PLC-XGI 초급

본 자료는 PLC-XGI 초급 연수원 교육자료로 XGI를 제어하기 위해 CPU와 메모리의 개념을 이해하고 디바이스와 기본 명령어를 공부하여 프로그램을 구성할 수 있는 능력을 배양하기 위한 자료입니다.

본 교재는 XG5000 V4.23 기준으로 작성하였습니다.

목 차

1장. PLC 기초

1.1 PLC의 정의 및 적용분야 ---------- 4

1.2 PLC의 구조 ---------- 6

1.3 연산처리 ---------- 15

1.4 PLC의 동작원리 ---------- 17

2장. XGI 개요

2.1 XGI PLC의 특징 ---------- 21

2.2 소프트웨어 구조 ---------- 23

2.3 공통 요소 ---------- 25

2.4 LD(Ladder Diagram) ---------- 37

2.5 시스템 구성 ---------- 44

2.6 CPU 모듈 ---------- 47

3장. 데이터 메모리 구성

3.1 변수의 표현 방식 ---------- 55

Copyright ⓒ 2021 LS Electric Co., Ltd.

4장. XG5000 사용법

4.1 XG5000 특징 ---------- 63
4.2 기본 사용법 ---------- 65
4.3 프로젝트 ---------- 66
4.4 변수/설명 ---------- 68
4.5 LD 편집 ---------- 74
4.6 파라미터 ---------- 77
4.7 온라인 ---------- 82
4.8 모니터 ---------- 85

5장. 프로그래밍

5.1 시퀀스 프로그램 ---------- 93
5.2 펑션 프로그램 ---------- 102
5.3 펑션 블록 프로그램 ---------- 110

6장. 사용자정의 펑션/펑션블록

6.1 사용자정의 펑션 ---------- 129
6.2 사용자정의 펑션블록 ---------- 132
6.3 기타 기능 ---------- 133

부록1. 노이즈 대책 ---------- 134

부록2 유지보수 ---------- 136

Copyright ⓒ 2021 LS Electric Co., Ltd.

1장. PLC 기초

Copyright ⓒ 2021 LS Electric Co., Ltd.

1.1. PLC의 정의 및 적용분야

PLC의 정의

PLC(Programmable Logic Controller)란, 종래에 사용하던 제어반 내의 릴레이, 타이머, 카운터 등의 기능을LSI, 트랜지스터 등의 반도체 소자로 대체시켜, 기본적인 시퀀스 제어 기능에 수치 연산 기능을 추가하여 프로그램 제어가 가능하도록 한 자율성이 높은 제어 장치입니다.

미국 전기 공업회 규격(NEMA: National Electrical Manufacturers Association)에서는 "디지털 또는 아날로그 입출력 모듈을 통하여 로직, 시퀀싱, 타이밍, 카운팅, 연산과 같은 특수한 기능을 수행하기 위하여 프로그램 가능한 메모리를 사용하고 여러 종류의 기계나 프로세서를 제어하는 디지털 동작의 전자 장치"로 정의하고 있습니다.

PLC의 탄생 배경

1967 미국 자동차 업체인 GM에서 생산 기종 변경 시 자동차 제조 라인의 배선교체 작업을 간단히 하고 교체 비용과 시간 등을 절약하기 위하여 새로운 제어 기기의 조건을 제시하여 탄생한 기기가 PLC입니다. 당시에 10가지 조건을 제시하고 이에 가장 부합되는 기기를 만들어 줄 것을 주문하였는데 이 조건이 다음에 기술한 GM의 10가지 조건입니다.

1) 프로그램이 가능하고 프로그램의 작성과 변경이 용이할 것

2) 열악한 산업 환경에서도 작동가능 할 것

3) 입력은 AC 120V 신호 입력이 가능할 것

4) 출력은 액츄에이터를 직접 구동 가능할 것

5) 경제성이 있을 것

6) 확장성이 용이 할 것

7) 소형화가 가능 할 것

8) 신뢰성 및 유지 및 보수가 용이할 것

9) 통신이 가능 할 것

10) 4K 스텝 이상의 프로그램이 가능한 메모리를 가질 것

Copyright ⓒ 2021 LS Electric Co., Ltd.

PLC의 적용 분야

설비의 자동화와 고 능률화의 요구에 따라 PLC의 적용 범위는 확대 되고 있습니다. 특히 공장 자동화와 FMS(Flexible Manufacturing System)에 따른 PLC의 요구는 과거 중규모 이상의 릴레이 제어반 대체 효과에서 현재 고기능화, 고속화의 추세로 소규모 공작 기계에서 대규모 시스템 설비에 이르기 까지 적용되고 있습니다.

분 야	제 어 대 상
식료 산업	컨베이어 총괄 제어, 생산라인 자동 제어
제철, 제강 산업	작업장 하역 제어, 원료 수송 제어, 압연 라인 제어
섬유, 화학공업	원료 수입 출하 제어, 직조 염색 라인 제어
자동차 산업	전송 라인 제어, 자동 조립 라인 제어, 도장 라인 제어
기계 산업	산업용 로봇 제어, 공작 기계 제어, 송 · 배수 펌프 제어
상하수도	정수장 제어, 하수 처리 제어, 송 · 배수 펌프 제어
물류 산업	자동 창고 제어, 하역 설비 제어, 반송 라인 제어
공장 설비	압축기 제어
공해 방지사업	쓰레기 소각로 자동 제어, 공해 방지기 제어

PLC의 기능 요약

종래에 사용하던 제어반 내의 릴레이, 타이머, 카운터 등의 기능을 LSI, 트랜지스터 등의 반도체 소자로 대체시켜, 기본적인 시퀀스 제어 기능에 수치 연산, 아날로그, 고속카운터, PID제어, 모터 제어,통신 기능 등을 추가하여 사용자의 의도에 따라 프로그램 제어가 가능하도록 한 자율성이 높은 제어 장치

현장의 입력기기로부터 입력데이터를 받아들여 사용자가 미리 작성한 프로그램을 실행한 후, 그 결과를 출력기기를 통해 출력함으로써 기계나 설비를 제어하는 전자장치

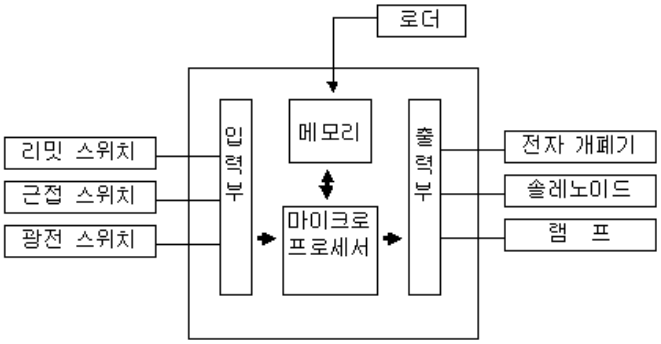
Copyright ⓒ 2021 LS Electric Co., Ltd.

1.2. PLC의 구조

하드웨어 구조

▶ 전체 구성

PLC는 마이크로프로세서(microprocessor) 및 메모리를 중심으로 구성되어 인간의 두뇌 역할을 하는 중앙처리장치(CPU), 외부 기기와의 신호를 연결시켜 주는 입·출력부, 각 부에 전원을 공급하는 전원부, PLC내의 메모리에 프로그램을 기록하는 주변 장치로 구성되어 있습니다.



▶ PLC의 CPU 연산부

PLC의 두뇌에 해당하는 부분으로서 메모리에 저장되어 있는 프로그램을 해독하여 실행합니다. CPU는 매우 빠른 속도로 반복 실행되며 모든 정보는 2진수로 처리됩니다.

▶ PLC의 CPU 메모리

IC 메모리 종류에는 ROM(Read Only Memory)과 RAM(Random Access Memory)이 있으며 ROM은 읽기 전용으로, 메모리 내용을 변경할 수 없습니다. 따라서, 맨 처음 한번 작성하면 이후에 변경되지 않는 시스템 관련된 프로그램을 저장하여 두는 역할을 합니다. ROM 영역의 정보는 전원이 끊어져도 메모리의 내용이 그대로 보존되는 비휘발성 메모리입니다. RAM은 메모리에 정보를 수시로 읽고 쓰기가 가능하여 정보를 일시 저장하는 용도로 사용되나, 전원이 끊어지면 기억시킨 정보 내용을 모두 상실하는 휘발성 메모리입니다. 그러나 필요에 따라 RAM 영역 일부를 전원이 오프 되어도 배터리에 의해 필요한 전원을 공급하여 메모리의 내용이 지워지지 않도록 하는 방법을 배터리 백업(Battery back-up)이라 하는데 이러한 방법을 통하여 RAM도 비휘발성 영역으로 사용할 수 있습니다. PLC의 데이터 영역과 사용자 프로그램은 변경이 가능해야 하므로 RAM영역에 저장됩니다.

PLC의 메모리는 사용자 프로그램 메모리, 데이터 메모리, 시스템 메모리 등의 3가지로 구분됩니다. 사용자 프로그램 메모리는 제어하고자 하는 시스템 규격에 따라 사용자가 작성한 프로그램이 저장되는 영역으로, 제어 내용이 프로그램 완성 전이나 완성 후에도 변경될 수 있으므로 RAM이 사용됩니다. 프로그램이 완성되어 고정되면, ROM에 기록하여 실행 할 수 있는데 이를 ROM운전 이라 합니다. 데이터 메모리는 입·출력 릴레이, 보조 릴레이, 타이머와 카운터의 접점 상태 및 설정 값, 현재 값 등의 정보가

Copyright ⓒ 2021 LS Electric Co., Ltd.

저장되는 영역으로 정보가 수시로 바뀌므로 RAM 영역이 사용됩니다. 시스템 메모리는 PLC 제작 회사에서 작성한 시스템 프로그램이 저장되는 영역입니다. 시스템 프로그램은 PLC의 명령어를 실행시켜주는 명령어 관련 프로그램과 자기 진단기능등과 같이 PLC동작시 발생하는 오류나 에러 등을 체크해주는 프로그램, XG5000과의 통신을 담당하는 프로그램 등등으로 구성되어 있으며 PLC 제작회사에서 파워를 ON/OFF하여도 지워지지 않도록 ROM에 저장하여 둡니다.

▶ PLC의 입출력부

PLC의 입출력부는 현장의 기기에 직접 접속하여 사용합니다. PLC 내부는 DC5V 의 전원(TTL 레벨)을 사용하지만 입출력부는 DC24V 또는 AC110V, 220V 등의 높은 전압 레벨을 사용하므로 PLC 내부 회로와 입·출력 회로의 접속(Interface)시 시스템 안정에 매우 많은 영향을 미치게 됨으로PLC의 입출력부는 다음과 같은 사항이 필수적으로 요구됩니다.

1) 외부 기기와 전기적 규격이 일치해야 합니다.

2) 외부 기기로부터의 노이즈가 CPU쪽에 전달되지 않도록 해야 합니다.

[광 절연 소자인 포토 커플러(Photocoupler) 사용]

3) 외부 기기와의 접속이 용이해야 합니다.

4) 입출력의 각 접점 상태를 감시할 수 있어야 합니다. (LED 부착) 입력부는 외부 기기의 상태를 검출하거나 조작 Panel을 통해 외부 장치의 움직임을 지시하고 출력부는 외부 기기를 움직이거나 상태를 표시합니다.

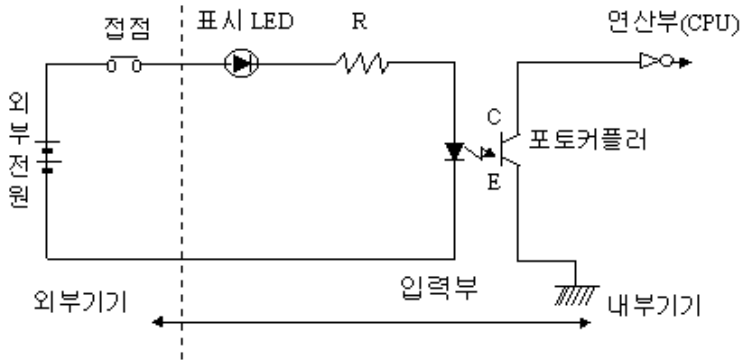
I/O	구 분	부 착 장 소	외부 기기의 명칭
입력부	조작 입력	제어반과 조작반	푸시 버튼 스위치 선택 스위치 토글 스위치
	검출 입력 (센서)	기계 장치	리밋 스위치 광전 스위치 근접 스위치 레벨 스위치
출력부	표시 경보 출력	제어반 및 조작반	파일럿 램프 부저
	구동 출력 (액추에이터)	기계장치	전자 밸브 전자 클러치 전자 브레이크 전자 개폐기

<입출력부에 접속되는 외부 기기 예>

Copyright ⓒ 2021 LS Electric Co., Ltd.

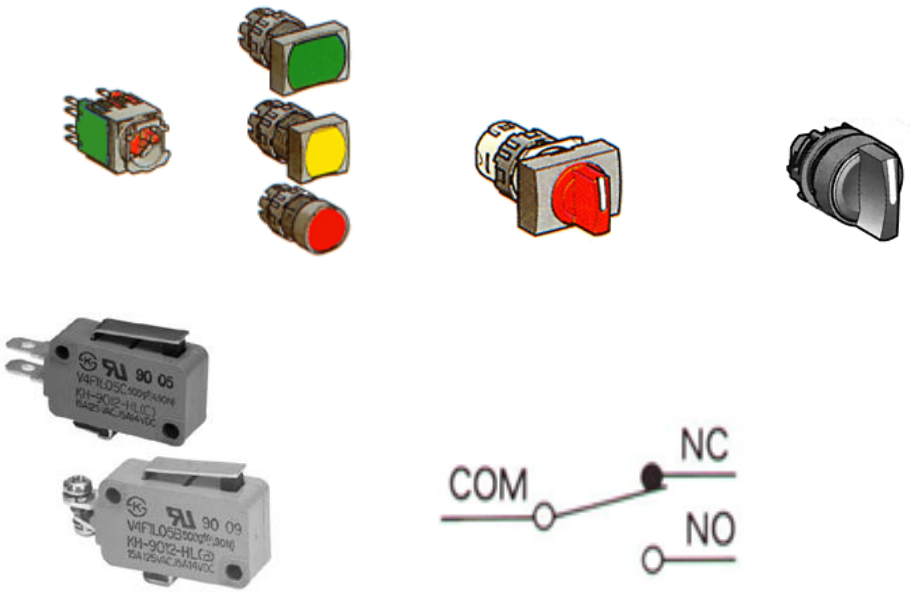
▶ 입력부

외부 기기로부터의 신호를 CPU의 연산부로 전달해 주는 역할을 합니다.. 입력의 종류로는 DC24[V], AC110[V] 등이 있고, 그 밖의 특수 입력 모듈로는 아날로그 입력(A/D) 모듈, 고속 카운터 (High Speed Counter) 모듈 등이 있습니다.



<DC24V 입력부 회로>

▶ 입력 스위치 종류 및 등가회로

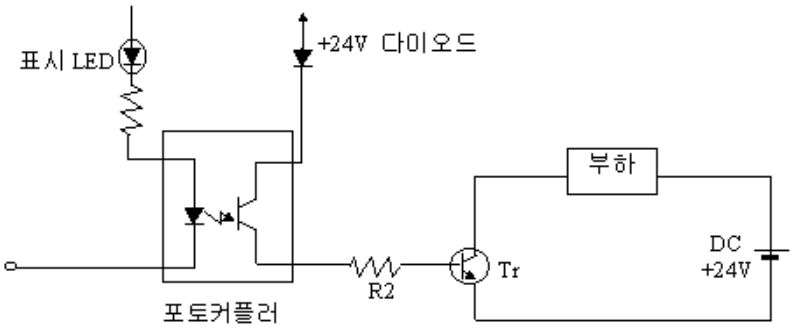


<입력 스위치 종류 및 등가회로>

▶ 출력부

내부 연산의 결과를 외부에 접속된 전자 접촉기나 솔레노이드에 전달하여 구동시키는 부분입니다. 출력의 종류에는 릴레이 출력, 트랜지스터 출력, SSR(Solid State Relay)출력 등이 있고, 그 밖의 출력 모듈로는 아날로그 출력(D/A) 모듈, 위치 결정 모듈 등이 있습니다. 트랜지스터 출력부 회로의 예는 아래와 같습니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.



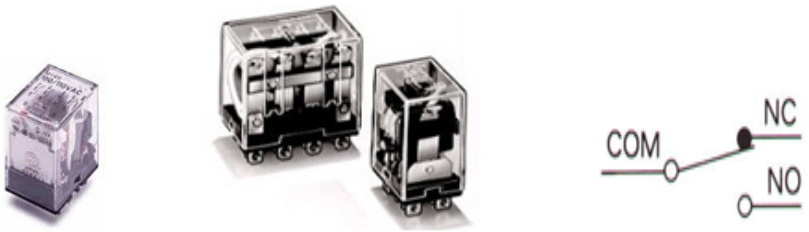
<트랜지스터의 출력부 회로>

출력 모듈을 출력 신호와 개폐 소자에 따라 분류하면 아래와 같습니다.

출력 사용 전원	개 폐 소 자	
	유접점	무접점 (반도체)
직류 (DC)	릴레이 출력	트랜지스터 출력
교류 (AC)	릴레이 출력	SSR 출력

<출력 모듈의 종류>

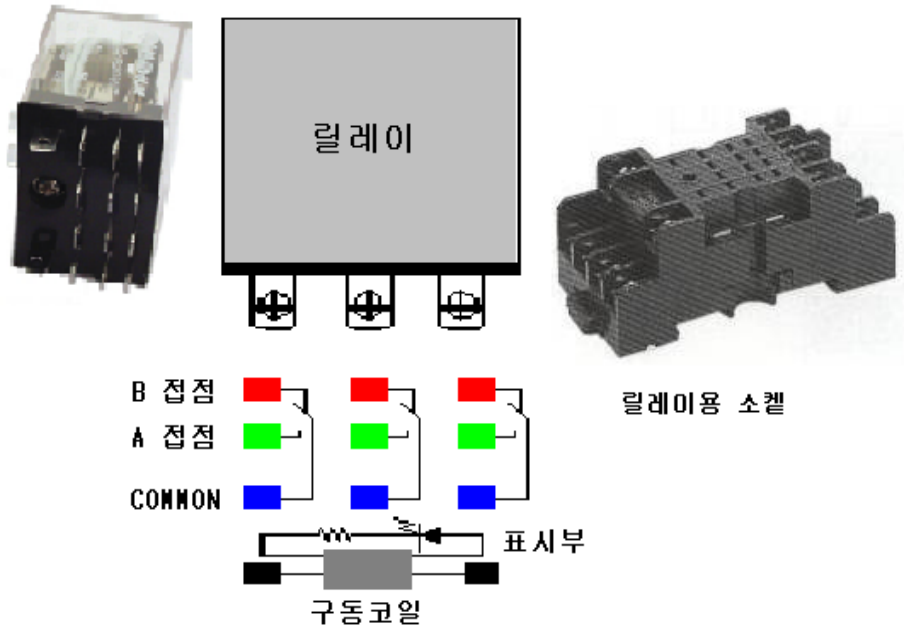
위의 표에서와 같이 릴레이 출력은 직류와 교류 모두 사용할 수 있으나, 기계적 수명의 한계 때문에 접점의 개폐가 빈번할 경우는 교류 전원 전용인 무접점 SSR 출력이나 직류 전원 전용인 트랜지스터 출력을 사용하는 것이 좋습니다.



<출력용 릴레이 종류 및 출력 접점>

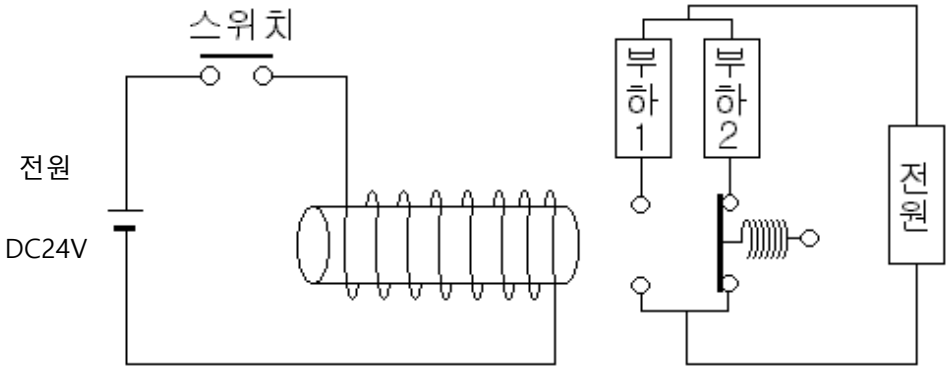
아래는 릴레이의 형태와 3회로용 릴레이 대한 그림입니다. 3회로용이므로 공통단자인 코먼(COMMON) 접점과 A, B 접점이 각각 3회로가 구성되었다는 의미입니다. 릴레이는 구동부인 코일과 동작상태를 표시해주는 표시부로 구성되어 있습니다. 릴레이는 기계적인 접점으로 수명에 한계(약 10만회~100만회)가 있기 때문에 릴레이용 소켓을 사용하여 고장 시에 교체가 쉽도록 설계하고 있습니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.



<릴레이의 형태 및 접점>

아래는 PLC 릴레이출력 구동 원리에 대한 그림입니다. 릴레이는 사용되는 전원이 DC24V용과 AC110V, 또는 AC220V용 등으로 구분됩니다. 따라서 사용되는 곳의 구동 전원에 따라서 릴레이를 선택해서 사용해야 합니다.

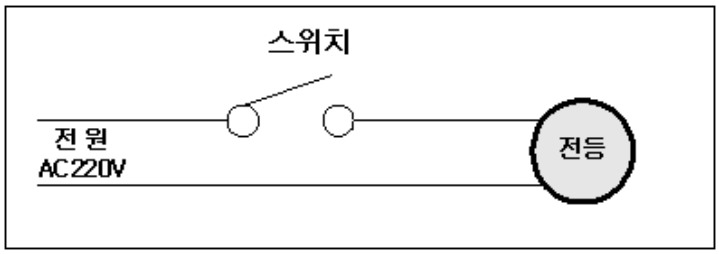


<릴레이의 구동 원리>

위 그림의 릴레이는 DC 24V용 릴레이입니다. 스위치를 온(ON)시키면 릴레이 코일에 전류가 흐르게 되고 전류가 흐름으로써 자력이 형성되어 철판을 끌어 당기게 됩니다. 따라서 부하2가 B접점으로 평상시에 접점이 닫혀있다가(CLOSE) 코일에 전류가 흐르게 되면 부하1쪽 접점이 닫히게 되고 부하2쪽 접점은 개방(OPEN) 되게 됩니다. 따라서 코일에 전류가 흐르지 않을 때 즉 스위치가 오프일 때는 부하2회로가 동작되고, 스위치가 온일 때는 부하1회로가 동작되게 됩니다. 스위치용 전원이 투입되지 않을 때 동작이 이뤄지는 접점을 B접점(부하2연결접점)이라 하고 스위치가 온 되어 릴레이가 동작 시에 온 되는 접점(부하1 연결접점)을 A접점이라 합니다.. 이러한 원리에 의하여 DC24V용 저전압용 전원의 스위치를 이용하여 AC110V 또는 AC220V용의 높은 전압의 부하를 온/오프 하여 제어가 가능하게 됩니다.

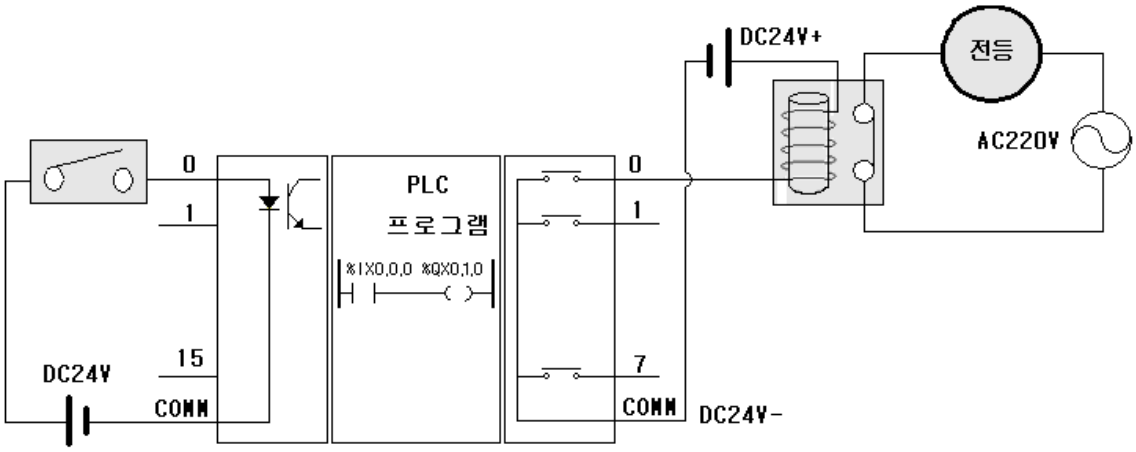
Copyright ⓒ 2021 LS Electric Co., Ltd.

아래는 일반적으로 사용되는 전등회로입니다. 스위치를 온/오프 함에 따라 전등이 온/오프 되는, 우리 주변에서 흔히 볼 수 있는 간단한 전등 온/오프 회로입니다.



<전기회로 구성>

위와 전기회로를 PLC로 제어할 경우에는 아래 그림처럼 스위치를 입력부, 전등을 출력부로 구분하여 회로를 별도로 구성해야 합니다. 입출력을 별도로 구성하고 입출력간의 연결 구성은 PLC의 프로그램에 의하여 작성합니다. PLC의 프로그램은 소프트웨어적으로 처리되기 때문에 수정 및 편집이 자유스럽습니다. 따라서 입출력만 결선하고 나면 프로그램에 의하여 모든 로직 회로를 자유롭게 변경 및 수정이 가능합니다.



<PLC의 회로 구성>

소프트웨어 구조

▶ 하드 와이어드와 소프트 와이어드

종래의 릴레이 제어 방식은 일의 순서를 회로도에 전개하여 그곳에 필요한 제어 기기를 결합하여 리드선으로 배선 작업을 해서 요구하는 동작을 실현합니다. 이 같은 방식을 하드와어어드 로직(Hardwired Logic)이라고 합니다.

하드와이어드 로직 방식에서는 하드웨어(기기)와 소프트웨어가 한 쌍이 되어 있어, 사양이 변경되면 하드웨어와 소프트웨어를 모두 변경해야 하므로, 여러 가지 문제를 발생시키는 원인이 됩니다. 따라서, 하드웨어와 소프트웨어를 분리하는 연구 끝에 컴퓨터 방식이 개발되었습니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

컴퓨터는 하드웨어(Hardware)만으로는 동작할 수 없습니다. 하드웨어 속에 있는 기억 장치에 일의 순서를 넣어야만 비로소 기대되는 일을 할 수가 있습니다. 이 일의 순서를 프로그램이라 하며, 기억 장치인 이 메모리에 일의 순서를 넣는 작업을 프로그래밍이라 합니다. 이는 마치 배선작업과 같다고 생각하면 됩니다. 이 방식을 소프트와이어드 로직(Softwired Logic)이라 하며, PLC는 이 방식을 취하고 있습니다.

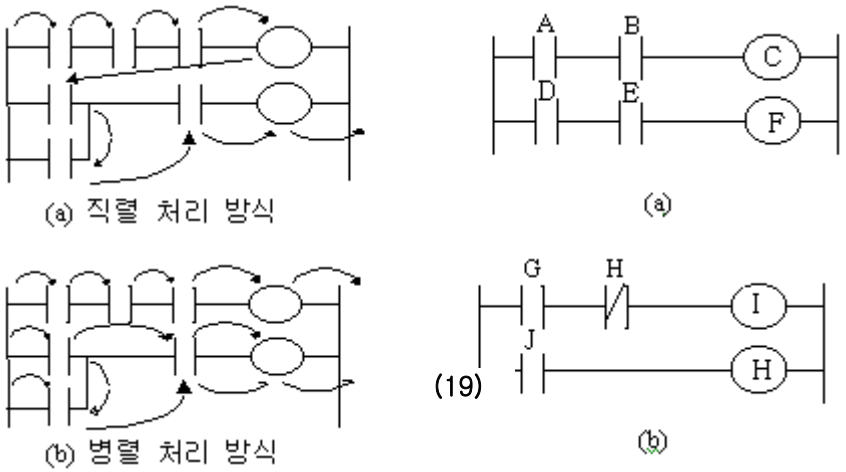
▶ 릴레이 시퀀스와 PLC 프로그램 차이점

PLC는 LSI등 전자 부품의 집합으로 릴레이 시퀀스와 같은 접점이나 코일은 존재하지 않으며, 접점이나 코일을 연결하는 동작은 소프트웨어로 처리되므로 실제로 눈에 보이는 것이 아닙니다. 또, 동작도 코일이 여자되면 접점이 닫혀 회로가 활성화되는 릴레이 시퀀스와는 달리 메모리에 프로그램을 기억시켜 놓고 순차적으로 내용을 읽어서 그 내용에 따라 동작하는 방식입니다. PLC제어는 프로그램의 내용에 의하여 좌우됩니다. 따라서 사용자는 자유 자재로 원하는 제어를 할 수 있도록 프로그램의 작성 능력이 요구됩니다.

직렬 처리와 병렬 처리

PLC 시퀀스와 릴레이 시퀀스의 가장 근본적인 차이점은 아래에 나타낸 것과 같이 "직렬 처리"와 "병렬 처리"라는 동작상의 차이에 있습니다.

PLC는 메모리에 있는 프로그램을 순차적으로 연산하는 직렬 처리 방식이고 릴레이 시퀀스는 여러 회로가 전기적인 신호에 의해 동시에 동작하는 병렬 처리 방식입니다. 따라서 PLC는 어느 한 순간을 포착해 보면 한 가지 일 밖에 하지 않습니다.



<연산처리 방식> <시퀀스도>

먼저 위의 그림에서 시퀀스도 (a)로 PLC와 릴레이의 동작상의 차이점을 설명합니다. 릴레이 시퀀스에

Copyright ⓒ 2021 LS Electric Co., Ltd.

서는 전원이 투입되어 접점 A와 B, 그리고 접점 D와 E가 동시에 닫히면, 출력 C와 F는 ON되고, 어느 한쪽이 빠를수록 먼저 동작합니다. 이에 비하면 PLC는 연산 순서에 따라 C가 먼저 ON되고 다음에 F가 ON됩니다.

PLC와 릴레이의 동작상의 차이점을 위 그림의 시퀀스도 (b)의 경우에서 살펴 보면 먼저 릴레이 시퀀스에서는 전원이 투입되면 점점 J가 닫힘과 동시에 H가 ON되어 출력 I는 동작될 수 없습니다.

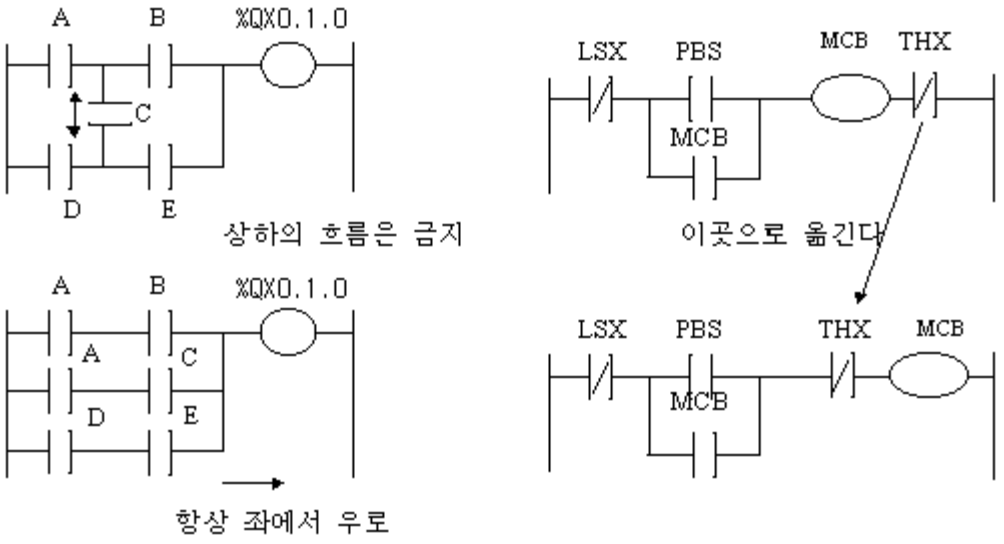
PLC는 직렬 연산 처리되므로 최초의 연산 때 G가 닫히면 I가 ON되고 J가 닫히면 H가 ON됩니다. H가 ON되면 b접점 H에 의해 I는 OFF됩니다.

▶ 사용 접점 수의 제한

릴레이는 일반적으로 1개당 가질 수 있는 접점 수에 한계가 있습니다. 따라서 릴레이 시퀀스를 작성할 때에는 사용하는 접점 수를 가능한 한 줄여야 합니다. 이에 비하여 PLC는 동일 접점에 대하여 사용 회수에 제한을 받지 않습니다. 이는 동일 접점에 대한 정보(ON/OFF)를 정해진 메모리에 저장해 놓고, 연산할 때 메모리에 있는 정보를 읽어서 처리하기 때문입니다.

▶ 접점이나 코일 위치의 제한

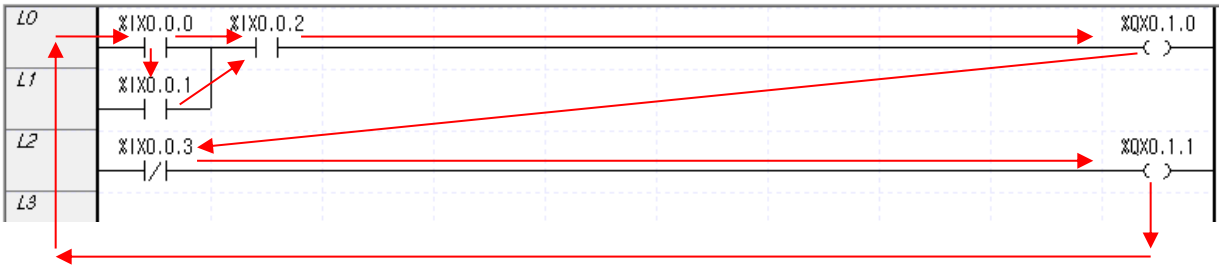
PLC 시퀀스에는 릴레이 시퀀스에는 없는 규정이 있습니다. 그 중 하나는 코일 이후 접점을 금지하는 사항입니다. 즉, PLC 시퀀스에서는 코일을 반드시 오른쪽 모선에 붙여서 작성해야 합니다. 또 PLC 시퀀스에서는 항상 신호가 왼쪽에서 오른쪽으로 전달되도록 구성되어 있습니다. 따라서, PLC 시퀀스는 릴레이 시퀀스와는 다르게 오른쪽에서 왼쪽으로 흐르는 회로나, 상하로 흐르는 회로 구성을 금지하고 있습니다. PLC 시퀀스의 규정을 그림 아래에 나타냅니다.



< PLC 시퀀스의 규정>

Copyright ⓒ 2021 LS Electric Co., Ltd.

PLC는 아래에서 보는 바와 같이 맨 위 좌측의 명령어부터 우측으로, 그리고 다시 아래방향으로 한 명령어씩 실행하게 됩니다. 이러한 방법으로 모든 명령어를 실행하고 나면 맨 마지막으로 END명령을 만나게 됩니다. END명령은 자기진단기능, 타이머, 카운터처리, 통신, 입출력 리프레시를 하고서 프로그램 실행 순서를 맨 처음으로 되돌려 주는 역할을 하게 됩니다. 따라서 프로그램 시작 부분으로 되돌아 가면서 동일한 방법으로 반복해서 프로그램을 연속하여 실행하게 됩니다.

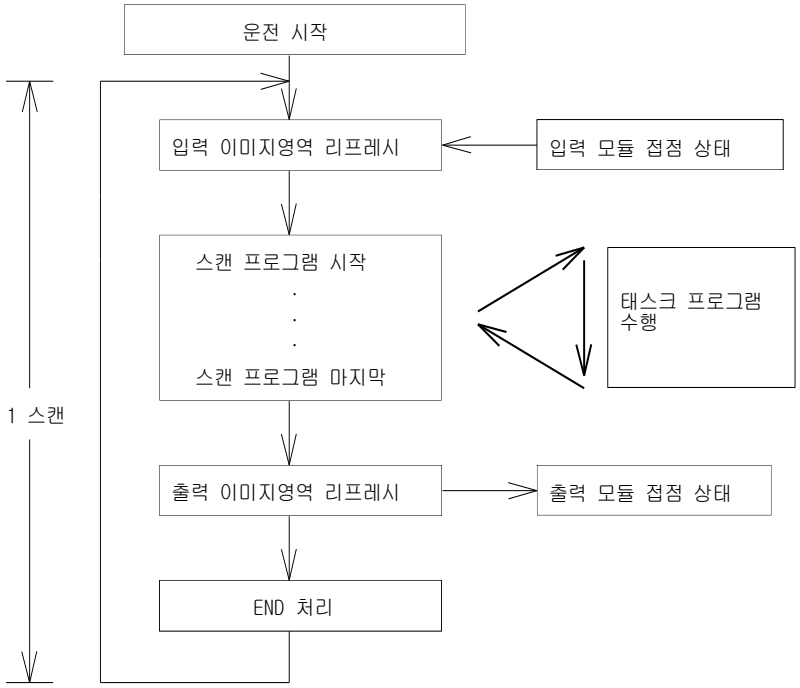


<PLC 시퀀스의 실행 순서>

Copyright ⓒ 2021 LS Electric Co., Ltd.

1.3. 연산 처리

PLC의 연산 처리 방법은 입력 리프레시(Refresh) 과정을 통해 입력의 상태를 PLC의 CPU가 인식하고, 인식된 정보를 조건 또는 데이터로 이용하여 프로그램 처음부터 마지막까지 순차적으로 연산을 실행하고 출력 리프레시(Refresh)를 합니다. 이러한 동작은 고속으로 반복되는데 이러한 방식을 '반복 연산 방식'이라 하고 한 바퀴를 실행하는데 걸리는 시간을 '1스캔 타임'(1연산 주기)라고 합니다.



입력 이미지 리프레시

PLC는 운전이 시작되면 입력 모듈을 통해 입력되는 정보들을 메모리의 입력 영역으로 받고, 이 정보들은 다시 입력 이미지 영역으로 복사되어 연산이 수행되는 동안의 입력 데이터로 이용됩니다. 이렇게 입력 영역의 데이터를 입력 이미지 영역으로 복사하는 것을 '입력 리프레시'(Input Refresh)라고 합니다. 입력 리프레시는 운전이 시작될 때뿐만 아니라 매 스캔 END처리가 끝나면 그 순간의 입력 정보를 입력 이미지 영역으로 복사하여 연산의 기본 데이터 또는 연산의 조건으로 활용하게 됩니다.

프로그램 연산

입력 리프레시 과정에서 읽어 들인 입력 접점의 정보를 조건 또는 데이터로 이용하여 사전에 입력된 프로그램에 따라 연산을 수행하고 그 결과를 내부 메모리 또는 출력 메모리에 저장하게 됩니다. GLOFA-GM PLC에서 프로그램은 크게 스캔 프로그램과 태스크 프로그램의 두 가지로 나눌 수 있는데, 스캔 프로그램이란 PLC의 CPU가 RUN 상태면 무조건 수행하는 프로그램이고, 태스크 프로그램이란 특정 조건을 만족해야만 동작하는 프로그램입니다. 스캔 프로그램 연산을 수행하는 도중에 태스크 프로그램의 실행 조건이 만족되면 스캔 프로그램의 연산을 멈추고, 태스크 프로그램을 수행한 후 태스크 프로그램으로 전이하기 직전에 연산이 수행되던 스캔 프로그램의 위치로 복귀하여 스캔 프로그램의 연산을 계속하게 됩니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

출력 리프레시

스캔 프로그램 및 태스크 프로그램의 연산 도중에 만들어진 결과는 바로 출력으로 보내어지지 않고 출력 이미지 영역에 저장되게 됩니다. 이 과정을 출력 이미지 리프레시라고 합니다.

자기 진단

연산의 과정에서 만들어진 결과는 바로 출력으로 내보내지 않고 출력 이미지 영역에 저장되게 됩니다. 그렇게 하는 이유는 프로그램의 마지막 스텝 연산이 끝나고 나면 PLC 의 CPU는 시스템 상에 오류가 있는지를 검사하고 오류가 없을 때만 출력을 내보내기 때문입니다. 만일 연산이 성공적으로 끝나서 그 결과가 출력 이미지 영역에 저장되었다고 해도 PLC의 CPU는 자기 시스템을 진단하여 시스템 상에 오류가 있다면 출력을 내보내지 않고 에러 메시지를 발생시키게 됩니다. 이것을 자기 진단이라고 합니다.

END 처리

연산이 성공적으로 수행되고 자기 진단 결과 시스템에 오류가 없으면 출력 이미지 영역에 저장된 데이터를 출력 영역으로 복사함으로써 실질적인 출력을 내보내게 됩니다. 이 과정을 END 처리라 하며 END 처리가 끝나면 다시 입력 리프레시를 실시함으로써 PLC 는 반복적인 연산을 수행하게 됩니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

1.4. PLC의 동작 원리

PLC는 사용자의 프로그램에 의하여 본체에 연결된 외부 입출력기기를 제어합니다. 따라서 정확한 동작을 위해서는 입출력기기의 올바른 배선과 프로그램 및 PLC 제어 특성에 대하여 이해해야 합니다.

PLC 프로그래밍 언어

현재 사용중인 프로그래밍 언어로 IL(Instruction List), ST(Structured Text), 래더(Ladder), FBD(Function Block Diagram), SFC(Sequential Function Chart)등이 있습니다. XGI PLC는 IL(Instruction List)과 ST(Structured Text), 래더(Ladder), SFC(Sequential Function Chart)의 4가지 언어를 제공합니다.

▶ IL (Instruction List)

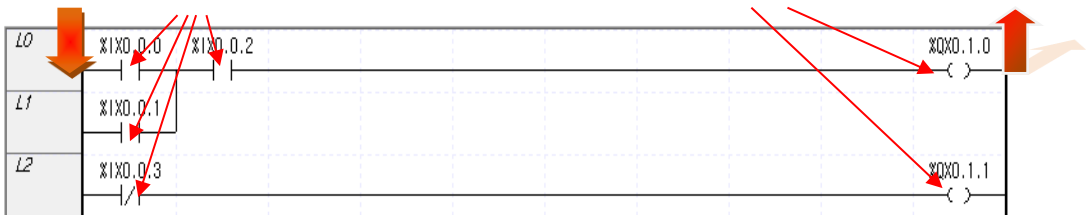
MASTER-K PLC에서 니모닉(Nemonic)이라고 불린 명령어이며, 어셈블리언어 형태의 문자 기반 언어로 MASTER-K에서는 휴대용 프로그램 입력기(Handy Loader)를 이용하여 현장에서 간단한 로직의 프로그래밍에 주로 사용되었습니다만 요즈음에는 노트북이 이를 대체함으로써 굳이 IL로 작성할 필요성이 사라짐으로써 거의 사용되지 않는 명령어 입니다.

번호	명령어	입력 파라미터	변수	변수 설명문
1	LOAD		%IX0.0.0	
2	OR		%IX0.0.1	
3	AND		%IX0.0.2	
4	OUT		%QX0.1.0	
5	LOAD NOT		%IX0.0.3	
6	OUT		%QX0.1.1	

▶ 래더 (Ladder): 사다리도

사다리 형태로 릴레이 로직과 유사한 도형기반의 언어로, 현재 가장 널리 사용되고 있습니다.

+DC 입력회로(스위치) 출력회로(전등) -DC전원

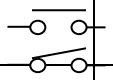
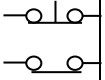
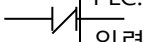
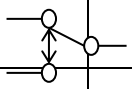
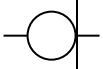
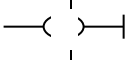
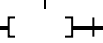


Copyright ⓒ 2021 LS Electric Co., Ltd.

PLC 동작 이해

▶ PLC 기본 약호 (명령어)

릴레이 로직과 유사한 형태의 스위치 형태의 입력과 출력 코일이 있습니다.

구분	릴레이 로직	PLC 로직	내용
A접점			평상시 개방(Open)되어 있는 접점 N.O. (Normally Open) PLC: 외부입력, 내부출력 ON/OFF상태를 입력
B접점			평상시 폐쇄(Closed)되어 있는 접점 N.C. (Normally Closed) PLC: 외부입력, 내부출력 ON/OFF상태의 반전된 상태를 입력
C접점		없음	a, b접점 혼합형으로 PLC에서는 로직의 조합으로 표현
출력 코일			이전까지의 연산 결과 접점 출력
응용 명령	없음		PLC응용 명령을 수행

▶ 기초 용어 정의

점(Point): 입력 8점, 출력 6점의 PLC는 스위치나 센서 등 입력기기를 최대 8개, 램프나 릴레이 등 출력기기를 최대 6개까지 연결할 수 있습니다. PLC의 입출력 용량을 표시할 때 사용합니다.

스텝(Step): PLC명령어의 최소 단위로 A접점, B접점, 출력 코일 등의 명령이 1스텝에 해당하는 명령이고 기타 응용 명령어의 경우 하나의 명령어가 다수의 스텝을 점유합니다.

프로그램 용량 및 CPU속도를 표시하는 단위로 사용됩니다. (용량: 30k step, 속도: sec/step)

스캔타임(Scan Time): 사용자 작성 프로그램의 1회 수행에 걸리는 시간을 의미합니다.

스텝수가 많은 프로그램의 경우 스캔 타임은 증가합니다.

WDT(Watch Dog Timer): 프로그램 연산 폭주나 CPU기능고장에 의하여 출력을 하지 못할 경우 설정한 시간(WDT)대기 후 에러를 발생시키는 시스템 감시 타이머입니다.

기본 200ms로 설정되어 있으며 파라미터 지정에 의해 변경시킬 수 있습니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

파라미터(Parameter): 프로그램과 함께 PLC에 저장되는 운전 데이터로 통신, 시스템 환경 등을 지정합니다.

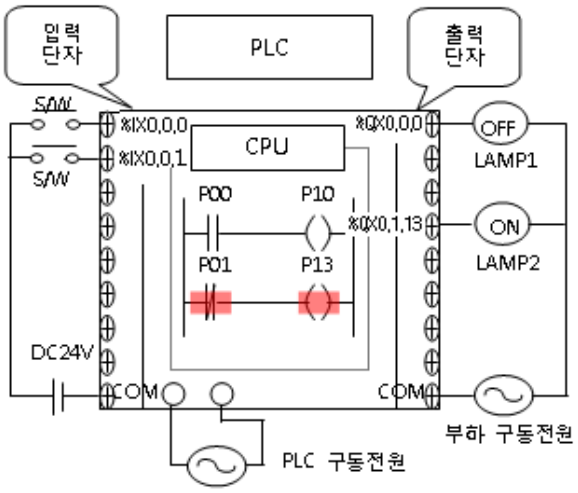
▶ PLC 기본 동작 이해

아래 그림은 PLC 기본 구성을 간략화한 것으로 외부접점과 PLC연산 관계에 대하여 설명합니다. 실선 내부는 PLC의 CPU에 저장되어 동작되는 프로그램으로 프로그램 Loader (XG5000)를 이용하여 입력하면 됩니다.

입력단자와 COM단자 사이에 DC24V를 인가해 주면 입력이 형성 됩니다.

출력단자와 COM단자 사이에 부하(LAMP)를 연결하고 부하구동전원을 연결하면 됩니다.

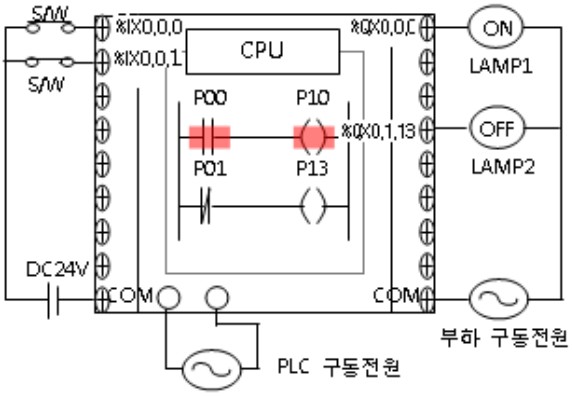
(DC부하일 경우 부하구동전원은 DC전원이 됩니다.)



