UNIVERSITI TEKNOLOGI MALAYSIA

DECLARATION OF THESIS / UNDERGRADUATE PROJECT REPORT AND COPYRIGHT

Author's full name : **HAMED RIYADH DAWOOD**

Date of Birth : **27 February 1989**

Title : **SMART JUMPING OBSTACLE FOR EQUESTRIAN SHOW**

Academic Session : **2014/2015**

I declare that this thesis is classified as:

☐	**CONFIDENTIAL**	(Contains confidential information under the Official Secret Act 1972)*
☐	**RESTRICTED**	(Contains restricted information as specified by the organization where research was done)*
✓	**OPEN ACCESS**	I agree that my thesis to be published as online open access (full text)

I acknowledged that Universiti Teknologi Malaysia reserves the right as follows:

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

Certified by:

SIGNATURE	**SIGNATURE OF SUPERVISOR**
201006M10037	Assoc. Prof. Dr. Rosbi bin Mamat
(NEW IC NO/PASSPORT)	**NAME OF SUPERVISOR**
Date: 14 JANUARY 2015	Date: 14 JANUARY 2015

NOTES: * If the thesis is CONFIDENTAL or RESTRICTED, please attach with the letter from the organization with period and reasons for confidentiality or restriction.

"I hereby declare that I have read this thesis and in my/our*
opinion this thesis is sufficient in terms of scope and quality for the
award of the degree of Bachelor of Engineering (Electrical - Mechatronics)"

Signature	:	……………………………........
Name of Supervisor	:	Assoc. Prof. Dr. Rosbi bin Mamat
Date	:	14 JANUARY 2015

SMART JUMPING OBSTACLE FOR EQUESTRIAN SHOW

HAMED RIYADH DAWOOD

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Bachelor of Engineering (Electrical - Mechatronics)

Faculty of Electrical Engineering
Universiti Teknologi Malaysia

JANUARY 2015

I declare that this thesis entitled " *Smart Jumping Obstacle* " is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature	:	....
Name	:	HAMED RIYADH DAWOOD
Date	:	14 January 2015

Specially dedicated to my parents and my family.

ACKNOWLEDGEMENT

In the name of Allah, the Beneficent, the Merciful. First, I would like to take this opportunity to thanks Allah, the Almighty, on whom we depend for guidance and help.

Then, I would like to express my deepest gratitude to my advisor, Assoc. Prof. Dr. Rosbi bin Mamat for his support, help and patience. His suggestions and contribution helped me all the way until the project was finished.

I am forever indebted to my family who always supports me morally and financially. Whatever happens to me, I am sure they will be there for me to support and guide me.

Last but not least, I would like to express my appreciation to all my friends for their help, encouragement and Dua'a.

ABSTRACT

Equestrian show jumping is a sports where the horse is trained to jump over a well-designed obstacle. The basic jumping obstacle consist of three main parts, wings, rail and cups. However, there is no electromechanical system designed for the equestrian obstacle so far. So, the action of changing the height is done manually which is quite tedious and boring task. In this project, an electromechanical system of equestrian obstacle was designed and fabricated.An ultrasonic sensor is used to calculate the actual height. Then, it will send information to Arduino UNO which is the microcontroller. Arduino UNO will compare between the actual height and desired height and send commands to the motor driver that will control the motors. By doing this, the obstacle rail will move electromechanically up or down without the need to hire an assistant. This is all done on a jumping obstacle prototype which has been built where the wings and rail are made of woods while the cup is a 3D printed item.

ABSTRAK

Persembahan lompatan ekuestrian adalah sukan di mana kuda dilatih untuk melompat melepasi halangan yang telah direka khas. Asas bagi struktur halangan ini terdiri daripada tiga bahagian utama, sayap, rel dan cawan. Walau bagaimanapun, masih tidak ada komponen elektromekanikal yang terlibat dalam halangan ekuestrian setakat ini. Jadi, tindakan untuk menukar ketinggian halangan secara manual adalah membosankan dan menjemukan. Dalam projek ini, sensor ultrasonik digunakan untuk mengukur ketinggian sebenar bagi halangan. Kemudian, maklumat ketinggian ini akan dihantar kepada Arduino UNO yang menjadi sebagai mikropengawal. Arduino UNO akan membandingkan antara ketinggian sebenar dan ketinggian yang dikehendaki lalu menghantar arahan kepada pemandu motor yang akan mengawal pergerakan motor. Dengan kaedah ini, halangan akan bergerak secara elektromekanikal tanpa perlu mengupah seseorang untuk melakukannya. Semua ini telah dibina dan diaplikasikan menjadi prototaip untuk sistem melompat halangan di mana sayap dan rel diperbuat daripada kayu manakala cawan diperbuat daripada hasil pencetak 3D.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATION

LCD	-	Liquid Crystal Display
DC	-	Direct Current
PWM	-	Pulse Width Modulation
FTDI	-	Future Technology Devices International
PC	-	Personal Computer
USB	-	Universal Serial Bus
AC	-	Alternating Current
I/O	-	Input / Output
Gnd	-	Ground
NMOS	-	N-type metal-oxide-semiconductor
cm	-	Centimeter
LED	-	Light-Emitting Diode

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Introduction

This chapter will discuss and explain the background of the project, the problem statement, the objective, the purpose of the study, the scope and thesis organization. The background discuss the significant information that help to improve the basic ordinary concept of the targeted project. The objective and scope will be described. Finally, the organization of thesis gives a basic idea of each chapter in the report.

1.2 Background of the Project

Equestrian show jumping is one of the most interesting sports where the horse is trained to jump over a well designed obstacle in which the stretching muscular body of the horse can be improved and built perfectly. This sport is all about the good judgment and high skillful ability of the rider to take control of his trained horse by pulling and pushing the bridle left, right and both at the same time, also using the legs to poke the horse's abdomen for making the horse obeys the order and follows what's the rider ask him to do. The bridle is basically the main tool for taking control of the horse where the skilled rider can use it technically to move his/her horse freely and in a convenient way that let the horse easily follow the orders and focus on the task or obstacle that he is heading to. What makes the sport more interesting is that the challenging factors, the variety of artful obstacles design, the good communication between the rider and his/her horse and finally the different skill level of competitors that urges riders and horse to perform specially when there is a well organized and arranged environment and neat playground.

Faults and penalties are applied strictly in the equestrian show jumping events. There are two types of penalties, Elimination and 4 Faults. One of the penalties applied is that if the horse committed a knock-down of the rail/bar of an obstacle it cause him a 4 Fault penalty. The fall of the rider, horse or both causes an Elimination penalty. Moreover, failure of breaking the timer starting and finishing would results in a Elimination penalty. However, if the horse only touches the rail of an obstacle without knocking it down, it will not cause or result in any type of penalty.

The basic jumping obstacle components consist of three main parts, wings, rail and cups. Wing is the vertical standard that located at both sides of the jumping obstacle, left and right where each obstacle must have two wings in which the cups would be stuck in to hold the rail. Cups are the grippers where the rail is held by, two cups for each rail to hold where the rail must be moving freely on the cups so that in case of strong touch, it drops. Rail is the horizontal bar/pole that is held by the cups and shows how high a horse should jump over while the max height of an obstacle is 2 meters.

A lot of jumping obstacle types and designs do exist in the equestrian world. Basically, the simplest type is the vertical which consists of two wings, two cups on rail, see Figure 1.1, More types of obstacles would be discussed in chapter 2, Oxer, Filler and so on. Artful variation of the obstacle design is highly recommended to attract the horse to jump and make him get used to multiple design. The most important thing about an obstacle is to be attracting and safe, which means well designed and wide, so the horse comes towards it confidently. Variety of obstacle design teaches the horse how to not be afraid of taking new obstacles types, enables the rider to take more control of the horse in case of being stubborn and improves both skills, rider and horse.

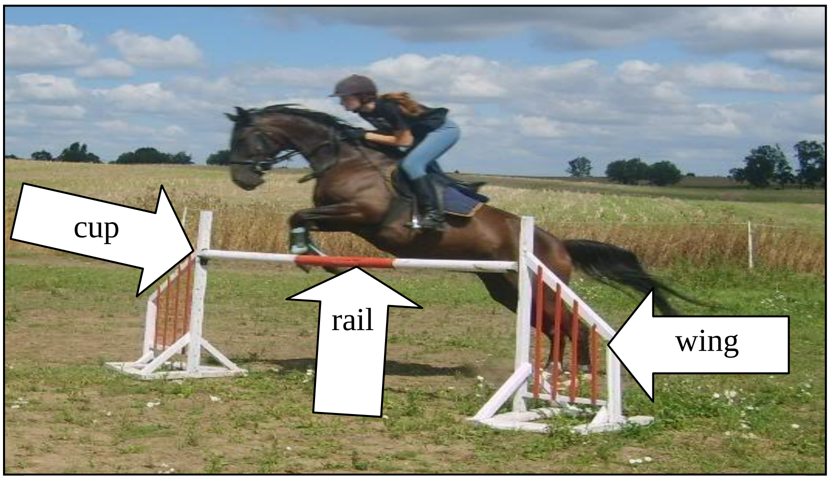

Figure 1.1: Basic Type of Horse Jumping Obstacle

Horse jumping obstacles are being used in many fields and places around the equestrian fans and societies. Private stables use the obstacle jumping for the sake of training kids, horses and individual athletes, besides having fun. Schools and clubs use obstacles to train and teach people and horses. Also, equestrian show jump events uses obstacles in the main field of competition and in the warming-up field.

1.3 Objectives

The objectives of this project are:

1) To design an obstacle which involves electrical and mechanical parts.
2) Adjusting the obstacle height automatically in a very convenient way.

1.4 Scope

Smart jumping obstacle project is covering the part of changing height level of a rail where two motors will be located at each wing bottom shafted with gears to move the chain with sticking cups, feed backed by a ranging sensor. The project will be applied with smaller dimension but not standard. The obstacle will be a attached with the following components:

- Interface programming between Arduino UNO microcontroller and ultrasonic sensor
- power motors will be used

1.5 Problem Statement

There was no electromechanical system designed for the equestrian obstacle so far while the need of changing height level is obvious where the whole obstacle developments were in the field of art design. Therefore, The height level of the current jumping obstacle is changed manually, which is a tedious task. It requires a designated person, which is a waste for human capital force, thus, how do we build an obstacle that is controlled electromechanically in order to change the height level instead of employing someone to change it.

1.6 Organization of Thesis

The thesis of this project consists of 5 chapters. Chapter 1 discuss the main idea of the project which is divided into 5 parts; background of the project, objectives, purpose of the study, scope and problem statement. Chapter 2 provides the specified information on smart jumping obstacle standards and literature review of some previous lifting mechanism,. It continues discussing the documentation of previous students' work. Chapter 3 is about the methodology of in the project. The system overall, mechanical structure, the components, the techniques and the process done in this project are all mentioned in details. Chapter 4 shows the results and outcomes of this project followed by the discussion based on the results of the project. Chapter 5 covers the conclusion for the project in general and some recommendations are provided for future development.

CHAPTER2

LITERATURE REVIEW

2.1 Introduction

A This chapter provides the specified information on smart jumping obstacle standards and literature review of some previous lifting mechanism, motor drivers, ultrasonic sensor use and plenty knowledge about what’s related to the smart jumping obstacle.

