

PLC의 기초



PLC의 모습



PLC의 모습



PLC의 정의 및 적용분야



PLC의 정의



PLC(Programmable Logic Controller) -

- 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 장치

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



PLC 탄생배경



- 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 탄생
- 10가지 조건을 제시하고 이에 가장 부합되는 기기를 만들어 줄 것을 주문



PLC 탄생배경



- 1) 프로그램이 가능하고 프로그램의 작성과 변경이 용이할 것.
- 2) 열악한 산업 환경에서도 작동 가능 할 것
- 3) 입력은 AC 120V 신호 입력이 가능할 것
- 4) 출력은 액추에이터를 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



적용분야



설비의 자동화와 고능률화 요구에 따라 PLC 의 적용 범위는 확대

4차산업요구인 공장 자동화와 FMS(Flexible Manufacturing System)에 따른 PLC 의 요구는 과거 중규모 이상의 릴레이 제어반 대체 효과에서 현재 고기능화, 고속화의 추세로 소규모 공작 기계에서 대규모 시스템 설비에 이르기 까지 적용



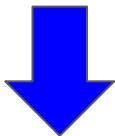
적용분야



분야	제어 대상
식료 산업	컨베이어 총괄 제어, 생산라인 자동 제어
제철, 제강 산업	작업장 하역 제어, 원료 수송 제어, 압연 라인 제어, 하역 운반 제어
섬유, 화학공업	원료 수입 출하 제어, 직조 염색 라인 제어
자동차 산업	전송 라인 제어, 자동 조립 라인 제어, 도장 라인 제어, 용접기 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

PLC 기능 요약

기존 릴레이 타이머, 카운터 등의 기능을 IC와 트랜지스터 등의 반도체 소자로 대체, 기본적인 시퀀스 제어 기능에수치 연산, 아날로그, 고속카운터, PID제어, 모터 제어,통신 기능 등을 추가하여 사용자의 의도에 따라 프로그램 제어가 가능하도록 한 노리 제어 장치



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

PLC의 구조



하드웨어 구조 - 구성

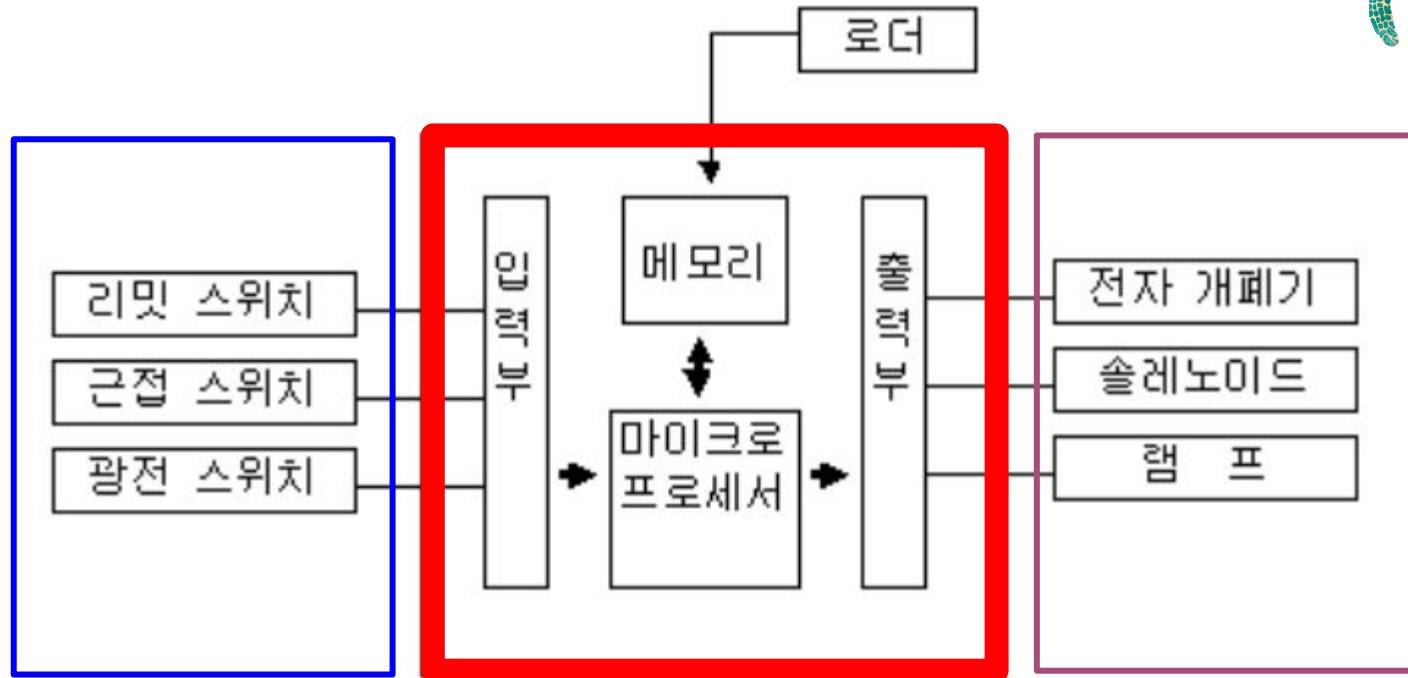


- 구성 - PLC는 마이크로프로세서(microprocessor) 및 메모리를 중심으로 구성되어 인간의 두뇌 역할을 하는 중앙처리장치(CPU), 외부 기기와의 신호를 연결시켜 주는 입·출력부, 각 부에 전원을 공급하는 전원부, PLC내의 메모리에 프로그램을 기록하는 주변 장치로 구성





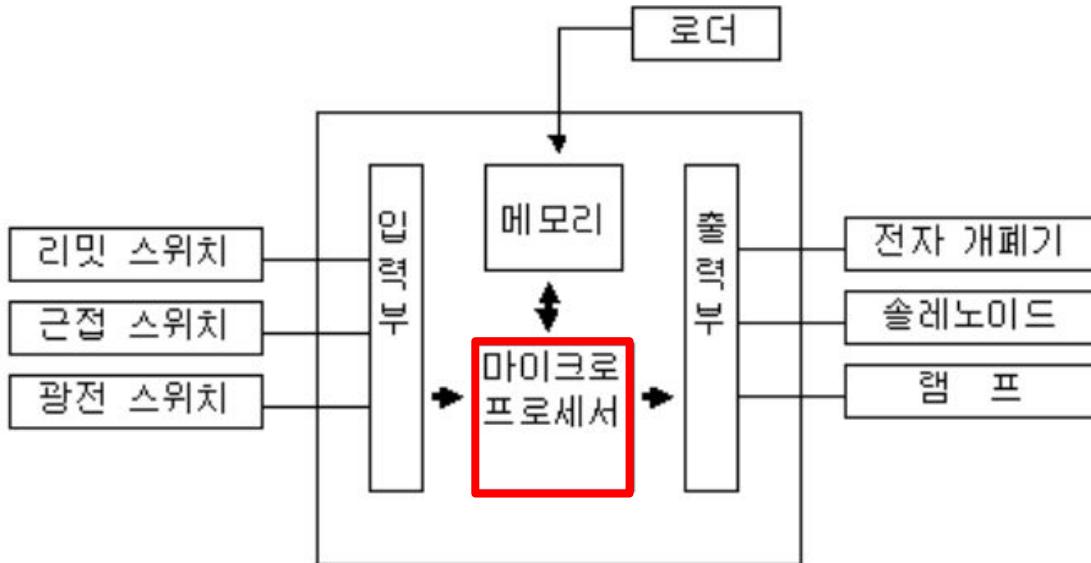
하드웨어 구조 - 구성





하드웨어 구조 - CPU 연산부

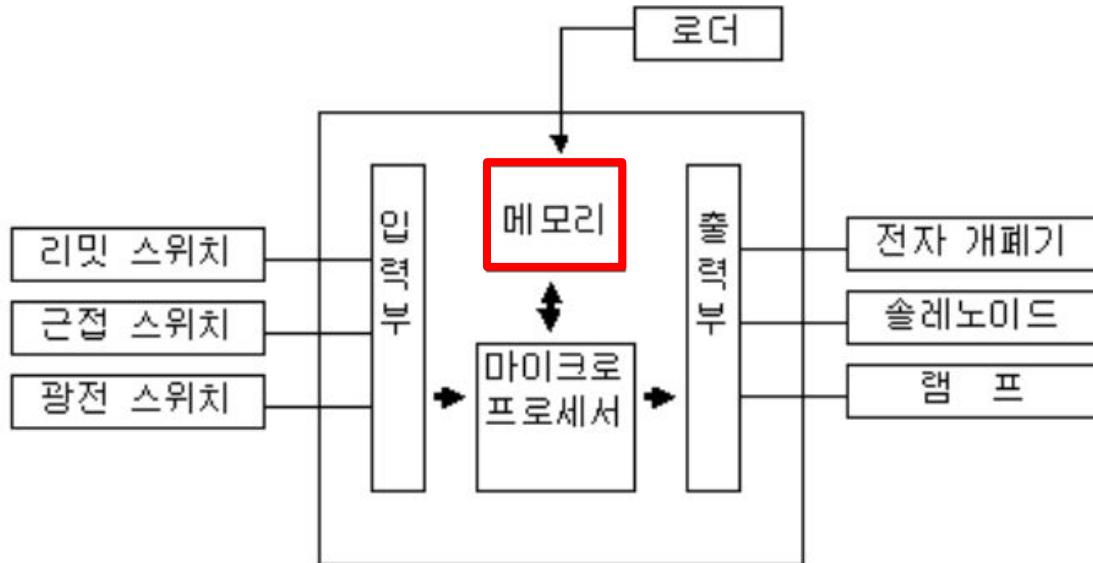
- PLC의 두뇌에 해당하는 부분으로서 메모리에 저장되어 있는 프로그램을 해독하여 실행합니다. CPU는 매우 빠른 속도로 반복 실행되며 모든 정보는 2진수로 처리됩니다.





하드웨어 구조 - CPU 메모리

- ROM(Read Only Memory)과 RAM(Random Access Memory)
- ROM - 읽기 전용, 메모리 내용을 변경할 수 없음. 한번 작성하면 이후에 변경되지 않는 시스템 관련된 프로그램을 저장
- 전원이 끊어져도 메모리의 내용이 그대로 보존되는 비휘발성 메모리



하드웨어 구조 - CPU 메모리



- RAM - 메모리에 정보를 수시로 읽고 쓰기가 가능하여 정보를 일시 저장하는 용도로 사용되나, 전원이 끊어지면 기억시킨 정보 내용을 모두 상실하는 휘발성 메모리
- 필요에 따라 RAM 영역 일부를 전원이 오프 되어도 배터리에 의해 필요한 전원을 공급하여 메모리의 내용이 지워지지 않도록 하는 방법을 배터리 백업(Battery back-up) 이라 하는데 이러한 방법을 통하여 RAM도 비휘발성 영역으로 사용할 수 있습니다. PLC의 데이터 영역과 사용자 프로그램은 변경이 가능해야 하므로 RAM영역에 저장



하드웨어 구조 - CPU 메모리



- PLC의 메모리
 - 사용자 프로그램 메모리,
 - 데이터 메모리,
 - 시스템 메모리



하드웨어 구조 - CPU 메모리



- 사용자 프로그램 메모리
 - 제어하고자 하는 시스템 규격에 따라 사용자가 작성한 프로그램이 저장되는 영역
 - 제어 내용이 프로그램 완성 전이나 완성 후에도 변경될 수 있으므로 RAM이 사용
 - 프로그램이 완성되어 고정되면, ROM에 기록하여 실행 할 수 있는데 이를 ROM운전 이라 한다.



하드웨어 구조 - CPU 메모리



- 데이터 메모리
 - 입·출력 릴레이, 보조 릴레이, 타이머와 카운터의 접점 상태 및 설정 값, 현재 값 등의 정보가 저장되는 영역
 - 정보가 수시로 바뀌므로 RAM 영역이 사용



하드웨어 구조 - CPU 메모리



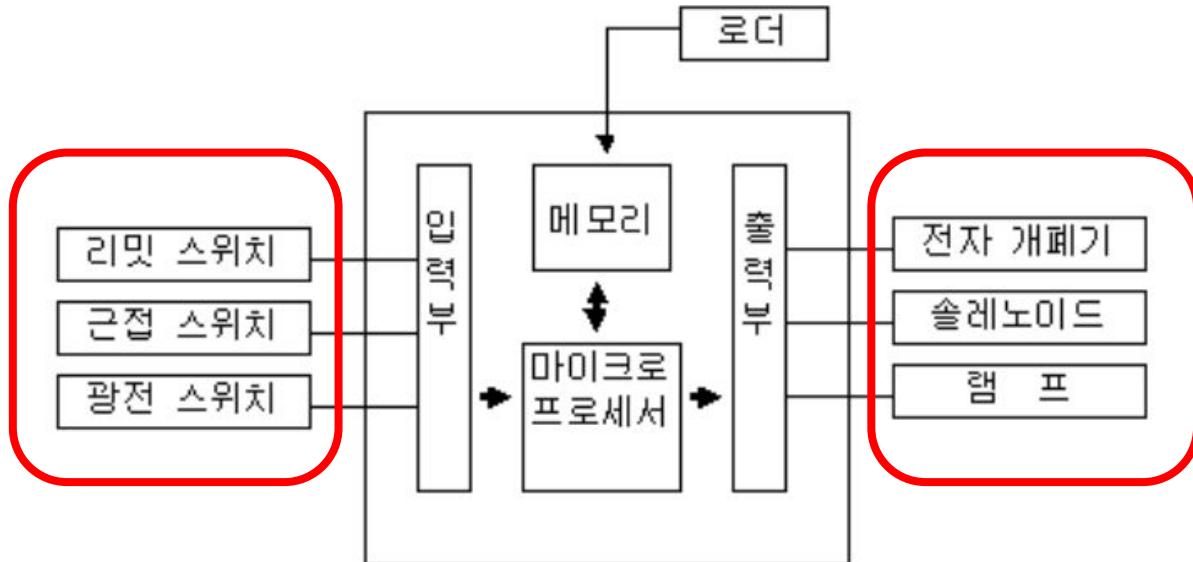
- 시스템 메모리
 - PLC 제작 회사에서 작성한 시스템 프로그램이 저장되는 영역
 - 시스템 프로그램은 PLC의 명령어를 실행시켜주는 명령어 관련 프로그램과 자기 진단기능등과 같이 PLC동작시 발생하는 오류나 에러 등을 체크해주는 프로그램,
 - PLC 제작회사에서 파워를 ON/OFF하여도 지워지지 않도록 ROM에 저장





하드웨어 구조 - 입출력부

- PLC의 입출력부는 현장의 기기에 직접 접속하여 사용
- 내부 전원 레벨과 외부 입출력 전원 레벨이 다름
 - PLC 내부는 DC5V 의 전원(TTL 레벨)을 사용
 - 입출력부는 DC24V 또는 AC110V, 220V 등의 높은 전압 레벨



