Economic power consumption system without full area blackout

Md Abid Hasan Raihan	**ID:EEE2003021025**
Md Nur Alam Patawary	**ID:EEE2002020060**
Ibrahim Khalil	**ID:EEE2101022079**
Din Islam	**ID:EEE2002020030**
Nazifa An-Noor	**ID:EEE1803015073**
Niamul Karim	**ID:EEE1901016141**

Department of Electrical and Electronic Engineering

Sonargaon University, Dhaka.

January, 2024

Economic power consumption system without full area blackout

A PROJECT

Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF SCIENCE

IN

ELECTRICAL AND ELECTRONIC ENGINEERING

BY

Md Abid Hasan Raihan
ID No:EEE2003021025

Md Nur Alam Patawary
ID No:EEE2002020060

Ibrahim Khalil
ID No: EEE2101022079

Din Islam
ID No:EEE2002020030

Nazifa An-Noor
ID No:EEE1803015073

Niamul Karim
ID No:EEE1901016141

Under Supervision of

Nurul Ambia Alaul

Lecturer,
Department of EEE
Sonargaon University, Bangladesh

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING.

Sonargaon University Dhaka

January, 2024

ACKNOWLEDGEMENT

We would like to express our deep gratitude to our honorable supervisor **Nurul Ambia Alaul,** Department of Electrical & Electronic Engineering, **Sonargaon University**, Bangladesh, Dhaka. We are very much grateful to make this project under his supervision. He has supported us with his valuable and constructive suggestions during the planning and development of this project. His willingness to give his time so generously has been very much appreciated.

Finally, we would like to thanks again to the respected **Vice- Chancellor of SU, Professor Shamim Ara Hassan** also thanks to **Head of Department of SU,** Electrical & Electronics Engineering**, Assistant Professor Md. Rais Uddin Mollah** because they are designated such an environment for learning through which we got the opportunity to acquire knowledge under B.Sc in EEE program, and that will be very helpful for our prospective career.

We would also like to extend our thanks to the staffers of the laboratory of the Department of Electrical & Electronic Engineering for their help in offering us there sources to complete the project. Finally, we wish to acknowledge those who directly or indirectly helped us to complete this project.

ABSTRACT

As the distribution network has become more sophisticated, substation automation has become a requirement for every utility firm. It is vital to understand what type of constraint has occurred to improve the quality of electricity. Furthermore, if a power system's safety, monitoring, and control are insufficient. As a result, a monitoring system that can automatically discover, monitor, and classify existing electrical line restrictions is required. The goal of this project is to collect remote electrical data such as the current, and the power station, and deliver these real-time values across the network. This project also includes a relay that will protect the electrical circuits. When the electrical parameters surpass the predefined values, this relay is actuated. This system can update real-time electrical parameters regularly (depending on time settings). This system can be set up to transmit alerts whenever the relay trip current exceeds predetermined limitations. This project uses a microcontroller; however, because this is a prototype of the proposed concept, we have used an Arduino Uno for demonstration purposes. The controller can communicate with the computer promptly. When we turn on the power to our prototype, all of the sensors begin sensing current and sending real-time information to the server and onto the display. It compares all real-time values to pre-defined values and sends a fault alarm to the relay and buzzer, as well as updates the display if any of the values exceed the pre-defined values. Relay isolates the loads from the rest of the system if the problem persists for the pre-determined time. Meanwhile, the comparison continues as before, and if the fault is cleared, the loads are reconnected to the rest of the system via relays.

CONTENTS

Acknowledgment i

Abstract ii

List of Figures vii

List of tables ix

Acronyms and abbreviations x

CHAPTER 1: INTRODUCTION

1.1 Introduction 1

1.2 Problem Statement 1

1.3 Objectives 2

1.4 Necessity of Automation 2

1.5 Advantages of Automation 3

1.6 Disadvantage of Automation 3

1.7 Chapter Summary 3

CHAPTER 2 LITERATURE REVIEW

2.1 Introduction 4

2.2 Fundamentals of Automation 4

2.3 Various Methods of Load Controlling and Monitoring 4

2.3.1 Design and Implementation of SCADA System 4

2.3.2 Microcontroller-Based Substation with GSM 5

2.3.3 Real-Time Monitoring of Substation with Cloud Computing 5

2.3.4 Zigbee-Based Wireless System for a Substation 7

2.3.5 Intelligent Substation Automation Systems 8

2.3.6 Development of a Condition Monitoring System 9

2.3.7 Comparing the Reliability of Ethernet Network Topologies 10

2.3.8 Smart Substation Art and Future Development 11

2.3.9 Impacts of Automated Control Systems on Substation Reliability 11

2.4 Chapter Summary 12

CHAPTER 3: DETAILS OF EQUIPMENT

3.1 Introduction 13

3.2 Equipment Description 13

3.2.1 ESP8266 Microcontroller 13

3.2.1.1 Overview 14

3.2.1.2 Features of ESP8266 14

3.2.1.3 Pin Diagram of ESP8266 15

3.2.1.4 Function of the Pins 16

3.2.2 Relay Module 17

3.2.2.1 Description 18

3.2.2.2 Specification 18

3.2.2.3 Schematic Diagram of 1 relay module 18

3.2.3 LCD Display(20*4) 19

3.2.3.1 Description 19

3.2.3.2 Features 20

3.2.3.3 Pin Diagram of LCD display 21

3.2.3.4 Pin Function of LCD 22

3.2.3.5 Various Parameters of LCD Display 23

3.2.4 Current sensor 25

3.2.5 Push Button Switch 26

3.2.6 Chapter Summary 27

Chapter 4 Hardware Implementation

4.1 Introduction 28

4.2 Block Diagram of the Project 28

4.3 Flow Chart of the Project 30

4.4 Multiplexer Design Using 2 Relay 31

4.5 Connection Diagram of the Project 32

4.6 App Development Using MIT App Inventor 33

4.6.1 Introduction 33

4.6.2 Basic App Development Performed In App Inventor Compass 33

CHAPTER 5 RESULT AND DISCUSSION

5.1 Introduction 34

5.2 Pictorial View of Final Set-Up 34

5.3 Monitoring of Substation Feeder 35

5.3.1 When Feeders are Operated at Normal Load 35

5.3.2 When feeder-1 is Overloaded &Tripped 36

5.3.3 When feeder-2 is Overloaded &Tripped 36

5.4 Controlling of Substation Feeder 37

5.4.1 App Controlling Mode and All Feeders Are On 37

5.4.2 When Send Feeder-A Off Command from the App 38

5.4.3 When Send Feeder-A & B Off command from the App 39

5.5 Discussion 40

CHAPTER6: CONCLUSION AND FUTURE SCOPE

6.1 Conclusion 41

6.2 Future Scope 42

List of Figures

Figure 3.1: ESP 8266 microcontroller **12**

Figure 3.2: ESP8266 pin diagram 13

Figure 3.3: 1 channel relay module 16

Figure 3.4: Schematic diagram of 1-channel relay module 17

Figure 3.5: LCD Display(20*4) 18

Figure 3.6: Pin diagram of 20*4 LCD display 19

Figure 3.8: Push button switch 23

Figure 4.1: Block diagram of the project 24

Figure 4.2: Flow chart of the project 25

Figure 4.3: 4:1 Mux design using 2-channel relay 26

Figure 4.4: Connection diagram 27

Figure 4.5: Drag and drop code window 29

Figure 4.6: App Interface 30

Figure 4.7: Real-time database 30

Figure 5.1: Pictorial view of the project 32

Figure 5.2: Feeders operated at normal load 33

Figure 5.3: Display and app status at normal condition 34

Figure 5.4: When feeder-1 is overloaded& tripped 34

Figure 5.5: Display and app status when feeder 1 is overloaded 35

Figure 5.6: When feeder-2 is Overloaded & tripped 35

Figure 5.7: Display and app status when feeder 2 is overloaded 36

Figure 5.8: When feeder-3 is Overloaded & tripped 36

Figure 5.9: Display and app status when feeder 3 is overloaded 38

Figure 5.10: App controlling mode and display status 37

Figure 5.11: All feeders are ON 37

Figure 5.12: Feeder-A of command from App and display status 38

Figure 5.13: Feeder-1 is tripped 39

Figure 5.14: Feeder-A of command from App and display status 39

Figure 5.15: Feeder-1 & 2 are tripped 40

LIST OF TABLES

Table 3.1: GPIO overview 15

Table 3.2: Pin description of LCD 20

Table 3.3: Various parameters of LCD display 21

Table 4.1: Truth table of 4:1 Mux design using 2-channel relay 26

LIST OF ACRONYMS AND ABBREVIATIONS

SCADA -Supervisory Control and Data Acquisition

RTU -Remote Terminal Unit

PCS -Process Control Systems

PLC- Programmable Logic Controller

GSM -Global System for Mobile Communications

SVCMS -Smart Voltage and Current Monitoring System

IoT-Internet of Things

SAS-Substation Automation Systems

USB- Universal Serial Bus

ADC -Analog to Digital Converter

MTU-Master Terminal Unit

LAN- Local Area Network

EPRI-Electric Power Research Institute

IEDs -Intelligent Electronic Devices

PWM- Pulse With Modulation

SPI Pins-Serial Peripheral Interface

SCK-Serial Clock

TTL-Transistor Transistor Logic

CHAPTER 1
INTRODUCTION

1.1 Introduction

Electricity is an especially handy and helpful variety of energy. It plays an associate diver-growing role in our modern industrial society. The electric power systems area unit was extremely non-linear, very immense, and sophisticated networks. Such wattage systems are unit unified for economical edges, enhanced irresponsibleness, and operational blessings. They're one of the foremost vital parts of each national and international infrastructure, and once these systems collapse it results in major direct and indirect impacts on the economy and national security. Transmission and distribution systems are the two subsystems that make up electric power systems. A transmission system's primary function is to transfer electricity from generators to customers, whereas a distribution system serves as the final link between high-voltage transmission systems and consumer services. In other words, power is delivered from the distribution system to various customers via feeders, distributors, and service mains. It is vital to understand what type of constraint has happened to increase the quality of power with sufficient solutions. Furthermore, if a power system's safety, monitoring, and control are inadequate, the system may become unstable. As a result, a monitoring system that can automatically discover, monitor, type, and classify existing electrical line restrictions is required. This benefits both the end users and the utility corporations.

1.2 Problem Statement

As the distribution network has become more complicated, substation automation has become a requirement for every utility business to maximize efficiency and improve the quality of power distribution. Because of the lack of automated analysis and the utility's inadequate sight over the grid, there are still power failures and blackouts today.WSN will provide the utility with the required view by gathering data from the

grid's many sub-systems. A sensor node will decide whether to instantly alert the sink about this information or to slightly delay this notification. The distance between the generators and the load could be hundreds of miles, resulting in a massive quantity of power exchange over vast distances as a result of poor quality electric power. Issues with power quality were not regularly observed throughout the early stages of development. The alarm has been triggered due to the increased demand for electricity on the customer side, which has raised the quality of power delivered to the user side. During the transportation of general power, a significant amount of power is lost, resulting in a drop in the quality of power received at the substation. It is vital to be familiar with it to improve the quality of electricity with a suffering solution.

1.3 Objectives

The objectives of this project are-

i. To control and monitor the substation.
ii. To develop an Android app for controlling and monitoring substations.
iii. To develop the economy by providing reliable energy to the consumer load.

1.4 Necessity of Automation

i. The equipment and machinery used to make production more efficient are referred to as automation in the textile industry. Automation has several advantages, including fewer work hours for the same amount of output, better working conditions, and a higher-quality product.
ii. Automation, also known as automatic control, is the use of various control systems to operate equipment such as machinery, factory processes, boilers, heat-treating ovens, telephone network switching, ship steering and stabilization, and other applications and vehicles with minimal or no human intervention.

iii. Automation in the industrial workplace improves productivity and quality while reducing errors and waste, enhancing safety, and giving the manufacturing process more flexibility. Finally, industrial automation improves safety, dependability, and profitability.

1.5 Advantages of Automation

i. **Lower operating costs**: Depending on the assignment, robots can do the work of three to five people. In addition to labor savings, energy savings from automated operations can be significant due to lower heating requirements. Robots speed up processes and improve part accuracy, resulting in less waste in your operation.

ii. **Improved worker safety:** Workers have been freed from harmful tasks thanks to automated cells. Your workers will thank you for protecting them from the dangers of working in a factory.

iii. **Faster ROI:** Automation solutions are tailored to your specific requirements and objectives, and they quickly pay for themselves through lower operating costs, shorter lead times, increased output, and other benefits.

iv. **Consistent and improved part production and quality:**In comparison to human workers, automated cells often complete the production process with less unpredictability. As a result, product quality can be more controlled and consistent.

1.6 Disadvantage of Automation

i. One of the most common drawbacks of automation, job displacement, has already been explored. Regardless of the potential social benefits of retraining displaced workers for other employment, the worker whose job has been taken over by a machine will almost always experience emotional stress. Aside from being displaced from his or her job, the person may also be relocated geographically. A person may have to relocate to find new work, which is another source of stress.

ii. Other downsides of automated equipment include the significant capital investment necessary (an automated system can cost millions of dollars to

design, build, and install), a higher level of maintenance required than with a humanly controlled machine, and a generally lesser degree of adaptability.

1.7 Chapter Summary

Automatic control and monitoring system is required not only sub-station load control and monitoring but also very necessary in our everyday life at home as well as in industry. But there are some drawbacks also. In this chapter, we tried to all of these topics clearly

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The operation of a substation involving an automatic control system is developing day by day. The automatic control system is more convenient and easy to operate and protects the substation. But this automation system is not invented at once. During the earlier development stages, the issues on the quality of power are not frequently reported. It is very simple when it is developed initially. Then the automation system is upgraded day by day. In this chapter, we will try to describe the background of the automation system using substation.

2.2 Fundamentals of Automation

Automation is the use of machines to accomplish jobs that were formerly performed by humans or, increasingly, things that would otherwise be impossible. Although the term "mechanization" is sometimes used to refer to the simple substitution of machines for human labor, automation normally refers to the integration of machines into a self-governing system. Automation has revolutionized the fields in which it has been implemented, and it has influenced almost every facet of modern life.

2.3 Various Methods of Load Controlling and Monitoring

Various Methods of Load Controlling and Monitoring are described below-

2.3.1 Design and Implementation of SCADA System [1]

The combination of telemetry and data gathering is known as SCADA (supervisory control and data acquisition system). SCADA entails gathering data using an RTU (remote terminal unit), sending it to a central location, performing any necessary analysis and control, and then displaying it on several operator screens or displays. A SCADA system collects information from sensors and devices that are positioned on the other side of the building. The data is then sent to a central location for monitoring by the controller. One or more field data interface devices form the basis of a SCADA system. A radio, telephone, cable, satellite, or other kind of communication. A

centralized host computer server (also known as a SCADA center or Master Terminal Unit (MTU)).

SCADA systems, which integrate humans, computers, communications, and procedures, control these processes. This Automation systems used to improve process control efficiency by substituting low computer system expenses for high people costs. They also help to boost performance by utilizing faster computer control rather than human reaction times. The extensive usage of these automation systems, also known as process control systems (PCS) or supervisory control and data acquisition (SCADA) systems, makes them crucial to the safe, dependable, and efficient running of many physical processes. SCADA implements more efficient control, improves plant and personnel safety, and reduces costs of operation.

PLC-based SCADA system is complex so it requires skilled operators, analysts, and programmers. In our project, we try to remove the complexity of the SCADA system.

