DEVELOPMENT OF SPEED CONTROL SYSTEM OF AN INDUCTION MOTOR USING PID CONTROLLER

A Thesis
By

Md Mottalib Hossain- ME2103025140
Jahidul islam- ME2103025393
Md Asik Sarkar- ME2103025139
Sudip Talukder Rikon- ME2103025211
Jewel Rahman- ME2103025180



Department of Mechanical Engineering
Sonargaon University (SU)
Dhaka, Bangladesh
May 2025

DEVELOPMENT OF SPEED CONTROL SYSTEM OF AN INDUCTION MOTOR USING PID CONTROLLER

A Thesis report Submitted to the Department of Mechanical Engineering for the partial fulfillment of the award of degree of “Bachelor of Science in Mechanical Engineering”.

Submitted By:

Md Mottalib Hossain
ME2103025140

Jahidul islam
ME2103025393

Md Asik Sarkar
ME2103025139

Sudip Talukder Rikon
ME2103025211

Jewel Rahman
ME2103025180

Supervised By:

Hasan Tareq Mahin
Lecturer
Department of Mechanical Engineering
Sonargaon University (SU)
Dhaka,Bangladesh

Acknowledgement

We would like to express all appreciation and gratitude to almighty Allah for giving us the chance and Capability to undertake successfully Complete this project for our study.

We are immensely grateful to our honorable supervisor **Hasan Tareq Mahin** whose guidance and expertise have been invaluable throughout this journey. Their insightful feedback, constant encouragement, and patience have truly shaped the direction of this project.

I would also like to extend my thanks to my Classmates who have provided assistance and support along the way.Their constructive discussions, brainstorming sessions, and willingness to share their knowledge have significantly enriched my understanding of the subject matter.

Furthermore, I am indebted to the participants who generously contributed their time and insights to the research conducted in this project. Their valuable input has been crucial in shaping the findings and conclusions.

In conclusion, I am sincerely thankful to all those mentioned above and anyone else who has contributed to this project in any way. Your support has been instrumental in making this endeavor a success.

Abstract

This project focuses on the design and implementation of a motor speed control system using Siemens automation components, including a Programmable Logic Controller (PLC), Human Machine Interface (HMI), Variable Frequency Drive (VFD), induction motor, and rotary encoder. The primary objective is to achieve precise and flexible control of motor RPM to simulate real-world industrial applications. The system allows users to input desired speed values through the HMI, which are processed by the PLC to control the VFD accordingly. The rotary encoder provides real-time feedback on the actual motor speed, enabling closed-loop control and ensuring accuracy. This training project serves as a practical platform for understanding industrial automation concepts, PLC programming, and motor control techniques. It equips trainees with the skills necessary for handling automation tasks in manufacturing and process industries.

Table of Contents

	Page No
Acknowledgement	i
Abstract	ii
Table of contents	iii
List of Figures	iv
List of Abbreviations	iv
Chapter 1 Introduction	
1.1 General	1
1.2 Objectives	5
Chapter 2 Literature Review	
2.1 Background	6
Chapter 3 Hardware Implementation	
3.1 Induction Motor	11
3.2 Variable Frequency Drive	12
3.3 Rotary Encoder	15
3.4 Programmable Logic Controller	17
3.5 Human Machine Interface	18
3.6 Banana Cable	20
3.7 Pushbutton Switch	21
3.8 Programing Device	22
3.9 Power Supply	23
Chapter 4 Methodology	
4.1 Working Process	24
4.2 Electrical Cycle	27
4.3 Schematic Views Of the Project	31

Chapter 5 Results and Discussion

5.1 Table 35
5.2 RPM and Time Graph 37
5.3 Results Analysis 41
5.4 Discussion 43

Chapter 6 Advantage and Limitations

6.1 Advantages 46
6.2 Limitations 46
6.3 Applications 47
6.4 Conclusion 48

Reference 50

List of figures

3.1 Induction Motor 11
3.2 Variable Frequency Drive 12
3.3 Rotary Encoder 15
3.4 Programmable Logic Controller 17
3.5 Human Machine Interface 18
3.6 Banana Cable 20
3.7 Pushbutton Switch 21
3.8 Programing Device 22
3.9 Power Supply 23
4.2 Electrical Circuit 27
4.3 Schematic Views of the Project 31

List of Abbreviations

VFD	Variable Frequency Drive
HMI	Human Machine Interface
PLC	Programmable Logic Controller
PID	Proportional Integral Derivative
MS	Milli Seconds
RPM	Revolutions Per Minute

Chapter 1
Introduction

1.1 General

The induction motor was invented by Nikola Tesla. He developed the basic idea of this motor in 1887 and received a patent for it in 1888. However, at the same time, an Italian scientist named Galileo Ferraris was also working on a similar idea of an induction motor and published it around the same time. But Nikola Tesla is better known and recognized in terms of patents and commercial production.

Induction motors work primarily based on Faraday's electromagnetic induction and Lenz's law. Induction motor speed control can be achieved using several methods, depending on whether you're working with a squirrel cage or a wound rotor motor, and whether the control needs to be precise, efficient, or economical. Here's an overview of common methods:

1. Stator Voltage Control (for small motors)
Principle: Stator voltage control for small motors works by using electronic switches (like triacs or thyristors) to reduce the RMS voltage supplied to the motor's stator windings. Since the motor's torque is strongly linked to the square of this voltage, lowering the voltage directly results in a significant drop in the motor's output torque. This allows for speed reduction in applications where less torque is needed at lower speeds, such as fans and pumps.

Effect: Reducing the stator voltage in small motors directly lowers the motor's torque (proportional to the voltage squared). This causes the motor to slow down until its reduced torque matches the load's requirement, making it suitable for speed control in applications like fans and pumps.

Used for: tator voltage control for small motors is used for speed control in applications like fans and pumps. These loads require less torque at lower speeds, and reducing the stator

voltage effectively lowers the motor's torque and thus its speed, offering a simple and cost-effective control method.

2. Frequency Control (V/f Control or Variable Frequency Drive - VFD)

Principle: Frequency control (V/F control) for induction motors works by simultaneously adjusting both the frequency and the voltage supplied to the motor, while maintaining a constant ratio between them (Voltage/Frequency = Constant).

The key idea is to keep the magnetic flux in the motor at a relatively constant level. If only the frequency were changed, the flux would vary, leading to either saturation (if frequency decreases) or weak torque (if frequency increases). By proportionally changing the voltage with the frequency, the air gap flux remains approximately constant, allowing for smoother and more efficient speed control across a wider range.

Effect: Frequency control (V/F control) changes an AC motor's speed by adjusting the supply frequency. By also proportionally changing the voltage (keeping V/F constant), it maintains relatively consistent torque capability across a range of speeds, allowing for efficient operation in variable speed applications like fans and pumps.

Used for: Fans and Pumps: This is a very common application because the required torque decreases significantly at lower speeds, aligning well with the V/F control's ability to maintain relatively constant torque capability across a wide speed range. It allows for efficient flow control.

Conveyors: V/F control enables smooth starting and stopping, adjustable speed for varying material flow rates, and energy savings by running the conveyor at the required speed.

General Industrial Machinery: It's used in a wide range of machinery where adjustable speed is beneficial, such as mixers, crushers, and some types of machine tools where precise torque control isn't the primary requirement.

3. Pole Changing (for wound stator motors)

Principle: Pole changing for a wound stator motor achieves speed variation by physically reconfiguring the stator winding connections to change the number of magnetic poles produced by the stator.

The synchronous speed of an AC induction motor is inversely proportional to the number of poles ($N_s = 120f/P$). By rewiring the stator coils, the number of effective magnetic poles (P) can be altered. This results in a step change in the motor's synchronous speed.
For a wound stator motor, specific winding arrangements are designed during manufacturing to allow for these reconnections, often providing two or more distinct speeds. Switching between these pre-configured pole numbers is typically done using external switches or contactors.

Effect: Pole changing in a wound stator motor directly results in a stepped change in the motor's synchronous speed. By reconfiguring the stator windings to alter the number of magnetic poles, the motor operates at a different fixed speed. More poles lead to lower speed, and fewer poles result in higher speed. This provides discrete speed options rather than continuous variation.

Used for: Fans: To provide different airflow settings (e.g., low, medium, high). Pumps: To adjust the flow rate in stages.
Hoists and Cranes: To offer different lifting or travel speeds (e.g., fast for movement, slow for precise positioning).

4. Rotor Resistance Control (for wound rotor motors)
Principle: Rotor resistance control for a wound rotor motor works by introducing external resistance into the rotor circuit via the slip rings.
The key principle is that increasing the rotor resistance:

- Increases the slip at which maximum torque occurs.
- Increases the starting torque of the motor.
- For a given load torque, it causes the motor to run at a lower speed.

By varying the external resistance (typically using a rheostat), the motor's torque-speed characteristic is altered, allowing for speed control below the synchronous speed. As the motor speeds up, this external resistance can be gradually reduced or even shorted out for normal running with higher efficiency.

Effect: Rotor resistance control in a wound rotor motor primarily results in speed reduction. By adding external resistance to the rotor circuit, the motor develops maximum torque at a higher slip,

causing it to run slower for a given load. It also allows for higher starting torque with lower starting current. However, it leads to reduced efficiency at lower speeds due to power dissipation in the external resistors.

Used for: High Starting Torque: Adding resistance to the rotor circuit increases the starting torque, making it suitable for heavy loads that are difficult to start, such as some hoists, cranes, and heavily loaded conveyors.

Stepped Speed Control: By varying the external resistance in steps, a few discrete lower speeds can be achieved. This is utilized in applications like hoists and some industrial machinery where specific operating speeds are needed for different tasks.

Smooth Acceleration: The ability to gradually reduce the external resistance as the motor speeds up allows for smoother acceleration of large inertia loads.

It's important to note that while it offers high starting torque and some speed control, rotor resistance control is inefficient at lower speeds due to power loss in the external resistors and is only applicable to wound rotor motors. Modern applications often favor more efficient and versatile methods like Variable Frequency Drives (VFDs) when continuous speed control is required.

It is possible to control the speed of the motor by adopting the above process. However, in that case, it takes a lot of time to change the speed of the motor. So we have developed a system that allows for very fast speed changes.

In conclusion, Speed Control System represents a promising alternative to general Speed Control System. This allows for much faster speed changes compared to conventional speed control systems, allowing for faster work and less time consuming tasks

1.2 Objectives

The Objective of the Project is given bellow:

- To Study the design and construction of speed control system.
- To Analyze the performance test of the project.
- To Acquire real time motor speed feedback using a rotary encoder.

Chapter 2
Literature Review

2.1 Background

The history of speed control system of motor is a long and continuous story of technological development, with many stages from the need for electric motor control to the emergence of modern power electronics. Below is a detailed history of speed control system of motor:

Early Stages: The Concept of Motor Speed Control (1900 - 1940)

In the early 1900s, although AC motors were introduced, controlling their speed was difficult. At that time, speed control was mainly done using DC motors, because DC motors are relatively easy to control. Mechanical gears, pulley systems, etc. were used for AC motors at that time.

The Beginning of Power Electronics (1940s - 1960s)

The transistor was invented in 1947 (Bell Labs), which later laid the foundation for VFD technology. Then the thyristor (SCR - Silicon Controlled Rectifier) was invented in the 1950s. The SCR was the first high-power electronic switch, which made it possible to control high voltage and current.

The Birth of the Modern VFD (1960s - 1980s)

In the 1960s, the first "Variable Voltage Variable Frequency" (VVVF) drives were developed using SCR and inverter technology. At that time, several companies (such as General Electric, Siemens, Toshiba) began experimenting with speed control by varying the frequency of AC motors. These VFD drives were large, expensive, and complex.

