Experiment (3)
Introduction to PLC and Ladder Logic Programming (1)

Introduction

A PLC (Programmable Logic Controller) is an industrial computer used for automation of electromechanical processes, such as control of machinery on factory assembly lines, amusement rides, or light fixtures. PLCs are expected to work flawlessly for years in industrial environments that are hazardous to the very microelectronic components that give modern PLCs their excellent flexibility and precision.

Prior to PLCs, many of these control tasks were solved with contactor or relay controls. This is often referred to as hardwired control. Circuit diagrams had to be designed, electrical components specified and installed, and wiring lists created. Electricians would then wire the components necessary to perform a specific task. If an error was made the wires had to be reconnected correctly. A change in function or system expansion required extensive component changes and rewiring.

Objectives

This experiment aims to:

1- Learn the basics of ladder logic programming.
2- Familiarize the students with SIMATIC S7 software to program Siemens S7-400 PLC.
3- Implement different logic functions using PLC.
4- Understand the function of each Siemens S7-400 PLC modules.

Theory

PLC (Programmable Logic Controller)

Before you start using PLC, it is convenient to know and understand its architecture. See figure 1.

As shown in figure 1, PLC consists of the following parts:

1) POWER SUPPLY: Provides the voltage needed to run the primary PLC components.
2) I/O MODULES: Provides signal conversion and isolation between the internal logic-level signals inside the PLC and the field’s high level signal.
3) PROCESSOR SYSTEM: Provides intelligence to command and govern the activities of the entire PLC systems.
4) PROGRAMMING DEVICE: Used to enter the desired program that will determine the sequence of operation.

