SMART MINI HYBRID AUTOMATION SYSTEM

(SIHAS)

MUHAMMAD SADIQ BIN SULIAMAN

This report is submitted in Partial Fulfilment of Requirement for the

Degree of Bachelor in Electrical Engineering

(Control, Instrumentation and Automation)

Faculty of Electrical Engineering

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2015

I declare that this report entitle “Smart Mini Hybrid Automation System” is the result of my own research except as cited in the references. The report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : .....

Name : Muhammad Sadiq bin Suliaman

Date : 3 / 6 / 2015

" I hereby declare that I have read through this report entitle "Smart Mini Hybrid Automation System" and found that it has comply the partial fulfilment for awarding the Degree of Electrical Engineering (Control, Instrumentation, Automation)"

Signature : .....

Supervisor's Name : Ahmad Fairuz bin Muhammad Amin

Date : 3 / 6 / 2015

ACKNOWLEDGEMENT

First and foremost, I would like to express my gratitude to my supervisor, Mr. Ahmad Fairuz bin Muhammad Amin for assisting and guiding me in order to complete this final report.

Next, I would like to express my appreciation to my family who has been so tolerant and supports me in these years. Besides that, special thanks to my friends and seniors who helped me directly or indirectly in completing my project report. No forgetting and grateful appreciation for the lab technician and lecturers who give me great assistance and opinion during the process in accomplishing my project.

ABSTRACT

Hybrid is a combination of the two supplies in this project by using Photovoltaic and Generator charge to Battery Bank. Components in the home need energy to operate 24 hours each time to meet human needs. The battery is used as an alternative energy during the night and the replenishment of batteries should occur during the daytime. All loads in the home will continue to function as voltage batteries will not come down to the minimum and to prolong the lifespan of batteries. This project aims to produce energy for home appliances distributor. Circuit of the solar charger adapters available in the market will be carried out and the controller is able to distribute energy to all applications in the home. Research and analysis on the charger, solar and electric generator available in the market need for modification of the circuit and replace it with arduino. This project also includes research in solar charger, generator and the means of modification of the circuit. A standby generator is a back-up electrical system that operates automatically. Within seconds of a utility outage an automatic transfer switch senses the power loss, commands the generator to start and then transfers the electrical load to the generator. The standby generator begins supplying power to the circuits. After utility power returns, the automatic transfer switch transfers the electrical load back to the utility and signals the standby generator to shut off. It then returns to standby mode where it awaits the next outage. Most units run on diesel. Battery charger device used to put energy into the cells, secondary or rechargeable battery by forcing electric current through it. In smart hybrid automation system, it have additional operation to the system that can be monitor through web server by develop the interface.

ABSTRAK

Hibrid adalah gabungan dua bekalan dalam projek ini dengan menggunakan Photovoltaic dan caj Generator kepada Bank Bateri.. Komponen dalam rumah memerlukan tenaga untuk beroperasi 24 jam setiap masa bagi memenuhi kehendak manusia. Bateri digunakan sebagai tenaga alternatif semasa waktu malam dan proses pengisian semula batteri harus berlaku semasa waktu siang. Semua beban dalam rumah akan terus berfungsi kerana voltan batteri tidak akan turun ke nilai minimum dan ini untuk memanjangkan jangka hayat batteri. Projek ini bertujuan untuk menghasilkan tenaga pengedar untuk aplikasi rumah. Pengubahsuai litar daripada pengecas solar yang terdapat di pasaran akan dijalankan dan pengawal ini dapat mengagihkan tenaga kepada semua aplikasi dalam rumah. Kajian dan analisis ke atas pengecas solar dan penjana elektrik (generator) yang ada di pasaran diperlukan untuk pengubahsuaian litar dan menggantikannya dengan arduino. Projek ini juga merangkumi kajian pada pengecas solar, pejanaan elektrik (generator) dan cara-cara pengubahsuaian litar. Pada masa yang akan datang, pembaikan litar boleh fokus pada tempat pengecas supaya mencapai kecekapan yang tinggi dalam simpanan tenaga. Keadaan penjana yang siap sedia adalah sistem siap sedia elektrik yang beroperasi secara automatik. Dalam masa beberapa saat gangguan utiliti deria pemindahan kehilangan kuasa suis automatik, mendapat penjana untuk memulakan dan kemudian memindahkan beban elektrik penjana. Penjana siap sedia bermula membekalkan kuasa kepada litar. Selepas pulangan kuasa utiliti, suis pemindahan automatik memindahkan beban elektrik kembali ke utiliti dan isyarat penjana siap sedia untuk menutup. Ia kemudian kembali ke mod siap sedia di mana ia menunggu gangguan seterusnya. Kebanyakan unit menjalankan terhadap diesel. Pengecas bateri peranti yang digunakan untuk meletakkan tenaga ke dalam sel menengah atau bateri yang boleh dicas semula dengan memaksa arus elektrik melaluinya. Dalam sistem automasi pintar hibrid, ia mempunyai operasi tambahan kepada sistem yang boleh memantau melalui pelayan web dengan membangunkan antara muka.

TABLE OF CONTENTS

LIST OF FIGURE

LIST OF TABLES

LIST OF APPENDICES

LIST OF ABBREVIATIONS

AC - Alternating Current

ADC - Analogue Digital Converter

Ah - Ampere per Hour

Cm - Centimetre

DC - Direct Current

APD - Analogue Panel Display

FKE - Fakulti Kejuruteraan Elektrik

I - Current

L - Length

LED - Light-Emitting Diode

LVD - Low Voltage Disconnect

MPPT - Maximum Power Point Tracker

PCB - Printed Circuit Board

PV - Photovoltaic

PWM - Pulse Width Modulation

SPDT - Single pole, double throw

V - Voltage

W - Watt


UNIVERSITI TEKNIKAL MALAYSIA MELAKA
UTeM
اونيۏرسيتي تيکنيکل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 1

INTRODUCTION

1.1 Research Background

1.1.1 Research Background

First and foremost, hybrid is not a new idea and it has been existed few years before. Apart from that, solar hybrid generator offers great condition for all human especially for the countries that face four seasons. Moreover, we can reduce the use of coal and fuel to produce electricity. For example, we can generate power electric by using photovoltaic (PV), battery, and back-up generator.

Besides, the electric system has many factors that can be incorporated into the final product. We know that when the winter season is coming, only little diffusion of sunlight can reach to the Earth. On the other hand, our country Malaysia is located few degrees north of the equator and therefore the temperature is almost uniform throughout the years.

Moreover, some equipment in the house need energy generated from the heat or light from the sun which can be used to produce heat, electricity, and cooling. Apart from that, all the equipment in hybrid house requires energy to function. We need a controller that can convert the energy from the PV panel and supply to all the equipment in house. Moreover, the equipment will be lack of energy during night and a battery is an alternative solution for the equipment to operate without any interruption.

The importance of the server is growing with the growth of the Internet and the log data. The load on the server has rapidly grown with the increase in the number of Web server users. Thus, load monitoring has become an important technical aspect in the performance of the server for these reasons. The web-based is used to monitor the system of voltage and power in hybrid. The details of hardware and software platform are design. The concept of symbol table based on the analysis of the characteristics of energy monitoring data is introduced as an asynchronous interactive manner of the control software and Web interface to implement data collection. It also offers powerful means for helping and supporting the special needs of people with disabilities and in particular, the elderly. This application domain is very important and will steadily increase in the future.

1.2 Problem Statement

House usually involves different type of components for example fan, lamp and so on. These components require energy to operate efficiently. Moreover in hybrid situation, energy storage is needed for the components to operate. Besides, the component will be lack of energy during night or cloudy weather and battery bank is used for alternative supply during that period. Furthermore, the battery needs to be charged during day time to store the energy efficiently. Thus, the usage of smart hybrid system can reduce the energy use at home and provides savings in electricity consumption. In addition, it can reduce the use of coal and fuel to produce electricity. In addition, the smart hybrid systems that can be controlled and manages through the internet make it more convenient and user-friendly.

To provide twenty-four (24) hour electricity supply, the usage of smart hybrid can reduces the cost of electricity compared to the grid supply. Generator engine life also proved unsatisfactory. For isolated areas, fuel logistics and handling will increase operational costs. In the other hand, a solar electric system has many factors that should be incorporated into the final product. Hence, an engineered system design maximizes reliability and efficiency along with lowering maintenance requirements. The applications of renewable energy and alternative power supply should be established widely as a solution to replace the usage of grid supply.

It would be easy for the consumer if they can monitor their hybrid system by using the internet. Hence, the implementation of automation system that uses Wi-Fi technology as a network infrastructure would help. The servers which present system core can locally (LAN) or remotely (internets) manages and control the hybrid system.

1.3 Objective

There are three objectives in this project:

I. To develop a controller for energy distribution in the house.
II. To develop the system for automatically charged and discharged the battery by using unused energy (solar), and generator.
III. To develop web-server for monitoring the system by using internet.

1.4 Scope of Research

This project focuses on the power supply from the PV panel and the energy distributed to battery storage and loads used. Other than that, this project was involved in circuit development for the existing solar charger controller in the market. Beside, this project also involved in circuit development of automatic generator controller method to back-up the low battery. This was the fundamental charging battery. Beside, all the system runs in DC and all equipment in the house are using DC 12 Volts to operate but in the reality, inverter was needed to convert DC to AC. These projects were used one piece of PV panel and one unit of 12 volts battery due to the limitation of cost. Furthermore, MySQL database is used to collect every single data and send to real-time web-server internets to monitor the energy when charging and discharging processes.

1.5 Motivation

The function of hybrid system is to replace the grid supply. It is not easy to control the system operate automatically when there is lack of energy during cloudy weather because there is no renewable energy to process. To avoid this, the smart hybrid automation system is use to operate the system automatically by charging and discharging it and also can be monitor through the web server anywhere.

The use of smart hybrid automation technique is designed and implements a remotely controlled, energy-efficient, safe cost, and an alternative power. As a central controller, an Arduino microcontroller is use to communicates with the web server, the user interface. The house network brings together both Arduino and Wi-Fi router, thus making it a cost-efficient hybrid system. Events can be programmed to be triggered under specific conditions, and this can have a great role in reducing the total energy consumed by some appliances.

1.6 Contribution of Research

The development of controller will be able to monitor the energy distribution in the house and the development of system that will be able to monitor the usage of unused energy (solar) and generator to charge and discharge the battery automatically. The development of system that can monitor energy distribution in the house as well as automatically charged and discharged the battery through web server using internet.

1.7 Report Outlines

This report includes of five chapters which Chapter 1 for Introduction, Chapter 2 for Literature Review, Chapter 3 for Methodology, Chapter 4 for Result and lastly Chapter 5 for discussion on whole research.

1. CHAPTER 1

Chapter 1 is related for Introduction, discussed more about project background, objectives on the research, problem statement, scope of the project and also summary of methodology.

2. CHAPTER 2

For Chapter 2, this chapter is included more about Literature review which is include review the previous work and also all material principle and mathematical theory use in this research.

3. CHAPTER 3

Then, Chapter 3 is more about methodology where flow of the process of research, software and hardware use in this project. Gantt chart and flow chart also included in Chapter 3 to show the flow of the research going.

4. CHAPTER 4

For chapter 4 which is the final result gain from this project. The result of the best analysis to charging and discharging process is by monitor on web-server. The prototype the smart hybrid with coding for between Adruino and Router (PHP language, Python Scripting, MySQL, and Tera term) also inserted in this chapter.

5. CHAPTER 5

Lastly, Chapter 5 included a conclusion for a whole project referred on database that collect from web interface and explain the achievement from the objective.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter discusses on literature review of previous research on smart hybrid automation and web-server. Beforehand, there a few functional component parts that are used to combine and develop the smart hybrid system. There were Photovaltaic system, rechargeable battery, solar charge controller, generator controller, Adruino, Router and web-server (PHP language, Python Scripting, MySQL, OpenWRT, and Tera Term). This method has been actualized much of the time in many cases in engineering and software engineering study. Most of the paper specified that the Adruino is very efficient and prove to be superior for almost all the test problems and also in many applications depends on different application and terms.

2.2 Theory and Basic Principles

2.2.1 Photovoltaic

Generally, PV comes from the Greek word where ―photo is for the meaning of light whereas ―volt defined from the electricity pioneer Alessandro Volta. Solar cell is the

other name for photovoltaic and it can convert light into energy. Besides, the photoelectric effect describes the absorption photons from light and the release of electron [1].

Photovoltaic cells are made from high-grade silicon (semiconductor) which can be electrically conductive when supplied with light or heat and operate as insulators at low temperature. A typical PV cells is build-up of an ultra-thin layer of phosphorus-doped (N-type) silicon and thicker layer of boron-doped (P-type) silicon. Moreover, P-type silicon contains fewer electrons whereas N-type silicon contains excess of electrons. P-N (positive-negative) junction is created when the N-type and P-type silicon contact together [2].

Besides that, the PV cells will absorb photon when being exposed to sun light and the electrons will be knocked loose from the atoms in the semiconductor material. The electrons will start traveling from the PV cells to the load. Apart from that, it will form an electric circuit when connecting a load on the positive and negative site on the PV cells. Furthermore, this electricity can be used to power up a load for such as a lighting system.

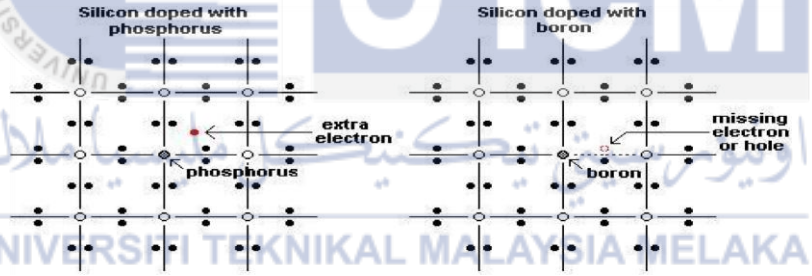


Figure 2.1: N-type silicon (phosphorous) and P-type Silicon (Boron)

A typical PV cells produces about 0.5-0.6 volt DC under open-circuit or no-load condition. Meanwhile, the current of the PV cells depends on its efficiency and the surface of the PV cells. The intensity of the sunlight striking the surface also affect the current (or power) from the PV cells, for example, the power will be higher under peak sunlight condition.

2.2.2 The Photovoltaic Effect

Photovoltaic effect is the process where the PV cells convert the sunlight into electricity. This process discovered by French physicist Edmund Becquerel in 1839 while studying an electrolytic cell with two metal electrodes. The sunlight will emit photon and this photon will cause the electron to move from the valence band to the conduction band. Electron in the n region will migrate back to the hole in the p region and this will cause a potential difference (current passed by) when connecting a load on the two region [3].

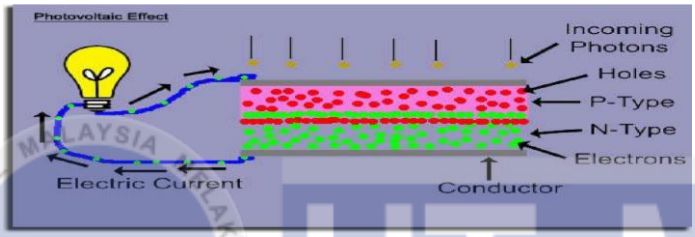


Figure 2.2: The Photovoltaic Effect

2.2.3 Rechargeable Battery

Battery is required for the stand-alone systems and the power generated by the solar panel is normally used to charge a lead-acid battery. Besides, nickel-cadmium batteries can be other option rather than lead-acid battery. However, the nickel-cadmium batteries are more expensive compared to the lead-acid type. A battery is made by individual cells and each cell can produce a voltage about 2 Volts DC. Therefore, a 12 Volt battery needs six cells. The capacity of a battery is measured in Ampere-hours or Amp-hours (Ah) and this mean that larger capacity of battery will have larger Ah.

Apart from that, the cycle life refers to the rate of a battery that can be discharged and this is important when using with solar cells. Solar application requires a battery that is capable to be discharged hundred or even thousands of times. This high rate of discharged battery is known as a deep-cycle battery [4]. For instance, car batteries are one of the most

common types of lead-acid battery, yet this battery can merely survive in 5 or 10 cycles and therefore this battery is not suitable for solar system.

Figure 2.3: GP 12V 7.2Ah Battery

2.2.4 Solar Charger Controller

Figure 2.4: Solar Charge Controller

A solar charger controller is a vital part of all power system that is used to charge batteries. This charger controller can protect batteries in solar system from being overcharged or being over-discharged and this can increase the battery life span. Moreover,

the basic function of a controller is quite simple which is to block reverse current and prevent battery from overcharging [5]. Apart from that, some controllers have the function to protect from electrical overload, display battery status, and prevent battery over discharge and the flow of power.

2.2.5 Generator Charger Controller

A generator charger controller is a second part of all power system that is used to charge batteries when battery is low. It uses to back up a power system. This charger controller also operates automatically from being overcharged or being over-discharged and this can increase the battery life span [6]. Moreover, the basic function of a controller is quite simple which is to block reverse current and prevent battery from overcharging. Apart from that, some controllers have the function to protect from electrical overload, display battery status, and prevent battery over discharge and the flow of power.

2.2.6 Arduino

Figure 2.5: Arduino UNO Board

Arduino is a popular open-source single-board microcontroller, descendant of the open source [7]. Wiring platform, designed to make the process of using electronics in multidisciplinary projects more accessible. The hardware consists of a simple open hardware design for the Arduino board with an Atmel AVR processor and on-board input/output support. The software consists of a standard programming language compiler and the boot loader that runs on the board [8].

Arduino hardware is programmed using a Wiring-based language (syntax and libraries), similar to C++ with some slight simplifications and modifications, and a Processing based integrated development environment [8].

An Arduino board consists of an 8-bit Atmel AVR microcontroller with complementary components to facilitate programming and incorporation into other circuits. An important aspect of the Arduino is the standard way that connectors are exposed, allowing the CPU board to be connected to a variety of interchangeable add-on modules known as shields. Official Arduinos have used the megaAVR series of chips, specifically the ATmega8, ATmega168, ATmega328, ATmega1280, and ATmega2560. A handful of other processors have been used by Arduino compatibles [8]. Most boards include a 5 volt linear regulator and a 16 MHz crystal oscillator (or ceramic resonator in some variants). An Arduino's microcontroller is also pre-programmed with a boot loader that simplifies uploading of programs to the on-chip flash memory, compared with other devices that typically need an external programmer.

