428452

IOT ENABLED QUENCHING TANK TEMPERATURE MONITORING

A PROJECT REPORT

Submitted by

PAWAR YASH ANILBHAI

210230111501

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING
in
Electronics & Communication Engineering
Dr. S. & S. S. Ghandhy Government Engineering College,
Surat.

Gujarat Technological University, Ahmedabad

[Jan-Apr, 2024]

428452

Dr. S. & S. S. Ghandhy Government Engineering College, Surat.

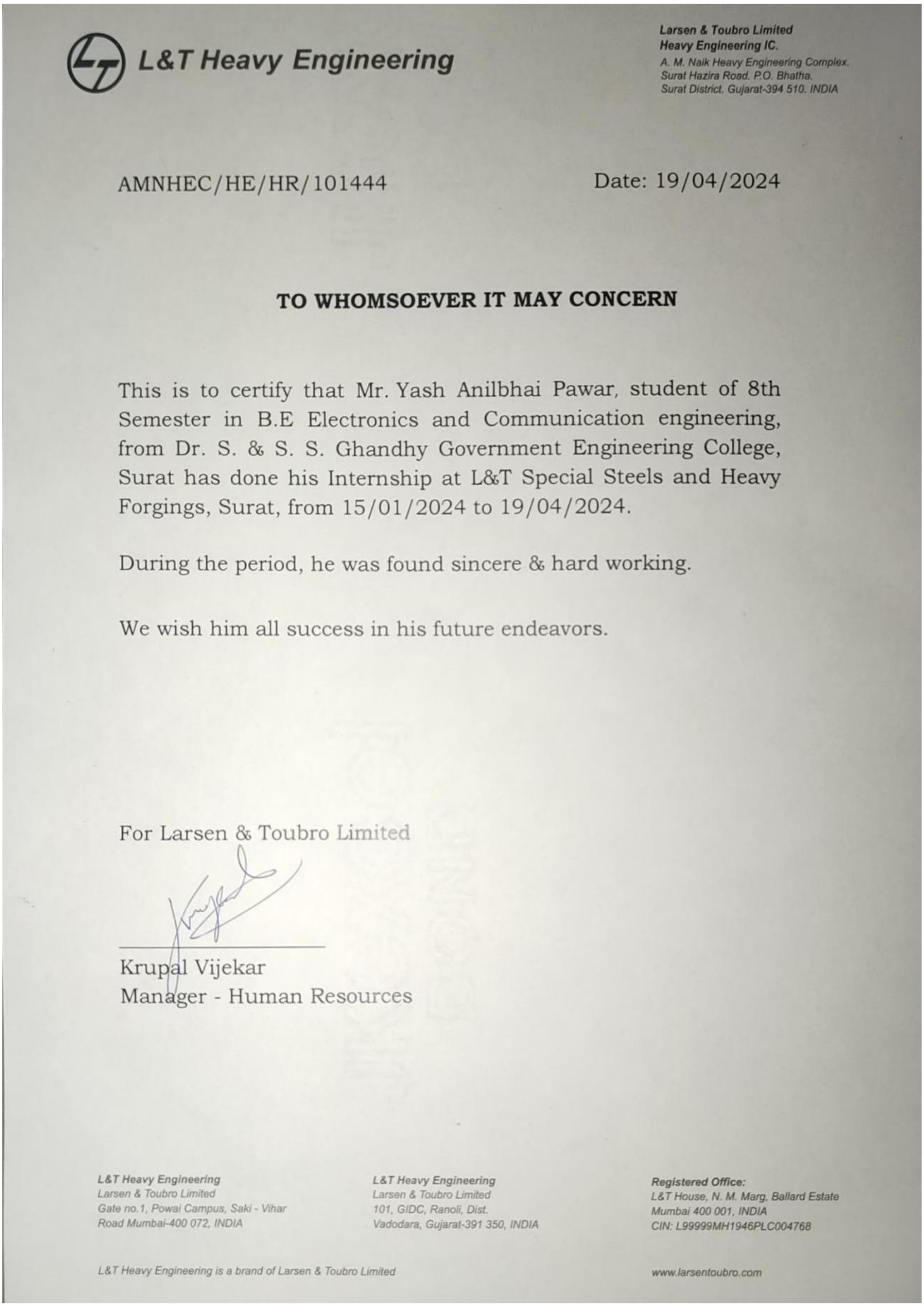
CERTIFICATE

This is to certify that the project report submitted along with the project entitled **IoT enabled Quench Tank Temperature Monitoring** has been carried out by **Pawar Yash Anilbhai** under my guidance in partial fulfillment for the degree of Bachelor of Engineering in Electronics & Communication Engineering, 8th Semester of Gujarat Technological University.

Prof. Kunjal Tandel
(Internal Guide)

Prof. T. P. Dave
(Head of the Department)

Company Certificate



L&T Heavy Engineering

Larsen & Toubro Limited
Heavy Engineering IC.
A. M. Naik Heavy Engineering Complex.
Surat Hazira Road. P.O. Bhatha.
Surat District. Gujarat-394 510. INDIA

AMNHEC/HE/HR/101444 Date: 19/04/2024

TO WHOMSOEVER IT MAY CONCERN

This is to certify that Mr. Yash Anilbhai Pawar, student of 8th Semester in B.E Electronics and Communication engineering, from Dr. S. & S. S. Ghandhy Government Engineering College, Surat has done his Internship at L&T Special Steels and Heavy Forgings, Surat, from 15/01/2024 to 19/04/2024.

During the period, he was found sincere & hard working.

We wish him all success in his future endeavors.

For Larsen & Toubro Limited

Krupal Vijekar
Manager - Human Resources

L&T Heavy Engineering
Larsen & Toubro Limited
Gate no.1, Powai Campus, Saki - Vihar
Road Mumbai-400 072, INDIA

L&T Heavy Engineering
Larsen & Toubro Limited
101, GIDC, Ranoli, Dist.
Vadodara, Gujarat-391 350, INDIA

Registered Office:
L&T House, N. M. Marg, Ballard Estate
Mumbai 400 001, INDIA
CIN: L99999MH1946PLC004768

L&T Heavy Engineering is a brand of Larsen & Toubro Limited

www.larsentoubro.com

Dr. S. & S. S. Ghandhy Government Engineering College, Surat.

DECLARATION

I hereby declare that the Internship report submitted along with the Internship Project entitled **IoT enabled Quenching Tank Temperature Monitoring** submitted in partial fulfillment for the degree of Bachelor of Engineering in Electronics & Communication Engineering to Gujarat Technological University, Ahmedabad, is a bonafide record of original project work carried out by me at L&T Special Steels & Heavy Forgings Pvt. Ltd. under the supervision of Mr. Gopinathan Narayanaswamy and that no part of this report has been directly copied from any students' reports or taken from any other source without providing due reference.

Name of the Student	Signature
Pawar Yash Anilbhai	________________

ACKNOWLEDGEMENT

I wish to express my sincere gratitude to my External Guides , Mr. Gopinathan Narayanaswamy, Mr. Sunil Singh, and Mr. Galav Bhatt for continuously guiding me at the company and answering all my doubts with patience. I would also like to thank my Internal Guide Prof. Kunjal Tandel for helping me through my internship by giving me the necessary suggestions and advice along with their valuable co-ordination in completing this internship.

In addition to that, I would also like to mention the company personals who gave me the permission to use and experience the valuable resources required for the internship.

Thus, in conclusion of the above, I once again thank the other members of Larsen & Toubro Limited for their valuable support in the completion of the internship.

ABSTRACT

This project revolves around the development of an IoT enabled Quench Tank Temperature Monitoring System, undertaken at Larsen & Toubro Limited's Special Steels & Heavy Forgings facility. The current manual monitoring of quenching tanks demands significant operator presence, resulting in time inefficiencies. The proposed system leverages IoT sensors, a PID 500 controller, and a MOXA 5250A Serial Device Server to establish an end-to-end solution. Advanced thermocouples measure tank temperatures, while the PID 500 controller ensures precise regulation. Data is transmitted via a MOXA device and stored in a Microsoft SQL Server. A Python-based dashboard, utilizing Tkinter and Pymodbus, provides real-time insights, optimizing quenching processes. This project aligns with Industry 4.0 practices, promising transformative outcomes in efficiency, cost reduction, and operational excellence.

List of Figures

List of Abbreviations

1. IOT - Internet of Things
2. LTSSHF - Larsen and Toubro Special Steels and Heavy Forgings
3. PID - Proportional Integral Derivative Control
4. SQL - Structured Query Language
5. ITER - International Thermonuclear Experimental Reactor
6. QTTMS - Quenching Tank Temperature Monitoring System
7. RTD - Resistance Temperature Detector
8. SSR - Solid State Relay
9. MODBUS - Modicon Bus
10. CE - Conformité Européenne (French for "European Conformity")
11. RoHS - Restriction of Hazardous Substances
12. GUI - Graphical User Interface
13. OS - Operating System
14. SSMS - SQL Server Management Studio
15. SSL - Secure Socket Layer
16. VM - Virtual Machine
17. SMTP - Simple Mail Transfer Protocol
18. TCP - Transmission Control Protocol
19. IP - Internet Protocol
20. DVCS - Distributed Version Control System

Table of Contents

CHAPTER 1: COMPANY OVERVIEW

1.1 HISTORY

Larsen & Toubro, is an Indian multinational conglomerate that operates in various sectors such as engineering, construction, manufacturing, technology, and finance. It was founded in 1938 in Bombay (now Mumbai) by two Danish engineers, Henning Holck-Larsen and Soren Kristian Toubro.

L&T has grown into one of the largest and most respected companies in India, with a global presence across more than 30 countries. The company's commitment to excellence, innovation, and sustainability has earned it a reputation for delivering world-class projects and solutions across various sectors.

L&T has established itself as a global player in the engineering and construction industry. The company leverages its international expertise, local knowledge, and strategic partnerships to undertake projects and expand its footprint in key markets worldwide. L&T's global reach, combined with its strong track record and reputation for excellence, positions it as a trusted partner for clients seeking reliable and innovative solutions to their most complex challenges.

L&T is committed to sustainability and corporate social responsibility, integrating environmental, social, and governance (ESG) considerations into its business practices. The company prioritizes sustainable development, energy efficiency, waste reduction, and community engagement in its operations.

The company manufactures a wide range of industrial equipment, machinery, and systems across various industries, including heavy engineering, defense, aerospace.

