



# **SENSORS AND AUTOMATION**

**Sarvesh Anand Mankar**  
**142203013**  
**SY Comp Div-2, S5 Batch**

# Practical-6: Arithmetic and Logical Operators

## Aim:

1. Develop a ladder using standard procedure.
2. Understand the use of Logical instructions.
3. Understand the use of Arithmetic instructions.

## Theory:

Digitization is a process where continuous analog signal is converted into a finite number of discrete states. These states are well separated so that noise does not create errors.

The resulting digital signal has following advantages:

1. storage over arbitrary periods of time.
2. flawless retrieval and reproduction of the stored information.
3. flawless transmission of the information.

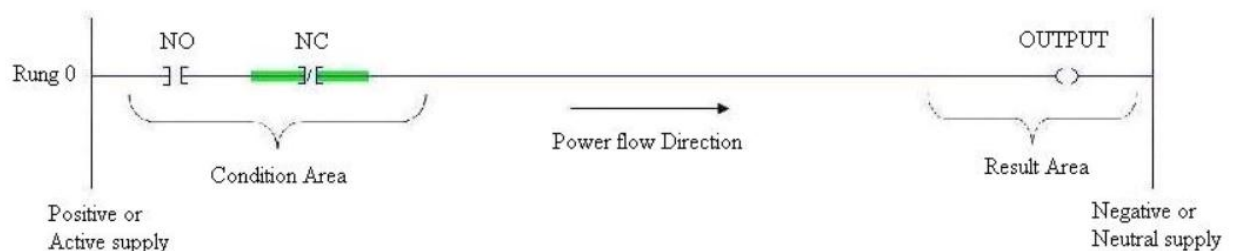
Boolean Logic	Boolean Algebra	Voltage State (positive true)	Voltage State (negative true)
True (T)	1	High(H)	Low(L)
False (F)	0	Low(L)	High(H)

Each manufacturer of PLC systems has own style of writing the instructions. Different PLCs has different instruction sets but even some common basic instructions are shared by all the PLCs. All manufacturers give different software packages for programming PLCs. Ladder is most commonly used programming language. Prior to PLCs, relay logic was used in industry. Ladders were developed to mimic or imitate relay logic. Relay Logic / Instructions A relay is simple magnetic device which acts as a control switch.

When the switch is on, current will flow through the coil on iron piece. This iron core acts as an electromagnet and due to the magnetic field the upper contact gets attracted towards the lower one and the circuit gets completed, allowing current to flow from the load.

### Ladder Programming:

Ladder diagram is a popular language of programming PLCs. A ladder diagram shows the sequence of logic execution which is presented diagrammatically. In a ladder diagram, there are two vertical lines generally called as Phase (positive) or neutral. Rungs which show current flow in horizontal direction are the sequence in which the logic executes. The analogous to a relay, a ladder has two main symbols which are contacts and output coil. Generally, each rung has inputs (contacts) on the left-hand side and outputs (coil) on the right-hand side. These contacts and coils are called as bits of the relays. Each input and output are individual bits in I/O files. An instruction in a ladder instructs PLCs how to respond to the bits in I/O files which are stored in the memory. Input contacts are the condition area; the conditions must be fulfilled to change the status of the output coils.



Typical Ladder Diagram

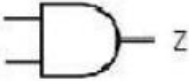
COEP Tech

## Trurth Table:

X	Y	Z
0	0	0
0	1	0
1	0	0
1	1	1

Truth Table of AND

X   Y   Z

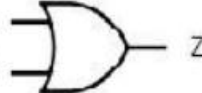


X   Y   Z

X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	1

Truth Table of OR

X   Y   Z

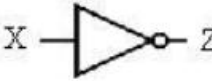


X   Y   Z

X	Z
0	1
1	0

Truth Table Of NOT

X   Z

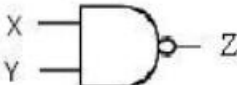


X   Z

X	Y	Z
0	0	1
0	1	1
1	0	1
1	1	0

Truth Table Of NAND

X   Y   Z

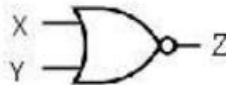


X   Y   Z

X	Y	Z
0	0	1
0	1	0
1	0	0
1	1	0

Truth Table Of NOR

X   Y   Z




X   Y   Z

X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	0

Truth Table Of XOR (Exclusive OR)