Serial Arduino boards contain a simple inverter circuit to convert between RS-232level and TTL-level signals. Current Arduino boards are programmed via USB, implemented using USB-to-serial adapter chips such as the FTDI FT232. Some variants, such as the Arduino Mini and the unofficial Boarduino, use a detachable USB-to-serial adapter board or cable, Bluetooth or other methods. When used with traditional microcontroller tools instead of the Arduino IDE, standard AVR ISP programming is used [8].

The Arduino board exposes most of the microcontroller's I/O pins for use by other circuits. The Diecimila, now superseded by the Duemilanove, for example, provides 14 digital I/O pins, six of which can produce pulse-width modulated signals, and six analog inputs.

2.2.7 Monitoring (Internet web-based)

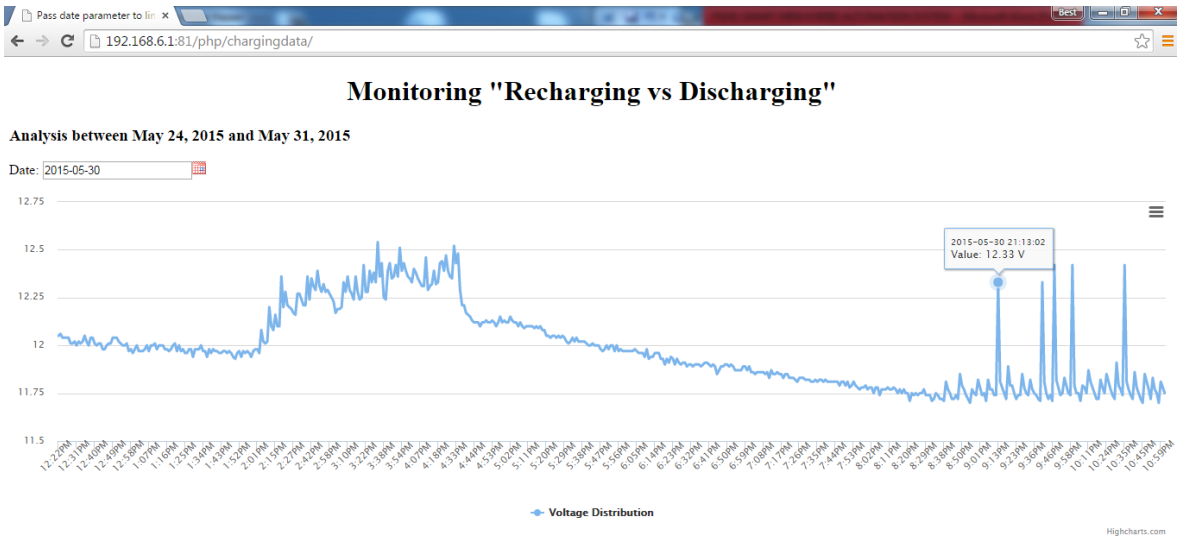


Figure 2.6: Web Interface

The function of Web-based equipment monitoring system is to collect real-time data of the on-site equipment, publish it through a Web form, and remotely send the data in the form of the user-defined data transmission style [9]. It will provide flexible remote monitoring and diagnosis function combining the configuration software based on standard browser. The data will be published through web page form by the web server in various user defined formats. If the parameter value is different from the original set value by the user, a corrected signal will be send to the control unit automatically by the processor.

With the growing popularity of Internet and the development of embedded technology, Web technology has been extended to the development and application of embedded system. It is the end of the network era which takes PC equipment as the basic network node. Besides PC equipment, the network nodes include various types of embedded devices. How to use Embedded and Web technology to perform remote monitoring, diagnosis, management, and controlling and maintenance operation of embedded devices from different subnets and physical areas, is a problem that needs to be solved. Embedded Web based equipment condition monitoring and controlling system directly connects the equipment to network as a node [10].

2.2.8 Router (Open WRT)

Figure 2.7: Router MR3020

OpenWrt [11] is targeted for networking embedded platforms and offers tremendous support for off-the-shelf wireless routers, e.g. Netgear, D-Link, Linksys, Asus and Thomson.

In [12], [13], the authors have demonstrated the benefits of OpenWrt, and used OpenWrt to validate their new designs and systems in real-time platforms. However, the authors have not discussed the internals of OpenWrt, the hardware description and the steps to build a testbed.

The router will function correctly when powered with voltages as low as 3.3 Volts (determined experimentally) instead of 5V USB-Power. Thus, it can be powered directly from one single Li-Ion battery (which usually starts fully charged at 4.2V and has a nominal voltage of 3.7) without the need for an external 5V adapter [15]. This router is standardly powered via USB at 5V. The voltage regulators' input voltage should be at least between 3.7V – 5.5V, but not over 5.5V. The device will get damaged at too high voltages. Maximum current draw at 5V is 255mA (Active Download + LAN + WLAN + USBboot), average current draw with WiFi is 125mA, idle is 68mA. Hence the average router power consumption is 0.6W, which is incredibly low [15].

2.3 Review of Previous Related Work

From Khusvinder Gill, Shuang-Hua Yang, Fang Yao, and Xin Lu. They proposed a ZigBee-Based Home Automation System [14]. A ZigBee based home automation system is implemented for the monitoring and control of household devices. To cater for the household's high data rate needs, such as multimedia entertainment, a Wi-Fi network is implemented. A home gateway has been developed to provide interoperability between these networks. The home gateway presents a unified interface for users to locally and remotely access home networks. The security and safety of the home automation network is realised through the development of the earlier described virtual home on the Home Gateway. To demonstrate the feasibility and effectiveness of the proposed system four devices, a light switch, radiator valve, safety sensor and ZigBee remote controller have been developed and integrated with the home automation system.

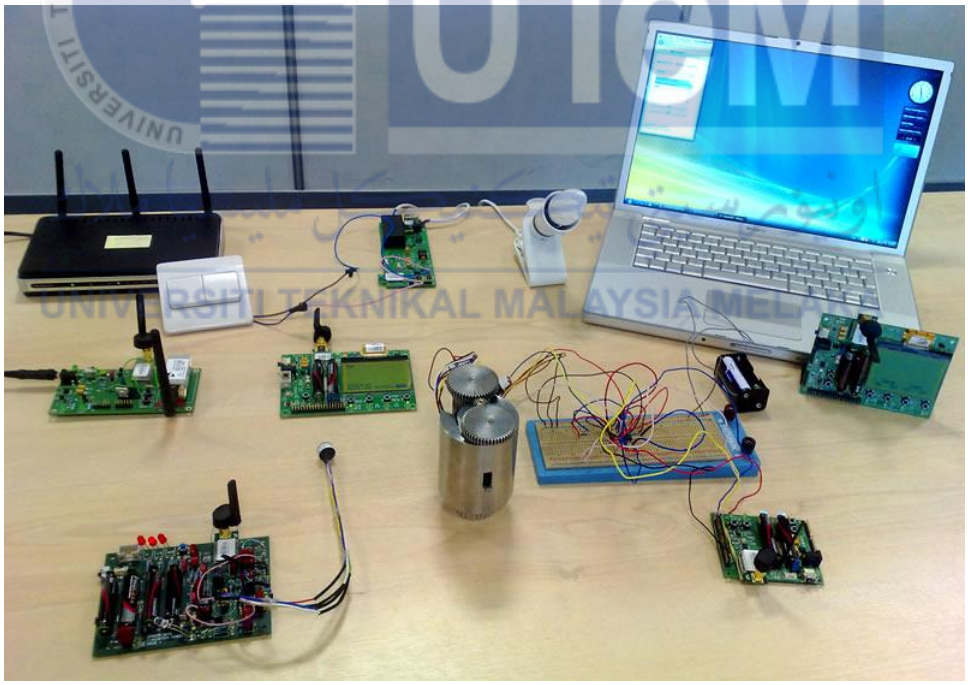


Figure 2.8: System Implementation

From R.Piyare and M.Tazil, they proposed a Bluetooth Based Home Automation System Using Cell Phone [15]. In this project, they present a low cost secure cell phone

based, flexible home automation system. Appliances at home are connected to the Arduino BT board. The communication between the cell phone and the Arduino BT board is wireless. Additional devices can be connected into the system with little modifications. Since the cell phone script is written in Python, it is portable and can run on any Symbian Operating System platform. This home automation system consists of two main hardware components: the cell phone and the Arduino BT board. The cell phone hosts the Python script which enables the user to access the home appliances and also the control commands for the appliances. This Python script communicates with the Arduino BT board and sets up an ad-hoc communication protocol between the two devices, which allows controlling the behaviour of the Arduino BT board.

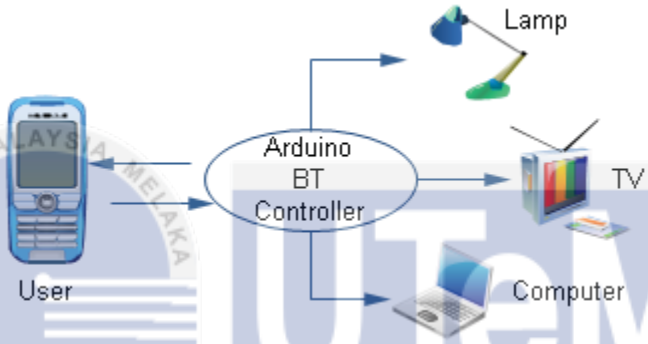


Figure 2.9: Block diagram of home automation system

From A. R. Al-Ali, Member, IEEE & M. AL-Rousan. They proposed Java-Based Home Automation System [16]. In this project, they presented an attractive low-cost solution for home automation via the Internet. Using Internet access, homeowners can remotely monitor and control almost any appliance at home. Local control is also offered in their design. Some security is imposed when logging into the system using Java Beans and Java Server Pages (JSPs). The core of the home automation system consists of two hardware components: the home server and the E-board. The home server is a high-end PC that hosts the Java-based management and control algorithm that enables the user to access the home appliances through the Internet. It also communicates with the E-board via the parallel port to download and upload the control commands and appliance's status. The E-board is connected to the home server via the parallel port. Using the E-board to control the appliances gives the user the ability to control the home locally even if the internet service is not available or even if the PC is not ON.

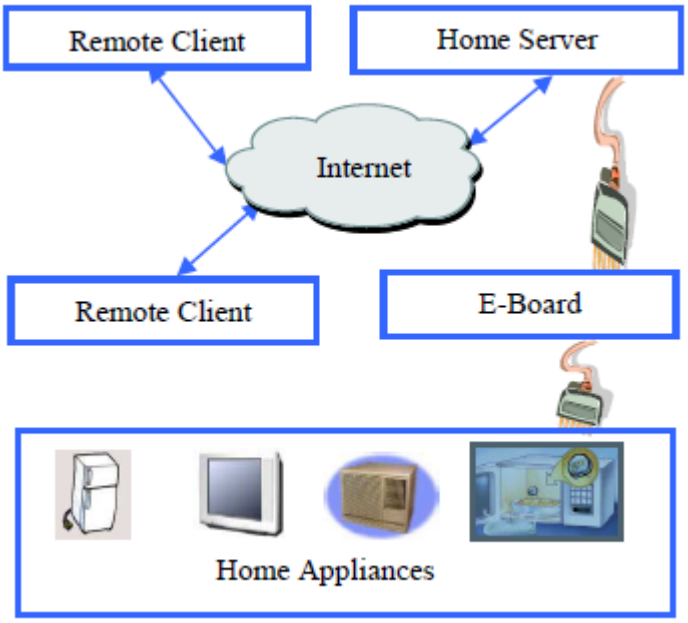


Figure 2.10: System hardware layout

From Zhi-Ming Lin, they proposed A Remote Telephone-Controlled Home Automation System [17]. In this paper they present a Home Automation System using telephone lines. The system consists of two subsystems. One is the Remote Control system. The other one is the Phone Monitoring system. The Remote Control system used the Dual Tone Multi-frequency signals to control the operations of various appliances. The hardware and software are designed based on the standard telephone system. The Phone Monitoring system provides convenient services for the user to better monitoring the usage of their phones. The hardware circuit of the remote control system includes a power supply circuit, a programmable logic chip 895 1, a multifrequency MT8870 chip and a multifrequency decoder.

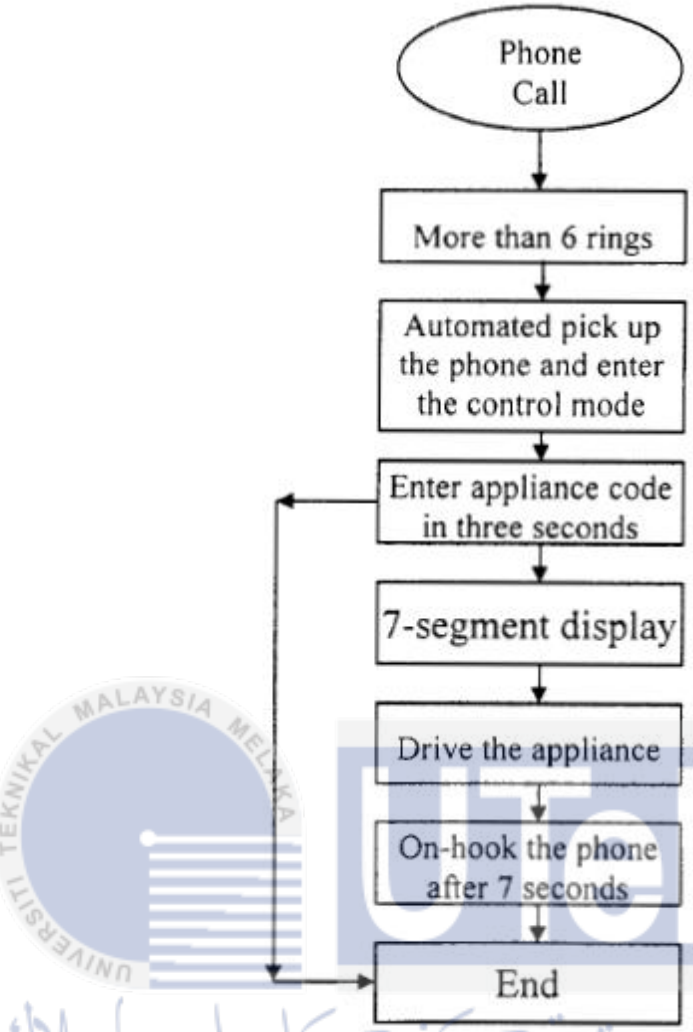


Figure 2.11: Flow chart of the remote control system [17]

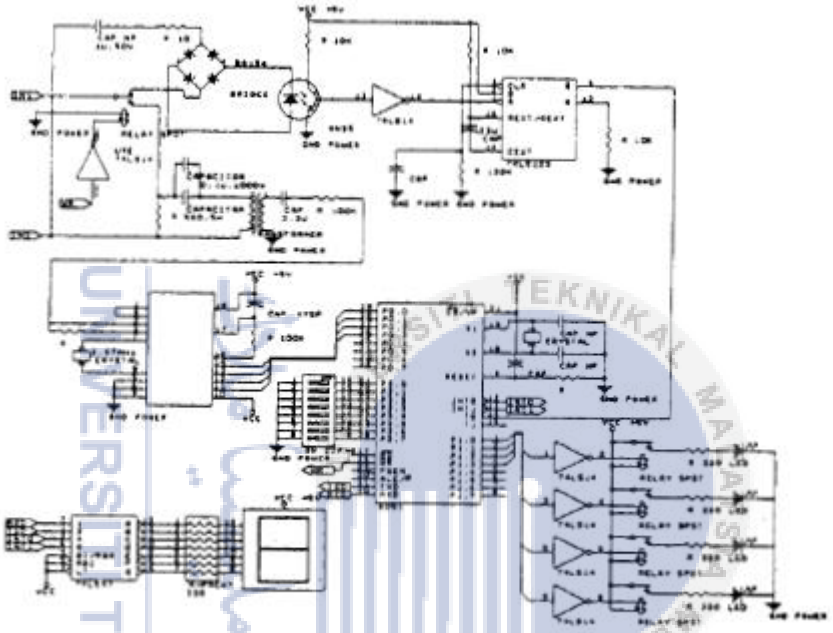


Figure 2.12: Hardware Circuit of the Remote Control System [18]

From Rozita Teymourzadeh, Ceng, Member IEEE/IET, Salah Addin Ahmed, Kok Wai Chan, and Mok Vee Hoong. They proposed Smart GSM Based Home Automation System [18]. This research work implements SMS based control for home appliances using the GSM architecture without accessing the local network. In the proposed system design, incoming SMS message is sent from the user phone to the GSM modem as a text message via cellular network. The GSM modem then sends the commands in text mode to the PIC microcontroller using an RS232 interface. The RS232 voltage levels are at ±12V whereas both the microcontroller input and output operates at 0V to +5V. Since RS232 is not compatible with microcontroller, MAX232 is utilized to enable the communication between both the GSM modem and PIC microcontroller by converting RS232 level signals to TTL level signal. Outgoing message from the system containing the home appliances status is delivered to the mobile phone through GSM modem.

