



과정명	PLC 제어응용	차시명	2 차시. 점점 명령
-----	----------	-----	-------------

■ 학습목표

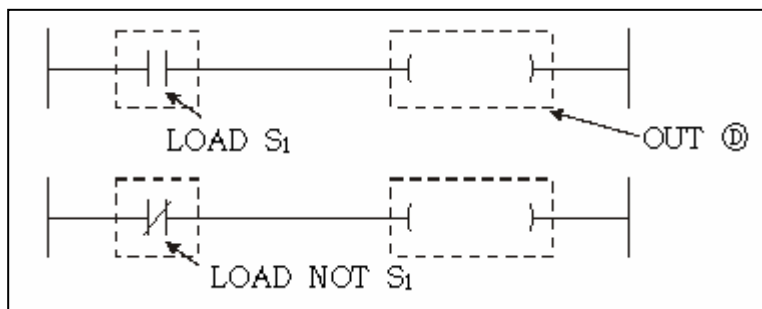
1. 연산개시 명령(Load, Load Not)에 대하여 설명할 수 있다.
2. 직렬접속 명령(AND, AND NOT)에 대하여 설명할 수 있다.
3. 병렬접속 명령(OR, OR NOT)에 대하여 설명할 수 있다.
4. PLC의 점점 명령을 가지고 간단한 프로그램을 작성할 수 있다.

■ 학습내용

1. 연산개시 명령

1) 연산개시명령(Load, Load Not)

- Load : a 점점 연산개시
- Load Not : b 점점 연산개시
- OUT : 연산결과 출력



(1) Load S₁

- ① 한 회로의 a 점점 연산개시
- ② 지정 점점(S₁)의 On/Off 정보를 연산결과로 한다.

(2) Load Not S₁

- ① 한 회로의 b 점점 연산개시
- ② 지정 점점(S₁)의 On/Off 정보를 연산결과로 한다.

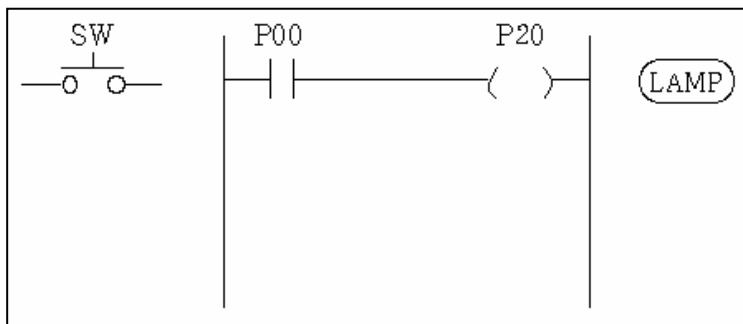


(3) OUT ①

- ① OUT 명령까지의 연산결과를 지정한 접점에 출력한다.
- ② OUT 명령은 병렬 사용이 가능하다.

2) a 점점연산개시(LOAD) 동작이해

■ a 점점연산개시(LOAD) 동작이해



◆스위치 SW 누르기 전(OFF) → a 점점인 프로그램 P00 은 단선
출력코일 P20 OFF → 램프 OFF

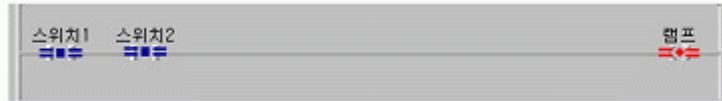
◆스위치 SW 누르면
스위치 SW ON → 프로그램 P00 연결
출력코일 P20 ON → 램프 ON

