INTERNET OF THINGS (IoT) BASED GRID TENSION MONITORING SYSTEM

LINA KHALIDA BINTI NORAZMI

BACHELOR OF ENGINEERING TECHNOLOGY (ELECTRICAL) WITH HONS

UNIVERSITI MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name	:	LINA KHALIDA BINTI NORAZMI
Date of Birth	:	
Title	:	INTERNET OF THINGS (IoT) BASED OF GRID TENSION MONITORING SYSTEM
Academic Session	:	2021/2022

I declare that this thesis is classified as:

☐	CONFIDENTIAL	(Contains confidential information under the Official Secret Act 1997)*
☐	RESTRICTED	(Contains restricted information as specified by the organization where research was done)*
☑	OPEN ACCESS	I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

_______________ (Student's Signature)	_______________ (Supervisor's Signature)
_______________ New IC/Passport Number	DR. NOORAZLIZA SULAIMAN Name of Supervisor
Date: 23/2/2022	Date: 27/2/2022

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

اونيۏرسيتي مليسيا ڤهڠ
UNIVERSITI MALAYSIA PAHANG

SUPERVISOR'S DECLARATION

I hereby declare that I have checked this thesis, and, in my opinion, this thesis is adequate in terms of scope and quality for the award of the degree in BACHELOR OF ENGINEERING TECHNOLOGY (ELECTRICAL) WITH HONS

____

(Supervisor's Signature)

Full Name : DR. NOORAZLIZA BINTI SULAIMAN

Position : SENIOR LECTERUR

Date : 23/2/2022

اونيۏرسيتي مليسيا ڤهڠ
UNIVERSITI MALAYSIA PAHANG

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

____

(Student's Signature)

Full Name : LINA KHALIDA BINTI NORAZMI

ID Number : TB18034

Date : 16 February 2022

INTERNET OF THINGS (IoT) BASED GRID TENSION MONITORING SYSTEM

LINA KHALIDA BINTI NORAZMI

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Bachelor of Engineering Technology in Electrical

Faculty of Electrical & Electronics Engineering Technology

UNIVERSITI MALAYSIA PAHANG

FEBRUARY 2022

ACKNOWLEDGEMENTS

First and foremost, praises and thankful to ALLAH Subhanahu Wa Taala for giving me wisdom, strength, patience, and assistance to complete my project work. Had it not been due to His will and favour, the completion of this study would not have been achievable.

I would like to convey my sincere appreciation to my senior project design supervisor, Dr.Noorazliza binti Sulaiman, for her encouragement and essential advice throughout the project. Her enthusiasm, vision, sincerity, and motivation have profoundly influenced me. She taught me the methods for carrying out the project and presenting it as plainly as practicable. Learning under her direction was a wonderful privilege and honour.

I also like to express my huge thanks to the faculty (FTKEE) for providing the laboratory facilities for this project. My deepest gratitude also goes to all my friends, lecturers, teaching engineers and others who supplied assistances and suggestions, including the vital input for my planning and outcomes. I would like to say huge thank you to my group member Muhammad Saiful Amin bin Mohd Suhaimi on his contribution which very important to finish this project .The advice and assistance obtained from everyone was critical to the success of our project.

I would especially want to take this opportunity to express to my family for their continuous encouragement, both financially and emotionally, during my studies. My sincere thanks go to my family for their encouragement, patience, love, and faith during my studies. Finally, I would like to thank everyone who was directly or indirectly engaged in this study.

ABSTRAK

Objektif projek ini adalah untuk mencipta prototaip sistem pemantauan ketegangan grid untuk talian penghantaran atas. Sistem pemantauan masa nyata untuk suhu talian penghantaran overhed ketegangan grid, paksi sag telah dibangunkan. Memandangkan kemalangan sentiasa dikurangkan dan diagnosis semakin pantas untuk grid automatik, penderiaan dan kawalan jauh masa nyata ialah konfigurasi untuk pelaksanaan grid pintar. Sekiranya berlaku kerosakan, voltan dan arus melebihi had operasi, yang memerlukan pemantauan dalam masa serta pelaporan diagnosis. Projek itu termasuk teknologi Internet of Things, yang menyumbang kepada pemantauan bacaan sensor, serta sistem amaran yang berfungsi sebagai sistem maklum balas, bertindak balas kepada bacaan sensor dengan menghantar makluman kepada pengguna pada nilai paksi dan suhu yang tidak normal. Kaedah projek bergantung pada pemindahan data daripada modul sensor ke mikropengawal, di mana ia direkodkan dan dipaparkan pada tapak web dan aplikasi mudah alih yang boleh diakses oleh pengguna untuk memantau bacaan sensor. Penggunaan kaedah Internet of things dalam sistem pemantauan ini adalah dengan menggunakan web ThingSpeak dan aplikasi Blynk. Menggunakan penerbitan dan memantau pelaksanaan, data dihantar melalui protokol internet. Algoritma mesti dibina untuk mengenal pasti input dan output projek, serta prosedur di antaranya. Penemuan prototaip menunjukkan laman web berfungsi dan aplikasi mudah alih di mana paparan digital memaparkan bacaan dengan tepat mengikut tujuan yang dimaksudkan. Walau bagaimanapun, projek ini tidak mempunyai papan pemuka tambahan dengan bacaan pelbagai fungsi, seperti memantau kualiti kendur dan menghadapi masalah tanpa sokongan bateri bebas, yang memerlukan bekalan voltan langsung.

ABSTRACT

The objective of this project is to create a prototype of grid tension monitoring system for overhead transmission lines. A real-time monitoring system for grid tension overhead transmission line temperature, sag axis has been developed. As accidents are constantly being reduced and diagnoses are accelerating for automated grids, real time remote sensing and control is the configurations for smart grid implementation. In case of malfunction, the voltage and current exceed the operating limit, which requires in-time monitoring as well as diagnosis reporting. The project includes Internet of Things technology, which contributes to the monitoring of sensor readings, as well as an alert system that functions as a feedback system, responding to sensor readings by sending alerts to users on abnormal axis and temperature values. The project method relies on the transfer of data from the sensor module to the microcontroller, where it is recorded and displayed on a website and mobile application that users may access to monitor sensor readings. The Internet of things method use in this monitoring system is by using ThingSpeak web and Blynk application. Using the publish and monitor implementation, data is transmitted over the internet protocol. The algorithm must be constructed to identify the project input and output, as well as the procedures in between. The findings of the prototype showed a functional website and mobile application in which the digital display accurately displays readings in accordance with its intended purpose. However, the project lacks an additional dashboard with multifunctional readings, such as monitoring the quality of the sag and having issues with no independent battery support, which requires direct voltage supply.

