GOVERNMENT OF TAMILNADU
DIRECTORATE OF TECHNICAL EDUCATION
CHENNAI – 600 025
STATE PROJECT COORDINATION UNIT

Diploma in Electrical and Electronics Engineering
Course Code: 1030
M – Scheme

e-TEXTBOOK
on
PROGRAMMABLE LOGIC CONTROLLER
for
V Semester DEEE

Convener for EEE Discipline:

Er.R.Anbukarasi ME.,
Principal,
Tamilnadu Polytechnic College,
Madurai, 625011.

Team Members for PROGRAMMABLE LOGIC CONTROLLER

THIRUMATHI V.THENMOZHI , M.E., M.I.S.T.E
PRINCIPAL I/C
GOVERNMENT POLYTECHNIC COLLEGE, KARUR

THIRU G.RAJAGOPAL, M.E.,
LECTURER/ELECTRICAL
P.A.C. RAMASAMY RAJA POLYTECHNIC COLLEGE, RAJAPALAYAM

THIRU M. SARAVANAKUMAR, B.E.,
PARTTIME LECTURER /EEE
TAMILNADU POLYTECHNIC COLLEGE, MADURAI

Validated by:

THIRU. M.VARATHARAJAN,
ASSISTANT PROFESSOR / EEE
THIAGARAJAR COLLEGE OF ENGINEERING,
MADURAI.

DIPLOMA IN ELECTRICAL AND ELECTRONICS ENGINEERING

M - SCHEME

Course Name: Diploma in Electrical and Electronics Engineering

Subject Code: 1030 **Semester: V**

Subject Title: PROGRAMMABLE LOGIC CONTROLLER

RATIONALE

Various control operations are to be performed automatically and sequentially on the electrical machines to suit the industrial requirements. Programmable controllers are mainly employed to control the process in industries. In order to impart knowledge on programmable Logic Controller this theory subject is introduced.

OBJECTIVES

Unit :1

After completing this chapter, students should able to:

- Explain the meaning of automation and List the types of automation
- Define PLC and Explain why their use is valuable
- Explain what PLC can do
- Compare fixed and modular PLC
- Explain the advantages of PLC
- Explain the functions of various elements of power supply unit

Unit :2

After completing this chapter, students should able to:

- Know the difference between digital and analog input and output signals
- Observe how digital field device information gets into a PLC
- Observe how analog field device information gets into a PLC
- Understand I/O addresses and how they are used in a PLC

Unit :3

After completing this chapter, students should able to:

- Describe PLC timer instruction and differentiate between a non-retentive and retentive timer
- Program the control of outputs using the timer instruction
- List and describe the functions of PLC counter instructions
- Create PLC programs involving program control instructions, math instructions.

Unit :4

After completing this chapter, students should able to:

- Explain the functionality of different levels of industrial network
- Explain the concept of network topology and network protocols

- Explain the concept of I/O bus networks etc.,

Unit :5

After completing this chapter, students should able to:

- Describe the computer control of process
- Explain the operation of SCADA
- Explain the functions of the major components of a process control system
- Explain how on/off control and PID control works

DETAILED SYLLABUS

33072 - PROGRAMMABLE LOGIC CONTROLLER (M - SCHEME)

Unit –I INTRODUCTION TO PLC **Page No: 5 - 23**

Automation – Types of Automation (manufacturing and Non-Manufacturing) – Advantages of automation - PLC Introduction - Definition - Block diagram of PLC – Principle of operation – Modes of operating – PLC Scan - Hardwire control system compared with PLC system - Advantages and Disadvantages of PLCs – Criteria for selection of suitable PLC – Memory organization – Input Types – Discrete input – Analog inout - Elements of Power supply unit - PLC Types (Fixed I/O and Modular I/O) - List of various PLCs available – Applications of PLC.

Unit – II INPUT/OUTPUT MODULES **Page No: 24 - 45**

The I/O Section - Discrete I/O modules (DC and AC) – Analog I/O modules - Special I/O Modules – I/O Module Specification - Typical Discrete and Analog I/O field Devices – Sensors – Limit switch – Reed switch – Proximity sensor (Inductive and Capacitive) – Types of Photo Electric Sensor - Sinking and Sourcing I/O modules – TTL output module – Relay output module – Isolated output module -Input/Output Addressing scheme in important commercial PLCs..

Unit –III PLC PROGRAMMING **Page No: 46 - 88**

Types of programming methods – Types of programming devices – Logic Functions – AND Logic – OR Logic – NOT Logic - Relay type instructions –Timer Instructions – ON Delay and OFF Delay Timer – Retentive Timer Instruction – Cascading Timers – Counter Instruction – UP Counter – DOWN Counter – UP/DOWN Counter – Cascading Counters – Program Control Instructions – Data Manipulation Instruction – Data Compare Instructions – Math Instructions - Sequencer Instructions - PID Instruction – PWM Function – Simple programs using above instructions.

Develop ladder logic for: Bottle filling system – Automatic car parking system - EB to Generator Changeover system – Batch process – Elevator system - Automatic Star-Delta Starter – Traffic light control.

Unit –IV NETWORKING **Page No: 89 - 106**

Levels of industrial network – Network Topology – Network Protocol – OSI Reference Model - Networking with TCP / IP Protocol - I/O Bus networks – Block diagram of I/O Bus networks – Types of I/O Bus networks - Protocol standards – Advantages of I/O Bus networks - Gateway – Token passing – Data Highway – Serial Communication – Device Net – Control Net – EtherNet – Modbus – Fieldbus – Profibus - Subnetting – Subnet mask - File transfer protocol.

Unit –V DATA ACQUISITION SYSTEMS **Page No: 107 - 121**

Computers in Process control – Types of processes - Structure of control system – ON/OFF Control – Closed loop control - PID Control – Motion Control – Block diagram of Direct Digital Control - Supervisory Control and Data Acquisition (SCADA) – Block diagram of SCADA – Features of SCADA – Functions of SCADA - SCADA software - Data Loggers – Tags – Alarms - landlines for SCADA – use of modems in SCADA..

REFERENCE BOOKS:

#	Name of the Book	Author	Publisher
1.	Control of Electrical Machines	S.K. Bhattacharya	New Age International Publisher, New Delhi
2.	Introduction to Programmable Logic Controllers	Gary Dunning	Thompson Delmer Learning II Edition

UNIT-1 INTRODUCTION TO PLC

OBJECTIVES:

After studying this chapter, the student will be able to:

- Explain the types of automation.
- Explain the different parts and modes of operation of PLC
- Explain the different types of PLC programming device.
- Compare the hard wired logic and PLC system.
- Explain the criteria for selection of PLC.

1.0 Introduction:

The word “Automation” was first used at the Ford motor company in the late 1940. One definition of automation was proposed in 1947 as “the automatic handling of work pieces into, between, and out of machines.

The word ‘Automation’ is derived from Greek words “Auto” (self) and “Matos” (moving). Automation therefore is the mechanism for systems that “move by itself”.

Automation means automatic manufacturing without human control. Automation is a technology concerned with the application of mechanical, electronics and computer based systems to operate and control production.

The objective of automation is to cause the work system to be automatic that is self-acting, self-regulating and self-reliant.

1.1 Automation:

Automation is a set of technologies that results in operation of machines and systems without significant human intervention and achieves performance superior to manual operation.

1.2 Types of automation:

1. Manufacturing automation:
    a. Fixed Automation
    b. Programmable Automation
    c. Flexible Automation
2. Non-Manufacturing automation:
    a. Office Automation
    b. Home Automation
    c. Building Automation

1.2.1 Manufacturing Automation:

This include fixed automation, programmable automation and flexible automation.

1.2.1.1 Fixed Automation:

Fixed automation refers to the use of special purpose equipment to automate a fixed sequence of processing or assembly operations. This system involves automation and integration of various fixed sequences of operation. It is also known as hard automation. It is used to produce product such as gears, nuts and bolts etc., High specialized equipment, called special purpose machine tools are utilized to produce a product very efficiently and at high production rates.

Advantages:

i) Maximum efficiency

ii) Low unit cost

iii) High production rate

Disadvantages:

i) Large initial investment

ii) Inflexible in accommodating product variety.

Applications:

i) Bottling plants

ii) Packaging plants

1.2.1.2 Programmable Automation:

In programmable automation, changes in the sequence of operations is possible by changing the program. The variation in the sequence is achieved by varying the control instructions of the automation system. New programs can be developed and entered to improve the flexibility. Programmable automation is used when the volume of production is relatively low and there are varieties of products to be made. In the Programmable automation products are produced in batches. When one batch is completed, the equipment is reprogrammed to process the next batch.

Advantages:

i) Flexibility to deal with variation and changes in product

ii) Low unit cost for large batches

iii) High production rate

Disadvantages:

i) New products requires long set up time

Applications:

i) CNC Lathe

1.2.1.3 Flexible Automation:

In flexible automation system, different products can be produced on the same equipment in any order or mix. It is also called as Flexible Manufacturing System (FMS). The ability of the system to produce various combination and schedule of products makes this system highly flexible in accommodating the dynamic needs of an industry. There is no lost production time while reprogramming the system and altering the physical setup.

Advantages:

i) Minimum production time losses

ii) Flexibility to deal with products design variations

iii) Customized products.

Disadvantages:

i) Large initial investment

ii) High unit cost compared to fixed and programmable automation

Applications:

i) Industrial Robots

1.2.2 Non-Manufacturing automation:

This include office automation and integrated data processing mechanism, automatic elevators, transportation ticket selling equipments etc.,

1.2.2.1 Office automation:

Office automation refers to the varied computer machinery and software used to digitally create, collect, store, manipulate, and relay office information needed for accomplishing basic tasks. The information may be of many processes and formats-payroll preparation, transportation, reservation, scheduling, banking, security transaction and cost price analysis etc., Increased productivity per office worker is the indeed a major advantages of office automation. Information required for business management is rapidly managed, integrated and sent long distance through office automation.

1.2.2.2 Home automation:

Home automation refers to the automatic and electronic control of household features, activity, and appliances. Home automation systems are composed of hardware, communication and electronic interfaces that work to integrate electrical devices with one another. The three main elements of a home security system are sensors, controllers and actuators. Sensors can monitor changes in daylight, temperature or motion detection; home automation systems can then adjust settings to the preferred levels of a user. Controllers refer to the devices—personal computers, tablets or smartphones—used to send and receive messages about the status of automated features in users' homes. Actuators may be light switches, motors or motorized valves that control a mechanism or function of a home automation system.

1.2.2.3 Building automation:

Building automation is the centralized control of a buildings heating, ventilation, air conditioning, lighting, fire fighting, water distribution and parking systems etc.,. It is controlled by a building management system (BMS) or a building automation system (BAS). These are used in smart buildings. The intelligent system/computers are used to control these systems.

1.3Advantages of automation:

- Increase in productivity
- Reduction in production cost
- Minimization of human fatigue.
- Less floor area required
- Reduced maintenance requirement
- Better working conditions for workers
- Effective control over production process
- Improvement in production quality
- Reduction in accidents
- Uniform components are produced.
- More safety

1.3.1Disadvantages of automation:

- High capital cost
- Increased unemployment
- Failure of one part may affect others

1.3.2 Systems used for automation:

a. CAD – Computer Aided Design
b. CAM – Computer Aided Manufacturing
c. CAE – Computer Aided Engineering
d. CIM – Computer Integrated Manufacturing
e. FMS – Flexible Manufacturing System
f. PC – Personal Computers
g. Robots - Robotics
h. PLC – Programmable Logic Controller
i. SCADA – Supervisory Control And Data Acquisition
j. DCS – Distributes Control System
k. CNC – Computer Numerical Control

1.4 PLC Introduction:

The programmable Logic Controller is an assembly of solid state digital logic elements designed to make logical decision and provide control. PLCs are used for the control and operation of manufacturing process equipment and machinery. The PLC is an industrially hardened computer designed to perform control functions in industrial environment. This means that unlike your personnel computer, the PLC must be capable of operating in high temperature with poor power conditions in dusty, dirty, corrosive atmospheres and withstand shock and vibration. In addition, PLCs are designed to be programmed by individuals who are familiar with motor control circuits. There, most PLC program in a language that resembles ladder diagrams, which makes learning PLCs very easy for most electricians. In addition to switching functions, PLCs can also perform counting, calculations, comparison, processing of analog signals and more.

1.4.1A Historical background of PLC:

- In the late 1960s the American motor car manufacturer General Motors was interested in the application of computers to replace the relay sequencing used in the control of car plants.

- The Hydramatic Division of the General Motors Corporation specified the design criteria for the first programmable controller in 1968. Their primary goal was to eliminate the high costs associated with inflexible, relay controlled systems.

- The first PLC was invented in 1969 by Richard Dick E. Morley, who was the founder of the MODICON Corporation.

- The specifications required a solid-state system with computer flexibility able to:
  - i) Survive in an industrial environment
  - ii) Be easily programmed and maintained by plant engineers and technicians
  - iii) Be reusable

1.5 Definition of PLC:

A programmable logic controller is a digital electronic apparatus with a programmable memory for storing instructions to implement specific functions such as logic, sequencing, timing, counting and arithmetic to control machines and processes.

1.6 Block diagram of PLC :

PLC consists of three basic sections:

1. Central processing unit
2. Input/output Modules
3. Programming Device

1. CPU:

The Central Processing Unit (CPU) Module is the brain of the PLC. The Primary functions are to read inputs, execute the control program, and update outputs.

The CPU consists of following three components:

i) Processor ii) Memory system iii) Power supply

i) Processor:

The processor executes the user program stored in the memory system in the form of ladder diagrams. The processor accepts input data from various sensing devices, executes the stored program from memory and sends appropriate output commands to control devices. It can also perform arithmetic functions, data manipulation and communication between the local I/O, remotely located I/O and other networked PLC.

ii) Memory system:

The memorysystem is the area in the CPU where all the *programs*, are stored and executed by the processor to provide the desired control of field devices.

iii)Power supply:

Power supply is necessary to convert 120V or 240V a.c into the low voltage d.c (+5V & -5V) required for processor and internal power required for the I/O modules. This power supply unit does not supply power for the actual input or output devices. This can be built into the PLC or be an external unit. Common voltage levels required by the PLC are 24Vdc, 120Vac, 220Vac etc.,

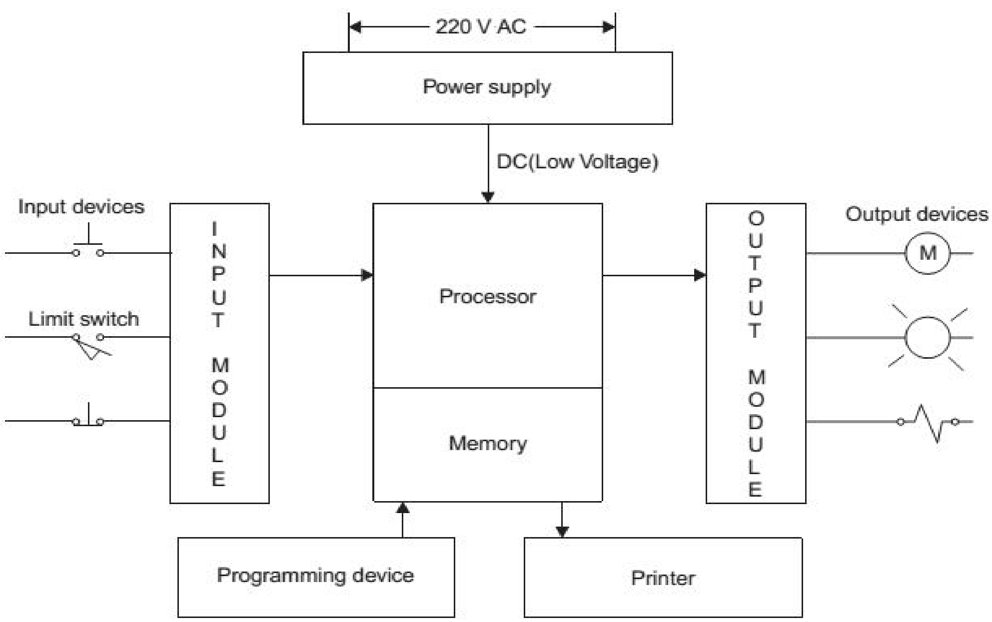


Figure 1.1 Block Diagram of a PLC

2. ***Input/output Section:***

Input modules:

- It senses the presence or absence of an input signal at each of its input terminals.
- It accepts signal from the machine or process and convert them into signals that can be used by the controller.
- The input module provides isolation between the input signal and the PLC.
- The status of input signals are stored in the input image table.

Output Modules:

- It receives the signal from the CPU.
- It converts the controller signals into external signals used to control the machine or process.
- It switches ON or OFF the outputs.
- It provides isolation between CPU and output stage.
- The status of output signals are stored in the output image table.

3. ***Programming device:***

The programming unit allows the engineer or technician to enter and edit the program to be executed. The programming device must be connected through cable to the controller when entering or monitoring the control program.

1.7 Principle of Operation:

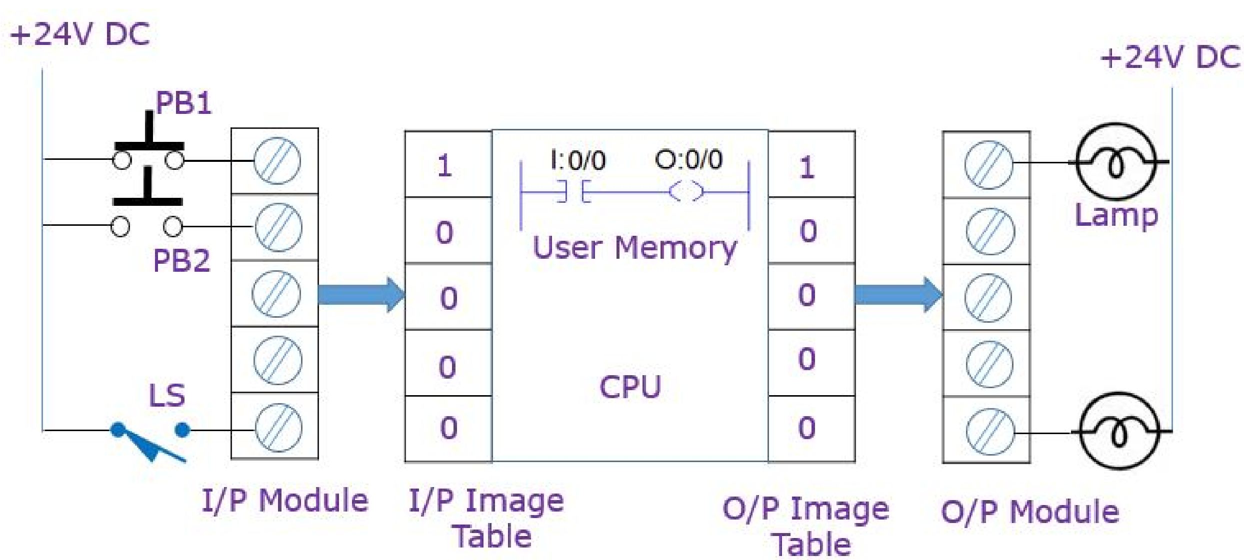


Figure 1.2 (a) Principle of Operation

- ➢ The CPU accepts input signal from sensors like push buttons, limit switches, analog sensors, selector switches, and thumbwheel switches.
- ➢ Stores the status of input in the memory area called input image table.

- Execute the stored user program from memory and sends appropriate output commands to control devices like lamp, motor starters, solenoid valves, pilot lights, and position valves through output image table.
- Update the content of output image table.
- The system power supply provides all the voltages required for the proper operation of the various central processing unit sections.

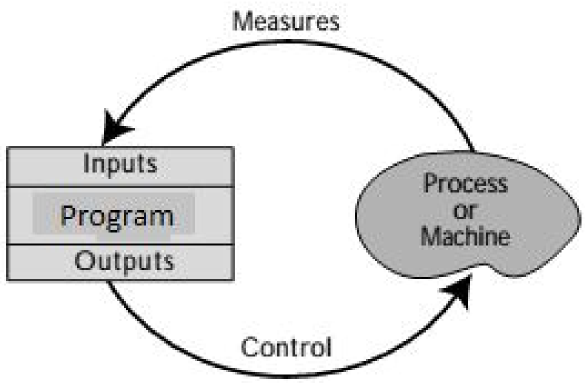


Figure 1.2 (b) Operating Sequence

1.8 Modes of Operation:

Run Mode:

- ✓ Places the processor in the RUN mode.
- ✓ Executes the ladder program and energizes output devices.
- ✓ Prevents online program editing.

PROG Mode:

- ✓ Places the processor in the program mode.
- ✓ Allows you to perform program entry and editing.
- ✓ Prevents the processor from scanning or executing the ladder program.

REM Mode:

- ✓ Places the processor in the Remote mode.
- ✓ Allows you to change the processor mode from a program/operator device.
- ✓ Allows you to perform online program editing.

1.9 PLC Scan:

PLC scan is a sequential process in which the PLC processor performs specific duties on a cyclic basis. During its operation, the CPU completes three processes:

i) **Reads** the input data from the field devices via the input interfaces
ii) **Executes** the control program stored in the memory system
iii) **Writes** the output devices via the output interfaces.

This process of sequentially reading the inputs, executing the program in memory, and updating the outputs is known as **scanning.** Figure 1.3 illustrates a graphic representation of a scan.

In each process cycle, the processor begins by reading or taking a snapshot of the status of the devices connected to input modules and storing the ON/OFF status of each in a special memory called the input image table. After the inputs have been read and stored in the input image table, the processor executes the control program.

During program execution, the processor evaluates the conditions of each statement by looking back at the input image table to see if the input condition are met. If the conditions controlling an output are met, the processor immediately writes a 1 in a special memory called output image table.

The scan time is the specific amount of time required for a PLC to perform both the I/O scan and the program scan. Each PLC's scan time is different. The PLC scan time specification indicates how fast the controller can react to changes in inputs. Scan time varies with program content and length. The time taken to scan the user program is also dependent on the clock frequency of the microprocessor system.

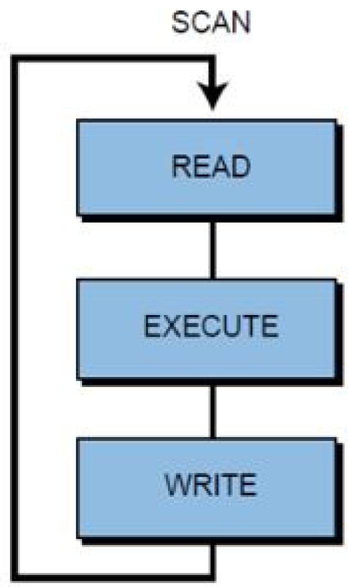


Figure 1.3 PLC Scan

1.10 Comparison between hardwire control system and PLC System:

#	Criterion for comparison	PLC	Hardwired
1	Requirement of Instruction	Instructions are required	Instructions are not required
2	Time required for modification	Less time	More time
3	Memory Requirement	Memory is required for storing the program	Memory is not required
4	Cost	Initial cost is high	Initial cost is low
5	Power Consumption	Less	High
6	Space Required	Small	Large
7	Heat generated	Less	High
8	Technology	Latest	Old
9	Fault identification	Easy	Not easy
10	Error Correction	Easy	Not easy
11	Control circuit	No control circuit. It is controlled by program	Control circuit involves lot of internal wiring

1.11 Advantages of PLC:

The major advantages of PLC circuit over hardwired relay panel circuits are listed below:

1. Flexibility
2. Large quantity of contacts
3. Speed of operation
4. Reliability & Security
5. Documentation
6. Fail safe operation
7. Newer technology
8. Less power consumption
9. Required less floor space
10. Fault location and rectification is easy
11. Logic changes can be reprogrammed
12. Ladder programming method
13. Documentation
14. Pilot running

1.11.1 Disadvantages of PLC:

1. High Cost
2. Need software for programming

1.12 Criteria for selection of suitable PLC:

The process of selecting a PLC is listed below:

1. List the number and types of Input & Output.
2. Memory capacity needed to store the program
3. Speed of processing
4. Communication requirements
5. Special or specific module required
6. Consider safety, reliability and expandability

1.13 Memory Organization;

Figure 1.4 illustrates this memory organization, known as a **memory map**. The memory system is composed of two major sections:

1. System memory
2. Application memory

1. System Memory:

The executive and scratch pad areas are hidden from the user and can be considered a single area of memory that, for our purpose, is called *system memory*.

i) Executive Area:

The executive is a permanently stored collection of programs that are considered part of the system itself. These supervisory programs direct system activities, such as execution of the control program, communication with peripheral devices, and other system housekeeping activities.

ii) Scratch Pad Area:

This is a temporary storage area used by the CPU to store a relatively small amount of data for interim calculations and control. The CPU stores data that is needed quickly in this memory area to avoid the longer access time involved with retrieving data from the main memory.

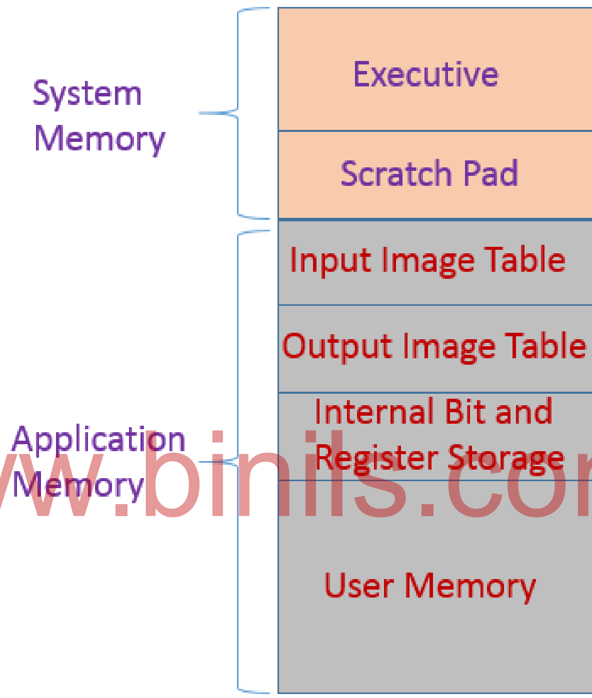


Figure 1.4 Memory Organasation

2. Application Memory:

The application memory stores user programs and any data the processor will use to perform its control functions.

i) Input Image Table:

The portion of memory area used to store the status of input field devices is called input image table. If the input switch is ON, the corresponding bit will be set as 1. During PLC operation, the processor will read the status of each input in the input module and place a value (1 or 0) in the corresponding address in the input image table.

ii) Output Image Table:

The portion of memory area used to store the status of output field devices is called output image table. The output tableis an array of bits that controls the status of digital output devices that are connected to the PLC's output interface. If a bit in the table is turned ON, then the connected output is switched ON.

iii) Internal Bit and Register storage:

The internal bit storage area contains storage bits that are referred to as internal outputs, internal coils, and internal relays. These internals provide an output, for interlocking purposes, of ladder sequences in the control program. The register/word storage area is used to store groups of bits (bytes and words).

iv) User Memory:

This area provides storage for programmed instructions entered by the user. The user program area also stores the control program.

1.13.1 Factors affecting the memory size of PLC:

1. Number of I/Os
2. Size of control program
3. Data collecting requirements
4. Supervisory functions required
5. Future Expansion

1.14 Input Types:

1. Discrete ***2. Analog***

1.14.1Discrete :

PLC can only understand a signal that is ON or OFF. Binary 1 indicates that a signal is present, or the switch is ON. Binary 0 indicates that the signal is not present or the switch is OFF.

The logic concept exists only in two predetermined states. In logic input (digital) systems, these two-state conditions can be thought of as signals that are present or not present, activated or not activated, high or low, on or off, etc.

Here, binary 1 represents the presence of a signal, while binary 0 represents the absence of the signal. In digital systems, these two states are actually represented by two distinct voltage levels, +V and 0V. One voltage is more positive than the other. Often, binary 1 (or logic 1) is referred to as TRUE, ON, or HIGH, while binary 0 (or logic 0) is referred to as FALSE, OFF, or LOW.

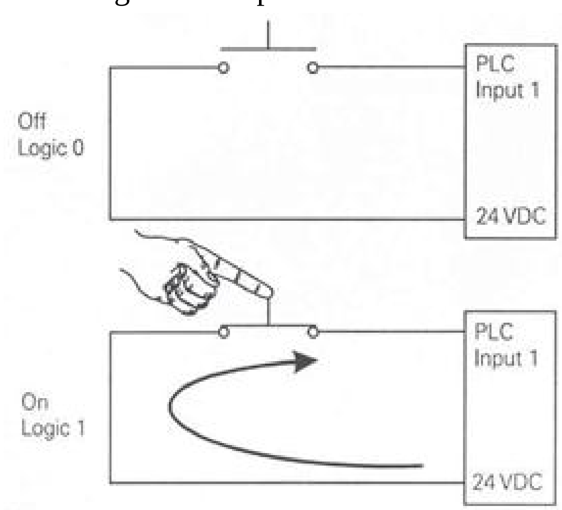


Figure 1.5 Logic Input

Digital Inputs include push-buttons, limit switches, relay contacts, proximity switches, photo sensors (On/Off), pressure switches and more. Digital inputs devices are available in both DC as well as AC and some are voltage independent such as a switch contact.

1.14.2 Analog:

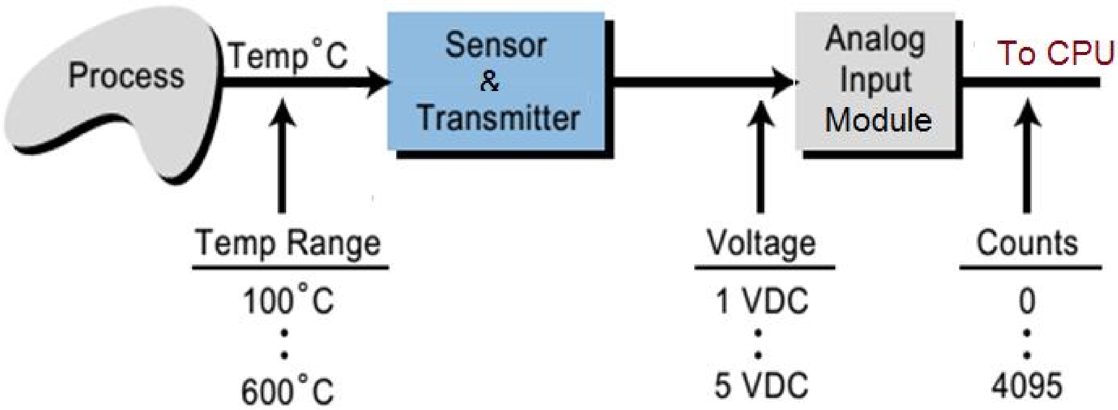


Figure 1.6 Analog Input

An analog input is an input signal that has a continuous signal. Analog input modules, are used in applications where the field device's signal is continuous. Typical analog inputs may vary from 4 to 20mA or 0-10 volts d.c. Analog signals are used to represent changing values such as speed, temperature, weight and level. This analog signal is converted into binary data that can be stored in the data table and used by the PLC when it solves the PLC program.

1.15 Output Types:

i) Logic (Discrete) output

ii) Analog Output

1.15.1 Digital Output:

A discrete output can either turns a device ON or OFF such as lights, LEDs, small motors, and relays. Some examples are motors that need just be ON or OFF, Lighting, solenoid valves, door locks. Digital output modules are available for DC output, AC output or a mix.

Typical digital output devices are:

i) Motor starter coils
ii) Pilot lights
iii) Solenoids
iv) Alarms
v) Control relays
vi) Horns
vii) Start / stop signals to VFD/VSD

1.15.2 Analog Output:

Analog output (AO) are for variable level or range of output between OFF or stopped and ON or full speed as for an electric motor for instance. Examples of analog outputs are a VFD (Variable Frequency Drive), a valve position actuator, and a industrial variable power supply.

The output signal can be divided into 32,767 increments and represented in a 16 bit word. To achieve precision in controlling the valve, the 0 to 10 volt signal will be split into 32,767 steps. Since the output module automatically converts the 16 bit output word to the proper analog value.

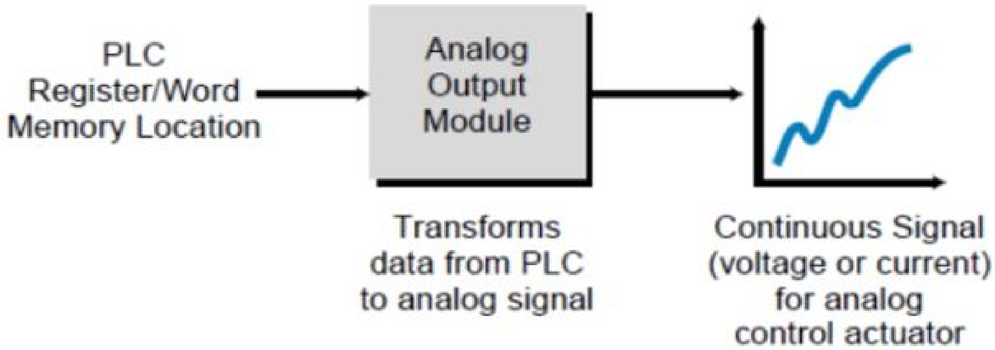


Figure 1.7 Analog Output

Example of analog output could be a valve in a tank filling application. The positioning of the valve can be represented by a voltage level of zero volts when the valve is closed and upto +10V dc when the valve is completely open. Any voltage between 0V dc and 10V dc would represent how far open the valve would be compared to its fully open state. The analog output module transforms a digital bit value from the processor's output status table to an analog output voltage or current level. Inside the analog output module a digital to analog converter (D/A converter) is the solid state device that makes the conversion.

