

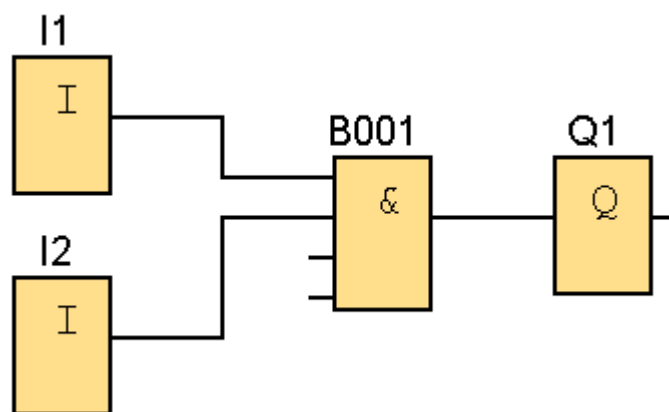


ثانوية التكنولوجيا التطبيقية  
Applied Technology High School

---

# PLC Fundamentals

## Module 3: Programming with Function Blocks



---

PREPARED BY

**Academic Services Unit**

August 2011

© Applied Technology High Schools, 2011

# Module 3: Programming with Function Blocks

## Module Objectives

Upon successful completion of this module, students should be able to:

1. Gain an overview of the important basic PLC concepts and functions in control technology.
2. Demonstrate how different logic functions work, such as the AND, OR and NOT.
3. Represent logic function operations in truth tables.
4. Use the timing and counting functions in programs.
5. Gain an overview of the most important functions of LOGO!Soft.
6. Create and test control programs using function blocks.
7. Demonstrate how to transfer programs from the PC to the LOGO! controller.

## Module Contents

	<b>Topic</b>	<b>Page No.</b>
3.1	Boolean Equations	3
3.2	Logic Design	5
3.3	Programming Methods	6
3.4	LOGO! Soft Overview	7
3.5	Memory Function	10
3.6	Timing Function	10
3.7	Counting Function	11
3.8	Summary of Basic Control Functions	11
3.9	Lab Activity 1	13
3.10	Lab Activity 2	15
3.11	Lab Activity 3	19
3.12	Review Exercise	20

### 3.1 Boolean Equations

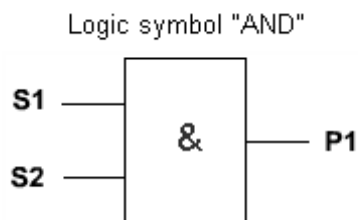
**Boolean equations** consist of variables and operations and look very similar to normal algebraic equations. The three basic operators are AND, OR and NOT.

**Truth table** is a simple method for showing all the possible combinations that will turn an output ON or OFF.

#### A. Logic Functions

PLC Systems execute logic functions. Therefore, understanding logic functions of binary numbers is important for PLC programming. The basic logic functions include AND, OR, NOT, NAND, NOR & EXOR.

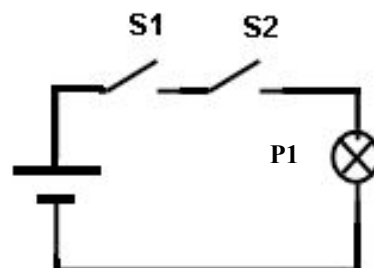
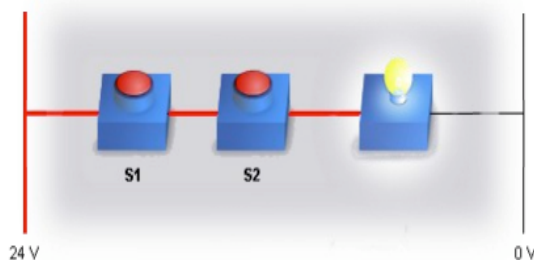
1. **AND Function:** AND function has two or more inputs and only one output. It produces a HIGH output only when all its inputs are HIGH.



**Truth Table**

S1	S2	P1
0	0	0
0	1	0
1	0	0
1	1	1

**Boolean Expression:  $P1 = S1.S2$**



*Lamp P1 is ON, only if S1 and S2 are closed or ON.*

Figure 3.1: Switch Equivalent of AND function

2. **OR Function:** OR function has two or more inputs and only one output. It produces a HIGH output only if one or more of the inputs are HIGH.

