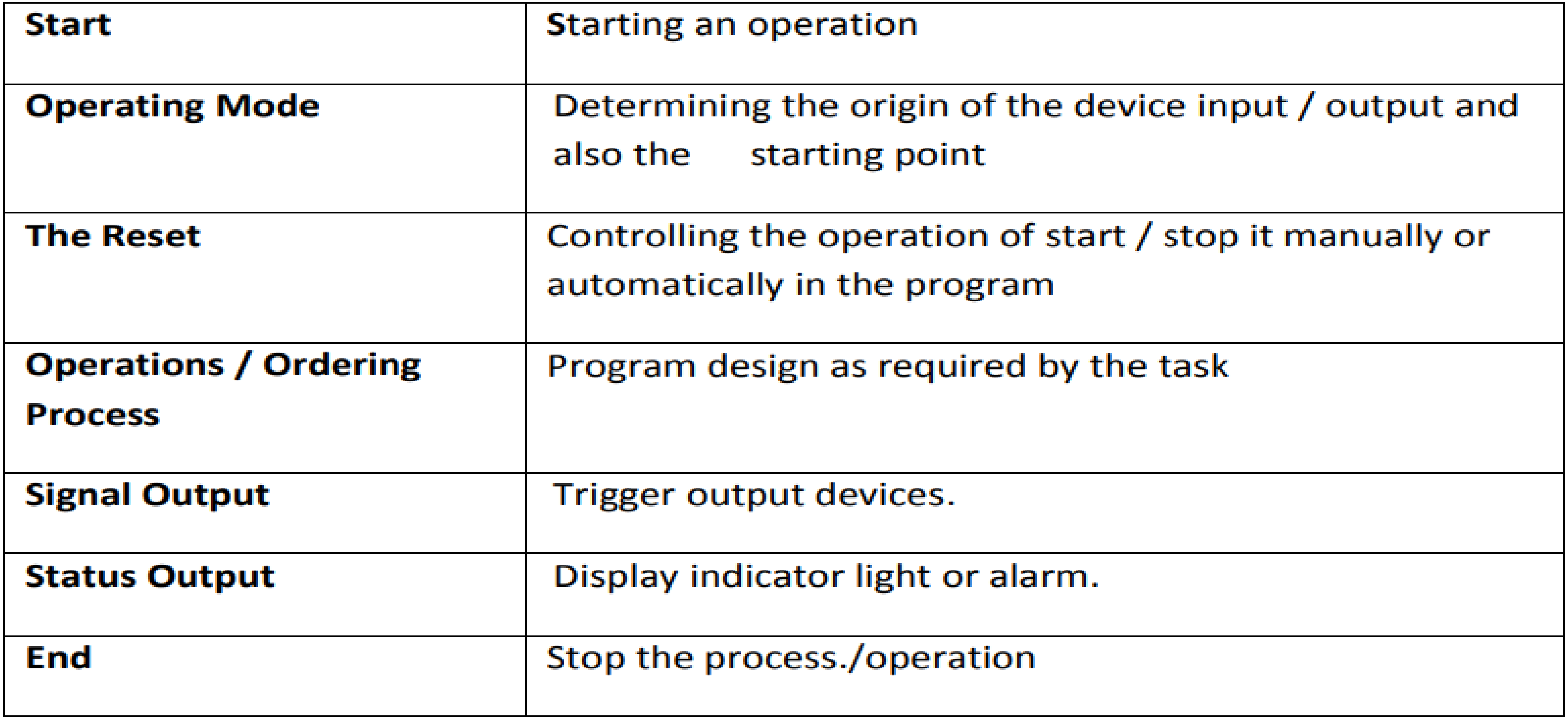
PLC Programming

Professor Suraiya Pervin, Ph.D.

PLC Programming

- When using a PLC, it’s important to design and implement concepts depending on particular use case.
- PLC programming process is to plan activities such as design and write a program to perform the required tasks.
- A PLC program consists of a set of instructions either in textual or graphical form, which represents the logic that governs the process the PLC is controlling.
- **IEC 1131-3** is the international standard for PLC programming languages.

Here are the parts that should be there in a PLC program



Start	Starting an operation
Operating Mode	Determining the origin of the device input / output and also the starting point
The Reset	Controlling the operation of start / stop it manually or automatically in the program
Operations / Ordering Process	Program design as required by the task
Signal Output	Trigger output devices.
Status Output	Display indicator light or alarm.
End	Stop the process./operation

PLC Programming Languages

There are two main classifications of PLC programming languages

1. Graphical Form

- Ladder Diagrams (LD) (i.e. Ladder Logic)
- Function Block Diagram (FBD)
- Sequential Function Chart (SFC)

2. Textual Language

- Instruction list
- Structured text

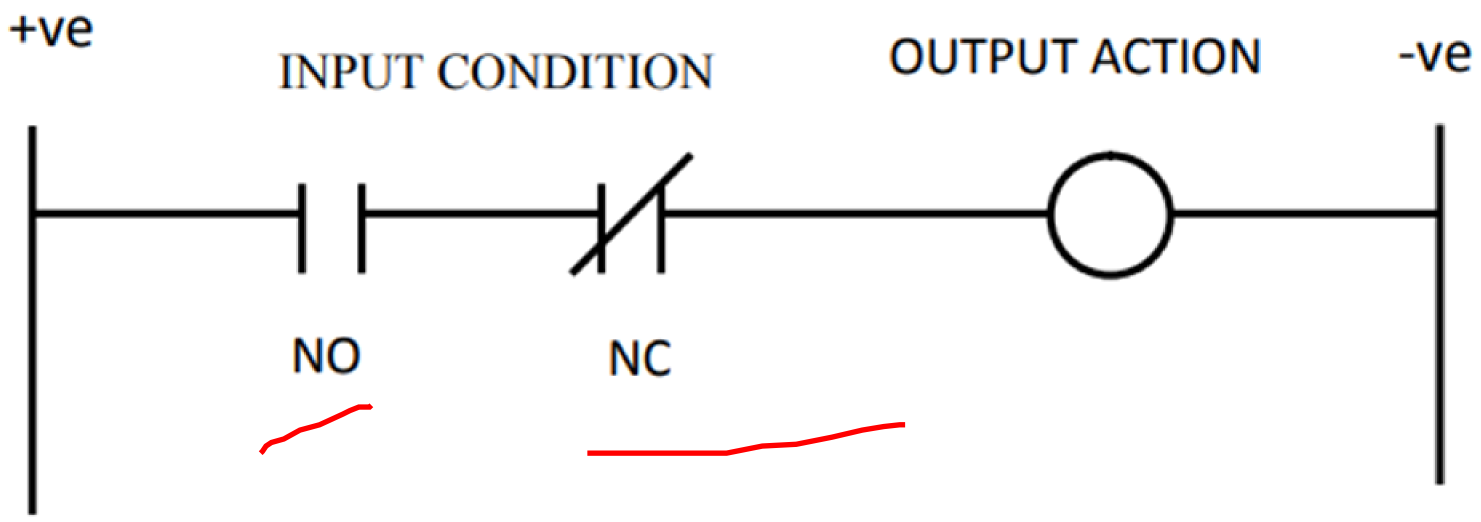
Although all of these PLC programming languages can be used to program a PLC, graphical languages are typically preferred to textual languages.

