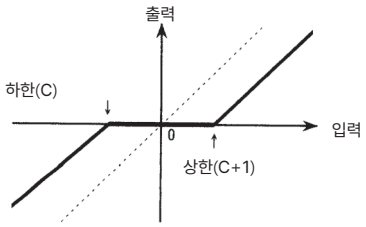
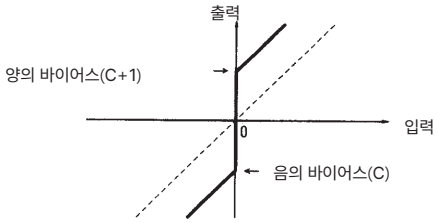
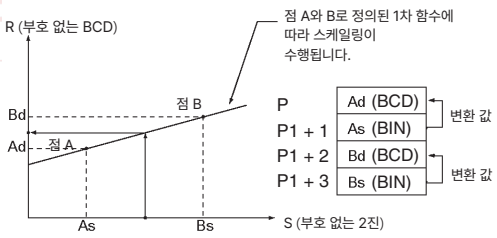
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
부호 없는 1워드 레코드 검색 명령어	RSRCH + <, <=, =, >, >= @RSRCH + <, <=, =, >, >=	360 (<), 361 (<=), 362 (=), 363 (>), 364 (>=)	 C: 선두 제어 워드 S1: 검색할 첫 레코드의 선두 워드 S2: 검색 데이터 D1: 선두 저장처 워드 D2: 저장처 인덱스 레지스터	부호 없는 1워드 레코드 검색 명령어는 S1으로 지정한 테이블부터 시작하여 S2로 지정한 데이터(1워드)를 검색합니다. 지정한 조건에 일치하는 레코드를 찾으면 그 레코드 번호와 데이터가 D1 이후로 출력됩니다. D2에 인덱스 레지스터를 지정하면 일치한 레코드의 어드레스가 그 인덱스 레지스터로 출력됩니다. 인덱스 레지스터 출력을 사용하지 않으려면 D2에 #00000000을 설정하십시오.	출력	필요
부호 없는 2워드 레코드 검색 명령어	RSRCH2 + <, <=, =, >, >= @RSRCH2 + <, <=, =, >, >=	370 (<), 371 (<=), 372 (=), 373 (>), 374 (>=)	 C: 선두 제어 워드 S1: 검색할 첫 레코드의 선두 워드 S2: 검색 데이터의 선두 워드 D1: 선두 저장처 워드 D2: 저장처 인덱스 레지스터	부호 없는 2워드 레코드 검색 명령어는 S1으로 지정한 테이블부터 시작하여 S2로 지정한 데이터(2워드)를 검색합니다. 지정한 조건에 일치하는 레코드를 찾으면 그 레코드 번호와 데이터가 D1 이후로 출력됩니다. D2에 인덱스 레지스터를 지정하면 일치한 레코드의 어드레스가 그 인덱스 레지스터로 출력됩니다. 인덱스 레지스터 출력을 사용하지 않으려면 D2에 #00000000을 설정하십시오.	출력	필요
부호 없는 4워드 레코드 검색 명령어	RSRCH4 + <, <=, =, >, >= @RSRCH4 + <, <=, =, >, >=	380 (<), 381 (<=), 382 (=), 383 (>), 384 (>=)	 C: 선두 제어 워드 S1: 검색할 첫 레코드의 선두 워드 S2: 검색 데이터 D1: 선두 저장처 워드 D2: 저장처 인덱스 레지스터	부호 없는 4워드 레코드 검색 명령어는 S1으로 지정한 테이블부터 시작하여 S2로 지정한 데이터(4워드)를 검색합니다. 지정한 조건에 일치하는 레코드를 찾으면 그 레코드 번호와 데이터가 D1 이후로 출력됩니다. D2에 인덱스 레지스터를 지정하면 일치한 레코드의 어드레스가 그 인덱스 레지스터로 출력됩니다. 인덱스 레지스터 출력을 사용하지 않으려면 D2에 #00000000을 설정하십시오.	출력	필요

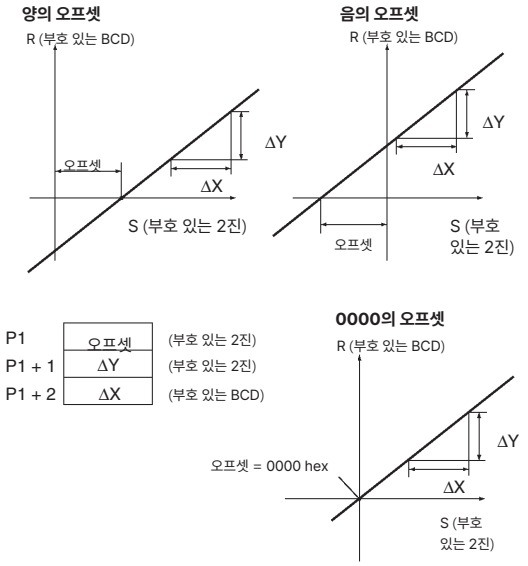
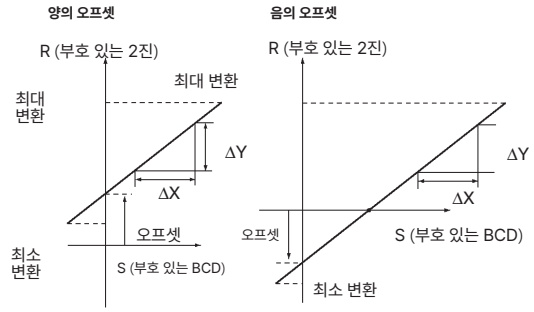
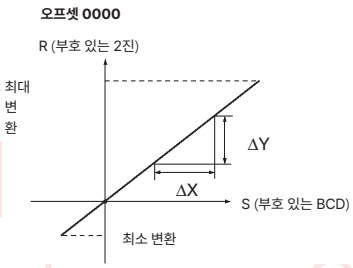
-Reference Only-

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
부호 없는 1워드 레코드 정렬	RSORT @RSORT	203	RSORT(203) C S D1 D2 C: 선두 제어 워드 S: 정렬할 첫 레코드의 선두 워드 D1: 정렬 결과의 선두 워드 D2: 저장처 인덱스 레지스터	S로 지정한 테이블의 레코드(1워드)를 제어 워드에 따라 정렬합니다.	출력	필요
부호 없는 2워드 레코드 정렬	RSORT2 @RSORT2	204	RSORT2(204) C S D1 D2 C: 선두 제어 워드 S: 정렬할 첫 레코드의 선두 워드 D1: 정렬 결과의 선두 워드 D2: 저장처 인덱스 레지스터	S로 지정한 테이블의 레코드(2워드)를 제어 워드에 따라 정렬합니다.	출력	필요
부호 없는 4워드 레코드 정렬	RSORT4 @RSORT4	205	RSORT4(205) C S D1 D2 C: 선두 제어 워드 S: 정렬할 첫 레코드의 선두 워드 D1: 정렬 결과의 선두 워드 D2: 저장처 인덱스 레지스터	S로 지정한 테이블의 레코드(4워드)를 제어 워드에 따라 정렬합니다.	출력	필요

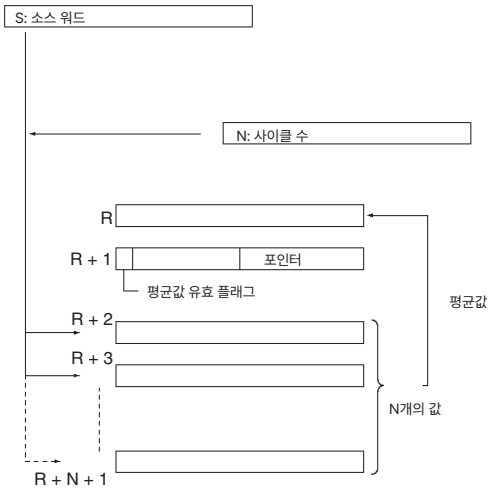
A-1-17 데이터 제어 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
PID 제어	PID	190	PID(190) S C D S: 입력 워드 C: 선두 파라미터 워드 D: 출력 워드	지정한 파라미터에 따라 PID 제어를 실행합니다. 	출력	필요
오토튜닝 포함 PID 제어	PIDAT	191	PIDAT(191) S C D S: 입력 워드 C: 선두 파라미터 워드 D: 출력 워드	지정한 파라미터에 따라 PID 제어를 실행합니다. PID 상수는 PIDAT(191)로 오토튜닝할 수 있습니다.	출력	필요

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
리미트 제어	LMT @LMT	680	LMT(680) S C D S: 입력 워드 C: 선두 리미트 워드 D: 출력 워드	입력 데이터가 상한과 하한 범위 내에 있는지 여부에 따라 출력 데이터를 제어합니다. 	출력	필요
데드 밴드 제어	BAND @BAND	681	BAND(681) S C D S: 입력 워드 C: 선두 리미트 워드 D: 출력 워드	입력 데이터가 데드 밴드 범위 내에 있는지 여부에 따라 출력 데이터를 제어합니다. 	출력	필요
데드 존 제어	ZONE @ZONE	682	ZONE(682) S C D S: 입력 워드 C: 선두 리미트 워드 D: 출력 워드	입력 데이터에 지정한 바이어스를 가산하여 그 결과를 출력합니다. 	출력	필요
시간 비례 출력	TPO	685	TPO (685) S C R S: 입력 워드 C: 선두 파라미터 워드 R: 펄스 출력 비트	지정한 워드에서 듀티비 또는 조작량을 입력하고, 지정한 파라미터에 따라 듀티비를 시간 비례 출력으로 변환하여 그 결과를 지정한 출력에서 출력합니다.	출력	필요
스케일링	SCL @SCL	194	SCL(194) S P1 R S: 소스 워드 P1: 선두 파라미터 워드 R: 결과 워드	지정한 1차 함수에 따라 부호 없는 바이너리 데이터를 부호 없는 BCD 데이터로 변환합니다. 	출력	필요

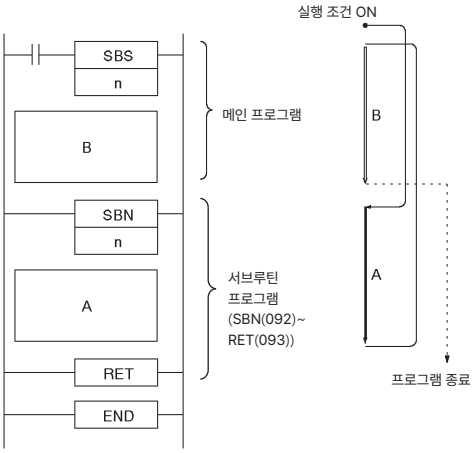
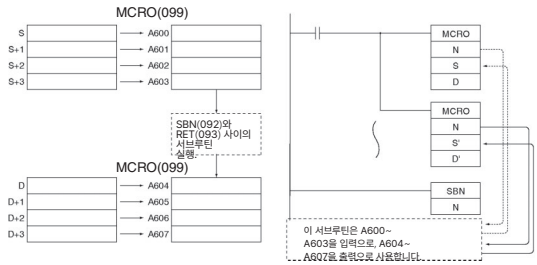
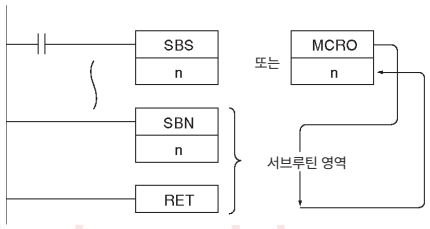
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
스케일링 2	SCL2 @SCL2	486	SCL2(486) S P1 R S: 소스 워드 P1: 선두 파라미터 워드 R: 결과 워드	지정한 1차 함수에 따라 부호 있는 바이너리 데이터를 부호 있는 BCD 데이터로 변환합니다. 1차 함수를 정의할 때 오프셋을 입력할 수 있습니다.  P1 오프셋 (부호 있는 2진) P1 + 1 ΔY (부호 있는 2진) P1 + 2 ΔX (부호 있는 BCD) 	출력	필요
스케일링 3	SCL3 @SCL3	487	SCL3(487) S P1 R S: 소스 워드 P1: 선두 파라미터 워드 R: 결과 워드	지정한 1차 함수에 따라 부호 있는 BCD 데이터를 부호 있는 2진 데이터로 변환합니다. 1차 함수를 정의할 때 오프셋을 입력할 수 있습니다.  	출력	필요

AI
-For Reference Only-

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
평균	AVG	195	AVG(195) S N R S: 소스 워드 N: 사이클 수 R: 결과 워드	지정한 사이클 수 동안 입력 워드의 평균값을 산출합니다. 	출력	필요

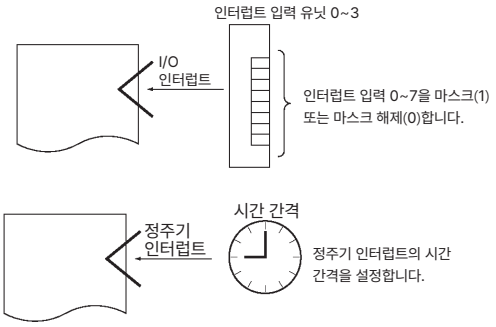
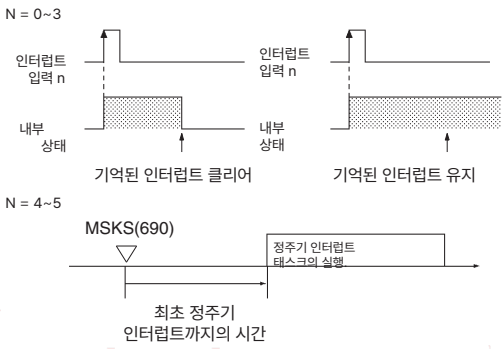
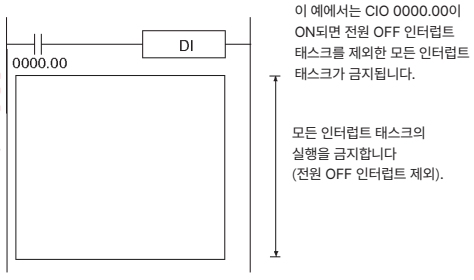
AI 번역본
-For Reference Only-

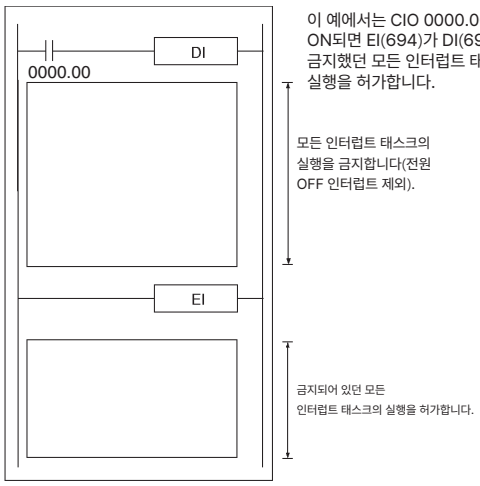
A-1-18 서브루틴 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
서브루틴 콜	SBS @SBS	091	SBS(091) N N: 서브루틴 번호	지정한 서브루틴 번호의 서브루틴을 호출하여 그 프로그램을 실행합니다. 	출력	필요
매크로	MCRO @MCRO	099	MCRO(099) N S D N: 서브루틴 번호 S: 선두 입력 파라미터 워드 D: 선두 출력 파라미터 워드	지정한 서브루틴 번호의 서브루틴을 호출하여, S~S+3의 입력 파라미터와 D~D+3의 출력 파라미터를 사용해 그 프로그램을 실행합니다. 	출력	필요
서브루틴 엔트리	SBN	092	SBN(092) N N: 서브루틴 번호	지정한 서브루틴 번호를 가진 서브루틴 프로그램의 시작을 나타냅니다. 	출력	필요
서브루틴 리턴	RET	093	RET(093)	서브루틴 프로그램의 끝을 나타냅니다.	출력	필요 없음
글로벌 서브루틴 콜	GSBS @GSBS	750	GSBS(750) N N: 서브루틴 번호	지정한 서브루틴 번호의 서브루틴을 호출하여 그 프로그램을 실행합니다.	출력	필요
글로벌 서브루틴 엔트리	GSBN	751	GSBN(751) N N: 서브루틴 번호	지정한 서브루틴 번호를 가진 서브루틴 프로그램의 시작을 나타냅니다.	출력	필요 없음
글로벌 서브루틴 리턴	GRET	752	GRET(752)	서브루틴 프로그램의 끝을 나타냅니다.	출력	필요 없음

-for Reference Only-

A-1-19 인터럽트 제어 명령어

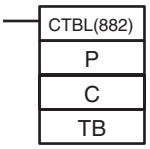
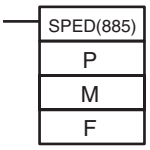
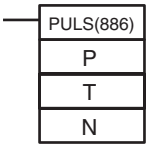
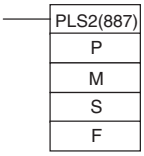
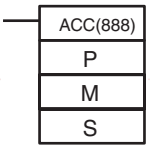
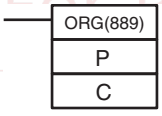
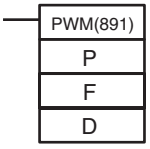
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
인터럽트 마스크 설정	MSKS @MSKS	690	MSKS(690) N C N: 인터럽트 번호 C: 제어 데이터	I/O 인터럽트, 입력 인터럽트 또는 정주기 인터럽트의 인터럽트 처리를 설정합니다. I/O 인터럽트, 입력 인터럽트, 정주기 인터럽트는 PLC의 전원을 처음 ON 했을 때 마스크(무효) 상태입니다. MSKS(690)을 사용하여 인터럽트의 마스크를 해제 또는 설정하고 정주기 인터럽트의 시간 간격을 설정할 수 있습니다. 	출력	필요
인터럽트 마스크 읽기	MSKR @MSKR	692	MSKR(692) N D N: 인터럽트 번호 D: 저장처 워드	MSKS(690)으로 설정한 현재의 인터럽트 처리 설정을 읽습니다.	출력	필요
인터럽트 클리어	CLI @CLI	691	CLI(691) N C N: 인터럽트 번호 C: 제어 데이터	I/O 인터럽트에서는 기억된 인터럽트 입력을 클리어 또는 유지하고, 정주기 인터럽트에서는 최초 정주기 인터럽트까지의 시간을 설정합니다. 	출력	필요
인터럽트 금지	DI @DI	693	DI(693)	전원 OFF 인터럽트를 제외한 모든 인터럽트 태스크의 실행을 금지합니다. 	출력	필요

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
인터럽트 허가	EI	694	EI(694)	DI(693)으로 금지했던 모든 인터럽트 태스크의 실행을 허가합니다. 	출력	필요 없음

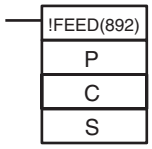
A-1-20 고속 카운터/펄스 출력 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
모드 제어	INI @INI	880	INI(880) P C NV P: 포트 지정자 C: 제어 데이터 NV: 새 값이 들어 있는 선두 워드	INI(880)은 비교표의 비교를 개시 · 정지하거나, 고속 카운터의 현재값(PV)을 변경하거나, 카운터 모드인 입력 인터럽트의 현재값을 변경하거나, 링 카운터의 최대값을 변경하거나, 펄스 출력의 현재값을 변경(예: 원점을 확립하기 위해 0으로)하거나, 펄스 출력을 정지하거나, 원점 서치/복귀 설정을 변경하는 데 사용합니다. (INI(880)은 CJ2M/CJ3M에서만 지원됩니다.)	출력	필요
고속 카운터 현재값 읽기	PRV @PRV	881	PRV(881) P C D P: 포트 지정자 C: 제어 데이터 D: 선두 저장처 워드	PRV(881)은 고속 카운터, 펄스 출력 또는 입력 인터럽트(카운터 모드)의 현재값(PV)을 읽는 데 사용합니다. (PRV(881)은 CJ2M/CJ3M에서만 지원됩니다.)	출력	필요
카운터 주파수 변환	PRV2 @PRV2	883	PRV2(883) C C2 D C: 제어 데이터 C2: 회전당 펄스 수 D: 선두 저장처 워드	고속 카운터로부터 입력된 펄스 주파수를 읽어, 주파수를 회전 속도(회전수)로 변환하거나 카운터의 현재값을 총 회전수로 변환합니다. 결과는 8자리 16진수로 저장처 워드에 출력됩니다. (PRV2(883)은 CJ2M/CJ3M에서만 지원됩니다.)	출력	필요

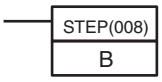
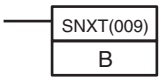
샘플 번역본

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
비교표 등록	CTBL @CTBL	882	 P: 포트 지정자 C: 제어 데이터 **TB: 선두** 비교표 워드	CTBL(882)는 고속 카운터의 현재값(PV)에 대해 목표값 비교 또는 범위 비교를 수행하는 데 사용합니다. (CTBL(882)는 CJ2M/CJ3M에서만 지원됩니다.)	출력	필요
속도 출력	SPED @SPED	885	 P: 포트 지정자 M: 출력 모드 F: 선두 펄스 주파수 워드	SPED(885)는 주파수를 지정하여 가속 · 감속 없이 펄스 출력을 수행하는 데 사용합니다. (SPED(885)는 CJ2M/CJ3M에서만 지원됩니다.)	출력	필요
펄스 설정	PULS @PULS	886	 P: 포트 지정자 T: 펄스 종별 N: 펄스 수	PULS(886)은 펄스 출력의 펄스 수를 설정하는 데 사용합니다. (PULS(886)은 CJ2M/CJ3M에서만 지원됩니다.)	출력	필요
펄스 출력	PLS2 @PLS2	887	 P: 포트 지정자 M: 출력 모드 S: 설정 테이블 선두 워드 F: 시작 주파수 선두 워드	PLS2(887)는 펄스 주파수와 가감속 비율을 설정하고, 서로 다른 가감속 비율로 가감속하면서 펄스 출력을 수행하는 데 사용합니다. (PLS2(887)는 CJ2M/CJ3M에서만 지원됩니다.)	출력	필요
가속 제어	ACC @ACC	888	 P: 포트 지정자 M: 출력 모드 S: 설정 테이블 선두 워드	ACC(888)는 펄스 주파수와 가감속 비율을 설정하고, 가감속(동일한 가감속 비율)하면서 펄스 출력을 수행하는 데 사용합니다. (ACC(888)는 CJ2M/CJ3M에서만 지원됩니다.)	출력	필요
원점 서치	ORG @ORG	889	 P: 포트 지정자 C: 제어 데이터	ORG(889)는 원점 서치와 원점 복귀를 수행하는 데 사용합니다. (ORG(889)는 CJ2M/CJ3M에서만 지원됩니다.)	출력	필요
가변 듀티비 펄스	PWM @PWM	891	 P: 포트 지정자 F: 주파수 D: 듀티비	PWM(891)은 가변 듀티비 펄스를 출력하는 데 사용합니다. (PWM(891)은 CJ2M/CJ3M에서만 지원됩니다.)	출력	필요

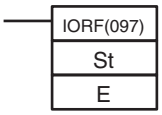
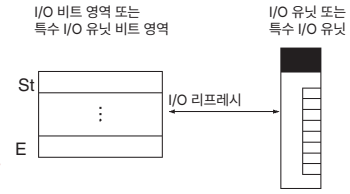
AI 번역본
For Reference Only

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
인터럽트 피딩	IFEED @IFEED	892	 P: 포트 지정자 C: 제어 데이터 S: 설정 테이블 선두 워드	IFEED(892)는 입력 인터럽트를 트리거로 하여 속도 제어에서 위치 제어로 전환하고 지정한 펄스 수만큼 이동합니다. (IFEED(892)는 CJ2M/CJ3M에서만 지원됩니다.)	출력	필요

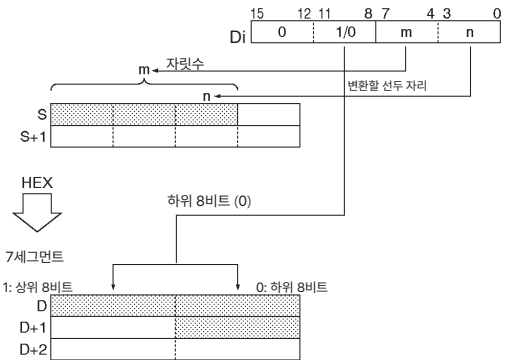
A-1-21 스텝 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
스텝 정의	STEP	008	 B: 비트	STEP(008)은 그 위치와 제어 비트 지정 여부에 따라 다음 2가지로 동작합니다. (1) 특정 스텝을 개시합니다. (2) 스텝 프로그래밍 영역(즉, 스텝 실행)을 종료합니다. 스텝 프로그램 영역은 최초의 STEP(008) 명령어(항상 제어 비트를 취함)부터 마지막 STEP(008) 명령어(제어 비트를 취하지 않음)까지입니다.	출력	필요
스텝 개시	SNXT	009	 B: 비트	SNXT(009)는 다음 세 가지로 사용합니다: (1) 스텝 프로그래밍 실행을 개시합니다. (2) 다음 스텝 제어 비트로 진행합니다. (3) 스텝 프로그래밍 실행을 종료합니다.	출력	필요

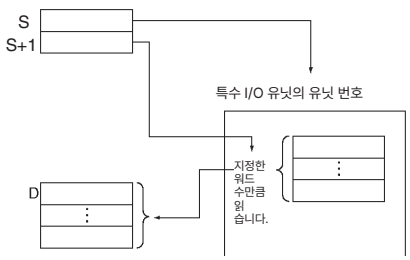
A-1-22 기본 I/O 유닛 명령어

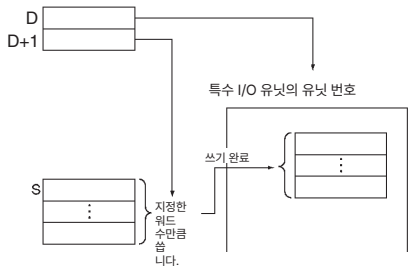
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
I/O 리프레시	IORF @IORF	097	 St: 선두 워드 E: 최종 워드	지정한 I/O 워드를 리프레시합니다. 	출력	필요
특수 I/O 유닛 I/O 리프레시	FIORF @FIORF	225	 N: 유닛 번호	지정한 유닛 번호의 특수 I/O 유닛에 할당된 I/O 워드를 즉시 리프레시합니다.	출력	필요
CPU 버스 유닛 I/O 리프레시	DLNK @DLNK	226	 N: 유닛 번호	지정한 유닛 번호의 CPU 버스 유닛에 할당된 I/O 워드를 즉시 리프레시합니다.	출력	필요

OMRON 전역본 -For Reference Only-

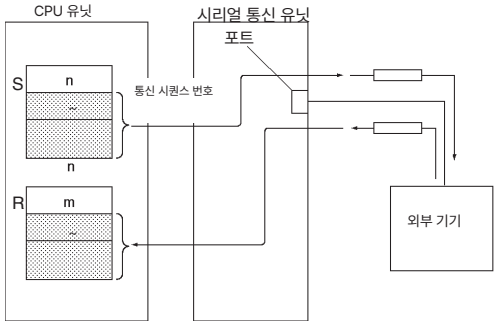
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
7세그먼트 디코더	SDEC @SDEC	078	SDEC(078) S Di D S: 소스 워드 Di: 자릿수 지정자 D: 선두 저장처 워드	지정한 자리의 16진 내용을 8비트 7세그먼트 표시 코드로 변환하여, 지정한 저장처 워드의 상위 또는 하위 8비트에 저장합니다. 	출력	필요
디지털 스위치 입력	DSW	210	DSW (210) I O D C1 C2 **I**: 데이터 입력 워드 (D0~D3) O: 출력 워드 **D**: 선두 결과 워드 **C1**: 자릿수 C2: 시스템 워드	입력 유닛 또는 출력 유닛에 접속된 외부 디지털 스위치(또는 썸휠 스위치)에 설정된 값을 읽어, 4자리 또는 8자리 BCD 데이터를 지정한 워드에 저장합니다.	출력	필요
텐키 입력	TKY	211	TKY (211) I D1 D2 **I**: 데이터 입력 워드 D1: 선두 레지스터 워드 D2: 키 입력 워드	입력 유닛에 접속된 텐키 키패드로부터 수치 데이터를 읽어, 최대 8자리의 BCD 데이터를 지정한 워드에 저장합니다.	출력	필요
16진 키 입력	HKY	212	HKY (212) I O D C **I**: 데이터 입력 워드 O: 출력 워드 D: 선두 레지스터 워드 C: 시스템 워드	입력 유닛 및 출력 유닛에 접속된 16진 키패드로부터 수치 데이터를 읽어, 최대 8자리의 16진 데이터를 지정한 워드에 저장합니다.	출력	필요
매트릭스 입력	MTR	213	MTR (213) I O D C **I**: 데이터 입력 워드 O: 출력 워드 D: 선두 저장처 워드 C: 시스템 워드	입력 유닛 및 출력 유닛에 접속된 8 × 8 매트릭스로부터 최대 64개 신호를 입력하고(입력 8점과 출력 8점 사용), 그 64비트 데이터를 4개의 저장처 워드에 저장합니다.	출력	필요

번역본
-Reference Only-

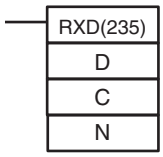
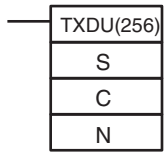
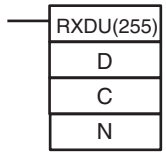
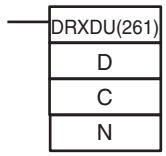
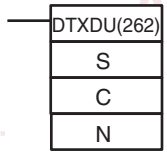
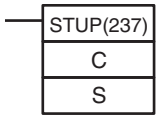
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
7세그먼트 표시 출력	7SEG	214	7SEG(214) S O C D S: 선두 소스 워드 O: 출력 워드 C: 제어 데이터 D: 시스템 워드	소스 데이터(4자리 또는 8자리 BCD)를 7세그먼트 표시 데이터로 변환하여, 그 데이터를 지정한 출력 워드에 출력합니다.	출력	필요
아날로그 입력 다이렉트 변환 (CJ1W-AD042 용)	AIDC @AIDC	216	AIDC(216) N A N: 유닛 번호 A: 아날로그 입력 번호	다이렉트 변환 모드의 CJ1W-AD042 아날로그 입력 유닛으로부터 지정한 아날로그 입력 번호의 입력 변환값을 읽습니다.	출력	필요
아날로그 출력 다이렉트 변환 (CJ1W-DA042V 용)	AODC @AODC	217	AODC(217) N A N: 유닛 번호 A: 아날로그 출력 번호	지정한 아날로그 출력 번호의 출력 설정값을 다이렉트 변환 모드의 CJ1W-DA042V 아날로그 출력 유닛에 출력합니다.	출력	필요
PCU 고속 위치 결정 (CJ1W-NC□□4 또는 CJ1W-NC□81 전용)	NCDMV @NCDMV	218	NCDMV(218) C A C: 제어 데이터 A: 다이렉트 운전 커맨드 영역 선두 워드	NCDMV(218)는 CJ1W-NC□□4 또는 CJ1W-NC□81 위치 제어 유닛의 지정한 축에 대해 절대 또는 상대 고속 포인트 투 포인트 위치 결정을 개시합니다.	출력	필요
PCU 위치 결정 트리거 (CJ1W-NC□81 전용)	NCDTR @NCDTR	219	NCDTR(219) C C: 제어 데이터	NCDTR(219)는 시퀀스의 개시 조건이 NCDTR(219)로부터의 커맨드 대기일 때, CJ1W-NC□81 위치 제어 유닛의 메모리 운전 시퀀스를 개시하는 데 사용합니다.	출력	필요
인텔리전트 I/O 읽기	IORD @IORD	222	IORD(222) C S D C: 제어 데이터 S: 전송원 및 워드 수 D: 전송처 및 워드 수	특수 I/O 유닛 또는 CPU 버스 유닛의 메모리 영역 내용을 읽습니다(주 참조). 	출력	필요

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
인텔리전트 I/O 쓰기	IOWR @IOWR	223	IOWR(223) C S D C: 제어 데이터 S: 전송원 및 워드 수 D: 전송처 및 워드 수	CPU 유닛의 I/O 메모리 영역 내용을 특수 I/O 유닛 또는 CPU 버스 유닛에 출력합니다(주 참조). 	출력	필요

A-1-23 시리얼 통신 명령어

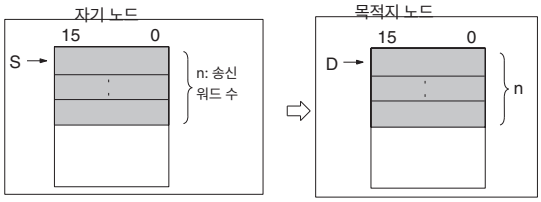
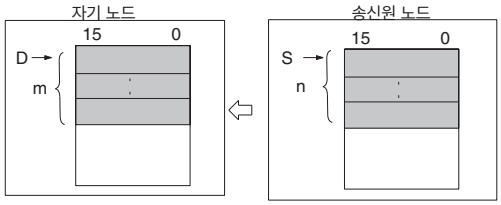
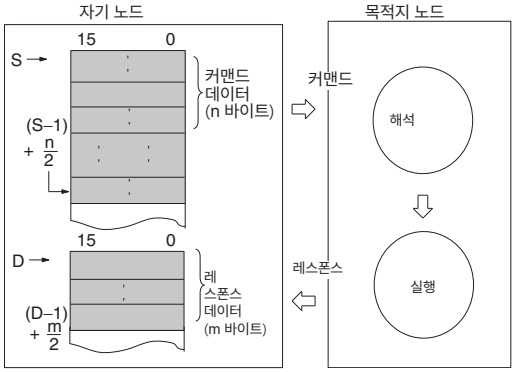
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
프로토콜 매크로	PMCR @PMCR	260	PMCR(260) C1 C2 S R C1: 제어 워드 1 C2: 제어 워드 2 S: 선두 송신 워드 R: 선두 수신 워드	시리얼 통신 보드(CS 시리즈 전용) 또는 시리얼 통신 유닛에 등록된 통신 시퀀스를 호출하여 실행합니다. 	출력	필요
프로토콜 매크로 2	PMCR2 @PMCR2	264	PMCR2(264) C1 C2 S R I C1: 제어 워드 1 C2: 제어 워드 2 S: 선두 송신 워드 R: 선두 수신 워드 I: 선두 통신 정보 워드	PMCR2 명령어는 최대 64개를 동시에 사용할 수 있습니다. 그 밖의 점은 PMCR과 같습니다.	출력	필요
송신	TXD @TXD	236	TXD(236) S C N S: 선두 소스 워드 C: 제어 워드 N: 바이트 수 0000~0100 hex (10진수 0~256)	PLC 설정에서 무순서 모드용으로 지정한 스타트 코드와 엔드 코드에 따라, CPU 유닛에 내장된 RS-232C 포트(무순서 모드) 또는 유닛 버전 1.2 이후의 시리얼 통신 보드나 시리얼 통신 유닛의 시리얼 포트(무순서 모드)로부터 지정한 바이트 수의 데이터를 변환하지 않고 출력합니다.	출력	필요

번역본

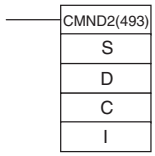
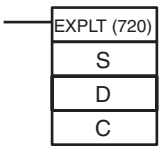
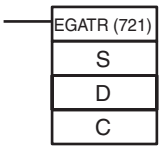
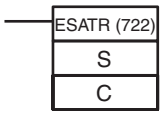
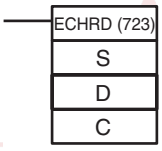
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
수신	RXD @RXD	235	 D: 선두 저장처 워드 C: 제어 워드 N: 저장할 바이트 수 0000~0100 hex (10진수 0~256)	PLC 설정에서 무순서 모드용으로 지정한 스타트 코드와 엔드 코드에 따라, CPU 유닛에 내장된 RS-232C 포트(무순서 모드) 또는 유닛 버전 1.2 이후의 시리얼 통신 보드나 시리얼 통신 유닛의 시리얼 포트(무순서 모드)로부터 지정한 선두 워드를 시작으로 지정한 바이트 수의 데이터를 읽습니다.	출력	필요
시리얼 통신 유닛 경유 송신	TXDU @TXDU	256	 S: 선두 소스 워드 C: 선두 제어 워드 N: 바이트 수 (0000~0256 BCD)	유닛 버전 1.2 이후의 시리얼 통신 유닛의 시리얼 포트로부터 지정한 바이트 수의 데이터를 변환하지 않고 출력합니다. 데이터는 할당된 DM 영역에 지정된 스타트 코드와 엔드 코드(있는 경우)를 붙여 무순서 모드로 출력됩니다.	출력	필요
시리얼 통신 유닛 경유 수신	RXDU @RXDU	255	 D: 선두 저장처 워드 C: 선두 제어 워드 N: 저장할 바이트 수 (0000~0100 hex)	유닛 버전 1.2 이후의 시리얼 통신 유닛의 시리얼 포트로부터 지정한 선두 워드를 시작으로 지정한 바이트 수의 데이터를 읽습니다. 데이터는 할당된 DM 셋업 영역에 지정된 스타트 코드와 엔드 코드(있는 경우)를 붙여 무순서 모드로 읽힙니다.	출력	필요
시리얼 통신 유닛/ 시리얼 포트 경유 다이렉트 수신 (CJ1W-SCU22, CJ1W-SCU32 또는 CJ1W-SCU42 전용)	DRXDU @DRXDU	261	 D: 선두 저장처 워드 C: 선두 제어 워드 N: 바이트 수 0000~0100 hex (0~256)	CJ1W-SCU22/SCU32/SCU42 시리얼 통신 유닛의 시리얼 포트로부터 지정한 바이트 수의 데이터를 지정한 선두 워드부터 CPU 유닛 메모리에 읽어 들입니다. 데이터는 할당된 DM 셋업 영역에 지정된 스타트 코드와 엔드 코드(있는 경우)를 붙여 무순서 모드로 읽힙니다. 이 명령어는 고속 데이터 수신을 실현하기 위해, 명령어가 실행되는 즉시 시리얼 통신 유닛으로부터 데이터를 읽습니다.	출력	필요
시리얼 통신 유닛/ 시리얼 포트 경유 다이렉트 송신 (CJ1W-SCU22, CJ1W-SCU32 또는 CJ1W-SCU42 전용)	DTXDU @DTXDI	262	 S: 선두 소스 워드 C: 선두 제어 워드 N: 바이트 수 0000~0100 hex (0~256)	CJ1WSCU22/SCU32/SCU42 시리얼 통신 유닛의 시리얼 포트로부터 지정한 바이트 수의 데이터를 출력합니다. 데이터는 지정한 선두 워드부터, 할당된 DM 셋업 영역에 지정된 스타트 코드와 엔드 코드(있는 경우)를 붙여 무순서 모드로 출력됩니다. 이 명령어는 고속 데이터 송신을 실현하기 위해, 명령어가 실행되는 즉시 시리얼 통신 유닛에 데이터를 보냅니다.	출력	필요
시리얼 포트 설정 변경	STUP @STUP	237	 C: 제어 워드 (포트) S: 선두 소스 워드	CPU 유닛, 시리얼 통신 유닛(CPU 버스 유닛) 또는 시리얼 통신 보드에 있는 시리얼 포트의 통신 파라미터를 변경합니다. 이에 따라 STUP(237)은 PLC 운전 중에 프로토콜 모드를 변경할 수 있게 합니다.	출력	필요

Reference Only

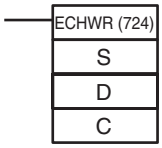
A-1-24 네트워크 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
네트워크 송신	SEND @SEND	090	SEND(090) S D C S: 선두 소스 워드 D: 선두 저장처 워드 C: 선두 제어 워드	네트워크 내의 노드에 데이터를 송신합니다. 	출력	필요
네트워크 송신 2	SEND2 @DEND2	491	SEND2(491) S D C I S: 선두 소스 워드 D: 선두 저장처 워드 C: 선두 제어 워드 I: 선두 통신 정보 워드	SEND2 명령어는 최대 64개를 동시에 사용할 수 있습니다. 그 밖의 점은 SEND와 같습니다.	출력	필요
네트워크 수신	RECV @RECV	098	RECV(098) S D C S: 선두 소스 워드 D: 선두 저장처 워드 C: 선두 제어 워드	네트워크 내의 노드에 데이터 송신을 요구하여 그 데이터를 수신합니다. 	출력	필요
네트워크 수신 2	RECV2 @RECV2	492	RECV2(492) S D C I S: 선두 소스 워드 D: 선두 저장처 워드 C: 선두 제어 워드 I: 선두 통신 정보 워드	RECV2 명령어는 최대 64개를 동시에 사용할 수 있습니다. 그 밖의 점은 RECV와 같습니다.	출력	필요
커맨드 전달	CMND @CMND	490	CMND(490) S D C S: 선두 커맨드 워드 D: 선두 레스폰스 워드 C: 선두 제어 워드	FINS 커맨드를 송신하고 레스폰스를 수신합니다 	출력	필요

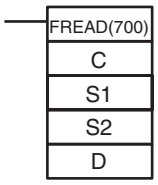
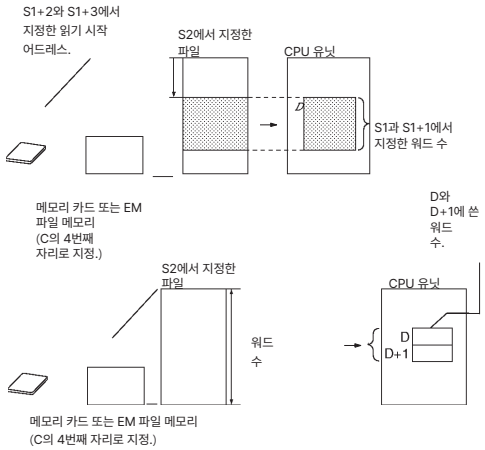
사본 번역본

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
커맨드 전달 2	CMND2 @CMND2	493	 S: 선두 커맨드 워드 D: 선두 레스폰스 워드 C: 선두 제어 워드 I: 선두 통신 정보 워드	CMND2 명령어는 최대 64개를 동시에 사용할 수 있습니다. 그 밖의 점은 CMND와 같습니다.	출력	필요
익스플리시트 메시지 송신	EXPLT	720	 **S**: 송신 메시지 선두 워드 D: 수신 메시지 선두 워드 C: 선두 제어 워드	임의의 서비스 코드로 익스플리시트 메시지를 송신합니다.	출력	필요
익스플리시트 속성 취득	EGATR	721	 **S**: 송신 메시지 선두 워드 D: 수신 메시지 선두 워드 C: 선두 제어 워드 메시지	익스플리시트 메시지로 스테이터스 정보를 읽습니다(Get Attribute Single, 서비스 코드: 0E hex).	출력	필요
익스플리시트 속성 설정	ESATR	722	 S: 송신 메시지 선두 워드 C: 선두 제어 워드	익스플리시트 메시지로 스테이터스 정보를 씁니다(Set Attribute Single, 서비스 코드: 0E hex)	출력	필요
익스플리시트 워드 읽기	ECHRD	723	 S: 상대 CPU 유닛의 선두 소스 워드 D: 자체 CPU 유닛의 선두 저장처 워드 C: 선두 제어 워드	네트워크 내의 상대 CPU 유닛에서 자체 CPU 유닛으로 데이터를 읽습니다. (상대 CPU 유닛이 익스플리시트 메시지를 지원해야 합니다.)	출력	필요

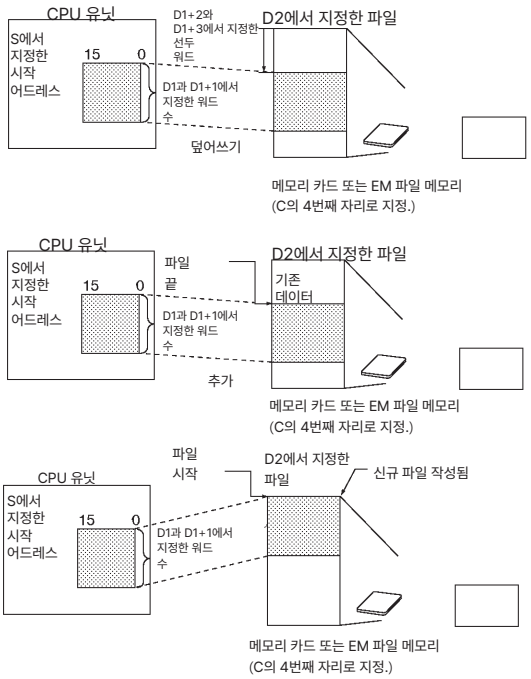
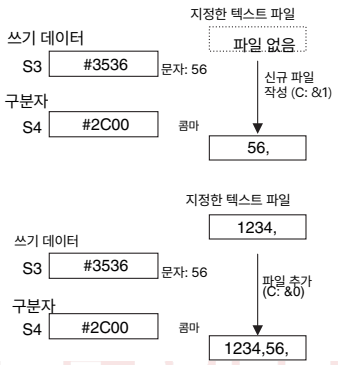
-Reference Only-

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
익스플리시트 워드 쓰기	ECHWR	724	 S: 자체 CPU 유닛의 선두 소스 워드 D: 상대 CPU 유닛의 선두 저장처 워드 C: 선두 제어 워드	자체 CPU 유닛에서 네트워크 내의 상대 CPU 유닛으로 데이터를 씁니다. (상대 CPU 유닛이 익스플리시트 메시지를 지원해야 합니다.)	출력	필요

A-1-25 파일 메모리 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
데이터 파일 읽기	FREAD @FREAD	700	 C: 제어 워드 S1: 선두 소스 워드 S2: 파일명 D: 선두 저장처 워드	파일 메모리의 지정한 데이터 파일에서 지정한 데이터 또는 지정한 양의 데이터를 CPU 유닛의 지정한 데이터 영역으로 읽습니다. 	출력	필요