Typical analog output devices are:

i) Analog valves
ii) Actuators
iii) Chart recorders
iv) Electric motor drives
v) Analog meters
vi) Pressure Transmitter

Typical analog signals are:

i) 0 to 10V d.c
ii) -10 to +10V d.c
iii) 0 to 5v d.c
iv) 1 to 5 v d.c
v) 0 to 20mA
vi) -20 to +20mA
vii) 4 to 20mA

1.16 Elements of Power Supply Unit:

Usually, PLC power supplies require input from an AC power source; however, some PLCs will accept a DC power source.

I/O Module need : 120 VAC or 220 VAC or 24 VDC

CPU need : +5V and -5V

Line Conditioner:

This unit is used to purify the AC wave form. The input waveform is normally a perfect sine wave, but it can be distorted at times by two external factors.

1. Distortion due to switching operation
2. Harmonics due to load

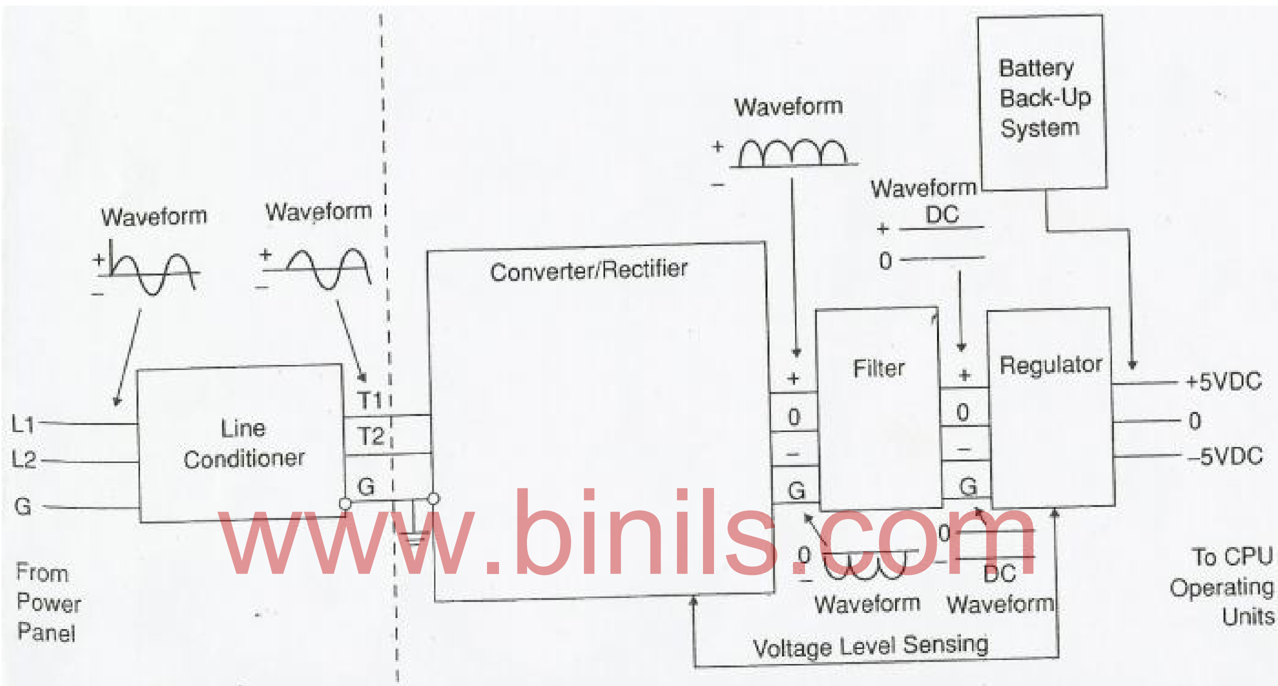


Figure 1.8 Block Diagram of Power Supply Unit

Step down transformer: The 230 volt a.c voltage is connected to the primary of the transformer. Then the transformer step down the a.c voltage, to the level required for the desired d.c output.

Converter /rectifier: An electronic circuit used for converting a.c voltage into unidirectional voltage is called as converter/rectifier. It changes the bidirectional a.c into a pulsating, unidirectional dc waveform. Internally a transformer step-down the voltage to an appropriate level. Then bridge rectifier produce pulsating dc output.

Filter:

The filter consists of internal circuitry, including capacitors and resistors or inductors. This circuit is used to reduce the ripple content in the pulsating d.c and tries to make it smoother.

Regulator:

It keeps the output voltage constant under variable load conditions as well. Thus the voltage regulator keeps the voltages as or near +5v & -5V levels regardless of load.

1.17 PLC Types (Fixed and Modular) :

1.17.1 Fixed I/O PLC:

This PLC has the sections like input module, CPU and associated memory, power supply and output module. These sections are built into one self-contained unit. Fixed PLCs are also referred as "Shoebox" or "brick" by manufacturers due to their shape and size. In this type the number of inputs and outputs cannot be expanded.

Example : Allen Bradley Micrologic 1000 has 6 discrete inputs and 4 discrete outputs.

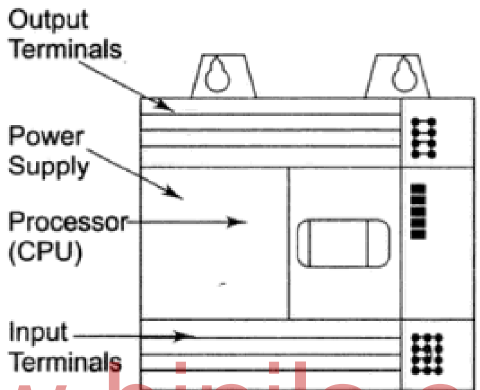


Figure 1.9 Fixed I/O PLC

Advantages:

i) Less scanning time
ii) Low cost
iii) Less trouble shooting time

Disadvantages:

i) Limited number of inputs and outputs
ii) Not suited for future expansion
iii) If part of the unit fails, need to replace entire unit.

1.17.2 Modular I/O PLC:

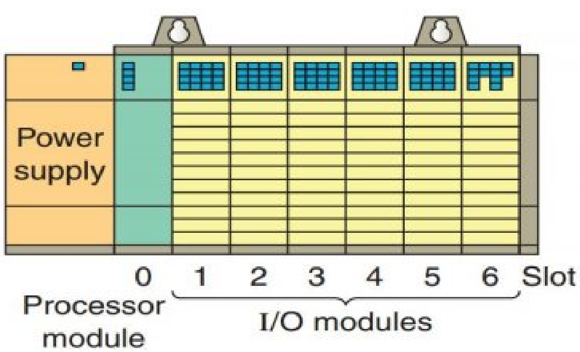


Figure 1.10 Modular I/O PLC

A modular PLC is built with several components that are plugged into a common rack with extendable I/O capabilities. It consists of a rack, power supply, CPU and I/O modules. On a rack these modules are fixed as separate hardware items. Modular PLCs are further divided into small, medium and large PLCs based on the program memory size and the number of I/O features.

Example: Siemens: S7-300 and S7-400

Allen Bradley: SLC 5/01, SLC 5/02 etc.,

1.18 List of various Brands of PLCs available:

1. SIEMENS
2. Allen Bradley
3. Omron
4. GE Fanuc
5. MODICON
6. Keyence
7. Mitsubishi
8. Toshiba
9. Honeywell
10. Festo Corp.

1.19 Applications of PLCS:

- Conveyor Systems,
- Food Processing Machinery,
- Bottle filling (water, soft drinks, canned food)]
- Auto Assembly (e.g. automobile industry)
- Fluid Level Control (e.g. water tanks)
- Mixing Fluids (paint industries)

UNIT – I INTRODUCTION TO PLC

MODEL QUESTIONS

Two Mark Questions:

1. Define the term automation.
2. What are the advantages of fixed automation?
3. What are the advantages of programmable automation?
4. What are the advantages of flexible automation?
5. Give any two example for office automation.
6. Give any two example for home automation.
7. Give any two example for building automation.
8. State any two advantages of automation.
9. What are the major parts of a PLC?
10. What are the different operating modes of a PLC?
11. What is the use of input image table?
12. What is the use of output image table?
13. List the types of input module.
14. What is the function of input module?
15. List the types of output module.
16. What is the function of output module?
17. State the advantages of fixed PLC.
18. What is a modular PLC?
19. List the name of any two market available PLC.

Three Mark Questions:

1. What are the types of automation?
2. List the advantages of automation.
3. Explain about fixed automation.
4. Explain about programmable automation.
5. Explain about flexible automation.
6. What do you know about office automation?
7. What do you know about home automation?
8. What do you know about building automation?
9. Define PLC
10. Write short notes on modes of operation of PLC
11. Write short notes on PLC scan.
12. Compare PLC circuit versus hardwired circuits. (any 3)
13. Write short note on logic input in PLC.
14. Write short note on analog input in PLC.
15. Write short note on discrete output in PLC.
16. Write short note on analog output in PLC.
17. List the criteria for selection of suitable PLC for particular application?

18. List the name of any six market available PLC.
19. List any three advantages of PLC?
20. List out few applications of PLC.

Ten Mark Questions:

1. Draw and explain the block diagram of PLC.
2. With neat sketch explain the principle of operation of PLC.
3. Draw and explain the block diagram of memory organization.
4. With neat sketch explain about PLC scan.
5. With neat sketch explain the construction of modular PLC's
6. Compare hardwire control system versus PLC system.
7. Write brief notes on logic input and analog input.
8. Draw and explain the power supply unit of PLC

UNIT- 2 INPUT/OUTPUT MODULES

OBJECTIVES:

After studying this chapter, the student will be able to:

- Know the difference between digital and analog input and output signals
- Observe how digital field device information gets into a PLC
- Observe how analog field device information gets into a PLC
- Understand I/O addresses and how they are used in a PLC

2.1 I/O Section:

The input/output (I/O) section of a PLC is the section to which all field devices are connected and provides the interface between them and the CPU. Input interface modules accept signals from the machine or process devices and convert them into signals that can be used by the controller. Output interface modules convert controller signals into external signals used to control the machine or process. Input/output arrangements are built into a fixed PLC while modular types use external I/O modules that plug into the PLC.

2.1 Discrete I/O Module:

The most common type of I/O interface module is the discrete type. This type of interface connects field input devices of the ON/OFF nature such as selector switches, pushbuttons, and limit switches. Likewise, output control is limited to devices such as lights, relays, solenoids, and motor starters that require simple ON/OFF These are used to connect the sensors that are of digital in nature, i.e., only for switch ON and OFF purpose. These modules are available on both AC and DC voltages and currents with variable number of digital inputs.

2.1.1 DC Discrete Input Module:

Figure 2.1 shows a circuit diagram of a typical DC Discrete input module.

i) Power section:

A DC input module interfaces with field input devices that provide a DC output voltage. The difference between a DC input interface and an AC/DC input interface is that the DC input does not contain a bridge circuit, since it does not convert an AC signal to a DC signal. The input voltage range of a DC input module varies between 5 and 30 VDC.

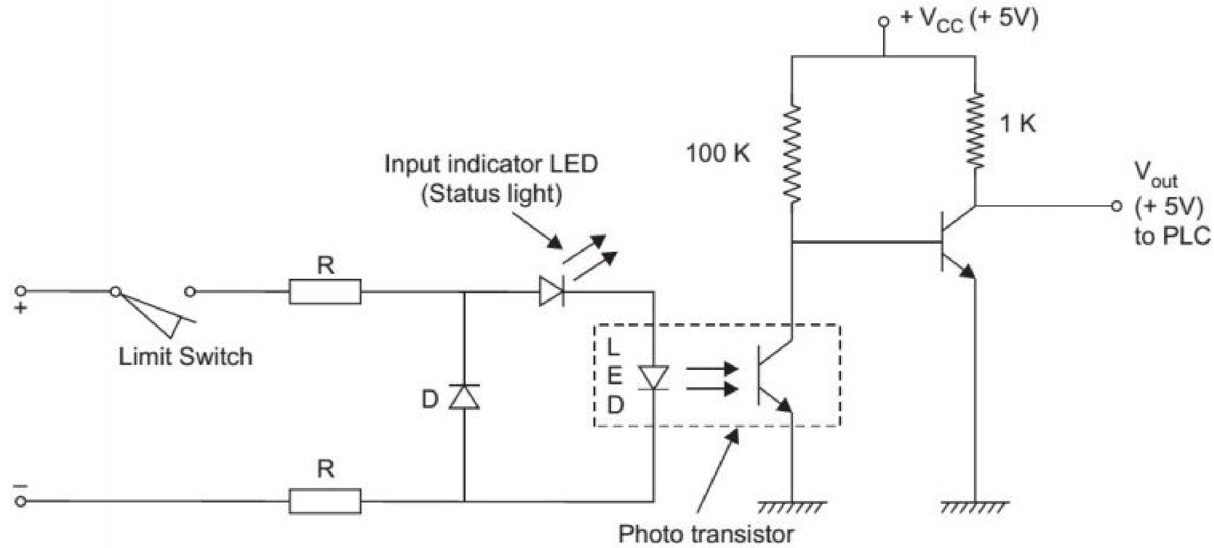


Figure 2.1 Circuit Diagram of DC Discrete Input Module

ii) Isolator section:

After the detection of a valid signal, it passes the signal to logic section through an isolation circuit. The isolator circuit is usually made up of an optical isolator (called as opto- coupler). It is used to isolate the logic circuit from power circuit.

This optical isolation:

1. Helps to reduce the effect of electrical noise
2. Prevents damage to the processor due to line voltage transient.

ii) Logic Section:

DC signals from the opto-coupler are used by the logic section to pass the input signal to the CPU. The LED in the logic circuit is used to indicate the presence of a logic 1 signal in the logic section.

LED glow - input signal is present
LED not glow - input signal is not present

2.1.2 AC Discrete Input Module:

Figure 2.2 (a) and (b) shows a schematic and circuit diagram of a typical AC Discrete input module.

It has three primary parts:

i) Power section *ii) Isolator section* *iii) Logic section*

i) Power Section:

The bridge rectifier circuit of the power section converts the incoming AC signal to a DC-level signal. It then passes the signal through a filter circuit, which protects the signal against bouncing and electrical noise on the input power line. Threshold circuit detects if the incoming signal has reached or exceeded a predetermined value for a predetermined time.

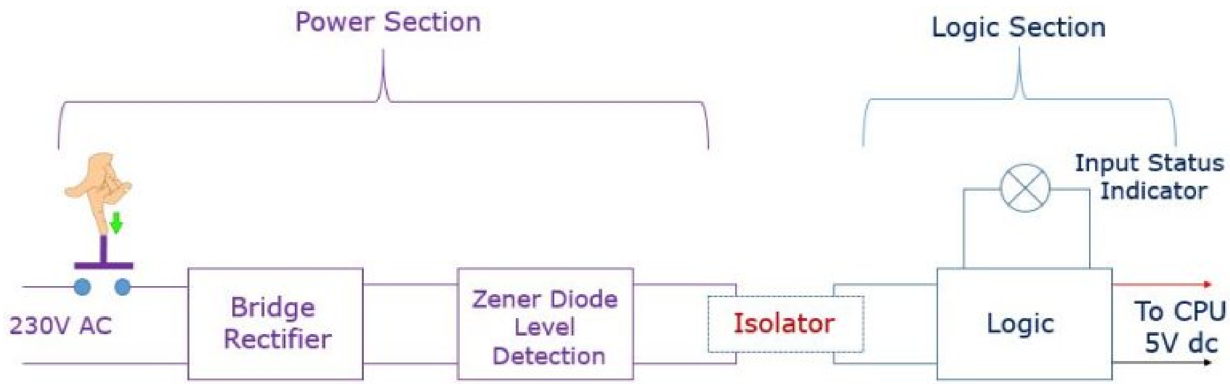


Figure 2.2 (a) Schematic Diagram of AC Discrete Input Module

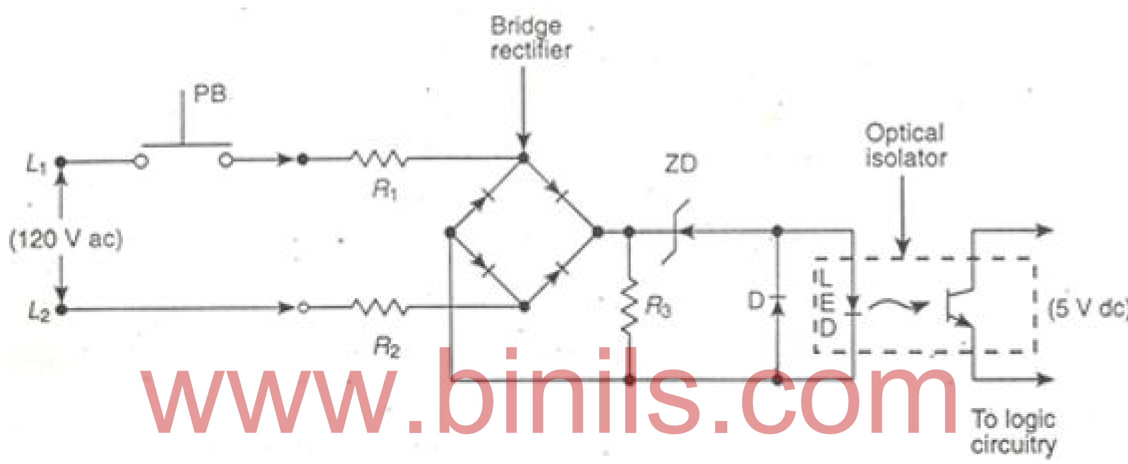


Figure 2.2 (b) Circuit Diagram of AC Discrete Input Module

ii) Isolator section:

After the detection of a valid signal, it passes the signal to logic section through an isolation circuit. The isolator circuit is usually made up of an optical isolator (called as opto- coupler). It is used to isolate the logic circuit from power circuit.

This optical isolation helps to reduce the effect of electrical noise and Prevents damage to the processor due to line voltage transient.

ii) Logic Section:

DC signals from the opto-coupler are used by the logic section to pass the input signal to the CPU. The LED in the logic circuit is used to indicate the presence of a logic 1 signal in the logic section.

LED glow - input signal is present

LED not glow - input signal is not present.

2.1.3 D.C Discrete Output module:

This block diagram and circuit of D.C Output module is shown in figure 2.3(a) and (b). This output module consists primarily of the logic and power sections, coupled by an isolation circuit.

i) Logic Section:

During normal operation, the processor sends an output's status, according to the logic program, to the module's logic circuit. If the output is to be energized, the logic section of the module will latch, or maintain, a 1. This sends an ON signal through the isolation circuit, which in turn, switches the voltage to the field device through the power section of the module. The LED in the logic circuit is used to indicate the presence of a logic 1 signal in the logic section.

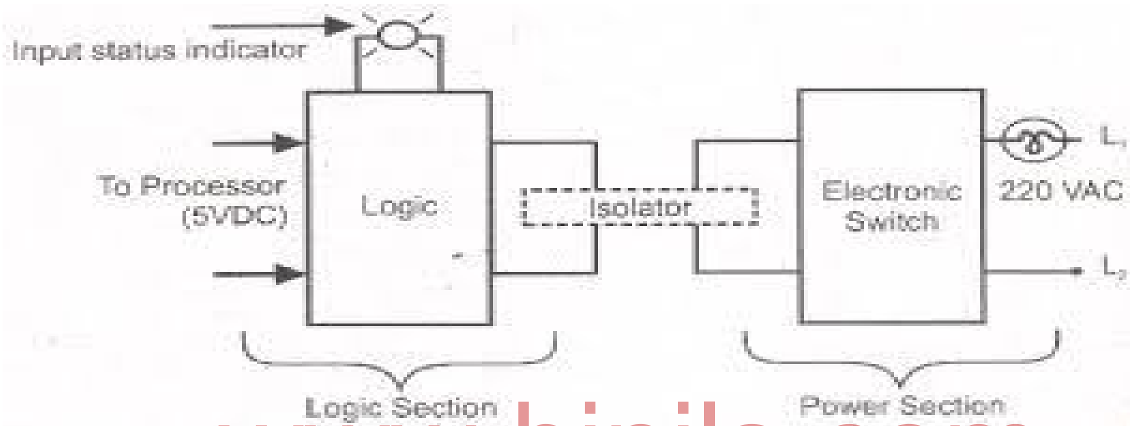


Figure 2.3 (a) Schematic Diagram of DC Discrete Output Module

ii) Isolator section:

After the detection of a signal from CPU, it passes the signal to power section through an isolation circuit. The isolator circuit is usually made up of an optical isolator (called as opto-coupler). It is used to isolate the logic circuit from power circuit. This electrical separation helps prevent large voltage spikes from damaging either the logic side of the interface or the PLC.

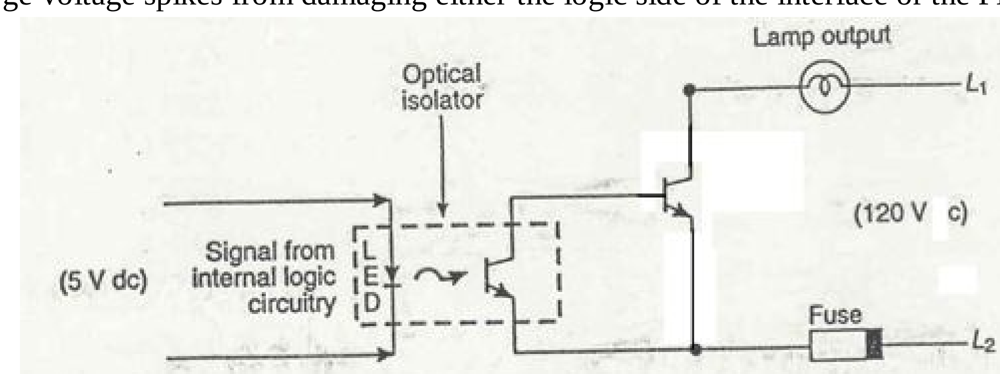


Figure 2.3 (b) Circuit Diagram of DC Discrete Output Module

iii) Power section:

The switching circuit in the power section of a DC output module uses a power transistor to switch ON the load. The AC switch is normally protected by a freewheeling diode across the load. DC outputs may also incorporate a fuse to protect the transistor during moderate overloads.

2.1.4 A.C Discrete Output Module:

Figure 2.4 (a) and (b) shows a schematic and circuit diagram of A.C Discrete output Module. The circuit consists of logic and power sections, coupled by an isolation circuit.

i) Logic Section:

During normal operation, the processor sends an output's status, according to the logic program, to the module's logic circuit. If the output is to be energized, the logic section of the module will latch, or maintain, a 1. This sends an ON signal through the isolation circuit, which in turn, switches the voltage to the field device through the power section of the module. The LED in the logic circuit is used to indicate the presence of a logic 1 signal in the logic section.

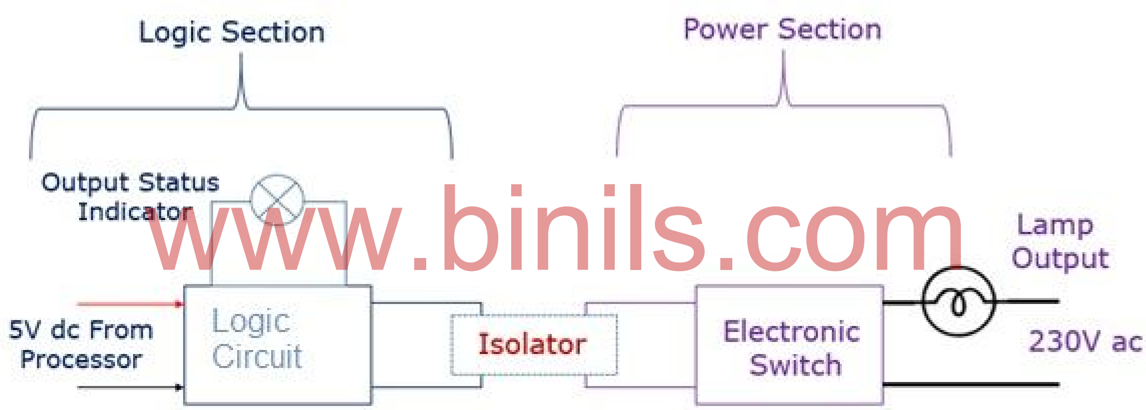


Figure 2.4 (a) Schematic Diagram of AC Discrete Output Module

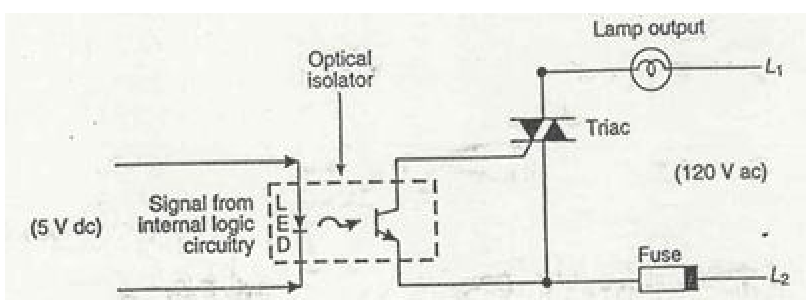


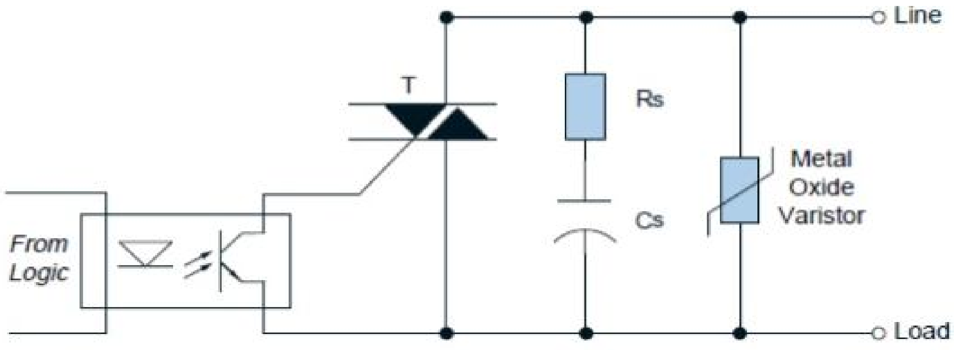
Figure 2.4 (b) Circuit Diagram of AC Discrete Output Module

ii) Isolator section:

After the detection of a signal from CPU, it passes the signal to power section through an isolation circuit. The isolator circuit is usually made up of an optical isolator (called as opto-coupler). It is used to isolate the logic circuit from power circuit. This electrical separation helps prevent large voltage spikes from damaging either the logic side of the interface or the PLC.

iii) Power section:

The switching circuit in the power section of an AC output module uses either a TRIAC or SCR to switch power. The AC switch is normally protected by an RC snubber or a metal oxide varistor (MOV). This snubber and MOV circuits prevent electrical noise from affecting the circuit operation. Furthermore, an AC output circuit may contain a fuse that prevents excessive current from damaging the switch.



2.2 Analog I/O Module:

2.2.1 Analog input module:

The Analog input module is used to convert analog signal from analog devices, such as analog sensors, temperature probes, pressure indicator etc., to equivalent digital values using analog to digital converter. The analog input signal is usually a varying voltage in the range of 0-10V or current in the range of 4-20mA. The transformed analog value is the digital equivalent of the analog input signal. Each converter value is stored in the memory in a digital form, typically as a 16-bit word for internal processing. They provide optical isolation for electrical noise protection.

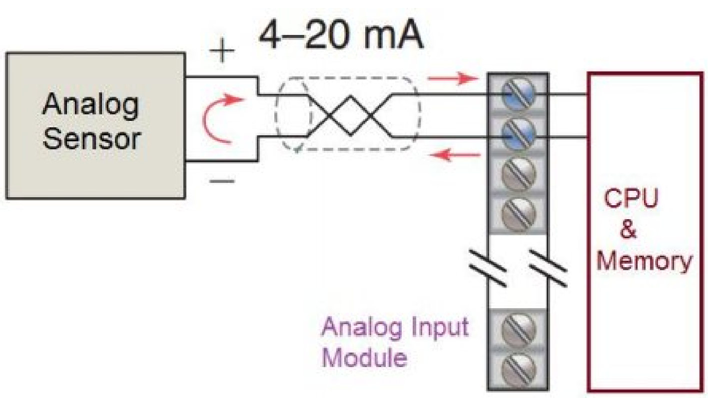


Figure 2.5 Schematic diagram of Analog Input Module

2.2.2 Analog Output Module:

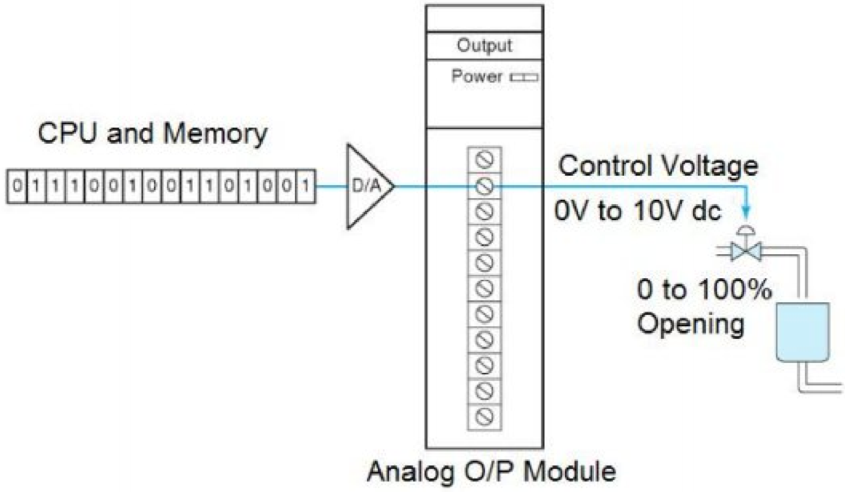


Figure 2.6 Schematic Diagram of Analog Output Module

The analog output modules converts processed digital values from the CPU into equivalent analog signals, typically in the range of 0 -10V or 4 -20mA, to operate analog output devices using digital to analog converter.This analog output value is proportional to the digital numerical value received by the module.

Analog output modules are selected to send out either a varying current or voltage signal. An analog output could send a 4 to 20mA signal to a variable speed drive. The drive will control the speed of a motor in proportion to the analog signal received from analog output module.

Example of analog output field devices: Analog valves, Actuators, Chart recorders, Electric motor drives, analog meters and pressure transducers.

2.3 Special I/O Module:

#	Special Input Module	Description
1	**Basic Module**	A basic module is a specialized module that allows the PLC to interface to serial peripheral devices. Basic modules usually reside in the local chassis, close to the CPU
2	**Communication module:**	
	i) ASCII I/O MODULE	ASCII I/O modules allow the interfacing of bar code readers, meters, printers and data terminals to a PLC.
	ii) RS-232C Interface Module	It is used to connect a PLC to telephone lines using a modem. By using this facility operator can modify or edit program from remote place.

3	**High Speed Counter Module**	It is used to count pulses from sensors, encoders and switches at very high speeds. When input pulses come in faster rate a high speed counter module is used.
4	**Remote I/O Subscanner**	It is used to scan the status of I/O and relieves the burden of the CPU.
5	**RTD Module**	A RTD input module interfaces a PLC to RTD temperature-sensing elements and other types of resistance input devices such as potentiometers.
6	**Stepper Motor Control Module**	It provides a digital output pulse train for stepper motor application.
7	**Thermocouple/Millivolt Module**	It converts inputs from various thermocouple or millivolt devices into values that can be input and stored into PLC data tables.
8	**TTL Module**	It allows the transmitting and receiving of TTL signals for communication with the PLC processor.
9	**Thumbwheel Module**	It allows the uses of thumbwheel switches for feeding information in parallel to the PLC to be used in the control program. It information is usually in BCD form.
10	**PID Module**	It is used in process control applications that incorporate PID algorithm. It prevents the CPU from being burdened with complex calculations.

2.4 I/O Module Specification:

2.4.1 Input Module Specification:

#	Term	Description
1	**Nominal Input Voltage**	It specifies the magnitude (e.g., 5 V, 24 V, 230 V) and type (AC or DC) of user-supplied voltage that a module is designed to accept.
2	**Input Threshold Voltages**	It specifies two values: i) a minimum ON-state voltage that is the minimumvoltage at which logic 1 is recognized as absolutely ON; ii) a maximum OFF-state voltage which is the voltage at which logic 0 is recognized as absolutely OFF.

3	**Ambient Temperature Rating**	It specifies what the maximum temperature of the air surrounding the I/O modules should be for best operating conditions.
4	**Output Voltage**	It specifies the magnitude (e.g., 5V, 115 V, 230 V) and type (AC or DC) of user-supplied voltageat which a discrete output module is designed to operate.
5	**Output Current**	These values specify the maximum current that a singleoutput and the module as a whole can safely carry under load (at rated voltage).
6	**Inrush Current**	An inrush current is a momentary surge of current that an AC or DC output circuit encounters when energizing inductive, capacitive, or filament loads.
7	**Short Circuit Protection**	It designate whether the particular module's design has individual protection for each circuit or if fuse protection is provided for groups (e.g., 4 or 8) of outputs.
8	**Leakage Current**	This value specifies the amount of current still conducting through an output circuit even after the output has been turned off.

2.4.2 Output Module Specification:

#	Term	Description
1	**Channels per module**	Circuits on analog I/O modules are often referred to as channels. These modules normally have 4, 8, or 16 channels.
2	**Input current/voltage range**	These are the voltage or current signal ranges that an analog input module is designed to accept.
3	**Output current/voltage range**	It defines the current or voltage signal ranges that a particular analog output module is designed to output under program control.
4	**Input protection**	Analog input circuits are usually protected against accidentally connecting a voltage that exceeds the specified input voltage range.
5	**Resolution**	The resolution of an analog I/O module specifies how accurately an analog value can be represented digitally.
6	**Input impedance and capacitance**	For analog I/Os, these values must be matched to the external device connected to the module.

7	**Common-Mode Rejection**	It refers to an analog module's ability to prevent noise from interfering with data integrity on a single channel and from channel to channel on the module. It is normally expressed in decibels or as a ratio.