▶S/W1이 OFF이므로a접점인 %IX0.0.0은 S/W1의 OFF상태를 적용, 단전(Disconnect)되어 출력 %QX0.1.0은 OFF 됩니다.

▶S/W2가 OFF이므로b접점인 %QX0.1.1은 S/W2의 OFF상태반전 적용, 연결(Connect)되어 출력 %QX0.1.13은 ON됩니다.

접점의 연결 및 출력상태를 나타냅니다.
: 전점 단힌 (연결)



▶S/W1이 ON이므로a접점인 %IX0.0.0은 S/W1의 ON상태를 적용, 연결(Connect)되어 출력%QX0.1.0은 ON됩니다.

▶S/W2가 ON이므로b접점인 %QX0.1.1은 S/W2의 ON상태반전 적용, 단전(Disconnect)되어 출력%QX0.1.13은 OFF됩니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

2장. XGI 개요

Copyright ⓒ 2021 LS Electric Co., Ltd.

그 동안 PLC 고객은 메이커(maker)마다 사용 언어와 통신 네트워크가 서로 달라 많은 불편함을 겪어 왔습니다. 이러한 불편함을 해소하고, PLC 고객에게 편리성을 도모하고자 IEC (International Electrotechnical Commission; 국제 전기 표준 회의)에서 PLC 국제 표준화 규격이 제정 되었습니다.

국제 표준화 규격(IEC61131)은 크게 5 Part로 구성되어 있는데

Part 1은 PLC의 기본 기능 및 용어 정의

Part 2는 설비의 요구 기능 및 시험 조건

Part 3은 프로그램 언어

Part 4는 사용자 지침

Part 5는 통신 네트워크로 구성되어 있습니다.

2.1. XGI PLC의 특징

IEC 표준 언어

IEC 언어에서 새로 도입한 가장 중요한 특징들은 다음과 같습니다.

① 다양한 데이터 타입 (type)을 지원합니다.

② 펑션, 펑션블록, 프로그램과 같은 프로그램 구성 요소가 도입되어 상향식 또는 하향식 설계가 가능하며 프로그램을 구조적으로 작성할 수 있습니다.

③ 사용자가 작성한 프로그램을 라이브러리 화하여 다른 프로젝트에서 소프트웨어를 재 사용할 수 있습니다.

④ 다양한 언어를 지원하므로 사용자는 최적의 언어를 선택하여 사용할 수 있습니다.

IEC에서 표준화한 PLC용 언어는 두 개의 도형 기반 언어와 두 개의 문자 기반 언어, 그리고 SFC로 이루어져 있습니다.

▶ 도형식(graphic) 언어

① LD(Ladder Diagram): 릴레이 로직 표현 방식의 언어

② FBD(Function Block Diagram): 블록화한 기능을 서로 연결하여 프로그램을 표현하는 언어

Copyright ⓒ 2021 LS Electric Co., Ltd.

▶ 문자식(Text) 언어

① IL (Instruction List): 어셈블리 언어 형태의 언어

② ST (Structured Text): 파스칼 형식의 고 수준 언어

▶ SFC (Sequential Function Chart): 플로우 차트(Flow Chart)와 유사한 형태로 순차적으로 전개되는 프로그램 전개 방식

국제 규격의 통신 프로토콜

① Open 네트워크를 지향하여 타기종, 멀티 벤더 간의 통신이 가능합니다.

② 상위 네트워크로 Ethernet(10Mbps) 채용

③ 하위 네트워크로 Fieldbus(1Mbps), Device net(500Kbps MAX.), Profibus-DP(12Mbps MAX.) 채용

윈도우 환경의 프로그래밍 Tool (XG5000) 지원

① XG5000(Programming & Debugging Tool)의 윈도우환경 채용으로 프로그램 작성, 수정 시 윈도우 장점을 모두 이용할 수 있습니다.

② MDI(Multiple Document Interface) 지원: 하나의 화면에 각기 다른 언어를 사용하여 동시에 프로그램 작성, 수정 및 모니터링이 가능합니다.

프로그램 작성 용이

① 프로그램의 구조화, 모듈화에 의해 프로그램 작성이 매우 편리합니다.

② 입출력 식별자명을 실제 접속되는 기기명(한글/한자 또는 영문)으로 프로그래밍이 가능합니다.

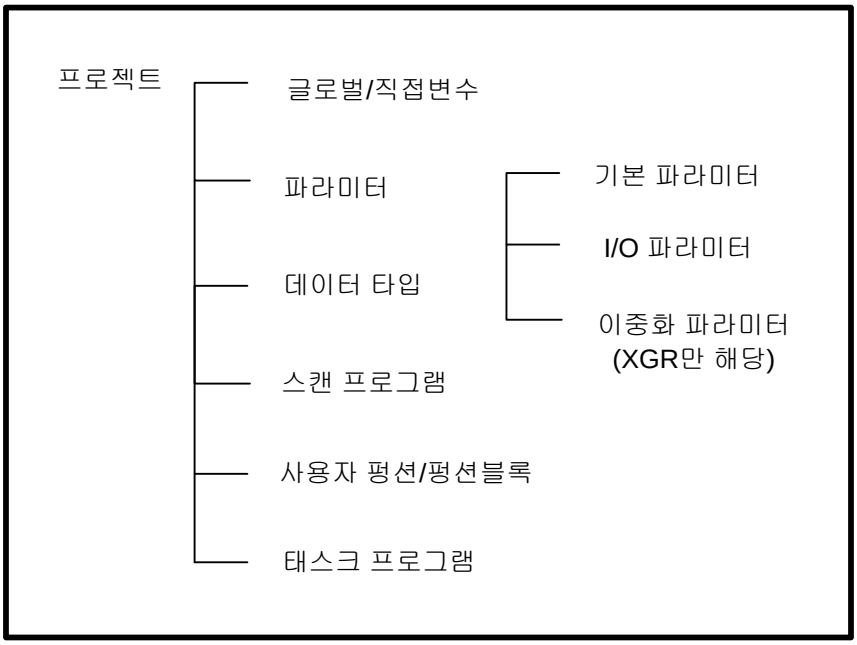
XGK	XGI
P0000	리밋_스위치

Copyright ⓒ 2021 LS Electric Co., Ltd.

2.2. 소프트웨어 구조

개요

PLC 응용 프로그램을 작성하기 전에 소프트웨어 측면에서 전체적인 PLC 시스템을 구성합니다. XGI/XGR/XEC PLC에서는 PLC 시스템 전체를 하나의 프로젝트로 정의합니다. 프로젝트 안에는 하나의 PLC 시스템에 필요한 모든 것이 계층적으로 정의되어 있습니다



프로젝트 (Project)

XGI/XGR/XEC PLC의 프로그램을 작성하기 위해서는 우선 프로젝트를 구성하여야 합니다. 하나의 프로젝트를 구성한다는 것은 하나의 PLC 시스템에 필요한 모든 구성 요소를 작성한다는 의미입니다. 즉, 가장 기본적인 스캔 프로그램(일반적인 PLC 프로그램)뿐만 아니라 기본 파라미터, I/O 파라미터 등을 작성합니다.

글로벌/직접 변수

글로벌 변수 설정 부분, 직접 변수 설명문, 플래그 부분의 탭으로 보여주며 사용자가 필요한 정보를 작성 또는 사용하는 부분입니다.

파라미터

Copyright ⓒ 2021 LS Electric Co., Ltd.

파라미터 부분은 PLC 시스템의 기동 시 필요한 여러 가지 정보를 작성하는 부분입니다.

▶ 기본 파라미터: 기본 파라미터 정보 중 기본 운전, 시간, 출력 제어 설정을 위한 부분과 PLC 전원이 꺼져도 데이터를 보존하는 영역 설정 부분, PLC에 에러가 발생했을 때 동작 방법의 설정을 위한 부분 그리고 MODBUS 정보 설정 부분으로 구성되어 있습니다.

▶ I/O 파라미터: PLC 슬롯에 사용할 I/O 종류를 설정하고, 해당 슬롯 별로 파라미터를 설정합니다.

데이터 타입

데이터는 그 데이터의 고유 성질을 나타내는 데이터 타입을 가지고 있습니다. 예를 들어 ANY_NUM으로 나타내면 LREAL, REAL, LINT, DINT, INT, SINT, ULINT, UDINT, UINT, USINT를 모두 포함합니다.

스캔 프로그램

입력 모듈에서 입력 데이터를 읽은 후 프로그램을 처음부터 끝까지 한번 수행하고, 그 수행 결과를 출력 모듈에 쓰는 일련의 동작을 반복하여 수행하는 응용 프로그램입니다.

사용자 펑션/펑션 블록

▶ 펑션: 펑션은 내부에 상태를 보관하고 있는 데이터를 갖지 않습니다. 즉 입력이 일정하면 출력 값도 일정해야만 펑션이 됩니다.

▶ 펑션 블록: 펑션 블록은 내부에 데이터를 가질 수 있습니다. 펑션 블록은 사용하기 전에 변수를 선언하는 것처럼 인스턴스를 선언하여야 합니다. 인스턴스라는 것은 펑션 블록에서 사용하는 변수들의 집합입니다. 즉, 펑션 블록은 내부에서 사용하는 변수뿐 아니라 출력 값도 펑션 블록 자체에서 보관합니다. 따라서 인스턴스가 보관된 데이터 메모리를 기억하고 있습니다. 프로그램도 펑션 블록의 일종이라고 볼 수 있으며, 프로그램 역시 인스턴스를 선언하여야 합니다. 그러나 프로그램은 펑션 블록과 다르게 프로그램 안이나 펑션 블록 안에서 불러 사용할 수는 없습니다.

태스크 프로그램

태스크 프로그램은 스캔 프로그램처럼 매 스캔 반복처리를 하지 않고, 실행 조건이 발생할 때만 실행을 합니다. 실행해야 할 태스크가 여러 개 대기하고 있는 경우는 우선 순위가 높은 태스크 프로그램부터 처리합니다. 우선 순위가 동일한 태스크가 대기 중일 때는 발생한 순서대로 처리합니다. 태스크 종류는 정주기 태스크와 내부 접점 태스크가 있습니다.

2.3. 공통 요소

Copyright ⓒ 2021 LS Electric Co., Ltd.

XGI/XGR/XEC PLC의 프로그램 구성 요소(프로그램, 펑션, 펑션 블록)는 LD, SFC, ST등 각기 다른 언어로 작성할 수 있습니다. 하지만 그 언어들도 공통적으로 사용하는 문법 요소들을 가지고 있습니다.

표현

▶ 식별자

① 영문자나 밑줄 문자(_)로 시작하는 모든 문자, 숫자, 밑줄 문자 조합이 식별자가 될 수 있습니다.

② 식별자는 변수의 이름으로 쓰입니다.

③ 식별자는 빈 칸(Space)을 포함하지 않아야 합니다.

④ 식별자는 보통 변수 또는 인스턴스 이름인 경우에는 한글, 영문, 한자 모두 제한이 없습니다.

⑤ 영문자의 경우, 대·소문자를 구별하지 않고 모두 대문자로 인식합니다.

종 류	사용 예
대문자와 숫자	IW210, IW215Z, QX75, IDENT
대문자와 숫자, 밑줄 글자	LIM_SW_2, LIMSW5, ABCD, AB_CD
밑줄 글자로 시작하는 대문자와 숫자	_MAIN, _12V7, _ABCD

▶ 데이터의 표현

XGI/XGR/XEC PLC에서 데이터로 사용하는 것은 숫자(Numeric Literals)와 문자열(Character String), 시간 문자(Time Literals) 등입니다.

종 류	사용 예
정 수	-12, 0, 123_456, +986
실 수	-12.0, 0.0, 0.456, 3.14159_26
승수부를 갖는 실수	-1.34E-12, 1.0E+6, 1.234E6
2진수	2#1111_1111, 2#11100000
8진수	8#377(십진수 255) 8#340(십진수 224)
16진수	16#FF(십진수 255) 16#E0(십진수 224)
BOOL 데이터	0, 1, TRUE, FALSE

▶ 숫자 (Numeric Literals)

Copyright ⓒ 2021 LS Electric Co., Ltd.

숫자에는 정수(Integer Literals)와 실수(Real Literals)가 있습니다.

연속되지 않은 밑줄 글자(_)가 숫자 사이에 올 수 있으며 그 의미는 무시됩니다.

십진수는 일반적인 십진 표현법을 따르고 소수점이 있으면 실수로 구별됩니다.

승수(Exponent) 표현 시 +, -의 부호가 올 수 있습니다. 승수부를 구분하는 문자 'E'는 대소문자를 구분하지 않습니다.

승수부가 있는 실수의 사용시 다음은 가능하지 않습니다.

예) 12E-5 (×) 12.0E-5 (○)

정수에는 십진수 이외에 2,8,16진수가 올 수 있으며, 숫자의 앞부분에 진수 #을 사용하여 구분합니다. 아무것도 붙이지 않으면 십진수로 간주합니다.

16진수 표현 시 0 - 9, A - F를 쓰며 소문자 a - f도 쓸 수 있습니다.

16진수 표현 시에는 부호(+, -)가 올 수 없습니다.

BOOL 데이터(Boolean Data)는 정수 0과 1로도 표현할 수 있습니다.

▶ 문자열 (Character String)

작은 따옴표(')로 둘러싸인 모든 문자가 문자열에 해당됩니다.

그 길이는 문자열 상수일 때에는 31자 이내이며, 초기화에 사용할 때 역시 31자로 제한합니다.

예) 'CONVEYER'

▶ 시간 숫자 (Time Literals)

시간 문자는 제어 사건(Control Event)의 경과 시간(Elapsed Time)을 재거나 조절하기 위한 경과 시간(Duration) 데이터와, 제어 사건의 시작점과 끝점의 시각을 표시하기 위한 날짜와 시각(Time Of Day And Date) 데이터로 구분됩니다.

① 경과 시간 (Duration)

경과 시간 데이터는 예약어 'T#'또는 't#'으로 시작합니다.

일(d), 시(h), 분(m), 초(s), ms의 순으로 써야 하고 어느 단위에서 시작되어도 상관없으며, 최소 단위인 ms까지 꼭 쓰지 않아도 되나 중간 단위를 생략할 수는 없습니다.

밑줄 글자(_)는 사용하지 않습니다.

최대 단위에서의 오버플로(Overflow)는 허용되며, 최소 단위에서의 소수점 이하 표현도 ms 이외에는 가능합니다. 단 최대는 T#49d17h2m47s295ms을 초과할 수 없습니다. (즉, ms 단위로 32 비트)

Copyright ⓒ 2021 LS Electric Co., Ltd.

소수점 이하 자릿수의 제한은 현재 초(s)단위에서의 3자리까지입니다.

ms 단위에서는 소수점이 올 수 없습니다.

단위를 나타내는 문자로는 대·소문자 어느 경우나 다 가능합니다.

내 용	사용 예
경과 시간 (Underline 없음)	T#14ms, T#14.7s, T#14.7m, T#14.7h t#14.7d, t#25h15m, t#5d14h12m18s356ms

② 날짜와 시각 (Time Of Day And Date)

날짜와 시각의 표현 방법에는 날짜, 시각, 날짜와 시각의 3가지가 있으며 다음과 같습니다.

내 용	접두 예약어
날짜 접두어	D#
시각 접두어	TOD#
날짜 시각 접두어	DT#

날짜의 시작점은 1984년 1월 1일을 기점으로 합니다. 시각과 날짜 시각의 표현에는 엄격한 자릿수의 제한이 있으며, 초를 나타낼 경우 ms 단위는 소수점 이하 세 자리까지 가능합니다. (1ms 단위)시각과 날짜 시각의 표현 시에는 모든 단위에서 오버플로우(Overflow)가 허용되지 않습니다.

내 용	사용 예
날짜	D#1984-06-25 d#1984-06-25
시각	TOD#15:36:55.36 tod#15:36:55.369
날짜 시각	DT#1984-06-25-15:36:55.36 dt#1984-06-25-15:36:55.369

Copyright ⓒ 2021 LS Electric Co., Ltd.

데이터 타입

데이터는 그 데이터의 고유 성질을 나타내는 데이터 타입을 가지고 있습니다.

▶ 기본 데이터 타입

XGI/XGR/XEC PLC에서는 20종의 기본 데이터 타입을 지원합니다.

▶ 데이터 타입 계층도

XGI/XGR/XEC PLC에서 사용되는 데이터 타입은 크게 ANY_NUM과 ANY_BIT로 구분됩니다.

① ANY_NUM 으로 나타내면 LREAL, REAL, LINT, DINT, INT, SINT, ULINT, UDINT, UINT, USINT 를 모두 포함합니다.

② ANY_BIT로 표현되면 LWORD, DWORD, WORD, BYTE, BOOL중 하나를 사용할 수 있습니다.

▶ 초기값

데이터의 초기값을 지정하지 않으면 자동적으로 아래와 같이 지정됩니다.

데이터 타입	초기값
SINT, INT, DINT, LINT	0
USINT, UINT, UDINT, ULINT	0
BOOL, BYTE, WORD, DWORD, LWORD	0
REAL, LREAL	0.0
TIME	T#0s
DATE	D#1984-01-01
TIME_OF_DAY	TOD#00:00:00
DATE_AND_TIME	DT#1984-01-01-00:00:00
STRING	'' (empty string)

※ VAR_EXTERNAL의 선언은 외부에서 선언한 변수를 간접 지정한 것이므로 초기값을 줄 수 없습니다.

※ 변수 선언 시 %I와 %Q로 할당한 변수는 입출력에 해당하므로 초기값을 줄 수 없습니다

Copyright ⓒ 2021 LS Electric Co., Ltd.

변수

변수란 프로그램 안에서 사용하는 데이터로서 값을 가지고 있습니다. 변수는 PLC의 입력이나 출력, 내부 메모리 등과 같이 변할 수 있는 대상을 가리킵니다.

▶ 변수의 표현

변수의 표현에는 2가지가 있습니다.

① 식별자에 의한 변수: 식별자에 의해 변수에 이름을 부여하는 것

② 직접변수: PLC의 입·출력 또는 기억 장소에 대하여 직접적으로 표현하는 것

식별자에 의한 변수는 다른 변수들과 구별하기 위하여 그 이름의 변수가 선언된 프로그램 안에서 유일해야 합니다.

직접 변수의 표현은 퍼센트 문자(%)를 시작으로 위치를 나타내는 접두어와 데이터의 크기를 나타내는 접두어 그리고 마침표로 분리되는 하나 이상의 부호 없는 정수의 순으로 표현할 수 있습니다. 그 접두어들은 다음에 나타나 있습니다.

위치 접두어

번호	접두어	의미
1	I	입력 위치(Input Location)
2	Q	출력 위치(Output Location)
3	M	내부 메모리 중 M영역 위치(Memory Location)
4	R	내부 메모리 중 R영역 위치(Memory Location)
5	W	내부 메모리 중 W영역 위치(Memory Location)

크기 접두어

번 호	접두어	의 미
1	X	1 비트의 크기
2	None	1 비트의 크기
3	B	1 바이트(8 비트)의 크기
4	W	1 워드(16 비트)의 크기
5	D	1 더블 워드(32 비트)의 크기
6	L	1 롱 워드(64 비트)의 크기

Copyright ⓒ 2021 LS Electric Co., Ltd.

표현 형식

%[위치 접두어][크기 접두어] n1.n2.n3

번 호	I, Q	M, R, W
n1	베이스 번호(0부터 시작)	[크기 접두어]에 따른 n1번째 데이터 (0부터 시작)
n2	슬롯 번호(0부터 시작)	n1번째 데이터상의 n2번째 비트 (0부터 시작): 생략 가능
n3	[크기 접두어]에 따른 n3번째 데이터 (0부터 시작)	사용하지 않음

예시)

%QX3.1.4 또는 %Q3.1.4 3번 베이스의 1번 슬롯의 4번 출력(1 비트)

%IW2.4.1 2번 베이스의 4번 슬롯의 워드 단위로 1번 입력(16비트)

%MD48 48의 위치에 있는 더블 워드 단위의 메모리

%MW40.3 40의 위치에 있는 워드 단위의 메모리 중 3번 비트

(내부 메모리는 베이스, 슬롯 등의 개념이 없음)

※ 접두어로는 소문자가 올 수 없습니다.

※ 크기 접두어를 붙이지 않으면 그 변수는 1비트로 처리합니다.

※ 직접변수는 선언하지 않고 사용할 수 있습니다.

▶ 변수의 선언

프로그램 구성 요소(즉 프로그램, 펑션, 펑션 블록)는 그 구성 요소에서 사용할 변수를 선언할 수 있는 선언 부분을 가지고 있습니다.

프로그램 구성 요소에서 변수를 사용하기 위해서는 우선 사용할 변수를 선언해야 합니다.

변수의 선언에서 설정해야 할 사항은 다음과 같습니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

① 변수 종류: 변수를 어떻게 선언할 것인가를 설정합니다.

변수종류	내 용
VAR	읽고 쓸 수 있는 일반적인 변수
VAR_RETAIN	정전 유지 변수
VAR_CONSTANT	읽기만 할 수 있는 변수
VAR_EXTERNAL	VAR_GLOBAL로 선언된 변수를 사용하기 위한 선언

② 데이터 타입: 변수의 데이터 타입을 지정합니다.

③ 메모리 할당: 변수가 차지할 메모리를 할당합니다.

자동 (블랭크): 컴파일러가 변수의 위치를 자동으로 지정 (내부 데이터)

사용자 정의: 사용자가 직접변수 (%I 또는 %Q)를 사용하여 강제로 위치를 지정

(참고)

자동 배치 변수는 그 실제 위치가 고정되어 있지 않습니다. 예를 들어 VAL1이란 변수를 BOOL 데이터타입으로 선언하였다면 그 변수가 내부 데이터 영역의 어느 위치에 있는지 고정되어 있지 않다는 것입니다. 그 위치는 프로그램을 다 작성한 후 컴파일러와 링커에 의해 정해집니다. 만약 프로그램을 수정한 후에 다시 컴파일 하였다면 그 위치가 변할 수 있습니다. 자동 배치 변수의 장점은 사용자가 내부 변수로 사용하는 것들의 위치에 신경 쓰지 않아도 된다는 것입니다. 다른 이름으로 선언한 변수들은 결코 데이터 메모리에 중복되어 위치하지 않기 때문입니다. 직접변수는 변수의 위치가 정해지기 때문에 %I와 %Q를 제외하고는 될 수 있으면 사용하지 않는 것이 좋습니다. 직접변수는 자동 배치 변수가 아니므로 사용자가 잘못 사용할 경우, 중복될 수 있습니다.

④ 초기값 (Initial Value) 지정: 변수의 초기값을 지정합니다. 지정하지 않으면 2.3.2.3. 항의 초기값으로 지정됩니다.

(참고)

VAR_EXTERNAL의 선언 시에는 초기값을 줄 수 없습니다.

변수 선언 시 %I와 %Q로 강제 할당한 변수에는 초기값을 줄 수 없습니다.

⑤ PLC의 전원이 끊긴 후에도 데이터의 값을 유지할 필요가 있는 변수는 정전 유지(Retention)의 기능이 제공되는 VAR_RETAIN을 써서 선언할 수 있으며 다음의 규칙을 따릅니다. -

정전 유지 변수는 시스템의 웜 리스타트 시 그 값이 유지됩니다.

시스템의 콜드 리스타트 시에는 사용자가 정의한 초기값이나 기본 초기값으로 초기화됩니다

.

Copyright ⓒ 2021 LS Electric Co., Ltd.

⑥ VAR_RETAIN으로 선언되지 않은 변수는 콜드 리스타트나 웜 리스타트 어느 경우에도 사용자가 정의한 초기값이나 기본 초기값으로 초기화됩니다.

(참고)
변수 선언 시 %I와 %Q로 강제 할당한 변수는 변수종류를 VAR_RETAIN, VAR_CONSTANT로 선언할 수 없습니다.

⑦ 변수는 기본 데이터 타입을 인자로 갖는 어레이로 선언하여 사용할 수 있습니다. 어레이 변수로 선언할 때에는 인자로 사용할 데이터의 타입과 어레이의 크기를 설정하여야 합니다. 단, 기본 데이터 타입 중에 STRING 데이터 타입은 인자로 설정할 수 없습니다.

⑧ 변수 선언의 유효 영역(Scope), 즉 변수를 사용할 수 있는 영역은 그 변수가 선언된 프로그램 구성 요소에 한합니다. 따라서 다른 프로그램 구성 요소에서 선언된 변수는 사용할 수 없습니다. 글로벌 변수로 선언된 변수는 이와 달리 모든 곳에서 VAR_EXTERNAL 선언에 의해 변수 접근이 가능합니다.

변수의 선언 예

변수 이름	변수형	데이터 타입	초기값	메모리 할당
I_VAL	VAR	INT	1234	자동
BIPOLAR	VAR_RETAIN	REAL	-	자동
LIMIT_SW	VAR	BOOL	-	%IX1.0.2
GLO_SW	VAR_EXTERNAL	DWORD	-	자동
READ_BUF	VAR	ARRAY OF INT[10]	-	자동

▶ 예약 변수

예약 변수는 시스템에서 미리 선언한 변수들로서 플래그로 사용됩니다. 사용자가 이 변수 이름으로 변수 선언을 할 수는 없습니다.

이 예약 변수를 사용할 때에는 변수 선언 없이 사용합니다.

① 예약어

예약어는 시스템에서 사용하기 위해 미리 정의한 단어들입니다. 따라서 식별자로 이 예약어를 사용할 수는 없습니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

프로그램 종류

프로그램 종류로는 사용자 펑션, 사용자 펑션 블록, 프로그램이 있습니다.

프로그램에서 자기 자신의 프로그램을 호출할 수는 없습니다. (재귀 호출 금지)

▶ 사용자 펑션

사용자 펑션은 내부에 상태를 보관하고 있는 데이터를 갖지 않습니다. 즉 입력이 일정하면 출력 값도 일정해야만 사용자 펑션이 됩니다.

사용자 펑션의 내부 변수는 초기값을 가질 수 없습니다.

사용자 펑션은 변수를 VAR_EXTERNAL로 선언하여 사용할 수 없습니다.

사용자 펑션 안에서는 직접변수들을 사용할 수 없습니다.

사용자 펑션은 프로그램 구성 요소에서 호출하여 사용합니다.

사용자 펑션을 호출하는 프로그램 구성 요소에서 사용자 펑션으로의 데이터 전달은 입력을 통하여 실행합니다.

사용자 펑션 안에서는 사용자 펑션 블록이나 프로그램을 호출할 수 없습니다.

사용자 펑션은 그 사용자 펑션 이름과 하나의 출력 데이터 타입 변수 이름이 같습니다. 이 변수는 하나의 사용자 펑션을 만들 때 자동적으로 생성되며 사용자 펑션에서 결과값을 이 변수에 넣어서 출력시킵니다.

▶ 사용자 펑션 블록

사용자 펑션 블록은 출력이 여러 개가 될 수 있습니다.

사용자 펑션 블록은 내부에 데이터를 가질 수 있습니다. 사용자 펑션 블록은 사용하기 전에 변수를 선언하는 것처럼 인스턴스를 선언하여야 합니다. 인스턴스라는 것은 사용자 펑션 블록에서 사용하는 변수들의 집합입니다. 사용자 펑션 블록은 내부에서 사용하는 변수뿐 아니라 출력 값도 자체에서 보관하여야 하므로 데이터 메모리를 가지고 있어야 하는데 바로 그것이 인스턴스입니다. 프로그램도 사용자 펑션 블록의 일종이라고 볼 수 있으며, 프로그램 역시 인스턴스를 선언하여야 합니다.
사용자 펑션 블록 안에서는 직접 변수를 사용할 수 있습니다. 또한 글로벌 변수로 선언되고 사용자 정의(AT)로 강제 배치된 직접변수는 VAR_EXTERNAL로 선언하여 사용할 수 있습니다.
사용자 펑션 블록 안에서는 프로그램을 호출할 수 없습니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

▶ 프로그램

프로그램은 사용자 펑션 블록과 같이 인스턴스를 선언하여 사용합니다.

프로그램 안에서는 직접 변수를 사용할 수 있습니다.

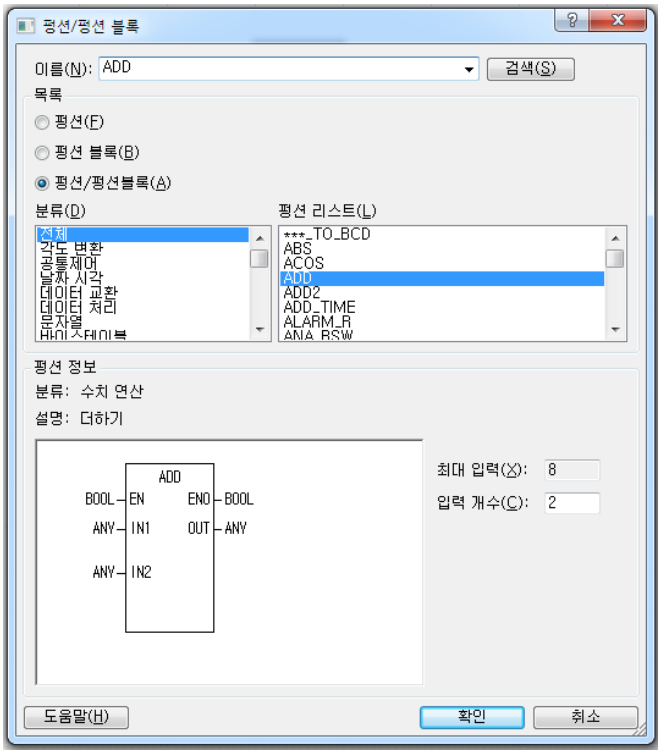
프로그램에는 입/출력변수가 없습니다.

프로그램에는 사용자 펑션 및 사용자 펑션 블록을 호출할 수 있습니다.

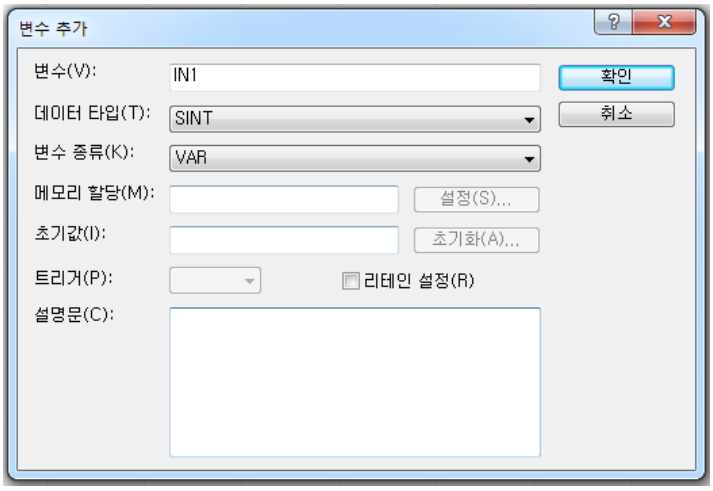
명령어 선정

▶ 내부적으로 결정되는 명령어

펑션의 경우 하나의 이름을 갖지만 여러 종류의 변수타입을 입력할 수 있는 명령어는 사용되는 변수에 따라서 내부적으로 여러 명령어로 구분됩니다. 예를 들어 ADD의 경우, 설정한 입력개수 및 입력/출력 변수 타입에 따라서 여러 종류로 구분되어 처리됩니다. 아래와 같이 선택한 경우에는 래더 프로그램 상에서 보이는 명령어는 ADD이지만, 내부적으로 ADD2_SINT 명령어를 수행하게 되는 것입니다.

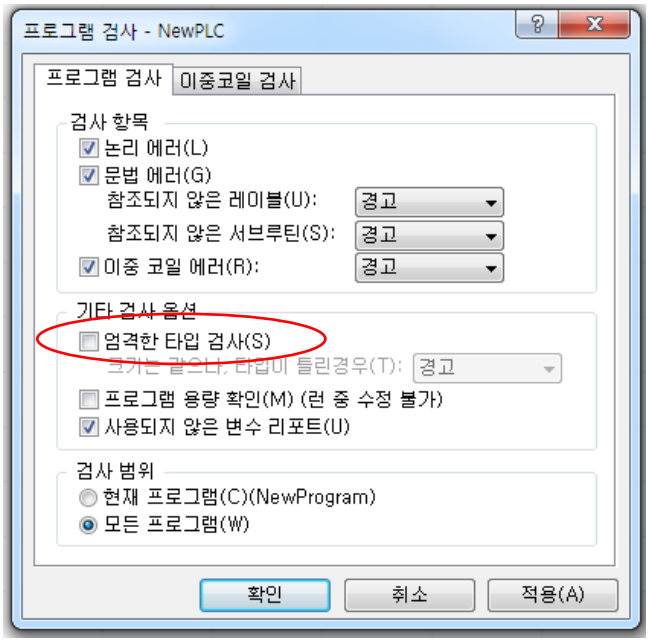


Copyright ⓒ 2021 LS Electric Co., Ltd.



내부적으로 사용되는 명령어의 선정은 사용자가 선택한 변수 타입에 따라 XG5000에서 자동 선정하게 됩니다. 예를 들어 ADD명령 중 입력개수 2개를 선택하고, 입력과 출력변수를 모두 DINT로 선택하면 위에서 설명한 것처럼 ADD2_DINT선정됩니다.

IEC에서는 같은 타입끼리의 연산만을 허용하고 있으나, XG5000에서는 검사옵션에 엄격한 검사 옵션을 두어 변수의 크기(BYTE, WORD, DWORD, LWORD)만 같으면 연산을 허용하는 옵션이 있습니다.



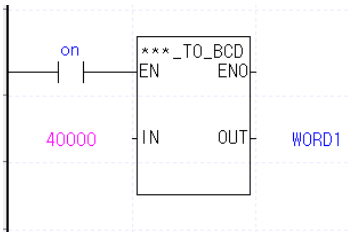
▶ 명령어 선정 규칙

사용된 변수가 한가지 타입이 아니고, 크기가 같은 여러 타입일 경우, 입력 변수의 타입이 우선 반영됩니다. 만약, 입력변수에 모두 상수만 사용되었을 경우, 출력변수에 선언된 타입에 따라 내부적으로 사용되는 명령어가 결정됩니다.

입력변수의 타입은 여러 가지가 올 수 있고 출력변수의 타입은 한가지만 올 수 있는 명령어에서, 입력에 상수를 사용하게 되면 XG5000에서는 상수의 값에 따라 명령어를 결정합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

(예시) ***_TO_BCD로 예를 들어 보면,



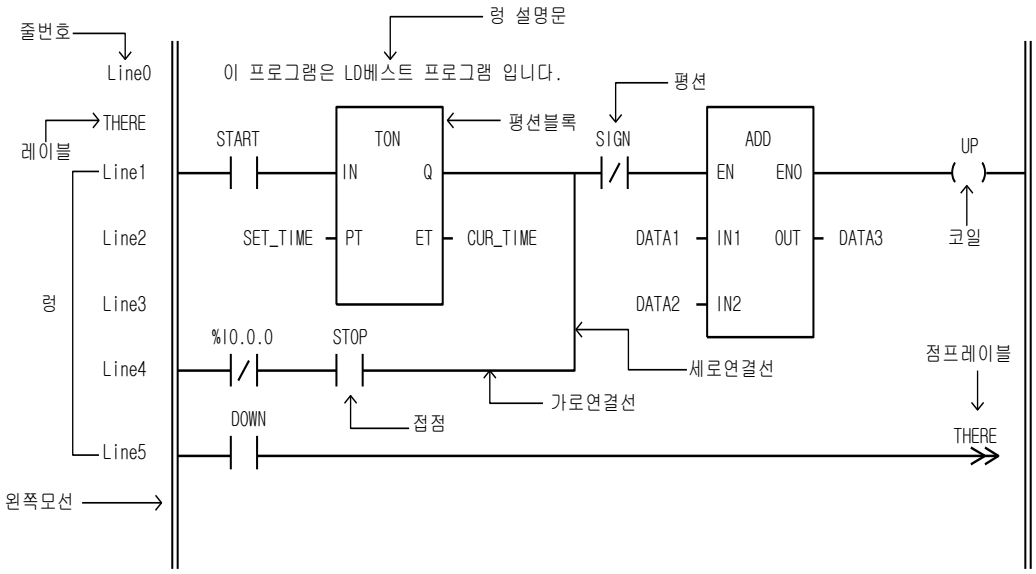
위와 같이 사용했을 경우, 입력변수가 상수이므로 출력변수의 타입을 보고 상세 명령어를 결정하게 되는데 이 명령어는 출력이 WORD인 명령어가INT_TO_BCD_WORD/UINT_TO_BCD_WORD 두 개 존재합니다. 이 경우에는 상수의 타입에 따라 UINT_TO_BCD_WORD 가 선정됩니다. 상수 사용시 양의 값이면 무조건 부호 없는 수(Unsigned)로 판단하고 음수일 경우에는 부호 있는 수(Signed)로 판단을 합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

2.4. LD (Ladder Diagram)

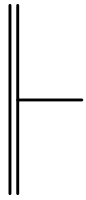
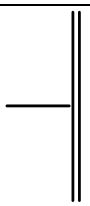
개요

LD 프로그램은 릴레이 로직 다이어그램에서 많이 사용하는 코일이나 접점 등의 그래픽 기호를 통하여 PLC의 프로그램을 표현하는 것입니다.



모선

LD 그래픽 구성도의 왼쪽 끝과 오른쪽 끝에는 전원선 개념의 모선이 세로로 양쪽에 놓여 있게 됩니다.

No	기호	이름	설명
1		왼쪽 모선	언제나 BOOL 1의 값을 가지고 있습니다.
2		오른쪽 모선	값은 정해지지 않습니다

Copyright ⓒ 2021 LS Electric Co., Ltd.

연결선

왼쪽 모선의 BOOL 1 값은 작성한 도면에 따라 오른쪽으로 전달됩니다. 그 전달되는 값을 가진 선을 전원 흐름선 또는 연결선이라고 하며, 접점이나 코일에 연결되어 있는 선입니다. 전원 흐름선은 언제나 BOOL 값을 가지고 있으며, 한 렁(Rung)에서 하나만 존재합니다. 여기서 렁이란 LD의 처음부터 밑으로 내려가는 선이 없는 줄까지를 말합니다.

LD의 각 요소를 연결하는 연결선에는 가로 연결선과 세로 연결선이 있습니다.

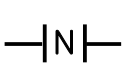
No.	기호	이름	설명
1	————	가로연결선	왼쪽의 값을 오른쪽으로 전달
2	|	세로연결선	왼쪽에 있는 가로 연결선들의 논리합

접점

접점은 왼쪽에 있는 가로연결선의 상태와 현 접점과 연관된 BOOL 입력, 출력, 또는 메모리 변수 간의 논리곱(Boolean AND)을 한 값을 오른쪽에 위치한 가로 연결선에 전달합니다. 접점과 관련된 변수 값 자체는 변화시키지 않습니다. 표준 접점 기호는 다음 표와 같습니다.

정적 접점			
No	기호	이름	설명
1	*** ┤ ├	평상시 열린 접점 (Normally Open Contact)	BOOL 변수("***" 로 표시된 것)의 상태가 On일 때에는 왼쪽의 연결선 상태는 오른쪽의 연결선으로 복사됩니다. 그렇지 않을 경우에는 오른쪽의 연결선 상 태가 Off입니다.
2	*** ┤/├	평상시 닫힌 접점 (Normally Closed Contact)	BOOL 변수("***" 로 표시된 것)의 상태가 Off일 때에는 왼쪽의 연결선 상태는 오른쪽의 연결선으로 복사됩니다. 그렇지 않을 경우에는 오른쪽의 연결선 상태가 Off입니다.

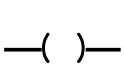
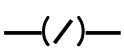
Copyright ⓒ 2021 LS Electric Co., Ltd.

상태 변환 검출 접점			
No	기호	이름	설명
3	***	양 변환 검출 접점 (Positive Transition-Sensing Contact)	BOOL 변수("***" 로 표시된 것)의 값이 전 스캔에서 Off였던 것이 현재 스캔에서 On으로 되고, 왼쪽 연결선 상태가 On되어 있는 경우에 한해서 오른쪽의 연결선 상태는 현재 스캔 동안에 On이 됩니다.
4	*** 	음 변환 검출 접점 (Negative Transition-Sensing Contact)	BOOL 변수("***" 로 표시된 것)의 값이 전 스캔에서 On이었던 것이 현재 스캔에서 Off되고 왼쪽 연결선 상태가 On되어 있는 경우에 한해서 오른쪽의 연결선 상태는 현재 스캔 동안에 On이 됩니다.

코일

코일은 왼쪽의 연결선의 상태 또는 상태 변환에 대한 처리 결과를 연관된 BOOL 변수에 저장시킵니다. 표준 코일 기호는 다음 표와 같습니다.

코일은 LD의 가장 오른쪽에만 올 수 있습니다. 즉 코일의 우측에는 언제나 오른쪽 모선만 있습니다.

임시 코일(Momentary Coils)			
No.	기호	이름	설명
1	*** 	코일(Coil)	왼쪽에 있는 연결선의 상태를 관련된 BOOL 변수("***" 로 표시된 것)에 넣습니다.
2	*** 	역코일 (Negated Coil)	왼쪽에 있는 연결선 상태의 역(Negated)값을 관련된 BOOL 변수("***" 로 표시된 것)에 넣습니다. 즉, 왼쪽 연결선 상태 Off 이면 관련된 변수를 On 시키고, 왼쪽 연결선 상태가 On 이면 관련된 변수를 Off 시킵니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

래치 코일(Latched Coils)			
No.	기호	이름	설명
3	*** —(S)—	Set(Latch) Coil	왼쪽의 연결선 상태가 On이 되었을 때에는 관련된 BOOL 변수("***" 로 표시 된 것)는 On이 되고 Reset 코일에 의해 Off되기 전까지는 On 되어 있는 상태로 유지됩니다.
4	*** —(R)—	Reset(Unlatch) Coil	왼쪽의 연결선 상태가 On이 되었을 때에는 관련된 BOOL 변수("***" 로 표시 된 것)는 Off되고 Set 코일에 의해 On되기 전까지는 Off되어 있는 상태로 유지됩니다.

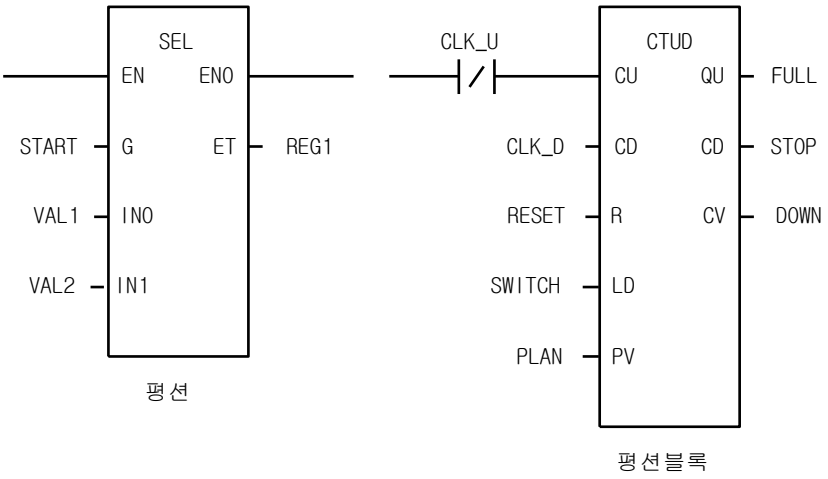
상태 변환 검출 코일(Transition-Sensing Coils)			
No.	기호	이름	설명
5	*** —(P)—	양 변환 검출 코일 (Positive Transition-Sensing Coil)	왼쪽 연결선 상태가 바로 전 스캔에서 Off였던 것이 현재 스캔에서 On이 되어 있는 경우에 관련된 BOOL 변수("***" 로 표시된 것)의 값은 현재 스캔 동안만 On이 됩니다.
6	*** —(N)—	음 변환 검출 코일 (Negative Transition-Sensing Coil)	왼쪽 연결선 상태가 바로 전 스캔에서 On이었던 것이 현재 스캔에서 Off되어 있는 경우에 관련된 BOOL 변수("***" 로 표시된 것)의 값은 현재 스캔 동안만 On이 됩니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

펑션과 펑션 블록의 호출

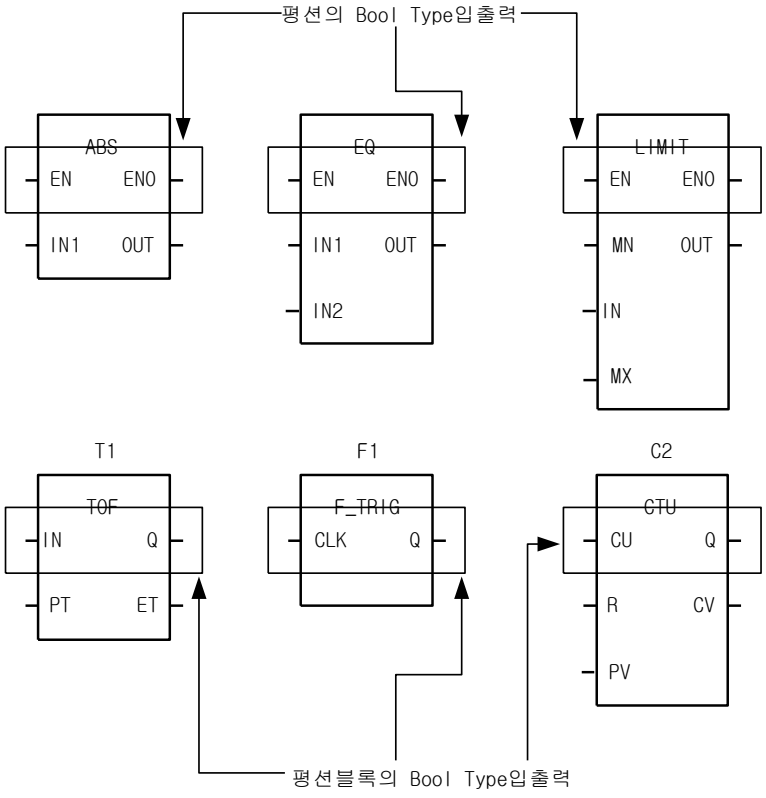
펑션과 펑션 블록에 대한 실제적인 입출력 연결은 입출력 표시가 있는 블록 외부에 적절한 데이터 또는 변수를 기입함으로써 이루어집니다.

(예)



펑션이나 펑션 블록 내부로의 전원 흐름(Power Flow)을 허용하기 위해서는 적어도 한 개의 BOOL 타입 입력과 BOOL 타입 출력이 각 펑션이나 펑션 블록마다 존재해야 합니다. 펑션에서는 EN과 ENO가 BOOL 타입 입출력이며, 펑션 블록에서는 첫 번째 입력과 첫 번째 출력의 데이터 타입이 BOOL입니다.

