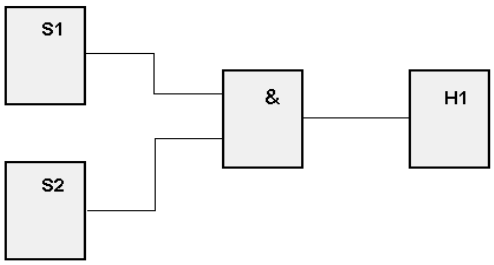
معهد التكنولوجيا التطبيقية

INSTITUTE OF APPLIED TECHNOLOGY

PLC

Module 2:
Programming with Function Blocks



PREPARED BY

IAT Curriculum Unit

Jan 2010

Module 2:
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:

2.1 PLC Basics

A Programmable Logic Controller is a device that can be programmed to perform control functions. Since it is a digital device, it stores information in the form of ON/OFF conditions referred to as binary digits or bits. Even though the PLC uses both digital and analog signals, the CPU can understand only digital signals.

Fig 2.1 shows the relation between the logic level and the switch condition. **Logic 1** indicates that a signal is present and the **switch is ON**. **Logic 0** indicates that the signal is absent or the **switch is OFF**.

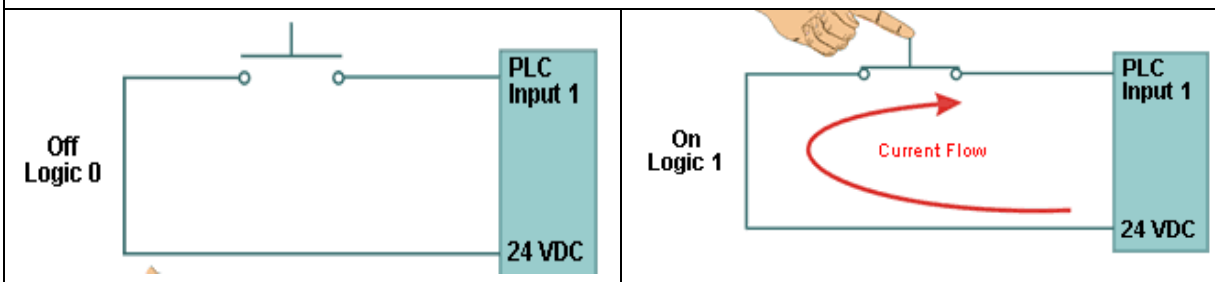


Fig 2.1: Normally Open Pushbutton demo

A normally open (NO) pushbutton is used in the above example. When the switch is not pressed, no voltage is present at the PLC Input 1 and sets it to OFF state. When the switch is pressed, 24V dc is applied to the PLC input and sets it to ON state. A normally closed (NC) pushbutton acts opposite to the normally open (NO) pushbutton. Fig 2.2 shows the pushbutton symbols.

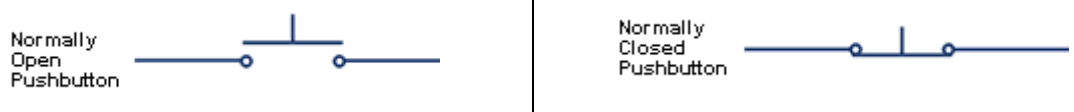


Fig 2.2: Pushbutton Symbols

Conduct Lab Activity 1

Basic Control Functions

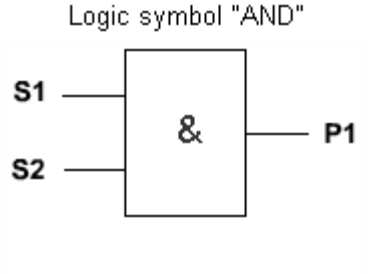
The Basic Control Functions include the following:

A. Logic functions
B. Memory functions
C. Timing Functions &
D. Counting Functions.

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.

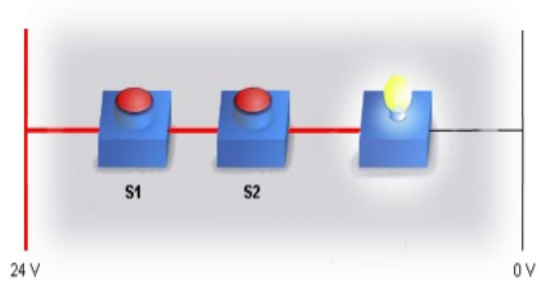
1. **AND Function:** AND function has two or more inputs and only one output. It produces a HIGH output only when all its inputs are HIGH.