하드웨어 구조 - 입출력부의 요구사항



- 외부 기기와 전기적 규격이 일치 할것
- 외부 기기로부터의 노이즈가 CPU쪽에 전달되지 않도록 할것
 - 포토커플러 사용
- 외부 기기와의 접속 용이성
- 입출력의 각 접점 상태를 감시할 수 있어야 함
 - 신호용 LED 사용



하드웨어 구조 - 입출력부의 외부기기 예



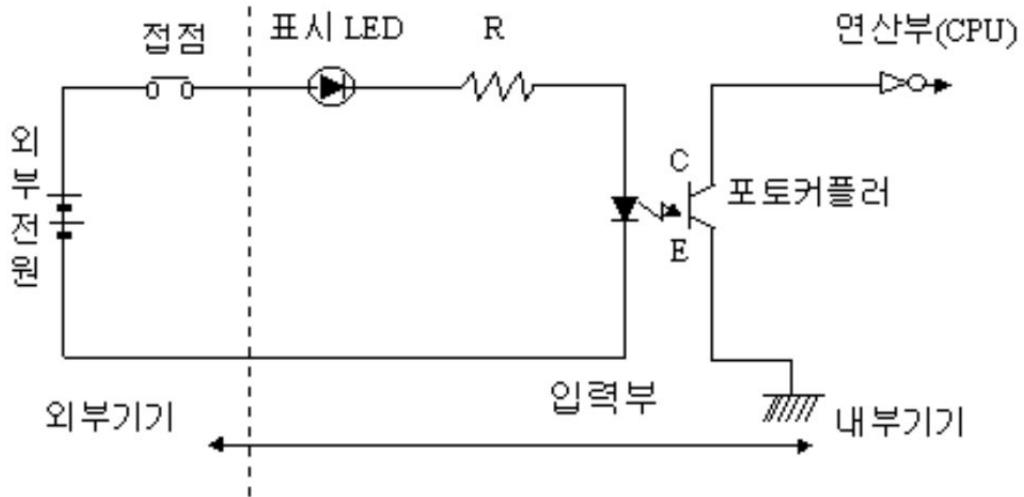
I/O	구 분	부 착 장 소	외부 기기의 명치
입력부	조작 입력	제어반과 조작반	푸시 버튼 스위치 선택 스위치 토글 스위치
	검출 입력 (센서)	기계 장치	리밋 스위치 광전 스위치 근접 스위치 레벨 스위치
출력부	표시 경보 출력	제어반 및 조작반	파일럿 램프 부저
	구동 출력 (액추에이터)	기계장치	전자 밸브 전자 클러치 전자 브레이크 전자 개폐기



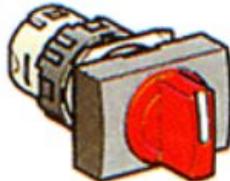
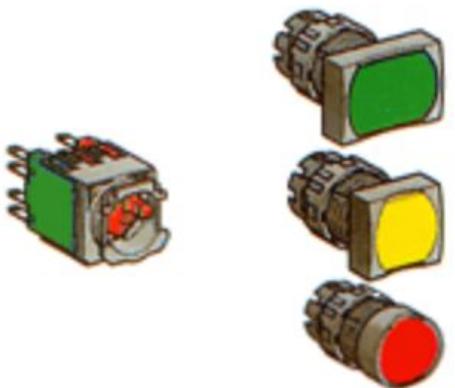
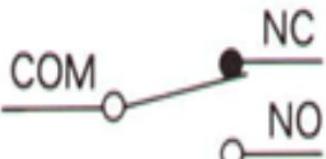


하드웨어 구조 - 입력부

- 외부 기기로부터의 신호를 CPU의 연산부로 전달해 주는 역할
- 입력의 종류로는 DC24[V], AC110[V] 등
 - 특수 입력 모듈로 아날로그 입력(A/D) 모듈,
 - 고속 카운터 (High Speed Counter) 모듈 등



하드웨어 구조 - 입력 스위치 및 등가회로



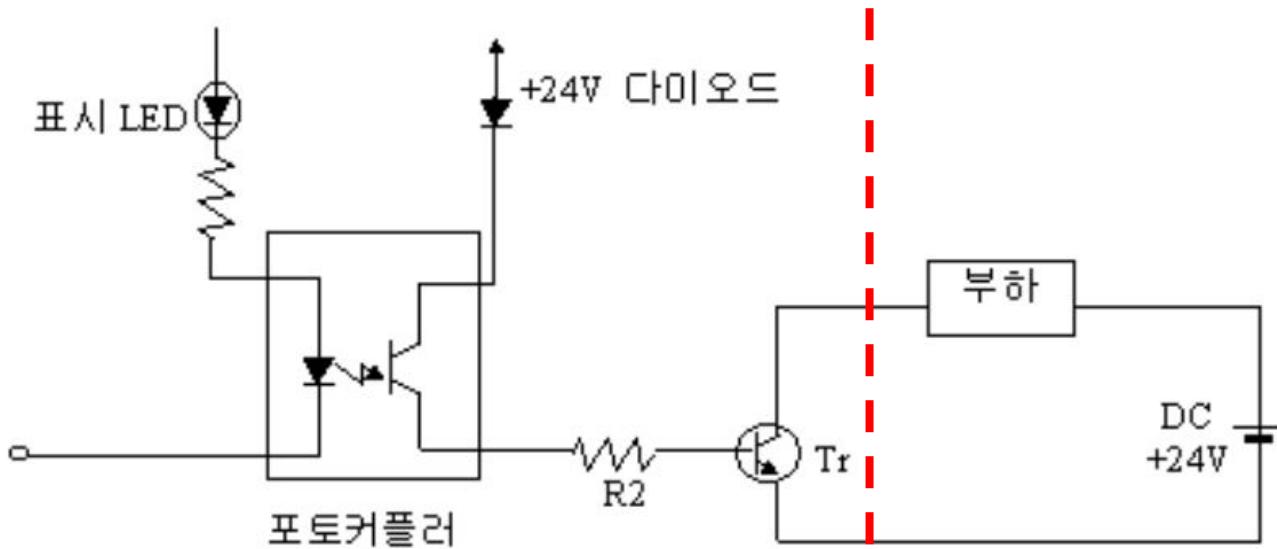
하드웨어 구조 - 출력부



- 내부 연산의 결과를 외부에 접속된 전자 접촉기나 솔레노이드에 전달하여 구동시키는 부분
 - 릴레이 출력,
 - 트랜지스터 출력,
 - SSR(Solid State Relay) 출력
 - 아날로그 출력(D/A) 모듈,
 - 위치 결정 모듈 등



하드웨어 구조 - 트랜지스터 출력부 Schematics





하드웨어 구조 - 출력부

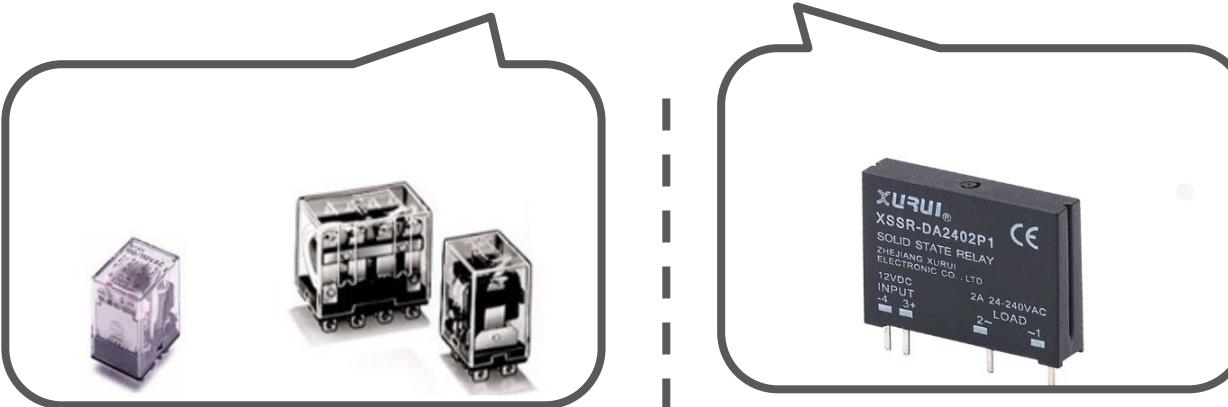
출력 회로용 전원	개폐소자	
	유접점	무접점(반도체)
직류(DC24V)	릴레이 출력	트랜지스터 출력
교류(AC220V)	릴레이 출력	SSR 출력



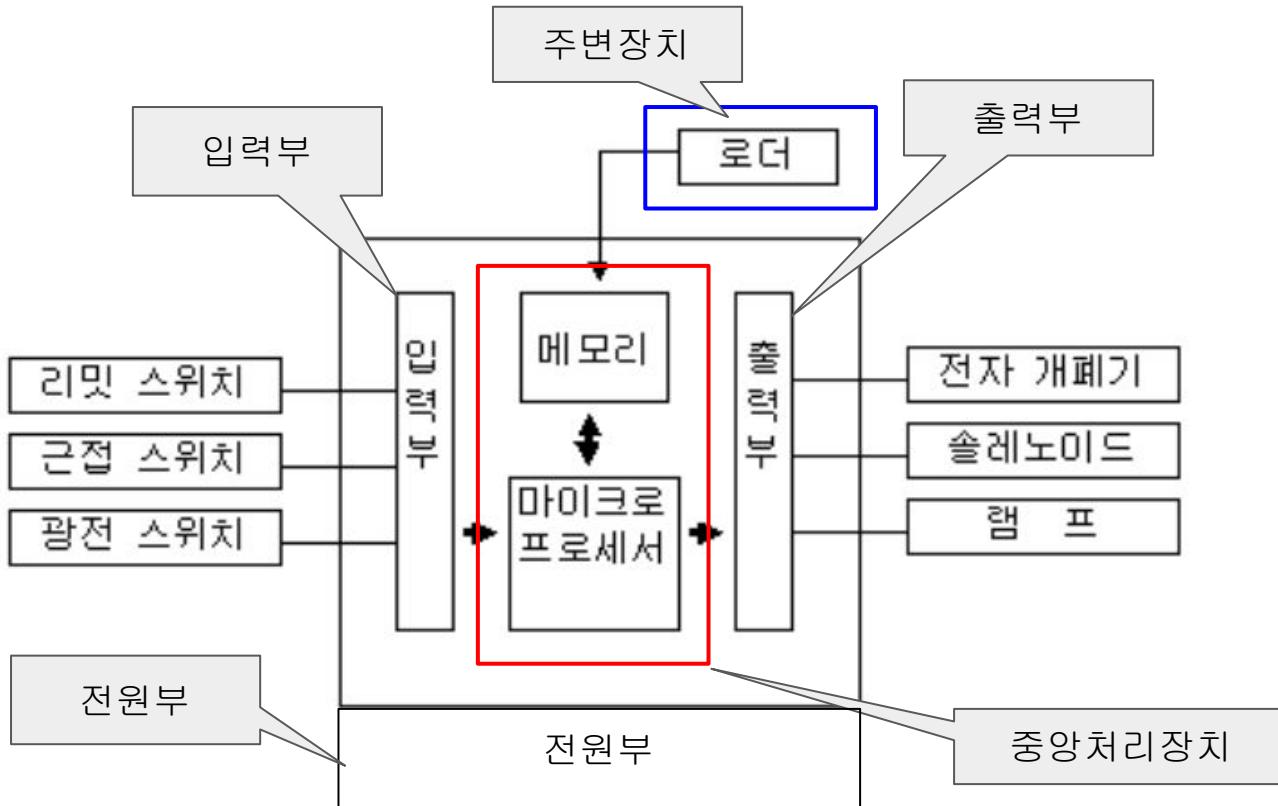


하드웨어 구조 - 출력부

출력 회로용 전원	개폐소자	
	유접점	무접점(반도체)
직류(DC24V)	릴레이 출력	트랜지스터 출력
교류(AC220V)	릴레이 출력	SSR 출력



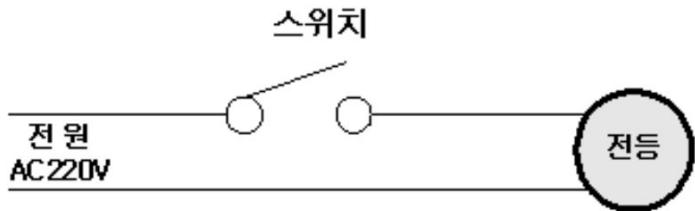
PLC 구성 [복습]



PLC 이론과 실습



전기회로 구성

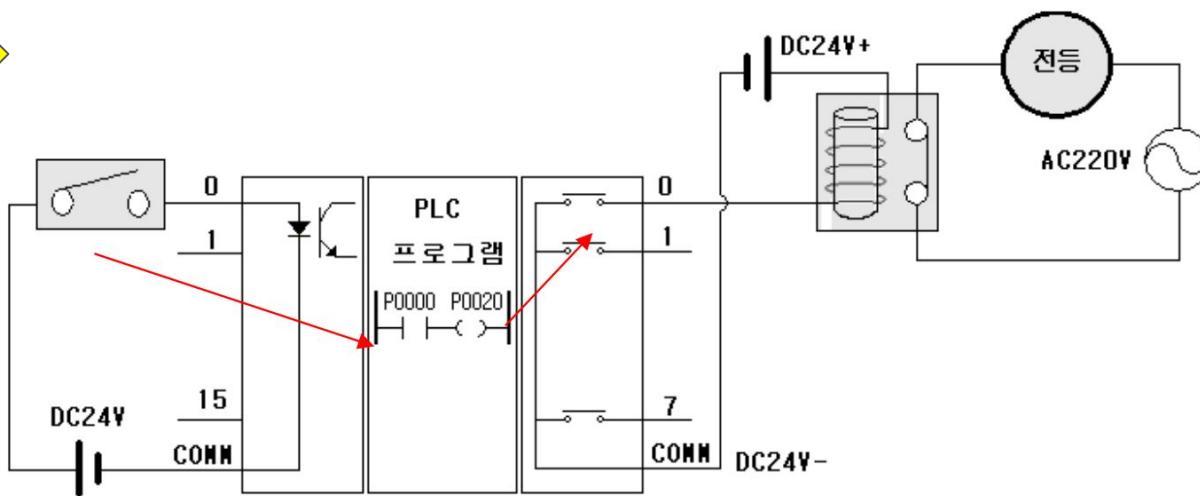
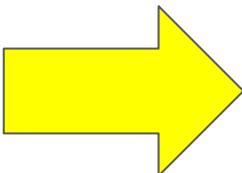
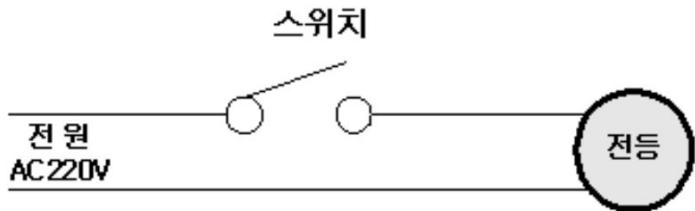


위의 전기회로를 PLC로 제어할 경우에는 스위치를 입력부, 전등을 출력부로 구분하여 회로를 별도로 구성한다. 입출력은 별도로 구성하고 입출력간의 연결은 PLC 프로그램으로 작성한다. PLC 프로그램은 소프트웨어이므로 수정 및 편집이 자유롭다. 입출력만 결선하고 나면 프로그램에 의하여 로직 회로를 자유롭게 변경 및 수정이 가능하다.