◆용어설명

- 연결(Connect) : 접점이 붙은
- 단선(Disconnect) : 접점이 떨어진



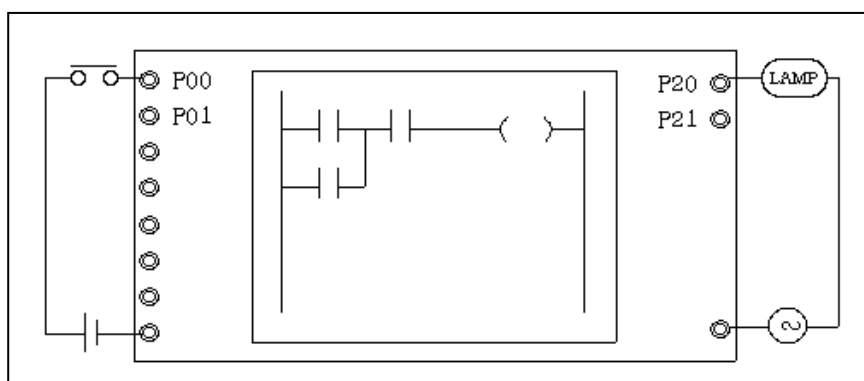
<그림 참조>



3) PLC 시스템 구성원리 이해

■ PLC 시스템 구성원리 이해

- ① 점선 내부는 PLC 의 CPU 에 저장되어 동작되는 프로그램으로 프로그램 로더를 이용하여 입력하면 됩니다
- ② CPU 에 저장되어 있는 프로그램은 PLC 의 입력단자와 출력단자에 연결되어 있고, 입출력단자에 스위치와 같은 입력기기, 램프와 같은 출력기기가 접속하게 됩니다
(a 점점과 b 점점과 같은 입력점점은 입력단자에, 출력코일은 출력단자에 연결)
- ③ 입력단자와 COM 단자 사이에 DC24V 를 인가해 주면 입력이 형성됩니다
- ④ 출력단자와 COM 단자 사이에 부하(LAMP)를 연결하고 부하구동전원을 연결하면 됩니다
(DC 부하일 경우 부하구동전원은 DC 전원이 사용됩니다)

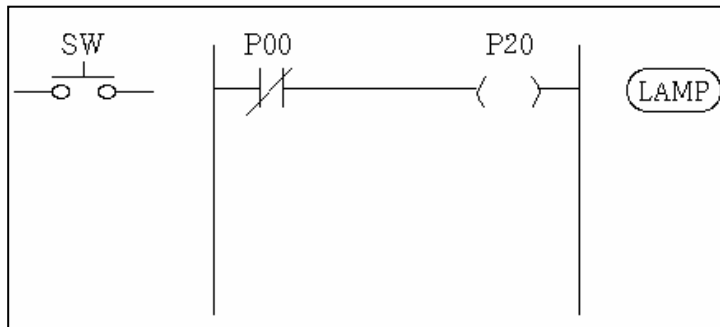


<PLC 그림>



4) b 점점연산개시(Load NOT) 동작 이해

■ b 점점연산개시(Load NOT) 동작 이해



◆스위치 SW 누르기 전(OFF) → b 점점인 프로그램 P00 은 연결
→ 출력코일 P20 ON → 램프 ON

◆스위치 SW 누르면
스위치 SW ON → 프로그램 P00 단선(Disconnect : b 점점이므로)
→ 출력코일 P20 OFF → 램프 OFF

{ 설명 }

입력부 P00 단자에 접속되어 있는 푸시버튼 스위치 SW 를 누르게 되면 PLC 내부 메모리에 프로그램되어 있는 P00 이 단선 되면서 출력코일 P20 이 OFF 되고, 출력부의 P20 단자에 접속되어 있는 램프가 소등하게 됩니다



5 시퀀스 연산자

: 시퀀스 연산자에는 접점(Contact), 코일(Coil) 등이 있다.

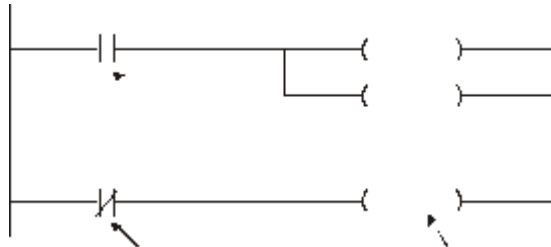
No.	기호	이름	기 능
1		a 접점 평상시 열린 접점 (Normally Open Contact)	BOOL 변수의 상태가 On 일 때에 해당접점 연결
2		b 접점 평상시 닫힌 접점 (Normally Closed Contact)	BOOL 변수의 상태가 Off 일 때에 해당접점 연결
3		출력 코일(Coil)	외쪽에 있는 연결선의 상태 를 지정된 BOOL 변수 접점으로 출력

