

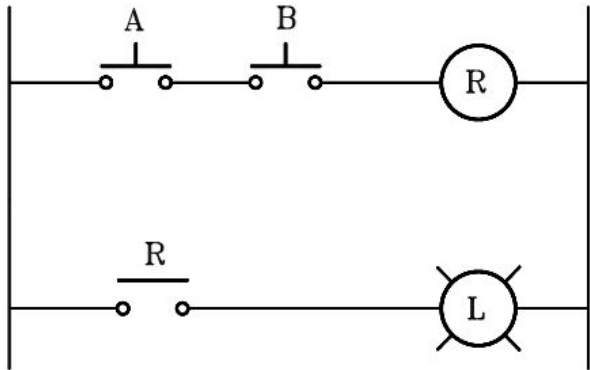
#PLC #자동제어

GX WORKS

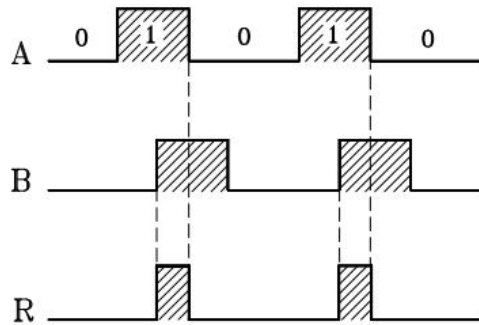
**PLC - 멜섹**

**프로그래밍 기초 복습**

# ex-01. 직렬회로, AND



(a) 릴레이 회로



(b) 타임 차트

입 력		출 력
A	B	R
0	0	0
0	1	0
1	0	0
1	1	1

(c) 진리표

# ex-01. 직렬회로, AND

여러 개의 입력이 있을 때 모든 입력이 존재할 때에만 출력이 나타나는 회로를 AND회로라고 하며 직렬 스위치 회로와 같다. 두 개의 입력 A와 B가 모두 ON일 때에만 릴레이 코일 R이 여자 되고 R접점이 닫혀 램프가 점등되는 AND회로이다.

이와 같은 직렬회로는 한 대의 프레스에 여러 명의 작업자가 함께 작업할 때, 안전을 위해 각 작업자마다 프레스 기동용 누름버튼을 설치하여 모든 작업자가 스위치를 누를 때에만 동작 되도록하는 경우에 적용된다. 또 기계의 각 부분이 소정의 위치까지 진행되지 않으면 다음 동작으로 이행을 금지하는 경우 등 그 응용 범위가 넓은 회로이다.

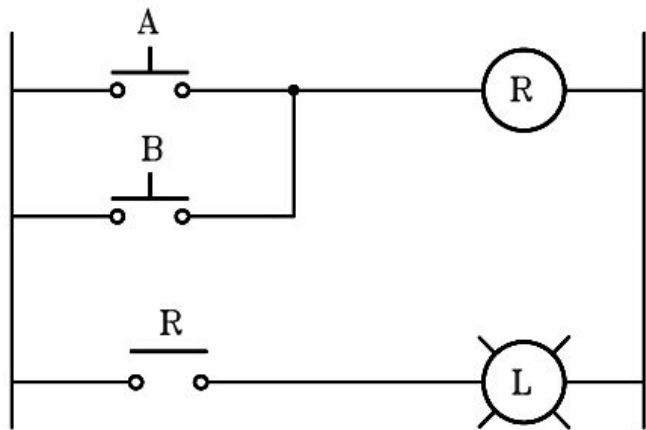
# ex-01. 직렬회로, AND

입력 2개 가 모두 ON 일때 출력이 ON 되는 회로 구성

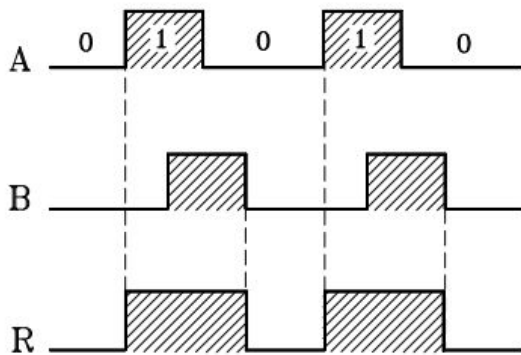


# ex-02. 병렬회로

OR회로는 여러 개의 입력신호 중 하나 또는 그 이상의 신호가 ON되었을 때 출력을 내는 회로로서 병렬회로라고 한다. 누름버튼 스위치 A가 눌러지거나, 아니면 B가 눌러져도, 또는 A와 B가 동시에 눌러져도 릴레이 R이 동작되어 램프가 점등된다