TABLE OF CONTENT

DECLARATION

TITLE PAGE

ACKNOWLEDGEMENTS ii

ABSTRAK iii

ABSTRACT iv

TABLE OF CONTENT v

LIST OF TABLES viii

LIST OF FIGURES ix

LIST OF SYMBOLS x

LIST OF ABBREVIATIONS xi

LIST OF APPENDICES xii

CHAPTER 1 INTRODUCTION 1

1.1 Project Background 1

1.2 Problem Statement 2

1.3 Objectives 3

1.4 Project Scope 3

1.5 Thesis Organization 3

CHAPTER 2 LITERATURE REVIEW 5

2.1 Introduction 5

2.2 Internet of Things (IoT) System 5

2.3 Existing Related Projects 6

2.3.1 Internet of Things-Based Smart Electricity Monitoring and Control System Using Usage Data 6

2.3.2 Maintenance of Transmission Line by Using Robot 7

2.3.3 A Romanian Solution for Real-time Monitoring of Overhead Transmission Lines 8

2.3.4 Overhead Transmission Line Monitoring System for Dynamic Rating 9

2.4 Internet of Things (IoT) Tools 11

2.4.1 ThingSpeak 11

2.4.2 Blynk 16

2.4.3 GSM (Global system mobile communication) Module 17

2.5 Summary 18

CHAPTER 3 METHODOLOGY 19

3.1 Introduction 19

3.2 Flowchart of Overall System 19

3.3 Proposed Application Design 21

3.4 System Design 23

3.5 GSM Module 24

3.5.1 AT command used 24

3.6 ThingSpeak 25

3.7 Blynk 26

3.8 Summary 26

CHAPTER 4 RESULTS AND DISCUSSION 27

4.1 Introduction 27

4.2 Interface of grid tension monitoring system 27

4.3 Discussion 30

4.4 Summary 32

CHAPTER 5 CONCLUSION **33**

5.1 Conclusion 33

5.2 Recommendation 34

REFERENCES **35**

APPENDICES **37**

LIST OF TABLES

Table 5.1 Material and cost analysis 42

LIST OF FIGURES

Figure 2.1	Transmission Line Robot	7
Figure 2.2	OHL conductor integrating OHLM-HV module	9
Figure 2.3	Inside structure of data collecting terminal	10
Figure 2.4	Profile after signing up to ThingSpeak account	12
Figure 2.5	Interface to create chanel	14
Figure 2.6	Interface of Chanel	14
Figure 2.7	API key required for coding	15
Figure 2.8	System architecture of Blynk application	16
Figure 3.1	Flowchart of proposed algorithm	20
Figure 3.2	ThingSpeak website	22
Figure 3.3	ThingSpeak and Blynk graphical interface	22
Figure 3.4	System architecture	24
Figure 4.1	Interface monitoring system in Mobile apps via ThingSpeak	28
Figure 4.2	Interface monitoring system in Website via ThingSpeak	28
Figure 4.3	Interface monitoring system in Blynk apps	29
Figure 4.4	Measured Temperature Data from ThingSpeak mobile apps	31
Figure 4.5	Measured Axis Data from ThingSpeak mobile apps	31
Figure 4.6	Measured Temperature and Axis from Blynk mobile apps	32
Figure 5.1	Blynk setup and libraries	37
Figure 5.2	Coding for reading sensors	38
Figure 5.3	Coding for ThingSpeak server	39
Figure 5.4	Coding for transmit to ThingSpeak and Blynk apps	40
Figure 5.5	Hardware design via Fritzing	40
Figure 5.6	Full view of prototype	41
Figure 5.7	Side view of prototype	41

LIST OF SYMBOLS

°C	Degree Celsius
°	Degree for angle
RM	Ringgit Malaysia

LIST OF ABBREVIATIONS

kWh	Kilo Watt Per Hour
IoT	Internet Of Thing
API	Application Programming Interface
ID	Identity Document
iOS	iPhone Operating System
GSM	Global System for Mobile Communication
GPRS	General Packet Radio Service
Rx	Receiver
Tx	Transmitter
UART	Universal Asynchronous Receiver - Transceiver
U.FL	Ultra-Small Surface Mount Coaxial Connectors
SMA	Sub Miniature Version A
PC	Personal Computer
bps	Bits Per Second
TCP/IP	Transmission Control Protocol/Internet Protocol
AT	Attention
SMS	Short Message Service
GUI	Graphical User Interface
DHT	Digital Humidity and Temperature
ADXL	3 Axis Accelerometer Sensors
DC	Direct Current
M2M	Machine To Machine
CCTV	Closed - Circuit Television
kV	Kilo Volt
OHL	Overhead Line
LCD	Liquid Crystal Display
SMACS	Self-Organizing Medium Access Control for Sensor Networks
LTE	Long Term Evolution
RF	Radio Frequency
GPS	Global Positioning System
MMS	Multimedia Message Service

LIST OF APPENDICES

Appendix A: Coding of the system 37

Appendix B: Design 40

Appendix C: Final Prototype 41

Appendix D: Cost of Analysis 42

Appendix E: Gantt Chart Project SDP 2 43

CHAPTER 1

INTRODUCTION

1.1 Project Background

Over the past two decades in Malaysia, the demand for energy has increased considerably from 1406kWh in 1993 to 4110kWh by the person of 2013[1]. Fast economic growth led to rise in industrial energy and power demand[1]. As the result of the next standard of living, more users presently claim and use technology-based energy consuming machines and devices[1]. Modernization and business development have also increased power consumption ,through more facilities, production facilities, shopping malls and other entertainment facilities. Increment on energy consumption will affect the efficiency of the grid. Since the demand of electricity is increasing. Therefore, to maintain the quality of the grid is important. The overhead transmission line addition is a big component to step up grid capacity as top notch to use the available grid with maximum efficiency. As a consideration of human safety and natural surroundings, the aim for this project is to develop Internet of Things (IoT) based grid tension monitoring system. The system consists of clipper, Arduino Uno,DHT22 temperature and humidity sensor,ADXL335 3 axis accelerometer sensor and GSM900. The clamped will be clipped at end to end of the transmission lines. The utilization of human being is minimized by utilizing clamped. The size of clipper is compact and small it is suitable for gripping the grid lines. The clipper also can overcome from any impediments on power lines for this require inflexible mechanical technology having power sources and power sensors.

1.2 Problem Statement

Every year, numerous of workers are killed because of electrocutions caused by overhead and underground power lines. Due to this risk , it is important that a method of maintenance with direct contact to the transmission line are eliminated and replace with the use of sensors. Other than that, the system must also be able to observe the transmission line from a safe distance by allowing authorized personnel to access the readings obtained from the sensor. Maintenance might take many hours, days, or even weeks. This is mostly determined by the ease with which they can access the towers, as well as the scope and intricacy of the repair operation[2]. It is basically dependent on how easily they can access the towers as well as the volume and intricacy of the repair job. In fact, in certain circumstances, access to the towers may be through highways or country roads. The traditional method of creating a schedule of trips to the line on a regular basis is still in use[2]. This entails sending a team of two to three personnel, often electrical technicians, and mechanics, to the tower base. The second technique of inspecting a line includes the use of technology that does not require the line to be taken out of operation[2]. To collect data locally or remotely, technicians employ infrared and ultraviolet light cameras, as well as ultrasound detectors. In the latter arrangement, data is sent to a control centre through cellular networks[2].

Because transmission line today is longer and widely spread, some are in harsh terrain and weather condition. Due to this reason transmission line monitoring are using the combination of human, mobile robot, and helicopter. Each of these methods are expensive and require a large labour force. Furthermore, transmission line conditions cannot be continuously monitored, and all transmission lines cannot be covered at the same time[3].

1.3 Objectives

The objectives of the prototyping of tension power grid monitoring system on overhead transmission line is to monitor the maximum capacity of power lines.

1. To analyse the relationship of power losses with grid tension.

2. To develop grid tension detector to reduce power losses.

3. To develop real-time monitoring system for maintenance purpose.

1.4 Project Scope

The scope of work of this project focuses on designing a dynamic line rating monitoring system that detect the thermal and transmission line sag. This project also focuses on the development of system to monitor transmission line condition in real-time. Finally, the scope of work also includes the development of web monitoring by using IoT that will be designed based on Arduino Uno module and Blynk and ThingSpeak application which is a graphical programming.

1.5 Thesis Organization

Chapter 1 introduces the project background and highlights the issue statements as well as the aims for resolving them. Aside from that, it explains the limitations of the existing product.

The chapter 2 contains research on existing projects that are similar to this Grid Tension Monitoring System. Grid sag, IoT system, and monitoring system technique on grid tension system are all part of this project. This chapter was utilized as a reference to construct this project.

Chapter 3 is describing the method as well as the materials and IoT platform implemented in this project. ThingSpeak and Blynk application specifications are included in the software system. This chapter explains why the content was chosen. The project's design prototype and flowchart are also presented in Chapter 3.

The results of this project were explained in Chapter 4, which shows the final design of the prototype, the system's ability to monitor behaviour movement of sag tension, and the interface of the grid tension monitoring system.

The prototype of grid tension monitoring system's contribution was described in Chapter 5. It also discusses the project's limitations and suggestions for improvement.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Throughout this chapter, the theory of Internet of Things (IoT) and existing related projects are presented. This chapter also go through what is IoT tools such as ThingSpeak web and Blynk application and how the process of interface working. We will go through what ThingSpeak and Blynk apps are as a flow-based Internet of Things tool and how its capabilities have made the creation of Internet of Things (IoT) goods much simpler to do.

2.2 Internet of Things (IoT) System

Internet of Things (IoT) is a modern communication model for the near future in which everyday objects equipped with microcontrollers, digital communication transmitters and receivers, are controlled via appropriate protocols to form a kind of communication network with each other and with users, thereby making it an integral part of the Internet[5]. The Internet of Things (IoT) idea intends to popularize the Internet. Furthermore, by allowing for simple interaction with a broad variety of devices such as home applications, CCTV, surveillance sensors, engines, monitors, cars, and so on[2][7]. Now, this idea is progressively being defined as an ecosystem of networked people and things, rather than only M2M. The Internet of Things is about human and machine communication-based shared services[2].

Due to the obvious wide use of IoT, numerous platforms have been created to connect disparate items over the Internet. Due to a lack of standards such as communication protocols and connectivity, it is challenging to create IoT solutions for various smart items from multiple manufacturers. As a result, a new development paradigm based on the cloud user interface has been developed. There are various platforms designed for business communities, such as Xively, that enable organizations to employ Internet features to securely link their goods with their users, manage their data, increase partner and customer happiness, and generate new income streams[7]. Many more sources, such as ThingSpeak and Blynk, enable users to manage data in various forms and exchange data from other sources.

