UNOPLANTER: ARDUINO AUTOMATIC WATERING SYSTEM

SIAH BING SHENG

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

BORANG PENGESAHAN STATUS TESIS

JUDUL: UNOPLANTER: ARDUINO AUTOMATIC WATERING SYSTEM

SESI PENGAJIAN: 2016/2017

Saya SIAH BING SHENG

(HURUF BESAR)

mengaku membenarkan tesis (PSM/Sarjana/Doktor Falsafah) ini disimpan di Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dengan syarat-syarat kegunaan seperti berikut:

1. Tesis dan projek adalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. ** Sila tandakan (/)

____	SULIT	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
____	TERHAD	(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
/	TIDAK TERHAD	

(TANDATANGAN PENULIS)

Alamat tetap: 33a, Lorong Singa Baru 2/2, Taman Singa Baru 2, 32000, Sitiawan, Perak.

Tarikh: 23 AUGUST 2017

(TANDATANGAN PENYELIA)

DR. ASLINDA BINTI HASSAN

Nama Penyelia

Tarikh: 23 AUGUST 2017

CATATAN: * Tesis dimaksudkan sebagai Laporan Akhir Projek Sarjana Muda (PSM)
** Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa.

UNOPLANTER: ARDUINO AUTOMATIC WATERING SYSTEM

SIAH BING SHENG

This report is submitted in partial fulfilment of the requirements for the Bachelor of Computer Science (Computer Networking)

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2017

DECLARATION

I hereby declare that this project report entitled

UNOPLANTER: ARDUINO AUTOMATIC WATERING SYSTEM

is written by me and is my own effort and that no part has been plagiarized without citations.

STUDENT : _____ Date : 23 AUGUST 2017

(SIAH BING SHENG)

I hereby declare that I have read this project report and found this project report is sufficient in term of the scope and quality for the award of Bachelor of Computer Science (Computer Networking) With Honours.

SUPERVISOR : _____ Date : 23 AUGUST 2017

(DR. ASLINDA BINTI HASSAN)

DEDICATION

I dedicate this thesis to my family who sacrificial care for me with affections and love and make me possible to complete this project within the time limit.

ACKNOWLEDGEMENTS

I would like to express my special thanks of gratitude to my supervisor, Dr. Aslinda Binti Hassan, who provided me suggestions and assistance along the project development until complete this project successfully. Secondly I would also like to thank to my friends that is helping me and giving me the precious advice on this project.

I also appreciate the support given by my family who always provided me with moral and emotional support when I facing problems.

ABSTRACT

This project is about the development of the prototype to water the plant automatically using the Arduino. There is no doubt that the modern people is busy with their daily schedule and has fewer times to take care of their plants. Therefore, UnoPlanter is developed to solve this problem by automatically watering the plants when the soil moisture sensor has detected the soil is insufficient of water. UnoPlanter is using the Arduino as the centre core, and other hardware is assembled to it to function based on the code that has been written into the Arduino. UnoPlanter is a fully function prototype which consists of one soil moisture sensor which use to detect the moisture of the soil, and a LCD display to show the moisture percentage and pump status, and a relay module which used to control the on and off of the water pump, and a water pump which connects directly to the relay module. When the soil moisture sensor sense the dry soil, it will show the moisture percentage on the LCD display, and the relay module will switch on the water pump automatically to start the watering process, or vice versa. Therefore, the home gardener will never be worry about their plants even when they are going far for vacation.

ABSTRAK

Projek ini adalah pembangunan prototaip untuk automatik menyiram tumbuhan di rumah dengan menggunakan Arduino. Pada masa moden ini, orang semakin sibuk dengan kerja dan kehidupan dan mereka selalunya tidak mempunyai masa untuk menjaga tumbuhan mereka. Oleh itu, UnoPlanter telah dicipta untuk menyelesaikan masalah ini dengan menyiram tumbuhan automatik tanpa orang untuk mengoperasi, tumbuhan akan disiram apabila sensor lembapan tanah mengesan tanah itu tidak mengcukupi air. UnoPlanter menggunakan Arduino sebagai otak bagi projek tersebut, perkakasan lain akan dipasang kepada Arduino dan difungsi berasaskan coding yang telah ditulis ke dalam Arduino. UnoPlanter adalah sebuah prototaip yang sudah boleh berfungsi dalam kehidupan harian, ia mempunyai satu sensor lembapan tanah untuk mengsesan kelembapan tanah, dan satu LCD untuk memaparkan peratusan kelembapan tanah dan status pump, dan satu relay modul untuk mengawal buka dan tutup air pump, air pump tersebut dipasang terus kepada relay modul. Apabila sensor tersebut mengesan tanah yang kering, peratusan akan dipaparkan ke atas LCD, dan relay modul itu akan membuka air pump itu secara automatik untuk menyiram, atau sebaliknya. Oleh itu, pengguna usahlah risau tumbuhannya walaupun sedang bercuti di tempat yang jauh.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURE

CHAPTER I

INTRODUCTION

1.1. Introduction

There is no doubt that in this modern life there is getting more and more people likes to become a green finger whether they are planting vegetables, fruits, or even a small plantation pot. Due to the busy schedule of the daily routine, people tend to forget to take good care of their plants such as water them in regular basis, this may lead to rotting of the plant as the times getting longer and longer. Besides, long vacation is unavoidable sometimes, and there will be another trouble when there are a few pots of plants that need to take care of, it will be a worry when there is no one can help to water the plants when people are far away.

To overcome all the problems stated above for the home gardener, a system is going to be develop which is an automatic watering system which named UnoPlanter. UnoPlanter is using a microcontroller which is Arduino Uno for physical computing which can be used to control devices and hardware that attached to it including the soil

moisture sensor. For this project, the main sensor used is the soil moisture sensor which is connected to Arduino Uno to get the reading of the soil to constantly check for the soil humidity of the plant.

UnoPlanter is the solution to solve the above problems as it can automatically water the plant as the sensor get to know that the soil is currently dry. When the soil is wet the system will automatically stop the watering process and keeps on monitor the soil humidity, it will start to water the plant until it detects the soil is dry again. Besides, it has a LCD screen that will shows the humidity percentage of the soil and the water pump status whether it is on or off. With this UnoPlanter, it will never be worry about when is the last time to water the plant as the system will do it automatically. Long vacation will never be a problem for the home gardener anymore as the watering plants will leave for the system to do for you. With UnoPlanter, the plants will hardly be wither and the survive rate will be increase if the users take good care of it.

1.2. Problem Statement

There will be several problems that maybe face by the people who plant, which is difficulties in determining when is the suitable time to water the plants, because it is hard to tell from the soil whether the plant is dry or wet if the weather is not too extreme. The second problem will be no automation watering system, which people may distract by the busy routine, therefore automatically water system will be needed.

Table 1.1: Summary of problem statement

PS	Problem Statement
PS1	Difficulties in determining when is the suitable time to water the plants.
PS2	No automation watering system.

1.3. Project Question

Project Question is the important part that use to identify each problem that being stated early. Below table show summary of project question based on problem statement.

Table 1.2: Summary of Project Question

PS	PQ	Project Question
PS1	PQ1	How to determine the level of moisture in the soil?
PS2	PQ2	How to develop a watering system that provides water when it is needed?

1.4. Project Objective

Project objective defines the things that want to achieve. The objectives must be considered based on the problem statement and the project question of this project.

PO1: To detect the moisture of the soil using soil moisture sensor.

Using the soil moisture sensor, we can detect and check the soil humidity of the plant.

PO2: To display the moisture of the soil and pump condition using LCD screen.

Using the LCD screen, we can read the level of soil humidity that detects from the soil moisture sensor and the pump condition which is ‘ON’ or ‘OFF’.

PO3: To develop an automatic watering system using Arduino.

After getting the soil moisture level from the soil, the system will let the water pump to automatic water the plant when it is too dry and off the water pump when the soil of the plant is wet.

Table 1.3: Summary of Project Objectives

PS	PQ	PO	Project Objectives
PS1	PQ1	PO1	To detect the moisture of the soil using soil moisture sensor.
		PO2	To display the moisture of the soil and pump condition using LCD screen.
PS2	PQ2	PO3	To develop an automatic watering system using Arduino.

1.5. Project Scope

Scope of the project going to be handle as follows:

1.) This project uses Arduino Uno microcontroller a device that interact to hardware.
   - Soil moisture sensor been use to put into soil to detect the soil humidity.
   - Water pump is ready to on and off based on the soil humidity that detected from the sensor.
   - C/C++ programming language to run the script in Arduino Uno.

2.) For this project less hardware will be obtaining to minimize cost.

3.) This project is focus for the small plant pots in the house for the home gardeners.

1.6.Project Contribution

Project contribution defines the expected output from this project. This part can be referred to the objectives of this project.

Table 1.4: Summary of Project Contribution

PS	PQ	PO	PC	Project Contribution
PS1	PQ1	PO1	PC1	Proposed the soil humidity condition of the plant for the users.
		PO2		
PS2	PQ2	PO3	PC2	Proposed a system which will automatically water the plant when the plant is dry and stop water the plant when it is wet.

In this project contribution, there have two contributions that have been considered of this project as shown on Table 1.4.

1.7. Thesis Organization

In this section, thesis organization that define summary of each chapter include in this report. There are total of seven chapter that need to be implement. All description of each chapter shown below:

Chapter 1: Introduction

In this chapter concentrate on introduction of project that involve brief introduction and background of project. It also contains specific problem that influence to create this project. Focus on introduction, problem statement, project question, objective, scope, contribution and conclusion of chapter.

Chapter 2: Literature Review

In this chapter concentrate in the idea of this project and explanation that referred to any reading material or publish thesis, research that related to project that guide the concept of this project.

Chapter 3: Project Methodology

In this chapter concentrate on project management, that use waterfall methodology development approach. This chapter will ease task for implementing and organize project. It also includes milestone of this project.

Chapter 4: Analysis and Design

In this chapter concentrate in analysis requirement being use in this project including graphical user interface (GUI), Database use to develop this project and software design.

Chapter 5: Implementation

In this chapter concentrate on implementing of this project which involve configure software management, environment setup and implementation status.

Chapter 6: Testing

In this chapter concentrate in testing project that being implement, by creating test plan, strategy, design and finding result of project.

Chapter 7: Project Conclusion

In this chapter concentrate on explaining project summarize by stating objective, contribution of project, limitation of project and future work that can be improve in system.

1.8.Conclusion

In conclusion, end of this chapter aiming that providing information above will help smooth the process of project and achieve objective. In The next chapter will concentrate in literature review, which will indicate idea of creating this project.

CHAPTER II

LITERATURE REVIEW

2.1. Introduction

In this chapter will discuss about the literature review to perform this project. To achieve the objectives on the project and let the process go smoothly, better understanding of concept and technique must be clear so it will be achieve easier.

Literature review is involve in collecting data from related publish information and materials from anywhere example like books, internet, journals, published papers and so on. The result of finding will cover the objectives of this project. This chapter is focus on finding the previous project, research and technique which related to the project, to prove that this project could really working and function well based on the previous evidence.

2.2. Literature Review

2.2.1. Domain

The domain of this project is focusing on how the system development interact with the Arduino Uno microcontroller, and know what is the technique used while interact with the other components.

In this project the Arduino is getting the data from the soil moisture sensor, and show the digital data in the LCD screen for the users to see the moisture level of the soil. When the moisture level of the soil is below some threshold level, the water pump will be start by the relay to water the soil until the humidity of the soil reached some level, the water pump will be stopped.

2.3. Facts and Findings

2.3.1. Arduino Uno Rev3

Definition

Figure 2.1 shows the view of the Arduino Uno which is the microcontroller that used in this project. Based on the figure, it is clearly show that all the pins and terminals of the Arduino Board. Arduino is like the brain of this project, it controls all the center of the process and the decision making. It is an open-source physical computing platform based on a simple microcontroller board, and a development environment for writing software for the board. Arduino Uno is based on the ATmega328P, and 8-Bit microcontroller. It uses a USB cable with an AC-to-DC adapter or battery to get started.

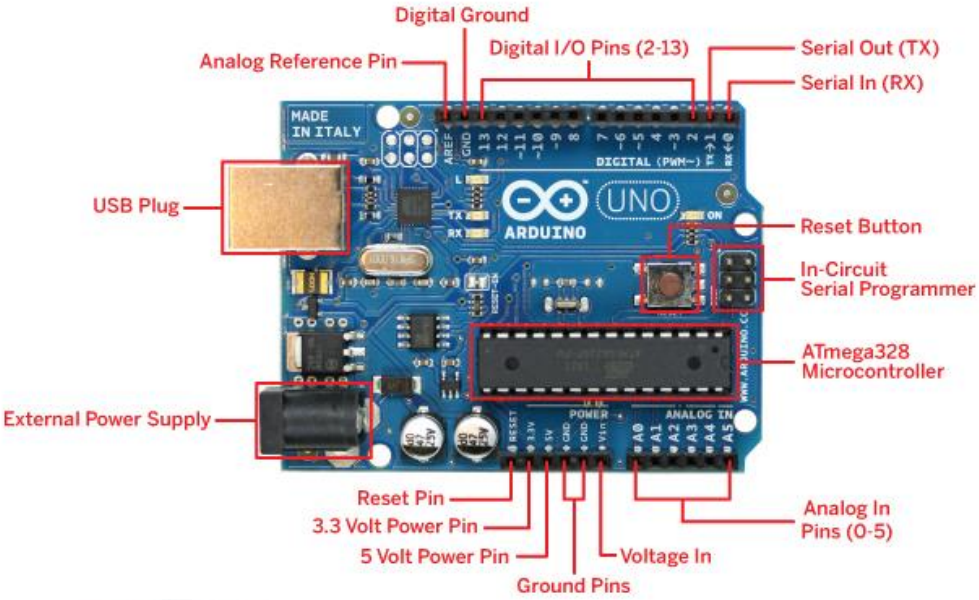


Figure 2.1 Arduino Uno Rev3

Hardware

There are several hardware that interfacing with the Arduino Uno to makes the project function which stated below:

- Soil Moisture Sensor YL-69: It is a soil moisture sensor or the hygrometer that detects the moisture of the soil based on the humidity. It is a perfect tools to detect the moisture of the soil as it can detect the humidity in the soil very well. The sensor is set up by two pieces which is the electronic board and the probe with two pads that use to detect the humidity in the soil. When the soil is wet, it will send the low output voltage. When it is dry, it will send the high output voltage.

