

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

Symbol	Name	Description
	<i>Examine if closed</i>	<i>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.</i>
	<i>Examine if open</i>	<i>It works as normally closed switch in a ladder program, and it works exactly opposite to that of the examine if closed.</i>
	<i>Normally open coil</i>	<i>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).</i>

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.

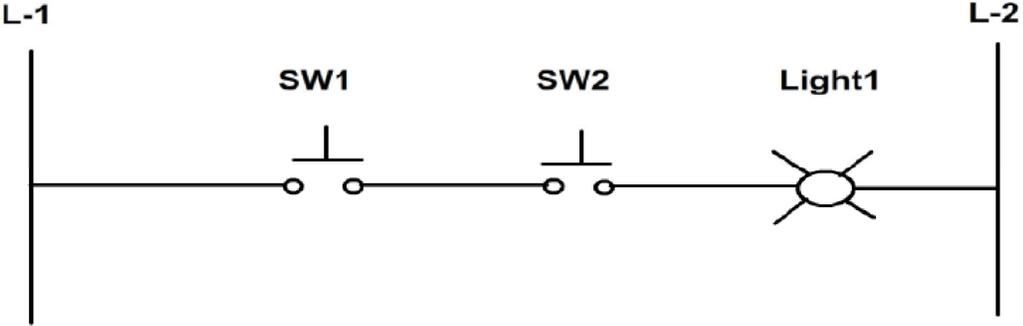


Figure 4: AND-logic function

The circuit shown in figure 5 represents a schematic ladder logic rung for the circuit shown in figure 4. When switch 1 and switch 2 are closed the output coil will be energized.

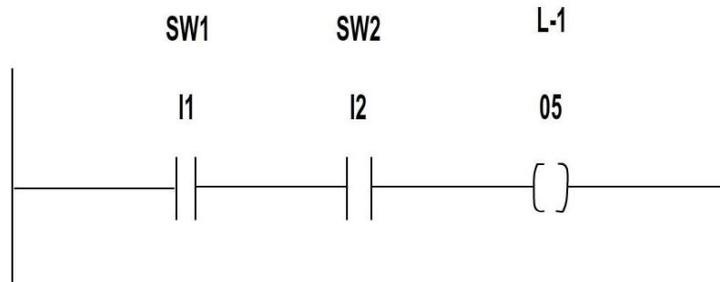


Figure 5: Ladder logic diagram for AND function

**2. OR-logic function:**

A parallel circuit of two switches can be regarded as OR logic function. In figure 6, one of the switches (SW1 **OR** SW2) must be closed to have electrical continuity to energize the output (Light-1). Hence the keyword here is OR.

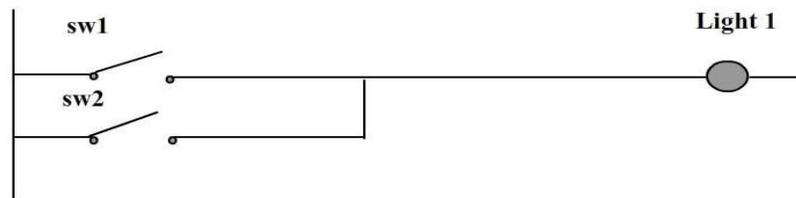


Figure 6: OR-logic function

The circuit shown in figure 7 represents a schematic ladder logic rung for the circuit shown in figure 6. If switch 1 or switch 2 is closed the output coil will be energized.

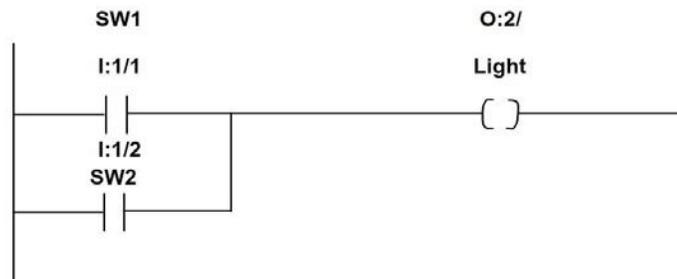


Figure 7: Ladder logic diagram for OR function

**3. The PARALLEL NOT logic function:**

Figure 8 shows ladder diagram for the parallel NOT logic function and its truth table is illustrated in table 3.

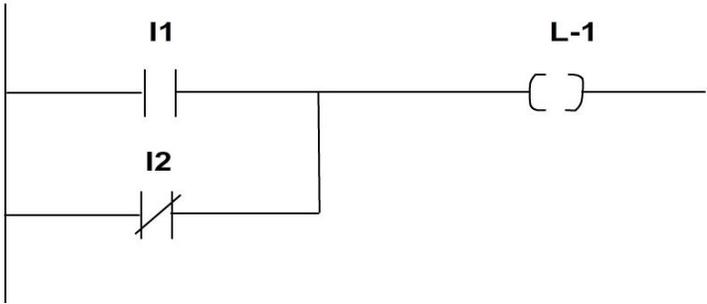


Figure 8: Ladder logic diagram for parallel NOT function

Table 3: Parallel NOT logic function truth table

INPUTS		INPUT STATUS FILE		OUTPUTS
I:1	I:2	I:1	I:2	O:0
0	0	0	1	1
0	1	0	0	0
1	0	1	1	1
1	1	1	0	1

**Equipment**

- 1) Computer with SIMATIC software.
- 2) Siemens S7-400 PLC.
- 3) LEDs.
- 4) Switches.
- 5) Connecting wires.
- 6) Motor.
- 7) Limit switches.