2.5.1 Typical Discrete Input Field Devices:

- Circuit breakers contacts
- Level switches
- Limit switches
- Motor starter contacts
- Photoelectric eyes
- Proximity switches
- Push buttons
- Relay contacts
- Selector switches
- Toggle switch
- Thumbwheel switches (TWS)

2.5.2 Typical discrete output field devices:

- Alarms
- Control relays
- Fans
- Horns
- Lights
- Motor starters
- Solenoids
- Valves

2.6 Sensor:

Sensors are devices used to provide information on the presence or absence of an object. Sensors are connected to the input of a PLC. A pushbutton is one example of a sensor that is connected to the PLC input. An electrical signal is sent from the pushbutton to the PLC indicating the condition (open/closed) of the pushbutton contacts.

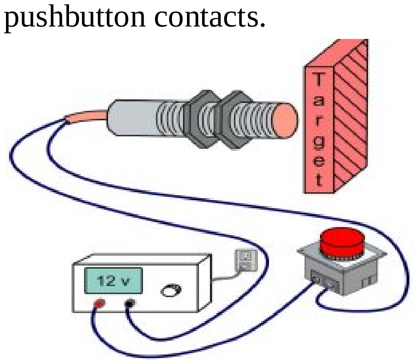


Figure 2.7 Proximity Sensor

Hence sensor allows a PLC to detect the state of a process. Logical sensors can only detect a state that is either true or false.

2.7 Limit Switch:

The term *limit switch* is used for a switch which is used to detect the presence or passage of a moving object. A limit switch will change its output from NO to close or NC to open when an object is physically touching the switch. Its contacts are available in several configurations. They may be normally open (NO), normally closed (NC), or a combination of NO & NC contacts.

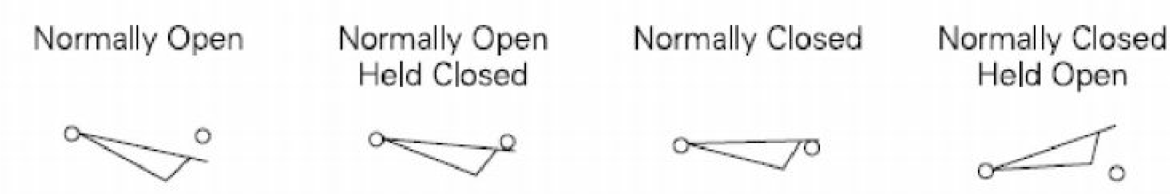


Figure 2.8Symbol of Limit Switch

A typical limit switch consists of a switch body and an operating head. The switch body includes electrical contacts to energize and de-energize a circuit. The operating head incorporates some type of lever arm or roller or cam, referred to as an actuator.

When no external force is applied the actuator position is called as free position. When external force is applied or moving object is detected the actuator moves from free position to operating position. In the operating position the contacts of the limit switch change from their normal state (NO or NC) to their operated state.

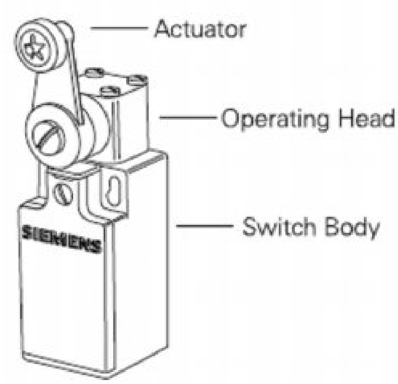


Figure 2.8Limit Switch

2.8 Reed Switches:

Reed switches are very similar to relays, except a permanent magnet is used instead of a wire coil. It consists of two overlapping, but not touching strips (contacts) sealed in a glass or plastic envelop.

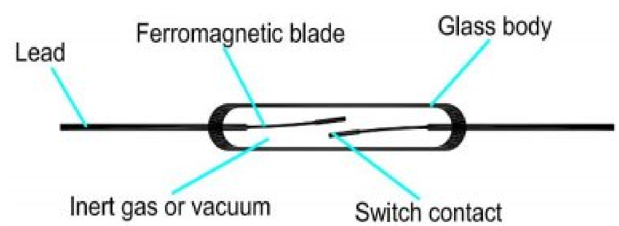


Figure 2.9 Hermetically sealed reed switch

When a magnet is brought close to the switch, the strips become magnetized and attract each other. As the permanent magnet is moved further away, the contact tab ends are demagnetised and return to their original positions. The magnetic reed switch is also inertia free. Because of the sealed, they are unaffected by dust, humidity, and fumes. When the switch contact is operated by an electromagnet, it is known as reed relay. Such a switch is widely used with burglar alarms.

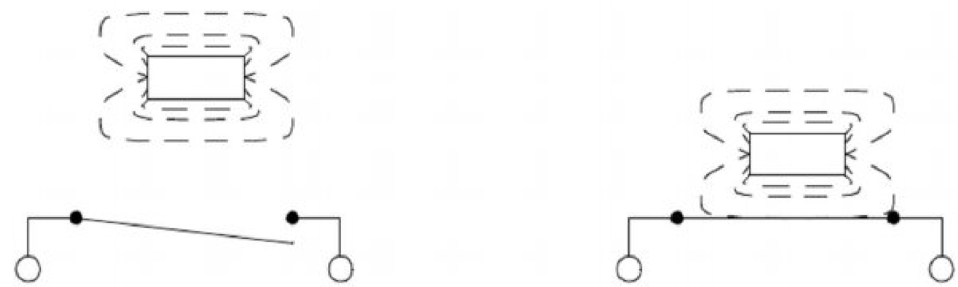


Figure 2.10 Working of Reed switch

2.9 Proximity Sensor:

Proximity sensors are discrete sensors that sense when an object has come near to the sensor face.

Types of Proximity sensor:

1. Inductive proximity sensor
2. Capacitive proximity sensor

2.9.1 Inductive Proximity Sensor:

Inductive proximity sensor operates on the principle that the inductance of a coil and the power losses in the coil vary as a metallic object is passed near it. Because of this principle, inductive proximity sensors are only used for sensing metal object. They will not work with non-metallic materials.

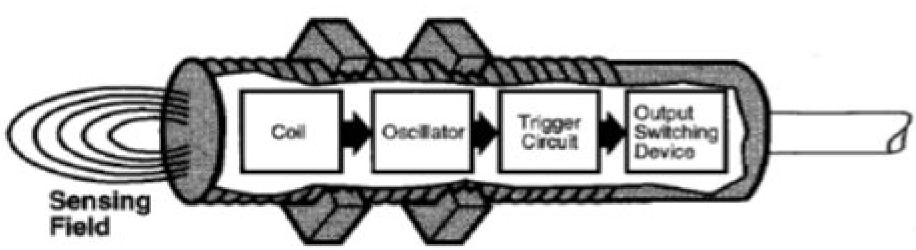


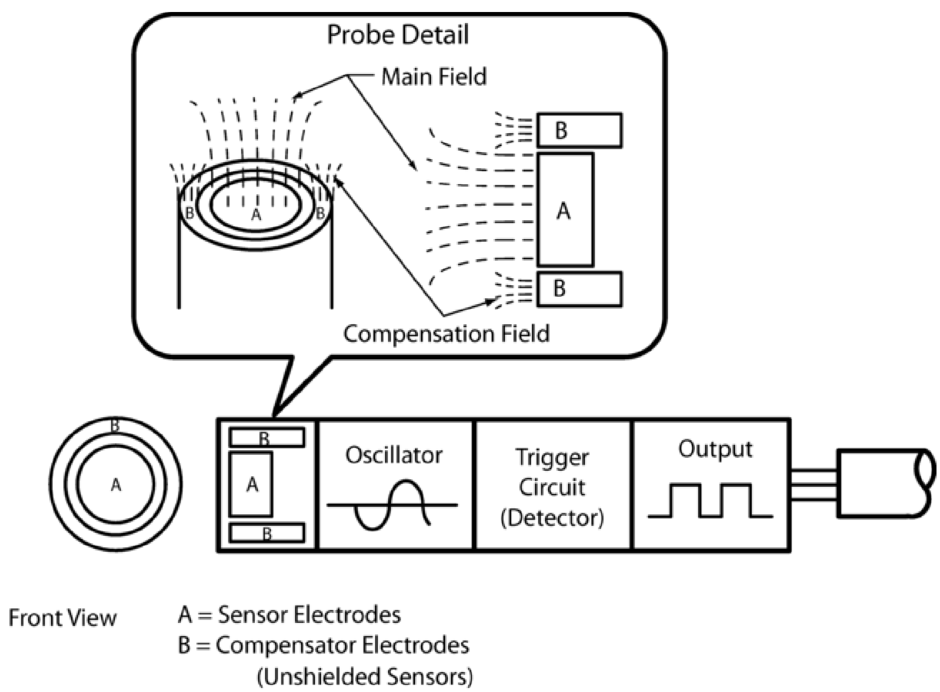
Figure 2.11 Block Diagram of Inductive Proximity Sensor

The block diagram of inductive proximity sensor is shown in figure. It consists of tuned oscillator, triggering circuit and output switching element. When the oscillator operates, there is an alternating magnetic field produced by the coil. This magnetic field radiates through the face of the sensor. The oscillator circuit is tuned such that as long as the sensing field senses non-magnetic material it will continue to oscillate, it will trigger the trigger circuit, and the output switching device will be off (the output contains inverter). The sensor will therefore send an OFF signal through the cable.

When metallic object comes near the face of the sensor, the alternating magnetic field in the target produces circulating eddy currents are a power loss. As the target moves nearer, the eddy current loss increases, which loads the output of oscillator. This loading effect causes the output amplitude of the oscillator to decrease and causes the output switching device to switch ON.

2.9.2 Capacitive Proximity Sensor:

Capacitive sensing is a noncontact technology suitable for detecting metals, nonmetals, solids, and liquids, although it is best suited for nonmetallic targets because of its characteristics and cost relative to inductive proximity sensors. In most applications with metallic targets, inductive sensing is preferred because it is both a reliable and a more affordabletechnology.



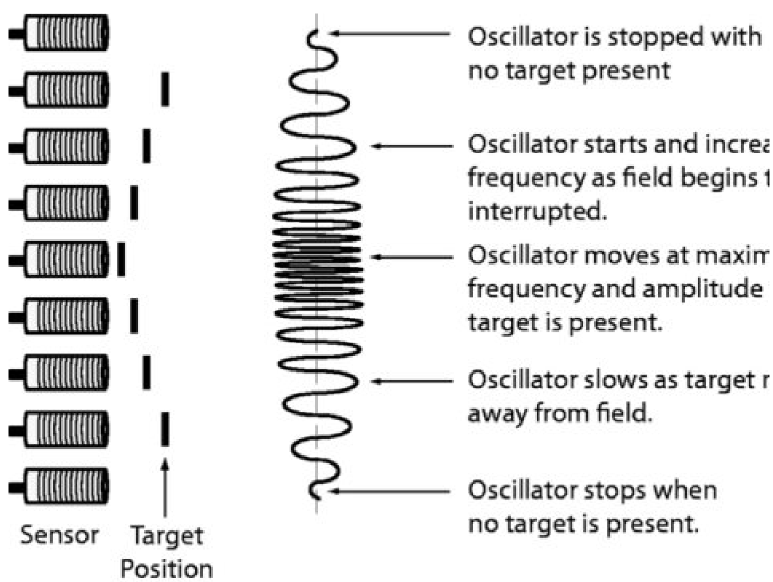
Capacitive sensor components

The sensor consists of four basic components:

- A capacitive probe or plate
- An oscillator
- A signal level detector
- A solid-state output switching device
- An adjustment potentiometer

Capacitive proximity sensors are similar in size, shape, and concept to inductive proximity sensors. However, unlike inductive sensors which use induced magnetic fields to sense objects, capacitive proximity generate an electrostatic field and reacts to changes in capacitance caused when a target enters the electrostatic field.

When the target is outside the electrostatic field, the oscillator is inactive. As the target approaches, a capacitive coupling develops between the target and the capacitive probe. When the capacitance reaches a specified threshold, the oscillator is activated, triggering the output circuit to switch states between ON and OFF.



Capacitive proximity operation

The ability of the sensor to detect the target is determined by the target's size, dielectric constant and distance from the sensor. The larger the target's size, the stronger the capacitive coupling between the probe and the target. Materials with higher dielectric constants are easier to detect than those with lower values. The shorter the distance between target and probe, the stronger the capacitive coupling between the probe and the target.

2.10 Photo Electric Sensor:

A photoelectric sensor is a device that detects a change in light intensity. Typically, this means either non-detection or detection of the sensor's emitted light source. Photoelectric sensors are made up of a light source (LED), a receiver (phototransistor), a signal converter, and an amplifier. The phototransistor analyzes incoming light, verifies that it is from the LED, and appropriately triggers an output. The emitter and detector are positioned so that an object will block or reflect a beam when present

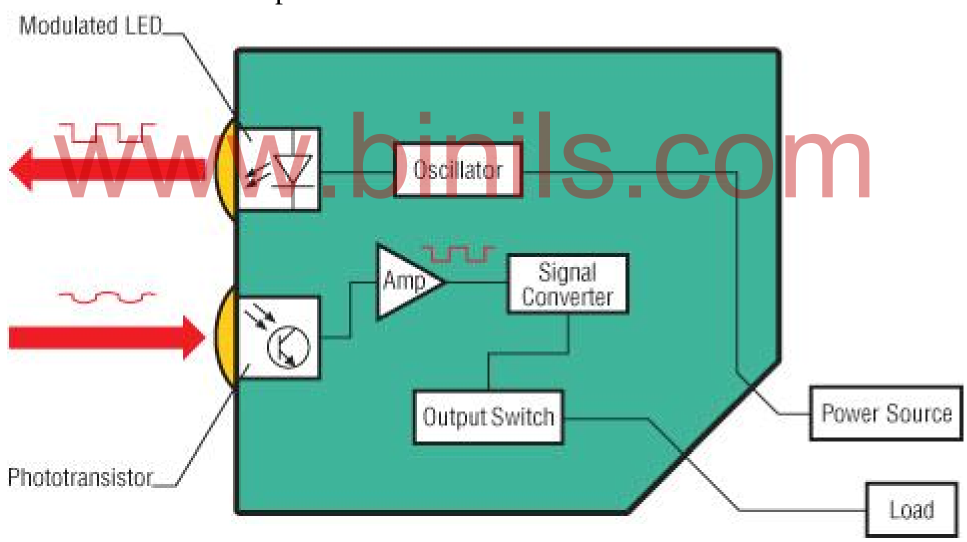


Figure 2.13 Block Diagram of Photo electric sensor

Operation:

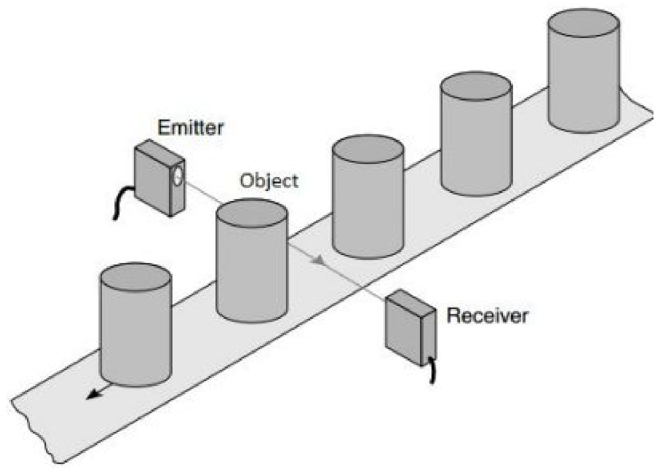
In the figure the light beam is generated on the top, focused through a lens. At the detector side the beam is focused on the detector with a second lens. If the beam is broken the detector will indicate an object is present. The oscillating light wave is used so that the sensor can filter out normal light in the room. The light from the emitter is turned on and off at a set frequency. When the detector receives the light it checks to make sure that it is at the same frequency. If light is being received at the right frequency then the beam is not broken. The frequency of oscillation is in the KHz range, and too fast to be noticed. When the beam is broken the part will be detected.

Types of photo electric Sensor:

i) Thru-beam Sensor
ii) Retro reflective sensor
iii) Diffuse mode sensor

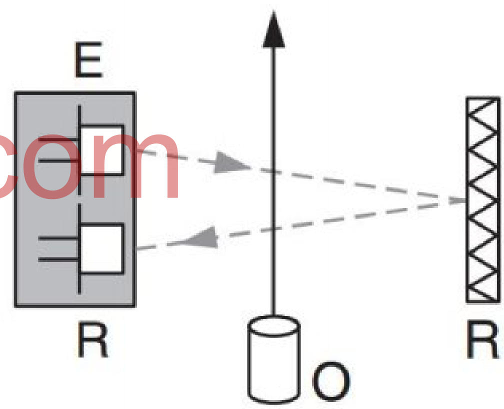
2.10.1 Thru-beam Sensor:

The transmitter and receiver of the thru-beam sensor are housed in different cases that are separated from each other. The Emitter (E) transmits directly to the Receiver (R). If an object (O) interrupts the light beam, the receiver voltage drops and the switching function is initiated.



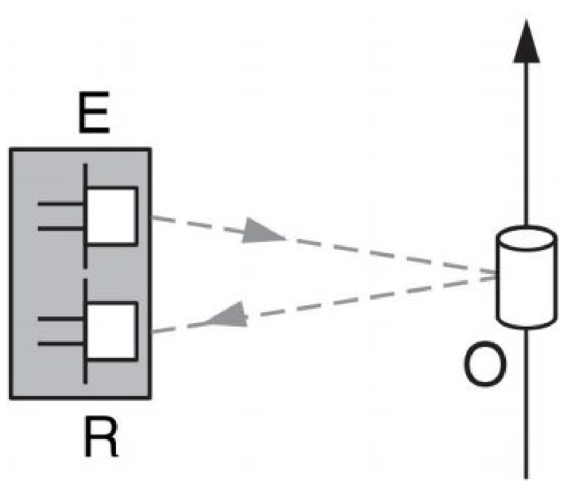
2.10.2 Retro-Reflective sensor:

The retro-reflective sensor contains the emitter and receiver in a single housing. The light from the transmitter is beamed back from a reflector to the receiver. If the beam of light is interrupted, the switching function is initiated.



2.10.3 Diffuse Mode Sensor:

The structure of the diffuse mode sensor is based on the same principle as a retro reflective sensor. In this sensor, the light reflected from the recorded object is evaluated by the receiver.



2.11 Sinking and Sourcing:

Sinking and sourcing references are terms used to describe a current flow relationship between field input and input module.

2.11.1 Input module sourcing:

Figure shown below is an example of a sourcing application. The positive potential is connected to the input module and the negative potential is connected to the input device. Using conventional current flow (*positive to negative*) it is said that the input module is the source of supply for the real world input device. Now the input device receives current from the input module. If an input module *provides* current, it is said to be sourcing.

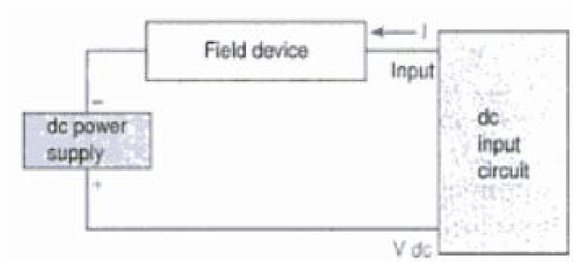


Fig: Input Module Sourcing

2.11.2 Input module sinking:

Figure shown below is an example of a sinking application. In this configuration, the positive potential is connected to the input device and the negative potential is connected to the input module. In this case, using a conventional (positive to negative) current flow, the input device is said to be providing current to the input module. If the input module is receiving current, it is said to be sinking.

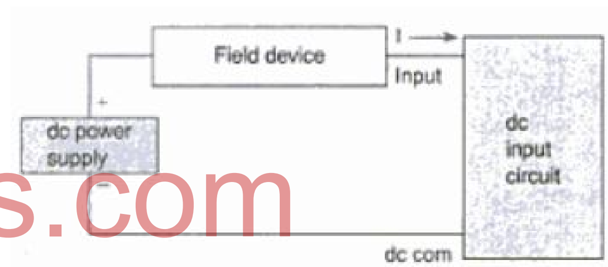


Fig: Input Module Sinking

2.11.3 Output module sourcing:

Figure shown below is an example of a sourcing application. The positive potential is connected to the output module and the negative potential is connected to the output device. Using conventional current flow (*positive to negative*) it is said that the output module is the source of supply for the real world output device. Now the output device receives current from the output module. If an output module *provides* current, it is said to be sourcing.

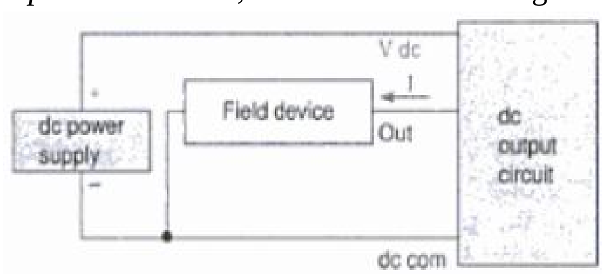


Fig: Output Module Sourcing

2.11.4 Output module sinking:

Figure shown below is an example of a sinking application. In this configuration, the positive potential is connected to the output device and the negative potential is connected to the output module. In this case, using a conventional (*positive to negative*) current flow, the output device is said to be providing current to the output module. If the output module is receiving current, it is said to be sinking.

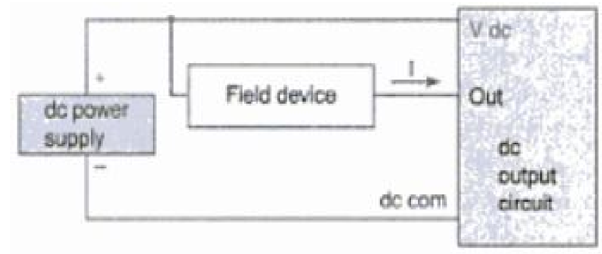


Fig: Output Module Sourcing

As a general rule, sinking modules are used with output modules when interfacing with electronic equipment (TTL or CMOS), while sourcing modules are used for dc load such as solenoids.

2.12 TTL Output Module:

Transistor-Transistor Logic (TTL) output modules switch 5V dc signals. A TTL output module allows for interface between the PLC and TTL comparable devices. An example of a TTL interface would be interfacing a PLC to various 5V dc field devices including ICs and Seven segment displays.

TTL modules usually have eight available output terminals; however, high-density TTL modules may be connected to as many as sixteen devices at a time. A TTL output interface requires an external power supply.

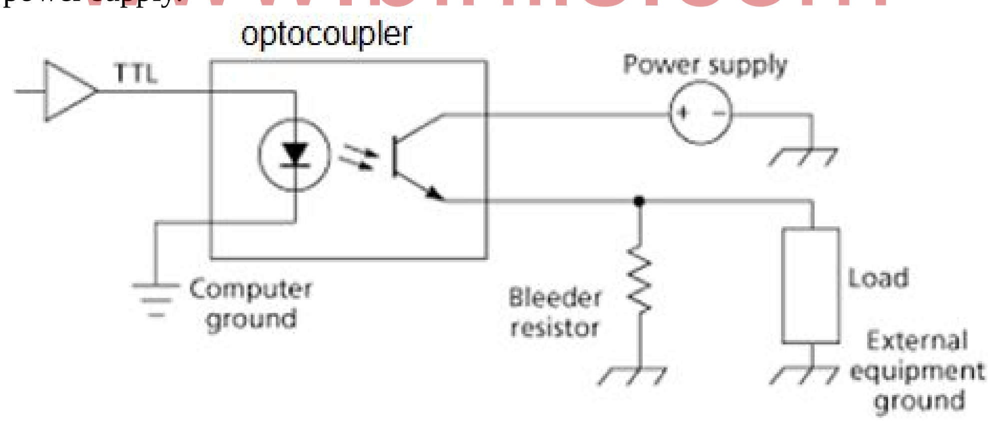


Fig: TTL Output Module

Digital output modules take TTL-level signals from a measurement system to control an SSR's output. A bleeder resistor assures that the load will turn off when the SSR's output achieves a high-impedance state.

2.13 Relay Output:

Relay output modules are also known as contact outputs or dry contact outputs. The relay output modules are used to switch a.c or d.c loads at low voltages and low current.

Relay contacts are described as three main arrangements or forms. The three arrangements are FORM A, FORM B and FORM C. A FORM A relay contact is a single pole normally open contact. The FORM B relay contact is a single pole normally closed contact which is similar to a single normally closed switch. The FORM C relay contact is a single pole double throw contact.

PLC output units are available with all three contact arrangements but typically FORM A and FORM C are used. Relay outputs are also available with a common terminal and as isolated contacts.

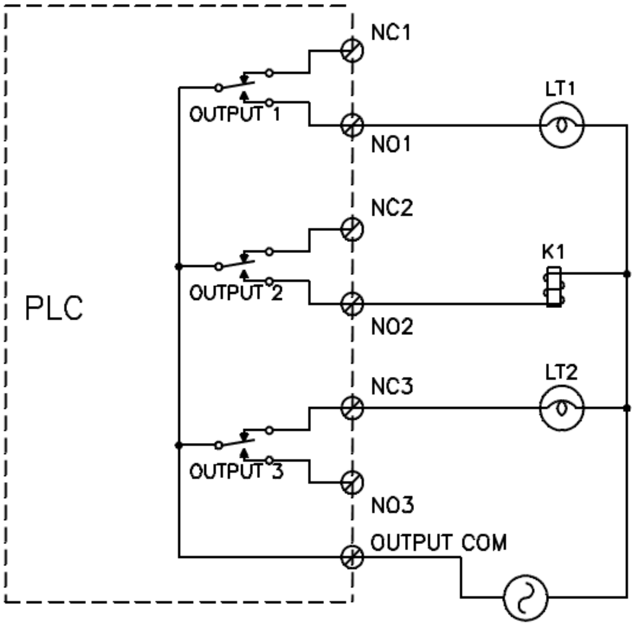


Fig: FORM C Relay Output Module

An output unit with three FORM C contacts having a common terminal is shown in Figure. Note in this figure that the common terminal of each of the three relays is connected to one common terminal of the output unit labeled OUTPUT COM.

2.14 Isolated Output module:

Isolated AC and DC outputs modules allows the interface to control output devices powered by different sources, which may also be at different ground levels. For isolated output module extra terminals are necessary for the independent return lines. Figure shows a typical system output wiring diagram using an output unit having three FORM C isolated outputs. In this type output unit, the relay contacts have no connection between them. Each output has three terminals. The C terminal is the common terminal of the relay.

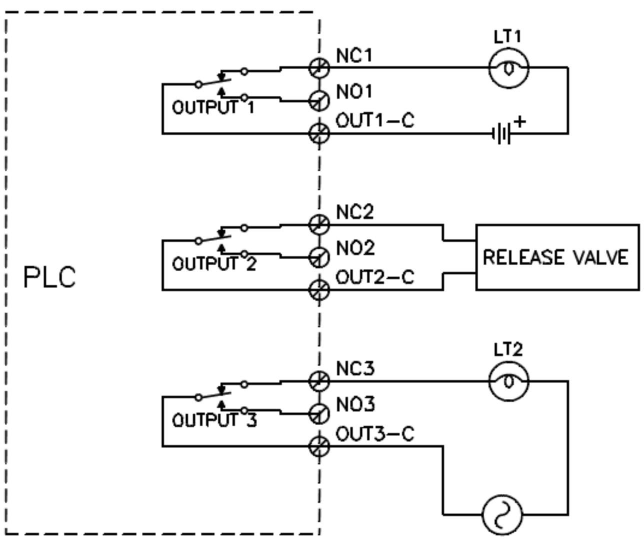
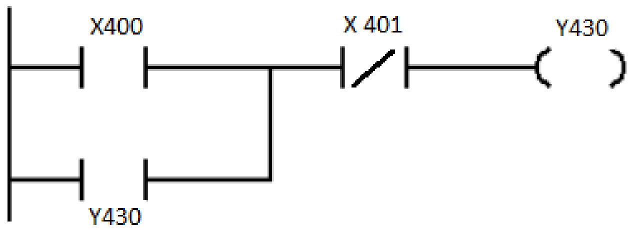


Fig: FORM C Isolated Relay Output Module

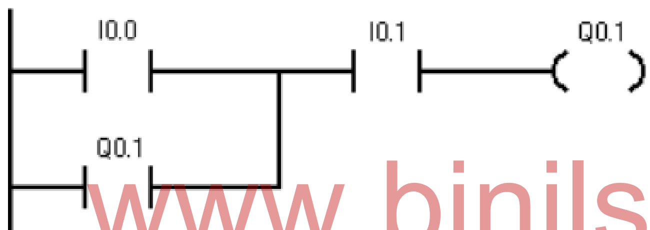
2.15 I/O ADDRESSING SCHEME IN IMPORTANT PLCs

1. Mitsubishi PLC:
    a. Input Address: X400, X401, X402 etc Where, X - Input
    b. Output address: Y430, Y431, Y432 etc., Y - Output



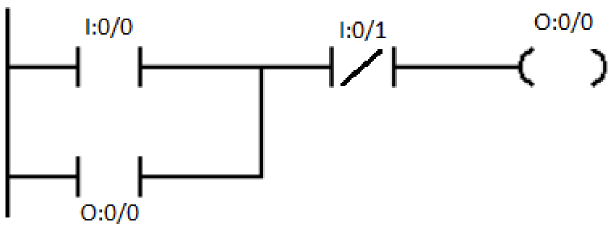
2. Siemens PLC:
   Input: I0.0, I0.1, I0.2 etc., Where, I - Input
   Output: Q0.0, Q0.1, Q0.2 etc., Q - Output

   Example:



3. Allen Bradley PLC: X:XXX/XX

   Example:
   Input: I:0/0, I:0/1, I:0/2
   Output: O:0/0, O:0/1, O:0/2



UNIT- II: INPUT AND OUTPUT MODULES

PART- A: Two Mark Questions

1 List the name of two types of I/O module.

2 Give example for discrete inputs.

3 What is sinking input module?

4 What is sourcing input module?

5 List the name of any two special input modules.

6 Give example for analog inputs.

7 What are the types of photo electric sensor?

8 Suggest suitable type of proximity sensor for sensing non-metal object.

9 Assign the address for NO and NC contacts of Siemens PLC.

10 Assign the address for NO and NC contacts of Allen Bradley PLC.

11 Assign the address for NO and NC contacts of Mitsubishi PLC.

12 Expand the term TTL.

13 Suggest the suitable type of output module for D.C load only.

14 Suggest the suitable type of output module for A.C load only.

15 Suggest the suitable type of output module for both A.C and D.C loads.

16 What is an FORM A relay output?

17 What is an FORM B relay output?

18 What is an FORM C relay output?

19 What is an isolated output module?

20 What is the voltage and current rating normally used with analog I/O module?

21 Assign the address for output coil instruction of Mitsubishi PLC.

22 Assign the address for output coil instruction of AB PLC.

23 Assign the address for output coil instruction of SIEMENS PLC.

PART- B: 3 Mark Questions

1 Draw the block diagram of D.C discrete input module.

2 What are the 5 main sections of an A.C input module?

3 List the advantages of relay output over solid state output.

4 List the name of special input modules.

5 What are the different types of sensors used in automation industry using PLC

6 What is the function of limit switch?

7 What is the function of reed switch?

8 What is photo electric sensor?

9 List out various input and output elements connected to PLC.

10 What is the necessity of TTL output module?
11 Explain where you would use an isolated I/O module.
12 Name any three analog outputs.

PART-C: 10 Marks Questions

1 With neat sketch explain the operation of A.C discrete input module.
2 With neat sketch explain the operation of D.C discrete input module.
3 With neat sketch explain about sinking and sourcing input module.
4 Explain the working of limit switch and reed switch with neat sketch.
5 Briefly explain about different types of photo electric sensor.
6 With neat sketch explain the operation of inductive type proximity sensor.
7 With neat sketch explain about sinking and sourcing output module.
8 With neat sketch explain the operation of A.C discrete output module.
9 With neat sketch explain the operation of D.C discrete output module.
10 With neat sketch explain the operation of relay type output module.
11 With neat sketch explain the operation of transistor type output module.
12 With neat sketch explain the operation of TRIAC type output module.
13 Briefly explain about surge suppression in PLC output.
14 With suitable example explain about analog output.

UNIT-3 PLC PROGRAMMING

OBJECTIVES:

After studying this chapter, the student will be able to:

- Explain the different types of PLC programming methods.
- Develop ladder logic using relay type instruction, Timer and counter instructions.
- Develop ladder logic diagram for Bottle filling system, Automatic car parking system, EB to Generator Changeover system, Batch process, Elevator system, Automatic Star-Delta Starter and Traffic light control etc.,

3.1 Types of Programming Methods:

The IEC (International Electrotechnical Commission) has created a standard (ICE 1131-3) for five programming languages for PLC. These five languages are known as:

1. Function Block Diagram (FBD)
2. Instruction List (IL)
3. Ladder diagram (LD)
4. Sequential Function Chart (SFC)
5. Structured Text (ST)

3.1.1 Functional Block Diagram: (FBD)

It is a graphical language for depicting signal and data flows through functional blocks. A FBD program is constructed using function blocks that are connected together to define the data exchange. This programming language is a graphic language that uses a library functions in combination with custom functions to create programs. The inputs and outputs of function blocks can be inverted.

3.1.2 Instruction List (IL):

It is a low level ‘assembler like’ language using text. Whenever the rung is started, it must use a rung code. Use LD to indicate the rung is starting with NO contacts and LDI to indicate the rung is starting with NC contacts. All rungs must end with an OUT i.e output. It is best suited for small applications and fast execution.

