

# **Chapter 5**

# **HMI 실습**

## 5.1 GOT와 GT Desigher3

### 1. 용어 정리

- HMI(Human Machine Interface)
- 시각이나 청각과 관련된 인간의 아날로그적인 인지의 세계와 컴퓨터나 통신의 디지털의 처리하는 기계의 세계를 연결하는 인터페이스.
- PLC자동화시스템에서는 주로 터치스크린을 이용하여 사용자 인터페이스를 직관적으로 쉽게 판단하고 조작되도록 하는 것을 말함.
  
- 터치 패널
- 전기 수동 아날로그 조작식 제어반을 컴팩트한 터치스크린으로 대체.
- 인터페이스가 간단하고 조작이 쉬우며, 그래픽이 직관적으로 알기 쉬게 구현.



그림 5.1 컴팩트한 터치패널

- GOT(Graphic Operation Terminal)
  - 미쯔비시 HMI의 명칭.
  - 기존에는 조작반에서만 가능하던 스위치 조작, 램프 표시, 데이터 표시, 메시지 표시 등을 모니터 화면에서 실행할 수 있는 터치패널식 표시기를 의미.
- 작화
  - 터치스크린을 디자인하는 소프트웨어를 이용하여 HMI 화면을 설계하고 PLC와 연동하도록 작업하는 것.

## 2. 미쓰비시사의 HMI 개요

- GOT 시리즈의 종류
  - 보급형인 GOT1000과 고급형인 GOT2000시리즈

표 5.1 주요 GOT 시리즈의 종류

시리즈	종류	특징
GOT1000	GT16	<ul style="list-style-type: none"> <li>• 사용자 메모리 용량 15MB(GT16□□~VNB□는 11MB)</li> <li>• USB 호스트&amp;USB 디바이스 탑재</li> <li>• Ethernet · RS-422/485 · RS-232 인터페이스 표준 탑재</li> <li>• 멀티미디어 모듈 · 비디오/RGB 모듈에 대응 *</li> <li>• 아날로그 터치 패널 채택</li> </ul>
	GT15	<ul style="list-style-type: none"> <li>• 사용자 메모리 용량 9MB(GT15□□~VNB□는 5MB)</li> <li>• USB 디바이스 탑재</li> <li>• RS-232 인터페이스 표준 탑재</li> <li>• 비디오/RGB 모듈에 대응 *</li> </ul>
	GT14	<ul style="list-style-type: none"> <li>• 사용자 메모리 용량 9MB</li> <li>• USB 호스트&amp;USB 디바이스 탑재</li> <li>• Ethernet · RS-422/485 · RS-232 인터페이스 표준 탑재 *</li> <li>• SD 메모리 카드 인터페이스 표준 탑재(SDHC, SD)</li> </ul>
GOT2000	GT27	<ul style="list-style-type: none"> <li>• 멀티 터치 · 제스처 기능 탑재 모델</li> <li>• 21인치, 800X600 지원</li> <li>• 사용자 메모리 용량</li> <li>• 저장용 메모리(ROM) : 57MB*1</li> <li>• 동작용 메모리(RAM) : 128MB*2</li> <li>• USB 호스트&amp;USB 디바이스 탑재</li> </ul>
	GT25	<ul style="list-style-type: none"> <li>• 고성능과 저렴한 가격을 모두 갖춘 미들 레인지 모델</li> <li>• 사용자 메모리 용량 21인치, 800X600 지원</li> <li>• 저장용 메모리(ROM) : 32MB</li> <li>• 동작용 메모리(RAM) : 80MB</li> <li>• USB 호스트&amp;USB 디바이스 탑재</li> </ul>

- I/O 인터페이스
- GOT에서 표시되는 화면 데이터는 PC에서 전용 작화 소프트웨어(GT Designer3)에서 작성하여 GOT로 전송.
- GT Designer3을 사용하면 PC에서 작성한 화면을 GOT에서 재현 가능.
- GOT2000시리즈의 I/O인터페이스에서는 이더넷을 지원하므로 다수의 장치를 연결할 때 매우 효율적으로 사용.

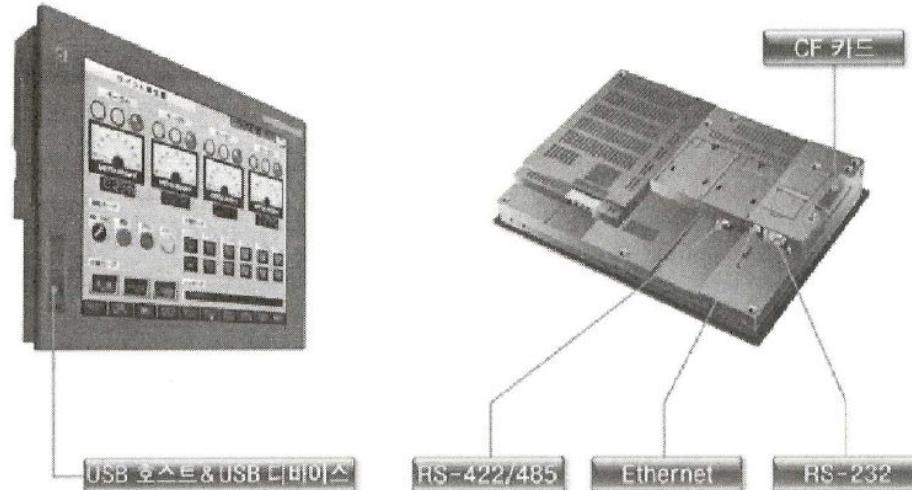


그림 5.2 GOT2000의 I/O 포트  
구성

- 이더넷 기반 시스템 구성
- PC, PLC 및 GOT2000이 이더넷을 통해 접속되어 있음.
- PC에서 작화도구(GT Designer3)를 이용하여 화면 작화를 하고 이더넷으로 GOT에 전송.
- 스크린에 있는 장치를 터치하여 PLC를 작동시킴.

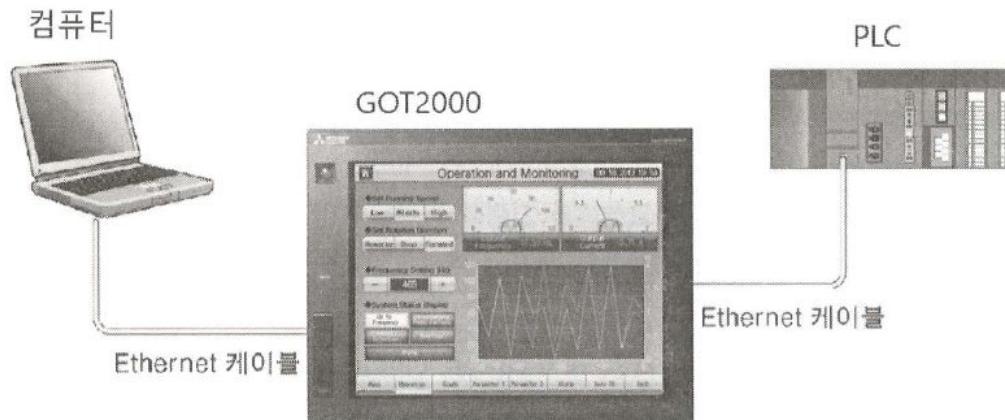


그림 5.3 GOT2000의 I/O 포트  
구성

### 3. HMI 동작 예

- PLC와 연동
  - HMI 작화시 디바이스 그림(스위치, 램프 등)에 PLC 디바이스 번호를 부여하여 PLC와 연동.
  - PLC의 래더 프로그램 작성시 HMI에 부여된 디바이스 번호의 장치로 래더 프로그램이 구동되도록 작성.

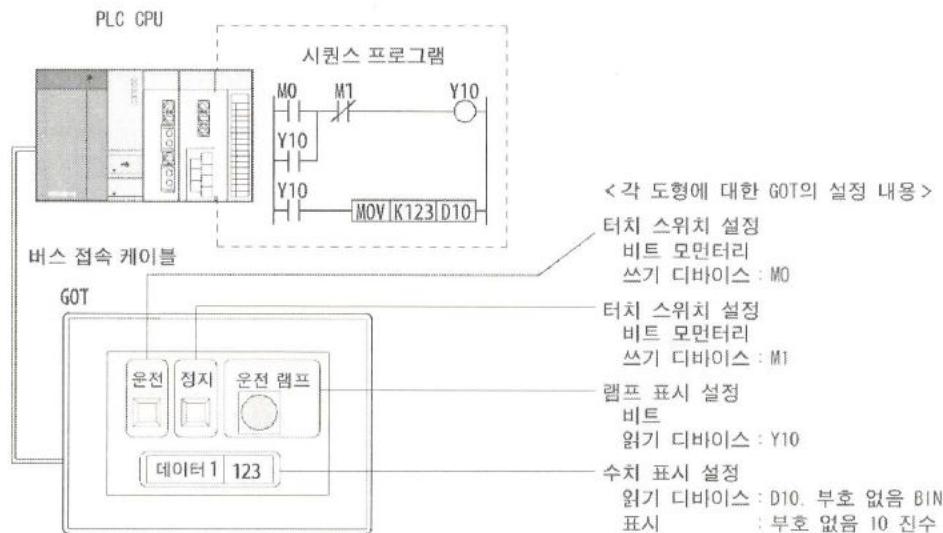


그림 5.4 동작 구성의 예

- 운전 스위치 터치시 동작
  - HMI의 운전스위치(M0)를 터치하면 PLC래더에서 M0가 ON된다.

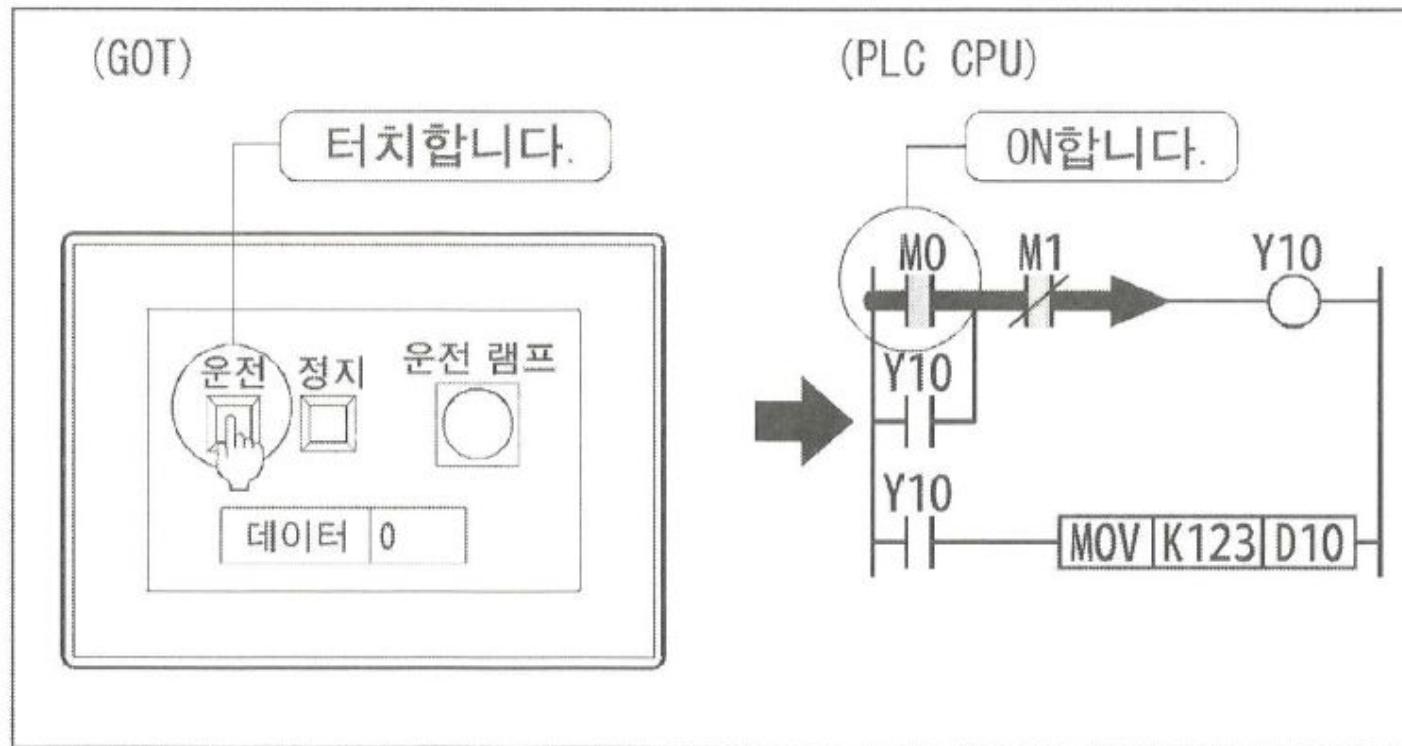


그림 5.5 운전 스위치 터치시의 동작 흐름

- 램프 점등 표시 동작
- 래더에서 M0가 ON되면 출력 디바이스 Y10이 ON된다.
- HMI의 램프가 PLC Y10 디바이스로 설정하였으므로 운전 램프가 점등된다.

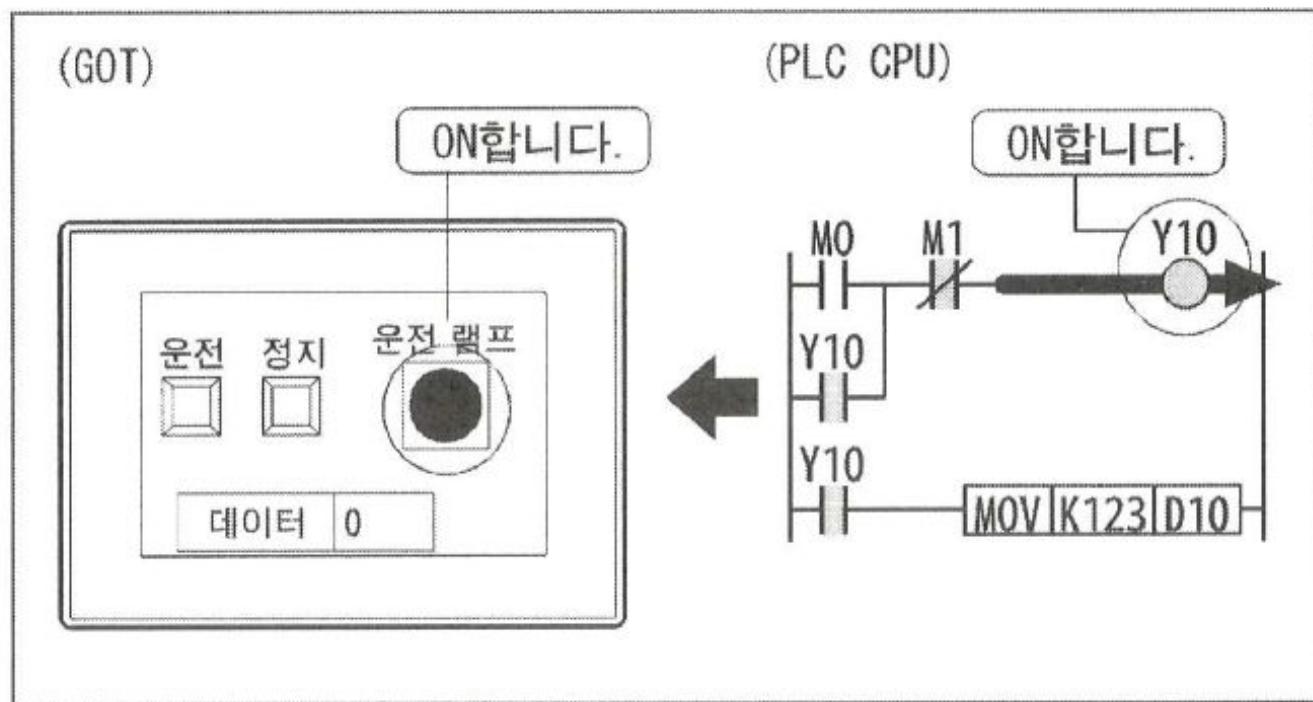


그림 5.6 램프 동작의 흐름

- 데이터 표시 동작
- 래더HMI 숫자 표시기의 연결 디바이스는 D10이고, PLC 래더에서 Y10이 ON되어 숫자 123이 PLC 데이터메모리 D10에 저장되므로 D10의 값이 HMI 숫자 표시기에 나타난다.