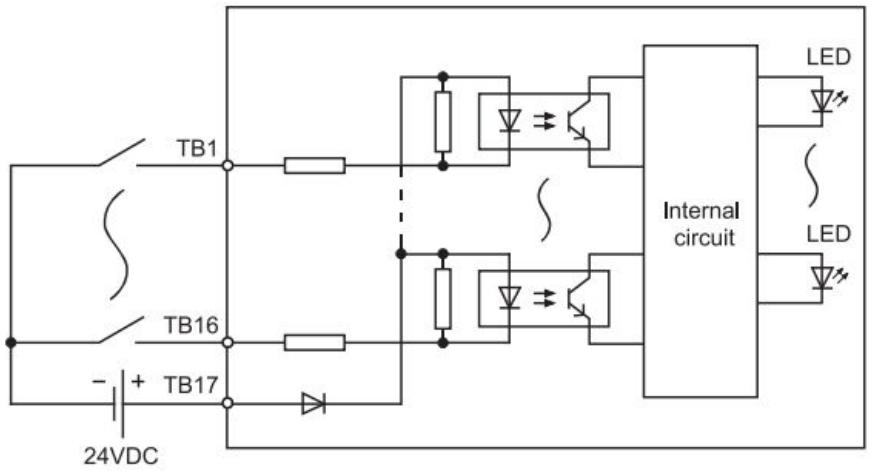




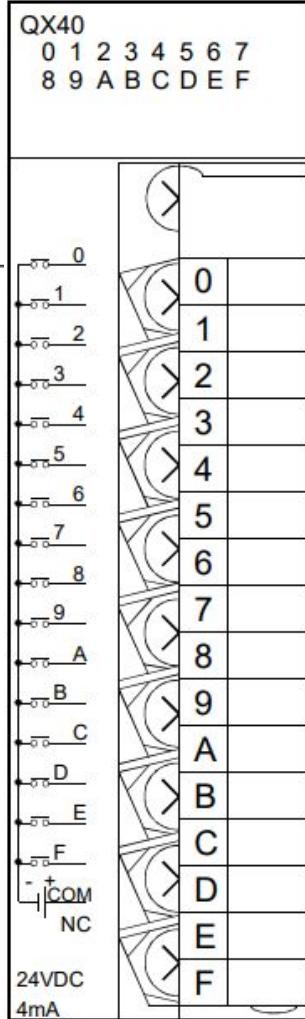
전기회로 구성



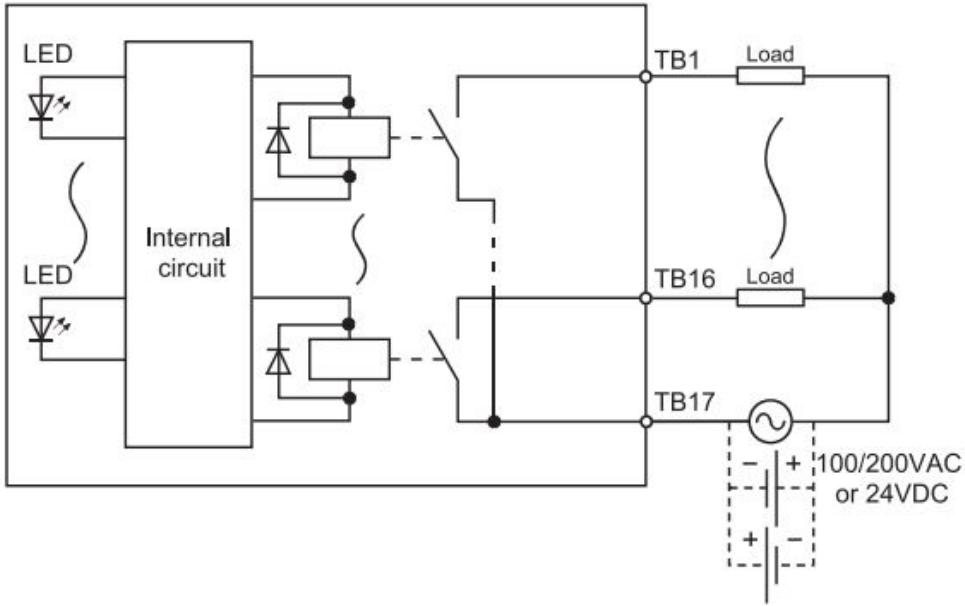
하드웨어 구조 'QX40' 입력모듈



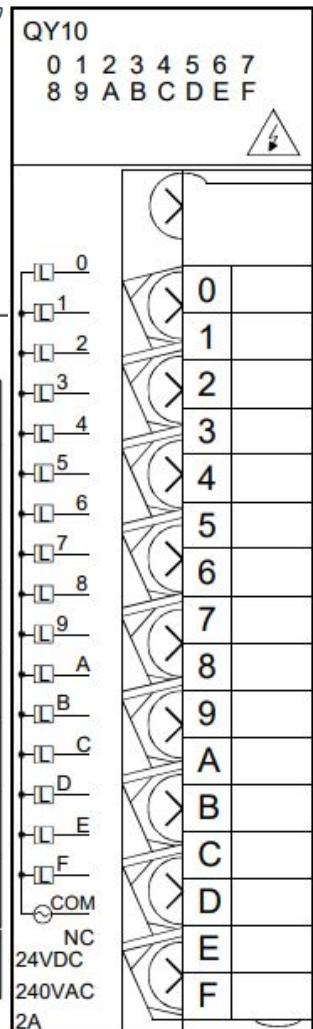
Terminal block No.	Signal name
TB1	X00
TB3	X02
TB5	X04
TB7	X06
TB9	X08
TB11	X0A
TB13	X0C
TB15	X0E
TB17	COM
TB18	Vacant



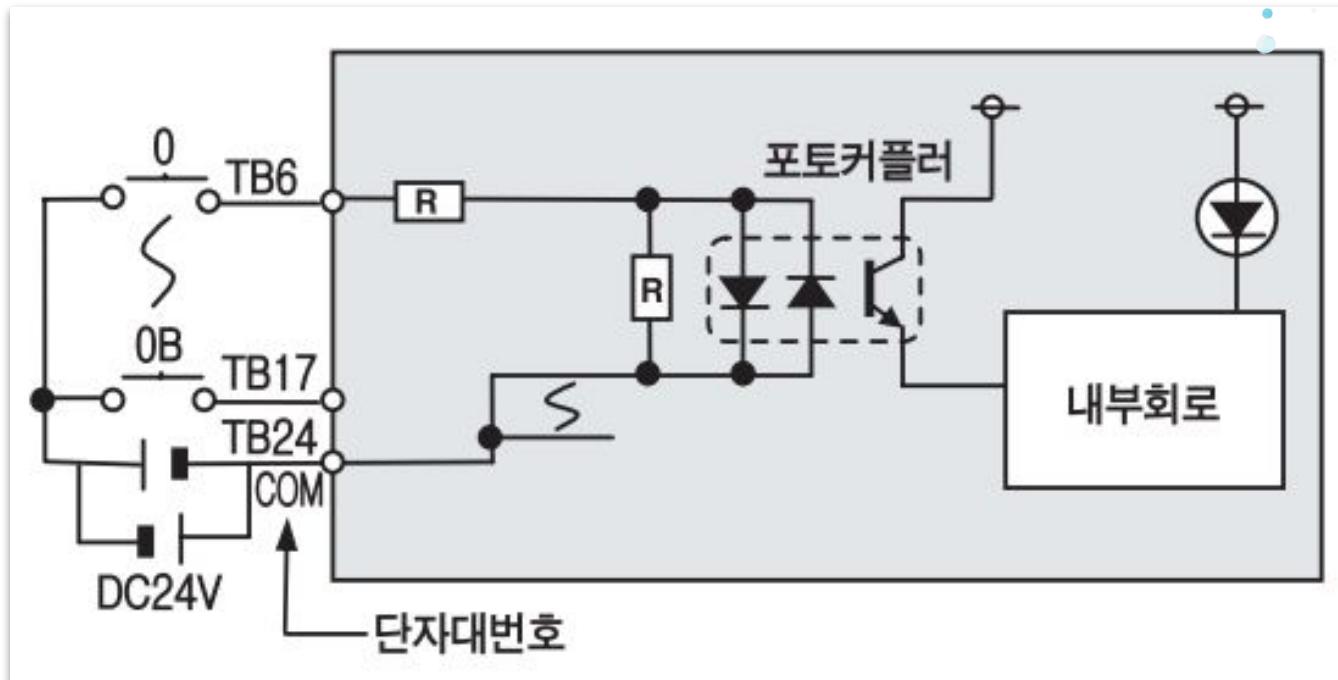
하드웨어 구조 'QY10' 출력모듈



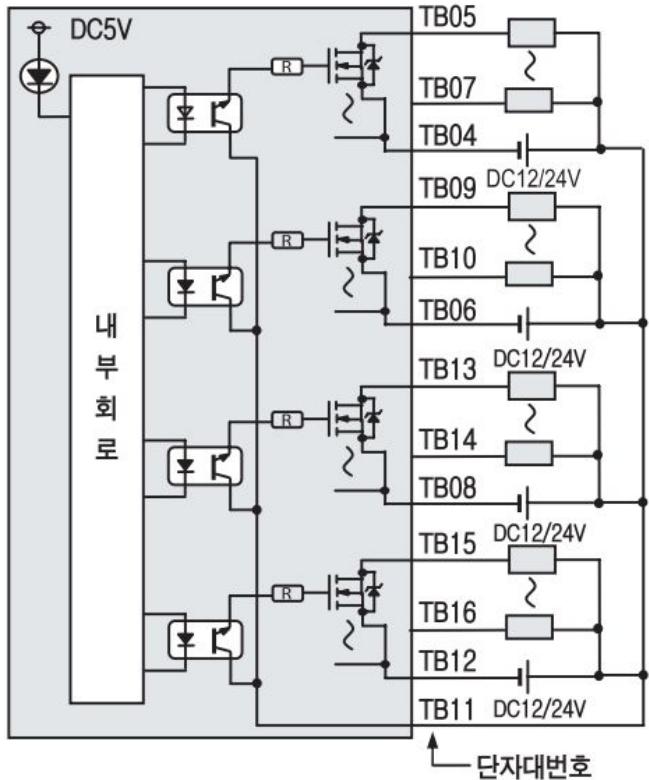
Terminal block No.	Signal name
TB1	Y00
TB2	Y01
TB3	Y02
TB4	Y03
TB5	Y04
TB6	Y05
TB7	Y06
TB8	Y07
TB9	Y08
TB10	Y09
TB11	Y0A
TB12	Y0B
TB13	Y0C
TB14	Y0D
TB15	Y0E
TB16	Y0F
TB17	COM
TB18	Vacant



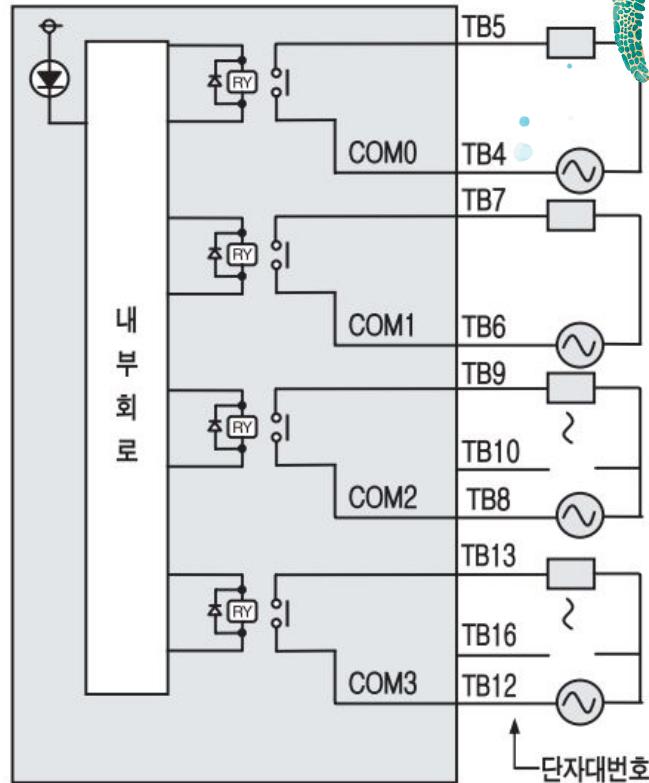
하드웨어 구조 'XBC-DR20'



하드웨어 구조 'XBC-DR20'



TR 출력부 (싱크타입)



