Chapter 1

Programmable Logic Controllers (PLCs)
An Overview

1.1

Programmable Logic Controllers

Programmable logic controllers are now the most widely used industrial process control technology.

A PLC is basically a digital computer designed for use in machine control. It has been designed to operate in the industrial environment and is equipped with special input/output interfaces and a control programming language.

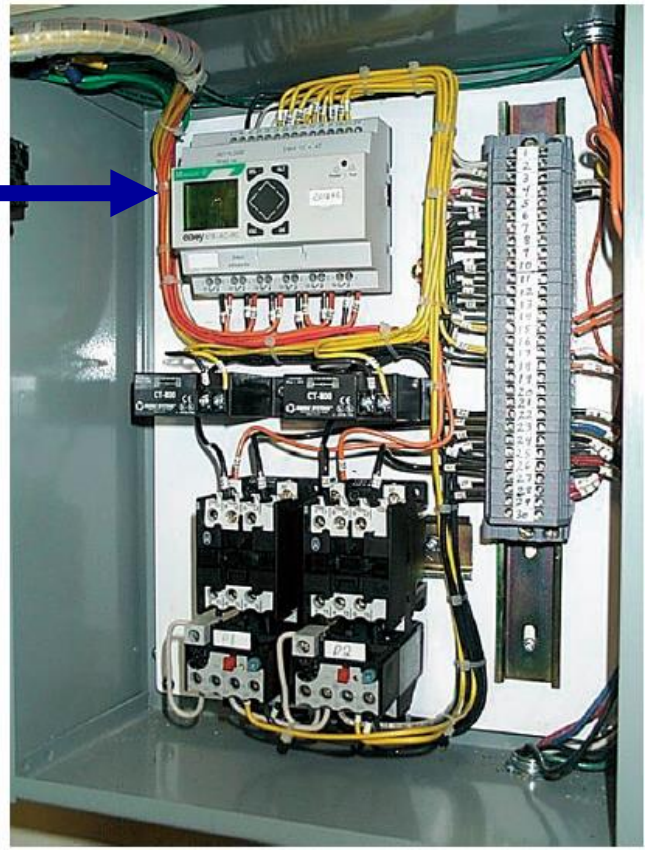
Initially the PLC was used to replace relay logic, but its ever-increasing range of functions includes timing, counting, calculating, comparing, and the processing of analog signals.

The structure of a PLC is based on the same principles as those employed in computer architecture.

Programmable Logic Controller Advantages

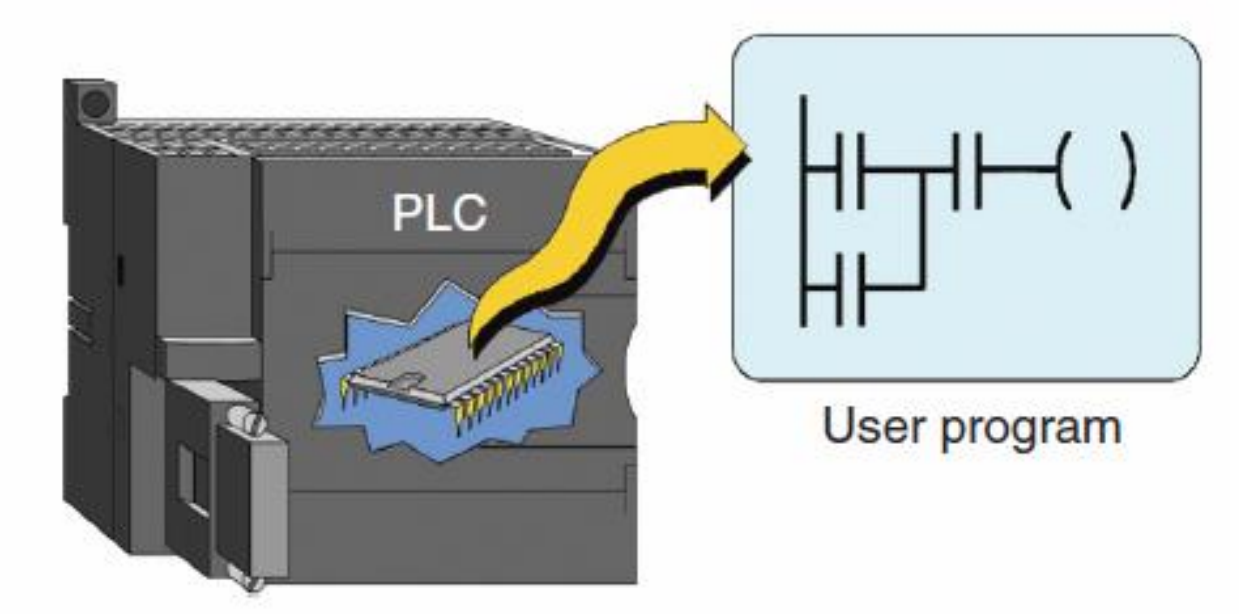
Relay control panel

PLC control panel



Elimination of much of the hardwiring associated with conventional relay control circuits.

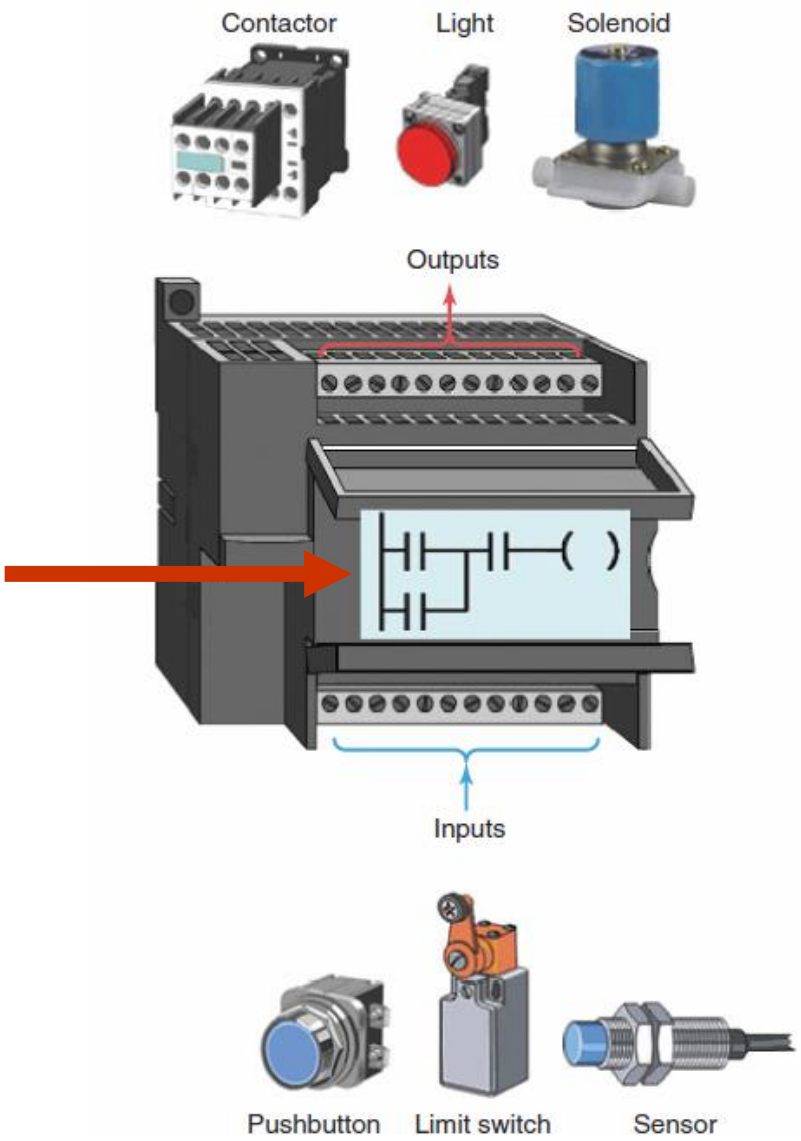
Programmable Logic Controller Advantages



Increased Reliability **- Since all the logic is contained in the PLC's memory, there is no chance of making a logic wiring error.**

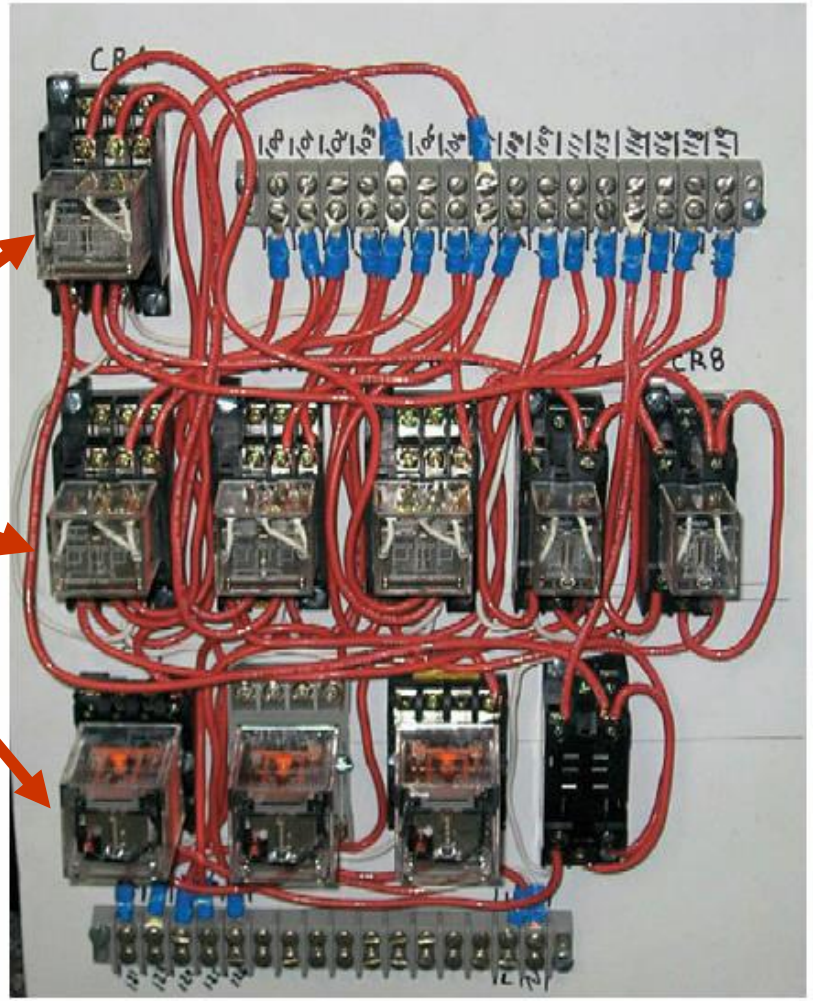
Programmable Logic Controller Advantages

More Flexibility **- It is easier to create and change a program in a PLC than to wire and rewire a circuit.**



Programmable Logic Controller Advantages

Lower Cost **- Generally, if an application has more than about a half-dozen control relays, it will probably be less expensive to install a PLC.**



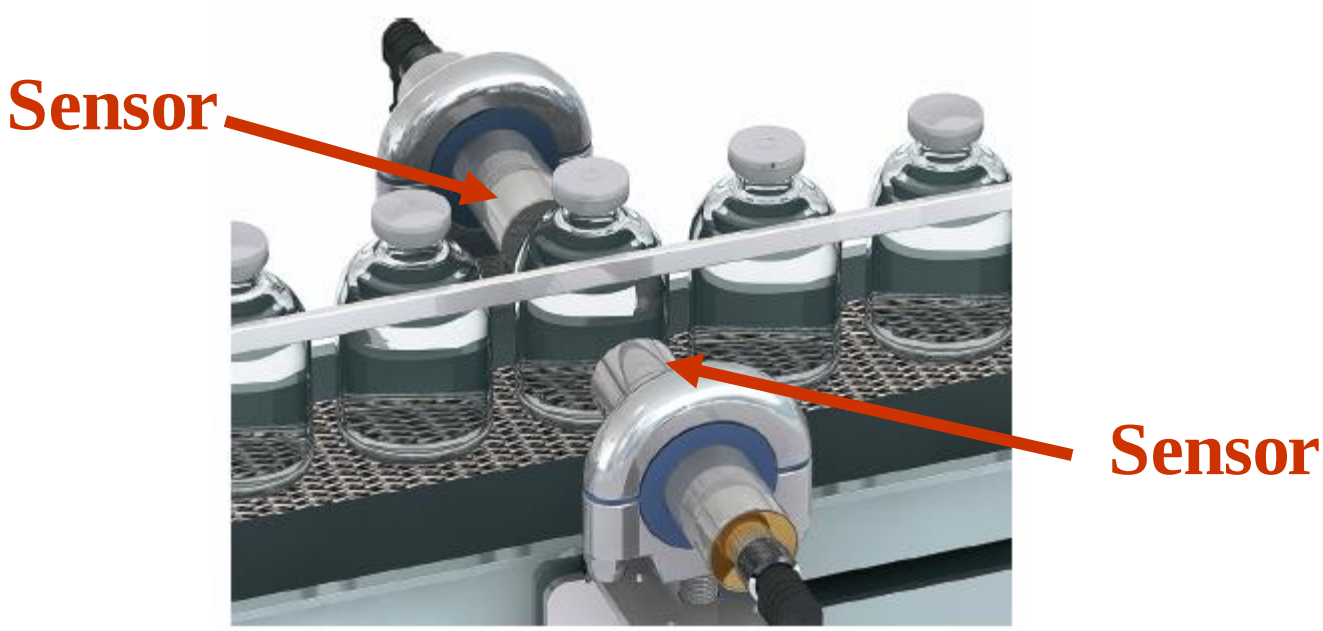
Programmable Logic Controller Advantages

Communications Capability

- A PLC can communicate with other controllers or computer equipment to perform such functions as supervisory control, data gathering, monitoring devices and process parameters, and download and upload of programs.