3.1.3 Ladder Diagram (LD):

Ladder programming has evolved from the wiring diagrams that are used in the car industry for describing the relay control schemes. This method is easy to understand by people who are familiar with simple electronic or electrical circuits. Also it is well accepted by electrician an d plant technician. Faults can be quickly traced is the advantage of this method. The ladder symbols and facilities vary between different PLC products. It has limited facilities for building complex sequences.

3.1.4 Sequential Function Charts (SFC):

It is a graphical language for depicting sequential behavior of a control system. It is used for defining control sequences that are time and event driven. While providing structure and coordination of sequential events, alternative and parallel sequences are supported as well.It containsFlowchart of steps and transitions.

3.1.5 Structured Text (ST):

It is a high level textual language that encourages structured programming. It has a language structure (syntax) that strongly resembles PASCAL. ST is an excellent language for complex processes or calculations that are not graphic friendly.

5.2 Ladder Diagram:

Ladder diagram are very similar to ladder schematics. A ladder diagram is a symbolic representation of an electrical circuit.

A very commonly used method of programming PLCs is based on the use of ladder diagrams. Writing a program is then equivalent to drawing a switching circuit. The ladder diagram consists of two vertical lines representing the power rails. Circuits are connected as horizontal lines, i.e. the rungs of the ladder, between these two verticals.

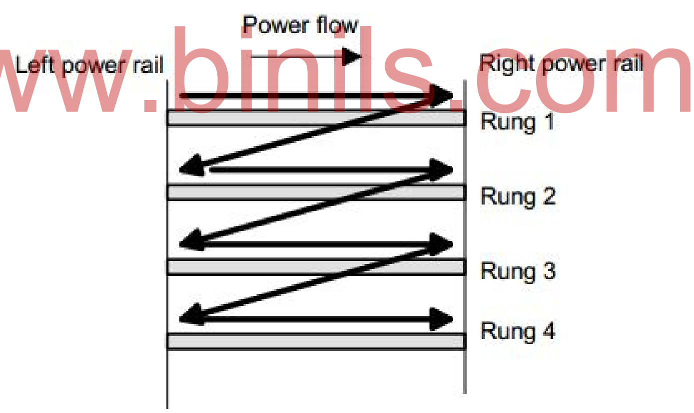


Figure: Scanning the ladder diagram

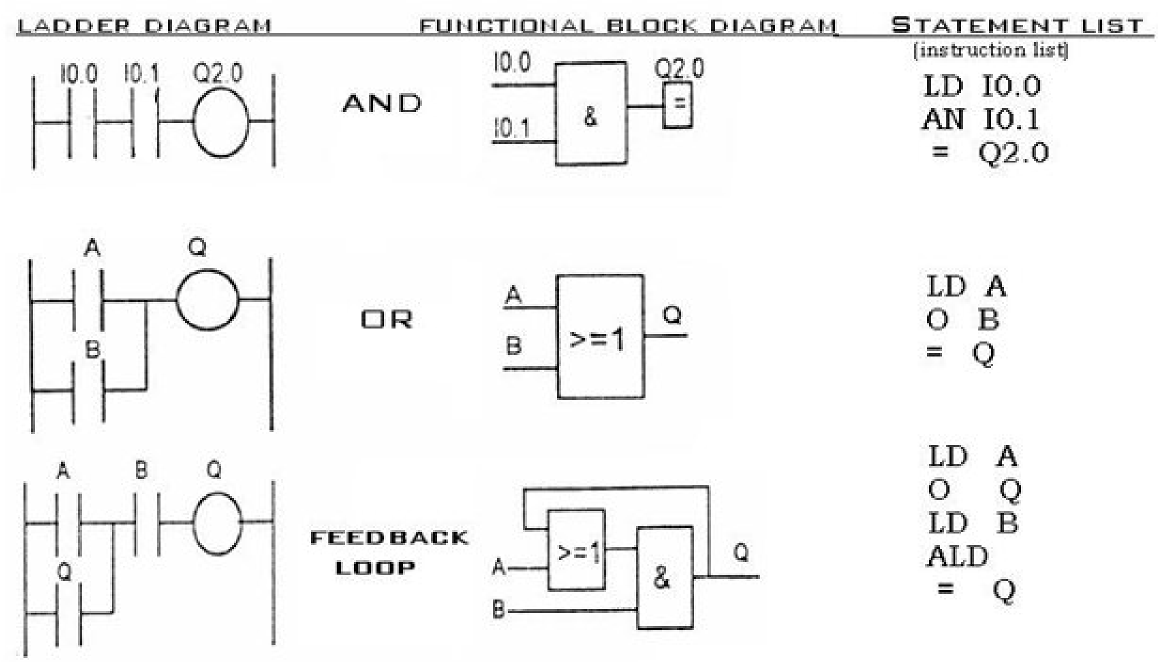
Rules for ladder diagram:

i) The vertical lines of the diagram represent the power rails between which circuits are connected.
ii) Horizontal lines represent rung of a ladder.
iii) Input devices are connected towards left rail.
iv) Output devices are connected towards right rail.
v) A ladder diagram is read from left to right and from top to bottom. The top rung is read from left to right. Then the second rung down is read from left to right and so on.
vi) Each rung must start with an input or inputs and must end with at least one output.

The inputs and outputs are all identified by their addresses, the notation used depending on the PLC manufacturer.

Comparison:

Example program: (For SIEMENS PLC)



3.2 Types of Programming Device:

Programming device:

- ✓ A Programming device is needed to enter, modify, monitor and troubleshoot the PLC program.
- ✓ Once the program has been entered and the PLC is running, the programming device may be disconnected.

Types of programming Device:

1. Hand held Instrument
2. Dedicated Desktop programmer
3. Personal Computer

3.2.1 Hand held Instrument:

A hand held programmers are smaller, cheaper and more portable. This unit contain multicolored, multifunction keys, and LCD or LED display window. The keys are used entering and editing the instruction, navigation keys for moving around the program. It has minimum display capability. It is well suited for parameter changes in the user program.

Advantages of hand held instrument:

1. Easy transfer of PLC program
2. Low cost
3. Easy to use and easy to learn
4. Compact size
5. Works with industrial environment

Figure 4.11 Handheld Instrument

Disadvantages of hand held instrument:

1. Limited ladders can be displayed
2. Documentation not displayed
3. Program stored in the memory will lost when battery is failure.
4. Need more keystroke to enter the program
5. Take more time to enter a big program
6. Modification of program is difficult.

3.2.2 Dedicated Desktop programmer:

It is designed for programming and monitoring the PLC. They are not capable of performing other computer functions. It consists of a keyboard, Video Display Terminal (VDT) and necessary electronic circuits and memory unit. Most dedicated programmer keyboards have electrical symbol keys for NO, NC, Timers etc., VDT is used to give visual display of the program.

Advantages of Dedicated desktop programmer:

1. Portable and withstand the mechanical shock
2. It can works with industrial environment
3. Not affected by electrical noise, high temperature and humidity
4. Easy for electrician or technician to enter or modify the program.

Figure 4.12 Dedicated Desktop of Industrial Computer

3.2.4 Personal Computer:

With software available for all major brands of PLCs, the PC is now most common programming device.

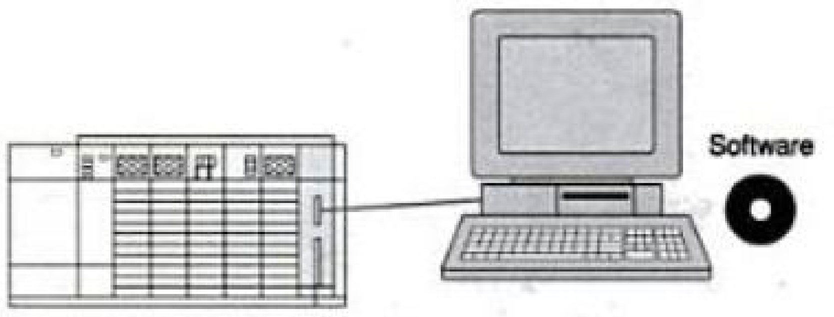


Figure 4.13 Personal Computer (PC)

Advantages of PC programmer:

1. It can display multiple rungs
2. Easy for trouble shooting operation
3. Program can stored on the computer hard disk
4. Program can be stored in floppy, pen drive etc.,
5. Running comments, symbols and other related text can be displayed.
6. Data table can be easily monitored.
7. Easy editing and modifying the program

Disadvantages of PC programmer:

1. PC is not designed for industrial environment
2. It is affected by electrical noise, temperature and humidity

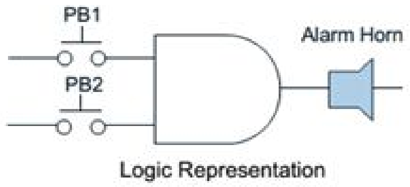
3.3 Logic Function:

3.3.1 Logic:

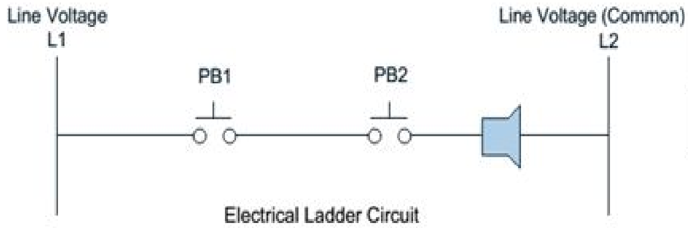
- ✓ Logic is the ability to make decisions.
- ✓ The binary concept has two predetermined states. For instance, a light can be on or off, a switch open or closed, or a motor running or stopped.
- ✓ This two-state concept can be the basis for making decisions. The binary 1 represents the presence of a signal, while binary 0 represents the absence of the signal.
- ✓ These two states are actually represented by two distinct voltage levels:+V & 0V.
- ✓ The binary 1 (or logic 1) is referred to as TRUE, ON, or HIGH, while binary 0 (or logic 0) is referred to as FALSE, OFF, or LOW.
- ✓ The use of binary logic to represent the more positive voltage level as 1 is referred to as positive logic.Negative logic uses 0 to represent the more positive voltage level.

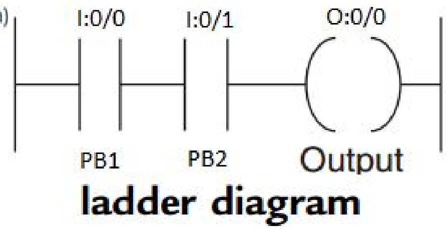
PLCs are based on three fundamental logic functions—AND, OR, and NOT.

3.3.2 The AND function:



PB1	PB2	Alarm Horn
Not pushed (0)	Not pushed (0)	Silent (0)
Not pushed (0)	Pushed (1)	Silent (0)
Pushed (1)	Not pushed (0)	Silent (0)
Pushed (1)	Pushed (1)	Sounding (1)

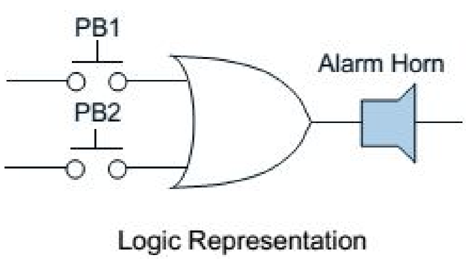




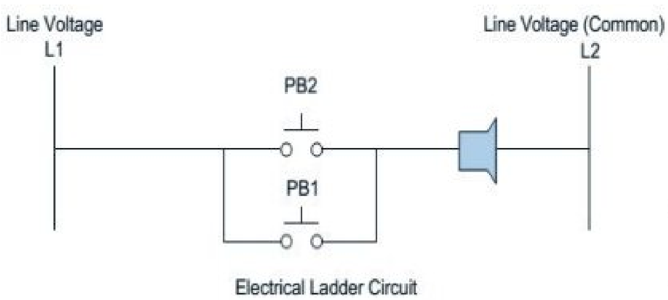
- ✓ An AND gate is a device with two or more inputs and one output.
- ✓ The AND gate output is 1 only if all inputs are 1.
- ✓ In Practical application, when push button PB1 and PB2 are operated, the alarm gives sound.
- ✓ The AND gate operates like a series circuit that produces an output voltage when a voltage appears at each of its inputs.

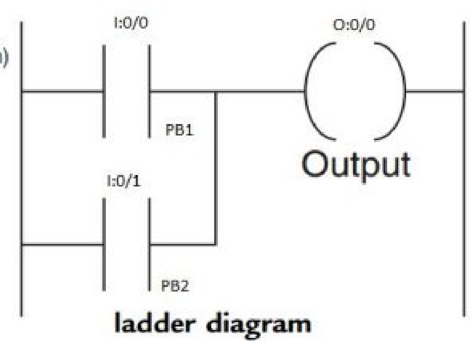
3.3.3 The OR function:

- ✓ An OR gate can have any number of inputs but only one output.
- ✓ The OR gate output is 1 if one or more inputs are 1.
- ✓ In Practical application, when either PB1 or PB2 is operated, the alarm gives sound.
- ✓ The OR gate is essentially a parallel circuit that produces an output voltage when a voltage appears at any input.

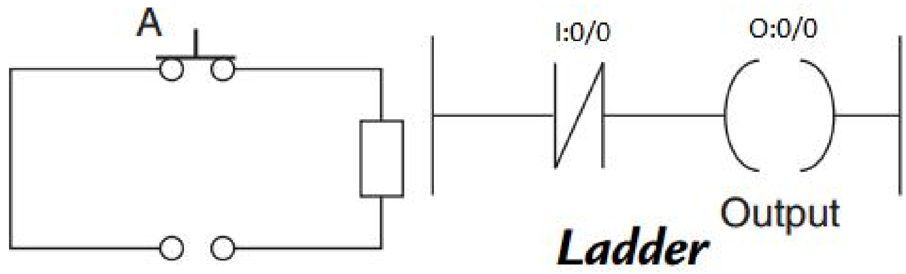


PB1	PB2	Alarm Horn
Not pushed (0)	Not pushed (0)	Silent (0)
Not pushed (0)	Pushed (1)	Sounding (1)
Pushed (1)	Not pushed (0)	Sounding (1)
Pushed (1)	Pushed (1)	Sounding (1)





3.3.4 The NOT function:



- ✓ The NOT function can have only one input.
- ✓ The NOT output is 1 if the input is 0. The output is 0 if the input is 1.
- ✓ The result of the NOT operation is always the inverse of the input, and the NOT function is therefore called an inverter.
- ✓ In application circuit, when the push button is not actuated, the output is ON and when the pushbutton is actuated, the output is OFF.

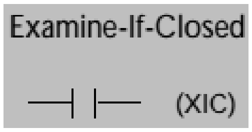
3.4 Relay type Instruction:

Relay instructions form the category of programming instructions that deals with the simple energizing and de-energizing of inputs and outputs. Contacts and coils fall into this category.

- ✓ There are two kinds of relay contact input instructions:
  - examine-if-closed
  - examine-if-open
- ✓ There are four kinds of relay coil output instructions:
  - simple output
  - internal output
  - latch/unlatch output
  - one-shot rising instructions

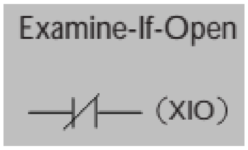
Instruction	**Used to**
XIC – Examine if closed	Examine a bit for an ON condition
XIO – Examine if open	Examine a bit for an OFF condition
OTE – Output Enable	Turn ON or OFF a bit
OTL – Output Latch	Latch a bit ON
OTU – Output Unlatch	Unlatch a bit OFF
ONS – One shot	Detect an OFF to ON transition
OSR –One Shot Rising	Detect an OFF to ON transition
OSF – One Shot Falling	Detect an ON to OFF transition

3.4.1 Examine if closed: (XIC)



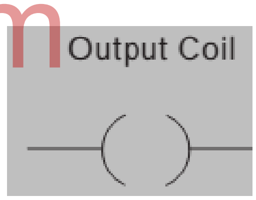
- ✓ Typically represents any input to the control logic.
  Ex: Pushbutton, contact etc.,
- ✓ It has a bit level address.
- ✓ The status bit will be either 1 (ON) or 0 (OFF).
- ✓ The status bit is examined for an ON condition.
- ✓ If the status bit is 1 (ON), then the instruction is TRUE.
- ✓ If the status bit is 0 (OFF), then the instruction is FALSE.

3.4.2 Examine if opened: (XIO)



- ✓ Typically represents any input to the control logic.
  Ex: Pushbutton, contact etc.,
- ✓ It has a bit level address.
- ✓ The status bit will be either 1 (ON) or 0 (OFF).
- ✓ The status bit is examined for an OFF condition.
- ✓ If the status bit is 0 (OFF), then the instruction is TRUE.
- ✓ If the status bit is 1 (ON), then the instruction is FALSE.

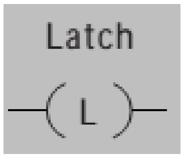
3.4.3 Output Energize: (OTE)



- ✓ It represents any output that is controlled by some combination of input logic.
- ✓ An output can be a connected device or an internal relay output.
- ✓ It has a bit level address.
- ✓ The status bit is set to 1 (ON), when the rung is TRUE.
- ✓ The status bit is set to 0 (OFF), then the instruction is FALSE.
- ✓ If the status bit is 0 (OFF), then the instruction is FALSE.

3.4.4 Output Latch: (OTL)

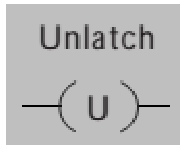
- ✓ It is an output instruction with a bit level address.
- ✓ When the instruction is true, it sets a bit in the output image table.
- ✓ It is a retentive instruction because the bit remains in the previous state when the latch instruction goes false.



3.4.5 Output Unlatch: (OTU)

- ✓ It is an output instruction with a bit level address.
- ✓ When the instruction is true, it resets a bit in the output image table.

- ✓ It is a retentive instruction because the bit remains in the previous state when the unlatch instruction goes false.



3.4.6 One Shot Rising : (OSR)

- ✓ A one-shot rising instruction is not a coil instruction, but rather, a contact instruction.
- ✓ This instruction is used to energize an output coil for only one scan.
- ✓ It is usually the last contact in a rung, located just before the output coil.
- ✓ This instruction's reference address bit can be located in either the binary file (file 3) or the integer file (file 7).
- ✓ Its address cannot be shared by another contact or coil, and it cannot correspond to a real input or output device.
- ✓ Also, the MicroLogix allows only one one-shot rising instruction per rung.

3.4.7 One shot :(ONS)

ONS or the one shot instruction is an instruction used to trigger a signal for one scan of the plc. PLCs generally scan their ladder programs many times per second and can execute things over and over again quite quickly. However with a one shot you can execute something for one scan.

3.5 Timer Instruction:

- ✓ Timers are output instructions.
- ✓ Timing instructions are programming instructions that replace the need for electromechanical timers.
- ✓ It is used to activate or deactivate a device after a preset interval of time.
- ✓ The advantages of PLC timer:
  - o Their settings can be altered easily.
  - o Timer accuracy and repeatability are extremely high
- ✓ Allen Bradley PLC 5 and SLC – 500 PLC timer elements each take three data table word: *the control word, preset word and accumulated word.*

- ✓ **The control word** uses three control bits: EN, TT & DN
  - ✓ ***Enable bit (EN):*** *The enable bit is true whenever the timer instruction is true. When the timer instruction is false, the enable bit is false.*

- ✓ ***Timer Timing bit (TT):*** *The timer timing bit is true whenever the accumulated value of the timer is changing which means the timer is timing. When the timer is not timing, the accumulated value is not changing, so the timer timing bit is false.*
- ✓ ***Done bit (DN):*** *The done bit changes state whenever the accumulated value reaches the preset value.*

✓ **Preset: (PRE)**

- The preset word is the set point of the timer.
- The preset word has a range of 0 through 32767 and is stored in binary form.
- The preset will not store a negative number.
-

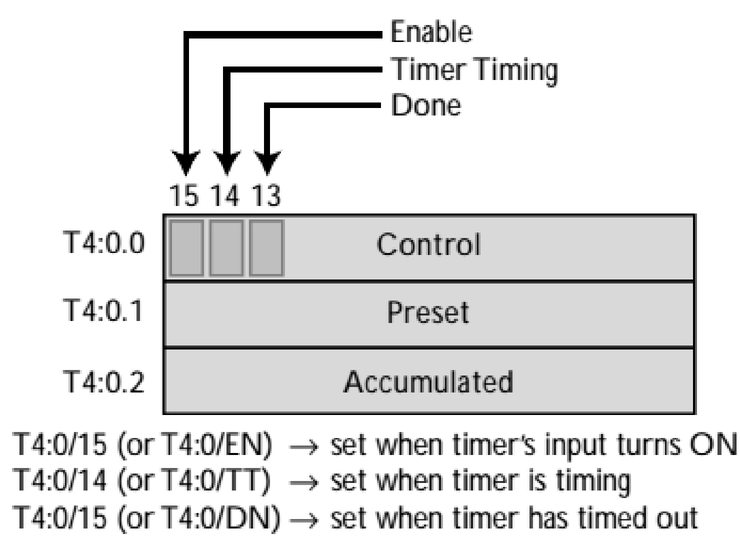
✓ **Accumulated: (ACC)**

- The accumulated value word is the value that increments as the timer is timing.
- The accumulated value will stop incrementing when its value reaches the preset value.

Time base:

- ✓ Timer instruction also requires that you enter a time base which is either 1.0s or 0.01 s or 0.001s.
- ✓ Actual time delay = Preset Value x Time base

Ex: *Preset value = 50, Time base = 0.01s,*
Actual Time delay = Preset value x time base = 50 x 0.01s = 5 sec.



The data stored in each word of a timer's address

3.6 Types of Timer Instruction:

1. Timer ON delay (TON) 2. Timer OFF delay (TOF) 3. Retentive timer (RTO)

3.6.1 Timer ON Delay Instruction: (TON)

- ✓ The timer ON-delay instruction is a block-format instruction that is represented by the symbol shown in Figure below.
- ✓ This block has two outputs:
  - an enable output coil (EN)
  - a done output coil (DN)
- ✓ A timer ON-delay instruction energizes its done output (DN) after the timer block's input turns on and a specified delay has occurred.
- ✓ This instruction is sometimes called a timer ON-delay energize instruction.
- ✓ When the timer block's input has logic continuity, the block's enable output (EN) will turn on. As a result, a 1 will be stored in bit 15 of the timer's control word.
- ✓ Once the timer is enabled, it will start to time. Thus, a 1 will be stored in bit 14, which is the timer timing bit.

- ✓ As the timer times, the accumulated value increases until it equals the preset value. At that point, the timer timing bit (TT) will become a 0, and the done bit (DN) will become a 1, meaning that the done output (DN) will turn on.
- ✓ This done output (DN) is the timer's delay action contact.
- ✓ The length of the time delay can be adjusted by changing the preset value.

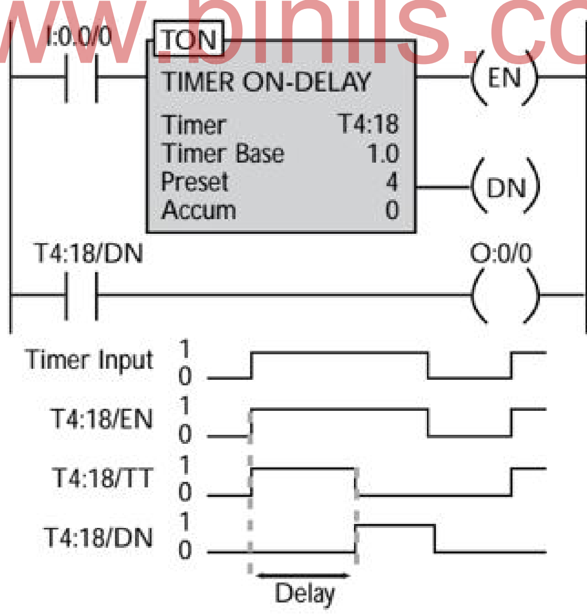


Figure: A timer ON-delay block and its associated timing diagram

Sequence of operation:

1. When the input I:0.0/0 is closed, it will cause the timer's enable output (EN) to turn ON.
2. At the same time the timer starts counting and counts until the accumulated value equals the preset value.
3. For example the preset time for this timer is 4 seconds.

4. When the accumulated value reaches 4 seconds, the done bit (DN) goes from false to true and timer timing bit (TT) goes from true to false and causing the output coil O:0/0 to turn on.
5. When input I:0.0/0 goes false, the timer instruction goes false and also resets, at which time the control bits are all reset and the accumulated value resets to 0.

3.6.2 Timer OFF Delay Instruction: (TOF)

- ✓ The timer OFF delay instruction is also a block-format instruction that is represented by the symbol shown in Figure below.
- ✓ This block has two outputs:
  - an enable output coil (EN)
  - a done output coil (DN)
- ✓ A timer OFF delay instruction de-energizes its done output (DN) after the timer block's input turns off and a specified delay has occurred.
- ✓ Thus, the timer OFF delay instruction is also called a timer OFF-delay de-energize instruction.

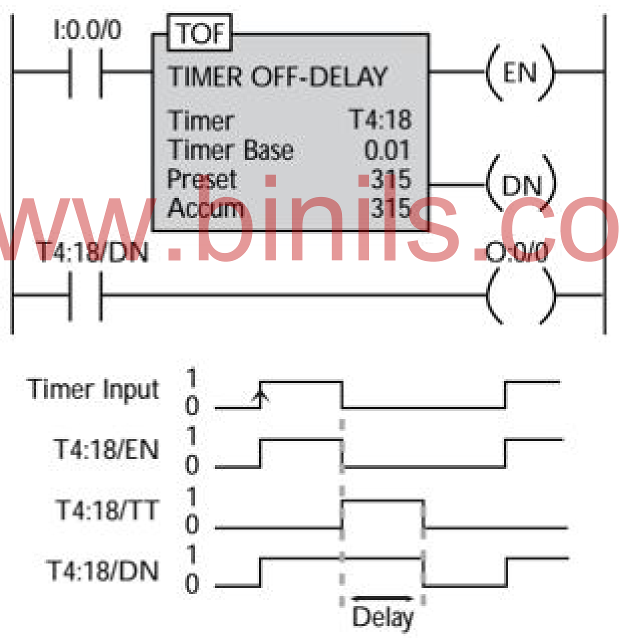


Figure: A timer OFF-delay block and its associated timing diagram

<u>Sequence of Operation:</u>

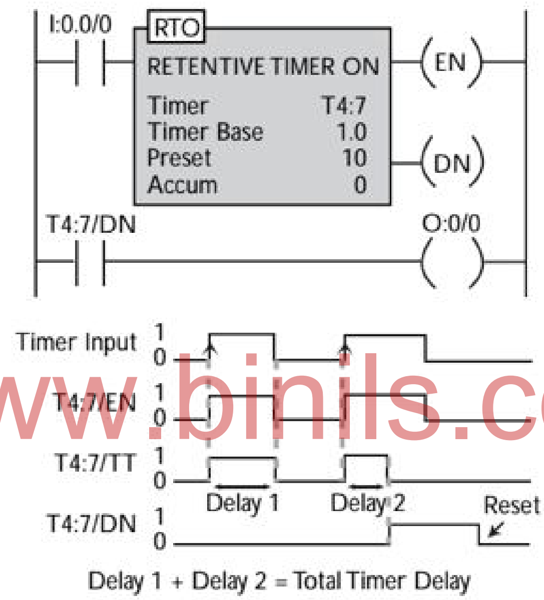
- ✓ The done output (DN) will be off when the program is first started and the timer's input is off.
- ✓ When the input logic turns on, both the block's enable output (EN) and done output (DN) will turn on. However, the timer will not start timing because it is waiting for an OFF signal instead of an ON signal
- ✓ When the block's input turns off, the enable output will turn off and the timer will start timing. The done output will stay on because it is waiting for the timer to time out before it will turn off.

- ✓ Once the accumulated value equals the preset value, the timer will stop timing and the done output (DN) will turn off, implementing the OFF-delay de-energize function.
- ✓ This done output (DN) is the timer's delay action contact.
- ✓ The length of the time delay can be adjusted by changing the preset value.

3.6.3 Retentive Timer Instruction:

Retentive Timer ON : (RTO)

A retentive timer instruction operates much like a timer ON-delay instruction. A retentive timer, however, can stop timing and then start timing again without its accumulated value resetting to 0.



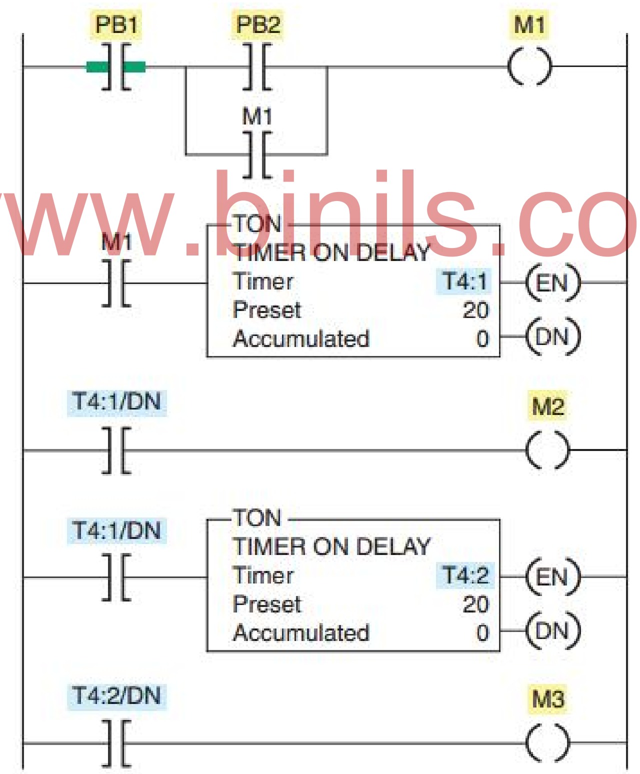
- ✓ When the input logic turns on, the enable output will turn on, and the timer will start timing.
- ✓ If the input logic turns off, the enable output will turn off, and the timer will stop timing.
- ✓ The accumulated value, however, will not reset to 0.
- ✓ When the timer starts timing again, it will pick up where it left off.
- ✓ When the accumulated value finally reaches the preset value, the done output will turn on.
- ✓ Once a retentive timer has timed out, its done output will remain on even if its input logic and enable output turn off.
- ✓ A reset instruction must be used to turn the done output off and reset the timer's accumulated value.
- ✓ The RES instruction has the same address as the timer it is to reset.

3.6.4 Cascading Timers

The programming of two or more timers together is called cascading. The timer can be interconnected or cascaded to satisfy any required control logic.

The operation of the circuit can be summarized as follows:

i) Motor starter coil M1 is energized when the momentary start pushbutton PB2 is actuated.
ii) As a result, motor 1 starts, contact M1 closes to seal in PB2, and timer coil T4:1 is energized to begin the first time-delay period.
iii) After the preset time period of 20 s, T4:1/DN contact closes to energize motor starter coil M2.
iv) As a result, motor 2 starts and timer coil T4:2 is energized to begin the second time-delay period.
v) After the preset time period of 20 s, T4:2/DN contact closes to energize motor starter coil M3, and so motor 3 starts.
vi) Hence actual time delay to start Motor 3 = Time dlay of T4:1 + Time delay of T4:2

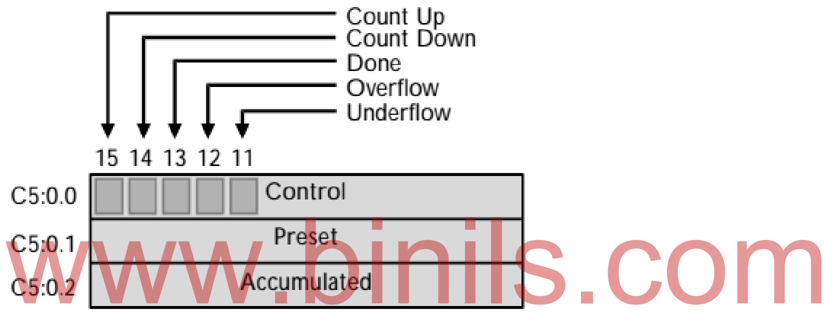


3.7 Counter Instruction

- ✓ Counter instructions are output instructions.
- ✓ Counter instructions are programming instructions that replace the need for electromechanical counter.
- ✓ A counter instruction has two values associated with it:
  - ✓ • the preset value
  - ✓ • the accumulated value

- ✓ The preset value specifies the target number of counts, while the accumulated value indicates the actual number of counts that have already occurred.
- ✓ In a counter, the preset and accumulated values always increase or decrease in increments of one.
- ✓ Each Allen-Bradley PLC 5 and SLC 500 counter instruction occupies three memory word locations in the C5 counter data file.
- ✓ These three data words are the control word, preset word and accumulated word.

- ✓ **Control word:**
  - ✓ **Count-down Enable bit:** The count down enable bit is used with the count down counter and is true whenever the count down instruction is true.
  - ✓ **Done bit (DN) :** The done bit is true whenever the accumulated value is equal to or greater than the preset value of the counter.
  - ✓ **Overflow bit (OV) :** The overflow bit is true whenever the counter counts past its maximum value which is 32,767.
  - ✓ **Underflow bitt (UV) :** The underflow bit will go true when the counter counts below -32768.



The data stored in each word of a counter's address

- ✓ **Preset Value: (PRE)**
  - o The preset value specifies the value that the counter must count to before it changes the state of the done bit. The preset value is the set point of the counter and ranges from -32,768 to +32,768.

- ✓ **Accumulated value: (ACC)**
  - o It is the current count based on the number of times the rung goes from false to true. It can also ranges from -32,768 to +32,768.

3.7.1 UP Counter

Up-counters perform a counting function when the associated input element transitions from an OFF to ON state. Up-counters begin at some preset value and increment upward. Up-counters are retentive and require an associated reset element to clear the counted values.

Counter Up Instruction: (CTU)

- ✓ A count up instruction is represented by the symbol shown in Figure below.
- ✓ The function of a count up instruction is to increase its accumulated value by one every time the block's input makes an OFF-to-ON transition.