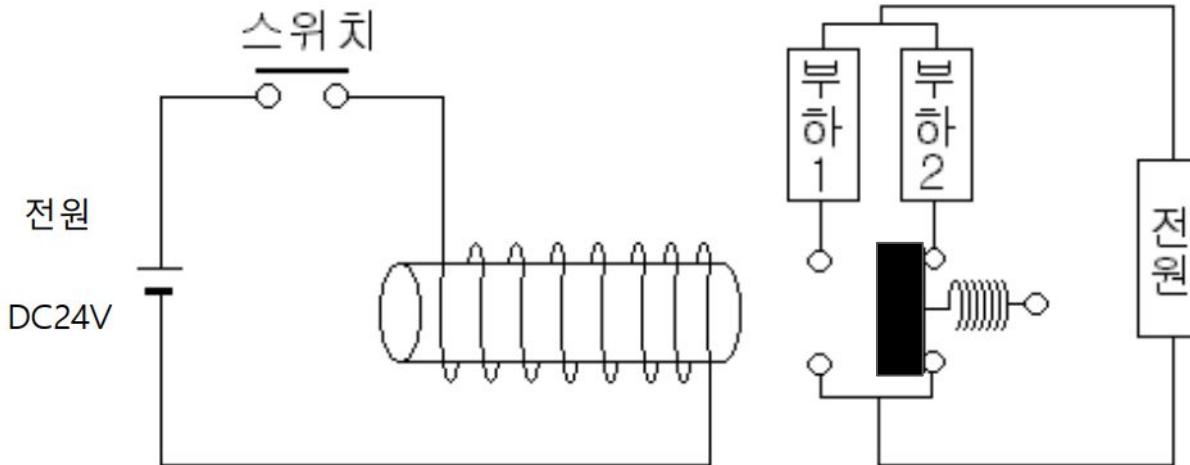
릴레이 출력부



하드웨어 구조 - 출력부



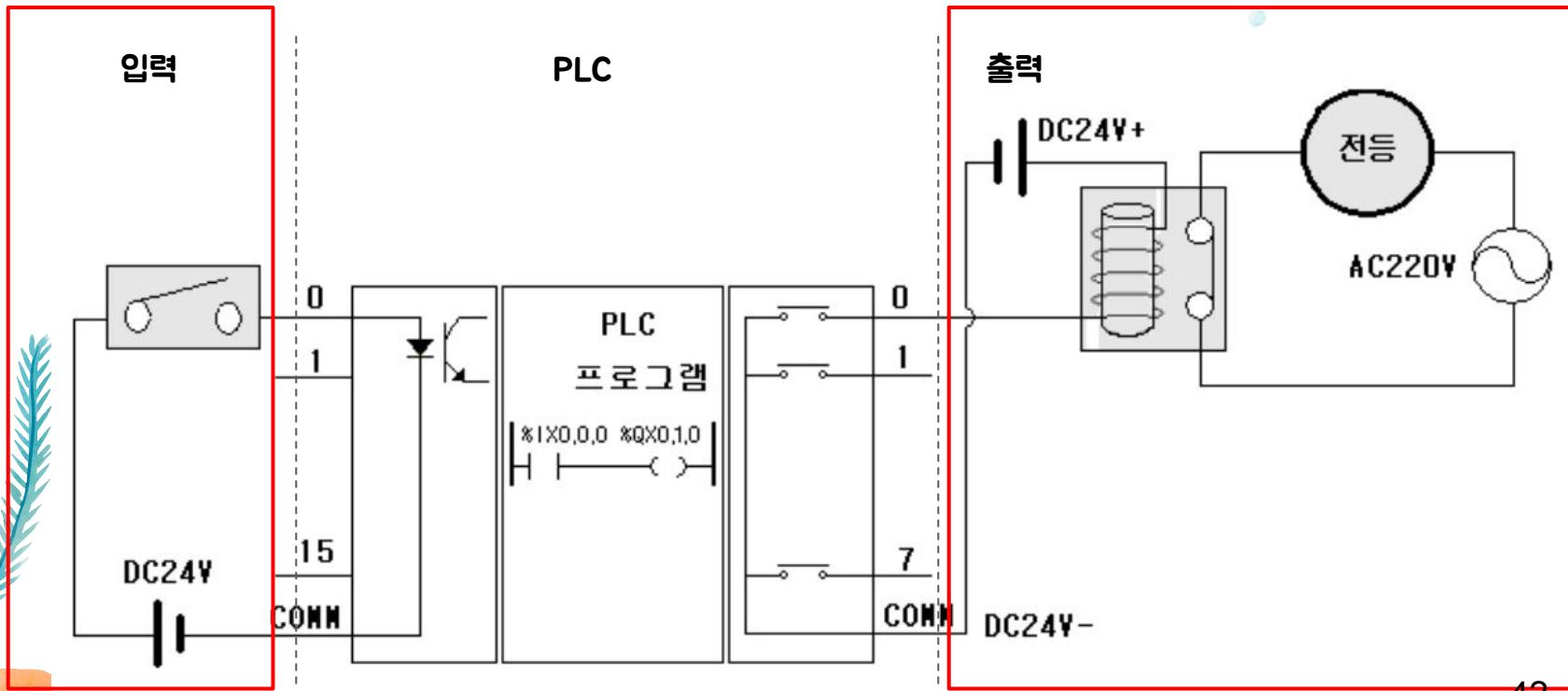
- 릴레이 출력은 직류와 교류 모두 사용 가능
- 그러나 기계적 수명의 한계 때문에 접점의 개폐가 빈번할 경우는
 - 교류 전원 전용인 무접점 SSR 출력이나
 - 직류 전원 전용인 트랜지스터 출력을 사용
- 릴레이는 기계접점방식으로 10만회 ~ 100만회 수명을 가짐





하드웨어 구조 - 전기회로 구성

- PLC 회로 구성 스위치 ON/OFF



소프트웨어 구조

- 하드웨어 로직
 - 종래의 릴레이 제어 방식
 - 일의 순서를 회로도에 전개하여 그곳에 필요한 제어 기기를 결합하여 리드선으로 배선 작업
 - 하드웨어(기기)와 소프트웨어가 한 쌍이 되어 있어, 사양이 변경되면 하드웨어와 소프트웨어를 모두 변경
 - 하드웨어와 소프트웨어를 분리하는 연구 진행



소프트웨어 구조



- 소프트웨어 구조
 - 컴퓨터는 하드웨어(Hardware)만으로는 동작할 수 없음
 - 하드웨어 속에 있는 기억 장치에 일의 순서 넣어 작동
 - 기억 장치 메모리에 일의 순서를 넣는 작업이 프로그래밍
 - 배선작업과 유사하여 붙여진 이름



릴레이 시퀀스와 PLC 프로그램 차이점



- PLC는 IC 와 반도체 전자 부품의 집합으로 릴레이 시퀀스와 같은 접점이나 코일은 존재하지 않고, 접점이나 코일을 연결하는 동작은 소프트웨어로 처리된다.
- 코일이 여자되면 접점이 닫혀 회로가 활성화되는 릴레이 시퀀스와는 달리 메모리에 프로그램을 기억시켜 놓고 순차적으로 내용을 읽어서 동작시키는 방식이다.
- PLC제어는 프로그램의 내용에 의하여 좌우된다. 사용자는 자유 자재로 원하는 제어를 할 수 있도록 프로그램 작성 능력이 요구된다.





직렬처리와 병렬처리

- PLC는 메모리에 있는 프로그램을 순차적으로 연산하는 직렬 처리 방식
- 릴레이 시퀀스는 여러 회로가 전기적인 신호에 의해 동시에 동작하는 병렬 처리 방식