2.2 Smart Jumping Obstacle Standards

There are many types of obstacles and the most used ones are the followings:

- Oxer:
  It is a combination of two or three vertical obstacles with multiple rails, it comes with different styles, ascending, descending and so on, Figure 2.1a.

- Filler:

  It is basically a vertical obstacle but the difference is it’s filled with blocks of flowers, wall and so on, Figure 2.1b.

- Liverpool:

  It is an Oxer obstacle with a small water pool located at the bottom under the obstacle, Figure 2.1c.

a b c

Figure 2.1: Types of Horse Jumping Obstacles

Main components of an obstacles are, wing, cups and rail and here is the standards of dimension, material, weight, height and so on.

- The Wings:

  They are the two side vertical stands that hold the cups, they can be made of Wood, plastic or metal,2 meters maximum height and 0.6-0.9 meters wide, Figure 2.2a.

- The cups:

  They are the tools that carry rails where the rail must be able to roll freely, cups must be made from unbreakable material, plastic, steel or cast iron, besides they must have 18-30mØ depth, Figure 2.2b.

- The rail:

  It is the horizontal bar that usually made of pine, it must be in the range of 3.5-4 meters wide, 13-23 kilograms heavy and 8-10 centimeters of the diameter, Figure 2.2c.

a b c

Figure 2.2: Main Components of the Obstacle

2.3 Lifting Mechanism

There are several mechanisms that can be used to lift the cups, although they are not used for the horse jumping obstacles so far.

2.3.1 Scissors Mechanism

A scissors lifting uses mechanical means to extend or position platform. The term scissor refers to its mechanism where it uses linked, folding support in a crisscross 'X' pattern as shown in Figure 2.3. The working principle of this machine based on applying force such as hydraulic, mechanical and pneumatic means. These techniques are utilized to make extension by applying pressure to the set of supports that are located at one end of the mechanism and this cause elongating to the crisscross pattern.

Figure 2.3: Scissors Mechanism

2.3.2 Jack Device

Jack device is a mechanical device where it is used to lift heavy equipment or applying strong forces and this is done by using screw thread or hydraulic cylinder. The hydraulic jacks are power full more than the jacks that use screw thread. The hydraulic one is used to lift loads for greater distance. A way of developing the jack device is to add a motor to the driver pen and once the motor rotates the driver pen, the device lifts up, but it won’t be capable to be added to the jumping obstacle.

2.3.3 Sliding Mechanism

Sliding door is slide horizontally to be opened where the door is mounted on or hanged from a track. There are many types of door such as Arcadia doors, pocket doors and bypass doors. These types of doors are commonly used for shower door, glass door, wardrobe door and screen door. There are two types of standard systems, top hung and bottom rolling. Top hung system is commonly used; the door is lifted by two trolley hangers while the door running on a track. The weight of the door is taken by the hangers so the door can move easily. Bottom rolling system is used when the weight of the door can't be handled from above. This system uses two rollers at the bottom running of the door and two guides at the top running to guide the sliding movement of the door. The all weight of the door is concentrated on the two bottom wheels.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter explains the steps, methods, tools and components used to design a smart jumping obstacle. Figure 3.1 shows the flow of this project divided along two semesters. This chapter contains 3 main parts of semester one which are the project overview procedure, project design and components. The part of project overview procedure will contain a flowchart which will show the whole process that is done so far and will be done to accomplish the project easily. The project design is divided into two main sections, the mechanical design and electronic circuit design. A detailed explanation will be given on the structure of both where the mechanical design will show the hardware design while the electronic circuit design will focus on the type of device and components used in this project. Finally, a suitable programming language will be used to control the microcontroller which is considered to be the brain of the project.

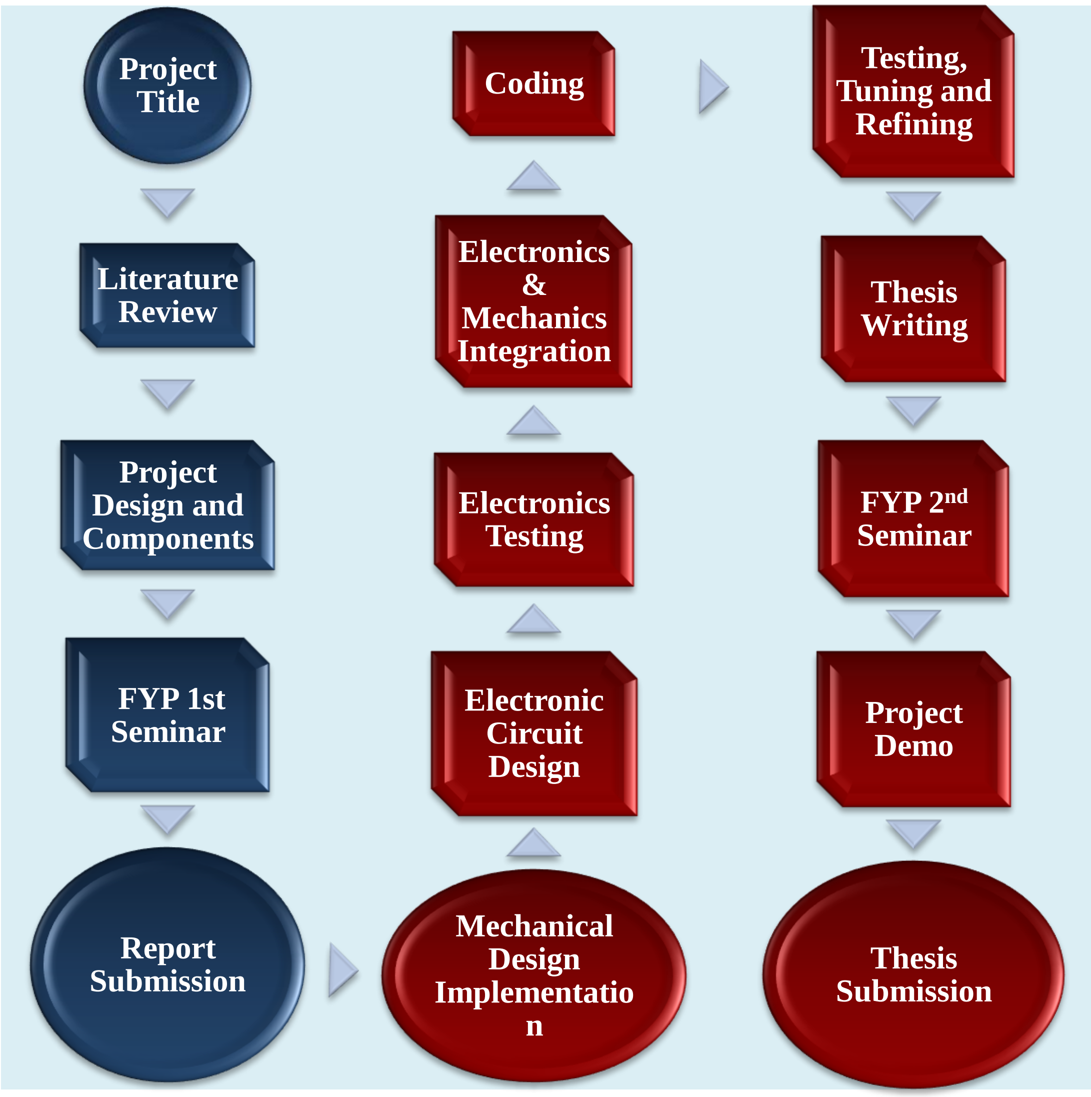

Figure 3.1: Project Flow

3.2 Project Overview

The project is divided into two parts: hardware and software design. The flowchart in Figure 3.2 shows the process for the whole project. The process starts by searching and finding a solution for the problem statement. That is done by reading and acquaint with previous projects and studies related to the problem statement which has been briefly explained in chapter 2. After gaining enough information, the process will be divided, as mentioned, into hardware and software design which will be discussed in this chapter. Using the information and knowledge gained from the literature review, the mechanical structure can be determined as well as the electronic circuit design. This includes the interfacing between input and output, we will start with the most important part which is the programming. It is important because it will determine whether or not the smart obstacle will function as desired.

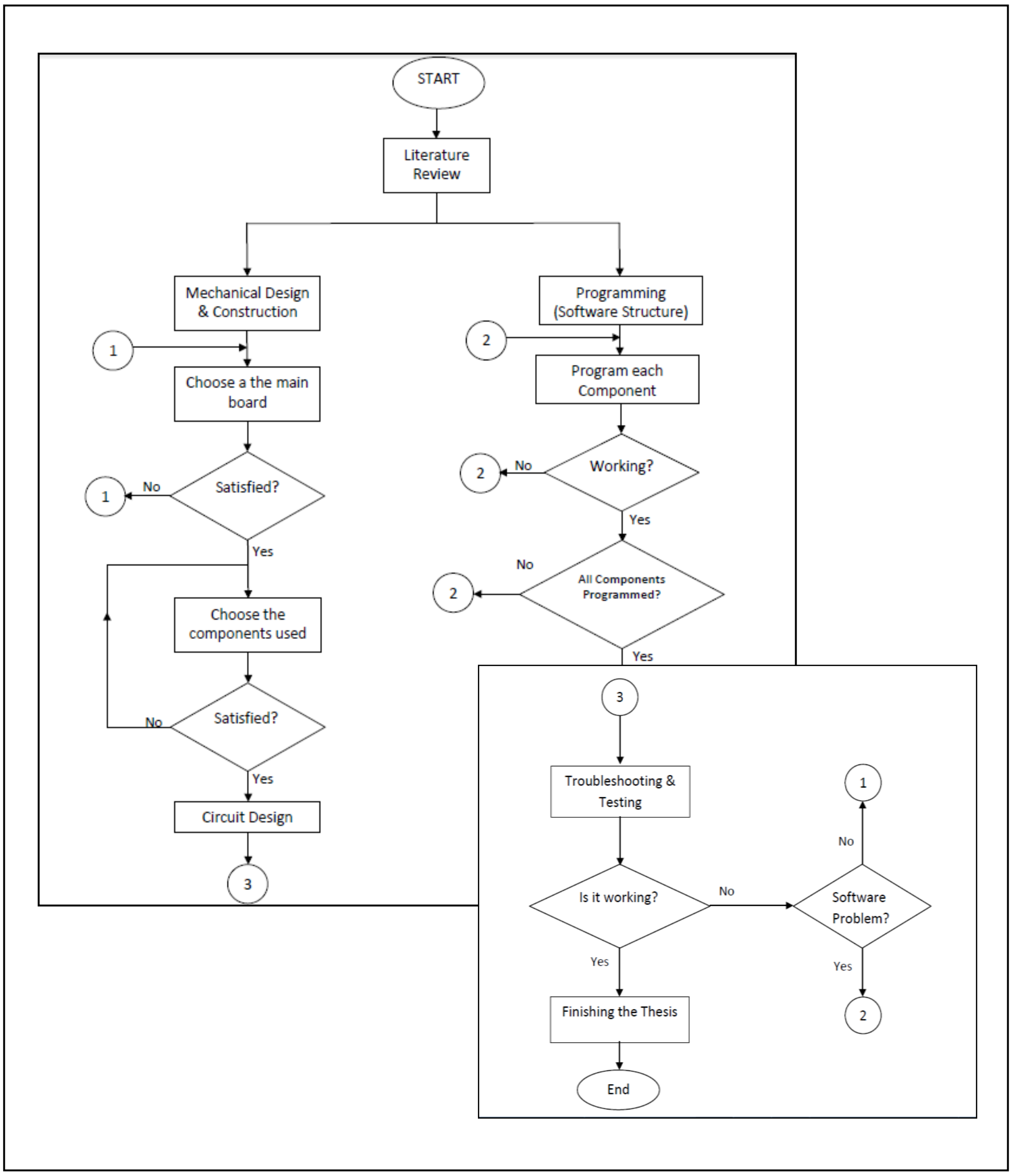

Figure 3.2: The Flowchart for the Whole Project

3.3 Mechanical Design

Project design is divided into two main sections, the mechanical design and electronic circuit design. The mechanical design illustrates how the mechanical parts are designed and where they are placed. The design went through 3 steps; manual sketching, sketching using SolidWorks 2013 and the actual hardware design. There were 4 main components for mechanical design which are the body structure, the motor, the cup and the rail. Figure 3.3 shows the whole project where the 2 wood wings are placed with a 1m distance between them. The two power window motors are placed at the bottom of the wood wings; one at each wood wing while the cup is the 3D printed used to carry the rail which is a horizontal wood bar.