1.2 PRODUCT PORTFOLIO

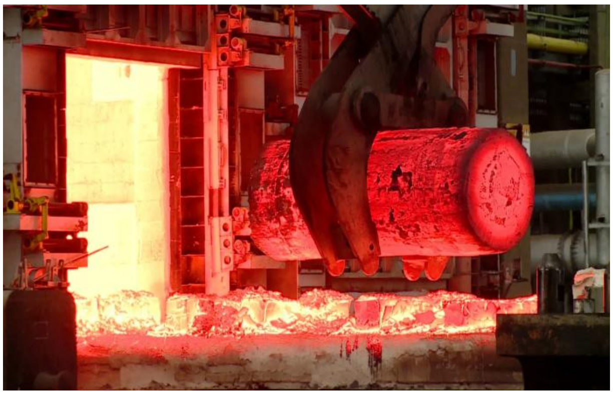


Fig. 1.1 Plant & Facility

L&T Special Steels and Heavy Forgings (LTSSHF) is a joint venture between **Larsen & Toubro Limited (L&T)** and the **Nuclear Power Corporation of India (NPCIL)**. It was commissioned in 2012 with the aim of securing a supply chain of critical forgings for India's nuclear programme along with undertaking development work for the new generation reactors in the nuclear as well as the hydrocarbon sector.

LTSSHF is a state of the art fully integrated facility for manufacturing of heavy forgings . It’s one of its kind in South-East Asia owing to the fact that all operations are under one roof. Our custom-made products will serve the heavy forging requirement for the industrial spectrum from the process plants to the mining industry and nuclear power to hydro power. LTSSHF is committed to providing custom-made steel and heavy forgings that are at par with the best in the world.

It has supplied all critical forged components for the ITER project of international importance and also caters to specialised requirements of Nuclear, Refinery, Petrochemical, Heavy Engineering Industries, Power and the strategic sectors.

1.3 CAPACITY OF PLANT

LTSSHF has a 65000 m^2 integrated facility at the Hazira Manufacturing Complex which offers state-of-the-art melting equipment. They also have one of the world's largest forging presses, capable of producing forgings as large as 5 m in height.

All this is supported by a fully-equipped machine shop and an advanced testing facility with one of the biggest Electric Arc melting furnaces of 125 Tons & the largest Hydraulic press in India (capacity of 9000 Tons).

1.4 DEPARTMENT

The Electrical and Automation department at LTSSHF is responsible for designing, installing, testing, and maintaining the electrical and automation systems for the forging facility. Some of the tasks of the department are:

- Providing power supply and distribution for the melting and forging equipment, such as the electric arc furnace, the hydraulic press, the heat treatment furnace, and the machining shop.
- Developing and implementing automation solutions for the forging process, such as the automatic ingot loading and unloading system, the press control system, the furnace control system, and the material handling system.
- Ensuring the safety and reliability of the electrical and automation systems, by conducting regular inspections, audits, and troubleshooting.

The electrical and automation department at LTSSHF plays a vital role in ensuring the quality and productivity of the heavy forgings, as well as the safety and sustainability of the facility. The department consists of qualified and experienced engineers, technicians, and operators, who work together to deliver the best results for the customers.

The Electrical & Automation Department is also responsible for development and adoption of Industry 4.0 practices to increase efficiency and adopt the best technological practices.

CHAPTER 2: INTERNSHIP PROJECT

2.1 PROBLEM STATEMENT

In the present system, the operator has to physically go to the site where the Quenching Tanks are installed and monitor the process indicators and then plan the Quenching process. A significant amount of time is consumed in this process.

2.2 SOLUTION

The aim of the project is to develop an end-to-end IoT enabled Quenching Tank Tempeature Monitoring System. It will include a dashboard application with connectivity to a SQL Database.

2.3 OUTCOME

The implementation of the IoT-enabled Quenching Tank Temperature Monitoring System is expected to yield transformative outcomes in the industrial setting. By replacing manual monitoring with real-time data collection through advanced IoT sensors, the system promises significant time savings and operational efficiencies.

The instant accessibility of crucial parameters through the user-friendly dashboard facilitates timely decision-making, optimizing quenching processes and potentially reducing operational costs. With improved accuracy and consistency in data, operators can proactively address issues and enhance overall productivity.

The deployment of this system aligns with the broader industry shift towards leveraging IoT technologies, promising a future where industrial processes are not only more streamlined but also technologically advanced, fostering a more efficient and responsive operational environment.

CHAPTER 3: ARCHITECTURE & HARDWARE

3.1 PROJECT ARCHITECTURE

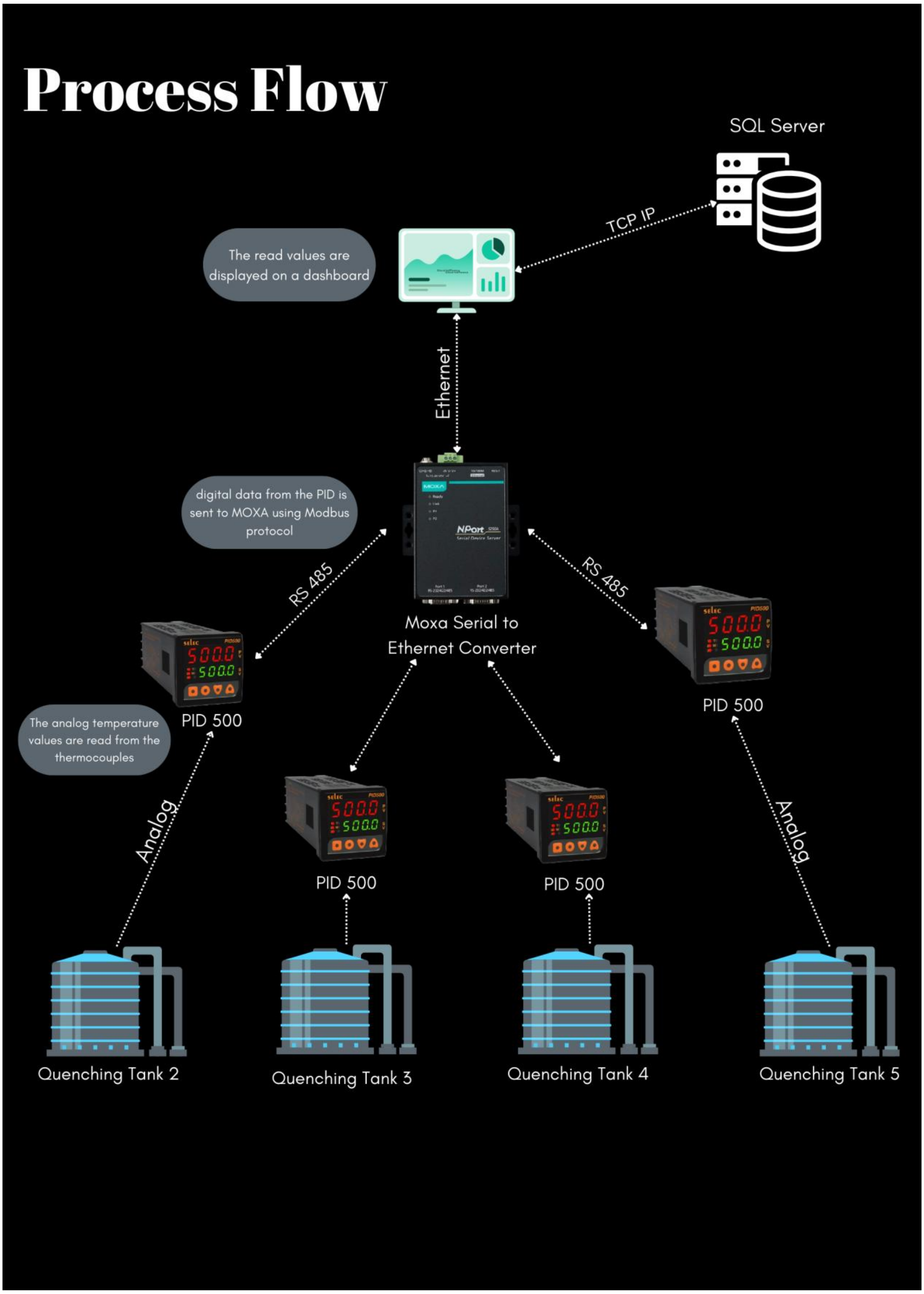

Process Flow
SQL Server
TCP IP
The read values are displayed on a dashboard
Ethernet
digital data from the PID is sent to MOXA using Modbus protocol
RS 485
RS 485
Moxa Serial to Ethernet Converter
PID 500
PID 500
The analog temperature values are read from the thermocouples
Analog
Analog
PID 500
PID 500
Quenching Tank 2
Quenching Tank 3
Quenching Tank 4
Quenching Tank 5

Fig. 3.1 Project Architecture

3.2 QUENCHING TANK

Quenching tanks play a crucial role in the heat treatment process of special steels and heavy forgings, contributing significantly to the enhancement of their mechanical properties and overall performance. Special steels, often alloyed with specific elements to achieve desired properties, and heavy forgings, which are typically large and intricate components, require controlled heat treatment to achieve specific hardness, strength, and durability characteristics. The quenching process involves rapidly cooling the heated steel or forging in a quenching tank filled with a quenching medium, such as oil, water, or polymer solutions. This rapid cooling alters the micro structure of the material, resulting in improved hardness and other mechanical properties.

3.3 THERMOCOUPLE

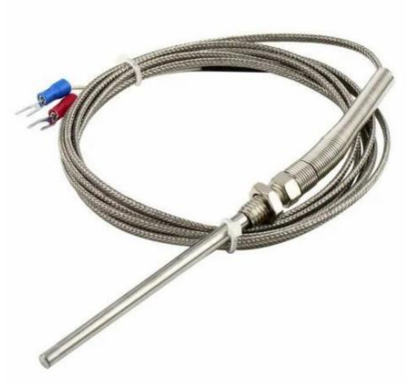


Fig. 3.2 Thermocouple

A thermocouple is a temperature sensor that consists of two different conductive metals joined at one end. The junction of these metals generates a voltage that is proportional to the temperature difference between the junction and the reference point. This phenomenon is known as the Seebeck effect.

Thermocouples are commonly used because they are relatively inexpensive, durable, and can operate in a wide range of temperatures. Different types of thermocouples are available, each with its own temperature range and characteristics, making them suitable for various applications.

In this project thermocouples are used to measure the temperature of the Quenching Tanks.

3.4 SELEC PID 500 CONTROLLER

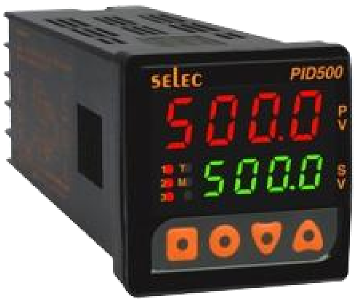


Fig. 3.3 Selec PID 500 Controller

A Selec PID 500 controller is a type of temperature controller that can accept various types of inputs, such as thermocouples, RTDs, or voltage/current signals, and provide different types of outputs, such as relay, SSR, or analog signals. It can perform ON-OFF or auto tune PID control, and has features such as zone PID, single step ramp-soak, and RS485 communication.