(예)

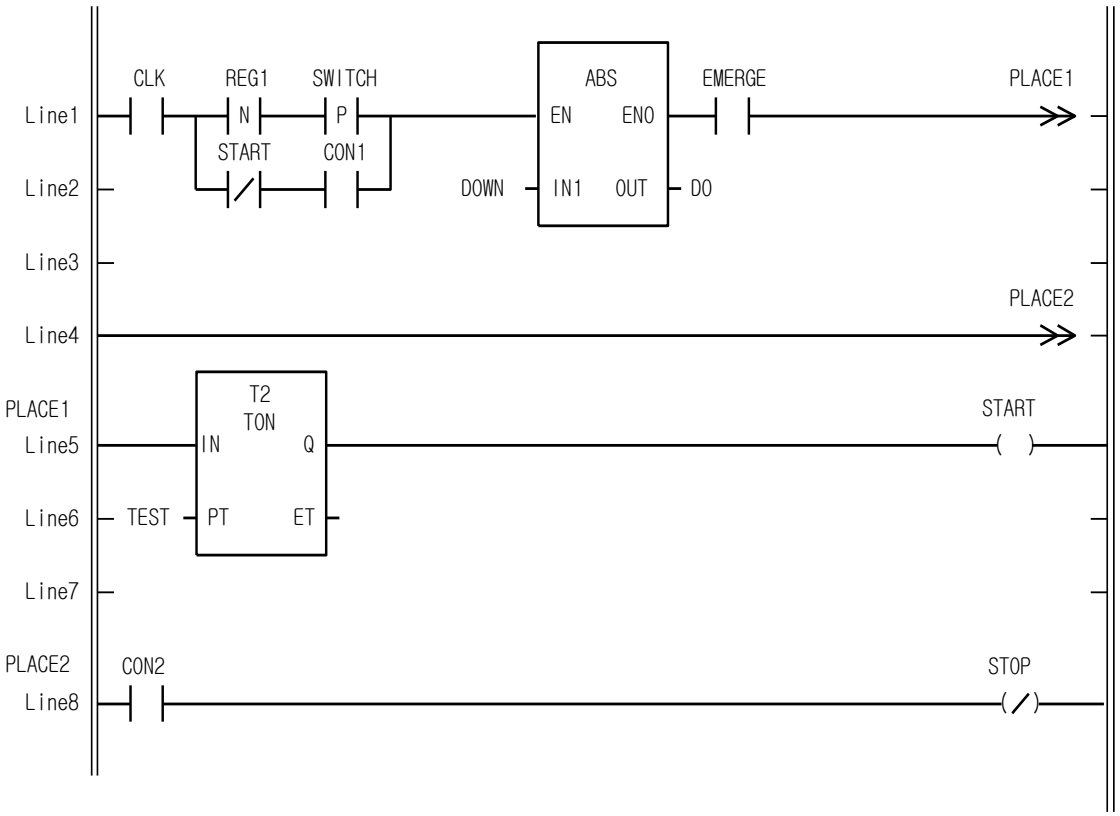


Copyright ⓒ 2021 LS Electric Co., Ltd.

LD에서의 펑션은 EN이라는 입력과 ENO라는 출력이 붙습니다. 두 개 다 BOOL 데이터 타입을 가지고 있으며, EN 입력 값이 BOOL 1이면 그 펑션을 수행하고, BOOL 0이면 그 펑션을 수행하지 않습니다. ENO 출력은 보통 EN값이 그대로 나오지만 그 펑션의 수행 시 에러가 발생하면 EN값이 BOOL 1이라도 ENO 값은 BOOL 0이 나옵니다. 펑션의 EN은 언제나 전원 흐름선이 되어야 하지만 ENO는 꼭 전원 흐름선이 될 필요는 없습니다. 하지만 ENO가 아닌 펑션 출력에 전원 흐름선을 연결할 때에는 그 출력의 데이터 타입이 반드시 BOOL이어야 합니다. 또한 ENO가 아닌 펑션 출력에 전원 흐름선을 연결할 때에는 ENO 출력에는 아무것도 연결하면 안됩니다. 펑션의 모든 입력은 펑션의 왼쪽에 그 값을 기입함으로써 지정되는데 빠짐없이 지정하여야 합니다. 펑션의 출력 값은 펑션의 오른쪽에 지정한 변수에 보관됩니다.

LD 펑션 블록의 입력도 펑션과 같은 방법으로 지정합니다. 펑션 블록의 출력은 그 인스턴스 안에 저장되어 있으므로 변수를 지정하지 않더라도 상관없습니다. 펑션 블록에는 EN, ENO 입출력이 없으므로 펑션 블록을 만나면 항시 수행합니다. 따라서 어떤 로직 결과에 따라 펑션 블록의 수행 여부를 결정하기 위해서는 점프(-->>)를 사용하여야 합니다. 펑션 블록에 전원 흐름선을 연결할 때는 역시 데이터 타입이 BOOL인 입출력에 연결하여야 합니다.

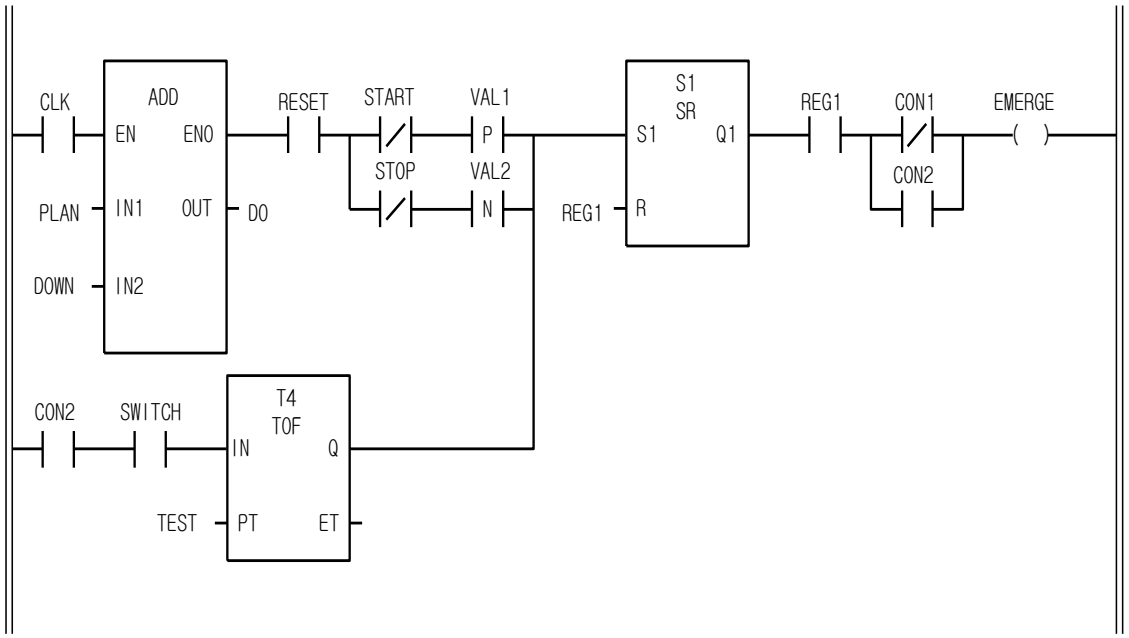
(예)



LD에서 펑션과 펑션 블록은 어느 곳에라도 올 수 있습니다. 펑션과 펑션 블록의 출력에 전원 흐름선을 연결하고 거기에 접점 등을 연결하여 로직 연산을 계속할 수도 있습니다.

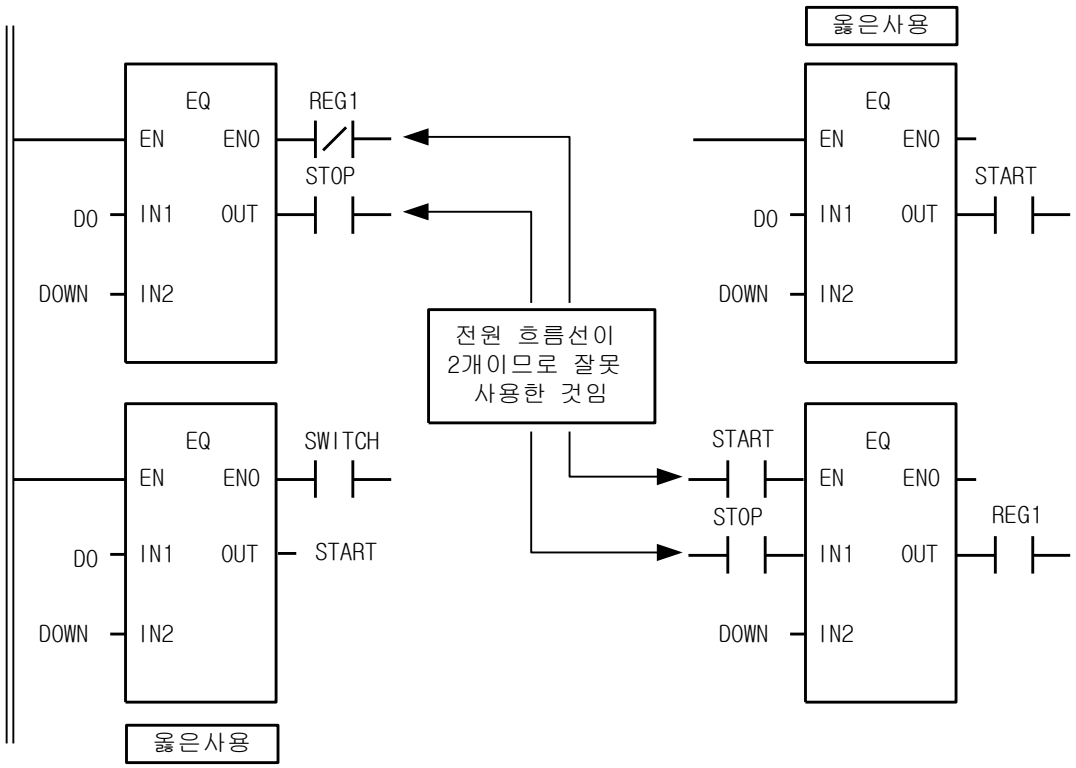
Copyright ⓒ 2021 LS Electric Co., Ltd.

(예)



하나의 펑션이나 펑션 블록에 연결할 수 있는 전원 흐름선은 단 하나입니다.

(예)



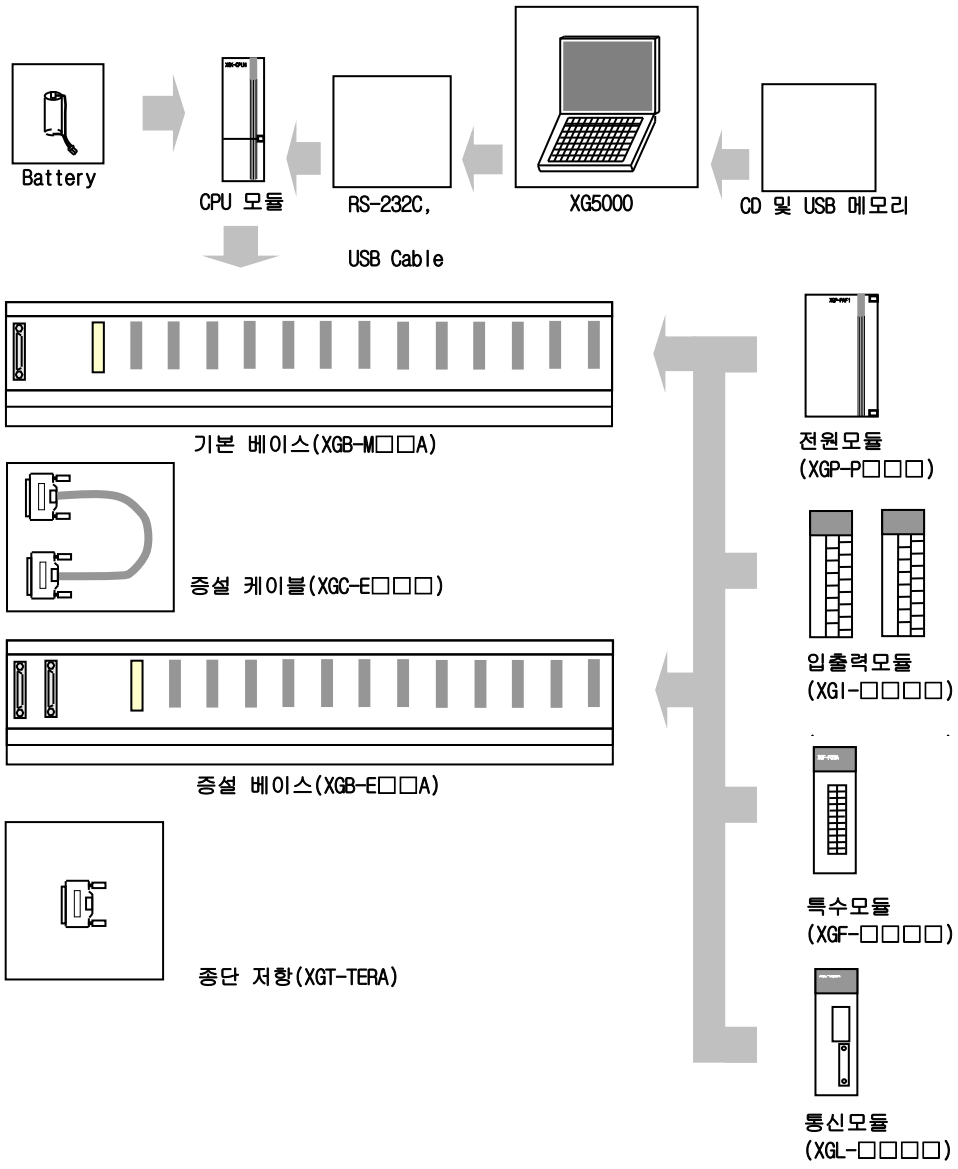
Copyright ⓒ 2021 LS Electric Co., Ltd.

2.5. 시스템 구성

XGI시리즈는 기본 시스템, 컴퓨터 링크 및 네트워크 시스템 구성에 적합한 각종 제품을 구비하고 있습니다. 본 장은 각 시스템의 구성 방법 및 특징에 대해 설명합니다.

XGI 시리즈 시스템 구성

XGI 시리즈의 시스템 구성은 아래 그림과 같습니다.



Copyright ⓒ 2021 LS Electric Co., Ltd.

기본 시스템 구성 시 모듈 선정

기본 시스템 구성 시 각 모듈 별 데이터 리프레시 영역의 크기를 고려해야 합니다. 데이터 리프레시 영역은 XGK/XGI CPU 시스템에서 CPU와 장착된 모듈간 데이터 입출력을 위해 사용되는 영역입니다. 각 모듈 별 사용하는 데이터 리프레시 영역의 크기는 채널의 운전 및 입/출력 동작과 관계 없이 아래와 같은 크기가 입출력 영역으로 사용됩니다. 따라서 XGK/XGI CPU 시스템 구성 시 최대 데이터 리프레시 영역 크기인 1,024 WORD 이내로 사용해야 하며, 이를 초과하여 사용 시 시스템이 정상 동작하지 않습니다.

▶ 모듈 별 데이터 리프레시 영역 크기

품 명	형 명	리프레시크기	품 명	형 명	리프레시크기
디지털 입력 모듈	XGI-A12A	1	디지털 출력 모듈	XGQ-RY1A	1
	XGI-A21A	1		XGQ-RY2A	1
	XGI-A21C	1		XGQ-RY2B	1
	XGI-D21A	1		XGQ-SS2A	1
	XGI-D22A/B	1		XGQ-TR1C	1
	XGI-D24A/B	2		XGQ-TR2A/B	2
	XGI-D28A/B	4		XGQ-TR4A/B	4
디지털 입출력 모듈	XGH-DT4A	2		XGQ-TR8A/B	8
아날로그 입력 모듈	XGF-AC8A	22	온도 입력 모듈	XGF-RD4A	30
	XGF-AV8A	22		XGF-RD4S	30
	XGF-AD8A	22		XGF-TC4S	30
	XGF-AD16A	21		XGF-RD8A	23
	XGF-AD4S	12	온도 제어 모듈	XGF-TC4RT	31
	XGF-AW4S	12		XGF-TC4UD	31
	XGF-AC4H	11	고속카운터 모듈	XGF-HO2A	25
아날로그 출력 모듈	XGF-DC8A	11		XGF-HD2A	25
	XGF-DV8A	11		XGF-HO8A	25
	XGF-DC4A	11	SOE 모듈	XGF-SOEA	2
	XGF-DV4A	11	데이터 로그 모듈	XGF-DL16A	32
	XGF-DC4S	11	통신 모듈	XGL-EFMT(B)	16
	XGF-DV4S	11		XGL-EFMF	16
	XGF-DC4H	7		XGL-ESHF	16
아날로그 입/출력 모듈	XGF-AH6A	11		XGL-DMEA(B)	16
위치제어 모듈	XGF-PO1A	2		XGL-PSEA	16
	XGF-PO2A	2		XGL-PMEA(B)	16
	XGF-PO3A	2		XGL-PMEC(B)	16
	XGF-PD1A	2		XGL-EDMT	16
	XGF-PD2A	2		XGL-EDMF	16
	XGF-PD3A	2		XGL-EDST	16
	XGF-PO1H	2		XGL-EDSF	16
	XGF-PO2H	2		XGL-RMEA	16

▶ 데이터 리프레시 영역 크기 계산

Copyright ⓒ 2021 LS Electric Co., Ltd.

① 시스템에 장착 된 데이터 리프레시 영역 크기 제한

시스템에 장착 된 모든 모듈의 데이터 리프레시 영역 크기의 총합 (WORD) ≤ 1,024 WORD

② 예제

디지털 입력 64점(20대), 디지털 출력 32점(10대), XGF-AC8A(20대), XGF-RD4A(10대) 장착된 경우의 크기 계산

➔ (4 * 20) + (2 * 10) + (22 * 20) + (30 * 10) = 840 WORD ≤ 1,024 WORD

알아두기

1) XGK/XGI 모듈 장착 대수는 데이터 리프레시 영역 크기의 합이 1,204 WORD 를 초과하지 않아야 합니다.

2) XGK/I 시스템에서 데이터 리프레시 영역이 최대 범위를 초과 사용하여 운전할 경우 시스템이 정상 동작하지 않습니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

2.6. CPU 모듈

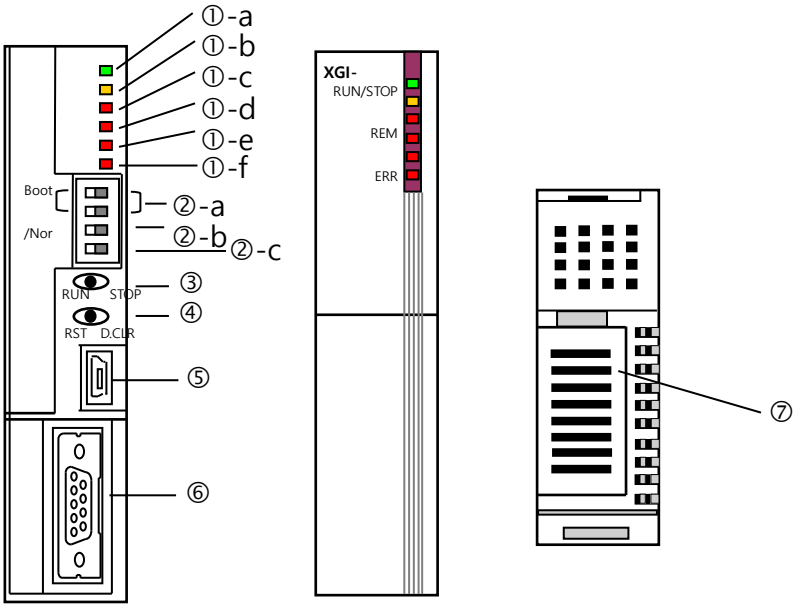
CPU 모듈 (XGI CPU)의 성능 규격은 다음과 같습니다.

항 목			제 품					비 고
			XGI-CPUUN	XGI-CPUU	XGI-CPUH	XGI-CPUS	XGI-CPUE	
연산 방식			반복연산, 정주기 연산, 고정주기 스캔					
입출력 제어 방식			스캔동기 일괄처리 방식 (리프레시 방식), 명령어에 의한 다이렉트 방식					
프로그램 언어			LD, SFC, ST					
명령어수	연산자		18개					
	기본 펑션		136 종 + 실수연산 펑션					
	기본 펑션블록		43개					
	전용 펑션블록		특수기능 모듈별 전용 펑션 블록, 통신전용 펑션블록(P2P)					
연산 처리 속도 (기본 명령)	기본		8.5ns/명령어	28 ns/명령어			84 ns/명령어	
	MOVE		25.5ns/명령어	84 ns/명령어			252 ns/명령어	
	실수연산		±: 119ns(S), 281ns(D) x: 272ns(S), 680ns(D) ÷: 81ns(S), 685ns(D)	±: 392ns(S), 924ns(D) x: 896ns(S), 2,240ns(D) ÷: 924ns(S), 2,254ns(D)			±: 1,442 ns(S), 2,870 ns(D) x: 1,948 ns(S), 4,186 ns(D) ÷: 1,974 ns(S), 4,200 ns(D)	S: 단장 D: 배장
프로그램 메모리 용량			2MB	1MB	512KB	128KB	64KB	
입출력 점수(설치가능)			6,144점			3,072점	1,536점	
최대 입출력 메모리 접점			131,072점			32,768점		
데이터 메모리	자동변수영역(A)		1024KB (최대512KB 리테인 설정 가능)	512KB (최대256KB 리테인 설정 가능)		128KB (최대64KB 리테인 설정 가능)	64KB (최대32KB 리테인 설정 가능)	
	입력변수(I)		16KB			4KB		
	출력변수(Q)		16KB			4KB		
	직접변수	M	256KB (최대128KB 리테인 설정 가능)			64KB (최대32KB 리테인 설정 가능)	32KB (최대16KB 리테인 설정 가능)	
		R	64KB*16 블록	64KB*2블록		64KB x 1블록	32KB x 1블록	
		W	1,024KByte	128KB		64KB	32KB	R과 동일 영역
	플래그 변수	F	8KB	4KB				시스템 플래그
		K	16KB			4KB		PID 운전 영역
		L	22KB					고속링크 플래그
		U	8KB			4KB	2KB	아날로그 데이터 리프레시 영역
플래시 영역			2 MB, 32 블록				1 MB, 16 블록	R디바이스 이용

Copyright ⓒ 2021 LS Electric Co., Ltd.

각 부 명칭 및 기능 (기존)

XGI-CPUU/D, XGI-CPUU, XGI-CPUH, XGI-CPUS, XGI-CPUE에 대한 각 부의 명칭 및 기능입니다.



No.	명 칭	용 도
①-a	RUN/STOP LED	CPU 모듈의 동작 상태를 나타냅니다. • 녹색 점등: 'RUN' 모드 상태로 운전 중 을 표시 ▶ RUN/STOP 모드 스위치에 의해 'RUN' 운전 중 ▶ RUN/STOP 모드 스위치가 'STOP' 인 상태에서 '리모트 RUN' 운전 중 • 적색점등: 'STOP' 모드 상태로 운전 중을 표시 ▶ RUN/STOP 모드 스위치에 의해 'STOP' 운전 중 ▶ 모드 스위치가 'STOP' 인 상태에서 리모트 'STOP' 운전 중 ▶ 운전을 정지하는 에러를 검출한 경우
①-b	REM LED	• 점등(황색): 리모트 허용 상태임을 표시 ▶'REMOTE' 스위치가 'On' 인 경우 • 소등: 리모트 금지 상태임을 표시 ▶ 'REMOTE' 스위치가 'Off' 인 경우
①-c	ERR LED	• 점등(적색): 운전이 불가한 에러가 발생한 경우를 표시 • 소등: 이상 없음을 표시
①-d	PS LED (Programmable Status)	• 점등(적색): ▶ '사용자 지정 플래그' 가 'On' 인 경우 ▶ '에러시 운전 속행'설정으로 에러 상태에서 운전 중인 경우 ▶ 'M.XCHG' 스위치가 'On' 인 상태에서 모듈을 빼거나 다른 모듈을 장착한 경우 • 소등: ▶ 이상 없음을 표시

Copyright ⓒ 2021 LS Electric Co., Ltd.

No.	명 칭	용 도
①-e	BAT LED	• 점등(적색): 배터리 전압이 저하된 경우 • 소등: 배터리 이상 없음
①-f	CHK LED	• 점등(적색): 표준설정과 다른 내용이 설정되어 있는 경우에 표시 (파라미터로 추가/삭제[해제]가 가능함) ▶ '모듈교체' 스위치가 '모듈교체'로 설정 된 경우 ▶ '디버그 모드' 에서 운전 중 인 경우 ▶ '강제 ON' 설정 상태 ▶ '고장마스크', 'SKIP' 플래그가 설정 된 경우 ▶ 운전 중 경고장(Warning)이 발생한 경우 ▶ 증설베이스 전원 이상 • 소등: 표준설정으로 운전 중에 표시
②-a	Boot/Nor 스위치	출하 전 O/S를 다운로드 하는 경우 사용합니다. • On (우측): 정상운전 모드에서 제어동작을 수행 • Off (좌측): 제조 시 사용하는 모드로 사용자 조작 금지(O/S의 다운로드 모드)
		Boot/Nor 스위치는 항상 On(우측)상태로 유지해야 합니다. Off(좌측) 상태로 설정하게 되면 모듈 소손의 원인이 됩니다.
②-b	REMOTE허용 스위치	리모트 접속을 통한 PLC의 동작을 제한 합니다. • On(우측): 모든 기능 허용 (REMOTE모드) • Off(좌측): 리모트 기능 제한 ▶ 프로그램의 D/L, 운전모드 조작 제한 ▶ 모니터, 데이터 변경 등은 조작 허용
②-c	M.XCHG (모듈교체 스위치)	운전 중 모듈교체를 실시하는 경우 사용합니다. • On (우측): 모듈교체 실시 ▶ 키스위치의 조작만으로 모듈교체가 가능 • Off(좌측): 모듈교체 완료
③	RUN/STOP 모드 스위치	CPU 모듈의 운전모드를 설정합니다. • STOP → RUN: 프로그램의 연산 실행 • RUN → STOP: 프로그램의 연산 정지 REMOTE 스위치에 우선하여 동작 합니다.

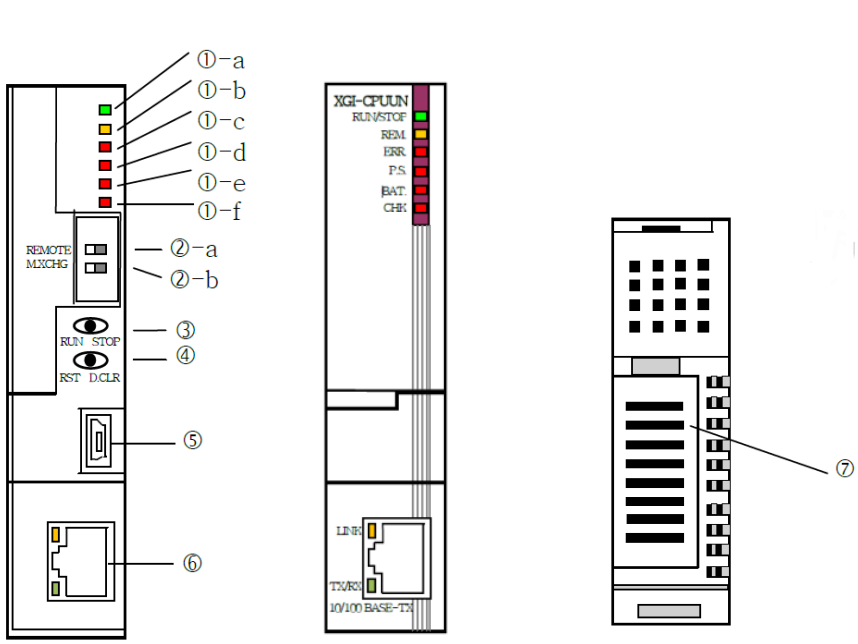
Copyright ⓒ 2021 LS Electric Co., Ltd.

<table>
<tr><th>No.</th><th>명 칭</th><th>용 도</th></tr>
<tr><td>④</td><td>리셋/D.Clear 스위치</td><td>리셋/D.Clear에 대해 "XG5000 → 기본 파라미터 → 기본 동작 설정"에 대해 동작 차단을 설정 할 수 있습니다.
• 리셋 스위치 동작 차단 설정을 하지 않을 경우
<table>
<tr><th>동 작</th><th>결 과</th></tr>
<tr><td>좌측이동 → 중앙복귀</td><td>리셋</td></tr>
<tr><td>좌측이동 → 3초 이상 유지 → 중앙복귀</td><td>오버올 리셋</td></tr>
</table>
• D.Clear 스위치 동작 차단 설정을 하지 않을 경우
<table>
<tr><th>동 작</th><th>결 과</th></tr>
<tr><td>우측으로 누름
→ 중앙복귀</td><td>일반 데이터 영역 및 리테인 설정 영역 (M, 자동변수) 데이터 지움</td></tr>
<tr><td>우측으로 누름
→ 3초 이상 유지
→ 중앙복귀</td><td>일반 데이터 영역, 리테인 설정 영역 (M, 자동변수) 데이터 지움과 R영역 데이터 까지 지움</td></tr>
</table>
• 데이터 클리어 동작은 "STOP" 운전모드에서만 동작합니다.</td></tr>
<tr><td>⑤</td><td>USB 커넥터</td><td>주변기기 (XG5000 등)와 접속하기 위한 커넥터 (USB 1.1 지원)</td></tr>
<tr><td>⑥</td><td>RS–232C 커넥터</td><td>주변기기와 접속하기 위한 커넥터
• XG5000 접속: 기본적으로 지원
• Modbus 기기 접속: Modbus 프로토콜 지원
TX: 7번Pin, RX: 8번Pin, GND: 5번 Pin</td></tr>
<tr><td>⑦</td><td>배터리 장착 커버</td><td>백업 배터리 장착용 커버</td></tr>
</table>

Copyright ⓒ 2021 LS Electric Co., Ltd.

각 부 명칭 및 기능 (XGI-CPUUN)

XGI-CPUUN에 대한 각 부의 명칭 및 기능은 다음과 같습니다.



No.	명칭	용도
①-a	RUN/STOP LED	CPU 모듈의 동작 상태를 나타냅니다. • 녹색 점등: 'RUN' 모드 상태로 운전 중 을 표시 ▶ RUN/STOP 모드 스위치에 의해 'RUN' 운전 중 ▶ RUN/STOP 모드 스위치가 'STOP' 인 상태에서 '리모트 RUN' 운전 중 • 적색점등: 'STOP' 모드 상태로 운전 중을 표시 ▶ RUN/STOP 모드 스위치에 의해 'STOP' 운전 중 ▶ 모드 스위치가 'STOP' 인 상태에서 리모트 'STOP' 운전 중 ▶ 운전을 정지하는 에러를 검출한 경우
①-b	REM LED	• 점등(황색): 리모트 허용 상태임을 표시 ▶'REMOTE' 스위치가 'On' 인 경우 • 소등: 리모트 금지 상태임을 표시 ▶ 'REMOTE' 스위치가 'Off' 인 경우
①-c	ERR LED	• 점등(적색): 운전이 불가한 에러가 발생한 경우를 표시 • 소등: 이상 없음을 표시

Copyright ⓒ 2021 LS Electric Co., Ltd.

No.	명 칭	용 도
①-d	PS LED (Programmable Status)	• 점등(적색): ▶ '사용자 지정 플래그' 가 'On' 인 경우 ▶ '에러 시 운전 속행'설정으로 에러 상태에서 운전 중인 경우 ▶ 'M.XCHG' 스위치가 'On' 인 상태에서 모듈을 빼거나 다른 모듈을 장착한 경우 • 소등: ▶ 이상 없음을 표시
①-e	BAT LED	• 점등(적색): 배터리 전압이 저하된 경우 • 소등: 배터리 이상 없음
①-f	CHK LED	• 점등(적색): 표준설정과 다른 내용이 설정되어 있는 경우에 표시 (파라미터로 추가/삭제[해제]가 가능함) ▶ '모듈교체' 스위치가 '모듈교체'로 설정 된 경우 ▶ '디버그 모드' 에서 운전 중 인 경우 ▶ '강제 ON' 설정 상태 ▶ '고장마스크', 'SKIP' 플래그가 설정 된 경우 ▶ 운전 중 경고장(Warning)이 발생한 경우 ▶ 증설베이스 전원 이상 • 소등: 표준설정으로 운전 중에 표시
②-a	REMOTE허용 스위치	리모트 접속을 통한 PLC의 동작을 제한 합니다. • On(우측): 모든 기능 허용 (REMOTE모드) • Off(좌측): 리모트 기능 제한 ▶ 프로그램의 D/L, 운전모드 조작 제한 ▶ 모니터, 데이터 변경 등은 조작 허용
②-b	M.XCHG (모듈교체 스위치)	운전 중 모듈교체를 실시하는 경우 사용합니다. • On (우측): 모듈교체 실시 ▶ 키스위치의 조작만으로 모듈교체가 가능 • Off(좌측): 모듈교체 완료
③	RUN/STOP 모드 스위치	CPU 모듈의 운전모드를 설정합니다. • STOP → RUN: 프로그램의 연산 실행 • RUN → STOP: 프로그램의 연산 정지 REMOTE 스위치에 우선하여 동작 합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

<table>
<tr><th>No.</th><th>명 칭</th><th>용 도</th></tr>
<tr><td>④</td><td>리셋/ D.Clear
스위치</td><td>
리셋/D.Clear에 대해 "XG5000 → 기본파라미터 → 기본 동작 설정"에 대해 동작 차단을 설정 할 수 있습니다.
• 리셋 스위치 동작 차단 설정을 하지 않을 경우
<table>
<tr><th>동 작</th><th>결 과</th></tr>
<tr><td>좌측이동 → 중앙복귀</td><td>리셋</td></tr>
<tr><td>좌측이동 → 3초 이상 유지 → 중앙복귀</td><td>오버올 리셋</td></tr>
</table>
• D.Clear 스위치 동작 차단 설정을 하지 않을 경우
<table>
<tr><th>동 작</th><th>결 과</th></tr>
<tr><td>우측으로 누름
→ 중앙복귀</td><td>일반 데이터 영역 및 리테인 설정 영역 (M, 자동변수) 데이터 지움</td></tr>
<tr><td>우측으로 누름
→ 3초 이상 유지
→ 중앙복귀</td><td>일반 데이터 영역, 리테인 설정 영역 (M, 자동변수) 데이터 지움과 R영역 데이터 까지 지움</td></tr>
</table>
• 데이터 클리어 동작은 "STOP" 운전모드에서만 동작합니다.
</td></tr>
<tr><td>⑤</td><td>USB 커넥터</td><td>주변기기 (XG5000 등)와 접속하기 위한 커넥터 (USB 1.1 지원)</td></tr>
<tr><td>⑥</td><td>Ethernet 커넥터</td><td>주변기기와 접속하기 위한 커넥터
• XG5000 접속: 기본적으로 지원
• TCP/IP 서버 접속</td></tr>
<tr><td>⑦</td><td>배터리 장착 커버</td><td>백업 배터리 장착용 커버</td></tr>
</table>

Copyright ⓒ 2021 LS Electric Co., Ltd.

3장. 데이터 메모리 구성

Copyright ⓒ 2021 LS Electric Co., Ltd.

3.1. 변수의 표현 방식

프로그램 안에서 사용되는 데이터는 값을 가지고 있는데, 프로그램이 실행되는 동안에 값이 바뀌지 않는 상수와 그 값이 변하는 변수가 있습니다. 프로그램 블록, 펑션, 펑션블록 등의 프로그램 구성 요소에서 변수를 사용하기 위하여 먼저 변수의 표현 방식을 설명합니다. 변수의 표현에는 다음의 2가지가 있습니다.

변수의 표현
- **직접 변수: PLC의 입·출력 또는 기억 장소에 대하여 직접적으로 표현하는 것(변수 선언 불필요)**
- **심볼릭(Symbolic) 변수: 변수 선언 필요**

첫 번째 변수 표현 방식은 사용자가 이름을 부여하지 않고 이미 Maker에 의해 지정된 메모리 영역의 식별자를 사용하는 **직접 변수** 방식이고, 두 번째 변수 표현 방식은 사용자가 이름을 부여하고 사용하는 **심볼릭(Symbolic; Named) 변수** 방식입니다. 프로그램 구성 요소(프로그램 블록, 펑션, 펑션블록)에서 심볼릭변수를 사용하기 위해서는 반드시 변수를 선언해야 합니다.

직접 변수

직접 변수는 사용자가 변수 이름과 형 등의 선언이 없이 이미 Maker에 의해 정해진 메모리 영역의 식별자와 주소를 사용합니다. **직접 변수에는 %I, %Q의 입출력 변수와 %M, %R, %W 의 내부 메모리 변수가 있습니다.** 입출력 변수와 내부 메모리 변수 크기는 PLC 종류에 따라 차이가 있습니다. 직접 변수는 별도의 변수 선언 없이 식별자의 위치를 표현하는 방식이므로 프로그램의 가독성이 떨어지며 사용자의 잘못으로 어드레스가 중복될 우려가 있습니다.

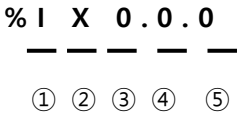
직접 변수는 반드시 퍼센트 문자(%)로 시작하고 다음에 위치 접두어와 크기 접두어를 붙이며 그리고 마침표로 분리되는 하나 이상의 부호 없는 정수의 순으로 나타냅니다.

종류	사용 예
입력 변수	%IX0.0.0, %IB0.0.1, %IW0.0.1, %ID0.0.0
출력 변수	%QX0.1.0, %QB0.1.1, %QW0.1.1, %QD0.1.0
내부 메모리	%MX100, %MB50, %MW100, %MD100
	%MB50.3, %MW100.10, %MD100.31

Copyright ⓒ 2021 LS Electric Co., Ltd.

▶ 입출력 메모리의 할당

XGI 시리즈 PLC의 입,출력 메모리의 할당은 다음의 5가지 인자에 의해 결정됩니다.



① **위치 접두어** - 변수의 종류를 나타냅니다.

접두어	의 미
I	입력 위치(Input Location)
Q	출력 위치(Output Location)
M	내부 메모리 중 M영역 위치(Memory Location)
R	내부 메모리 중 R영역 위치(Memory Location)
W	내부 메모리 중 W영역 위치(Memory Location)

② **크기 접두어** - 변수가 차지하는 메모리 공간의 크기를 나타냅니다.

접두어	의 미
X	1 비트의 크기 (“X”문자에 한하여 생략 가능)
B	1 바이트(8 비트)의 크기
W	1 워드(16 비트)의 크기
D	1 더블 워드(32 비트)의 크기
L	1 롱 워드(64 비트)의 크기

③ **베이스의 번호**

CPU가 장착되어 있는 베이스(기본 베이스)를 0번 베이스라 하며, 증설 시스템을 구성했을 때 기본 베이스에 접속된 순서에 따라 베이스 번호가 증가 됩니다.

XGI-CPUH는 총 8베이스(1기본 + 7증설)로 베이스 번호는 0 ~7로 설정됩니다.

XGI-CPUS는 총 4베이스(1기본 + 3증설)로 베이스 번호는 0 ~3으로 설정됩니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

④ **슬롯 번호**

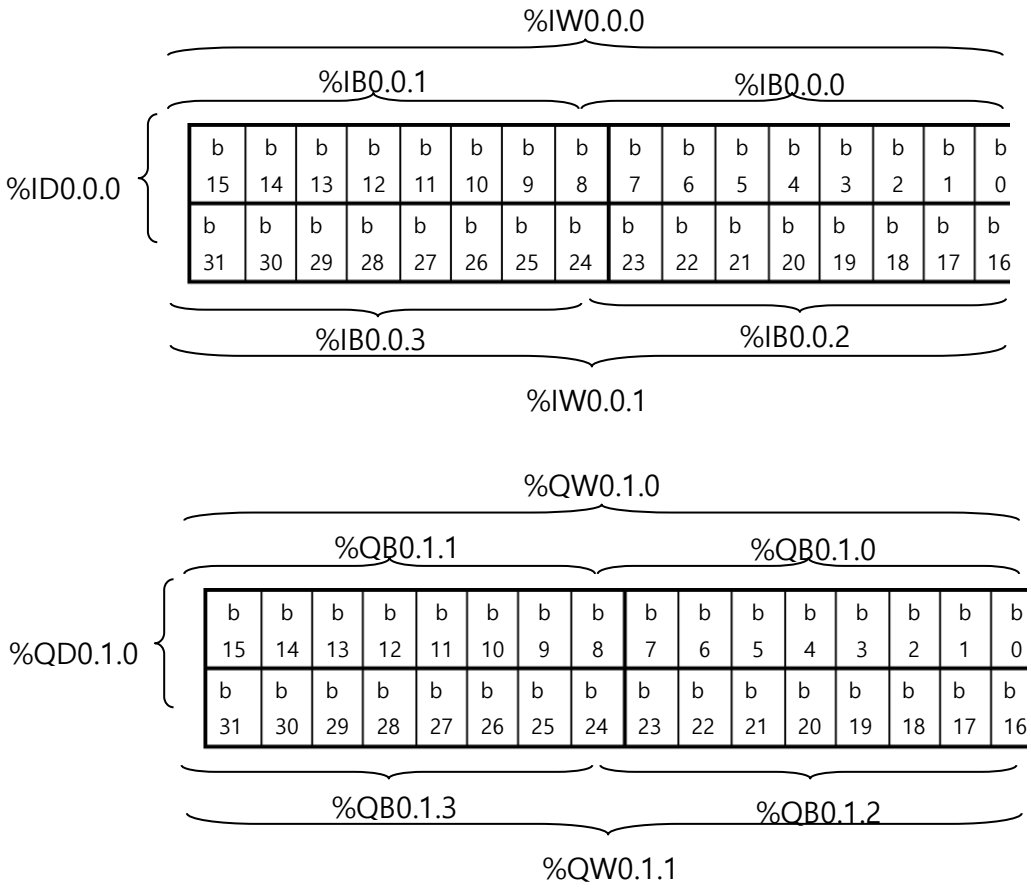
슬롯 번호는 기본 베이스의 경우 CPU 우측이 0번이 되며 우측으로 진행하며 번호가 1씩 증가하게 됩니다. 증설 베이스의 경우 전원부 우측이 0번이 되며, 우측으로 진행하며 번호가 1씩 증가하게 됩니다. XGI-CPUH 및 XGI-CPUS 기본 베이스는 총 12 슬롯으로 슬롯 번호는 0 ~11로 설정됩니다.

⑤ **크기 접두어 번호**

슬롯에 장착되어 있는 접점들을 0번 비트부터 크기 접두어 단위로 나누었을 때 몇 번째 크기 접두어 단위가 되는지를 나타냅니다. 예를 들면 0번 슬롯에 32점 입력 모듈이 장착되어 있고, 이것을 바이트 단위로 나누어 사용한다면 처음 8점(%IX0.0.0~%IX0.0.7)은 %IB0.0.0이 되고, 그 다음 8점(%IX0.0.8~%IX0.0.15)은 %IB0.0.1이 되며, 그 다음 8점(%IX0.0.16~%IX0.0.23)은 %IB0.0.2가 됩니다. 그리고 마지막 8점(%IX0.0.24 ~%IX0.0. 31)은 %IB0.0.3이 됩니다. 또한 1번 슬롯에 32점 출력 모듈이 장착되어 있고, 이것을 워드 단위로 나누어 사용한다면 처음 16점(%QX0.1.0~%QX0.1.15)은 %QW0.1.0이 되며, 그 다음 16점(%QX0.1.16~%QX0.1.31)은 %QW0.1.1이 됩니다.

※ 접두어로는 소문자가 올 수 없습니다.

※ 크기 접두어를 붙이지 않으면 그 변수는 1비트로 처리합니다.



Copyright ⓒ 2021 LS Electric Co., Ltd.

▶ 내부 메모리 할당

내부 메모리의 할당은 위에서 설명한 입, 출력 메모리의 할당과 기본적인 방법은 동일하나 베이스 번호와 슬롯 번호를 지정하지 않습니다. 내부 메모리를 표현하는 다음의 두 가지 방법이 있습니다.

① 크기 접두어 단위의 표현

$\%MX N_1$ (N_1 은 숫자)

① ② ③

①번 항목 %M은 내부 메모리를 나타내는 위치 접두어 입니다.

②번 항목은 크기 접두어로서 입출력 메모리와 동일 합니다.

③번 항목은 크기 접두어 번호를 나타냅니다.

② 크기 접두어를 이용한 비트 표현

$\%MB N_1 . N_2$ (N_1, N2 는 숫자)

① ② ③ ④

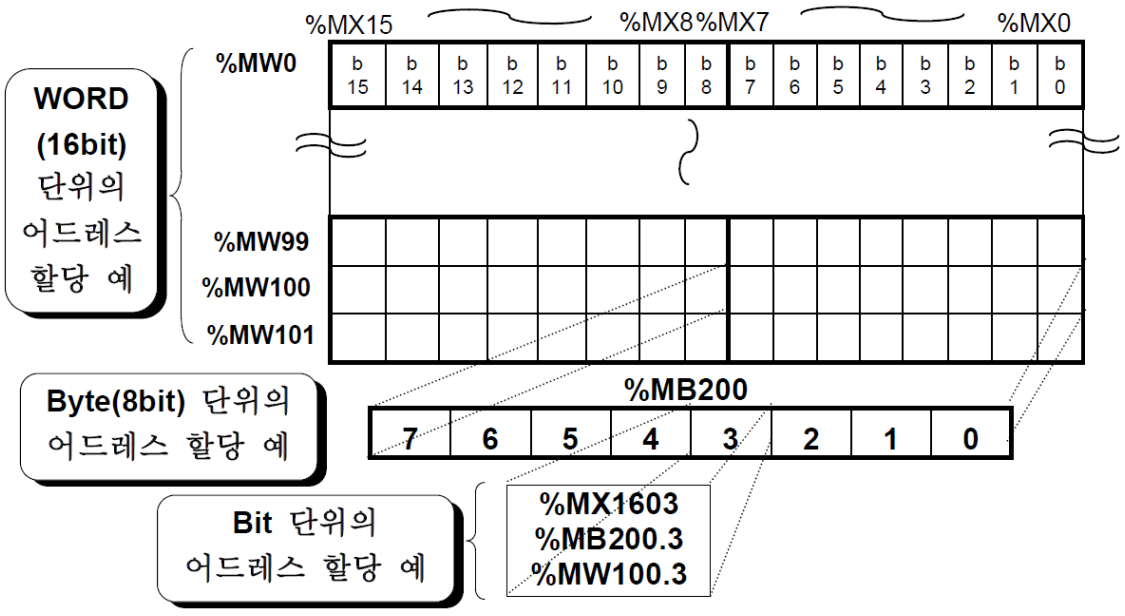
①번 항목 %M은 내부 메모리를 나타내는 위치 접두어 입니다.

②번 항목은 크기 접두어로서 X를 제외한 B, W, D, L을 사용할 수 있습니다.

③번 항목은 크기 접두어 번호를 나타냅니다.

④번 항목은 비트 번호입니다.

예를 들어 %MW100.3이라고 하면 100워드의 3번 비트를 의미합니다.



Copyright ⓒ 2021 LS Electric Co., Ltd.

심볼릭 변수 (Symbolic Variable)

심볼릭 변수는 사용자가 변수 이름과 형 등을 선언하고 사용합니다.

심볼릭 변수의 이름은 일반적으로 자 수의 제한이 없으며 한글, 영문, 숫자 및 밑줄 문자(_)를 조합하여 사용할 수 있습니다.

또한, 영문자의 경우 대·소 문자 모두 입력 가능하며 동일한 문자면 구별하지 않고 모두 같은 변수로 인식합니다. 단, 변수 이름에 빈 칸을 포함할 수 없습니다.

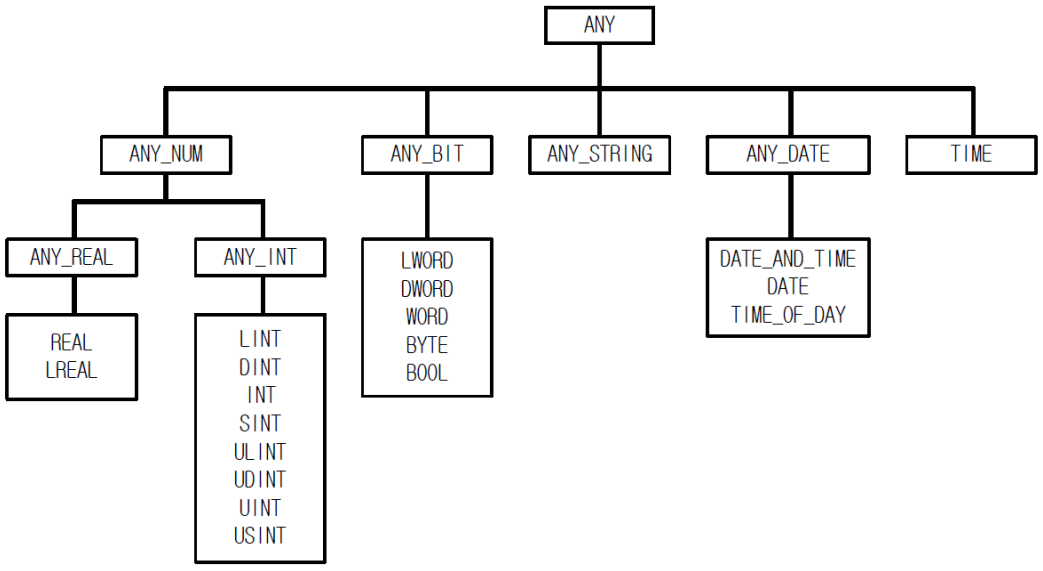
종류	사용 예
한글, 숫자 및 밑줄 문자	모터10, 디지털_스위치1 , 누름_검출, 수동_배출_스위치 밸브1, 설비_자동_운전중, 사이클_정지_완료
한글, 영문, 숫자 및 밑줄 문자	AGV_주행_완료, 모터2_ON, BCD값, , VAL2, 자동_SOL_배출

※ 심볼릭 변수의 변수 선언 절차는 다음과 같습니다.

