People's Democratic Republic of Algeria Ministry of Higher Education and Scientific Research University M'Hamed Bougara -Boumerdes

Institute of Electrical and Electronic Engineering

GRADUATION PROJECT REPORT PRESENTED IN PARTIAL

FULFILMENT OF THE REQUIREMENTS OF THE DEGREE OF

"MASTERS"

IN COMPUTER ENGINEERING

Title:

Automation and remote management of Cathodic protection of pipelines system

---

Presented by:

GHOMRANI AMANI

Supervisor :

Dr. H. BELAIDI

Co-Supervisor :

A. FEGHOUL

JUNE 2023

Abstract

In the realm of corrosion prevention methods, two broad categories exist: electrical and non-electrical approaches. While non-electrical methods encompass coatings, sealers, and corrosion inhibiting admixtures, cathodic protection falls under the electrical approach umbrella. Its main goal is to shift the reinforcing steel into a protected state and, consequently, prevent corrosion.

This project aims to develop an automated and remotely managed cathodic protection system for pipelines. The proposed system utilizes a master code to communicate with smart voltmeters and collect voltage readings and establish a web-based dashboard for displaying the collected data. In addition, the system includes a control module that allows for the adjustment of the cathodic protection parameters remotely (probe distances, description). The proposed system is expected to improve the efficiency and effectiveness of cathodic protection systems, while reducing maintenance costs and increasing the lifespan of pipelines.

Acknowledgement

Praise God the supreme, for giving me the strength, patience, will, and motivation to carry out this project to its end.

I would like to express my sincere gratitude to all those who have contributed to the successful completion of this project. Firstly, I am deeply thankful to my supervisors the engineer **Amine FEGHOUL** & Doctor **Hadjira BELAIDI** for their guidance, support, and invaluable insights throughout the entire project. Their expertise and encouragement have been instrumental in shaping the direction of this project.

I would also like to extend my appreciation to the members of my team, **Rania HELLAS** and **Oussama BOUROUAIAH**, for their collaborative efforts and dedication. Their contributions in designing, implementing, and testing the smart voltmeter network have been invaluable.

Furthermore, I would like to acknowledge the contributions of **DEO ELECTRONIQUE** for their assistance and support in providing necessary resources and equipments for the sake of realizing this work.

Dedication

I would like to dedicate this project to my loving family, whose unwavering support and encouragement have been the driving force behind my success. To my parents and brothers, Wael and Wafik, who have made countless sacrifices and believed in me from the beginning, I am forever grateful. Your constant support, guidance, and belief in my abilities have shaped me into the person I am today, and I owe my achievements to you. To my second family, my aunt Sihem for being a mother and a friend through my years in university.

I also dedicate this project to my incredible friends & cousins, who have stood by my side through thick and thin. Your unwavering support, motivation, and camaraderie have been instrumental in keeping me focused and inspired throughout this journey. Your friendship has brought joy, laughter, and meaningful moments to my life, and I am grateful to have you as my friends.

Contents

List of Figures

Listings

List of Tables

Presentation of the partner

DEO ELECTRONIQUE (Development Electronique Ordinateur) is a well-established national organization founded in 1999 and located in CHERAGA - Algiers. The company specializes in various sectors, including industrial applications, centralized technical management, and remote operation technical management of buildings.

With a focus on electronic security, DEO ELECTRONIQUE provides solutions for access control, pedestrian and vehicle flow control, intelligent video surveillance, IP cameras, anti-intrusion systems, and biometrics. The company has recently expanded its expertise to serve the oil and gas industry, offering innovative project solutions to address challenges faced by the national petroleum company.

DEO ELECTRONIQUE takes pride in being the first integrator of LENEL's ONGUARD solutions by UTC Fire and Security in Algeria, further solidifying its position as a leading provider of cutting-edge security technologies in the region.

Figure 1: Deo Electronique logo

General introduction

The gas and oil industry has long grappled with the challenge of corrosion [1], which necessitates the continuous development of solutions. Among the various proposed methods, the cathodic protection system [2] has demonstrated remarkable effectiveness in mitigating corrosion-related issues. However, even minor errors in the system's diagnosis process can lead to catastrophic real-life incidents, such as pipeline explosions resulting from misdiagnosis. Such errors are often attributed to human factors. To prevent such mistakes, we propose an automated solution for monitoring the cathodic protection system.

This project report aims to automate the process of measuring, collecting, and displaying data related to the cathodic protection system. The automation involves three main tools: smart voltmeters or voltage probes, which are currently being designed and implemented by our co-working team, act as the system slaves responsible for voltage data collection upon command. A controller (NGC 1642) acts as the master to the probe slaves, executing commands and storing the collected data. Additionally, a web application is developed to showcase the data stored in the controller and facilitate control over the entire system's operation. This automated process significantly reduces human errors, minimizes data collection time, and streamlines data visualization.

The subsequent chapters of the report will provide a detailed explanation of the project. Chapter one offers an overview of the cathodic protection system, outlining its fundamental concepts and significance. The second chapter delves into our theoretical solution for automating the system, discussing the underlying principles and mechanisms. The third chapter focuses on the implementation process, providing insights into the practical aspects of the realization of our solution. Finally, the last chapter presents and visualizes some of the obtained results achieved while implementing the proposed automated solution.

By automating the measurement, collection, and display of cathodic protection system data, this project aims to enhance efficiency, accuracy, and safety while reducing human error. The subsequent chapters will delve into each aspect in further detail, providing a comprehensive understanding of the project's objectives, methods, and outcomes.

Chapter 1

Introduction to cathodic protection

This chapter aims to introduce the corrosion problem, present the chosen anticorrosion technique (Cathodic Protection), and outline the reasons for automating the anticorrosion solution.

1.1 Corrosion

Corrosion is a herbal system that happens whilst metals are uncovered to the environment. It involves the gradual deterioration of metal due to chemical or electrochemical reactions between the metal and its surroundings [1]. The process can result in a loss of material, weakening of the metal, and ultimately, failure of the metal structure. Corrosion can be caused by various factors, including moisture, oxygen, and other environmental factors, as well as exposure to chemicals or saltwater. It can occur in many different types of metals and can take on different forms, such as rust, pitting, or cracking. It can be prevented or controlled through various methods, such as the use of protective coatings, **cathodic protection**, or alloying the metal with other elements [2].

In this project, Cathodic Protection has been chosen as the adopted technique for combating corrosion. A comprehensive explanation of this anti-corrosion method will be presented in the subsequent sections for a deeper understanding.

1.2 Cathodic protection (CP)

1.2.1 Definition

Cathodic protection is a technique used to prevent corrosion of metal structures by providing an electrical current that counteracts the electrochemical reactions that cause corrosion. Despite its effectiveness, designing and implementing a cathodic protection system can be challenging due to its complexity. However, at its core, the principle behind cathodic protection is relatively simple and is based on electrochemical principles. Essentially, corrosion occurs when there is a potential difference between anodic and cathodic areas on a metallic structure. By supplying additional electrons to the structure through direct electric current, the rate of cathodic reaction is increased, reducing or even eliminating the rate of anodic reaction, which ultimately prevents corrosion [3].

The goal of cathodic protection is to eliminate or minimize the corrosion of metallic structures, which is achieved by applying a sufficient amount of direct current to shift the potential of the cathode to the potential of the anodic area [3]. When this potential difference is eliminated, corrosion will no longer occur.

1.2.2 Types of Cathodic protection systems

There are two primary types of cathodic protection systems: sacrificial anode cathodic protection (SACP or galvanic) and impressed current cathodic protection (ICCP).

- **Impressed Current Cathodic Protection (ICCP) systems:** work by introducing an external electrical current to neutralize the corrosion current that is present within the structure, thereby mitigating the corrosion activity on steel reinforcement [4] (Figure 1.1).

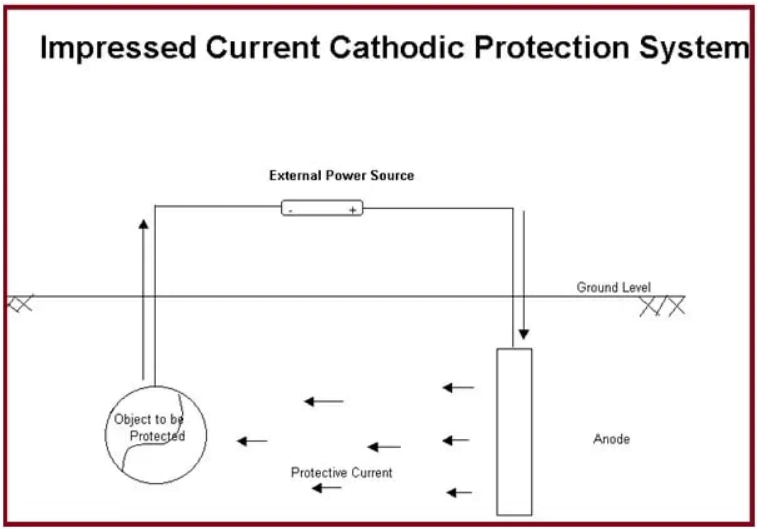


Figure 1.1: Impressed Current Cathodic Protection [5]

- **Sacrificial anode cathodic protection (SACP or galvanic):** involves attaching a sacrificial anode to the structure, which has a more negative electrochemical potential than the structure's metal. As a result, the difference in potential between the anode and the metal creates a positive current in the electrolyte, causing the metal to become more negatively charged and act as a cathode [4]. Hence, the sacrificial anode corrodes instead of the metal structure, providing long-term protection against corrosion (Figure 1.2).

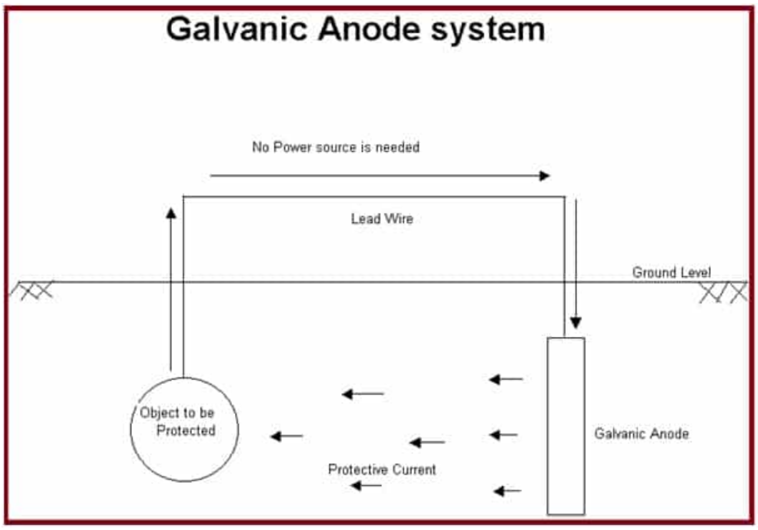


Figure 1.2: Sacrificial anode cathodic protection [5]

The selection of a suitable cathodic protection (CP) system is a complex decision that can be influenced by various factors, including the structural condition, available budget, and expected service life of the structure after the required repairs have been completed [4].

1.2.3 Cathodic Protection's Fields of use

Cathodic protection finds application in a diverse range of domains, showcasing its versatility as an effective corrosion control method. The following domains exemplify the varied areas where cathodic protection is employed:

- Marine Structures: offshore oil platforms, ships and docks.
- Water and Sewage Treatment Facilities: tanks, pipelines and pumps.
- Transportation Infrastructure: tunnels, encompassing bridges and highways.
- Power Generation Facilities: condenser tubes, cooling towers and boilers
- Oil and Gas Industry: safeguarding pipelines responsible for the transportation of crude oil, natural gas, and refined petroleum products across vast distances.

The focus of the system developed in this project is the latter industry mentioned, which pertains to the transportation of crude oil, natural gas, and refined petroleum products over long distances through underground pipelines.

1.2.4 Monitoring of Cathodic Protection System

The measurement of structure potentials involves assessing the electrical characteristics of materials to determine their ability to conduct electricity. This is crucial for evaluating their electrochemical behavior and assessing their compatibility in various environments.
The measurement process typically involves the use of a voltmeter connected to the material of interest and a reference electrode placed in its immediate surroundings. The voltmeter, selected for its high impedance, accurately measures the potential difference between the material and the reference electrode, providing valuable insights into the material's electrochemical behavior(Figure 1.3). Different types of reference electrodes are available:

1. Copper sulfate reference electrode (Cu/CuSO4) (CSE)
2. Silver chloride reference electrode (SSC), commonly used in seawater
3. Saturated calomel reference electrode (SCE)
4. Hydrogen reference electrode (SHE)
5. Zinc electrode (Zn)

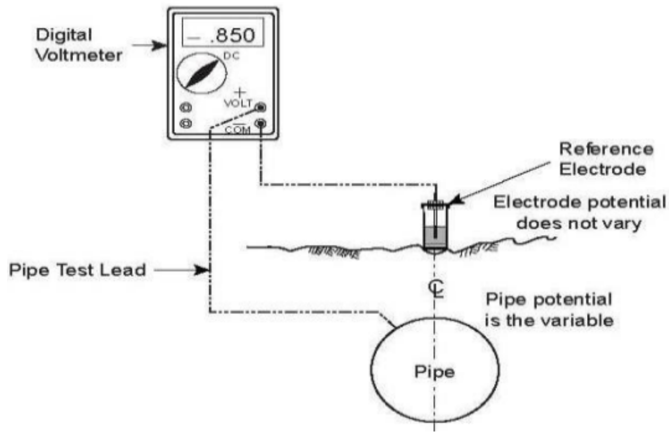


Figure 1.3: Measuring CP Voltages

Materials exhibit characteristic potentials that reflect their inherent properties and behavior which are:

- **Natural potential:** This refers to the material's potential during its manufacturing process in a controlled laboratory environment.
- **Abandonment potential:** It represents the potential measured immediately after the material is installed in a specific environment (soil or water).
- **Polarization potential:** This is the potential of the material measured after the application of cathodic protection, which aims to induce cathodic polarization and enhance the material's resistance to corrosion.

During potential measurements, the ohmic drop caused by the protective current flowing through the soil or electrolyte is taken into account. The ohmic drop is the result of the current and soil resistance. To obtain accurate measurements, the current is reduced to zero, and the material's potential is measured before depolarization occurs due to the discontinuation of cathodic protection. The ON/OFF mode enables cyclic activation and deactivation of cathodic protection, facilitating effective monitoring of the material's polarization behavior as illustrated in Figure 1.4.