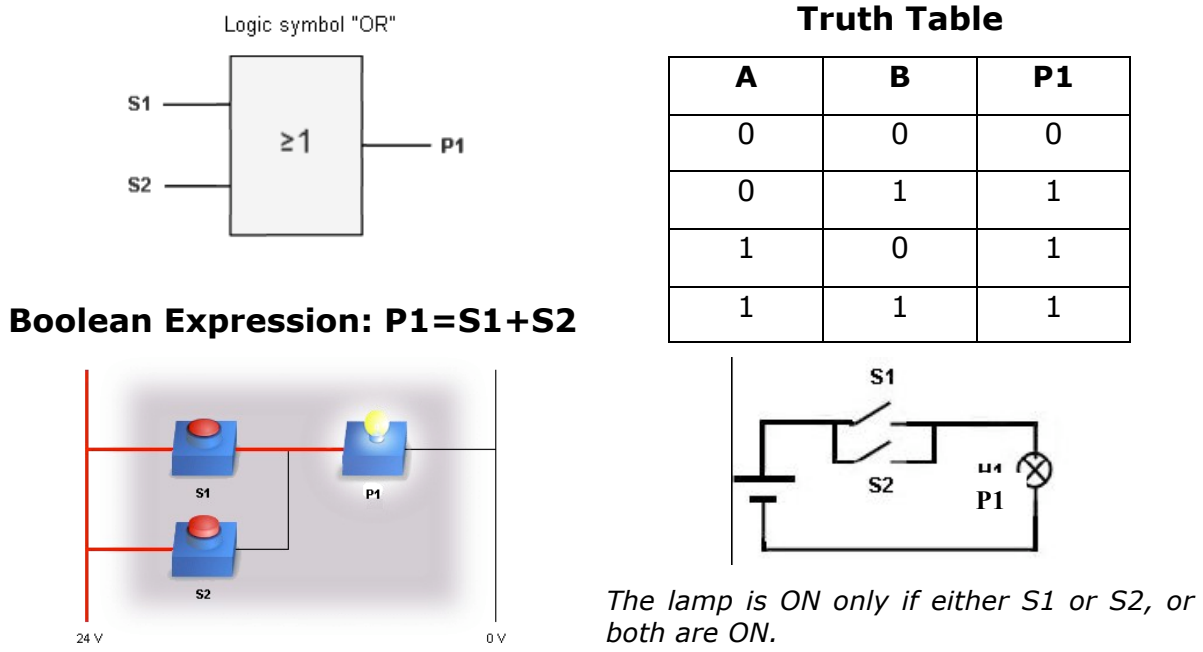


Figure 3.2: Switch Equivalent of OR function

3. **NOT Function:** A NOT function has only one input and one output. It produces an output that is opposite to the input.