2.3 Existing Related Projects

2.3.1 Internet of Things-Based Smart Electricity Monitoring and Control System Using Usage Data

This paper discussed the integration of hardware and software. The program is used to monitor power use and consumption of home appliances and control systems through overcurrent relay, as well as to notify of any mismatches[8]. Arduino UNO, a Wi-Fi module (ESP8266), a relay, a low current sensor breakout (ACS712), and a liquid crystal display comprise the constructed system (LCD)[8]. The Arduino UNO is a microcontroller that can be programmed to execute custom code at any time. It is also a powerful microcontroller capable of receiving and transmitting data over the Internet through different modules and shield platforms[8].

The data is display on web servers, Thing-speak, and mobile apps like Virtuino using the Wi-Fi module. The data are also monitored in real time and on the cloud database. The monitoring system allows customers to see usage rates in real time. The SMACS control system may ensure safety, monitor individual or collective home appliances, and educate on power use. Future work will monitor voltage and current in a smart home context[8].

2.3.2 Maintenance of Transmission Line by Using Robot

This paper discusses the maintenance of transmission lines using robots for monitoring transmission lines and detecting transmission line faults. Overhead transmission line maintenance is tough to manage. Therefore, robotics will play a significant part in the electrical system to maintain the same. This reduces maintenance time and increases predictive maintenance for transmission lines. Given the employees' safety when working on overhead lines, it has a lot of promise. The robot is connected to the transmission line in the 500kV power line indefinitely. This approach is outfitted with a voltage sensor for measuring voltage on the transmission line and a current sensor for detecting current on the transmission line[9]. The RF module is used for communication. A visual camera is mounted on the robot to collect pictures that are then relayed to the control area. For simulation, Proteus software is used[9].

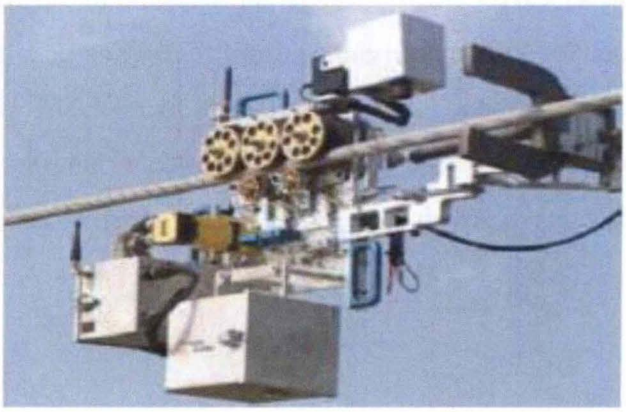

Figure 2.1 Transmission Line Robot

The robot consists of a current coil for measuring insulation strength. A 12V dc supply powered 12V DC motors and a micro-controller device with GPS and a thermal imaging camera. The current coil is used to measure the flow of current via live transmission lines and the change in reading indicates the insulation quality. It also determines transmission voltage. The thermal imaging camera measures the temperature of components and identifies obstructions in the path of the robot system. The picture is sent to the control room using GPS software. The robot is placed on the transmission line by turning on its control supply. If high current runs over ACSR transmission lines, current coils will monitor the current. The voltage is measured using a voltage transformer. The received current from CT determines the system's health, which is shown on the robot's LCD display. This technology lets us see live pictures of transmission lines and any obstructions that could hurt them. They use GPS software, an RF transmitter, and a receiver to send data to control rooms[9].

2.3.3 A Romanian Solution for Real-time Monitoring of Overhead Transmission Lines

The article presents the OHLM system, a Romanian innovative system for online monitoring of the operational parameters of the OHLs (based primarily on real-time measurements such as current through the line, conductor temperature, conductor inclination angle/sag, mechanical tension of the line, conductor oscillations, and so on), which includes intelligent electronic devices for data acquisition, processing, and storage, as well as remote data transmission (to the power system)[10].

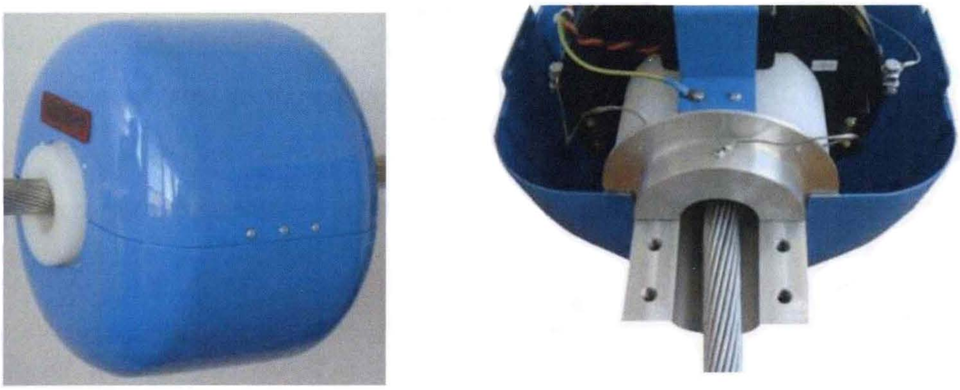

Figure 2.2 OHL conductor integrating OHLM-HV module

Data is wirelessly sent from the OHLMHV module installed on the line conductor to the OHLM-IED. Data transfer from the OHLM – IED to the OHLM PC server can be accomplished via fibre optics (simplified version), radio, or/and GSM/GPRS. The OHLM online monitoring system connects bidirectionally with the Data Management Centre while data and alarm transfers are done automatically and at the system operator's request.

2.3.4 Overhead Transmission Line Monitoring System for Dynamic Rating

In this paper discussed about the technology under development at Sumitomo Electric consists of many data-gathering terminals directly installed on transmission lines. These terminals monitor the temperature and current value of transmission lines. The data collection terminal, which is directly installed on a high-voltage wire, must be insulated, which makes it impossible to feed power to it from an external source. Therefore, the data gathering terminal was designed to produce the required power by itself from the transmission line current. The self-powering feature avoids the requirement for regular maintenance as would be necessary with battery-powered connections. In addition, the data collecting terminal includes a shopping feature that conveys information provided from a data collecting terminal in the vicinity wirelessly. Using this function, temperature, and the current values that the collecting terminals measure are transmitted to the concentrator using the radio of the 920 MHz zone while relaying them between collecting terminals and are finally transmitted from the concentrator to a data collection server using LTE lines[11].

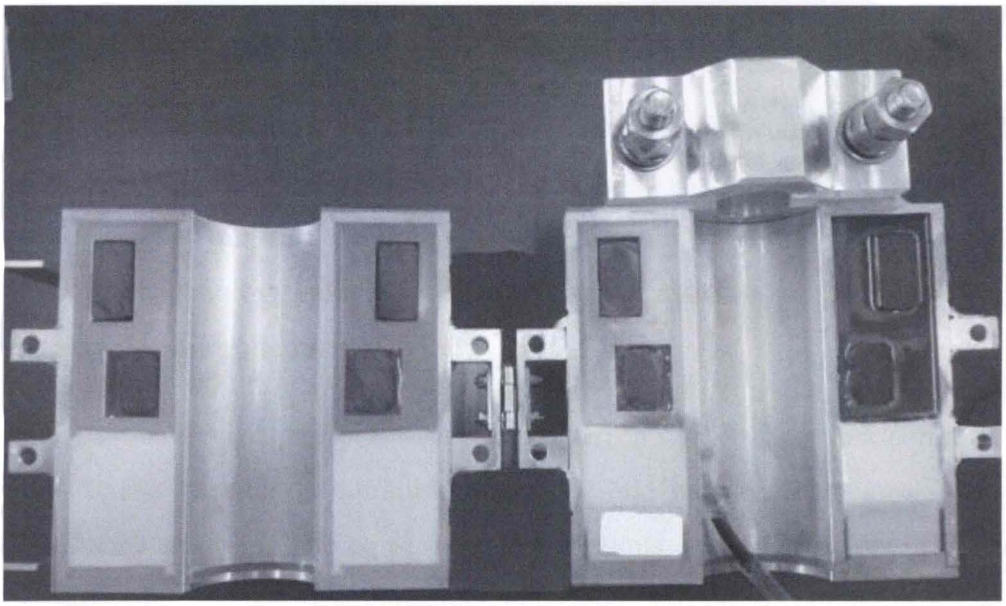

Figure 2.3 Inside structure of data collecting terminal

2.4 Internet of Things (IoT) Tools

Internet of Things (IoT) tools is a network or link of computers, cars, equipment with embedded electronics, household appliances, and buildings, among other things. This enables the collection and exchange of many types of data. It also enables the user to remotely operate the gadgets over a network, which are nonstandard computer devices that connect wirelessly to a network and have the capacity to communicate data (IoT)[12].

The Internet of Things entails expanding internet connectivity beyond ordinary devices such as PCs, laptops, smartphones, and tablets to any number of traditionally or non-internet-enabled physical equipment and daily things. These gadgets, which are embedded with technology, may connect, and interact over the internet. They can also be monitored and controlled remotely[12].

Connected devices are part of an ecosystem in which each item communicates with other related devices to automate household and industrial processes. They can provide valuable sensor data to consumers, corporations, and other parties. The gadgets are divided into three categories: consumer, enterprise, and industrial[12].