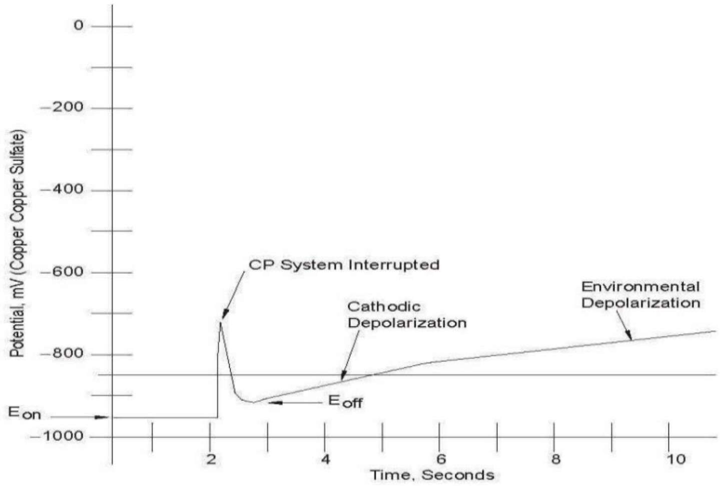


Figure 1.4: Monitoring polarization behavior

1.2.5 Methods and Techniques for Measuring Structure Potentials

The two common methods for monitoring CP systems are **Close interval potential survey (CIS/CIPS)** and **Direct Current Voltage Gradient (DCVG)**.

Close interval potential survey (CIS/CIPS)

The CIS/CIPS method involves measuring the pipe-to-soil potential at close intervals along the length of the pipeline to identify any areas of inadequate cathodic protection.

CIS/CIPS surveys use portable instruments called reference electrodes, which are placed in contact with the soil at known intervals along the pipeline (Figure1.5), such as 1 m (3-ft), 1.5 m (5-ft), and 3 m (10-ft) [6]. A current is then applied to the pipeline, and the resulting potential difference is measured at each reference electrode. Several types of reference electrodes are available for use,the most commonly used ones being the Copper Sulphate electrode **(CuSo4)** (Figure 1.6) and Zinc electrode **(Zn)**.

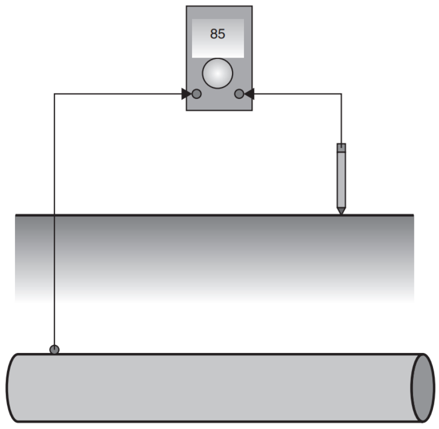


Figure 1.5: CIS/CIPS configuration [6]

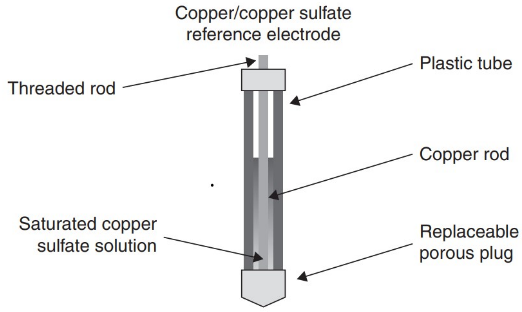


Figure 1.6: Reference electrode [6]

When CIS is conducted over a structure with CP, it aims to obtain a representative series of potentials throughout the structure to evaluate the adequacy of the CP, identify any local deficiencies, and determine the cause(s) of such deficiencies. Additionally, it assists in determining remedial actions to enhance the level of CP. On the other hand, when CIS is conducted over a structure without CP, its primary objective is to identify areas that may be corroding, commonly known as hot-spots, and to identify any possible interference from other structures with CP [6].
pipe-to-soil potential measurements are taken for many different purposes, such as testing for the following:

- confirming compliance with adequate CP criteria.
- electrical isolation of the structure.
- electrical continuity.
- shielding.
- stray current.
- evaluation of coating condition.

These measurements provide an indication of the level of cathodic protection on the pipeline at each location, and any areas with inadequate protection can be identified and targeted for repair.

Direct Current Voltage Gradient (DCVG)

Direct Current Voltage Gradient (DCVG) is a non-destructive testing method used to assess the integrity of buried pipelines. DGVC measures the voltage gradient on the surface of the structure being protected. This method involves placing two electrodes on the surface of the structure, applying a DC current between them, and measuring the voltage gradient along the structure's length, as illustrated in Figure1.7. Areas of inadequate CP protection exhibit a higher voltage gradient than well-protected areas.

Figure 1.7: Direct Current Voltage Gradient (DCVG) [7]

1.2.6 Limitations

While Close Interval Potential Survey (CIS/CIPS) and Direct Current Voltage Gradient (DCVG) monitoring techniques are useful in cathodic protection, they can have some limitations, namely manual data collection, delayed detection, limited coverage, and data interpretation...

- **Manual Data Collection:** data collection at specific intervals along the pipeline can be time-consuming, labor-intensive, and prone to human errors.
- **Delayed Detection of Problems:** Due to the interval-based nature of the surveys, the detection of cathodic protection system issues or failures may be delayed. Any changes in the system's performance between the intervals may go unnoticed until the next survey.

- **Limited Coverage:** CIS/CIPS and DCVG provide localized measurements along the pipeline, focusing on specific points or intervals. This limited coverage may result in missing critical areas or localized defects that are not captured by the survey.
- **Data Interpretation Challenges:** data interpretation requires expertise and experience to analyze and identify potential issues accurately. Interpreting complex data patterns and distinguishing between normal variations and abnormal conditions can be challenging.

An effective approach to address the limitations associated with manual data collection, delayed problem detection, and limited coverage in cathodic protection monitoring is the automation of the monitoring system.

1.3 Automation in Cathodic Protection

1.3.1 Reasons for Automating Cathodic Protection Systems

Cathodic protection is a vital technique used to prevent corrosion in metallic structures, such as pipelines, tanks, and offshore platforms. Traditionally, cathodic protection systems have been manually operated, requiring periodic monitoring and adjustments by skilled personnel. However, the integration of automation in cathodic protection offers several compelling rationales:

- **Enhanced Efficiency:** Automation streamlines the monitoring and control of cathodic protection systems, reducing the need for constant human intervention. This leads to improved operational efficiency and productivity, allowing personnel to focus on other critical tasks.
- **Real-time Monitoring:** Automated systems enable continuous and real-time monitoring of key parameters, such as voltage, current, and potential, providing immediate feedback on the system's performance. This allows for proactive identification of issues and prompt corrective actions, minimizing the risk of corrosion-related failures.
- **Data-driven Decision Making:** Automation facilitates data collection and analysis, enabling informed decision making regarding system adjustments, maintenance schedules, and performance optimization. By leveraging data insights, operators can make more accurate and strategic decisions, leading to improved long-term corrosion control.

1.3.2 Advantages and Challenges of Automation

Automating Cathodic Protection (CP) Systems offers numerous advantages in terms of efficiency, accuracy, and overall effectiveness. Automation brings forth several benefits:

- **Accuracy and Precision:** Automation eliminates human errors associated with manual data recording and adjustment, ensuring accurate and precise measurements and control settings. This enhances the overall effectiveness of the cathodic protection system.
- **Increased Safety:** Automation reduces the need for personnel to physically access hazardous or remote locations for system monitoring and adjustment. This minimizes the risk of accidents, injuries, and exposure to dangerous environments.
- **Cost Optimization:** Automated systems offer potential cost savings by optimizing power consumption, reducing manual labor requirements, and prolonging the service life of assets through more effective corrosion control.

While automating Cathodic Protection (CP) Systems offers significant advantages, it is not without its challenge. Implementing automation in CP systems requires careful consideration of various factors:

- **Initial Investment:** Implementing an automated cathodic protection system involves upfront costs for hardware, software, and system integration. However, the long-term benefits often outweigh the initial investment, particularly for large-scale or critical infrastructure projects.
- **Technical Expertise:** Automation necessitates skilled personnel who are proficient in programming, network infrastructure, and data analysis. Adequate training and ongoing support are crucial to ensure efficient operation and maintenance of the automated system.
- **System Compatibility and Integration:** Integrating automation into existing cathodic protection infrastructure may pose challenges related to compatibility with legacy systems, data synchronization, and connectivity. Seamless integration and interoperability should be considered during the planning and implementation stages.

1.4 Web Application

1.4.1 Definition

A web application is a software application that is accessed through a web browser or web-enabled device and is typically built using web technologies such as HTML, CSS, and JavaScript. It is designed to provide interactive functionality and deliver services or information over the internet.

Web applications run on web servers and utilize **client-server** architecture, where the client (web browser) sends requests to the server, and the server processes the requests, retrieves data from databases or other sources, performs computations, and generates dynamic web pages or responses that are sent back to the client. They are highly accessible as they can be accessed from various devices with an internet connection, including desktop computers, laptops, tablets, and smartphones. They provide a convenient way for users to interact with services and information without the need for installing dedicated software on their devices.

Figure 1.8: Web Application / Dashboard

1.4.2 Why Web Application?

The web application is specifically designed for engineers responsible for monitoring the cathodic protection system. The web app serves as a user interface where the collected voltage data is displayed in an organized and accessible manner.

Engineers can access the web app to view real-time or historical data from the voltage probes. The displayed data provides valuable insights into the system's voltage measurements, allowing engineers to monitor the overall state and performance of the cathodic protection system.

By analyzing the data presented in the web app, specialists can further examine and evaluate the system's condition. They can identify any potential issues, patterns, or anomalies in the voltage measurements, enabling them to make informed decisions and take appropriate actions to maintain the cathodic protection system's integrity.

1.5 Conclusion

This chapter provided an overview of the cathodic protection system and introduced our proposed automated solution. It highlighted the significance of accurate monitoring and the need to minimize human errors. The subsequent chapters of this report will delve into the detailed explanation of the design, implementation, and evaluation of this automated cathodic protection monitoring system.

Chapter 2

Theoretical Framework and Design of an Automated CP System

In this chapter, we will delve into our proposed solution for automating the cathodic protection system, including a detailed exploration of the tools and techniques employed to achieve this goal. Comprehensive insights into our approach will be provided, highlighting the various components and methodologies utilized in the automation process.

2.1 Hardware

2.1.1 NGC 1642 CONTROLLER

The DEO ELECTRONIQUE NGC 1642 controller serves as the microcontroller for this project, representing their latest advancement in next-generation controllers. Its selection as the core component aligns seamlessly with the proposed solution for automating the Cathodic Protection (CP) system. Notably, it encompasses several key features that enhance system functionality and performance.
One significant feature of the NCG 1642 controller is its integrated Ethernet port, supporting both TCP/IP mode and RS-485 communication protocol, which we have chosen for our specific project requirements. This ensures reliable and efficient data transmission between the controller and other system components. More technical specifications of NGC 1642 are given in Table 2.1

Table 2.1: Technical specifications of NGC 1642

Dimensions:	210mm x 140 x 25mm
Power supply:	24V dc 500 mA power supply with a 12V DC output
Communication:	Ethernet (TCP/IP) RS-485
Number of inputs/outputs:	16 discrete / pulse inputs 4 analog inputs 4-20mA 16 control relays 2 analogue outputs 0-10V
Memory:	2Mo
Extension:	Expansion capability up to 16 interfaces
Battery:	3V lithium
Frequency:	Frequency of 200Mhz
Settings:	Integrated Web Interface with Access Authentication for Testing, Configuration, and Verification of Input/Output States
Assembly:	6 plastic supports and screws for mounting on a flat surface.
Environment:	-20°C to 60°C (-4°F to 140°F)
LED indicators:	32 LEDs
USB input:	1 input
Restart:	1 restart button

To ensure data integrity and continuity, the controller incorporates a non-volatile memory capable of storing up to 10,000 events. This feature safeguards critical information even during unexpected server connection interruptions, enabling seamless operation and reliable data retention.

Moreover, the NGC 1642 controller, shown in Figure 2.1, is equipped with PID interlocking capabilities, enabling effective regulation and interlocking functions within the CP system. It also efficiently manages scheduled tasks received from the management software, facilitating seamless integration with the overall control infrastructure.

Figure 2.1: NGC 1642 controller

In terms of hardware specifications, the NGC 1642 controller boasts a 42 MHz microcontroller and a 2Mo memory. Its design features six mounting holes suitable for DIN rail installation, as well as plastic supports and screws for surface mounting. This design facilitates easy adaptability in various environments, including enclosed and dry spaces, industrial settings, warehouses, offices, commercial premises, and wastewater treatment plants.

The DEO controller (NGC 1642), based on a PIC microcontroller, utilizes a FreeRTOS-like (Real Time Operating System) logic to handle tasks with different priorities and separate heap memories. These tasks are managed by a scheduler. Upon the initial power-up of the MCU, comprehensive testing and initialization procedures are performed to detect any hardware issues.

FreeRTOS-like logic implies that the MCU employs a similar task scheduling mechanism as provided by FreeRTOS. While the actual implementation may differ, the use of prioritized tasks and separate heap memories aligns with the core principles of FreeRTOS.

2.1.2 Smart voltmeters (Probes)

Our collaborative team has put forward a detailed plan to implement a network of intelligent voltage probes that will be positioned at regular 1-meter intervals s illustrated in **Figure2.2**. The primary objective of this implementation is to streamline the collection of voltage data from the pipeline surface. These intelligent probes will be equipped with the capability to acquire and aggregate voltage measurements. The collected data will be temporarily stored in the integrated memory of the PIC microcontroller, which has a capacity of **64kb**.