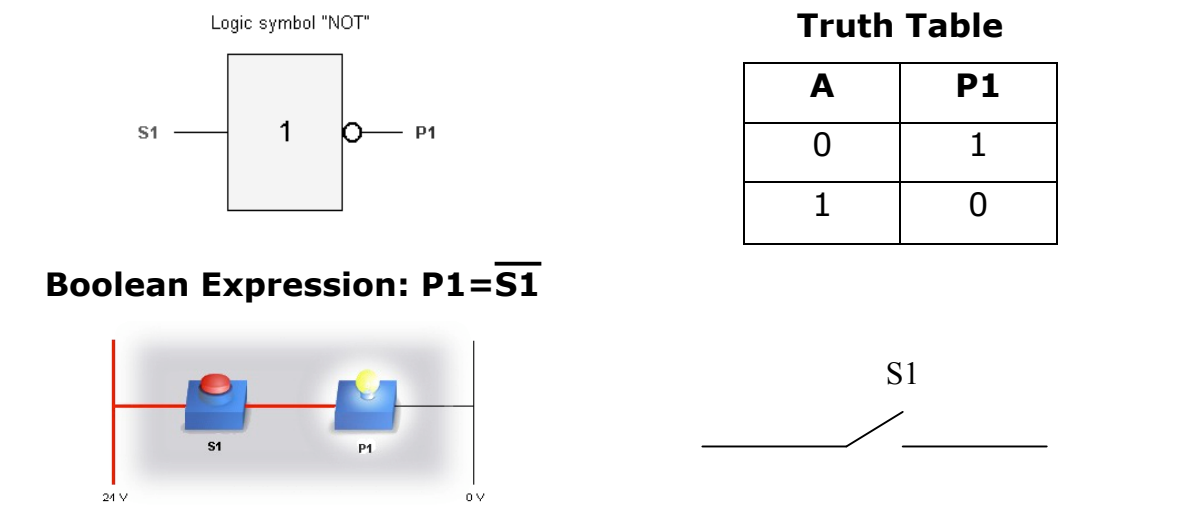


Figure 3.3: Switch Equivalent of NOT function

### 3.2 Logic Design

To design logic circuits for a particular application it is important to describe how a controller works. In order to describe the actual condition for a controller to work, it is usually expressed in terms of a Boolean equation. In this case it is written out in the form of a sentence first, and then converted to a Boolean expression. The Boolean expression may then be converted to a desired form: either the Function Block Diagram (FBD) or the Ladder Diagram (LAD).

#### **Example:**

A machine can be only switched ON if its door is closed (this can be sensed using a limit switch) and two pushbuttons are pressed. Write down the Boolean expression for this problem.

To find the Boolean expression we need first to determine the inputs and outputs:

<i>Inputs</i>	Limit switch (L)	Pushbutton1 (P1)	Pushbutton2 (P2)
<i>Outputs</i>	Machine		

Then we need to find the relation between the inputs, what is the condition to turn ON the output?

The machine will turn on when L is pressed **AND** P1 is pressed **AND** P2 is pressed.

$$M=L.P1.P2$$

Boolean expressions can be written to describe any condition, and it makes it easier to build logic circuits.

### 3.3 Programming Methods

A program is a set of instructions to do a task. The following methods could be used for PLC programming:

- Function Block Diagram
- Ladder Logic
- Statement Lists

In this module, function blocks will be used to develop circuits.

#### Function Block Diagram

A block in LOGO! is a function which converts the input information to output information. Figure 3.4 shows an example of a function block diagram. Here S1 & S2 are the terminals that are connected to the AND function block. The output H1 will be ON only if both the switches S1 and S2 are ON.

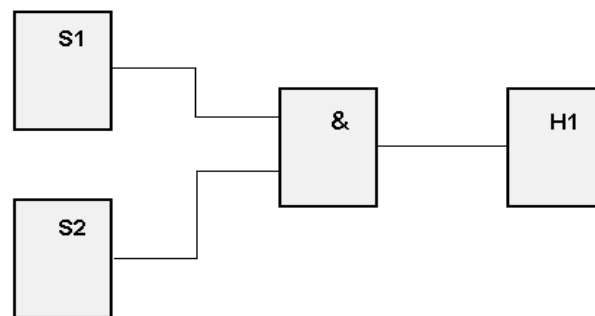


Figure 3.4: Function Block Diagram Illustration

LOGO! provides various elements in programming mode.

- Co: Connector List
- GF: List of basic functions AND, OR, and so on
- SF: List of special functions

### 3.4 LOGO! Soft Overview

LOGO!Soft Comfort is a programming software for the LOGO! Using LOGO!Soft, you can create control programs and transfer them to LOGO!

The LOGO!Soft user interface has the following function areas as shown in figure 3.5 below:

1. Menu bar
2. Standard tool bar
3. Programming Interface
4. Info box
5. Status Bar
6. Toolbox
7. Constants, connectors, functions