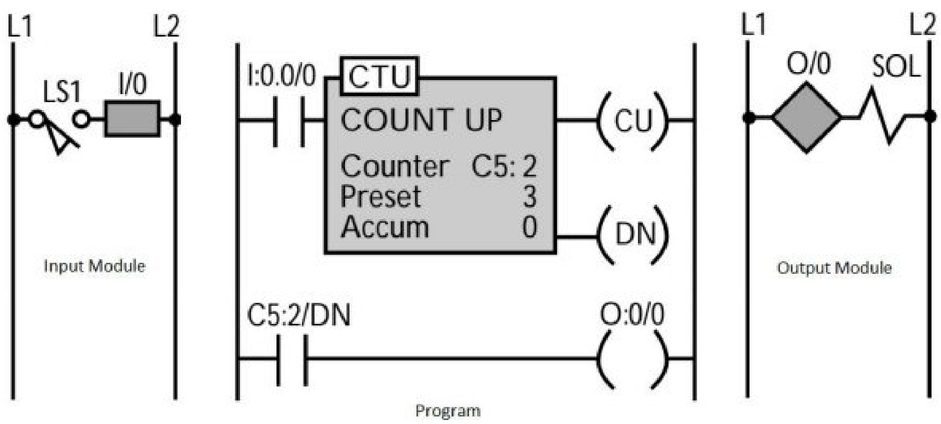
- ✓ After a certain number of OFF-to-ON transitions have occurred, the count up instruction will energize its output.

- ✓ A count up block has two output coils:
  - *a count up output coil (CU) : which indicates that the counter block is energized*
  - *a done output coil (DN), which indicates that the count is complete*

Example:

The solenoid (SOL) should turn on after the limit switch (LS1) has turned on three times. The circuit operates as follows:

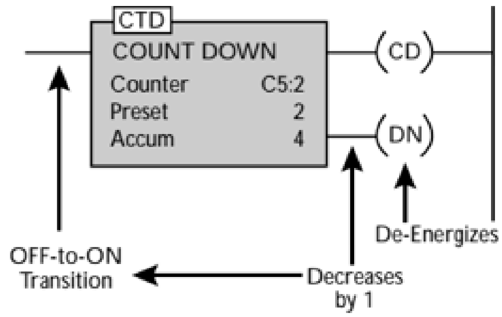
- ✓ When the limit switch turns on for the first time, the count up output will be energized, and the accumulated value will increase to 1.
- ✓ When the limit switch turns off then on again, the accumulated value will increase to 2.
- ✓ When the switch makes its third OFF-to-ON transition, the accumulated value will increase to 3 and the done output will turn on because the accumulated value is equal to the preset value.
- ✓ When the done output turns on, the solenoid output in the second rung will be energized.
- ✓ In a counter circuit, the counter will continue to count even after the accumulated value has reached the preset value.
- ✓ The done output will remain on as long as the accumulated count is greater than or equal to the preset count.
- ✓ The only way to reset the accumulated value and turn off the done output is to use a reset instruction.



3.7.2 Down Counter:

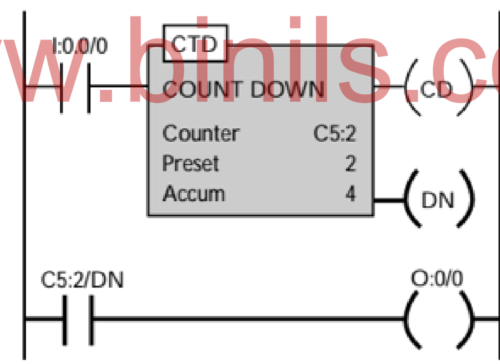
Down-counters perform a counting function when the associated input element transitions from an OFF to ON state. Down-counters begin at some preset value and decrement downward. Down-counters are retentive and require an associated reset element to clear the counted values.

Counter Down Instruction: (CTD)



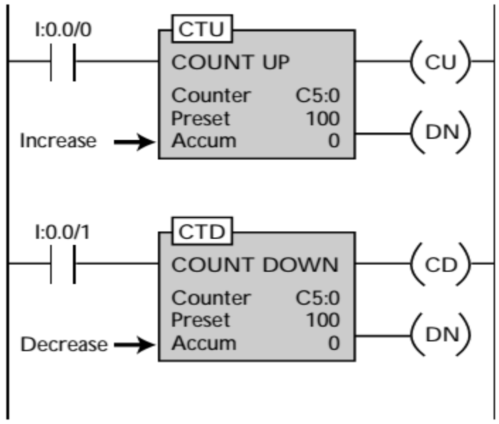
- ✓ A count down instruction decreases its accumulated value by one every time the block's input makes an OFF-to-ON transition.
- ✓ When the accumulated value becomes less than the preset value, the count down instruction de-energizes its output.
- ✓ When the counter's accumulated value is greater than or equal to its preset value, the counter's output will be on.
- ✓ A count down instruction also has two outputs:
  - a count down output : which indicates that the counter is energized
  - a done output : which signals that the target count value has been reached.

Example:



- ✓ In this circuit, the count down block's done output will already be on because the accumulated value is greater than the preset value.
- ✓ When the block's input I0.0/0 is turns from OFF to ON, the accumulated value will decrease to 3.
- ✓ When the block's input makes this OFF-to-ON transition again, the accumulated value will decrease to 2.
- ✓ When the input makes one more OFF-to-ON transition, the accumulated value will drop to less than the preset value and the done output will turn off, deenergizing the done output C5:2/DN and output O:0/0.
- ✓ The CTD instruction requires the RES instruction to reset its accumulated value and status bits.

3.7.3 UP/DOWN Counter:

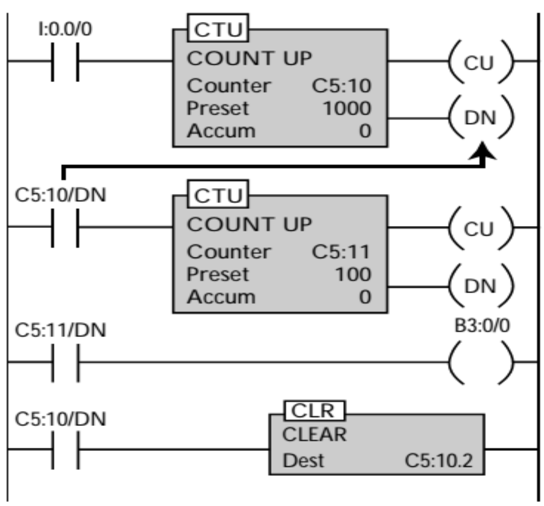


- ✓ In the up/down counter shown in Figure below, both counters share the same address and the same preset and accumulated values.
- ✓ As a result, the up counter increases the accumulated value every time a certain event occurs, while the down counter decreases the same accumulated value if another event occurs.

3.7.4 Cascading counters:

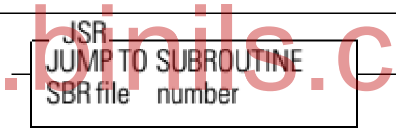
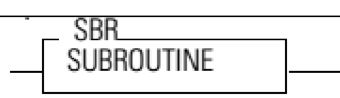
Depending on the applications, it may be necessary to count events that exceed the maximum number allowable per counter instruction. One way of accomplishing this count is by interconnecting or cascading two counters.

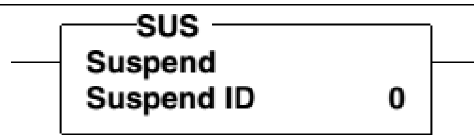
- ✓ A counter instruction's accumulated value has a range from –32,768 to +32,767.
- ✓ Once a counter reaches a count of +32,767, it cannot go any higher. Therefore, it wraps the accumulated count back around to –32,768 and starts counting up again.
- ✓ To count past the +32,767 count value, you must cascade two counters, making sure that they self-reset in each scan.
- ✓ When two counters are cascaded, they are programmed so that one counter provides the input to the other counter (see Ladder below).
- ✓ Ladder shows below has two cascaded counters that implement a count to 100,000. These cascaded counters have addresses C5:10 and C5:11.



3.8 Program control instruction:

The program control instructions allow for greater flexibility and greater efficiency in the program scan.

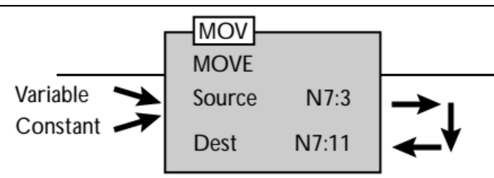
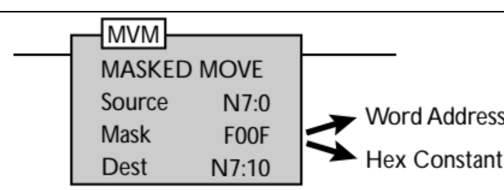
Command	Name	Description
JMP	Jump to Label	 It is an output instruction and it Jump forward/backward to a corresponding label instruction. The label is a target for the jump. A jump instruction jumps to a label with the same address. If the jump coil is energised, all logic between the jump and label instruction is bypassed and the processor continues scanning after the LBL instruction.
LBL	Label	 Specifies label location. The label is a target for the jump. It is the first instruction in the rung and it is always true.
JSR	Jump to Subroutine	 Jump to a designated subroutine instruction. When rung condition is true for this output instruction, it causes the processor to jump to the targeted subroutine file. Each subroutine must have a unique file number.
RET	Return from Subroutine	 It is an output instruction that marks the end of the subroutine file. It exits current subroutine and returns to previous condition.
SBR	Subroutine	 It is first instruction on the first rung in the subroutine file. It identifies the subroutine program.
TND	Temporary End	

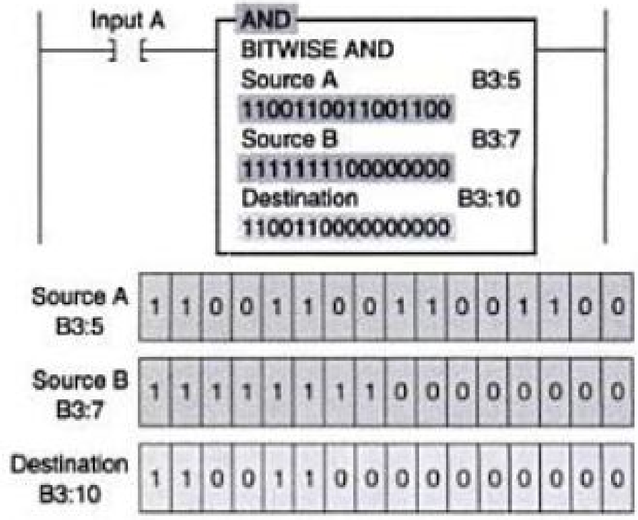
		It is an output instruction used to conditionally omit the balance of the current program file of subroutines.
MCR	Master Control Reset	 Clears all set outputs between the paried MCR instructions. This instruction can be programmed to control an entire rungs or selected rungs of the program. The MCR instruction establishes a zone is the user program in which all non-retentive outputs can be turned off simultaneously. The MCR zone is enclosed by a start fence, which is a rung with a conditional rung, and an end fence, which is a rung with an unconditional MCR.
SUS	Suspend	 Identify specific conditions for program debugging and system troubleshooting.

3.9 Data Manipulation Instruction:

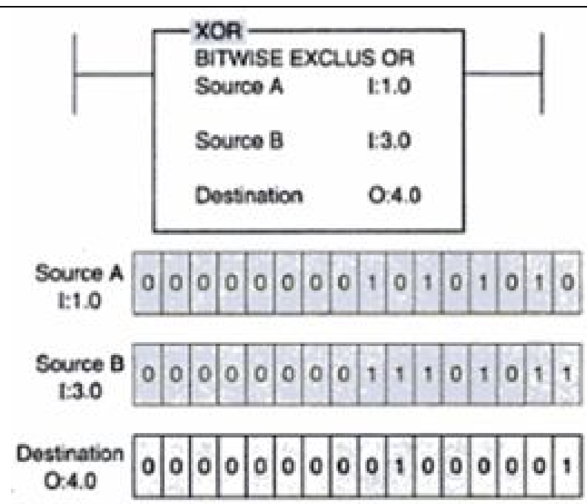
Data manipulation involves transfer of data and operation on data with math functions, data conversion, data comparison and logical operations. Data manipulation can be placed in two categories: 1. Data transfer 2. Data comparison

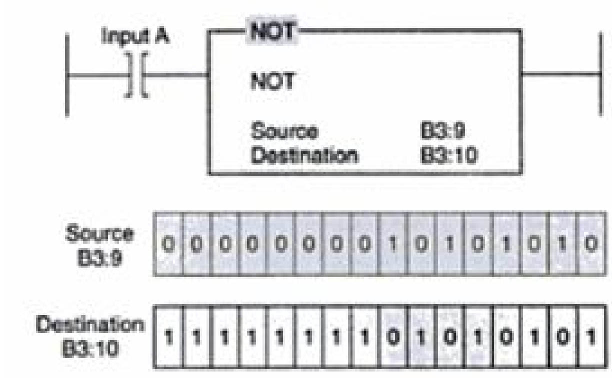
1. Data transfer instruction: It simply involves the transfer of the contents from one word or register to another.

Command	Name	Description
MOV	Move	 It is used to copy the value in one word to another word. It moves the data from source word to a destination word.
MVM	Masked Move	 A masked move instruction.

<table>
<tr><td></td><td></td><td>It is used to copy the desired part of a 16-bit word by masking the rest of the value.</td></tr>
<tr><td>AND</td><td>And</td><td>

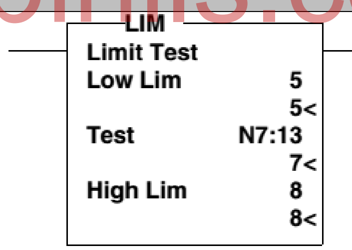
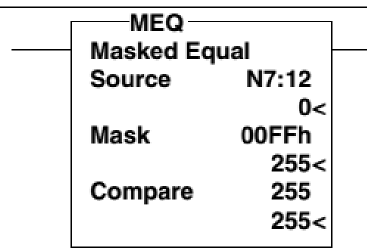
Performs a bitwise AND operation.</td></tr>
<tr><td>OR</td><td>Or</td><td>

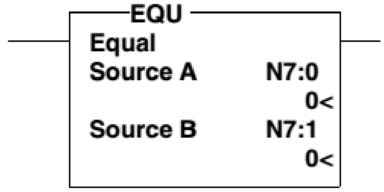
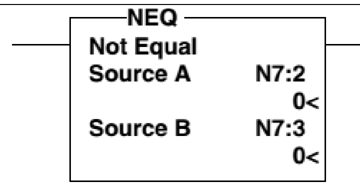
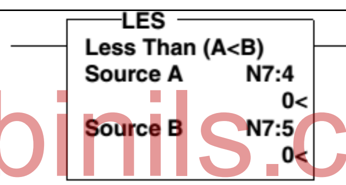
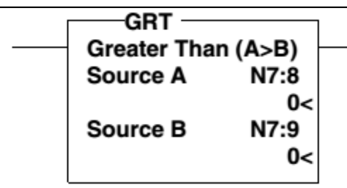
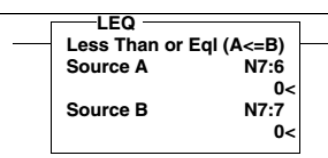
Performs a bitwise OR operation.</td></tr>
<tr><td>XOR</td><td>Exclusive Or</td><td>
</td></tr>
</table>

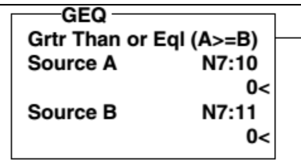
		Performs a bitwise XOR operation.
NOT	Not	 Performs a NOT operation.
CLR	Clear	Sets all bits of a word to zero.

3.10 Data Compare Instruction:

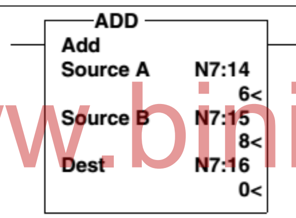
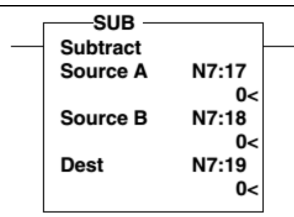
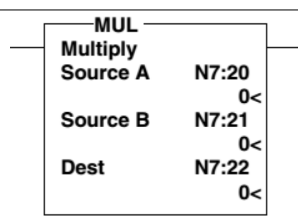
Data transfer instructions are all output instruction whereas data compare instructions are input instruction. It compare the data stored in two or more words or registers and take decision based on the program instructions.

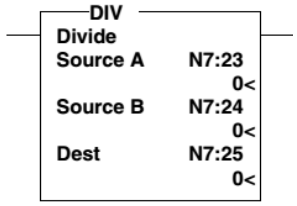
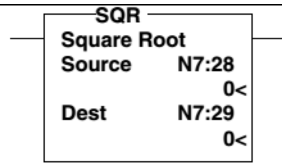
Command	Name	Description
LIM	Limit Test	 Tests whether one value is within the limit range of two other values.
MEQ	Masked comparison for equal	 If the data at the source address match the data at the compare address bit-by-bit, the instruction is true otherwise it is logically false.

EQU	Equal	 It is an input instruction that compares source A to source B. When source A is equal to source B, the instruction is logically true; otherwise it is logically false.
NEQ	Not Equal	 It is an input instruction that compares source A to source B. When source A is not equal to source B, the instruction is logically true; otherwise it is logically false.
LES	Less than	 It is an input instruction that compares source A to source B. When source A is less than source B, the instruction is logically true; otherwise it is logically false.
GRT	Greater than	 It is an input instruction that compares source A to source B. When source A is greater than source B, the instruction is logically true; otherwise it is logically false.
LEQ	Less than or Equal	 It is an input instruction that compares source A to source B. When source A is less than or equal to source

		B, the instruction is logically true; otherwise it is logically false.
GEQ	Greater than or Equal	 It is an input instruction that compares source A to source B. When source A is greater than or equal to source B, the instruction is logically true; otherwise it is logically false.

3.11 Math Instruction:

Command	Name	Description
CPT	Compute	Evaluates an expression and sores the result in the destination
ADD	Add	 Add source A to source B and stores the results in the destination.
SUB	Subtract	 Subtract source B from source A and stores the results in the destination.
MUL	Multiply	 Multiplies source A by source B and stores the results in the destination.

DIV	Divide	 Divide source A by source B and stores the results in the destination and math register.
SQR	Square Root	 Calculate the square root of the source and places the integer result in the destination.
NEQ	Negate	Changes the sign of the source and places it in the destination.
TOD	To BCD	Converts a 16 bit integer source value to BCD and stores it in the math register of the destination.
FRD	From BCD	Converts a BCD value in the math register or the source to an integer and stores it in the destination.

3.12 Sequencer Instruction:

Programmed sequencers can perform the same specific on or off patterns of outputs that are continuously repeated with a drum switch, but with much more flexibility. For example, the on/off operation of 16 discrete outputs can be controlled, using a sequencer instruction, with only one ladder rung.

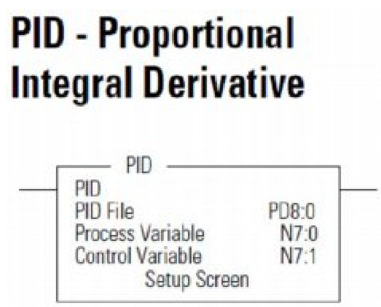
Command	Name	Description
SQO	Sequencer Output	It is an output instruction that uses a file to control various output devices.
SQI	Sequencer Input	It is an input instruction that compares bits from an input file to corresponding bits from a source address. The instruction is true if all pairs of bits are the same.
SQC	Sequencer Compare	It is an output instruction that compares bits from an input source file to corresponding bits from data words in a sequence file. If all pairs of bits are the same, then a bit in the control register is set to 1.
SQL	Sequencer Load	It is an output instruction used to transfers data from the input source module to the sequencer file. The instruction functions much like a file-to-word transfer instruction.

3.13 PID Instruction:

The PID instruction is an output instruction that controls physical properties such as temperature, pressure, liquid level, or flow rate using process loops.

The PID Concept:

The PID instruction normally controls a closed loop using inputs from an analog input module and providing an output to an analog output module. For temperature control, you can convert the analog output to a time proportioning on/off output for driving a heater or cooling unit.



Process Variable:

PV is an element address that stores the process input value. This address can be the location of the analog input word where the value of the input A/D is stored. This value could also be an integer value if you choose to pre-scale your input value to the range 0 to 16383.

Control Variable:

CV is an element address that stores the output of the PID instruction. The output value ranges from 0 to 16383, with 16383 being the 100% on value. This is normally an integer value, so that you can scale the PID output range to the particular analog range your application requires.

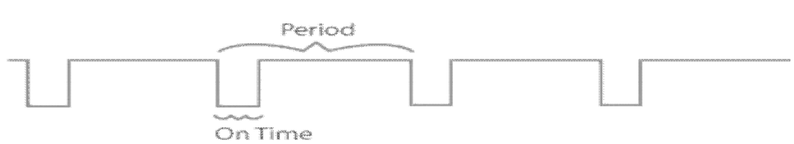
Setup Screen:

Double click Setup Screen on the instruction to bring up a display that prompts you for other parameters you must enter to fully program the PID instruction.

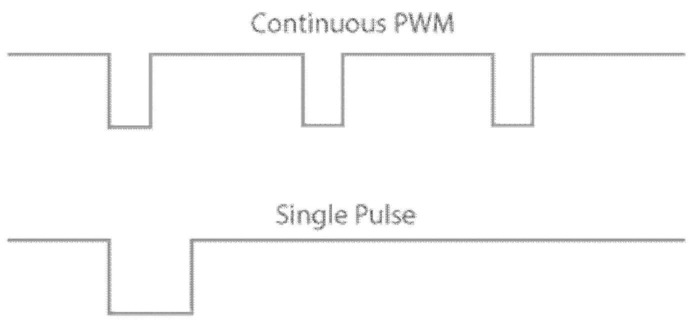
3.14 PWM Function:

PWM is the acronym for "Pulse Width Modulation". Pulse width modulation is a very powerful function that is commonly used to control the positioning of servo valves. It is also used for laser pulse control, motion control, light intensity controls and other other applications. The single pulse feature of the function is applicable to any single pulse output need.

A pulse width modulated output signal is one that has a period and an output "on time", or duty cycle, within that period.



The figure , on the right, illustrates a PWM output signal. Most PLC digital outputs are "sinking" transistor outputs – so when the output is on, the output sinks, pulling the signal low.



The next figure show a standard PWM on the top, and a single pulse PWM output on the bottom. A single pulse PWM output is active for its defined "on time" within the PWM period. It does not repeat. If you want to output another pulse, your program must execute another "Start PWM pulse" block.

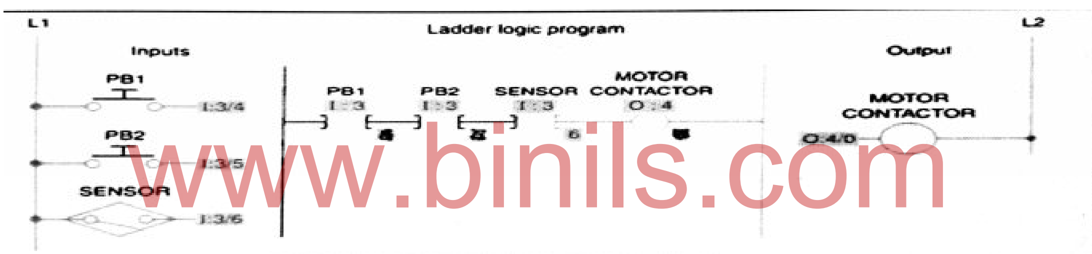
Perform PWM Function in PLC

- To perform this two timers are used to Turn ON and OFF an output according to the length of a pulse.
- Timer preset value should vary such that when preset of one timer is increased, preset value of other timer should decrease in order to maintain Turn ON and OFF time of output.
- Select input bits such that we can directly enter digits and place it into Preset value of a timer.
- This can be done by using a digital input device which generates 0-9 numerical digits.
- Output of this Digital device is always in BCD form and Timers preset values store data in Hexadecimal, so whichever data are sent to preset register of a time, it has to be converted into Hexadecimal form.
- FRD instruction can be used to perform BCD to Hex conversion.
- Output of this conversion is directly moved to preset register of timer which can be performed by MOV instruction.

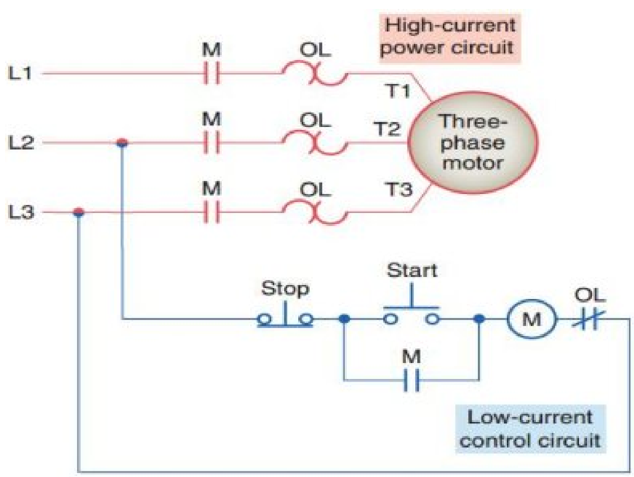
Description of language ladder

There are different types of programming languages for PLCs. Perhaps the most common is the ladder programming. The ladder diagrams are commonly used schemes to represent the control logic of industrial systems. Is called "ladder" diagram because they resemble a ladder, with two vertical rails (supply) and "rungs" (horizontal lines), in which there are control circuits that define the logic through functions. In this way the main ladder language features are:

- Input instructions are entered on the left.
- Output Instructions are located on the right.
- Power rails are the power supply lines L1 and L2 for alternating current circuits, and 24 V - earth for DC circuits.
- Most PLC allows more than one output for each row (Rung).
- The Processor (or "Controller") explores rungs of the ladder from top to bottom and from left to right.



3.15.1 Ladder Diagram for DOL starter:



- **Figure : Hard Wire Circuit**

DESCRIPTION:

- Starts directly the motor through M coil by means of contacts ‘M’ connected with motor.
- Assume that control circuit as mentioned in hard wire circuit.
- As like the hard wire control STOP button is normally close one so we choose NC for STOP while enter in ladder. Likewise for START chooses make contact for this one.
- For the output M coil we could choose normal output energizer in ladder entry and make LATCH circuit contact as a feedback loop which is of for START operation.

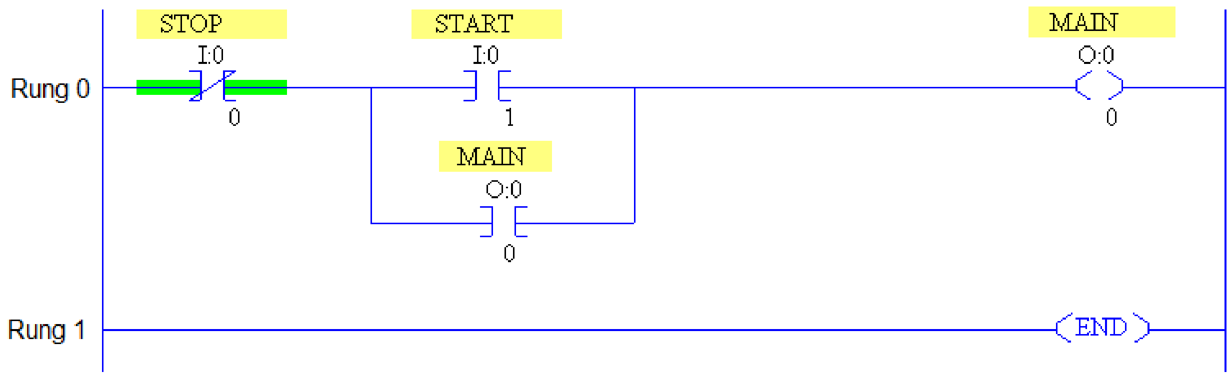
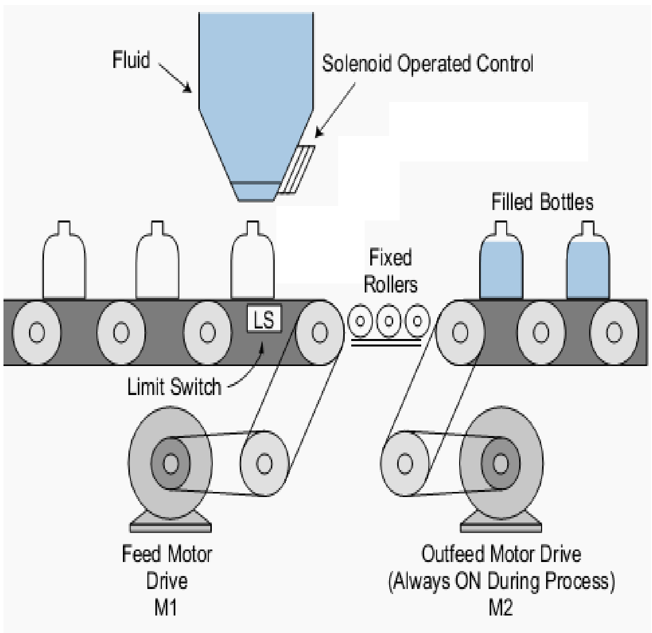


Figure: PLC Ladder Diagram for DOL Starter (Allen Bradley PLC)

Sequence of operation:

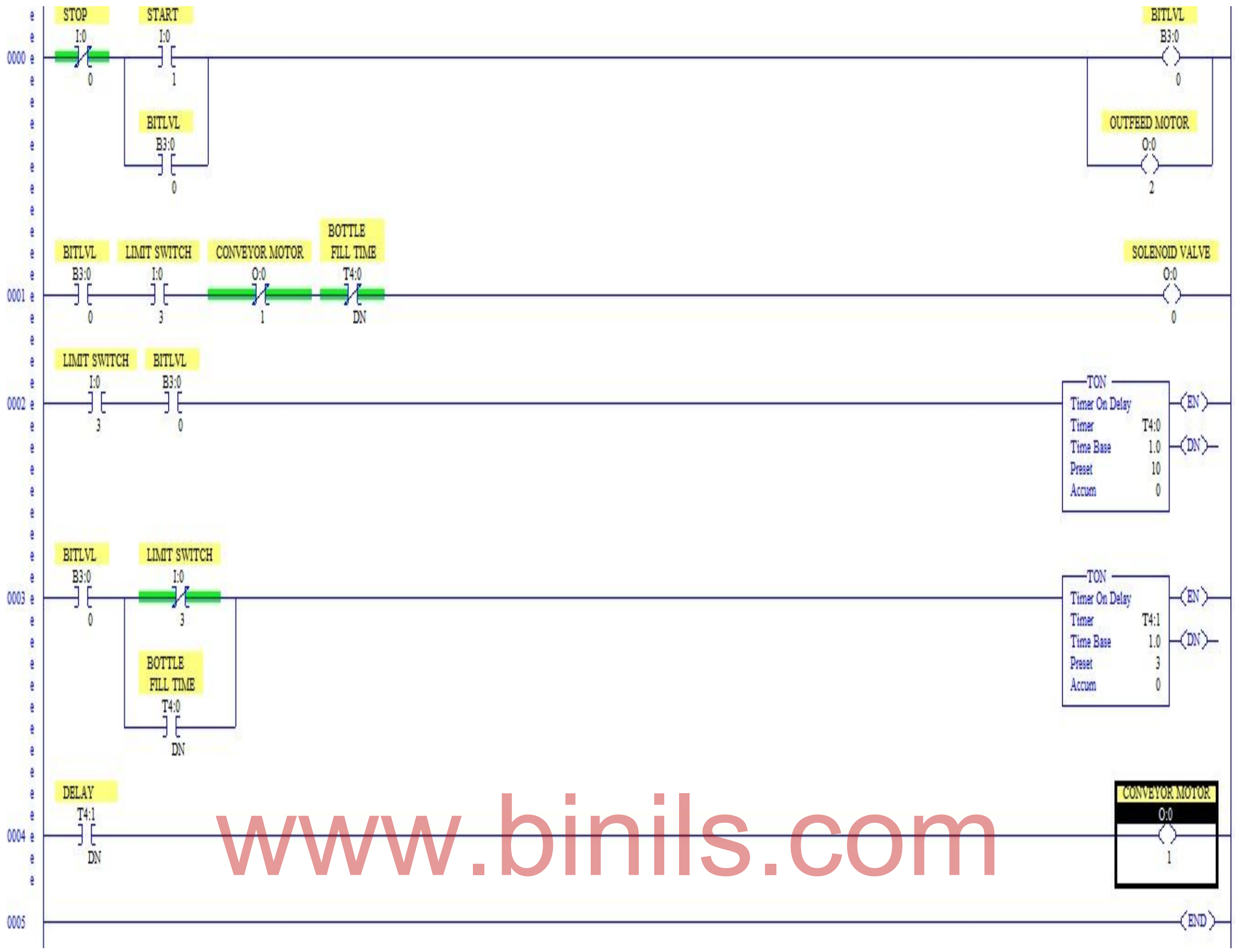
i)	Rung 0:	When the Start push button is pressed, the NO instruction I:0/1 is set to HIGH. The Rung 0 is TRUE and allows to energise Main coil which is connected to output instruction O:0/0. The NO contact of output instruction O:0/0 is set to high and provides sealing effect. The motor is started with full line voltage and continues to run.
ii)	Rung 0:	When the stop push button is pressed, the NC instruction I:0/0 is set to LOW. Now Rung 0 is FALSE and deenergise Main coil and stop the motor.