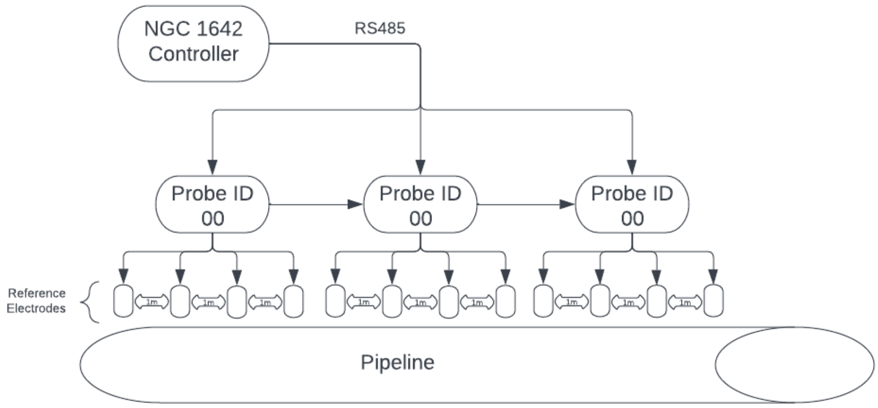


Figure 2.2: Controller-Probes connection

At a later stage, the collected voltage data will be transmitted to the controller for further analysis and processing. To facilitate seamless communication between the probes and the controller, the probes will utilize the **RS485 communication protocol** as their integrated communication protocol. This protocol ensures efficient data transfer and enables effective interaction between the probes and the controller.

Probe ID

Each probe in the network is equipped with a set of switches, illustrated in Figure 2.3,that can be manually configured to assign a unique probe ID. The set of switches consists of five switches, allowing for a binary representation of the ID. As there are five switches, the maximum number of unique probe IDs that can be generated is 32, which corresponds to 2^5. This configuration ensures that each probe in the network has a distinct and identifiable ID for efficient identification.

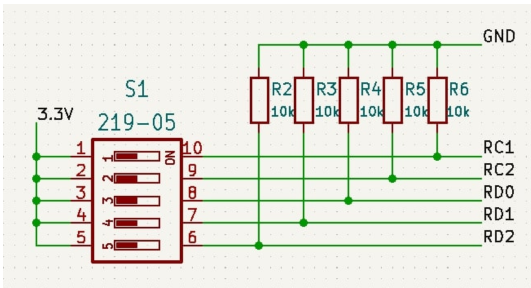


Figure 2.3: Probe ID switches

ADC reader

The voltage probes within our network are designed to read four distinct voltages from the reference electrode. Each probe is equipped with four channels dedicated to this purpose. These channels are connected to an **Analog-to-Digital Converter(ADC)** as shown in Figure 2.4, which converts the analog voltage readings into hexadecimal digital values. Later on the hex data is saved into the PIC memory to be transferred to the controller on a command using the RS485 communication protocol.

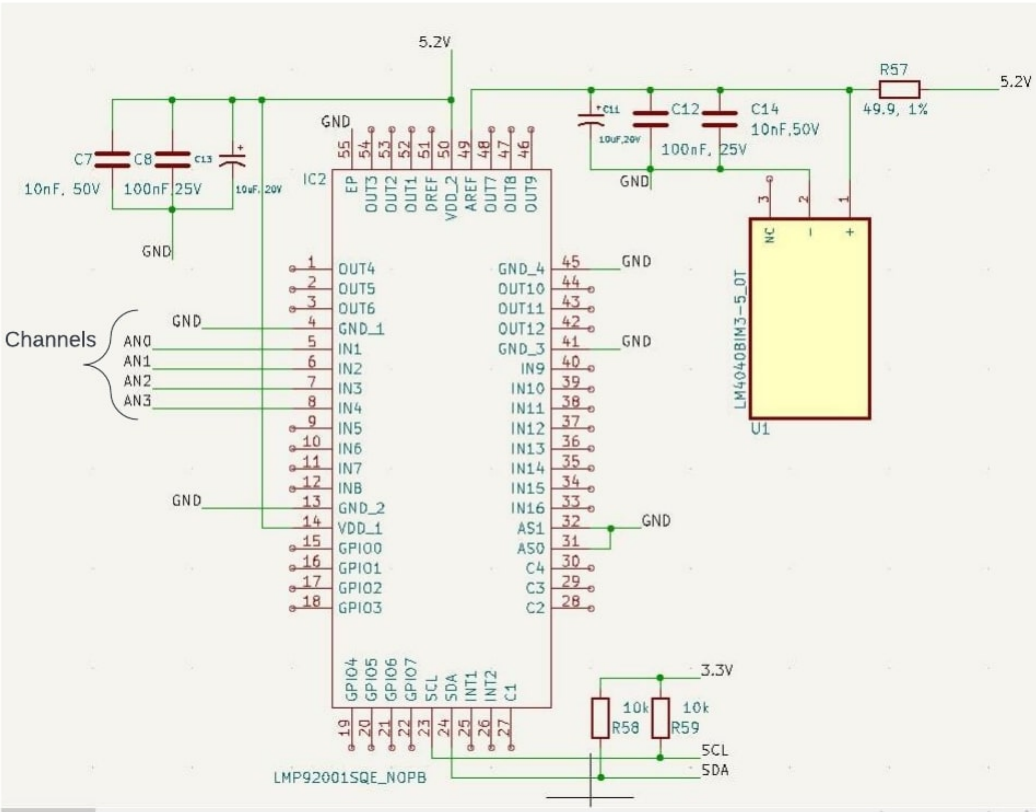


Figure 2.4: Probe ADC & Channels

2.1.3 Web Application

A web application serves as our software solution to visualize transmitted data from the controller, the communication is established through the use of sockets. For a successful connection some conditions must be applied:

- the controller and the web server are both connected to the **same network** as illustrated in Figure 2.5.
- The socket at the server's end is created with the Controller's specific IP address & its corresponding port.

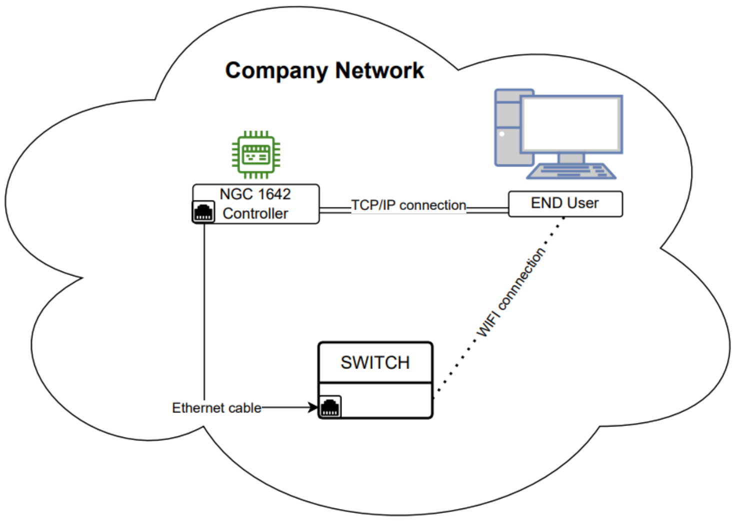


Figure 2.5: Controller-User connection

The **CP-Supervision** web application is tailored for engineers and focuses on presenting data in tables and charts to facilitate system diagnosis. The dashboard provides the engineers with a clear and organized view of the collected data, allowing them to analyze and interpret the information effectively. In addition to data display, the application allows to interact with the system by providing functionality to input information about the probes. This feature enables engineers to fill the database **(PostgreSQL)** with relevant details about the probes (Their distances, site name (zone) and description). The combination of tables, charts, and interactive features allows for efficient system diagnosis, while the ability to input probe information enhances the overall functionality and usability of the application.

2.2 Data transfer mechanism

Two protocols are used for data transmission mechanism; the RS485 communication protocol for **MCU-to-probe** and **probe-to-probe** communication and TCP protocol for **server-client** communication. These two protocols are described in the following sections.

2.2.1 RS-485 communication Protocol

RS-485 is an industrial specification designed to define the electrical interface and physical layer for reliable point-to-point communication between electrical devices. It offers several key advantages, making it well-suited for various applications, particularly in electrically noisy environments.

One of its notable features is the capability to support long cabling distances, enabling communication over extended distances without significant signal degradation. Furthermore, RS-485 is a versatile standard that supports multiple devices connected to the same bus, allowing for efficient and cost-effective networking solutions. This makes it an ideal choice for scenarios where reliable and robust communication is essential.

2.2.2 Features of RS-485

The RS485 communication protocol is employed for MCU-to-probe and probe-to-probe communication because it offers several features that make it suitable for long-distance, multi-node communication, some of them are:

1. **Differential Signaling**: which means the data transmission will be the difference in voltage between two copper wires.(Figure2.6)

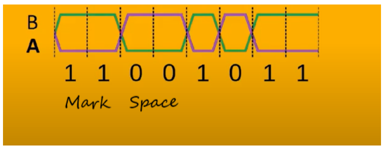


Figure 2.6: RS-485 Differential signaling [8]

2. **Multi-Point Communication**: it means allowing multiple devices to be connected on a single bus. Each device has a unique address, and they can transmit and receive data independently (Figure 2.7).

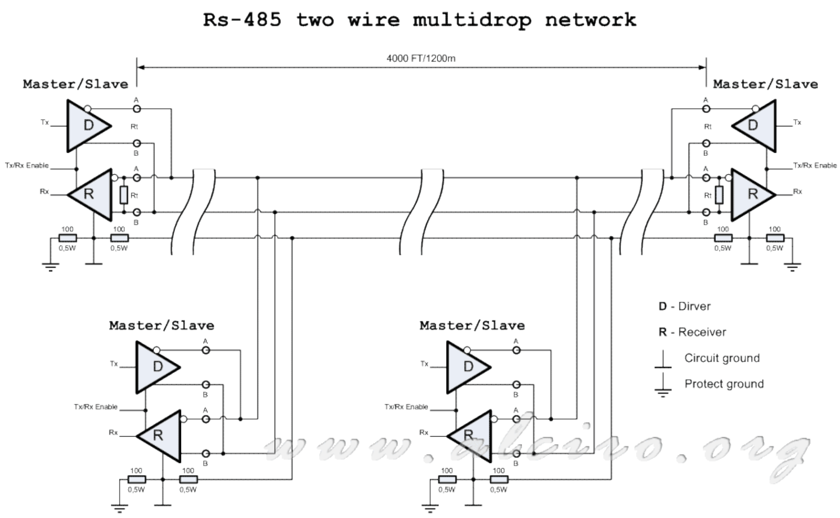


Figure 2.7: RS-485 Multi-Point Communication [9]

3. **Full-Duplex/Half-Duplex Operation**: RS-485 can operate in both full-duplex and half-duplex modes. In full-duplex mode, simultaneous bidirectional communication is possible, while in half-duplex mode, communication alternates between transmission and reception.

4. **High Data Rates**: RS-485 supports relatively high data rates, typically up to 10 Mbps, making it suitable for applications requiring fast communication.

5. **Long Cable Lengths**: RS-485 can support cable lengths of up to 1200 meters even longer, depending on factors such as data rate, cable quality, and noise levels.

2.2.3 TCP sockets communication

In networking, a socket is an endpoint for sending and receiving data across a computer network. It represents the combination of an IP address and a port number as illustrated in Figure 2.8.

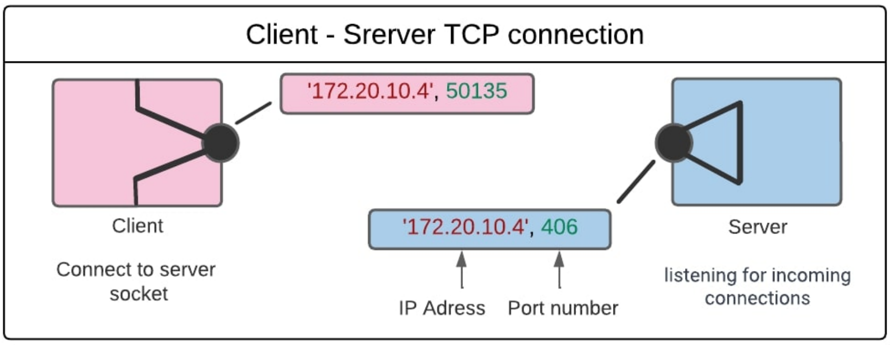


Figure 2.8: Socket Client-Server

In the context of TCP sockets communication, each endpoint (client and server) has its own socket (Figure 2.9 summarizes the client-server connection). The server socket listens for incoming connections, and upon accepting a connection, a new socket (client socket) is created for communication with that specific server. TCP/IP sockets are chosen for communication because they provide a reliable, stream-oriented, bidirectional communication channel between a client and a server.

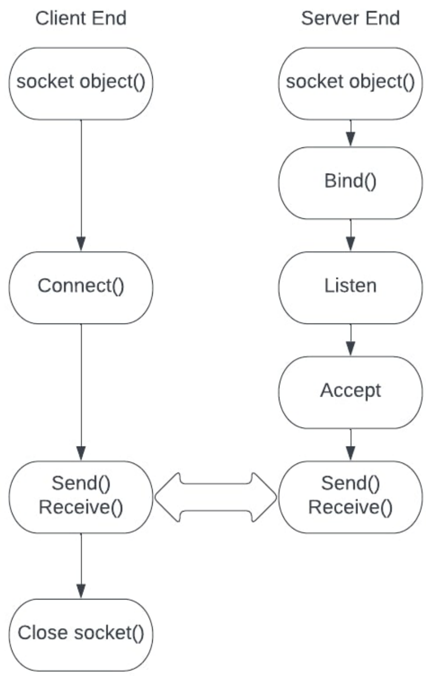


Figure 2.9: Socket Client-Server Connection Diagram

2.2.4 Controller-Probe Communication

Master code

The connection between the controller and the probes in the network follows a master-slave relationship. In this setup, the master code, developed specifically for this purpose, is responsible for managing and controlling the communication between the controller and the probes.

The master code is designed to send instructions and commands to the probes, directing them to operate in different states as required. These states include the following actions:

- **State_ON:** Instructs the probes to perform voltage measurements when the Cathodic protection is turned ON. This is achieved by sending a constructed frame that corresponds to this state.
- **State_OFF:** Instructs the probes to perform voltage measurements after turning the Cathodic protection OFF. This is done by sending a constructed frame specific to this state.
- **State_ON/OFF:** In this state, the Cathodic protection is briefly turned ON and then turned OFF. All the probes are simultaneously instructed to take readings. This is

achieved by sending a constructed frame that triggers the probes to read simultaneously.

- **Probe Test State:** The master code includes a TEST state that allows for evaluating and verifying the functionality of the probes. In this state, the master code can assess their connection and ensure proper operation.

