**CHAPTER ONE**

**INTRODUCTION**

**1.1 BACKGROUND HISTORY**

In today's industrial landscape, efficiency, reliability, and precision are paramount. The Supervisory Control and Data Acquisition (SCADA) process control system stands at the forefront of achieving these goals, serving as a critical component in the management and automation of complex industrial processes. By integrating Programmable Logic Controllers (PLCs), SCADA systems offer an unparalleled level of control and monitoring capabilities, ensuring seamless operation and enhanced productivity across various sectors, including manufacturing, energy, water treatment, and transportation.

SCADA systems function as the central nervous system of industrial operations, enabling real-time data collection, processing, and analysis. They facilitate the remote control of equipment and processes, thereby minimizing downtime, reducing operational costs, and improving overall system reliability. PLCs, which are robust, flexible, and scalable, play a vital role in this ecosystem. They are the backbone of local automation, interfacing with sensors and actuators to execute precise control logic and communicate crucial data back to the SCADA system.

This integration of SCADA systems with PLCs allows for the effective management of complex processes by providing a comprehensive view of operational data and control mechanisms. Through the use of advanced communication protocols and software tools, such as Visual Studio for SCADA application development, these systems can be tailored to meet specific industry requirements, offering custom solutions that enhance process efficiency and accuracy.

The following sections of this report delve deeper into the components and architecture of SCADA systems, the role and functionality of PLCs, and the development of SCADA applications using Visual Studio. By exploring these aspects, this report aims to provide a thorough understanding of the implementation and benefits of SCADA process control systems in modern industrial environments.

**1.2 AIM AND OBJECTIVE**

The aim of this project is to design, develop, and implement a robust SCADA process control system using Programmable Logic Controllers (PLCs) to enhance the monitoring, control, and automation of industrial processes. The system is intended to improve operational efficiency, ensure real-time data acquisition and analysis, and provide a scalable solution adaptable to various industrial applications.

**1.2.1 SPECIFIC OBJECTIVES**

i. System Design and Architecture

- Develop a comprehensive system architecture that integrates PLCs with a SCADA application to manage industrial processes effectively.

- Define the hardware and software requirements for the SCADA system and ensure compatibility with existing industrial infrastructure.

ii. PLC Programming and Configuration

- Program and configure PLCs to interface with sensors and actuators, enabling precise data acquisition and control of industrial processes.

- Implement control logic within the PLCs to handle automated responses and ensure seamless operation of connected equipment.

iii. SCADA Application Development

- Develop a user-friendly SCADA application using Visual Studio that provides real-time data visualization, control interfaces, and historical data logging.

- Ensure the application supports multiple communication protocols (e.g., Modbus, OPC, Ethernet/IP) for reliable data exchange between PLCs and the SCADA system.

iv. Integration and Communication

- Establish and test communication links between PLCs and the SCADA application to ensure reliable data transmission and control.

- Implement security measures to protect the communication network from cyber threats and ensure data integrity.

v. System Testing and Validation

- Conduct comprehensive testing of the SCADA system, including simulation and field tests, to validate its performance under various operational conditions.

- Identify and resolve any issues or bugs to ensure the system operates reliably and efficiently.

vi. Training and Documentation

- Develop detailed documentation covering the system architecture, PLC programming, SCADA application development, and integration procedures.

- Provide training for operators and maintenance personnel to ensure they are proficient in using and maintaining the SCADA system.

vii. Deployment and Maintenance

- Deploy the SCADA system in the target industrial environment and monitor its performance during the initial operation phase.

- Establish a maintenance schedule and provide ongoing support to address any operational issues and ensure long-term system reliability.

viii. Scalability and Future Enhancements

- Design the SCADA system with scalability in mind, allowing for easy expansion and integration of additional PLCs and sensors as needed.

- Explore and implement future enhancements, such as IoT integration, advanced data analytics, and cloud-based monitoring, to continuously improve the system’s capabilities and performance.