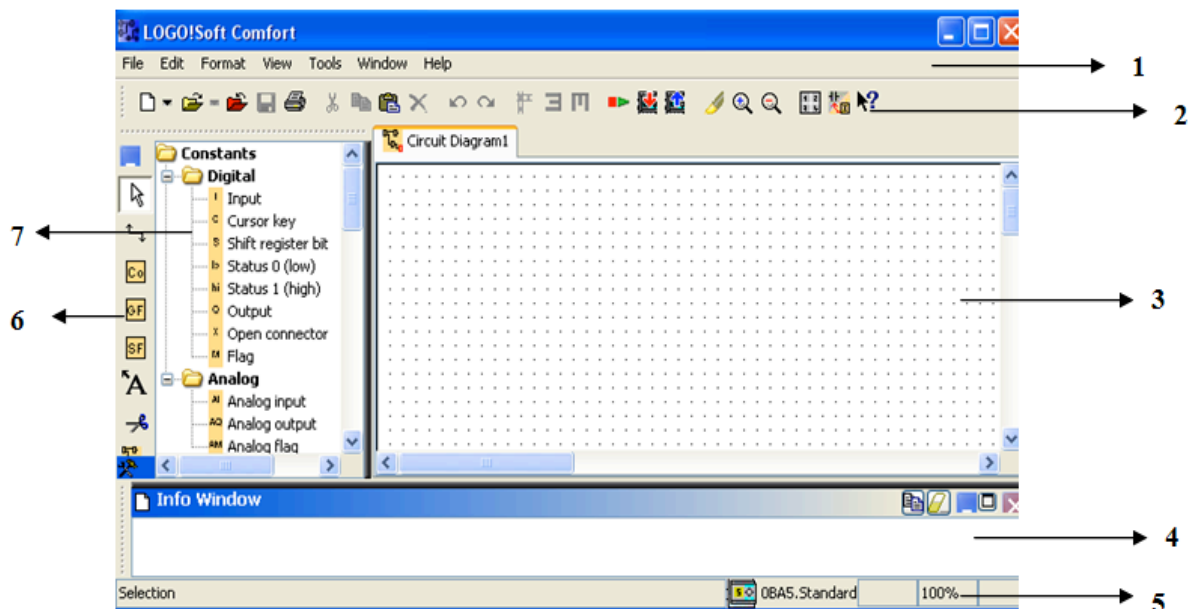


Figure 3.5: LOGO!Soft User Interface





The following rules must be followed while connecting the blocks:

- Connection can be made between one block output and one block input.
- An output can be connected to several inputs.
- An input cannot be connected to several outputs.

### Program Simulation

The program can be tested using simulation. A program is tested by specifying the input signals and observing desired output signal as shown in figure 3.8. The red lines carry a 1-signal and the blue lines carry a 0-signal.

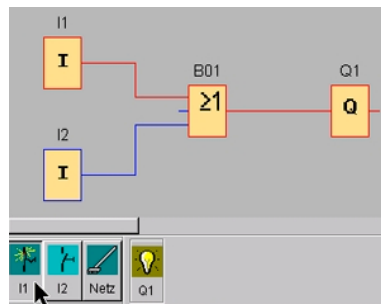
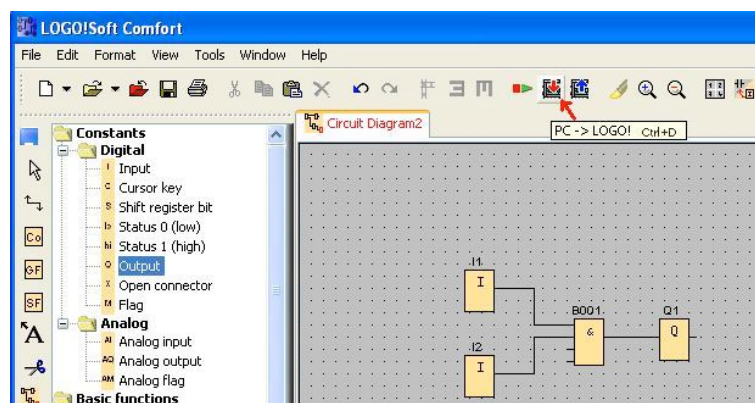


Figure 3.8: Program Simulation demo

### Program Transfer to LOGO!

The program can be downloaded from the PC to LOGO! and then tested.



Start the program on LOGO!

- ESC key
- Select "START"
- OK key

Figure 3.9: Program transfer to LOGO!

### Conduct Lab Activity 1

### 3.5 Memory function

Memory function is used to store a signal. It has SET (S) and RESET (R) inputs as shown in figure 3.10. A high at the set input sets (stores 1) the memory function, and a high at the reset input resets the memory function as shown in figure 3.11.

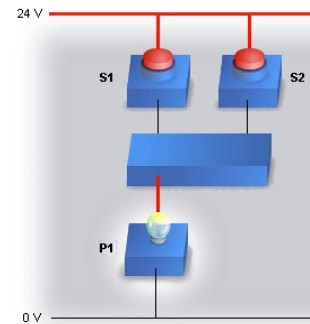
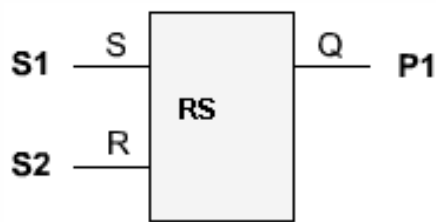


Figure 3.10: Memory function block

Figure 3.11: Memory function SET

### 3.6 Timing function

This function is used to provide a time-delay in a circuit. In the example shown in figure 3.12, a switch-off delay is used to delay the off-time. The lamp P1 will be off after 5 seconds.