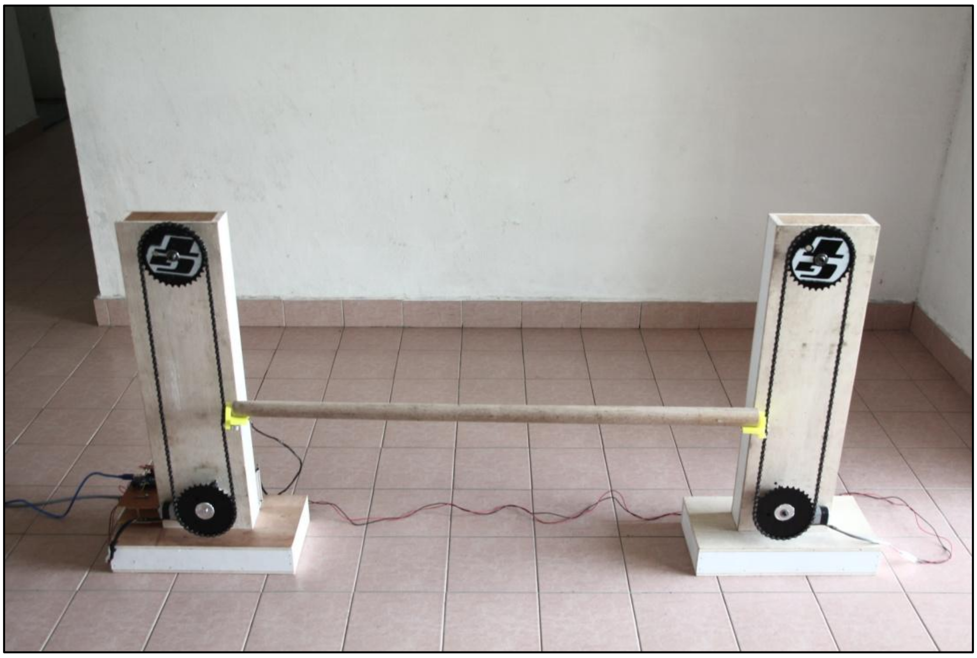

Figure 3.3: Mechanical Design of The Whole Project

3.3.1 Body Structure

The main body is a two wood wings. Each wood wing has a base with the dimensions 40cm x 20cm x 8m. While the main part of the wing is a cuboid with the dimensions 20cm x 8cm x 70m. Figure 3.4 shows the basic structure of the wing in

Solidworks 2013. The main electronic circuit is placed on the base as well as the motors. However, the main electronic circuit is placed only on one of the wing to reduce the cost while the motor is placed on the back of both wings. As seen in Figure 3.3, a gear is attached closely to the motor in order to move the chain which the cup is solidly to the chain. At the top of the front back, the supportive and informative electronic circuit is placed where the LCD and two buttons are placed to monitor and control the movement of the chain which holds the cups. However, in general, the basic role of the wings is to hold the cups.

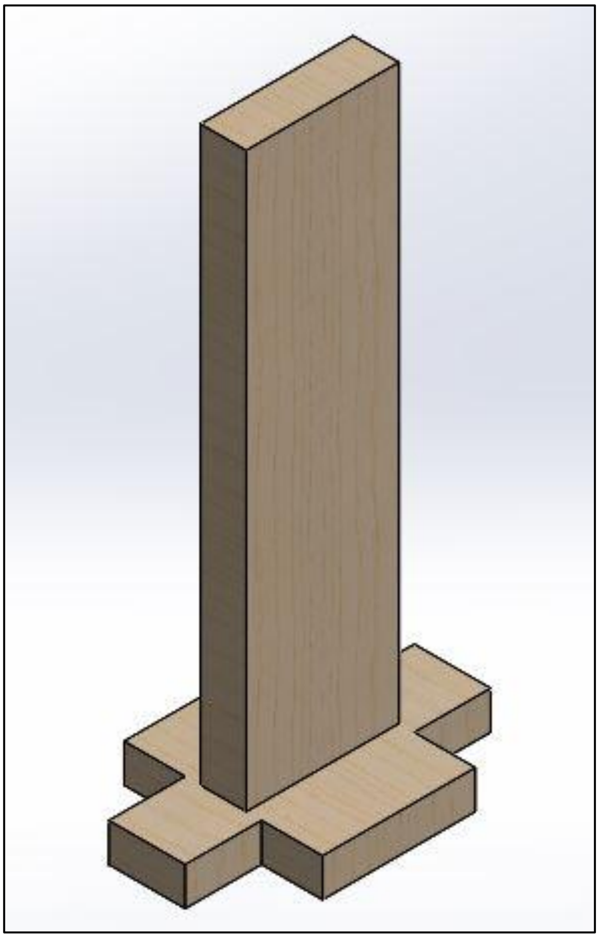

Figure 3.4: Basic View of the Wing

3.3. 2 Power Window Motors

Power window motors are special types of DC motors. Therefore they have the same concept as any DC motor. In general, DC Motor has two wires which are known as positive and negative terminal to distinguish between them. The motor rotates clockwise when the positive wire of the motor is connected to the positive terminal of the power supply and the negative wire is connected to the negative terminal of the power supply. The motor will rotate anti clockwise if the wire polarity is reversed. Figure 3.5 shows one example of DC motor when the polarity is positive the rotation will clockwise and vice versa.

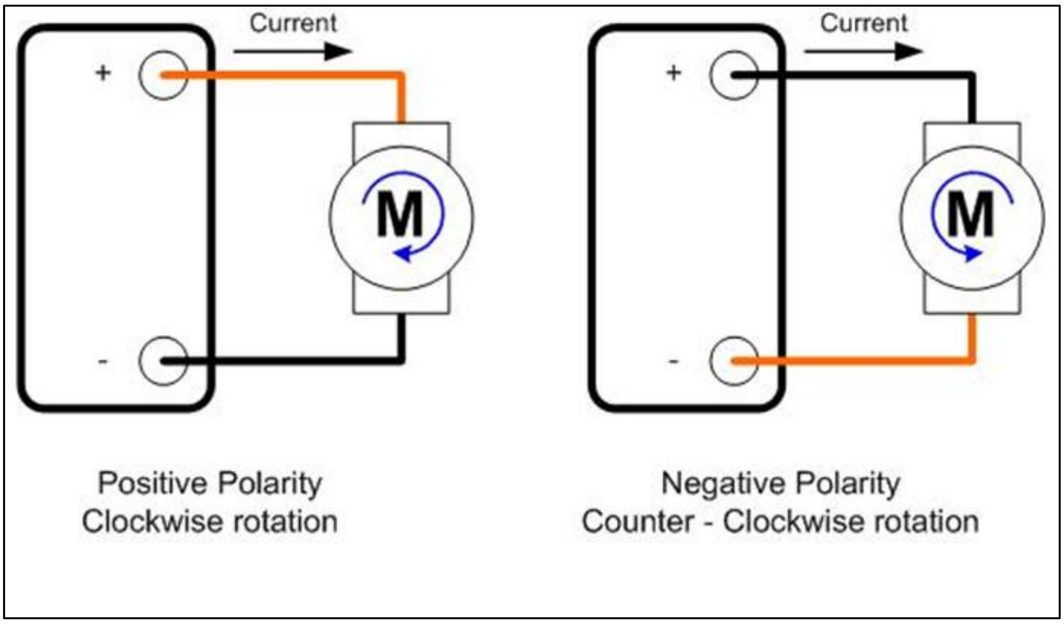

Figure 3.5: Basic DC motor

DC motor consists of two main parts: a stator and a rotor (armature). The armature rotates due to the electromagnetic interaction between the stator and rotor. The rotation speed is proportional to the voltage applied to the armature. The speed of the motor can be controlled by regulating the amount of voltage across the motor terminals. One popular way to achieve voltage regulation is Pulse Width Modulation (PWM).

The idea behind Pulse Width Modulation speed control is driving the DC motor using a series of ON and OFF pulses of voltage level and varying the duty cycle while keeping the frequency constant. Figure 3.6 illustrated a pulse width modulated waveforms. When the duty cycle is small the average voltage is low, therefore, the motor speed will be small. When the duty cycle is big the average voltage will be high, therefore, the motor speed will be high.

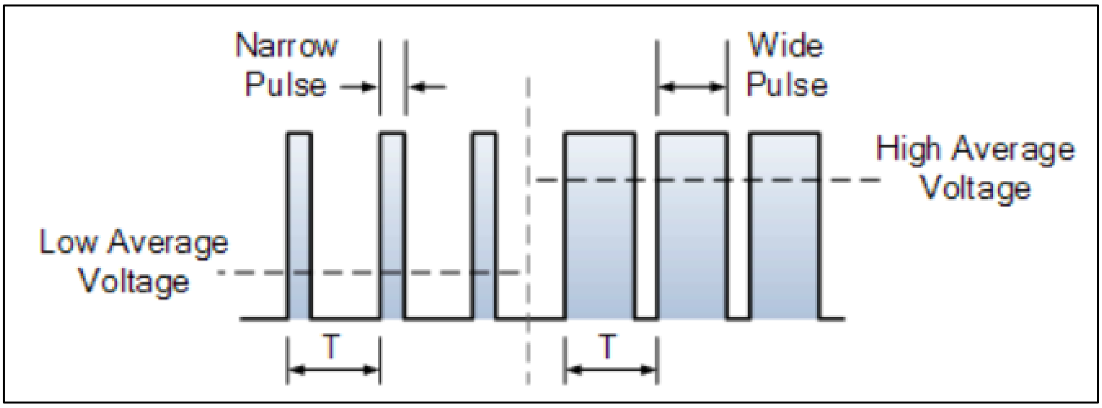

Figure 3.6: Pulse Width Modulation

Power window motors are also known as automobile motors which can be controlled by few switches. These switches can raise and lower the motor's speed and determine its direction by being pressed. They are mainly used in the cars inside the car door to control and allow the window to roll up and down. The same idea is used here to move the cups - which holds the rail – up and down. In this project, two power window motors were used; one right hand side and the other was a left hand side. The motor is placed at the bottom of each wing. Figure 3.7 shows a left hand side power window motors. The specification of the power window motor is as follows:

- Voltage Rating: 12VDC
- Speed (No Load): ~ 85RPM
- Current (No Load): <3A
- Current (Load): <7A
- Current (Lock): <20A (Stall Current)
- Torque: 30Kg.cm

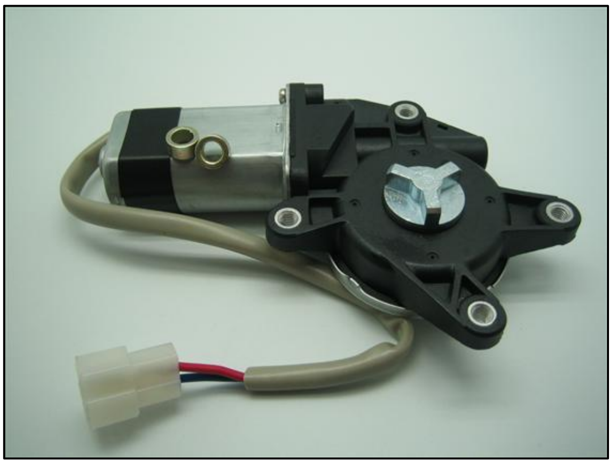

Figure 3.7: Power Window Motor

Two bicycle's gears were used for each wing. One gears is connected directly to the motor while the two gears are connected through a chain. As seen from Figure 3.7, the motor has a three phase shaft connector but the gear hole is a circle as seen in Figure 3.8. Figure 3.8 shows a sample of the gear, however, it is not the same as the used one.