**Procedures**

To write and download a ladder diagram using SIMATIC Manager software, do the following steps:

1. Run SIMATIC Manager software.
2. Start a new project wizard, then click on “Next”.
3. Select the appropriate CPU type, then click on “Next”. You can check the type and order no. from the hardware.
4. Choose “LAD” since we are going to use ladder diagram programming, then click on “Next”.
5. Write a name for your project, then click on “Finish”.

- Click in the left half of the project window on the box containing a plus sign to display the full tree structure of the project. The resulting structure will look something like figure 9. Double click on “SIMATIC 400 Station” then choose “Hardware”.

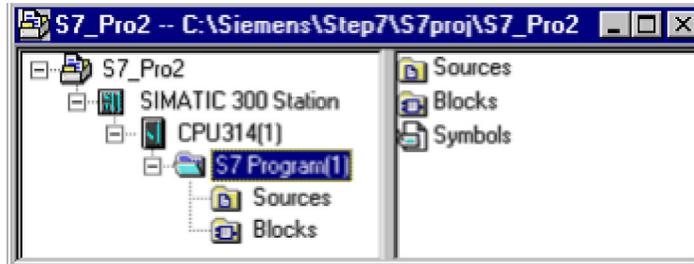


Figure 9: Project window

- Configure the station hardware (Power supply (PS), CPU, Digital Input (DI), ... etc.) according to their order in the racks, see figure 10. You can check the type and order no. from the hardware.

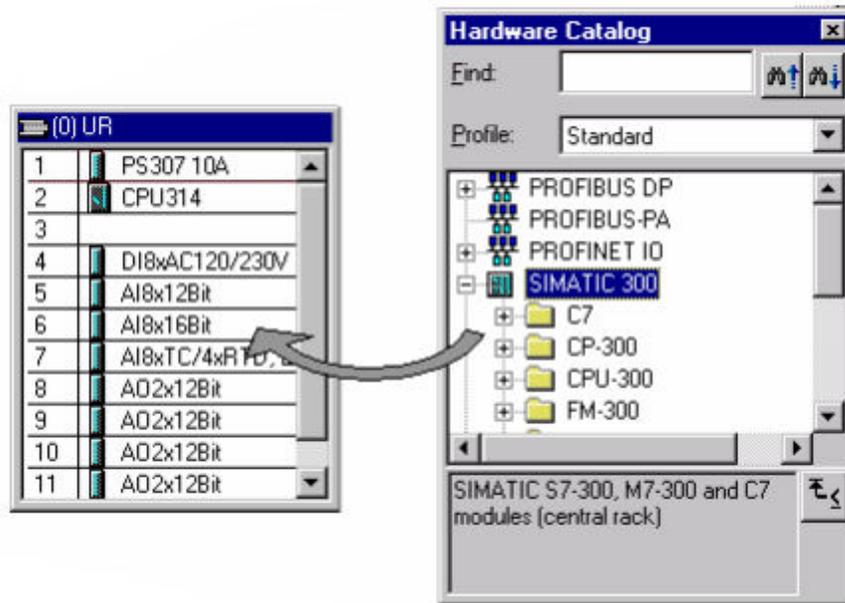


Figure 10: Basic steps for configuring a station

- Save the configurations.
- STOP the PLC station from the switch and download the configurations.
- Click in the left half of the project window on the box containing a plus sign to display the full tree structure of the project, see figure 9.
- Double click on “Blocks” then choose “OB1”, it will open a new window where you have to write your ladder logic diagram.
- After you finish the ladder diagram, download it to the PLC station.

### Part 1: Logic functions implementation

- 1- Connect two switches to the digital input module.
- 2- Connect one LED to the digital output module.
- 3- Write down the ladder diagram to implement AND-logic function using SIMATIC Manager software.
- 4- Download the program to the PLC.
- 5- Repeat steps 1 through 4 to implement OR-logic function.
- 6- Repeat steps 1 through 4 to implement Parallel NOT-logic function.

### Part 2: NOR-logic function implementation

- 1- Write down the Boolean sum-of-product expression for NOR logic function, see table 4.

Table 4: NOR-logic function truth table

Switch 1	Switch 2	Output
0	0	1
1	0	0
0	1	0
1	1	0

- 2- Connect two switches to the digital input module.
- 3- Connect one LED to the digital output module.
- 4- Write down the ladder diagram to implement NOR-logic function using SIMATIC Manager software.
- 5- Download the program to the PLC.