**1.3 SCOPE AND LIMITATION**

**Scope**

1. System Architecture and Design

- Develop a comprehensive SCADA process control system integrating PLCs to manage and automate industrial processes.

- Define the hardware and software components required, including sensors, actuators, PLCs, communication networks, and SCADA software.

2. PLC Programming and Configuration

- Program and configure PLCs to interface with various sensors and actuators, ensuring accurate data acquisition and control of processes.

- Implement control logic within PLCs to automate responses and ensure the seamless operation of industrial equipment.

3. SCADA Application Development

- Develop a SCADA application using Visual Studio, featuring real-time data visualization, control interfaces, alarm management, and historical data logging.

- Ensure the SCADA application supports multiple communication protocols (e.g., Modbus, OPC, Ethernet/IP) for reliable data exchange between PLCs and the central system.

4. Integration and Communication

- Establish robust communication links between PLCs and the SCADA system, ensuring reliable data transmission and control.

- Implement cybersecurity measures to protect the system from potential threats and ensure data integrity.

1. System Testing and Validatio - Conduct thorough testing, including simulation and field tests, to validate the performance and reliability of the SCADA system.

- Identify and resolve any issues or bugs to ensure optimal system functionality.

6. Deployment and Maintenance

- Deploy the SCADA system in the target industrial environment, providing real-time monitoring and control of processes.

- Develop a maintenance schedule and provide ongoing support to address operational issues and ensure long-term system reliability.

7. Training and Documentation

- Create detailed documentation covering system architecture, PLC programming, SCADA application development, and integration procedures.

- Provide training for operators and maintenance personnel to ensure proficient use and upkeep of the SCADA system.

8. Scalability and Future Enhancements

- Design the SCADA system to be scalable, allowing for the easy addition of PLCs and sensors to accommodate future growth.

- Explore and implement advanced features such as IoT integration, predictive maintenance, and cloud-based monitoring for continuous improvement.

**Limitations**

1. System Complexity

- The integration of multiple hardware and software components increases system complexity, requiring specialized knowledge for design, implementation, and maintenance.

2. Initial Cost

- The initial setup cost of a SCADA system, including hardware, software, and installation, can be high, which may be a barrier for smaller enterprises.

3. Cybersecurity Risks

- Despite implementing robust security measures, SCADA systems are still vulnerable to cyber attacks, which can compromise system integrity and data security.

4. Maintenance Requirements

- Regular maintenance is required to ensure the reliability and performance of the SCADA system, including updates to software and replacement of hardware components.

5.Integration Challenges

- Integrating the SCADA system with existing infrastructure and different types of PLCs and sensors can present compatibility and communication challenges.

6. Data Accuracy and Reliability

- The accuracy and reliability of data collected by the SCADA system depend on the quality and calibration of sensors and PLCs. Malfunctioning components can lead to incorrect data and control actions.

7. Training and Skill Requirements

- Effective operation and maintenance of the SCADA system require skilled personnel with specialized training, which can be a limitation for organizations with limited technical expertise.

8. Scalability Constraints

- While the system is designed to be scalable, adding new components may require significant reconfiguration and testing to ensure seamless integration and performance.

**1.4 CONTRIBUTION TO KNOWLEDGE**

Research into SCADA (Supervisory Control and Data Acquisition) process control systems significantly contributes to the body of knowledge in multiple ways, encompassing technological, operational, and strategic dimensions. Below are several key contributions:

1. Technological Advancements

A. Development of Advanced Control Algorithms

Research in SCADA systems often leads to the development of more efficient and reliable control algorithms. These algorithms optimize the operation of industrial processes, improving response times and reducing errors.

B. Enhanced Communication Protocols

Investigation into SCADA communication protocols enhances data transmission efficiency and security. Improved protocols like OPC UA, Modbus TCP/IP, and DNP3 contribute to more reliable and secure data exchange between field devices and control centers.

C. Integration with Emerging Technologies

SCADA research facilitates the integration of emerging technologies such as the Internet of Things (IoT), Artificial Intelligence (AI), and Machine Learning (ML). This integration allows for advanced predictive maintenance, anomaly detection, and data analytics, pushing the boundaries of what SCADA systems can achieve.

