CHAPTER 1

A SHORT HISTORY OF INDUSTRIAL AUTOMATION

Introduction

Industrial automation refers to the use of control systems, such as computers, robots, and information technologies, for handling various processes and machinery in an industry, replacing human intervention. The history of industrial automation is a dynamic narrative that spans from the early innovations of the 18th century to the advanced technologies shaping today's smart factories. It reflects the evolution of engineering, technological advances, and economic needs that have driven humanity towards increasingly efficient production methods.

1 Pre-Industrial Revolution: The Seeds of Automation

The foundations of industrial automation trace back to the pre-industrial period when early mechanization was primarily focused on agricultural and manual labor-saving devices. The use of water wheels, windmills, and basic tools laid the groundwork for mechanized processes that would be central to the later industrial revolution.

During the Renaissance, inventors such as Leonardo da Vinci conceptualized machines that could perform repetitive tasks, such as the first mechanical looms. Though most remained theoretical or rudimentary, these early ideas planted the seeds for future technological innovations.

2 Industrial Revolution: The Birth of Mechanization (1760-1840)

The Industrial Revolution in the late 18th century marked the first major shift towards industrial automation, primarily driven by mechanization. In this period, manual labor began to be replaced by machines powered by steam engines and later electricity. Notable inventions such as James Watt's steam engine (1765) and the spinning jenny (1770) revolutionized textile manufacturing and other sectors.

Key contributions during this period include:

- **The Jacquard Loom (1801):** Invented by Joseph Marie Jacquard, this loom used punched cards to control the weaving of complex patterns in textiles. It is one of the earliest examples of a programmable machine and is often cited as a precursor to modern-day computer programming.
- **Steam Power:** Steam engines were widely used in factories to power machines such as pumps, looms, and presses, enhancing productivity and reducing human labor.

This era was dominated by the transition from manual labor to machine-based production, setting the stage for more sophisticated control systems and automation in subsequent periods.

3 The Second Industrial Revolution: Electrification and Mass Production (1870-1914)

The second Industrial Revolution brought with it major innovations in electricity, which had a profound effect on automation. The development of electrical power enabled the creation of more advanced machinery, which was faster, more precise, and more reliable than steam-powered machines.

- **Henry Ford's Assembly Line (1913):** A pivotal moment in automation history, Ford introduced the assembly line in automobile manufacturing, drastically reducing production time. The assembly line relied on standardization and division of labor, where each worker performed specific tasks in a repetitive fashion. Though the process was manually driven, it laid the foundation for future automation in factories.
- **Electric Motors and Controllers:** The invention of electric motors enabled machines to run continuously and with greater precision. The introduction of basic electric control systems, such as relays and switches, allowed for early forms of automated machine operation.

4 The Post-War Era: The Dawn of Automation (1940s-1960s)

The mid-20th century marked the dawn of true industrial automation, driven by advancements in electronics, computation, and control theory. The development of transistors, solid-state electronics, and early computers during and after World War II accelerated the automation of manufacturing processes.

- **Numerical Control (NC):** The 1940s saw the rise of numerical control, where machines were controlled by inputting instructions via punched tape or cards. This allowed machines such as lathes and milling machines to perform precise, repetitive tasks without the need for constant human intervention.

- **CNC Machines (1950s-1960s):** Computer Numerical Control (CNC) machines replaced manual controls with computer-based systems, increasing the accuracy, speed, and flexibility of production processes. CNC technology was pivotal in industries such as automotive, aerospace, and precision manufacturing.
- **The First Industrial Robots (1961):** The introduction of the Unimate robot by George Devol and Joseph Engelberger marked the first major use of robotics in industrial settings. Deployed at General Motors, the Unimate could move and weld heavy parts, handling dangerous tasks typically performed by humans.

5 The Rise of Digital Automation and Programmable Logic Controllers (1960s-1980s)

The 1960s and 1970s witnessed the advent of digital technology, which revolutionized industrial automation. The use of microprocessors and software-enabled machines to be programmed for complex tasks with greater flexibility and precision.

- **Introduction of PLCs (1968):** The Programmable Logic Controller (PLC), invented by Dick Morley, was a revolutionary device that replaced hard-wired relay systems with a flexible, programmable system. PLCs allowed manufacturers to reprogram machinery quickly for different tasks, facilitating more efficient production lines. PLCs became a core element of industrial automation, widely adopted in industries from automotive to chemical plants.
- **Computer-Aided Design (CAD) and Computer-Aided Manufacturing (CAM):** The integration of digital systems into design and manufacturing enabled manufacturers to automate not only production but also the design process. CAD/CAM systems made it possible to go from design to production with minimal human intervention.
- **Rise of Flexible Manufacturing Systems (FMS):** During this period, there was a shift towards flexible automation, where machines could be reprogrammed to manufacture different products in smaller batches, as opposed to mass production systems that required dedicated machinery for a single product.

6 Automation and Robotics in the 21st Century: The Age of Smart Manufacturing (1990s-present)

The digital revolution of the late 20th century led to the era of smart manufacturing, where automation technologies are integrated with advanced computing, networking, and artificial intelligence to create more flexible, efficient, and self-regulating manufacturing systems.

- **Industrial Robots:** Robotics continues to evolve with the development of sophisticated robots capable of handling tasks with precision, flexibility, and speed. Collaborative robots (cobots), designed to work alongside human workers, have become common in factories, allowing humans to focus on more complex tasks while robots handle repetitive functions.

- **The Internet of Things (IoT):** The integration of IoT technologies into industrial settings has enabled machines and sensors to communicate and share data in real-time. This has paved the way for predictive maintenance, remote monitoring, and data-driven optimization of manufacturing processes.
- **Artificial Intelligence and Machine Learning:** AI is increasingly being integrated into industrial automation systems, allowing machines to learn from data, improve processes autonomously, and make intelligent decisions. AI-driven automation systems can analyze vast amounts of data to predict failures, optimize production, and reduce downtime.
- **Smart Factories and Industry 4.0:** The concept of Industry 4.0 refers to the ongoing automation of traditional manufacturing and industrial practices, using modern smart technology. This includes cyber-physical systems, cloud computing, big data, and IoT. Smart factories are highly automated and interconnected, with machines, systems, and people seamlessly communicating to optimize efficiency, quality, and flexibility.

7 The Future of Industrial Automation: Challenges and Opportunities

As technology continues to advance, industrial automation is expected to become even more sophisticated. Future trends include:

- **Autonomous Systems:** Fully autonomous factories where machines make all decisions without human input are a growing possibility with advancements in AI and machine learning.
- **Quantum Computing:** The integration of quantum computing could revolutionize industrial automation by solving complex optimization problems in real-time, leading to significant improvements in efficiency and productivity.
- **Sustainability:** Automation is increasingly being used to develop more sustainable manufacturing practices. Energy-efficient machines, smart grids, and automated systems that reduce waste will play a crucial role in addressing environmental challenges.

However, these advances also bring challenges such as the displacement of jobs, cybersecurity concerns, and the need for new regulations and standards to govern autonomous systems.

Conclusion

The history of industrial automation is a testament to humanity's relentless pursuit of efficiency and progress. From the mechanization of the Industrial Revolution to the dawn of the digital era and the advent of Industry 4.0, industrial automation has continuously evolved to meet the needs of society. As we look to the future, the integration of artificial intelligence, IoT, and advanced robotics promises to reshape industries in ways that were once unimaginable. While this progress brings challenges, it also offers tremendous opportunities for creating a more productive, sustainable, and innovative industrial landscape.

CHAPTER 2

INTRODUCTION TO INDUSTRIAL AUTOMATION

1 Production systems

The word manufacturing derives from two Latin words, *manus* (hand) and *factus* (make), so that the combination means made by hand. This was the way manufacturing was accomplished when the word first appeared in the English language around 1567. Commercial goods of those times were made by hand. The methods were handicraft, accomplished in small shops, and the goods were relatively simple, at least by today's standards. As many years passed, factories came into being, with many workers at a single site, and the work had to be organized using machines rather than handicraft techniques. The products became more complex, and so did the processes to make them. Workers had to specialize in their tasks. Rather than overseeing the fabrication of the entire product, they were responsible for only a small part of the total work. More up-front planning was required, and more coordination of the operations was needed to keep track of the work flow in the factories. Slowly but surely, the systems of production were being developed. A *production system* is a collection of people, equipment, and procedures organized to perform the manufacturing operations of a company. It consists of two major components as indicated in Figure 2.1

1. Facilities. The physical facilities of the production system include the equipment, the way the equipment is laid out, and the factory in which the equipment is located.

2. Manufacturing support systems. These are the procedures used by the company to manage production and to solve the technical and logistics problems encountered in ordering materials, moving the work through the factory, and ensuring that products meet quality standards. Product design and certain business functions are included in the manufacturing support systems.

In modern manufacturing operations, portions of the production system are automated and/or computerized. In addition, production systems include people. People make these systems work. In general, direct labor people (blue-collar workers) are responsible for operating the facilities, and professional staff people (white-collar workers) are responsible for the manufacturing support systems.

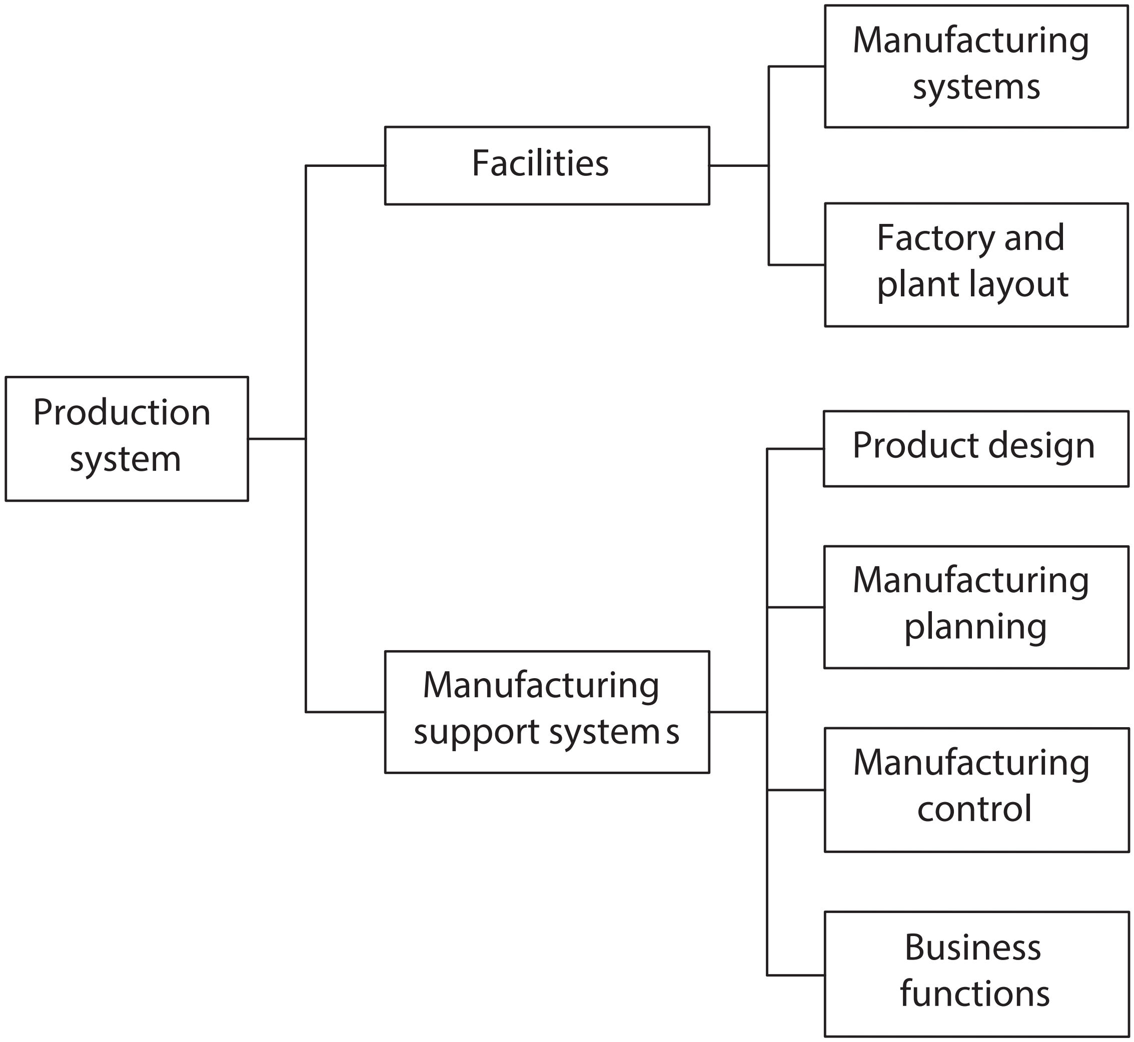


Figure 2.1: Production system as Facilities and Manufacturing support systems

1.1 Facilities

The facilities in the production system consist of the factory, production machines and tooling, material handling equipment, inspection equipment, and computer systems that control the manufacturing operations. Facilities also include the plant layout, which is the way the equipment is physically arranged in the factory. The equipment is usually organized into manufacturing systems, which are the logical groupings of equipment and workers that accomplish the processing and assembly operations on parts and products made by the factory. Manufacturing systems can be individual work cells consisting of a single production machine and a worker assigned to that machine. More complex manufacturing systems consist of collections of machines and workers, for example, a production line. The manufacturing systems come in direct physical contact with the parts and/or assemblies being made. They "touch" the product. In terms of human participation in the processes performed by the manufacturing systems, three basic categories can be distinguished, as portrayed in Figure : (a) manual work systems, (b) worker-machine systems, and (c) automated systems

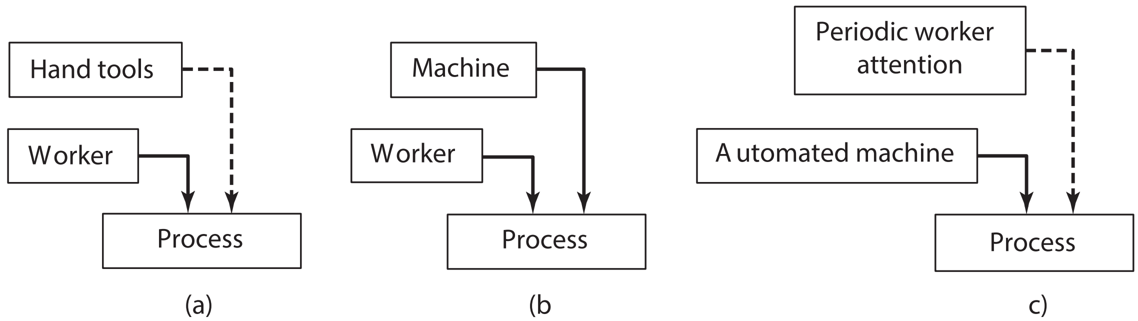


Figure 2.2: Manufacturing Systems Categories: (a) manual work systems, (b) worker-machine systems, and (c) automated systems

Manual work systems

A manual work system consists of one or more workers performing one or more tasks without the aid of powered tools. Manual material handling tasks are common activities in manual work systems. Production tasks commonly require the use of hand tools, such as screwdrivers and hammers. When using hand tools, a workholder is often employed to grasp the work part and position it securely for processing. Examples of production-related manual tasks involving the use of hand tools include

- A machinist using a file to round the edges of a rectangular part that has just been milled
- A quality control inspector using a micrometer to measure the diameter of a shaft
- A material handling worker using a dolly to move cartons in a warehouse
- A team of assembly workers putting together a piece of machinery using hand tools.

Worker-machine systems

In a worker-machine system, a human worker operates powered equipment, such as a machine tool or other production machine. This is one of the most widely used manufacturing systems. Worker-machine systems include combinations of one or more workers and one or more pieces of equipment. The workers and machines are combined to take advantage of their relative strengths and attributes, which are listed in Table 1.1. Examples of worker-machine systems include the following:

- A machinist operating an engine lathe to fabricate a part for a product
- A fitter and an industrial robot working together in an arc–welding work cell
- A crew of workers operating a rolling mill that converts hot steel slabs into flat plates
- A production line in which the products are moved by mechanized conveyor and the workers at some of the stations use power tools to accomplish their processing or assembly tasks.

Human	Machine
Sense unexpected stimuli	Perform repetitive tasks consistently
Develop new solutions to problems	Store large amounts of data
Cope with abstract problems	Retrieve data from memory reliably
Adapt to change	Perform multiple tasks simultaneously
Generalize from observations	Apply high forces and power
Learn from experience	Perform simple computations quickly
Make decisions based on incomplete data	Make routine decisions quickly

Table 2.1: Relative Strengths and Attributes of Humans and Machines

Automated Systems

An automated system is one in which a process is performed by a machine without the direct participation of a human worker. Automation is implemented using a program of instructions combined with a control system that executes the instructions. Power is required to drive the process and to operate the program and control system. There is not always a clear distinction between worker-machine systems and automated systems, because many worker-machine systems operate with some degree of automation. Two levels of automation can be identified: semi-automated and fully automated. A semi-automated machine performs a portion of the work cycle under some form of program control, and a human worker tends to the machine for the remainder of the cycle, by loading and unloading it, or by performing some other task each cycle. A fully automated machine is distinguished from its semi-automated counterpart by its capacity to operate for an extended period of time with no human attention. Extended period of time means longer than one work cycle; a worker is not required to be present during each cycle. Instead, the worker may need to tend the machine every tenth cycle, or every hundredth cycle. An example of this type of operation is found in many injection molding plants, where the molding machines run on automatic cycles, but periodically the molded parts at the machine must be collected by a worker. Figure 2.1-(c) depicts a fully automated system. The semi-automated system is best portrayed by Figure 2.1-(b). In certain fully automated processes, one or more workers are required to be present to continuously monitor the operation, and make sure that it performs according to the intended specifications. Examples of these kinds of automated processes include complex chemical processes, oil refineries, and nuclear power plants. The workers do not actively participate in the process except to make occasional adjustments in the equipment settings, perform periodic maintenance, and spring into action if something goes wrong.

2 Automation in Production Systems

Some components of a production system are likely to be automated, whereas others will be operated manually or clerically. The automated elements of the production system can be separated into two categories: (1) automation of the manufacturing systems in the factory, and (2) computerization of the manufacturing support systems. In modern production systems, the two categories are closely related, because the automated manufacturing systems on the factory floor are themselves usually implemented by computer systems that are integrated with the manufacturing support systems and management information system operating at the plant and enterprise levels. The two categories of automation are shown in Figure 2.3 as an overlay on Figure 2.1.

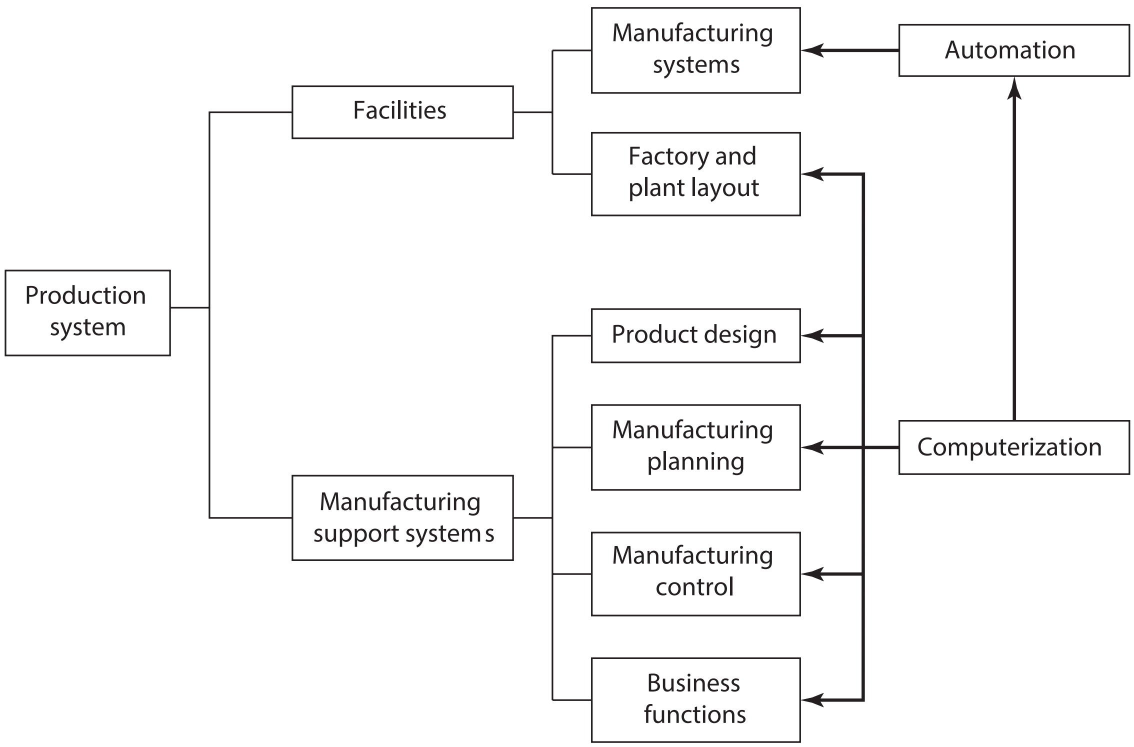


Figure 2.3: Opportunities for automation and computerization in a production system

2.1 Automated Manufacturing System

Automated manufacturing systems operate in the factory on the physical product. They perform operations such as processing, assembly, inspection, and material handling, in many cases accomplishing more than one of these operations in the same system. They are called automated because they perform their operations with a reduced level of human participation compared with the corresponding manual process. In some highly automated systems, there is virtually no human participation. Examples of automated manufacturing systems include:

- Automated machine tools that process parts
- Transfer lines that perform a series of machining operations
- Automated assembly systems
- Manufacturing systems that use industrial robots to perform processing or assembly operations

- Automatic material handling and storage systems to integrate manufacturing operations
- Automatic inspection systems for quality control.

Automated manufacturing systems can be classified into three basic types: (1) fixed automation, (2) programmable automation, and (3) flexible automation. They generally operate as fully automated systems although semi-automated systems are common in programmable automation. The relative positions of the three types of automation for different production volumes and product varieties are depicted in Figure 2.4

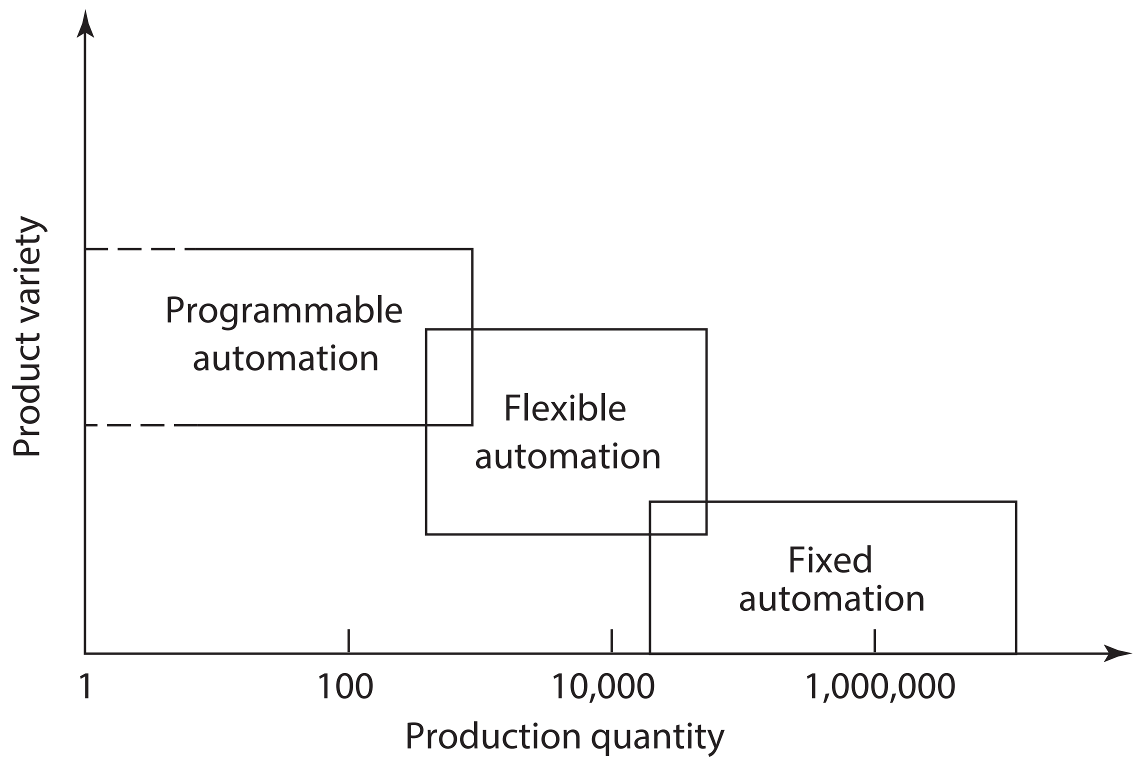


Figure 2.4: Three types of automation vs. the production quantity

Fixed Automation

Fixed automation is a system in which the sequence of processing (or assembly) operations is fixed by the equipment configuration. Each operation in the sequence is usually simple, involving perhaps a plain linear or rotational motion or an uncomplicated combination of the two, such as feeding a rotating spindle. It is the integration and coordination of many such operations in one piece of equipment that makes the system complex. Typical features of fixed automation are (1) high initial investment for custom-engineered equipment, (2) high production rates, and (3) inflexibility of the equipment to accommodate product variety. The economic justification for fixed automation is found in products that are made in very large quantities and at high production rates. The high initial cost of the equipment can be spread over a very large number of units, thus minimizing the unit cost relative to alternative methods of production. Examples of fixed automation include machining transfer lines and automated assembly machines.