(a) 릴레이 회로



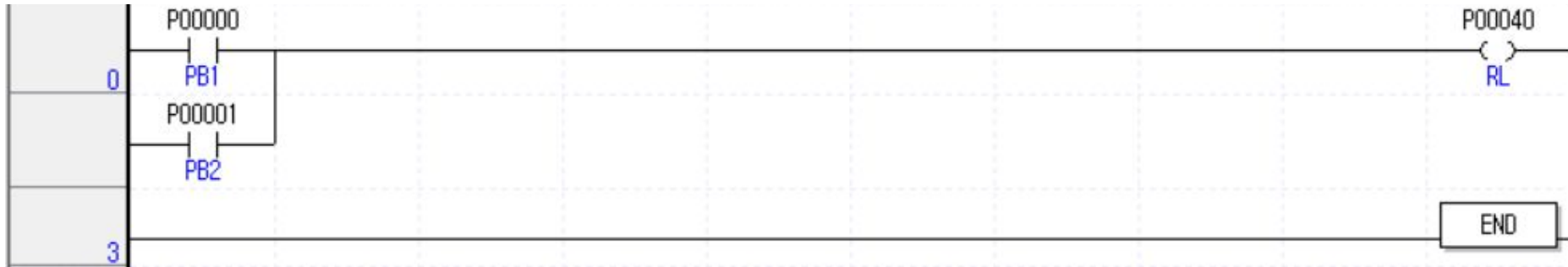
(b) 타임 차트

입 력		출 력
A	B	R
0	0	0
0	1	1
1	0	1
1	1	1

(c) 진리표

# ex-02. 병렬회로

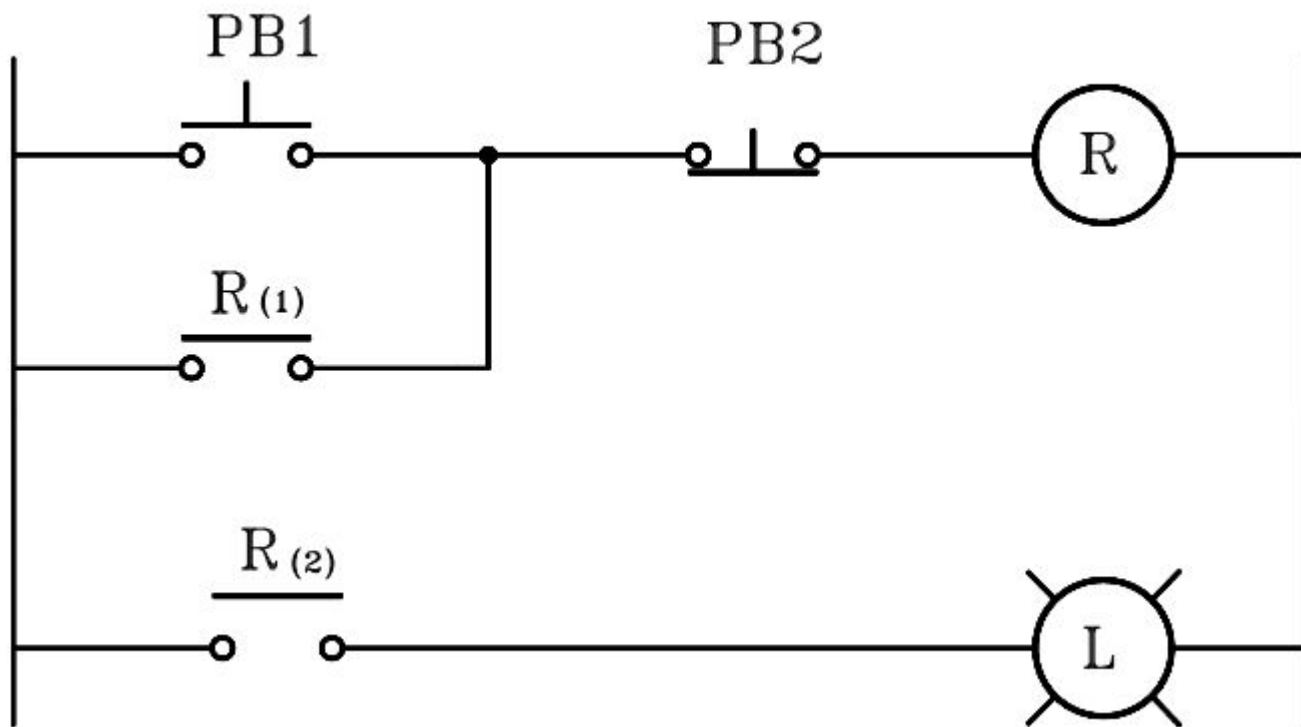
입력 2 중 어느 하나라도 ON 일때 출력이 ON 되는 회로 구성



#PLC #자동제어

# PLC - 자기유지회로

# ex-03. 자기유지회로

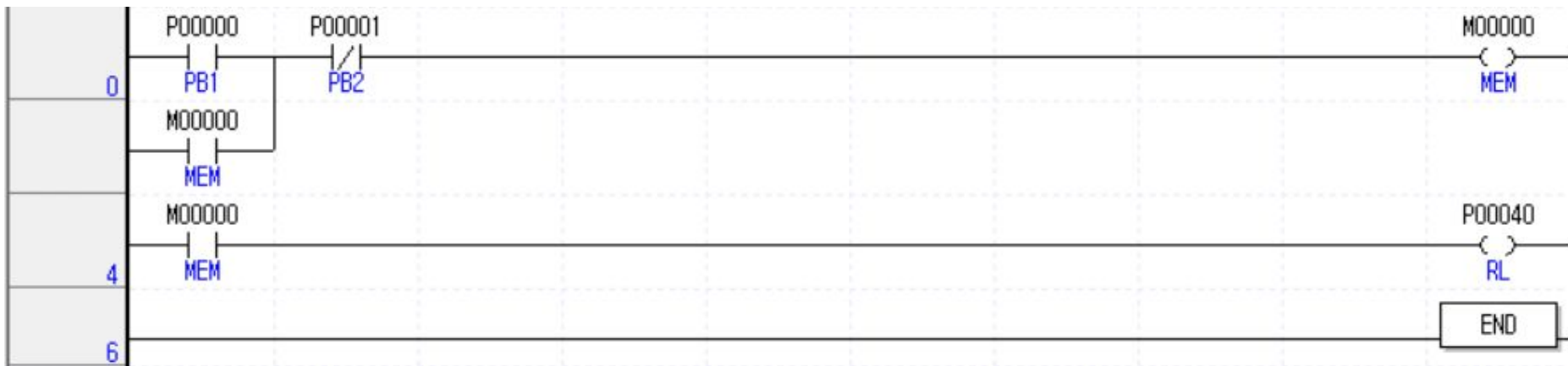




# ex-03. 자기유지회로

전류의 흐름을 유지하는 기능

입력을 한번 ON 하면 이후 입력이 OFF 되더라도 출력이 계속해서 ON 되는 회로

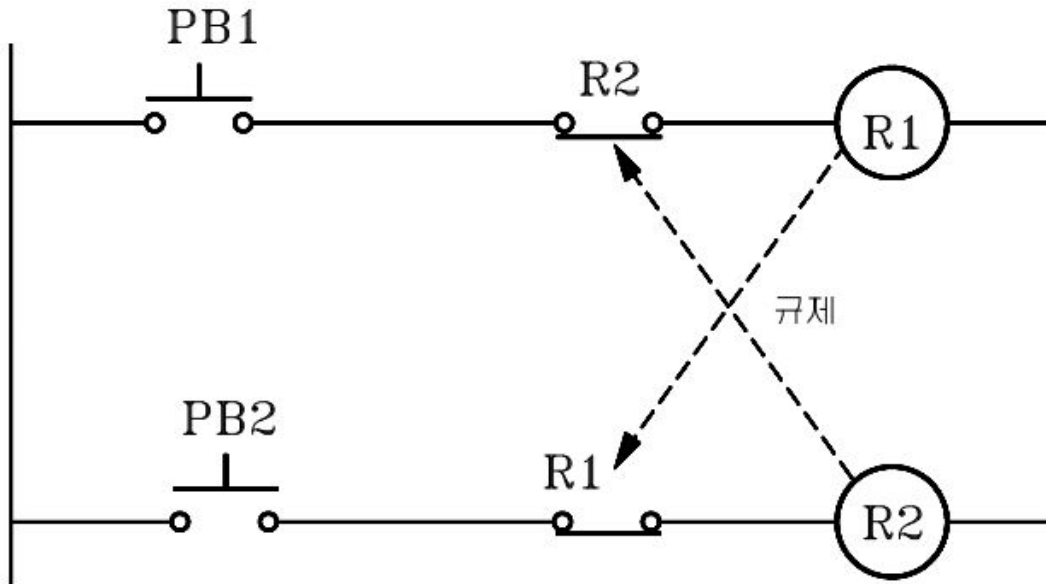


#PLC #자동제어

# PLC - 선입력우선회로 인터록

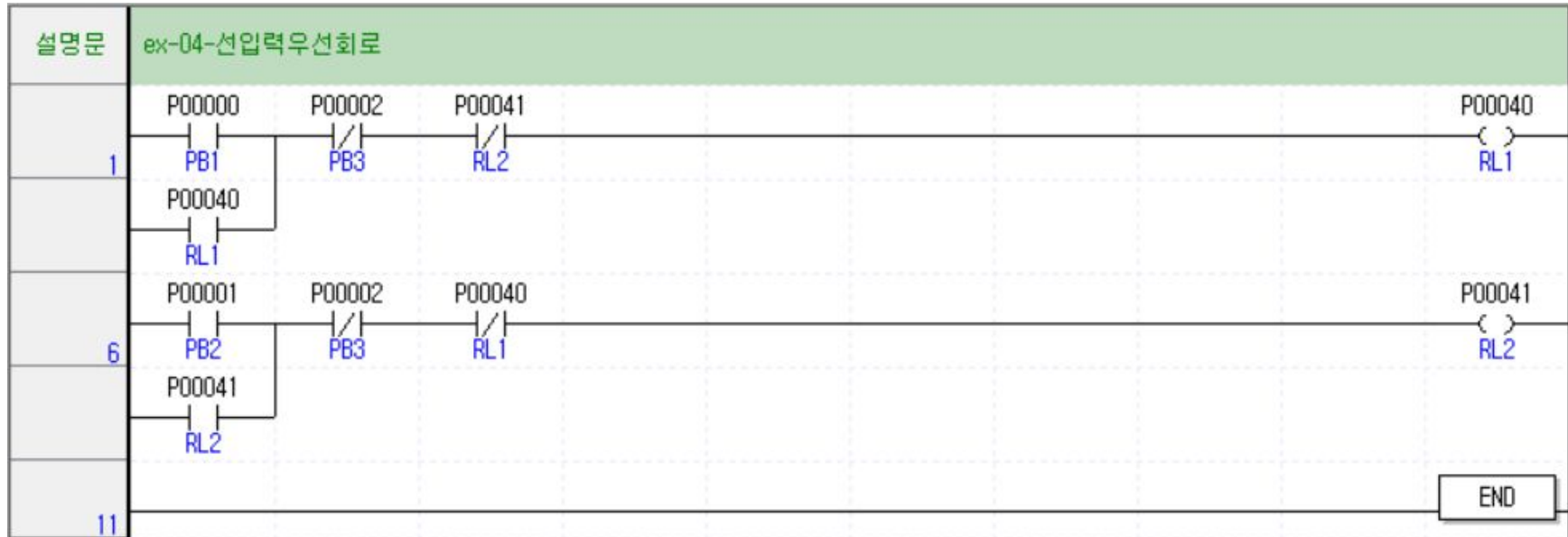
# ex-04. 선입력우선회로

선행 동작 우선 회로  
전동기 정-역 운전에 사용  
인터록회로



# ex-04. 선입력우선회로

선행 동작 우선 회로  
전동기 정-역 운전 에 사용  
인터록 회로



#PLC

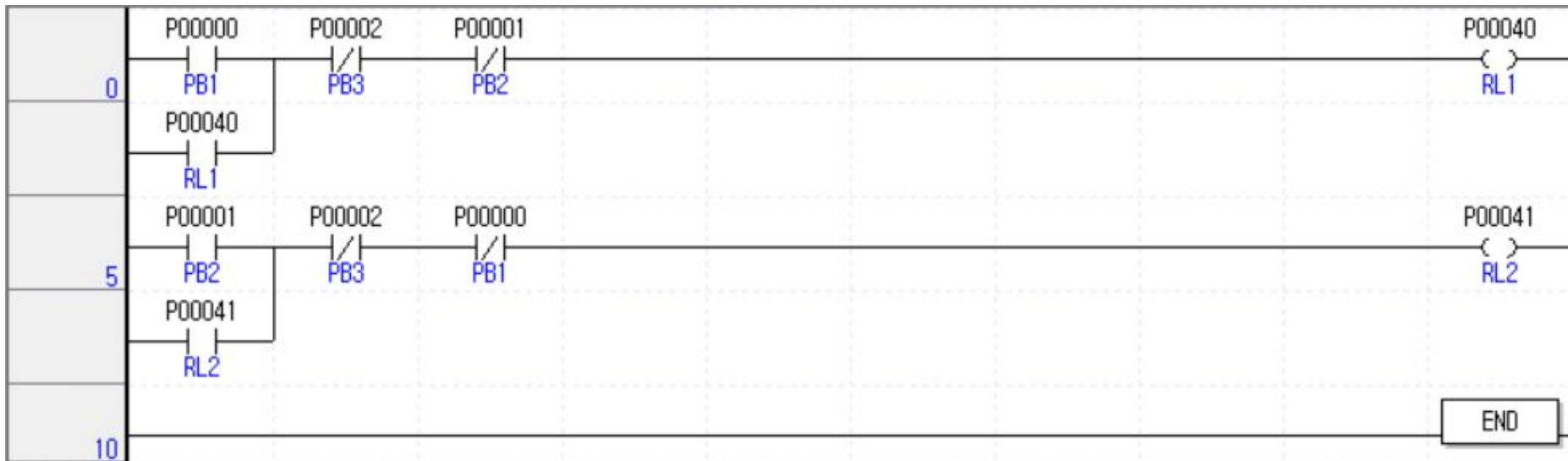
PLC -

후입력우선회로

# ex-05. 후입력우선회로

새로운 입력 신호 우선 동작

A기기 동작시 B 기기 복귀, B 기기 동작 시 A기기 복귀



#PLC

# PLC - 양검출, 음검출

# ex-06. 양검출

양검출은 입력이 들어오는 순간을 검출  
음검출은 입력이 들어온 후 떨어지는 순간을 검출



P00000															
P00040															



# ex-07. 음검출

양검출은 입력이 들어오는 순간을 검출  
음검출은 입력이 들어온 후 떨어지는 순간을 검출

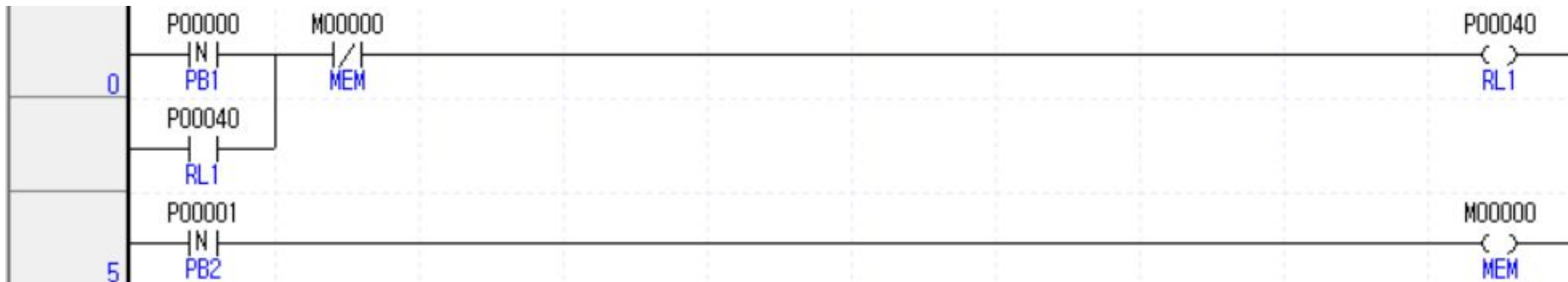
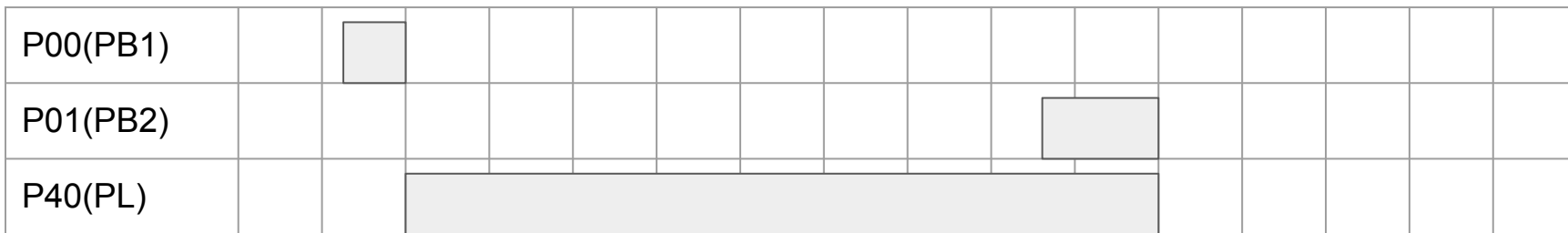