AI 번역본
-For Reference Only-

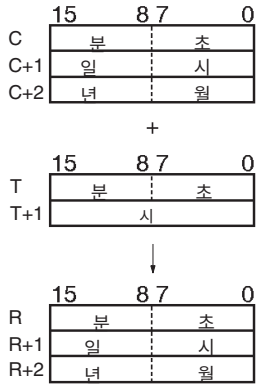
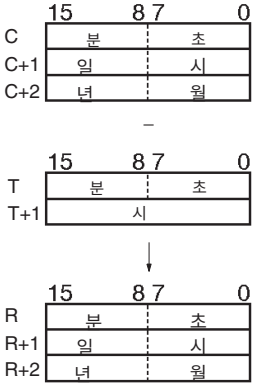
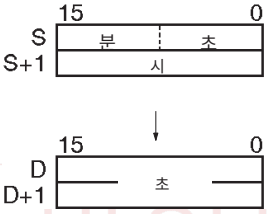
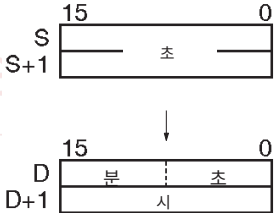
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
데이터 파일 쓰기	FWRIT @FWRIT	701	FWRIT(701) C D1 D2 S C: 제어 워드 D1: 선두 저장처 워드 D2: 파일명 S: 선두 소스 워드	CPU 유닛의 데이터 영역에 있는 지정한 데이터로 파일 메모리의 지정한 데이터 파일을 덮어쓰거나 추가합니다. 지정한 파일이 없으면 그 파일명으로 새 파일이 작성됩니다. 	출력	필요
텍스트 파일 쓰기	TWRIT @TWRIT	704	TWRIT C S1 S2 S3 S4 C: 제어 워드 S1: 쓸 바이트 수 S2: 디렉터리 및 파일명 S3: 쓰기 데이터 S4: 구분자	I/O 메모리에서 ASCII 데이터를 읽어, 그 데이터를 텍스트 파일로 메모리 카드에 저장합니다(신규 파일 작성 또는 파일 추가). 데이터는 TXT 형식으로 저장됩니다. 	출력	필요

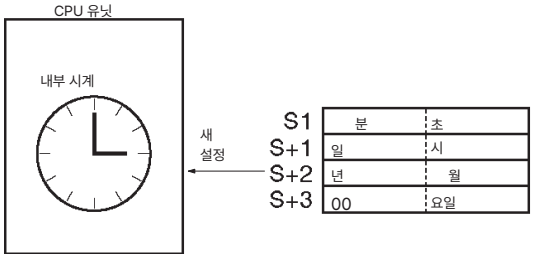
A-1-26 표시 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
메시지 표시	MSG @MSG	046	MSG(046) N M N: 메시지 번호 M: 선두 메시지 워드	지정한 16워드의 확장 ASCII를 읽어, 프로그래밍 콘솔 등의 주변 장치에 그 메시지를 표시합니다.	출력	필요

AI 문서론
-For Reference Only-

A-1-27 클록 명령어

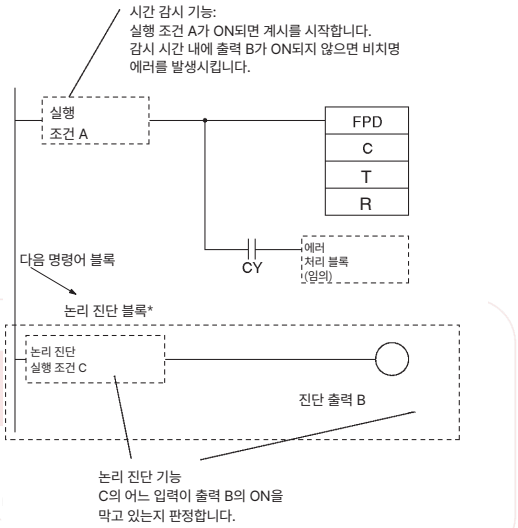
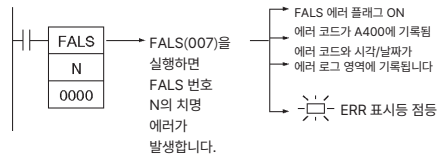
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
캘린더 가산	CADD @CADD	730	CADD(730) C T R C: 선두 캘린더 워드 T: 선두 시각 워드 R: 선두 결과 워드	지정한 워드의 캘린더 데이터에 시간을 가산합니다. 	출력	필요
캘린더 감산	CSUB @CSUB	731	CSUB(731) C T R C: 선두 캘린더 워드 T: 선두 시각 워드 R: 선두 결과 워드	지정한 워드의 캘린더 데이터에서 시간을 감산합니다. 	출력	필요
시간 → 초	SEC @SEC	065	SEC(065) S D S: 선두 소스 워드 D: 선두 저장처 워드	시/분/초 형식의 시간 데이터를 초 단위만의 동등한 시간으로 변환합니다. 	출력	필요
초 → 시간	HMS @HMS	066	HMS(066) S D S: 선두 소스 워드 D: 선두 저장처 워드	초 데이터를 시/분/초 형식의 동등한 시간으로 변환합니다. 	출력	필요

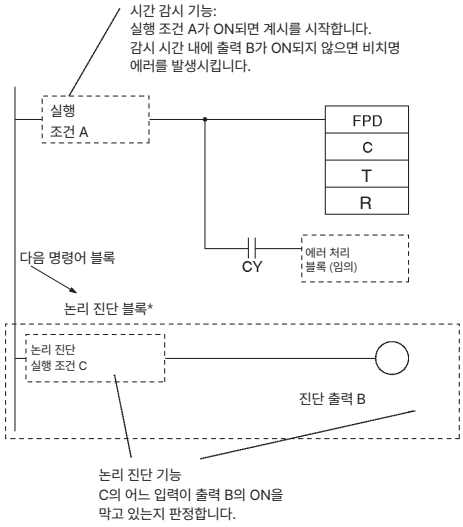
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
시계 조정	DATE @DATE	735	DATE(735) S S: 선두 소스 워드	내부 시계 설정을 지정한 소스 워드의 설정으로 변경합니다. 	출력	필요

A-1-28 디버깅 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
트레이스 메모리 샘플링	TRSM	045	TRSM(045)	TRSM(045)이 실행되면 미리 선택한 비트 또는 워드의 상태가 샘플링되어 트레이스 메모리에 저장됩니다. TRSM(045)은 프로그램 내 어디에서나 몇 번이든 사용할 수 있습니다.	출력	필요 없음

A-1-29 고장 진단 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
고장 알람	FAL @FAL	006	FAL(006) N S N: FAL 번호 S: 발생시킬 선두 메시지 워드 또는 에러 코드	사용자 정의 비치명 에러를 발생시키거나 해제합니다. 비치명 에러는 PC 운전을 정지시키지 않습니다. 시스템에 의한 비치명 에러도 발생시킵니다. 	출력	필요
중고장 알람	FALS	007	FALS(007) N S N: FALS 번호 S: 발생시킬 선두 메시지 워드 또는 에러 코드	사용자 정의 치명 에러를 발생시킵니다. 치명 에러는 PC 운전을 정지시킵니다. 시스템에 의한 치명 에러도 발생시킵니다. 	출력	필요

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
고장점 검출	FPD	269	FPD(269) C T R C: 제어 워드 T: 감시 시간 R: 선두 레지스터 워드	FPD(269)의 실행부터 진단 출력의 실행까지의 시간을 감시하고, 어느 입력이 출력의 ON을 막고 있는지 찾아내어 명령어 블록의 고장을 진단합니다. 	출력	필요

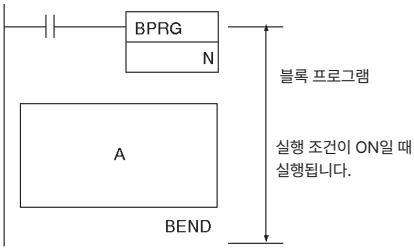
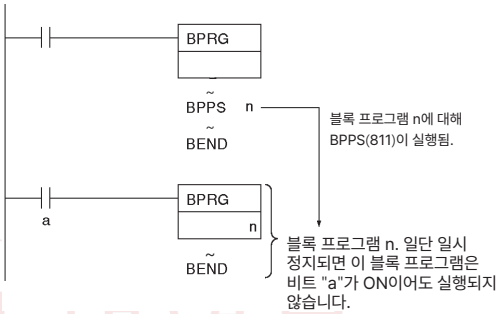
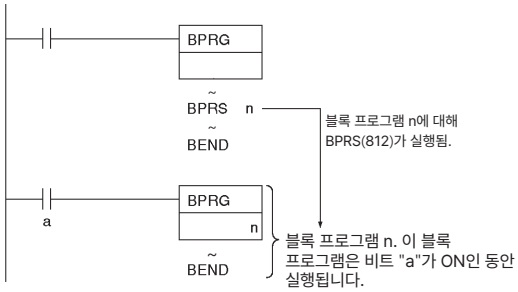
A-1-30 기타 명령어

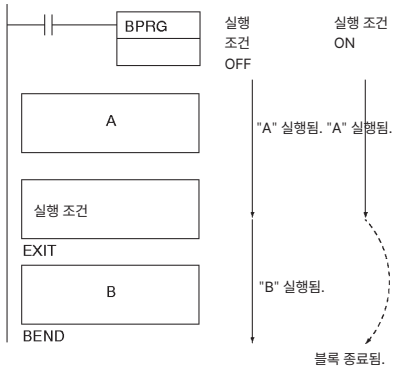
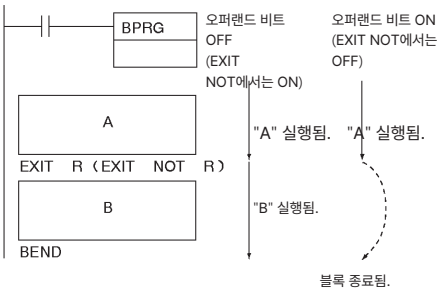
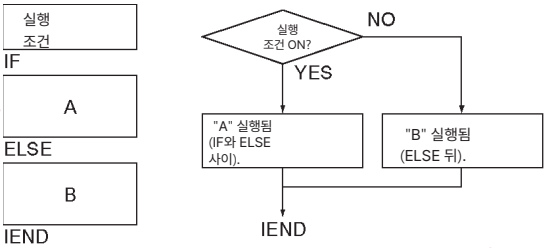
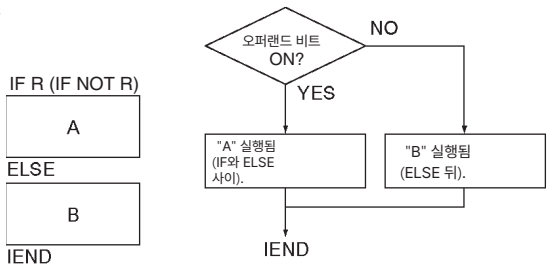
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
캐리 세트	STC @STC	040	STC(040)	캐리 플래그(CY)를 세트합니다.	출력	필요
캐리 클리어	CLC @CLC	041	CLC(041)	캐리 플래그(CY)를 OFF합니다.	출력	필요
EM 뱅크 선택	EMBC @EMBC	281	EMBC(281) N N: EM 뱅크 번호	현재 EM 뱅크를 변경합니다.	출력	필요
최대 사이클 타임 연장	WDT @WDT	094	WDT(094) T T: 타이머 설정값	최대 사이클 타임을 연장합니다. 단, 이 명령어가 실행되는 사이클에 한합니다.	출력	필요
조건 플래그 저장	CCS @CCS	282	CCS(282)	조건 플래그의 상태를 저장합니다.	출력	필요
조건 플래그 로드	CCL @CCL	283	CCL(283)	저장된 조건 플래그의 상태를 읽습니다.	출력	필요
CV에서 어드레스 변환	FRMCV @FRMCV	284	FRMCV(284) S D S: CV 시리즈 메모리 어드레스가 들어 있는 워드 D: 저장처 인덱스 레지스터	CV 시리즈 PLC 메모리 어드레스를 이에 대응하는 CS/CJ 시리즈 PLC 메모리 어드레스로 변환합니다.	출력	필요

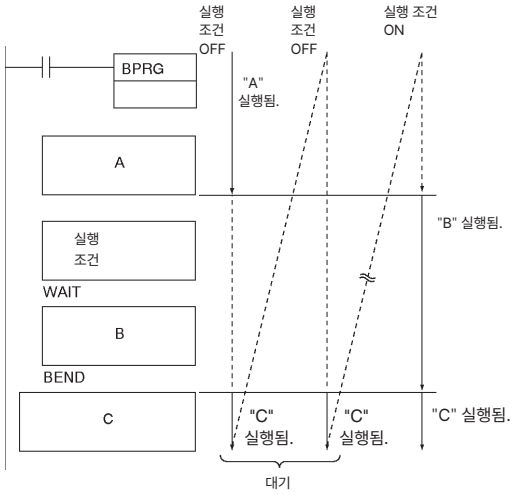
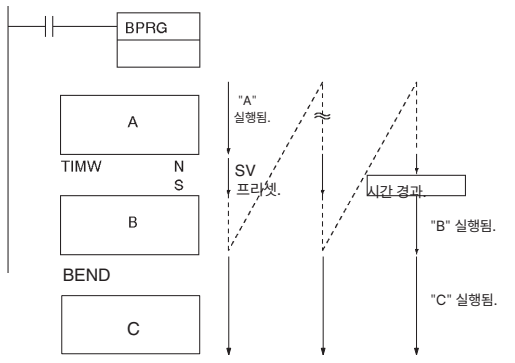
AI 번역본 -For Reference Only-

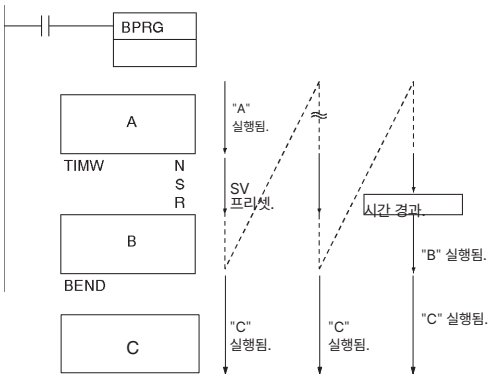
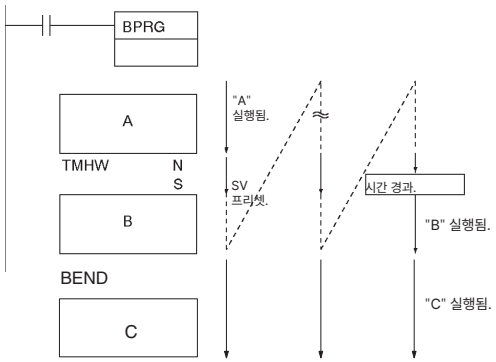
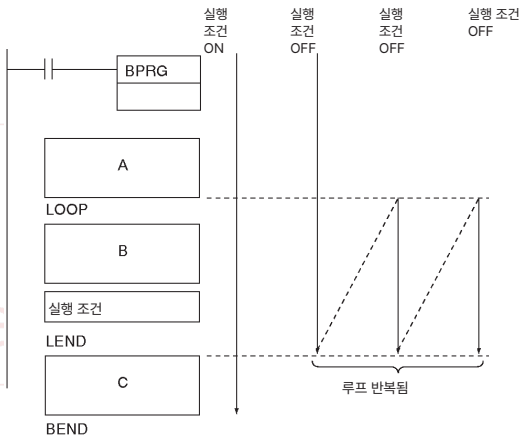
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
CV로 어드레스 변환	TOCV @TOCV	285	TOCV(285) S D S: CS 시리즈 메모리 어드레스가 들어 있는 인덱스 레지스터 D: 저장처 워드	CS/CJ 시리즈 PLC 메모리 어드레스를 이에 대응하는 CVM1/CV 시리즈 PLC 메모리 어드레스로 변환합니다.	출력	필요

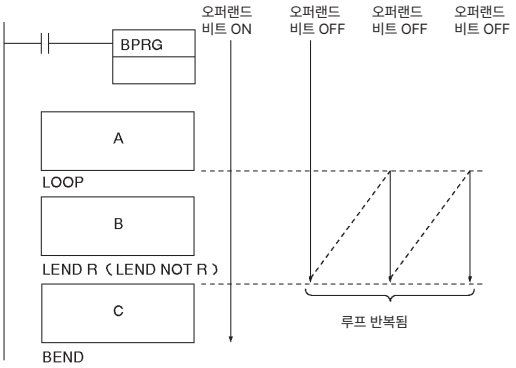
A-1-31 블록 프로그래밍 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
블록 프로그램 개시	BPRG	096	BPRG(096) N N: 블록 프로그램 번호	블록 프로그램 영역을 정의합니다. BPRG(096)마다 반드시 대응하는 BEND(801)이 있어야 합니다. 	출력	필요
블록 프로그램 종료	BEND	801	---	블록 프로그램 영역을 정의합니다. BPRG(096)마다 반드시 대응하는 BEND(801)이 있어야 합니다.	블록 프로그램	필요
블록 프로그램 일시 정지	BPPS	811	BPPS (811) N N: 블록 프로그램 번호	다른 블록 프로그램에서 지정한 블록 프로그램을 일시 정지 및 재개합니다. 	블록 프로그램	필요
블록 프로그램 재개	BPRS	812	BPRS (812) N N: 블록 프로그램 번호	다른 블록 프로그램에서 지정한 블록 프로그램을 일시 정지 및 재개합니다. 	블록 프로그램	필요

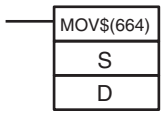
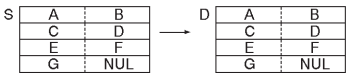
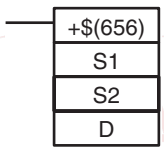
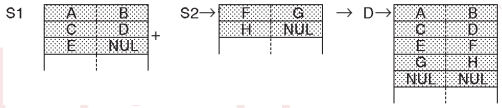
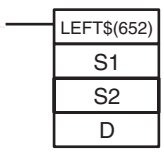
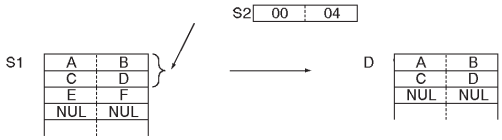
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
조건부 블록 종료	EXIT	806	EXIT(806) B: 비트 오퍼랜드	오퍼랜드 비트가 없는 EXIT(806)은 실행 조건이 ON이면 프로그램을 빠져나갑니다. 	블록 프로그램	필요
조건부 블록 종료	EXIT	806	EXIT(806)B B: 비트 오퍼랜드	오퍼랜드 비트가 없는 EXIT(806)은 실행 조건이 ON이면 프로그램을 빠져나갑니다. 	블록 프로그램	필요
조건부 블록 종료 NOT	EXIT NOT	806	EXIT NOT(806) B B: 비트 오퍼랜드	오퍼랜드 비트가 없는 EXIT(806)은 실행 조건이 OFF이면 프로그램을 빠져나갑니다.	블록 프로그램	필요
조건부 블록 분기	IF	802	IF (802)	실행 조건이 ON이면 IF(802)와 ELSE(803) 사이의 명령어가 실행되고, 실행 조건이 OFF이면 ELSE(803)과 IEND(804) 사이의 명령어가 실행됩니다. 	블록 프로그램	필요
조건부 블록 분기	IF	802	IF (802) B B: 비트 오퍼랜드	오퍼랜드 비트가 ON이면 IF(802)와 ELSE(803) 사이의 명령어가 실행됩니다. 오퍼랜드 비트가 OFF이면 ELSE(803)과 IEND(804) 사이의 명령어가 실행됩니다. 	블록 프로그램	필요

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
조건부 블록 분기 (NOT)	IF NOT	802	IF (802) NOT B B: 비트 오퍼랜드	IF(802)와 ELSE(803) 사이의 명령어가 실행되고, 오퍼랜드 비트가 ON이면 ELSE(803)과 IEND(804)의 명령어가 오퍼랜드 비트가 OFF이면 실행됩니다.	블록 프로그램	필요
조건부 블록 분기 (ELSE)	ELSE	803	---	ELSE(803) 명령어를 생략하고 오퍼랜드 비트가 ON이면 IF(802)와 IEND(804) 사이의 명령어가 실행됩니다	블록 프로그램	필요
조건부 블록 분기 종료	IEND	804	---	오퍼랜드 비트가 OFF이면 IEND(804) 뒤의 명령어만 실행됩니다.	블록 프로그램	필요
1사이클 대기	WAIT	805	WAIT(805)	WAIT(805)에서 실행 조건이 ON이면 블록 프로그램의 나머지 명령어는 건너뜁니다. 	블록 프로그램	필요
1사이클 대기	WAIT	805	WAIT(805) B B: 비트 오퍼랜드	오퍼랜드 비트가 OFF(WAIT NOT(805)에서는 ON)이면 블록 프로그램의 나머지 명령어는 건너뜁니다. 다음 사이클에서는 WAIT(805) 또는 WAIT(805) NOT의 실행 조건을 제외하고 블록 프로그램이 전혀 실행되지 않습니다. 실행 조건이 ON(WAIT(805) NOT에서는 OFF)되면 WAIT(805) 또는 WAIT(805) NOT부터 프로그램 끝까지의 명령어가 실행됩니다.	블록 프로그램	필요
1사이클 대기 (NOT)	WAIT NOT	805	WAIT(805) NOT B B: 비트 오퍼랜드	오퍼랜드 비트가 OFF(WAIT NOT(805)에서는 ON)이면 블록 프로그램의 나머지 명령어는 건너뜁니다. 다음 사이클에서는 WAIT(805) 또는 WAIT(805) NOT의 실행 조건을 제외하고 블록 프로그램이 전혀 실행되지 않습니다. 실행 조건이 ON(WAIT(805) NOT에서는 OFF)되면 WAIT(805) 또는 WAIT(805) NOT부터 프로그램 끝까지의 명령어가 실행됩니다.	블록 프로그램	필요
타이머 대기	TIMW (BCD)	813	TIMW(813) N SV N: 타이머 번호 SV: 설정값	지정한 시간이 경과할 때까지 블록 프로그램의 실행을 지연시킵니다. 타이머가 타임 업되면 TIMW(813)/TIMWX(816) 다음 명령어부터 실행이 계속됩니다. SV: BCD는 0~999.9 s, 바이너리는 0~6,553.5 s 	블록 프로그램	필요
	TIMWX (바이너리)	816	TIMWX(816) N SV N: 타이머 번호 SV: 설정값			

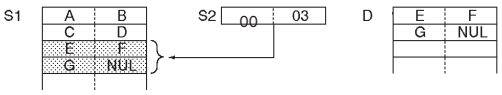
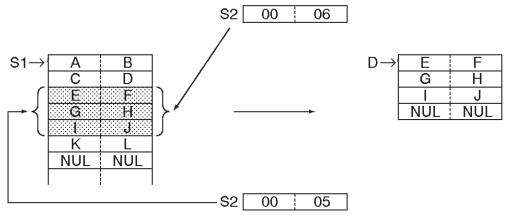
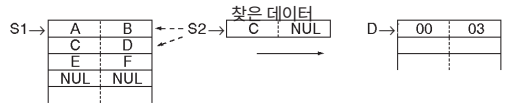
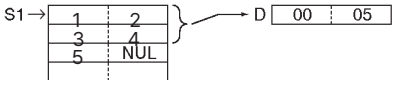
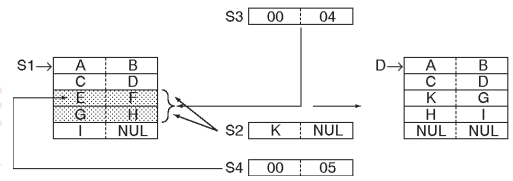
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
카운터 대기	CNTW (BCD)	814	CNTW(814) N SV N: 카운터 번호 SV: 설정값 I: 계수 입력	지정한 계수에 도달할 때까지 블록 프로그램의 나머지 실행을 지연시킵니다. 카운터가 카운트 아웃되면 CNTW(814)/CNTWX(817) 다음 명령어부터 실행이 계속됩니다. SV: BCD는 0~9,999회, 바이너리는 0~65,535회 	블록 프로그램	필요
	CNTWX (바이너리)	818	CNTWX(818) N SV N: 카운터 번호 SV: 설정값 I: 계수 입력			
고속 타이머 대기	TMHW (BCD)	815	TMHW(815) N SV N: 타이머 번호 SV: 설정값	지정한 시간이 경과할 때까지 블록 프로그램의 나머지 실행을 지연시킵니다. 타이머가 타임 업되면 TMHW(815) 다음 명령어부터 실행이 계속됩니다. SV: BCD는 0~99.99 s, 바이너리는 0~655.35 s 	블록 프로그램	필요
	TMHWX (바이너리)	817	TMHW(817) N SV N: 타이머 번호 SV: 설정값			
LOOP	LOOP	809	---	LOOP(809)는 루프 프로그램의 시작을 지정합니다. 	블록 프로그램	필요
LEND	LEND	810	LEND (810)	LEND(810) 또는 LEND(810) NOT은 루프의 끝을 지정합니다. LEND(810) 또는 LEND(810) NOT에 도달하면, LEND(810) 또는 LEND(810) NOT의 오퍼랜드 비트가 각각 ON 또는 OFF 될 때까지, 또는 LEND(810)의 실행 조건이 ON 될 때까지 프로그램 실행은 바로 앞의 LOOP(809)로 되돌아가 반복됩니다.	블록 프로그램	필요

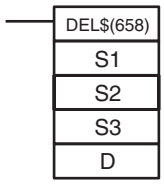
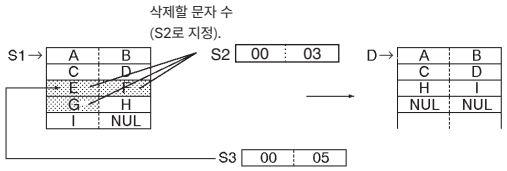
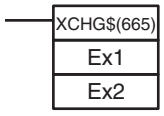
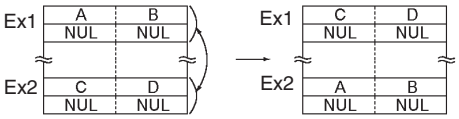
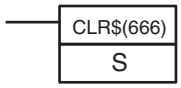
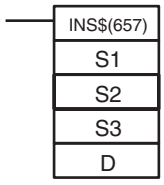
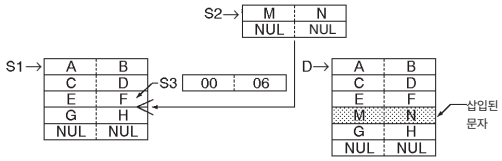
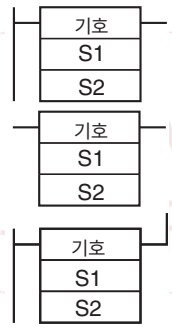
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
LEND	LEND	810	LEND (810) B B: 비트 오퍼랜드	LEND(810)의 오퍼랜드 비트가 OFF(LEND(810) NOT에서는 ON)이면 LOOP(809) 다음 명령어부터 루프의 실행이 반복됩니다. LEND(810)의 오퍼랜드 비트가 ON(LEND(810) NOT에서는 OFF)이면 루프가 종료되고 LEND(810) 또는 LEND(810) NOT 다음 명령어로 실행이 계속됩니다.  **주** LEND(810) NOT에서는 오퍼랜드 비트의 상태가 반대가 됩니다.	블록 프로그램	필요
LEND NOT	LEND NOT	810	LEND(810) NOT B: 비트 오퍼랜드	LEND(810) 또는 LEND(810) NOT은 루프의 끝을 지정합니다. LEND(810) 또는 LEND(810) NOT에 도달하면, LEND(810) 또는 LEND(810) NOT의 오퍼랜드 비트가 각각 ON 또는 OFF 될 때까지, 또는 LEND(810)의 실행 조건이 ON 될 때까지 프로그램 실행은 바로 앞의 LOOP(809)로 되돌아가 반복됩니다.	블록 프로그램	필요

A-1-32 문자열 처리 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
문자열 전송	MOV$ @MOV$	664	 S: 선두 소스 워드 D: 선두 저장처 워드	문자열을 전송합니다.  	출력	필요
문자열 결합	+$ @+$	656	 S1: 문자열 1 S2: 문자열 2 D: 선두 저장처 워드	하나의 문자열을 다른 문자열에 연결합니다.  	출력	필요
문자열 좌측 취득	LEFT$ @LEFT$	652	 S1: 문자열 선두 워드 S2: 문자 수 D: 선두 저장처 워드	문자열의 왼쪽(시작)부터 지정한 문자 수만큼 가져옵니다.  	출력	필요

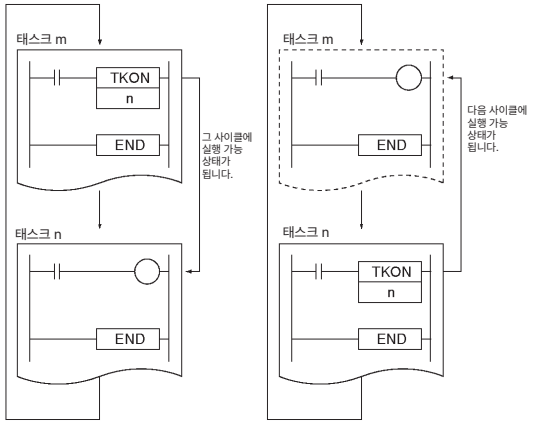
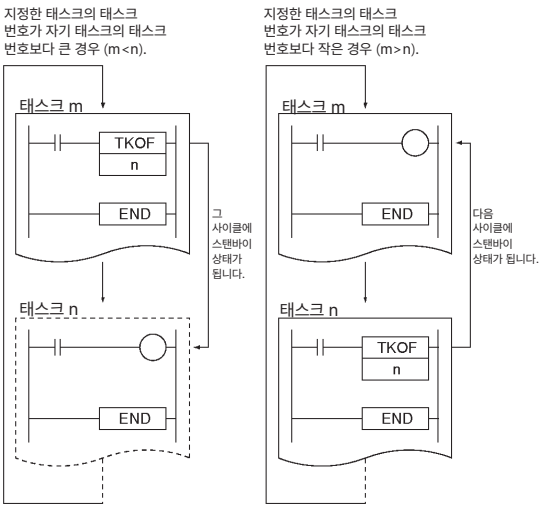
-For Reference Only-

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
문자열 우측 취득	RGHT$ @RGHT$	653	RGHT$(653) S1 S2 D S1: 문자열 선두 워드 S2: 문자 수 D: 선두 저장처 워드	문자열의 오른쪽(끝)부터 지정한 문자 수만큼 읽습니다. 	출력	필요
문자열 중간 취득	MID$ @MID$	654	MID$(654) S1 S2 S3 D S1: 문자열 선두 워드 S2: 문자 수 S3: 시작 위치 D: 선두 저장처 워드	문자열 중간의 임의 위치부터 지정한 문자 수만큼 읽습니다. 	출력	필요
문자열 내 검색	FIND @FIND$	660	FIND$(660) S1 S2 D S1: 소스 문자열 선두 워드 S2: 검색할 문자열 선두 워드 D: 선두 저장처 워드	문자열 안에서 지정한 문자열을 찾습니다. 	출력	필요
문자열 길이	LEN$ @LEN$	650	LEN$(650) S D S: 문자열 선두 워드 D: 선두 저장처 워드	문자열의 길이를 산출합니다. 	출력	필요
문자열 내 치환	RPLC$ @RPLC$	661	RPLC$(654) S1 S2 S3 S4 D S1: 문자열 선두 워드 S2: 치환 문자열 선두 워드 S3: 문자 수 S4: 시작 위치 D: 선두 저장처 워드	지정한 위치부터 문자열을 지정한 문자열로 치환합니다. 	출력	필요

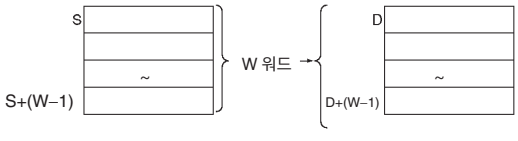
명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
문자열 삭제	DEL$ @DEL$	658	 S1: 문자열 선두 워드 S2: 문자 수 S3: 시작 위치 D: 선두 저장처 워드	문자열 중간에서 지정한 문자열을 삭제합니다. 	출력	필요
문자열 교환	XCHG$ @XCHG$	665	 Ex1: 선두 교환 워드 1 Ex2: 선두 교환 워드 2	지정한 문자열을 다른 지정한 문자열과 바꿉니다. 	출력	필요
문자열 클리어	CLR$ @CLR$	666	 S: 문자열 선두 워드	문자열 전체를 NUL(00 hex)로 클리어합니다. 	출력	필요
문자열 내 삽입	INS$ @INS$	667	 S1: 기준 문자열 선두 워드 S2: 삽입 문자열 선두 워드 S3: 시작 위치 D: 선두 저장처 워드	문자열 중간에서 지정한 문자열을 삭제합니다. 	출력	필요
문자열 비교	LD, AND, OR + =$, <>$, <$, <=$, >$, >=$	670 (=$) 671 (<>$) 672 (<$) 673 (<=$) 674 (>$) 675 (>=$)	 S1: 문자열 1 S2: 문자열 2	문자열 비교 명령어(=$, <>$, <$, <=$, >$, >=$)는 두 문자열을 처음부터 ASCII 코드 값으로 비교합니다. 비교 결과가 참이면 LOAD, AND 또는 OR에 대해 ON 실행 조건이 생성됩니다.	LD: 논리 시작 AND, OR: 렁에서 계속	필요

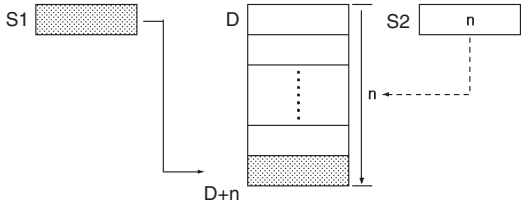
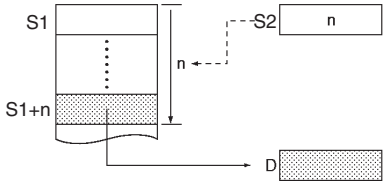
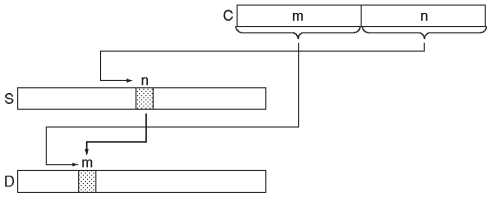
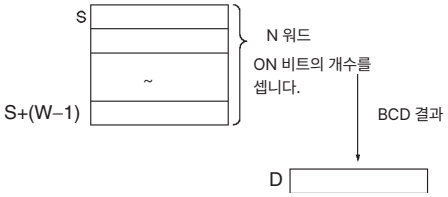
번역본
-Reference Only-

A-1-33 태스크 제어 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
태스크 ON	TKON @TKON	820	TKON(820) N N: 태스크 번호	지정한 태스크를 실행 가능 상태로 만듭니다. 지정한 태스크의 태스크 번호가 자기 태스크의 태스크 번호보다 큰 경우 (m<n). 지정한 태스크의 태스크 번호가 자기 태스크의 태스크 번호보다 작은 경우 (m>n). 	출력	필요
태스크 OFF	TKOF @TKOF	821	TKOF(821) N N: 태스크 번호	지정한 태스크를 스탠바이 상태로 만듭니다. 지정한 태스크의 태스크 번호가 자기 태스크의 태스크 번호보다 큰 경우 (m<n). 지정한 태스크의 태스크 번호가 자기 태스크의 태스크 번호보다 작은 경우 (m>n). 	출력	필요

A-1-34 기종 변환 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
블록 전송	XFERC @XFERC	565	XFERC(565) N S D N: 워드 수 S: 선두 소스 워드 D: 선두 저장처 워드	지정한 수만큼의 연속된 워드를 전송합니다. 	출력	필요

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
1워드 분배	DISTC @DISTC	566	DISTC(566) S Bs Of S: 소스 워드 Bs: 저장처 베이스 어드레스 Of: 오프셋	베이스 어드레스에 오프셋 값을 더해 산출한 저장처 워드로 소스 워드를 전송합니다.  스택에 쓸 수도 있습니다(스택 푸시 조작).	출력	필요
데이터 수집	COLLC @COLLC	567	COLLC(567) Bs Of D Bs: 소스 베이스 어드레스 Of: 오프셋 (BCD) D: 저장처 워드	소스 워드(베이스 어드레스에 오프셋 값을 더해 산출)를 저장처 워드로 전송합니다.  스택에서 데이터를 읽을 수도 있습니다(스택 읽기 조작).	출력	필요
비트 전송	MOVBC @MOVBC	568	MOVBC(568) S C D S: 소스 워드 또는 데이터 C: 제어 워드 (BCD) D: 저장처 워드	지정한 비트를 전송합니다. 	출력	필요
비트 카운터	BCNTC @BCNTC	621	BCNTC(621) N S R N: 워드 수 (BCD) S: 선두 소스 워드 R: 결과 워드	지정한 워드의 ON 비트 총 개수를 셉니다. 	출력	필요

A-1-35 특수 펑션 블록 명령어

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
변수 ID 취득	GETID @GETID	286	GETID(286) S D1 D2 S: 변수 또는 어드레스 D1: ID 코드 D2: 저장처 워드	지정한 변수 또는 어드레스에 대한 FINS 커맨드의 변수 종별(데이터 영역) 코드와 워드 어드레스를 출력합니다. 이 명령어는 일반적으로 펑션 블록 내 변수에 할당된 어드레스를 얻는 데 사용합니다.	출력	필요
스텝 활성화	SA @SA	784	SA(784) D	액션의 실행을 개시하기 위해 지정한 스텝 또는 서브차트를 액티브 상태로 만듭니다.	출력	필요

명령어	니모닉	코드	기호 · 오퍼랜드	기능	위치	실행 조건
스텝 비활성화	SE @SE	785	SE(784) D	액션의 실행을 종료하기 위해 지정한 서브차트의 스텝을 비액티브 상태로 만듭니다.	출력	필요
설정 타이머 읽기	TSR @TSR	780	TSR(780) S D	S로 지정한 스텝 타임의 현재값이 D부터 저장됩니다.	출력	필요
스텝 타이머 설정	TSW @TSW	781	TSW(781) S D	S로 지정한 스텝 타이머의 현재값이 D부터 지정된 값으로 변경됩니다.	출력	필요
SFC ON	SFCON	789	SFCON(789) N N: SFC 태스크 번호	다른 SFC 태스크 제어 명령어로 종료되거나 일시 정지된 SFC 태스크의 실행을 재개합니다.	출력	필요
SFC OFF	SFCOFF	790	SFCOFF(790) N N: SFC 태스크 번호	SFC 태스크의 실행을 종료합니다. 모든 출력의 상태는 유지됩니다. SFC 태스크의 실행이 재개되면 초기 스텝부터 실행됩니다.	출력	필요
SFC 일시 정지 (리셋 있음)	SFCPR	793	SFCPR(793) N N: SFC 태스크 번호	SFC 태스크의 실행을 일시 정지합니다. 모든 출력의 상태는 리셋됩니다.	출력	필요
SFC 일시 정지 (리셋 없음)	SFCPRN	791	SFCPRN(791) N N: SFC 태스크 번호	SFC 태스크의 실행을 일시 정지합니다. 모든 출력의 상태는 유지됩니다.	출력	필요

AI 번역본
-For Reference Only-

A-2 명령어 실행 시간과 스텝 수

다음 표는 CPU 유닛이 지원하는 모든 명령어의 실행 시간을 나타냅니다.

하나의 사용자 프로그램 전체에 포함된 명령어의 총 실행 시간은 사이클 타임을 계산할 때의 프로그램 실행 처리 시간입니다(*1).

*1 사용자 프로그램에는 사이클릭 태스크 내에서 실행할 수 있는 태스크와 인터럽트 조건을 만족하는 인터럽트 태스크가 할당됩니다.

대부분의 명령어 실행 시간은 사용하는 CPU 유닛과 명령어가 실행될 때의 조건에 따라 달라집니다.

실행 조건이 OFF일 때도 실행 시간이 달라질 수 있습니다.

다음 표는 각 명령어의 길이도 길이(스텝) 열에 나타냅니다. 각 명령어가 사용자 프로그램 영역에서 필요로 하는 스텝 수는 그 명령어와 함께 사용하는 오퍼랜드에 따라 달라집니다.

프로그램의 스텝 수는 명령어 수와 같지 않습니다.

주1 대부분의 명령어는 미분형(↑, ↓, @, %로 표시)을 지원합니다.
미분을 지정하면 실행 시간이 다음만큼 늘어납니다.

(단위: μs)

기호	CJ2 CPU 유닛		CJ3 CPU 유닛
	CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	CJ3M-CPU□□
↑ 또는 ↓	+0.24	+0.32	+0.32
@ 또는 %	+0.24	+0.32	+0.32

2 명령어가 실행되지 않을 때는 다음 시간을 기준으로 삼으십시오.

(단위: μs)

CJ2 CPU 유닛		CJ3 CPU 유닛
CJ2H-CPU6□(-EIP)	CJ2M-CPU□□	CJ3M-CPU□□
0.016	0.020	0.020

실행 시간은 명령어가 실행되는 순서와 명령어의 실행 여부에 따라 다소 달라집니다. 최악의 경우에 대한 최대 변동폭은 다음 식으로 계산하십시오.

