

Module 1

What is a PLC?

A Programmable Logic Controller (PLC) is a solid/ state/ computerised industrial controller that performs discrete or sequential logic in a factory environment. It was originally developed to replace mechanical relay, timers and counters.

PLCs are used successfully to execute complicated control operations in a plant. A sequence of instructions is programmed by the user to the PLC memory and when the program is executed, the controller operates a system to the correct operation specifications

The first commercial PLC was introduced in the late sixties, when microprocessors were not available. In the late 1970s, with the advent of microprocessors, the role of PLCs was greatly enhanced. Capabilities such as intelligent operator interface, arithmetic data manipulation and computer communication added new dimension to the PLC application. Hardware enhancement included large memory capacity analog input/output and high-speed data communication between programmable logic controllers. PLC technology is being continuously upgraded, even today

Today PLC is used as standalone control or as a part of complex Computer Integrated Manufacturing system. A PLC may also provide an interface between a machine and robots or a machine and color graphics which are known as Human Machine Interface (HMI). A PLC can now be found in every type of industry imaginable

A PLC is as important to the world of industrial automation as the computer is to the business world

Technical Definition of PLC

A programmable logic controller is currently defined by the National Electrical Manufacturers Association (NEMA) as a “digital electronic device that uses a programmable memory to store instructions and to implement specific functions such as logic, sequence, timing, counting and arithmetic operations to control machine and processes”

What are its Advantages?

PLCs have gained popularity in industry as they offer specific advantages, which are given below:

- Faster scan time
- Intelligent I/O
- High Speed counters
- Supervisory control capability
- Reliability in operation
- Flexibility in programming and reprogramming in the plant
- Cost effective for controlling complex systems
- Flexibility in control techniques
- Minimum maintenance
- Small physical size
- Superior computational capabilities to execute complex control technique
- ASCII message handling capability
- Ability to communicate with computer systems in the plant
- Programmable troubleshooting aids which reduce downtime

Characteristic Functions of a PLC

Characteristic Functions of a PLC

A PLC offers certain specific characteristic functions that make it advantageous in an industrial environment. It is superior in modern industrial control and communication. Some of the most characteristic functions of a PLC are as follows:

(i) It is field programmable by the user.

It allows the user to modify the program as per the requirements of a plant without having to send the unit back to manufacturer for this purpose.

(ii) It contains programmed functions.

PLCs can contain at least logic, timing, counting and memory functions that the user can access through some type of control oriented programming language.

(iii) It scans memory and inputs and outputs (I/O) in a predetermined manner.

The critical feature of a PLC is that it allows the control engineer to determine precisely how the machine or process will respond to the program.

(iv) It provides error checking and diagnostics.

A PLC will periodically carry out internal tests of its memory, processor and I/O systems through in-built program execution to ensure that what it is doing to the machine or system, against what it was programmed to do

(v) A PLC can provide some form of monitoring capabilities

A PLC can provide some form of monitoring capabilities either through indicating lamps that show the status of inputs and outputs, or by an external device that can display the program execution status.

(vi) Packaging

It is packaged in such a way that it can withstand the temperature, humidity, vibration and noise found in most factory environments

(vii) Design

A PLC can be effectively defined for a wide variety of control tasks, although it can also be dedicated only for a particular task.

Types of PLC

Unitary PLC

Unitary PLC has a power supply, a CPU and a limited number of inputs and outputs (20 inputs -12 outputs, 32 I/O). It is sometimes called “shoebox type” and mainly used for the control of a small system.

