

# PLC-XGK 초급



본 자료는 PLC-XGK 초급 연수원 교육자료로 XGK 를 제어하기 위해 CPU 와 메모리의 개념을 이해하고 디바이스와 기본 명령어를 공부하여 프로그램을 구성할 수 있는 능력을 배양하기 위한 자료입니다.

본 교재는 XG5000 V4.23 기준으로 작성하였습니다.

## 목 차

### 1 장. PLC 의 개요

1.1 PLC 의 정의 및 적용분야	-----	4
1.2 PLC 의 구조	-----	5
1.3 CPU 연산처리	-----	11
1.4 PLC 프로그래밍 언어	-----	12

### 2 장. XGK 개요

2.1 시스템 구성	-----	14
2.2 XGK PLC 사양	-----	15
2.3 XGK PLC 시스템 구성	-----	18
2.4 XGK PLC 증설 시스템	-----	22
2.5 XGK PLC 입출력 모듈	-----	23
2.6 입출력 메모리 할당	-----	24
2.7 내부 메모리 할당	-----	26
2.8 데이터 래치	-----	30
2.9 운전 모드	-----	31

**3 장. XG5000**

3.1 XG5000 특징	-----	35
3.2 XG5000 개요	-----	36
3.3 XG5000 기본 사용법	-----	37
3.4 XG5000 프로젝트 구조	-----	64

**4 장. 데이터와 프로그램**

4.1 PLC 데이터	-----	78
4.2 접점/코일 프로그램	-----	82
4.3 타이머	-----	95
4.4 카운터	-----	104

<b>부록 1. 수치체계 및 데이터구조</b>	-----	<b>112</b>
---------------------------	-------	------------

<b>부록 2 데이터의 종류 및 사용방법</b>	-----	<b>120</b>
----------------------------	-------	------------

<b>부록 3. XGK 추가 디바이스영역</b>	-----	<b>128</b>
----------------------------	-------	------------

# 1 장. PLC 의 개요

## 1.1 PLC의 정의 및 적용분야

### PLC의 정의

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

### 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 10 가지 조건을 제시하고 이에 가장 부합되는 기기를 만들어 줄 것을 주문하였는데 이 조건이 다음에 기술한 GM의 10 가지 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

### 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.

표 1-1은 PLC 제어 대상에 따른 적용 분야를 나타낸 것입니다.

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

[표 1-1 PLC 적용 분야]

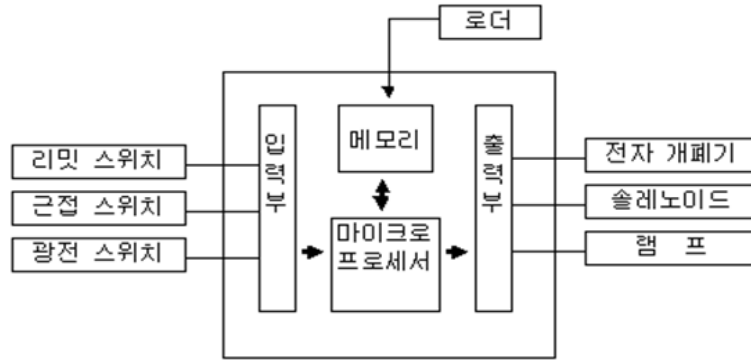
## 1.2 PLC의 구조

### 하드웨어 구조

#### (1) 전체구성

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

그림 1-1은 PLC의 전체 구성도를 나타낸 것입니다.



[그림 1-1 PLC 전체 구성도]

#### (2) PLC의 CPU 연산부

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

#### (3) PLC의 CPU 메모리

##### 메모리 종류

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 불휘발성 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 오프되어도 배터리에 의해 필요한 전원을 공급하여 메모리의 내용이 지워지지 않도록 하는 방법을 배터리 백업(Battery back-up)이라 하는데 이러한 방법을 통하여 RAM도 불휘발성 영역으로 사용할 수 있습니다. PLC의 데이터 영역과 사용자 프로그램은 변경이 가능해야 하므로 RAM 영역에 저장됩니다.

## 메모리 내용

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 실행할 수 있는데 이를 ROM 운전이라 합니다.

데이터 메모리는 입·출력 릴레이, 보조 릴레이, 타이머와 카운터의 접점 상태 및 설정값, 현재값 등의 정보가 저장되는 영역으로 정보가 수시로 바뀌므로 RAM 영역이 사용됩니다.

시스템 메모리는 PLC 제작 회사에서 작성한 시스템 프로그램이 저장되는 영역입니다. 이 시스템 프로그램은 PLC의 명령어를 실행시켜주는 명령어 관련 프로그램과 자기 진단기능등과 같이 PLC 동작시 발생하는 오류나 에러 등을 체크해주는 프로그램, XG5000과의 통신을 담당하는 프로그램 등으로 구성되어 있으며 PLC 제작회사에서 파워를 ON/OFF 하여도 지워지지 않도록 ROM에 저장하여 둡니다.

### (4) PLC의 입·출력부

PLC의 입·출력부는 현장의 기기에 직접 접속하여 사용합니다. PLC 내부는 DC5V의 전원(TTL 레벨)을 사용하지만 입·출력부는 DC24V 또는 AC110V, 220V 등의 높은 전압 레벨을 사용하므로 PLC 내부 회로와 입·출력 회로의 접속(Interface)시 시스템 안정에 매우 많은 영향을 미치게 됨으로 PLC의 입·출력부는 다음과 같은 사항이 필수적으로 요구됩니다.

- 1) 외부 기기와 전기적 규격이 일치해야 합니다..
- 2) 외부 기기로부터의 노이즈가 CPU 쪽에 전달되지 않도록 해야 합니다..  
[광 절연 소자인 포토 커플러(Photocoupler) 사용]
- 3) 외부 기기와의 접속이 용이해야 합니다..
- 4) 입·출력의 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..

입·출력부에 접속되는 외부 기기 예는 표 1-2와 같습니다.

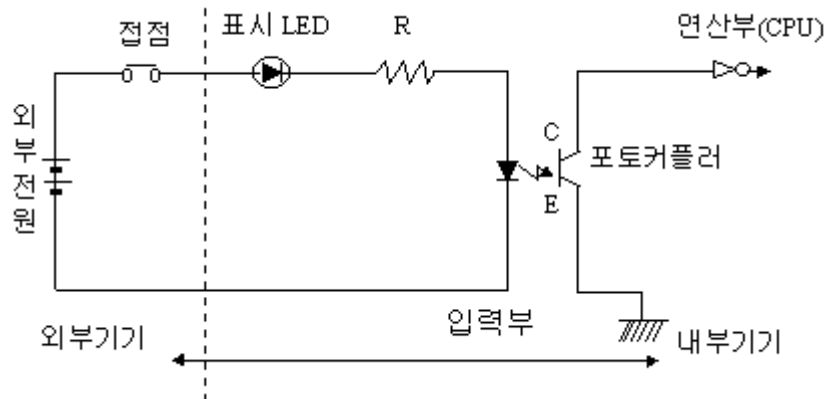
I/O	구 분	부 착 장 소	외부 기기의 명칭
입력부	조작 입력 (명령·지시 입력)	제어반과 조작반	푸시 버튼 스위치 선택 스위치 토글 스위치
	검출 입력 (센서)	기계 장치	리밋 스위치 광전 스위치 근접 스위치 레벨 스위치
출력부	표시·경보 출력	제어반 및 조작반	파일럿 램프 부저
	구동 출력 (액추에이터)	기계장치	전자 밸브 전자 클러치 전자 브레이크 전자 개폐기

[표 1-2 입출력 기기]

가) 입력부

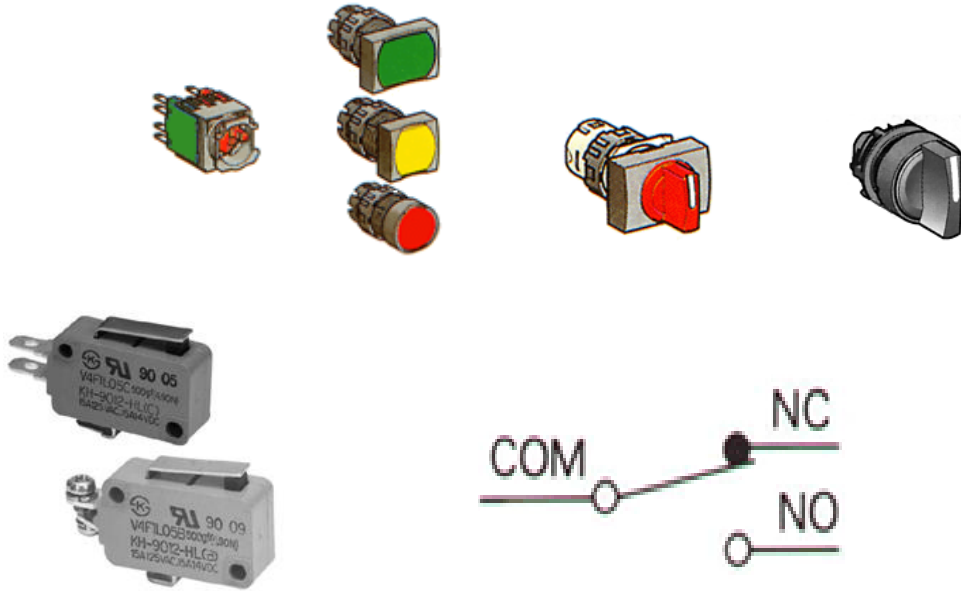
외부 기기로부터의 신호를 CPU의 연산부로 전달해 주는 역할을 합니다. 입력의 종류로는 DC24[V], AC100~240[V] 등이 있고, 그 밖의 특수 입력 모듈로는 아날로그 입력(A/D) 모듈, 고속 카운터(High Speed Counter) 모듈 등이 있습니다.

그림 1-2는 입력부 회로의 예를 나타내었습니다. (입력 회로용 전원: DC24V)



[그림 1-2 DC24V 입력부 회로]

그림 1-3은 PLC의 입력부에 사용되는 각종 스위치류에 대한 사진 및 회로입니다.

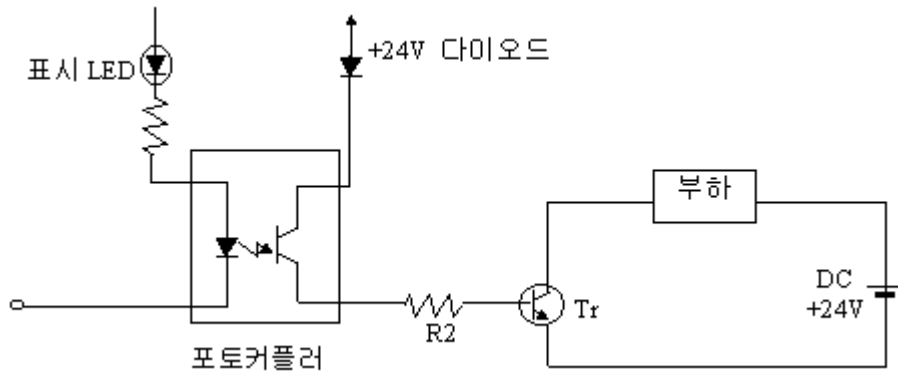


[그림 1-3 입력스위치 종류 및 등가회로]

나) 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 그림 1-4 과 같습니다.



[그림 1-4. 트랜지스터의 출력부 회로]

출력 모듈을 출력 신호와 개폐 소자에 따라 분류하면 표 1-3 과 같습니다.

출력 회로용 전원	개 폐 소 자	
	유 접 점	무접점(반도체)
직류(DC24V)	릴레이 출력	트랜지스터 출력
교류(AC220V)	릴레이 출력	SSR 출력

[표 1-3 출력 모듈의 종류]

표 1-3 에서와 같이 릴레이 출력은 직류와 교류 모두 사용할 수 있으나, 기계적 수명의 한계 때문에 접점의 개폐가 빈번할 경우는 교류 전원 전용인 무접점 SSR 출력이나 직류전원 전용인 트랜지스터

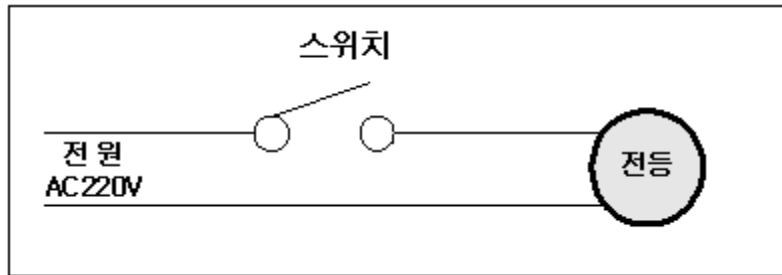
출력을 사용하는 것이 좋습니다.

그림 1-5 는 PLC 의 출력부에 사용되는 릴레이에 대한 사진 및 출력 접점회로입니다.



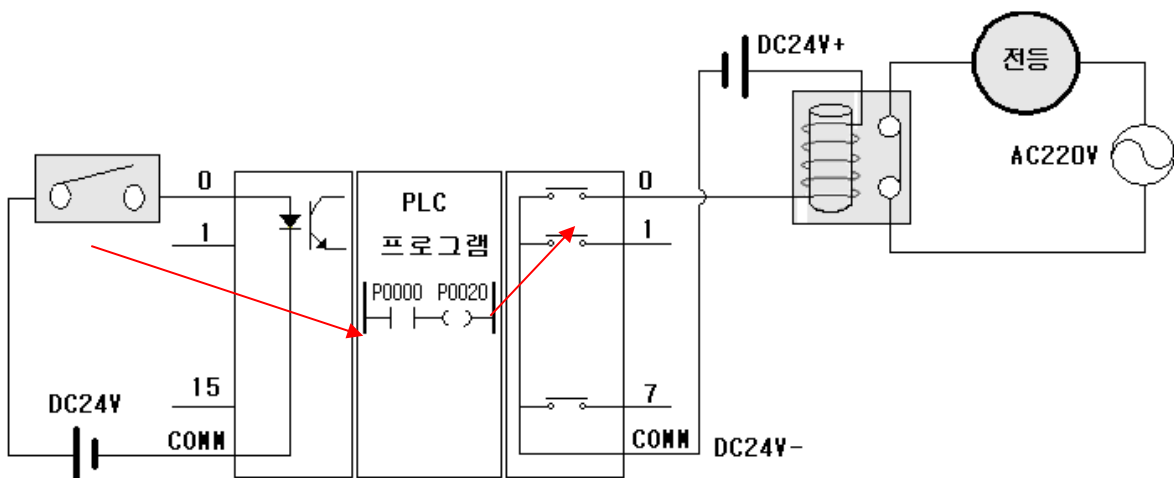
[그림 1-5 출력용 릴레이 종류 및 출력 접점]

그림 1-6 은 일반적으로 사용되는 전등회로입니다. 스위치를 온/오프 함에 따라 전등이 온/오프되는, 우리 주변에서 흔히 볼 수 있는 간단한 전등 온/오프 회로입니다.



[그림 1-6 전기회로 구성]

그림 1-6 의 전기회로를 PLC 로 제어할 경우에는 아래 그림 1-7 처럼 스위치를 입력부, 전등을 출력부로 구분하여 회로를 별도로 구성해야 합니다. 입출력을 별도로 구성하고 입출력간의 연결 구성은 PLC 의 프로그램에 의하여 작성합니다. PLC 의 프로그램은 소프트웨어적으로 처리되기 때문에 수정 및 편집이 자유스럽습니다. 따라서 입출력만 결선하고 나면 프로그램에 의하여 로직 회로를 자유롭게 변경 및 수정이 가능합니다.



[ 그림 1-7 PLC 회로 구성]

소프트웨어 구조

(1) 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 하드와이어드 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.

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, PLC 는 이 방식을 취하고 있습니다.

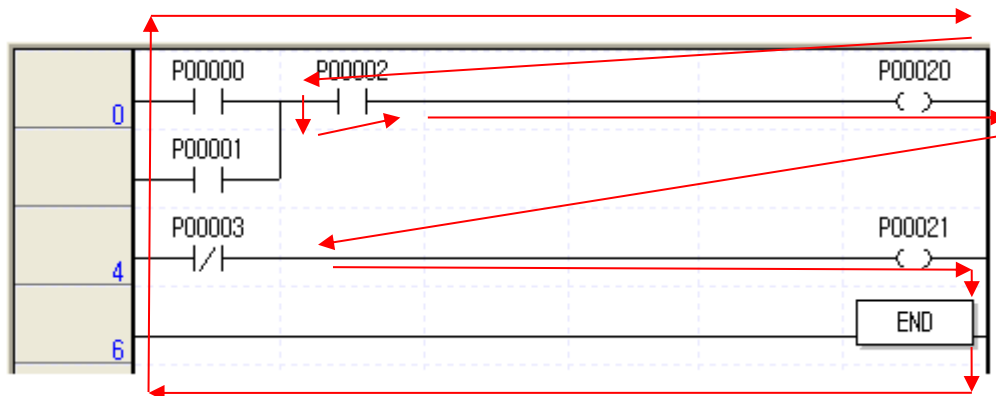
(2) 릴레이 시퀀스와 PLC 프로그램

PLC 는 전자 부품의 집합으로 릴레이 시퀀스와 같은 접점이나 코일은 존재하지 않으며, 접점이나 코일을 연결하는 동작은 소프트웨어로 처리되므로 실제로 눈에 보이는 것이 아닙니다.

또, 동작도 코일 접점이 닫혀 회로가 활성화되는 릴레이 시퀀스와는 달리 메모리에 프로그램을 기억시켜 놓고 순차적으로 내용을 읽어서 그 내용에 따라 동작하는 방식입니다.

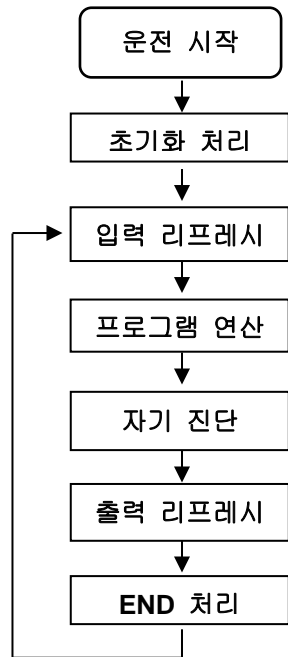
PLC 제어는 프로그램의 내용에 의하여 좌우됩니다. 따라서 사용자는 자유 자재로 원하는 제어를 할 수 있도록 프로그램의 작성 능력이 요구됩니다.

PLC 는 그림 1-8 에서 보는 바와 같이 맨 위 좌측의 명령어부터 우측으로, 그리고 다시 아래방향으로 한 명령어씩 실행하게 됩니다. 이러한 방법으로 모든 명령어를 실행하고 나면 맨 마지막으로 END 명령을 만나게 됩니다. END 명령은 자기진단기능, 타이머, 카운터처리, 통신, 입출력 리프레쉬를 하고서 프로그램 실행 순서를 맨 처음으로 되돌려 주는 역할을 하게 됩니다. 따라서 프로그램 시작 부분으로 되돌아 가면서 동일한 방법으로 반복해서 프로그램을 연속하여 실행하게 됩니다. 이에 대한 자세한 내용은 1.3 장을 참조 하십시오.



[그림 1-8 PLC 시퀀스의 실행 순서]

1.3 CPU 연산처리



- (1) 초기화 처리: PLC 가 RUN 모드로 변경될 때 한 번만 실행하며, 다음과 같은 처리를 수행합니다.
  - 입, 출력 모듈 리셋: 베이스에 설치된 모듈을 리셋하고, 모듈 파라미터에 설정된 운전 방식이 적용됩니다.
  - 데이터 클리어: 데이터 메모리에 저장되어 있는 데이터를 클리어 합니다. 이 때, 래치로 설정된 데이터 메모리 영역의 데이터는 클리어하지 않고 유지합니다.
  - 입, 출력 모듈 주소 할당 및 종류 등록: 입, 출력 모듈의 데이터 메모리 주소를 할당하고, 모듈의 종류를 등록합니다.
  - 초기화 Task 처리: 초기화 Task 프로그램이 등록된 경우 초기화 Task 프로그램의 연산을 실행합니다.
  
- (2) 입력 리프레시: 디지털 입력 모듈의 입력 상태를 읽어 PLC 데이터 메모리의 입력 영역에 저장합니다. 입력 리프레시 과정을 거쳐 입력 데이터 메모리 영역에 저장된 입력 정보는 그 스캔이 완료될 때까지 변경되지 않습니다.
  
- (3) 프로그램 연산: 사용자가 작성한 프로그램을 해석하여 데이터 메모리에 저장된 데이터를 처리합니다. 프로그램은 스캔 프로그램과 Task 프로그램으로 나누어지며, 스캔 프로그램은 PLC 가 RUN 가 RUN 상태이면 수행하는 프로그램이며, Task 프로그램은 PLC 가 RUN 상태에서 Task 의 조건으로 지정된 조건이 만족될 때 수행하는 프로그램이며 내부 디바이스 Task 는 지정된 내부 디바이스의 조건이 만족된 스캔의 스캔 프로그램 수행이 완료된 후 수행되는 프로그램입니다. 프로그램 연산 과정에서 발생하는 출력 데이터는 데이터 메모리의 출력 영역에 저장됩니다.
  
- (4) 자기 진단: PLC 시스템을 진단하여 에러 존재 여부를 판단합니다. 에러가 감지되면 PLC 는 RUN 상태가 되지 않고 에러 상태로 됩니다.

- (5) 출력 리프레시: 데이터 메모리의 출력 영역에 저장된 데이터를 출력 모듈을 통해 출력합니다. 출력 상태의 갱신은 이 때 이루어집니다.
- (6) END 처리: 1 스캔의 처리를 종료한 후 스캔의 처음으로 돌아가기 위한 처리를 합니다.
- (7) 즉시 입출력 명령을 사용한 경우 (IORF): 프로그램 연산 도중 다시 입력 데이터를 갱신하고자 할 때 IORF 명령어를 이용하여 입력 데이터를 리프레시 할 수 있습니다.

**REMARK**

1 Scan: 프로그램을 수행하기 전에 입력 Unit 에서 입력 Data 를 Read 하여 Data Memory 의 입력용 영역(P)에 일괄 저장 후 프로그램 0 번 Step 부터 END 까지 수행하고 자기진단, Timer, Counter 등의 처리를 한 후 Data Memory 의 입력용 영역(P)의 Data 를 출력 Unit 에 일괄 출력하는 일련의 동작.

### 1.4 PLC 프로그래밍 언어

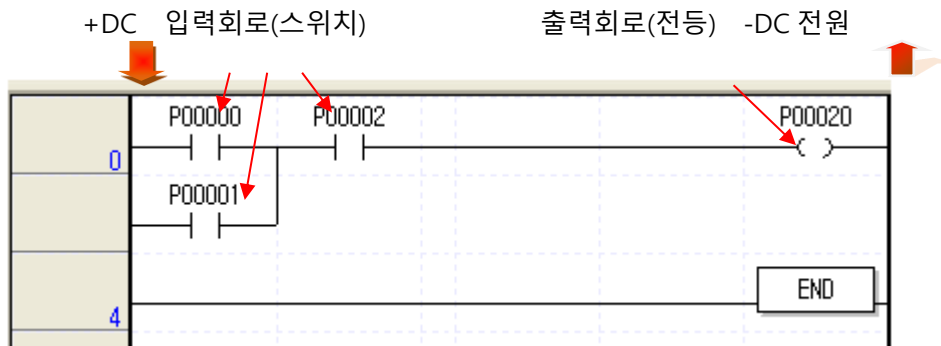
현재 사용중인 프로그래밍 언어로 IL(Instruction List), 래더(Ladder), SFC(Sequential Function Chart), ST(Structured Text)등이 있습니다. XGK PLC는 IL(읽기 전용)과 래더(Ladder) 등의 언어를 제공하며, 상호 호환이 가능 합니다.

(1) IL(Instruction List)

MASTER-K PLC 에서 니모닉(Mnemonic)이라고 불린 언어이며, 어셈블리어 형태의 문자 기반 언어로 MASTER-K 에서는 휴대용 프로그램 입력기(Handy Loader)를 이용하여 현장에서 간단한 로직의 프로그래밍에 주로 사용되었습니다만 요즘에는 노트북이 이를 대체함으로써 굳이 IL로 작성할 필요성이 사라짐으로써 거의 사용되지 않는 언어 입니다.

(2) 래더(Ladder): 사다리도

사다리 형태로 전원을 생략하여 로직을 표현하는 릴레이 로직과 유사한 도형기반의 언어로, 현재 가장 널리 사용되고 있습니다. (PLC 언어를 대표함)



( 래더 프로그램 예 )

## 2 장. XGK 개요

## 2.1 시스템 구성

PLC 단위시스템은 베이스(Base), 전원부(SMPS), CPU 부, Digital 입·출력부(Di, Do) 포함한 기본구성에 특수, 통신모듈 등을 추가한 시스템으로 구분할 수 있습니다. 위 구성을 하나의 제품에 포함한 TYPE 을 **블록형** 이라 합니다. 이에 속하는 기종으로 XGB 시리즈가 있습니다. 이 밖에 각각의 구성품으로 이루어진 TYPE 을 **모듈형**이라고 하며 위의 기종을 제외한 전제품이 포함됩니다.



블록형 (XGB 시리즈)



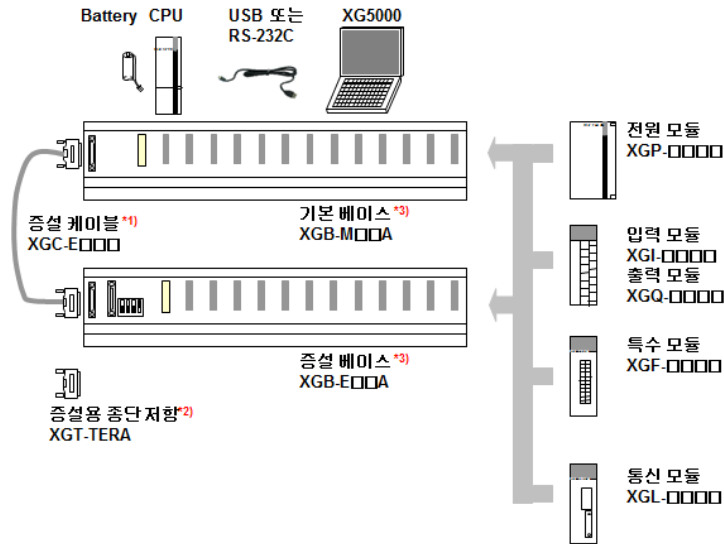
모듈형 (XGK, XGI 시리즈)

### 모듈형의 구성

위 그림과 같이 기본 시스템은 전원부가 가장 좌측에 위치하며, 다음에 CPU 부, 이후에 입출력부가 위치 하게 됩니다. 각각의 구성품은 베이스 위에 장착되며 베이스의 슬롯 수는 **전원부와 CPU** 를 제외한 슬롯 수로 표시합니다.

### PLC 증설 시스템

XGK-CPU의 경우에 최대 1,536 점까지 제어가 가능합니다. 베이스당 최대 12 슬롯이 장착 가능하므로 12 슬롯 베이스에 64 점 모듈(XGK 1 모듈당 최대 점수)을 모두 장착하였을 경우 총 768 점(12 X 64)으로 최대 제어점수 1,536 점의 1/2 밖에 안됩니다. 1,536 점을 모두 하나의 베이스에서 제어할 경우 24 슬롯의 베이스가 필요하며, 이런 베이스를 사용할 경우에 전원부의 용량, 외형의 크기가 커져서 사실상 사용이 불가능 해집니다. 증설 시스템이란 위와 같은 문제에 대한 해결책으로 24 슬롯의 베이스를 12 슬롯 단위로 나눈 후 각 베이스간을 증설 케이블로 연결하여 마치 하나의 베이스 위에서 모든 입출력을 제어하는 효과를 나타냅니다. 이와 같은 경우 CPU 가 장착된 베이스를 기본 베이스라고 하며 전원, CPU, 입출력이 장착되고 증설 케이블 취부용 Connector 1 개를 포함합니다. 증설 베이스에는 CPU 없이 전원, 입출력으로만 구성되고 증설 케이블 취부용 Connector 2 개를 포함합니다.



XGK PLC 시스템 구성 시 아래 사항을 주의 하시기 바랍니다.

- 1) 증설 케이블 길이의 총합은 15m 이하로 제한됩니다.
- 2) 증설 베이스를 연결할 경우 증설 길이에 관계없이 마지막 베이스에 종단 저항을 반드시 설치해야 합니다.
- 3) 베이스에 모듈을 장착하지 않을 경우 더미 모듈(XGT-DMMA)를 장착하여 슬롯 커넥터를 보호해 주시기 바랍니다.

## 2.2 XGK PLC 사양

### XGK PLC의 특징

XGK를 포함한 XGT 시리즈 PLC는 다음과 같은 특징이 있습니다.

- 1) 업계 최고 수준의 CPU 처리속도 (8.5ns/step)와 향상된 Backplane Interface를 통해 초고속 연산 속도를 보장하며 USB를 통한 고속 업/다운로드는 유지/보수의 편리성을 극대화 합니다.
- 2) 동급 최소 사이즈의 (모듈크기 27 \* 98 \* 90) 각 구성품은 Compact한 Panel 제작을 통해 원가 절감 및 다양한 Application 분야에의 적용을 가능하게 합니다.
- 3) Fast Ethernet 및 Open Fieldbus에 기반한 System은 고 신뢰도의 고속 전송을 가능하게 하고 다양한 Field Device interface를 통해 System 구성상의 한계를 극복합니다.
- 4) Fast Ethernet 기술 기반의 전기/광 링 네트워크 구축으로 네트워크의 신뢰도를 대폭 향상시킬 수 있습니다.
- 5) Ethernet 기반으로 산업계 표준 Field Network로 자리매김 하고 있는 EtherNet/IP, Modbus TCP/IP를 지원함으로써 Ethernet 기반으로 타사 장비와 손쉬운 접속이 가능합니다.
- 6) XG5000 Package는 향상된 사용자 Interface와 편리한 조작성을 바탕으로 Multi PLC Multi Programming을 지원하고 다양한 모니터링 및 강력한 진단 기능을 제공합니다.
- 7) 프로그램의 작성 없이 파라미터 설정으로 특수/통신 모듈의 운전 설정 가능합니다.
- 8) 한 대의 PLC에 최대 256개의 프로그램을 작성하고, 각 프로그램을 Export/Import 할 수 있어 프로그램의 모듈화 가능합니다.

- 9) Dtag & Drop, Auto Fill, 특수 모듈 변수/설명문 자동 등록 기능 등 다양한 기능으로 프로그램 편의성을 증대시켰습니다.
- 10) XG5000의 모든 데이터는 Microsoft EXCEL과 호환이 가능합니다.

**일반 사양**

XGK를 포함한 XGT 시리즈에 공통적으로 적용되는 일반 사양은 다음과 같습니다.

항목		일반 사양				
사용/보관 온도		0 ~ 55 °C / -25 ~ 70 °C				
사용/보관 습도		5 ~ 95%RH, 이슬이 맺히지 않을 것.				
내진동	구분	주파수	가속도	진폭	횟수	
	단속적 진동	10 ≤ f < 57Hz	-	0.075mm	X,Y, Z 각 방향 10 회	
		57 ≤ f ≤ 150Hz	9.8m/s <sup>2</sup> {1G}	-		
	연속적 진동	10 ≤ f < 57Hz	-	0.035mm		
		57 ≤ f ≤ 150Hz	4.9m/s <sup>2</sup> {0.5G}	-		
내충격	<ul style="list-style-type: none"> <li>• 최대 충격 가속도:147 m/s<sup>2</sup>{15G}</li> <li>• 인가 시간: 11ms</li> <li>• 펄스 파형: 정현 반파 펄스 (X, Y, Z 각 방향 3 회)</li> </ul>					
내노이즈	방형파 임펄스 노이즈			± 1,500 V		
	정전기 방전			전압: 4kV (접촉 방전)		
	방사 전자계 노이즈			27 ~ 500 MHz, 10 V/m		
	패스트 트랜지언트 & 버스트 노이즈			구분	전원	모듈
			전압	2kV	1kV	
주위 환경	부식성 가스가 없을 것.					
사용 고도	2,000m					
오염도	2 이하					
냉각 방식	자연 공랭식					

**XGK 사양**

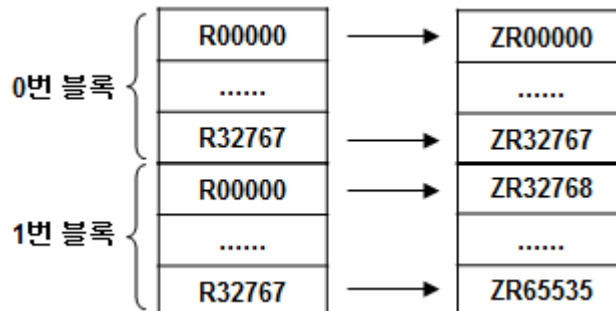
표준형 CPU 모듈(XGK-CPUE/S)과 고급형 CPU 모듈(XGK-CPUA/H/U)의 기능사양은 다음과 같습니다.

항 목		XGK 기 능 사 양							
		XGK-CPUE	XGK-CPUS	XGK-CPUA	XGK-CPUH	XGK-CPUU	XGK-CPUSN	XGK-CPUHN	XGK-CPUUN
연산 방식		반복연산, 정주기 연산, 고정주기 스캔							
입출력 제어 방식		스캔동기 일괄처리 방식 (리프레시 방식), 명령어에 의한 다이렉트 방식							
프로그램 언어		래더 다이어그램(Ladder Diagram), 명령 리스트(Instruction List) SFC(Sequential Function Chart), ST(Structured Text)							
명령어 수	기본	40 개							
	응용	232 종 717 개							
연산처리 속도 (기본명령)	LD	0.084 $\mu$ s/Step		0.028 $\mu$ s/Step		0.0085 $\mu$ s/Step			
	MOV	0.252 $\mu$ s/Step		0.084 $\mu$ s/Step		0.0255 $\mu$ s/Step			
	실수 연산	$\pm$ : 1.44 $\mu$ s(S), 2.87 $\mu$ s(D) x: 1.94 $\mu$ s(S), 4.186 $\mu$ s(D) $\div$ : 1.97 $\mu$ s(S), 4.2 $\mu$ s(D)		$\pm$ : 0.602 $\mu$ s(S), 1.078 $\mu$ s(D) x: 1.106 $\mu$ s(S), 2.394 $\mu$ s(D) $\div$ : 1.134 $\mu$ s(S), 2.66 $\mu$ s(D)		$\pm$ : 182.2 ns(S), 327.3ns(D) x: 336 ns(S), 427ns(D) $\div$ : 345 ns(S), 808ns(D)			
프로그램 메모리		16kstep	32kstep	32kstep	64kstep ep	128kstep	64kstep	128kstep	256kstep
입출력점수		1,536 점	3,072 점	3,072 점	6,144 점		3,072 점	6,144 점	
데이터 영역	P	P00000 ~ P2047F (32,768 점)				P00000 ~ P4095F (65,536 점)			
	M	M00000 ~ M2047F (32,768 점)				M00000 ~ M4095F (65,536 점)			
	K	K00000 ~ K2047F (32,768 점)				K00000 ~ K4095F (65,536 점)			
	L	L00000 ~ L11263F (180,224 점)				L00000 ~ L11263F (180,224 점)			
	F	F00000 ~ F2047F (32,768 점)				F00000 ~ F4095F (65,536 점)			
	T	100ms: T0000 – T0999 10ms: T1000 – T1499 1ms: T1500 – T1999 0.1ms: T2000 – T2047				100ms: T0000 – T2999 10ms: T3000 – T5999 1ms: T6000 – T7999 0.1ms: T8000 – T8191			
	C	C0000 ~ C2047				C0000 ~ C4095			
	S	S00.00 ~ S127.99				S00.00 ~ S255.99			
	D	D0000 ~ D19,999		D0000 ~ D32,767		D0000 ~ D262,143		D0000 ~ D524,287	
	U	U0.0~U1F.3 1	U0.0~U3F.3 1	U0.0~U3F.3 1	U0.0~U7F.31		U0.0~U3F.3 1	U0.0~U7F.31	
	Z	128 점				256 점			
N	N00000 ~ N21,503								

	R	1 블록(R0 ~ R32767)	2 블록(R0 ~ R32767, 2 개)		8 블록	16 블록	
플래시 영역		2 Mbyte, 32 블록					
항 목		XGK 기 능 사 양					
		XGK-CPUE	XGK-CPUS	XGK-CPUA	XGK-CPUH	XGK-CPUU	XGK-CPUSN
프로그램 구성	총 프로그램 수	256 개					
	초기화 태스크	1 개					
	정주기 태스크	32 개					
	내부 디바이스 태스크	32 개					
운전모드		RUN, STOP, DEBUG					
자기진단 기능		연산지연감시, 메모리 이상, 입출력 이상, 배터리 이상, 전원이상 등					
프로그램 포트		RS-232C(1CH), USB(1CH)			Ethernet(1CH), USB(1CH)		
정전 시 데이터 보존방법		기본 파라미터에서 래치영역 설정					
최대 베이스 확장		1 단	3 단	7 단	3 단	7 단	
내부 소비 전류		940mA		960mA			
중 량		0.12kg					

메모리 사용 시 주의사항은 다음과 같습니다.

- 1) CPU에서 PID 연산 기능을 사용할 때 K1000 워드부터 PID 연산 파라미터 설정 및 PID 운전 결과가 저장되는 영역이므로 CPU 에서 PID 연산 기능을 사용할 경우 K 영역 사용에 제한이 있습니다.
- 2) R 영역과 ZR 영역은 동일한 데이터 메모리 영역이며, R 은 1 개 블록 내의 주소, ZR 은 전체 블록 내의 주소입니다. R 을 사용할 경우 블록 번호는 RSET 명령어를 이용하여 지정합니다.
- 3) N 영역은 XG-PD 에서 설정하는 P2P 통신 파라미터 저장 영역으로 P2P 통신 관련 명령어(P2PSN, P2PWRD, P2PWWR, P2PBRD, P2PBWR) 이외의 명령어를 이용하여 접근할 수 없습니다.

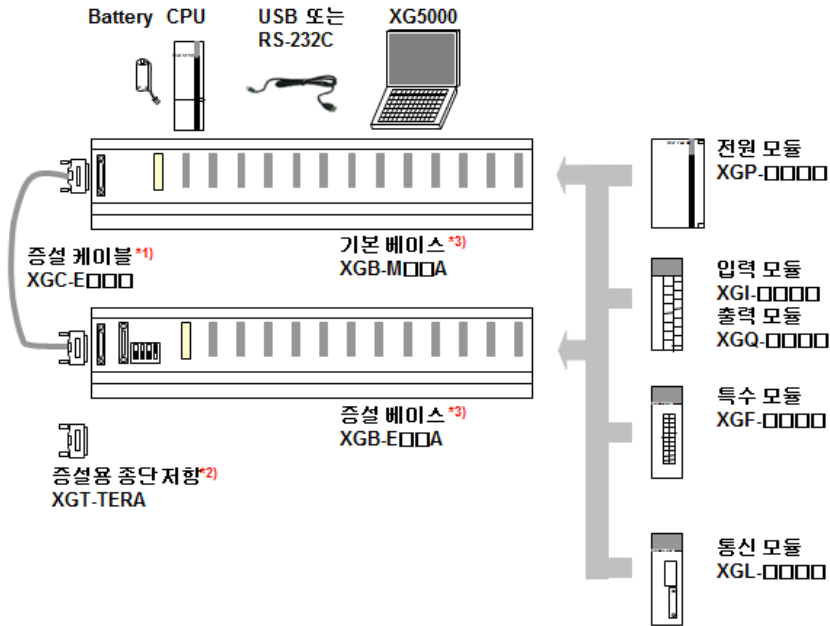


### 2.3 XGK PLC 시스템 구성

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

### XGK 시스템 구성

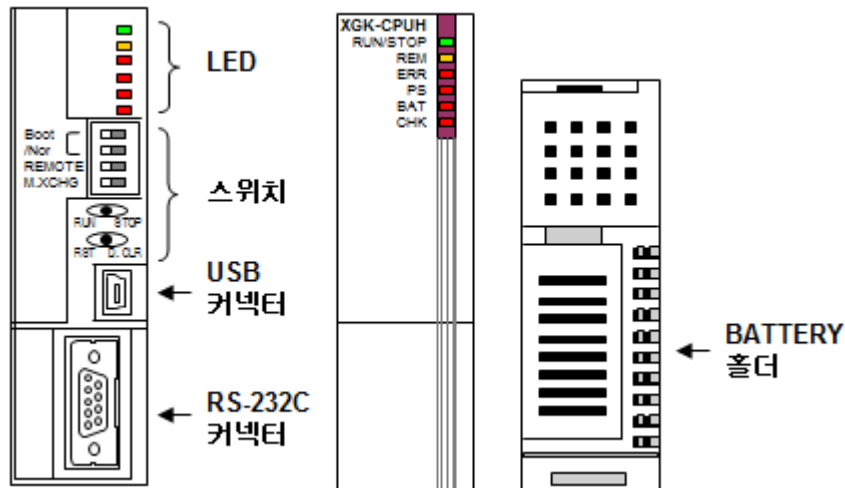
XGK 시리즈의 시스템 구성은 아래 그림과 같습니다.



- 1) 증설 케이블 길이의 총합은 15m 이하로 제한됩니다.
- 2) 증설 베이스를 연결할 경우 증설 길이에 관계없이 마지막 베이스에 종단 저항을 반드시 설치해야 합니다.
- 3) 베이스에 모듈을 장착하지 않을 경우 더미 모듈(XGT-DMMA)을 장착하여 슬롯 커넥터를 보호해 주시기 바랍니다.

### CPU 표시 사양

XGK 시리즈의 CPU 전면부 LED 표시 내용은 다음과 같습니다.



RS-232C 커넥터: XG5000 접속용으로 사용되며, XG5000 접속 포트 외 Modbus 서버 기능을 지원하는 포트가 내장되어 있으며 통신 파라미터는 XG5000의 기본 파라미터에서 설정합니다.

- XG5000 접속용 포트: 2(Rx), 3(Tx), 5(SG)
- Modbus 통신용 포트: 8(Rx), 7(Tx), 5(SG)

XGK-CPUSN, XGK-CPUHN, XGK-CPUUN 은 RS-232C 대신 이더넷 포트가 내장되어 있습니다. XG5000 접

속 및 XGT 서버, 모드버스 TCP 서버 기능을 수행할 수 있습니다. 클라이언트 기능은 내장 되어 있지 않습니다.

No.	명 칭	용 도
①-a	RUN/STOP LED	<p>CPU 모듈의 동작 상태를 나타냅니다.</p> <ul style="list-style-type: none"> <li>• 녹색 점등: 'RUN' 모드 상태로 운전 중 을 표시</li> <li>▶ RUN/STOP 모드 스위치에 의해 'RUN' 운전 중</li> <li>▶ RUN/STOP 모드 스위치가 'STOP' 인 상태에서 '리모트 RUN' 운전 중</li> <li>• 녹색 점멸: RUN 운전 중에 경고 또는 에러 표시</li> <li>• 적색점등: 'STOP' 모드 상태로 운전 중을 표시</li> <li>▶ RUN/STOP 모드 스위치에 의해 'STOP' 운전 중</li> <li>▶ 모드 스위치가 'STOP' 인 상태에서 리모트 'STOP' 운전 중</li> <li>• 적색점멸: 'STOP' 운전 중에 경고 또는 에러 표시</li> <li>▶ 운전을 정지하는 에러를 검출한 경우</li> </ul>
①-b	REM LED	<ul style="list-style-type: none"> <li>• 점등(황색): 리모트 허용 상태임을 표시</li> <li>▶ 'REMOTE' 스위치가 'On' 인 경우</li> <li>• 소등: 리모트 금지 상태임을 표시</li> <li>▶ 'REMOTE' 스위치가 'Off' 인 경우</li> </ul>
①-c	ERR LED	<ul style="list-style-type: none"> <li>• 점등(적색): 운전이 불가능한 에러가 발생한 경우를 표시</li> <li>• 소등: 이상 없음을 표시</li> </ul>
①-d	PS LED (Programmable Status)	<ul style="list-style-type: none"> <li>• 점등(적색):</li> <li>▶ '사용자 지정 플래그' 가 'On' 인 경우</li> <li>▶ '에러시 운전 속행' 설정으로 에러 상태에서 운전 중인 경우</li> <li>▶ 'M.XCHG' 스위치가 'On' 인 상태에서 모듈을 빼거나 다른 모듈을 장착한 경우</li> <li>• 소등: 이상 없음을 표시</li> </ul>
①-e	BAT LED	<ul style="list-style-type: none"> <li>• 점등(적색): 배터리 전압이 저하된 경우</li> <li>• 소등: 배터리 이상 없음</li> </ul>
①-f	CHK LED	<ul style="list-style-type: none"> <li>• 점등(적색): 표준설정과 다른 내용이 설정되어 있는 경우에 표시 (파라미터로 추가/삭제[해제]가 가능함)</li> <li>▶ '모듈교체' 스위치가 '모듈교체'로 설정 된 경우</li> <li>▶ '디버그 모드' 에서 운전 중 인 경우</li> <li>▶ '강제 ON' 설정 상태</li> <li>▶ '고장마스크', 'SKIP' 플래그가 설정 된 경우</li> <li>▶ 운전 중 경고장(Warning)이 발생한 경우</li> <li>▶ 증설베이스 전원 이상</li> <li>• 점멸: 연산에러시 운전속행 설정이 되어 있는 상태에서 에러가 발생한 경우</li> <li>• 소등: 표준설정으로 운전 중에 표시</li> </ul>

CPU 전면부 기능

XGK 시리즈의 CPU 전면부 스위치 및 부속 내용은 다음과 같습니다.

No.	명 칭	용 도
②-a	Boot/Nor 스위치	<p>출하 전 O/S 를 다운로드 하는 경우 사용합니다.</p> <ul style="list-style-type: none"> <li>On (우측): 정상운전 모드에서 제어동작을 수행</li> <li>Off (좌측): 제조 시 사용하는 모드로 사용자 조작 금지 (O/S 의 다운로드 모드)</li> </ul> <p>• 주의: Boot/Nor 스위치는 항상 On(우측)상태로 유지해야 합니다. Off(좌측) 상태로 설정하게 되면 모듈 소손의 원인이 됩니다.</p>
②-b	REMOTE 허용 스위치	<p>리모트 접속을 통한 PLC 의 동작을 제한 합니다.</p> <ul style="list-style-type: none"> <li>On(우측): 모든 기능 허용 (REMOTE 모드)</li> <li>Off(좌측): 리모트 기능 제한</li> </ul> <p>▶ 프로그램의 D/L, 운전모드 조작 제한 ▶ 모니터, 데이터 변경 등은 조작 허용</p>
②-c	M.XCHG (모듈교체 스위치)	<p>운전 중 모듈교체를 실시하는 경우 사용합니다.</p> <ul style="list-style-type: none"> <li>On (우측): 모듈교체 실시</li> </ul> <p>▶ 키스위치의 조작만으로 모듈교체가 가능</p> <ul style="list-style-type: none"> <li>Off(좌측): 모듈교체 완료</li> </ul>
③	RUN/STOP 모드 스위치	<p>CPU 모듈의 운전모드를 설정합니다.</p> <ul style="list-style-type: none"> <li>STOP → RUN: 프로그램의 연산 실행</li> <li>RUN → STOP: 프로그램의 연산 정지</li> </ul> <p>REMOTE 스위치에 우선하여 동작 합니다.</p>
④	리셋/D.Clear 스위치	<p>스위치를 좌측으로 옮기면 리셋 동작을 수행합니다.</p> <ul style="list-style-type: none"> <li>좌측이동 → 중앙복귀: RESET 동작 수행</li> <li>좌측이동 → 3 초 이상 유지 → 중앙복귀: Overall RESET 동작 수행</li> </ul> <p>스위치를 우측으로 눌렀다 놓으면 데이터 클리어 동작을 수행합니다.</p> <ul style="list-style-type: none"> <li>우측으로 누름 → 중앙복귀: 래치 1 영역 데이터와 일반 데이터영역 지움</li> <li>우측으로 누름 → 3 초 이상 유지 → 중앙복귀: 래치 2 영역 데이터와 일반 데이터영역 지움</li> </ul> <p>• 주의: 데이터 클리어 동작은 "STOP" 운전모드 만에서 동작합니다.</p>
⑤	USB 커넥터	주변기기 (XG5000 등)와 접속하기 위한 커넥터 (USB 1.1 지원)
⑥	RS-232C 커넥터	<p>주변기기와 접속하기 위한 커넥터</p> <ul style="list-style-type: none"> <li>XG5000 접속: 기본적으로 지원</li> <li>Modbus 기기 접속: Modbus 프로토콜 지원</li> </ul> <p>TX: 7 번 Pin, RX: 8 번 Pin, GND: 5 번 Pin</p>
⑦	배터리 장착 커버	백업 배터리 장착용 커버

## 2.4 XGK PLC 증설 시스템

### 입출력 증설 시스템

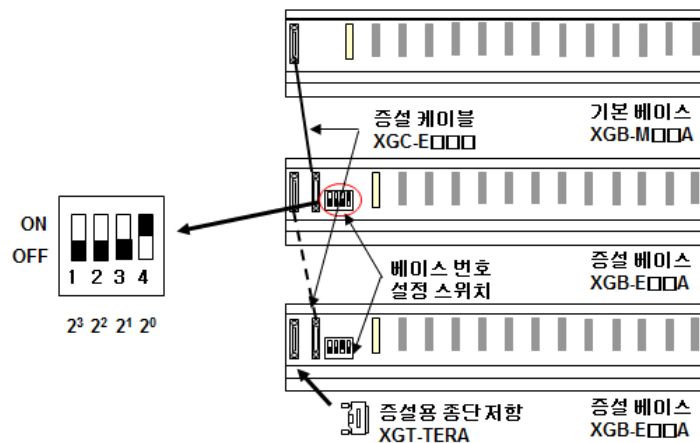
XGT PLC 에 사용하는 베이스는 기본 베이스(XGB-M □□ A)와 증설 베이스 (XGB-E □□ A) 로 구분됩니다. 1 개의 베이스는 전원 및 CPU 를 제외하고 4/6/8/12 개의 모듈을 장착할 수 있습니다. 기본 베이스는 전원 모듈이 취부되는 슬롯의 왼쪽에 증설 커넥터가 1 개 있으며, 증설 베이스는 두 개의 증설 커넥터가 있으므로 외형으로도 구분이 됩니다. 증설 베이스를 사용할 때 증설 단자 보호 커버를 열면 베이스 번호 설정용 DIP 스위치가 4 개 있는데, 이 스위치를 이용하여 베이스 번호를 설정해야 하며 최대 증설 거리(증설 케이블 길이의 합)는 15m 입니다.

증설 케이블의 연결은 기본 베이스에서 시작하여 증설 베이스의 안쪽 커넥터에 연결합니다.

2 개 이상의 증설 베이스를 설치할 경우 증설 베이스의 안쪽 커넥터는 CPU 방향으로 연결하고, 바깥쪽 커넥터는 하부 증설 베이스에 연결합니다.

증설 베이스의 증설 커넥터를 열면 베이스 번호 설정 스위치가 있습니다. 이 스위치를 이용하여 베이스 번호를 설정해 주어야 합니다.

증설 베이스를 설치할 경우 마지막 베이스의 바깥쪽 증설 커넥터에 종단 저항(XGT-TERA)를 반드시 설치해야 합니다.



### 입출력 증설용 베이스

XGT PLC 에 사용하는 기본 및 증설베이스는 베이스에 장착되는 입출력 모듈 수에 따라서 4, 6, 8, 12 슬롯의 4 종류가 있습니다. 사용자의 시스템 규모 또는 현장 패널의 공간에 따라서 선택하여 적절히 사용할 수 있습니다.

다음은 기본 및 증설베이스에 대한 형명을 표기하는 방식을 나타내고 있습니다. 사용하고자 하는 슬롯에 맞는 베이스 또는 증설용 케이블 등을 형명에 맞게 선택 사용해야 합니다.

M 은 기본베이스 즉 메인(Main)의 M 을 의미하고 E 는 증설(Extention)의 E 를 의미합니다.

#### ➤ 베이스

• XGB - □□□A

베이스 종류	장착 모듈 수
M: 기본 베이스	04: 4모듈 장착
E: 증설 베이스	06: 6모듈 장착
	08: 8모듈 장착
	12: 12 모듈 장착

증설케이블은 베이스와 베이스간을 연결하여 CPU와 입출력 모듈간의 데이터를 연결해 주는 중요한 역할을 합니다. 따라서 가급적이면 짧게 배선하여 외부의 노이즈에 영향을 적게 받도록 해야 합니다.

➤ 증설 케이블

• XGC - E □□□

길이

041: 0.4m	061: 0.6m
121: 1.2m	301: 3m
501: 5m	102: 10m
152: 15m	

2.5 XGK PLC 입출력 모듈

디지털 입력 모듈 종류

XGT PLC에 사용하는 입력 모듈의 종류는 DC24V 입력과 AC110/220V 용 입력과 특수모듈용 입력으로 나뉩니다.

입력모듈 종류에 대한 형명을 지정하는 기준은 다음과 같습니다.

• XGI - □□□□

전원 종류	접점 수(바이트)	옵션
D2: DC24V	1: 8점 (1 바이트)	A: +/- 공통(DC 입력)
A1: AC110V	2: 16점 (2 바이트)	B: + 공통(DC 입력)
A2: AC220V	4: 32점 (4바이트)	
	8: 64점 (8바이트)	

예) XGI-A12A: AC110V 16점 입력 모듈  
 XGI-D24A: DC24V 32점 입력 모듈  
 XGI-D28B: DC24V 64점 입력 모듈, + 공통

출력 모듈 종류

출력 모듈은 릴레이 출력, 트랜지스터출력, SSR 출력 등 출력 모듈에 사용되는 접점의 형태에 따라서 구분됩니다. 출력 모듈에 따라서 사용되는 전류의 용량과 사용전압이 결정 되므로 사용시 주의가 요합니다.

출력모듈 종류에 대한 형명을 지정하는 기준은 다음과 같습니다.

• XGQ - □□□□

출력 소자	접점 수(바이트)	옵션
RY: Relay	1: 8점 (1 바이트)	A: - 공통(TR 출력)
TR: Transistor	2: 16점 (2 바이트)	B: + 공통(TR 출력)
SS: SSR	4: 32점 (4바이트)	서지킬러(RY 출력)
	8: 64점 (8바이트)	

예) XGQ-RY2B: 릴레이 출력 16점, 서지 킬러 내장  
 XGQ-TR4A: Transistor 출력 32점, -공통  
 XGQ-TR8B: Transistor 출력 64점, +공통

특수 모듈 종류

특수 모듈은 디지털 입·출력 모듈과는 다르게 각 모듈마다 마이크로프로세서를 내장하여 CPU 모듈과는 별도로 각 모듈에 맞는 특수 기능을 독자적으로 수행합니다. 특수모듈의 종류로는 A/D, D/A, 고속카운터, 위치결정모듈, 통신모듈 등 여러 종류가 있습니다. 현장의 특수 상황에 따라서 특수모듈을 선택하여 사용할 수 있습니다. 단, XGK 시리즈는 별도의 PID 특수모듈 없이 CPU에 PID 기능이 내장되어 있습니다.