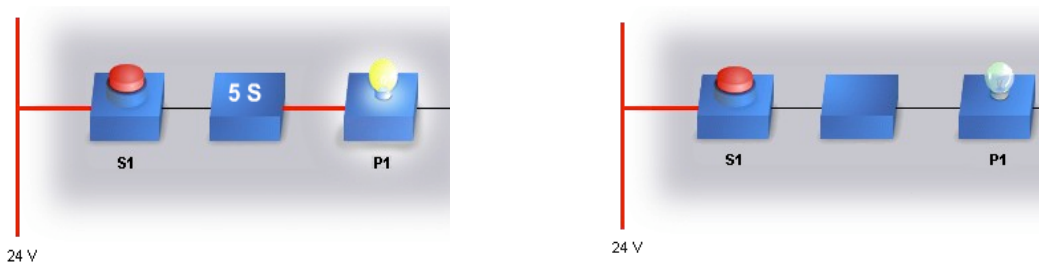


Figure 3.12: Timing function demonstration

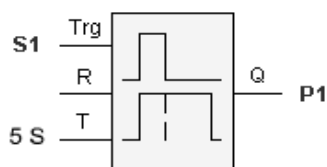


Figure 3.13: Timing Function Block

Figure 3.13 shows the timing function block. T input is used to set the time. The Q output will be high for the time set at the T input.

**There are different types of timers; some examples are given below:**

- 1- OFF Delay timer: It is used when there is a need to stop an action after a certain time.
- 2- ON Delay timer: It is used to start an action after a certain time. For example, some machines do not require running immediately when switched on, but after a certain time, either for safety or for lubrication.

### 3.7 Counting function

Counting functions are used to detect the number of items and events. For example, a counter may be required to count exactly 10 identical parts that need to be fed onto a conveyor of the sorting system.

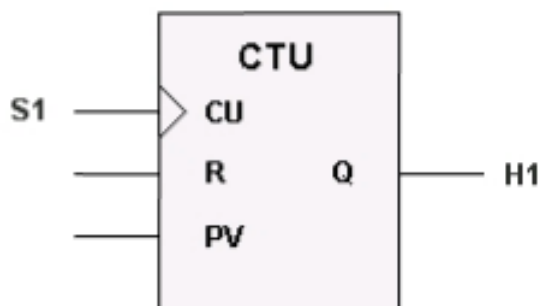


Figure 3.14:Counter function block

Figure 3.14 shows the counter function block. PV is the Preset Value. The counter value increases by 1 for each input signal at CU. When the preset value is reached, the output Q becomes '1'.

### 3.8 Summary of Basic Control Functions

The Basic Control Functions include the following:

- A. Logic functions (AND,OR,NOT)
- B. Memory functions (Latching)
- C. Timing Functions &
- D. Counting Functions.

**Skill 1:** To create function block diagrams from logic expressions.

Draw FBDs for the following logic expressions:

A.  $Q1 = I1 + I2 + I3.I4$

B.  $Q8 = I1.I2 + I3.\overline{I4}$

**Skill 2:** To create function block diagrams from statement problems.

Design a simple house alarm control circuit.



- The house has two doors D1 & D2, and two windows W1 & W2.
- If any one of the doors OR any one of the windows is opened by the thief, the alarm should sound.

Draw a function block diagram for this logic operation.

Fig 3.16 Connecting IO Simulator to LOGO!

### 3.9 Lab Activity 1

**Objective:** To test the basic logic functions using the IO Simulator.

The IO Simulator shown in fig 3.15 could be used to interact with the LOGO! in order to simulate the input and output signals.