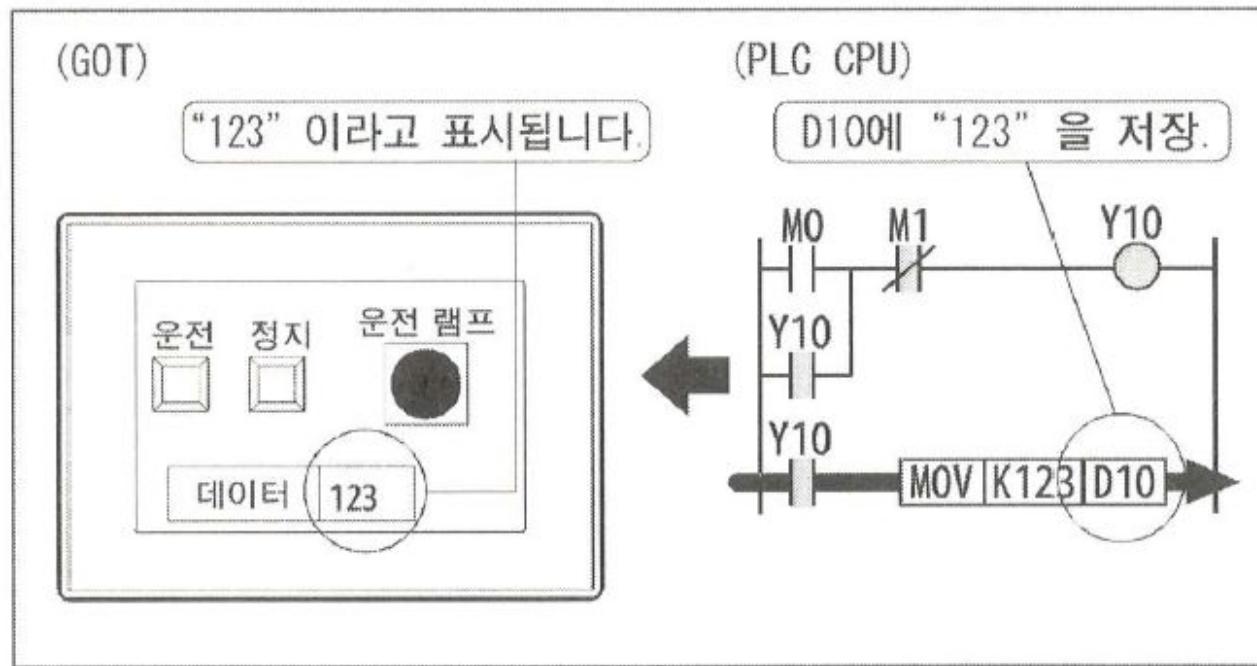


그림 5.7 HMI의 데이터 표시 동작의 흐름

- 정지 스위치 클릭시 동작
- HMI에서 정지 스위치(M1)을 터치하면 PLC내부 장치 M1이 작동한다.
- 래더상에서 M1이 b접점으로 설정되어 있으므로 접속이 끊어지되 되므로 출력 Y10은 HMI의 램프에 할당되어 있으므로 램프가 소등하게 된다.

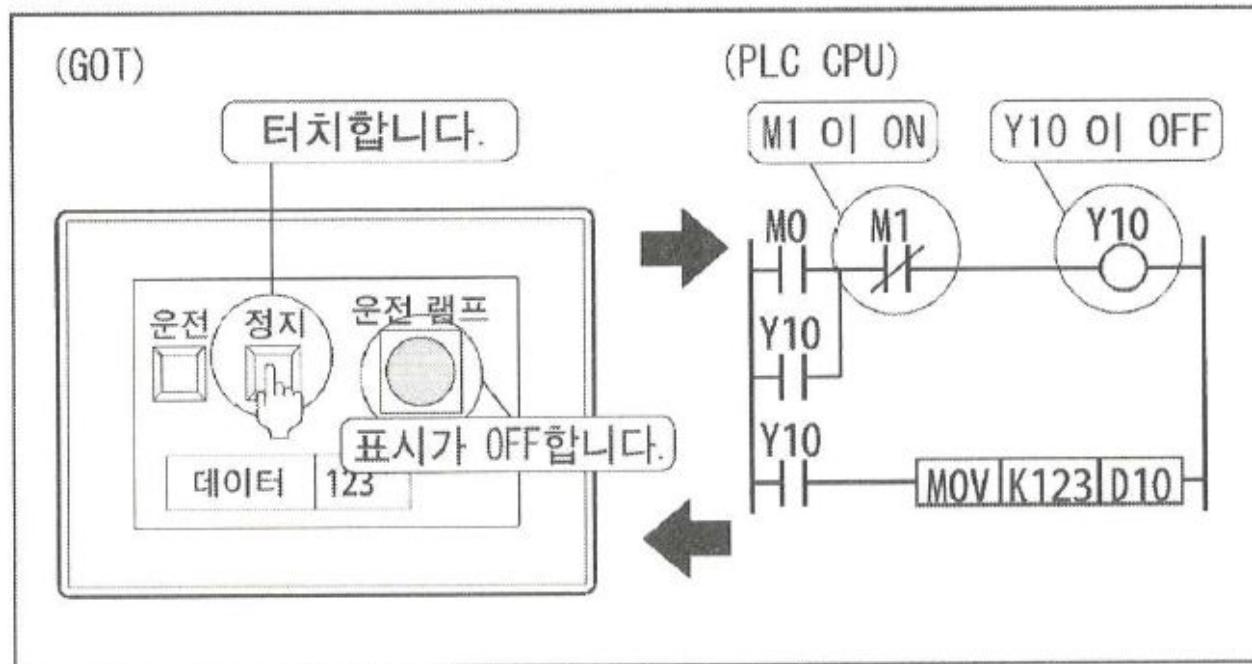


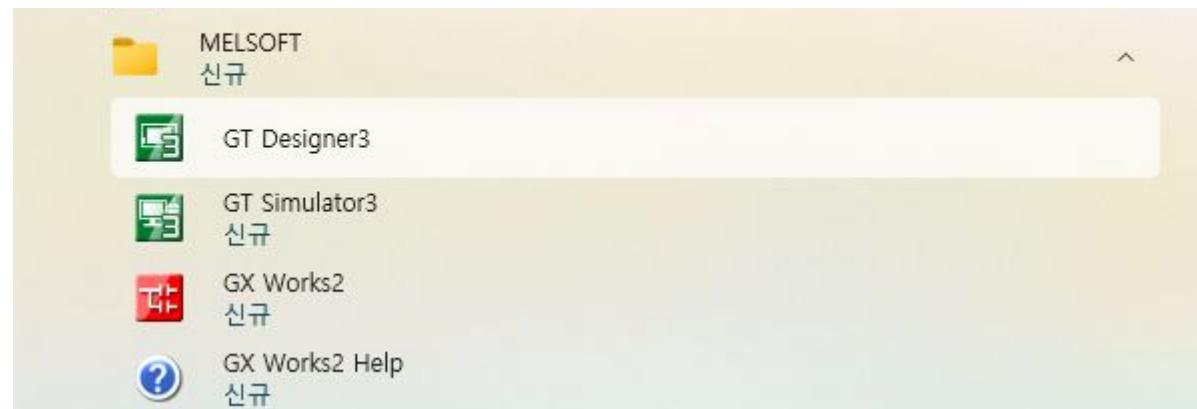
그림 5.8 HMI의 정지 스위치 터치시의 동작 흐름

## 5.2 GT Designer3 시작하기

### 1. GT Designer3 실행과 프로젝트 생성

- 주의 사항
  - GT Designer3 프로젝트는 복수의 파일로 구성.
  - GT Designer3 프로젝트의 파일/폴더 조작(이동, 이를 변경, 복사, 삭제)은 GT Designer3에서 실행해야 함.
  - 탐색 등의 PC툴에서 GX Designer3 프로젝트 파일/폴더를 조작하면, GT Designer3의 프로젝트가 열리지 않게 됨.