Figure 2.2 Soil Moisture Sensor YL-69: Electronic board (right) and probe (left)

- Liquid Crystal Display (LCD): A LCD is a flat panel display that uses the light-modulating properties of liquid crystals. Liquid crystals using backlight to produce the screen images in color or monochrome instead of emit light directly, because the LCD screen does not emit light. LCD are available to display arbitrary images or fixed images with low information content, such as words, digits.

Figure 2.3 Liquid Crystal Display (LCD)

- Relay Module: A relay module is a switch that is controlled by an electromagnet. It is an automatic electric switch that used an electromagnet to move the switch from OFF to ON or vice versa instead of person from doing it. Relay module is use to control the DC water pump to switch ON and OFF.

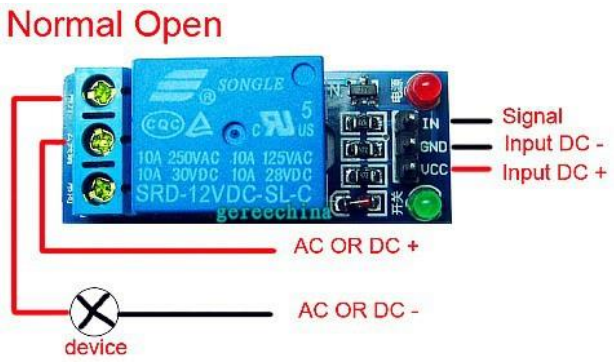


Figure 2.4 Relay Module

- DC Water Pump: The DC water pump we choose for this project is H-Bridge type, it is 12V DC water pump which is easy find it on market. The water pump will be connect to power supply and artificially supply water for the plant. It is control by the microcontroller electronically. It can be automatically ON and OFF by just sending signals as required which is control by the microcontroller and relay module.

Figure 2.5 DC Water Pump

- Integrated circuit (I2C) Module: I2C is a serial bus short distance protocol developed by Philips Semiconductor to enhance communication between the core on the board and various other ICs involved. I2C module is appropriate for interfacing to devices on a single board, and can be stretched across multiple boards inside a closed system. It has 16 pins which will directly connected to LCD. The brightness of the LCD screen can be adjust by tuning the display contrast button.

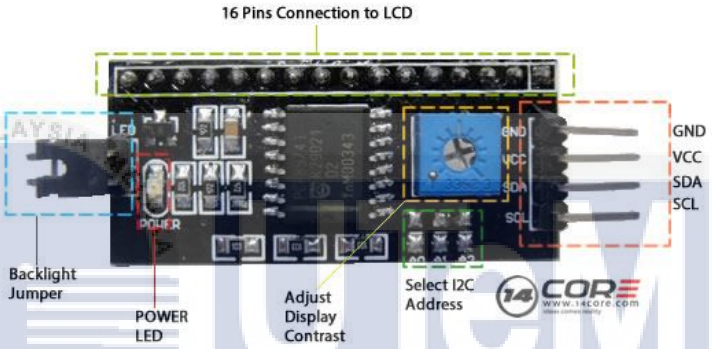


Figure 2.6 Integrated circuit (I2C) module

- DC Adapter 12V: The DC adapter is use to supply power supply to the DC power pump. It consist of plugs which is male-ended and jacks which is female-ended for supplying direct current power at which flow of electrical charge carriers that always takes place in the same direction.

Figure 2.7 12V DC Adapter

Software

Software is used in this project to write the code for sending the instruction to all the hardware to function. Besides, it is also a decision maker which control all the process of this project. The software used in this project is known as Arduino IDE. This software is use for sketches, in sketches all program will be written and load to hardware. This software is using C++ programming language to write. There are two part of function which makes the program run which is setup() and loop(). Setup function is use to initialize pin nodes or begin serial, while the loop function is allow us to control, change and respond to Arduino board.

Figure 2.8 Arduino IDE

2.3.2. Communication in Arduino

In this topic will explain about how to communicate to Arduino. There are plenty of ways to be made to connect to Arduino, such as using Ethernet Shield, GSM module, Wi-Fi module or using Serial communication through USB cable. In this project, Serial communication is chosen as the method that connect to the Arduino.

Serial

Serial communication is the method will be use in this project. It has the simplest way to make connection between Arduino and laptop or other devices. In this type of communication the digital pulse are send and receive between the devices. Before the connection between the devices can be made, the rate of communication and synch to rate must be set to same rate.

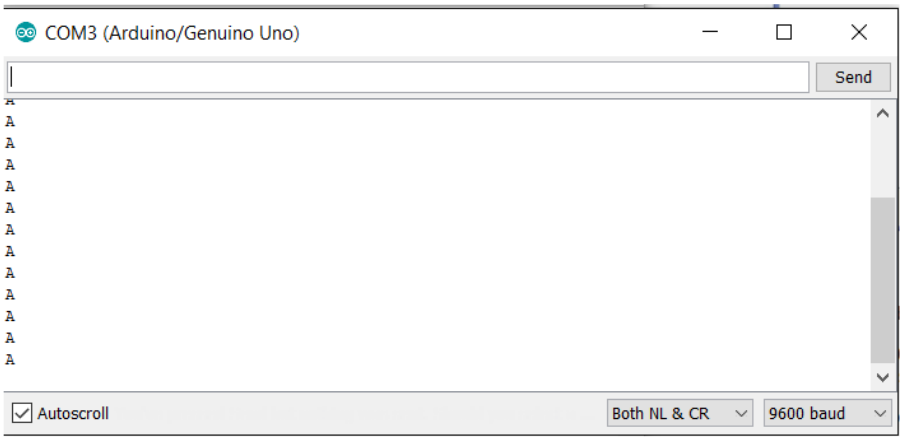


Figure 2.9 Serial monitor in Arduino IDE

2.4. Related Work

Based on previous research and findings from the reviews, it is found that a few projects that has been done which related to this project. Therefore, it is worth to be study for the reference in this project. In this chapter will elaborate the research that being studied.

2.4.1. Smart Irrigation System Using Raspberry Pi

Published in International Research Journal of Engineering and Technology, wrote by (Chate and Rana, 2016). This research is mainly a smart irrigation system that monitor the plant status using raspberry Pi. The parameters that focus is the temperature and the soil moisture. This system is developed to help the farmers to know its field status and bring convenience to the agriculture field. Nowadays all the things has go for the automation, apparently automation is one of the important role in the human life. No doubt that automation brings comfort and also reduce the energy and the wasting of water in the ecosystem. Therefore, automatic watering the field will never be an exception, it is efficiency and time saving, and also helps to reduce the man energy because the workload has been reduced. The irrigation system will automatic watering the fields depends on the condition and situation of the corps and fields. The objectives of this paper is to control the water motor automatically, and monitor the plant growth using webcam and watch the live streaming of farm. This research is mainly using the soil moisture sensor to get the data and use the Raspberry Pi as the core microcontroller. If the soil moisture value goes high, the signal condition circuit will give the notification to the raspberry pi board, and it will notify the relay to start the motor to water through the driver circuit as the raspberry pi cannot direct communicate with relay, the condition can be vice versa.

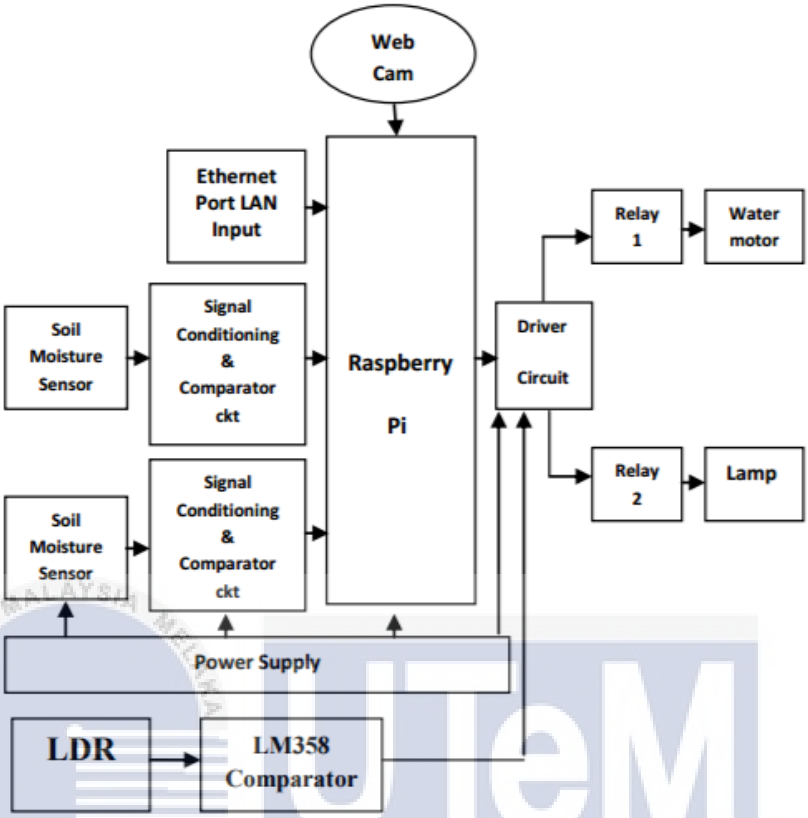


Figure 2.10 System Design of the Water Irrigation System (Chate and Rana, 2016)

2.4.2. Arduino Based Automatic Plant Watering System

Published in International Journal of Advanced Research in Computer Science and Software Engineering, wrote by (Devika et. al, 2014). This research is about the Arduino microcontroller used to control two functional components which are the moisture sensors and the motor/water pump to automatically water the plant. The Arduino Board is programmed using the Arduino IDE software. This project is using serial communication to connect the Arduino with the devices such as computer. When the

moisture sensor sense the soil is dry it will trigger the water pump to supply the water to the plants until the desired moisture level is reached.

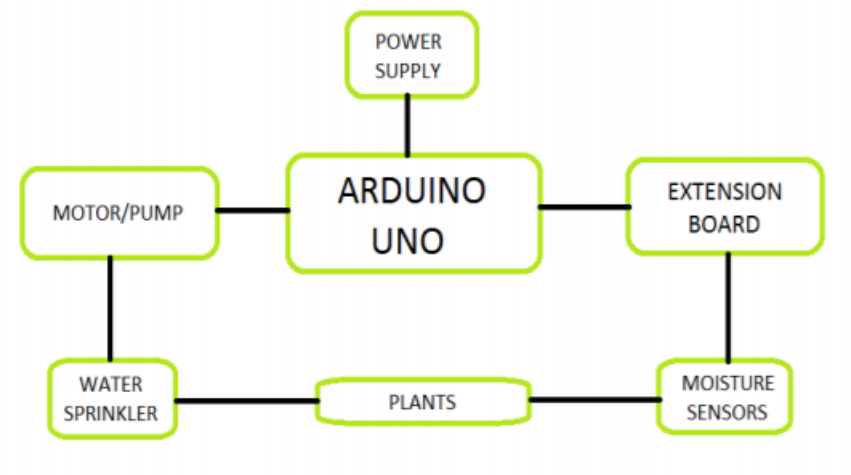


Figure 2.11 Design of the Automatic Plant Watering System (Devika et. al, 2014)

2.4.3. Automatic Gardening System using Arduino

Published in SSRG International Journal of Electronics and Communication Engineering, wrote by (Ankit Vashista et. al, 2016). This research is about the project named "GARDUINO", which is an automated gardening monitoring mechanism which will switch the water pump in between ON and OFF states on detecting the moisture level of the soil. This project is developed to lessen the human interference to give the convenience to human kind. This project is using the programmed Arduino microcontroller to control all the hardware that attached to it. The sensor used is soil moisture sensor and temperature sensor. Besides, it is using the GSM to make a user friendly remote access. Besides it will automatically supply the water to plants according to their needs, it will transmit the data and various SMS to the user. It also contain a LCD screen to display the data.

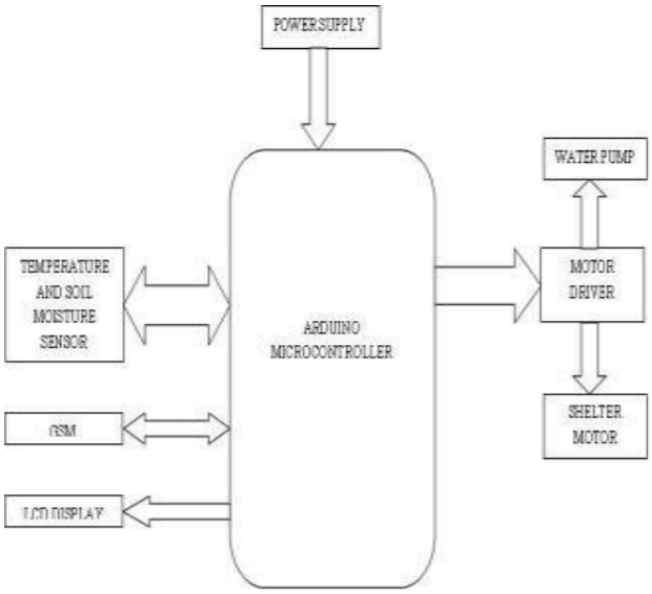


Figure 2.12 Design of the Automated Gardening System (Ankit Vashista et. al, 2016)

2.4.4. Microcontroller Based Automatic Plant Watering System

Published in International Journal of Computer Science and Engineering, wrote by (Mritunjay et. al, 2016). This research is about using the Arduino microcontroller to control the automatic watering system to water the plant due to increasing the wasting of water. It is using the soil moisture sensor and temperature sensor to detect the need to watering the plant. The Arduino is programmed by the C++ programming language to control the whole system, the water pump will supply the water after the sensor detect the dryness in the air and in the soil. After enough supplied of the water, the water pump will be switch off to save the water. This project is using the GSM Modem to transfer the data from distant places such as from one area to the area of the same city or from another city. In this project they are using SMS technique to quick transfer the data to the required destination. Besides, this project is using the serial communication which is using the USB

port to connect to the computer. Last but not least, this project is using a LCD display screen to show the information about the temperature and soil humidity.