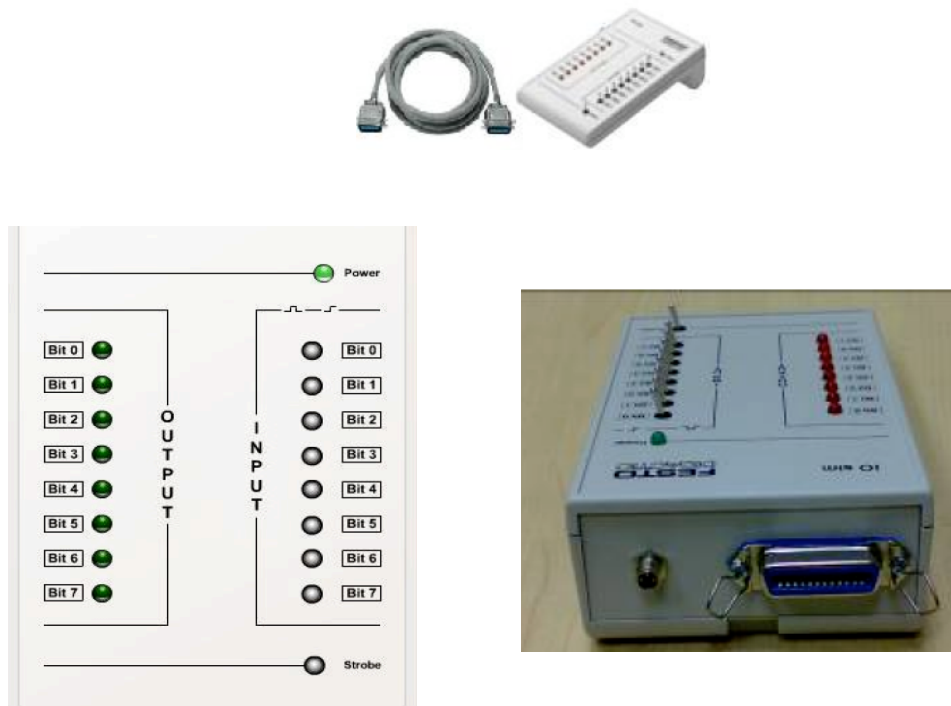


Figure 3.15 : IO Simulator

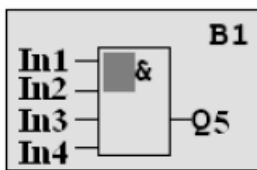
**Procedure:**

- Connect the IO Simulator to the LOGO! as shown in fig 3.16



- Use the on-board keys on the LOGO! Control unit and program the LOGO! for the AND, OR and NOT operations with the function blocks:
- Switch on the strobe switch and test the different logic functions.
- Write the inputs and outputs in each of the truth tables given below.  
(Refer to the IO Map in Figure 3.17 for input/output details)

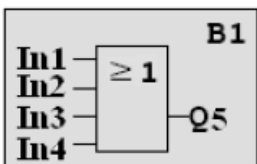
**AND**



**Truth Table**

I9	I10	Q5

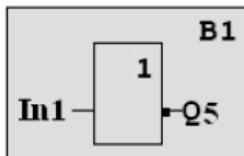
**OR**



**Truth Table**

I11	I12	Q6

**NOT**



**Truth Table**

I11	Q7

<i>IO Sim Input</i>	<i>LOGO! Input</i>
Bit 0	I9
Bit 1	I10
Bit 2	I11
Bit 3	I12
Bit 4	I13
Bit 5	I14
Bit 6	I15
Bit 7	I16

<i>IO Sim Output</i>	<i>LOGO! Output</i>
Bit 0	Q5
Bit 1	Q6
Bit 2	Q7
Bit 3	Q8
Bit 4	Q9
Bit 5	Q10
Bit 6	Q11
Bit 7	Q12

Figure 3.17: IO Map for LOGO! & IO Simulator

## 3.10 Lab Activity 2

**Objective:** To familiarize with the logic function blocks.

### Part A: AND Function Block

**Procedure:** Program the PLC for the following operation:

- The table must move forward when the selector switch is ON.
- The table must move backward when the green pushbutton and the white pushbutton are pressed.