Modular PLC

A modular PLC is that which can be constructed using separate modules of power supply, CPU, inputs, outputs, timers counters, ADC,DAC expansion modules. These modular PLCs are sometimes called rack mounted type PLC. Modular PLC can be sub-divided into following type:

- **Small PLC**
PLCs having less than 100 inputs and outputs are designated as small PLCs. Out of the I/Os, 20 input and 12 outputs are mounted locally with the processor. Additional I/Os can be added through remote I/O racks to accommodate the extra inputs and outputs. These PLCs generally have a memory from 2KB to 10KB to store the user’s logic programs
- **Medium PLC**
These have extended instruction sets that include mathematical functions, file functions, PID process control etc. These PLCs can have between 4000 to 8000 inputs and outputs. They are also made to support wide variety of special module such as ASCII communication modules, BASIC programming modules, 16-bit multiplexing modules, analog input and output modules and communication modules.
- **Large PLC**
The purpose of introducing large PLC was to provide enough user memory space and I/O for a complete factory automation. However the major disadvantage of these large PLCs is that the whole factory may have collapsed if that PLC starts malfunctioning. The advent of Local Area Networking helped to introduce the concept of distributed control, where small or medium PLCs are connected together the appropriate network. The entire factory is brought under the control of a number of PLCs, but failure in one system will not disturb any other system

Block Diagram of PLC

A simplified block diagram of a PLC is shown in fig. It is seen that a PLC has three basic section (i) I/O section (ii) Processor Section and (iii) program loader/ monitor section.

The input section converters convert the field signals to logic-level signals that the PLC’s processor can read directly

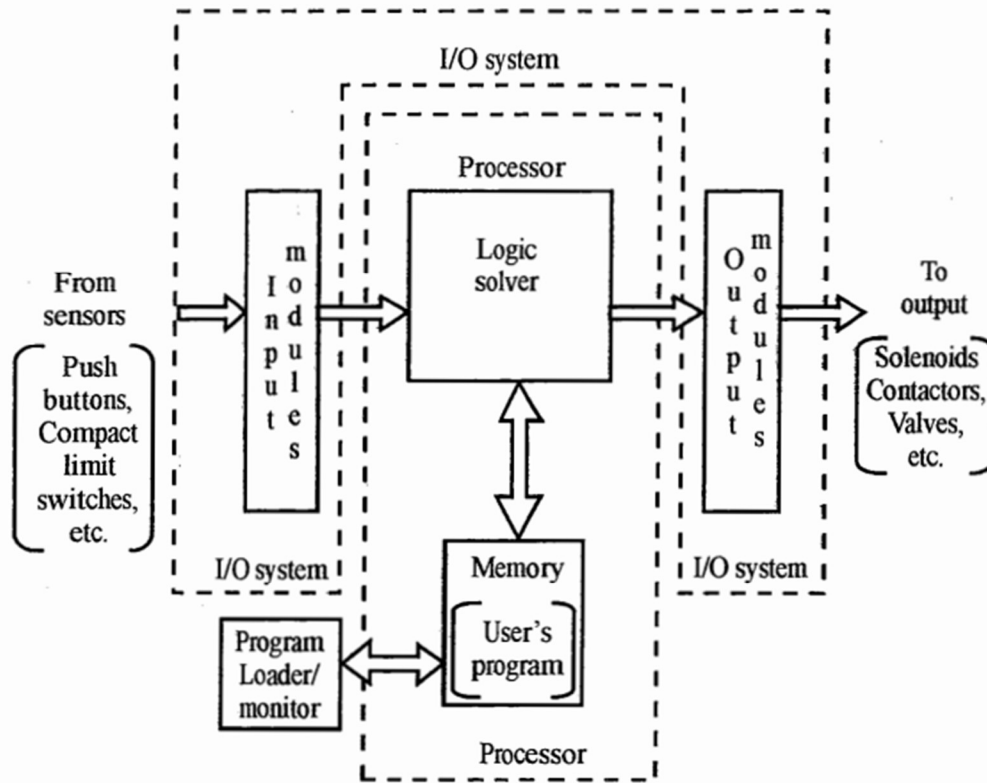


Fig.1.1 :Block Diagram of a PLC

The Processor section reads these input and processes the signals executing the program and prepares the output signals.