P00001															
P00041															

# ex-08. 음검출 자기유지회로

양검출은 입력이 들어오는 순간을 검출

음검출은 입력이 들어온 후 떨어지는 순간을 검출

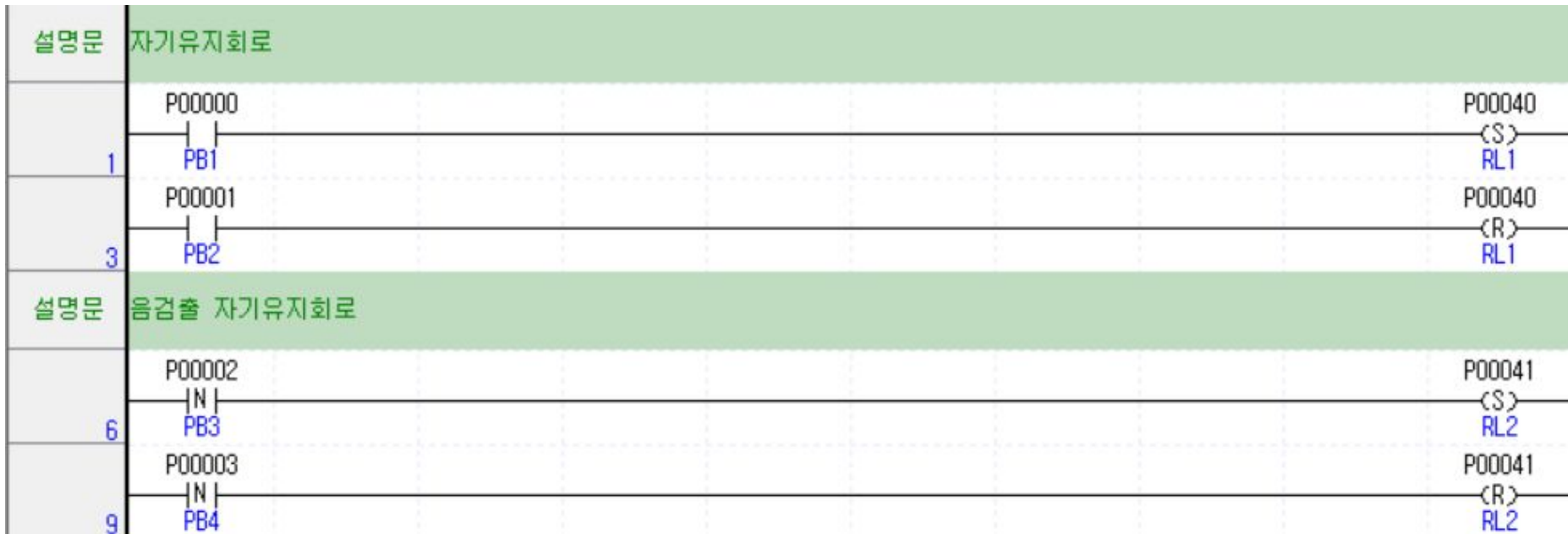


#PLC

# PLC - set, reset

# ex-09. set, reset

자기유지회로를 set, reset 을 이용하여 구현

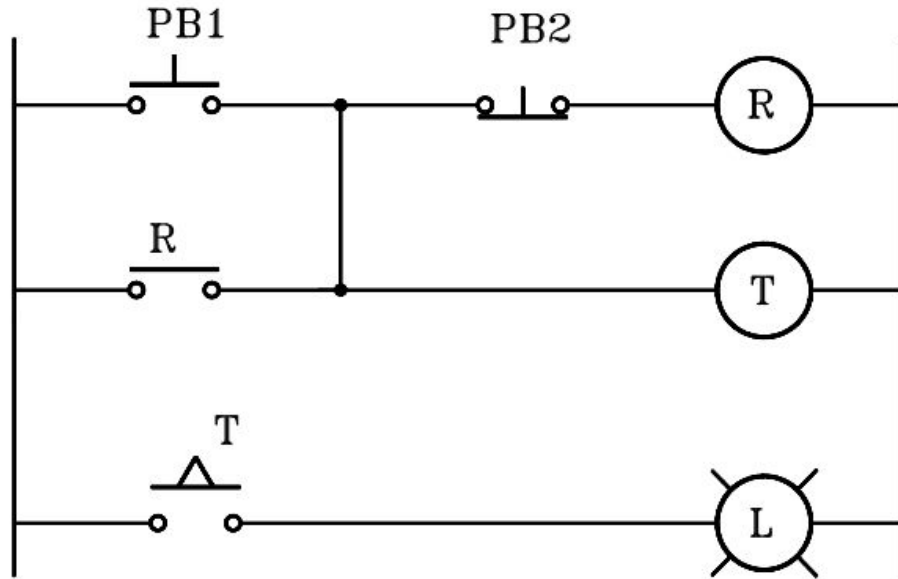


# 타이머 명령

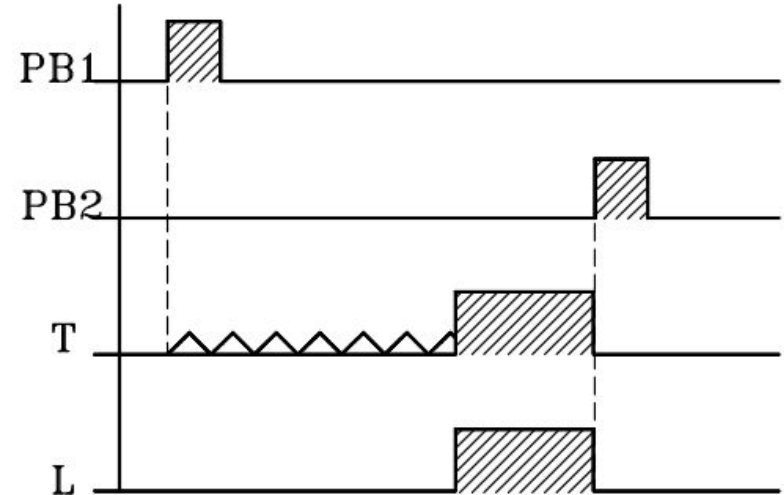
#PLC

# PLC - 타이머

# ex-10. 타이머 TON



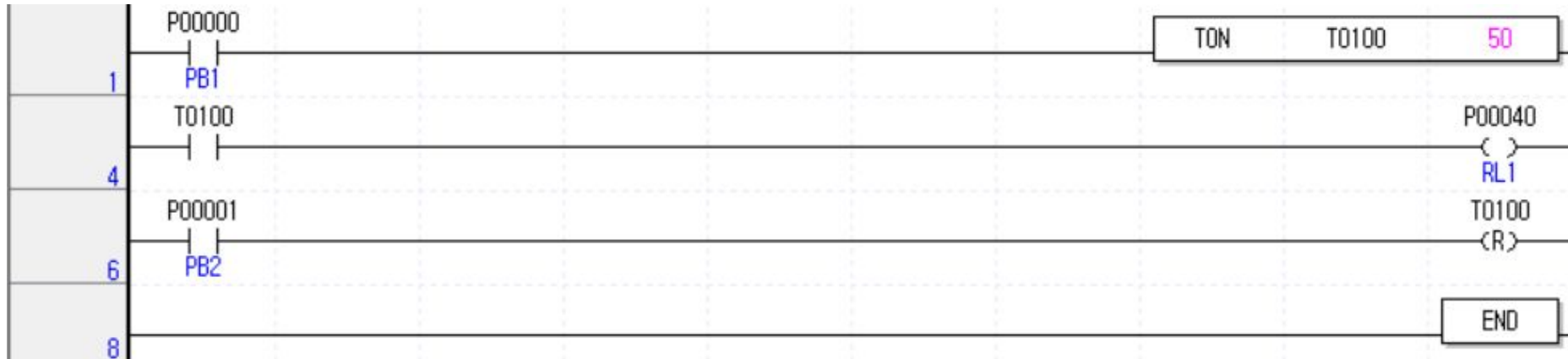
(a) 회로도



(b) 타임 차트

# ex-10. 타이머 TON

[F10] - TON T1 50  
100ms, 10ms, 1ms







#PLC

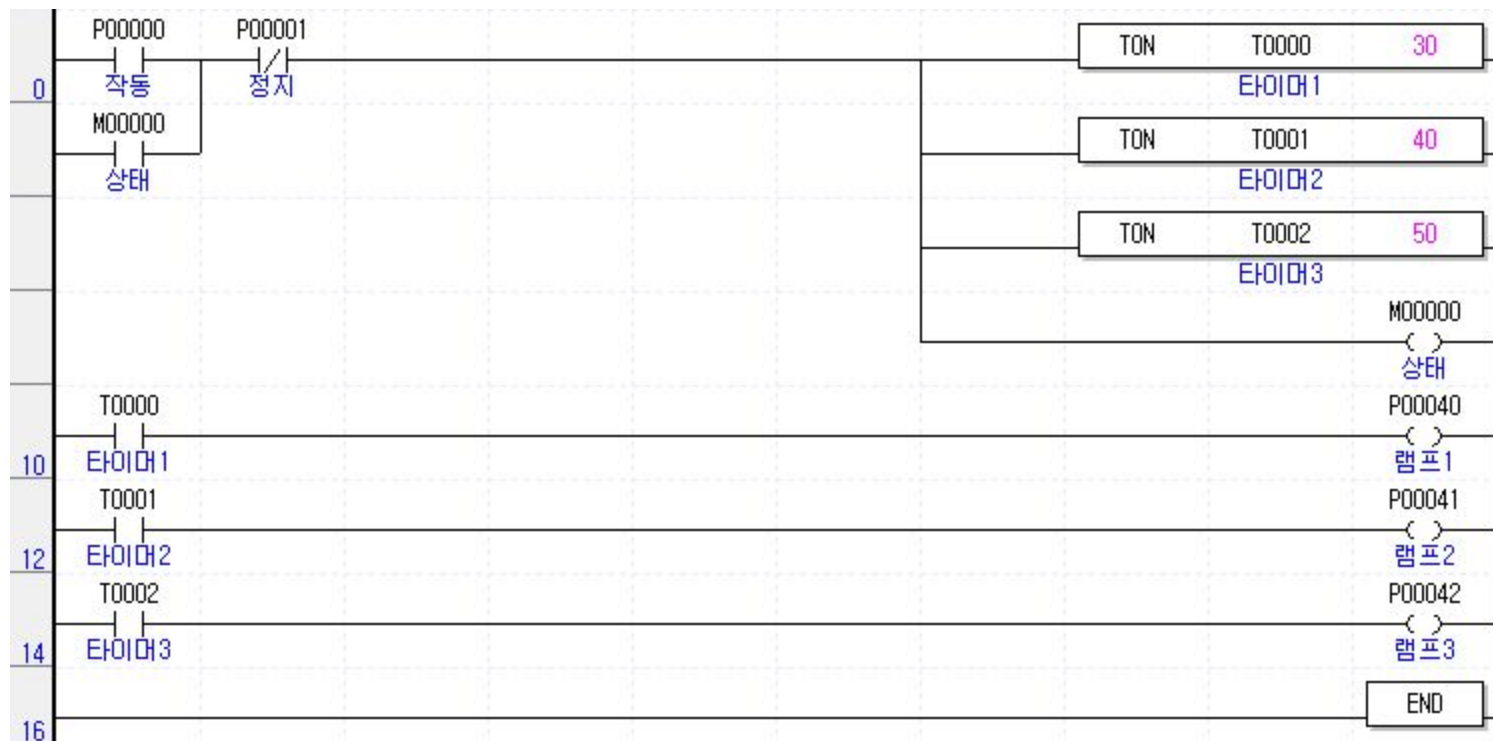
# PLC - 과제 풀이

# 실습과제

공장에 있는 기기를 순차적으로 작동시키려고 한다.

이때 [작동] 스위치 P0 를 한번만 눌러도 P40, P41, P42 가 각각 3초, 4초, 5초 후에 시작되도록 프로그램하시오.

[정지] 스위치 P1 을 누르면 모든 기기가 바로 중단되도록 프로그램하시오.



