

Programmable Logic Controllers (PLC) and its Programming

Mukul Pathade and Gaurav Yeole

Sardar Patel College of Engineering, Andheri (W), Mumbai- 400 058.

Abstract: - *This paper describes the important concepts of PLC and its applications. The main difference between PC and PLC is that latter is capable of functioning in severe conditions (such as dust, moisture, heat, cold, etc.) and have the facility for input/output (I/O) arrangements. PLC is highly customized system so the cost of package PLC is low as compared to other control systems. A PLC is an example of a hard real time system since output results must be produced in response to input conditions within a bounded time, otherwise unintended operation will result. In industrial sector PLCs offers several advantages over other control systems, hence it is widely used.*

Keywords: - *Automation, PLC, Ladder logic.*

I. INTRODUCTION

Automation means using automatic methods in factories, offices, farms, labs, houses, space vehicles, atomic power station, shops, schools anywhere they are possible. It means giving machines the power to control work in their places, to do jobs without the help of man's skills and experience. 'Automaton' is the word and idea from which automation sprang, is a mechanical man, robot. Automatic processes are not new they have been developing from centuries. A new word, 'Automation' was introduced to cover them. These machines have served the purpose to work for long hours and relieve the pain of mind as well as body [5].

Programmable Logic Controller (PLC) is a digital computer used for automation of electromechanical processes. The first PLC was manufactured by Bedford Associates and it was designated as 084 in seventies [3]. One of the people who worked on that project was *Dick Morley* and he is considered to be the "Father" of PLC [1]. Unlike

personal computers (PC) PLCs are designed such that they can withstand different physical conditions such as temperatures, electrical noises according to requirements in industry. The development of PLC has brought a revolution in seventies [2].

II. WHAT IS PLC?

A PLC (Programmable Logic Controller) is a small computer with a built in operating system. A PLC has input lines where sensors are placed to notify upon incoming event (e. g., temperature level, liquid level) and output lines to signal any reaction to incoming events. These are real time devices. It was designed to replace relay logic systems as the technicians needed to rewire the relays to change the logic. But with PLC, logic can be changed by making changes in program. PLC has a programmable memory to store and implement instructions which PLC receives as an input from field which can be either current or voltage depending on the type of sensors.

There are two types of PLC viz., Modular PLC and Compact PLC. Compact PLC has inbuilt I/O (Input-Output) modules, while modular has separate I/O modules. Fig.1 shows Siemens S7-300 modular PLC.



Fig. 1- Siemens S7-300 PLC.

1- Power Supply; **2-** CPU; **3,4 –** Digital I/O; **5,6 –** Analog I/O.

Fig. 2 illustrates the basic working of PLC. Irrespective of the manufacturer of the PLC this block diagram remains the same.

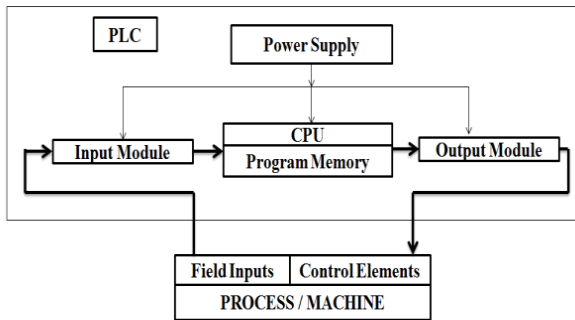


Fig 2- Block Diagram of PLC

III. Why PLC?

Originally PLCs were designed to replace relay control logic. The cost savings using PLCs have been so significant that relay control is becoming rare, except for power applications. PLC eliminates much of the hard wiring that was associated with conventional relay control circuits. Once the program using software has been written on PG and tested it can be downloaded to other PLCs too. PLCs have resident diagnostic and override functions allowing user to trace and correct software and hardware problems. When running a PLC program a visual operation can be seen on the screen. Hence troubleshooting a circuit is really quick, easy and simple. One single PLC can easily run many machines.

IV. COMPONENTS OF PLC

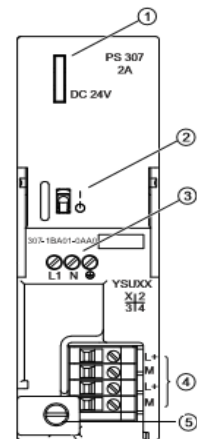
A basic PLC has following components:

- Power Supply Module
- CPU
- Signal Modules (Digital I/O and Analog I/O)
- Interface Module
- Function Module
- Communication Module
- Memory Unit

4.1) Power Supply Module: It converts the supply voltage (AC 120/230 V, DC 24 to 110V) to the operating voltage 24V. This converted voltage is further supplied to all various modules connected to

PLC. Fig.3 shows the front panel of 2A power supply module. Power supply modules are available for different current ratings ranging from 2A to 10A [6].