2.5. Critical review of current problem and justification

From the research or the related work that being study on the above, it shows different kind of development has been used to create the automatic watering system using the Arduino Uno. Table below will show the comparison.

Table 2.1: Critical Review of Current Problem and Justification

Research	Central Processing	Sensors	Water irrigation system	LCD screen	Communication	Strengths	Weakness
Smart Irrigation System Using Raspberry Pi	Raspberry Pi	Humidity	Yes	No	Serial	Complex design using LAN and webcam for recording	Used more than one sensors to function this project
Arduino Based Automatic	Arduino	Humidity	Yes	No	Serial	Using only one power supply to	It uses a more difficult way to

Plant Watering System						control all the hardware	automatically start the water pump
Automatic Gardening System using Arduino	Arduino	Temperature Humidity	Yes	Yes	GSM	Using only one power supply to control all the hardware	Using two motors to start the water pump which is costly
Microcontroller Based Automatic Plant Watering System	Arduino	Temperature Humidity	Yes	Yes	GSM Serial	Using relay to start the water pump which will not cost too much	Using GSM modem to transfer data which is not necessary

2.6. Proposed Solution

The previous research that has been studied above shows that the "Microcontroller Based Automatic Plant Watering System" is more useful and implement to the project. The other research that may have resemblance will give some other useful information for the project too. Since the project proposed is using the Arduino controller, it is best choice that to study the same microcontroller that used. Well, all the research have used the soil moisture sensor which is a must for the project. The most important thing is the automatic water irrigation system. All the researches may have the automatic system but the

"Microcontroller Based Automatic Plant Watering System" provides the best and most detail about the automatic water irrigation system. Besides, the project needs the LCD screen to show the soil moisture percentage and the status of the pump. For the communication method, serial communication will always be the first choice due to its flexibility and it is the simplest way. Last but not least, relay module is the simple way to use to control the on and off of the water pump with its minimal hardware cost.

2.7. Conclusion

As conclusion, literature review is an important part that contribute to the project as it can establish the concept for the project. It helps to understand more and give a guidance of how to do it. With all the previous research that studied can helps to see a clearer concept about this project.

CHAPTER III

PROJECT METHODOLOGY

3.1. Introduction

In this chapter is discuss about the project methodology and the milestone that has been set to complete this project. This chapter is mainly for the planning purposes which is from the starting point till the end of the project according to the deadline. This chapter is to document the schedule and the milestone so that it is easy to trace back what is missed or what is have not done to ensure that all the progress are going smoothly.

Methodology is refer to the method of collecting data, interpret the data and solve and problem that faced in the project. By having a methodology is just like having a great planning for the project, it will help to conduct the project in the right ways and the procedure to goes smoothly. Therefore, having a systematic procedure follow by the methodology is important to any of the projects. Methodology is associate with planning,

analysis, design, implementing and testing of the project. By having a methodology, it helps the project to find the results in a proper way but not provide the absolute solution.

Last but not least, this chapter will also discuss about the milestone of this project, it is to ensure that the project is follow the dateline set and is running well follow the milestone to monitor the performance of the project until it achieve the objectives.

3.2. Methodology

For this project, it will choose to implement the Rapid Application Development (RAD) model, it is the alternatives to the conventional waterfall model. The main different is the RAD emphasize more on the process and put less prior on the planning, it is totally contrast compare to the waterfall model. It is consider as an incremental model, this model is used when the duration to build an application or a system is too short. Therefore, this model is designed to give more time on the development and produce high quality system compare to other lifecycle software development.

The RAD model is need to deliver the system in a very fast phase, it consists of four phase development cycle that combines the Standard System Development Life Cycle. But the phases are shortened and combined to increase the productivity. These four stages include Requirement Planning, Prototyping (include User Design and Construction), Testing and Cutover.

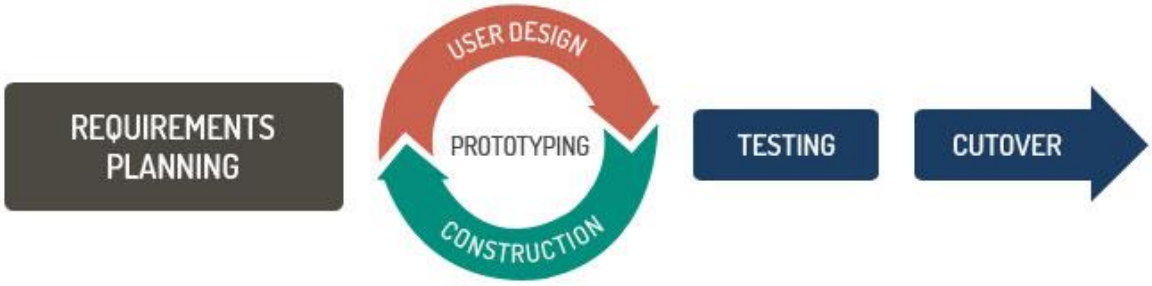


Figure 3.1 RAD Methodology Phase

3.2.1. Phase 1: Requirement Planning

This phase is the initial stage for the project, which specify about the problems if the project, and the objectives that wants to achieve. Right technique and module should be chosen in this phase and stated all the information needed including the hardware and software that will be used to build UnoPlanter.

- Software requirement

The software require for the system are Arduino IDE and Serial Monitor in Arduino IDE.

- Hardware requirement

The hardware require for the system are Arduino Uno, Soil Moisture Sensor, LCD, Relay Module, DC Water Pump, Integrated Circuit Module and DC Adapter.

- User requirement

User requirement are based on the problem statement which indicates that the problem that users have faced. First problem which is do not know when is the last time watered the plant and the second problem will be do not have automatically water system when the plant needs to be water.

3.2.2. Phase 2: Prototyping

Prototyping phase contain 2 stages which is user design and construction.

- User Design

In this stage, the prototype models must be design first in order to continue a better development. All the processes, inputs and outputs of UnoPlanter must be well designed. Example like how to assemble all the hardware so that the pin will not be overused or taken by the others. The water pump should be attach with what type of water pipe that is suitable and would be able to water in the most efficient way. The coding must be set correctly to show the contents on the LCD screen. All the process will be repeat until it is perfect.

- Construction

In this stage, the project that designed early will be construct and assemble accordingly. Construction phase is where the actual system coding, hardware assembling and integration takes place. The very beginning will be the hardware assembling, which attach all the hardware to the microcontroller Arduino Uno and make it all functional. After that will be setting up the Arduino IDE for the Arduino Uno and having a platform for the coding.

3.2.3. Phase 3: Testing

For the testing phase, the prototype that finished in the construction stage will be tested to see its functionality and testing for the errors. Testing will be include the moisture sensor that setup correctly and can be getting data from the soil. Besides, the LCD screen must be able to show the content that desired which is the moisture level and the pump status. Next, the relay module must be able to function so that the water pump is able to switch on and off based on the coding. Lastly, the water pump must be able to suck in the water from the water source and water out according to the percentage of the plant moisture.

3.2.4. Phase 4: Cutover

Last phase of the RAD for the project will be cutover. This is the phase that the UnoPlanter is successfully build in the aspect of hardware and software. The system is tested with numerous time until there is no error occur, and it is consider a complete system, which can be start to operate.

3.3. Project Milestones

To make sure that the project is going according to the dateline, project milestone is created to ensure all the activities planned will be completed accordingly. By using Gantt chart, it is easier to see and visualize the milestone and tracking all the activities that needs to be complete within the timeline.

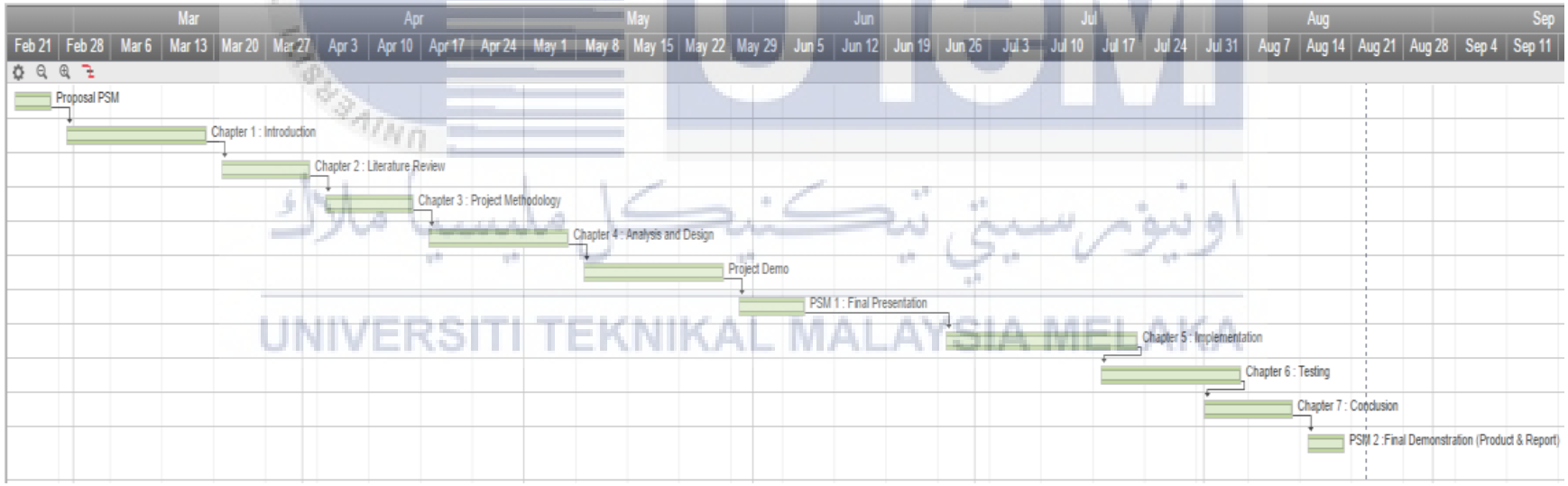


Figure 3.2 Gant Chart

3.4. Conclusion

As a conclusion, methodology is an important chapter to all of the projects, as it is use for planning, and having a milestone to take good care of the project schedule, to prevent the over time to happen. This chapter is not only about the duration of the project, but also the method and procedure used to carry out the project in a right and efficient way, to prevent wasting the time and energy. Without all of these planning and scheduling, the project will hardly goes smooth and many things will be unpredictable.

CHAPTER IV

ANALYSIS AND DESIGN

4.1. Introduction

In this chapter the project will be describe in a more detail and specific ways. Example like how to carry out the project and describes the solution to the problem that being stated in chapter 1. This is the chapter that will put much attentions on the detail which how to design the project. Besides the design, all requirements such as hardware and software that used on this project will be stated and analyze one by one too. The architecture of the system which includes block diagram and analysis in more detail are very important steps to be consider in doing this project.

4.2. Problem Analysis

According the problem statement in chapter 1, it is known that nowadays there are getting more and more people having some plants or even garden at home. But the problem is, plenty of them do not know when to water the plants as they could not see the water content inside the soil with their naked eyes. It is no doubt that it is hard to differentiate whether the plants is dry or wet, especially in the normal days, which the weather is not too extreme. People often water their plants according to the time, but sometimes the raining had just over and the soil is still too wet to be water, and sometimes the hot weather takes too much of the water in the soil, and the plants need the water immediately. Furthermore, when people is going out for vacation for long time, they do not have the automatically watering system to water their plants, which may eventually lead to the death of the plants.

Therefore, the objectives of this project is to build a system to monitor the soil humidity of the plant, to tell the users what is the soil humidity now through the LCD screen using percentage. Next, the system will have the automatically watering system which will automatic water the plant when it is under certain percentage which is indicate the soil is dry.

4.3. Requirement Analysis

4.3.1. Data Requirement

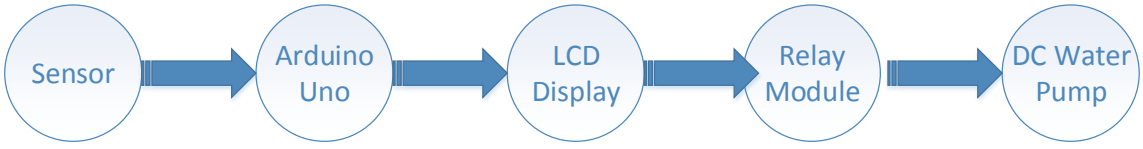


Figure 4.1 Data Flow

From the above figure, the project data flow is shown from the starting from the soil moisture sensor which is taking the voltage data to be fetch to the microcontroller Arduino. In the Arduino, the coding will translate the analog reading to digital reading. Next, the digital data will be fetch to the LCD display to show on the screen which is the percentage of the soil moisture humidity.

At the same time, the data will be fetch to the relay module as well to control the DC watering pump. When the relay module is signal by Arduino to switch on, the relay module will send the signal to water pump to switch on the pump. Likewise, if the relay module is signal by Arduino to switch off because it detects the wet soil, the relay module will signal the water pump to stop pumping the water.