The Selec PID 500 controller is significant for this project because it can be used to monitor the temperature of the Quenching Tank.

The specifications of the Selec PID 500 controller are as follows:

- **Display:** Dual 4-digit 7-segment LED, white for process value and green for selectable lower display.
- **Input:** Universal input for 17 types of thermocouples, RTDs, or voltage/current signals.
- **Output:** Relay, SSR, current, or voltage output for control or re-transmission.
- **Control Mode:** ON-OFF or autotune PID control, with zone PID and single step ramp-soak features.
- **Alarm Output**: Relay output for up to 3 alarms, with various alarm modes and hysteresis settings.
- **Communication:** RS485 MODBUS communication (optional).
- **Supply Voltage:** 90 to 270 V AC/DC.
- **Size:** 48 x 48 mm.
- **Certification:** CE, RoHS.

3.5 MOXA NPORT 5250A SERIAL DEVICE SERVER

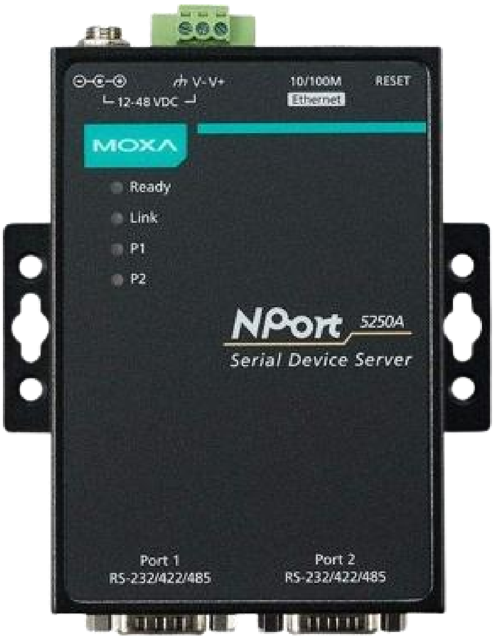


Fig. 3.4 MOXA NPort 5250A Serial Device Server

A MOXA 5250A Serial to Ethernet converter is a device that allows serial devices, such as sensors, meters, or controllers, to communicate with Ethernet networks, such as LANs or the internet. It supports various serial standards, such as RS-232, RS-422, and RS-485, and has features such as surge protection, encryption, and authentication.

The MOXA 5250a Serial to Ethernet converter is significant for this project because it can be used to connect the Selec PID 500.

- **MOXA 5250A Specifications:**
- **Housing:** Metal
- **Dimensions:** 100 x 111 x 26 mm (3.94 x 4.37 x 1.02 in)
- **Weight:** 340 g (0.75 lb)
- **Installation:** Desktop, DIN-rail mounting (with optional kit), wall mounting
- **Serial ports:** 2 RS-232/422/485 ports
- **Ethernet ports:** 1 copper RJ45 port
- **Baud Rate:** 50 to 921.6Kbs
- **Serial connection type:** DB9-M
- **Stop bits:** 1, 1.5, 2

- **Parity:** None, Even, Odd, Space, Mark
- **Surge immunity:** Tested and proven compliant with IEC 61000-4-5

3.6 THE HOST

The host will execute the application program which reads the data from the field devices. The host is important for this project because it allows the user to remotely monitor the quenching process. Also, the connection with the SQL server is managed by the host computer.

3.7 GATEWAY

In the Industrial Internet of Things (IIoT) landscape, gateways serve as vital conduits between the physical and digital realms of industrial operations. Acting as intelligent intermediaries, gateways facilitate the seamless flow of data from myriad sensors, actuators, and devices deployed across factory floors or industrial environments to centralized systems or the cloud.

3.8 SERVER

In our project, servers play a pivotal role in facilitating the deployment and operation of the quenching tank temperature monitoring system. By deploying the system onto an on-premises Level 2 server, we ensure centralized management, robust data storage, and seamless integration within the existing industrial infrastructure.

CHAPTER 4: TECHNOLOGY STACK & TOOLS

4.1 PYTHON

Fig. 4.1 Python Logo

Python is a high-level, interpreted programming language known for its simplicity, versatility, and readability. It was created by Guido van Rossum and first released in 1991.

Key features of Python include:

Simplicity: Python's syntax is straightforward and easy to understand, making it ideal for beginners and experienced programmers alike.

Versatility: Python is a general-purpose language with a wide range of applications, including web development, data analysis, machine learning, artificial intelligence, automation, scripting, and more.

Readability: Python emphasizes code readability and uses indentation to define code blocks, reducing the need for explicit syntax like braces or semicolons.

Interpreted: Python code is executed line by line by an interpreter, eliminating the need for compilation and enabling rapid development and testing.

Dynamic Typing: Python is dynamically typed, meaning that variable types are inferred at runtime and can change during execution.

Extensive Standard Library: Python comes with a comprehensive standard library that provides modules and functions for common programming tasks, reducing the need for external dependencies.

Large Ecosystem: Python has a vibrant ecosystem of third-party libraries and frameworks that extend its functionality and address various programming needs.

Community and Documentation: Python has a large and active community of developers who contribute to its development, share knowledge, and provide support. Extensive documentation and tutorials are available to help developers learn and use Python effectively.

Implementation in project:

The Python pymodbus library is a module that provides a full implementation of the Modbus protocol in Python. It offers both client and server functionalities, as well as various utilities and simulators. The library is significant for this project because it can be used to communicate with the Selec PID 500 controller, which monitors and controls the temperature of the quenching tank, using the RS485 MODBUS protocol. By using the python pymodbus library, the data from the controller can be read and written using simple and consistent methods, and the data can be encoded and decoded using the payload builder and decoder features. Other modules used with python will be discussed further.

4.2 TKINTER TOOLKIT

Fig. 4.2 Tkinter Logo

Tkinter is a Python binding to the Tk GUI toolkit. It is the standard Python interface to the Tk GUI toolkit, and is Python's standard GUI. Tkinter is included with standard Linux, Microsoft Windows and macOS installs of Python. This toolkit was used to create the dashboard application.

Key features of Tkinter include:

Cross-Platform Compatibility: Tkinter is included with standard Python distributions for Windows, macOS, and Linux, ensuring that applications developed with Tkinter can run seamlessly on different platforms without modification.

Ease of Use: Tkinter's simple and easy-to-understand API allows developers to quickly create GUI applications with minimal code. It provides a wide range of built-in widgets and layout managers for designing user interfaces.

Customizable Widgets: Tkinter offers a variety of customizable widgets, including buttons, labels, entry fields, checkboxes, radio buttons, listboxes, and more. Developers can customize the appearance and behavior of these widgets to suit their application's requirements.

Event-Driven Programming: Tkinter follows an event-driven programming model, where user interactions and system events (such as button clicks or mouse movements) trigger event handlers. This allows developers to create interactive and responsive GUI applications.

Integration with Python: As a Python library, Tkinter seamlessly integrates with other Python libraries and frameworks, making it easy to incorporate functionality from third-party modules into Tkinter applications.

Support for Internationalization: Tkinter also supports internationalization and localization by providing built-in support for Unicode text rendering and localization of GUI elements.

Community and Documentation: Tkinter has a large and active community of developers, providing extensive documentation, tutorials, and resources to help developers learn and use Tkinter effectively.

4.3 STRUCTURED QUERY LANGUAGE

SQL (Structured Query Language) is a domain-specific language used for managing and manipulating relational databases. It provides a standardized syntax and set of commands for performing various operations on databases, such as querying data, modifying data, creating and modifying database schemas, and managing database users and permissions.

Key aspects of SQL:

Data Querying: SQL allows users to retrieve data from databases using SELECT statements. Queries can filter, sort, and aggregate data to meet specific criteria.

Data Manipulation: SQL supports commands for modifying data in databases, such as INSERT (to add new records), UPDATE (to modify existing records), and DELETE (to remove records).

Data Control: SQL provides commands for managing database users, permissions, and access control. Users can grant or revoke privileges to control who can access and modify data in the database.

Data Integrity: SQL includes mechanisms for enforcing data integrity constraints, such as primary key constraints, foreign key constraints, unique constraints, and check constraints. These constraints ensure the consistency and correctness of data stored in the database.

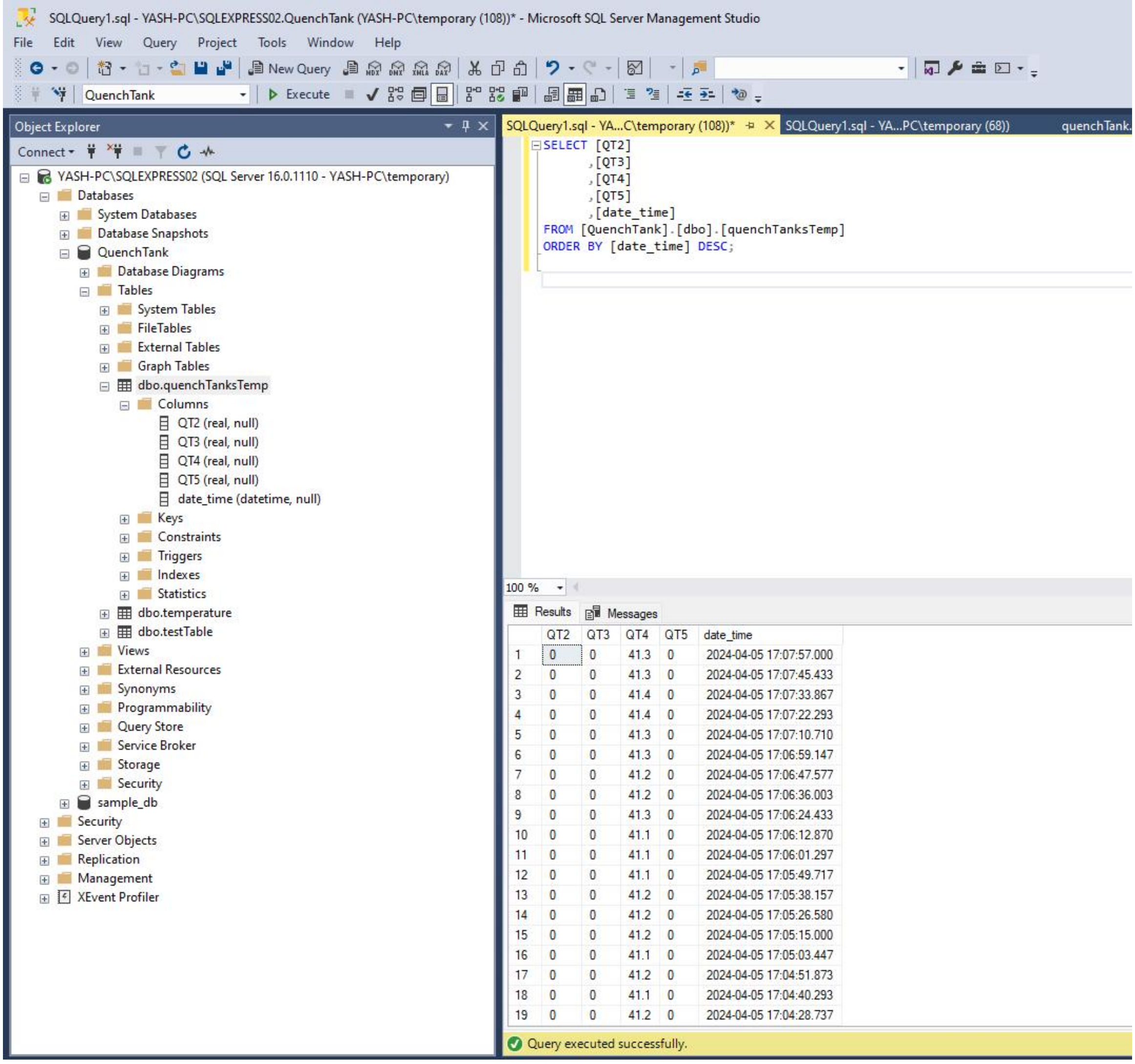
Transaction Management: SQL supports transactions, which allow multiple database operations to be grouped together and executed atomically. Transactions ensure data consistency and integrity by either committing all changes or rolling back to a previous state in case of errors or failures.