## 2.6 입출력 메모리 할당

PLC 프로그램 작성, 외부 입출력 결선 및 유지 보수에 있어서 PLC 외부 단자대와 PLC 메모리와 대응 관계를 정확히 이해해야 합니다. 예를 들어, 아래와 같은 경우 PLC 외부 접점과 메모리와의 정확한 대응 관계를 이해하지 못하면 프로그램의 작성 및 이해가 불가능해지기 때문입니다.

### PLC 외부 입·출력(P)

외부 입출력 번호의 할당은 첨자 (Device 이름) P로 표현하며 형식은 아래와 같습니다.



입·출력 메모리 할당: 고정식/가변식 메모리 할당 방식에 따라 입출력 메모리 할당은 달라집니다. 가변식의 경우 각 입출력 모듈의 점수에 따라 메모리 할당이 달라집니다. 예를 들어 32 점 모듈의 경우 16 점 모듈 2 개에 해당하는 접점이 한 슬롯에 설치된 것으로 보면 됩니다. 다음 그림은 가변식 할당방법에 따른 입출력 메모리 할당의 예입니다.

		슬롯번호: 0 1 2 3 4 5 6 7														
전원	CPU	P0000	P0010	P0030	P0040	P0050	P0060	P0080	P0090	~	~	~	~	~	~	~
		P000F	P002F	P003F	P004F	P005F	P007F	P008F	P009F	~	~	~	~	~	~	~
		입력	입력	출력	통신	출력	출력	A/D	XPM							

가변식 입출력 메모리 할당

- \* 입출력 메모리 할당에서 고정식으로 할당하는 경우는 각 슬롯에 64 점씩 할당이 됩니다.
- \* 가변식 메모리 할당에서는 모듈에 따라 할당되는 P 영역이 다릅니다.
  - 1) 입·출력 모듈: 실제 입출력 모듈 점점만큼 할당이 됩니다.
  - 2) 특수 모듈: 아날로그 입출력 (AD, DA), 고속카운터, 위치결정, 모션제어 모듈은 16 점이 할당됩니다.
  - 3) 통신 모듈: Cnet, Dnet, Enet (오픈형, 전용), Rnet, Pnet 은 16 점이 할당됩니다.

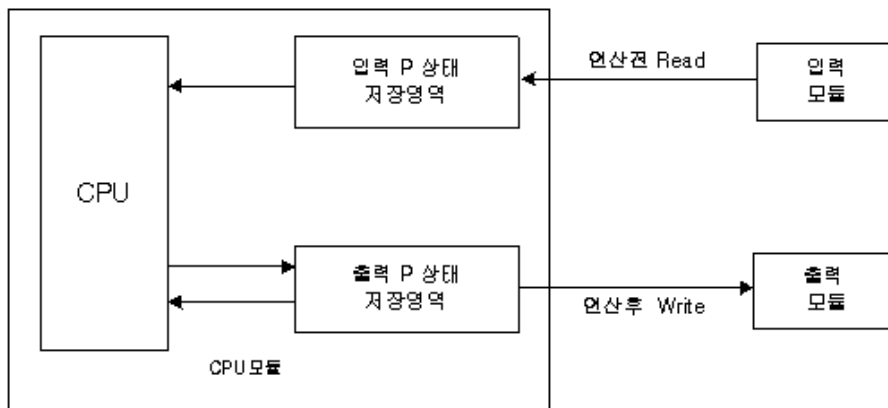
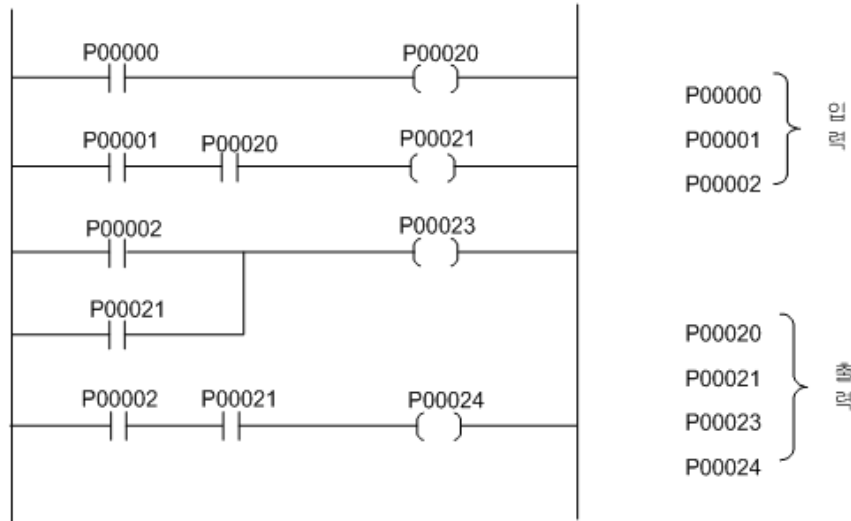
		슬롯번호: 0 1 2 3 4 5 6 7														
전원	CPU	P0000	P0040	P0080	P0120	P0160	P0200	P0240	P0280	~	~	~	~	~	~	~
		P003F	P007F	P011F	P015F	P019F	P023F	P027F	P031F	~	~	~	~	~	~	~
		입력	입력	출력	통신	출력	출력	A/D	XPM							

메모리 할당(점) 64 64 64 64 64 64 64 64

고정식 입출력 메모리 할당

**입·출력 영역 (P)**

입·출력 P 는 외부기기와 대응되는 영역으로서 입력기기로 사용되는 푸시-버튼, 전환 스위치, 리미트 스위치 등의 신호를 받아들이는 입력부와 출력기기로 사용되는 솔레노이드, 모터, 램프 등에 연산결과를 전달하는 출력부로 이루어진 영역입니다. 입력부 P 에 대해서는 PLC 내부의 메모리에 입력상태가 보존되므로 a,b 접점 사용이 가능하고 출력부 P 역시 a, b 접점의 출력 모두 가능합니다. P 영역 중 입·출력으로 사용되지 않는 부분은 보조 릴레이 M 과 동일하게 사용할 수 있습니다. 명령어에 따라 워드 단위로 사용이 가능합니다. 다음 그림은 입출력 회로 구성 예를 보여 줍니다.



위 그림과 같이 P 영역은 입·출력 모듈의 각 접점 하나 하나에 대해서 1:1 로 대응되는 영역을 가지고 있어서, PLC 가 스캔 중(연산 중)일 때는 입·출력 모듈의 접점 상태와는 관계없이 CPU 내부의 메모리(P 영역) 상태를 가지고 연산을 하고, 연산이 끝난 후 출력 접점에 대응되는 내부메모리 P 영역의 내용을 출력 모듈에 일괄 출력하고, 다음 번 연산을 위하여 입력 모듈의 접점 상태를 입력접점에 대응되는 내부 메모리 P 영역에 저장합니다. 입력, 출력의 접점 상태는 구분 없이 모두 P 영역에 할당되므로 프로그래밍 시 입력 P 영역과 출력 P 영역의 혼동에 따른 오류가 없도록 주의 바랍니다.

**I/O 슬롯 점수 할당**

한 슬롯마다 16, 32, 64 단위로 입·출력의 예약점수를 지정할 수 있으며 특수/통신 모듈의 경우에는 해당하는 특수/통신 모듈을 지정합니다. 빈 슬롯은 고정식인 경우는 64 점, 가변식인 경우는 16 점을 점유합

니다. 입출력 번호의 할당방식은 기본 파라미터의 설정에 따라서 고정식과 가변식(XGB 는 지원안함)의 선택이 가능합니다.

## 2.7 내부 메모리 할당

PLC 외부 입출력에 관계되지 않는, 즉 P 영역을 제외한 모든 메모리 영역을 내부 메모리라고 합니다. ON/OFF, Data 등이 외부 입력이나 출력에 직접적으로 의존하지 않고, 오로지 PLC 기동 시 내부에서만 연산이 이루어지는 메모리를 통칭 합니다. 특히, 접점(Bit)영역으로 사용될 때 릴레이 시퀀스의 보조 릴레이와 동작이 유사하여 보조 접점 혹은 보조 릴레이라고 합니다.

### 1) 내부 메모리의 종류

- (1) 보조 릴레이 **M**: PLC 내부 릴레이로서 외부로 직접 출력은 불가능하지만 입·출력 릴레이와 연결하면 외부 출력이 가능합니다. 프로그램 연산 중 내부 정보를 가공할 때 정보를 전달해 주는 용도로 사용됩니다. a, b 접점의 사용이 가능하며, 식별자로서 M의 기호를 사용합니다.
- (2) 정전유지 릴레이 **K (불휘발성 영역)**: 보조 릴레이와 사용 용도는 동일하나 PLC 정전 시 정전 이전의 Data를 보존하여 정전 복구 시 Data가 복구됩니다. (파라미터 설정에 관계없이 배터리 백업)
- (3) 특수 릴레이 **F**: PLC의 내부 시스템 상태, 펄스 등을 제공하는 내부 접점으로 PLC 이상 체크 및 특수한 기능을 제공합니다.
- (4) Data Register **D**: 수치 연산을 위해 내부 데이터를 저장하는 영역으로 기본 16Bit(1Word) 또는 32Bit(2Word)단위로 데이터의 쓰고 읽기가 가능합니다. 파라미터 사용에 의해 일부 영역을 불휘발성 영역으로 사용할 수 있습니다.
- (5) 타이머 **T**: 시간을 제어하는 용도로 사용되며 타이머 일치 접점과 설정시간 경과된 시간을 저장하는 별도의 영역으로 구성됩니다.
- (6) 카운터 **C**: 수를 세는 용도로 사용되며 카운터 일치 접점과 설정값 현재값을 저장하는 별도의 영역으로 구성됩니다.
- (7) 기타: 링크릴레이 **L**, 간접지정 Register: **#D**

### 2) 디바이스의 구분

디바이스는 크게 표현방법 및 오퍼랜드 처리 방법에 따라 비트 디바이스와 워드 디바이스로 나뉩니다.

< 비트 (BIT) 디바이스 >

- (1) LOAD 나 OUT 과 같은 기본 명령어에 사용할 때, .(점)없이 비트 표현이 가능한 디바이스
- (2) P, M, K, F, T(비트접점), C(비트접점), L, S
- (3) 인덱스 기능 사용시: 비트 디바이스에 인덱스 기능을 사용하면 비트의 위치에 인덱스 레지스터의 값을 더한 비트를 가리키게 됩니다. 단, 비트 디바이스를 응용명령에서 사용했고, 그 명령어의 오퍼랜드가 워드 데이터인 경우에는 워드로 계산된다.

예) LOAD P00001[Z1] → Z1=8 이면, LOAD P(1+8) = LOAD P00009  
 MOV P00001[Z1] D10 → Z1=8 이면, MOV P00009 D00010

< 워드 (WORD) 디바이스 >

- (1) 디바이스의 기본 표현이 워드 단위로 되는 디바이스입니다.
- (2) 디바이스 번호의 원하는 비트 위치를 지정하고자 할 경우 .(점)을 사용합니다.  
 예) D10 의 BIT4 의 표현은 D10.4 가 됩니다.
- (3) 해당 디바이스: D, R, U, T(현재값 영역), C(현재값 영역), Z
- (4) 인덱스 기능 사용시: 워드 단위의 인덱싱을 합니다. 또한, 워드 디바이스의 비트 표현을 한 오퍼랜드에 인덱스를 사용했을 경우 역시 워드 단위로 인덱싱을 합니다. 예를 들어, D10.4 인 오퍼랜드에 Z10 을 사용하고자 할 경우, 표기는 D10[Z10].4 와 같이 하고 의미는 D(10+Z10 의 값).4 와 같습니다.

3) 내부메모리의 구조

◆ Bit(접점) 영역(P, M, L, K, F 영역)

워드	비트															
M00	M0F	M0E	M0D	M0C	M0B	M0A	M09	M08	M07	M06	M05	M04	M03	M02	M01	M00
M01	M1F	M1E	M1D	M1C	M1B	M1A	M19	M18	M17	M16	M15	M14	M13	M12	M11	M10
.....	.....															
M09	M9F	M9E	M9D	M9C	M9B	M9A	M99	M98	M97	M96	M95	M94	M93	M92	M91	M90
M10	M10F	M10E	M10D	M10C	M10B	M10A	M109	M108	M107	M106	M105	M104	M103	M102	M101	M100
.....	.....															

◆ Word 영역

워드 메모리는 수치 데이터, 문자열 데이터 등을 저장하는 용도로 사용됩니다. XGK PLC 에서 워드 메모리는 D 영역, R 영역, U 영역, Z 영역이 있으며, 타이머(T 영역)와 카운터(C 영역)의 현재값 저장 영역도 워드 메모리에 해당합니다. 그리고, F 영역 중 시스템 데이터를 저장하는 영역은 워드 단위를 사용합니다.

워드	비트															
D00	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
D01	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
.....	.....															
D09	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
D10	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
.....	.....															

**보조 릴레이 M**

PLC 내의 내부 릴레이로서 외부로 직접 출력이 불가능하나 입·출력 P 와 연결하여 외부출력이 가능합니다. 전원 On 시와 RUN 시에 파라미터 설정에 의해 래치 영역으로 지정되지 않은 영역은 전부 0 으로 지워집니다. a, b 접점의 사용이 가능합니다.

**킵 릴레이 K**

보조 릴레이 M 과 사용 용도는 동일하나 전원 On 시나 RUN 시에는 그전의 데이터를 보존하는 래치 영역으로 기본 파라미터의 래치 영역 1 로 설정한 영역과 같은 동작을 합니다. a, b 접점의 사용이 가능합니다. 아래와 같은 3 가지의 경우에만 데이터가 0 으로 지워집니다. (래치 영역 1 의 동작 특성과 동일함.)

- (1) 프로그램을 작성하여 실행
- (2) XG5000 의 PLC 지우기 메뉴 중 메모리 지우기 기능 실행
- (3) CPU 모듈의 리셋 키조작 또는 XG5000 을 통한 Overall 리셋

**링크 릴레이 L**

통신모듈 장착시 해당 통신모듈의 정보(OS 정보(XGB 는 지원안함), 서비스 정보, 플래그 정보)를 제공하는 영역으로 통신모듈 전용 플래그 영역입니다. 래치영역 1 의 동작 특성과 동일하게 데이터를 유지합니다. 통신모듈을 사용하지 않는 경우에는 보조릴레이 M 과 동일하게 사용할 수 있습니다.

**타이머 T**

기본주기 0.1ms(XGB 는 지원안함), 1ms, 10ms, 100ms 의 4 종류가 있으며 5 종의 명령어 (TON, TOFF, TMR, TMON, TRTG)에 따라 계수 방법이 각각 다르게 됩니다.

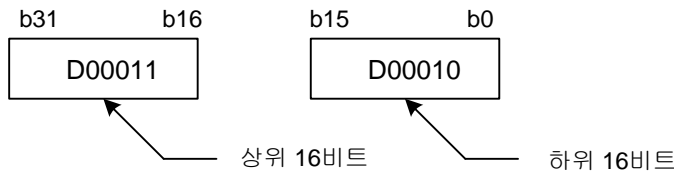
**카운터 C**

입력조건인 Rising Edge (Off→On)에서 카운트하며 Reset 입력에서 카운터의 동작을 중지하고 현재치를 0 으로 소거하거나 설정치로 대체합니다. 4 종의 명령어 (CTU, CTD, CTUD, CTR) 에 따라 각 계수 방법이 다르고 최대 설정치는 hFFFF 까지 가능합니다.

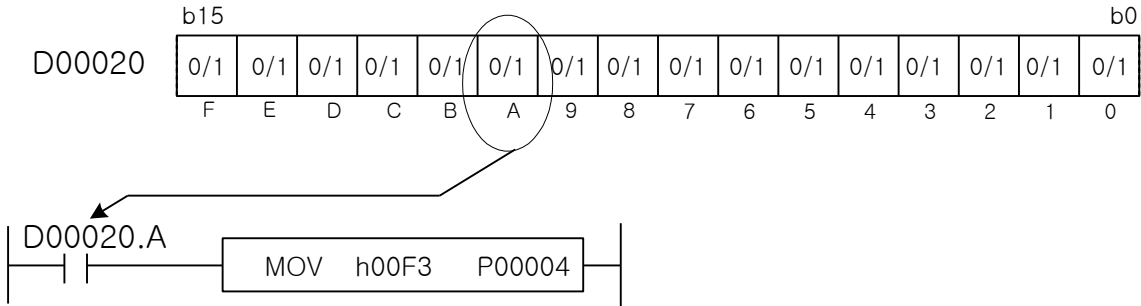
**데이터 레지스터 D**

내부 데이터를 보관하는 곳으로 16 비트, 32 비트로 읽고 쓰기가 가능할 뿐만 아니라, 비트 표현을 이용하여 한 비트씩 읽고 쓰기도 가능합니다. 32 비트 경우에는 (지정된 번호)가 하위 16 비트, (지정한 번호 + 1) 이 상위 16 비트 처리 됩니다. 데이터 레지스터의 비트 표현방법은 "지정된 번호. 지정된 비트"의 형식으로 사용합니다. 이때 지정된 비트의 표현은 16 진수로 합니다. 전원 On 시와 RUN 시작시에는 파라미터로 지정한 불휘발성 영역을 제외한 부분을 0 으로 소거하고 불휘발성 영역은 이전상태를 그대로 유지합니다. 불휘발성 영역설정은 파라미터 설정을 참조하여 주십시오.

예) 32 비트명령 사용시 D10 을 지정한 경우



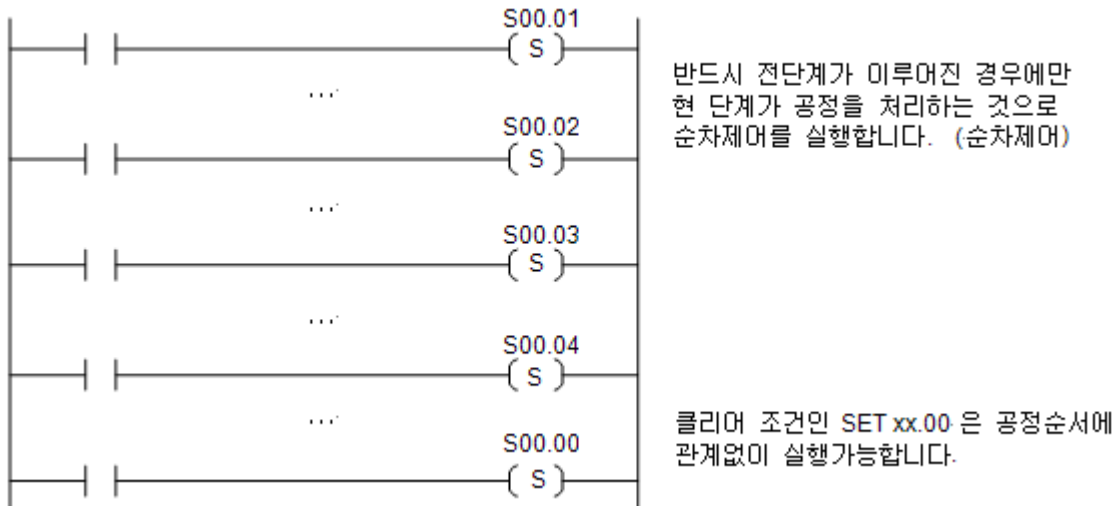
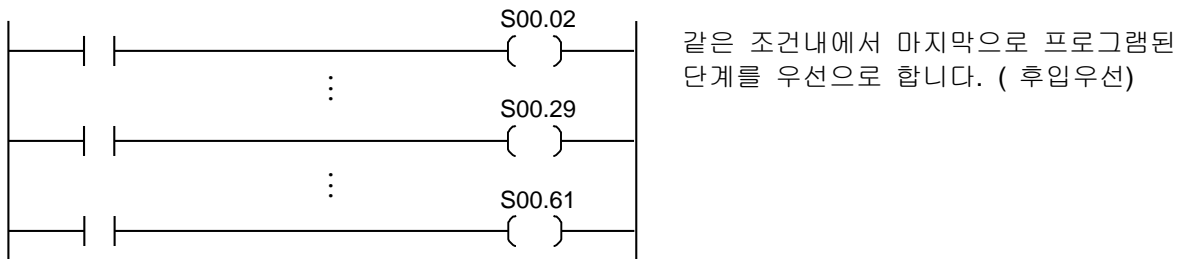
예) 데이터 레지스터 D의 비트 표현



설명: D00020의 A 번째 비트의 값에 따라 MOV 명령의 실행 여부가 결정된다.

**스텝 제어 릴레이 S**

스텝 제어용 릴레이로 명령어 (OUT, SET)사용에 따라서 후입우선, 순차제어로 구분됩니다. 전원 On 시와 RUN 시작시에 파라미터로 지정한 영역 이외는 첫 단계인 0으로 소거 됩니다.



### 특수 릴레이 F

시스템 관련 정보를 제공하는 영역으로 F0000~F1023(XGB의 경우 F200)워드까지는 읽기만 가능한 영역입니다. PLC의 현재 상태, O/S 정보, RTC 데이터, 시스템 클럭 등 PLC 운영에 필요한 전반적인 정보가 제공됩니다. F1024(XGB의 경우 F199)워드 이후 영역의 경우 전용명령을 이용하여 제한적으로 쓰기가 가능한 영역입니다. 이 영역은 외부기기 경고장 및 중고장 검출에 사용될 수 있습니다.

## 2.8 데이터 래치

### 데이터 래치 영역 설정

운전에 필요한 데이터 또는 운전 중 발생한 데이터를 PLC가 정지 후 재 기동하였을 때도 계속 유지시켜서 사용하고자 할 경우에 데이터 래치를 사용하며, 일부 데이터 디바이스의 일정 영역을 파라미터 설정에 의해서 래치 영역으로 사용 할 수 있습니다. 아래는 래치 가능 디바이스에 대한 특성표입니다.

디바이스	1 차 래치	2 차 래치	특 성
P	X	X	입출력 접점의 상태를 저장하는 이미지영역
M	O	O	내부 접점 영역
K	X	X	정전 시 접점 상태가 유지되는 접점
F	X	X	시스템 플래그 영역
T	O	O	타이머 관련 영역 ( 비트/워드 모두 해당 )
C	O	O	카운터 관련 영역 ( 비트/워드 모두 해당 )
S	O	O	스텝 제어용 릴레이
D	O	O	일반 워드 데이터 저장 영역
U	X	X	아날로그 데이터 레지스터 ( 래치 안 됨 )
L	X	X	통신 모듈의 고속링크/P2P 서비스 상태 접점(래치 됨)
N	X	X	통신 모듈의 P2P 서비스 주소 영역(래치 됨)
Z	X	X	인덱스 전용 레지스터 ( 래치 안 됨 )
R	X	X	플래시 메모리 전용 영역 ( 래치 됨 )

#### 알아두기

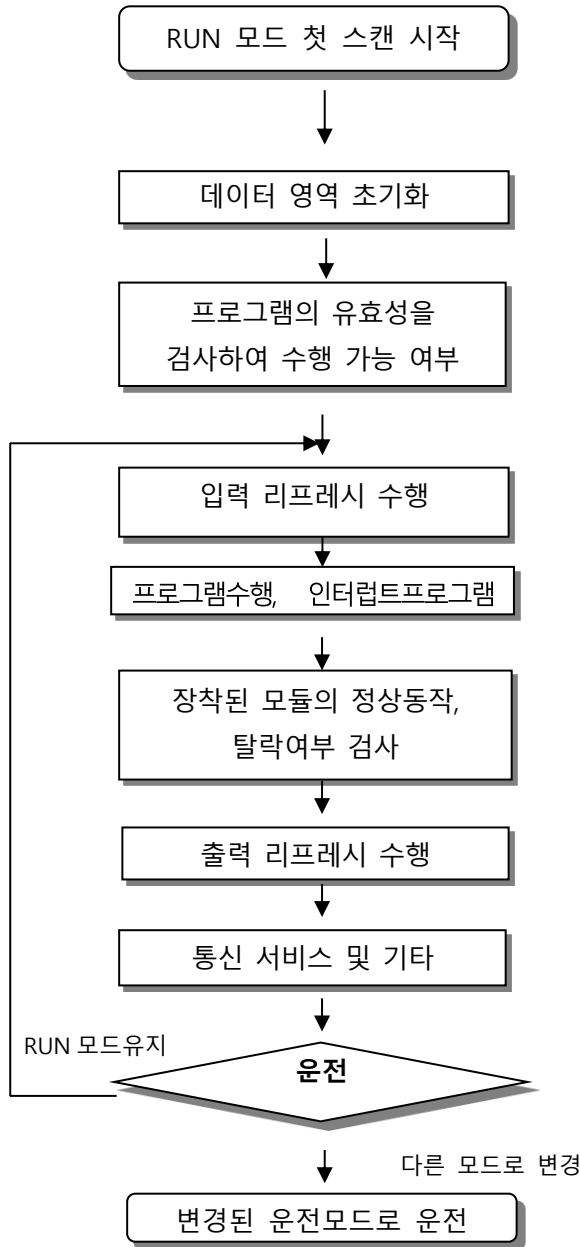
- 1) K, L, N, R 디바이스들은 기본적으로 래치 됩니다.
- 2) K, L, R 디바이스는 1 차 래치와 같이 동작합니다. 즉, Overall 리셋 또는 CPU 모듈 D.CLR 스위치 조작으로 지워집니다.
- 3) N 디바이스는 XG5000 온라인메뉴 PLC 지우기의 메모리 지우기 창에서 지울 수 있습니다.

## 2.9 운전 모드

CPU 모듈의 동작 상태에는 RUN 모드, STOP 모드, DEBUG 모드 등 3 종류가 있습니다. 각 동작 모드 시 연산 처리에 대해 설명합니다.

### RUN 모드

프로그램 연산을 정상적으로 수행하는 모드입니다.



#### (1) 모드 변경 시 처리

시작 시에 데이터 영역의 초기화가 수행되며, 프로그램의 유효성을 검사하여 수행 가능 여부를 판단합니다.

#### (2) 연산 처리 내용

입출력 리프레시와 프로그램의 연산을 수행합니다. 인터럽트 프로그램의 기동 조건을 감지하여 인터럽트 프로그램을 수행합니다. 장착된 모듈의 정상 동작, 탈락 여부를 검사합니다. 통신 서비스 및 기타 내부 처리를 합니다.

**STOP 모드**

프로그램 연산을 하지 않고 정지 상태인 모드입니다. 리모트 STOP 모드에서만 XG5000 을 통한 프로그램의 전송이 가능합니다.

- (1) 모드 변경시의 처리
  - 출력 이미지 영역을 소거하고 출력 리프레시를 수행합니다.
- (2) 연산처리 내용
  - 입출력 리프레시를 수행합니다.
  - 장착된 모듈의 정상 동작, 탈락 여부를 검사합니다.
  - 통신 서비스 및 기타 내부 처리를 합니다.

**디버그 (DEBUG) 모드**

프로그램의 오류를 찾거나, 연산 과정을 추적하기 위한 모드로 이 모드로의 전환은 STOP 모드에서만 가능합니다. 프로그램의 수행상태와 각 데이터의 내용을 확인해 보며 프로그램을 검증할 수 있는 모드입니다.

- (1) 모드 변경시의 처리
  - 모드 변경 초기에 데이터 영역을 초기화합니다.
  - 출력 이미지 영역을 소거하고, 입력 리프레시를 수행합니다.
- (2) 연산처리 내용
  - 입출력 리프레시를 수행합니다.
  - 설정 상태에 따른 디버그 운전을 합니다.
  - 프로그램의 마지막까지 디버그 운전을 한 후, 출력 리프레시를 수행합니다.
  - 장착된 모듈의 정상 동작, 탈락 여부를 검사합니다.
  - 통신 등 기타 서비스를 수행합니다.
- (3) 디버그 운전 조건
  - 디버그 운전조건은 아래 4 가지가 있고 브레이크 포인트에 도달한 경우 다른 종류의 브레이크 포인트의 설정이 가능합니다.

운전 조건	동작 설명
한 연산 단위씩 실행(스텝 오버)	운전 지령을 하면 하나의 연산 단위를 실행 후 정지합니다.
브레이크 포인트 (Break Point) 지정에 따라 실행	프로그램에 브레이크 포인트를 지정하면 지정한 포인트에서 정지합니다.
접점의 상태에 따라 실행	감시하고자 하는 접점 영역과 정지하고자 하는 상태지정(Read, Write, Value)을 하면 설정한 접점에서 지정한 동작이 발생할 때 정지합니다.
스캔 횟수의 지정에 따라 실행	운전할 스캔 횟수를 지정하면 지정한 스캔 수만큼 운전하고 정지합니다.

- (4) 조작방법
  - XG5000 에서 디버그 운전 조건을 설정한 후 운전을 실행합니다.
  - 인터럽트 프로그램은 각 인터럽트 단위로 운전 여부(Enable / Disable)를 설정할 수 있습니다.
  - (자세한 조작방법은 XG5000 사용설명서 제 12 장 디버깅을 참조하여 주십시오.)

**운전 모드 변경**

(1) 운전 모드의 변경 방법

운전 모드의 변경에는 다음과 같은 방법이 있습니다.

CPU 모듈의 모드 키에 의한 모드 변경

프로그래밍 툴(XG5000)을 CPU 의 통신 포트에 접속하여 변경

CPU 의 통신 포트에 접속된 XG5000 으로 네트워크에 연결된 다른 CPU 모듈의 운전 모드 변경

네트워크에 연결된 XG5000, HMI, 컴퓨터 링크 모듈 등을 이용하여 운전 모드 변경

프로그램 수행 중 STOP 명령에 의한 변경

(2) 운전 모드의 종류

운전 모드 설정은 다음과 같습니다.

운전모드 스위치	XG5000 지령	XGK		XGB
		리모트 허용 스위치	운전모드	운전모드
RUN	X	X	RUN	로컬 런(RUN)
STOP	RUN	On	리모트 RUN	리모트 런(RUN)
	STOP		리모트 STOP	리모트 스톱 (STOP)
	Debug		Debug RUN	디버그(Debug)
	모드 변경 수행	Off	이전 운전 모드	변경된 운전 모드
RUN -> STOP	-	X	STOP	리모트 스톱 (STOP)

리모트 모드 변환은 'XGK 는 리모트 허용: On', '모드 스위치: STOP' 인 상태에서 가능 하며, XGB 는 키 스위치가 스톱일 때 가능합니다.

리모트 'RUN' 상태에서 스위치에 의해 'STOP' 으로 변경하고자 할 경우는 스위치를 STOP → RUN → STOP 으로 조작하여 주십시오.

**알아두기**

리모트 RUN 모드에서 스위치에 의해 RUN 모드로 변경되는 경우 PLC 동작은 중단 없이 연속 운전을 합니다. 스위치에 의한 RUN 모드에서 런 중 수정은 가능합니다만 XG5000 을 통한 모드 변경 동작이 제한됩니다. 원격지에서 모드 변경을 허용하지 않을 경우에만 설정하시길 바랍니다

## 3 장. XG5000

### 3.1 XG5000 특징

XG5000은 XGT PLC 시리즈에 대해서 프로그램을 작성하고 디버깅하는 소프트웨어 툴입니다. XG5000은 다음과 같은 특징과 장점을 가지고 있습니다.

#### 1) 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.

#### 2) 다양한 드래그 & 드롭

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

#### 3) 사용자 단축키 설정

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

#### 4) 다양한 메시지 창

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

#### 5) 편리한 변수/설명 편집

5-1) 엑셀을 이용하여 편집 가능합니다.

5-2) 변수 위주 보기, 디바이스 위주 보기, 플래그 보기 등 다양한 형식으로 편집이 가능합니다.

5-3) 오토필(Auto Fill)을 이용하여 비슷한 용도의 변수를 쉽게 추가할 수 있습니다.

5-4) 드래그 & 드롭을 이용하여 다른 변수/설명 창에서 쉽게 복사할 수 있습니다.

5-5) 대화 상자를 띄우지 않고 엑셀처럼 직접 편집할 수 있습니다.

#### 6) 편리한 프로그램 편집

6-1) 제한 없는 Undo/Redo 기능을 제공합니다.

6-2) 셀 단위 블록 편집이 가능합니다.

6-3) 화면 분할 편집이 가능합니다.

6-4) 찾기/바꾸기 기능을 강화하였습니다.

6-5) 블록 마스크 기능을 이용하여 령 단위로 실행을 금지할 수 있습니다.

6-6) 북 마크 기능을 이용하여 특정 위치에 쉽게 찾아갈 수 있습니다.

6-7) LD 편집을 할 때 선택된 디바이스에 대해서 메모리 참조를 볼 수 있습니다.

7) 다양한 모니터 기능

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

8) 사용자 이벤트

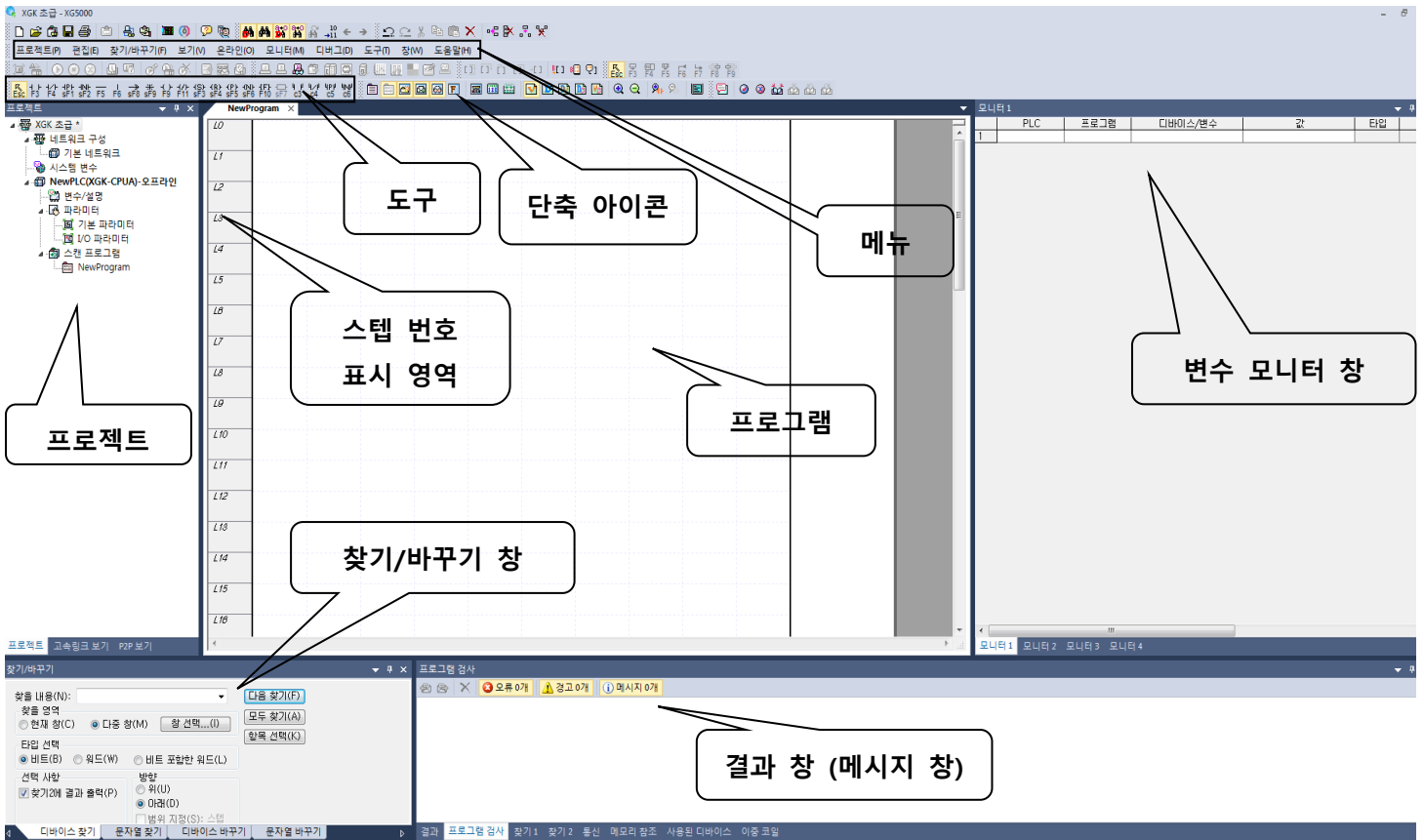
특정 디바이스에 대해서 사용자가 정의한 조건이 만족 시 Data 를 Logging 하고 열람할 수 있습니다.

9) 모듈 교환 마법사

런 중에 PLC 를 정지시키지 않고 안전하고 쉽게 모듈을 교환할 수 있습니다.

3.2 XG5000 개요

XGT PLC 를 사용하기 위해 사용자는 프로그램을 작성해야 하며, 이 때 사용하는 소프트웨어가 XG5000 입니다. 즉, XG5000 은 XGT PLC 에 대해 프로그램을 작성하고 디버깅하는 소프트웨어 입니다. 여기서는 XG5000 을 이용하여 프로젝트를 구성하고, 프로젝트 각 항목에 대해 설명합니다. XG5000 의 화면은 아래 그림과 같은 구성으로 이루어져 있습니다.



메뉴: 프로그램을 위한 기본 메뉴입니다.

도구모음: 메뉴를 간편하게 실행할 수 있습니다.

프로젝트 창: 현재 열려있는 프로젝트의 구성 요소를 나타냅니다.

메시지 창: XG5000 사용 중에 발생하는 각종 메시지가 나타납니다.

### 3.3 XG5000 기본 사용법

XG5000 은 XGT 시리즈 PLC 의 프로젝트를 작성, 전송하고, 엔지니어링 하는 소프트웨어 입니다. 여기서 는 XG5000 을 이용하여 프로젝트를 구성하고, 간단한 프로그램을 작성하여 PLC 로 전송하고 간단한 엔지니어링을 통해 XG5000 의 사용 방법을 익힙니다.

#### 1) 프로젝트의 구성

XG5000 에서 프로젝트를 구성하는 방법은 다음의 3 가지가 있습니다.

**새 프로젝트:** 새로운 프로젝트를 구성합니다.

**PLC 로부터 열기:** PLC 에 저장되어 있는 프로젝트를 PC 로 읽어(업로드) 프로젝트를 PC 에 저장하거나 내용을 수정합니다.

**KGLWIN 파일 열기:** MASTER-K PLC 용 프로젝트 파일을 XGK/XBM/XBC PLC 용 파일로 변환합니다.

**GMWIN 파일 열기:** GLOFA-GM PLC 용 프로젝트 파일을 XGR/XGI/XEC PLC 용 파일로 변환합니다.

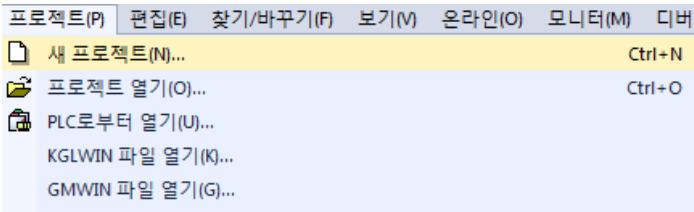
KGLWIN/GMWIN 파일 열기를 할 때 KGLWIN/GMWIN 파일을 선택하고 적용하고자 하는 XGT 시리즈의 PLC 를 선택하면 XG5000 프로젝트로 변환이 됩니다. 이 때, KGLWIN/GMWIN 파일에 편집되어 있는 변수 /설명문도 XG5000 프로젝트로 변환됩니다.

KGLWIN 파일의 경우 주소가 변경된 특수 디바이스까지 자동 변경됩니다. 예를 들면 MASTER-K 에서 상시 ON 플래그가 F0010 이지만 XGK PLC 에서는 F00099 로 변경되었습니다.

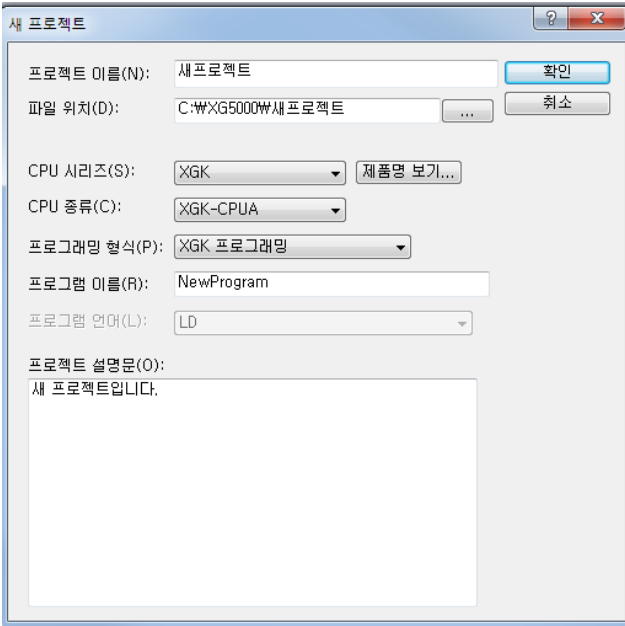
그러나, KGLWIN/GMWIN 파일에서 사용된 명령어 중 XGT 시리즈에서 지원하지 않는 명령어는 XG5000 프로젝트에 Err.로 표시 되므로, 변환 후 프로그램의 수정이 필요할 수 있습니다. 특히, MASTER-K/GLOFA-GM PLC 에서 특수/통신 모듈 관련 명령어는 XGT 시리즈 PLC 에서는 모두 파라미터로 처리되므로 KGLWIN/GMWIN 파일의 특수/통신 모듈 관련 명령어를 확인하여 파라미터로 설정해야 합니다.

MASTER-K 에서 특수 모듈의 운전 파라미터를 설정하는 PUT(P) 명령어는 XGT 시리즈에서도 지원하지만, 특수 모듈의 메모리 주소 및 데이터가 변경된 것이 있으므로 변환된 XG5000 파일의 PUT(P) 명령어를 삭제하고 특수 모듈 파라미터로 변경하는 것이 좋습니다.

여기서는 새 프로젝트를 구성하는 방법에 대하여 설명합니다.



프로젝트 메뉴의 '새 프로젝트'를 선택하거나 '새 프로젝트' 아이콘 (📁)을 눌러 '새 프로젝트 구성 대화 상자'를 엽니다.



새 프로젝트 창에서 각 항목을 입력합니다.

프로젝트 이름: 프로젝트 이름을 입력합니다.

파일 위치: XG5000 은 파일 위치로 지정된 폴더 하부에 프로젝트 이름과 같은 폴더를 만들고 그 폴더에 프로젝트 파일을 저장합니다.

파일 위치 폴더는 XG5000 의 도구 메뉴 옵션 항목에서 변경할 수 있습니다.

PLC 시리즈 및 CPU 종류: PLC 기종을 선택한 후 CPU 종류를 선택합니다.

XGK 시리즈와 XGB 시리즈 중 XBM/XBC 는 서로 기종 변환이 가능하며, XGI/XGR 시리즈 와 XGB 시리즈 중 XEC 는 서로 기종 변환이 가능합니다.

프로그램 이름: XGT PLC 는 한 대의 PLC 가 연산할 프로그램을 여러 개로 나누어서 작성할 수 있습니다. 그 중 첫 번째 프로그램의 이름을 입력합니다.

프로젝트 설명문: 프로젝트에 대한 설명문을 입력합니다.

\* 프로그래밍 형식: 기존 XGK 프로그래밍 형식과 자동할당 형식을 선택할 수 있습니다. 자동할당 형식은 XGI 처럼 데이터 메모리 중 일부 영역을 자동할당 메모리 영역으로 사용하는 기능입니다. XGI 처럼 사

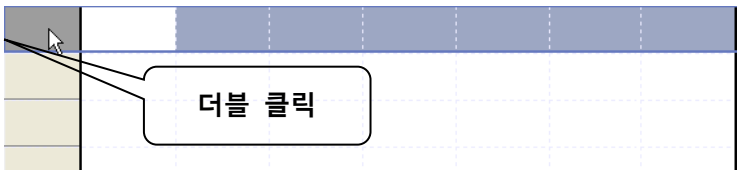
용자 평선/평선 블록을 사용할 수 있고, ST 나 SFC 로 프로그램을 구성할 수 있습니다. 자동할당 할 메모리 영역은 '메모리 자동할당' 파라미터에서 설정할 수 있습니다.

2) 프로그램 작성

Ex 3-1) P00000 입력 접점에 연결된 스위치가 ON 되면 P00020 출력 접점에 연결된 LED 를 점등 되고 스위치가 OFF 되면 LED 가 소등됩니다.

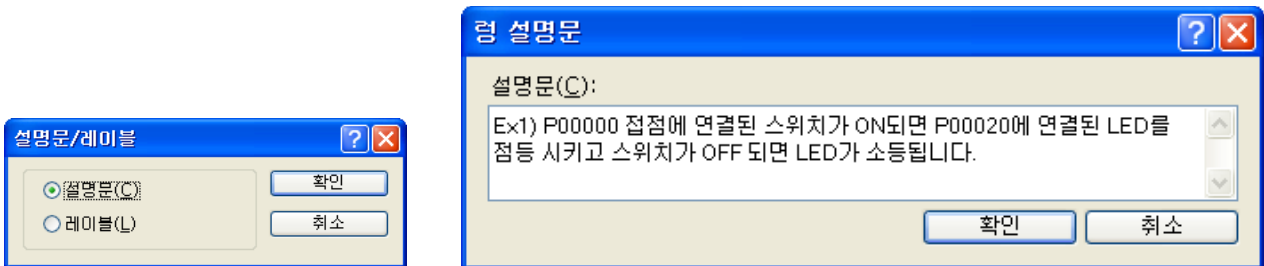
링 설명문 편집: 부분적인 프로그램에 간단한 설명문을 편집합니다.

- 링 설명문을 편집 하고자 하는 행의 스텝 번호 표시 영역을 더블 클릭 합니다.

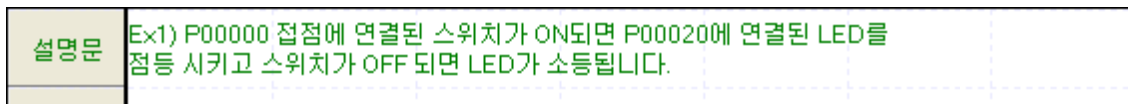


- 설명문/레이블 선택 대화 상자에서 설명문을 선택하고 '확인' 버튼을 클릭합니다.

링 설명문 대화 상자의 설명문 부분에 프로그램에 관한 간단한 설명을 편집한 후 '확인' 버튼을 클릭 합니다.

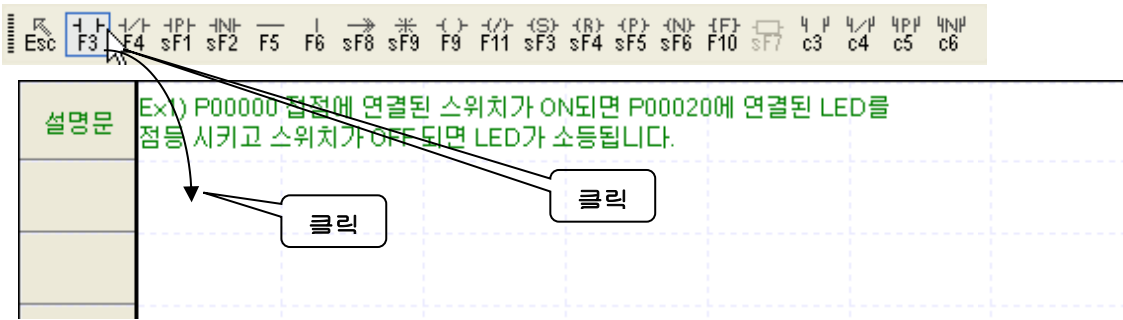


- 프로그램 창에 링 설명문이 입력되었습니다.

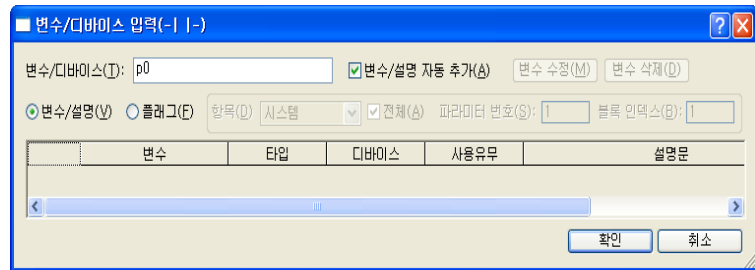
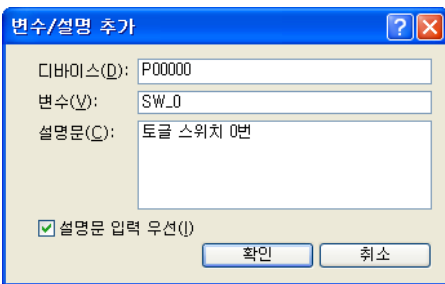


접점 편집: 스위치의 ON/OFF 상태를 읽어오기 위해 접점을 편집합니다.

- 도구 상자에서 평상시 열린 접점을 클릭한 후 마우스 커서를 프로그램 창으로 이동하면 커서 하단에 현재 선택된 프로그래밍 기호가 표시 됩니다.



- 프로그램 창에서 다시 한 번 마우스를 클릭하면 변수/디바이스 입력 대화 상자가 나타납니다. '변수/디바이스' 항목에 P0 를 입력하고, '확인' 버튼을 클릭합니다. 이 때, '변수/설명 자동 추가'가 선택(체크)되어 있으면 변수/설명 추가 대화 상자가 나타납니다.

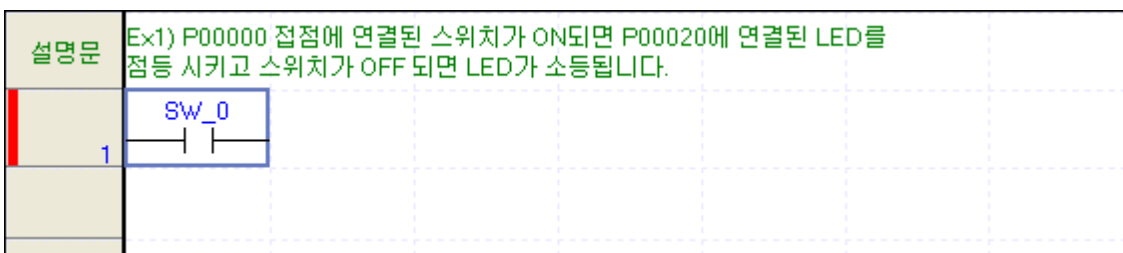


\* 변수/설명 추가 대화 상자에서 '설명문 입력 우선'이 선택되어 있을 경우 커서가 설명문 입력 창에서 활성화 되며, 선택되어 있지 않을 경우 변수 입력 창에서 활성화 됩니다.

\* 변수 등록 후 해당 데이터를 호출할 때 디바이스 또는 변수 이름을 이용하여 호출할 수 있으며, 1 개의 변수에는 1 개의 디바이스가 할당 되어야 합니다.

\* 변수에 사용할 수 있는 기호는 문자, 숫자, 특수 문자(\_)로 제한이 되며, 변수의 선두에는 숫자를 사용할 수 없습니다.

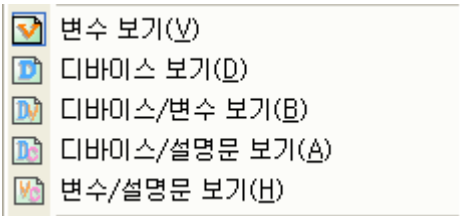
- 프로그램 창에 접점이 등록되었습니다.



- 좌측의 적색 세로선은 프로그램에 오류가 있다는 의미입니다.

- 좌측의 숫자 '1'은 링의 시작 스텝 번호입니다.

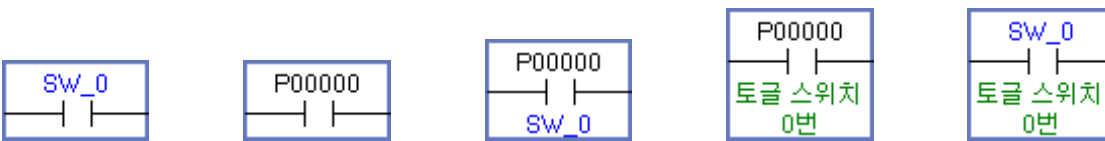
- XG5000 에는 몇 가지의 보기 옵션을 선택할 수 있습니다. 보기 옵션에 따라 변수의 표시 형태가 달라 집니다. 보기 옵션은 보기 메뉴에서 선택할 수 있으며, XG5000 의 단축 아이콘 창에 단축아이콘으로 표 시됩니다.



보기 메뉴의 보기 옵션

단축 아이콘 창의 보기 옵션 단축 아이콘

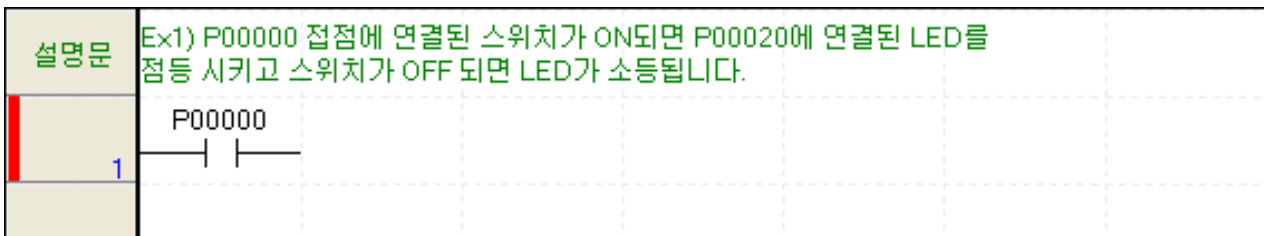
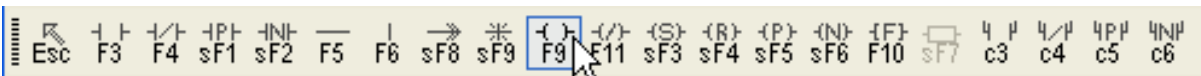
- 변수 보기: 프로그램 작성 시 편집한 변수를 표시합니다. 변수를 편집하지 않았을 경우 디바이스 보기로 표시됩니다.
- 디바이스 보기: 프로그램 작성 시 사용한 디바이스(PLC 메모리 주소)를 표시합니다.
- 디바이스/변수 보기: 디바이스와 변수를 동시에 표시합니다. 변수를 편집하지 않았을 경우 디바이스만 표시됩니다.
- 디바이스/설명문 보기: 디바이스와 설명문을 동시에 표시합니다. 설명문이 편집되지 않았을 경우 디바이스만 표시됩니다.
- 변수/설명문 보기: 변수와 설명문을 표시합니다. 변수가 편집되지 않았을 경우 변수는 디바이스로 표시되며, 설명문이 편집되지 않았을 경우 설명문은 표시되지 않습니다.