2.4.1 ThingSpeak

This cloud based IoT analytics solution gathers, displays, and analyses real-time data streams. ThingSpeak is a platform for developing Internet of Things (IoT) applications. Some of the capabilities include real-time data collection, graph visualization, and the ability to build plugins and apps that connect with web services, social networks, and other APIs. A 'ThingSpeak Channel' is the core of ThingSpeak. A channel contains data that we send to ThingSpeak and has 8 fields for storing data of any kind, a channel name, and a description. These may be used to store data from a sensor or an embedded device. A location's latitude, longitude, and elevation may all be recorded in three fields. These are great for tracking down a moving object. 1 status field a short description of the channel's data[13].

There were few hardware components available when ioBridge announced the IO-204 web gateway for the Internet of Things in 2008. IoBridge launched ThingSpeak.com in 2010 as an open data platform for IoT [14]. We need to join ThingSpeak and create a channel before using it. Once a channel is setup, we may send data to ThingSpeak and receive it.

To create for a Chanel, go to https://thingspeak.com/ and select the 'Get Started Now' button in the centre of the website. Fill in the required fields and click 'Create Account'.

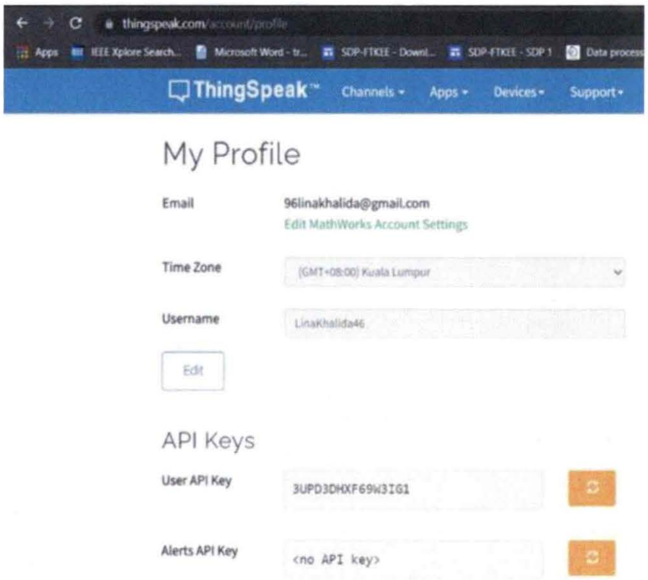

Figure 2.4 Profile after signing up to ThingSpeak account

2.4.1.1 Chanel

To communicate or retain info in Chanel. The description Chanel may be modified by the user. Latitude, longitude, and elevation are crucial characteristics for transporting things. I f the channel is made public, anybody may see the data stream and visualizations. If this check box is not checked, the channel is private, and each read or write requires an API key[14]. If provided, the URL of a user's blog or website will be shown publicly on the channel. Your video ID is linked to your YouTube or Vimeo account. If specified, the video is shown publicly. Fields 1-8 pertain to data from a sensor or 'object.' A field must be added before it may store data. Field 1 is pre-filled[14]. In this case, the request will be accepted, but you will not be able to see the field in the charts or data. To add a field, click the tiny box next to the 'add field' text. When you click the 'add field' box, the wording next to each field changes from 'add field' to remove field'. The default field wording may be modified to make more sense. Remove a field by selecting it and clicking remove'. 'Add field' is substituted for the text remove field'[14].

Figure 2.5 Interface to create chanel

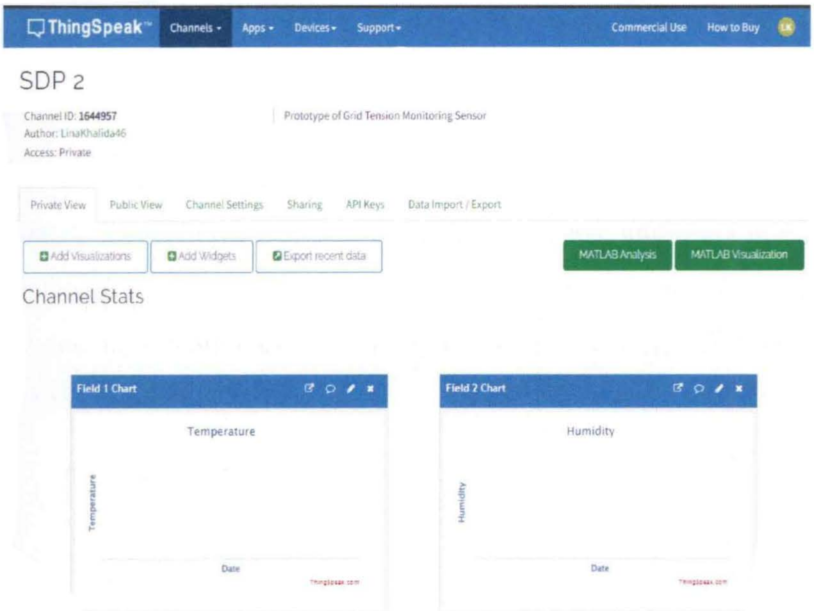

Figure 2.6 Interface of Chanel

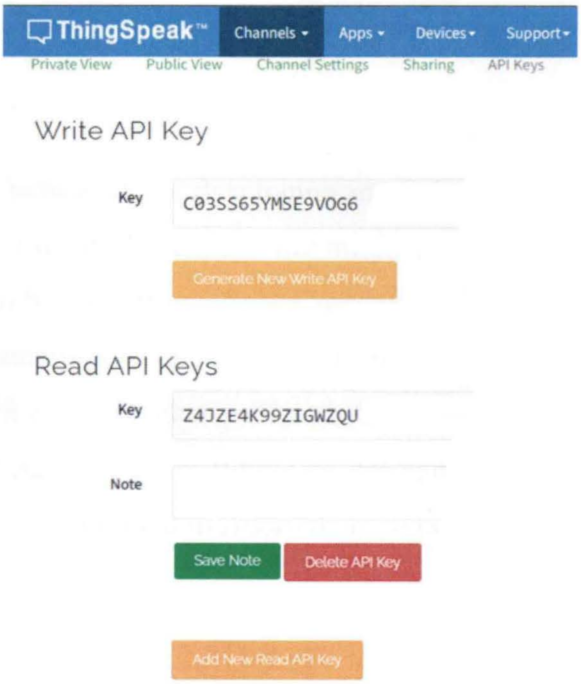

Figure 2.7 API key required for coding

2.4.1.2 Application of ThingSpeak

TimeControl (automatically perform actions at predetermined times with the ThingSpeak app), TweetControl (listen to the Twitterverse and react in real time), React (reacts when channel data meets some certain condition), and TalkBack (queue up command for the user's device) are some of the applications available. enhance the reaction time measurements[15].

2.4.2 Blynk

Blynk is a framework for developing smartphone applications that interact with a wide range of microcontrollers, requiring minimal mobile programming and focusing on functionality. To be sure, it is a free programme that used anywhere in the globe. It can also construct smartphone apps that let us to connect with microcontrollers or even whole computers like the Raspberry Pi and Arduino. Creating a mobile app that can communicate with Arduino is as simple as dragging a widget and specifying a pin. Blynk application also supported in Android and IOS.

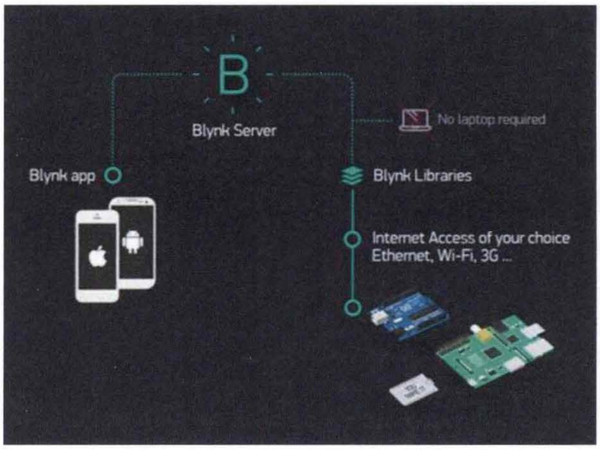

Figure 2.8 System architecture of Blynk application

The Blynk apps are particularly useful for monitoring systems or home systems that can be conveniently monitored utilising phones. When someone touches the button in the Blynk programme, the data is sent to the Blynk Cloud, where it miraculously finds its way to the installed hardware. It works the other way around, and everything happens in the blink of an eye.

2.4.2.1 Benefits of Blynk

The API and UI are the same for all supported hardware and devices. WIFI, Bluetooth, Ethernet, USB (Serial), and GSM are all used to connect to the cloud. Blnyk also has an easy-to-use widgets feature[16]. It consists of direct pin manipulation without the need for coding and the ability to easily integrate and add additional features utilizing virtual pins. In addition, historical data monitoring through the SuperChart widget and device-to-device connectivity via the Bridge Widget have been included. It is also simple to send emails, tweets, and push alerts[16].