CJ2H CPU 유닛: 명령어 스텝 수 × 0.016 (μs)
CJ2M/CJ3M CPU 유닛: 명령어 스텝 수 × 0.020 (μs)

AI 번역본
-For Reference Only-

A-2-1 시퀀스 입력 명령어

명령어	니모닉	길이 (스텝)	ON 실행 시간 (μs)			조건
			CJ2H	CJ2M	CJ3M	
로드	LD	1~2	0.016	0.040	0.040	---
	!LD	2~14	0.99	1.260	1.260	---
로드 NOT	LD NOT	1~2	0.016	0.040	0.040	---
	!LD NOT	2~14	0.99	1.260	1.260	---
AND	AND	1~2	0.016	0.040	0.040	---
	!AND	2~14	0.99	1.260	1.260	---
AND NOT	AND NOT	1~2	0.016	0.040	0.040	---
	!AND NOT	2~14	0.99	1.260	1.260	---
OR	OR	1~2	0.016	0.040	0.040	---
	!OR	2~14	0.99	1.260	1.260	---
OR NOT	OR NOT	1~2	0.016	0.040	0.040	---
	!OR NOT	2~14	0.99	1.260	1.260	---
AND 로드	AND LD	1	0.016	0.040	0.040	---
OR 로드	OR LD	1	0.016	0.040	0.040	---
NOT	NOT	1	0.016	0.040	0.040	---
조건 ON	UP	3	0.26	0.36	0.36	---
조건 OFF	DOWN	4	0.27	0.40	0.40	---
로드 비트 테스트	LD TST	4	0.11	0.16	0.16	---
로드 비트 테스트 NOT	LD TSTN	4	0.11	0.16	0.16	---
AND 비트 테스트	AND TST	4	0.11	0.16	0.16	---
AND 비트 테스트 NOT	AND TSTN	4	0.11	0.16	0.16	---
OR 비트 테스트	OR TST	4	0.11	0.16	0.16	---
OR 비트 테스트 NOT	OR TSTN	4	0.11	0.16	0.16	---

AI 번역본
-For Reference Only-

A-2-2 시퀀스 출력 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간 (µs)			조건
				CJ2H	CJ2M	CJ3M	
출력	OUT	---	1~2	0.016	0.040	0.04	---
	!OUT	---	2~14	0.99	1.320	1.32	---
출력 NOT	OUT NOT	---	1~2	0.016	0.040	0.04	---
	!OUT NOT	---	2~14	0.99	1.320	1.32	---
KEEP	KEEP	011	1~2	0.048	0.060	0.06	---
	!KEEP	011	16	0.99	1.340	1.34	---
상승 미분	DIFU	013	2~2	0.28	0.30	0.30	---
하강 미분	DIFD	014	2~2	0.24	0.30	0.30	---
SET	SET	---	1~2	0.016	0.040	0.04	---
	!SET	---	2~14	0.99	1.360	1.36	---
리셋	RSET	---	1~2	0.016	0.040	0.04	---
	!RSET	---	2~14	0.99	1.360	1.36	---
다중 비트 세트	SETA	530	4	3.68	4.12	3.1	1비트 세트 시
				15.5	24.4	23.3	1,000비트 세트 시
다중 비트 리셋	RSTA	531	4	3.7	4.1	3.0	1비트 리셋 시
				15.5	24.4	23.3	1,000비트 리셋 시
단일 비트 세트	SETB	532	2	0.19	0.280	0.28	---
	!SETB	---	16	0.99	1.120	1.12	---
단일 비트 리셋	RSTB	534	2	0.19	0.280	0.28	---
	!RSTB	---	16	0.99	1.120	1.12	---
단일 비트 출력	OUTB	534	2	0.19	0.280	0.28	---
	!OUTB	---	16	0.99	1.180	1.18	---

A-2-3 시퀀스 제어 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간 (µs)			조건
				CJ2H	CJ2M	CJ3M	
END	END	001	1	2.6	3.5	3.5	---
노 오퍼레이션	NOP	000	1	0.016	0.040	0.04	---
인터록	IL	002	1	0.048	0.060	0.06	---
인터록 클리어	ILC	003	1	0.048	0.060	0.06	---
멀티 인터록 미분 유지	MILH	517	3	2.3	3.3	3.0	인터록 조건 불성립 (입력 조건 ON)
				3.4	4.6	3.7	인터록 조건 성립 (입력 조건 OFF)
				3.8	5.2	4.1	인터록 중에 인터록 조건이 다시 성립 (입력 조건 OFF)

-For Reference Only-

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
멀티 인터록 미분 해제	MILR	518	3	2.3	3.1	2.8	인터록 조건 불성립(입력 조건 ON)
				3.4	4.5	3.7	인터록 조건 성립(입력 조건 OFF)
				3.8	5.1	4.1	인터록 중에 인터록 조건이 다시 성립(입력 조건 OFF)
멀티 인터록 클리어	MILC	519	2	1.2	1.7	0.7	인터록 중이 아님
				1.6	2.2	1.1	인터록 중
점프	JMP	004	2	0.31	0.34	0.34	---
점프 종료	JME	005	2	---	---	---	---
조건부 점프	CJP	510	2	0.31	0.34	0.34	점프 조건 성립 (입력 조건 ON)
조건부 점프 NOT	CJPN	511	2	0.31	0.34	0.34	점프 조건 성립 (입력 조건 OFF)
다중 점프	JMP0	515	1	0.048	0.060	0.06	---
다중 점프 종료	JME0	516	1	0.048	0.060	0.06	---
FOR 루프	FOR	512	2	0.27	0.42	0.42	상수를 지정한 경우
BREAK 루프	BREAK	514	1	0.048	0.060	0.06	---
NEXT 루프	NEXT	513	1	0.14	0.16	0.16	루프가 계속되는 경우
				0.18	0.18	0.18	루프가 종료되는 경우

A-2-4 타이머 · 카운터 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
100ms 타이머	TIM	---	3	0.67	0.84	0.84	---
	TIMX	550		0.67	0.84	0.84	
10ms 타이머	TIMH	015	3	0.67	0.84	0.84	---
	TIMHX	551		0.67	0.84	0.84	
1ms 타이머	TMHH	540	3	0.67	0.84	0.84	---
	TMHHX	552		0.67	0.84	0.84	
0.1ms 타이머	TIMU	541	3	0.67	0.84	0.84	---
	TIMUX	556		0.67	0.84	0.84	
0.01ms 타이머	TMUH	544	3	0.67	0.84	0.84	---
	TMUHX	557		0.67	0.84	0.84	
적산 타이머	TTIM	087	3	9.2	12.1	11.0	---
				6.9	8.4	8.4	리셋 시
				5.0	6.5	4.9	인터록 시
	TTIMX	555	3	8.8	11.7	10.7	---
				6.8	8.5	8.5	리셋 시
				5.0	6.5	4.9	인터록 시

사본 원본
For Reference Only

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
롱 타이머	TIML	542	4~5	5.8	7.0	4.9	---
				3.9	4.1	3.2	인터록 시
	TIMLX	553	4~5	5.7	7.0	5.1	---
				3.6	3.7	3.1	인터록 시
다출력 타이머	MTIM	543	4	6.4	7.2	5.3	---
				3.7	4.3	2.5	리셋 시
	MTIMX	554	4	5.5	6.4	5.1	---
				3.4	3.8	3.1	리셋 시
타이머 리셋	TRSET	549	2	0.58	0.8	0.8	---
카운터	CNT	---	3	0.51	0.58	0.58	---
	CNTX	546		0.51	0.58	0.58	
리버시블 카운터	CNTR	012	3	9.1	11.8	11.0	---
	CNTRX	548		8.0	10.3	9.4	
타이머/카운터 리셋	CNR	545	3	4.8	6.7	5.9	1 워드 리셋 시
				2839	2555	1744	1,000 워드 리셋 시
	CNRX	547	3	4.7	6.7	5.9	1 워드 리셋 시
				2839	2555	1744	1,000 워드 리셋 시

A-2-5 비교 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
입력 비교 명령어(부호 없음)	=	300	4	0.08	0.16	0.16	---
	<>	305					
	<	310					
	<=	315					
	>	320					
	>=	325					
입력 비교 명령어(더블, 부호 없음)	=L	301	4~6	0.08	0.24	0.24	---
	<>L	306					
	<L	311					
	<=L	316					
	>L	321					
	>=L	326					
입력 비교 명령어(부호 있음)	=S	302	4	0.08	0.16	0.16	---
	<>S	307					
	<S	312					
	<=S	317					
	>S	322					
	>=S	327					

AI 번역본
-For Reference Only-

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
입력 비교 명령어(더블, 부호 있음)	=SL	303	4~6	0.08	0.24	0.24	---
	<>SL	308					
	<SL	313					
	<=SL	318					
	>SL	323					
	>=SL	328					
시각 비교 명령어	=DT	341	4	16.300	27.9	6.1	---
	<>DT	342					
	<DT	343					
	<=DT	344					
	>DT	345					
	>=DT	346					
비교	CMP	020	3	0.06	0.080	0.08	---
	!CMP	020	30	2.06	2.6	2.6	---
더블 비교	CMPL	060	3~5	0.064	0.120	0.12	---
부호 있는 BIN 비교	CPS	114	3~5	0.064	0.080	0.08	---
	!CPS	114	30	2.06	2.6	2.6	---
더블 부호 있는 BIN 비교	CPSL	115	3~5	0.064	0.120	0.12	---
테이블 비교	TCMP	085	4	10.3	12.5	10.8	---
다중 비교	MCMP	019	4	15.2	20.3	17.9	---
부호 없는 블록 비교	BCMP	068	4	16.3	20.5	18.0	---
확장 블록 비교	BCMP2	502	4	5.0	8.7	5.2	데이터 워드 수: 1
				217.2	278	275	데이터 워드 수: 255
영역 범위 비교	ZCP	088	3	0.14	0.400	0.40	---
더블 영역 범위 비교	ZCPL	116	3~5	0.14	0.640	0.64	---
부호 있는 영역 범위 비교	ZCPS	117	3	0.14	0.400	0.40	---
더블 부호 있는 영역 범위 비교	ZCPSL	118	3~5	0.14	0.640	0.64	---

AI 번역본
-For Reference Only-

A-2-6 데이터 전송 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
전송	MOV	021	3	0.05	0.12	0.12	---
	!MOV	021	30	1.98	2.6	2.6	---
더블 전송	MOVL	498	3~4	0.05	0.20	0.20	---
전송 NOT	MVN	022	3	0.05	0.12	0.12	---
더블 전송 NOT	MVNL	499	3~4	0.05	0.20	0.20	---
비트 전송	MOVB	082	4	0.19	0.32	0.32	---

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
자리 전송	MOVD	083	4	0.19	0.32	0.32	---
복수 비트 전송	XFRB	062	4	6.6	9.4	7.7	1 비트 전송 시
				85.8	119	80.1	255 비트 전송 시
블록 전송	XFER	070	4	0.29	0.28	0.28	1 워드 전송 시
				240.1	220	220	1,000 워드 전송 시
블록 설정	BSET	071	4	0.21	0.20	0.20	1 워드 설정 시
				142.2	140	140	1,000 워드 설정 시
데이터 교환	XCHG	073	3	0.32	0.48	0.48	---
더블 데이터 교환	XCGL	562	3~4	0.12	0.29	0.29	---
싱글 워드 분배	DIST	080	4	4.5	4.7	3.9	---
데이터 수집	COLL	081	4	4.6	4.7	3.9	---
레지스터로 전송	MOVR	560	3	0.064	0.200	0.20	---
타이머/카운터 현재값을 레지스터로 전송	MOVRW	561	3	0.064	0.200	0.20	---

A-2-7 데이터 시프트 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
시프트 레지스터	SFT	010	3	2.86	3.47	0.69	1 워드 시프트 시
				315	422	376	1,000 워드 시프트 시
리버시블 시프트 레지스터	SFTR	084	4	6.22	6.38	5.2	1 워드 시프트 시
				319	422	392	1,000 워드 시프트 시
비동기 시프트 레지스터	ASFT	017	4	5.3	6.3	5.3	1 워드 시프트 시
				948	1285	1181	1,000 워드 시프트 시*1
워드 시프트	WSFT	016	4	2.3	9.0	8.0	1 워드 시프트 시
				233	187	185	1,000 워드 시프트 시
산술 좌 시프트	ASL	025	2	0.18	0.260	0.26	---
더블 좌 시프트	ASLL	570	2	0.32	0.420	0.42	---
산술 우 시프트	ASR	026	2	0.18	0.260	0.26	---
더블 우 시프트	ASRL	571	2	0.32	0.420	0.42	---
좌 로테이트	ROL	027	2	0.18	0.260	0.26	---
더블 좌 로테이트	ROLL	572	2	0.32	0.420	0.42	---
캐리 없는 좌 로테이트	RLNC	574	2	0.18	0.260	0.26	---
캐리 없는 더블 좌 로테이트	RLNL	576	2	0.32	0.420	0.42	---
우 로테이트	ROR	028	2	0.18	0.260	0.26	---

AI 번역본
-For Reference Only-

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
더블 우 로테이트	RORL	573	2	0.32	0.420	0.42	---
캐리 없는 우 로테이트	RRNC	575	2	0.18	0.260	0.26	---
캐리 없는 더블 우 로테이트	RRNL	577	2	0.32	0.420	0.42	---
1 자리 좌 시프트	SLD	074	3	3.7	4.4	0.69	1 워드 시프트 시
				317.8	429	376	1,000 워드 시프트 시
1 자리 우 시프트	SRD	075	3	4.5	5.4	5.2	1 워드 시프트 시
				479.5	656	392	1,000 워드 시프트 시
N 비트 데이터 좌 시프트	NSFL	578	4	4.6	29.4	5.3	1 비트 시프트 시
				31.5	36.1	1181	1,000 비트 시프트 시
N 비트 데이터 우 시프트	NSFR	579	4	4.5	29.4	8.0	1 비트 시프트 시
				39.0	50.2	185	1,000 비트 시프트 시
N 비트 좌 시프트	NASL	580	3	0.18	0.38	0.38	---
더블 N 비트 좌 시프트	NSLL	582	3	0.32	0.54	0.54	---
N 비트 우 시프트	NASR	581	3	0.18	0.38	0.38	---
더블 N 비트 우 시프트	NSRL	583	3	0.32	0.54	0.54	---

*1 명령어 실행 시간은 데이터 양의 영향을 크게 받습니다. 이는 사이클 타임에 영향을 줍니다. 사이클 타임에 미치는 영향을 줄이려면 백그라운드 실행을 지정할 수 있습니다. 자세한 내용은 10-2-5 백그라운드 실행을 참조하십시오.

A-2-8 인크리먼트/디크리먼트 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
BIN 인크리먼트	++	590	2	0.18	0.24	0.24	---
BIN 더블 인크리먼트	++L	591	2	0.18	0.24	0.24	---
BIN 디크리먼트	– –	592	2	0.18	0.24	0.24	---
BIN 더블 디크리먼트	– –L	593	2	0.18	0.24	0.24	---
BCD 인크리먼트	++B	594	2	3.0	3.4	2.7	---
BCD 더블 인크리먼트	++BL	595	2	3.2	3.5	2.7	---
BCD 디크리먼트	– –B	596	2	3.0	3.5	2.7	---
BCD 더블 디크리먼트	– –BL	597	2	3.2	3.5	2.7	---

AI 번역본 For Reference Only

A-2-9 심볼 연산 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
캐리 없는 부호 있는 BIN 가산	+	400	4	0.18	0.34	0.34	---

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
캐리 없는 부호 있는 BIN 가산(더블)	+L	401	4~6	0.18	0.24	0.24	---
캐리 있는 부호 있는 BIN 가산	+C	402	4	0.18	0.34	0.34	---
캐리 있는 부호 있는 BIN 가산(더블)	+CL	403	4~6	0.18	0.24	0.24	---
캐리 없는 BCD 가산	+B	404	4	4.0	4.8	3.8	---
캐리 없는 BCD 가산(더블)	+BL	405	4~6	4.9	6.0	5.4	---
캐리 있는 BCD 가산	+BC	406	4	4.4	5.2	4.1	---
캐리 있는 BCD 가산(더블)	+BCL	407	4~6	5.2	6.6	6.1	---
캐리 없는 부호 있는 BIN 감산	–	410	4	0.18	0.340	0.34	---
캐리 없는 부호 있는 BIN 감산(더블)	–L	411	4~6	0.18	0.24	0.24	---
캐리 있는 부호 있는 BIN 감산	–C	412	4	0.18	0.340	0.34	---
캐리 있는 부호 있는 BIN 감산(더블)	–CL	413	4~6	0.18	0.24	0.24	---
캐리 없는 BCD 감산	–B	414	4	4.1	4.9	3.7	---
캐리 없는 BCD 감산 (더블)	–BL	415	4~6	4.9	5.9	5.8	---
캐리 있는 BCD 감산	–BC	416	4	4.5	5.2	4.1	---
캐리 있는 BCD 감산(더블)	–BCL	417	4~6	5.2	6.3	6.1	---
부호 있는 BIN 승산	*	420	4	0.26	0.520	0.52	---
부호 있는 BIN 승산(더블)	*L	421	4~6	3.6	3.9	3.2	---
부호 없는 BIN 승산	*U	422	4	0.26	0.26	0.26	---
부호 없는 BIN 승산(더블)	*UL	423	4~6	3.6	3.9	3.2	---
BCD 승산	*B	424	4	3.6	4.6	3.1	---
BCD 승산(더블)	*BL	425	4~6	4.9	6.2	4.0	---
부호 있는 BIN 제산	/	430	4	0.29	0.540	0.54	---
부호 있는 BIN 제산(더블)	/L	431	4~6	4.2	4.8	3.3	---

AI 번역본
-For Reference Only-

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
부호 없는 BIN 제산	/U	432	4	0.29	0.540	0.54	---
부호 없는 BIN 제산(더블)	/UL	433	4~6	3.8	4.2	3.3	---
BCD 제산	/B	434	4	5.0	5.9	5.1	---
BCD 제산(더블)	/BL	435	4~6	4.8	5.9	3.8	---

A-2-10 변환 명령어

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
BCD→BIN 변환	BIN	023	3	0.18	0.280	0.28	---
BCD→BIN 변환(더블)	BINL	058	3~4	3.3	3.5	2.7	---
BIN→BCD 변환	BCD	024	3	0.19	0.300	0.30	---
BIN→BCD 변환(더블)	BCDL	059	3~4	3.3	3.7	2.7	---
2의 보수	NEG	160	3	0.14	0.240	0.24	---
2의 보수(더블)	NEGL	161	3~4	0.26	0.440	0.44	---
16비트 → 32비트 부호 있는 BIN 변환	SIGN	600	3	0.26	0.340	0.34	---
데이터 디코더	MLPX	076	4	0.17	0.280	0.28	1 자리 디코딩 (4~16)
				0.42	0.770	0.77	4 자리 디코딩 (4~16)
				1.14	1.760	1.76	1 자리 디코딩 8~256
				2.17	3.370	3.37	4 자리 디코딩 (8~256)
데이터 인코더	DMPX	077	4	3.3	4.6	4.4	1 자리 인코딩 (16~4)
				3.7	5.2	10.5	4 자리 인코딩 (16~4)
				17.3	26.3	24.6	1 자리 인코딩 (256~8)
				35	47	47	2 자리 인코딩 (256~8)
ASCII 변환	ASC	086	4	4.0	4.5	3.2	1 자리를 ASCII로 변환 시
				4.6	5.2	3.1	4 자리를 ASCII로 변환 시
ASCII→HEX 변환	HEX	162	4	3.3	3.8	2.8	1 자리 변환 시
열 → 행 변환	LINE	063	4	10.5	13.1	10.9	---
행 → 열 변환	COLM	064	4	13.8	17.6	15.5	---

AI 번역본
-For Reference Only-

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
부호 있는 BCD→BIN 변환	BINS	470	4	3.6	4.0	3.1	데이터 형식 설정 No. 0
				3.6	4.0	3.1	데이터 형식 설정 No. 1
				3.6	4.0	3.2	데이터 형식 설정 No. 2
				3.6	4.0	3.2	데이터 형식 설정 No. 3
부호 있는 BCD→BIN 변환(더블)	BISL	472	4~5	3.7	4.1	3.2	데이터 형식 설정 No. 0
				3.6	4.1	3.2	데이터 형식 설정 No. 1
				3.7	4.2	3.2	데이터 형식 설정 No. 2
				3.7	4.2	3.2	데이터 형식 설정 No. 3
부호 있는 BIN→BCD 변환	BCDS	471	4	3.7	4.0	3.0	데이터 형식 설정 No. 0
				3.7	4.1	3.1	데이터 형식 설정 No. 1
				3.7	4.2	3.1	데이터 형식 설정 No. 2
				3.7	4.2	3.1	데이터 형식 설정 No. 3
부호 있는 BIN→BCD 변환(더블)	BDSL	473	4~5	4.0	4.5	3.2	데이터 형식 설정 No. 0
				4.0	4.6	3.3	데이터 형식 설정 No. 1
				4.0	4.6	3.3	데이터 형식 설정 No. 2
				4.1	4.6	3.3	데이터 형식 설정 No. 3
그레이 코드 변환	GRY	474	4	26.5	49.1	7.0	8비트 바이너리
				27.6	51.1	7.3	8비트 BCD
				30.9	57.2	7.4	8비트 각도
				35.3	66.0	7.1	15비트 바이너리
				36.3	68.0	7.7	15비트 BCD
				39.6	74.0	7.7	15비트 각도
그레이 코드 → BIN 변환	GRAY_BIN	478	3	0.1	0.3	0.3	---
그레이 코드 → BIN 변환(더블)	GRAY_BINL	479	3~4	0.1	0.4	0.4	---
BIN → 그레이 코드 변환	BIN_GRAY	480	3	0.1	0.3	0.3	---
BIN → 그레이 코드 변환(더블)	BIN_GRAYL	481	3~4	0.1	0.4	0.4	---
4자리 수치 → ASCII 변환	STR4	601	3	8.4	14.2	2.9	---
8자리 수치 → ASCII 변환	STR8	602	3~4	10.2	16.4	3.5	---

AI 번역본 For Reference Only

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
16자리 수치 → ASCII 변환	STR16	603	3	15.8	28.2	6.2	---
ASCII → 4자리 수치 변환	NUM4	604	3~4	10.5	18.5	3.2	---
ASCII → 8자리 수치 변환	NUM8	605	3	14.8	27.1	4.3	---
ASCII → 16자리 수치 변환	NUM16	606	3	27.4	52.0	7.3	---

A-2-11 논리 명령어

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
논리곱	ANDW	034	4	0.14	0.340	0.34	---
더블 논리곱	ANDL	610	4~6	0.26	0.640	0.64	---
논리합	ORW	035	4	0.18	0.340	0.34	---
더블 논리합	ORWL	611	4~6	0.26	0.640	0.64	---
배타적 논리합	XORW	036	4	0.18	0.340	0.34	---
더블 배타적 논리합	XORL	612	4~6	0.26	0.640	0.64	---
배타적 부정 논리합	XNRW	037	4	0.18	0.340	0.34	---
더블 배타적 부정 논리합	XNRL	613	4~6	0.26	0.640	0.64	---
보수	COM	029	2	0.18	0.240	0.24	---
더블 보수	COML	614	2	0.32	0.440	0.44	---

A-2-12 특수 연산 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
BIN 제곱근	ROTB	620	3	15.4	24.2	3.2	---
BCD 제곱근	ROOT	072	3	17.1	25.3	3.8	---
산술 연산	APR	069	4	4.6	5.3	4.7	SIN과 COS 지정 시
				5.7	6.9	5.7	선분 근사 지정 시
부동소수점 제산	FDIV	079	4	76	149	6.4	---
비트 카운터	BCNT	067	4	0.24	0.360	0.36	1 워드 카운트 시

-For Reference Only-

A-2-13 부동소수점 연산 명령어

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
부동소수점 → 16비트 변환	FIX	450	3~4	0.13	0.24	0.24	---
부동소수점 → 32비트 변환	FIXL	451	3~4	0.13	0.32	0.32	---
16비트 → 부동소수점 변환	FLT	452	3~4	0.13	0.30	0.30	---
32비트 → 부동소수점 변환	FLTL	453	3~4	0.13	0.32	0.32	---
부동소수점 가산	+F	454	4~6	0.24	0.66	0.66	---
부동소수점 감산	–F	455	4~6	0.24	0.66	0.66	---
부동소수점 제산	/F	457	4~6	0.4	0.9	0.90	---
부동소수점 승산	*F	456	4~6	0.24	0.66	0.66	---
도 → 라디안 변환	RAD	458	3~4	2.7	3.3	3.0	---
라디안 → 도 변환	DEG	459	3~4	3.0	3.2	2.8	---
사인	SIN	460	3~4	3.8	4.3	3.0	0° 지정 시
				4.5	5.4	3.1	45° 지정 시
				5.0	6.0	3.4	90° 지정 시
고속 사인	SINQ	475	8~9	0.59	0.86	0.86	0°, 45° 또는 90° 지정 시
코사인	COS	461	3~4	3.7	4.3	3.1	0° 지정 시
				4.4	5.2	3.1	45° 지정 시
				5.3	6.7	3.4	90° 지정 시
고속 코사인	COSQ	476	8~9	0.59	0.86	0.86	0°, 45° 또는 90° 지정 시
탄젠트	TAN	462	3~4	3.9	4.5	3.5	0° 지정 시
				6.1	8.2	3.5	45° 지정 시
고속 탄젠트	TANQ	477	15~16	1.2	1.7	1.7	0°, 45° 또는 90° 지정 시
아크 사인	ASIN	463	3~4	5.8	7.1	2.9	0° 지정 시
				24.8	33.0	3.4	45° 지정 시
				5.6	7.0	3.0	90° 지정 시
아크 코사인	ACOS	464	3~4	5.3	6.8	2.9	0° 지정 시
				27.2	34.6	3.3	45° 지정 시
				6.4	7.1	3.5	90° 지정 시
아크 탄젠트	ATAN	465	3~4	4.0	5.0	3.7	0° 지정 시
				5.6	7.0	3.5	45° 지정 시
제곱근	SQRT	466	3~4	0.42	0.66	0.66	---
지수	EXP	467	3~4	3.8	4.5	3.6	---
로그	LOG	468	3~4	5.8	6.5	4.4	---
지수 거듭제곱	PWR	840	4~6	35.7	56.6	5.2	---

AI 번역본
-For Reference Only-

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
부동소수점 기호 비교	=F	329	3~5	0.13	0.26	0.26	---
	<>F	330					
	<F	331					
	<=F	332					
	>F	333					
	>=F	334					
부동소수점 → ASCII 변환	FSTR	448	4~5	15.6	23.9	10.2	---
ASCII → 부동소수점 변환	FVAL	449	3	21.2	31.4	5.9	---
부동소수점 전송 (단정밀도)	MOVF	469	3~4	0.18	0.20	0.20	---

A-2-14 배정밀도 부동소수점 명령어

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
기호 비교 (배정밀도)	=D	335	4	5.1	6.7	6.7	---
	<>D	336					
	<D	337					
	<=D	338					
	>D	339					
	>=D	340					
부동소수점 → 16비트 BIN 변환(배정밀도)	FIXD	841	3	5.1	6.0	5.5	---
부동소수점 → 32비트 BIN 변환(배정밀도)	FIXLD	842	3	5.1	6.2	5.5	---
16비트 BIN → 부동소수점 변환(배정밀도)	DBL	843	3	3.5	5.6	4.9	---
32비트 BIN → 부동소수점 변환(배정밀도)	DBLL	844	3	3.5	5.6	4.9	---
부동소수점 가산(배정밀도)	+D	845	4	6.0	7.1	7.1	---
부동소수점 감산(배정밀도)	–D	846	4	6.1	7.1	7.1	---
부동소수점 승산(배정밀도)	*D	847	4	6.1	7.1	7.1	---
부동소수점 제산(배정밀도)	/D	848	4	6.4	7.5	7.5	---
도 → 라디안 변환(배정밀도)	RADD	849	3	6.1	6.5	6.5	---
라디안 → 도 변환(배정밀도)	DEGD	850	3	6.0	6.4	6.4	---
사인(배정밀도)	SIND	851	3	14.7	21.5	9.5	0° 지정 시
				20.4	35.4	9.5	45° 지정 시
				18.5	35.0	9.6	90° 지정 시

AI 번역본
-For Reference Only-

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
코사인(배정밀도)	COSD	852	3	14.1	20.6	9.7	0° 지정 시
				19.6	29.9	9.8	45° 지정 시
				19.1	29.8	9.8	90° 지정 시
탄젠트(배정밀도)	TAND	853	3	7.3	9.4	7.7	0° 지정 시
				27.4	50.3	10.2	45° 지정 시
아크 사인(배정밀도)	ASIND	854	3	7.5	9.8	9.6	0° 지정 시
				55.0	75.2	10.1	45° 지정 시
				6.1	8.3	8.7	90° 지정 시
아크 코사인(배정밀도)	ACOSD	855	3	8.3	10.9	10.2	0° 지정 시
				55.9	72.8	10.2	45° 지정 시
				43.7	72.8	10.2	90° 지정 시
아크 탄젠트(배정밀도)	ATAND	856	3	6.1	7.4	7.4	0° 지정 시
				29.7	36.5	9.3	45° 지정 시
제곱근(배정밀도)	SQRTD	857	3	16.6	23.4	8.4	---
지수(배정밀도)	EXPD	858	3	39.7	58.4	10.3	---
로그(배정밀도)	LOGD	859	3	35.5	52.2	10.6	---
지수 거듭제곱(배정밀도)	PWRD	860	4	66	99	12.0	---

A-2-15 테이블 데이터 처리 명령어

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
스택 설정	SSET	630	3	7.6	9.4	8.9	스택 영역에 5 워드 지정 시
				107	65	63	스택 영역에 1,000 워드 지정 시
스택에 푸시	PUSH	632	3	4.9	5.9	5.1	---
선입선출	FIFO	633	3	4.8	5.0	3.9	스택 영역에 5 워드 지정 시
				231	178	177	스택 영역에 1,000 워드 지정 시
후입선출	LIFO	634	3	5.3	7.1	6.0	---
레코드 테이블 정의	DIM	631	5	11.1	19.7	6.7	---
레코드 위치 설정	SETR	635	4	3.8	5.5	4.7	---
레코드 번호 취득	GETR	636	4	4.6	7.9	6.6	---
데이터 검색	SRCH	181	4	13.9	25.0	4.9	1 워드 검색 시
				1940	3257	670	1,000 워드 검색 시*1
바이트 교체	SWAP	637	3	10.1	17.5	3.3	1 워드 교체 시
				1421	2098	837	1,000 워드 교체 시*1

AI 번역본
-For Reference Only-

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(μs)			조건
				CJ2H	CJ2M	CJ3M	
최대값 검색	MAX	182	4~5	4.8	5.8	4.6	검색 대상 값의 개수: 1
				465	672	670	검색 대상 값의 개수: 1,000*1
더블 최대값 검색	MAXL		4~5	4.8	5.8	4.8	검색 대상 값의 개수: 1
				465	773	706	검색 대상 값의 개수: 1,000*1
부동소수점 최대값 검색	MAXF		4~5	5.2	6.5	4.8	검색 대상 값의 개수: 1
				682	1090	840	검색 대상 값의 개수: 1,000*1
더블 최대값 검색 부동소수점	MAXD		4~5	5.4	6.4	6.4	검색 대상 값의 개수: 1
				1435	2333	1950	검색 대상 값의 개수: 1,000*1
최소값 검색	MIN	183	4~5	4.8	5.8	4.6	검색 대상 값의 개수: 1
				465	677	670	검색 대상 값의 개수: 1,000*1
더블 최소값 검색	MINL		4~5	4.8	5.9	4.7	검색 대상 값의 개수: 1
				189	774	705	검색 대상 값의 개수: 1,000*1
부동소수점 최소값 검색	MINF		4~5	5.2	6.5	4.9	검색 대상 값의 개수: 1
				683	1091	830	검색 대상 값의 개수: 1,000*1
더블 최소값 검색 부동소수점	MIND		4~5	5.2	6.4	6.4	검색 대상 값의 개수: 1
				1402	2303	1950	검색 대상 값의 개수: 1,000*1
합계	SUM	184	4	17.5	31.3	5.2	1 워드 가산 시
				900	1696	393	1,000 워드 가산 시*1
프레임 체크섬	FCS	180	4	14.1	25.2	3.9	테이블 길이 1 워드인 경우
				1235	2089	650	테이블 길이 1,000 워드인 경우*1
스택 사이즈 읽기	SNUM	638	3	4.5	5.3	4.4	---
스택 데이터 읽기	SREAD	639	4	4.6	5.4	4.5	---
스택 데이터 덮어쓰기	SWRIT	640	4	4.3	5.0	4.1	---

AI 번역본
-For Reference Only-

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
스택 데이터 삽입	SINS	641	4	8.2	9.3	8.8	---
				275	256	179	1,000 워드 테이블인 경우
스택 데이터 삭제	SDEL	642	4	6.1	7.8	6.5	---
				247	180	179	1,000 워드 테이블인 경우

*1 명령어 실행 시간은 데이터 양의 영향을 크게 받습니다. 이는 사이클 타임에 영향을 줍니다. 사이클 타임에 미치는 영향을 줄이려면 백그라운드 실행을 지정할 수 있습니다. 자세한 내용은 10-2-5 백그라운드 실행을 참조하십시오.

A-2-16 트래킹 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
부호 없는 1워드 레코드 서치 명령어	RSRCH	360~364	6	13.9	15.9	11.4	레코드 수: 1
				504	585	566	레코드 수: 1,000
부호 없는 2워드 레코드 서치 명령어	RSRCH2	370~374	6	14.7	17.6	14.4	레코드 수: 1
				838	932	864	레코드 수: 1,000
부호 없는 4워드 레코드 서치 명령어	RSRCH4	380~384	6	17.0	19.1	15.0	레코드 수: 1
				1544	1684	1653	레코드 수: 1,000
부호 없는 1워드 레코드 정렬	RSORT	203	5	149	156	130	레코드 100건, 분할 정렬 무효, "99, 98, 97...0"을 "0, 1, 2...99"로 정렬(최악의 경우)
부호 없는 2워드 레코드 정렬	RSORT2	204	5	250	249	219	
부호 없는 4워드 레코드 정렬	RSORT4	205	5	457	440	405	

A-2-17 데이터 제어 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
PID 제어	PID	190	4	297	526	10.2	초기 실행 시
				234	423	64.1	입력 ON이고 샘플링 중
				71	117	12.0	입력 ON이고 샘플링 중이 아님
				7.2	10.5	1.3	입력 OFF
오토튜닝 포함 PID 제어	PIDAT	191	4	302	600	10.4	초기 실행 시
				237	428	65.3	입력 ON이고 샘플링 중
				73	118	12.5	입력 ON이고 샘플링 중이 아님
				7.3	10.5	1.3	입력 OFF
				120	203	5.0	오토튜닝 초기 실행 시
리미트 제어	LMT	680	4~5	10.8	18.3	5.1	---

-For Reference Only-

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
데드 밴드 제어	BAND	681	4~5	11.2	19.2	5.5	---
데드 존 제어	ZONE	682	4~5	10.9	17.7	5.2	---
시간 비례 출력	TPO	685	4	6.9	10.2	0.77	입력 OFF
				37	65	9.6	입력 ON이고 듀티 지정 또는 출력 제한 무효
스케일링	SCL	194	4	7.6	9.3	7.4	---
스케일링 2	SCL2	486	4	6.8	9.2	8.0	---
스케일링 3	SCL3	487	4	7.8	9.9	7.5	---
평균	AVG	195	4	22	40	7.8	연산 1회의 평균
				212	351	113	연산 64회의 평균

A-2-18 서브루틴 명령어

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
서브루틴 콜	SBS	091	2	0.90	2.8	2.8	---
서브루틴 엔트리	SBN	092	2	2.8	4.1	3.7	---
서브루틴 리턴	RET	093	1	0.43	2.0	2.0	---
매크로	MCRO	099	4	16.8	21.7	20.9	---
글로벌 서브루틴 리턴	GSBS	750	2	0.90	2.8	2.8	---
글로벌 서브루틴 콜	GSBN	751	2	2.7	3.6	3.2	---
글로벌 서브루틴 엔트리	GRET	752	1	0.43	2.0	2.0	---

AI 번역본
-For Reference Only-

A-2-19 인터럽트 제어 명령어

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(μs)			조건
				CJ2H	CJ2M	CJ3M	
인터럽트 마스크 설정	MSKS	690	3	10.6	22.1	5.6	---
인터럽트 마스크 읽기	MSKR	692	3	9.6	14.8	1.3	---
인터럽트 클리어	CLI	691	3	10.1	21.6	5.6	---
인터럽트 금지	DI	693	1	10.3	20.4	5.4	---
인터럽트 허가	EI	694	1	9.3	16.0	5.0	---

A-2-20 고속 카운터/펄스 출력 명령어

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(μs)			조건
				CJ2H	CJ2M	CJ3M	
모드 제어	INI	880	4	---	7.4	7.4	고속 카운터 비교 개시
					4.0	4.0	고속 카운터 비교 정지
					9.9	9.9	펄스 출력의 현재값 변경
					8.0	8.0	고속 카운터 현재값 변경
					5.0	5.0	카운터 모드 인터럽트 입력의 현재값 변경
					9.2	9.2	펄스 출력 정지
					5.2	5.2	PWM 출력 정지
					48.2	48.2	원점 서치/원점 복귀 설정 변경

AI 번역본
-For Reference Only-

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(μs)			조건
				CJ2H	CJ2M	CJ3M	
펄스 현재값 읽기	PRV	881	4	---	7.6	7.6	펄스 출력 현재값 읽기
					6.3	6.3	고속 카운터 스테이터스 읽기
					3.7	3.7	카운터 모드 인터럽트 입력의 현재값 읽기
					7.1	7.1	펄스 출력 스테이터스 읽기
					6.3	6.3	고속 카운터 스테이터스 읽기
					4.1	4.1	PWM 스테이터스 읽기
					17.5	17.5	고속 카운터 8영역 범위 비교 결과 읽기
					34.6	34.6	고속 카운터 32영역 범위 비교 결과 읽기
					4.9	4.9	고속 카운터 0의 주파수 읽기
카운터 주파수 변환	PRV2	883	4	---	5.2	5.2	---

AI 번역본
-For Reference Only-

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
비교표 등록	CTBL	882	4	---	31.7	31.7	목표값 1건의 목표값 테이블 등록 및 비교 개시
					1528	1528	목표값 48건의 목표값 테이블 등록 및 비교 개시
					45.0	45.0	8영역의 범위 테이블 등록 및 비교 개시
					150.8	150.8	32영역의 범위 테이블 등록 및 비교 개시
					26.2	26.2	목표값 1건의 목표값 테이블 등록. 단 비교는 개시하지 않음
					1520	1520	목표값 48건의 목표값 테이블 등록. 단 비교는 개시하지 않음
					41.8	41.8	범위 8건의 범위 비교 테이블 등록. 단 비교는 개시하지 않음
					150.0	150.0	범위 32건의 범위 비교 테이블 등록. 단 비교는 개시하지 않음
속도 출력	SPED	885	4	---	23.5	23.5	연속 모드
					24.6	24.6	독립 모드
펄스 설정	PULS	886	4	---	8.2	8.2	---
위치 결정	PLS2	887	5	---	39.4	39.4	---
가속 제어	ACC	888	4	---	35.0	35.0	연속 모드
					44.6	44.6	독립 모드

AI 번역본
-For Reference Only-

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
원점 서치	ORG	889	3	---	28.1	28.1	원점 서치
					23.6	23.6	원점 복귀
PWM 출력	PWM	891	4	---	7.8	7.8	---
인터럽트 피딩	!FEED	892	4	---	42.4	42.4	---

A-2-21 스텝 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
스텝 정의	STEP	008	2	8.7	10.6	5.7	스텝 제어 비트 ON
				8.7	9.8	5.7	스텝 제어 비트 OFF
스텝 개시	SNXT	009	2	2.2	2.8	2.8	---

A-2-22 기본 I/O 유닛 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
I/O 리프레시	IORF	097	3	10.1	12.2	10.3	기본 I/O 유닛의 1 워드 리프레시(IN)
				10.5	13.0	5.8	기본 I/O 유닛의 1 워드 리프레시(OUT)
특수 I/O 유닛 I/O 리프레시	FIORF	225	2	*1	*1	*1	---
CPU 버스 I/O 리프레시	DLNK	226	4	234	256	158	1 워드 할당 시
7세그먼트 디코더	SDEC	078	4	2.5	3.3	1.5	---
디지털 스위치 입력	DSW	210	6	24.8	39.6	6.2	4자리, 데이터 입력값: 0
				24.8	40.2	6.1	8자리, 데이터 입력값: 00
텐키 입력	TKY	211	4	7.2	9.7	0.93	데이터 입력값: 00
				6.5	8.6	1.74	데이터 입력값: FF
16진 키 입력	HKY	212	5	25.9	40.8	5.5	데이터 입력값: 00
				25.9	41.0	6.6	데이터 입력값: FF
매트릭스 입력	MTR	213	5	25.0	38.5	4.7	데이터 입력값: 00
				25.0	38.5	6.1	데이터 입력값: FF
7세그먼트 표시 출력	7SEG	214	5	31.4	51.9	7.1	4자리
				34.6	59.4	8.8	8자리
아날로그 입력 다이렉트 변환 (CJ1W-AD042용)	AIDC	216	3	30.5	31.7	31.7	아날로그 입력 번호: 1, 사용하는 아날로그 입력 수: 4
				45.5	46.6	46.6	아날로그 입력 번호: 0, 사용하는 아날로그 입력 수: 4

AI 번역본
For Reference Only

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
아날로그 출력 다이렉트 변환 (CJ1W-DA042V용)	AODC	217	3	28.9	29.7	29.7	아날로그 출력 번호: 1, 사용하는 아날로그 출력 수: 4
				50.0	50.6	50.6	아날로그 출력 번호: 0, 사용하는 아날로그 출력 수: 4
PCU 고속 위치 결정(CJ1W-NC□□4 또는 CJ1W-NC□81 전용)	NCDMV	218	4	81.7	95.3	54.7	---
PCU 위치 결정 트리거(CJ1W-NC□81 전용)	NCDTR	219	3	22.9	25.5	23.2	---
인텔리전트 I/O 읽기	IORD	222	4	*1	*1	*1	---
인텔리전트 I/O 쓰기	IOWR	223	4	*1	*1	*1	---

*1 IORD, IORW, FIORF 명령어의 실행은 해당 명령어가 실행되는 특수 I/O 유닛에 따라 달라집니다.

A-2-23 시리얼 통신 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
프로토콜 매크로	PMCR	260	5	57.8	97.8	20.4	다이렉트 지정
				77	132	25.1	오퍼랜드 지정, 송신 1 워드, 수신 1 워드
프로토콜 매크로 2	PMCR2	264	6	49.5	96.0	22.0	다이렉트 지정
				69	129	28.3	오퍼랜드 지정, 송신 1 워드, 수신 1 워드
송신	TXD	236	4	57.5	93.8	8.7	1 바이트 송신 시
				517	947	165	256 바이트 송신 시
수신	RXD	235	4	79	128	14.0	1 바이트 저장 시
				570	1033	157	256 바이트 저장 시
시리얼 통신 유닛/시리얼 포트 경유 다이렉트 송신	TXDU	256	4	75	130	21.1	1 바이트 송신 시
시리얼 통신 유닛/시리얼 포트 경유 다이렉트 수신	RXDU	255	4	74	128	21.9	1 바이트 저장 시
시리얼 통신 유닛 경유 다이렉트 송신	DTXDU	262	4	25.8	37.0	18.3	1 바이트 송신 시
				179	203	185	256 바이트 송신 시
시리얼 통신 유닛 경유 다이렉트 수신	DRXDU	261	4	27.8	39.7	12.5	1 바이트 저장 시
				188	205	45.7	256 바이트 저장 시
시리얼 포트 셋업 변경	STUP	237	3	233	276	231	CPU 유닛의 COM 포트 대상

AI 번역본
-For Reference Only-

A-2-24 네트워크 명령어

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(μs)			조건
				CJ2H	CJ2M	CJ3M	
네트워크 송신	SEND	090	4	44.3	79.4	21.6	---
네트워크 송신 2	SEND2	491	5	43.4	82.8	23.3	---
네트워크 수신	RECV	098	4	43.9	79.9	21.8	---
네트워크 수신 2	RECV2	492	5	44.5	82.8	23.5	---
커맨드 송신	CMND	490	4	52.7	95.1	25.4	---
커맨드 송신 2	CMND2	493	5	53.0	98.1	26.9	---
익스플리시트 메시지 송신	EXPLT	720	4	78	134	28.6	---
익스플리시트 어트리뷰트 취득	EGATR	721	4	74	127	27.4	---
익스플리시트 어트리뷰트 설정	ESATR	722	3	69	117	24.4	---
익스플리시트 워드 읽기	ECHRD	723	4	65	110	23.7	---
익스플리시트 워드 쓰기	ECHWR	724	4	64	110	23.4	---

A-2-25 파일 메모리 명령어

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(μs)			조건
				CJ2H	CJ2M	CJ3M	
데이터 파일 읽기	FREAD	700	5	217	372	49.7	바이너리
데이터 파일 쓰기	FWRIT	701	5	216	366	46.9	바이너리
텍스트 파일 쓰기	TWRIT	704	5	205	370	48.9	---

AI 번역본
-For Reference Only-

A-2-26 표시 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
메시지 표시	MSG	046	3	6.9	10.5	1.5	메시지 표시 시
				6.6	9.5	1.2	표시된 메시지 삭제 시

A-2-27 시계 명령어

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
캘린더 가산	CADD	730	4	15.6	22.5	14.5	---
캘린더 감산	CSUB	731	4	16.4	24.9	15.1	---
시간 → 초 변환	SEC	065	3	3.6	4.1	2.9	---
초 → 시간 변환	HMS	066	3	3.5	4.0	2.9	---
시계 조정	DATE	735	2	29.6	53.2	20.4	---

A-2-28 디버깅 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
트레이스 메모리 샘플링	TRSM	045	1	8.9	12.6	9.0	1 비트 및 0 워드 샘플링 시
				31.6	33.1	25.8	31 비트 및 6 워드 샘플링 시
				38.8	39.2	29.8	31 비트 및 16 워드 샘플링 시

A-2-29 고장 진단 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
고장 알람	FAL	006	3	7.9	14.7	2.3	에러 기록 시
				14.7	22.3	6.1	에러 삭제 시(우선순위 순)
				12.9	22.5	6.1	에러 삭제 시(전체 에러)
				117	210	24.7	에러 삭제 시 (개별)
중대 고장 알람	FALS	007	3	---	---	---	---

-For Reference Only-

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
고장점 검출	FPD	269	4	111	188	39.2	비트 어드레스 출력, 시간 감시
				107	202	50.5	비트 어드레스 출력, 최초 에러 검출
				129	242	52.0	메시지 문자 출력, 시간 감시
				159	244	128.2	메시지 문자 출력, 최초 에러 검출

A-2-30 기타 명령어

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
캐리 세트	STC	040	1	0.048	0.060	0.06	---
캐리 클리어	CLC	041	1	0.048	0.060	0.06	---
EM 뱅크 선택	EMBC	281	2	7.6	14.6	7.1	---
최대 사이클 타임 연장	WDT	094	2	7.6	17.1	5.0	---
조건 플래그 저장	CCS	282	1	5.8	8.3	4.9	---
조건 플래그 로드	CCL	283	1	6.4	9.9	4.9	---
CV로부터 어드레스 변환	FRMCV	284	3	9.4	15.7	2.5	---
CV로 어드레스 변환	TOCV	285	3~4	10.3	18.2	3.0	---
주변 서비스 금지	IOSP	287	1	---	---	---	---
주변 서비스 허가	IORS	288	1	---	---	---	---

A-2-31 블록 프로그래밍 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
블록 프로그램 개시	BPRG	096	2	7.8	14.1	13.1	---
블록 프로그램 종료	BEND	801	1	8.8	13.4	13.4	---
블록 프로그램 일시 정지	BPPS	811	2	5.4	8.4	8.4	---
블록 프로그램 재개	BPRS	812	2	3.6	4.8	4.7	---
조건부 블록 탈출	(실행 조건) EXIT	806	1	8.6	13.2	13.2	블록 탈출(입력 조건 ON)
				2.0	2.6	2.0	블록 탈출 안 함 (입력 조건 OFF)

For Reference Only

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
조건부 블록 탈출	EXIT(비트 어드레스)	806	2	9.8	14.8	14.8	블록 탈출(비트 ON)
				3.6	4.2	3.9	블록 탈출 안 함(비트 OFF)
조건부 블록 탈출 (NOT)	EXIT NOT (비트 어드레스)	806	2	3.6	4.3	3.9	블록 탈출(비트 OFF)
				8.9	14.9	14.9	블록 탈출 안 함(비트 ON)
분기	IF(실행 조건)	802	1	1.9	2.4	1.4	IF 참(입력 조건 ON)
				3.8	6.4	7.0	IF 거짓(입력 조건 OFF)
	IF(비트 어드레스)	802	2	3.2	4.0	3.2	IF 참(비트 ON)
				5.1	8.0	1.8	IF 거짓(비트 OFF)
분기 (NOT)	IF NOT(비트 어드레스)	802	2	5.1	8.2	8.2	IF 참(비트 OFF)
				3.2	4.1	1.9	IF 거짓(비트 ON)
분기	ELSE	803	1	3.5	5.7	3.1	IF 참
				5.3	7.3	6.4	IF 거짓
분기	IEND	804	1	5.3	8.5	3.1	IF 참
				2.0	2.4	1.9	IF 거짓
1 사이클 대기	WAIT (실행 조건)	805	1	10.0	15.9	14.8	대기하지 않음(입력 조건 ON)
				1.4	1.9	0.88	대기(입력 조건 OFF)
	WAIT(비트 어드레스)	805	2	9.2	13.5	13.5	대기하지 않음(비트 ON)
				2.6	3.7	2.8	대기(비트 OFF)
1 사이클 대기 (NOT)	WAIT NOT (비트 어드레스)	805	2	9.2	13.5	13.5	대기하지 않음(비트 OFF)
				2.8	3.7	2.8	대기(비트 ON)
100ms 타이머 대기	TIMW	813	3	15.6	22.9	22.4	디폴트 설정
				16.0	23.2	15.7	통상 실행
	TIMWX	816	3	15.1	21.7	15.9	디폴트 설정
				16.0	23.2	22.4	통상 실행
10ms 타이머 대기	TMHW	815	3	15.7	22.6	22.4	디폴트 설정
				17.5	24.9	15.8	통상 실행
	TMHWX	817	3	15.2	22.1	21.8	디폴트 설정
				16.4	23.4	15.9	통상 실행
카운터 대기	CNTW	814	4	13.7	20.5	20.2	디폴트 설정
				13.4	19.8	16.0	통상 실행
	CNTWX	818	4	13.1	19.5	19.6	디폴트 설정
				13.5	19.7	16.0	통상 실행
루프 제어	LOOP	809	1	4.6	9.1	5.1	---
루프 제어	LEND (실행 조건)	810	1	4.2	8.6	5.3	루프하지 않음(입력 조건 ON)
				3.9	6.5	5.4	루프함(입력 조건 OFF)
	LEND(비트 어드레스)	810	2	6.7	10.4	7.0	루프하지 않음(비트 ON)
				6.6	8.2	10.3	루프함(비트 OFF)
루프 제어 (NOT)	LEND NOT (비트 어드레스)	810	2	6.7	10.9	10.5	루프하지 않음(비트 OFF)
				6.6	8.2	7.0	루프함(비트 ON)

AI 번역본 -For Reference Only

A-2-32 문자열 처리 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(μs)			조건
				CJ2H	CJ2M	CJ3M	
문자열 전송	MOV$	664	3	31.5	58.3	14.9	1 문자 전송 시*1
문자열 결합	+$	656	4	56	104	23.1	1 문자 + 1 문자*1
문자열 좌측 취득	LEFT$	652	4	33.5	62.2	14.2	2 문자에서 1 문자 취득 시*1
문자열 우측 취득	RGHT$	653	4	33.4	62.1	21.9	2 문자에서 1 문자 취득 시*1
문자열 중간 취득	MID$	654	5	32.3	60.8	21.6	3 문자에서 1 문자 취득 시*1
문자열 내 검색	FIND$	660	4	30.3	56.3	9.5	2 문자에서 1 문자 검색 시*1
문자열 길이	LEN$	650	3	14.0	24.9	4.7	1 문자 검출 시*1
문자열 치환	RPLC$	661	6	110	213	42.9	2 문자 중 첫 번째를 1 문자로 치환 시*1
문자열 삭제	DEL$	658	5	45.6	86.8	14.9	2 문자 중 선두 문자 삭제 시*1
문자열 교환	XCHG$	665	3	40.3	75.4	20.0	1 문자와 1 문자 교환 시*1
문자열 클리어	CLR$	666	2	15.9	28.3	6.4	1 문자 클리어 시*1
문자열 삽입	INS$	657	5	85	162	33.6	2 문자 중 첫 번째 뒤에 1 문자 삽입 시*1
문자열 비교 명령어	=$	670	4	27.0	50.9	7.4	1 문자와 1 문자 비교 시
	<>$	671					
	<$	672					
	<=$	673					
	>$	674					
	=$	675					

AI 번역본
-For Reference Only-

*1 명령어 실행 시간은 데이터 양의 영향을 크게 받습니다. 이는 사이클 타임에 영향을 줍니다. 사이클 타임에 미치는 영향을 줄이려면 백그라운드 실행을 지정할 수 있습니다. 자세한 내용은 10-2-5 백그라운드 실행을 참조하십시오.