Digital Control and the Invention of IGBT (1980 - 2000)

In 1980, the IGBT (Insulated Gate Bipolar Transistor) was invented, which could act as a much faster and more efficient switch than the SCR. IGBTs made VFDs smaller, more reliable, and more energy-efficient. Additionally, the addition of a microcontroller and digital processing unit makes software-controlled speed control possible.

Modern Era (2000 - Present)

Modern VFDs are now very compact, user-friendly and packed with smart features.

These include:PLC integration, Automated feedback loops, Internet/IoT connectivity, Energy saving features.

VFDs are currently being used in almost all sectors—HVAC, pumps, conveyors, elevators, fans, industrial machines.

The Role of the Siemens PLC in Speed Control

The Siemens Programmable Logic Controller (PLC) serves as the central intelligence of the speed control system. Its primary functions include:

Receiving Feedback: The PLC acquires the actual motor speed from the rotary encoder. This feedback signal, typically in the form of pulses, is processed by high-speed counter modules or integrated counter functionalities within the PLC.

Implementing the Control Algorithm: The core of the speed control logic resides within the PLC program. This typically involves a PID (Proportional-Integral-Derivative) control algorithm, although other advanced control strategies can also be implemented. The PLC calculates the necessary control output based on the desired speed (setpoint) received from the HMI and the actual speed feedback.

Generating the Control Signal: The calculated control output from the PLC is then sent to the Variable Frequency Drive (VFD). This signal is usually an analog voltage or current signal (e.g., 0-10V, 4-20mA) or a digital communication signal (e.g., Profibus, Profinet, Modbus).

Handling Safety and Interlocks: The PLC also plays a crucial role in implementing safety features and interlocks, such as over-speed protection, fault detection, and emergency stop functionalities.

Communication with HMI and other Systems: The PLC communicates with the Human-Machine Interface (HMI) to receive the desired speed setpoint and display real-time operating parameters (actual speed, error, etc.). It might also communicate with other automation systems or supervisory control and data acquisition (SCADA) systems.

Literature Relevant to PLC-Based Speed Control

Research and application in this area often focus on several key aspects related to the PLC implementation:

PID Control Algorithm Implementation on PLC: A significant body of literature discusses the implementation of PID control algorithms on PLCs. This includes considerations for discrete-time implementation, tuning methods suitable for PLC environments, and techniques to handle issues like integral windup and bumpless transfer during setpoint changes or mode switching.

Discrete-Time PID: Articles often detail the conversion of continuous-time PID equations into their discrete-time equivalents using methods like positional and velocity algorithms. The choice of discretization method can impact the stability and performance of the control loop.

Tuning Methodologies for PLCs: Literature explores various PID tuning techniques adapted for PLC-based systems, including manual tuning, Ziegler-Nichols methods implemented within the PLC environment, and more advanced auto-tuning algorithms that can be integrated into the PLC program.

Anti-Windup and Bumpless Transfer: Research addresses the practical challenges of integral windup due to actuator saturation (VFD output limits) and the need for smooth transitions when control parameters or modes are changed. PLC programming techniques for implementing anti-windup and bumpless transfer are often discussed.

Interfacing Rotary Encoders with Siemens PLCs: A crucial aspect is the accurate and reliable acquisition of speed feedback from the rotary encoder. Literature in this area covers:

High-Speed Counter Modules: Siemens PLCs offer specialized high-speed counter modules designed to capture the rapid pulses from incremental encoders. Application notes and technical documentation detail the configuration and programming of these modules.

Direct Encoder Interface: Some Siemens PLC CPUs have integrated high-speed counter inputs, and literature discusses their capabilities and programming considerations.

Pulse to Speed Conversion: Research explores different methods implemented in the PLC to convert the encoder pulse frequency or pulse count over a specific time window into a meaningful speed value (e.g., RPM). Accuracy and resolution of the speed measurement are key considerations.

Communication between Siemens PLC and VFD: The method of communication between the PLC and the VFD is critical for effective speed control. Literature covers various communication protocols:

Analog Control (0-10V, 4-20mA): While simpler, analog control is susceptible to noise. Literature discusses proper wiring and scaling of analog signals for VFD speed control.

Digital Communication Protocols (Profibus, Profinet, Modbus RTU/TCP): These industrial communication protocols offer more robust and feature-rich communication, allowing for parameterization, diagnostics, and advanced control commands to be exchanged between the PLC and VFD. Literature details the configuration and programming aspects of these protocols within the Siemens PLC environment.

HMI Integration for Speed Control: The HMI provides the user interface for setting the desired speed and monitoring the system's performance. Literature focuses on:

Setpoint Input: Designing user-friendly interfaces on the Siemens HMI for entering the desired motor speed.

Real-time Data Display: Displaying crucial parameters like actual speed, error, and VFD status on the HMI.

Alarm and Fault Handling: Implementing visual and interactive mechanisms on the HMI to display alarms and fault conditions reported by the PLC or VFD.

Safety and Interlock Implementation in PLC for Motor Control: Safety is paramount in motor control applications. Literature discusses:

Emergency Stop Circuits: Implementing hardwired and PLC-based emergency stop functionalities.

Over-Speed Protection: Using the PLC to monitor the actual speed and trigger safety mechanisms if the motor exceeds a predefined limit.

Fault Detection and Handling: Programming the PLC to detect faults reported by the VFD or encoder and implement appropriate safety responses.

Advanced Control Strategies on Siemens PLCs: While PID is common, literature also explores the implementation of more advanced control techniques on Siemens PLCs for improved speed control performance, such as:

Feedforward Control: Using knowledge of the load or system dynamics to proactively adjust the VFD output.

Cascade Control: Implementing nested control loops (e.g., an inner current loop within the VFD and an outer speed loop in the PLC).

Model Predictive Control (MPC): Some advanced Siemens PLCs have the computational power to implement MPC algorithms for optimized control.

Key Considerations in PLC-Based Speed Control System Development

Literature emphasizes several crucial considerations for successful development:

System Requirements: Clearly defining the required speed accuracy, dynamic response, operating range, and environmental conditions is essential for selecting appropriate hardware and designing the control algorithm.

Hardware Selection: Choosing the right Siemens PLC CPU, communication modules, HMI panel, VFD, motor, and encoder based on the application's needs and budget.

Software Development: Efficient and well-structured PLC programming using Siemens TIA Portal or STEP 7 software is critical. This includes implementing the control algorithm, communication protocols, and safety logic.

Commissioning and Tuning: Proper commissioning of the hardware and meticulous tuning of the PID controller (or other control algorithm) are necessary to achieve the desired performance.

Troubleshooting and Maintenance: Understanding the system architecture and having effective diagnostic tools within the PLC and HMI are important for troubleshooting and long-term maintenance.

Chapter 3
Hardware Implementation

3.1 Induction Motor: An induction motor (also called an asynchronous motor) is an electric motor that operates based on electromagnetic induction. It's one of the most commonly used types of motors, especially in industrial and household appliances.

When alternating current (AC) flows through the stator (the stationary part), it creates a rotating magnetic field. This magnetic field induces a current in the rotor (the rotating part), and the interaction between the magnetic field and the induced current generates torque, causing the rotor to spin.

It is the most commonly used motor in industrial and domestic applications due to its simplicity, robustness, and lo

w cost**.**

Main Parts of an Induction Motor:

i. Stator (Stationary Part)
ii. Rotor (Rotating Part)

Types of Induction Motors:

i. Single-Phase Induction Motor
ii. Three-Phase Induction Motor

Working Principle:

i. When AC supply is given to the stator, it produces a rotating magnetic field**.**
ii. This field cuts through the rotor conductors and induces a current in them (as per Faraday's Law of Electromagnetic Induction**).**
iii. The induced current in the rotor produces its own magnetic field.
iv. The interaction between the stator's rotating field and rotor's magnetic field causes the rotor to turn.

v. The rotor always rotates at a speed slightly less than the synchronous speed**,** which is called slip**.**



Figure: 3.1 Induction Motor

Advantages of Induction Motor:

i. Simple and rugged construction
ii. Low maintenance (no brushes or commutators)
iii. Cost-effective
iv. Reliable and durable

3.2 VFD (Variable Frequency Drive): A VFD (Variable Frequency Drive) is an electronic device that controls the speed and torque of an electric motor by varying the frequency and voltage of the power supplied to it.

Electric motors (especially induction motors) typically run at a constant speed when connected directly to AC power. A VFD allows you to adjust motor speed to match the application's needs, improving efficiency and performance.

It is commonly used with induction motors in industrial applications to improve process control, reduce energy consumption, and extend equipment life.

Main Components of a VFD:

1. Rectifier (AC to DC Converter)
    - Converts incoming AC power to DC power
    - Usually consists of diodes or thyristors
2. DC Link (Intermediate Circuit)
    - Smooths and filters the DC voltage
    - Contains capacitors and inductors
3. Inverter (DC to Variable AC Converter)
    - Converts the filtered DC into AC with variable frequency and voltage
    - Uses IGBTs or other power semiconductors

Working Principle of VFD:

1. AC Input is supplied to the VFD.
2. Rectifier section converts it to DC.
3. DC Link filters the DC to remove ripples.
4. Inverter converts this DC into AC with controlled frequency and voltage**.**
5. Output is given to the motor, allowing speed and torque control.



Figure: 3.2 VFD (Siemens V20)

Benefits of Using a VFD:

i. Energy savings**:** Especially in variable torque applications like fans and pumps
ii. Speed control**:** Precise and smooth motor speed regulation
iii. Reduced mechanical stress**:** Smooth acceleration and deceleration
iv. Extended equipment life**:** Less wear and tear
v. Improved process control**:** Especially in automation and manufacturing
vi. Soft starting: No sudden jerk during motor startup

Common Applications of VFD:

i. HVAC systems (fans, blowers, compressors)
ii. Water pumps and sewage systems
iii. Conveyor belts and material handling
iv. Industrial machinery
v. Elevators and cranes

3.3 Rotary Encoder: A rotary encoder is an electro-mechanical device that converts the angular position or motion of a shaft or axle into an electrical signal. It's commonly used in automation, robotics, CNC machines, and motor control systems.

As the shaft turns, the encoder generates electrical signals (usually square waves). These signals are read by a controller (like a PLC or microcontroller), which calculates position or speed.

Types of Rotary Encoders

Rotary encoders are mainly divided into two types:

1. **Incremental Encoder**
    - It provides information about the change in position, not the absolute position.
    - It generates pulses as the shaft rotates, which can be counted to determine movement.
    - Common outputs: A, B (and sometimes Z for index pulse).
    - Used in applications where only relative movement is needed (e.g., CNC machines, printers).
2. **Absolute Encoder**
    - It provides a unique digital code for each shaft position.
    - Even after a power loss, the position can be recovered.
    - More complex and expensive than incremental encoders.
    - Used in applications where exact position data is critical (e.g., robotics, industrial automation).

Working Principle

- A rotary encoder usually consists of a rotating disk with patterns (optical or magnetic), a light or magnetic sensor, and an output circuit.
- As the shaft rotates, the sensor detects the patterns and generates corresponding signals.
- These signals are processed to determine:
    - Direction of rotation (clockwise or counterclockwise)
    - Speed of rotation
    - Position

Applications of Rotary Encoders

- Industrial automation and control systems
- Robotic arms and servo motors
- Elevators and escalators
- CNC machines and 3D printers
- Volume control in audio equipment
- Position feedback in electric motors



Figure: 3.3 Rotary Encoder

Advantages

- High precision and resolution
- Compact size
- Reliable position and speed feedback
- Long operational life due to non-contact sensing in optical types

3.4 PLC (Programmable Logic Controller): A PLC (Programmable Logic Controller) is a rugged, digital computer used for automating industrial processes, such as controlling machinery on factory assembly lines, robotic devices, or other automation systems. It is used to monitor inputs**,** make decisions based on programmed logic, and control outputs to automate a process.