Programmable Automation

In programmable automation, the production equipment is designed with the capability to change the sequence of operations to accommodate different product configurations. The operation sequence is controlled by a program, which is a set of instructions coded so that they can be read and interpreted by the system. New programs can be prepared and entered into the equipment to produce new products. Some of the features that characterize programmable automation include (1) high investment in general-purpose equipment, (2) lower production rates than fixed automation, (3) flexibility to deal with variations and changes in product configuration, and (4) high suitability for batch production. Programmable automated systems are used in low- and medium-volume production. The parts or products are typically made in batches. To produce each new batch of a different item, the system must be reprogrammed with the set of machine instructions that correspond to the new item. The physical setup of the machine must also be changed: Tools must be loaded, fixtures must be attached to the machine table, and any required machine settings must be entered. This changeover takes time. Consequently, the typical cycle for a given batch includes a period during which the setup and reprogramming take place, followed by a period in which the parts are produced. Examples of programmable automation include numerically controlled (NC) machine tools, industrial robots, and programmable logic controllers.

Flexible Automation

Flexible automation is an extension of programmable automation. A flexible automated system is capable of producing a variety of parts or products with virtually no time lost for changeovers from one design to the next. There is no lost production time while reprogramming the system and altering the physical setup (tooling, fixtures, machine settings). Accordingly, the system can produce various mixes and schedules of parts or products instead of requiring that they be made in batches. What makes flexible automation possible is that the differences between parts processed by the system are not significant, so the amount of changeover between designs is minimal. Features of flexible automation include (1) high investment for a custom-engineered system, (2) continuous production of variable mixtures of parts or products, (3) medium production rates, and (4) flexibility to deal with product design variations. Examples of flexible automation are flexible manufacturing systems that perform machining processes.

3 Computerized Manufacturing Support Systems

Automation of the manufacturing support systems is aimed at reducing the amount of manual and clerical effort in product design, manufacturing planning and control, and the business functions of the firm. Nearly all modern manufacturing support systems are implemented using computers. Indeed, computer technology is used to implement automation of the manufacturing systems in the factory as well. Computer-integrated manufacturing (CIM) denotes the pervasive use of computer systems to design the products, plan the production, control the operations, and perform the various information-processing functions needed in a manufacturing firm. True CIM involves integrating all of these functions in one system that operates throughout the enterprise. Other terms are used to identify specific elements of the CIM system; for example, computer-aided design (CAD) supports the product design function. Computer-aided manufacturing (CAM) is used for functions related to manufacturing engineering, such as process planning and numerical control part programming. Some computer systems perform both CAD and CAM, and so the term CAD/CAM is used to indicate the integration of the two into one system. Computer-integrated manufacturing involves the information-processing activities that provide the data and knowledge required to successfully produce the product. These activities are accomplished to implement the four basic manufacturing support functions identified earlier: (1) business functions, (2) product design, (3) manufacturing planning, and (4) manufacturing control.

3.1 Reasons to use Automation

Companies undertake projects in automation and computer-integrated manufacturing for good reasons, some of which are the following:

1. *Increase labor productivity.* Automating a manufacturing operation invariably increases production rate and labor productivity. This means greater output per hour of labor input.

2. *Reduce labor cost.* Increasing labor cost has been, and continues to be, the trend in the world's industrialized societies. Consequently, higher investment in automation has become economically justifiable to replace manual operations. Machines are increasingly being substituted for human labor to reduce unit product cost.

3. *Mitigate the effects of labor shortages.* There is a general shortage of labor in many advanced nations, and this has stimulated the development of automated operations as a substitute for labor.

4. *Reduce or eliminate routine manual and clerical tasks.* An argument can be put forth that there is social value in automating operations that are routine, boring, fatiguing, and possibly irksome. Automating such tasks improves the general level of working conditions.

5. *Improve worker safety.* Automating a given operation and transferring the worker from active participation in the process to a monitoring role, or removing the worker from the operation altogether, makes the work safer. The safety and physical well-being of the worker has become a national objective with the enactment of the Occupational Safety and Health Act (OSHA) in 1970. This has provided an impetus for automation.

6. *Improve product quality.* Automation not only results in higher production rates than manual operation, it also performs the manufacturing process with greater consistency and conformity to quality specifications.

7. *Reduce manufacturing lead time.* Automation helps reduce the elapsed time between customer order and product delivery, providing a competitive advantage to the manufacturer for future orders. By reducing manufacturing lead time, the manufacturer also reduces work-in-process inventory.

8. *Accomplish processes that cannot be done manually.* Certain operations cannot be accomplished without the aid of a machine. These processes require precision, miniaturization, or complexity of geometry that cannot be achieved manually. Examples include certain integrated circuit fabrication operations, rapid prototyping processes based on computer graphics (CAD) models, and the machining of complex, mathematically defined surfaces using computer numerical control. These processes can only be realized by computer-controlled systems.

9. *Avoid the high cost of not automating.* There is a significant competitive advantage gained in automating a manufacturing plant. The advantage cannot always be demonstrated on a company's project authorization form. The benefits of automation often show up in unexpected and intangible ways, such as in improved quality, higher sales, better labor relations, and better company image. Companies that do not automate are likely to find themselves at a competitive disadvantage with their customers, their employees, and the general public.

CHAPTER 3

AUTOMATION AND CONTROL

1 Basic Elements of an Automated System

Automation can be defined as the technology by which a process or procedure is accomplished without human assistance. It is implemented using a program of instructions combined with a control system that executes the instructions. To automate a process, power is required, both to drive the process itself and to operate the program and control system. Although automation is applied in a wide variety of areas, it is most closely associated with the manufacturing industries. It was in the context of manufacturing that the term was originally coined by an engineering manager at Ford Motor Company in 1946 to describe the variety of automatic transfer devices and feed mechanisms that had been installed in Ford's production plants (See Chapter 1). It is ironic that nearly all modern applications of automation are controlled by computer technologies that were not available in 1946.

An automated system consists of three basic elements: (1) power to accomplish the process and operate the system, (2) a program of instructions to direct the process, and (3) a control system to actuate the instructions. The relationship among these elements is illustrated in Figure 3.1. All systems that qualify as being automated include these three basic elements in one form or another. They are present in the three basic types of automated manufacturing systems: fixed automation, programmable automation, and flexible automation (See Previous Chapter).

1.1 Power to accomplish the automated process

An automated system is used to operate some process, and power is required to drive the process as well as the controls. The principal source of power in automated systems is electricity. Electric power has many advantages in automated as well as nonautomated processes:

- Electric power is widely available at moderate cost. It is an important part of the industrial infrastructure.

- Electric power can be readily converted to alternative energy forms: mechanical, thermal, light, acoustic, hydraulic, and pneumatic.

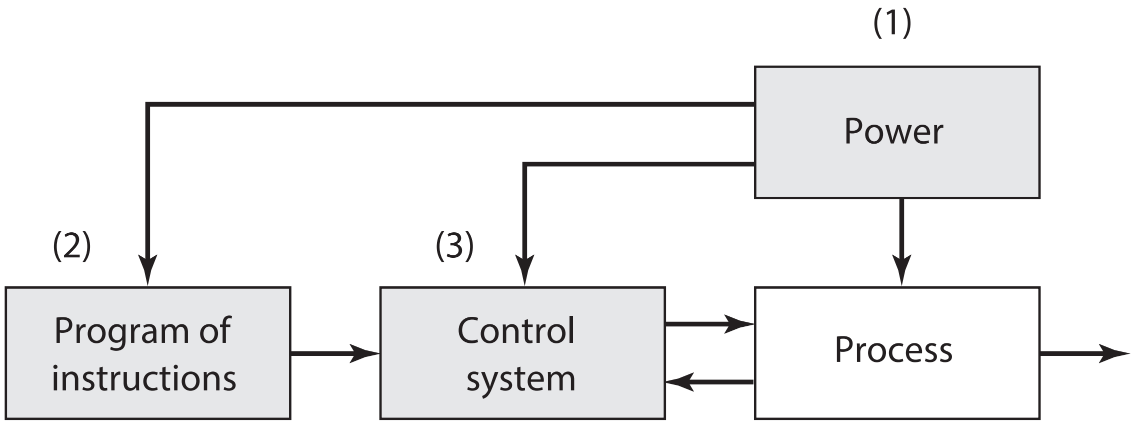


Figure 3.1: Elements of an automated system: (1) power, (2) program of instructions, and (3) control systems

- Electric power at low levels can be used to accomplish functions such as signal transmission, information processing, and data storage and communication.
- Electric energy can be stored in long-life batteries for use in locations where an external source of electrical power is not conveniently available.

Alternative power sources include fossil fuels, atomic, solar, water, and wind. However, their exclusive use is rare in automated systems. In many cases when alternative power sources are used to drive the process itself, electrical power is used for the controls that automate the operation. For example, in casting or heat treatment, the furnace may be heated by fossil fuels, but the control system to regulate temperature and time cycle is electrical. In other cases, the energy from these alternative sources is converted to electric power to operate both the process and its automation. When solar energy is used as a power source for an automated system, it is generally converted in this way.

Power for the Process.

In production, the term process refers to the manufacturing operation that is performed on a work unit. In Table 3.1, a list of common manufacturing processes is compiled along with the form of power required and the resulting action on the work unit. Most of the power in manufacturing plants is consumed by these kinds of operations. The "power form" indicated in the middle column of the table refers to the energy that is applied directly to the process.

Process	Power Form	Action Accomplished
Casting	Thermal	Melting the metal before pouring into a mold cavity where solidification occurs.
Electric discharge machining	Electrical	Metal removal is accomplished by a series of discrete electrical discharges between electrode (tool) and workpiece. The electric discharges cause very high localized temperatures that melt the metal.
Forging	Mechanical Metal work part is deformed by opposing dies.	Work parts are often heated in advance of deformation, thus thermal power is also required.
Heat-treating	Thermal	Metallic work unit is heated to temperature below melting point to effect microstructural changes.
Injection molding	Thermal and mechanical	Heat is used to raise temperature of polymer to highly plastic consistency, and mechanical force is used to inject the polymer melt into a mold cavity.
Laser beam cutting	Light and thermal	A highly coherent light beam is used to cut material by vaporization and melting.
Machining	Mechanical	Cutting of metal is accomplished by relative motion between tool and workpiece.
Sheet metal punching and blanking	Mechanical	Mechanical power is used to shear metal sheets and plates.
Welding	Thermal (maybe mechanical)	Most welding processes use heat to cause fusion and coalescence of two (or more) metal parts at their contacting surfaces. Some welding processes also apply mechanical pressure.

Table 3.1: Common Manufacturing Processes and Their Power Requirements

As indicated earlier, the power source for each operation is often converted from electricity. In addition to driving the manufacturing process itself, power is also required for the following material handling functions:

- Loading and unloading the work unit. All of the processes listed in Table 4.1 are accomplished on discrete parts. These parts must be moved into the proper position and orientation for the process to be performed, and power is required for this transport and placement function. At the conclusion of the process, the work unit must be removed. If the process is completely automated, then some form of mechanized power is used. If the process is manually operated or semiautomated, then human power may be used to position and locate the work unit.
- Material transport between operations. In addition to loading and unloading at a given operation, the work units must be moved between operations

Power for Automation.

Above and beyond the basic power requirements for the manufacturing operation, additional power is required for automation. The additional power is used for the following functions:

- *Controller unit.* Modern industrial controllers are based on digital computers, which require electrical power to read the program of instructions, perform the control calculations, and execute the instructions by transmitting the proper commands to actuating devices.
- *Power to actuate the control signals.* The commands sent by the controller unit are carried out by means of electromechanical devices, such as switches and motors, called actuators. The commands are generally transmitted by means of low-voltage control signals. To accomplish the commands, the actuators require more power, and so the control signals must be amplified to provide the proper power level for the actuating device.
- *Data acquisition and information processing.* In most control systems, data must be collected from the process and used as input to the control algorithms. In addition, for some processes, it is a legal requirement that records be kept of process performance and/or product quality. These data acquisition and record-keeping functions require power, although in modest amounts.

1.2 Program of instructions

The actions performed by an automated process are defined by a program of instructions. Whether the manufacturing operation involves low, medium, or high production, each part or product requires one or more processing steps that are unique to that part or product. These processing steps are performed during a work cycle. A new part is completed at the end of each work cycle (in some manufacturing operations, more than one part is produced during the work cycle: for example, a plastic injection molding operation may produce multiple parts each cycle using a multiple cavity mold). The particular processing steps for the work cycle are specified in a work cycle program, called part programs in numerical control. Other process control applications use different names for this type of program.

Work Cycle Programs.

In the simplest automated processes, the work cycle consists of essentially one step, which is to maintain a single process parameter at a defined level, for example, maintain the temperature of a furnace at a designated value for the duration of a heat-treatment cycle. (It is assumed that loading and unloading of the work units into and from the furnace is performed manually and is therefore not part of the automatic cycle, so technically this is not a fully automated process.) In this case, programming simply involves setting the temperature dial on the furnace. This type of program is **set-point control**, in which the set point is the value of the process parameter or desired value of the controlled variable in the process (furnace temperature in this example). A **process parameter** is an input to the process, such as the temperature dial setting, whereas a process variable is the corresponding output of the process, which is the actual temperature of the furnace. To change the program, the operator simply changes the dial setting. In an extension of this simple case, the one-step process is defined by more than one process parameter, for example, a furnace in which both temperature and atmosphere are controlled. Because of dynamics in the way the process operates, the process variable is not always equal to the process parameter. For example, if the temperature setting suddenly were to be increased or decreased, it would take time for the furnace temperature to reach the new set-point value. (This is getting into control system issues, which is the topic of the Section). Work cycle programs are usually much more complicated than in the furnace example described. Following are five categories of work cycle programs, arranged in approximate order of increasing complexity and allowing for more than one process parameter in the program:

- *Set-point control*, in which the process parameter value is constant during the work cycle (as in the furnace example).
- *Logic control*, in which the process parameter value depends on the values of other variables in the process.
- *Sequence control*, in which the value of the process parameter changes as a function of time. The process parameter values can be either discrete (a sequence of step values) or continuously variable.
- *Interactive program*, in which interaction occurs between a human operator and the control system during the work cycle.
- *Intelligent program*, in which the control system exhibits aspects of human intelligence (e.g., logic, decision making, cognition, learning) as a result of the work cycle program. Most processes involve a work cycle consisting of multiple steps that are repeated with no deviation from one cycle to the next.

Most discrete part manufacturing operations are in this category. A typical sequence of steps (simplified) is the following: (1) load the part into the production machine, (2) perform the process, and (3) unload the part. During each step, there are one or more activities that involve changes in one or more process parameters.

Example 1.1. Consider an automated turning operation that generates a cone-shaped product. The system is automated and a robot loads and unloads the work units. The work cycle consists of the following steps: (1) load starting work-piece, (2) position cutting tool prior to turning, (3) turn, (4) reposition tool to a safe location at end of turning, and (5) unload finished work-piece. Identify the activities and process parameters for each step of the operation.
Solution - In step (1), the activities consist of the robot manipulator reaching for the raw work part, lifting and positioning the part into the chuck jaws of the lathe, then retreating to a safe position to await unloading. The process parameters for these activities are the axis values of the robot manipulator (which change continuously), the gripper value (open or closed), and the chuck jaw value (open or closed). In step (2), the activity is the movement of the cutting tool to a "ready" position. The process parameters associated with this activity are the x- and z-axis position of the tool. Step (3) is the turning operation. It requires the simultaneous control of three process parameters: rotational speed of the work-piece (rev/min), feed (mm/rev), and radial distance of the cutting tool from the axis of rotation. To cut the conical shape, radial distance must be changed continuously at a constant rate for each revolution of the work-piece. For a consistent finish on the surface, the rotational speed must be continuously adjusted to maintain a constant surface speed (m/min); and for equal feed marks on the surface, the feed must be set at a constant value. Depending on the angle of the cone, multiple turning passes may be required to gradually generate the desired contour. Each pass represents an additional step in the sequence. Steps (4) and (5) are the reverse of steps (2) and (1), respectively, and the process parameters are the same.

Many production operations consist of multiple steps, sometimes more complicated than in the turning example. Examples of these operations include automatic screw machine cycles, sheet metal stamping, plastic injection molding, and die casting. Each of these manufacturing processes has been used for many decades. In earlier versions of these operations, work cycles were controlled by hardware components, such as limit switches, timers, cams, and electromechanical relays. In effect, the assemblage of hardware components served as the program of instructions that directed the sequence of steps in the processing cycle. Although these devices were quite adequate in performing their logic and sequencing functions, they suffered from the following disadvantages: (1) They often required considerable time to design and fabricate, forcing the production equipment to be used for batch production only (Figure 3.2); (2) making even minor changes in the program was difficult and time consuming; and (3) the program was in a physical form that was not readily compatible with computer data processing and communication. Modern controllers used in automated systems are based on digital computers. Instead of cams, timers, relays, and other hardware components, the programs for computer-controlled equipment are contained in computer memory, internet cloud servers and other modern storage technologies. Virtually all modern production equipment is designed with some form of computer controller to execute its respective processing cycles. The use of digital computers as the process controller allows improvements and upgrades to be made in the control programs, such as the addition of control functions not foreseen during initial equipment design. These kinds of control changes are often difficult to make with the hardware components mentioned earlier. A work cycle may include manual steps, in which

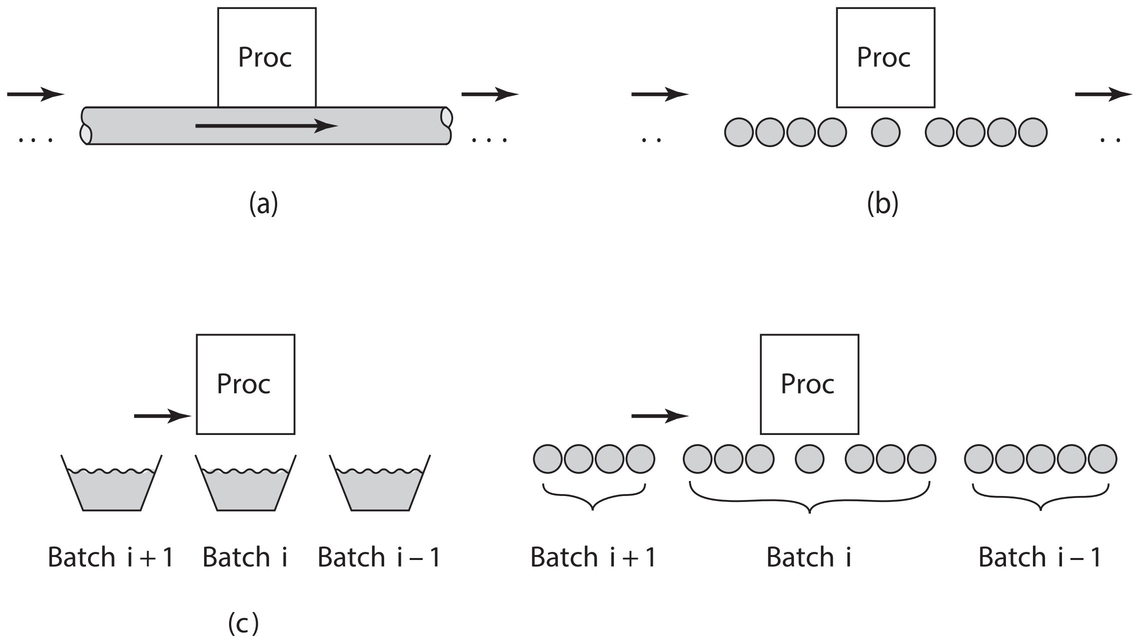


Figure 3.2: Continuous and batch production in the process and discrete manufacturing industries, including (a) continuous production in the process industries, (b) continuous production in the discrete manufacturing industries, (c) batch production in the process industries, and (d) batch production in the discrete manufacturing industries.

the operator performs certain activities during the work cycle, and the automated system performs the rest. These are referred to as semi-automated work cycles. A common example is the loading and unloading of parts by an operator into and from a numerical control machine between machining cycles, while the machine performs the cutting operation under part program control. Initiation of the cutting operation in each cycle is triggered by the operator activating a "start" button after the part has been loaded. Decision Making in the Programmed Work Cycle. In this example, the only two features of the work cycle were (1) the number and sequence of processing steps and (2) the process parameter changes in each step. Each work cycle consisted of the same steps and associated process parameter changes with no variation from one cycle to the next. The program of instructions is repeated each work cycle without deviation. In fact, many automated manufacturing operations require decisions to be made during the programmed work cycle to cope with variations in the cycle. In many cases, the variations are routine elements of the cycle, and the corresponding instructions for dealing with them are incorporated into the regular part program. These cases include:

- Operator interaction. Although the program of instructions is intended to be carried out without human interaction, the controller unit may require input data from a human operator in order to function. For example, in an automated engraving operation, the operator may have to enter the alphanumeric characters that are to be engraved on the work unit (e.g., plaque, trophy, belt buckle). After the characters are entered, the system accomplishes the engraving automatically. (An everyday example of operator interaction with an automated system is a bank customer using an automated teller machine. The customer must enter the codes indicating what transaction the teller machine must accomplish.)

- Different part or product styles processed by the system. In this instance, the automated system is programmed to perform different work cycles on different part or product styles. An example is an industrial robot that performs a series of spot welding operations on car bodies in a final assembly plant. These plants are often designed to build different body styles on the same automated assembly line, such as two-door and four-door sedans. As each car body enters a given welding station on the line, sensors identify which style it is, and the robot performs the correct series of welds for that style.

- Variations in the starting work units. In some manufacturing operations, the starting work units are not consistent. A good example is a sand casting as the starting work unit in a machining operation. The dimensional variations in the raw castings sometimes necessitate an extra machining pass to bring the machined dimension to the specified value. The part program must be coded to allow for the additional pass when necessary.

In all of these examples, the routine variations can be accommodated in the regular work cycle program. The program can be designed to respond to sensor or operator inputs by executing the appropriate subroutine corresponding to the input. In other cases, the variations in the work cycle are not routine at all. They are infrequent and unexpected, such as the failure of an equipment component. In these instances, the program must include contingency procedures or modifications in the sequence to cope with conditions that lie outside the normal routine. Various production situations and work cycle programs have been discussed here. The following summarizes the features of work cycle programs (part programs) used to direct the operations of an automated system:

- Process parameters. How many process parameters must be controlled during each step? Are the process parameters continuous or discrete? Do they change during the step, for example, a positioning system whose axis values change during the processing step?

- Number of steps in work cycle. How many distinct steps or work elements are included in the work cycle? A general sequence in discrete production operations is (1) load, (2), process, (3) unload, but the process may include multiple steps.