2.3.2 Microcontroller-Based Substation with GSM [2]

The goal of this project is to collect remote electrical data such as voltage, current, and frequency, as well as the temperature at the power station and broadcast these real-time numbers via the GSM network using a GSM modem/phone.

This project also includes the use of an Electromagnetic Relay to protect the electrical circuits. When the electrical parameters surpass the predefined values, this Relay is activated. The Relay can be used to turn off the main electrical supply by operating a Circuit Breaker. To read the distant electrical characteristics, the user can send commands in the form of SMS messages. This system can also provide real-time electrical parameters such as SMS messages regularly (depending on time settings). This system can be set up to send SMS notifications anytime the circuit breaker trips or the voltage or current exceeds predetermined limitations.

By implementing this project we can control and monitor the substation from any distance, But in this system, there are some limitations also. The GSM system does not work properly. Sometimes it takes time to send and receive SMS. To solve this problem

we use Android apps that do not need a GSM module and this app can easily and first communicate with Arduino.

2.3.3 Real-Time Monitoring of Substation with Cloud Computing [3]

Smart grid technology has recently advanced, allowing for real-time control and monitoring of the power system via bidirectional communication. With the application of its architecture, the smart grid is necessary to increase reliability, efficiency, security, and cost-effective power management. In this project researcher creates a proposal based on the application of cloud computing for energy management and information management of the smart grid, taking into account all of these consumer expectations.

In this project, the disadvantages of a smart grid without cloud computing are overcome. Multiple powers, such as active and reactive power, were also implemented, including reactive power adjustment. Smart grids are made up of smart measuring devices that send data to the cloud. In the smart grid, cloud computing serves a variety of purposes. Energy management, information management, and bidirectional communication between the consumer and the utility are all provided via cloud computing. Utility companies work to balance demand and supply by solving consumer problems.

The capacitor bank in this project is a cost-increasing factor in the circuit. As a result, for continuous circuit operation, it is necessary to avoid capacitor banks. in our project, we do need not any capacitor bank for controlling and monitoring so our project is lower-costing

2.3.4 Zigbee-Based Wireless System for a Substation [5]

To overcome the constraints of the current substation monitoring system, this article implemented a Zigbee-based wireless network system in both the traditional and wireless online monitoring systems. A Zigbee sensor network node and a Zigbee coordinator node make up this system. Sensor terminal nodes are made up of various types of sensors and Zigbee wireless modules that are evenly distributed throughout the monitoring area and are responsible for data collecting and transmission. The control

center of the Zigbee network, the coordinator node is responsible for network maintenance and data processing.

This paper's advances resulted in more efficient monitoring and control. This project serves as a prototype for any electrical systems that require online monitoring and control. This project can implement a wide range of commercial buildings, structures, and other projects. The Arduino is used to control this project. XCTU can interface with several Zigbee and substation regulating parameters can be monitored by a substation app produced by an MIT app developer. The Node MCU is a Wi-Fi module that monitors and controls current and voltage in the current and voltage transformers, as well as temperature across the transformer. By connecting relays to transformers and controlling them remotely through the Internet, the supervision and control of a substation utilizing the Internet of Things (IoT) have been experimentally proven to work satisfactorily. The system is meant to not only monitor sensor data such as voltage, current, and temperature but also to actuate a process following the requirements. This control system is designed for substation applications with numerous output and input configurations. The module is small, quiet, and adaptable, and it can conduct a variety of I/O operations using Zigbee. The existing system has several flaws, including complicated wiring, expensive maintenance costs, and a limited control range. This system was created to address the shortcomings of the previous system. This system eliminates the need to dispatch a service technician to a specific application and maintains industry control. As a result, it saves labor, time, and money.

But In this project, there are some limitations also. The replacement cost is high in the ZigBee-based system when there need for any replacement of equipment but in the case of our project replacement cost is very low for this system.

2.3.5 Intelligent Substation Automation Systems [6]

The Substation Automation Systems (SAS) used in the smart grid were provided in this research as a bedrock for ensuring reliable smart grid development in electric utilities. As a result of the state-of-the-art computerized monitoring, control, and protection capabilities, there was a lower outage rate. In other words, the smart grid system's

reliability index has improved. The inextricable interdependence of cyber and power components in an automated substation, on the other hand, makes the operation process more vulnerable. In this way, unlike a power component failure in a substation, a failure of cyber components in an Ethernet network might also disrupt operations. As a result, selecting a SAS package that provides more consistent performance can help to protect the system against enormous misbehavior. Interoperability of multiple SAS components with several manufacturers' brands has become possible since the launch of the multi-vendor SAS-based IEC-61850 protocol. This study examines the most efficient and widely used SAS package, as well as the arrangement in a high-voltage substation that leads to high reliability. This study examines the most efficient and widely used SAS package, as well as the arrangement in a high-voltage substation that leads to high reliability. The findings can help to pave the way for future smart grid development.

This system is most suitable for high voltage substations but not economically applicable in low voltage substations and homes, So we try to develop a system that is most suitable in high or low voltage substations as well as home appliance load control and monitor.

2.3.6 Development of a Condition Monitoring System [7]

Maintenance is a critical task for utilities to ensure that their networks run smoothly. And it necessitates a significant investment of both human and financial resources. To reduce maintenance expenses, it is necessary to invest in innovation. A shift from preventive maintenance based on time (TBM) to preventive maintenance based on condition (CBM) is underway in the electrical utilities industry to increase maintenance effectiveness. The establishment of a condition monitoring system is a key component of the CBM approach.

This paper discusses the creation and deployment of a condition monitoring system in an electric utility's pilot plant, which includes a 66/25 kV, 30 MVA power transformer as well as 66 kV and 25 kV circuit breakers. The monitoring functionalities of commercial protection relays with IEC61850 communication capability were used to

create the system. The monitoring system is described first, followed by a description of the pilot plant where the system was placed, and finally, conclusions for future work are offered in this study. A condition monitoring system has been installed in a substation. It is an open system that may be installed in any type of substation, new or old, regardless of the manufacturer's assets. It was also mentioned that IEDs had monitoring capabilities, as well as the new IEC 61850 standard. Monitoring systems will make maintenance more efficient because they will be carried out based on the present state of the assets. It will also improve the reliability of power infrastructure and, as a result, the power quality that customers receive.

In this there also some limitation also. To establish a generic maintenance management system, more work on data management is required. Monitoring is the initial step toward a CBM, followed by RCM. But in our project, no need for more work on data management, the working process of our project is very simple and easily handle.

2.3.7 Comparing the Reliability of Ethernet Network Topologies [8]

In some substations, electric utilities have installed Ethernet networks. More electric power substation networks appear to be incorporating Ethernet shortly. The Internet and Ethernet local area networks (LANs) in offices and industries have resulted in a plethora of Ethernet equipment, software, tools, and expertise. The Utility Communications Architecture is being defined and deployed by electric utilities, the Electric Power Research Institute (EPRI), and equipment manufacturers (UCA). The UCA is undergoing formal standardization. Manufacturers of intelligent electronic devices (IEDs) are increasingly offering technology that connects to an Ethernet network. The reliability of substation connection Ethernet topologies to link devices are compared in this article. These topologies are used to meet the instrumentation and control (I&C) requirements of a hypothetical substation.

If Ethernet equipment makers create devices with significantly lower Mean Time To Fail than current devices, the same topology with redundant, independent networks will remain the most available for general I&C scenarios. Networks with switches will outperform those with shared hubs in terms of ranking. Networks with switches will outperform those with shared hubs in terms of ranking. All of the systems that use

devices with a longer MTTF have a higher availability than existing systems. We recommend that IED makers design and deliver Ethernet components with larger Mean Times To Fail if Ethernet is to be utilized in critical industrial and substation systems. We propose the Redundant Servers, Routers, and Switches if Ethernet is used for station I&C. Topology of switches We advocate direct fiber for relay-to-relay protection communications. links that are not part of the network.

2.3.8Smart Substation: Art and Future Development [9]

The evolution of the smart grid has a significant impact on the design, building, and operation of substations, which provide access to various power system elements such as transmission lines, transformers, generators (including renewable sources), and loads. Substations are strategically essential because they are a typical location for enabling monitoring, control, and secure operation of the geographically distributed bulk power system. Traditional substations are incapable of meeting the demands of today's complicated power grid, which is characterized by greater size, increased renewable energy consumption, and complex operational conditions. High criteria for dependability, security, interoperability, configurability, controllability, maintainability, flexibility, decreased cost, and environmental impact should all be considered in modern substation design.

The state-of-the-art and future critical technological difficulties of smart substations are presented in this article, which illustrates the substation technology development trend. The most notable distinction is that a smart substation fulfills functionalities through field configuration, whereas traditional ones do so through engineering design. The operation practice revealed that smart substations are more flexible and responsive to changing operating conditions, making them more suitable for modern smart grids with higher renewable energy usage and more complex operational situations. The smart substation concept will become increasingly essential in future substation design and construction as practical challenges are resolved.

2.3.9 Impacts of Automated Control Systems on Substation Reliability

By accelerating service restoration, automation can dramatically improve substation dependability. Even though much research has been done on substation control system reliability, there are still various issues to consider in automated substation reliability evaluation. So far, the effects of automation systems on substation reliability indices have not been thoroughly studied in the literature.

The purpose of this work is to propose a complete method for quantitatively assessing the reliability of automated substations. The method is based on the event tree methodology and conditional probability theory principles. Using a common substation automation architecture known as simple star, a step-by-step evaluation approach is described. The proposed technique is tested on certain standard substation layouts to determine its effectiveness as well as the level of reliability improvement gained through automation. Furthermore, sensitivity assessments for both automatic switching action and a specific automated substation have been performed. The proposed technique for automated substation reliability with arbitrary automation architectures is simple to implement and analyze. Furthermore, automated substation reliability assessment tools can be created in great detail and tailored to the unique needs of substation design.

This study describes a method for assessing the quantitative reliability of automated substations. The proposed method is both informative and comprehensive, and it can help designers improve the reliability of automated substations. This method is broad and can be used to assess the substation's reliability indices in any automation design, including ring and cascade configurations. As a result, the reliability assessment approach presented in this work can quantitatively give a basis for comparison when deciding which type of automation system is acceptable for a specific substation.

This paper describes the effect of an automated control system on substation reliability. By reading this paper we realize the various effect o the automation system and try to implement our project to remove the various limitations of an automated system.

2.4 Chapter Summary

In this chapter, we describe various substation load-controlling and monitoring research papers. We try to present many methods which are used or proposed to use in substation load control and monitoring. This research paper helps us to complete our project successfully. We learned various knowledge advantages and disadvantages in a world of all types of things related to our project by studying research papers.

CHAPTER 3

DETAILS OF EQUIPMENT

3.1 Introduction

To implement the project we need various equipment with fixed ratings. If there is any mismatch in the rating of the equipment we cannot find the desirable outcome. So we discussed various equipment and their specification which we used in our project.

3.2 Equipment Description

The equipment and their specification used in our project are described below-

3.2.1 ESP 8266

3.2.2 Overview

Node MCU is an open-source Lua-based firmware and development board designed specifically for Internet of Things (IoT) applications. It comprises firmware that operates on Express if Systems' ESP8266 and hardware based on the ESP-12 module. As a Wi-Fi module, the ESP 8266mod is used. It has a microcontroller attached to it. It is primarily used for IoT development (Internet of Things). Like a Pocket router, this WiFi module sends all of our base station's signals. All of the information must be kept in a standard database. As a result, we used Firebase as our database. All input data or signals must first pass via the Wi-Fi module. This information is then saved in Firebase for real-time monitoring. Our mobile phone is immediately connected to this firebase.

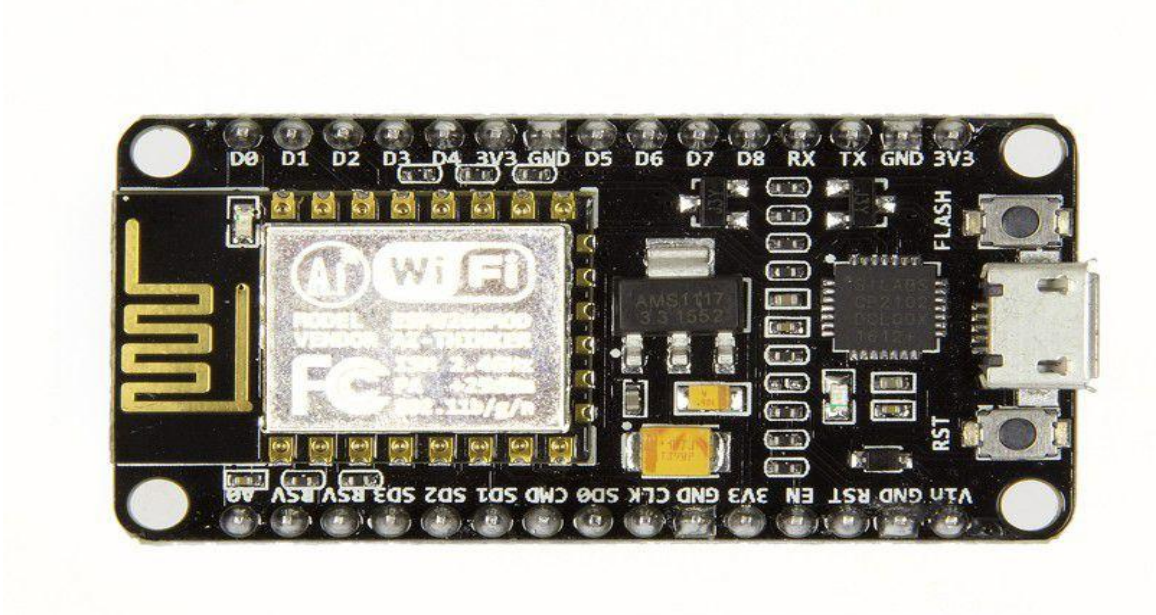


Figure 3.1: ESP 8266front view

3.2.1.2 Power

The Arduino can be powered using the USB association or with an outside control supply. The power source is chosen automatically. Outside (non-USB) power can come either from an AC-to -DC connector. The Barrel jack, or DC Power Jack can be used to power your Arduino board. The barrel jack is usually connected to a wall adapter. The board can be powered by 5-20 volts but the manufacturer recommends keeping it between 7-12 volts. Above 12 volts, the regulators might overheat, and below 7 volts, might not suffice. The power pins are:

i. **VIN**: This is the input voltage pin of the Arduino board used to provide input supply from an external power source. The voltage should be within the range mentioned above.

ii. **5V**: This pin of the Arduino board is used as a regulated power supply voltage and it is used to give supply to the board as well as onboard components.

iii. **3.3V**: This pin of the board is used to provide a supply of 3.3V which is generated from a voltage regulator on the board

iv. **GND:** In the Arduino Uno pin out, you can find 5 GND pins, which are all interconnected. The GND pins are used to close the electrical circuit and provide a common logic reference level throughout your circuit. Always make sure that all GNDs (of the Arduino, peripherals, and components) are connected and have a common ground.

3.2.1.3 Arduino Uno Pin out - Analog IN

The Arduino Uno has 6 **analog pins**, which utilize ADC (Analog to Digital converter). These pins serve as analog inputs but can also function as digital inputs or digital outputs.