Frame construction

In every state (mode), the frames are constructed based on the selected operating state, the following diagram in Figure2.10 explains well the behaviour of the **frame-construct** function:

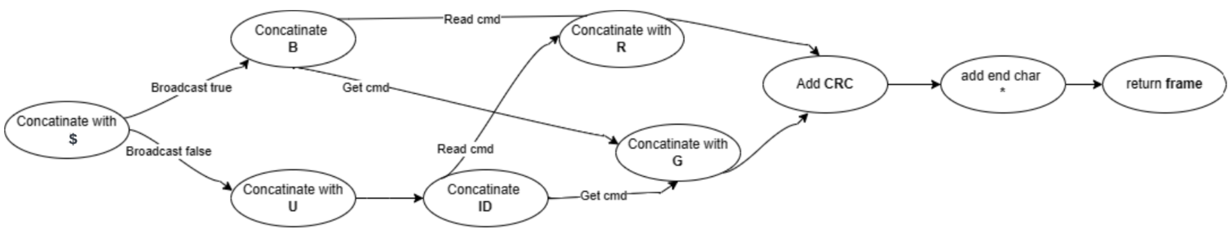


Figure 2.10: Frame preparation diagram

The frame begins with the starting character **'$'**, and the subsequent characters are concatenated based on the selected mode. The specific meaning and significance of each character will be explained in detail in the upcoming chapter (section 3.2.1).

CRC calculator

To ensure accurate data transmission and reception, a CRC (Cyclic Redundancy Check) error check mechanism is employed. CRC is an error-checking technique commonly used in digital communication to detect errors in data transmission. It is a mathematical algorithm that generates a fixed-size checksum value based on the data being transmitted. This checksum value is appended to the data and sent along with it [10].
At the sender's end, the CRC algorithm calculates the checksum value by performing a **bit-wise XOR** operation on the data bits. The result is a unique value that represents the data content. This checksum is then added to the transmitted data. Upon receiving the data by the probe, the receiver performs the same CRC calculation on the received data. If the computed checksum matches the received checksum, it indicates that the data has been

received without errors. If the checksums do not match, it implies that an error has occurred during transmission.

The flowchart given in Figure 2.11 describes the structure of the master code:

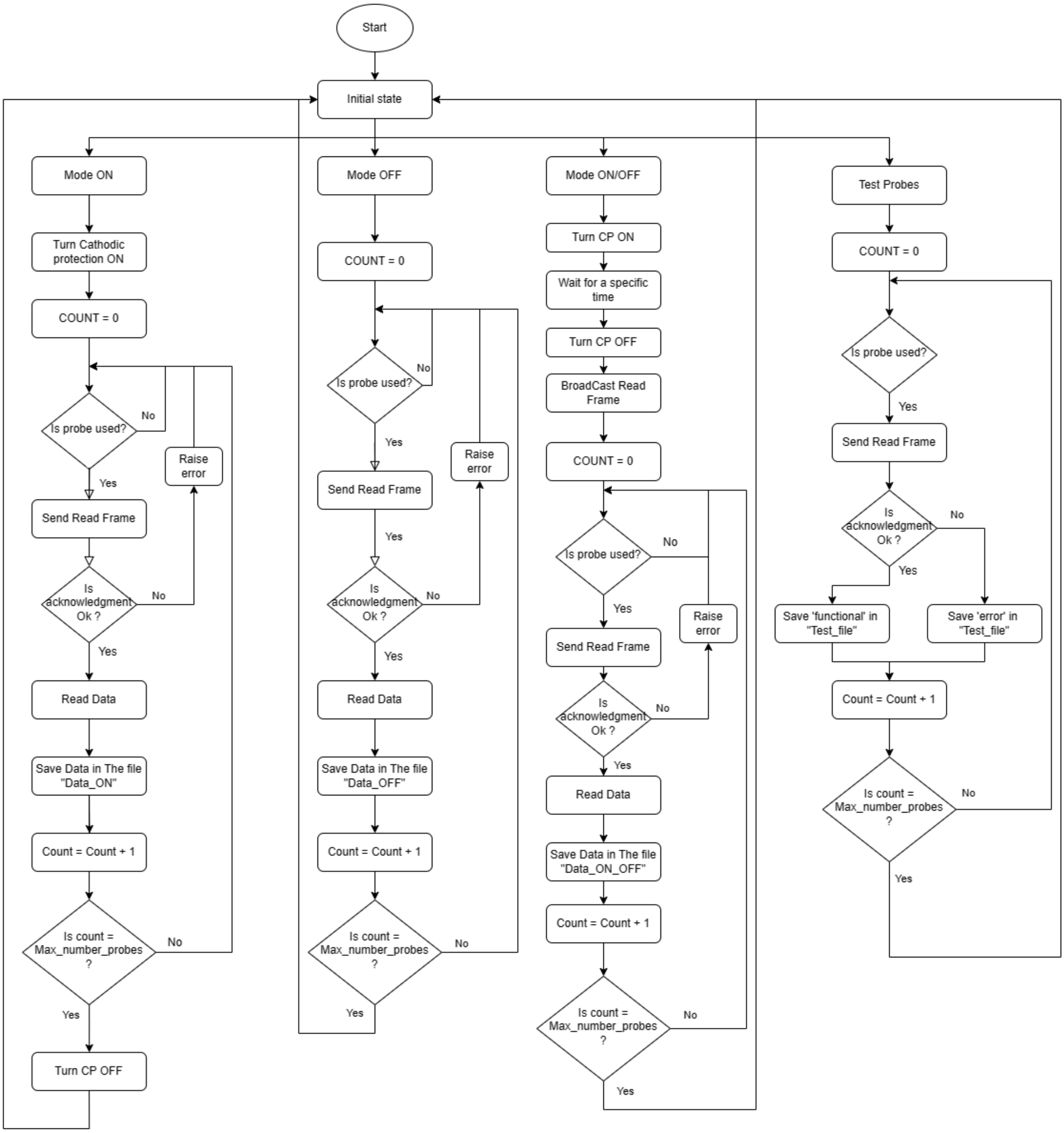


Figure 2.11: Master code Flowchart

2.2.5 Web-Controller Communication

The web application structure can be divided into three main parts: system configuration, data collection, and data display.

- **System Configuration:** This section focuses on configuring the system by adding probes to the database along with their specifications such as locations and descriptions. It provides a means to manage and organize the probes within the application, allowing users to define their properties and attributes.
- **Data Collection:** In this part, the application facilitates the collection of voltage readings from the probes. the server initiates the process by establishing a TCP/IP socket connection with the controller. Through this connection, the server sends instructions to the controller to trigger the probes for voltage measurements. The obtained data is then saved into the database.
- **Data Display:** The final part involves retrieving the stored data from the database and presenting it in a user-friendly manner. The application provides various visualization options, such as tables or charts, to display the voltage readings effectively. Users can access and explore the collected data, enabling them to gain insights, analyze trends, and make informed decisions based on the displayed information.

The following flowchart describes the system well (see Figure 2.12).

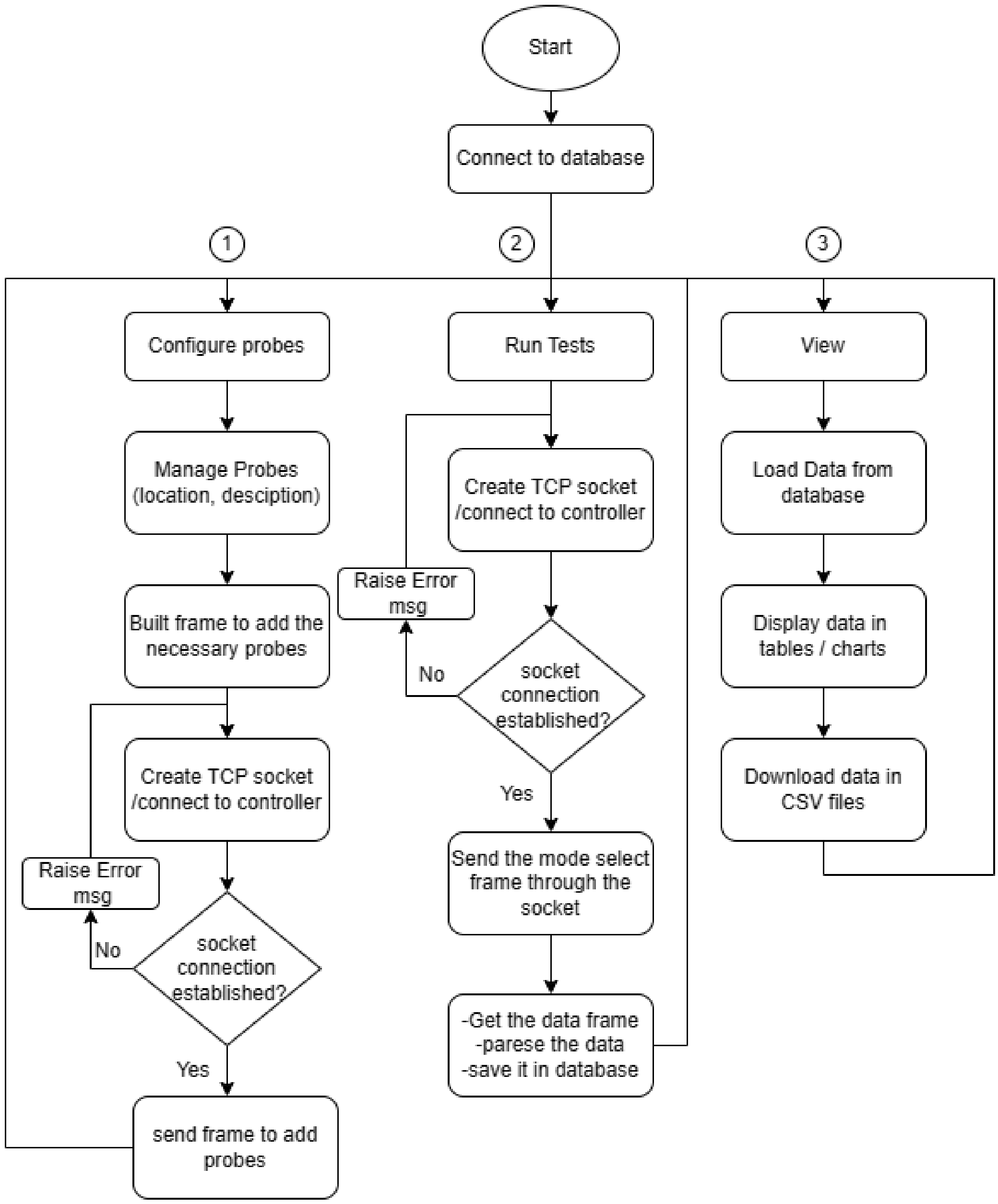


Figure 2.12: Web application structure

2.3 Conclusion

This chapter explored the solution structure of the automated CP system, discussing the master code framework and the web application's mechanism for data management and display. It provided insights into how the system is organized and how data is handled within the framework.

Chapter 3

Software Implementation and Toolset of the Automated CP System

The communication protocol acts as the foundation for smooth data exchange between different endpoints within the automation solution. These endpoints include the DEO ELECTRONIQUE intelligent controller NGC 1642, the smart voltmeters functioning as system slaves, and the Web App. The chapter delves into the detailed implementation of the automation solution discussed earlier, focusing on how the communication protocol is utilized to facilitate efficient and reliable data transfer.

3.1 Master Code

The Master Code takes center stage as it forms the heart of system control and monitoring. Developed in the C language specifically for the NGC 1642 controller, the Master Code empowers the controller with the capability to orchestrate and coordinate the activities of the CP system. The chapter delves into the functionalities of the Master Code, emphasizing its role in retrieving data from the voltmeters, and displaying that data is user-friendly interface.

3.1.1 Programming Language

The C language has been chosen as a programming language for various reasons, here are some key reasons why C:

- **Efficiency:** C is renowned for its efficiency in terms of memory usage and execution speed. It allows for fine-grained control over hardware resources, making it suitable for resource-constrained systems like microcontrollers.

- **Low-level programming:** C provides direct access to memory and hardware resources, allowing for fine-grained control and enabling tasks like device drivers and firmware development.
- **Language stability:** C has a stable and mature language specification, ensuring reliability and compatibility.
- **Community and support:** C has a large and active community of developers, offering access to valuable resources, forums, and documentation for support and collaboration.

3.1.2 Code Structure

The DEO MCU effectively manages tasks, as previously stated, where the master code of the cathodic protection system is implemented as a task. The execution of this task is initiated by invoking the **probe_init** function, which takes charge of configuring the communication setup required for seamless interaction with the smart voltmeters. This configuration defines the serial bus used for RS485 communication and sets the system timeout for the frame exchange process. If the timeout is reached, the connection is automatically terminated.
In addition to the communication setup, the **probe_init** function also initializes the necessary files to store the collected data from the probes. These files serve as system files (referred to as fs) for storing the collected data from the probes. These system files provide a structured and organized storage mechanism for the data generated during the operation of the cathodic protection system.

```
/*init and task*/
void prob_init(void)
{
  /*
   initializing files and other variables
  */
  state = STATE_INIT;

  on_file_index = fs_register("on file");
  off_file_index = fs_register("off file");
  static_file_index = fs_register("static file");
  test_file_index = fs_register("probe test file");

  configSB1.sb = 1;
  configSB1.txtimeout = 1000;
  configSB1.rxtimeout = 1000;
}
```

Figure 3.1: Probe_init Function

As mentioned before, the readings of the cathodic protection system should occur in three different states: the state where the CP is ON, the reading when CP is off (the natural state of the pipeline), and when the cathodic protection is ON/OFF. The reading in the ON/OFF state should occur after turning the CP off; however, these readings should occur within the specific interval of [150ms - 500ms] which is handled at the probe's end.

```
void prob_task(void)
{
  switch (state)
  {

  case STATE_INIT: // based on the tcp/ip frame the state changes
  {⋯

  case STATE_OFF: // the OFF state
  {⋯

  case STATE_ON: // the ON state
  {⋯

  case STATE_ON_OFF: // the ON OFF state
  {⋯

  case STATE_TEST_PROBE: // the TEST state
  {⋯
  default:

    break;
  }
```

Figure 3.2: Switch-Case Logic

The code will remain in the initial state until one of the modes is triggered through a TCP/IP frame, which will be sent via the web app . The frame will redirect the code to a different state by changing the state through the function **"probe_change_state"** in the frame_processor.c file.

The collected data is extracted probe by probe, where each probe provides four voltage reading, these voltage readings are concatenated into a frame, where each voltage is represented using a **16-bit** unit. The received voltages are parsed and stored in **fs** (file system) files, which are stored in the memory of the microcontroller.