A-2-33 태스크 제어 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
태스크 ON	TKON	820	2	12.5	19.0	2.8	사이클릭 태스크 지정 시
				13.6	22.7	3.9	엑스트라 태스크 지정 시
태스크 OFF	TKOF	821	2	240	393	11.5	사이클릭 태스크 지정 시
				15.5	25.8	3.9	엑스트라 태스크 지정 시

A-2-34 기종 변환 명령어

명령어	니모닉	FUN No.	길이 (스텝)	ON 실행 시간(µs)			조건
				CJ2H	CJ2M	CJ3M	
블록 전송	XFERC	565	4	6.7	8.2	7.1	1 워드 전송 시
				362	409	392	1,000 워드 전송 시
싱글 워드 분배	DISTC	566	4	4.6	5.3	4.5	데이터 분배
				6.0	7.3	6.2	스택 조작
데이터 수집	COLLC	567	4	5.3	6.5	5.6	데이터 분배
				4.5	13.0	11.5	스택 조작
				5.8	6.1	3.5	스택 조작 1 워드 FIFO 읽기
				42	142	124	스택 조작 1,000 워드 FIFO 읽기
비트 전송	MOVBC	568	4	4.9	5.7	4.6	---
비트 카운터	BCNTC	621	4	5.5	6.4	5.5	1 워드 카운트 시
				873	974	562	1,000 워드 카운트 시

AI 번역본
-For Reference Only-

A-2-35 특수 펑션 블록 명령어

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(μs)			조건
				CJ2H	CJ2M	CJ3M	
변수 ID 취득	GETID	286	4	7.6	12.5	2.8	---

A-2-36 SFC 명령어

명령어	니모닉	FUN No.	길이(스텝)	ON 실행 시간(μs)			조건
				CJ2H	CJ2M	CJ3M	
스텝 활성화	SA	784	2	9.1	11.8	5.4	---
스텝 비활성화	SE	785	2	9.1	11.9	5.4	---
설정 타이머 읽기	TSR	780	3	3.9	4.2	2.1	---
스텝 타이머 설정	TSW	781	3	5.7	8.2	2.8	---
SFC ON	SFCON	789	2	14.0	20.4	2.9	---
SFC OFF	SFCOFF	790	2	249	402	15.9	---
리셋 포함 SFC 일시 정지	SFCPR	793	2	249	405	15.9	---
리셋 없는 SFC 일시 정지	SFCPRN	791	2	249	405	15.8	---

A-2-37 펑션 블록 인스턴스 실행 시간

펑션 블록 정의를 작성하고 그 인스턴스를 사용자 프로그램에 복사한 경우, 인스턴스 실행이 사이클 타임에 미치는 영향은 다음 식으로 계산합니다.

인스턴스 실행이 사이클 타임에 미치는 영향
= 기동 시간(A)
+ I/O 파라미터 전송 처리 시간(B)
+ 펑션 블록 정의 내 명령어의 실행 시간(C)

A, B, C의 시간은 다음 표와 같습니다.

동작			CPU 유닛 형식	
			CJ2H-CPU6□(-EIP)	CJ2M-CPU□□ CJ3M-CPU□□
A	기동 시간	I/O 파라미터 전송을 포함하지 않는 기동 시간	3.3 μs	7.4 μs
B	I/O 파라미터 전송 처리 시간 데이터형은 괄호 안에 표시되어 있습니다.	1비트(BOOL) 입력 심볼 또는 출력 심볼	0.24 μs	0.88 μs
		1워드(INT, UINT, WORD) 입력 심볼 또는 출력 심볼	0.19 μs	0.88 μs
		2워드(DINT, UDINT, DWORD, REAL) 입력 심볼 또는 출력 심볼	0.19 μs	1.2 μs
		4워드(LINT, ULINT, LWORD, LREAL) 입력 심볼 또는 출력 심볼	0.38 μs	2.96 μs
		I/O 심볼	0.114 μs	0.4 μs
C	펑션 블록 정의의 명령어 실행 시간	명령어 처리 시간의 합계(표준 사용자 프로그램과 동일)		

For Reference Only

예: CJ2H-CPU67-EIP
1워드 데이터형(INT) 입력 변수: 3
1워드 데이터형(INT) 출력 변수: 2
펑션 블록 정의부의 명령어 처리 시간 합계: 10 μs 인스턴스 1개의 실행 시간 = 3.3 μs
+ (3 + 2) × 0.19 μs + 10 μs = 14.25 μs

주 실행 시간은 여러 인스턴스의 개수에 따라 늘어납니다. 동일한 펑션 블록 정의를 여러 위치에 복사한 경우입니다.



참고

펑션 블록의 프로그램 스텝 수

펑션 블록 정의를 작성하고 그 인스턴스를 사용자 프로그램에 복사한 경우, 프로그램 스텝 수는 다음 식으로 계산합니다.

스텝 수
= 인스턴스 수 × (호출부 사이즈 m + I/O 파라미터 전송부 사이즈 n × 파라미터 수) +
펑션 블록 정의의 명령어 스텝 수 p (주 참조)

주 펑션 블록 정의의 명령어 스텝 수(p)는 후속 인스턴스에서 줄어들지 않습니다. 동일한 펑션 블록 정의를 여러 위치에 복사한 경우(즉 여러 인스턴스의 경우)입니다. 따라서 위 식에서 인스턴스 수에는 펑션 블록 정의의 명령어 스텝 수(p)를 곱하지 않습니다.

내용			CJ2/CJ3 CPU 유닛
m	호출부		57 스텝
n	I/O 파라미터 전송부 데이터형은 괄호 안에 표시되어 있습니다.	1비트(BOOL) 입력 심볼 또는 출력 심볼	6 스텝
		1워드(INT, UINT, WORD) 입력 심볼 또는 출력 심볼	6 스텝
		2워드(DINT, UDINT, DWORD, REAL) 입력 심볼 또는 출력 심볼	6 스텝
		4워드(LINT, ULINT, LWORD, LREAL) 입력 심볼 또는 출력 심볼	18 스텝
		I/O 심볼	6 스텝
p	펑션 블록 정의의 명령어 스텝 수	명령어 스텝 수의 합계(표준 사용자 프로그램과 동일) + 27 스텝.	

예:
1워드 데이터형(INT) 입력 변수: 5
1워드 데이터형(INT) 출력 변수: 5
펑션 블록 정의부: 100 스텝
인스턴스 1개의 스텝 수 = 57 + (5 + 5) × 6 스텝 + 100 스텝 + 27 스텝 = 244 스텝

AI 번역본
-For Reference Only-

A-3 보조 기억 영역

A000~A447: 읽기 전용 영역, A448~A1000: 읽기/쓰기 영역

A-3-1 읽기 전용 영역(시스템 설정)

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A0	---	10 ms 증분 프리런 타이머	이 워드에는 전원 ON 후에 사용되는 시스템 타이머가 저장됩니다. 전원 ON 시 A0은 0000 hex로 설정되고, 이 값은 10 ms마다 자동으로 1씩 증분됩니다. 값은 FFFF hex(655,350 ms)에 도달한 뒤 0000 hex로 돌아가며, 그 후에도 계속 10 ms마다 자동으로 1씩 증분됩니다. 운전 모드를 RUN 모드로 전환해도 타이머는 계속 증분됩니다. 예: 타이머 명령어를 사용하지 않고 처리 A와 처리 B 사이의 간격을 셀 수 있습니다. 이는 처리 A 시점의 A000 값과 처리 B 시점의 A000 값의 차를 계산하여 구합니다. 간격은 10 ms 단위로 셉니다.	유지	클리어	전원 ON 후 10 ms마다
A1	---	100 ms 증분 프리런 타이머	이 워드에는 전원 ON 후에 사용되는 시스템 타이머가 저장됩니다. 전원 ON 시 A1은 0000 hex로 설정되고, 이 값은 100 ms마다 자동으로 1씩 증분됩니다. 값은 FFFF hex(6,553,500 ms)에 도달한 뒤 0000 hex로 돌아가며, 그 후에도 계속 100 ms마다 자동으로 1씩 증분됩니다. 운전 모드를 RUN 모드로 전환해도 타이머는 계속 증분됩니다.	유지	클리어	전원 ON 후 100 ms마다
A2	---	1 s 증분 프리런 타이머	이 워드에는 전원 ON 후에 사용되는 시스템 타이머가 저장됩니다. 전원 ON 시 A2는 0000 hex로 설정되고, 이 값은 1 s마다 자동으로 1씩 증분됩니다. 값은 FFFF hex(65,535 s)에 도달한 뒤 0000 hex로 돌아가며, 그 후에도 계속 1 s마다 자동으로 1씩 증분됩니다. 운전 모드를 RUN 모드로 전환해도 타이머는 계속 증분됩니다.	유지	클리어	전원 ON 후 1 s마다
A50	A50.00~ A50.07	기본 I/O 유닛 정보, 랙 0 슬롯 0	부하 단락 보호 기능의 알람 출력이 발생하면 해당 비트가 ON됩니다. CJ1W-OD202에서는 최하위 4비트만(비트당 2점), CJ1W-OD212, OD204, MD232에서는 최하위 1비트만, CJ1W-OD232에서는 최하위 2비트만 사용됩니다. ON: 단락됨 OFF: 정상	---	---	매 사이클 리프레시됩니다.
	A50.08~ A50.15	기본 I/O 유닛 정보, 랙 0 슬롯 1		---	---	
A51~ A89	A51.00~ A89.15	기본 I/O 유닛 정보, 랙 0 슬롯 2~ 랙 7 슬롯 9		---	---	
A90~ A93	---	사용자 프로그램 일시	이 워드들에는 사용자 프로그램이 마지막으로 덮어써진 일시가 BCD로 저장됩니다. A90.00~A90.07: 초(00~59) A90.08~A90.15: 분(00~59) A91.00~A91.07: 시(00~23) A91.08~A91.15: 일(01~31) A92.00~A92.07: 월(01~12) A92.08~A92.15: 연(00~99) A93.08~A93.07: 요일 (00: 일요일, 01: 월요일, 02: 화요일, 03: 수요일, 04: 목요일, 05: 금요일, 06: 토요일) 배터리가 접속되어 있지 않으면 이 값은 불안정해집니다.	유지	유지	---

AI 번역본
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A94~ A97	---	파라미터 일시	이 워드들에는 파라미터가 마지막으로 덮어써진 일시가 BCD로 저장됩니다. 형식은 위와 같습니다 배터리가 접속되어 있지 않으면 이 값은 불안정해집니다.	유지	유지	---
A99	00	UM 읽기 보호 상태	PLC 내 사용자 프로그램 전체가 읽기 보호되어 있는지 나타냅니다. OFF: UM이 읽기 보호되어 있지 않음. ON: UM이 읽기 보호되어 있음.	유지	유지	보호를 설정 또는 해제할 때
	01	태스크 읽기 보호 상태	개별 태스크에 읽기 보호가 설정되어 있는지 나타냅니다. OFF: 태스크가 읽기 보호되어 있지 않음. ON: 태스크가 읽기 보호되어 있음.			---
	02	읽기 보호 설정 시의 프로그램 쓰기 보호 상태	프로그램이 쓰기 보호되어 있는지 나타냅니다. OFF: 쓰기 가능. ON: 쓰기 보호됨.			
	03	프로그램의 메모리 카드 백업 가부 상태	백업 프로그램 파일(.OBJ) 작성이 가능한지 불가능한지 나타냅니다. OFF: 가능. ON: 불가능.			
	14	태스크 간 IR/DR 동작	모든 태스크에서 인덱스 레지스터와 데이터 레지스터를 공유하려면 이 비트를 ON하십시오. 태스크마다 인덱스 레지스터와 데이터 레지스터를 별도로 사용하려면 이 비트를 OFF하십시오. OFF: 독립 ON: 공유(디폴트)			---
A100 ~ A199	---	에러 로그 영역	에러가 발생하면 에러 코드, 에러 내용, 에러 발생 일시가 에러 로그 영역에 저장됩니다. 최근 에러 20건의 정보를 저장할 수 있습니다. 각 에러 레코드는 5 워드를 차지하며, 이 5 워드의 기능은 다음과 같습니다. 첫 번째 워드: 에러 코드(비트 0~15) 두 번째 워드: 에러 내용(비트 0~15) 에러 내용: 상세 정보가 있는 보조 기억 영역 워드의 어드레스 또는 0000. 세 번째 워드: 분(비트 8~15), 초(비트 0~7) 초: 00~59, BCD 분: 00~59, BCD 네 번째 워드: 일(비트 8~15), 시(비트 0~7) 시: 00~23, BCD 일: 01~31, BCD Fifty 워드: 연(비트 8~15), 월(비트 0~7) 연: 00~99, BCD 월: 00~12, BCD 배터리가 접속되어 있지 않으면 에러 발생 시각이 불안정해집니다. FAL(006)과 FALS(007)로 발생시킨 에러도 이 에러 로그에 저장됩니다. 에러 로그 영역은 CX-Programmer에서 리셋할 수 있습니다. 에러 로그 영역이 가득 찬 상태(20건)에서 다른 에러가 발생하면, A100~A104의 가장 오래된 레코드가 클리어되고 나머지 19건의 레코드가 아래로 시프트된 뒤 새 레코드가 A195~A199에 저장됩니다.	유지	유지	에러 발생 시 리프레시됩니다. A50014 A300 A400

AI 번역본
-AI Version Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A200	A200.11	최초 사이클 플래그	PLC 운전이 개시된 후(예: 모드가 PROGRAM에서 RUN 또는 MONITOR로 전환된 후) 1 사이클 동안 ON됩니다. 최초 사이클 동안 ON	---	---	---
	A200.12	스텝 플래그	STEP(008)으로 스텝 실행이 개시되면 1 사이클 동안 ON됩니다. 이 플래그는 스텝 개시 시의 초기화 처리에 사용할 수 있습니다. STEP(008) 실행 후 최초 사이클 동안 ON됩니다.	클리어	---	---
	A200.14	태스크 기동 플래그	태스크가 STANDBY 또는 DISABLED 상태에서 READY 상태로 전환되면, 이 플래그는 해당 태스크 안에서 1 사이클 동안만 ON됩니다. ON: 최초 사이클 동안 ON(STANDBY 및 DISABLED 상태에서의 전환 포함) OFF: 그 밖의 경우 이 플래그와 A200.15의 유일한 차이는, 태스크가 STANDBY에서 READY 상태로 전환될 때도 이 플래그가 ON된다는 점입니다.	클리어	클리어	---
	A200.15	최초 태스크 기동 플래그	태스크가 처음 실행될 때 ON됩니다. 이 플래그는 현재 태스크가 처음 실행되는지 확인하여, 필요에 따라 초기화 처리를 수행할 수 있도록 하는 데 사용할 수 있습니다. ON: 최초 실행 OFF: 최초 실행이 아니거나 실행 중이 아님.	클리어	---	---
A201	A201.10	온라인 에디팅 대기 플래그	온라인 에디팅 처리가 대기 중일 때 ON됩니다. ON: 온라인 에디팅 대기 중 OFF: 온라인 에디팅 대기 중이 아님 대기 중에 다른 온라인 에디팅 커맨드를 수신하면, 그 커맨드는 기록되지 않고 에러가 발생합니다.	클리어	클리어	A527
	A201.11	온라인 에디팅 플래그	온라인 에디팅 처리가 실행 중일 때 ON됩니다. ON: 온라인 에디팅 중 OFF: 온라인 에디팅 중이 아님	클리어	클리어	A527
A202	A202.00 ~ A202.07	통신 포트 사용 가능 플래그	네트워크 명령어(SEND, RECV, CMND, PMCR, TXDU, RXDU) 또는 백그라운드 실행을 해당 포트 번호로 실행할 수 있을 때 ON됩니다. 비트 00~07은 통신 포트 0~7에 대응합니다. ON: 네트워크 명령어가 실행 중이 아님 OFF: 네트워크 명령어가 실행 중(포트 비지) 동일한 포트 번호로 네트워크 명령어를 둘 이상 프로그래밍한 경우, 해당 플래그를 실행 조건으로 사용하여 명령어가 동시에 실행되지 않도록 하십시오. (해당 포트 번호의 네트워크 명령어가 실행되는 동안, 그 포트의 플래그는 OFF됩니다.) 명령어가 실행되면 클리어됩니다.	클리어	---	---
	A202.08	CJ2/CJ3 명령어 사용 가능 플래그	CJ2/CJ3 명령어를 사용할 수 있을 때 ON됩니다. 이 플래그는 디폴트로 ON입니다. 이 플래그는 다음 명령어에만 사용할 수 있습니다: SEND2, CMND2, PMCR2, RECV2.	---	내부 상태에 따라 갱신(클리어)됩니다.	---
	A202.15	네트워크 통신 포트 할당 가능 플래그	통신 명령어(SEND, RECV, CMND, PMCR, TXDU, RXDU) 실행 시 자동 할당에 사용할 수 있는 통신 포트가 있을 때 ON됩니다. ON: 통신 포트 사용 가능 OFF: 통신 포트 사용 불가 통신 명령어를 9개 이상 동시에 사용할 때는, 통신 명령어를 실행하기 전에 이 플래그로 자동 할당 가능한 통신 포트가 있는지 확인하십시오.	클리어	---	---

사본 양도
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A203 ~ A210	---	통신 포트 완료 코드	이 워드들에는 네트워크 명령어(SEND, RECV, CMND, PMCR) 또는 백그라운드 실행이 실행되었을 때 해당 포트 번호의 완료 코드가 저장됩니다. (백그라운드 실행이 완료되면 해당 워드는 0000 hex로 클리어됩니다.) A203~A210 워드는 통신 포트 0~7에 대응합니다. 0 이외: 에러 코드 0000: 정상 익스플리시트 메시지 명령어(EXPLT, EGATR, ESATR, ECHRD, ECHWR)가 실행되었을 때 다음 코드가 저장됩니다. 익스플리시트 통신 에러 플래그가 OFF되면 0000 hex가 저장됩니다. 익스플리시트 통신 에러 플래그가 ON이고 네트워크 통신 에러 플래그가 ON이면, FINS 종료 코드가 저장됩니다. 익스플리시트 통신 에러 플래그가 ON이고 네트워크 통신 에러 플래그가 OFF이면, 익스플리시트 메시지 종료 코드가 저장됩니다. 통신 중에는 0000 hex가 저장되고, 실행이 완료되면 해당 코드가 저장됩니다. 이 코드는 운전을 개시할 때 클리어됩니다. (해당 포트 번호의 네트워크 명령어가 실행되면, 그 포트의 완료 코드는 0000으로 클리어됩니다.) 명령어가 실행되면 클리어됩니다.	클리어	---	---
A211	---	CJ2/CJ3 네트워크 통신에 사용 가능한 포트 수 명령어	사용할 수 있는 CJ2/CJ3 네트워크 통신 명령어 수가 0이 되면 A202.08이 OFF됩니다. 이 워드는 다음 명령어에만 사용할 수 있습니다: SEND2, CMND2, PMCR2, RECV2. 이 워드의 내용으로 통신 트래픽을 확인할 수 있습니다.	---	내부 상태에 따라 갱신(클리어)됩니다.	---
A213	A213.00 ~ A213.07	익스플리시트 통신 에러 플래그	익스플리시트 메시지 명령어(EXPLT, EGATR, ESATR, ECHRD, ECHWR) 실행 시 에러가 발생하면 ON됩니다. 비트 00~07은 통신 포트 0~7에 대응합니다. ON: 이상 종료 OFF: 정상 종료 익스플리시트 메시지를 송신할 수 없는 경우와, 익스플리시트 메시지에 대해 에러 응답이 반환된 경우 모두 해당 비트가 ON됩니다. 이 상태는 다음 익스플리시트 메시지 통신이 실행될 때까지 유지됩니다. 다음 익스플리시트 메시지 명령어가 실행되면 이 비트는 항상 OFF됩니다.	클리어	---	A219.00~A219.07 A203~A210
A214	A214.00 ~ A214.07	네트워크 통신 종료 후 최초 사이클 플래그	각 플래그는 통신이 완료된 후 1 사이클 동안만 ON됩니다. 비트 00~07은 포트 0~7에 대응합니다. A218에 저장된 사용 통신 포트 번호로 어느 플래그를 참조할지 판단하십시오. 1: 통신 종료 후 최초 사이클만 2: 그 밖의 상태 이 플래그들은 통신 명령어가 실행된 다음 사이클이 되어야 유효해집니다. 최소 1 사이클 뒤에 참조하십시오. 사용 중인 포트의 올바른 비트를 참조하려면 A218(사용 통신 포트 번호)에 지정된 포트 번호를 사용하십시오.	유지	클리어	---
A215	A215.00 ~ A215.07	네트워크 통신 에러 후 최초 사이클 플래그	각 플래그는 통신 에러가 발생한 후 1 사이클 동안만 ON됩니다. 비트 00~07은 포트 0~7에 대응합니다. A218에 저장된 사용 통신 포트 번호로 어느 플래그를 참조할지 판단하십시오. 에러 원인은 A203~A210에 저장된 통신 포트 완료 코드로 판단하십시오. ON: 통신 에러 후 최초 사이클만 OFF: 그 밖의 상태 이 플래그들은 통신 명령어가 실행된 다음 사이클이 되어야 유효해집니다. 최소 1 사이클 뒤에 참조하십시오. 사용 중인 포트의 올바른 비트를 참조하려면 A218(사용 통신 포트 번호)에 지정된 포트 번호를 사용하십시오.	유지	클리어	---

AI 번역본
-Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A216 및 A217	---	네트워크 통신 완료 코드 저장 어드레스	통신 명령어의 완료 코드는 이 워드들에 들어 있는 I/O 메모리 어드레스의 위치에 자동으로 저장됩니다. 이 어드레스를 인덱스 레지스터에 넣고 인덱스 레지스터를 통한 간접 지정으로 통신 완료 코드를 읽으십시오.	유지	클리어	---
A218	---	사용 통신 포트 번호	통신 포트 자동 할당을 사용하여 통신 명령어를 실행했을 때 사용된 통신 포트 번호를 저장합니다. 0000~0007 hex: 통신 포트 0~7	유지	클리어	---
A219	A219.00 ~ A219.07	통신 포트 에러 플래그	네트워크 명령어(SEND, RECV, CMND, PMCR) 실행 중에 에러가 발생하면 ON됩니다. 비트 00~07은 통신 포트 0~7에 대응합니다. ON: 에러 발생 OFF: 정상	유지	---	---
A220 ~ A259	A220.00 ~ 259.15	기본 I/O 유닛 입력 응답 시간	이 워드들에는 실제 입력 응답 시간이 저장됩니다. 0~17 16진수 A220~A259 워드의 값과 기본 I/O 유닛의 입력 응답 시간은 다음과 같이 대응합니다. 00 hex: 8 ms(디폴트 설정) 10 hex: 0 ms(필터 없음) 11 hex: 0.5 ms 12 hex: 1 ms 13 hex: 2 ms 14 hex: 4 ms 15 hex: 8 ms 16 hex: 16 ms 17 hex: 32 ms A220~A259 워드와 랙 번호 · 슬롯 번호는 다음과 같이 대응합니다. A220.00~A220.07: 랙 0 슬롯 0 A220.08~A220.15: 랙 0 슬롯 1 A221.00~A221.07: 랙 0 슬롯 2 A221.08~A221.15: 랙 0 슬롯 3 A222.00~A222.07: 랙 0 슬롯 4 A222.08~A222.15: 랙 0 슬롯 5 A223.00~A223.07: 랙 0 슬롯 6 A223.08~A223.15: 랙 0 슬롯 7 A224.00~A224.07: 랙 0 슬롯 8 A224.08~A224.15: 랙 0 슬롯 9 마찬가지로 다음 워드들은 랙 1~7의 슬롯 0~9에 대응합니다. A225~A229: 랙 1, 슬롯 0~9 A230~A234: 랙 2, 슬롯 0~9 A235~A239: 랙 3, 슬롯 0~9 A240~A244: 랙 4, 슬롯 0~9 A245~A249: 랙 5, 슬롯 0~9 A250~A254: 랙 6, 슬롯 0~9 A255~A259: 랙 7, 슬롯 0~9 PLC가 PROGRAM 모드인 상태에서 PLC 설정의 기본 I/O 유닛 입력 응답 시간 설정을 변경한 경우, 전원을 OFF했다가 다시 ON하지 않으면 PLC 설정의 설정값과 기본 I/O 유닛의 실제 값이 일치하지 않습니다. 그때는 이 워드들에서 실제 값을 모니터할 수 있습니다.	유지	기능 열을 참조하십시오.	PLC 설정 (기본 I/O 유닛 입력 응답 시간 설정)
A260	---	I/O 할당 상태	I/O 할당의 현재 상태, 즉 자동 I/O 할당인지 사용자 설정 I/O 할당인지 나타냅니다. 0000 hex: 자동 I/O 할당 BBBB hex: 사용자 설정 I/O 할당	유지	유지	---

참조용 번역본
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A261	A261.00	I/O 테이블 작성 에러 상세	ON: CPU 버스 유닛 셋업 에러 I/O 테이블이 정상적으로 생성되면 OFF됩니다. ON: CPU 버스 유닛 셋업 에러 OFF: I/O 테이블이 정상적으로 생성됨	유지	클리어	I/O 테이블 생성 시
	A261.02		ON: 최대 I/O 점수 오버플로 I/O 테이블이 정상적으로 생성되면 OFF됩니다. ON: 최대 I/O 점수 오버플로 OFF: I/O 테이블이 정상적으로 생성됨			A401.11(I/O 점수 과다)
	A261.03		ON: 동일한 유닛 번호가 두 번 이상 사용되었습니다. I/O 테이블이 정상적으로 생성되면 OFF됩니다. ON: 동일한 유닛 번호가 두 번 이상 사용되었습니다. OFF: I/O 테이블이 정상적으로 생성됨			A401.13(번호 중복)
	A261.04		ON: I/O 버스 에러 I/O 테이블이 정상적으로 생성되면 OFF됩니다. ON: I/O 버스 에러 OFF: I/O 테이블이 정상적으로 생성됨			A401.14 (I/O 버스 에러)
	A261.06		ON: SYSMAC BUS 슬레이브를 검출할 수 없어 I/O 테이블 에러 I/O 테이블이 정상적으로 생성되면 OFF됩니다. ON: SYSMAC BUS 슬레이브 없음 OFF: I/O 테이블이 정상적으로 생성됨			---
	A261.07		ON: 특수 I/O 유닛 에러 I/O 테이블이 정상적으로 생성되면 OFF됩니다. ON: 특수 I/O 유닛 에러 OFF: I/O 테이블이 정상적으로 생성됨			---
	A261.09		ON: I/O 검출이 완료되지 않았습니다. I/O 테이블이 정상적으로 생성되면 OFF됩니다. ON: I/O 검출이 완료되지 않았습니다. OFF: I/O 테이블이 정상적으로 생성됨			---
A262 및 A263	---	최대 사이클 타임(0.1 ms 단위)	이 워드들에는 PLC 운전 개시 이후의 최대 사이클 타임이 저장됩니다. 사이클 타임은 8자리 16진수로 기록되며, 상위 4자리는 A263에, 하위 4자리는 A262에 저장됩니다. 0~FFFFFFFF: 0~429,496,729.5 ms(0.1 ms 단위)	클리어	클리어	---
A264 및 A265	---	현재 사이클 타임(0.1 ms 단위)	이 워드들에는 현재 사이클 타임이 0.1 ms 단위로 저장됩니다. 시간은 매 사이클 8자리 16진수로 기록되며, 상위 4자리는 A264에, 하위 4자리는 A265에 저장됩니다. 00000000~FFFFFFFF(0.0~429,496,729.5 ms)	클리어	클리어	---
A266 및 A267	---	현재 사이클 타임(0.01 ms 단위)	이 워드들에는 현재 사이클 타임이 0.01 ms 단위로 저장됩니다. 시간은 매 사이클 8자리 16진수로 기록되며, 상위 4자리는 A266에, 하위 4자리는 A267에 저장됩니다. 00000000~FFFFFFFF(0.0~42,949,672.95 ms)	클리어	클리어	---

사내 참고용
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A270 및 A271	---	고속 카운터 0 현재값	고속 카운터 0의 현재값이 저장됩니다. 하위 4자리: A270, 상위 4자리: A271	클리어	클리어	· 공통 처리 중에 매 사이클 리프레시됩니다. · PRV(881) 명령어를 실행하여 현재값 또는 스테이터스를 읽을 때 리프레시됩니다. · PRV2(883) 명령어를 실행하여 고속 카운터 현재값을 총 펄스 수로 변환할 때 리프레시됩니다. · INI(880) 명령어를 실행하여 현재값 또는 링 카운터 최대값을 변경할 때 리프레시됩니다.
A272 및 A273	---	고속 카운터 1 현재값	고속 카운터 1의 현재값이 저장됩니다. 하위 4자리: A272, 상위 4자리: A273	클리어	클리어	· 공통 처리 중에 매 사이클 리프레시됩니다. · PRV(881) 명령어를 실행하여 현재값 또는 스테이터스를 읽을 때 리프레시됩니다. · INI(880) 명령어를 실행하여 현재값 또는 링 카운터 최대값을 변경할 때 리프레시됩니다.

AI 번역본
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A274	A274.00	고속 카운터 0 범위 비교 조건 1 범위 내 플래그	이 플래그들은 고속 카운터 0을 상한 · 하한이 있는 범위 비교 모드로 동작시킬 때, 현재값이 8개 범위 중 어느 하나에 들어 있는지를 나타냅니다. 단 범위 내 플래그는, 범위에 들어갈 때 또는 나갈 때 인터럽트 태스크를 실행하도록 고속 카운터가 설정되어 있는지와 무관하게, 비교값이 범위 안에 있으면 항상 ON됩니다. OFF: 범위 외, ON: 범위 내	클리어	클리어	· 공통 처리 중에 매 사이클 리프레시됩니다. · PRV(881) 명령어를 실행하여 범위 비교 결과를 읽을 때 리프레시됩니다. · INI(880) 명령어를 실행하여 현재값 또는 링 카운터 최대값을 변경할 때 리프레시됩니다. · 리셋될 때 리프레시됩니다.
	A274.01	고속 카운터 0 범위 비교 조건 2 범위 내 플래그				
	A274.02	고속 카운터 0 범위 비교 조건 3 범위 내 플래그				
	A274.03	고속 카운터 0 범위 비교 조건 4 범위 내 플래그				
	A274.04	고속 카운터 0 범위 비교 조건 5 범위 내 플래그				
	A274.05	고속 카운터 0 범위 비교 조건 6 범위 내 플래그				
	A274.06	고속 카운터 0 범위 비교 조건 7 범위 내 플래그				
	A274.07	고속 카운터 0 범위 비교 조건 8 범위 내 플래그				
	A274.08	고속 카운터 0 비교 동작	이 플래그는 고속 카운터 0에 대해 비교 동작이 실행 중인지 나타냅니다. OFF: 정지, ON: 비교 중	클리어	클리어	비교 동작이 개시 또는 정지할 때 리프레시됩니다.
	A274.09	고속 카운터 0 오버플로/언더플로 플래그	이 플래그는 고속 카운터 0의 현재값에 오버플로 또는 언더플로가 발생했음을 나타냅니다. (리니어 모드 카운트 범위에서만 사용합니다.) OFF: 정상, ON: 에러	클리어	클리어	· 현재값이 변경될 때 클리어됩니다. · 오버플로 또는 언더플로가 발생할 때 리프레시됩니다.
	A274.10	고속 카운터 0 카운트 방향	이 플래그는 고속 카운터가 현재 증분 중인지 감분 중인지 나타냅니다. 방향은 현재 사이클의 카운터 현재값과 직전 사이클의 현재값을 비교하여 판단합니다. OFF: 감분, ON: 증분	---	---	· 고속 카운터에 사용되는 설정이며, 카운터 동작 중에 유효합니다. · 공통 처리 중에 매 사이클 리프레시됩니다. · PRV(881) 명령어를 실행하여 현재값 또는 스테이터스를 읽을 때 리프레시됩니다.

AI 번역본
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A275	A275.00	고속 카운터 1 범위 비교 조건 1 범위 내 플래그	이 플래그들은 고속 카운터 1을 상한 · 하한이 있는 범위 비교 모드로 동작시킬 때, 현재값이 8개 범위 중 어느 하나에 들어 있는지를 나타냅니다. 단 범위 내 플래그는, 범위에 들어갈 때 또는 나갈 때 인터럽트 태스크를 실행하도록 고속 카운터가 설정되어 있는지와 무관하게, 비교값이 범위 안에 있으면 항상 ON됩니다. OFF: 범위 외, ON: 범위 내	클리어	클리어	· 공통 처리 중에 매 사이클 리프레시됩니다. · PRV(881) 명령어를 실행하여 범위 비교 결과를 읽을 때 리프레시됩니다. · INI(880) 명령어를 실행하여 현재값 또는 링 카운터 최대값을 변경할 때 리프레시됩니다. · 리셋될 때 리프레시됩니다.
	A275.01	고속 카운터 1 범위 비교 조건 2 범위 내 플래그				
	A275.02	고속 카운터 1 범위 비교 조건 3 범위 내 플래그				
	A275.03	고속 카운터 1 범위 비교 조건 4 범위 내 플래그				
	A275.04	고속 카운터 1 범위 비교 조건 5 범위 내 플래그				
	A275.05	고속 카운터 1 범위 비교 조건 6 범위 내 플래그				
	A275.06	고속 카운터 1 범위 비교 조건 7 범위 내 플래그				
	A275.07	고속 카운터 1 범위 비교 조건 8 범위 내 플래그				
	A275.08	고속 카운터 1 비교 실행 중 플래그	이 플래그는 고속 카운터 1에 대해 비교 동작이 실행 중인지 나타냅니다. OFF: 정지, ON: 비교 중	클리어	클리어	비교 동작이 개시 또는 정지할 때 리프레시됩니다.
	A275.09	고속 카운터 1 오버플로/언더플로 플래그	이 플래그는 고속 카운터 1의 현재값에 오버플로 또는 언더플로가 발생했음을 나타냅니다. (리니어 모드 카운트 범위에서만 사용합니다.) OFF: 정상, ON: 에러	클리어	클리어	· 현재값이 변경될 때 클리어됩니다. · 오버플로 또는 언더플로가 발생할 때 리프레시됩니다.
	A275.10	고속 카운터 1 카운트 방향	이 플래그는 고속 카운터 1이 현재 증분 중인지 감분 중인지 나타냅니다. 방향은 현재 사이클의 카운터 현재값과 직전 사이클의 현재값을 비교하여 판단합니다. OFF: 감분, ON: 증분	---	---	· 고속 카운터에 사용되는 설정이며, 카운터 동작 중에 유효합니다. · 공통 처리 중에 매 사이클 리프레시됩니다. · PRV(881) 명령어를 실행하여 현재값 또는 스테이터스를 읽을 때 리프레시됩니다.

AI 번역본
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A276 및 A277	---	펄스 출력 0 현재값	해당 펄스 출력 포트에서 출력된 펄스 수가 저장됩니다. 8000 0000~7FFF FFFF hex(−2,147,483,648~2,147,483,647) CW 방향으로 펄스를 출력 중일 때는 펄스 1개마다 현재값이 1씩 증분됩니다. CCW 방향으로 펄스를 출력 중일 때는 펄스 1개마다 현재값이 1씩 감분됩니다. 오버플로 후 현재값: 7FFF FFFF hex 언더플로 후 현재값: 8000 000 hex 하위 4자리: A276/A278/A322/A324 상위 4자리: A277/A279/A323/A325	클리어	클리어	· 공통 처리 중에 매 사이클 리프레시됩니다. · INI(880) 명령어를 실행하여 현재값을 변경할 때 리프레시됩니다. · 펄스 출력 리셋 비트가 ON되면 클리어됩니다. · 펄스 출력이 개시될 때(원점이 확정되지 않은 경우) 클리어됩니다. · PRV(881) 명령어를 실행하여 현재값 또는 스테이터스를 읽을 때 리프레시됩니다.
A278 및 A279	---	펄스 출력 1 현재값				
A280	A280.00	펄스 출력 0 펄스 출력 스테이터스 플래그	이 플래그는 ORG(889), ACC(888), PLS2(887), IFEED(892) 명령어에 따라 펄스 출력 0에서 펄스를 출력 중이고 출력 주파수가 단계적으로 변화(가속 또는 감속)하고 있을 때 ON됩니다. OFF: 정속, ON: 가속/감속 중	클리어	클리어	· 공통 처리 중에 매 사이클 리프레시됩니다.
	A280.01	펄스 출력 0 오버플로/언더플로 플래그	이 플래그는 펄스 출력 0의 현재값에 오버플로 또는 언더플로가 발생했음을 나타냅니다. OFF: 정상, ON: 에러	클리어	클리어	· INI(880) 명령어를 실행하여 현재값을 변경할 때 클리어됩니다. · 오버플로 또는 언더플로가 발생할 때 리프레시됩니다.
	A280.02	펄스 출력 0 펄스 수 설정 완료 플래그	PULS(886) 명령어로 펄스 출력 0의 출력 펄스 수가 설정되면 ON됩니다. OFF: 미설정, ON: 설정 완료	클리어	클리어	· PULS(886) 명령어가 실행될 때 리프레시됩니다. · 펄스 출력이 정지될 때 클리어됩니다.
	A280.03	펄스 출력 0 출력 완료 플래그	PULS(886), PLS2(887), IFEED(892) 명령어로 설정한 출력 펄스 수가 펄스 출력 0을 통해 출력되면 ON됩니다. OFF: 출력 미완료, ON: 출력 완료	클리어	클리어	독립 모드에서 펄스 출력이 개시 또는 완료될 때 리프레시됩니다.
	A280.04	펄스 출력 0 펄스 출력 중 플래그	펄스 출력 0에서 펄스를 출력 중일 때 ON됩니다. OFF: 정지, ON: 출력 중	클리어	클리어	펄스 출력이 개시 또는 정지할 때 리프레시됩니다.
	A280.05	펄스 출력 0 원점 미확정 플래그	펄스 출력 0의 원점이 확정되지 않았을 때 ON되고, 원점이 확정되면 OFF됩니다. OFF: 원점 확정, ON: 원점 미확정	운전이 개시될 때 ON 됩니다.	운전이 개시될 때 ON 됩니다.	· 펄스 출력이 리셋될 때 ON됩니다. · 원점 서치가 개시될 때 ON됩니다. · 리미트 입력을 수신하고 클리어가 설정되어 있을 때 ON됩니다. · 오버플로 또는 언더플로가 발생할 때 ON됩니다. · 원점 서치가 완료될 때 OFF 됩니다. · INI(880) 명령어를 실행하여 현재값을 변경할 때 OFF됩니다.

사 번역본
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A280	A280.06	펄스 출력 0 원점 정지 플래그	펄스 출력 0의 현재값이 원점(0)과 일치할 때 ON됩니다. OFF: 원점 정지 아님, ON: 원점 정지	유지	클리어	· 원점에서 정지할 때 ON됩니다. · 원점을 벗어날 때 OFF 됩니다.
	A280.07	펄스 출력 0 출력 정지 에러 플래그	펄스 출력 0의 원점 서치 기능에서 펄스 출력 중에 에러가 발생하면 ON됩니다. 펄스 출력 0 출력 정지 에러 코드가 A444에 기록됩니다. 0: 에러 없음, ON: 정지 에러	클리어	클리어	· 전원이 ON될 때 클리어됩니다. · 원점 서치가 개시될 때 클리어됩니다. · 원점 서치 중에 치명 펄스 출력 에러가 발생하면 ON됩니다. · 펄스 출력용 리미트 입력 신호가 PLC 설정에서 항상 유효하도록 설정되어 있으면, 이 플래그는 다음 조건에 따라 ON되거나 클리어됩니다: · 리미트 입력이 펄스 출력을 정지시키면 ON됩니다. · CW 또는 CCW 리미트 입력이 없는데 CW · CCW 리미트 입력 신호 플래그 중 하나가 ON이면 클리어됩니다.
	A280.08	펄스 출력 0 인터럽트 피딩 실행 중 플래그	IFEED(892) 명령어로 펄스 출력 0의 출력을 개시한 뒤 인터럽트 입력을 수신하면 이 플래그가 ON됩니다. OFF: 인터럽트 피딩 실행 중이 아님. ON: 인터럽트 피딩 실행 중.	클리어	클리어	· 인터럽트 피딩이 완료된 뒤 공통 처리 중에 클리어됩니다. · IFEED(892) 명령어로 펄스 출력을 개시한 뒤 인터럽트 입력이 ON되면 ON됩니다.
	A280.09	펄스 출력 0 인터럽트 피딩 에러 플래그	IFEED(892) 명령어로 펄스 출력 0의 출력을 개시한 뒤, 인터럽트 입력을 수신했을 때 또는 지정한 펄스 수만큼 이동했을 때 오버플로나 언더플로가 발생하면 이 플래그가 ON됩니다. ON: 에러 없음. OFF: 오버플로/언더플로 또는 지정한 펄스 수만큼 이동했음.	클리어	클리어	· IFEED(892) 명령어 처리가 개시될 때 클리어됩니다. · 원점이 확정된 상태에서 IFEED(892) 명령어로 운전을 개시한 뒤, 인터럽트 입력을 수신했을 때 오버플로나 언더플로가 발생하거나, 지정한 펄스 수만큼 이동하는 동안 오버플로나 언더플로가 발생하면 ON됩니다.

사내 참고용
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A281	A281.00	펄스 출력 1 펄스 출력 스테이터스 플래그	이 플래그는 ORG(889), ACC(888), PLS2(887), IFEED(892) 명령어에 따라 펄스 출력 1에서 펄스를 출력 중이고 출력 주파수가 단계적으로 변화(가속 또는 감속)하고 있을 때 ON됩니다. OFF: 정속, ON: 가속/감속 중	클리어	클리어	공통 처리 중에 매 사이클 리프레시됩니다.
	A281.01	펄스 출력 1 오버플로/언더플로 플래그	이 플래그는 펄스 출력 1의 현재값에 오버플로 또는 언더플로가 발생했음을 나타냅니다. OFF: 정상, ON: 에러	클리어	클리어	· INI(880) 명령어를 실행하여 현재값을 변경할 때 클리어됩니다. · 오버플로 또는 언더플로가 발생할 때 리프레시됩니다.
	A281.02	펄스 출력 1 펄스 수 설정 완료 플래그	PULS(886) 명령어로 펄스 출력 1의 출력 펄스 수가 설정되면 ON됩니다. OFF: 미설정, ON: 설정 완료	클리어	클리어	· PULS(886) 명령어가 실행될 때 리프레시됩니다. · 펄스 출력이 정지될 때 클리어됩니다.
	A281.03	펄스 출력 1 출력 완료 플래그	PULS(886), PLS2(887), IFEED(892) 명령어로 설정한 출력 펄스 수가 펄스 출력 1을 통해 출력되면 ON됩니다. OFF: 출력 미완료, ON: 출력 완료	클리어	클리어	독립 모드에서 펄스 출력이 개시 또는 완료될 때 리프레시됩니다.
	A281.04	펄스 출력 1 펄스 출력 중 플래그	펄스 출력 1에서 펄스를 출력 중일 때 ON됩니다. OFF: 정지, ON: 출력 중	클리어	클리어	펄스 출력이 개시 또는 정지할 때 리프레시됩니다.
	A281.05	펄스 출력 1 원점 미확정 플래그	펄스 출력 1의 원점이 확정되지 않았을 때 ON되고, 원점이 확정되면 OFF됩니다. OFF: 원점 확정, ON: 원점 미확정	운전이 개시될 때 ON 됩니다.	전원이 ON될 때 ON 됩니다.	· 펄스 출력이 리셋될 때 ON됩니다. · 원점 서치가 개시될 때 ON됩니다. · 리미트 입력을 수신하고 클리어가 설정되어 있을 때 ON됩니다. · 오버플로 또는 언더플로가 발생할 때 ON됩니다. · 원점 서치가 완료될 때 OFF 됩니다. · INI(880) 명령어를 실행하여 현재값을 변경할 때 OFF됩니다.
	A281.06	펄스 출력 1 원점 정지 플래그	펄스 출력 1의 현재값이 원점(0)과 일치할 때 ON됩니다. OFF: 원점 정지 아님, ON: 원점 정지	유지	클리어	· 원점에서 정지할 때 ON됩니다. · 원점을 벗어날 때 OFF 됩니다.