PLC Communications Module

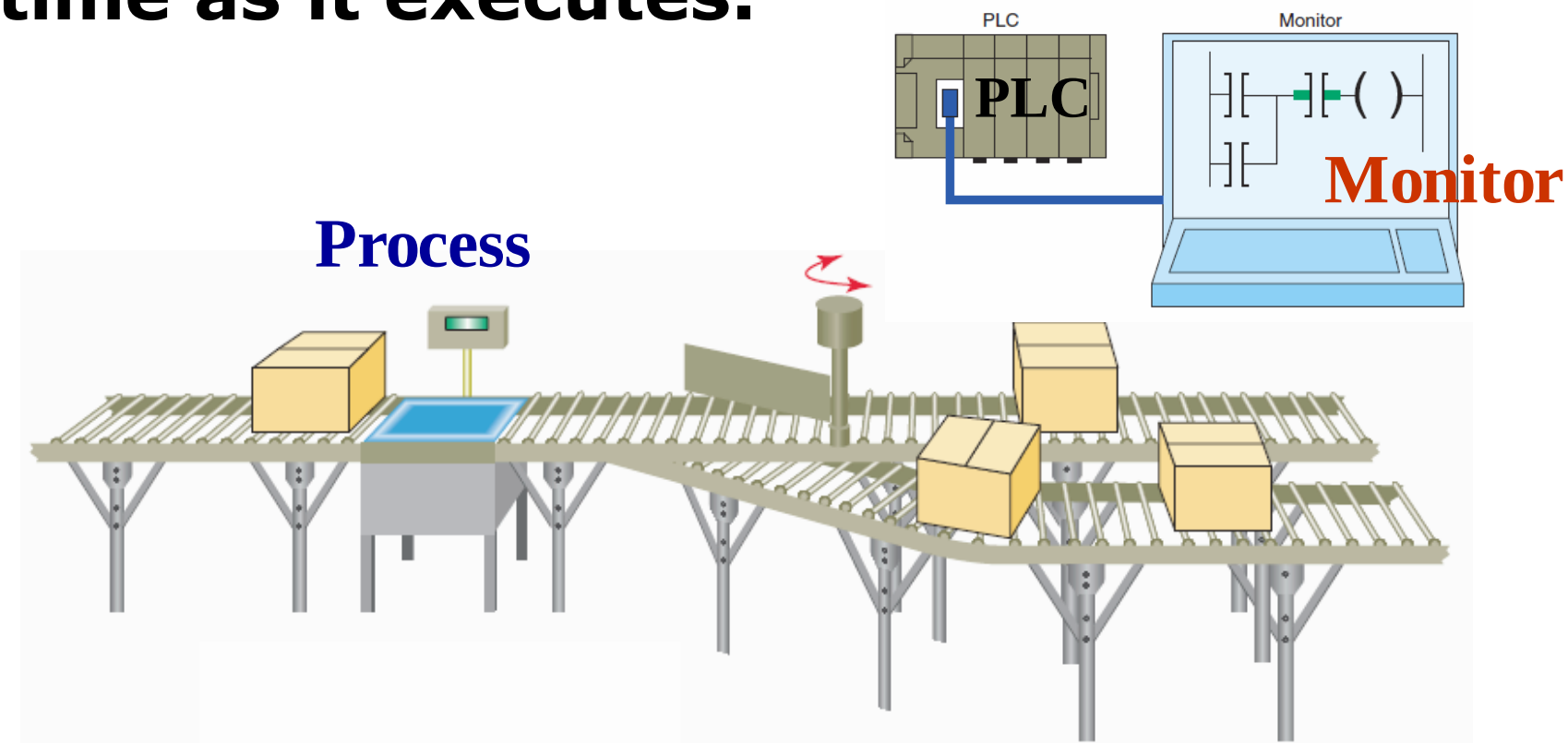
Programmable Logic Controller Advantages



Faster Response Time **- Machines that process thousands of items per second and objects that spend only a fraction of a second in front of a sensor require the PLC's quick-response capability.**

Programmable Logic Controller Advantages

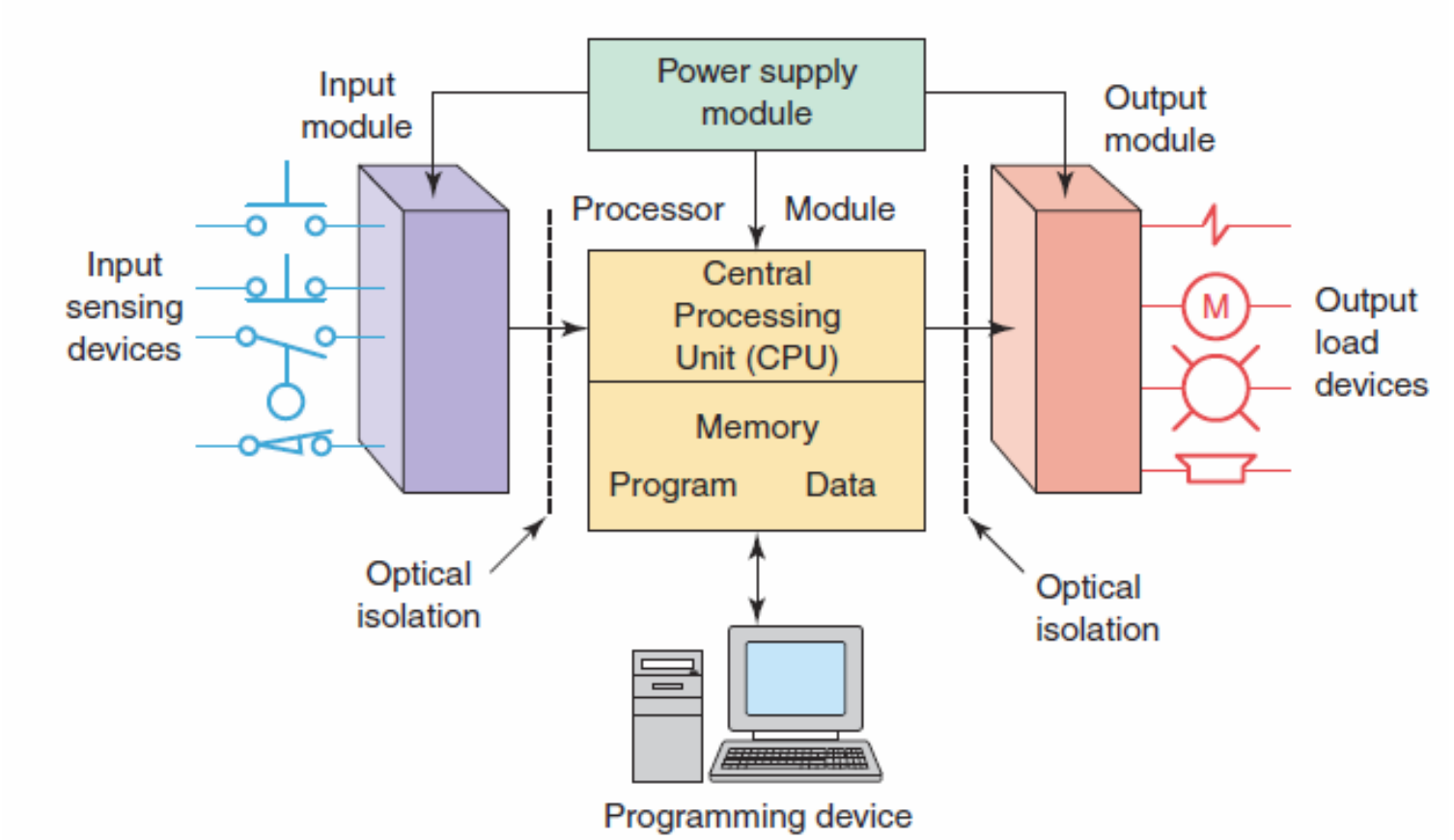
Easier to Troubleshoot **- To find and fix problems, users can display the control program on a monitor and watch it in real time as it executes.**



1.2

Parts of a PLC

The major parts of a PLC system are the *central processing unit (CPU)*, the *input/output (I/O)* section, the *power supply*, and the *programming device.*

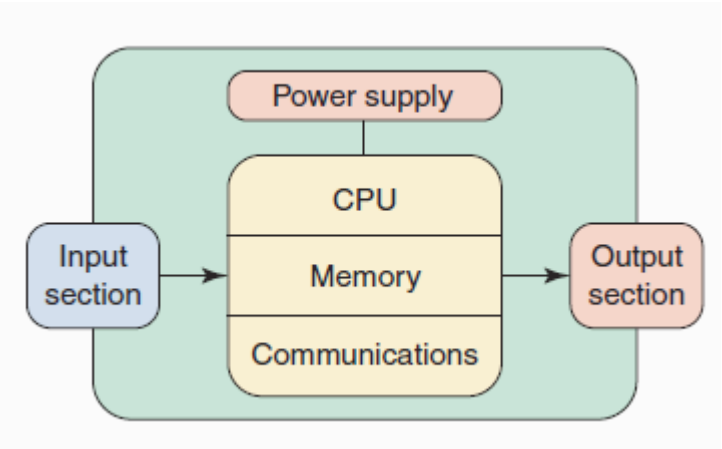


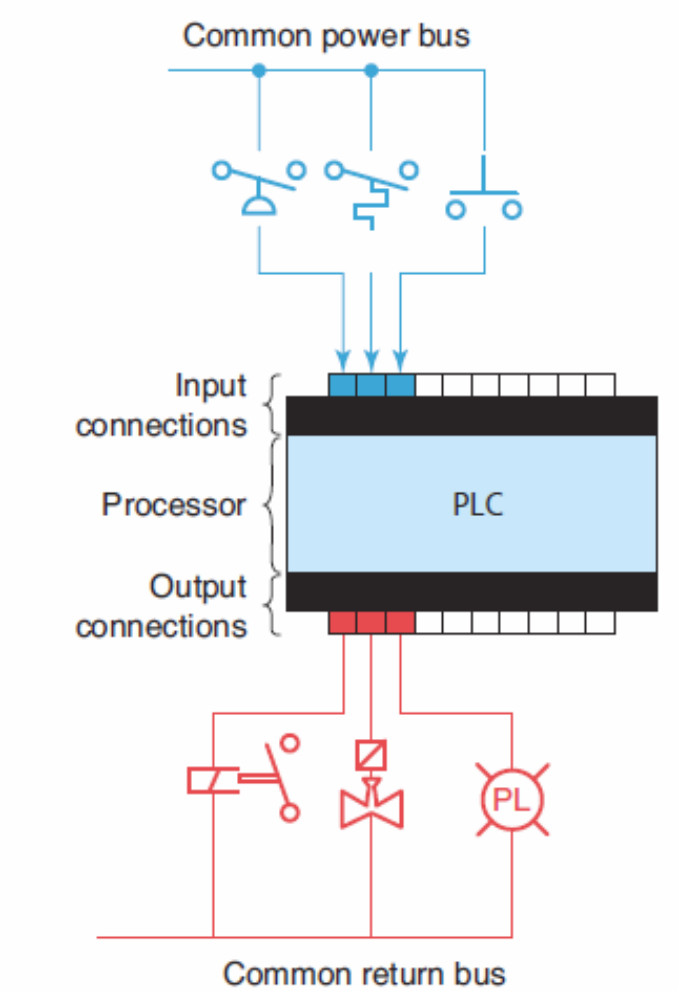
The term *architecture* can refer to PLC hardware, to PLC software, or to a combination of both.

An *open architecture* design allows the system to be connected easily to devices and programs made by other manufacturers.