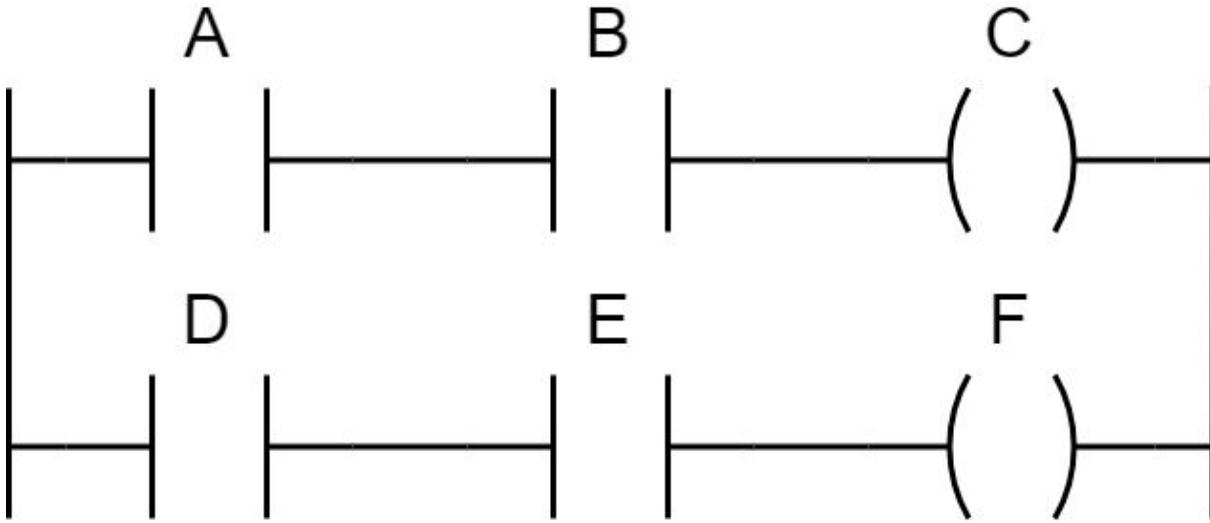


(a) 직렬 처리 방식



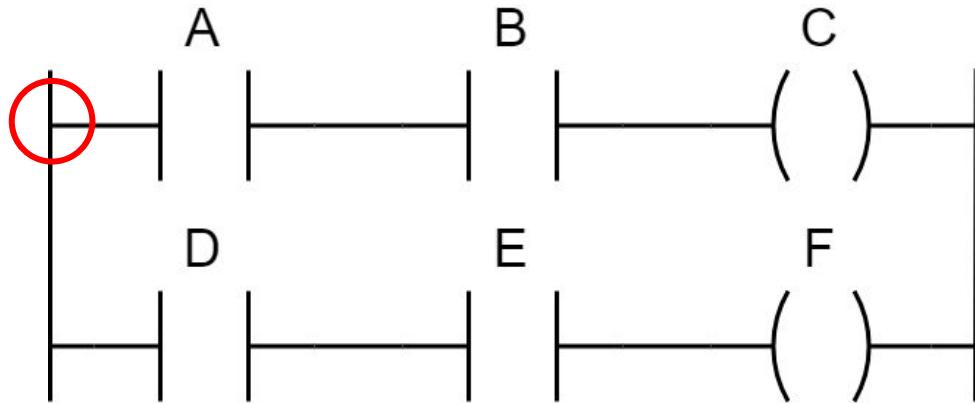
(b) 병렬 처리 방식

직렬처리와 병렬처리 - 시퀀스도



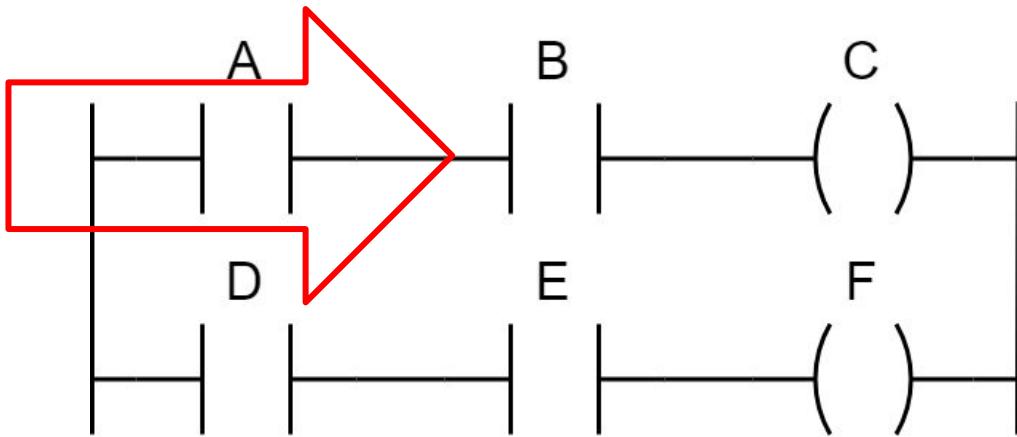
직렬처리와 병렬처리 - 시퀀스도

- 직렬처리



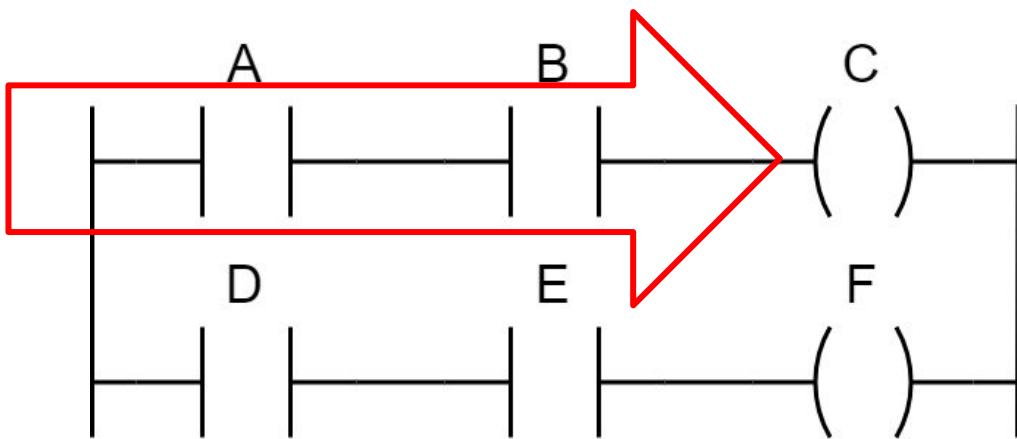
직렬처리와 병렬처리 - 시퀀스도

- 직렬처리



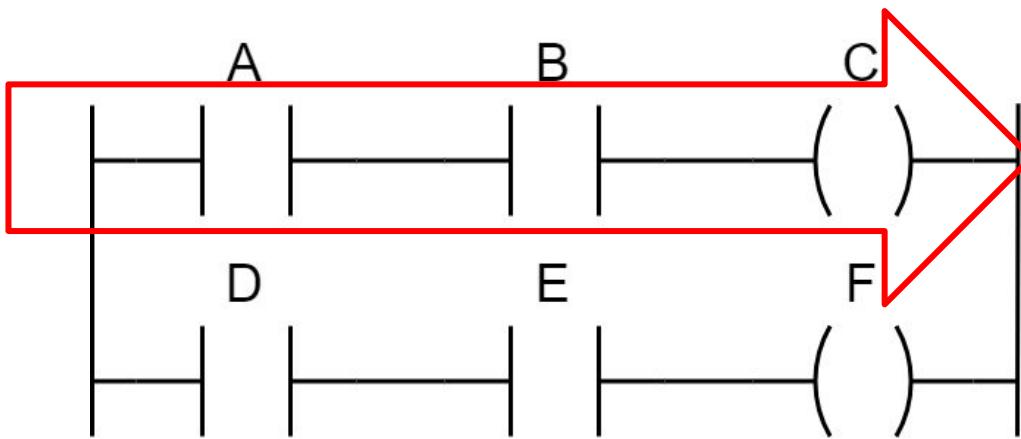
직렬처리와 병렬처리 - 시퀀스도

- 직렬처리



직렬처리와 병렬처리 - 시퀀스도

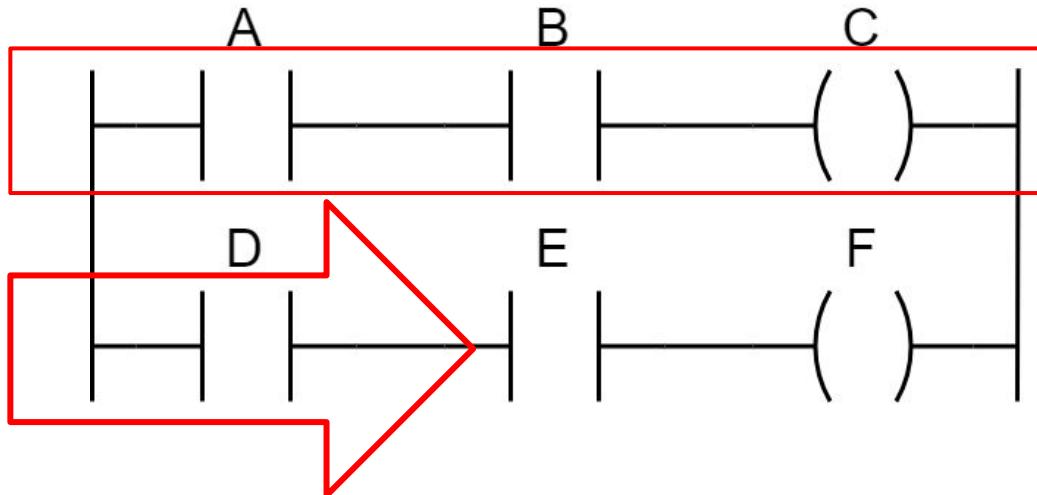
- 직렬처리



직렬처리와 병렬처리 - 시퀀스도



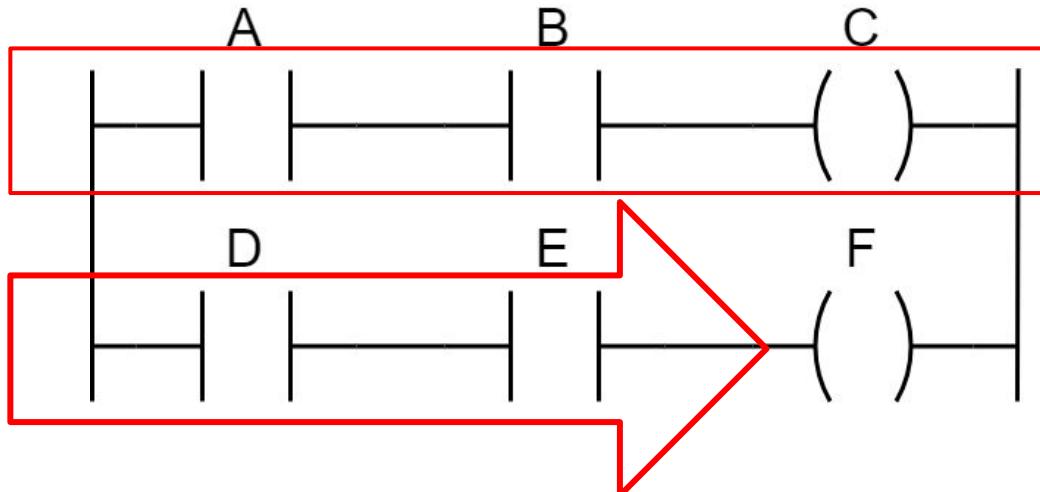
- 직렬처리



직렬처리와 병렬처리 - 시퀀스도



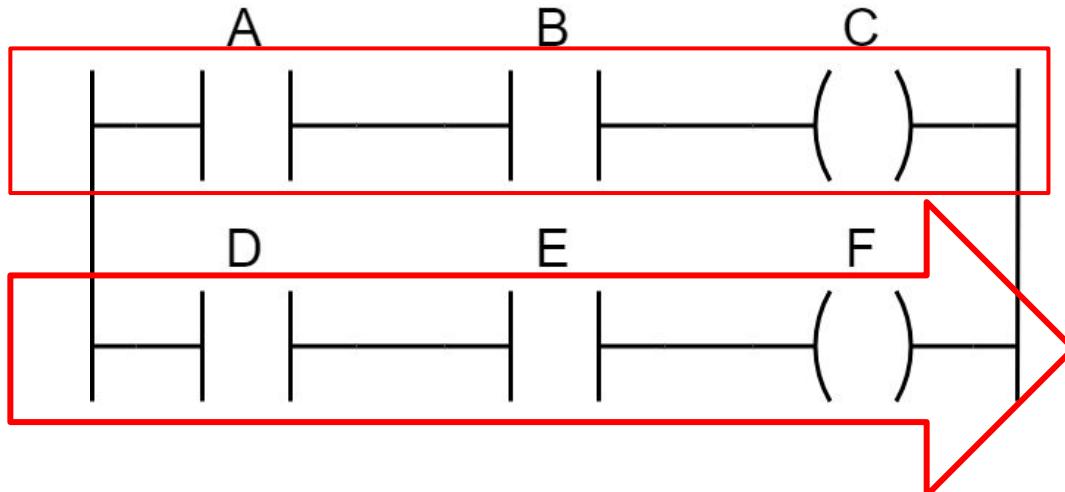
- 직렬처리



직렬처리와 병렬처리 - 시퀀스도



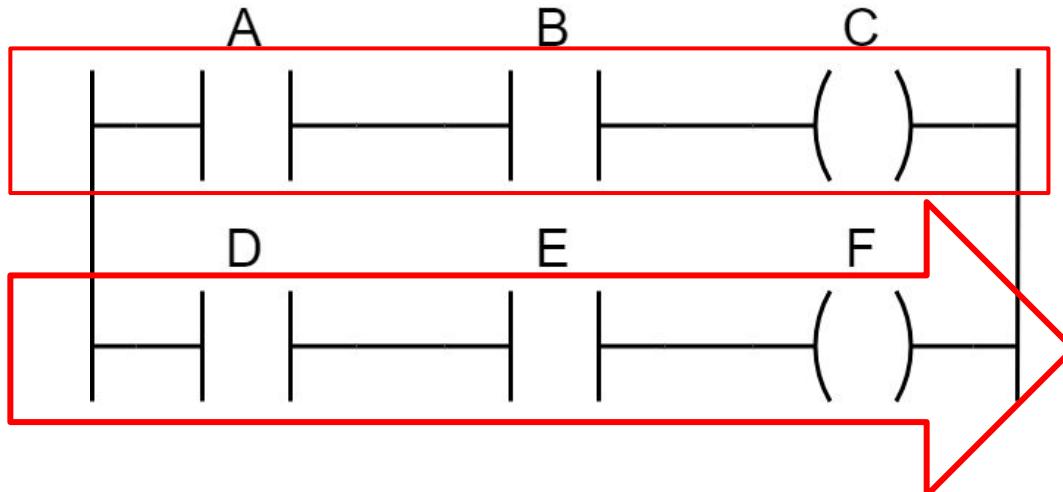
- 직렬처리



직렬처리와 병렬처리 - 시퀀스도

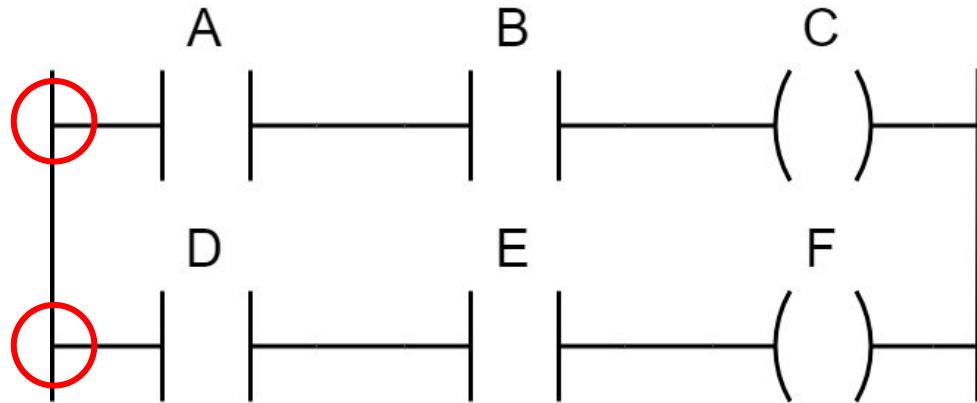


- 직렬처리
 - C 가 먼저 처리된 후 F 가 출력됨



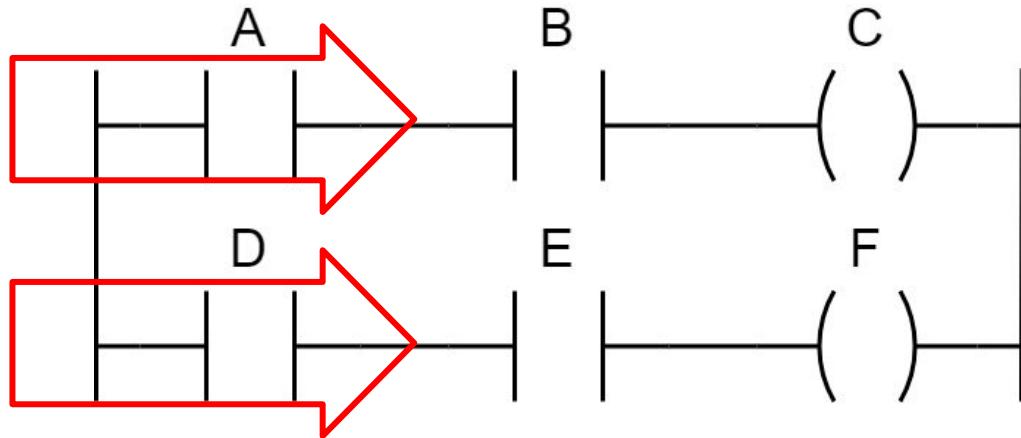
직렬처리와 병렬처리 - 시퀀스도

- 병렬처리



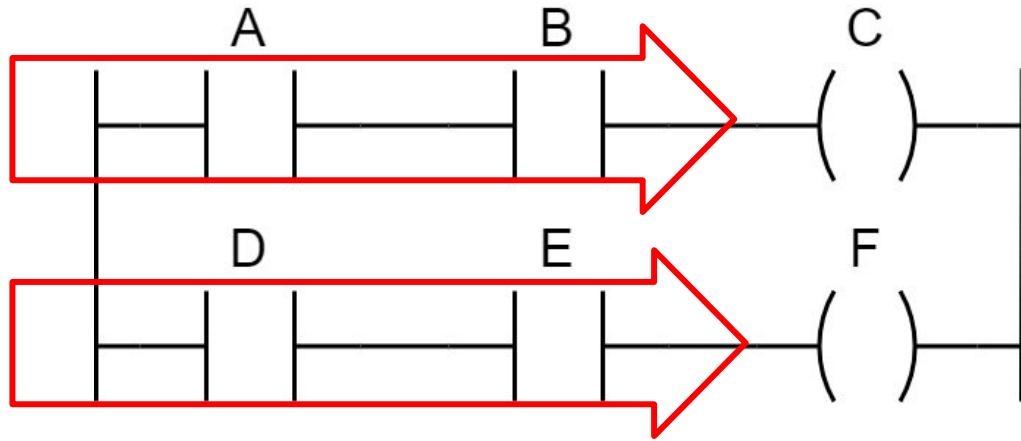
직렬처리와 병렬처리 - 시퀀스도

- 병렬처리



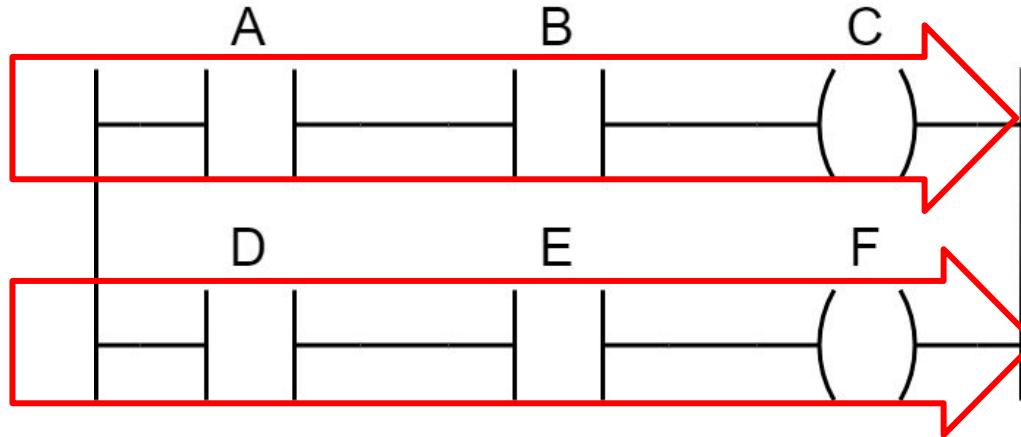
직렬처리와 병렬처리 - 시퀀스도

- 병렬처리



직렬처리와 병렬처리 - 시퀀스도

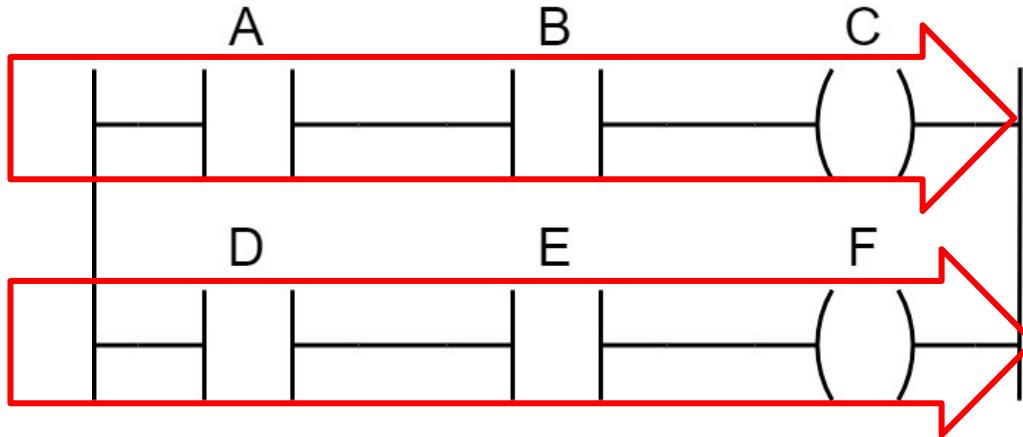
- 병렬처리



직렬처리와 병렬처리 - 시퀀스도



- 병렬처리
 - C 나 F 중 무엇이 먼저 처리되는지 알수 없음



릴레이와 PLC 의 사용점점 수 제한



- 릴레이는 일반적으로 1개당 가질 수 있는 접점 수에 한계가 있으므로 릴레이 시퀀스를 작성할 때에는 사용하는 접점 수를 가능한 한 줄여야 사용
- 이에 반해 PLC는 동일 접점에 대하여 사용 회수에 제한을 받지 않음.
 - 동일 접점에 대한 정보(ON/OFF)를 정해진 메모리에 저장해 놓고, 연산할 때 메모리에 있는 정보를 읽어서 처리

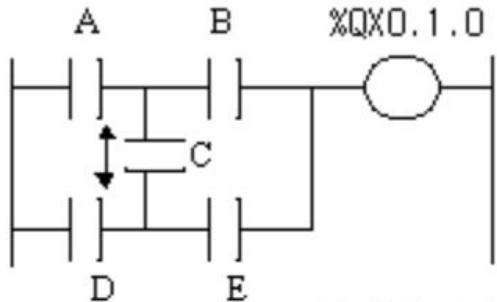


접점이나 코일 위치의 제한

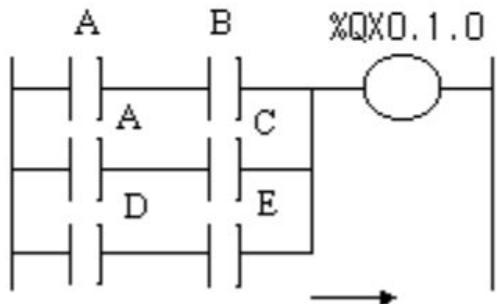
- PLC 시퀀스에는 몇가지 규정이 있음
- 코일 이후 접점을 금지
 - PLC 시퀀스에서는 코일을 반드시 오른쪽 모선에 붙여서 작성
 - PLC 시퀀스에서는 항상 신호가 왼쪽에서 오른쪽으로 전달
 - 오른쪽에서 왼쪽으로 흐르는 회로 금지
 - 상하로 흐르는 회로 구성 금지