3.16 Bottle filling system:



Description:

- In this automatic bottle filling system conveyer action done by an output feed motor drive M1 and filling fluid action is performed by an output solenoid valve.
- For to make sensing input (bottle) limit switch will done that action as PLC input.
- Timers used in ladder for filling time and delay operation.
- Fixed rollers used to pass the filled bottle to out feed conveyor that it always in ON during process.



List of Input and outputs

I:0/0 -STOP(NC) ; I:0/1 - START(NO) ; I:0/3 - SENSOR(LIMIT SWITCH)

O:0/0 - SOLENOID VALVE ; O:0/1 - CONVEYOR MOTOR ;O:0/2 –OUT FEED MOTOR

T4:0 – FOR FILLING TIME; T4:1 – DELAY AFTER FILLING TO MOVE.

Sequence of operation:

- When the start button is pressed the bit address B3:0/0 will get energized and out feed motor starts run by means of energisation of O:0/2.
- For the energisation of output B3:0/0 it make the conveyor motor to start till the limit switch senses (I:0/3).
- If the limit switch senses conveyor motor O: 0/1 output get deenergized and make the energization of solenoid value for some time period (T4:0) to fill.
- After filling the process waits for some delay (T4:1) and starts the conveyor to move and repeat the operation till the stop button pressed

3.17 Automatic car parking system:

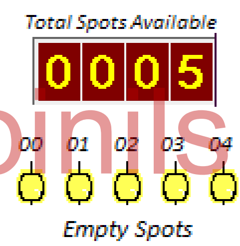
Description

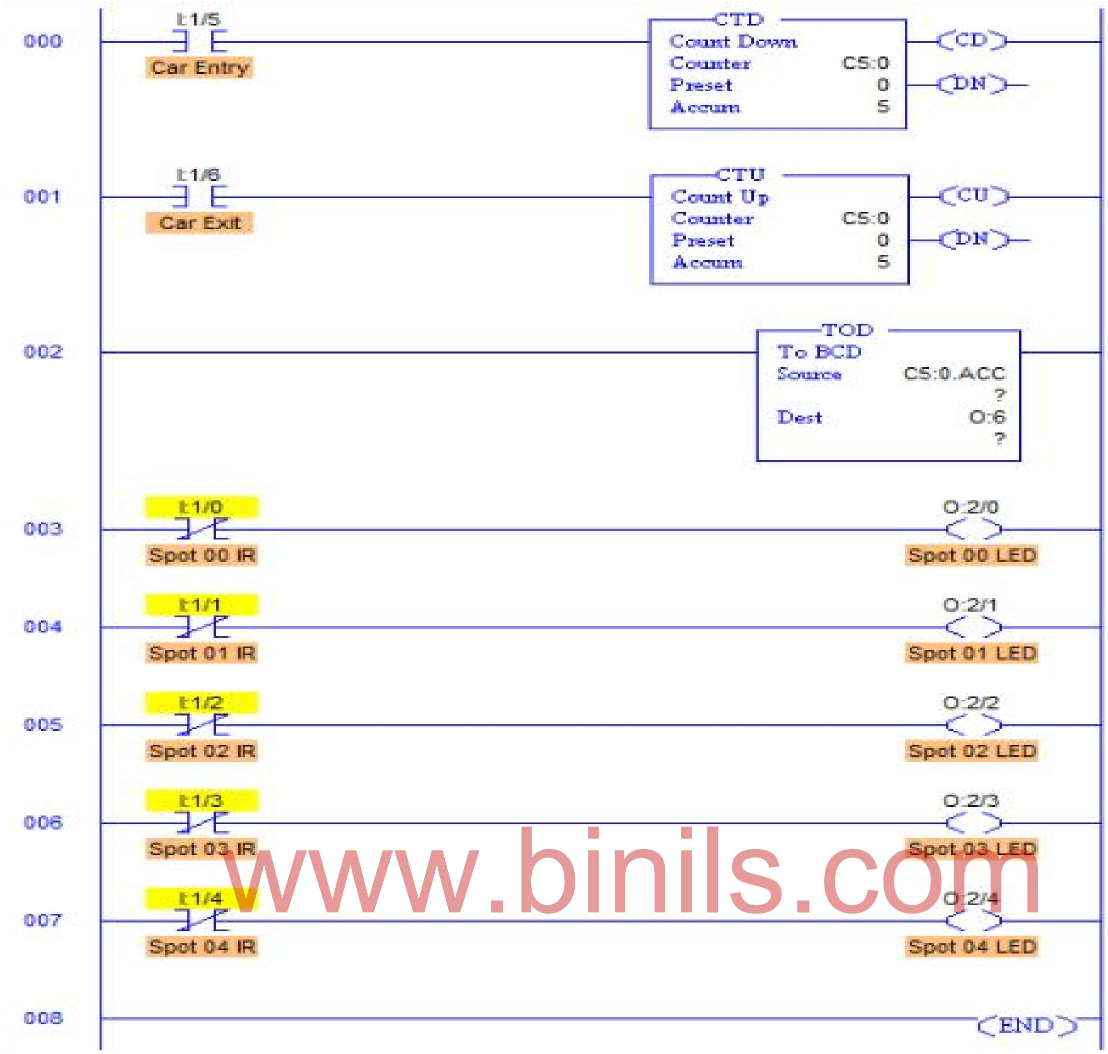
A parking plot has total capacity of Cars. Numbers of empty spots are displayed on the display outside the Parking Plot and which spots are available is to be indicated by LEDs. Implement this in PLC using Ladder Diagram programming language.

Problem Solution

- Counter is used to count the number of empty spots.
- Proximity Sensors or IR Sensors are used to detect the presence of car.
- Here in this system IR Sensor can be well installed to make this system cost efficient since Proximity Sensors are costly than IR Sensors.
- Value of counter is displayed on the display which is mounted outside the parking plot.
- This counter value is converted into decimal.

Display arrangement:





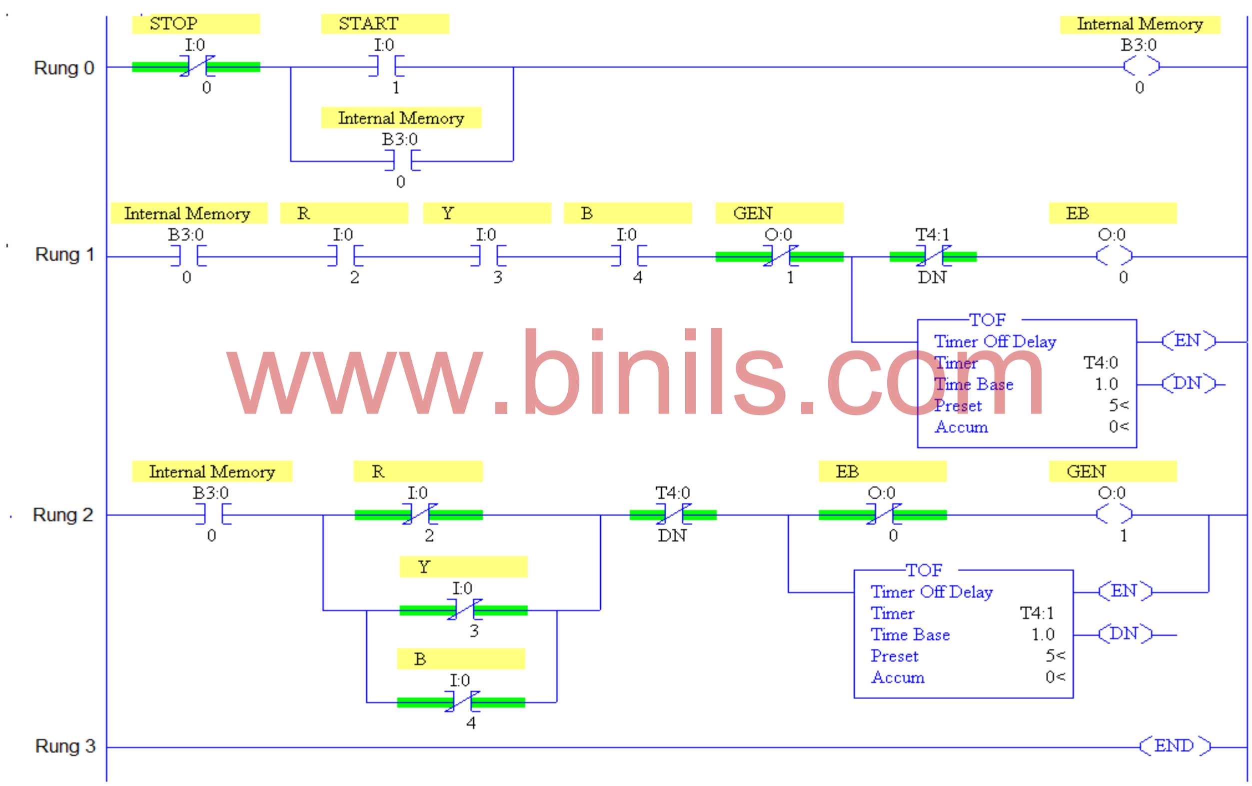

```
List of Inputs and Outputs
 I:1/0 to I:1/4 = IR Sensor to detect the presence of cars (Inputs)
 O:2/0 to O:2/4 = LEDs to indicate presence of car spots   (Outputs)
 C5:0           = To increment when Car exits              (Counter Up)
 C5:0           = To decrement when Car enters             (Counter Down)
 O:6            = Display address                          (Output)
```

Sequence of operation:

- Counter Up CTU and Counter Down CTD are used to determine the Exit and Entry of cars respectively.
- Value 5 is already stored in the accumulator since only 5 number of spots are there in this Parking Plot.
- So whenever car enters or exits from the Parking area, the value in the counter is incremented and decremented accordingly.

- Accumulator holds decimal values, this value thus sent to the display through BCD converter which converts Decimal digits into equivalent Binary Coded Decimal signals.
- Display receives whatever the value Accumulator holds, in terms of BCD.
- I:1/5 and I:1/6 are two inputs from other two IR Sensors to detect the exit and entry of cars accordingly.
- Here again, CTU and CTD both have the same address in order to vary accumulator value of both counters according to Exit and Entry of cars.
- XIO (Normally Closed) contact is used here for IR Sensor outputs so that LED is ON when the spot is empty.

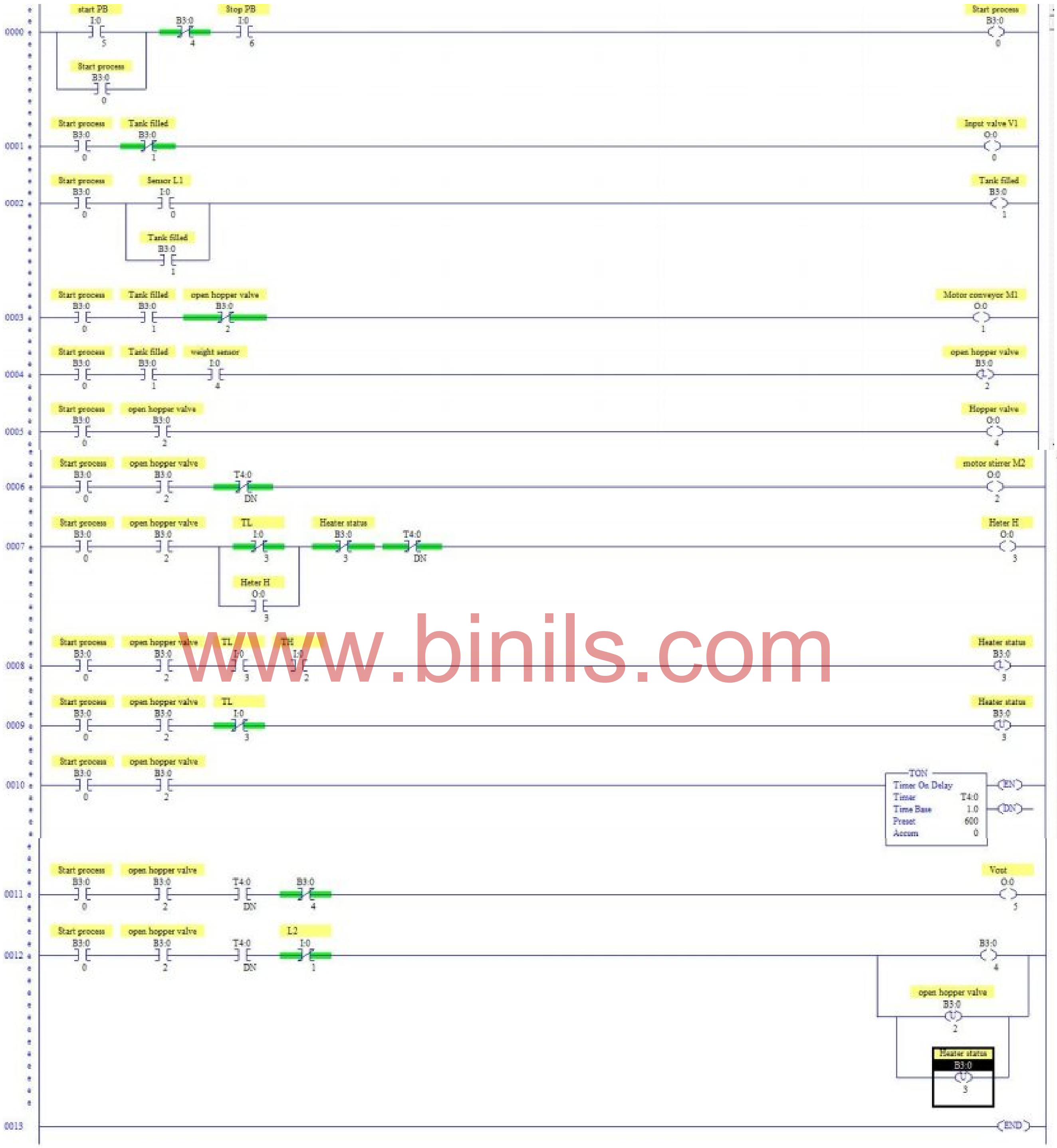
3.18 EB to Generator Changeover:



Sequence of operation:

i)	Rung 0:	When the Start push button is pressed, the NO instruction I:0/1 is set to HIGH. The Rung 0 is TRUE and allows to set internal memory B3:0/0 to HIGH. The NO contact internal memory B3:0/0 to high in rung 0, rung 1 and 2. No contact B3:0/0 in rung provides sealing effect.
ii)	Rung 1:	i) Three sensors can be used to sense the availability of phases. If 'R', 'Y' and 'B' phases are available, the corresponding NO contact I:0/2, I:0/3 and I:0/4 will set to high. Now EB coil which is connected with output instruction O:0/0 and OFF delay Timer instruction T4:0 are energized. Now EB supply lines will be connected to the load circuit. ii) If any one of the three phases are not-available, the corresponding NO contact will set to LOW. Now EB coil which is connected with output instruction O:0/0 and OFF delay Timer instruction T4:0 are de-energized. Now EB supply lines will be disconnected to the load circuit. Timer T4:0 will operate its NC contact T4:0/DN in rung 2 after preset time delay.
iii)	Rung 2:	i) Because of presence of 'R', 'Y' and 'B' phases, the corresponding NC contact I:0/2, I:0/3 and I:0/4 will set to low. Without time delay Timer T4:0 will open its NC contact T4:0/DN immediately and the EB interlock contact O:0/0 also opened to make rung 2 as FALSE. Now GEN coil which is connected with output instruction O:0/1 and OFF delay Timer T4:1 are set to LOW. ii) If any one of the three phases are not-available, the corresponding NC contact will set to HIGH. EB interlock contact O:0/0 is closed and after preset time delay NC contact T4:0/DN will come to closed condition to make rung 1 as TRUE. Now GEN coil which is connected with output instruction O:0/1 and OFF delay Timer T4:1 are set to HIGH. Without time delay Timer T4:1 will open its NC contact T4:1/DN immediately and the GEN interlock contact O:0/1 also opened to make rung 1 as FALSE.
iv)	Rung 0:	When the stop push button is pressed, the NC instruction I:0/0 is set to LOW. Now Rung 0 is FALSE and deenergise remaining coils in remaining rungs.

3.19 BATCH PROCESS:



Sequence of Operation:

- When the start push button is pressed, input valve V1 opens, allowing water to come in the tank.
- In case water goes above high level L1, the input valve closes.
- The motor conveyor turns ON allowing weighted quantity of dry material to pour in to the hopper. When the weight sensor is ON, the conveyor motor turns OFF.
- The hopper valve turns open allowing the weighted dry material to add to the liquid.
- The stirrer motor and heater turn ON to stir the liquid. The heater turns ON to maintain the temperature between TL and TH for a period of 10 minutes. Both motor, stirrer and heater turns OFF.
- The output valve V_{out} turns ON, to empty the mixture down to level Le (LOW). When the mixer level goes below the low level L2 output valve V_{out}closes.
- Close hopper valve. Once the hopper valve closespress preset/stop push button.

3.20 Elevator system:

The elevator as a control system has a number of components. These can basically be divided into the following:

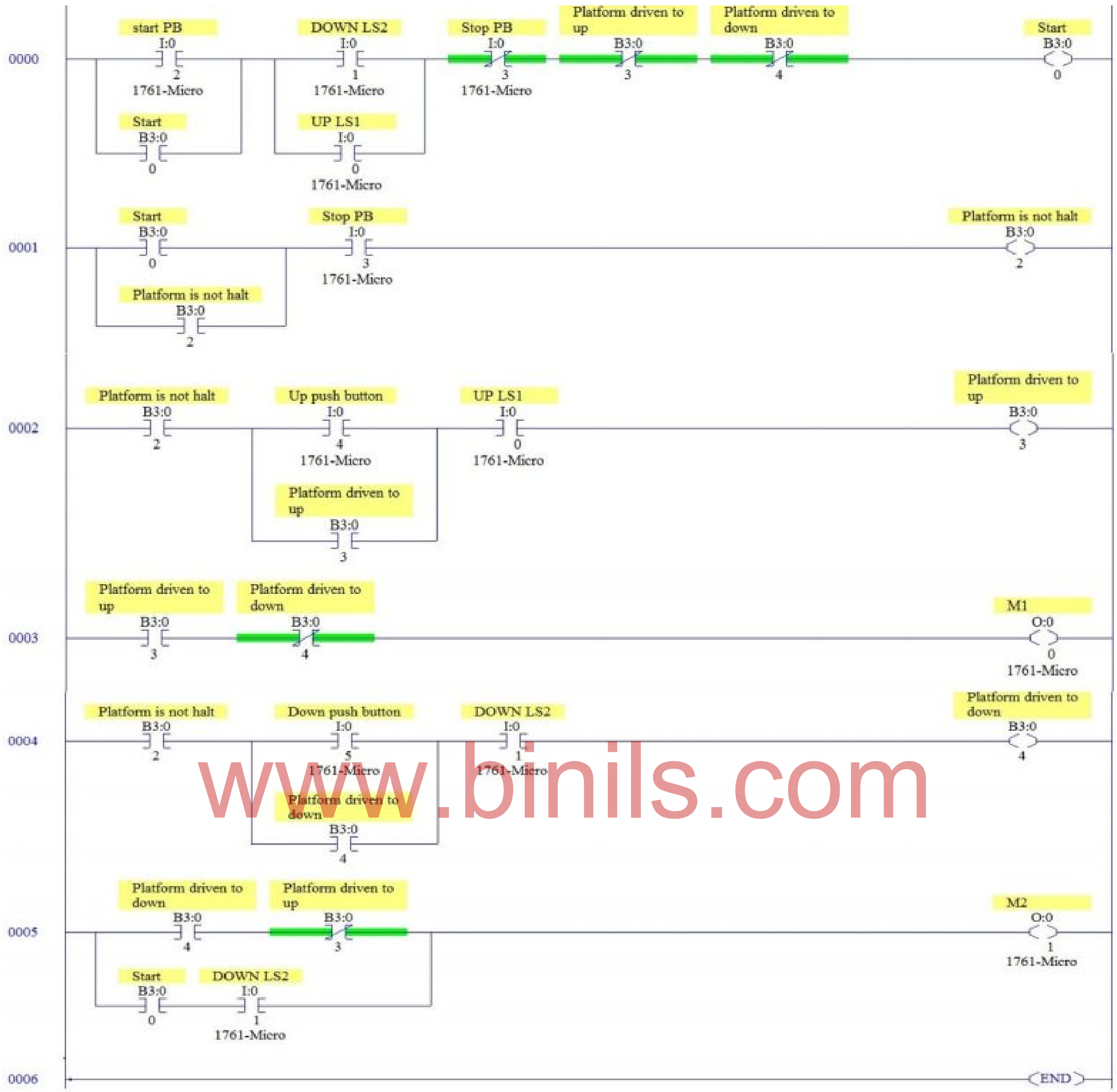
1. Inputs.
2. Outputs.

1- Inputs, which include:

A- Sensors. (Limit switch)
B- Buttons. (Start and Stop push button)
C- Key controls/hall switch (Up/ Down push button)

2- Outputs, which include:

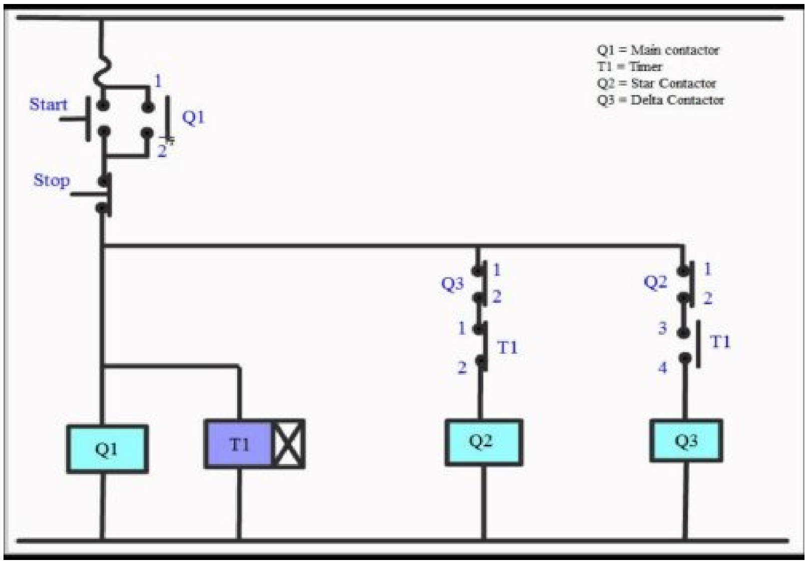
A- Actuators (Motors(Up/Down)).



Sequence of Operation:

- When the start push button is pressed, the platform is driven to the down position.
- When the stop push button is pressed, the platform is halted at whatever the position it occupies at that time.
- When the up push button is pressed, the platform if it is not in a down motion, is driven to the up position.
- When the down push button is pressed, the platform if it is not in a up motion, is driven to the down position.

3.21 Automatic Star Delta Starter:



Control circuit of star delta starter

Sequence of operation:

i)	Rung 0:	When the Start push button is pressed, the NO instruction I:0/1 is set to HIGH. The Rung 0 is TRUE and allows to energise Star coil which is connected to output instruction O:0/0. The NO contact of output instruction O:0/0 is set to high and provides sealing effect.
ii)	Rung 1:	The NO contact of star coil output instruction O:0/0 is set to high and allows to energise Main coil which is connected to output instruction O:0/1 and ON delay Timer instruction T4:0. Now the motor is started with star connection.
iii)	Rung 2:	Timer T4:0 will closed its NO contact T4:0/DN after the preset time delay. Now the motor is disconnected from star connection. (Neither star nor delta).
iv)	Rung 3:	After the preset time delay, T4:0/DN is closed and allows to energise ON delay Timer T4:1.
v)	Rung 4:	After the preset time delay of T4:1, it closes its NO contact T4:1/DN and allows to energise Delta coil which connected to output instruction O:0/2. Timer T4:1 is used to provide pause time between star to delta transition. Now the Motor continues to run with delta connection.
vi)	Rung 5:	When the stop push button is pressed, the NC instruction I:0/0 is set to LOW. Now Rung 0 is FALSE and deenergise star coil, main coil and delta coils.

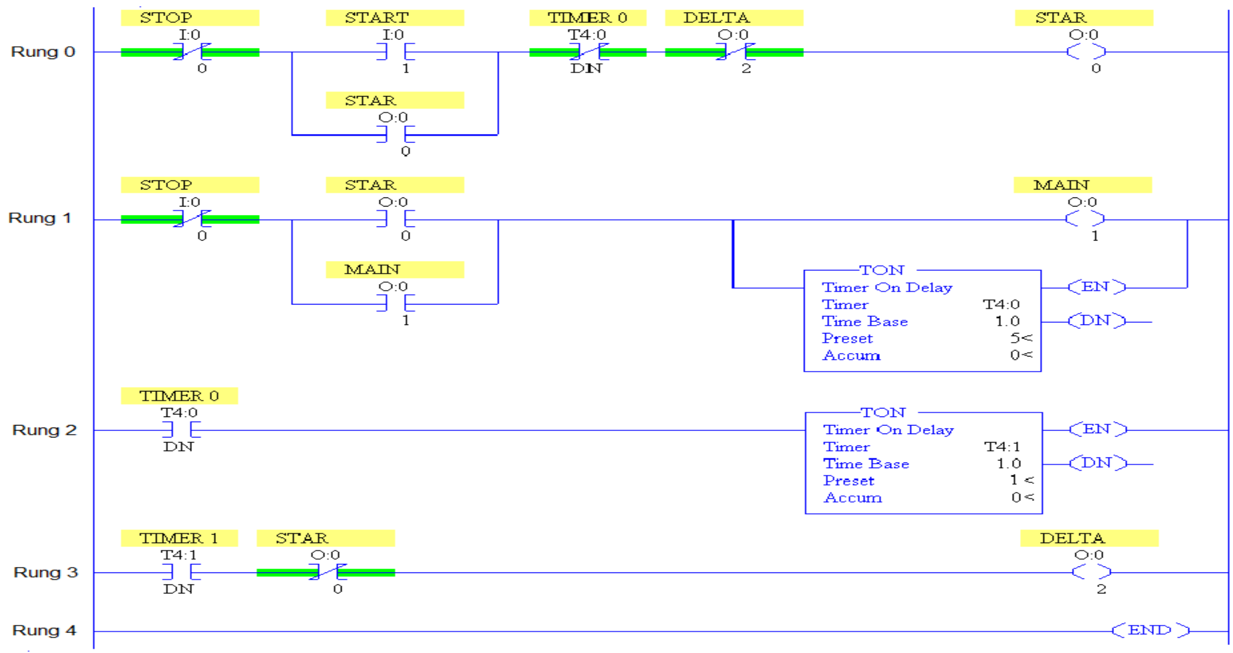


Figure : PLC Ladder Diagram for Star-Delta Starter (Allen Bradley PLC)

3.22 Traffic light control:

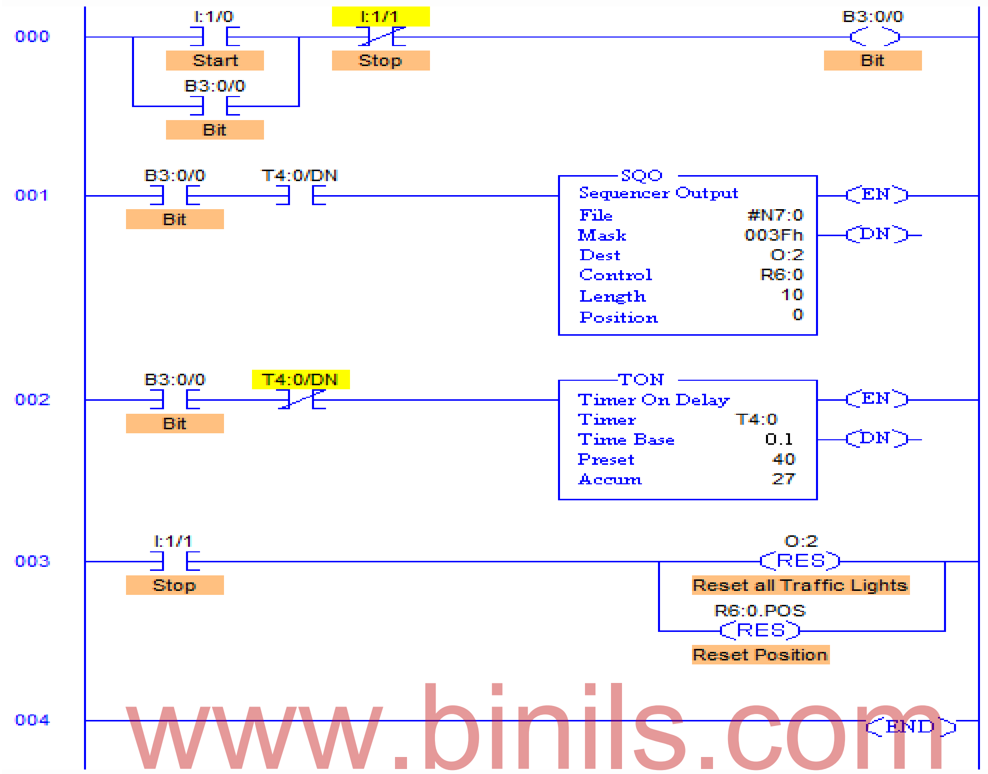
Description

Implement controlling of Traffic Lights and Pedestrian Lights using PLC in Ladder Diagram programming language.

Problem Solution

- As we have solved Traffic Lights problem, similarly we can solve this problem using Sequencer Output SQO instruction.
- In this program, two bits and outputs are added.
- File length, Control, File and Destination remain same.
- Mask data changes due to 2 added outputs.
- As 2 outputs are added, Mask will now have value 00FFh as total numbers of bits used are 8. In order to pass all 8bits, data flow is masked with 11111111 and moved to output.
- When Green light of South-North is ON, Pedestrian Light and Red light of East-West should be ON and vice-versa.
- Use Coil to Master Start and Stop the entire process.

- While using ordinary method to Master Start and Stop, when stop is pressed, the process is just paused and is not entirely reset, hence resetting of Position in SQO instruction and Outputs using the same Stop PB can be done manually.




```
List of Inputs and Outputs
 I:1/0  = Start                                  (Input)
 I:1/1  = Stop                                   (Input)
 B3:0/0 = Latched Coil Bit                       (Bit)
 T4:0   = Timer to update output sequence        (Timer)
 SQO    = Sequencer output                       (Sequencer)
 O:2/0  = North-South Green Light                (Output)
 O:2/1  = North-South Yellow Light               (Output)
 O:2/2  = North-South Red Light                  (Output)
 O:2/3  = East-West Green Light                  (Output)
 O:2/4  = East-West Yellow Light                 (Output)
 O:2/5  = East-West Red Light                    (Output)
```

Sequence of operation:

i)	Rung 0:	Master Start and Stop the process.
ii)	Rung 1:	• File; #N7:0 and File length is 10, hence output sequence is varied from N7:0 to N7:10 with each input. • Destination is set to O:2 hence with each transition, N7:0 to N7:10 are moved to O:2 with masking. • O:2/0 to O:2/5 are used as the output address to Traffic Lights and hence Mask has value 003Fh which means data flow of N7:0/0…N7:10/0 to N7:0/5…N7:10/5 is passed and the remaining N7:0/6…N7:10/6 to N7:0/15…N7:10/15 are blocked. • Control parameters are assigned to register R6:0.
iii)	Rung 2:	• Time base is set to 4secs, hence after every 4secs, output sequence is changed to its next register pattern outputs which is then transferred to O:2 and O:2/0 to O:2/5 are energized accordingly. • As we can see, from N7:1 to N7:4 have the same bit pattern. So, these bits are set to 1 for 4 cycles that is 16secs. These bits are used for South-North Green light and East-West Red light. • Similarly the entire sequence is followed.
iv)	Rung 3:	When Stop I:1/1 is pressed, Position is reset to 0 and all the outputs are de-energized.

UNIT – III PLC ROGRAMMING

MODEL QUESTIONS

Two Mark Questions:

1. Name the different types of PLC programming languages.
2. Expand the term FBD and STL.
3. Expand the term SFC and ST.
4. What is ladder diagram?
5. Give any 2 relay type instructions.
6. Explain EXAMINE IF CLOSED instruction and assign its address.
7. Explain EXAMINE IF OPEN instruction and assign its address.
8. Draw the symbol of LATCH instruction and assign the address
9. Draw the symbol of UNLATCH instruction and assign the address
10. State the different types of timer instruction.
11. What is a TON instruction?
12. What is a TOF instruction?
13. State the different types of timer instruction.
14. What is a CTU instruction?
15. What is a CTD instruction?
16. What is preset in timer instruction?
17. What is accumulator in timer instruction?
18. What is preset in counter instruction?
19. What is accumulator in counter instruction?
20. What is the need of cascaded timer instruction?
21. What is the need of cascaded counter instruction?

Three Mark Questions:

1. Write short notes on ladder diagram.
2. Draw the schematic diagram of scanning of ladder diagram.
3. Write short notes on FBD
4. Write shot notes on STL
5. Write short note on XIC instruction.
6. Write short note on XIO instruction.
7. Write short note on OTE instruction.
8. Write short note on OTL instruction.
9. Write short note on OTU instruction.
10. Write short note on OSR instruction.
11. Why do we need timer instructions?
12. Explain various parameters associated with timer instruction.

13. Develop simple ladder logic using TON instruction.
14. Develop simple ladder logic using TOF instruction.
15. Develop simple ladder logic using CTU instruction.
16. Develop simple ladder logic using CTD instruction.
17. What is a RTO instructions?
18. Why do we need counter instructions?
19. Explain various parameters associated with counter instruction.