Programmable: You can write and change logic using software.

Rugged: Built to withstand harsh industrial environments (dust, heat, vibration).

Reliable: Designed for 24/7 operation with minimal failure.

How PLC Works:

- Reading Inputs
- Executing Program Logic
- Updating Outputs
- Performing Diagnostics and Communication



Figure: 3.4 PLC Siemens

Advantages of PLC:

- Easy to program and modify
- Highly reliable and durable
- Real-time operation
- Compact and modular
- Easy troubleshooting and diagnostics
- Suitable for complex automation

Example: If a button is pressed (input), the PLC checks the logic, and if the condition is true, it turns on a motor (output). The whole decision-making happens in milliseconds.

3.5 HMI (Human Machine Interface): An HMI (Human-Machine Interface) is a user interface or dashboard that connects a person (the operator) to a machine, system, or device. It allows real-time interaction**,** monitoring**,** and control of machinery and industrial processes.

It is commonly used in automation systems and works closely with PLCs, sensors, and other control devices.

Purpose of HMI:

- To display data from machines or processes.
- To allow users to input commands or control settings.
- To provide status updates**,** warnings, or alarms**.**
- To visualize operations through graphics, charts, or animations.

Key Features:

i. User-Friendly Interface
ii. Data Visualization
iii. Real-Time Monitoring
iv. Alarm Management
v. Historical Data Logging
vi. Control Access

How HMI Works:

- The HMI collects data from PLCs or sensors**.**
- It processes and displays this data to the operator in a meaningful way.
- The operator can give commands (e.g., start/stop a motor, change a setpoint).
- The HMI sends these commands back to the control system.



Figure: 3.5 HMI Display

Advantages:

- Improves productivity by simplifying interaction.
- Enhances safety with alarms and monitoring.
- Enables quick decision-making with real-time data.
- Reduces downtime through effective troubleshooting tools.

3.6 Banana Cable: Banana plugs are single-wire electrical connectors used for joining wires to equipment with a cylindrical pin/plug. A Banana Cable refers to a wire or lead that is terminated with banana plugs at one or both ends. These cables are commonly used for creating temporary and safe electrical connections in testing, measuring, and prototyping environments, especially in electronics labs and educational settings**.**

Specifications to Consider:

- Wire Gauge (e.g., 18 AWG, 22 AWG)
- Length (e.g., 30 cm, 1 meter)
- Voltage/Current Rating (e.g., 30V–1000V, 10A–15A)
- Insulation Type (standard PVC vs. high-flex silicone)

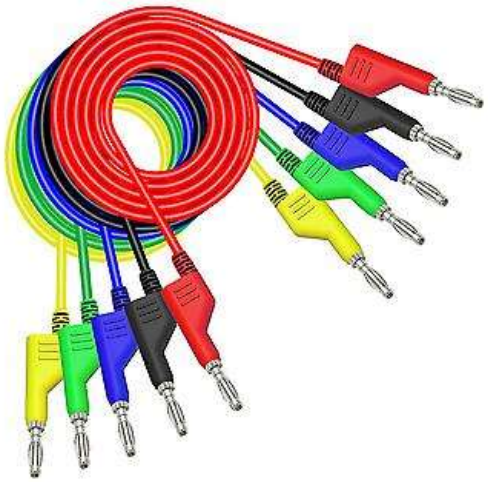


Figure: 3.6 Banana Cable

Advantages:

- Easy to plug and unplug
- Reusable and durable
- Provides reliable and stable connections
- Safe for handling in lab environments
- Widely compatible with test equipment

3.7 Pushbutton switch: It's a type of electrical switch that's used to control an electrical circuit by physically pressing a button. A Pushbutton switch, also known as a push switch, is a type of mechanical switch used to control the flow of electric current in an electronic circuit.

Working Principle:

- When the button is pressed, it causes an internal metal contact to complete (or break) a circuit.
- When the button is released, a spring mechanism returns it to its original position (in momentary type).
- Depending on the contact type (NO/NC), it either allows current to flow or stops it.



Figure: 3.7 Pushbutton switch

Advantages:

- Simple to use and install
- Reliable for manual control
- Available in various colors and sizes
- Some types offer tactile or audible feedback
- Can be easily integrated into circuit

3.8 Programing Device (Computer or Laptop): In industrial automation, a programming device is the hardware used to create, edit, upload, and monitor control programs in devices such as PLCs (Programmable Logic Controllers)**,** HMIs, and other automation components.

A laptop is one of the most commonly used programming devices due to its portability, flexibility, and compatibility with various programming software.

Connection Methods:

- USB: Most modern PLCs and HMIs support USB communication.
- Ethernet (RJ45)**:** Common in networked PLCs and SCADA systems.
- Serial (RS-232/RS-485)**:** Used for legacy systems (with USB-to-Serial converter if needed).
- Wi-Fi/Bluetooth (in advanced systems)



Figure: 3.8 Laptop

3.9 Power Supply: A switched-mode power supply (switching-mode power supply, switch mode power supply, switched power supply, SMPS, or switcher) is an electronic power supply that incorporates a switching regulator to convert electrical power efficiently. Like other power supplies, an SMPS transfers power from a DC or AC source (often mains power) to DC loads, such as a personal computer, while converting voltage and current characteristics. This higher power conversion efficiency is an important advantage of a switched-mode power supply. Switched mode power supplies may also be substantially smaller and lighter than a linear supply due to the smaller transformer size and weight. Switching 10 regulators are used as replacements for linear regulators when higher efficiency, smaller size or lighter Weight is required. Switched-mode power supplies are classified according to the type of input and output voltages. The four major categories are: • AC to DC • DC to DC • DC to AC • AC

to AC A basic isolated AC to DC switched-mode power supply consists of: • Input rectifier and filter • Inverter consisting of switching devices such as MOSFETs • Transformer • Output rectifier and filter • Feedback and control circuit The input DC supply from a rectifier or battery is fed to the inverter where it is turned on and off at high frequencies of between 20 KHz and 200 KHz by the switching MOSFET or power transistors. The high-frequency voltage pulses from the inverter are fed to the transformer primary winding, and the secondary AC output is rectified and smoothed to produce the required DC voltages. A feedback circuit monitors the output voltage and instructs the control circuit to adjust the duty cycle to maintain the output at the desired level. Basic working concept of an SMPS A switching regulator does the regulation in the SMPS.A series switching element turns the current supply to a smoothing capacitor on and off. The voltage on the capacitor controls the time the series element is turned. The Continuous switching of the capacitor maintains the voltage at the required level.



Figure: 3.9 Power Supply

Chapter 4
Methodology

4.1 Working Process: A motor speed control system is designed to regulate the speed of an electric motor precisely and respond quickly to varying load or command conditions. This setup involves the integration of a PLC (Programmable Logic Controller)**,** a VFD (Variable Frequency Drive), an Encoder, and a PID (Proportional-Integral-Derivative**)** controller in a closed-loop system**.**

Step-by-Step Working Procedure

1. Component Collection

First, gather all necessary components. These may come in different brands and models, so it is important to:

- Select compatible devices in advance.
- Ensure all components can communicate and integrate properly.
- Required components:
    - PLC (with PID function block)
    - VFD
    - Induction motor
    - Encoder
    - Start & Stop push buttons
    - Power supply and cables

2. Wiring and Connections

◉ Push Buttons:

- Start Button is connected to PLC Input I0.4
- Stop Button is connected to PLC Input I0.5

🔁 Encoder Connection:

- The encoder's A and B pulse signals are connected to PLC digital inputs I0.0 and I0.1
- These provide actual speed feedback to the PLC.

⚙ VFD and Motor:

- PLC Output Q0.0 is connected to VFD DI1 (Digital Input 1) to send the motor RUN command.

📥 PID Loop Connection:

- Encoder speed data is sent to PID input address IW64
- The setpoint (desired speed) is stored in MD20
- PID calculates the control output and sends it through QW112, which is connected to VFD AI1 (Analog Input 1) for speed control.

3. Power Supply and Testing

After completing the wiring:

- Supply power to PLC and VFD.
- Use PLC software to check all input and output connections.
- Verify the status of start/stop switches, encoder feedback, and VFD communication.

4. PLC Programming and PID Control Logic

- The PLC is programmed to control motor start/stop using push buttons.
- A PID function block is used for automatic speed regulation.
    - IW64: Input for actual speed (from encoder)
    - MD20**:** Setpoint (target speed)
    - QW112**:** Output from PID block to VFD (AI1)

 How PID Works:

1. Actual Speed is read from encoder (IW64).
2. Error is calculated:

   Error = Setpoint (MD20) − Actual Speed (IW64)
3. PID uses:
    - Proportional (P)**:** Adjusts output proportional to error
    - Integral (I)**:** Eliminates steady-state error
    - Derivative (D)**:** Predicts future error based on rate of change
4. The PID output (QW112) is sent to the VFD's analog input (AI1), which controls the frequency supplied to the motor—thus adjusting its speed.

5. System Operation

- Pressing the Start button (I0.4) triggers Q0.0**,** which starts the motor via the VFD.

- The encoder constantly feeds back the actual speed.
- The PID controller compares actual vs. desired speed, calculates the error, and adjusts motor speed in real time by changing the VFD frequency (via analog output QW112).
- Pressing the Stop button (I0.5) stops the motor.