Data Aggregation and Analysis: SQL provides functions for aggregating and analyzing data, such as SUM, AVG, COUNT, MAX, and MIN. These functions are used to perform calculations and summarize data for reporting and analysis purposes.

SQL is widely used in various industries and applications, including web development, enterprise software, data analytics, business intelligence, and more. It is supported by most relational database management systems (RDBMS) such as MySQL, PostgreSQL, Oracle Database, Microsoft SQL Server, and SQLite, making it a versatile and essential tool for working with relational databases.

4.4 MICROSOFT SQL SERVER

Fig. 4.3 Microsoft SQL Server Logo



```
SELECT [QT2]
      ,[QT3]
      ,[QT4]
      ,[QT5]
      ,[date_time]
FROM [QuenchTank].[dbo].[quenchTanksTemp]
ORDER BY [date_time] DESC;
```

	QT2	QT3	QT4	QT5	date_time
1	0	0	41.3	0	2024-04-05 17:07:57.000
2	0	0	41.3	0	2024-04-05 17:07:45.433
3	0	0	41.4	0	2024-04-05 17:07:33.867
4	0	0	41.4	0	2024-04-05 17:07:22.293
5	0	0	41.3	0	2024-04-05 17:07:10.710
6	0	0	41.3	0	2024-04-05 17:06:59.147
7	0	0	41.2	0	2024-04-05 17:06:47.577
8	0	0	41.2	0	2024-04-05 17:06:36.003
9	0	0	41.3	0	2024-04-05 17:06:24.433
10	0	0	41.1	0	2024-04-05 17:06:12.870
11	0	0	41.1	0	2024-04-05 17:06:01.297
12	0	0	41.1	0	2024-04-05 17:05:49.717
13	0	0	41.2	0	2024-04-05 17:05:38.157
14	0	0	41.2	0	2024-04-05 17:05:26.580
15	0	0	41.2	0	2024-04-05 17:05:15.000
16	0	0	41.1	0	2024-04-05 17:05:03.447
17	0	0	41.2	0	2024-04-05 17:04:51.873
18	0	0	41.1	0	2024-04-05 17:04:40.293
19	0	0	41.2	0	2024-04-05 17:04:28.737

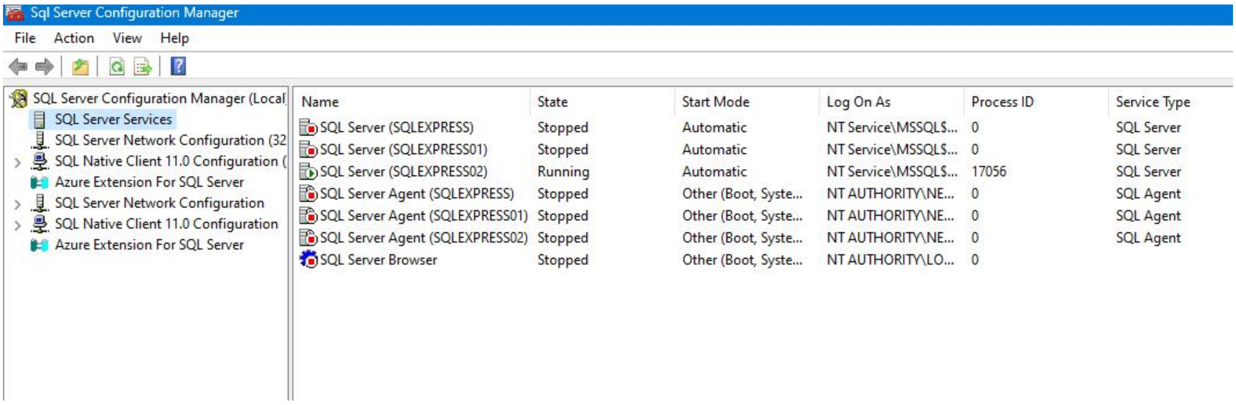
✓ Query executed successfully.

Fig. 4.4 SQL Server Management Studio 2019

4.4.1 SQL Server Management Studio 2019:

SQL Server Management Studio 2019 is a robust software application designed for managing SQL Server databases efficiently. Upon launching the application, users are greeted with a familiar interface comprising various panels and toolbars. One of the key features of SSMS is its Object Explorer panel, situated on the left-hand side, offering a hierarchical view of servers, databases, and other objects.

4.4.2 SQL Server Configuration Manager:


Sql Server Configuration Manager
File
Action
View
Help
SQL Server Configuration Manager (Local
SQL Server Services
SQL Server Network Configuration (32
SQL Native Client 11.0 Configuration (
Azure Extension For SQL Server
SQL Server Network Configuration
SQL Native Client 11.0 Configuration
Azure Extension For SQL Server

Name	State	Start Mode	Log On As	Process ID	Service Type
SQL Server (SQLEXPRESS)	Stopped	Automatic	NT Service\MSSQL$...	0	SQL Server
SQL Server (SQLEXPRESS01)	Stopped	Automatic	NT Service\MSSQL$...	0	SQL Server
SQL Server (SQLEXPRESS02)	Running	Automatic	NT Service\MSSQL$...	17056	SQL Server
SQL Server Agent (SQLEXPRESS)	Stopped	Other (Boot, Syste...	NT AUTHORITY\NE...	0	SQL Agent
SQL Server Agent (SQLEXPRESS01)	Stopped	Other (Boot, Syste...	NT AUTHORITY\NE...	0	SQL Agent
SQL Server Agent (SQLEXPRESS02)	Stopped	Other (Boot, Syste...	NT AUTHORITY\NE...	0	SQL Agent
SQL Server Browser	Stopped	Other (Boot, Syste...	NT AUTHORITY\LO...	0	

Fig. 4.5 SQL Server Configuration Manager

SQL Server Configuration Manager is an essential tool provided by Microsoft for efficiently managing various aspects of SQL Server instances on a Windows system. Accessible through the Microsoft SQL Server folder in the Start menu, it falls under the "Configuration Tools" section.

Upon opening, users are presented with a straightforward interface offering functionalities to manage SQL Server services, network protocols, and service accounts.

Through SQL Server Configuration Manager, administrators can control the lifecycle of SQL Server services, including starting, stopping, pausing, resuming, or restarting them as needed. This feature ensures the smooth operation of SQL Server instances and enables administrators to troubleshoot service-related issues efficiently.

Moreover, the tool allows for the configuration of network protocols utilized by SQL Server for communication. Administrators can enable or disable protocols such as TCP/IP, Named Pipes, and Shared Memory based on their specific requirements. SQL Server Configuration Manager facilitates the assignment of port numbers for each protocol, providing flexibility in network configuration and enhancing security.

4.5 TOOLS

4.5.1 Visual Studio Code:

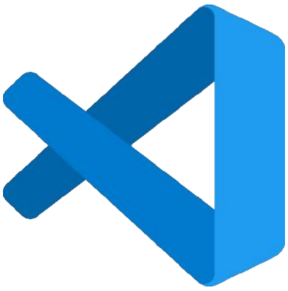
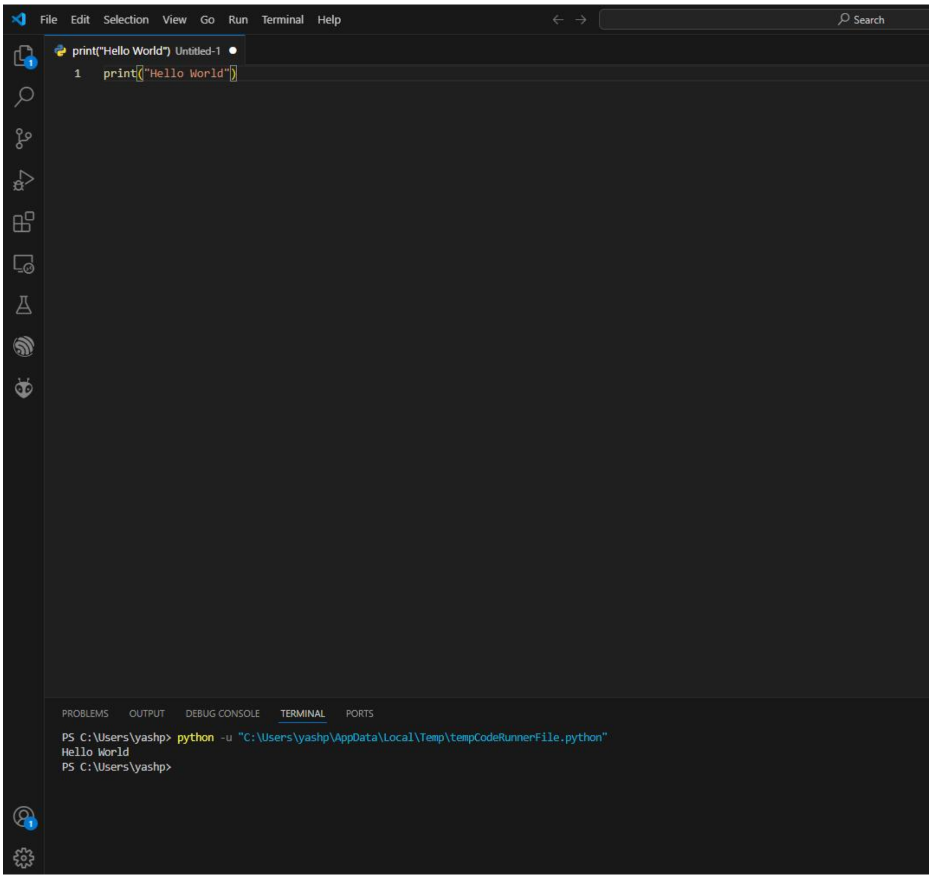


Fig. 4.6 Visual Studio Code Logo

Visual Studio Code (VS Code) is a widely favored Integrated Development Environment among developers for several reasons. Firstly, it's freely available across multiple platforms, including Windows, macOS, and Linux, ensuring consistency in development environments regardless of the operating system. This accessibility has contributed significantly to its popularity.



```
print("Hello World")
```

```
PS C:\Users\yashp> python -u "C:\Users\yashp\AppData\Local\Temp\tempCodeRunnerFile.python"
Hello World
PS C:\Users\yashp>
```

Fig. 4.7 Visual Studio Code IDE

One of the key features of VS Code is its extensibility and customizability. With a rich ecosystem of extensions, developers can tailor their environment to suit their specific needs.