Ladder Logic

- Ladder logic is a programming language that represents a program by a graphical diagram based on the circuit diagrams of relay logic hardware.
- It is primarily used to develop PLC software for industrial control applications.
- Ladder logic can be thought of as a rule-based language.
- A "**rung**" in the ladder represents a rule.
- When implemented with relays and other electromechanical devices, the various rules execute simultaneously and immediately.
- When implemented in a PLC, the rules are typically executed sequentially by software in a continuous loop, or "**scan**".
- By executing the loop fast enough, typically many times per second, the effect of simultaneous and immediate execution is achieved.

Ladder Diagram

- Ladder Diagram contains tracks from left to right contact diagram .
- This platform is connected to contact elements available **Normally open (NO)** or available **Normally closed (NC)** through the current path and loop elements.
- Ladder diagram also shows the control circuit and the display function and a combination of the sequence of operations for each branch of the horizontal lines separately.



Ladder Diagram

Syntax and examples

- The language itself can be seen as a set of connections between logical checkers (contacts) and actuators (coils).
- Ladder logic has contacts that make or break circuits to control coils.
- Each coil or contact corresponds to the status of a single bit in the PLC,s memory.

Rung input: Checkers (contacts)

- **—[ ]—** Normally open contact.
- **—[\]—** Normally closed ("not") contact .

Rung output: Actuators (coils)

- **—( )—** Normally inactive coil.
- **—(\)—** Normally active ("not") coil.

The "coil" (output of a rung) may represent a physical output which operates some device connected to the programmable controller, or may represent an internal storage bit for use elsewhere in the program.

Logical AND

- **------[ ]------------------[ ]------------------( )**

  Key switch 1   Key switch 2   Door motor

- The above realizes the function: **Door motor = Key switch 1 AND Key switch 2**
- This circuit shows two key switches that security guards might use to activate an electric motor on a bank vault door.
- When the normally open contacts of both switches close, electricity is able to flow to the motor which opens the door.

Logical AND with NOT

```
--------[ ]---------------[\]----------------( )
  Close door     Obstruction        Door motor
```

- The above realizes the function: **Door motor = Close door AND NOT (Obstruction).**
- This circuit shows a push button that closes a door, and an obstruction detector that senses if something is in the way of the closing door.
- When the normally open push button contact closes and the normally closed obstruction detector is closed (no obstruction detected), electricity is able to flow to the motor which closes the door.

Logical OR

-
  ```
  --+---------[ ]---------+-----------------( )
    | Exterior unlock |              Unlock
    |                 |
    +---------[ ]----------+
      Interior unlock
  ```
- The above realizes the function: **Unlock = Interior unlock OR Exterior unlock**

- This circuit shows the two things that can trigger a car's power door locks.
- The remote receiver is always powered.
- The unlock solenoid gets power when either set of contacts is closed.

Industrial STOP/START

In common industrial latching start/stop logic we have a "Start" button to turn on a motor contactor, and a "Stop" button to turn off the contactor.

```
--+----[ ]---+----[\]---( )
  |   Start  |   Stop   Run
  |          |
  +----[ ]---+
       Run

-------[ ]--------------( )
       Run             Motor
```

The above realizes the function: **Run = (Start OR Run) AND (NOT Stop)**

For safety reasons, an emergency stop ("ES") may be hardwired in series with the "Start" switch, and the relay logic should reflect this.

```
 - - [\]- - - - [\]- - - - + - - [ ]- - + - - - - - - - - - -( )
     ES         Stop       |   Start  |                      Run
                           |          |
                           + - - [ ]- - +
                                 Run


     - - - - - - -[ ]- - -  - - - - - - - - - - - -( )
                  Run                              Motor
```

The above realizes the function: **Run = (NOT ES) AND (NOT Stop) AND (Start OR Run)**

Complex logic

- Here is an example of what two rungs in a ladder logic program might look like.
- In a complex system there will be hundreds or thousands of "rungs" on a ladder, which are numbered in order of evaluation.
- Typically, complex ladder logic is "read" left to right and top to bottom.
- As each of the lines (or rungs) are evaluated the output coil of a rung may feed into the next stage of the ladder as an input.

1. Realising the function: **A/C = Switch AND (HiTemp OR Humid).**

```
- - - -[ ]- - - - - - - - - - +- - - - -[ ]- - - - - +- - - - -( )
     Switch                   |    HiTemp            |      A/C
                              |                      |
                              +- - --[ ]-  -- - --+
                                   Humid
```

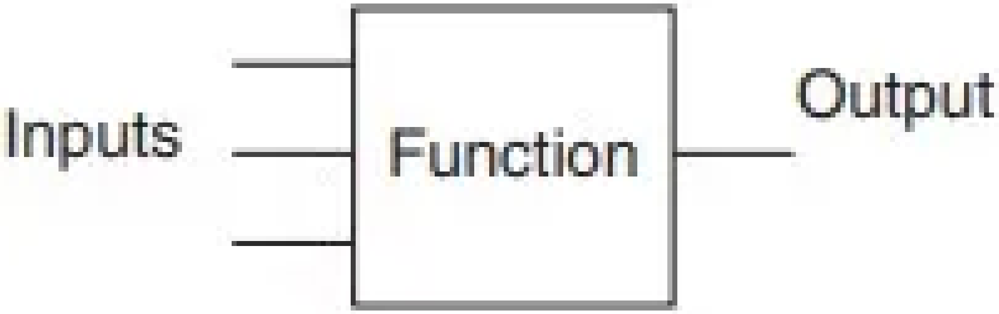
2. Realizing the function: **Cooling = A/C AND (NOT Heat).**

```
- - - -[ ]- - - -[\]- - - - - - - - - - - - - - - - - - - - - -( )
       A/C   Heat                                    Cooling
```