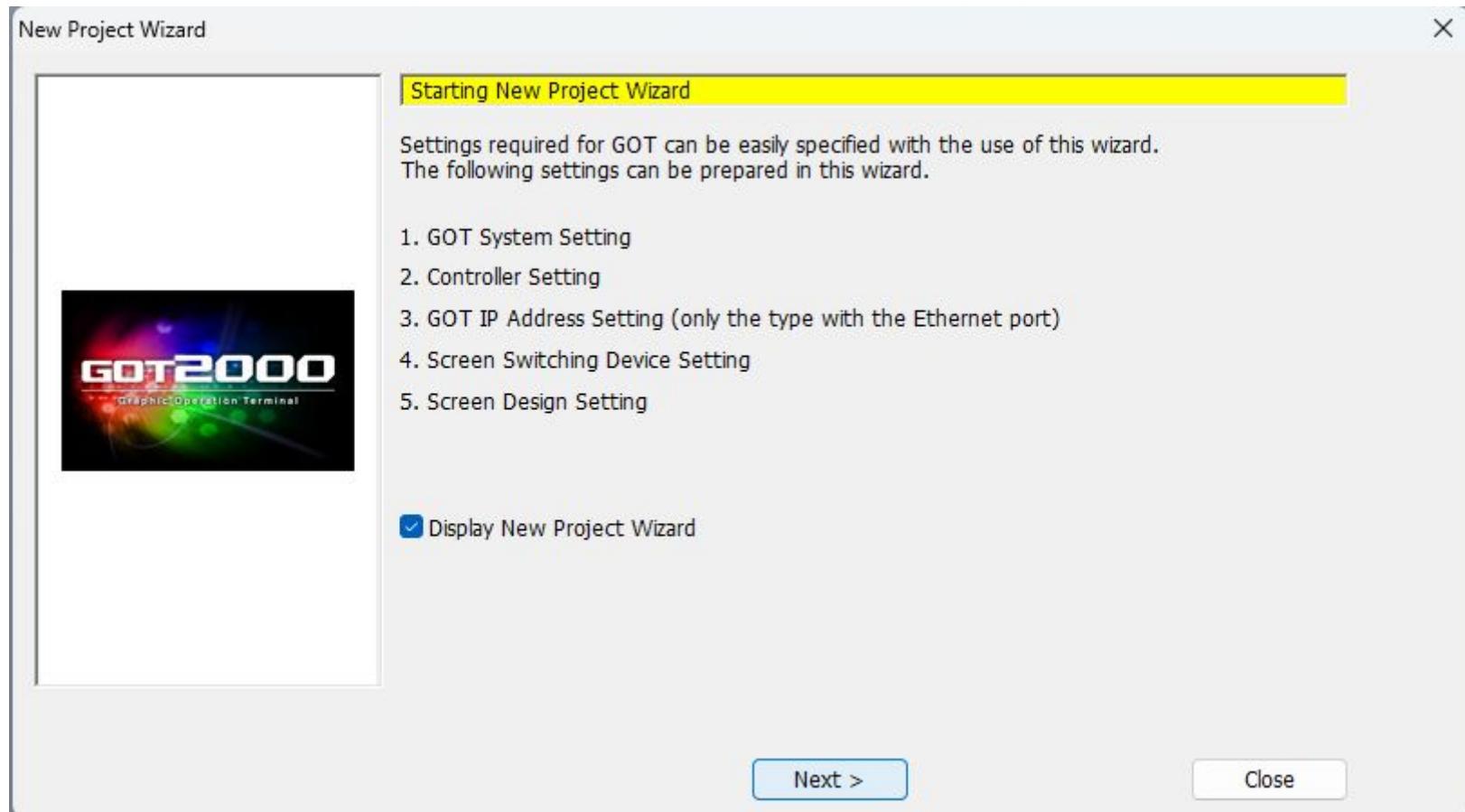
#### (1) [GT Designer3] 을 클릭하여 실행



(2) [New] 를 클릭

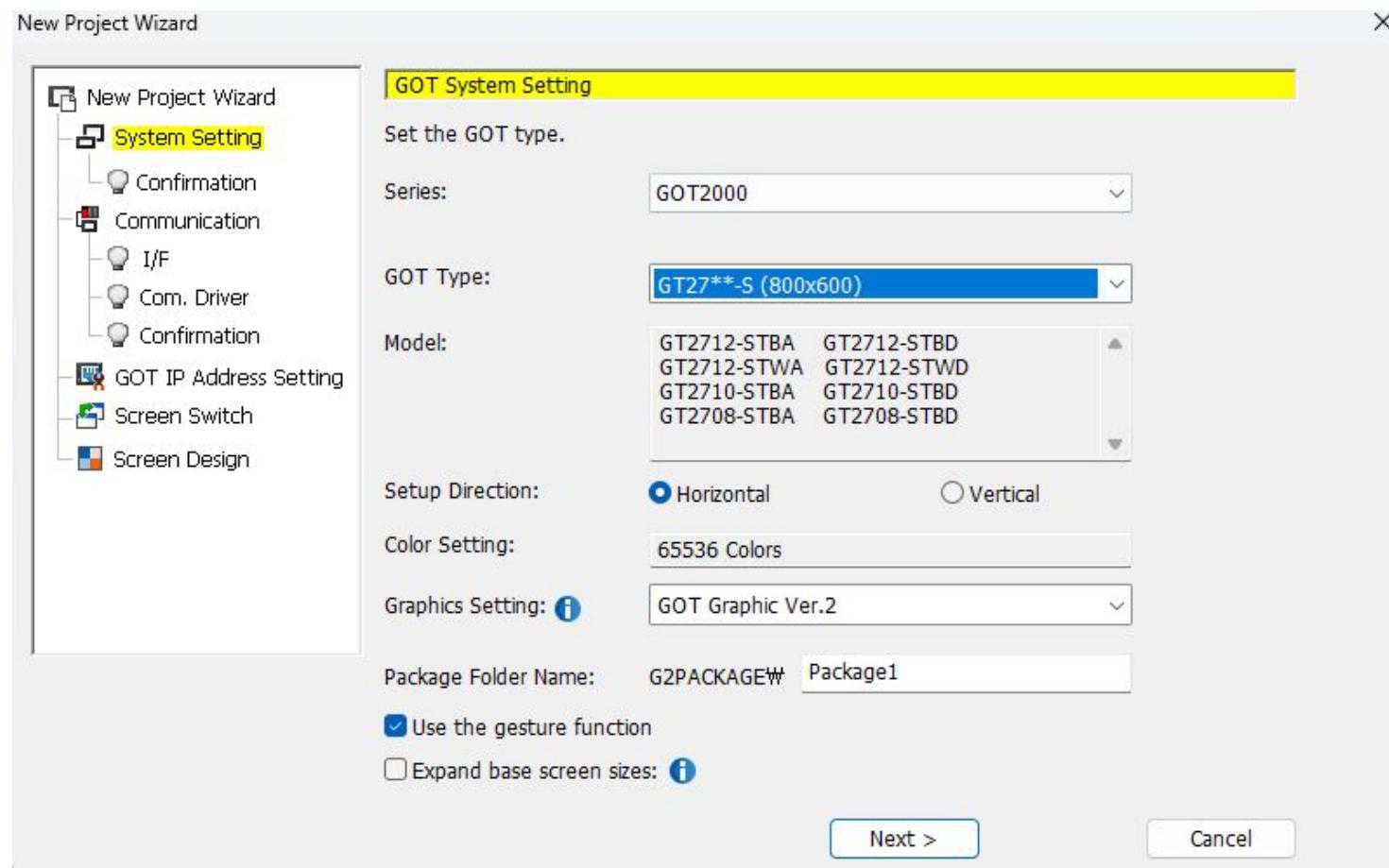


### (3) [Next]를 클릭

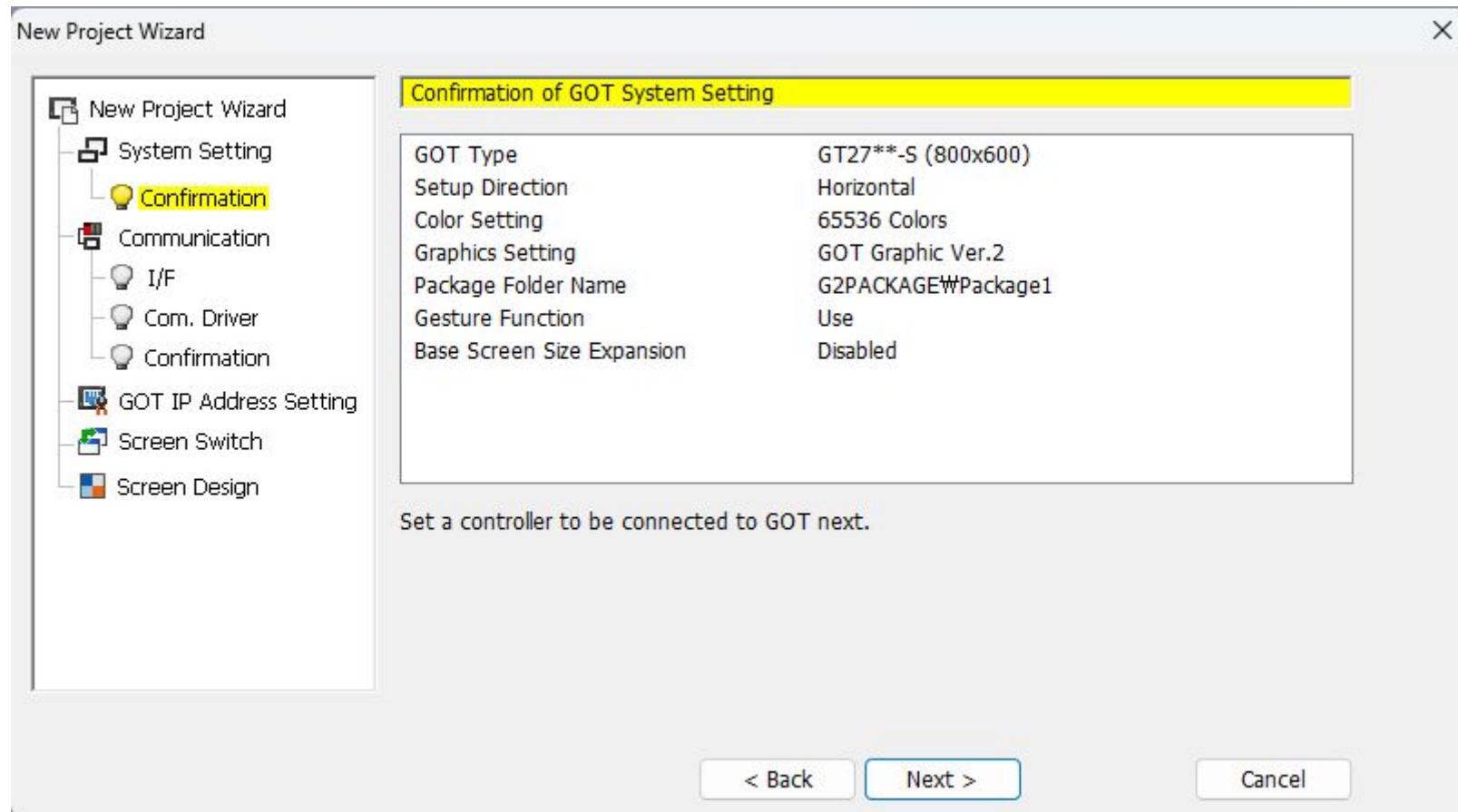


### (4) System setting

- GOT Type : GT27\*\*-S(800 x 600)을 선택하고 [Next] 클릭.

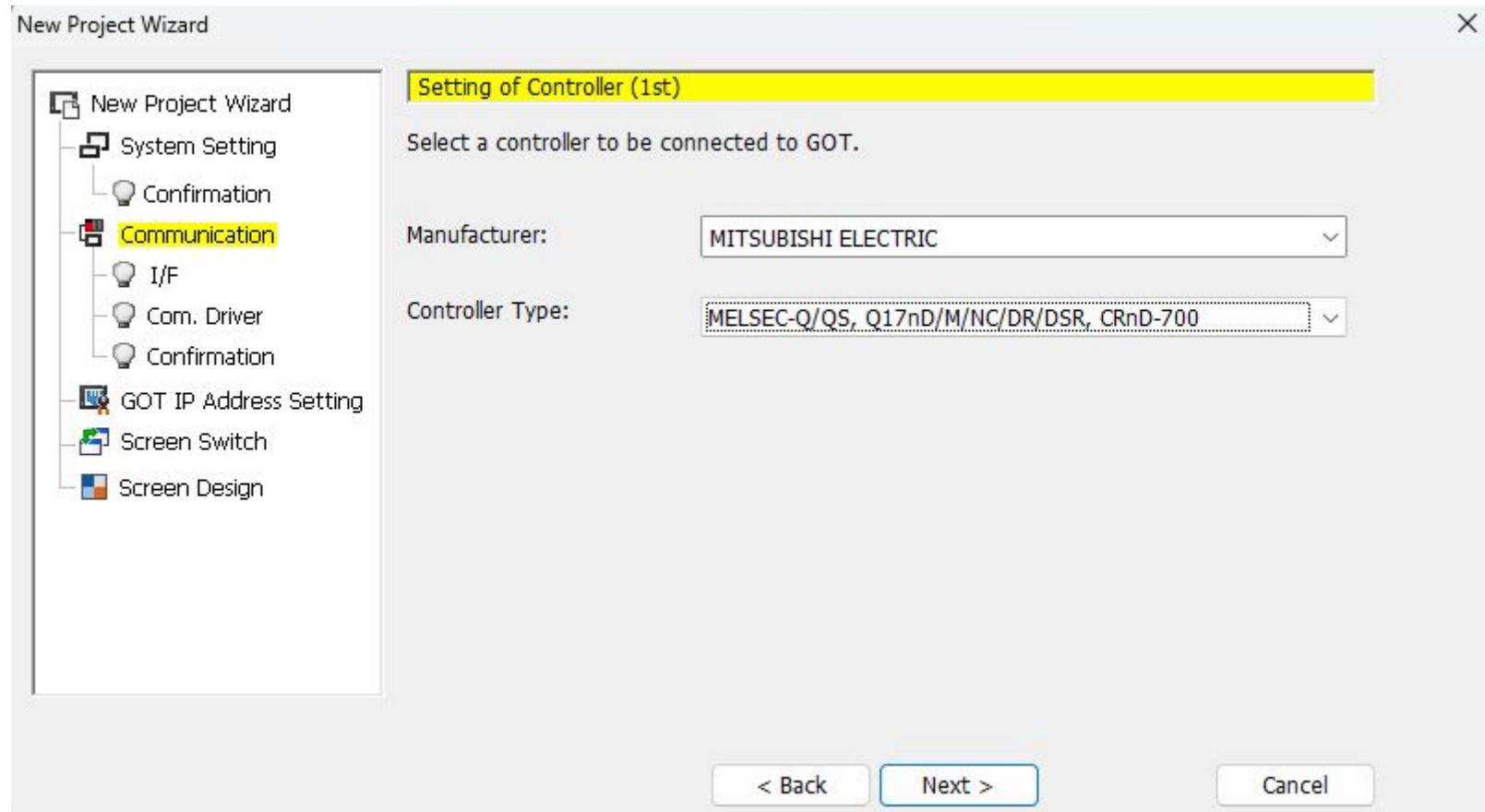


### (5) [Next]를 클릭



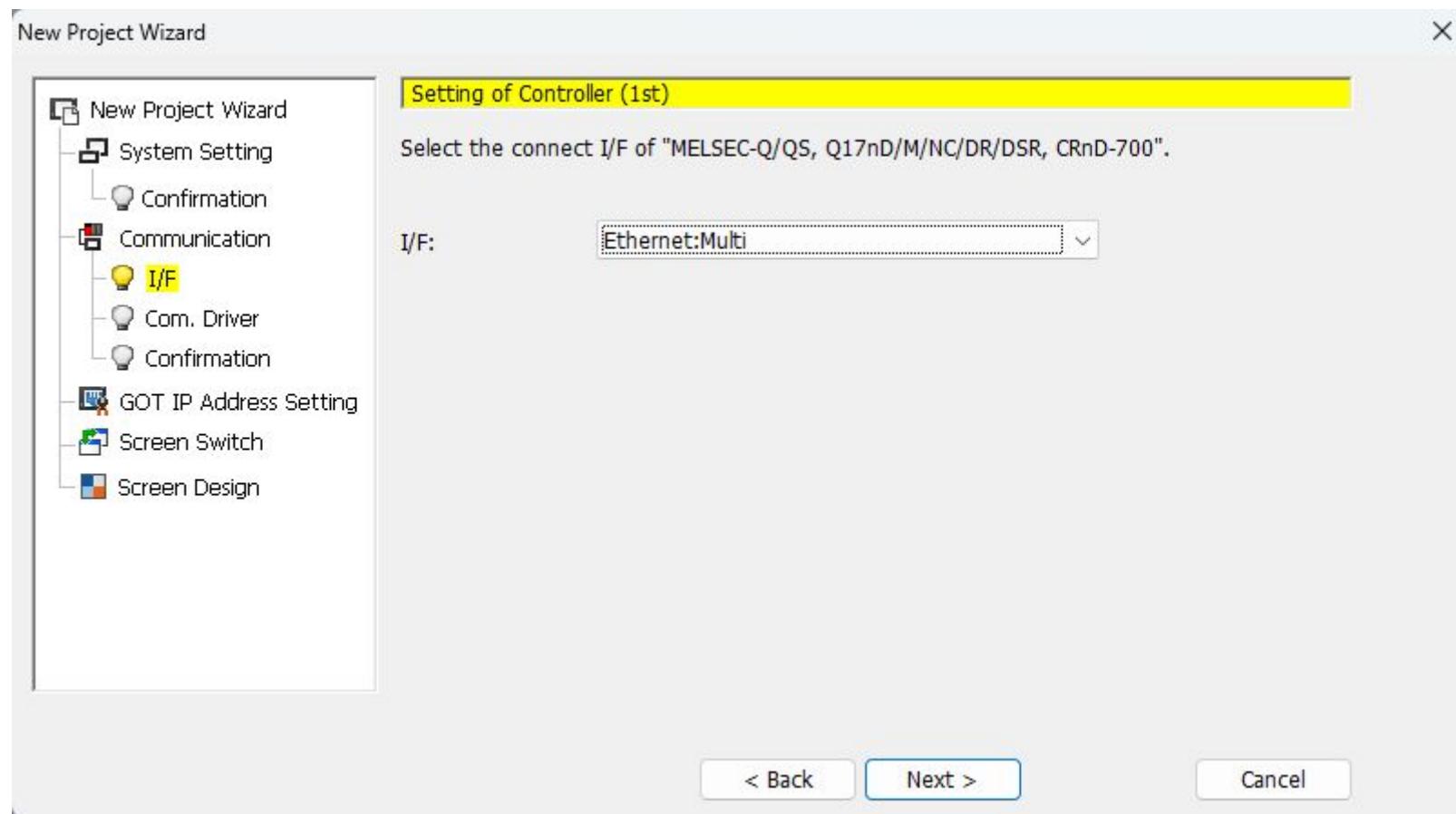
### (6) Controller Type

- MELSEC-Q/QS 선택하고 [Next]를 클릭.