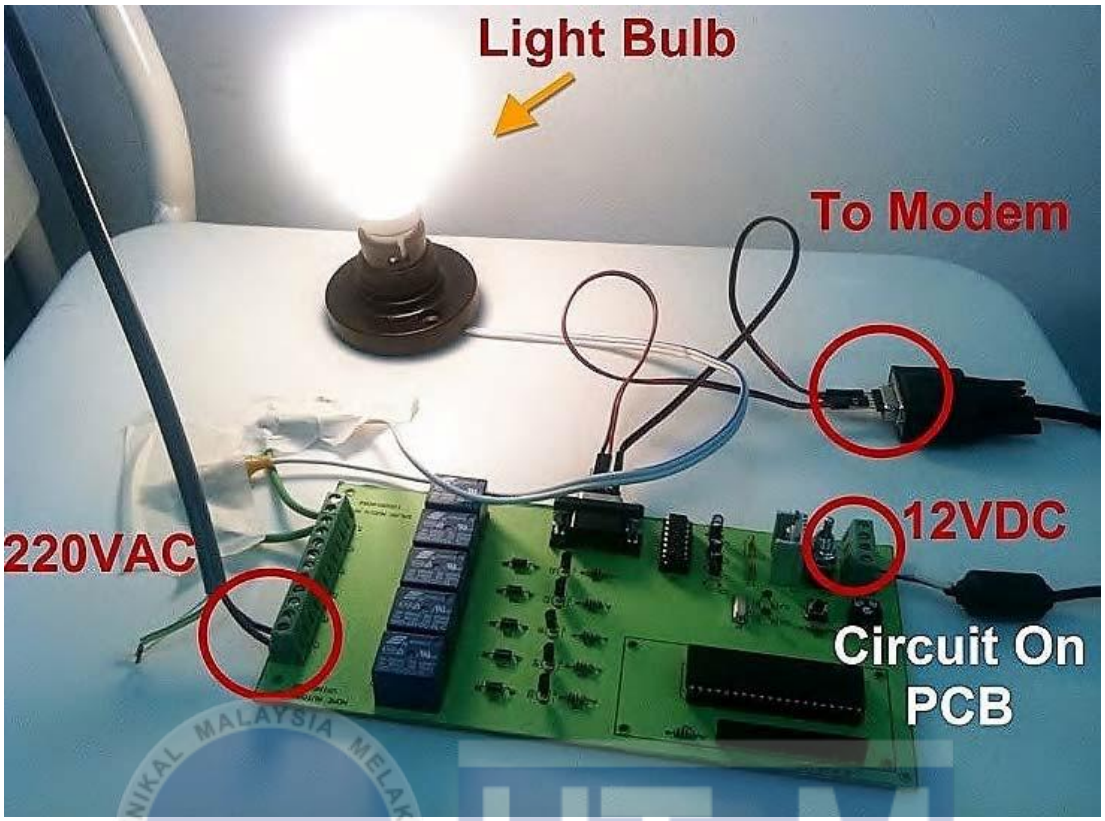


Figure 2.13: System Implementation [18]

2.4 Summary and Discussion of The Review

TITLE	WRITER	ADVANTAGES	DISADVANTAGES
Zigbee-based Home Automation System	Khusvinder Gill, Shuang-Hua Yang, Fang Yao, and Xin Lu	-Power saving -Reliability -Low cost of modules and Zigbee Protocol is patent free fee -Short time delay (30 ms for device searching, 15 ms for standby to	-Short range (10-100m line of sight) -low complexity -Low data speed

		activation, 15 ms for channel access of active devices) -Large network capacity -Safety	
Bluetooth-based Home Automation System using cell phone	R.Piyare, M.Tazil	-Easy to use -Can be used as a test bed for any appliances that require on-off switching application without any internet connection -Easy to carry since smartphone has Bluetooth system -Eliminates wire	-Relatively short range (<50m in a concrete building, maximum 100m in open range) -mediocre data transfer rate -incurring an access delay
Java-based Home Automation System	A. R. Al-Ali, Member, IEEE & M. AL-Rousan	-Use of Java technology, which incorporates built-in network security features produces a secure solution	-Requires an intrusive and expensive wired installation and the used of high end PC
Remote Telephone-Controlled Home Automation System	Zhi-Ming Lin	-Phone monitoring system can record the conversation on the phone - Circuit will not affect the performance of	-Users are not provided with a graphical user interface -Users have to remember an access code

		telephones - Helpful in lighting up the room for picking up the phone when a deep night call is coming	-Users have to remember which button to press for the control of connected device
Smart GSM based Home Automation System	Rozita Teymourzadeh, Ceng, Member IEEE/IET, Salah Addin Ahmed, Kok Wai Chan, and Mok Vee Hoong	-Easy to access and use -Portable -Has a global range -Low operational cost in general	-Delays in mobile network -Need of a mobile phone entirely dedicated for the system

All the previous studies and searches that related to this project are defined in this chapter. There are many researches from previous work about monitoring and controlling to improve the system more efficiency. All of the research can be as guidance and reference for this project. There is no previous research relate and same as my project. This is a new idea from me because mostly the function and the working principle of the home automation system are explained to improving and changing with monitoring by web interface. I will use an Adruino uno and router (OpenWRT) to change the ZigBee remote controllers. For additional, all information from theoretical and literature review will be analysed thoroughly as to be reference to complete this project.

CHAPTER 3

METHODOLOGY

3.1 Introduction

In this chapter of methodology, it discussed on how to carry out this study that is focused on how to identify the desired objectives through the selected tools. This chapter is to cover the methodology from the project planning, project flow chart, product case study and design the flow chart that to achieve the objectives.

3.2 Principle of the Method and Technique

First and foremost, the project is focusing on the controller development for energy distribution of hybrid house. Moreover, this project is collecting energy data from battery to show on web-server and automate the charging process by generator or solar panel and store the energy on battery. The controller distributes energy to all the equipment in the house.

Flowchart

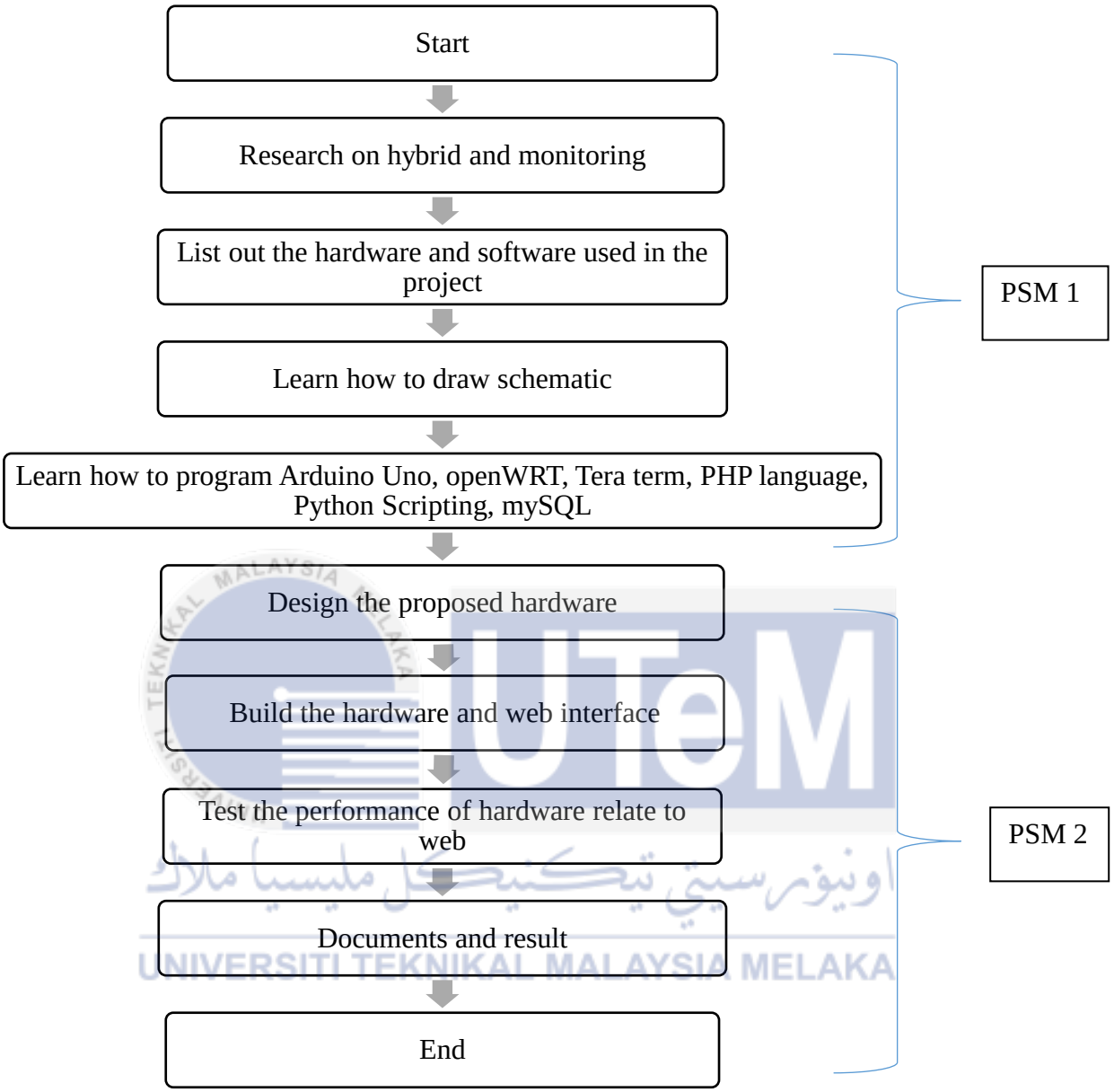


Figure 3.1: PSM Flow Chart

Figure 3.1 shows the PSM Flow Chart by stages.

At the first stage of this project, some previous studies have been referred and the components' datasheet are studied for extra understanding about the in developing the controller. Apart from that, the basic and fundamental circuit operate theory have been studied before developing the circuit is starting.

The construction of the system hardware is started once all the circuit is developed and all the components are purchased. Apart from that, the components include the electronic components for solar charge controller, low voltage disconnect controller, voltage indicator and analogue panel display. All the components are selected carefully since there are limited of budget.

The simulation has been done with the use of Proteus software. Furthermore, the circuits of the controller will be constructing on the project board and test the functioning. Then, the real time circuit results will be compare with the simulation results and troubleshooting process will be carry out when the results are different.

Apart from that, the low voltage disconnect controller is tested for the loads to disconnect when the voltage drop to the set point for protect battery lifespan. Moreover, the Arduino is developing using programming language and testing of software and hardware will be carrying out.

Part of web monitoring is developing by PHP language, Python Scripting, and Json for interface the data logger and graph. All information will be saving on MySQL database and it will be connect with hardware.

There are few steps or procedures which have been carried out in order to complete this project. The work flow is shown in Figure 3.2.

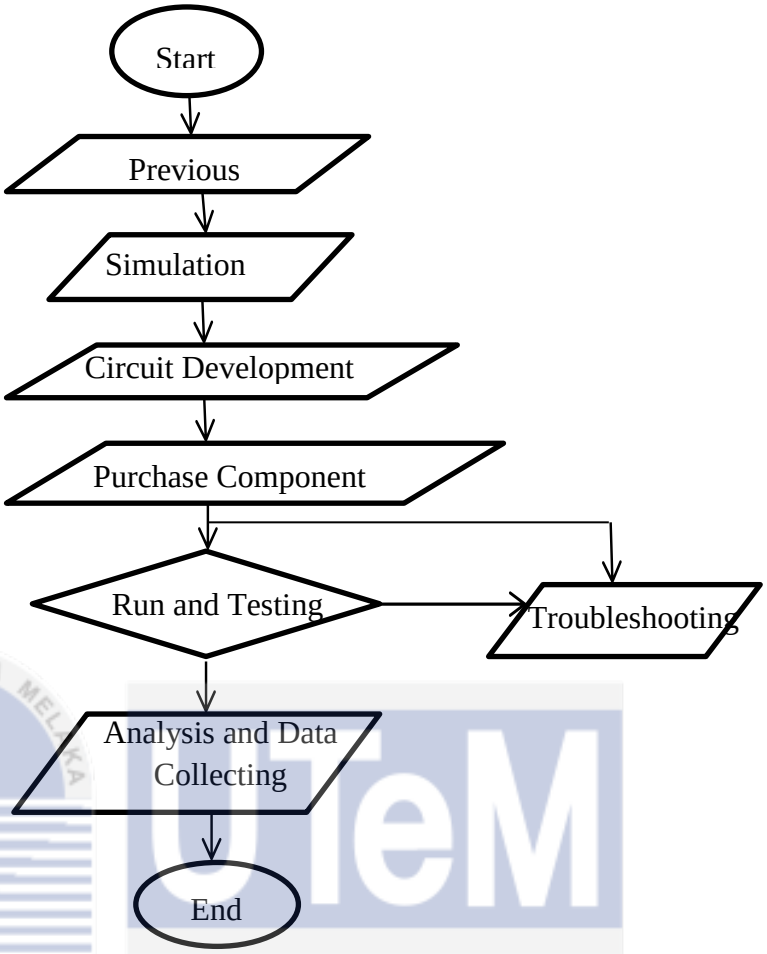


Figure 3.2: Work Flow

3.3 Project Operation

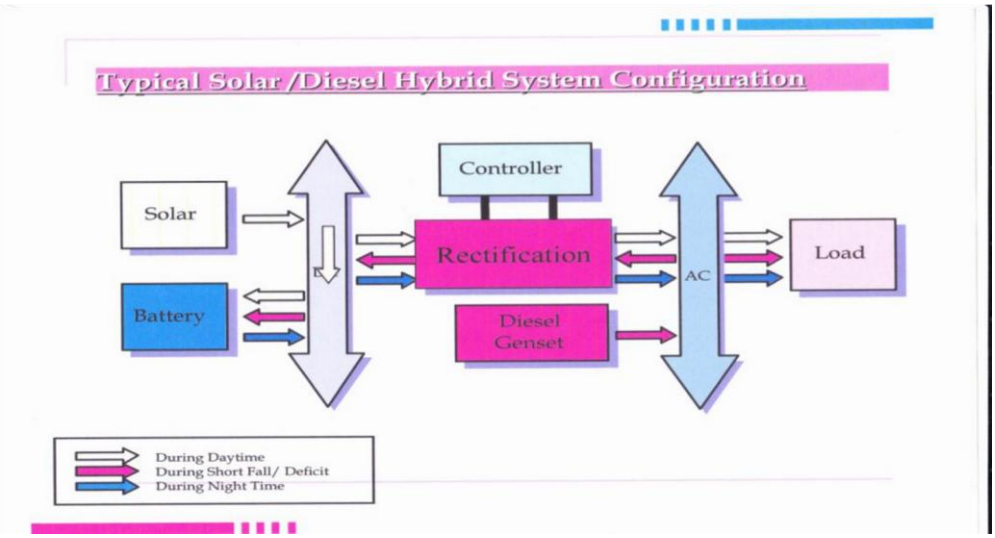


Figure 3.3: System Configuration

Figure 3.3 show the system derives most of its energy through the solar Photovoltaic (PV) array. The PV array supplies DC power to an inverter during the day and at the same time charges a bank of batteries. The inverter converts the DC power into AC at 120/240 Volts as required to operate appliances, tools, office equipment and more. At night time the batteries supply power to the inverter thereby providing reliable power 24 hours a day. A back-up generator is built in to the equipment shelter to charge the batteries during heavy energy usage or periods of cloudy weather.

3.4 Project Description

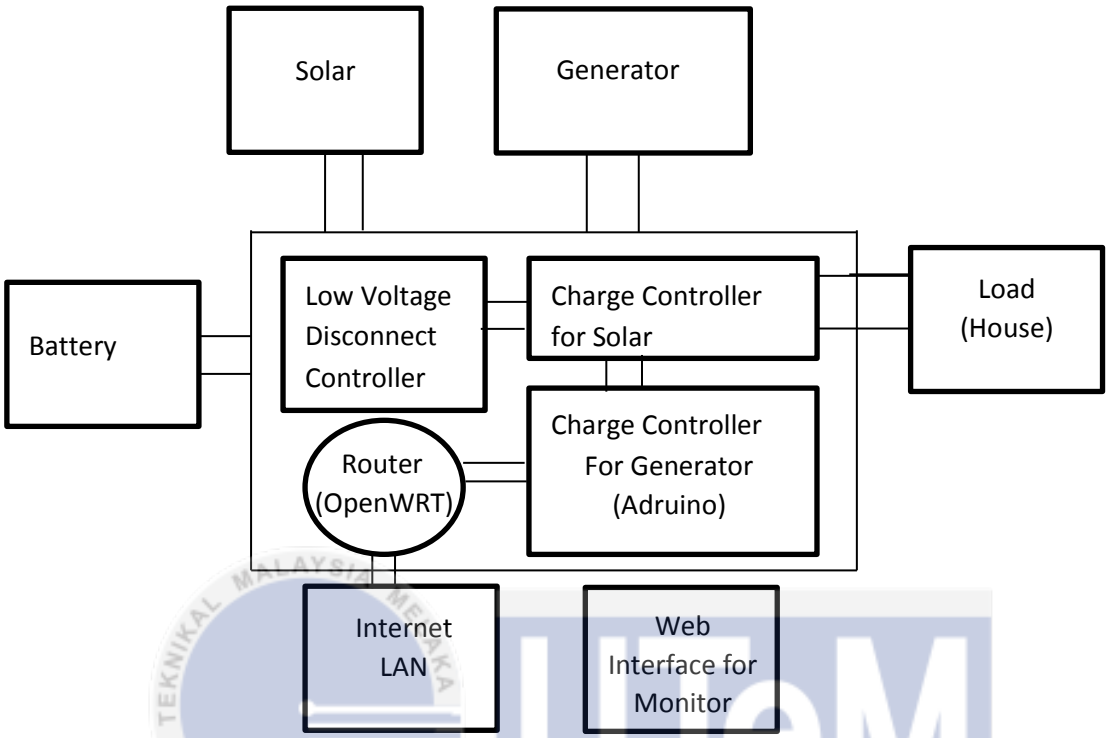


Figure 3.4: Project Overview

Figure 3.4 shows the project overview and this include solar panel, generator, battery, low voltage disconnect controller, charge controller for solar, charge controller for generator (Arduino), router (OpenWRT), internet LAN, web-interface for monitor, and load (equipment use in house). This project is separated into two parts which are hardware and software development. In line with this, the hardware parts include charge controller for solar and generator, and low voltage disconnect controller. On the other hand, the software focuses on the Proteus, Adruino programming, Tera Term, WinSCP, MySQL, and PHY language, Python scripting language.