Before taking voltage readings from any mode, it is important to ensure that the probes are functioning properly and are able to communicate. To address this, a separate state has been added to the system **STATE_TEST_PROBE**. This state is responsible for testing the communication of the probes. After the communication test is performed, the system saves the state of each probe in a file and later displays it in the web application.

3.2 Communication Mechanism Between Probes and MCU

As previously mentioned, the communication between the MCU and the probes is facilitated by the RS485 communication protocol. It is important to note that the RS485 protocol does not inherently include predefined mechanisms for automatic message acceptance or rejection. Consequently, the decision-making process regarding the received messages is typically implemented at the software level within the firmware of each individual node.

When transmitting data through RS485 to a specific node, all nodes within the RS485 network receive the transmitted message. However, it is the responsibility of each node to independently assess the received message and determine whether it should be accepted or disregarded, based on specific criteria, namely the **node ID** and the **Broadcast flag**. By considering the **node ID**, the appropriate nodes are identified for targeted communication, while the **Broadcast flag** allows for communication with all nodes in the network. The comprehensive discussion in upcoming sections will shed further light on these selection criteria and their significance in the communication process.

3.2.1 Frame Structure

The frame incorporates a header section comprising essential data such as the **probe ID**, the **desired action**, and an error check mechanism (**CRC**). Additionally, the frame is terminated with an end character denoted by **' * '**.

Frame Header

The header, encompassing the probe ID and action, plays a crucial role in identifying the intended recipient and the specific operation to be performed. Every frame consistently begins with the header string **' $ '** where the character **' $ '** denotes the frame start. Subsequently, the character representing the state of the Broadcast flag is appended to the header. In the case of a set Broadcast flag, the character **'B'** is concatenated, whereas the character **'U'** is concatenated for an unset flag. Following this, the probe ID and command character are assigned based on the flag selection. Within this communication framework, three distinct types of commands are dispatched to the probes:

1. **'R':** Signifying that the probe should read the voltages and store them in its memory. This command can be either broadcasted to all probes, which is predominantly the case,

or individually addressed to each probe for the purpose of testing their functionality.

2. **'G':** Indicating that the probe should retrieve the stored data and transmit it back to the microcontroller. This command exclusively applies to individual (unicast) transmissions.

3. **'T':** This command, exclusively sent via unicast, serves to verify whether the broadcasted frame has been received by the probe or not.

Error Check Mechanism (CRC)

By employing CRC, errors caused by noise, interference, or transmission issues can be detected, enabling effective error detection and error correction mechanisms in digital communication systems. The CRC used for error checking is an 8-bit string that will be concatenated at the end of the frame, it is generated by applying a bitwise **XOR** operation on the constructed frame. The process begins by converting the first character of the frame into its 8-bit binary representation and storing it in a variable. As the loop iterates through the remaining characters of the string (frame), the XOR operation is performed on the variable, updating its value. Figure 3.3 illustrates this process through a code function. After this process the end character ' * ' is then added.

```c
/*Fuction that finds the checksum of a given string*/
void ChecksumCalc(char* dataString){
    uint8_t xorTemp;
    xorTemp = (uint8_t)dataString[0];
    for(int i = 1; i < strlen(dataString); i++){
        xorTemp ^= (uint8_t)dataString[i];
        }
    sprintf(CRCresult,"%.2x",xorTemp);
}
```

Figure 3.3: XOR CRC Chechsum Calculator

Frame End Character

The concluding ' * ' character signifies the completion of the frame. An example of a sent frame would be **" $U01R06* "**, further explanation of this frame is illustrated in Figure 3.4.

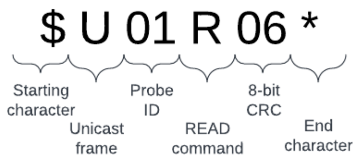


Figure 3.4: Unicast read command frame (controller-to-probe

3.2.2 Prepare the frame

The previously explained frames are constructed using a function called **"Prob_prepare_frame"** which accepts parameters such as the probe's ID, the command character (**'g'**: GET command, **'r'**: READ command, **'t'**: TEST command), and a Boolean flag for broadcast check. Based on these inputs, the function generates the appropriate frame. As an example, consider the construction of a GET command frame (the other commands are constructed in the same manner). The following code block illustrates the process:

```
void probe_prepare_frame(char *frame, int id, char cmd, bool broad)
{
  char func, dest;
  static char frame[10];
  if (broad)
    dest = BROADCAST_CHAR;
  else
    dest = UNICAST_CHAR;
  sprintf(frame, "%c%c\0", STARTING_CHAR, dest);
  switch (cmd)
  {
  case GET_CMD_CHAR:
    if (broad)
      app_dbg_msg("BROADCAST + GET CMD ERROR ERROR !");
    else
    {
      sprintf(frame + strlen(frame), "%.2d%c\0", id, GET_CMD_CHAR);
      // the frame is "SU(ID)G"
    }
    break;
```

Listing 3.1: Frame Preparing Function (case of "GET command frame")

3.2.3 Frame Exchange between Probes and MCU

Within the communication segment, to enable effective data exchange, three frames will be constructed, each containing either the **"read"** or **"get"** command. Two of the frames will instruct the probes to read voltages and store them temporarily (Broadcast/Unicast READ Frame), while the third frame will instructs the probes to transmit the stored voltage readings to the controller (Unicast GET Frame). This approach ensures a clear focus on either collecting voltage readings or retrieving the stored data from the probes during the communication process.

Sent Frames

- **Broadcast READ Frame:** The purpose of this frame is to serve as a broadcast message, instructing all probes to concurrently perform voltage readings. It is utilized in the **on-off mode** to synchronize the voltage measurement process across all probes.
- **Unicast READ Frame:** The unicast READ frame is individually transmitted to each probe in a sequential manner. Its primary purpose is to instruct each probe to read four specific voltages. This frame is utilized in modes CP-ON and CP-OFF, as well as for testing the functionality of the probes.
- **Unicast GET Frame:** Following the read command, the unicast GET Frame is sent in a sequential manner, ensuring that each probe is addressed individually to retrieve its specific data. Its purpose is to retrieve the previously recorded data from the probes.

Handshake Frames

Handshaking is a vital process in establishing a communication channel between two entities before normal data transfer can occur. It allows for the dynamic negotiation of parameters that govern the communication [11]. In the context of our system, there are two types of handshake frames: read-handshake and broadcast-handshake, each serving a specific purpose as illustrated in Figure 3.5.

1. **The read-handshake** is employed before retrieving data from the probes and is used in both the ON and OFF modes. Additionally, in the PROBE TEST state, this handshake ensures that the probe is responsive to the communication, confirming its functionality.

- **Unicast READ Frame:** as previously stated this frame is sent to the probes individually to indicate that the probes should perform the voltage reading action.
- **READ Acknowledgement Frame:** As a response to the Unicast READ Frame, the probes generate the READ Acknowledgement Frame. This frame contains an 'ok' message, confirming the successful reception of the instructions. In case of an error, the specific probe is flagged, necessitating appropriate action.

2. **The broadcast-handshake** is used before retrieving data in the ON-OFF mode. Since the read frame is broadcasted to the probes, the controller needs to verify if each probe has received the read frame before retrieving data. This step ensures that the data is not duplicated, and prevents the possibility of the probe sending back outdated information stored within it. By performing this handshake, the controller can ensure the accuracy and reliability of the retrieved data.

   - **Broadcast Reception Check Frame:** this command is sent after the Broadcast read to the probes individually (probe by probe) to insure the reception of the Braodcast READ.
   - **Broadcast Acknowledgement Frame:** This frame serves as a response to the Broadcast Reception Check Frame. It contains an 'ok' message, indicating the successful reception of the broadcasted frame. Failure to receive this frame indicates a potential issue with a specific probe, requiring further action.

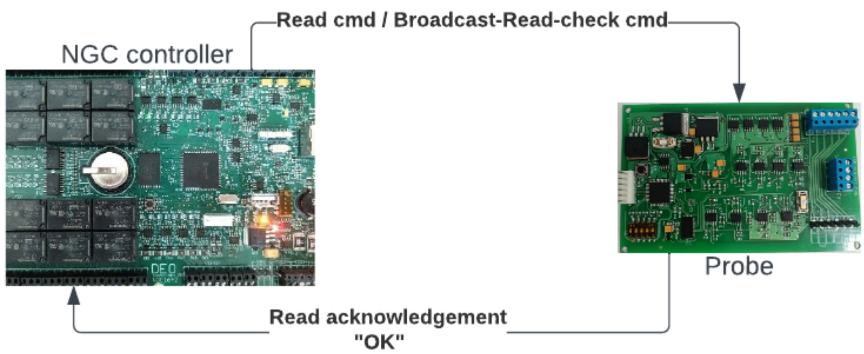


Figure 3.5: Handshake signaling

Data Frame

Once the handshake signals confirm a successful connection, the "get" command is transmitted to the probes to retrieve the stored voltage readings. Subsequently, the probes generate a data frame to send the collected data back to the controller. The structure of the frame consists of the following components:

- **Frame Header:** This section contains the initial part of the "get" frame, including the start character, probe ID, and the "get" character indicating the purpose of the frame.
- **Data:** This segment contains the probe ID represented by two hexadecimal characters, followed by concatenated voltage readings. Each voltage reading is represented by four hexadecimal characters.
- **CRC:** To ensure data integrity, an 8-bit checksum is calculated for the entire data section and appended to the frame.
- **End Character:** The frame concludes with an end character, denoted by an asterisk (*), signifying the completion of the frame. a pracical example is given in Figure 3.6

Figure 3.6: Data Frame

3.2.4 Data Saving

Upon the reception of the frame that contains the collected data, each state contains a mechanism to save these voltages, first the frame is parsed and saved in an array of structures (structure **probe_reading**), as interpreted in Figure 3.7

```
typedef struct
{
  int ID;
  int v1, v2, v3, v4;
} probe_reading;
```

Figure 3.7: Readings structure

Upon saving the voltages in the array, they are first converted into integer values. The array is then saved in an FS (file system) file. Each mode has a different file, which is initialized in the **"probe_init"** function. We use **"writeatomic"** to save the data, as illustrated in Figure 3.8, which means that whenever a new reading occurs, the existing data in the file will be **overwritten**.

```
if (static_file_index != -1)
{
    fs_writeatomic(static_file_index, static_readings, LENGTH);
}
```

Figure 3.8: write to files

3.3 Web Application

To implement the web application, which plays a vital role in our solution, we have chosen the **Django framework**. Django provides a powerful set of tools and features that facilitate the development of robust and scalable web applications.

3.3.1 Django Framework

Django is a high-level web framework written in Python. It follows the Model-View-Controller (MVC) architectural pattern and is designed to simplify and accelerate web development. Django provides a set of tools, libraries, and conventions that allows to build web applications quickly and efficiently [12]. Sating the key features of the framework as follows:

- **Object-Relational Mapping (ORM):** Django includes a powerful ORM that enables developers to interact with databases using Python code instead of writing complex SQL queries. The ORM provides an abstraction layer that maps database tables to Python objects, as illustrated in the example shown in Figure 3.9, making it easier to work with data and perform database operations.

```

zone = Zone.objects.all()
probes = Probe.objects.all()
mode_on = Mode.objects.filter(mode = "ON")
mode_off = Mode.objects.filter(mode = "OFF")
mode_on_off = Mode.objects.filter(mode = "ON-OFF")

```

Figure 3.9: Database models access in python

- **URL routing:** Django offers a flexible and customizable URL routing system (see Figure 3.10). Developers can define URL patterns and map them to corresponding views or controller functions. This allows for clean and organized URL structures, making it easier to handle different requests and navigate through the application.

```
path('send_static_command/', frame.send_static_command, name='send_static_command'),
path('send_off_command/', frame.send_off_command, name='send_off_command'),
path('send_on_command/', frame.send_on_command, name='send_on_command'),
path('send_test_command/', frame.send_test_command, name='send_test_command'),
path("probe_test", probe_views.probe_test, name="probe_test"),
path("probe_charts", probe_views.probe_charts, name="probe_charts"),
path("Home", views.index, name="Home"),
path("", pages.Login, name="login"),
```

Figure 3.10: URL routing in Django

- **Template engine:** Django provides a built-in template engine that allows developers to separate the presentation logic from the application's business logic. Templates are written using HTML and can include dynamic data using Django's template tags and filters. This helps in creating reusable and modular templates for consistent rendering of web pages.

- **Form handling:** Django simplifies form handling by providing a Form API that streamlines the process of rendering and validating forms. Developers can define forms with fields, validation rules, and error handling as shown in Figure 3.11. The framework automatically generates form HTML, handles data submission, and performs server-side validation.

```
from django import forms

class addzone(forms.Form):
    zone = forms.CharField()
    def clean(self):
        cleaned_data = self.cleaned_data
        zone = cleaned_data.get("zone")
```

Figure 3.11: Zones form in Django

- **Authentication and authorization:** Django includes built-in authentication and authorization mechanisms that make it easy to implement user registration, login, and password management. It provides secure handling of user sessions, password hashing, and permission-based access control. It can be set as explained in Figure 3.12

```python
from django.contrib.auth.models import User, auth

def Login(request):
    if request.method == "POST":
        username = request.POST["username"]
        password = request.POST["password"]
        user = auth.authenticate(username=username, password=password)
        if user is not None:
            auth.login(request, user)
            return redirect("Home")
        else:
            messages.info(request, "Invalid Credentials!")
            return redirect("/")
    else:
        return render(request, "pages_login.html")
```

Figure 3.12: Login User Authentication in Django

- **Administration interface:** Django offers a ready-to-use administration interface that allows developers to manage site content without writing custom code. The admin interface provides an intuitive and customizable interface for creating, updating, and deleting data records.