AI 번역본
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A281	A281.07	펄스 출력 1 출력 정지 에러 플래그	펄스 출력 1의 원점 서치 기능에서 펄스 출력 중에 에러가 발생하면 ON됩니다. 펄스 출력 1 출력 정지 에러 코드가 A445에 기록됩니다. 0: 에러 없음, ON: 정지 에러	유지	클리어	· 전원이 ON될 때 클리어됩니다. · 원점 서치가 개시될 때 클리어됩니다. · 원점 서치 중에 치명 펄스 출력 에러가 발생하면 ON됩니다. · 펄스 출력용 리미트 입력 신호가 PLC 설정에서 항상 유효하도록 설정되어 있으면, 이 플래그는 다음 조건에 따라 ON되거나 클리어됩니다: · 리미트 입력이 펄스 출력을 정지시키면 ON됩니다. · CW 또는 CCW 리미트 입력이 없는데 CW · CCW 리미트 입력 신호 플래그 중 하나가 ON이면 클리어됩니다.
	A281.08	펄스 출력 1 인터럽트 피딩 실행 중 플래그	IFEED(892) 명령어로 펄스 출력 1의 출력을 개시한 뒤 인터럽트 입력을 수신하면 이 플래그가 ON됩니다. OFF: 인터럽트 피딩 실행 중이 아님. ON: 인터럽트 피딩 실행 중.	클리어	클리어	· 인터럽트 피딩이 완료된 뒤 공통 처리 중에 클리어됩니다. · IFEED(892) 명령어로 펄스 출력을 개시한 뒤 인터럽트 입력이 ON되면 ON됩니다.
	A281.09	펄스 출력 1 인터럽트 피딩 에러 플래그	IFEED(892) 명령어로 펄스 출력 1의 출력을 개시한 뒤, 인터럽트 입력을 수신했을 때 또는 지정한 펄스 수만큼 이동했을 때 오버플로나 언더플로가 발생하면 이 플래그가 ON됩니다. ON: 에러 없음. OFF: 오버플로/언더플로 또는 지정한 펄스 수만큼 이동했음.	클리어	클리어	· IFEED(892) 명령어 처리가 개시될 때 클리어됩니다. · 원점이 확정된 상태에서 IFEED(892) 명령어로 운전을 개시한 뒤, 인터럽트 입력을 수신했을 때 오버플로나 언더플로가 발생하거나, 지정한 펄스 수만큼 이동하는 동안 오버플로나 언더플로가 발생하면 ON됩니다.
A283	A283.00	PWM 출력 0 출력 실행 중 플래그	PWM 출력 0에서 펄스를 출력 중일 때 ON됩니다. OFF: 정지, ON: 출력 중	클리어	클리어	펄스 출력이 개시 또는 정지할 때 리프레시됩니다.
	A283.08	PWM 출력 1 출력 실행 중 플래그	PWM 출력 1에서 펄스를 출력 중일 때 ON됩니다. OFF: 정지, ON: 출력 중	클리어	클리어	펄스 출력이 개시 또는 정지할 때 리프레시됩니다.

AI 번역본
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A293	---	버전 에러 정보	전송된 사용자 프로그램에 CPU 유닛의 유닛 버전이 지원하지 않는 기능이 포함되어 있으면 여기에 값이 설정됩니다. 0000 hex: 에러 없음. 0001 hex: 에러	클리어	클리어	운전 개시 시 기록됩니다
A294	---	프로그램 정지시의 태스크 번호	이 워드에는 프로그램 에러로 프로그램 실행이 정지되었을 때 실행 중이던 태스크의 태스크 번호가 저장됩니다. 일반 태스크: 0000~007F hex (태스크 0~127) 인터럽트 태스크: 8000~80FF hex (태스크 0~255) A298과 A299에는 프로그램 실행이 정지된 프로그램 어드레스가 저장됩니다.	클리어	클리어	A298/A299
A295	A295.08	명령어 처리 에러 플래그	명령어 처리 에러가 발생하고 PLC 설정이 명령어 에러 시 운전을 정지하도록 설정되어 있으면, 이 플래그와 에러 플래그(ER)가 ON됩니다. 이 플래그가 ON되면 CPU 유닛의 운전이 정지하고 ERR/ALM 표시등이 점등됩니다. ON: 에러 플래그 ON OFF: 에러 플래그 OFF	클리어	클리어	PLC 설정 (명령어 에러가 발생했을 때의 동작) 에러가 발생한 태스크 번호는 A294에, 프로그램 어드레스는 A298과 A299에 저장됩니다.
	A295.09	DM/EM 간접 지정 BCD 에러 플래그	DM/EM 간접 지정 BCD 에러가 발생하고 PLC 설정이 DM/EM 간접 지정 BCD 에러 시 운전을 정지하도록 설정되어 있으면, 이 플래그와 액세스 에러 플래그(AER)가 ON됩니다. (이 에러는 BCD 모드가 선택되어 있는데도 간접 지정된 DM 또는 EM 워드의 내용이 BCD가 아닐 때 발생합니다.) 이 플래그가 ON되면 CPU 유닛의 운전이 정지하고 ERR/ALM 표시등이 점등됩니다. ON: BCD가 아님 OFF: 정상			
	A295.10	부정 액세스 에러 플래그	불법 액세스 에러가 발생하고 PLC 설정이 불법 액세스 에러 시 운전을 정지하도록 설정되어 있으면, 이 플래그와 액세스 에러 플래그(AER)가 ON됩니다. (이 에러는 메모리 영역을 불법적으로 액세스할 때 발생합니다.) 이 플래그가 ON되면 CPU 유닛의 운전이 정지하고 ERR/ALM 표시등이 점등됩니다. 다음 조작은 불법 액세스로 간주됩니다: I/O 메모리 영역 밖의 영역 읽기/쓰기 시스템이 예약한 영역 읽기/쓰기 EM 파일 메모리 읽기/쓰기 쓰기 보호된 영역에 쓰기 DM/EM 간접 지정 BCD 에러(BCD 모드에서) ON: 부정 액세스 발생 OFF: 정상			

AI 번역본
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A295	A295.11	END 없음 에러 플래그	태스크 내 각 프로그램에 END(001) 명령어가 없을 때 ON됩니다. 이 플래그가 ON되면 CPU 유닛의 운전이 정지하고 ERR/ALM 표시등이 점등됩니다. ON: END 없음 OFF: 정상	클리어	클리어	에러가 발생한 태스크 번호는 A294에, 프로그램 어드레스는 A298과 A299에 저장됩니다.
	A295.12	태스크 에러 플래그	태스크 에러가 발생하면 ON됩니다. 다음 조건에서 태스크 에러가 발생합니다. 실행 가능한(기동된) 일반 태스크가 하나도 없습니다. 태스크에 할당된 프로그램이 없습니다. ON: 에러 OFF: 정상			
	A295.13	미분 오버플로 에러 플래그	미분 명령어에 대응하는 미분 플래그의 허용값을 초과했습니다. 이 플래그가 ON되면 CPU 유닛의 운전이 정지하고 ERR/ALM 표시등이 점등됩니다. ON: 에러 OFF: 정상			
	A295.14	부정 명령어 에러 플래그	실행할 수 없는 프로그램이 저장되어 있을 때 ON됩니다. 이 플래그가 ON되면 CPU 유닛의 운전이 정지하고 ERR/ALM 표시등이 점등됩니다. ON: 에러 OFF: 정상			
	A295.15	UM 오버플로 에러 플래그	UM(사용자 메모리)의 최종 어드레스를 초과했을 때 ON됩니다. 이 플래그가 ON되면 CPU 유닛의 운전이 정지하고 ERR/ALM 표시등이 점등됩니다. ON: 에러 OFF: 정상			
A298 및 A299	---	프로그램이 정지한 프로그램 어드레스	이 워드들에는 프로그램 에러로 프로그램 실행이 정지된 명령어의 8자리 바이너리 프로그램 어드레스가 저장됩니다.	클리어	클리어	(A294에는 프로그램 실행이 정지된 태스크의 태스크 번호가 저장됩니다.)
A300	---	에러 로그 포인터	에러가 발생하면 에러 로그 포인터가 1씩 증분되어, 다음 에러 레코드가 기록될 위치를 에러 로그 영역(A100~A199)의 선두로부터의 오프셋으로 나타냅니다. 00~14 16진수 A500.14(에러 로그 리셋 비트)를 OFF에서 ON으로 전환하면 에러 로그 포인터를 00으로 클리어할 수 있습니다. 에러 로그 포인터가 14 hex(10진수 20)에 도달한 뒤 다음 에러가 발생하면, 다음 레코드는 A195~A199에 저장됩니다.	유지	유지	에러 발생 시 리프레시됩니다. A500.14
A301	---	현재 EM 뱅크---	이 워드에는 현재 EM 뱅크 번호가 4자리 16진수로 저장됩니다. 현재 뱅크 번호는 EMBC(281) 명령어로 변경할 수 있습니다. 0000~0018 16진수	클리어	클리어	---
A302	A302.00 ~ A302.15	CPU 버스 유닛 초기화 중 플래그	해당 CPU 버스 유닛의 CPU 버스 유닛 재기동 비트(A501.00~A501.15)가 OFF에서 ON으로 전환되거나 전원이 ON된 뒤, 그 유닛이 초기화 중인 동안 이 플래그들이 ON됩니다. 비트 00~15는 유닛 번호 0~15에 대응합니다. 유닛이 초기화 중인 동안 CPU 버스 유닛의 리프레시 데이터가 사용되지 않도록 프로그램에서 이 플래그들을 사용하십시오. CPU 버스 유닛이 초기화 중일 때는 IORF(097)와 FIORF(225)를 실행할 수 없습니다. OFF: 초기화 중이 아님 ON: 초기화 중 (초기화 후 자동으로 0으로 리셋됩니다.) 초기화가 완료되면 이 비트들은 자동으로 OFF됩니다.	유지	클리어	초기화 중에 기록됩니다 A501.00

-Japanese Version Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A315	A315.04	시스템 워크 에러(비치명 에러)	시스템 워크(온라인 에디트용 메모리 등)에서 에러가 발생하면 ON됩니다.	유지	클리어	---
	A315.12	태그 메모리 에러 플래그(비치명 에러)	네트워크 심볼이 저장되는 태그 메모리에서 에러가 발생하면 ON됩니다.	유지	클리어	---
	A315.13	옵션 보드 에러 플래그	전원이 공급되는 상태에서 옵션 보드를 분리하거나 지원하지 않는 옵션 보드를 장착하면 ON됩니다. OFF: 에러 없음, ON: 에러	클리어	클리어	· 전원이 ON될 때 기록 됩니다. · 공통 처리 중에 매 사이클 리프레시됩니다.
	A315.15	백업 메모리 에러 플래그	내부 플래시 메모리의 백업 데이터 영역, 소스 메모리 영역 또는 코멘트 메모리 영역에 쓰기가 실패하면 ON됩니다. 쓰기가 정상적으로 완료되면 이 비트는 OFF됩니다.	유지	클리어	---
A316 및 A317	---	고속 카운터 2 현재값	고속 카운터 2의 현재값이 저장됩니다. 하위 4자리: A316, 상위 4자리: A317	클리어	클리어	---
A318 및 A319	---	고속 카운터 3 현재값	고속 카운터 3의 현재값이 저장됩니다. 하위 4자리: A318, 상위 4자리: A319	클리어	클리어	---

AI 번역본
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A320	A320.00	고속 카운터 2 범위 비교 조건 1 범위 내 플래그	이 플래그들은 고속 카운터 2를 상한 · 하한이 있는 범위 비교 모드로 동작시킬 때, 현재값이 8개 범위 중 어느 하나에 들어 있는지를 나타냅니다. 단 범위 내 플래그는, 범위에 들어갈 때 또는 나갈 때 인터럽트 태스크를 실행하도록 고속 카운터가 설정되어 있는지와 무관하게, 비교값이 범위 안에 있으면 항상 ON됩니다. OFF: 범위 외, ON: 범위 내	클리어	클리어	· 공통 처리 중에 매 사이클 리프레시됩니다. · PRV(881) 명령어를 실행하여 범위 비교 결과를 읽을 때 리프레시됩니다. · INI(880) 명령어를 실행하여 현재값 또는 링 카운터 최대값을 변경할 때 리프레시됩니다. · 리셋될 때 리프레시됩니다.
	A320.01	고속 카운터 2 범위 비교 조건 2 범위 내 플래그				
	A320.02	고속 카운터 2 범위 비교 조건 3 범위 내 플래그				
	A320.03	고속 카운터 2 범위 비교 조건 4 범위 내 플래그				
	A320.04	고속 카운터 2 범위 비교 조건 5 범위 내 플래그				
	A320.05	고속 카운터 2 범위 비교 조건 6 범위 내 플래그				
	A320.06	고속 카운터 2 범위 비교 조건 7 범위 내 플래그				
	A320.07	고속 카운터 2 범위 비교 조건 8 범위 내 플래그				
	A320.08	고속 카운터 2 비교 실행 중 플래그	이 플래그는 고속 카운터 2에 대해 비교 동작이 실행 중인지 나타냅니다. OFF: 정지, ON: 비교 중	클리어	클리어	비교 동작이 개시 또는 정지할 때 리프레시됩니다.
	A320.09	고속 카운터 2 오버플로/언더플로 플래그	이 플래그는 고속 카운터 2의 현재값에 오버플로 또는 언더플로가 발생했음을 나타냅니다. (리니어 모드 카운트 범위에서만 사용합니다.) OFF: 정상, ON: 에러	클리어	클리어	· 현재값이 변경될 때 클리어됩니다. · 오버플로 또는 언더플로가 발생할 때 리프레시됩니다.
	A320.10	고속 카운터 2 카운트 방향	이 플래그는 고속 카운터 2가 현재 증분 중인지 감분 중인지 나타냅니다. 방향은 현재 사이클의 카운터 현재값과 직전 사이클의 현재값을 비교하여 판단합니다. OFF: 감분, ON: 증분	---	---	· 고속 카운터에 사용되는 설정이며, 카운터 동작 중에 유효합니다. · 공통 처리 중에 매 사이클 리프레시됩니다. · PRV(881) 명령어를 실행하여 현재값 또는 스테이터스를 읽을 때 리프레시됩니다.

AI 번역본
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A321	A321.00	고속 카운터 3 범위 비교 조건 1 범위 내 플래그	이 플래그들은 고속 카운터 3을 상한 · 하한이 있는 범위 비교 모드로 동작시킬 때, 현재값이 8개 범위 중 어느 하나에 들어 있는지를 나타냅니다. 단 범위 내 플래그는, 범위에 들어갈 때 또는 나갈 때 인터럽트 태스크를 실행하도록 고속 카운터가 설정되어 있는지와 무관하게, 비교값이 범위 안에 있으면 항상 ON됩니다. OFF: 범위 외, ON: 범위 내	클리어	클리어	· 공통 처리 중에 매 사이클 리프레시됩니다. · PRV(881) 명령어를 실행하여 범위 비교 결과를 읽을 때 리프레시됩니다. · INI(880) 명령어를 실행하여 현재값 또는 링 카운터 최대값을 변경할 때 리프레시됩니다. · 리셋될 때 리프레시됩니다.
	A321.01	고속 카운터 3 범위 비교 조건 2 범위 내 플래그				
	A321.02	고속 카운터 3 범위 비교 조건 3 범위 내 플래그				
	A321.03	고속 카운터 3 범위 비교 조건 4 범위 내 플래그				
	A321.04	고속 카운터 3 범위 비교 조건 5 범위 내 플래그				
	A321.05	고속 카운터 3 범위 비교 조건 6 범위 내 플래그				
	A321.06	고속 카운터 3 범위 비교 조건 7 범위 내 플래그				
	A321.07	고속 카운터 3 범위 비교 조건 8 범위 내 플래그				
	A321.08	고속 카운터 3 비교 실행 중 플래그	이 플래그는 고속 카운터 3에 대해 비교 동작이 실행 중인지 나타냅니다. OFF: 정지, ON: 비교 중	클리어	클리어	비교 동작이 개시 또는 정지할 때 리프레시됩니다.
	A321.09	고속 카운터 3 오버플로/언더플로 플래그	이 플래그는 고속 카운터 3의 현재값에 오버플로 또는 언더플로가 발생했음을 나타냅니다. (리니어 모드 카운트 범위에서만 사용합니다.) OFF: 정상, ON: 에러	클리어	클리어	· 현재값이 변경될 때 클리어됩니다. · 오버플로 또는 언더플로가 발생할 때 리프레시됩니다.
	A321.10	고속 카운터 3 카운트 방향	이 플래그는 고속 카운터 3이 현재 증분 중인지 감분 중인지 나타냅니다. 방향은 현재 사이클의 카운터 현재값과 직전 사이클의 현재값을 비교하여 판단합니다. OFF: 감분, ON: 증분	---	---	· 고속 카운터에 사용되는 설정이며, 카운터 동작 중에 유효합니다. · 공통 처리 중에 매 사이클 리프레시됩니다. · PRV(881) 명령어를 실행하여 현재값 또는 스테이터스를 읽을 때 리프레시됩니다.

AI 번역본
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A322 및 A323	---	펄스 출력 2 현재값	해당 펄스 출력 포트에서 출력된 펄스 수가 저장됩니다. 해당 펄스 출력 포트에서 출력된 펄스 수가 저장됩니다. 8000 0000~7FFF FFFF hex(−2,147,483,648~2,147,483,647) CW 방향으로 펄스를 출력 중일 때는 펄스 1개마다 현재값이 1씩 증분됩니다. CCW 방향으로 펄스를 출력 중일 때는 펄스 1개마다 현재값이 1씩 감분됩니다. 오버플로 후 현재값: 7FFF FFFF hex 언더플로 후 현재값: 8000 000 hex 하위 4자리: A276/A278/A322/A324 상위 4자리: A277/A279/A323/A325	클리어	클리어	· 공통 처리 중에 매 사이클 리프레시됩니다. · INI(880) 명령어를 실행하여 해당 카운터의 현재값을 변경할 때 리프레시됩니다. · 펄스 출력 리셋 비트가 ON되면 클리어됩니다. · 펄스 출력이 개시될 때(원점이 확정되지 않은 경우) 클리어됩니다. · PRV(881) 명령어를 실행하여 현재값 또는 스테이터스를 읽을 때 리프레시됩니다.
A324 및 A325	---	펄스 출력 3 현재값		클리어	클리어	
A326	A326.00	펄스 출력 2 펄스 출력 스테이터스 플래그	이 플래그는 ORG(889), ACC(888), PLS2(887), IFEED(892) 명령어에 따라 펄스 출력 2에서 펄스를 출력 중이고 출력 주파수가 단계적으로 변화(가속 또는 감속)하고 있을 때 ON됩니다. OFF: 정속, ON: 가속/감속 중	클리어	클리어	공통 처리 중에 매 사이클 리프레시됩니다.
	A326.01	펄스 출력 2 오버플로/언더플로 플래그	이 플래그는 펄스 출력 2의 현재값에 오버플로 또는 언더플로가 발생했음을 나타냅니다. OFF: 정상, ON: 에러	클리어	클리어	· INI(880) 명령어를 실행하여 현재값을 변경할 때 클리어됩니다. · 오버플로 또는 언더플로가 발생할 때 리프레시됩니다.
	A326.02	펄스 출력 2 펄스 수 설정 완료 플래그	PULS(886) 명령어로 펄스 출력 2의 출력 펄스 수가 설정되면 ON됩니다. OFF: 미설정, ON: 설정 완료	클리어	클리어	· PULS(886) 명령어가 실행될 때 리프레시됩니다. · 펄스 출력이 정지될 때 클리어됩니다.
	A326.03	펄스 출력 2 출력 완료 플래그	PULS(886), PLS2(887), IFEED(892) 명령어로 설정한 출력 펄스 수가 펄스 출력 2를 통해 출력되면 ON됩니다. OFF: 출력 미완료, ON: 출력 완료	클리어	클리어	독립 모드에서 펄스 출력이 개시 또는 완료될 때 리프레시됩니다.
	A326.04	펄스 출력 2 펄스 출력 중 플래그	펄스 출력 2에서 펄스를 출력 중일 때 ON됩니다. OFF: 정지, ON: 출력 중	클리어	클리어	펄스 출력이 개시 또는 정지할 때 리프레시됩니다.

AI 번역본
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A326	A326.05	펄스 출력 2 원점 미확정 플래그	펄스 출력 2의 원점이 확정되지 않았을 때 ON되고, 원점이 확정되면 OFF됩니다. OFF: 원점 확정, ON: 원점 미확정	운전이 개시될 때 ON 됩니다.	전원이 ON될 때 ON 됩니다.	공통 처리 중에 매 사이클 리프레시됩니다.
	A326.06	펄스 출력 2 원점 정지 플래그	펄스 출력 2의 현재값이 원점(0)과 일치할 때 ON됩니다. OFF: 원점 정지 아님, ON: 원점 정지	클리어	클리어	공통 처리 중에 매 사이클 리프레시됩니다.
	A326.07	펄스 출력 2 출력 정지 에러 플래그	펄스 출력 2의 원점 서치 기능에서 펄스 출력 중에 에러가 발생하면 ON됩니다. 펄스 출력 2 출력 정지 에러 코드가 A438에 기록됩니다. 0: 에러 없음, ON: 정지 에러	클리어	클리어	· 전원이 ON될 때 클리어됩니다. · 원점 서치가 개시될 때 클리어됩니다. · 원점 서치 중에 치명 펄스 출력 에러가 발생하면 ON됩니다. · 펄스 출력용 리미트 입력 신호가 PLC 설정에서 항상 유효하도록 설정되어 있으면, 이 플래그는 다음 조건에 따라 ON되거나 클리어됩니다: · 리미트 입력이 펄스 출력을 정지시키면 ON됩니다. · CW 또는 CCW 리미트 입력이 없는데 CW · CCW 리미트 입력 신호 플래그 중 하나가 ON이면 클리어됩니다.
	A326.08	펄스 출력 2 인터럽트 피딩 실행 중 플래그	IFEED(892) 명령어로 펄스 출력 2~3의 출력을 개시한 뒤 인터럽트 입력을 수신하면 이 플래그들이 ON됩니다. OFF: 인터럽트 피딩 실행 중이 아님. ON: 인터럽트 피딩 실행 중.	클리어	클리어	· 인터럽트 피딩이 완료된 뒤 공통 처리 중에 클리어됩니다. · IFEED(892) 명령어로 펄스 출력을 개시한 뒤 인터럽트 입력을 수신할 때 리프레시됩니다.
	A326.09	펄스 출력 2 인터럽트 피딩 에러 플래그	IFEED(892) 명령어로 펄스 출력 2의 출력을 개시한 뒤, 인터럽트 입력을 수신했을 때 또는 지정한 펄스 수만큼 이동했을 때 오버플로나 언더플로가 발생하면 이 플래그가 ON됩니다. OFF: 오버플로/언더플로 또는 지정한 펄스 수만큼 이동했음. ON: 에러 없음.	클리어	클리어	· IFEED(892) 명령어 처리가 개시될 때 클리어됩니다. · 원점이 확정된 상태에서 IFEED(892) 명령어로 운전을 개시한 뒤, 인터럽트 입력을 수신했을 때 오버플로나 언더플로가 발생하거나, 지정한 펄스 수만큼 이동하는 동안 오버플로나 언더플로가 발생하면 ON됩니다.

AI 번역본
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A327	A327.00	펄스 출력 3 펄스 출력 스테이터스 플래그	이 플래그는 ORG(889), ACC(888), PLS2(887), IFEED(892) 명령어에 따라 펄스 출력 3에서 펄스를 출력 중이고 출력 주파수가 단계적으로 변화(가속 또는 감속)하고 있을 때 ON됩니다. OFF: 정속, ON: 가속/감속 중	클리어	클리어	공통 처리 중에 매 사이클 리프레시됩니다.
	A327.01	펄스 출력 3 오버플로/언더플로 플래그	이 플래그는 펄스 출력 3의 현재값에 오버플로 또는 언더플로가 발생했음을 나타냅니다. OFF: 정상, ON: 에러	클리어	클리어	· INI(880) 명령어를 실행하여 현재값을 변경할 때 클리어됩니다. · 오버플로 또는 언더플로가 발생할 때 리프레시됩니다.
	A327.02	펄스 출력 3 펄스 수 설정 완료 플래그	PULS(886) 명령어로 펄스 출력 3의 출력 펄스 수가 설정되면 ON됩니다. OFF: 미설정, ON: 설정 완료	클리어	클리어	· PULS(886) 명령어가 실행될 때 리프레시됩니다. · 펄스 출력이 정지될 때 클리어됩니다.
	A327.03	펄스 출력 3 출력 완료 플래그	PULS(886), PLS2(887), IFEED(892) 명령어로 설정한 출력 펄스 수가 펄스 출력 3을 통해 출력되면 ON됩니다. OFF: 출력 미완료, ON: 출력 완료	클리어	클리어	독립 모드에서 펄스 출력이 개시 또는 완료될 때 리프레시됩니다.
	A327.04	펄스 출력 3 펄스 출력 중 플래그	펄스 출력 3에서 펄스를 출력 중일 때 ON됩니다. OFF: 정지, ON: 출력 중	클리어	클리어	펄스 출력이 개시 또는 정지할 때 리프레시됩니다.
	A327.05	펄스 출력 3 원점 미확정 플래그	펄스 출력 3의 원점이 확정되지 않았을 때 ON되고, 원점이 확정되면 OFF됩니다. OFF: 원점 확정, ON: 원점 미확정	운전이 개시될 때 ON 됩니다.	전원이 ON될 때 ON 됩니다.	· 펄스 출력이 리셋될 때 ON됩니다. · 원점 서치가 개시될 때 ON됩니다. · 리미트 입력을 수신하고 클리어가 설정되어 있을 때 ON됩니다. · 오버플로 또는 언더플로가 발생할 때 ON됩니다. · 원점 서치가 완료될 때 OFF 됩니다. · INI(880) 명령어를 실행하여 현재값을 변경할 때 OFF됩니다.
	A327.06	펄스 출력 3 원점 정지 플래그	펄스 출력 3의 현재값이 원점(0)과 일치할 때 ON됩니다. OFF: 원점 정지 아님, ON: 원점 정지	클리어	클리어	· 원점에서 정지할 때 ON됩니다. · 원점을 벗어날 때 OFF 됩니다.

AI 번역본
-For Reference Only-

어드레스		명칭	기능	모드 변경 후의 상태	기동 시의 상태	쓰기 타이밍/ 관련 플래그 · 설정
워드	비트					
A327	A327.07	펄스 출력 3 출력 정지 에러 플래그	펄스 출력 3의 원점 서치 기능에서 펄스 출력 중에 에러가 발생하면 ON됩니다. 펄스 출력 3 출력 정지 에러 코드가 A439에 기록됩니다. 0: 에러 없음, ON: 정지 에러	클리어	클리어	· 전원이 ON될 때 클리어됩니다. · 원점 서치가 개시될 때 클리어됩니다. · 원점 서치 중에 치명 펄스 출력 에러가 발생하면 ON됩니다. · 펄스 출력용 리미트 입력 신호가 PLC 설정에서 항상 유효하도록 설정되어 있으면, 이 플래그는 다음 조건에 따라 ON되거나 클리어됩니다: · 리미트 입력이 펄스 출력을 정지시키면 ON됩니다. · CW 또는 CCW 리미트 입력이 없는데 CW · CCW 리미트 입력 신호 플래그 중 하나가 ON이면 클리어됩니다.
	A327.08	펄스 출력 3 인터럽트 피딩 실행 중 플래그	IFEED(892) 명령어로 펄스 출력 3~3의 출력을 개시한 뒤 인터럽트 입력을 수신하면 이 플래그들이 ON됩니다. OFF: 인터럽트 피딩 실행 중이 아님. ON: 인터럽트 피딩 실행 중.	클리어	클리어	· 인터럽트 피딩이 완료된 뒤 공통 처리 중에 클리어됩니다. · IFEED(892) 명령어로 펄스 출력을 개시한 뒤 인터럽트 입력을 수신할 때 리프레시됩니다.
	A327.09	펄스 출력 3 인터럽트 피딩 에러 플래그	IFEED(892) 명령어로 펄스 출력 3의 출력을 개시한 뒤, 인터럽트 입력을 수신했을 때 또는 지정한 펄스 수만큼 이동했을 때 오버플로나 언더플로가 발생하면 이 플래그가 ON됩니다. ON: 에러 없음. OFF: 오버플로/언더플로 또는 지정한 펄스 수만큼 이동했음.	클리어	클리어	· IFEED(892) 명령어 처리가 개시될 때 클리어됩니다. · 원점이 확정된 상태에서 IFEED(892) 명령어로 운전을 개시한 뒤, 인터럽트 입력을 수신했을 때 오버플로나 언더플로가 발생하거나, 지정한 펄스 수만큼 이동하는 동안 오버플로나 언더플로가 발생하면 ON됩니다.
A329	A329.00	PWM 출력 2 출력 실행 중 플래그	PWM 출력 2에서 펄스를 출력 중일 때 ON됩니다. OFF: 정지, ON: 출력 중	클리어	클리어	펄스 출력이 개시 또는 정지할 때 리프레시됩니다.
	A329.08	PWM 출력 3 출력 실행 중 플래그	PWM 출력 3에서 펄스를 출력 중일 때 ON됩니다. OFF: 정지, ON: 출력 중	클리어	클리어	펄스 출력이 개시 또는 정지할 때 리프레시됩니다.

AI 번역본
-For Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A330 ~ A335	A330.00 ~ A335.15	특수 I/O 유닛 초기화 중 플래그	특수 I/O 유닛 재시작 비트(A502.00~A507.15)가 OFF에서 ON으로 전환되거나 전원이 ON된 후, 해당 특수 I/O 유닛이 초기화 중인 동안 이 플래그가 ON됩니다. 이 워드들의 비트는 다음과 같이 유닛 번호 0~95에 대응합니다: A330.00~A330.15: 유닛 0~15 A331.00~A331.15: 유닛 16~31 ---- A335.00~A335.15: 유닛 80~95 유닛이 초기화 중일 때 특수 I/O 유닛의 리프레시 데이터가 사용되지 않도록 프로그램에서 이 플래그를 사용하십시오. 또한 특수 I/O 유닛이 초기화 중일 때는 IORF(097)과 FIORF(225)를 실행할 수 없습니다. OFF: 초기화 중 아님 ON: 초기화 중 (초기화 후 자동으로 0으로 리셋됩니다.) 초기화가 완료되면 이 비트들은 자동으로 OFF됩니다.	유지	클리어	A502.00~A507.15
A336 ~ A337	A336.00 ~ A337.15	기동 시 검출된 유닛 수(랙 0 ~7)	각 랙에서 검출된 유닛 수가 1자리 16진수(0~A hex)로 저장됩니다. 랙 0: A336.00~A336.03 랙 1: A336.04~A336.07 랙 2: A336.08~A336.11 랙 3: A336.12~A336.15 랙 4: A337.00~A337.03 랙 5: A337.04~A337.07 랙 6: A337.08~A337.11 랙 7: A337.12~A337.15 예: 랙 0에 1유닛, 랙 1에 4유닛, 랙 2에 8유닛, 랙 3에 10유닛이 있으면 다음과 같이 저장됩니다: A336 = A 8 4 1	유지	클리어	---
A339 및 A340	---	최대 미분 플래그 번호	이 워드들에는 미분 명령어가 사용 중인 미분 플래그 번호의 최대값이 저장됩니다.	기능 열 참조.	클리어	운전 개시 시에 기록 A295.13

AI 번역본
-For Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A343	A343.00 ~ A343.02	메모리 카드 종별	장착된 메모리 카드가 있는 경우 그 종별을 나타냅니다. 0: 없음 4: 플래시 ROM 이 정보는 PLC 전원이 ON되거나 메모리 카드 전원 스위치가 ON될 때 기록됩니다.	유지	기능 열 참조.	기능 열 참조.
	A343.04	파일 암호화 유효 플래그	파일 암호화의 상태가 저장됩니다. (CJ3M CPU 유닛 전용) 0: 암호화 무효 1: 암호화 유효	유지	내부 상태가 반영됨	파일 암호화가 설정될 때
	A343.06	EM 파일 메모리 포맷 에러 플래그	파일 메모리에 할당된 선두 EM 뱅크에서 포맷 에러가 발생하면 ON됩니다. ON: 포맷 에러 OFF: 포맷 에러 없음 포맷이 정상적으로 완료되면 이 플래그는 OFF됩니다.	유지	클리어	---
	A343.07	메모리 카드 포맷 에러 플래그	메모리 카드가 포맷되지 않았거나 포맷 에러가 발생하면 ON됩니다. (포맷이 정상적으로 완료되면 이 플래그는 OFF됩니다.) ON: 포맷 에러 OFF: 포맷 에러 없음 이 플래그는 PLC 전원이 ON되거나 메모리 카드 전원 스위치가 ON될 때 기록됩니다.	유지	기능 열 참조.	기능 열 참조.
	A343.08	파일 전송 에러 플래그	파일 메모리에 데이터를 쓰는 중에 에러가 발생하면 ON됩니다. ON: 에러 OFF: 에러 없음	유지	클리어	파일 데이터를 쓸 때 리프레시됩니다.
	A343.09	파일 쓰기 에러 플래그	파일을 쓰는 중에 SD 메모리 카드의 LOCK 스위치가 ON인 경우, 쓰려는 파일이 쓰기 금지(읽기 전용)인 경우, 또는 메모리 카드의 용량을 초과하게 되어 쓸 수 없는 경우에 ON됩니다. ON: 쓰기 불가 OFF: 정상 상태	유지	클리어	파일 데이터를 쓸 때 리프레시됩니다.
	A343.10	파일 읽기 에러	이상(파일 손상 또는 데이터 파손)으로 파일을 읽을 수 없는 경우에 ON됩니다. ON: 읽기 불가 OFF: 정상 상태 또는 읽기 처리 실행 중	유지	클리어	파일 데이터를 읽을 때 리프레시됩니다.
	A343.11	파일 없음 플래그	존재하지 않는 파일을 읽으려고 하거나, 존재하지 않는 디렉터리의 파일에 쓰려고 하면 ON됩니다. ON: 지정한 파일 또는 디렉터리가 없음 OFF: 정상 상태 또는 읽기 처리 실행 중	유지	클리어	파일 데이터를 읽을 때 리프레시됩니다.
	A343.13	파일 메모리 동작 플래그	다음 조작 중 하나가 실행되고 있는 동안 ON됩니다. 어느 것도 실행되고 있지 않으면 OFF됩니다. 자체 CPU 유닛에 FINS 커맨드를 송신하는 CMND 명령어. 파일 메모리 명령어의 실행. 보조 기억 영역의 제어 비트를 사용한 프로그램 교체. 간이 백업 조작. ON: 명령어 실행 중. OFF: 명령어 실행 중 아님.	유지	클리어	파일 메모리 명령어가 실행될 때 리프레시됩니다.
	A343.14	파일 데이터 액세스 중 플래그	파일 데이터에 액세스하는 동안 ON됩니다. ON: 파일 액세스 중 OFF: 파일 액세스 중 아님 두 개의 파일 메모리 명령어가 동시에 실행되지 않도록 이 플래그를 사용하십시오.	유지	클리어	---
	A343.15	메모리 카드 검출 플래그	메모리 카드가 검출되면 ON됩니다. 메모리 카드가 검출되지 않으면 OFF됩니다. ON: 메모리 카드 검출됨 OFF: 메모리 카드 검출 안 됨	유지	클리어	메모리 카드가 삽입될 때 또는 전원이 ON될 때 리프레시됩니다.

AI 번역본
-For Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A344	A344.00 ~ A344.07	EM 파일 메모리 선두 뱅크	EM 트레이스 메모리 또는 EM 파일 메모리의 선두 뱅크 번호가 저장됩니다. 이 선두 뱅크부터 EM 영역의 마지막 뱅크까지의 모든 EM 뱅크가 파일 메모리 또는 트레이스 메모리로 사용되도록 포맷됩니다. EM 영역 뱅크가 파일 메모리나 트레이스 메모리로 포맷되어 있지 않으면 이 A344는 FFFF hex가 됩니다. EM 영역의 일부를 파일 메모리 또는 트레이스 메모리로 변환하려면 **CX-Programmer에서 PLC - Memory Allocation - EM Memory Settings를** 선택한 다음 File Memory 또는 Trace Memory를 선택하십시오.	유지	유지	PLC 설정(EM File Setting Enabled 파라미터 및 EM Start File No. 파라미터)
	A344.14	EM 트레이스 메모리 플래그	A344가 FFFF hex가 아니고 이 플래그가 ON이면, A344.00~A344.07에 지정된 뱅크부터 EM 영역의 끝까지의 EM 영역 뱅크가 트레이스 메모리로 포맷됩니다.	유지	유지	
	A344.15	EM 파일 메모리 플래그	A344가 FFFF hex가 아니고 이 플래그가 ON이면, A344.00~A344.07에 지정된 뱅크부터 EM 영역의 끝까지의 EM 영역 뱅크가 파일 메모리로 포맷됩니다.	유지	유지	
A345	A345.00	FB 프로그램 소스 정보 플래그	소스/코멘트 메모리에 FB 프로그램 소스 정보가 있으면 ON됩니다. ON: 정보 있음, OFF: 정보 없음	유지	내부 상태 갱신됨	CX-Programmer에서 전송될 때
	A345.01	심볼 테이블 정보 플래그	소스/코멘트 메모리에 심볼 테이블 정보가 있으면 ON됩니다. ON: 정보 있음, OFF: 정보 없음	유지	내부 상태 갱신됨	
	A345.02	코멘트 정보 플래그	소스/코멘트 메모리에 코멘트 정보가 있으면 ON됩니다. ON: 정보 있음, OFF: 정보 없음	유지	내부 상태 갱신됨	
	A345.03	프로그램 인덱스 정보 플래그	소스/코멘트 메모리에 프로그램 인덱스 정보가 있으면 ON됩니다. ON: 정보 있음, OFF: 정보 없음	유지	내부 상태 갱신됨	
	A345.05	SFC 프로그램 소스 정보 플래그	소스/코멘트 메모리에 SFC 프로그램 소스 정보가 있으면 ON됩니다. ON: 정보 있음, OFF: 정보 없음	유지	내부 상태 갱신됨	
	A345.07	네트워크 심볼 (태그) 정보 플래그	태그 메모리에 네트워크 심볼(태그) 정보가 있으면 ON됩니다. ON: 정보 있음, OFF: 정보 없음	유지	내부 상태 갱신됨	
A346 및 A347	---	전송 잔여 워드 수	이 워드들에는 FREAD(700) 또는 FWRIT(701)로 전송할 남은 워드 수가 8자리 16진수로 저장됩니다. 이 명령어 중 하나가 실행되면 전송할 워드 수가 A346과 A347에 기록됩니다. 데이터가 전송되는 동안 이 워드들의 값은 감산됩니다. A346에는 하위 4자리, A347에는 상위 4자리가 저장됩니다. 예정한 워드 수가 정상적으로 전송되었는지 여부를 판단하려면 이 워드들의 내용을 확인하십시오.	유지	클리어	FREAD 또는 FWRIT가 실행되는 동안 기록됩니다. 데이터가 실제로 전송됨에 따라 감산됩니다.
A351 ~ A354	---	달력/시계 영역	이 워드들에는 CPU 유닛의 내부 시계 데이터가 BCD로 저장됩니다. 시계는 CX-Programmer, DATE(735) 명령어 또는 FINS 커맨드(CLOCK WRITE, 0702)로 설정할 수 있습니다. 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: 연(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 배터리가 접속되어 있지 않으면 이 값은 불안정해집니다.	유지	유지	매 사이클 기록

-Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A360 ~ A391	A360.01 ~ A391.15	FAL 실행 번호 플래그	FAL(006)이 실행되면 지정한 FAL 번호에 대응하는 플래그가 ON됩니다. 비트 A360.01~A391.15는 FAL 번호 001~511에 대응합니다. ON: 해당 FAL이 실행됨 OFF: 해당 FAL이 실행되지 않음 에러가 해제되면 이 플래그는 OFF됩니다.	유지	클리어	에러 발생 시 리프레시됩니다. A402.15
A392	A392.04	시리얼 포트 에러 플래그	RS-232C 포트에서 에러가 발생하면 ON됩니다. (페리페럴 버스 모드, 1:N NT 링크 모드, 시리얼 PLC 링크 폴링/폴드 유닛 모드에서는 이 비트에 액세스하지 마십시오.) ON: 에러 OFF: 에러 없음	유지	클리어	에러 발생 시 리프레시됩니다.
	A392.05	시리얼 포트 송신 가능 플래그(무순서 모드)	RS-232C 포트가 무순서 모드에서 데이터를 송신할 수 있으면 ON됩니다. ON: 송신 가능 OFF: 송신 불가	유지	클리어	송신 후에 기록
	A392.06	시리얼 포트 수신 완료 플래그 (무순서 모드)	RS-232C 포트가 무순서 모드에서 수신을 완료하면 ON됩니다. ・바이트 수를 지정한 경우: 지정한 바이트 수를 수신하면 ON됩니다. ・엔드 코드를 지정한 경우: 엔드 코드를 수신하거나 256바이트를 수신하면 ON됩니다.	유지	클리어	수신 후에 기록
	A392.07	시리얼 포트 수신 오버플로 플래그 (무순서 모드)	무순서 모드에서 RS-232C 포트를 통한 수신 중에 데이터 오버플로가 발생하면 ON됩니다. ・바이트 수를 지정한 경우: 수신이 완료된 후 RXD(235)가 실행되기 전에 데이터가 더 수신되면 ON됩니다. ・엔드 코드를 지정한 경우: 엔드 코드를 수신한 후 RXD(235)가 실행되기 전에 데이터가 더 수신되면 ON됩니다. 엔드 코드 이전에 257바이트를 수신하면 ON됩니다. ON: 오버플로 OFF: 오버플로 없음	유지	클리어	
A393	A393.00 ~ A393.07	시리얼 포트 PT 통신 플래그	시리얼 포트가 NT 링크 모드 또는 시리얼 PLC 링크 모드에서 통신 중이면 해당 비트가 ON됩니다. 비트 0~7은 유닛 0~7에 대응합니다. ON: 통신 중 OFF: 통신 중 아님	유지	클리어	토큰에 대한 정상 응답이 있을 때 리프레시됩니다.
	A393.08 ~ A393.15	시리얼 포트 PT 우선 등록 플래그	RS-232C 포트가 NT 링크 모드에서 통신 중일 때 우선권을 가진 PT에 대해 해당 비트가 ON됩니다. 비트 0~7은 유닛 0~7에 대응합니다. 이 플래그는 우선 등록 커맨드를 수신했을 때 기록됩니다. ON: 우선 등록됨 OFF: 우선 등록 안 됨	유지	클리어	기능 열 참조.
	A393.00 ~ A393.15	시리얼 포트 수신 카운터(무순서 모드)	RS-232C 포트가 무순서 모드일 때 수신한 데이터의 바이트 수를 2진수로 나타냅니다.	유지	클리어	데이터 수신 시 리프레시됩니다.

AI 번역본
For Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A395	A395.06	파일 삭제 플래그	정전이 발생했을 때 갱신 중이던 메모리 카드 파일의 나머지를 시스템이 삭제했습니다. (CJ3M CPU 유닛은 지원하지 않습니다.) ON: 파일 삭제됨 OFF: 삭제된 파일 없음	클리어	클리어	시스템이 파일을 삭제할 때 리프레시됩니다.
	A395.07		정전이 발생했을 때 갱신 중이던 EM 파일 메모리 파일의 나머지를 시스템이 삭제했습니다. ON: 파일 삭제됨 OFF: 삭제된 파일 없음	클리어	클리어	시스템이 파일을 삭제할 때 리프레시됩니다.
	A395.10	백그라운드 실행용 ER/AER 플래그	백그라운드 처리 중에 명령어 처리 에러 또는 불법 영역 액세스 에러가 발생하면 ON됩니다. ON: 에러. 전원이 ON될 때 OFF(0). 운전이 개시될 때 OFF(0). OFF: 에러 없음. 백그라운드 처리가 개시될 때 OFF(0).	클리어	클리어	---
	A395.11	메모리 파손 검출 플래그	전원이 ON될 때 메모리 파손이 검출되면 ON됩니다. ON: 메모리 파손 OFF: 정상 동작	유지	기능 열 참조.	전원이 ON될 때 리프레시됩니다.
	A395.12	딥 스위치 핀 6 상태 플래그	CPU 유닛 전면의 딥 스위치 핀 6의 상태가 매 사이클 이 플래그에 기록됩니다. ON: 핀 6 ON OFF: 핀 6 OFF	유지	기능 열 참조.	매 사이클 기록됩니다.
A400	---	에러 코드	비치명 에러(사용자 정의 FALS(006) 또는 시스템 에러)나 치명 에러(사용자 정의 FALS(007) 또는 시스템 에러)가 발생하면 4자리 16진수의 에러 코드가 이 워드에 기록됩니다. (A-164쪽의 A-3-3 보조 기억 영역 동작 상세를 참조하십시오). 두 개 이상의 에러가 동시에 발생하면 가장 높은 에러 코드가 기록됩니다.	클리어	클리어	에러 발생 시 리프레시됩니다.