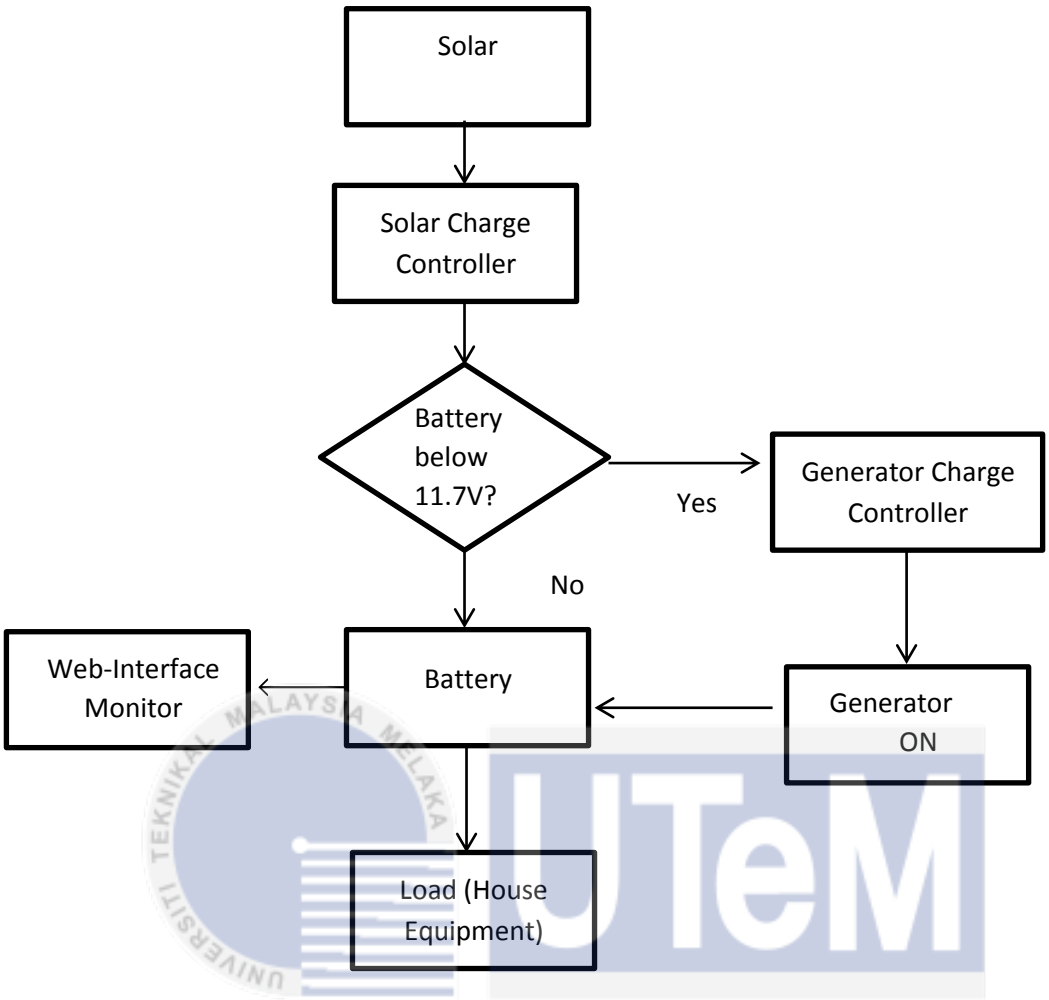


Figure 3.5: Flow Chart of Solar and Generator System

Figure 3.5 show that the charge controller is charging the battery when the battery voltage decreases to the minimum set point of voltage. Apart from that, the battery is supplied to the load or equipment used in the house. The charging process is controlled by the Arduino and voltage from solar and battery is sensed by Arduino for controlling process.

Furthermore, the Analog-to-Digital Converter (ADC) in the Adruino will sense the battery voltage and the loads are disconnecting from the controller when battery voltage decrease below 12V. The charge controller for generator will turn on when the battery voltage below than 11.7V. Then, the generator will start charging when the battery that drops below 11.7V. On the other hand, the generator stop charging when the battery voltage above 11.7V. The low voltage disconnect controller is totally shut off if the battery voltage drop below 11.5V.

Meanwhile, the voltage of the battery, VBAT and the voltage from the solar panel is sensed by the Arduino and display on web interface. Moreover, the charging process does not start when the battery voltage is above 13.2V. On the other hand, the charging process is started when the battery voltage drops below 13.2V. Each voltage value will be display on the web interface.

3.5 Requirement for Monitoring System

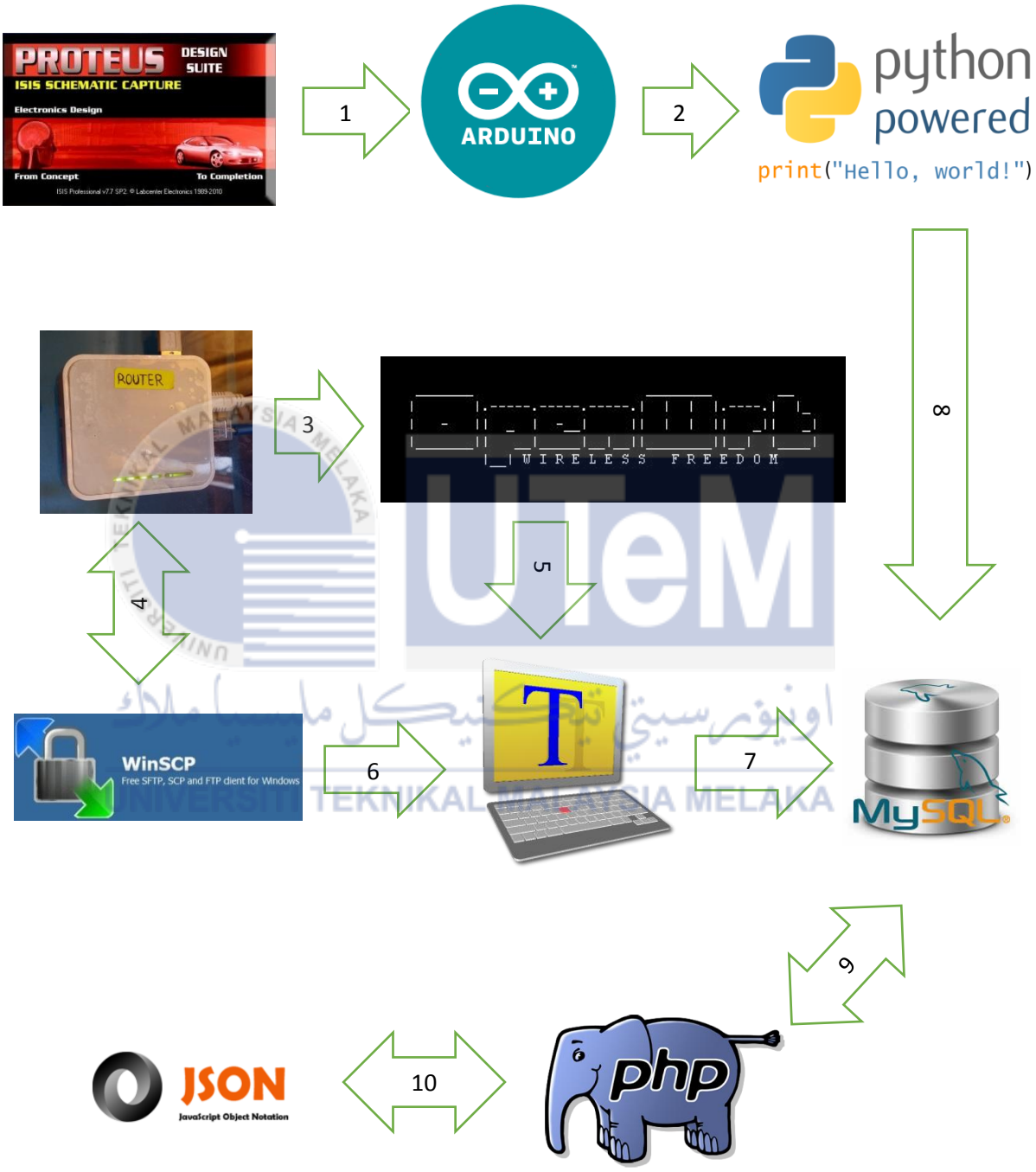
3.5.1 Hardware

1. Arduino
2. Router : Model Tplink MR3020

3.5.2 Software

1. OpenWRT: Operating system and installation monitor
2. MySQL: Database for data storage
3. Python Scripting: Develop programming serial interface between Arduino and router
4. PHP Language: Web based application for displaying data from database to web interface
5. Tera Term: Router configuration
6. WinSCP: File sharing, uploading, download, and editing

3.6 Project Design Stimulation and Software



3.6.1. Proteus Design

3.6.1.1 Solar Charge Controller

A solar charge controller is the fundamental equipment in a generator system that is used to charge the batteries. The purpose of charge controller is to protect the batteries from being overcharged or being over-discharged when used in solar system. Furthermore, this charge controller in the meantime increases the life span of the batteries and it works as simple blocks the reverse current and prevents battery from being overcharged.

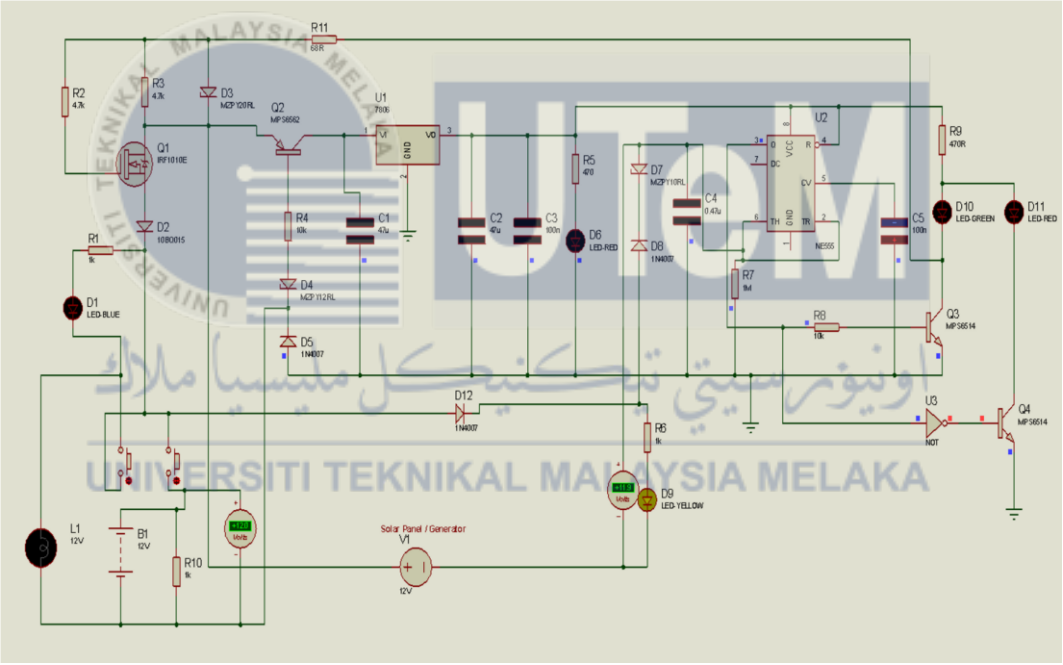


Figure 3.6: Solar Charge Controller Circuit in Proteus 7.1

Apart from that, this charging process is also using the fundamental charging theory that is ―ON‖ and ―OFF‖ charging. Furthermore, the transistor drives the gate of the MOSFET and active the relay for charging process. The charging process stops when the battery voltage increases to 13.5V controlling by using NE 555. The voltage regulator to maintain the voltage in 6V and the zener diode is to give a accurate voltage flow on it with clim if more voltage on it.

3.6.1.2 Generator Charge Controller

This project is using Arduino as the controller and there are eight (ACD) Convertor modules in this Arduino. Apart from that, the ADC module only can receive 5V and voltage divider is used for measuring the solar voltage and battery voltage. The ADC module in Arduino with 10 bits is producing 1024 steps. The calculation of the voltage divider and the ADC is use in this project.

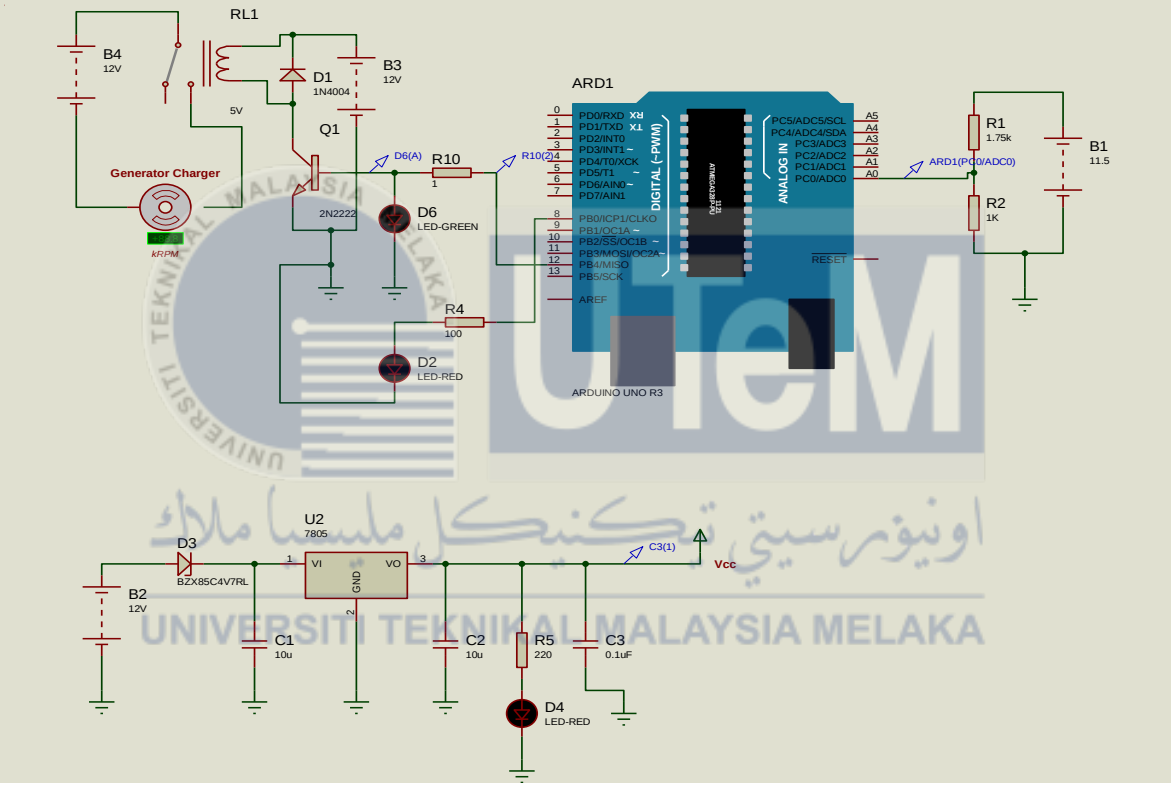


Figure 3.7: Generator Charge Controller Circuit in Proteus 7.1

The battery voltage decreases below 11.3V and the Adruino send signal to active transistor and active the charging process. Apart from that, the charging process stops when the battery reaches above 11.3V. There are two LED indicator which indicates that charging and not charging process. The RED LED indicated the charging process is started and the GREEN LED indicates that the battery is full.

Apart from that, this charging process is also using the fundamental charging theory that is ―ON‖ and ―OFF‖ charging. The Adruino receive signal from PORT analog A0 because only PORT A can read ADC. Then, Adruino sends signal from PORT D function to the transistor when the battery voltage decreases below 11.3V. Furthermore, the transistor drives the gate of active the relay for charging process. The charging process stops when the battery voltage increases to 11.3V.

3.6.1.3 Load Distribution

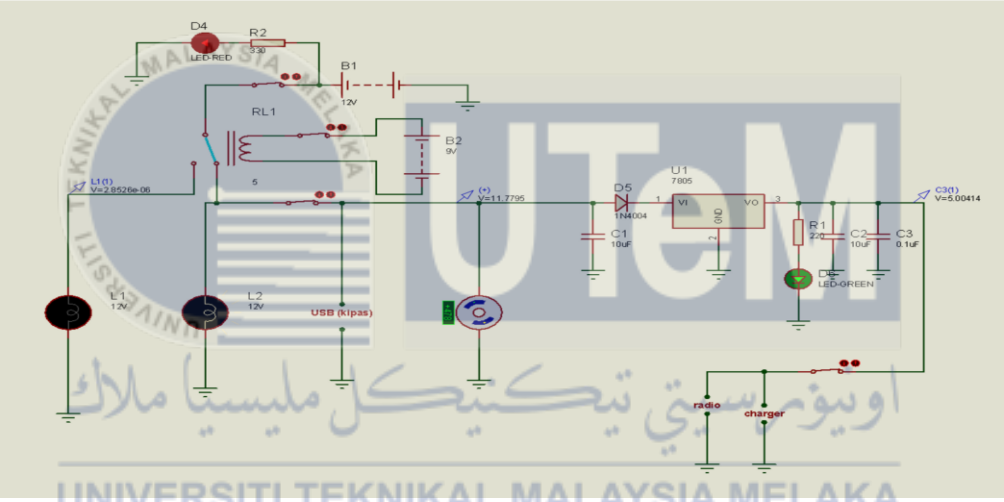


Figure 3.8: Load Distribution in Proteus 7.1

In Figure 3.8, the loads are separate into two distribute which is 12V and 5V equipment. Moreover, it also has USB port and portable charger. In order to make LED D$ is on, the current must flow to thought it. Then, when green LED turns on to give signal that current flow to regulator and ready to use. Apart from that, the equipment will be used are fan, bulb, USB port, radio, and portable charger.

3.6.2 Adruino Code for Generator

```
49
50  const int threshold = 858;   // 4.25V an arbitrary threshold level that's in the range of the analog input
51  // if the analog value is high enough, turn on the GENERATOR:
52  if (analogValue < threshold) {
53    digitalWrite(generator, HIGH); //genarator
54    digitalWrite(solar, LOW);
55    delay (100000);
56  }
57  else {
58    digitalWrite(generator, LOW); //solar
59   digitalWrite(solar, HIGH);
60
```

Figure 3.9: Adruino Coding

The threshold means that binary number for input analogue has been lock. 858 binary is same with 11.7V. As theoretically, charging process using generator need 5.4 minute to make the bank battery full because the generator that is use only produces 2A output current and the specific battery is 7.2AH. By observation, delay (100 millisecond) is used to the generator charge because every 100 second can charge the battery up to 0.15V. The mathematic calculation shows on figure 3.10.

12V = 7.2AH 11.7V = 7.02AH 7.2 AH – 7.02AH = 0.18AH
0.18 AH x 60 minutes = 10.8 minutes $\frac{10.8}{2}$ = 5.4 minutes. 5.4 x 60 second = 324000 millisecond

Figure 3.10: Calculation for Delay

3.6.2.1 Mathematical Equation

A voltage divider is used to control the value of input and value of voltage when charge and discharge. ADC calculation, 5V is equal to 1024 steps and therefore Vfull is above to 858 steps and Vcharge is below to 858 steps. 858 steps are equal to 4.23V.