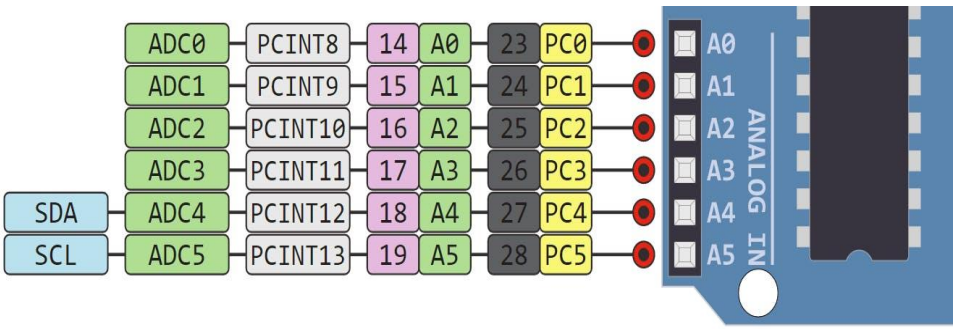


Figure 3.2: Arduino UNO pin out-Analog

3.2.1.4 Arduino Uno pin out - Digital Pins

Pins 0-13 of the Arduino Uno serve as digital input/output pins.

Pin 13 of the Arduino Uno is connected to the built-in LED.

In the Arduino Uno - pins 3,5,6,9,10,11 have PWM capability.

It's important to note that:

- Each pin can provide/sink up to 40 mA max. But the recommended current is 20 mA.
- The absolute max current provided (or sank) from all pins together is 200mA

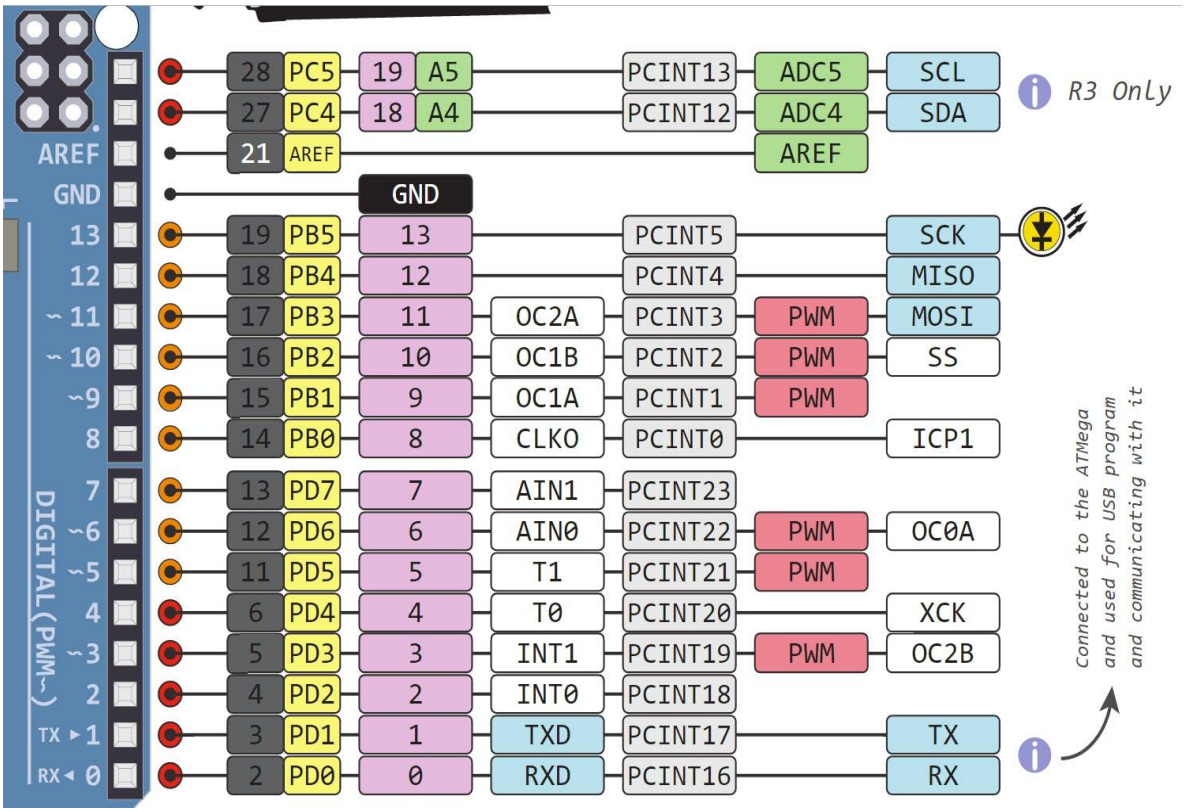


Figure 3.3: Arduino Uno pin out - Digital Pins

3.2.1.5 Others Pins

i. **Pins:** These pins are also known as a UART pins. It is used for communication between the Arduino board and a computer or other devices. The transmitter PIN 1 and receiver PIN 0 are used to transmit and receive the data resp.
ii. **External Interrupt Pins:** This pin of the Arduino board is used to produce the External interrupt and it is done by PINs 2 and 3.
iii. **PWM Pins:** The serial pins of the board are used to convert the digital signal into an analog by varying the width of the Pulse. The PINs 3,5,6,9,10 and 11 are used as a PWM pin.
iv. **SPI Pins:** This is the Serial Peripheral Interface pin, it is used to maintain SPI communication with the help of the SPI library. SPI pins include:
v. **SS:** Pin 10 is used as a Slave Select
vi. **MOSI:** Pin 11 is used as a Master Out Slave In
vii. **MISO:** Pin 12 is used as a Master In Slave Out
viii. **SCK:** Pin 13 is used as a Serial Clock
ix. **LED Pin:** The board has an inbuilt LED using digital pin-13. The LED glows only when the digital pin becomes high.
x. **AREF Pin:** This is an analog reference pin of the Arduino board. It is used to provide a reference voltage from an external power supply.

3.2.1.6Communication Protocols

Serial (TTL) - Digital pins 0 and 1 are the serial pins of the Arduino Uno. They are used by the onboard USB module. Serial communication is used to exchange data between the Arduino board and other serial devices such as computers, displays, sensors, and more. Each Arduino board has at least one serial port. Serial communication occurs on digital pins 0 (RX) and 1 (TX) as well as via USB. Arduino supports serial communication through digital pins with the Software Serial Library as well. This allows the user to connect multiple serial-enabled devices and leave the main serial port available for the USB. Software serial and hardware serial - Most microcontrollers have hardware designed to communicate with other serial devices. Software serial ports use a pin-change interrupt system to communicate. There is a built-in library for Software Serial communication.

SPI - SS/SCK/MISO/MOSI pins are the dedicated pins for SPI communication. They can be found on digital pins 10-13 of the Arduino Uno and the ICSP headers. Serial

Peripheral Interface (SPI) is a serial data protocol used by microcontrollers to communicate with one or more external devices in a bus-like connection. The SPI can also be used to connect 2 microcontrollers. On the SPI bus, there is always one device that is denoted as a Master device and all the rest as Slaves. In most cases, the microcontroller is the Master device. The SS (Slave Select) pin determines which device the Master is currently communicating with.

3.2.1.7 Memory

Microcontroller	Family	Architecture	Flash	SRAM	EEPEROM
ESP 8266	AVR	Harvard	4 mb	64kb	1kb

Table 3.1 Memory of Arduino Uno

3.2.2 4 Relay Module

A power relay module is an electrical switch that is operated by an electromagnet. The electromagnet is activated by a separate low-power signal from a microcontroller. When activated, the electromagnet pulls to either open or close an electrical circuit.

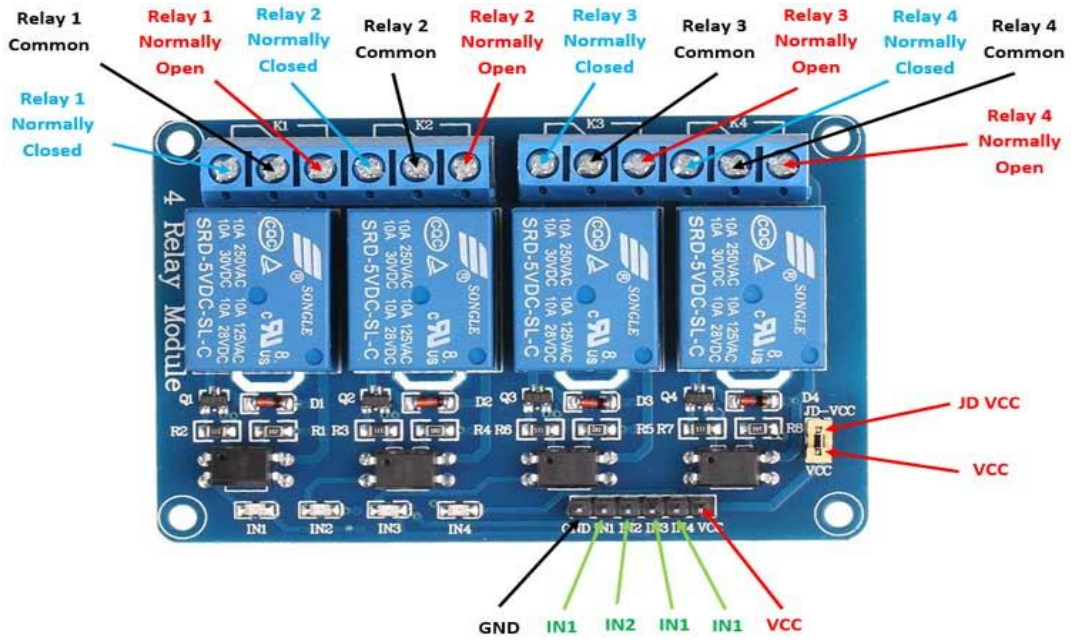


Figure 3.4: 4-channel relay module

3.2.3.1 Description

The 4 Channel Relay Module is a convenient board that can be used to control high voltage, high current loads such as motors, solenoid valves, lamps, and AC loads. It is designed to interface with microcontrollers such as Arduino, PIC, etc. The relay terminal (COM, NO, and NC) is being brought out with a screw terminal. It also comes with a LED to indicate the status of the relay.

3.2.3.2 Specification:

i. Digital output controllable
ii. Compatible with any 5V microcontroller such as Arduino.
iii. Rated through-current: 10A (NO) 5A (NC)
iv. Control signal: TTL level
v. Max. switching voltage 250VAC/30VDC
vi. Max. switching current 10A
vii. Size: 76mm x 56mm x 17mm

3.2.3.3 Schematic diagram of 4 relay module

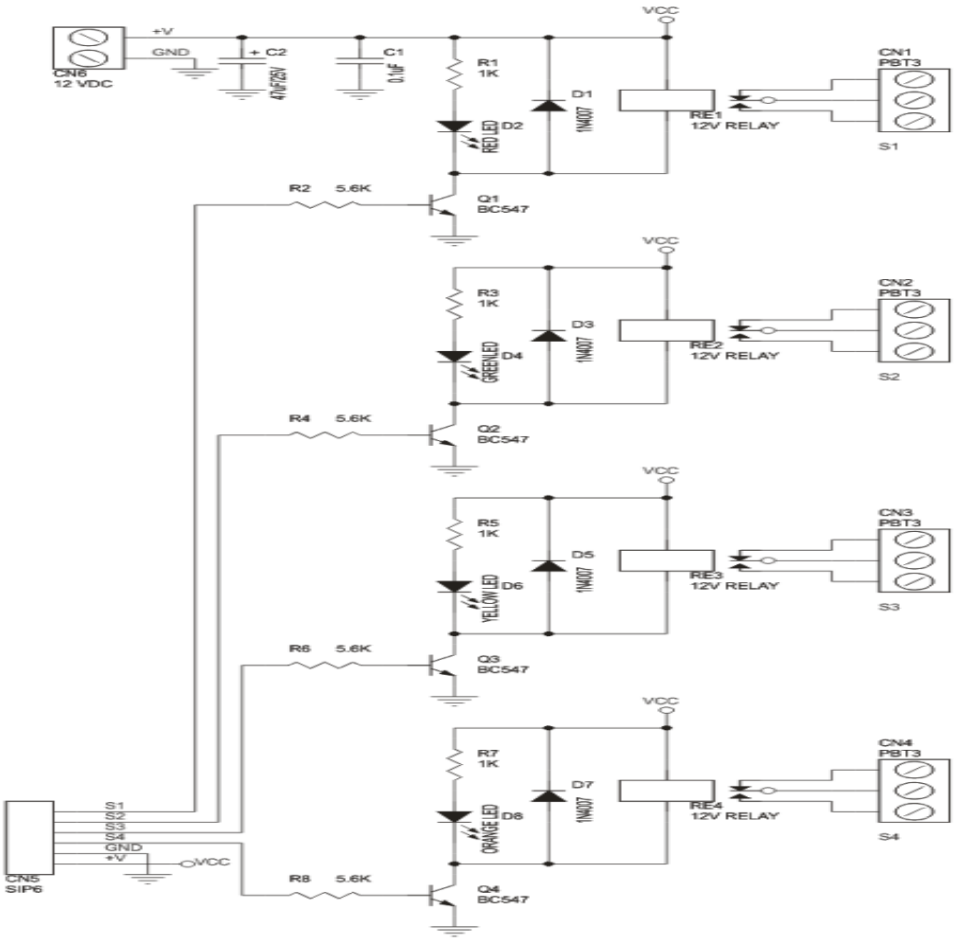


Figure 3.5: Schematic diagram of 4-channel relay module

3.2.4 LCD Display(20*4)

3.2.4.1Description

A 20x4 LCD display is a very basic module and is very commonly used in various devices and circuits. These modules are preferred over seven segments and other multi-segment LEDs. The reasons are: LCDs are economical; easily programmable; have no limitation of displaying special & even custom characters (unlike in seven segments), animations, and so on.

A 20x4 LCD means it can display 20 characters per line and there are 4 such lines. In this LCD each character is displayed in a 5x7 pixel matrix. This LCD has two registers,

namely, Command and Data. This is a standard HD44780 controller LCD. There is no change code for interfacing standard 16x2 or 20x4 LCD.

3.2.4.2 Features

i. 20 Characters x 4 Lines
ii. Built-in HD44780 Equivalent LCD Controller
iii. Works directly with ATMEGA, ARDUINO, PIC, and many other microcontrollers/kits.
iv. or 8-bit data I/O interface
v. Low power consumption



Figure 3.6: 20*4 LCD display

vi. Single LED backlight with a resistor included, you can power it directly from 5V. If it's too bright for you, it can be dimmed easily with a resistor or PWM and uses much less power than LCD with EL (electroluminescent) backlights.
vii. Can be fully controlled with only 6 digital lines!
viii. Built-in character set supports English/Japanese text, see the HD44780 datasheet for the full character set

3.2.2.1 Pin diagram of LCD display

A liquid-crystal display (LCD) is a flat panel display, electronic visual display, or video display that uses the light-modulating properties of liquid crystals. Liquid crystals do not emit light directly. Here, in this able we're going to use a monochromatic 20x4 alphanumeric LCD. 20x4 means that 20 characters can be displayed in each of the 4 rows of the 20x4 LCD, thus a total of 80 characters can be displayed at any instance of time.