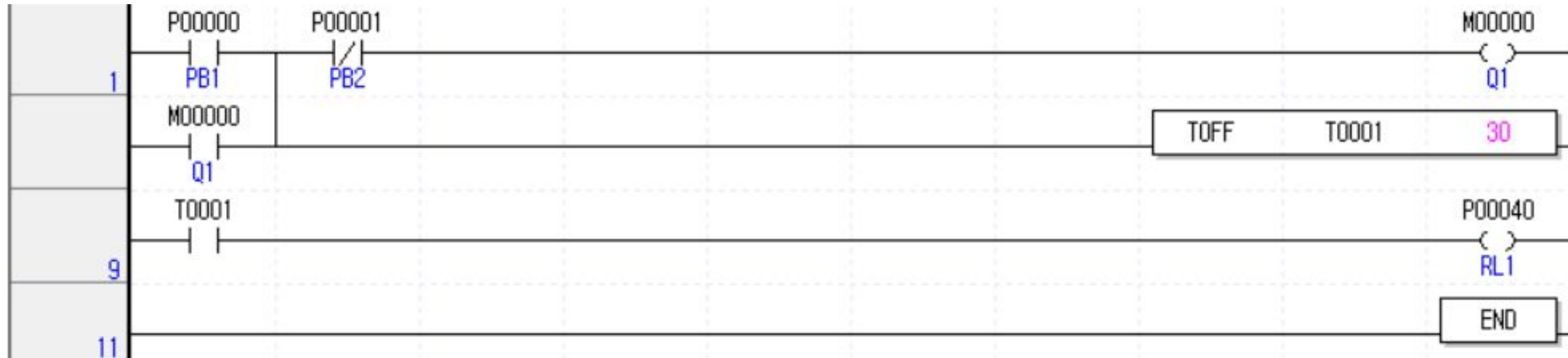
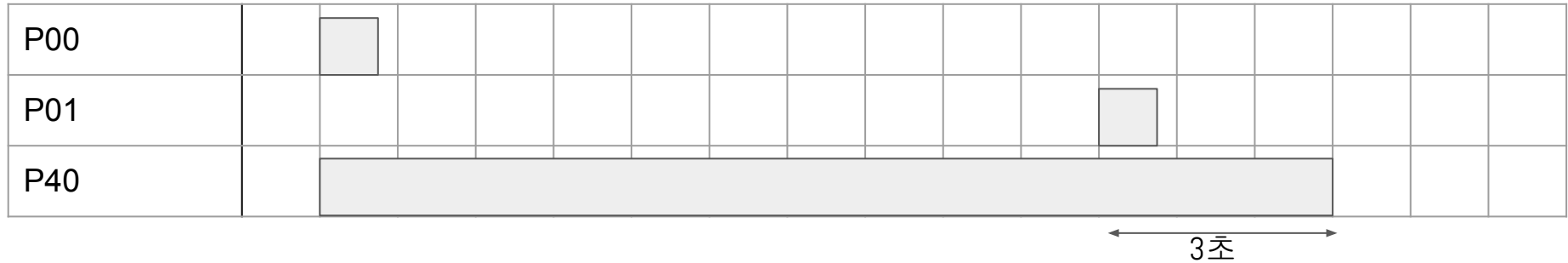


#PLC

# PLC - 타이머 TOF

# ex-11. 타이머 TOFF

T0F T1 30 // 100ms, 10ms, 1ms



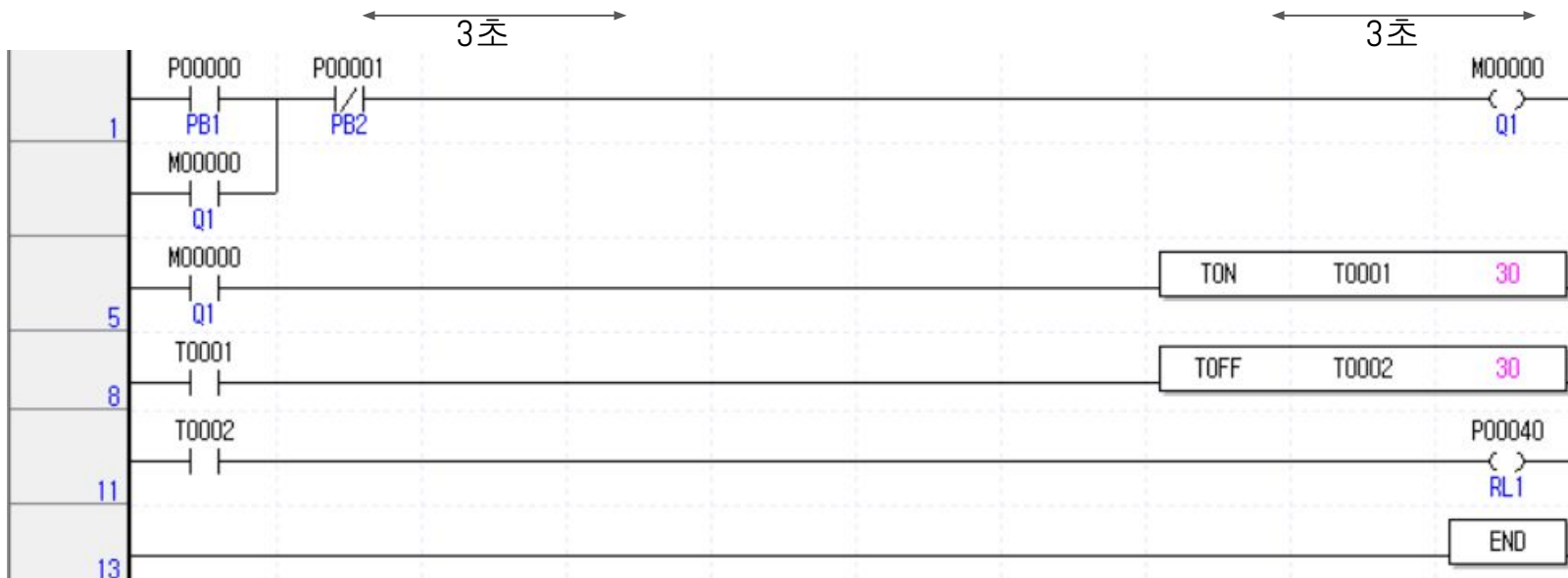
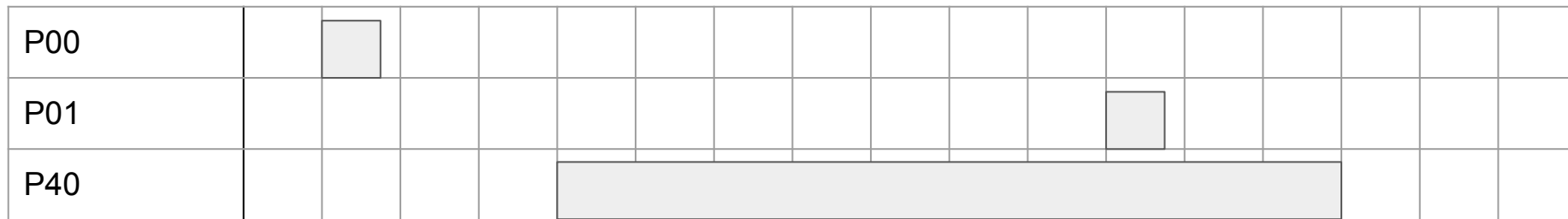




#PLC

# PLC - ON OFF delay

## ex-12. ON OFF Delay



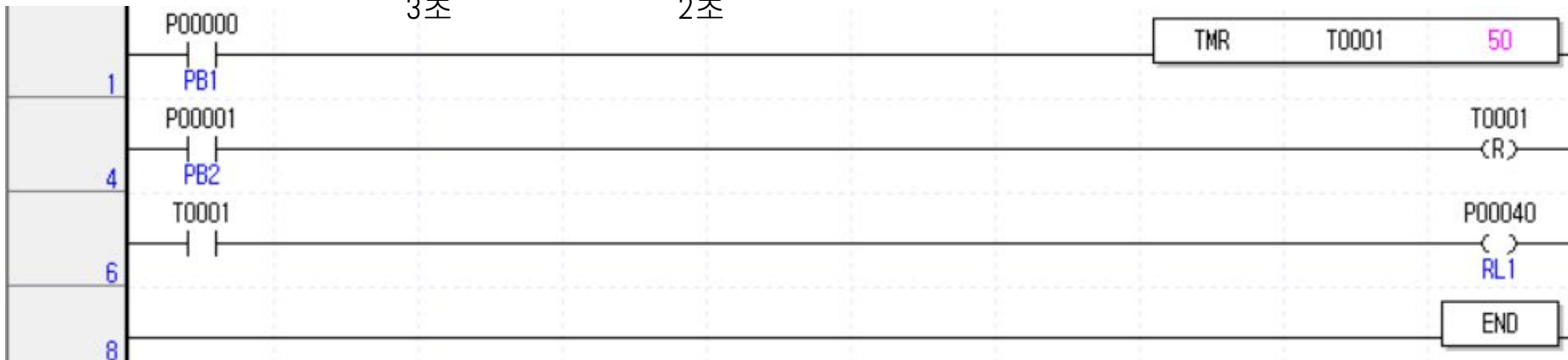
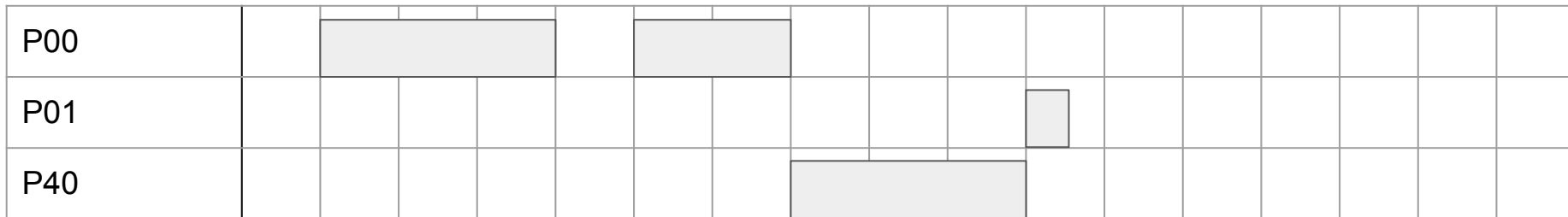