- Manual participation in the work cycle. Is a human worker required to perform certain steps in the work cycle, such as loading and unloading a production machine, or is the work cycle fully automated?

- Operator interaction. For example, is the operator required to enter processing data for each work cycle?

- Variations in part or product styles. Are the work units identical each cycle, as in mass production (fixed automation) or batch production (programmable automation), or are different part or product styles processed each cycle (flexible automation)?

- Variations in starting work units. Variations can occur in starting dimensions or materials. If the variations are significant, some adjustments may be required during the work cycle.

1.3 Control System

The control element of the automated system executes the program of instructions. The control system causes the process to accomplish its defined function, which is to perform some manufacturing operation. A brief introduction to control systems is provided here. The following chapter describes this technology in more detail. The controls in an automated system can be either closed loop or open loop. A closed loop control system, also known as a feedback control system, is one in which the output variable is compared with an input parameter, and any difference between the two is used to drive the output into agreement with the input. As shown in Figure 3.3, a closed-loop control system consists of six basic

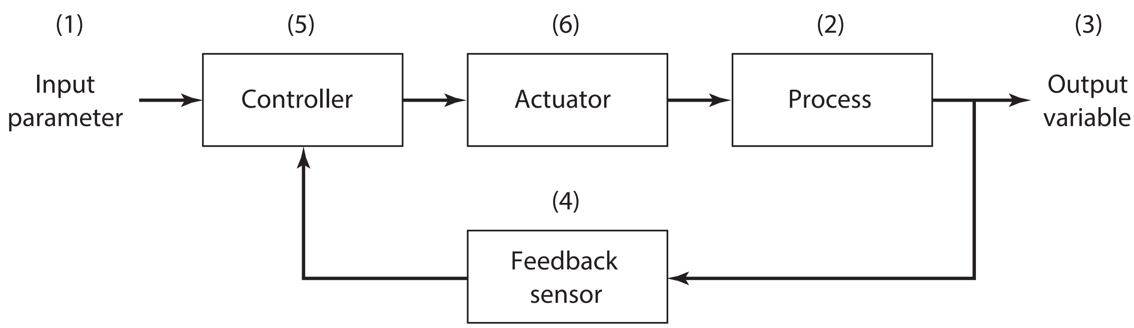


Figure 3.3: Closed loop control

elements: (1) input parameter, (2) process, (3) output variable, (4) feedback sensor, (5) controller, and (6) actuator. The input parameter (i.e., set point) represents the desired value of the output. In a home temperature control system, the set point is the desired thermostat setting. The process is the operation or function being controlled. In particular, it is the output variable that is being controlled in the loop. In the present discussion, the process of interest is usually a manufacturing operation, and the output variable is some process variable, perhaps a critical performance measure in the process, such as temperature or force or flow rate. A sensor is used to measure the output variable and close the loop between input and output. Sensors perform the feedback function in a closed-loop control system. The controller compares the output with the input and makes the required adjustment in the process to reduce the difference between them. The adjustment is accomplished using one or more actuators, which are the hardware devices that physically carry out the control actions, such as electric motors or flow valves. It should be mentioned that Figure 4.3 shows only one loop. Most industrial processes require multiple loops, one for each process variable that must be controlled. In contrast to a closed-loop control system, an open-loop control system operates without the feedback loop, as in Figure 3.4. In this case, the controls operate without measuring the output variable, so no comparison is made between the actual value of the output and the desired input parameter. The controller relies on an accurate model of

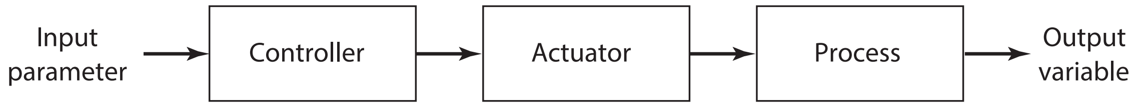


Figure 3.4: Open-loop control

the effect of its actuator on the process variable. With an open-loop system, there is always the risk that the actuator will not have the intended effect on the process, and that is the disadvantage of an open-loop system. Its advantage is that it is generally simpler and less expensive than a closed-loop system. Open-loop systems are usually appropriate when the following conditions apply: (1) the actions performed by the control system are simple, (2) the actuating function is very reliable, and (3) any reaction forces opposing the actuator are small enough to have no effect on the actuation. If these characteristics are not applicable, then a closed-loop control system may be more appropriate. Consider the difference between a closed-loop and open-loop system for the case of a positioning system. Positioning systems are common in manufacturing to locate a work part relative to a tool or work head. Figure 3.5 illustrates the case of a closed-loop positioning system. In operation, the system is directed

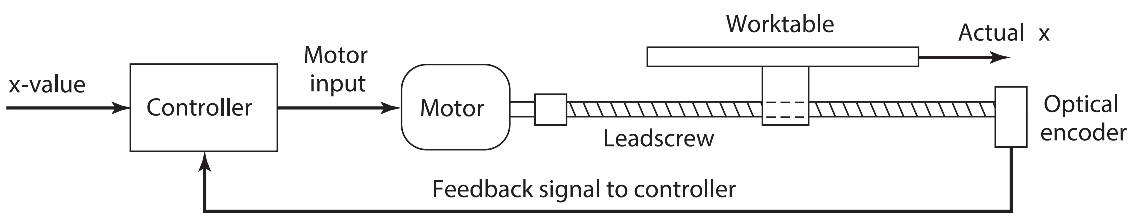


Figure 3.5: Positioning System

to move the worktable to a specified location as defined by a coordinate value in a Cartesian (or other) coordinate system. Most positioning systems have at least two axes (e.g., an x–y positioning table) with a control system for each axis, but the diagram only illustrates one of these axes. A dc servomotor connected to a leadscrew is a common actuator for each axis. A signal indicating the coordinate value (e.g., x-value) is sent from the controller to the motor that drives the leadscrew, whose rotation is converted into linear motion of the positioning table. The actual x-position is measured by a feedback sensor (e.g., an optical encoder). As the table moves closer to the desired x-coordinate value, the difference between the actual x-position and the input x-value decreases. The controller continues to drive the motor until the actual table position corresponds to the input position value. For the open-loop case, the diagram for the positioning system would be similar to the preceding, except that no feedback loop is present and a stepper motor would be used in place of the dc servomotor. A stepper motor is designed to rotate a precise fraction of a turn for each pulse received from the controller. Since the motor shaft is connected to the leadscrew, and the leadscrew drives the worktable, each pulse converts into a small constant linear movement of the table. To move the table a desired distance, the number of pulses corresponding to that distance is sent to the motor. Given the proper application, whose characteristics match the preceding list of operating conditions, an open-loop positioning system works with high reliability.

1.4 Levels of Automation

Automated systems can be applied to various levels of factory operations. One normally associates automation with the individual production machines. However, the production machine itself is made up of subsystems that may themselves be automated. For example, think about a computer numerical control (CNC) system. A modern CNC machine tool is a highly automated system that is composed of multiple control systems. Any CNC machine

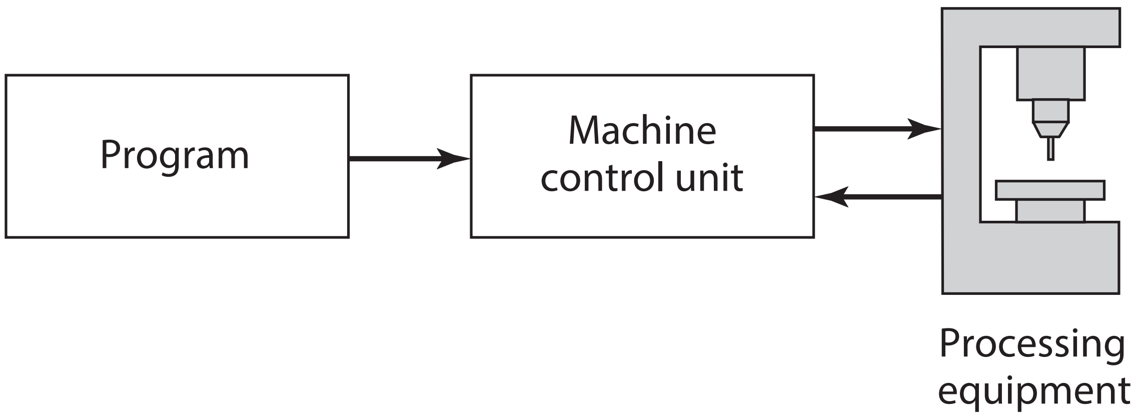


Figure 3.6: CNC Scheme

has at least two axes of motion, and some machines have more than five axes. Each of these axes operates as a positioning system, as previously seen, and is, in effect, an automated system. Similarly, a CNC machine is often part of a larger manufacturing system, and the larger system may be automated. For example, two or three machine tools may be connected by an automated part handling system operating under computer control. The machine tools also receive instructions (e.g., part programs) from the computer. Thus three levels of automation and control are included here (the positioning system level, the machine tool level, and the manufacturing system level). For the purposes of this text, five levels of automation can be identified, and their hierarchy is depicted in Figure 3.7:

1. *Device level.* This is the lowest level in the automation hierarchy. It includes the actuators, sensors, and other hardware components that comprise the machine level. The devices are combined into the individual control loops of the machine, for example, the feedback control loop for one axis of a CNC machine or one joint of an industrial robot.

2. *Machine level.* Hardware at the device level is assembled into individual machines. Examples include CNC machine tools and similar production equipment, industrial robots, powered conveyors, and automated guided vehicles. Control functions at this level include performing the sequence of steps in the program of instructions in the correct order and making sure that each step is properly executed.

3. *Cell or system level.* This is the manufacturing cell or system level, which operates under instructions from the plant level. A manufacturing cell or system is a group of machines or workstations connected and supported by a material handling system,

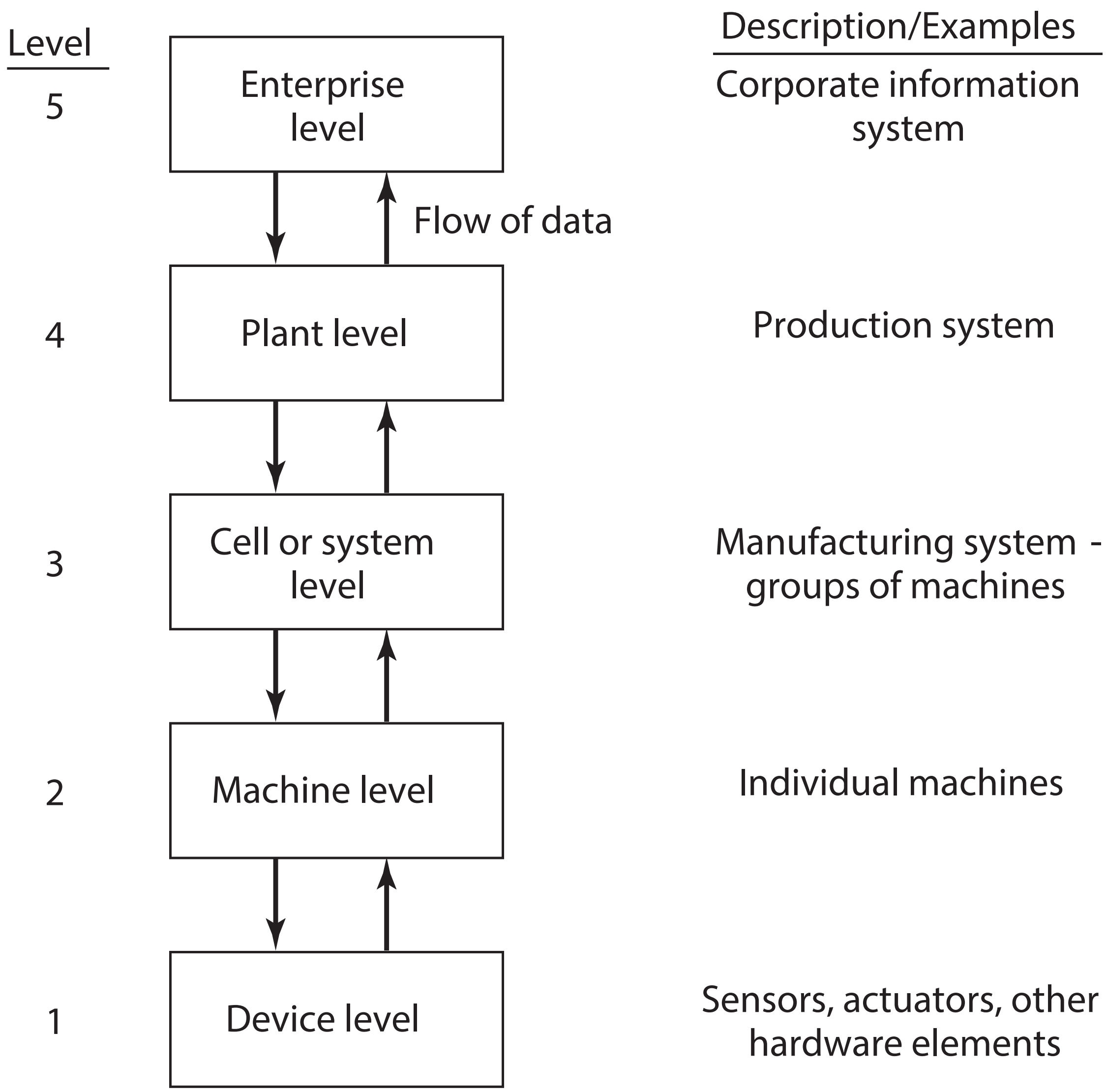


Figure 3.7: Levels of Automation

computer, and other equipment appropriate to the manufacturing process. Production lines are included in this level. Functions include part dispatching and machine loading, coordination among machines and material handling system, and collecting and evaluating inspection data.

4. *Plant level.* This is the factory or production systems level. It receives instructions from the corporate information system and translates them into operational plans for production. Likely functions include order processing, process planning, inventory control, purchasing, material requirements planning, shop floor control, and quality control.

5. *Enterprise level.* This is the highest level, consisting of the corporate information system. It is concerned with all of the functions necessary to manage the company: marketing and sales, accounting, design, research, aggregate planning, and master production scheduling.

2 Industrial Control Systems

The control system is one of the three basic components of an automated system. Here we will focus in particular on industrial control systems, in particular how digital computers are used to implement the control function in production.

2.1 How industrial plant are *viewed*

Industrial control is defined here as the automatic regulation of unit operations and their associated equipment, as well as the integration and coordination of the unit operations in the larger production system. More in detail, every industrial production process consists of a series of simple or complicated machines that, through the combination of raw materials, undergo a sequential transformation and integration in order to produce a final product. The term "machine" denotes every kind of electromechanical device on the industrial floor, e.g., from a simple motor (such as a drilling or a cutting machine) up to a complicated chemical machine (e.g., a chemical combustion machine). The whole set of machines (namely non-homogeneous machines), which are being integrated and combined in an industrial production process, will be referred to as an "integrated machine". As an example of an integrated machine, Figure 3.8 depicts the typical production line of an integrated paper machine, where the initial raw pulp is undergoing the sequential processes of pretreatment and grinding, refining, pulp bleaching, and pulp pressing and drying, until it is transformed into the final paper of predefined quality. Figure 3.9 shows the various stages of the papermaking process.

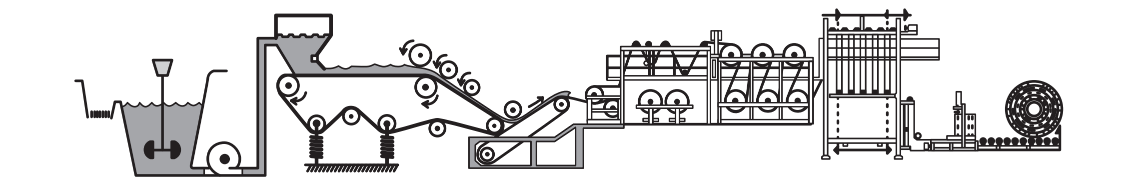


Figure 3.8: Schematic of a pulp-and-paper industrial process

During the pre-treat and grinding, in the first stage of the papermaking process, debarked and washed wood logs are preheated in order to become easier to grind and are inserted into large wood log grinders, which produce wood chips. Refining is the second stage of the paper manufacturing process, when the quality of the final product is highly dependent on that specific subprocess. During that stage, the wood chips are being received and transformed into pulp via high energy consumption, water infusion, and addition of chemical compounds. During the next stage of pulp bleaching, the pulp produced by the refining system is fed to the machine that is responsible for the discoloration of the mixture. Bleaching is a chemical process applied to cellulosic materials in order to increase their brightness. The last stage of the paper manufacturing process is the drying and pressing process. During this stage, bleached pulp is dried and pressed in order to form the desired production paper. In the case of an integrated machine, the whole sequence of operations for all the involved machines, the exact transformations and integrations of the raw materials, as well as the overall operational requirements, are a priori detailed and clearly defined for the industrial automation

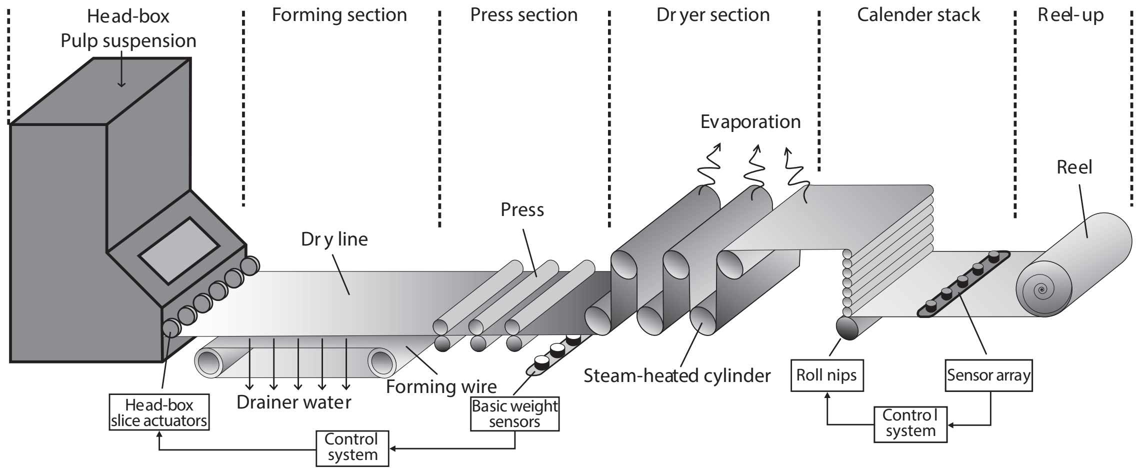


Figure 3.9: Simplified version of the pulp-processing stage

engineer, who is in charge of designing and implementing the desired process automation. For a specific production line, the sequence of operations and transformations, applied to the products, are generated from the production process itself and it is not possible, due to simplifications, these process stages to be altered. For example, in the case of an integrated machine of producing biscuits, it has been already defined from the process of production (the total manual and human-based processes) that in the mixture chamber, first the milk should be inserted and in a certain quantity, while in the sequence, the flour should be inserted at specific feeding rates and quantities. In this example, it is not possible, in order to simplify the overall automation process, to override this procedure by either designing an automation system that will either inverse the previous sequence of operations (e.g., first the flour will be inserted and then the milk) or completely ignore the predescribed sequence by allowing both materials to be inserted at the same time in the mixture chamber. Overall, and for all the produced industrial automations, the automated procedure should always satisfy the rules and sequences of the manual produced product, independently of the related complexity in the automation solution. Before analyzing the procedures needed for automating an industrial production line, it is of paramount importance to initially define in detail the various components that the automation and their specific functionalities and properties consist of. In an industrial production line, the "movement" is the fundamental and generalized characteristic of the overall process, since it is impossible to consider an industrial process without the existence of a linear, circular, or any other form of movement. Even in the case of a chemical reaction, where the existence of motion is not obvious, the movement also exists in this case and more specifically in the form of an electrical valve control, which opens in order to supply the reactor with the necessary amount of the reacted components. Furthermore, the existence of the need for movement is significantly evident, either in the cases where the product should be transferred to the various process points of the production line, or in the cases of integrated machines, where parts of the machines should be moved in order to produce the desired processing of the developing product. The machines that can be exploited for the creation of the movement can be categorized into two large categories, as displayed in Figure 3.10 The first category includes the different types of motors, independently of the

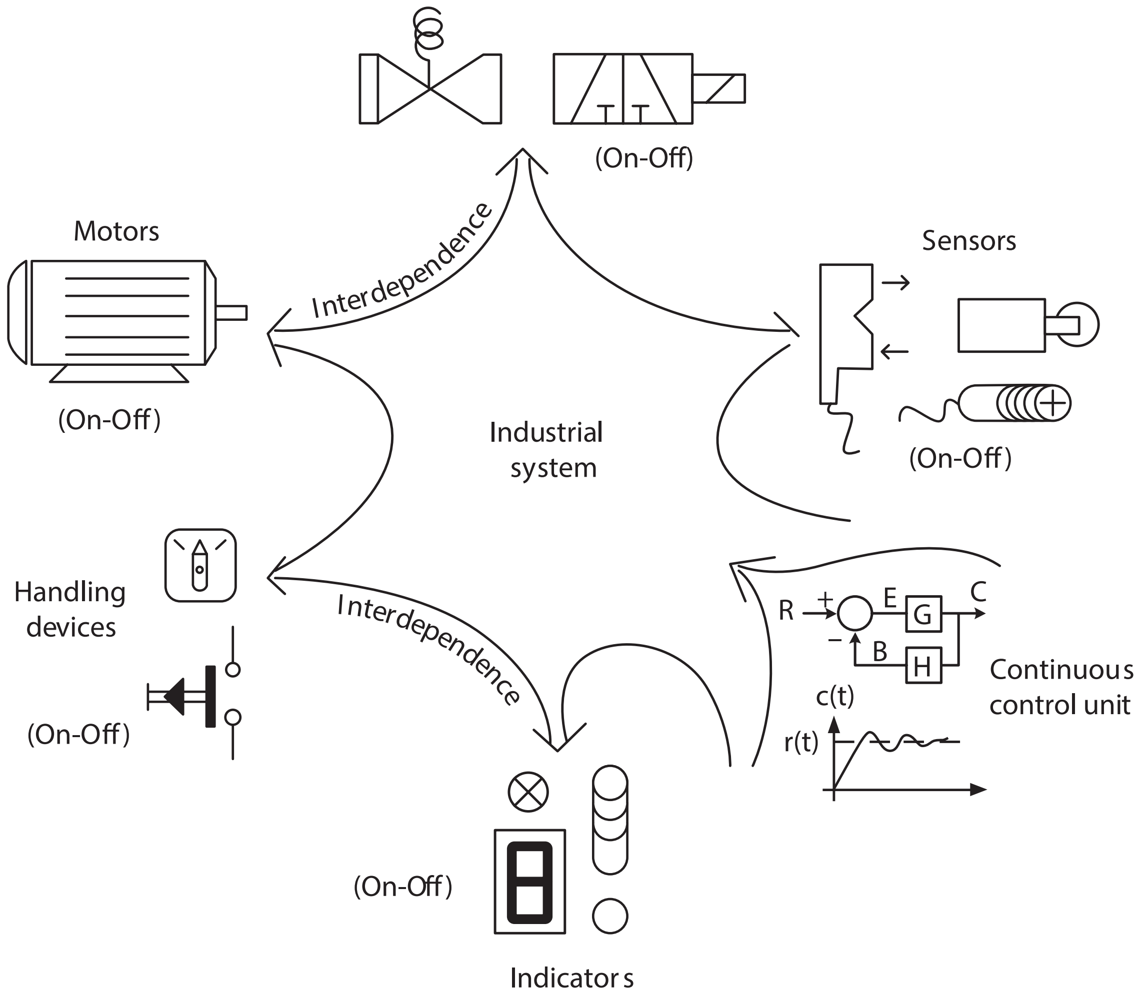


Figure 3.10: Basic kinds of industrial-type equipment composing an "industrial system"

operating principle (e.g., one-phase motors, three-phase motors, motors with short-circuit rotors, motors with direct start, motors that start in a Y/Δ mode, etc.) that creates a primary rotational movement, which can be further transformed by the use of appropriate mechanisms in a linear or other type of movements. The second category includes all the actuators, where a linear movement is created as the result of the attraction generated by an electromagnet (coil) on a ferromagnetic core, such as the various forms of electro-vanes, electrovalves, etc. The common characteristic of motors and actuators is the fact that they have only two possible states of operation. For expressing these states, usually we refer to them as "the motor is in operation", "the motor is not operating", "the valve is energized", "the valve is not energized", "the coil is under voltage", and "the coil is not under voltage". In general, there are two states of operation that can be defined as the ON and the OFF operation, which can be further associated directly with the digital logic symbols of 1 and 0. If one motor has, for example, two rotation directions or two rotational speeds and thus two states of operation, ON1 and ON2, then this consideration is not in conflict with the previous association. Actually, it can be considered as the case of having two motors, where one motor has the two states OFF-ON1 and the other one has the states OFF-ON2. The operation of the two motors, and more specifically the supply of the motors with the required electrical power, is achieved by the power relays that also have two states of operation, the ON and the OFF state. The control of the motors is achieved through the proper control of the relays, and thus the desired control system is applied on the corresponding relays controlling the electrical supply to the motors and it is not applied directly on the motors. After the definition of the control action being applied directly on the relays, the previous situation with the existence of multiple ON states for a machine will be explained more, through the following example. A three-phase motor is being considered with two directions of rotation. For the operation of the motor, two power relays are needed, which will be denoted by C_1 and C_2, as shown in Figure 3.11 When the C_1 relay is energized (relay C_2 is not energized),