4.3.2. Functional Requirement

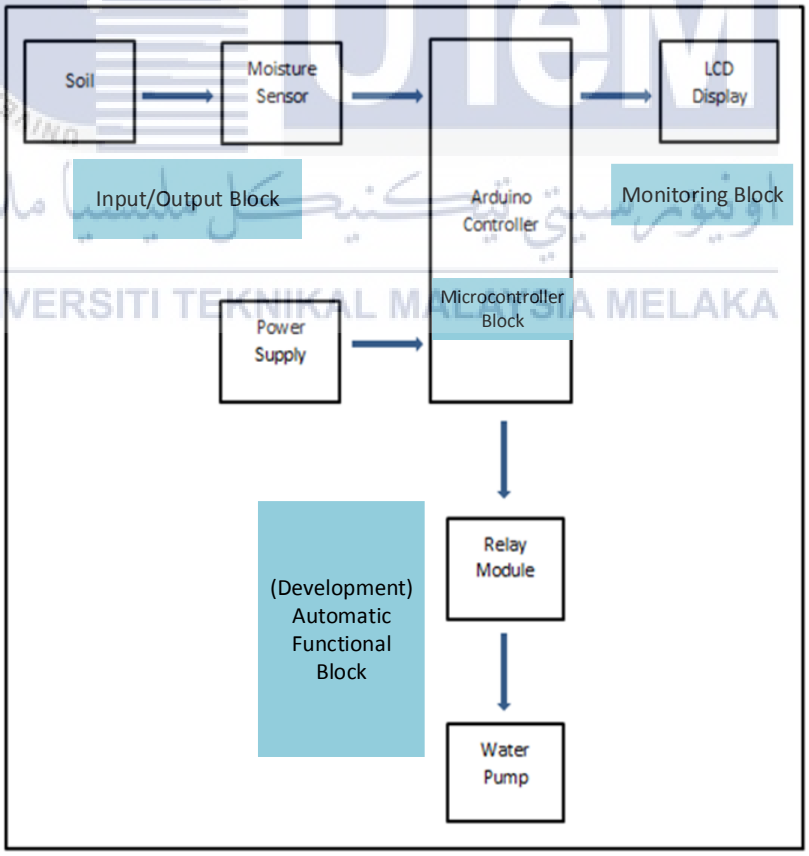


Figure 4.2 Block Diagram

This project is design to have several blocks as shown in the block diagram above, namely: input/output block, microcontroller block, development block and monitoring block.

- Input/output block

  Consist of one soil moisture sensor which takes the data from the soil. It depends on the humidity of the soil whether to send high or low voltage to the microcontroller to show that it is wet or dry. This sensor is directly connected to Arduino Uno.

- Microcontroller block

  Arduino Uno is the microcontroller which is the core hardware of this project. It receive the input and any activity from the soil moisture sensor and process in the board based on the coding. It is attached by all the hardware which all the hardware function is based on the Arduino coding. It also provide a platform for users to code and compile.

- Monitoring block

  LCD display is used to monitor the humidity of soil moisture by showing the percentage of the moisture on the screen. When the soil is dry the percentage will be lower and vice versa. Besides, it also show the pump status which is on or off, therefore users will know the current pump status.

- Automatic Functional block (Development block)

  Inside this block, it is mainly describe about the automatic watering function of this system, which consists of two main controlling hardware, which is relay module and DC watering pump. Based on the coding set in the Arduino board, the relay which has magnetic will control the open and close for the electric to pass

through the water pump. When the moisture level is below the threshold level, it will be tell to the system that the plant needs to be water now. Therefore, the relay module will automatically open the path for the electric to pass through to switch on the water pump to water the plant. After the system detects the sufficient of the water, the relay will close the path for electric and thus the water pump will be stop immediately.

4.3.3. Other Requirement

4.3.3.1. Hardware Requirement

1) Arduino Uno Rev3

The Arduino Uno is the microcontroller board based on the ATmega328P. Based on the figure below, It is shown that the Arduino board contain 14 digital input/output pins, 6 analog inputs, a USB connection, and ICSP header, a power jack, a 16MHz quartz crystal and a reset button. The Arduino is the central core of this project as it controls all the hardware that is attached to it. It contain a platform for coding when connect it to a computer with a USB cable with a self-download software named Arduino IDE.

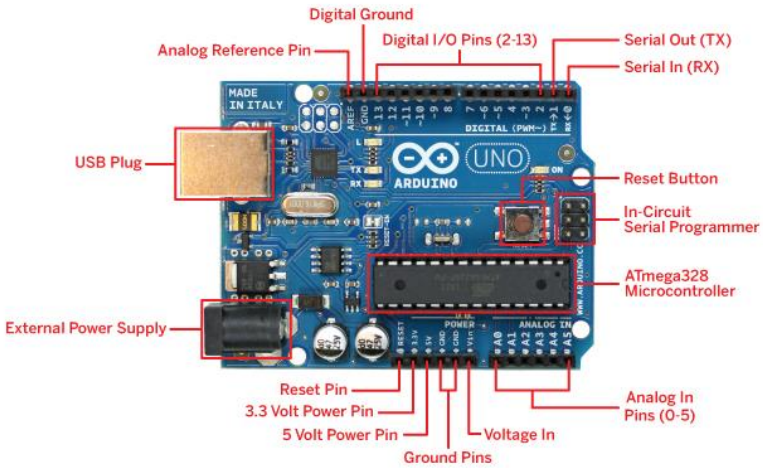


Figure 4.3 Arduino Uno Rev3

2) Soil Moisture Sensor YL-69

This sensor is generally used to detect the moisture content of the soil which is the humidity rate. It is connected to the microcontroller which is Raspberry Pi. The threshold value can be set in the sensor, and when the soil humidity is high, the DO port output will be high too, and vice versa.

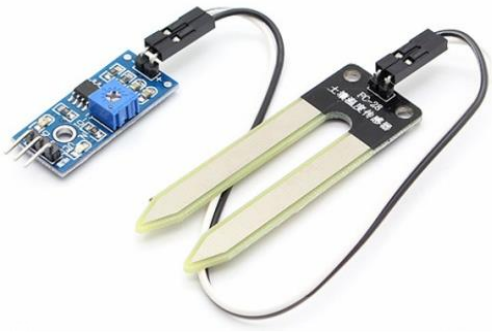


Figure 4.4 Soil Moisture Sensor YL-69

3) Liquid Crystal Display (LCD)

It is a flat panel display that uses light-modulating properties of liquid crystals. The backlight will produce the screen images for showing the content that comes from the coding in the Arduino. In this project, the LCD screen is used to show the moisture level of the soil and the pump status which is set it early in the Arduino board through coding.

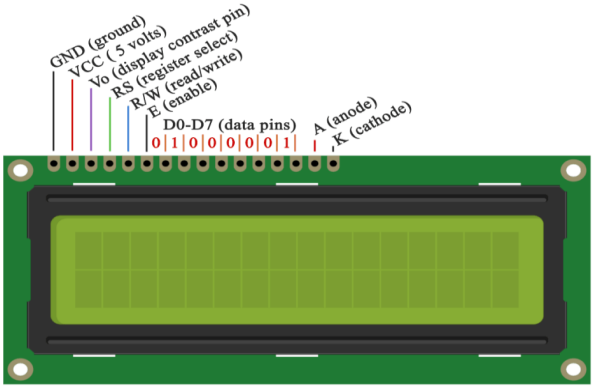


Figure 4.5 Liquid Crystal Display (LCD)

4) Relay Module

Relay module is a switch that controlled by an electromagnet. It is used to control the on and off of the DC watering pump by open or close the electric path that passes to the watering pump. It is control by the code from the Arduino.

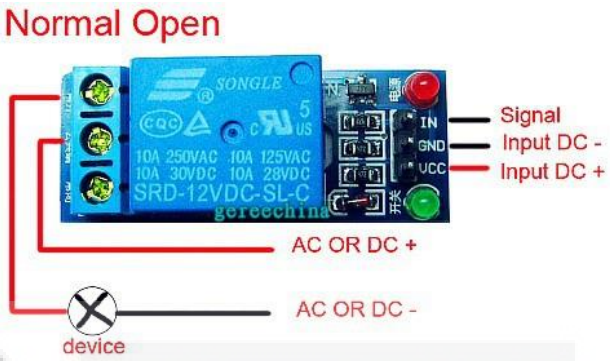


Figure 4.6 Relay Module

5) USB Cable

An USB cable is used to wire among the computer and the Arduino. The USB cable stated A is connect to the computer. While the cable that stated in B is connect to the Arduino.

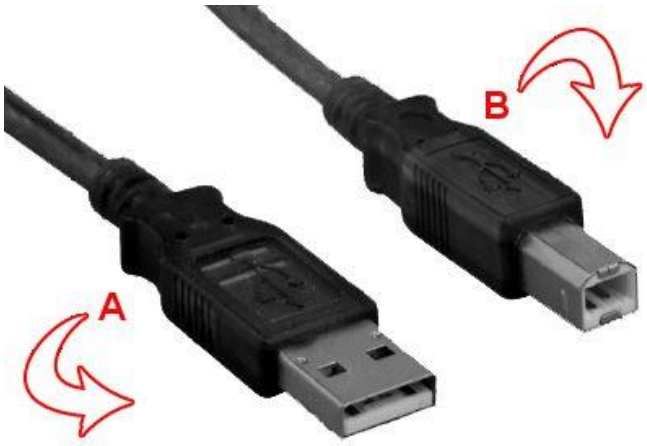


Figure 4.7 USB cable

6) Jumper Wire

Jumper wire are used with the breadboard to exchange the electrical signal from one another upon the circuit format that has been planned. Jumper wire can be in different size and length depends on the need. In this project, both male-to-female wire and male-to-male wire is used.

Figure 4.8 Jumper wire

7) DC Water Pump

The DC water pump used in this project is H-Bridge type. It is used to water the plant by sucking the water from the source and push out the water from the second hole to make the water process complete. It is control by the relay module which can be switch on and off automatically.

Figure 4.9 DC Water Pump

8) Integrated circuit (I2C) Module

I2C is a serial bus short distance protocol developed by Philips Semiconductor to enhance communication between the core on the board and various other ICs involved. I2C module is appropriate for interfacing to devices on a single board, and can be stretched across multiple boards inside a closed system. It has 16 pins which will directly connected to LCD. The brightness of the LCD screen can be adjust by tuning the display contrast button.

Figure 4.10 Integrated circuit (I2C) module

9) DC Adapter 12V

The DC adapter is use to supply power supply to the DC power pump. It consist of plugs which is male-ended and jacks which is female-ended for supplying direct current power at which flow of electrical charge carriers that always takes place in the same direction.

Figure 4.11 12V DC Adapter

4.3.3.2 Software Requirement

1. Arduino IDE

   This software is use for sketches, in sketches all program will be written and load to hardware. This software is using C++ programming language to write. There are two part of function which makes the program run which is setup() and loop(). Setup function is use to initialize pin nodes or begin serial, while the loop function is allow us to control, change and respond to Arduino board.