The voltage divider for battery full, Vfull = 4.23V = R2/R1+R2 (Vcharge)

Using R1 = 1.770 K ohm

Using R2 = 1 K ohm

$$\text{Vcharge} = 4.23 \times \frac{1k + 1.77k}{1k}$$

Vcharge = 11.7V

Figure 3.11: Mathematical Voltage Divider

3.6.3 OpenWRT

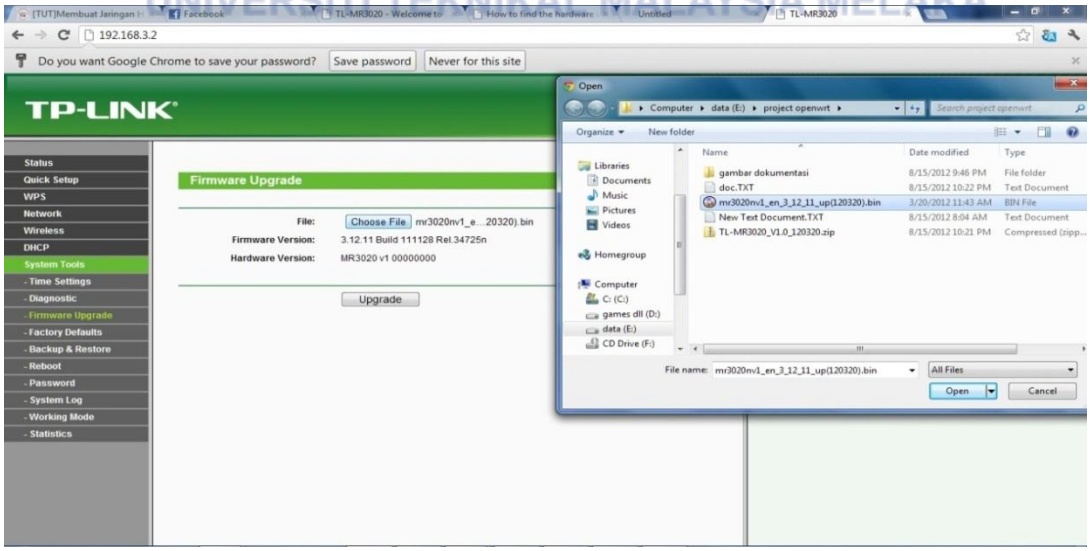


Figure 3.12: Installation OpenWRT

OpenWRT is firmware that upgrade to replace the original operating system OS. These have one ways to install OpenWRT in the devices. The method is via Original Equipment Manufacturer (OEM) firmware. The steps are we open the website of OEM firmware with the web browser and install the OpenWRT factory firmware image file using “Firmware Upgrade” option. Then it finished. However not all devices support this function.

Open WRT framework can be configured in three distinctive ways. The principal is the configuration by means of graphical interface. To arrange the framework through a graphical interface one should utilize any web program. An alternate approach to arrange the working framework is through arranging the command line (terminal). The third path is to change OpenWRT framework arrangement records, which are put away *in/etc../config/directory.* To arrange the framework it is conceivable to utilize inherent supervisor.

3.6.4 WinSCP

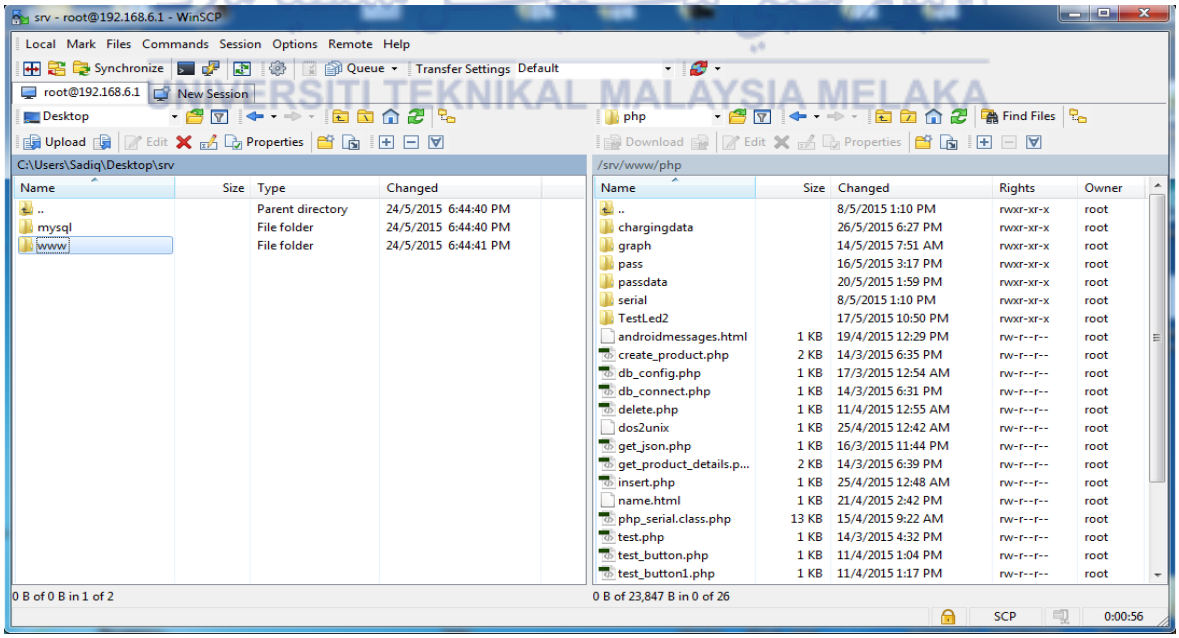


Figure 3.13: WinSCP File Manager

WinSCP (Windows Secure CoPy) is a free and open-source SFTP, SCP and FTP client for Microsoft Windows. Its main function is secure file transfer between a local and a remote computer. Beyond this, WinSCP offers basic file manager and file synchronization functionality. For secure transfers, it uses Secure Shell (SSH) and supports the SCP protocol in addition to SFTP. The SCP protocol use to link IP address router by open the file. The size memory on router has added by flash drive 8GB for support the file manager.

3.6.5 Tera Term

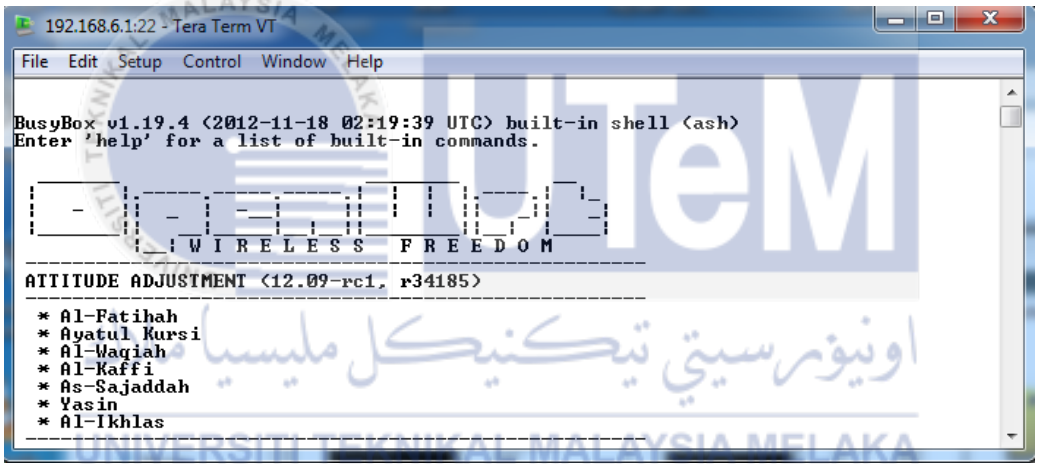


Figure 3.14: OpenWRT Interface

Tera Term (rarely TeraTerm) have be install to use as an open-source, free, software implemented, terminal emulator (communications) program. It emulates different types of computer terminals, from DEC VT100 to DEC VT382. It supports telnet, SSH1 & 2 and serial port connections. It also has a built-in macro scripting language (supporting Oniguruma regular expressions) and a few other useful plugins. For any configurations do it by program in tera term.

3.6.6 Code Tera Term Connect with WinSCP

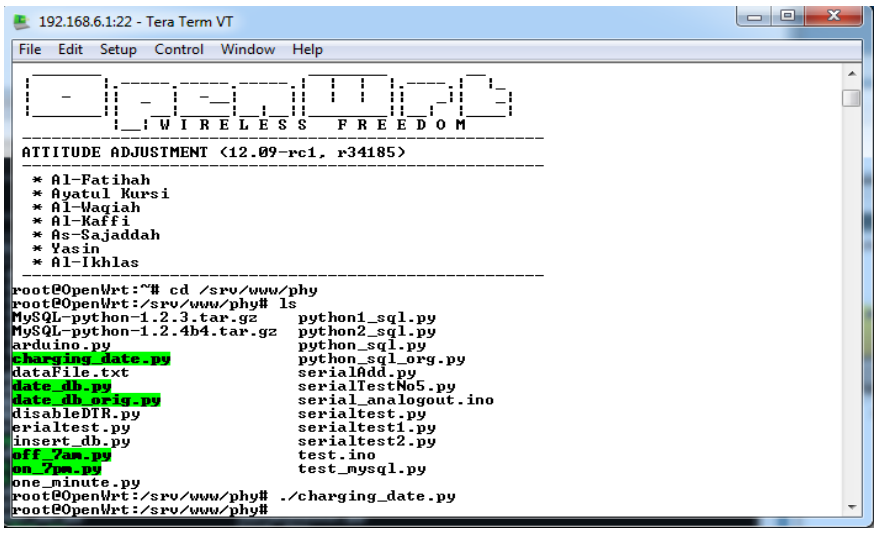


Figure 3.15: OpenWRT List Document

From figure 3.15, tera term can be access to file manager on router that be install by using WinSCP. Any file python can be active by select the directory as ‘charging_date.py’. The process capture and collecting data by python from input will be stored in the MySQL database. Python program act as the middle layer, which will bridge the sensor information coming from Arduino with the MySQL database.

3.6.7 Code Tera Term To Create Database MySQL

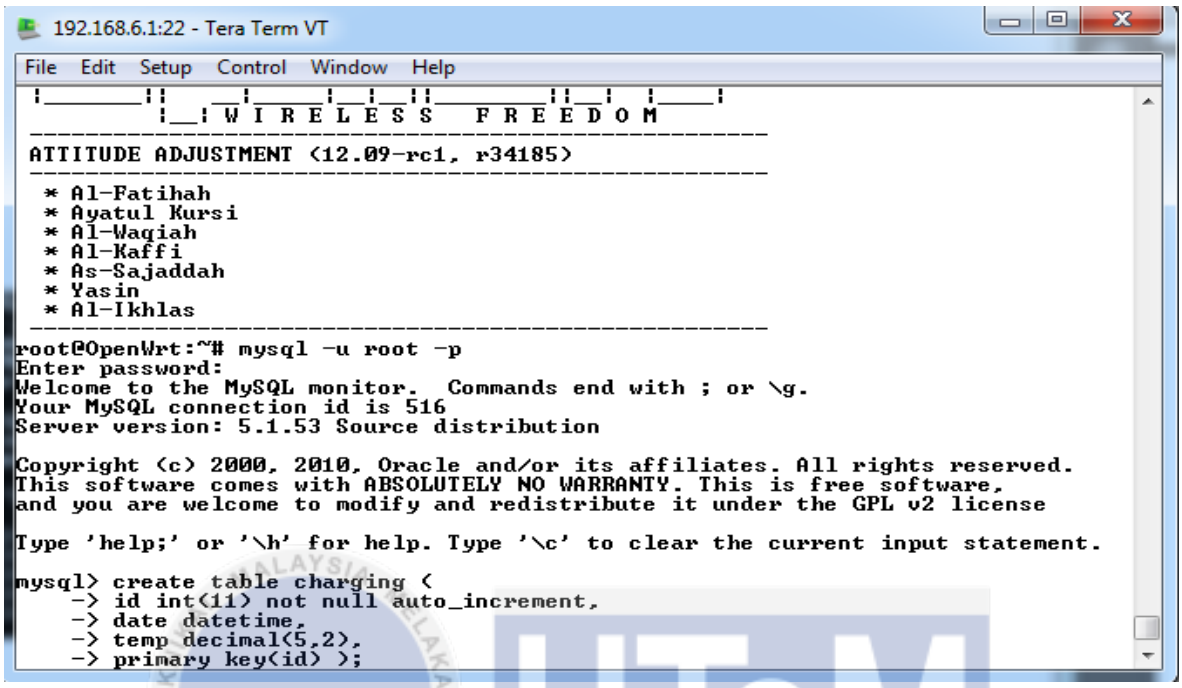


Figure 3.16: Coding Create Table in MySQL

A database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible PROGRAMMING environment. From here, the table database must be created first before sending any data. The step to create table create by using tera term code shows on figure 3.16. *Auto_increment* function as numbering on data by timing have been set. MySQL database can be store for 1 month and auto delete to sustain the collecting data.

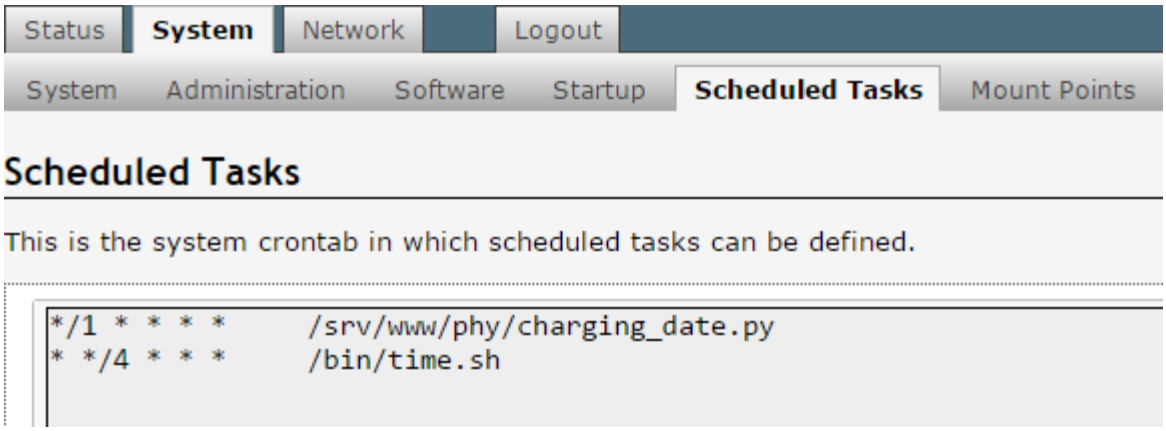


Figure 3.17: System Scheduled Tasks

This figure 3.17 shows the time schedule to log data every one minute by using python scripting language (charging_date.py).

3.6.8 Python Scripting

3.6.8.1 Python in Adruino

```
int val = 0;       // variable to store the data from the serial port
int sensorValue = 0;        // value read from the pot
int outputValue = 0;

if (Serial.available()) {
    val = Serial.read();      // read the serial port
    if (val > '0' && val <= '9' ) {
      Serial.print(val);

        digitalWrite(solar,HIGH);
        delay(150);
        // read the analog in value:
        sensorValue = analogRead(analogPin);
        String str = String(sensorValue);
        Serial.print("  \t  ");
        Serial.println(str);
      }
  // print the analog value:
  Serial.println(analogValue);
  delay(1);        // delay in between reads for stability
}
}
```

Figure 3.18: Coding Adruino for Python

This figure 3.18 shows the python scripting in Adruino is to relate and link with file 'charging_date.py'. *val* as a *serial.read* to read analogue value and sense by *string str*. The data transmit by *serial.printIn.* Among the common computer to hardware interfacing methods, serial-port based communication is the most popular, and it is really simple to establish, especially for the Arduino platform. Python provides a library called *pySerial* that is really easy to use and quick to implement to interface a serial port. It is really simple to use similar libraries and Python's interactive programming abilities to rapidly test and implement the project ideas.

3.5.8.2 Python in PHP

```
#send the data#while True:
s.write("5")
data = s.readline()
#print data #print s.readline() # Read the newest output from the Arduino
pieces = data.split("\t")  #split the data by the tab
sensor1 = pieces[0]
sensor2 = (float(pieces[1]) /1024) * 12 #read the data and times 12V
sensor2_2dec = float("{0:.3f}".format(sensor2))

now = datetime.datetime.now()
f = '%Y-%m-%d %H:%M:%S'
dt = now.strftime(f)

try:
    con = MySQLdb.connect('127.0.0.1', 'root', 'qwe123', 'monitoring')
    cursor=con.cursor()
    sqlstr="INSERT INTO charging (date,voltage) VALUES(%s, %s)", (dt,sensor2_2dec) #send data to MySQL directory
    cursor.execute("INSERT INTO charging (date,voltage) VALUES (%s,%s)", (dt, sensor2_2dec))
    #print  sqlstr #for debugging
    con.commit()
    cursor.close()
```

Figure 3.19: Coding PHP for Python

Figure 3.19 shows the scripting take over the process on transfer the collecting data from Adruino to database MySQL. The sensor use formula *(float(piece[1])/1024)*12* to process the binary number 1024 = 5V times 12V as a same real value battery. Battery use voltage divider to reduce from 12V to 5V for sense in Adruino and back to an actual value by use this formula. Code *sqlstr* is use for direction table to store the data. Python program act as the middle layer, which will bridge the sensor information coming from Arduino with the MySQL database.

3.6.9 PHP Language for Monitoring

PHP (recursive acronym for PHP: Hypertext Preprocessor) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML. The PHP Programming used in this project to create a web interface to monitoring charging and discharging process on hybrid home.

3.6.10 PHP Language for Json

```
8  mysql_select_db("monitoring", $con);
9  //$date = DateTime::createFromFormat('dd-mm-yy', $_GET["dateParam"]);
10 if (isset($_GET["dateParam"])) {
11
12     $sql = mysql_query("SELECT date, voltage FROM charging WHERE date LIKE '".$_GET["dateParam"]."%'");
13     #$sql = mysql_query("SELECT date, voltage FROM charging WHERE date LIKE '".$_GET["dateParam"]."%'");
14
15 } else {
16     $sql = mysql_query("SELECT date, voltage FROM charging WHERE date LIKE '2015-05-25%'");
17 }
18 $result['name'] = 'Voltage Distribution';
19 while($r = mysql_fetch_array($sql)) {
20     $datetime = $r['date'];
21     $result['category'][] = $datetime;
22     $result['data'][] = $r['voltage'];
23 }
24
25 print json_encode($result, JSON_NUMERIC_CHECK);
26
27 mysql_close($con);
28 ?>
```

Figure 3.20: Coding PHP for Python

JSON is Java Scripting. Figure 3.20 shows the fragment of the codes for JSON output. JSON is the middle of 2 parts of PHP language and MySQL database. This JSON to connect the data from MySQL and sent to PHP for develop the graph on web-server. As a code *$sql = mysql_query* is select data from table MySQL depend on date and voltage. *Print json_encode* as a function. From the graph we can analysis real-time data and monitoring the process.

3.7 Project Hardware Implementation

To accomplish any project, many procedures have to take care to make the project goes smooth and follow the compatibility of the circuit that will produce.