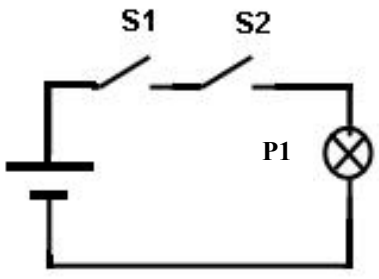


Boolean Expression: P1=S1.S2

Truth Table

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

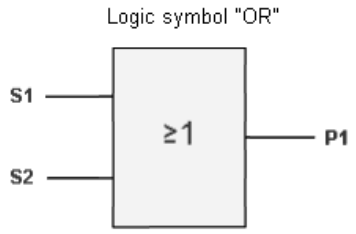




Lamp P1 is ON, only if S1 and S2 are closed or ON.

Fig 2.3: Switch Equivalent of AND function

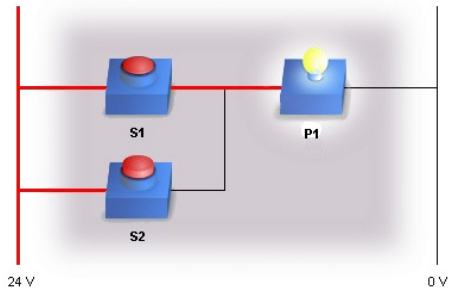
2. **OR Function:** OR function has two or more inputs and only one output. It produces a HIGH output only if one or more of the inputs are HIGH.

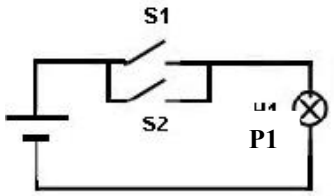


Boolean Expression: P1=S1+S2

Truth Table

A	B	P1
0	0	0
0	1	1
1	0	1
1	1	1

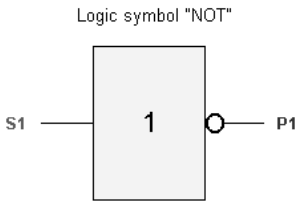




The lamp is ON only if either S1 or S2, or both are ON.

Fig 2.4: Switch Equivalent of OR function

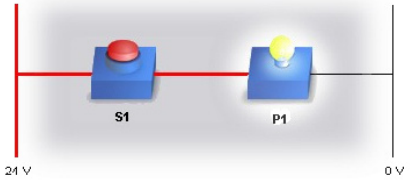
3. **NOT Function:** A NOT function has only one input and one output. It produces an output that is opposite to the input.



Boolean Expression: P1=$\overline{S1}$

Truth Table

A	P1
0	1
1	0



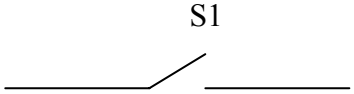


Fig 2.5: Switch Equivalent of NOT function

Conduct Lab Activity 2

B. Memory function: Memory function is used to store a signal. It has SET (S) and RESET (R) inputs as shown in fig 2.6. 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 fig 2.7.

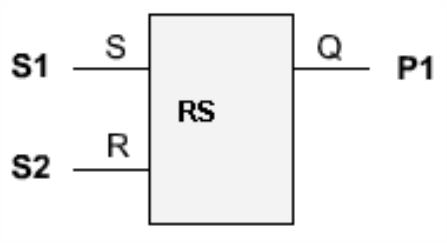


Fig 2.6: Memory function block

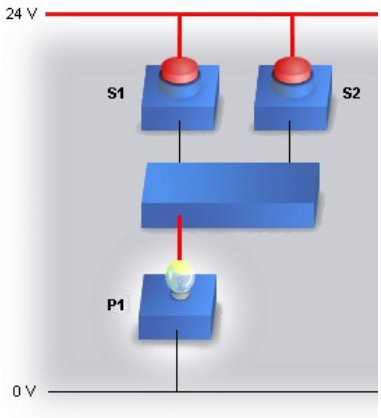
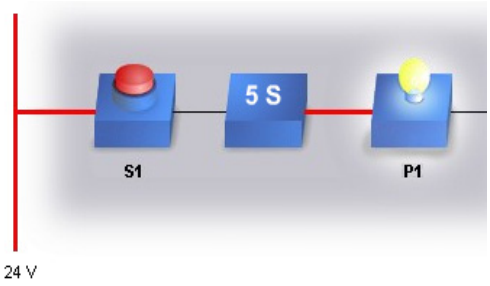


Fig 2.7: Memory function SET

C. **Timing function:** This function is used to provide a time-delay in a circuit. In the example shown in fig 2.8, a switch-off delay is used to delay the off-time. The lamp P1 will be off after 5 seconds.