- ① "24 VDC output voltage present" display
- ② 24 VDC On/Off switch
- ③ Mains and protective conductor terminals
- ④ Terminals for 24 VDC output voltage
- ⑤ Strain-relief



4.2) Central Processing Unit (CPU) of PLC: Fig 3- Power supply module

It is most important module of the PLC. CPU consists of microprocessor and memory. Its processor talks to the periphery (i.e., DI, DQ, AI, AQ) and function modules via P-bus. The communication to CPU to outside devices is through K-bus. The P-bus and K-bus are handled by separate processor inside CPU. There are various ports on CPU such as RS232, Ethernet port for the communication purpose. There are different registers inside CPU for data handling, address handling and for control functions. In Siemens S7-300 PLC there are two 32 bits registers are available for performing logical and arithmetic operations. They are known as accumulators. Number of accumulator varies with CPU.

CPU memory is mainly divided into work memory, load memory and system memory. Load memory is used to store the program. It may be RAM, ROM or EPROM depending on the programmable controller. Programmer may use a memory card (RAM or EEPROM) to extend the load memory. The work memory which is an integrated RAM used to store the parts of user program required for program processing.

And the third is system memory which consists of PII (Process Image Input), PIQ (Process Image Output), bit memories (M), Timers/Counters (T)/(C) and Local stack (L) areas. These memory areas are updated cyclically by the operating system. At the beginning of the cyclic scan signal states are

transferred from input modules to PII region; at the end of the each program scan signal states are transferred from the PIQ region to output modules. The size of each timer and counter is 16 bit (word). They can be accessed by user program by means of instruction. Their number depends on CPU. Bit memory is used to store the intermediate results. Bit memory is accessible everywhere in program. Unlike bit memory system memory also has local stack which is valid for particular part of the program only.

To ensure the fast execution of the user program and to avoid the unnecessary load on the work memory, only parts of the program relevant for program execution are loaded in the work memory. The parts of the block which are not required for the executing the program are retained in the load memory.

The front panel of CPU has status and line indicator LEDs, Slot for MMC, Operating Mode switch, MAC Address and 2D barcode, connection for voltage supply and various interface ports for MPI(multi point interface), PPI(point to point interface), PROFINet Industrial Ethernet etc. These interfaces are used to connect the CPU with the Programming device (PG) and other PLC CPUs if required in large plants. Status LEDs may flash yellow, red, green and orange to indicate various states of the PLC. Mode switch is a mechanical switch to put PLC in STOP, RUN and memory reset mode (MRES). MRES is often used by programmer to clear the work memory area. When one resets the CPU memory, programs stored in RAM are lost while programs in EPROM are retained.

4.3) Signal Modules: These are status indicators of the incoming and outgoing signals of PLC [6]. The status of the field input is stored in this module. These modules may be digital or analog depending on the peripherals. The communication between CPU and these modules is done via P-bus.

4.4) Memory Unit: It is mainly used to expand the load memory. It may be a RAM or an EPROM memory card. EPROM memory cards are mostly preferred as these are non-volatile. The supportable size of MMC depends on the CPU. The maximum available size of MMC is 8 MB.

4.5) Rack: All above mentioned modules along with interface module are placed on the rectangular support known as Rack. Number of modules on a particular rack are fixed and decided by the manufacturer.

4.6) Interface Module: This module is used for interfacing the racks. It allows number of PLCs and other intelligent device to communicate and pass PLC data over high speed local area communication network. Any device may interface with the network, because the network is not restricted to the product designed by the network's manufacturer. The receiving network interface accepts the transmission, passes the information to the CPU and if necessary sends a command to the intended field device.

V. PROGRAMMING ON PLC

As the name suggests PLC is a programmable device, programming is mainly done on Programming device (PG). Different softwares are available for programming depending on the manufacturer of PLC (e. g., Software SIMATIC Manager STEP 7 is used to program the Siemens PLC).

Generally in SIMATIC Manager programming can be done in following three languages [7]:

1. Ladder (LAD)
2. Statement List (STL)
3. Function Block Diagram (FBD)

Fig. 4 shows the Graphical User Interface (GUI) of the SIMATIC Manager STEP 7. From here one can access all the functions installed for standard packages and any optional packages. Programmer can work with this software in one of two ways:

1. Offline, without PLC connection
2. Online, with the PLC connection



Fig 4 - SIMATIC Manager GUI

The left half of the fig. 4 shows the directory structure of the folders and files and it is an example of object hierarchy. The above directory has a project name CPU314C_2DP which can be changed. Programmer needs to install a station (mentioned on CPU) on which he is working; station can be installed by clicking on + box next to project name. The SIMATIC Manager displays the hardware object in the right half of the window. Programmer needs to configure hardware network. Configuring means inserting the configuration of the actual hardware modules that programmer has used in PLC on the software. Hardware configuration table gives the information about MLFB (Machine Readable Product Designation) number, MPI address of CPU, addresses of analog and digital I/O modules. System data in fig. 4 saves the hardware configuration.