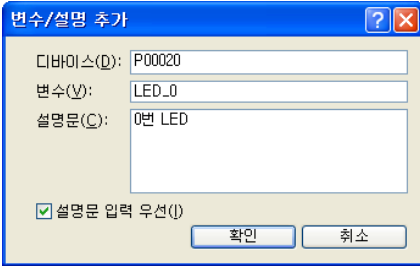
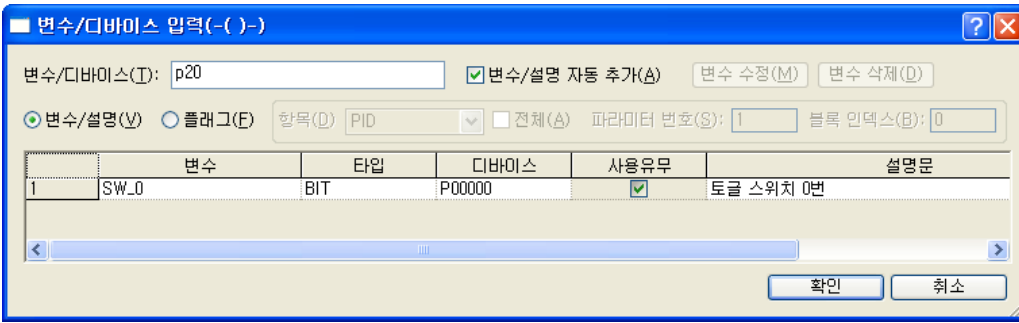
변수 보기    디바이스 보기    디바이스/변수 보기    디바이스/설명문 보기    변수/설명문 보기

코일 편집: 연산 결과를 출력하기 위해 코일을 편집합니다.

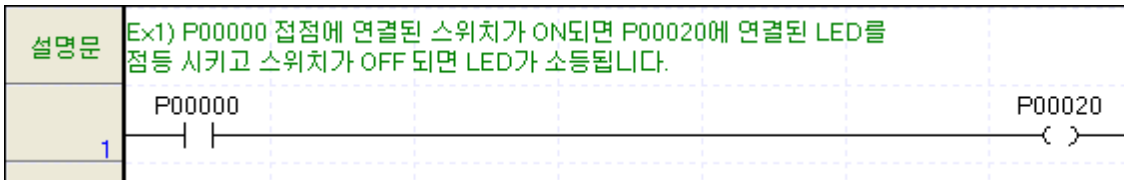
- 도구 상자에서 코일을 선택한 후 프로그램 창에서 편집된 접점의 다음 셀에서 다시 한 번 클릭합니다.



- 변수/디바이스 창에 P20 을 입력한 후 '확인' 버튼을 누릅니다. 이 때 변수/설명 자동 추가가 선택(체크)되어 있으면 변수/설명 추가 창이 나타납니다. 변수/설명 추가 창에서 아래의 그림과 같이 편집한 후 '확인' 버튼을 클릭합니다.



- 변수/디바이스 창에 P20 을 입력한 후 '확인' 버튼을 누릅니다. 이 때 변수/설명 자동 추가가 선택(체크) 되어 있으면 변수/설명 추가 창이 나타납니다. 변수/설명 추가 창에서 아래의 그림과 같이 편집합니다.

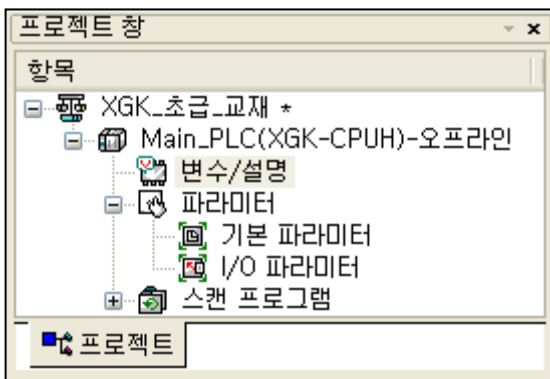


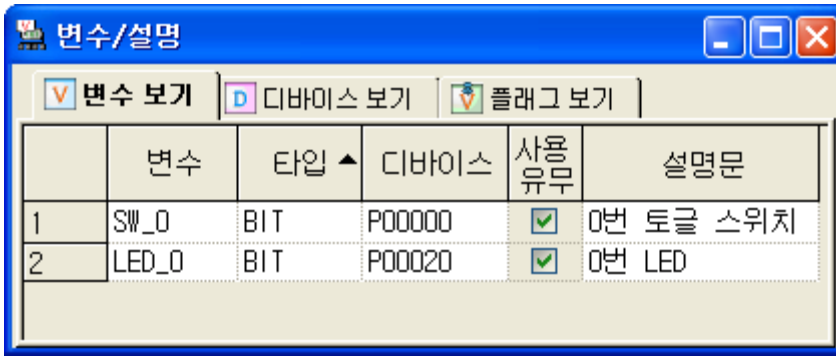
### 3) 변수/설명문

디바이스에 대한 변수 및 설명문은 앞에서 설명한 바와 같이 프로그램에서 디바이스를 최초로 호출할 때 편집할 수도 있지만, 프로그램에서 디바이스를 호출하기 전에 먼저 등록할 수도 있습니다.

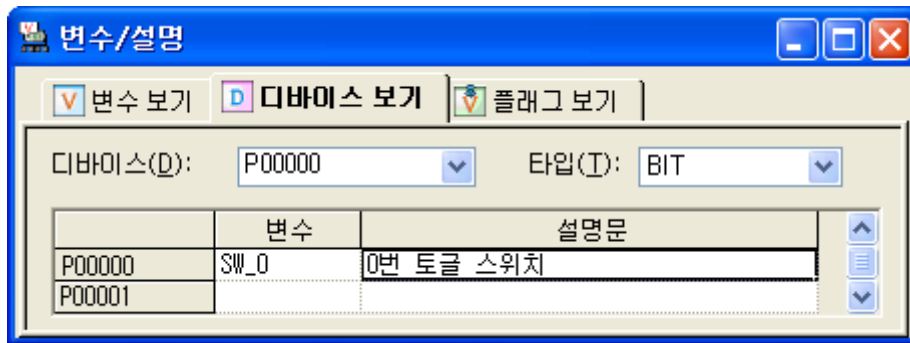
XG5000 의 기능을 이용한 일괄 변수/설명 편집: XG5000 의 변수/설명 창에서 디바이스에 대한 변수/설명을 편집할 수 있습니다. XG5000 변수/설명 창에 복사, 자동 증가 기능을 사용하면 좀 더 편리하게 변수 또는 설명을 편집할 수 있습니다.

XG5000 프로젝트 창에서 변수/설명을 더블 클릭하여 변수/설명 창을 활성화 시킵니다. 변수 보기 탭을 선택하면 현재 등록되어 있는 변수 또는 설명문이 표시됩니다.

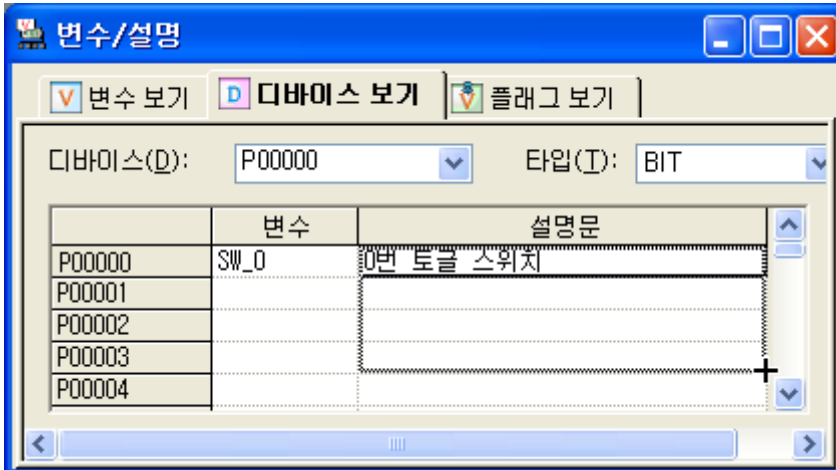




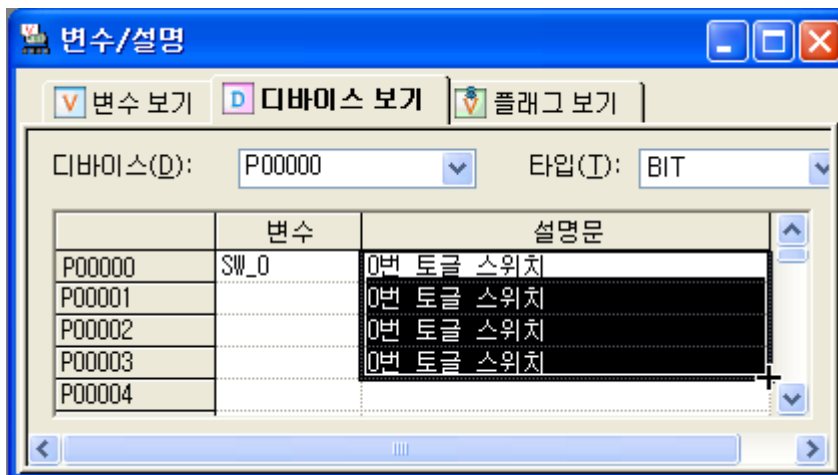
디바이스 보기 탭을 선택한 후 디바이스 항목에 표시할 디바이스의 선두 주소를 입력하고 타입에 데이터 타입 (BIT 또는 WORD)을 선택 하면 디바이스 항목에 지정된 주소부터 타입에 지정된 크기 주소가 순서대로 표시됩니다. 디바이스에 대해 변수 또는 설명문을 셀을 선택한 후 변수 또는 설명문을 편집하면 됩니다.



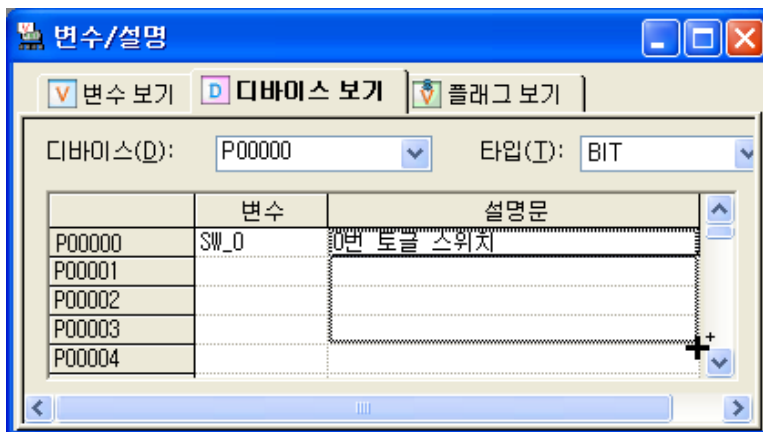
디바이스 설명문 복사: 연속된 주소에 비슷한 내용의 설명문을 편집하고자 할 때 1 개 디바이스의 설명문을 편집한 후 마우스를 아래 또는 위로 드래그 하면 선택된 설명문이 복사됩니다. 이렇게 복사 후 수정할 내용을 수정할 수 있습니다.



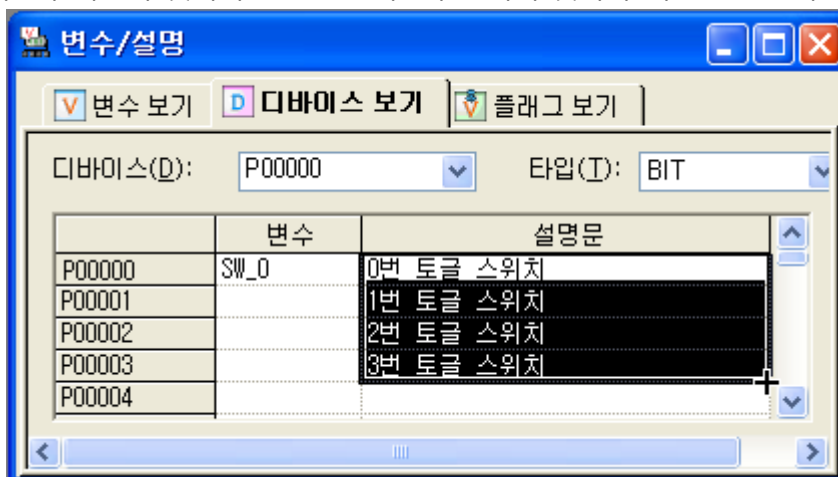
각 디바이스에 동일한 설명문이 복사되었습니다.



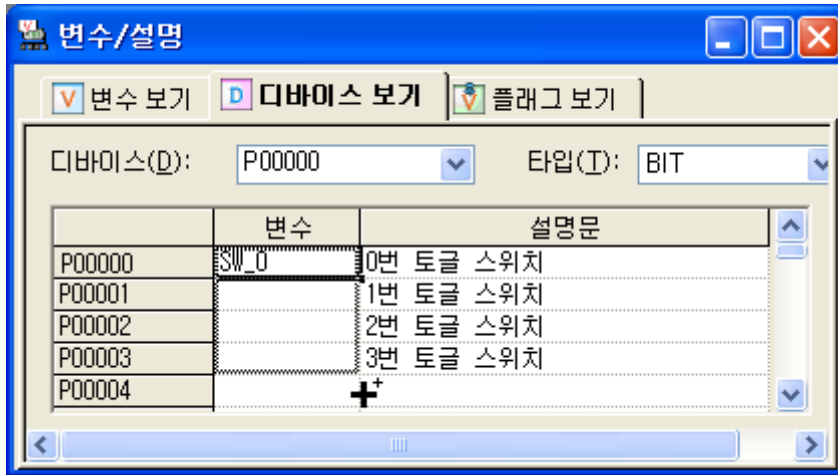
디바이스 설명문에 포함된 숫자 자동 증가: 숫자가 포함된 디바이스 설명문을 선택하고 Ctrl 키를 누른 상태에서 마우스를 아래 또는 위로 드래그 하면 드래그 되는 위치에 표시되는 십자 커서 우측 상단에 작은 십자 커서가 나타납니다.



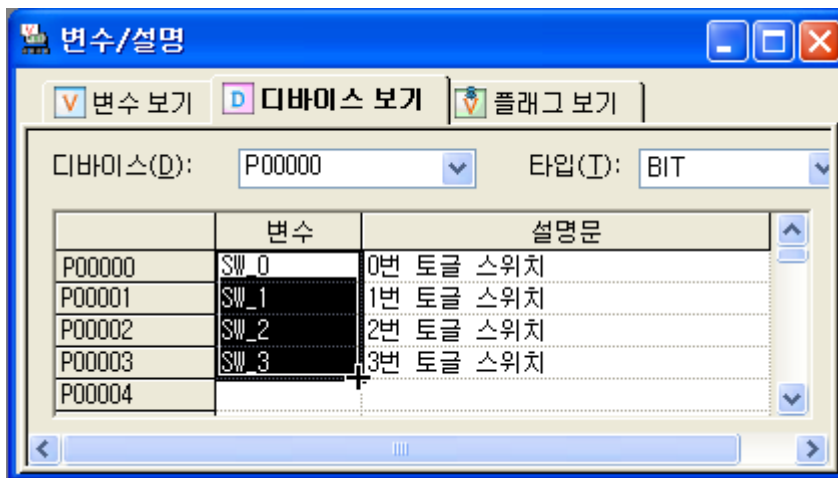
최초 선택된 디바이스의 설명문에 포함된 숫자가 자동으로 증가했습니다. 만일, 최초 선택된 디바이스의 설명문에 두 개 이상의 숫자가 포함된 경우 제일 뒤의 숫자가 자동으로 증가됩니다.



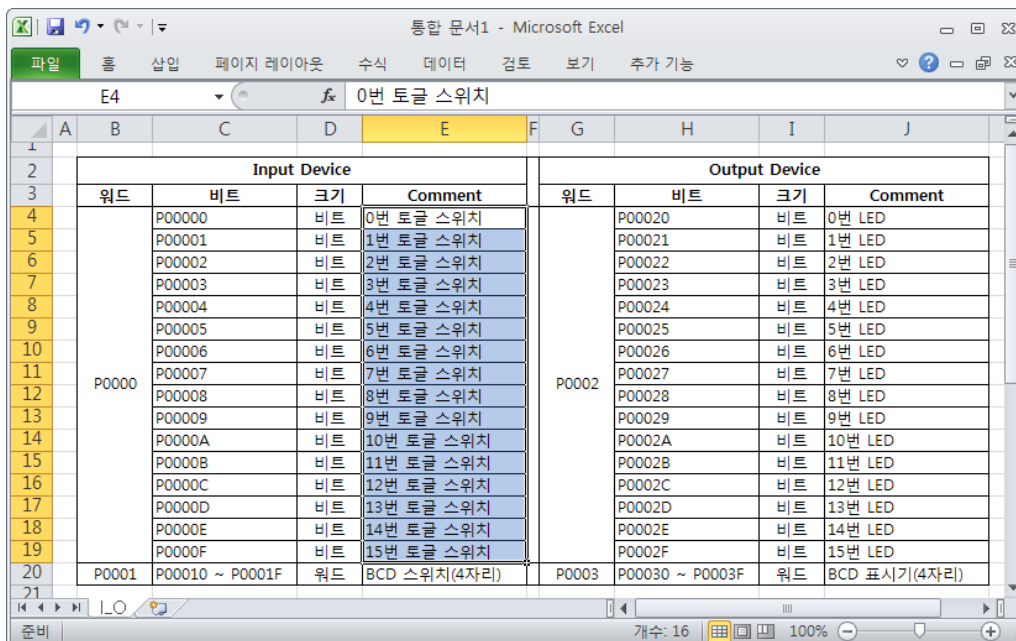
변수에 포함된 숫자 자동 증가: 숫자가 포함된 변수를 선택하여 마우스를 아래 또는 위로 드래그 하면 드래그 되는 위치에 표시되는 십자 커서 우측 상단에 작은 십자 커서가 나타납니다.



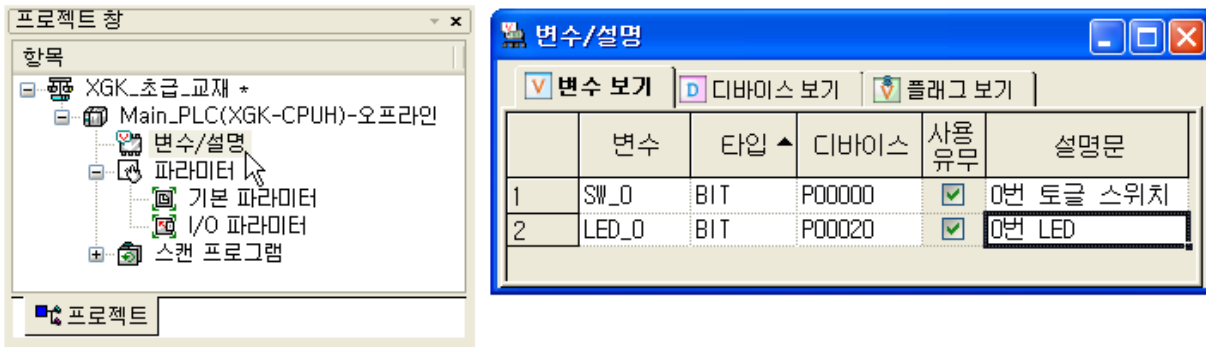
최초 선택된 변수에 포함된 숫자가 자동으로 증가했습니다. 만일, 최초 선택된 디바이스의 설명문에 두 개 이상의 숫자가 포함된 경우 제일 뒤의 숫자가 자동으로 증가됩니다.



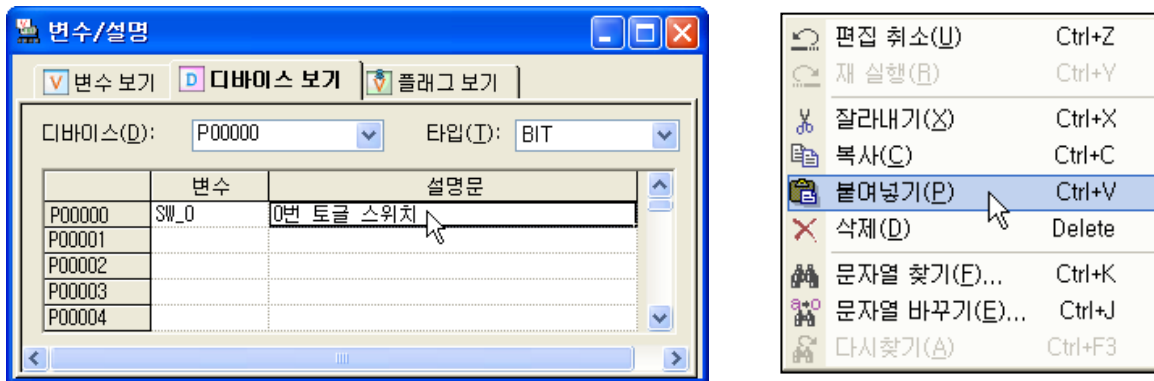
엑셀 등을 이용하여 디바이스의 사용 용도를 정리해 놓은 자료가 있을 경우 별도의 편집 없이 자료의 내용을 복사하여 XG5000 에 붙여넣기를 함으로써 간단히 변수 또는 설명문을 편집할 수 있습니다. 디바이스의 사용 용도를 정리한 파일을 열어 복사하고자 하는 부분을 선택하여 복사합니다.



XG5000 프로젝트 창에서 변수/설명을 더블 클릭하여 변수/설명 창을 활성화 시킵니다. 변수 보기 탭이 선택되면 현재 변수 또는 설명문이 등록되어 있는 디바이스가 표시됩니다.



디바이스 보기를 선택한 후 디바이스에 시작 주소를 입력하고, 타입을 선택합니다. 변수 또는 설명문을 마우스로 선택한 후 마우스 오른쪽 버튼을 클릭해서 나타나는 팝-업 메뉴에서 붙여넣기(Ctrl + V)를 선택합니다.



파일에서 복사된 내용이 XG5000의 설명문으로 등록되었습니다.

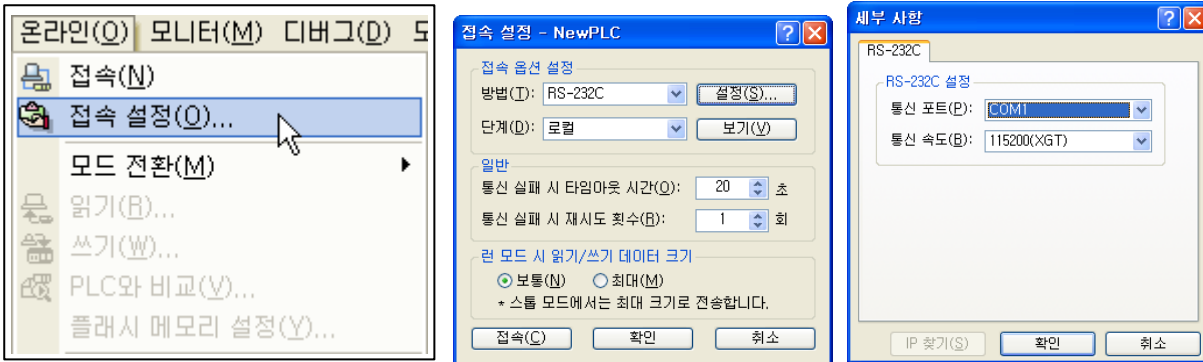


- \* 위의 과정과 반대로 XG5000의 변수/설명문을 복사하여 엑셀 등에 붙여넣기를 실행할 수 있습니다.
- \* XG5000 프로젝트 메뉴에서 변수/설명 파일로 저장을 선택하면 변수/설명문 파일을 'PLC 이름.csv' 파일로 저장할 수 있습니다.

4) 접속

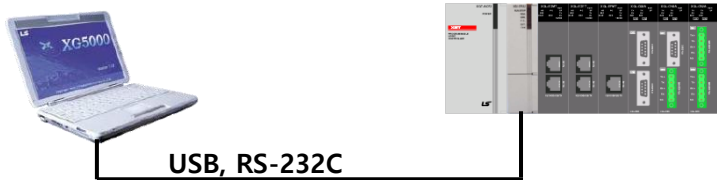
프로그램의 작성이 완료 되었으면 작성된 프로그램 및 파라미터를 PLC 로 전송하고 PLC 를 RUN 상태로 전환해 주어야 PLC 는 동작합니다. PC 에서 작성한 프로젝트를 PLC 로 전송하기 위해서 PC 와 PLC 간 접속이 이루어 져야 하며, XGT 는 다음과 같은 방법으로 PC 와 접속할 수 있습니다.

접속 설정: XG5000 의 '온라인' 메뉴에서 '접속 설정'을 선택하면 접속 방법 및 단계를 선택하고, 선택된 통신 방법 및 단계에 따른 통신 세부 사항을 설정하는 단계로 구성됩니다. XG5000 에서 접속 단계에 따라 5 가지 접속 방법을 제공합니다.

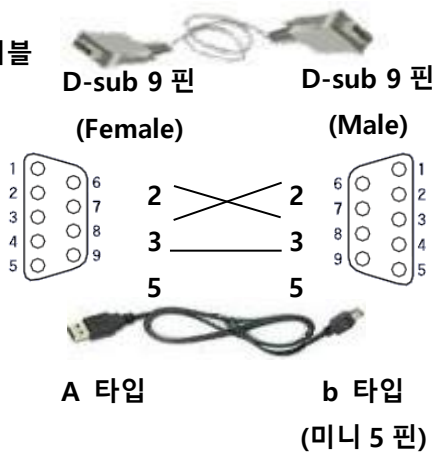


온라인' 메뉴 >> 접속 설정 선택      접속 방법 및 단계 설정      통신 세부 사항 설정

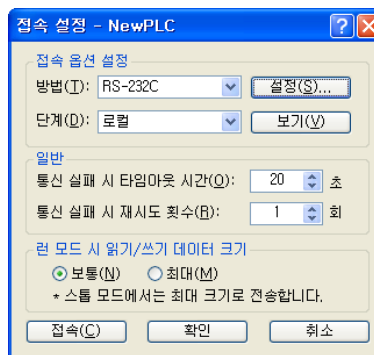
로컬 접속: PC 와 PLC CPU 간 접속하는 방법입니다. XGT CPU 에 USB 포트 및 RS-232C 포트 및 이더넷 포트가 있으므로 로컬 접속은 USB 및 RS-232C 통신을 이용할 수 있습니다. 로컬 접속 시 통신 방법으로 RS-232C 나 이더넷을 선택했을 경우 통신 세부 사항을 설정해야 하며, USB 를 선택했을 경우 통신 세부 사항을 설정하지 않습니다.



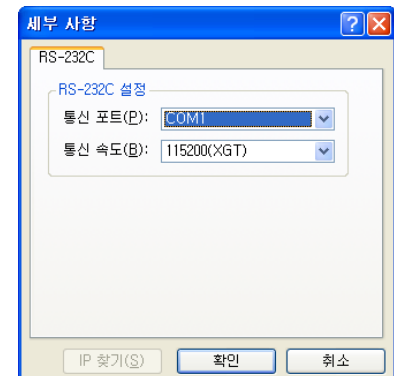
RS-232C 케이블



USB 케이블



통신 방법 및 단계 설정



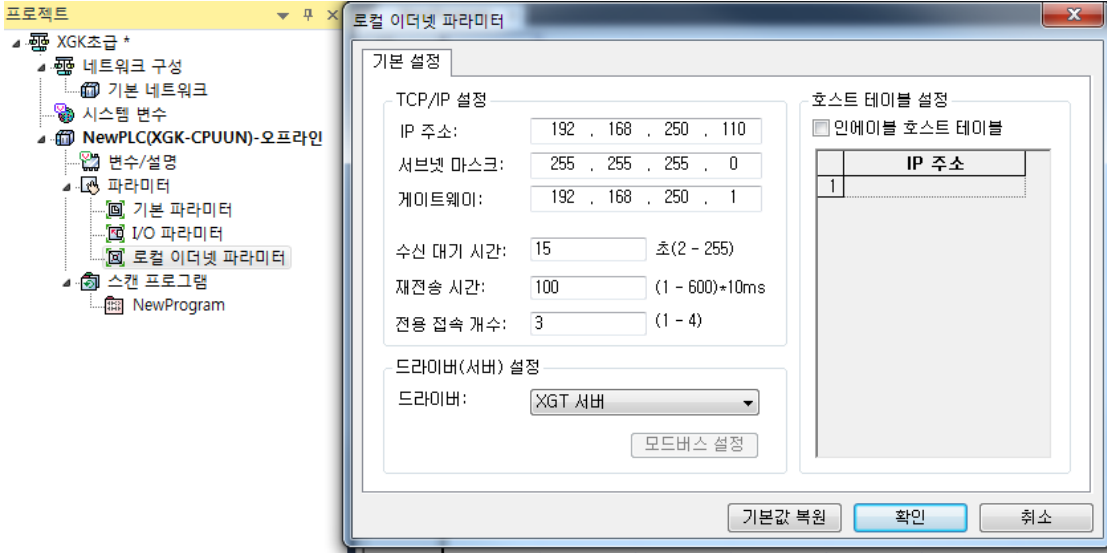
통신 세부 사항 설정

RS-232C 케이블 형명: K1C-050A, USB 케이블 형명: USB-301A

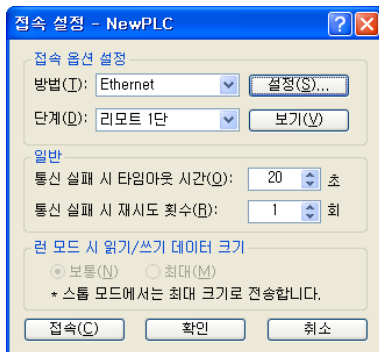
\* PC 측에 RS-232C 포트가 없고 USB 포트만 있는데 RS-232C 를 사용하고자 할 경우, USB to Serial 컨버터를 이용할 수 있으며, 컨버터의 포트 번호는 Windows 제어판에서 확인 및 변경할 수 있습니다.

\* XGT 의 USB 통신 드라이버는 XG5000 Setup 시 자동으로 인스톨되며, 사용자 선택에 의해 인스톨 하지 않을 경우 드라이버 파일은 W...W\XG5000\Drivers 폴더에 복사됩니다.

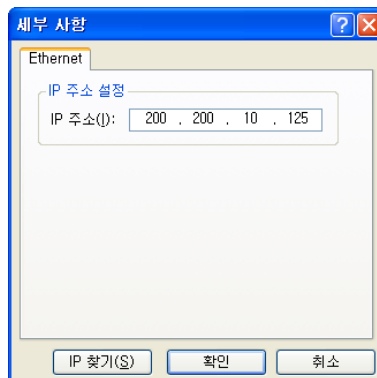
\* XGK-CPUUN, CPUHN, CPUUN 은 이더넷이 내장되어 있고 로컬 이더넷 파라미터에서 초기 IP 주소를 확인할 수 있습니다. 이 IP 주소로 로컬 이더넷 접속이 가능합니다.



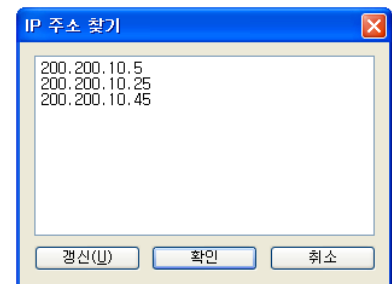
**리모트 1 단 접속:** PC 와 PLC 에 장착된 통신 모듈을 이용하여 CPU 에 접속하는 방법입니다. XGT PLC 에 Ethernet 계열의 통신 모듈 또는 Cnet 모듈 중 RS-232C 통신 모듈이 장착되어 있을 때 이용할 수 있습니다.



통신 방법 및 단계 설정



통신 세부 사항 설정



IP 찾기

FNet 또는 EtherNet/IP 통신을 이용하여 리모트 1 단 접속하고자 하는 경우 접속 방법으로 Ethernet 을 선택해야 하며, 통신 세부 사항 설정에서 IP 주소는 접속하고자 하는 PLC 에 장착된 FNet 또는 EtherNet/IP 모듈에 설정된 IP 주소를 입력합니다.

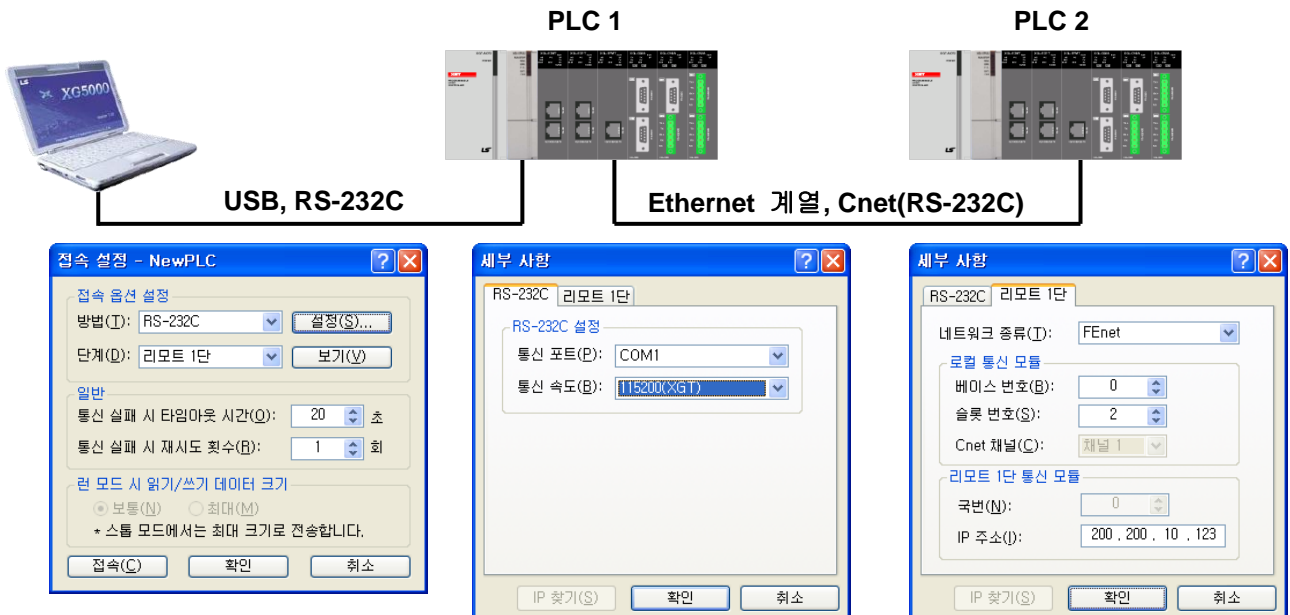
이 화면에서 'IP 찾기' 버튼을 선택하면 PC 가 포함된 Ethernet 네트워크에 연결된 모든 XGT PLC 의 Ethernet 모듈에 설정된 IP 주소가 표시됩니다. 접속하고자 하는 PLC 에 장착된 Ethernet 모듈의 IP 를 선택하고 확인 버튼을 클릭하면 선택된 PLC 를 접속할 수 있습니다.

또, RS-232C 를 이용한 리모트 접속의 경우 로컬 접속으로 선택 후 통신 세부 사항 설정 화면에서 PC 측의 통신 포트 번호 및 XGT Cnet 에 설정된 통신 속도를 이용하여 접속합니다. (115200/38400 bps 중 선택)

\* XGT Ethernet 모듈은 IP 주소가 설정되지 않은 상태로 출하됩니다.

\* Ethernet 을 이용한 리모트 1 단 접속 시 PC 의 IP 주소를 PLC Ethernet 통신 모듈의 IP 주소와 동일한 대역으로 설정해야 합니다.

**로컬 & 리모트 1 단 접속:** 로컬로 접속된 PLC 와 통신하고 있는 다른 PLC 를 접속할 경우 사용하는 방법입니다.



통신 방법 및 단계 설정

RS-232C 통신 세부 사항 설정

리모트 1 단 통신 세부 사항 설정

접속 방법을 RS-232C 또는 USB, 단계를 리모트 1 단으로 설정했을 때 세부 사항 설정 화면에는 로컬 과 리모트 1 단의 2 개의 탭이 생성 됩니다. 로컬 탭에서는 로컬 접속되는 RS-232C 통신을 설정하고, 리모트 1 단에서는 리모트 통신을 설정합니다.

리모트 접속에 사용될 수 있는 네트워크 종류로는 Cnet, FEnet, FDEnet, RAPIenet, EtherNet/IP 등이 있으며, 로컬 통신 모듈 항목에서 로컬로 접속되는 PLC(PLC 1)에서 리모트 접속에 사용될 통신 모듈이 장착된 베이스 번호 및 슬롯 번호를 설정합니다.

리모트 1 단 통신 모듈에서 접속하고자 하는 PLC(PLC 2)에 장착된 통신 모듈의 통신 파라미터를 설정 합니다. RAPIenet 또는 Cnet 통신 모듈이 사용될 경우 국번을, FEnet, FDEnet, EtherNet/IP 통신 모듈이 사용될 경우 IP 주소를 입력합니다.\* 리모트 1 단 접속 통신에 사용할 통신 모듈로 FEnet, FDEnet,

EtherNet/IP 를 사용할 경우 PC 가 직접 Ethernet 통신을 하는 것이 아니므로 PC 의 IP 는 설정할 필요가 없습니다.

**리모트 2 단 접속:** 리모트 1 단으로 접속된 PLC(PLC 1)가 리모트 1 단 접속하고 있는 통신과 다른 종류의 통신 모듈을 이용하여 다른 PLC(PLC 2)와 통신하고 있을 때, PLC1 과 PLC2 간 연결되어 있는 통신을 이용하여 접속하는 방법입니다.

The diagram illustrates two connection methods for PLCs:

- Left Method:** FNet, EtherNet/IP, Cnet(RS-232C)
- Right Method:** Ethernet 계열, Cnet(RS-232C)

Below the diagrams are three software configuration windows:

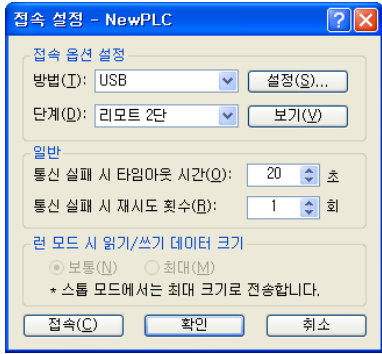
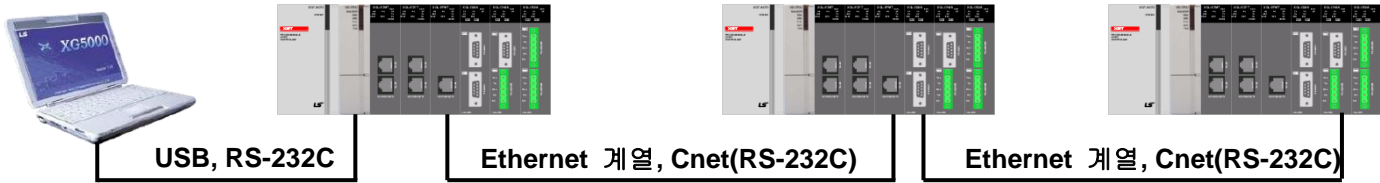
- 접속 설정 - NewPLC:** Shows connection method (Ethernet) and stage (리모트 2단).
- 세부 사항 (Remote 1):** Shows IP 주소 설정 (200 . 200 . 10 . 125).
- 세부 사항 (Remote 2):** Shows network type (FNet), remote 1 module (베이스 번호: 0, 슬롯 번호: 2), and remote 2 module (국번: 0, IP 주소: 200 . 200 . 20 . 123).

통신 방법 및 단계 설정      리모트 1 단 통신 세부 사항 설정      리모트 2 단 통신 세부 사항 설정

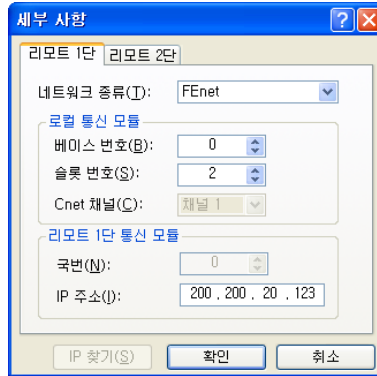
FNet 또는 EtherNet/IP 통신을 이용하여 리모트 1 단 접속하고자 하는 경우, 접속 방법으로 Ethernet 을 선택해야 하며, 통신 세부 사항 설정에서 IP 주소는 접속하고자 하는 PLC(PLC 1)에 장착된 FNet 또는 EtherNet/IP 모듈에 설정된 IP 주소를 입력합니다.

이 화면에서 'IP 찾기' 버튼을 선택하면 PC 가 포함된 Ethernet 네트워크에 연결된 모든 XGT PLC 의 Ethernet 모듈에 설정된 IP 주소가 표시됩니다. 접속하고자 하는 PLC 에 장착된 Ethernet 모듈의 IP 를 선택하고 확인 버튼을 클릭하면 선택된 PLC 를 접속할 수 있습니다. 또, RS-232C 를 이용한 리모트 접속의 경우 로컬 접속으로 선택 후 통신 세부 사항 설정 화면에서 PC 측의 통신 포트 번호 및 XGT Cnet 에 설정된 통신 속도를 이용하여 접속합니다. (115200/38400 bps 중 선택)

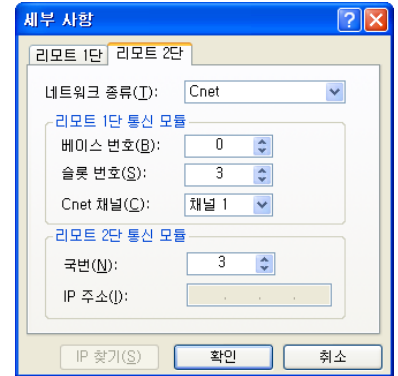
**로컬 & 리모트 2 단 접속:** 로컬 & 리모트 1 단으로 접속된 PLC(PLC 2)가 리모트 1 단 접속하고 있는 통신과 다른 종류의 통신 모듈을 이용하여 다른 PLC(PLC 3)와 통신하고 있을 때, PLC2 와 PLC3 간 통신하고 있는 모듈을 이용하여 접속하는 방법입니다.



통신 방법 및 단계 설정



리모트 1 단 통신 세부 사항 설정

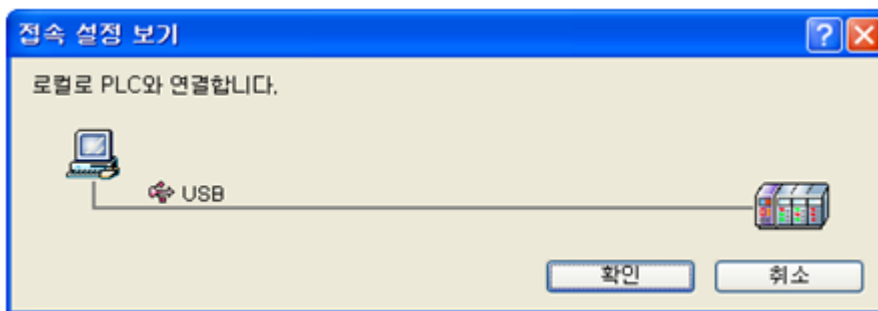


리모트 2 단 통신 세부 사항 설정

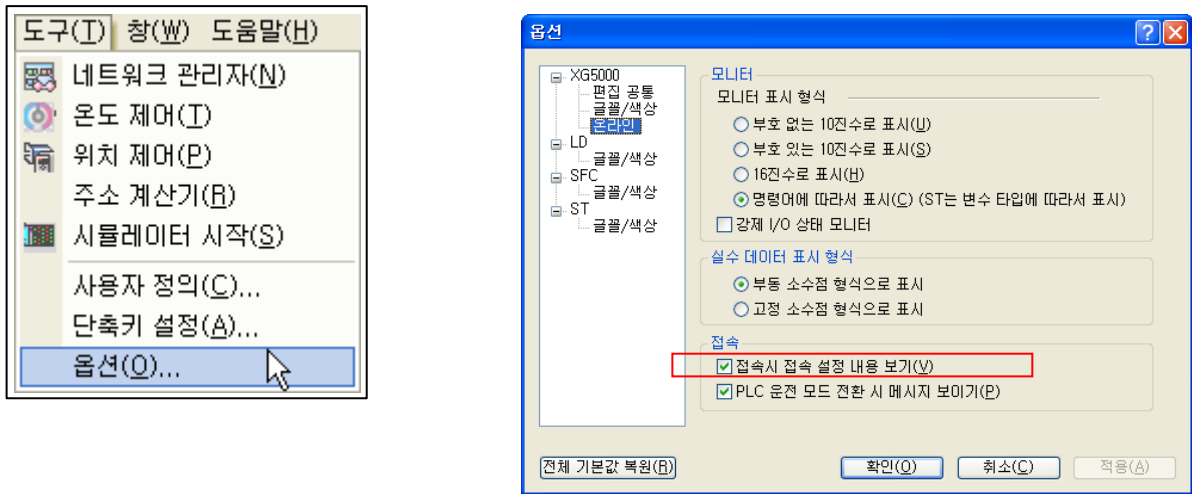
통신 세부 사항 설정에서 입력하는 IP 주소는 접속하고자 하는 PLC 에 장착된 Ethernet 모듈에 설정된 IP 주소입니다.

이 화면에서 'IP 찾기' 버튼을 선택하면 PC 가 포함된 Ethernet 네트워크에 연결된 모든 XGT PLC 의 Ethernet 모듈에 설정된 IP 주소가 표시됩니다. 접속하고자 하는 PLC 에 장착된 Ethernet 모듈의 IP 를 선택하고 확인 버튼을 클릭하면 선택된 PLC 를 접속할 수 있습니다. 또, RS-232C 를 이용한 리모트 1 단 또는 2 단 접속의 경우 통신 세부 사항 설정 화면에서 로컬(리모트 1 단 접속 시) 또는 리모트 1 단 (리모트 2 단 접속 시) 통신 모듈의 채널 번호를 선택하고 리모트 1 단 또는 리모트 2 단 통신 모듈 설정 항목의 국번에 접속할 PLC 에 장착된 Cnet 모듈의 국번을 입력합니다.

접속 접속 방법이 결정되었으면 통신 방법 및 단계 설정 화면에서 '접속' 버튼을 선택하거나, '온라인' 메뉴의 '접속'을 선택하면 접속 방법이 그림으로 표시됩니다. 이 화면에서 '확인' 버튼을 선택하면 PC 와 PLC 간 접속이 이루어 집니다.

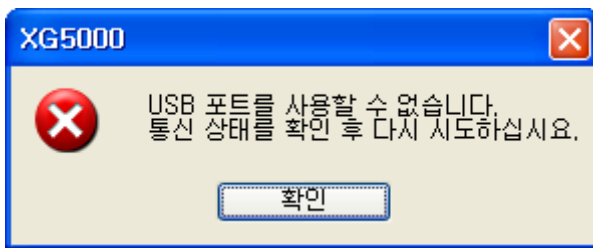


\* 접속 설정 보기 화면은 XG5000 '도구' 메뉴의 옵션 선택에 따라 표시되지 않을 수도 있습니다. 옵션의 '온라인'에서 '접속 시 접속 설정 내용 보기(V)'가 선택(체크)되어 있을 때 '접속'하면 접속 설정 내용이 화면에 표시 됩니다.

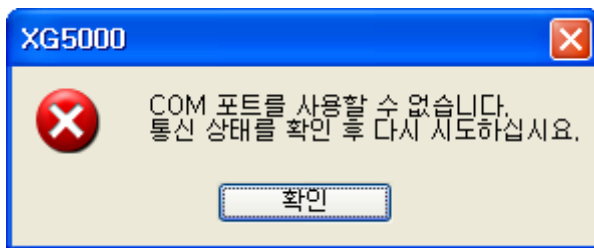


PC 와 PLC 와 접속을 했을 때 상황에 따라 몇 가지 메시지가 발생할 수 있습니다. 각 메시지에 따른 조치를 취한 후 다시 접속해 주십시오.

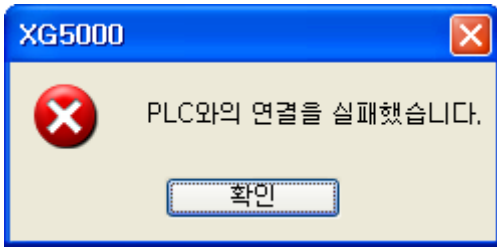
접속 방법으로 USB 를 선택하고 접속을 시도 했을 때 USB 케이블이 연결되지 않거나 PC 에 XGT PLC 의 USB 드라이브가 설치되지 않았을 때 나타나는 메시지 입니다. USB 케이블의 연결 상태를 점검하고, 만일 PC 에 XGT PLC 의 USB 드라이브가 설치되지 않았을 경우 USB 드라이브를 설치해 주십시오.



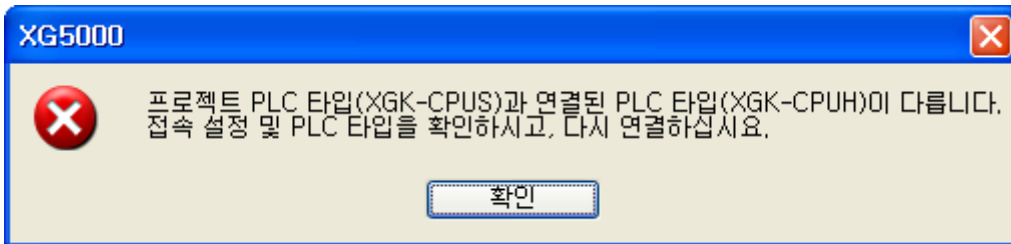
접속 방법으로 RS-232C 를 선택하고, 세부 사항 설정에서 지정한 통신 포트를 PC 에서 사용할 수 없을 경우 나타나는 메시지 입니다. 세부 사항 설정 창에서 PC 에서 사용 가능한 통신 포트를 지정해 주십시오.



접속 설정에서 선택한 통신 포트는 사용할 수 있으나 PLC와 통신이 이루어지지 않을 때 발생하는 메시지입니다. 로컬 접속을 사용할 경우 접속 케이블을 점검하고, 리모트 연결을 사용할 경우 접속 케이블 및 통신 파라미터를 점검해 주십시오.




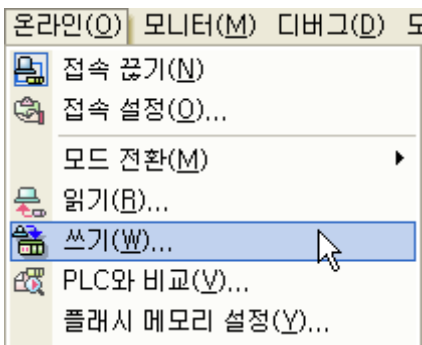
프로젝트의 PLC에 설정된 CPU 타입과 CPU 모듈의 타입이 서로 다른 경우 발생하는 메시지입니다. PLC의 속성에서 PLC CPU 타입을 변경해 주십시오.



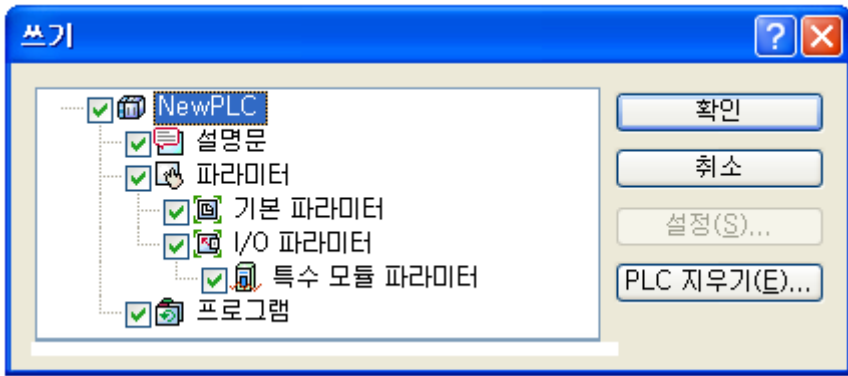
### 5) PLC로 전송 및 PLC 운전

프로그램의 작성이 완료되고 PC와 PLC 간 접속이 완료되면 PC에서 작성한 파라미터, 프로그램, 설명문을 PLC로 전송하고 PLC를 운전 시킵니다.

XG5000의 온라인 메뉴에서 '쓰기' 또는 단축 아이콘의 '쓰기' 아이콘 ()을 선택합니다.

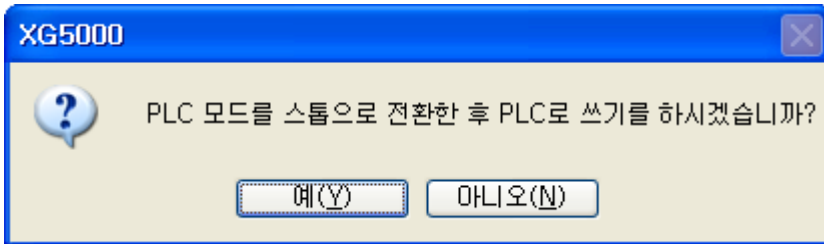


PLC로 전송할 항목 선택: 파라미터 및 프로그램이 PLC로 전송되면 그 이전에 PLC에 저장되어 있던 파라미터 및 프로그램은 모두 삭제되므로 이 전의 프로그램 및 파라미터를 보존하고자 할 경우 PLC로 전송하기 전 프로젝트 메뉴에서 'PLC로부터 열기' 또는 온라인 메뉴의 읽기를 실행하여 PLC에 저장되어 있는 파라미터 및 프로그램을 PC에 저장해야 합니다.

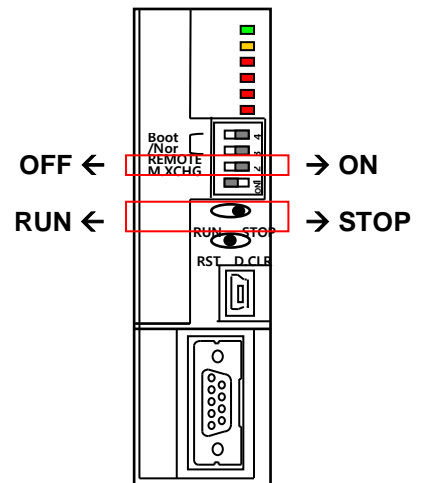
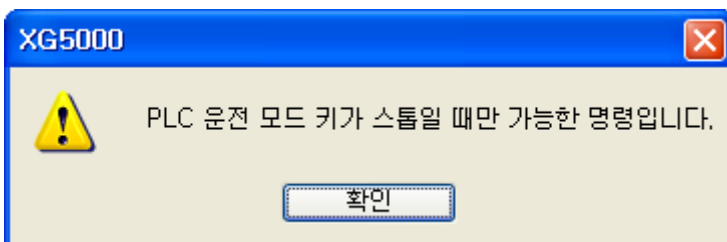
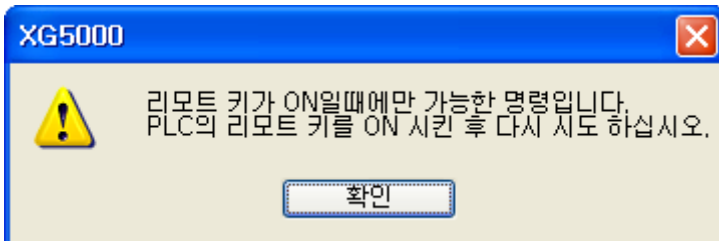


PLC 지우기는 PLC가 STOP 상태에서 선택이 가능하며, PLC에 새로운 파라미터 및 프로그램이 전송되면 PLC에 저장되어 있던 파라미터와 프로그램은 삭제되고, 래치 영역이 아닌 데이터 메모리 영역의 데이터는 삭제되지만, 래치 영역(K, R 영역 포함)에 저장되어 있는 데이터는 지워지지 않습니다. 새로운 프로그램을 전송하기 전에 래치 영역에 저장된 데이터를 지우고자 할 경우 'PLC 지우기' 버튼을 선택하여 래치 영역의 데이터를 지울 수 있습니다.