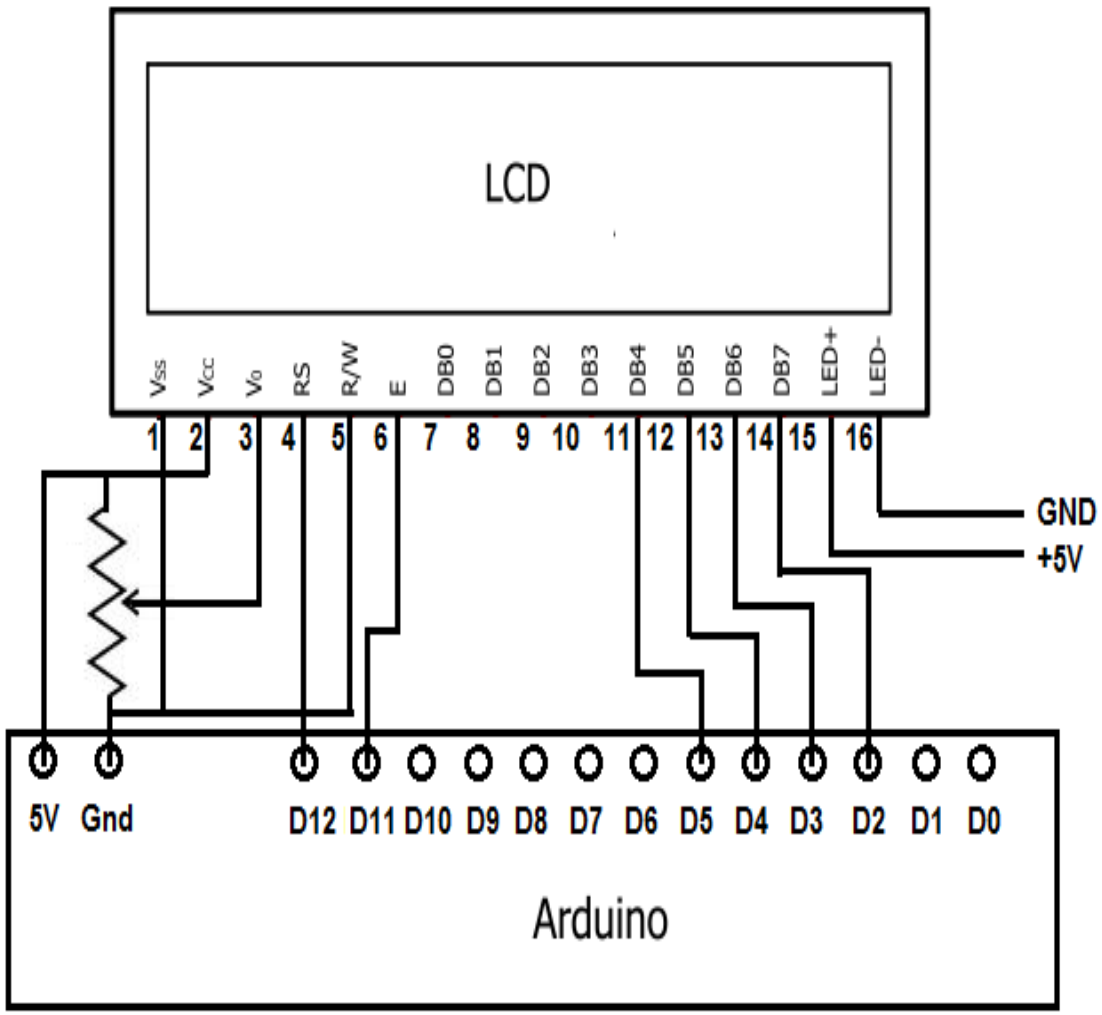


Figure 3.: Pin diagram of 20*4 LCD display

Pin#1	It denoted as Vss	It is ground pinout potential at this pinout is zero.
Pin#2	It denoted as Vdd	At this pinout, five volts are provided.
Pin#3	This pinout denoted as Vo	This pinout is used to set the contrast of the screen.
Pin#4	This pin denoted as RS	It used to H/L register select signal.
Pin#5	It denoted as R/W	It used for H/L read/write signal.
Pin#6	This pinout denoted as E	It used for H/L enable signal.
Pin#7-14	The pinouts from seven to fourteen denoted as DB0 – DB7.	It used for H/L data bus for 4 bit or 8-bit mode.
Pin#15	It identified as A (LED+)	It used to set backlight anode.
Pin#16	It recognized as K (LED-).	It used to set backlight cathode.

Table 3.2: Pin description of LCD

3.2.2.2 various parameters of LCD Display

Parameters	Symbol	Conditions
Input Voltage	It denoted as VDD	The value of VDD is plus five volts.
Supply Current	It denoted as IDD	Its value is ten milliamperes.
LC Driving Voltage for Normal Temperature Version Module	Its symbol is VDD to V0.	Its value is 5.2 volts
LED Forward Voltage	It is denoted as VF.	Its value is 4.3V
LED Forward Current	It denoted as IF.	Its value is 4.6V.
EL Supply Current	This pinout denoted as EL	VEL = 110 VAC, and four hundred frequency.

Table 3.2: various parameters of LCD display

3.2.3 Current Transformer

A current transformer (CT) is a type of transformer that is used to measure AC. It produces an alternating current (AC) in its secondary which is proportional to the AC in its primary. Current transformers, along with voltage or potential transformers are Instrument transformers. Current transformers are designed to provide a scaled-down replica of the current in the HV line and isolate the measuring instruments, meters, relays, etc., from the high-voltage power circuit.



Figure 3.8: Current transformer

The primary and secondary currents of the current transformers are proportional to each other. The current transformer is used for measuring the high voltage current because of the difficulty of inadequate insulation in the meter itself. The current transformer is used in meters for measuring the current up to 100 amperes.

3.2.4 Buck Module

Step-down converters and voltage regulators both regulate voltage through an electrical circuit. Connecting to varied power sources, or changing the devices on a circuit, can change how much current a circuit draws. If a circuit draws more or less power than a device should run with, many undesirable outcomes could happen. The circuit could be damaged, the device could fail to run, or the hardware could crash.



Figure 3.9:Buck module

Step-down converters and voltage regulators are vital to the safe operation of electronics. Without them, machines can experience catastrophic failures or cause fires. An integrated circuit is an important part of step-down converters and voltage regulators. Modern voltage regulators are only possible because of semiconductor-based integrated circuits. The integrated circuit monitors current through an electronic device with a feedback loop. In both components, an integrated circuit is linked to a capacitor, which keeps electrical fluctuations contained. This capacitor can also provide extra power during unexpected shortages. Astep-down converter is a subclass of voltage regulators, which is why there is so much overlap between the two. Some types of step-down converters can be run bidirectional, making them step-up converters as well. This functionality emphasizes the common abilities between step-down converters and voltage regulators.

3.2.5 Current Sensor

A current sensor is a device that detects and converts current to an easily measurable output voltage, which is proportional to the current through the measured path. There are a wide variety of sensors, and each sensor is suitable for a specific current range and environmental conditions. Among these sensors, a current-sensing resistor is the most commonly used. It can be considered a current-to-voltage converter, where inserting a resistor into the current path, the current is linearly converted to a voltage.

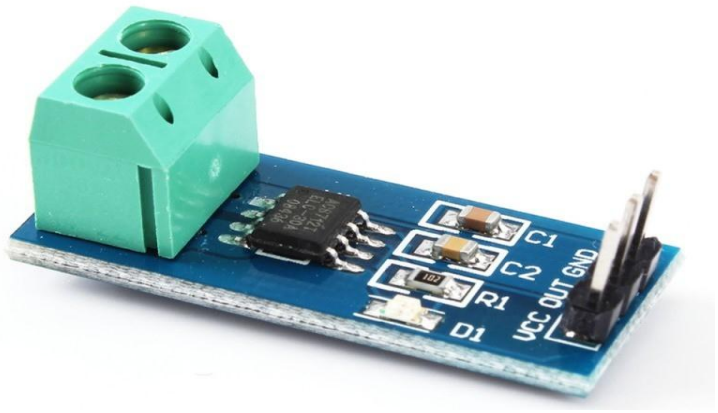


Figure 3.10: Current sensor

The technology used by the current sensor is important because different sensors can have different characteristics for a variety of applications. Current sensors are based on either open or closed-loop hall effect technology. A closed-loop sensor has a coil that is actively driven to produce a magnetic field that opposes the field produced by the current being sensed. The hall sensor is used as a null-detecting device, and the output signal is proportional to the current being driven into the coil, which is proportional to the current being measured. In an open loop current sensor, the magnetic flux created by the primary current is concentrated in a magnetic circuit and measured using a hall device. The output from the hall device is the signal conditioned to provide an exact (instantaneous) representation of the primary current.

3.2.6 Optocoupler

An opt coupler or optoelectronic coupler is an electronic component that acts as an **interface** between two separate circuits with **different voltage levels**. Opt couplers are common components by which electrical isolation can be supplied between the input and output source. It is a **6-pin device** and can have any number of photo detectors.

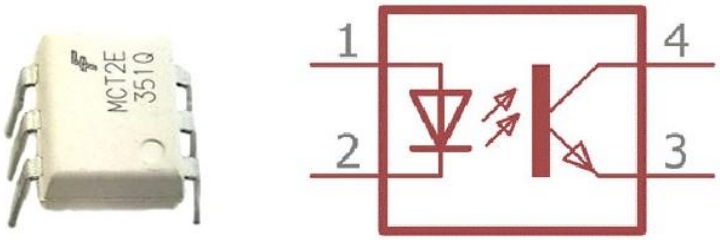


Figure 3.11:Opt coupler

Here, a beam of light emitted by a light source exists as the only contact between input and output. Due to this, we can have an **insulation resistance** of mega ohms between the two circuits. In high-voltage applications where the voltage difference between the

two circuits differs by several thousand volts, such isolation is favorable. The use of all such electronic isolators lies in all the conditions where the signal is to pass between two isolated circuits. Relays usually are driven using opt couplers, which offer isolation between analog and digital grounds to ensure that switching currents in the relay coils do not affect the noise margin of the digital circuit. For driving a large number of relays through a bus-based backplane, the onboard complex programmable logic device (CPLD)/FPGA provides the interfacing and diagnostics to the I/O board. A ULN2803 Darlington transistor with a base current controlled by the opt coupler drives the relay. To minimize power for CPLDs, their outputs are connected to the cathode of the opt coupler, while the anode is connected to either +5 V through a current-limiting resistor.

CHAPTER 4

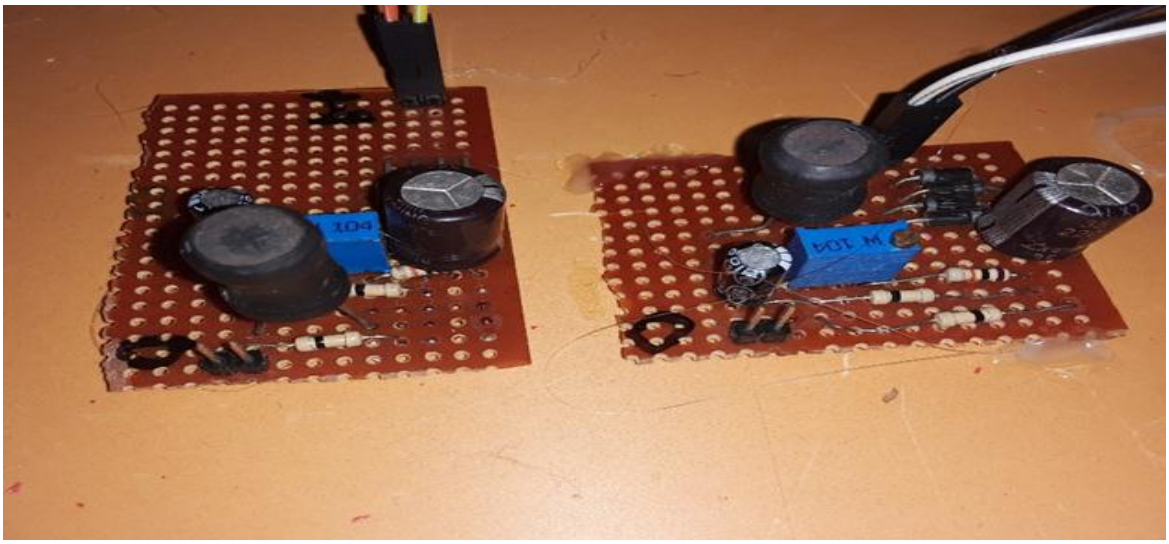
HARDWARE IMPLEMENTATION

4.1 Introduction

This chapter is about our hardware implementation which is the kernel part of our project. Without proper implementation, the project would not be possible to do. So in this chapter, we describe how we implement our hardware and how successful our project is.

4.2 Interfacing Current Sensor With Arduino

We know Arduino cannot measure current. so the output of the current transformer cannot be measured by Arduino directly, this current would be converted to voltage by the Hall effect current transformer and then applied to Arduino. So it is very important to interface the current sensor with Arduino.



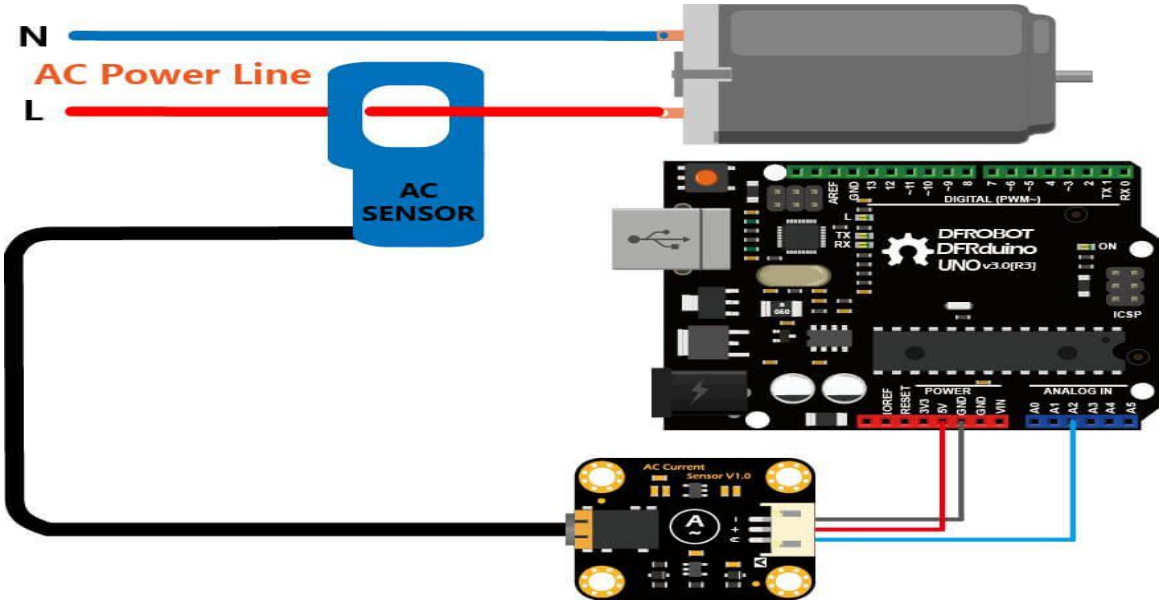


Figure 4.1Interfacing Current Sensor

Arduino una interface with acs712 current sensor circuit diagram is given above. The circuit is universal and the above three codes for different versions of the acs712 current sensor can work with the same circuit shown below. All you need to do is insert the acs712 current sensor into the circuit and upload its corresponding code (given in the post) to the Arduino una, The rest of the connections are the same for all sensors with different acs712 rated currents.