The programming software provides various types of blocks in which the user program and the related data can be stored. Depending on the requirement of the process, program can be structured in different blocks. Different blocks available for programming in PLC are as follows:

i) *Organization Blocks (OB):*

They form the interface between the operating system and the user program. They are named as OB followed by integer number from 1 to 127 (e.g., OB1). Similarly there are different types of OBs which can be used for different purposes. OBs cannot be called by programmer they are implicitly called by operating system. OB1 is a cyclic OB and

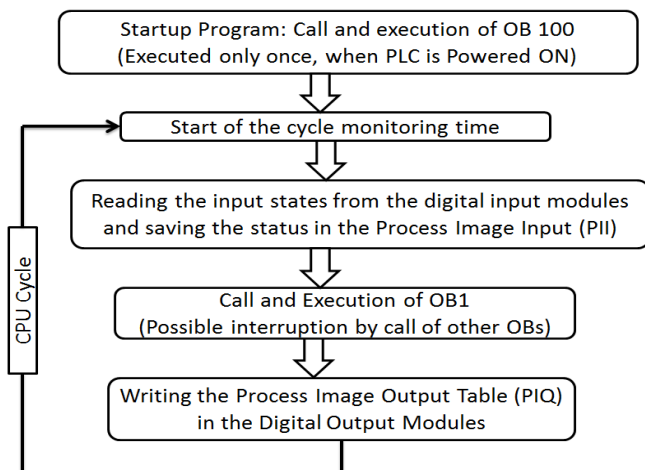


Fig. 5 -Sequence of cyclic Program Processing

the entire program can be stored in OB1 that is cyclically called by operating system. The OBs such as OB100, OB101 and OB102, have startup program stored in them and is executed only once when CPU is turned ON. Fig. 5 shows cyclic program execution.

ii) *Functions (FC):*

A function contains the partial functionality of the program. It is a logic block “without memory”. Temporary variables belonging to FC are saved in the Local data stack (L-Stack). Temporary variables are deleted after the associated block is executed. An FC is always executed by calling it in another block. Number of FCs applicable will be CPU specific. FC can be used to return a value to calling block (for example, mathematical functions signal control with binary logic operations). There are different types of functions used in programming they are,

- User Functions (User needs to insert)
- Standard Functions (User needs to import it from std. library of software)
- System Functions (It is an integrated part of the CPU user needs to call it in logic blocks, abbreviated as SFC) such as Timer and Counter Functions to implement timing and counting sequences respectively.

iii) *Data Blocks (DB):*

These blocks are used by the logic blocks of the user program for storing values. Unlike the temporary data, the data in data blocks is not overwritten when execution of the logic blocks is completed. DBs are of two types Shared (Global) DB and Instance DB. The information stored in global DB can be accessed by all the logic blocks in user program, while it is not the case with instance as they are always assigned to a particular Function Block (FB). Data is stored in variables of the user program which are uniquely identified by:

- a) Storage location: Addressing of variable follows the format of identifier followed by byte.bit address e.g., I124.2 (I-Identifier, 124-byte no, 2-bit no), M20. This format is applicable for PII, PIQ and bit memory.
- b) Data types: All data stored in the user data block must be identified by the data type.

Elementary, Complex and User defined data types (UDT) are the available data types. The data types determine the amount of memory space required. Elementary data types are never more than 32bits long (BOOL, byte, word, etc.); while complex can be longer than 32bits (array, structure, date & time). UDT are special data structures user creates that can be used in the program once defined.

iv) *Function Blocks (FB):*

They belong to the blocks that need to be programmed by the user. A FB is block “with memory”. It is assigned a data block as its memory (instance DB). A FB contains a program that is always executed when a different logic block calls the FB. FB makes it much easier to program frequently occurring complex functions. They can be called by OBs, FCs and FBs. Unlike FCs, FBs are initialised i.e., a FB is assigned its own private data area in which FB can remember process states. FBs can have static as well as temporary variables. Static variables are retained even after the block is executed and is accessible throughout the programme.

Elements used in programming the PLC: The elements used in PLC programming depend on the programming language. In LAD language some of the elements used in logic are:

- -| |- Normally Open (NO) contacts
- -|/|- Normally Closed (NC) contacts
- -()- regular coil.

The fig. 6 shows an example of a small control system. A tank is to be filled with water it also has a lamp connected to it. Owner of the plant has demanded that lamp should be on if water level is within limits.