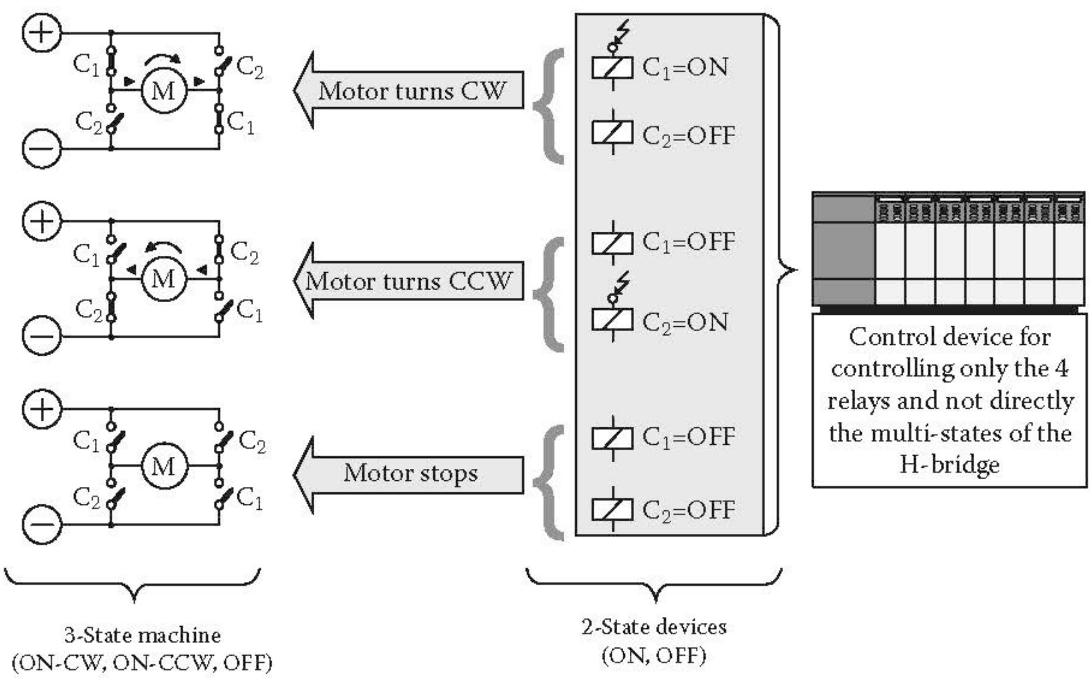


Figure 3.11: Multi-state electric motors are controlled by two state power relays.

the motor's coil ends are connected to the phases R, S, and T of the power network and thus the motor has a certain direction of rotation. When the power relay C_2 is energized (relay C_1 is not energized), the same motor's coil ends are connected to the phases T, S, and R of the power network and thus the motor has the opposite direction of rotation. As has been explained before, due to the fact that the control system is being applied on the power relays, the two states ON1 and ON2, of the same motor, correspond to the states of ON and OFF of two different devices, which are the power relays C_1 and C_2. As a result, the control system, instead of the states OFF-ON1-ON2 of a motor, with two directions of rotation, is being equivalently applied on the OFF-ON states of two different power relays.

To control the operation of an integrated machine, a set of specific operation control devices needs to be incorporated in the overall automation, like a simple push button, a rotational selector switch (knob), etc. In the case that the operation of the integrated machine is set in the "manual" mode, the operator is exploiting the operation devices for turning on the desired motors or the actuators and in the proper sequence. In the case that the integrated machine is set in the "auto" mode, the operator is again exploiting the operation devices, either for initiating the operation mode, or for instructing the integrating machine to change the operational state. As an example, in an integrating machine for chocolate production, the operator is capable, by the press of a button, to order the control system to alter the current recipe production for another one. In this case, the control system should allow the integrated machine to complete the current operation and afterwards, ensuring the prerequisite quantities for executing the ordered recipe change, to command the integrated machine in executing it. In most cases, the automation system of an integrated machine provides both the functionalities of an automatic or manual mode of operation, especially for dealing with the emergency fault situations, where direct manual control of all the provided automatic functionalities of the integrated machine is needed.

The operation control devices have also two states of operation, OFF and ON, similar to the cases of the motors and the actuators. As presented in Figure 3.12 a pressed button is

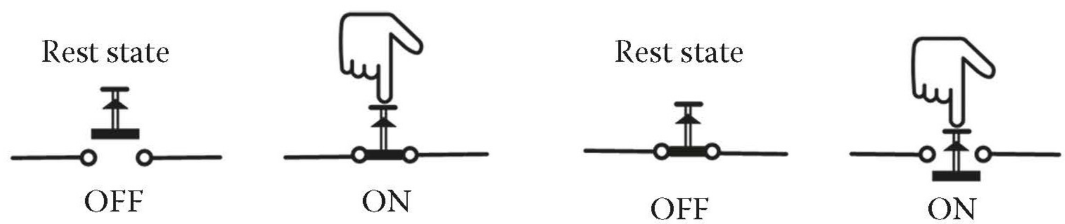


Figure 3.12: Multi-state electric motors are controlled by two state power relays.

energized and thus is in the ON state, while a non-pressed button is not energized and is in the OFF state. The ON state is independent of the time duration that the button is pressed and of the switching contact type (open or closed) in the not-energized state.

The next category of the machines presented in Figure 3.10 are very commonly installed in industrial environments, and are the indication devices that are exploited for transmitting process information from the integrated machine to the operator. In most cases, the industrial process is widely geographically distributed, and thus the operator that is in charge of the whole process has no direct visual or audio feedback from the process and the overall operation of the multiple integrated machines. However, even in the case that a visual or audio feedback is available, for safety reasons human senses are considered unreliable, and these monitoring, displaying, and visualizing devices are still needed to track the performance and state of operation of the industrial process. Especially in the cases of measuring variables without a direct visual or audio effect (like the variables of pressure, temperature, flow of a liquid in a non-transparent tube, etc.), such monitoring devices are of paramount importance. In most cases, this information is transmitted to the operators through the use of light or audio indicators, which can be again considered as devices with an ON and OFF state.

In a large set of integrated machines, specialized sensors for performing specific measuring of quantities are exploited extensively. For example, such sensors can be exploited to sense if there is a flow of a liquid in an opaque tube, if the level of a tank has reached a certain height, if the moving part of a machine has reached the desired place, if the temperature of a reactor has been set to the nominal one, etc. In general, these sensors can be categorized in digital and analog sensors. The digital sensors are characterized by two states of operation, namely ON and OFF or 1 and 0, correspondingly. The analog sensors are able to produce an analog (continuous) measurement of the quantity under study and thus more complicated hardware and software is needed to incorporate the industrial automation for exploiting this information.

The operation of an industrial automation device in an industrial process is not independent of the operation of the rest of the devices in the same process. In all cases, there is a strong dependence among all the utilized devices in the automation, as is also highlighted in the following example. Consider an industrial process of adding color to textile products like fibers or yarns, called "dyeing", where (as shown in Figure 3.13) it is desired to have the

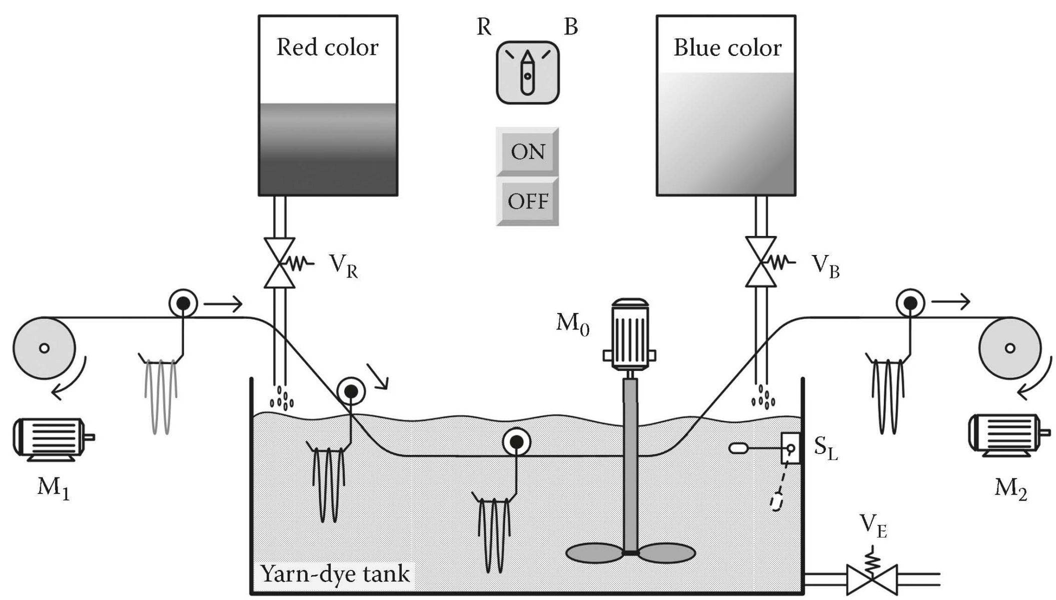


Figure 3.13: Schematic of a simplified batch-dyeing process for textile materials.

following operation:

... IF the selector switch is in the "R" position AND by pressing the push button "ON", the electric valve V_R opens AND motors M_1 and M_2 operate AND when the level sensor S_L produces an output signal the electric valve V_R closes AND simultaneously the motor M_0 starts to operate AND IF, due to a fault, the motor M_1 stops, THEN the motor M_2 also stops AND ...

All the previous demands, which are being produced from the type of the industrial process, constitute a set of dependencies that, without their formal satisfaction, the production of the final product is not feasible. The devices from these five categories, which are indicated in Figure 1.3, with their dependencies and the rest of the purely mechanical parts of the integrated machines or processes constitute the overall industrial system. This industrial system needs proper automation in order for the whole operation to be executed with a minimization of human intervention. The control of the devices that are characterized as ON and OFF states is denoted as automation, and is being carried out by the use of automation circuits; while the control of the devices that are analog is denoted as process control, and is being carried out by the use of automatic control systems.

2.2 Process Control and Industrial Automation

Special attention should be focused on the differences in the fundamental meanings among the concepts of automatic control and industrial automation. Automatic control can be defined as the continuous control of a physical analog variable through the use of any kind of actuators, while industrial automation refers to the sequential or digital ON-OFF control of a two-state device. As has been presented in Figure 3.10, among the discrete devices, a continuous time control device has also been included in the industrial system, in order to present the overall concept thatin an industrial control system, multiple continuous time control units can be integrated and act in a cooperative way with the rest of the automation control units.

In the case of industrial control processes (batch processes), there are multiple process variables that, although we would like to have them set at constant values, show random variations, mainly due to multiple external disturbances during the production phases. The reduction and elimination of these variations can be achieved through the proper application of automatic control principles. In many cases, it is also desirable for a process variable to alter the set value from an existing converged one into another operating point, while certain specifications usually are amended to achieve this transition, e.g., a fast or slow transition time, a minimum control effort change, a low overshoot during the alteration of the set point, a fast convergence, etc. This problem can also be addressed by the theory of automatic control and by applying existing theoretical and applied approaches e.g., the theory of Proportional-Integral-Differential (PID) controllers.

In contrast to the automatic control principles, the theory of industrial automation focuses on physical variables and machines that are in one of two states, e.g., “a liquid flow exists or not”, “the pressure has reached the desired value or not”, or “the compressed air piston has been extended or not”. Moreover, industrial automation refers to devices, machines, and circuits; and, in general, electronic, electromechanical, and electro-pneumatic integrated machines, where their operational principle is described from the Boolean logic and the corresponding sequential interconnections among the production stages. In the automation field, the action of control is restricted by being applied by two state actuators, and therefore the applied control action can only have the specific values of either ON or OFF.

In Figure 3.14, a simple process of controlling the level of a liquid in a tank is presented. In this process, it is assumed that the supply of the liquid in the tank is provided by an uncontrollable variable, while a valve is controlling the liquid’s output flow from the tank. In the described setup, it is desired that the level of the tank be kept at a specific height h_0,

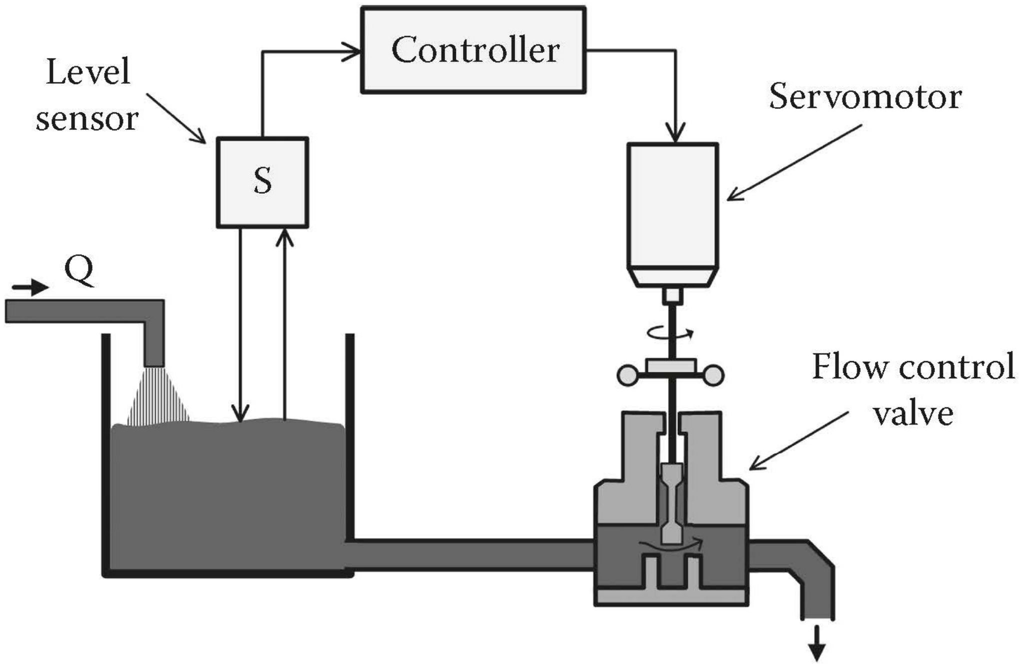


Figure 3.14: Flow control.

independent of the liquid supply. To solve this problem, after the initial achievement of the specified height h_0, the output flow should be equal to the input flow. To implement this control law, due to the fact that in this example the input flow is not directly measured, the control scheme should be able to measure the level of the liquid in the tank by a level sensor and, subsequently, appropriately tune the outflow valve. In this case, the outflow valve has not only two states "Fully Open—$Q_{\max}$ flow" and "Fully Closed—0 flow", but it can take any kind of desired state value, thus allowing for a flow within the $(0,\ Q_{\max})$ continuous space. In the era of classical industrial automation, this control scheme would have been implemented by analog circuits, whereas now it is commonly implemented by the use of computers and, more specifically, programmable logic controllers (PLCs), which are computational devices designed and configured for operating in industrial environments.

In Figure 3.15, a lead screw setup is presented, where the worktable can be translated by the proper connection and rotation of the lead screw into two directions (left and right). For this reason, the motor generating the rotation of the lead screw has two directions of rotation. Moreover, the motor has two rotation speeds, which means that the worktable can be translated in two speed profiles. With the help of the indicated position sensors (limit switches) and the provided motor, we can design an industrial automation with the following desired operation. As shown in Figure , the worktable should be continuously moving between the final positions A and D, while in the translation space from B to C the table should move at the fast speed and at the low speed for the remaining ones. In this setup,

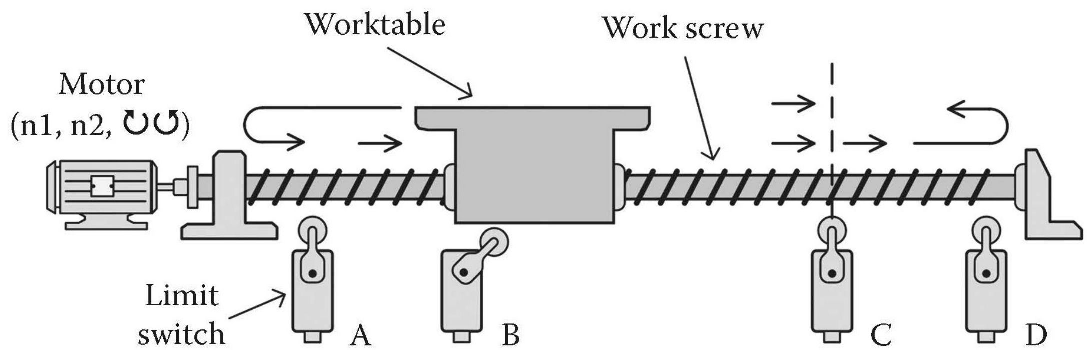


Figure 3.15: Lead-screw setup.

there is no continuous variable that needs to be controlled, while all the devices have two states of operation, ON and OFF. In this case, the controller is an automation circuit, which is responsible for implementing the described sequential (or Boolean) control logic, e.g., partially described in the following form

... *When sensor B is powered, the motor should be set (start operating) and remain in the fast speed. When sensor C is powered, the motor should be set and remain in the slow speed. When the sensor D is powered the motor rotation verse should be inverted, without changing the speed of translation* ...

At this point, it should be mentioned that the aforementioned translational system is not described by a specific transfer function, as in the case of automatic control systems, but from a Boolean function that expresses the desired operational logic.

Many books in the field of automatic control refer to the sequential control of two states as the fundamental form of industrial control, while providing minimum reference to this topic and concentrating on the analog and continuous time closed-loop control (feedback control). On the contrary, Industrial Automation will focus on the methods needed for designing and implementing industrial automation systems, which cover a significantly larger set of the current trends in the area of industrial control systems.

After defining that, in an industrial system, both continuous and sequential control setups exist, the term "industrial automation" now has a wider meaning, which includes every kind of system being designed for implementing an automatic operation of an industrial process.

2.3 How an Industrial Automation Scheme is really implemented

As stated in the first chapter, the industrial era started with the efforts to automate existing industrial setups as a way to improve the quality of the produced products and the overall production volumes. It is important to remark that, contrary to what is generally understood, industrial automation is not a discovery of the recent past, but it is rather as old as industry itself. From the beginning, the designer of an industrial production system has attempted to achieve an operation as autonomous as possible, always based on the available instrumental tools. The initial industrial production processes have based their operation on the workers' eyes, hands, and brain, as alternatives to contemporary sensors, actuators, and computational units. All of the current automated operations of industrial processes are based on these three factors. Through the sensors, the necessary signals and measurements are being gathered from the controlled process, as presented in Figure 3.16. Subsequently, this information is

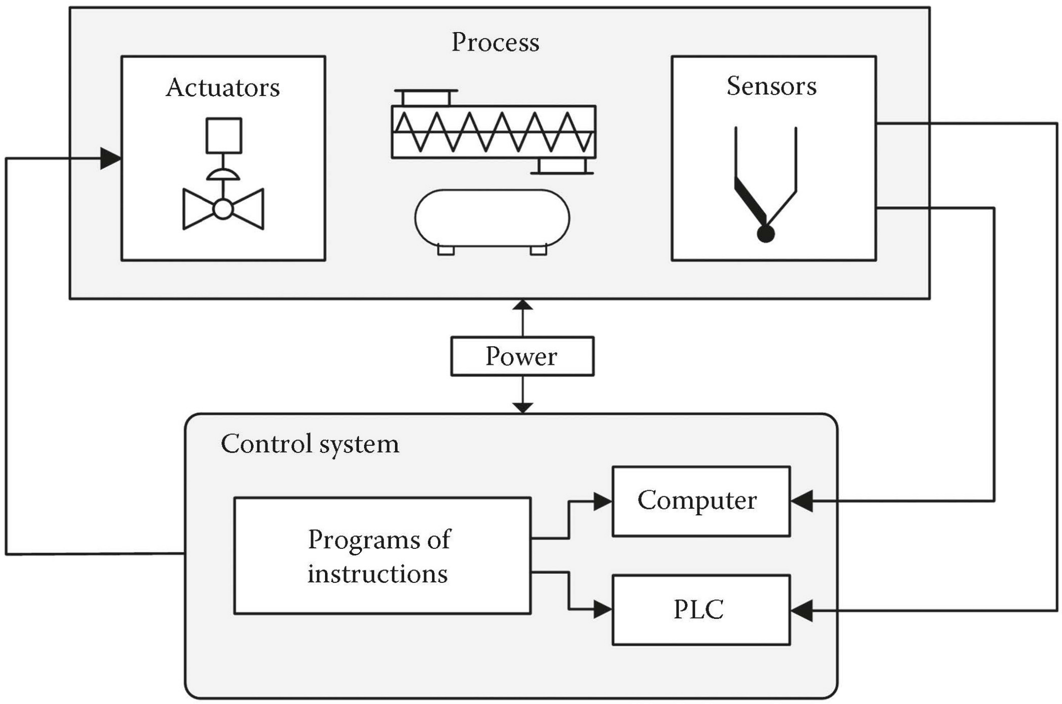


Figure 3.16: Automation System scheme.

being analyzed by the control logic, running in a computational unit and, in the final step, the control actions are interacting with the controlled process, through proper control of the provided actuators In the beginning, efforts to automate industrial processes were focused on the replacement of human labor by independent machines, each one being able to accomplish a specific task and in a limited surrounding space, under the premise that you feed the machine with the raw material, and as an output you are receiving the complete product. The automatic operation of those machines were initially independent of each other, and thus there was a constant need for a human-centric coordination of these machines. Subsequently, through the evolution of multiple, related technologies and through the developments in the field of analog and discrete control in the era of microprocessors and PLCs, a transition took place from having small-scale, centralized, industrial automation, to a decentralized and large-scale one, fully controlled by numerous distributed PLCs, able to synchronize multiple industrial processes.

Remember that fundamental motivations for the automation of an industrial process (see also Chapter 2, Section 3.1) are related to the viability of the enterprise, and particularly business profit that can be achieved through the following objectives:

- Production increase
- Cost reduction, mainly due to the reduction of human-related cost
- Improvement of the product's quality
- Improvement of the raw material utilization (reduced loss in materials)
- Reduction of energy consumption

Many other secondary benefits may be derived from the automation of an industrial process, i.e., plant safety, environmental pollution reduction, etc. The aims of industrial automation are frequently difficult to achieve for several reasons, such as inherent limitations of the plant, implementation costs and general situations in the marketplace. Regardless of these difficulties, there has been continual development of industrial automation from a control tools and methods point of view. Consequently, advancements in automation and control made possible the development of larger and more complex processes of various kinds (e.g., oil and gas refining, chemical, pharmaceutical, food and beverage, water and wastewater, pulp and paper, mining, iron and steel, cement, etc.), thus bringing numerous new technological and economic benefits.