4.3 Flow Chart Of The Project

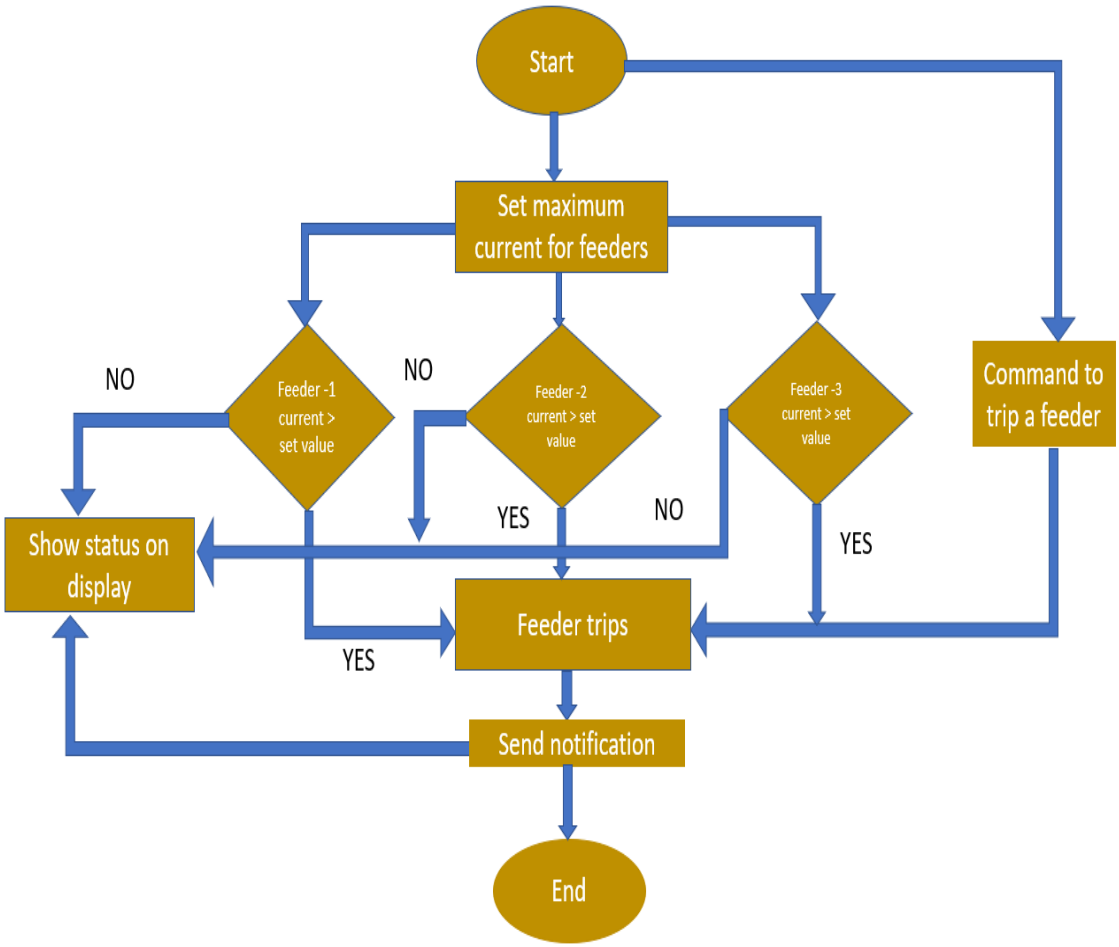


Figure 4.2: Flowchart of the project

From the flow chart, we can see any of the three feeders are overloaded or faulted feeder tripped and GSM activated.GSM then sends an SMS to the mobile phone "OVERLOADED". Otherwise, the display always shows the normal value of the load.

4.4 Block Diagram of the Project

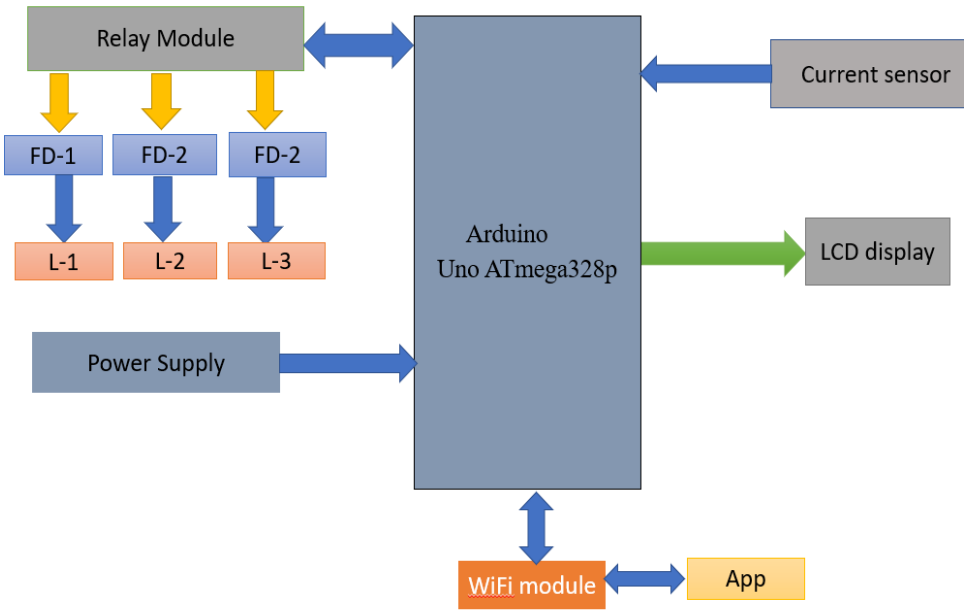


Figure4.3: Block diagram of the project

The above figure shows us the block diagram of our project. In this diagram, we can see various parts of our project. They are Arduino ate mega 328p,LCD display, relay module power supply unit GSM module keypad. By using a keypad we can set the highest load of the three feeders. Whenthe load of any feeder exceeds the predetermined value the Relay module GSM module is activated according to the instruction given to the Arduino ate mega 328p.So the feeder is tripped quickly and GSM sends the SMS of the feeder tripped to the particular number. If we send a particular Fixed format sms is sent to the particular number from a particular number we can turn off any feeder as we wish. This is a short description of our project.

4.5 Circuit Diagram of Our Project

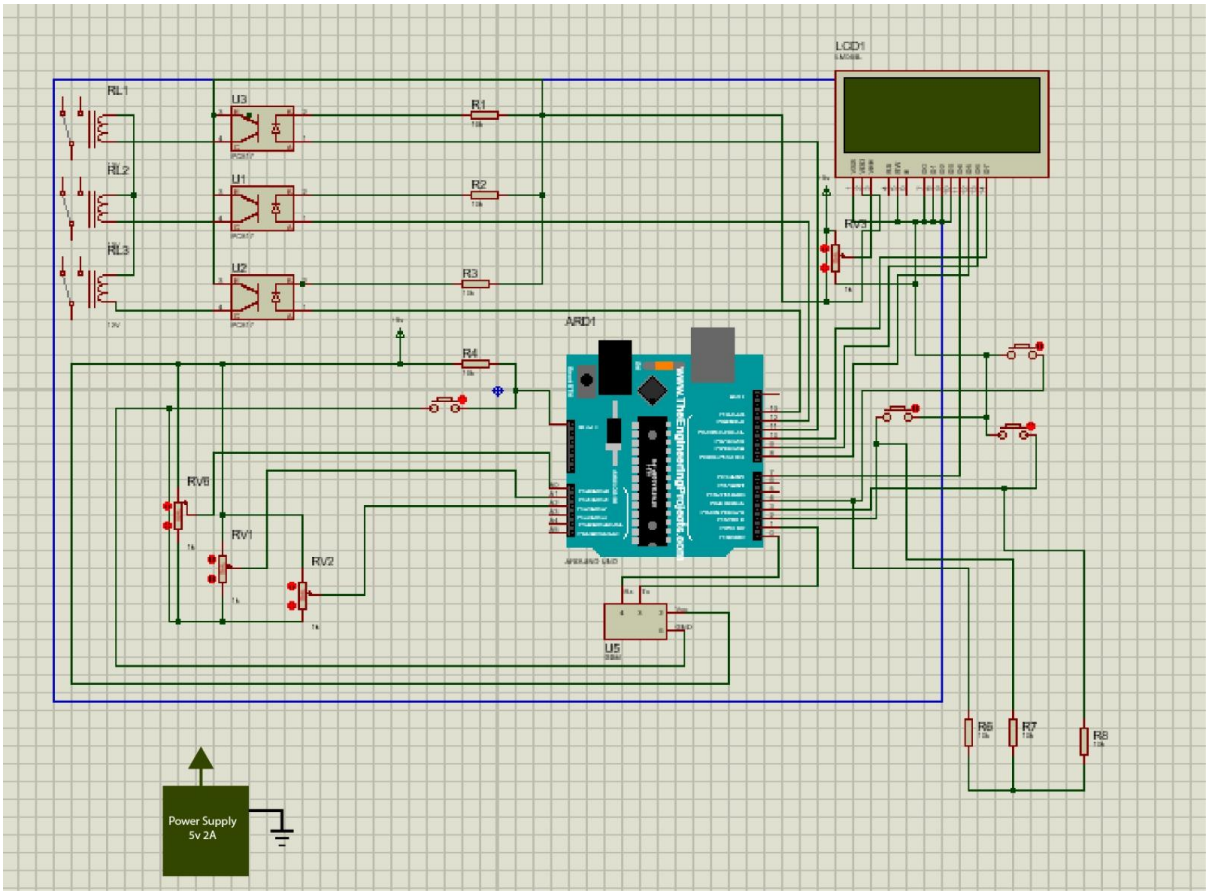


Figure4.4: Circuit diagram of the project

In the circuit AC 22ov of load is converted to 24v dc and then converted to 5 and 4.7v dc.5v dc is applied to Arduino and 4.7v dc is given to GSM. LCD takes power from an Arduino 5v port. The relay module also operates DC 5v.when the current excess limits the highest value current sensor senses and then the Arduino sends a signal to the relay and the relay operates, Arduino also sends SMS to mobile numbers with the help of the GSM module.

4.6 App development using MIT app inventor

The Massachusetts Institute of Technology maintains MIT App Inventor, a web application integrated development environment that was previously offered by Google and is now maintained by the Massachusetts Institute of Technology (MIT).It enables novice programmers to construct application software (apps) for two operating systems (OS): Android and iOS, which is currently in final beta testing as of July 8, 2019.It's free and open-source software with a Creative Commons Attribution Share Alike 3.0 Unsorted license for the source code and an Apache License 2.0 for the rest. It uses a graphical user interface (GUI) similar to the programming languages Scratch (programming language) and Star Logo, which allows users to drag and drop visual objects to create an app that can run on Android devices, while an App-Inventor Companion (The program that allows the app to run and debug on) for iOS devices is still in the works.

App Inventor utilizes a variety of programming languages. The "App Inventor Language" is a new programming language created by MIT App Inventor 2 blocks code. It doesn't have a name yet. TAIL = Textual App Inventor Language is a textual version of this language that we're working on. The blocks editor runs in a web browser since it is built on top of the Blackly blocks language infrastructure Google drew on extensive earlier research in educational computing as well as internal Google work on online development environments to create App Inventor.

Many users will find this difficult to comprehend, yet the block code you see on your screen *is* your language's source code. It describes your App Inventor program's syntax tree in the same manner that traditional text programs describe their languages' syntax trees. When you save an.aia file (which is just a.zip file), the blocks are saved in an XML textual format that represents the same nested tree structure as the blocks on your screen. So, if you like to think of text as source code, this XML format is the App Inventor program's source code.

CHAPTER 5

RESULT AND DISCUSSION

5.1 Pictorial view of the final setup

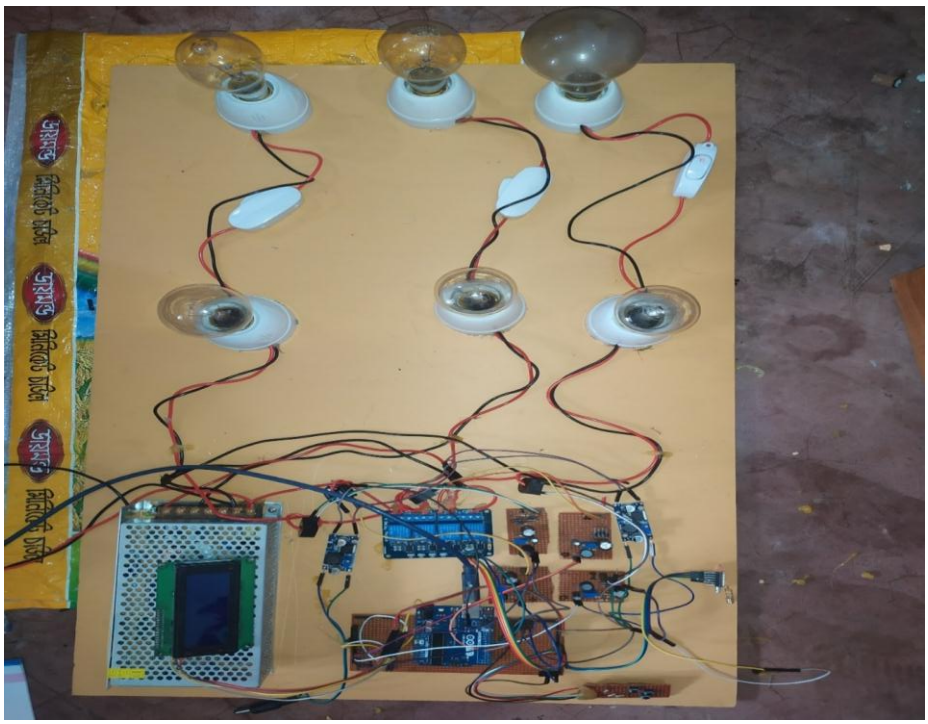


Figure 5.140: Pictorial view of the project

The above figure is our final prototype project. Here overload or high current flow is created by switching off the extra load. When we switched on we saw the specified feeder was tripped and a saw an sms in our mobile phone. Here may be noted that the load was supplied 220v ac directly.