Whether it's support for different programming languages, frameworks, or tools, VS Code's extensions marketplace offers a wide array of options to enhance productivity and streamline workflows.

Moreover, VS Code provides intelligent code editing features such as syntax highlighting, IntelliSense (code completion), and code refactoring tools. These features help developers write code faster and with fewer errors, thereby boosting efficiency and reducing debugging time.

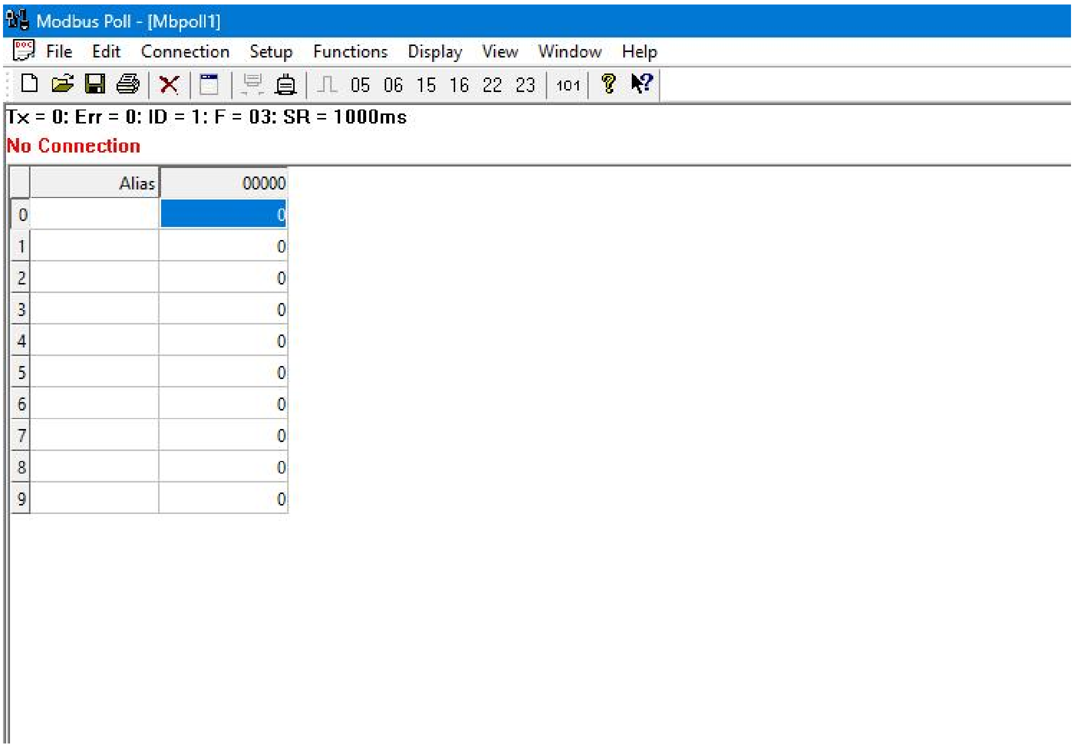
Another advantage of VS Code is its seamless integration with Git, a popular version control system.

The built-in Git integration allows developers to perform version control operations directly within the editor, including committing changes, viewing diffs, and managing branches. This tight integration simplifies the development workflow and promotes collaboration among team members.

Additionally, VS Code comes with an integrated terminal, enabling developers to run commands, scripts, and terminal-based tools without switching to a separate terminal window. This feature enhances convenience and productivity by keeping all development tasks within a single interface.

4.5.2 Modbus Poll Software

Modbus Poll is a vital software tool for industrial automation, used for testing, monitoring, and debugging Modbus communication protocols. It facilitates seamless communication between devices like PLCs, RTUs, sensors, and actuators by providing users with a platform to perform various tasks related to Modbus communication.



Tx = 0: Err = 0: ID = 1: F = 03: SR = 1000ms

No Connection

	Alias	00000
0		0
1		0
2		0
3		0
4		0
5		0
6		0
7		0
8		0
9		0

Fig. 4.8 Modbus Poll Software

One of the primary functions of Modbus Poll is its ability to simulate both master and slave devices, allowing users to emulate Modbus communication between different devices for testing purposes. This feature is particularly useful during the development and validation phases of industrial automation projects, where ensuring the compatibility and reliability of Modbus communication is crucial.

Another key feature of Modbus Poll is its capability to monitor real-time data values from Modbus registers. By visualizing the status of connected devices and monitoring data in real-time, users can gain insights into the performance and behavior of their industrial systems. This feature facilitates troubleshooting and enables users to identify and address communication issues promptly.

Modbus Poll also offers data logging functionality, allowing users to log Modbus data over time. This capability is valuable for analyzing trends, diagnosing intermittent issues, and generating reports for system optimization. By logging Modbus data, users can gain a deeper understanding of their industrial processes and make informed decisions to improve efficiency and reliability.

4.5.3 Version Control

Git:

Fig. 4.9 Git Logo

Git is a distributed version control system (DVCS) widely used for tracking changes in source code during software development.

Version Control: Git allows developers to track changes to their codebase over time, enabling collaboration, code review, and the ability to revert to previous versions if needed. It provides a complete history of changes made to files, along with metadata such as authorship and timestamps.

Distributed: Git is a distributed version control system, meaning that each developer has a complete copy of the repository, including the entire history of changes. This enables developers to work offline, commit changes locally, and synchronize their work with remote repositories when they have an internet connection.

Branching and Merging: Git's branching and merging capabilities are central to its workflow. Developers can create lightweight branches to work on new features or bug fixes independently of the main codebase. Branches can be merged back into the main branch (typically 'master' or 'main') once work is complete, allowing for parallel development and experimentation.

Collaboration: Git facilitates collaboration among developers by providing mechanisms for sharing code changes and reviewing contributions. Developers can push their changes to a shared repository hosted on platforms like GitHub or GitLab, where they can be reviewed, discussed.

Performance: Git is known for its speed and efficiency, even when dealing with large codebases. Its use of content-addressable storage and sophisticated algorithms for diffing and merging enable fast operations, even on repositories with thousands of files.

GitHub:

Fig. 4.10 GitHub Logo

GitHub is a web-based platform for hosting Git repositories and fostering collaboration among developers. It offers tools for code review, issue tracking, and pull requests, as well as project management features like project boards and milestones.

GitHub was utilized in the IoT-enabled Quenching Tank Temperature Monitoring project for efficient development and collaboration. It offers version control, allowing tracking of code changes and facilitates collaboration among team members. With a centralized repository, all team members have access to the latest codebase, eliminating version conflicts. GitHub's issue tracking system aids project management by organizing tasks and monitoring progress. The URL to the project’s GitHub has been added to the refernces page.

Integration with CI/CD tools enables automated build and deployment processes. Additionally, GitHub provides documentation features, fostering knowledge sharing among team members. Overall, GitHub enhances development, collaboration, version control, and project management processes in the project.

4.5.4 Virtualization

Virtualization is the process of creating a virtual (rather than actual) version of something, such as an operating system, a server, a storage device, or a network resource. In the context of computing, virtualization refers to the creation of virtual instances of computer resources that mimic the functionality of physical hardware or software.

Environment Isolation: Virtualization allows you to create separate environments for different projects or testing scenarios. This can help prevent conflicts between different sets of dependencies or system configurations.

This enables testers to accurately replicate various configurations, operating systems, and hardware setups without impacting the actual production systems. Each virtual environment can be customized to test specific combinations of operating systems, software versions, and configurations.

Compatibility Testing: Virtualization enables testers to validate the compatibility of applications across different operating systems, platforms, and devices. By creating virtual machines with various configurations, testers can verify that the software functions correctly and consistently across a wide range of environments.

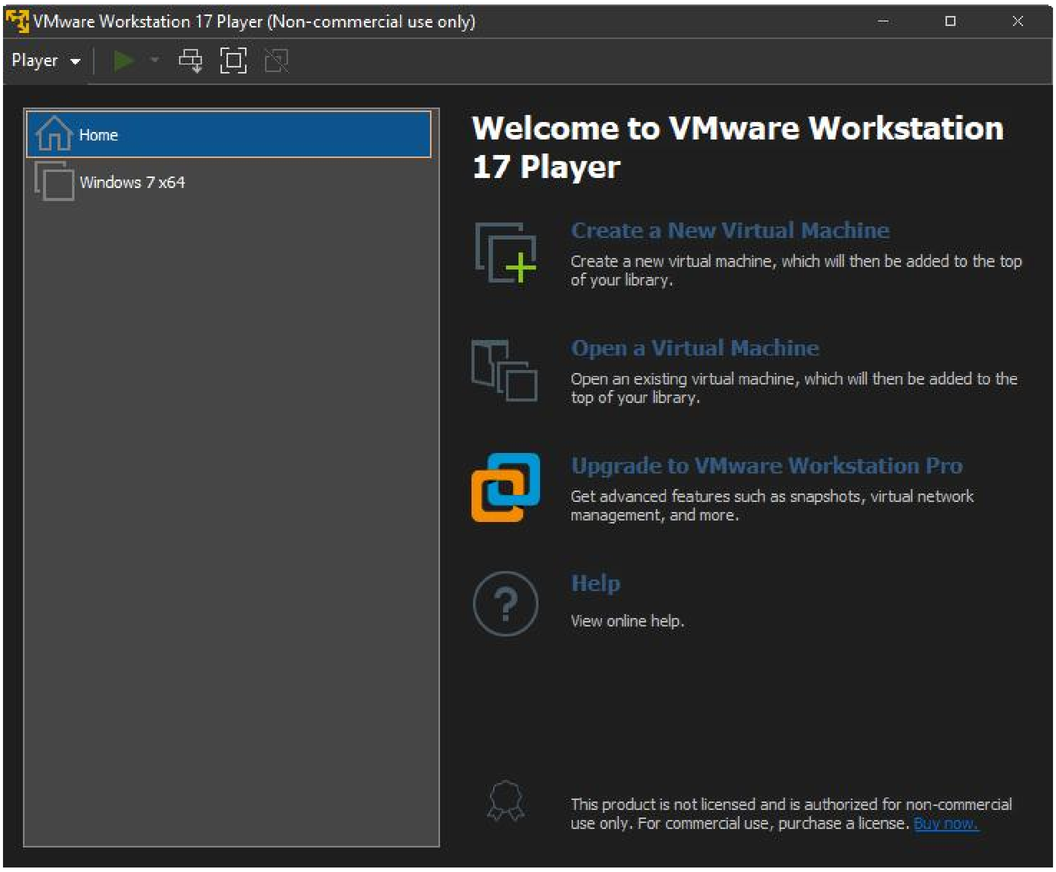
VMWare Workstation 17 Player:

Fig. 4.11 VMWare Workstation 17 Player Logo

VMware Workstation Player is a desktop virtualization application that allows users to run multiple operating systems on a single physical machine. It is designed primarily for personal use, enabling users to create, run, and evaluate virtual machines for educational, testing, or development purposes.