2.4 Industrial Automation Circuits

The design of automation circuits, from a functional standpoint, is essential knowledge for all industrial automation engineers. It enables them to understand industrial operations, identify needs, design and simulate solutions, and ultimately deliver cost-effective outcomes, whether through traditional industrial automation or PLC software. In the broader field of electrical engineering, various types of circuits are encountered, such as those involving basic components (R, L, and C), electronic circuits, power circuits, telephone circuits, integrated circuits, and more. However, the electrical circuits used in the study and implementation of an industrial manufacturing plant can be categorized as follows:

- Power circuits
- Automation circuits
- Wiring diagrams

Power circuits (also called "main circuits") indicate the type of power supply for the utilized motors and all other related power devices. As an example, Figure 3.17 depicts: (a) the single-line, three-phase circuit of a motor with two directions of rotation; (b) the complete multi-line circuit of the same motor; and (c) the power circuit of a direct starting motor. Automation circuits (which can also be referred to as control circuits, auxiliary circuits, secondary circuits, or schematic circuits) represent the operational logic and control of the power devices, as indicated in Figure 3.18, for a start/stop operation of the previously depicted motor in Figure 3.17-c. Wiring diagrams are circuits representing both the power circuit and the automation circuit while, at the same time, representing the actual positioning of all the devices and components in the industrial installation, which is ideal information for the

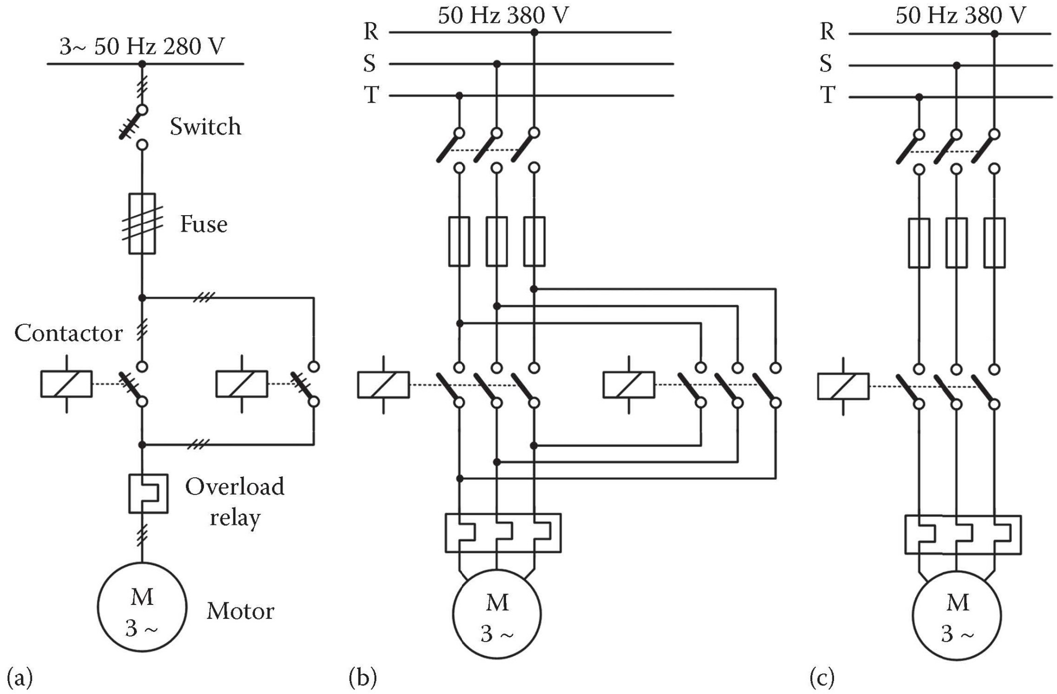


Figure 3.17: Examples of power circuits: (a) Single-line, three-phase circuit design, (b) complete multi-line design of circuit, and (c) circuit design of a direct starting motor.

technician executing the wiring and overall installation. In Figure 3.19, the wiring diagram which is produced from the synthesis of the power circuit shown in Figure 3.17-c, and the automation circuit shown in Figure 3.18, is displayed. It should be noted that in the wiring diagram of Figure 3.19, the wires can intersect among each other and thus it is very difficult to follow the route of each wire and the overall functionality of the design, even in this very simple case, where we have only four devices (one relay, two buttons, and one overload protection). Having this in mind, it can be easily generalized how this complexity will grow in cases with more intersections, e.g., in wirings with 50 devices. In contrast to this, in the automation circuit of Figure 3.18, there are no wiring intersections and thus it is very easy to follow and understand the overall operation logic. For this reason, automation circuits provide an overview of the automation functionalities are most commonly utilized during the development, installation, and operation of an automation system.

Automation circuits are being developed in branches (sectors), which are presented in Figure 3.20.

Each branch denotes the operational function of a corresponding relay, solenoid, or actuator, while the whole automation circuit denotes the operational logic of the overall industrial process automation. Each branch in the automation circuit can have multiple parallel sub-branches, depending on the complexity of the logical function being implemented. In Figure 3.20, the indicating branches have the simplest form. Each one implements the logic "If the rotary switch RS_i is closed, then the corresponding i motor will be in operation". An industrial system with an automation circuit of the form presented in Figure 3.20 has a full manual operation. In reality, this is not common, since the start and stop operations of an

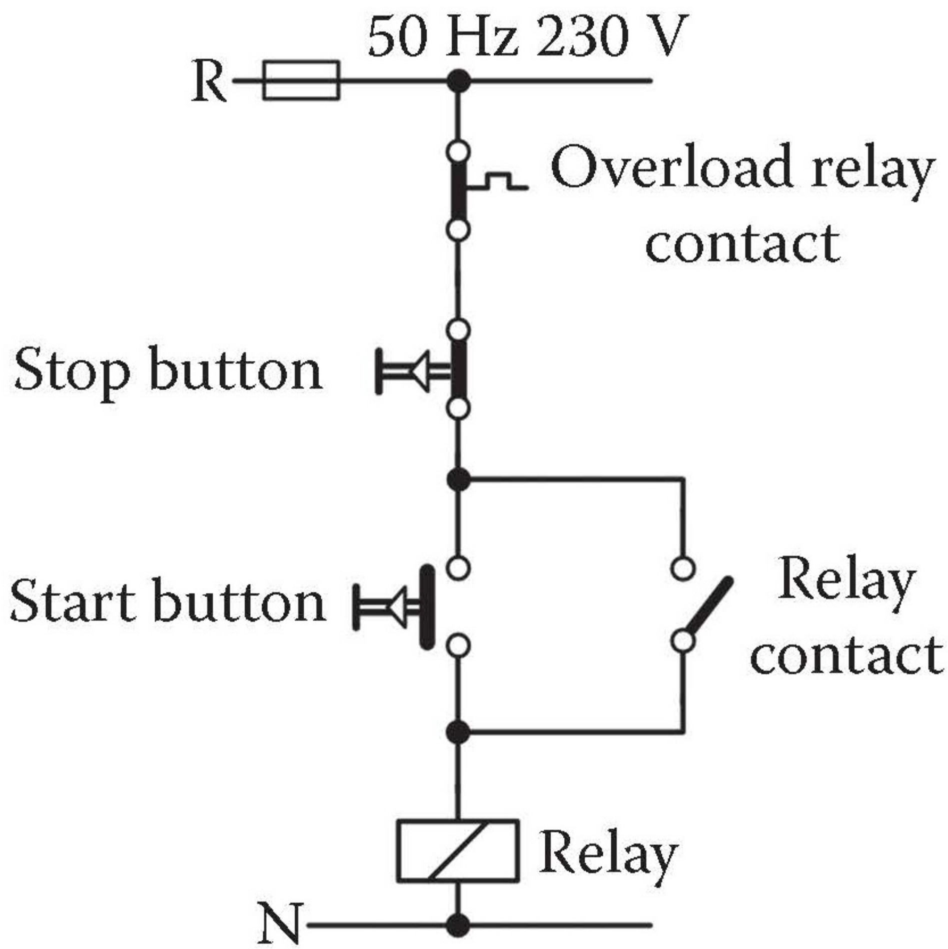


Figure 3.18: Example of a simple automation circuit.

industrial process with similar machines are executed in an automated manner, based on the sequence of the sensing signals and the corresponding status of the machines. In these cases, which are dominant in industrial automation, automation circuits are becoming more complex and thus a proper methodology for designing such automation circuits is needed. Nowadays, industrial automation circuits have been transformed, as has been mentioned before, into a set of software programs for PLCs. However, it is of paramount importance to note that although the final implementation of the industrial circuits has been changed from a hardwire approach to a soft approach, the need for understanding and designing the electrical drawings for solving an automation problem cannot be replaced, except for cases where the focus is on very simple and small automation problems. In the next chapter will present that the first step in writing the PLC program is to solve the problem, based on the methodologies that will be discussed subsequently, independent of the selected PLC or software language for the program implementation.

At this point, it should be mentioned that it was the authors' aim, when writing this

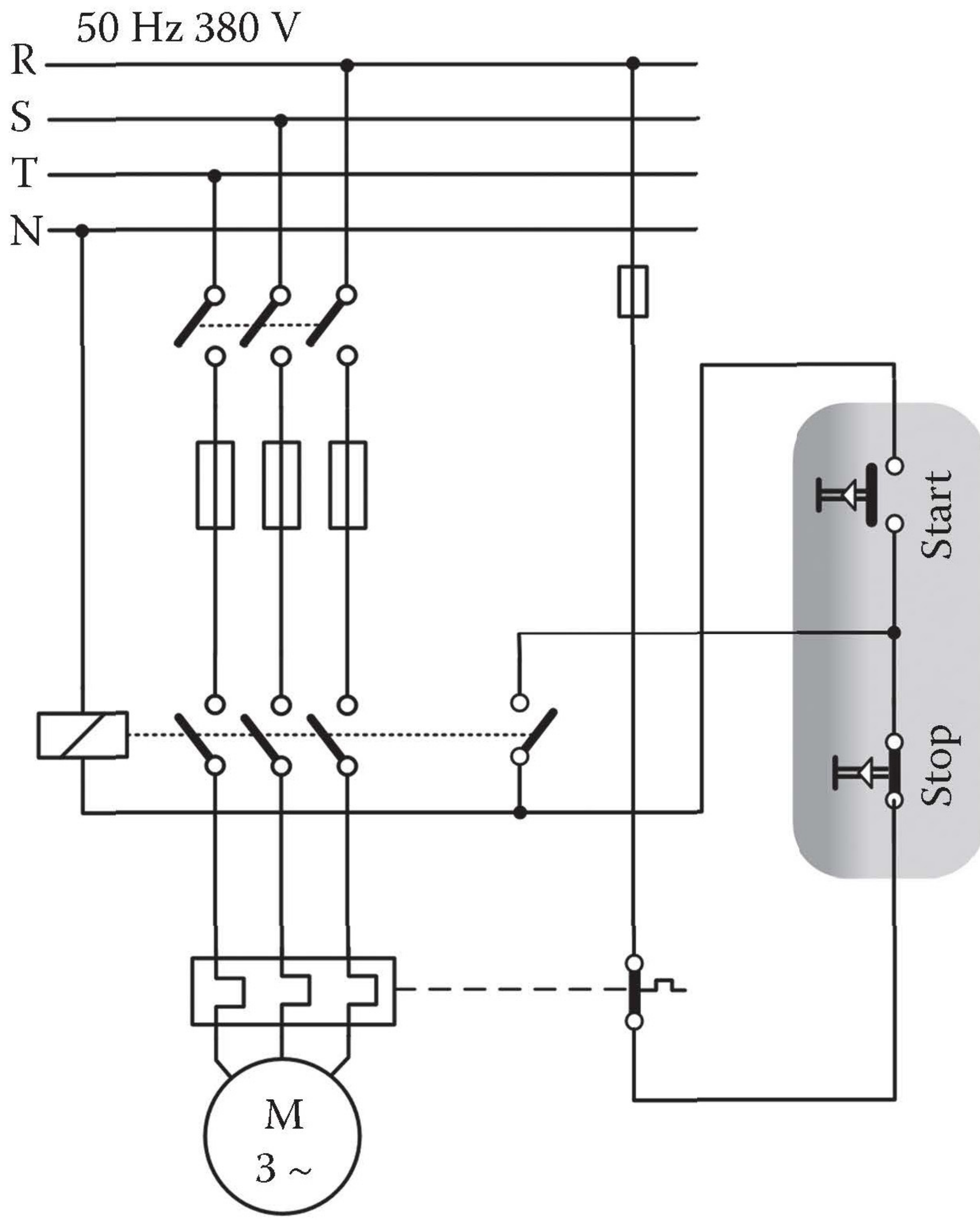


Figure 3.19: The actual wiring diagram of a direct start-up motor, including both power and automation circuits.

book, to present all the necessary steps to the interested engineers or automation students for understanding the concepts of an industrial automation, mastering the procedures and the methodologies for developing the automation circuits, mastering the design methodologies for more complicated automations based on state machines, and understanding and mastering the principles of PLC programming and PLC networking. However, it should be highlighted

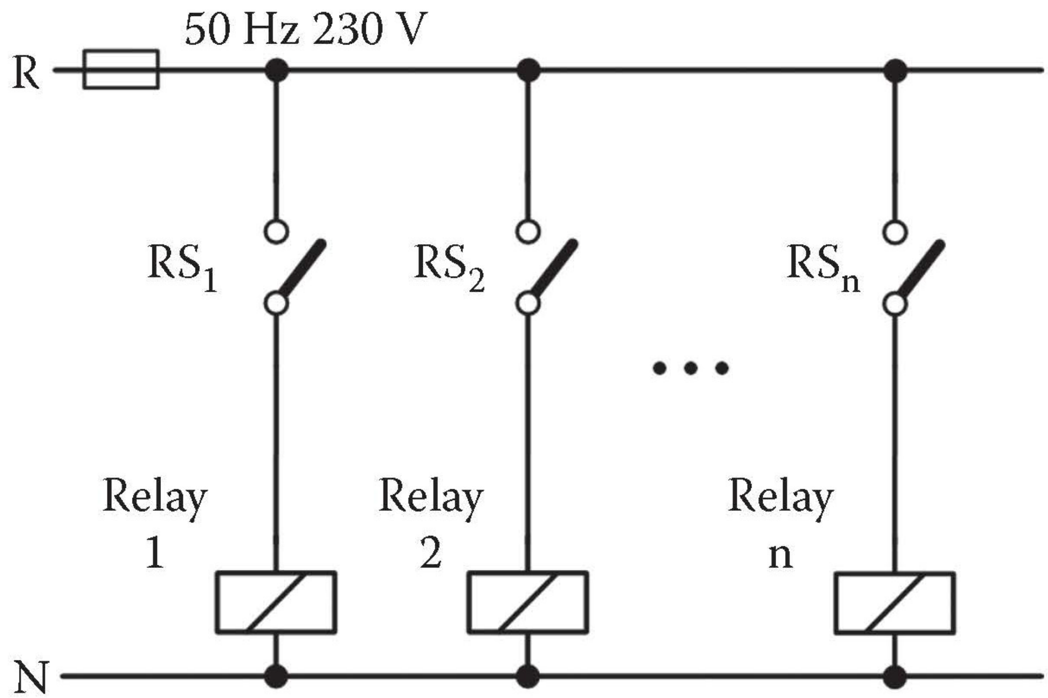


Figure 3.20: The simplest form of an automation circuit which is equivalent to a full manual operation.

at this stage that an industrial automation engineer is not a software programmer of a PLC, which is a mistake usually reproduced by lots of books in the field. The implementation of a fully functional and optimal industrial automation, as will be described subsequently, involves the understanding of fundamental principles in the area of sensors, actuators, electrical wiring, electrical machines, electrical circuits, process control, programming, and networking. However, conversely, a good programmer is not an industrial automation engineer, and thus programming of PLCs is just a small subset of the capabilities found in an experienced and professional automation engineer.

2.5 Computer-Based Industrial Control and Automation

The task of controlling an industrial process has evolved a lot over recent years, starting from a complete manual operation, continuing in the analog control and low-level automation era, and recently reaching a totally computer-based control and automation approach. Prior to the introduction of solid-state electronics, the designer of an industrial production process was attempting to make the automation operate as automatically as possible, based on the various instrumental tools. To enable the vision of a full automation technology and after the appearance of various digital processors, a rapid increase in process control computers and minicomputers took place, especially in small plants, which changed radically the situation in the field of industrial process control and automation.

Nowadays, an industrial control and automation system, from a hardware point of view, is a general term that encompasses several types of digital devices, such as industrial personal computers (I-PCs), programmable logic controllers (PLCs), programmable automation controllers (PACs), embedded PLCs, and other specific digital controllers. Furthermore, the larger control and automation system configurations include software and hardware platforms, such as supervisory control and data acquisition (SCADA) subsystems, distributed control subsystems (DCS), and industrial communication subsystems. At this point, it should be highlighted that the utilization of all of the aforementioned technologies in the industrial sector is of critical importance in order to achieve the desired performance and quality, while a proper mixture of all these computer-based solutions should always be considered.

After the introduction of the first powerful personal computers and PLCs, automation engineers have been divided into two groups. The first group was in favor of utilizing the PCs, equipped with proper input/output (I/O) hardware in order to accomplish a proper industrial automation functionality; while the second group rejected the PCs as inappropriate computational devices in an industrial environment, while promoting PLCs for the same purpose. However, these two categories have specific characteristics with certain advantages and disadvantages. PCs provide the user with the ability to utilize various software sets, spanning from simplified to extremely complicated software applications for implementing advanced control laws and industrial automations, providing extended graphical user interfaces and advanced interaction capabilities, increased computational power, and, in general, a simpler and more flexible programming environment for the user. From the other side, PCs are generally not suitable for a pure industrial environment. Even if the PCs can be equipped with the proper I/O hardware, they have the general disadvantage of not having been designed for installation in rugged industrial environments, and thus are characterized with generally reduced operational stability and durability. In contrast, PLCs have been designed specifically for industrial control and automation applications, are characterized by a high operational durability, and are equipped with a reconfigurable digital and analog I/O hardware that could be specifically tuned to the needs of the current application. Finally, PLCs provide fully optimized software for the exact needs of the industrial automation and process control, and nowadays this technology is considered as a standard solution in the industry. It could be left unattended and in continuous full operation for decades without operational errors or faults. For these reasons, PLCs are considered as the first choice of automation engineers, especially when compared with the classical PCs targeting a more home-based operation. From another point of view, PLCs are unable to support advanced control algorithms, are dedicated platforms for developing automation algorithms and have no support for other types of software. As a disadvantage, PLCs don't have a universal, standardized, and widely accepted way of communication with other types of devices from other vendors, thus restricting automation engineers in integrating products from specific vendors. Furthermore, after the finalization of the automation programming (the hardware connection to the I/O field devices and the initialization of the run state), a PLC operates as a "black box", without the ability to provide to the user any kind of online information, except for elementary information via optical light-emitting diodes, which indicate only the states of the digital I/Os. Regardless of these disadvantages, the PLC is still a very effective solution for general-purpose industrial control and digital I/O automation, mainly because of its reliability and transparent scope for which it has been developed.

The natural and acceptable competition among PLC vendors and the aforementioned industrial engineering groups, and the prevailing analogous situation in the marketplace of industrial controllers, were the reasons for various vendors to develop ways to remove boundaries between these two hardware technologies and add advanced functionalities, one of which has been the "industrial PCs". During the last few years, industrial PCs have been significantly expanded and improved in order to cover the existing gap between PCs and PLCs, but this category still has not replaced PLCs, nor has it been widely accepted and installed to a large extent. Additionally, industrial PCs have introduced multiple integration issues to engineers, due to the included multivendor hardware and software and the missing compatibility across different platforms.

The vendors of industrial automation systems for supporting the increased demands of the current industrial applications have developed industrial automation devices that could combine the advantages of PLCs for classical control and automation of a complex machine or of a process, with the advantages of the PC-based systems that provide the user with significantly high flexibility in configuring and integrating them into the industrial enterprise. Such a digital device has been established in the industrial world with the term programmable automation controller (PAC). A PAC is generally a multifunctional industrial controller, which can simultaneously monitor and control digital, analog, and serial I/O signals from multiple sources based on a single platform, while supporting multiple, built-in communication protocols and data acquisition capabilities. Although PACs represent the latest proposal in the programmable controllers' world at this time, the authors are not able to predict the future and the overall applicability of this technology. However, it is commonly agreed that PACs are an efficient and promising solution for complex industrial control and automation applications. In Figure 3.21, an overview of the available fundamental computational components for the implementation of industrial automation and control systems is presented.

In parallel with the developments in computational power in the control and automation devices, their ability to communicate, interact, and exchange information has also been developed in recent decades, thus leading to the introduction of industrial networks. Starting from a small number of industrial networks, and being introduced by three or four large vendors of industrial automation equipment, there exist more than 20 industrial networks today, addressing all levels of industrial production. Industrial networks differ quite significantly from traditional enterprise networks due to their specific operational requirements. More specifically, industrial networking concerns the implementation of communication protocols between field equipment, digital controllers, various software suites, external systems, and graphical user interfaces. In general, by allowing the connection of digital industrial controllers, the industrial network offers mainly the possibility of sensing messages and control commands through a decentralized approach, which can be geographically spanned. Thus, today the controller of a specific production process could sample the information from another part of the factory automation or control the operation of a machine in a remote part of the industrial field. Since this concept can be fully generalized on the full automation floor, the ability to control the whole industrial process and to have a complete overview of the ongoing sub-processes has been made more achievable than ever before, and thus the concept of supervisory control and data acquisition (SCADA) has been introduced. Furthermore, today's industrial networks can interconnect industrial controllers from different producers,

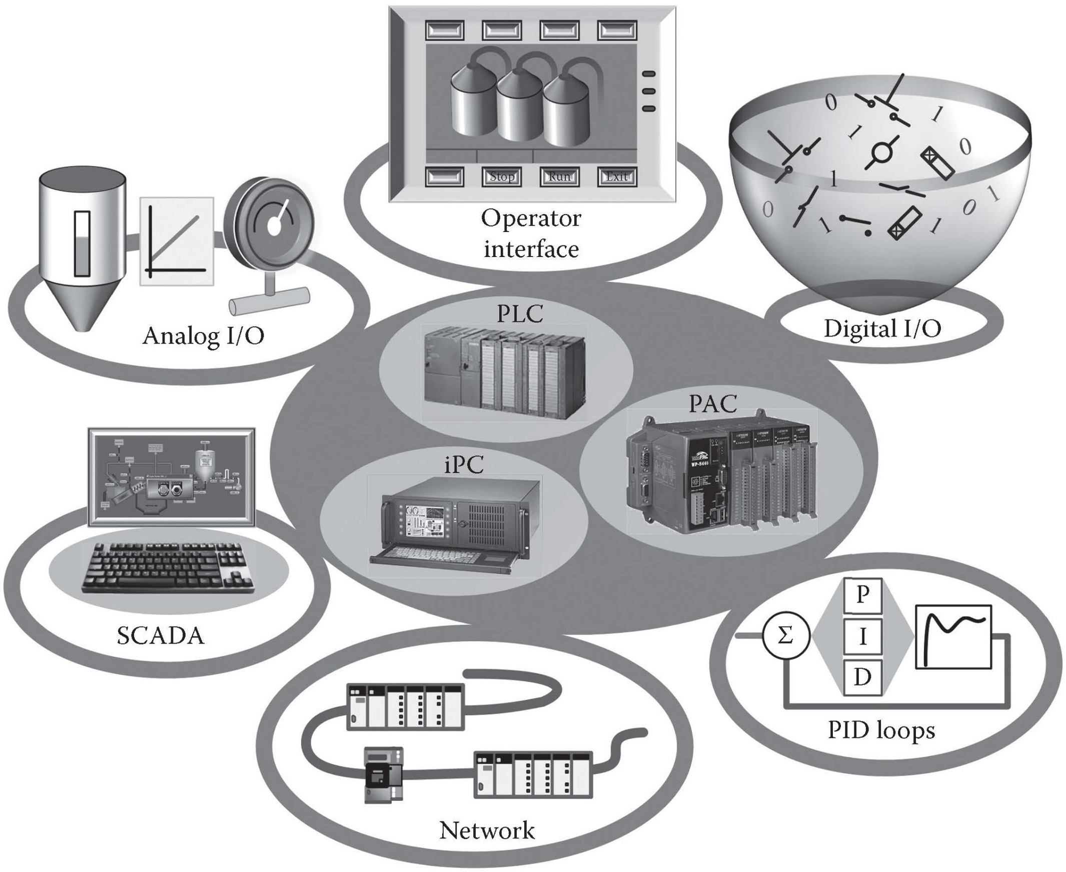


Figure 3.21: PLCs, iPCs, and PACs support control, communication, and other tasks.

converging in a similar way as the well-known “open communications” demand.