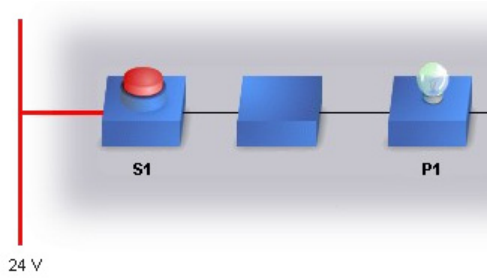


Fig 2.8: Timing function demonstration

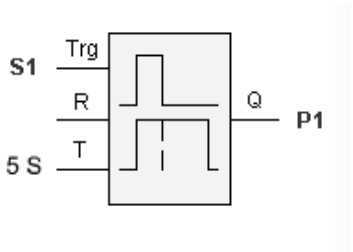


Fig 2.9: Timing Function Block

Fig 2.9 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. Refer to the example in fig 2.8.

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 certain time, either for safety or for lubrication.

D. 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.

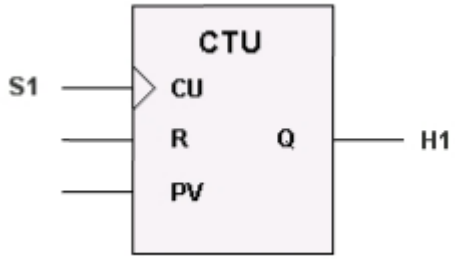


Fig 2.11:Counter function block

Fig 2.11 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'.

Conduct Lab Activity 2

2.2 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. Fig 2.13 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.

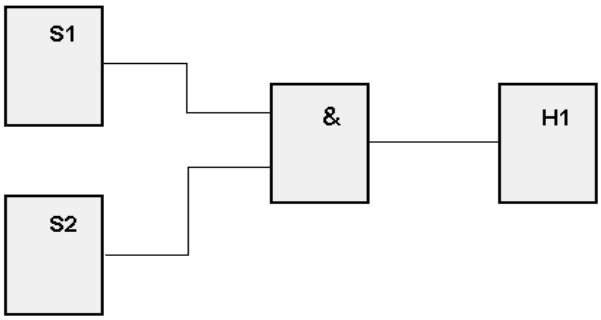


Fig 2.13: Function Block Diagram Illustration

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

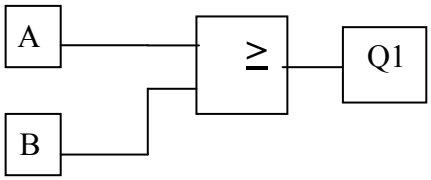
The logic function blocks are available under **GF** (General Function) category, and the terminals can be selected by choosing the **Co** function. Co is named after the term Connector (terminal).

Skill 1: To create function block diagrams from statement problems.

Follow Example A and draw the FBD for the logic operation in Example B :

A. A staircase light is controlled by two switches (A and B). The light must turn ON if ANY ONE of the switches, OR BOTH the switches are ON.

Draw a function block diagram to implement this logic operation.



B. 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.

Skill 2: To create function block diagrams from logic expressions.
Draw FBDs for the following logic expressions:
A. $Q1 = I1 + \overline{I2} + I3.I4$
B. $Q8 = I1.I2 + I3.\overline{I4}$
Conduct Lab Activities 3 & 4.