(데이터 타입(type) 지정 **→ 변수 종류** 설정 **→ 메모리 할당)**

① 심볼릭 변수의 데이터 타입 (Type)

데이터의 고유 성질을 나타냅니다. 데이터형은 크게 수치(ANY_NUM)와 비트 상태(ANY_BIT)로 구분할 수 있습니다. 수치의 대표적인 경우는 정수(INT; Integer)인데 셀 수 있고 산술 연산을 할 수 있습니다. 정수의 예는 카운터의 현재 값, A/D(아날로그 입력) 변환 값 등이 있습니다. 비트 상태는 BOOL(1비트), BYTE(8 개의 비트 열), WOTD(16 개의 비트 열)등이 있는데 비트 열의 On/Off 상태를 나타내며 논리 연산을 할 수 있습니다. 비트 상태의 예는 입력 스위치의 On/Off 상태, 출력 램프의 소등/점등 상태 등이 있습니다. 비트 상태는 산술연산이 불가능하지만 형(Type) 변환 펑션을 사용수치로 변환하면 산술 연산이 가능합니다. BCD는 10진수를 4 비트의 2진 코드로 나타낸 것이므로 비트 상태(ANY_BIT)입니다.



Copyright ⓒ 2021 LS Electric Co., Ltd.

※ ANY_BIT는 위의 계층도와 같이 BOOL, BYTE, WORD, DWORD, LWORD를 모두 포함 합니다.

※ ANY_NUM은 위의 계층도와 같이 LREAL, REAL, LINT, DINT, INT, SINT, ULINT, UDINT, UINT, USINT를 모두 포함하여 사용됩니다.

기본 데이터 형 (Type)

구 분	예 약 어	데이터 형	크기 (비트)	범 위
수치 (ANY_NUM)	SINT	Short Integer	8	-128 ~ 127
	INT	Integer	16	-32768 ~ 32767
	DINT	Double Integer	32	-2147483648 ~ 2147483647
	LINT	Long Integer	64	-2^{63} ~ $2^{63}-1$
	USINT	Unsigned Short Integer	8	0 ~ 255
	UINT	Unsigned Integer	16	0 ~ 65535
	UDINT	Unsigned Double Integer	32	0 ~ 4294967295
	ULINT	Unsigned Long Integer	64	0 ~ $2^{64}-1$
	REAL	Real Numbers	32	-3.402823466e+038~ -1.175494351e-038 or 0 or 1.175494351e-038 ~ 3.402823466e+038
	LREAL	Long Reals	64	-1.7976931348623157e+308 ~ -2.2250738585072014e-308 or 0 or 2.2250738585072014e-308 ~ 1.7976931348623157e+308
시간	TIME	Duration	32	T#0S ~ T#49D17H2M47S295MS
날짜	DATE	Date	16	D#1984-01-01 ~ D#2163-6-6
	TIME_OF_DAY	Time Of Day	32	TOD#00:00:00 ~ TOD#23:59:59.999
	DATE_AND_TIME	Date And Time Of Day	64	DT#1984-01-01-00:00:00 ~ DT#2163-12-31-23:59:59.999
문자열	STRING	Character String	30*8	-
비트 상태 (ANY_BIT)	BOOL	Boolean	1	0, 1
	BYTE	Bit String Of Length 8	8	16#0 ~ 16#FF
	WORD	Bit String Of Length 16	16	16#0 ~ 16#FFFF
	DWORD	Bit String Of Length 32	32	16#0 ~ 16#FFFFFFFF
	LWORD	Bit String Of Length 64	64	16#0 ~ 16#FFFFFFFFFFFFFFFF

Copyright ⓒ 2021 LS Electric Co., Ltd.

① 심볼릭 변수의 종류

변수를 어떻게 선언할 것인가를 설정합니다.

변수 종류	내 용
VAR	읽고 쓸 수 있는 일반적인 변수
VAR_CONSTANT	항상 고정된 값을 가지고 있는 읽기만 할 수 있는 변수 (상수)
VAR_EXTERNAL	VAR_GLOBAL로 선언된 변수를 사용하기 위한 선언
VAR_EXTERNAL_ CONSTANT	VAR_GLOBAL로 선언된 상수를 사용하기 위한 선언

※ 정전 유지 설정은 XG5000 변수 편집 창 하단에 있는 '리테인 설정'을 체크합니다.

※ 글로벌 변수란 하나의 프로젝트에 포함되는 여러 프로그램 블록에서 동일한 변수 이름으로 동시에 사용할 수 있는 변수입니다.

② 네임드(Named) 변수의 메모리 할당

네임드 변수의 메모리 할당에는 **자동 할당**과 **사용자 정의**가 있습니다.

자동 할당이란 컴파일러가 내부 메모리 영역에 변수 위치를 자동으로 지정 합니다. 예를 들어 "**밸브"**란 변수를 자동 메모리 할당으로 지정할 경우 변수의 내부 위치는 프로그램이 작성된 후 컴파일(Compile) 과정에서 정해지므로 사용자는 변수 위치에 신경을 쓸 필요가 없습니다. 선언된 변수는 외부 입출력과 관계 없이 내부 연산 도중 신호의 중계, 신호 상태(내부 정보)의 일시 저장, 타이머나 카운터의 접점 이름(펑션 블록의 인스턴스) 지정 등에 사용됩니다.

※ 자동 할당 지정은 XG5000의 변수 편집 창에서 메모리 할당 란을 블랭크로 하면 됩니다

사용자 정의란 사용자가 직접 변수(%I, %Q 및 %M, %R)를 사용하여 강제로 위치를 지정합니다. 선언된 변수는 입출력용(%I, %Q) 변수에 사용합니다.

※ 사용자 정의 메모리 할당 지정은 XG5000의 변수 편집 창에서 메모리 할당란에 입출력 주소를 직접 입력하면 됩니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

4장. XG5000 사용법

Copyright ⓒ 2021 LS Electric Co., Ltd.

4.1 XG5000 특징

XG5000은 XGT PLC 시리즈에 대해서 프로그램을 작성하고 디버깅하는 소프트웨어 툴입니다.

XG5000은 다음과 같은 특징과 장점을 가지고 있습니다.

멀티 PLC, 멀티 프로그램

한 프로젝트에 여러 개의 PLC를 포함시켜서 서로 연동되는 PLC 시스템을 동시에 편집, 모니터, 관리할 수 있습니다.

또한 프로그램을 스캔 프로그램, 다양한 태스크 프로그램으로 나누어 작성할 수 있습니다.

다양한 드래그 & 드롭

프로젝트, 변수/설명, LD 편집, 변수 모니터 등 대부분의 편집기에서 드래그 & 드롭 기능을 적용하여 편집을 쉽고 편리하게 할 수 있습니다.

사용자 단축키 설정

디폴트로 제공되는 단축키 변경이 가능하며 사용자 본인에게 익숙한 단축키를 추가할 수 있습니다.

다양한 메시지 창

프로그램 편집과 검사 등을 쉽게 하기 위하여 다양한 메시지 창을 제공합니다.

편리한 변수/설명 편집

엑셀을 이용하여 편집 가능합니다.
변수 위주 보기, 디바이스 위주 보기, 플래그 보기 등 다양한 형식으로 편집이 가능합니다.
오토필(Auto Fill)을 이용하여 비슷한 용도의 변수를 쉽게 추가할 수 있습니다.
드래그 & 드롭을 이용하여 다른 변수/설명 창에서 쉽게 복사할 수 있습니다.
대화 상자를 띄우지 않고 엑셀처럼 직접 편집할 수 있습니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

편리한 프로그램 편집

제한 없는 Undo/Redo 기능을 제공합니다.

셀 단위 블록 편집이 가능합니다.

화면 분할 편집이 가능합니다.

찾기/바꾸기 기능을 강화하였습니다.

블록 마스크 기능을 이용하여 렁 단위로 실행을 금지할 수 있습니다.

북 마크 기능을 이용하여 특정 위치에 쉽게 찾아갈 수 있습니다.

LD 편집을 할 때 선택된 디바이스에 대해서 메모리 참조를 볼 수 있습니다.

다양한 모니터 기능

변수 모니터, 디바이스 모니터, 시스템 모니터, 트렌드 모니터, 특수 모듈 모니터 등 다양한 모니터 기능을 제공합니다.

사용자 이벤트

특정 디바이스에 대해서 사용자가 정의한 조건이 만족 시 Data를 Logging하고 열람할 수 있습니다.

모듈 교환 마법사

런 중에 PLC를 정지시키지 않고 안전하고 쉽게 모듈을 교환할 수 있습니다.

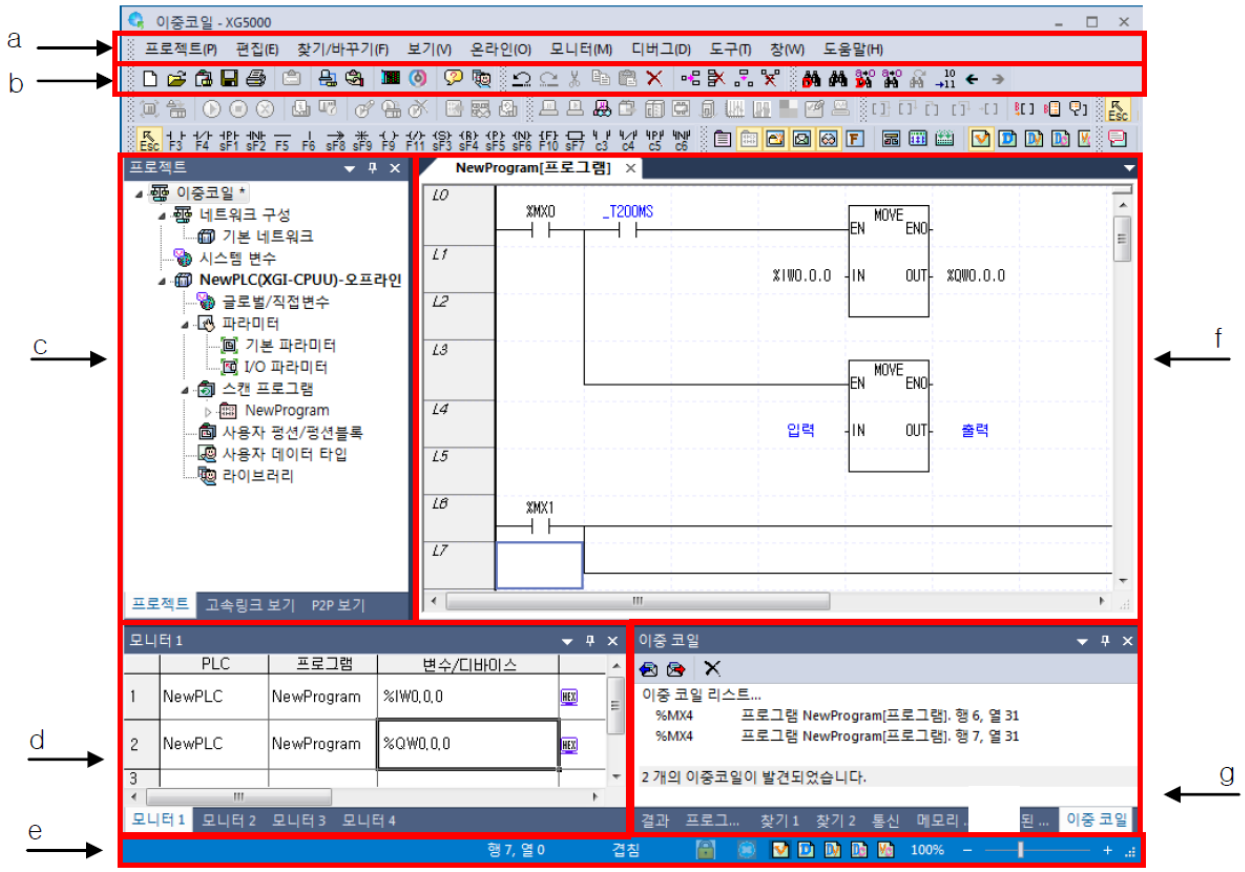
Copyright ⓒ 2021 LS Electric Co., Ltd.

4.2 기본 사용법

화면 구성

XG5000의 화면은 아래 그림과 같은 구성으로 이루어져 있습니다.

[대화 상자]



[대화 상자 설명]

a. 메뉴: 프로그램을 위한 기본 메뉴입니다.

b. 도구모음: 메뉴를 간편하게 실행할 수 있습니다.

c. 프로젝트 창: 현재 열려있는 프로젝트의 구성 요소를 나타냅니다.

d. 변수 모니터 창: 변수를 등록하여 모니터 할 수 있습니다.

e. 상태 바: XG5000의 상태, 접속된 PLC의 정보 등을 나타냅니다.

f. 편집 창: 현재 LD 편집 창이 보이고 있습니다.

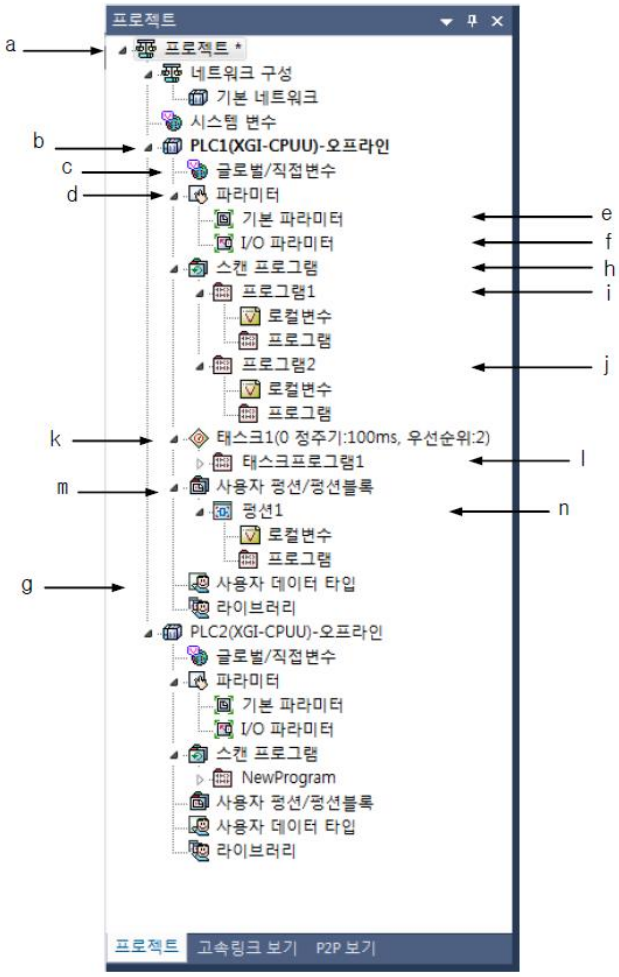
g. 메시지 창: XG5000 사용 중에 발생하는 각종 메시지가 나타납니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

4.3 프로젝트

프로젝트 구성

프로젝트의 구성 항목은 다음과 같습니다.



a. 프로젝트: 시스템 전체를 정의합니다. 하나의 프로젝트에 여러 개의 관련된 PLC를 포함시킬 수 있습니다.

b. PLC: CPU 모듈 하나에 해당되는 시스템을 나타냅니다.

c. 글로벌/직접변수: 글로벌 변수 선언과 직접변수 설명문을 편집하고 볼 수 있습니다.

d. 파라미터: PLC 시스템의 동작 및 구성에 대한 내용을 정의합니다.

e. 기본 파라미터: 기본적인 동작에 대하여 정의합니다.

f. I/O 파라미터: 입출력 모듈 구성에 대하여 정의합니다.

g. 데이터 타입: 구조체(Structure) 타입을 정의합니다.

h. 스캔 프로그램: 항시 실행되는 프로그램을 하위 항목에 정의합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

i. 프로그램1: 사용자가 정의한 항시 실행되는 프로그램입니다.

j. 프로그램2: 사용자가 정의한 항시 실행되는 프로그램입니다.

k. 태스크1: 사용자가 정의한 정주기 태스크입니다.

l. 태스크프로그램1: 태스크1 조건에 따라 실행되는 프로그램입니다.

m. 사용자 펑션/펑션블록: 하위 항목에 사용자가 펑션/펑션 블록을 작성합니다.

n. 펑션1: 사용자가 작성한 펑션입니다.

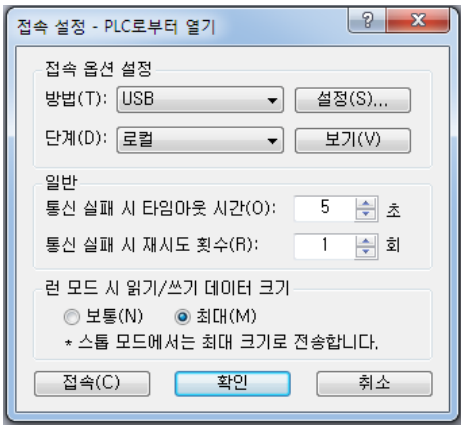
※ 하나의 프로젝트에 여러 개의 PLC가 포함될 수 있습니다. 이처럼, 한 프로젝트에 여러 PLC를 사용할 경우 관리가 용이하고, 하나의 XG5000을 실행한 후 여러 PLC에 동시 접속하여 모니터 할 수도 있습니다.

▶ PLC로부터 열기

PLC에 저장된 내용을 읽어와 프로젝트를 새로 만들어 줍니다. XG5000에 이미 프로젝트가 열려 있다면 이 프로젝트는 닫고 프로젝트를 새로 만들어 줍니다.

[순서]

1. 메뉴 [프로젝트]-[PLC로부터 열기]를 선택합니다.



2. 대화 상자에서 접속할 대상을 선택하고 확인을 누릅니다. 통신 설정의 자세한 내용은 온라인의 접속 옵션을 참조하십시오.

3. 새로운 프로젝트가 생성됩니다.

PLC로부터 읽은 프로젝트는 메뉴 [프로젝트]-[프로젝트 저장]을 선택해야 PC에 저장됩니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

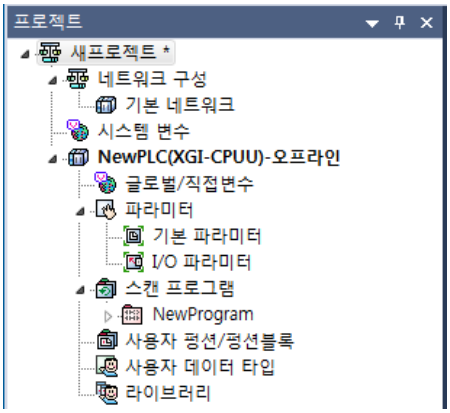
▶ 프로젝트 저장

변경된 프로젝트를 저장합니다.

[순서]

메뉴 [프로젝트]-[프로젝트 저장]을 선택합니다.

※ 프로젝트가 편집되어 저장할 필요가 있을 경우에는 프로젝트 창의 프로젝트 이름 옆에 "*"가 나타납니다.



4.4 변수/설명

사용자들은 프로그램에 따라 변수를 사용합니다. 일반적으로, 글로벌 변수는 모든 프로그램에서 사용 가능한 변수입니다. 글로벌 변수를 로컬 변수에서 사용 하려면 EXTERNAL으로 선언하고 사용하여야 합니다. 로컬 변수는 해당 프로그램에서만 사용이 가능한 변수 입니다. 프로그램에서 직접 변수를 사용 할 수 있습니다. 또한 해당 직접 변수에 설명문을 입력할 수 있습니다.

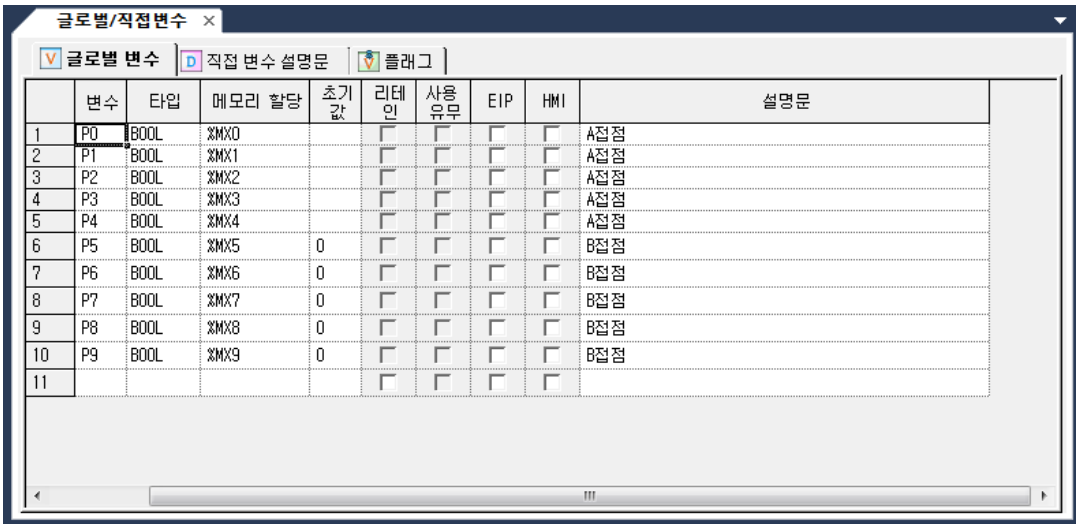
글로벌/직접변수

글로벌/직접변수는 글로벌 변수, 직접 변수 설명문, 플래그로 구성되어 있습니다. 글로벌 변수는 프로그램에서 사용될 변수를 선언하거나, 선언된 변수 목록 전체를 변수 위주로 보여줍니다. 직접 변수 설명문은 프로그램에서 사용될 직접 변수 설명문을 선언하거나, 설명문을 보여줍니다. 플래그는 선언해서 제공해주는 플래그 목록을 보여줍니다. 플래그 종류는 시스템 플래그, 고속 링크 플래그, P2P 플래그, PID 플래그로 분류할 수 있습니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

▶ 글로벌 변수

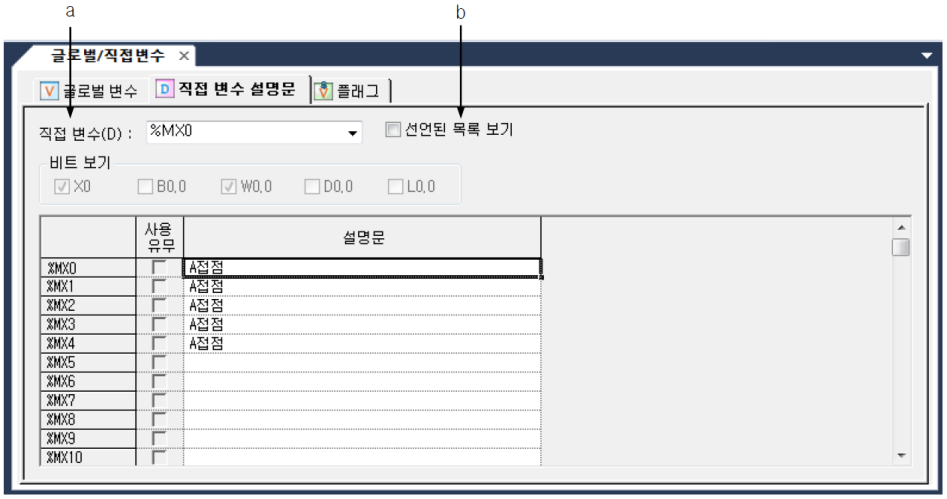
변수를 선언하고 선언된 글로벌 변수 목록 전체를 보여줍니다.



	변수	타입	메모리 할당	초기값	리테인	사용유무	EIP	HMI	설명문
1	P0	BOOL	%MX0						A접점
2	P1	BOOL	%MX1						A접점
3	P2	BOOL	%MX2						A접점
4	P3	BOOL	%MX3						A접점
5	P4	BOOL	%MX4						A접점
6	P5	BOOL	%MX5	0					B접점
7	P6	BOOL	%MX6	0					B접점
8	P7	BOOL	%MX7	0					B접점
9	P8	BOOL	%MX8	0					B접점
10	P9	BOOL	%MX9	0					B접점
11									

▶ 직접 변수 설명문

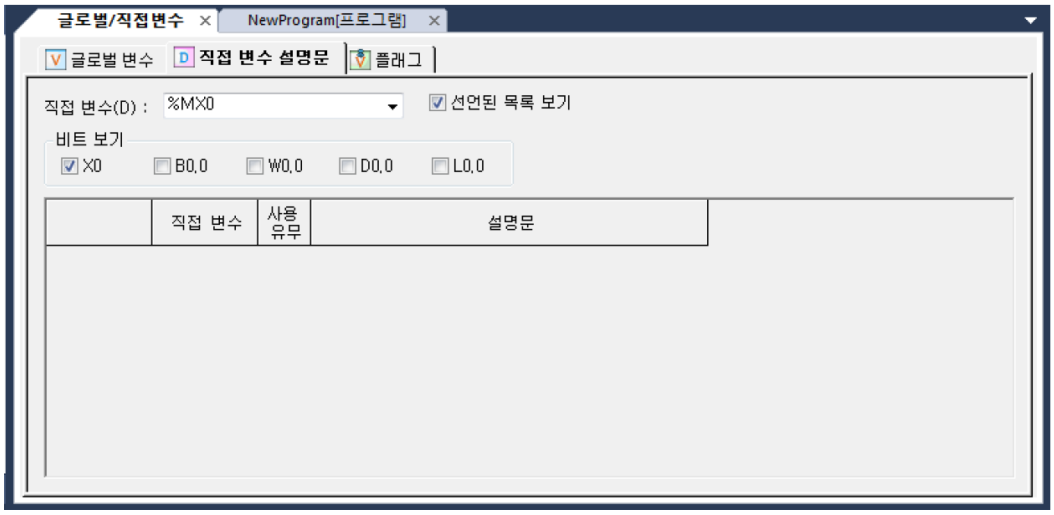
직접변수를 입력하거나 입력된 직접 변수로부터 선언된 설명문을 보여줍니다.



a. 직접 변수: 해당하는 직접 변수를 입력하면 직접 변수 설명문의 데이터가 표시됩니다.

b. 선언된 목록 보기: 직접 변수 설명문이 있는 목록을 표시 합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

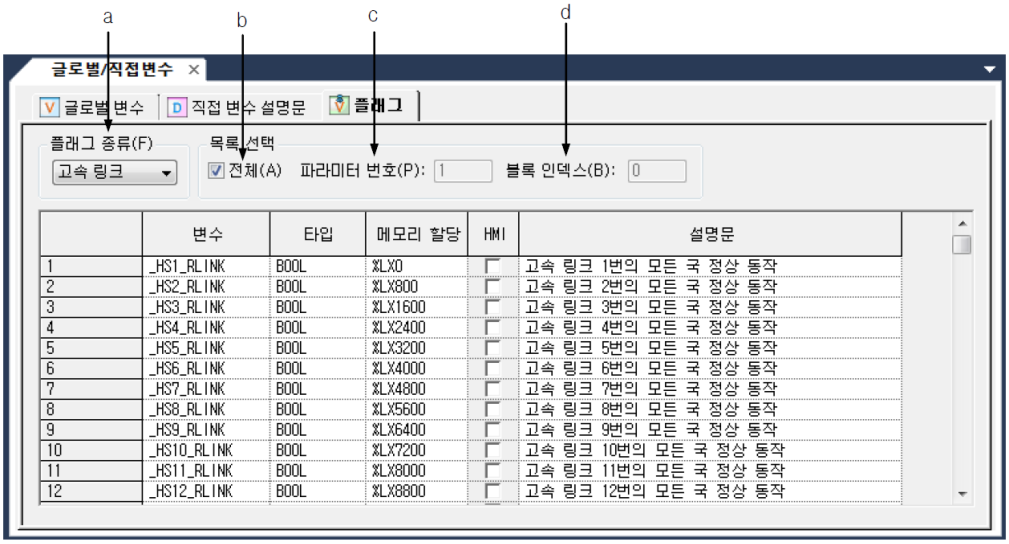


※ 직접 변수는 CPU 종류에 따라 달라집니다.

※ 비트 보기는 선언된 목록 보기가 체크 되면 활성화 됩니다.

※

▶ 플래그

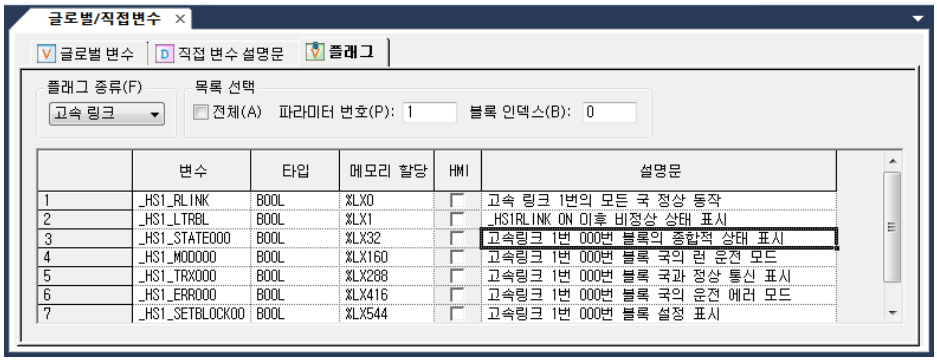


a. 플래그 종류: 플래그 종류(시스템, 고속 링크, P2P, PID) 중 하나를 선택합니다.

b. 전체: [플래그 종류]에서 선택된 플래그 목록 전체를 표시합니다. 시스템 플래그인 경우에는 전체 내용만 화면에 표시합니다. 전체 항목이 체크되지 않은 경우는 [파라미터 번호]와 [블록 인덱스]에 맞는 플래그 항목만 표시합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

c. 파라미터 번호: 고속 링크, P2P, PID 플래그인 경우에만 활성화 됩니다. 입력된 파라미터 번호의 플래그 항목만 보여줍니다. (예, 파라미터 번호 1 입력한 경우 아래 그림과 같습니다.)



d. 블록 인덱스: 고속 링크, P2P 플래그인 경우에만 활성화 됩니다. 입력된 블록 인덱스의 플래그 항목만 보여줍니다. (예, 블록 인덱스 120 입력한 경우 아래 그림과 같습니다.)

※ 플래그는 읽기 전용으로 플래그를 선언할 수 있습니다.

※ 플래그 종류는 PLC 종류에 따라 달라질 수 있습니다.

글로벌/직접변수 편집

현재 선언된 글로벌/직접변수 목록에서 변수종류, 변수, 타입, 메모리 할당, 초기값, 리테인, 사용 유무, 설명문 항목을 편집할 수 있습니다. 또한 프로그램에서 사용할 글로벌 변수를 글로벌/직접변수 목록에 추가합니다.

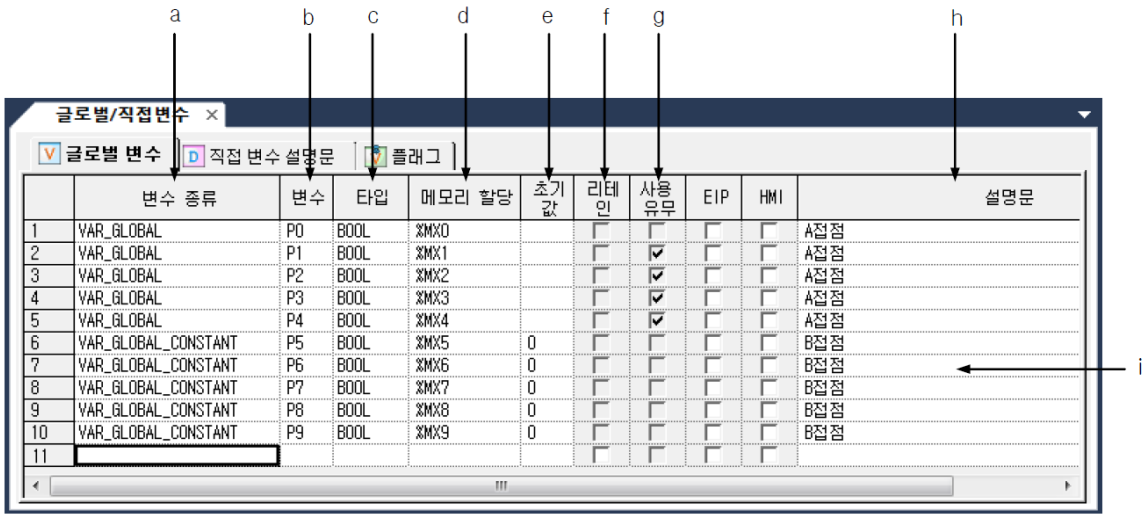
▶ 글로벌/직접변수 등록

프로그램에서 사용할 글로벌/직접변수를 등록합니다. 글로벌/직접변수 목록에 등록하기 위해서는 글로벌 변수에서 등록할 수 있습니다.

① 글로벌 변수에서 등록

글로벌 변수 목록에 변수를 추가하거나, 수정 또는 삭제할 수 있습니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.



	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무	EIP	HMI	설명문
1	VAR_GLOBAL	P0	BOOL	%MX0		☐	☐	☐	☐	A접점
2	VAR_GLOBAL	P1	BOOL	%MX1		☐	☑	☐	☐	A접점
3	VAR_GLOBAL	P2	BOOL	%MX2		☐	☑	☐	☐	A접점
4	VAR_GLOBAL	P3	BOOL	%MX3		☐	☑	☐	☐	A접점
5	VAR_GLOBAL	P4	BOOL	%MX4		☐	☑	☐	☐	A접점
6	VAR_GLOBAL_CONSTANT	P5	BOOL	%MX5	0	☐	☐	☐	☐	B접점
7	VAR_GLOBAL_CONSTANT	P6	BOOL	%MX6	0	☐	☐	☐	☐	B접점
8	VAR_GLOBAL_CONSTANT	P7	BOOL	%MX7	0	☐	☐	☐	☐	B접점
9	VAR_GLOBAL_CONSTANT	P8	BOOL	%MX8	0	☐	☐	☐	☐	B접점
10	VAR_GLOBAL_CONSTANT	P9	BOOL	%MX9	0	☐	☐	☐	☐	B접점
11						☐	☐	☐	☐	

a. 변수 종류: 변수 종류에는 VAR_GLOBAL, VAR_GLOBAL_CONSTANT만 올 수 있습니다.

b. 변수: 선언된 변수는 같은 이름으로 중복하여 선언할 수 있습니다.

- 첫 번째 문자로 숫자를 사용할 수 없습니다.

- 특수 문자를 사용할 수 없습니다. (단, '_'는 사용 가능합니다.)

- 빈 문자를 사용할 수 없습니다.

- 직접 변수와 같은 이름으로 사용할 수 없습니다. (예, MX0, WB0,...)

- 라인이 모두 비어있는 경우, 변수를 입력하면 타입이 디폴트로 BOOL가 표시됩니다.

c. 타입: 입력되는 타입은 총 23개로, 기본 타입 20개와 유도된 타입 3개로 설정되어 있습니다.

- 기본 타입 (20개): (BOOL, BYTE, WORD, DWORD, LWORD, SINT, INT, DINT, LINT, USINT, UINT, UDINT, ULINT, REAL, LREAL, TIME, DATE, TIME_OF_DAY, DATE_AND_TIME, STRING)

- 유도된 타입 (3개): ARRAY(예, ARRAY[0..6,0..2,0..4] OF BOOL) => 인자 제한 (3차까지), STRUCT(예, STRUCT명 표시) => STRUCT안에 STRUCT형태 못함, FB_INST(예, FB명 표시)

d. 메모리 할당: 직접 변수 (I, Q, M, R, W)를 사용하여 입력합니다.

e. 초기값: 초기값을 설정할 수 있습니다.

f. 리테인: 메모리 할당을 설정한 경우 리테인 열은 비 활성 됩니다.

- R, W: 항상 리테인 영역 입니다.

- M: 기본 파라미터 정보를 얻어 체크합니다.

- I, Q: 항상 비 리테인 영역 입니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

g. 사용유무: 선언한 변수의 사용 유무를 표시합니다.

h. 설명문: 모든 문자가 입력이 가능합니다.

- Ctrl + Enter 키를 사용하여 멀티 라인 입력이 가능합니다.

i. 라인 유효성: 글로벌 변수 창에 등록하려면 변수종류, 변수, 타입이 있어야 합니다.

- 글로벌 변수에 등록되지 않는 경우 분홍색으로 표시합니다.

※ 셀 편집 시, 에러가 발생하면 분홍색으로 표시합니다.

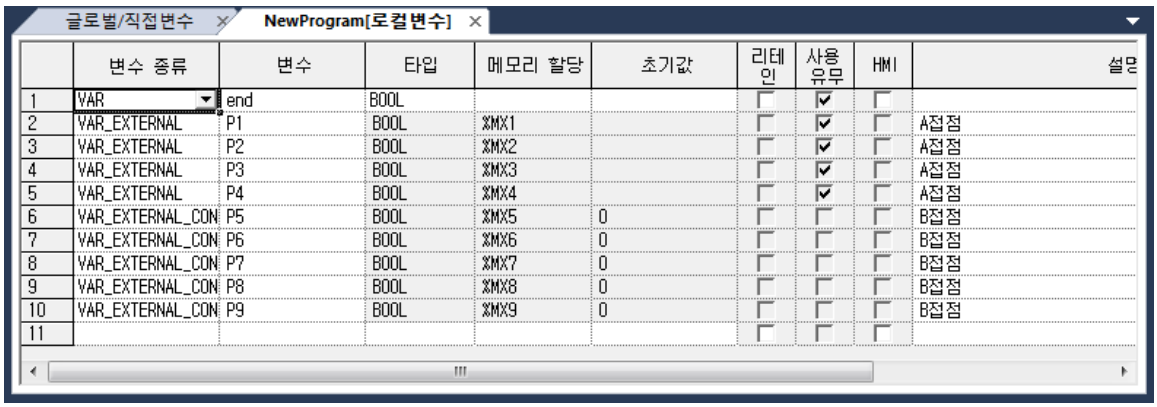
※ 셀 편집 시, ESC 키를 누르면 이전 값으로 복원됩니다

로컬 변수

로컬 변수는 프로그램에서 사용될 변수를 선언하거나, 선언된 변수 목록 전체를 변수 위주로 보여줍니다. 글로벌 변수에서 선언된 변수를 사용할 경우는 VAR_EXTERNAL, VAR_EXTERNAL_CONSTANT로 선언하여야 합니다.

▶ 로컬 변수

변수를 선언하고 선언된 로컬 변수 목록 전체를 보여줍니다.



	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무	HMI	설명
1	VAR	end	BOOL			☐	☑	☐	
2	VAR_EXTERNAL	P1	BOOL	%MX1		☐	☑	☐	A접점
3	VAR_EXTERNAL	P2	BOOL	%MX2		☐	☑	☐	A접점
4	VAR_EXTERNAL	P3	BOOL	%MX3		☐	☑	☐	A접점
5	VAR_EXTERNAL	P4	BOOL	%MX4		☐	☑	☐	A접점
6	VAR_EXTERNAL_CON	P5	BOOL	%MX5	0	☐	☐	☐	B접점
7	VAR_EXTERNAL_CON	P6	BOOL	%MX6	0	☐	☐	☐	B접점
8	VAR_EXTERNAL_CON	P7	BOOL	%MX7	0	☐	☐	☐	B접점
9	VAR_EXTERNAL_CON	P8	BOOL	%MX8	0	☐	☐	☐	B접점
10	VAR_EXTERNAL_CON	P9	BOOL	%MX9	0	☐	☐	☐	B접점
11						☐	☐	☐	

Copyright ⓒ 2021 LS Electric Co., Ltd.

4.5 LD 편집

LD 프로그램은 릴레이 논리 다이어그램에서 사용되는 코일이나 접점 등의 그래픽 기호를 통하여 PLC 프로그램을 표현합니다.

제한 사항

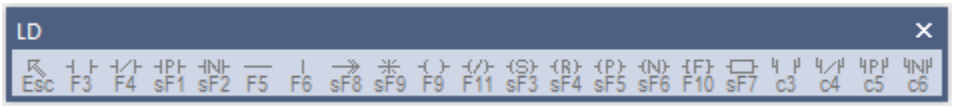
LD 프로그램 편집 시 다음과 같은 기능 제한이 있습니다.

목	내용	제한사항
최대 접점 개수	한 라인에 입력할 수 있는 최대 접점의 개수를 의미합니다.	31개
최대 라인 수	편집 가능한 최대 라인의 수를 의미합니다.	65,535 라인
최대 복사 라인 수	한 번에 복사할 수 있는 최대 라인 수를 의미합니다.	300 라인
최대 붙여넣기 라인 수	한 번에 붙여 넣을 수 있는 최대 라인 수를 의미합니다.	300 라인

프로그램 편집

▶ 편집 도구

LD 편집 요소의 입력은 LD 도구 모음에서 입력할 요소를 선택한 후 지정한 위치에서 마우스를 클릭하거나 단축키를 눌러 시작합니다.



기호	단축키	설명
Esc	Esc	선택 모드로 변경
F3	F3	평상시 열린 접점
F4	F4	평상시 닫힌 접점
sF1	Shift + F1	양 변환 검출 접점
sF2	Shift + F2	음 변환 검출 접점
F5	F5	가로선
F6	F6	세로선
sF8	Shift + F8	연결선
sF9	Shift + F9	반전 입력
F9	F9	코일
F11	F11	역 코일
sF3	Shift + F3	셋(latch) 코일

Copyright ⓒ 2021 LS Electric Co., Ltd.

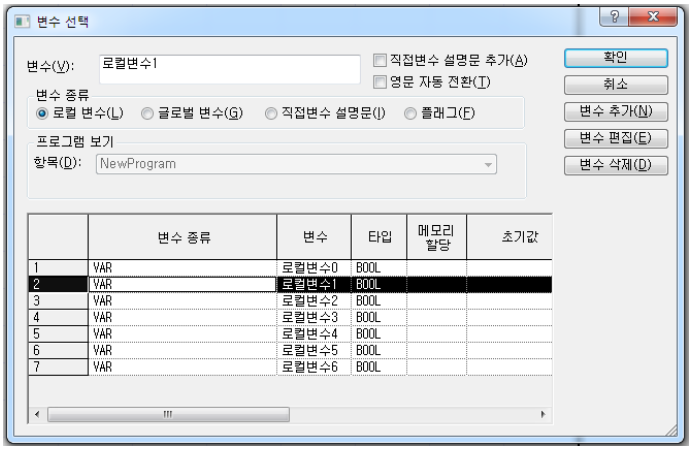
기호	단축키	설명
{R} sF4	Shift + F4	리셋(unlatch) 코일
{P} sF5	Shift + F5	양 변환 검출 코일
{N} sF6	Shift + F6	음 변환 검출 코일
{F} F10	F10	펑션/펑션 블록
sF7	Shift + F7	확장 펑션
c3	Ctrl+3	평상시 열린 OR 접점
c4	Ctrl+4	평상시 닫힌 OR 접점
c5	Ctrl+5	양 변환 검출 OR 접점
c6	Ctrl+6	음 변환 검출 OR 접점

▶ 변수/디바이스 입력

선택 된 영역 또는 커서 위치에 변수를 입력합니다.

[순서]

입력하고자 하는 위치로 커서를 이동 시킨 후 메뉴 [편집]-[변수선택/추가]를 선택합니다.



1. 로컬 변수: 선언된 로컬 변수 목록을 표시합니다.

2. 글로벌 변수: 선언된 글로벌 변수 목록을 표시합니다. EXTERNAL 변수로 등록 할 수 있습니다.
   (글로벌 변수는 편집, 삭제를 제공하지 않습니다.)

3. 직접 변수 설명문: 직접 변수 설명문 목록을 표시합니다.

4. 플래그: 목록에 플래그를 표시합니다. 플래그 항목에서 종류를 선택할 수 있습니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

▶ 펑션(블록)의 입력

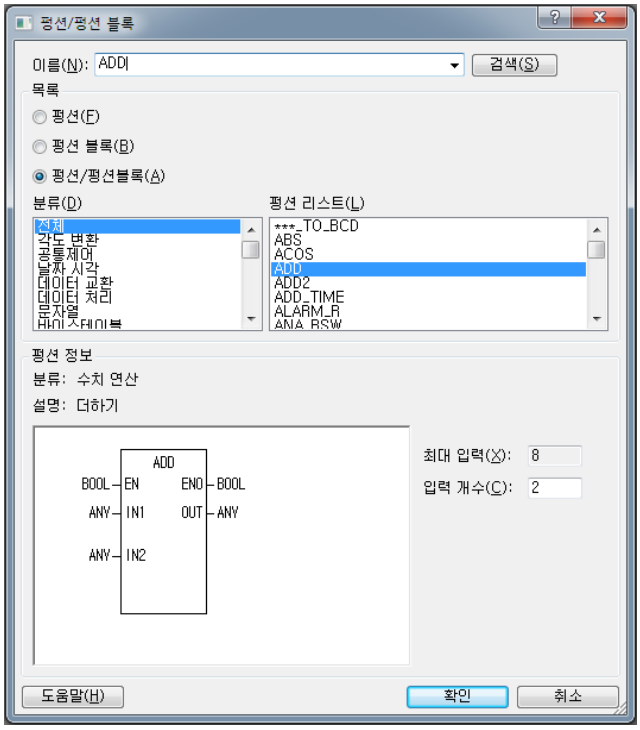
연산을 위한 펑션(블록)을 입력합니다.

[순서]

1. 도구 모음에서 펑션(블록)을 선택하고 편집 영역을 클릭합니다. 또는 펑션(블록) 입력 단축키(F10)를 누릅니다.

2. 펑션/펑션블록을 선택합니다.

3. 입력 개수를 설정 후 확인을 누릅니다.



Copyright ⓒ 2021 LS Electric Co., Ltd.

4.6 파라미터

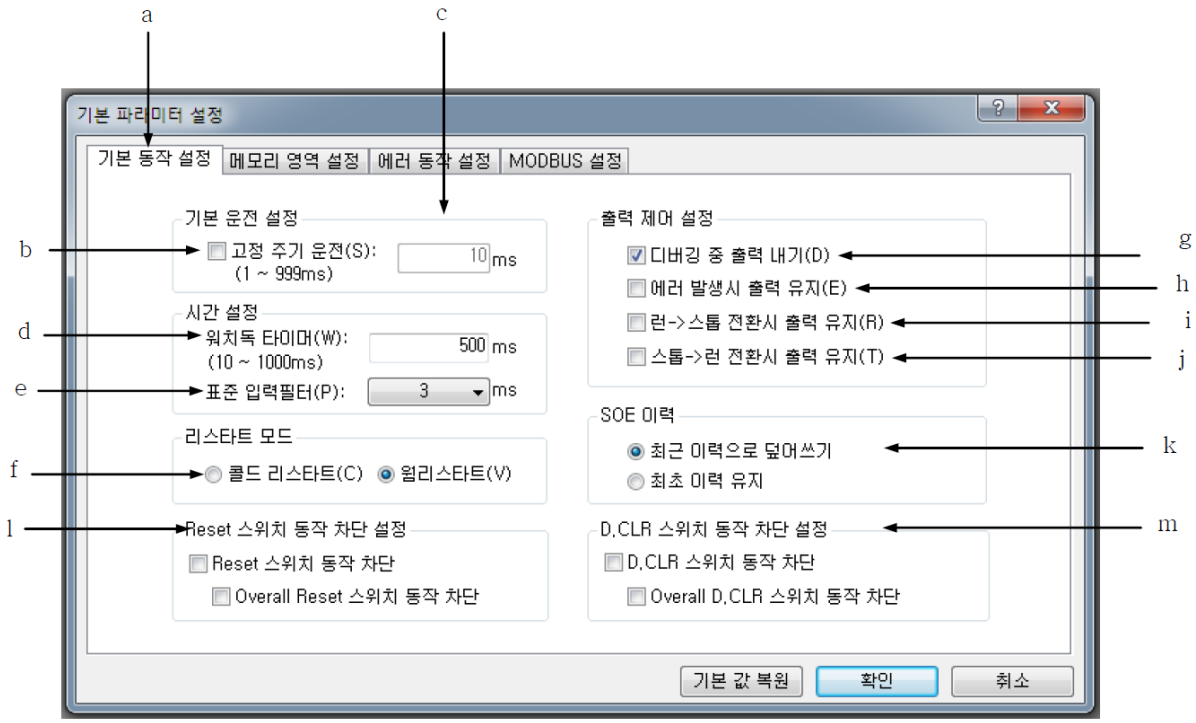
기본 파라미터

PLC 동작에 관계된 기본 파라미터를 설정합니다.