2.4.3 GSM (Global system mobile communication) Module

A GSM module is a chip or circuit that will be used to connect a mobile device or computer equipment to a GSM. The modem (modulator-demodulator) is an important component in this process. These modules include a GSM module powered by a power supply circuit as well as computer connection interfaces (such as RS-232, USB 2.0, and others)[17]. A GSM modem might be a standalone modem with a serial, USB, or Bluetooth connection, or it might be a mobile phone with GSM modem capabilities. They can do all the functions of a mobile phone through a computer, such as making and receiving calls, SMS, MMS, and so on. These are most often used for computer-based SMS and MMS services[17].

The AT(Attention) commands are shown in the GSM/GPRS module. They can do all the functions of a mobile phone through a computer, such as making and receiving calls, SMS, MMS, and so on. These are used mostly for computer-based SMS and MMS services[17].

2.5 Summary

This chapter explains the project's system as well as the system that was implemented in developing the prototype of the Grid Tension Monitoring System. This included information about IoT devices and systems, as well as ThingSpeak, Blynk, and a GSM module.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This project method is described in detail in Chapter 3, along with the materials and IoT platform that were used in the project. The software system includes the ThingSpeak and Blynk application standards, as well as other components. The purpose of this chapter is to explain why the platform was selected. Moreover, in Chapter 3, the design prototype, flowchart, and system architecture for the project are shown.

3.2 Flowchart of Overall System

A flowchart of the overall system for this prototype grid tension monitoring system is shown in Figure 3.1. Figure 3.1 system consists of two conditions to be monitored in this system. When the full system is switched on, the sensor starts to read and returns to the starting condition. The first condition is to collect a measurement of temperature. If the temperature is less than 15°C and greater than 50°C, data will be transmitted from the microcontroller to the ThingSpeak and Blynk apps through the GSM module. If no exceeding results are obtained on the first condition, it will move to the second condition, which is the axis of the sag. If the axis value is less than -70° and more than 70°, the data will also be transferred and transmitted as a response of SMS and notifications. In addition, all data will be preserved on the ThingSpeak cloud. The system will loop until users chose to stop it; at which time it will quit looping, system off.

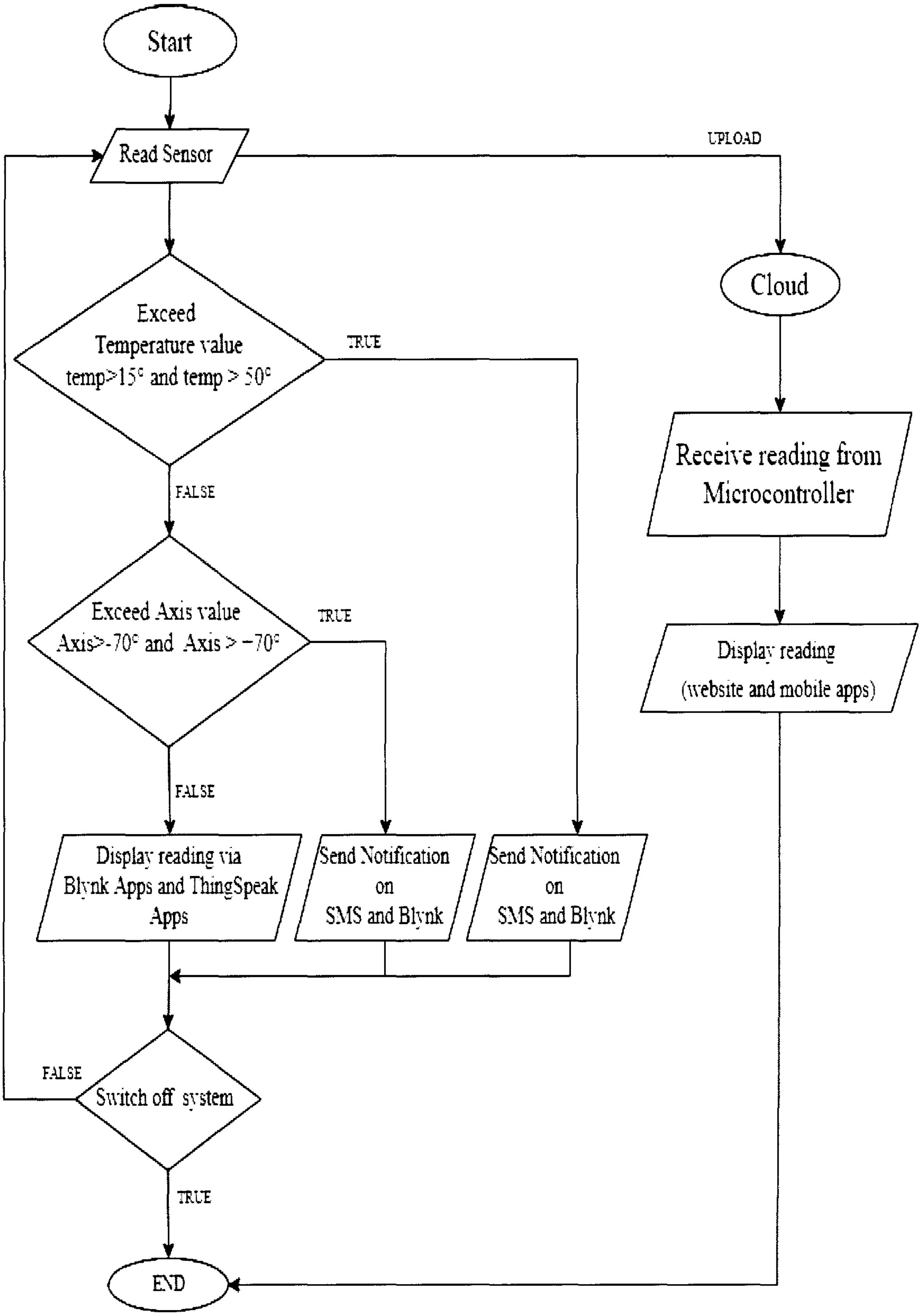

Figure 3.1 Flowchart of proposed algorithm

3.3 Proposed Application Design

The graphical design is completed by using the Arduino Uno's ThingSpeak and Blynk servers for sensor reading to be sent through the GSM900 module. The GSM900 works as a medium for the interaction between device and device. Without the GSM900 module, the transmission and receiving of data will not be done at all. Hence, for a wireless transfer of data, a GSM900 module must be inserted in one of the devices in Arduino Uno. To interact with an Arduino, the SIM900 GSM/GPRS shield use the UART protocol. Linked the shield's (RX,TX) to the Arduino's Software Serial (D8,D7) or Hardware Serial (D1,D0) via jumpers. The shield offers two antenna interfaces: a U.FL connection and a SMA connector. Thus, can be used right away after installation.

The system information feedback will rely on IoT platforms to transmit sensor reading to a website and mobile apps. The website can be accessed by both PCs and phones both Android and iOS, if they are connected to the same local network as the host and in this case the microcontroller is set as the host as shown on the figure, where the server runs at a local IP address, which is also the address of the website, which any use can access by searching www.https://thingspeak.com [8] .