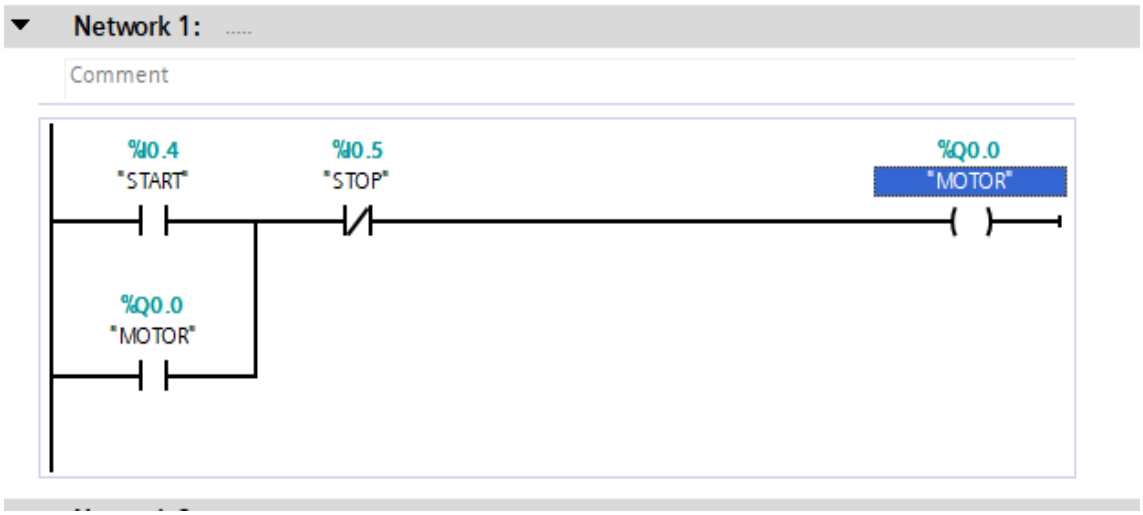


Fig: 4.1(a)- Motor control Circuit

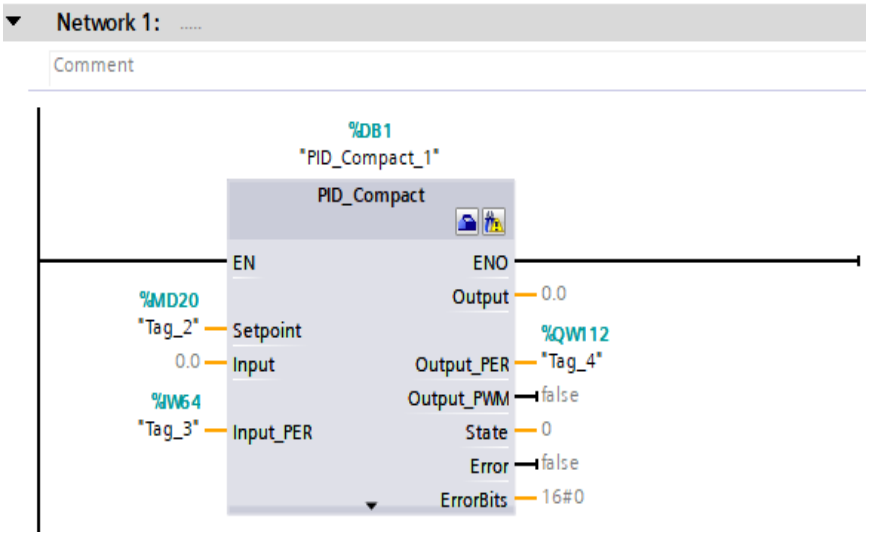


Fig: 4.1(b)- PID control Circuit

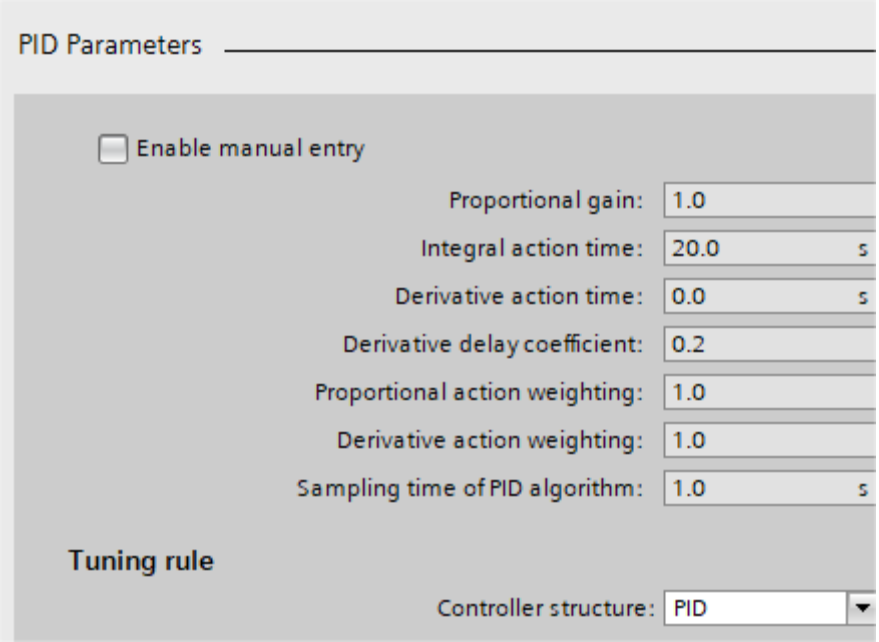


Fig: 4.1(c)- PID configure

Once the system is running, the motor speed is continuously monitored through the encoder feedback. The data collected at the current time is compared with previously recorded data to evaluate the system's performance.

- The system can calculate the difference in speed over time to determine:
    - Response time of the PID controller.
    - Accuracy of maintaining the desired setpoint.
    - How fast the system corrects deviations.

4.2 Electrical Circuit:

To wire a PLC, VFD (Variable Frequency Drive), HMI, and a rotary encoder together in an electric control system, here's a general overview of how the connections are typically made:

1. Rotary Encoder to PLC:

To PLC (for position/speed monitoring)

Connect encoder A, B, Z (incremental) or communication wires (absolute encoders with RS485, SSI, etc.) to the high-speed inputs of the PLC.

Connect encoder power supply (typically 5V or 24V DC).

2. PLC to VFD

Control Signals:

Digital Outputs of PLC (e.g., RUN, STOP, FORWARD, REVERSE) to VFD's control terminals.

Analog Output from PLC (0–10V or 4–20mA) for speed reference to VFD.

Feedback Signals:

VFD's status outputs or analog feedback (speed, current) to PLC's inputs or analog inputs.

3. Power Supply

Ensure proper isolated power to each device:

VFD gets single-phase or three-phase AC power.

PLC, HMI and encoder typically use 24V DC.

4. Communication (optional):

Use protocols like Modbus, Profibus, Ethernet/IP, or Profinet for advanced control and diagnostics.

Sample Wiring Diagram (Basic Layout):

[Rotary Encoder]

|---- A, B, Z --> [PLC High-Speed Inputs] or [VFD Encoder Inputs]

|---- +V, GND --> [24V DC Power Supply]

[PLC Outputs]

|---- Digital OUT --> [VFD DI: Run/Stop/Dir]

|---- Analog OUT --> [VFD AI: Speed Ref]

[HMI]

|-- RS-232/485 or Ethernet --→ [PLC Communication Port]

[VFD Outputs]

|---- Analog/Status OUT --> [PLC Inputs]

[Motor]