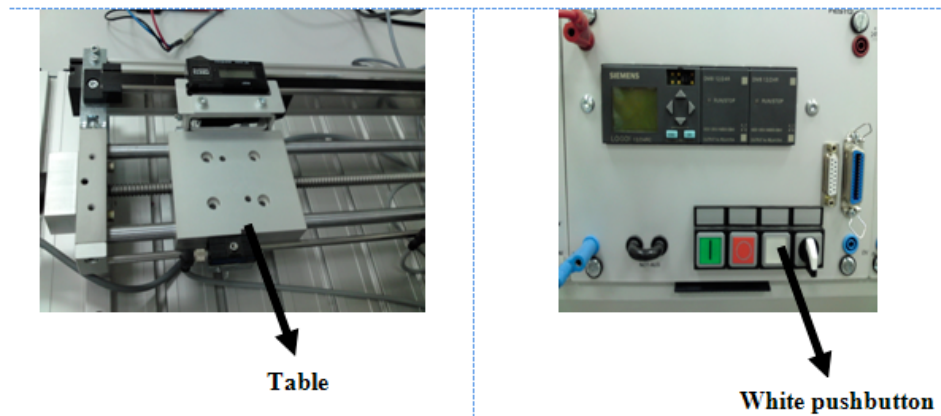


Figure 3.18: Table & White pushbutton

Function Block Diagram:

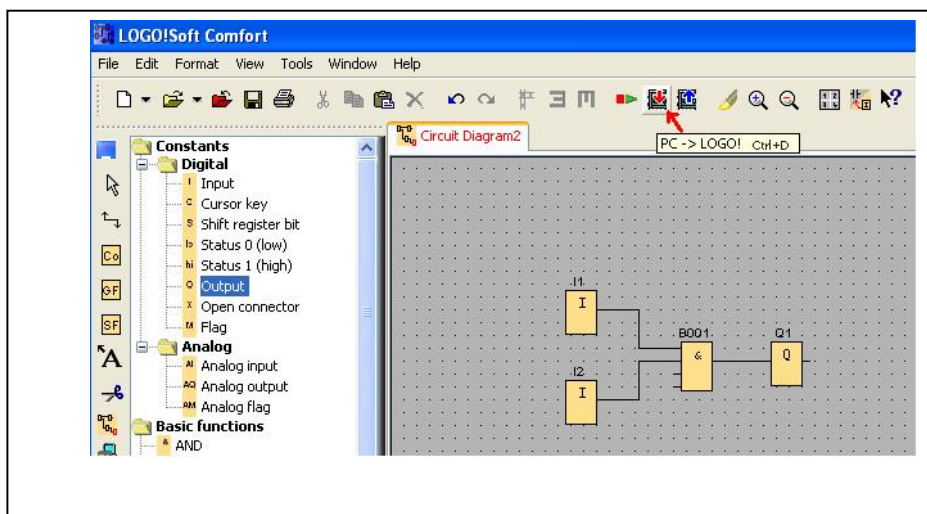


Figure 3.19: FBD

**Step 2:**

Use LOGO!Soft software to create the FBD. Clear the previous program on the PLC. Download the program from the PC to the PLC, run and observe the result.

**Step 3:**

Enter the control elements and the PLC Input/Output in the table below:

<b>Control Elements</b>	<b>PLC Input/Output</b>



**Part B: NOT & OR Function Blocks**

**Procedure:** Program the PLC for the following operation:

- The conveyor belt motor must start when the green pushbutton OR the white pushbutton is ON, OR when both are ON.
- The conveyor belt motor must stop when the inductive sensor senses a metal.
- The branching arm must extend when the inductive sensor senses the metal.

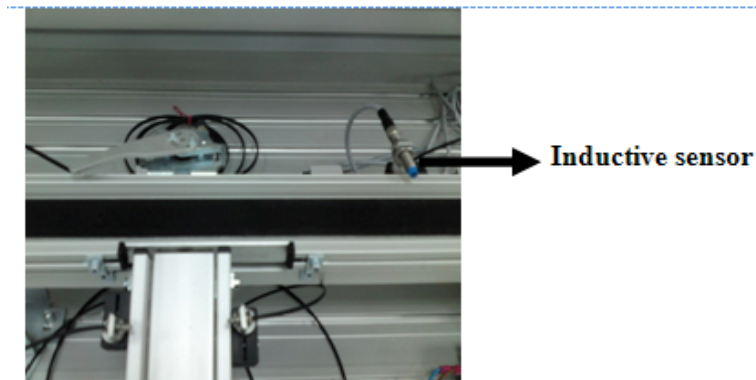


Figure 3.20: Inductive Sensor

**Step 1:**

Use the on-board keys on the LOGO! Control Unit to program the PLC. Run the program and observe the result. Draw the FBD in the box below:

Function Block Diagram:

**Step 2:**

Use LOGO!Soft software to create the FBD. Clear the previous program on the PLC. Download the program from the PC to the PLC, run and observe the result.