[순서]

1. 프로젝트 트리 [파라미터]-[기본 파라미터]를 두 번 누릅니다.

① 기본 동작 설정



a. 기본 동작 설정: [기본 파라미터] 정보 중 기본 운전, 시간, 리스타트 방법, 출력 제어 설정을 위한 탭입니다.

b. 고정 주기 운전: PLC 프로그램을 고정된 주기에 따라 동작을 시킬 것인지, 스캔 타임에 의해 동작 시킬 것인지를 결정합니다.

c. 고정 주기 운전 시간 설정: 고정 주기 운전 설정이 체크되어 있을 때 동작 시간을 사용자가 ms 단위로 입력합니다.

d. 워치독 타이머: 프로그램 오류에 의해 PLC가 멈추는 현상을 제거하기 위한 스캔 워치독 타이머의 시간 값 설정합니다.

e. 표준 입력필터: 표준 입력 값을 설정합니다.

f. 리스타트 모드: 리스타트 모드를 설정합니다. 콜드/웜 리스타트 중 하나를 선택합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

g. 디버깅 중 출력 내기: 디버깅 중에도 출력 모듈에 데이터를 정상적으로 출력할지를 결정합니다.

h. 에러 발생시 출력 유지: 에러나 특정한 입력이 발생될 때에도 모듈에 데이터를 정상적으로 출력할지를 결정합니다.

i. 런->스톱 전환시 출력 유지: PLC 동작 모드 RUN에서 STOP로 전환 중에 모듈에 데이터를 정상적으로 출력할지를 결정합니다.

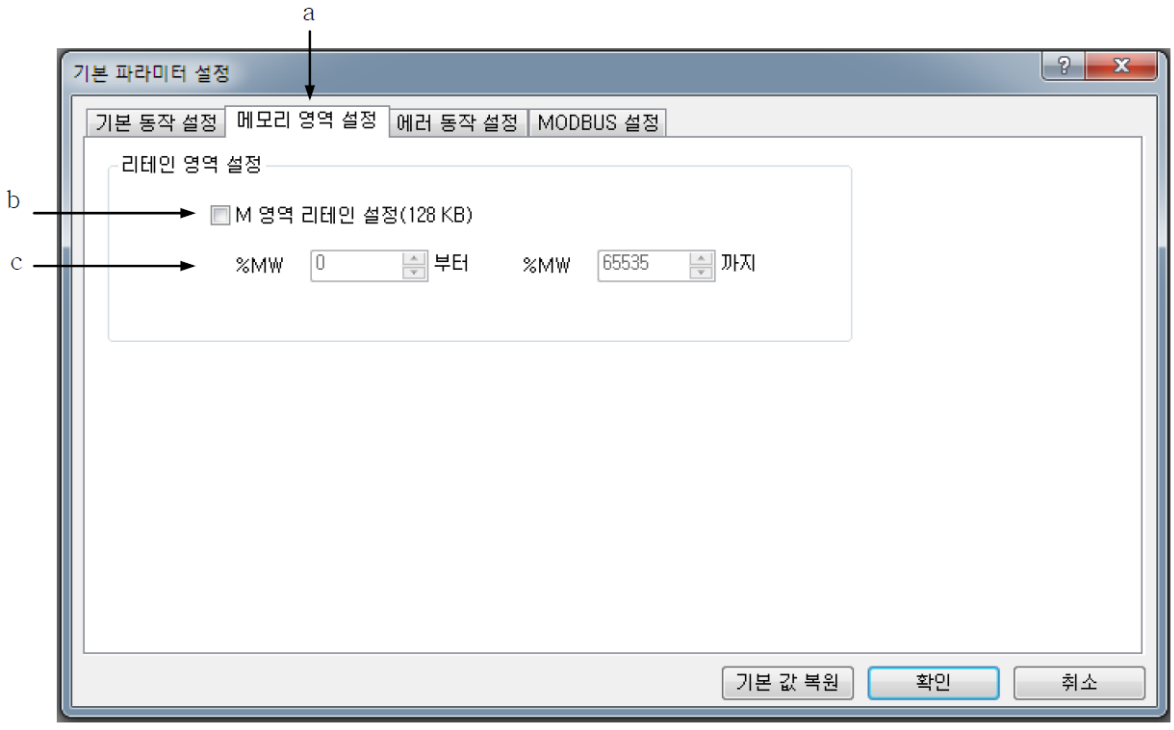
j. 스톱->런 전환시 출력 유지: PLC 동작 모드 STOP에서 RUN으로 전환 중에 모듈에 데이터를 정상적으로 출력할지를 결정합니다.

k. 이벤트 입력 모듈 전용 기능: 19장 이벤트 입력 모듈 전용 기능 참조

l. Reset 스위치 동작 차단 설정: CPU모듈의 RST(Reset) 스위치의 동작을 차단할 것인지 결정합니다. Overall Reset 동작 차단을 설정할 경우 Overall Reset 동작만 차단됩니다.

m. D.CLR 스위치 동작 차단 설정: CPU모듈의 D.CLR 스위치의 동작을 차단할 것인지 결정합니다. Overall D.CLR 동작 차단을 설정할 경우 Overall D.CLR 동작만 차단됩니다.

2. 컴파일 옵션 설정



a. 메모리 영역 설정: [기본 파라미터] 정보 중 메모리 영역을 설정합니다.

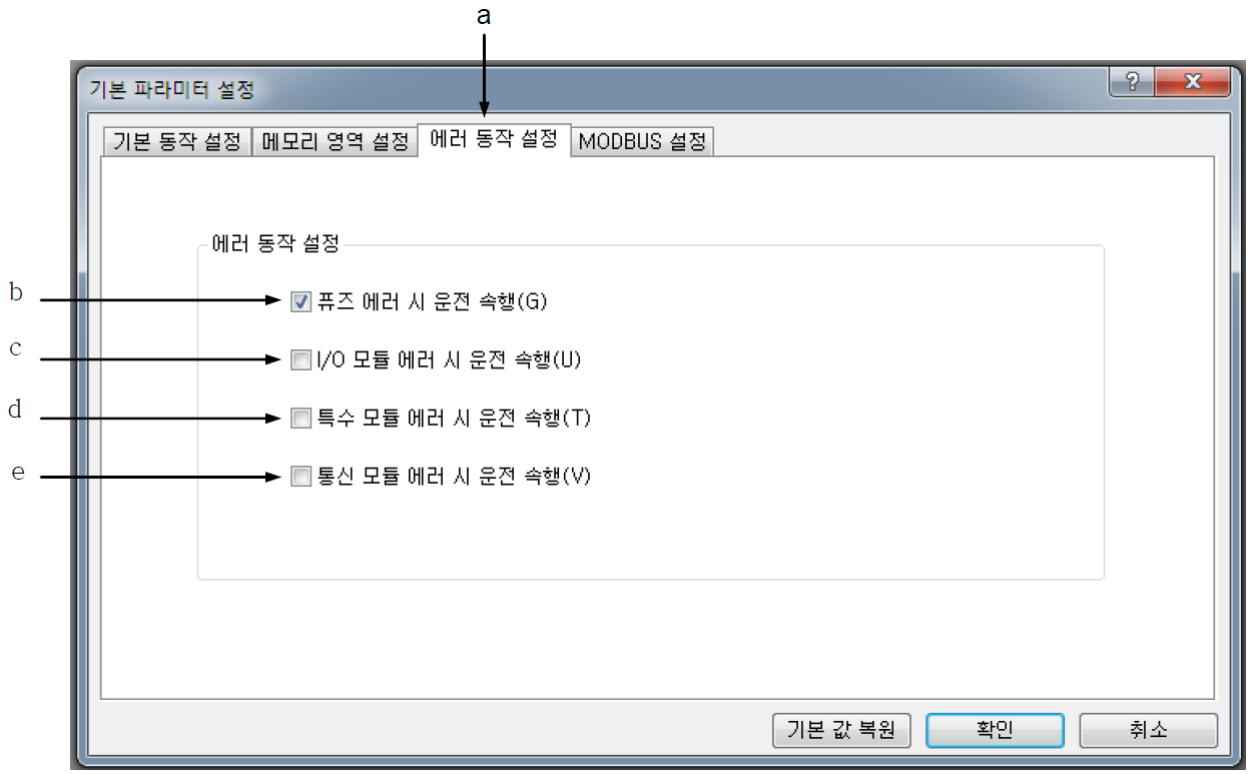
b. M영역 리테인 영역 설정: PLC 전원 투입 시 데이터를 보존할 M 영역(리테인 영역)을 설정합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

c. 데이터 보존 영역의 크기를 설정합니다. 디바이스 WORD 단위로 M 영역 크기 안에서 설정할 수 있습니다. M 영역으로 설정된 크기는 전체 M 영역 크기의 반[65,536]을 넘을 수 없습니다.

d. 프로그램 다운로드 시에 M영역을 초기화 하도록 설정합니다.

3. 에러 동작 설정



a. 에러 동작 설정: [기본 파라미터] 정보 중 PLC에 에러가 발생되었을 때 동작 방법 설정을 위한 탭입니다.

b. 이 옵션을 선택하면 PLC 동작 중 모듈의 퓨즈 연결 상태에 에러가 발생하였을 때에도 PLC가 계속 동작합니다.

c. 이 옵션을 선택하면 PLC 동작 중 I/O 모듈에 에러가 발생하였을 때에도 PLC가 계속 동작합니다.

d. 이 옵션을 선택하면 PLC 동작 중 특수 모듈에 에러가 발생하였을 때에도 PLC가 계속 동작합니다.

e. 이 옵션을 선택하면 PLC 동작 중 통신 모듈에 에러가 발생하였을 때에도 PLC가 계속 동작합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

리스타트 모드

리스타트 모드는 전원을 재 투입 하거나 또는 모드 전환에 의해서 RUN 모드로 운전을 시작할 때 변수 및 시스템을 어떻게 초기화한 후 RUN 모드 운전을 할 것인가를 설정하는 것으로 콜드, 웜의 2종류가 있으며 각 리스타트 모드의 수행 조건은 다음과 같습니다.

▶ 콜드 리스타트

1) 파라미터의 리스타트 모드를 콜드 리스타트로 설정 하는 경우 수행됩니다.

2) 초기값이 설정된 변수를 제외한 모든 데이터를 '0'으로 소거하고 수행합니다.

3) 파라미터를 웜 리스타트 모드로 설정해도 수행할 프로그램이 변경된 후 최초 수행 시는 콜드 리스타트 모드로 수행됩니다.

4) 운전중 수동 리셋 스위치를 누르면(온라인 리셋명령과 동일) 파라미터에 설정된 리스타트 모드에 관계없이 콜드 리스타트 모드로 수행됩니다.

▶ 웜 리스타트

1) 파라미터의 리스타트 모드를 웜 리스타트로 설정 하는 경우 수행됩니다.

2) 이전 값 유지를 설정한 데이터는 이전 값을 그대로 유지하고 초기값만 설정된 데이터는 초기값으로 설정합니다. 이외의 데이터는 '0'으로 소거합니다.

3) 파라미터를 웜 리스타트 모드로 설정해도, 데이터 내용이 비정상일 경우(데이터의 정전 유지가 되지 못함)에는 콜드 리스타트 모드로 수행됩니다.

I/O 파라미터

PLC의 슬롯에 사용할 I/O 종류를 설정하고, 해당 슬롯 별로 파라미터를 설정합니다.

[순서]

1. 프로젝트 트리 [파라미터]-[I/O 파라미터]를 선택합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

a. 모든 베이스: 베이스 모듈 정보와 슬롯 별 모듈 정보를 표시합니다. 슬롯에 모듈을 지정하지 않은 경우'디폴트'로 표시됩니다.

b. 설정된 베이스: 모듈이 선택된 베이스만 표시합니다.

c. 슬롯 정보: 베이스의 슬롯 별 모듈 종류를 편집하거나 표시합니다. 모듈 별 상세 정보 및 할당 정보가 표시됩니다.

d. 적용: 설정한 I/O 파라미터를 적용합니다.

e. 소비전류: 장착된 모듈들의 총 소비전류가 표시됩니다.

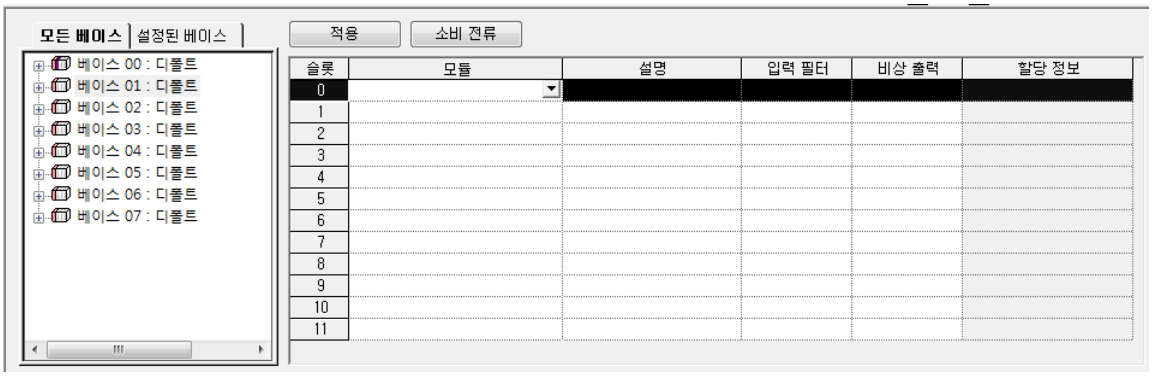
※ XGI 프로젝트 타입은 고정 할당 방식만을 지원합니다. 따라서, 할당 정보 열은 항상 비 활성화 상태로 표시됩니다.

슬롯별 모듈 설정

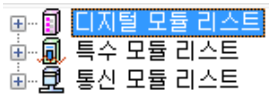
[순서]

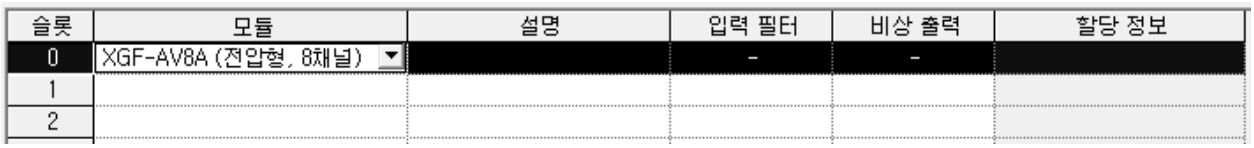
1. 슬롯 정보에서 모듈을 설정할 슬롯을 선택합니다.

2. 모든 열의 화살표를 선택하면, 모듈 선택 상자가 표시됩니다. 또는 마우스 오른쪽 버튼을 눌러 [편집]을 선택합니다.



3. 선택 상자를 눌러 모듈을 선택합니다.





4. 설명 열을 선택하고 오른쪽 마우스 버튼을 눌러 [편집] 항목을 선택합니다. 해당 슬롯에 대한 설명문을 입력합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

4.7. 온라인

PLC와 연결 되었을 때만 가능한 기능을 설명합니다.

접속 옵션

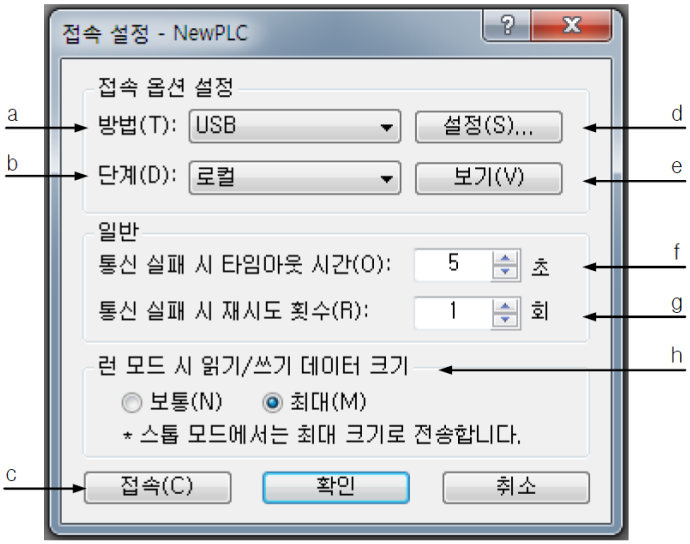
PLC와의 연결 네트워크 설정을 합니다.

▶ 로컬 접속 설정

로컬 접속 설정은 RS-232C 또는 USB 연결이 가능합니다.

[순서]

1. 메뉴 [온라인]-[접속 설정]을 선택합니다.



a. 접속 방법: PLC와 연결 시 통신 미디어를 설정합니다. RS-232C, USB, Ethernet, Modem으로 설정을 할 수 있습니다.

b. 접속 단계: PLC와의 연결 구조를 설정합니다. 로컬, 리모트 1단, 리모트 2단 연결 설정을 할 수 있습니다.

c. 접속: 설정된 접속 옵션 사항으로 PLC와 연결을 시도합니다.

d. 설정: 선택된 접속 방법에 따른 상세 설정을 할 수 있습니다.

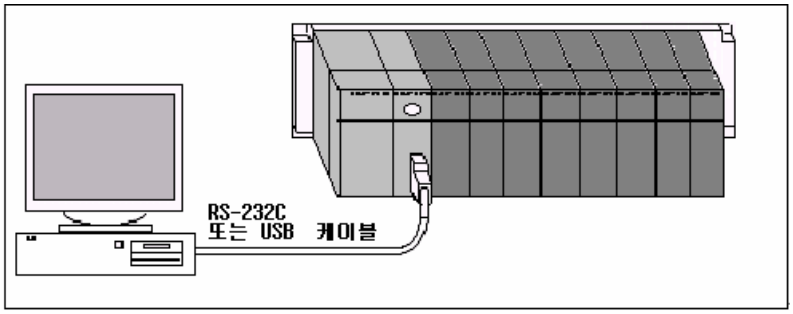
e. 보기: 전체적인 접속 옵션을 한 눈에 확인할 수 있습니다.

f. 타임 아웃 시간: 설정된 시간 내에 PLC와의 통신 연결을 재개하지 못할 경우 타임아웃이 발생하여 연결 재시도를 할 수 있습니다.

g. 재시도 횟수: PLC와의 통신 연결 실패 시 몇 회를 더 다시 통신 연결 할지를 설정합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

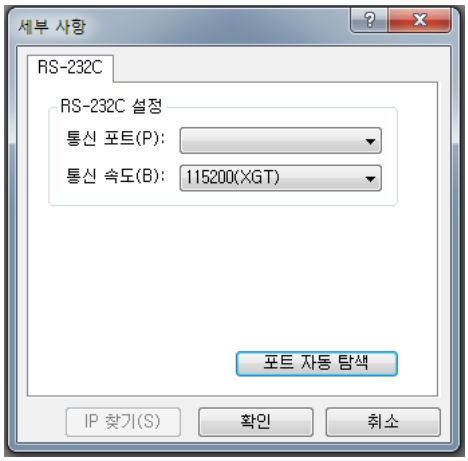
h. 런 모드 시 읽기/쓰기 데이터 크기: 데이터 전송 프레임의 크기를 설정합니다. 이 옵션은 PLC 운전모드가 런 일 때만 적용되며 그 외 운전모드는 최대 프레임 크기로 전송합니다.



① 로컬 RS-232C 연결

[순서]

1. 접속 방법을 RS-232C로 선택합니다.
2. 설정 버튼을 눌려 통신 속도 및 통신 COM 포트를 설정합니다.
3. 확인 버튼을 눌러 접속 옵션을 저장합니다.



XG5000에서 접속과 디바이스 모니터, 시스템 모니터에서의 접속이 하나의 PLC에 동시에 가능합니다. 단, 접속 옵션의 사항이 동일할 경우에만 가능합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

모드 전환

PLC의 운전 모드를 전환할 수 있습니다.

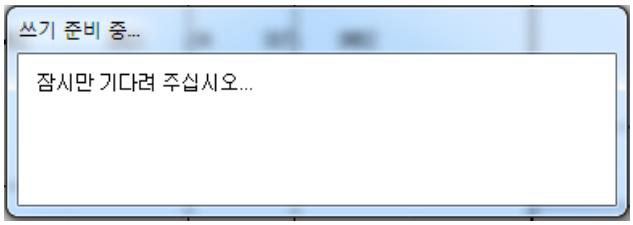
[순서]

1. 메뉴 [온라인]-[접속]을 선택하여 PLC와 연결합니다.
2. 메뉴 [온라인]-[모드 전환]-[런/스톱/디버그]를 선택합니다.
3. PLC의 운전 모드가 사용자가 선택한 운전 모드로 전환됩니다.

※ PLC의 리모트 딥 스위치가 ON 이어야 하고, 운전 모드 딥 스위치가 스톱이어야 합니다.

※ PLC내의 프로그램과 프로젝트의 프로그램이 같아야지만 디버그 모드로 전환할 수 있습니다.

※ 스톱 모드에서 런 모드로 전환하면 PLC 내부에서 프로그램을 실행 코드로 변환 중임을 표시하는 대화 상자가 나옵니다. 이 대화 상자는 프로그램의 크기에 따라 최대 30초 가량 닫히지 않을 수 있습니다.



※ 런 또는 디버그로 모드 전환 시 PLC에 에러가 발생한 경우는 런 또는 디버그 기능을 정상적으로 수행 할 수 없습니다. PLC의 에러를 해결한 후 운전 모드 전환을 하십시오.

※ 모드 전환 시 '확인 메시지를 안 보시려면 [도구]-[옵션]-[온라인 탭]의 'PLC 운전 모드 전환 시 메시지 보이기' 항목의 체크를 하지 않으면 됩니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

4.8. 모니터

모니터 공통

XG5000의 모니터 기능 중 공통적인 기능(모니터 시작/끝, 현재 값 변경, 모니터 일시 정지, 모니터 다시 시작, 모니터 일시 정지 설정)을 설명합니다.

▶ 모니터 시작/끝

[모니터 시작]

1. 메뉴 [온라인]-[접속] 항목을 선택하여 PLC와 온라인으로 연결합니다.
2. 메뉴 [모니터]-[모니터 시작/끝]을 선택하여 모니터를 시작합니다.
3. LD 또는 IL 프로그램이 활성화 되어 있으면 모니터 모드로 변경됩니다.

※ 모니터 시작 시 PLC의 프로그램과 XG5000의 프로그램이 불일치 하는 경우는 정확한 값이 모니터 되지 않을 수 있습니다.

[모니터 끝]

메뉴 [모니터]-[모니터 시작/끝] 항목을 선택하여 모니터를 정지합니다.

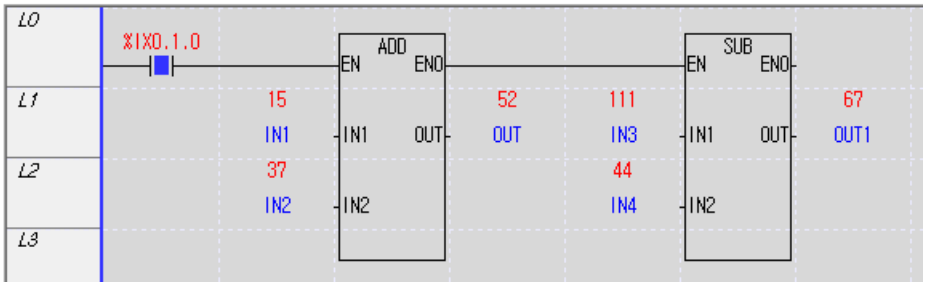
※ 이전에 모니터 시작이 되어 있으면 모니터 끝이 수행됩니다. 모니터를 수행하지 않았으면 모니터 시작이 수행됩니다.

▶ 현재 값 변경

모니터 중에 선택된 디바이스의 현재 값 또는 강제 I/O 설정을 변경할 수 있습니다.

[순서]

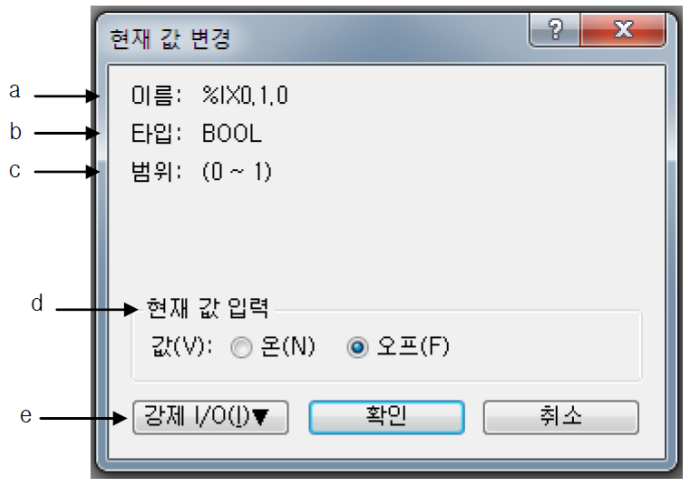
1. 메뉴 [온라인]-[접속] 항목을 선택하여 PLC와 온라인으로 연결합니다.
2. 메뉴 [모니터]-[모니터 시작] 항목을 선택하여 모니터를 수행합니다.
3. 프로그램 또는 변수 모니터 창에서 디바이스나 변수를 선택합니다.



Copyright ⓒ 2021 LS Electric Co., Ltd.

4. 메뉴 [모니터]-[현재 값 변경] 항목을 선택합니다.

5. 대화 상자에 현재 값을 입력 후 확인을 선택 시 현재 값이 변경됩니다.



a. 이름: 현재 값 변경 대상 변수의 이름입니다.

b. 타입: 현재 값 변경 대상 변수의 타입입니다.

c. 범위: 타입에 따른 현재 값의 입력 가능 범위입니다.

d. 현재 값 입력: 타입이 BOOL인 경우 변수의 On/Off를 설정합니다.

e. 강제 I/O: 변수가 "I/Q"영역이고 BOOL 타입인 경우 강제 I/O 설정을 가능하게 합니다.

※ 값의 초기 값은 변수의 디스플레이 타입에 따라 표시됩니다. 즉, 모니터 시 16진수로 표시되고 있으면 현재 값 변경은 16진수로 표시됩니다. 값 입력은 디스플레이 타입에 따라 입력하지 않아도 됩니다. 즉, 16진수로 표시되고 있을 때 부호 없는 10진수로 입력 가능합니다.확인 버튼을 누를 시 입력 값의 유효성 및 범위를 검사하여 에러 메시지가 발생할 수도 있습니다. 16진수로 입력 방법은 16#1234 같이 16#으로 시작합니다. STRING 타입인 경우 작은 따옴표('abcde') 사이에 현재 값(문자열)을 입력해야 합니다.

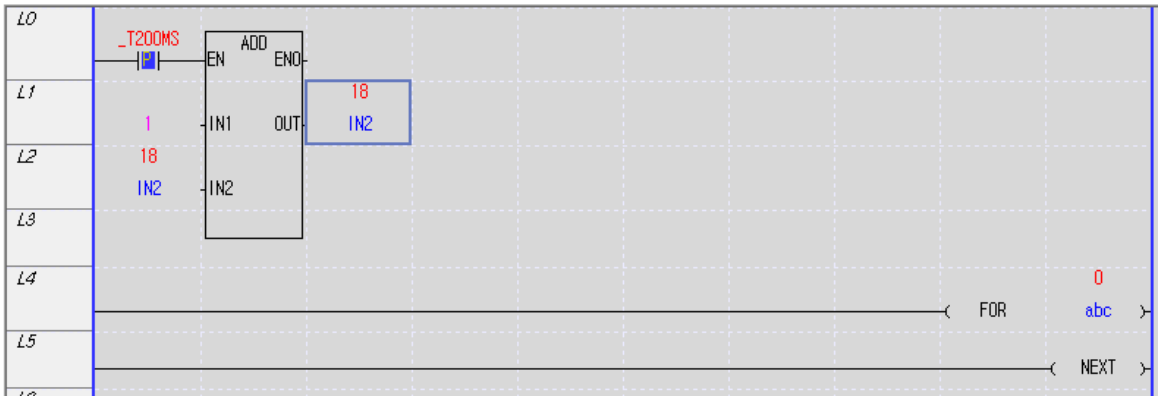
Copyright ⓒ 2021 LS Electric Co., Ltd.

LD 프로그램 모니터

XG5000이 모니터 상태에서 LD 다이어그램에 작성된 접점(평상시 열린 접점, 평상시 닫힌 접점, 양 변환 검출 접점, 음 변환 검출 접점), 코일(코일, 역 코일, 셋 코일, 리셋 코일, 양 변환 검출 코일, 음 변환 검출 코일) 및 펑션(블록)의 입출력 파라미터 등의 현재 값을 표시합니다.

[모니터 시작 순서]

1. 메뉴 [모니터]-[모니터 시작/끝] 항목을 선택합니다.
2. LD 프로그램이 모니터 모드로 변경됩니다.

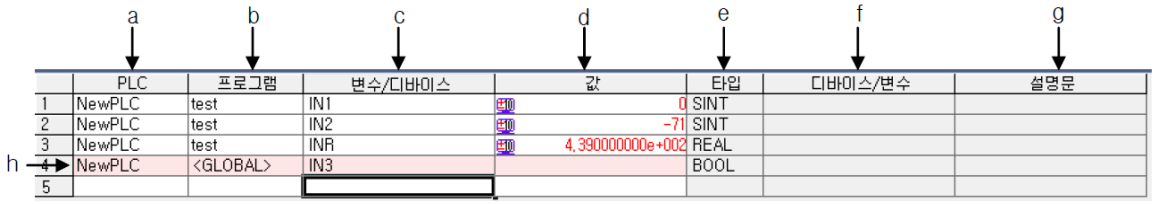


3. 현재 값 변경: 메뉴 [모니터]-[현재 값 변경] 항목을 선택합니다.

변수 모니터

특정 변수 또는 디바이스를 등록하여 모니터 할 수 있습니다.

[변수 모니터 창]



[변수 모니터 창 설명]

a. PLC: 등록 가능한 PLC의 이름을 보여줍니다. XG5000은 멀티 PLC 구성이 가능합니다. 그러므로 변수 모니터 창에서도 구별해 줍니다.

b. 프로그램: 등록 변수가 존재할 프로그램의 이름을 선택합니다.

c. 변수/디바이스: 변수 또는 디바이스 이름을 입력합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

d. 값: 모니터 시 해당 디바이스의 값을 표시합니다. 모니터 현재 값 변경을 통해 값을 변경할 수 있습니다.

e. 타입: 변수의 타입을 표시합니다.

f. 디바이스/변수: 메모리 할당이 되어 있으면 할당된 주소나 변수 이름을 보여줍니다. Enter 키 또는 마우스를 더블 클릭하면 로컬 변수 목록에서 변수를 선택할 수 있습니다.

g. 설명문: 변수 설명문을 표시합니다.

h. 에러 표시: 붉게 표시됩니다.

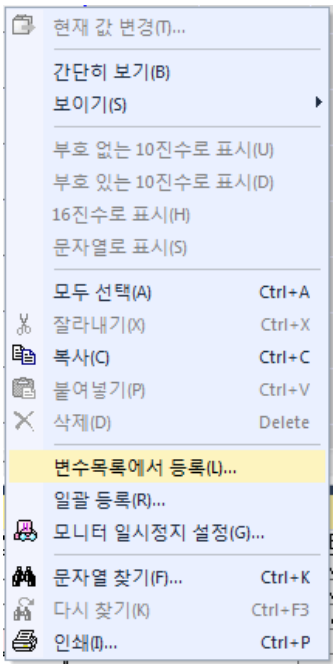
<에러 종류>

- PLC 이름, 디바이스, 타입 중 하나라도 입력하지 않은 경우
- 디바이스 주소가 잘못된 경우
- 프로그램 이름이 없거나 PLC에 존재하지 않는 경우
- 지원하지 않는 디바이스 타입이거나, 존재하지 않는 PLC 이름인 경우

▶ 변수 목록에서 모니터 등록

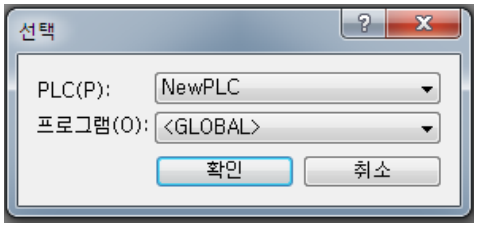
[순서]

1. 모니터 창에서 마우스 오른쪽 버튼을 눌러 [로컬변수에서 등록] 메뉴를 선택합니다.

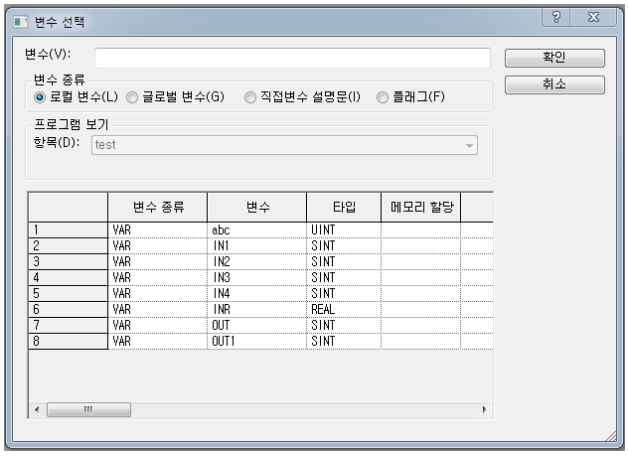


Copyright ⓒ 2021 LS Electric Co., Ltd.

2. 프로젝트 내에 포함된 PLC가 두 개 이상이거나 한 PLC에 프로그램이 두 개 이상일 경우 [선택] 대화 상자가 나옵니다. 등록할 PLC와 프로그램을 선택합니다.



3. 대화 상자가 나오고 변수 선택 후 변수를 변수 모니터 창에 등록합니다.

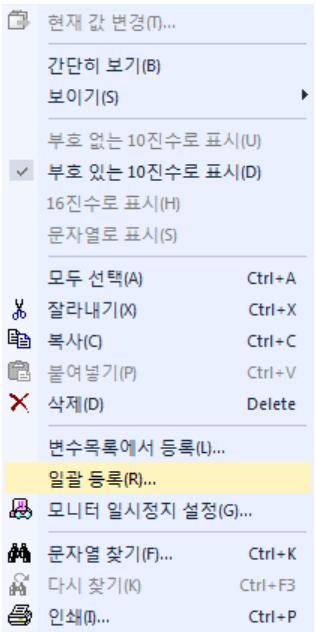


▶ 일괄 등록

같은 직접 변수를 다수 등록할 수 있습니다.

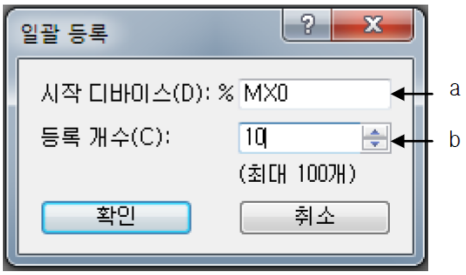
[순서]

1. 변수 모니터 창에서 마우스 오른쪽 버튼을 눌러 [일괄 등록] 메뉴를 선택합니다.



Copyright ⓒ 2021 LS Electric Co., Ltd.

2. [일괄 등록] 대화 상자가 나옵니다.



a. 시작 디바이스: 등록할 디바이스의 시작 주소를 입력합니다.
(디바이스 예: IX0.0.0, QB1.2.0, MW12 ...)

b. 등록 개수: 시작 디바이스에서부터 등록 개수만큼 등록하게 됩니다.

	PLC	프로그램	변수/디바이스	값	타입	디바이스/변수	설명문
1	NewPLC	test	IN1		SINT		
2	NewPLC	test	IN2		SINT		
3	NewPLC	test	INR		REAL	%MD2	시작
4	NewPLC	<GLOBAL>	%MX0		BOOL		
5	NewPLC	<GLOBAL>	%MX1		BOOL		
6	NewPLC	<GLOBAL>	%MX2		BOOL		
7	NewPLC	<GLOBAL>	%MX3		BOOL		
8	NewPLC	<GLOBAL>	%MX4		BOOL		
9	NewPLC	<GLOBAL>	%MX5		BOOL		
10	NewPLC	<GLOBAL>	%MX6		BOOL		
11	NewPLC	<GLOBAL>	%MX7		BOOL		
12	NewPLC	<GLOBAL>	%MX8		BOOL		
13	NewPLC	<GLOBAL>	%MX9		BOOL		
14							

※ 한 번에 일괄 등록할 수 있는 개수는 최대 100개까지 입니다.

※ 이전 등록된 항목과 동일한 항목을 등록할 경우에도 등록됩니다.

▶ 보기 기능

① 자세히 보기/간단히 보기

변수 모니터 창에서 화면에 최대한 많은 변수를 보기를 원할 때 유용한 기능입니다.

[순서]

1. 변수 모니터 창에서 마우스 우클릭 후 메뉴에서 [간단히 보기]를 선택합니다.

2. 그림과 같이 보이게 됩니다.

	변수/디바이스	값	변수/디바이스	값
1	IN1		IN2	
2	INR		%MX0	
3	%MX1		%MX2	
4	%MX3		%MX4	
5	%MX5		%MX6	
6	%MX7		%MX8	
7	%MX9			

변수/디바이스, 값 칼럼만 표시합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

※ 간단히 보기 모드일 때는 마우스 툴 팁을 지원합니다. 마우스 툴 팁은 PLC, 타입, 디바이스만 표시합니다. 단 변수가 선언되어 있으면 변수가 표시됩니다.

	변수/디바이스	값	변수/디바이스	값
1	IN1		IN2	
2	INR		%MX0	
3				

PLC: NewPLC
타입: BOOL
디바이스: %MX0

3. 다시 한번 [자세히 보기]를 선택 시 그림과 같이 여러 열로 보입니다.

	PLC	프로그램	변수/디바이스	값	타입	디바이스/변수	설명문
1	NewPLC	test	IN1		SINT		
2	NewPLC	test	IN2		SINT		
3	NewPLC	test	INR		REAL	%MD2	시작
4	NewPLC	<GLOBAL>	%MX0		BOOL		
5	NewPLC	<GLOBAL>	%MX1		BOOL		
6	NewPLC	<GLOBAL>	%MX2		BOOL		
7	NewPLC	<GLOBAL>	%MX3		BOOL		
8	NewPLC	<GLOBAL>	%MX4		BOOL		
9	NewPLC	<GLOBAL>	%MX5		BOOL		
10	NewPLC	<GLOBAL>	%MX6		BOOL		
11	NewPLC	<GLOBAL>	%MX7		BOOL		
12	NewPLC	<GLOBAL>	%MX8		BOOL		
13	NewPLC	<GLOBAL>	%MX9		BOOL		
14							

② 보이기 기능

사용자의 기호에 따라 보고 싶은 칼럼을 선택할 수 있습니다.

[순서]

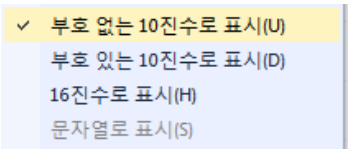
1. 변수 모니터 창에서 마우스 오른쪽 버튼에 의해서 발생되는 메뉴에서 [보이기]-[칼럼 이름(PLC, 프로그램, 변수/디바이스, 값, 디바이스/변수, 설명문)]을(를) 선택합니다.

③ 디스플레이

변수 모니터에 등록된 디바이스의 모니터 값의 표현 방법을 바꿀 수 있습니다.

[순서]

1. 변수 모니터 창에서 마우스 오른쪽 버튼을 눌러 발생되는 메뉴에서 부호 없는 10진수로, 부호 있는 10진수로, 16진수로, 문자열으로 표시를 선택합니다.



Copyright ⓒ 2021 LS Electric Co., Ltd.

5장. 프로그래밍

Copyright ⓒ 2021 LS Electric Co., Ltd.

5.1. 시퀀스 프로그램

시퀀스 연산자

XGI PLC의 시퀀스 연산자는 접점(Contact), 코일(Coil), 점프(Jump) 등이 있습니다.

▶ 입력 접점의 종류 및 기능

XGI PLC 입력 접점의 종류 및 기능은 다음과 같습니다.

접점	이름	기능
┤├	평상시 열린 접점	BOOL변수의 상태가 ON일 때에 해당 접점 연결
┤/├	평상시 닫힌 접점	BOOL변수의 상태가 OFF일 때에 해당 접점 연결
┤P├	양 변환 검출 접점	BOOL변수의 값이 OFF→ON으로 변화하는 순간 해당 접점을 현재1 스캔 동안 연결
┤N├	음 변환 검출 접점	BOOL변수의 값이 ON→OFF으로 변화하는 순간 해당 접점을 현재1 스캔 동안 연결

▶ 출력 코일의 종류 및 기능

XGI PLC 출력 코일의 종류 및 기능은 다음과 같습니다.

코일	이름	기능
-( )-	출력 코일	왼쪽에 있는 연결선의 상태를 지정된 BOOL변수 접점으로 출력
-(/)-	반전 코일	왼쪽에 있는 연결선의 상태를 반전하여 지정된 BOOL변수 접점으로 출력
-(P)-	양 변환 검출 코일	왼쪽 연결선 상태가 전 스캔 Off에서 현재 스캔 On시, 지정된 BOOL 변수 출력 접점을 현재 1 스캔 동안 ON
-(N)-	음 변환 검출 코일	왼쪽 연결선 상태가 전 스캔 On에서 현재 스캔 Off시, 지정된 BOOL 변수 출력 접점을 현재 1 스캔 동안 ON
-(S)-	셋 코일	왼쪽 연결선 상태가 1회 ON되었다가 OFF 되어도, 지정된 BOOL 변수 출력 접점은ON 상태를 유지
-(R)-	리셋 코일	왼쪽 연결선 상태가 ON 시, 지정된 BOOL 변수 출력 접점의 On상태를 Off(리셋) 시킴

Copyright ⓒ 2021 LS Electric Co., Ltd.

▶ 연산회로의 반전

입력 접점의 상태를 연산하여 그 결과를 반전 하고자 할 경우 사용되는 연산회로 반전 기능이 있습니다.

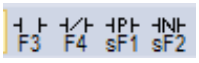
연산자	이름	기능
	NOT	현재까지 연산된 결과를 반전함. (1->0), (0->1)

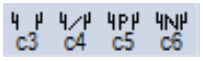
▶ 기타 시퀀스 연산자 (확장명령어)

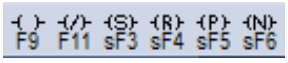
연산자	이름	기능
BREAK	브레이크	FOR~NEXT를 회전도중에서 탈출할 때 사용
CALL	서브루틴 콜	메인 프로그램 연산 도중 서브루틴 프로그램 호출
END	엔드	프로그램 연산 종료
FOR	포	FOR~NEXT를 N회 실행할 경우 반복문의 시작을 알림
NEXT	넥스트	FOR~NEXT반복문의 끝을 알림
INIT_DONE	이닛던	초기화 태스크의 종료를 알려주는 플래그
JMP	점프(Jump)	지정된 LABLE 위치로 연산 이동
SBRT	서브루틴	콜에 의해 실행될 프로그램 위치
RET	리턴(Return)	서브루틴 연산 완료 후 메인 프로그램으로 복귀

▶ 시퀀스 연산자 기호

XG5000에서 사용되는 입력 또는 출력 접점의 아이콘 표시는 다음과 같습니다. 표시하단에 있는 F3, F4등의 표기는 펑션키를 의미합니다.

① 입력접점 표시용 아이콘: 

② 논리합용 입력접점 표시용 아이콘: 

③ 출력접점 표시용 아이콘: 

④ 확장 명령어 표시용 아이콘:

Copyright ⓒ 2021 LS Electric Co., Ltd.

▶ 펑션 또는 펑션블록 입력용 연산자 기호

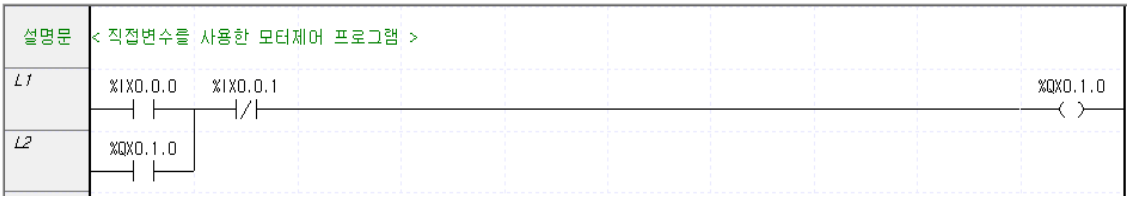
GMWIN에서는 펑션과 펑션블록이 도구모음에서 [F],[FB]로 나뉘어 있었지만 XG5000에서는 [F] 하나의 펑션키로 사용되고 펑션키를 선택 후에 펑션과 펑션블록으로 구분하여 입력할 수 있습니다.

입력 접점 및 출력 코일 프로그램

▶ 직접 변수 프로그램

직접 변수를 사용하여 MASTER-K등과 같은 방식으로 모터 제어 프로그램을 작성한 예입니다.

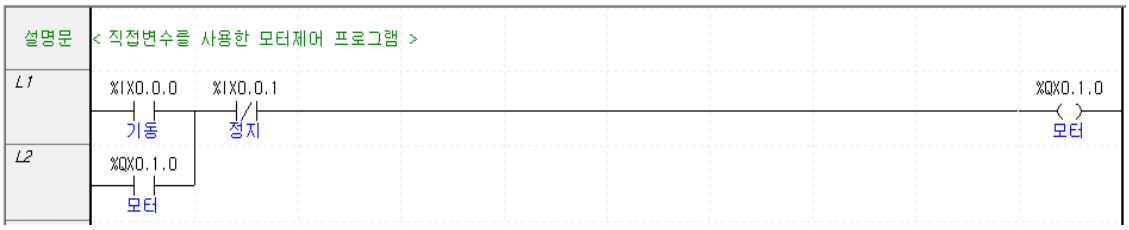
① 직접 변수를 사용한 모터 제어 프로그램



※ 직접 변수를 사용하면 변수 선언이 불필요 하므로 지역 변수 목록에 포함되지 않습니다.

② 직접 변수를 사용하고 설명문 단 모터제어 프로그램

XGI PLC는 **직접 변수**를 사용했을 경우 설명문(코멘트)을 달 수 있습니다.

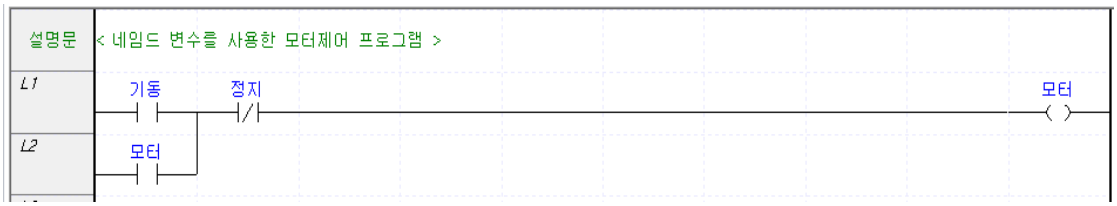


Copyright ⓒ 2021 LS Electric Co., Ltd.

▶ 네임드 변수 프로그램

네임드 변수를 사용하여 모터 제어 프로그램을 작성한 예입니다.

① 네임드 변수를 사용한 모터 제어 프로그램



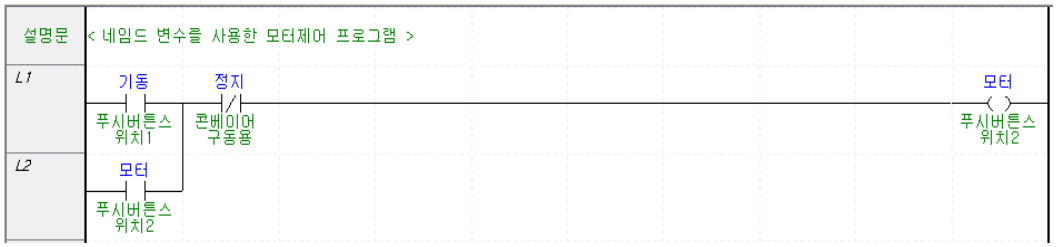
	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무	HMI	설명문
1	VAR	기동	BOOL	%IX0.0.0		☐	☑	☐	푸시버튼스위치1
2	VAR	모터	BOOL	%QX0.1.0		☐	☑	☐	푸시버튼스위치2
3	VAR	정지	BOOL	%IX0.0.1		☐	☑	☐	콘베이어 구동용

※ 변수명을 지정하고 메모리 할당을 사용자 정의로 한 경우

사용자 정의 메모리 할당은 직접변수 선언 방법과 동일한 방법으로 표현

② 네임드 변수를 사용하고 설명문을 단 모터 제어 프로그램

네임드 변수로 변수 선언을 하고 추가로 설명문(코멘트)을 달 수 있습니다.