|---- Connected to VFD Output Terminals

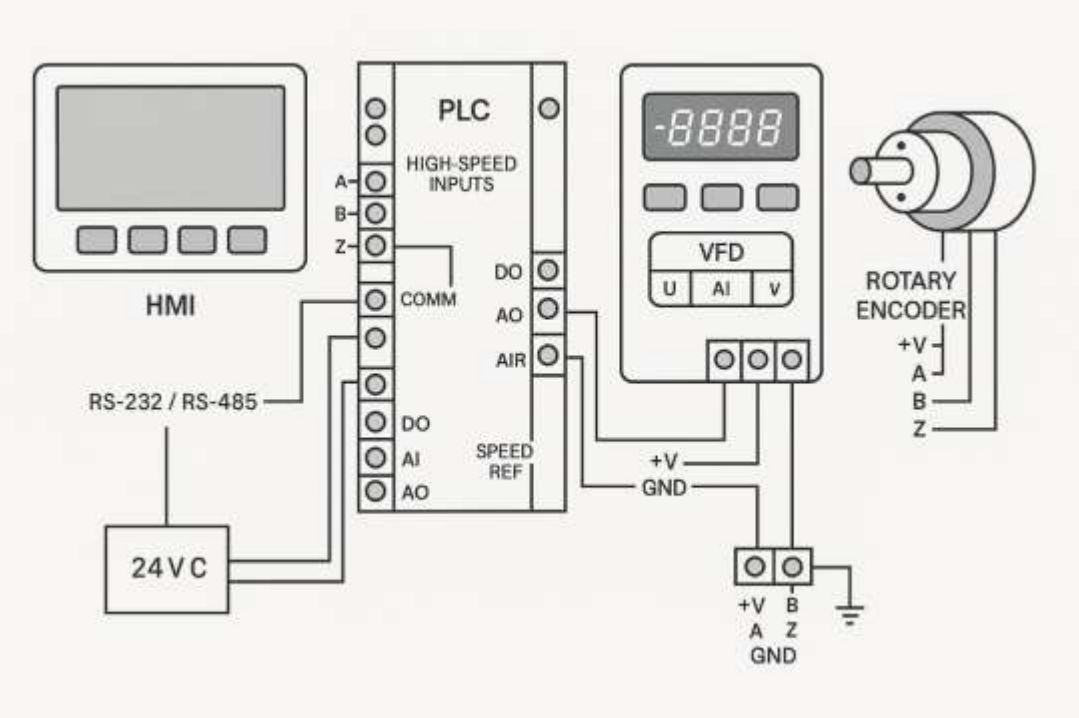


Fig:4.2(a)- Electrical Circuit

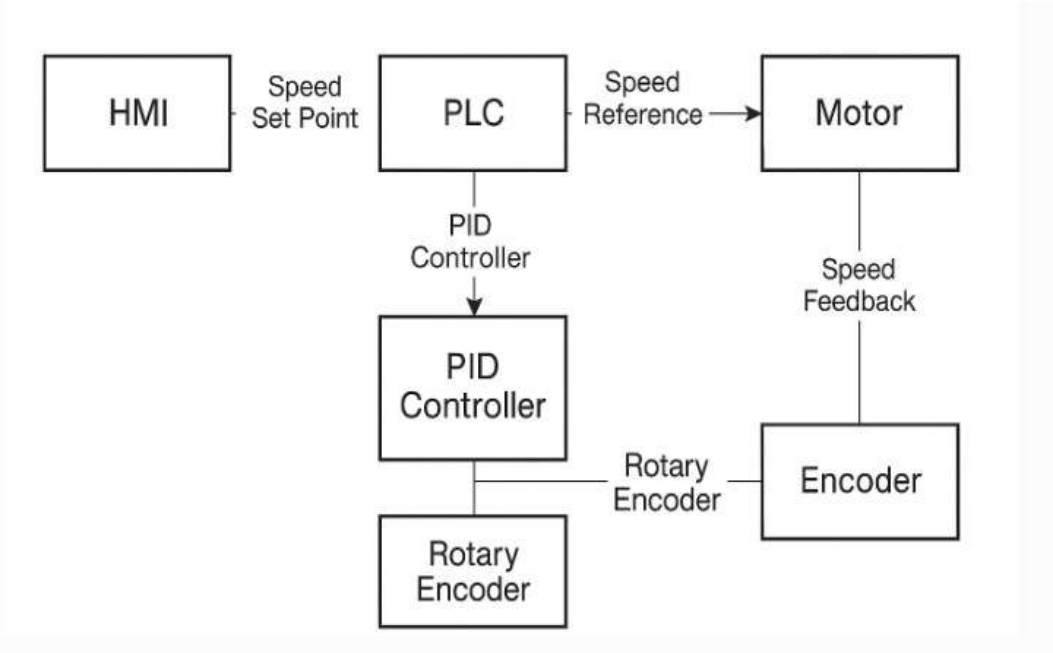


Fig: 4.2(b): PID Speed Control System Flow Chart

4.3 Schematic Views of the Project

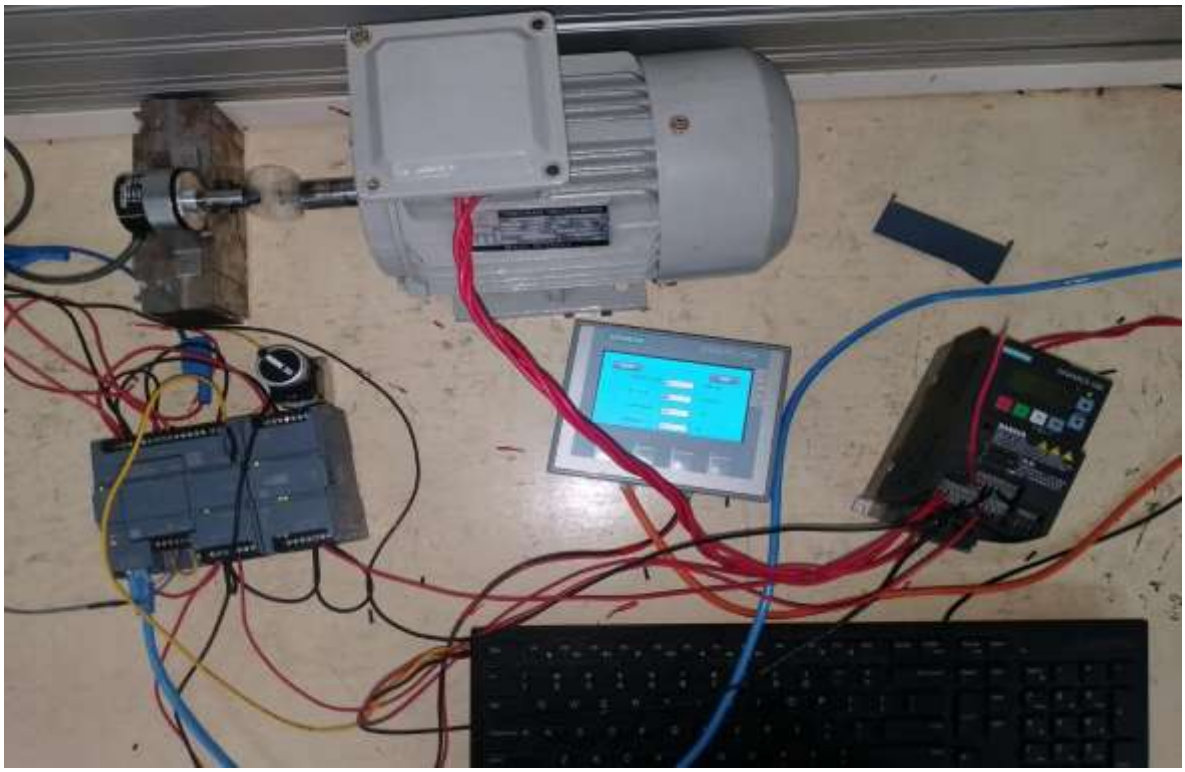


Fig: 4.3(a)-Top View

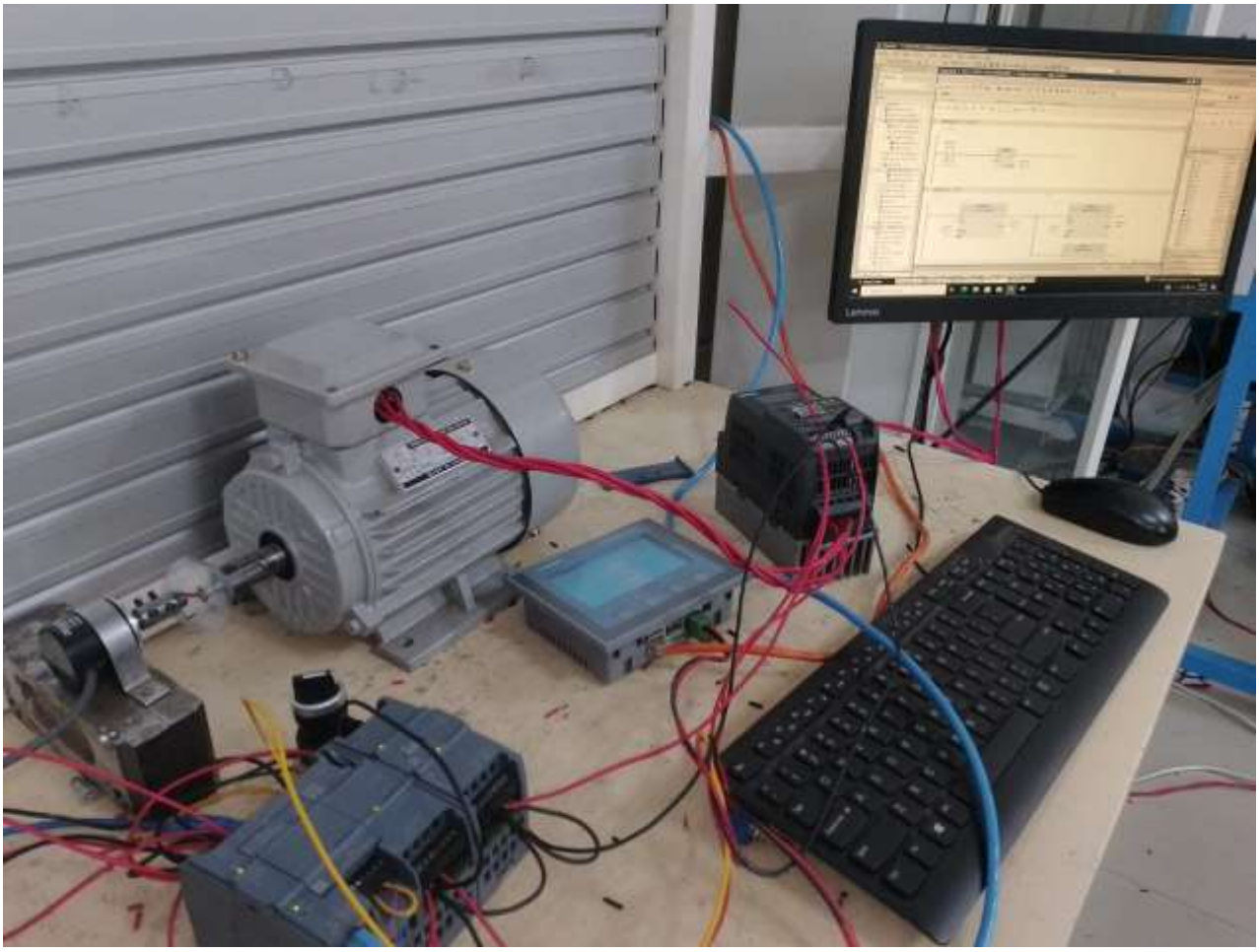


Fig: 4.3(b)- Side View

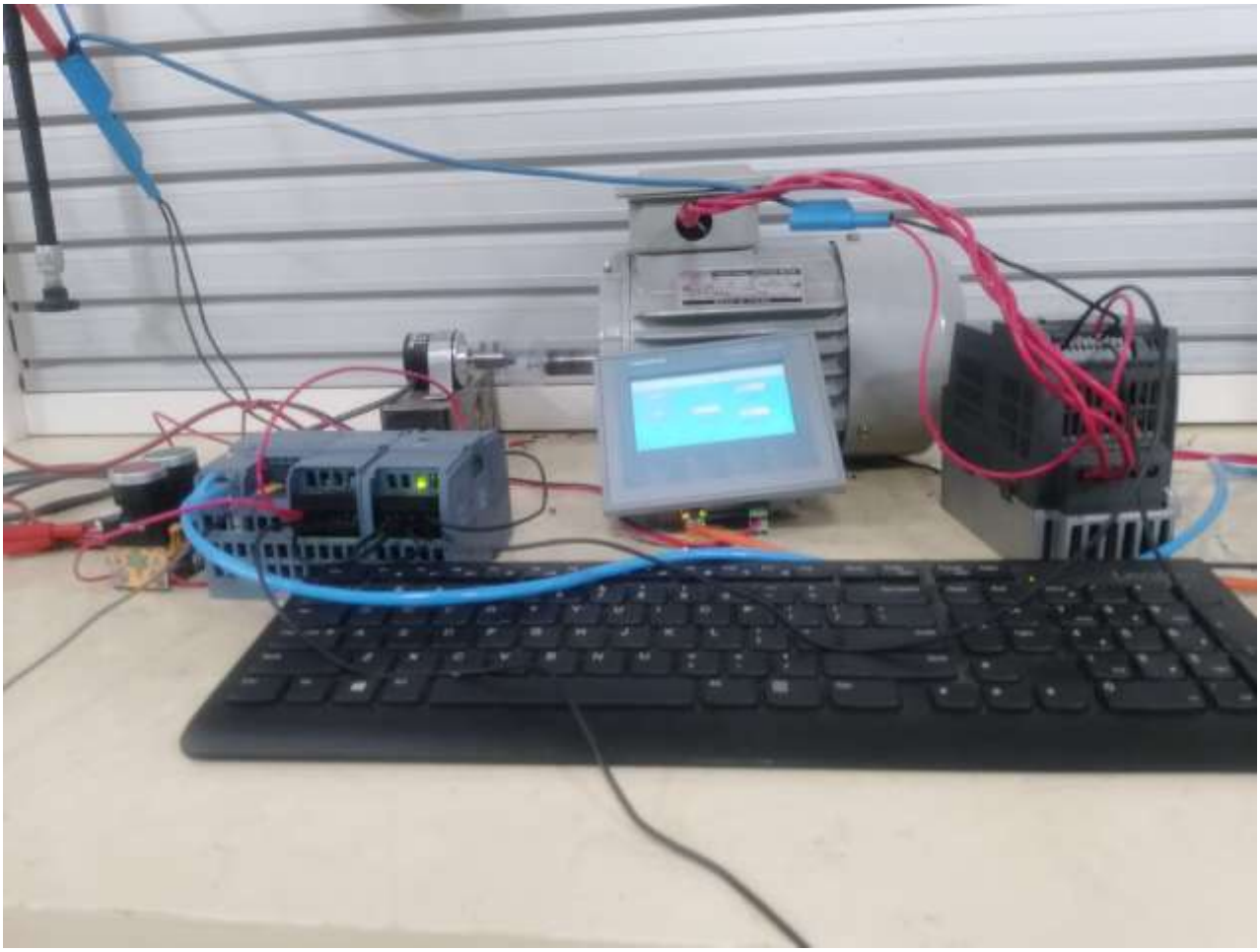


Fig: 4.3(c)- Front View

Chapter 5
Results and Discussion

Description: The chart illustrates motor response time data under a conventional open-loop control system across speed ranges from 0–200 RPM to 0–850 RPM. Each range includes three response time samples (in milliseconds) to assess system consistency and performance. At lower speeds, such as 0–200 RPM, the motor demonstrates relatively fast and stable responses (e.g., Sample-1: 2312 ms, Sample-2: 2289 ms, Sample-3: 2392 ms). However, as the target speed increases—especially between 0–300 RPM and 0–400 RPM—response times rise noticeably, with values such as 3162 ms recorded at 0–400 RPM. This trend becomes more pronounced beyond 500 RPM; at 0–700 RPM, Sample-1 reaches 4299 ms, and at 0–850 RPM, it peaks at 4406 ms. These delays reflect the growing limitations of open-loop control in managing higher-speed operations. Slight discrepancies among the samples at each speed range also highlight the system's lack of consistency and adaptability.

These findings strongly suggest that while open-loop control may suffice for low-speed scenarios, it becomes increasingly inefficient and unstable under dynamic or high-speed conditions. To address these challenges, the implementation of a PID (Proportional-Integral-Derivative) control strategy is recommended. A PID controller introduces feedback, enabling the system to self-correct in real time—minimizing overshoot, reducing response time, and improving speed accuracy. The proportional and derivative components reduce rise time and dampen oscillations, while the integral term eliminates steady-state errors, ensuring precise convergence to the target speed. Furthermore, the adaptive nature of feedback enhances consistency by allowing the system to respond effectively to load variations or external disturbances. Overall, this conventional data serves as a benchmark for evaluating the superiority of PID-based speed control across the motor's full operational range.