Figure 4.12 Arduino IDE

4.4. High Level Design

4.4.1. System Architecture

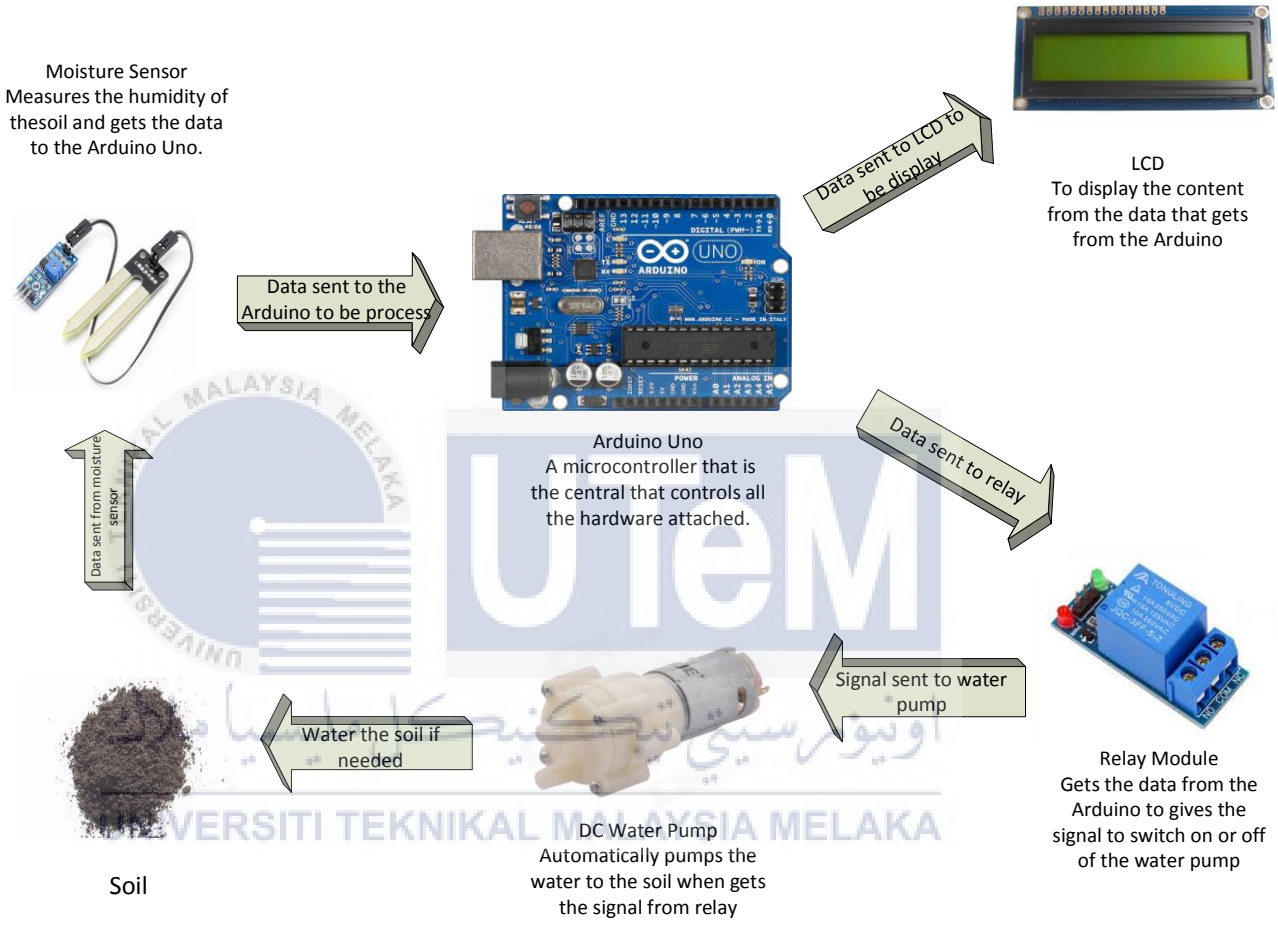
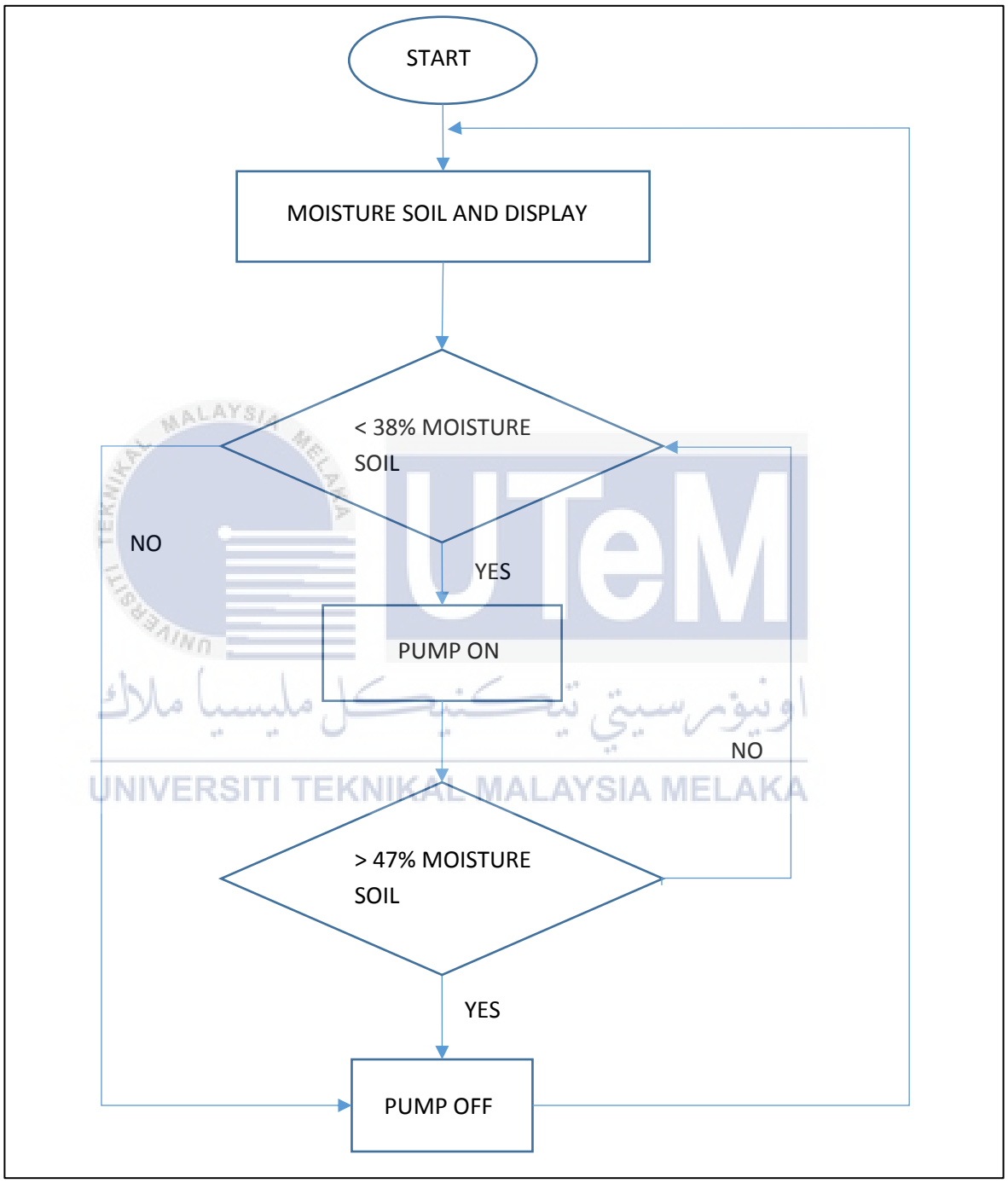


Figure 4.13 System Architecture of UnoPlanter

From the figure 4.13, it is shown that the system architecture of the UnoPlanter project. The whole process is shown starting from the sensor that detects the moisture from the soil, and the data is sent to Arduino to being process. Next the data will be sent to the LCD screen to display the desire data. At the same time, the data will be sent to relay to make a decision to signal the water pump to switch on or off.

4.5. Detailed Design

4.5.1. Flow Chart



4.14 Flowchart of UnoPlanter

Figure 4.14 shows the flow chart of UnoPlanter. From the flow chart, it is shown that when it detects the soil moisture which under 38%, it is consider dry and the water pump will be switch on. When it watered until 47% or more than it, the water pump will be automatically switch off to stop the watering process. It has to be notice that, when the percentage is dropped to less than 47%, it will check back to the first condition which is the under 38% of soil moisture, and when the percentage is among the 38% and 47%, the water pump will still in the switch off condition.

4.6. Conclusion

This is the important chapter before going for the implementation phase. Before implement all software and hardware requirement, it must be analyze and design carefully first. Besides, this chapter provide an information for determine the flow of project, by specifically explain each of the software and hardware requirement of this UnoPlanter system. By doing a great analysis and design, the implementation phase can go more smoothly and easily.

CHAPTER V

IMPLEMENTATION

5.1. Introduction

In this chapter will mainly describe the implementation of UnoPlanter which involve in the setup of the project including hardware and software. The expected output will be stated after all this implementation phase is completed.

5.2. Development Environment Setup

In this section will mainly discuss about the project setup about the UnoPlanter, all the hardware and software setup will be documented. The assembling of the hardware setup will be cover step by step to show in details. While the software setup will mainly cover about how to write and upload the code to the Arduino IDE.

5.2.1. Hardware Development Setup

In this project, it involves the use of all the hardware that stated in Chapter 4. All the hardware will be assemble to Arduino Uno. Arduino Uno is a microcontroller that use to control all the hardware that attached to it and let it function.

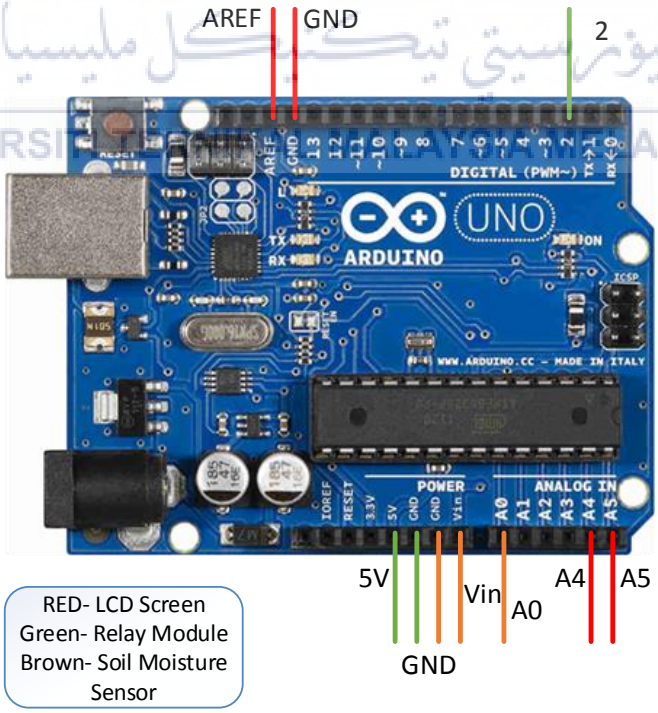


Figure 5.1 Detail of Arduino Pin

Figure 5.1 shows the detail of Arduino pin, based on the table below will show the pin that are used in this project.

Table 5.1 Hardware that will be attach in Arduino pin

Hardware	Pin
Soil moisture sensor	A0, Vin, GND
LCD screen	A4, A5, AREF, GND
Relay module	5V, GND, 2

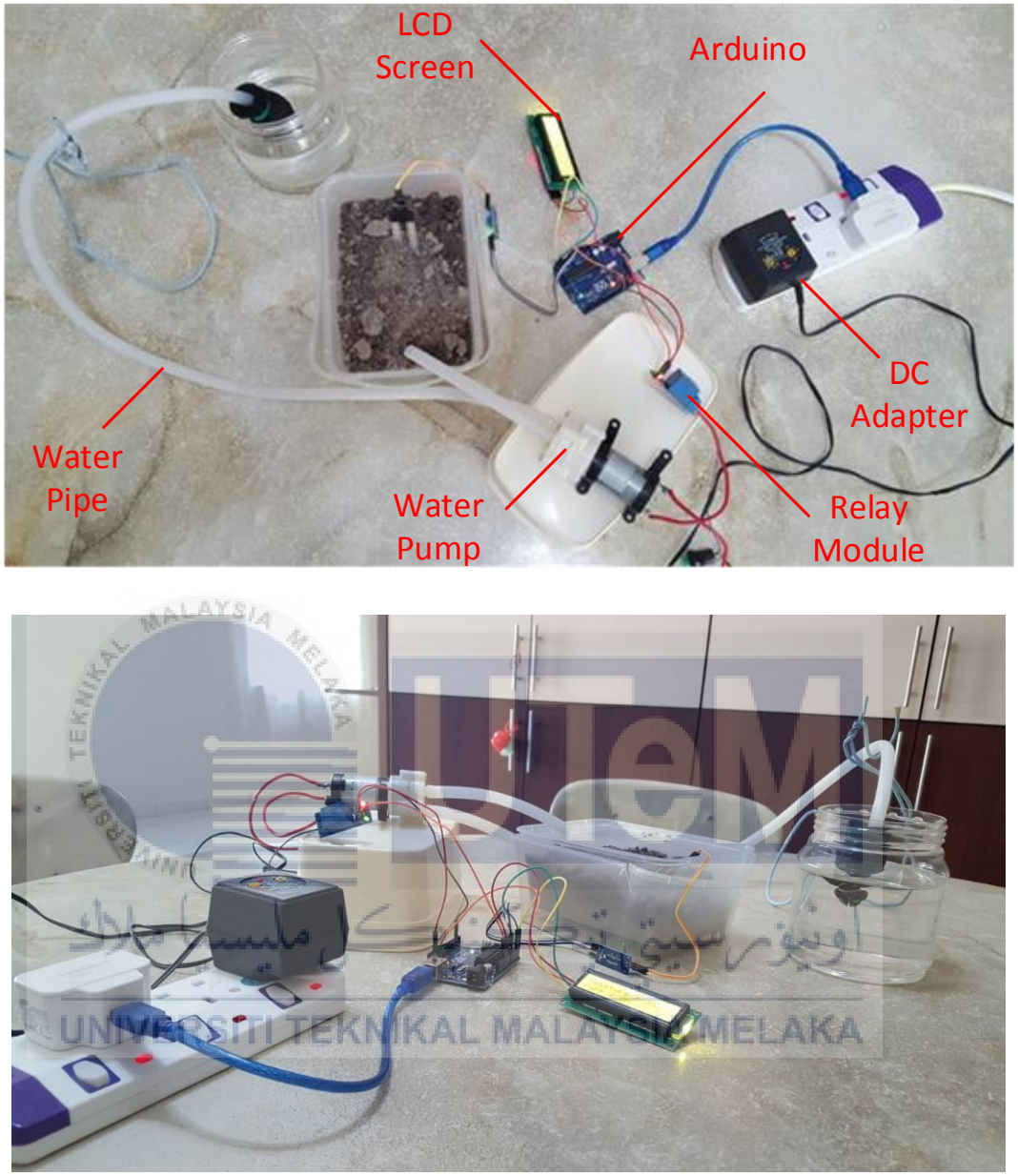


Figure 5.2 Hardware assemble for the whole project (Top and side view)

From figure 5.2 it is shown that the complete hardware setup of UnoPlanter which include the Arduino board and all the necessary attached hardware. This project is consider a complete prototype which is a system that can be used and function for the daily life. From the figure, it can be see that the Arduino is the center of this project which connects all the require hardware. From the soil moisture sensor which use to get the soil humidity data

from the plant, and it is transfer to the Arduino board to process and make decision. The LCD display will show the require content that the Arduino has sent, and at the same time, the data is sent to relay module to decide to switch on or off of the water pump. If the condition is require the water pump to switch on, the water pipe attached to the pump will begin to suck up the water from the water source, and the other side of water pipe will push out the water for the plant to complete the watering process.

Figure 5.3 Connection of Arduino to PC

Figure 5.3 shows the connection of Arduino to the laptop, which is also include the hardware setup. Arduino is connected with laptop using USB which will provide communication to each other. The software interaction of Arduino and laptop is done by this USB communication.

5.2.2. Software Development Setup

In this part, it will explain the software environment of this project. Figure 5.4 is shown to explain the software deployment.

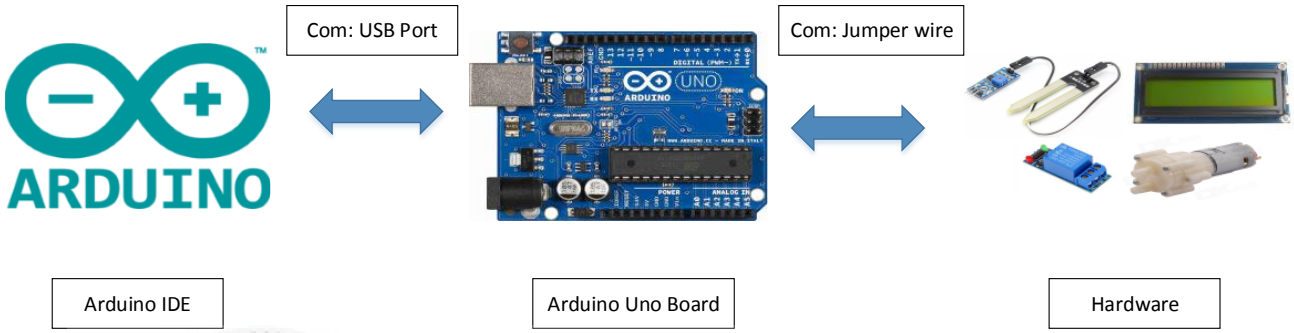


Figure 5.4 Deployment of software environment setup

From the figure above it is explaining the software flow for the entire project. The Arduino IDE is a platform used for code for the hardware to function. It is installed in the laptop and the software is download from Arduino website. After sketch and code, it is compiled and upload into the Arduino board. The laptop is connected with Arduino board with USB cable.