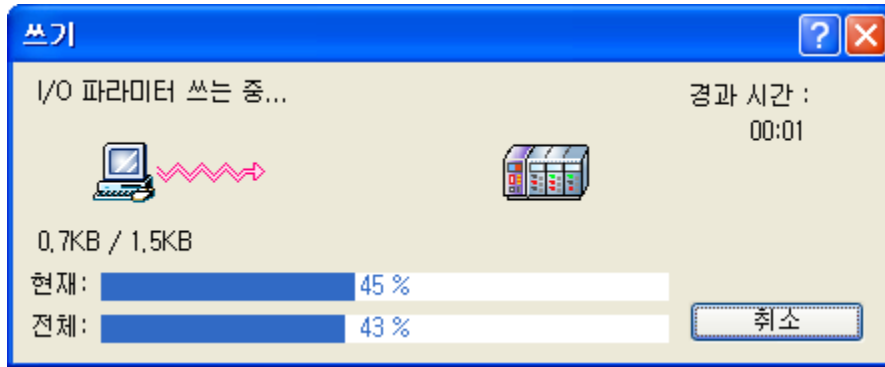
PLC STOP: 프로젝트를 전송할 때 PLC는 STOP 상태가 되어야 합니다. '쓰기'를 실행할 때 PLC가 RUN 상태이면 PLC를 STOP 모드로 전환할지 여부를 물어봅니다. 여기서 '예(Y)'를 선택하면 PLC를 STOP 시킨 후 프로젝트를 전송하며, '아니오(N)'를 선택하면 PLC는 RUN 상태를 유지하며, 프로젝트를 전송하지 않습니다. 단, 전송 목록에서 설명문만 선택되었을 경우 PLC가 RUN 상태에서도 설명문을 전송합니다.



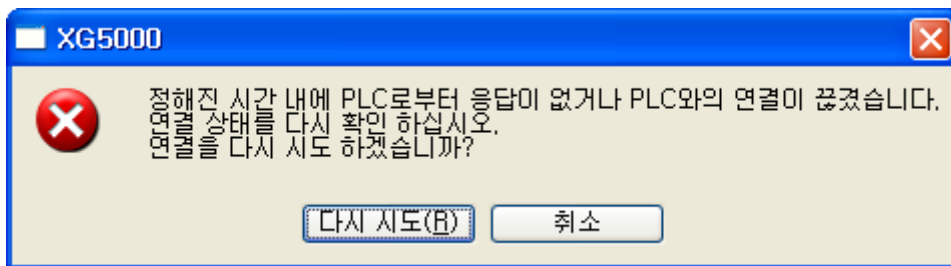
운전 모드 키 확인: 프로젝트를 전송할 때 PLC가 RUN 상태이면 XG5000에서 PLC의 운전 모드를 전환할 수 있어야 합니다. 그러나 CPU의 리모트 스위치가 OFF로 설정되어 있거나 운전 모드 스위치가 RUN 상태로 설정되어 있을 경우 다음과 같은 메시지가 표시됩니다. 이 경우 CPU의 REMOTE 스위치 및 RUN/STOP 스위치의 설정을 확인하십시오.



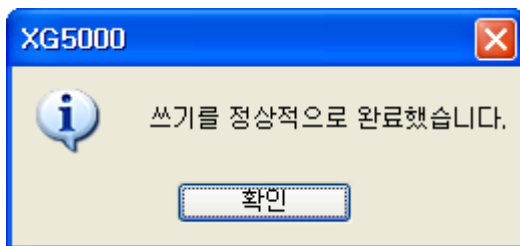
전송: PLC 가 STOP 모드로 전환되면 PC 에서 작성한 파라미터, 프로그램 및 설명문이 PLC 로 전송됩니다.  
 '쓰기'는 기본 파라미터, I/O 파라미터, 프로그램 및 설명문 순으로 PC 에서 PLC 로 전송됩니다.



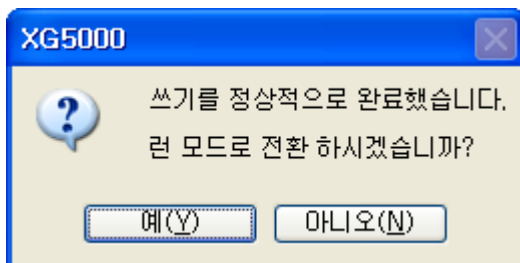
전송 도중 통신이 끊어졌을 경우, 다시 PC 와 PLC 간 접속한 후 '쓰기'를 재실행 해 주십시오.



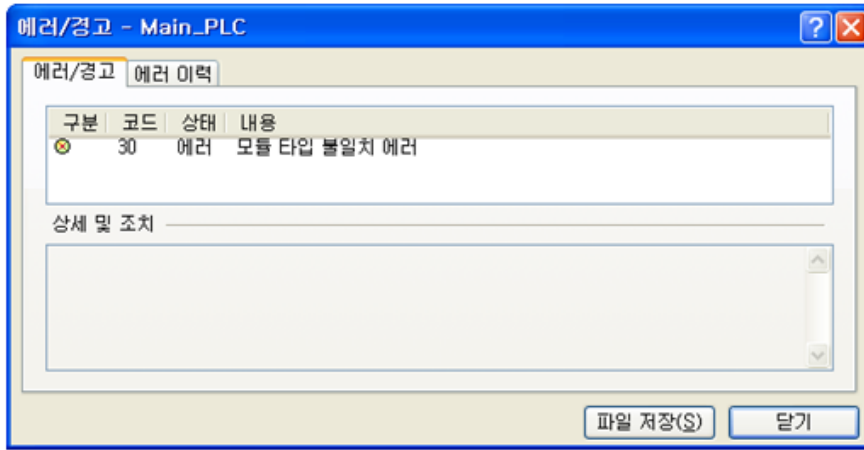
- '쓰기'가 정상적으로 완료되었으면 다음과 같은 메시지가 표시됩니다.



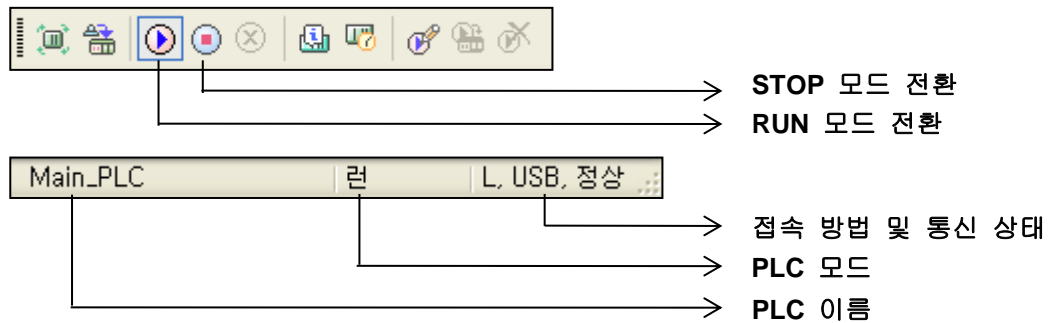
- '쓰기'를 하는 시점에서 PLC 가 RUN 상태였다면 전송이 완료된 후 다음과 같은 메시지가 표시됩니다. 여기서 '예(Y)' 버튼을 선택하면 PLC 는 RUN 모드로 전환되며, 정상적인 프로그램의 연산 및 수행이 시작됩니다. '아니오(N)' 버튼을 선택하면 PLC 는 STOP 모드를 유지하고, XG5000 에서 RUN 모드로 전환하거나 CPU 의 RUN 스위치를 ON 시키면 RUN 모드로 전환됩니다.



- PLC 를 RUN 상태로 전환했을 때 PLC 에 에러 또는 경고가 발생하면 다음과 같은 에러/경고 화면이 나타납니다. 에러/경고 화면에는 현재 발생하고 있는 에러 또는 경고의 내용이 나타납니다.



PLC 가 RUN 모드로 전환되면 단축 아이콘의 PLC 모드 전환 아이콘에 RUN 상태가 표시되며, XG5000 상태 표시 창에 PLC 이름, 운전 모드, 접속 방법 등이 표시됩니다.




### 6) 모니터링

PLC 가 가지고 있는 데이터를 XG5000 화면에 표시하는 기능을 모니터링이라고 합니다.

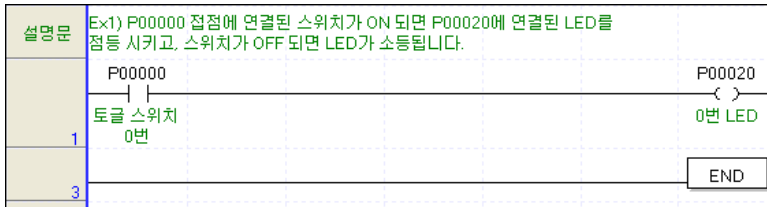
모니터링은 PLC 의 운전 상태 및 데이터를 시각적으로 표시해 주는 기능이므로 프로그램의 디버깅에 유용하게 사용할 수 있습니다.

XG5000 은 운전 데이터를 화면에 표시하는 기능 외에 변수 모니터, 시스템 모니터, 디바이스 모니터, 사용자 이벤트, 데이터 트레이스 등의 다양한 모니터링 기능을 제공함으로써 사용자가 쉽고 편하게 디버깅 할 수 있습니다. 또한, 내부 디바이스의 경우 XG5000 에서 데이터의 변경도 가능합니다.

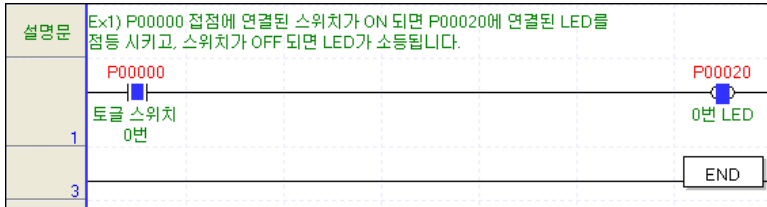
XG5000 과 PLC 가 접속된 상태에서 '모니터' 메뉴의 '시작'을 선택하거나 단축 아이콘의 '모니터 시작/끝'

버튼 (  )을 선택하면 모니터링 기능이 시작됩니다.

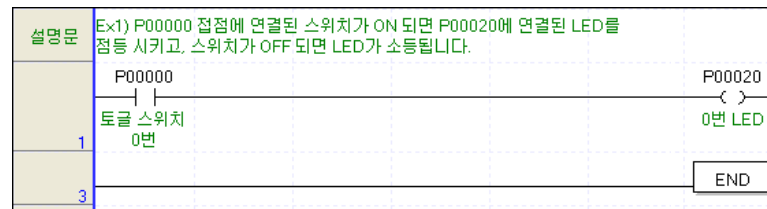
래더 모니터링(Ladder Monitoring)은 XG5000 에서 래더 프로그램이 열려있는 상태에서 모니터가 시작되면 래더 프로그램에 PLC 의 현재 데이터를 표시하는 것을 의미합니다. 래더 모니터링에서 접점, 코일 등 비트 데이터 표시: 디바이스 또는 변수의 표시 색과 래더 기호의 색을 통해 데이터 및 도통 상태를 확인할 수 있습니다.



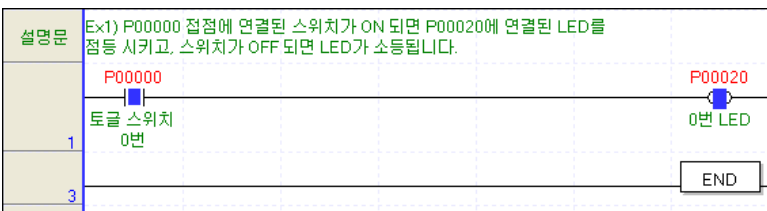
P00000 접점이 OFF 되었을 때



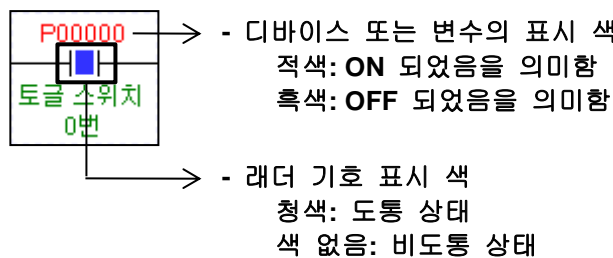
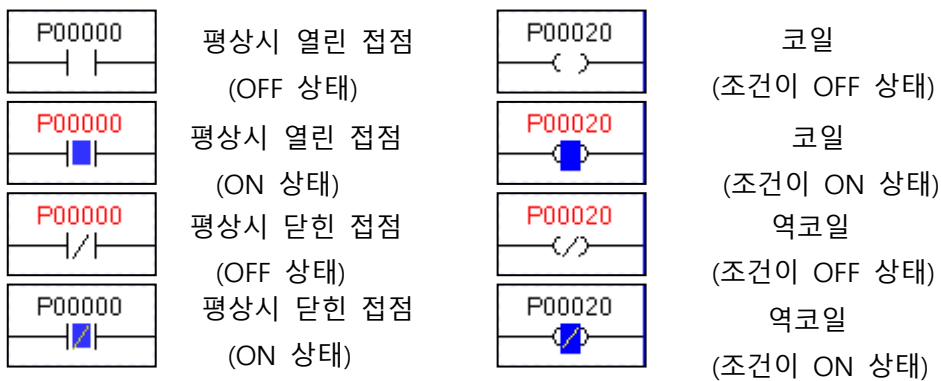
P00000 접점이 ON 되었을 때



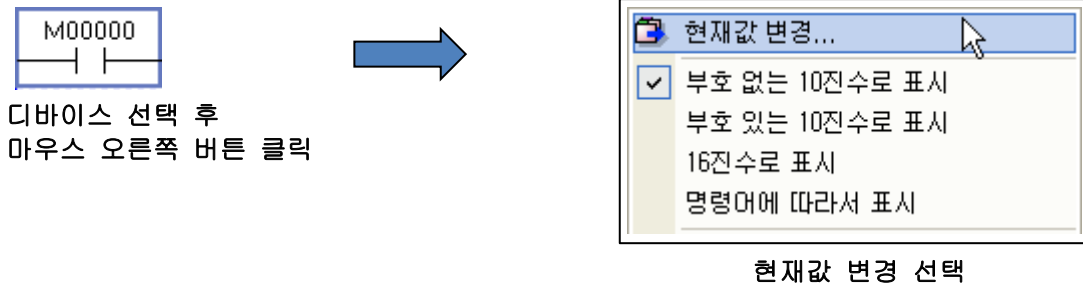
P00000 접점이 OFF 되었을 때



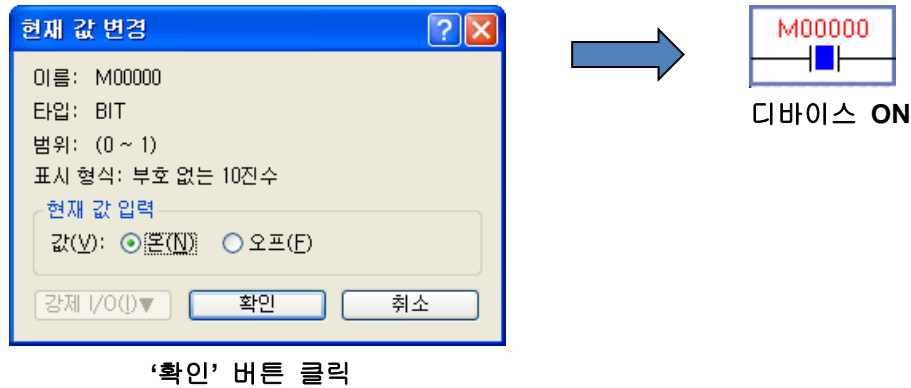
P00000 접점이 ON 되었을 때



래더 모니터링에서 데이터 변경: 래더 모니터링 중 XG5000 에서 PLC 의 데이터를 변경할 수 있습니다. 데이터를 변경하고자 하는 디바이스를 선택한 후 마우스 오른쪽 버튼을 누르면 팝-업 메뉴가 나타납니다. 팝-업 메뉴에서 '현재값 변경'을 선택합니다.



선택된 디바이스가 비트(접점)일 경우 현재값 변경 화면에서 '온(N)' 또는 '오프(F)'를 선택할 수 있으며, 현재값 변경 화면이 표시될 때 디바이스 현재값이 반전되어 표시되므로 '확인' 버튼을 누르면 데이터가 변경됩니다. 즉, PLC 에서 선택된 디바이스의 현재값이 OFF 상태이면 현재값 변경 화면에 '온(N)'이 선택되므로 확인 버튼을 누르면 됩니다.

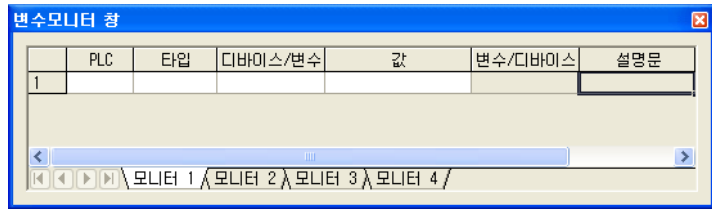


\* PLC 가 RUN 모드일 때 연산 결과를 저장하는 영역일 경우 데이터를 변경했을 때 연산 조건에 따라 데이터가 변경되지 않을 수 있습니다.

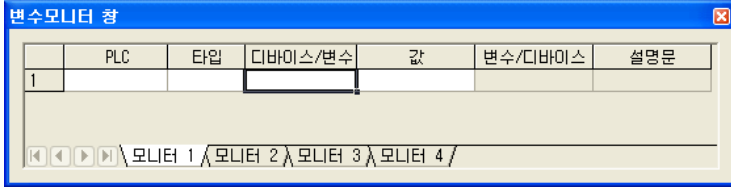
\* 선택된 디바이스가 디지털 입력 모듈의 입력 주소일 경우 현재값 변경 기능을 사용하여 데이터를 변경하더라도 값이 변경되지 않으며, 실제 입력 데이터가 표시됩니다.

\* 워드 데이터의 현재값 변경 화면에서 표시되는 데이터 타입은 XG5000 의 모니터링 데이터 타입에 영향을 받습니다. 예를 들면 모니터링 데이터 타입을 부호 있는 십진수로 선택하고 데이터를 50,000 으로 입력했을 경우, 래더 프로그램에서 -15,536 으로 표시됩니다.

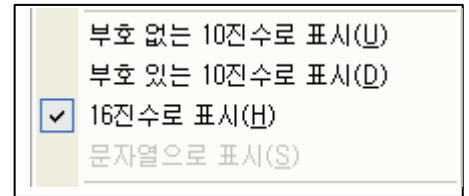
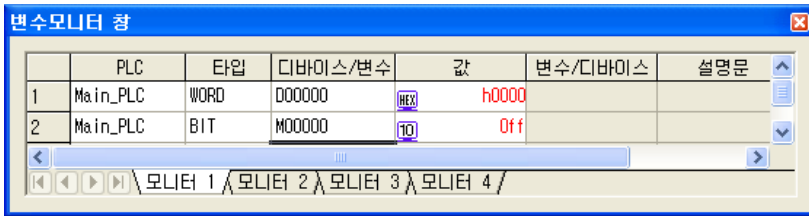
변수 모니터(Variable Monitoring): 래더 프로그램에 보여지지 않는 데이터 메모리의 데이터를 모니터링 할 때 사용하는 기능입니다. 변수 모니터 창에 모니터링 하고자 하는 디바이스를 등록하면 등록된 디바이스에 저장되어 있는 현재값이 표시됩니다.



디바이스 등록: 변수 모니터 창이 실행되면 모니터링할 변수를 등록해야 합니다.



'디바이스/변수' 열을 선택합니다. 모니터링하고자 하는 디바이스를 입력하면 '값' 열에 해당 디바이스의 현재 값이 표시됩니다. 셀 선택 후 마우스 오른쪽 버튼을 누르면 표시되는 데이터의 타입을 변경할 수 있습니다.

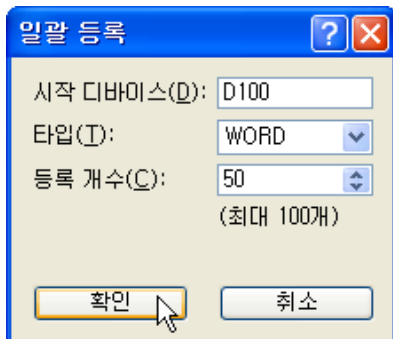


'디바이스/변수' 입력 후 타입을 선택하면 데이터 크기 및 종류를 선택할 수 있습니다.

일괄 등록: 모니터링할 변수의 주소가 연속적일 때 '일괄 등록' 기능을 사용하면 편리하게 모니터링할 변수를 등록할 수 있습니다.

'디바이스/변수' 열을 선택합니다.

마우스 오른쪽 버튼을 눌러 나타나는 팝-업 메뉴에서 '일괄 등록(R)'을 선택합니다.



시작 디바이스, 타입, 등록 개수를 입력한 후 확인 버튼을 선택하면 시작 디바이스부터 등록 개수만큼 연속된 디바이스로 지정된 타입으로 변수 모니터창에 등록되며, 각 디바이스의 현재값이 표시됩니다. 시작 디바이스가 워드 디바이스(D, R, ZR 등)일 경우 타입은 WORD로 자동 변경되며, 비트 디바이스(P, M, K, L 등)일 경우 워드 단위로 모니터링 하고자 하면 타입을 워드로 바꾸어 주어야 합니다.

한 번에 등록할 수 있는 디바이스의 최대 수는 100 개입니다. 100 개 이상의 연속된 디바이스를 등록하기 위해서는 위의 과정을 여러 번 반복하여 등록할 수 있습니다. 하나의 모니터 창에 등록할 수 있는 데이터의 수는 제한이 없습니다.

### 7) 런 중 수정

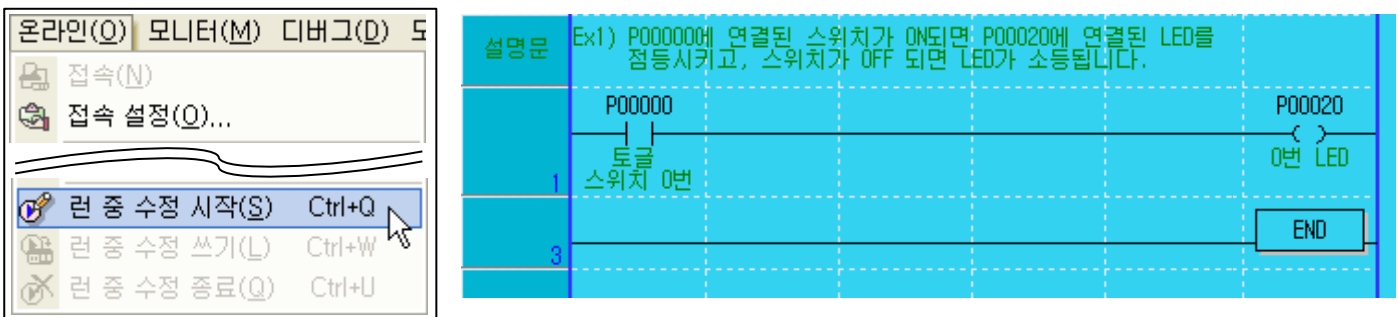
PC 에서 작성한 프로그램을 PLC 로 전송하기 위해서는 PLC 를 정지시켜야 합니다. 그러나 PLC 특성상 PLC 를 정지시키지 않은 상태에서 프로그램을 수정 또는 추가해야 할 경우가 있습니다. 이럴 경우 '런 중 수정' 기능을 이용하면 PLC 를 정지시키지 않고 프로그램을 수정 또는 추가할 수 있습니다.

XGT PLC 에서 PLC 를 정지 시키지 않은 상태에서 수정할 수 있는 것은 PLC 에 저장되어 있는 프로그램의 수정으로 한정되며, 프로그램 블록의 추가 또는 프로그램 블록 전체의 삭제를 할 수 없으며, 파라미터의 수정도 할 수 없습니다.

기본 파라미터 및 I/O 파라미터를 변경하고자 할 경우 PLC 를 정지 시킨 후 'PLC 로 쓰기'를 수행해야 수정된 파라미터가 PLC 의 운전에 반영됩니다.

#### 런 중 수정 순서

런 중 수정 시작: 단축 아이콘 ( )을 선택하거나 온라인 >> 런 중 수정(단축키: Ctrl + Q)을 선택하여 '런 중 수정'을 시작합니다. '런 중 수정'이 시작되면 XG5000 프로그램 창의 바탕색이 변경됩니다.

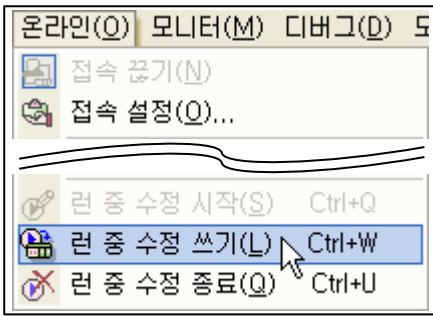


\* 런 중 수정 시작의 단축키는 Ctrl + Q 입니다.

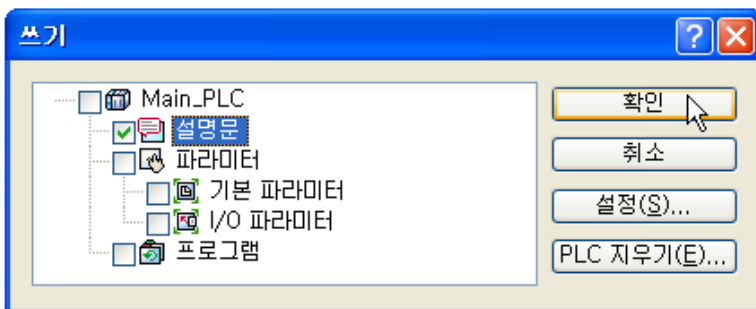
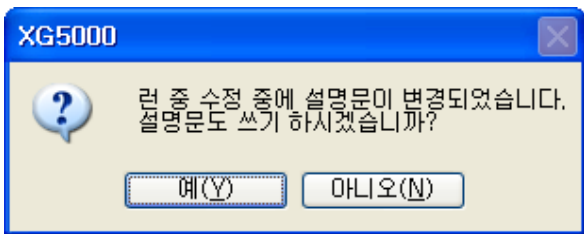
\* 런 중 수정 시 프로그램 화면의 배경색은 XG5000 의 도구 >> 옵션 >> LD >> 글꼴/색상에서 변경할 수 있습니다.

런 중 수정 쓰기: 프로그램의 수정이 완료되면 수정된 프로그램을 PLC 로 전송합니다.

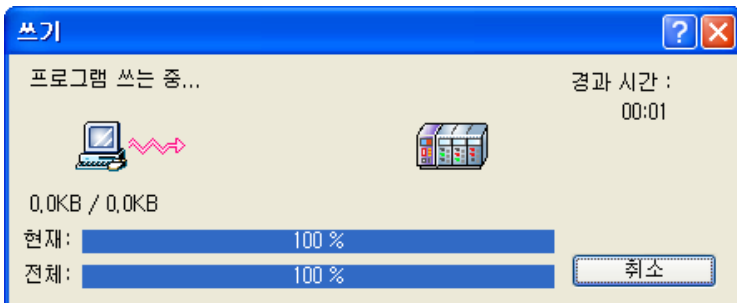
단축 아이콘의 ( )을 선택하거나 온라인 >> 런 중 수정 쓰기(단축키: Ctrl + W)를 선택합니다.



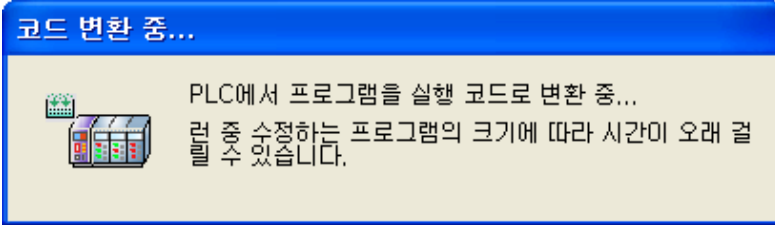
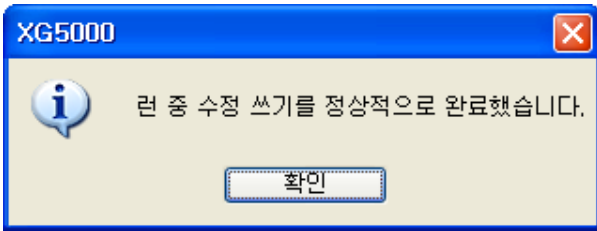
런 중 수정 과정에서 디바이스에 대한 설명문이 추가되거나 등록되어 있던 설명문을 삭제했을 경우 설명문을 PLC 에 저장할 지 여부를 확인합니다. 여기서 '예 ( Y )'를 선택하면 런 중 수정된 프로그램을 PLC 로 전송할 때 설명문도 전송이 되지만, '아니오( N )'를 선택하면 런 중 수정 과정에서 변경된 설명문은 PLC 로 전송되지 않습니다. 여기에서 '아니오( N )'를 선택하여 수정된 설명문을 PLC 로 전송하지 않았을 경우 런 중 수정 완료 후 온라인 >> 쓰기를 선택한 후 '설명문'만 선택하여 '확인'을 선택하면 PLC 운전 중 설명문만 PLC 로 전송됩니다.



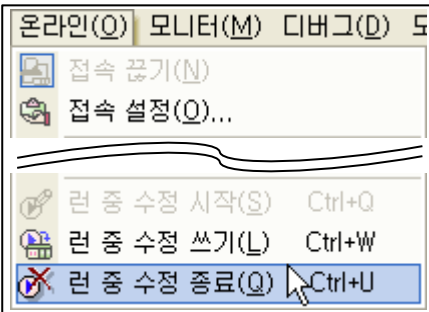
런 중 수정 쓰기가 시작되면 프로그램이 PLC 로 전송됩니다. 이 때 런 중 수정 과정에서 변경된 설명문 쓰기를 선택한 경우 설명문도 PLC 로 전송됩니다.



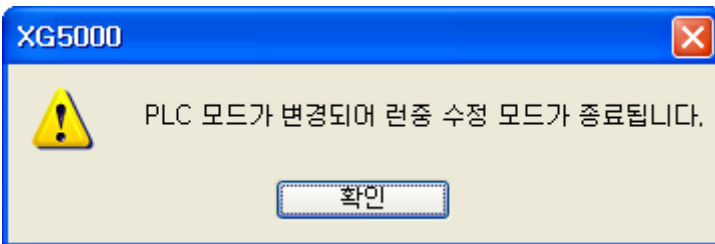
프로그램 쓰기가 완료되면 PLC 는 XG5000 으로부터 전송된 프로그램을 PLC 의 실행코드로 변환합니다. 변환이 완료되면 런 중 수정 완료 메시지가 나타나며, 이 때부터 수정된 프로그램이 PLC 에서 연산됩니다.



런 중 수정 종료: 런 중 수정 쓰기가 완료되면 런 중 수정 종료 아이콘 ( )을 선택하거나 온라인 >> 런 중 수정 종료를 선택하여 런 중 수정을 종료합니다. 런 중 수정이 종료되면 프로그램 창의 바탕색이 흰색으로 바뀝니다.

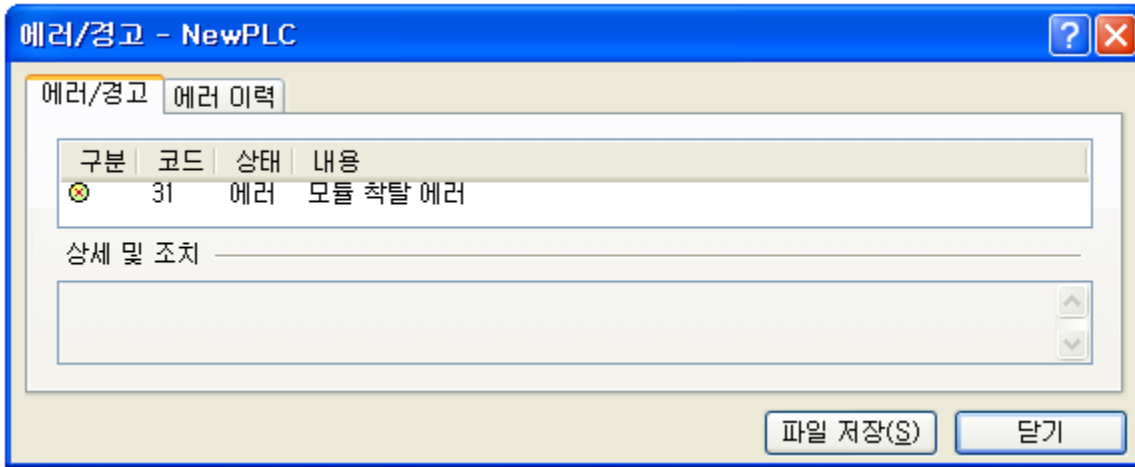


런 중 수정 중 에러 사항에 대한 조치: 런 중 수정은 PLC가 런 상태이면서 XG5000과 PLC 간 통신이 이루어져야 가능한 작업입니다. 여기서 런 중 수정 중에 발생할 수 있는 에러 사항에 대한 조치에 대해서 설명합니다. 런 중 수정 도중 PLC가 STOP 모드로 변경되었을 경우 아래와 같은 메시지가 발생하며, 런 중 수정 모드는 자동으로 해제됩니다. 수정된 내용이 PLC로 전송되지 않았을 경우 전체 쓰기를 실행합니다.



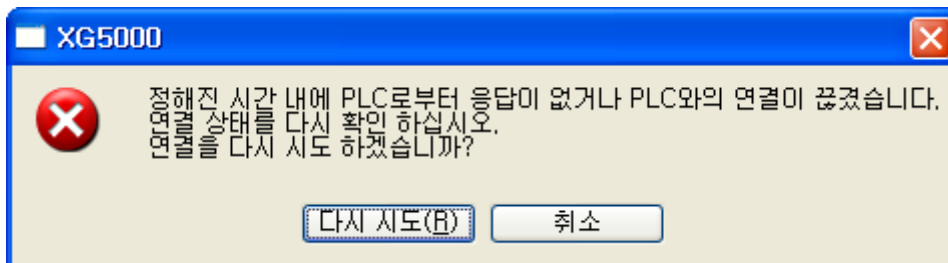
런 중 수정 도중 PLC에 에러가 발생했을 경우 아래와 같이 에러 메시지가 나타나지만, 런 중 수정 모드는 유지하고 있습니다.

PLC에 에러가 발생했을 때 에러 원인을 제거하고 PLC를 리셋해야 에러가 해제되기 때문에 런 중 수정 도중 PLC에 에러가 발생하면 런 중 수정하던 프로젝트를 '저장'하고 에러 원인을 제거한 후 PLC를 리셋한 후 저장된 프로그램 전체를 PLC로 전송합니다.

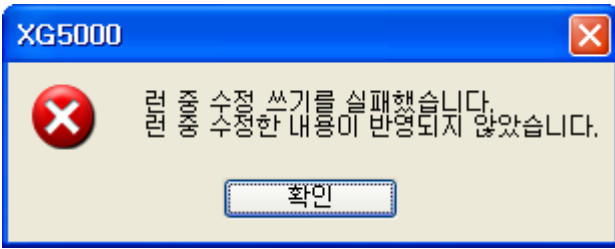


런 중 수정 도중 PLC 와 통신이 끊겼을 경우 아래와 같은 메시지가 발생합니다. 이 때, XG5000 을 그대로 놓아 둔 상태에서 통신을 복구한 후 '다시 시도 ( R )' 버튼을 눌러 정상적으로 통신이 이루어 지면 계속 런 중 수정 상태로 유지되므로 계속적인 런 중 수정 및 런 중 수정 쓰기를 진행할 수 있습니다. 만일 이 메시지에서 '취소'를 선택하면 런 중 수정 모드가 해제되며, 통신 복구 후 전체 쓰기를 실행합니다. 그러나 수정한 내용이 많고 PLC 를 정지시킬 수 없는 상황이라면 다음의 과정을 통해 간단히 복구할 수 있습니다.

- ① 런 중 수정하던 파일을 다른 이름으로 저장
- ② 통신 복구 후 PLC 로부터 열기 → 런 중 수정 시작
- ③ 새로운 XG5000 을 실행시켜 다른 이름으로 저장한 파일 열기
- ⑤ 새로운 XG5000(다른 이름으로 저장된 파일)에서 수정된 부분 선택 후 복사 실행
- ⑥ 런 중 수정 중인 XG5000 에 붙여넣기 실행
- ⑦ 새로운 XG5000(다른 이름으로 저장된 파일)에서 변수/설명을 열어 전체 복사 실행
- ⑧ 런 중 수정 중인 XG5000 에 변수/설명 전체 선택하여 삭제 후 붙여넣기 실행

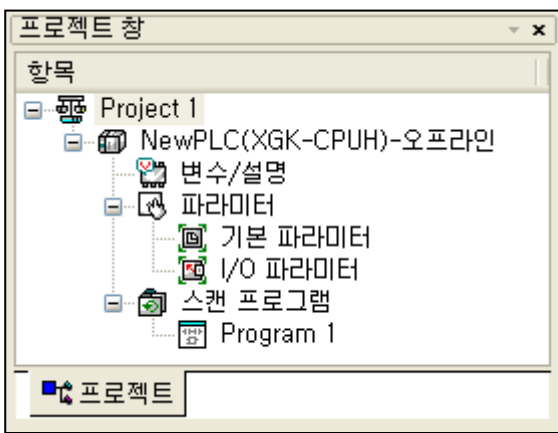


런 중 쓰기 도중 PLC 와 통신이 끊겼을 경우도 위의 메시지가 발생하며, 통신을 복구한 후 다시 시도 ( R )을 선택하면 수정된 프로그램이 다시 PLC 로 전송됩니다. 전송이 완료되면 실행 코드로 변환이 시작됩니다. 런 중 쓰기 도중 PLC 에 에러가 발생했을 경우 아래의 메시지가 발생하며 런 중 수정한 내용이 실행되지 않습니다. 에러 해제한 후 전체 쓰기를 합니다.



### 3.4 XG5000 프로젝트 구조

XG5000 에서 프로젝트는 PLC 로 구성되며, 한 개의 프로젝트에는 한 대 이상의 PLC 를 등록할 수 있습니다. PLC 의 하부 구성 요소로서 변수/설명, 파라미터, 프로그램, 태스크가 있으며, 프로그램은 스캔 프로그램과 태스크 프로그램이 있습니다.



프로젝트: 제어의 총괄 개념으로 1 대 이상의 PLC 로 구성됩니다. 특히, 멀티 PLC 프로젝트(2 대 이상의 PLC 가 등록된 프로젝트)는 PLC 간 네트워크 연결 시 강력한 기능을 발휘할 수 있습니다. 멀티 PLC 구성 시 구성 제품군에 다음과 같은 제약이 있습니다.

- XGK 언어 계열 PLC: XGK 시리즈, XGB 시리즈 중 XBM, XBC CPU
- XGI 언어(IEC 언어) 계열 PLC: XGR 시리즈, XGI 시리즈, XGB 시리즈 중 XEC CPU

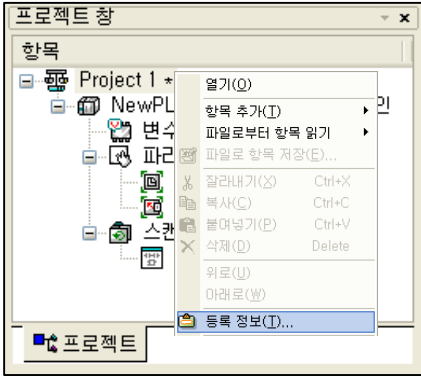
PLC: 실제 제어를 담당하는 단위(CPU)로 1 대의 PLC 는 각각의 변수/설명, 파라미터, 프로그램으로 구성되며, 프로그램은 스캔 프로그램과 태스크(Task) 프로그램으로 구분됩니다.

- 변수/설명: PLC 프로그램에서 사용한 변수 및 각 변수에 대한 설명문을 편집할 수 있습니다. 또한 PLC 프로그램 과정에서 변수 및 설명문을 편집했을 경우 그 내용을 변수/설명문에서 확인 및 수정할 수 있습니다.
- 파라미터: PLC 기종에 따라 파라미터의 종류가 달라집니다. XGK PLC 의 경우 기본 파라미터와 I/O 파라미터로 구성됩니다.
- 스캔 프로그램: PLC 가 RUN 상태이면 연산이 수행되는 프로그램입니다.
- 태스크 프로그램: PLC 가 RUN 상태이면서 특정 조건(정주기, 내부 접점 ON/OFF)이 만족될 때만 수행되는 프로그램입니다. 태스크 프로그램을 작성하기 위해서는 먼저 태스크가 등록되어야 합니다.

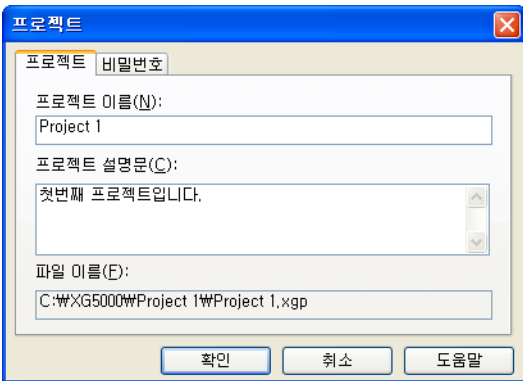
**프로젝트 항목의 편집**

XG5000 에서 프로젝트는 PLC 로 구성되며, 한 개의 프로젝트에는 한 대 이상의 PLC 를 등록할 수 있습니다. PLC 의 하부 구성 요소로서 변수/설명, 파라미터, 프로그램, 태스크가 있으며, 프로그램은 스캔 프로그램과 태스크 프로그램이 있습니다.

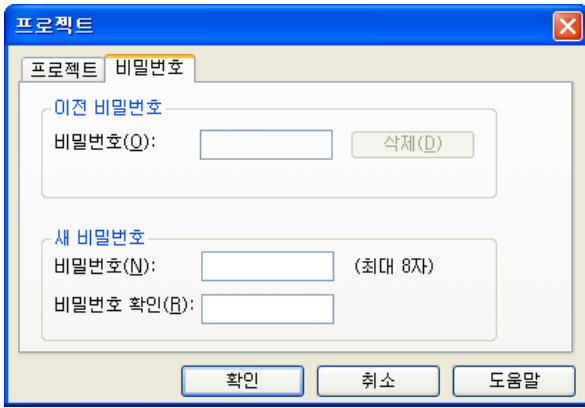
프로젝트 등록 정보 편집: 프로젝트 생성 시에 지정했던 프로젝트 정보를 편집할 수 있습니다.



마우스로 프로젝트를 선택합니다. (클릭)  
 마우스의 오른쪽 버튼을 클릭하여 팝업 메뉴 창을 호출합니다.  
 팝업 메뉴 창에서 '등록 정보(T)...'를 선택합니다.

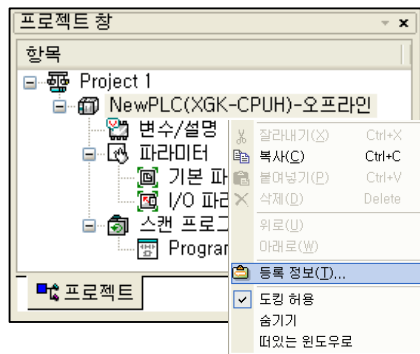


프로젝트 이름: 프로젝트 이름을 변경할 수 있습니다.  
 프로젝트 설명문: 프로젝트 설명문을 변경할 수 있습니다.  
 파일 이름: 프로젝트 파일의 저장 위치와 파일명을 표시합니다. 여기서 파일의 저장 위치 및 파일 이름을 변경할 수 없습니다. 따라서 프로젝트 이름을 변경했을 경우 프로젝트 이름과 프로젝트 파일명이 달라질 수 있습니다. 프로젝트 이름과 프로젝트 파일명을 동일하게 변경하고자 할 경우 XG5000 의 프로젝트 메뉴에서 '다른 이름으로 저장'을 실행하십시오.



비밀번호: 프로젝트에 비밀번호를 설정합니다. 여기서 설정된 비밀번호는 XG5000 에서 파일을 열 때 적용되는 비밀번호입니다. 즉, 여기서 비밀번호를 설정하고 파일을 저장한 뒤 다시 열 때 XG5000 에서 비밀번호를 물어봅니다. 입력한 비밀번호가 여기서 설정한 비밀번호와 일치하지 않을 경우 파일은 열리지 않습니다. 여기서 설정한 비밀번호는 PLC 로 전송되지 않습니다. 따라서 여기서 비밀번호를 설정하고 PLC 로 전송한 다음 다시 PLC 로부터 프로젝트를 읽었을 때는 비밀번호를 묻지 않습니다. 'PLC 로부터 열기'(업로드) 기능에 대한 비밀번호는 온라인 메뉴의 PLC 정보 항목의 비밀번호에서 설정해야 합니다. 비밀번호를 설정했을 경우 반드시 기억해야 합니다. 비밀번호를 잊어버렸을 경우 프로젝트를 열 수 없으며, 비밀번호를 확인할 수 있는 방법도 없습니다.

PLC 속성 편집: 프로젝트 생성 시 PLC 이름은 NewPLC 로 주어지며, 프로젝트 생성 시 선택했던 PLC CPU 기종이 표시됩니다. 속성 변경을 통해 PLC 이름 및 PLC 기종을 변경할 수 있습니다.

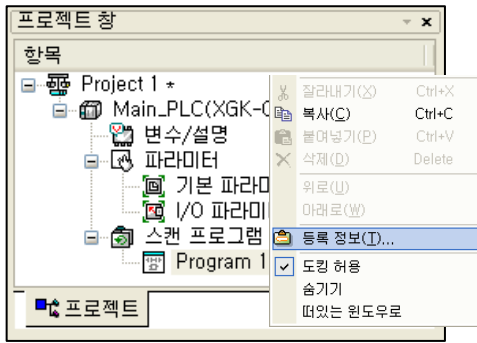


PLC 이름 변경 순서

- PLC 이름을 선택한 후 마우스 오른쪽 버튼 클릭합니다.
- 팝업 메뉴에서 '등록 정보(T)...' 선택합니다.
- PLC 등록 정보 대화 상자에서 PLC 이름 및 설명문의 변경이 가능합니다.
- PLC 종류에서 다음과 같이 PLC CPU 기종 변경이 가능합니다.

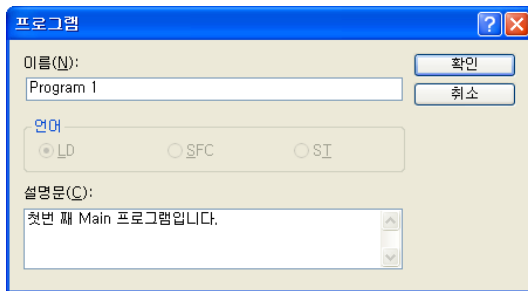
XGK ← → XGB(XBM , XBC)      XGR ← → XGI ← → XGB(XEC)

프로그램 속성 편집: 프로젝트 생성 시 지정했던 프로그램 이름 및 프로그램에 대한 설명문을 편집할 수 있습니다.

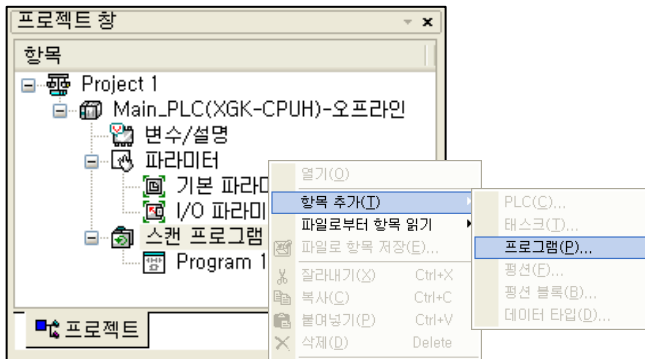


**프로그램 속성 편집 순서**

- 프로그램 이름을 선택하고 마우스 오른쪽 버튼을 클릭합니다.
- 팝업 메뉴에서 '등록 정보(T)...'를 선택합니다.
- 프로그램 등록정보 팝업 창에서 프로그램 이름 및 설명문을 편집할 수 있습니다.



프로그램 추가: XGT PLC 는 최대 256 개(XBM 은 128 개)의 프로그램으로 나누어 작성할 수 있습니다. 각 프로그램의 연산 순서는 스캔 프로그램에 등록된 순서대로 연산 됩니다.



**프로그램 추가 순서**

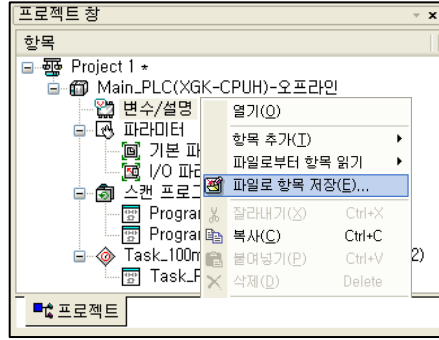
- 프로젝트 창에서 스캔 프로그램을 선택하고 마우스 오른쪽 버튼 클릭합니다.
- 팝업 창에서 '항목 추가'를 선택하고 '프로그램(P)...'를 선택합니다.
- 프로그램 이름 및 설명문을 입력합니다.

프로그램 등록 순서 변경: 여러 개의 프로그램이 등록된 경우 등록된 수서에 따라 연산을 수행하기 때문에 경우에 따라서 프로그램의 등록 순서를 변경해 주어야 할 수 있습니다.

- 프로젝트 창에서 1 개의 프로그램을 선택하고 마우스 오른쪽 버튼 클릭합니다.
- 팝업 창에서 '위로(U)' 또는 '아래로(W)'를 선택하여 프로그램의 등록 위치를 변경합니다.

프로젝트 항목의 저장: 프로젝트에 등록된 프로그램, 변수/설명, 파라미터 등은 별도의 파일로 저장하여

다른 프로젝트에서 재사용 할 수 있습니다.

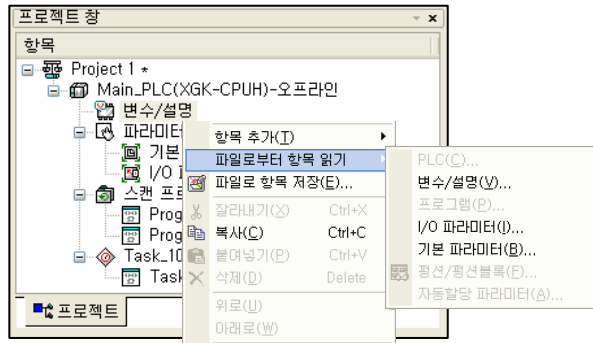


**항목 저장 순서**

- 프로젝트 창에서 저장할 항목을 선택하고 마우스 오른쪽 버튼을 클릭합니다.
- 저장 대화상자에서 저장할 파일 이름을 선택하고 '저장' 버튼을 누르면 선택된 프로젝트 항목이 저장됩니다.
- 저장하는 항목에 따라 확장자가 달라집니다.

- PLC: \*\*\*\*.plc
- 변수/설명: \*\*\*\*.cmt
- 기본 파라미터: \*\*\*\*.bsp
- I/O 파라미터: \*\*\*\*.iop
- 프로그램: \*\*\*\*.prg

파일로부터 항목 열기: 다른 프로젝트에서 작성하여 파일로 저장되어 있는 프로젝트의 항목을 현재의 프로젝트에서 재사용 할 수 있습니다.



**항목 읽기 순서**

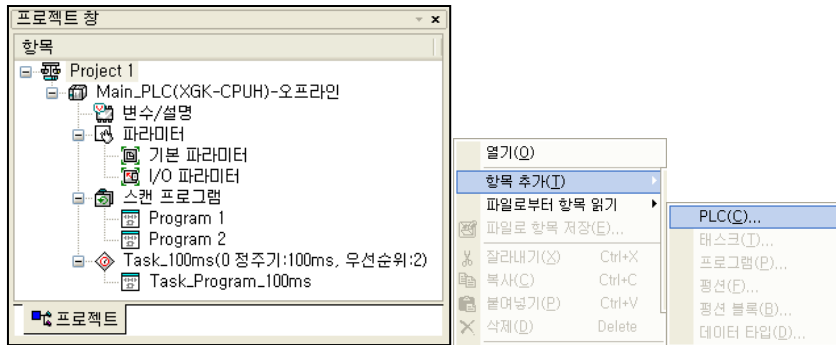
- 프로젝트 창에서 읽을 항목을 선택하고 마우스 오른쪽 버튼을 클릭합니다.
- 파일 열기 대화상자에서 저장할 파일을 선택하고 '열기' 버튼을 누르면 선택된 프로젝트 항목이 현재의 프로젝트에 포함됩니다.
- 선택하는 항목에 따라 추가할 수 있는 항목이 달라집니다.

- 프로젝트선택 시: PLC 추가 가능
- PLC 선택 시: 변수/설명, I/O 파라미터, 기본 파라미터 추가 가능
- 스캔/태스크 프로그램 선택 시: 프로그램 추가 가능

- 읽는 항목에 따라 찾는 파일의 확장자가 달라집니다.

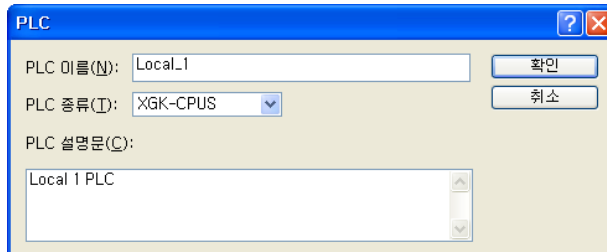
- PLC: \*\*\*\*.plc
- 변수/설명: \*\*\*\*.cmt
- 기본 파라미터: \*\*\*\*.bsp
- I/O 파라미터: \*\*\*\*.iop
- 프로그램: \*\*\*\*.prg

PLC 추가: XG5000 프로젝트는 1 대 이상의 PLC 로 구성할 수 있습니다. 네트워크로 연결된 여러 대의 PLC 가 1 개의 프로젝트에 포함될 경우 여러 PLC 를 동시에 접속, 모니터링, 프로그램 수정할 수 있습니다.