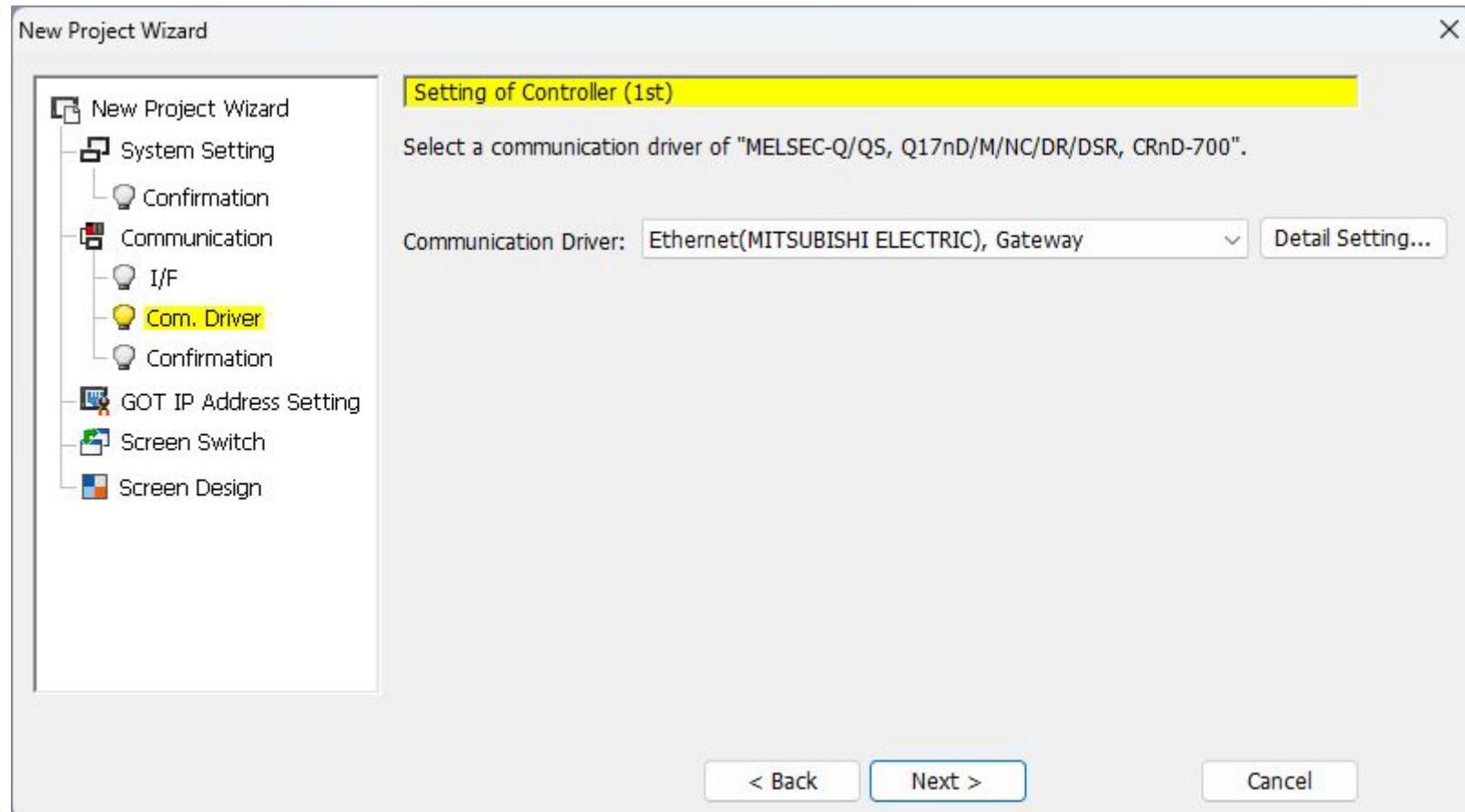


### (7) I/F (Interface) 설정

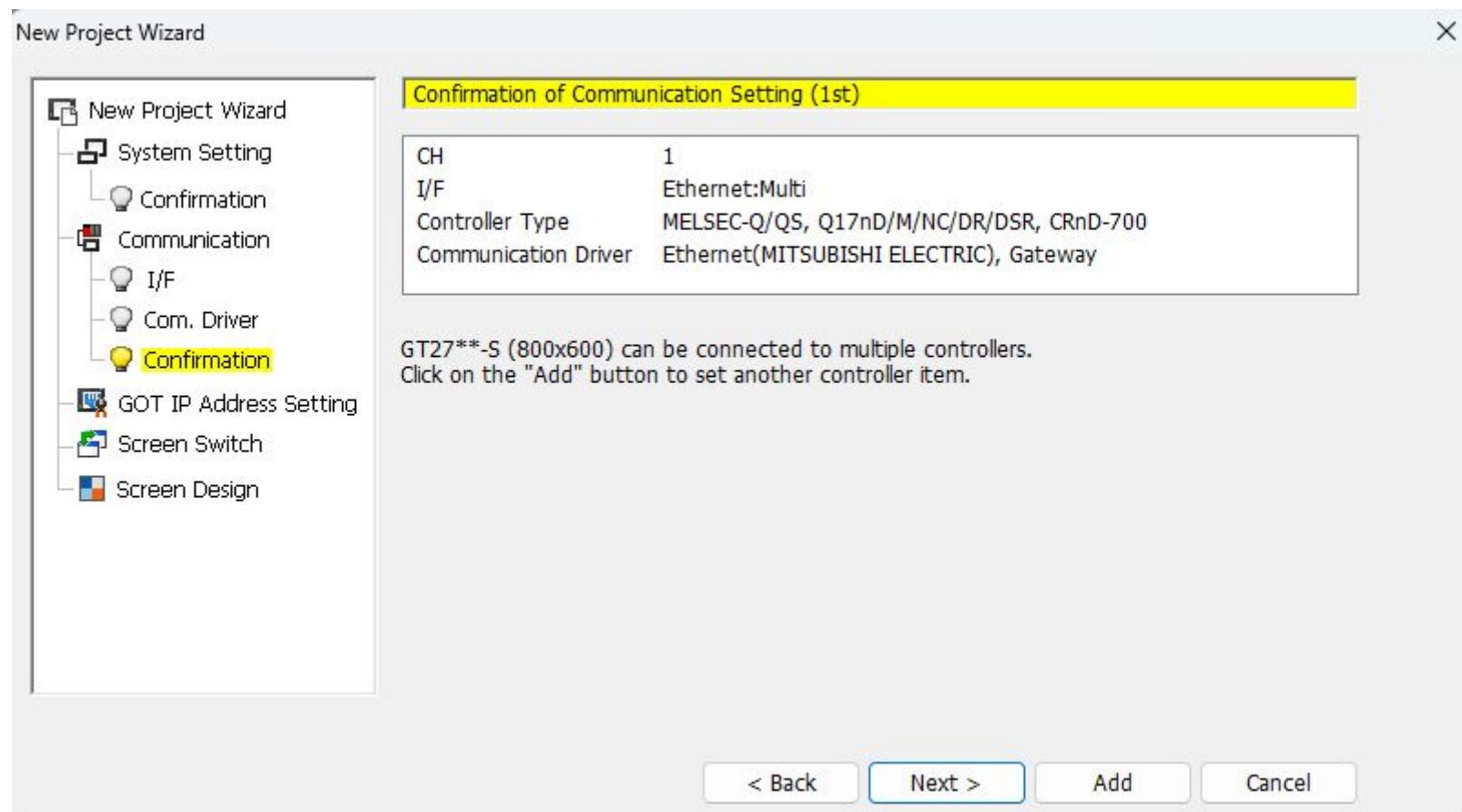
- Ethernet:Multi 선택하고 [Next] 클릭.