VMs provide a reliable and reproducible development environment, ensuring consistency across different stages of the software development life cycle. Developers can create a virtual machine with the necessary operating system, development tools, and libraries required for building and testing the monitoring software, eliminating potential compatibility issues and dependencies on individual developer workstations

VMs enable sandboxed testing and experimentation, allowing developers to simulate various scenarios and configurations without impacting the production environment. In the case of quenching tank temperature monitoring software, developers can deploy multiple virtual machines to emulate different quenching processes, temperature profiles, and sensor configurations, facilitating comprehensive testing and validation of the software's functionality and performance under diverse conditions.



Welcome to VMware Workstation 17 Player

Create a New Virtual Machine

Create a new virtual machine, which will then be added to the top of your library.

Open a Virtual Machine

Open an existing virtual machine, which will then be added to the top of your library.

Upgrade to VMware Workstation Pro

Get advanced features such as snapshots, virtual network management, and more.

Help

View online help.

This product is not licensed and is authorized for non-commercial use only. For commercial use, purchase a license. Buy now.

Fig. 4.12 VMWare Workstation 17 Player

The QTTMS Software was also developed and tested on a Windows 7 Ultimate virtual machine to test its backwards compatibility feature.

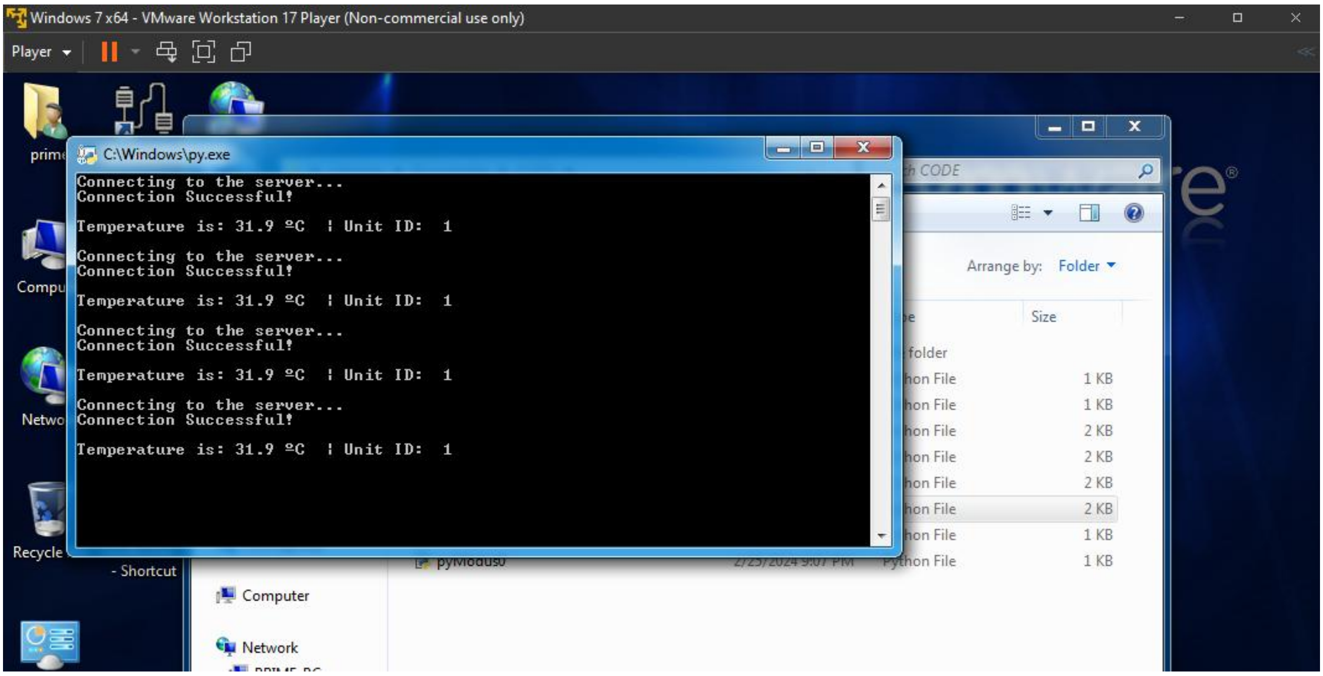
CHAPTER 5: DEVELOPMENT

5.1 PHASE-1: DATA ACQUISITION

The project execution is planned in a phased manner. The Phase-1 incorporates Data Acquisition from the PID Controllers which will be connected with the thermocouple. The data will be read using the MODBUS communication protocol.

Fig. 5.1 Modbus Protocol Logo

In short, Modbus is a communication protocol used in industrial automation. It facilitates communication between electronic devices in a master-slave architecture, supporting both serial and Ethernet connections. Data is organized into coils, discrete inputs, input registers, and holding registers, with function codes specifying actions. Each device has a unique address, and error handling is included. Its openness enables interoperability among devices from different manufacturers.



```
Connecting to the server...
Connection Successful!

Temperature is: 31.9 ºC  ¦ Unit ID:  1

Connecting to the server...
Connection Successful!

Temperature is: 31.9 ºC  ¦ Unit ID:  1

Connecting to the server...
Connection Successful!

Temperature is: 31.9 ºC  ¦ Unit ID:  1

Connecting to the server...
Connection Successful!

Temperature is: 31.9 ºC  ¦ Unit ID:  1
```

Fig. 5.2 Console Application

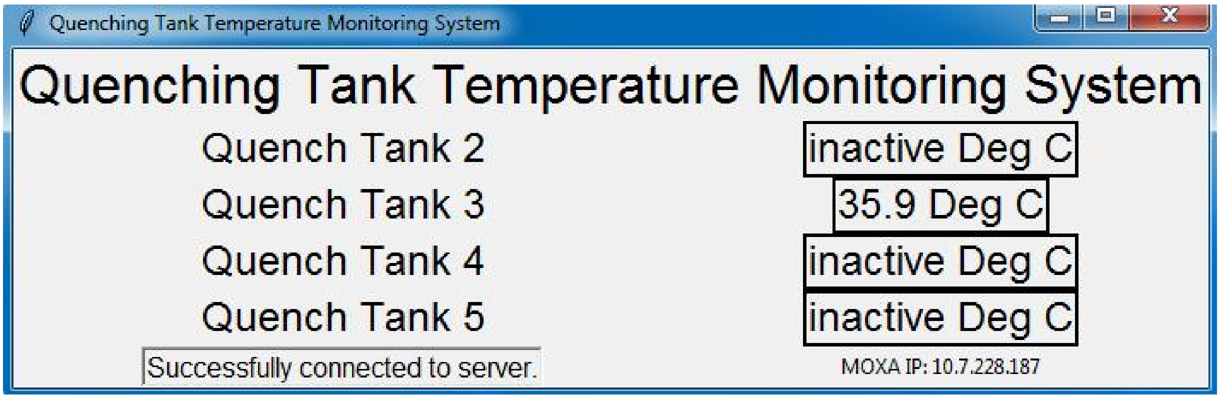
The console application demonstrates the connectivity between the host and the serial device. It displays the temperature from one of the PIDs along with the unit ID. The application was tested on Windows 7 to ensure backwards compatibility.

PyModbus is a Python library that provides a convenient way to implement Modbus communication in Python-based applications.

It allows you to create Modbus client and server applications, making it easy to interact with Modbus-compatible devices such as PLCs (Programmable Logic Controllers), HMIs (Human Machine Interfaces), and other industrial automation equipment.

5.2 PHASE-2: GUI DEVELOPMENT

5.2.1 Application Version 1.0



Quenching Tank Temperature Monitoring System

Quenching Tank Temperature Monitoring System

Quench Tank 2 inactive Deg C

Quench Tank 3 35.9 Deg C

Quench Tank 4 inactive Deg C

Quench Tank 5 inactive Deg C

Successfully connected to server.

MOXA IP: 10.7.228.187

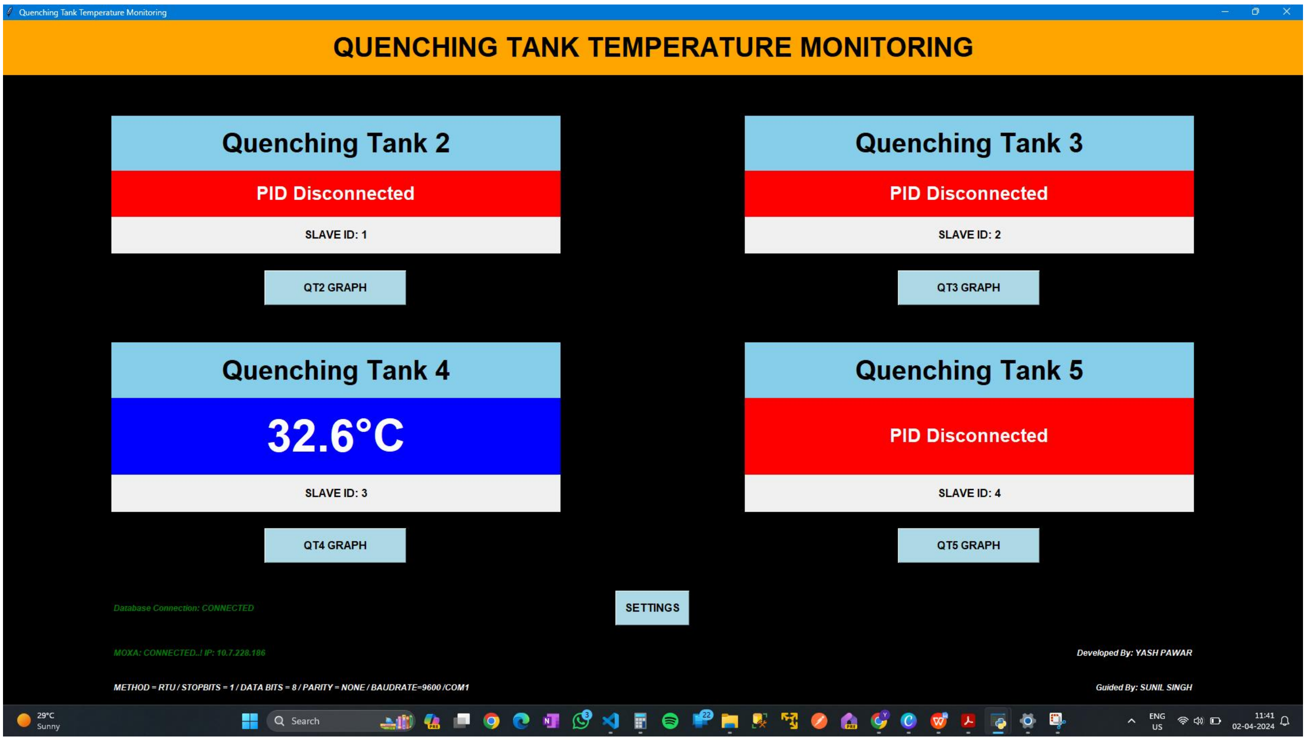
Fig. 5.3 Application Version 1.0

The version 1.0 of the GUI showed the temperature readings of individual tanks, the IP address of the MOXA Server. It also displays the connectivity status of the MOXA server.