2.3 LOGO!Soft Overview

LOGO!Soft 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 fig 2.13 below:

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

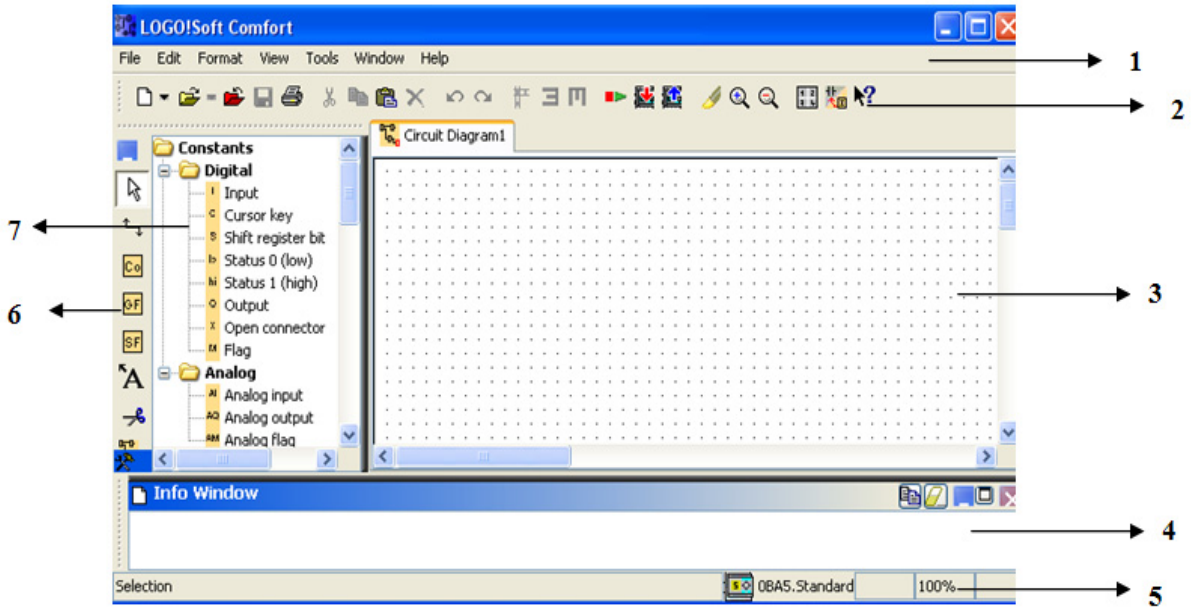


Fig 2.14: LOGO!Soft User Interface

The Co, GF and SF tabs (fig 2.15) could be used to select the terminals and function blocks.

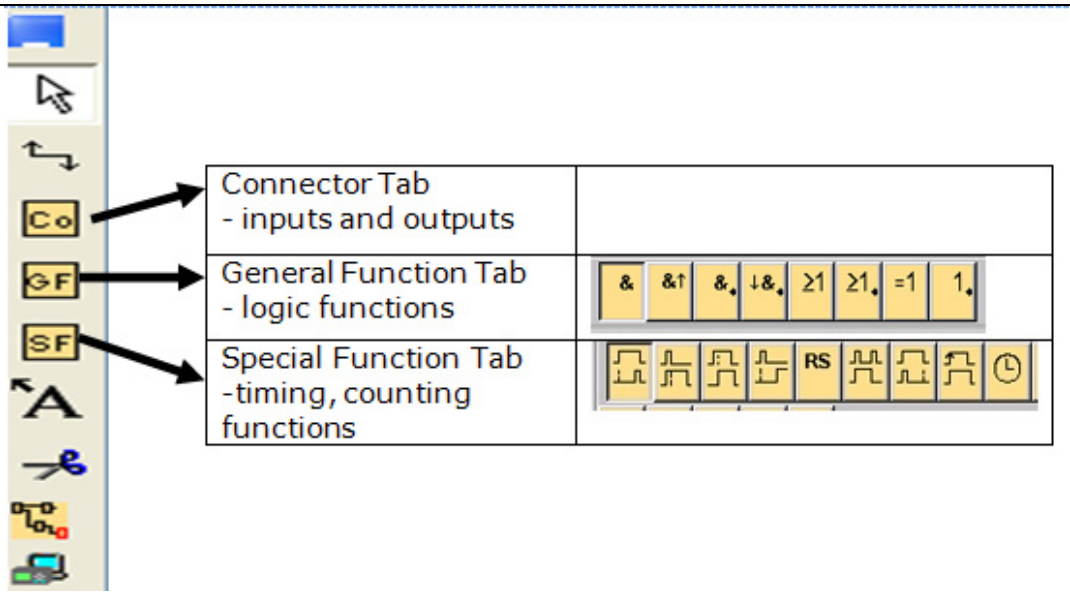


Fig 2.15: Toolbox tabs

Creation of a control program using Function Blocks

The following program demonstrates the OR function. The input and output terminals and the OR function are used to create the function block diagram as shown in fig 2.16.

