

What Is A Programmable Logic Controller (PLC)?

▶ PLC 란 무엇인가 ?

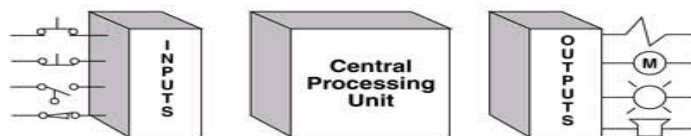
PROGRAMMABLE LOGIC CONTROLLER (PLC)는 입력 디바이스의 상태를 지속적으로 모니터하고 출력장치의 상태를 제어하기 위한 맞춤 프로그램에 기반하여 제어를 하는 공업컴퓨터 제어시스템이다. 즉, PLC는 논리연산, 순서조작, 타이머, 카운터, 수리연산 등의 제어동작을 행하고 디지털/아날로그 입출력을 통해 행하는 디지털 조작 공업용 제어장치이다. 대부분의 설비라인, 기계기능, 또는 공정은 PLC 제어시스템을 사용하여 최대로 작동될 수 있다. 그럼에도, PLC 사용에서 가장 큰 혜택은 생생한 정보를 수집하고 소통하면서 조작과 처리를 반복하고 바꾸는 능력이다. PLC 시스템의 다른 장점은 이것이 모듈화되어 있다는 점이다. 즉, 사용자의 필요에 의하여 입력과 출력장치를 조합하고 맞출 수 있다.

▶ PLC의 역사

최초의 PLC는 GM과 Landis 용 릴레이 대체품으로 모디콘에 의해 디자인되고 개발되었다.

- 이 제어기는 각각의 새로운 로직 구성의 부가적인 하드웨어를 다시 연결하고 부가해야 할 필요성을 없앴다.
- 새로운 시스템은 철저히 로직을 덮고 있는 케비넷 공간을 줄이는 동시에 제어의 기능을 철저하게 증가시켰다.
- 최초의 PLC, model 084,는 Dick Morley에 의해 1969년 창안되었다.
- 최초 수치적으로 성공적인 PLC인 184는 1973년 소개되었고 Michael Greenberg에 의해 디자인되었다.
-

▶ PLC의 내부 구성은 ?



CPU는 PLC가 아래와 같은 기능을 어떻게 수행하는가를 말해주는 내부 프로그램을 포함하고 있다:

- 사용자 프로그램에 포함된 제어기구를 실행. 이 프로그램은 전원이 꺼져도 프로그램이 저장되어 있는 불휘발성(nonvolatile) 메모리에서 저장되어 있다.

- I/O 디바이스, 프로그램 장치, 네트워크, 심지어 다른 PLC 등 여타 장치와 통신한다.
- 통신, 내부진단 등과 같은 하우스키핑(housekeeping) 활동을 수행한다.

🔥 어떻게 PLC 를 작동시키는가?

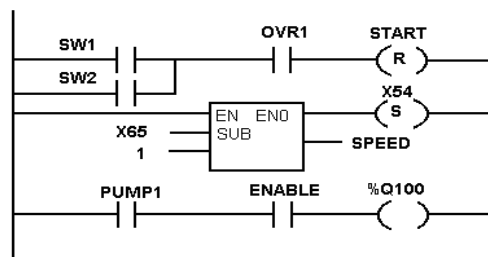
모든 PLC의 작동에는 네 가지의 기본적인 절차가 있다: **Input Scan**, **Program Scan**, **Output Scan**, 그리고 **Housekeeping**. 이들 각 단계들은 반복 루프(loop)에서 지속적으로 일어난다.

- 1.) **Input Scan**: PLC와 연결된 모든 입력 장치들의 상태를 탐색한다.
- 2.) **Program Scan**: 사용자가 지정한 로직(logic)을 실행한다.
- 3.) **Output Scan**: PLC에 연결된 모든 출력장치에 전원을 공급하거나 제거한다.
- 4.) **Housekeeping**: 프로그램 터미널, 내부진단 등에 의한 통신을 포함한다.

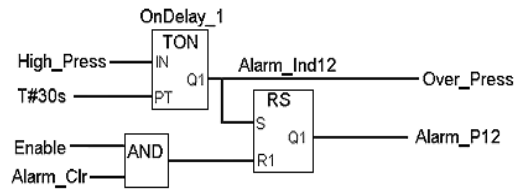
🔥 PLC 를 프로그래밍 하기 위해 사용되는 프로그래밍 언어로는 무엇이 있는가?

레더 로직이 가장 일반적으로 사용된 PLC 프로그래밍 언어이지만, 유일 한 것은 아니다. 다음 테이블 목록은 PLC 프로그램에 사용된 언어들이다.

레더 다이어그램(LD): 전통적인 레더 로직은 그래픽 포로그래밍 언어이다. 내부적으로 릴레이의 개폐가 시뮬레이트 된 것을 간단히 접촉하는 것에 의해 내부적으로 프로그램 된 레더 로직 프로그래밍은 카운터, 타이머, 쉬프트 레지스터, 그리고 계산기와 같은 여러기능을 포함하는데 까지 확장되어 왔다.



기능 블록 다이어그램(FBD) - graphical language for 재사용 가능한 기능 블록에 의한 서술된 시그널과 데이터 플로우용 그래픽 언어. FBD 는 제어시스템 알고리즘과 로직의 상호교호를 나타내는데 매우 유용하다.



구조화된 텍스트(ST): 구조화된 프로그래밍을 조장하는 높은 레벨의 텍스트 언어. 이것은 PASCAL 과 매우 유사한 언어구조(체계 syntax)로 기본 기능과 조작의 광범위한 기능들을 지원한다. 예를 들면,

```

If Speed1 > 100.0 then
    Flow_Rate: = 50.0 + Offset_A1;
Else
    Flow_Rate: = 100.0; Steam: = ON
End_If;

```

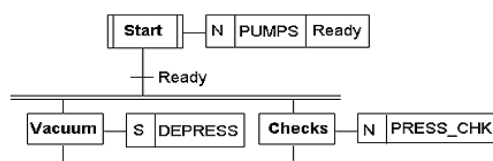
명령 목록(IL): 낮은 수준의 “**assembler like**” 언어는 오늘날 PLC 의 광범위하게 보여지는 유사한 명령 목록 언어에 기초한다.

```

LD R1
MPC RESET
LD PRESS_1
ST MAX_PRESS
RESET: LD 0
ST A_X43

```

순차적 기능 차트(SFC): 보다 고도로 구조화된 수준의 프로그래밍 복합 제어 시스템의 한 방법이다. SFC 프로그램은 기본적으로 구축된 블록이 완전한 프로그램 파일들에서 제어 시스템의 개요이다. 개별 프로그램 파일은 프로그래밍 언어의 다른 타입 중 하나를 사용하는 것을 야기한다. SFC 접근은 폭넓고 복잡한 프로그래밍 과업을 단순하고, 보다 조작 가능한 과업으로 조정한다.



▶ Input/Output 장치로는 무엇이 있나?

INPUTS



– Switches and Pushbuttons

– Sensing Devices: • Limit Switches • Photoelectric Sensors • Proximity Sensors



– Conditon Sensors

– Encoders: • Pressure Switches • Level Switches • Temperature Switches • Vacuum Switches • Float Switches

OUTPUTS



– Valves – Motor Starters – Solenoids – Acuators



– Horns and Alarms – Stack lights – Control Relays – Counter/Totalizer – Pumps –
Printers – Fans