#PLC

# PLC - 적산타이머

# ex-13. TMR 적산타이머

입력조건이 0n 인 동안 현재값이 증가하여 누적된 값이 타이머 설정시간에 도달하면 접점 0n





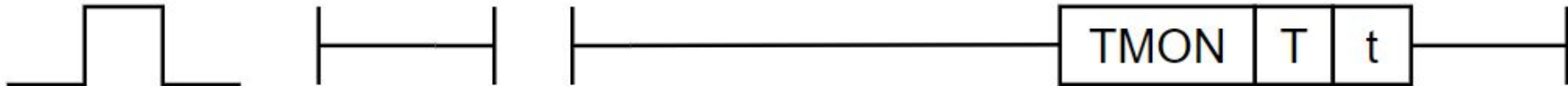
#PLC

# PLC – TMON

# ex-14. TMON (Monostable)

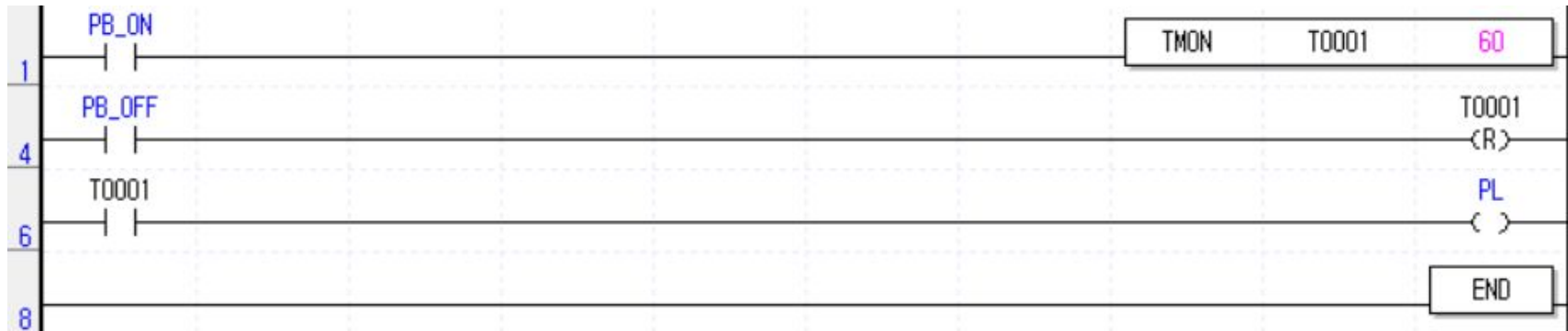
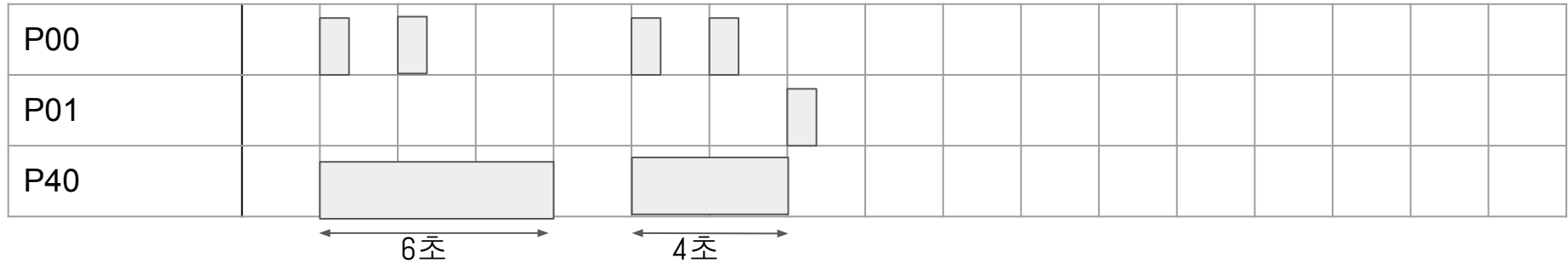
1. 입력조건이 On 되는 순간 타이머 출력은 On 되고, 타이머의 현재값이 설정값부터 감소하기 시작한 후 타이머 현재값이 0이 되면 타이머 출력이 OFF
2. 타이머 출력이 On 된 후 입력조건이 On, Off 변화가 있어도 감산은 계속 진행
3. 리셋 입력이 있으면 타이머 접점은 OFF 되고 현재값은 0 이 됨

입력조건접점





# ex-14. TMON (Monostable)





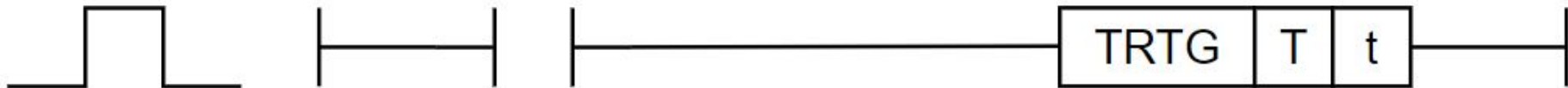
**#PLC**

# **PLC - TRTG**

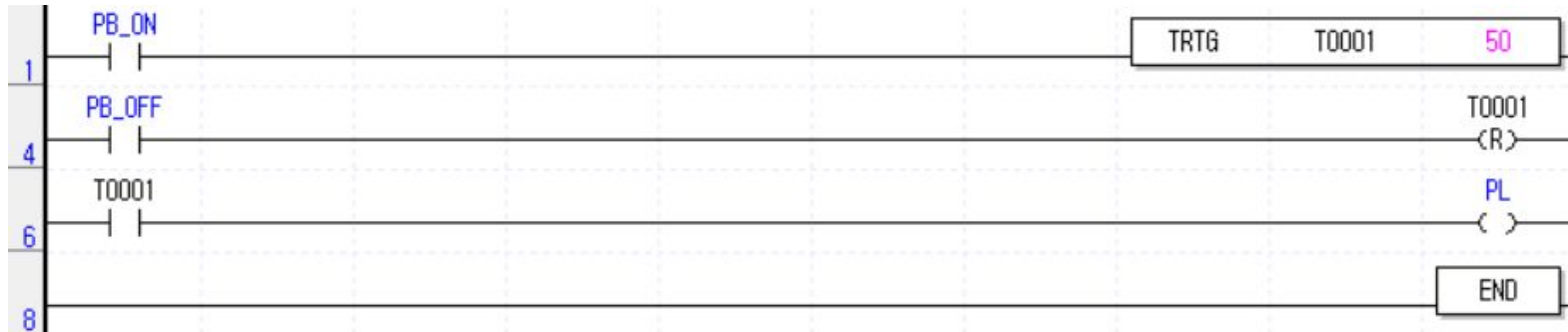
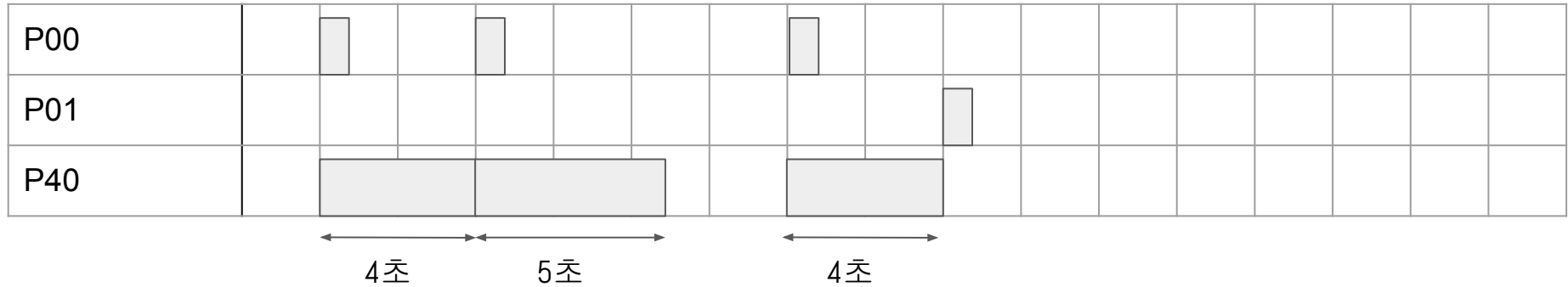
# ex-15. TRTG (Retriggerable)

1. 입력조건이 성립되면 타이머 출력이 ON 되고 타이머 현재값이 감소하여 0이 되면 출력이 OFF
2. 타이머 현재값이 0 되기 전 입력조건이 OFF 에서 ON 으로 변하면 타이머 현재값은 초기값으로 재설정
3. 리셋 입력이 있으면 타이머 접점은 OFF 되고 현재값은 0 이 됨

입력조건접점



# ex-15. TRTG (Retriggerable)





#PLC

# PLC - 타이머실습

# 타이머 실습 1 - TOF 이용한 소등

제목 : timer\_exam\_01\_tof

입력과 출력이 아래와 같이 연결되어 있다. 이때 다음 내용을 PLC로 프로그램하시오.

PB1 을 누르면 램프가 점등하고, PB2를 누르면 10초 후 램프가 꺼집니다.

입력			출력		
디바이스	변수	설명	디바이스	변수	설명
P00000	PB1	점등스위치	P00040	PL	램프
P00001	PB2	소등스위치			



# 타이머 실습 2 - 순차점등

제목 : timer\_exam\_02\_ton

입력과 출력이 아래와 같이 연결되어 있다. 이때 다음 내용을 PLC로 프로그램하시오.

PB1 을 누르면 즉시 램프1 점등, 3초후 램프2 점등, 6초후 램프3 점등

PB2 를 누르면 모두 소등

입력			출력		
디바이스	변수	설명	디바이스	변수	설명
P00000	PB1	점등스위치	P00040	PL1	램프1
P00001	PB2	소등스위치	P00041	PL2	램프2
			P00042	PL3	램프3

# 타이머 실습 3 - 교대 점멸 회로

제목 : timer\_exam\_03\_ton\_blink

입력과 출력이 아래와 같이 연결되어 있다. 이때 다음 내용을 PLC로 프로그램하시오.

PB1 을 누르면 PL1과 PL2가 1초 간격으로 교대로 점멸합니다.

PB2를 누르면 PL1, PL2 램프가 꺼집니다.

입력			출력		
디바이스	변수	설명	디바이스	변수	설명
P00000	PB1	점등스위치	P00040	PL1	램프1
P00001	PB2	소등스위치	P00041	PL2	램프2