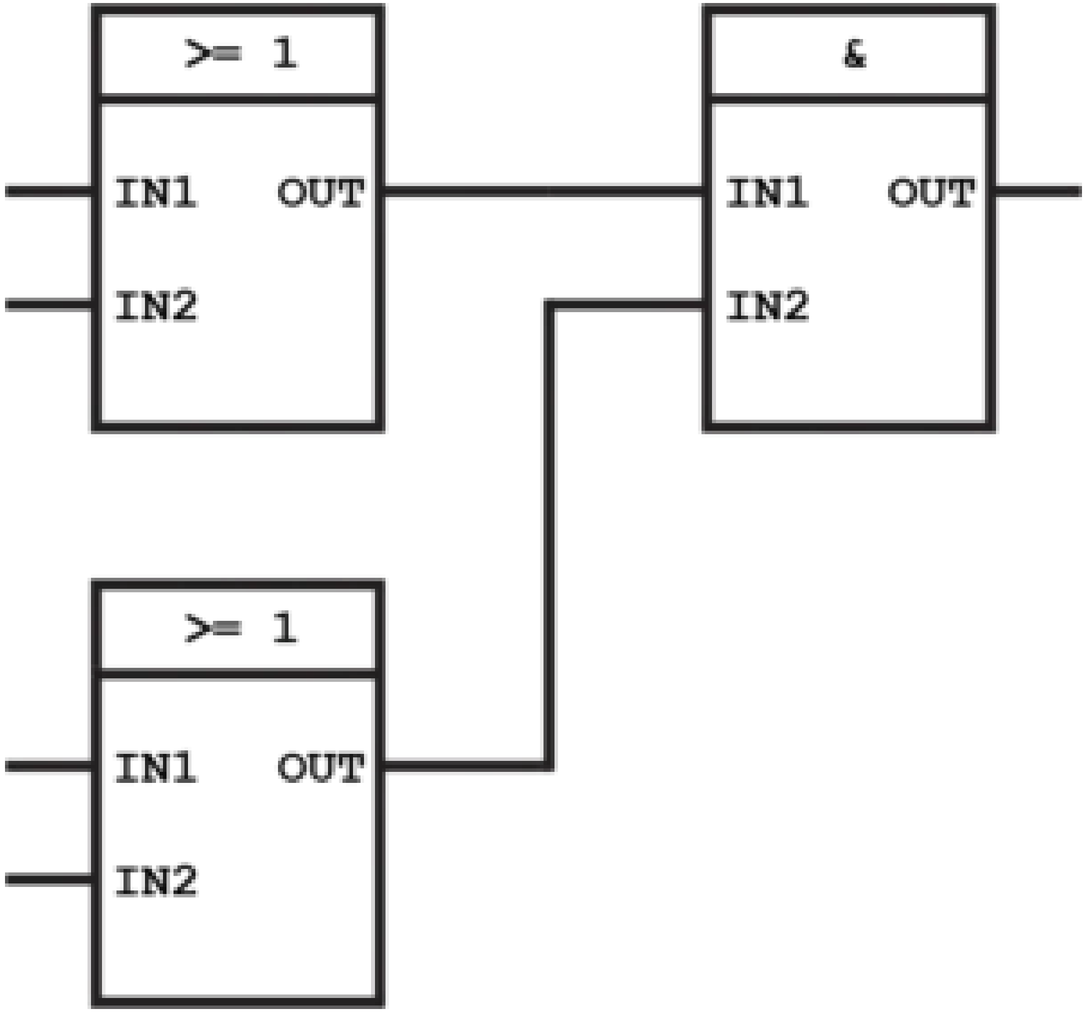
- This represents a slightly more complex system for rung 2.
- After the first line has been evaluated, the output coil "A/C" is fed into rung 2, which is then evaluated and the output coil "Cooling" could be fed into an output device "Compressor" or into rung 3 on the ladder.
- This system allows very complex logic designs to be broken down and evaluated.