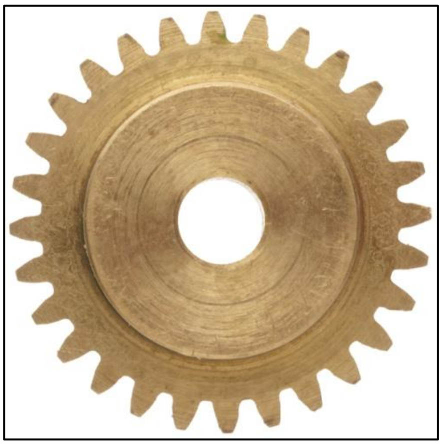

Figure 3.8: A Sample Bike Gear

Since the gear hole and the motor shaft did not match, a 3D printed item was needed to connected between them. The back side is attached to the motor so this side will have a triangle design while the front side will be attached to the gear so the side will have a cylindrical outcrop with a 24mm radius and a small hole for the screw with a 5mm radius. Figure 3.9 shows both sides of the 3D printed item where the left picture shows the inner (back) side and the right shows the outer (front) side.

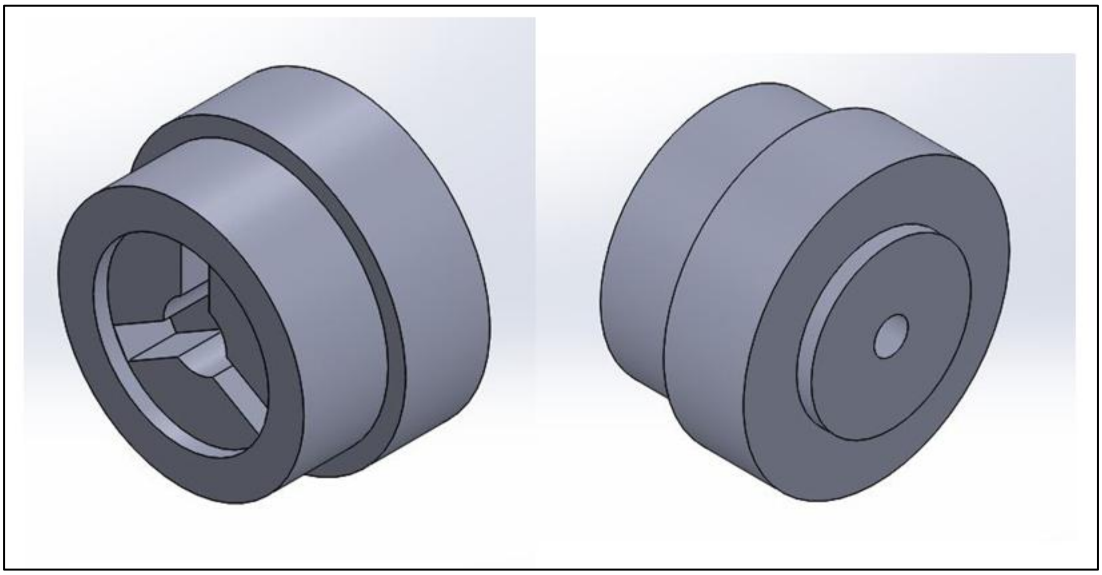

Figure 3.9: 3D printed Motor Shaft Connector

3.3.3 3D Printed Cup

As mentioned, the 3D printed cup is used to hold the rail which is a 1m cylinder wood. The 3D printed cup was printed in order to be able to allow the rail to roll freely and it is attached solidly to the chain on each wing. Figure 3.10 shows both sides of the 3D printed cup where the left picture shows the back side attached to the chain and the right shows the front side which holds the rail. The three at the back side are used for the screws to tightly hold the cup with the chain so the cup will not move right and left. The ultrasonic will be attached to the bottom of the cup to determine the distance between the base and the cup. The distance is known as the height.

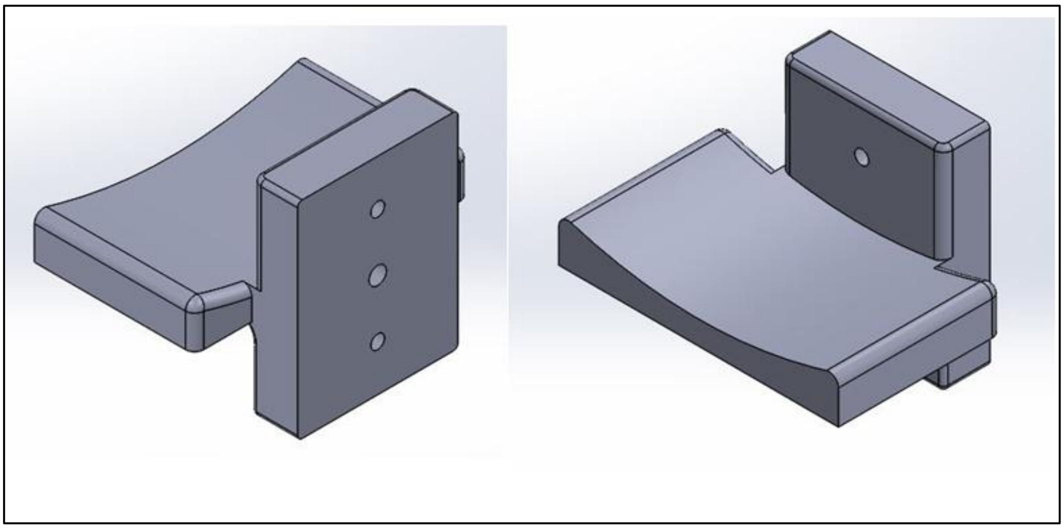

Figure 3.10: 3D Printed Cup

3.4 Electronic Circuit Design

Electronic circuit design describes the electronic components of the project. It talks about the different components used in the project and their specifications. The main components used are Arduino UNO, ultrasonic sensor, LCD and a motor driver module as shown in Figure 3.11. Understanding the specifications of each component and device is crucial because the components cannot do the tasks individually. Therefore, their capabilities and limitation should be taken into account to avoid any failure and to determine the right power supply to run the project without causing any problems.

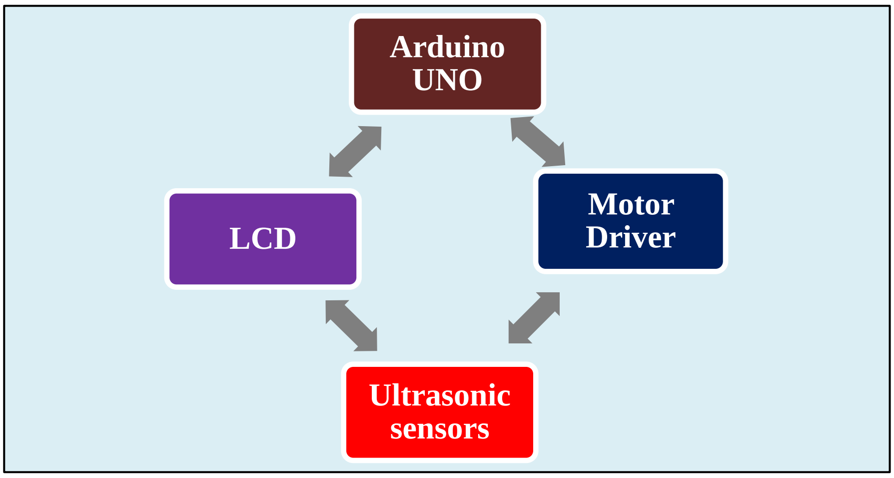

Figure 3.11: The Main Electronics Components

The process of the project starts when the ultrasonic sensor determine the height or distance between the base and the cup. If the distance is not the same as the desired height, the Arduino UNO will send to the motor driver to drive the motor which, in turn, will rotate the chain holding the cup. Therefore, the cup go up or down until it reaches the desired height. If the distance is the same as the desired height, the Arduino UNO will send to the motor driver to stop the motor. The LCD will always show the height and the commands, if any.

3.4.1 Arduino UNO

Microcontroller is the brain for the project. Therefore, it should be chosen carefully since it will affect significantly the performance of the whole project. In this project, Arduino Uno R3 is used to be the microcontroller. It will be responsible for controlling and giving commands to the other components. It gives the current height given from the ultrasonic sensor and compare it with the desired height and then give orders to the motor driver whether to move the cup up or down or not to move at all. It will also keep changed the LCD according to the given command. However, Arduino Uno R3 is a microcontroller board based on the ATmega328. It is the latest version of the Arduino UNO board. It uses ATmega8U2 instead of using the FTDI chip. This is done to apply a faster transfer rate. Figure 3.12 shows the Arduino Uno R3 used in this project.