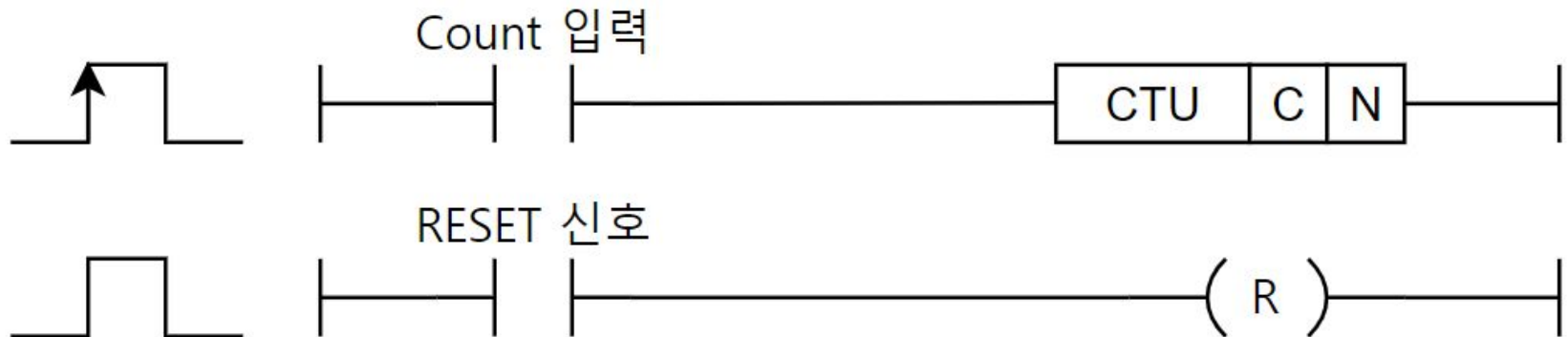
#PLC

# PLC - 카운터 CTU, CTR, CTUD

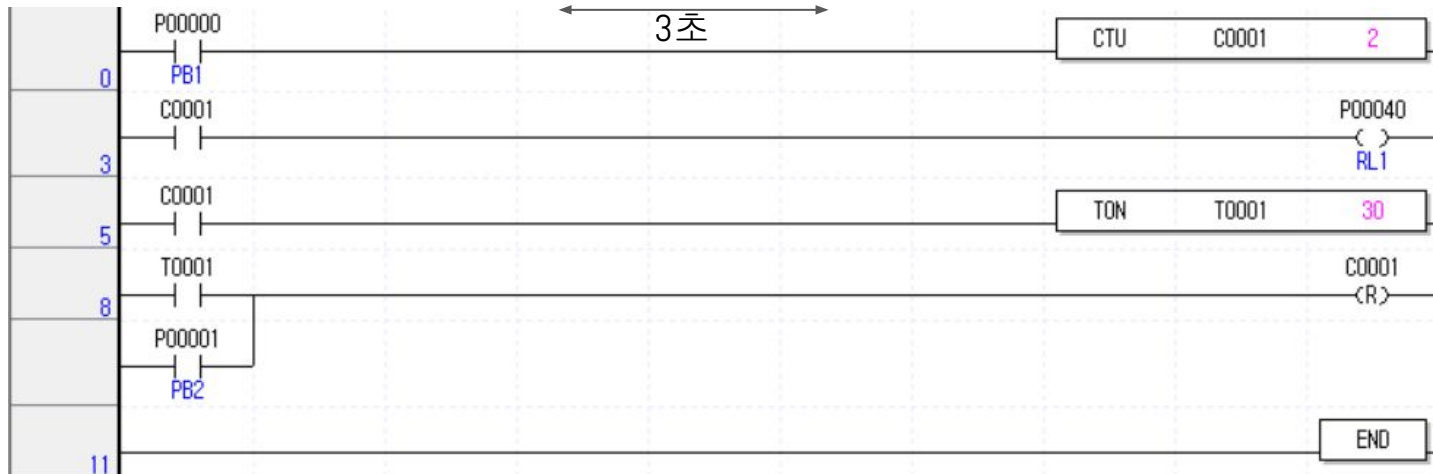
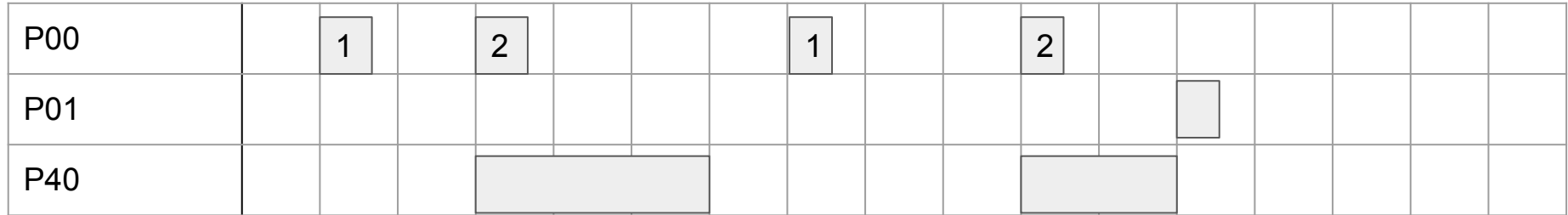
# 카운터 명령

# ex-16. 카운터 CTU

1. 입력 상승 펄스가 들어올 때마다 카운터의 값을 +1
2. 카운터값이 설정치 이상이면 출력을 ON
3. RESET 신호가 들어오면 출력 OFF, 카운터 값 0



# ex-16. 카운터 CTU

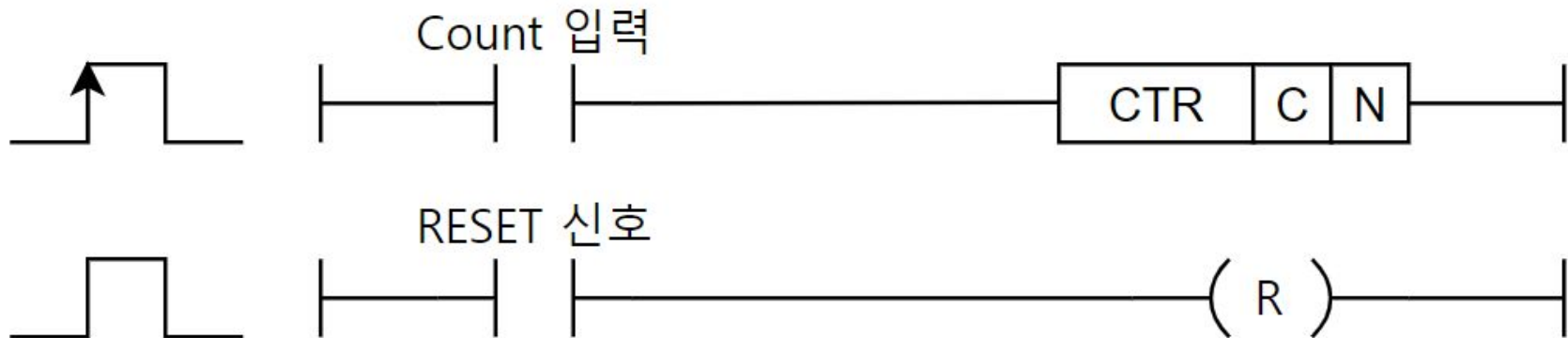






# ex-17. 카운터 CTR

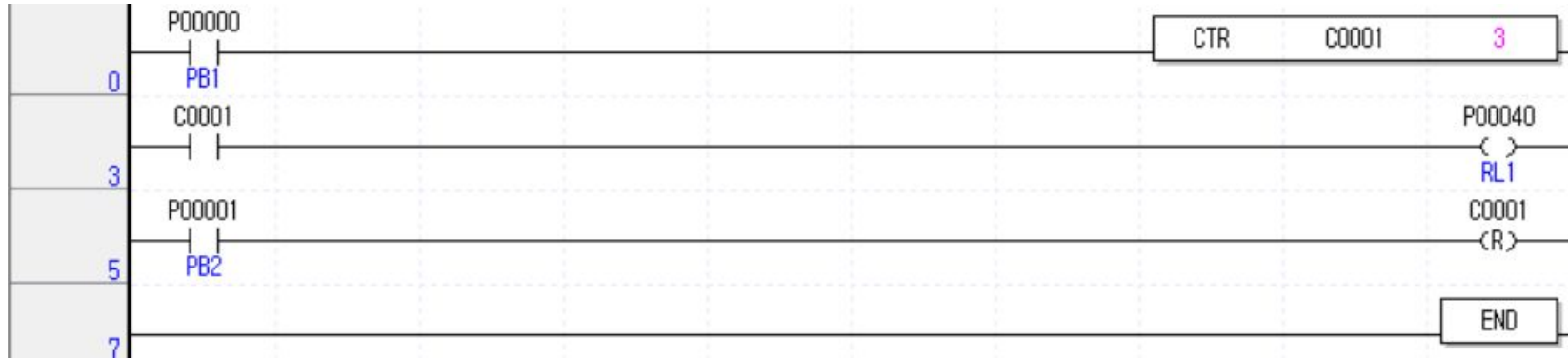
1. Ring Counter
2. CTU 와 유사하나 현재 카운터 값이 설정값에 도달한 후 그 다음 카운터 입력이 들어오면 0 이 됨
3. RESET 신호가 들어오면 출력 OFF, 카운터 값 0



# ex-17. 카운터 CTR

상승펄스 입력때마다 현재값에 +1 하며 설정값 도달 후 0부터 다시 시작

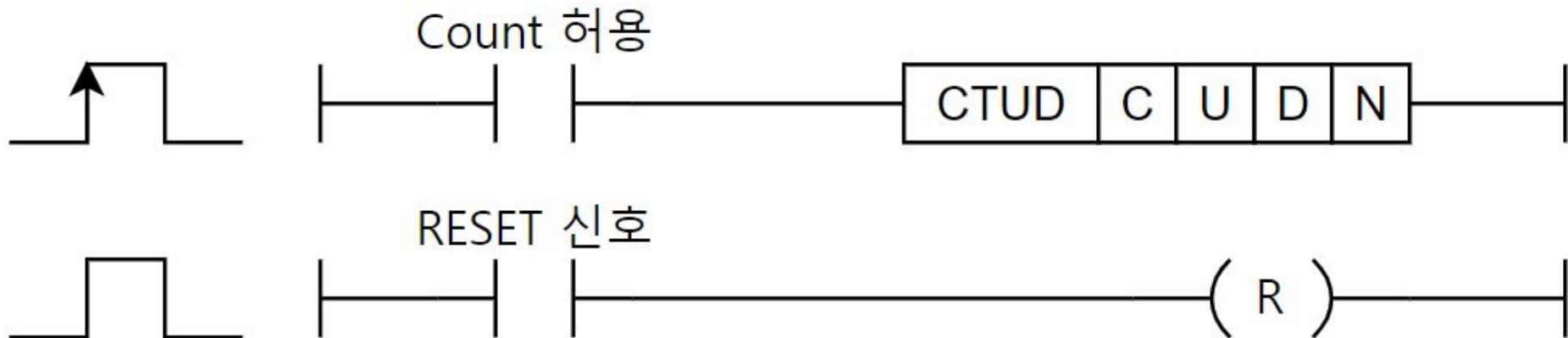
P00		1	2	3			0	1	2	3						
P01																
P40																





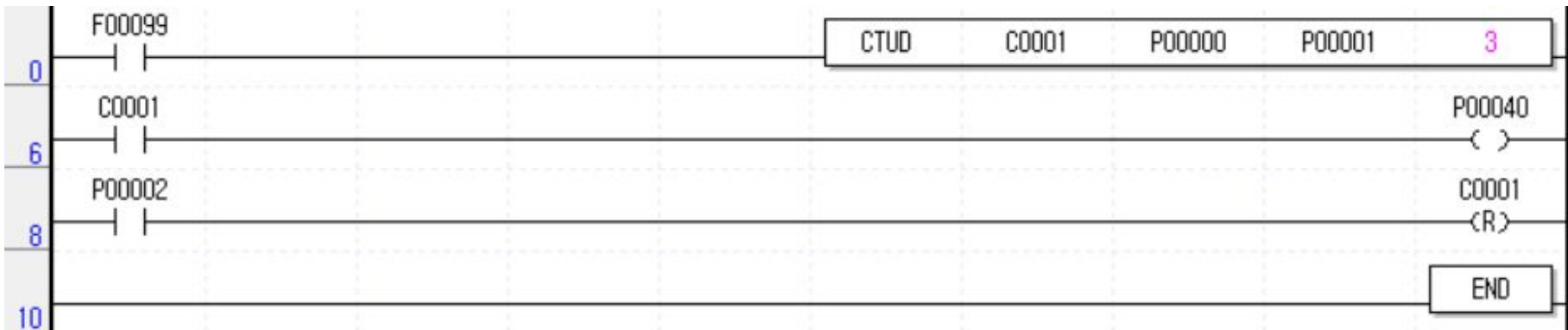
# ex-18. 카운터 CTUD

1. Up and Down Counter
2. U 에 상승신호가 들어오면 카운터+1, D에 상승신호가 들어오면 카운터-1
3. 현재값이 설정값 이상이면 출력 ON
4. RESET 신호가 들어오면 출력 OFF, 카운터 값 0



## ex-18. 카운터 CTUD

P00(UP)	1	2	3				3									
P01(DOWN)						2										
P02(RESET)										0						
P40(PL)																