Table: 5.1
Conventional Speed, Frequency & Time Data Table:

SL. No	Conventional				
	Speed (rpm)	**Frequency (Hz)**	**Sample-1 Time (ms)**	**Sample-2 Time (ms)**	**Sample-3 Time (ms)**
1	0-200	7	2312	2289	2392
2	0-250	9	2371	2300	2392
3	0-300	11	2406	2315	2419
4	0-350	13	2326	2321	2530
5	0-400	14.5	3162	3390	3320
6	0-450	16	3392	3414	3236
7	0-500	18	3371	3342	3335
8	0-550	20	3407	3345	3336
9	0-600	22	3476	3316	3340
10	0-650	23.5	3370	3358	3302
11	0-700	25	4299	3532	4259
12	0-750	27	4260	4394	4323
13	0-800	29	4286	4364	4292
14	0-850	31	4406	4325	4312
15	0-900	32.5	4332	4401	4362
16	0-950	34	4490	5327	5272
17	0-1000	36	5419	5236	5358
18	0-1050	38	5425	5359	5382
19	0-1100	40	5446	5386	5366
20	0-1150	41.5	5474	5390	5350
21	0-1200	43	5416	5347	5313
22	0-1250	45	5346	6498	6317
23	0-1300	47	6462	6466	6444
24	0-1350	49	6340	6262	6579

We used a Siemens S7-1200 PLC to measure how the speed of a motor varies over time. A rotary encoder was connected to the motor shaft, which generates pulses as the shaft rotates. These pulses were used to calculate the speed.

The rotary encoder output was connected to a digital input of the PLC. The encoder generates a specific number of pulses per revolution (PPR). By counting the number of pulses within a fixed time interval, We was able to calculate the motor speed.

Using a timer, We counted how many pulses were received in that period. Then, the speed in RPM was calculated using the formula:

$$\text{Speed (RPM)} = \frac{\text{Pulse Count Second} * 60}{\text{Pulse Per Revolutions}}$$

This speed value was stored in a Data Block (DB) array every second.

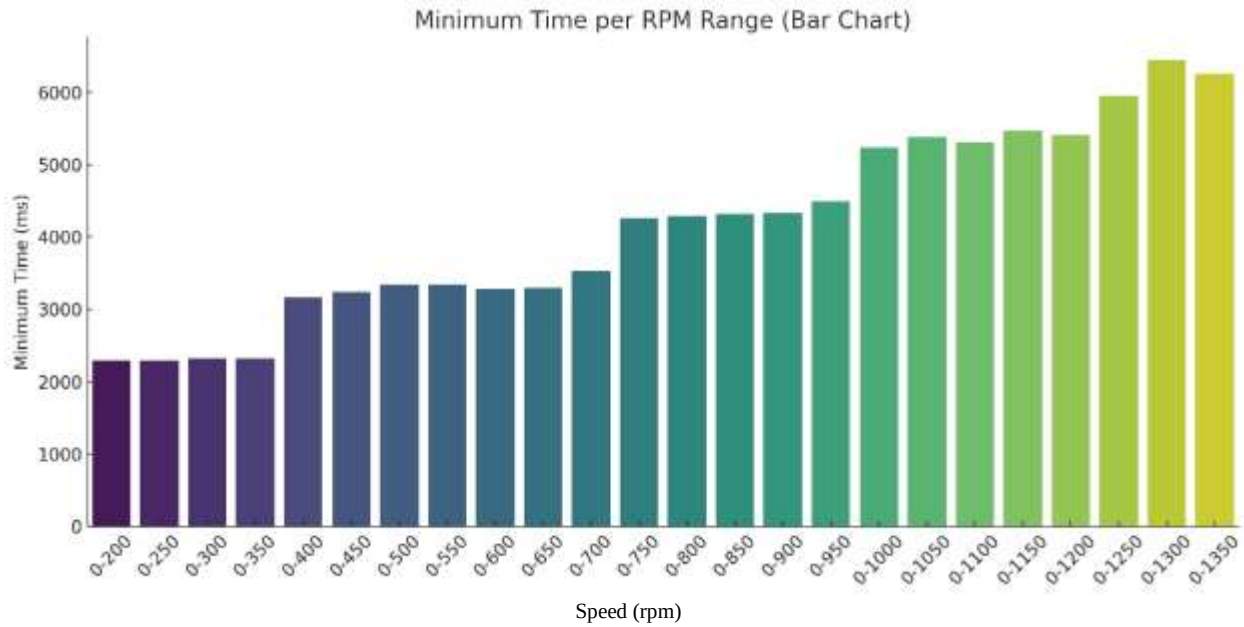


Fig:5.1(a)- Conventional Minimum Time Per RPM Range (Bar Chart)

The provided bar chart illustrates the time (in milliseconds) recorded for each RPM range, from 0–200 up to 0–1350 RPM. From 0–200 to 0–350 RPM, the minimum response time remains relatively steady around 2300 MS. Between 0–400 to 0–700 RPM, the response time starts rising, reaching around 3300–3600 MS. Between 0-1000 to 0-1200 RPM, the response time reaching around 5000-5200 MS. Between 0-1250 to 0-1350 RPM, the response time reaching around 6000-6200 MS.

Table: 5.2
Using PID Speed, Frequency & Time Data Table:

SL. No	Using PID						
	Speed (rpm)	**Frequency (Hz)**	**Sample-1 Time (ms)**	**Sample-2 Time (ms)**	**Sample-3 Time (ms)**	**Sample-4 Time (ms)**	**Sample-5 Time (ms)**
1	0-200	7	1112	1180	1102	1098	1109
2	0-250	9	1200	1168	1169	1880	1120
3	0-300	11	1190	1209	2038	2031	2140
4	0-350	13	2130	2088	2092	2036	2028
5	0-400	14.5	2033	2087	2138	2029	2121
6	0-450	16	1880	2090	2101	2156	2165
7	0-500	18	2037	2049	2103	2223	2074
8	0-550	20	2119	2180	2136	2100	2181
9	0-600	22	2140	2120	2110	2095	2100
10	0-650	23.5	2108	2089	2160	2091	2150
11	0-700	25	2129	2065	2162	2172	2065
12	0-750	27	2129	2108	2122	2090	2183
13	0-800	29	2134	2108	2161	2113	2196
14	0-850	31	2073	3000	2124	2084	2094
15	0-900	32.5	2112	2033	2085	2109	2162
16	0-950	34	2124	2159	2080	2115	2125
17	0-1000	36	2147	2137	3071	2162	3093
18	0-1050	38	3148	2137	3080	2162	3162
19	0-1100	40	3052	3019	3080	3138	3166
20	0-1150	41.5	3129	3131	3060	3120	3086
21	0-1200	43	3063	2940	3061	3111	3132
22	0-1250	45	3119	3111	3158	3030	3202
23	0-1300	47	3017	3104	3094	3073	3032
24	0-1350	49	3151	3134	3052	3127	3084

After using PID control, the time it takes for the motor speed to reach the setpoint and become stable is called settling time. To find this time, time-based speed data is collected. As soon

as the set point is given, a timer is turned on in the PLC. When the process value reaches the set point, the timer is turned off. From there, the time is collected and the timer is reset.

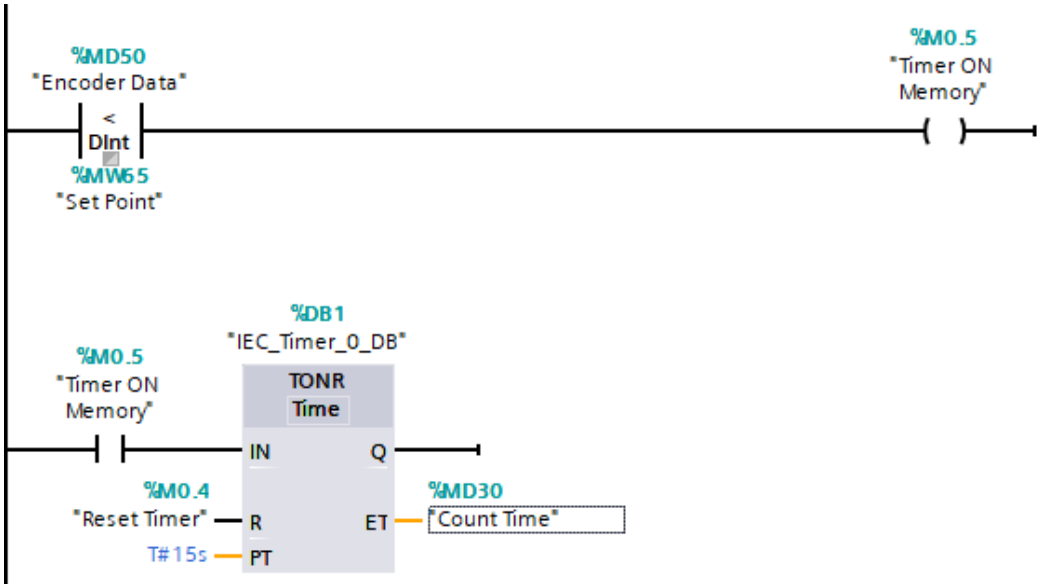


Fig:5.2(a)- Time collect program

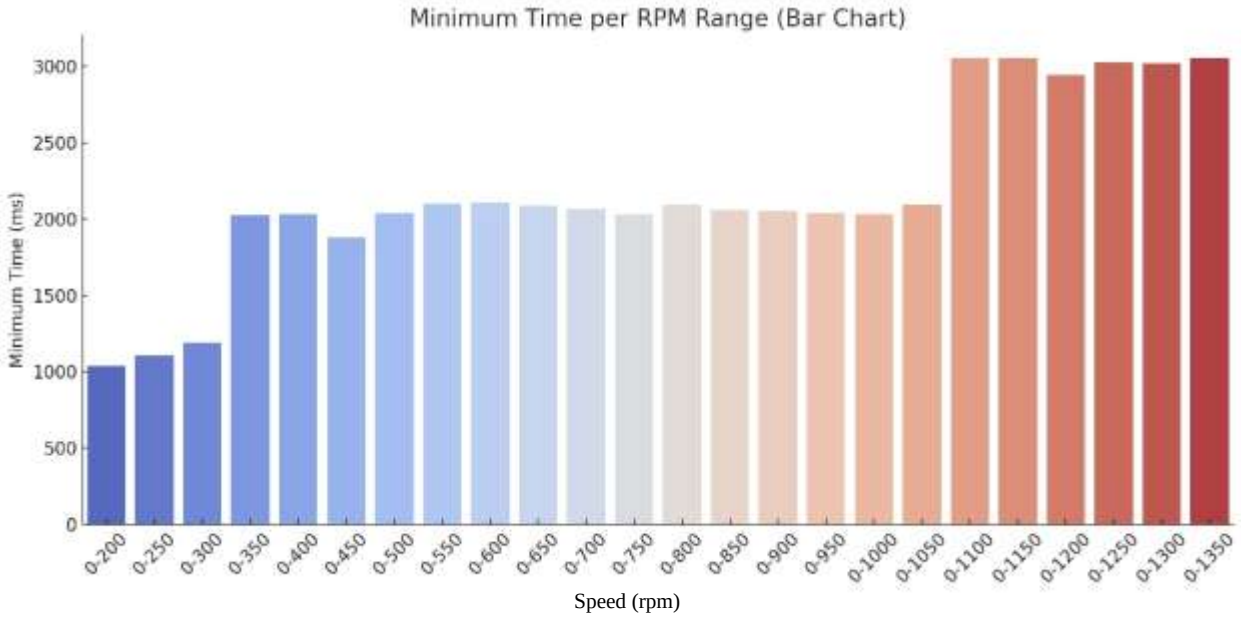


Fig:5.2(b)- PID Control Minimum Time Per RPM Range (Bar Chart)

The presented bar chart shows the minimum time (in milliseconds) recorded across multiple samples for each RPM range from 0–200 to 0–1350. In the early RPM ranges (0–200 to 0–300), the system shows excellent responsiveness with minimum times staying near 1000–1200 ms. From 0–350 to around 0–800, there's a gradual rise in minimum response time, suggesting that as speed increases. Starting from 0–1000 RPM onwards, there's a noticeable spike in minimum time, jumping up to 3000+ ms by 0–1100 to 0–1350 RPM.