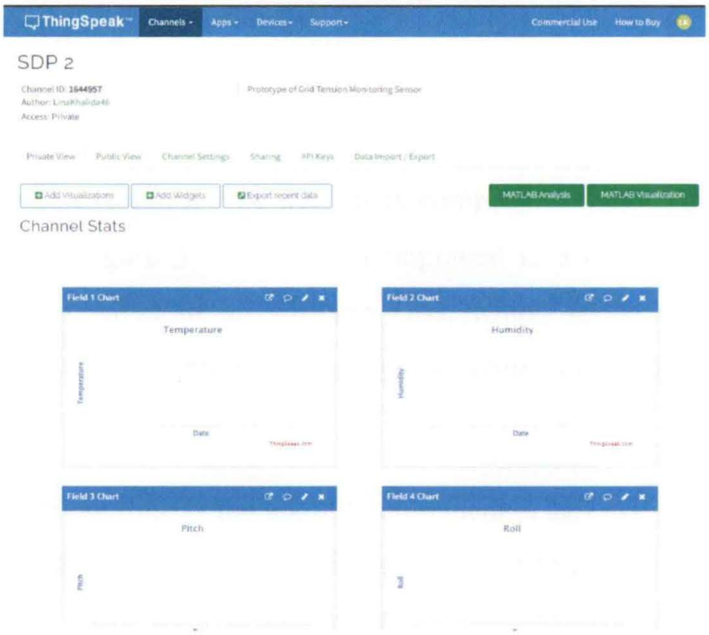

Figure 3.2 ThingSpeak website

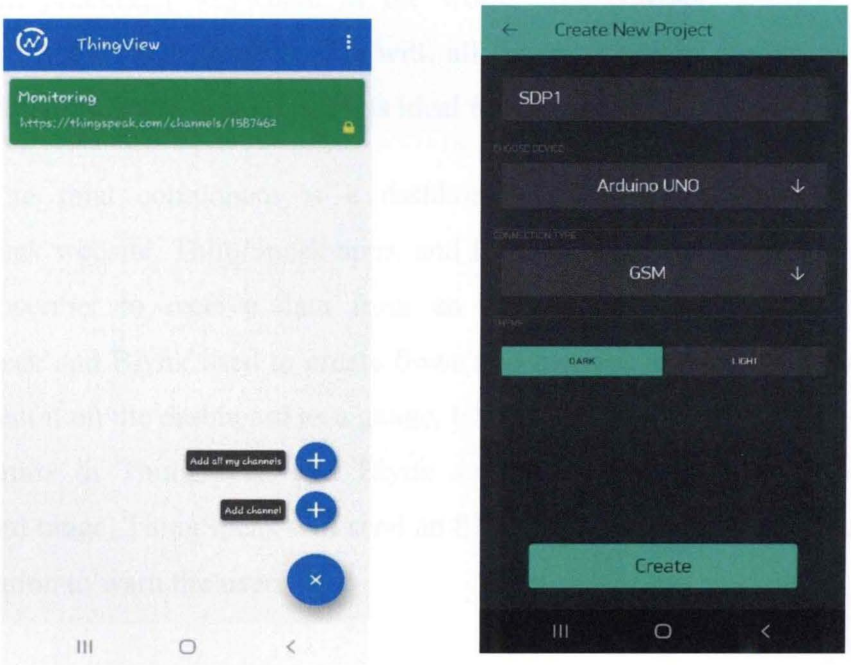

Figure 3.3 ThingSpeak and Blynk graphical interface

3.4 System Design

The grid tension monitoring system is composed of three major components, which are shown in Figure 3.4. The first component is a device for measuring grid tension. The sensors wired to an Arduino Uno, which serves as a publisher. The Arduino Uno will gather, interpret, and manipulate sensor data before transmitting it to GSM900.

The GSM900 module is the second component. The UART protocol is used by the SIM900 GSM/GPRS shield. The device supports baud rates ranging from 1200bps to 115200bps with Auto-Baud detection. It can use this shield to perform almost whatever a typical mobile phone can do, such as send SMS text messages, make, or receive phone calls, connect to the internet via GPRS, TCP/IP, and more. To top it all off, the shield is compatible with a quad-band GSM/GPRS network, which means it will work practically anywhere in the world. The SIM900 GSM/GPRS shield is designed to cover the SIM900 chip with all the components needed to communicate with Arduino. It is also small, which is ideal for our system design.

The final component is a dashboard for measuring grid tension on the ThingSpeak website, ThingSpeak apps, and Blynk apps. ThingSpeak and Blynk utilize as a subscriber to receive data from an Arduino Uno through a GSM module. ThingSpeak and Blynk used to create flows that manage and handle the collected data and present it on the dashboard as a gauge, text, and chart. Furthermore, additional were added limits in ThingSpeak and Blynk so that if the value of data exceeds the configured range, ThingSpeak will send an SMS to the user and Blynk apps would send a notification to warn the users.

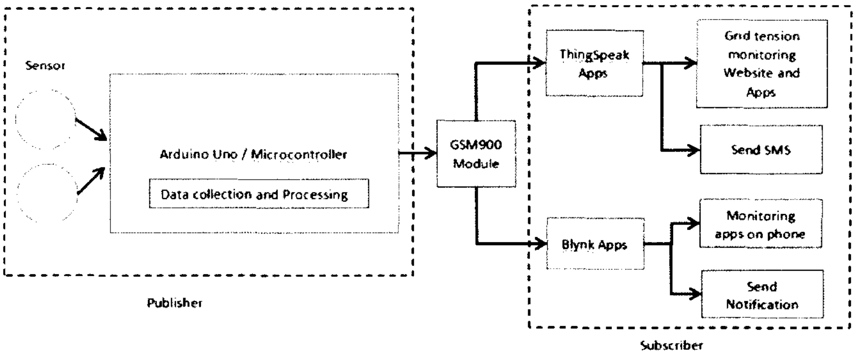

Figure 3.4 System architecture

3.5 GSM Module

In an alarming situation, GSM (Global System for Mobile Communication) sends an SMS to the user's mobile phone. This GSM module has an RS232 port for serial communication as well as external peripherals. Alert SMS are sent over the GSM network, which operates on AT instructions. The coded results are received by the microcontroller and then transmitted to the receiver to ThingSpeak and Blynk through the controller. The microcontroller has been pre-programmed to receive and send data at a baud rate of 9600.

3.5.1 AT command used

The AT command is used to verify communication between the module and the computer.

- AT+CPIN is to set command transmits a password to the device, which is required before it may be used.

- AT+CREG is depending on the option, the set command allows or disables network registration reporting.
- AT+CGATT is depending on the parameter, the execution command is used to connect or disconnect the terminal from the GPRS service.
- AT+CMGF are used to define the SMS format. The text format is implemented here.
- AT+CMGW are used to record messages in SIM cards.
- AT+CMGS are used to send text messages to a phone number.

3.6 ThingSpeak

ThingSpeak gathers, visualises, and analyses real-time data streams on the cloud. ThingSpeak major qualities include the ability to simply setup devices to transfer data to ThingSpeak using conventional IoT protocols. Real-time display of sensor data On-demand data aggregation from GSM900. The recorded data allows utilities providers and users to monitor load statistics on devices and PCs. The recorded data offers load patterns, dynamic invoicing, and aids in better balancing demand and supply of power between production and consumption.

The signal is amplified before being sent into the Arduino board to be processed. The procedure of saving temporary data on a ThingSpeak cloud, among other things, was then carried out. The GSM900 module is used to send data to the ThingSpeak server. Because of the ThingSpeak service, this temporary data storing step is essential. Therefore, data is gathered once per minute. The Arduino IDE is also used for coding throughout the process.

3.7 Blynk

The Blynk applications are especially beneficial for monitoring systems that can be easily monitored with phones. When users press the Blynk program's button, the data is routed to the Blynk Cloud, where it somehow finds its way to the installed hardware. Everything occurs in the blink of an eye the other way around. Blynk can also communicate with a variety of microcontrollers, requiring minimum mobile programming and focused on functionality. To be sure, it is a free software that may be used from anywhere on the planet. It can also build smartphone applications that let us to communicate with microcontrollers or even whole computers, such as the Raspberry Pi and Arduino. It is as easy as dragging a widget and specifying a pin to create a mobile app that can connect with Arduino. The Blynk app is also available for Android and iOS.

3.8 Summary

This chapter begins with flowchart of overall system from beginning until end of the system flow for prototype grid tension monitoring system. Finally, it also contains system architecture,GSM900 module, ThingSpeak and Blynk methodology for this project.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

The chapter present the outcomes of the ThingSpeak website, mobile application, and Blynk application are addressed in depth in this chapter. We will talk about the interface of the grid tension monitoring system as well as the GUI design of the ThingSpeak and Blynk.

4.2 Interface of grid tension monitoring system

To begin, we use two types of sensors which is DHT22 temperature and humidity sensor and the ADXL355 three-axis accelerometer. We utilize the DHT22 sensor as an alternative since it senses the external temperature and humidity. Meanwhile, sensor ADXL355 detects movement in the axes x, y, and z. It is quite easy to identify irregular movement. Because sagging that is too tight or too loose might lead to problems in the future. As an example, figure 4.2 interface the grid tension monitoring system interface.

Figure 4.1 and 4.2 shows a graph of sensor readings vs time. The time will be updated in the 1-minute delay between reading and displaying on the monitor and mobile applications. It also displays the date of data collected daily if the system is turned on. If the system is turned off, the graph resets to zero and begins again to present sensor readings. All the data reading is stored in cloud of ThingSpeak. Therefore, we can analyse the behaviour the sag tension.

Figure 4.1 Interface monitoring system in Mobile apps via ThingSpeak

Figure 4.2 Interface monitoring system in Website via ThingSpeak

Figure 4.3 Interface monitoring system in Blynk apps

The gauge indication for temperature reading and axis reading will be shown on a digital dashboard. The value display will then show the temperature and axis sensed by the DHT22 and ADXL355 sensors. Finally, the notification indicator will monitor the sent alert and store it to the database. Figure 4.3 shows the Blynk layout design utilized in this project. Blynk bridge is another technique to connect two Arduino Uno to GSM 900. It may link all its projects by using the cloud Blynk server as a real-time central activity manager. At the same time, a database was constructed in this Blynk application to record the temperature and axis via sensors.