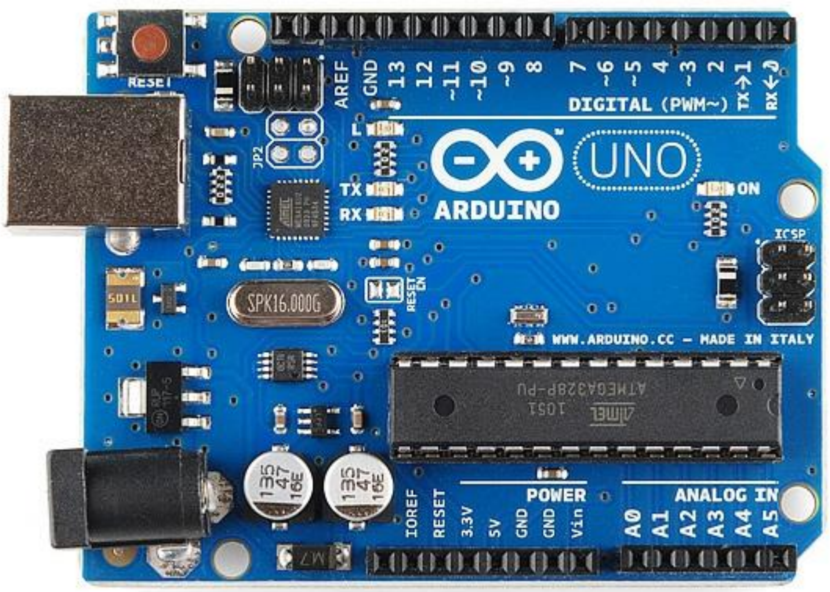

Figure 3.12: Arduino Uno R3 Board

As seen in Figure 3.12, Arduino Uno R3 has 14 digital input/output and 6 analog inputs as well as a USB connection, a reset button, and a power jack. Figure 3.13 show better picture about the pins in Arduino UNO. Arduino Uno is preferred due to its advantages. One of these advantages is its simplicity of being powered from the PC by using its USB cable. However, this is not the only way to power the Arduino, it can be powered by using an external battery or using a AC-to-DC adapter after stepping down its voltage in order to avoid damaging the Arduino. Table 3.1 shows summary about Arduino UNO R3 specifications.

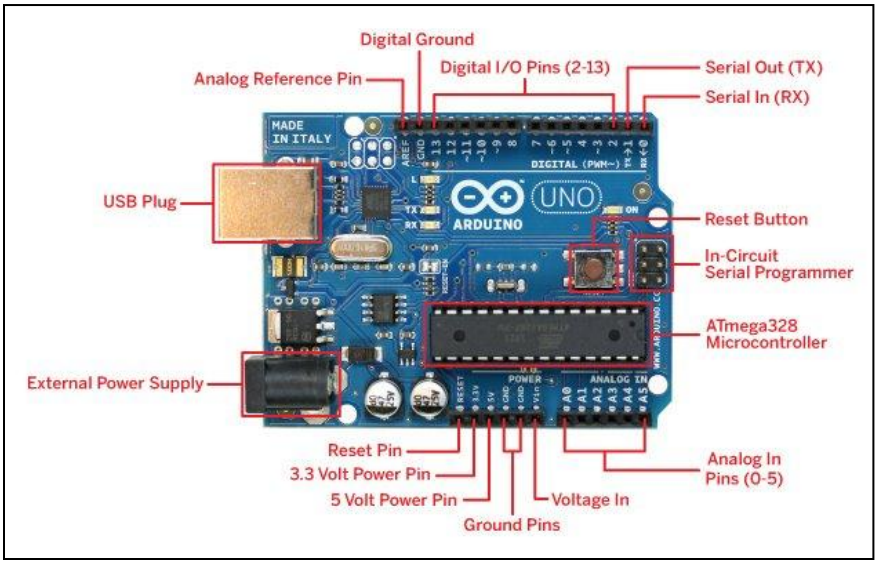

Figure 3.13: Arduino UNO Pins Reference

Table 3.1: Arduino Uno R3 Specifications

Microcontroller	ATmega 328
Operating Voltage	5V
Input Voltage (Recommended)	7-12V
Input Voltage (Limits)	6-20V
Digital I/O Pins	14 (6 provides PWM output)
Analog Input Pins	6
DC Current per I/O Pins	40mA
DC Current for 3.3V Pin	50mA
Flash Memory	32KB (ATmega 328) ; O.5KB used by bootloader
SRAM	2KB (ATmega 328)
EEPROM	1KB (ATmega 328)
Clock Speed	16MHz

Another reason why Arduino is preferred and widely used is because it is an open source which can be programmed easily. Its software is download easily and for free and also contains many open source libraries which ease the programming for the Arduino users. Its software is shown in Figure 3.14.

Figure 3.14: Arduino Software Interface

3.4.2 Ultrasonic Sensors

Ultrasonic sensor is a transceivers which contains both transmitter and receiver. It uses the same principle as the radar where the echo of the wave determines the attributes of a target. Ultrasonic sensor generate and send high frequency waves and then receives the reflected echo of the wave after hitting the base of the wing. The distance is determined by measuring the interval time between the transmission and receiving process. The type of ultrasonic sensor used in the project is HC-SR04 which is shown in Figure 3.15 where the upper picture shows the front side and the below picture shows the back side. The HC-SR04 ultrasonic sensor uses sonar to determine distance to an object like bats or dolphins do. It offers excellent non-contact range detection with high accuracy and stable readings in an easy-to-use package. From 2cm to 400 cm or 1” to 13 feet. It operation is not affected by sunlight or black material like Sharp rangefinders are (although acoustically soft materials like cloth can be difficult to detect). It comes complete with ultrasonic transmitter and receiver module. The feature/specification of the HC-SR04 ultrasonic sensor is as follows:

- Power Supply :+5V DC
- Quiescent Current : <2mA
- Working Currnt: 15mA
- Effectual Angle: <15°
- Ranging Distance : 2cm – 400 cm/1" - 13ft
- Resolution : 0.3 cm
- Measuring Angle: 30 degree
- Trigger Input Pulse width: 10uS
- Dimension: 45mm x 20mm x 15mm

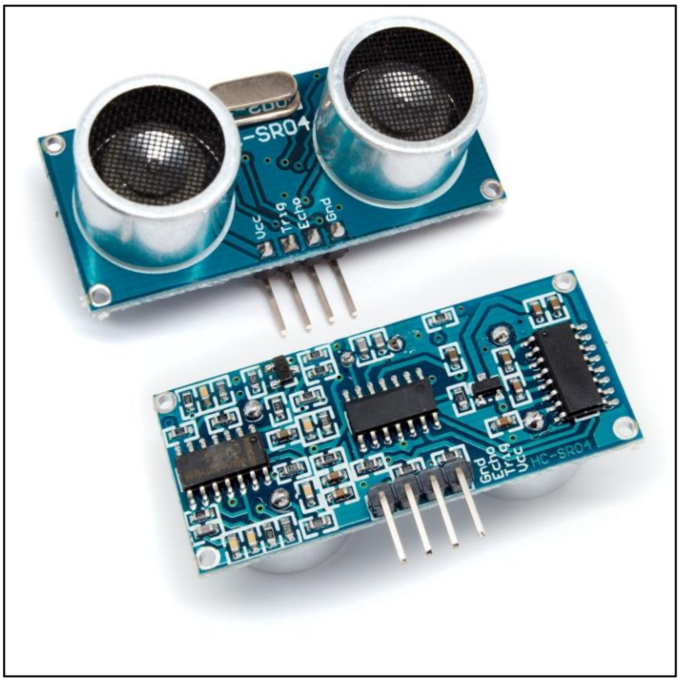

Figure 3.15: HC-SR04 Ultrasonic Sensor

Figure 3.16 shows the connections between the Ardunio UNO and the ultrasonic sensor. It can be seen that only two pins are needed, despite of the V_{CC} and the Gnd. These two pins are the Trig and Echo pins.

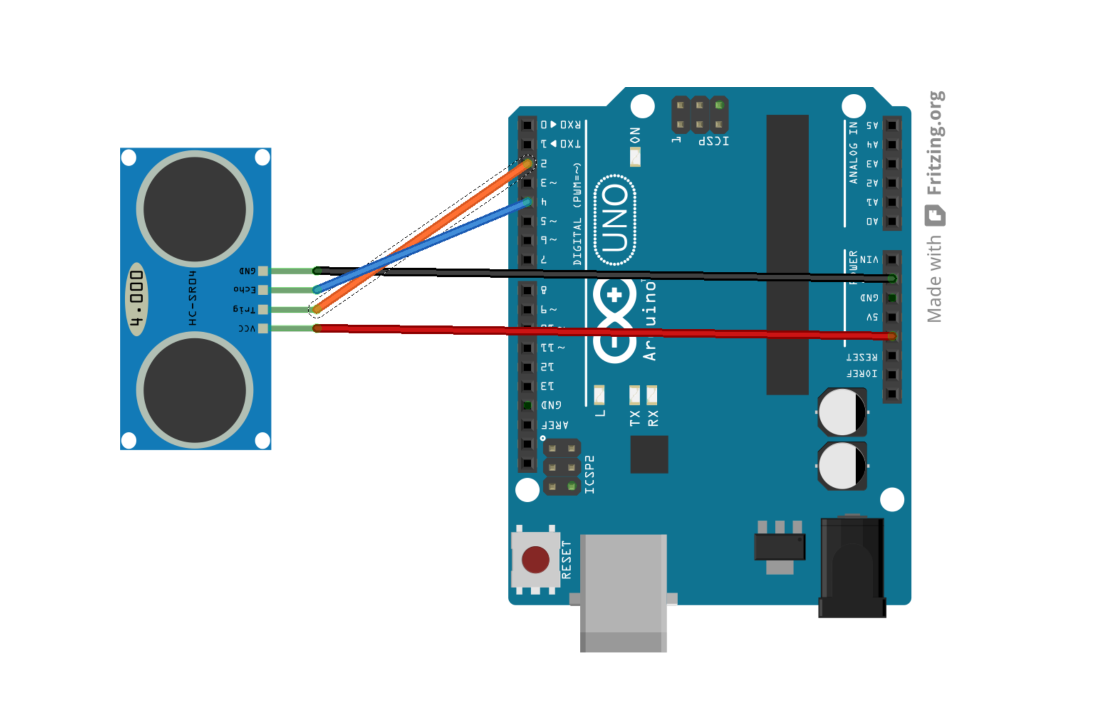

Figure 3.16: Connection Between Ultrasonic Sensor and Arduino UNO

3.4.3 Motor Driver Module

Arduino cannot control the motor directly because the motor requires high current while the Arduino produce a very low current. From the motor specification, it can be seen that one motor requires high current to rotate while Arduino only provides 40mA for each I/O pin. Therefore, if the motors are connected directly to the Arduino, Arduino will be damaged and may break down. For that, a motor driver is required as a connecter between Arduino and DC motors. Motor driver receives the orders from the Arduino and controls the motor based on the orders. The advantages of the motor driver is that it can be connected to an external power supply which will drive the motor with the sufficient and suitable power. In this project, the external power supply has a voltage of 12V and a current of 4.2A.

The type of motor driver used in the project is MDD10A which is shown in Figure 3.17. MDD10A is the dual channel version of MD10C which is designed to drive 2 brushed DC motors with high current up to 10A continuously. Just like MD10C, the MDD10A also supports locked-antiphase and sign-magnitude PWM signal. MDD10A has been designed with the capabilities and features of:

- Bi-directional control for 2 brushed DC motors.
- Support motor voltage ranges from 5V to 25V.
- Maximum current up to 10A continuous and 30A peak (10 second) for each channel.
- Solid state components provide faster response time and eliminate the wear and tear of mechanical relay.
- Fully NMOS H-Bridge for better efficiency and no heat sink is required.
- Speed control PWM frequency up to 20KHz.
- Support both locked-antiphase and sign-magnitude PWM operation.
- Onboard push button to control the motor manually.

Figure 3.17: MDD10A Motor Driver

3.4.4 LCD Module

LCD which is short for Liquid-Crystal Display is a flat panel display used to display a certain message to the end user. The message could contain information about the device, instruction on how to use the device, or any type of message the programmer wants to inform the end users. LCDs are used widely in many applications with different size and capacity. This is because of its low power dissipation, flexibility for both programmer and end user, and high readability. Figure 3.18 shows the LCD used in the project which is a 16x2 characters LCD display.