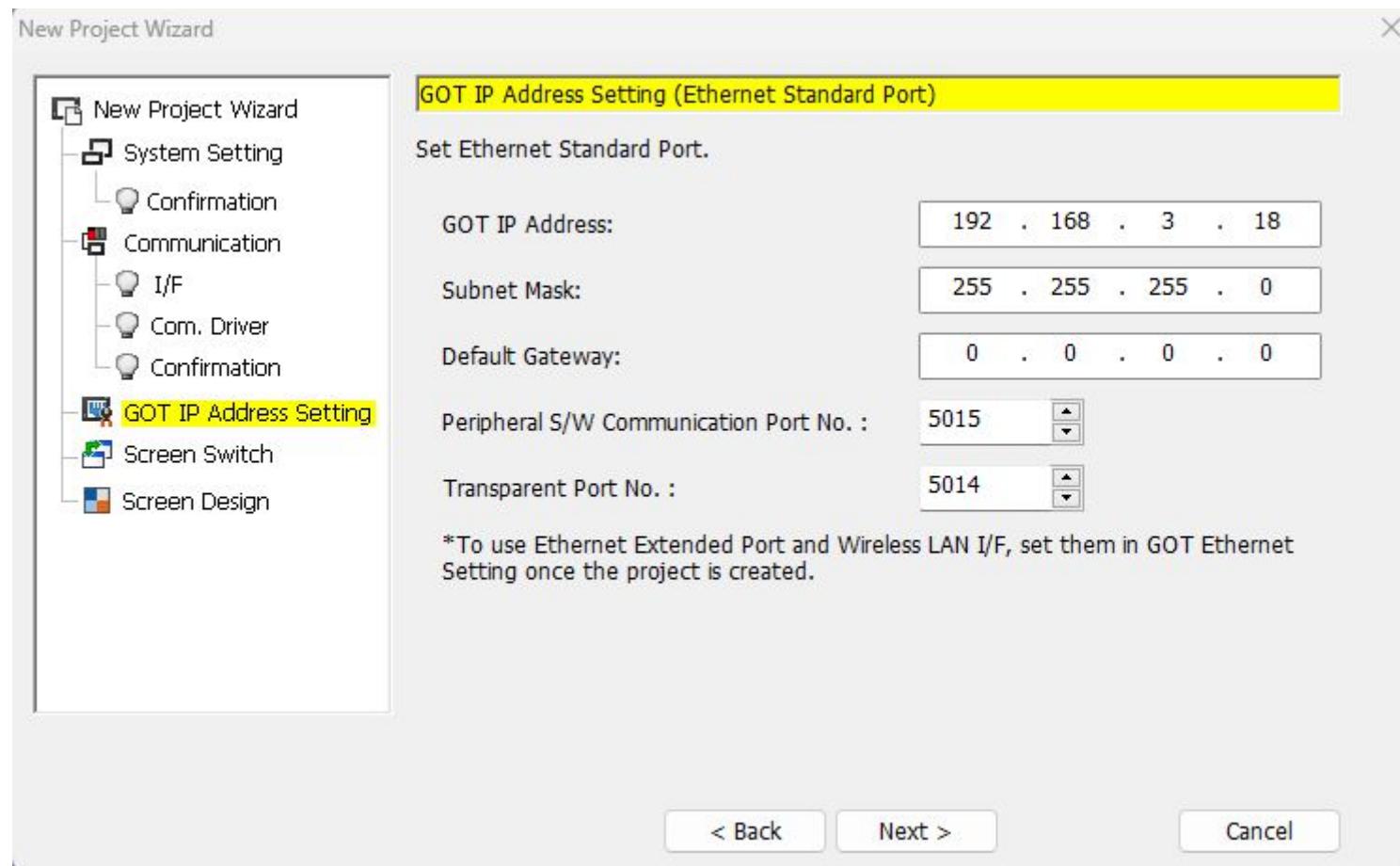
### (8) [Next]를 클릭



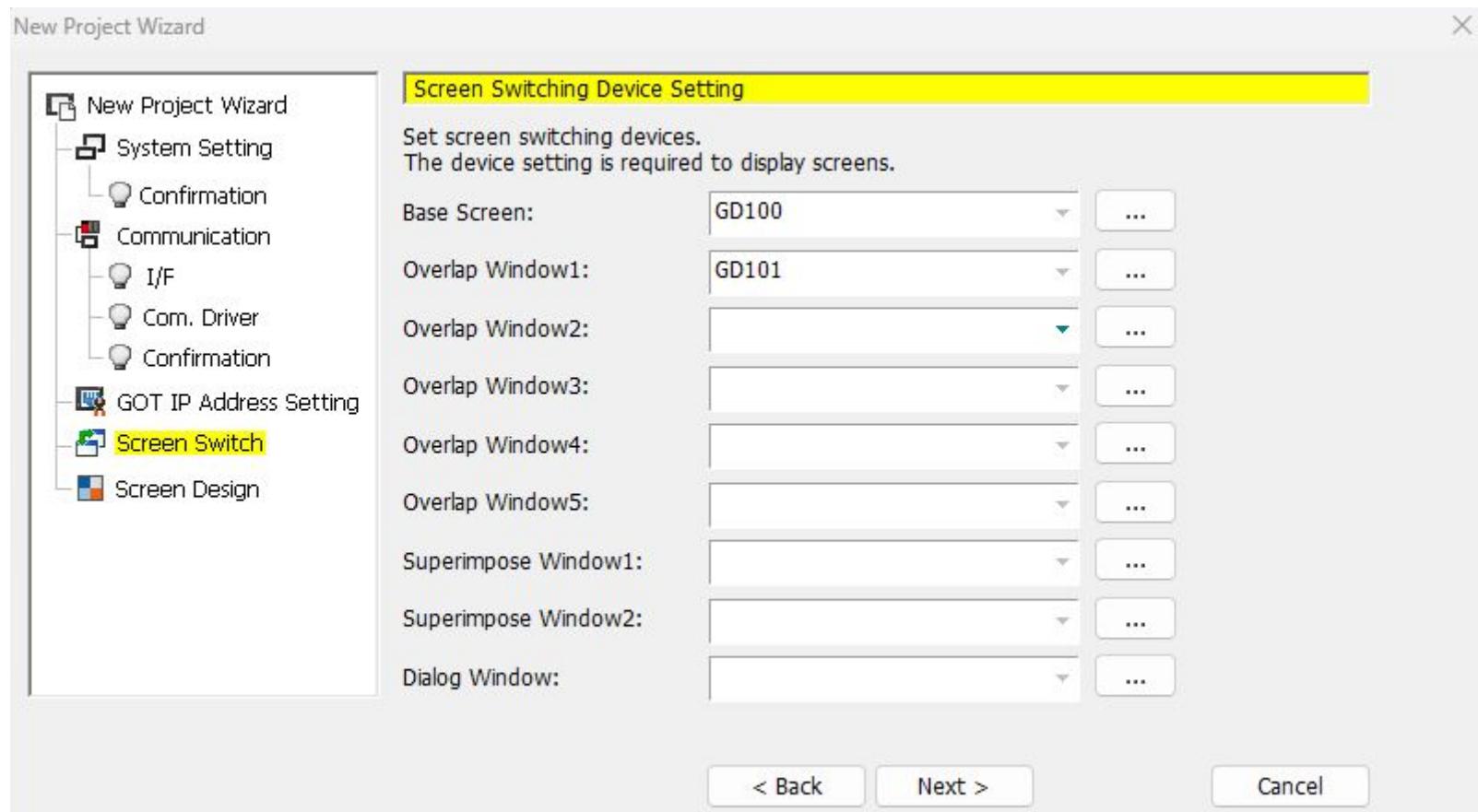
### (9) [Next]를 클릭



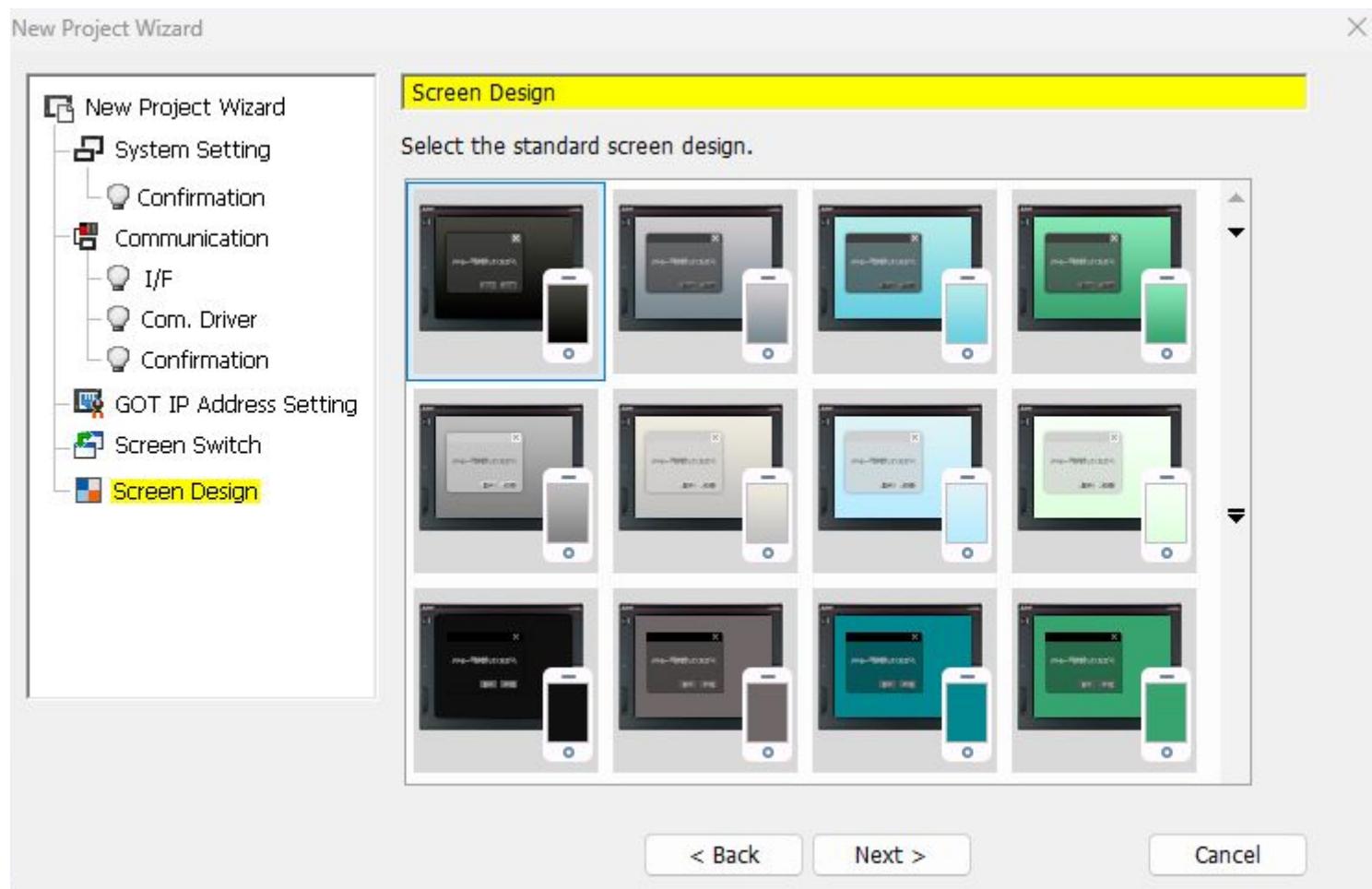
### (10) [Next]를 클릭



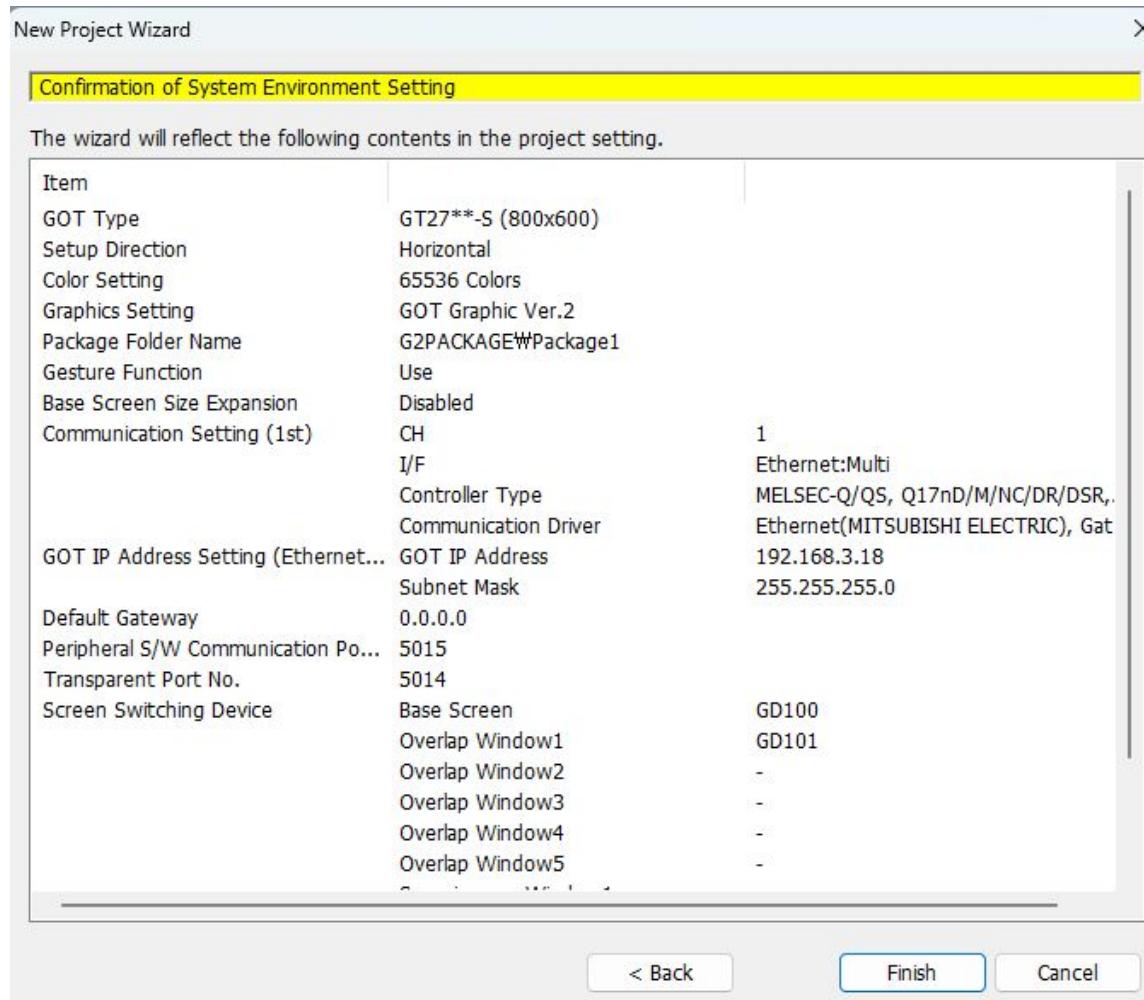
### (11) [Next]를 클릭



### (11) [Next]를 클릭



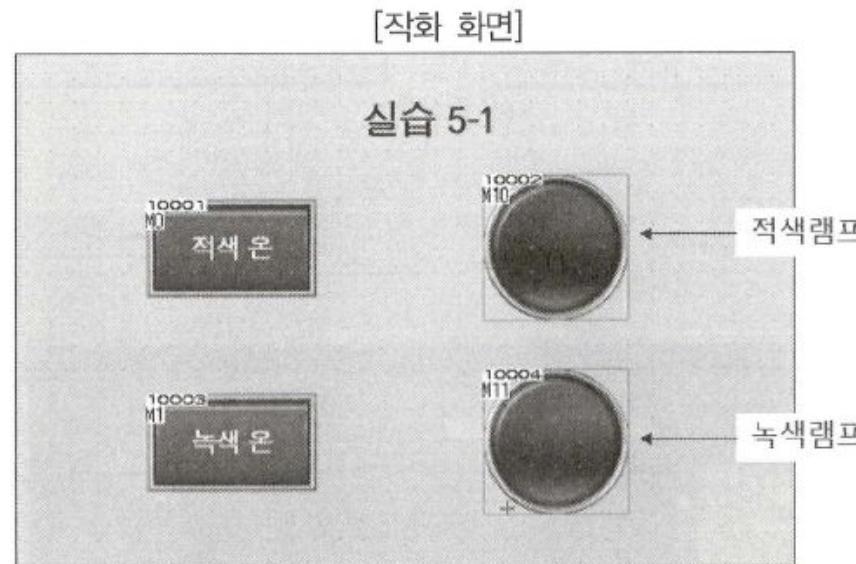
### (11) [Finish]를 클릭



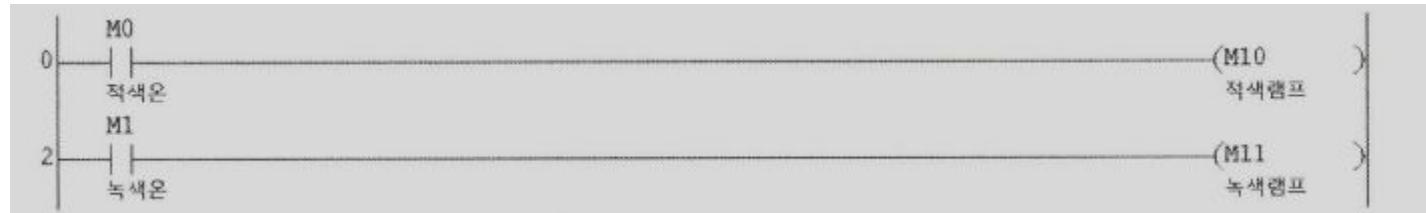
## 5.3 HMI 기초 실습

### 실습 1. 스위치를 이용한 램프 점등

- 문제
  - 2개의 비트 스위치와 램프를 그림과 같이 작화하여 스위치를 누르면 해당 램프가 점등되도록 한다.



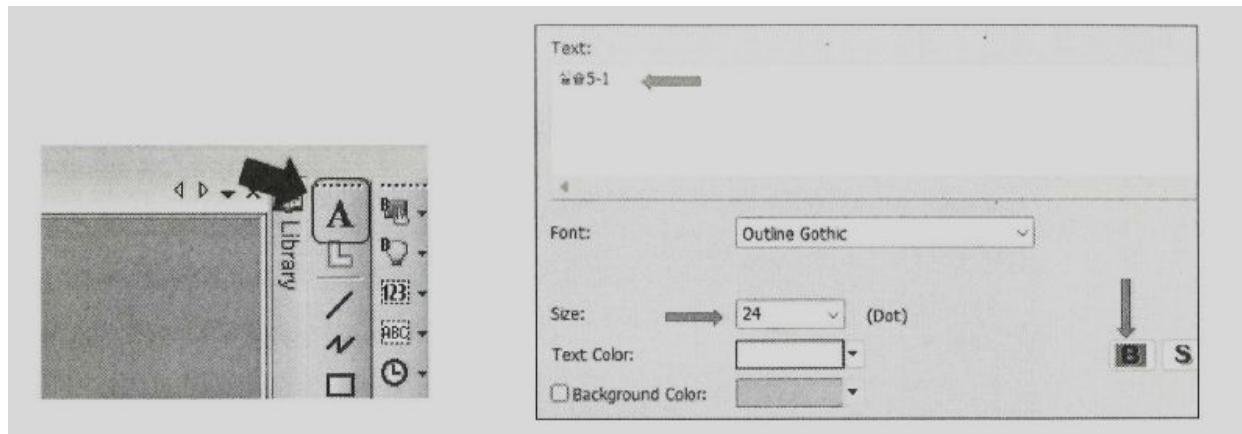
- 래더 프로그램



- 따라하기

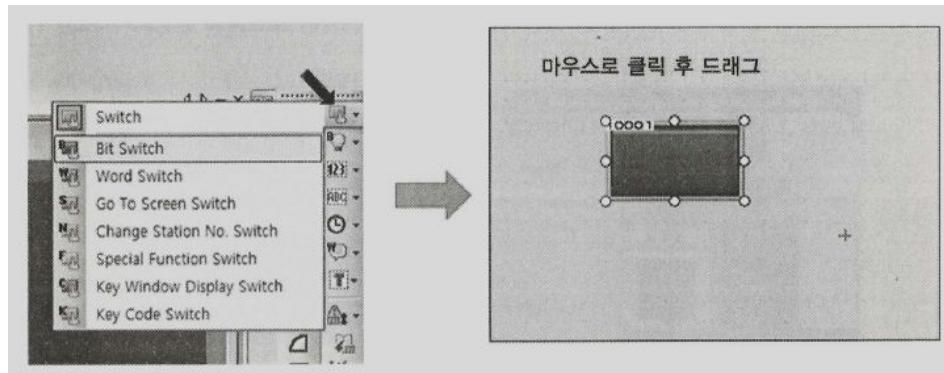
(1) 제목 텍스트 설정

- 텍스트 아이콘을 [A]을 클릭한 다음, 화면상에 배치할 곳에 마우스를 클릭.
- 속성창에서 텍스트(실습 1)를 입력.
- 폰트 사이즈를 크게 설정하고 진하게 [B] 옵션을 지정.



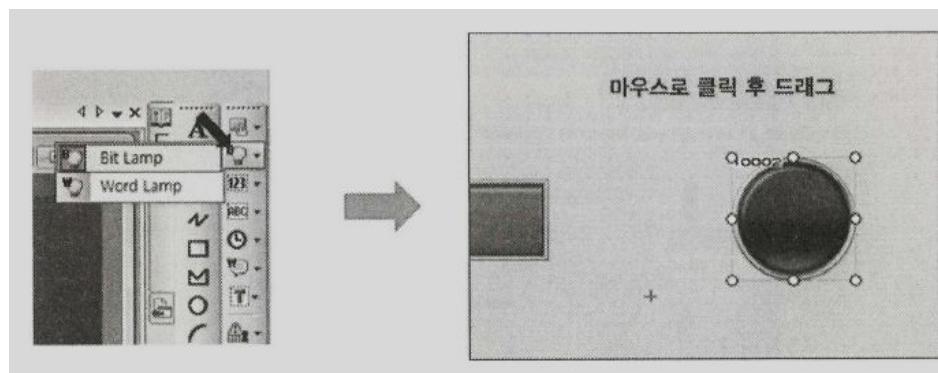
## (2) Bit Switch 배치

- 스위치 메뉴 아이콘을 클릭하여 Bit Switch를 선택한 후 화면에 드래그하여 스위치를 적당한 크기로 2개 배치.



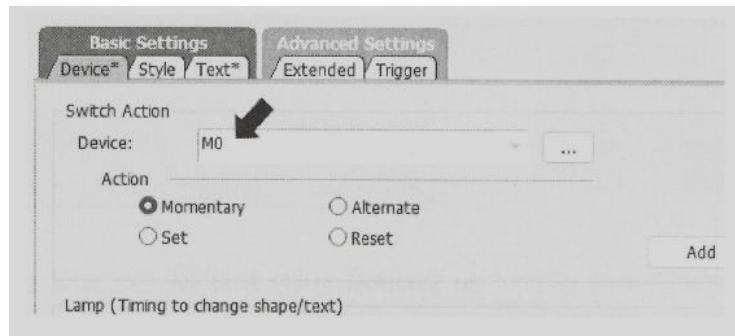
## (3) Bit Lamp 배치

- 램프 메뉴 아이콘을 클릭하여 Bit Lamp를 선택한 후 화면에 드래그하여 램프를 적당한 크기로 2개 배치.

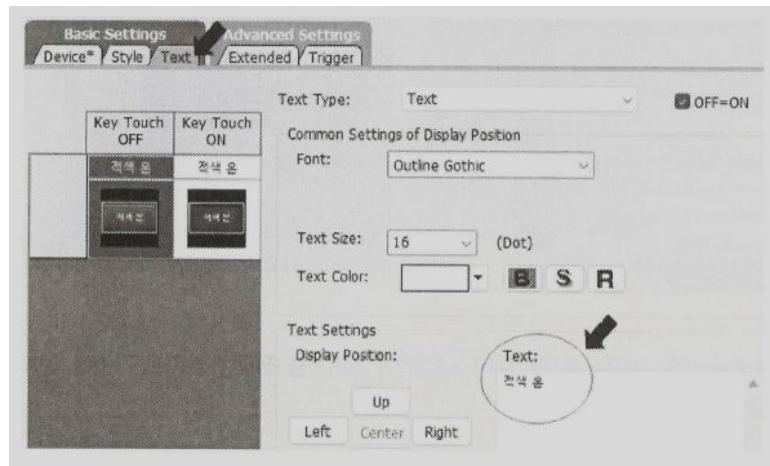


### (4) 스위치 속성 설정 : 연결 디바이스와 텍스트 설정

- 첫 번째 스위치를 클릭하여 Device를 **M0**로 설정.
- 두 번째 스위치를 클릭하여 Device를 **M1**로 설정.

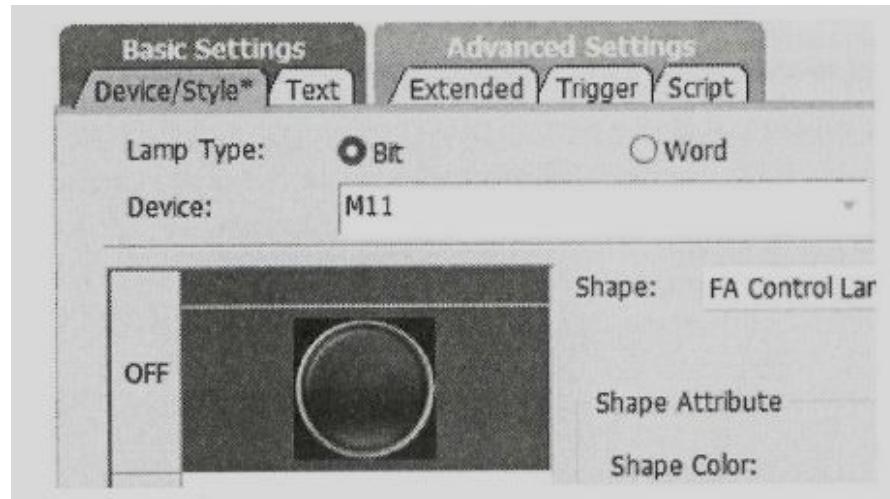


- Text 탭을 선택하여 입력란에 적색 온이라고 입력.