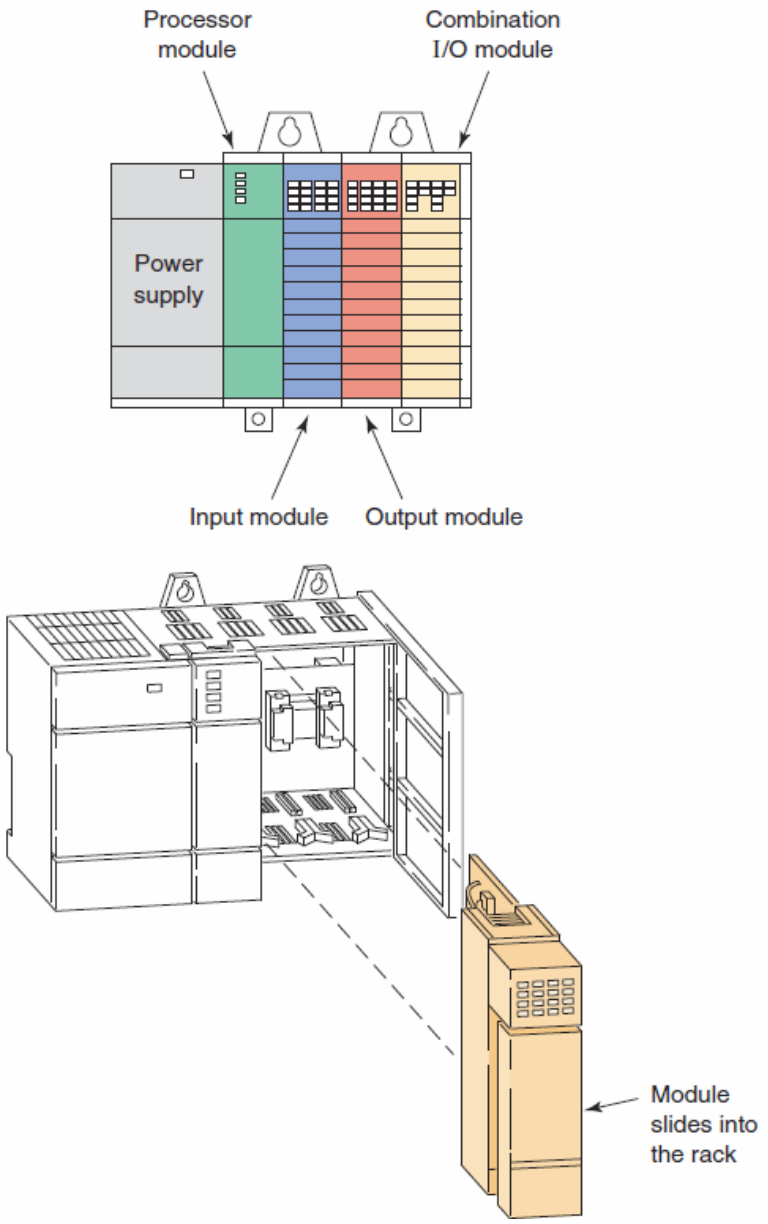
A system with a *closed architecture* is one whose design is *proprietary*, making it more difficult to connect to other systems.

Fixed I/O is typical of small PLCs that come in one package with no separate, removable units. The processor and I/O are packaged together.





Modular I/O **is divided by compartments into which separate modules can be plugged. This feature greatly increases your options and the unit's flexibility.**



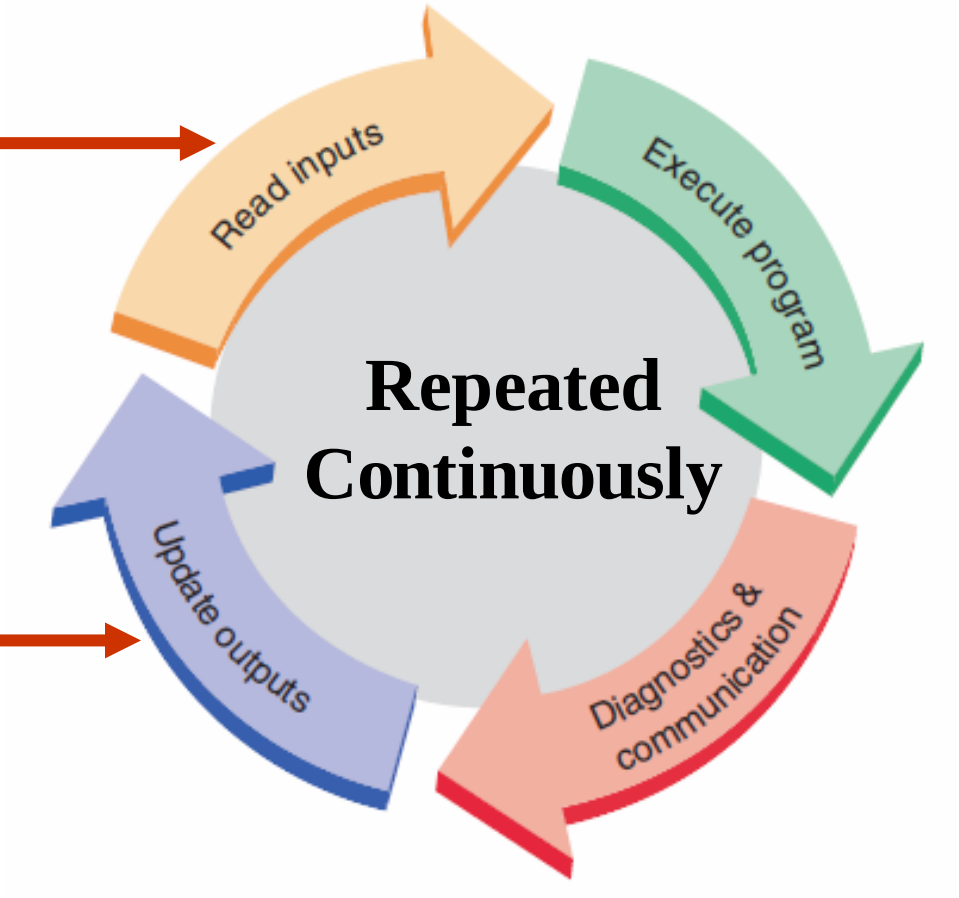
The *power supply* provides DC power to all modules that plug into the rack. For large PLC systems, this power supply does not normally supply power to the field devices.

The *processor* (CPU) consists of a microprocessor for implementing the logic and controlling the communications among the modules.

The processor requires memory for storing the results of the logical operations performed by the microprocessor as well as the operating system and the PLC program.

The PLC program is executed as part of a repetitive process referred to as a *scan*.

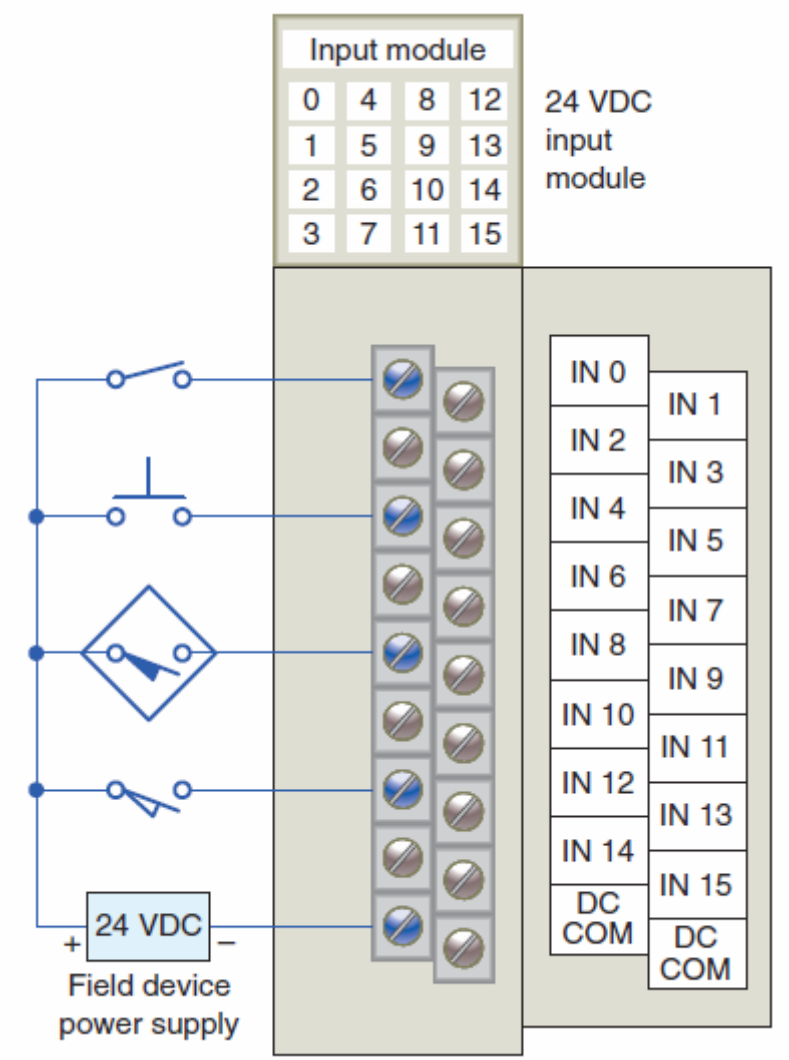
Starts with the CPU reading the status of all inputs.



Concludes with the updating of all outputs.

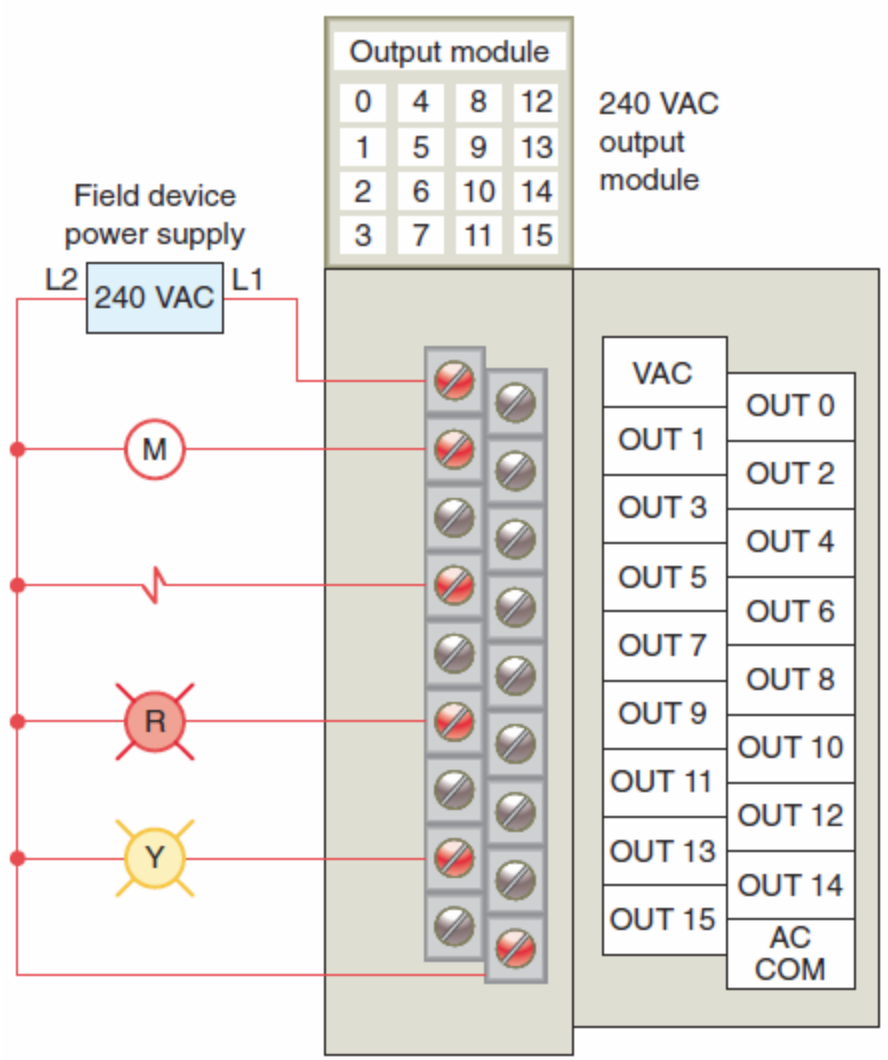
The *I/O (Input/Output) system* forms the interface by which field devices are connected to the controller.

Input devices such as pushbuttons, limit switches, and sensors are hardwired to the *input module* terminals.

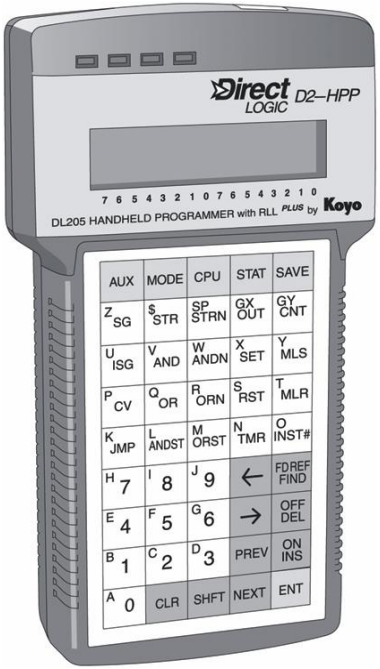


The *I/O (Input/Output) system* forms the interface by which field devices are connected to the controller.

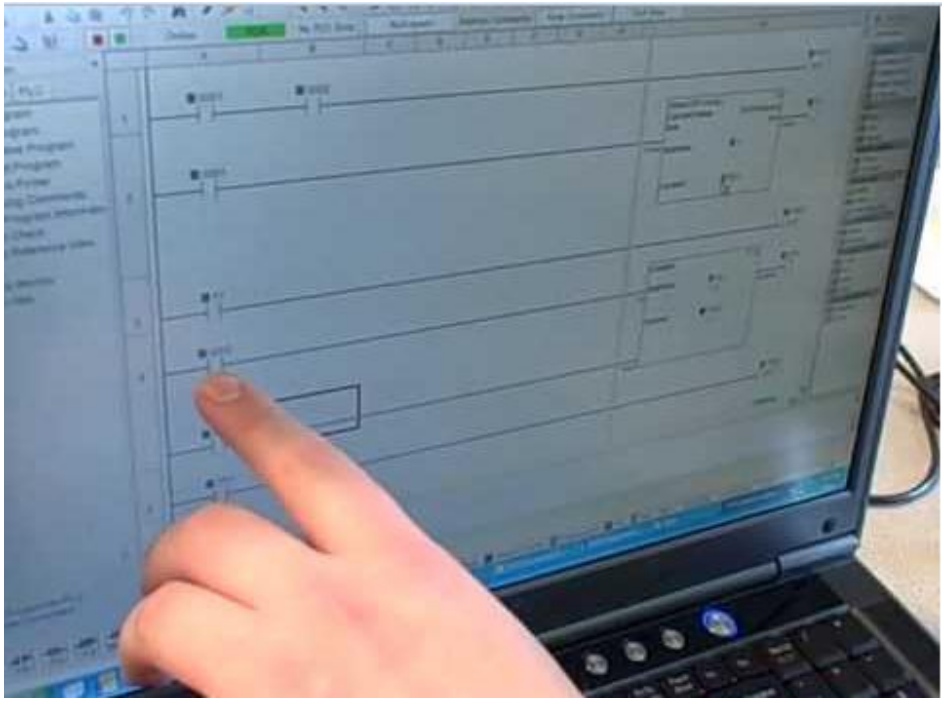
Output devices such as motor starters, solenoid valves, and indicator lights are hardwired to the *output module* terminals.