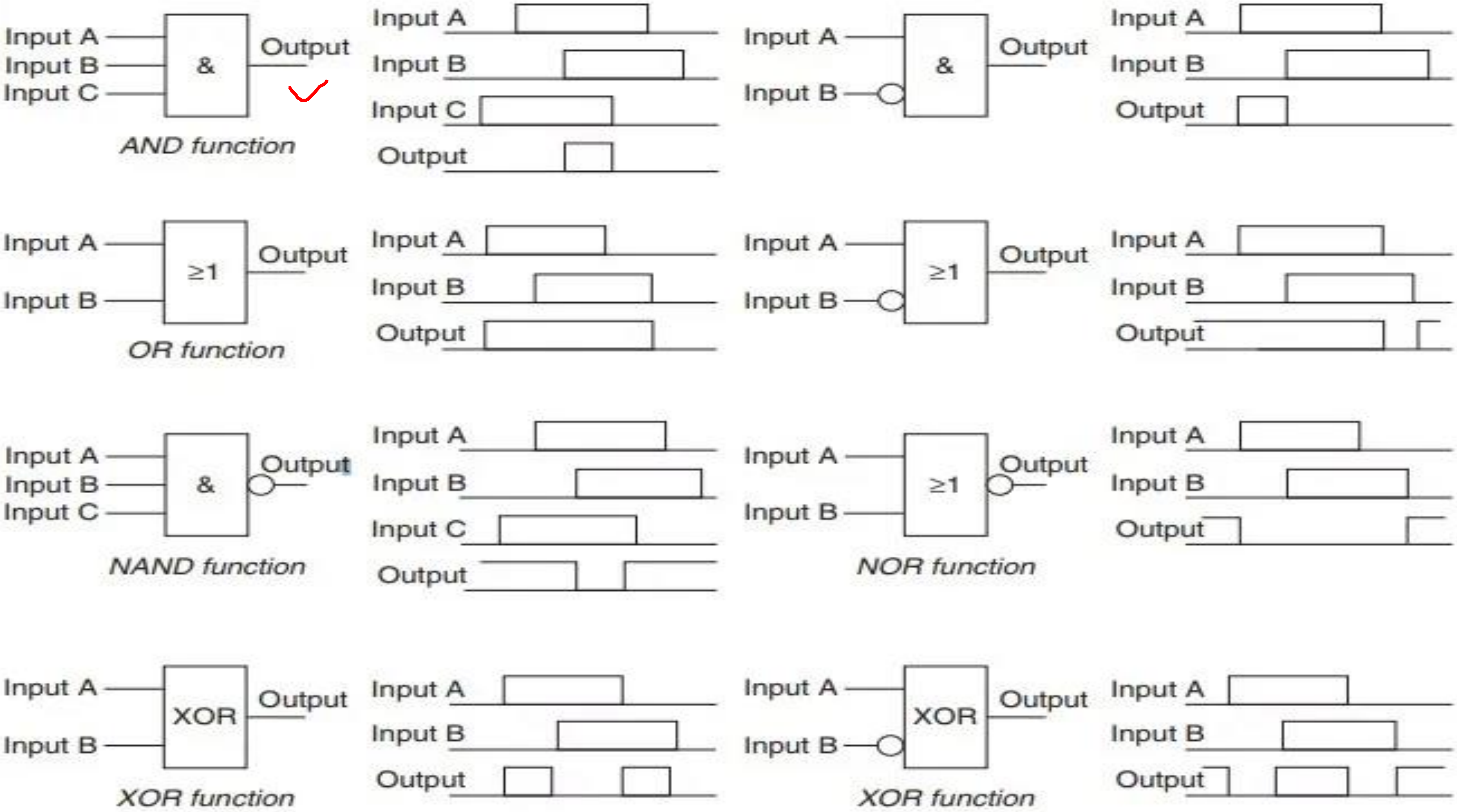
Functional Block Diagrams

- Functional Block Diagram (FBD) is a simple and graphical method to program multiple functions in PLC.
- PLC Open has described using FBD in the standard IEC 61131-3.
- A function block is a program instruction unit that, when executed, yields one or more output.
- It is represented as a rectangular block with inputs entering on left and output lines leaving at the right. It gives a relation between the state of input and output



- The advantage of using FBD is that any number of inputs and outputs can be used on the functional block.
- When using multiple input and output, you can connect the output of one function block to the input of another.
- Various function blocks used in FBD programming are shown in the next slide.





AND function

OR function

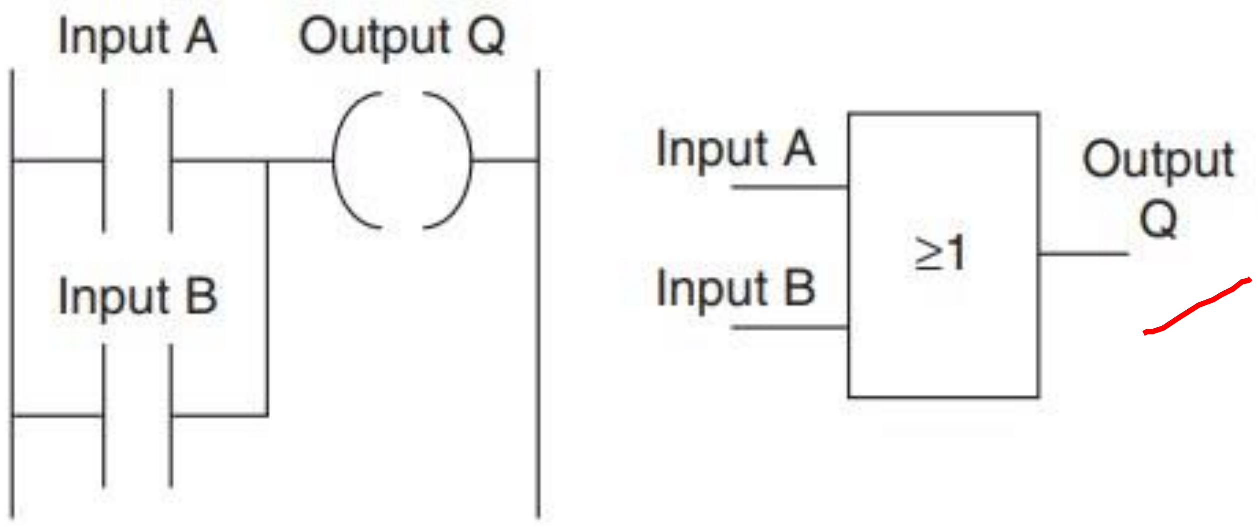
NAND function

NOR function

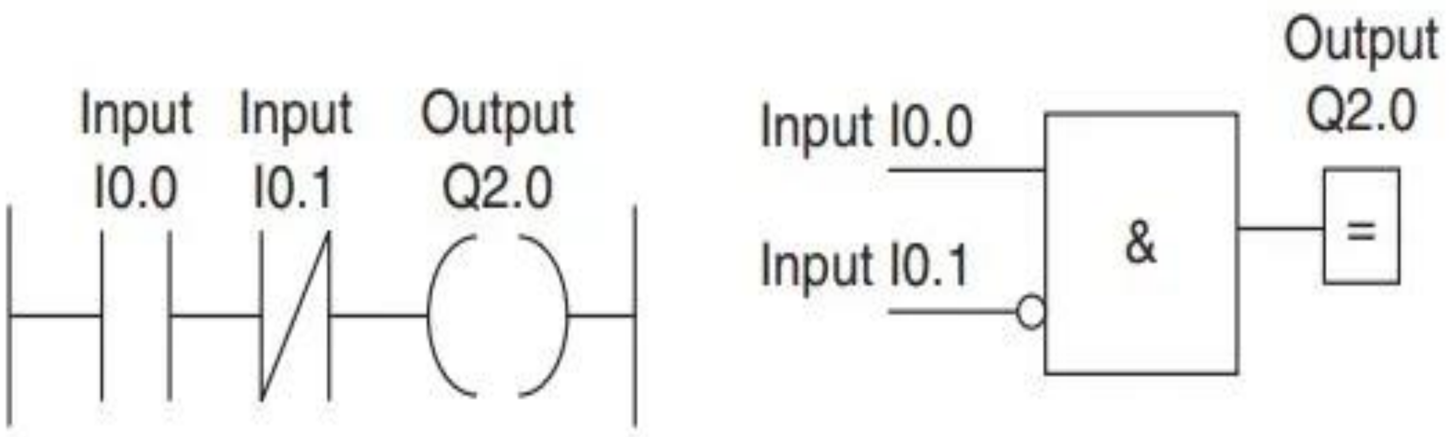
XOR function

XOR function

The figure below shows a ladder diagram and its function block equivalent in Siemens notation.