3.3.2 TCP/IP Frames

To establish a connection with the controller, the TCP/IP frames sent from the web server to the controller need to follow a specific structure:

- **Frame Header:** The frame header should contain the characters **"@00G"** before sending any commands.
  - **The "@00"** serves as the start bits of the TCP frame
  - **the character "G"** performs a callback to the **"cb_voltage_prob"** function in the NGC controller. This function handles the necessary actions based on the subsequent character as explained in Figure 3.13.

```c
static int (*cmd_callbacks[])(const char *) =
{
  /* CMD = A */ cb_analog,

  /* CMD = B */ cb_digital_control,

  /* CMD = C */ cb_counter,

  /* CMD = D */ cb_digital,

  /* CMD = E */ cb_electric_panel,

  /* CMD = F */ cb_freq,

  /* CMD = G */ cb_voltage_prob,

  /* CMD = H */ cb_history,
```

```c
static int cb_voltage_prob(const char *frame)
{
  int opt;
  int ret = OK;
  opt = frame[OPT_INDEX]; // OPT_INDEX = 4 CMD_INDEX = 3
  switch (opt)
  {
    case 'a':
    {…
    case 'd':
    {…
    case 'C':
    {…
    case 'R':
    {…
    default:
      return ret;
  }
```

Figure 3.13: "cb_voltage_prob" function

- **Data:** The data appended to the frame specifies the action to be performed by the microcontroller unit (MCU) in the controller. The following commands are possible:
  - **"a":** This command signals the addition of a probe to the system. It should be followed by the **probe ID**. For example, "@00Ga01" indicates that a probe with ID "01" is being added.
  - **"d":** This command is used to delete previously added probes. Similar to the add command, this is followed by the **probe ID**. For instance, "@00Gd01" signifies the deletion of the probe with ID "01".
  - **"C":** This command changes the operating state or mode of the MCU. It is followed by a number representing the selected mode/state as stated in Figure 3.14.

```
typedef enum
{
    STATE_INIT,             // 0
    STATE_OFF,              // 1
    STATE_ON,               // 2
    STATE_ON_OFF,           // 3
    STATE_TEST_PROBE,       // 4

} states_t;
```

Figure 3.14: States structure in the master code

  - **"R":** This command is sent to request the saved voltages from the controller. A frame is subsequently built according to the requested mode. The details of how these voltages are saved beforehand would need to be further explained.

- **CRC:** A 16-bit CRC (Cyclic Redundancy Check) is appended to the frame for error detection. The CRC value is calculated based on the contents of the frame using **fcs** function shown in Figure 3.15.

- **End Char " * ":** the frame is finalized by appending the end character, represented by an asterisk **" * "**. The end character serves as a delimiter, indicating the end of the frame.

```
def prepareframe(frame: str) -> str:
    fcs_str = str(format(calcfcs(frame), 'X')).zfill(4)
    # print(fcs_str)
    return frame + fcs_str + '*'
```

Figure 3.15: 16-bit CRC calculator & end character

From the controller's side, upon receiving the TCP frames, the microcontroller unit (MCU) responds back with one of three different types of responses:

- **Successful Execution:** If the command received in the frame is executed successfully by the MCU, it sends a response indicating that the command has been executed without any issues. by including an **'OK'** message in the response frame.

- **Error Occurrence:** In case an error occurs during the execution of the received command, an error frame is sent as a response. the error frame contains **'ER'** message.

- **Data Frame:** When the voltage readings are requested from the web server, the data frame is sent. This frame contains the probe information (probe ID concatenated with its 4 voltage readings), it is constructed as shown in Figure 3.16

```c
int i;
for (i = 0; i < 16; i++)
{
    sprintf(str + strlen(str), "%.2x%.4x%.4x%.4x%.4x\0", arr[i].ID, arr[i].v1, arr[i].v2, arr[i].v3, arr[i].v4);
}
app_dbg_msg("the built frame is %s\n", str);
```

Figure 3.16: TCP/IP Data-frame Constructor

3.3.3 PostgreSQL Database

PostgreSQL is an open-source Database Management Systems (DBMS) that is maintained and developed by a global community of volunteers. It is not owned or controlled by any specific company or organization. PostgreSQL's source code is freely available to the public without any associated costs. This DBMS follows a relational model for managing databases and offers a wide range of features and functionalities [13]. PostgreSQL is integrated with django as illustrated in Figure 3.17 PostgreSQL has been selected as the primary database for our web app, and it is configured as the main database in the Django settings file.

```python
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.postgresql',
        'NAME': 'manou',
        'USER': 'postgres',
        'PASSWORD': '3214',
        'HOST': 'localhost',
        'PORT': '5771',
    }
}
```

Figure 3.17: PostgreSQL integration with Django

Upon receiving the data frame, the data is parsed to extract the relevant information. Once extracted, the data is then mapped into floating-point readings, converting the numerical representation back into a format of the relevant voltage readings (see Figure 3.18).

```
ID = int(key)
# find the probe with this id
v1 = float_map(-5, 5, 4095, int(dict[key][0]))
voltage1 = "{:.3f}".format(v1)
v2 = float_map(-5, 5, 4095, int(dict[key][1]))
voltage2 = "{:.3f}".format(v2)
v3 = float_map(-5, 5, 4095, int(dict[key][2]))
voltage3 = "{:.3f}".format(v3)
v4 = float_map(-5, 5, 4095, int(dict[key][3]))
voltage4 = "{:.3f}".format(v4)
# find the probe
probe = Probe.objects.get(probe_id= ID)
```

```
data = Mode(
        mode = "ON",
        probe = probe,
        voltage1 = voltage1,
        voltage2 = voltage2,
        voltage3 = voltage3,
        voltage4 = voltage4,
    )
data.save()
print("mode on saved")
```

Figure 3.18: Mapping & saving voltage readings

3.3.4 HTML / CSS

HTML (Hypertext Markup Language) and **CSS** (Cascading Style Sheets) are foundational technologies used for creating and styling web pages [14]. HTML and CSS work together to create the structure, content, and presentation of a web page. HTML defines the elements and their relationships, while CSS defines the visual properties and layout [14]. They are often used in conjunction with other web technologies such as JavaScript to create interactive and dynamic web pages.

3.4 Conclusion

In this chapter, we delved into the software implementation of the theoretical solution that was proposed in chapter two. We discussed the steps that were taken to transform the conceptual design into a functional software system. The proposed protocols and codes are validated in the upcoming chapter (chapter 4)

Chapter 4

Results and discussion

This chapter focuses on testing the functionality of the overall system and verifying the communication between the three ends. Detailed explanations and findings will be presented in the upcoming sections.

4.1 Web Interface

The web application includes a login and registration page as the initial interface presented to the users. On these pages, users can enter their login credentials or register as new users. The entered information is authenticated using the authentication function provided by the Django framework, which verifies the user's credentials against stored data. Upon successful authentication, the user's information is saved in the database for future logins as illustrated in Figure 4.1

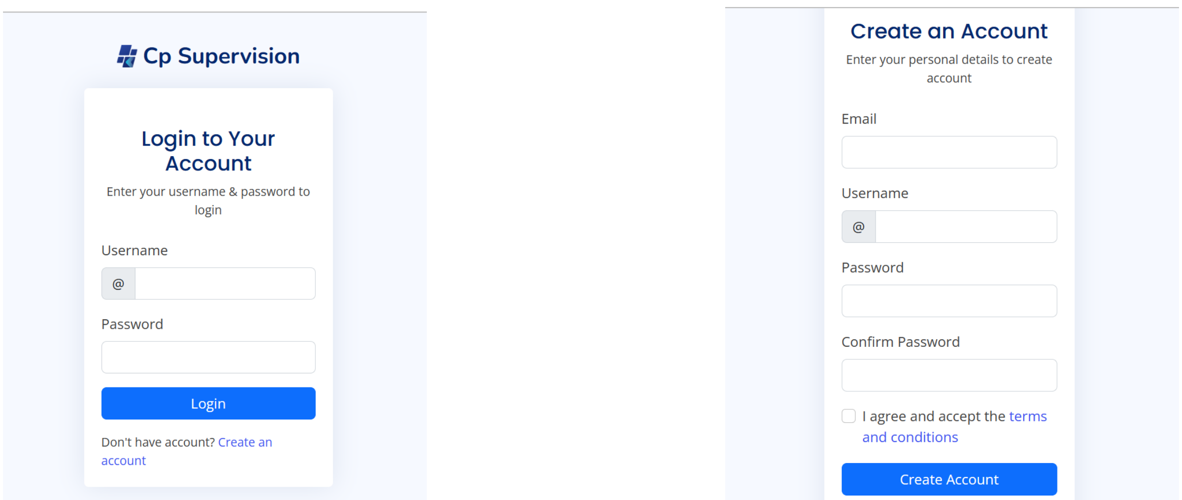


Figure 4.1: Login & Registration pages

If there is any inaccuracy or discrepancy with the logged-in information provided by the user, an error message is displayed to the user. This error message serves as a notification to inform the user about the issue encountered during the authentication process as indicated in Figure 4.2.

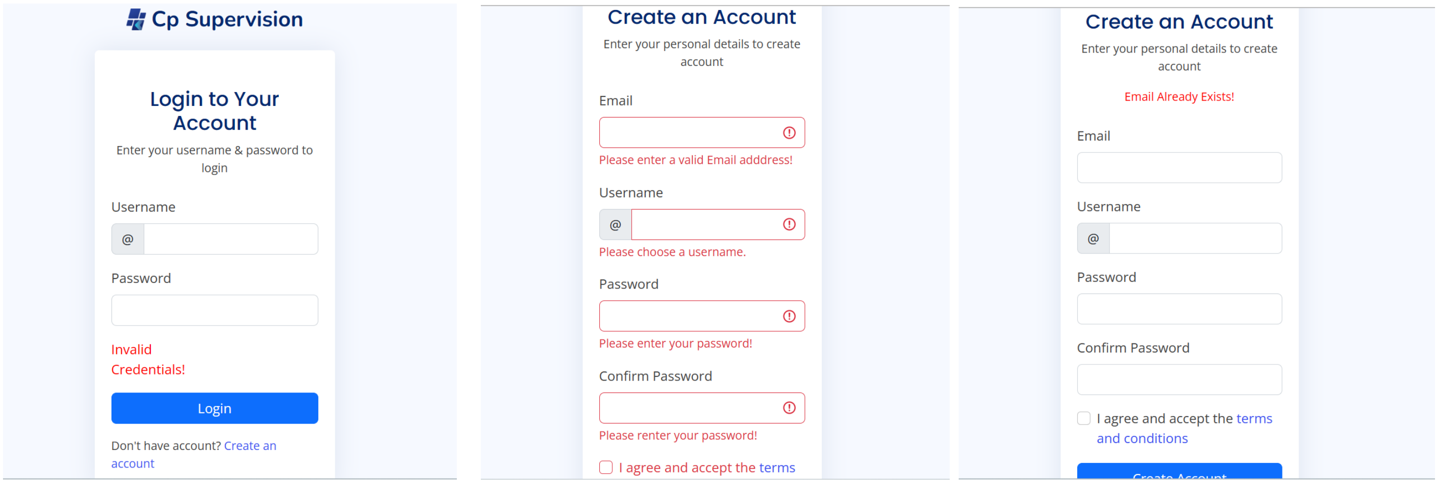


Figure 4.2: Error messages in Login & Registration pages

After a successful authentication, the user is redirected to the home page as shown in Figure 4.3, which features a welcoming interface. The home page provides a user-friendly layout with a sidebar that presents the different functionalities available within the web application.

Figure 4.3: CP Supervision Homepage

The Web application has 3 main functionalities **Configuration of the system**, **Running the Tests** and finally **Displaying the data**

4.1.1 System Configuration

In the Configuration section of the web application, users have access to two pages; **Add Zone** & **Add probe** (see Figure 4.4).

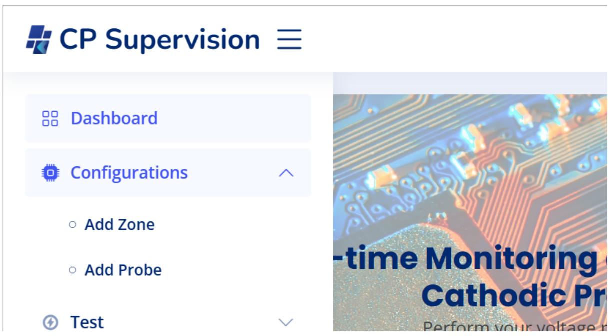


Figure 4.4: Configuration sections

The first page is dedicated to adding different settings for the cathodic protection system, referred to as **"Zones"**. As the cathodic protection system is spread across a wide area of underground pipelines in the field, the entire pipeline area is divided into smaller sections or zones. On this page illustrated by Figure 4.5, users can enter and define the specific zones present in the field.

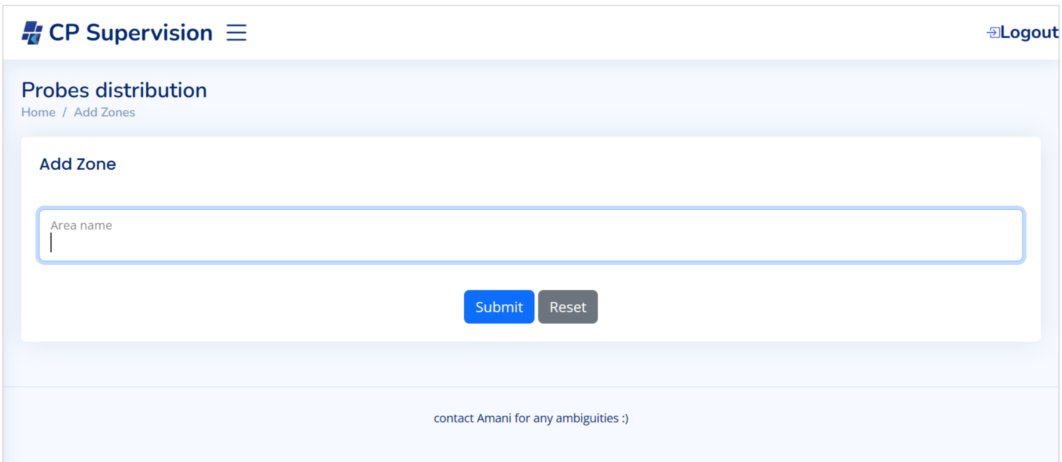


Figure 4.5: Zone form

The second page, shown in Figure 4.6, is a **form** where users can input information about each probe in the cathodic protection system. The form collects details such as the probe's **location area**, **ID**, the **location of the channel's reference electrode**, and optionally, a **description** of the probe's environment if it is relevant. This information helps in accurately identifying and configuring the probes within the system.