2. Operational Improvements

A. Increased System Reliability and Efficiency

Research efforts contribute to the development of more robust SCADA systems that enhance the reliability and efficiency of industrial processes. By identifying and mitigating potential failure points, these systems ensure continuous and efficient operations.

B. Real-Time Data Processing and Decision Making

Improving real-time data acquisition and processing capabilities allows for better decision-making and quicker response to operational issues. This is crucial in environments where timely interventions can prevent significant disruptions.

C. Enhanced User Interfaces

Research into human-machine interfaces (HMI) within SCADA systems leads to more intuitive and user-friendly control panels. Better interfaces improve operator efficiency and reduce the likelihood of human errors.

3. Strategic and Management Insights

A. Cost Reduction

Research in SCADA systems often identifies ways to reduce operational and maintenance costs. By optimizing process control and improving resource management, companies can achieve significant cost savings.

B. Cybersecurity Improvements

With the growing threat of cyber attacks, SCADA research focuses on developing advanced security measures to protect critical infrastructure. Contributions in this area help secure industrial processes against unauthorized access and cyber threats.

C. Regulatory Compliance

Research helps organizations understand and comply with regulatory requirements for safety, environmental protection, and operational standards. This ensures that SCADA systems not only improve efficiency but also adhere to legal and safety standards.

4. Academic and Educational Contributions

A. Knowledge Dissemination

Academic research in SCADA systems contributes to the dissemination of knowledge through publications, conferences, and seminars. This helps in spreading advanced concepts and innovations to a broader audience.

B. Curriculum Development

Research findings are often integrated into academic curricula, providing students with up-to-date knowledge and practical skills relevant to modern industrial automation and control systems.

C. Interdisciplinary Collaboration

SCADA research fosters collaboration across various fields such as electrical engineering, computer science, cybersecurity, and industrial engineering. This interdisciplinary approach broadens the scope and impact of research findings.

5. Industry Standards and Best Practices

A. Setting Industry Benchmarks

Research in SCADA systems helps establish industry standards and best practices. These benchmarks guide companies in implementing effective and reliable SCADA solutions.

B. Case Studies and Lessons Learned

Documenting and analyzing case studies from SCADA implementations provides valuable lessons and best practices. This information helps other organizations in planning and executing their SCADA projects more effectively.

**1.5 AREAS OF APPLICATION**

SCADA (Supervisory Control and Data Acquisition) process control systems are essential for monitoring and controlling industrial processes across a wide range of sectors. Their ability to collect real-time data, analyze it, and execute control actions makes them indispensable in many applications. Below are key areas where SCADA systems are applied:

i. Manufacturing and Production

A. Assembly Lines

- Monitoring and controlling the operations of assembly lines.

- Ensuring consistent quality and detecting faults in real-time.

B. Process Industries

- Managing chemical, pharmaceutical, food and beverage production processes.

- Controlling temperature, pressure, and flow rates to maintain optimal conditions.

ii. Energy and Utilities

A. Power Generation

- Monitoring and controlling operations in power plants (thermal, hydro, nuclear).

- Managing electrical grids to ensure stability and efficiency.

B. Oil and Gas

- Overseeing exploration, drilling, and production operations.

- Monitoring pipeline pressures, flow rates, and detecting leaks.

C. Renewable Energy

- Managing wind farms, solar power plants, and hydroelectric facilities.

- Optimizing energy production and integration into the grid.

iii. Water and Wastewater Management

A. Water Treatment Plants

- Monitoring and controlling water purification processes.

- Ensuring safe water quality by managing chemical dosing and filtration.

B. Wastewater Treatment

- Overseeing wastewater treatment processes.

- Controlling aeration, sedimentation, and disinfection stages.

C. Distribution Networks

- Managing water supply networks to ensure consistent delivery.

- Detecting leaks and managing pressure in pipelines.

iv. Transportation

A. Railways

- Monitoring and controlling railway signaling systems.