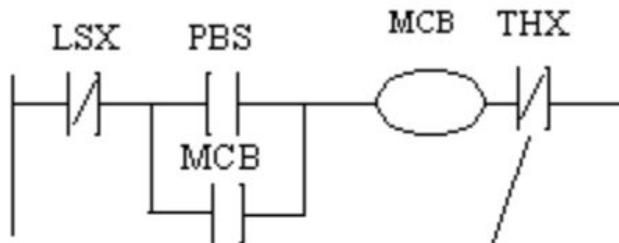
접점이나 코일 위치의 제한



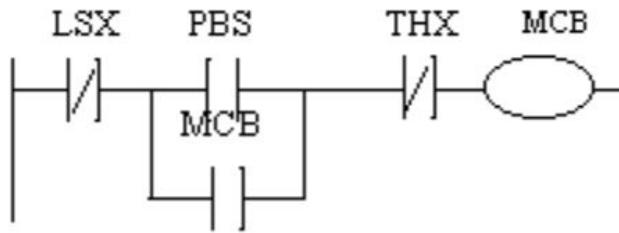
상하의 흐름은 금지



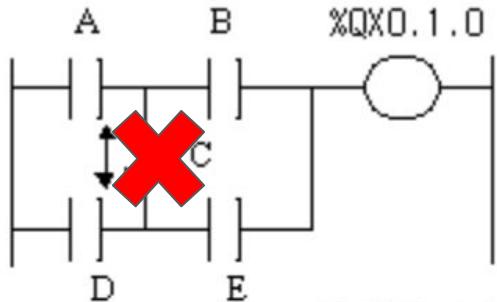
항상 좌에서 우로



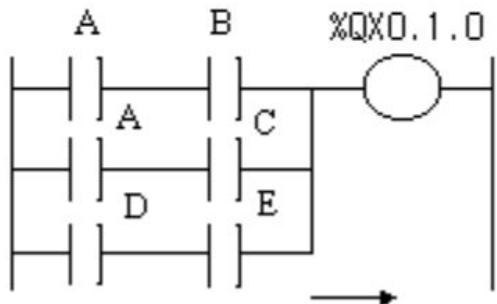
이곳으로 옮긴다



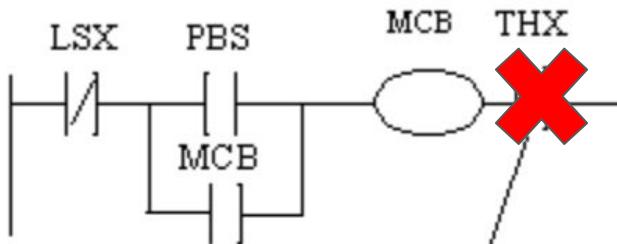
접점이나 코일 위치의 제한



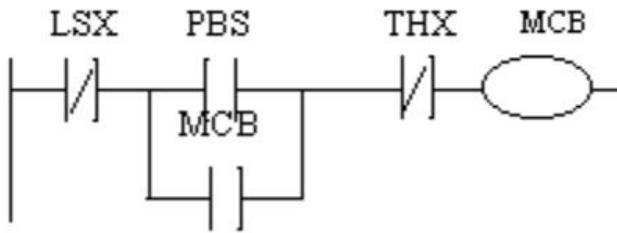
상하의 흐름은 금지



항상 좌에서 우로

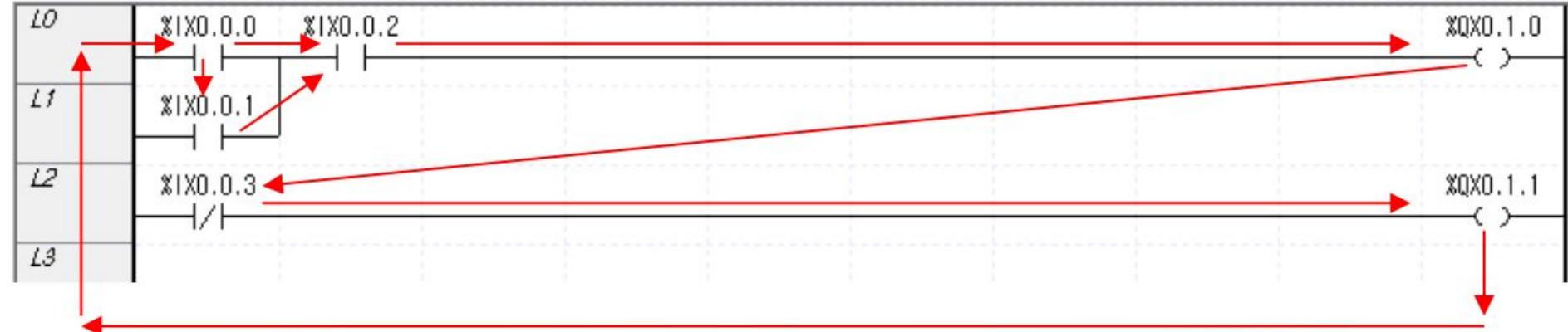


이곳으로 옮긴다





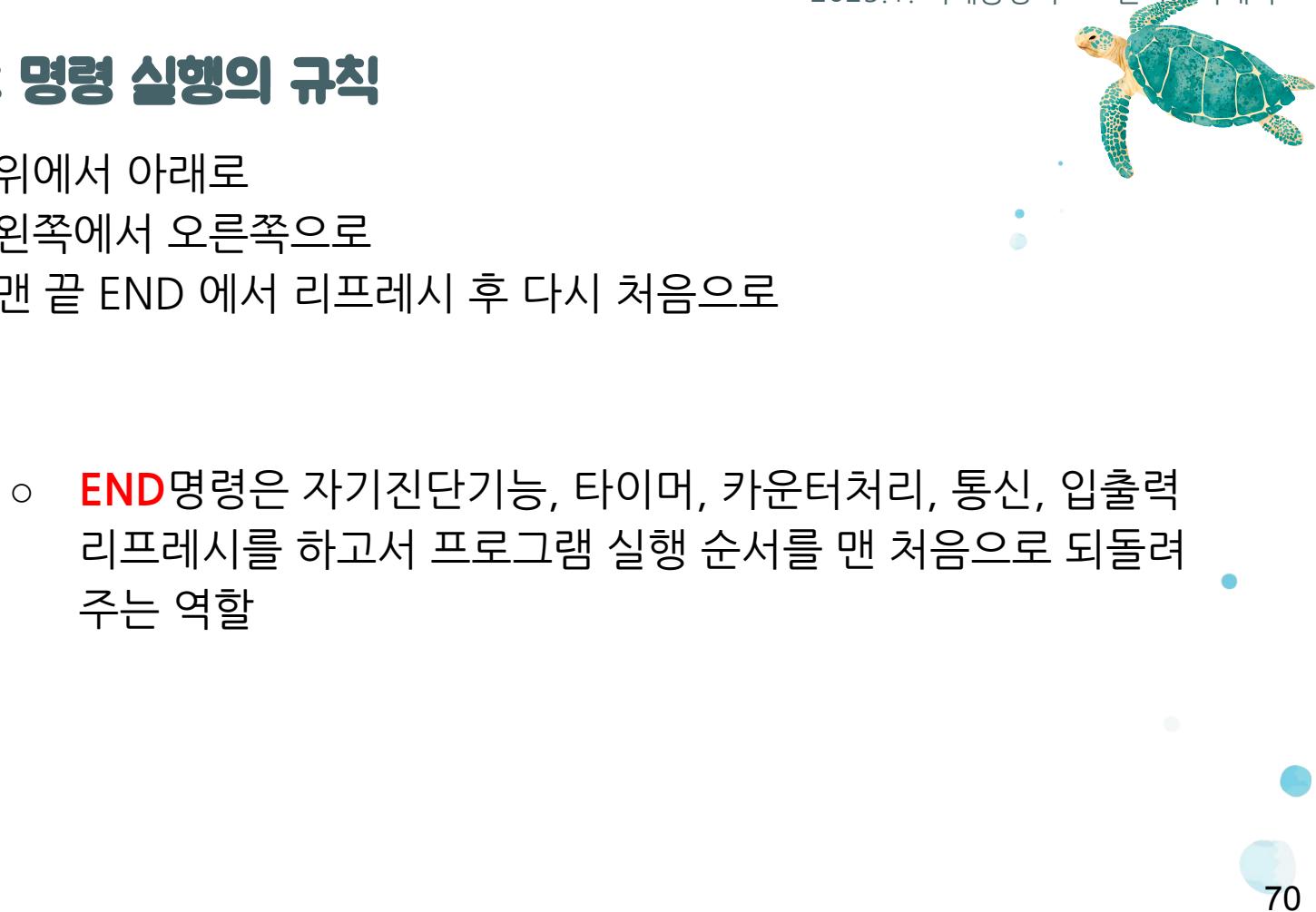
PLC 명령 실행의 규칙



- 위에서 아래로
- 왼쪽에서 오른쪽으로
- 맨 끝 END에서 리프레시 후 다시 처음으로



PLC 명령 실행의 규칙

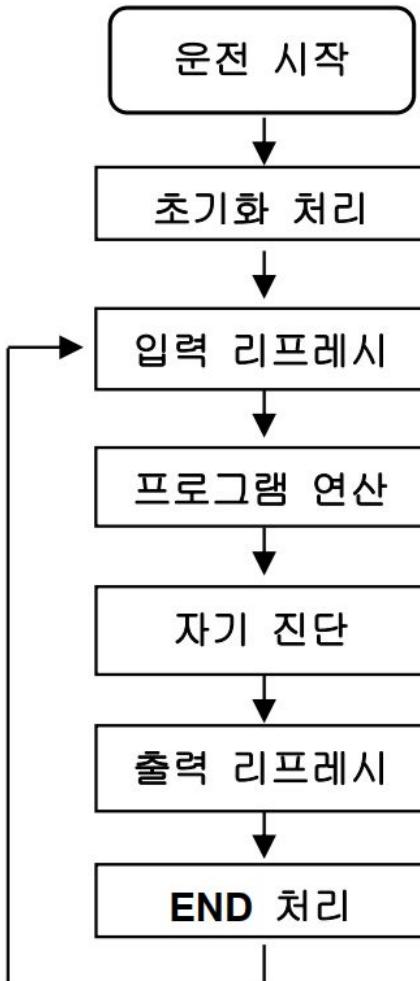


- 위에서 아래로
- 왼쪽에서 오른쪽으로
- 맨 끝 END에서 리프레시 후 다시 처음으로

- **END**명령은 자기진단기능, 타이머, 카운터처리, 통신, 입출력 리프레시를 하고서 프로그램 실행 순서를 맨 처음으로 되돌려 주는 역할

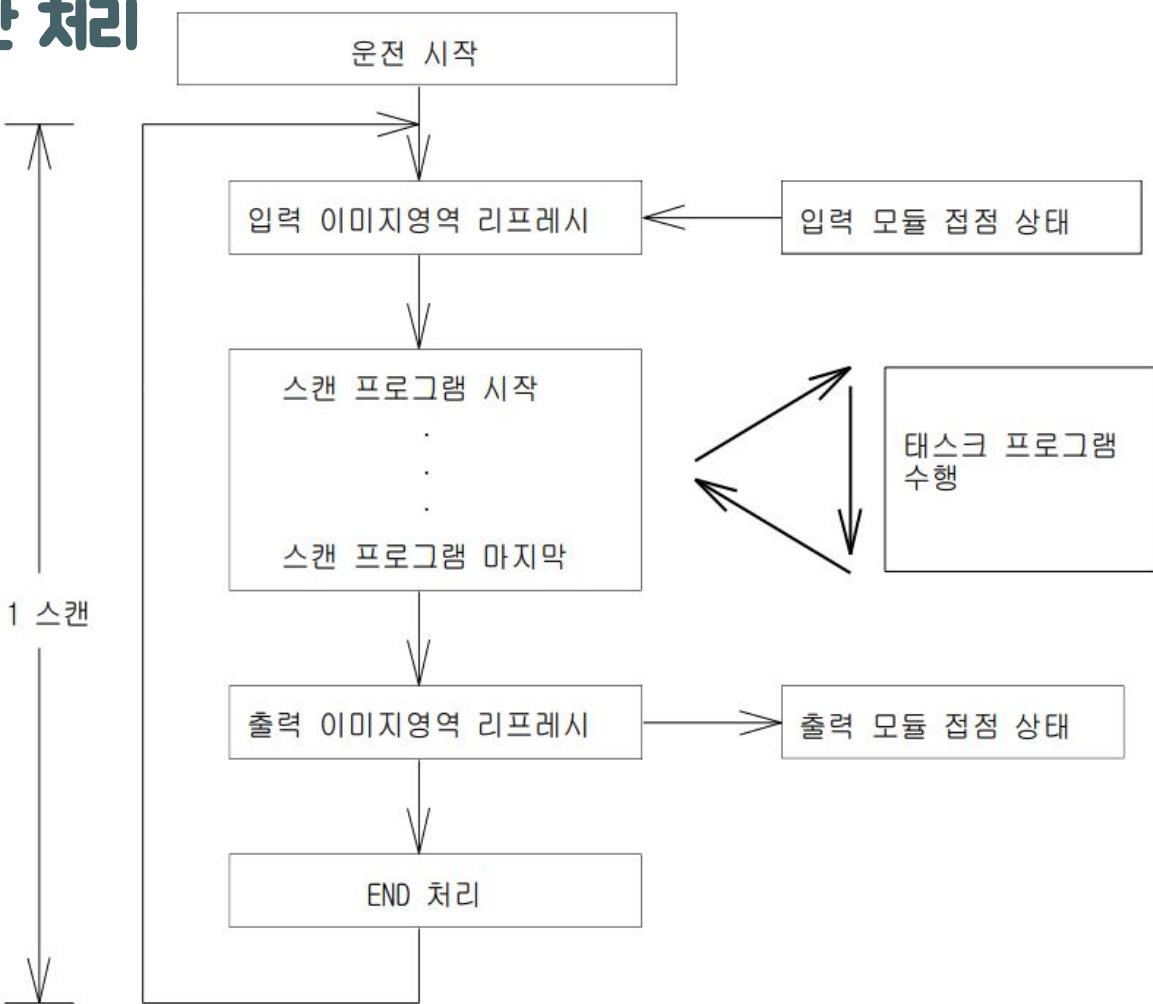
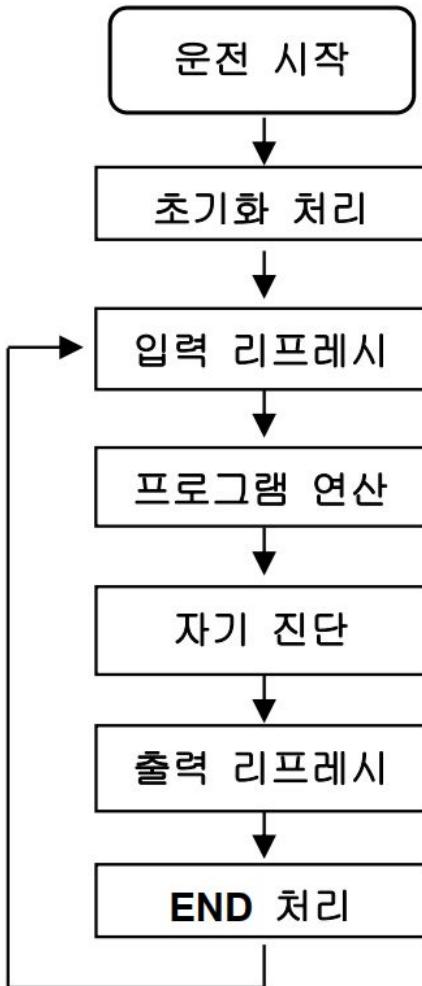