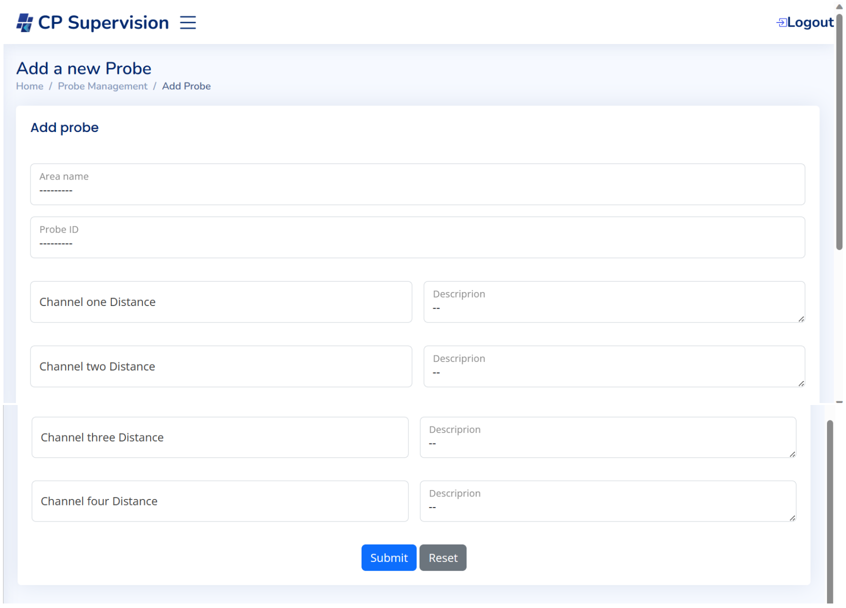


Figure 4.6: Probe Registration Form

The data entered via the form, including zone settings and probe information, is stored in the database,indicated in Figure 4.7 ,for future reference. Users have the flexibility to delete or update the data as needed. This allows for easy maintenance and management over time.

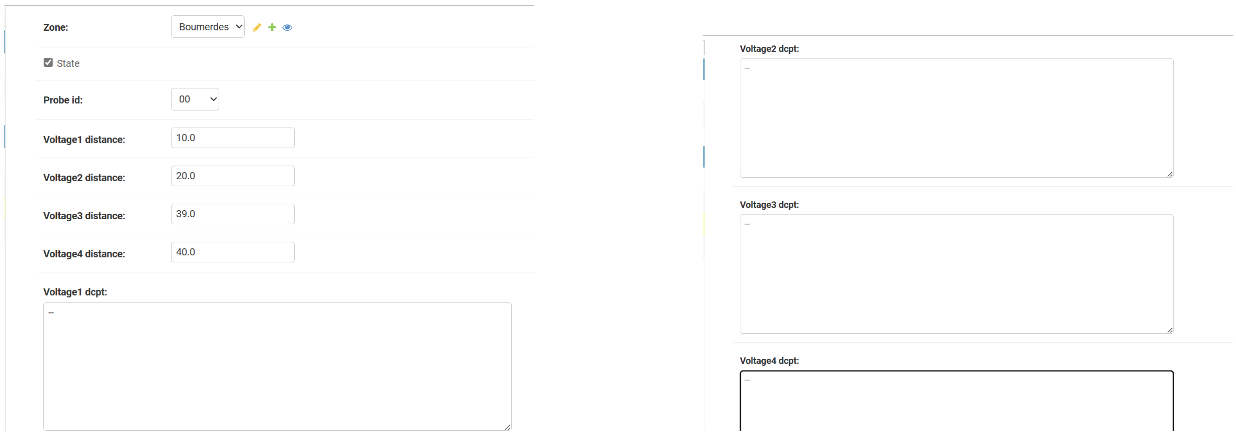


Figure 4.7: Probe Registration Database

4.1.2 Testing & data collection

In the testing section of the web application, users have the ability to execute three different modes to command the probes and obtain voltage readings. Subsequently, the collected data can be gathered by clicking on a separate button as Figure 4.8 illustrates.

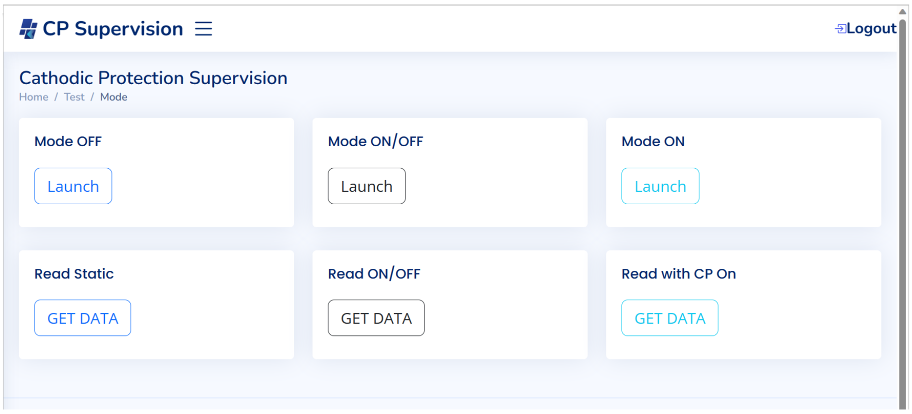


Figure 4.8: Modes Interface

Once collected, the data is parsed and accurately mapped to its corresponding readings, since the data contained in the frame is in hexadecimal format and to scale ADC readings from a 0-5 range to a -5 to +5 range, the data is mapped with the following equation.

$$Data = \frac{(max - min)}{2^{12}} + min \qquad ==> \qquad Data = \frac{(5-(-5))}{4095} - 5$$

These parsed values are then stored in the database. The data is registered using a model that includes the following fields: **probe information** (probe), operation **mode**, four **readings** & gains (calculated as on/off_voltage - off_voltage), **threshold**, and **date** (see Figure 4.9).

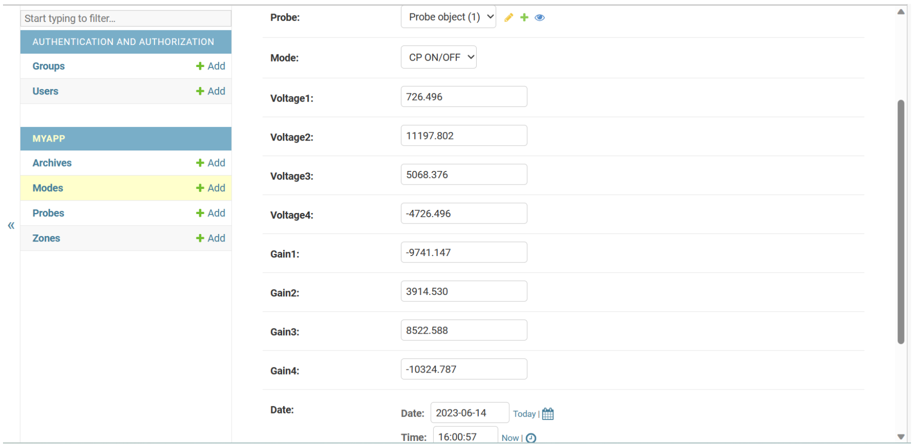
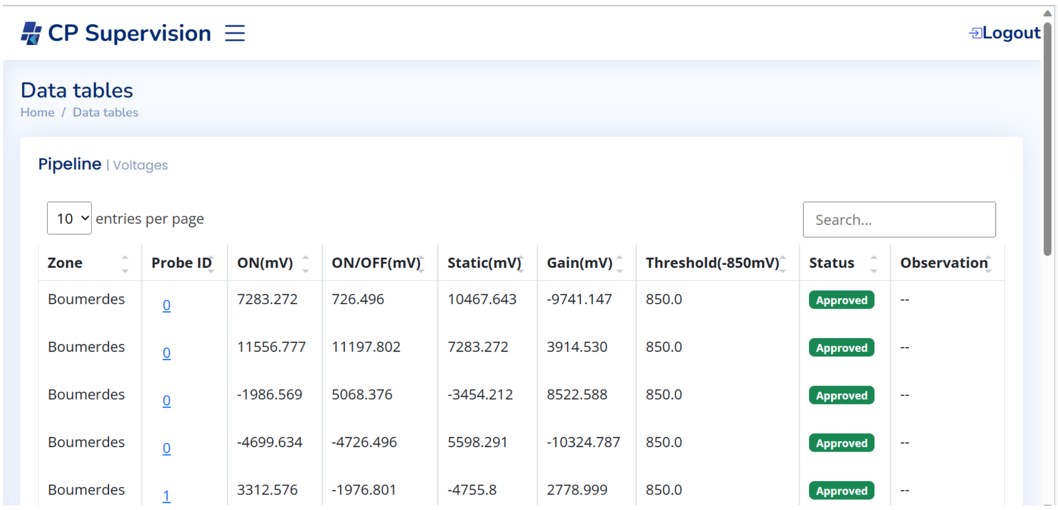


Figure 4.9: Mode ON/OFF database

4.1.3 Data Visualization

Once the data is stored, it is presented in a table format that includes the fields mentioned earlier, along with an additional column for the probe status as shown in Figure 4.10. This status column provides information about the functionality of each probe. If a probe is responding properly and there are no communication errors, the status column displays a green "connected" indicator. However, if there is an issue with the probe or a communication error, the status column shows a "disconnected" indication.



Zone	Probe ID	ON(mV)	ON/OFF(mV)	Static(mV)	Gain(mV)	Threshold(-850mV)	Status	Observation
Boumerdes	0	7283.272	726.496	10467.643	-9741.147	850.0	Approved	--
Boumerdes	0	11556.777	11197.802	7283.272	3914.530	850.0	Approved	--
Boumerdes	0	-1986.569	5068.376	-3454.212	8522.588	850.0	Approved	--
Boumerdes	0	-4699.634	-4726.496	5598.291	-10324.787	850.0	Approved	--
Boumerdes	1	3312.576	-1976.801	-4755.8	2778.999	850.0	Approved	--

Figure 4.10: Data table

Alternatively, the data is also displayed in the form of charts (Figure 4.11), which facilitate enhanced system diagnosis and analysis.

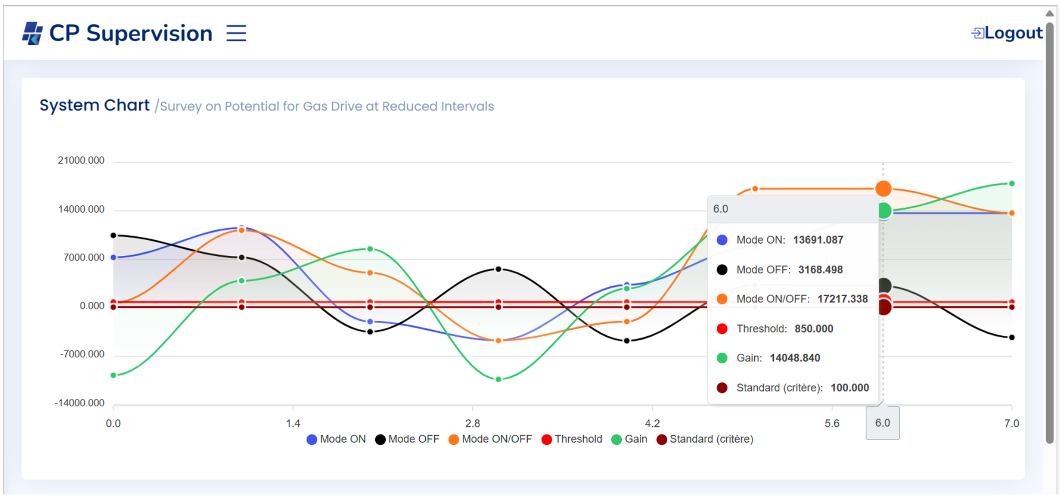


Figure 4.11: Data Charts

4.2 Debugging tools

During the development of our software solution, we performed extensive testing and debugging of our code. One of the tools we utilized for this purpose was the CuteCom debugger.

CuteCom provided us with a convenient and user-friendly interface for monitoring and analyzing the communication between our software and various serial devices. By configuring the appropriate serial port settings in CuteCom, we were able to establish a connection and observe the data exchanged in real-time.

Using CuteCom, we could closely examine the incoming and outgoing data, ensuring that it aligned with the system's expectations and met the requirements of the serial communication protocol, we were able to identify and resolve issues related to data transmission, parsing, and synchronization, as illustrated in Figure 4.12.

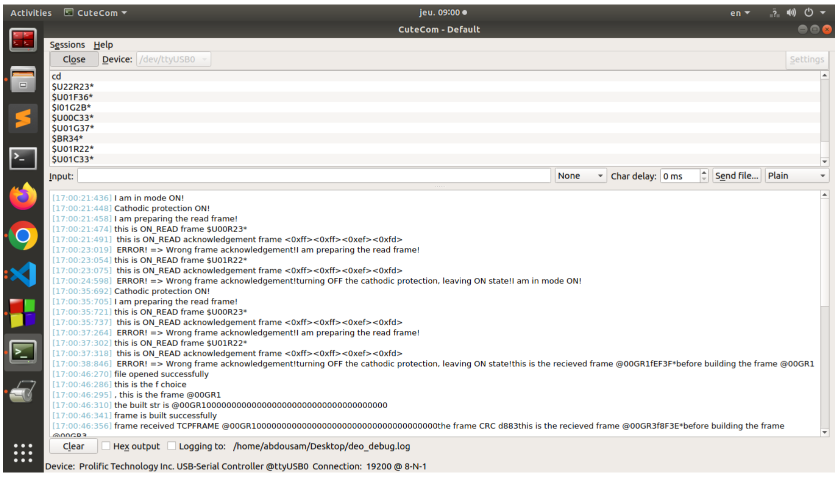


Figure 4.12: Cutecom debugger Interface

4.3 Server-Controller & Controller-Probe Communication

The connection between the three parties (the web application, the controller, and the probes) is monitored and tracked through the debugger and the terminal of the web application's server. When the mode button is clicked in the web application, a frame is sent from the web application to the controller. This frame instructs the controller to enter the corresponding state, based on the selected mode.

Upon receiving the frame, the master code in the controller sends out the necessary frames to initiate the voltage readings from the probes. The probes then perform the voltage measurements as instructed and send the collected data back to the controller.

4.3.1 Mode OFF

When the **"Mode OFF"** button is clicked, the web application sends a frame with the value **"@00GC12E01*"** to the controller. This frame instructs the controller to switch to the off mode.

- **@00** –> Frame header, this is indicates the starting bits of the frame.

- **G** –> This character selects the master code task by the frame processor file in the controller.

- **C** –> This character indicates that the following character should be a number between (1-3) each selects a mode:

  - **1**: selects mode **OFF**
  - **2**: selects mode **ON**
  - **3**: selects mode **ON-OFF**

- **2E01** –> The frame's 16-bit **CRC** and the end character **' * '**.