5.3 PHASE-3: GUI UPDATE

The phase 3 included a complete user interface overhaul. After exploring multiple frontend options availabe in Python such as the PyQt and WixPython. Keeping rapid prototyping in mind, the Tkinter GUI Toolkit was selected.

The latest update to our system introduces several key enhancements aimed at improving functionality, user experience, and system reliability. One notable addition is the integration of real-time graphing using the Matplotlib library in Python, providing users with dynamic visualizations of data trends. Furthermore, the inclusion of a PID Controller disconnection indication enhances system monitoring and maintenance, ensuring uninterrupted operation. Additionally, the display of slave IDs for connected nodes enhances transparency and facilitates troubleshooting processes. These updates collectively contribute to a more robust and user-friendly system, aligning with our commitment to delivering high-quality solutions to our stakeholders.



Quenching Tank Temperature Monitoring

QUENCHING TANK TEMPERATURE MONITORING

Quenching Tank 2
PID Disconnected
SLAVE ID: 1

Quenching Tank 3
PID Disconnected
SLAVE ID: 2

Quenching Tank 4
32.6°C
SLAVE ID: 3

Quenching Tank 5
PID Disconnected
SLAVE ID: 4

Database Connection: CONNECTED

MOXA: CONNECTED..! IP: 10.7.228.186

METHOD = RTU / STOPBITS = 1 / DATA BITS = 8 / PARITY = NONE / BAUDRATE=9600 /COM1

Developed By: YASH PAWAR

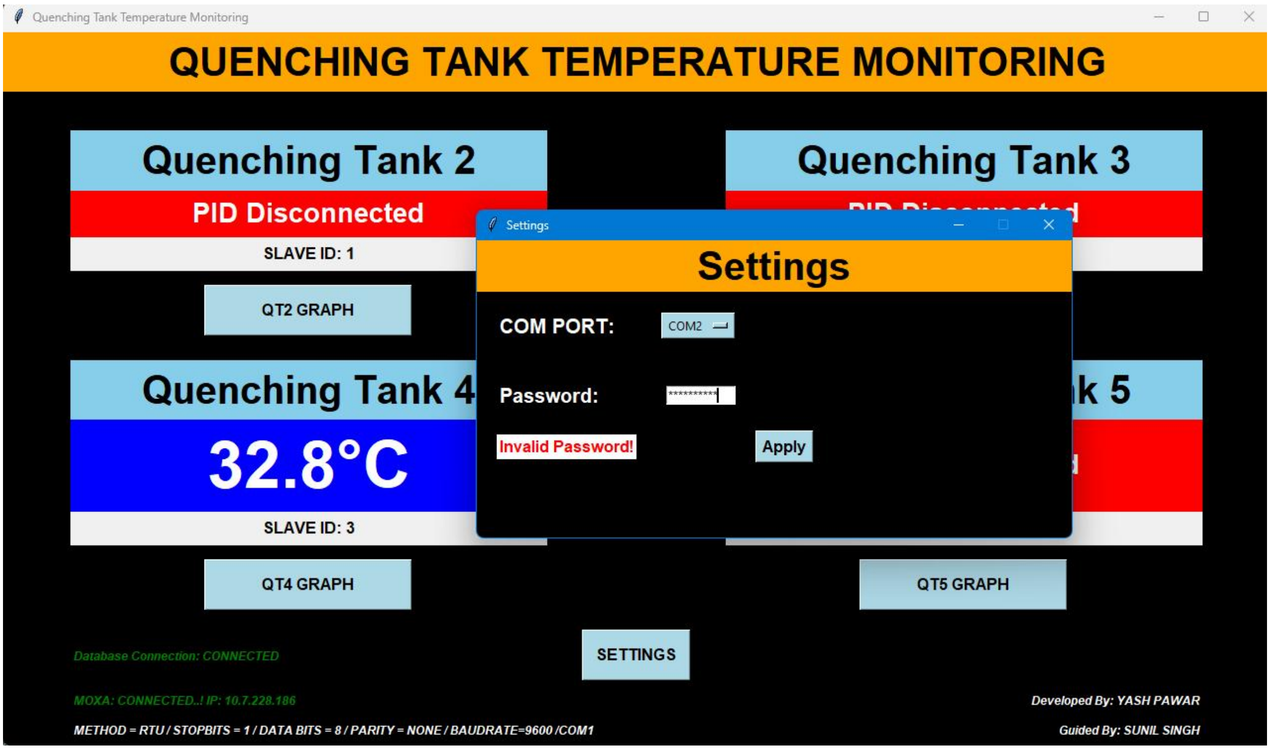
Guided By: SUNIL SINGH

Fig. 5.4 Application Version 2.0

Features of version 2.0:

- Backwards compatibility with Windows 7/10.
- Refreshed layout.
- Database connectivity status.
- Real time graphs.
- PID connectivity status.
- MOXA connectivity status.
- Password authentication.
- COM port configuration in settings.

5.3.1 Settings Menu



COM PORT: COM2

Password: **********

Invalid Password!

Database Connection: CONNECTED

MOXA: CONNECTED..! IP: 10.7.228.186

METHOD = RTU / STOPBITS = 1 / DATA BITS = 8 / PARITY = NONE / BAUDRATE=9600 /COM1

Developed By: YASH PAWAR

Guided By: SUNIL SINGH

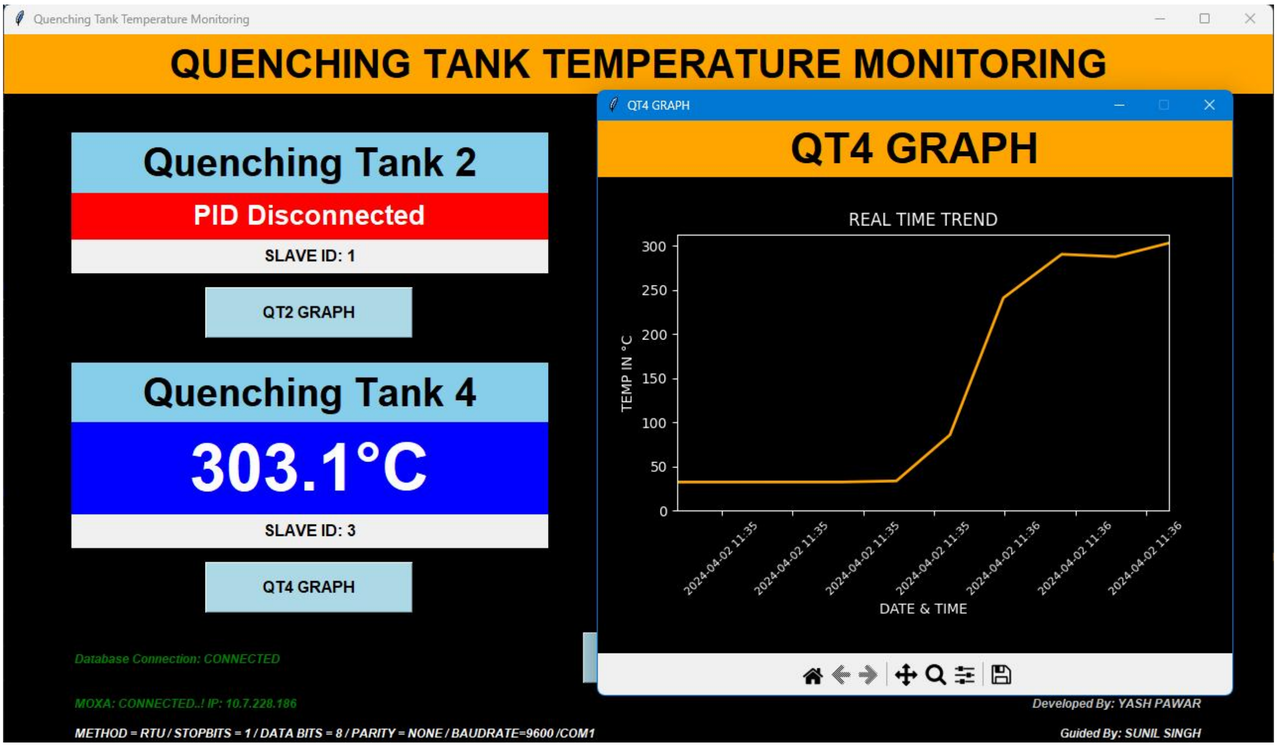
Fig. 5.5 Settings Configuration Window

The settings window includes setup of the COM port for the serial device. It includes password based authentication to prevent unauthorized changes of the COM port. The application remembers the saved settings and stores them in a settings.json file.

5.3.2 Real Time Graphs

Fig. 5.6 Matplotlib Logo

To plot real time graphs, the mattplotlib library was used. Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK.