A *programming device* is used to enter the desired program into the memory of the processor.

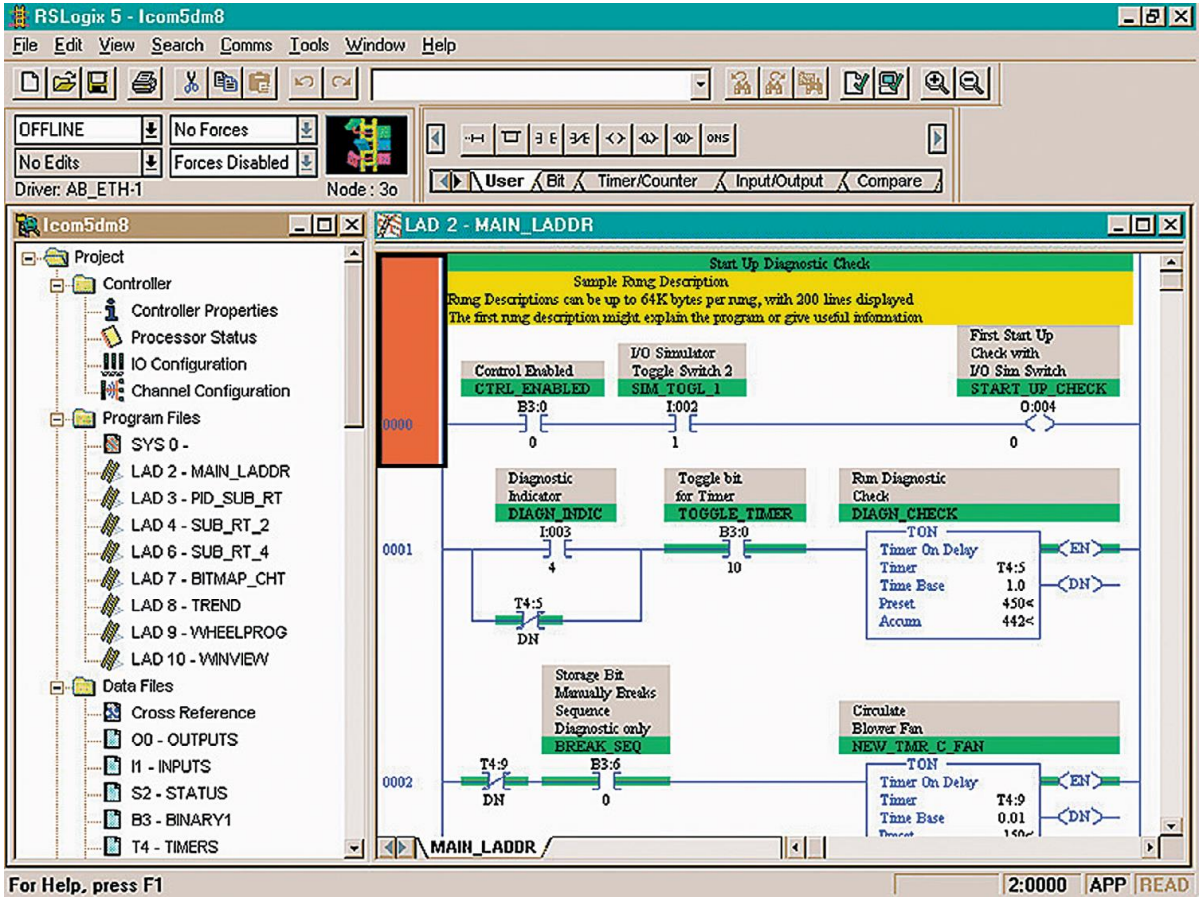


Hand-held programming device.



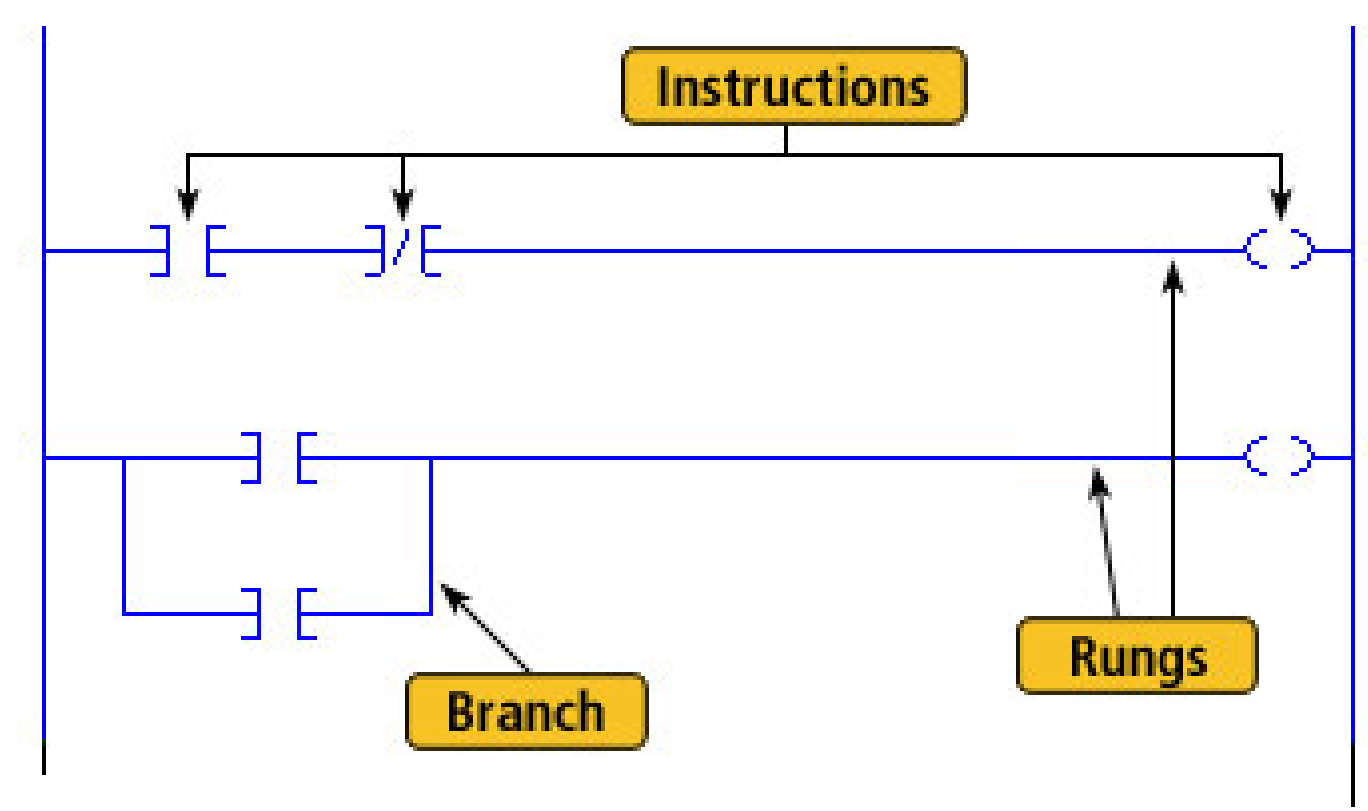
The program is entered using the relay ladder logic programming language.

A *personal computer (PC)* is the most commonly used programming device.



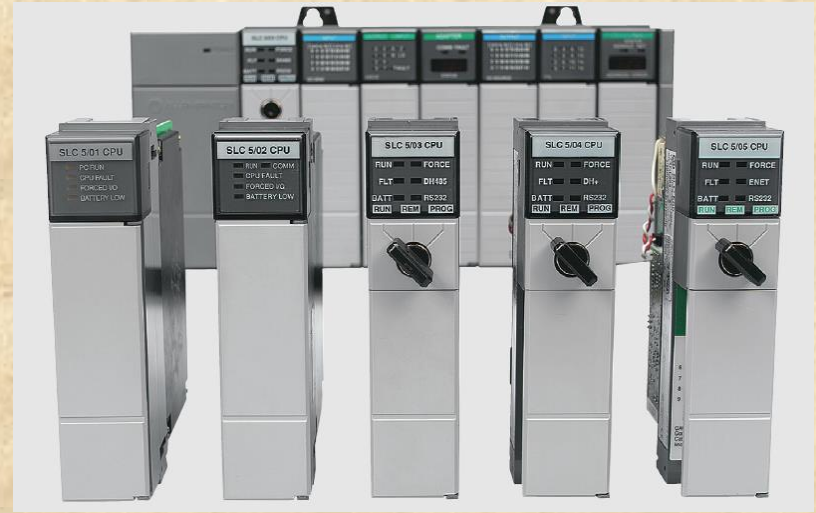
The computer monitor is able to display more logic on the screen thus simplifying interpretation of the program.

A *program* is a series of instructions that directs the PLC to execute actions.



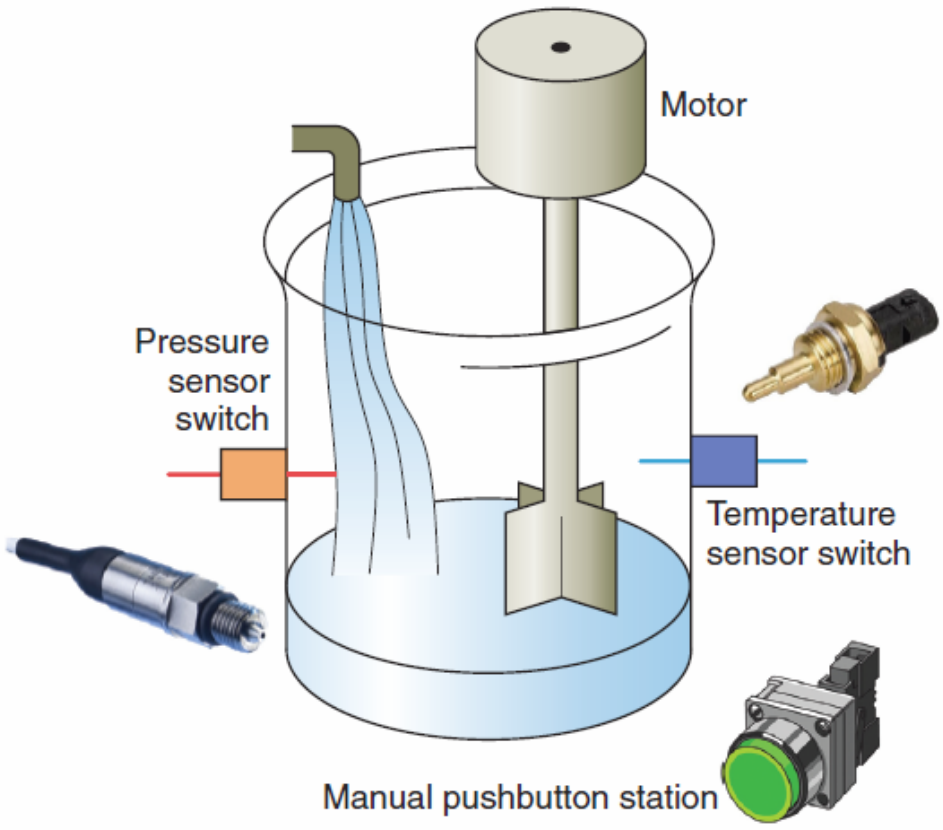
***Relay ladder logic*, the standard programming language, is based on electromagnetic relay control.**