X   Y   Z




X   Y   Z



## Experiment:

### Addition

 **Virtual Labs**  
The World's Best of Online Education

Implementation Of PLC Arithmetic Instructions

**Programmable Logic Controller**

Basic **Arithmetic** Counters-Timers

ADD SUB MUL DIV CPT


Open Save Compile Run Development

Tab 1


0

ADD: ADDITION

A	
I1	100
B	
I2	200
Result	
RESULT	300.00



### Subtraction

 **Virtual Labs**  
The World's Best of Online Education

Implementation Of PLC Arithmetic Instructions

**Programmable Logic Controller**

Basic **Arithmetic** Counters-Timers

ADD SUB MUL DIV CPT


Open Save Compile Run Development

Tab 1

0

SUB: SUB

A	
I1	1000
B	
I2	500
Result	
RESULT	500.00



## Multiplication

Virtual Labs  
The Best Way of Understanding

Implementation Of PLC Arithmetic Instructions

**Programmable Logic Controller**

Basic Arithmetic Counters-Timers

ADD SUB MUL DIV CPT

Open Save Compile Run Development

Tab 1

0

MUL:MUL

A	200
B	3
Result	6

END

## Division

Virtual Labs  
The Best Way of Understanding

Implementation Of PLC Arithmetic Instructions

**Programmable Logic Controller**

Basic Arithmetic Counters-Timers

ADD SUB MUL DIV CPT

Open Save Compile Run Development

Tab 1

0

DIV:DIV

A	1000
B	5
Result	200.00

END

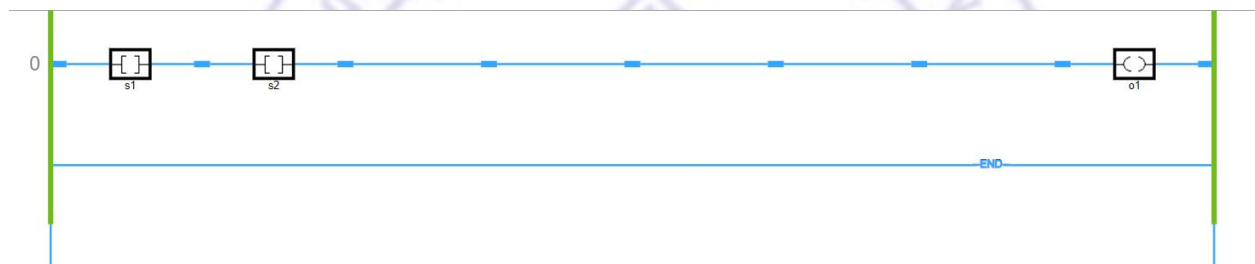
## AND Logic:

AND Gate

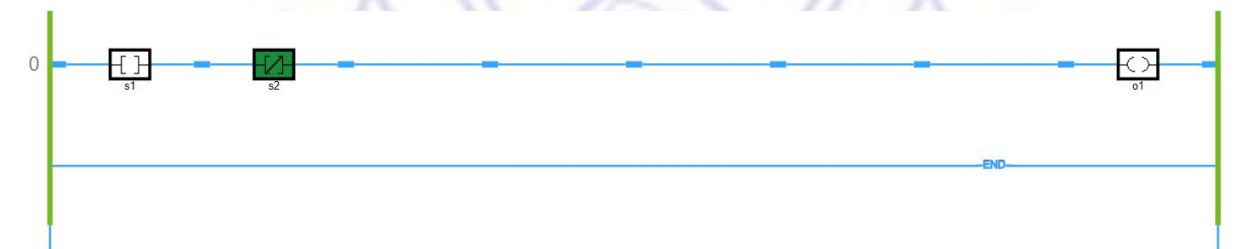


INPUT		OUTPUT
A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

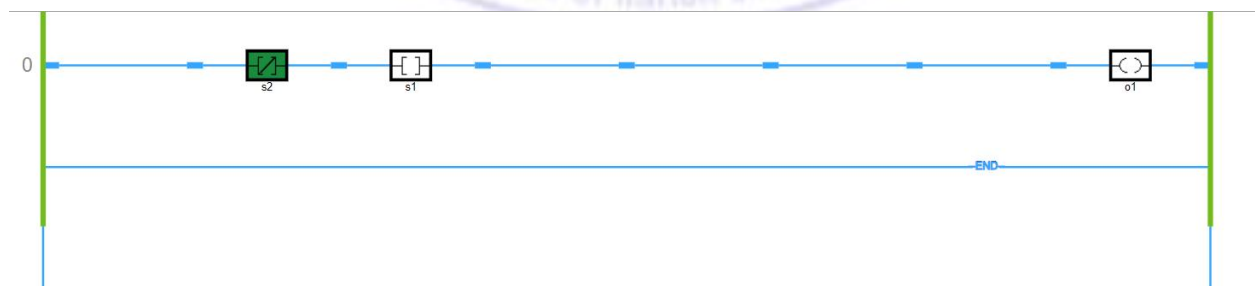
$$\text{AND}(0,0)=0$$



$$\text{AND}(0,1)=0$$

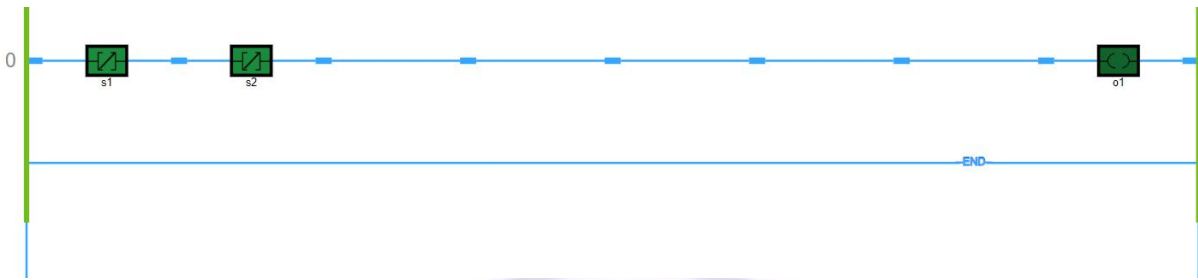


$$\text{AND}(1,0)=0$$

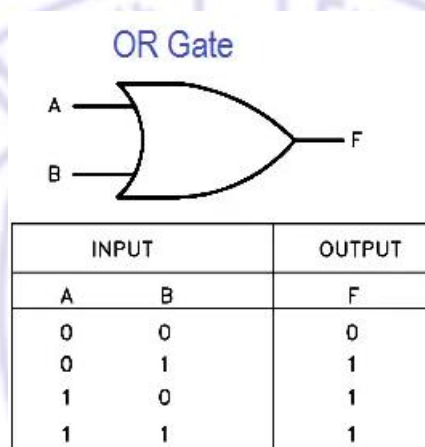




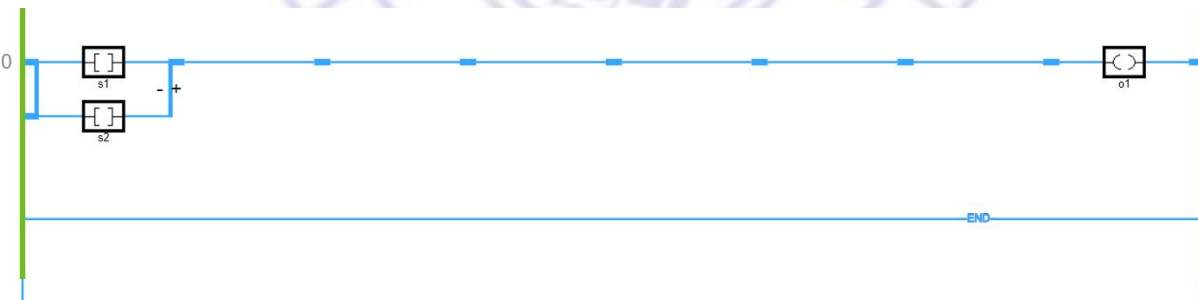
$$\text{AND}(1,1)=0$$



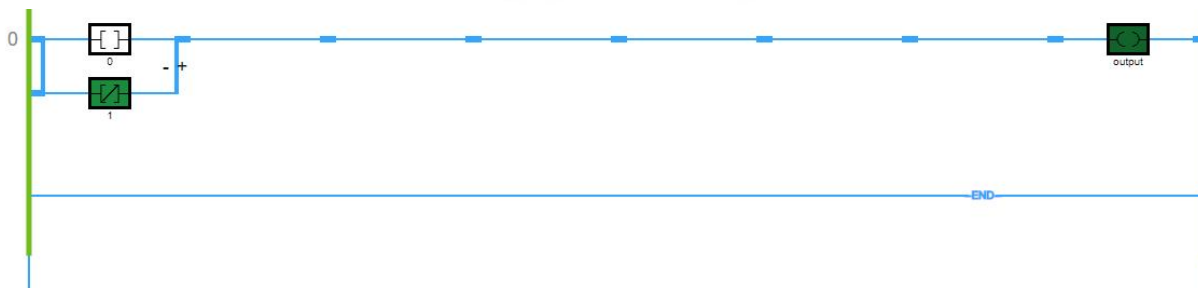
OR Logic:



$$\text{OR}(0,0)=0$$

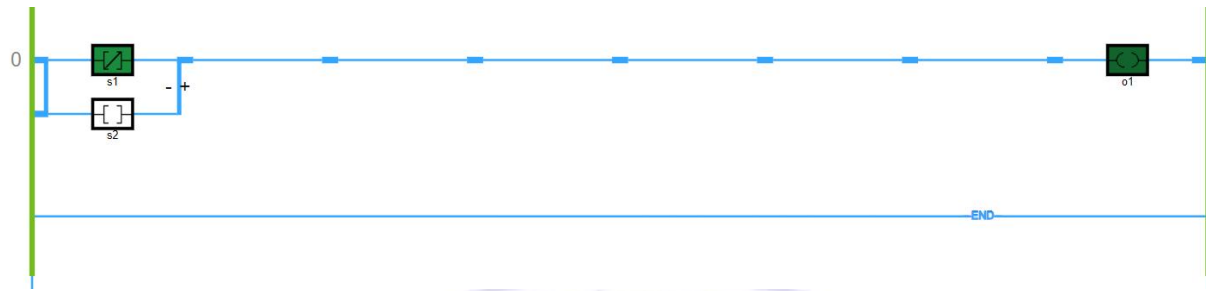


$$\text{OR}(0,1)=1$$

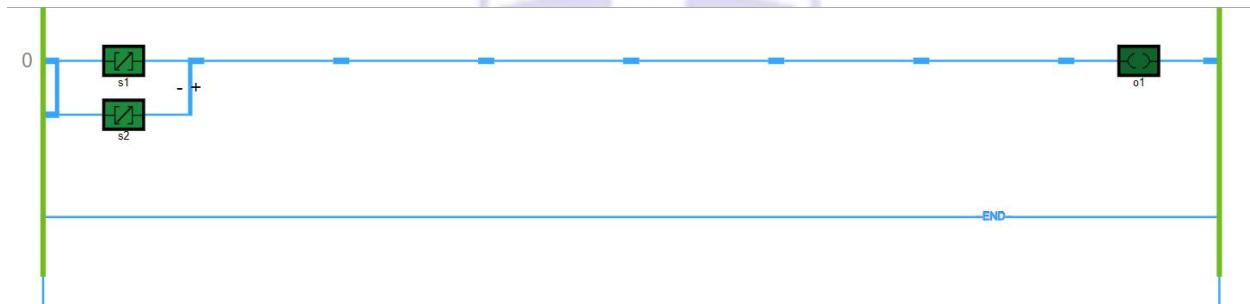




$$\text{OR}(1,0)=1$$



$$\text{OR}(1,1)=1$$



### **Conclusion:**

The experiment aimed to develop a ladder using a standard procedure, understand the use of logical instructions, and understand the use of arithmetic instructions. Through the experiment, the participant was able to gain knowledge and hands-on experience in developing a ladder using the standard procedure and identify the different types of logical and arithmetic instructions used in ladder programming. Logical instructions included AND, OR operators, while arithmetic instructions included addition, subtraction, multiplication, and division. The experiment provided a comprehensive understanding of ladder programming and its application in various industrial processes, enabling the participant to develop efficient and effective ladder programs to control automation systems.