5.2 Data Displaying

5.2.1 When a display is on Initially

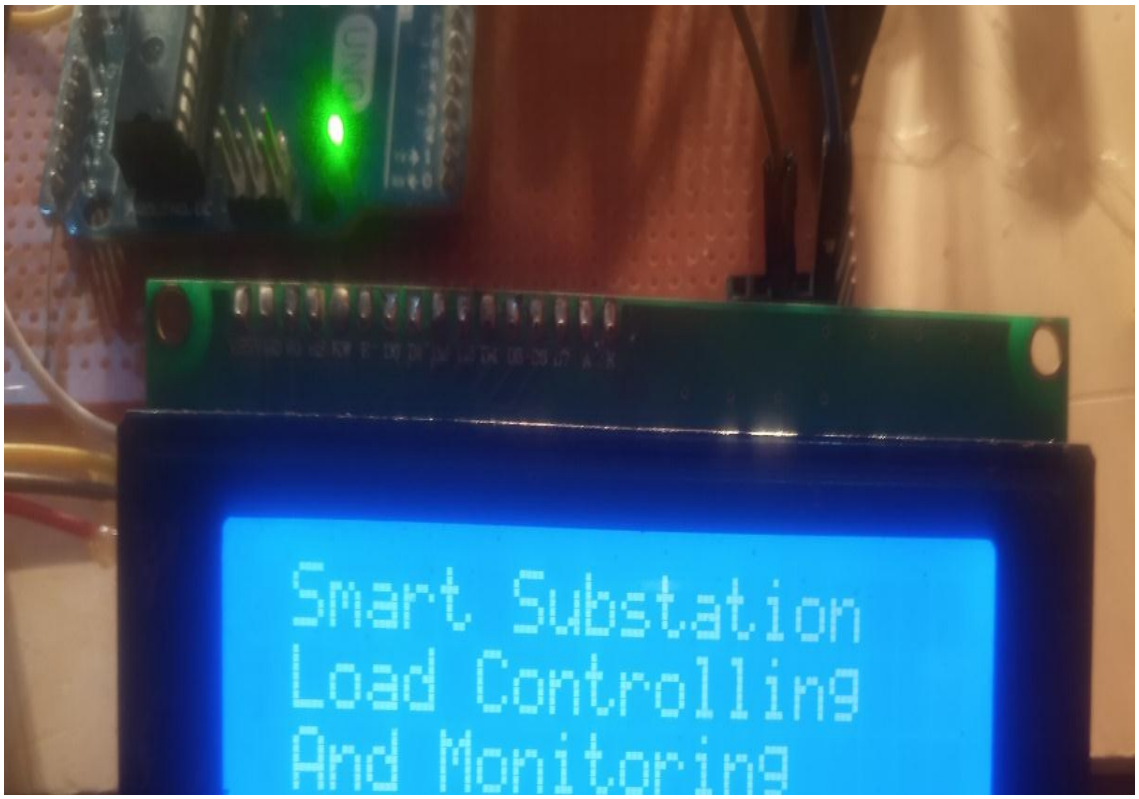
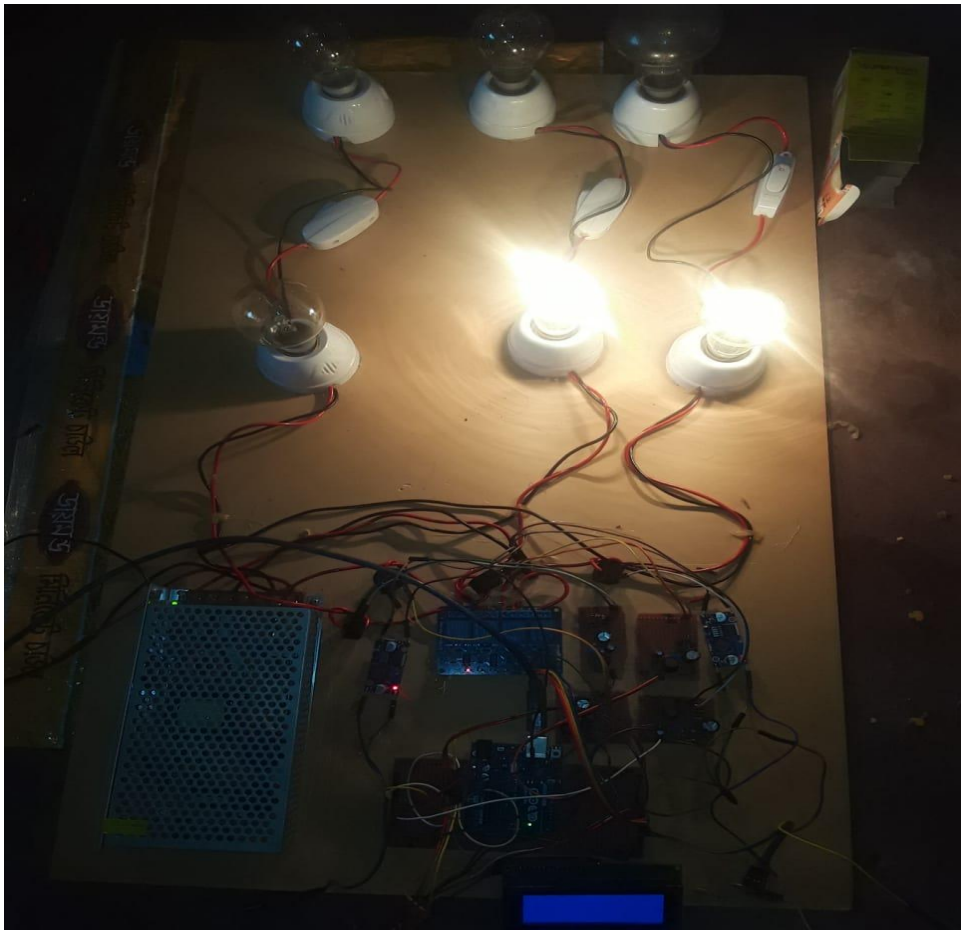


Figure 5.2: Initial status of display

After giving power on display it shows as above figure. We can see the name of our project on the display.

5.2.3 When feeder-1 is Overloaded

5.2.3.1 Feeder 1 is tripped



In this figure, we can see when two light are on in feeder 1 this feeder is overloaded as soon as the feederis tripped but other feedershavea normal load or only one light is ON so this feederis not tripped and consumed normal power.

5.2.3.2 Display at overload condition



Figure 5.4: display status when feeder 1 is overloaded

When feeder 1 is overloaded feeder 1 is tripped and the display shows us that ‘Feeder 1 is over current’

5.2.3.3App interface

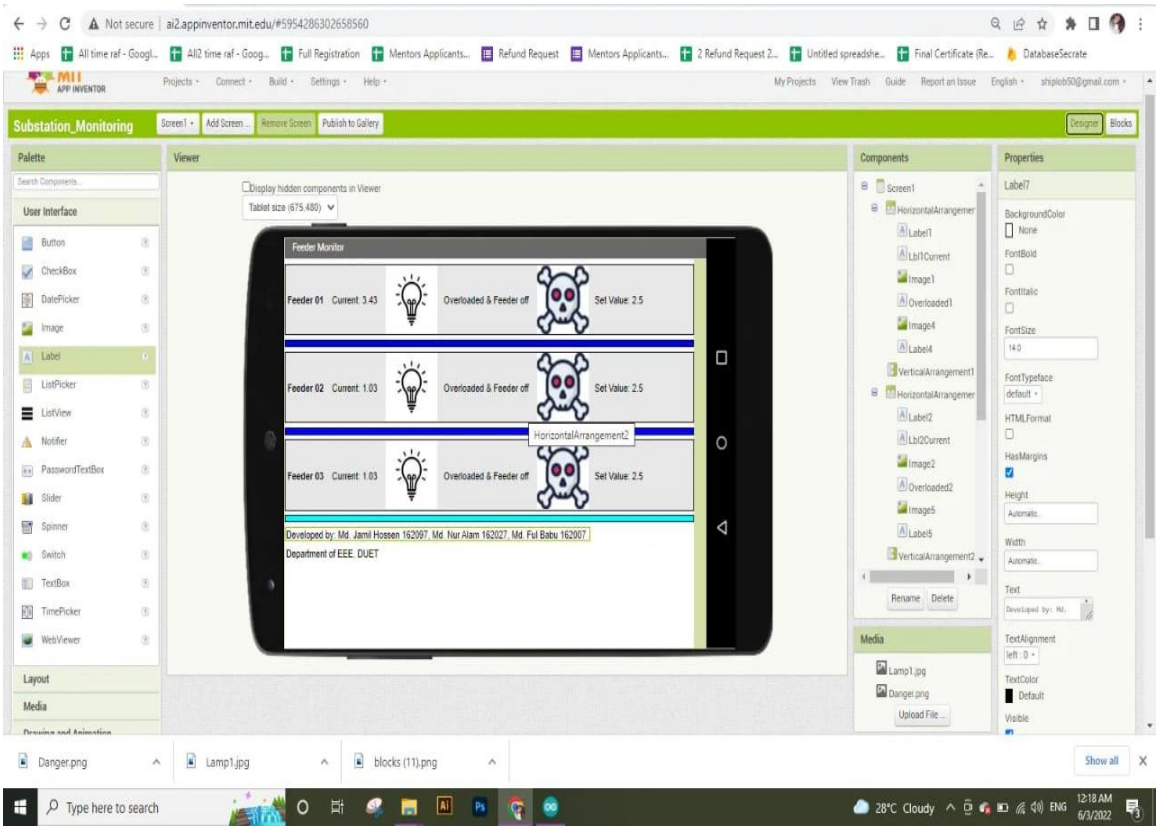


Figure 5.5: App interface

This figure shows us our app interface, In the figure we can see our set value of individual feeders and also see how much current is following in each feeder. This app is developed by us which is shown below in the figure. We develop the app with the help of the MIT app inventor.

5.2.3.4 App output when feeder 1 is overloaded

We can see when feeder 1 current increases the set maximum value feeder 1 is overloaded and tripped. Another feeder is at the normal condition as the current of this feeder is below of set value

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1Conclusion

This project is designed to control and monitor the feeders of the substation through mobile using GSM. The system consists of a GSM modem, current sensors, relay switches, and a microcontroller. When the load current of any feeder is increased due to overload or short circuit feeder trips Arduino sends a signal to GSM send an SMS that the feeder is tripped and the microcontroller is programmed in such a way that if a particular fixed format of SMS is sent to GSM modem from mobile phone, which is fed as input to the microcontroller which operates the appropriate devices. A return feedback message will be sent to the mobile from the GSM modem. So the operator can easily handle the substation from any place and there is no need for any apps it is not a complex system as PLC PLC-based SCADA system so do not need skilled operator analysts and programmers.

6.2 Future Scope

Our Project is not only applicable to substation load control and monitoring rather we can use it to control and monitoring of residential load in the home and any load in the industry. In the future, we can use this project in several applications by adding additional components to this project. This project can be extended by using GPRS technology, which helps in sending monitored and controlled data to any place in the world. In this project, we only use the current sensor to control load current but adding a voltage sensor, and a frequency sensor, would be included to control and monitor voltage, and frequency.

REFERENCES

[1] Win, Khin Thu Zar, and Hla Myo Tun. "Design and implementation of SCADA system-based power distribution for primary substation (control system)." Int. J. Electron. Comput. Sci. Eng 3 (2014): 254-261.

[2] Sachan, A., 2012. Microcontroller based substation monitoring and control system with gsm modem. IOSR Journal of Electrical and Electronics Engineering, 1(6), pp.13-21.

[3] SATHISH, R. and RAMRAJ, R., IOT BASED SUBSTATION MONITORING AND CONTROL.

[4] Raut, Ashwini. "Real time monitoring of substation by using cloud computing." In 2017 Third International Conference on Sensing, Signal Processing and Security (ICSSS), pp. 138-147. IEEE, 2017.

[5] Rajendhar, P., Kumar, P.P. and Venkatesh, R., 2017, June. Zigbee based wireless system for remote supervision and control of a substation. In 2017 International Conference on Innovative Research In Electrical Sciences (IICIRES) (pp. 1-4). IEEE. [6]

[6] Zeynal, Hossein, Mostafa Eidiani, and Dariush Yazdanpanah. "Intelligent substation automation systems for robust operation of smart grids." 2014 IEEE Innovative Smart Grid Technologies-Asia (ISGT ASIA) (2014): 786-790.

[7]Velasquez, Juan Lorenzo, Roberto Villafáfila, Pau Lloret, Lluís Molas, Andreas Sumper, Samuel Galceran, and Antoni Sudrià. "Development and implementation of a condition monitoring system in a substation." In 2007 9th International Conference on Electrical Power Quality and Utilisation, pp. 1-5. IEEE, 2007.

[8] Scheer, Gary W., and David J. Dolezilek. "Comparing the reliability of Ethernet network topologies in substation control and monitoring networks." In Western Power Delivery Automation Conference, Spokane, Washington. 2000.

[9] Huang, Q., Jing, S., Li, J., Cai, D., Wu, J. and Zhen, W., 2016. Smart substation: State of the art and future development. IEEE Transactions on Power Delivery, 32(2), pp.1098-1105.

[10] Hajian-Hoseinabadi, H., 2011. Impacts of automated control systems on substation reliability. IEEE Transactions on Power Delivery, 26(3), pp.1681-1691.