### Part 3: Alarm system implementation using 2 LEDs

Assume that we have 3 signals (Inputs) that we get from the field that warn us about some kind of fire danger. The system works as follows:

Any 1 signal, turn the green light on.

Any 2 signals will turn the orange light on.

All 3 signals, will turn the two lights simultaneously.

- 1- Connect three switches to the digital input module.
- 2- Connect two LEDs to the digital output module.
- 3- Write down the ladder diagram to implement the previous system using SIMATIC Manager software.
- 4- Download the program to the PLC.

**Part 4: The sealed in coil pattern implementation**

This pattern is the basis for remembering state in a PLC. The pattern consists of a trigger condition and a coil, where a contact from the coil branches around the trigger condition to “seal itself in” once the coil is energized, see figure 11.

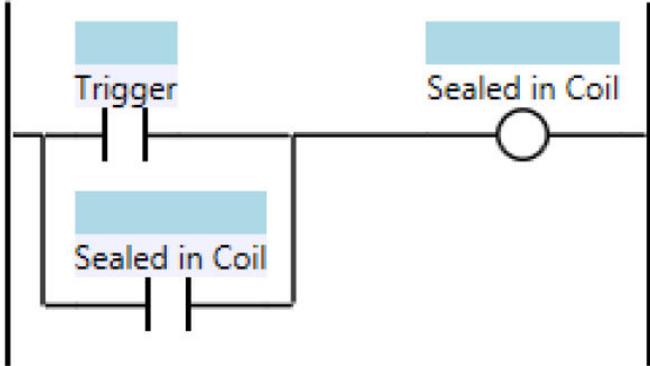


Figure 11: The sealed in coil pattern

- 1- Connect one switch to the digital input.
- 2- Connect one LED to the digital output.
- 3- Write down the ladder diagram to implement this function using SIMATIC Manager software.
- 4- Download the program to the PLC.

**Part 5: The state coil pattern implementation**

This pattern is an extension to the Sealed in Coil pattern. The pattern consists of a trigger condition, a coil, with a contact from the coil branching around the trigger condition to “seal itself in”, and then some other condition to “break the seal”, see figure 12.

- 1- Connect one switch to the digital input.
- 2- Connect one LED to the digital output.
- 3- Write down the ladder diagram to implement this function using SIMATIC Manager software,
- 4- Download the program to the PLC.

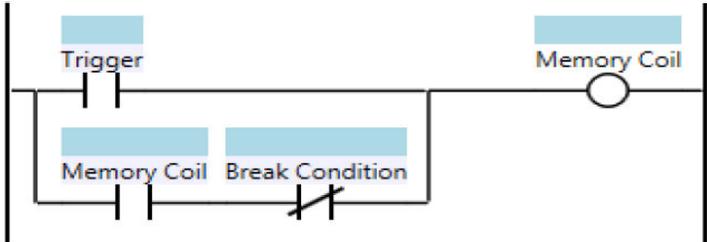


Figure 12: The state coil pattern

### Part 6: The start/stop circuit pattern implementation

A very useful Ladder Logic Programming Pattern is the Start/Stop Circuit. This pattern is an extension to the Sealed in Coil pattern and is similar to the State Coil, see figure 13.

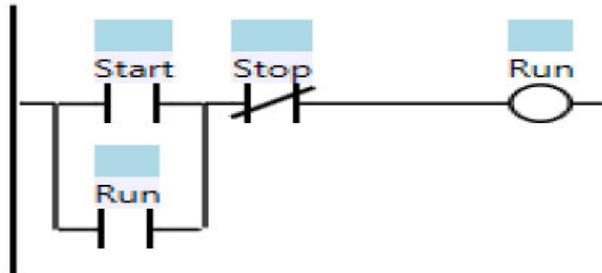


Figure 13: The start/stop circuit pattern

- 1- Connect one switch to the digital input.
- 2- Connect one LED to the digital output.
- 3- Write down the ladder diagram to implement this function using SIMATIC Manager software,
- 4- Download the program to the PLC.

### Part 7: DC motor direction control

- 1- Connect three switches to the digital input, as follows:  
Forward pushbutton, Backward pushbutton, and Overall system switch.
- 2- Connect forward motor and backward motor connections to the digital output.
- 3- Write a ladder diagram to control the direction of the motor using the pushbuttons.

#### Hints:

- 1) Use the idea of an interlock (electrical and mechanical); to prevent the situation of pressing the backward and forward pushbuttons at the same time, and stop the motor when it reaches its maximum limits.
- 2) Use the idea of state coil pattern since we are using pushbuttons to control the direction of motion.

### Discussion and Conclusions

- 1- Discuss the advantages of PLC controllers in comparison with general purpose computers.
- 2- Illustrate the operation of sealed in coil pattern? How it can be de-energized?
- 3- Illustrate the operation of state coil pattern? How it can be de-energized?
- 4- Illustrate the operation of start/stop circuit pattern?
- 5- What is the main difference between state coil pattern and start/stop circuit pattern?
- 6- Mention four different applications where the PLC can be used?
- 7- Illustrate the mechanical and electrical interlocks?