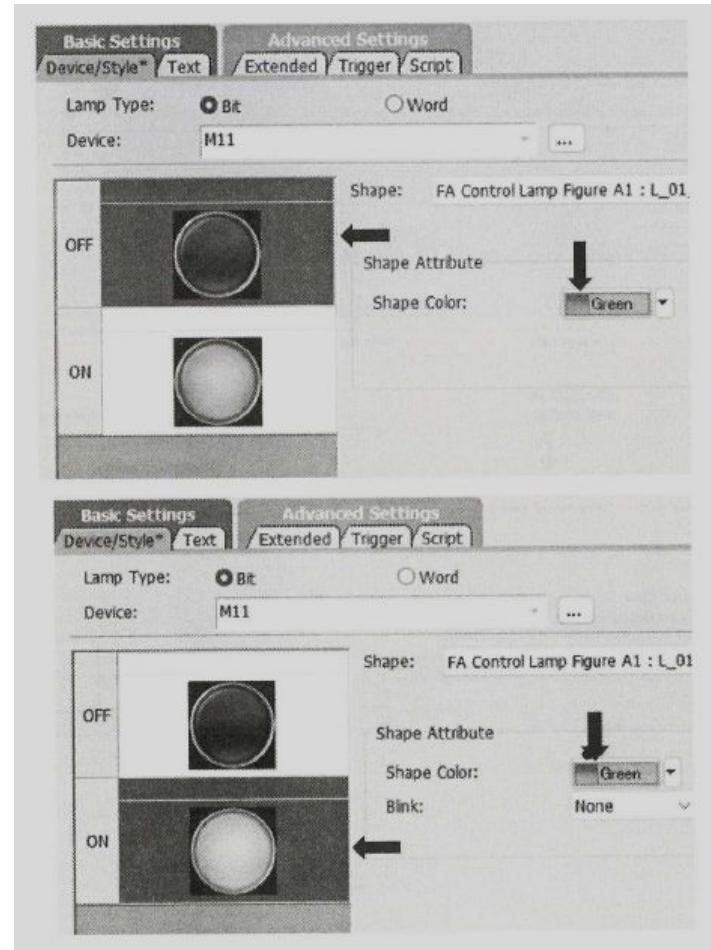


### (5) 램프 속성 설정 : 연결 디바이스와 텍스트 설정

- 첫 번째 램프를 클릭하여 Device를 **M10**로 설정.
- 두 번째 램프를 클릭하여 Device를 **M11**로 설정.

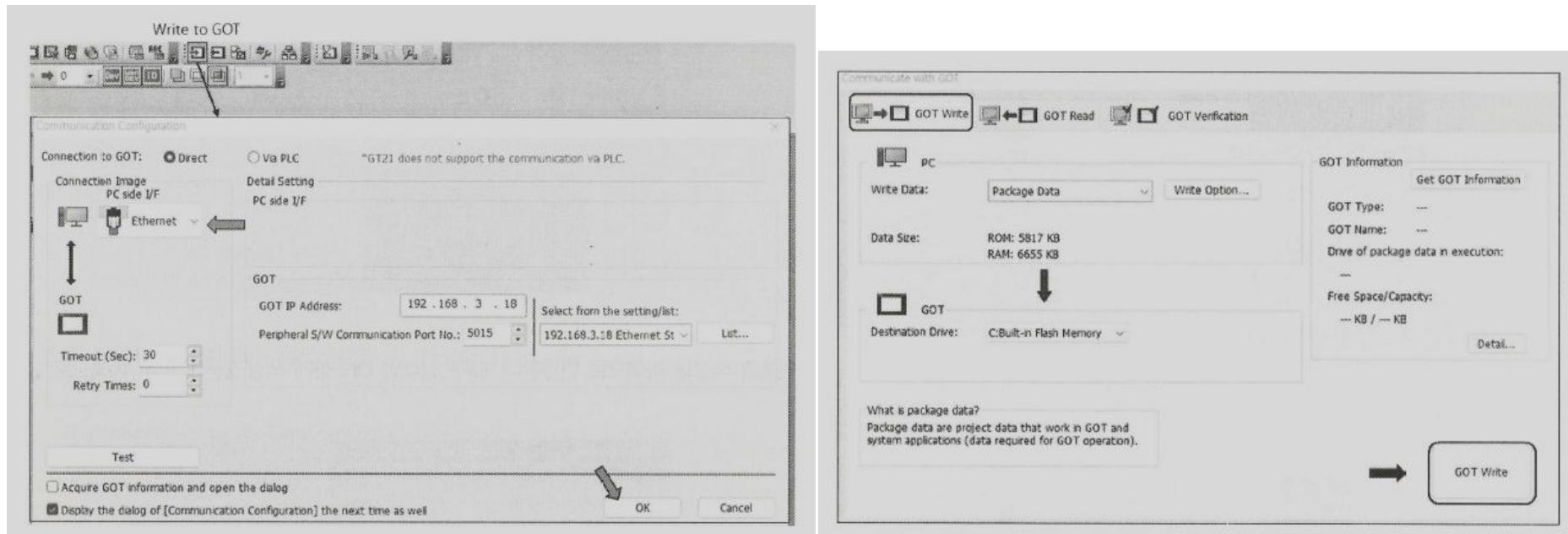


- M11 램프의 색상을 Green 으로 변경.
- OFF 상태와 ON 상태 색상을 모두 녹색으로 설정.



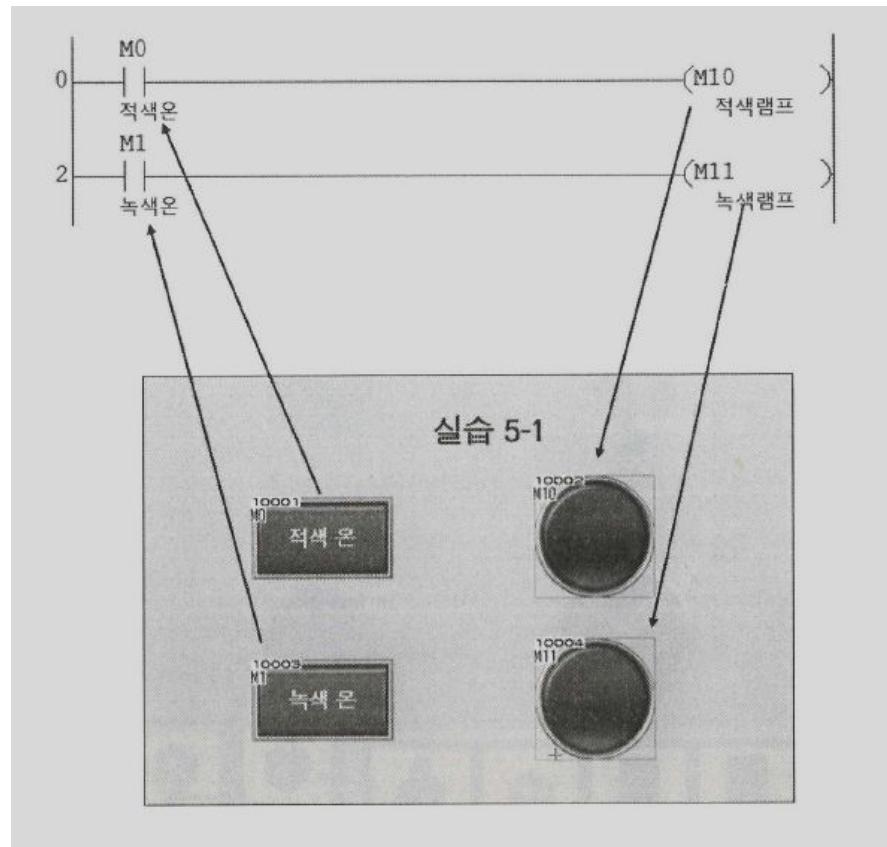
### (6) HMI 화면에 쓰기

- Write to GOT 아이콘 클릭.
- 통신환경은 Ethernet으로 선택하고 [OK] 클릭.
- 다음 Write창에서 [GOT Write]를 클릭.



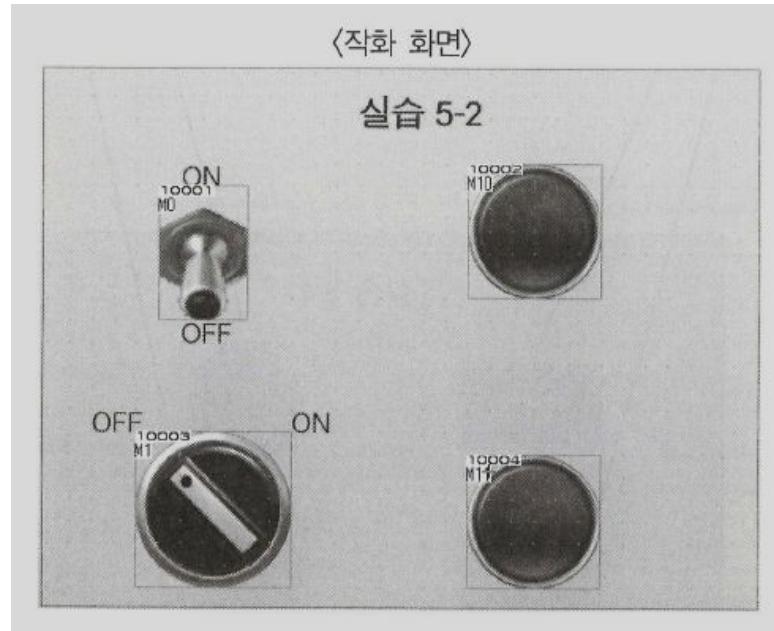
## (7) PLC 래더 프로그램 작성과 실행

- GX-WORKS2에 문제에 제시된 래더 작성한후 PLC에 전송.
- HMI 화면의 스위치를 클릭하여 실행.



## 실습 2. 유지형 스위치를 이용한 램프 점등

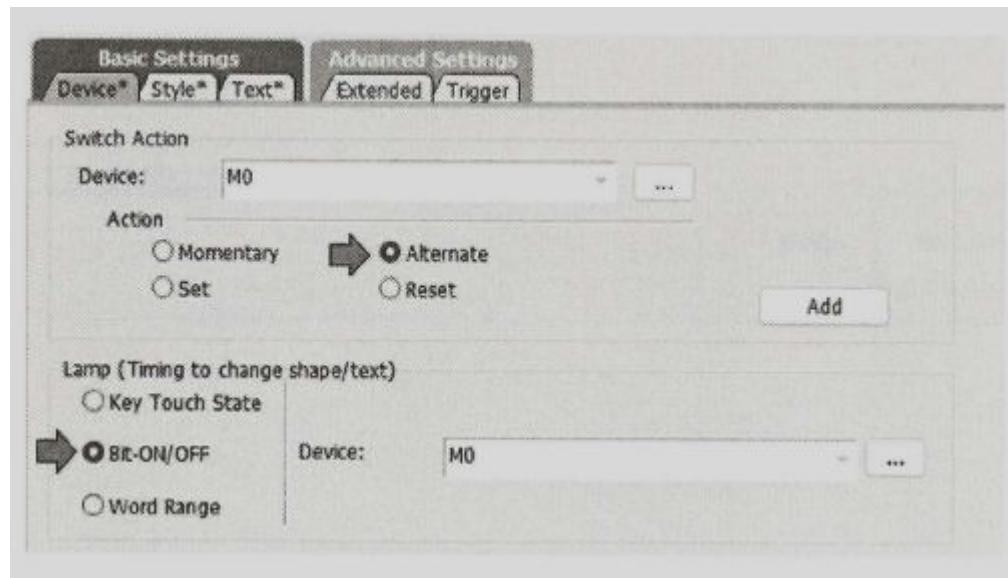
- 문제
- 실습 1에서 스위치를 작화 화면처럼 유지형 스위치인 토글 스위치와 셀렉터 스위치로 수정하여 실행해 보자.



- 따라하기

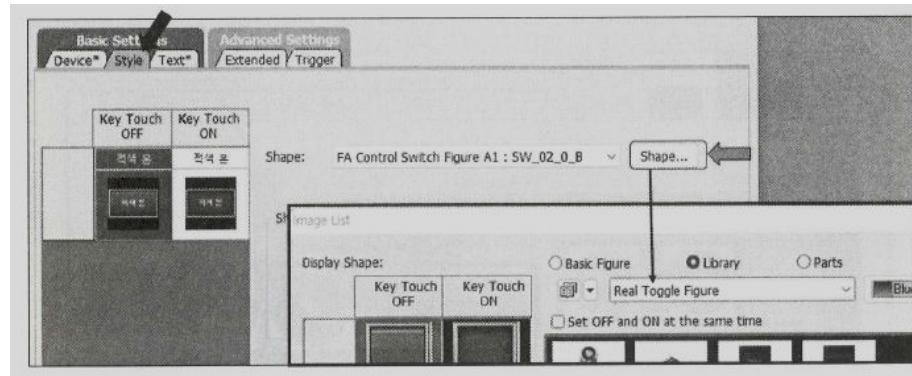
(1) 디바이스 속성을 유지형으로 변경

- M0, M1 스위치를 각각 더블 클릭하여 디바이스 속성을 유지형으로 변경.  
: Alternate, Bit-ON/OFF 를 선택.

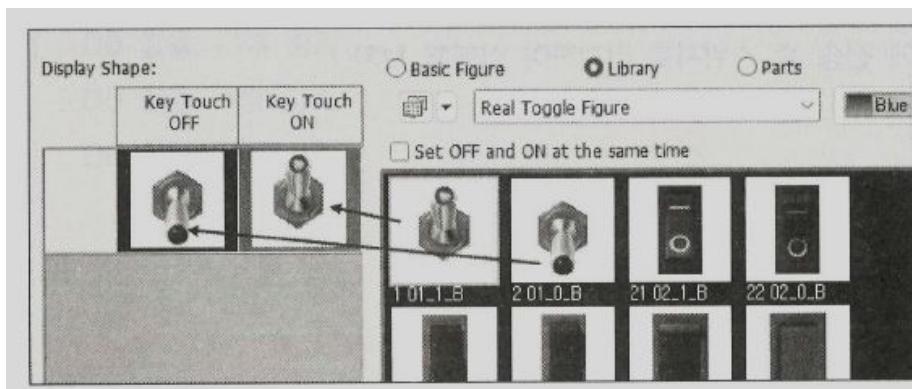


### (2) M0 스위치를 토글 스위치로 변경

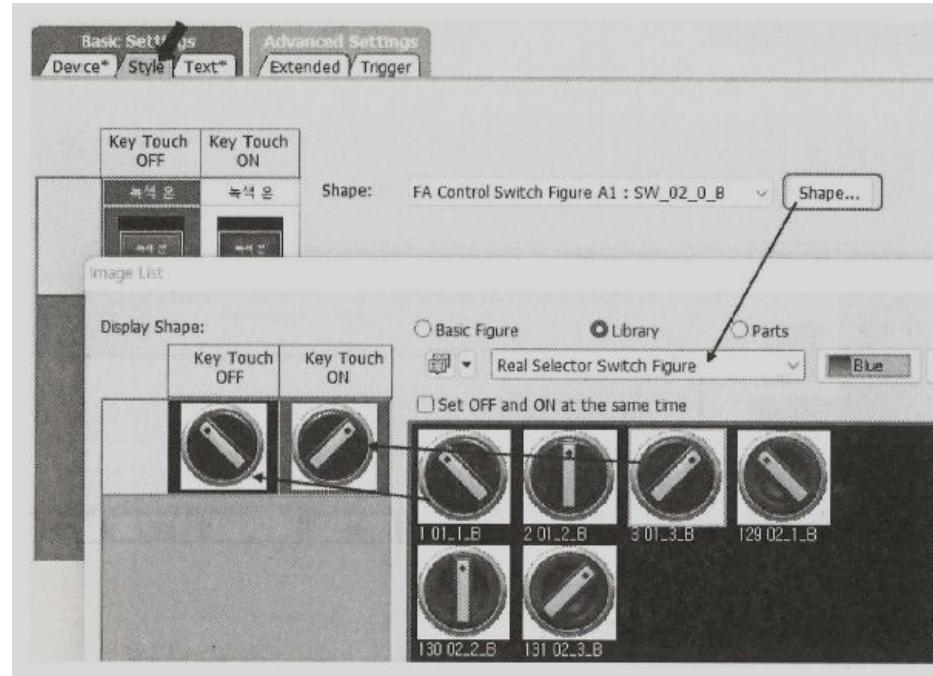
- 스위치 메뉴 아이콘을 클릭하여 속성 창에서 Style 탭에서 [Shape] 버튼을 클릭하여 스크롤바를 맨 위에서부터 찾아 내려가서 [Real Toggle Figure]를 선택.