Programmes

```
#include <EEPROM.h>
 #include <Wire.h>
 #include <LiquidCrystal_I2C.h>
 #include <SoftwareSerial.h>
 SoftwareSerial mySerial(3, 2); //SIM800L Tx & Rx is
connected to Arduino #3 & #2
 LiquidCrystal_I2C lcd(0x27,20,4);  // set the LCD address
to 0x27 for a 16 chars and 2 line display
 #define bt_up    8
 #define bt_down  9
 #define bt_Enter 10
 #define Feeder1 7
 #define Feeder2 6
 #define Feeder3 5
 #define buzzer
int sensorPin0 = A0;    // select the input pin for the
potentiometer
int sensorPin1 = A1;
int sensorPin2 = A2;
int ledPin = 13;      // select the pin for the LED
float sensorValue0 = 0;  // variable to store the value
coming from the sensor
float sensorValue1 = 0;
float sensorValue2 = 0;
float Value0 = 0;
float Value1 = 0;
float Value2 = 0;
float d1=8, d2=7, d3=6, d4=5, d5=4, d6=3, d7=2, d8=1;
//int flag1=0, flag2=0, timer=0;
int cnt=0;
int cnt1=0;
void setup() { Serial.begin(9600);   mySerial.begin(9600);

lcd.init();lcd.backlight();lcd.setCursor(3,0);lcd.print("H
ello, world!");lcd.setCursor(2,1);lcd.print("Ywrobot
Arduino!");lcd.setCursor(0,2);lcd.print("Arduino LCM IIC
2004");
  lcd.setCursor(2,3);lcd.print("Power By Ec-
yuan!");delay(100);lcd.clear();
  pinMode(bt_up,  INPUT_PULLUP);
  pinMode(bt_down, INPUT_PULLUP);
  pinMode(bt_Enter, INPUT_PULLUP);
  pinMode(Feeder1, OUTPUT);
  pinMode(Feeder2, OUTPUT);
  pinMode(Feeder3, OUTPUT);
```

```
  pinMode(ledPin, OUTPUT);
 if(EEPROM.read(50)==0){ }else{WriteEeprom();}
EEPROM.write(50,0); ReadEeprom();
}

void loop() {
  while(1)
  { sensorValue0 = analogRead(sensorPin0);
   Value0=sensorValue0*5.0/255.0;
   sensorValue1 = analogRead(sensorPin1);
   Value1=sensorValue1*5.0/1023.0;
   sensorValue2 = analogRead(sensorPin2);
   Value2=sensorValue2*5.0/1023.0;
 //  delay(1000);lcd.clear();
     dispiay();
  // turn the ledPin on
  digitalWrite(ledPin, HIGH);
  // stop the program for <sensorValue> milliseconds:
  delay(Value0);
  // turn the ledPin off:
  digitalWrite(ledPin, LOW);
  // stop the program for for <sensorValue> milliseconds:
  delay(Value0);
  if( Value0>d1){digitalWrite(Feeder1,
HIGH);lcd.clear();feder1();}
  if( Value1>d2){digitalWrite(Feeder2,
HIGH);lcd.clear();feder2();}
  if( Value2>d3){digitalWrite(Feeder3,
HIGH);lcd.clear();feder3();}
  if(digitalRead (bt_Enter)==0)  {set1();}

}
}
void set1()
{lcd.clear();
  while(1)
  { lcd.setCursor (0,0);lcd.print("Set Current-
1");lcd.setCursor(4,1); lcd.print(d1);
    if(digitalRead (bt_up) == 0)   { lcd.clear(); d1=d1+1;
cnt=2;if(d1>99){d1=0;} lcd.setCursor (0,0);lcd.print("Set
Current-1");lcd.setCursor(4,1); lcd.print(d1);
WriteEeprom();delay(200);}
    if(digitalRead (bt_down) == 0) { lcd.clear(); d1=d1-
1;cnt=2;if(d1<0){d1=0;} lcd.setCursor (0,0);lcd.print("Set
Current-1");lcd.setCursor(4,1); lcd.print(d1);
WriteEeprom();delay(200);}
```

```
    if(digitalRead (bt_Enter) ==0 && cnt==2) {cnt=0;
lcd.clear(); lcd.setCursor (0,0);lcd.print("Set Curren-
2");delay(400);set2();}}

}
void set2()
{lcd.clear();
while(1)
{   lcd.setCursor (0,0);lcd.print("Set Current-
2");lcd.setCursor(4,1); lcd.print(d2);
    if(digitalRead (bt_up)   == 0)   { lcd.clear();
d2=d2+1; if(d2>99){d2=0;} lcd.setCursor
(0,0);lcd.print("Set Current-2");lcd.setCursor(4,1);
lcd.print(d2); WriteEeprom();delay(200);}
    if(digitalRead (bt_down) == 0) { lcd.clear(); d2=d2-
1;if(d2<0){d2=0;} lcd.setCursor (0,0);lcd.print("Set
Current-2");lcd.setCursor(4,1); lcd.print(d2);
WriteEeprom();delay(200);}
    if(digitalRead (bt_Enter)==0) { lcd.clear();
lcd.setCursor (0,0);lcd.print("Set Current-
3");delay(400);set3();}}
}
 void set3()
{lcd.clear();
while(1)
{lcd.setCursor (0,0);lcd.print("Set Current-
3");lcd.setCursor(4,1); lcd.print(d3);
    if(digitalRead (bt_up) == 0)   { lcd.clear(); d3=d3+1;
if(d3>99){d3=0;} lcd.setCursor (0,0);lcd.print("Set
Current-3");lcd.setCursor(4,1); lcd.print(d3);
WriteEeprom();delay(200);}
    if(digitalRead (bt_down) == 0) { lcd.clear(); d3=d3-
1;if(d3<0){d3=0;} lcd.setCursor (0,0);lcd.print("Set
Current-3");lcd.setCursor(4,1); lcd.print(d3);
WriteEeprom();delay(200);}
    if(digitalRead (bt_Enter) ==0) {lcd.clear(); loop();}}
}


void dispiay()
{ReadEeprom();
 lcd.setCursor (0,0);lcd.print("Feeder");lcd.setCursor
(7,0);lcd.print("Ampers");lcd.setCursor
(14,0);lcd.print("Set-A");
 lcd.setCursor (0,1);lcd.print("Feder-
1");lcd.setCursor(8,1); lcd.print(Value0);
lcd.setCursor(14,1); lcd.print(d1);
```

```
 lcd.setCursor (0,2);lcd.print("Feder-
2");lcd.setCursor(8,2); lcd.print(Value1);
lcd.setCursor(14,2); lcd.print(d2);
 lcd.setCursor (0,3);lcd.print("Feder-
3");lcd.setCursor(8,3); lcd.print(Value2);
lcd.setCursor(14,3); lcd.print(d3);



}

void feder1()
{ while(1){lcd.setCursor (0,0);lcd.print("Feeder-1 Over
curren"); }}//gsm1();

void feder2()
{ while(1){lcd.setCursor (0,0);lcd.print("Feeder-2 Over
curren"); }}//gsm2();

void feder3()
{ while(1){lcd.setCursor (0,0);lcd.print("Feeder-3 Over
curren");}} //gsm3();



void gsm1()
{
 Serial.println("Initializing...");
 updateSerial();
  mySerial.println("AT"); //Once the handshake test is
successful, it will back to OK
updateSerial();
  mySerial.println("AT+CMGF=1"); // Configuring TEXT mode
  updateSerial();
  mySerial.println("AT+CMGS=\"01672841425\"");//change ZZ
with country code and xxxxxxxxxxx with phone number to sms
   updateSerial();
  mySerial.print("LFeeder-1 OFF"); //text content
   updateSerial();
  mySerial.write(26);
 loop();

}
void gsm2()
{
 Serial.println("Initializing...");
 updateSerial();
```

```
  mySerial.println("AT"); //Once the handshake test is
successful, it will back to OK
updateSerial();
  mySerial.println("AT+CMGF=1"); // Configuring TEXT mode
  updateSerial();
  mySerial.println("AT+CMGS=\"01672841425\"");//change ZZ
with country code and xxxxxxxxxx with phone number to sms
   updateSerial();
  mySerial.print("Feeder-2 OFF"); //text content
   updateSerial();
  mySerial.write(26);
 loop();
}

void gsm3()
{
 Serial.println("Initializing...");
 updateSerial();
  mySerial.println("AT"); //Once the handshake test is
successful, it will back to OK
updateSerial();
  mySerial.println("AT+CMGF=1"); // Configuring TEXT mode
  updateSerial();
  mySerial.println("AT+CMGS=\"01672841425\"");//change ZZ
with country code and xxxxxxxxxx with phone number to sms
   updateSerial();
  mySerial.print("Feeder-3 OFF"); //text content
   updateSerial();
  mySerial.write(26);
  updateSerial();
 loop();
}

void updateSerial()
{
 delay(500);
  while (Serial.available())
  {
    mySerial.write(Serial.read());//Forward what Serial
received to Software Serial Port
  }
  while(mySerial.available())
  {
    Serial.write(mySerial.read());//Forward what Software
Serial received to Serial Port
  }
}
```

```
 void ReadEeprom() {d1=EEPROM.read(1); d2=EEPROM.read(2);
d3=EEPROM.read(3);d4=EEPROM.read(4);d5=EEPROM.read(5);d6=E
EPROM.read(6);d7=EEPROM.read(7);d8=EEPROM.read(8);}
 void WriteEeprom() { EEPROM.write(1, d1);EEPROM.write(2,
d2);EEPROM.write(3, d3); EEPROM.write(4,
d4);EEPROM.write(5, d5);EEPROM.write(6,
d6);EEPROM.write(7, d7);EEPROM.write(8, d8);}
#include <EEPROM.h>
 #include <Wire.h>
 #include <LiquidCrystal_I2C.h>
 #include <SoftwareSerial.h>
 SoftwareSerial mySerial(3, 2); //SIM800L Tx & Rx is
connected to Arduino #3 & #2
 LiquidCrystal_I2C lcd(0x27,20,4);  // set the LCD address
to 0x27 for a 16 chars and 2 line display
 #define bt_up    8
 #define bt_down  9
 #define bt_Enter 10
 #define Feeder1 7
 #define Feeder2 6
 #define Feeder3 5
 #define Total 4

int sensorPin0 = A0;    // select the input pin for the
potentiometer
int sensorPin1 = A1;
int sensorPin2 = A2;
int sensorPin3 = A3;
int ledPin = 13;      // select the pin for the LED
float sensorValue0 = 0;  // variable to store the value
coming from the sensor
float sensorValue1 = 0;
float sensorValue2 = 0;
float sensorValue3 = 0;
float Value0 = 0;
float Value1 = 0;
float Value2 = 0;
float Value3 = 0;
float d1=8, d2=7, d3=6, d4=5, d5=4, d6=3, d7=2, d8=1;
int flag1=0, flag2=0,flag3=0,flag4=0,
flag5=0,flag6=0,flag7=0,flag8=0,flag9=0  ;
int cnt=0,cnt1=0,cnt2=0,cnt3=0,cnt4=0,cnt5=0;
 int one=1;
void setup() { Serial.begin(9600);
   mySerial.begin(9600);
```

```
lcd.init();lcd.backlight();lcd.setCursor(3,0);lcd.print("H
ello, world!");lcd.setCursor(2,1);lcd.print("Ywrobot
Arduino!");lcd.setCursor(0,2);lcd.print("Arduino LCM IIC
2004");
  lcd.setCursor(2,3);lcd.print("Power By Ec-
yuan!");delay(100);lcd.clear();
  pinMode(bt_up,  INPUT_PULLUP);
  pinMode(bt_down, INPUT_PULLUP);
  pinMode(bt_Enter, INPUT_PULLUP);
  pinMode(Feeder1, OUTPUT);
  pinMode(Feeder2, OUTPUT);
  pinMode(Feeder3, OUTPUT);
  pinMode(ledPin, OUTPUT);
 if(EEPROM.read(50)==0){ }else{WriteEeprom();}
EEPROM.write(50,0); ReadEeprom();
}
 void ReadEeprom() {d1=EEPROM.read(1); d2=EEPROM.read(2);
d3=EEPROM.read(3);d4=EEPROM.read(4);d5=EEPROM.read(5);d6=E
EPROM.read(6);d7=EEPROM.read(7);d8=EEPROM.read(8);}
 void WriteEeprom() { EEPROM.write(1, d1);EEPROM.write(2,
d2);EEPROM.write(3, d3); EEPROM.write(4,
d4);EEPROM.write(5, d5);EEPROM.write(6,
d6);EEPROM.write(7, d7);EEPROM.write(8, d8);}

void loop() {
  while(1)
  {   sensorValue0 = analogRead(sensorPin0);
    Value0=sensorValue0*5.0/255.0;
   sensorValue1 = analogRead(sensorPin1);
   Value1=sensorValue1*5.0/1023.0;
   sensorValue2 = analogRead(sensorPin2);
   Value2=sensorValue2*5.0/1023.0;
   sensorValue3 = analogRead(sensorPin3);
   Value3=sensorValue2*5.0/1023.0;

  if( Value3>=d4){digitalWrite(Total,
HIGH);lcd.clear();al1();}
  if( Value3<=d4){digitalWrite(Total,
HIGH);lcd.clear();cnt4=0;}
  if(Value0>d1){digitalWrite(Feeder1,
HIGH);lcd.clear();feder1();}
  if(Value0<d1){digitalWrite(Feeder1,
HIGH);lcd.clear();cnt1=0;}
  if( Value1>d2){digitalWrite(Feeder2,
HIGH);lcd.clear();feder2();}
  if( Value1<d2){digitalWrite(Feeder2,
HIGH);lcd.clear();cnt2=0;}
```

```
  if( Value2>d3){digitalWrite(Feeder3,
HIGH);lcd.clear();feder3();}
  if( Value2<d3){digitalWrite(Feeder3,
HIGH);lcd.clear();cnt3=0;}
 if(digitalRead (bt_Enter) ==0 &&(digitalRead (bt_up) ==
0)){if(flag1==0){ flag1=1;delay(200);total();}}
  else{flag1=0;}
  dispiay();
}
}
void total()
{  lcd.clear();
   while(1)
  { lcd.setCursor  (0,0);lcd.print("Total Current-
1");lcd.setCursor(4,1); lcd.print(d4);
    if(digitalRead (bt_up)== 0)   { lcd.clear();
d4=4+1;if(d4>99){d4=0;}
lcd.setCursor(0,0);lcd.print("Total Current-
1");lcd.setCursor(4,1); lcd.print(d4);
WriteEeprom();delay(200);}
    if(digitalRead (bt_down)== 0) { lcd.clear(); d4=d4-
1;if(d4<0){d4=0;} lcd.setCursor(0,0);lcd.print("Total
Current-1");lcd.setCursor(4,1); lcd.print(d4);
WriteEeprom();delay(200);}
    if(digitalRead (bt_Enter)==0 ) {cnt=0; lcd.clear();
lcd.setCursor (0,0);lcd.print("Set Curren-
1");delay(400);set1();}}
   }
void set1()
{lcd.clear();
  while(1)
  { lcd.setCursor  (0,0);lcd.print("Set Current-
1");lcd.setCursor(4,1); lcd.print(d1);
    if(digitalRead (bt_up) == 0)   { lcd.clear(); d1=d1+1;
if(d1>99){d1=0;} lcd.setCursor (0,0);lcd.print("Set
Current-1");lcd.setCursor(4,1); lcd.print(d1);
WriteEeprom();delay(200);}
    if(digitalRead (bt_down) == 0) { lcd.clear(); d1=d1-
1;if(d1<0){d1=0;} lcd.setCursor (0,0);lcd.print("Set
Current-1");lcd.setCursor(4,1); lcd.print(d1);
WriteEeprom();delay(200);}
    if(digitalRead (bt_Enter) ==0 ) {cnt=0; lcd.clear();
lcd.setCursor (0,0);lcd.print("Set Curren-
2");delay(400);set2();}}
}
void set2()
{lcd.clear();
while(1)
```

```
{   lcd.setCursor (0,0);lcd.print("Set Current-
2");lcd.setCursor(4,1); lcd.print(d2);
    if(digitalRead (bt_up)   == 0)   { lcd.clear();
d2=d2+1; if(d2>99){d2=0;} lcd.setCursor
(0,0);lcd.print("Set Current-2");lcd.setCursor(4,1);
lcd.print(d2); WriteEeprom();delay(200);}
    if(digitalRead (bt_down) == 0) { lcd.clear(); d2=d2-
1;if(d2<0){d2=0;} lcd.setCursor (0,0);lcd.print("Set
Current-2");lcd.setCursor(4,1); lcd.print(d2);
WriteEeprom();delay(200);}
    if(digitalRead (bt_Enter)==0) { lcd.clear();
lcd.setCursor (0,0);lcd.print("Set Current-
3");delay(400);set3();}}
}
 void set3()
{lcd.clear();
while(1)
{   lcd.setCursor (0,0);lcd.print("Set Current-
3");lcd.setCursor(4,1); lcd.print(d3);
    if(digitalRead (bt_up) == 0){lcd.clear(); d3=d3+1;
if(d3>99){d3=0;} lcd.setCursor (0,0);lcd.print("Set
Current-3");lcd.setCursor(4,1); lcd.print(d3);
WriteEeprom();delay(200);}
    if(digitalRead (bt_down) == 0) { lcd.clear(); d3=d3-
1;if(d3<0){d3=0;} lcd.setCursor (0,0);lcd.print("Set
Current-3");lcd.setCursor(4,1); lcd.print(d3);
WriteEeprom();delay(200);}
    if(digitalRead (bt_Enter) ==0) {lcd.clear(); loop();}}
}
void dispiay()
{ReadEeprom();