- Managing train schedules and ensuring passenger safety.

B. Highways and Traffic Management

- Overseeing traffic light systems and road condition monitoring.

- Managing toll collection and traffic flow.

C. Airports

- Managing runway lighting, baggage handling systems, and environmental controls.

- Monitoring fuel supply and aircraft movements.

v. Building and Facilities Management

A. Smart Buildings

- Controlling HVAC (Heating, Ventilation, and Air Conditioning) systems.

- Managing lighting, security, and energy consumption.

B. Industrial Facilities

- Overseeing operations of large industrial complexes.

- Monitoring environmental conditions and optimizing resource usage.

vi. Telecommunications

A. Network Management

- Monitoring and controlling telecom networks.

- Ensuring uptime and performance of communication systems.

B. Data Centers

- Managing power, cooling, and security systems in data centers.

- Monitoring server performance and environmental conditions.

vii. Agriculture

A. Precision Farming

- Monitoring soil moisture, weather conditions, and crop health.

- Controlling irrigation systems to optimize water usage.

B. Livestock Management

- Monitoring the health and productivity of livestock.

- Controlling feeding systems and environmental conditions in barns.

viii. Environmental Monitoring

A. Air Quality Control

- Monitoring and controlling air pollution levels in urban and industrial areas.

- Ensuring compliance with environmental regulations.

B. Weather Stations

- Collecting and analyzing weather data.

- Providing real-time data for weather forecasting and disaster management.

ix. Mining and Metals

A. Mineral Extraction

- Monitoring and controlling extraction processes.

- Managing equipment and ensuring safety in mines.

B. Smelting and Refining

- Overseeing the smelting and refining processes.

- Controlling temperature and chemical composition to ensure product quality.

x. Healthcare

A. Hospital Facilities Management

- Controlling HVAC systems, lighting, and security in healthcare facilities.

- Monitoring critical systems like power supply and medical gas distribution.

B. Pharmaceutical Manufacturing

- Ensuring the consistency and quality of pharmaceutical products.

- Controlling environmental conditions in cleanrooms and production areas.

**CHAPTER TWO**

**LITERATURE REVIEW**

**2.1 Review of past related work**

**2.1.1 Design of Low Cost Multi Channel Data Acquisition System for Meteorological Application BY: Nisha Kashyap, Department of Electronics and Communication Engineering National Institute of Technology, Rourkela, Odisha May 2015.**

The primary objective of the present work is to design a low cost multi-channel data acquisition system which can be used for meteorological application. The proposed multi-channel data acquisition system acquires ambient temperature, barometric pressure, altitude, humidity and light intensity data from environment and stores the data for future use. The sensors are connected with a low-cost microcontroller (ATmega328) unit which performs the data acquisition and data logging operation. A real time clock is used to keep current time with the measurement. Different communication interfaces such as serial communication, wireless communication and Ethernet communication are used in the data acquisition system. In serial communication, the sensor data are logged in COM port of PC. LabVIEW based application is developed which provides graphical user interface for the user. VISA protocol is used to communicate the COM port data with LabVIEW. Other communication protocol such as wireless communication and Ethernet communication protocol is used to transmit the sensor data over a communication channel. The data is uploaded in Ethernet which can be viewed using a web browser. In wireless communication Xbee transmitter and Xbee receiver modules are used to transmit data over a longer distance. The salient features of this developed system is that the system is low cost, uses open source software’s like Arduino and python and the system is economical (that is - INR 6500). This developed system performs satisfactorily under different condition. The system is tested for 24 hour in April 2015 and provides satisfactory results.