1.3



Principles of Operation

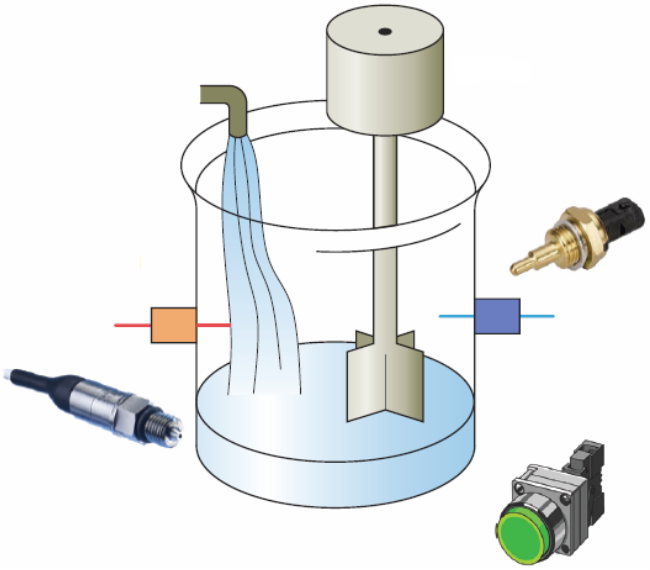
Mixer Process Control Problem

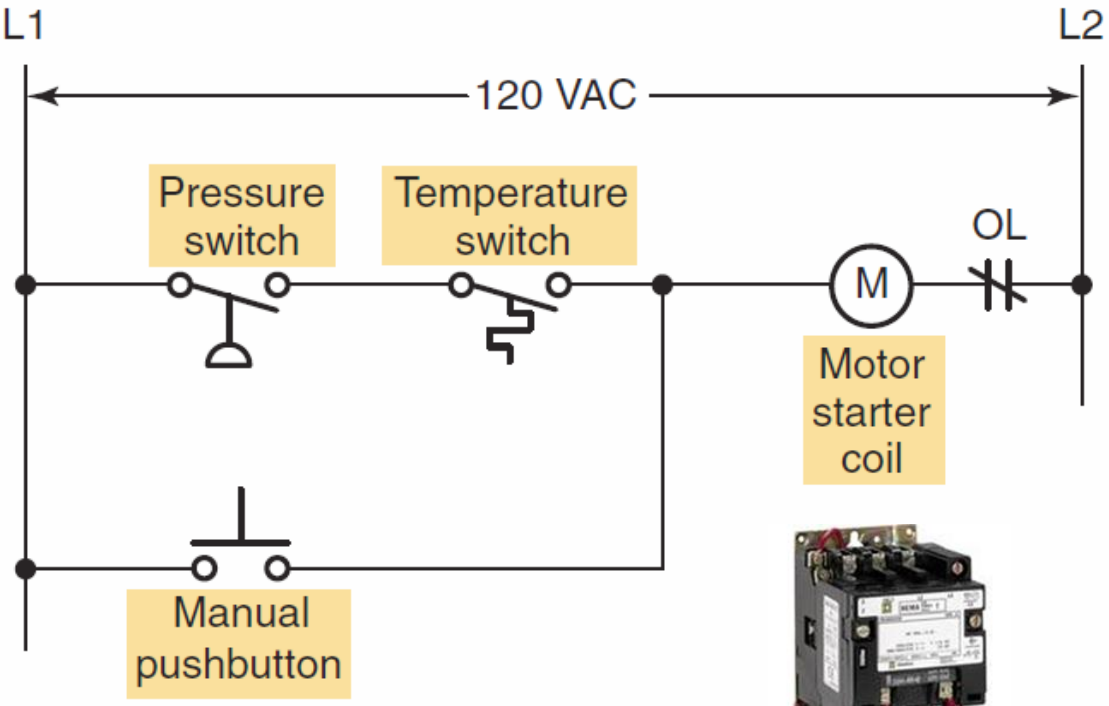


- A mixer motor is to be used to automatically stir the liquid in a vat when the temperature and pressure reach preset values.

- Manual operation of the motor is provided by means of a separate pushbutton station.

Hardwired relay method for control of motor

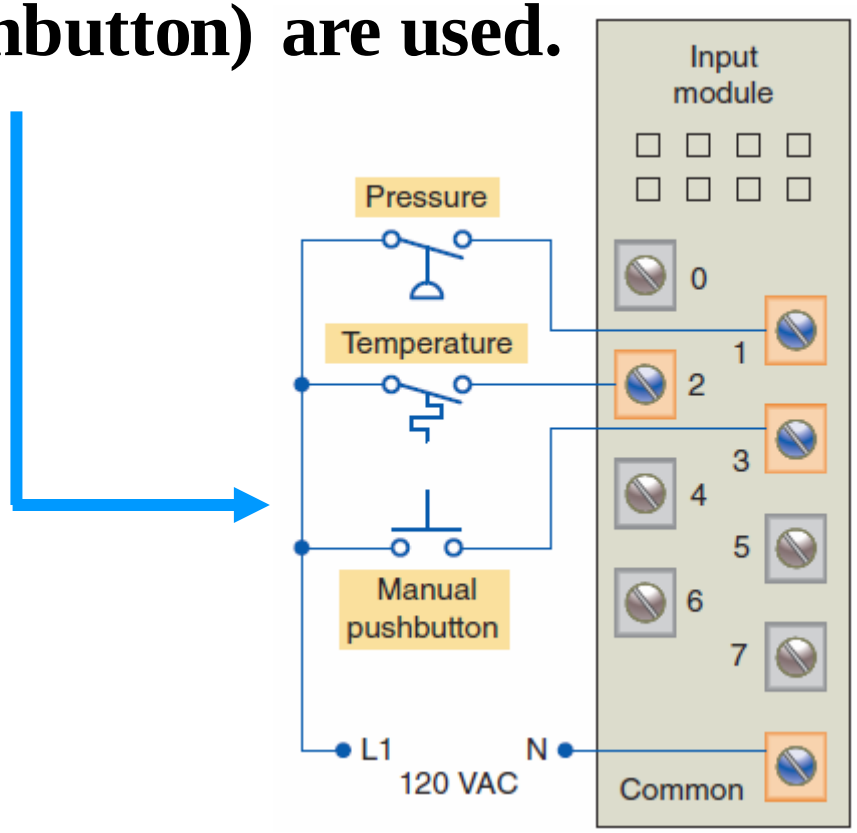




The motor starter coil (M) is energized when *both* the pressure and temperature switches are closed *or* when the manual pushbutton is pressed.

PLC method for control of motor

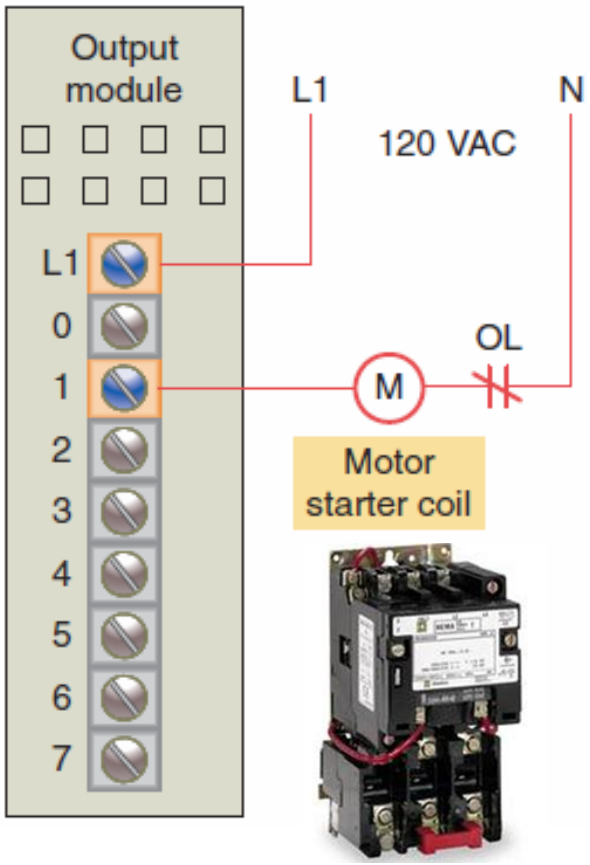
The same input field devices (pressure switch, temperature switch, and pushbutton) are used.



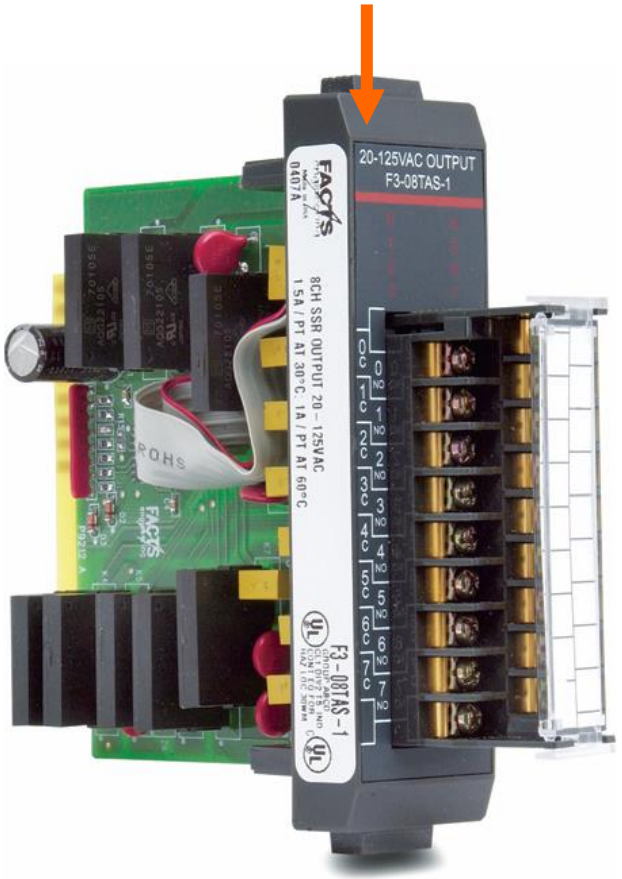
Field devices are hardwired to an input module.

PLC method for control of motor

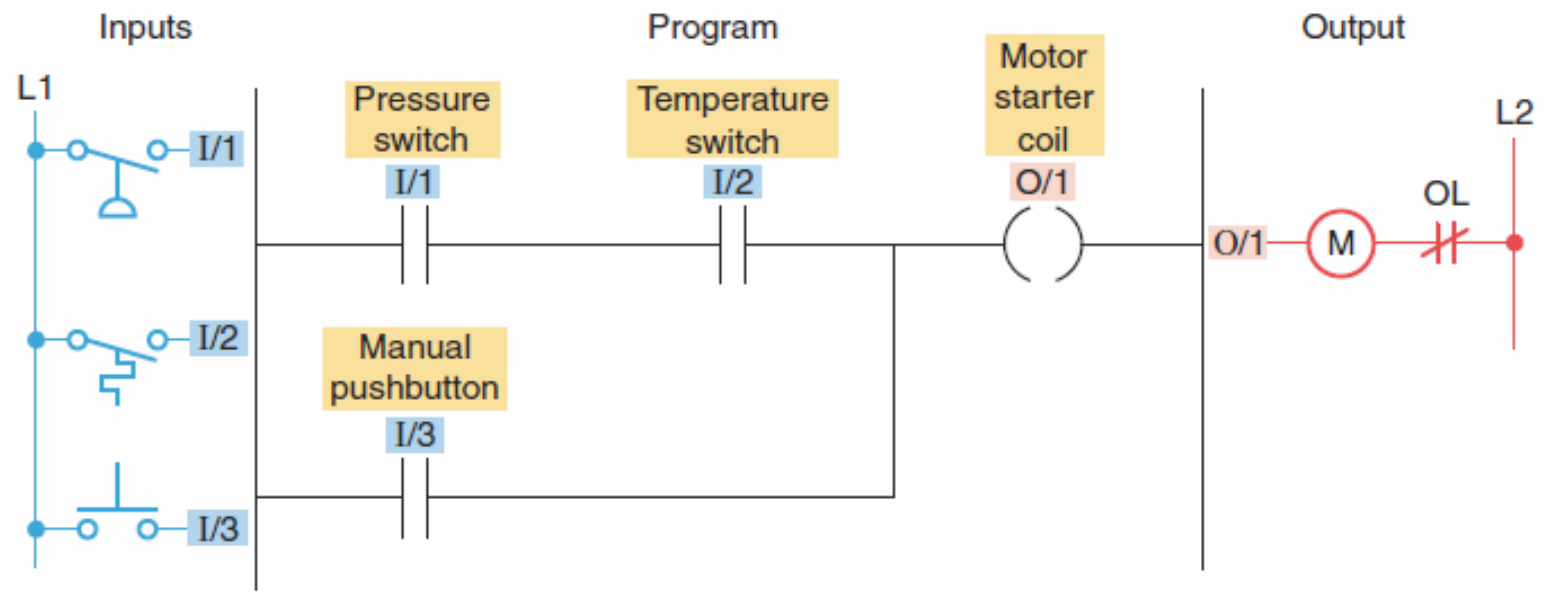
The same output field device (motor starter coil) is used.



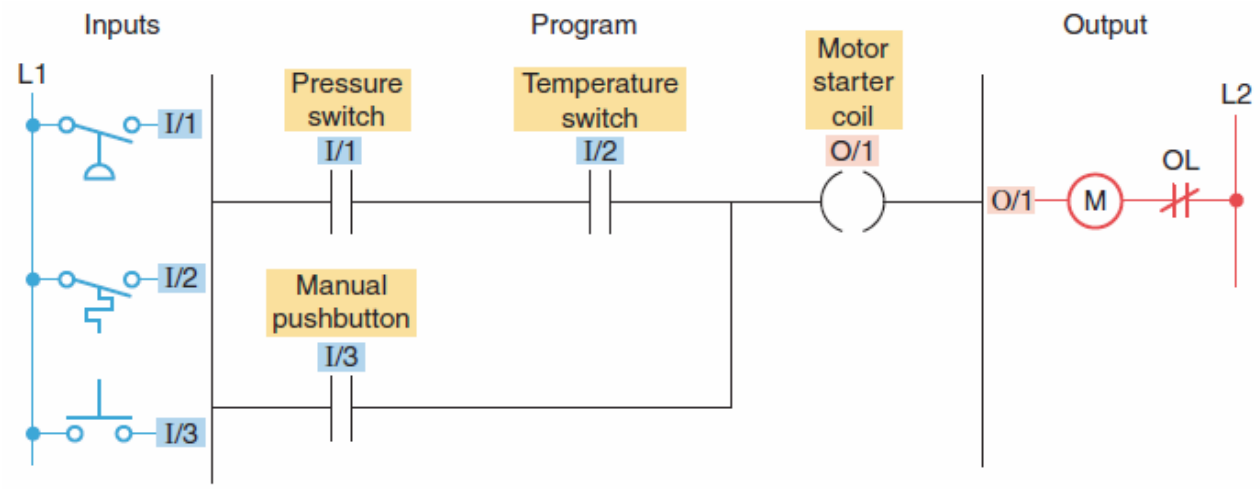
Starter coil is hardwired to an output module.