From the Arduino board, the code is use to command the function of the hardware. The Arduino board is connected with the hardware with jumper wires. It must be attached follow the pins so that it can be function correctly. The data will be get or send to the hardware such as soil moisture sensor, LCD display and relay from the Arduino board.

5.3. Software Configuration Management

5.3.1. Configuration Environment Setup

In this section there will be describing the configuration management for this project. It will be discuss into separate parts that listed below.

1. Arduino IDE Configuration

Arduino IDE will be used as the platform to write the code and upload into the Arduino board and hardware.

a) Installing Arduino IDE

This software can be found at https://www.arduino.cc/en/Main/Software.

b) Process of writing

After the Arduino IDE has installed, the code can be start to write and sketch for the project.

- Code that written in this project will be listed at appendix of this report.

2. Code for the soil moisture sensor to check the upper and lower sensor value

The soil moisture sensor must be check the upper and lower boundaries of the analog value before it is convert to digital value. It is because every soil moisture sensor upper and lower sensor value varies. Therefore it is better to check it when we change the soil moisture sensor or use it for the first time.

```
void setup() {

// initialize serial communication at 9600 bits per
second:

Serial.begin(9600);

}

void loop() {

// read the input on analog pin 0:

int sensorValue = analogRead(A0);
Serial.println(sensorValue);

delay(100);

}
```

Code for checking the sensor values

From the code above, it is used to check the sensor value of the soil moisture sensor. By connect the soil moisture sensor to the pin A0 on the Arduino board, and open the serial monitor in the Arduino IDE, we can see the soil moisture sensor value append for 0.1 second.

3. Creating the Liquid Crystal I2C library

There are two options to let the LCD display function which is wired all the LCD pin to the Arduino pin and code for it one by one, which is very troublesome and have a little complicated, because it is easily confuse when it comes to code the pin one by one. The second option which is attach a hardware name Integrated Circuit (I2C) Module beneath the LCD display and let it connect with Arduino board with 4 jumper wires. This is a method which is more convenient to let the LCD display function as it is compile the code into one library and insert into the main code.

4. Main code

Main code is include all the function of the hardware which is the soil moisture sensor, LCD display and the water pump function. It also include the insert of the Liquid Crystal library.

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>


#define WATERPIN 2


LiquidCrystal_I2C lcd(0x3F,16,2);
int i;
int soil= 0;
```

Part of the coding in main code (first part)

This part of coding is mainly show for the insertion of the Liquid Crystal library. The library is compressed into rar file and it is insert to the Arduino IDE. Other than library, it just include some declarations.

```
void setup()
{
  Serial.begin(9600);
  Serial.print("\r");
  lcd.init();
  lcd.init();

  lcd.backlight();
  lcd.clear();
  pinMode(WATERPIN, OUTPUT);
  digitalWrite(WATERPIN, LOW);

  lcd.setCursor(0,0);
  lcd.print("Arduino Watering");
  lcd.setCursor(5,1);
  lcd.print("System");
  delay(5000);

  lcd.clear();
  lcd.setCursor(2,0);
  lcd.print("SIAH'S PLANT");
  delay(5000);
```

```
  lcd.clear();
  readSensor();
  lcd.setCursor(0,0);
  lcd.print("Moisture = ");
  lcd.print(soil);
  lcd.print("%");
  lcd.setCursor(0,1);
  lcd.print("Pump : OFF");
}
```

Part of the coding in main code (second part)

This part of coding is mainly for the initialization of the LCD screen, and print the require content on the LCD screen. Such as the moisture percentage and pump status, and so on.

```
void loop()
{
  readSensor();
  lcd.setCursor(0,0);
  lcd.print("Moisture = ");
  lcd.print(soil);
  lcd.print("%");

  if (soil <= 38)
  {
  lcd.setCursor(0,1);
  lcd.print("Pump : ON ");
  digitalWrite(WATERPIN,LOW);

   while(soil <= 47)
    {
     readSensor();
     delay(100);
    }
  lcd.setCursor(0,1);
  lcd.print("Pump : OFF");
  digitalWrite(WATERPIN,HIGH);
  }
}
```


Part of the coding in main code (third part)

This part of coding is mainly for the printing of the percentage that reads from the soil moisture sensor. The reading is all comes from voltage that gets from soil moisture sensor whether it is wet or dry. Next, it still include the open and off of the water pump. From the code, it is shown that when the soil moisture percentage is below 38%, the water pump will be switch on automatically, and the LCD screen will show the water pump status is 'ON'. While the soil moisture percentage is reach to 47%, the water pump will be switch off automatically, and the LCD screen will show the water pump status is 'OFF'. As a matter of fact, the value of soil moisture is not more than 47% since average soils have a pore volume; while the value fall between 15% to 30% is consider dry (meteoblue, n.d.). Thus, the predefined threshold value is set at 38% to call for watering action before the humidity value is too low to prevent water loss from plants. The practice of this UnoPlanter is further supported by the statement of Vashita et al. (2016), who mentioned that it helps to avoid the plant soil moisture stress by providing a standard amount of water after it has detected lower than its threshold level. Hence, the use of this irrigation system can automatically control the required amount of water for a plant, which resulted in optimize the use of water in the process.


```
void readSensor()
{
  sensorvalue=analogRead(A0);
  sensorvalue=constrain(sensorvalue,400,900);
  soil=map(sensorvalue,400,900,99,0);
}
```

Part of the coding in main code (fourth part)

This part of coding is mainly for reading the sensor value from the soil moisture sensor. From the last part it already tested the upper and lower value for this sensor are 400 and 900 respectively. This value is analog value which needs to be convert into digital value to be display on the LCD screen. Therefore, the coding is converted the 400 into 100% and 900 into 0% respectively in the last sentence of the code.

5.3.2. Version Control Procedure

UnoPlanter Arduino Automatic Watering System product is the first project that documented. Therefore the version of this product is 1.0.

5.4. Implementation Status

In this part, it will explain about the status of development for the entire project whether it is the hardware assembling or the software configuration.

Table 5.2 Implementation Status

No	Component	Description	Duration of complete
1	Assemble of Hardware	Process of assembling all the hardware to the Arduino board	15 days

		based on the pin, and solder the wired for the require part to let the circuit pass through.	
2	Configuring and uploading script into Arduino IDE	Process of writing the code based on requirement and upload it to the Arduino board using Arduino IDE	10 days
3	Build for the prototype and testing	Design and build the best ways for the water to pump out from the water pump to the plant, from the diameter and the length of the pipe, and the height to put the pipe on the plant for the watering testing.	5 days

5.5. Conclusion

This chapter is mainly explain about the implementation of this project which cover from the assembling of hardware and the software setup. All the setup is clearly stated with steps and with pictures to provide a vivid process that actually for the produce of the prototype of this project in order to achieve the objectives.

CHAPTER VI

TESTING

6.1. Introduction

This chapter will discuss about the testing of the prototype – UnoPlanter. Last chapter have discussed the implementation of the project, which starts from the assembling of hardware until the configuration of the software. In this chapter, all the hardware and software will be testing to see whether it is fully function, and make sure it is a flawless prototype.

6.2. Test Plan

This test plan will explain about the basis for testing the system. Test plan is used to find the issues and problems that occur on this system.

6.2.1. Test Organization

This section is discuss about who will be responsible to test the system for its functionality. For this project, the testing process will be carried out by two tester which is system developer that is experience in testing system based product and another is a normal tester which is a normal person that are inexperience. The same testing activity and environment will be conducted for the two tester to know its feedback. The feedback results will be record and amendments will be make based on it.

1. System Developer
    - The person who is having experience in testing the system, and finding the possible error or issues which is the flaw of the system. He/she must make sure that the system that developed can be run smoothly and systematically.
2. Normal Tester
    - A normal person who is inexperience which will test the system and give his/her feedback that helps to enhance the system.

6.2.2. Test Environment

This section will be discuss about the test environment that use to test the project. First of all, the project will be test the connection of Arduino and PC using the USB cable to connect, and the code that is written in Arduino IDE is able to compile and upload successfully into the Arduino board. Next, is the testing of the sensor value of soil moisture sensor, which is put the soil moisture sensor in and out of the water and gets it upper and lower boundaries value in the serial monitor. Next hardware that need to test is the LCD display which can light on and show the content that are coded into the Arduino board. Lastly will be the testing of the relay module and the water pump, which the relay module is able to light on and off based on the soil moisture sensor and the water pump are able to suck in and pump out the water.

6.2.3. Test Schedule

This section discuss the period of time to carry out the testing by developer. During the testing process, if any error or flaw that is find out, the testing process must be stop and pass back to the implementation process to carry out the experiment again. It must be reconfigure the software or reassemble the hardware to find the possible error. The error that has been found out will be solve in the implementation phase. Therefore, the testing process is able to continue to carry out.

6.3. Test Strategy

This section describes the test strategy that is used for this project. For this project, the test strategy used is the black box test, this test is use to determine whether the project is able to function properly. Hardware and software will be test early to make sure that it is functioning according to the requirement. This is to ensure that the software and hardware could be work together without any errors, because they are integrated and none of it can be having the problems.

6.4. Test Design & Test Result and Analysis

6.4.1. Test Description

The test description discuss about the test purpose, test environment, test step and expected result for each scenario which are designed and documented. The test description is basically about the integration test and functionally test.

Table 6.1: Connection of Arduino to laptop and the Arduino IDE Test Case

Test	Connection and Arduino IDE Test
Test Purpose	1. To check whether the Arduino has successfully connected with laptop. 2. To check whether any error will occur while writing the code in Arduino IDE.
Test Environment	Windows 8.1

Expected Step	1. Check the Arduino IDE whether there is port connected to laptop. 2. Write the code and upload to Arduino without any error or problem.
Expected Result	Coding will run without any error and can be upload to Arduino board.

Connection of Arduino with laptop and Arduino IDE test

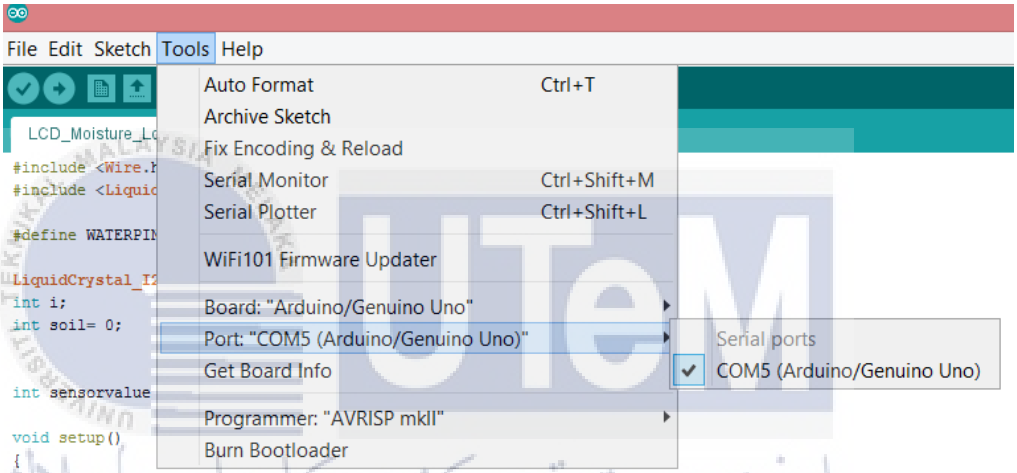


Figure 6.1 Connection of Arduino with the laptop

From figure 6.1, it is shown that the port is connect to COM5 which is the Arduino Uno successfully.

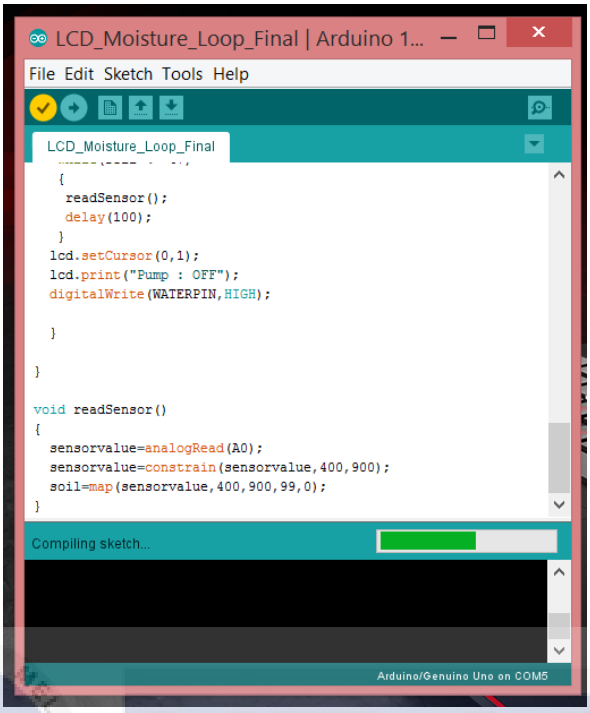


Figure 6.2 Verify the coding that written in Arduino IDE

Figure 6.3 Uploading the written code to Arduino successfully

From figure 6.2 and 6.3, it is shown that the Arduino IDE is used to write the code. The code has been verify and uploaded successfully into the Arduino Uno board.

Table 6.2: Soil Moisture Sensor Test Case

Test	Soil Moisture Sensor
Test Purpose	To test the sensor values and its functionality.
Test Environment	A glass of water and Arduino IDE.
Expected Step	1. Code to test the soil moisture sensor with Arduino IDE. 2. Open the serial monitor in Arduino IDE and see the sensor value for the dry condition. 3. Immersed the soil moisture sensor into a glass of water and see for the wet condition in the serial monitor in Arduino IDE too.
Expected Result	The soil moisture sensor is light up in the controller when it is switched on, and it can show the lower and upper boundaries of the sensor value in dry and wet conditions.

Soil Moisture Sensor Test