1.3. 연산 처리

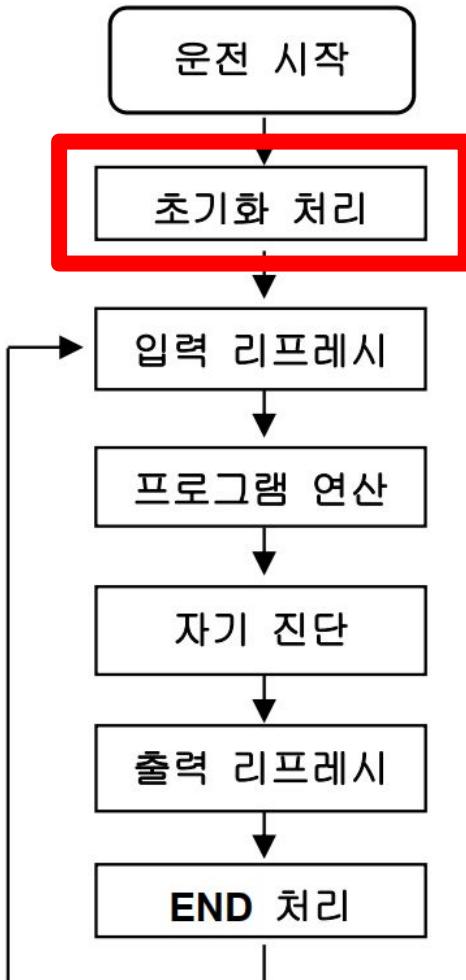


- 입력 리프레시(Refresh) 과정을 통해 입력의 상태를 PLC의 CPU가 인식하고, 인식된 정보를 조건 또는 데이터로 이용하여 프로그램 처음부터 마지막까지 순차적으로 연산을 실행하고 출력 리프레시(Refresh)를 한다.
- 이러한 동작은 고속으로 반복되는데 이러한 방식을 ‘반복 연산 방식’이라 하고 한 바퀴를 실행하는데 걸리는 시간을 ‘1스캔 타임’(1연산 주기)

1.3. 연산 처리



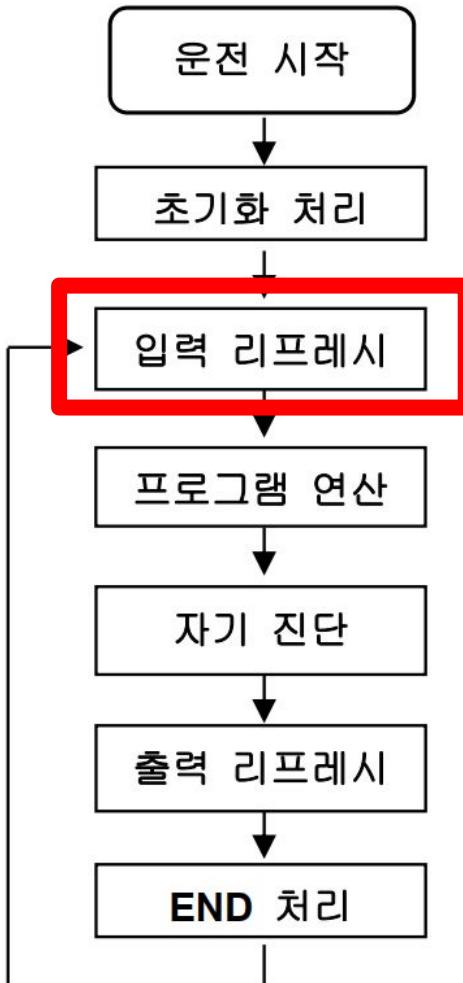
1.3. 연산 처리



(1) 초기화 처리: PLC 가 RUN 모드로 변경될 때 한 번만 실행하며, 다음과 같은 처리를 수행합니다.

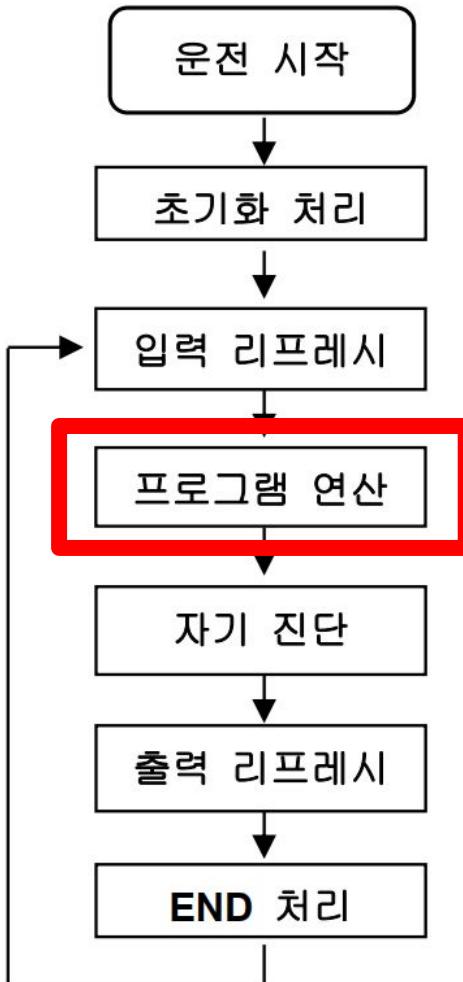
- 입, 출력 모듈 리셋: 베이스에 설치된 모듈을 리셋하고, 모듈 파라미터에 설정된 운전 방식이 적용됩니다.
- 데이터 클리어: 데이터 메모리에 저장되어 있는 데이터를 클리어 합니다. 이 때, 래치로 설정된 데이터 메모리 영역의 데이터는 클리어하지 않고 유지합니다.
- 입, 출력 모듈 주소 할당 및 종류 등록: 입, 출력 모듈의 데이터 메모리 주소를 할당하고, 모듈의 종류를 등록합니다.
- 초기화 Task 처리: 초기화 Task 프로그램이 등록된 경우 초기화 Task 프로그램의 연산을 실행합니다.

1.3. 연산 처리



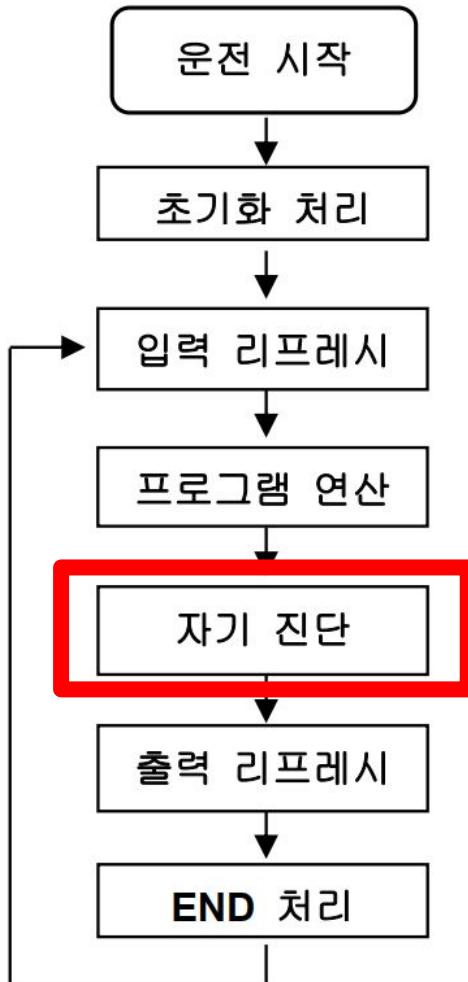
(2) 입력 리프레시: 디지털 입력 모듈의 입력 상태를 읽어 PLC 데이터 메모리의 입력 영역에 저장합니다. 입력 리프레시 과정을 거쳐 입력 데이터 메모리 영역에 저장된 입력 정보는 그스캔이 완료될 때까지 변경되지 않습니다

1.3. 연산 처리



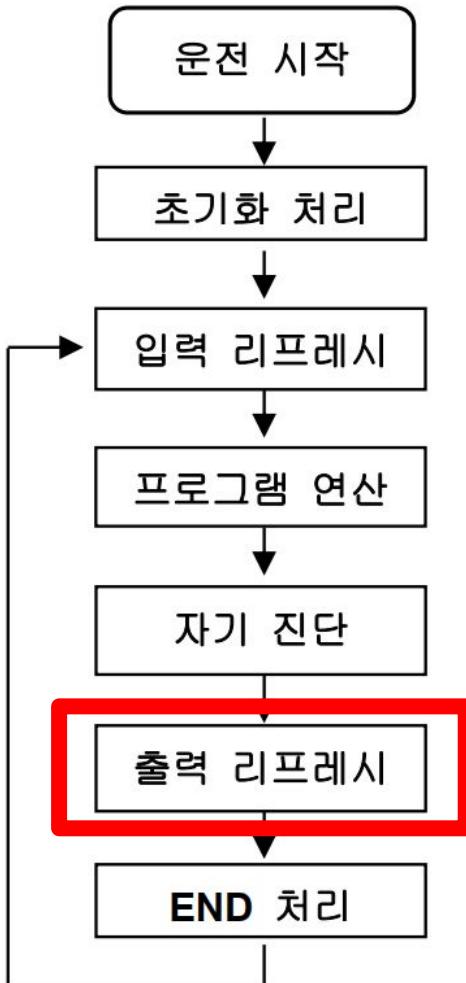
(3) 프로그램 연산: 사용자가 작성한 프로그램을 해석하여 데이터 메모리에 저장된 데이터를 처리합니다. 프로그램은 스캔 프로그램과 Task 프로그램으로 나누어 지며, 스캔 프로그램은 PLC 가 RUN 상태이면 수행하는 프로그램이며, Task 프로그램은 PLC 가 RUN 상태에서 Task 의 조건으로 지정된 조건이 만족될 때 수행하는 프로그램이며 내부 디바이스 Task 는 지정된 내부디바이스의 조건이 만족된 스캔의 스캔 프로그램 수행이 완료된 후 수행되는 프로그램입니다. 프로그램 연산 과정에서 발생하는 출력 데이터는 데이터 메모리의 출력 영역에 저장됩니다.

1.3. 연산 처리



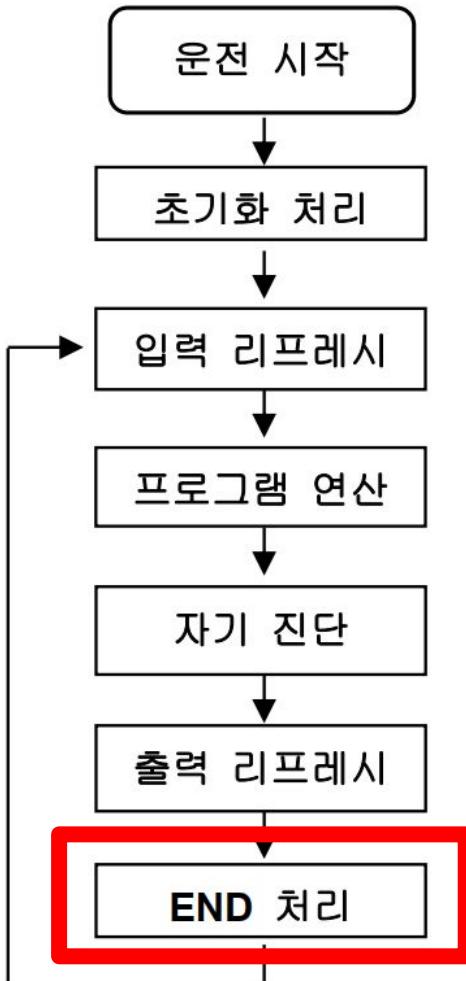
(4) 자기 진단: PLC 시스템을 진단하여 에러 존재 여부를 판단합니다. 에러가 감지되면 PLC는 RUN 상태가 되지 않고 에러 상태로 됩니다.

1.3. 연산 처리



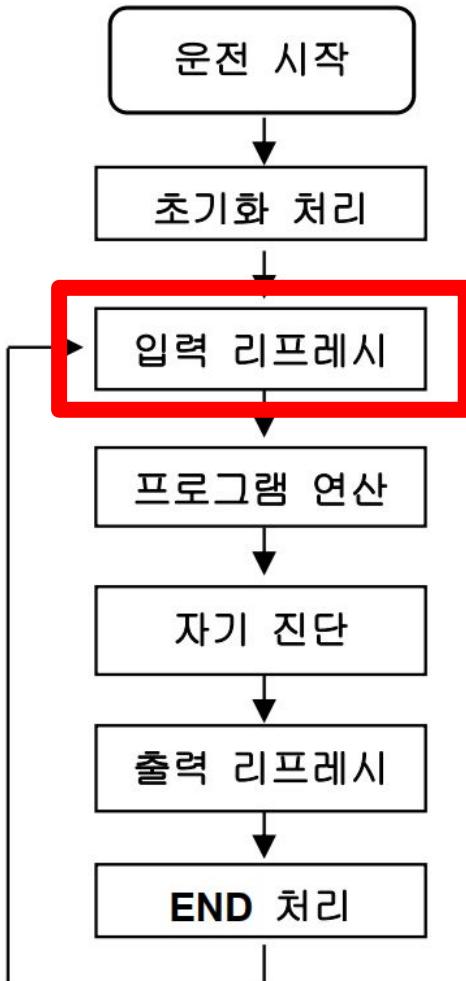
(5) 출력 리프레시: 데이터 메모리의 출력 영역에 저장된 데이터를 출력 모듈을 통해 출력합니다. 출력 상태의 갱신은 이 때 이루어집니다.

1.3. 연산 처리



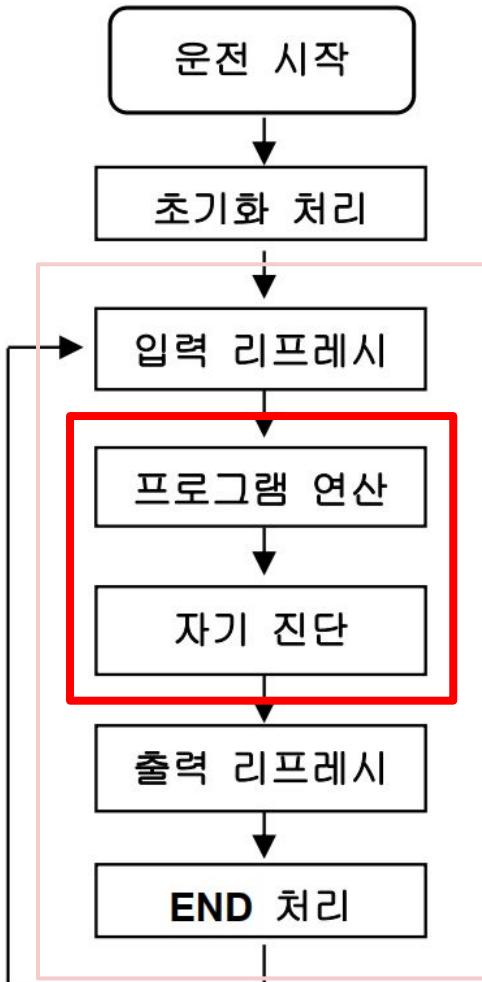
(6) END 처리: 1 스캔의 처리를 종료한 후 스캔의 처음으로 돌아가기 위한 처리를 합니다.