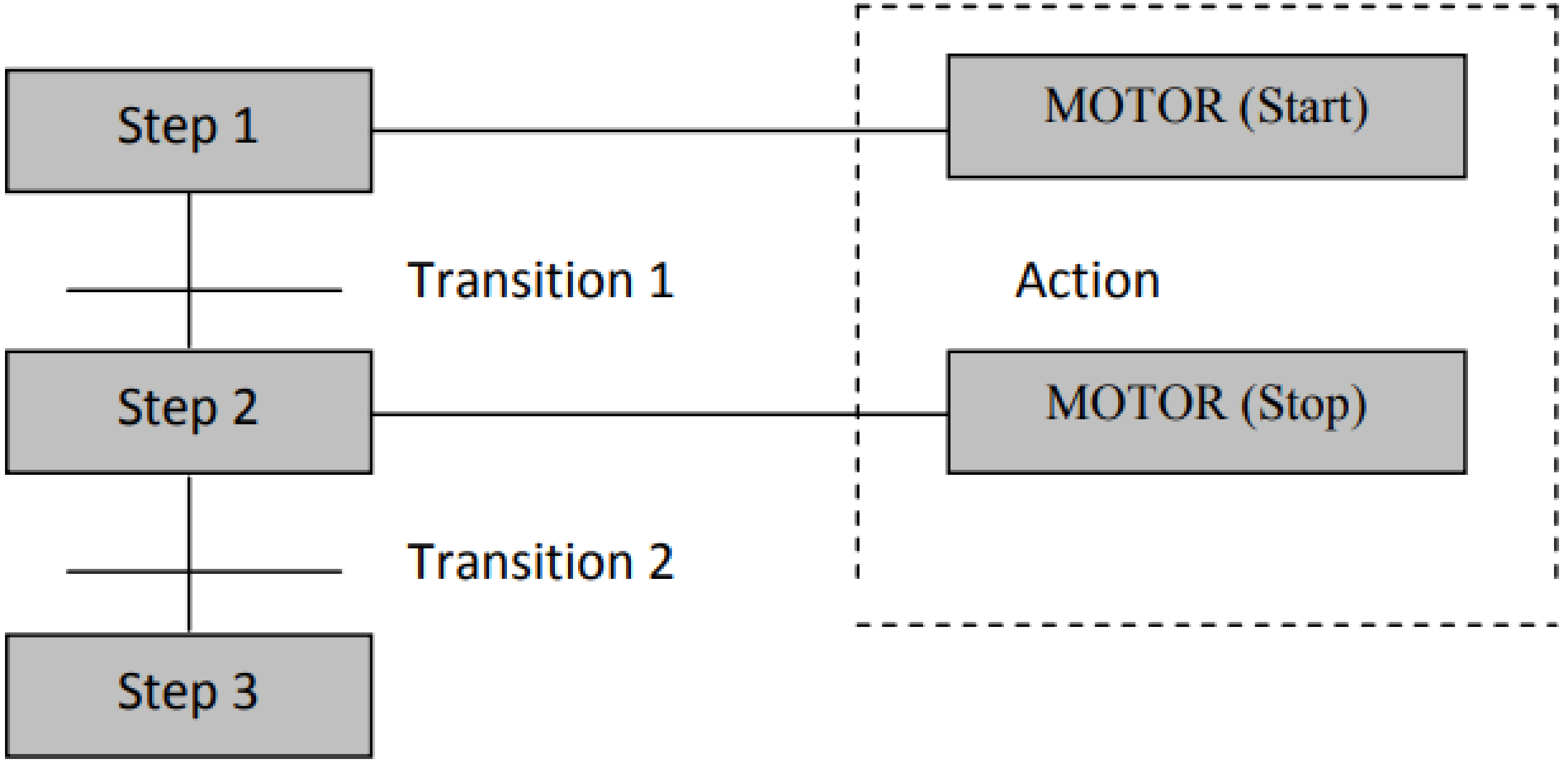
Ladder to functional block



Ladder to functional block diagram

Sequential Function Chart

- These are similar to flowcharts, but much more powerful.
- SFC programming offers a graphical method of organizing the program.
- The three main components of an SFC are **steps, actions and transitions.**
- **Steps** are merely chunks of logic, i.e., a unit of programming logic that accomplishes a particular control task.
- **Actions** are the individual aspects of that task.
- **Transitions** are the mechanisms used to move from one task to another.
- Control logic for each Step, Action and Transition is programmed in one of the other languages such as Ladder Diagram or Structured Text.
- One step consists of action based on the transition.
- The action consists of the sequence structure itself.



Sample Program In Sequencial Function Chart Language

Structured Text

- Structured Text (ST) is a high level textual language that is a Pascal like language.
- It is very flexible and intuitive for writing control algorithms.
- Structured Text uses operators such as logical branching, multiple branching, loops.
- A series of statements (logic) is constituted of expressing assignments and relationships using several operators.
- ST is ideal for tasks requiring complex math, algorithms or decision-making.

Benefits of Structured Text

- ➢ People trained in computer languages can easily program control logic
- ➢ Symbols make the programs easy to understand
- ➢ Programs can be created in any text editor
- ➢ Runs as fast as ladder

Structured Text Programming Example

- We have Motor that will be controlled manually by 2 push buttons (Start and Stop).
- When the Start Push Button is pushed then the Motor will be turned ON.
- When the Stop Push Button is Pushed then we want to stop the Motor.

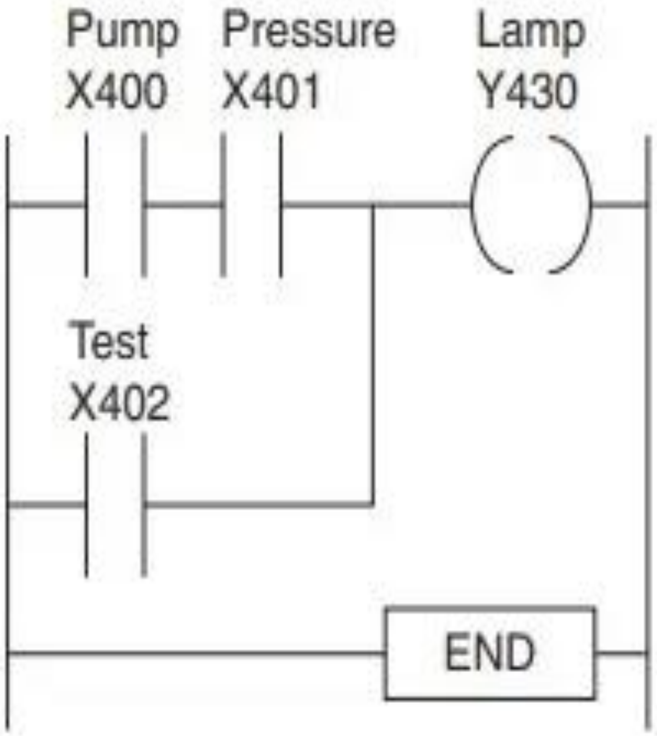
```
IF StartPb THEN
Motor := 1;
END_IF;
IF StopPb THEN
Motor := 0;
END_IF;
```