Below are the processes that been followed:

I. Election for circuit that will be use

II. Study the circuit

III. Tracing for PCB board

IV. Drilling process

V. Installation component process

VI. Soldering process

VII. Cutting process

VIII. Testing

3.7.1 Process to Make Circuit On The SB (strip board).

To make a ―SB‖ , there are 5 process that important to do. After the ―SB‖ is done, the ―SB‖ will be placed to the casing.

Below are the steps to make a ―SB‖ :

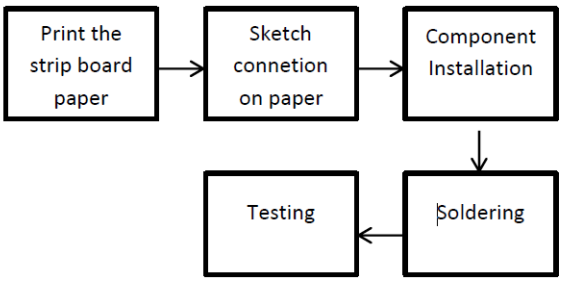


Figure 3.21: Circuit Process

3.7.2 Soldering Procedure

I. Make sure the surface to be solder is clean, dry, and no grease and flat.

II. Make sure the solder is in clean condition.

III. Make sure the heat of the solder is enough to solder.

IV. Heat up first the part to be solder.

V. Then, touch the lead to the connection part and cover the entire connection.

VI. Finally, lift the lead soldering and let it dry.

3.7.3 Project Flow Chart for Processing

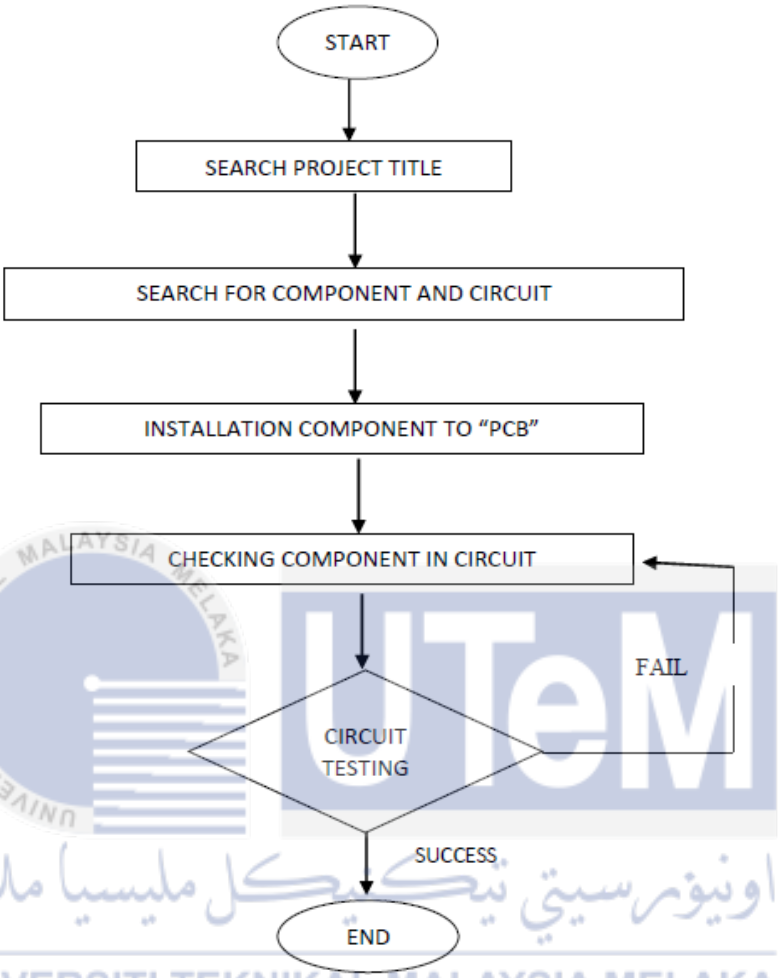


Figure 3.22: Flow Chart Process Implants

3.7.4 Controller Allocation

Figure 3.23: Allocation Hardware

All the circuits and control are arranged as shown in the Figure 3.22. Furthermore, there are switch, relay control circuit, analogue meter panel display, voltage indicator, usb hub, router, generator, battery, solar panel and Adruino. Apart from that, the 12V supply for load is allocated on upper of the low voltage disconnect controller for easier connection. All part have be connect each other to complete process of this project.

3.7.5 Controller Combine Testing

The controller is combined for a few days non-stop with hybrid controller and monitoring system controller. The connection controller is connected to 12V 7.2Ah battery without any charging process. If the battery below 11.7V, the generator will turn on and charge the battery.

3.7.6 Controller Testing With Solar Panel

The controller is tested by connecting a 20W solar panel and a 12V 7.2Ah battery. Apart from that, Figure 3.22 show the connection of the PV, the controller and the battery and the system is tested outdoor in the morning session. Furthermore, the battery is clamped with a millimetre for read the battery voltage. The charging process is started when the battery voltage is below 13.5V and stopped charging when the battery voltage increases above 13.5V.

3.8 Gantt Chart and Project Milestone

PSM 1

Table 3.1: Gantt Chart for PSM 1

Task/Week	1	2	3	4	5	6	7	8	9	10	11	12	13
Deal with Supervisor and Confirmation of Title													
Research about hybrid and monitoring													
Research about openWRT, Tera term, and Mongoose													
Research about Arduino Uno													
Write Progress Report													
List out and find the hardware and software used													
Develop coding for arduino Uno and installing OpenWrt													
Design Proposed hardware													
Presentation Project Proposal and PSM Seminar													

Table 3.2: Project Milestone

Task	Date
Deal with Supervisor and Confirmation Project Title	08/09/2014 – 24/09/2014
Develop Arduino Coding and Install OpenWrt	23/10/2014
Design Schematic Proposed Hardware	31/10/2014
Seminar and Presentation Project Proposal	26/11/2014
Due Date for Send Full Report PSM I	17/12/2014

PSM 2

Table 3.1: Gantt Chart for PSM 1

Task/Week	1	2	3	4	5	6	7	8	9	10	11	12	13
Deal with Supervisor and Start Program													
Program Database using MySQL													
Program python to collect data													
Setup Adruino and link with python													
Write Progress Report													
List out and develop the hardware													
Develop coding PHP for interface web													
Collect data and analyses													
Presentation Project Proposal and PSM Seminar													

Table 3.2: Project Milestone

Task	Date
Deal with Supervisor and Start Program	015/01/2015 – 28/03/2015
Develop Hardware and Link Program with Web Interface	2/04/2015
Collect Data and Analyses	15/05/2015
Seminar and Presentation Project Proposal	09/06/2015
Due Date for Send Full Report PSM II	01/06/2015

CHAPTER 4

RESULT AND DISCUSSION

4.1 Introduction

First and foremost, this chapter will display all the simulation results that have been carried out. Moreover, the hardware of voltage indicator, hybrid controller, charger controller, and low voltage disconnect controller are included in this chapter. The simulation results include the solar charge controller, generator charge controller, load, monitoring system, webserver, and database storage. The house prototype is also included in this chapter.

4.2 Project Achievement

4.2.1 Solar Charge Controller Proteus 7.1 Simulation Result

The circuit is constructed as in Figure 4.1 and the battery is set at full voltage at 13.5V. Furthermore, the LED is connected to transistor and used to indicate the charging and discharging operation. The simulation shows the YELLOW LED is on and this indicates that the current has follow on the circuit and the charging or discharging process has operate.

Moreover, the battery is set at 12V and the circuit is shown in Figure 4.1. The simulation shows that the BLUE LED D1 will blink to show the low voltage and charging process at RED LED D10 is ON to show the controller connects to battery. Apart from that, the LED RED D11 will ON when battery is full and the LED RED D6 is ON when the controller connects and supplies energy to load.

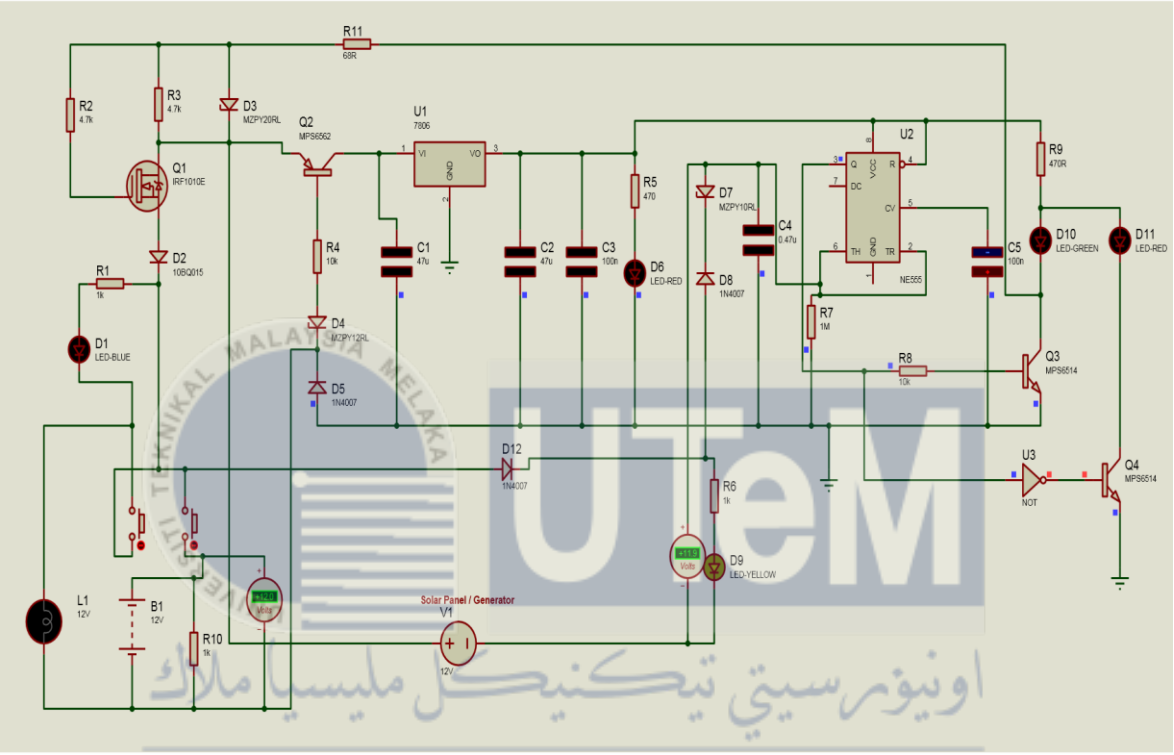


Figure 4.1: Battery Charging and Discharging Circuit

4.2.2 Solar Charge Controller Hardware Construction

4.2.2.1 Low Voltage Disconnect Controller

The charge controller is constructed on the PCB same with the low voltage disconnect controller as shown in Figure 4.2. Furthermore, the circuit is tested before constructing on PCB. The GREEN LED at solar display is on which indicates that the

charging process is carried on. Apart from that, the low voltage disconnect controller on the PCB is tested with 12V DC supply module before testing with the battery. The circuit connection is shown in Figure 4.2.

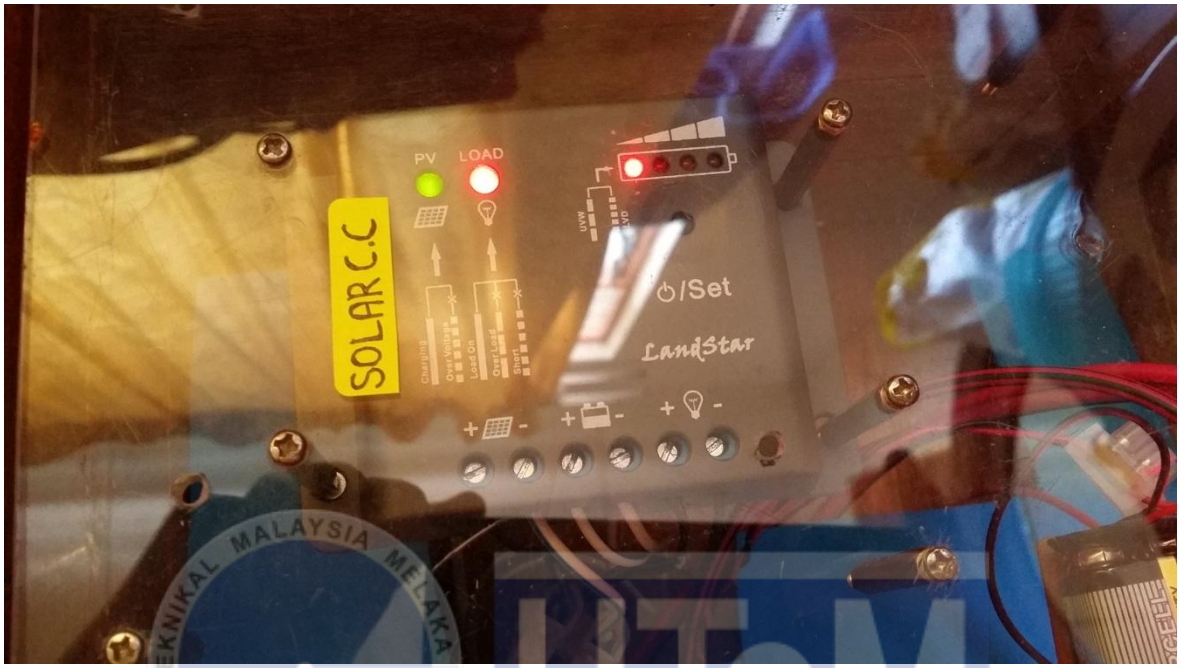


Figure 4.2: Solar Charge Controller

Moreover, the extension circuit of load disconnect controller and controller cut off circuit is shown in Figure 4.2. The load to disconnect when battery voltage dropped to 11.5V and this same for controller cut off circuit. Furthermore, the load can be disconnected with the switch and the battery low voltage blinker. The RED LED at load display is blinked when the battery voltage dropped to 11.6V and it stop when the battery voltage increases above 11.6V.

4.2.2.2 Voltage Indicator Hardware Construction

The voltage indicator is constructed on the PCB with the low voltage disconnect controller as shown in Figure 4.3. Meanwhile, the Figure 4.3 shown the voltage indicator on the hardware. On the other hand, it shows all the LED will light up when the voltage increase to 24V as shown the maximum voltage.

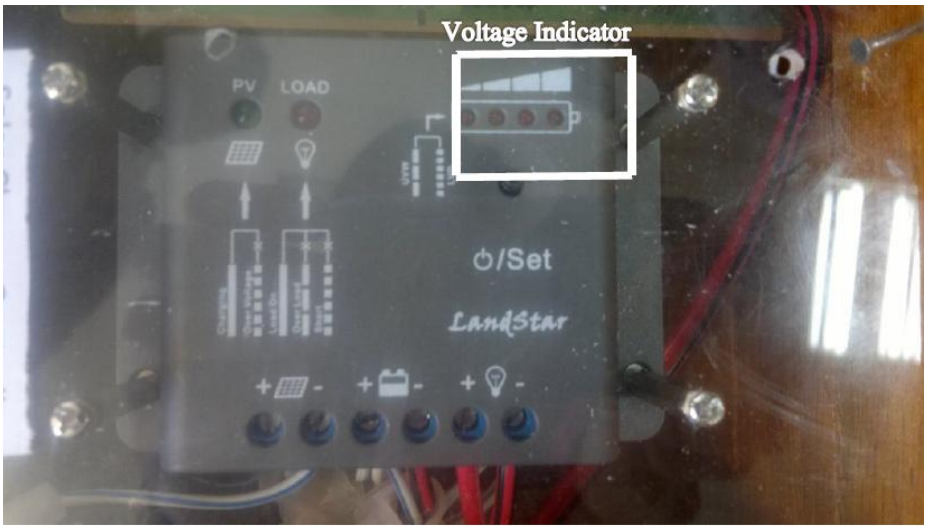


Figure 4.3 : Voltage Indicator On Low Voltage Disconnect

4.3 Generator Charge Controller Proteus 7.1 Simulation Result

The circuit is constructed as in Figure 4.4 and the battery is set at voltage at 11.8V. Furthermore, the RED LED D2 is connected to relay and used to indicate the relay operation. The simulation shows the RED LED D2 is on and this indicates that the battery in a safe condition to operate at a load and the charging process has stopped.

Moreover, the battery is set at 11.5V and the circuit is shown in Figure 4.5. The simulation shows that the GREEN LED D6 is on. Apart from that, it shows that the battery is being charged by generator and the GREEN LED D6 indicates that the charging process is carried on.