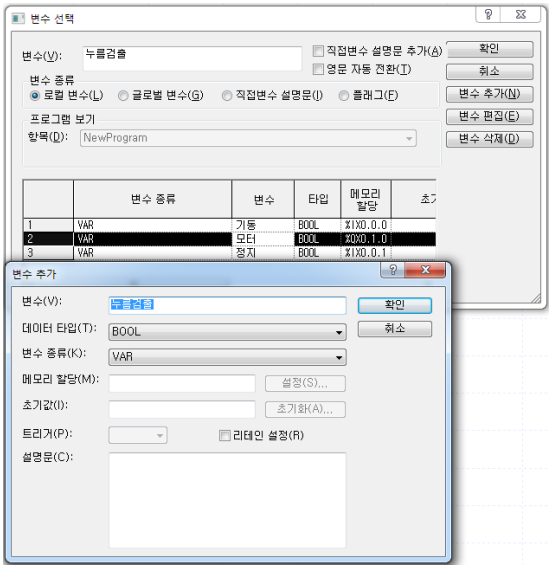


	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무	HMI	설명문
1	VAR	기동	BOOL	%IX0.0.0		☐	☑	☐	푸시버튼스위치1
2	VAR	모터	BOOL	%QX0.1.0		☐	☑	☐	푸시버튼스위치2
3	VAR	정지	BOOL	%IX0.0.1		☐	☑	☐	콘베이어 구동용
4						☐	☐	☐	

※ 네임드 변수명을 사용하고 설명문을 사용한 경우

변수의 설명문란에 커서를 이동하여 설명문을 입력합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.



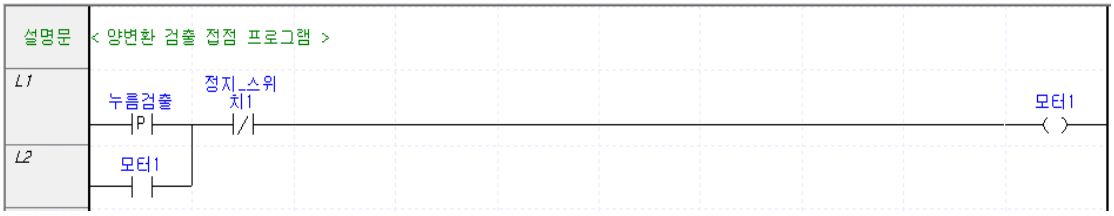
변환 검출 접점, 코일 프로그램

▶ 양 변환 검출 접점 및 음 변환 검출 접점 프로그램

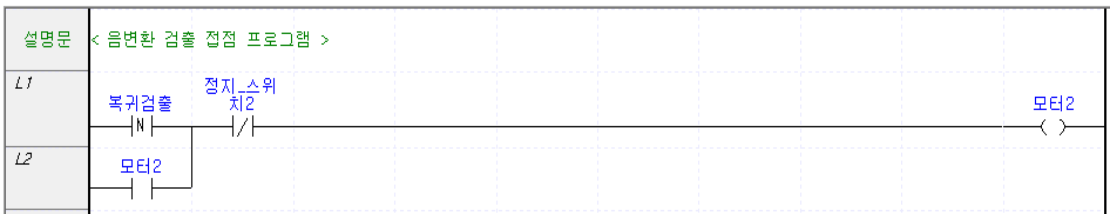
1. 누름 검출이 0에서 1로 변하는 순간을 검출하여 오른쪽 연결선이 해당 스캔에서 다음 1스캔 동안만 연결됩니다.

2. 누름 검출에 병렬 점속된 램프1을 자기유지 회로라고 합니다.

3. 점등된 램프1은 소등 스위치1에 의해서 소등됩니다.



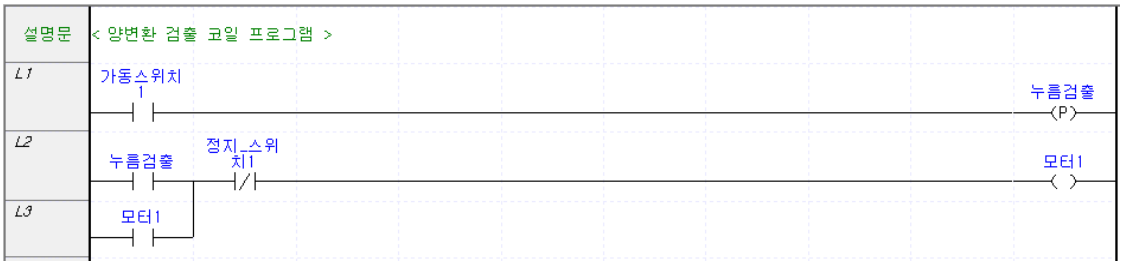
4. 복귀 검출이 0에서 1로 변하는 순간을 검출하여 오른쪽 연결선이 해당 스캔에서 다음 1스캔 동안만 연결됩니다.



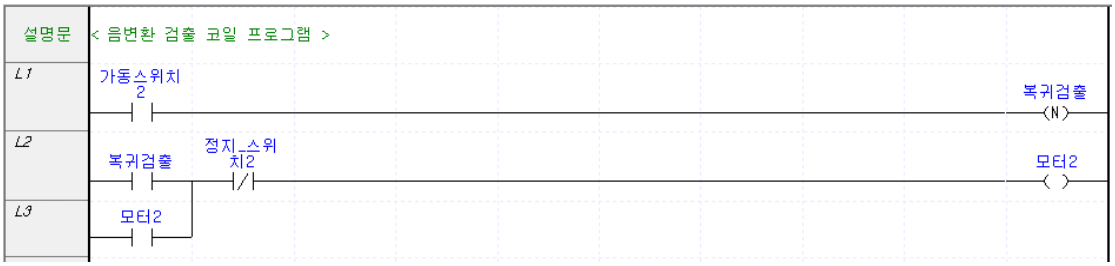
Copyright ⓒ 2021 LS Electric Co., Ltd.

▶ 양 변환 검출 코일 및 음 변환 검출 코일 프로그램

1. 기동스위치1을 누르는 순간(접점 상태가0(off)에서1(on)으로 변환) 모터1은 기동되고 정지스위치1에 의해 정지됩니다.



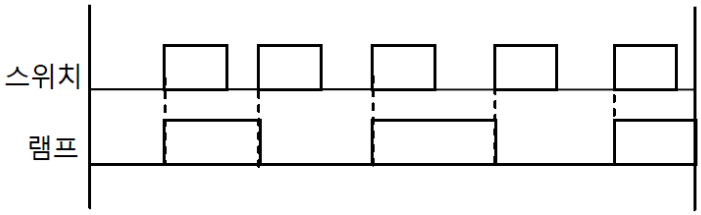
2. 기동스위치2를 누르는 순간(접점 상태가1(on)에서0(off)으로 변환) 모터2는 기동되고 정지스위치2에 의해 정지됩니다.



▶ 다이내믹 플립플롭

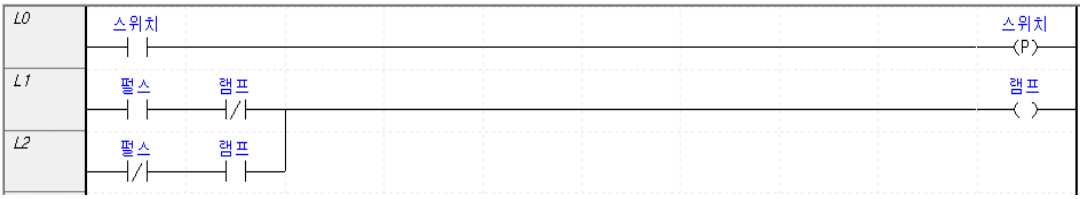
스위치를 한번 OFF →ON 하면 램프가 ON되고 다시 OFF→ON하면 램프가 OFF됩니다.

타임차트



프로그램 및 변수 설정

	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무	HMI	설명문
1	VAR	램프	BOOL	%IX0.0.0		☐	☑	☐	LED
2	VAR	스위치	BOOL	%IX0.0.1		☐	☑	☐	신호입력
3	VAR	펄스	BOOL			☐	☑	☐	펄스



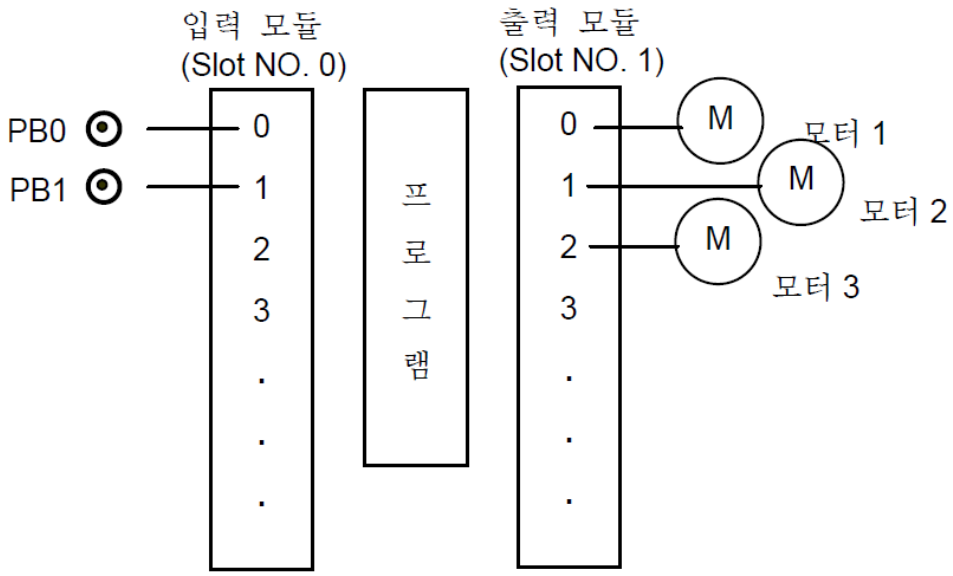
Copyright ⓒ 2021 LS Electric Co., Ltd.

▶ 모터의 기동 수 제어

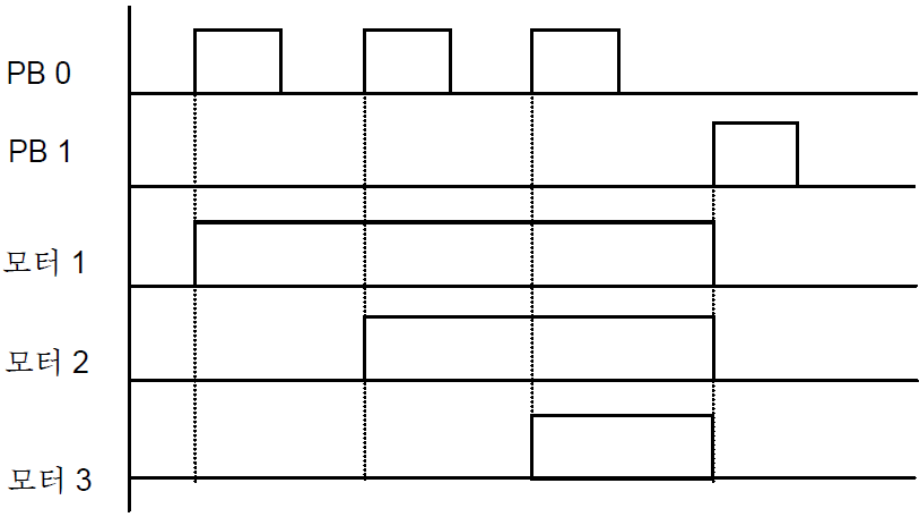
순간 접촉 푸쉬 버튼 PB0을 첫 번째 누르면 모터1이 ON, 두 번째 누르면 모터2가 ON, 세 번째 누르면 모터3이 ON 됩니다. 결국 순간 접촉 푸쉬 버튼 PB0를 세 번 누르면 세대의 모터가 모두 기동하게 됩니다. 순간 접촉 푸쉬 버튼 PB1을 누르면 모든 모터의 기동이 중지됩니다.

본 예제는 PLC시퀀스 의 정확한 이해로 작성할 수 있습니다. 직렬처리방식과 입출력 리프레시의 관계를 고려하여야만 정확한 결과를 나타낼 수 있습니다.

시스템도



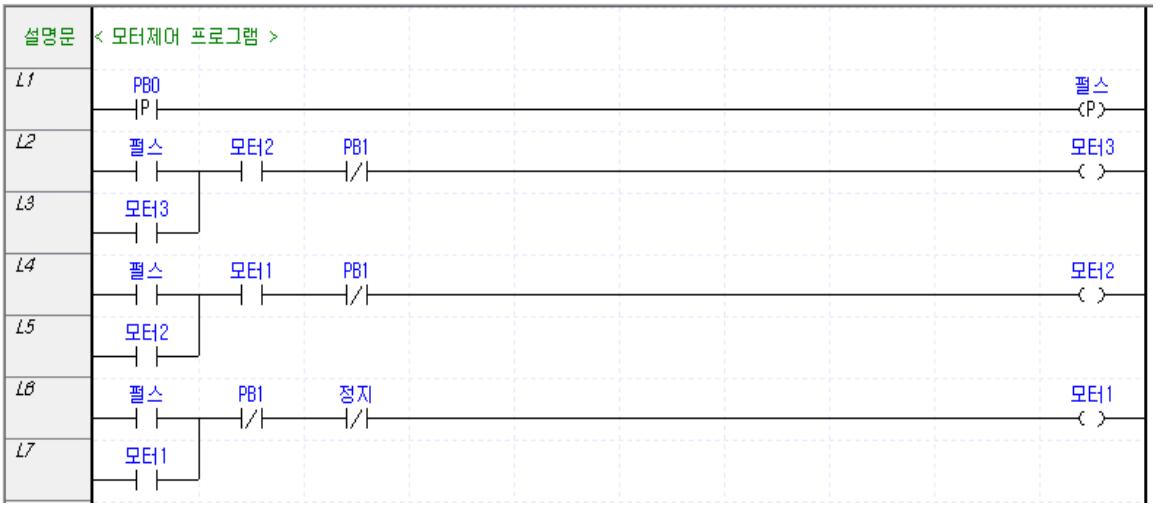
타임차트



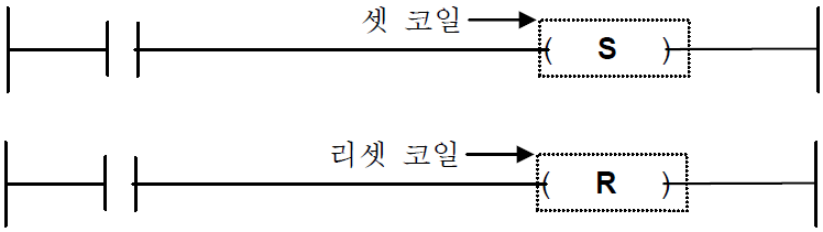
Copyright ⓒ 2021 LS Electric Co., Ltd.

프로그램 및 변수 설정

	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무	HMI	설명문
1	VAR	PB0	BOOL	%IX0.0.0		☐	☑	☐	
2	VAR	PB1	BOOL	%IX0.0.1		☐	☑	☐	
3	VAR	모터1	BOOL	%QX0.1.0		☐	☑	☐	
4	VAR	모터2	BOOL	%QX0.1.1		☐	☑	☐	
5	VAR	모터3	BOOL	%QX0.1.2		☐	☑	☐	
6	VAR	정지	BOOL			☐	☑	☐	
7	VAR	펄스	BOOL			☐	☑	☐	



▶ **셋 및 리셋 코일 프로그램**



구분	사용 가능 영역
S	%Q, %M
R	%Q, %M

① 셋 코일

입력 조건이 ON되면 지정된 비트 영역이 ON 됩니다. 지정된 비트 영역이 ON된 후 입력 조건이 OFF되어도 지정된 비트 영역은 ON 상태를 유지합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

② 리셋 코일

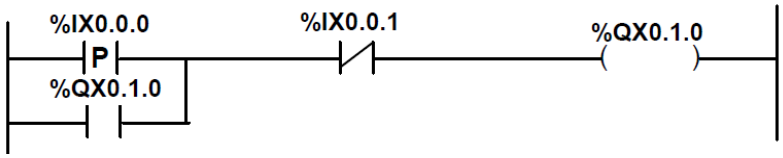
리셋 코일의 입력 조건이 ON 되었을 때, 리셋 코일로 지정된 비트 영역이 ON 상태이면 OFF상태로 만듭니다.

리셋 코일의 입력 조건이 ON 되었을 때, 리셋으로 지정된 비트 영역이 OFF 상태이면 아무런 변화도 일어나지 않습니다.

▶ 자기 유지 회로

푸시 버튼 스위치 입력 조건을 받아 출력 코일을 ON시킨 후 푸시 버튼에서 손을 떼어도 계속 출력을 ON으로 유지 시킬 수 있는 방법은 자기 유지 회로 방식과 SET/RESET 코일을 이용하는 두 가지 방법이 있습니다.

자기 유지 회로란 출력 접점을 한 번 작동시킨 후 그 출력 접점의 ON/OFF 정보를 다시 자기 입력으로 받는 방식입니다.



셋 및 리셋 프로그램 예

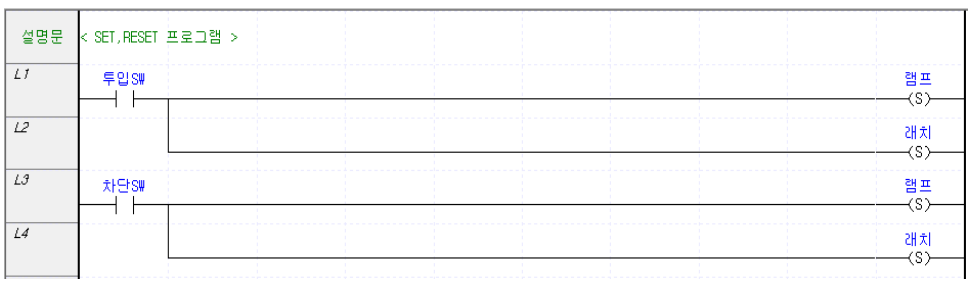
셋 코일은 자기유지 기능을 갖고 있기 때문에 코일 출력이 한번이라도 셋 되고 나면

'차단' 입력이 들어올 때까지 셋 되어 있는 상태를 유지합니다.

정전 후 복전 시 변수설정에서 리테인 설정여부에 따라 동작이 달라 집니다.

기본파라미터 설정에서 리스타트 모드가 웜 리스타트로 설정되고 리테인으로 설정된 변수 '래치'는 정전 후(PLC 전원 오프) 복전(PLC전원 ON)시에도 정전이전의 상태를 그대로 유지합니다.

	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무	HMI	설명문
1	VAR	래치	BOOL			☑	☑	☐	
2	VAR	램프	BOOL	%QX0.1.0		☐	☑	☐	
3	VAR	차단SW	BOOL	%IX0.0.1		☐	☑	☐	
4	VAR	투입SW	BOOL	%IX0.0.0		☐	☑	☐	
5						☐	☐	☐	



Copyright ⓒ 2021 LS Electric Co., Ltd.

5.2. 펑션 프로그램

펑션 프로그램

▶ 펑션(Function)과 펑션 블록(Function Block)

XGI 시리즈 PLC에서 사용되는 언어 구성체는 크게 펑션과 펑션 블록으로 구분됩니다.

① 펑션은 입력에 대한 연산 결과를 1 스캔에 즉시 출력합니다.

② 펑션은 출력이 하나입니다.

펑션은 1스캔에 입력을 받아 동일 스캔에 연산을 실행하여 그 결과를 만들어 내는 언어 구성체입니다.

① 펑션 블록은 여러 스캔에 걸쳐 누계된 연산 결과를 출력합니다.

② 펑션 블록은 출력이 여러 개가 될 수 있습니다.

펑션 블록은 여러 스캔에 걸쳐 누계된 연산 결과를 출력하므로, 연산 중 누계되는 데이터를 보관하기 위한 내부 메모리가 필요합니다. 따라서 펑션 블록은 사용하기 전에 인스턴스 변수를 선언 합니다. 인스턴스 변수는 펑션 블록 내에서 사용하는 변수들의 집합입니다.

펑션과 펑션 블록의 차이점을 표로 나타내면 다음과 같습니다.

구분	펑션	펑션 블록
입력의 수	1개 이상(최대 8개)	2개 이상
출력의 수	오직 1개	1개 이상
연산 시간	1스캔에 결과 출력	여러 스캔 누계 결과 출력
데이터	입, 출력 데이터를 모두 반드시 지정	입력 데이터는 반드시 지정하고, 출력 데이터는 생략 가능함
데이터 타입	입력 변수와 출력 변수의 모든 데이터 타입이 동일	변수의 기능에 따라 다양한 데이터 타입
예	전송 펑션, 형 변환 펑션, 비교 펑션, 산술 연산 평션 등	타이머, 카운터, 응용 펑션 블록 특수모듈 초기화 펑션 블록 등

※ 펑션 또는 펑션 블록 사용 시 여러 개의 입, 출력 변수가 있어도 좌측 모선 과 우측 모선에 연결한 수 있는 입력 단자는 1개이며, 우측 모선에 연결할 수 있는 출력 단자도 1개 입니다.

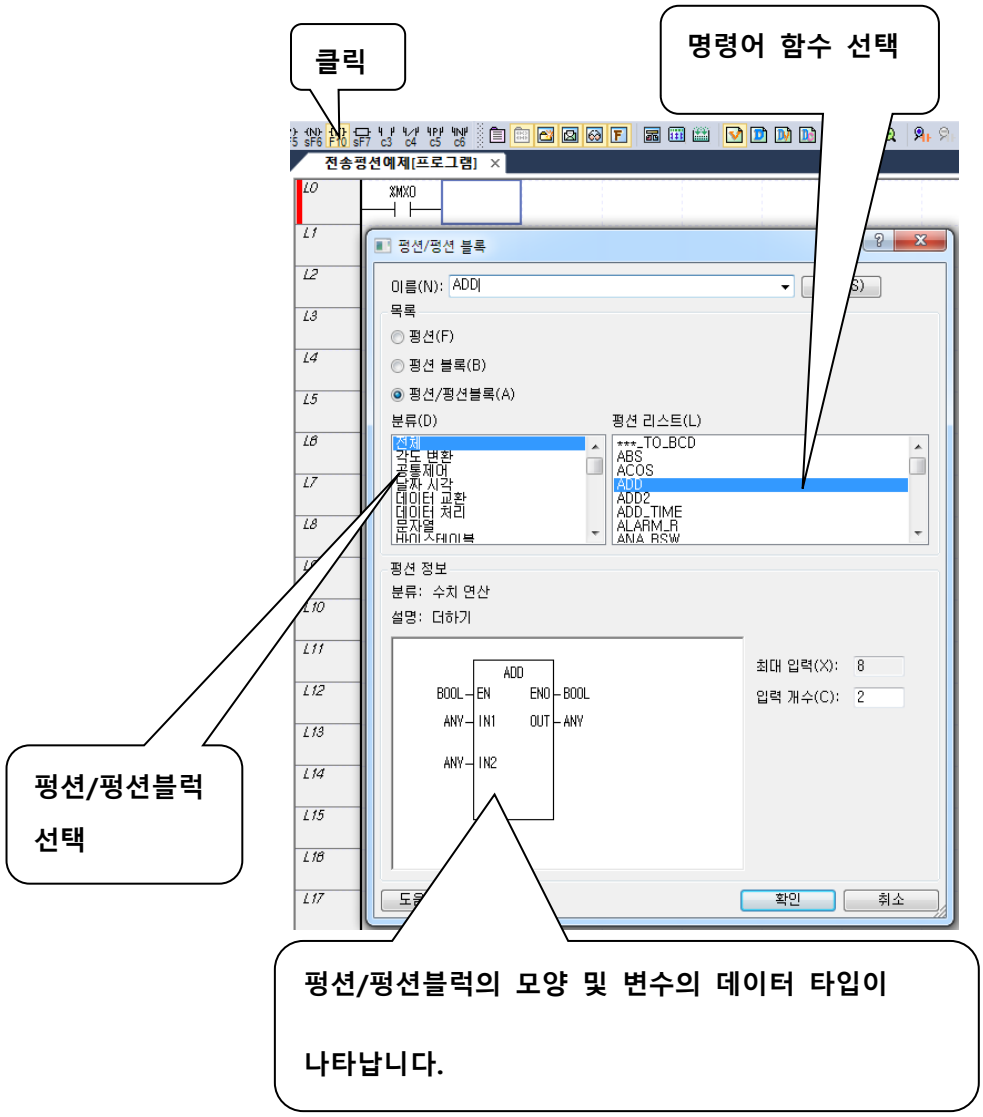
기본 펑션의 종류

기본 펑션에는 전송 펑션, 형 변환 펑션, 비교 펑션, 산술 연산 펑션, 논리 연산 펑션, 비트 시프트 펑션 등이 있습니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

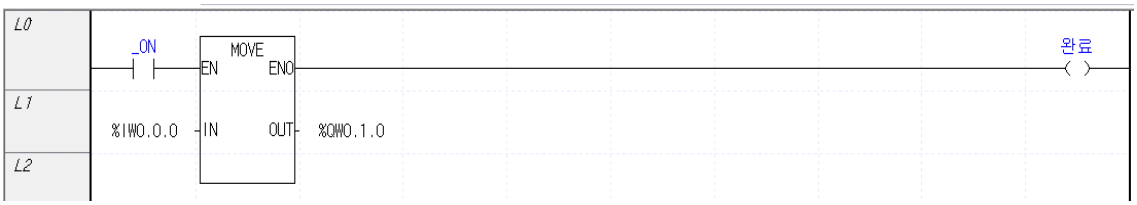
기본 펑션 프로그램

① 도구 바에서 펑션(F10)을 선택한 후 프로그램 창에서 클릭하면 펑션 선택 화면이 나타납니다. 펑션 선택 화면에서 사용하고자 하는 펑션의 이름을 입력하고 "확인"을 클릭하면 프로그램 창에 펑션이 등록됩니다.



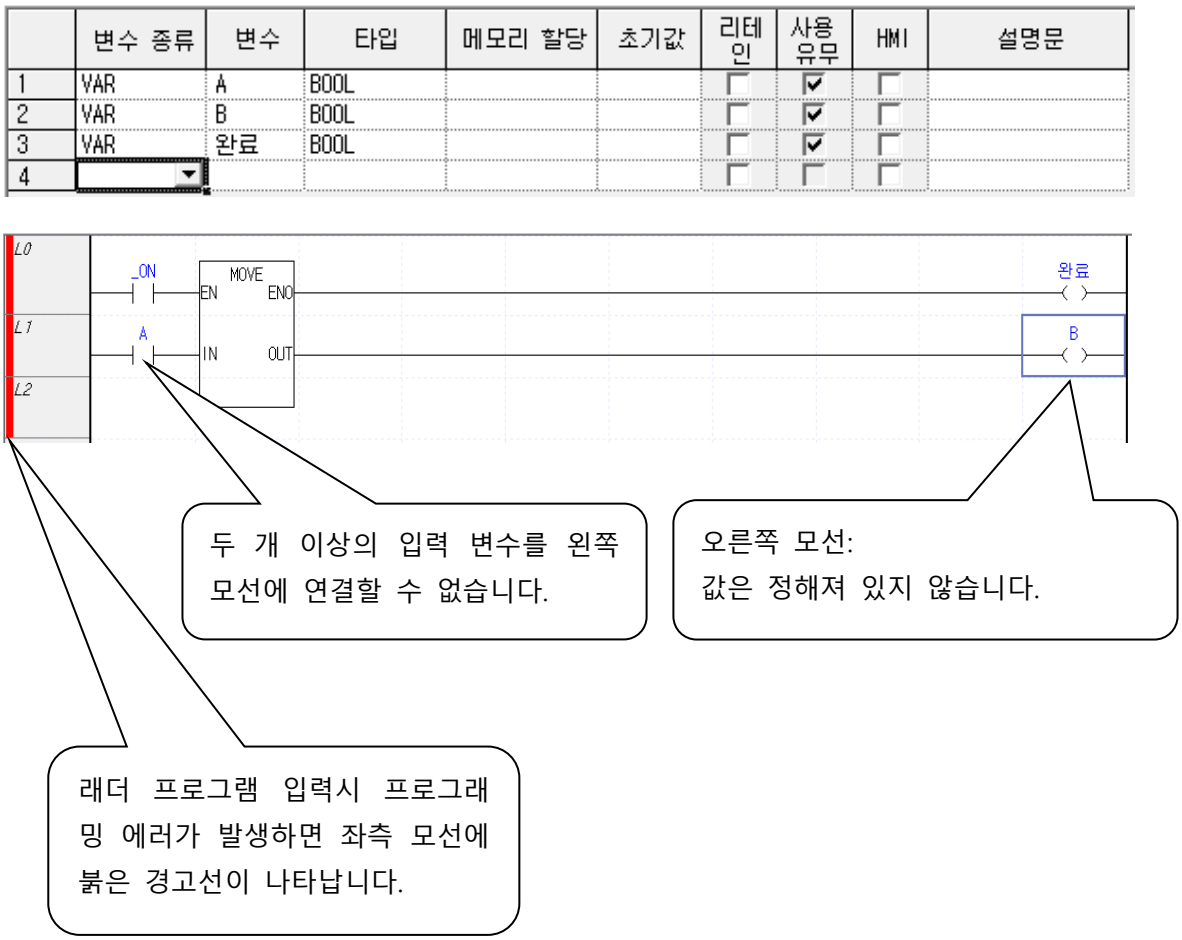
② 펑션이 등록되었으면 시퀀스 연산자를 이용하여 입력 변수 및 출력 변수를 설정합니다.

	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무	HMI	설명문
1	VAR	완료	BOOL			☐	☑	☐	
2						☐	☐	☐	



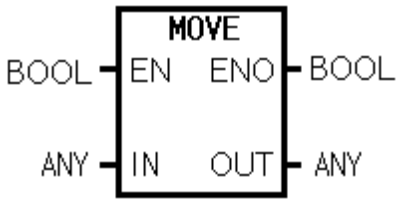
Copyright ⓒ 2021 LS Electric Co., Ltd.

③ 다음과 같이 입력 변수 2개 이상을 좌측 모선에 연결 하거나 출력 변수 2개 이상을 우측 모선에 연결하면 에러가 발생합니다.



▶ 전송 펑션

MOVE

펑션	설명
	입력: EN: EN이 1일 때 펑션 실행 IN: 전송할 값 또는 전송할 데이터가 저장된 변수 출력: EN: 펑션이 수행되면 1 출력 OUT: 데이터가 저장될 영역

① 기능

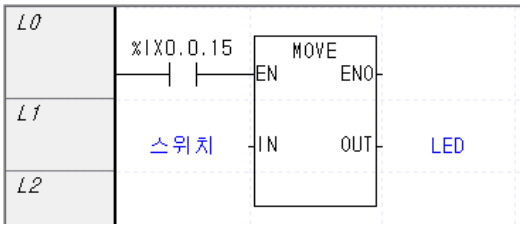
EN 이 ON되면 IN으로 입력되는 데이터를 OUT으로 전송합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

② 프로그램 예1

실행 조건이 ON 되면 스위치의 하위 1Byte(%IB0.0.0 = %IX0.0.0 ~ %IX0.0.7)까지의 ON/OFF 정보가 복사되어 LED의 하위 1Byte(%QB0.1.0 = %QX0.1.0 ~ %QX0.1.7)로 전송됩니다.

	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무	HMI	설명문
1	VAR	LED	BYTE	%QB0.1.0		☐	☑	☐	
2	VAR	스위치	BYTE	%IB0.0.0		☐	☑	☐	
3	VAR	완료	BOOL			☐	☑	☐	
4						☐	☐	☐	

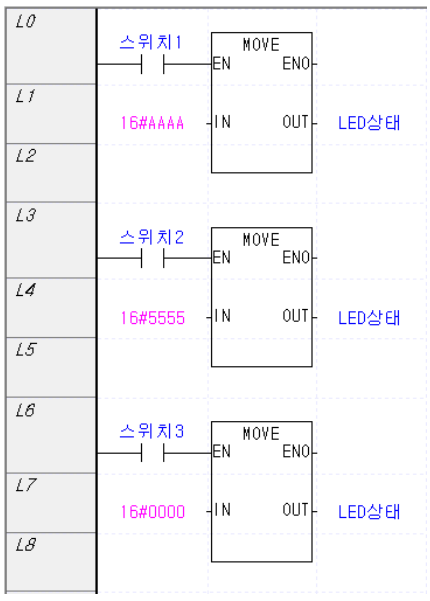


펑션에서 EN은 입력 변수가 아닌 펑션 실행의 조건이며, ENO는 출력 변수가 아닌 펑션 실행 완료 신호입니다. 그리고, 입력 변수와 출력 변수의 데이터 타입은 동일해야 합니다.

③ 프로그램 예2

스위치0, 1, 2 중 하나를 ON 하면 MOVE 펑션이 실행되어 해당 코드값을 LED(%QW0.1.0)로 전송합니다.

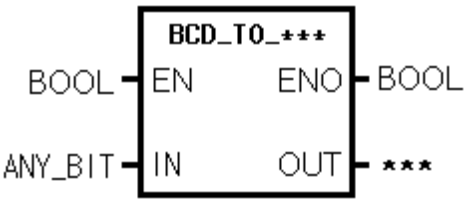
	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무
1	VAR	LED상태	WORD	%QW0.1.0		☐	☑
2	VAR	스위치1	BOOL	%IX0.0.0		☐	☑
3	VAR	스위치2	BOOL	%IX0.0.1		☐	☑
4	VAR	스위치3	BOOL	%IX0.0.2		☐	☑
5						☐	☐



Copyright ⓒ 2021 LS Electric Co., Ltd.

▶ 형 변환 펑션

BCD_TO_***

펑션	설명
	입력: EN: EN이 1일 때 펑션 실행 IN: BCD 형태의 데이터를 갖는 ANY_BIT 입력값 출력: EN: 펑션이 수행되면 1 출력 OUT: 타입 변환된 데이터가 저장될 변수

① 기능

BCD 코드의 입력 데이터를 바이너리 코드(정수)로 바꾸어 OUT으로 설정된 변수에 저장합니다.

펑션 그룹	펑션 이름	입력 데이터 타입	출력 데이터 타입	비 고
BCD_TO_***	BYTE_BCD_TO_SINT	BYTE(BCD)	SINT	-
	WORD_BCD_TO_INT	WORD(BCD)	INT	-
	DWORD_BCD_TO_DINT	DWORD(BCD)	DINT	-
	LWORD_BCD_TO_LINT	LWORD(BCD)	LINT	-
	BYTE_BCD_TO_USINT	BYTE(BCD)	USINT	-
	WORD_BCD_TO_UINT	WORD(BCD)	UINT	-
	DWORD_BCD_TO_UDINT	DWORD(BCD)	UDINT	-
	LWORD_BCD_TO_ULINT	LWORD(BCD)	ULINT	-

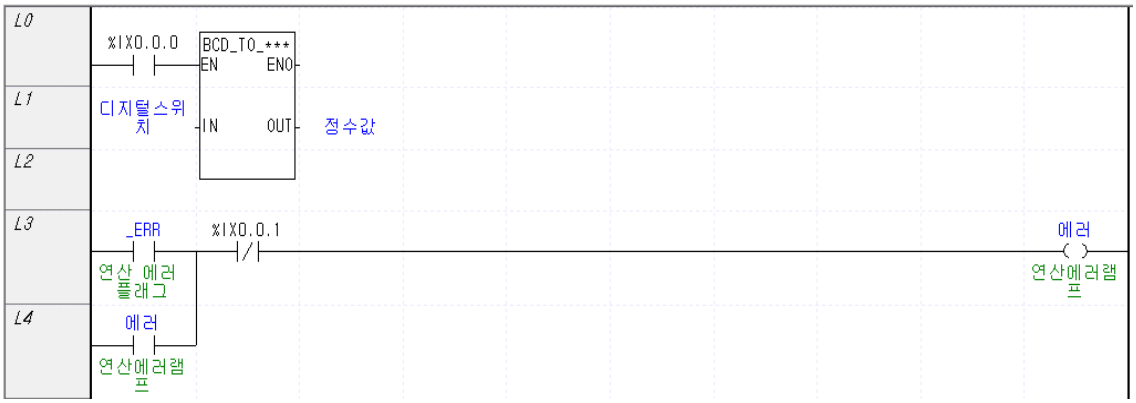
BCD 코드란 A~F까지를 사용할 수 없는 16진수를 말합니다. 따라서 입력 변수에 16#1A, 16#AF등은 사용할 수 없습니다. 입력 변수가 BCD 형이 아닐 경우 출력은 0이 되고, _ERR(연산 에러 플래그), _LER(연산 에러 래치 플래그)가 ON 됩니다.

② 프로그램 예

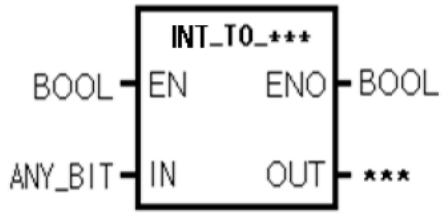
디지털 스위치(%IW0.0.1)를 사용하여 BCD 값을 입력하고, 스위치0 (%IX0.0.0)를 ON 하면 정수로 변환되어 정수값에 저장됩니다. 만일, 입력의 데이터가 BCD형이 아닐 경우, 에러 램프가 0N 됩니다.

	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무	HMI	설명문
1	VAR	디지털스위치	WORD	%IW0.0.1		☐	☑	☐	
2	VAR	에러	BOOL	%QX0.1.0		☐	☑	☐	연산에러램프
3	VAR	정수값	INT			☐	☑	☐	

Copyright ⓒ 2021 LS Electric Co., Ltd.



INT_TO_***

펑션	설명
	입력: EN: EN이 1일 때 펑션 실행 IN: BCD 형태의 데이터를 갖는 ANY_BIT 입력값 출력: EN: 펑션이 수행되면 1 출력 OUT: 타입 변환된 데이터가 저장될 변수

① 기능

정수형의 입력 데이터를 BCD 코드로 형 변환하여 OUT으로 설정된 변수에 저장합니다.

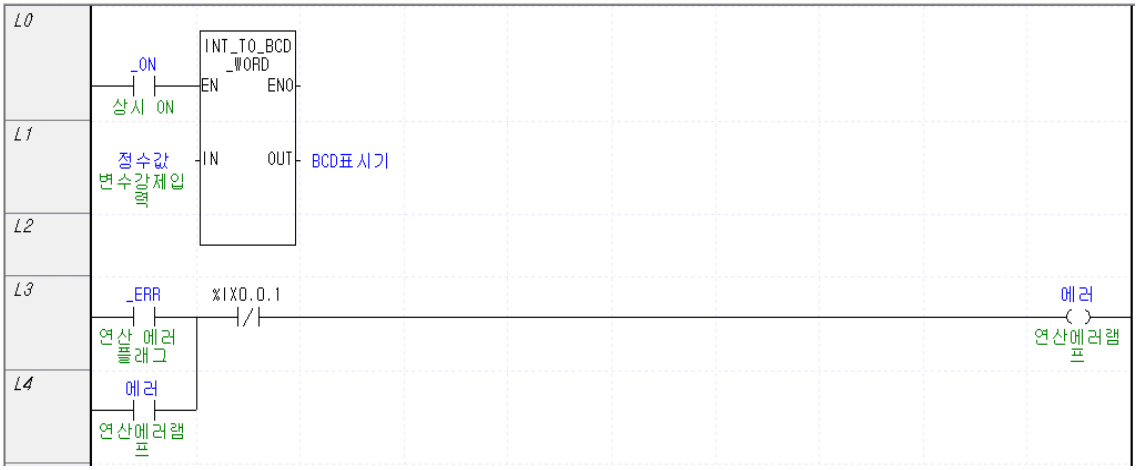
펑션 이름	입력타입	출력타입	비 고
INT_TO_SINT	INT	SINT	입력이 –128 ~ 127일 때 정상 변환
INT_TO_DINT	INT	DINT	정상 변환
INT_TO_LINT	INT	LINT	정상 변환
INT_TO_USINT	INT	USINT	입력이 0~255일 때 정상 변환
INT_TO_UINT	INT	UINT	입력이 0~32767일 때 정상 변환
INT_TO_UDINT	INT	UDINT	입력이 0~32767일 때 정상 변환
INT_TO_ULINT	INT	ULINT	입력이 0~32767일 때 정상 변환
INT_TO_BOOL	INT	BOOL	하위 1비트를 취해서 BOOL로 변환
INT_TO_BYTE	INT	BYTE	하위 1바이트를 취해서 BYTE로 변환
INT_TO_WORD	INT	WORD	비트 배열의 변화 없이 WORD로 변환
INT_TO_DWORD	INT	DWORD	상위 비트열을 0으로 채움
INT_TO_LWORD	INT	LWORD	상위 비트열을 0으로 채움
INT_TO_REAL	INT	REAL	INT타입을 REAL타입으로변환
INT_TO_LREAL	INT	LREAL	INT타입을 REAL타입으로변환
INT_TO_STRING	INT	STRING	INT타입을 STRING타입으로변환

Copyright ⓒ 2021 LS Electric Co., Ltd.

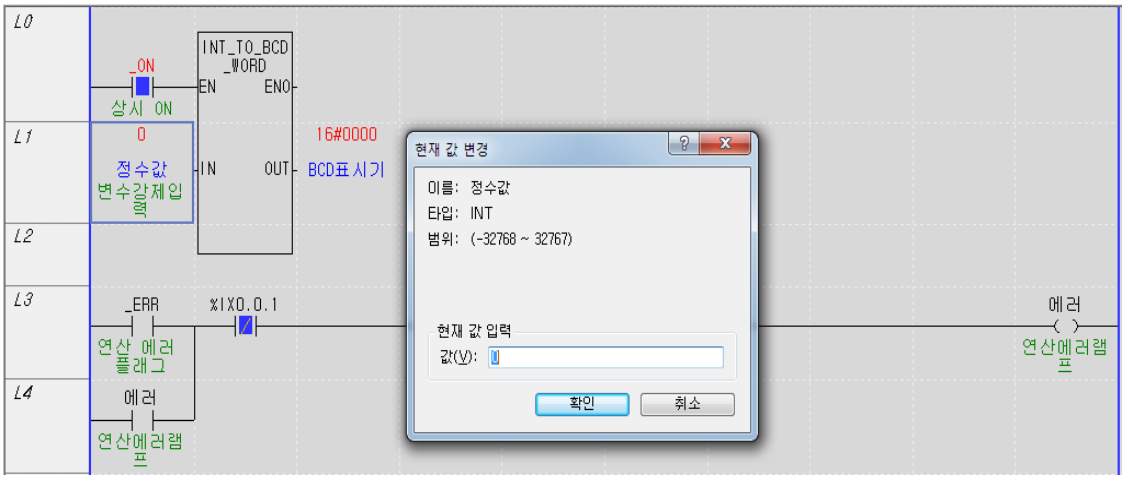
② 프로그램 예

변수 강제 입력 기능을 이용하여 정수값에 0 ~ 9999사이의 임의의 값을 입력했을 때 BCD타입으로 형 변환하여 BCD 표시기로 표시됩니다.입력값이 0 ~ 9999사이의 값이 아닐 경우 에러 램프가 ON 됩니다.

	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무	HMI	설명문
1	VAR	BCD표시기	WORD	%QW0.1.1		☐	☑	☐	
2	VAR	에러	BOOL	%QX0.1.0		☐	☑	☐	연산에러램프
3	VAR	정수값	INT			☐	☑	☐	변수강제입력
4						☐	☐	☐	

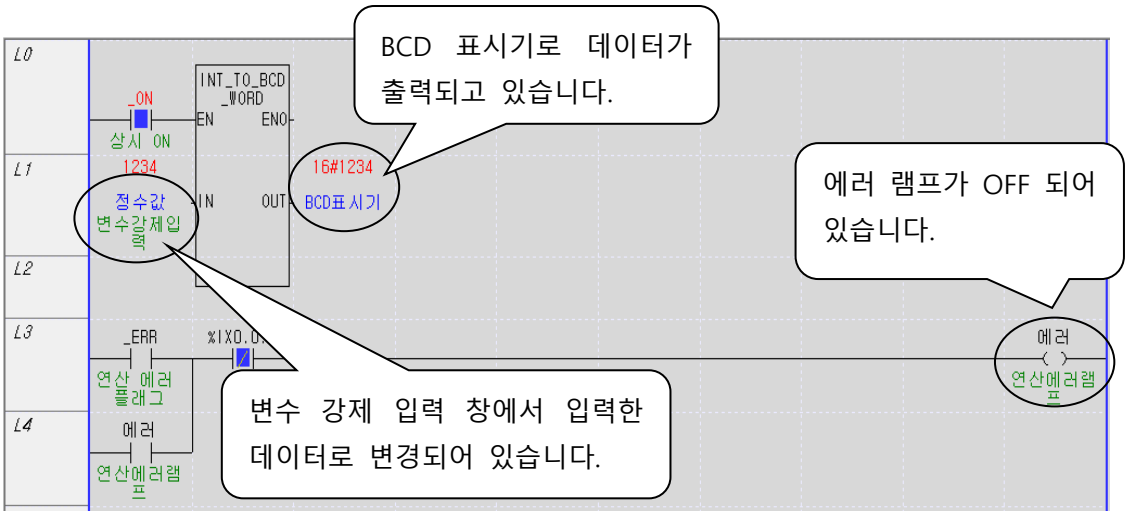


프로그램을 작성하여 PLC로 전송한 후 모니터를 실행합니다. 모니터를 실행하면서 펑션의 입력 변수 '정수값'을 더블 클릭하면 다음과 같은 변수 강제 입력 창이 나타납니다.

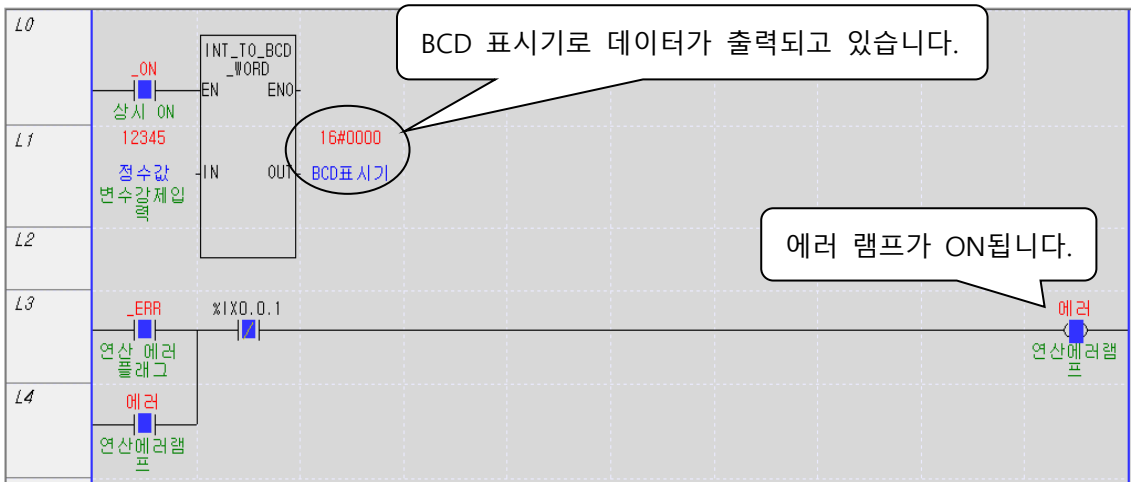


Copyright ⓒ 2021 LS Electric Co., Ltd.

변수 강제 입력 창에서 0 ~ 9999사이의 임의의 값을 입력하고 확인을 클릭합니다.



변수 강제 입력 창에서 0 ~ 9999이외의 임의의 값을 입력하고 확인을 클릭합니다.