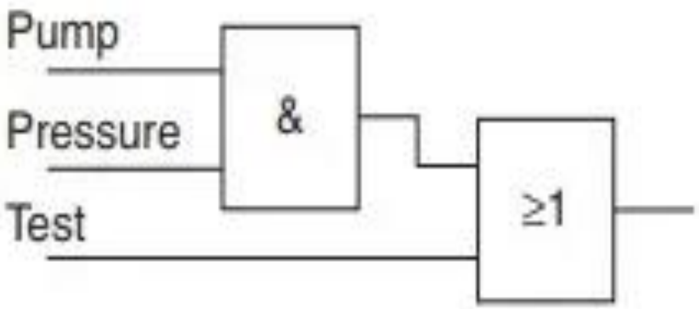
Instruction List (IL): A low level “assembler like” language that is based on similar instructions list languages found in a wide range of today’s PLCs.

```
LD          R1
MPC         RESET
LD          PRESS_1
ST          MAX_PRESS
RESET:      LD    0
ST          A_X43
```

PLC Programming Examples cont….

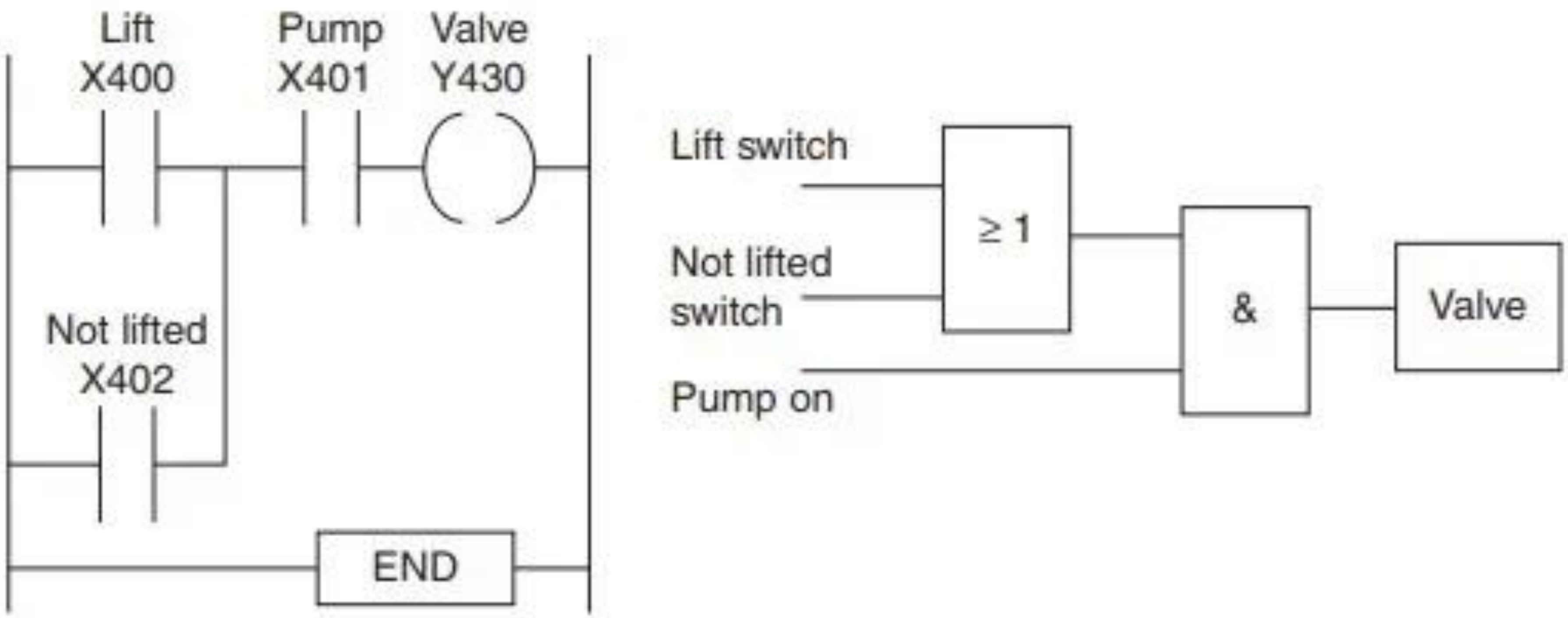
- A signal lamp is required to be switched on if a pump is running and the pressure is satisfactory, or if the lamp test switch is closed.
- In this application, if there should be an output from the lamp inputs from both pump and pressure sensors are required. Hence, AND logic gates are used.
- OR logic is used for the test input condition, it is required to give an output of lamp on regardless of whether there is a signal from the AND system.
- By using END or RET instruction in the ladder diagram, PLC has reached the end of the program.