6) 프로그램 예

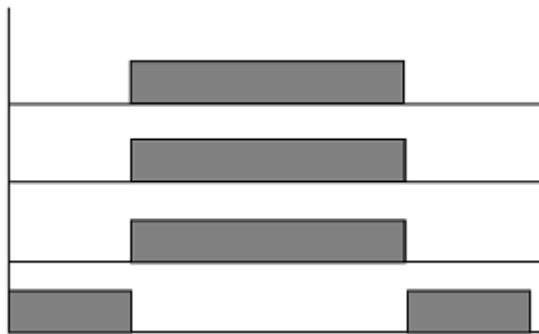
- ① 입력조건 P000 가 On 되면 지정 출력(P0020, P0021)이 모두 On 됨과 동시에 P023 출력은 Off 되는 프로그램



② 프로그램



③ 타임 차트

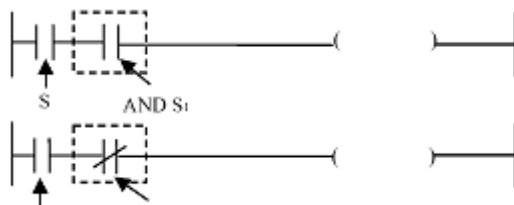


2. 직렬 접속 명령

1) 직렬접속명령(AND, AND NOT)

■ AND : a 점점 직렬접속

■ AND NOT : b 점점 직렬접속





(1) AND

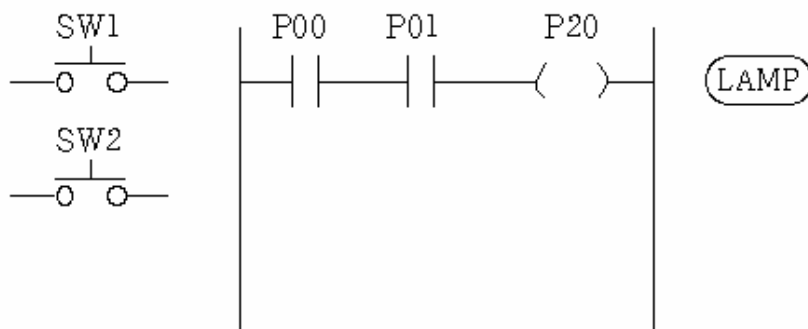
- ① a 접점 직렬 접속 명령이다.
- ② 지정 접점(S_1)의 a 접점을 직렬로 연결하고, S 와 S_1 의 AND 연산값을 연산결과로 한다.

(2) AND NOT

- ① b 접점 직렬 접속 명령이다.
- ② 지정 접점(S_1)의 b 접점을 직렬로 연결하고, S 와 S_1 의 AND 연산값을 연산결과로 한다.

2) a 접점 직렬접속명령(AND) 동작이해

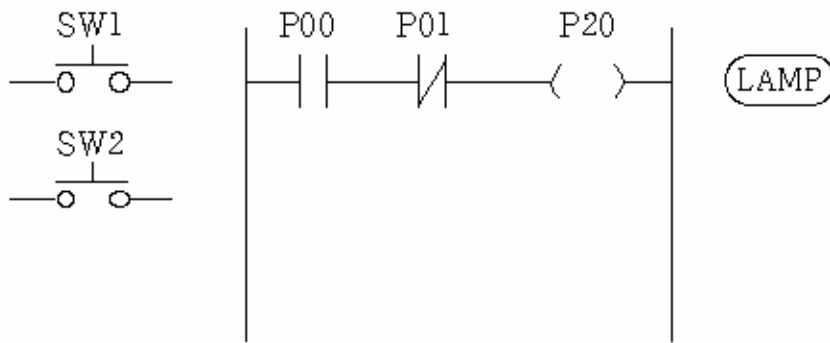
■ a 접점 직렬접속명령(AND) 동작이해



- 스위치 SW1 ON → 프로그램 P00 연결 → 프로그램 P01 단선
→ 출력코일 P20 OFF → 램프 OFF
- 스위치 SW2 ON → 프로그램 P00 단선 → 프로그램 P01 연결
→ 출력코일 P20 OFF → 램프 OFF
- 스위치 SW1 과 SW2 가 ON → 프로그램 P00 연결 → 프로그램 P01 연결
→ 출력코일 P20 ON → 램프 ON