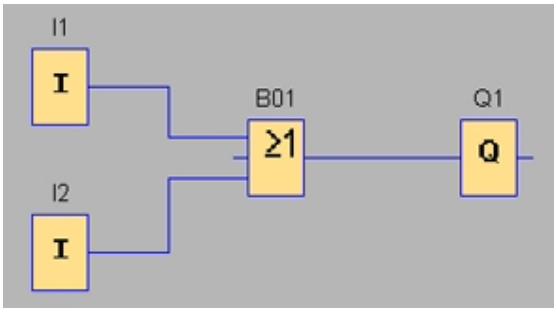


Fig 2.16

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 fig 2.17. The red lines carry a 1-signal and the blue lines carry a 0-signal.

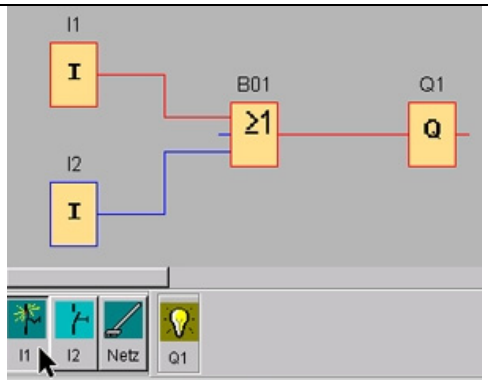


Fig 2.17: Program Simulation demo

Program Transfer to LOGO!

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

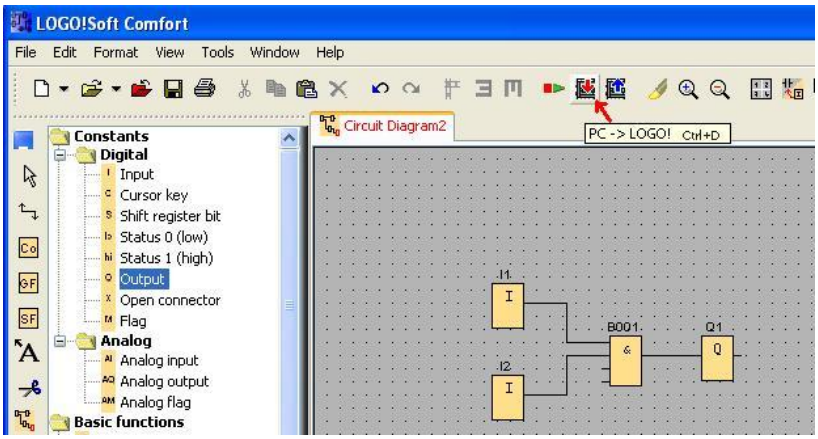


Fig 2.18: Program transfer to LOGO!

Starting the program on LOGO!

- ESC key
- Select "START"
- OK key

Conduct Lab Activity 3

2.4 Lab Activity 1

Objective: To identify the Normally Open and Normally Closed pushbuttons on the LOGO! BASIC Control unit.

Fig 2.19: LOGO! Control Unit

The LOGO! Control Unit has three pushbuttons as shown in fig 2.19. Perform this activity to identify the NO and NC pushbuttons.

Procedure:

A. Enter a program to *turn on the white indicator light (Q3) when the green pushbutton (I1) is pressed*. Run the program and observe the result. Has the white light turned ON?

_____

If yes, the green pushbutton is a NO pushbutton.

B. Enter a program to *turn on the white indicator light (Q3) when the red pushbutton (I2) is pressed*. Run the program and observe the result. Has the white light turned ON?

_____

If not, the red pushbutton is a NC pushbutton. It becomes NO when pressed.

C. Enter a program to *turn on the white indicator light (Q3) when the white pushbutton (I3) is pressed*. Run the program and observe the result. Has the white light turned ON?

_____

If yes, the white pushbutton is a NO pushbutton.

D. Enter your results in the table below:

Pushbutton	**Type (**NO/NC?)	**Condition in normal state** (open/close?)
Green		
Red		
White		

2.5 Lab Activity 2

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

The IO Simulator shown in fig 2.20 could be used to interact with the LOGO! in order to simulate the input and output signals.