```
void setup() {

// initialize serial communication at 9600 bits per
second:

Serial.begin(9600);

}

void loop() {

// read the input on analog pin 0:

int sensorValue = analogRead(A0);
Serial.println(sensorValue);

delay(100);

}
```

Code for checking the sensor values

Attached the soil moisture sensor to the Arduino board, and upload this coding to the Arduino board to start the testing of the sensor values.

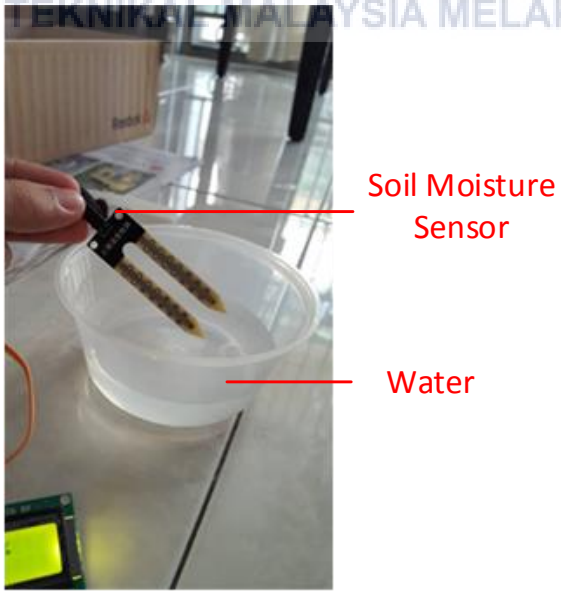


Figure 6.4 Soil Moisture Sensor that is dry

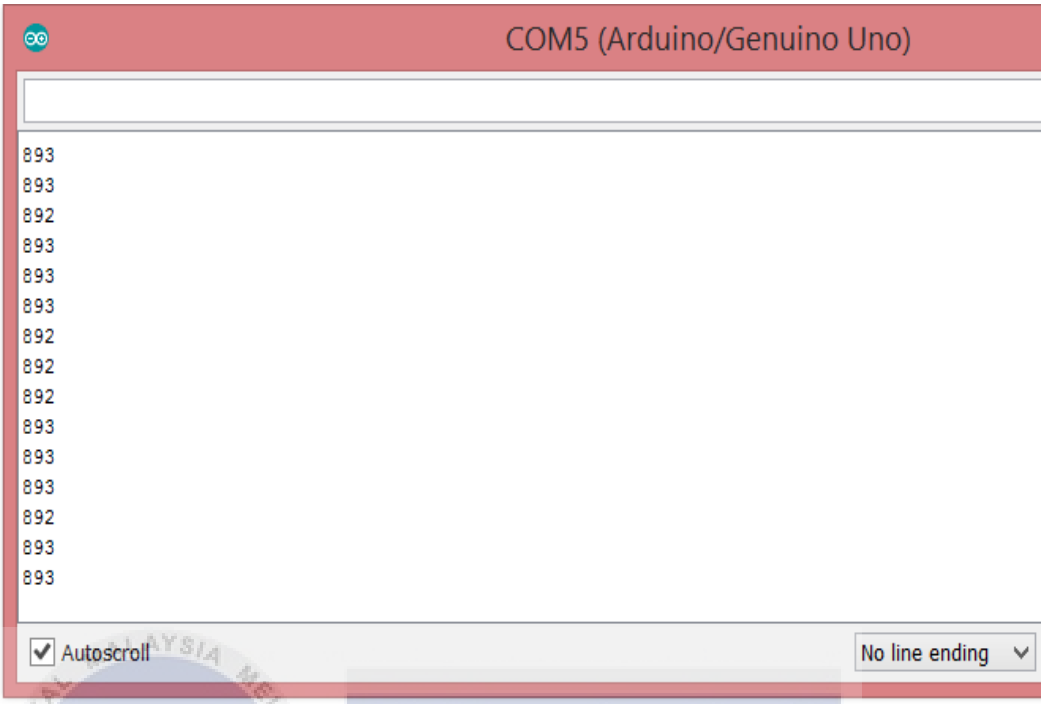


Figure 6.5 Dry sensor value shown in serial monitor

From the figure 6.4 and 6.5, it is shown that when the soil moisture sensor is in the dry condition, the sensor value shown in serial monitor is high which is around 893, it will be round up to 900 to use as the upper boundary.

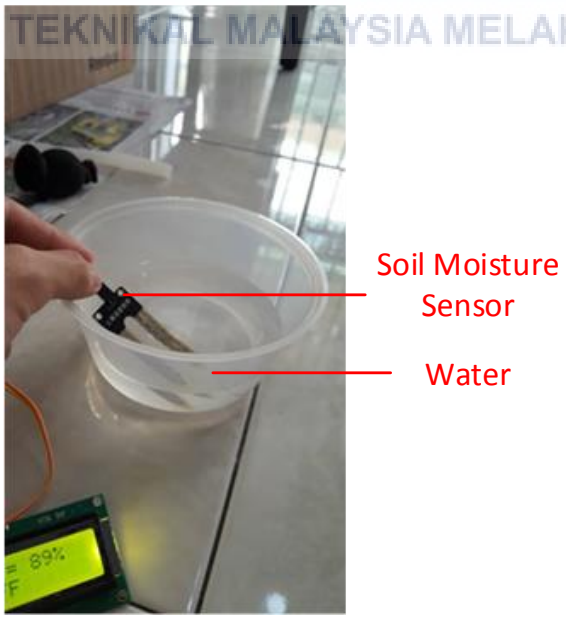


Figure 6.6 Soil Moisture Sensor that is wet

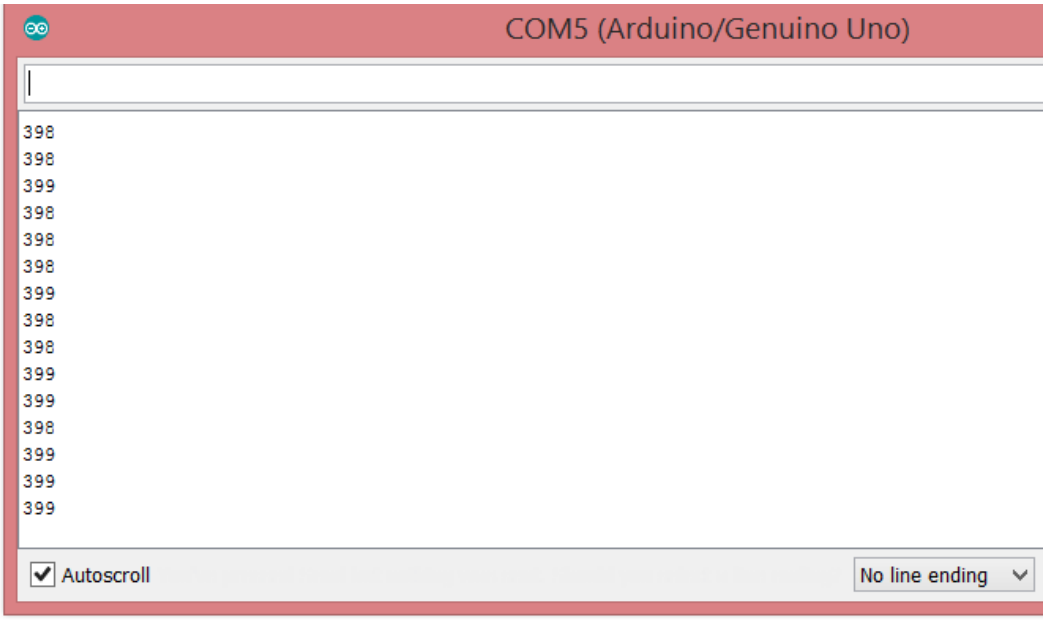


Figure 6.7 Wet sensor value shown in serial monitor

From figure 6.6 and 6.7, it is shown that when the soil moisture sensor is in the wet condition, the sensor value shown in serial monitor is low which is around 399, it will be round up to 400 to use as the lower boundary.

Table 6.3: Monitoring System Test Case

Test	Monitoring System
Test Purpose	To test the function of monitoring system.
Test Environment	Arduino IDE, Soil Moisture Sensor, LCD Display and Soil (Wet and Dry).
Expected Step	1. Code using the Arduino IDE based on the function for soil moisture sensor and LCD Display. 2. Put the soil moisture sensor into the dry soil and see the content in the LCD screen. 3. Repeat the step 2 by putting into the wet soil.

Expected Result	The soil moisture sensor must be able to differentiate between the wet and dry soil by showing the percentage on the LCD display screen. If the soil is dry the percentage shown should be low, while the soil is wet the percentage shown should be high.

Monitoring System Test

Figure 6.8 Dry soil

From figure 6.8, it is shown that the soil moisture sensor is immersed into a dry soil, and the LCD display is able to show the content which is the 3% moisture percentage and the pump status is 'ON'. This means that the soil is insufficient of water and the water pump will be switch on to start the watering process.

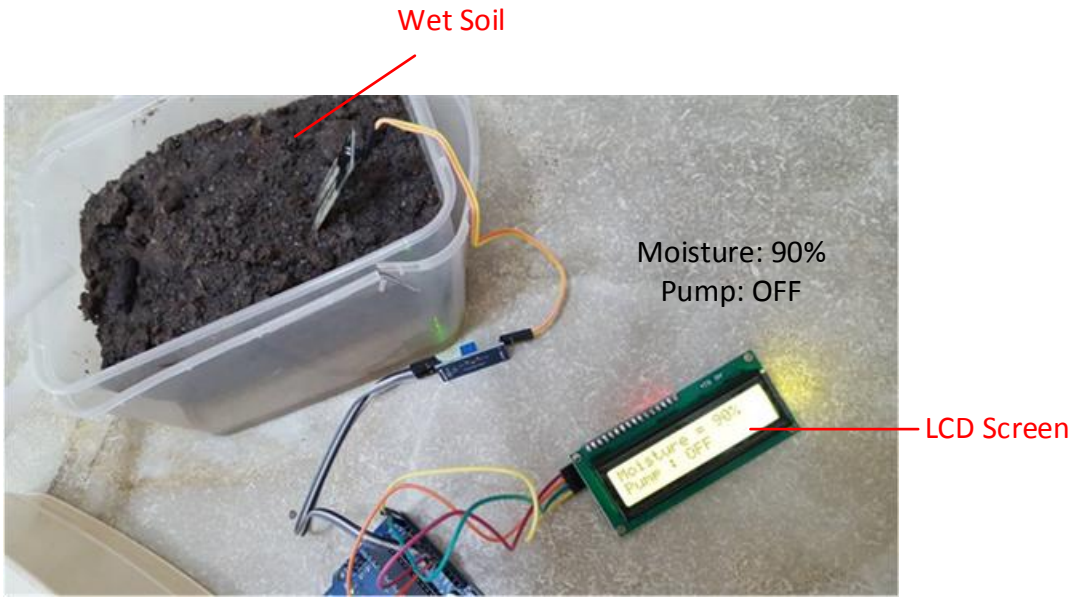


Figure 6.9 Wet soil

From figure 6.9, it is shown that the soil moisture sensor is immersed into a wet soil, and the LCD display is able to show the content which is the 90% moisture percentage and the pump status is 'OFF'. This means that the soil is sufficient of water and the water pump will be switch off to stop the watering process.

Table 6.4: Automatic Watering System Test Case

Test	Automatic Watering System
Test Purpose	To test the function of automatic watering system.
Test Environment	Arduino IDE, Soil Moisture Sensor, Relay Module and Water Pump
Expected Step	1. Code using the Arduino IDE based on the function for the automatic watering system.

	2. Place for the soil moisture sensor into the soil to detect the soil moisture. 3. Prepare the water source for the water pump to suck in the water and push out from the other side.
Expected Result	If the soil moisture sensor detects the water inside the soil, the relay module will light up and it will switch on the water pump. Therefore, the water pump will start to suck in the water and push out from the other hole that will goes to the soil. If the soil moisture sensor detects the dry soil, the relay module will not light up and the water pump will not function too.

Automatic Watering System Test

After having a full code uploaded into the Arduino board, the system will be able to automatically water when the soil moisture sensor is in dry condition.

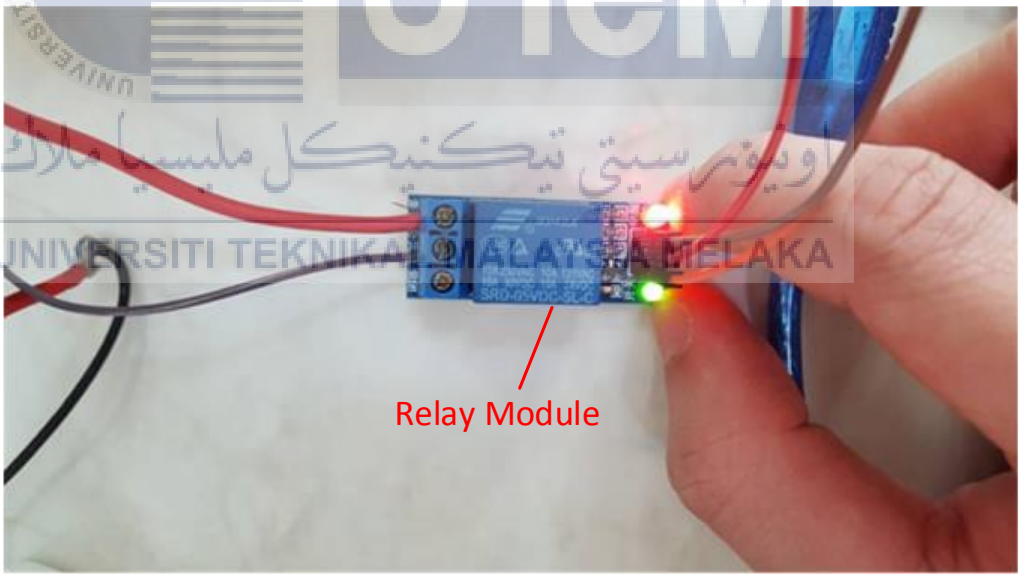


Figure 6.10 Relay module that is switched on

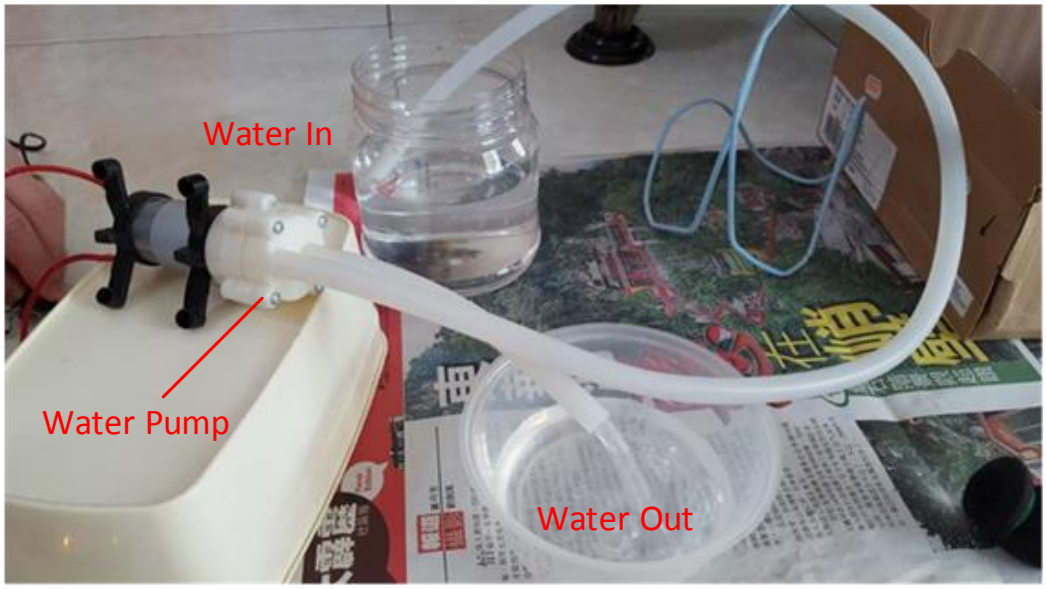


Figure 6.11 Start the watering process

From the figure 6.10 and 6.11, it is shown that when the relay module is switched on, it will be light up and the water pump will be starting the watering process, which is suck in the water from the water source and push out the water from the other hole. This process is start to function when the soil moisture sensor is in the dry condition.

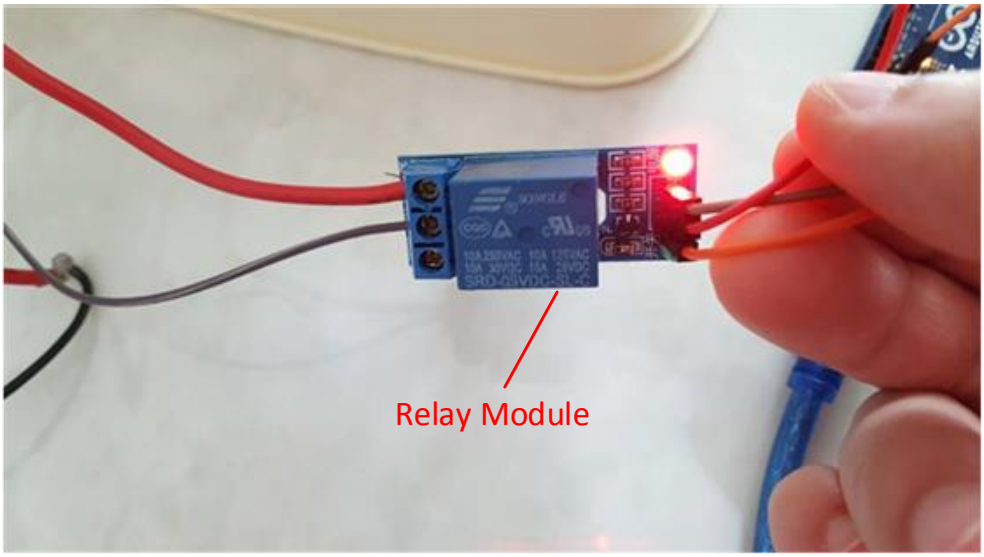


Figure 6.12 Relay module that is switched off

When the soil moisture sensor is immersed into the water which is in the wet condition, the relay module will be switched off which the green light does not light up like figure 6.12. Therefore, the water pump will be switch off too which the watering process will be stop.

6.5. Conclusion

As a conclusion, this chapter is vital for the whole project as it is the test bed for testing all the functions of the system, which helps to identify any problems or malfunction that could be happen. It also helps to test the functionality of the product prototype, to check if it is user friendly and easy to use for the normal users. Testing phase is very important in any of the product which could never ever be skip.

CHAPTER VII

PROJECT CONCLUSION

7.1. Introduction

The final chapter of the report will be discuss the summarization of overall of the project, from the beginning until the end of it. The objective that being stated should be achieved and all the testing should be go smoothly and without any error. Besides, this chapter will discuss the project summarization, project weakness and strengths, contribution of the project and what are the limitation of this project. All the discussion in this chapter is important for the future improvement of this product, so that the product can be made more perfect and flawless.

7.2. Project Summarization

This project is about the automatic watering system for the home gardener which is name UnoPlanter. UnoPlanter is designed for the automatic water the plants without any human hands. Therefore, people will not be afraid that the plant die because of forget to water it. Besides, it also has a LCD display to show the percentage of the soil moisture. It will keeps on reporting the current moisture of the plant to let the user know whether the plant is in sufficient water or not.

The first objective of this project is to detect moisture of the soil using the soil moisture sensor. By simply using a soil moisture sensor and some simple code, it can detect and check the soil humidity of the plant. This is cooperate with the LCD screen to show the moisture percentage of the plant on the screen while the soil moisture sensor gets the data.

The second objective is to display the moisture of the soil and pump condition using LCD screen. By using the LCD display, it can see the soil humidity in digital form which is easier for the user to read and know the current moisture condition of the plant. Besides, the LCD display also show another information which is the pump condition. It will be showing ON or OFF based on the percentage of the soil humidity.

The third objective is to develop an automatic watering system. After getting the soil moisture level from the soil with the soil moisture sensor. The system will decide to switch on or off the water pump to water the plant based on the soil humidity. When the soil is too dry, the system will switch on the water pump to automatically water the plant, and vice versa.

This Arduino automatic watering system has been developed, tested and the product – UnoPlanter is consider a fully function prototype that has met all the requirements and objectives.

1. Project Weaknesses:
    - **Power consume**, as the system needs to be operate all of the time to keep on detecting the soil humidity moisture and automatically water the plant. Besides, it is using two plugs (one for Arduino board and one for the water pump). Therefore it is consider quite consume of the electricity.
    - **Does not advance enough to water the plant based on the type of the plant.** Not every plant needs the same amount of water, as the plant species need the water amount varies. Example like cactus, it only need small amount of water every week, this system which will automatically water until the soil is wet is not suitable for this kind of plant.
    - **Sensor which have limited function.** This project is using the cheapest sensor that can found in the market, therefore the function and its sensitivity is not as good as we think. The sensor only function when it comes to touch with water, when it leaves the water and touch the dry surface, it will not detect any changes.

2. Project Strength:
    - **A monitoring system for the plant.** With the detection of the soil moisture sensor and the displaying monitor of the LCD screen, this project is consider a monitoring system for the plant. As it continues to detect the soil moisture and shows the percentage on the LCD display to let user know the current moisture condition.
    - **An automatic watering system.** The main purpose of this project is to automatic water the plant without any manually control. It is fully automatically as the user just put the sensor into the plant and switch on the power, and it can be leave there for all the time.
    - **Easy installation and portable.** This project is easy to use as it does not user to configure or install anything. User just need to put on the water pipe on the water pump and prepare the water source, next is put the sensor into the soil and switch on the power for both water pump and Arduino, all will

be go smoothly. Furthermore, it is easy to carry as it is light and does not occupy much of the space.

7.3. Project Contribution

UnoPlanter is aim to provide a monitoring and automatically watering system for the home gardener that has small or medium pots. It is developed to help them to automatically water the plant as they may have forget to water due to everyday busy schedule. Besides, they will be having vacation sometimes which will leave the house for weeks and this product will surely having a great help to them as it is automatically water the plant. Therefore, the users are able to go for vacation without to worry that their plant will die because of insufficient of water.

Besides the watering system, UnoPlanter also provide the monitoring function which users are able to check the soil moisture based on the reading on the LCD display. Therefore, users can judge the soil humidity based on the percentage and will not over water it. This function is to let the users do the watering process themselves while not wasting the water at the same time.

7.4. Project Limitation

In this project there are some limitation. The first one is the electricity, which is the power source for the water pump and the Arduino board. When it is carry to the place that does not have the plug or the house is having electricity termination, this product will be unable to function anymore.

Besides, the soil moisture sensor will be quite short and small if put into a bigger pots. It is hard to detect the water coming when the pot is too big and huge. Therefore, when the sensor gets the detection maybe the pot is already full of water which is over water.

7.5. Future Works

There still have many improvements for this system to works in perfection, and a lot of function can be add. For example this system can be add one more hardware that is loaded with battery packs, so that user can carry and use it anywhere even there is no plugs or power supply.

Besides, the monitoring system can be improve by doing a user interface which can be seen on the computer screen. By doing a more delicate user interface which can record all the previous soil moisture data, the users are able to clearly look for the previous moisture condition and have a clearer picture for the condition of their plants. This function will require another hardware which is a WiFi card that use to connect with the computer.

Furthermore, this system can be improve by sending the notifications to the users to tell about the current condition which is more user friendly. Therefore the users are able to track their plants all the time even when they are out to work or vacation.

7.6. Conclusion

The conclusion of this project is the system that develop – UnoPlanter is successful and is consider as a prototype which can be use and function fully for the users. This project has achieved its objectives which can check and monitor for the soil moisture percentage and let the users know the soil moisture condition. While the other objective is able to automatic watering the plant without any manually control, which gives a pretty big convenience for the users. Last but not least, it is a successful project that meets the requirement and contributes for the human kind.

REFERENCES

Chate, B. and Rana, P. (2016). Smart Irrigation System Using Raspberry Pi. *International Research Journal of Engineering and Technology (IRJET)*, 3(5), pp.247-249.

Devika, S., Khamuruddeen, S., Khamurunnisa, S., Thota, J. and Shaik, K. (2014). Arduino Based Automatic Plant Watering System. *International Journal of Advanced Research in Computer Science and Software Engineering*, 4(10), pp.449-456.

Jadhav, S. and Hambarde, S. (2013). Android based Automated Irrigation System using Raspberry Pi. *International Journal of Science and Research (IJSR)*, 5(6), pp.2345-2351.

meteoblue. (n.d.). *Soil and surface*. [online] Available at: https://content.meteoblue.com/en/help/weather-variables/soil-and-surface.

Ojha, M., Mohite, S., Kathole, S. and Tarware, D. (2016). Microcontroller Based Automatic Plant Watering System. 5(3), pp.25-36.

Vashista, A., Rathore, H. and Jain, G. (2016). Automatic Gardening System using Arduino. *SSRG International Journal of Electronics and Communication Engineering (SSRG-IJECE)*, 3(8), pp.119-120.

APPENDIX A

Grantt Chart

Gantt Chart of Project Activities

Week / Task	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
PSM Proposal																			
Introduction																			
Methodology Stage																			
Analysis Stage																			
Design Stage																			
Implementation Stage																			
Integration Stage																			
Testing Stage																			
System Demonstration																			

Presentation																		
Final Report																		

APPENDIX B

Coding