Upon the reception of the previous frame, the controller responds with an acknowledgement frame, **"@00GC1OK8BF5*"**. The responded frame contains half of the sent frame along with the message **"OK"** which signify a positive acknowledgement. This acknowledgement indicates that the state has been successfully switched, and the probe is now operating in the set mode.

On the controller-probe side, communication is initiated with the registered probes. The program loops over all the used probes and sends the **"READ"** and **"GET"** commands. First, the **"READ"** command is sent, and the controller expects an acknowledgement from the probe before sending the **"GET"** command.

In Figure 4.15, an example of the communication process for the **"Mode OFF"** is illustrated. For the first extension, the **"READ"** frame sent is **"$U00R23*"**, and the acknowledgement received is **"$U00ROK27*"**. Upon receiving the correct acknowledgement, the controller proceeds to send the "GET" frame, **"$U00G36"**. As a response to this, the probes send the read voltages as follows: **"$U00G0000000000000000036"**.

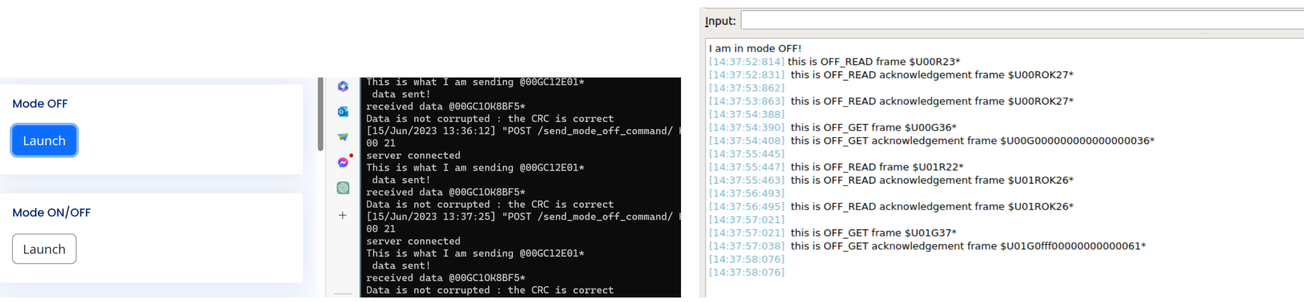


Figure 4.13: Mode OFF communication

4.3.2 Mode ON

similar to the behavior of the off mode, The communication process is the same. However the Cathodic protection is first **turned ON** before initiating the readings. When the **"Mode ON"** button is clicked, the web application sends a frame with the value **"@00GC22F41*"**, the corresponding acknowledgement is **"@00GC2OK8B05*"**. Figure 4.14 is a practical example.

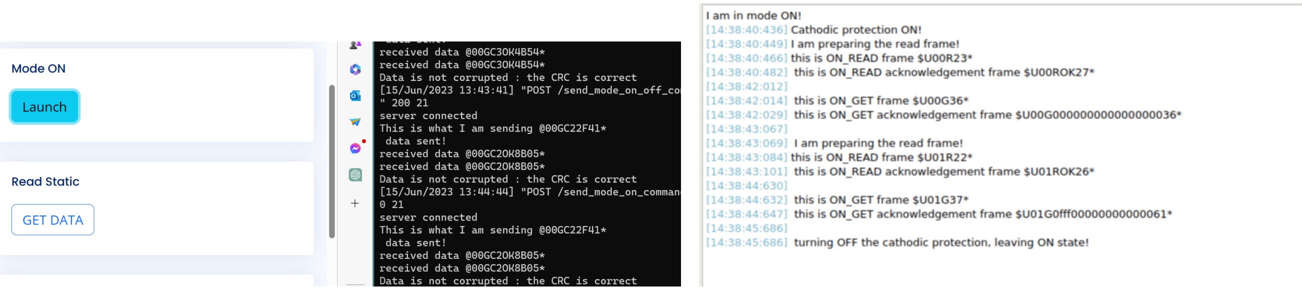


Figure 4.14: Mode ON Communication

4.3.3 Mode ON/OFF

In this mode, the communication between the web application and the controller follows the same pattern as before, with the frame **"@00GC3EF80*"** being sent from the web application and the acknowledgement **"@00GC3OK4B54*"** being received from the controller. However, the communication between the controller and the probes is slightly different in this mode. As mentioned earlier, the readings in this mode need to be synchronized. To achieve this, a broadcast frame is sent first **"$BR34*"**, this frame is received by all connected probes.

First, a broadcast check frame is sent to verify if the broadcast frame has been received by the probe. The frame is **"$U00C32*"**, and an acknowledgement is expected in response. This step is similar to the **READ-GET mechanism** used in the previous modes. The acknowledgement frame received from the probe is **"$U00CYS38*"**. The **"YS"** string is the indicator that the probe is confirming its reception to the broadcast frame.

Once the correct acknowledgement is received, the controller proceeds to send the **"GET"** frame, **"$U00G36"**, to the probes. In response, the probes send the read voltages in the following format: **"$U00G000000000000000036"**. This is similar to the previously explained GET section. Figure 4.15 is an example of this mode.

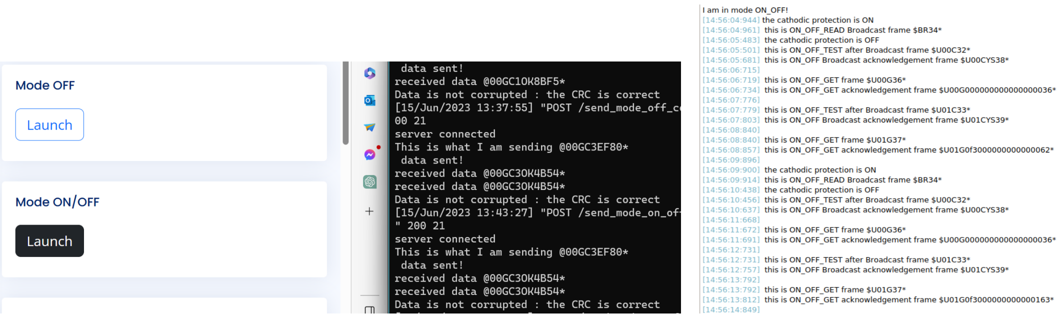


Figure 4.15: Mode ON/OFF Communication

4.3.4 Getting data from Controller

Once all the readings are completed, the user can retrieve the data by clicking on the **"Get data"** button. The data is then transmitted back in a frame format, following the structure (ID-4VOLTAGES-ID-4VOLTAGES-...), where each ID corresponds to a connected probe and the subsequent four voltages represent the readings obtained from that probe. The frame contains data for all the connected probes, which is stored in the **"fs"** file on the controller. Upon receiving the data frame, it is parsed and registered in the database using the **"mode"** model. Each entry in the database corresponds to a specific probe ID and its associated four readings. An example on the obtained results (Mode ON) can be visualized in the Figure 4.16

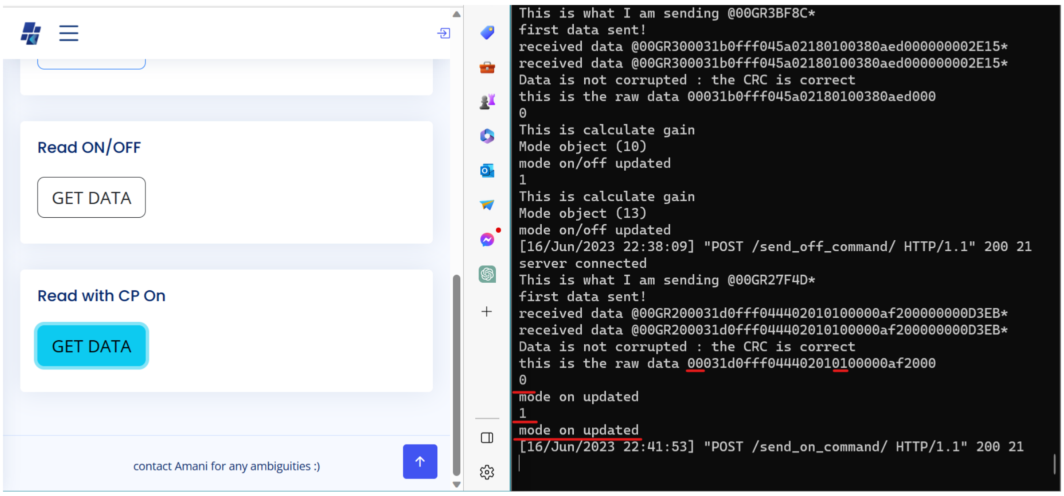
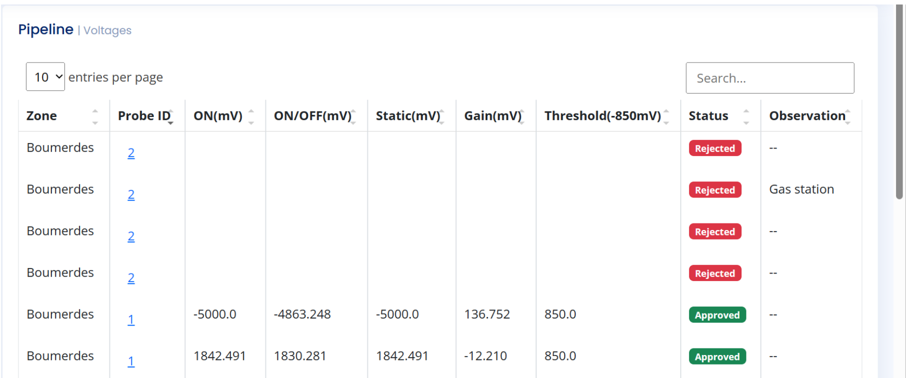


Figure 4.16: Get Mode ON Data Communication

If any communication issues arise with the probes, such instances will be reflected in the datatable. The status of the affected probes will be displayed as **"rejected"** to indicate the miscommunication or failure to establish a connection. Figure 4.17 illustrates a failed case where Probe with ID 2 was not connected to the system. As a result, it is depicted in the figure with no data and flagged as "rejected" to indicate the communication failure or absence of readings from that particular probe.


Pipeline | Voltages

10 entries per page

Search...

Zone	Probe ID	ON(mV)	ON/OFF(mV)	Static(mV)	Gain(mV)	Threshold(-850mV)	Status	Observation
Boumerdes	2						Rejected	--
Boumerdes	2						Rejected	Gas station
Boumerdes	2						Rejected	--
Boumerdes	2						Rejected	--
Boumerdes	1	-5000.0	-4863.248	-5000.0	136.752	850.0	Approved	--
Boumerdes	1	1842.491	1830.281	1842.491	-12.210	850.0	Approved	--

Figure 4.17: Probe Communication ERROR

4.4 Conclusion

In conclusion, the results obtained from the implementation of our automated solution demonstrate its effectiveness. The communication between all the endpoints, including the controller, probes, and web application, is successfully initiated and established. This ensures seamless data exchange and enables efficient monitoring and control of the cathodic protection system.

Conclusion & Perspectives

In conclusion, this project has successfully developed and implemented a smart voltmeter network along a pipeline, revolutionizing the data collection and analysis processes. Through our team's meticulous design and engineering efforts, the smart voltmeters have demonstrated their effectiveness in streamlining operations.

The utilization of a PIC architecture-based microcontroller has ensured efficient data handling and storage, providing a robust foundation for data acquisition. Moreover, the integration of a web application has enhanced the accessibility and usability of the collected data, offering a user-friendly interface for data interpretation and analysis.

Looking towards the future, there is immense potential for further advancements. One notable area of development is the automation of data analysis, leveraging advanced algorithms to gain deeper insights from the collected data. Additionally, implementing an alert system within the web application can promptly notify users of any system errors or anomalies, enabling timely interventions and maintenance.

Hence, this project has laid a solid foundation for the ongoing enhancement and optimization of the smart voltmeter network. It has demonstrated the potential for leveraging advanced technologies to improve operations in the gas and oil industry. We are proud of the achievements thus far and look forward to continued progress in this field.

Bibliography

[1] "Corrosion and corrosion prevention," Available at https://www.electrochem.org/corrosion-science/.

[2] T. Collins, "5 different types of corrosion prevention methods," Available at https://eoncoat.com/corrosion-prevention-methods/.

[3] Z. Ahmad, *Principles of Corrosion Engineering and Corrosion Control*, 1st ed. Elsevier, 2006.

[4] V. N. Keir Wilson, Mohammed Jawed, "The selection and use of cathodic protection systems for the repair of reinforced concrete structures," *Construction and Building Materials*, vol. 39, pp. 19–25, February 2013.

[5] A. K. Dey, "Cathodic protection: Definition, working principles, types, design, advantages, applications," Available at https://whatispiping.com/cathodic-protection.

[6] M. E. Orazem, *Underground Pipeline Corrosion*, 1st ed. Woodhead Poblishing, 2014.

[7] J. Tetazoo, "direct current voltage-gradient dcvg survey," Available at http://www.alphaleak.com/direct-current-voltage-gradient-dcvg-survey/.

[8] E. CLASSES, "What is rs485 serial communication?" Available at https://www.youtube.com/watch?v=Q9faEKid7RA.

[9] alciro, "Topology of rs-485 connections," Available at http://www.alciro.org/alciro/RS-485_16/topologia-conexiones-RS-485_329_en.htm.

[10] Cardinal_Peak, "Understanding the cyclic redundancy check," Available at https://www.cardinalpeak.com/blog/understanding-the-cyclic-redundancy-check.

[11] H. Njoroge, "4.1: Fundamentals i/o- handshake and buffering," Available at https://workforce.libretexts.org/Bookshelves/Information_Technology/Information_

Technology_Hardware/Advanced_Computer_Organization_Architecture_(Njoroge)/04%3A_Strategies_and_Interface_I_O/4.01%3A_Fundamentals_I_O-_handshake_and_buffering#:~:text=interrupt%20status%20set.-,Handshaking,O%20devices%20data%20transfer%20rate.

[12] "Django introduction," Available at https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/IntroductionÂš.

[13] "Postgresql - overview," Available at https://www.tutorialspoint.com/postgresql/postgresql_overview.htm.

[14] "Html & css," Available at https://www.w3.org/standards/webdesign/htmlcss#:~:text=HTML%20(the%20Hypertext%20Markup%20Language,for%20a%20variety%20of%20devices.