#PLC

# PLC - 데이터 전송 명령

## MOV

# 데이터 전송 명령



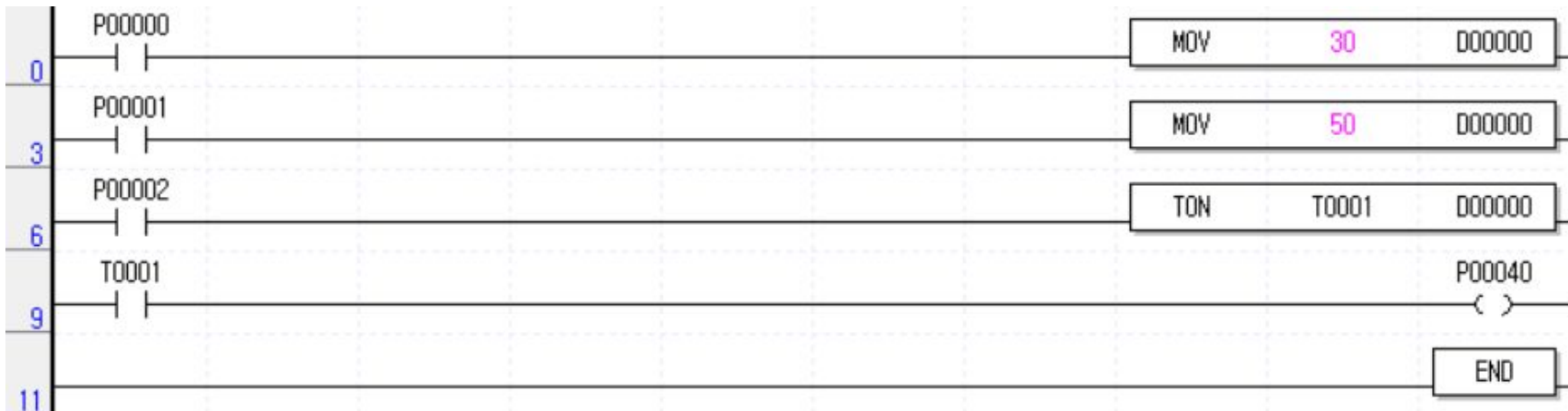
# ex-19. MOV, MOVP

1. S 로 지정된 16비트 워드 데이터를 D 로 전송
  - a. S는 데이터 혹은 데이터가 들어있는 디바이스 번호
  - b. D는 디바이스 번호



# ex-19. MOV, MOVN

1. 스위치1 누르면 타이머 설정값으로 3초 저장
2. 스위치2 누르면 타이머 설정값으로 5초 저장
3. 스위치3 누르면 타이머1 TON 동작



# ex-19. MOV, MOVP

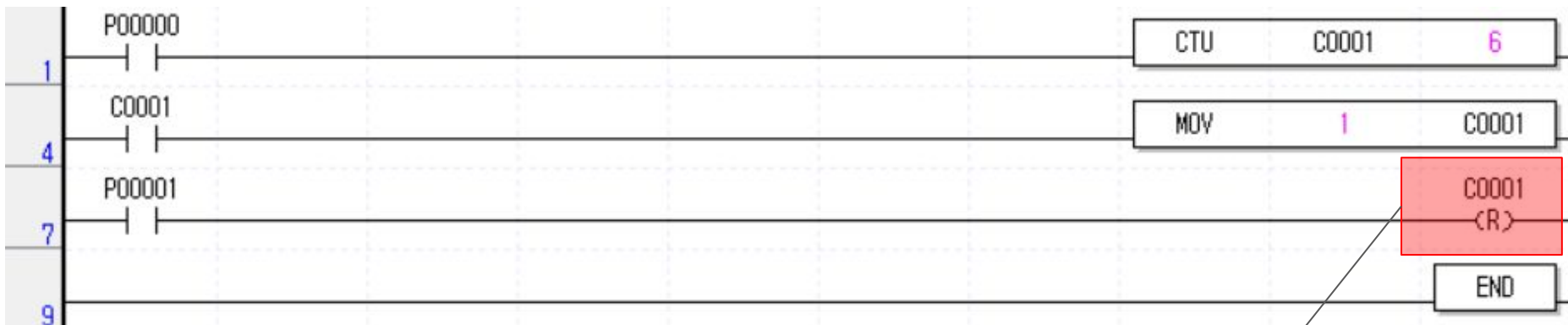
MOV 연습

1. 업카운터(CTU)의 현재값을 6이 되면 1로 바꾸어 1~5를 반복하는 회로 구성

# ex-19. MOV, MOVP

MOV 연습

1. 업카운터(CTU)의 현재값을 6이 되면 1로 바꾸어 1~5를 반복하는 회로 구성



카운터 리셋

# 연습문제

1. P0 에 신호가 들어오면,
2. 업카운터(CTU)의 현재값을 6이 되면 1로 바꾸어 1~5를 반복하는 회로 구성
3. 카운터가 5이면 P40 의 램프를 켜다. 5가 아니면 램프를 끈다.

- 타이머만을 이용하거나
- 타이머 + 카운터를 이용하여 프로그램



#PLC

**PLC** - 비교 명령 =, <, ...

# 비교 명령



# ex-20. 비교, < <= =

1. S1, S2 를 비교하여 조건과 일치하면 연산 결과를 True (ON, 1), 아니면 False (OFF, 0)
2. S1, S2 비교는 Signed 연산
  - a. S1 은 S2와 비교하게 되는 데이터나 데이터의 주소, INT/DINT
  - b. S1, S2 는 -32,768 ~ 32,767 사이의 수



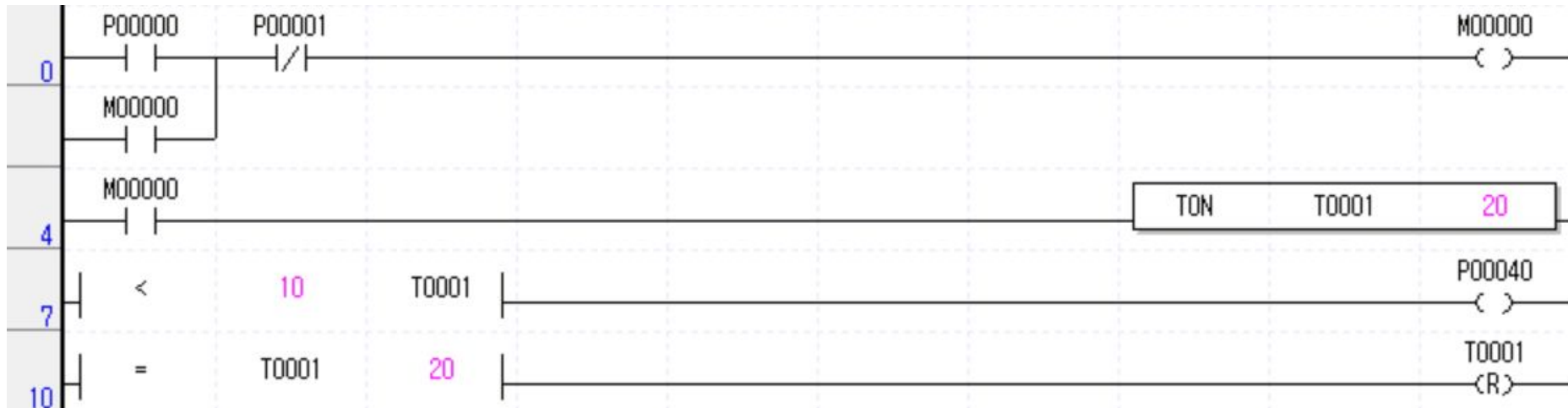
# ex-20. 비교, < <= =

비교	내용	True 면 On, 1
=	$S1 = S2$ , S1과 S2는 같다	On
<=	$S1 \leq S2$ , S1은 S2보다 작거나 같다	On
<	$S1 < S2$ , S1은 S2보다 작다	On
<>	$S1 \neq S2$ , S1과 S2는 같지 않다	On
>=	$S1 \geq S2$ , S1은 S2보다 크거나 같다	On
>	$S1 > S2$ , S1은 S2보다 크다	On

# ex-20. 비교, < <= =

1. 스위치1 누르면 1초 간격으로 램프가 점멸(On - Off 반복), 스위치2 누르면 램프 소등

P00(스위치 1)																	
P01(스위치 2)																	
P40(램프)																	





# ex-21. 3항비교

1. S1, S2, S3 를 각각 2개씩 비교하여 조건과 일치하면 연산 결과를 True (ON, 1), 아니면 False (OFF, 0)
2. S1, S2, S3 비교는 Signed 연산
  - a. S1, S2, S3 는 비교하게 되는 데이터나 데이터의 주소, INT
  - b. S1, S2 는 -32,768 ~ 32,767 사이의 수



# ex-21. 3항비교

비교	내용	설명
$=3$	$S1 = S2 = S3$	S1, S2, S3는 모두 같다
$\leq 3$	$S1 \leq S2 \leq S3$	S1은 S2보다 작거나 같고, S2은 S3보다 작거나 같다
$<3$	$S1 < S2 < S3$	S1은 S2보다 작고, S2은 S3보다 작다
$\langle \rangle 3$	$S1 \neq S2 \neq S3$	S1, S2, S3는 모두 같지 않다
$\geq 3$	$S1 \geq S2 \geq S3$	S1은 S2보다 크거나 같고, S2은 S3보다 크거나 같다
$>3$	$S1 > S2 > S3$	S1은 S2보다 크고, S2은 S3보다 크다

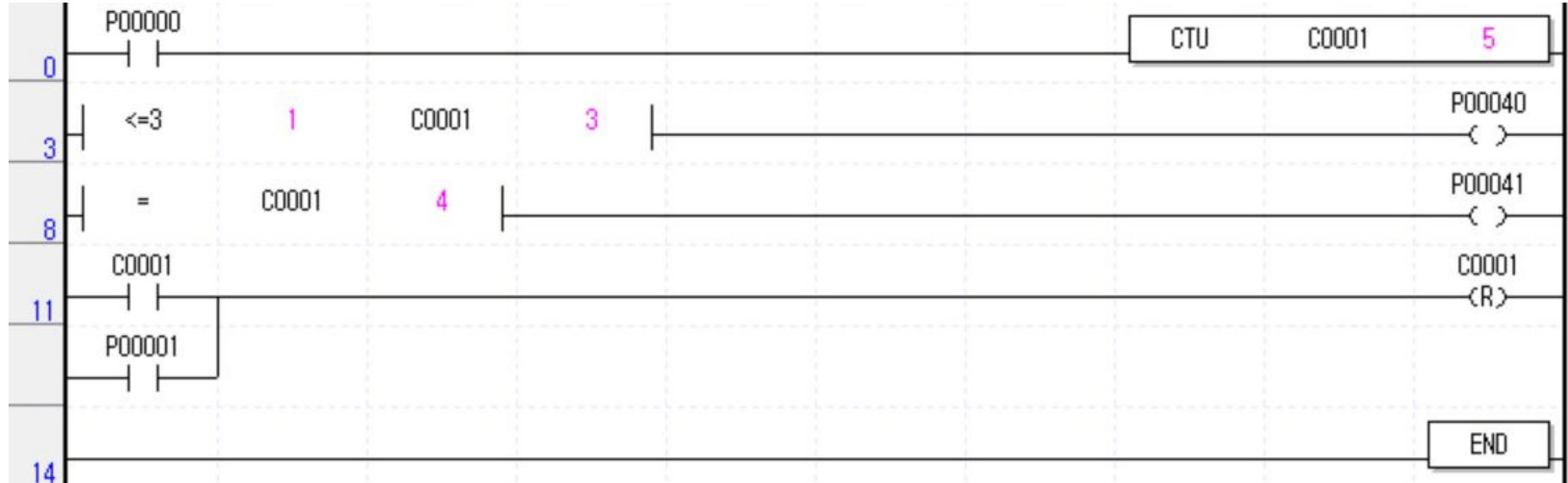
# ex-21. 3항비교

1. 카운터값이 1 이상 3 이하일 때 램프1 출력을 On, 4일때 램프2 출력 On, , 0일때 램프1, 램프2 Off
2. 카운터는 0에서 4까지 반복

P00(PB_UP)		1	2	3	4	0	1	2	3								
P01(PB_RES)																	
P40(램프1)																	
P41(램프2)																	

# ex-21. 3항비교

1. 카운터값이 1 이상 3 이하일 때 램프1 출력을 On, 4일때 램프2 출력 On, , 0일때 램프1, 램프2 Off
2. 카운터는 0에서 4까지 반복