**2.1.2 Bluetooth Based Data Acquisition System BY: Suresh Gurjar, Department of Electrical Engineering National Institute of Technology Rourkela Rourkela 769008 INDIA. May 2015**

Data acquisition systems are devices used to collect information to document or analyse some physical phenomenon such as voltage, force or temperature. Data acquisition systems available in the market are very expensive, bulky and power hungry. However, PC based data acquisition system offers a lot of benefit in terms of processing speed, display resolution and connectivity capabilities. The Project aims at designing and implementing a portable, economical and power efficient real-time data acquisition system. The proposed system comprises of a hardware circuitry and a Graphical User Interface (GUI) based on MATLAB environment. The hardware device consists of an 8-bit microcontroller interfaced with a serial ADC chip and a Bluetooth serial module. The Bluetooth HC-05 module is used to provide a wireless connectivity between the hardware and the PC. For testing purpose, the sampling rate of ADC is set to 833 Hz, capturing 50 values per 0.06 second. Whereas on the PC side, the GUI receives the sampled values transmitted by the hardware device and plots the real-time signal waveform. It has been found that the GUI plots the signal waveform with good quality and efficiency. The proposed system can be deployed in number of industrial applications such as remote device controlling and ECG data acquisition with some adjustments in the hardware. The wireless connectivity reduces the complexity of cables and probability of occurrences of the accidents in industrial areas.

**2.1.4 Real-Time Measurement And Monitoring System Using Zigbee And Labview By: Muhammad Hairul Bin Yahaya, Faculty of Electronic and Computer Engineering Universiti Teknikal Malaysia Melaka JUNE 2015.**

Wireless communication technologies become popular in real-time industrial environments. This project describes the fundamental work of real time measurement and monitoring system with Arduino, ZigBee, and Graphical Programming language (LabVIEW). Thickness of the item is taken as a parameter that will convert into the number of items. By acquiring the analogue data from the ultrasonic sensor, the suitable signal conditional is fed to Arduino. It is connected to the ZigBee transmitter module, which transmits the data to the ZigBee receiver. As a result, the data received will be viewed on a PC via LabVIEW. The methodology describe in the proposed work is useful for designing Wireless Sensor Network (WSN) system in the industrial application such as real levelling storage. The experimental of transmission range and observation of data required in LabVIEW was reported. The combination of measurement, wireless transmission and monitoring system can be improved of accuracy of measurement, especially for sustainability of better working place.

**2.1.5 Principles and Applications of Computerized Process Control in the Department Of Chemical Engineering, Rivers State University Of Science And Technology,**

**Port Harcourt, Nigeria. by DONGESIT EFFIONG SAMPSON (2016).**

Automatic process control, supervised control and data acquisition, integration of information systems or enterprise resource integration (ERI) activities are widely applied by process industries to remain competitive

and improve efficiency and profit margins. Computerized process control is more efficient than manual control as it

safes time and effort and with the digital inclusion new technologies that enhance the Nigerian vision for digital

human capital in the 21stCentury are introduced. In Nigeria, most industries use control systems which lack the features necessary for effective communication and efficient operation, thus imposing a limitation on the use of the system to achieve high operational efficiency, high quality product, yields and profit. An efficient use of a good process control computer system enhances efficiency, increased production rate and reduced operating cost

**2.1.6 A Simplified Approach for Using PLC and SCADA System in 330 kV distribution Substations in Nigeria by Ngang Bassey Ngang Anthony Lordson Amana UkoOfe**

**(2018).**

Introduction of SCADA system in Generation, Transmission and Distribution sectors of our electricity industry has changed the traditional way of doing our routine work in the industries.SCADA the substations and sends control signals to the remote control equipment on a continuous basis. It gets the historical data of the substation and generates the alarms in the event of electrical accidents or faults.Most substations now utilize the SCAD system to monitor measurements, of relevant parameters for correction and regulation of end devices.Certain measurements, supervision,control, operation and protection functions are performed using the technology The problem of not controlling power substation properly by the conventional approach has made someareas not to have power supply as at the time they are meant to have it. That is the reason SCADA control of power substation applying programmable Logic Controllers is now adopted by power systems Engineers. It has improved the overall efficiency of operating power system components. The design is done by taking cognizance of the fact that a membership function for SCADA control is required. You can now proceed by designing a SCADA rule for controlling of power substation, training these rules in ANN, designing a visual basic model for SCADA control of power substation using PLC and designing a Simulink model for SCADA control of power substation using PLC. The result obtained when compared to the conventional one was 10% better.