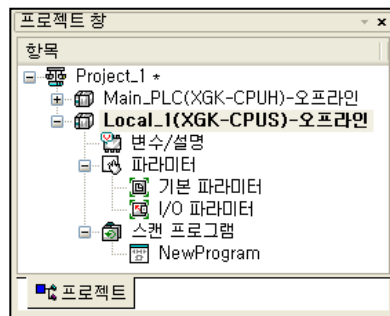
PLC 추가 순서

- 프로젝트 창에서 프로젝트 이름 선택하고 마우스 오른쪽 버튼을 클릭합니다.
- 팝업 메뉴에서 '항목 추가'를 선택하고 'PLC(C)...'를 선택합니다.



PLC CPU 종류를 선택하고 PLC 이름 및 설명문을 입력합니다.

선택할 수 있는 PLC 종류는 프로젝트 구성 시 등록된 PLC 기종에 따라 XGK, XBM, XBC 계열과 XGR, XGI, XEC 계열로 구분됩니다.



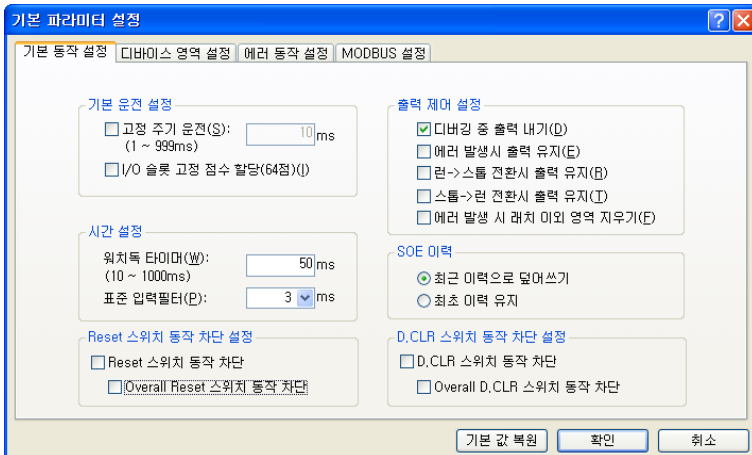
1 개의 프로젝트에 2 대의 PLC 가 등록되었습니다.

프로젝트에 등록된 PLC 는 별도의 파일로 저장 및 읽기, 복사 및 붙여 넣기 등의 기능을 이용하여 다른 프로젝트로 복사가 가능합니다.

동일 프로젝트에 등록된 여러 대의 PLC 간 서로 통신을 할 경우 동시 접속, 동시 모니터링, 동시 런 중 수정 등 다양한 편리 기능을 사용할 수 있습니다.

### 기본 파라미터

기본 파라미터는 PLC 운전의 기본적인 속성을 지정합니다. 기본 파라미터를 편집하기 위해서 파라미터 창에서 기본 파라미터를 더블 클릭하여 기본 파라미터 설정 화면을 호출합니다.



#### 1) 기본 동작 설정

##### \* 기본 운전 설정

- 고정 주기 운전: PLC의 스캔 시간을 고정 시간으로 운전할 때 설정합니다. 설정 시간은 1 ~ 999ms까지 1ms 단위로 설정할 수 있으며, 실제 설정할 수 있는 시간은 프로그램 작성 후 프로젝트를 PLC로 전송한 후 최대 스캔 시간을 확인하고 최대 스캔 시간보다 길게 지정해야 합니다.
- I/O 슬롯 고정 점수 할당(64점): I/O 주소를 고정식으로 할당할 지 가변식으로 할당할 지 선택합니다. 이 항목이 선택되면(체크) 고정식 I/O 할당 방식을 사용합니다.

##### \* 시간 설정

- 위치독 타이머: PLC의 연산 폭주, 무한루프 등을 감시하기 위해 설정하는 시간입니다. 여기에 지정된 시간 내에 PLC의 1 스캔이 완료되어야 하며, 위치독 타이머 설정 시간 이내에 1 스캔의 처리가 완료되지 않으면 PLC는 에러 상태가 됩니다.
- 표준 입력 필터: 디지털 입력 모듈에 대해 ON 또는 OFF 상태를 유지해 주어야 하는 시간입니다. 디지털 입력 모듈의 입력 상태가 바뀌고 (예: OFF → ON) 표준 입력 필터 시간보다 짧은 시간 안에 다시 원래 상태로 복귀하게 되면 (ON → OFF) 입력 신호가 바뀌지 않은 것으로 간주합니다. 이것은 전기적 노이즈 입력에 의한 오동작을 방지하기 위한 기능으로 전기적 노이즈가 많은 현장에서는 표준 입력 필터 시간을 길게 설정함으로써 전기적 노이즈 입력에 의한 오동작을 방지할 수 있습니다.

여기서 설정한 입력 필터 시간은 PLC 시스템 전체에 적용되는 필터 시간이며, I/O 파라미터에서 디지털 입력 모듈 별로 입력 필터 시간을 지정할 수도 있습니다.

\* 출력 제어 설정

- 디버깅 중 출력 내기: PLC를 디버깅 모드로 운전할 때 출력 모듈에 출력을 할지 여부를 택합니다.
  - 에러 발생 시 출력 유지: PLC가 에러 상태가 되면 PLC는 연산을 수행할 수 없으므로 안전을 고려하여 디지털 출력을 리셋 시킵니다. 에러 발생 시 출력 유지를 선택(체크)하면, 디지털 출력 모듈의 I/O 파라미터에서 비상 출력에 홀드로 선택된 채널에 대해 출력을 리셋 시키지 않고 유지 시킵니다.
  - 런 → 스톱 전환 시 출력 유지: 에러 상태와 마찬가지로 PLC가 STOP 모드가 되었을 때 PLC는 연산을 하지 않으므로 디지털 출력을 리셋 시킵니다. 런 → 스톱 전환 시 출력 유지를 선택(체크)하면, 디지털 출력 모듈의 I/O 파라미터에서 비상 출력에 홀드로 선택된 채널에 대해 출력을 리셋 시키지 않고 유지 시킵니다.
  - 스톱 → 런 전환 시 출력 유지: PLC가 STOP 모드에서 RUN 모드로 변경될 때 래치 영역으로 설정된 데이터 메모리 영역이 외의 데이터를 클리어 시킵니다. 스톱 → 런 전환 시 출력 유지를 선택(체크)하면, 출력 데이터 메모리 영역을 클리어하지 않고 STOP 상태의 데이터를 가지고 RUN 모드로 진입합니다.
  - 에러 발생 시 래치 이외 영역 지우기: PLC가 에러 상태가 될 때 모든 데이터를 유지한 상태에서 프로그램의 연산을 실행하지 않습니다. 그러나 PLC가 통신 기능을 사용하고 있고 통신 기능 중 서버 기능 및 고속링크 기능을 사용하고 있다면 PLC가 에러 상태에서도 통신 기능은 계속 수행되어 PLC 데이터 메모리에 저장되어 있는 데이터를 송신하게 됩니다. 에러 발생 시 래치 이외 영역 지우기 기능을 선택(체크)하면 PLC가 에러 상태로 전환될 때 래치 영역 이외의 데이터를 클리어 합니다.
- \* '에러 발생 시 출력 유지', '런 → 스톱 전환 시 출력 유지'를 선택(체크) 하더라도, 출력 모듈의 I/O 파라미터에서 홀드를 선택하지 않으면 출력은 클리어 됩니다.

SOE 이력: SOE (Sequence of Events) 모듈은 이벤트가 발생한 순서와 시간을 기록하는 모듈입니다. 일반 디지털 입력과는 달리 PLC 연산 중에 발생한 이벤트도 인식하고 기록할 수 있습니다. 한 개의 SOE 모듈은 최대 300 개의 이벤트를 기록할 수 있는데, SOE 모듈의 메모리가 소진되었을 경우 SOE 모듈의 운전 상태를 설정합니다.

- 최근 이력으로 덮어쓰기: SOE 모듈의 메모리가 소진된 이후 이벤트가 발생했을 때 과거 데이터를 삭제한 후 최근 이벤트를 계속 기록합니다.
- 최초 이력 유지: SOE 모듈의 메모리가 소진된 이후 이벤트가 발생하더라도 더 이상 기록하지 않고 과거의 데이터를 그대로 유지합니다.

Reset 스위치 동작 차단 설정: CPU의 Reset 스위치의 동작을 설정합니다.

- Reset 스위치 동작 차단이 선택(체크)되어 있으면 CPU의 Reset 스위치를 동작시키더라도 CPU가 Reset 되지 않습니다. Reset 스위치를 3 초 이상 ON 시키더라도 Overall Reset 기능을 수행하지 않습니다.
- Overall Reset 스위치 동작 차단: Overall Reset이란 CPU의 Reset 스위치를 3 초 이상 ON 시키면 CPU가 Reset 되면서 래치 영역의 데이터를 삭제시키는 기능입니다. Overall Reset 스위치 동작 차단이 선택(체크)되어 있으면 CPU의 Reset 스위치를 3 초 이상 ON 시키더라도 래치 영역의 데이터가 삭제되지 않습니다.

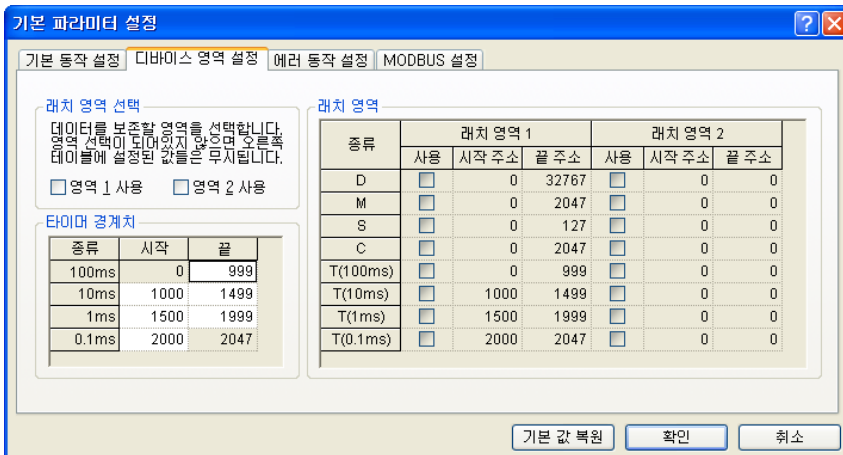
D.CLR 스위치 동작 차단 설정: CPU가 STOP된 상태에서 CPU의 D.CLR 스위치를 ON시키면 래치 1 영역(K, R 영역 포함)의 데이터가 삭제되며 (Data Clear), D.CLR 스위치를 3초 이상 ON시키면 래치 2 영역의 데이터까지 삭제됩니다.(Overall Data Clear)

- D.CLR 스위치 동작 차단이 선택(체크)되어 있으면 CPU가 STOP 상태에서 D.CLR 스위치를 ON시켜도 래치 1 영역의 데이터를 삭제하지 않으며, D.CLR 스위치를 3초 이상 ON시켜도 래치 2 영역의 데이터를 삭제하지 않습니다.

- Overall D.CLR 스위치 동작 차단이 선택(체크)되어 있으면 CPU가 STOP되어 있는 상태에서 D.CLR 스위치를 3초 이상 ON시켜도 래치 2 영역의 데이터를 삭제하지 않습니다.

\* Reset 스위치 또는 D.CLR 스위치의 동작을 차단하더라도 XG5000 온라인 메뉴의 Reset 및 데이터 클리어 기능은 동작합니다.

2) 디바이스 영역 설정: 래치 영역을 설정합니다. 래치 영역이란 PLC가 리셋 되었을 때 데이터를 클리어하지 않고 유지하는 영역을 말합니다. XGK 계열 PLC에서 래치 영역은 래치 1 영역과 래치 2 영역으로 나누어지며 각 영역은 리셋 기능 및 데이터 클리어 기능에 따라 데이터를 유지 또는 클리어 합니다. 그리고 데이터 메모리 영역 중 K 영역과 R 영역은 디바이스 영역 설정에서 설정하지 않더라도 래치 기능을 수행합니다.



• 래치 영역

	K, R영역	래치 1	래치 2
<b>CPU STOP/RUN</b>	유지	유지	유지
<b>전원 OFF/ON</b>	유지	유지	유지
<b>Reset</b>	유지	유지	유지
<b>Overall Reset</b>	클리어	클리어	유지
<b>래치 1 클리어</b>	클리어	클리어	유지
<b>래치 2 클리어</b>	클리어	클리어	클리어
<b>프로그램 쓰기</b>	유지	유지	유지

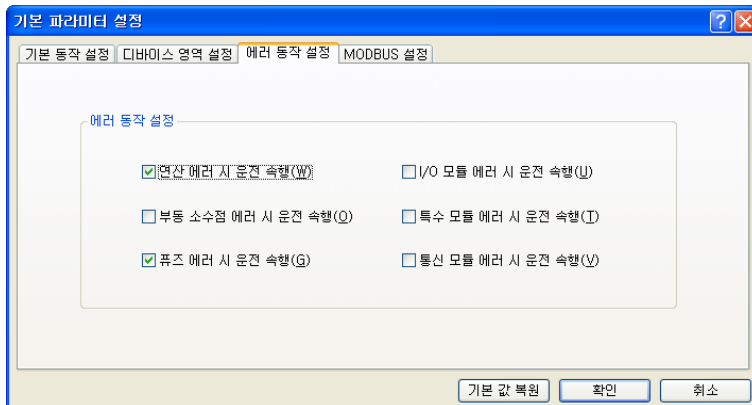
타이머 경계치: XGK PLC에는 T0000 ~ T2047 까지 2048 개의 타이머가 있으며, 타이머 설정 시간으로

100ms, 10ms, 1ms, 0.1ms 단위로 설정할 수 있습니다. 여기에서 설정 시간 별 타이머의 수를 조절하여 사용할 수 있습니다.

래치 영역 설정 방법

- 래치 영역 선택에서 영역 1 사용, 영역 2 사용을 선택(체크)하면 래치 영역에서 각 래치 영역별 '사용' 선택 행이 활성화 됩니다.
- 래치 영역으로 사용하고자 하는 메모리 영역의 '사용'을 선택(체크)하면 시작 주소 끝 주소를 입력할 수 있습니다.
- D 영역과 M 영역에 대해서는 워드 단위, S 영역(스텝 콘트롤러)에 대해서는 조 단위, C(카운터) 영역과 T(타이머) 영역에 대해서는 카운터 또는 타이머 번호 단위로, 연속적으로 설정할 수 있습니다.
- 래치 영역 1 과 래치 영역 2 가 정상적으로 동작하기 위해서는 메모리 영역이 중복되지 않게 주소를 설정해야 합니다.
- 래치 영역 1 과 래치 영역 2 에 중복으로 등록된 메모리 영역은 래치 영역 1 로 동작합니다.

3) 에러 동작 설정: PLC 에 에러가 발생했을 때 운전을 정지 할 것인지 에러를 무시하고 계속 운전을 할 지 여부를 선택합니다. 여기서 운전 속행으로 설정된 에러가 발생했을 때 PLC 는 해당하는 에러가 발생했을 때 운전을 계속하고, CPU 의 CHK LED 가 점멸하게 됩니다.



I/O 파라미터

I/O 파라미터는 PLC 에 장착된 디지털 입력 모듈, 디지털 출력 모듈, 특수 모듈의 운전 속성을 지정합니다. 단, 특수 모듈 중 위치 제어 모듈 (XG-PM)과 온도 제어 모듈(XG-TCO)N 은 별도의 소프트웨어 툴을 이용하여 파라미터를 설정합니다.

디지털 입력 모듈의 I/O 파라미터 내용은 입력 필터 시간을 설정하는 것이며, 별도로 I/O 파라미터를 설정하지 않을 경우 기본 파라미터에서 설정한 표준 입력 필터 시간으로 동작합니다.

디지털 출력 모듈의 I/O 파라미터 내용은 채널 별(8 점 단위)로 PLC 가 STOP 또는 에러 상태로 전환될 때 출력을 제어하는 비상 출력에 관한 내용으로 기본 설정은 출력을 클리어 시키는 것으로 설정되어 있습니다. 만일 I/O 파라미터에서 비상 출력을 유지(홀드)로 설정하더라도 기본 파라미터에서 '에러 발생

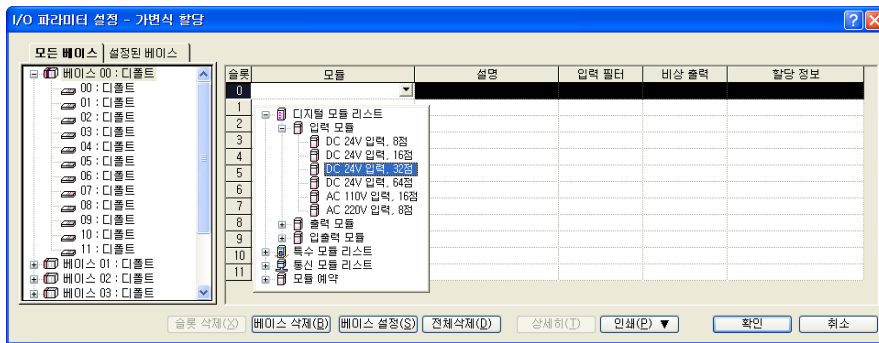
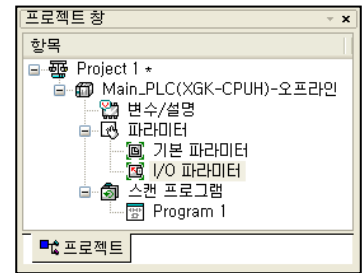
시 출력 유지' '런 → 스톱 시 출력 유지'를 선택하지 않으면 비상 시 출력을 유지되지 않습니다.

모듈 등록: I/O 파라미터를 설정하기 위해서는 I/O 파라미터에 모듈을 등록해야 합니다. 모듈을 등록하는 방법은 PLC 를 연결하여 PLC 에 장착되어 있는 모듈을 읽어 등록하는 방법 (온라인 등록)과 사용자가 모듈을 선택하여 등록하는 방법(사용자 등록)이 있습니다.

① 사용자 등록: PLC 각 슬롯 별로 장착할 모듈을 사용자가 선택하여 등록합니다.

사용자 모듈 등록 순서

- 프로젝트 창에서 I/O 파라미터를 더블 클릭합니다.
- I/O 파라미터 설정 창에서 슬롯을 클릭하면 모듈 종류가 표시됩니다.
- 모듈 종류를 확장시켜 장착할 모듈 종류를 선택하면 선택된 슬롯에 선택한 모듈이 등록됩니다.

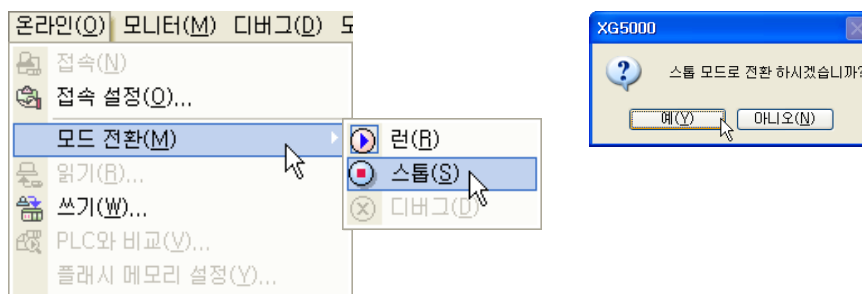


② 온라인 등록: PLC 시스템이 구성되어 있을 때 PLC 에 장착된 모듈의 종류를 읽어 I/O 파라미터에 등록합니다. 온라인 등록은 PLC 와 XG5000 이 접속되어 있을 때 사용 가능합니다.

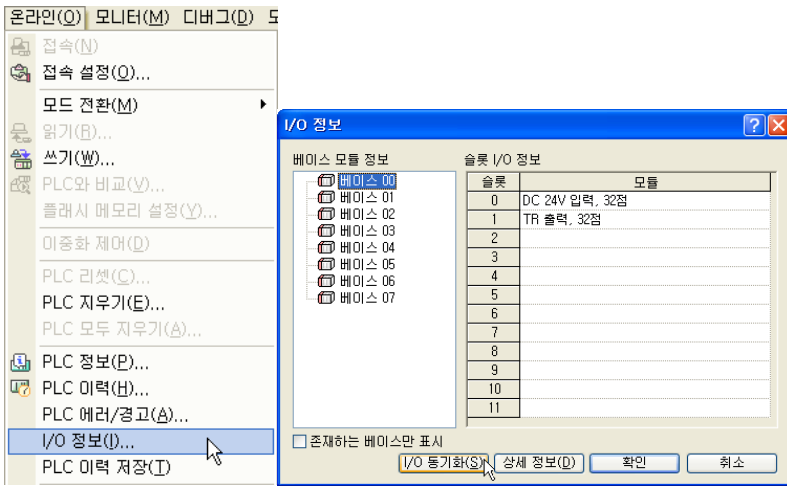
온라인 모듈 등록 순서

- PLC STOP 모드 전환: PLC 를 접속한 후 온라인

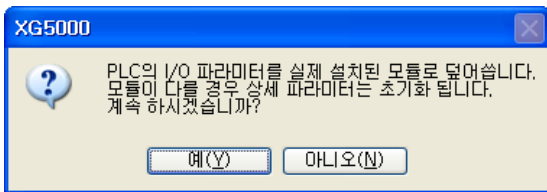
메뉴 >> 모드 전환 >> 스톱을 눌러 PLC 를 STOP 모드로 전환합니다.



- 온라인 메뉴 >> I/O 정보(I)를 선택하여 PLC 에 장착된 모듈의 I/O 정보를 읽어 옵니다. I/O 정보 창에서 I/O 동기화 버튼을 클릭하면 PLC 에서 읽어온 I/O 정보를 I/O 파라미터로 저장합니다.
- 모듈을 선택한 후 I/O 정보 창에서 상세 정보를 선택하면 모듈의 상세 정보를 확인할 수 있습니다.
- PLC 가 STOP 모드가 아닐 경우 I/O 정보를 읽어 오지만 'I/O 동기화(S)' 버튼이 활성화 되지 않아 I/O 동기화를 할 수 없습니다.

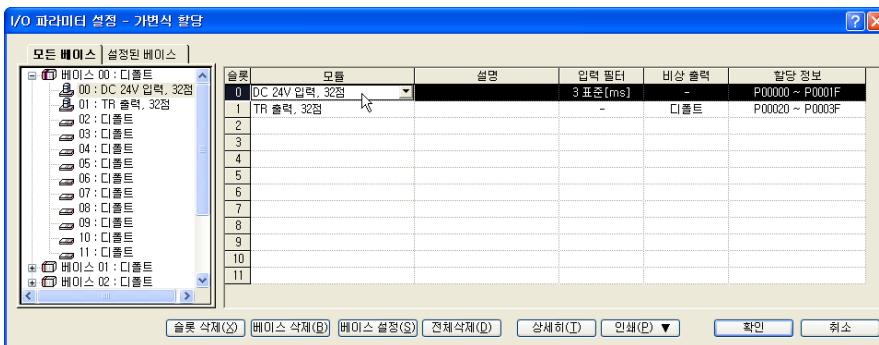


- I/O 동기화를 실행하면 PLC 로부터 읽어온 모듈 정보를 XG5000 프로젝트와 PLC 내부의 I/O 파라미터에 등록합니다.
- 만일, I/O 파라미터가 설정되어 있을 경우 모든 I/O 파라미터가 초기화 됩니다.
- PLC 에 저장되어 있는 I/O 파라미터를 확인하고자 할 경우 'PLC 로부터 열기'를 실행하여 I/O 파라미터 읽기를 실행해야 합니다.

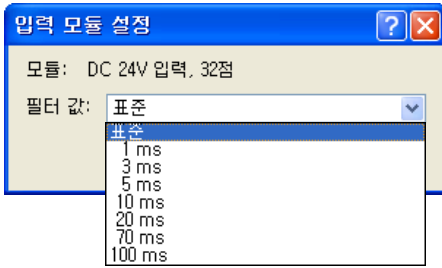


**I/O 파라미터 설정**

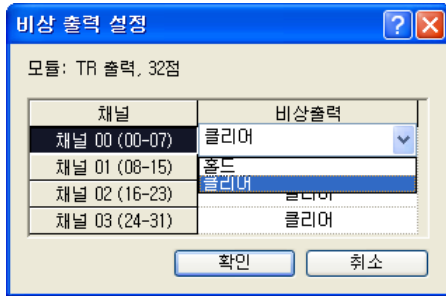
- I/O 파라미터설정 창에서 설정하고자 하는 모듈을 더블 클릭합니다.



- 디지털 입력 모듈의 경우 모듈 별로 입력 필터 시간을 설정할 수 있습니다. I/O 파라미터에서 입력 모듈에 대해 입력 필터 시간을 지정하지 않으면 기본 파라미터에서 설정한 '표준 입력 필터' 시간으로 입력 필터 시간이 설정됩니다.



- 디지털 출력 모듈의 경우 채널 별로 비상 출력을 홀드(유지) 또는 클리어를 선택할 수 있습니다.



\* 디지털 출력 모듈의 I/O 파라미터에서 비상 출력을 홀드로 선택하더라도 기본 파라미터에서 '에러 시 출력 유지', '런 → 스톱 시 출력 유지'를 선택하지 않으면 비상 출력이 유지되지 않습니다.

## 4 장. 데이터와 프로그램

### 4.1 PLC 데이터

PLC 에서 다루는 데이터는 수치 데이터와 문자 데이터가 있습니다. 수치 데이터는 다시 정수형 수치 데이터와 실수형 수치 데이터가 있으며, 문자 데이터는 8 bit 문자 데이터와 16 비트 문자 데이터가 있습니다. 여기에서 각 데이터의 표현 및 해석 방법을 설명합니다.

#### 1) 수치 데이터의 종류 및 범위

수치 데이터란 숫자로 표시되는 데이터를 의미하며, 진법에 따라 이진수, 8 진수, 10 진수, 16 진수의 형태로 PLC 에서 사용됩니다. 또, PLC 에서 사용하는 수치 데이터는 정수형 수치 데이터와 실수형 수치 데이터로 구분될 수 있으며, 정수형 수치 데이터는 한 개의 데이터가 점유하는 비트의 수에 따라 니블(Nibble, 4 bit), 바이트(Byte, 8 bit), 워드(Word, 16 bit), 더블 워드(Double Word, 32 bit) 크기가 있으며, 실수형 수치 데이터는 플롯(Float, 32 bit)과 롱(Long, 64 bit)의 크기가 있습니다.

종류	크기	데이터 범위		
		16 진수	부호없는 십진수	부호있는 십진수
정수형	니블(4 bit)	h0 ~ hF	0 ~ 15 (0 ~ (2 <sup>4</sup> - 1))	-8 ~ 7 (-2 <sup>3</sup> ~ (2 <sup>3</sup> - 1))
	바이트(8 bit)	h00 ~ hFF	0 ~ 127(0 ~ (2 <sup>8</sup> - 1))	-64 ~ 63 (-2 <sup>7</sup> ~ (2 <sup>2</sup> - 1))
	워드(16 bit)	h0000 ~ hFFFF	0 ~ 65,535 (0 ~ (2 <sup>16</sup> - 1))	-32,768 ~ 32,767 (-2 <sup>15</sup> ~ (2 <sup>15</sup> - 1))
	더블워드(32 bit)	h00000000~hFFFFFFF	0 ~ 4,294,967,295 (0 ~ (2 <sup>32</sup> - 1))	-2,147,483,648 ~ 2,147,483,647 (-2 <sup>31</sup> ~ (2 <sup>31</sup> - 1))
실수형	Float(32 bit)	-3.402823466e+038 ~ -1.175494351e-038, 0, 1.175494351e-038 ~ 3.402823466e+038		
	Long(64 Bit)	-1.7976931348623157e+308 ~ -2.2250738585072014e-308, 0, 2.2250738585072014e-308 ~ 1.7976931348623157e+308		

#### 2) 수치 데이터의 표현 및 해석

수치 데이터를 PLC 에 저장할 때 지정된 메모리 주소에서 0 번 비트부터 아래와 같은 방식으로 데이터를 메모리에 저장하며, 메모리에 저장되어 있는 데이터를 읽어 올 경우도 동일한 방법으로 해석하여 읽어옵니다.

**16 진수(Hexa-Decimal)의 표현:** 16 진수 1 자리가 가질 수 있는 데이터는 0 ~9, A ~ F 까지 16 개 데이터를 가질 수 있으며, 16 개 데이터를 표시하기 위해서는 4 개의 비트가 필요합니다. (2<sup>4</sup> = 16) 따라서 1 개의 워드에 저장할 수 있는 16 진수 최대 데이터는 hFFFF 가 됩니다. 아래의 표는 16 진수를 메모리에 저장할 때 각 비트 별 가수가 됩니다. 여기서 가수란 해당 비트가 ON 되었을 때 해석되는 수치입니다.

2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
X 1000 <sub>(h)</sub>				X 100 <sub>(h)</sub>				X 10 <sub>(h)</sub>				X 1 <sub>(h)</sub>			

예를 들어 16 진수 h129E 를 1 워드 메모리에 저장할 다음과 같이 해석되어 각 비트 별로 저장됩니다.

$$h129E = 1 \times 1000_{(h)} + 2 \times 100_{(h)} + (8 + 1) \times 10_{(h)} + (8 + 4 + 2) \times 1_{(h)}$$

8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
X 1000 <sub>(h)</sub>				X 100 <sub>(h)</sub>				X 10 <sub>(h)</sub>				X 1 <sub>(h)</sub>			

반대로 1 워드 크기의 메모리에 다음과 같이 bit 가 ON 되어 있고 16 진수 수치로 해석한다면 아래의 설명과 같이 해석됩니다.

8	4	2		8	4			8	4	2	1	8		2	1
X 1000 <sub>(h)</sub>				X 100 <sub>(h)</sub>				X 10 <sub>(h)</sub>				X 1 <sub>(h)</sub>			

$$(8 + 4 + 2) \times 1000_{(h)} + (8 + 4) \times 100_{(h)} + (8 + 4 + 2 + 1) \times 10_{(h)} + (8 + 2 + 1) \times 1_{(h)} = ECFB_{(h)} \Rightarrow hECFB$$

**부호 없는 십진수(Undsigned Integer)의 표현:** 비트 배열을 부호 없는 십진수로 해석하는 방법은 각 자리 수 별 가수를 사용하는 16 진수와 달리 메모리의 0 번 비트부터 2<sup>n</sup> 가수를 사용합니다. 예를 들어, 16 비트 크기의 메모리를 사용하는 워드의 경우 각 비트별로 2<sup>0</sup> 부터 2<sup>15</sup> 까지 가수를 가지며, 수치의 계산은 ON 비트 가수의 합으로 계산됩니다. 만일, 0 번 비트와 15 번 비트가 가 ON 되어 있다면 2<sup>0</sup> = 1 로 해석되고, 2<sup>15</sup> = 32,768 로 해석되어 전체 수치는 1 + 32,768 = 32,769 로 해석됩니다.

2 <sup>15</sup>	2 <sup>14</sup>	2 <sup>13</sup>	2 <sup>12</sup>	2 <sup>11</sup>	2 <sup>10</sup>	2 <sup>9</sup>	2 <sup>8</sup>	2 <sup>7</sup>	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>
-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------

예를 들어, 1234 를 1 워드 메모리에 부호 없는 십진수로 저장할 때 다음과 같이 해석되어 각 비트 별로 저장됩니다.

$$1234 = 1024 + 128 + 64 + 16 + 2 = 2^{10} + 2^7 + 2^6 + 2^4 + 2^1$$

2 <sup>15</sup>	2 <sup>14</sup>	2 <sup>13</sup>	2 <sup>12</sup>	2 <sup>11</sup>	2 <sup>10</sup>	2 <sup>9</sup>	2 <sup>8</sup>	2 <sup>7</sup>	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>
-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------

또, 1 워드 크기의 메모리에 다음과 같이 bit 가 ON 되어 있을 때 부호 없는 십진수로 해석하면 아래의 설명과 같이 해석됩니다.

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

$$2^{15} + 2^{14} + 2^{13} + 2^{11} + 2^{10} + 2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^1 + 2^0 = 32,768 + 16,384 + 8,192 + 2,048 + 1,024 + 128 + 64 + 32 + 16 + 8 + 2 + 1 = 60,667$$

**부호 있는 십진수(Signed Integer)의 표현:** 비트 배열을 부호 있는 십진수로 해석하는 방법은 최상위 비트는 부호 비트가 되며 0 번 비트부터 최상위 -1 번 비트까지가 가수가 됩니다. 부호 비트가 OFF 상태(0) 일 때 전체 수치는 0 또는 양수임을 의미하고, 수치는 부호 없는 정수와 마찬가지로 ON 비트 가수의 합으로 계산되며, 부호 비트가 ON 상태(1) 일 때 전체 수치는 음수임을 의미하고, 수치는 OFF 비트 가수의 합 + 1(2의 보수)이 됩니다.

부호	2 <sup>14</sup>	2 <sup>13</sup>	2 <sup>12</sup>	2 <sup>11</sup>	2 <sup>10</sup>	2 <sup>9</sup>	2 <sup>8</sup>	2 <sup>7</sup>	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>
----	-----------------	-----------------	-----------------	-----------------	-----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------

예를 들어 -1234 를 부호 있는 십진수로 메모리에 저장할 때 다음과 같이 해석되어 각 비트 별로 저장 됩니다.

$$-1234 = -(1233 + 1) = -(1024 + 128 + 64 + 16 + 1 + 1) = -(2^{10} + 2^7 + 2^6 + 2^4 + 2^0 + 1)$$

부호	2 <sup>14</sup>	2 <sup>13</sup>	2 <sup>12</sup>	2 <sup>11</sup>	2 <sup>10</sup>	2 <sup>9</sup>	2 <sup>8</sup>	2 <sup>7</sup>	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>
----	-----------------	-----------------	-----------------	-----------------	-----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------

또, 1 워드 크기의 메모리에 다음과 같이 bit 가 ON 되어 있을 때 부호 있는 십진수로 해석하면 아래의 설명과 같이 해석됩니다.

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

$$-(2^{12} + 2^9 + 2^8 + 2^2 + 1) = -(4,096 + 512 + 256 + 4 + 1) = -4,869$$

**실수형 데이터:** XGK PLC 에서 사용할 수 있는 실수형 데이터는 32 비트 실수형(REAL)과 64 비트 실수형(LREAL)이 있습니다. 32 비트 실수형은 최상위 비트(31 번 비트)가 부호 비트이며, 30 ~ 23 번 비트가 지수부, 22 ~ 0 번 비트가 소수부가 되고, 64 비트 실수형은 최상위 비트(63 번 비트)가 부호 비트이며, 62 ~ 52 번 비트가 지수부, 51 ~ 0 번 비트가 소수부가 됩니다.

**- 32 비트 실수 표현의 예**

$$2.5 = 2 \times 1.25 = +1 \times 2^{(128 - 127)} \times (1 + 0.25) = +1 \times 2^{(128 - 127)} \times (1 + 2^{-2})$$

128 을 8 비트로 표시하면: 10000000

2<sup>-2</sup>을 비트로 표시하면 0.010

따라서 2.5 를 32 비트 실수로 표현하면

0 10000000 0100000000000000000000 이 되며 16 진수로 표현하면 h40200000 이 된다.

**- 32 비트 실수 해석의 예**

0 10000001 1010000000000000000000

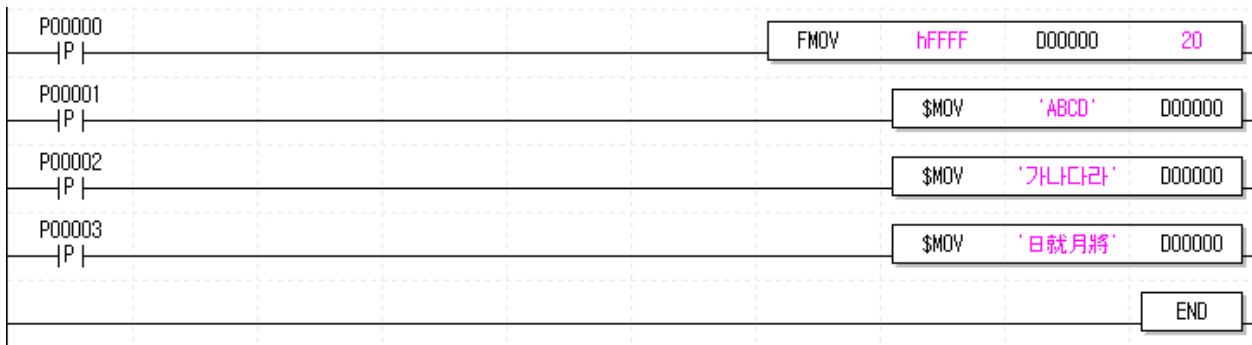
$$+1 \times 2^{(129 - 127)} \times \{1 + (2^{-1} + 2^{-2})\} = 1 \times 4 \times 1.625 = 6.5$$

3) 문자열 데이터

XGK PLC 에서 처리할 수 있는 문자열 데이터는 영문자, 숫자, 특수 문자 외 윈도우에서 지원하는 언어입니다. 즉 한글 윈도우를 사용할 경우 한글이 사용 가능하며, 한글 윈도우에서 지원하는 한자도 사용 가능합니다. 키보드에서 기본적으로 지원하는 영문자, 숫자 또는 특수 문자의 경우 1 개의 글자는 1Byte 를 점유하며, 한글, 한자의 경우 1 개 글자가 2Byte 를 점유합니다. XGK 에서 문자열 처리 명령어는 최대 32 Byte 의 문자열 데이터를 처리할 수 있으므로, 영문자, 숫자, 특수 문자로 구성된 문자열의 경우 최대 32 개의 문자를 처리할 수 있으며, 한글, 한자의 경우 16 개의 문자까지 처리할 수 있습니다.

PLC 에 문자 데이터를 저장할 경우 영문자, 숫자, 특수 문자의 경우 아스키 코드로 변환되어 저장 영역의 선두 바이트부터 저장되며, 다른 언어 문자의 경우 해당 언어 윈도우에서 표준으로 사용하는 코드로 변환되어 1 개의 문자가 2 Byte 에 저장됩니다. 예를 들어 영문 대문자 'A'를 XGK 데이터 메모리 D00000 에 저장하면 영문 대문자 'A'에 대한 아스키 코드 (h41)을 저장 데이터 메모리의 선두 바이트에 해당하는 D00000 워드의 하위 바이트(0 ~ 7 번 비트)에 저장하고, 한글 '가'를 D00000 에 저장하면 '가'에 대한 KSC5601 코드 (hA1B0)를 D0000 1 개 워드에 저장합니다. 여기서 주의할 점은 1 개의 문자를 PLC 에 저장하더라도 32 Byte 메모리를 사용합니다. 즉, 영문 대문자 'A'를 D00000 에 저장할 경우 'A'는 D00000 의 하위 바이트에 저장되고, D00000 의 상위 바이트(8 ~ F 번 비트)부터 D00015 의 상위 바이트까지 31 개의 바이트는 모두 NULL(h00)로 채워집니다.

XGK PLC 에서 문자열 데이터를 처리할 경우 연산의 접두어로 '\$'를 사용하며 문자열 상수는 작은 따옴표 ( ' ' )로 표시합니다. 아래의 프로그램 예는 각 비트가 ON 되면 영문자 'ABCD', 한글 '가나다라' 한자 '日就月將'을 D00000 부터 저장합니다. 특히, P00000 을 ON 시킨 D00000 부터 16 워드 이상의 데이터 메모리를 모니터하면서 문자열 데이터를 저장하는 조건을 ON 시키면 데이터가 저장되지 않는 영역은 0 으로 클리어 되는 것을 확인할 수 있습니다.



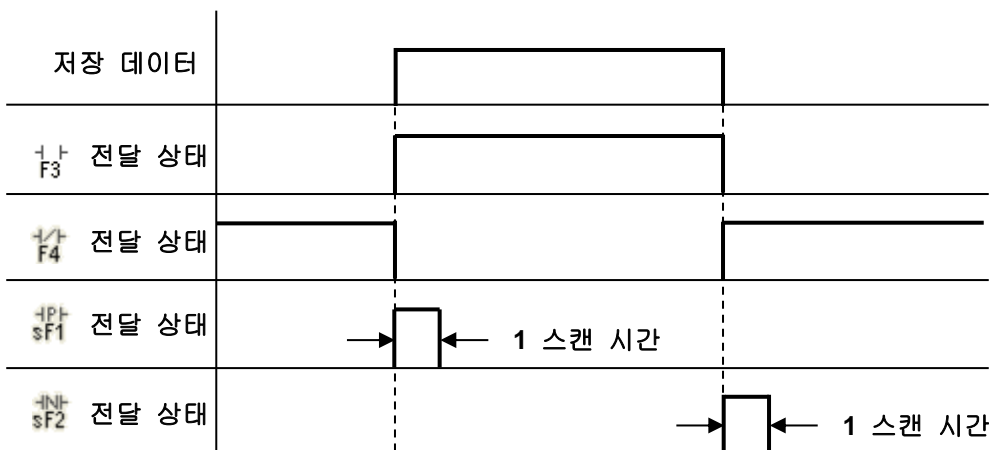
### 4.2 접점/코일 프로그램

시퀀스 프로그램이란 디지털 입력 모듈을 통해 입력되는 센서, 스위치 등 비트 신호의 ON 또는 OFF 상태에 따라 출력 모듈의 접점을 ON 또는 OFF 시키는 제어를 의미합니다. 시퀀스 프로그램을 작성하기 위해서는 2장에서 설명한 입력 신호 및 출력 신호의 메모리 주소를 정확히 이해해야 합니다. 그리고 시퀀스 프로그램에 사용하는 시퀀스 기호에 대해서도 정확한 이해가 필요합니다.

#### 1) 접점

접점이란 데이터 메모리에 저장되어 있는 비트의 상태 정보를 읽어 데이터를 읽어 그 상태를 좌측에서 우측으로 전달하는 프로그래밍 기호입니다. XGK PLC 프로그래밍에서 사용하는 접점의 종류는 평상시 열린 접점(a 접점), 평상시 닫힌 접점(b 접점), 양변환 검출 접점(P 접점), 음변환 검출 접점(N 접점)이 있으며, 각 접점의 동작 특성은 다음과 같습니다.

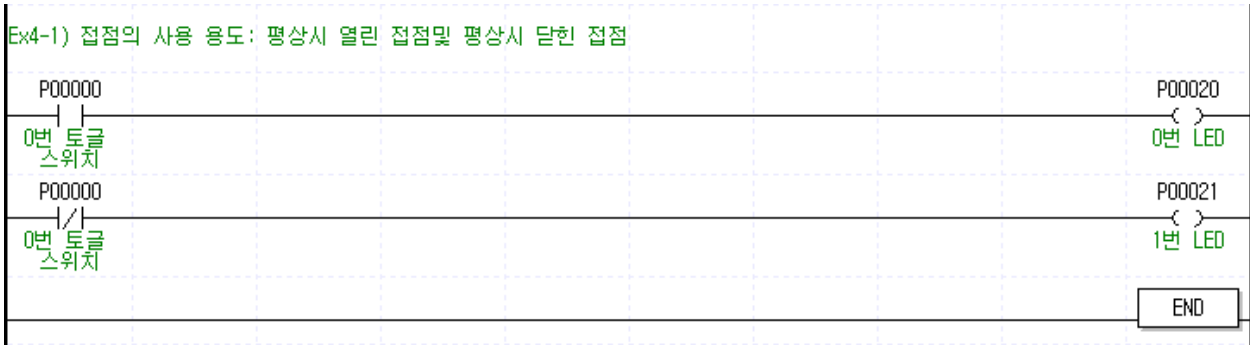
- 평상시 열린 접점(  $\frac{1}{F3}$  ): 데이터 메모리에 저장된 비트 데이터를 읽어 그 상태를 우측으로 전달합니다.
- 평상시 닫힌 접점(  $\frac{1}{F4}$  ): 데이터 메모리에 저장된 비트 데이터를 읽어 상태를 반전시킨 후 그 상태를 우측으로 전달합니다.
- 양변환 검출 접점(  $\frac{1}{SF1}$  ): 지정된 비트가 OFF 에서 ON 으로 변경될 때 1 스캔 시간 동안 ON 상태를 우측으로 전달합니다.
- 음변환 검출 접점(  $\frac{1}{SF2}$  ): 지정된 비트가 ON 에서 OFF 로 변경될 때 1 스캔 시간 동안 ON 상태를 우측으로 전달합니다.



- 평상시 열린 접점: 평상시 열린 접점은 비트 메모리의 상태 정보를 그대로 좌측에서 우측으로 전달하는 접점입니다. 평상시 열린 접점에 정논리의 입력 접점 신호를 사용할 경우 '입력 신호가 ON 일 때'의 의미를 가지며, 부논리의 입력 접점 신호를 사용할 경우 '입력 신호가 OFF 일 때'의 의미를 가집니다.

- 평상시 닫힌 접점: 평상시 닫힌 접점은 비트 메모리의 상태 정보를 반전하여 좌측에서 우측으로 전달하는 접점입니다. 평상시 닫힌 접점에 정논리의 입력 접점 신호를 사용할 경우 '입력 신호가 OFF 일 때의 의미를 가지며, 부논리의 입력 접점 신호를 사용할 경우 '입력 신호가 OFF 일 때'의 의미를 가집니다.

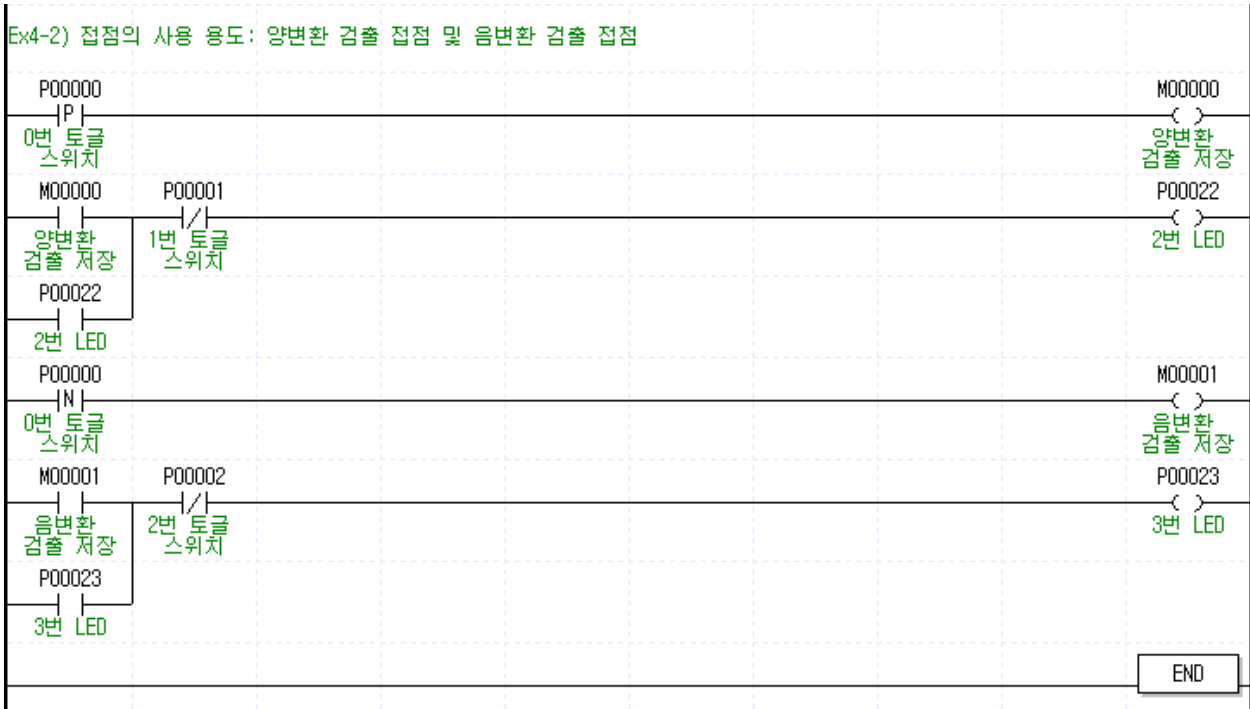
Ex 4-1) 다음의 프로그램을 작성하여 PLC 로 전송한 후 P00000 토글 스위치를 ON/OFF 시키면서 P00020, P00021 LED 의 상태를 확인하고, 평상시 열린 접점과 평상시 닫힌 접점의 기능을 확인하십시오.



- 양변환 검출 접점: 양변환 검출 접점은 비트 메모리의 상태 정보가 OFF 에서 ON 으로 변화할 때 1 스캔 시간 동안만 ON 상태를 좌측에서 우측으로 전달하는 접점입니다. 양변환 검출 접점에 정논리의 입력 접점 신호를 사용할 경우 '입력 신호가 ON 될 때'의 의미를 가지며, 부논리의 입력 접점 신호를 사용할 경우 '입력 신호가 OFF 될 때'의 의미를 가집니다.

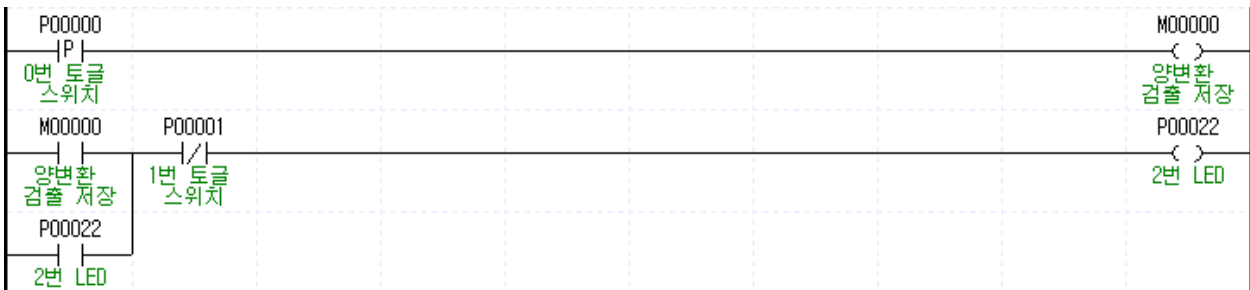
- 음변환 검출 접점: 음변환 검출 접점은 비트 메모리의 상태 정보가 ON 에서 OFF 로 변화할 때 1 스캔 시간 동안만 ON 상태를 유지합니다.

Ex 4-2) 다음의 프로그램을 작성하여 PLC 로 전송한 후 P00000, P00001, P00002 토글 스위치를 ON /OFF 시키면서 P00022, P00023 LED 의 상태를 확인하고, 양변환 검출 접점과 음변환 검출 접점의 기능을 확인하십시오.

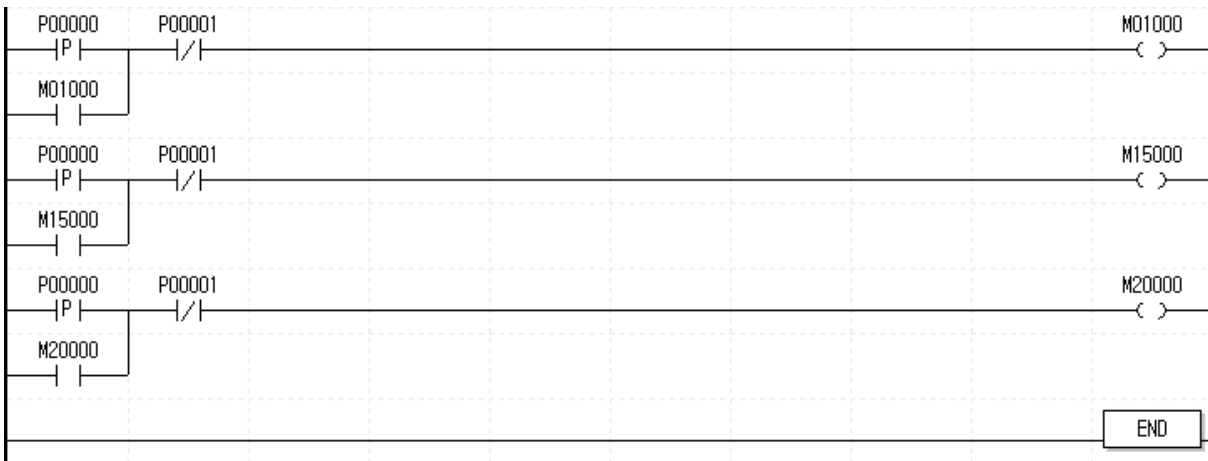
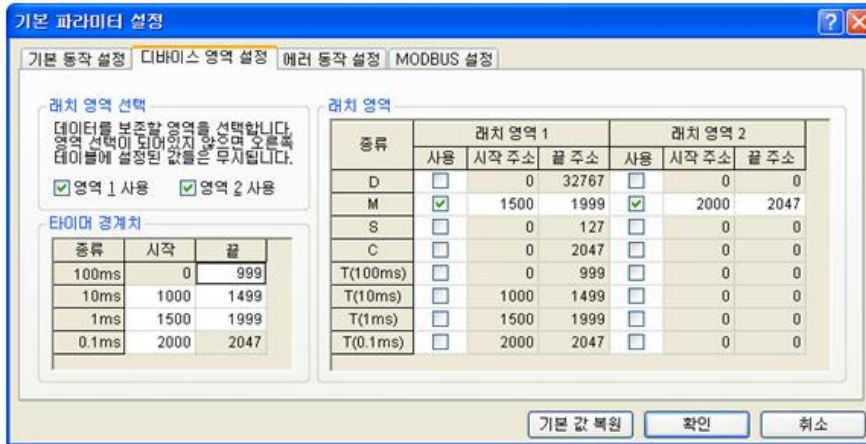


- 자기 유지 회로: 아래의 그림과 같이 출력 코일의 주소를 다시 입력으로 사용하는 회로를 자기 유지 회로라고 합니다. 자기 유지 회로는 처음의 조건이 만족되었을 때 출력 코일에 사용된 비트의 상태를 ON 시킨 후 사용자가 원하는 시점에 출력 코일에 사용된 비트의 상태를 OFF 시킬 때 사용됩니다.

위의 프로그램에서 P00000 접점이 ON 되면 M00000 비트가 ON 되고, P00001 이 OFF 되어 있으면 P00022 출력 접점이 ON 됩니다. 그 다음 스캔에 P00000 접점이 ON 상태를 유지하더라도 M00000 접점은 OFF 되지만, P00022 비트가 그 전 스캔에서 ON 되었으므로 P00022 접점의 ON 상태에 의해서 코일의 P00022 비트는 계속 ON 상태를 유지합니다. P00001 입력 접점이 ON 되면 접점 P00022 은 ON 상태를 좌측에서 우측으로 전달하지만, 평상시 닫힌 접점 P00001 에 의해 회로가 차단되므로, 코일의 P00022 비트는 OFF 됩니다. 즉, P00000 접점이 ON 되면 P00022 출력 접점이 ON 되어 그 상태를 계속 유지하다가 P00001 접점이 ON 되면 P00022 접점이 OFF 됩니다. 이와 같이 코일에 사용된 비트의 상태를 다시 입력으로 받아 그 상태를 계속 유지하는 회로를 자기 유지 회로라고 합니다.