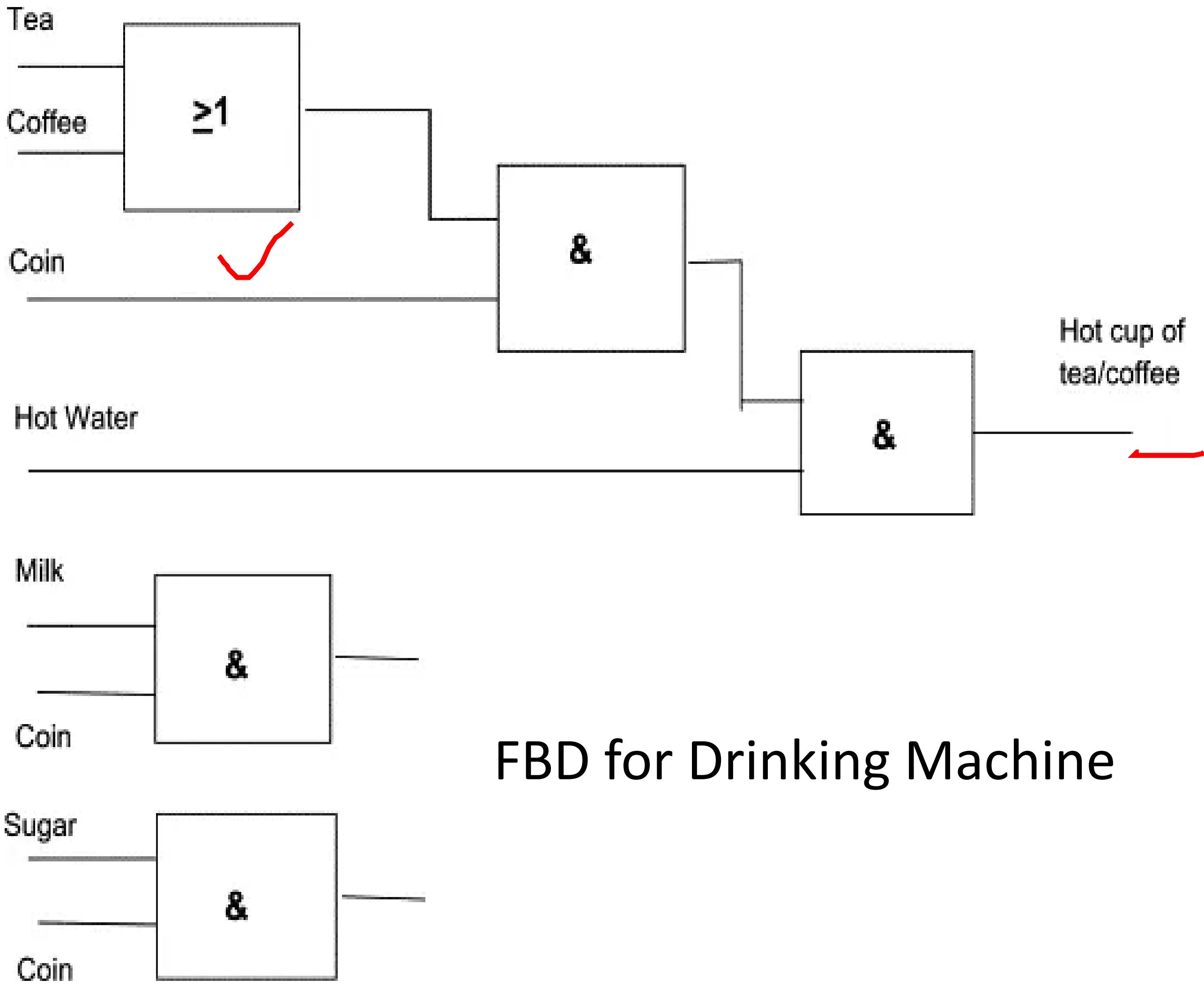
PLC Programming Examples cont….

- As another example, consider a valve that is to be operated to lift a load when a pump is running and either the lift switch is operated or a switch operated indicating that the load has not already been lifted and is at the bottom of its lift channel.
- OR logic is used for two switches and an AND logic is used with two switches and the pump. Valve will be operated only if the pump is ON and two switches are operated.



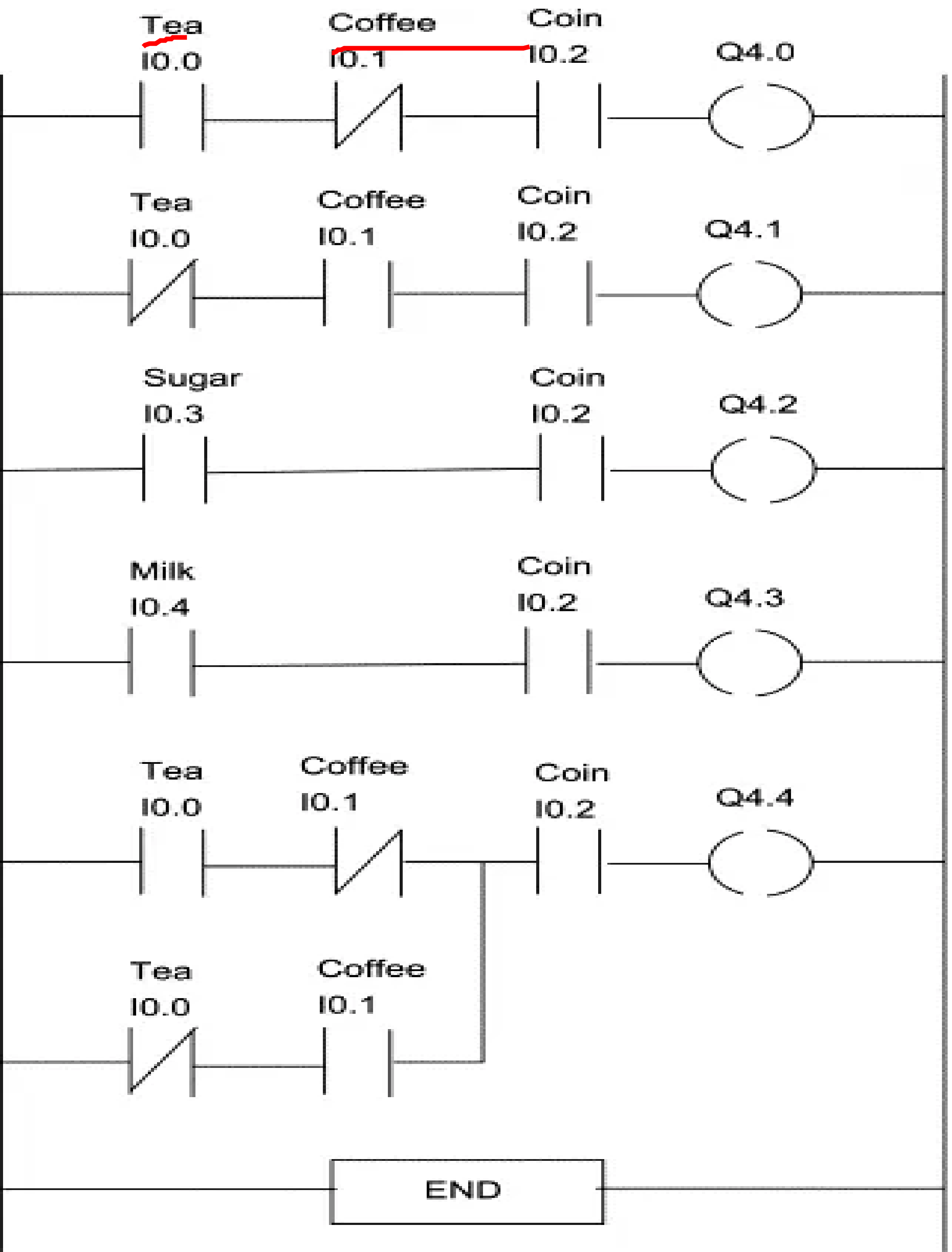
PLC Programming Examples cont….