The output section converts the logic level output signals coming from the processor section into high level signals and releases signals for various field devices

The program loader is used to enter the user's program into the memory or change it and to monitor the execution of the program.

Input/Output (I/O) Section

The input output sections establish the interfacing between the real world outside the PLC and the digital arena inside the PLC. The signals from machines or instruments of a system are received through the I/O section by a PLC whereas after processing, the output signals are transmitted to the machines or the instruments for the desired control action through the output section.

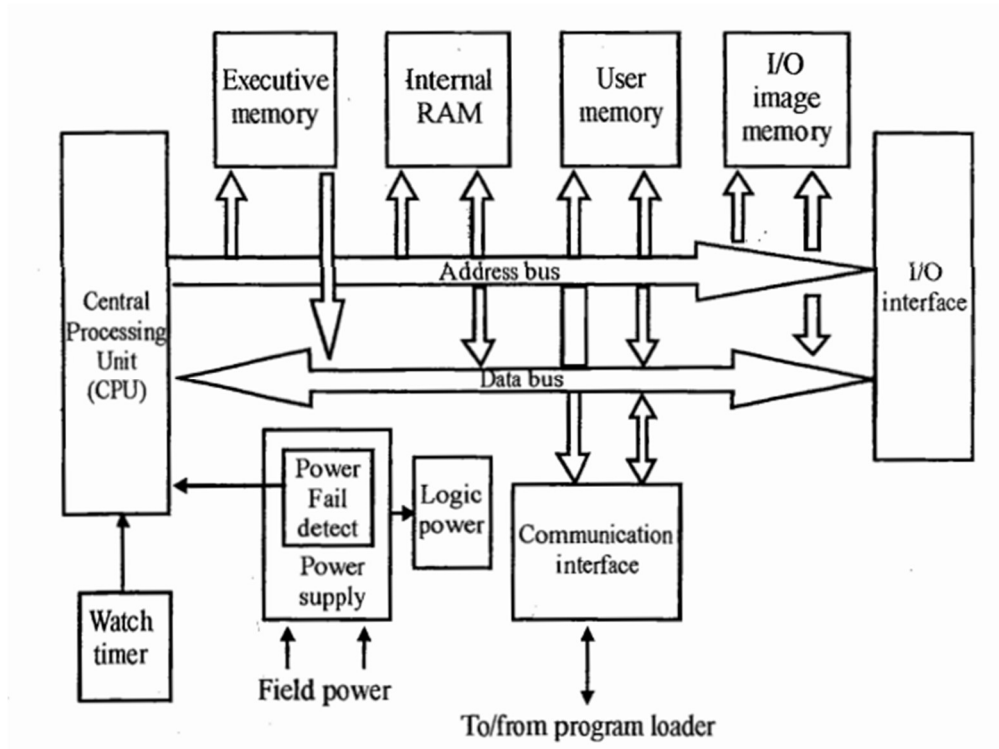
The tasks of the I/O section can be classified as: (a) conditioning (b) isolation (c) termination and (d) indication

The field signal available is 200V or 440V. Since a PLC can only support a voltage of 0-10 volts, the field signal has to be conditioned before it is applied to the PLC.

To protect a PLC from any incoming surges from the field, isolating devices such as transformer or opto-isolator are used. The I/O module provides a proper terminal. It indicates the status of the I/O terminal, i.e., when the voltage is off, the status indicator is off.

Processor Section

A block diagram of the processor section of a PLC is shown in fig. This section consists of four major elements (i) Power supply (ii) memory (iii) CPU and (iv) I/O interface



Power supply

The basic function of the power supply is to supply a conditioned power to the components of a PLC. The power supply is one of the most critical component of the PLC for two reasons:

- It is typically non-redundant. Hence, failure of the PLC power supply can cause the entire control system to fail
- It contains high voltage components. Hence, an insulation failure can be a serious potential for injury or fire

The following guidelines should be followed when choosing the power supply of a PLC:

- The power supply should be packed properly, so that the heat generated by the power supply can be dissipated in order to prevent overheating. This increases reliability
- The power supply should be tested for temperature and electrical insulation by a proper certification agency
- The power supply should meet at least one reputable standard for noise immunity. It should also be capable of withstanding line voltage variations such as chop outs, burn outs and surges which are common to industrial facilities
- Suitable redundancy will increase reliability

Memory

There are two types of memory in a PLC (i) Volatile memory and (ii) non-volatile memory

Non-volatile memories are generally used for storing users programs so that they are preserved during power cuts. It is important that all non-volatile memory in a PLC uses some form of error checking to ensure that the memory has not changed. In order to ensure safe execution of the users programs, this error checking should also be done while the PLC is controlling a machine or a process.

Different types of memory that are generally used in PLC are as follows

(a) Battery backed up CMOS RAM

This is probably most widely used type of memory. Although most RAMs are inherently volatile, the CMOS variety consumes such a small amount of power that a small battery will retain the memory during power losses. The batteries used vary from short-life primary cells (alkaline and Hg) which require periodic replacement, after 6 months to a year, to long life cells (such as lithium) which may last up to 10 years, to rechargeable secondary cells

(b) EPROM Memory

An electrically programmable read only memory is programmed using electrical pulses and can only be erased by exposing the circuit to UV light also called UV- EPROM

(c) EEPROM Memory

This memory is similar to EPROM memory, but it can also be erased using electrical pulses.

Central Processing Unit

The flexibility of a PLC is dependent on (i) how the CPU is constructed i.e whether or not the PLC can be expanded and modified for future enhancement. And (ii) the overall speed of the PLC. The speed is expressed in terms of how fast the PLC will scan a given amount of memory. This measure, called the scan rate, is expressed in milliseconds per thousand words of memory. The scan time of a PLC determines its appropriate field of application.

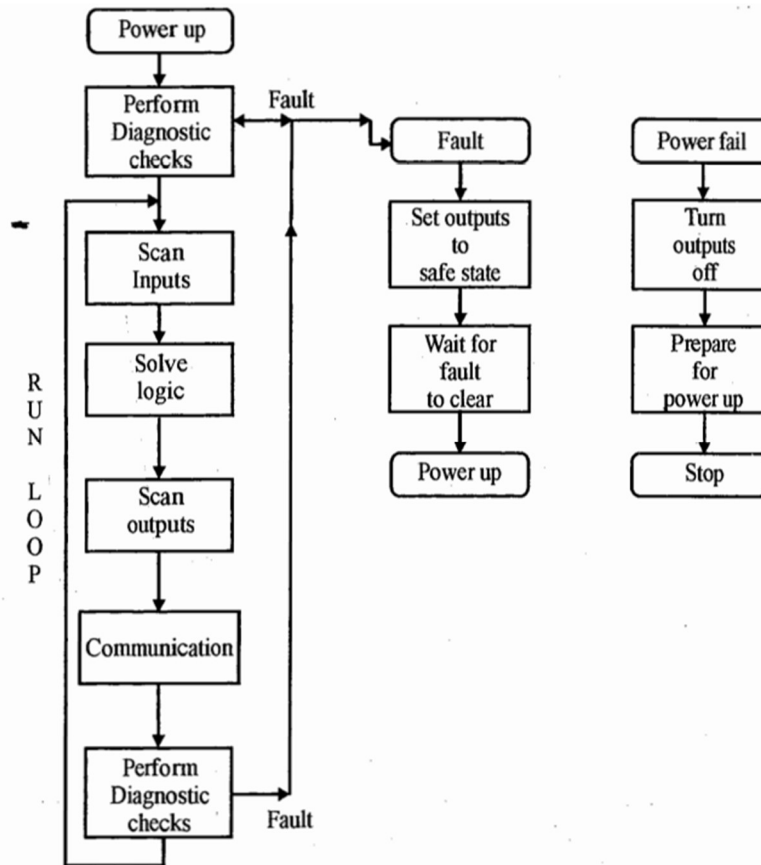
A real program that uses other functions such as timers, counters and other mathematical functions, may take considerably larger time to execute. Also, in procuring a PLC, one should include the scan time of the I/O, the scan time of the memory and any additional time overheads the processor requires for making a predictions of the overall scan time for a given application.