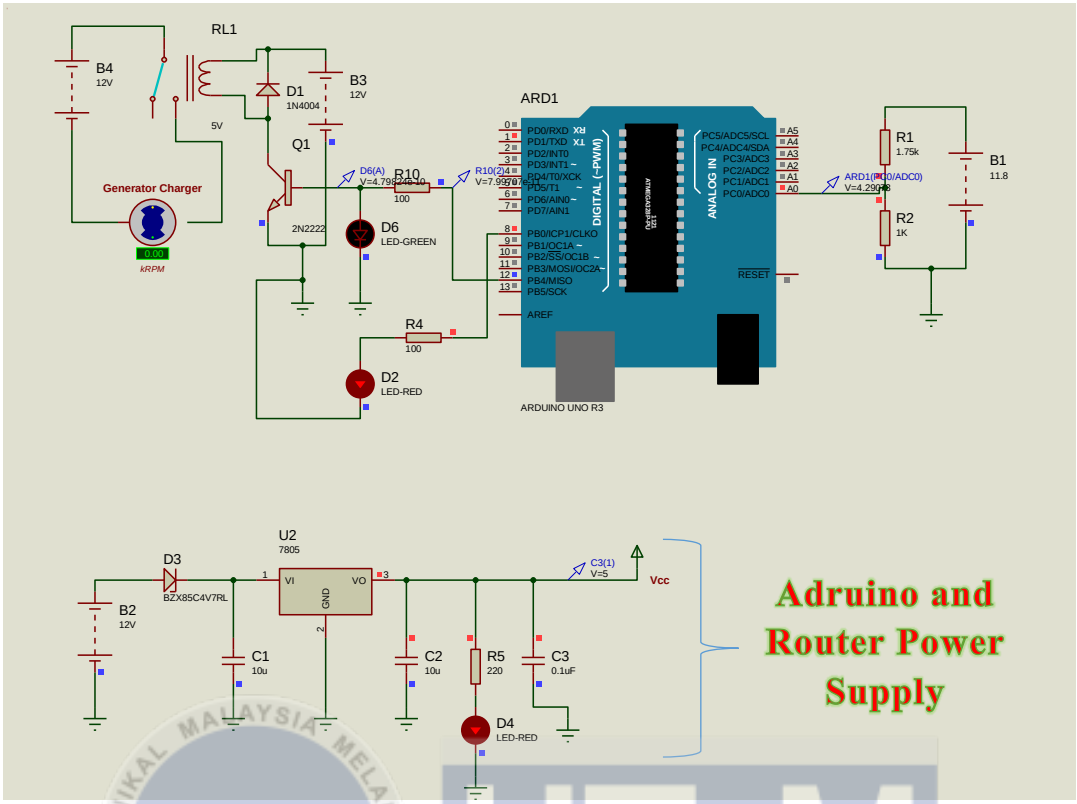


Figure 4.4: Generator Charger Controller Off

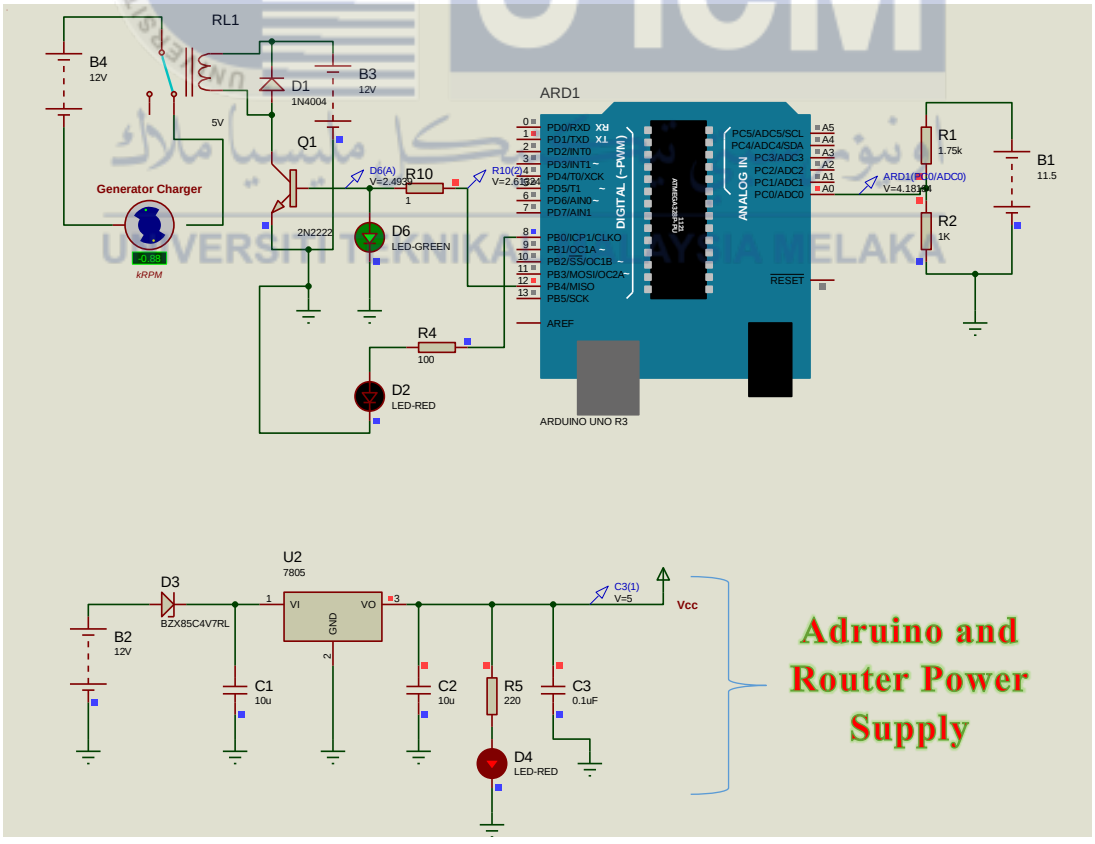


Figure 4.5: Generator Charger Controller On

4.3.1 Generator Charge Controller Hardware Construction

The controller is constructed on the strip board as shown in Figure 4.6. This part is relay system that changing the way of charging process. The RED LED is show that the battery on normal condition but if GREEN LED show that the battery charging with generator. This part also triggers the value of voltage and sense part for monitoring. Method to decrease the voltage by using voltage divider has been applied.

Figure 4.6: Controller Distribution

4.4 Analogue Display Hardware Construction

The analogue panel display is constructed on acrylic and shown in Figure 4.7. There are the digital panel displays by using millimetre. Apart from that, the connection of both displays is parallel to battery. Then, the display will show the value of voltage. The analogue and digital panel display is tested with DC supply and Figure 4.8 shows that the analogue display voltage is slightly different with the millimetre value.

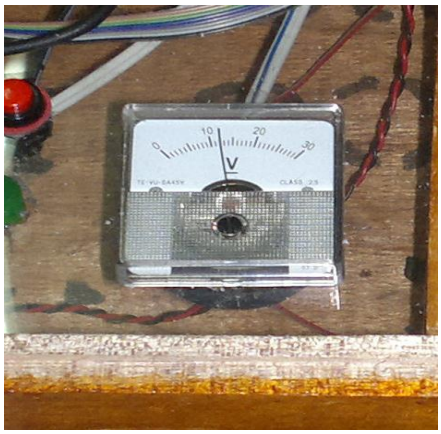


Figure 4.7: Hardware Construction of Digital Panel Display

Figure 4.8: Display Voltage Almost Same With The Multimeter

4.5 Controller Allocation

All the circuit is arranged and allocated according as shown in Figure 4.9. The 12V is connected to fan and lamp. There are acrylic on the top of this controller and this shown in Figure 4.10. It can easily see the charging process by generator or solar. Apart from that, the analogue panel display is displays the battery voltage and all the circuit is tested with 12V supply before testing with battery.

Figure 4.9: Hardware Setup Implementation

4.6 Controller Smart Hybrid and Monitoring

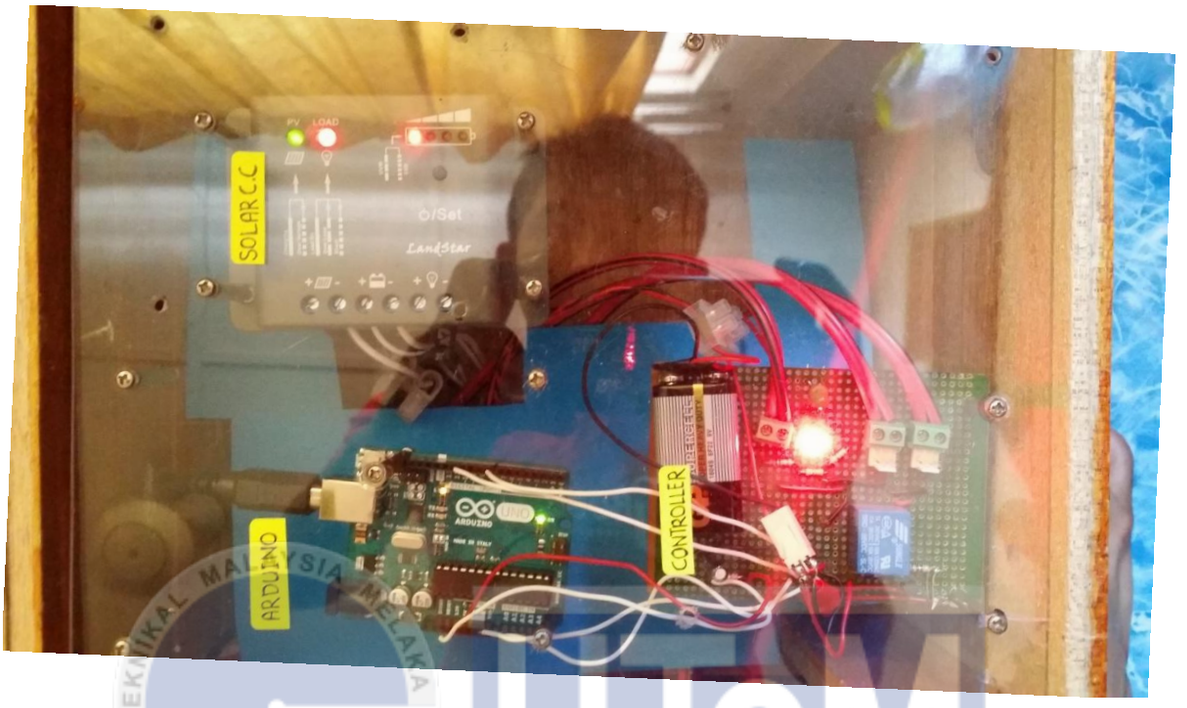
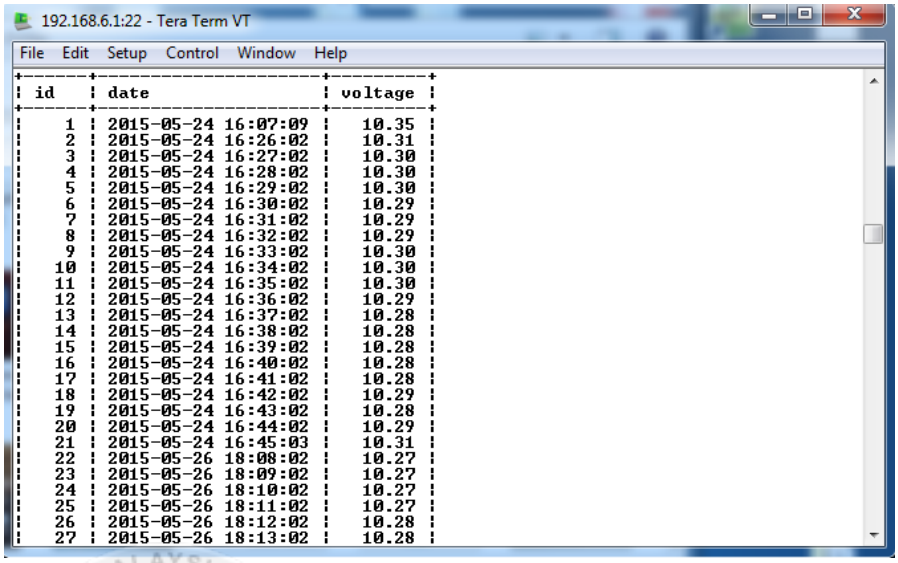


Figure 4.10: Part of Controller

Adruino as a multifunction controller has control the smart hybrid automation and monitoring part. The port have be used A0 as input, 12 as output, 8 as output, reset to disabling auto reset on serial connection, VCC as supply and ground. Solar have GREEN LED part to show the charging with solar panel and RED LED part to show the load process.

4.7 Database Storage



id	date	voltage
1	2015-05-24 16:07:09	10.35
2	2015-05-24 16:26:02	10.31
3	2015-05-24 16:27:02	10.30
4	2015-05-24 16:28:02	10.30
5	2015-05-24 16:29:02	10.30
6	2015-05-24 16:30:02	10.29
7	2015-05-24 16:31:02	10.29
8	2015-05-24 16:32:02	10.29
9	2015-05-24 16:33:02	10.30
10	2015-05-24 16:34:02	10.30
11	2015-05-24 16:35:02	10.30
12	2015-05-24 16:36:02	10.29
13	2015-05-24 16:37:02	10.28
14	2015-05-24 16:38:02	10.28
15	2015-05-24 16:39:02	10.28
16	2015-05-24 16:40:02	10.28
17	2015-05-24 16:41:02	10.28
18	2015-05-24 16:42:02	10.29
19	2015-05-24 16:43:02	10.28
20	2015-05-24 16:44:02	10.29
21	2015-05-24 16:45:03	10.31
22	2015-05-26 18:08:02	10.27
23	2015-05-26 18:09:02	10.27
24	2015-05-26 18:10:02	10.27
25	2015-05-26 18:11:02	10.27
26	2015-05-26 18:12:02	10.28
27	2015-05-26 18:13:02	10.28

Figure 4.11: Table in Tera Term

All data has collected from the analogue input with real time and store into this table MySQL. This part achieved because every single minute the data never miss to collect and store. The smart system make user easy to monitor hybrid system anywhere and anytime.

4.8 Controller Analysis

The controller is tested with 12V 7.2Ah battery without charging solar process. Apart from that, the controller is tested with load, fan, and lamp. The result of test with load is shown in figure 4.12 below. The controller is tested within a few days and recorded the voltage of battery. Every minute the real time data will be collect and store into MySQL. Graph will be plot by using this data.

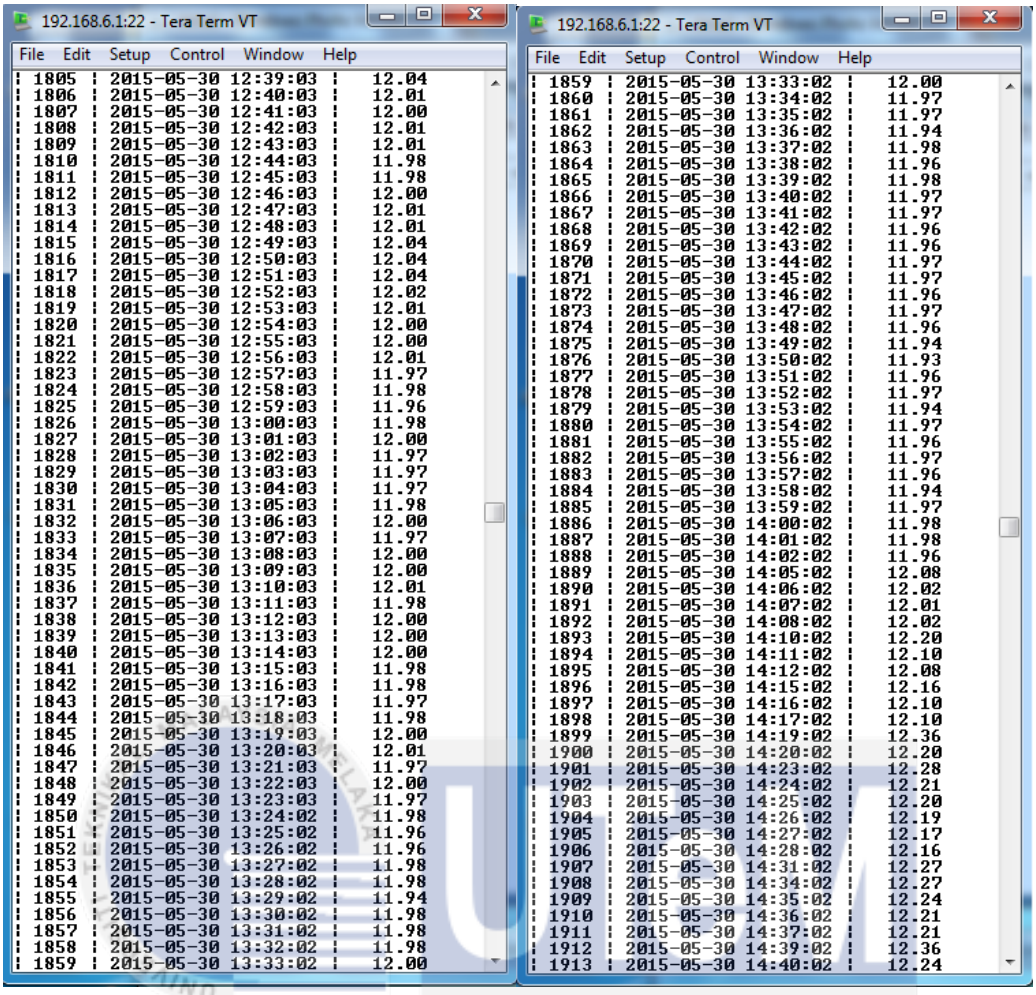


Figure 4.12: Table on 30th May 2015

Figure 4.12 shows the data of voltage on 30th May 2015. The sensor has been activated from 12.22 pm until 10.59pm. The data have been taken every one minutes in that day.

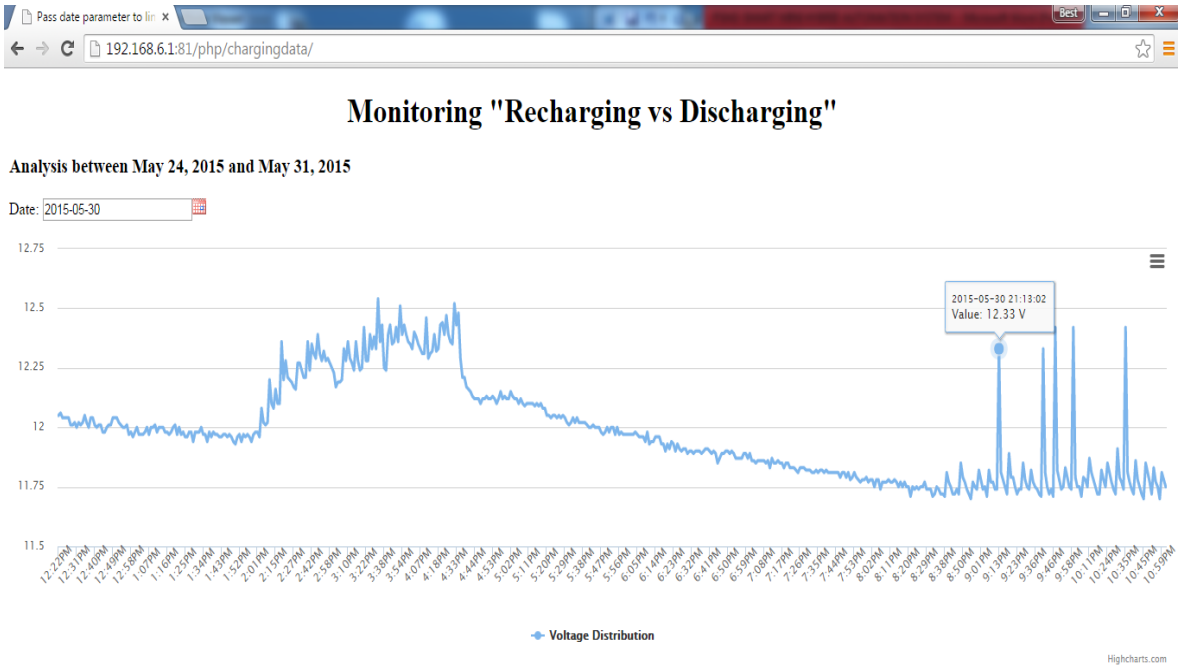


Figure 4.13: Graph on 30th May 2015

In Figure 4.13 shows that Monitoring "Recharging vs Discharging" by using webserver and the day is 30th may 2015. The date can be choose and flexible. Form the graph show that the battery voltage decrease dramatically at the initial to the first half an hour from 12.22pm and this is due to the self-discharge of the battery and the starting current of the fan and the energy consumption of the another load. Moreover, the battery voltage increases until 12.6V stage by stage at 2.01pm until 4.33pm because the sun intensity to the charging process and the load will still operating as normal.