Enter the PLC ladder logic program into the memory of the CPU.



- The symbols represent instructions and the numbers represent the instruction's addresses.
- Each input and output device is given an *address*, which lets the PLC know where it is connected.

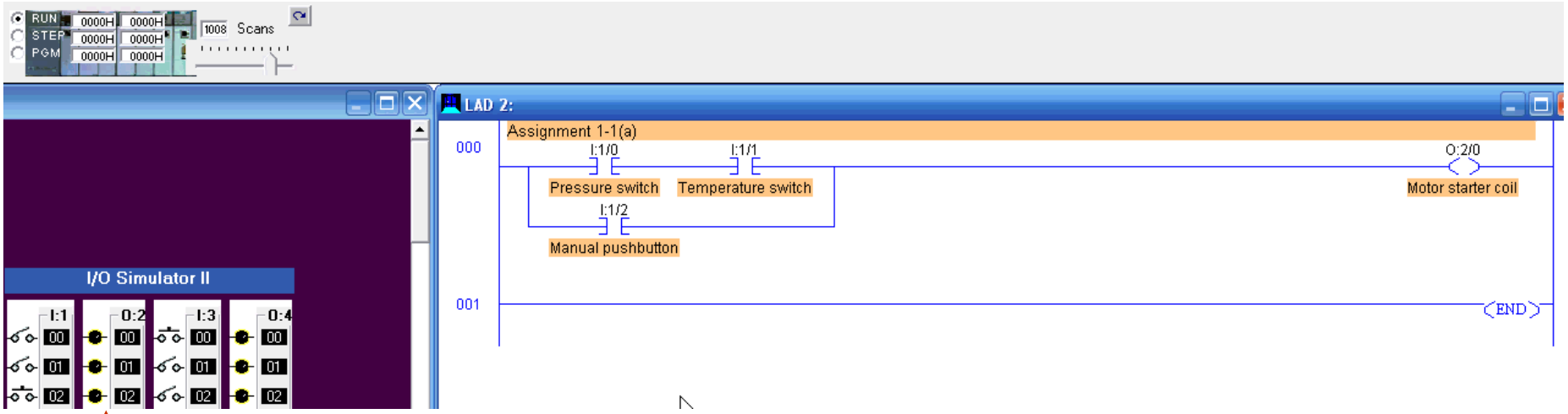


- **For the program to operate, the controller is placed in the RUN mode.**

- **During each scan the controller examines the status of input devices, executes the user program, and changes outputs accordingly.**
- **The coil O/1 is energized when contacts I/1 and I/2 are closed or when contact I/3 is closed.**

- **Either of these conditions provides a continuous logic path across the rung that includes the coil.**

LogicPro simulation of the PLC program

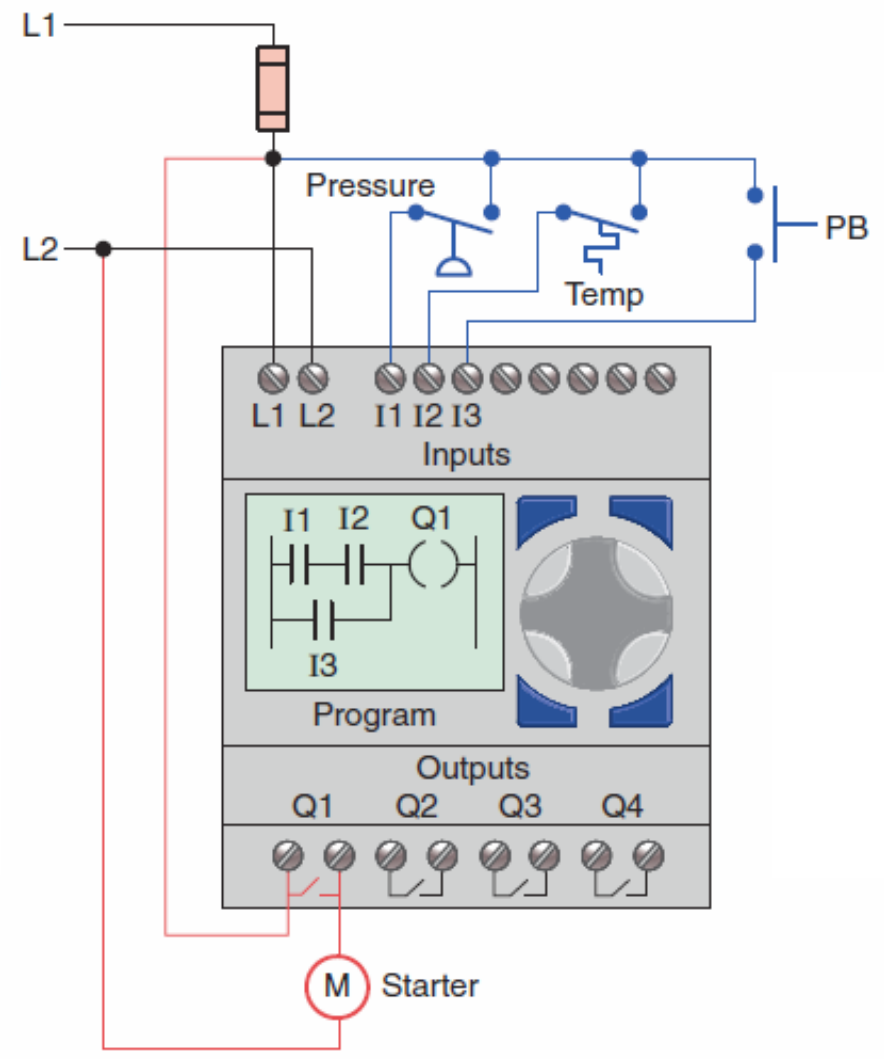


Input Module

Output Module

Ladder Logic Program

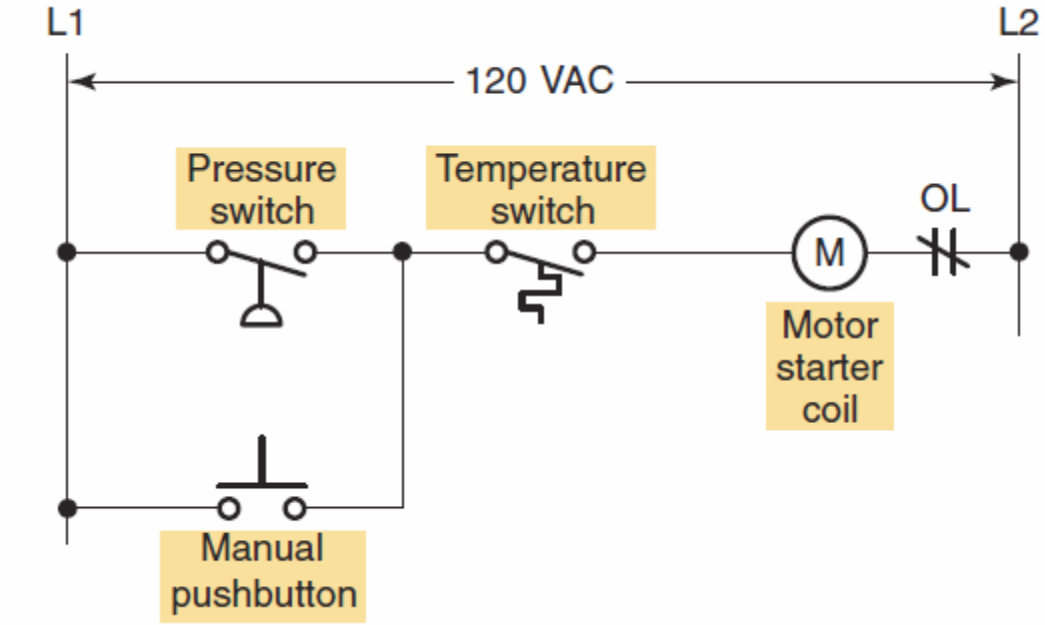
Typical wiring required to implement the process control scheme using a fixed PLC controller.



1.4

Modifying the Operation

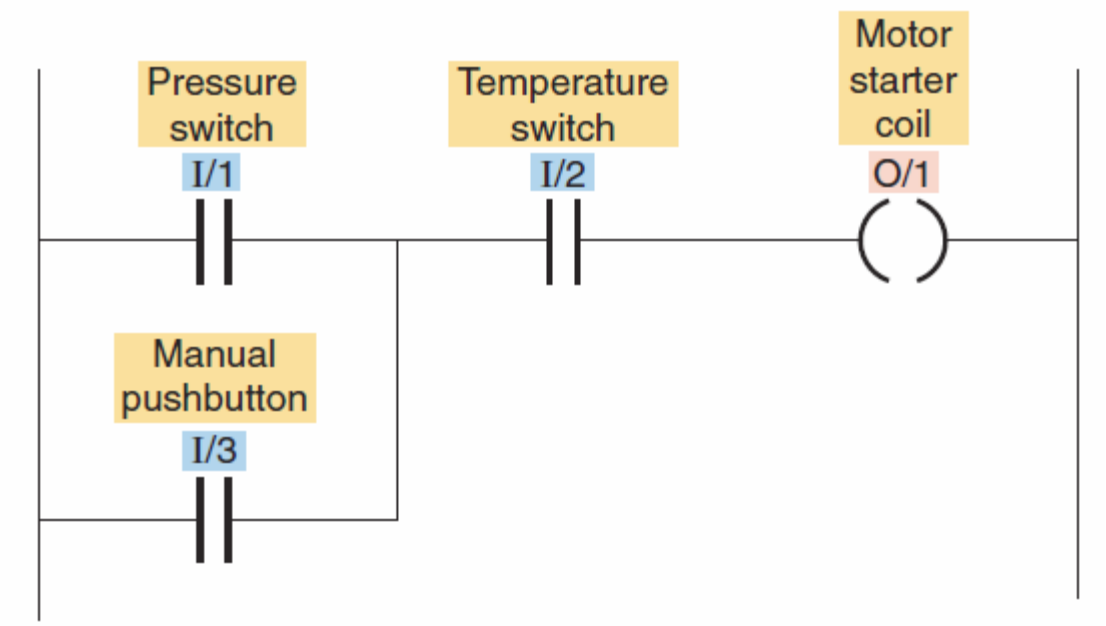
One of the important features of a PLC is the ease with which the program can be changed.



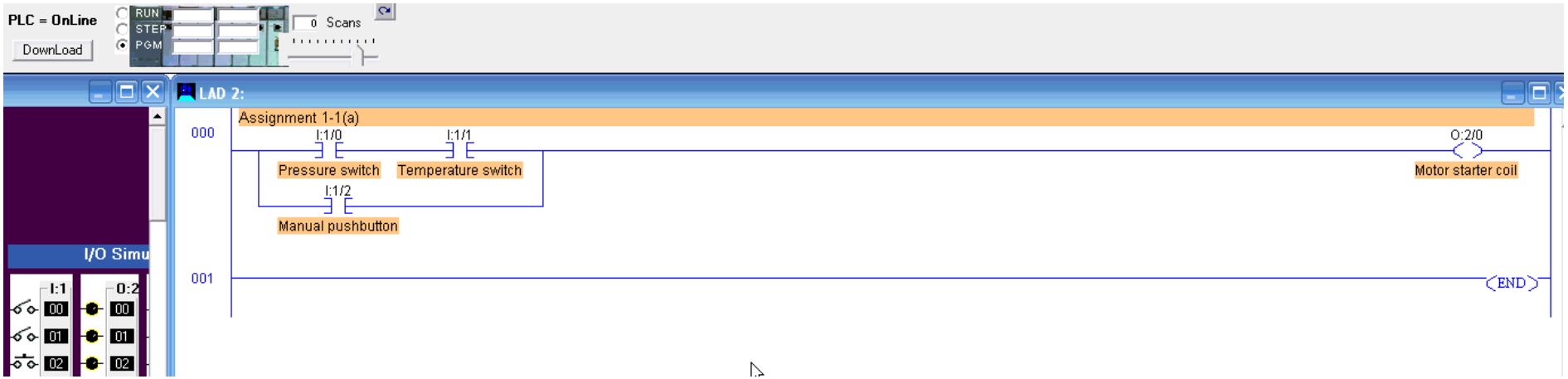
- **Original hardwired process control circuit modified.**

- **Change requires that the manual pushbutton control be permitted to operate at any pressure, but not unless the specified temperature setting has been reached.**

- **If a relay system were used, it would require some rewiring of the circuit**
- **If a PLC system were used, no rewiring would be necessary.**
- **The inputs and outputs are still the same.**
- **All that is required is to change the PLC ladder logic program.**