Copyright ⓒ 2021 LS Electric Co., Ltd.

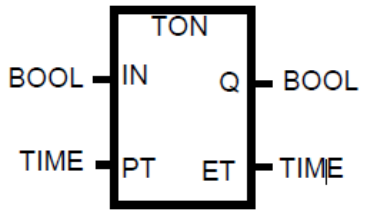
5.3. 펑션 블록 프로그램

▶ 타이머 종류

타이머는 시간을 측정하는데 사용되며 다음과 XGI에서는 기본적으로 다음과 같은 종류의 타이머가 있습니다.

No.	펑션블록	기 능	비 고
1	TP	펄스 타이머(Pulse Timer)	-
2	TON	On 딜레이 타이머(On-Delay Timer)	-
3	TOF	Off 딜레이 타이머(Off-Delay Timer)	-
4	TMR	적산 타이머(Integrating Timer)	-
5	TP_RST	펄스 타이머의 출력 Off가 가능한 노스테이블 타이머	-
6	TRTG	리트리거블 타이머(Retriggerable Timer)	-
7	TOF_RST	동작 중 출력 Off가 가능한 Off 딜레이 타이머(Off-Delay Timer)	-
8	TON_UINT	정수 설정 On 딜레이 타이머(On-Delay Timer)	-
9	TOF_UINT	정수 설정 Off 딜레이 타이머(Off-Delay Timer)	-
10	TP_UINT	정수 설정 펄스 타이머(Pulse Timer)	-
11	TMR_UINT	정수 설정 적산 타이머(Integrating Timer)	-
12	TMR_FLK	점멸 기능 타이머	-
13	TRTG_UINT	정수 설정 리트리거블 타이머	-

▶ TON (ON-Delay Timer)

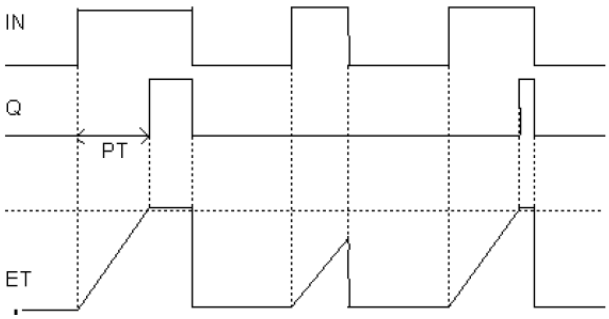
펑션	설명
	입력: IN: 타이머의 기동 조건 PT: 설정 시간 출력: Q: 타이머 접점 출력 CV: 경과 시간

① 기능

IN이 1이 된 후 경과 시간이 ET로 출력됩니다. 만일, 경과 시간 ET가 설정 시간에 도달하기 전에 IN이 0이면, 경과 시간은 0으로 됩니다. Q가 1이 된 후 IN이 0이 되면, Q는 0이 됩니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

② 타임 차트



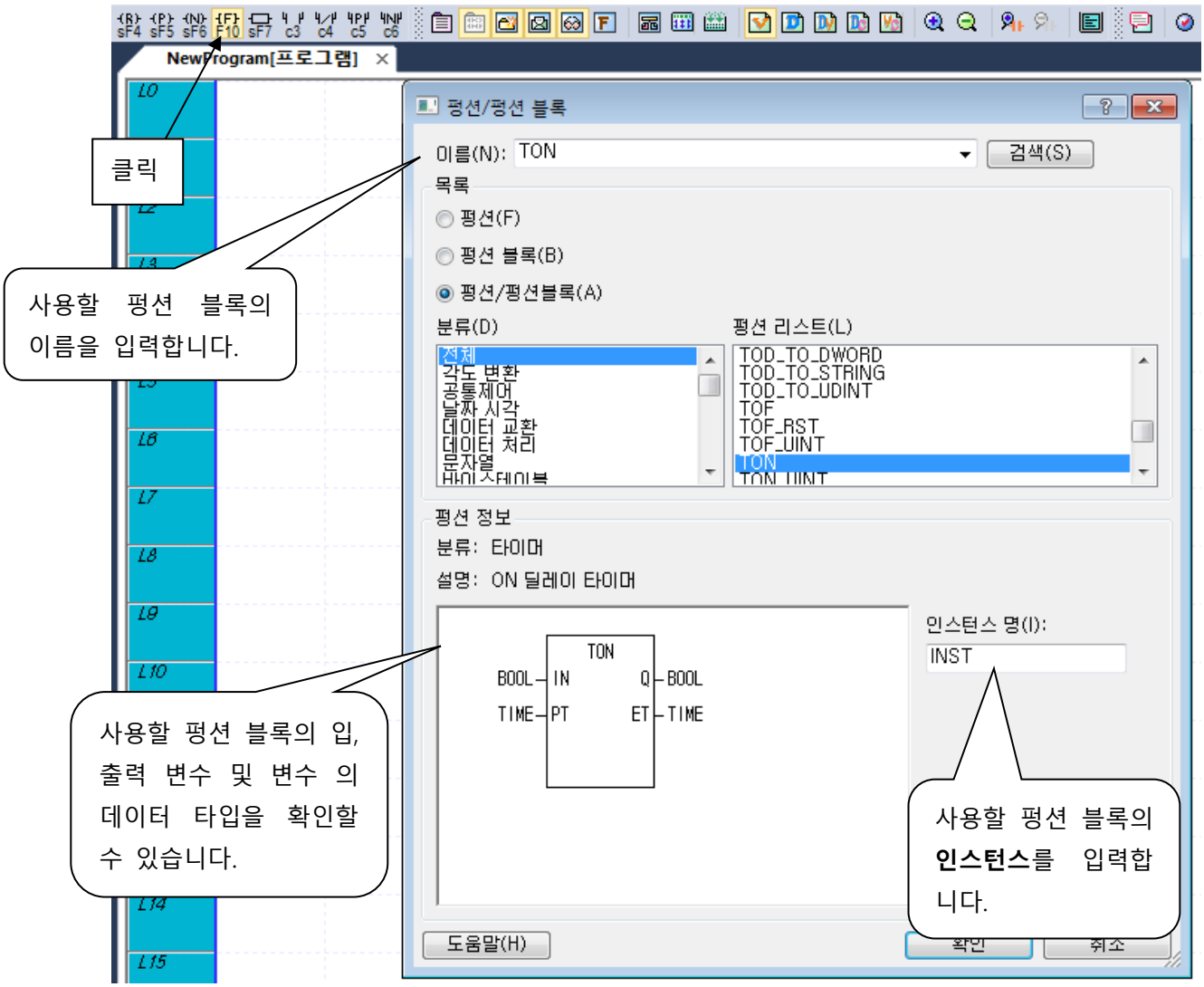
③ 프로그램

ON스위치를 ON시킨 후 5초 이상 ON상태를 유지시키면 5초 후에 LED 램프가 ON됩니다. 만일 ON스위치를 5초 이전에 OFF 시키면 경과시간은 0으로 리셋되고, LED 램프는 ON 되지 않습니다.

① 타이머의 최대 설정시간: **T#49D17H2M47S295MS**

타이머의 설정 시간은 TIME 형으로 설정하며, 날짜는 D, 시간은 H, 분은 M, 초는 S, 1/1000초는 ms단위를 사용합니다.

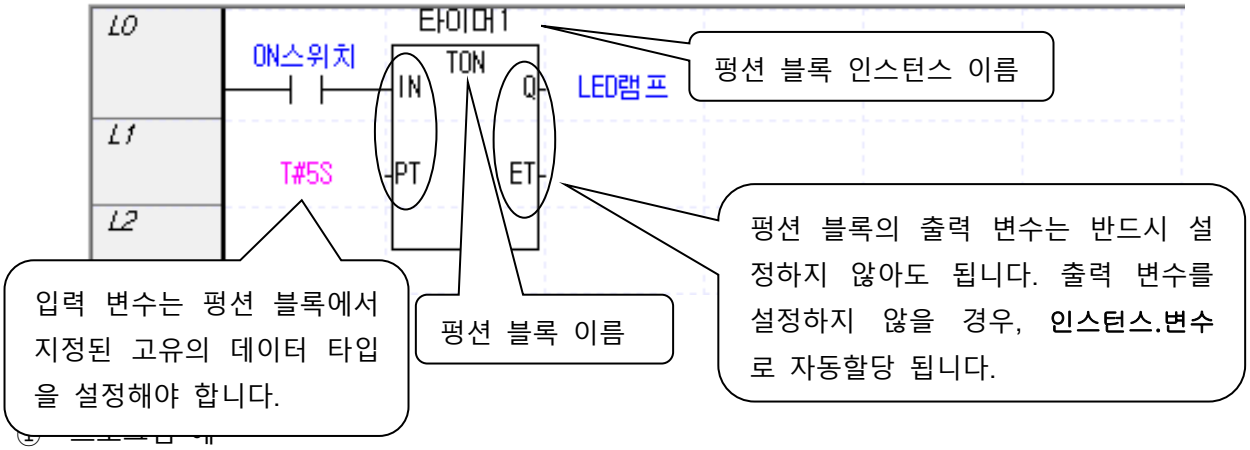
예) 1일 2시간 3분 4초 567MS => T#1D2H3M4S567MS



Copyright ⓒ 2021 LS Electric Co., Ltd.

② 펑션 블록 인스턴트란 펑션 블록에 관련된 입, 출력 변수를 통합적으로 관리하기 위해 설정하는 변수명입니다.

	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무
1	VAR	LED램프	BOOL	%QX0.1.0		☐	☑
2	VAR	ON스위치	BOOL	%IX0.0.0		☐	☑
3	VAR	타이머1	TON			☐	☑
4						☐	☐



TON은 펑션 블록이므로 연산 중 누계되는 데이터를 잠시 보관하기 위한 **인스턴스 변수**를 반드시 선언해야 합니다.

XG5000에서프로그램 편집 시 TON의 인스턴스 변수를 선언하면 타이머 출력은 **인스턴스 이름.Q**, 경과 시간은 **인스턴스 이름.ET**로 변수가 자동 생성됩니다.

TON의 인스턴스 변수 T1을 선언합니다.

타이머 T1의 설정 시간을 7초(T#7S)로 설정합니다.

기동 스위치0 (%IX0.0.0)를 ON하면 T1.ET에 경과 시간이 표시됩니다.

경과 시간 T1.ET가 설정 시간 7초에 도달하면 타이머 출력 T1.Q가 ON됩니다.

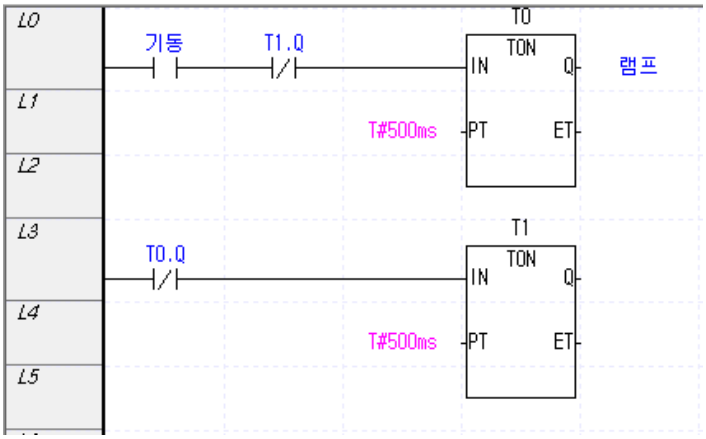
T1.Q가 ON된 후 기동 SW (%IX0.0.0)를 OFF하면 T1.Q는 OFF 됩니다.

	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무
1	VAR	T1	TON			☐	☑
2	VAR	기동스위치	BOOL	%IX0.0.0		☐	☑
3	VAR	램프1	BOOL	%QX0.1.0		☐	☑
4						☐	☐

Copyright ⓒ 2021 LS Electric Co., Ltd.

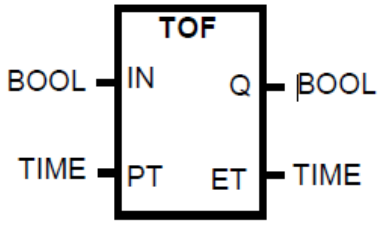
예제1) ON-Delay Timer를 이용한 프로그램: 플리커 회로

ON-Delay 타이머 두 개를 사용하여 램프를 플리커 시킵니다.



	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용 유무
1	VAR	T0	TON			☐	☑
2	VAR	T1	TON			☐	☑
3	VAR	T2	TON			☐	☐
4	VAR	기동	BOOL	%IX0.0.0		☐	☑
5	VAR	램프	BOOL	%QX0.1.0		☐	☑
6						☐	☐

▶ TOF (OFF-Delay Timer)

펑션	설명
	입력: IN: 타이머의 기동 조건 PT: 설정 시간 출력: Q: 타이머 접점 출력 CV: 경과 시간

① 기능

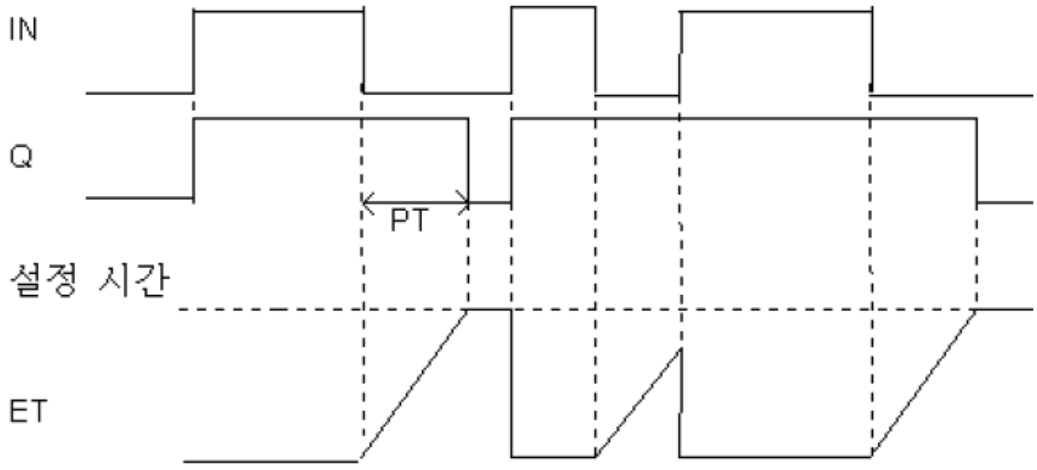
IN이 ON 되면, Q가 ON 되고, IN이 OFF 된 후부터 PT에 의해서 지정된 설정 시간이 경과한 후 Q가 OFF 됩니다.

IN이 OFF된 후 경과 시간이 ET로 출력됩니다.

만일 경과 시간 ET가 설정 시간에 도달하기 전에 IN이 ON 되면, 경과 시간은 다시 0으로 됩니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

② 타임 차트



③ 프로그램 예

TOF는 펑션 블록이므로 연산 중 누계되는 데이터를 잠시 보관하기 위한 **인스턴스 변수**를 반드시 선언해야 합니다.

XG5000에서프로그램 편집시 TOF의 인스턴스 변수를 선언하면 타이머 출력은 **인스턴스 이름.Q**, 경과 시간은 **인스턴스 이름.ET**로 변수가 자동 생성됩니다.

TOF의 인스턴스 변수 T2를 선언합니다.

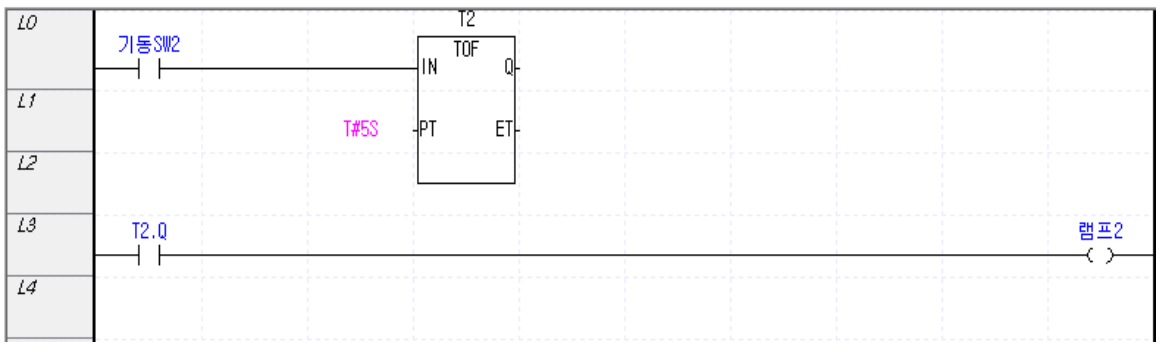
타이머 T2의 설정 시간을 5초(T#5S)로 설정합니다.

기동 SW1 (%IX0.0.1)을 ON하면 타이머 출력 T2.Q가 ON됩니다.

기동 SW1 (%IX0.0.1)을 OFF하면 T2.ET에 경과 시간이 표시됩니다.

경과 시간 T2.ET가 설정시간 5초에 도달하면 타이머 출력 T2.Q가 OFF됩니다.

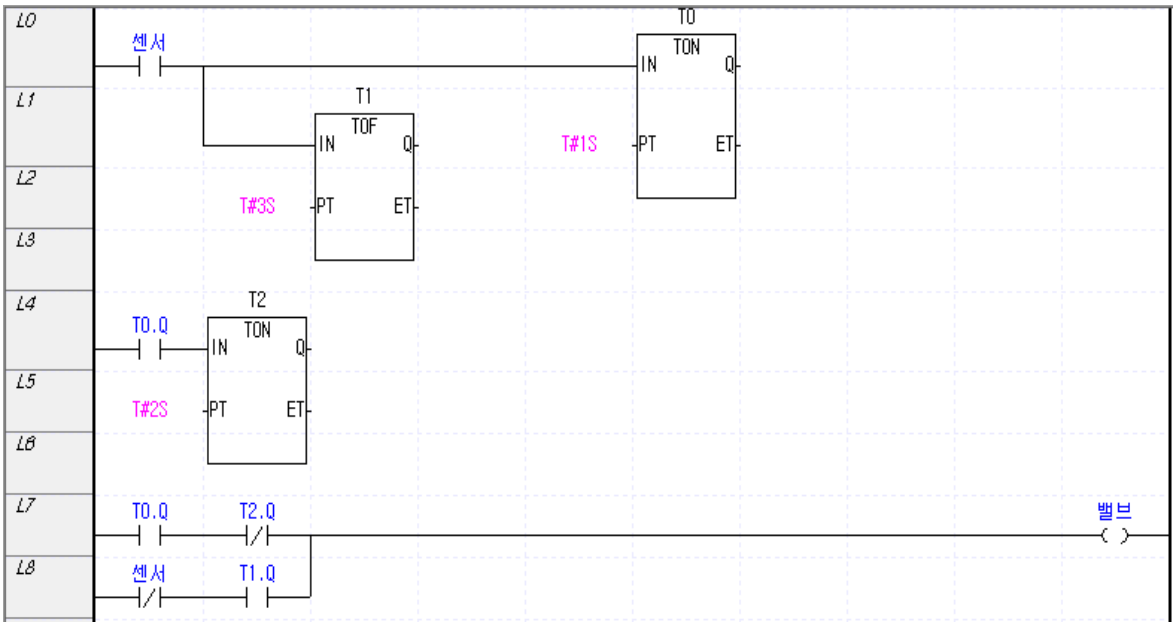
	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무
1	VAR	T2	TOF			☐	☑
2	VAR	기동SW2	BOOL			☐	☑
3	VAR	램프2	BOOL			☐	☑
4						☐	☐



Copyright ⓒ 2021 LS Electric Co., Ltd.

예제2) TON, TOF를 이용한 프로그램: 화장실 자동 밸브 제어

사용자가 변기에 접근한 후 1초 뒤 2초간 물이 나오고 이탈 후 즉시 3초간 물이 공급되는 회로입니다.

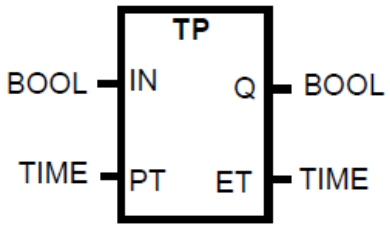


<프로그램>

	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용 유무
1	VAR	T0	TON			☐	☑
2	VAR	T1	TOF			☐	☑
3	VAR	T2	TON			☐	☑
4	VAR	밸브	BOOL	%QX0.1.0		☐	☑
5	VAR	센서	BOOL	%IX0.0.0		☐	☑
6						☐	☐

<지역변수>

▶ TP (Pulse Timer)

펑션	설명
	입력: IN: 타이머의 기동 조건 PT: 설정 시간 출력: Q: 타이머 접점 출력 CV: 경과 시간

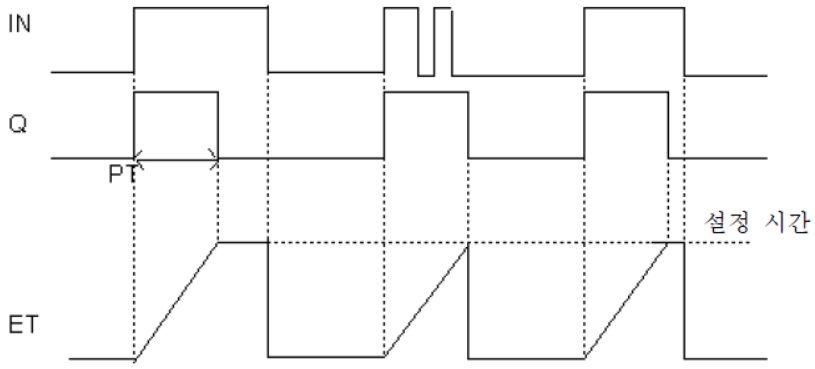
① 기능

IN이 ON 되면 PT에 의해서 지정된 설정 시간 동안만 Q가 ON 되고, ET가 PT에 도달하면 자동으로 0이 됩니다. 경과 시간 ET는 IN이 ON 되었을 때부터 증가하며 PT에 이르면 값을 유지하다가

Copyright ⓒ 2021 LS Electric Co., Ltd.

IN이 0이 될 때 0의 값이 됩니다. ET가 증가할 동안 IN이 OFF 되거나 다시 ON 되어도 영향이 없습니다. ET가 PT에 도달한 후 IN이 다시 ON되면 ET가 증가합니다.

② 타임 차트



③ 프로그램 예

TP는 펑션 블록이므로 연산 중 누계되는 데이터를 잠시 보관하기 위한 **인스턴스 변수**를 반드시 선언해야 합니다.

XG5000에서프로그램 편집 시 TP의 인스턴스 변수를 선언하면 타이머 출력은 **인스턴스 이름.Q**, 경과 시간은 **인스턴스 이름.ET**로 변수가 자동 생성됩니다.

TP의 인스턴스 변수 T3를 선언합니다.

타이머 T3의 설정 시간을 5초(T#5S)로 설정합니다.

기동 SW0 (%IX0.0.0)를 OFF → ON 하면 타이머 출력 T3.Q가 5초 동안 ON 했다 OFF합니다.

T3.ET가 증가할 동안 기동 스위치0가 OFF 되거나 다시 ON 되어도 영향이 없습니다.

T3.ET가 증가할 동안 T3.ET에 경과 시간이 표시됩니다.

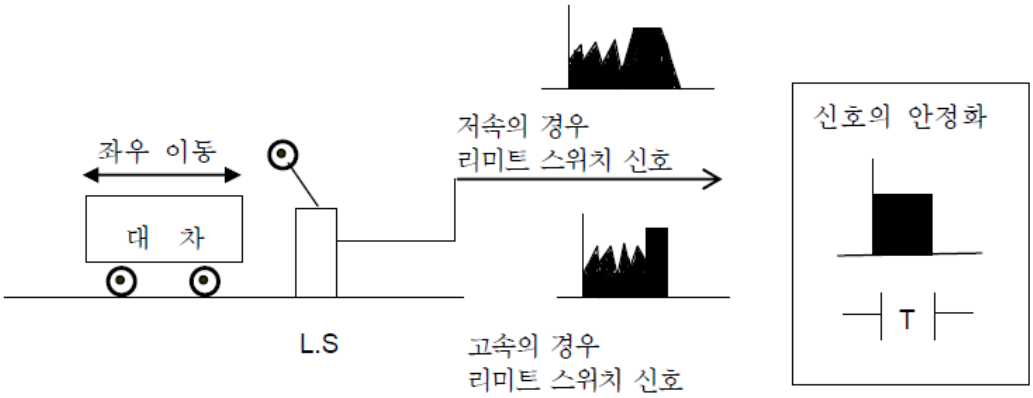
	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무
1	VAR	T3	TP			☐	☑
2	VAR	기동SW0	BOOL	%IX0.0.0		☐	☑
3	VAR	램프0	BOOL	%QX0.1.0		☐	☑

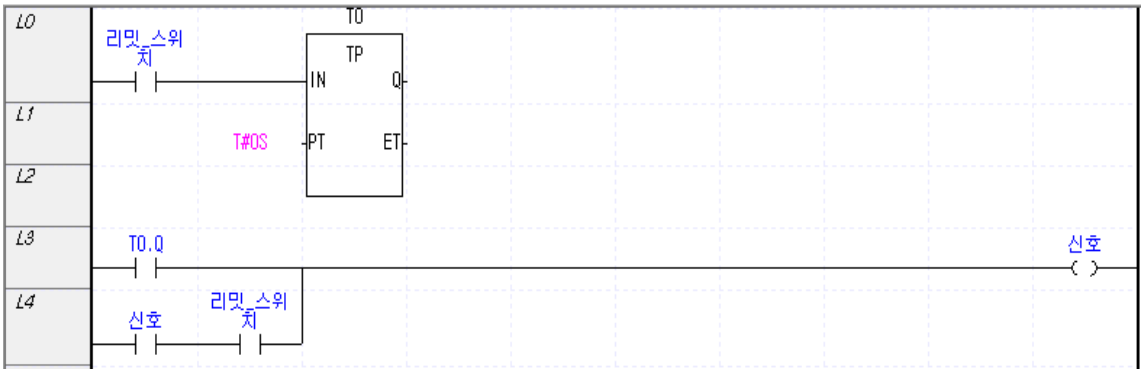
Copyright ⓒ 2021 LS Electric Co., Ltd.

예제3) TP를 이용한 프로그램: 신호떨림 방지 회로

신호를 얻습니다.

<시스템도>



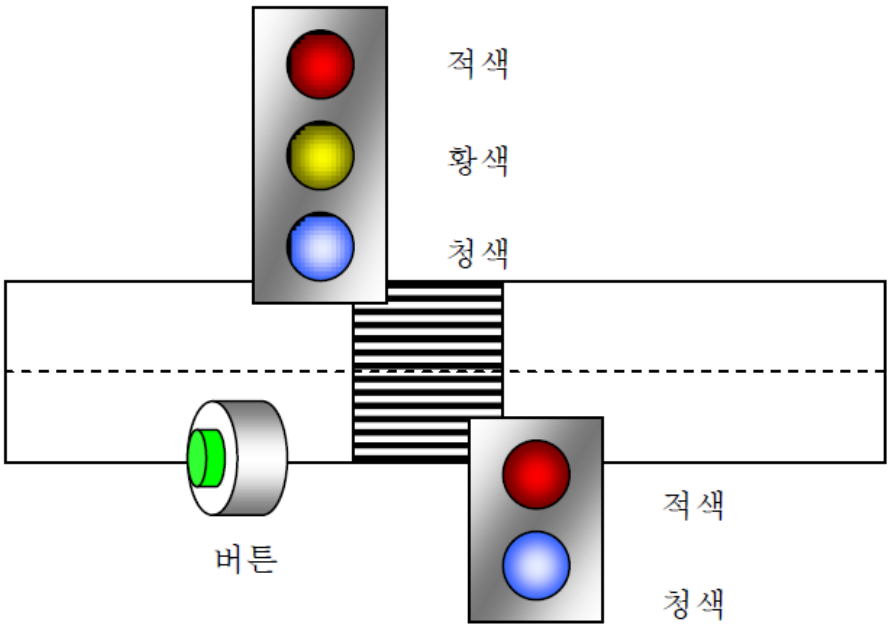


	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무
1	VAR	T0	TP			☐	☑
2	VAR	리밋_스위치	BOOL	%IX0.0.0		☐	☑
3	VAR	신호	BOOL	%QX0.1.0		☐	☑
4						☐	☐

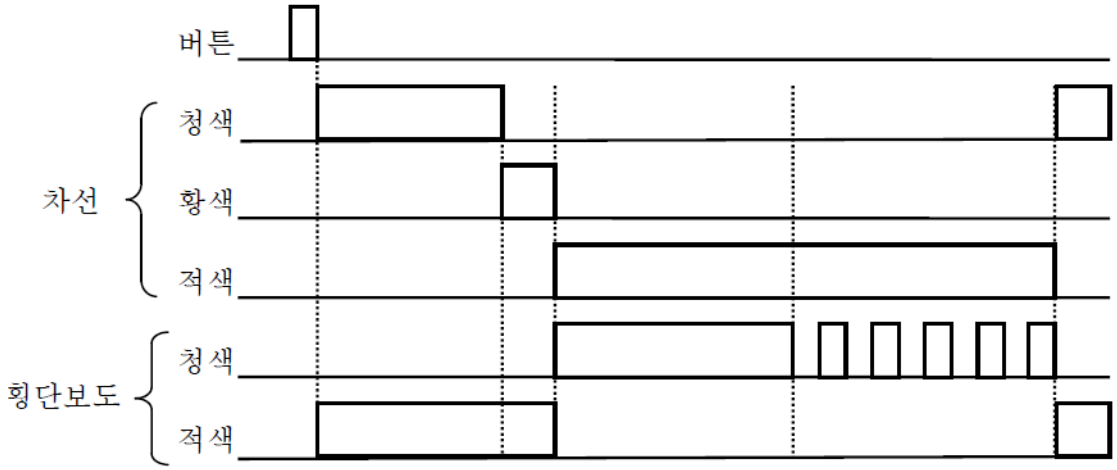
Copyright ⓒ 2021 LS Electric Co., Ltd.

예제4) TON을 이용한 프로그램: 신호등 제어

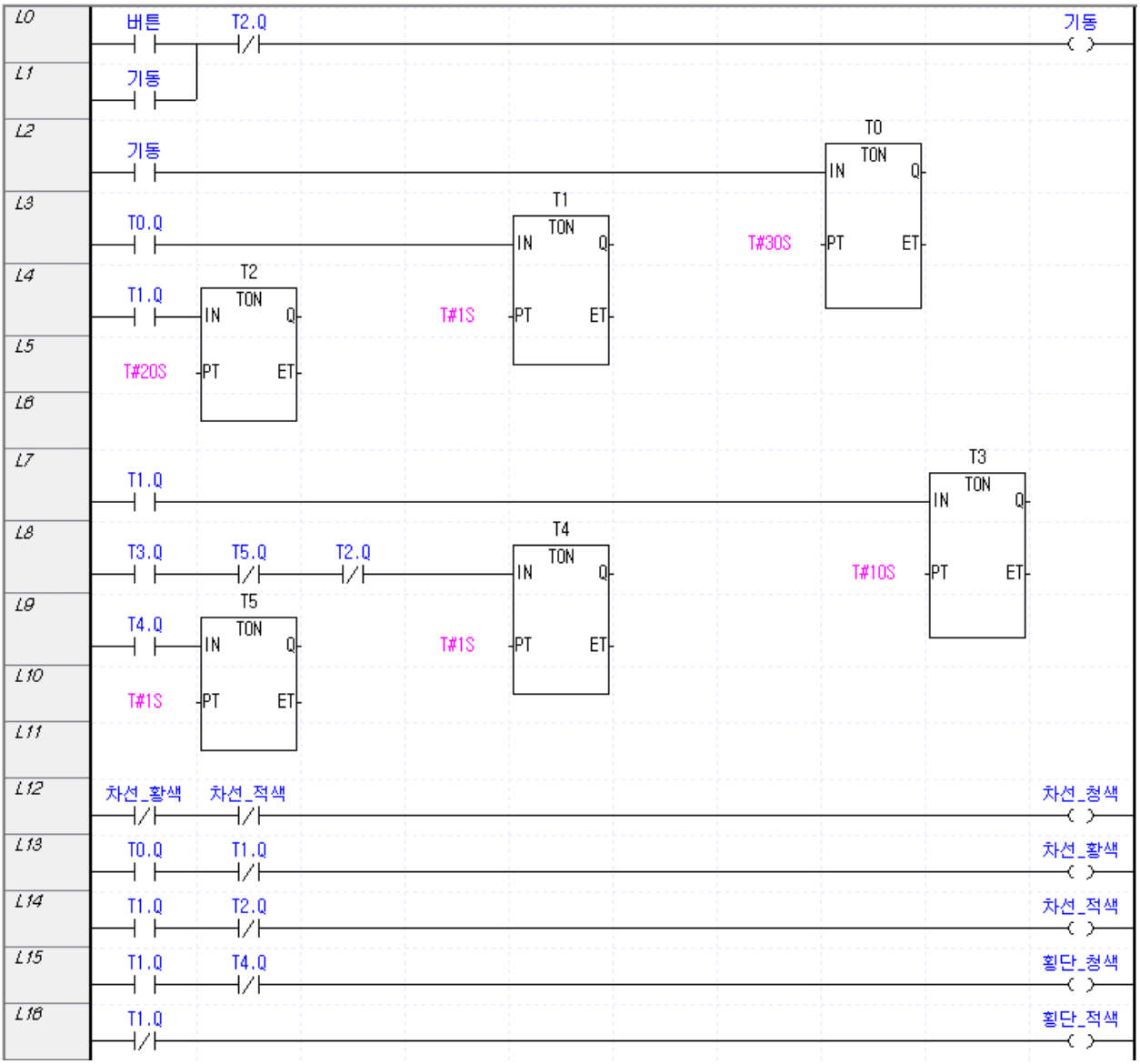
보행자가 보행버튼을 누르면 30초 후 차선의 신호등은 황색램프가 점등되며 1초 후 적색으로 바뀝니다. 이때 보행자신호등은 청색램프가10초간 점등된 뒤 10초간 점멸하며 이후 적색으로 바뀝니다.



① 타임 차트



Copyright ⓒ 2021 LS Electric Co., Ltd.



<프로그램>

	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무
1	VAR	T0	TON			☐	☑
2	VAR	T1	TON			☐	☑
3	VAR	T2	TON			☐	☑
4	VAR	T3	TON			☐	☑
5	VAR	T4	TON			☐	☑
6	VAR	T5	TON			☐	☑
7	VAR	기동	BOOL			☐	☑
8	VAR	버튼	BOOL	%IX0.0.0		☐	☑
9	VAR	차선_적색	BOOL	%QX0.1.2		☐	☑
10	VAR	차선_청색	BOOL	%QX0.1.0		☐	☑
11	VAR	차선_황색	BOOL	%QX0.1.1		☐	☑
12	VAR	횡단_적색	BOOL	%QX0.1.4		☐	☑
13	VAR	횡단_청색	BOOL	%QX0.1.3		☐	☑

<지역변수>

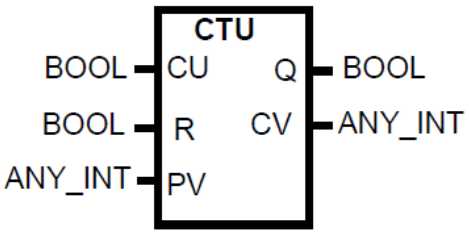
Copyright © 2021 LS Electric Co., Ltd.

카운터 종류

XGT에는 카운터 종류가 다음과 같이 기본적으로 4종이 제공 됩니다.

No.	펑션 블록 이름	기능	비고
1	CTU_***	가산 카운터(Up Counter) INT,DINT,LINT,UINT,UDINT,ULINT	-
2	CTD_***	감산 카운터(Down Counter) INT,DINT,LINT,UINT,UDINT,ULINT	-
3	CTUD_***	가감산 카운터(Up Down Counter) INT,DINT,LINT,UINT,UDINT,ULINT	-
4	CTR	링 카운터(Ring Counter)	-

▶ CTU (업 카운터)

펑션	설명
	입력: CU: 펄스 신호 입력 R: 리셋 신호 입력 PV: 설정값 출력: Q: 카운터 접점 신호 출력 CV: 카운터 현재값 출력

① 기능

가산 카운터 펑션 블록 CTU는 업 카운터 펄스입력 CU가 0에서 1이 되면 현재값 CV가 이전값 보다 1만큼 증가하는 카운터입니다

단, CV가 PV의 최대값 미만일 때만 증가하고, 최대값이 되면 더이상 증가하지 않습니다.

리셋 입력 R이 1이 되면 현재값 CV는 0으로 클리어(Clear)됩니다.

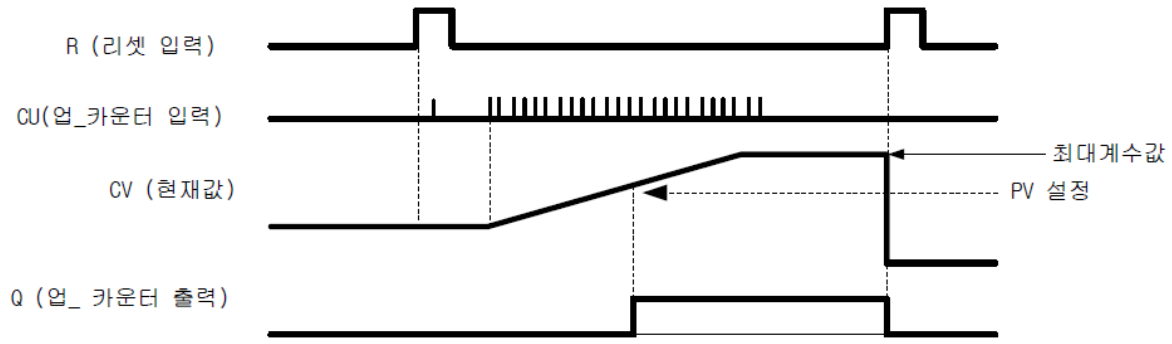
출력 Q는 CV가 PV이상이 될 때만 1이 됩니다.

PV값은 CTU 펑션 블록을 수행 시 설정값을 새롭게 가져와 연산합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

펑션 블록	PV	동작 설명
CTU_INT	INT	INT(설정값)의 최대값(32767)만큼 증가합니다.
CTU_DINT	DINT	DINT(설정값)의 최대값(2147483647)만큼 증가합니다.
CTU_LINT	LINT	LINT(설정값)의 최대값(9223372036854775807)만큼 증가합니다.
CTU_UINT	UINT	UINT(설정값)의 최대값(0)만큼 증가합니다.
CTU_UDINT	UDINT	UDINT(설정값)의 최대값(0)만큼 증가합니다.
CTU_ULINT	ULINT	ULINT(설정값)의 최대값(0)만큼 증가합니다.

② 타임 차트

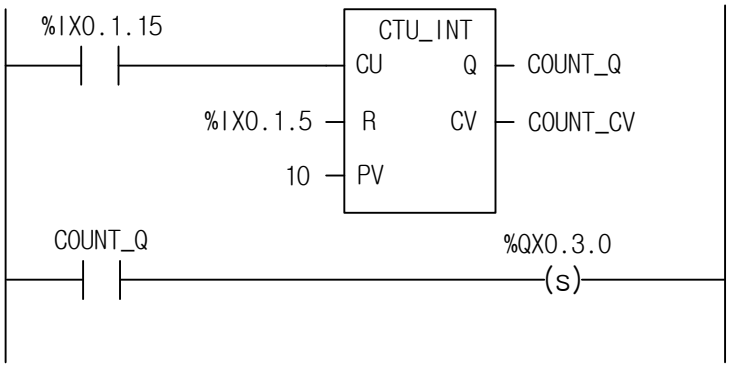


③ 프로그램

카운터의 설정값을 정수(INT)로 설정 합니다.

스위치가 ON될 때 마다 카운터의 현재값이 증가하게 되며 카운터 현재값이 설정값 이상이면 램프가 ON됩니다.

입력접점 %IX0.1.15에 10번의 펄스가 들어오면, 출력접점 %QX0.3.0을 SET 시키는 프로그램



※ 평션 블록은 여러 스캔에 걸쳐 출력을 만들기 때문에 펑션 블록을 사용할 때는 연산 중 누계되는 데이터를 잠시 보관하기 위한 **인스턴스 변수**를 반드시 선언해야 합니다.

※ XG5000에서 프로그램 편집 시 카운터 인스턴스 변수를 선언하면 카운터 출력은 **인스턴스 이름.Q**, 현재값은 **인스턴스 이름.CV**로 변수가 자동 생성됩니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

④ 프로그램 예

CTU는 펑션 블록이므로 연산 중 누계되는 데이터를 잠시 보관하기 위한 **인스턴스 변수**를 반드시 선언해야 합니다.

XG5000에서프로그램 편집 시 CTU의 인스턴스 변수를 선언하면 카운터 출력은 **인스턴스 이름.Q,** 현재값은 **인스턴스 이름.CV**로 변수가 자동 생성됩니다.

CTU의 인스턴스 변수 C1을 선언합니다.

토글 스위치0 (%IX0.0.0)로 CU에 상승(Rising Edge) 펄스를 입력하면 현재값이 증가 합니다.

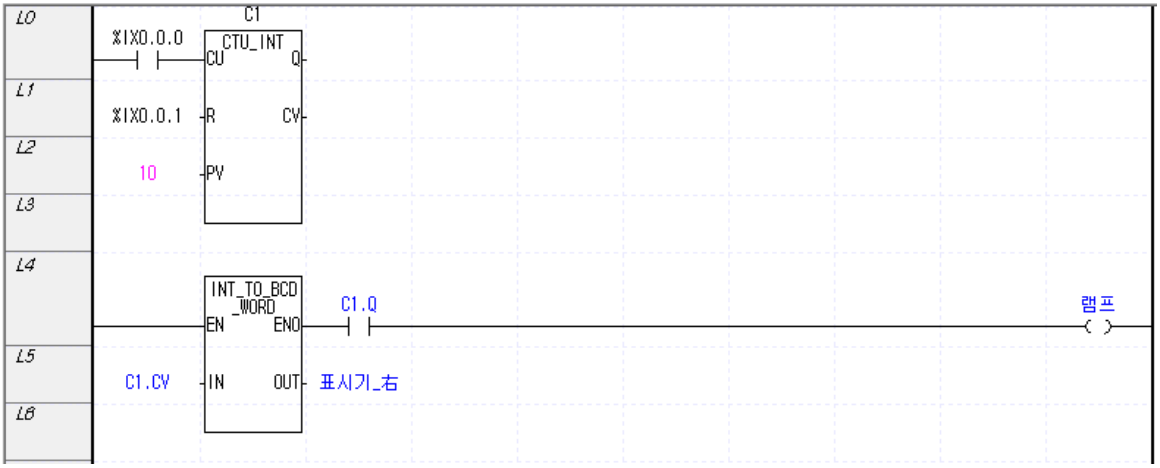
현재값을 우측 디지털 표시기(%QW0.1.1)에 출력합니다.

현재값이 설정값 이상이면 카운터 출력(C1.Q)이 1이 되어 램프(%QX0.1.0)가 점등됩니다.

토글 스위치1 (%IX0.0.1)을 ON 하면 현재값 및 카운터 출력이 리셋되어 0이 됩니다.

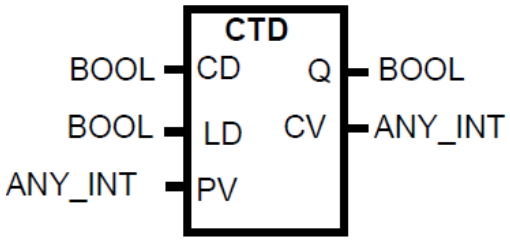
현재값(C1.CV)이 0 ~ 9999 사이를 벗어나면 펑션 INT_TO_BCD에 의해 _ERR, _LER플래그가 ON 됩니다.

	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무
1	VAR	C1	CTU_INT			☐	☐
2	VAR	램프	BOOL	%QX0.1.0		☐	☑
3	VAR	리셋	BOOL	%IX0.0.1		☐	☐
4	VAR	표시기_右	WORD	%QW0.1.1		☐	☑



Copyright ⓒ 2021 LS Electric Co., Ltd.

▶ CTD (다운 카운터)

펑션	설명
	입력: CU: 펄스 신호 입력 LD: 설정값 입력 PV: 설정값 출력: Q: 카운터 접점 신호 출력 CV: 카운터 현재값 출력

① 기능

감산 카운터 펑션 블록 CTD는 다운 카운터 펄스입력 CD가 0에서 1이 되면, 현재값 CV가 이전 값보다1만큼 감소하는 카운터 입니다.

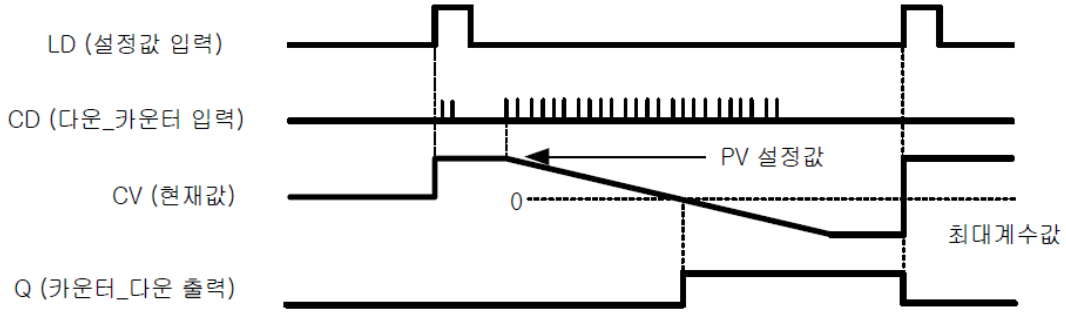
단, CV는 PV의 최소보다 클 때만 감소하고, 최소값이 되면 더이상 감소하지 않습니다.

설정값 입력 LD가 1이 되면 현재값 CV에는 설정값 PV값이 로드 됩니다. (CV = PV)

출력 Q는 CV가 0이하일 때만 1이 됩니다.

펑션 블록	PV	동작 설명
CTD_INT	INT	INT(설정값)의 최소값(-32,768)만큼 감소합니다.
CTD_DINT	DINT	DINT(설정값)의 최소값(-2,147,483,648)만큼 감소합니다.
CTD_LINT	LINT	LINT(설정값)의 최소값(-9,223,372,036,854,775,808)만큼 감소합니다.
CTD_UINT	UINT	UINT(설정값)의 최소값(0)만큼 감소합니다.
CTD_UDINT	UDINT	UDINT(설정값)의 최소값(0)만큼 감소합니다.
CTD_ULINT	ULINT	ULINT(설정값)의 최소값(0)만큼 감소합니다.

② 타임 차트

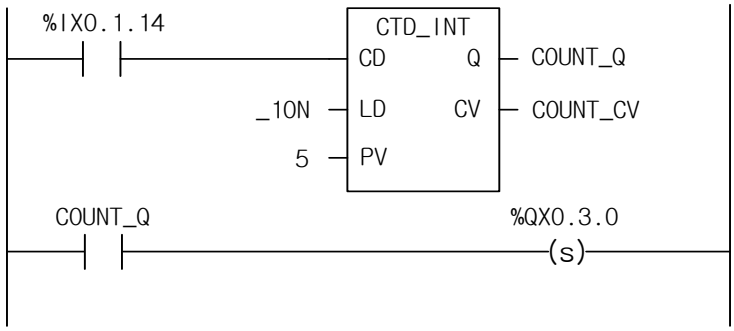


Copyright ⓒ 2021 LS Electric Co., Ltd.

③ 프로그램

정수(INT)로 카운터의 설정값을 설정하고 로드 스위치를 ON하면 카운터의 설정값이 현재값으로 로드 됩니다.

토글 스위치가 ON될 때 마다 카운터의 현재값이 감소하게 되며 카운터 현재값이 0보다 작거나 같으면 램프가 ON됩니다.



④ 프로그램 예

CTD의 인스턴스 변수 C2를 선언하고, 설정값을 5로 셋팅합니다.

초기에 _1ON(첫 스캔 ON)에 의해 LD가 1이 되어 설정값이 현재값에 로드 됩니다.