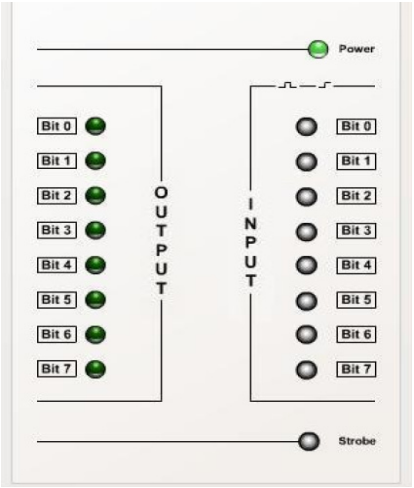


Fig 2.20: IO Simulator

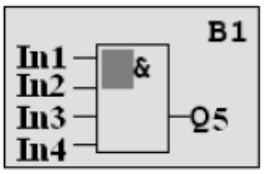
Procedure:

- Connect the IO Simulator to the LOGO! as shown in fig 2.21

Fig 2.21 Connecting IO Simulator to LOGO!

- 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 Fig 2.22 for input/output details)

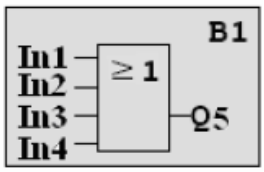
AND



Truth Table

I9	I10	Q5

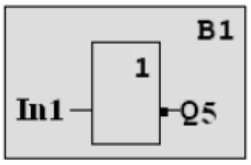
OR



Truth Table

I11	I12	Q6

NOT



Truth Table

I11	Q7

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

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

Fig 2.22: IO Map for LOGO! & IO Simulator

2.6 Lab Activity 3

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.

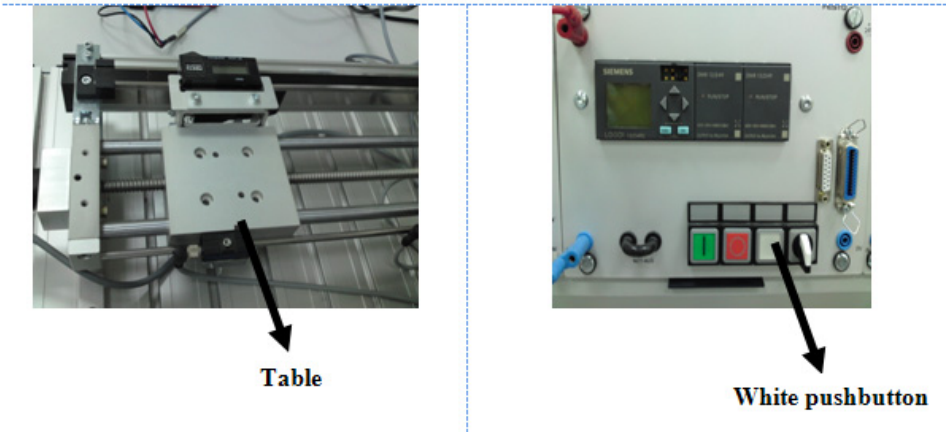


Fig 2.23: Table & White pushbutton

Function Block Diagram:

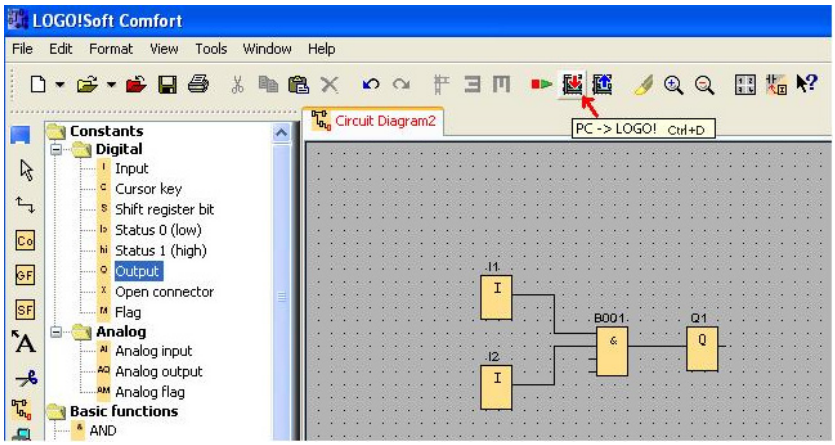


Fig 2.24: Table & White pushbutton

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:

Control Elements	PLC Input/Output

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.