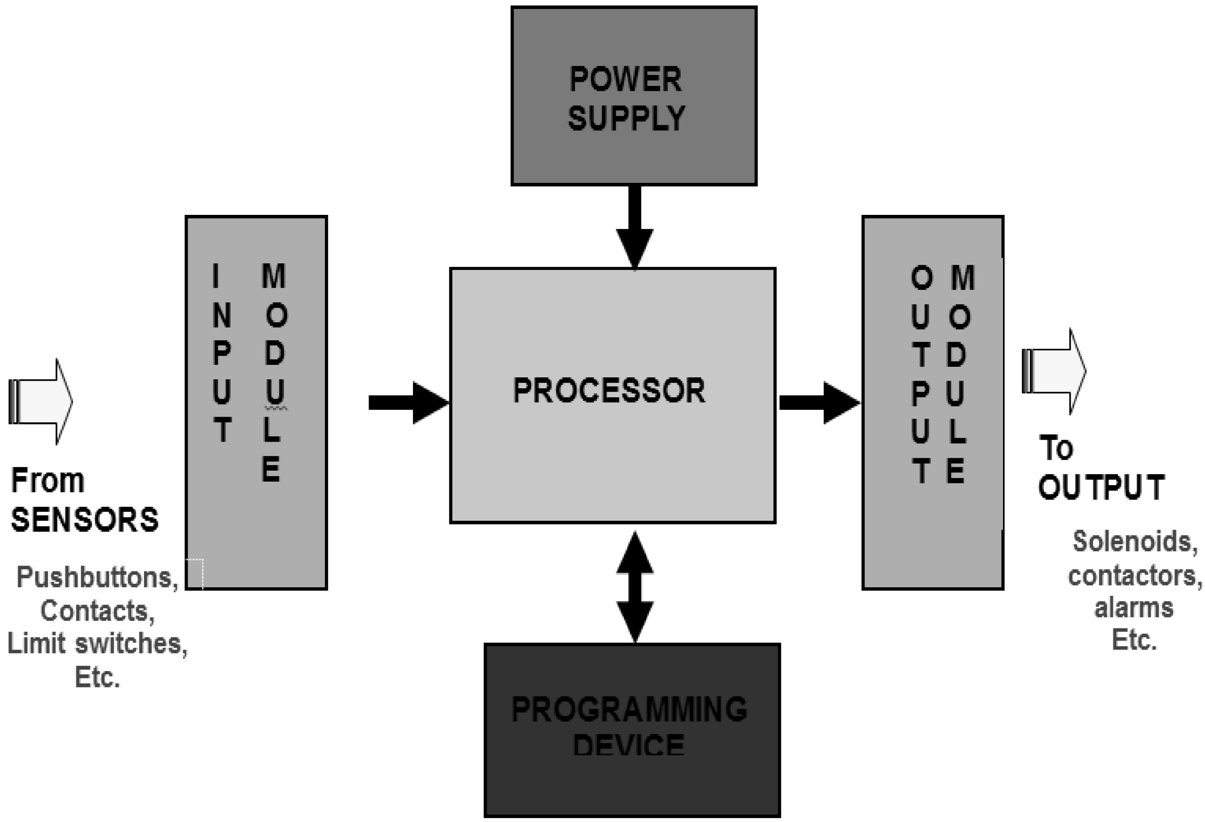


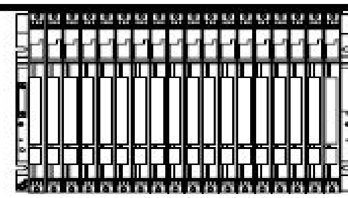
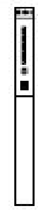
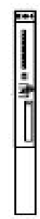
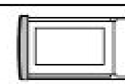
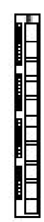
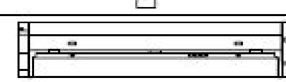
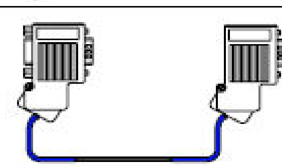
Figure 1: PLC architecture

The following are some advantages of PLC over other microcontrollers:

1) Cost effective for controlling complex systems.
2) Flexible and can be reapplied to control other systems.
3) Computational abilities allow more sophisticated control.
4) Trouble shooting aids making programming easier and reduce downtime.
5) Small physical size, so shorter project time.

In this experiment Siemens S7-400 PLC will be used, table 1 presents the main components of this model.

Table 1: Siemens S7-400 PLC main components

Components	Function	Illustration
Racks (UR: Universal Rack) (CR: Central Rack) (ER: Expansion Rack)	... provide the mechanical and electrical connections between the S7-400 modules.	
Power Supply Modules (PS = Power Supply) Accessories: Backup battery	... convert the line voltage (120/230 VAC or 24 VDC) to the 5 VDC and 24 VDC operating voltages required to power the S7-400.	
CPUs Central Processing Units (CPUs)	... execute the user program; communicate via the multipoint interface (MPI) with other CPUs or with a programming device (PG).	
Memory cards	... store the user program and parameters.	
IF 964-DP interface module	... used to connect distributed I/Os via PROFIBUS-DP	
Signal Modules (SM = Signal Module) (digital input modules, digital output modules, analog input modules, analog output modules) Accessories: Front connector with three different terminal systems	... match the different process signal levels to the S7-400. ... form the interface between PLC and process.	
Interface modules (IM = Interface Module) Accessories: Connecting cable Terminator	... interconnect the individual racks of an S7-400.	
Cable ducts	...are used for routing cables and as ventilation.	
PROFIBUS bus cables	...connect CPUs to programming devices.	

You can notice from figure 2 that the common of the digital input module is connected to the ground of the circuit while the common of the digital output module is connected to the power source.