 lcd.setCursor (0,0);lcd.print("Total");lcd.setCursor
(7,0);lcd.print(Value3);lcd.setCursor
(14,0);lcd.print(d4);
 lcd.setCursor (0,1);lcd.print("Feder-
1");lcd.setCursor(8,1); lcd.print(Value0);
lcd.setCursor(14,1); lcd.print(d1);
 lcd.setCursor (0,2);lcd.print("Feder-
2");lcd.setCursor(8,2); lcd.print(Value1);
lcd.setCursor(14,2); lcd.print(d2);
 lcd.setCursor (0,3);lcd.print("Feder-
3");lcd.setCursor(8,3); lcd.print(Value2);
lcd.setCursor(14,3); lcd.print(d3);
 delay(500);
}
void al1( )
{while(cnt4<one){lcd.setCursor (0,0);lcd.print("Tatol Over
curren");gsm1();loop();}}
```

```
void feder1()
{while(cnt1<one){lcd.setCursor (0,0);lcd.print("Feeder-1
Over curren");gsm2();loop();}}
void feder2()
{while(cnt2<one){lcd.setCursor (0,0);lcd.print("Feeder-2
Over curren"); gsm3(); loop();}}//gsm2();
void feder3()
 {while(cnt3<one){lcd.setCursor (0,0);lcd.print("Feeder-3
Over curren");gsm4();loop();}} //gsm3();
void gsm1()
{while(cnt4<one){
 Serial.println("Initializing...");
  updateSerial();
  mySerial.println("AT"); //Once the handshake test is
successful, it will back to OK
  updateSerial();
  mySerial.println("AT+CMGF=1"); // Configuring TEXT mode
  updateSerial();
  mySerial.println("AT+CMGS=\"01751345652\"");//change ZZ
with country code and xxxxxxxxxxx with phone number to sms
  updateSerial();
  mySerial.print("Main power Overload"); //text content
  updateSerial();
  mySerial.write(26);
  cnt4++;
}}
void gsm2()

{while(cnt1<one){
 Serial.println("Initializing.1");
 updateSerial();
  mySerial.println("AT"); //Once the handshake test is
successful, it will back to OK
 updateSerial();
  mySerial.println("AT+CMGF=1"); // Configuring TEXT mode
  updateSerial();
  mySerial.println("AT+CMGS=\"01672841425\"");//change ZZ
with country code and xxxxxxxxxxx with phone number to sms
   //mySerial.println("AT+CMGS=\"01951856987\"");//change
ZZ with country code and xxxxxxxxxxx with phone number to
sms
   updateSerial();
  mySerial.print("Feeder-1 OFF"); //text content
   updateSerial();
  mySerial.write(26);
  cnt1++;
}}
```

```
void gsm3()
{while(cnt2<one){
 Serial.println("Initializing..2");
 updateSerial();
  mySerial.println("AT"); //Once the handshake test is
successful, it will back to OK
updateSerial();
  mySerial.println("AT+CMGF=1"); // Configuring TEXT mode
  updateSerial();
  mySerial.println("AT+CMGS=\"01672841425\"");//change ZZ
with country code and xxxxxxxxxx with phone number to sms
   updateSerial();
  mySerial.print("Feeder-2 OFF"); //text content
   updateSerial();
  mySerial.write(26);
  updateSerial();
  cnt2++;
}}
void gsm4()
{while(cnt3<one){
 Serial.println("Initializing.3..");
 updateSerial();
  mySerial.println("AT"); //Once the handshake test is
successful, it will back to OK
  updateSerial();
  mySerial.println("AT+CMGF=1"); // Configuring TEXT mode
  updateSerial();
  mySerial.println("AT+CMGS=\"01672841425\"");//change ZZ
with country code and xxxxxxxxxx with phone number to sms
  updateSerial();
  mySerial.print("Feedwe-3 OFF"); //text content
  updateSerial();
  mySerial.write(26);
  updateSerial();
 cnt3++;
}}
void updateSerial()
{
 delay(500);
  while (Serial.available())
  {
    mySerial.write(Serial.read());//Forward what Serial
received to Software Serial Port
  }
  while(mySerial.available())
  {
    Serial.write(mySerial.read());//Forward what Software
Serial received to Serial Port
```

```
  }
}


#include <EEPROM.h>
 #include <Wire.h>
 #include <LiquidCrystal_I2C.h>
 #include <SoftwareSerial.h>
 SoftwareSerial mySerial(3, 2); //SIM800L Tx & Rx is
connected to Arduino #3 & #2
 LiquidCrystal_I2C lcd(0x27,20,4);  // set the LCD address
to 0x27 for a 16 chars and 2 line display
 #define bt_up    8
 #define bt_down  9
 #define bt_Enter 10
 #define Feeder1 6
 #define Feeder2 7
 #define Feeder3 5
 #define Total 4

int sensorPin0 = A0;    // select the input pin for the
potentiometer
int sensorPin1 = A1;
int sensorPin2 = A2;
int sensorPin3 = A3;
int ledPin = 13;      // select the pin for the LED
float sensorValue0 = 0;  // variable to store the value
coming from the sensor
float sensorValue1 = 0;
float sensorValue2 = 0;
float sensorValue3 = 0;
float Value0 = 0;
float Value1 = 0;
float Value2 = 0;
float Value3 = 0;
float totalcurrent=0;
float d1=8, d2=7, d3=6, d4=5, d5=4, d6=3, d7=2, d8=1;
int flag1=0, flag2=0,flag3=0,flag4=0,
flag5=0,flag6=0,flag7=0,flag8=0,flag9=0  ;
int cnt=0,cnt1=0,cnt2=0,cnt3=0,cnt4=0,cnt5=0;
int one=1,x=0,y=0,z=0,m=0;
void setup() { Serial.begin(9600);
   mySerial.begin(9600);

lcd.init();lcd.backlight();lcd.setCursor(1,0);lcd.print("S
mart Substation ");lcd.setCursor(1,1);lcd.print("Load
Controlling");lcd.setCursor(1,2);lcd.print("And
Monitoring");
```

```
  lcd.setCursor(1,3);lcd.print("With SmS
Alarming");delay(5000);lcd.clear();
  pinMode(bt_up,  INPUT_PULLUP);
  pinMode(bt_down, INPUT_PULLUP);
  pinMode(bt_Enter, INPUT_PULLUP);
  pinMode(Feeder1, OUTPUT);
  pinMode(Feeder2, OUTPUT);
  pinMode(Feeder3, OUTPUT);
  pinMode(ledPin, OUTPUT);
  digitalWrite(Total, HIGH);
  digitalWrite(Feeder1, HIGH);
  digitalWrite(Feeder2, HIGH);
  digitalWrite(Feeder3, HIGH);
  digitalWrite(Feeder1, LOW);


 if(EEPROM.read(50)==0){ }else{WriteEeprom();}
EEPROM.write(50,0); ReadEeprom();gsm2();
}
 void ReadEeprom() {d1=EEPROM.read(1); d2=EEPROM.read(2);
d3=EEPROM.read(3);d4=EEPROM.read(4);d5=EEPROM.read(5);d6=E
EPROM.read(6);d7=EEPROM.read(7);d8=EEPROM.read(8);}
 void WriteEeprom() { EEPROM.write(1, d1);EEPROM.write(2,
d2);EEPROM.write(3, d3); EEPROM.write(4,
d4);EEPROM.write(5, d5);EEPROM.write(6,
d6);EEPROM.write(7, d7);EEPROM.write(8, d8);}

void loop() {
  while(1)
  {
    sensorValue0 = analogRead(sensorPin0);
    Value0=sensorValue0*5.0/1023.0;
   sensorValue1 = analogRead(sensorPin1);
   Value1=sensorValue1*5.0/1023.0;
   sensorValue2 = analogRead(sensorPin2);
   Value2=sensorValue2*5.0/1023.0;
   sensorValue3 = analogRead(sensorPin3);
   Value3=sensorValue3*5.0/1023.0;
   totalcurrent=Value0+Value1+Value2+Value3-0.9;
  if( Value3>=d4){digitalWrite(Total,
LOW);lcd.clear();al1();}
  if( Value3<=d4){digitalWrite(Total,
HIGH);lcd.clear();cnt4=0;}
  if(Value0>d1){digitalWrite(Feeder1,
LOW);lcd.clear();feder1();}
  if(Value0<d1){digitalWrite(Feeder1,
HIGH);lcd.clear();cnt1=0;}
```

```
  if(
Value1>d2){digitalWrite(Feeder2,LOW);lcd.clear();feder2();
}
  if( Value1<d2){digitalWrite(Feeder2,
HIGH);lcd.clear();cnt2=0;}
  if( Value2>d3){digitalWrite(Feeder3,
LOW);lcd.clear();feder3();}
  if( Value2<d3){digitalWrite(Feeder3,
HIGH);lcd.clear();cnt3=0;}
 if(digitalRead (bt_Enter) ==0 &&(digitalRead (bt_up) ==
0)){if(flag1==0){ flag1=1;delay(200);total();}}
  else{flag1=0;}
  dispiay();
}
}
void total()
{  lcd.clear();
   while(1)
  { lcd.setCursor  (0,0);lcd.print("Total
Current");lcd.setCursor(4,1); lcd.print(d4);
    if(digitalRead (bt_up)== 0)   { lcd.clear();
d4=4+1;if(d4>99){d4=0;}
lcd.setCursor(0,0);lcd.print("Total Current-
1");lcd.setCursor(4,1); lcd.print(d4);
WriteEeprom();delay(200);}
    if(digitalRead (bt_down)== 0) { lcd.clear(); d4=d4-
1;if(d4<0){d4=0;} lcd.setCursor(0,0);lcd.print("Total
Current-1");lcd.setCursor(4,1); lcd.print(d4);
WriteEeprom();delay(200);}
    if(digitalRead (bt_Enter)==0 ) {cnt=0; lcd.clear();
lcd.setCursor (0,0);lcd.print("Set Curren-
1");delay(400);set1();}}
   }
void set1()
{lcd.clear();
  while(1)
  { lcd.setCursor  (0,0);lcd.print("Set Current-
1");lcd.setCursor(4,1); lcd.print(d1);
    if(digitalRead (bt_up) == 0)   { lcd.clear(); d1=d1+1;
if(d1>99){d1=0;} lcd.setCursor (0,0);lcd.print("Set
Current-1");lcd.setCursor(4,1); lcd.print(d1);
WriteEeprom();delay(200);}
    if(digitalRead (bt_down) == 0) { lcd.clear(); d1=d1-
1;if(d1<0){d1=0;} lcd.setCursor (0,0);lcd.print("Set
Current-1");lcd.setCursor(4,1); lcd.print(d1);
WriteEeprom();delay(200);}
```

```
    if(digitalRead (bt_Enter) ==0 ) {cnt=0; lcd.clear();
lcd.setCursor (0,0);lcd.print("Set Curren-
2");delay(400);set2();}}
}
void set2()
{lcd.clear();
while(1)
{   lcd.setCursor (0,0);lcd.print("Set Current-
2");lcd.setCursor(4,1); lcd.print(d2);
    if(digitalRead (bt_up)   == 0)   { lcd.clear();
d2=d2+1; if(d2>99){d2=0;} lcd.setCursor
(0,0);lcd.print("Set Current-2");lcd.setCursor(4,1);
lcd.print(d2); WriteEeprom();delay(200);}
    if(digitalRead (bt_down) == 0) { lcd.clear(); d2=d2-
1;if(d2<0){d2=0;} lcd.setCursor (0,0);lcd.print("Set
Current-2");lcd.setCursor(4,1); lcd.print(d2);
WriteEeprom();delay(200);}
    if(digitalRead (bt_Enter)==0) { lcd.clear();
lcd.setCursor (0,0);lcd.print("Set Current-
3");delay(400);set3();}}
}
 void set3()
{lcd.clear();
while(1)
{   lcd.setCursor (0,0);lcd.print("Set Current-
3");lcd.setCursor(4,1); lcd.print(d3);
    if(digitalRead (bt_up) == 0){lcd.clear(); d3=d3+1;
if(d3>99){d3=0;} lcd.setCursor (0,0);lcd.print("Set
Current-3");lcd.setCursor(4,1); lcd.print(d3);
WriteEeprom();delay(200);}
    if(digitalRead (bt_down) == 0) { lcd.clear(); d3=d3-
1;if(d3<0){d3=0;} lcd.setCursor (0,0);lcd.print("Set
Current-3");lcd.setCursor(4,1); lcd.print(d3);
WriteEeprom();delay(200);}
    if(digitalRead (bt_Enter) ==0) {lcd.clear(); loop();}}
}
void dispiay()
{ReadEeprom();
 lcd.setCursor (0,0);lcd.print("Total");lcd.setCursor
(7,0);lcd.print(totalcurrent);lcd.setCursor
(14,0);lcd.print(d4);
 lcd.setCursor (0,1);lcd.print("Feder-
1");lcd.setCursor(8,1); lcd.print(Value0);
lcd.setCursor(14,1); lcd.print(d1);
 lcd.setCursor (0,2);lcd.print("Feder-
2");lcd.setCursor(8,2); lcd.print(Value1);
lcd.setCursor(14,2); lcd.print(d2);
```

```
 lcd.setCursor (0,3);lcd.print("Feder-
3");lcd.setCursor(8,3); lcd.print(Value2);
lcd.setCursor(14,3); lcd.print(d3);
 delay(500);
}
void al1( )
{while(cnt4<one){lcd.setCursor (0,0);lcd.print("Tatol Over
current");gsm1();delay(7000);loop();}}
void feder1()
{while(cnt1<one){lcd.setCursor (0,0);lcd.print("Feeder1
Over current");gsm2();delay(7000);loop();}}
void feder2()
{while(cnt2<one){lcd.setCursor (0,0);lcd.print("Feeder2
Over current"); gsm3(); delay(7000);loop();}}//gsm2();
void feder3()
 {while(cnt3<one){lcd.setCursor (0,0);lcd.print("Feeder3
Over current");gsm4();delay(7000);loop();}} //gsm3();
void gsm1()
{while(cnt4<one){
 Serial.println("Initializing...");
  updateSerial();
  mySerial.println("AT"); //Once the handshake test is
successful, it will back to OK
  updateSerial();
  mySerial.println("AT+CMGF=1"); // Configuring TEXT mode
  updateSerial();
  mySerial.println("AT+CMGS=\"01751345652\"");//change ZZ
with country code and xxxxxxxxxxx with phone number to sms
  updateSerial();
  mySerial.print("Main power Overload"); //text content
  updateSerial();
  mySerial.write(26);
  cnt4++;
}}
void gsm2()

{while(cnt1<one){
 Serial.println("Initializing.1");
 updateSerial();
  mySerial.println("AT"); //Once the handshake test is
successful, it will back to OK
 updateSerial();
  mySerial.println("AT+CMGF=1"); // Configuring TEXT mode
  updateSerial();
  //mySerial.println("AT+CMGS=\"01672841425\"");//change
ZZ with country code and xxxxxxxxxxx with phone number to
sms
```

```
  mySerial.println("AT+CMGS=\"01951856987\"");//change ZZ
with country code and xxxxxxxxxxx with phone number to sms
   updateSerial();
  mySerial.print("Feeder-1 OFF"); //text content
   updateSerial();
  mySerial.write(26);
  cnt1++;
}}

void gsm3()
{while(cnt2<one){
 Serial.println("Initializing..2");
 updateSerial();
  mySerial.println("AT"); //Once the handshake test is
successful, it will back to OK
updateSerial();
  mySerial.println("AT+CMGF=1"); // Configuring TEXT mode
  updateSerial();
  mySerial.println("AT+CMGS=\"01672841425\"");//change ZZ
with country code and xxxxxxxxxxx with phone number to sms
   updateSerial();
  mySerial.print("Feeder-2 OFF"); //text content
   updateSerial();
  mySerial.write(26);
  updateSerial();
  cnt2++;
}}
void gsm4()
{while(cnt3<one){
 Serial.println("Initializing.3..");
 updateSerial();
  mySerial.println("AT"); //Once the handshake test is
successful, it will back to OK
  updateSerial();
  mySerial.println("AT+CMGF=1"); // Configuring TEXT mode
  updateSerial();
  mySerial.println("AT+CMGS=\"01672841425\"");//change ZZ
with country code and xxxxxxxxxxx with phone number to sms
  updateSerial();
  mySerial.print("Feedwe-3 OFF"); //text content
  updateSerial();
  mySerial.write(26);
  updateSerial();
 cnt3++;
}}
void updateSerial()
{
 delay(500);
```

```
  while (Serial.available())
  {
    mySerial.write(Serial.read());//Forward what Serial
received to Software Serial Port
  }
  while(mySerial.available())
  {
    Serial.write(mySerial.read());//Forward what Software
Serial received to Serial Port
  }
}
```