AI 번역본
-For Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A401	A401.03	카드 전송 에러 플래그	· 전원이 ON될 때 메모리 카드의 내용이 CPU 유닛으로 올바르게 읽히지 않으면 ON됩니다. · 전원이 ON될 때 네트워크 심볼이 포함된 프로그램 파일(.OBJ)이 전송되면 ON됩니다.	에러 해제 조작으로 클리어됩니다.	클리어	---
	A401.05	버전 에러 플래그	전송된 사용자 프로그램을 CPU 유닛의 유닛 버전이 지원하지 않을 가능성이 있으면 ON됩니다.	에러 해제 조작으로 클리어됩니다.	클리어	---
	A401.06	FALS 에러 플래그 (치명 에러)	FALS(006) 명령어에 의해 비치명 에러가 발생하면 ON됩니다. CPU 유닛은 계속 운전하고 ERR/ALM 표시등이 점멸합니다. 해당 에러 코드가 A400에 기록됩니다. 에러 코드 C101~C2FF는 FALS 번호 001~511에 대응합니다. ON: FALS(006) 실행됨 OFF: FALS(006) 실행 안 됨 FALS 에러가 해제되면 이 플래그는 OFF됩니다.	클리어	클리어	에러 발생 시 리프레시됩니다. A400
	A401.08	사이클 타임 오버 플래그 (치명 에러)	사이클 타임이 PLC 설정에 설정된 최대 사이클 타임(사이클 타임 감시 시간)을 초과하면 ON됩니다. CPU 유닛의 운전이 정지하고 CPU 유닛 전면의 ERR/ALM 표시등이 점등합니다. OFF: 사이클 타임이 최대값 미만 ON: 사이클 타임이 최대값 초과 에러가 해제되면 이 플래그는 OFF됩니다.	클리어	클리어	사이클 타임이 최대값을 초과할 때 리프레시됩니다. PLC 설정(사이클 타임 감시 시간)
	A401.09	프로그램 에러 플래그 (치명 에러)	프로그램 내용이 올바르지 않으면 ON됩니다. CPU 유닛의 운전이 정지하고 CPU 유닛 전면의 ERR/ALM 표시등이 점등합니다. 에러가 발생한 태스크 번호는 A294에 저장되고 프로그램 어드레스는 A298과 A299에 저장됩니다. 발생한 프로그램 에러의 종류는 A295의 비트 8~15에 저장됩니다. 프로그램 에러에 관한 자세한 내용은 A295의 설명과 프로그래밍 매뉴얼을 참조하십시오. ON: 에러 OFF: 에러 없음 에러가 해제되면 이 플래그는 OFF됩니다.	클리어	클리어	A294, A295, A298 및 A299
	A401.10	I/O 설정 에러 플래그 (치명 에러)	· 등록 I/O 테이블이 실제 I/O 테이블과 일치하지 않으면(즉, 등록된 유닛이 실제로 접속된 유닛과 일치하지 않으면) ON됩니다. · CJ2H-CPU6□-EIP에서는 아래에 표시된 슬롯 이외의 슬롯에 인터럽트 입력 유닛이 장착되면 ON됩니다. CJ2H-CPU6□-EIP: CPU 랙 슬롯 0~3 CJ2H-CPU6□, CJ2M-CPU□□ 또는 CJ3M-CPU□□: CPU 랙 슬롯 0~4 · 내장 EtherNet/IP 포트가 있는 기종(CJ2H-CPU6□-EIP, CJ2M-CPU3□, CJ3M-CPU3□)의 등록 I/O 테이블을 내장 EtherNet/IP 포트가 없는 기종(CJ2H-CPU6□, CJ2M-CPU1□, CJ3M-CPU1□)에 다운로드한 경우, 또는 내장 EtherNet/IP 포트가 없는 기종의 등록 I/O 테이블을 내장 EtherNet/IP 포트가 있는 기종에 다운로드한 경우에 ON됩니다. CPU 유닛의 운전이 정지하고 CPU 유닛 전면의 ERR/ALM 표시등이 점등합니다. ON: 에러 OFF: 에러 없음 에러가 해제되면 이 플래그는 OFF됩니다.	클리어	클리어	A405.08
	A401.11	I/O 점수 초과 플래그 (치명 에러)	기본 I/O 유닛에서 사용 중인 I/O 점수가 PLC에서 허용되는 최대값을 초과하거나, 하나의 랙에 11유닛을 초과하여 접속되면 ON됩니다. CPU 유닛의 운전이 정지하고 CPU 유닛 전면의 ERR/ALM 표시등이 점등합니다. ON: 에러 OFF: 에러 없음 에러가 해제되면 이 플래그는 OFF됩니다.	클리어	클리어	A407

-Print Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A401	A401.13	중복 에러 플래그 (치명 에러)	다음의 경우에 ON됩니다: ・두 개의 CPU 버스 유닛에 동일한 유닛 번호가 할당된 경우. ・두 개의 특수 I/O 유닛에 동일한 유닛 번호가 할당된 경우. ・두 개의 기본 I/O 유닛에 동일한 데이터 영역 워드가 할당된 경우. CPU 유닛의 운전이 정지하고 CPU 유닛 전면의 ERR/ALM 표시등이 점등합니다. 중복된 유닛 번호는 A409~A416에 표시됩니다. ON: 중복 에러 OFF: 중복 없음 에러가 해제되면 이 플래그는 OFF됩니다.	클리어	클리어	A410~A416
	A401.14	I/O 버스 에러 플래그 (치명 에러)	CPU 유닛과 슬롯에 장착된 유닛 사이의 데이터 전송에서 에러가 발생하거나, CPU 랙 또는 증설 랙에 엔드 커버가 접속되어 있지 않으면 ON됩니다. CPU 유닛의 운전이 정지하고 CPU 유닛 전면의 ERR/ALM 표시등이 점등합니다. ON: 에러 OFF: 에러 없음 에러가 해제되면 이 플래그는 OFF됩니다. I/O 버스 에러가 발생한 슬롯 번호(00~09)가 A404.00~A404.07에 2진수로 기록됩니다. 엔드 커버가 접속되어 있지 않으면 0E hex가 저장됩니다. I/O 버스 에러가 발생한 랙 번호(00~07)가 A404.08~A404.15에 2진수로 기록됩니다. CJ2H-CPU6□-EIP, CJ2M-CPU3□ 또는 CJ3M-CPU3□의 내장 네트워크에서 I/O 버스 에러가 발생하면 이 비트들에는 0B hex가 저장됩니다. 엔드 커버가 접속되어 있지 않으면 0E hex가 저장됩니다.	클리어	클리어	A404
	A401.15	메모리 에러 플래그 (치명 에러)	메모리에 에러가 발생했거나, 전원이 ON될 때 메모리 카드에서의 자동 전송에 에러가 있었으면 ON됩니다. CPU 유닛의 운전이 정지하고 CPU 유닛 전면의 ERR/ALM 표시등이 점등합니다. ON: 에러 OFF: 에러 없음 에러가 발생한 위치는 A403.00~A403.08에 표시됩니다. 에러가 해제되면 이 플래그는 OFF됩니다.	클리어	클리어	A403

AI 번역본
-For Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A402	A402.00	기타 비치명 에러 플래그(시스템 워크 에러 등)	・A402.01~A402.15에 할당된 비치명 에러 이외의 비치명 에러가 발생하면 ON됩니다. (예: 온라인 에디팅용 메모리에 에러가 발생한 경우.) ・기타 비치명 에러의 상세는 A315에 저장됩니다.	에러 해제 조작으로 클리어됩니다.	클리어	A315
	A402.02	특수 I/O 유닛 설정 에러 플래그 (비치명 에러)	장착된 특수 I/O 유닛이 I/O 테이블에 등록된 특수 I/O 유닛과 일치하지 않으면 ON됩니다. CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다. ON: 에러 OFF: 에러 없음 에러가 해제되면 이 플래그는 OFF됩니다. 설정 에러가 발생한 유닛의 유닛 번호가 A428~A433에 표시됩니다. 동기 유닛 동작 설정에 등록된 유닛에 대해 다음 중 하나가 발생하면 ON됩니다. ・해당 유닛이 동기 유닛 동작 기능을 지원하지 않는 경우. ・해당 유닛이 PLC에 접속되어 있지 않은 경우. ・해당 유닛이 CPU 랙에 없는 경우(즉, 증설 랙에 있는 경우). CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다. ON: 에러 OFF: 에러 없음 에러가 해제되고 올바른 PLC 설정이 CPU 유닛에 전송되면 이 플래그는 OFF됩니다. 설정 에러가 발생한 유닛의 유닛 번호가 A428~A433에 표시됩니다.	클리어	클리어	A428~A433
	A402.03	CPU 버스 유닛 설정 에러 플래그 (비치명 에러)	장착된 CPU 버스 유닛이 I/O 테이블에 등록된 CPU 버스 유닛과 일치하지 않으면 ON됩니다. CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다. ON: 에러 OFF: 에러 없음 에러가 해제되면 이 플래그는 OFF됩니다. 설정 에러가 발생한 유닛의 유닛 번호가 A427에 기록됩니다	클리어	클리어	A427

AI 번역본
-For Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A402	A402.04	배터리 에러 플래그 (비치명 에러)	CPU 유닛의 배터리가 분리되었거나 전압이 낮고 PLC 설정에서 Detect Battery Error 설정이 설정되어 있으면 ON됩니다. CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다. ON: 에러 OFF: 에러 없음 에러가 해제되면 이 플래그는 OFF됩니다. 이 플래그는 배터리 교체가 필요함을 알리기 위해 외부 경고등이나 기타 표시기를 제어하는 데 사용할 수 있습니다.	클리어	클리어	PLC 설정(Detect Battery Error)
	A402.06	특수 I/O 유닛 에러 플래그 (비치명 에러)	CPU 유닛과 특수 I/O 유닛 사이의 데이터 교환에서 에러(특수 I/O 유닛 자체의 에러 포함)가 발생하면 ON됩니다. CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다. 에러가 발생한 특수 I/O 유닛은 운전을 정지합니다. ON: 에러 OFF: 에러 없음 에러가 해제되면 이 플래그는 OFF됩니다. 데이터 교환 에러가 발생한 유닛의 유닛 번호가 A418~A423에 표시됩니다.	클리어	클리어	A418~A423
	A402.07	CPU 버스 유닛 에러 플래그 (비치명 에러)	CPU 유닛과 CPU 버스 유닛 사이의 데이터 교환에서 에러(CPU 버스 유닛 자체의 에러 포함)가 발생하면 ON됩니다. CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다. 에러가 발생한 CPU 버스 유닛은 운전을 정지합니다. ON: 에러 OFF: 에러 없음 에러가 해제되면 이 플래그는 OFF됩니다. 데이터 교환 에러가 발생한 유닛의 유닛 번호가 A417에 표시됩니다.	클리어	클리어	A417
	A402.10	PLC 설정 에러 플래그 (비치명 에러)	PLC 설정에 설정 에러가 있으면 ON됩니다. CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다. ON: 에러 OFF: 에러 없음 에러가 해제되면 이 플래그는 OFF됩니다. 동기 유닛 동작 설정의 설정 에러인 경우에도 올바른 설정이 동기 유닛에 전송되면 OFF됩니다.	클리어	클리어	---
	A402.12	기본 I/O 유닛 에러 플래그 (비치명 에러)	기본 I/O 유닛에 에러가 발생하면 ON됩니다. CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다. ON: 에러 OFF: 에러 없음 에러가 해제되면 이 플래그는 OFF됩니다. 에러 위치가 A408에 기록됩니다.	클리어	클리어	A408
	A402.13	중복 리프레시 에러 플래그 (비치명 에러)	PLC 설정의 Detect Duplicate Refresh Errors 파라미터가 중복 리프레시 에러를 검출하도록 설정되어 있고, 동일한 특수 I/O 유닛에 대해 다음 중 하나가 발생하면 ON됩니다. ・사이클릭 태스크의 FIORF(225), IORF(097), IORD(222) 또는 IOWR(223)이 인터럽트 태스크의 FIORF(225), IORF(097), IORD(222) 또는 IOWR(223)과 경합하는 경우. ・I/O가 리프레시되고 있을 때 인터럽트 태스크에서 FIORF(225), IORF(097), IORD(222) 또는 IOWR(223)이 실행된 경우. 특수 I/O 유닛에 대해 PLC 설정에서 사이클릭 리프레시가 금지되어 있지 않고, 인터럽트 태스크에서 동일한 특수 I/O 유닛에 대해 FIORF(225), IORF(097), IORD(222) 또는 IOWR(223)이 실행되면 중복 리프레시 에러가 발생합니다.	클리어	클리어	A426, PLC 설정 (PLC 설정의 Detect Duplicate Refresh Errors 파라미터가 "Detect"로 설정되어 있음)

AI 번역본
-For Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A402	A402.15	FAL 에러 플래그 (비치명 에러)	FAL(006)을 실행하여 비치명 에러가 발생하면 ON됩니다. CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다. FALS(006)에서 지정한 FAL 번호에 대응하는 A360~A391의 비트가 ON되고 해당 에러 코드가 A400에 기록됩니다. 에러 코드 4101~42FF는 FAL 번호 001~2FF(0~511)에 대응합니다. ON: FALS(006) 에러 발생 OFF: FALS(006) 실행 안 됨 에러가 해제되면 이 플래그는 OFF됩니다.	클리어	클리어	A360~A391, A400
A403	A403.00 ~ A403.08	메모리 에러 위치	메모리 에러가 발생하면 메모리 에러 플래그(A401.15)가 ON되고, 에러가 발생한 메모리 영역을 나타내기 위해 다음 플래그 중 하나가 ON됩니다. A403.00: 사용자 프로그램 A403.04: PLC 설정 A403.05: 등록 I/O 테이블 A403.07: 라우팅 테이블 A403.08: CPU 버스 유닛 설정 메모리 에러가 발생하면 CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다. ON: 에러 OFF: 에러 없음 에러가 해제되면 해당 플래그는 OFF됩니다.	클리어	클리어	A401.15
A404	A404.00 ~ A404.07	I/O 버스 에러 슬롯 번호	I/O 버스 에러가 발생한 슬롯 번호(00~09)가 8비트 2진수로 저장됩니다. CJ2M-CPU3□, CJ3M-CPU3□의 내장 EtherNet/IP 부에서 I/O 버스 에러가 발생하면 0B hex가 저장됩니다. CPU 랙 또는 증설 랙에 엔드 커버가 접속되어 있지 않으면 F hex가 저장됩니다. CPU 유닛의 운전이 정지하고 CPU 유닛 전면의 ERR/ALM 표시등이 점등합니다. 00~09(슬롯 번호 00~09) I/O 버스 에러 플래그(A401.14)가 ON됩니다. 에러가 해제되면 이 플래그는 OFF됩니다.	클리어	클리어	A401.14
	A404.08 ~ A40.415	I/O 버스 에러 랙 번호	I/O 버스 에러가 발생한 랙 번호(00~07)가 8비트 2진수로 저장됩니다. CJ2M-CPU3□, CJ3M-CPU3□의 내장 EtherNet/IP 부에서 I/O 버스 에러가 발생하면 0B hex가 저장됩니다. CPU 랙 또는 증설 랙에 엔드 커버가 접속되어 있지 않으면 F hex가 저장됩니다. CPU 유닛의 운전이 정지하고 CPU 유닛 전면의 ERR/ALM 표시등이 점등합니다. 00~07(랙 번호 0~07) I/O 버스 에러 플래그(A401.14)가 ON됩니다. 에러가 해제되면 이 플래그는 OFF됩니다.	클리어	클리어	A401.14
A405	A405.08	인터럽트 입력 유닛 위치 에러 플래그	인터럽트 입력 유닛이 다음 범위에 접속되어 있지 않으면 ON됩니다. 이 플래그가 ON되면 I/O 설정 에러(치명 에러)가 발생하고 운전이 정지합니다. ・CJ2H-CPU6□-EIP: CPU 랙 슬롯 0~3(CPU 유닛 오른쪽의 4유닛) ・CJ2H-CPU6□, CJ2M-CPU□□, CJ3M-CPU□□: CPU 랙 슬롯 0~4(CPU 유닛 오른쪽의 5유닛) 유닛이 물리적으로는 위 범위에 장착되어 있어도 I/O 테이블에서 더미 유닛 등록으로 이 범위에 할당되어 있지 않으면 에러가 발생합니다.	클리어	클리어	A401.10

-For Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A407	A407.00 ~ A407.12	I/O 점수 초과 상세 1	다음 값 중 하나가 오버플로하거나 중복 에러가 발생하면 해당 값이 바이너리 데이터로 저장됩니다. CPU 유닛의 운전이 정지합니다. CPU 유닛 전면의 ERR/ALM/ 표시등이 점등합니다. 에러가 발생한 이유에 관한 정보는 I/O 점수 초과 상세 2 (A407.13~A407.15)를 참조하십시오. 1. I/O 테이블에 설정된 I/O 점수의 합계(슬레이브 랙 제외)가 CPU 유닛에서 허용되는 최대값을 초과하면 여기에 I/O 점수가 기록됩니다. 2. 인터럽트 입력 점수가 32를 초과하는 경우의 인터럽트 입력 점수. 3. 증설 랙 수가 최대값을 초과하면 여기에 랙 수가 기록됩니다. 에러가 발생하면 해당 값이 여기(A407.00~A407.12)에 기록됩니다. 에러가 해제되면 이 비트들은 클리어됩니다.	클리어	클리어	A401.11, A407.13~ A407.15
	A407.13 ~ A407.15	I/O 점수 초과 상세 2	이 비트들의 3자리 2진값은 I/O 점수 초과 에러의 원인을 나타내며, 비트 A407.00~A407.12에 기록된 값의 의미를 나타냅니다. 000~101(0~5)의 값은 위의 "I/O 점수 초과, 원인 1"에 설명된 원인 1~6에 대응합니다. 000: I/O 합계 초과 001: 인터럽트 입력 점수 초과 100: 펄스 I/O 모듈 수 초과 101: 랙 수 초과 111: 랙의 유닛 수 초과 에러가 해제되면 이 비트들은 클리어됩니다.	클리어	클리어	---
A408	A408.00 ~ A408.07	기본 I/O 유닛 에러 슬롯 번호	기본 I/O 유닛에 에러가 발생하면 A402.12가 ON되고 에러가 발생한 슬롯 번호가 여기에 2진수로 기록됩니다. CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다. 00~09 16진수 (슬롯 0~9) 에러가 해제되면 이 비트들은 클리어됩니다.	클리어	클리어	A402.12
	A408.08 ~ A408.15	기본 I/O 유닛 에러 랙 번호	기본 I/O 유닛에 에러가 발생하면 A402.12가 ON되고 에러가 발생한 랙 번호가 여기에 2진수로 기록됩니다. CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다. 00~07 16진수 (랙 0~7) 에러가 해제되면 이 비트들은 클리어됩니다.	클리어	클리어	A402.12
A409	A409.00 ~ A409.07	증설 랙 번호 중복 플래그	CX-Programmer에서 증설 랙의 선두 워드 어드레스를 설정했을 때 두 랙의 워드 할당이 겹치거나 랙의 선두 어드레스가 CIO 0901을 초과하면 해당 플래그가 ON됩니다. 비트 00~07은 랙 0~7에 대응합니다. ON: 서로 다른 두 랙에 동일한 워드가 할당됨 또는 랙 선두 어드레스가 CIO 0901을 초과함. OFF: 에러 없음 에러가 해제되면 해당 플래그는 클리어됩니다.	클리어	클리어	---
A410	A410.00 ~ A410.15	CPU 버스 유닛 번호 중복 플래그	CPU 버스 유닛의 유닛 번호가 중복되면 중복 에러 플래그(A401.13)와 A410의 해당 플래그가 ON됩니다. 비트 00~15는 유닛 번호 0~F에 대응합니다. CPU 유닛의 운전이 정지하고 CPU 유닛 전면의 ERR/ALM 표시등이 점등합니다. ON: 중복 검출됨 OFF: 중복 없음	클리어	클리어	A401.13

-For Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A411 ~ A416	A411.00 ~ A416.15	특수 I/O 유닛 번호 중복 플래그	특수 I/O 유닛의 유닛 번호가 중복되면 중복 에러 플래그(A401.13)와 A411~A416의 해당 플래그가 ON됩니다. 비트 00~15는 유닛 번호 0~F에 대응합니다. (비트 A411.00~A416.15는 유닛 번호 000~05F (0~95)에 대응합니다.) CPU 유닛의 운전이 정지하고 CPU 유닛 전면의 ERR/ALM 표시등이 점등합니다. 증설 랙의 선두 워드 설정 때문에 특수 I/O 유닛의 워드가 증설 랙의 기본 I/O 유닛에도 할당된 경우에도 해당 비트가 ON됩니다. ON: 중복 검출됨 OFF: 중복 없음	클리어	클리어	A401.13
A417	A417.00 ~ A417.15	CPU 버스 유닛 에러 유닛 번호 플래그	CPU 유닛과 CPU 버스 유닛 사이의 데이터 교환에서 에러가 발생하면 CPU 버스 유닛 에러 플래그(A402.07)가 ON되고, 에러가 발생한 유닛의 유닛 번호에 대응하는 A417의 비트가 ON됩니다. 비트 00~15는 유닛 번호 0~F에 대응합니다. CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다. ON: 에러 OFF: 에러 없음	클리어	클리어	A402.07
A418 ~ A423	A418.00 ~ A423.15	특수 I/O 유닛 에러 유닛 번호 플래그	CPU 유닛과 특수 I/O 유닛 사이의 데이터 교환에서 에러가 발생하면 특수 I/O 유닛 에러 플래그(A402.06)가 ON됩니다. 각 비트는 유닛 번호에 대응합니다. A418의 비트 00부터 A423의 비트 15까지가 유닛 번호 0~95에 대응합니다. CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다. ON: 에러 OFF: 에러 없음 에러가 해제되면 이 플래그는 OFF됩니다. 유닛의 유닛 번호가 불확실한 경우에는 어떤 플래그도 ON되지 않습니다.	클리어	클리어	A402.06
A426	A426.00 ~ A426.11	중복 리프레시 에러 유닛 번호	A426.13이 ON일 때 이 비트들에는: 중복 리프레시가 수행된 특수 I/O 유닛의 유닛 번호가 저장됩니다. 에러가 해제되면 이 비트들은 클리어됩니다. 유닛 번호: 000~05F(0~95) 에러가 해제되면 이 플래그는 OFF됩니다.	클리어	클리어	A402.13 A426.15
	A426.15	중복 리프레시 에러 원인	A402.13(중복 리프레시 에러 플래그)이 ON일 때 이 플래그는 에러의 원인을 나타냅니다. CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다. ON: 중복 리프레시	클리어	클리어	A402.13, A426.00~A426.11
A427	A427.00 ~ A427.15	CPU 버스 유닛 설정 에러 유닛 번호 플래그	CPU 버스 유닛 설정 에러가 발생하면 A402.03과 이 워드에서 해당 유닛의 유닛 번호에 대응하는 비트가 ON됩니다. 비트 00~15는 유닛 번호 0~F에 대응합니다. CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다. ON: 설정 에러 OFF: 설정 에러 없음	클리어	클리어	전원이 ON되거나 I/O가 인식될 때 리프레시됩니다. A402.03
A428 ~ A433	A428.00 ~ A433.15	특수 I/O 유닛 설정 에러 유닛 번호 플래그	특수 I/O 유닛 설정 에러가 발생하면 A402.02와 이 워드들에서 해당 유닛의 유닛 번호에 대응하는 비트가 ON됩니다. 비트 00~15는 유닛 번호 0~F에 대응합니다. ON: 설정 에러 OFF: 설정 에러 없음 CPU 유닛은 계속 운전하고 CPU 유닛 전면의 ERR/ALM 표시등이 점멸합니다.	클리어	클리어	전원이 ON되거나 I/O가 인식될 때 리프레시됩니다. A402.02

AI 번역본 -Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A438	---	펄스 출력 2 정지 에러 코드	펄스 출력 2에서 펄스 출력 정지 에러가 발생하면 그 에러 코드가 이 워드에 기록됩니다.	유지	클리어	· 전원이 ON될 때 클리어됩니다. · 원점 서치가 개시될 때 클리어됩니다. · 원점 서치 중에 치명 펄스 출력 에러가 발생하면 저장됩니다. · PLC 설정에서 펄스 출력의 리미트 입력 신호가 상시 유효로 설정되어 있으면, 플래그는 다음 조건에 따라 ON되거나 클리어됩니다: · 리미트 입력이 펄스 출력을 정지시키면 저장됩니다. · CW 또는 CCW 리미트 입력이 없는데 CW 및 CCW 리미트 입력 신호 플래그 중 하나가 ON이면 클리어됩니다.
A439	---	펄스 출력 3 정지 에러 코드	펄스 출력 3에서 펄스 출력 정지 에러가 발생하면 그 에러 코드가 이 워드에 기록됩니다.	유지	클리어	· 전원이 ON될 때 클리어됩니다. · 원점 서치가 개시될 때 클리어됩니다. · 원점 서치 중에 치명 펄스 출력 에러가 발생하면 저장됩니다. · PLC 설정에서 펄스 출력의 리미트 입력 신호가 상시 유효로 설정되어 있으면, 플래그는 다음 조건에 따라 ON되거나 클리어됩니다: · 리미트 입력이 펄스 출력을 정지시키면 저장됩니다. · CW 또는 CCW 리미트 입력이 없는데 CW 및 CCW 리미트 입력 신호 플래그 중 하나가 ON이면 클리어됩니다.
A440	---	최대 인터럽트 태스크 처리 시간	최대 인터럽트 태스크 처리 시간이 0.1 ms 단위로 저장됩니다. (이 값은 최대 처리 시간을 갖는 인터럽트 태스크가 실행된 후에 기록되며, PLC 운전이 개시될 때 클리어됩니다.) PLC 설정에서 고속 인터럽트 기능이 유효화되어 있으면 무효가 됩니다.	클리어	클리어	기능 열 참조.

AI 번역본
-For Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A441	---	최대 처리 시간 인터럽트 태스크	최대 처리 시간을 갖는 인터럽트 태스크의 태스크 번호가 저장됩니다. 16진값 8000~80FF는 태스크 번호 00~FF에 대응합니다. 인터럽트가 발생하면 비트 15가 ON됩니다. (이 값은 최대 처리 시간을 갖는 인터럽트 태스크가 실행된 후에 기록되며, PLC 운전이 개시될 때 클리어됩니다.) PLC 설정에서 고속 인터럽트 기능이 유효화되어 있으면 무효가 됩니다.	클리어	클리어	기능 열 참조.
A444	---	펄스 출력 0 정지 에러 코드	펄스 출력 0에서 펄스 출력 정지 에러가 발생하면 그 에러 코드가 이 워드에 기록됩니다.	유지	클리어	・전원이 ON될 때 클리어됩니다. ・원점 서치가 개시될 때 클리어됩니다. ・원점 서치 중에 치명 펄스 출력 에러가 발생하면 저장됩니다. ・PLC 설정에서 펄스 출력의 리미트 입력 신호가 상시 유효로 설정되어 있으면, 플래그는 다음 조건에 따라 ON되거나 클리어됩니다: ・리미트 입력이 펄스 출력을 정지시키면 저장됩니다. ・CW 또는 CCW 리미트 입력이 없는데 CW 및 CCW 리미트 입력 신호 플래그 중 하나가 ON이면 클리어됩니다.
A445	---	펄스 출력 1 정지 에러 코드	펄스 출력 1에서 펄스 출력 정지 에러가 발생하면 그 에러 코드가 이 워드에 기록됩니다.	유지	클리어	・전원이 ON될 때 클리어됩니다. ・원점 서치가 개시될 때 클리어됩니다. ・원점 서치 중에 치명 펄스 출력 에러가 발생하면 저장됩니다. ・PLC 설정에서 펄스 출력의 리미트 입력 신호가 상시 유효로 설정되어 있으면, 플래그는 다음 조건에 따라 ON되거나 클리어됩니다: ・리미트 입력이 펄스 출력을 정지시키면 저장됩니다. ・CW 또는 CCW 리미트 입력이 없는데 CW 및 CCW 리미트 입력 신호 플래그 중 하나가 ON이면 클리어됩니다.

AI 번역본
-For Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A450	---	CIO 영역 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A451	---	워드 영역 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A452	---	유지 영역 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A459	---	인덱스 레지스터 영역 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A460	---	DM 영역 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A461	---	EM 뱅크 0 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A462	---	EM 뱅크 1 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A463	---	EM 뱅크 2 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A464	---	EM 뱅크 3 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A465	---	EM 뱅크 4 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A466	---	EM 뱅크 5 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A467	---	EM 뱅크 6 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A468	---	EM 뱅크 7 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---

사용 금지
-Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A469	---	EM 뱅크 8 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A470	---	EM 뱅크 9 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A471	---	EM 뱅크 A 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A472	---	EM 뱅크 B 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A473	---	EM 뱅크 C 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A474	---	EM 뱅크 D 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A475	---	EM 뱅크 E 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A476	---	EM 뱅크 F 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A477	---	EM 뱅크 10 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A478	---	EM 뱅크 11 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A479	---	EM 뱅크 12 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A480	---	EM 뱅크 13 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A481	---	EM 뱅크 14 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---

AT 한국본 -Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A482	---	EM 뱅크 15 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A483	---	EM 뱅크 16 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A484	---	EM 뱅크 17 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A485	---	EM 뱅크 18 지정	이 워드에는 입출력 변수(VER_IN_OUT) 대신 어드레스를 펑션 블록의 파라미터로 지정할 때 영역을 지정하는 데 사용되는 고정 코드가 저장됩니다. 이 코드는 오므론 FB 라이브러리에서만 사용됩니다. 이 워드의 내용을 변경하지 마십시오.	고정값이 설정됩니다.	고정 값이 설정됩니다.	---
A10001 ~ A10003	---	제조 로트 번호	CPU 유닛의 제조 로트 번호가 BCD 데이터로 저장됩니다. 이 영역을 참조하여 하드웨어를 식별할 수 있습니다. hex A10000.00~A10000.07: 00 hex A10000.08~A10000.15: 00 hex A10001.00~A10000.07: 00 hex A10001.08~A10001.15: 일(01~31) A10002.00~A10002.07: 월(01~12) A10002.08~A10002.15: 연(00~09) A10003.00~A10003.07: 00 hex A10003.08~A10003.15: 00 hex	---	---	---
A10008	A10008.01	시계 정지 플래그	시각 설정 후 시계 발진이 정지하면 ON으로 바뀝니다. 시계를 설정하면 이 플래그는 클리어(OFF)됩니다. (CJ3M CPU 유닛 전용) 0: 시계 정지 없음 1: 시계 정지 있음	유지	---	· 시계가 정지하면 ON됩니다 · 시각을 설정하면 클리어됩니다
A10100	A10100.00	동기 유닛 동작 서비스 중 플래그	동기 유닛 동작이 수행되는 동안 ON됩니다. ON: 동기 유닛 동작 수행 중 OFF: 동기 유닛 동작 수행 중 아님	클리어	클리어	전원이 ON되거나 유닛이 재시작된 후 동기 신호가 두 번째로 ON될 때 리프레시됩니다.
A10101	---	동기 입력 데이터 리프레시 에러 코드	CPU 유닛이 지정 시간 내에 동기 유닛으로부터 동기 입력 데이터를 수신하지 못하면 이 워드에 0001 hex가 저장됩니다. 0001 hex: 에러 0000 hex: 정상	클리어	클리어	동기 데이터가 리프레시될 때.
A10102	---	동기 동작 사이클 타임	이 워드에는 PLC 설정에 설정된 동기 동작 사이클 타임이 저장됩니다.	유지	유지	전원이 ON되거나 유닛이 재시작될 때.
A10120 및 A10121	---	펄스 출력 0 주파수	데이터 트레이스로 펄스 출력 0을 트레이스할 때 펄스 출력 0의 주파수가 저장됩니다. 데이터 트레이스 파라미터가 설정되어 있을 때만 유효합니다.	---	클리어	---
A10122 및 A10123	---	펄스 출력 1 주파수	데이터 트레이스로 펄스 출력 1을 트레이스할 때 펄스 출력 1의 주파수가 저장됩니다. 데이터 트레이스 파라미터가 설정되어 있을 때만 유효합니다.	---	클리어	---
A10124 및 A10125	---	펄스 출력 2 주파수	데이터 트레이스로 펄스 출력 2를 트레이스할 때 펄스 출력 2의 주파수가 저장됩니다. 데이터 트레이스 파라미터가 설정되어 있을 때만 유효합니다.	---	클리어	---
A10126 및 A10127	---	펄스 출력 3 주파수	데이터 트레이스로 펄스 출력 3을 트레이스할 때 펄스 출력 3의 주파수가 저장됩니다. 데이터 트레이스 파라미터가 설정되어 있을 때만 유효합니다.	---	클리어	---

AI 번역본
-For Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A10128 및 A10129	---	고속 카운터 0 범위 비교 조건 1~32 범위 내 플래그	이 플래그는 고속 카운터 0이 상한값과 하한값을 갖는 범위 비교 모드로 동작할 때 현재값이 1~32개 범위 중 어느 하나의 안에 있는지를 나타냅니다. 다만 범위 내 플래그는 범위에 들어가거나 나갈 때 인터럽트 태스크를 실행하도록 고속 카운터가 설정되어 있는지와 관계없이, 비교값이 범위 안에 있으면 언제나 ON됩니다. OFF: 범위 밖, ON: 범위 내 하위 워드의 비트 00~15는 범위 1~16에 대응합니다. 상위 워드의 비트 00~15는 범위 17~32에 대응합니다.	클리어	클리어	· 공통 처리 중에 매 사이클 리프레시됩니다. · 1~32개 범위에 대해 비교가 실행될 때 리프레시됩니다. · 범위 비교 결과를 읽기 위해 PRV(881) 명령어가 실행될 때 리프레시됩니다. · 현재값이나 링 카운터 최대값을 변경하기 위해 INI(880) 명령어가 실행될 때 리프레시됩니다. · 리셋
A10130 및 A10131	---	고속 카운터 1 범위 비교 조건 1~32 범위 내 플래그	이 플래그는 고속 카운터 1이 상한값과 하한값을 갖는 범위 비교 모드로 동작할 때 현재값이 1~32개 범위 중 어느 하나의 안에 있는지를 나타냅니다. 다만 범위 내 플래그는 범위에 들어가거나 나갈 때 인터럽트 태스크를 실행하도록 고속 카운터가 설정되어 있는지와 관계없이, 비교값이 범위 안에 있으면 언제나 ON됩니다. OFF: 범위 밖, ON: 범위 내 하위 워드의 비트 00~15는 범위 1~16에 대응합니다. 상위 워드의 비트 00~15는 범위 17~32에 대응합니다.	클리어	클리어	· 공통 처리 중에 매 사이클 리프레시됩니다. · 1~32개 범위에 대해 비교가 실행될 때 리프레시됩니다. · 범위 비교 결과를 읽기 위해 PRV(881) 명령어가 실행될 때 리프레시됩니다. · 현재값이나 링 카운터 최대값을 변경하기 위해 INI(880) 명령어가 실행될 때 리프레시됩니다. · 리셋
A10132 및 A10133	---	고속 카운터 2 범위 비교 조건 1~32 범위 내 플래그	이 플래그는 고속 카운터 2가 상한값과 하한값을 갖는 범위 비교 모드로 동작할 때 현재값이 1~32개 범위 중 어느 하나의 안에 있는지를 나타냅니다. 다만 범위 내 플래그는 범위에 들어가거나 나갈 때 인터럽트 태스크를 실행하도록 고속 카운터가 설정되어 있는지와 관계없이, 비교값이 범위 안에 있으면 언제나 ON됩니다. OFF: 범위 밖, ON: 범위 내 하위 워드의 비트 00~15는 범위 1~16에 대응합니다. 상위 워드의 비트 00~15는 범위 17~32에 대응합니다.	클리어	클리어	· 공통 처리 중에 매 사이클 리프레시됩니다. · 1~32개 범위에 대해 비교가 실행될 때 리프레시됩니다. · 범위 비교 결과를 읽기 위해 PRV(881) 명령어가 실행될 때 리프레시됩니다. · 현재값이나 링 카운터 최대값을 변경하기 위해 INI(880) 명령어가 실행될 때 리프레시됩니다. · 리셋

AI 번역본
-For Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A10134 및 A10135	---	고속 카운터 3 범위 비교 조건 1~32 범위 내 플래그	이 플래그는 고속 카운터 3이 상한값과 하한값을 갖는 범위 비교 모드로 동작할 때 현재값이 1~32개 범위 중 어느 하나의 안에 있는지를 나타냅니다. 다만 범위 내 플래그는 범위에 들어가거나 나갈 때 인터럽트 태스크를 실행하도록 고속 카운터가 설정되어 있는지와 관계없이, 비교값이 범위 안에 있으면 언제나 ON됩니다. OFF: 범위 밖, ON: 범위 내 하위 워드의 비트 00~15는 범위 1~16에 대응합니다. 상위 워드의 비트 00~15는 범위 17~32에 대응합니다.	클리어	클리어	· 공통 처리 중에 매 사이클 리프레시됩니다. · 1~32개 범위에 대해 비교가 실행될 때 리프레시됩니다. · 범위 비교 결과를 읽기 위해 PRV(881) 명령어가 실행될 때 리프레시됩니다. · 현재값이나 링 카운터 최대값을 변경하기 위해 INI(880) 명령어가 실행될 때 리프레시됩니다. · 리셋
A10136 및 A10137	---	고속 카운터 0 링 카운터 최대값	고속 카운터 0을 링 카운터로 사용할 때의 링 카운터 최대값이 저장됩니다. 리니어 모드를 사용하면 이 값은 0으로 클리어됩니다. 하위 4자리: A10136, 상위 4자리: A10137	클리어	클리어	링 카운터 최대값을 변경하기 위해 INI(880) 명령어가 실행될 때 리프레시됩니다.
A10138 및 A10139	---	고속 카운터 1 링 카운터 최대값	고속 카운터 1을 링 카운터로 사용할 때의 링 카운터 최대값이 저장됩니다. 리니어 모드를 사용하면 이 값은 0으로 클리어됩니다. 하위 4자리: A10138, 상위 4자리: A10139	클리어	클리어	링 카운터 최대값을 변경하기 위해 INI(880) 명령어가 실행될 때 리프레시됩니다.
A10140 및 A10141	---	고속 카운터 2 링 카운터 최대값	고속 카운터 2를 링 카운터로 사용할 때의 링 카운터 최대값이 저장됩니다. 리니어 모드를 사용하면 이 값은 0으로 클리어됩니다. 하위 4자리: A10140, 상위 4자리: A10141	클리어	클리어	링 카운터 최대값을 변경하기 위해 INI(880) 명령어가 실행될 때 리프레시됩니다.
A10142 및 A10143	---	고속 카운터 3 링 카운터 최대값	고속 카운터 3을 링 카운터로 사용할 때의 링 카운터 최대값이 저장됩니다. 리니어 모드를 사용하면 이 값은 0으로 클리어됩니다. 하위 4자리: A10142, 상위 4자리: A10143	클리어	클리어	링 카운터 최대값을 변경하기 위해 INI(880) 명령어가 실행될 때 리프레시됩니다.
A10144 및 A10145	---	인터럽트 입력 0 래치 현재값	인터럽트 입력 0에 입력이 있으면 펄스 출력 0의 현재값 또는 고속 카운터 입력 0의 현재값이 저장됩니다. 인터럽트 입력 태스크가 기동되기 직전의 현재값을 읽어 보존합니다. 하위 4자리: A10144, 상위 4자리: A10145	클리어	클리어	입력 인터럽트 발생 시
A10146 및 A10147	---	인터럽트 입력 1 래치 현재값	인터럽트 입력 1에 입력이 있으면 펄스 출력 1의 현재값 또는 고속 카운터 입력 1의 현재값이 저장됩니다. 인터럽트 입력 태스크가 기동되기 직전의 현재값을 읽어 보존합니다. 하위 4자리: A10146, 상위 4자리: A10147	클리어	클리어	입력 인터럽트 발생 시
A10148 및 A10149	---	인터럽트 입력 2 래치 현재값	인터럽트 입력 2에 입력이 있으면 펄스 출력 2의 현재값 또는 고속 카운터 입력 2의 현재값이 저장됩니다. 인터럽트 입력 태스크가 기동되기 직전의 현재값을 읽어 보존합니다. 하위 4자리: A10148, 상위 4자리: A10149	클리어	클리어	입력 인터럽트 발생 시
A10150 및 A10151	---	인터럽트 입력 3 래치 현재값	인터럽트 입력 3에 입력이 있으면 펄스 출력 3의 현재값 또는 고속 카운터 입력 3의 현재값이 저장됩니다. 인터럽트 입력 태스크가 기동되기 직전의 현재값을 읽어 보존합니다. 하위 4자리: A10150, 상위 4자리: A10151	클리어	클리어	입력 인터럽트 발생 시
A10152 및 A10153	---	인터럽트 입력 4 래치 현재값	인터럽트 입력 4에 입력이 있으면 펄스 출력 4의 현재값 또는 고속 카운터 입력 4의 현재값이 저장됩니다. 인터럽트 입력 태스크가 기동되기 직전의 현재값을 읽어 보존합니다. 하위 4자리: A10152, 상위 4자리: A10153	클리어	클리어	입력 인터럽트 발생 시
A10154 및 A10155	---	인터럽트 입력 5 래치 현재값	인터럽트 입력 5에 입력이 있으면 펄스 출력 5의 현재값 또는 고속 카운터 입력 5의 현재값이 저장됩니다. 인터럽트 입력 태스크가 기동되기 직전의 현재값을 읽어 보존합니다. 하위 4자리: A10154, 상위 4자리: A10155	클리어	클리어	입력 인터럽트 발생 시

-For Reference Only-

어드레스		이름	기능	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트					
A10156 및 A10157	---	인터럽트 입력 6 래치 현재값	인터럽트 입력 6에 입력이 있으면 펄스 출력 6의 현재값 또는 고속 카운터 입력 6의 현재값이 저장됩니다. 인터럽트 입력 태스크가 기동되기 직전의 현재값을 읽어 보존합니다. 하위 4자리: A10156, 상위 4자리: A10157	클리어	클리어	입력 인터럽트 발생 시
A10158 및 A10159	---	인터럽트 입력 7 래치 현재값	인터럽트 입력 7에 입력이 있으면 펄스 출력 7의 현재값 또는 고속 카운터 입력 7의 현재값이 저장됩니다. 인터럽트 입력 태스크가 기동되기 직전의 현재값을 읽어 보존합니다. 하위 4자리: A10158, 상위 4자리: A10159	클리어	클리어	입력 인터럽트 발생 시

주 CJ 시리즈 PLC에서는 다음 플래그가 특수 읽기 전용 영역에 제공되며 표에 기재된 라벨로 지정할 수 있습니다.
이 플래그들은 보조 기억 영역에 포함되지 않습니다. 자세한 내용은 6-22 조건 플래그와 6-23 클록 펄스를 참조하십시오.

에러 플래그	조건 플래그 영역
액세스 에러 플래그	
캐리 플래그	
초과 플래그	
이퀄 플래그	
미만 플래그	
네거티브 플래그	
오버플로 플래그	
언더플로 플래그	
이상 플래그	
부등가 플래그	
이하 플래그	
상시 ON 플래그	
상시 OFF 플래그	
0.02s 클록 펄스	클록 펄스 영역
0.1s 클록 펄스	
0.2s 클록 펄스	
1s 클록 펄스	
1min 클록 펄스	

AI 번역본
-For Reference Only-

A-3-2 읽기/쓰기 영역(사용자 설정)

어드레스		이름	기능	설정	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트						
A500	A500.12	IOM 유지 비트	PROGRAM 모드에서 RUN 또는 MONITOR 모드로, 또는 그 반대로 전환할 때 I/O 메모리의 상태를 보존하려면 이 비트를 ON하십시오. (PLC 설정(IOM Hold Bit Status)에서 IOM 유지 비트 자체의 상태가 보존되면, PLC가 ON되거나 정전되어도 I/O 메모리 영역의 상태가 유지됩니다.)	ON: 유지 OFF: 유지 안 함	유지	클리어 (PLC 설정에서 상태를 유지하도록 설정할 수 있습니다.)	기능 열 참조. PLC 설정 (IOM Hold Bit Status 설정)
	A500.13	강제 상태 유지 비트	PROGRAM 모드에서 MONITOR 모드로 또는 그 반대로 전환할 때, 또는 전원을 ON할 때 강제 세트 또는 강제 리셋된 비트의 상태를 보존하려면 이 비트를 ON하십시오. 이 비트는 반드시 IOM 유지 비트(A500.12)와 함께 사용하십시오. 이 비트와 동시에 IOM 유지 비트(A500.12)를 ON하십시오. 정전 시에도 상태가 유지됩니다. PLC가 ON될 때 이 비트의 상태를 보존하려면 PLC 설정의 Forced Status Hold Bit Startup Hold Setting 파라미터를 설정해야 합니다.	ON: 유지 OFF: 유지 안 함	유지	클리어 (PLC 설정에서 상태를 유지하도록 설정할 수 있습니다.)	기능 열 참조. PLC 설정 (Forced Status Hold Bit Status 설정)
	A500.14	에러 로그 리셋 비트	에러 로그 포인터(A300)를 00으로 리셋하려면 이 비트를 ON하십시오. 에러 로그 영역 자체(A100~A199)의 내용은 클리어되지 않습니다. 이 비트는 에러 로그 포인터가 리셋된 후 자동으로 0으로 리셋됩니다.	OFF → ON: 클리어	유지	클리어	A100~A199, A300
	A500.15	출력 OFF 비트	I/O 메모리의 출력 비트 상태와 관계없이 기본 I/O 유닛의 모든 출력을 OFF하려면 이 비트를 ON하십시오(치명 에러가 발생했을 때와 동일). CPU 유닛 전면의 INH 표시등이 점등합니다. 주: 이 비트의 상태는 전원이 OFF되어도 유지됩니다. CIO 영역에서 특수 I/O 유닛에 할당된 출력 비트의 상태(예: 외부 아날로그 출력)는 각 특수 I/O 유닛의 기능에 따라 달라집니다.	---	유지	유지	---
A501	A501.00 ~ A501.15	CPU 버스 유닛 재시작 비트	해당 유닛 번호의 CPU 버스 유닛을 재시작(초기화)하려면 이 비트를 ON하십시오. 비트 00~15는 유닛 번호 0~F에 대응합니다. 재시작 비트가 ON되면 해당 CPU 버스 유닛 초기화 중 플래그(A302.00~A302.15)가 ON됩니다. 초기화가 완료되면 재시작 비트와 초기화 중 플래그가 모두 자동으로 OFF됩니다.	OFF→ON: 재시작 ON→OFF: 재시작 완료 유닛이 재시작되면 시스템에 의해 자동으로 OFF됩니다.	유지	클리어	A302.00~ A302.15
A502~ A507	A502.00 ~ A507.15	특수 I/O 유닛 재시작 비트	해당 유닛 번호의 특수 I/O 유닛을 재시작(초기화)하려면 이 비트를 ON하십시오. 비트 A502.00~ A507.15는 유닛 번호 0~95에 대응합니다. 재시작 비트가 ON되면 해당 특수 I/O 유닛 초기화 중 플래그(A330.00~A335.15)가 ON됩니다. 초기화가 완료되면 재시작 비트와 초기화 중 플래그가 모두 자동으로 OFF됩니다.	OFF→ON: 재시작 ON→OFF: 재시작 완료 유닛이 재시작되면 시스템에 의해 자동으로 OFF됩니다.	유지	클리어	A330.00~ A335.15
A508	A508.09	미분 모니터 완료 플래그	미분 모니터링 실행 중에 미분 모니터 조건이 성립하면 ON됩니다. 이 플래그는 미분 모니터링이 시작될 때 0으로 클리어됩니다.	ON: 모니터 조건 성립 OFF: 아직 성립하지 않음	유지	클리어	---
	A508.11	트레이스 트리거 모니터 플래그	트레이스 개시 비트(A508.14)에 의해 트리거 조건이 성립하면 ON됩니다. 샘플링 개시 비트(A508.15)에 의해 다음 데이터 트레이스가 개시되면 OFF됩니다.	ON: 트리거 조건 성립 OFF: 아직 성립하지 않음 또는 트레이스 중 아님	유지	클리어	---
	A508.12	트레이스 완료 플래그	트레이스 실행 중에 트레이스 메모리의 한 영역에 대한 샘플링이 완료되면 ON됩니다. 다음에 샘플링 개시 비트(A508.15)가 OFF에서 ON으로 전환되면 OFF됩니다.	ON: 트레이스 완료 OFF: 트레이스 중 아님 또는 트레이스 진행 중	유지	클리어	---

-For Reference Only-

어드레스		이름	기능	설정	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트						
A508	A508.13	트레이스 비지 플래그	샘플링 개시 비트(A508.15)가 OFF에서 ON으로 전환되면 ON됩니다. 트레이스가 완료되면 OFF됩니다.	ON: 트레이스 진행 중 OFF: 트레이스 중 아님 (샘플링 중 아님)	유지	클리어	---
	A508.14	트레이스 개시 비트	트리거 조건을 성립시키려면 이 비트를 OFF에서 ON으로 전환하십시오. 딜레이 값(양수 또는 음수)으로 표시되는 오프셋에 따라 어느 데이터 샘플이 유효한지 결정됩니다.	ON: 트레이스 트리거 조건 성립 OFF: 성립하지 않음	유지	클리어	---
	A508.15	샘플링 개시 비트	CX-Programmer에서 이 비트를 OFF에서 ON으로 전환하여 데이터 트레이스를 개시하면, PLC는 다음 세 가지 방법 중 하나로 트레이스 메모리에 데이터를 저장하기 시작합니다: 1. 일정 간격(10~2,550 ms)으로 데이터가 샘플링됩니다. 2. 프로그램에서 TRSM(045)이 실행될 때 데이터가 샘플링됩니다. 3. 매 사이클의 끝에 데이터가 샘플링됩니다. A50815의 동작은 CX-Programmer에서만 제어할 수 있습니다.	OFF→ON: 데이터 트레이스(샘플링) 개시 프로그래밍 디바이스에서 ON됩니다.	유지	클리어	---
A510 및 A511	---	기동 시간 *1	이 워드들에는 전원이 ON된 시각이 저장됩니다. 그 내용은 전원이 ON될 때마다 갱신됩니다. 데이터는 BCD로 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)	기능 열 참조.	유지	기능 열 참조.	전원이 ON될 때 리프레시됩니다.
A512 및 A513	---	정전 시간 *1	이 워드들에는 전원이 차단된 시각이 저장됩니다. 그 내용은 정전될 때마다 갱신됩니다. 데이터는 BCD로 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) 이 워드들은 기동 시에 클리어되지 않습니다.	기능 열 참조.	유지	유지	정전 시에 기록됩니다
A514	---	정전 횟수	전원이 처음 ON된 이후 정전된 횟수가 저장됩니다. 데이터는 2진수로 저장됩니다. 이 값을 리셋하려면 현재값을 0000 hex로 덮어쓰십시오. 이 워드는 기동 시에 클리어되지 않지만, 메모리 파손 검출 플래그(A395.11)가 ON되면 클리어됩니다.	0000~FFFF 16진수	유지	유지	전원이 ON될 때 리프레시됩니다. A395.11
A515~ A517	---	운전 개시 시간 *1	동작 모드를 RUN 또는 MONITOR 모드로 변경한 결과 운전이 개시된 시각이 여기에 BCD로 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: 연(00~99) 전원을 ON한 후 운전이 개시될 때까지는 이전 개시 시각이 저장되어 있습니다.	기능 열 참조.	유지	유지	기능 열 참조.