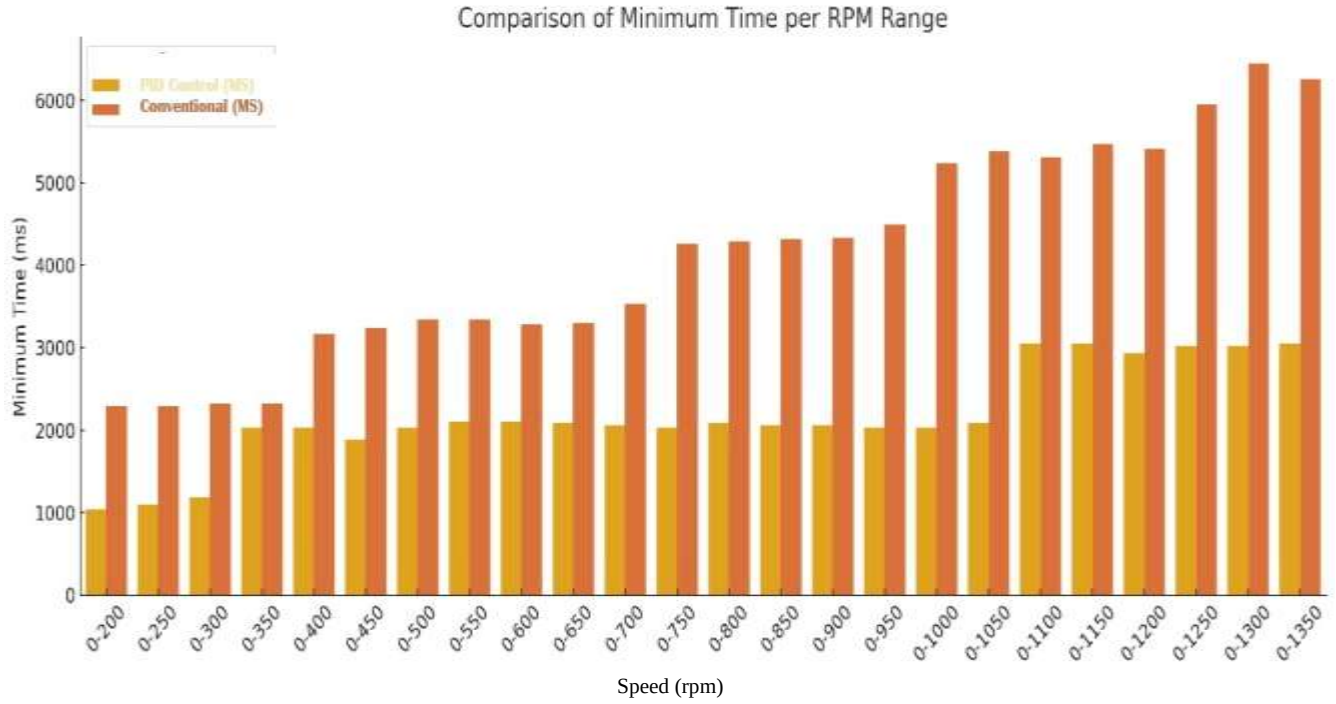


Fig:5.2(c)- Comparison of Minimum Time Per RPM Range

This report aims to compare two sets of data converted response times and PID response times across a range of RPM intervals. The objective is to assess how closely the modeled performance reflects real-world behavior. This comparative analysis reveals a critical insight: theoretical predictions alone are insufficient for understanding high-speed system performance. While they serve as a good baseline.

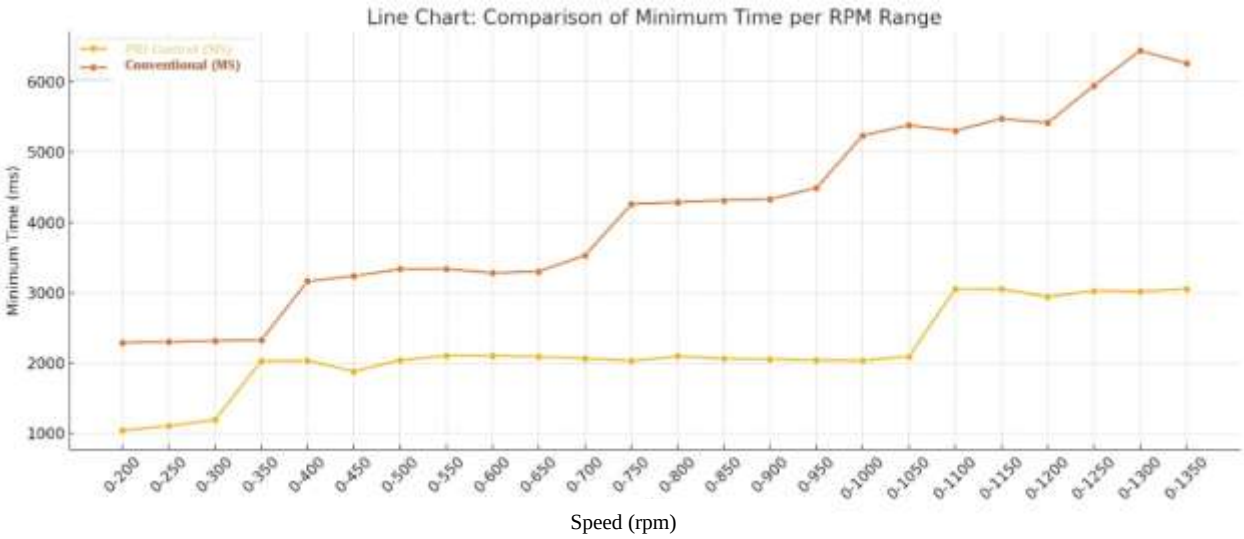


Fig:5.2(d)- Comparison of Minimum Time Per RPM Range (Line Chart)

Here the comparison of Conventional and PID Data in Line Chart.

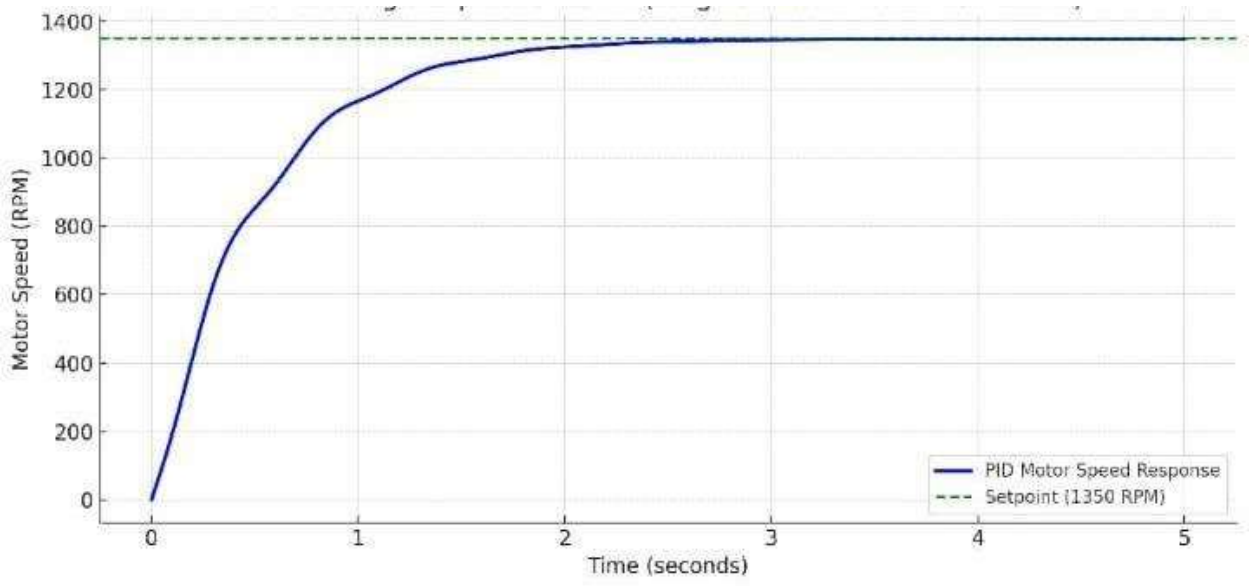


Fig:5.2(e)- PID Tuning Response Curve

When set point is 0-1200 than the PID respons time is around 3000 MS.

5.3 Results Analysis:

The start-up time of an induction motor is usually calculated in industry using a formula.

T = (HP/2) + 5 sec

Conventionaly the motor required approximately 6340 MS to accelerate from 0 to 1350 RPM. This relatively slow response time limited the overall efficiency and responsiveness of the system.

PID control is applied to the motor speed loop using VFD + Encoder. This will ensure that the motor responds quickly and smoothly to the setpoint RPM.

The Mathematical Formula for a PID:

$$u(t) = K_p \cdot e(t) + K_i \int_0^t e(\tau)\,d\tau + K_d \cdot \frac{d}{dt}e(t)$$

Where:

- u(t): control signal (output of the PID controller)
- e(t) = r(t) - y(t): error between the desired setpoint r(t) and the measured process variable y(t)
- K_p: proportional gain
- K_i: integral gain
- K_d: derivative gain

Response time depends on PID gain tuning:

1. Kp (Proportional Gain): If increased, the motor responds faster. But if it is higher, overshoot / instability occurs.

2. Ki (Integral Gain): Slowly brings the error to zero. If higher, the system slows down and overshoots.

3. Kd (Derivative Gain): Prevents rapid changes (reduces overshoot). Noise sensitive, so use carefully.

If you want to make the response faster:

Increase Kp, so that it takes action faster.

Add Kd slightly, to reduce overshoot.

Increase Ki gradually, so that the steady-state error is reduced.

Real world target response time:

On a 1.5 HP motor, it is possible to reach 0 to 1350 RPM in 2500 – 3000 MS seconds with PID tuning.

Must keep the motor inertia, VFD limit and overcurrent safety on.

However, after applying PID control through proper programming, the motor now reaches the same speed in just about 3151 MS. This marks a significant improvement, reducing the acceleration time by nearly 6340 MS per cycle.

This enhancement in response time has brought notable benefits to the system's performance. The motor now exhibits faster acceleration, greater stability, and improved accuracy in maintaining the desired speed. These improvements are especially valuable in industrial applications, where precise and timely motor control is critical for increasing productivity and minimizing process delays.

By reducing the time required for each operational cycle, the system is now capable of completing more tasks in a shorter amount of time. This leads to increased throughput and optimized performance across the entire operation. Furthermore, the application of PID control has resulted in smoother and more consistent motor behavior, ensuring reliability in automated and high-performance environments.

In summary, the integration of PID control has not only improved the motor's speed regulation but also significantly enhanced the efficiency, accuracy, and reliability of the overall system—making it highly suitable for demanding industrial applications.

5.4 Discussion:

This project focused on the design and implementation of a closed-loop motor speed control system, integrating multiple Siemens automation components. The primary objective was to enable precise control of an induction motor's speed based on user-defined inputs and real-time feedback. The core elements—Siemens PLC, HMI, VFD, rotary encoder, and induction motor—were chosen for their reliability, industrial compatibility, and ease of integration.