- Consider a drinks machine that allows the selection of tea or coffee, milk or no milk, sugar or no sugar, and will supply the required hot drink on the insertion of a coin.
- In figure, it is seen that either tea or coffee is selected using OR logic gate.
- The first AND gate give an output when either Tea or coffee is selected and a coin is inserted into the machine.
- The output from this AND gate is given to the second AND gate.
- The 2nd AND gate operate only when hot water combines with tea/coffee.
- Milk and sugar are optional additions.



FBD for Drinking Machine

PLC Programming Examples cont....



This output causes tea powder to be put into the cup

This output causes coffee powder to be put into the cup

This output causes sugar to be put into the cup

This output causes milk to be put into the cup

This output cause hot water to enter the cup when tea or coffee has been selected and a coin inserted into the machine

Ladder Logic for Drinking Machine Application

Mnemonic Instruction

- There are other methods to program PLCs.
- One of the earliest techniques involved mnemonic instructions. Examples:

INSTRUCTION ---------------------	**MNEMONIC NAME**
Examine if closed -->	XIC
Examine If Open -->	XIO
Branch Start -->	BST
Next Branch -->	NXB
Output Latch -->	OTL
Timer On-Delay -->	Ton
Timer off-Delay -->	Toff
Retentive Timer -->	RTO
Count Up -->	CTU
Count Down -->	CTD
Reset -->	RST

Timers

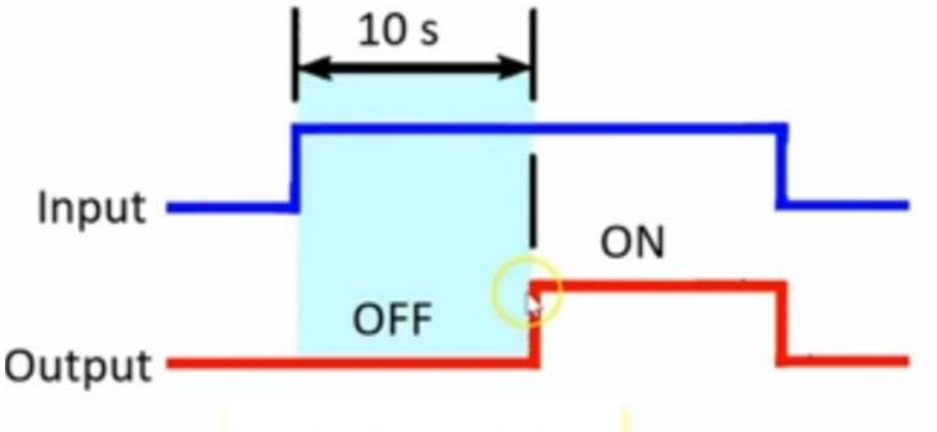
- A timer switch is a mechanism that controls the on and off timing of electrical equipment such as for appliances and devices.
- A timer is a small device that automatically opens and closes an electrical circuit for a specific period of time.
- PLC timers are internal PLC instructions that can be used to delay input and output signals in the PLC program.
- In short, we can say that it allows you to program the on/off switching of different devices in a simple and straightforward manner.
- A time switch are used in automation to regulate an electric circuit (for example appliances of a smart home).

- This switch can be used with any device connected to a power outlet/switch, such as TVs, lights, and HVAC units.
- We can link it to our home security system or use it to control our air conditioning with a switch that turns on and off automatically.
- Using a time switch is easy, after installation, the time switch will control our appliances depending on the settings we make.

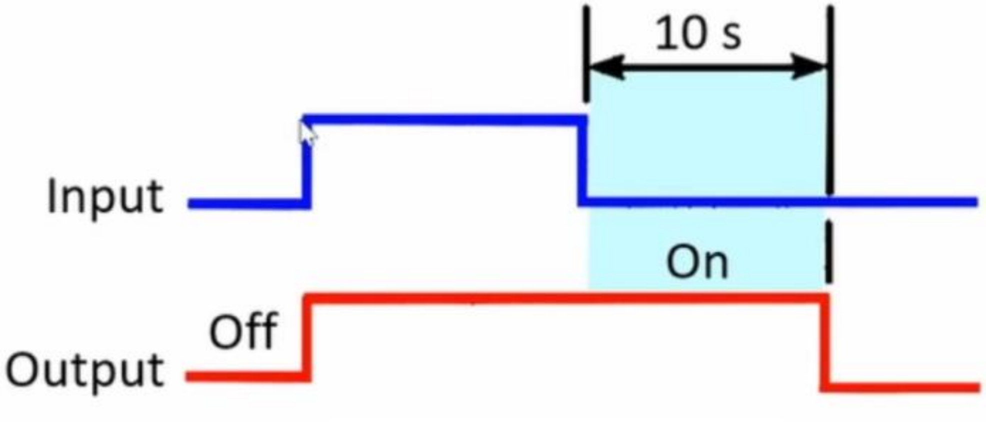
Types of PLC timer:

- The on-delay timer
- The off-delay timer.

- **ON DELAY**: The output relay of the timer will be ON only after the set time elapses.
- **OFF DELAY**: The output relay of the timer will be ON while the set time starts running. It will be OFF when the set time elapses.
- Basically On timer means it’s on in starting (for a set time) then it’s close… and off timer means it’s off in starting for a set time then it’s on after that time.
- After the input is turned on there is a delay before the output is turned on.
- After the input is turned off there is a delay before the output is turned off.



On-Delay Timer



Off-Delay Timer

Master and Jump control

- Master controls can be thought of as "emergency stop switches".
- An emergency stop switch typically is a big red button on a machine that will shut it off in cases of emergency.
- Master control relay (MCR) operators are responsible for monitoring the quality and accuracy of the on-air product, ensuring the transmission meets government regulations, troubleshooting equipment malfunctions, and preparing programming for play out.
- The JMP instruction is used to cause the PLC to skip over rungs.
- The Jump (JMP) instruction is paired with the Label (LBL) instruction by designating the same address number to each function.
- When a jump instruction is used, the PLC skips all the rungs between Jump Instruction and its associated Label Instruction.

What is a PLC counter?

- PLC counters are internal PLC instructions that can be used to count input or output signals in the PLC program.
- These counters operate like relay counters but you cannot hold a PLC counter in your hand and they do not need to be connected to wires to operate.
- The timer would time up to the value set by the user and the counter will count up to the value set by the user.
- The timers and counters are the fundamental PLC instructions and it is common to all PLCs.