검토용 번역본
-For Reference Only-

어드레스		이름	기능	설정	모드 변경 후 샹태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트						
A518~ A520	---	운전 종료 시간 *1	동작 모드를 PROGRAM 모드로 변경한 결과 운전이 정지된 시각이 여기에 BCD로 저장됩니다. A518.00~A518.07: 초(00~59) A518.08~A518.15: 분(01~59) 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: 연(00~99) 운전 중에 에러가 발생하면 그 에러가 발생한 시각이 저장됩니다. 그 후 동작 모드를 PROGRAM 모드로 변경하면 PROGRAM 모드로 들어간 시각이 저장됩니다.	기능 열 참조.	유지	유지	기능 열 참조.
A523	---	총 전원 ON 시간	PLC가 ON되어 있던 총 시간이 10시간 단위로 저장됩니다. 데이터는 2진수로 저장되며 10시간마다 갱신됩니다. 이 값을 리셋하려면 현재값을 0000 hex로 덮어쓰십시오. 이 워드는 기동 시에 클리어되지 않지만, 메모리 파손 검출 플래그(A395.11)가 ON되면 0000으로 클리어됩니다.	0000~FFFF 16진수	유지	유지	---
A526	A526.00	시리얼 포트 재시작 비트	RS-232C 포트를 재시작하려면 이 비트를 ON하십시오. (포트가 페리페럴 버스 모드로 동작 중일 때는 이 비트를 사용하지 마십시오.) 재시작 처리가 완료되면 이 비트는 자동으로 OFF됩니다.	OFF→ON: 재시작	유지	클리어	---
A527	A527.00 ~ A527.07	온라인 에디팅 금지 비트 유효화	온라인 에디팅 금지 비트(A527.09)는 이 바이트에 5A가 저장되어 있을 때만 유효합니다. CX-Programmer에서 온라인 에디팅을 금지하려면 이 바이트를 5A로 설정하고 A527.09를 ON하십시오. (온라인 에디팅이란 PLC가 MONITOR 모드로 운전 중일 때 프로그램을 변경하거나 추가하는 것을 말합니다.)	5 A: A527.09 유효 그 밖의 값: A527.09 무효	유지	클리어	A527.09
	A527.09	온라인 에디팅 금지 비트	온라인 에디팅을 금지하려면 이 비트를 ON하십시오. 이 비트의 설정은 A527.00~A527.07이 5 A로 설정되어 있을 때만 유효합니다.	ON: 금지 OFF: 금지 안 함	유지	클리어	A527.00~ A527.07
A528	A528.00 ~ A528.07	시리얼 포트 에러 플래그	이 플래그는 RS-232C 포트에서 어떤 종류의 에러가 발생했는지를 나타내며, RS-232C 포트가 재시작되면 자동으로 OFF됩니다. (이 플래그는 시리얼 게이트웨이 모드에서 유효합니다. 페리페럴 버스 모드에서는 유효하지 않으며, NT 링크 모드에서는 비트 5만 유효합니다.) 시리얼 PLC 링크 모드에서는 다음 비트만 유효합니다. 폴링 유닛: 비트 5: 타임아웃 에러 시 ON. 폴드 유닛: 비트 3: 프레이밍 에러 시 ON. 비트 4: 오버런 에러 시 ON. 비트 5: 타임아웃 에러 시 ON. 이 비트들은 CX-Programmer에서 클리어할 수 있습니다.	비트 0과 1: 사용하지 않음. 비트 2: 패리티 에러 시 ON. 비트 3: 프레이밍 에러 시 ON. 비트 4: 오버런 에러 시 ON. 비트 5: 타임아웃 에러 시 ON. 비트 6과 7: 사용하지 않음.	유지	클리어	---
A529	---	시스템 에러 시뮬레이션용 FAL/FALS 번호	FAL(006) 또는 FALS(007)로 시스템 에러를 시뮬레이션할 때 사용할 더미 FAL/FALS 번호를 설정합니다. FAL(006) 또는 FALS(007)이 실행되었을 때 A529의 번호가 명령어의 오퍼랜드에 지정된 번호와 같으면, 사용자 정의 에러 대신 명령어의 오퍼랜드에 지정된 시스템 에러가 발생합니다.	0001~01FF hex: FAL/FALS 번호 1~511 0000 또는 0200~ FFFF hex: 시스템 에러 시뮬레이션용 FAL/FALS 번호 없음. (에러가 발생하지 않습니다.)	유지	클리어	---

*1 배터리가 접속되어 있지 않으면 기동 시간(A510~A511 워드), 정전 시간(A512~A513 워드), 운전 개시 시간(A515~A517 워드), 운전 종료 시간(A518~A520 워드)이 불안정해집니다.

AI 번역본
For Reference Only

어드레스		이름	기능	설정	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트						
A530	---	전원 인터럽트 금지 영역 설정	DI(693)와 EI(694) 명령어 사이에서 전원 인터럽트(전원 OFF 인터럽트 태스크 제외)를 금지하려면 A5A5 hex로 설정하십시오.	A5A5 hex: 정전 처리 마스킹 유효 그 밖의 값: 정전 처리 마스킹 유효하지 않음.	클리어	클리어	---
A531	A531.00	고속 카운터 0 리셋 비트	리셋 방법이 Z상 신호 + 소프트웨어 리셋으로 설정되어 있으면, 이 비트가 ON인 동안 Z상 신호를 수신했을 때 해당 고속 카운터의 현재값이 리셋됩니다. 리셋 방법이 소프트웨어 리셋으로 설정되어 있으면, 이 비트 중 하나가 ON된 사이클에서 해당 고속 카운터의 현재값이 리셋됩니다.	OFF→ON: 리셋	유지	클리어	---
	A531.01	고속 카운터 1 리셋 비트					
	A531.02	고속 카운터 2 리셋 비트					
	A531.03	고속 카운터 3 리셋 비트					
	A531.08	고속 카운터 0 게이트 비트	이 비트 중 하나가 ON되면 펄스 입력이 카운트되지 않고 해당 고속 카운터의 현재값도 변경되지 않습니다. 비트를 OFF하면 카운트가 다시 개시되고 고속 카운터의 현재값이 갱신됩니다. 리셋 방법이 Z상 신호 + 소프트웨어 리셋으로 설정되어 있고 해당 고속 카운터 리셋 비트(A531.00~A531.03)가 ON이면, 게이트 비트는 유효하지 않습니다.	OFF→ON: 현재값 유지 ON→OFF: 현재값 갱신	유지	클리어	---
	A531.09	고속 카운터 1 게이트 비트					
	A531.10	고속 카운터 2 게이트 비트					
	A531.11	고속 카운터 3 게이트 비트					
A532	---	인터럽트 카운터 0 카운터 설정값	이 워드는 카운터 모드의 입력 인터럽트에 사용합니다. 인터럽트 태스크가 실행될 때까지의 카운트를 설정합니다. 인터럽트 카운터 0이 여기에 설정된 펄스 수를 카운트하면 인터럽트 태스크 140이 실행됩니다.	0000~FFFF hex	유지	클리어	---
A533	---	인터럽트 카운터 1 카운터 설정값	이 워드는 카운터 모드의 입력 인터럽트에 사용합니다. 인터럽트 태스크가 실행될 때까지의 카운트를 설정합니다. 인터럽트 카운터 1이 여기에 설정된 펄스 수를 카운트하면 인터럽트 태스크 141이 실행됩니다.	0000~FFFF hex	유지	클리어	---
A534	---	인터럽트 카운터 2 카운터 설정값	이 워드는 카운터 모드의 입력 인터럽트에 사용합니다. 인터럽트 태스크가 실행될 때까지의 카운트를 설정합니다. 인터럽트 카운터 2가 여기에 설정된 펄스 수를 카운트하면 인터럽트 태스크 142가 실행됩니다.	0000~FFFF hex	유지	클리어	---
A535	---	인터럽트 카운터 3 카운터 설정값	이 워드는 카운터 모드의 입력 인터럽트에 사용합니다. 인터럽트 태스크가 실행될 때까지의 카운트를 설정합니다. 인터럽트 카운터 3이 여기에 설정된 펄스 수를 카운트하면 인터럽트 태스크 143이 실행됩니다.	0000~FFFF hex	유지	클리어	---
A536	---	인터럽트 카운터 0 카운터 현재 값	카운터 모드의 입력 인터럽트용 인터럽트 카운터의 현재값이 저장됩니다. 인크리먼트 모드에서는 카운트가 카운터 설정값과 일치할 때까지 하나씩 증가하고, 일치하면 0으로 돌아갑니다. 디크리먼트 모드에서는 카운트가 설정값에서 0이 될 때까지 하나씩 감소하고, 0이 되면 설정값으로 돌아갑니다.	0000~FFFF hex	유지	클리어	---
A537	---	인터럽트 카운터 1 카운터 현재 값					
A538	---	인터럽트 카운터 2 카운터 현재 값					
A539	---	인터럽트 카운터 3 카운터 현재 값					

-For Reference Only-

어드레스		이름	기능	설정	모드 변경 후 샹태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트						
A540	A540.00	펄스 출력 0 리셋 비트	이 비트가 ON되면 펄스 출력 0의 현재값(A276과 A277에 저장)이 클리어됩니다.	---	유지	클리어	---
	A540.08	펄스 출력 0 CW 리미트 입력 신호 플래그	이것은 원점 서치에 사용되는 펄스 출력 0의 CW 리미트 입력 신호입니다. 이 신호를 사용하려면 실제 센서로부터의 입력을 래더 프로그램의 입력 조건으로 기술하고 그 결과를 이 플래그에 출력하십시오.				
	A540.09	펄스 출력 0 CCW 리미트 입력 신호 플래그	이것은 원점 서치에 사용되는 펄스 출력 0의 CCW 리미트 입력 신호입니다. 이 신호를 사용하려면 실제 센서로부터의 입력을 래더 프로그램의 입력 조건으로 기술하고 그 결과를 이 플래그에 출력하십시오.				
A541	A541.00	펄스 출력 1 리셋 비트	이 비트가 ON되면 펄스 출력 1의 현재값(A278과 A279에 저장)이 클리어됩니다.	---	유지	클리어	---
	A541.08	펄스 출력 1 CW 리미트 입력 신호 플래그	이것은 원점 서치에 사용되는 펄스 출력 1의 CW 리미트 입력 신호입니다. 이 신호를 사용하려면 실제 센서로부터의 입력을 래더 프로그램의 입력 조건으로 기술하고 그 결과를 이 플래그에 출력하십시오.				
	A541.09	펄스 출력 1 CCW 리미트 입력 신호 플래그	이것은 원점 서치에 사용되는 펄스 출력 1의 CCW 리미트 입력 신호입니다. 이 신호를 사용하려면 실제 센서로부터의 입력을 래더 프로그램의 입력 조건으로 기술하고 그 결과를 이 플래그에 출력하십시오.				
A542	A542.00	펄스 출력 2 리셋 비트	이 비트가 ON되면 펄스 출력 2의 현재값(A322와 A323에 저장)이 클리어됩니다.	---	유지	클리어	---
	A542.08	펄스 출력 2 CW 리미트 입력 신호 플래그	이것은 원점 서치에 사용되는 펄스 출력 2의 CW 리미트 입력 신호입니다. 이 신호를 사용하려면 실제 센서로부터의 입력을 래더 프로그램의 입력 조건으로 기술하고 그 결과를 이 플래그에 출력하십시오.				
	A542.09	펄스 출력 2 CCW 리미트 입력 신호 플래그	이것은 원점 서치에 사용되는 펄스 출력 2의 CCW 리미트 입력 신호입니다. 이 신호를 사용하려면 실제 센서로부터의 입력을 래더 프로그램의 입력 조건으로 기술하고 그 결과를 이 플래그에 출력하십시오.				
A543	A543.00	펄스 출력 3 리셋 비트	이 비트가 ON되면 펄스 출력 3의 현재값(A324와 A325에 저장)이 클리어됩니다.	---	유지	클리어	---
	A543.08	펄스 출력 3 CW 리미트 입력 신호 플래그	이것은 원점 서치에 사용되는 펄스 출력 3의 CW 리미트 입력 신호입니다. 이 신호를 사용하려면 실제 센서로부터의 입력을 래더 프로그램의 입력 조건으로 기술하고 그 결과를 이 플래그에 출력하십시오.				
	A543.09	펄스 출력 3 CCW 리미트 입력 신호 플래그	이것은 원점 서치에 사용되는 펄스 출력 3의 CCW 리미트 입력 신호입니다. 이 신호를 사용하려면 실제 센서로부터의 입력을 래더 프로그램의 입력 조건으로 기술하고 그 결과를 이 플래그에 출력하십시오.				
A544	---	인터럽트 카운터 4 카운터 설정값	이 워드는 카운터 모드의 입력 인터럽트에 사용합니다. 인터럽트 태스크가 실행될 때까지의 카운트를 설정합니다. 인터럽트 카운터 4가 여기에 설정된 펄스 수를 카운트하면 인터럽트 태스크 144가 실행됩니다.	0000~FFFF hex	유지	클리어	---
A545	---	인터럽트 카운터 5 카운터 설정값	이 워드는 카운터 모드의 입력 인터럽트에 사용합니다. 인터럽트 태스크가 실행될 때까지의 카운트를 설정합니다. 인터럽트 카운터 5가 여기에 설정된 펄스 수를 카운트하면 인터럽트 태스크 145가 실행됩니다.	0000~FFFF hex	유지	클리어	---
A546	---	인터럽트 카운터 6 카운터 설정값	이 워드는 카운터 모드의 입력 인터럽트에 사용합니다. 인터럽트 태스크가 실행될 때까지의 카운트를 설정합니다. 인터럽트 카운터 6이 여기에 설정된 펄스 수를 카운트하면 인터럽트 태스크 146이 실행됩니다.	0000~FFFF hex	유지	클리어	---

사 본 여 부
-For Reference Only-

어드레스		이름	기능	설정	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트						
A547	---	인터럽트 카운터 7 카운터 설정값	이 워드는 카운터 모드의 입력 인터럽트에 사용합니다. 인터럽트 태스크가 실행될 때까지의 카운트를 설정합니다. 인터럽트 카운터 7이 여기에 설정된 펄스 수를 카운트하면 인터럽트 태스크 147이 실행됩니다.	0000~FFFF hex	유지	클리어	---
A548	---	인터럽트 카운터 4 카운터 현재 값	카운터 모드의 입력 인터럽트용 인터럽트 카운터의 현재값이 저장됩니다. 인크리먼트 모드에서는 카운트가 카운터 설정값과 일치할 때까지 하나씩 증가하고, 일치하면 0으로 돌아갑니다. 디크리먼트 모드에서는 카운트가 설정값에서 0이 될 때까지 하나씩 감소하고, 0이 되면 설정값으로 돌아갑니다.	0000~FFFF hex	유지	클리어	---
A549	---	인터럽트 카운터 5 카운터 현재 값					
A550	---	인터럽트 카운터 6 카운터 현재 값					
A551	---	인터럽트 카운터 7 카운터 현재 값					
A580	A580.00 ~ A580.03	FB 통신 명령어 리트라이 횟수	PLC 설정에 지정된 FB 통신 명령어 설정의 리트라이 횟수가 자동으로 저장됩니다. 이 보조 기억 영역 비트 · 워드는 사용자가 쓰는 것이 아닙니다. 특히 오므론 FB 라이브러리의 펑션 블록을 사용하여 FINS 메시지나 DeviceNet 익스플리싯 메시지 통신을 실행할 때는, PLC 설정의 FB 통신 명령어 설정에서 재송신 횟수와 응답 감시 시간을 사용자가 설정해야 합니다. PLC 설정의 Settings for OMRON FB Library에서 설정한 값은 관련 보조 기억 영역 워드 A580~A582에 자동으로 저장되어 오므론 FB 라이브러리의 펑션 블록에서 사용됩니다.	0~F hex	PLC 설정에 설정된 대로	클리어	운전 개시 시에 기록
A581	---	FB 통신 명령어 응답 감시 시 간	PLC 설정에 설정된 FB 통신 명령어 응답 감시 시간이 자동으로 저장됩니다. 이 보조 기억 영역 비트 · 워드는 사용자가 쓰는 것이 아닙니다. 특히 오므론 FB 라이브러리의 펑션 블록을 사용하여 FINS 메시지나 DeviceNet 익스플리싯 메시지 통신을 실행할 때는, PLC 설정의 FB 통신 명령어 설정에서 재송신 횟수와 응답 감시 시간을 사용자가 설정해야 합니다. PLC 설정의 Settings for OMRON FB Library에서 설정한 값은 관련 보조 기억 영역 워드 A580~A582에 자동으로 저장되어 오므론 FB 라이브러리의 펑션 블록에서 사용됩니다.	0001~FFFF hex (단위: 0.1 s, 범위: 0.1~6553.5) 0000 hex: 2 s	PLC 설정에 설정된 대로	클리어	운전 개시 시에 기록
A582	---	FB DeviceNet 통신 명령어 응답 감시 시간	PLC 설정에 설정된 FB DeviceNet 통신 명령어 응답 감시 시간이 자동으로 저장됩니다. FAL(006) 또는 FALS(007)이 실행되었을 때 A529의 번호가 명령어의 오퍼랜드에 지정된 번호와 같으면, 사용자 정의 에러 대신 명령어의 오퍼랜드에 지정된 시스템 에러가 발생합니다.	0001~FFFF hex (단위: 0.1 s, 범위: 0.1~6553.5) 0000 hex: 2 s	PLC 설정에 설정된 대로	클리어	운전 개시 시에 기록
A595 및 A596	---	백그라운드 실행용 IR00 출 력	백그라운드에서 처리되는 명령어의 출력처로 인덱스 레지스터가 지정되면 IR00 대신 A595와 A596이 그 출력을 받습니다.	0000 0000~FFFF FFFF hex (A596에 상위 자리가 저장됩니다.)	클리어	클리어	---
A597	---	백그라운드 실행용 DR00 출 력	백그라운드에서 처리되는 명령어의 출력처로 데이터 레지스터가 지정되면 DR00 대신 A597이 그 출력을 받습니다.	0000~FFFF hex	클리어	클리어	---

AI 번역본 For Reference Only

어드레스		이름	기능	설정	모드 변경 후 샹태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트						
A598	A598.00	FPD 티칭 비트	티칭 기능으로 감시 시간을 자동으로 설정하려면 이 비트를 ON하십시오. A598.00이 ON인 동안 FPD(269)는 실행 조건이 ON된 후 진단 출력이 ON될 때까지 걸리는 시간을 계측합니다. 계측된 시간이 감시 시간을 초과하면, 계측된 시간에 1.5를 곱한 값이 새로운 감시 시간으로 저장됩니다. 티칭 기능은 감시 시간 오퍼랜드에 워드 어드레스가 지정되어 있을 때만 사용할 수 있습니다.	ON: 감시 시간 티칭 OFF: 티칭 기능 OFF	클리어	클리어	---
	A598.01	백그라운드 실행용 등가 플래그	백그라운드에서 실행된 SRCH(181) 명령어에서 일치하는 데이터가 발견되면 ON됩니다.	ON: 테이블에서 검색 데이터 발견 OFF: 검색 데이터 발견 안 됨	클리어	클리어	---
A600~ A603	---	매크로 영역 입력 워드	MCRO(099)가 실행되면 지정된 네 개의 소스 워드(선두 입력 파라미터 워드부터)의 내용이 여기에 복사됩니다. 복사된 네 워드의 내용은 그 후 서브루틴으로 전달됩니다.	입력 데이터: 4워드	클리어	클리어	---
A604~ A607	---	매크로 영역 출력 워드	MCRO(099)에 지정된 서브루틴이 실행된 후, 서브루틴의 결과가 A604~A607에서 지정된 저장처 워드로 전송됩니다. (출력 파라미터 워드)	출력 데이터: 4워드	클리어	클리어	---
A619	A619.02	시리얼 포트 설정 변경 중 플래그	RS-232C 포트의 통신 설정이 변경되는 동안 ON됩니다. 이 플래그는 STUP(237)이 실행될 때 ON되고 설정이 변경된 후에 OFF됩니다.	ON: 변경 중 OFF: 변경 중 아님	유지	클리어	---

AI 번역본
-For Reference Only-

어드레스		이름	기능	설정	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트						
A620	A620.01	통신 유닛 0, 포트 1 설정 변경 중 플래그	해당 포트의 설정이 변경되는 동안 해당 플래그가 ON됩니다. 이 플래그는 STUP(237)이 실행될 때 ON되고, 설정이 변경된 후 시리얼 통신 유닛에서 발행된 이벤트에 의해 OFF됩니다. 사용자가 이 플래그를 ON하여 시리얼 포트 설정의 변경을 알릴 수도 있습니다.	ON: 변경 중 OFF: 변경 중 아님	유지	클리어	---
	A620.02	통신 유닛 0, 포트 2 설정 변경 중 플래그					---
	A620.03	통신 유닛 0, 포트 3 설정 변경 중 플래그					---
	A620.04	통신 유닛 0, 포트 4 설정 변경 중 플래그					---
A621~ A635	A621.00 ~ A635.04	통신 유닛 1~15, 포트 1~4 설정 변경 중 플래그					---
A650	A650.00 ~ A650.07	프로그램 교체 완료 플래그	・정상 완료(즉, A650.14가 OFF일 때) 01 hex: 프로그램 파일(.OBJ)이 교체되었습니다. ・에러(즉, A650.14가 ON일 때) 00 hex: 치명 에러가 발생했습니다. 01 hex: 메모리 에러가 발생했습니다. 11 hex: 메모리가 쓰기 금지 상태입니다. 12 hex: 프로그램 교체 패스워드가 틀렸습니다. 21 hex: 메모리 카드가 없습니다. 22 hex: 지정한 파일이 존재하지 않습니다. 23 hex: 지정한 파일의 사이즈가 해당 기종의 용량을 초과합니다(메모리 에러). 31 hex: 다음 조작 중 하나가 실행 중입니다. ・파일 메모리와 관련된 조작이 실행 중입니다. ・사용자 프로그램이 쓰이는 중입니다. ・동작 모드가 변경되는 중입니다.	---	유지	클리어	---
	A650.14	교체 에러 플래그	프로그램을 교체하기 위해 교체 개시 비트(A650.15)가 ON되었으나 에러가 있으면 ON됩니다. 교체 개시 비트를 다시 ON하면 교체 에러 플래그는 OFF됩니다.	ON: 교체 에러 OFF: 교체 에러 없음, 또는 교체 개시 비트(A65015)가 ON.	유지	클리어	---
	A650.15	교체 개시 비트	프로그램 패스워드(A651)가 유효(A5A5 hex)하면 교체 개시 비트가 ON될 때 프로그램 교체가 개시됩니다. 프로그램 교체 중에는 교체 개시 비트를 OFF하지 마십시오. 전원이 ON되거나 프로그램 교체가 완료되면, 교체가 정상적으로 완료되었는지 에러로 끝났는지와 관계없이 교체 개시 비트는 OFF됩니다. CX-Programmer, PT 또는 호스트 컴퓨터에서 교체 개시 비트를 읽어 프로그램 교체가 실행 중인지 확인할 수 있습니다.	ON: 프로그램 교체 중 OFF: 교체 완료, 또는 전원이 ON된 후	유지	클리어	---

사내 번역본
-For Reference Only-

어드레스		이름	기능	설정	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트						
A651	---	프로그램 패스워드	프로그램을 교체하려면 패스워드를 입력하십시오. A5A5 hex: 교체 개시 비트(A650.15)가 유효합니다. 그 밖의 값: 교체 개시 비트(A650.15)가 무효입니다. 전원이 ON되거나 프로그램 교체가 완료되면, 교체가 정상적으로 완료되었는지 에러로 끝났는지와 관계없이 교체 개시 비트는 OFF됩니다.	---	유지	클리어	---
A654~ 657	---	프로그램 파일명	프로그램 교체가 개시되면 프로그램 파일명이 ASCII로 저장됩니다. 파일명은 확장자를 제외하고 최대 8자까지 지정할 수 있습니다. 파일명은 A654~A657의 순서(즉, 하위 워드에서 상위 워드로), 그리고 상위 바이트에서 하위 바이트의 순서로 저장됩니다. 파일명이 8자보다 짧으면 남은 하위 바이트와 남은 상위 워드는 공백(20 hex)으로 채워집니다. 파일명 안에는 널 문자와 공백 문자를 사용할 수 없습니다. 예: 파일명이 ABC.OBJ인 경우 15 0 A654 41 42 A655 43 20 A656 20 20 A657 20 20	---	유지	클리어	---
A720~ A722	---	전원 ON 시계 데이터 1 *2	이 워드들에는 워드 A510~A511에 저장된 기동 시간과 동일한 시각 데이터에 더해 월 · 연 정보가 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: 연(00~99)	기능 열 참조.	유지	유지	전원이 ON될 때 기록됩니다.
A723~ A725	---	전원 ON 시계 데이터 2 *2	이 워드들에는 워드 A510과 A511에 저장된 기동 시간보다 한 번 앞서 전원이 ON된 시각이 저장됩니다. 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: 연(00~99)	기능 열 참조.	유지	유지	전원이 ON될 때 기록됩니다.
A726~ A728	---	전원 ON 시계 데이터 3 *2	이 워드들에는 워드 A510과 A511에 저장된 기동 시간보다 두 번 앞서 전원이 ON된 시각이 저장됩니다. 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~A728.07: 월(01~12) A728.08~A728.15: 연(00~99)	기능 열 참조.	유지	유지	전원이 ON될 때 기록됩니다.

AI 번역본 For Reference Only

어드레스		이름	기능	설정	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트						
A729~ A731	---	전원 ON 시계 데이터 4 *2	이 워드들에는 워드 A510과 A511에 저장된 기동 시간보다 세 번 앞서 전원이 ON된 시각이 저장됩니다. 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: 연(00~99)	기능 열 참조.	유지	유지	전원이 ON될 때 기록됩니다.
A732~ A734	---	전원 ON 시계 데이터 5 *2	이 워드들에는 워드 A510과 A511에 저장된 기동 시간보다 네 번 앞서 전원이 ON된 시각이 저장됩니다. 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: 연(00~99)	기능 열 참조.	유지	유지	전원이 ON될 때 기록됩니다.
A735~ A737	---	전원 ON 시계 데이터 6 *2	이 워드들에는 워드 A510과 A511에 저장된 기동 시간보다 다섯 번 앞서 전원이 ON된 시각이 저장됩니다. 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: 연(00~99)	기능 열 참조.	유지	유지	전원이 ON될 때 기록됩니다.
A738~ A740	---	전원 ON 시계 데이터 7 *2	이 워드들에는 워드 A510과 A511에 저장된 기동 시간보다 여섯 번 앞서 전원이 ON된 시각이 저장됩니다. 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: 연(00~99)	기능 열 참조.	유지	유지	전원이 ON될 때 기록됩니다.
A741~ A743	---	전원 ON 시계 데이터 8 *2	이 워드들에는 워드 A510과 A511에 저장된 기동 시간보다 일곱 번 앞서 전원이 ON된 시각이 저장됩니다. 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: 연(00~99)	기능 열 참조.	유지	유지	전원이 ON될 때 기록됩니다.
A744~ A746	---	전원 ON 시계 데이터 9 *2	이 워드들에는 워드 A510과 A511에 저장된 기동 시간보다 여덟 번 앞서 전원이 ON된 시각이 저장됩니다. 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: 연(00~99)	기능 열 참조.	유지	유지	전원이 ON될 때 기록됩니다.

최신 번역본
-For Reference Only-

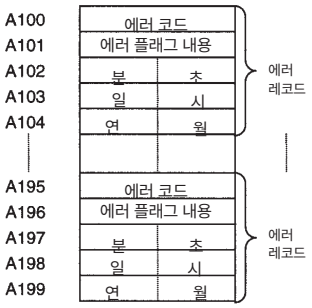
어드레스		이름	기능	설정	모드 변경 후 상태	기동 시 상태	쓰기 타이밍/ 관련 플래그, 설정
워드	비트						
A747~ A749	---	전원 ON 시계 데이터 10 *2	이 워드들에는 워드 A510과 A511에 저장된 기동 시간보다 아홉 번 앞서 전원이 ON된 시각이 저장됩니다. A747.00~A747.07: 초(00~59) A747.08~A747.15: 분(00~59) A748.00~A748.07: 시(00~23) A748.08~A748.15: 일(01~31) A749.00~A749.07: 월(01~12) A749.08~A749.15: 연(00~99)	기능 열 참조.	유지	유지	전원이 ON될 때 기록됩니다.
A937	A937.00	파일 액세스 중 정전 발생 플래그	SD 메모리 카드에 액세스하는 중에 CPU 유닛에서 정전이 발생하면 1(ON)로 바뀝니다. (CJ3M CPU 유닛 전용) 1: SD 메모리 카드 액세스 중에 정전 발생 0: 정상	0→1: 클리어	유지	유지	전원이 ON될 때 기록됩니다.

*2 배터리가 접속되어 있지 않으면 전원 ON 시계 데이터(A720~A749 워드)가 불안정해집니다.

AI 번역본
-For Reference Only-

A-3-3 보조 기억 영역 동작 상세

A100~A199: 에러 로그 영역



1998년 4월 1일 17:10:30에 PLC 설정(04 hex)에 위치한 메모리 에러(에러 코드 80F1)가 발생한 경우, 에러 기록에는 다음 데이터가 생성됩니다.

80	F1
00	04
10	30
01	17
98	04

1997년 5월 2일 8:30:15에 FALS 번호 001의 FALS 에러가 발생한 경우, 에러 기록에는 다음 데이터가 생성됩니다.

C1	01
00	00
30	15
02	08
97	05

에러 코드와 에러 플래그

데이터 종별	에러 코드	에러 이름	에러 플래그
시스템에서 발생하는 치명 에러	0x809F	사이클 타임 오버 에러	---
	80C0~80C7, 80CE, 80CF	I/O 버스 에러	A404
	80E0	I/O 설정 에러	---
	80E1	I/O 점수 초과 에러	A407
	80E9	유닛 번호 중복 에러	A410, A411~A416*1
	80EA	랙 번호 중복 에러	A409
	80F0	프로그램 에러	A295~A299*2
	80F1	메모리 에러	A403
	80F2	버전 에러	---
	80F4	메모리 카드 전송 에러	---
사용자 정의 치명 에러	C101~C2FF	FALS 명령어 실행됨*3	---
시스템에서 발생하는 비치명 에러	008B	중복 리프레시 에러	A426
	009A	기본 I/O 유닛 에러	A408
	009B	PLC 설정 에러	---
	00D1	옵션 보드 에러	A315.13
	00F1	백업 메모리 에러	A315.15
	00F2	태그 메모리 에러*4	A315.12
	00F3	시스템 워크 에러	A315.04
	00F7	배터리 에러	---
	0200~020F	CPU 버스 유닛 에러	A417
	0300~035F	특수 I/O 유닛 에러	A418~A423*5
	0400~040F	CPU 버스 유닛 셋업 에러	A427
	0500~055F	특수 I/O 유닛 셋업 에러	A428~A433*5
사용자 정의 비치명 에러	4101~42FF	FAL 명령어 실행됨*6	---

주 1 번호 중복 에러 시의 에러 플래그 내용은 다음과 같습니다:
비트 0~7: 유닛 번호(2진수), 특수 I/O 유닛은 00~5F hex,
CPU 버스 유닛은 00~0F hex
비트 8~14: 모두 0.
비트 15: 유닛 종별 OFF: CPU 버스 유닛
ON: 특수 I/O 유닛

2 프로그램 에러의 에러 플래그 내용으로는 A295의 내용만 저장됩니다.

3 FALS 번호 001~511에 대해 C101~C2FF가 저장됩니다.

4 CJ2H-CPU 버전 1.5 이하와 CJ2M-CPU 버전 2.1 이하는 지원되지 않습니다.

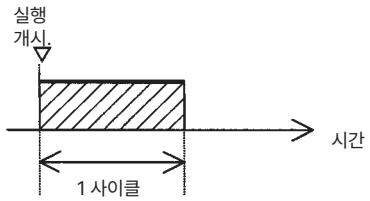
5 에러 플래그 내용으로 0000 hex 값이 저장됩니다.

6 FAL 번호 001~511에 대해 4101~42FF 값이 저장됩니다.

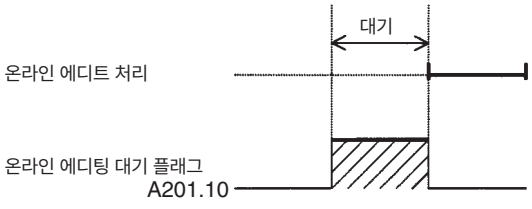
7 배터리가 접속되어 있지 않으면 에러 발생 시각이 불안정해집니다.

AI 번역
-For Reference Only

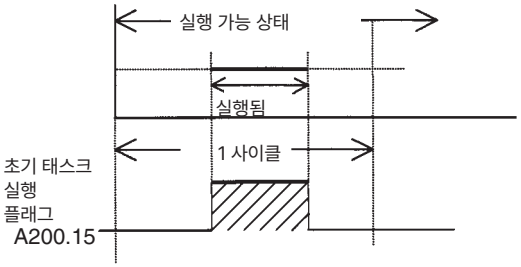
A200.11: 최초 사이클 플래그



A201.10: 온라인 에디팅 대기 플래그

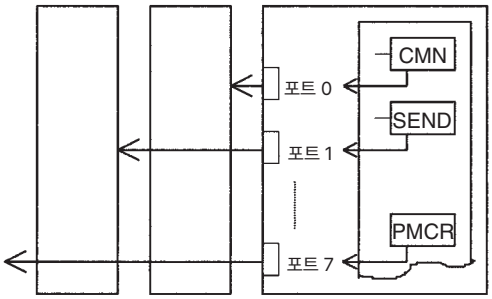


A200.15: 초기 태스크 플래그

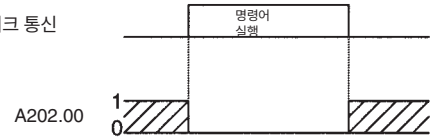


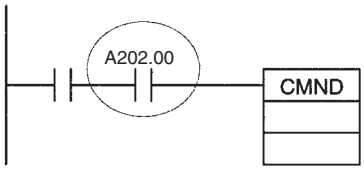
A200.15는 태스크가 실행 가능 상태가 된 후 처음 실행되는 동안 ON됩니다. 태스크가 실행되는 동안에만 ON되며 이후 사이클에서는 ON되지 않습니다.

A202.00~A202.07: 통신 포트 사용 가능 플래그



포트 0에 대해 네트워크 통신 명령어가 실행되었습니다.

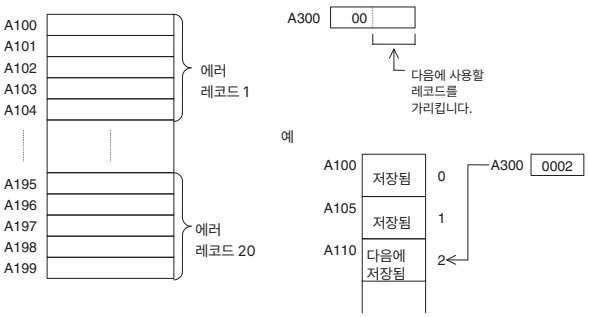




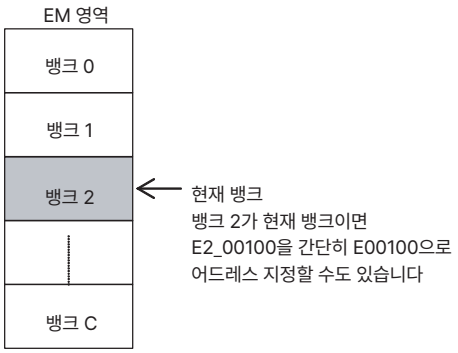
A202.00이 ON일 때만 CMND(490)이 실행되도록 프로그램이 작성되어 있습니다.

AI 번역본
-For Reference Only-

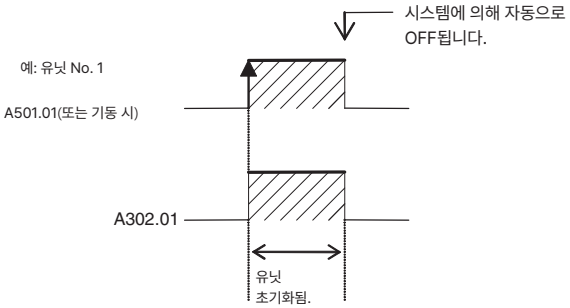
A300: 에러 레코드 포인터



A301: 현재 EM 뱅크



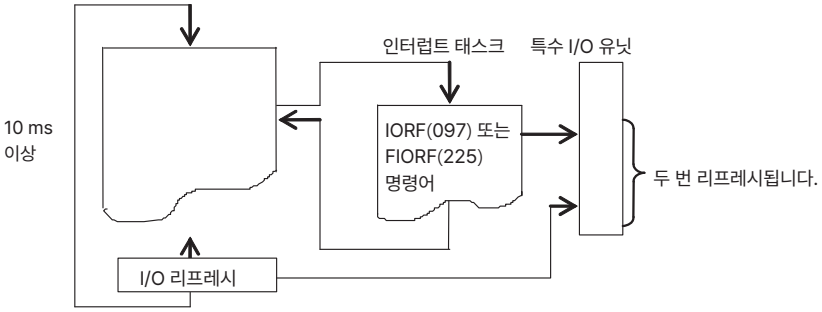
A501.01~A501.15: CPU 버스 유닛 재시작 비트 및



A401.09: 프로그램 에러

	에러	어드레스
A401.09가 ON일 때의 프로그램 에러	UM 오버플로 에러 플래그	A295.15
	부정 명령어 플래그	A295.14
	디스트리뷰션 오버플로 에러 플래그	A295.13
	태스크 에러 플래그	A259.12
	END(001) 없음 에러 플래그	A295.11
	부정 영역 액세스 에러 플래그	A295.10
	간접 DM/EM 어드레스 지정 에러 플래그	A295.09

A426.15: 중복 리프레시 에러 원인



AI 번역본
-For Reference Only-

A-4 PLC 메모리 어드레스의 메모리 맵

A-4-1 PLC 메모리 어드레스

PLC 메모리 어드레스는 I/O 메모리를 간접 지정하기 위해 인덱스 레지스터(IR00~IR15)에 설정합니다.
통상적으로는 레지스터로 전송(MOVR(560))과 타이머/카운터 현재값을 레지스터로 전송 (MOVRW(561)) 명령어를 사용하여 PLC 메모리 어드레스를 인덱스 레지스터에 설정합니다.

데이터 검색(SRCH(181)), 최대값 검색(MAX(182)), 최소값 검색(MIN(183)) 등 일부 명령어는 처리 결과를 인덱스 레지스터에 출력하여 PLC 메모리 어드레스를 나타냅니다.

다른 명령어가 인덱스 레지스터에 저장한 PLC 메모리 어드레스를 사용하기 위해 인덱스 레지스터를 직접 지정할 수 있는 명령어도 있습니다. 이러한 명령어에는 더블 전송 (MOVL(498)), 일부 심볼 비교 명령어(=L,<>L, <L, >L,<=L, >=L), 더블 비교 (CMPL(060)), 더블 데이터 교환(XCGL(562)), BIN 더블 인크리먼트 (++L(591)), BIN 더블 디크리먼트(--L(593)), 캐리 없는 부호 있는 BIN 가산(더블)
(+L(401)), 캐리 없는 부호 있는 BIN 감산(더블)(-L(411)), 레코드 위치 설정
(SETR(635)), 레코드 위치 취득(GETR(636))이 있습니다.

PLC 메모리 어드레스는 모두 연속되어 있으므로 사용자가 메모리 영역의 순서와 경계를 알고 있어야 합니다. 참고로 PLC 메모리 어드레스를 이 부록 끝의 표에 실었습니다.

사용상의 주의

프로그램에서 PLC 메모리 어드레스를 직접 설정하는 것은 가능한 한 피하십시오. 프로그램에 PLC 메모리 어드레스를 설정하면, 새로운 CPU 유닛 기종이나 메모리 배치가 변경된 CPU 유닛과의 호환성이 떨어집니다.

사용상의 주의

어드레스 오프셋과 배열 변수 주의사항

다음 기능을 사용할 때는 오프셋 값에 따라 다음 데이터 영역의 PLC 메모리 어드레스가 지정될 수 있습니다.

- 오프셋을 사용하여 어드레스를 간접 지정하는 경우.
- 배열 변수의 요소 번호를 간접 지정하는 경우.

데이터 영역을 벗어나지 않도록 프로그래밍하거나, 프로그래밍할 때 이 메모리 맵의 데이터 영역 배치 순서를 고려하십시오.

AI 번역본
For Reference Only

사용상의 주의

인덱스 레지스터를 사용한 간접 레지스터 어드레스 지정 시의 주의사항

I/O 메모리 밖의 영역이나 시스템이 예약한 영역에 액세스하기 위해 간접 레지스터 어드레스 지정을 사용하지 마십시오.

액세스를 시도하면 부정 액세스 에러가 발생합니다.

A-4-2 메모리 맵

다음 표는 I/O 메모리 영역의 PLC 메모리 어드레스를 나타냅니다. I/O 메모리 영역은 배터리로 백업됩니다. 배터리가 없으면 I/O 메모리 영역의 내용이 파손될 수 있습니다.

주 시스템 예약이라고 표시된 영역에는 액세스하지 마십시오.

분류	PLC 메모리 어드레스(hex)	사용자 어드레스	영역
I/O 메모리 영역	00000~0B7FF		시스템 예약(액세스가 제한됩니다.)
	0B800~0B801	TK0000~TK0031	태스크 플래그 영역
	0B802~0B83F		시스템 예약(액세스가 제한됩니다.)
	0B840~0B9FF	A000~A447	읽기 전용 보조 기억 영역
	0BA00~0BDFF	A448~A1471	읽기/쓰기 보조 기억 영역
		A10000~A11535	읽기 전용 보조 기억 영역
	0BE00~0BEFF	T0000~T4095	타이머 완료 플래그
	0BF00~0BFFF	C0000~T4095	카운터 완료 플래그
	0C000~0D7FF	CIO 0000~CIO 6143	CIO 영역
	0D800~0D9FF	H000~H511	유지 영역
	0DA00~0DDFF	H512~H1535	유지 영역 이 워드들은 펑션 블록에만 사용됩니다.
	0DE00~0DFFF	W000~W511	워크 영역
	0E000~0EFFF	T0000~T4095	타이머 현재값
	0F000~0FFFF	C0000~C4095	카운터 현재값
	10000~17FFF	D00000~D32767	DM 영역
	18000~1FFFF	E0_00000~E0_32767	EM 영역 뱅크 0
	20000~27FFF	E1_00000~E1_32767	EM 영역 뱅크 1 hex
	:	:	:
	D8000~DFFFF	E18_00000~E18_32767	EM 영역 뱅크 18 hex
	:	:	
	F8000~FFFFF	E_00000~E_32767	EM 영역, 현재 뱅크(주 참조.)

주 프로그램에서 현재 지정된 EM 영역 뱅크의 내용이 이 어드레스에 저장됩니다. 예를 들어
뱅크 1이 지정되어 있으면 20000~27FFF와 동일한 내용이 F8000~FFFFF에 저장됩니다.

-For Reference Only-

A-5 정전 시의 동작

A-5-1 전원 OFF 시의 동작

CPU 유닛의 전원이 OFF되면 다음 처리가 수행됩니다. CPU 유닛이 RUN 또는 MONITOR 모드일 때 전원 공급이 최저 정격 전압의 85%(CJ1W-PD025 DC 전원 유닛은 80%, CJ1W-PD022 DC 전원 유닛은 90%) 미만으로 떨어지면 전원 OFF 처리가 수행됩니다.

1 CPU 유닛이 정지합니다.

2 출력 유닛의 모든 출력이 OFF됩니다.

주 1 IOM 유지 비트의 상태나 PLC 설정의 기동 시 IOM 유지 비트 설정과 관계없이 모든 출력이 OFF됩니다.