Ten Mark Questions:

1. Briefly discuss about different types of PLC programming languages.
2. With neat symbol explain about different types of relay type instructions.
3. What is a TON instruction? Explain its working with the help of suitable ladder diagram and timing diagram.
4. What is a TOF instruction? Explain its working with the help of suitable ladder diagram and timing diagram.
5. Explain the operation of cascaded timers with suitable ladder diagram.
6. What is a CTU instruction? Explain its working with the help of suitable ladder diagram and timing diagram.
7. What is a CTD instruction? Explain its working with the help of suitable ladder diagram and timing diagram.
8. Make UP-DOWN counter using CTU and CTD instructions and explain its working.
9. Explain the operation of cascaded counters with suitable ladder diagram.
10. Develop ladder logic for DOL starter and explain the sequence of operation.
11. Develop ladder logic for star delta starter and explain the sequence of operation
12. Develop ladder logic for rotor resistance starter and explain the sequence of operation.
13. Develop ladder logic for EB to Generator changeover and explain the sequence of operation.

UNIT- 4 NETWORKING

OBJECTIVES:

After completing this chapter, students should able to:

- ✓ Explain the functionality of different levels of industrial network
- ✓ Explain the concept of network topology and network protocols
- ✓ Explain the concept of I/O bus networks etc.,

4.1 Levels of industrial network:

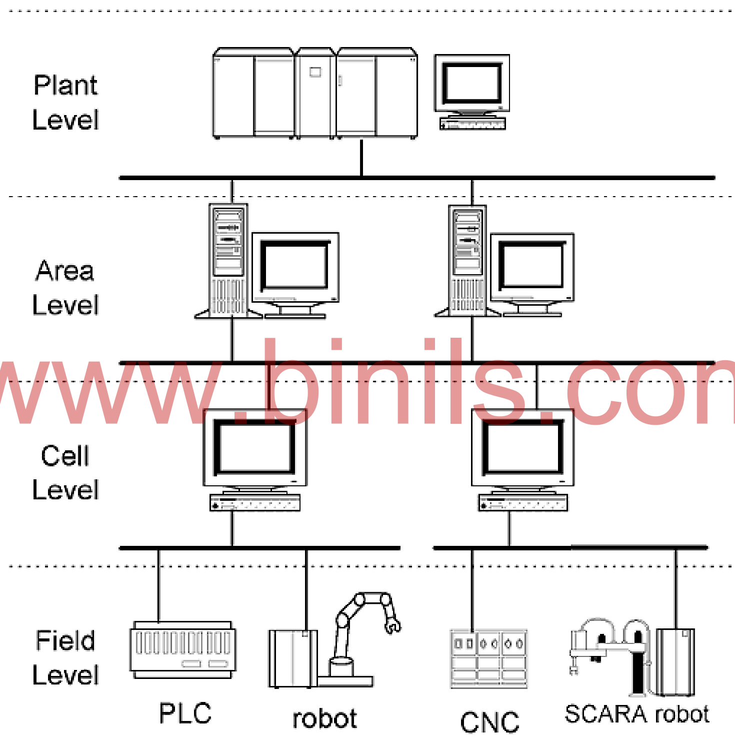


Figure 4.1: Levels of Industrial Networking

Field level:

The lowest level of the automation hierarchy is the field level which includes the field devices such as actuators and sensors. The task of the devices in the field level is to transfer data between the manufactured product and the technical process. The data may be either binary and analog. Nowadays, the field bus is often used for information transfer in the field level.

Cell level:

At the cell level, the information flow mainly consists of the loading of programs, parameters and data. In processes with short machine idle times and readjustments, this is done during the production process. In small controllers it may be necessary to load subroutines during one manufacturing cycle. This determines the timing requirements. For the cell level operations, machine synchronizations and event handlings may require short response times on the bus. In order to achieve the communication requirements in this level, local area networks have been used as the communication network.

Area level:

The area level consists of cells combined into groups. Cells are designed with an application-oriented functionality. By the area level controllers or process operators, the controlling and intervening functions are made such as the setting of production targets, machine start up and shutdown, and emergency activities.

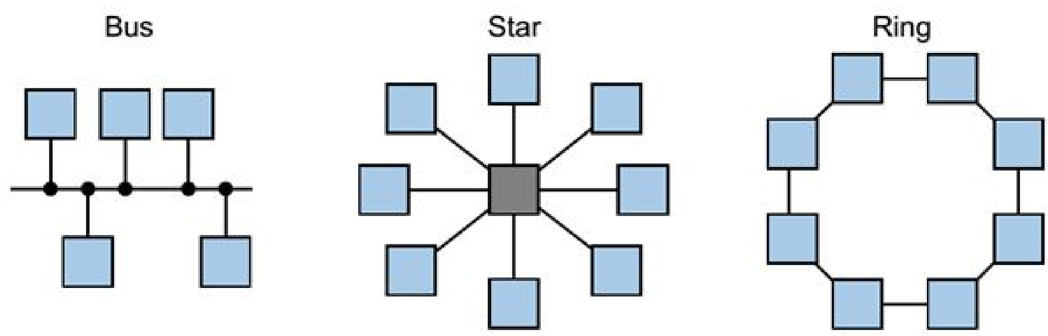
Plant level:

The plant level is the top level of a plant or an industrial automation system. The plant level controller gathers the management information from the area levels, and manages the whole automation system.

4.2 Network Topology:

The term network topology refers to the shape of how the computers and other network components are connected to each other.

The topology of a local area network is the geometry of the network, or how individual nodes are connected to it. The basic network topologies used today are star, common bus, and ring.



Bus, Star and Ring Topologies

Figure 4.2: Types of Network Topology

4.2.1 Star Topology:

In a star topology, a central control device or hub is connected to several nodes. This configuration allows for bidirectional communication between the central control device and each node. All transmission must be sent to the hub which then sends them to the correct node. The first PLC networks consisted of a multiport host computer with each port connected to the programming port of a PLC. Figure shows this arrangement, known as star topology. The network controller can be either a computer, a PLC, or another intelligent host.

The main advantage of this topology is that it can be implemented with a simple point to-point protocol—that is, each node can transmit whenever necessary.

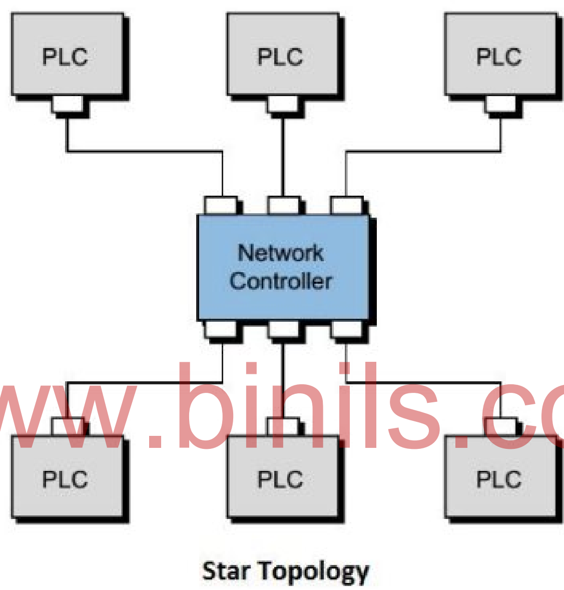


Figure 4.3: Star Topology

Disadvantages of Star topology:

- All nodes are dependence on a central node.
- The wiring costs are high for large installations.
- Messages between two nodes must pass through the central node (low throughput).
- Failure of the central node will crash the network.

4.2.2 Common bus topology:

The common bus topology has a main trunk line to which individual PLC nodes are connected in a multidrop fashion. A coaxial cable with proper terminators is typically the communication medium for the trunkline. In contrast to the star topology, communication in a common bus network can occur between any two nodes without passing information through a network controller. An inherent problem of this scheme, however, is determining which node may transmit at which time, to avoid data collision.

4.2.2 Common bus topology:

The common bus topology has a main trunk line to which individual PLC nodes are connected in a multidrop fashion. A coaxial cable with proper terminators is typically the communication medium for the trunkline. In contrast to the star topology, communication in a common bus network can occur between any two nodes without passing information through a network controller. An inherent problem of this scheme, however, is determining which node may transmit at which time, to avoid data collision.

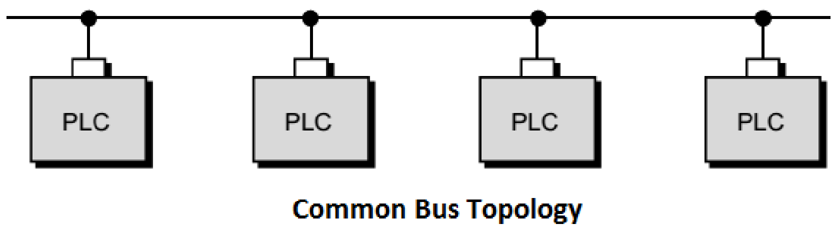


Figure 4.4: Common Bus Topology

Common bus topologies are very useful in distributed control applications, since each station has equal independent control capability and can exchange information at any given time. Also, this topology requires little reconfiguration to add or remove stations from the network. The main disadvantage of this topology is that all of the nodes depend on a common bus trunk line. A break in this trunk line can affect many nodes.

4.2.3 Ring Topology:

Ring topology, is not used in industrial environments because failure of any node will crash the network, unless the failed node is bypassed. Thus, it is a good candidate for fiber-optic networks, since fiber-optic transmission media allows fast communication speed and long distance connectivity.

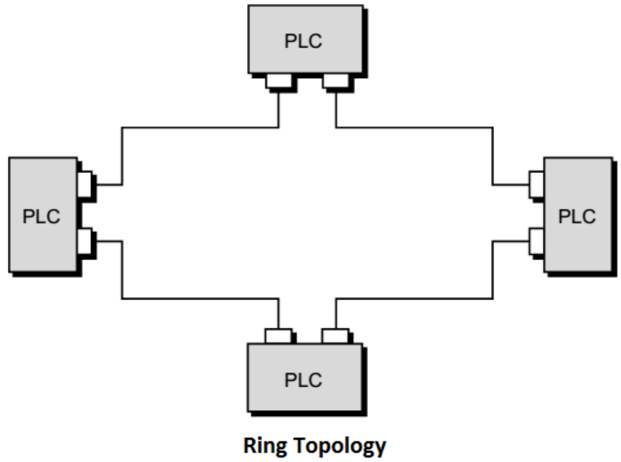


Figure 4.5: Ring Topology

4.3 Network Protocol:

A protocol is a set of rules that two or more devices must follow if they are to communicate with each other. Protocol includes everything from the meaning of data to the voltage levels on connection wires. A network protocol defines how a network will handle the following problems and tasks:

1. Communication line errors
2. Flow control (to keep buffers from overflowing)
3. Access by multiple devices
4. Failure detection
5. Data translation
6. Interpretation of messages

4.4 OSI Reference Model:

Networks follow a protocol to implement the transmission and reception of data over the network medium (e.g., coaxial cable). In 1979, the International Standards Organization (ISO) published the Open Systems Interconnection (OSI) reference model, also known as the ISO IS 7498, to provide guidelines for network protocols. This model defined seven layers of functions in a communication system.

In network protocols, the physical layer (layer 1) and the medium access control sub-layer (layer 2A) are usually implemented with hardware, while the remaining layers are implemented using software. The hardware components of layers 1 and 2A are generally referred to as modems (or transceivers) and drivers (or controllers), respectively.

Structure of simplified OSI model:

1) Physical Layer:

The physical layer is concerned with transmitting raw bits over a Communication channel. The design issues have to do with making sure that when one side sends a 1 bit, it is received by the other side as a 1 bit, not as a 0 bit.

The physical layer defines the electrical and physical specifications of the data connection. It defines the relationship between a device and a physical transmission medium (for example, an electrical cable, an optical fiber cable, or a radio frequency link). This includes the layout of pins, voltages, line impedance, cable specifications, signal timing and similar characteristics for connected devices and frequency (5 GHz or 2.4 GHz etc.) for wireless devices. It is responsible for transmission and reception of unstructured raw data in a physical medium. Bit rate control is done at the physical layer. It may define transmission mode as simplex, half duplex, and full duplex. It defines the network topology as bus, mesh, or ring being some of the most common.

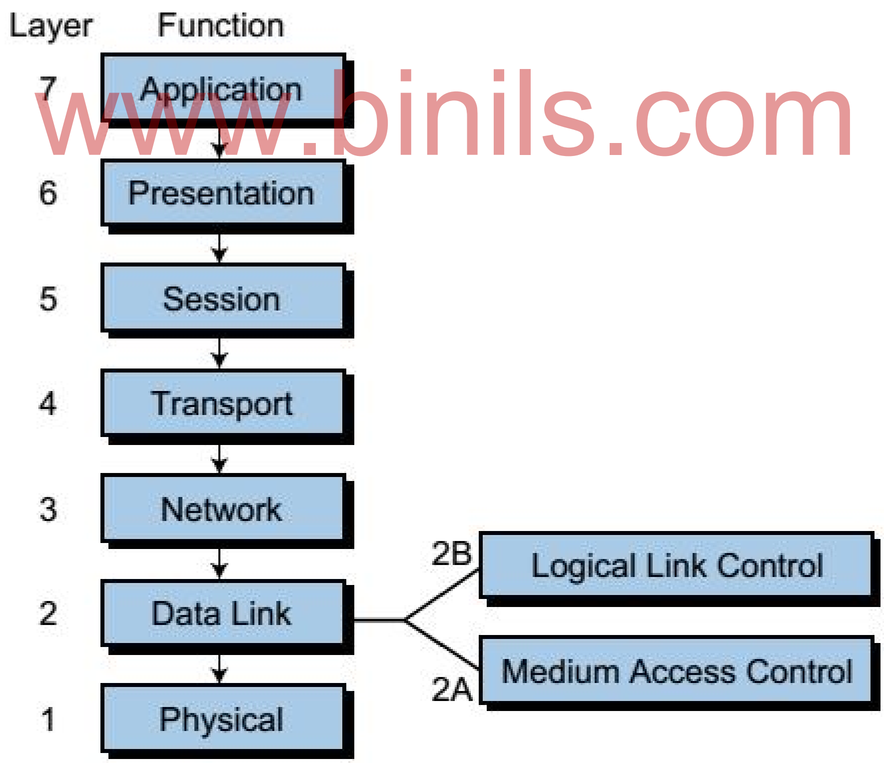


Figure 4.6 : OSI Reference Model

2) Data Link Layer

It is responsible for transmitting a group of bits between adjacent nodes. The group of bits so formed is called frame or packet. This layer adds header and trailer information to the data unit. If the service is reliable, the receiver confirms correct receipt of each frame by sending back an acknowledgement frame.

IEEE 802 divides the data link layer into two sub layers:

- Medium access control (MAC) layer – responsible for controlling how devices in a network gain access to a medium and permission to transmit data.
- Logical link control (LLC) layer – responsible for identifying and encapsulating network layer protocols, and controls error checking and frame synchronization.

The MAC and LLC layers of IEEE 802 networks such as 802.3 Ethernet, 802.11 Wi-Fi, and 802.15.4 ZigBee operate at the data link layer.

The Point-to-Point Protocol (PPP) is a data link layer protocol that can operate over several different physical layers, such as synchronous and asynchronous serial lines.

3) Network Layer:

This is responsible for addressing messages and data so they are sent to the correct destination, and for translating logical addresses and names (like a machine name FLAME) into physical addresses. This layer is also responsible for finding a path through the network to the destination computer.

4) Transport Layer:

Ensures that data is delivered error free, in sequence and with no loss, duplications or corruption. This layer also repackages data by assembling long messages into lots of smaller messages for sending, and repackaging the smaller messages into the original larger message at the receiving end.

5) Session Layer:

Allows two applications to establish, use and disconnect a connection between them called a session. Provides for name recognition and additional functions like security, which are needed to allow applications to communicate over the network

6) Presentation Layer:

Presentation layer is concerned with the syntax and semantics of the information transmitted. In order to make it possible for computers with different data representations to communicate, the data structures to be exchanged can be defined.

7) Application Layer:

It is the highest layer in the OSI model. Login, password checking, file transfer, etc., are some of the functions of the application layer. The application layer contains a variety of protocols that are commonly needed by users. One widely-used application protocol is HTTP (Hyper Text Transfer Protocol), which is the basis for the World Wide Web.

4.5 TCP/IP Protocol:

The **Internet protocol suite** is the conceptual model and set of communications protocols used on the Internet and similar computer networks. It is commonly known as **TCP/IP** because the foundational protocols in the suite are the Transmission Control Protocol (TCP) and the Internet Protocol (IP). It is occasionally known as the **Department of Defense (DoD) model**, because the development of the networking method was funded by the United States Department of Defense through DARPA.

The Internet protocol suite provides end-to-end data communication specifying how data should be packetized, addressed, transmitted, routed, and received. This functionality is organized into four abstraction layers which classify all related protocols according to the scope of networking involved. From lowest to highest, the layers are the link layer, containing communication methods for data that remains within a single network segment (link); the internet layer, providing internetworking between independent networks; the transport layer handling host-to-host communication; and the application layer, which provides process-to-process data exchange for applications.

TCP/IP protocol suite is a collection of protocols used for the internet services. The TCP/IP suite contains networking protocols used for setting up the internet.

Application Layer
Transport layer
Internet layer
Host to Network layer

Figure 4.7: TCP/IP REFERENCE MODEL

i) Application Layer:

Application layer consists of all higher level protocols such as virtual terminal (TELNET), file transfer (FTP) and electronic mail. The virtual terminal protocol allows a user on one machine to log on to a remote machine. Domain Name Service (DNS) is used for mapping host names onto their network addresses.

ii) Transport Layer

Host-to-Host or Transport layer provides end-to-end data transfer service. This layer may include reliability mechanism. It hides the details or underlying network or networks from the application layer. It provides support to TCP or UDP protocol.

The transport layer includes the following two transport level protocols:

1. Datagram Congestion Control Protocol (DCCP)
2. Stream Control Transmission Protocol (SCTP).

iii) Internet layer:

The internet layer holds the whole architecture together. It defines a packet format and protocol called IP (Internet Protocol). Internet layer is concerned with routing of data. It handles machine to machine communications.

iv) The Host to Network Layer:

Below the internet layer is great void. It is responsible for accepting and transmitting IP datagram. It also point out that the host has connect to the network using some protocol so it can transmit IP packets over it. This protocol is not specified and varies from host to host and network to network.

Understanding I/P address

An IP address is an address used in order to uniquely identify a device on an IP network. The address is made up of 32 binary bits, which can be divisible into a network portion and host portion with the help of a subnet mask. The 32 binary bits are broken into four octets (1 octet = 8 bits). Each octet is converted to decimal and separated by a period (dot). For this reason, an IP address is said to be expressed in dotted decimal format (for example, 172.16.81.100). The value in each octet ranges from 0 to 255 decimal, or 00000000 - 11111111 binary.

Here is how binary octets convert to decimal: The right most bit, or least significant bit, of an octet holds a value of 2^{0}. The bit just to the left of that holds a value of 2^{1}. This continues until the left-most bit, or most significant bit, which holds a value of 2^{7}. So if all binary bits are a one, the decimal equivalent would be 255 as shown here:

1	1	1	1	1	1	1	1
128	64	32	16	8	4	2	1 (128+64+32+16+8+4+2+1=255)

Here is a sample octet conversion when not all of the bits are set to 1.

0	1	0	0	0	0	0	1
0	64	0	0	0	0	0	1 (0+64+0+0+0+0+0+1=65)

And this sample shows an IP address represented in both binary and decimal.

10. 1. 23. 19 (decimal)

00001010.00000001.00010111.00010011 (binary)

These octets are broken down to provide an addressing scheme that can accommodate large and small networks. There are five different classes of networks, A to E. This document focuses on classes A to C, since classes D and E are reserved and discussion of them is beyond the scope of this document.

4.6 Introduction to I/O Bus Networks:

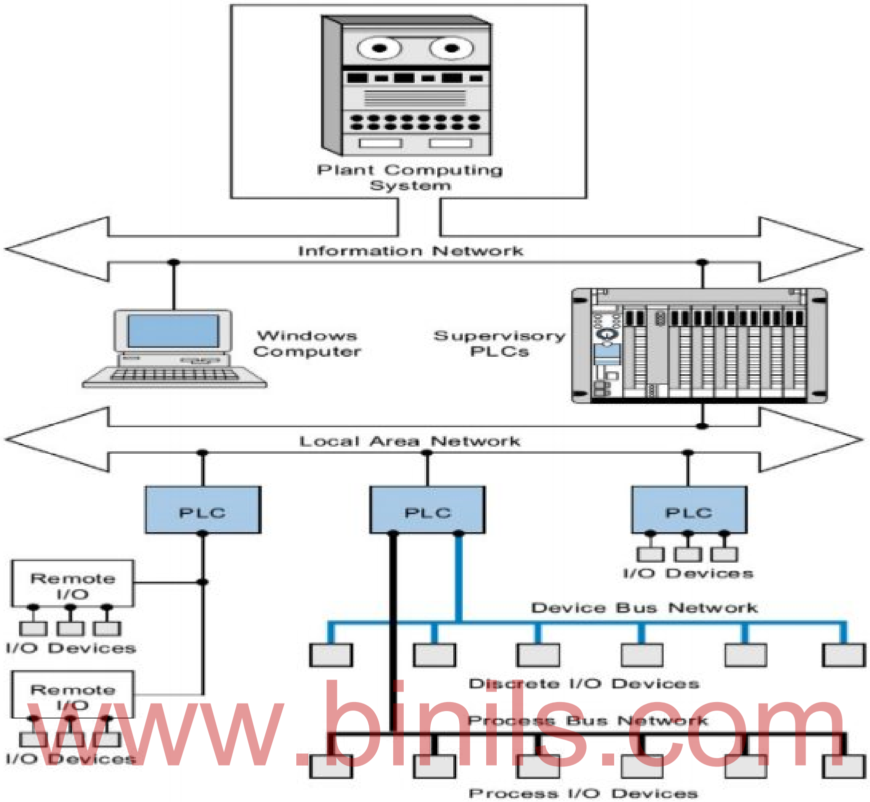


Figure 4.8: Block Diagram of I/O Bus Network

The basic function of an I/O bus network is to communicate information with, as well as supply power to, the field devices that are connected to the bus. In an I/O bus network, the PLC drives the field devices directly, without the use of I/O modules; therefore, the PLC connects to and communicates with each field I/O device according to the bus's protocol. These devices communicate not only the ON/OFF state of input and output controls, but also diagnostic information about their operating states. For example, a photoelectric sensor (switch) can report when its internal gain starts to decrease because of a dirty lens, or a limit switch can report the number of motions it has performed. This type of information can prevent I/O device malfunction and can indicate when a sensor has reached the end of its operating life, thus requiring replacement.

4.7 Types of I/O Bus Networks:

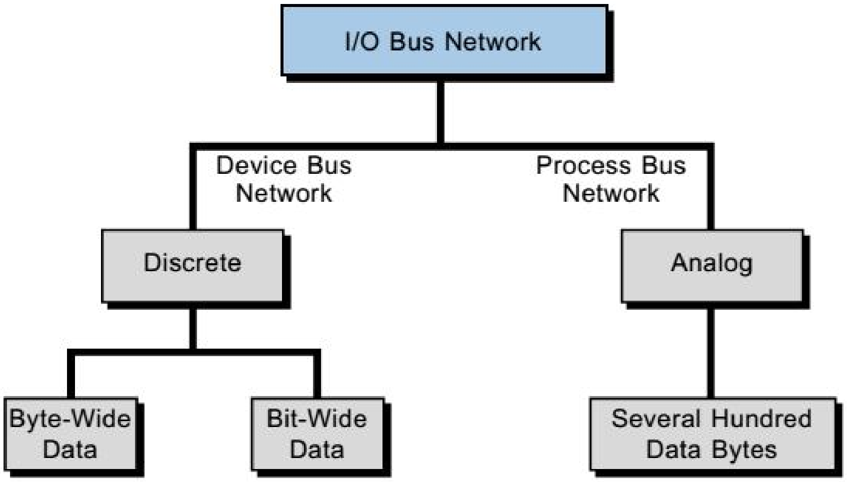


Figure 4.9: Types of I/O Bus Network

Device bus networks:

Device bus networks interface with low-level information devices (e.g., push buttons, limit switches, etc.), which primarily transmit data relating to the state of the device (ON/OFF) and its operational status (e.g., operating OK). Device bus networks that include discrete devices, as well as small analog devices, are called byte-wide bus networks. These networks can transfer between 1 and 50 or more bytes of data at a time. Device bus networks that only interface with discrete devices are called bit-wide bus networks.

Process bus networks:

Process bus networks, on the other hand, connect with high-level information devices (e.g., smart process valves, flow meters, etc.), which are typically used in process control applications. Process bus networks handle large amounts of data (several hundred bytes), consisting of information about the process, as well as the field devices themselves. Process bus networks work slower because of their large data packet size, so they are more applicable for the control of analog I/O devices, which do not require fast response times.

4.8 Protocols Standard:

Neither of the two I/O bus networks have established protocol standards; however, many organizations are working towards developing both discrete and process bus network specifications. In the process bus area, two main organizations, the Fieldbus Foundation and the Profibus (Process Field Bus) Trade Organization, are working to establish network and protocol standards. Other organizations, such as the Instrument Society of America (ISA) and the European International Electronics Committee (IEC), are also involved in developing these standards. This is the reason why some manufacturers specify that their analog products are compatible with Profibus, Fieldbus, or another type of protocol communication scheme. Figure below illustrates a block diagram of available network and protocol standards.

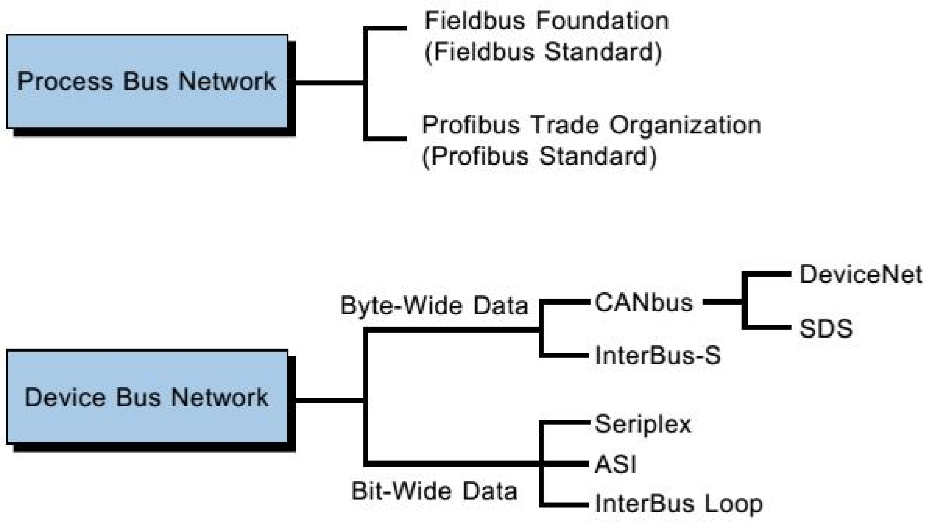


Figure 4.10: Protocol Standard

One of these de facto standards for the byte-wide device bus network is DeviceNet, originally from PLC manufacturer Allen-Bradley and now provided by an independent spin-off association called the Open DeviceNet Vendor Association. Another is SDS (Smart Distributed System) from Honeywell. Both of these device bus protocol standards are based on the control area network bus (CANbus), developed for the automobile industry, which uses the commercially available CAN chip in its protocol. InterBus-S from Phoenix Contact is another emerging de facto standard for byte-wide device bus network.

The de facto standards for low-end, bit-wide device bus networks include Seriplex, developed by Square D, and ASI (Actuator Sensor Interface), a standard developed by a consortium of European companies. Again, this is why I/O bus network and field device manufacturers will specify compatibility with a particular protocol (e.g., ASI, Seriplex, InterBus-S, SDS, or DeviceNet) even though no official protocol standard exists.

4.9 Advantages of I/O Bus networks:

- It allows more than one fielddevice to be connected to a wire due to addressing capabilities.
- PLCs in an I/O bus perform a minimal amount of analog-to-digital and digital-to-analog conversions, since the devices pass their data digitally through the bus to the controller.

- Process bus–compatible field devices can pass a digital value proportional to a real-world value to the PLC, thus eliminating the need to linearize or scale the process data.
- The reduction in the amount of wiring in a plant alone can provide incredible cost savings for manufacturing and process applications.

4.10 Gateway:

Gateways make communication possible between different architectures and protocols. They repackage and convert data going from one network to another network so that the one can understand the other's application data. Gateways can change the format of a message so that it will conform to the application program at the receiving end of the transfer. If network access translation is their only function, the interfaces are known as bridges. If the interface also adjusts data formats or performs data transmission control, then it is called a gateway.

4.11 Token passing:

In a token passing network, a node can transmit data on the network only when it has possession of a token. A token is simply a small packet that is passed from node to node. When a node finishes transmitting messages, it sends a special message to the next node in the sequence, granting it the token. The token passes sequentially from node to node, allowing each an opportunity to transmit without interference. Tokens usually have a time limit to prevent a single node from tying up the token for a long period of time.

4.12 Data Highway:

The Allen-Bradley Data Highway networks, Data Highway Plus (DH+) and DH-485, are proprietary communications networks. They use peer-to-peer communication implementing token passing. The medium is shielded twisted pair cable.

4.13 Serial Communication:

Serial data communication is implemented using standards such as RS-232, RS-422, and RS-485. The RS in the standard's name means Recommended Standard that specifies the electrical, mechanical, and functional characteristics for serial communications. The simplest type of connection is the RS-232 serial port. The RS-232 type of serial transmission is designed to communicate between one computer and one controller and is usually limited to lengths up to 50 feet. RS-422 and RS-485 serial transmission types are designed to communicate between one computer and multiple controllers, have a high level of noise immunity, and are usually limited to lengths of 650 feet (for RS-485) or 1650 feet (for RS-422).

4.14 Device Net:

DeviceNet is an open device-level network. It is relatively low speed but efficient at handling the short messages to and from I/O modules. As PLCs have become more powerful, they are being required to control an increasing number of I/O field devices. Therefore, at times it may not be practical to separately wire each sensor and actuator directly into I/O modules. Conventional systems have racks of inputs and outputs with each I/O device wired back to the controller. The DeviceNet protocol dramatically reduces costs by integrating all I/O devices on a 4-wire trunk network with data and power conductors in the same cable. This direct connectivity reduces costly and time-consuming wiring.

4.15 Control Net:

ControlNet is positioned one level above DeviceNet. It uses the Common Industrial Protocol (CIP) to combine the functionality of an I/O network and a peer-to-peer network providing high-speed performance for both functions. This open high-speed network is highly deterministic and repeatable. Electronic device data sheets (EDS-Files) are required for each ControlNet device. During the setup phase the ControlNet scanner must configure each device according to the EDS-Files.

4.16 EtherNet:

EtherNet/IP (Ethernet Industrial Protocol) is an open communications protocol based on the Common Industrial Protocol (CIP) layer used in both DeviceNet and ControlNet. It allows users to link information seamlessly between devices running the EtherNet/IP protocol without custom hardware.

4.17 Modbus:

Modbus is a serial communication protocol originally developed by Modicon for use with its PLCs. Basically, it is a method used for transmitting information over serial lines between electronic devices. The device requesting the information is called the Modbus Master and the devices supplying information are Modbus Slaves. Modbus is an open protocol, meaning that it’s free for manufacturers to build into their equipment without having to pay royalties. It has become a standard communications protocol in industry, and is one of the most commonly available means of connecting industrial electronic devices.

4.18 Fieldbus:

Fieldbus is an open, serial, two-way communications system that interconnects measurement and control equipment such as sensors, actuators, and controllers. At the base level in the hierarchy of plant networks, it serves as a network for field devices used in process control applications.

4.19 Profibus:

PROFIBUS-DP (where DP stands for Decentralized Periphery) is an open, international fieldbus communication standard that supports both analog and discrete signals. It is functionally comparable to DeviceNet. The physical media are defined via the RS-485 or fiber optic transmission technologies. PROFIBUS-DP communicates at speeds up to 12 Mbps over distances up to 1200 meters.

4.10 Subnetting:

Subnetting is a technique that lets network administrators use the 32 bits available in an IP address more efficiently by creating networks that aren't limited to the scales provided by Class A, B, and C IP addresses. With subnetting, you can create networks with more realistic host limits. Subnetting provides a more flexible way to designate which portion of an IP address represents the network ID and which portion represents the host ID. With standard IP address classes, only three possible network ID sizes exist: 8 bits for Class A, 16 bits for Class B, and 24 bits for Class C. Subnetting lets you select an arbitrary number of bits to use for the network ID.

Two reasons compel people to use subnetting. The first is to allocate the limited IP address space more efficiently. If the Internet was limited to Class A, B, or C addresses, every network would be allocated 254, 65 thousand, or 16 million IP addresses for host devices. Although many networks with more than 254 devices exist, few (if any) exist with 65 thousand, let alone 16 million. Unfortunately, any network with more than 254 devices would need a Class B allocation and probably waste tens of thousands of IP addresses.

The second reason for subnetting is that even if a single organization has thousands of network devices, operating all those devices with the same network ID would slow the network down to a crawl. The way TCP/IP works dictates that all the computers with the same network ID must be on the same physical network. The physical network comprises a single broadcast domain, which means that a single network medium must carry all the traffic for the network. For performance reasons, networks are usually segmented into broadcast domains that are smaller than even Class C addresses provide.

Example

In order to subnet a network, extend the natural mask with some of the bits from the host ID portion of the address in order to create a subnetwork ID. For example, given a Class C network of 204.17.5.0 which has a natural mask of 255.255.255.0, you can create subnets in this manner:

204.17.5.0 - 11001100.00010001.00000101.00000000

255.255.255.224 - 11111111.11111111.11111111.11100000

--------------------------|sub|----

By extending the mask to be 255.255.255.224, you have taken three bits (indicated by "sub") from the original host portion of the address and used them to make subnets. With these three bits, it is possible to create eight subnets. With the remaining five host ID bits, each subnet can have up to 32 host addresses, 30 of which can actually be assigned to a device *since host ids of all zeros or all ones are not allowed* (it is very important to remember this). So, with this in mind, these subnets have been created.