- OFF일 때와 ON일 때 모양을 선택.



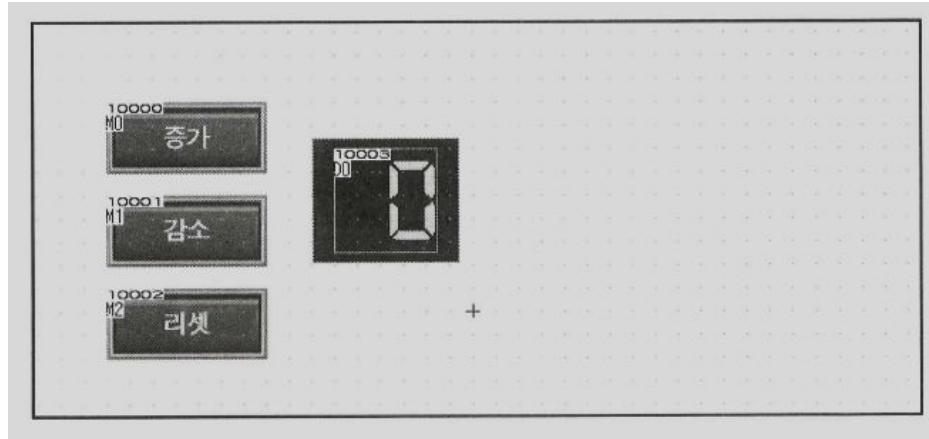
### (3) M1 스위치를 셀렉터 스위치로 변경



### (4) HMI 화면에 전송 후 스위치를 터치하여 실행

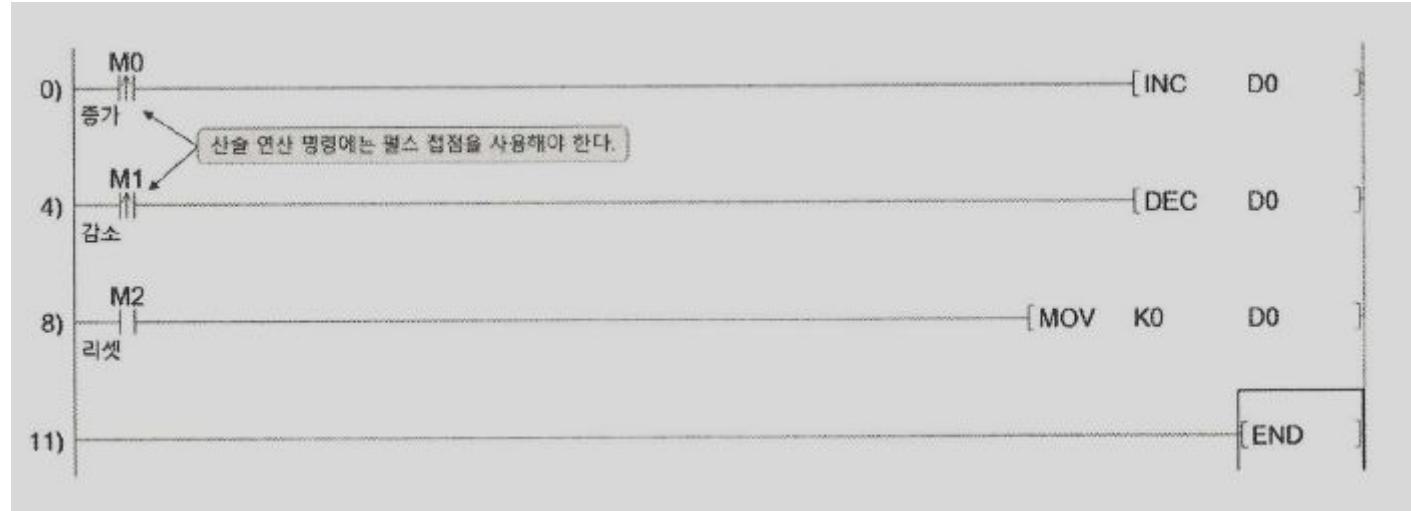
### 실습 3. 디스플레이 제어용 버튼 작화

- 문제
- HMI를 다음과 같이 작화하고, 조작 버튼에 의해 메모리 값이 변하는 것을 숫자표시기에 디스플레이한다.



- 버튼[증가] : D0 값을 +1씩 증가.
- 버튼[감소] : D0 값을 -1씩 감소.
- 버튼[리셋] : D0 값을 0으로 리셋.

- 래더 프로그램



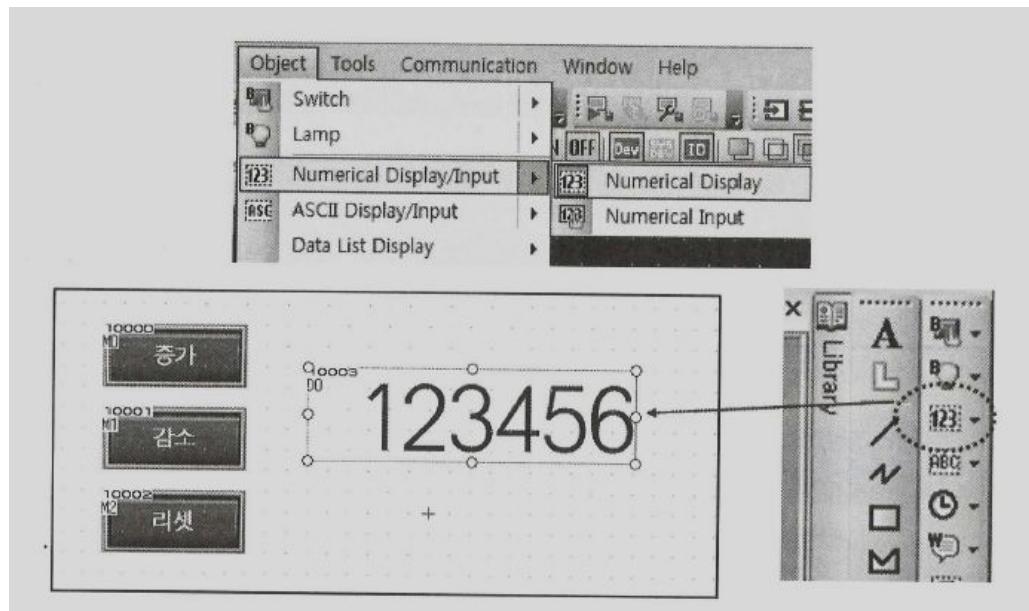
■ 따라하기

(1) 증가, 감소, 리셋 버튼 만들기

- 증가, 감소, 리셋 버튼 배치.
- M0 ~ M2까지 디바이스 설정.

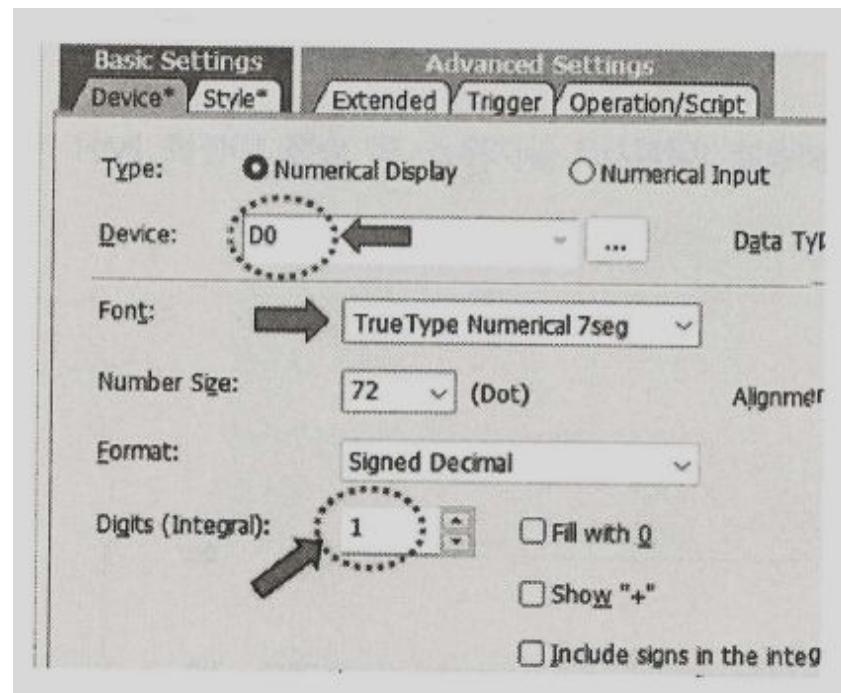
(2) 숫자표시기 표시

- 상단 탭에서 Object–Numerical Display/Input 아이콘에서 Numerical Display를 선택하여 드래그 앤 드롭한다.



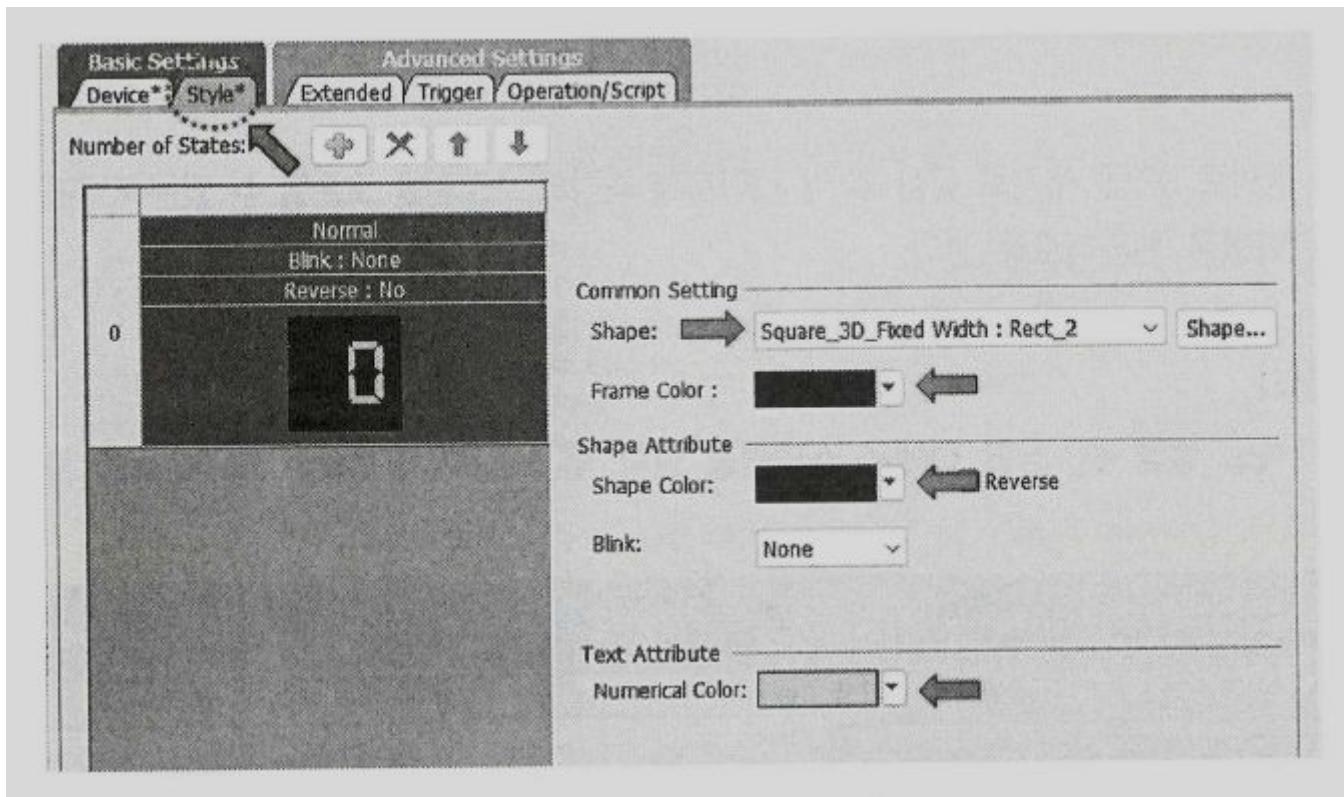
(3) 숫자표시기를 더블 클릭하여 설정창을 오픈한다.

- Device란에 D0 입력
- Font는 TrueType Numerical 7seg 선택
- Digit 란에는 1 설정



### (4) Style 설정

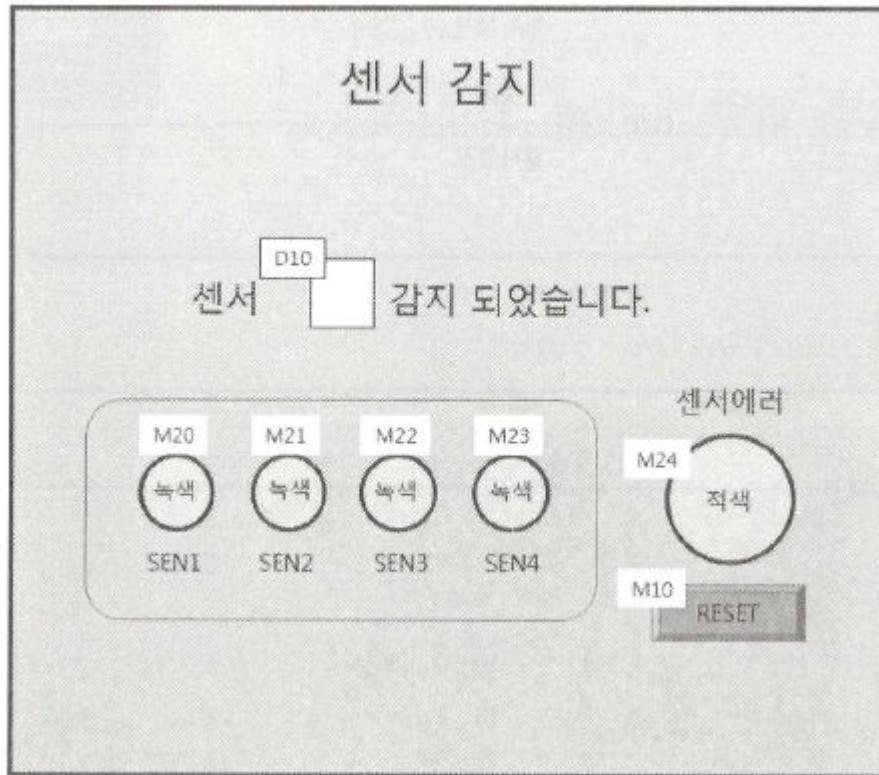
- 상단 Style 탭을 클릭한 후 Shape를 적절한 디자인으로 선택.
- Frame Color, Shape Color, Numerical Color을 이용하여 디자인 상 보기에 편한 적절한 색상을 선택.