Figure 3.18: 16x2 Characters LCD Module

Figure 3.19 shows the connection between Arduino and LCD. It can be seen that 8 pins are used to shows the message in the LCD. The most right two pins on the LCD is used to light the LCD screen. The third pin on the left is connected to the potentiometer which is a variable resistor. Every pin of the LCD pins should be connected correctly in order to display the message. If one pin is loose, the message will not appear.

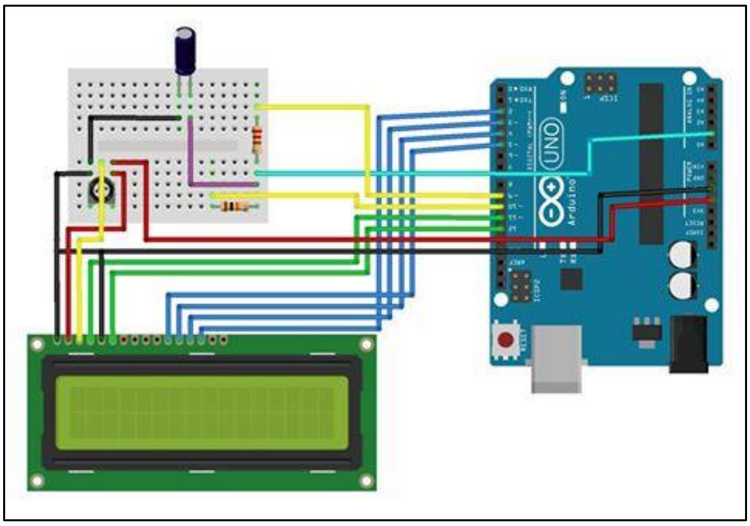

Figure 3.19: Connection Between LCD and Arduino UNO

Figure 3.20 shows the electronic circuit design for the project where the main devices are connected together. It can be seen that Arduino UNO is connected to the other devices to control and coordinate the project. The Ultrasonic sensor is placed at the left of the LCD while the motor driver is on the right. This is just an abstract of their connection where, in reality, there are put in different places. In the project, the motor driver and Arduino UNO are placed on the base of the wing where Arduino UNO is placed on a layer above the motor driver. On the other hand, LCD was placed at the top of the wing while ultrasonic sensor is placed under the 3D cup.

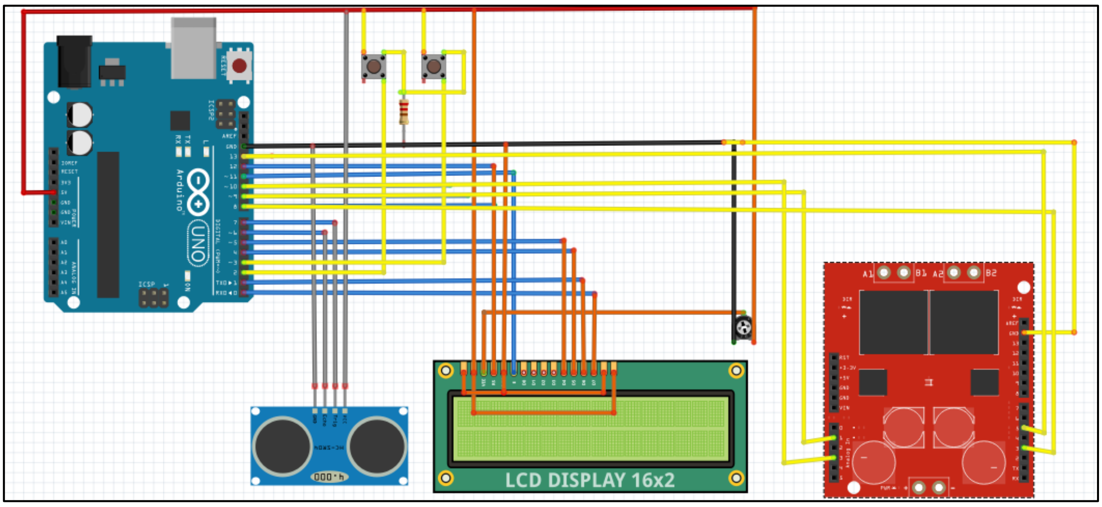

Figure 3.20: Electronic Circuit Design

3.5 Software Design

Programming is considered as the most essential and critical part of the project. The electronics components and devices are first programmed and tested separately. Then, they integrated and their programming is modified and tested until the project works as desired. Arduino UNO will be programmed using its own software to receive the data (height) from the ultrasonic sensor and then send the orders to the motor drive. Before programming any project, a flowchart of the whole project must be done. Designing the flowchart gives a better and clear view to track the flow of the programming. Figure 3.21 shows the flowchart of the project.

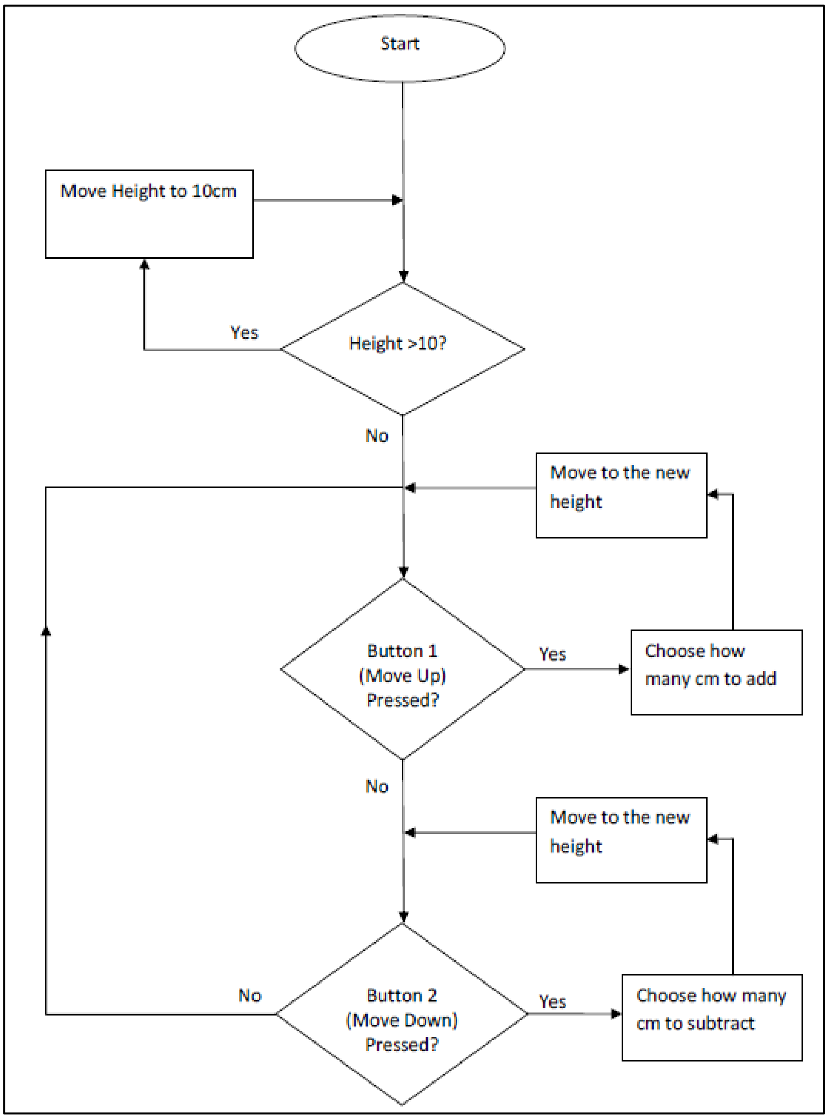

Figure 3.21: Flowchart of the Whole Project

At the beginning of the programming, the height of the rail should be reset and settling at the minimum height which is 10cm. Therefore, it can be seen from Figure 3.21 that the programming starts by receiving the current height of the rail from the ultrasonic sensor. If the current height is higher than 10cm, Arduino will send orders to the motor driver to control the motors and lower the rail until it reaches 10cm. After that, it will check if one of the button orders has been pressed where button 1 is the order of increasing the height and button 2 is the order of decreasing it. If any of the buttons is pressed, Arduino will send the corresponding order to the motor driver and the motor will move as desired. Appendix A shows the whole code used to in this project.

3.6 Summary of the Chapter

This chapter was all about the hardware and the software components used in the project. The mechanical part consists of two wings and a rail which connects between and hold on a cup at each end. The cups are attached firmly on the chain which is attached on two gears controlled by the motor. The motors on both wing moves simultaneously. While the electronics part consists of four modules. Arduino UNO is used as microcontrollers and connects the whole system together. Ultrasonic sensor always read and determines the current heights. Motor driver module connects between the power window motor and Arduino UNO. It receives the commands from Arduino and control the motor based on those commands. In order to achieve the objective, the hardware and software components should be carefully designed, integrated and tested.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This chapter discusses the outcomes and results of the project including the final design, the function of ultrasonic sensors and final result.

4.2 Final Design

The whole project was finally built after all the electronic devices are connected together and are integrated with the mechanical components. Figure 4.1 and Figure 4.2 shows the front and back sides of the main wing, respectively. The main wing is the wing which the electronics circuits are placed on it. The front view shows the LCD with the two buttons on the top while the Arduino UNO, motor driver and power supply is placed on the base of the wing. The back view shows the mechanical part such as the gears, chain, cup and the power window motor as well as the ultrasonic sensors which is placed below the cup. The secondary wing has the same mechanical components as the main wing, as shown in Figure 4.3, but does not contain any electronics components. Its motor is programming to make the same rotation as the motor of the main wing.