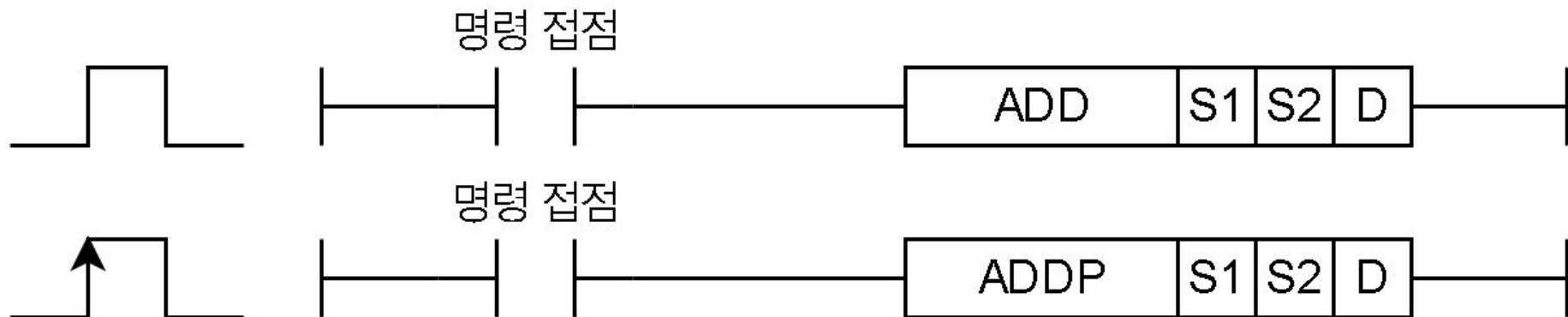
#PLC

# PLC – BIN 사칙 연산

# **BIN 사칙**

# ex-22. ADD, ADDP

1. 워드데이터 S1, S2 를 더한 후 결과를 D에 저장
2. Signed 연산, 결과가 32,767을 초과하거나 -32768 미만일 때 캐리 플래그 Set 되지 않음 - 주의 !!!
  - a. S1 과 S2 는 연산을 실행할 데이터
  - b. D는 연산결과를 저장할 주소



# ex-22. BIN 사칙

명령	내용	사용법	설명
ADD	덧셈	ADD D0 1 D0	$D0 = D0 + 1$ , D0의 값을 1 증가
SUB	뺄셈	SUB D0 1 D0	$D0 = D0 - 1$ , D0의 값을 1 감소
MUL	곱셈	MUL D0 2 D0	$D0 = D0 * 2$ , D0의 값을 2배 증가
DIV	나눗셈	DIV D0 2 D0	$D0 = D0 / 2$ , D0의 값을 2로 나눔

# ex-22. BIN 사칙

## 접두어

- 기본 명령어 앞에 위치, 명령어 기능을 보조
- 2개까지 접두어 사용

- ❑ D : 더블 워드, 32비트 정수 [ DMOV ]
- ❑ R : 32비트 실수[ RMOV ]
- ❑ L : 64비트 실수[ LMOV ]
- ❑ G : 그룹데이터 연산[ GMOV ]
- ❑ B : 비트연산[ BMOV ]
- ❑ \$ : 문자열 데이터 연산[ \$MOV ]
- ❑ 8 : 8비트 연산[ 8= ]
- ❑ 4 : 4비트 연산[ 4= ]

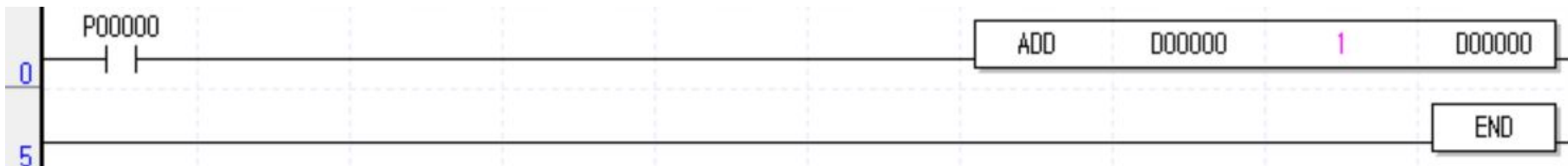
## 접미어

- 기본 명령어 뒤에 위치, 명령어 기능을 보조
- 기본 명령어 뒤에 1개 혹은 2개 붙을 수 있음

- ❑ U : 부호없는 10진 정수 [ ADDU ]
- ❑ P : 상승에지 연산[ MOVP ]
- ❑ B : BCD데이터 연산[ ADDB ]
- ❑ 8 : 8비트 데이터 연산[ MOV8 ]
- ❑ 4 : 4비트 데이터 연산[ MOV4 ]
- ❑ 3 : 3개의 데이터 처리 [ =3 ]

# ex-22. ADD, ADDP

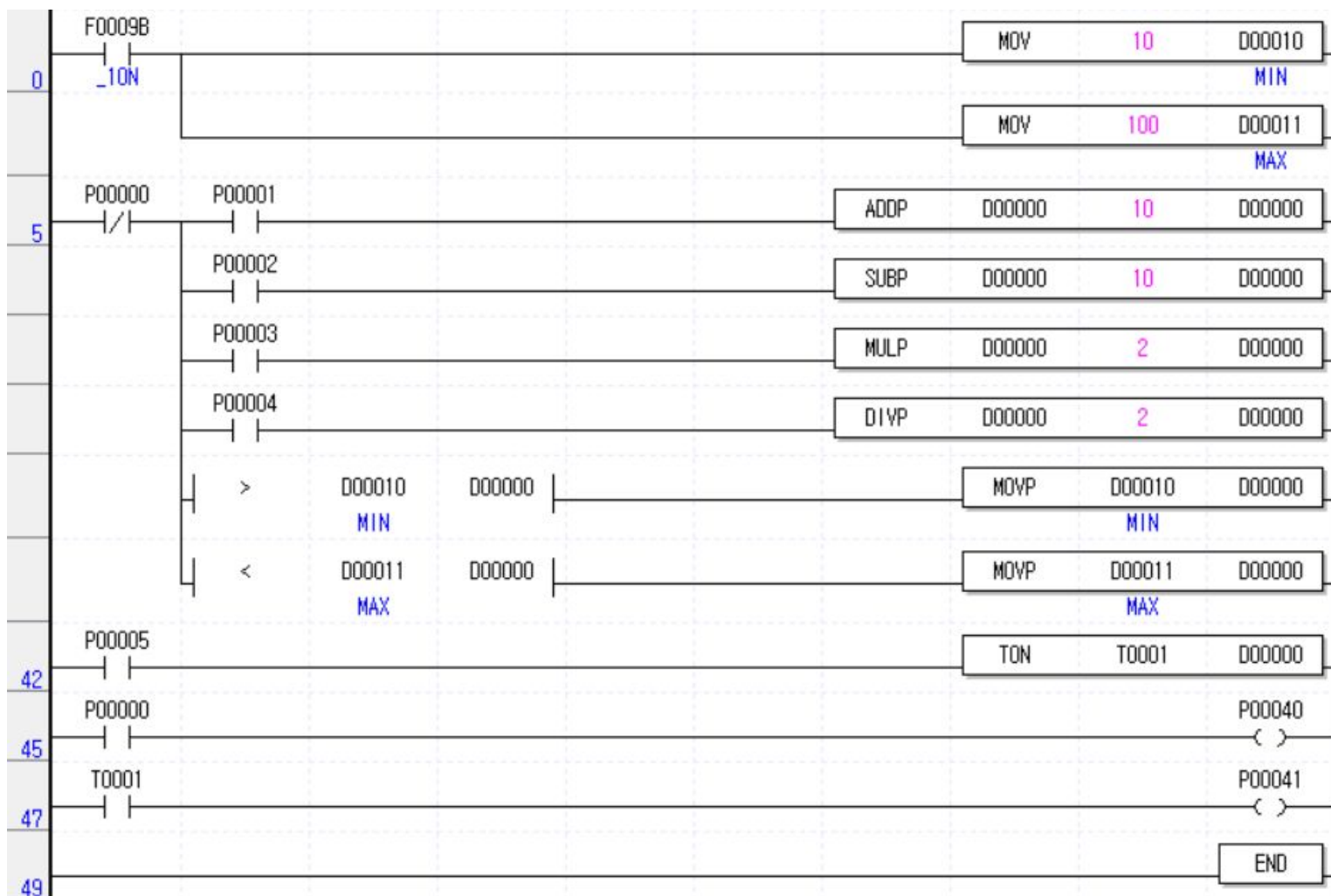
1. P0 을 누르면 D0 에 1 씩 증가



# ex-23. 사칙연산

1. P0 ON 상태에서
  - a. P40 ON
  - b. P1 입력시 D0 에 10 증가
  - c. P2 입력시 D0 에 10 감소
  - d. P3 입력시 D0 를 2배 증가
  - e. P4 입력시 D0 를 2로 나눔
2. D0 는 최소 10, 최대 100 사이의 값을 가질 수 있음
3. P5 ON 시 D0 값으로 타이머1을 설정
4. 타이머1 ON 시 P41 ON







#PLC

# PLC - 사칙 연산 연습 문제

# 연습문제 pr\_01

1. P0 스위치를 누르면,
  - a. D0 가 100이 된다.
2. P1 스위치를 누르면
  - a. D0 가 10씩 증가한다.
3. D0 가 20 의 배수이면 P40 램프에 불이 켜지고, 30 의 배수이면 P41 램프에 불이 켜진다.

위 문제를 풀고

캡춰 화면과 XG5000 프로젝트 파일을 압축파일로 묶어서 과제로 제출합니다.

# 연습문제 pr\_02

1. P0 스위치를 누르면 D0 가 0 이 된다.
2. P1 스위치를 누르면 변수 “step”(D1) 이 10 이 된다.
3. P2 스위치를 누르면 변수 “step”(D1) 이 20 이 된다.
4. P3 스위치를 누르면 D0 가 변수 step 에 들어간 값만큼 증가한다.
5. P4 스위치를 누르면 D0 가 초기화된다. ( $D0 = 0$ )
6. D0 가 100 이상 150 이하면 PL1 에 불이 들어온다. 아니면 PL1 의 불이 꺼진다.
7. D0 가 150 초과 200 이하면 PL2 에 불이 들어온다. 아니면 PL2 의 불이 꺼진다.

위 문제를 풀고

캡춰 화면과 XG5000 프로젝트 파일을 압축파일로 묶어서 과제로 제출합니다.

# 연습문제 pr\_03

1. D0 에 32765 를 초기값으로 넣는다.
2. P0 스위치를 누를때마다 D0 가 1 씩 증가한다.
3. P0 를 3번 누른 다음 D0 을 값이 얼마가 되는지 래더코드를 짜서 확인하시오.

참고, 1번만 실행시키고 싶을 때는 플래그변수 **F9B** 를 사용합니다.



위 문제를 풀고

캡춰 화면과 XG5000 프로젝트 파일을 압축파일로 묶어서 과제로 제출합니다.



# 과제 (2025-04-05)

다음 내용으로 PLC 프로그래밍 하세요.

- X00 스위치를 누르면 Y20 에 연결된 램프에 불이 들어온 후 계속 켜져 있음
  - X01 스위치를 누르면 Y20 에 연결된 램프에 불이 꺼진 후 계속 꺼져 있음
  - Y21 에 연결된 램프는 0.5초 ON, 0.5초 OFF 를 시작부터 끝까지 계속 반복함
  - Y22 에 연결된 램프는 X00 스위치를 누르면 0.5초 ON, 0.5초 OFF를 X01 스위치가 눌러지기까지 반복함
- 
- 프로그램 완성된 GX Works2 화면을 캡춰하고 프로그램 소스코드와 함께 업로드
  - 첫줄에 자신의 이름, 학번을 쓰세요.
  - 둘째줄에 제목으로 "2025.04.05.과제 - 릴레이와 코일 입출력 실습" 이라고 제목을 쓰세요.



# 주석

; (세미콜론)

[보기] - 코멘트 표시, 스테이트먼트 표시 Check[v] 할 것

화면 왼쪽에 마우스 클릭 + [ENTER] 입력 후 “;” + 주석 입력