1.3. 연산 처리



(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.

1.3. 연산 처리



1 Scan: 프로그램을 수행하기 전에 입력 Unit에서 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.

PLC 는 래더(Ladder) 언어를 일반적으로 제공하여 사용 합니다



1.4. PLC 프로그래밍 언어

(1) IL(Instruction List)

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

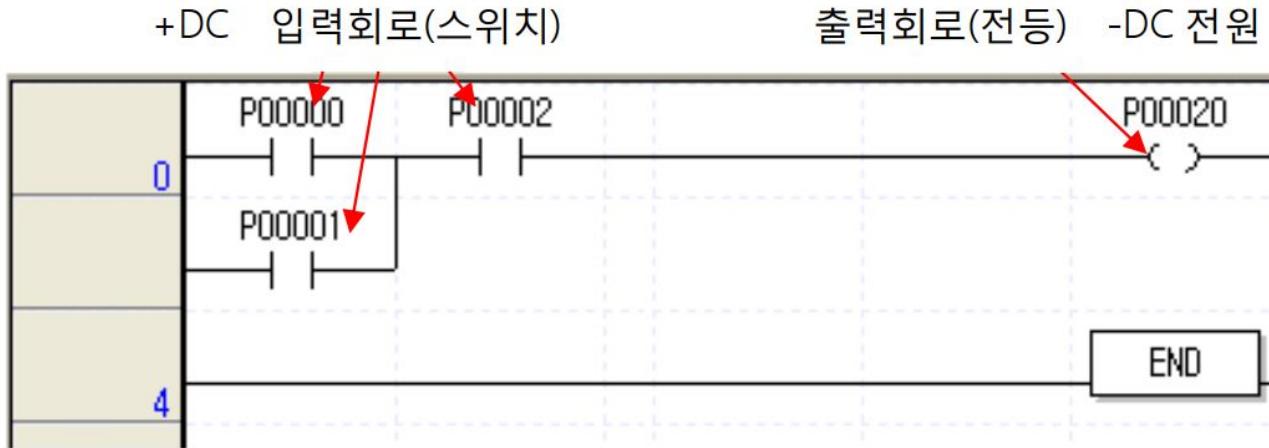


번호	명령어	입력 파라미터	변수	변수 설명문
1	LOAD		%IX0.0.0	
2	OR		%IX0.0.1	
3	AND		%IX0.0.2	
4	OUT		%QX0.1.0	
5	LOAD NOT		%IX0.0.3	
6	OUT		%QX0.1.1	

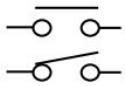
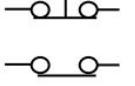
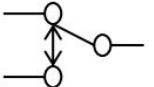
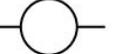
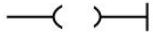
1.4. PLC 프로그래밍 언어

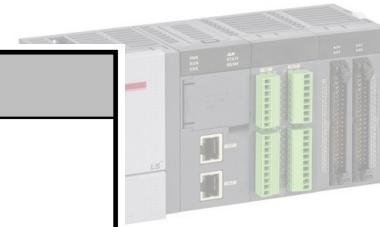
(2) 래더(Ladder): 사다리도

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



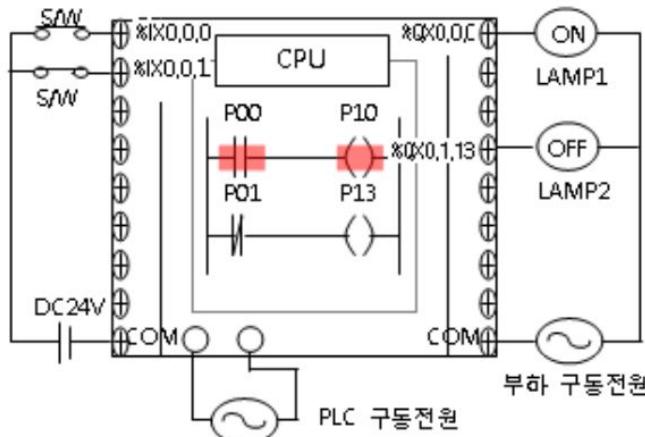
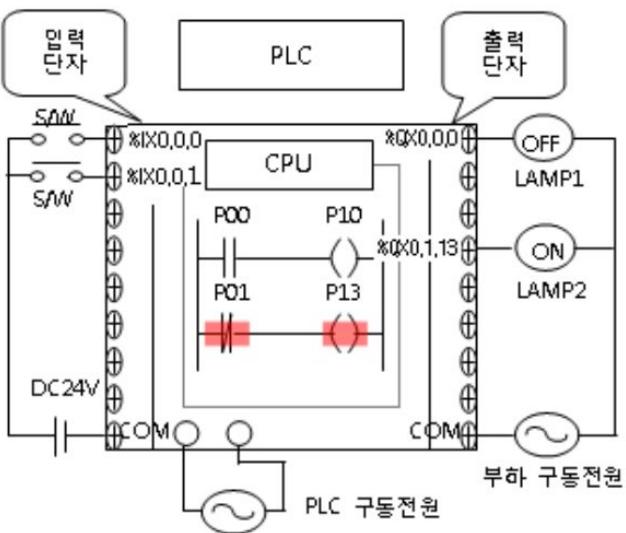
1.5. PLC 동작이해

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



1.5. PLC 동작이해

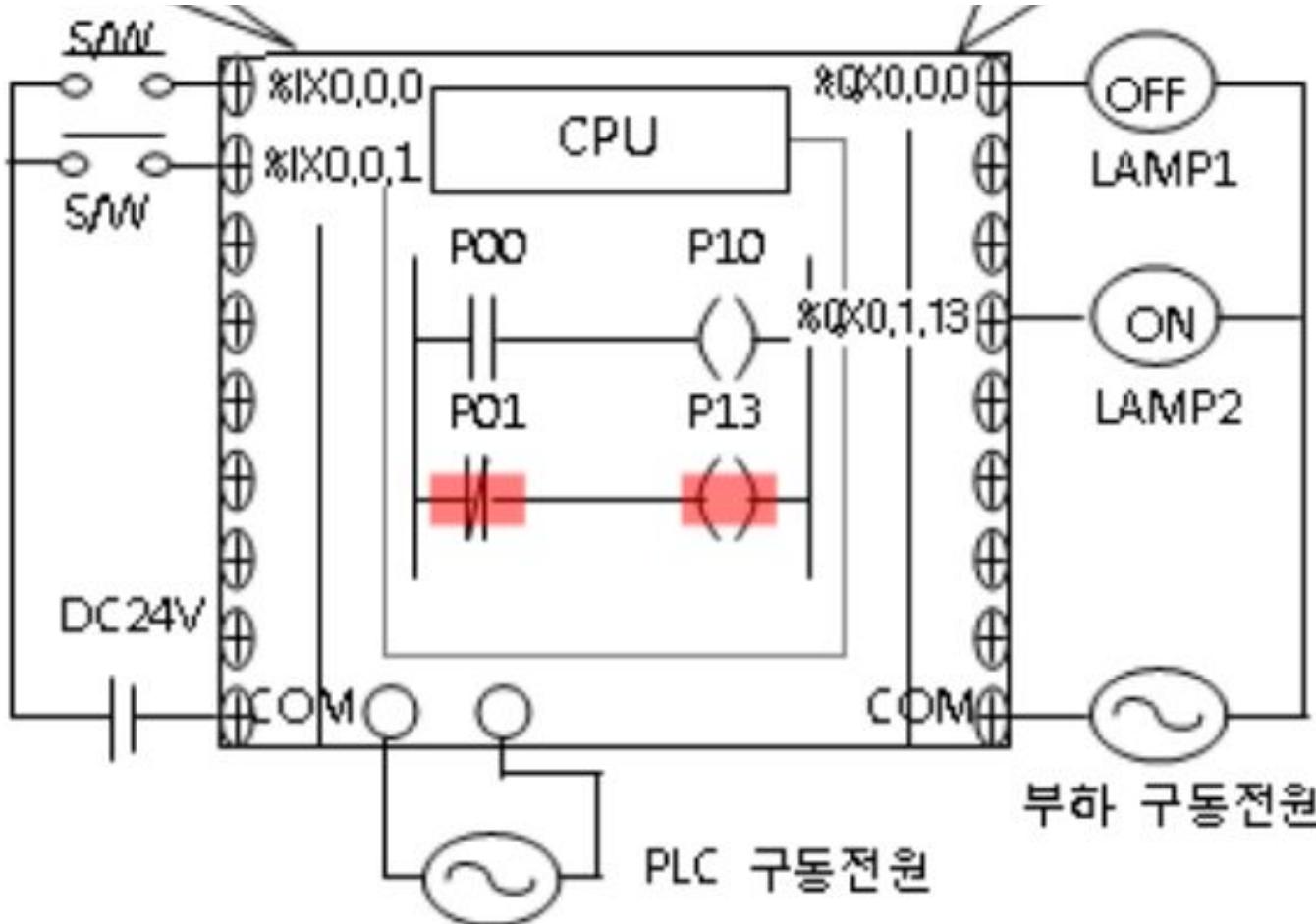
- 아래 그림은 PLC 기본 구성을 간략화한 것으로 외부접점과 PLC연산 관계에 대하여 설명합니다. 실선내부는 PLC의 CPU에 저장되어 동작되는 프로그램으로 프로그램 Loader 를 이용하여 입력하면 됩니다. 입력단자와 COM단자 사이에 DC24V를 인가해 주면 입력이 형성 됩니다. 출력단자와 COM단자 사이에 부하 (LAMP)를 연결하고 부하구동전원을 연결하면 됩니다. (DC부하일 경우 부하구동



1.5. PLC 동작이해



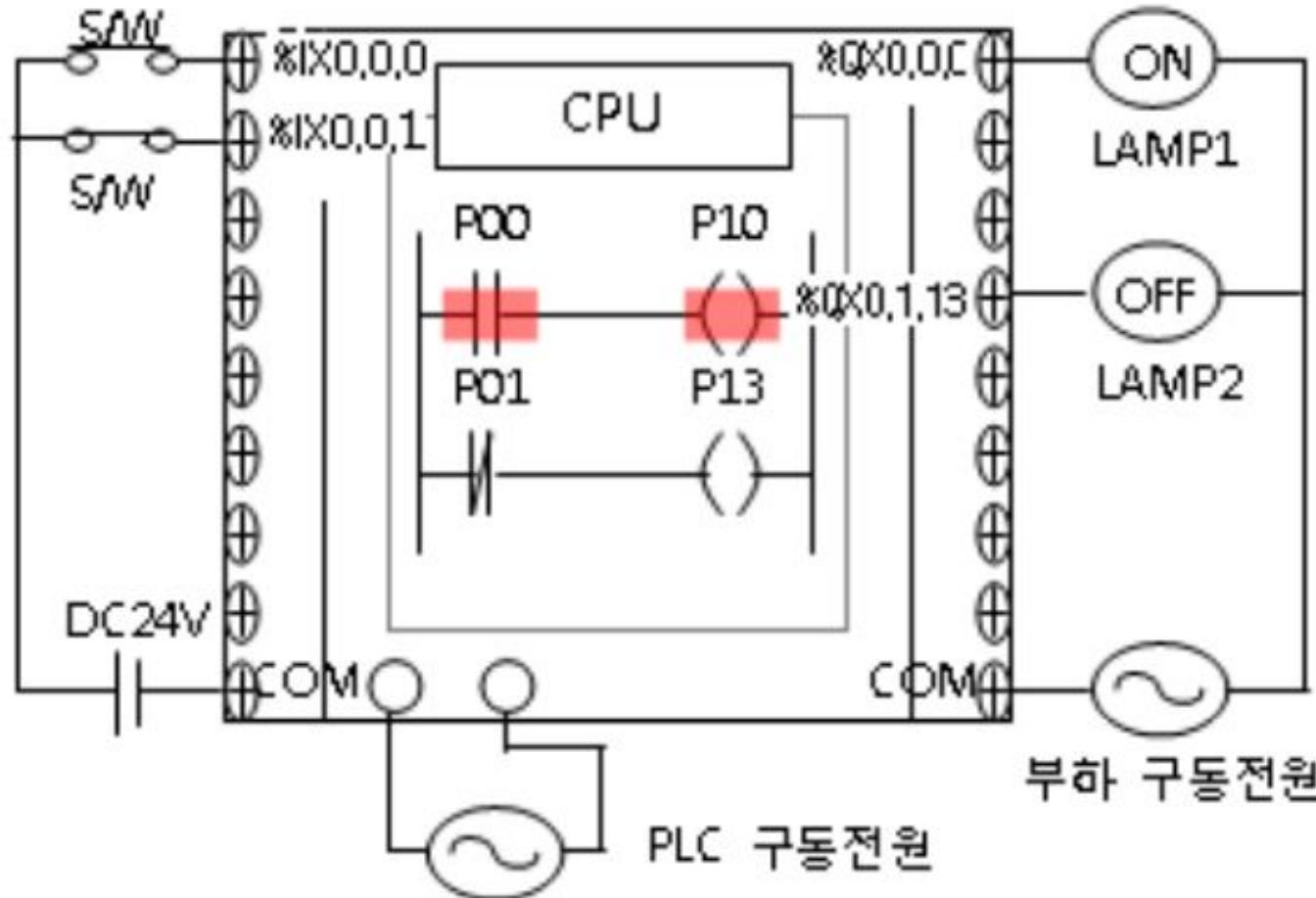
접점의 연결과 출력
상태 :
점접 닫힘(Closed, 연결)



1.5. PLC 동작이해



접점의 연결과 출력
상태 :
점접 닫힘(Closed, 연결)



1.6. PLC 용어정리



- 점(Point): 입력 8점, 출력 6점의 PLC는 스위치나 센서 등 입력기기를 최대 8개, 램프나 릴레이 등 출력기기를 최대 6개까지 연결 가능. PLC의 입출력 용량을 표시할 때 사용
- 스텝(Step): PLC 명령어의 최소 단위로 A접점, B접점, 출력 코일 등의 명령이 1스텝에 해당하는 명령이고 기타 응용 명령어의 경우 하나의 명령어가 다수의 스텝을 점유합니다. 프로그램 용량 및 CPU 속도를 표시하는 단위로 사용 (용량: 30k step, 속도: step/sec)
- 스캔타임(Scan Time): 사용자 작성 프로그램의 1회 수행에 걸리는 시간을 의미합니다. 스텝수가 많으면 스캔 타임은 증가



1.6. PLC 용어정리



- WDT(Watch Dog Timer): 프로그램 연산 폭주나 CPU기능고장에 의하여 출력을 하지 못할 경우 설정한 시간(WDT) 대기 후 에러를 발생시키는 시스템 감시 타이머입니다. 기본 200ms로 설정되어 있으며 파라미터 지정에 의해 변경시킬 수 있습니다.
- 파라미터(Parameter): 프로그램과 함께 PLC에 저장되는 운전 데이터로 통신, 시스템 환경 등을 저장합니다