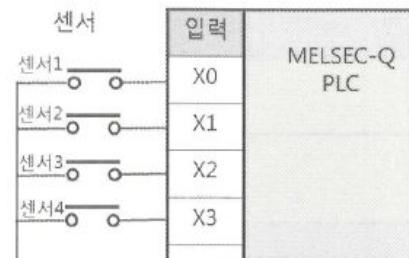
## 5.4 HMI 활용 실습

### 실습 1. 센서의 입력과 표시

- 화면 작화와 디바이스 배정



- 문제
- 다음과 같이 센서가 입력에 연결되어 있다. 입력된 센서 신호에 따라 센서 출력 문구에 센서 번호가 출력되고 해당 램프가 점등되도록 하시오. 단 센서는 4개 중 1개만 감지된다고 가정한다.
- 센서2와 센서3이 동시에 감지되면 센서에러 램프(적색)가 1초 간격으로 계속 점멸한다.
- RESET 버튼을 클릭하면 센서에러 램프는 OFF되고 센서 문구의 숫자는 0이 되도록 한다.



센서 입력	문구 숫자 출력	SEN1 SEN2 SEN3 SEN4			
		SEN1	SEN2	SEN3	SEN4
센서1	센서 [1] 감지되었습니다	ON	OFF	OFF	OFF
센서2	센서 [2] 감지되었습니다	OFF	ON	OFF	OFF
센서3	센서 [3] 감지되었습니다	OFF	OFF	ON	OFF
센서4	센서 [4] 감지되었습니다	OFF	OFF	OFF	ON

### ▪ Ladder Program 작성

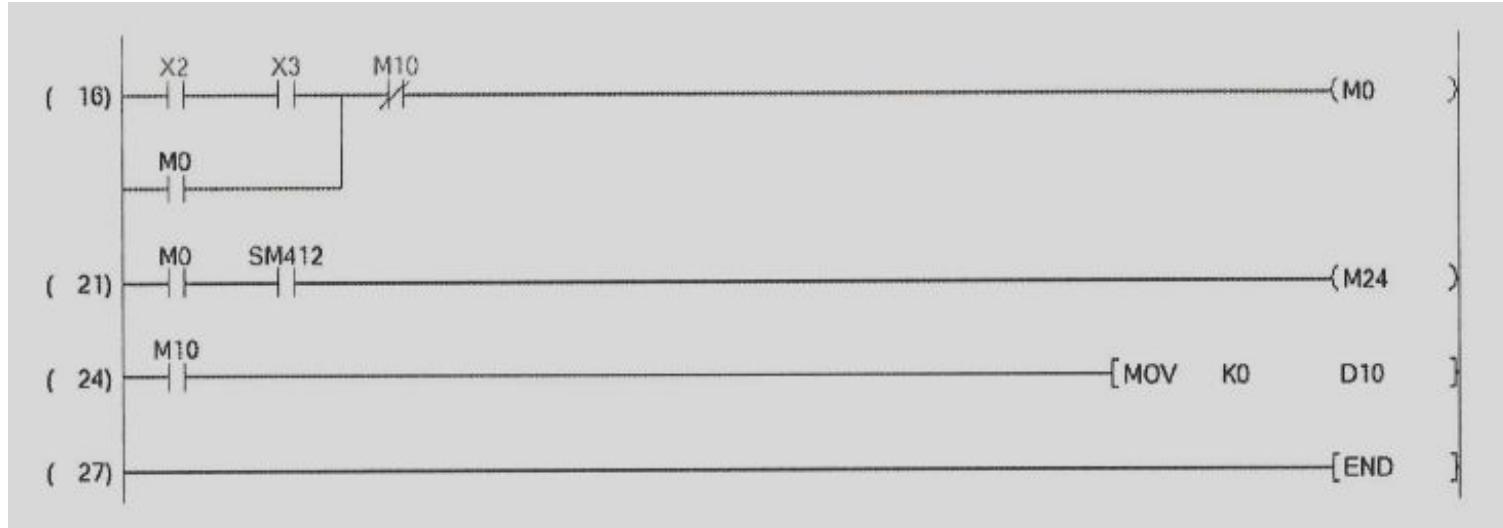
#### ① 센서값 저장과 표시 및 램프 온



#### ❖ 데이터 전송 명령

- MOV S D : S를 D에 전송.

② 센서2, 센서3 동시감지 처리

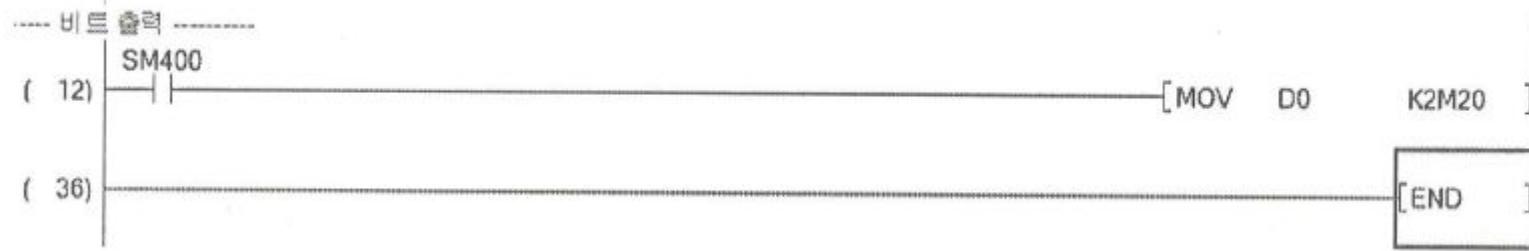


❖ 특수릴레이

- SM400 : 상시 ON
- SM401 : 상시 OFF
- SM402 : Run 후 1스캔만 ON
- SM403 : Run 후 1스캔만 OFF
- SM409 : 0.01초 클럭(일정시간 간격으로 ON/OFF를 반복하는 접점)
- SM410 : 0.1초 클럭
- SM411 : 0.2초 클럭
- SM412 : 1초 클럭
- SM413 : 2초 클럭

③ 램프 8비트 출력.

- [D0.0 ~ D0.7] → [M20 ~ M27]

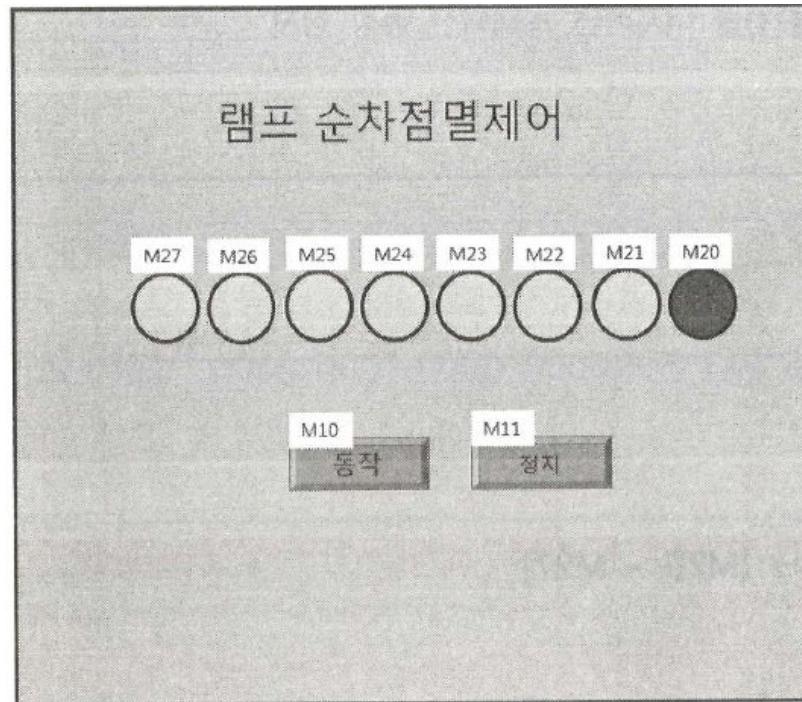


❖ [MOV D0 K2M20]

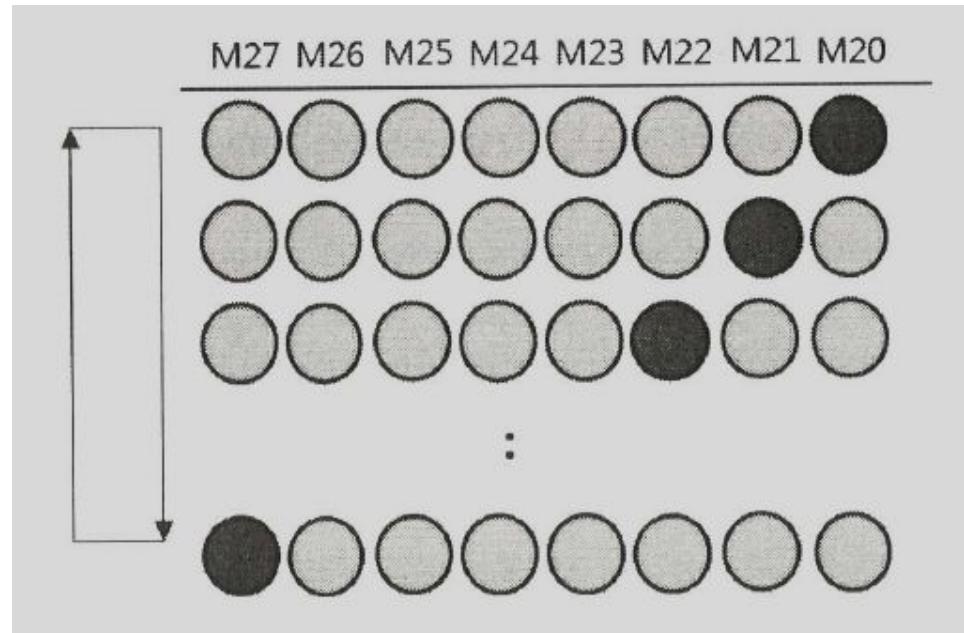
- D0의 하위바이트(D0.0 ~ D0.7)을 내부메모리 M20 ~ M27에 저장

## 실습 2. 램프 순차 점등제어 1

- 화면 작화와 디바이스 배정



- 문제
- 다음과 같이 8개의 램프를 0.2초 간격으로 순차적으로 한 개씩 시프트 점등시키도록 한다.
- [동작] 버튼을 클릭하면 다음과 같이 순차로 무한반복 되도록 한다.
- [정지] 버튼을 클릭하면 램프는 시프트 동작을 중지한다. 다시 [동작] 버튼을 누르면 현 점등 위치에서 다시 시프트 동작을 개시한다.



- Ladder Program 작성

- ① 16진수 워드데이터 저장

- 초기값 16진수 워드데이터 H0101(이진수 : 00000001 00000001)을 D0에 저장.



- ❖ 특수릴레이

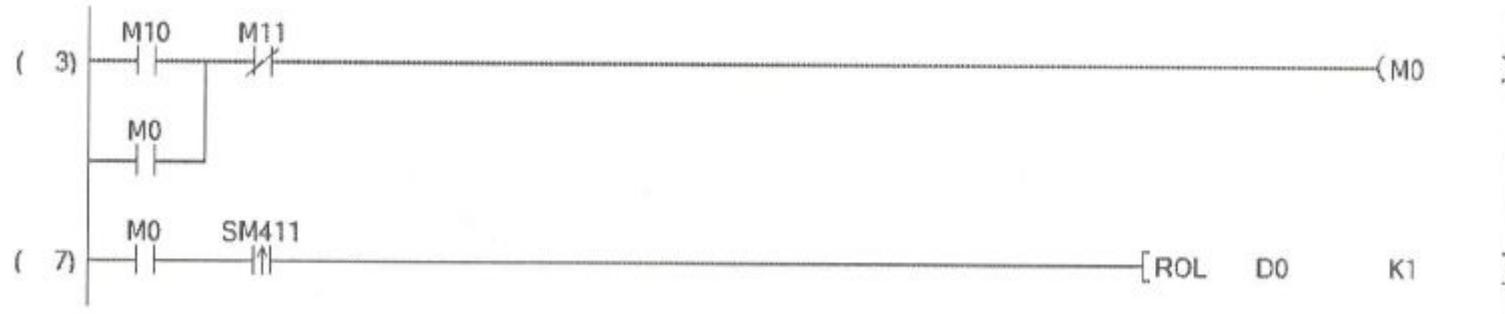
- SM400 : 상시 ON
- SM401 : 상시 OFF
- SM402 : Run 후 1스캔만 ON
- SM403 : Run 후 1스캔만 OFF
- SM409 : 0.01초 클럭(일정시간 간격으로 ON/OFF를 반복하는 접점)
- SM410 : 0.1초 클럭
- SM411 : 0.2초 클럭
- SM412 : 1초 클럭
- SM413 : 2초 클럭

- ❖ 데이터 전송 명령

- MOV S D : S를 D에 전송.

## ② 왼쪽 로테이션 수행

- 시작 버튼(M10)을 터치하면 0.2클록(SM411)을 이용하여 0.2초마다 1비트씩 왼쪽 로테이션 수행.
- 정지 버튼(M11)을 터치하면 로테이션 동작 정지.

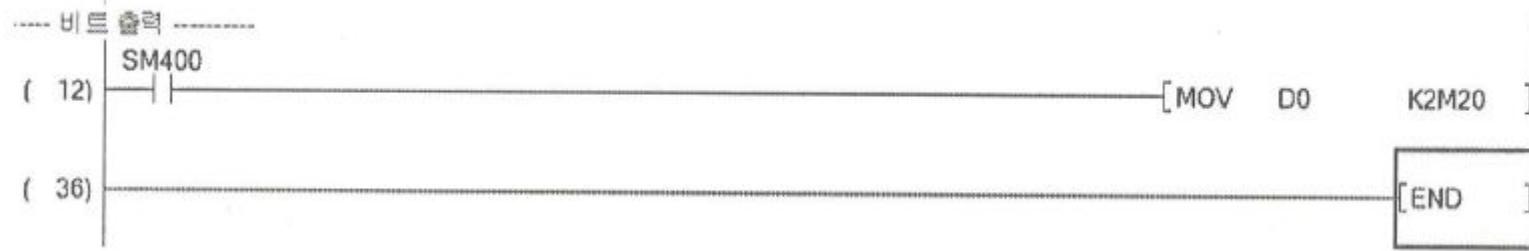


### ❖ 왼쪽 로테이션 명령

- [ROL D0 1] : D0의 데이터를 1비트씩 왼쪽으로 로테이션 수행.

③ 램프 8비트 출력.

- [D0.0 ~ D0.7] → [M20 ~ M27]



❖ [MOV D0 K2M20]

- D0의 하위바이트(D0.0 ~ D0.7)을 내부메모리 M20 ~ M27에 저장