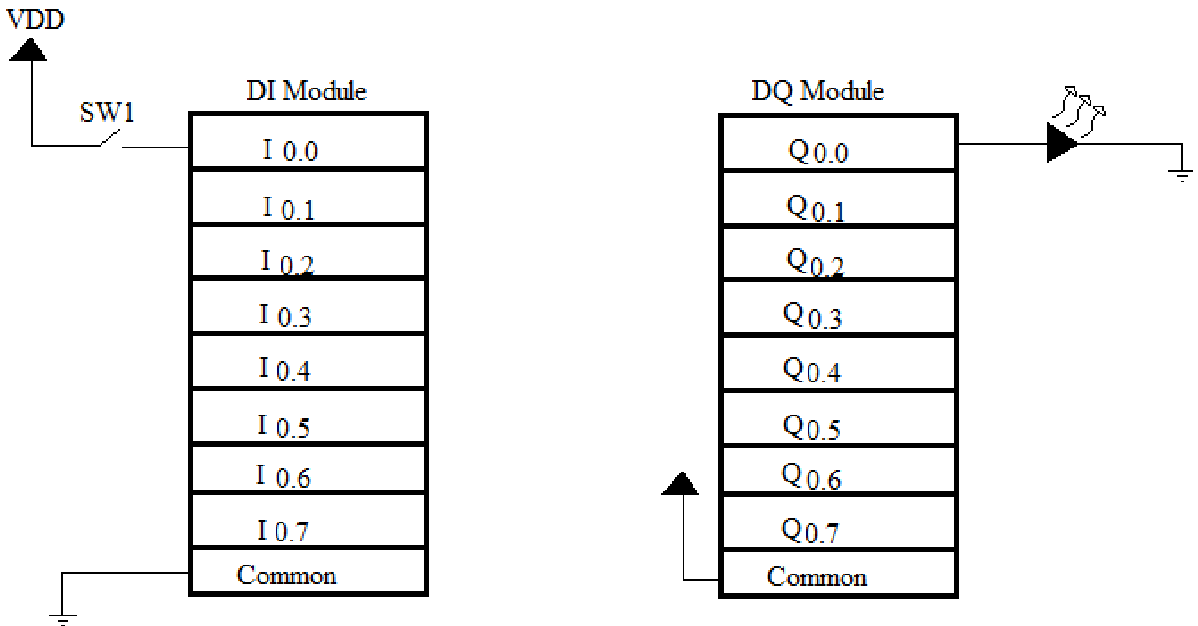


Figure 2: Simple Connection using Siemens S7-400 PLC

The last figure shows a switch connected to the input I0.0 in the digital input module, and an LED connected to the output Q0.0 in the output module.

Ladder Logic Programming

Figure 3 shows electrical continuity, when SW1 is closed, the current will flow from L-1 to L-2 and energize the load.

Figure 3: Hardwire switch-lamp circuit

Even though PLC ladder logic was modeled after the conventional relay ladder, there is no electrical continuity in PLC ladder logic. PLC ladder rungs should have logical continuity in order for the output to be energized. PLC ladder program uses familiar terms like “rungs”, “normally open” and “normally closed” contacts, as illustrated in table 2.

Table 2: Fundamental contacts and coils instructions of PLC ladder logic programming

Symble	Name	Description
—| |—	***Examine if closed***	*It works as normally open switch in a ladder program. If it is ON, the contact will close and allow power (logic) to flow from left to right. If the status is OFF (logical 0), the contact is Open, power (logic) will NOT flow from left to right.*
—|/|—	***Examine if open***	*It works as normally closed switch in a ladder program, and it works exactly opposite to that of the examine if closed.*
—( )—	***Normally open coil***	*This can be used to represent any discrete output from the control logic. When "solved" if the logic to the left of the coil is TRUE, the referenced output is ON (logical 1).*

In a ladder logic program, there is no physical conductor that carries the input signal through to the output. Each rung in the ladder diagram is a program statement. This program statement consists of a condition or sometimes conditions, along with some type of action.

Inputs are the conditions, and the action, or output, is the result of the conditions. As in case of physical wiring hardware devices connected in series or parallel, PLC also combines ladder program instructions in series or parallel. However, rather than working in series or parallel, the PLC combines instructions logically using logic operators like: AND, OR, and NOT. These operators are used to combine the instructions on a PLC rung to make the outcome of each rung either true or false.

1. **AND-logic function:**

   A series circuit of two switches can be regarded as AND logic function. In figure 4, both switches (SW1 **AND** SW2) must be closed to have electrical continuity to energize the output (Light-1). Hence the keyword here is AND.

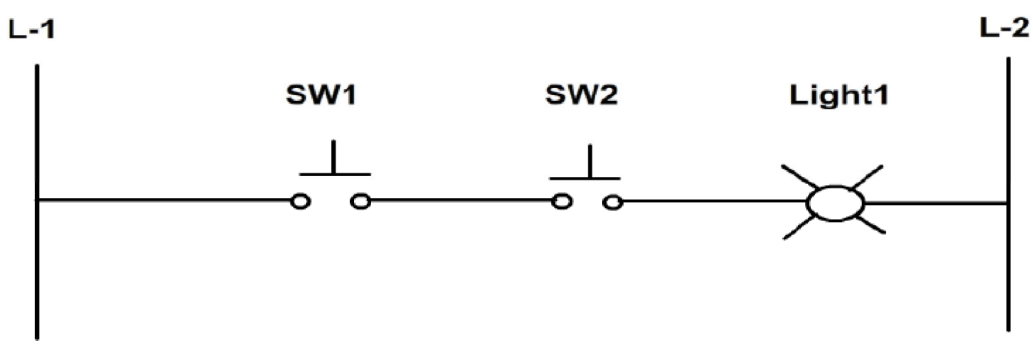


Figure 4: AND-logic function