- 자기 유지 회로 와 리테인: 자기 유지 회로에서 출력이 ON 된 상태에서 PLC 가 리셋된 후 재기동 할 때 출력 코일에 사용된 메모리 영역의 리테인 설정 여부에 따라 출력의 유지 상태가 달라집니다. 기본 파라미터에서 디바이스 영역 설정을 아래의 그림과 같이 설정하고 프로그램을 작성하여 PLC 로 전송하고, PLC 를 RUN 시킨 후 출력을 ON 시킨 상태에서 PLC 를 리셋, Overall 리셋 시킨 후 출력 상태를 확인 하십시오. 다시 출력을 On 시킨 후 PLC 를 STOP 시키고 D.CLR, Overall D.CLR 시킨 후 각 출력의 상태를 확인하십시오.



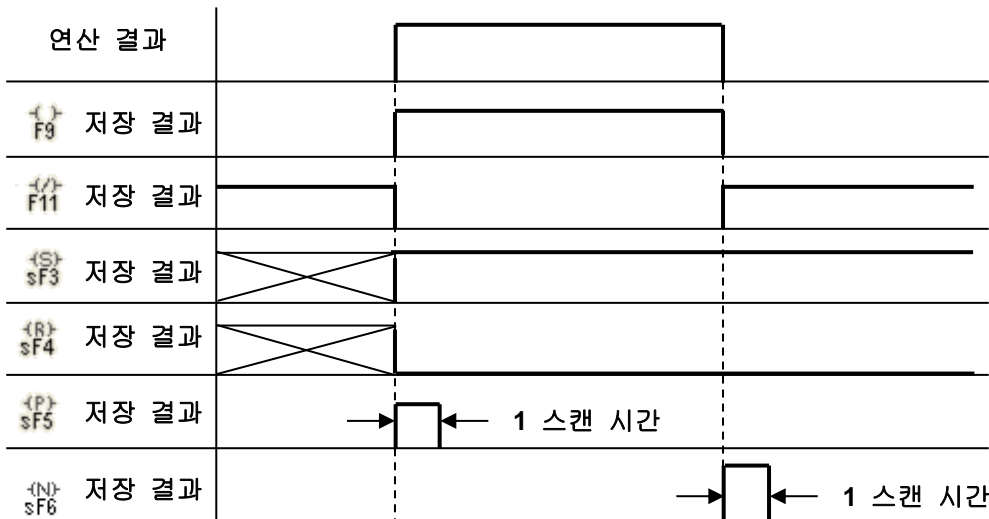
\* 파라미터는 런 중 수정이 불가능합니다. 파라미터를 수정했을 경우 PLC 를 STOP 시킨 후 PLC 로 전송 해야 합니다.

2) 코일

코일은 연산의 결과가 비트로 출력될 때, 출력된 연산 결과를 지정된 비트 메모리에 저장하는 프로그래밍 기호입니다.

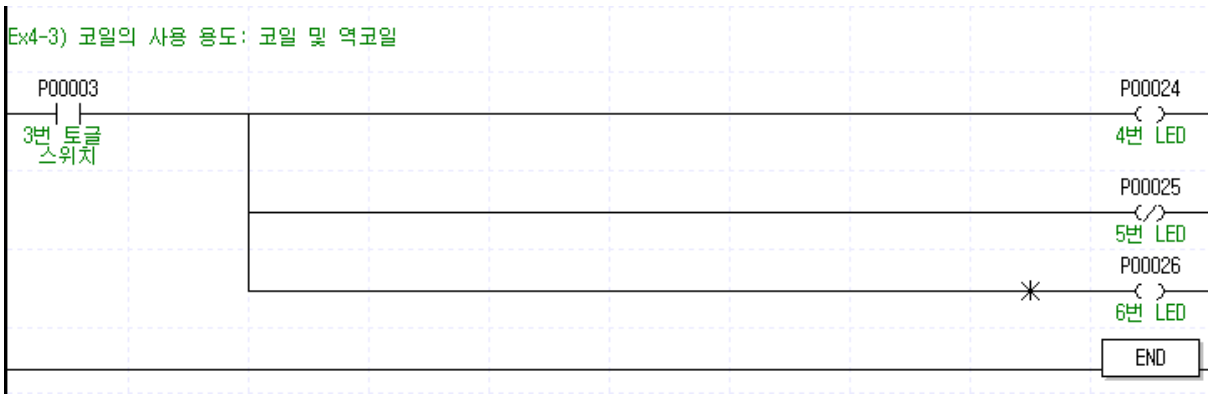
XGK PLC 프로그램에서 사용하는 코일의 종류는 코일, 역코일, 셋 코일, 리셋 코일, 양변환 검출 코일(P 코일), 음변환 검출 코일(N 코일)이 있으며, 각 코일의 동작 특성은 다음과 같습니다.

- 코일(  $\{F\}$  F9 ): 출력된 비트 결과를 지정된 메모리에 저장합니다.
- 역코일(  $\{F\}$  F11 ): 출력된 비트 결과를 반전하여 지정된 메모리에 저장합니다.
- 셋 코일(  $\{S\}$  sF3 ): 조건이 만족될 때 지정된 비트 메모리를 ON 시키고, 조건이 해제 되더라도 지정된 비트를 다시 OFF 시키지 않습니다. 지정된 비트 메모리가 ON 되어 있는 상태에서 조건이 만족될 때 지정된 비트에는 변화가 없습니다.
- 리셋 코일(  $\{R\}$  sF4 ): 조건이 만족될 때 지정된 비트 메모리를 OFF 시키고, 조건이 해제 되더라도 지정된 다시 ON 시키지 않습니다. 지정된 비트 메모리가 OFF 되어 있는 상태에서 조건이 만족될 때 지정된 비트에는 변화가 없습니다.
- 양변환 검출 코일(  $\{P\}$  sF5 ): 연산 결과가 OFF 에서 ON 으로 변경될 때 1 스캔 시간 동안 ON 상태를 지정된 비트에 저장합니다. 1 스캔 시간이 지난 후 지정된 비트는 OFF 됩니다.
- 음변환 검출 코일(  $\{N\}$  sF6 ): 연산 결과가 ON 에서 OFF 로 변경될 때 1 스캔 시간 동안 ON 상태를 지정된 비트에 저장합니다. 1 스캔 시간이 지난 후 지정된 비트는 OFF 됩니다.



- 코일: 코일은 논리 연산의 결과를 그대로 지정된 비트 메모리에 저장하는 역할을 합니다. 즉, 논리 연산의 결과가 1 이면 지정된 비트에 1 을 저장하고 (ON), 논리 연산의 결과가 0 이면 지정된 비트에 0 (OFF)를 저장합니다.
- 역코일: 역코일은 논리 연산의 결과를 반전하여 지정된 비트 메모리에 저장하는 역할을 합니다. 즉, 논리 연산의 결과가 1 이면 지정된 비트에 0 을 저장하고 (OFF), 논리 연산의 결과가 0 이면 지정된 비트에 1 (ON)을 저장합니다.

Ex 4-3) 다음의 프로그램을 작성하여 PLC 로 전송한 후 P00003 토글 스위치를 ON /OFF 시키면서 P00024, P00025, P00026 LED 의 상태를 확인하고 코일, 역코일, 반전 기호의 기능을 확인하십시오.



\* 위의 프로그램에서 P00025 와 P00026 출력 접점은 동일한 결과를 가집니다. 즉, 역코일과 반전 기호 + 코일은 동일한 결과를 가집니다.



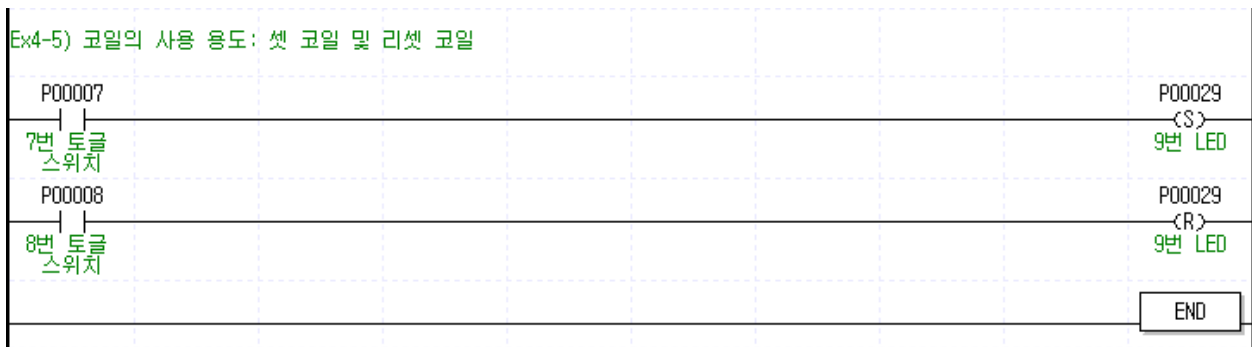
- 양변환 검출 코일: 양변환 검출 코일은 논리 연산의 결과가 OFF 에서 ON 으로 변화될 때 지정된 비트 메모리를 ON 시켰다가 그 다음 스캔에서 OFF 시킵니다. 결과적으로 양변환 검출 코일은 논리 연산 결과의 상승 에지가 발생한 순간부터 1 스캔 시간 동안 지정된 비트를 ON 시킵니다.
- 음변환 검출 코일: 음변환 검출 코일은 논리 연산의 결과가 ON 에서 OFF 로 변화될 때 지정된 비트 메모리를 ON 시켰다가 그 다음 스캔에서 OFF 시킵니다. 결과적으로 음변환 검출 코일은 논리 연산 결과의 하강 에지가 발생한 시간부터 1 스캔 시간 동안 지정된 비트를 ON 시킵니다.

Ex 4-4) 다음의 프로그램을 작성하여 PLC 로 전송한 후 P00004, P00005, P00006 토글 스위치를 ON /OFF 시키면서 P00027, P00028 LED 의 상태를 확인하고, 양변환 검출 코일과 음변환 검출 코일의 기능을 확인 하십시오.

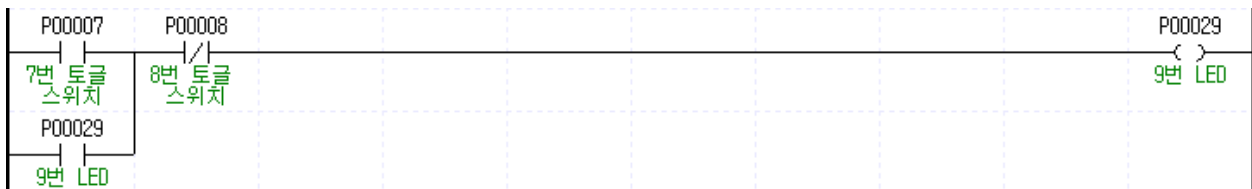


- 셋 코일: 셋 코일은 논리 연산의 결과가 1 이 될 때 지정된 비트에 1 을 저장(ON) 합니다. 지정된 비트에 1 이 저장된 후 논리 연산의 결과가 0 으로 변화되어도 지정된 비트에 저장되어 있는 1 을 변경하지 않습니다. 또 지정된 비트에 1 이 저장되어 있는 상태에서 논리 연산의 결과가 1 이 될 경우 지정된 비트에는 아무런 변화가 없습니다.
- 리셋 코일: 리셋 코일은 논리 연산의 결과가 1 이 될 때 지정된 비트에 0 을 저장(OFF) 합니다. 지정된 비트에 0 이 저장된 후 논리 연산의 결과가 0 으로 변화되어도 지정된 비트에 저장되어 있는 0 을 변경하지 않습니다. 또 지정된 비트에 0 이 저장되어 있는 상태에서 논리 연산의 결과가 1 이 될 경우 지정된 비트에는 아무런 변화가 없습니다.

Ex 4-5) 다음의 프로그램을 작성하여 PLC 로 전송한 후 P00007, P00008 토글 스위치를 ON /OFF 시키면서 P00029 LED 의 상태를 확인하고, 양변환 검출 코일과 음변환 검출 코일의 기능을 확인하십시오.



셋/리셋 코일과 자기 유지: 셋 코일은 지정된 비트를 ON 시키는 역할을 하고, 리셋 코일은 지정된 비트를 OFF 시키는 역할을 합니다. 위의 프로그램과 같이 동일한 주소를 셋/리셋 시키는 경우 아래의 자기 유지 회로와 동일하게 동작합니다.



### 3) 논리 연산

PLC 의 가장 기본적인 기능은 센서, 스위치 등의 디지털 입력 신호에 따라 PLC 디지털 출력 모듈의 접점을 ON/OFF 시킴으로써 출력 모듈에 연결된 부하를 구동시키는 전원을 공급 또는 차단함으로써 부하를 구동 또는 정지시키는 것입니다. 부하를 구동 또는 정지 시키는 것은 1 개의 입력 신호에 의해 제어되는 경우도 있지만, 여러 개의 입력 신호의 조합에 의해 제어되는 경우가 많습니다. 여러 개의 입력 신호에 의해 1 개의 부하가 구동될 때 각 입력 신호의 논리 연산에 의해 부하는 구동 또는 정지를 하게 됩니다. 여기에서는 PLC 가 기본적으로 제공하는 비트 신호의 논리 연산에 대해 설명합니다.

PLC 는 두 개의 비트 디바이스의 상태를 읽어 논리 연산을 수행합니다. 여러 개의 비트 디바이스의 논리

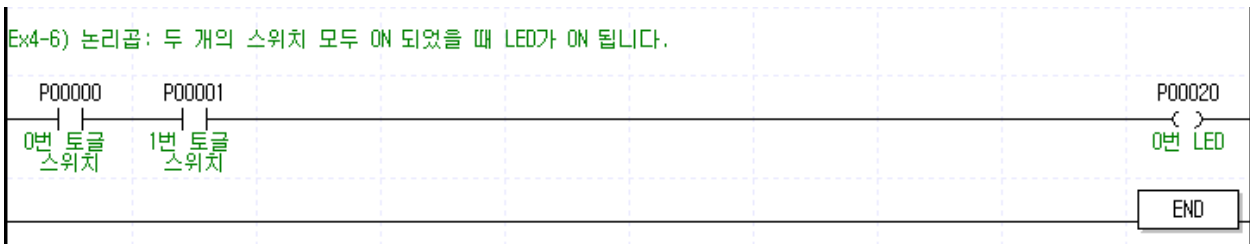
연산이 요구될 경우 선두 2 개 비트 디바이스의 상태를 읽어 논리 연산을 수행하고 그 결과와 그 다음 디바이스의 상태와의 논리 연산을 수행합니다.

- 논리곱(AND) 연산: 논리곱(AND) 연산은 두 개의 비트 디바이스의 상태를 읽어 그 중 1 개의 비트 디바이스라도 0 의 상태(OFF)를 가지면 0 을 출력하고, 두 개의 비트 디바이스 모두 1 의 상태(ON)이면 1 을 출력하는 논리 연산입니다. 즉 두 비트 디바이스의 상태에 따라 다음과 같은 결과를 출력합니다.

$$0 \& 0 = 0, \quad 0 \& 1 = 0, \quad 1 \& 0 = 0, \quad 1 \& 1 = 1$$

PLC 에서 논리곱 (AND) 연산은 비트 디바이스의 수평적 연결로 이루어지며, '조건 모두가 만족되었을 때'라는 의미가 있습니다.

Ex 4-6) P00000, P00001 두 개의 스위치가 모두 ON 되었을 때 P00020 LED 를 ON 시키고, 두 개의 스위치 중 1 개라도 OFF 되면 LED 가 OFF 됩니다.

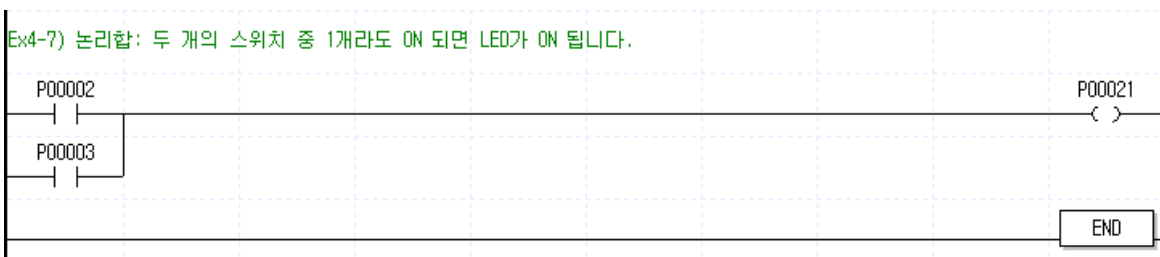


- 논리합(OR) 연산: 논리합(OR) 연산은 두 개의 비트 디바이스의 상태를 읽어 그 중 1 개의 비트 디바이스라도 1 의 상태(ON)를 가지면 1 을 출력하고, 두 개의 비트 디바이스 모두 0 의 상태 (OFF)이면 0 을 출력하는 논리 연산입니다. 즉 두 비트 디바이스의 상태에 따라 다음과 같은 결과를 출력합니다.

$$0 \parallel 0 = 0, \quad 0 \parallel 1 = 1, \quad 1 \parallel 0 = 1, \quad 1 \parallel 1 = 1$$

PLC 에서 논리합 (OR) 연산은 비트 디바이스의 수직적 연결로 이루어지며, '여러 조건 중 1 개의 조건이라도 만족되었을 때'라는 의미가 있습니다.

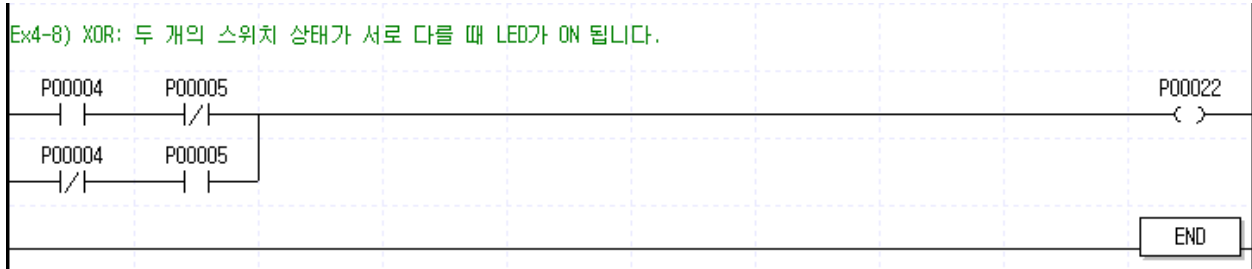
Ex 4-7) P00002, P00003 두 개의 스위치가 중 1 개의 스위치라도 ON 되면 P00021 LED 를 ON 시키고, 두 개의 스위치 모두 OFF 되면 LED 가 OFF 됩니다.



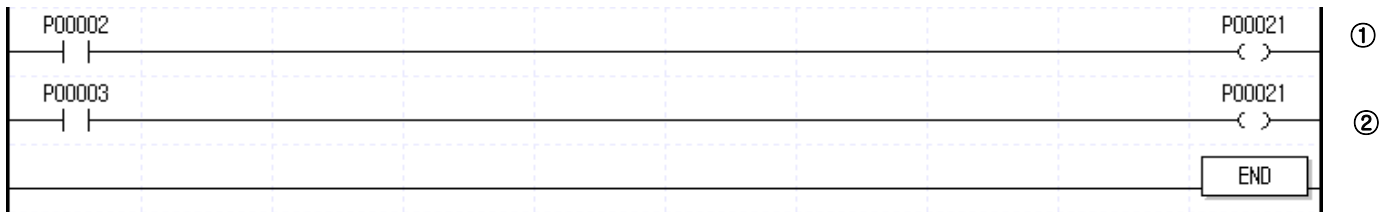
Ex 4-8) 논리곱과 논리합을 이용하여 P00004 와 P00005 스위치의 XOR 연산 결과를 P00022 LED 로 표시합니다.

XOR 연산은 두 비트의 데이터가 동일하면 0, 서로 다르면 1 의 결과를 출력하는 논리 연산입니다.

$$0 \text{ XOR } 0 = 0, \quad 0 \text{ XOR } 1 = 1, \quad 1 \text{ XOR } 0 = 1, \quad 1 \text{ XOR } 1 = 0$$



앞의 Ex 4-7)을 다음과 같이 프로그램 했을 경우를 생각해 보겠습니다.



P00002 접점이 ON 되면 P00021 코일이 ON 되고, P00003 접점이 ON 되어도 P00021 코일이 ON 되어 예제에서 제시한 조건대로 동작할 듯 합니다. 그러나, 자세히 살펴보면 프로그램이 ①까지 실행되었을 때 P00021 코일은 P00002 접점에 의해 출력이 결정됩니다.

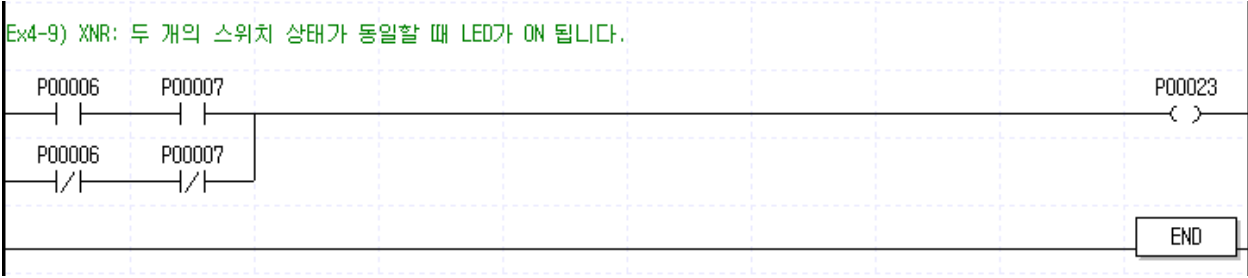
그리고, 프로그램이 ②까지 실행되었을 때는 P00003 에 의해 P00021 의 출력이 결정됩니다. 실제 출력은 프로그램의 실행이 완료된 후 출력 리프레시를 할 때 이루어지므로 결국 P00021 의 출력은 항상 P00003 에 의해 결정된 출력을 하게 됩니다.

이와 같이 출력 코일에 동일한 비트 주소를 사용하는 것을 이중코일이라고 합니다. 이러한 이중 코일을 오류로 처리할 지 경고로 처리할 지 XG5000 의 보기 메뉴 >> 프로그램 검사 항목에서 선택할 수 있습니다.

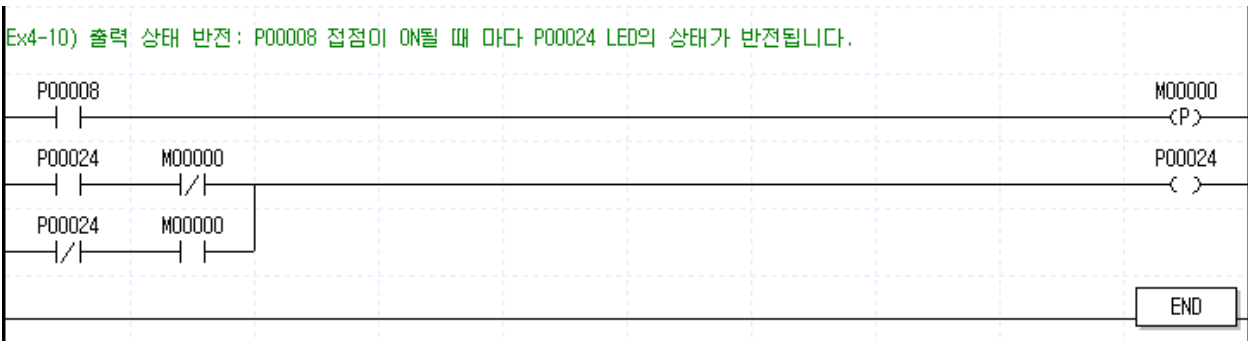
XG5000 에서 프로그램을 PLC 로 전송할 때 프로그램을 검사한 후 PLC 로 전송하게 되는데, 이중 코일을 오류로 처리하면 프로그램에 이중 코일이 있을 경우 프로그램을 PLC 로 전송하지 않으며, 경고로 처리하면 프로그램에 이중 코일이 있더라도 PLC 로 전송하고, 위에서 설명한 대로 코일이 등록된 프로그램의 제일 아래의 조건들에 의해 출력이 동작하게 됩니다.

Ex 4-9) 논리곱과 논리합을 이용하여 P00006 과 P00007 스위치의 XNR 연산 결과를 P00023 LED 로 표시합니다. XNR 은 XOR 과 반대로 두 비트의 데이터가 동일하면 1, 서로 다르면 0 의 결과를 출력하는 논리 연산입니다.

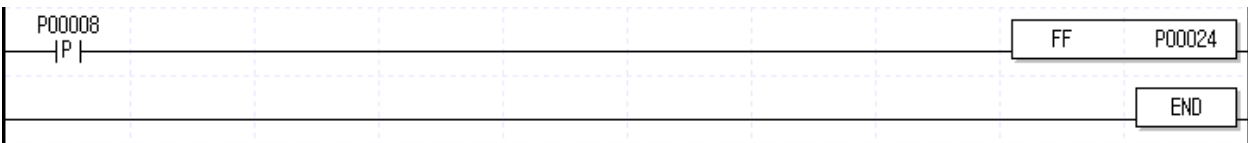
0 XNR 0 = 1, 0 XNR 1 = 0, 1 XNR 0 = 0, 1 XNR 1 = 1



Ex 4-10) P00008 스위치를 ON 할 때 마다 P00024 LED의 상태를 반전합니다.

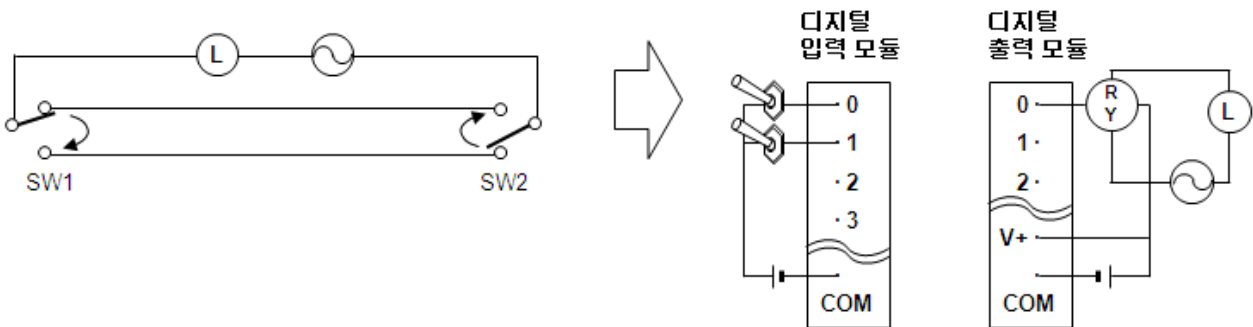


위의 프로그램은 아래와 같이 비트 반전 응용 명령어 FF를 이용하여 간단히 구현할 수 있습니다.



Ex 4-11) 두 개의 스위치에 의한 출력 상태 반전 회로 (삼로 스위치)

긴 복도나 계단의 한 쪽 끝에 있는 스위치를 조작하면 램프가 점등되며, 반대쪽에서 스위치를 조작하면 램프가 소등됩니다.



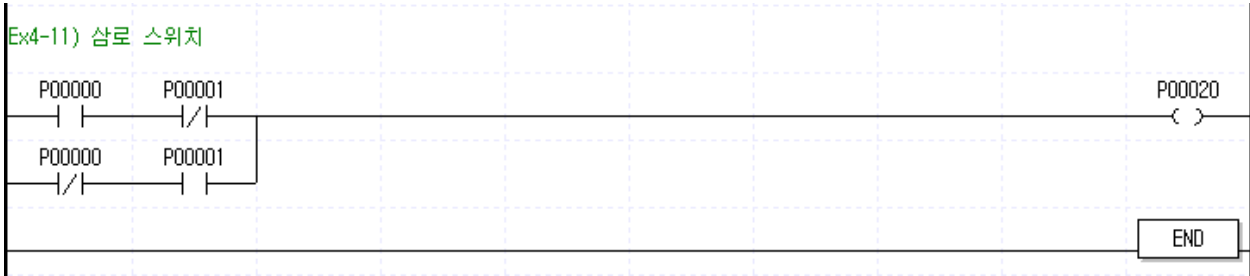
\* DC 전원을 사용하는 디지털 입력 및 출력 모듈 배선 시 모듈 종류에 따라 극성에 주의해야 합니다.

- DC 전원 입력 모듈: XGI-D2 □ A: 무극성, XGI-D2 □ B: + COM

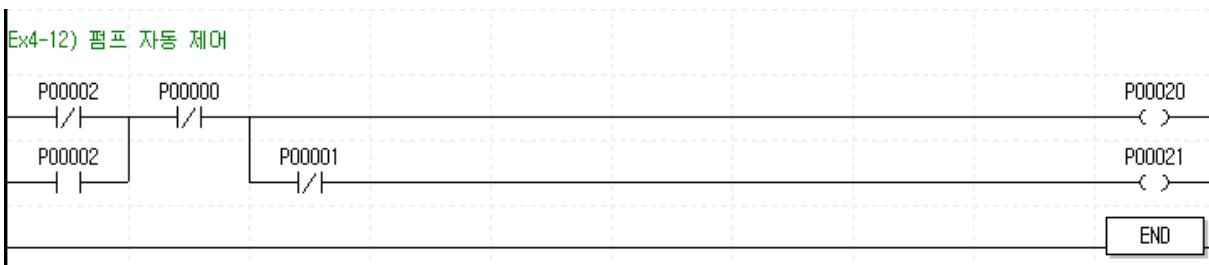
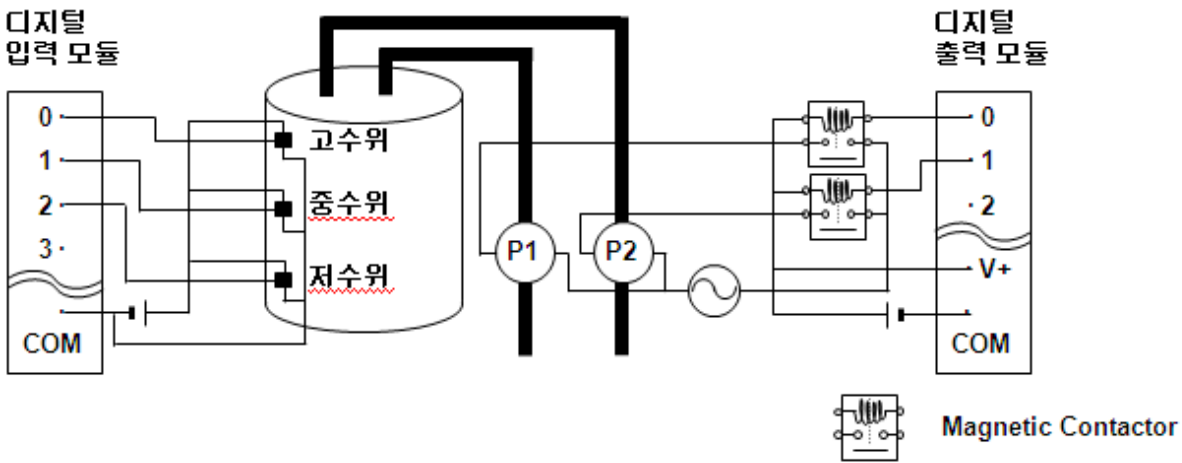
- TR 출력 모듈: XGQ-TR □ A: - COM, XGQ-TR □ B: + COM

디지털 출력 모듈 선정 시 부하가 사용하는 전원의 종류(AC/DC)에 따라 사용할 수 있는 모듈의 선정에 주의해야 하며, 부하의 소비 전류가 디지털 출력 모듈의 정격 전류 용량을 초과하는 경우 릴레이 또는 MC(Magnetic Contactor) 등을 사용하여 배선해야 합니다.

- DC 부하: 릴레이 출력(1A/1P, 5A/1COM), TR 출력(XGQ-TR2A/B: 0.5A/1P, 4A/1COM, 그 외: 0.1A/1P, 2A/1COM)
- AC 부하: 릴레이 출력 (1A/1P, 5A/1COM), SSR 출력 (0.6A/1P, 4A/1COM)

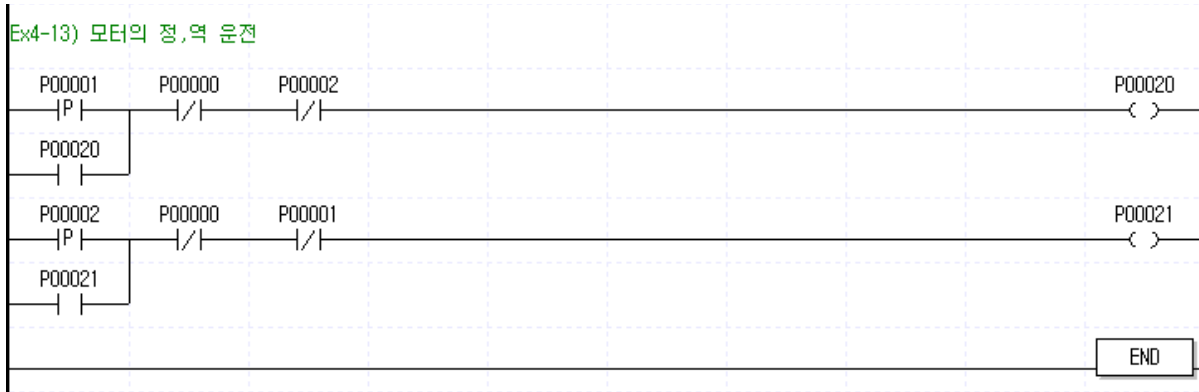
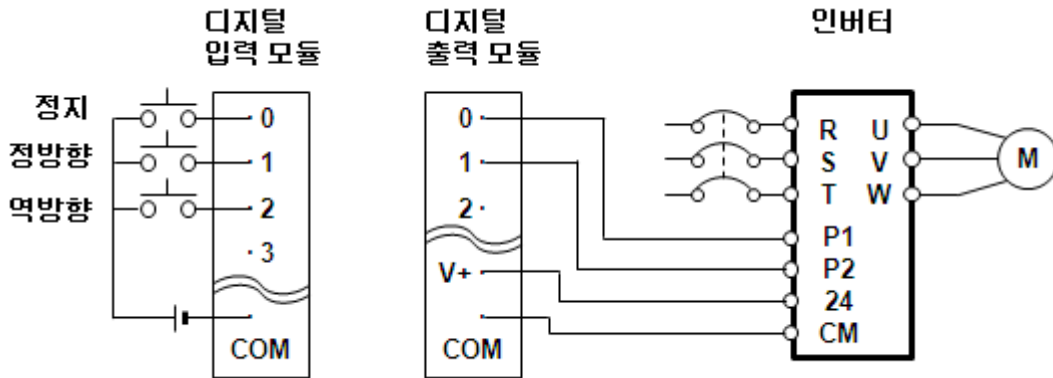


Ex 4-12) 고수위, 중수위, 저수위 3 개의 센서가 부착된 탱크에 두 대의 펌프를 이용하여 수위를 고수위 ~ 저수위 사이를 유지합니다. 저수위 센서만 감지 되었을 때 두 대의 펌프 모두 가동을 하고, 중수위 센서가 감지되면 1 대의 펌프만 가동하며, 고수위 센서가 감지되면 두 대의 펌프 모두 정지합니다.



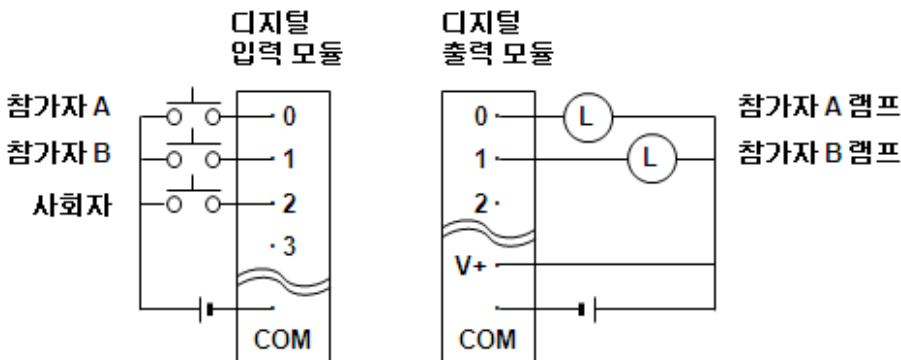
위의 프로그램에서 P00002 는 평상시 닫힌 접점과 평상시 열린 접점이 병렬로 사용되었습니다. 그 이유는 최초 동작 또는 다른 이유로 수위가 저수위 센서 아래 일 때 기동이 가능하게 하기 위함입니다. 이 경우 저수위 센서는 제거해도 제어에는 영향을 미치지 않습니다. 다만, 저수위 센서를 알람 등의 용도로 사용할 경우에는 저수위 센서를 제거할 수 없습니다.

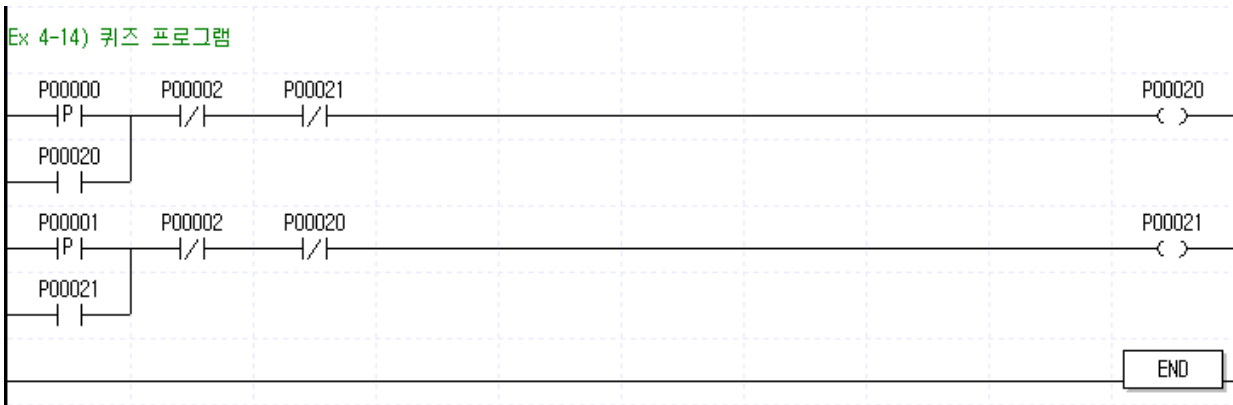
Ex 4-13) 정방향 Push Button 을 누르면 모터는 정방향으로 회전(P1 ON, P2 OFF)하고, 역방향 Push Button 을 누르면 모터가 역방향으로 회전(P1 OFF, P2 ON) 합니다. 정지 Push Button 을 누르면 모터는 정지(P1 OFF, P2 OFF) 합니다.



\* 위의 예제에서와 같이 Push Button 을 이용하여 장비를 조작할 때 양변환 검출 접점 + 자기 유지 회로를 사용하는 것이 편리합니다.

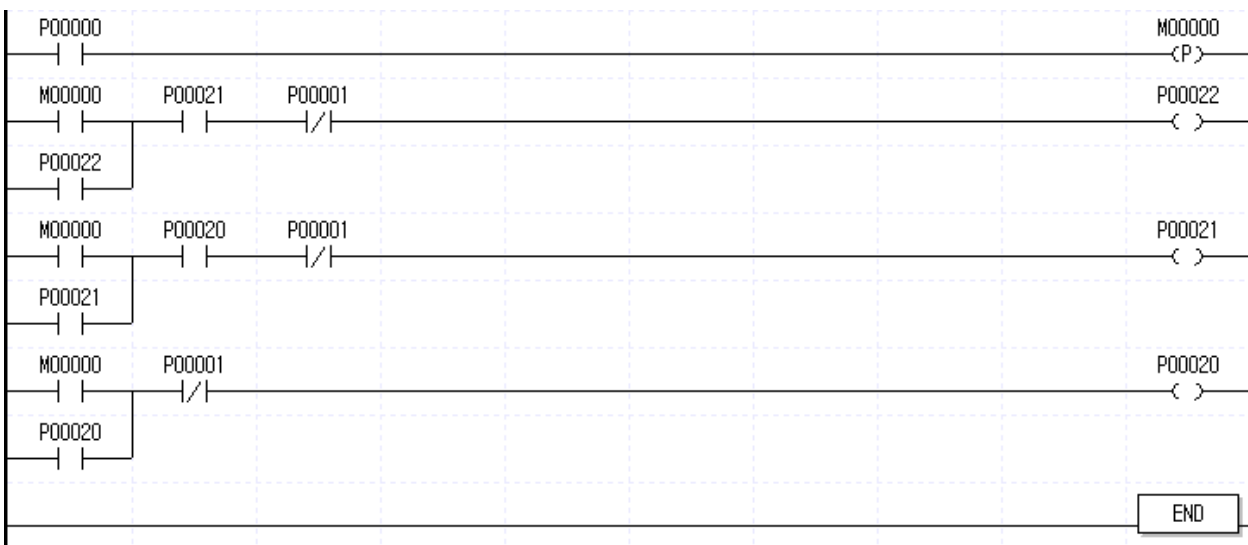
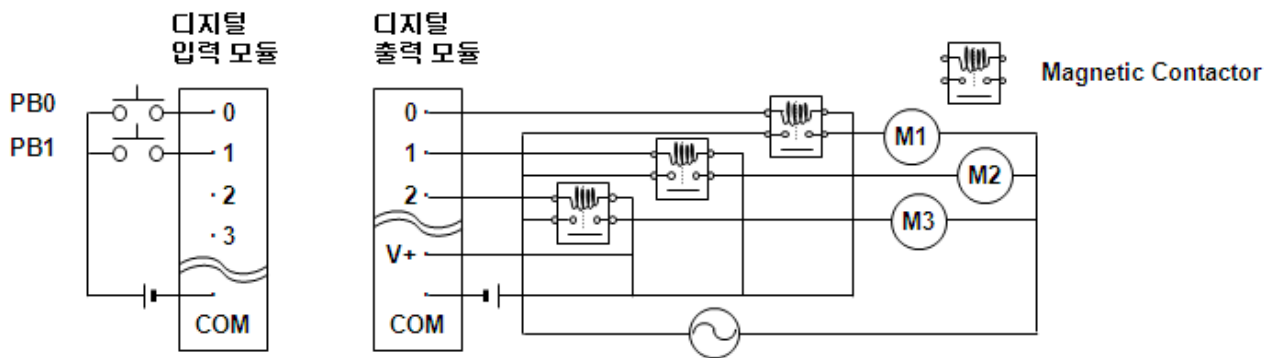
Ex 4-14) 퀴즈 참가자 A, B 가 사회자의 문제에 따라 Push Button 을 누를 때, 먼저 누른 참가자 측의 램프가 점등되어, 사회자가 Reset Button 을 누르기 전까지 램프 출력을 유지합니다. 먼저 누른 참가자의 램프가 점등되어 있는 동안 다른 참가자가 Button 을 눌러도 램프는 동작하지 않습니다.





예제 13, 14 번 처럼 다른 출력의 조건 또는 다른 출력 그 자체에 의해 출력이 제어되는 회로를 인터록 회로라고 합니다. 예제 13 번의 경우 다른 출력 조건의 평상시 닫힌 접점(b 접점)이 조건으로 삽입된 경우 후입 우선 인터록이 되며, 예제 14 번처럼 다른 출력의 평상시 닫힌 접점 (b 접점)이 조건으로 사용된 경우 선입우선 인터록 회로가 됩니다.

Ex 4-15) PB0 을 한 번 누르면 M1 이 ON 되고, 두 번 누르면 M1 과 M2 가 ON 되며, 세 번 누르면 M1, M2, M3 가 모두 ON 됩니다. PB1 을 누르면 ON 되어 있는 모든 모터가 정지합니다.



### 4.3 타이머

시퀀스 프로그램 중 지정된 시간 이상의 시간 동안 조건 만족을 유지할 때 어떤 동작을 수행해야 하는 경우 등 시간 지연 요소가 필요할 경우 사용하는 프로그램 요소가 타이머 입니다.

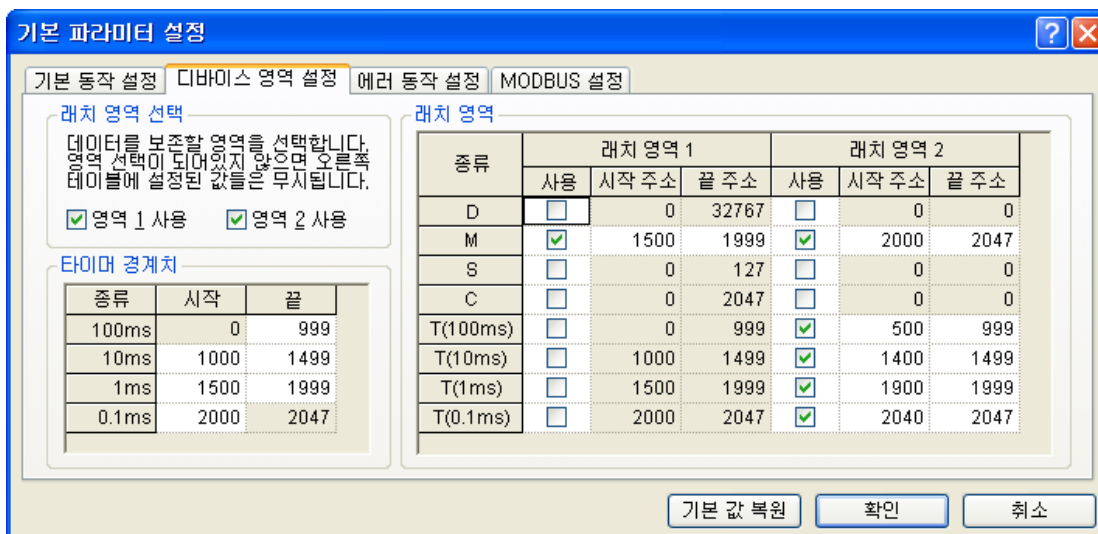
XGK PLC 에는 2,048 개의 타이머가 있으며, 타이머의 기능별로 TON, TOFF, TMR, TMON, TRTG 등 5 종, 시간 설정 단위로 100ms, 10ms, 1ms, 0.1ms 의 4 종의 타이머가 있습니다. 또, 타이머를 사용하지 않고, 비트 디바이스를 직접 시간으로 제어하는 TFLK 가 있습니다. 1 개의 타이머는 최대 6,553.5 초(65,535 X 100ms)까지 시간 제어를 할 수 있으며, 타이머를 사용하지 않는 TFLK 는 최대 65,535 초까지 시간 제어를 할 수 있습니다.

#### 1) 타이머 경계치 및 래치 설정


XGK PLC 타이머의 시간 설정 단위는 100ms, 10ms, 1ms, 0.1ms 의 4 종류가 있습니다. 시간 설정 단위란 프로그램에서 타이머 편집 시 설정값을 '1'로 설정했을 때 '1'의 의미가 됩니다. 즉, 100ms 타이머에 설정값을 1로 설정했다면 100ms 시간 제어가 된다는 의미입니다.

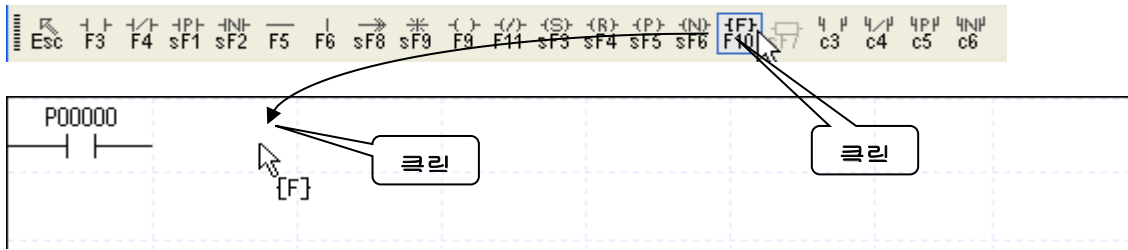
시간 설정 단위 별로 사용할 수 있는 타이머의 수는 기본 파라미터의 '디바이스 영역 설정' 항목에서 설정할 수 있으며, 각 설정 시간 단위 별 타이머에 최소 1 개 이상의 타이머가 할당되어야 합니다.

정전 시 타이머 동작 시간 데이터가 보존 되어야 할 타이머의 경우 래치 영역의 타이머를 사용해야 하며, 기본 파라미터의 '디바이스 영역 설정' 항목에서 각 설정 시간 별 타이머에 래치 영역을 설정하여 사용할 수 있습니다. 초기 설정은 모든 타이머 영역이 휘발성 영역으로 설정 되어 있으므로 래치 타이머가 필요한 경우 사용자가 지정해서 사용해야 합니다.



2) 타이머의 편집

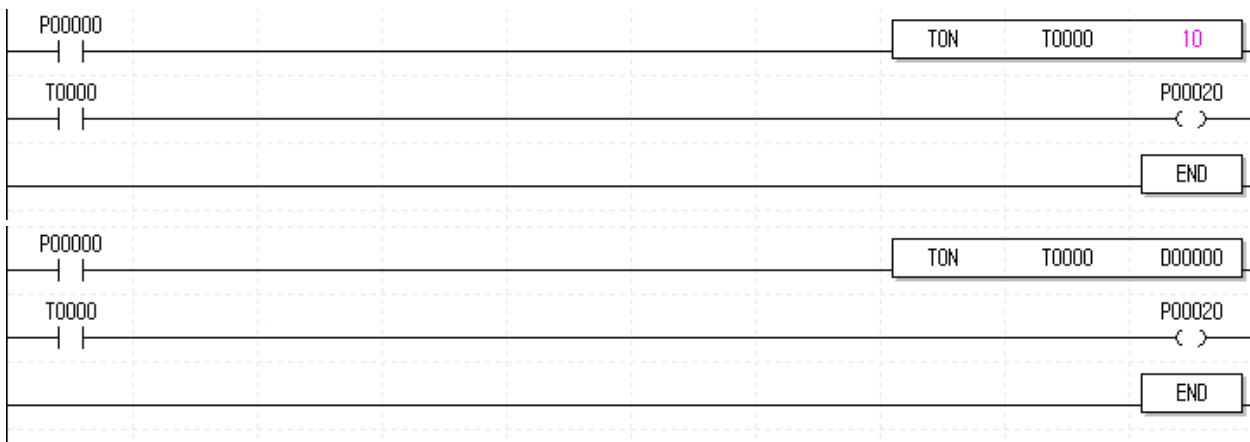
도구 상자에서 평선/평선블록(  )을 선택하거나 평선/평선블록 단축키 (F10)를 누릅니다.




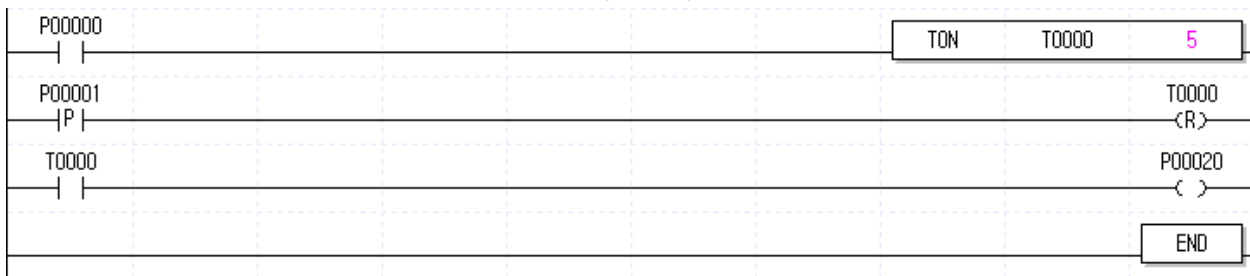
응용명령 창에서 타이머 종류, 타이머 번호, 설정값 순으로 입력합니다. 만일 타이머의 설정값을 고정값으로 하지 않고 가변적으로 사용하고자 할 경우 설정값 위치에 워드형 디바이스를 입력합니다. 타이머의 최대 설정값은 65,535 이며, 설정값에 따른 설정 시간은 기본 파라미터의 디바이스 영역 설정 항목에서 설정된 타이머 경계치에 따라 그 의미가 달라집니다. 설정값을 HMI 등 외부에서 설정할 경우 16 비트 부호 없는 10진 정수로 설정해야 합니다.

타이머는 조건이 만족된 후 타이머 동작 특성에 따라 시간 지연 후 타이머 접점을 제어하게 됩니다. 이 후 타이머 접점을 이용하여 시퀀스 프로그램을 작성하면 됩니다.

본 예제에서 설정값을 10으로 설정한 경우 P00000 입력 접점이 1초 이상 ON을 유지할 경우 P00020 출력 접점이 ON되며, P00000 입력 접점이 OFF 될 때 P00020 출력 접점이 OFF 됩니다.



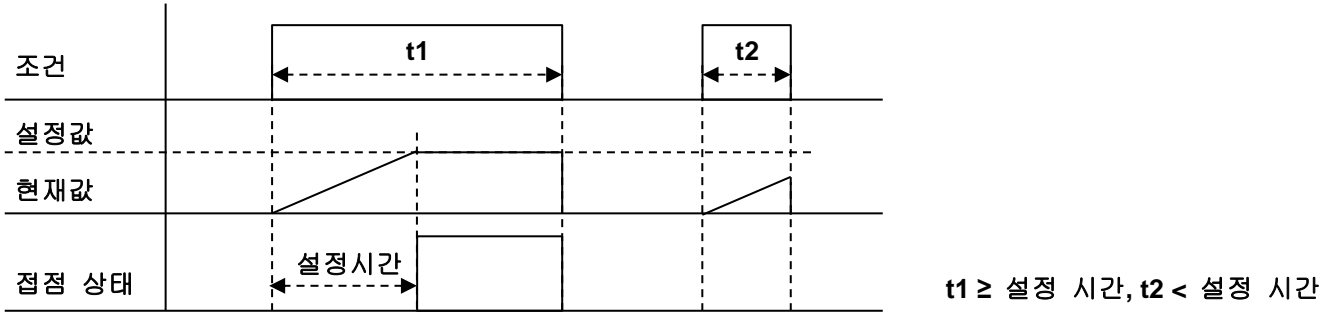
타이머를 리셋 할 때 아래와 같이 리셋 코일(  )을 이용하여 타이머를 리셋 합니다.