Processor Software/ Executive Software

The Hardware of a PLC does not differ significantly from that of a computer. What makes the PLC special is its software, which called executive software.

The executive software is the program that the PLC manufacturer provides internal to the PLC, which executes the user's program. This executive software determines what functions are available to the user's program, how the program is solved, how the I/O is serviced and what the PLC does during power up or power down and fault conditions

What the executive program does is shown in the form of a flow chart shown in fig. In fact it can be said that executive software controls the whole functionality of PLC



The following points can be noted from the flow chart:

- (1) The diagnostic checks must be run during power up condition as well as during the run loop, which is executed during the controlling of the process by PLC
- (2) When faults are detected, the outputs must be set to a predetermined safe state

A specific PLC may perform the basic functions shown somewhat differently, e.g. some PLC's may perform diagnostics only at a single point in the executive program, while others may perform diagnostics on line.

Multi-Tasking

Some PLCs are capable of executing multiple tasks with a single processor. In particular, PLCs that support sequence function charts (SFCs) tend to support limited multi-tasking capability. Multi-tasking takes several forms, two of which are time driver and event driver.

In some time driver multi-tasking system the user writes programs and assigns a separate I/O for each task. The user may then be allowed to configure the processor to run each task at periodic time intervals.

In event driver multitasking (also called interrupt drivers), the user defines a particular event that causes each task to run. Some multitasking systems allow any task to access variable, such as an I/O point

Languages

Variables – These are the way the language allows the user to access the I/O internal data

On line vs off line – An on line language is one that can be programmed interactively with the PLC. An off line program requires that the program be generated separately and then downloaded to the PLC.

Flow control - This feature dictates how easy it is to take decisions and to control the program's execution.

Functions – Some languages only provide the minimum set of logic, timing, counting and memory functions while others may provide additional instructions, such as down controllers, motion operation, shift registers, mathematical functions and many others.

Maintainability – This refers to how easy the language is to debug, modify and teach to others

Speed – Some languages take a longer time to execute than others. Choosing a slow but powerful language in time critical applications can make the programming much more difficult than using a fast but simple language

Efficiency – This is a measure of how much memory is required for a language to implement a given function

Ladder Language

This is still the main language of the PLC. It has many advantages, as given below:

Ladder logic program is exactly similar to electrical Ladder diagram. It is readily understood and maintained by skilled workers familiar with relay logic. It simplifies training ladder language, however, it generally lacks good flow control instructions.

- It provides a graphical display of program execution by showing power flow through a ladder diagram, thereby making it easier to debug.
- The program is fast
- It generates more readable programs for sequence control.

Modern PLCs are very powerful and have vast manuals which describe all the features. PLC languages are of the graphical programming type. One can drag and drop any functions blocks to develop a complete program line.

All the blocks are context sensitive, for example clicking on a timer and clicking on help will bring up a window describing the timer instruction and how it is used. So program development environment is user friendly.

PLC vs PC

From the discussions so far, a Programmable Logic Controller (PLC) appears to resemble a Personal Computer (PC). However there are certain difference.

Personal Computer are data processing computers. Such computers process and analyse a large number of data, store the database, generate report and display the data and information to operators.

There is also another type of computer, known as process control computer or industrial computer. The main function of this type of computer manufacturing and industrial process, which are even driven.

PLCs are a type of process control computer, small, relatively inexpensive, environmentally hardened and easy to program, maintain and repair. They can handle real time data and direct I/O tasks whereas computers can handle non-real time tasks. PLCs are often installed close to a machine or process they control, and are often considered as an extension of a piece of industrial equipment