3) b 접점 직렬접속명령(AND NOT) 동작이해

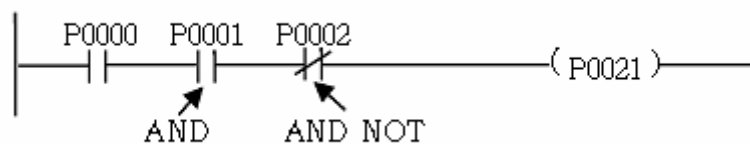
■ b 접점 직렬접속명령(AND NOT) 동작이해



- 스위치 SW1 ON → 프로그램 P00 연결 → 프로그램 P01 연결
→ 출력코일 P20 ON → 램프 ON
- 스위치 SW2 ON → 프로그램 P00 단선 → 프로그램 P01 단선
→ 출력코일 P20 OFF → 램프 OFF
- 스위치 SW1 과 SW2 가 ON → 프로그램 P00 연결 → 프로그램 P01 단선
→ 출력코일 P20 OFF → 램프 OFF

4) 프로그램 예

- ① 입력 조건 P0000 과 P0021 을 AND 연산하고 그 결과와 P0002 를 AND NOT 연산하여 P0021 에 출력하는 프로그램



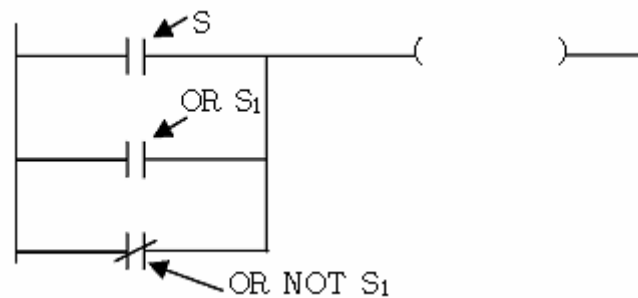


3. 병렬 접속 명령

1) 병렬접속명령(OR, OR NOT)

■ OR : a 접점 병렬접속

■ OR NOT : b 접점 병렬접속



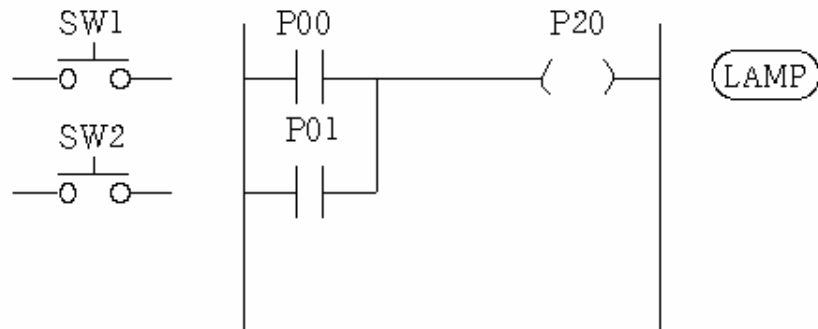
(1) OR

- ① a 접점 병렬 접속 명령이다.
- ② 지정 접점(S_1)의 a 접점을 병렬로 연결하고, S 와 S_1 의 OR 연산값을 연산결과로 한다.