4.3 Discussion

This project includes a significant variety of technologies from hardware to software, and then integrate them to develop a grid tension monitoring system. A significant amount of technical study and testing were done to figure out relevant technologies for these projects. During development, there are various difficulties appearing which enable the developer gains not only technical but also problem-solving abilities by figuring out how to face such obstacles. The complete project length is scheduled to be 8 weeks, but it takes shorter time to grid tension monitoring system from scratching. It is apparent that many functions still can be enhanced further.

Figure 4.4 represents a graph of the temperature measured with a DHT22 temperature and humidity sensor from October 2022 to January 2022. The horizontal axis shows the actual time, while the vertical axis displays the temperature. The graph shows that the temperature has fluctuated between rising and falling. The temperature began to fall at 1.26 p.m., but the temperature generally rises and falls depending on the surroundings. We mostly tested indoors for this project. As a result, the temperature range is set at room temperature, which is 23 °C to 26 °C.

On Figure 4.5 represents a graph of the axis measured with ADXL355 3 axis accelerometer sensor from October 2022 to January 2022. The horizontal axis shows the actual time, while the vertical axis displays the pitch and roll. The graph shows that changes of pitch and roll has fluctuated between tighten and loose. On the pitch graph shows the loose part at 0.01° and the highest pitch measured at 26.83°,meanwhile on roll graph shows the loose part is -19° and the highest roll is 5.26°, but the roll and pitch generally tight and loose depending on the movement of axis x, y and z .

Figure 4.4 Measured Temperature Data from ThingSpeak mobile apps

Figure 4.5 Measured Axis Data from ThingSpeak mobile apps

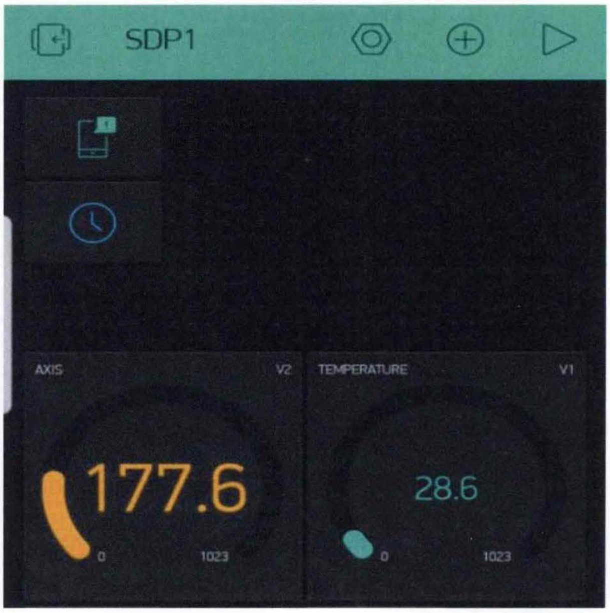

Figure 4.6 Measured Temperature and Axis from Blynk mobile apps

Figure 4.6 illustrates a graphical display of the measured temperature and sag axis. Blynk will display real-time data collected from the sensors. The value will change over time, and if it exceeds the abnormal measure, the user will be notified by Blynk. Figure 4.6 depicts the temperature of 28.6 °C, which is a slightly hot temperature measured, but the axis is still in normal condition, with no overtightening or under tightening at 177.6 °.

4.4 Summary

This chapter begins with the result of the interface of ThingSpeak on the website and mobile application. It also included the interface of the Blynk application's digital design. Lastly, discussed the results obtained from the data collected from testing the system.

CHAPTER 5

CONCLUSION

5.1 Conclusion

View of the foregoing information, we can conclude that by establishing a real-time monitoring system for maintenance reasons, we can reduce the time that will influence the other systems during repair. By developing a grid tension detector, it is possible to cut power losses while also solving the safety problems connected with existing monitoring systems. It will be useful in the future to carry out transmission line maintenance. This system runs on an automated mechanism that accepts and monitors data using Blynk and ThingSpeak apps. The website and mobile application satisfy most of the standards, however certain issucs remain.

As a result, future developments may be considered. We encountered two challenges while working on this project. First, the Blynk program did not loop the system again. It indicates that only read once. During the testing stage, Blynk read and displayed the output. However, when we attempted to move the sag where the on-axis value changed, the Blynk did not read and did not show the real time reading by the ADXL355 sensor and DHT22 sensor. Secondly, the GSM900 connection coverage limitations, wherein a certain location has both excellent and terrible coverage. The data flowing from sensors to the application may be disrupted if the Internet connection fails. Though there is obstacle on the system, this system has featured a small and simple design that is great for cost- saving.

Results from the experiments show the potential impact of these real-time monitoring systems in terms of better grid tension utilization and an improvement in system reliability. This project prototype also results in the development of a useful system for improving the efficiency with which grid tension transmission line facilities are maintained.

5.2 Recommendation

To summarize, to strengthen this project in future, some can be added for improvement. First, additional dashboard with multifunctional reading such as for check the quality of the sag on ThingSpeak and Blynk apps. The addition of function offers additional demands of this system.

Other than that, built in standalone battery and solar can make the prototype more compatible , taken into consideration that being that the present project prototype employing DC voltage. To have excellent feature having standalone battery and solar make the gadgets always switch on without trouble and simple to care and maintenance.

REFERENCES

[1] "The National Grid MYR 8.48 Volume 6 *Price includes GST," Accessed: Feb. 14, 2022. [Online]. Available: www.st.gov.my.

[2] "The maintenance of a transmission line." http://www.theenergyofchange.com/maintenance-transmission-line (accessed Feb. 16, 2022).

[3] G. Xu, Q. Wu, M. Daneshmand, Y. Liu, and M. Wang, "A data privacy protective mechanism for wireless body area networks," *Wirel. Commun. Mob. Comput. Wirel. Commun. Mob. Comput*, vol. 16, pp. 1746–1758, 2016, doi: 10.1002/wcm.2649.

[4] M. S. Gertler, "Tacit knowledge and the economic geography of context, or the undefinable tacitness of being (there)," *J. Econ. Geogr.*, vol. 3, no. 1, pp. 75–99, 2003, doi: 10.1093/jeg/3.1.75.

[5] A. H. Bagdadee, M. Z. Hoque, and L. Zhang, "IoT Based Wireless Sensor Network for Power Quality Control in Smart Grid," in *Procedia Computer Science*, 2020, vol. 167, pp. 1148–1160, doi: 10.1016/j.procs.2020.03.417.

[6] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014, doi: 10.1109/JIOT.2014.2306328.

[7] A. H. Miry and G. A. Aramice, "Water monitoring and analytic based thingspeak," *Int. J. Electr. Comput. Eng.*, vol. 10, no. 4, p. 3588, Jan. 2020, doi: 10.11591/IJECE.V10I4.PP3588-3595.

[8] M. K. Hasan, M. M. Ahmed, B. Pandey, H. Gohel, S. Islam, and I. F. Khalid, "Internet of Things-Based Smart Electricity Monitoring and Control System Using Usage Data," *Wirel. Commun. Mob. Comput.*, vol. 2021, 2021, doi: 10.1155/2021/6544649.

[9] P. B. Wale and K. Kamal Sandeep, "Maintenance of transmission line by using robot," in *International Conference on Automatic Control and Dynamic Optimization Techniques, ICACDOT 2016*, Mar. 2017, pp. 538–542, doi: 10.1109/ICACDOT.2016.7877643.

[10] C. Moldoveanu, I. Ionita, S. Zaharescu, V. Florea, L. Iacobici, and I. Hategan, "A Romanian Solution for Real-time Monitoring of Overhead Transmission Lines," *Proc. 2021 9th Int. Conf. Mod. Power Syst. MPS 2021*, Jun. 2021, doi: 10.1109/MPS52805.2021.9492597.

[11] "Overhead Transmission Line Monitoring System for Dynamic Rating."

[12] "What are IoT Devices?" https://internetofthingsagenda.techtarget.com/definition/IoT-device (accessed Feb. 15, 2022).

[13] F. Khan, M. A. B. Siddiqui, A. U. Rehman, J. Khan, M. T. S. A. Asad, and A. Asad, "IoT Based Power Monitoring System for Smart Grid Applications," Feb. 2020, doi: 10.1109/ICEET48479.2020.9048229.

[14] "Internet of Things Apps - ThingSpeak IoT." https://thingspeak.com/apps (accessed Feb. 12, 2022).

[15] "(PDF) Water monitoring and analytic based ThingSpeak | International Journal of Electrical and Computer Engineering (IJECE) - Academia.edu." https://www.academia.edu/43785373/Water_monitoring_and_analytic_based_ThingSpeak (accessed Feb. 12, 2022).

[16] "1. What is Blynk? - Tech Explorations." https://techexplorations.com/guides/blynk/1-what-is-blynk/ (accessed Feb. 16, 2022).