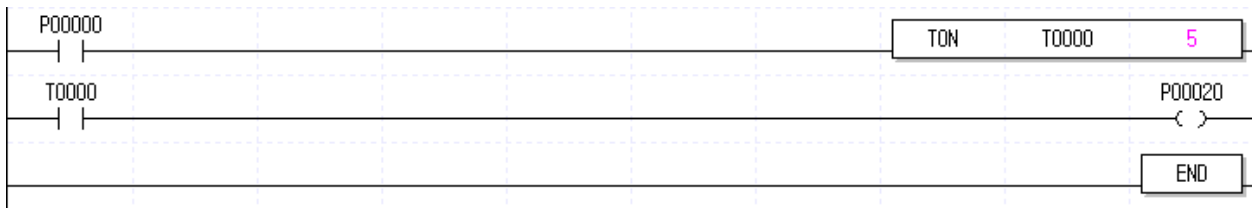
3) 타이머의 동작 특성

**TON (ON Delay Timer):** ON Delay Timer 는 조건이 만족된 후 설정된 시간 이상 조건이 유지될 때 타이머의 접점이 ON 되는 타이머로 세부 동작 사항은 다음과 같습니다.

- 타이머의 기동 조건이 만족되면 현재값을 설정 시간 단위로 1 씩 증가시키며, 현재값 = 설정값이 될 때 타이머 접점이 ON 됩니다.
- 타이머 동작 중 기동 조건이 해제되면 현재값 = 0 이 됩니다.
- 타이머 접점이 ON 된 상태에서 기동 조건이 해제되면 타이머 접점이 OFF 됩니다.
- 타이머를 리셋하면 현재값 = 0 이 되며, 타이머 접점이 ON 되어 있는 경우 접점은 OFF 됩니다.

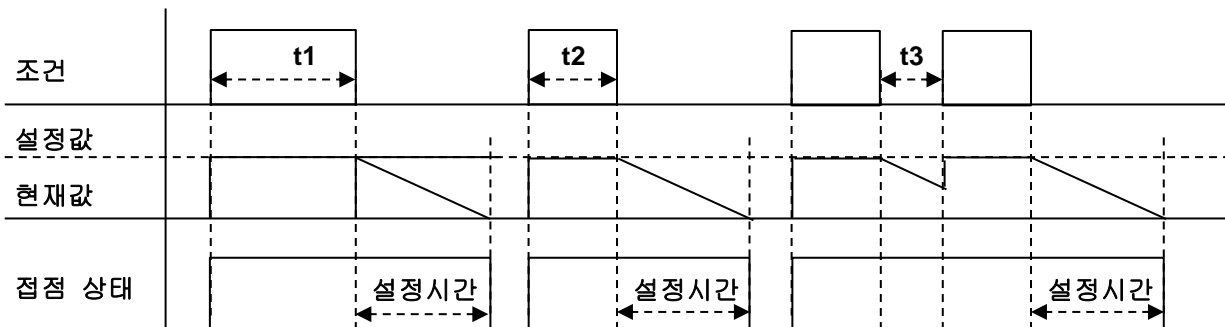


Ex 4-16) P00000 입력 스위치가 ON 된 후 0.5 초 이상 ON 상태를 유지하면 P00020 이 ON 되고, P00000 이 OFF 되면 P00020 이 OFF 됩니다.

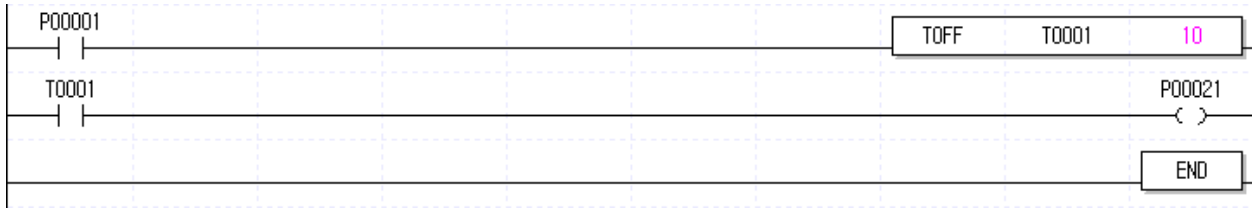


**TOFF (OFF Delay Timer):** OFF Delay Timer 는 조건이 만족될 때 타이머 접점이 ON 되고, 조건이 OFF 된 후 설정된 시간 동안 접점이 ON 상태를 유지하는 타이머로 세부 동작 사항은 다음과 같습니다.

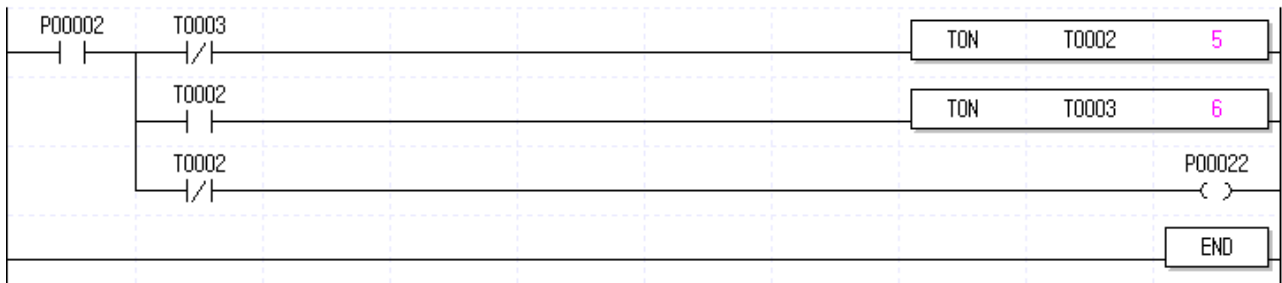
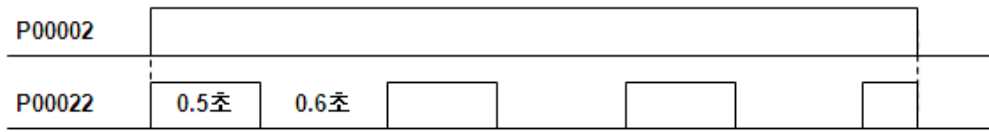
- 타이머의 기동 조건이 만족되면 현재값 = 설정값이 되고, 타이머 접점은 ON 됩니다.
- 타이머의 기동 조건이 해제되면 현재값을 설정 시간 단위로 1 씩 감소 시키고 현재값이 0 이 될 때 타이머의 접점은 OFF 됩니다.
- 타이머의 현재값이 감소하고 있는 상태에서 다시 기동 조건이 만족되면 현재값 = 설정값이 되고, 접점은 ON 됩니다.
- 타이머를 리셋하면 현재값 = 0 이 되며, 타이머 접점이 ON 되어 있는 경우 접점은 OFF 됩니다.



Ex 4-17) P00001 입력 스위치가 ON 되면 P00021 이 ON 되고, P00001 이 OFF 되면 1 초 후에 P00021 이 OFF 됩니다.



Ex 4-18) 플리커 회로: 두 개의 타이머를 사용하여 P00002 입력 스위치가 ON 되면 P00022 LED 가 0.5 초 동안 ON, 0.6 초 동안 OFF 를 반복합니다.



위의 프로그램은 타이머를 사용하지 않고 TFLK 명령어를 사용하여 아래와 같이 구현할 수 있으며, TFLK 명령어의 내용은 다음과 같습니다.

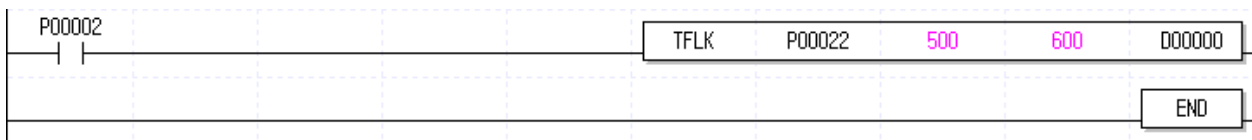
명령어 형식: [ TFLK D1 S1 S2 D2 ]

여기서 D1 은 ON/OFF 할 접점(bit), S1 은 ON 시간(word), S2 는 OFF 시간(word)입니다.

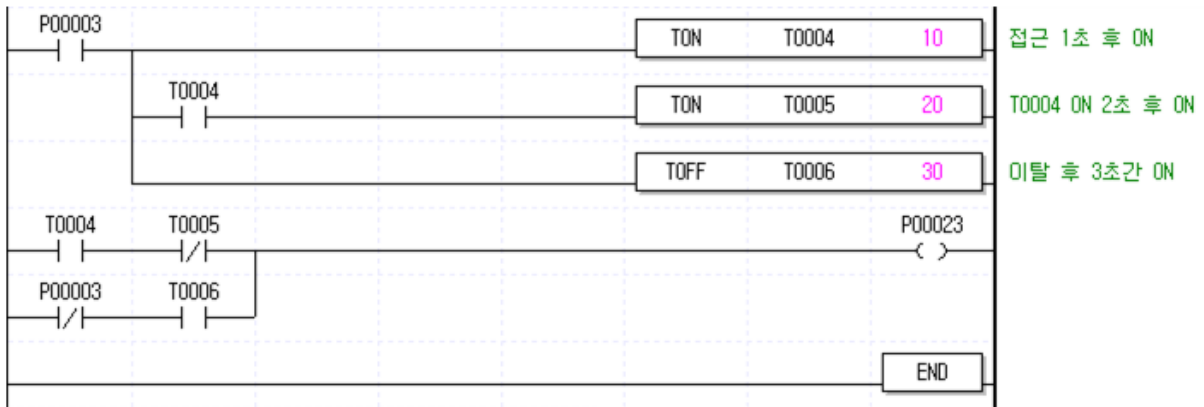
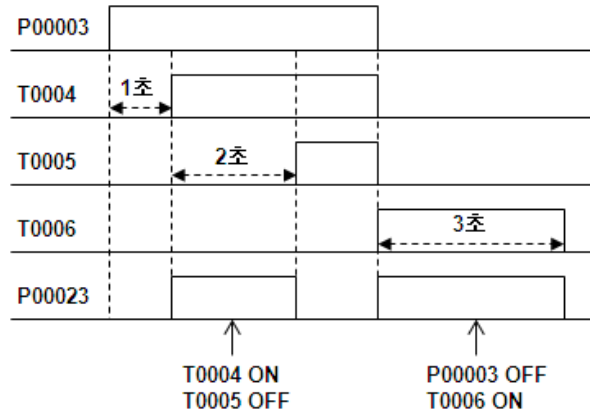
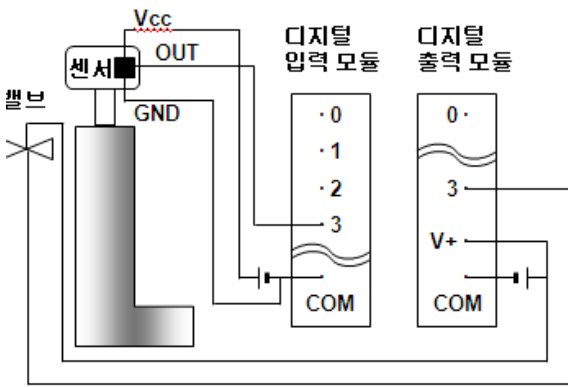
D2 는 총 4word 의 데이터 메모리 영역이 사용되며 각 메모리 영역의 내용은 다음과 같습니다.

D(2+0): 현재치 저장                      D(2+1): 시간 단위 설정 (0: ms, 1: 10ms, 2: 100ms, 3: 1s)

D(2+2) ~ D(2+3): 명령어 사용 영역

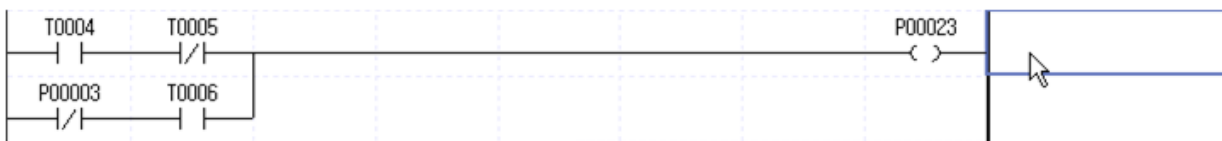


Ex 4-19) 화장실 자동 밸브 제어: 사용자가 변기에 접근한 후 1 초 뒤 2 초간 물이 나오고 이탈 후 즉시 3 초간 물이 공급되는 회로입니다.

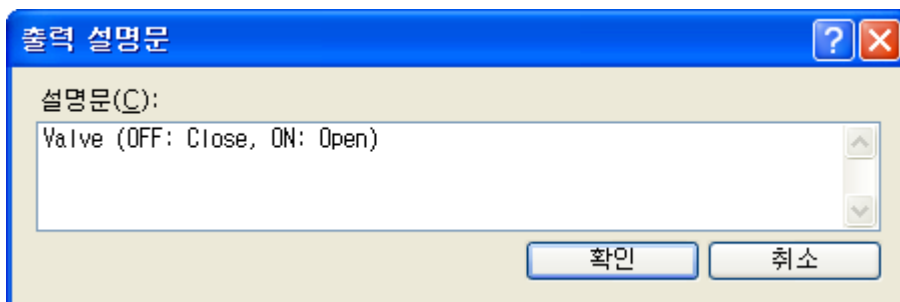


\* 출력 설명문 편집 방법: 프로그램 라인의 각 라인에 설명문을 추가함으로써 좀 더 쉽게 프로그램을 이해할 수 있습니다. 다음의 과정을 통해 출력 설명문을 편집할 수 있습니다.

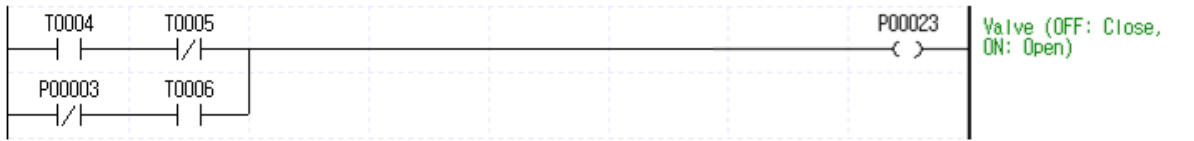
프로그램의 편집이 완료된 행의 우측 모션 바깥을 더블 클릭 합니다.



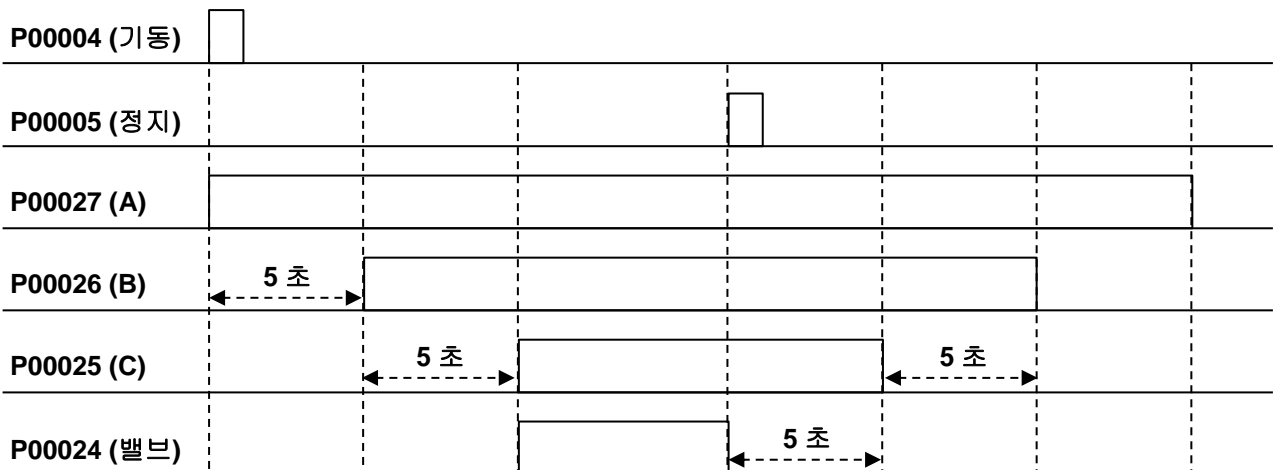
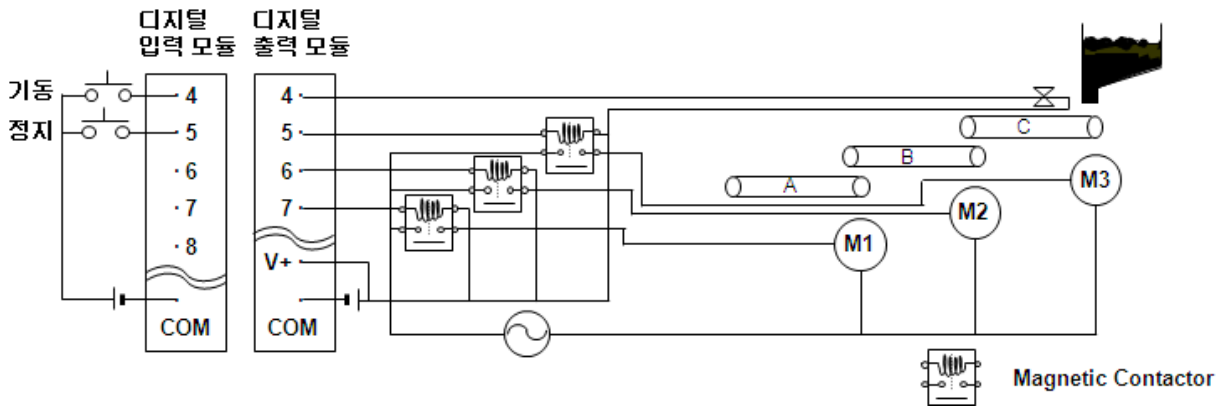
- 출력 설명문 대화 상자에서 프로그램 라인에 대한 설명문을 입력하고 '확인' 버튼을 선택 합니다.

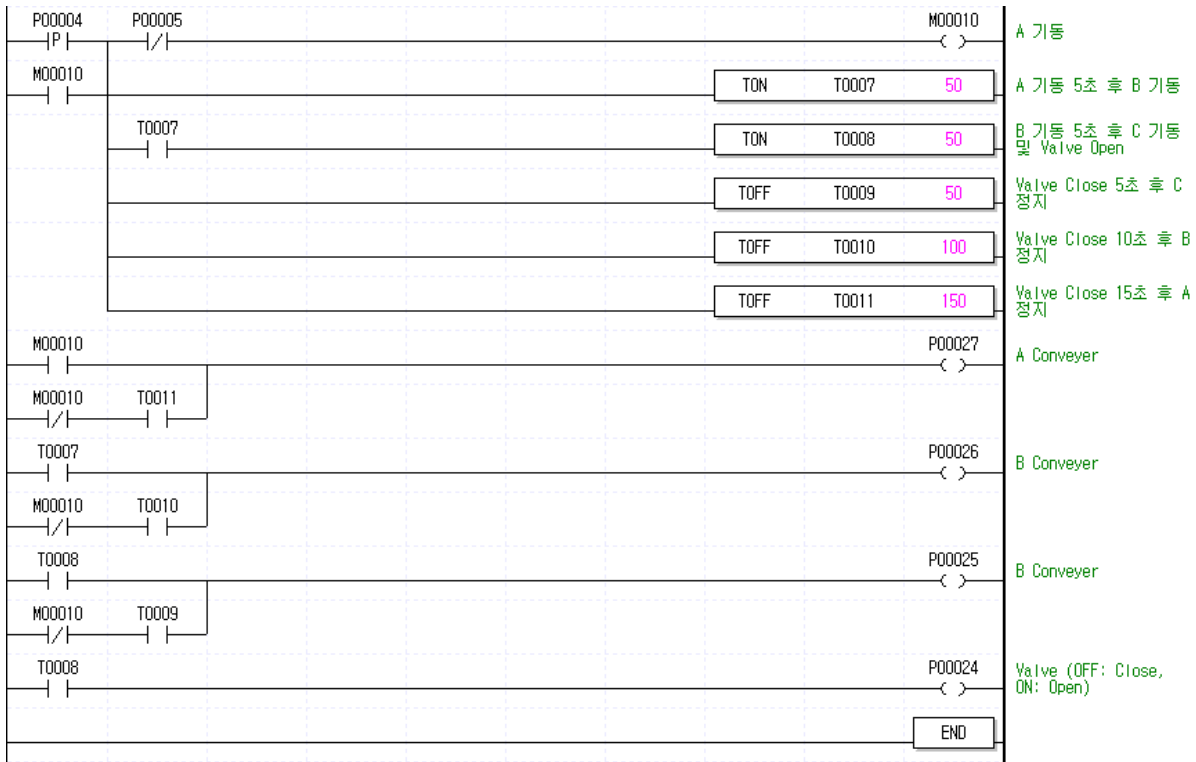


- 출력 설명문이 편집되었습니다.



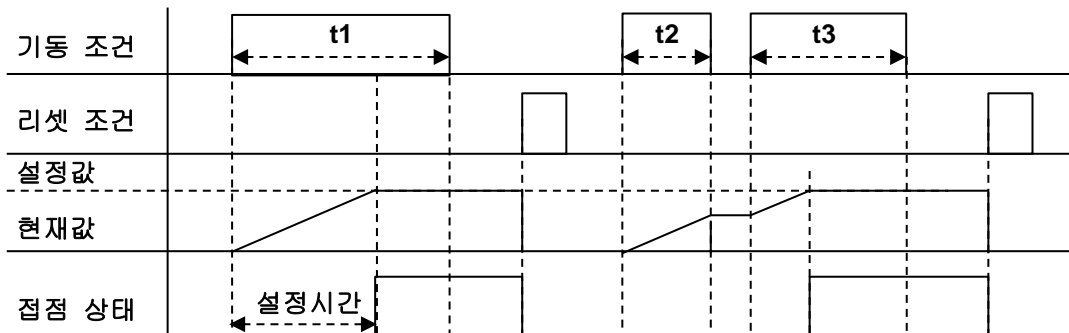
Ex 4-20) 컨베이어 제어: 여러 대의 컨베이어를 순서에 따라 5 초 간격으로 기동(A → B → C), 정지(C → B → A)합니다.



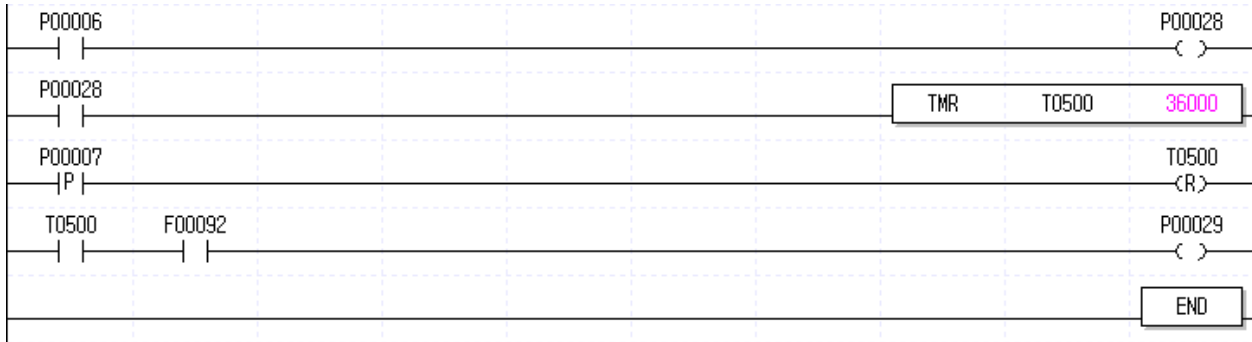


**TMR (적산 Timer):** 적산 Timer 는 조건이 만족된 후 설정된 시간 이상 조건이 유지될 때 타이머의 접점이 ON 되는 타이머로 세부 동작 사항은 다음과 같습니다.

- 타이머의 기동 조건이 만족되면 현재값을 설정 시간 단위로 1 씩 증가시키며, 현재값 = 설정값이 될 때 타이머 접점이 ON 됩니다.
- 타이머의 접점이 ON 된 상태에서 기동 조건을 계속 만족하더라도 더 이상 현재값은 증가하지 않습니다.
- 타이머 동작 중 기동 조건이 해제되면 현재값을 유지합니다.
- 타이머 접점이 ON 된 상태에서 기동 조건이 해제되면 타이머 접점이 ON 상태를 유지합니다.
- 타이머를 리셋하면 현재값 = 0 이 되며, 타이머 접점이 ON 되어 있는 경우 접점은 OFF 됩니다.



Ex 4-21) P00006 입력 스위치가 ON 되면 P00028 LED 가 ON 되고, P00028 LED 의 누적된 ON 시간이 1 시간이 되면 P00029 LED 가 200ms 주기로 점멸(100ms ON, 100ms OFF 반복)합니다.



\* 앞의 프로그램에서 F00092 비트를 사용했습니다. F00092 비트는 200ms 주기의 클럭으로 100ms ON, 100ms OFF 를 반복합니다.

이와 같이 사용자가 제어하지 않아도 PLC 시스템에서 제어해 주는 데이터를 플래그 또는 시스템 메모리 라고 합니다.

XGT PLC 에서는 시스템 플래그(F 영역), PID 플래그(K 영역), P2P 플래그(L 영역), 고속링크 플래그(L 영역) 등 기능별 플래그를 제공하고 있으며, 대부분의 플래그는 읽기 전용 영역으로 사용됩니다. 플래그는 XG5000에서 자동으로 변수 및 설명문을 등록하기 때문에 사용자가 변수 또는 설명문 등록하지 않아도 변수 또는 설명문을 볼 수 있습니다.

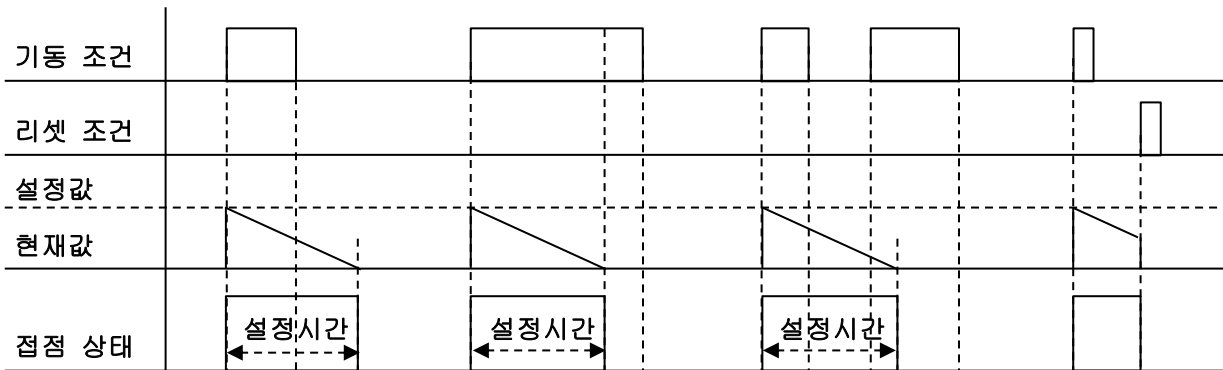
아래의 표는 프로그램에서 자주 사용되는 시스템 플래그 입니다.

디바이스	변수	설명	디바이스	변수	설명
F00000	_RUN	PLC Run 시 ON	F00096	_T20S	20 초 주기 CLOCK (10 초 ON, 10 초 OFF)
F00001	_STOP	PLC Run 시 ON	F00097	_T60S	1 분 주기 CLOCK (30 초 ON, 30 초 OFF)
F00002	_ERROR	Error 발생 시 ON	F00099	_ON	항상 ON
F00090	_T20MS	20ms 주기 CLOCK (10ms ON, 10ms OFF)	F0009A	_OFF	항상 OFF
F00091	_T100MS	100ms 주기 CLOCK (50ms ON, 50ms OFF)	F0009B	_1ON	첫 스캔 ON

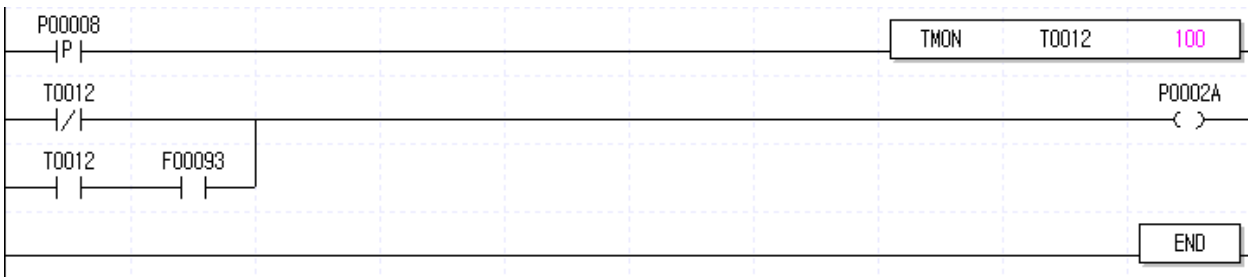
F00092	_T200MS	200ms 주기 CLOCK (100ms ON, 100ms OFF)	F0009C	_1OFF	첫 스캔 OFF
F00093	_T1S	1 초 주기 CLOCK (0.5 초 ON, 0.5 초 OFF)	F0009D	_STOG	매 스캔 반전
F00094	_T2S	2 초 주기 CLOCK (1 초 ON, 1 초 OFF)	F00110	_LER	연산 에러 (1 스캔 ON)
F00095	_T10S	10 초 주기 CLOCK (5 초 ON, 5 초 OFF)	F00112	_CARRY	연산 캐리 발생 시 ON

TMON (Mono-Stable Timer): Mono-Stable Timer 는 조건이 만족되면 설정된 시간 동안 타이머 접점이 ON 되는 타이머로 세부 동작 사항은 다음과 같습니다.

- 타이머의 기동 조건이 만족되면 현재값 = 설정값이되고, 타이머 접점이 ON 됩니다.
- 타이머의 기동 조건이 만족된 후 기동 조건의 변화와 관계없이 현재값이 설정 시간 단위로 1 씩 감소되며, 현재값 = 0 이 될 때 접점이 OFF 됩니다.
- 타이머를 리셋하면 현재값 = 0 이 되며, 타이머 접점이 ON 되어 있는 경우 접점은 OFF 됩니다.



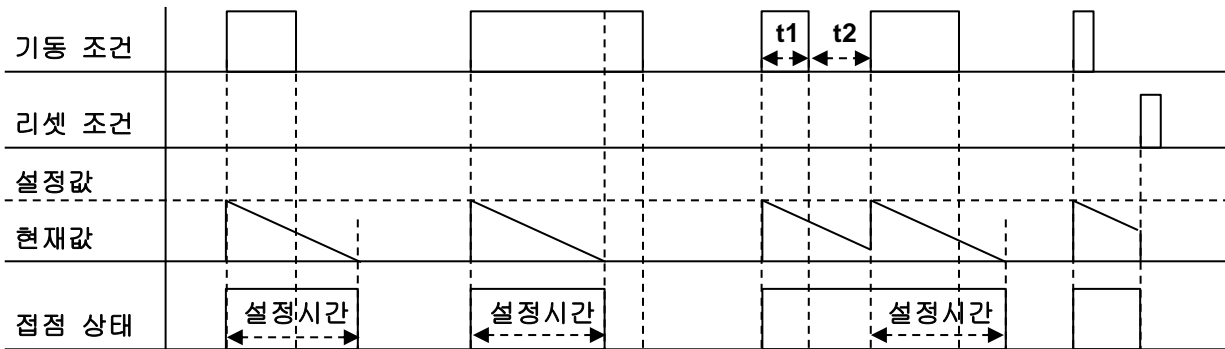
Ex 4-22) 상시 출력 P0002A LED 가 P00008 입력 스위치가 ON 되면 5 초간 점멸합니다.



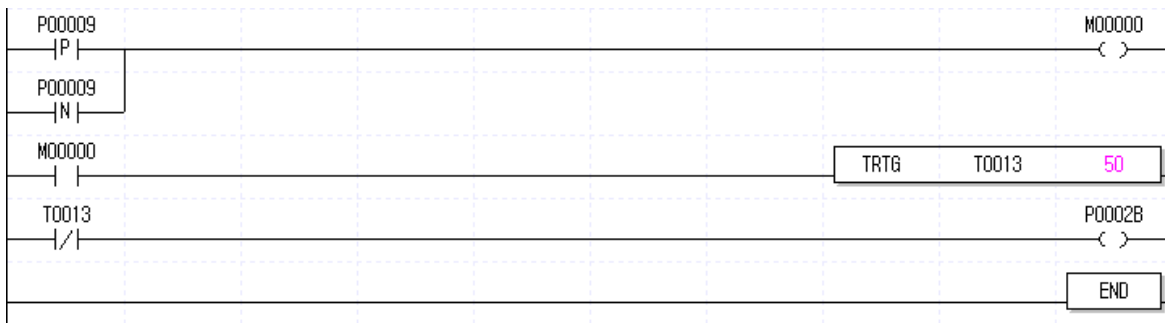
위의 프로그램과 같이 TMON 타이머는 운전 중 발생하는 이벤트에 대해 시간 인터록으로 사용할 수 있습니다.

**TRTG (리트리거블 Timer):** 리트리거블 Timer 는 조건이 만족되면 설정된 시간 동안 타이머 접점이 ON 되는 타이머로 세부 동작사항은 다음과 같습니다.

- 타이머의 기동 조건이 만족되면 현재값 = 설정값이 되고, 타이머 접점이 ON 됩니다.
- 타이머의 기동 조건이 만족된 후 설정 시간 단위로 1 씩 감소되며, 현재값 = 0 이 될 때 접점이 OFF 됩니다.
- 타이머 동작 중 타이머의 기동 조건에서 상승 에지가 발생 (OFF → ON)하면 현재값 = 설정값이 된 후 현재값이 다시 설정 시간 단위로 1 씩 감소하고 현재값 = 0 이 될 때 접점이 OFF 됩니다.
- 타이머를 리셋하면 현재값 = 0 이 되며, 타이머 접점이 ON 되어 있는 경우 접점은 OFF 됩니다.



Ex 4-23) P00009 입력 스위치가 5 초 이상 ON 또는 OFF 상태를 유지하면 P0002B LED 가 ON 됩니다.



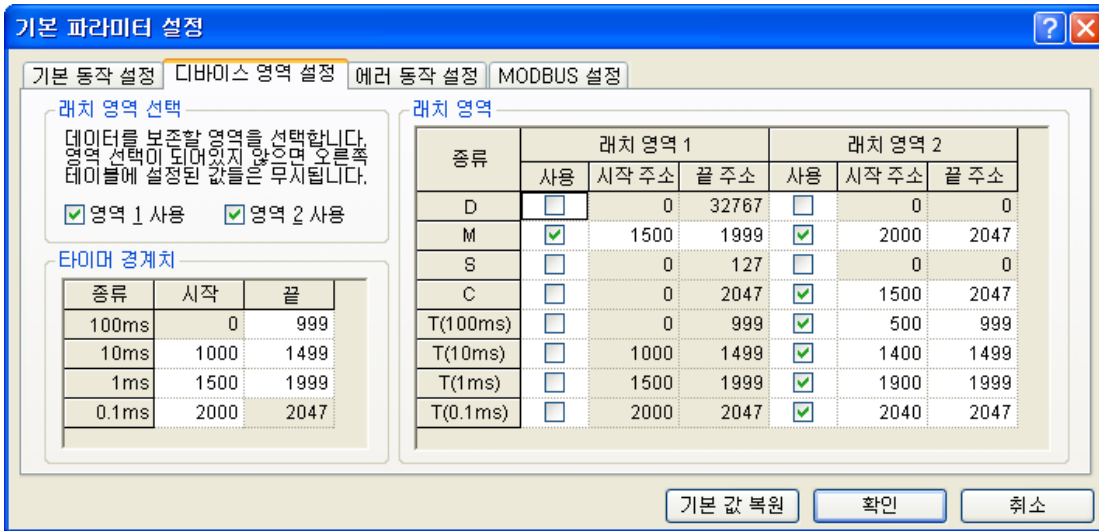
### 4.4 카운터

카운터는 조건의 만족 회수를 계수하는 프로그램 요소입니다. 즉, 조건이 만족될 때 현재값을 1 씩 증가 또는 감소 시켜 조건이 만족되는 회수를 누적합니다.

XGK PLC 에는 2,048 개의 카운터가 있으며, 기능별로 CTU, CTD, CTUD, CTR 등 4 종의 카운터가 있으며, 1 개의 카운터는 최대 65,535 까지 계수할 수 있습니다.


#### 1) 카운터의 정전 유지

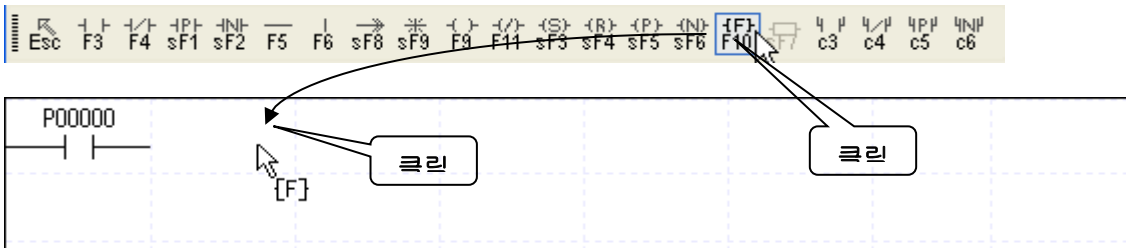
PLC 의 정전 시 카운터의 현재값이 유지되어야 하는 경우가 있습니다. 특히 장기적인 데이터의 누적이 필요한 경우 래치 영역의 카운터를 사용하는 것이 권장됩니다. 카운터 영역의 래치는 기본 파라미터에서 설정할 수 있으며, 초기 설정은 모든 영역이 휘발성으로 설정되어 있으므로 래치 카운터가 필요한 경우 사용자가 직접 래치 영역을 설정해야 합니다.



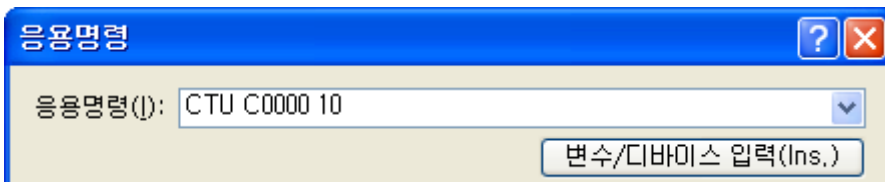
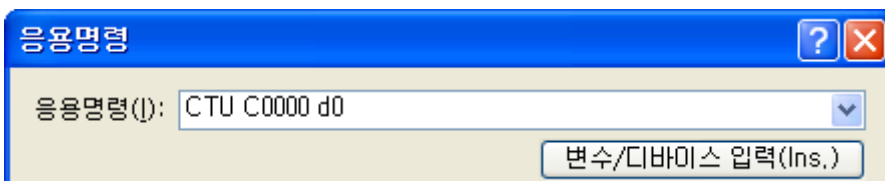
2) 카운터의 편집

CTU, CTD, CTR 편집

도구 상자에서 평선/평선블록 (  )을 선택하거나 평선/평선블록 단축키 (F10)를 누릅니다.

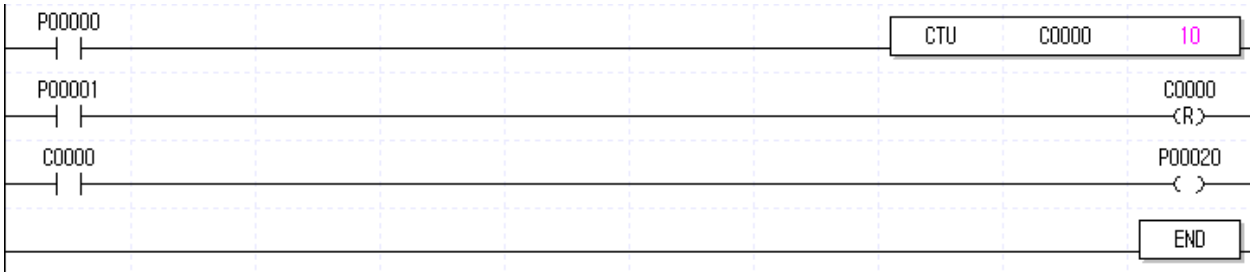


응용명령 창에서 카운터 종류, 카운터 번호, 설정값 순으로 입력합니다. 만일 카운터의 설정값을 고정값으로 하지 않고 가변적으로 설정하고자 할 경우 설정값 위치에 워드형 디바이스를 입력합니다. 카운터의 최대 설정값은 65,535 이며, HMI 등 외부에서 설정값을 설정할 경우 16 비트 부호 없는 10 진 정수 형식으로 입력해야 합니다.

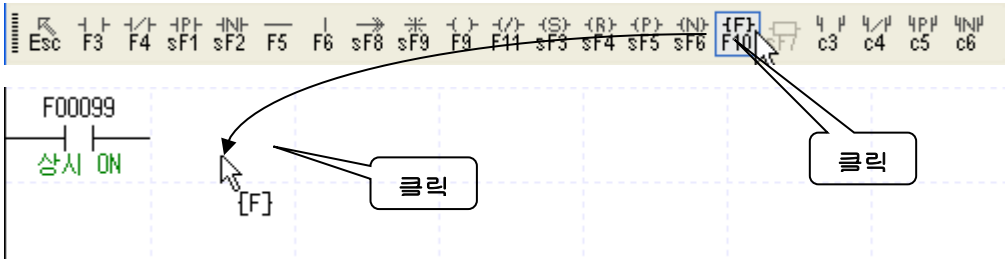


리셋 코일( )을 이용하여 카운터를 리셋하는 프로그램을 작성하고, 카운터 접점을 이용한 프로그램을 작성합니다.

본 예제에서 P00000 입력 접점을 10 번 ON 시키면 P00020 LED 가 ON 되며, P00001 입력 접점을 ON 시키면 카운터가 리셋 됩니다



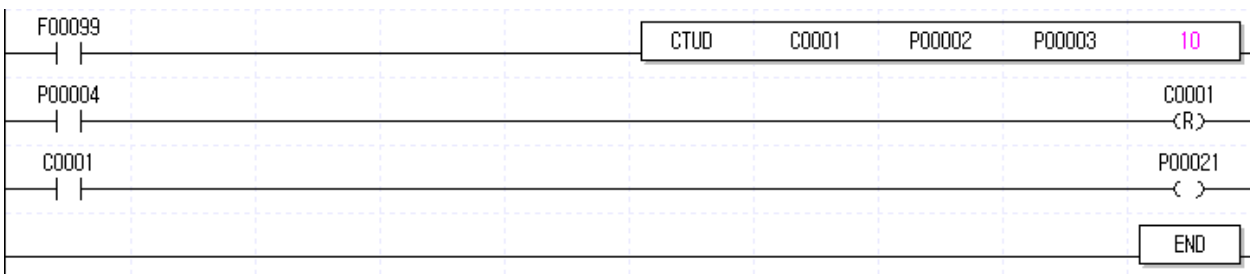
CTUD 를 편집하고자 할 경우 조건을 F00099 (상시 ON)로 사용해야 하며, 카운터 번호, 카운터 증가 조건, 카운터 감소 조건 및 설정값 순으로 편집 합니다. 아래의 편집 예에서 C0001 을 UP/DOWN 카운터로 사용했으며, P00002 가 증가 카운터, P00003 이 감소 카운터 조건으로 사용됩니다.



**응용명령** ? X

응용명령():

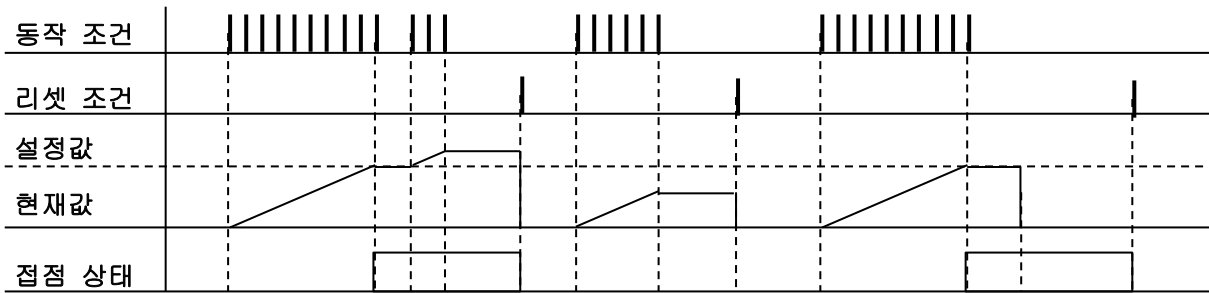
CTUD 의 경우도 리셋 코일( )을 이용하여 카운터를 리셋하는 프로그램을 작성하고, 카운터 접점을 이용한 프로그램을 작성합니다.



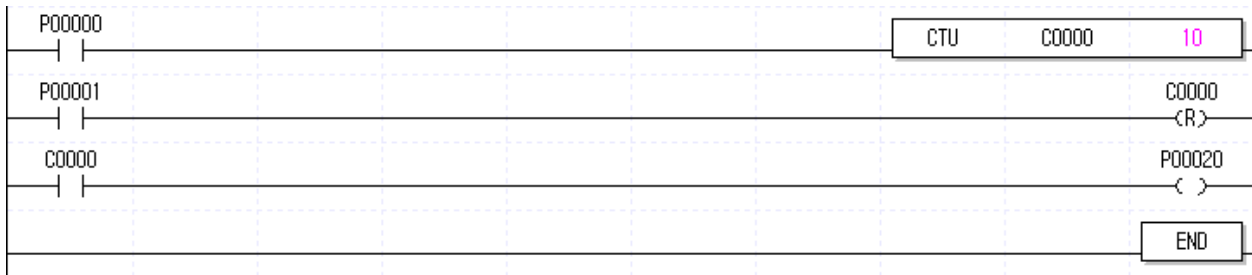
3) 카운터의 동작 특성

**CTU (UP Counter):** UP Counter 의 초기 현재값 = 0 입니다. 카운터의 동작 조건에 상승 에지가 발생(OFF → ON)할 때 마다 카운터의 현재값이 1 씩 증가되고, 카운터의 현재값 = 설정값이 될 때 카운터의 접점이 ON 되는 카운터입니다.

- 카운터의 초기 현재값 = 0 이며 동작 조건에 상승 에지가 발생하면 현재값이 1 증가 됩니다.
- 카운터의 현재값 = 설정값이 될 때 카운터의 접점이 ON 됩니다.
- 카운터의 접점이 ON 된 상태에서 동작 조건에 상승 에지가 발생하면 현재값은 1 증가 되고, 접점은 ON 상태를 유지합니다.
- 카운터의 현재값이 65,535 인 상태에서 동작 조건에 상승 에지가 발생해도 현재값은 변경되지 않습니다.
- 카운터를 리셋하면 현재값 = 0 이 되며, 카운터의 접점이 ON 되어 있는 경우 접점은 OFF 됩니다.
- 카운터의 현재값을 강제로 변경시킬 경우 접점 상태는 변경되지 않습니다.

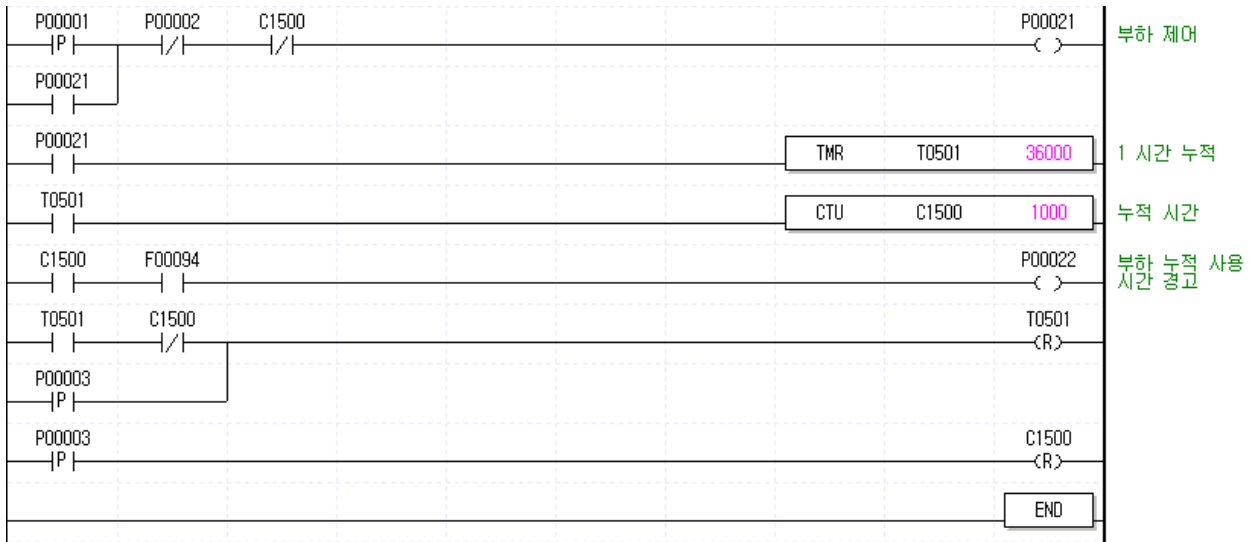


Ex 4-24) P00000 입력 스위치가 10 회 이상 ON 되면 P00020 LED 가 ON 됩니다.



디지털 입력 접점을 이용하여 펄스 입력 신호를 계수할 때, 입력되는 펄스의 ON 및 OFF 상태가 입력 필터 시간 + 1 스캔 시간 이상 유지되어야 정확한 계수가 가능합니다.

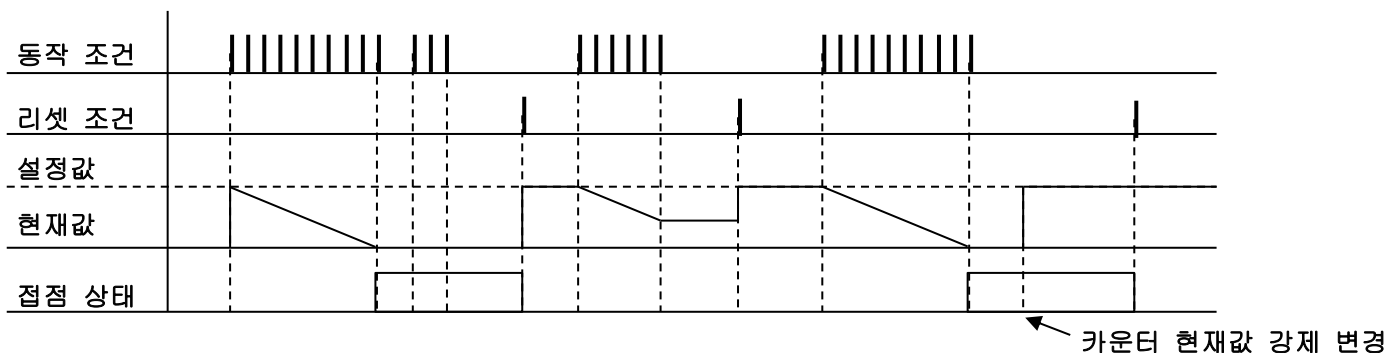
Ex 4-25) 카운터를 이용한 타이머 계측 시간 연장: P00001 입력 스위치가 ON 되면 P00021 LED 가 ON 됩니다. P00021 LED 의 누적 ON 시간이 1000 시간이 되면 P00022 LED 가 2 초 주기로 점멸합니다.



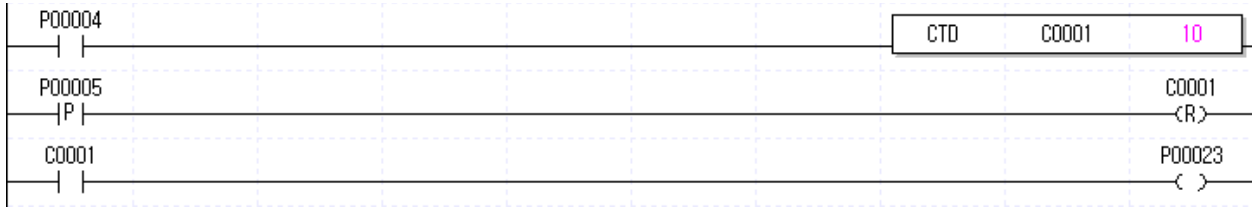
1 개의 타이머에 설정할 수 있는 최대 시간은 6,553.5 초 (1 시간 40 분 13.5 초)입니다. 타이머의 최대 설정 시간을 초과하는 시간 제어 요소가 있을 경우 위의 프로그램과 같이 타이머와 카운터를 조합하면 설정 시간을 늘릴 수 있습니다.

**CTD (DOWN Counter):** DOWN Counter의 초기 현재값 = 설정값이 됩니다. 카운터의 동작 조건에 상승 에지가 발생(OFF → ON)할 때 마다 카운터의 현재값이 1 씩 감소되고, 카운터의 현재값 = 0 이 될 때 카운터의 접점이 ON 되는 카운터입니다.

- 카운터의 초기 현재값 = 설정값이 됩니다.
- 카운터의 동작 조건에 상승 에지가 발생하면 현재값이 1 감소 됩니다.
- 카운터의 현재값 = 0 이 될 때 카운터의 접점이 ON 됩니다.
- 카운터의 접점이 ON 된 상태(현재값 = 0)에서 동작 조건에 상승 에지가 발생해도 현재 값은 변화되지 않습니다.
- 카운터를 리셋하면 현재값 = 설정값이 되며, 카운터의 접점이 ON 되어 있는 경우 접점은 OFF 됩니다.
- 카운터의 현재값을 강제로 변경시킬 경우 접점 상태는 변경되지 않습니다.

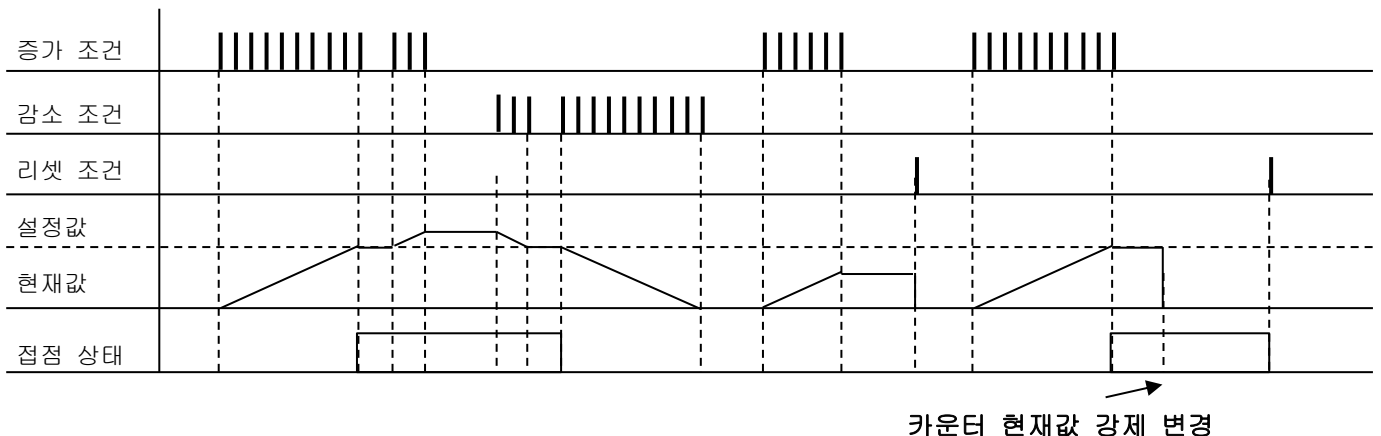


Ex 4-26) P00004 입력 스위치가 10 회 이상 ON 되면 P00023 LED 가 ON 됩니다.