The SCADA concept has been introduced from the real need to gather data and supervision like control subsystems on a large industrial process plant in real time. Regardless of its initial definition, the term SCADA today represents a combined hardware and software system, including the remote field devices, the network, the central station equipment and the software platform. This software platform, in the case of SCADA, offers the user all the functionality required to receive or send data, represent data graphically, manage alarm signals, perform statistic calculations, communicate with other databases or software applications, schedule control actions, print various reports, and many other user facilities. Although the focus of SCADA systems is data acquisition and presentation on a centralized human machine interface (HMI), it also allows for high-level commands to be sent through the network to the control hardware, for example, for the command to start a motor or change a set point in a remote place. A characteristic example of a SCADA system is presented in Figure 3.22.

Similar to SCADA systems are distributed control systems (DCS), even if these systems existed before the era of SCADA systems, especially in the cases of the oil and gas refiners' industries. The DCS system consists of a strong dedicated network and advanced process controllers, often with very powerful processors, while implementing multiple, closed-loop

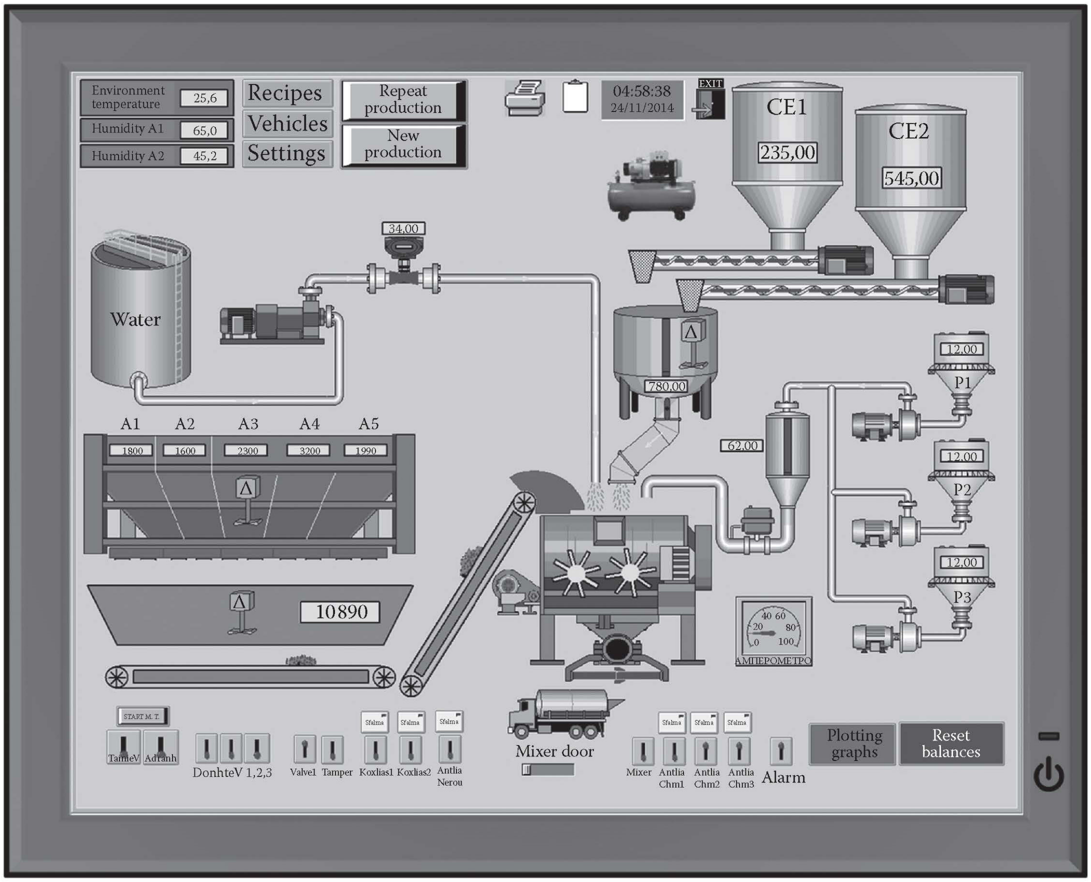


Figure 3.22: Typical example of a SCADA screen for process visualization.

controls of critical importance. In general, it should be highlighted that there is some confusion about the differences between these two types of automation systems, mainly due to the numerous common characteristics that these systems possess. A basic difference is the fact that the DCS is process-oriented, as opposed to a general-purpose software suite, and generally focused on presenting a steady stream of process information. This means that although the two systems appear similar, their internal operations may be quite different. SCADA systems, on the other hand, does not have the control of processes as a primary role, even if they have all the capabilities to apply limited closed-loop control and automation. The main focus of the SCADA system is the monitoring and the supervision of a process, which has been geographically distributed, most commonly through a multi-network communication structure. In contrast, the DCS is not concerned with determining the quality of data and visualization approaches, as communication with the corresponding control hardware is much more reliable. Even if the boundaries between these systems seem to be more blurred as time goes by, the computer and network technologies have become an intimate part of control and automation system engineering.

Based on the technology of industrial networks and the powerful computational automation units, the optimal implementation of the concept of computer integrated manufacturing (CIM) has been achieved, a concept that was initially introduced in the early 1970s. With respect to the CIM model, an industrial process can be organized in a three-layer hierarchical structure, where the lowest layer is comprised of sensors, actuators, and embedded microcontrollers. The middle layer is the control layer where the industrial controllers, industrial PCs, and industrial PACs are connected. The highest layer is the management level, where the mainframe computers for SCADA and resources planning functionalities are located. This three-layer generic structure is presented in Figure 3.23.

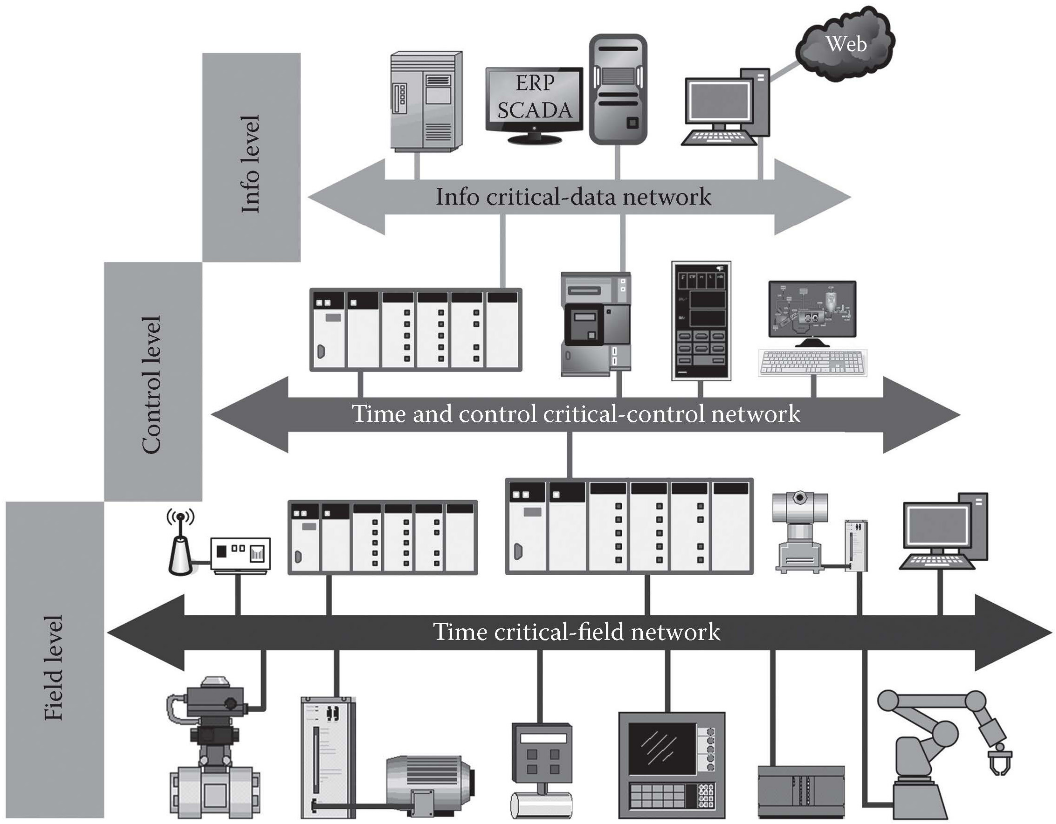


Figure 3.23: The CIM model integrates all levels of industrial production.

Every level of the CIM model has its own dedicated network, with its own technical characteristics for the network speed, the number of nodes, etc., due to different operational goals. As an example, it could be desirable to have a real-time control loop in the lowest field level and a periodic supervisor control loop in the highest level. It should also be apparent that the networks in the various levels are interconnected among themselves in order to allow for the transfer of information from the bottom layer to the top ones, and vice versa. Consequently, the target of CIM in the industry process is the integration and utilization of the overall information.

CHAPTER 4

THE PROGRAMMABLE LOGIC CONTROLLER (PLC)

Programmable logic controllers (PLCs) are used in every aspect of industry to expand and enhance production. Where older automated systems would use hundreds or thousands of relays. A single PLC can be programmed as a replacement. The functionality of the PLC has evolved over the years to include capabilities beyond typical relay control: sophisticated motion control, process control, distributed control systems, and complex networking have now been added to the PLC's list of functions.

What is a relay? *An electromechanical relay is an electrical switch actuated by an electromagnet coil (see the schematic in Figure 4.1). As switching devices, they exhibit simple "on" and "off" behavior with no intermediate states. Relays are very useful devices, as they allow a single discrete (on/off) electrical signal to control much greater levels of electrical power, and/or multiple power or control signals that are otherwise isolated from each other. For example, a relay may be controlled by a low-voltage, low-current signal that passes through a delicate switch of some sort (e.g. limit switch, proximity switch, optical sensor), and then the switching contacts of that relay may be used to control a much higher-voltage, higher-current circuit, and even multiple circuits given multiple sets of switching contacts.*

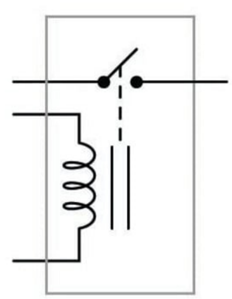


Figure 4.1: Relay schematic.

1 What is a PLC?

A programmable logic controller (PLC) is a specialized computer used to control machines and processes (see Figure 4.2). It uses a programmable memory to store instructions and ex-

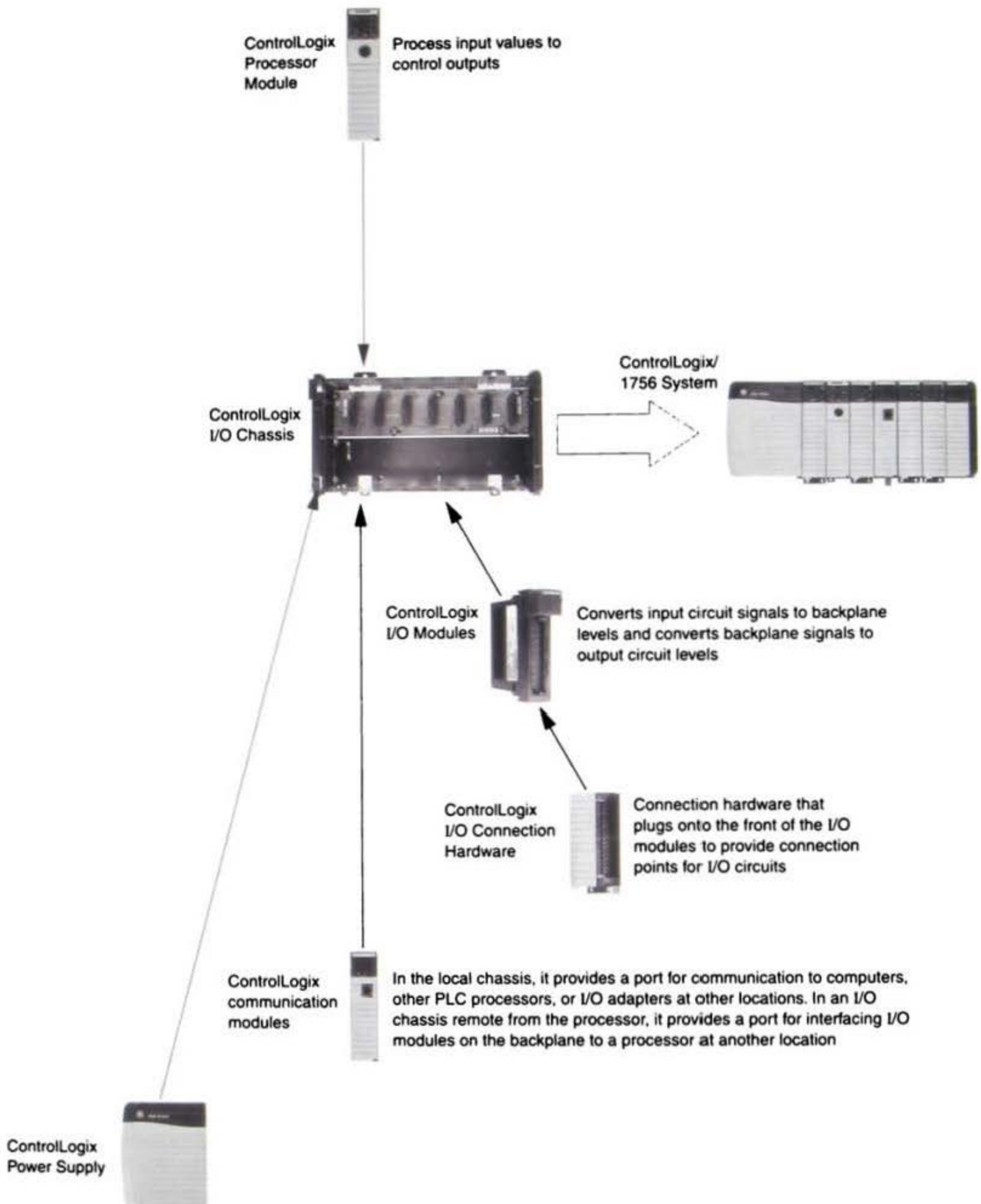


Figure 4.2: Programmable Logic Controller

ecute specific functions that include on-off control, timing, counting, sequencing. arithmetic, and data handling. The design of most PLCs is similar to that of other computers. Basically. the PLC is an assembly of solid-state digital logic elements designed to make logical decisions and provide outputs. Programmable logic controllers are used for the control and operation of manufacturing process equipment and machinery. The programmable logic controller is then basically a computer designed for use in machine control. Unlike an PC/Laptop, it has been designed to operate in the industrial environment and is equipped with special input/output interfaces and a control programming language. The common abbreviation used in industry for these devices, PC, can be confusing because it is also the abbreviation for "personal computer." Therefore, some manufacturers refer to their programmable controller as a PLC, which stands for "programmable logic controller."

PLCs initially appeared in the industry during the 1960s and had a completely different form than those implemented today, since they were built out of logical components that only replaced the operation of the auxiliary relays. Even primitive PLCs were very reliable for a long time when compared to relays, demanding much less space in the overall automation. Subsequently, their evolution passed through multiple stages, while the most important ones were inclusion of digital components for timing, synchronization and counting, and use of microprocessors. The microprocessors had already started to be a fundamental part of the personal computers (PCs). Nowadays, PLCs can be either simple or complex, come in a variety of sizes, and are equipped with a wide variety of extensions and interfaces that fulfill all the type of needs found at factory level, including the need to communicate with other devices and computers. It should also be mentioned that there are multiple programming languages for tuning the behavior of PLCs so that they can match the different programming skills of the end users.

Every PLC, independently of its type and size, can be characterized as a digital device with a microcontroller and a programmable memory that can store and execute user instructions expressing Boolean logic, sequential logic, timing, counting, and mathematical processing, in order to control the operation of a complex machine or an overall industrial process through the utilization of digital and/or analog inputs and outputs (I/Os). PLCs have the basic structure of a personal computer, with two significant differences. The first is related to the available hardware for the I/Os of the PLC, while the second is related to the microcontroller operation manner and its interaction with the rest of the electronic components of the PLC. A PC's main objective is to communicate with the end user for the successful execution of various arithmetic and algebraic calculations, graphical editing and representation, communication tasks, etc. Thus, in these cases, the end user provides the corresponding commands through a proper interface, such as a keyboard or mouse, while the outcome of these actions is either displayed on the monitor of the PC or printed. The PLC's main task is to communicate with the industrial environment and, more specifically, with either the input devices that are providing the sensorial measurements or with the actuators that interact with the process. For example, typical input devices are sensors, buttons, and switches, while typical output devices are power relays, valve coils, and indicating lights. Since these devices are operating at a different power level than the one that PLCs are usually operating at, it is necessary for PLCs to have the proper I/O hardware to adjust and adapt the power levels accordingly. In Figure 4.3 the basic parts of a PLC are presented: the CPU, the I/O modules, the RAM, and the power supply. A programming device is a peripheral device that is used only for the programming stage of PLCs, and is not necessary

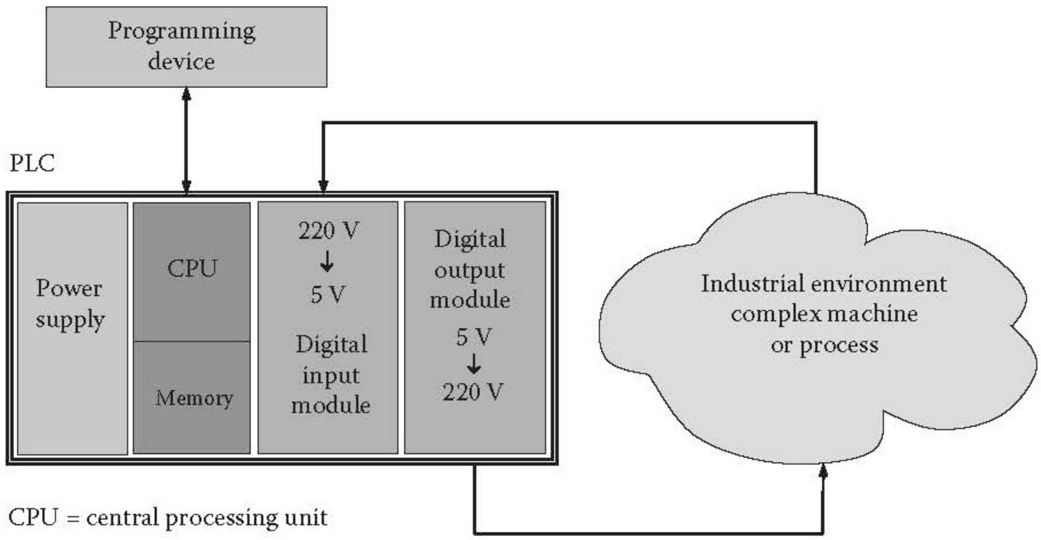


Figure 4.3: Programmable Logic Controller

for its operation, therefore it is removed afterwards. In some specific types of small PLCs, the programming device is embedded in its main body. In general, the programming device may be either a specially manufactured digital device (usually portable and specific to a PLC) or a classical PC equipped with the software that the PLC's manufacturer is developing for PLC programming. Before proceeding in analyzing the operation and the interaction of the PLC components, it is very important to define which hardware devices and tasks of the classical industrial automation the PLC is replacing. Remember that a classical industrial automation system needs the following:

1. Auxiliary devices (such as time relays, hour meters, counters, auxiliary relays, etc.) that constitute the basic electrical components of the automation and are mounted in an electrical enclosure

2. Design of the overall automation electrical circuit that has to achieve the desired operation of the controlled process

3. Wiring that is needed inside the electrical enclosure for connecting the auxiliary devices between them and also with the I/O devices existing in the enclosure

4. Wiring that is needed for connecting the electrical enclosure with the I/O devices as a whole, existing far from the enclosure. Input devices may be photoelectric switches, proximity switches, selector switches, etc., while output devices may be motors, electrovalve coils, other actuators, indication lights, etc.

As indicated in Figure 4.4, the first three cases are now embedded in the operation and programming of the PLC, while the last case remains the same, as in classical industrial automation. More analytically, a PLC contains several dozens of all the necessary classical industrial automation components (such as auxiliary relays, timers, counters, etc.) due to

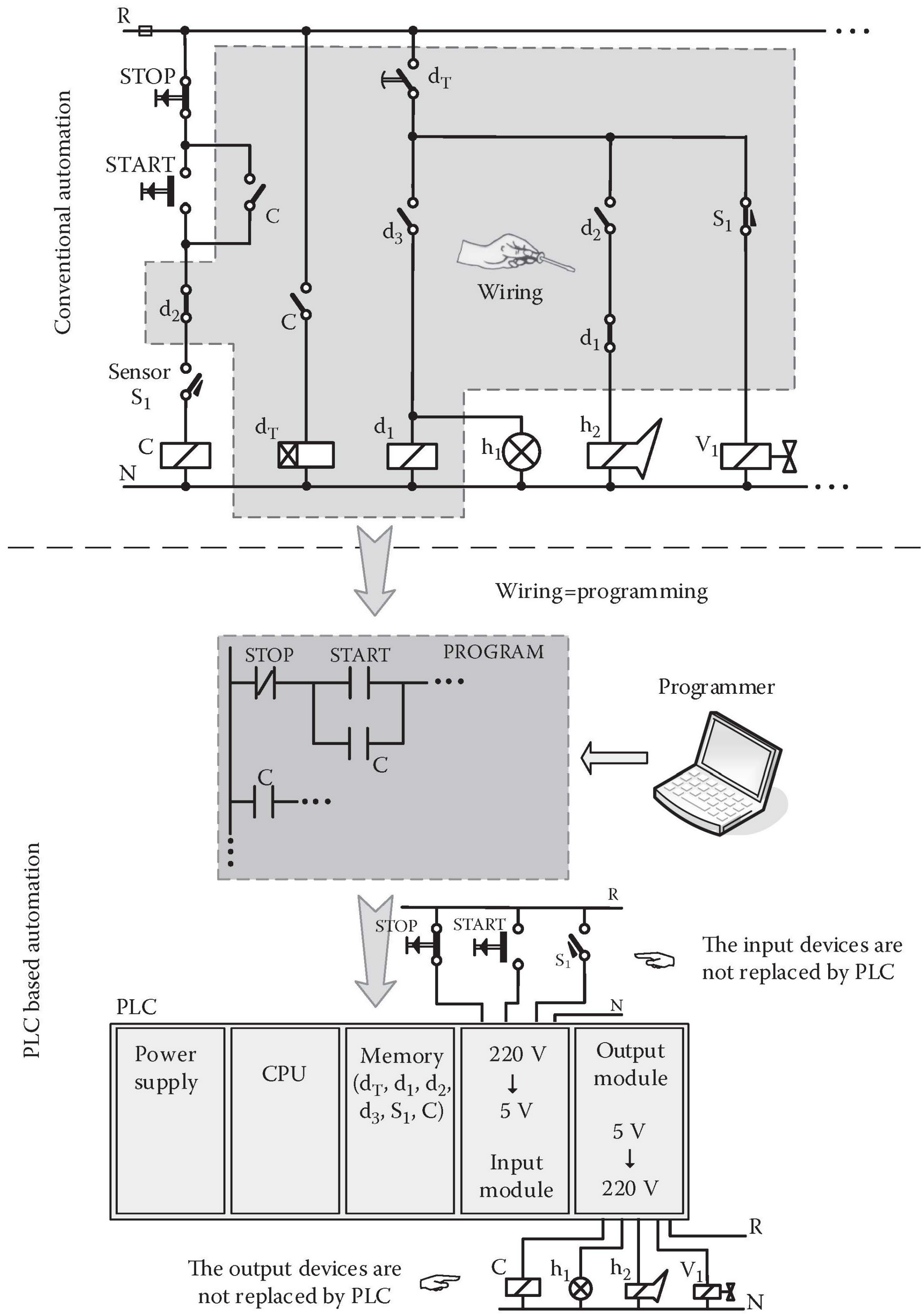


Figure 4.4: Conventional automation in comparison to programmable automation.

its digital form. Thus, the implementation of an industrial automation system does not require the purchase and integration of any kind of auxiliary devices. The design of the classical automation circuit, in most cases, is replaced by direct PLC programming. The effectiveness of the overall operation is dependent on the overall complexity of the code (software) for industrial automation, versus the complexity of the wiring needed to embed the automation logic in the electrical circuits. Thus, the role of a PLC nowadays is to transform the hardwiring into flexible software, and to serve as an expert tool for the industrial engineer to solve hard and demanding problems. At this point is should also be highlighted that the PLC is not replacing all the components of an industrial automation, since the power units still remain unchanged (e.g., power relays). As illustrated in Figure 4.4, all the corresponding I/Os remain unchanged, and are used to interact through the software that is running in the PLC.