Database Connection: CONNECTED

MOXA: CONNECTED..! IP: 10.7.228.186

METHOD = RTU / STOPBITS = 1 / DATA BITS = 8 / PARITY = NONE / BAUDRATE=9600 /COM1

Developed By: YASH PAWAR

Guided By: SUNIL SINGH

Fig. 5.7 Real-time Graph

5.3.3 Multithreading

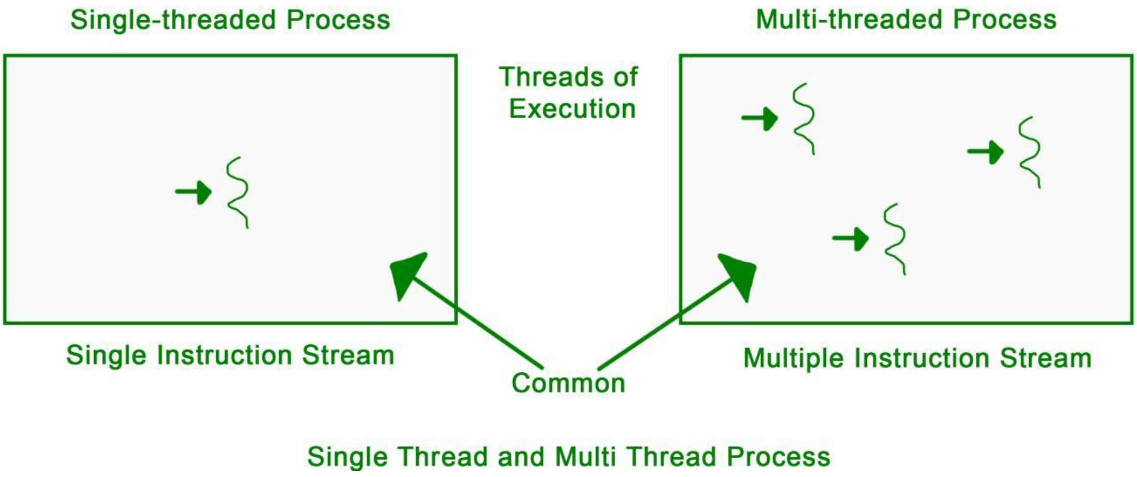

Single-threaded Process
Multi-threaded Process
Threads of Execution
Single Instruction Stream
Multiple Instruction Stream
Common
Single Thread and Multi Thread Process

Fig. 5.8 Multithreading Illustration

Multithreading is a programming concept that allows concurrent execution of multiple threads within a single process. A thread is a lightweight process that can perform tasks independently, allowing programs to perform multiple operations simultaneously and efficiently utilize the available CPU resources.

Multithreading is commonly used in applications where concurrent tasks need to be executed concurrently or where responsiveness and performance are critical. Some common use cases for multithreading include:

Responsive User Interfaces: Multithreading is often used in graphical user interface (GUI) applications to

ensure responsiveness. By offloading time-consuming tasks, such as file I/O or network operations, to separate threads, the main UI thread remains responsive, providing a smooth user experience.

Parallel Processing: Multithreading can be used to divide a task into smaller subtasks that can be executed concurrently, leveraging the processing power of multicore CPUs. This approach is commonly used in applications that require parallel processing, such as multimedia processing, data analysis, and scientific computing.

Asynchronous Programming: Multithreading enables asynchronous programming models, where tasks can be executed concurrently without blocking the main thread. Asynchronous operations are commonly used in web development, networking, and I/O-bound applications to improve performance and scalability.

Server-Side Applications: Multithreading is widely used in server-side applications, such as web servers and application servers, to handle multiple client requests simultaneously. Each client request can be processed by a separate thread, allowing the server to handle a large number of concurrent connections efficiently.

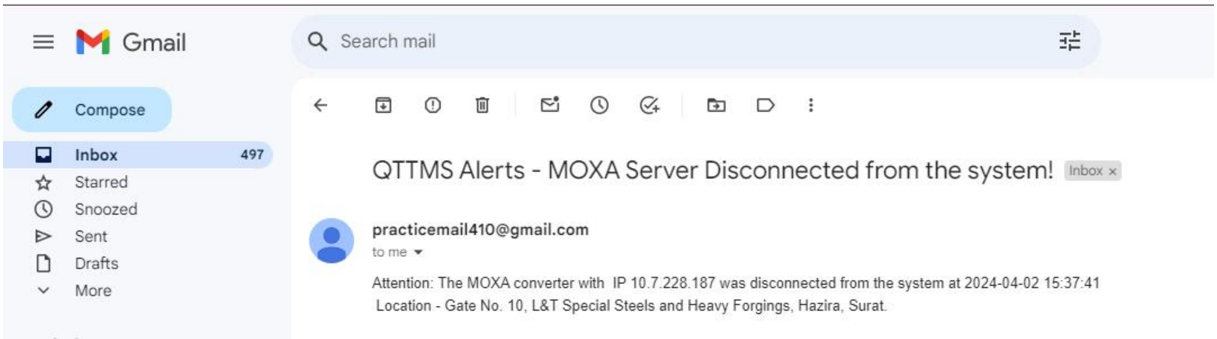
Implementing multithreading in programming languages typically involves creating and managing threads, synchronization mechanisms to coordinate access to shared resources, and handling thread communication and synchronization issues to avoid race conditions and deadlocks.

While multithreading can improve performance and concurrency, it also introduces challenges such as thread safety, synchronization overhead, and potential issues like race conditions and deadlocks. Therefore, careful design and implementation are essential to ensure correct and efficient multithreaded programs. Additionally, modern programming languages and frameworks often provide higher-level abstractions and libraries to simplify multithreading and mitigate these challenges.

As the application involves multiple processes running concurrently for e.g. temperature reading, dumping data to the database, GUI update. So, for achieving concurrency, a **multithreaded** approach was selected so that the main thread is not blocked. The GUI will keep updating with the latest fetched values from the PIDs along with that the database will fed with the updated values in the background.

The values read from the PIDs are stored in a Queue data structure to ensure a thread safe operation. As the temperature variables are shared between the read temperature, database dump and the realtime graph function threads this step was crucial so that smooth functionality of the application is maintained.

5.3.4 Email Alerts



QTTMS Alerts - MOXA Server Disconnected from the system!

practicemail410@gmail.com

to me

Attention: The MOXA converter with IP 10.7.228.187 was disconnected from the system at 2024-04-02 15:37:41

Location - Gate No. 10, L&T Special Steels and Heavy Forgings, Hazira, Surat.

Fig. 5.9 Email Alerts

As required by the client, email alerts functionality was added. An alert mail would be triggered to a concerned email address when the MOXA server is physically disconnected from the gateway or a malfunction occurs in the MOXA server itself.

SMTP Protocol:

The **Simple Mail Transfer Protocol**, is a standard protocol used for sending and receiving email messages on the internet. It is part of the application layer of the TCP/IP protocol suite, which governs how data is transmitted over networks. SMTP is primarily used for sending emails from a client (such as an email program) to a server (such as an email server) or between email servers.

SSL and SMTPLIB Modules:

The ssl and smtplib modules in Python are used in the project for securely sending emails over networks using SMTP servers.

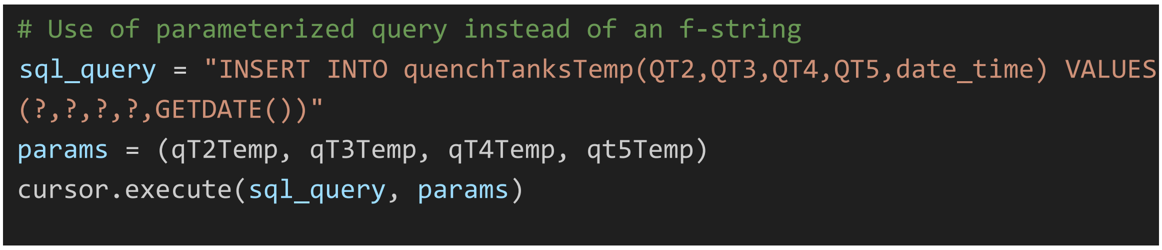
SSL: This module provides functionality for working with SSL (Secure Sockets Layer) and TLS (Transport Layer Security) protocols. It enables the creation of SSL/TLS encrypted connections, ensuring secure communication over networks.

It ensures the confidentiality and integrity of data transmitted between a client and a server by encrypting the information exchanged between them. SSL is commonly used to secure sensitive transactions on the internet, such as online banking, e-commerce transactions, and secure email communication.

smtplib: This module facilitates sending emails via SMTP servers. It offers a convenient interface for connecting to SMTP servers, sending emails, and handling errors effectively.

5.3.5 Security

Parameterized SQL Queries:



```
# Use of parameterized query instead of an f-string
sql_query = "INSERT INTO quenchTanksTemp(QT2,QT3,QT4,QT5,date_time) VALUES
(?,?,?,?,GETDATE())"
params = (qT2Temp, qT3Temp, qT4Temp, qt5Temp)
cursor.execute(sql_query, params)
```

Fig. 5.10 Parameterized SQL Queries

The sql_query string contains placeholders (?) for the values to be inserted into the database. These placeholders are then replaced with the values in the params tuple when the cursor.execute(sql_query, params) line is executed.

This approach prevents **SQL Injection attacks** because the values to be inserted are sent separately from the SQL command and are not treated as part of the SQL command itself. This means that even if a value contains potentially harmful SQL code, it will not be executed as SQL code but will instead be treated as a simple string value.

CHAPTER 6: CONCLUSION

6.1 CONCLUSION:

In the first phase of the project, which focused on Data Acquisition from the PID Controllers, significant groundwork was laid for the subsequent phases. This initial phase involved establishing communication with the PID 500 controller and the MOXA device using the MODBUS communication protocol. MODBUS, a widely-used protocol in industrial automation, enables seamless communication between electronic devices in a master-slave architecture, supporting both serial and Ethernet connections. The data obtained from the PID controllers, organized into coils, discrete inputs, input registers, and holding registers, was crucial for monitoring parameters within the system. Each device was assigned a unique address, and error handling mechanisms were incorporated to ensure reliability and data integrity.

In the completed second phase of the project, significant advancements have been made towards enhancing the Graphical User Interface (GUI) application. The overhaul undertaken has not only refreshed the interface but has also integrated additional functionalities to increase user experience and efficiency. Rigorous testing procedures have been conducted post-overhaul to ensure the reliability and stability of the updated GUI. This testing phase has been instrumental in identifying and rectifying any bugs or issues that may have arisen during the development process, thereby bolstering the overall robustness of the application.

In the final phase of the project, the culmination of efforts leads to the deployment of the system onto the on-premises Level 2 server. This deployment represents the transition from development to operational use, where the monitoring software becomes an integral part of the industrial infrastructure.

REFERENCES

1. https://www.ltshf.com/our-facilities/india/hazira/

2. https://www.selec.com/product-details/temperture-controller-pid

3. https://www.moxa.com/en/products/industrial-edge-connectivity/serial-device-servers/general-device-servers/nport-5200a-series/nport-5250a

4. https://www.python.org/

5. https://docs.python.org/3/library/tkinter.html

6. https://www.microsoft.com/en-us/sql-server

7. https://code.visualstudio.com/

8. https://www.modbustools.com/modbus_poll.html

9. https://git-scm.com/

10. https://github.com/

11. https://github.com/pawaryash/Quenching-Tank-Project

12. https://www.vmware.com/products/workstation-player/workstation-player-evaluation.html

13. https://pymodbus.readthedocs.io/en/latest/

14. Matplotlib — Visualization with Python

15. https://www.geeksforgeeks.org/multithreading-in-operating-system/