2 AC 전원 유닛의 정격 전압 85%:
100 VAC:85 VAC
200 VAC:170 VAC
100~240-V 입력(와이드 레인지): 85 VAC

3 DC 전원 유닛의 정격 전압 80%/90%:
CJ1W-PD025: 19.2 VDC
CJ1W-PD022: 21.6 VDC

순시 정전

전원이 순간적으로만 떨어진 경우(순시 정전)에는 다음 처리가 수행됩니다.

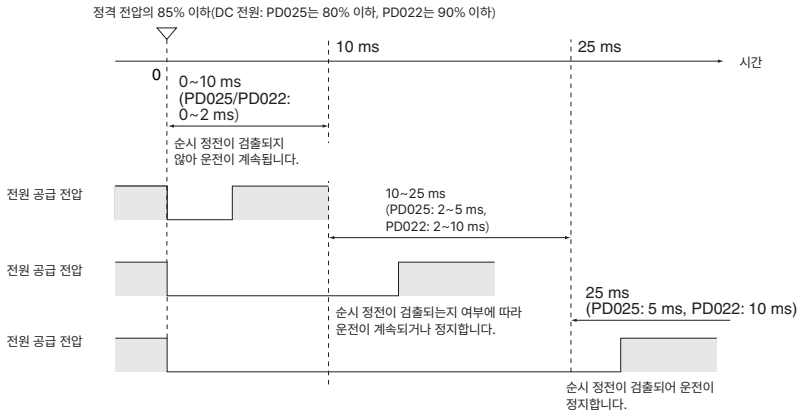
・순시 정전이 10 ms 미만 지속되는 경우, 즉 최저 정격 전압의 85% 이하에서 85% 이상으로 복귀하는 데 걸리는 시간이 10 ms 미만인 경우에는 시스템이 무조건 계속 운전합니다.

주 DC 전원을 사용하는 경우, 2 ms 미만이 필요한 것은 CJ1W-PD025의 정격 전압이 80% 이하에서 80% 이상으로 복귀할 때, 또는 CJ1W-PD022의 정격 전압이 90% 이하에서 90% 이상으로 복귀할 때입니다.

・10 ms를 넘고 25 ms 미만 지속되는 순시 정전(DC 전원을 사용하는 경우, CJ1W-PD025는 2 ms를 넘고 5 ms 미만, CJ1W-PD022는 2 ms를 넘고 10 ms 미만)은 판별하기 어려워 정전이 검출될 수도 있고 검출되지 않을 수도 있습니다.

・순시 정전이 25 ms 이상 지속되면(DC 전원을 사용하는 경우, CJ1W-PD025는 5 ms 이상, CJ1W-PD022는 10 ms 이상) 시스템이 무조건 정지합니다.

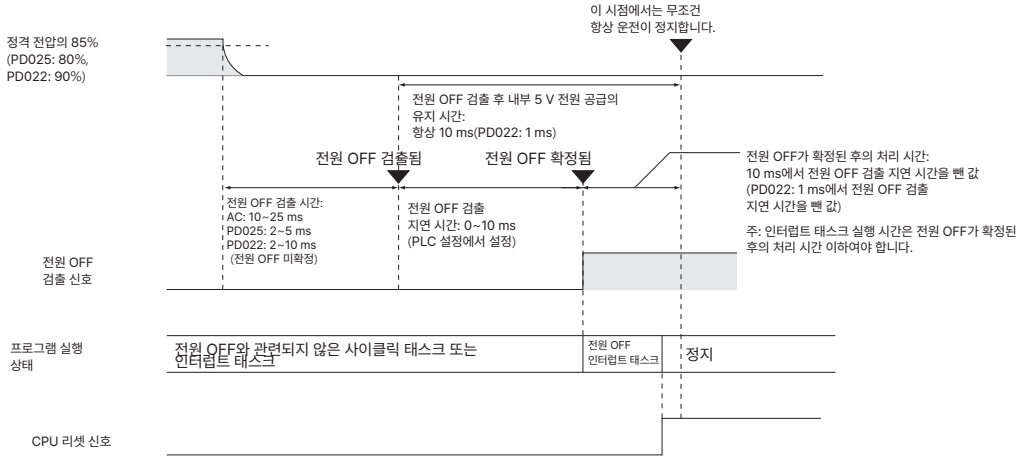
따라서 정전을 검출하는 데 필요한 시간은 10~25 ms입니다(DC 전원을 사용하는 경우, CJ1W-PD025는 2~5 ms, CJ1W-PD022는 2~10 ms). 위의 항목 2와 3의 조건에서 운전이 정지하는 경우, PLC 설정에서 전원 OFF 검출 지연 시간(0~10 ms)을 설정하여 운전을 정지시키는 타이밍(또는 전원 OFF 인터럽트 태스크의 실행을 개시하는 타이밍)을 늦출 수 있습니다. 다만 CJ1W-PD022 전원 유닛의 정전 시 5 VDC 출력 유지 시간은 1 ms이므로, 전원 OFF 인터럽트 태스크 금지 및 전원 OFF 검출 시간 파라미터는 사용할 수 없습니다.



주 위의 타이밍 차트는 전원 OFF 검출 지연 시간을 0 ms로 설정한 경우의 예를 나타냅니다.

전원 OFF 타이밍 차트

다음 타이밍 차트는 CPU 유닛의 전원 OFF 동작을 더 자세히 나타냅니다.



전원 OFF 검출 시간: 전원 공급이 최저 정격 전압의 85%(CJ1W-PD025 DC 전원 유닛은 80%, CJ1W-PD022 DC 전원 유닛은 90%) 미만으로 떨어진 후 전원 OFF를 검출하기까지 걸리는 시간.

전원 OFF 검출 지연 시간: 전원 OFF가 검출된 후 그것이 확정될 때까지의 지연 시간. PLC 설정에서 0~10 ms 범위로 설정할 수 있습니다.

불안정한 전원이 정전을 일으키는 경우에는 PLC 설정에서 전원 OFF 검출 지연 시간을 더 길게(최대 10 ms) 설정하십시오. 다만 CJ1W-PD022 전원 유닛은 유지 시간 1 ms만 지원하므로 이 설정은 불가능합니다.

전원 유지 시간: 전원이 차단된 후 내부적으로 5 V가 유지되는 최대 시간(10 ms 고정). 전원 OFF 인터럽트 태스크의 실행에 걸리는 시간은 10 ms에서 전원 OFF 검출 지연 시간(전원 OFF가 확정된 후의 처리 시간)을 뺀 값을 초과해서는 안 됩니다. 이 시간이 다 되는 순간 전원 OFF 인터럽트 태스크는 완전히 실행되지 않았더라도 종료됩니다. 다만 CJ1W-PD022 전원 유닛은 유지 시간 1 ms만 지원하므로 이 설정은 불가능합니다.

동작 설명

・100~240-VDC 전원 공급이 전원 OFF 검출 시간(AC 전원 유닛은 10~25 ms 사이, CJ1W-PD025 DC 전원 유닛은 2~5 ms 사이, CJ1W-PD022 DC 전원 유닛은 2~10 ms 사이) 동안 최저 정격 전압의 85%(CJ1W-PD025 DC 전원 유닛은 80%, CJ1W-PD022 DC 전원 유닛은 90%) 미만으로 떨어지면 전원 OFF가 검출됩니다.

・PLC 설정에서 전원 OFF 검출 지연 시간(0~10 ms)이 설정되어 있으면, 설정한 시간이 지났을 때 다음 동작이 수행됩니다.

・전원 OFF 인터럽트 태스크가 금지되어 있는 경우(PLC 설정의 디폴트)
CPU 리셋 신호가 ON되고 CPU가 즉시 리셋됩니다.

・전원 OFF 인터럽트 태스크가 (PLC 설정에서) 허가되어 있는 경우에는, 전원 OFF 인터럽트 태스크가 실행된 후 CPU 리셋 신호가 ON되고 CPU가 리셋됩니다. 전원 OFF 인터럽트 태스크가 10 ms에서 전원 OFF 검출 지연 시간을 뺀 값 = 전원 OFF 후의 처리 시간 안에 실행을 끝내도록 하십시오. 내부 5 V 전원 공급은 전원 OFF가 검출된 후 10 ms 동안만 유지됩니다. 다만 CJ1W-PD022 전원 유닛의 정전 시 내부 5 V 전원 공급 유지 시간은 1 ms이므로, 전원 OFF 검출 지연 시간과 전원 OFF 인터럽트 태스크 금지 기능은 사용할 수 없습니다.

사용상의 주의

무정전 전원 장치(UPS)를 사용하는 경우에는 사인파 출력 제품을 사용하십시오. 구형파 출력 제품은 사용하지 마십시오. 또한 정전 시 UPS의 전환 시간이 PLC의 전원 OFF 검출 시간보다 짧은지 확인하십시오. PLC의 전원 OFF 검출 시간이 UPS의 전환 시간보다 짧으면, UPS가 백업 전원을 공급하기 전에 PLC가 정전을 검출하여 운전을 정지합니다. PLC 설정에서 전원 OFF 검출 지연 시간을 설정하면 PLC의 전원 OFF 검출 시간을 늦출 수 있습니다.

A-5-2 정전 시의 명령어 실행

CPU 유닛이 RUN 또는 MONITOR 모드로 운전 중일 때 정전이 발생하여 정전이 확정되면, 현재 실행 중인 명령어가 완료된 후*1 다음 정전 처리가 수행됩니다.

・전원 OFF 인터럽트 태스크가 허가되어 있지 않으면 CPU 유닛이 즉시 리셋됩니다.

・전원 OFF 인터럽트 태스크가 허가되어 있으면 그 태스크가 실행된 후 CPU 유닛이 즉시 리셋됩니다.

전원 OFF 인터럽트 태스크의 허가 · 금지는 PLC 설정에서 설정합니다.

*1 현재 명령어는 실행을 완료하는 데 필요한 시간이 정전 검출 후의 처리 시간(10 ms – 정전 검출 지연 시간) 이하일 때만 완료될 수 있습니다. 이 시간 안에 명령어가 완료되지 않으면 명령어가 중단되고 위의 처리가 수행됩니다.

주 CJ1W-PD022가 장착된 경우 정전이 검출된 후의 처리 시간은 1 ms입니다.

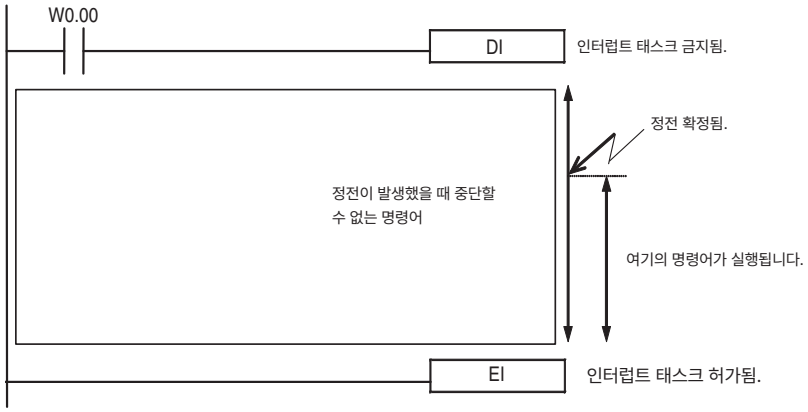
-시 번역본
-For Reference Only-

프로그램에서 정전 처리 금지

전원 OFF 인터럽트 태스크가 금지되어 있는 경우, 전원이 차단되더라도 CPU 유닛이 전원 OFF 처리를 수행하기 전에 명령어가 실행되도록 프로그램의 일부 영역을 정전으로부터 보호할 수 있습니다. 이것은 인터럽트 금지(DI(693))와 인터럽트 허가(EI(694)) 명령어를 사용하여 실현합니다.

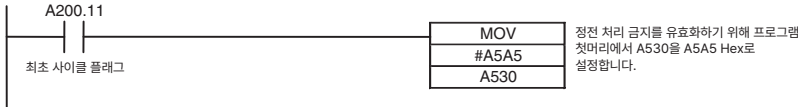
다음 순서를 사용합니다.

1 보호할 프로그램 구간 앞에 DI(693)을 삽입하여 인터럽트를 금지하고, 그 구간 뒤에 EI(694)를 두어 인터럽트를 허가합니다.



2 A530의 전원 OFF 인터럽트 금지 설정을 A5A5 Hex로 설정하여 정전 처리 금지를 유효화합니다.

주 A530은 통상적으로 전원이 OFF될 때 클리어됩니다. 이를 방지하려면 IOM 유지 비트(A500.12)를 ON하고 기동 시 IOM 유지 비트 설정을 유지하도록 PLC 설정을 설정하거나, A530을 A5A5 Hex로 설정하는 다음과 같은 명령어를 프로그램 첫머리에 넣어야 합니다.



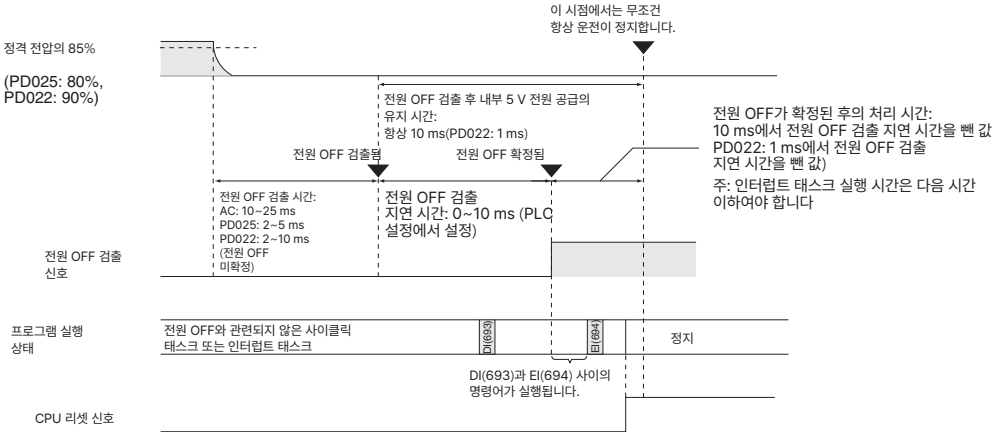
3 PLC 설정에서 전원 OFF 인터럽트 태스크를 금지합니다.
위의 순서를 따르면, DI(693)과 EI(694) 사이의 명령어를 실행하는 중에 정전이 발생하더라도 DI(693)과 EI(694)(또는 END) 사이의 모든 명령어가 완료된 후*1 전원 OFF 인터럽트가 실행됩니다.

*1 보호된 명령어는 실행을 완료하는 데 필요한 시간이 정전 검출 후의 처리 시간(10 ms − 정전 검출 지연 시간) 이하일 때만 완료될 수 있습니다. 이 시간 안에 명령어가 완료되지 않으면 명령어가 중단되고 위의 처리가 수행됩니다.

주 1 CJ1W-PD022가 장착된 경우 정전이 검출된 후의 처리 시간은 1 ms입니다.

2 PLC 설정에서 전원 OFF 인터럽트 태스크가 금지되어 있지 않으면, 전원 OFF 인터럽트 태스크가 실행되고 정전이 검출되는 즉시 보호된 명령어를 실행하지 않은 채 CPU 유닛이 리셋됩니다.

3 DI(693)이 실행되는 중에 정전이 검출되면, 보호된 명령어를 실행하지 않은 채 CPU 유닛이 리셋됩니다.



A530(전원 인터럽트 금지 영역 설정)이 A5A5 hex 이외의 값으로 설정되어 정전 처리 마스킹이 유효화되면, 현재 명령어의 실행이 완료된 후 전원 인터럽트 처리가 개시됩니다.

참고

인터럽트 처리는 A530의 내용과 PLC 설정에 따라 아래와 같이 수행됩니다.

전원 인터럽트 처리		A530(전원 인터럽트 금지 영역 설정)	
		A5A5 hex(전원 인터럽트 처리 마스킹 유효)	그 밖(전원 인터럽트 처리 마스킹 무효)
전원 OFF 인터럽트 태스크 (PLC 설정)	**금지**	DI(693)과 EI(694) 사이의 모든 명령어가 실행되고 CPU 유닛이 리셋됩니다.	현재 명령어의 실행이 완료되고 CPU 유닛이 리셋됩니다.
	허가	현재 명령어의 실행이 완료되고, 전원 OFF 인터럽트 태스크가 실행된 후 CPU 유닛이 리셋됩니다.	

AI 번역본
-For Reference Only-

A-6 Windows 7 이후 Windows OS에서의 EtherNet/IP 접속

Windows 7 이후 운영체제의 방화벽 보안이 강화되어 이더넷 포트의 데이터 통신에 대한 제한이 늘었습니다. 컴퓨터의 이더넷 포트에서 CPU 유닛에 EtherNet/IP 접속을 사용할 때는, CX-Programmer 통신을 사용할 수 있도록 Windows Firewall 설정을 변경해야 합니다.

⚠ 경고

- OS의 취약성으로 인한 보안 위험을 피하기 위해 컴퓨터의 OS를 최신 상태로 유지하십시오.
- 새로운 기능 추가, 조작성 향상, 보안 강화를 위해 항상 이 소프트웨어의 최신 버전을 사용하십시오.

사용상의 주의

- CX-One를 포함한 각 주변 툴이 지원하는 Windows 운영체제(OS)에 관한 자세한 내용은 오므론 웹사이트(www.fa.omron.co.jp 또는 www.ia.omron.com)를 참조하십시오.
- Windows Firewall은 주로 외부 기기로부터의(예: 인터넷을 통한) 부적절한 액세스를 방지하기 위한 것입니다. 이 문서에 설명된 Windows Firewall 설정 변경은 CX-Programmer가 EtherNet/IP 접속을 사용할 수 있게 합니다. 같은 컴퓨터를 사내 네트워크나 기타 네트워크에서 사용하고 있는 경우에는, 변경을 진행하기 전에 그 변경이 보안 문제를 일으키지 않는지 확인하십시오. 이 문서에 설명된 변경은 이더넷 포트를 통해 EtherNet/IP로 접속할 때만 필요합니다. USB 포트 등 다른 포트로 접속하는 경우에는 변경이 필요하지 않습니다.

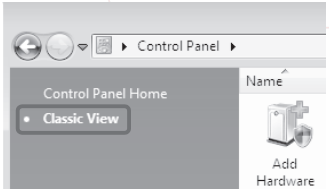
A-6-1 Windows Firewall 설정 변경

Windows 7 이후

CX-Programmer에서 접속을 시도하기 전에 다음 순서로 설정을 변경하십시오.
이 순서 중에 User Account Control 대화 상자가 표시될 수 있습니다. 표시되면 Continue 버튼을 클릭하고 순서를 계속하십시오.

AI 번역본
For Reference Only

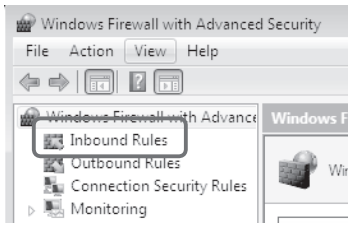
1 Windows 시작 메뉴에서 Control Panel을 선택하고 표시를 Classic View로 변경합니다.



2 Administrative Tools를 열고 표시된 대화 상자에서 Windows Firewall with Advanced Security를 선택합니다.



3 Windows Firewall with Advanced Security 대화 상자 왼쪽의 Windows Firewall with Advanced Security on Local Computer 아래에서 Inbound Rules를 선택합니다.



4 대화 상자 오른쪽의 Actions 영역에 있는 Inbound Rules 아래에서 New Rule을 선택합니다.



5 New Inbound Rule Wizard 대화 상자에서 각 단계에 대해 다음 설정을 하고, Next 버튼을 클릭하여 단계 사이를 이동합니다.

Rule Type	Custom을 선택합니다.
Program	All Programs를 선택합니다.
Protocol and Ports	프로토콜 종류로 ICMPv4를 선택합니다. 
Scope	전부 Any IP address를 선택합니다.
Action	Allow the connection을 선택합니다.
Profile	Domain, Private, Public을 선택합니다.
Name	임의의 이름(예: Omron_EIP)을 입력합니다.

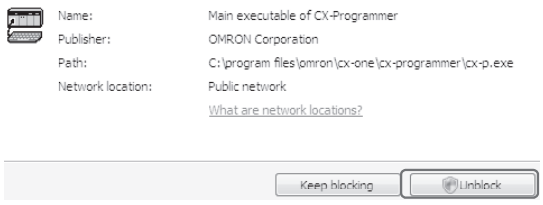
6 Finish 버튼을 클릭합니다. 정의한 규칙이 Inbound Rules에 등록됩니다(예: Omron_EIP).



7 Windows Firewall with Advanced Security 대화 상자를 닫습니다.

8 이더넷 포트를 통해 EtherNet/IP 네트워크상의 PLC에 CX-Programmer를 접속하려고 하면 Windows Security Alert 대화 상자가 표시됩니다.

9 Unblock 버튼을 클릭합니다.



CX-Programmer로부터의 EtherNet/IP 접속이 허용되며, 이후에도 EtherNet/IP 접속을 사용할 수 있게 됩니다.

AI 번역본
-For Reference Only-

A-7 유닛 버전별 지원 기능

A-7-1 CJ2H CPU 유닛

유닛 버전 1.6 이후에서 지원되는 기능

유닛 버전 1.6에서 추가된 기능을 사용하려면 CX-Programmer 버전 9.81 이상을 사용해야 합니다.

유닛	CJ2H CPU 유닛	
형식	CJ2H-CPU6□ *1	
유닛 버전 / 기능	유닛 버전 1.6 이후	유닛 버전 1.5 이전
네트워크 심볼 및 태그 액세스 기능	지원.	지원하지 않음.

*1 유닛 버전 1.6의 CJ2H-CPU6□-EIP에는 유닛 버전 1.5에 대해 추가된 기능이 없습니다.

유닛 버전 1.5 이후에서 지원되는 기능

유닛 버전 1.5에서 추가된 기능을 사용하려면 CX-Programmer 버전 9.6 이상을 사용해야 합니다.

유닛	CJ2H CPU 유닛	
형식	CJ2H-CPU6□-EIP CJ2H-CPU6□	
유닛 버전 / 기능	유닛 버전 1.5 이후	유닛 버전 1.4 이전
패스워드를 사용한 읽기 보호	지원.	지원하지 않음.

유닛 버전 1.4 이후

유닛 버전 1.4에서 추가된 기능을 사용하려면 CX-Programmer 버전 9.3 이상을 사용해야 합니다.

유닛	CJ2H CPU 유닛	
형식	CJ2H-CPU6□-EIP CJ2H-CPU6□	
유닛 버전 / 기능	유닛 버전 1.4 이후	유닛 버전 1.3 이전
동기 유닛 동작 기능 EtherCAT 인터페이스를 갖춘 위치 제어 유닛 CJ1W-NC□82는 동기 유닛 동작에 대응합니다.	지원.	지원하지 않음.

-For Reference Only-

유닛 버전 1.3 이후에서 지원되는 기능

유닛 버전 1.3에서 추가된 기능을 사용하려면 CX-Programmer 버전 9.1 이상을 사용해야 합니다.

유닛		CJ2H CPU 유닛			
형식		CJ2H-CPU6□-EIP CJ2H-CPU6□			
기능	유닛 버전	유닛 버전 1.3 이후	유닛 버전 1.2	유닛 버전 1.1	유닛 버전 1.0
일부 특수 I/O 유닛용 특수 명령어	CJ1W-NC281/NC481/NC881 위치 제어 유닛: PCU 고속 위치 결정 (NCDMV(218))	지원.	지원하지 않음.	지원하지 않음.	지원하지 않음.
	CJ1W-NC281/NC481/NC881 위치 제어 유닛: PCU 위치 결정 트리거 (NCDTR(219))	지원.	지원하지 않음.	지원하지 않음.	지원하지 않음.
새로 추가된 특수 명령어	부호 있는 영역 범위 비교: ZCPS(088)	지원.	지원하지 않음.	지원하지 않음.	지원하지 않음.
	부호 있는 영역 범위 비교 (더블): ZCPSL(116)	지원.	지원하지 않음.	지원하지 않음.	지원하지 않음.

유닛 버전 1.2 이후

유닛 버전 1.2에서 추가된 기능을 사용하려면 CX-Programmer 버전 8.3 이상을 사용해야 합니다.

유닛	CJ2H CPU 유닛	
형식	CJ2H-CPU6□-EIP CJ2H-CPU6□	
기능 / 유닛 버전	유닛 버전 1.2 이후	유닛 버전 1.2 미만
EM 영역 강제 세트/리셋	지원.	지원하지 않음.

주 유닛 버전 1.2 이후 CJ2H CPU 유닛의 기능을 사용하는 사용자 프로그램은 CJ2H
유닛 버전 1.1 이전의 CPU 유닛에서는 사용할 수 없습니다. 이러한 기능을 사용하는 프로그램을 CX-Programmer에서
유닛 버전 1.1 이전의 CPU 유닛으로 전송하려고 하면 에러가 표시되고 CPU 유닛에 다운로드할 수 없습니다.

AI 번역본
-For Reference Only-

유닛 버전 1.1 이후에서 지원되는 기능

유닛 버전 1.1에서 추가된 기능을 사용하려면 CX-Programmer 버전 8.1* 이상을 사용해야 합니다.

유닛	CJ2H CPU 유닛	
형식	CJ2H-CPU6□-EIP CJ2H-CPU6□	
유닛 버전 기능	유닛 버전 1.1	유닛 버전 1.0
고속 인터럽트 기능 인터럽트 태스크의 오버헤드 타임 단축 정주기 인터럽트 태스크 실행의 최소 간격: 0.1 ms	지원.	지원하지 않음.
운전 중 최소 사이클 타임 설정 변경	지원.	지원하지 않음.
동기 유닛 동작	지원.	지원하지 않음.
일부 특수 I/O 유닛용 즉시 리프레시 명령어 추가 CJ1W-AD042 고속 아날로그 입력 유닛: 아날로그 입력 다이렉트 변환(AIDC(216)) CJ1W-DA042V 고속 아날로그 출력 유닛: 아날로그 출력 다이렉트 변환(AODC(217))	지원.	지원하지 않음.
고속 시리얼 통신 유닛 CJ1W-SCU22/SCU32/SCU42 고속 시리얼 통신 유닛: 시리얼 통신 유닛 경유 다이렉트 수신(DRXDU(261)) 및 시리얼 통신 유닛 경유 다이렉트 송신 (DTXDU(262)	지원.	지원하지 않음.

* 고속 인터럽트 기능과 운전 중 최소 사이클 타임 설정 변경 기능을 사용하려면 CX-Programmer 버전 8.02 이상이 필요합니다.

주 유닛 버전 1.1 이후 CJ2H CPU 유닛의 기능을 사용하는 사용자 프로그램은 CJ2H 유닛 버전 1.0 이전의 CPU 유닛에서는 사용할 수 없습니다. 이러한 기능을 사용하는 프로그램을 CX-Programmer에서 유닛 버전 1.0의 CPU 유닛으로 전송하려고 하면 에러가 표시되고 CPU 유닛에 다운로드할 수 없습니다. 이러한 기능을 사용하는 프로그램 파일(확장자: .OBJ)을 유닛 버전 1.0의 CPU 유닛에 전송하면, 운전이 개시될 때 또는 그 기능이 시작될 때 프로그램 에러가 발생하고 CPU 유닛의 운전이 정지합니다.

AI 번역본
-For Reference Only-

A-7-2 CJ2M CPU 유닛

유닛 버전 2.2 이후에서 지원되는 기능

유닛 버전 2.2에서 추가된 기능을 사용하려면 CX-Programmer 버전 9.81 이상을 사용해야 합니다.

유닛	CJ2M CPU 유닛	
형식	CJ2M-CPU1□ *1	
기능 \ 유닛 버전	유닛 버전 2.2 이후	유닛 버전 2.1 이전
네트워크 심볼 및 태그 액세스 기능	지원.	지원하지 않음.

*1 유닛 버전 2.2의 CJ2M-CPU3□에는 유닛 버전 2.1에 대해 추가된 기능이 없습니다.

유닛 버전 2.1 이후에서 지원되는 기능

유닛 버전 2.1에서 추가된 기능을 사용하려면 CX-Programmer 버전 9.6 이상을 사용해야 합니다.

유닛	CJ2M CPU 유닛	
형식	CJ2M-CPU□□	
기능 \ 유닛 버전	유닛 버전 2.1 이후	유닛 버전 2.0 이전
패스워드를 사용한 읽기 보호	지원.	지원하지 않음.

유닛 버전 2.0 이후에서 지원되는 기능

유닛 버전 2.0에서 추가된 기능을 사용하려면 CX-Programmer 버전 9.12 이상을 사용해야 합니다.

유닛	CJ2M CPU 유닛	
형식	CJ2M-CPU□□	
기능 \ 유닛 버전	유닛 버전 2.0 이후	유닛 버전 1.0
펄스 출력	지원.	지원하지 않음.

AI 번역본
-For Reference Only-

AI 번역본
-For Reference Only-

색인

A

어드레스 오프셋 5-97
자동 어드레스 할당 5-62, 6-30
자동 백업 10-41
자동 온라인 접속 11-3, 11-14
기동 시 자동 전송 7-13, 10-28
보조 기억 영역 6-2, 6-25, A-111

B

백그라운드 실행 10-11
백그라운드 실행 설정 9-11
백업 메모리 10-41
백업 · 복원 조작 7-13
백업 서비스 3-5
기본 I/O 유닛 랙 응답 시간 9-20
배터리리스 운전 9-9
블록 프로그램 섹션 5-112
내장 플래시 메모리 2-2
내장 RAM 2-3

C

프로그램 체크 5-100
CIO 영역 6-2, 6-6, 6-9
  CPU 버스 유닛 영역 6-2
  데이터 링크 영역 6-2
  DeviceNet 영역 6-2
  I/O 영역 6-2
  내부 I/O 영역 6-2
  특수 I/O 유닛 영역 6-2
시계 기능 10-3
클록 펄스 ..... 6-3, ...6-47
코멘트 파일 7-10
코멘트 메모리 10-44
코멘트 2-4
FB 내 통신 명령어 설정 9-12
통신 포트 서비스 3-5
통신 설정 9-22
조건 플래그 ..... 5-107, 6-3, ...6-45
일정 사이클 타임 9-14
카운터 영역 ..... 6-3, ...6-36
  카운터 완료 플래그 영역 6-3
  카운터 완료 플래그 6-36
  카운터 현재값 영역 6-3
  카운터 현재값 6-36
CPU 버스 유닛 영역 ..... 6-2, ...6-16
CPU 버스 유닛 셋업 2-4
CPU 버스 유닛 셋업 영역 ..... 4-6, ...4-16
CPU 버스 유닛
  I/O 할당 ..... 6-16, ...6-29
CPU 처리 모드 9-29
CPU 유닛 설정 9-8
현재 뱅크 6-32
사이클 타임 3-4
  사이클 타임 계산 12-4
  사이클 타임 모니터 12-2
사이클 타임 연장 12-13
사이클릭 리프레시 5-82
사이클릭 태스크 5-14

D

데이터 파일 7-9
데이터 형식 5-78
데이터 링크 영역 6-2, 6-14
데이터 메모리 영역 6-2, 6-27
데이터 레지스터 6-3, 6-43
데이터 구조체 5-59
데이터형 5-57
디버깅 10-80
시뮬레이터를 사용한 디버깅 5-101
DeviceNet 영역 6-2, 6-20
FB 내 DeviceNet 통신 명령어 9-13
미분 모니터 10-81
딥 스위치 4-2
SIOU 사이클릭 리프레시 무효 9-18
전원 OFF 인터럽트 금지 10-26
배터리 전압 저하 검출하지 않음 9-8

E

EM 파일 메모리 7-2
EM 파일 메모리 기능 10-43
프로그램 덮어쓰기 허가/금지 10-58
메모리 카드 저장 허가/금지 10-57
에러
  불법 명령어 에러 5-105
  프로그램 에러 5-106
  사용자 프로그램 영역 오버플로 에러 5-105
에러 시뮬레이션 기능 5-103
CPU 버스 유닛 이벤트 서비스 3-5
특수 I/O 유닛 이벤트 서비스 3-5
실행 처리 9-8
실행 설정 9-7
확장 데이터 메모리 영역 ..... 6-3, ...6-30
외부 인터럽트 태스크 5-27

F

고장 알람 명령어 10-94
FAL(006) 명령어 10-94
FALS(007) 명령어 10-94
FB 프로그램 영역 2-4
파일 액세스 서비스 3-5
파일 메모리 6-31, 7-2
  코멘트 파일 7-10
  데이터 파일 7-9
  파라미터 파일 7-9
  프로그램 인덱스 파일 7-10
  프로그램/네트워크 심볼 파일 7-8
  심볼 테이블 파일 7-10
  유닛 백업 파일 7-10
기동 시 자동 전송되는 파일 7-15
FINS 보호 9-30
태스크 관련 플래그 5-33
강제 세트/리셋 10-80

사내 참고용
-For Reference Only-

강제 상태 유지 비트 기동 시 유지 설정 9-5
FOR-NEXT 루프 5-112
FPD(269) 명령어 10-96
프리런 타이머 10-7
펑션 블록 5-42

G

범용 파일 7-16
글로벌 심볼 5-49, 5-51

H

고속 입력 10-10
고속 인터럽트 기능 10-20
유지 영역 6-2, 6-23
동작 모드 변경 및
  기동 시의 유지 설정 10-23
호스트 링크 시스템 11-19

I

I/O 할당 8-2
  자동 할당 8-5
  수동 할당 8-9
I/O 영역 6-2
I/O 인터럽트 태스크 5-25
I/O 메모리 영역 2-4
I/O 메모리 영역 6-2
I/O 모듈 9-32
I/O 리프레시 시간 12-7
I/O 리프레시 타이밍 5-82
I/O 리프레시 .....3-6, 6-9, ...6-11
I/O 응답 시간 12-13
I/O 테이블 8-3
  자동 할당 4-10
  수동 할당 4-10
I/O 테이블 .....2-4, ...4-5
IL - ILC 섹션 5-112
불법 명령어 에러 5-105
즉시 리프레시 .....5-83, 6-10, ...6-12
인덱스 레지스터 .....5-87, 6-3, ...6-38
파일 메모리 초기화 7-3
입력 비트 6-9
입력 인터럽트 태스크 .....5-21, ...5-29
명령어 실행 조건 5-66
명령어 실행 시간 A-80
명령어 기능 A-3
명령어 베리에이션 5-66
통합 시뮬레이션 5-102
내부 I/O 영역 6-2
인터럽트 오버헤드 타임 5-22
인터럽트 태스크 5-20
인터럽트 태스크
  외부 인터럽트 태스크 5-20
  엑스트라 사이클릭 태스크 5-20
  I/O 인터럽트 태스크 5-20
  전원 OFF 인터럽트 태스크 5-20
  정주기 인터럽트 태스크 5-20
IOM 유지 비트 기동 시 유지 설정 9-6
IORF(097) 리프레시 .....6-10, ...6-12

J

JMP0 - JME0 섹션 5-112

L

래더 다이어그램 5-6
링크 영역 6-14
로컬 네트워크 테이블 4-12
로컬 심볼 5-49, 5-51

M

최대 사이클 타임 10-9
메모리 영역
  I/O 메모리 영역 2-4
  네트워크 심볼(태그) 2-4
  파라미터 영역 2-4
    CPU 버스 유닛 셋업 2-4
    I/O 테이블 2-4
    PLC 이름 2-4
    PLC 설정 2-4
    라우팅 테이블 2-4
  소스 · 코멘트 영역 2-4
    코멘트 2-4
    프로그램 인덱스 2-4
    소스 코드 2-4
    심볼 테이블 2-4
  사용자 프로그램 영역 2-4
메모리 카드 .....7-2, ...7-5
메모리 카드 인식 시간 7-6
메모리 관리 기능 10-41
PLC 메모리 어드레스의 메모리 맵 A-167
최소 사이클 타임 10-8
MONITOR 모드 3-8
사이클 타임 모니터 10-10

N

네트워크 심볼 5-52
네트워크 심볼(태그) 2-4
무순서 통신 시스템 11-20
NT 링크 시스템 11-21
스텝 수 A-80

O

온라인 에디팅 10-82
온라인 에디팅 서비스 3-5
동작 모드 3-8
동작 모드 9-7
  MONITOR 모드 3-8
  PROGRAM 모드 3-8
  RUN 모드 3-8
운전 종료 시간 10-6
정전 시의 동작 A-169
운전 개시 시간 10-6
출력 비트 6-11
공통 처리 3-4

AI 번역본
-For Reference Only-

P

파라미터 영역 2-4
파라미터 날짜 10-6
파라미터 파일 7-9
주변 서비스 9-29
주변 서비스 3-5
  백업 서비스 3-5
  통신 포트 서비스 3-5
  CPU 버스 유닛의 이벤트 서비스 3-5
  특수 I/O 유닛의 이벤트 서비스 3-5
  파일 액세스 서비스 3-5
  온라인 에디팅 서비스 3-5
  페리페럴 USB 포트 서비스 3-5
  시리얼 포트 서비스 3-5
PLC Backup Tool 7-15
PLC 이름 ..... 2-4, 4-4, ... 10-77
PLC 설정 ..... 4-4, 4-8, ... 9-2
  백그라운드 실행 설정 9-11
  기본 I/O 유닛 랙 응답 시간 9-20
  배터리리스 운전 9-9
  FB 내 통신 명령어 설정 9-12
  통신 설정 9-22
  CPU 처리 모드 9-29
  CPU 유닛 설정 9-8
  사이클 타임 9-14
  SIOU 사이클릭 리프레시 무효 9-18
  실행 처리 9-8
  실행 설정 9-7
  FINS 보호 9-30
  강제 상태 유지 비트 기동 시 유지 설정 9-5
  I/O 모듈 9-32
  IOM 유지 비트 기동 시 유지 설정 9-6
  동작 모드 9-7
  주변 서비스 9-29
  전원 OFF 검출 지연 시간 9-16
  전원 OFF 인터럽트 무효 9-17
  랙 응답 시간 9-20
  정주기 인터럽트 인터벌 9-15
  시리얼 포트 설정 9-22
  모든 이벤트에 대한 시간 설정 9-29
  네트워크를 경유한 FINS 쓰기 보호 설정 9-30
  특수 I/O 유닛 사이클릭 리프레시 9-18
  기동 시 유지 설정 9-5
  기동 시 동작 설정 9-5
  시간 · 인터럽트 설정 9-13
  감시 사이클 타임 9-13
PLC 설정 2-4
정전 시각 10-5
전원 OFF 검출 지연 설정 10-25
전원 OFF 검출 지연 시간 9-16
전원 OFF 인터럽트 무효 9-17
전원 OFF 인터럽트 태스크 ..... 5-22, ...9-17
전원 ON 클록 데이터 10-4
정전 시 처리 3-7
프로그램 용량 5-5
프로그램 에러 5-106
프로그램 실행 3-4
프로그램 인덱스 2-4
프로그램 인덱스 파일 7-10
PROGRAM 모드 3-8
프로그램 동작 보호
  제조 로트 번호 10-59
프로그램 섹션
  블록 프로그램 섹션 5-112
  FOR-NEXT 루프 5-112
  IL - ILC 섹션 5-112
  JMP0 - JME0 섹션 5-112
  스텝 래더 섹션 5-112
  서브루틴 5-112
프로그램/네트워크 심볼 파일 7-8
프로그래밍 언어 5-4
프로토콜 매크로 11-23
펄스 I/O 영역 ..... 6-2, ...6-18

R

패스워드에 의한 읽기 보호 10-53
중계 네트워크 테이블 4-12
전체 프로그램 교체 10-45
시리얼 PLC 링크의 응답 성능 12-15
입력 인터럽트 응답 시간 12-14
리트라이 횟수 9-12
로터리 스위치 4-2
라우팅 테이블 2-4, 4-5, 4-11
RUN 모드 3-8
RUN 출력 10-27

S

안전 주의사항 1-19
정주기 인터럽트 인터벌 9-15
정주기 인터럽트 태스크 5-23
섹션 5-40
보안 기능 10-53
  패스워드에 의한 읽기 보호 10-53
  FINS 커맨드에 의한 쓰기 보호 10-60
  딥 스위치에 의한 쓰기 보호 10-53
자기 유지 비트 6-23
시리얼 PLC 링크 11-24
시리얼 PLC 링크 영역 ..... 6-2, ...6-19
시리얼 포트 설정 9-22
모든 이벤트에 대한 시간 설정 9-29
DM 영역 할당 워드 설정
  CPU 버스 유닛 4-15
DM 영역 할당 워드 설정
  특수 I/O 유닛 4-15
CPU 버스 유닛과 특수 I/O 유닛의 설정 8-20
네트워크를 경유한 FINS 쓰기 보호 설정 9-30
SFC 언어 5-9
간이 백업 파일 7-15
간이 백업 조작 7-13
시스템 에러 시뮬레이션 10-95
슬롯 선두 어드레스 설정 8-12
소스 및 코멘트 영역 2-4
소스 코드 2-4
특수 I/O 유닛 영역 6-2, 6-17
특수 I/O 유닛 사이클릭 리프레시 9-18
특수 I/O 유닛
  I/O 할당 6-29
특수 프로그램 섹션 5-112
배열 지정 5-58
ST 언어 5-8
기동 시 유지 설정 9-5
기동 시 초기화 3-2
기동 시 동작 설정 9-5
기동 설정과 메인터넌스 10-23
스텝 래더 섹션 5-112

-For Reference Only-

에러 시 정지 위치 저장 10-93
명령어의 구조
플래그 5-64
명령어 조건 5-64
오퍼랜드 5-65
파워 플로우 5-63
서브루틴 5-112
심볼 테이블 2-4
심볼 테이블 5-48
심볼 5-48
심볼 테이블 파일 7-10
동기 데이터 리프레시 영역 6-15
동기 동작 사이클 10-99
동기 유닛 동작 10-98

T

태그 5-52
태스크 제어 명령어 5-17
태스크 플래그 영역 6-3
태스크 플래그 6-37
태스크 5-11
일시 릴레이 영역 6-2, 6-26
테스트 입력 10-81
시간 · 인터럽트 설정 9-13
타임 영역
타이머 완료 플래그 영역 6-3
타이머 영역 .....6-3, ...6-34
타이머 PV 영역 6-3
메모리에 저장되는 시각
운전 개시 시각과 운전 종료 시각 10-6
정전 시각 10-5
전원 ON 클록 데이터 10-4
총 전원 ON 시간 10-5
사용자 프로그램 및 파라미터 날짜 10-6
총 전원 ON 시간 10-5
트레이스 메모리 6-31
데이터 트레이스 10-85
출력 OFF 10-84

U

유닛 백업 파일 7-10
유닛 관리 기능 10-37
사용자 프로그램 영역 2-4
사용자 프로그램 영역 오버플로 에러 5-105
사용자 프로그램 날짜 10-6
사용자 정의 데이터형 5-59

AI 번역본
-For Reference Only-

V

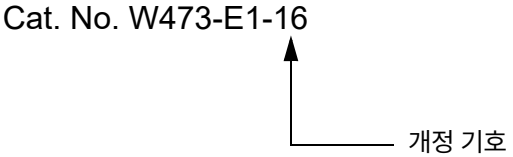
베리에이션 6-10

W

감시 사이클 타임 9-13
워크 영역 6-2, 6-22
FINS 커맨드에 의한 쓰기 보호 10-60
딥 스위치에 의한 쓰기 보호 10-53

개정 이력

매뉴얼의 앞표지에 있는 자료 번호 끝에 매뉴얼 개정 기호가 표기되어 있습니다.



다음 표는 각 개정에서 매뉴얼에 변경된 내용을 정리한 것입니다. 페이지 번호는 이전 판을 기준으로 합니다.

개정 기호	일자	개정 내용
01	July 2008	초판
02	December 2008	CJ2H-CPU6□ 추가. CJ2 CPU 유닛의 유닛 버전 1.1에 관한 정보 추가. 다음 기본 I/O 유닛 추가: CJ1W-ID212, CJ1W-ID233, CJ1W-OD213, CJ1W-OD234.
03	February 2009	동기 유닛 동작 기능에 관한 정보 추가. 다음 위치 제어 유닛 추가: CJ1W-NC214, CJ1W-NC234, CJ1W-NC414, CJ1W-NC434.
04	July 2009	CJ1W-AD042 아날로그 입력 유닛 추가. CJ1W-DA042V 아날로그 출력 유닛 추가.
05	October 2009	CJ1W-SCU22/32/42 시리얼 통신 유닛 추가. DRXDU(261) 및 DTXDU(262) 명령어 추가. EM 영역 강제 세트/리셋 기능(유닛 버전 1.2의 신규 기능)에 관한 정보 추가.
06	February 2010	CJ2M-CPU□□ 추가.
07	July 2010	CJ2M CPU 유닛용 CJ2M-MD21□ 펄스 I/O 모듈에 관한 정보 추가 오기 수정.
08	October 2010	CJ2M-CPU3□ CPU 유닛의 내장 EtherNet/IP 부 유닛 버전 2.1에 관한 정보 추가. 오기 수정.
09	March 2011	유닛 버전 1.4의 CJ2H CPU 유닛 추가. 동기 유닛 동작(CJ1W-NC□82용)에 관한 정보 추가. 오기 수정.
10	January 2015	오기 수정. 설명 추가.
11	October 2015	유닛 버전 1.5의 CJ2H CPU 유닛 추가. 유닛 버전 2.1의 CJ2M CPU 유닛 추가. 오기 수정.
12	July 2016	EtherNet/IP 로고 갱신. 오기 수정.
13	July 2020	오기 수정.
14	July 2022	보안에 관한 안전 주의사항 추가에 따른 개정.
15	July 2023	유닛 버전 1.6의 CJ2H CPU 유닛 추가. 유닛 버전 2.2의 CJ2M CPU 유닛 추가. CJ1W-EIP21S EtherNet/IP 유닛 추가. 오기 수정.
16	October 2026	CJ3M CPU 유닛 추가.

사내 배포용
-For Reference Only-

AI 번역본
-For Reference Only-

AI 번역본
-For Reference Only-

⚠ 안내 (For Reference Only)

본 문서는 AI 번역으로 작성된 참고용 국문자료 입니다.
일부 표현은 원문과 다소 다를 수 있으며,
중요한 내용은 원문과 함께 참고해 주시기 바랍니다.

한국오므론제어기기주식회사 https://www.ia.omron.co.kr/

서울특별시 서초구 강남대로 465, B동 18층(서초동, 교보타워)
TEL: 1522-8994 FAX: 02-3483-7788

Authorized Distributor:

©OMRON Corporation 2008 - 2026 All Rights Reserved.
In the interest of product improvement,
specifications are subject to change without notice.

Cat. No. W473-E1-16 1026