204.17.5.0 255.255.255.224 host address range 1 to 30

204.17.5.32 255.255.255.224 host address range 33 to 62

204.17.5.64 255.255.255.224 host address range 65 to 94

204.17.5.96 255.255.255.224 host address range 97 to 126

204.17.5.128 255.255.255.224 host address range 129 to 158

204.17.5.160 255.255.255.224 host address range 161 to 190

204.17.5.192 255.255.255.224 host address range 193 to 222

204.17.5.224 255.255.255.224 host address range 225 to 254

4.21 Subnet Mask :

A subnet mask is a number that defines a range of IP addresses that can be used in a network. Subnet masks are used to designate subnetworks, or subnets, which are typically local networks LANs that are connected to the Internet. Systems within the same subnet can communicate directly with each other, while systems on different subnets must communicate through a router. Therefore, subnetworks can be used to partition multiple networks and limit the traffic between them.

A subnet mask hides, or "masks," the network part of a system's IP address and leaves only the host part as the machine identifier. A common subnet mask for a Class C IP address is 255.255.255.0. Each section of the subnet mask can contain a number from 0 to 256, just like an IP address. Therefore, in the example above, the first three sections are full, meaning the IP addresses of computers within the subnet mask must be identical in the first three sections. The last section of each computer's IP address can be anything from 0 to 255. For example, the IP addresses 10.0.1.201 and 10.0.1.202 would be in the same subnet, while 10.0.2.201 would not. Therefore, a subnet mask of 255.255.255.0 allows for close to 256 unique hosts within the network.

SUBNET MASK IN IP ADDRESSING

Class A Subnet Mask	Netwok	Host	Host	Host
	255	0	0	0
Class B Subnet Mask	Netwok	Network	Host	Host
	255	255	0	0
Class C Subnet Mask	Netwok	Network	Network	Host
	255	255	255	0

4.22 File Transfer protocol:

File transfer protocol is the standard mechanism provided by the internet for copying a file from one host to another. FTP is a TCP/IP client-server application for copying files from one host to another. The problem occurred during file transfer is solved by FTP.

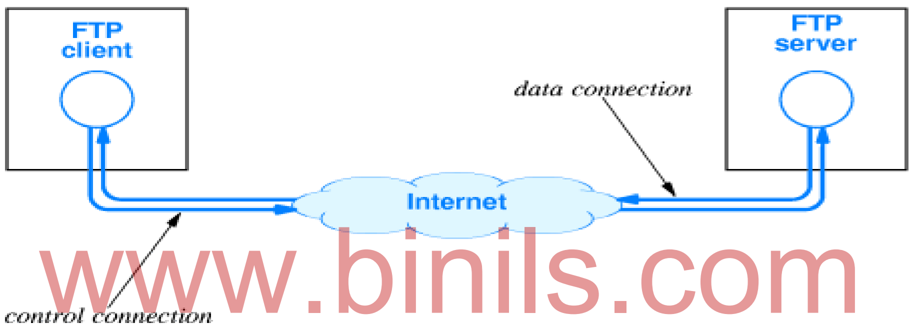


Fig: Block diagram of FTP

The client has three components:

1. User Interface
2. Client control process
3. Client data transfer process

The server has two components:

1. Server control process
2. Server data transfer process

FTP uses the service of TCP. It needs two TCP connections. Port 21 is used for the control connection and port 20 is used for the data connection.

The control connection is made between the data transfer process. The control connection is maintained during the entire interactive FTP session. The data connection is opened and then closed for each file transferred. It opens each time commands that involve transferring files are used, and it closes when the file is transferred. In other words, when a user starts an FTP session, the control connection opens. While the control connection is open, the data connection can be opened and closed multiple times if several files are transferred.

UNIT-4 NETWORKING

Model questions

Part A (2 mark)

1. What is the function of process bus network.
2. Mention the levels of industrial networking.
3. What is plant level?
4. Define network topology.
5. What is the main advantage of star topology.
6. What is protocol?
7. What is physical layer?
8. What is data link layer?
9. What is gateway?
10. Mention the components of FTP?
11. List any two advantages of networking
12. What is the main advantage of star topology.
13.

Part B

1. Explain about star, ring topologies.
2. Explain about physical and presentation layer.
3. Explain about controlnet,profibus.
4. Explain modbus, field bus.
5. Explain about subnetting.
6. What is the function of transport layer in OSI model?

Part C

1. Explain with structure of simplified OSI model.
2. Explain with block diagram of I/O bus networks.
3. Explain the different types of network topologies.
4. Explain with block diagram of FTP.
5. Explain TCP/IP protocol with reference model.

UNIT- 5DATA ACQUISITION SYSTEM

OBJECTIVES:

After completing this chapter, students should able to:

- ✓ Describe the computer control of process
- ✓ Explain the operation of SCADA
- ✓ Explain the functions of the major components of a process control system
- ✓ Explain how on/off control and PID control works

5.1 Computers in Process control:

Process control is the automated control of a process. Such systems typically deal with analog signals from sensors. The ability of a PLC to perform math functions and utilize analog signals makes it ideally suited for this type of control. Manufacturing is based on a series of processes being applied to raw materials. Typical applications of process control systems include chemical-processing industries, petrochemical production, oil refining, basic metals industries (iron and steel, aluminum), and food processing.

5.2 Types of processes control:

Usually, manufacturing industries are classified as

- Process industries
    - i) Continuous process
    - ii) Batch processing
    - iii) Hybrid
- Discrete manufacturing

Continuous process:

A continuous process is one in which raw materials enter one end of the system and the finished product comes out the other end of the system; the process itself runs continuously.

Batch processing:

In batch processing, there is no flow of product material from one section of the process to another. Instead, a set amount of each of the inputs to the process is received in a batch, and then some operation is performed on the batch to produce a product. Products produced using the batch process include food, beverages, pharmaceutical products, paint, and fertilizer.

Hybrid :

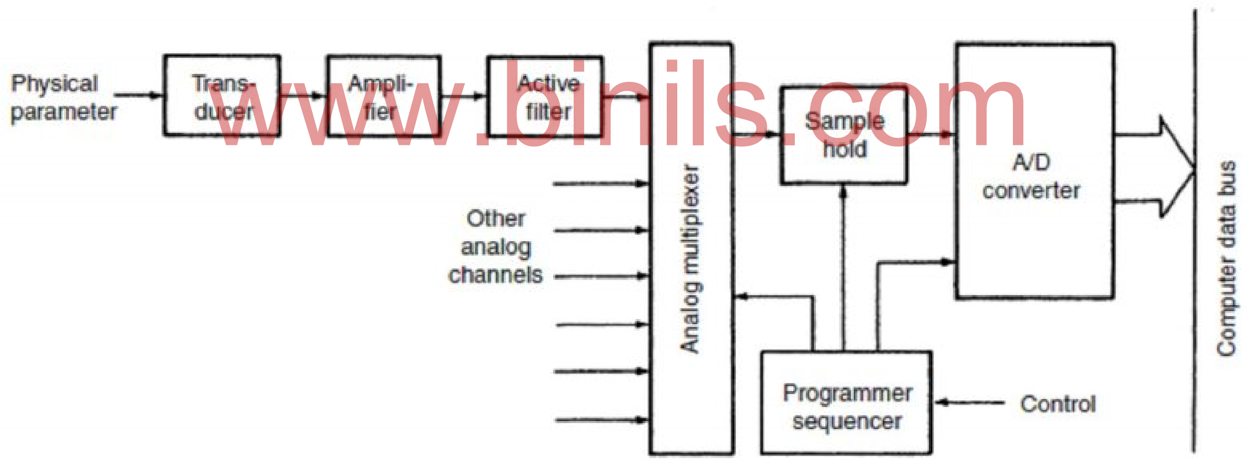
Applications having elements of batch and continuous process control are often called hybrid applications

Discrete manufacturing:

Discrete manufacturing is characterized by individual or separate unit production. With this manufacturing process, a series of operations produces a useful output product. Discrete manufacturing systems typically deal with digital inputs to PLCs that cause motors and robotic devices to be activated. The work piece is normally a discrete part that must be handled on an individual basis. Making car interiors is one example of discrete manufacturing.

5.3 Structure of control system:

Process control system can be defined as the functions and operations necessary to change a material either physically or chemically. Process control normally refers to the manufacturing or processing of products in industry. A block diagram representation of an industrial process control is shown in figure.



Components of process control system:

Transducer / Sensor:

i) Provide inputs from the process and from the external environment
ii) Convert physical information such as pressure, temperature, flow rate and position into electrical signals.
iii) It output can be used to monitor and control a process.

HMI:

i) Allows inputs from a human to set up the starting conditions or alter the control of a process.
ii) Allows human inputs through various types of switches, controls and keypads.
iii) Operates using supplied input information that may include emergency shut down or changing the speed, the type of process to be run, the number of pieces to be made or the recipe for a batch mixer.

Signal Conditioning:

i) Involves converting input and output signals to usable form.
ii) Includes signal condition techniques such as amplification, attenuation, filtering scaling A/D and D/A converters.

Actuators:

i) Convert system output electrical signals into physical action.
ii) Examples: Flow control valves, pumps, positing drives, VFD, clutches, brakes, solenoids, stepper motor and power relays.

Output devices

i. Indicate the state of the process variables through external actuators such as meters, CRT monitors, printers, alarms and pilot lights.
ii. It can send outputs directly from the controller to a computer for storage of data and analysis of results.

Controller:

I. Controller is the brain of the control system that takes decision to maintain the process variables at its desired value (set point).
II. It generates output signals which operate actuators to carry out the decision.

Multiplexer:

i. Multiplexer has large number of inputs and only one output.
ii. If the multiplexer is connected at input of ADC then it is called as analog multiplexing and if it is connected at output of ADC it is called as digital multiplexing.

5.4 ON/OFF control:

In an ON-OFF control action the output has only two states, fully ON or fully OFF. The time temperature response of an ON-OFF controller in a heating application is shown in the figure. In this control, the output turns ON when the temperature falls below the set point and turns OFF when the temperature reaches the set point.

Advantages: i) Simple control ii) Low cost

Disadvantages: i) Overshoot ii) Oscillation (Hunting) about set point

Remedy: Oscillation can be reduced by in amplitude buy increasing the sensitivity of the controller.

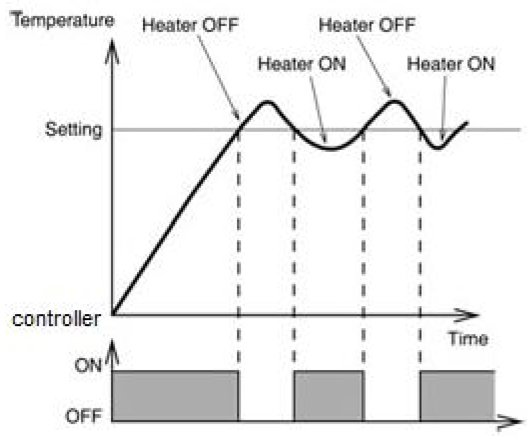


Fig: ON/OFF Control Waveform

5.5 Closed Loop Control:

The closed loop system is one of the control system in which the feedback taken from the output and then this feedback is feed is applied to the comparator for the comparing with the set point and the error is then applying to the further circuit. The operation and block diagram of the closed loop system is shown in the figure.

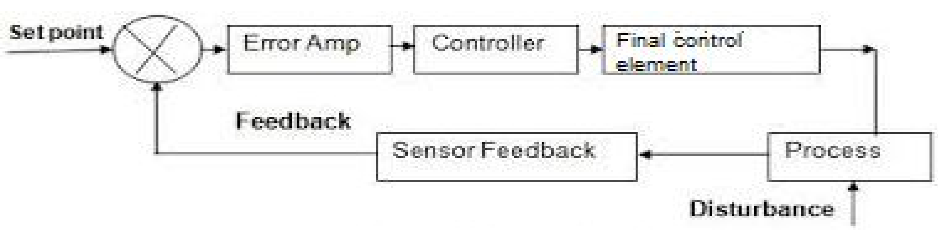


Fig: Closed Loop Block Diagram

Set point: It is the input that determines the desired operating point for the process.

Comparator: It is used to obtain the difference between actual output with set point. The set point signal is applied at the positive terminal and output signal is applied at negative terminal.

Error amplifier: It amplify the error signal and give to the controller because the error signal is very weak in amplitude, it may not directly operate the controller. So the weak signal is amplified by error amplifier.

Controller: It produces the output signal for the process based on the input error signal.

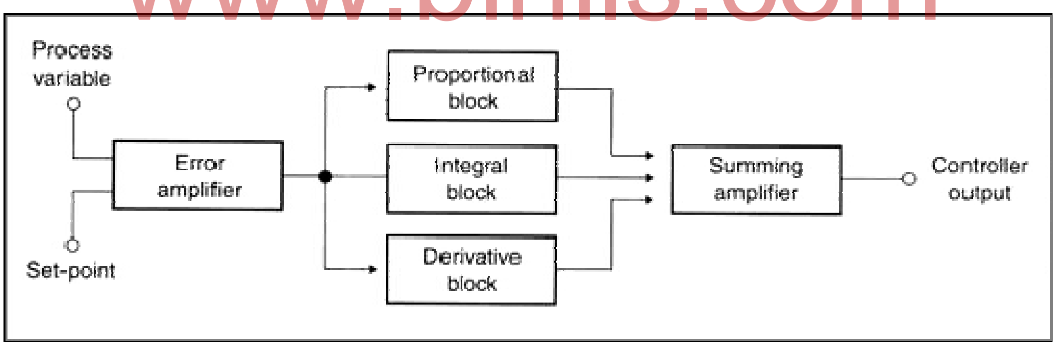
Final control element : It controls the output of the controller stage and input of the process state.

Sensor Feedback: It provides input from the process to the set point or comparator which compare the set point and feedback signal.

5.6 PID Control:

PID control is a feedback control method that combines proportional, integral, and derivative actions. The proportional controller reduces the offset. But offset will not become zero. The integral action automatically corrects offset. The derivative action responds quickly to large external disturbances. The PID controller is the most widely used type of process controller. When combined into a single control loop the proportional, integral and derivative modes complement each other to reduce the system error to zero faster than any other controller. Figure shows the block diagram of a PID control loop, the operation of which can be summarized as follows:

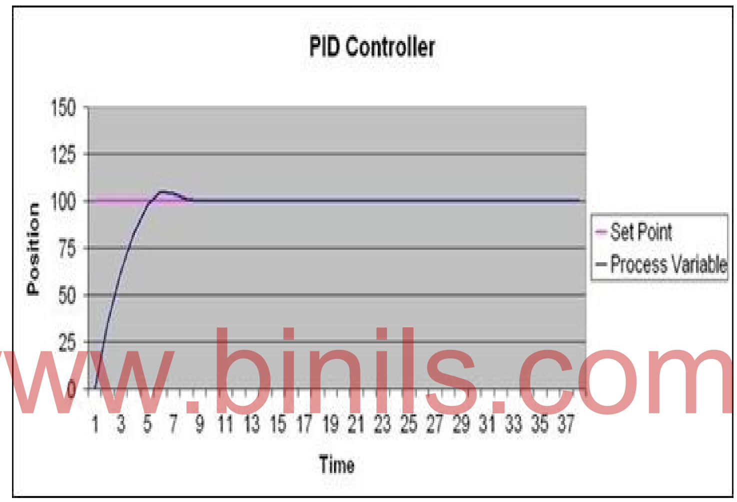
$$u(t) = \overbrace{K_p e(t)}^{\text{Proportional}} + \overbrace{K_i \int_0^t e(\tau)d\tau}^{\text{Integral}} + \overbrace{K_d \frac{d}{dt} e(t)}^{\text{Derivative}}$$



- ➢During setup, the set-point, proportional band, reset (integral), rate (derivative), and output limits are specified.
- ➢All these can be changed during operation to tune the process.
- ➢The integral term improves accuracy, and the derivative reduces overshoot for transient upsets.
- ➢The output can be used to control valve positions, temperature, flow metering equipment, and so on.

The response of a PID loop is the rate at which it compensates for error by adjusting the output. The PID loop is adjusted or tuned by changing the proportional gain, the integral gain, and/or the derivative gain. A PID loop is normally tested by making an abrupt change to the set point and observing the controller's response rate. Adjustments can then be made as follows:

- As the proportional gain is increased, the controller responds faster.
- If the proportional gain is too high, the controller may become unstable and oscillate.
- The integral gain acts as a stabilizer. Integral gain also provides power, even if the error is zero (e.g., even when an oven reaches its set point, it still needs power to stay hot). Without this base power, the controller will droop and hunt for the set-point.
- The derivative gain acts as an anticipator. Derivative gain is used to slow the controller down when change is too fast.



5.7 Motion Control:

A motion control system provides precise positioning, velocity, and torque control for a wide range of motion applications. PLCs are ideally suited for both linear and rotary motion control applications. Pick and Place machines are used in the consumer products industry for a wide variety of product transfer applications. The machine takes a product from one point to another. One example is the transfer of a product to a moving conveyor belt.

A basic PLC motion control system consists of a controller, a motion module, a servo drive, one or more motors with encoders, and the machinery being controlled. Each motor controlled in the system is referred to as an axis of motion. The controller stores and executes the user program that controls the process. This program includes motion instructions that control axis movements. A motion command represents the desired position, velocity, or torque of the servo motor at the particular time the calculations take place.

The motion module receives motion commands from the controller and transforms them into a compatible form the servo drive can understand. In addition it updates the controller with motor and drive information used to monitor drive and motor performance. The servo drive receives the signal provided by the motion module and translates this signal into motor drive commands.

The following diagram illustrates the essential components of a motion control system.

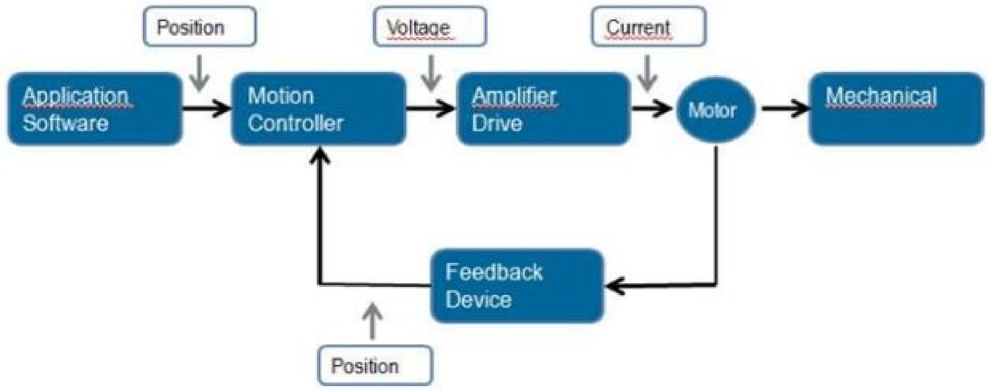


Figure The motion controller is the heart of the motion control system.

What's the Difference between a Motor and a Drive?

It is important to distinguish between motors and drives. A motor is the mechanical or electrical device that generates the rotational or linear force used to power a machine. A drive is the electronic device that harnesses and *controls* the electrical energy sent to the motor. The drive feeds electricity into the motor in varying amounts and at varying frequencies, thereby indirectly controlling the motor's speed and torque.

There are two types of drives: a standard *inverter drive* for controlling speed and torque only; and a *servo drive* for controlling speed and torque, as well as positioning machine components used in applications that require complex motion.

Together, a motor and drive form a "drive system."

5.8 Direct Digital Control (DDC):

A microprocessor based controller performing the monitoring, control loop processing and direct control of the mechanical system in response to system input. DDC is the automated control of a process by a digital device.

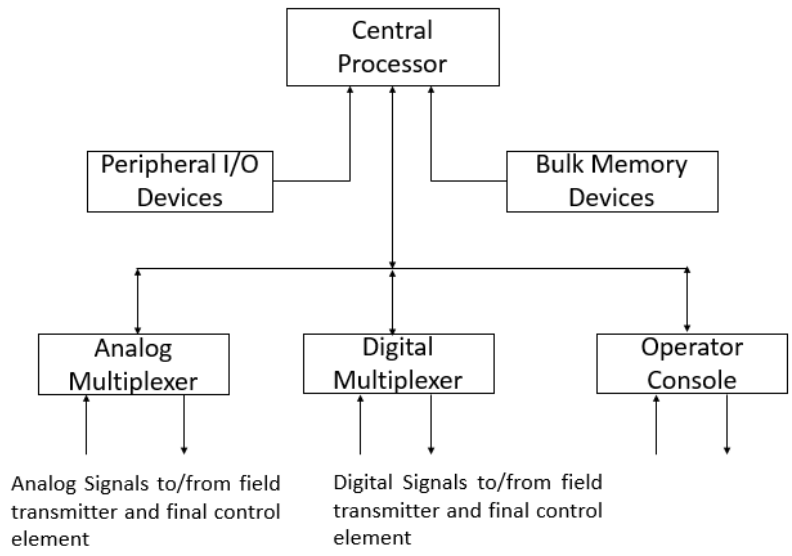


Fig: Block Diagram of DDC

DDC completely replaces the traditional pneumatic or electronic controllers. Direct Digital Control (DDC) is a type of energy management system with closed loop control. The term "Direct" means a microprocessor is directly in the control loop. The control is accomplished by "digital" electronics that read or control both digital and analog signals.

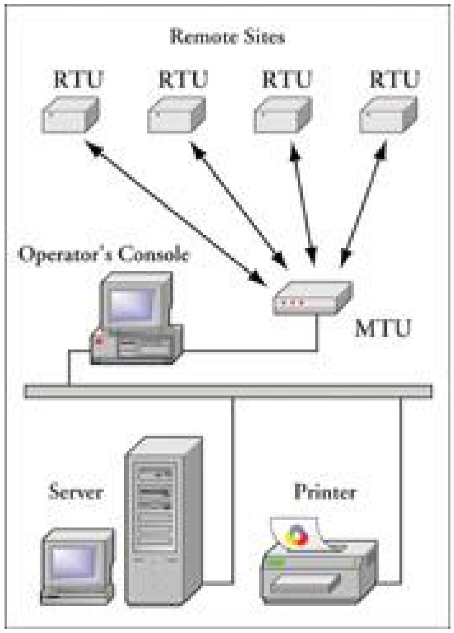
A basic DDC system is shown in figure. In this system a large number of transducer are connected and each transducer being connected to one input of a multiplier. Multiplexer has large number of inputs and only one output. Here the multiplexer is connected at input of ADC then it is called as analog multiplexing. The data from output of ADC is transmitted to CPU through data bus.

When CPU has analyzed the data from one or more transducer and has compared them with the appropriate set points in the computer program, it sends signals along the data bus to the values controlling the system as follows. The digital signal produced by the CPU is converted into an analog signal by a DAC and the analog signal is transmitted to the appropriate control valve through a demultiplexer. As this is being performed, data are displayed on the operator visual display unit and if necessary, he can remotely change the set points associated with various sections of the process.

5.9 Introduction about SCADA:

SCADA refers to the combination of telemetry and data acquisition system. SCADA encompasses the collecting of the information, transferring it back to the central site, carrying out any necessary analysis and control and then displaying that information on a number of operator screens or displays. The required control actions are then conveyed back to the process.

5.9.1 Block diagram of SCADA:



SCADA systems typically have four major elements:

1. Master Terminal Unit (MTU)
2. Remote Terminal Unit (RTU)
3. Communication Equipment
4. SCADA Software

1. Master Terminal Unit (MTU):

The Master Terminal Unit is usually defined as the master or heart of a SCADA system. It is located at the operator's central control facility. The MTU initiates virtually all communication with remote sites and interfaces with an operator. Data from remote field devices (pumps, valves, alarms, sensors etc.) is sent to the MTU to be processed, stored and/or sent to other systems.

2. Remote Terminal Unit (RTU):

The Remote Terminal Unit is usually defined as a communication satellite within the SCADA system and is located at the remote site. The RTU gathers data from field devices (pumps, valves, alarms, etc.) in memory until the MTU initiates a send command. Some RTUs are designed with microcomputers and programmable logic controllers (PLCs) that can perform functions at the remote site without any direction from the MTU.

3. Communication Equipment:

In the SCADA system network but there must be uninterrupted, bidirectional communication between the MTU and the RTU for a Data Acquisition system to function properly. Private wire lines, buried cable, telephone, telemetry hardware like wireless radios and modems, microwave dishes, satellites, or other atmospheric means, DSL (Digital Subscriber Line), Integrated Service Digital Network (ISDN), cable, fiber optics, WiFi, or other broadband services.

4. SCADA Software:

A typical SCADA system provides a Human Machine Interface (HMI) allowing the operator to visualize all the functions as the system is operating. The operator can also use the HMI to change set points, view critical condition alerts and warnings, and analyze or present data trends. Common HMI software packages include Cimplicity (GE-Fanuc), RSView (Rockwell Automation)

5.10 Features of SCADA:

- User interface
- Graphics displays
- Alarms
- Trends
- RTU (and PLC) interface
- Scalability
- Access to data
- Database
- Networking
- Fault tolerance and redundancy
- Client/server distributed processing

5.11 Functions of SCADA:

1. Data Collection (Data acquisition)
2. Data Transmission
3. Monitoring
4. Supervision and Alarm
5. Control and indication
6. Data logging
7. Data Display

5.12 SCADA Software:

SCADA software are two types: 1. Run time 2. Works

SCADA works software contains both application development software and run time software. Using this software user can develop application program and also run this application. SCADA runtime contains only runtime program. Using this software user can run the application program, but cannot develop application program.

Various SCADA Software available in the market:

#	Developer	Name of the Software
1	Siemens	WinCC
2	AllenBradly	RSView 32
3	GeFanuc	Cimplicity
4	Wonderware	InTouch
5	General Electric	Intellution iFIX
6	Honeywell	SCAN 3000 SCADA

5.13 Data Loggers:

Data loggers are electronic devices which automatically monitor and record environmental parameters over time, allowing conditions to be measured, documented, analysed and validated. The data logger contains a sensor to receive the information and a computer chip to store it. Then the information stored in the data logger is transferred to a computer for analysis. Data logging facility is also available with SCADA software.

5.14 Tags:

A Tag is a logical name for a variable in a device or in logical memory. Tags that receive data from an external source such as a PLC is referred as device tags. Tags that receive their data internally from SCADA software are referred to as memory tags. Tags are stored in the tag database and their names are then used in other parts.

Tag Types:

Sl.No	Types of Tag	Description
1	Digital Tag	It stores 0 or 1. These tags can represent device that can only be ON or OFF.
2	Analog Tag	It stores range of values. These tags represent variable states such as pressure, temperature, level, flow etc.,.
3	String Tag	It stores ASCII string, series of characters or whole words. These tags can represent devices that use text, such as a bar code scanner, which uses alphanumeric product code.
4	System Tag	It stores information generated while the system is running, including alarm information and the system time and date. We cannot edit or write to system tags but we can use them.

5.15 Alarm system:

Industrial processes are often supervised by an SCADA. It monitors analog and digital signals from the plant and checks on limits for the values. There is need to have the record of alarm and alarms acknowledgement information. An alarm occurs when something goes wrong. It can signal that a device or process has ceased operating within acceptable, predefined limits or can indicate breakdown, wear or a process malfunction. An alarm should only be activated when the process limit violated. Set up a systems of alarms in the tag database editor by linking alarms to value table, they are compared to the limits you assigned, when you configured the alarm. If a tag value exceeds the configured limits, an alarm of a present severity is triggered.

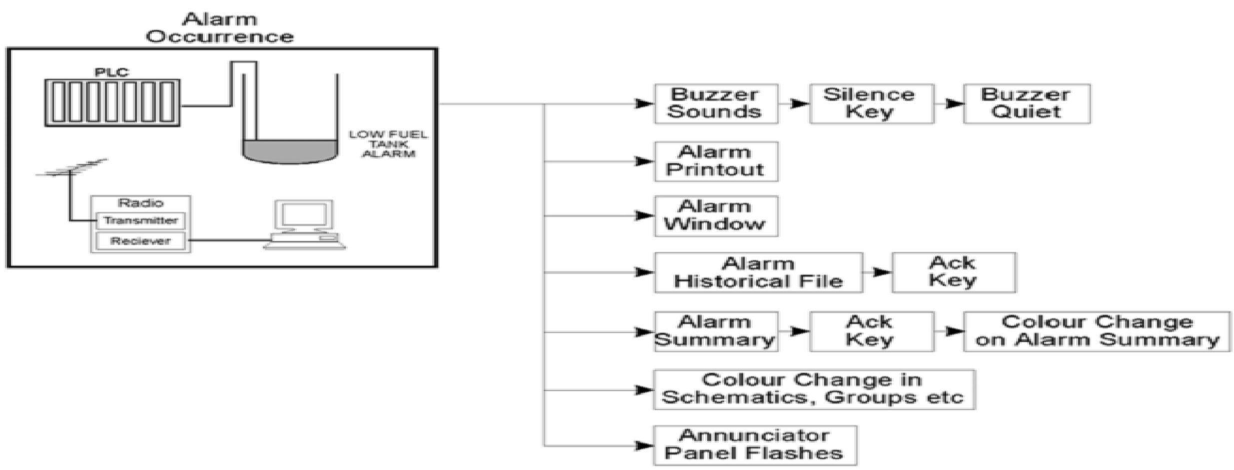


Fig: Alarm actions in an operator display

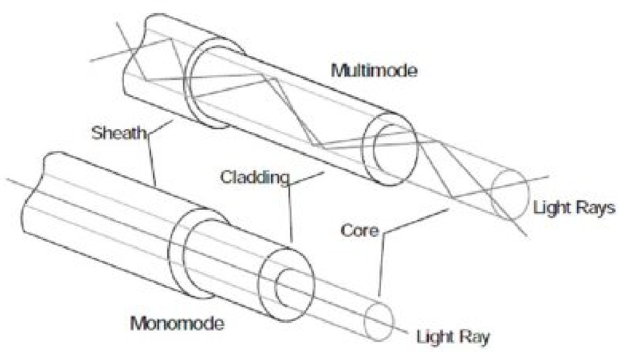
Requirements of good alarm system:

- Client server architecture
- Unmistakable
- Alarms are shared to all clients
- Alarms displayed in chronological order
- User-defined formats and colors
- Context-sensitive help
- On-line alarm disable and threshold modification

5.16 Landlines for SCADA:

Even with the reduced amount of wire when using a PC to IED system, there is usually a lot of wire in the typical SCADA system. This wire brings its own problems, with the main problem being electrical noise and interference. Interference and noise are important factors to consider when designing and installing a data communication system, with particular considerations required to avoid electrical interference. Noise can be defined as the random generated undesired signal that corrupts (or interferes with) the original (or desired) signal. This noise can get into the cable or wire in many ways. It is up to the designer to develop a system that will have a minimum of noise from the beginning. Because SCADA systems typically use small voltage they are inherently susceptible to noise. The use of twisted pair shielded cat5 wire is a requirement on most systems. Using good wire coupled with correct installation techniques ensures the system will be as noise free as possible.

Fiber optic cable is gaining popularity because of its noise immunity. At the moment most installations use glass fibers, but in some industrial areas plastic fibers are increasingly used.



Optical fiber cable

An optical fiber cable, also known as fiber optic cable, is an assembly similar to an electrical cable, but containing one or more optical fibers that are used to carry light. The optical fiber elements are typically individually coated with plastic layers and contained in a protective tube suitable for the environment where the cable will be deployed. Different types of cable are used for different applications, for example long distance telecommunication, or providing a high-speed data connection between different parts of a building.

Reliability and quality

Optical fibers are very strong, but the strength is drastically reduced by unavoidable microscopic surface flaws inherent in the manufacturing process. The initial fiber strength, as well as its change with time, must be considered relative to the stress imposed on the fiber during handling, cabling, and installation for a given set of environmental conditions.

There are three basic scenarios that can lead to strength degradation and failure by inducing flaw growth: dynamic fatigue, static fatigues, and zero-stress aging.

5.17 Modem use in SCADA systems:

Often in SCADA systems the RTU (remote terminal unit (PLC, DCS or IED)) is located at a remote location. This distance can vary from tens of meters to thousands of kilometers. One of the most cost-effective ways of communicating with the RTU over long distances can be by dialup telephone connection. With this system the devices needed are a PC, two dialup modems and the RTU (assuming that the RTU has a built in COM port). The modems are put in the auto-answer mode and the RTU can dial into the PC or the PC can dial the RTU. The software to do this is readily available from RTU manufacturers. The modems can be bought off the shelf at the local computer store.

Line modems are used to connect RTUs to a network over a pair of wires. These systems are usually fairly short (up to 1 kilometer) and use FSK (frequency shift keying)to communicate. Line modems are used to communicate to RTUs when RS-232 or RS-485 communication systems are not practical. The bit rates used in this type of system are usually slow, 1200 to 9600 bps.

Unit V

Model Questions

Part A

1. What is process control?
2. What are the types of process control?
3. What are the functions of SCADA?
4. Mention the types of SCADA software.
5. Mention SCADA softwares available in market.
6. What do you mean by data logger?
7. What is tags?
8. Expand the term SCADA.

Part B

1. Explain about any two types of process control.
2. Explain about closed loop control.
3. What are the features of SCADA.
4. Explain about the types of Tag.
5. What are the requirement of good alarm system.
6. Give example of any three SCADA software
7. Explain ON/OFF control with suitable example.
8. What are the benefits of SCADA?
9. What is data logger? State its functions

Part C

1. Draw the structure of control system and explain about the components of process control system.
2. Explain the PID control with neat sketch.
3. Explain with block diagram of DDC.
4. Explain the use of landlines and modems in SCADA
5. With neat block diagram explain the operation of SCADA.
6. With neat block diagram briefly explain about proportional integral process
7. Describe the hardware and software involved in SCADA system