At the heart of the system, the Siemens PLC played a crucial role in executing control logic. It continuously monitored the speed of the motor via feedback from the rotary encoder and compared it with the desired speed set through the HMI. The inclusion of a PID (Proportional-Integral-Derivative) control algorithm within the PLC logic ensured that the system could automatically adjust the motor input to minimize error, reduce overshoot, and maintain a steady state. This use of feedback control is a classic technique in automation, enhancing both accuracy and efficiency.

The rotary encoder served as the feedback device, translating mechanical motion into electrical pulses. These pulses were counted using a high-speed counter input on the PLC, which calculated the actual RPM of the motor. This real-time measurement was vital for dynamic response and precise control. The accuracy of speed detection directly affected the control system's ability to regulate the VFD output, making the encoder a critical part of the loop.

The Variable Frequency Drive (VFD) received the control signal from the PLC—either as an analog voltage (0–10V) or via digital communication (e.g., PROFINET). It modulated the power supply frequency and voltage to the induction motor, thus adjusting its speed. The VFD's role as the actuator in the loop proved highly effective, as it responded smoothly to control commands while maintaining motor performance and energy efficiency. The selection of a VFD over simpler methods such as resistive control or mechanical means highlighted the project's emphasis on modern, energy-efficient solutions.

An essential aspect of the project was user interaction and system monitoring through the Siemens HMI. The HMI interface allowed operators to input the desired speed (RPM) and visualize system parameters such as current RPM, motor status, and alarm conditions. This interface significantly

improved the system’s usability and operator safety. The integration of the HMI with the PLC using Siemens' TIA Portal simplified configuration, allowing seamless data exchange and real-time updates.

The wiring and system integration phase demonstrated the importance of structured planning. Proper isolation, signal grounding, and layout were necessary to avoid noise interference—especially in the encoder feedback loop. Safety features such as emergency stop buttons, overcurrent protection, and software-based interlocks were included to protect equipment and personnel, reinforcing the importance of safety in industrial automation design.

This project also highlighted challenges typical of real-world industrial applications, such as managing noise in encoder signals, tuning PID parameters for optimal performance, and ensuring communication stability between PLC, VFD, and HMI. Addressing these issues required a mix of theoretical understanding and practical troubleshooting skills.

In conclusion, the project effectively demonstrated the integration of control systems, feedback loops, and automation technologies to solve a real engineering problem—controlling motor speed. The knowledge and techniques applied here are scalable to larger systems, making the design not only functional but also expandable for future needs such as torque control, load detection, or predictive maintenance through IoT connectivity.

Importance of PID Control System: A PID control system (Proportional-Integral-Derivative controller) is one of the most widely used control mechanisms in engineering and industrial systems. Here's why it's important:

1. Precise Control

PID controllers provide accurate and stable control over various processes, such as temperature, speed, pressure, or position.

2. Simple Yet Powerful

Despite being conceptually simple, PID controllers can handle complex dynamic systems effectively.

3. Robustness

They work well across a wide range of operating conditions and are relatively easy to tune for stability and responsiveness.

4. Error Minimization

Proportional (P): Reduces the present error.

Integral (I): Eliminates steady-state error over time.

Derivative (D): Predicts and dampens future errors, improving stability.

5. Wide Application

Used in manufacturing, robotics, aerospace, automotive systems, energy management, and more.

6. Cost-Efficient

Implementation is inexpensive in both software and hardware, especially for systems not requiring adaptive or nonlinear control.

Chapter 6
Advantage and Limitations

6.1 Advantage:

1. Encoder feedback to PLC ensures accurate motor RPM.
2. Optimized speed matches production needs, increasing output.
3. VFD adjusts power use to the required speed.
4. Smooth starts/stops reduce wear on motor and machinery.
5. PLC-based control protects against faults, minimizing downtime.
6. PLC allows easy adjustments for changing needs.
7. HMI provides live speed data and control for operators.

6.2 Limitations:

1. High Cost: Initial setup can be expensive.
2. Sensor Failure: System relies on the encoder; its failure causes issues.
3. Noise Problems: Electrical noise can affect encoder accuracy.
4. Motor Limits: Induction motors have inherent performance boundaries.
5. Integration Hurdles: Combining different parts can be tricky.
6. Tuning Challenges: PID control needs careful parameter adjustment.
7. System Complexity: More components mean more potential issues.

6.3 Applicationas:

1. Problem Definition: The development begins by clearly defining the objective—to control the speed of a motor accurately and maintain it under varying loads or external disturbances using a PID controller. This involves understanding the required speed range, precision, and response characteristics.

2. System Selection and Modeling: A suitable motor (e.g., DC or BLDC) is selected based on the application's requirements. The system is then modeled mathematically using transfer functions or state-space representation to capture its dynamic behavior, including electrical and mechanical parameters like resistance, inductance, and inertia.

3. Controller Selection: A PID (Proportional-Integral-Derivative) controller is chosen for its effectiveness in managing dynamic systems. It provides a balance between fast response, accuracy, and stability by minimizing the error between the desired and actual speed.

4. PID Parameter Tuning: The PID gains (Kp, Ki, Kd) are tuned using methods such as Ziegler-Nichols, Cohen-Coon, trial and error, or software-based auto-tuning tools like MATLAB PID Tuner. Proper tuning ensures minimal overshoot, steady-state error, and oscillations.

5. Simulation: Before hardware implementation, simulations are run using platforms like MATLAB/Simulink or Python to analyze system behavior under different PID settings. This helps to validate the model and controller design virtually, saving time and resources.

6. Hardware Design: After successful simulation, the hardware setup is built. This includes the motor, motor driver circuit (e.g., H-bridge), microcontroller (like Arduino or STM32), speed sensor (e.g., rotary encoder or tachometer), and appropriate power supply based on motor ratings.

7. Software Implementation: Embedded code is developed to read sensor feedback, calculate the error, apply the PID algorithm, and generate PWM signals to control the motor speed. This software runs in a continuous loop to ensure real-time control.

8. Testing and Debugging: The system is tested under various conditions. Adjustments to the PID gains may be required based on the observed response, such as rise time, overshoot, and settling time. Fault handling and boundary conditions are also tested.

9. Validation and Performance Evaluation: The system performance is validated by comparing the actual speed with the reference input. Key metrics like response time, accuracy, and robustness against disturbances are analyzed to ensure the controller meets the design goals.

10. Optimization and Finalization: Finally, the system is optimized for efficiency, reliability, and minimal power consumption. Code is refined, safety features like speed limits are added, and the system is prepared for real-world deployment or integration into larger control systems.

6.4 Conclusion

The development of a motor speed control system using Siemens automation components has proven to be a robust, efficient, and scalable solution for industrial motion control applications. Through this project, we successfully demonstrated how modern control technologies can be integrated to precisely manage the speed of an induction motor, which is a cornerstone requirement in numerous manufacturing and processing industries.

The system relied on a Siemens PLC, which acted as the central controller. It processed input signals from a rotary encoder attached to the motor shaft, providing real-time feedback on motor speed. This feedback was essential in implementing a closed-loop control system, allowing the PLC to compare actual RPM values with user-defined setpoints and adjust the control signal accordingly. The use of a high-speed counter in the PLC ensured accurate pulse counting from the encoder, which translated into precise RPM calculations.

A Human Machine Interface (HMI) was employed to enhance the usability of the system. The HMI allowed operators to input desired RPM values, monitor real-time motor performance, and view system alerts or faults. This interface created an intuitive user experience and enabled operators to interact with the control system without needing in-depth programming knowledge.

The Variable Frequency Drive (VFD) served as the actuator in the control loop. It adjusted the frequency and voltage supplied to the induction motor, which directly influenced motor speed. The VFD received analog signals or digital commands from the PLC, based on the output of the internal PID controller. By integrating the PID control algorithm within the PLC, we achieved smooth, accurate, and dynamic speed adjustments, minimizing errors between the setpoint and the actual motor speed.

Using Siemens automation tools, particularly TIA Portal, made system development and integration more seamless. The environment provided a unified platform for configuring the PLC, HMI, and VFD, enabling efficient diagnostics, commissioning, and maintenance.

From a hardware perspective, the wiring and communication architecture were straightforward yet effective. The encoder was connected to the PLC's high-speed input terminals, the PLC was networked with the HMI and VFD via PROFINET, and safety considerations such as emergency stop circuits and overload protection were implemented to comply with industrial standards.

This project not only reinforced key engineering principles such as feedback control, signal processing, and system integration, but it also highlighted the importance of using industry-standard equipment for reliability and maintainability. The modular nature of Siemens systems also ensures that future scalability—such as adding temperature monitoring or torque control can be achieved with minimal redesign.

In summary, the motor speed control system achieved its goals of accurate, real-time RPM of the control using reliable automation components. It reflects a practical and efficient application of - mechatronic and automation engineering principles and can serve as a foundational model for larger more complex industrial control systems.

References

1. Åström, K. J., & Hägglund, T.PID Controllers: Theory, Design, and Tuning (2nd Edition) ISA –The Instrumentation, Systems, and Automation Society, 1995. Considered the classic PIDreference book, with theory, design methods, and real-world tuning strategies.

2. Ogata, Katsuhiko Modern Control Engineering (5th Edition) Prentice Hall, 2010.A standard control systems textbook that includes clear explanations of PID controllers frequency-domain design.

3. Dorf, Richard C., & Bishop, Robert H. Modern Control Systems (13th Edition) Pearson, 2016 Covers traditional and modern control methods, with detailed sections on PID tuning and implementation.

4. Nise, Norman S. Control Systems Engineering (7th Edition) Wiley, 2015 Accessible textbook for undergraduate students; includes PID examples, block diagrams, and real-world applications.

5. Franklin, Gene F., Powell, J. David, & Emami-Naeini, Abbas Feedback Control of Dynamic Systems (7th Edition) Pearson, 2014. Strong focus on dynamic modeling, feedback loops, and PID design techniques.

6. Kuo, Benjamin C., & Golnaraghi, Farid Automatic Control Systems (9th Edition) Wiley, 2009.Covers both time-domain and frequency-domain analysis of PID systems with examples.

7. Johnson, Curtis D. Process Control Instrumentation Technology (8th Edition) Pearson,2006Practical focus on industrial process control, including tuning and implementation of PID loops.

8. Duncan, William G. Introduction to PID Controllers: Theory, Tuning and Application to Frontier Areas Nova Science Publishers, 2018.Newer reference with a focus on applications in aerospace, robotics, and process automation.

9. Gopal, M. Control Systems: Principles and Design (4th Edition) McGraw Hill Education 2012.Widely used in Indian universities; offers comprehensive explanations of PID within the context of classical control.

10. Bennett, Stuart A History of Control Engineering, 1930–1955 Peter Peregrinus Ltd, 1993.While historical in nature,this book provides insight into the development and earlyimplementation of PID controllers.

11. Phillips, Charles L., & Parr, Royce D. Feedback Control Systems (5th Edition). Pearson 2010. Clear treatment of control systems with real-world engineering applications.

12. Levine, William S. The Control Handbook (3rd Edition). CRC Press, 2019. Comprehensive reference for control theory, including modern advancements.

13. Skogestad, Sigurd, & Postlethwaite, Ian. Multivariable Feedback Control: Analysis and Design (2nd Edition). Wiley, 2005. Advanced text focusing on multivariable systems and robustness.

14. Bryson, Arthur E., & Ho, Yu-Chi. Applied Optimal Control. Taylor & Francis, 1975. Classic reference for optimal control theory.

15. Kailath, Thomas. Linear Systems. Prentice Hall, 1980. Theoretical and rigorous foundation in linear system theory.

16. Brogan, William L. Modern Control Theory (3rd Edition). Prentice Hall, 1991. Focus on state-space methods and optimal control.