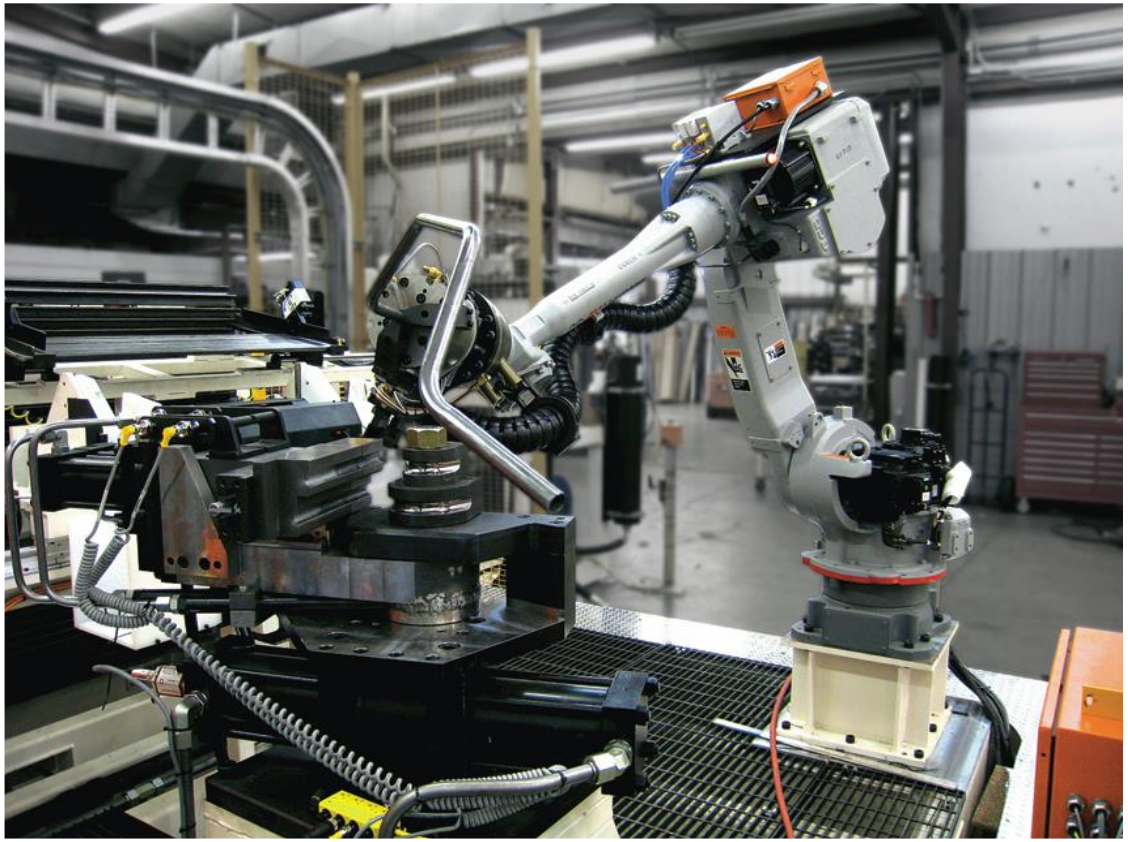
LogicPro simulation of the modified PLC program



1.5

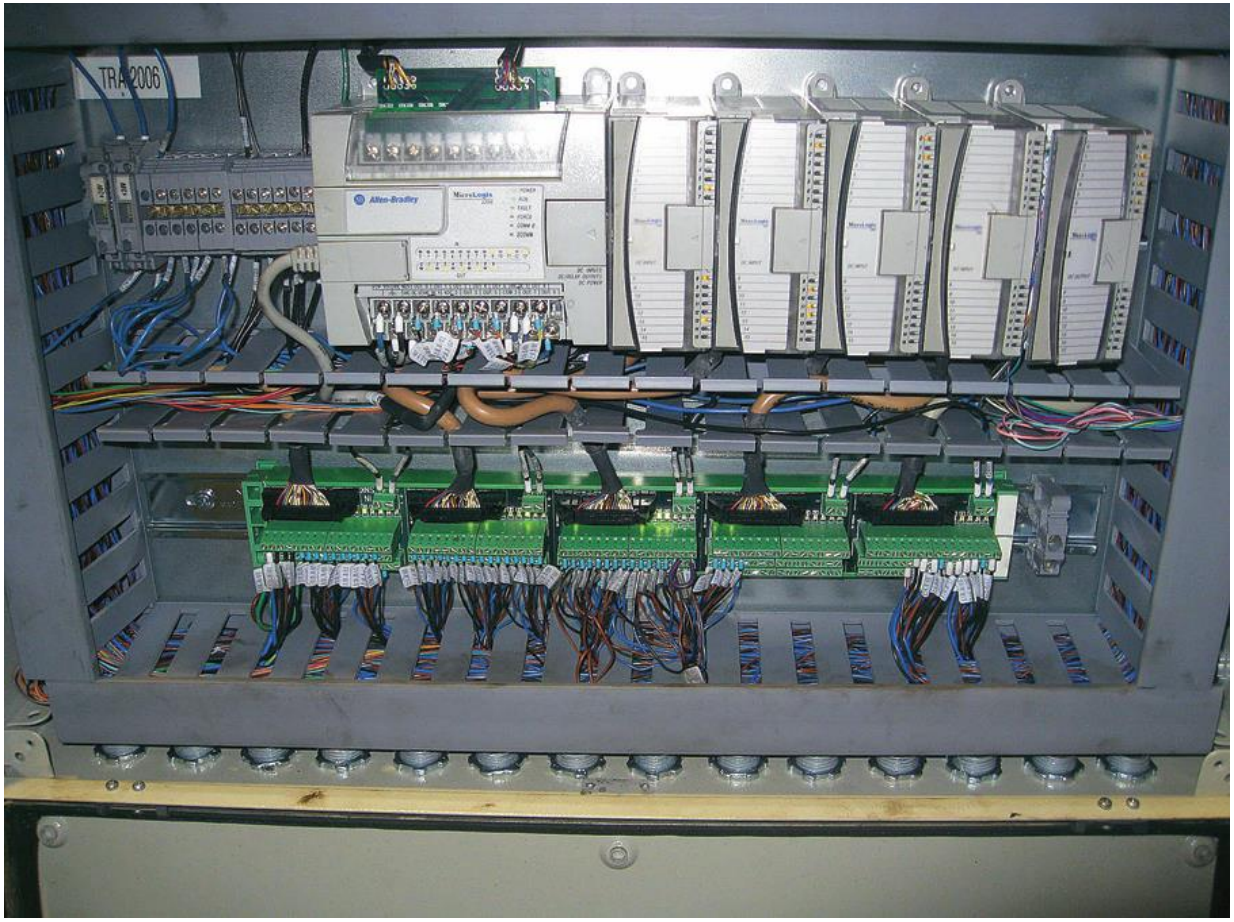
PLCs versus Computers

The architecture of a PLC is basically the same as that of a personal computer.

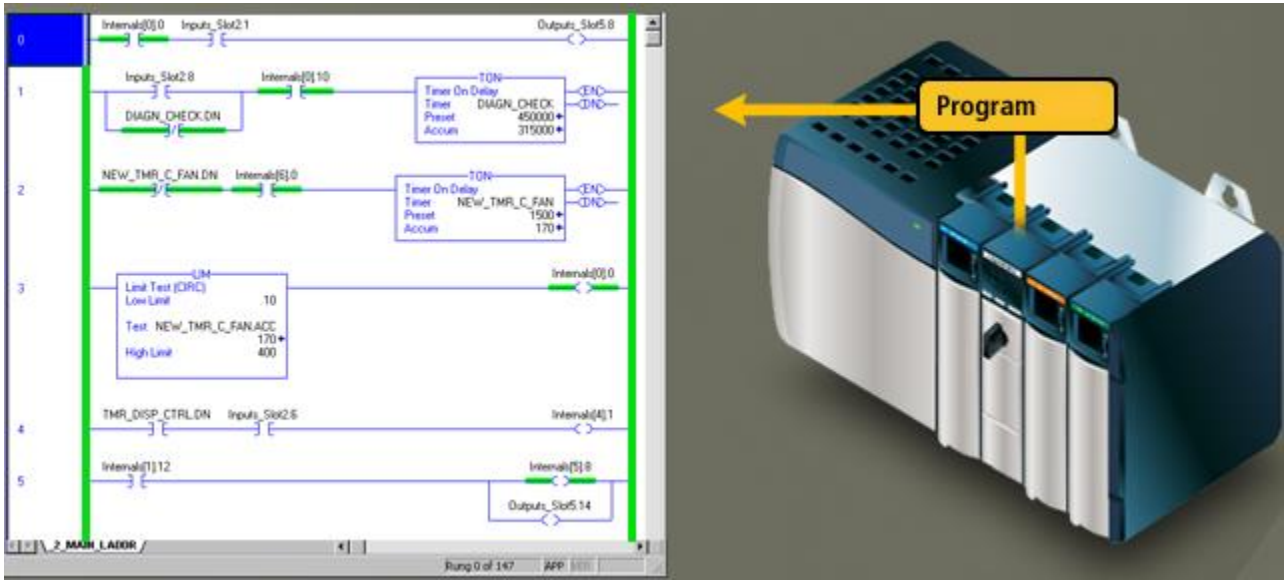


Unlike PCs, the PLC is designed to operate in the industrial environment with wide ranges of ambient temperature and humidity.

A properly designed PLC installation is not as affected by the electrical noise inherent in most industrial locations.



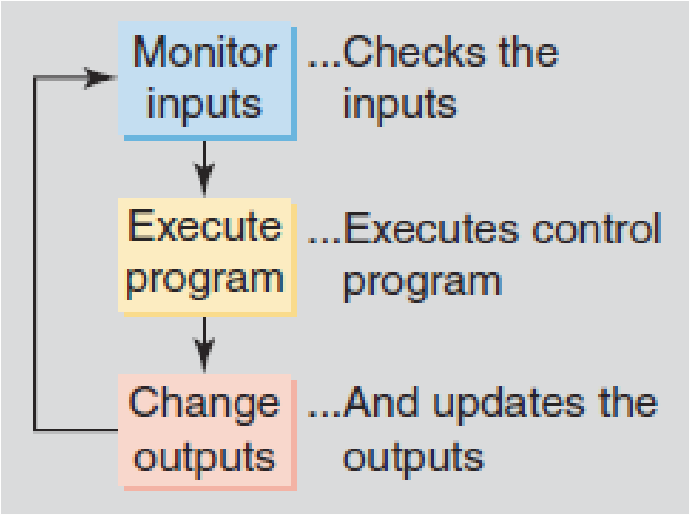
Unlike the personal computer, the PLC is programmed in relay ladder logic or other easily learned languages.



The PLC comes with its program language built into its memory and has no permanently attached keyboard, CD drive, or monitor.

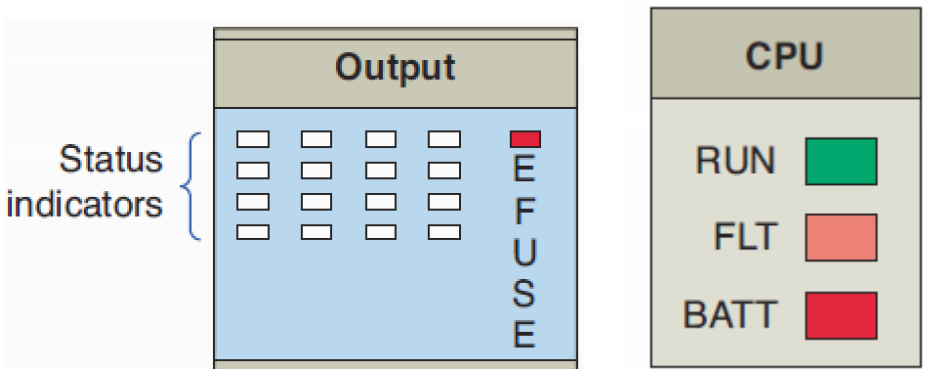
Computers are complex computing machines capable of executing several programs or tasks simultaneously and in any order.

PLCs execute a single program in an orderly and sequential fashion from first to last instruction.

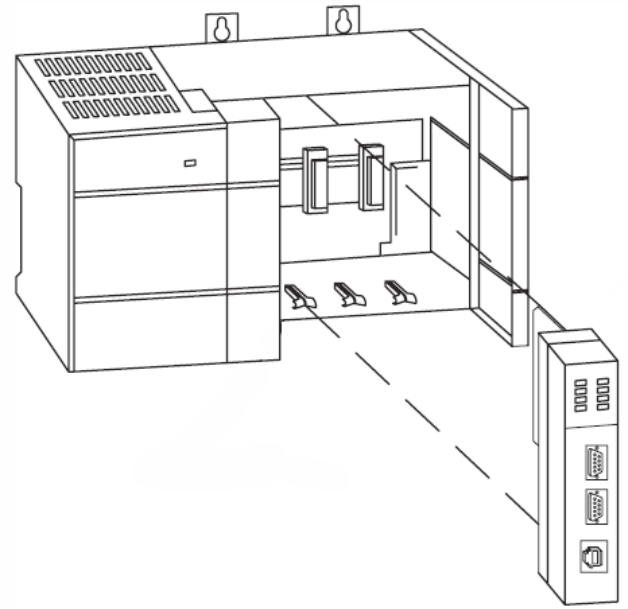


PLC control systems have been designed to be easily installed and maintained.