**Step 3:**

Enter the control elements and the PLC Input/Output in the table below:

<b>Control Elements</b>	<b>PLC Input/Output</b>

### 3.11 Lab Activity 3

**Objective:** To familiarize with the timing and counting function blocks.

**Procedure:** Program the PLC for the following operation:

- The green light must switch ON once the green pushbutton is pressed, and must remain ON.
- 5 seconds after pressing the green pushbutton, the white light must switch ON.

**Step 1:**

Use the on-board keys on the LOGO! Control Unit to program the PLC. Run the program and observe the result. Draw the FBD in the box below:

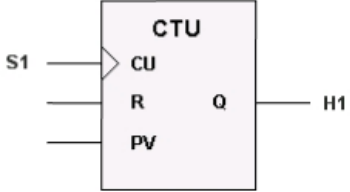
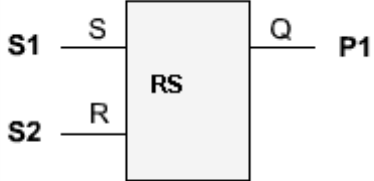
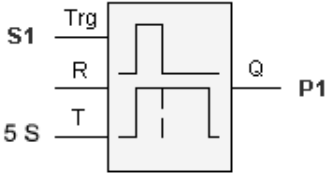
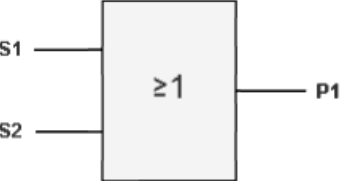
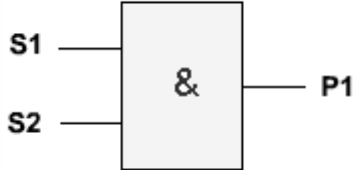
Function Block Diagram:

**Step 2:**

Use LOGO!Soft software to create the FBD. Clear the previous program on the PLC. Download the program from the PC to the PLC, run and observe the result.

### 3.12 Review Exercise

1. Match the following tasks with the proper function blocks:

	Task		Function Blocks
1	Turn ON a machine when the PB is pressed and the machine door is closed		 <p>A Counter Up (CTU) function block with inputs S1 (Set), R (Reset), and PV (Present Value). The output is Q, which is connected to H1.</p>
2	Count the number of produced Cola cans		 <p>An RS (Reset Set) function block with inputs S1 (Set) and S2 (Reset). The output is Q, which is connected to P1.</p>
3	Switch OFF a machine after a certain time		 <p>A Timer On-Delay (TON) function block with inputs S1 (Trigger), R (Reset), and T (Time delay, set to 5 S). The output is Q, which is connected to P1.</p>
4	Turn ON and Turn OFF a motor using the same function block		 <p>An OR function block with inputs S1 and S2. The output is P1.</p>
5	Turn ON a lamp using either PB1 or PB2 or both		 <p>An AND function block with inputs S1 and S2. The output is P1.</p>

2. For the following tasks *write the correct expressions and draw the Function Block Diagrams*:

a) A pump (P) turns ON when switch (S1) is pressed and it turns OFF when (S2) is pressed

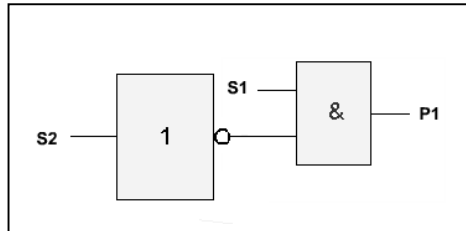
b) A light (L1) turns ON when switch (S1) is pressed and switch (S2) is NOT pressed (released)

c) A green light (Q1) will turn on when both pushbutton (I1) and pushbutton (I2) are pressed **OR** pushbutton (I3) is pressed.

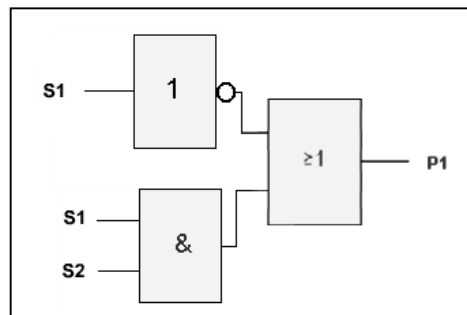
d) A machine (Q1) starts 20 seconds after pressing the switch S1

3. For the following FBDs write the correct expression:

a)



b)



c)