After that, the voltage decreases because the sunset and self-discharging process happen. This discharging process needs 3 hour and half from 5.00pm until 8.30pm before the battery recharging using generator. The lowest value have been set is 11.7V because generator is turn on automatically to charge the battery at 11.7V for 100 second delay and the load will still operating as normal.

From the graph 4.13 shows the spike happen on whole night start from 8.30pm to 10.30pm because the data collect when the voltage in charging process using generator. That has a pick voltage happen in charging condition.

This smart hybrid system continues charging and discharging atomically for whole night and the operating continues until the sun rise out and charging process by using solar energy. Every data have been taken will be store to monitor another days.

Since this controller is using fundamental charging method to charge the battery therefore the charging process is slow when the sun intensity is low. The charging process increases when the sun intensity becomes high and this boosts the charging process. Furthermore, this smart way to charge the battery by generator in case of cloudy weather or raining day.

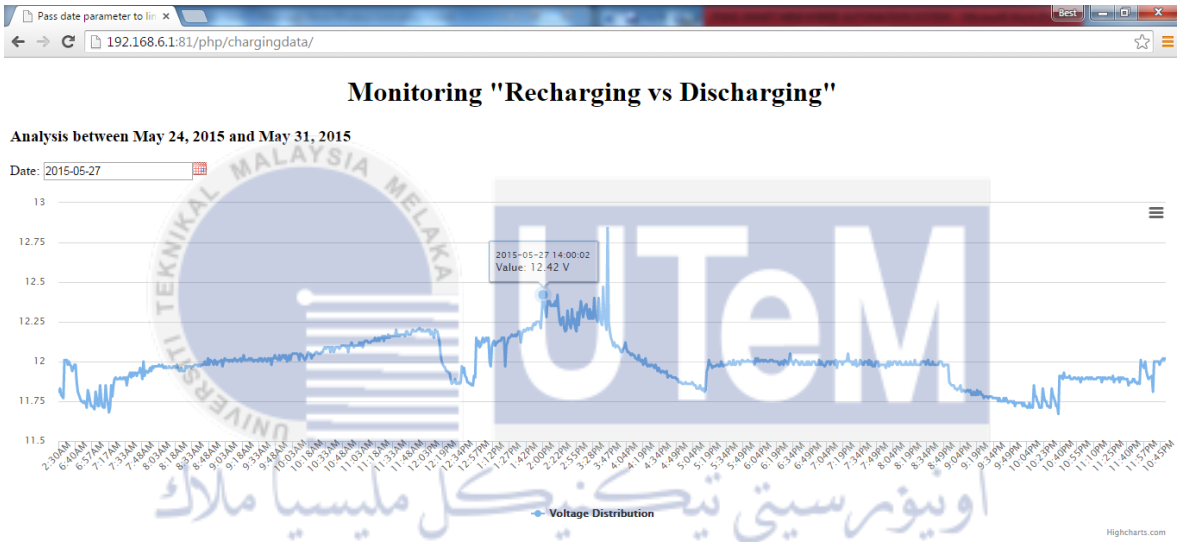


Figure 4.14: Graph on 27th May 2015

This graph on 27th may 2015. Its take for a whole day and the result is remain the same with 30th may 2015. But there have a bit different because the weather was a bit cloudy and dazzle, so the voltage got a slight decrease in the evening but generator still cover for the load. This graph just to approve that any way it goes the process charging and recharging will same. Method of smart hybrid home automation system is achieved.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

Renewable energy becomes more popular nowadays and this renewable energy can help to reduce the release of carbon dioxide. The controller designed is able to distribute energy to the components in our house for example lighting system, charger and fan. Furthermore, generator will turn on when the battery voltage reach the minimum desire. The cut-off charging will happen to give a stable voltage to load. In the same time, if generator not operates, the low voltage disconnects controller will cut-off the load when the battery voltage reaches the minimum set point.

Moreover, the controller is able to store the unused energy to the battery with the fundamental solar charge controller, yet the charging process is slow when the sun intensity is low. The 12V 7.2Ah battery is able to supply for the house equipment with the 20W solar panel. Apart from that, the system is able to work with other controller and the web-server monitoring able to display the charging and discharging energy and battery voltage. Furthermore, the voltage indicator can indicate the voltage range accurately and the blinker is working well when the battery voltage drops to minimum setting point. The objectives were achieved.

5.2 Recommendation

There are still rooms for improvement and enhancement for this project. Apart from that, the improvement and enhancement can be made with adding features for future development on the board. The solar charge controller can change to Maximum Power Point Trekker for higher efficiency charging the battery. Apart from that, the circuit size can be reduced to small size and the temperature sensor can be added to the controller to protect the increasing temperature of the battery for safety purpose. Furthermore, to apply to real house a bigger capacity of battery is required. Moreover, the controller can be improved by adding the inverter for AC equipment. The power switching between solar and line distributor controller can be added to the system for higher efficiency power supply to the house. Moreover, LCD can be installed to the controller to display the voltage of the battery and the voltage from solar panel. The monitoring in web server should be added more specification in monitoring real-time in power and temperature. Lastly, all the system can be control in web server is required.

REFERENCES

[1] Bailey, Robert L. 1980. *Solar* Electrics Research and *Development.* Ann Arbor, MI: Ann Arbor Sciences; pp. 2-186.

[2] Johnston, W.o., Jr. 1980. Solar *Voltaic* Cells. New York, NY: Marcel Dekker, Inc.; pp. 53-72.

[3] Chalmers, Bruce. 1976. "The Photovoltaic Generation of Electricity." *Scientific American.* Vol. 235 (No.4): pp.34-43.

[4] Tauc, Jan. 1972. *Photo and Thermoelectric Effects* in *Semiconductors.* New York, NY: Pergamon Press; pp.90-129.

[5] Bleicher, Samuel A. 1996. "The Mercury-Containing and Rechargeable Battery Management Act of 1996: A New Direction for Recycling." Environment Reporter. Vol. 27. The Bureau of National Affairs, Inc.

[6] Dr. Shahidul Islam Khan, Solar Home System, in Charge Controller (Bangladesh, 2013)

[7] Everett M. Barber and Joseph R. Provey, Convert Your Home to Solar Energy (Taunton Press, 2010)

[8] Introduction to the Arduino Microcontroller Shanghai Jiao Tong University June 17 – 29 , 2012

[9] Yong-tao ZHOU, Xiao-hu CHEN, Xu-ping WANG, Chun jiang YAO, 2008": Design of Equipment remote monitoring System Based on Embedded Web", IEEE, pp 73-78.

[10] Ali Ziya Alkar, Mehmat Atif Karaca, 2009", An Internet Based Interactive Embedded Data- Acquisition System for Real-Time Applications", IEEE, Vol .58.No.3, pp 522- 529.

[11] J. Tang, X. Zhang, "Cross-Layer-Model Based Adaptive Resource Allocation for Statistical QoS Guarantees in Mobile Wireless Network", IEEE Transactions on Wireless Communications, Vol. 7, No. 6, Jun 2008, 2318-2328.

[12] C.E. Palazzi, M. Brunati, and M. Roccetti. An openwrt solution for future wireless homes. In *Multimedia and Expo (ICME), 2010 IEEE International Conference on*, pages 1701 –1706, july 2010.

[13] A.P. Ortega, X.E. Marcos, L.D. Chiang, and C.L. Abad. Preventing arp cache poisoning attacks: A proof of concept using openwrt. In *Network Operations and Management Symposium, 2009. LANOMS 2009. Latin American*, pages 1 –9, oct. 2009.

[14] Khusvinder Gill, Shuang-Hua Yang, Fang Yao, and Xin Lu. "A ZigBee-Based Home Automation System". IEEE Transactions on Consumer Electronics, Vol. 55, No. 2, MAY 2009.

[15] R. Piyare and M. Tazil, "Bluetooth based HOME AUTOMATION SYSTEM using cell phone," in *Consumer Electronics (ISCE), 2011 IEEE 15th International Symposium on*, 2011, pp. 192-195.

[16] A.R.Al-Ali and M. AL-Rousan. "Java-based home automation system". IEEE Transaction on Consumer Electronics, Vol.50, No. 2, May 2004.

[17] Zhi-Ming Lin, “A remote telephone-controlled home automation system”, IEEE Southeastcon '98, pp. 148- 152, 1998.

[18] Rozita Teymourzadeh, CEng, Member IEEE/IET, Salah Addin Ahmed, Kok Wai Chan, and Mok Vee Hoong, "Smart GSM Based Home Automation System", 2013 IEEE Conference on Systems, Process & Control (ICSPC2013), 13 - 15 December 2013, Kuala Lumpur, Malaysia.

APPENDIX A

Source Code for Adruino

```
const int analogPin = A0;    // pin that the sensor is attached to
const int generator  = 12;
const int solar = 8;       // pin that the LED is attached to
const int threshold = 858;   // 4.25V an arbitrary threshold level that's in the range of the
analog input
int val = 0;       // variable to store the data from the serial port

int sensorValue = 0;        // value read from the pot
int outputValue = 0;

void setup() {
  // initialize the LED pin as an output:
  pinMode(generator, OUTPUT);
  pinMode(solar, OUTPUT);
  // initialize serial communications:
  Serial.begin(9600);
}

void loop() {
  // read the value of the potentiometer:
  int analogValue = analogRead(analogPin);

  // if the analog value is high enough, turn on the LED:
  if (analogValue < threshold) {
    digitalWrite(generator, HIGH); //genarator
    digitalWrite(solar, LOW);
    delay (100000);
  }
  else {
```

```
    digitalWrite(generator, LOW); //solar
   digitalWrite(solar, HIGH);

  }
if (Serial.available()) {
   // For the below examples, let's pretend that the passed-in serial
   // value is character '5'.
   // Since the declared variable val is an int, it converts the char
   // value passed in into an int.
   // If char val = '5', the numeric representation is 53.
   val = Serial.read();      // read the serial port

   // If the stored value is a single-digit number, blink the LED
   // that number of times.
   // Here we compare the int value of val against the int values
   // of the string.
   // Characters '0' and '9' are equivalent to integer 48 and 57.
   if (val > '0' && val <= '9' ) {
     Serial.print(val);
     // Convert from char to int:
     // From above, int conversion of val, which is char '5', is 53.
     // int conversion of char '0' is 48.
     // 53-48 = 5  : blink that # of times

       digitalWrite(solar,HIGH);
       delay(150);
       // read the analog in value:
       sensorValue = analogRead(analogPin);
       String str = String(sensorValue);
       Serial.print("  \t  ");
       Serial.println(str);
     }

  // print the analog value:
```

```
  Serial.println(analogValue);
  delay(1);        // delay in between reads for stability
}
}
```

APPENDIX B

Source Code for *Index.php*

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>


<title>Pass date parameter to line chart</title>
<html lang="en">
<head>
  <meta charset="utf-8" />
  <title>jQuery UI Datepicker - Icon trigger</title>
  <link rel="stylesheet" href="http://code.jquery.com/ui/1.10.1/themes/base/jquery-ui.css"
/>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <script src="http://code.jquery.com/ui/1.10.1/jquery-ui.js"></script>
  <link rel="stylesheet" href="/resources/demos/style.css" />
  <script>

  </script>

  <script type="text/javascript">
  var options;

  $(document).ready(function() {

              options = {
                chart: {
              renderTo: 'container',
```

```
    type: 'line'
},
title: {
    text: ''
},
subtitle: {
    text: ''
},
xAxis: {
    categories: [],
    labels: {
                                    align: 'center',
                                    x: -3,
                                    y: 20,
                                    formatter: function() {
                                        return Highcharts.dateFormat('%l:%M%p',
Date.parse(this.value +' UTC'));
                                    }
                                }
},
yAxis: {
    title: {
        text: ''
    }
},

    tooltip: {
pointFormat: "Value: {point.y:.2f} V"
    },
    plotOptions: {
series: {
    cursor: 'pointer',
    point: {
        events: {
```

```
                click: function () {
                    alert('Category: ' + this.category + ', value: ' + this.y);
                }
            }
        }
    }
},
    series: [{
                type: 'line',
                name: '',
                data: []
        }]
            }

$.getJSON("data.php", function(json){
    options.xAxis.categories = json['category'];
    options.series[0].name = json['name'];
    options.series[0].data = json['data'];
    chart = new Highcharts.Chart(options);
            });

});

$(function() {
        $( "#datepicker" ).datepicker({
          dateFormat: "yy-mm-dd",
          showOn: "button",
          buttonText: 'Show Date',
          buttonImage: "calendar.gif",
          buttonImageOnly: true,
          beforeShow: function(){
       $(".ui-datepicker").css('font-size', 10)
          },
          onSelect: function(dateText, inst) {
```

```
            $.getJSON("data.php?dateParam="+dateText, function(json){
                options.xAxis.categories = json['category'];
                options.series[0].name = json['name'];
                options.series[0].data = json['data'];
                chart = new Highcharts.Chart(options);
                        });
             }
           });
        });

        </script>

</head>
<body>
<script src="http://code.highcharts.com/highcharts.js"></script>
<script src="http://code.highcharts.com/modules/exporting.js"></script>

<h1><center>Monitoring "Recharging vs Discharging""</center></h2>
<h3>Select between May 1, 2015 and May 31, 2015</h2>

<p>Date: <input type="text" id="datepicker" /></p>

<div id="container" style="min-width: 400px; height: 400px; margin: 0 auto"></div>

</body>
</html>
```

APPENDIX C

Source Code for *data.php*

```
<?php
$con = mysql_connect("localhost","root","qwe123");
if (!$con) {
  die('Could not connect: ' . mysql_error());
}

mysql_select_db("monitoring", $con);
//$date = DateTime::createFromFormat('dd-mm-yy', $_GET["dateParam"]);
if (isset($_GET["dateParam"])) {

        $sql = mysql_query("SELECT date, voltage FROM charging WHERE date LIKE
'".$_GET["dateParam"]."%'");
        #$sql = mysql_query("SELECT date, voltage FROM charging WHERE date LIKE
'".$_GET["dateParam"]."%'");

} else {
        $sql = mysql_query("SELECT date, voltage FROM charging WHERE date LIKE
'2015-05-25%'");
}
$result['name'] = 'Voltage Distribution';
while($r = mysql_fetch_array($sql)) {
        $datetime = $r['date'];
        $result['category'][] = $datetime;
        $result['data'][] = $r['voltage'];
}

print json_encode($result, JSON_NUMERIC_CHECK);
mysql_close($con);
?>
```

APPENDIX D

Source Code for MySQL Database

```
<?php
$DB_NAME = 'weather';
$DB_HOST = 'localhost';
$DB_USER = 'root';
$DB_PASS = 'qwe123';
$mysqli = new mysqli($DB_HOST, $DB_USER, $DB_PASS, $DB_NAME);
if (mysqli_connect_errno()) {
printf("Connect failed: %s\n", mysqli_connect_error());
exit();
}
$query = "select * from johor";
if ($result = $mysqli->query($query)) {
{
$rows = array();
$table = array();
$table['cols'] = array (
array('label' => 'date', 'type' => 'string'),
array('label' => 'temp', 'type' => 'number')
);
while ($row = $result->fetch_assoc()) {
$temp = array();
$temp[] = array('v' => (string) $row['date']);
$temp[] = array('v' => (int) $row['temp']);
$rows[] = array('c' => $temp);
}
}
}
$table['rows'] = $rows;
$jsonTable = json_encode($table); // make sure the php_json plugin installed
```

```
?>
<html>
<head>
<!--Load the Ajax API-->
<script type="text/javascript" src="https://www.google.com/jsapi"></script>
<script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1.8.2/jquery.min.js"></script>
<script type="text/javascript">
// Load the Visualization API and the piechart package.
google.load('visualization', '1', {'packages':['corechart']});
// Set a callback to run when the Google Visualization API is loaded.
google.setOnLoadCallback(drawChart);
function drawChart() {
// Create our data table out of JSON data loaded from server.
var data = new google.visualization.DataTable(<?=$jsonTable?>);
var options = {
title: 'Temperature Chart',
is3D: 'true',
width: 800,
height: 600
};
// Instantiate and draw our chart, passing in some options.
// Do not forget to check your div ID
var chart = new google.visualization.LineChart(document.getElementById('chart_div'));
chart.draw(data, options);
}
</script>
</head>
<body>
<!--this is the div that will hold the pie chart-->
<div id="chart_div"></div>
<?php echo $jsonTable; ?>
</body> </html>
```

APPENDIX E

Source Code for *charging_date.py*

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

import MySQLdb
from time import sleep
import serial
import sys
import datetime

#open the serial port
#s = serial.Serial('/dev/ttyACM0',9600,timeout=0.0001)
s = serial.Serial('/dev/ttyACM0',9600,timeout=1)

#disable DTR
s.setDTR(level=False)

#wait for 2 seconds
sleep(0.5)

#send the data
#while True:
s.write("5")
data = s.readline()
#print data
#print s.readline() # Read the newest output from the Arduino
pieces = data.split("\t")  #split the data by the tab
sensor1 = pieces[0]
sensor2 = (float(pieces[1]) /1024) * 12
sensor2_2dec = float("{0:.2f}".format(sensor2))
```

```
#print "%s\t" % (float(pieces[1])) #for debugging
#print "%s\t%s" % (sensor1, pieces[1]) #for debugging
#print "\n" #for debugging
#today = time.strftime("%Y-%m-%d")
now = datetime.datetime.now()
f = '%Y-%m-%d %H:%M:%S'
dt = now.strftime(f)

try:
    con = MySQLdb.connect('127.0.0.1', 'root', 'qwe123', 'monitoring')
    cursor=con.cursor()
    sqlstr="INSERT INTO charging (date,voltage) VALUES(%s, %s)", (dt,sensor2_2dec)
    cursor.execute("INSERT INTO charging (date,voltage) VALUES (%s,%s)", (dt,
sensor2_2dec))
    #print  sqlstr #for debugging
    con.commit()
    cursor.close()

except MySQLdb.Error, e:

    print "Error %d: %s" % (e.args[0], e.args[1])
    sys.exit(1)

finally:

  if con:
   con.close()
```