In the following sections we will analyze the characteristics of all the components that construct a functional PLC in detail; however, for the proper understanding of this concept, we should initially emphasize the fundamental operational differences between PLCs and PCs. As has already been mentioned, a PLC contains a microprocessor that executes all the internal functionalities of the needed automation, as indicated in Figures 4.3 and 4.4. Furthermore, the processor is responsible for the execution of the user's programmed instructions; the utilization of the memory that stores the automation programs; as well as various types of data that concern the operation of the internal digital components; such as timers, counters, input components that transform high power signals into low power ones that are compatible with the digital logic of the PLC for their usage in the automation program, and output components that are transforming the low power commands from the PLC to the automation devices to proper and compatible high power signals. On the PLC side, there is a specific sequence in which the previous actions are executed. This sequence is cyclic and continuously repeated during the operation of the PLC in the RUN mode.

In Figure 4.5, the cyclic operation of the PLC, as well as the corresponding sequential actions in a more simplified approach, are depicted. Let's consider the fundamental circuit presented in Figure 4.5a. The corresponding logic is simple, and indicates that in the case that the rotary switch RS is closed, then the relay C is energized. If we want to implement the logic of this simple circuit in a PLC, the previous circuit is translated in proper software that it is stored in a specific place in the memory. Regarding the memory itself, there are two additional memory units, where one is dedicated to the storage of the output state and is called "Output Image Table" and the second is dedicated to the storage of the input states and is called "Input Image Table". Since the switch RS is an input device and is connected with the PLC through the input component of the PLC, let's assume the third input. Power relay C is an output device and is connected to the output component of the PLC, so let's assume the fourth output. The components of the program are instructions that are stored in the PLC memory and refer to the corresponding variables that in our case are input 3 and output 4. The switch RS in the beginning is closed. Let's assume that we would like the PLC to be placed in RUN mode, and that we would like to monitor all the initial steps, which the microprocessor executes based on the corresponding operating system. The input unit, controlled by the microprocessor, is sampling all the inputs, including input 3. This means that the PLC is detecting if there is a voltage or not in every input. Since the switch RS is closed, there is voltage in input 3, as indicated in Figure 4.5b. This voltage subsequently

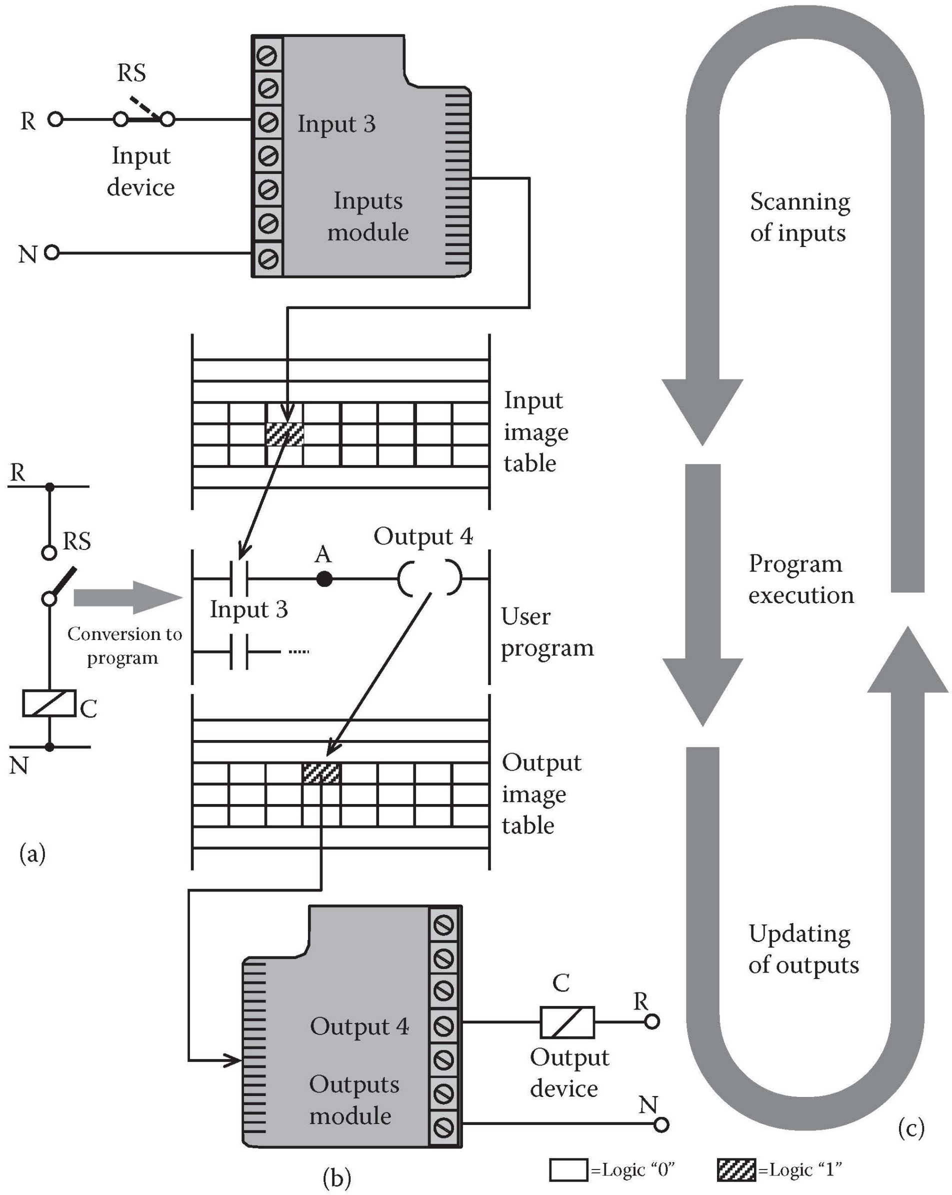


Figure 4.5: Cyclic operation (or scanning) of a PLC means continuous and repeated reading of inputs, user program execution, and updating of outputs: conventional circuit (a), I/O and user program memory in relation to I/O cards (b), and scanning cycle (c).

Figure 4.6: Ladder Symbols

is converted and properly adjusted from the input component in a low power TTL signal. The existence of this TTL signal is stored as a logical 1 in the memory of the input image table and at the position that corresponds to the third input. In the inputs where there is no application of voltage, a logical “0” is stored. After sampling of all the inputs, the microprocessor starts with the execution of the program. The instruction input 3, by definition means that the point A is at a logical “1” if input 3 is activated, and at a logical “0” if it is deactivated. Thus, for the microprocessor to execute this instruction, it is necessary to sample the status of input 3 through the input image table. Subsequently, the microprocessor executes the following in the list instruction output 4. By definition, this instruction also means that if point A is in logical “1” then output 4 should be energized, while if it is in a logical “0”, it should be deactivated. The activation of an output or not, as a direct result of a command execution, means the corresponding writing of a logical “1” or “0” at the output image table. In our example, the output 4 should be energized by the registration of a logical “1” in the corresponding memory position. Subsequently, the rest of the program's branches are executed, if more exist. When the whole program is executed, then all the states of the outputs have been stored in the output image table, digital “1” or “0”. Subsequently, the microprocessor transfers the output image table at the PLC's output component. Thus, in output 4 of the output component, a logical “1” TTL signal is transformed to a power signal that can energize a switching component, through which an output device is energized, or the relay C in our case. Subsequently, the microprocessor repeats the sampling of the inputs, executes the program from the beginning, and updates the outputs and repeats again the same cycle, as indicated in Figure 6.1c. This continuous cyclic operation of the PLC is known as the “scanning” mode. The time for a full scanning indicates the operational speed of the PLC, and should vary from a few milliseconds or less. If the variation of an input's state is faster that the scanning time, then these variations are not detectable from the PLC. However, it should also be noted that the scanning time is directly dependent on the speed of the microprocessor, while for a specific PLC, this time is dependent only on the size of the program (number of instructions) and the type of these instructions, since different instructions demand a correspondingly different execution time. As an indication of the scanning speed of a PLC, manufacturers usually provide the scanning time for a program that contains a set of instructions of 1 KB of memory. PLCs are not programmed according to an internationally standardized programming language that is adopted by all manufacturers. Instead, there are various forms of programming languages that vary from company to company under various names, even if they are similar in their functionality. Also, there is a significant incompatibility between similar programming languages developed by different manufacturers. The International Electrotechnical Commission standard 61131 (IEC 61131-3), adopted in 1993, deals specifically with PLC programming languages, and defines the most basic forms that fall into two categories: graphic languages and text-based languages. Despite the lack of an absolute standardization in the matter of scheduling, we can distinguish four main programming languages, which are: the cascade ladder diagram language (ladder), the instruction list language or Boolean (IL), the language of logic elements or function block diagram (FBD) and the sequential functional chart (SFC) language. The most popular of these are the ladder language, since it is very similar to the classic implementation of an automation circuit and the SFC language due to Modularity, Handling of Sate Machines and Parallel Processing. Ladder Language was also the reason for the adoption of this language by almost all manufacturers in the early years of PLCs, because in this way it was easier to spread the novel PLC technology and have it adapted by older engineers who were not familiar with the programming. In Figure 4.7, the general format of the SFC/Ladder programming languages

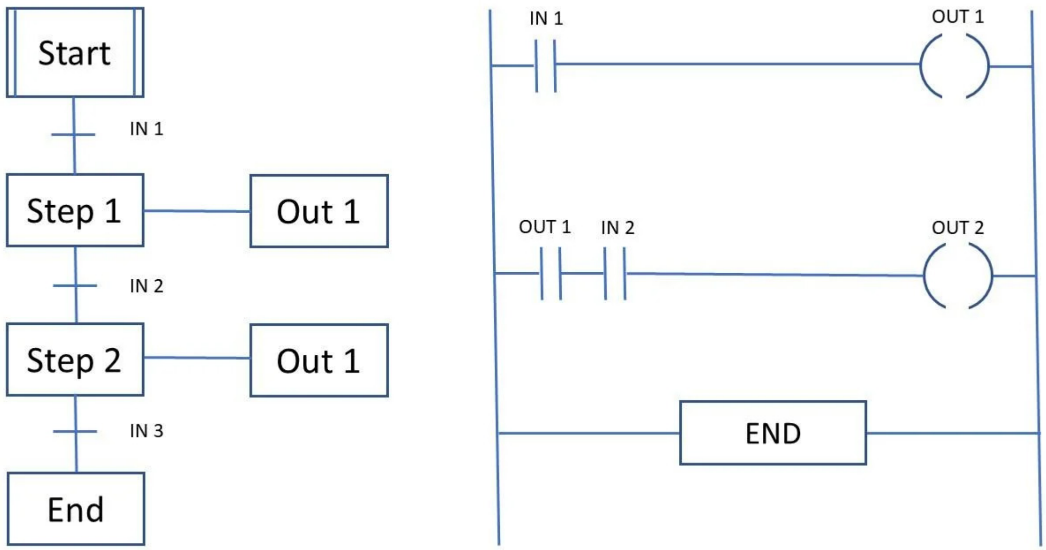


Figure 4.7: SFC/Ladder Languages

is presented. As mentioned above, the programming device for a PLC can be a PC or a specially designed digital device. In the second case, the programming device or programmer is not just a simple keyboard, but includes a liquid crystal display, a memory, a compiler, and various communica-tion ports to communicate with the PLC or other peripheral devices. These units are necessary in a programming device, since the programming is not performed directly in the PLC. For the programming procedure, initially the program is developed in the programming device and after-wards it is translated into a machine language that is stored initially in the programmer's memory and then transferred to the PLC's memory, provided that there is a communication link between the programmer and the PLC. From the programmer, it is usually feasible to monitor the opera-tion of the PLC in order to detect the status of the PLC's internal elements and to perform various diagnostic tests.

The power supply unit of a PLC is of secondary importance, since it simply provides the various voltages required in each section of the PLC. In certain types of PLCs, it is also possible to supply the circuits of the input devices, but in general the power supply unit never feeds the actuating circuits of the output devices, since the power required to activate the output devices is always supplied from external sources.

The development and spread in utilization of PLCs has been very rapid in recent years, while there is a continuous development of new models with more and more features, smaller sizes, and more affordable costs. Today, PLCs are used in any type of manufacturing process or complex machine, as well as in smaller applications (such as car washes, traffic lights, pumping stations, etc.), since PLCs are one of the most reliable automation solutions. The widespread use of PLCs in industrial automation is attributed to their numerous important advantages, which include:

- *Adequacy of the contacts.* When developing a conventional automation system, during the design of the corresponding automation circuit, we should always evaluate the efficiency of the auxiliary switching contacts of the power relays. When the required auxiliary contacts are numerous and are not available in the utilized power relay, then the engineer has the option to add these additional auxiliary contact blocks, or implement a parallel connection with a second relay in order to use these contacts as auxiliary ones. However, in the case of a PLC, there is no such issue, since the adequacy of the contacts is unlimited, as each internal memory bit location of a PLC can take the role of an auxiliary relay, which could be utilized as many times as we would like in a corresponding automation program. In reality, there is a limit that is dependent on the size of the PLC's memory.

- *Time saving.* For the development of a programmable automation system with a PLC, the writing of the program (design of the automation circuit) can be done in parallel with the installation of the PLC and its connections to the I/O devices, since the program is written in the programming device. In the case of conventional automation (classical automation wirings) this is not possible, since initially the automation circuit should be designed, then the industrial electrical enclosure should be constructed according to the designed automation circuit to perform the installation and its connections to the input and output devices.

- *Reduced need for space.* Since the PLCs are digital devices, they have a comparatively small volume as well as dozens of timers, counters, and hundreds of auxiliary relays, thus their volume is incomparably less than that of a conventional industrial automation enclosure with an equivalent number of auxiliary equipment.

- *Easy automation modification.* The alteration or simple modification of a conventional automation circuit can be performed only by means of removing cables, adding new changes of equipment and, in the worst case scenario, by stopping the operation of the control system for some time. However, in the case of PLCs, all the above modifications are simply equivalent to the direct alteration of the corresponding program that, after the required amendments, can be directly downloaded onto the PLC online or with a pause of the overall operation that lasts for a few seconds.

- *Easy fault detection.* With the help of the PLC's programming device, the status of the PLC's internal elements and the corresponding execution of the loaded program can be directly monitored. In addition, the ON or OFF state of all input and output devices can be further monitored through the utilization of indicative LEDs. Furthermore, the possibility of "forced" (virtual) or simulated notional state changes of an input device, for observing the reaction of the automation system and the overall control logic, can be directly performed in the PLC, mainly due to its digital structure that allows the performance of various tests that assist in the troubleshooting.

- *Modern and working tools.* PLCs have significantly contributed in altering the working environment of engineers, since they have transferred them from the field of cables, auxiliary relays, screwdrivers, etc., to working in an environment similar to one that the PCs have. Engineers have to work with a keyboard or mouse and a program in Windows or in another environment and simply print the automation program instead of designing the automation circuit and applying their knowledge of digital systems.

All these concepts created a different and modern operating environment, especially when compared to the corresponding one some decades ago.

Subsequently, in the next sections, the hardware and in particular the software (Programming Languages) of PLCs are presented in detail.

1.1 PLCs and PC/Laptops, main differences

PLC ad PC/Laptops serve different purposes, have different designs, and are intended for operation in distinct environments. Below, we explore the detailed comparison between these systems.

Purpose

PLC: Specifically designed for industrial automation and control applications. PLCs are used to monitor inputs, execute control logic, and manage outputs to control machinery or processes.
PC/Laptop: Designed for general-purpose computing tasks such as office work, web browsing, multimedia, gaming, and software development.

Operating Environment

PLC: Built to withstand harsh industrial environments, including extreme temperatures, dust, electrical noise, vibration, and humidity.
PC/Laptop: Designed for controlled environments such as offices or homes, where environmental factors are regulated.

Reliability and Stability

PLC: Extremely reliable, designed for continuous, long-term operation without failure. PLCs are expected to run 24/7, often without human intervention.
PC/Laptop: Less reliable for 24/7 operation in industrial settings. PCs may require frequent updates and reboots, which is not ideal for critical control systems.

Real-Time Operation

PLC: Operates in real-time, processing inputs and outputs with minimal delay. This is critical for controlling industrial machinery or processes.
PC/Laptop: Standard PCs are not designed for real-time applications. Operating systems like Windows or macOS may introduce latency, making them unsuitable for critical control tasks.

Durability and Life Cycle

PLC: Highly durable with a long operational life, often over 10 years. Designed to run in demanding conditions without frequent hardware failure.

PC/Laptop: Less durable for industrial use. Components such as hard drives and cooling fans may fail more quickly in harsh environments.

Expandability and I/O

PLC: Designed with extensive input/output (I/O) capabilities. PLCs handle digital and analog inputs/outputs and are compatible with various sensors, actuators, and industrial devices.
PC/Laptop: I/O capabilities are more limited. While PCs can be expanded with additional hardware, this is not as seamless or robust as a PLC's I/O system.

Programming

PLC: Uses specialized languages like Ladder Logic, Function Block Diagram (FBD), Sequential Function Chart (SFC), or Structured Text (ST), all tailored to industrial control tasks.
PC/Laptop: Programmed using general-purpose languages such as Python, C++, or Java, which are powerful but not inherently designed for industrial control.

Communication and Networking

PLC: Utilizes industry-standard communication protocols like Modbus, Profibus, EtherNet/IP, CAN, etc., for communication between industrial devices and systems.
PC/Laptop: Uses general-purpose networking protocols such as TCP/IP, Wi-Fi, and Bluetooth, suitable for everyday tasks like internet browsing and file sharing.

Cost

PLC: Typically more expensive due to its long-term reliability, durability, and industrial-grade design.
PC/Laptop: Generally cheaper in terms of upfront cost, but less reliable and durable for industrial use.

User Interface

PLC: Often lacks a built-in user interface (UI) beyond basic indicators. Typically requires external Human-Machine Interfaces (HMIs) or SCADA systems for interaction.
PC/Laptop: Comes with advanced graphical user interfaces (GUIs) and operating systems like Windows or macOS, making interaction intuitive and flexible.

Energy Consumption

PLC: Consumes less power, optimized for long-term efficiency in industrial environments.
PC/Laptop: Typically consumes more power, especially with high-performance components like processors and graphics cards.

Fault Tolerance and Redundancy

PLC: Many PLCs include built-in redundancy and fault tolerance features, ensuring operation continues even in the event of hardware failure.
PC/Laptop: Standard PCs lack these features, making them more prone to single-point failures.

PLCs and PCs/Laptops serve different purposes, with PLCs excelling in industrial environments where real-time processing, reliability, and ruggedness are critical, while PCs and laptops offer versatility for general-purpose computing tasks but lack the durability and control-specific features needed for industrial applications.

2 How a PLC is built (more detailed discussion)

A typical PLC can be divided into parts. as illustrated in Figures 4.8, 4.9 These components

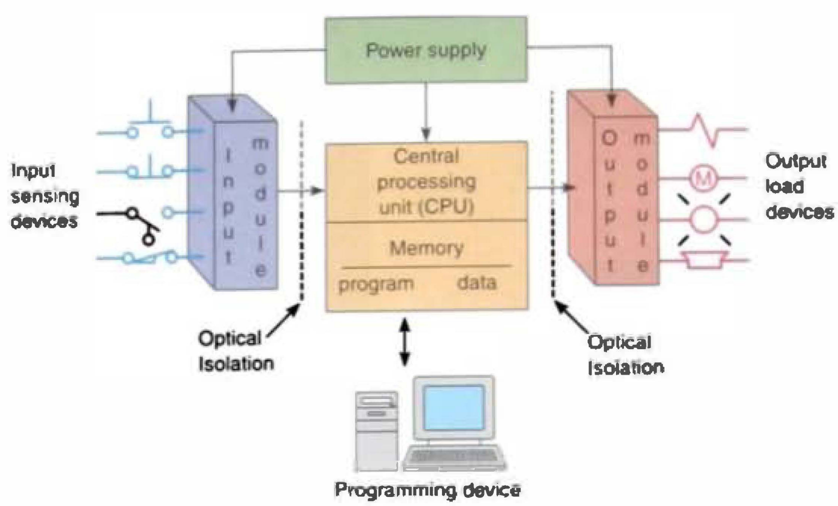
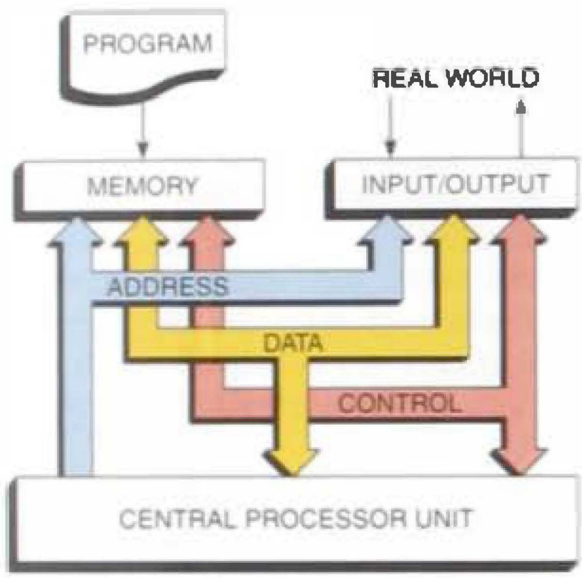


Figure 4.8: PLC Hardware Organization

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. Open architectures use off-the-shelf components that conform to approved standards. A system with a closed architecture is one whose design is proprietary, making it more difficult to connect the system to other systems. There are two ways in which I/O is incorporated into the PLC: fixed and modular.



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

Figure 4.9: PLC Architecture

Fixed I/O (Figure 4.10) is typical of small PLCs that come in one package with no separate removable units. The processor and I/O are packaged together and the I/O terminals are available but cannot be changed. The main advantage of this type of packaging is lower cost. The number of available I/O points varies and usually can be expanded by buying additional units of fixed I/O. One disadvantage of fixed I/O is its lack of flexibility: you are limited in what you can get in the quantities and types dictated by the packaging. Also, for some models, if any part in the unit fails the whole unit has to be replaced.