(2) OR NOT

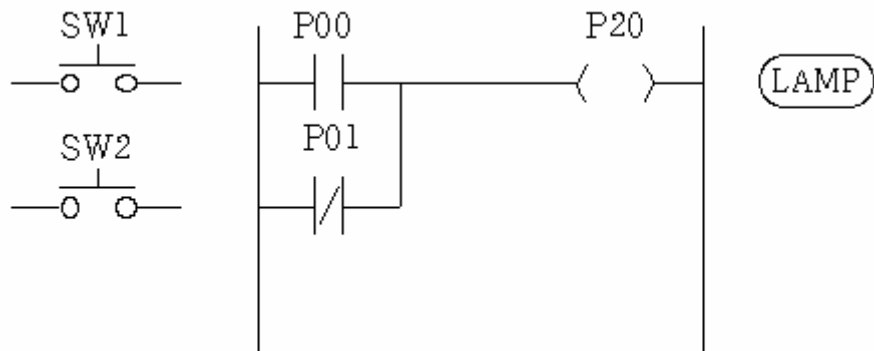
- ① b 접점 병렬 접속 명령이다.
- ② 지정 접점(S_1)의 b 접점을 병렬로 연결하고, S 와 S_1 의 OR 연산값을 연산결과로 한다.

2) a 접점 병렬접속명령(OR) 동작이해



3) b 접점 병렬접속명령(OR NOT) 동작이해

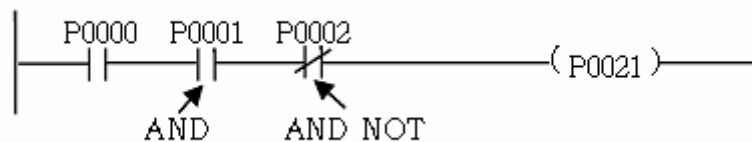
■ b 접점 병렬접속명령(OR NOT) 동작이해



- 스위치 SW1 ON → 프로그램 P00 연결 → 출력코일 P20 ON → 램프 ON
- 스위치 SW2 ON → 프로그램 P01 단선 → 출력코일 P20 OFF → 램프 OFF
- 스위치 SW1 과 SW2 가 ON → 프로그램 P00 연결 → 프로그램 P01 단선
→ 출력코일 P20 ON → 램프 ON

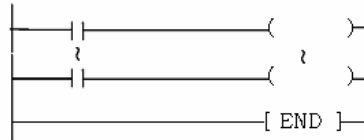
4) 프로그램 예

① 입력 조건 P0000, P0001 중 하나의 접점만 On 되어도 P0021 이 출력되는 프로그램



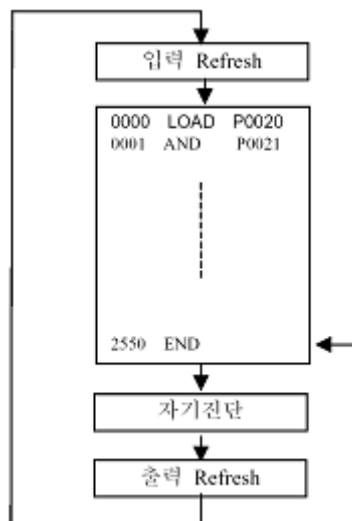


4. 종료 명령 (END)



(1) END

- ① 프로그램 종료를 표시한다.
- ② END 명령 처리 후 0000 스텝으로 돌아가 처리한다.
- ③ END 명령은 반드시 프로그램의 마지막에 입력한다.(제품중 MASTER-K 인 경우 입력하지 않으면 Error 발생)



● 참고

PLC 제품마다 차이가 있겠지만 GLOFA-GM 프로그램에서는 프로그램 종료를 표시(END)를 하지 않아도 됨



■ 학습정리

1. 연산개시 명령(LOAD, LOAD NOT)

- LOAD : 한 회로의 a접점 연산개시
- LOAD NOT : 한 회로의 b접점 연산개시
- OUT : 명령까지의 연산결과를 지정한 접점에 출력한다.

2. 직렬접속 명령(AND, AND NOT)

- AND : a접점 직렬접속
- AND NOT : b접점 직렬접속

3. 병렬접속 명령(OR, OR NOT)

- OR : a접점 병렬접속
- OR NOT : b접점 병렬접속