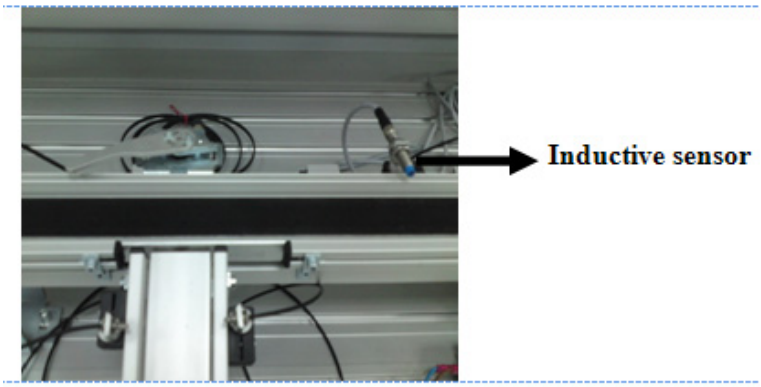


Fig 2.25: Inductive Sensor

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:

Control Elements	PLC Input/Output

2.7 Lab Activity 4

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:

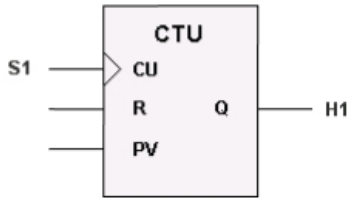
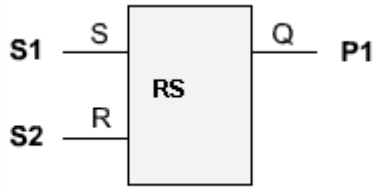
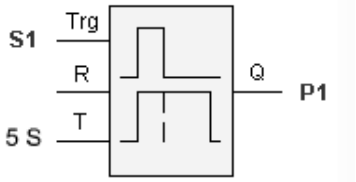
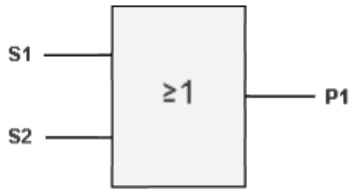
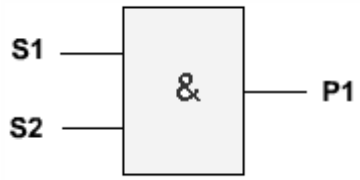
Function Block Diagram:

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.

2.8 Module 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		
2	Count the number of produced Cola cans		
3	Switch OFF a machine after a certain time		
4	Turn ON and Turn OFF a motor using the same function block		
5	Turn ON a lamp using either PB1 or PB2 or both		

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

a) A bump (B) turns ON when switch (S1) is pressed and it turns OFF when (S2) is pressed

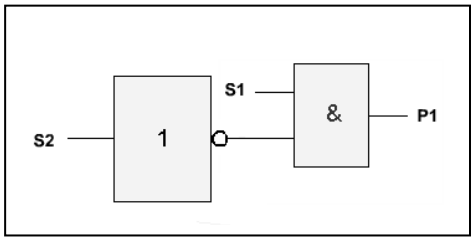
b) A light (L1) turns ON when switch (S1) is pressed and switch (S2) is NOT pressed (released)

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

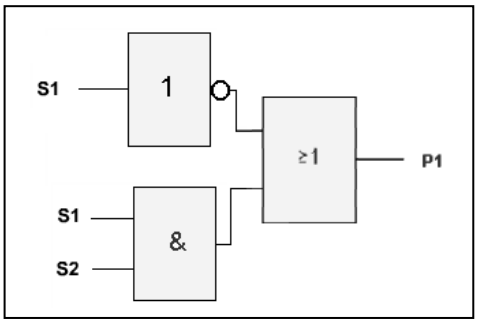
3. For the following FBDs write the correct expression:

a)



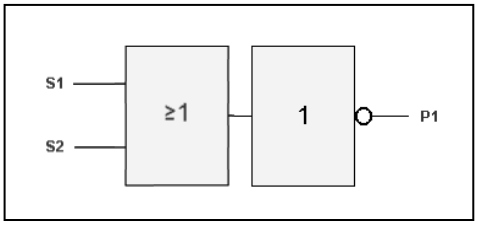
_____

b)



_____

c)



_____