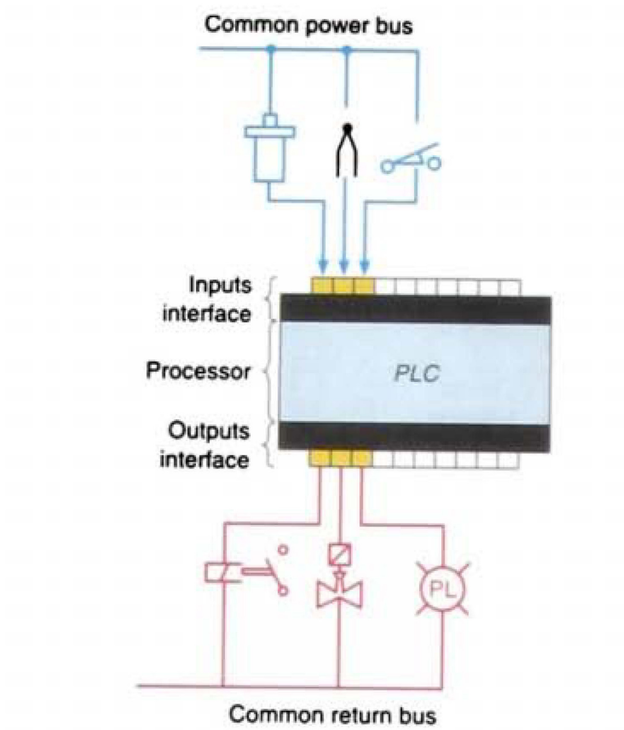


Figure 4.10: Fixed I/O

Modular I/O (Figure 4.11) is divided by compartments into which separate modules can be plugged. This feature greatly increases your options and the unit's flexibility. You can choose from the modules available from the manufacturer and mix them any way you desire. The basic modular controller consists of a rack, power supply. processor module (CPU). input/output (1/0 modules). and an operator interface for programming and monitoring. The modules plug into a rack and when a module is slid into the rack, it makes an electrical connection with a series of contacts called the backplane, located at the rear of the rack. The PLC processor is also connected to the backplane and can communicate with all the modules in the rack.

The power supply supplies dc power to other modules that plug into the rack. For large PLC systems, this power supply does not normally supply power to the field devices. With larger systems. power to field devices is provided by external alternating current (ac) or direct current (dc) supplies. For small and micro PLC systems, the power supply is used to power field devices.

The processor (CPU) is the "brain" of the PLC. A typical processor (Figure 4.12) usually

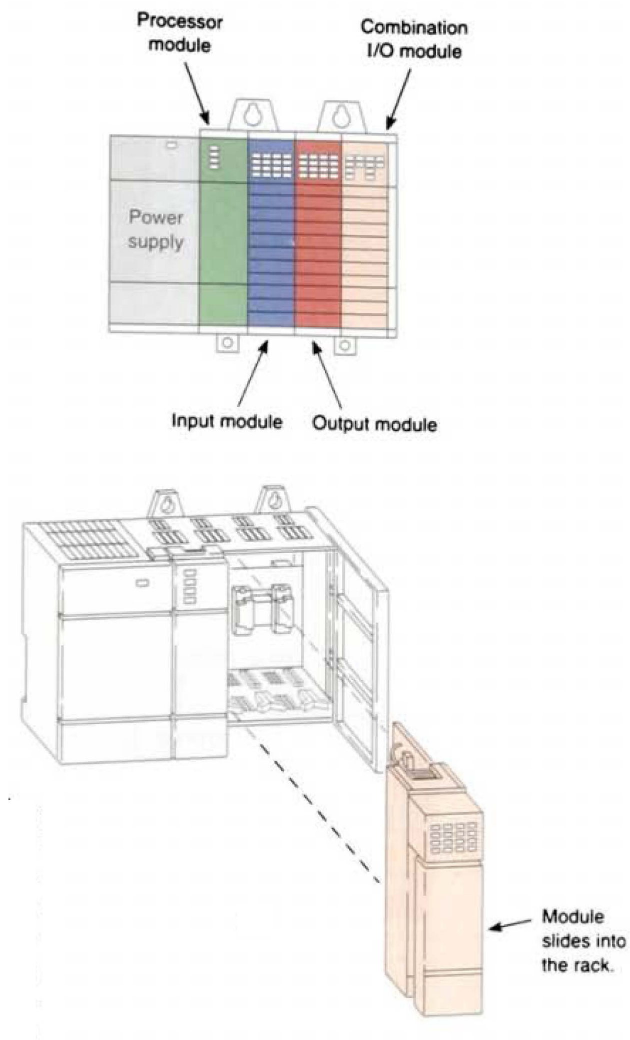


Figure 4.11: Modular I/O

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. Memory is also required for the program EPROM or EEPROM plus RAM. The CPU is designed so that the user can enter the desired circuit in relay ladder logic. The processor accepts (reads) input data from various sensing devices, executes the stored user program from memory, and sends appropriate output commands to control devices. A dc power source is required to produce the low-level voltage used by the processor. This power supply can be housed in the CPU unit or may be a separately mounted module, depending on the PLC system manufacturer.

The I/O section consists of input modules (Figure 4.13) and output modules (Figure 4.14). The I/O system forms the interface by which field devices are connected to the controller. The purpose of this interface is to condition the various signals received from or sent to external field devices. Input devices such as push buttons, limit switches. sensors. selector switches. and thumbwheel switches are hardwired to terminals on the input modules. Output devices

Figure 4.12: PLC CPU

such as small motors, motor starters, solenoid valves, and indicator lights are hardwired to the terminals on the output modules. To electrically isolate the internal components from the input and output terminals, PLCs employ an optical isolator which uses light to couple the circuits together.

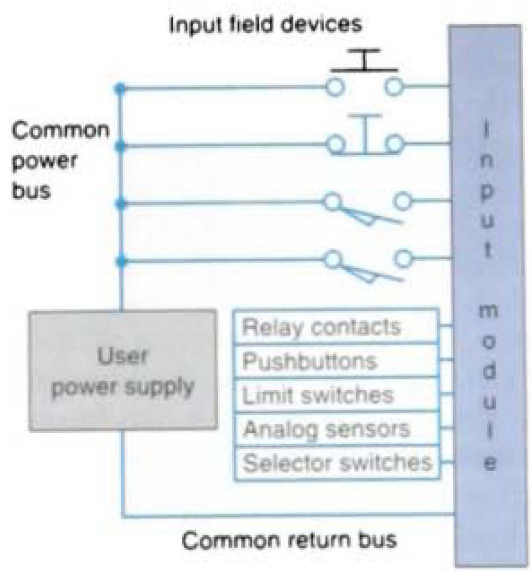


Figure 4.13: Input Module

The devices are usually defined as *field* or *real-word* inputs and outputs. The terms *field* or *real-word* are used to separate actual external devices that exist and must be physically wired from the internal user program that duplicates the function of relays, timers and counters.

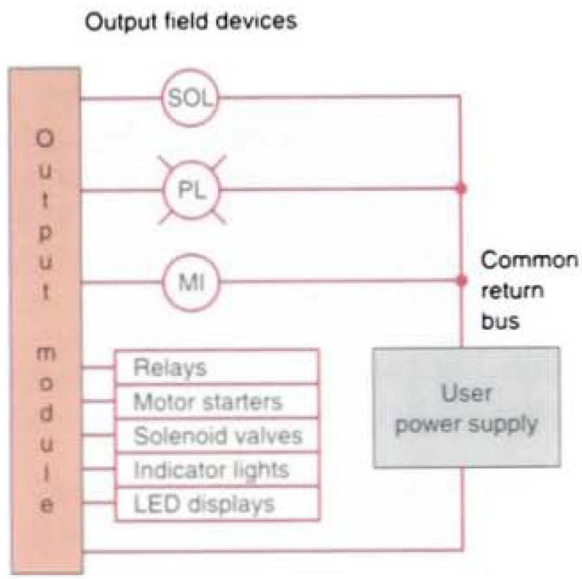


Figure 4.14: Output Module

The *programming device* or *terminal* is used to enter the desired program into the memory of the processor. This program is entered using mainly the *relay ladder logc* which is the most popular programming language used by PLCs manufacturers. As we will see in details Ladder logic uses instead of **verbal expressions** graphic symbols showing their intended outcome. It is a special language written to make it easy for people familiar with relay logic control to program the PLC. Handheld programmming devices (Figure 4.15) are sometimes used to program small PLCs because they are inexpensive and easy to use. Once plugged into the PLC they can be used to enter and supervise programs. Compact handheld units are frequently used on the factory floor for troubleshooting equipment, modifying programs and transferring programs to multiple machines. With some small handheld programming devices the program is entered using Boolean operators (AND, OR and NOT functions) individually or in combination to form logical statements.

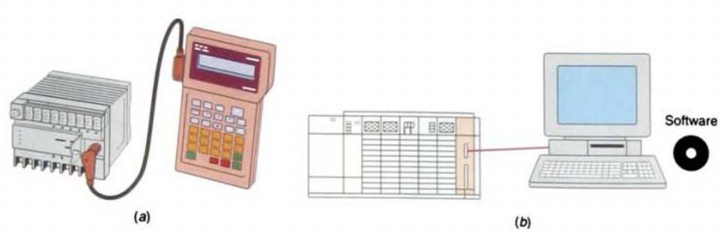


Figure 4.15: Handheld devices

A personal computer (PC) is the most commonly used programming device (please do not confuse the programming device with the PLC mainframe, they are separated entities). In general all PLC manufacturers provide proprietary software to write Ladder/SFC programs and then to code inside the PLC and eventually debug ladder logic programss. The personal computer communicates with the PLC processor via bluetooth/wi-fi facility links. If the programming unit is not in use, it may be unplugged and removed. Uncoupling the programming unit will not affect the operation of the user program.

Additional optional PLC components are often available. including:

- Operator interface devices to allow data entry and/or data monitoring by operators.
- Communication adaptors for remote I/O, so that a central controller can be connected to remote sensors and actuators.
- Network interfaces to allow interconnecting of PLCs and/or other controllers into distributed control systems.

3 How Does a PLC operate?

To get an idea of how a PLC operates, consider the simple process control problem illustrated in Figure 4.16. Here a mixer motor is to be used to automatically stir the liquid in a vat when the temperature and pressure reach preset values. In addition, direct manual operation of

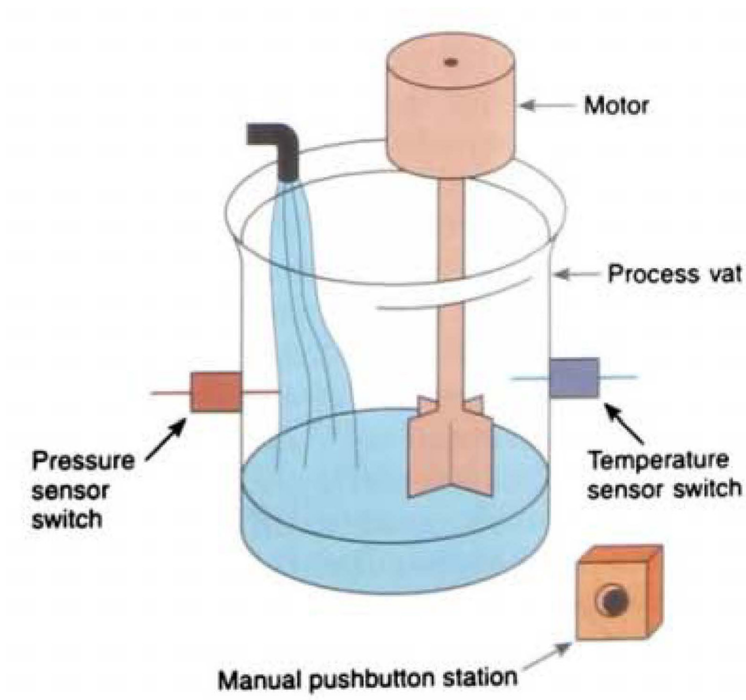


Figure 4.16: Mixer Process Control Problem

the motor is provided by means of a separate push-button station. The process is monitored with temperature and pressure sensor switches that close their respective contacts when conditions reach their preset values. This control problem can be solved using the relay method for motor control shown in the relay ladder diagram of Figure 4.17. The motor starter coil (M) is energized when both (AND) the pressure and temperature switches are closed OR when the manual push-button is pressed. Now let us look at how a PLC might

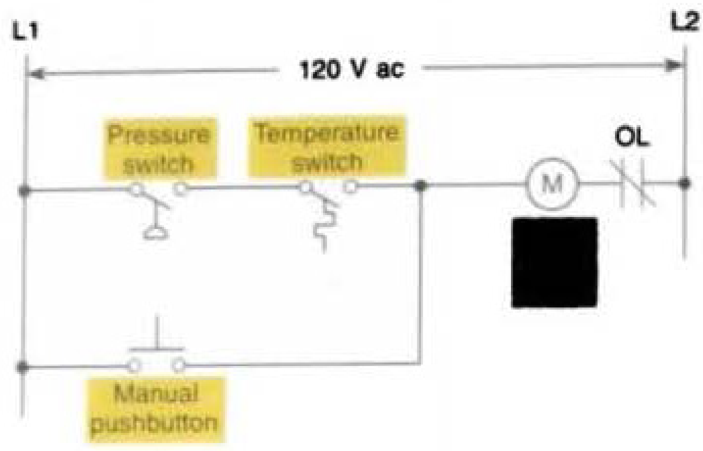


Figure 4.17: Relay diagram for the ON-OFF pressure/temperature monitoring

be used for this application. The same input field devices (pressure switch, temperature switch, and push-button) are used. These devices would be hardwired to an appropriate input module according to the manufacturer's labeling scheme. Typical wiring connections for a 120 V ac input module are shown in Figure 4.18. The same output field device (motor

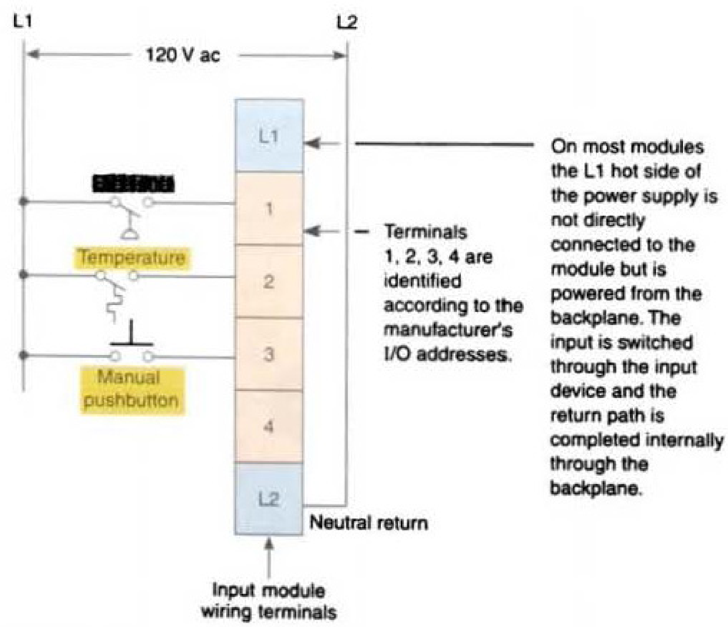


Figure 4.18: PLC Input module wiring connections

starter coil) would also be used. This device would be hardwired to an appropriate output module according to the manufacturer's labeling scheme. Typical wiring connections for a 120 V ac output module are shown in Figure 4.19 Next. the PLC ladder logic program would be constructed and entered into the memory of the CPU. A typical ladder logic program for this process is shown in Figure 4.20 The format used is similar to the layout of the hardwired relay ladder circuit. The individual symbols represent instructions, whereas the numbers represent

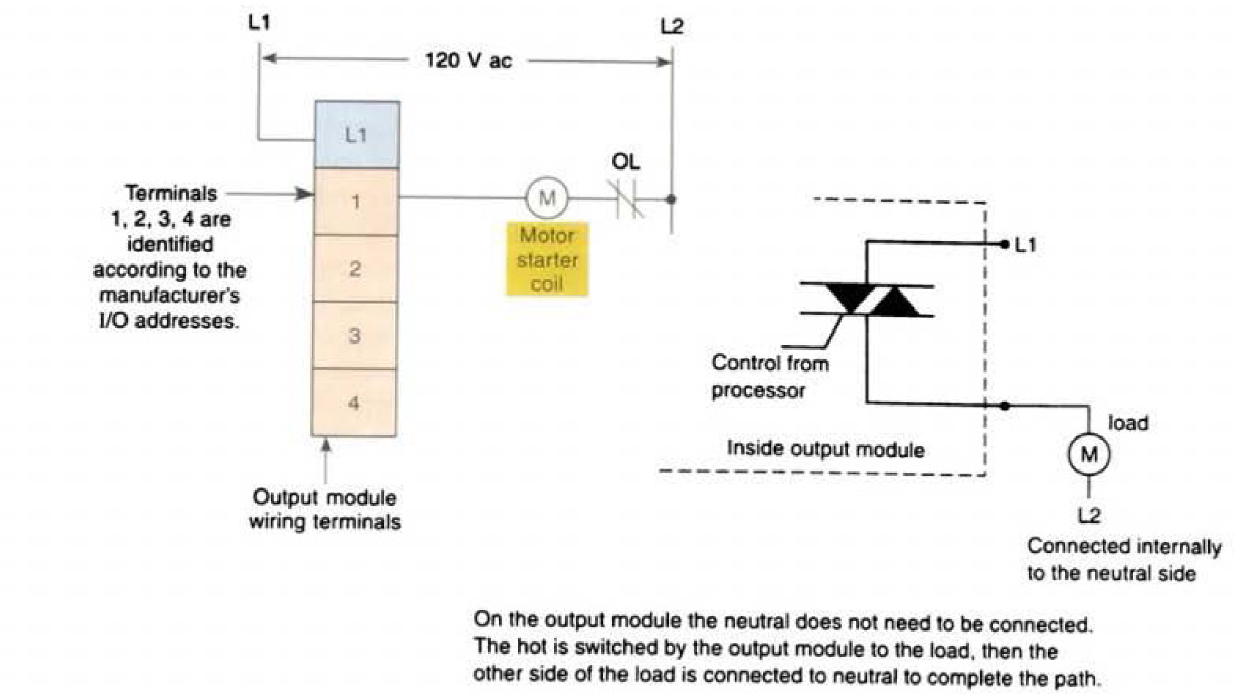


Figure 4.19: PLC Output module wiring connections

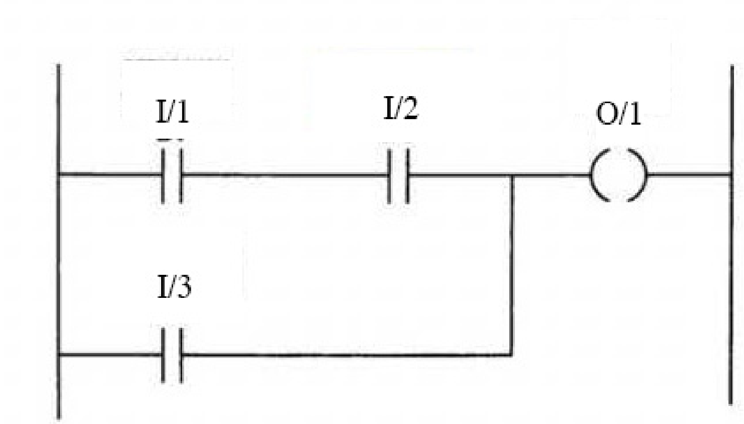


Figure 4.20: PLC Ladder Logic

the instruction addresses. To program the controller, you enter these instructions one by one into the processor memory from the programming device. Each input and output device is given an address. which lets the PLC know where they are physically connected. Note that the I/O address format may differ. depending on the PLC manufacturer. Instructions are stored in the user program portion of the processor memory. For the program to operate, the controller is placed in the RUN mode, or operating cycle. During each operating cycle, the controller examines the status of input devices, executes the user program, and changes outputs accordingly. Each -||- can be thought of as a set of normally open (NO) contacts. The -()- can be considered to represent a coil that, when energized, will close a set of contacts. In the ladder logic program of Figure the coil all is energized when contacts I/1 and I/22 are closed or when contact I/3 is closed. Either of these conditions provides a continuous path from left to right across the rung that includes the coil. The RUN operation of the controller can be described by the following sequence of events. First, the inputs are examined and their status is recorded in the controller's memory (a closed contact is recorded as a signal that is called a logic 1 and an open contact by a signal that is called a logic 0). Then the ladder diagram is evaluated. with each internal contact given OPEN or CLOSED status according to the record. If these contacts provide a current path from left to right in the diagram, the output coil memory location is given a logic 1 value and the output module interface contacts will close. If there is no conducting path on the program rung. the output coil memory location is t to logic 0 and the output module interface contacts will be open. The completion of one cycle of this sequence by the controller is called a scan. The *scan time*, the time required for one full cycle, provides a measure of the speed of response of the PLC. Generally, the output memory location is updated during the scan but the actual output is not updated until the end of the program scan during the I/O scan.

3.1 Changing the operation scenario

As mentioned. one of the important features of a PLC is the ease with which the program can be changed. For example. assume that the original process control circuit for the mixing operation must be modified as shown in the relay scheme of Figure 4.21 The change requires

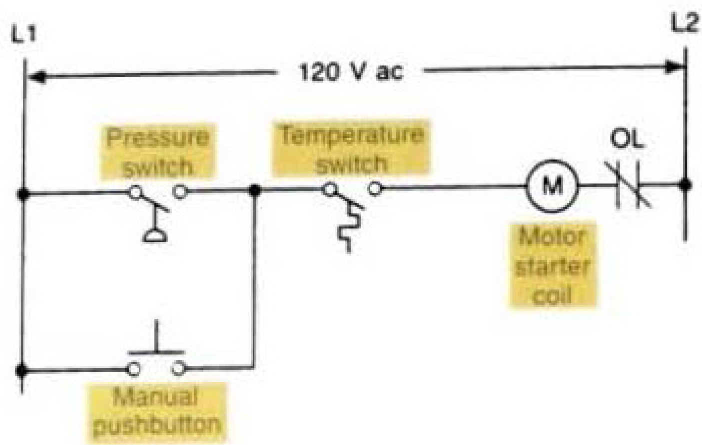


Figure 4.21: Relay rewiring

that the manual push-button control should be permitted to operate at any pressure but not unless the specified temperature setting has been reached. If a relay scheme were used it would require some *rewiring* of the system (as shown in Figure) to achieve the desired change. However 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 as shown in Figure 4.22

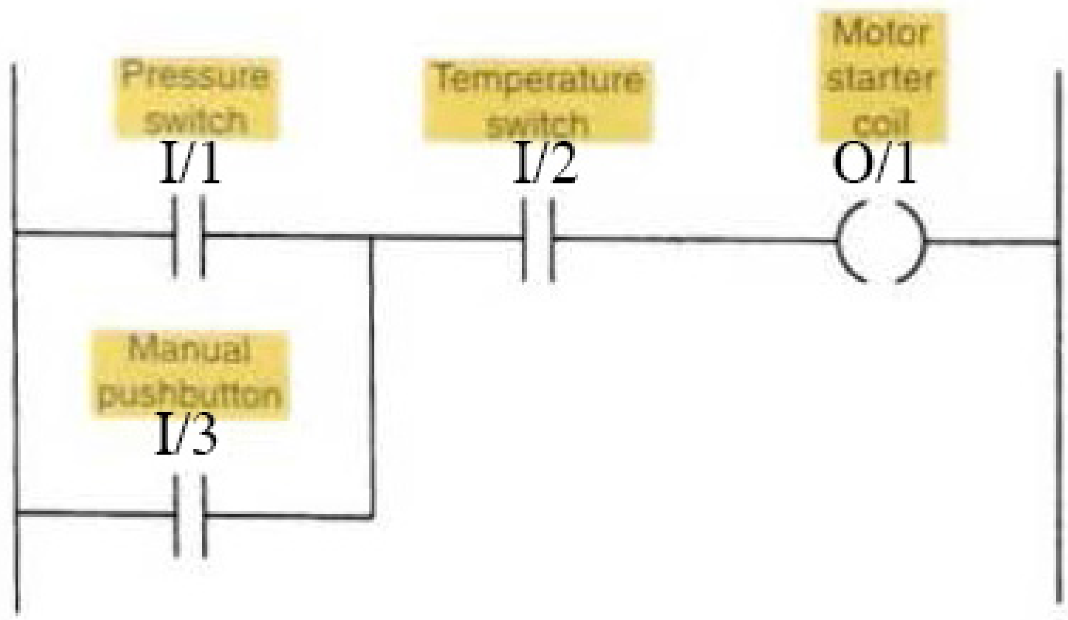


Figure 4.22: New Ladder Program