```
void setup() {

// initialize serial communication at 9600 bits per
second:

Serial.begin(9600);

}

void loop() {

// read the input on analog pin 0:

int sensorValue = analogRead(A0);
Serial.println(sensorValue);

delay(100);

}
```

Code for checking the sensor values

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

#define WATERPIN 2

LiquidCrystal_I2C lcd(0x3F,16,2);
int i;
int soil= 0;

int sensorvalue = 0;

void setup()
{
  Serial.begin(9600);
  Serial.print("\r");
  lcd.init();
  lcd.init();

  lcd.backlight();
  lcd.clear();
  pinMode(WATERPIN, OUTPUT);
  digitalWrite(WATERPIN, LOW);
```

```
  lcd.setCursor(0,0);
  lcd.print("Arduino Watering");
  lcd.setCursor(5,1);
  lcd.print("System");
  delay(5000);


  lcd.clear();
  lcd.setCursor(2,0);
  lcd.print("SIAH'S PLANT");
  delay(5000);


  lcd.clear();
  readSensor();
  lcd.setCursor(0,0);
  lcd.print("Moisture = ");
  lcd.print(soil);
  lcd.print("%");
  lcd.setCursor(0,1);
  lcd.print("Pump : OFF");



}
```

```
void loop()
{
  readSensor();
  lcd.setCursor(0,0);
  lcd.print("Moisture = ");
  lcd.print(soil);
  lcd.print("%");

  if (soil <= 38)
  {
  lcd.setCursor(0,1);
  lcd.print("Pump : ON ");
  digitalWrite(WATERPIN,LOW);

   while(soil <= 47)
   {
    readSensor();
    delay(100);
   }
  lcd.setCursor(0,1);
  lcd.print("Pump : OFF");
  digitalWrite(WATERPIN,HIGH);
  }
}
```

```
void readSensor()
{
  sensorvalue=analogRead(A0);
  sensorvalue=constrain(sensorvalue,400,900);
  soil=map(sensorvalue,400,900,99,0);
}
```

Main Code