**2.2 Block diagram of the SCADA Process control system**

PSU

PLC

FCE

COM Driver

HMI

FLOWRATE

SENSOR

INDICATORS AND ALARM SYSTEM

LEVEL

SENSOR

1. Power Supply Unit (PSU): The power supply unit converts the incoming electrical power supply (AC or DC) to the appropriate voltage and current required by the PLC and other components in the SCADA system.
2. PLC Unit: The PLC unit is the central processing unit of the SCADA system. It contains the CPU, memory, and interfaces for communication with other devices. It receives input signals from the I/O modules, processes them using the program stored in its memory, and generates output signals to control the connected devices.
3. COM Driver: These modules interface with the external world and connect to the PLC unit. They receive input signals from sensors and switches, and convert them into a format that the PLC can understand. Similarly, they take output signals from the PLC and convert them into signals suitable for controlling the actuators and other devices.
4. Human Machine Interface (HMI) Device: The HMI device provides a graphical user interface for monitoring and interacting with the PLC. It typically consists of a display screen, buttons, and/or a touchscreen. The HMI allows users to view and modify system parameters, monitor the status of inputs and outputs, and interact with the PLC program.
5. FEC(Final Control Element): These are the devices that interact with the physical world. Sensors detect physical variables such as temperature, pressure, or proximity and convert them into electrical signals. Actuators, on the other hand, receive electrical signals from the PLC and perform physical actions, such as moving a motor or turning on/off a valve.

The components in a PLC SCADA control system are interconnected using appropriate cables and connectors to establish the necessary communication and power connections.

## 2.3 Component Review of PLC based SCADA system

**2.3.1 PLC**

PLC stands for “Programmable Logic Controller”. A PLC is a computer specially designed to operate reliably under harsh industrial environments – such as extreme temperatures, wet, dry, and/or dusty conditions. PLCs are used to automate industrial processes such as a manufacturing plant’s assembly line, an ore processing plant, or a wastewater treatment plant.

PLCs share many features of the personal computer you have at home. They both have a power supply, a CPU (Central Processing Unit), inputs and outputs (I/O), memory, and operating software (although it’s a different operating software).

The biggest differences are that a PLC can perform discrete and continuous functions that a PC cannot do, and a PLC is much better suited to rough industrial environments. A PLC can be thought of as a ‘ruggedized’ digital computer that manages the electromechanical processes of an industrial environment.

PLCs play a crucial role in the field of automation, using forming part of a larger SCADA system. A PLC can be programmed according to the operational requirement of the process. In the manufacturing industry, there will be a need for reprogramming due to the change in the nature of production. To overcome this difficulty, PLC-based [control systems](https://www.electrical4u.com/control-system-closed-loop-open-loop-control-system/) were introduced. We’ll first discuss PLC basics before looking at various applications of PLCs.

## 2.3.2 BASICS OF PROGRAMMABLE LOGIC CONTROLLER

PLCs were invented by Dick Morley in 1964. Since then PLC has revolutionized the industrial and manufacturing sectors. There is a wide range of PLC functions like timing, counting, calculating, comparing, and processing various analog signals. The main advantage of PLC over a “hard-wired” control system is that you can go back and change a PLC after you’ve programmed it, at little cost

## 2.3.3 PHYSICAL STRUCTURE OF PROGRAMMABLE LOGIC CONTROLLER

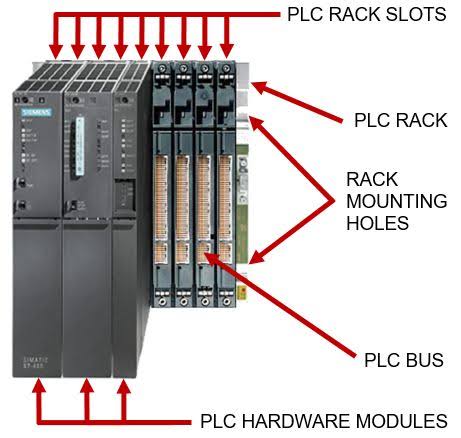
Programmable Logic Controllers continuously monitors the input values from various input sensing devices (e.g. accelerometer, weight scale, hardwired signals, etc.) and produces corresponding output depending on the nature of production and industry. A typical block diagram of PLC consists of five parts namely:

1. Rack or chassis
2. Power Supply Module
3. Central Processing Unit (CPU)
4. Input & Output Module
5. Communication Interface Module

### Rack or Chassis

In all PLC systems, the PLC rack or chassis forms the most important module and acts as a backbone to the system. PLCs are available in different shapes and sizes. When more complex control systems are involved, it requires larger PLC racks.

Small-sized PLC is equipped with a fixed I/O pin configuration. So, they have gone for modular type rack PLC, which accepts different types of I/O modules with sliding and fit in concept. All I/O modules will be residing inside this rack/chassis.



**PICTURES OF PLC RACK OR CHASSIS**

### Power Supply Module

This module is used to provide the required power to the whole PLC system. It converts the available AC power to DC power which is required by the CPU and I/O module. PLC generally works on a 24V DC supply. Few PLC uses an isolated power supply.

### CPU Module and Memory

CPU module has a central processor, ROM & RAM memory. ROM memory includes an operating system, drivers, and application programs. RAM memory is used to store programs and data. CPU is the brain of PLC with an [octal](https://www.electrical4u.com/binary-to-octal-and-octal-to-binary-conversion/) or hexagonal microprocessor.

Being a microprocessor-based CPU, it replaces timers, relays, and counters. Two types of processors as a single bit or word processor can be incorporated with a PLC. One bit processor is used to perform logic functions. Whereas word processors are used for processing text, numerical data, controlling, and recording data.

CPU reads the input data from sensors, processes it, and finally sends the command to controlling devices. DC power source, as mentioned in the previous discussion is required voltage signals. CPU also contains other electrical parts to connect cables used by other units.

### Input and Output Module

Have you ever thought about how to sense physical parameters like temperature, pressure, flow, etc? using PLC? Of course, PLC has an exclusive module for interfacing inputs and output, which is called an input & output module.

Input devices can be either start and stop pushbuttons, switches, etc and output devices can be an electric heater, valves, relays, etc. I/O module helps to interface input and output devices with a microprocessor.

## 2.3.4 TYPES OF PROGRAMMABLE LOGIC CONTROLLER

The two main types of PLC are fixed / compact PLC and modular PLC.

### Compact PLC

Within a single case, there would be many modules. It has a fixed number of I/O modules and external I/O cards. So, it does not have the capability to expand the modules. Every input and output would be decided by the manufacturer.

### Modular PLC

This type of PLC permits multiple expansion through “modules”, hence referred to as Modular PLC. I/O components can be increased. It is easier to use because each component is independent of each other.

PLC are divided into three types based on output namely Relay output, Transistor output, and Triac Output PLC. The relay output type is best suited for both AC and DC output devices. Transistor output type PLC uses switching operations and used inside microprocessors.

According to the physical size, a PLC is divided into Mini, Micro, and Nano PLC. Some of the manufacturers of PLCs include:

1. [Allen Bradley](https://ab.rockwellautomation.com/Programmable-Controllers/ControlLogix/5580-Controllers)
2. [ABB](https://new.abb.com/plc/programmable-logic-controllers-plcs)
3. [Siemens](https://new.siemens.com/global/en/products/automation/systems/industrial/plc.html)
4. [Mitsubishi PLC](https://www.mitsubishielectric.com/fa/products/cnt/plc/index.html)
5. [Hitachi PLC](https://www.hitachi-ies.co.jp/english/products/plc/index.htm)
6. [Delta PLC](https://www.deltaww.com/Products/CategoryListT1.aspx?CID=060301&PID=ALL&hl=en-US)
7. [General Electric (GE) PLC](http://www.geautomation.com/products/programmable-automation-controllers%EF%BB%BF)
8. [Honeywell PLC](https://www.honeywellprocess.com/en-US/explore/products/control-monitoring-and-safety-systems/scalable-control-solutions/Pages/masterlogic-plc.aspx)

## 2.3.5 PROGRAMMABLE LOGIC CONTROLLER PROGRAMMING

When using a PLC, it’s important to design and implement concepts depending on your particular use case. To do this we first need to know more about the specifics of PLC programming.