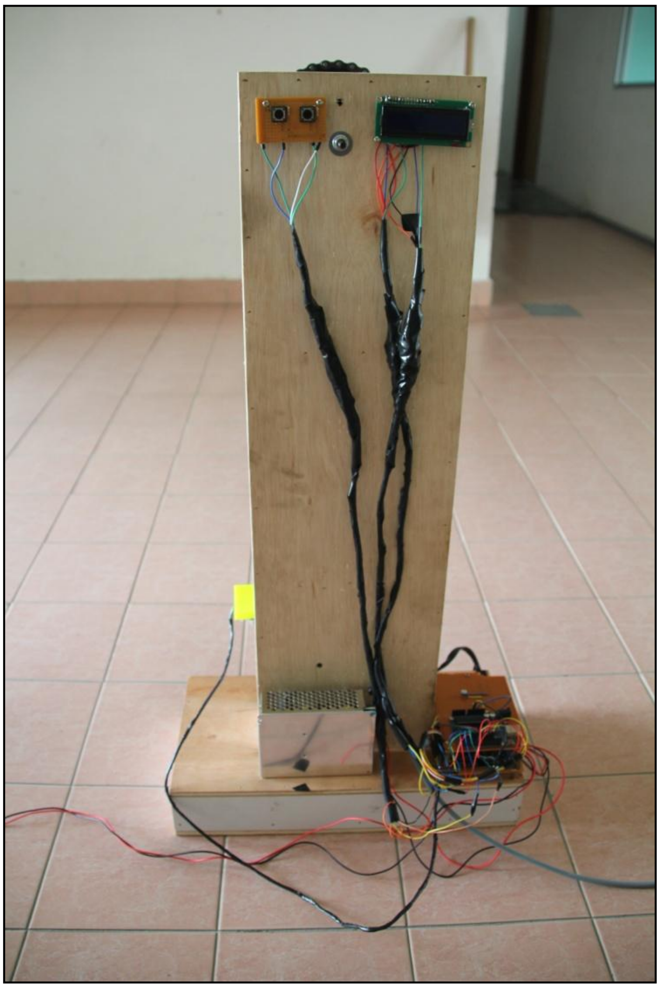

Figure 4.1: Front View of the Main Wing

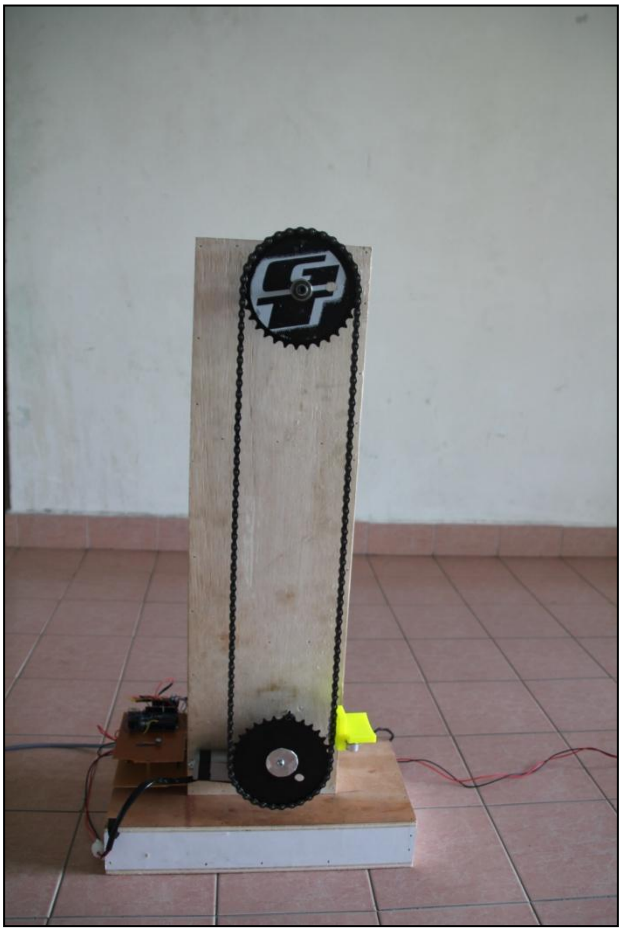

Figure 4.2: Back View of the Main Wing

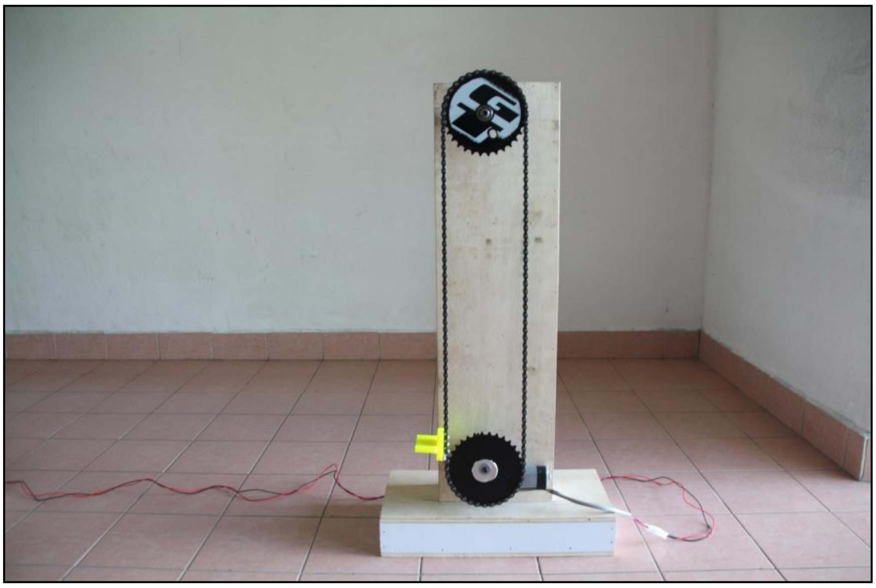

Figure 4.3: Back View of the Secondary Wing

4.3 Ultrasonic Sensor

It can be seen from Figure 4.4 that the ultrasonic sensor is placed under the 3D printed cup of the main wing. In this project, ultrasonic sensor is used to determine the range or distance between the rail and the base of the wing. As mentioned in 3.4.2, this done by measuring the interval time between the transmission and receiving process. Ultrasonic sensor has two important pins which are Trig pin and Echo pin. Trig pin is used to transmit the wave while Echo pin is used to receive the reflected wave. The current height of the rail will always be shown on the screen of the LCD. LCD shows two lines for the height, as shown in Figure 4.5. The first line is the approximated height which is written in Arduino UNO while the actual height is mentioned in the second line. The actual height is determined from the ultrasonic sensors.

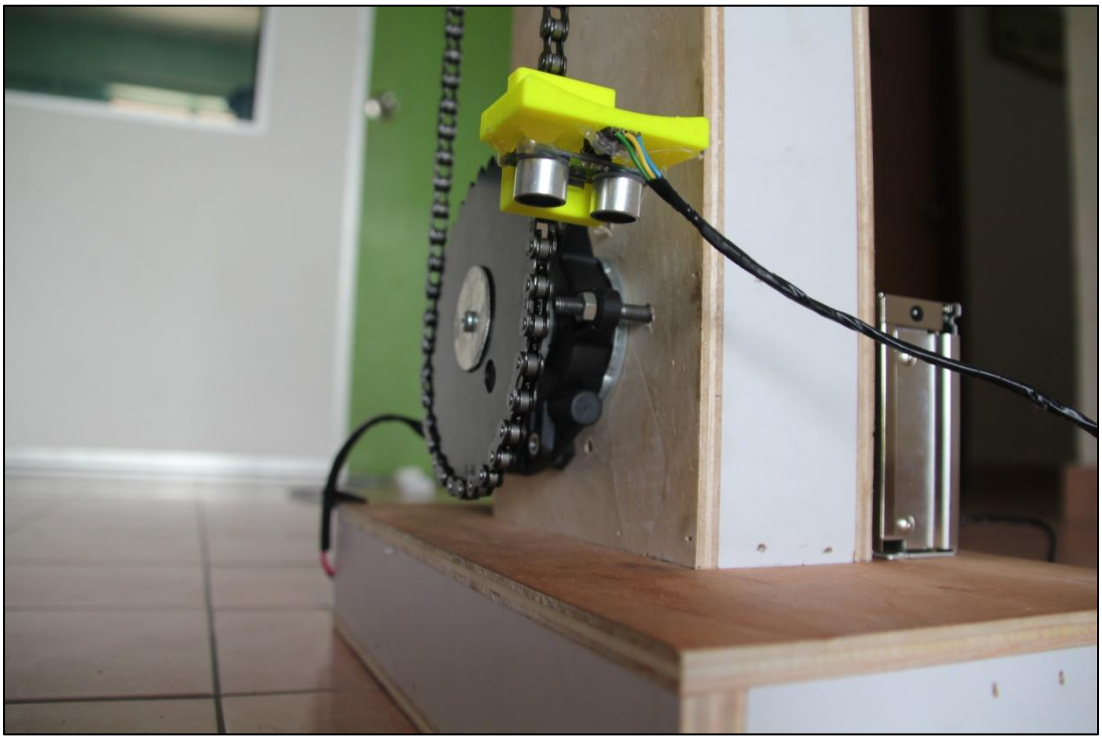

Figure 4.4: Ultrasonic Sensor Location

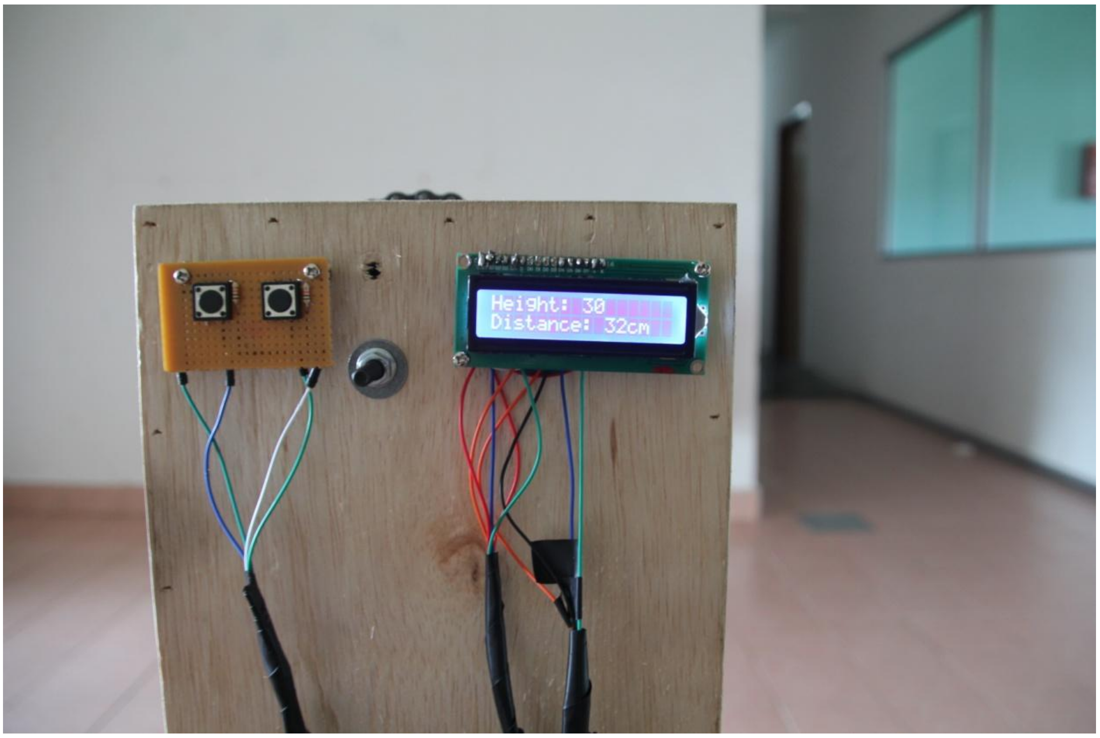

Figure 4.5: Height Shown on LCD

4.4 Final Result

After the whole project is completely built, the code, shown in appendix A, is tested and verified. If button 1 is pressed first, each hit on button 1 afterward will increase the height by 5cm. Then, no action will be done until button 2 is pressed where a new message will appear on LCD showing the new height (desired height), as in Figure 4.6a. Then, the motor will move the chain which holds the cup causing the rail to move up to the desired height. When the desired height is reached, Arduino will send a stopping order to the motor driver which will stop the motor, as shown is Figure 4.6b. However, if button 2 is pressed first, each hit on button 1 afterward will decrease the height by 5cm. Then, no action will be done until button 1 is pressed where a new message will appear on LCD showing the new height. By doing this, the horse rider is not required to hire an assistants or to get off his horse to change the rail's height.