[17] "What is GSM Module and GPRS Module? | Expert Class Notes." https://www.electronicsforu.com/resources/gsm-module (accessed Feb. 16, 2022).

[18] "Internet of Things Apps - ThingSpeak IoT." https://thingspeak.com/apps (accessed Feb. 14, 2022).

APPENDICES

Appendix A: Coding of the system

```
#include <math.h>
#include <SoftwareSerial.h>
#include <String.h>
#include <DHT.h>
#define DHTPIN A0

 //blynkk
#define BLYNK_PRINT Serial
#define TINY_GSM_MODEM_SIM900
#include <TinyGsmClient.h>
SoftwareSerial SerialAT(7,8);
#include <BlynkSimpleTinyGSM.h>

SoftwareSerial gprsSerial(7,8);
DHT dht(DHTPIN, DHT22);

const int x_out = A1;
const int y_out = A2;
const int z_out = A3;

//blynkk setup
char auth[] = "nCjf_NwW7EdAqTdHaKFJLIhWQktK1-xA";
char apn[]  = "Celcom4g";
char user[] = "";
char pass[] = "";
TinyGsm modem(SerialAT);

void setup()
{
  gprsSerial.begin(9600);          // the GPRS baud rate
  Serial.begin(9600);              // the GPRS baud rate
  dht.begin();

  delay(1000);
}
```

Figure 5.1 Blynk setup and libraries

```
void loop()
{
      float h = dht.readHumidity();
      float t = dht.readTemperature();

      int x_adc_value, y_adc_value, z_adc_value;
      double x_g_value, y_g_value, z_g_value;
      double roll, pitch, yaw;
      x_adc_value = analogRead(x_out);
      y_adc_value = analogRead(y_out);
      z_adc_value = analogRead(z_out);

      x_g_value = ( ( ( (double)(x_adc_value * 5)/1024) - 1.65 ) / 0.330 ); // Acceleration in x-direction in g units
      y_g_value = ( ( ( (double)(y_adc_value * 5)/1024) - 1.65 ) / 0.330 ); // Acceleration in y-direction in g units
      z_g_value = ( ( ( (double)(z_adc_value * 5)/1024) - 1.80 ) / 0.330 ); // Acceleration in z-direction in g units

      roll = ( ( (atan2(y_g_value,z_g_value) * 180) / 3.14 ) + 180 ); // Formula for roll
      pitch = ( ( (atan2(z_g_value,x_g_value) * 180) / 3.14 ) + 180 ); // Formula for pitch

      delay(100);

      Serial.print("Temperature = ");
      Serial.print(t);
      Serial.println(" °C");
      Serial.print("Humidity = ");
      Serial.print(h);
      Serial.println(" %");
      Serial.print("Roll = ");
      Serial.print(roll);
      Serial.print("\t");
      Serial.print("Pitch = ");
      Serial.print(pitch);
      Serial.print("\n\n");
```

Figure 5.2 Coding for reading sensors

```
if (gprsSerial.available())
  Serial.write(gprsSerial.read());

gprsSerial.println("AT");
delay(1000);

gprsSerial.println("AT+CPIN?");
delay(1000);

gprsSerial.println("AT+CREG?");
delay(1000);

gprsSerial.println("AT+CGATT?");
delay(1000);

gprsSerial.println("AT+CIPSHUT");
delay(1000);

gprsSerial.println("AT+CIPSTATUS");
delay(2000);

gprsSerial.println("AT+CIPMUX=0");
delay(2000);

ShowSerialData();

gprsSerial.println("AT+CSTT=\"celcom4g\",\"\",\"\"");//start task and setting the APN,
delay(1000);

ShowSerialData();

gprsSerial.println("AT+CIICR");//bring up wireless connection
delay(3000);
```

Figure 5.3 Coding for ThingSpeak server

```
String str="GET https://api.thingspeak.com/update?api_key=NLO4VE14D96XME84&field1=" + String(t) +"&field2="+String(h)+"&field3="+String(pitch)+"&field4="+String(roll);
Serial.println(str);
gprsSerial.println(str);//begin send data to remote server

delay(4000);
ShowSerialData();

gprsSerial.println((char)26);//sending
delay(5000);//waitting for reply, important! the time is base on the condition of internet
gprsSerial.println();

ShowSerialData();

gprsSerial.println("AT+CIPSHUT");//close the connection
delay(100);
ShowSerialData();

//blynkk display
Blynk.run();
Blynk.virtualWrite(V1, t);
Blynk.virtualWrite(V2, roll);
Blynk.virtualWrite(V3,pitch);
if (pitch <700)
Blynk.notify("Dangerous!"); //to notify when its have lower value what will happen


}
void ShowSerialData()
{
while(gprsSerial.available()!=0)
Serial.write(gprsSerial.read());
delay(5000);

}
```

Figure 5.4 Coding for transmit to ThingSpeak and Blynk apps

Appendix B: Design

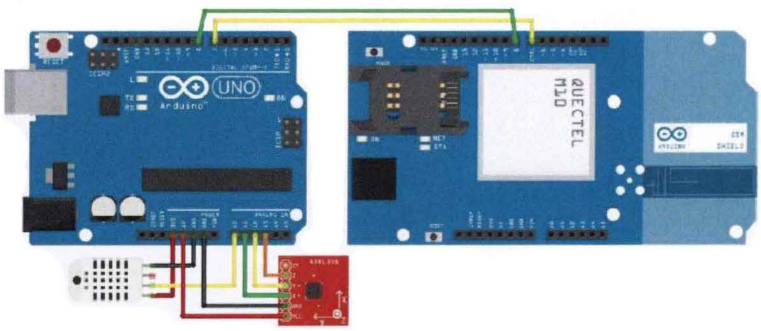

Figure 5.5 Hardware design via Fritzing

Appendix C: Final Prototype

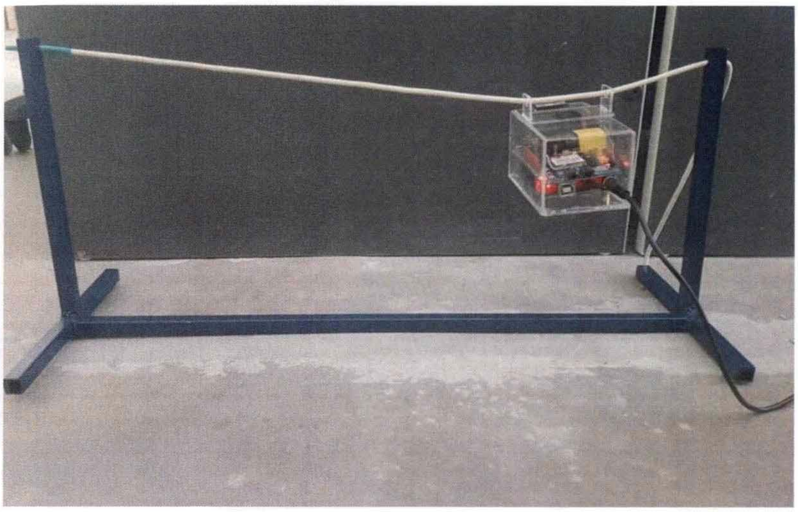

Figure 5.6 Full view of prototype

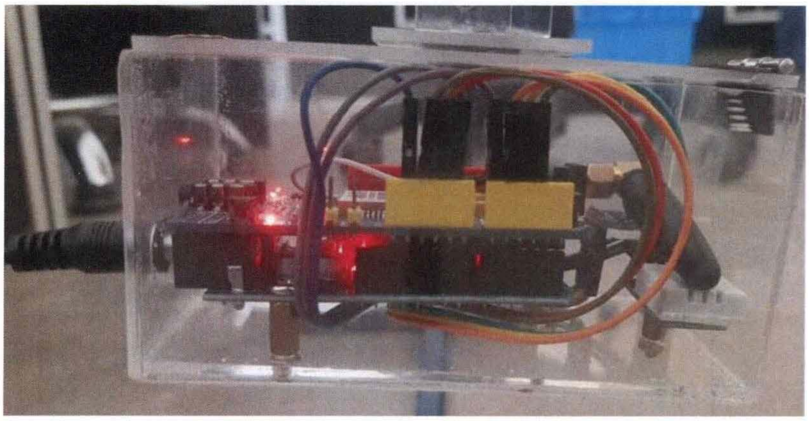

Figure 5.7 Side view of prototype

Appendix D: Cost of Analysis

Table 5.1 Material and cost analysis

Item	Price (RM)	Quantity
Arduino Uno	22.00	1
DHT22 Temperature and Humidity sensor	18.40	1
ADXL355 3 axis accelerometer sensor	23.36	1
GSM900 Module	51.95	1
2mm Perspex , 3ft x 3ft	30.0	1
1-inch Hinges	1.95	1
PCB Stand	11.49	1
Total	159.15	

Appendix E: Gantt Chart Project SDP 2

Figure 5.2 Gantt Chart Planning for SDP2