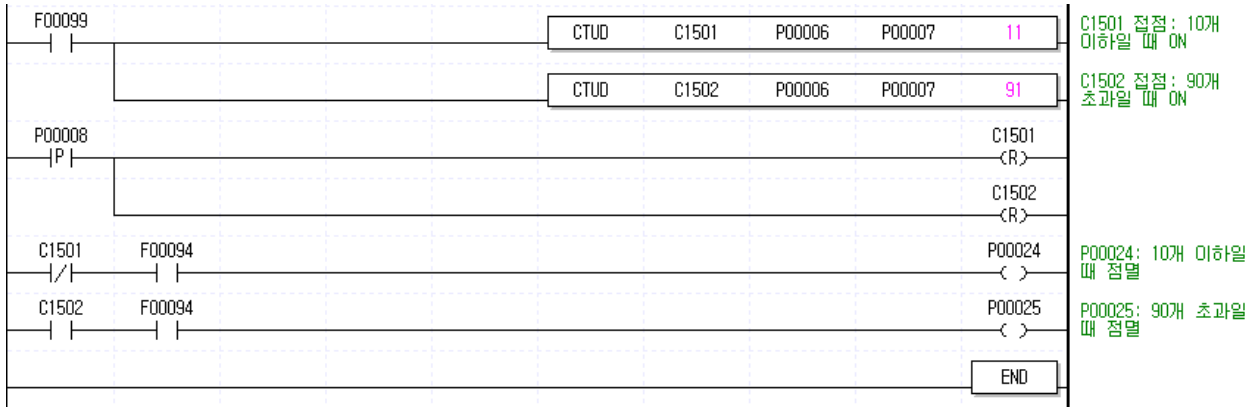


**CTUD (UP/DOWN Counter):** UP/DOWN Counter 의 초기 현재값 = 0 입니다. CTUD 의 동작 조건은 상시 ON (F00099)로 사용해야 하며, 명령어 편집 시 증가(UP) 및 감소(DOWN) 계수의 조건을 설정합니다. 증가 계수의 조건에 상승 에지가 발생(OFF → ON)할 때 카운터의 현재값이 1 씩 증가되고, 감소 계수의 조건에 상승 에지가 발생(OFF → ON)할 때 카운터의 현재값이 1 씩 감소 됩니다.

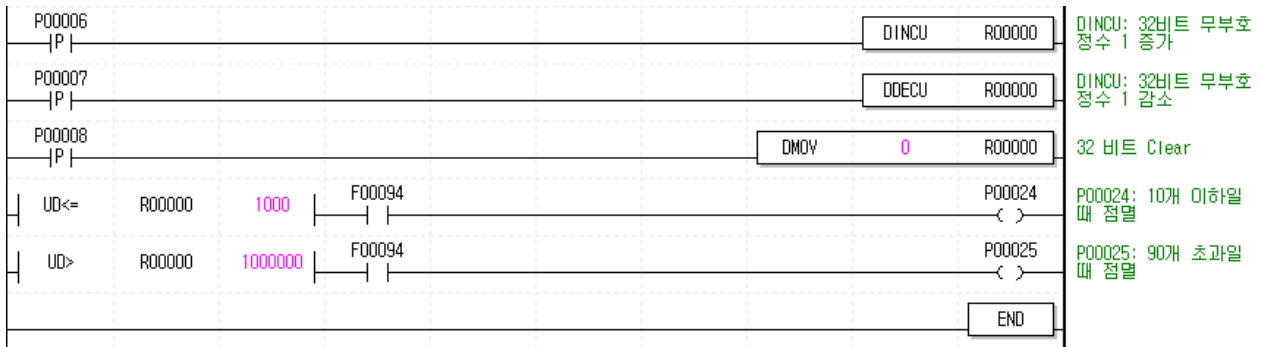
- 카운터의 초기 설정값 = 0 입니다.
- 증가 계수 조건에 상승 에지가 발생하면 현재값이 1 증가 되고, 감소 계수 조건에 상승 에지가 발생하면 현재값이 1 감소 됩니다.
- 카운터의 현재값 = 설정값이 될 때 카운터의 접점이 ON 됩니다.
- 카운터의 접점이 ON 된 상태에서 증가 계수 조건에 상승 에지가 발생하면 현재값이 1 증가되고 접점은 ON 상태를 유지합니다.
- 카운터의 현재값 = 65,535 인 상태에서 증가 계수 조건에 상승 에지가 발생해도 현재값이 변화되지 않습니다.
- 카운터의 현재값 = 0 인 상태에서 감소 계수의 조건에 상승 에지가 발생할 때 현재값은 변화되지 않습니다.
- 카운터를 리셋하면 현재값 = 0 이 되며, 카운터의 접점이 ON 되어 있는 경우 접점은 OFF 됩니다.
- 카운터의 현재값을 강제로 변경시킬 경우 접점 상태는 변경되지 않습니다.



Ex 4-27) 입고 센서 (P00006)와 출고 센서 (P00007)가 있는 창고에 재고가 10 개 이하일 때 P00024 LED 가 2 초 주기로 점멸하고, 90 개 초과할 때 P00025 LED 가 점멸합니다.

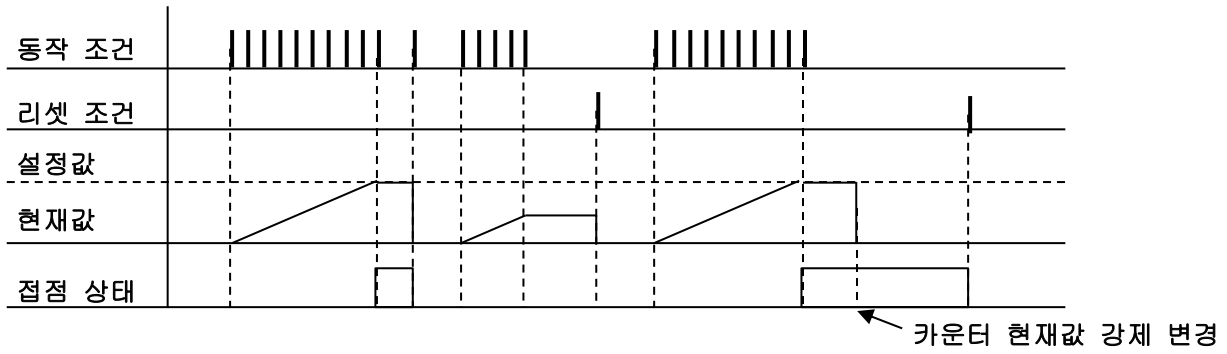


위의 프로그램에서 재고 관리 수량이 65,535 개를 초과할 경우 카운터를 사용할 수 없고 응용 명령어 중 증가/감소 명령 및 비교 연산을 사용해야 합니다.

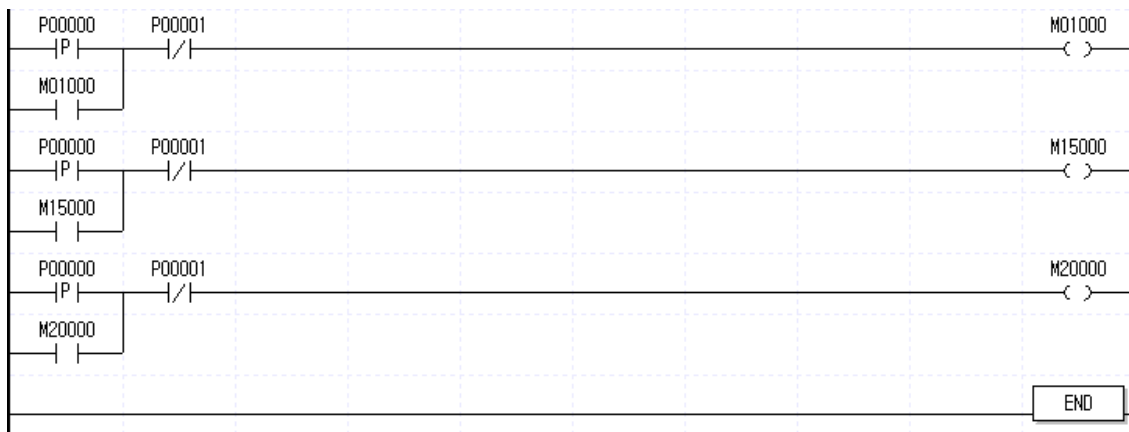
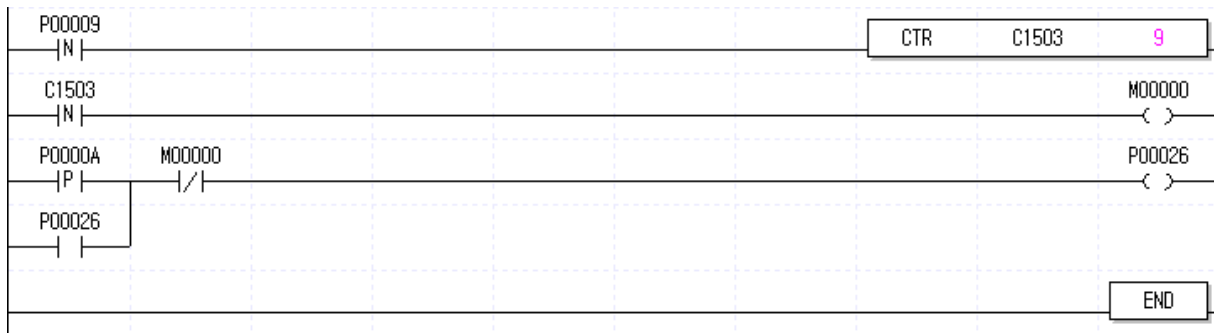
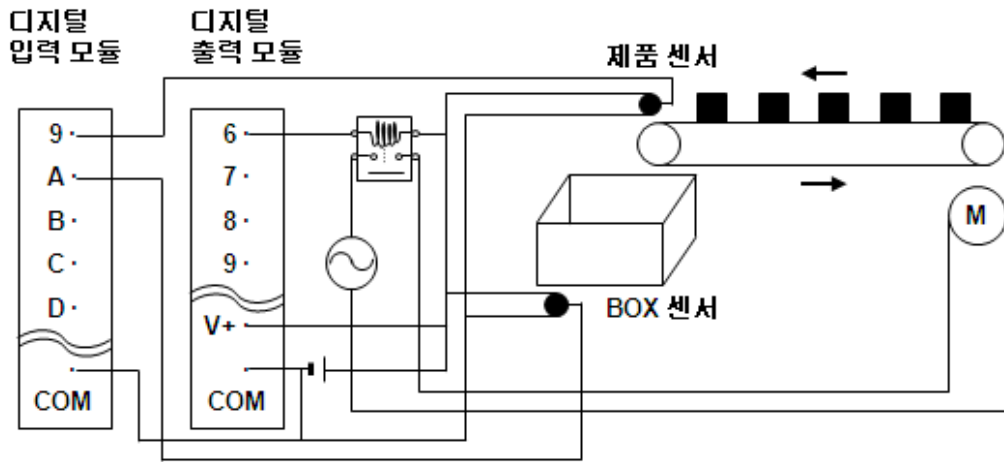


**CTR (RING Counter):** RING Counter 의 초기 현재값은 = 0 입니다. 카운터의 동작 조건에 상승 에지가 발생(OFF → ON)할 때 마다 카운터의 현재값이 1 씩 증가되고, 카운터의 현재값 = 설정값이 될 때 카운터의 접점이 ON 됩니다. 접점이 ON 된 상태에서 동작 조건에 상승 에지가 1 번 발생하면 카운터의 현재값이 0 으로 변경되고, 접점이 OFF 됩니다.

- 카운터의 동작 조건에 상승 에지가 발생하면 현재값이 1 증가 됩니다.
- 카운터의 현재값 = 설정값이 될 때 카운터의 접점이 ON 됩니다.
- 카운터의 접점이 ON 된 상태에서 동작 조건에 상승 에지가 발생하면 현재값 = 0 이 되고 접점은 OFF 됩니다.
- 카운터를 리셋하면 현재값 = 0 이 되며, 카운터의 접점이 ON 되어 있는 경우 접점은 OFF 됩니다.
- 카운터의 현재값을 강제로 변경시킬 경우 접점 상태는 변경되지 않습니다.



Ex 4-28) BOX 에 10 개의 제품이 들어가면 컨베이어가 정지하고, BOX 센서가 ON 되면 다시 컨베이어가 기동합니다.



## 부록 1. 수치체계 및 데이터구조

### 1) 수치(데이터)의 표현

PLC CPU 는 모든 정보를 On 과 Off, 또는 "1"과 "0"의 상태로 기억하고 처리합니다. 따라서 수치 연산도 1 과 0 으로 처리된 수치, 즉 2 진수 (Binary number .... BIN) 로 처리합니다.

한편, 일상 생활에서는 10 진수가 알기쉽고 가장 널리 사용되고 있습니다. 그래서 PLC 에 수치를 쓰거나 읽을 경우, 10 진수에서 16 진수로, 16 진수에서 10 진수로 변환이 필요합니다. 여기에서는 10 진수와 2 진수, 16 진수, 2 진화 10 진수(BCD)의 표현과 상호관계에 대해 설명합니다.

#### (1) 10 진수(Decimal)

10 진수란 " 0~9 의 종류의 기호를 사용하여 순서와 크기(량)를 표현하는 수"를 말합니다.

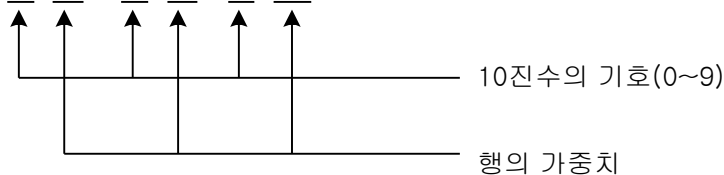
그리고 0, 1, 2, 3, 4, .....9 다음에 " 10"으로 자리올림하고 계속 진행됩니다.

예를 들면, 10 진수 153 을 행과 "행의 가중치"란 측면에서 보면 아래와 같습니다.

$$135=100+50+3$$

$$=1*100+5*10+3*1$$

$$=1*10^2 + 5*10^1 + 3*10^0$$



#### (2) 2 진수 (Binary ..... Bin)

2 진수란 "0 과 1 의 두 종류 기호를 사용하여 순서와 크기를 나타내는 수"를 말합니다. 그래서 0, 1 다음에 "10"으로 자리올림을 하고, 계속 진행됩니다.

즉, 0,1 의 한 자리 수를 비트라고 합니다.

2 진수	10 진수
0	0
1	1
10	2
11	3
100	4
101	5
110	6
111	7
1000	8
.....	.....

예를 들면 다음의 2 진수는 10 진수로 얼마나 되는지 생각해 봅시다.

“10011101”

10 진수에서 행번호와 행의 가중치를 고려하였듯이 우측부터 비트번호와 비트가중치를 붙여 봅시다.

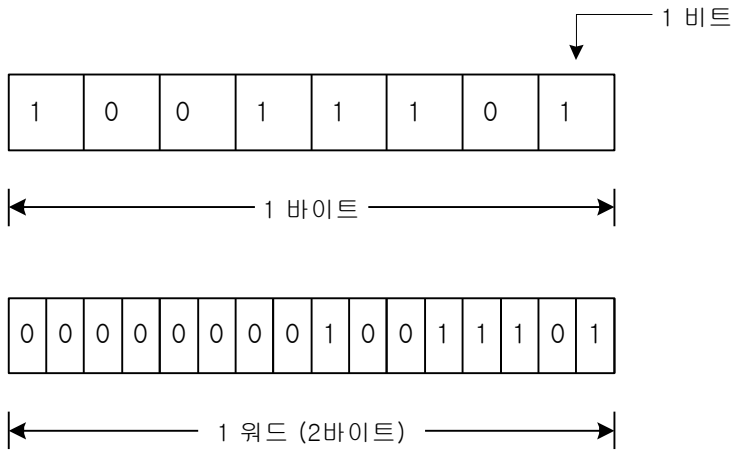
7	6	5	4	3	2	1	0	←	비트번호 2진수
1	0	0	1	1	1	0	1		
$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$		
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮		
128	64	32	16	8	4	2	1		비트의 가중치

10 진수와 같이 각 비트의 코드의 가중치의 곱의 합을 생각해 봅시다.

$$\begin{aligned}
 &= 1 \times 128 + 0 \times 64 + 0 \times 32 + 1 \times 16 + 1 \times 8 + 1 \times 4 + 0 \times 2 + 1 \times 1 \\
 &= 128 + 16 + 8 + 4 + 1 \\
 &= 157
 \end{aligned}$$

즉, 2 진수는 “코드가 1 인, 비트의 가중치를 가산한 것”이 10 진수로 되는 것입니다.

일반적으로 8 비트를 1 바이트, 16 비트 (2 바이트)를 1 워드라 말합니다.

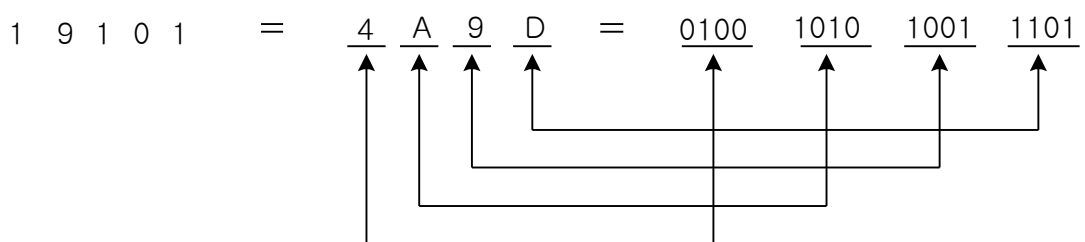


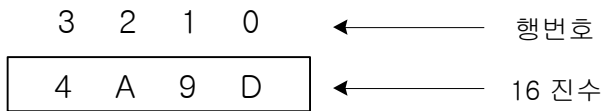
(3) 16 진수 (Hexadecimal ..... HEX)

16 진수도 10 진수, 2 진수와 동일하게 생각하여 “0 ~ 9, A ~ F 의 종류의 기호를 사용하여 순서와 크기를 나타내는 수”를 말합니다.

그리고 0, 1, 2, .....D,E,F 다음에 “10”으로 자리올림을 하고 계속 진행됩니다.

10 진수	16 진수	2 진수
0	0	0
1	1	1
2	2	10
3	3	11
4	4	100
5	5	101
6	6	110
7	7	111
8	⋮	1000
9		1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111
16	10	10000
17	11	10001
18	12	10010





$$\begin{aligned}
 &= (4) \times 16^3 + (A) \times 16^2 + (9) \times 16^1 + (D) \times 16^0 \\
 &= 4 \times 4096 + 10 \times 256 + 9 \times 16 + 13 \times 1 \\
 &= 19101
 \end{aligned}$$

16 진수의 한자리는 2 진수의 4 비트로 대응됩니다.

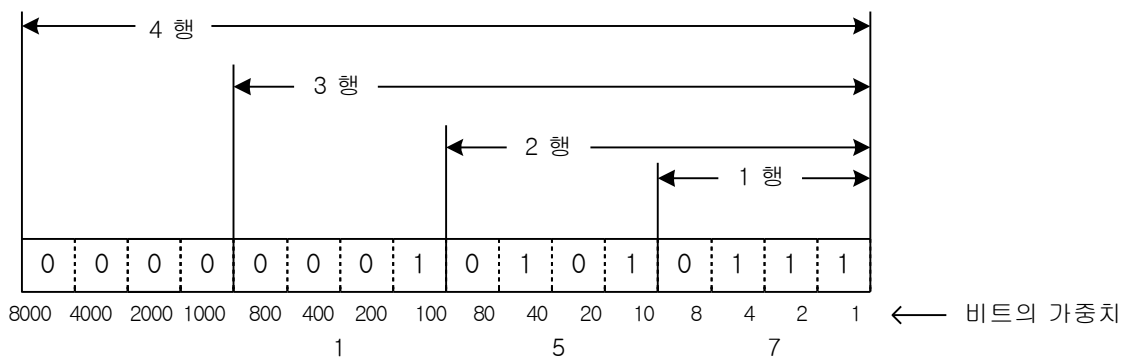
(4) 2 진화 10 진수 (Binary Coded Decimal ..... BCD)

2 진화 10 진수는 "10 진수의 각행의 숫자를 2 진수로 나타낸 수"를 말합니다.

예를 들면, 10 진수의 157 는 다음과 같이 나타낼 수 있으며,

따라서, 2 진화 10 진수는 10 진수의 0 ~ 9999 (4 행의 최대치)를 16 비트로 나타냅니다.

각 비트의 가중치는 다음과 같습니다.



(5) 수치 체계표

2 진화 10 진수 (Binary coded Decimal) BCD		2 진수 (Binary) BIN		10 진수 (Decimal)	16 진수 (Hexadecimal) H
00000000	00000000	00000000	00000000	0	0000
00000000	00000001	00000000	00000001	1	0001
00000000	00000010	00000000	00000010	2	0002
00000000	00000011	00000000	00000011	3	0003
00000000	00000100	00000000	00000100	4	0004
00000000	00000101	00000000	00000101	5	0005
00000000	00000100	00000000	00000100	6	0006
00000000	00000111	00000000	00000111	7	0007
00000000	00001000	00000000	00001000	8	0008
00000000	00001001	00000000	00001001	9	0009
00000000	00010000	00000000	00001010	10	000A
00000000	00010001	00000000	00001011	11	000B
00000000	00010010	00000000	00001100	12	000C
00000000	00010011	00000000	00001101	13	000D
00000000	00010100	00000000	00001110	14	000E
00000000	00010101	00000000	00001111	15	000F
00000000	00000110	00000000	00010000	16	0010
00000000	00000111	00000000	00010001	17	0011
00000000	00001000	00000000	00010010	18	0012
00000000	00001001	00000000	00010011	19	0013
00000000	00100000	00000000	00010100	20	0014

00000000 00100001	00000000 00010101	21	0015
00000000 00100010	00000000 00010110	22	0016
00000000 00100011	00000000 00010111	23	0017
00000001 00000000	00000000 01100100	100	0064
00000001 00100111	00000000 01111111	127	007F
00000010 01010101	00000000 11111111	255	00FF
00010000 00000000	00000000 11100000	1000	03E8
00100000 01000111	00000000 11111111	2047	07FF
01000000 10010101	00000000 11111111	4095	0FFF
10011001 10011001	00000111 00001111	9999	270F
	00100111 00010000	10000	2710
	01111111 11111111	32767	7FFF

2) 정수표현

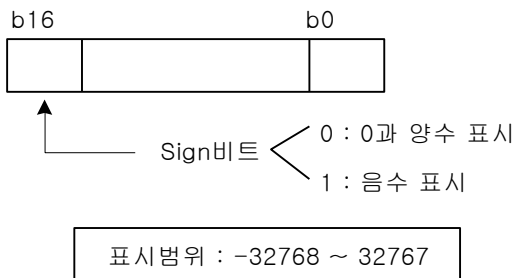
XGK 명령어에서는 음수체계연산(Signed)을 기본으로 합니다.

이때 정수표시는 최상위 비트(MSB)가 0 이되면 양수를 나타내고 1 이면 음수로 나타나게 됩니다.

음수, 양수를 표시하는 최상위 비트를 Sign 비트라고 합니다.

16 비트, 32 비트에서는 MSB 의 위치가 다르기 때문에 Sign 비트 위치에 주의해야 합니다.

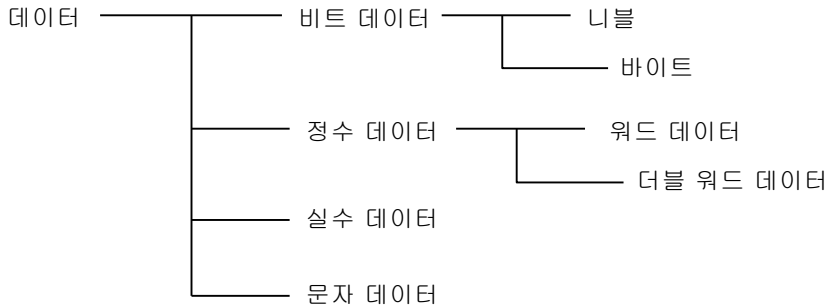
\* 16 비트 일 경우





## 부록 2 데이터의 종류 및 사용방법

### 2.1 데이터의 종류

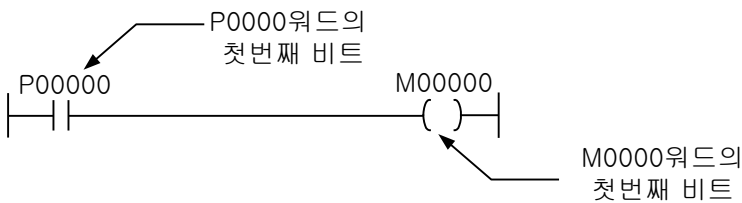
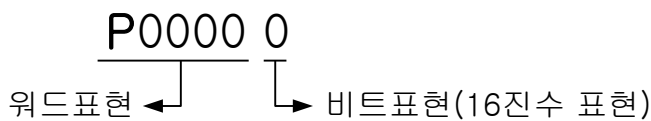


### 2.2 비트 데이터 (BIT)

비트 데이터는 접점이나 코일과 같이 1 비트로 On/Off를 표시하거나 입출력은 되지 않고 메모리 내에서 1 비트 단위로 처리되는 데이터를 말합니다. 비트 디바이스 혹은 워드 디바이스의 비트 지정 방법으로 비트 데이터를 사용할 수 있습니다.

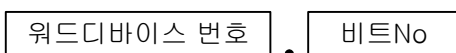
#### 1) 비트 디바이스

한 점 단위로 저장되거나 읽어올 수 있는 디바이스로 P, M, L, K, F, T, C, S 등이 있습니다(자세한 내용은 2.3 디바이스 영역 설명 참조). 비트 데이터를 액세스하기 위해 한 점(비트) 단위로 지정해서 사용합니다. 이때 가장 아래 자리는 16진수로 표기합니다. 이런 표현방식은 비트 디바이스로 워드 데이터의 표현을 쉽게 할 수 있게 합니다.

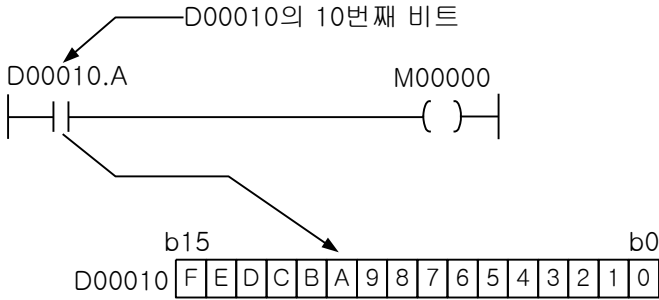


#### 2) 워드 디바이스의 비트지정 방법

워드 디바이스에 비트 No를 지정함으로써 비트 데이터를 사용할 수 있습니다. 표현 방법은 다음과 같습니다.



여기서 워드 디바이스 번호는 10 진수로 표기하고 비트 No 는 16 진수로 표기합니다. 예를 들어, D0010 의 두번째 비트를 표현 할려면, D0010.1 과 같이 지정합니다. D0011 의 b10 비트는 D0011.A 와 같이 지정합니다.



**알아두기**

비트 디바이스도 워드 디바이스처럼 워드 단위의 데이터 처리가 가능합니다. 하지만 워드 디바이스처럼 P0010.1 과 같은 표현은 사용할 수 없습니다.

2. 3 니블 / 바이트 데이터 (NIBBLE / BYTE)

니블과 바이트는 XGT 에 새로 추가된 데이터 종류로서 각각의 명령어 이름 뒤에 4 나 8 이 붙은 명령어에서 사용되는 데이터입니다.

니블과 바이트의 시작 비트를 입력함으로써 사용할 수 있고 입력한 접점부터 4/8 비트가 처리할 데이터가 됩니다.

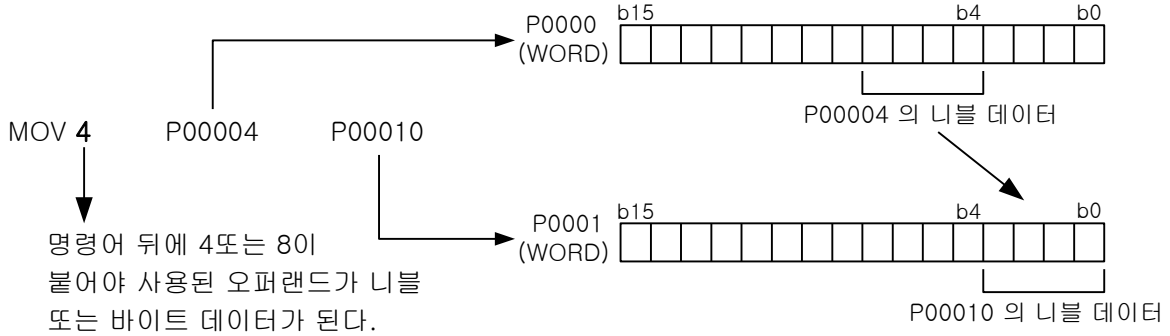
1) 표현 범위

니블: 0~15 (4 비트)

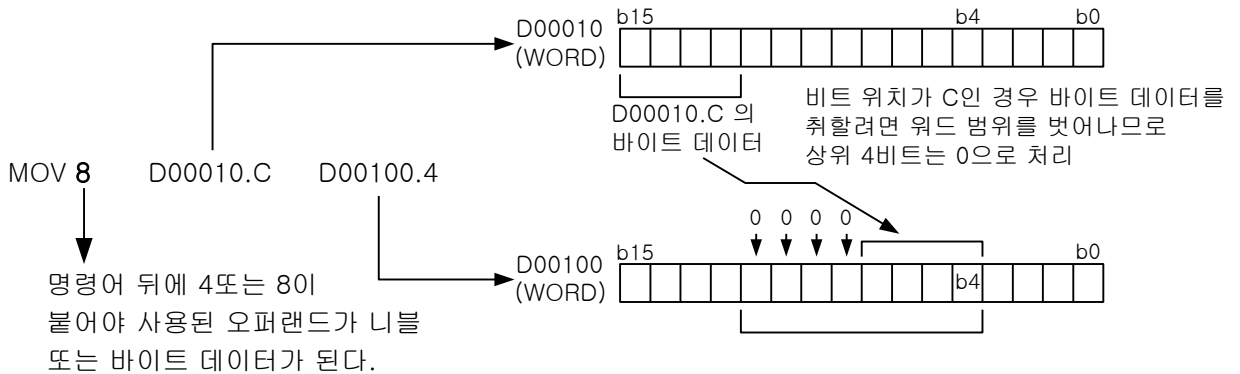
바이트: 0~255 (8 비트)

2) 사용 방법

(1) 비트 디바이스(P,M,K,F,L)의 경우: 오퍼랜드로 사용된 비트 디바이스의 접점부터 4 또는 8 비트를 취합니다. 이때, 4 비트나 8 비트를 취할 때 해당 비트 디바이스의 영역을 넘어갈 경우에는 넘어가는 비트만 0으로 처리합니다. 만약 Destination으로 지정된 오퍼랜드였다면, 영역을 넘어가는 부분의 데이터는 소실됩니다.



(2) 워드 디바이스의 경우: 오퍼랜드로 사용된 워드 디바이스의 비트접점부터 4 또는 8 비트를 취합니다. 이때, 지정한 비트 접점이 Source로 사용되었고 지정한 접점부터 4 비트나 8 비트를 취할 때 워드 단위를 넘어가게 되면 넘어간 비트에 대해서는 0으로 처리합니다. 마찬가지로 지정한 비트 접점이 Destination으로 사용되었다면 워드를 넘어가는 데이터는 소실됩니다.



**알아두기**

(1) T, C 디바이스는 니블/바이트 명령어에 사용할 수 없습니다. T와 C는 사용되는 명령어에 따라 비트 데이터로 사용되기도 하고 워드 데이터로 사용되기도 하기 때문에 사용상 혼란의 우려가 있기 때문에 니블/바이트 명령어에서는 사용이 불가능합니다.

2.4 워드 데이터 (WORD)

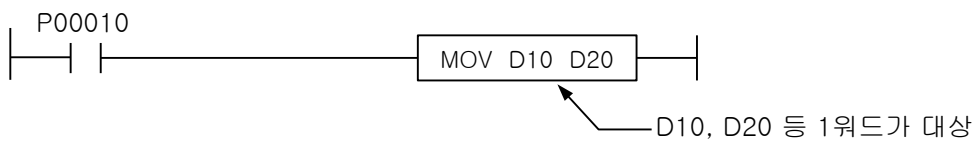
워드 데이터는 16 비트의 수치 데이터를 말합니다. 표기방법은 10 진수와 16 진수로 할 수 있고, 16 진수로 표기할 경우에는 숫자 앞에 H 를 붙입니다.

- 10 진수: -32,768 ~ 32,767 (Signed 연산) 또는 0 ~ 65,535 (Unsigned 연산)
- 16 진수: H0 ~ HFFFF

워드 디바이스나 비트 디바이스로 워드 데이터 표현이 가능합니다.

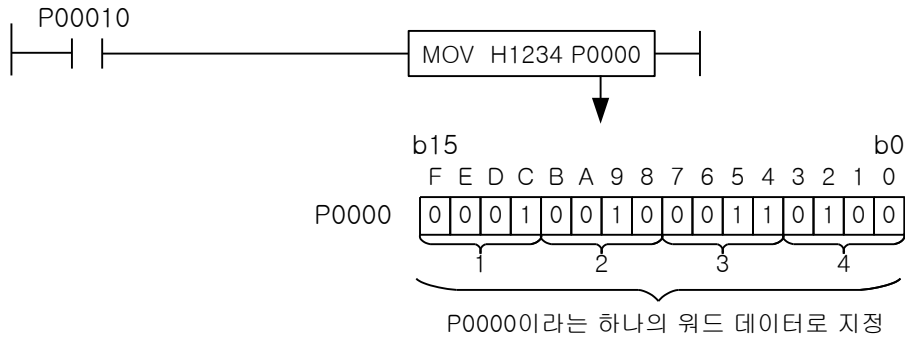
1) 워드 디바이스

워드 디바이스의 1 점(워드) 단위로 지정합니다.



2) 비트 디바이스

비트 디바이스 표기법에서 가장 하위 자리( 16 진수로 표기하는 자리 - 비트를 나타내는 위치 ) 를 빼고 표기하면 워드 데이터로 지정하게 됩니다.



**알아두기**

XGK/XGB 의 명령어는 Signed 연산을 기본으로 하고 있습니다. Unsigned 연산인 경우 명령어에 U 가 붙습니다.

예) ADD: Signed 연산

ADDU: Unsigned 연산

2.5 더블워드 데이터 (DWORD)

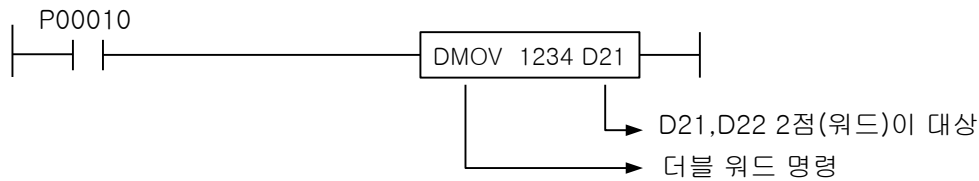
더블워드 데이터는 32 비트의 수치 데이터를 말합니다. 표기방법은 10 진수와 16 진수로 할 수 있고, 16 진수로 표기할 경우에는 숫자 앞에 H 를 붙입니다.

- 10 진수: -2,147,483,648 ~ 2,147,483,647 (Signed 연산) 또는  
0 ~ 4,294,967,295 (Unsigned 연산)
- 16 진수: h0 ~hFFFFFFF

워드 디바이스나 비트 디바이스로 더블워드 데이터 표현이 가능합니다.

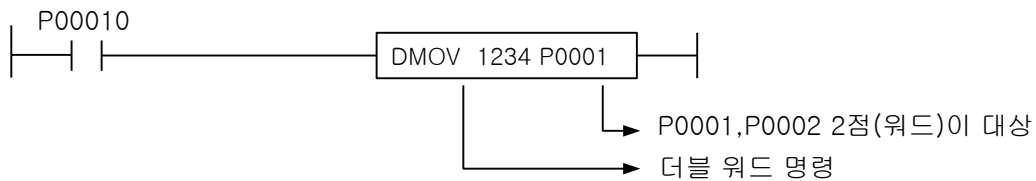
1) 워드 디바이스

32 비트 데이터 중 하위 16 비트 데이터에 해당하는 디바이스 번호를 지정합니다.  
(지정한 디바이스 번호)와 (지정한 디바이스 번호 + 1)의 데이터를 더블워드 데이터로 사용하게 됩니다.



2) 비트 디바이스

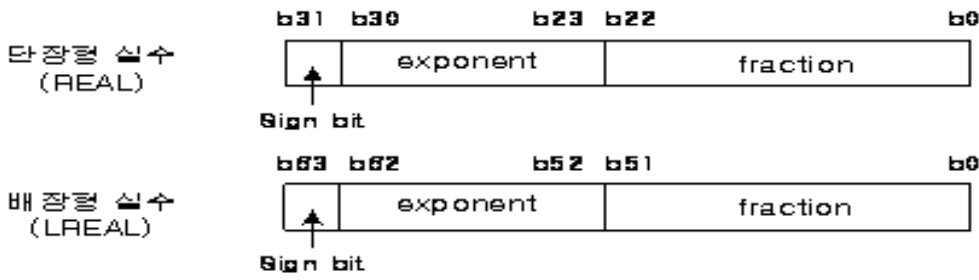
워드 데이터를 표기할 때와 마찬가지로 가장 하위 자리를 빼고 표기하며, (지정된 디바이스 번호)와 (지정된 디바이스 번호 + 1)의 데이터를 더블워드 데이터로 사용하게 됩니다.



2.6 실수 데이터 (REAL, LREAL)

실수 데이터는 32 비트/64 비트 부동 소수점 데이터를 말합니다. 여기서 32 비트 부동 소수점 데이터를 단장형 실수, 64 비트 부동 소수점 데이터를 배장형 실수라고 합니다.

표기방법은 10 진수형태(소수점 표현)로만 가능합니다. 그리고, 워드 디바이스와 비트 디바이스 모두 사용 가능합니다.



(1) 표현범위

단장형 실수: - 3.402823466e+038 ~ -1.175494351e-038 또는

0 또는 1.175494351e-038 ~ 3.402823466e+038

배장형 실수: - 1.7976931348623157e+308 ~ -2.2250738585072014e-308 또는

0 또는 2.2250738585072014e-308 ~ 1.7976931348623157e+308

(2) 지원되는 연산명령

사칙연산, 변환, 비교, 삼각함수 등의 명령어를 지원합니다.

(3) 표현 불가능한 부분이 존재합니다.(음수 부분도 대칭적으로 표현 불가능한 부분이 존재합니다.)

단장형 실수: Unsigned 0 ~ 1.40129846e-45  
Signed -1.175494351e-038 ~ 1.175494351e-038

배장형 실수: Unsigned 0 ~ 4.9406564584124654e-324  
Signed -2.2250738585072014e-308 ~ 2.2250738585072014e-308

※ 부동소수점 연산 예러: IEEE754 Standard 에서 지원하는 Exception( 연산 예러 )

플래그	명칭	연산 예러 조건	비고
F00570	부정확연산예러 래치	연산결과가 표현범위의 한계로 정확도가 소실된 경우	
F00571	언더플로우 래치	연산결과가 정규화 수 최소 절대값 미만 일 경우	
F00572	오버플로우 래치	연산결과가 정규화 수 최대 절대값 초과 일 경우	
F00573	0 나누기에러 래치	피제수가 0 이 아닌 유한한 값이고 제수가 0 일 경우	
F00574	무효연산예러 래치	연산 과정에서 부적절한 연산수행의 경우	
F0057A	부정확연산예러	연산결과가 표현범위의 한계로 정확도가 소실된 경우	
F0057B	언더플로우	연산결과가 정규화 수 최소 절대값 미만 일 경우	

F0057C	오버플로우	연산결과가 정규화 수 최대 절대값 초과 일 경우	
F0057D	0 나누기에러	피제수가 0 이 아닌 유한한 값이고 젯수가 0 일 경우	
F0057E	무효연산에러	연산 과정에서 부적절한 연산수행의 경우	
F0057F	비정규화값입력 에러	비정규화 입력이 들어왔을 경우	

**알아두기**

- (1) 실수 데이터를 표기할 때, IEEE754 형식을 만족하고 있으나, 그 형식으로 직접 입력하는 것은 불가능합니다.
- (2) XGB 의 경우 연산 에러 조건을 만족하여도 해당 플래그를 셋(SET)하지 않습니다.

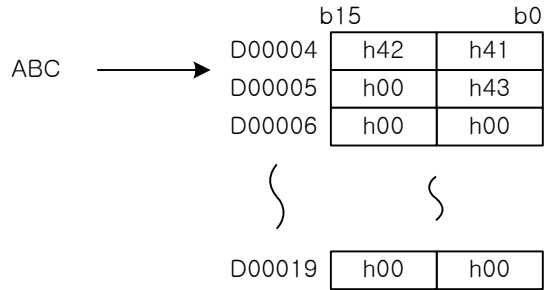
2.7 문자 데이터

응용 명령 중에 문자열 관련 명령어에서 사용 가능한 데이터 타입으로 숫자, 알파벳, 특수기호 등을 아스키 코드의 형태로 저장합니다. 또한 한글, 한자 등 16 비트 코드를 요하는 문자열도 사용 가능합니다.

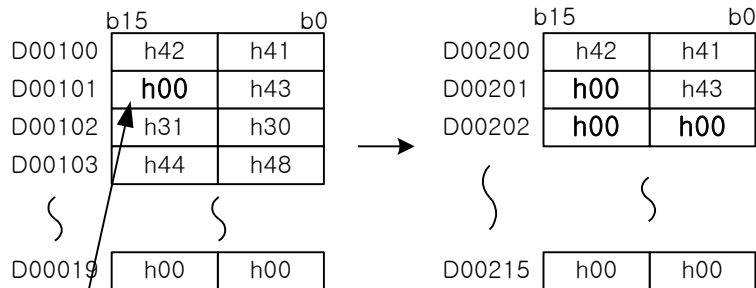
문자 데이터의 구분은 NULL 코드(h00)가 나올 때까지를 하나의 문자열로 취급하며 한 문자열의 최대 길이는 32 바이트(NULL 포함)까지 사용 가능합니다. 즉, 영문만 사용할 때는 31 글자, 국문만 사용하면 15 글자까지 사용가능하며 혼합해서도 사용 가능합니다.

직접 입력하는 문자열의 크기가 최대 크기를 넘어갈 경우, 프로그래밍 틀인 XG5000 에서 경고 메시지가 발생하여 최대 크기를 넘어가는 문자열을 입력할 수 없습니다. 최대 문자 입력의 경우 데이터 구조는 31 바이트 + NULL(1 바이트) 입니다.

사용 예)



\* \$MOV 명령어 사용시 문자열은 D00004~D00019까지  
무조건 16워드(31글자+null) 크기로 전송됩니다.



\* 문자열 사이에 null코드가 있을 경우에는  
null코드까지를 유효한 문자열로 보고 처리됩니다.

### 부록 3. XGK 추가 디바이스영역

#### 3.1 아날로그 데이터 레지스터 U

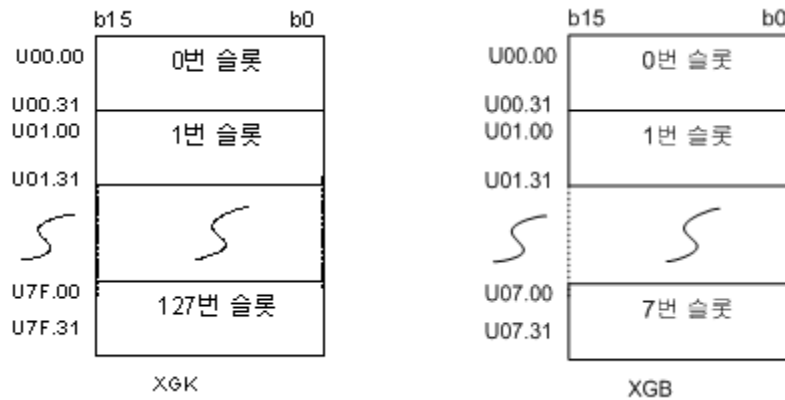
슬롯에 장착된 특수모듈(A/D,D/A,HSC 등)로부터 데이터를 읽어오는데 사용되는 레지스터입니다.

백플레인 컨트롤러에 의해 장착된 특수모듈의 데이터가 리프레시 영역에 자동으로 갱신됩니다.

U 영역은 한 슬롯당 32 개 워드가 할당되어 있습니다.

따라서 U 영역은 XGK 의 경우 총 4,096 워드( 8 베이스 \* 16 슬롯 \* 32 워드 = 4,096 워드)로 XGB 의 경우 총 256 워드( 1 베이스 \* 16 슬롯 \* 32 워드 = 256 워드) 로 이루어져 있습니다.

각 슬롯에 따라 사용하는 U 영역 값은 고정된 값으로, 모듈이 장착된 슬롯이건 빈 슬롯이건 상관 없이 고정된 값으로 사용됩니다.

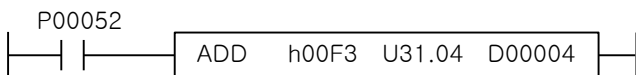


U 영역의 기본적인 표현방법은 U<sub>xy.z</sub>로 표현하며 여기서 x는 베이스번호 0~ 7를 설정하고, y는 슬롯번호인 0 ~ F를 설정하며, z는 특수모듈 내부메모리의 워드번호를 설정합니다.

U 영역 또한 비트표현이 가능하며 이때 표현 방법은 U3A.12.x (x:비트위치, 16 진수표기)

실제로 지정된 슬롯에 특수모듈이 없거나 유효한 데이터 영역을 벗어나게 지정했을 경우, 그 지정된 영역의 값은 0이 되고 에러는 발생하지 않습니다.

예를 들어, 3번 베이스의 1번 슬롯에 장착된 특수모듈의 리프레시 영역이 4개 (0번~3번)의 워드까지만 유효한 영역이라면, 4번워드 (U31.04)는 0으로 읽혀지게 됩니다. 따라서 D00004에는 h00F3이 저장됩니다.



장착된 특수모듈의 리프레시 영역 이외의 값을 읽거나 쓰고자 할 경우에는 PUT(P), GET(P)명령을 사용하십시오. 각 모듈의 영역에 대한 정보는 해당 모듈 설명서를 참조하십시오.

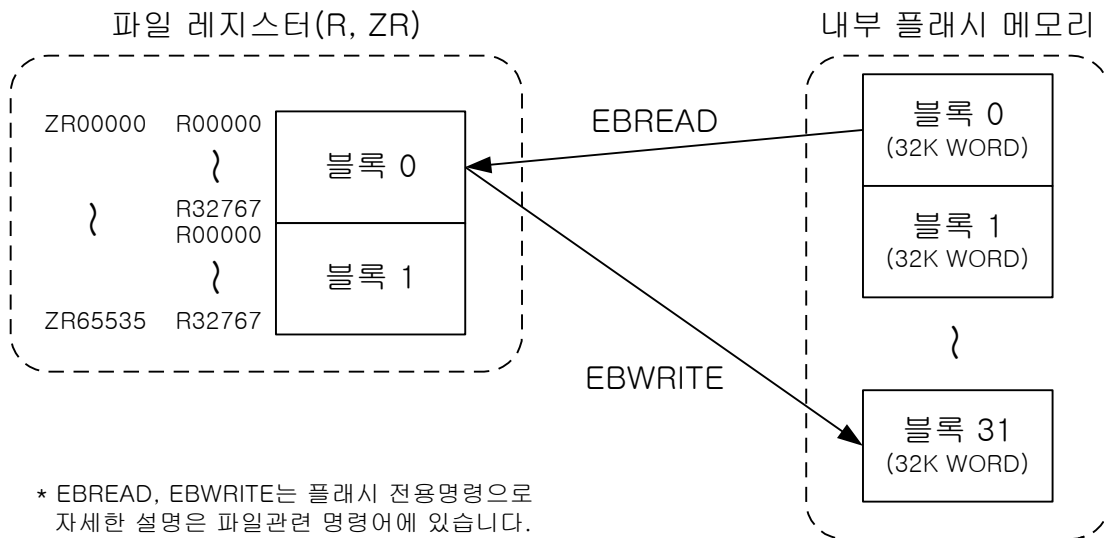
장착된 모듈이 D/A 변환모듈일 경우에는 해당 모듈의 U 영역에 데이터를 쓰면 스캔 END에서 리프레시되어 출력됩니다. 만약, D/A 변환모듈이 아닌 모듈이 장착된 위치에 데이터를 저장하는 명령어를 사용했을 경우에는 NOP 처리 합니다. 이 경우 에러가 발생하지 않습니다.

### 3.2 파일 레지스터 R

파일 레지스터는 내부 플래시 메모리 사용을 위한 전용 레지스터입니다. 플래시 메모리는 데이터를 저장하는데 약간의 시간이 소요되어 스캔 프로그램 수행중에 데이터를 저장할 수 없습니다. 이런 문제 때문에 플래시 메모리 데이터를 파일 레지스터로 옮겨서 스캔 프로그램에서 사용하고, 데이터 저장이 필요할 경우 다시 플래시 메모리로 저장하는 방식으로 사용됩니다.

#### 1) 특징

(1) 내부 플래시 메모리 사용을 위한 전용 레지스터로 내부 플래시의 블록 한 개를 파일 레지스터의 블록으로 읽어오거나 쓰는데 사용됩니다.



\* EBREAD, EBWRITE는 플래시 전용명령으로 자세한 설명은 파일관련 명령어에 있습니다.

- (2) 한 블록<sup>\*1)</sup>의 크기는 32K 워드(XGK), 10K 워드(XGB 콤팩트형)로 내부 플래시 메모리의 블록 크기와 같습니다.
- (3) 파일 레지스터의 데이터를 EBWRITE 명령을 이용해 플래시 메모리에 WRITE 하면 데이터를 영구적으로 보존할 수 있습니다.
- (4) 파일 레지스터는 래치영역 1 과 동일하게 동작합니다. 즉, 리셋 스위치를 통한 Overall 리셋, D.CLR 키를 이용한 리셋, XG5000 을 통한 리셋시에 데이터가 0 으로 지워집니다.
- (5) 파일레지스터의 블록을 플래시 메모리의 블록으로 READ/WRITE 하기 위해서는 여러 스캔이 걸립니다. 완료 여부는 F160(\_RBLOCK\_RD\_FLAG), F162(\_RBLOCK\_WR\_FLAG)의 해당 블록의 비트로 확인할 수 있습니다.

\*1) XGK-CPUS, XGK-CPUE, XGB 콤팩트형 (XBC-DxxH)의 경우 파일레지스터는 1 개의 블록(블록 0)만 제공됩니다. XGK-CPUU, XGK-CPUH, XGK-CPUA 는 2 개의 블록이 제공됩니다. 내부 플래시 메모리는 XGK 에서 32 블록, XGB 콤팩트형에서 2 블록이 제공됩니다.

※ XGB 표준형의 경우 파일 레지스터(R)를 제공하지 않습니다.



4) 에러 플래그

번호	크기	명칭	내용	비고
F158	Word	블럭 No.플래그	현재 사용중인 블럭 No 를 표기	
F1590	BIT	플래시 블록 읽기 대표플래그	플래시 블록 읽기 플래그가 하나라도 ON 되어 있으면 ON	
F1591	BIT	플래시 블록 쓰기 대표플래그	플래시 블록 쓰기 플래그가 하나라도 ON 되어 있으면 ON	
F1592	BIT	플래시 블록 쓰기 에러 대표플래그	플래시 블록 쓰기 에러 플래그가 하나라도 ON 되어 있으면 ON	
F1600 ~ F161F	BIT	플래시 블록 n 읽기 플래그	블록 n 에 데이터를 읽을 때 ON	32 개
F1620 ~ F163F	BIT	플래시 블록 n 쓰기 플래그	블록 n 에 데이터를 쓸 때 ON	32 개
F1640 ~ F165F	BIT	플래시 블록 n 쓰기 에러 플래그	블록 n 에 데이터 쓰기를 실패했을 때 ON. 에러 발생시 해당 쓰기 플래그 및 쓰기 대표 플래그도 ON 상태 유지	32 개

**알아두기**

(1) R, ZR 디바이스는 XGB 표준형에서 제공되지 않습니다.

따라서 플래시 읽기 쓰기를 지원하지 않으며 해당 에러 플래그도 지원하지 않습니다.

3.3 인덱스 레지스터 Z

## 특징

- 인덱스 레지스터를 사용해서 디바이스를 설정하는 방법으로 시퀀스 프로그램에서 인덱스 기능을 사용하면, 사용되는 디바이스는 직접 지정하는 디바이스 번호에 인덱스 레지스터의 값을 더한 위치를 가지게 됩니다. 예를 들어, P10[Z1]을 사용했을 경우, Z1의 내용이 5였다면, P(10+5) = P15가 사용대상이 됩니다.
- 인덱스 레지스터 Z0 ~ Z127 (128 개)
- 지정할 수 있는 값의 범위: -32768~32767
- 워드/비트 디바이스의 인덱스 기능
- 간접지정에도 사용가능
- 인덱스 결과 영역 초과시 연산 에러 플래그 SET (F110)

## 사용 가능 디바이스

- 비트 디바이스: P, M, L, K, F, T, C
- 워드 디바이스: Z, U, D, R, N, T의 현재값, C의 현재값

## 사용방법

- 사용하고자 하는 오퍼랜드 뒤에 []을 이용하여 사용.
- 비트 디바이스의 경우: 해당 명령어에 사용되는 오퍼랜드의 종류(비트/워드)에 따라 비트/워드 단위로 인덱스 처리.

예 1) LOAD P10[Z1]: 만약 Z1의 값이 5 라면 LOAD P(10+5) → LOAD P15(비트)가 됩니다.

예 2) MOV P10[Z1] D10: 여기서 P10은 워드를 의미하므로 P10[Z1]은 P(10+5) = P15 워드가 됩니다.

- 워드 디바이스의 경우: 워드 단위로만 인덱스 처리. 절대 비트 단위 인덱스 처리 불가능.

예) LOAD D10[Z1].5: 만약 Z1의 값이 5 라면, LOAD D(10+5).5 → LOAD D15.5 가 됩니다.

주의) LOAD D10.5[Z1]과 같은 표현은 사용할 수 없습니다.

- 이 인덱스 기능은 배열의 의미를 갖는 변수에서, 인덱스로 지정된 변수 값을 취하거나 인덱스로 지정된 변수에 값을 저장할 때 유용하게 사용할 수 있습니다.

- 프로그램 예제

D10의 값이 200일 때 다음과 같은 결과가 나옵니다.