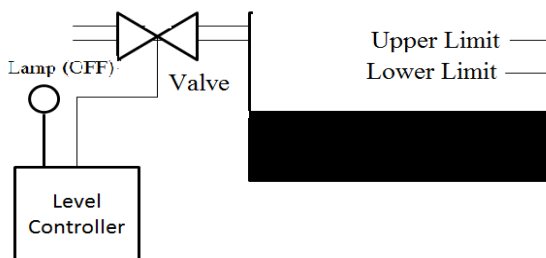


Fig 6: Level Control Application of PLC

In this process PLC is the Controller, which decides controlling action. PLC receives number of inputs from different devices located at different parts of the plants. PLC then decides controlling action on the basis of status of inputs & logic written in its memory. According to the controlling action the output device operates. As it uses programming logic to control the process, called as Programmable Logic Controller. PLC takes decisions without interference of man; hence it is used for automation.

VI. WORKING OF PLC

According to the requirements on the field, programmer designs the logic in the software. The designed programme needs to be downloaded on the load memory of the PLC. Load memory may be RAM or EPROM. Now, as the program is loaded in CPU of PLC it can be executed by processor of the CPU. The sensors present at the field sense the incoming signal and converts into electrical signal and this signal is further sent to input modules of the PLC. Corresponding data of input signal is stored in PII region of system memory. This data is further processed in work memory according to the logic developed by the programmer. After processing the input data the outcome of operation is stored in PIQ region of system memory. The data in the region can be read on output modules which controls the actuators on the field. If mechanism of plant is to be altered, then same PLC can be used by just changing the logic of the program in PG, while it's not the case with relay logic where the technician needs to change the whole wiring.

VII. COMPARISONS OF PLC WITH OTHER CONTROL SYSTEMS

PLCs are well-adapted to a range of automation tasks. This industrial process is mainly used where changes to the system would be expected during its operational life. PLC applications are typically highly customized systems so the cost of a packaged PLC is low compared to the cost of a specific custom-built controller design. Very complex process control, such as used in the chemical industry, may require algorithms and performance beyond the capability of even high-performance PLCs. Very high-speed or precision controls may also require customized

Parameters	Relays	Solid-State Controls	Microprocessor	Minicomputer	Digital Logic	PLCs
Hardware cost	Low	Equal	Low	High	Average (can be high in small quantities)	Depends on number of controls
Versatility	Low	Low	Yes	Yes		Yes
Troubleshooting and maintainability	Poor	Poor	Poor	Poor	Poor if IC's are soldered	Good
Computer compatible	No	No	Yes	Yes	Yes	Yes
Arithmetic capability	No	No	Yes	Yes	Yes	Yes
Programming cost	(Wiring) High	(Wiring) High	High	High	Low	Low
Reusable	No	No	Yes	Yes		Yes
Space required	Largest	Large	Small	Ok	Fairly compact	Small
Operating speed	Slow	Faster than electro-mechanical relays	Fairly fast	Fairly fast	Fairly fast	Fast

Fig. 7 - Comparison of PLCs with other control systems

solutions. PLC and other control devices such as Remote Terminal Units (RTU) and distributed control system (DCS) are increasingly beginning to overlap in responsibilities [4]. The above figure shows the comparison of PLC with other control systems.

VII. APPLICATIONS OF PLC

Depending on the type of technology used, controllers can be divided into pneumatic, hydraulic, electrical and electronic controllers. The fig. 8 shows gives some of the applications of PLC.

Application	Description
Radio Frequency Remote Control	To control large equipment at a safe distance
Residential Irrigation Control	Time controlled fertilizing
Airport Runway Lighting Control	Remote or from Tower
Smoke House Control	Regulating temperature and smoking process

Fig. 8 - Applications of PLC

VIII. CONCLUSION

PLC was invented in response to the needs of the American automotive manufacturing industry. Programmable logic controllers were initially adopted by the automotive industry where software revision replaced the re-wiring of relay control panels when production models changed. Ladder logic is the main programming method used for PLCs. It has been developed to mimic relay logic. The decision to use the relay logic diagrams was a strategic one. By selecting ladder logic as the main programming

method, the amount of retraining needed for engineers and trades people was greatly reduced. Modern control systems still include relays, but these are rarely used for logic.

References

- [1] D. Morley, looks back on the 40 th anniversary of the PLC Manufacturing Automation, 12 September, 2008.
- [2] E. A. Parr, Industrial Control Handbook, Industrial Press Inc., 1999.
- [3] M. A. Laughton, D. J. Warne, Electrical Engineer's Reference book, 16 th Edition, Newnes, 2003.
- [4] Richa Natto, Aditya Bagri, International Journal of Computer Applications, Vol 77, No. 11, September 2013.
- [5] Eric Bruton, Automation, English Language Book Society.
- [6] Siemens, SIMATIC S7-300 Module data Manual, February 2013.
- [7] Siemens, SIMATIC Programming with STEP 7 Manual, March 2006.