Troubleshooting is simplified by the use of fault indicators and messaging displayed on the programmer screen.

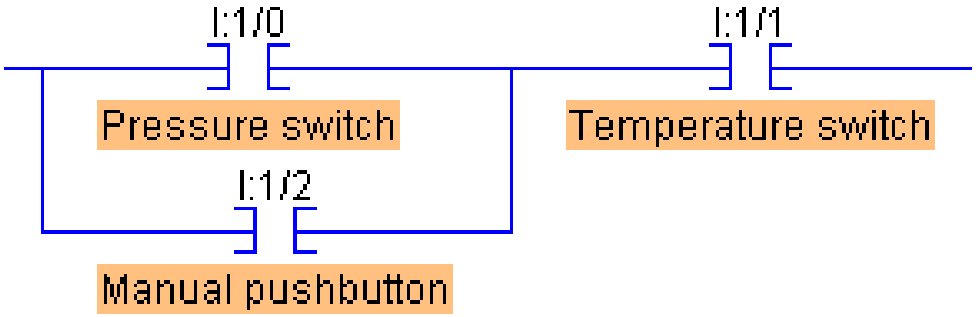


Input/output modules for connecting the field devices are easily connected and replaced.



Software associated with a PLC but written and run on a personal computer falls into the following two broad categories:

- PLC software that allows the user to program and document gives the user the tools to write a PLC program—using ladder logic or another programming language—and document or explain the program in as much detail as is necessary.



PLC software that allows the user to monitor and control the process is also called a *human machine interface (HMI).*

It enables the user to:

➢view a process - or a graphical representation of a process on a monitor

➢ determine how the system is running, trend values, and receive alarm conditions

Programmable Automation Controllers (PACs) combine PLC ruggedness with PC functionality.

Using PACs, you can build advanced systems incorporating software capabilities such as advanced control, communication, data logging, and signal processing with rugged hardware performing logic, motion, process control, and vision.

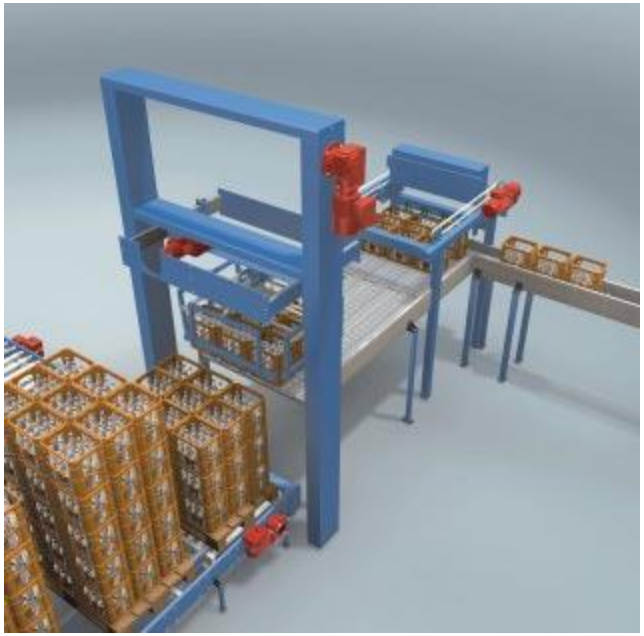
1.6

PLC Size and Application

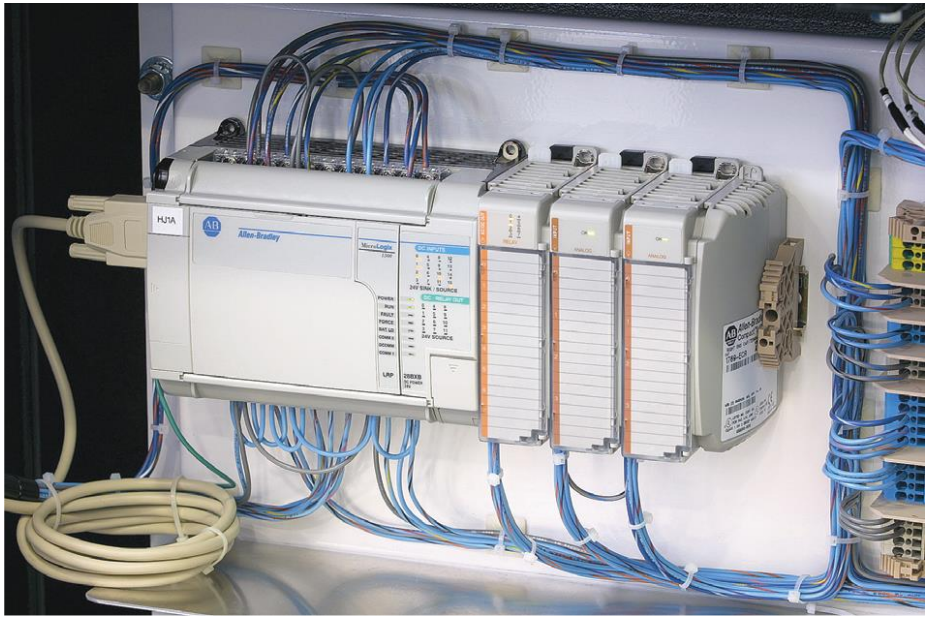
The criteria used in categorizing PLCs include functionality, number of inputs and outputs, cost, and physical size

The *I/O count* is the most important size factor. In general, the *nano* is the smallest size with less than 15 I/O points. This is followed by *micro* types (15 to 128 I/O points), *medium* types (128 to 512 I/O points), and *large* types (over 512 I/O points).

Matching the PLC with the application is a key factor in the selection process.



There are three major types of PLC applications: single ended, multitask, and control management.

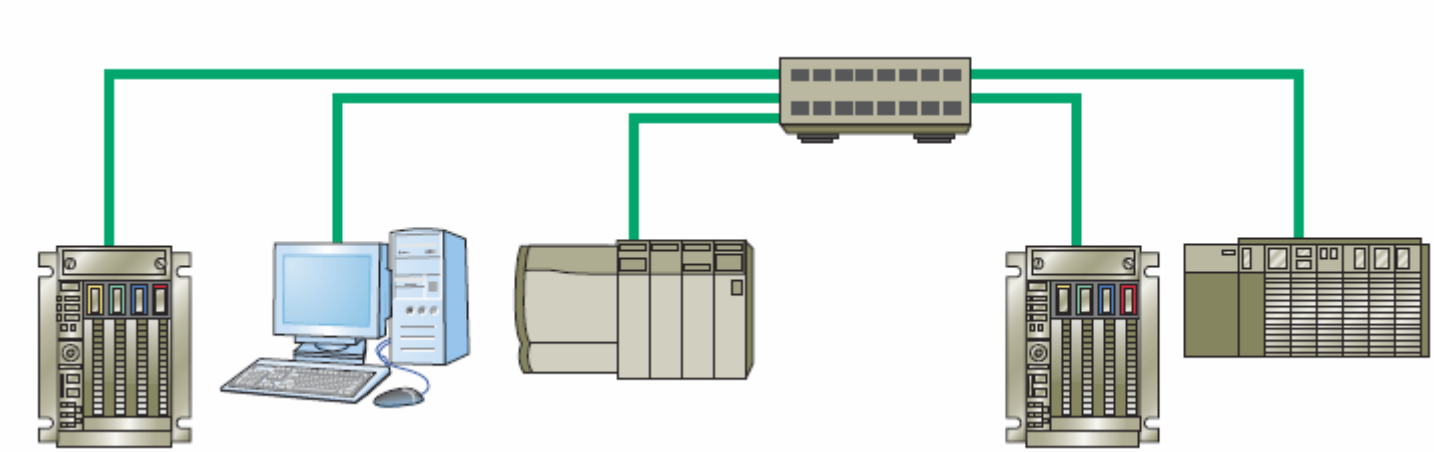


A *single ended* application involves one PLC controlling one process

A *multitask* PLC application involves one PLC controlling several processes.

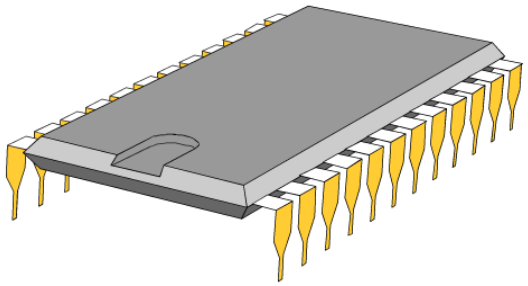
Adequate I/O capacity is a significant factor in this type of installation.

A *control management* PLC application involves one PLC controlling several others.



- This kind of application requires a large PLC processor designed to communicate with other PLCs and computers
- The control management PLC supervises several PLCs by downloading programs that tell the other PLCs what has to be done.

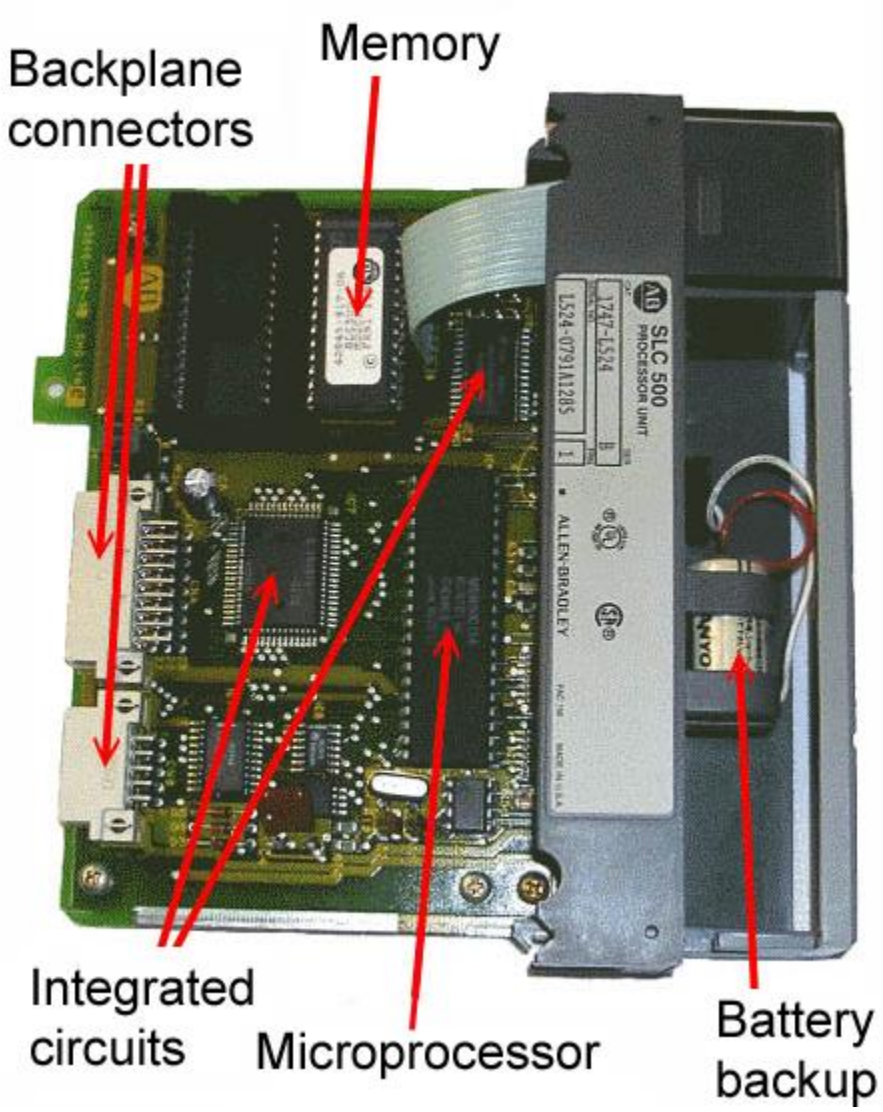
***Memory* is the part of a PLC controller that stores data, instructions, and the control program.**



- **Memory size is usually expressed in K values: 1 K, 6 K, 12 K, and so on.**
- **The measurement kilo, abbreviated K, normally refers to 1000 units.**
- **When dealing with computer or PLC memory, however, 1 K means 1024, because this measurement is based on the binary number system ($2^{10} = 1024$).**
- **Depending on memory type, 1 K can mean 1024 bits, 1024 bytes, or 1024 words.**

The amount of memory required depends on:

- Number of I/O points used
- Size of control program
- Data-collecting requirements
- Supervisory functions required
- Future expansion



The *instruction set* for a particular PLC lists the different types of instructions supported.

Table 1-1 Typical PLC Instructions

Instruction	Operation
XIC (Examine ON)	Examine a bit for an ON condition
XIO (Examine OFF)	Examine a bit for an OFF condition
OTE (Output Energize)	Turn ON a bit (nonretentive)
OTL (Output Latch)	Latch a bit (retentive)
OTU (Output Unlatch)	Unlatch a bit (retentive)
TOF (Timer Off-Delay)	Turn an output ON or OFF after its rung has been OFF for a preset time interval
TON (Timer On-Delay)	Turn an output ON or OFF after its rung has been ON for a preset time interval
CTD (Count Down)	Use a software counter to count down from a specified value
CTU (Count Up)	Use a software counter to count up to a specified value