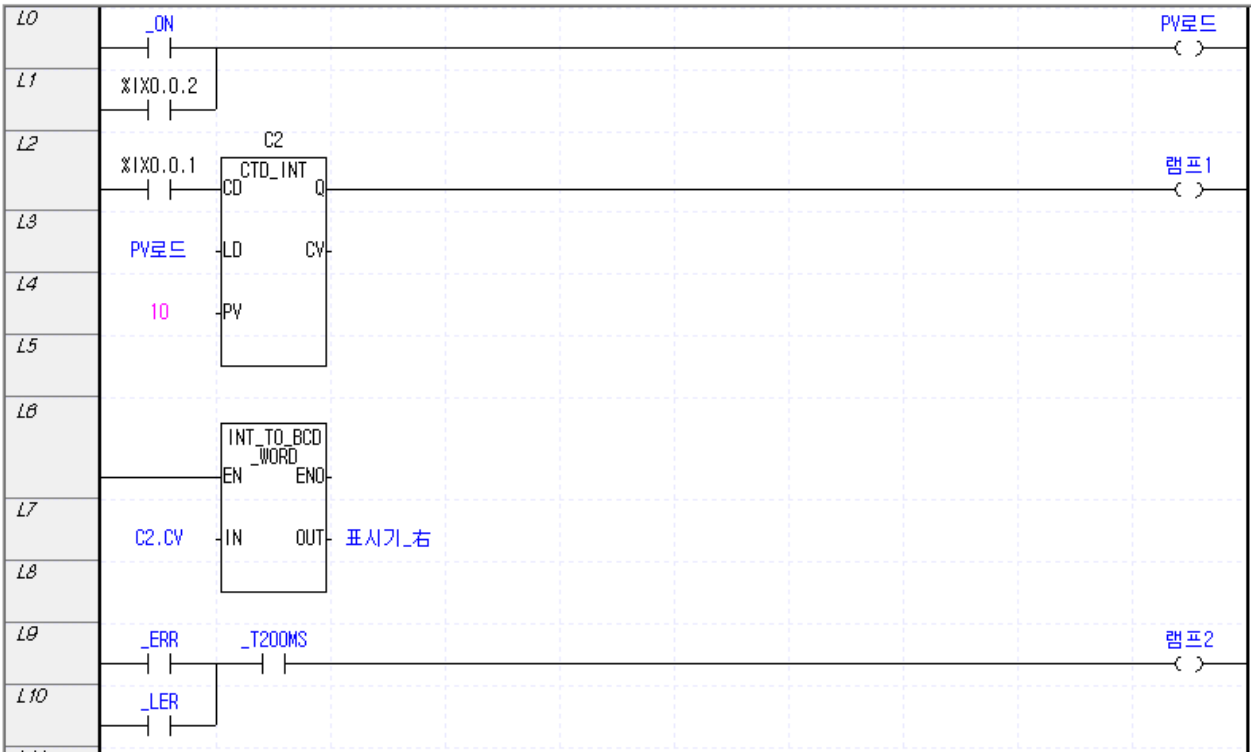
토글 스위치1 (%IX0.0.14)로 CD에 상승(Rising Edge) 펄스를 입력하면 현재값이 감소 합니다.

현재값이 0 이하이면 카운터 출력(C2.Q)이 1이 되어 램프1(%QX0.3.0)이 점등 됩니다.

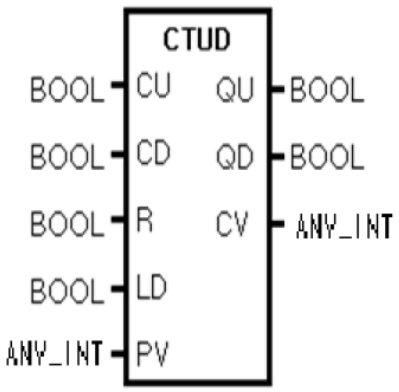
토글 스위치2 (%IX0.0.2)를 ON 하면 LD가 1이 되어 설정값이 현재값에 로드 됩니다.

	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무
1	VAR	C2	CTD_INT			☐	☑
2	VAR	PV로드	BOOL			☐	☑
3	VAR	램프1	BOOL	%QX0.1.1		☐	☑
4	VAR	램프2	BOOL	%QX0.0.2		☐	☑
5	VAR	표시기_右	WORD	%QW0.1.1		☐	☑

Copyright ⓒ 2021 LS Electric Co., Ltd.



▶ CTUD (업-다운 카운터)

펑션	설명
	입력: CU: 업 카운터 펄스 입력 CD: 다운 카운터 펄스 입력 R: 리셋 신호 입력 LD: 설정값 입력 PV: 설정값 출력: QU: 업 카운터 접점 출력 QD: 다운 카운터 접점 출력 CV: 카운터 현재값 출력

① 기능

가감산 카운터 펑션 블록 CTUD는 업카운터 펄스 입력 CU가 0에서 1이 되면 현재값 CV가 이전값보다 1만큼 증가하고, 다운 카운터 펄스입력 CD가 0에서 1 이 되면 현재값 CV가 이전값보다 1만큼 감소하는 카운터 입니다.

단, CV가 PV의 최소값과 최대값 사이의 값을 가지며 최대, 최소값에 이르면 각각 더이상 증가, 감소하지 않습니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

설정값 입력 LD가 1이 되면 현재값 CV에는 설정값 PV값이 로드됩니다. (CV = PV)

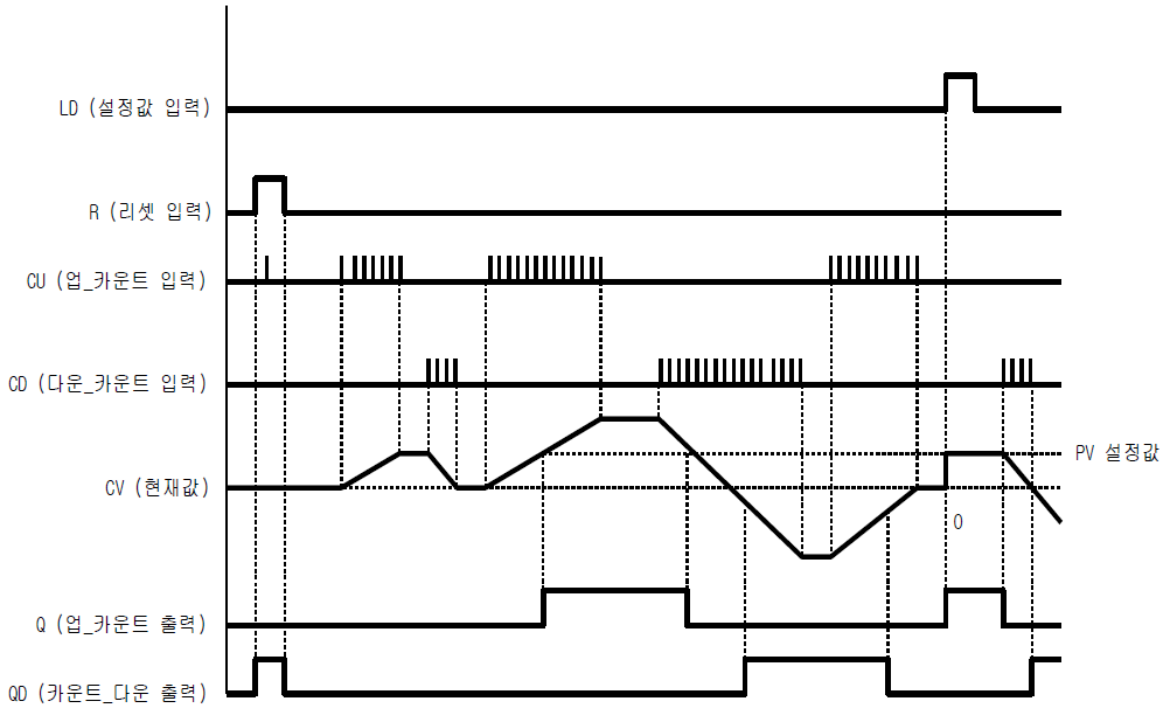
설정값 입력 R이 1이 되면 현재값 CV는 0으로 클리어(Clear) 됩니다. (CV = 0)

출력 QU는 CV가 PV보다 크면 1이 되고, QD는 CV가 0이하일 때 1이 됩니다.

각 입력신호에 대해서 R > LD > CU > CD순으로 동작을 수행하며, 신호의 중복 발생시 우선 순위가 높은 동작 하나만 수행합니다.

펑션 블록	PV	동작 설명
CTUD_INT	INT	INT(설정값)의 -32768 ~ 32767 만큼 증가, 감소합니다.
CTUD_DINT	DINT	DINT(설정값)의 0 ~ 2^{31}-1만큼 증가, 감소합니다.
CTUD_LINT	LINT	LINT(설정값)의 0 ~ 2^{63}-1만큼 증가, 감소합니다.
CTUD_UINT	UINT	UINT(설정값)의 0 ~ 65535 만큼 증가, 감소합니다.
CTUD_UDINT	UDINT	UDINT(설정값)의 0 ~ 2^{32}-1 만큼 증가, 감소합니다.
CTUD_ULINT	ULINT	ULINT(설정값)의 0 ~ 2^{63}-1만큼 증가, 감소합니다.

② 타임 차트



③ 프로그램 예

초기에 _1ON(첫 스캔 ON)에 의해 5가 C3.CV에 전송되어 현재값은 5가 됩니다.

토글 스위치0(%IX0.0.0)로 CU에 상승 펄스를 입력하면 현재값이 증가 합니다.

토글 스위치1(%IX0.0.1)로 CD에 상승 펄스를 입력하면 현재값이 감소 합니다.

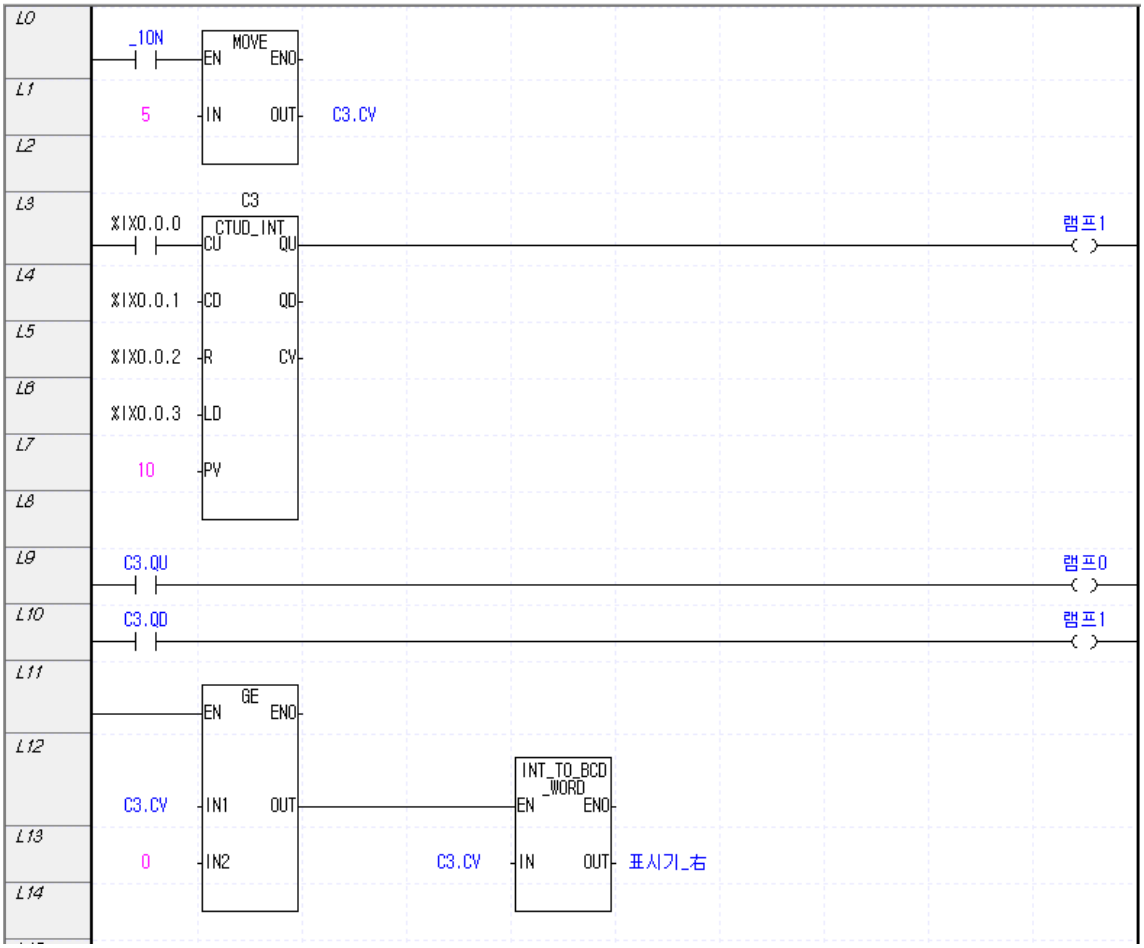
Copyright ⓒ 2021 LS Electric Co., Ltd.

현재값이 설정값 이상이면 C3.QU가 1이 되어 램프0(%QX0.1.0)이 점등 됩니다.

현재값이 0 이하이면 C3.QD가 1이 되어 램프1(%QX0.1.1)이 점등 됩니다.

토글 스위치2(%IX0.0.2)를 ON 하면 리셋되어 현재값은 0으로 클리어(Clear) 됩니다.

현재값이 0이상 이면 GE 펑션 출력이 ON되어 펑션 INT_TO_BCD가 실행됩니다. 단, 현재값 0 ~ 9999 사이를 벗어나면 펑션 INT_TO_BCD에 의해 _ERR, _LER플래그가 ON 됩니다.



	변수 종류	변수	타입	메모리 할당	초기값	리테인	사용유무
1	VAR	C3	CTUD_INT			☐	☑
2	VAR	램프0	BOOL	%QX0.1.0		☐	☑
3	VAR	램프1	BOOL	%QX0.1.1		☐	☑
4	VAR	표시기_右	WORD	%QW0.1.1		☐	☑

Copyright ⓒ 2021 LS Electric Co., Ltd.

6장. 사용자정의 펑션/펑션블록

Copyright ⓒ 2021 LS Electric Co., Ltd.

사용자가 직접 펑션 또는 펑션블록을 디자인 하여 동일한 로직을 재활용 하거나, 노하우를 보호 할 수 있으며, 프로그램 메모리를 효과적으로 활용할 수 있습니다.

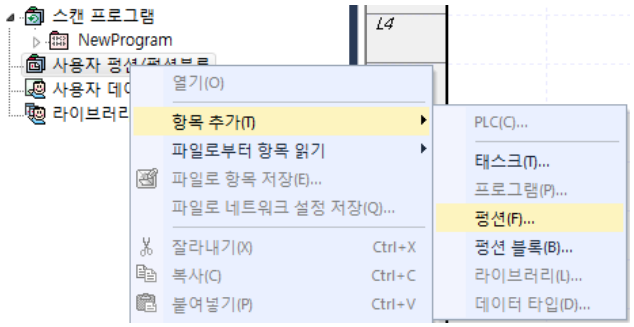
구분	펑션	펑션 블록
입력 개수	한 개 이상	한 개 이상
출력 개수	한 개	한 개 이상
실행	1스캔	1스캔 이상
명령어 예	MOV/BCD ADD/SUB/MUL/DIV EQ/GT/GE/LT/LE etc…	TON (On-delay Timer) CTU (Counter) etc…

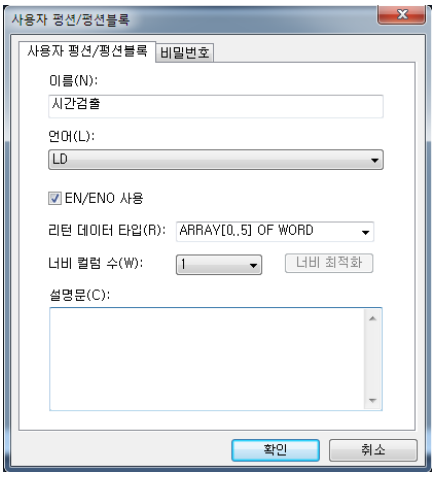
6.1 사용자 정의 펑션

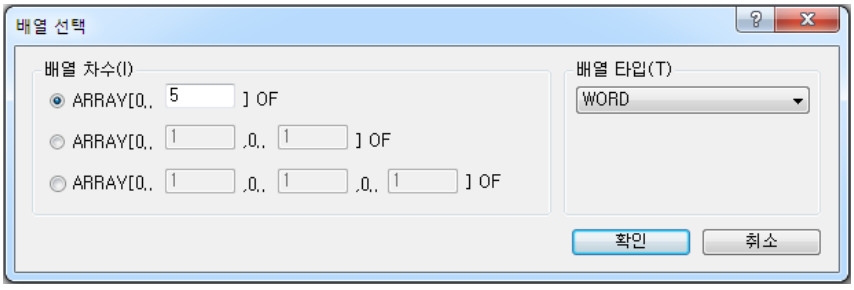
[작성방법]

예제: 트리거 조건 발생 시 현재 시간(년,월,일,시,분초)을 워드타입으로 지정한 디바이스에 저장하는 펑션을 작성합니다.

1. 사용자 정의 펑션/펑션블록을 마우스 우측 버튼을 클릭하여 항목추가>펑션을 선택합니다..
2. 리턴 데이터 타입을 6 개의 Word 데이터로 구성된 Array of Word 로 설정합니다.







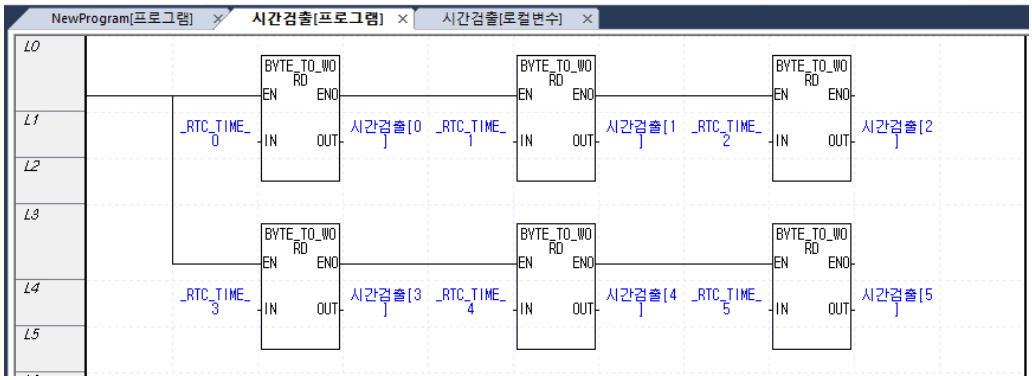
Copyright ⓒ 2021 LS Electric Co., Ltd.

3. 아래와 같이 로컬 변수를 등록하고 사용자 펑션 프로그램을 작성합니다. 스캔 프로그램에서 입력 데이터는 아래와 같이 VAR_INPUT 로 작성하고, 결과값인 시간검출은 VAR_RETURN 에 등록합니다. (입력 데이터는 Byte 타입으로 작성합니다.)

NewProgram[프로그램] | 시간검출[프로그램] | 시간검출[로컬변수]

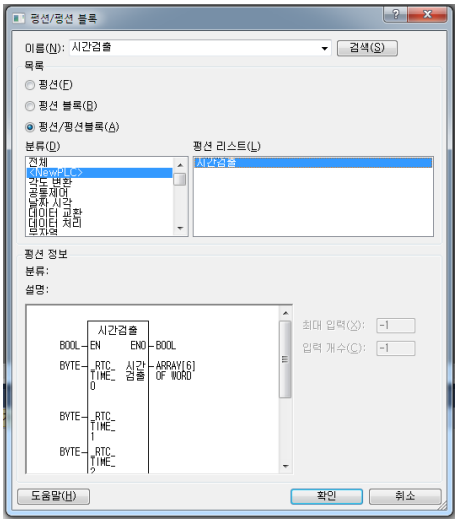
	변수 종류	변수	타입	사용 유무	설명문
1	VAR_INPUT	_RTC_TIME_0	BYTE	☑	_RTC_TIME[0]
2	VAR_INPUT	_RTC_TIME_1	BYTE	☑	_RTC_TIME[1]
3	VAR_INPUT	_RTC_TIME_2	BYTE	☑	_RTC_TIME[2]
4	VAR_INPUT	_RTC_TIME_3	BYTE	☑	_RTC_TIME[3]
5	VAR_INPUT	_RTC_TIME_4	BYTE	☑	_RTC_TIME[4]
6	VAR_INPUT	_RTC_TIME_5	BYTE	☑	_RTC_TIME[5]
7	VAR_RETURN	시간검출	ARRAY[0..5] OF WORD	☑	
8				☐	

<변수 등록>



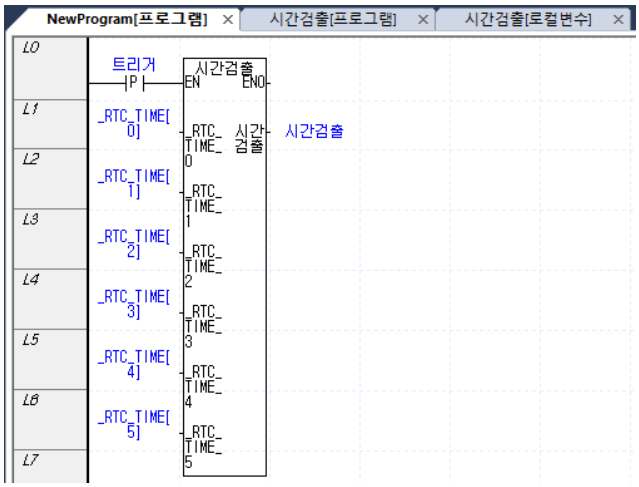
<시간검출 프로그램 작성>

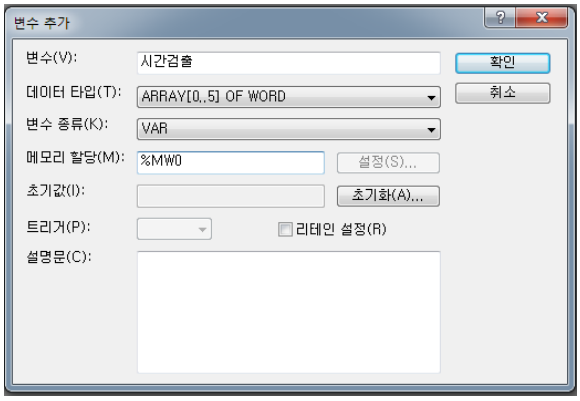
4. 스캔 프로그램에서 사용자 펑션인 "시간검출"을 사용하기 위해 F10 을 선택하고 화면에 클릭 한 후 현재 PLC 이름인 <NewPLC>를 선택하면 펑션리스트가 나옵니다. 여기서 "시간검출" 선택합니다.



Copyright ⓒ 2021 LS Electric Co., Ltd.

5. 아래와 같이 EN 에 트리거 조건(D00000.0)을 연결하고 입력데이터에 날짜관련 시스템 플래그인 _RTC_TIME[0] ~ _RTC_TIME[5]을 순서대로 입력합니다. Return 값인 시간검출 데이터에 어레이 타입으로 %MW0 의 메모리를 할당합니다.





6. 트리거 조건이 On 되면 아래와 같이 날짜 데이터가 %MW0 ~ %MW5 에 저장되는 것을 확인합니다.

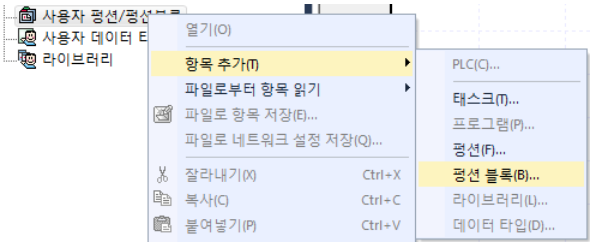
	PLC	프로그램	변수/디바이스	값	타입
1	NewPLC	<GLOBAL>	%MW0	HEX 16#0018	WORD
2	NewPLC	<GLOBAL>	%MW1	HEX 16#0005	WORD
3	NewPLC	<GLOBAL>	%MW2	HEX 16#0030	WORD
4	NewPLC	<GLOBAL>	%MW3	HEX 16#0018	WORD
5	NewPLC	<GLOBAL>	%MW4	HEX 16#0024	WORD
6	NewPLC	<GLOBAL>	%MW5	HEX 16#0045	WORD

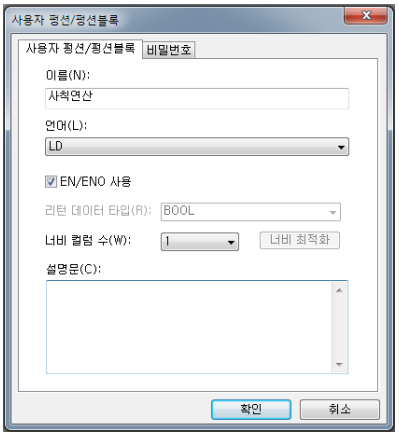
Copyright ⓒ 2021 LS Electric Co., Ltd.

6.2 사용자 정의 펑션블록

[작성방법]

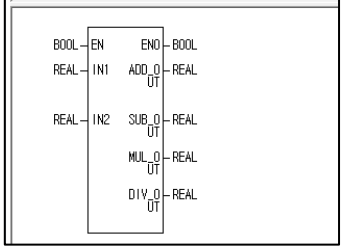
예제) A,B 두 값을 입력하면 덧셈값, 뺄셈값, 곱셈값,나눗셈 값등 4 개의 결과물을 반환해 주는 펑션 블록
이 경우 실행 결과물이 2 개 이상이므로 펑션 블록으로 작성합니다.



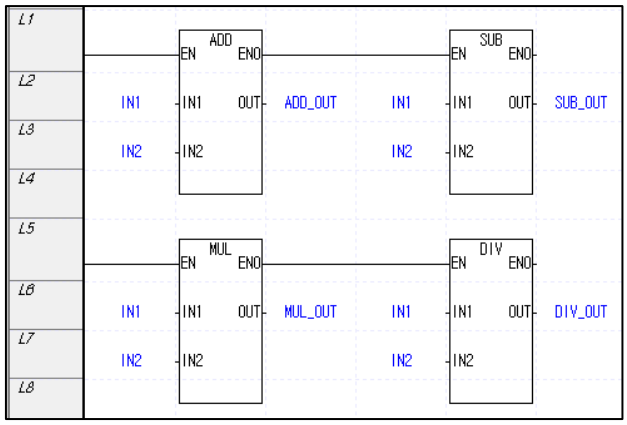


1. 펑션 블록의 내용 및 변수를 등록하여 펑션 블록을 작성합니다.
입출력 변수를 등록함과 동시에 작성되는 펑션 블록의 모양을 바로 확인할 수 있습니다.

	변수 종류	변수	타입
1	VAR_INPUT	IN1	REAL
2	VAR_INPUT	IN2	REAL
3	VAR_OUTPUT	ADD_OUT	REAL
4	VAR_OUTPUT	SUB_OUT	REAL
5	VAR_OUTPUT	MUL_OUT	REAL
6	VAR_OUTPUT	DIV_OUT	REAL

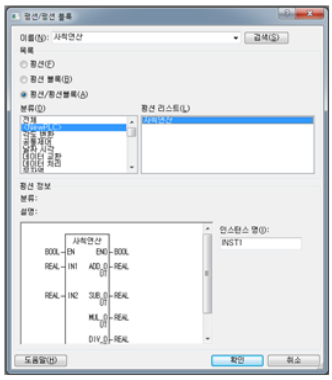


<변수 등록>

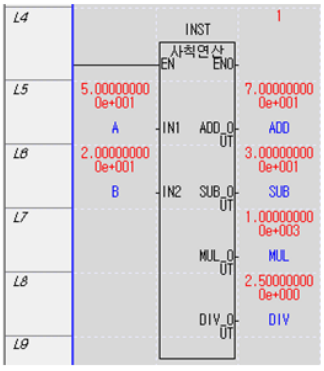


<사칙연산 프로그램 작성>

2. 함수를 프로그램에 등록한 뒤 IN1=50, IN2=20 을 입력하여 실행결과를 확인합니다.



프로그램에 펑션블록 등록



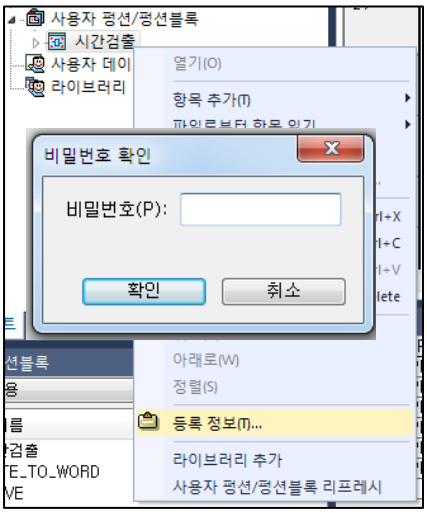
프로그램의 실행 결과 확인

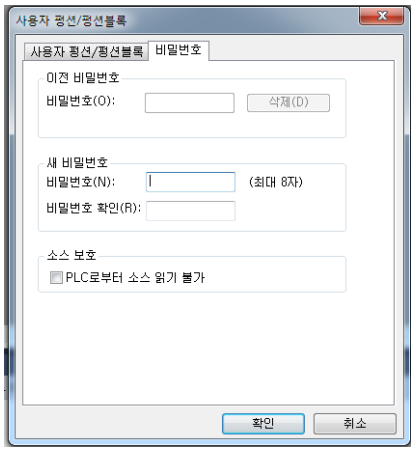
Copyright ⓒ 2021 LS Electric Co., Ltd.

6.3 기타 기능

암호설정

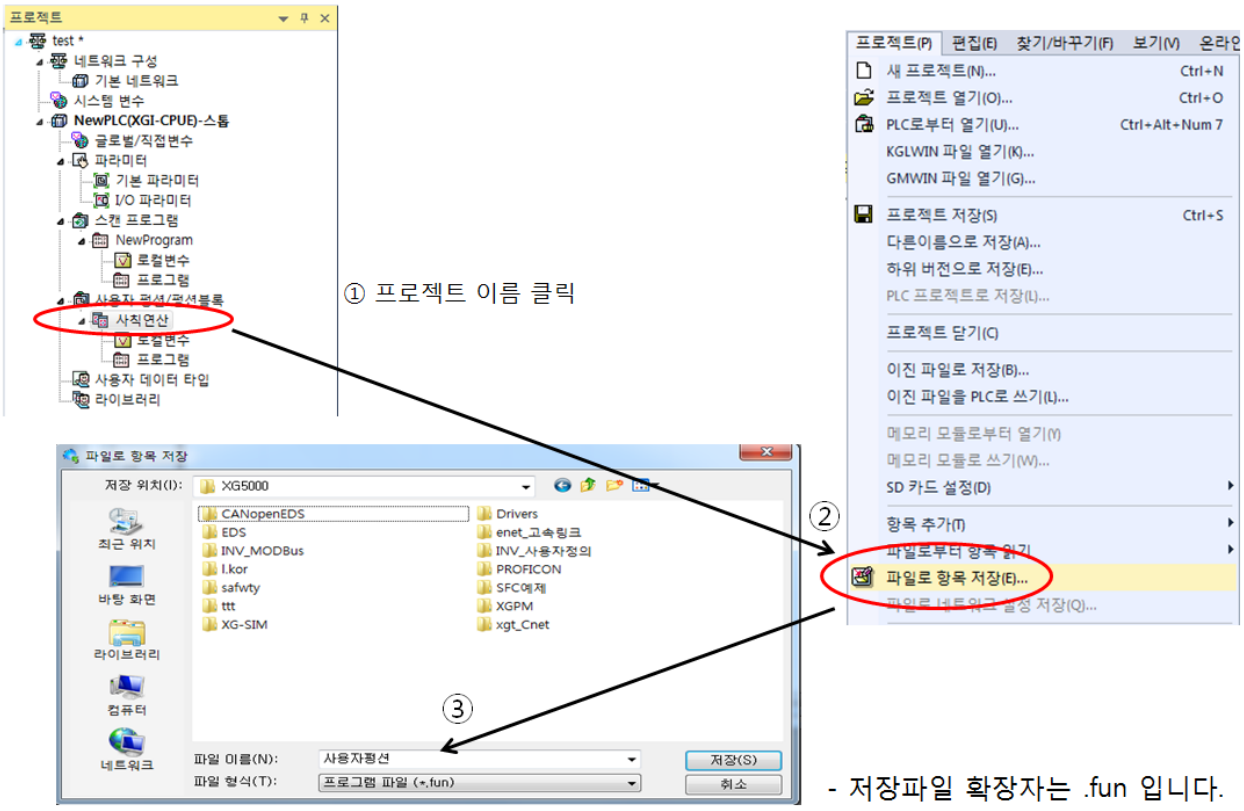
사용자 펑션/펑션블록의 이름을 마우스 우클릭한 후 등록정보를 클릭합니다. 등록정보의 비밀번호 탭에서 비밀번호를 입력합니다. 비밀번호입력을 완료한 후 XG5000 을 저장 후 종료합니다.
프로그램을 다시 불러와 평션/펑션블록의 비밀번호가 설정되었는지 확인합니다.





사용자 펑션/펑션블록 파일로 저장하기

Copyright ⓒ 2021 LS Electric Co., Ltd.



- 저장파일 확장자는 .fun 입니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

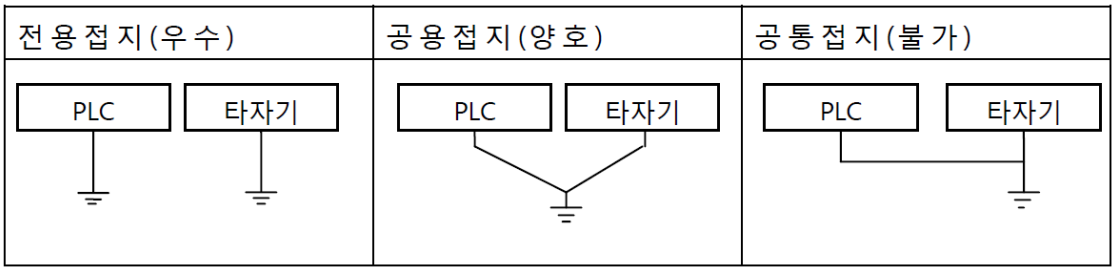
부록 1. 노이즈 대책

① 전원 계통

PLC에 관련되는 전원은 PLC용 전원과 입 · 출력 회로용 전원으로 나눌 수 있습니다. PLC 용 전원은 동력 계통과 별도로 분리하여 사용하는 것이 바람직하며, 차폐 변압기를 사용해 주는 것이 좋습니다. 이 밖에 라인 필터(line filter)에 의해 외부 노이즈를 막아 주면 효과적이며, 입 · 출력 회로용 전원도 차폐 변압기를 이용하여 외부로 부터의 영향을 막아 주는 것이 좋습니다. 출력부는 외부 구동 기기에서 발생한 돌입 전류 및 피크(peak) 전압 등이 PLC에 영향을 줄 수 있으므로 특히 유의해야 합니다.

② 접지

PLC의 접지 단자는 반드시 접지해야 하며, 일반적으로 제 3종 접지 (100[Ω]이하)를 실시하여 이것에 의해 노이즈의 영향을 상당히 줄일 수 있습니다. 그러나 잘못된 접지는 오히려 역효과를 가져올 수 있으므로 주의하여야 합니다.

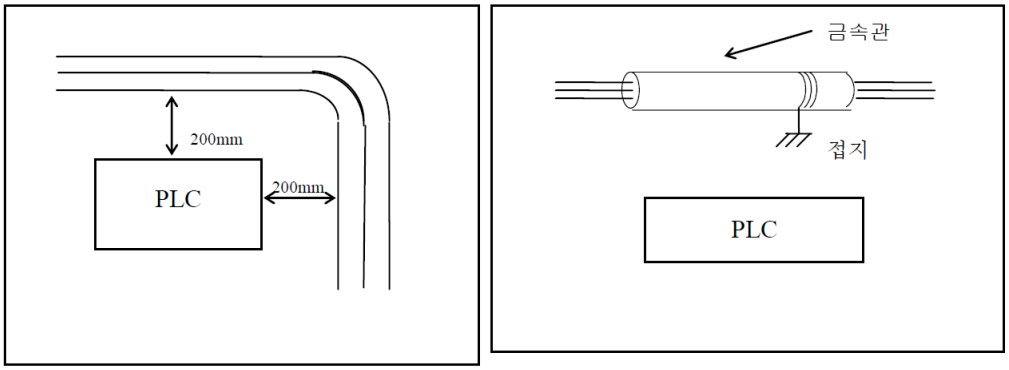


③ 유도 노이즈 대책

선간의 유도에 의한 노이즈를 줄이기 위하여 아래의 배선들을 서로 떼어 놓는 것이 좋습니다.

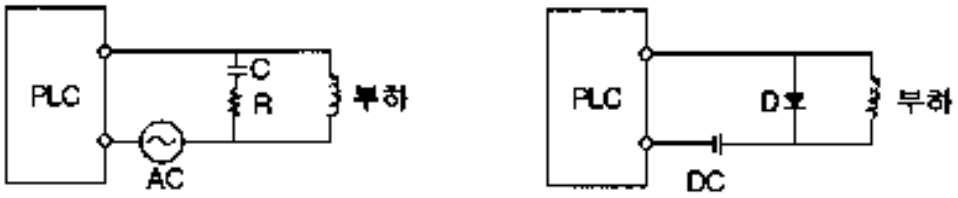
⇒ 동력선과 신호선 ⇒ 입력 신호와 출력 신호
⇒ 아날로그 신호와 디지털 신호 ⇒ 고레벨 신호와 저레벨 신호
⇒ 고속 펄스 신호와 저속 펄스 신호 ⇒ 직류 신호와 교류 신호

특히, 동력선에서 발생한 유도 노이즈가 PLC의 전자 회로에 영향을 미치지 않도록 아래와 같이 격리시켜야 합니다.



Copyright ⓒ 2021 LS Electric Co., Ltd.

PLC 설치 시 접지 및 배선의 유의 사항 등을 지켜주면 노이즈를 어느 정도 줄일 수 있습니다. 이 외에 다른 전자 기기에의 영향을 줄이기 위하여 다음과 같은 대책이 필요합니다. 교류 유도성 부하의 경우, 그림 E-3와 같이 RC 또는 바리스터(Varister)를 부하 양끝에 병렬 접속합니다. 직류 유도성 부하의 경우, 아래와 같이 부하 근처에 다이오드를 역방향으로 병렬 접속한다.



아래는 일반적인 노이즈의 종류입니다.

노이즈의 종류
· 개폐기 노이즈(전자 개폐기, 전자 접촉기등)
· 낙뢰 서지
· 입 · 출력 선이나 제어 신호선, 전송 케이블을 통하여 유도되는 노이즈
· 강전자계 발생원에 의한 전자 유도 노이즈

아래는 일반적인 노이즈의 대책이다.

항목	노이즈 대책
PLC 제어반 내부	· 차단기, 배선용 차단기, 전자 개폐기 등 아크가 발생하는 기기와는 가능한 분리 설치 · 전원선을 꼬아서 사용 · 입력선과 출력선의 분리
PLC와 외부 연결	· 차폐 변압기 · 최단 거리 배선 · 정전압 전원 · 필터
제어반의 외부 배선	· 실드(Shield) 케이블의 사용 · 입력선과 출력선의 분리 · 광 전송 케이블로 전송
접지	· 전용 접지는 우수 · 공용 접지는 양호 · 공통 접지는 불가
기타	· 서지 흡수식 전자 접촉기 · 서지 옵서버의 접속 · 릴레이 코일(DC인 경우)에 다이오드의 접속

Copyright ⓒ 2021 LS Electric Co., Ltd.

부록2. 유지보수

보전이란?

공장 자동화 시스템이 고장 없이 가동되어 생산성을 높이는 것이 PLC 사용의 최대 목표라 할 수 있습니다. PLC은 반도체를 사용한 전자 회로로 반영구적이므로 릴레이 제어반 처럼 예방 보전적인 부품의 교환 처리는 필요하지 않으나 릴레이 출력 카드나 전지 등의 정기적인 교환은 필요합니다. 만일 고장이 나면 모듈을 교환하면 됩니다. 그 밖에 시스템의 고장 요인으로는 다음 7가지가 있습니다.

- PLC의 하드웨어
- PLC의 소프트웨어
- PLC의 제어 및 조작반
- 기계의 검출부
- 기계의 구동부
- 기계의 본체
- 시스템 주변 기기의 환경

장치나 시스템이 가동될 때 그 기능이나 성능을 유지하기 위한 점검, 조정, 대체, 수리 등의 작업을 보전(保全)이라 하는데, 크게 예방 보전과 사후 보전의 2가지가 있습니다. 생산 설비, 항공기 등 경제적 손실이 크거나 중대 사고에 연결되는 것은 예방 보전이 적용되고 일반 제품은 사후 보전이 적용됩니다.

예방 보전

① 일상 점검

점검은 PLC본체에 관한 것과 외부에서 공급되는 전원이나 온도, 습도 등의 주위 환경에 관한 것입니다. 어느 것이나 매일 운전하기 전에 점검하는 것이 바람직하며, 구체적인 점검 항목을 작성하는 것이 필요합니다.

② 정기 점검

1개월, 3개월, 6개월 등의 비교적 긴 시간마다 점검하는 것으로 현상이 천천히 변화해 가기 때문에 매일 점검할 필요가 없는 것에 해당합니다.

Copyright ⓒ 2021 LS Electric Co., Ltd.

사후 보전

① 이상 발견

평소와는 다른 현상으로 동작되는 경우로서 무엇이 이상인지 원인을 명확히 찾아내는 것이 필요합니다. PLC의 자기 진단에 의한 것 외에 사용자 프로그램으로 중요한 동작 과정을 진단하여 기계 장치의 이상 유무를 판단할 수 있습니다. 이 외의 발견 방법으로는 PLC 하드웨어 및 주변 기기의 이상 상태 체크, 기계의 움직임에 의한 이상 상태체크, 제품의 형상이나 생산량에 의한 이상 체크 등이 있습니다.

② 이상 현상과 조치

이상의 발견되면 즉시 복구하여 시스템이 재가동 될 수 있도록 합니다. 이때 주의할 것은 이상이 다른 곳까지 파급되는 경우가 있으므로 다른 곳에서의 영향도 함께 진단할 필요가 있습니다.

PLC 점검 요령

① 점검 항목과 내용

점검항목	점검내용	점검주기	
		일상	정기
주변환경	주변온도, 습도, 먼지, 오일미스트 등을 확인	○	
전원	Maker지정 범위 내 인가 확인	○	
취부상태	Unit(I/O포함) 취부 상태의 느슨해짐 정도,절단	○	
(배선)	단자, 볼트의 조임 확인	○	
	배선 Cable의 손상, 열화확인	○	
	압착단자(cable)의 근접	○	
표지Lamp	동작(상태)표시기의 정상동작 확인	○	
Battery	전압은 정상인가, Maker 보증 기간 내인가 (표시램프, 모니터 등에서 check)		○
Relay	동작시에 「삐리리」音은 없는가		○
FusE	느슨해짐, 절단은 없는가		○
Program (usersoft)	Master Priogram(보관)과 Program내용을 비교, 조합하여 상호확인		○
제어반	냉각Fan 및 Air-Filte의 청소		○
이물제거	먼지, 이물 등을 청소제거		○
예비품	보관 개수 Check		○
	보관환경 Check		○
	동작 Check		○

Copyright ⓒ 2021 LS Electric Co., Ltd.

② 교환 부품

부품명	표준 교환 년 수	교환방법, 기타
Battery	2-3년(단, 수명은 Maker및 종류에 따라 다르다.)	신품과 교환
(전원회로) 평활 콘덴서	5년	신품과 교환 Maker와 상담 후에 결정
Relay류	-	개폐전류, 개폐빈도에 따라 다르기 때문에 Maker규정에 의해 결정
Fuse	10년	신품과 교환

PLC 가 다음의 항목에 해당할 때는 위의 교환 부품의 교환 년 수 단축을 고려할 필요가 있습니다.

※ 온도, 습도가 높은 장소 또는 그 변화가 심한 장소에서 사용할 경우

※ 전원(전압, 주파수, 파형 찌그러짐 등)이나 부하의 변동이 큰 경우

※ 진동, 충격이 심한 장소에 설치된 경우

※ 먼지, 염분, 아황산가스 및 유황수소 등의 나쁜 환경 속에서 사용할 경우

※ 사용 전 보관 환경이 나쁜 경우(장기보존, 장기정지 등)

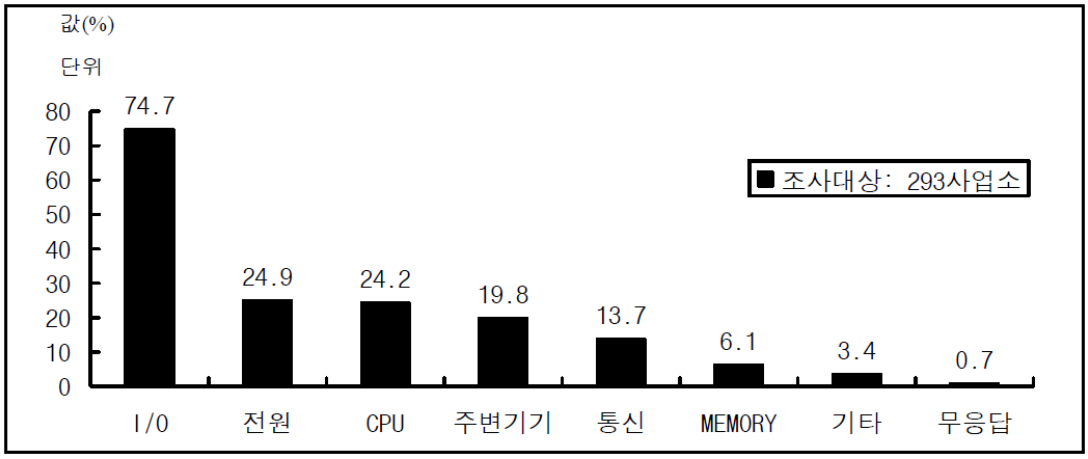
③ 필요한 예비품

NO	품명	수량	비고
1	Batery	1-2개	전지의 보존수명은 약 3년입니다. 1-2개는 예측할 수 없는 경우에 대비합니다.
2	Fuse	사용수	Fuse 는 단락이나 과전류 뿐만 아니라 전원ON/OFF등의 돌입전류에 의해 끊어질 수 있기 때문에 넉넉하게 준비합니다.

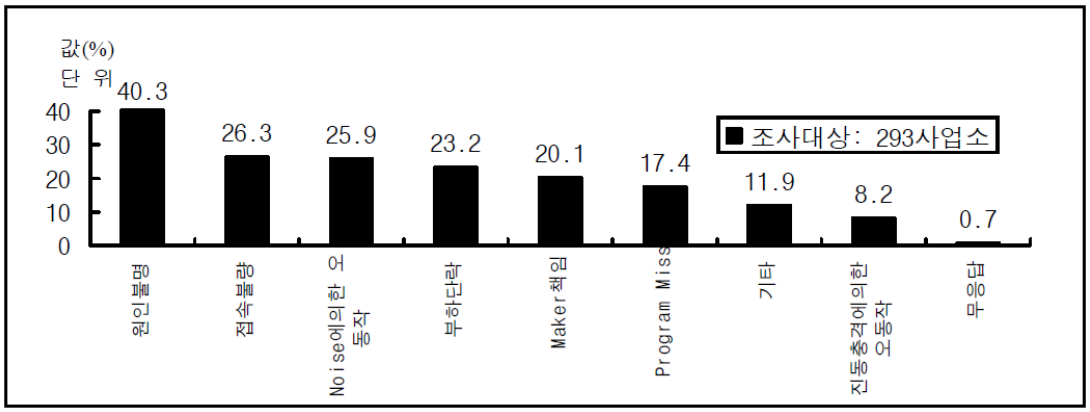
Copyright ⓒ 2021 LS Electric Co., Ltd.

④ 준비 권장 예비품

NO	품명	수량	비고
1	입.출력Unir	Unit의 각명에 붙여 1개	Relay출력 Unit는 접점마모가 있습니다.
2	CPU	1개	PLC의 핵심이 되는 부품이므로 만일 고장이 났을 때에는 System이 Down됩니다.
3	Memory	1개	
4	전원Unit	1개	



PLC의 고장 부위 (복수 응답)



PLC의 고장 원인 (복수 응답)

Copyright ⓒ 2021 LS Electric Co., Ltd.