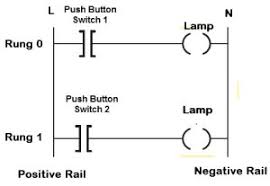
A PLC program consists of a set of instructions either in textual or graphical form, which represents the logic that governs the process the PLC is controlling. There are two main classifications of PLC programming languages, which are further divided into many sub-classified types.

1. **Textual Language**
   1. Instruction list
   2. Structured text
2. **Graphical Form**
   1. Ladder Diagrams (LD) (i.e. Ladder Logic)
   2. [Function Block Diagram](https://www.plcacademy.com/function-block-diagram-programming/) (FBD)
   3. Sequential Function Chart (SFC)

Although all of these PLC programming languages can be used to program a PLC, graphical languages (like ladder logic) are typically preferred to textual languages (like [structured text programming](https://www.plcacademy.com/structured-text-tutorial/)).

### Ladder Logic

[Ladder logic](https://www.plcacademy.com/ladder-logic-tutorial/) is the simplest form of PLC programming. Ladder logic has evolved into a [programming language](https://en.wikipedia.org/wiki/Programming_language" \o "Programming language) that represents a program by a graphical diagram based on the [circuit diagrams](https://en.wikipedia.org/wiki/Circuit_diagram" \o "Circuit diagram) of [relay logic](https://en.wikipedia.org/wiki/Relay_logic" \o "Relay logic) hardware. Ladder logic is used to develop software for [programmable logic controllers](https://en.wikipedia.org/wiki/Programmable_logic_controller" \o "Programmable logic controller) (PLCs) used in industrial control applications. The name is based on the observation that programs in this language resemble [ladders](https://en.wikipedia.org/wiki/Ladder" \o "Ladder), with two vertical rails and a series of horizontal rungs between themIt is also known as “relay logic”. The relay contacts used in relay controlled systems are represented using ladder logic (Edward, 2019). The below figure shows a simple example of a ladder diagram.



PLC Ladder Logic

In the above-mentioned example, two pushbuttons are used to control the same lamp load. When any one of the switches is closed, the lamp will glow.

The two horizontal lines are called rungs and the two vertical lines are called rails. Every rung forms the electrical connectivity between Positive rail (P) and Negative rail (N). This allows the [current](https://www.electrical4u.com/electric-current-and-theory-of-electricity/) to flow between input and output devices.

## 2.3.6 HISTORICAL BACKGROUND OF PLCS

Many early PLCs were not capable of graphical representation of the logic, and so it was instead represented as a series of logical expressions in a Boolean format (akin to [Boolean algebra](https://www.electrical4u.com/boolean-algebra-theorems-and-laws-of-boolean-algebra/)).

As programming terminals evolved, it became more common for ladder logic to be used, because it was a familiar format used for electro-mechanical control panels. More modern formats, such as state logic and Function Block diagrams exist, but they are still not as popular as ladder logic (Walker, 2012).

A possible reason for this is that programmers prefer the more visual appeal of ladder logic over structured text programming.

Until approximately the mid-1990s, PLCs were programmed using proprietary programming panels or special-purpose programming terminals, which often had dedicated function keys representing the various logical elements of PLC programs (Walker, 2012).

Some proprietary programming terminals displayed the elements of PLC programs as graphic symbols, but plain [ASCII code](https://www.electrical4u.com/alphanumeric-codes-ascii-code-ebcdic-code-unicode/) representations of contacts, coils, and wires were common.