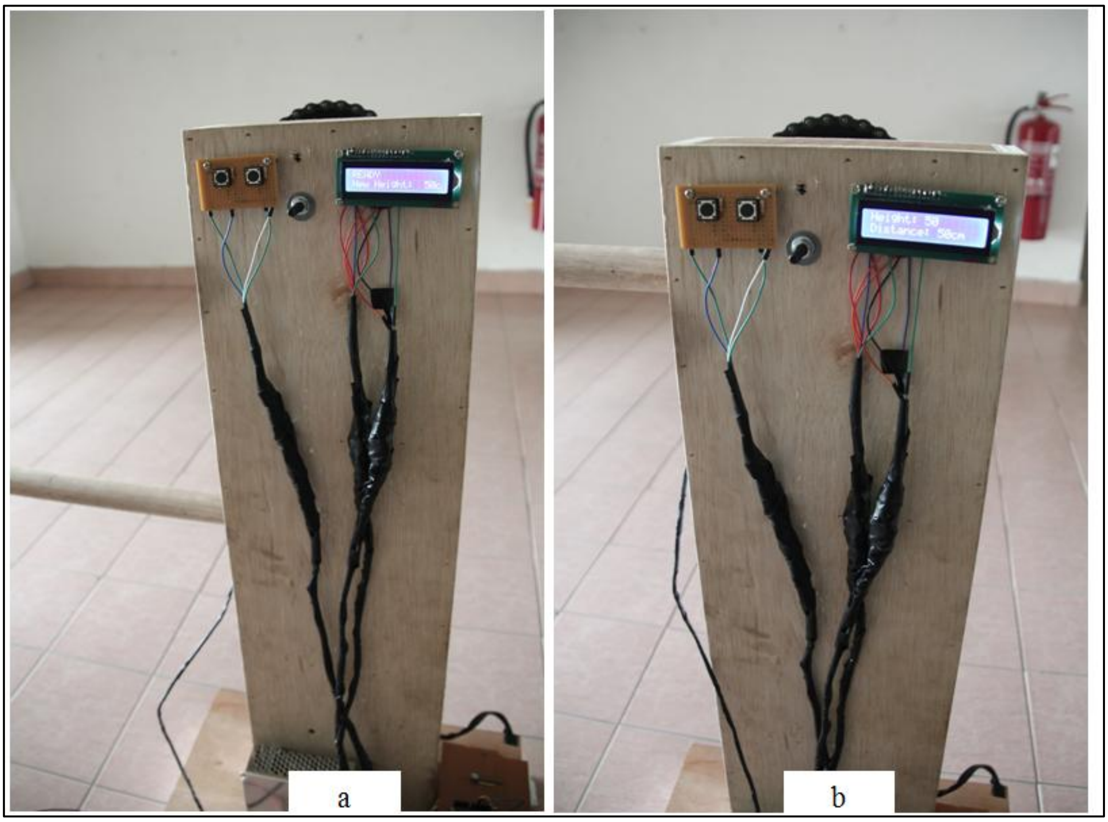

Figure 4.6: Movement Shown on LCD

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

The objectives in this project were completely achieved. First objective is fulfilled as shown in the picture of Chapter 4. The smart jumping obstacle involves both electrical and mechanical parts to assist the horse rider. The second objective is fulfilled by using the coding in Appendix A. The coding enables the motor to change the height of the rail according to the horse rider desire. However, this requires a direct connection between the horse rider and the buttons. Despite the fact that the objectives were successfully done, several challenges were faced during the process. The majority of the challenges were about the connecting between the mechanical parts where some parts did not match together. Therefore, some part were replaced with a suitable one and some 3D printed items were used as connection. Another problem that faced the project was the lack in accuracy of the ultrasonic sensor. There was a small difference between the actual height and the height in the microcontroller.

5.2 Recommendations

Since electromechanical does not involved in the equestrian obstacle so far, many improvements can be done to improve the functionality and efficiency of the project. More components and device can be used to improve the project such as the followings:

5.2.1 Vibration Sensor

A sensor should be placed on the rail to determine if the horse has hit the rail or not. Sometimes the horse does hit the rail but the rail does not fall down specially if it was a slightly hit. This is essential to train the horse to be able to jump higher and avoid hitting the rail by any chance. This is done by using vibration sensors where it can be change the physical touch between the sensor and the horse into electrical energy. One of the most used and famous vibration sensors is Piezoelectric sensor. An LED board or a long RGB LED strip could be connected to the Piezoelectric sensor to show if the horse hit the rail or not. When the horse touches the rail - even slightly-, the LED board or a long RGB LED strip would light.

5.2.2 Wireless Communication

Instead of using a physical movement to change the height of the rail, using any wireless communication such as Bluetooth HC-05will improve the function of the project. The Bluetooth module will be connect between the riders phone and Arduino UNO where the rider will be able to change the height from a certain distance without the need to get near the obstacle.

CHAPTER 6

PROJECT MANAGEMENT

6.1 Introduction

An effective and successful plan is needed to fulfill the objectives within a particular period. However, there are several factors that restricted the project such as budget, scope, given time and the availability of the resources. One of the effective methods used in time management is Gantt chart. Gantt chart is the guideline of the project and provide the ability of tracking the flow of the process. However, the cost of the project should be estimated to determine the efficiency of the project in terms of monetary; the minimum cost is required.

6.2 Project Schedule

Table 6.1 shows the Gantt chart for the first semester while Table 6.2 shows the Gantt chart for the second semester. Some delays happened in the second semester at the mechanical and electronics circuit design. This is due to the late delivery of some of the electronics components and the problem of searching for a suitable gear. Though a delay occurred, the project was successfully completed on time.

Table 6.1: Gantt Chart (Semester One)

Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
FYP Briefings		■														
SV Assignment			■	■												
1st meeting with SV					■											
Deciding a project						■										
Literature Review							■	■	■	■	■	■				
Proposal Submission							■									
Hardware Design											■	■	■			
FYP Presentation														■	■	
FYP - report																■

Table 6.2: Gantt Chart (Semester Two)

Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mechanical Implementation	■	■	■	■												
Electronic Circuit Design			■	■	■	■	■									
Electronics Testing						■	■	■								
Electronics &Mechanics Integration								■	■	■	■					
Coding										■	■					
Testing, Tuning and Refining											■	■	■			
FYP 2nd Seminar														■	■	
Project Demo															■	
Thesis Writing																■

6.3 Cost Estimation

The cost of the project is dividing into two parts; cost for mechanical design and cost for electrical components. Table 6.3 shows the estimated cost for mechanical parts while table 6.4 will illustrate the cost for the whole electronics design. It can be seen that the most expensive thing was the wood wing. However, the total estimated cost for the whole project is shown in table 6.5.

Table 6.3: Cost of the Mechanical Parts

Item	Cost per unit	unit	Price
Wood Wings	RM 110	2	RM 220
DC Motors	RM 32	2	RM 64
Gears With	RM 15	4	RM 60
Chain	RM 10	3	RM 30
Screws	RM 5		RM 5
1 Meter Wood Bar	RM 10	1	RM 10
Pieces 3d Printed	Free	4	Free
Subtotal			RM 389

As seen in table 6.3, the most expensive part in the mechanical design was the wood wing which the pair wings cost RM 200. Their cost was more than half of the total mechanical design. Table 6.4 shows the cost of the electronics components. It can be seen that the Arduino UNO is the most expensive of the electronics components with a RM 2 more than motor shield. The cost of Arduino UNO was around RM75. Lastly, table 6.5 shows the total cost used for the project by adding the cost of the mechanical parts and electronics components together.

Table 6.4: Cost of the Electronics Components

Item	Cost per unit	unit	Price
Arduino UNO	RM 75	1	RM 75
Ultrasonic Sensor	RM 40	1	RM 40
Motor Driver Module	RM 73	1	RM 73
16x2 LCD Screen	RM 15	1	RM 15
External Power Supply	RM 66	1	RM 66
Subtotal			RM 269

Table 6.5: Total Cost for the Whole Project

	Subtotal
Mechanical Part	RM 389
Electronics Components	RM 269
Total	RM 658

REFERENCES

[1] Carullo, A.; Parvis, M., "An ultrasonic sensor for distance measurement in automotive applications," *Sensors Journal*, IEEE , vol.1, no.2, pp.143, August 2001

[2] Patterson, M.; Doyle, J.; Cahill, E.; Caulfield, B.; Persson, U.M., "Quantifying show jumping horse rider expertise using IMUs," *Engineering in Medicine and Biology Society* (EMBC), 2010 *Annual International Conference of the IEEE* , vol., no., pp.684,687, Aug. 31 2010-Sept. 4 2010

[3] Club Australia, P. July (2008). *Show jumping Manual*.

[4] Chande, P.K.; Sharma, P.C., "A Fully Compensated Digital Ultrasonic Sensor for Distance Measurement," *Instrumentation and Measurement, IEEE Transactions* on , vol.33, no.2, pp.128,129, June 1984

[5] Equestrian Australia. January (2014). *National Jumping Rules*. Link: http://www.equestrian.org.au/Jumping-rules

[6] Cytron Technologies (2009). Power Window Motor (Wira). *Cytron Technologies Sdn. Bhd*.: User Manual.

[7] Nandy, D. (2011). *Grippers and lifting mechanism*.

[8] Tao Liu; Jian Sun, "Simulative calculation and optimal design of scissor lifting mechanism," *Control and Decision Conference*, 2009. CCDC '09. *Chinese* , vol., no., pp.2079,2082, 17-19 June 2009

[9] R.R. BRINK SYSTEMS, INC. (2011), *Sliding Door Electric Unlock Move System*. Link: http://www.rrbrink.com/catalog/57300.pdf

[10] Cytron Technologies (2009). 16x2 Characters LCD. *Cytron Technologies Sdn. Bhd*.: User Manual.

[11] Manesis, S.; Deligiannis, V., "Automata-based modeling and control of a truck and trailer vehicle equipped with a kingpin sliding mechanism," *Emerging Technologies & Factory Automation, 2009*. ETFA 2009. *IEEE Conference* on , vol., no., pp.1,4, 22-25 Sept. 2009

[12] Nurul Syuhadah Bt Haji Othman (2013). *Development Of Ultrasonic Communication System*. Bachelor of Engineering. UTM, Universiti Teknologi Malaysia, Skudai, June 2013.

[13] Engineering Design and Technology Series (2014), “Student’s Guide to Learning SolidWorks Software”, *Learn Solidworks*. Solidworks Tutorial.

[14] Cytron Technologies (2009). Dual Channel 10A DC Motor Driver. *Cytron Technologies Sdn. Bhd.*: User Manual.

[15] From Page " arduino.cc", Edited by: 2013, Retrieved on: 17/12/2013, Title: How an Atomic Clock Works, Link: http://arduino.cc/en/Main/arduinoBoardUno

APPENDIX A

The Code of the Whole Project

```
#include <Ultrasonic.h>
#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 1, 0);
Ultrasonic ultrasonic(7,6);

#define pwm1 9
#define dir1 8
#define pwm2 10
#define dir2 13

const int buttonPin = 2;
const int buttonPin2 = 3;

int buttonState = 0;
int buttonState2 = 0;

int pwm_value=0;
int hits = 10;
int height=hits, h1;

int qias = ultrasonic.Ranging(CM);
int out, range;

void setup() {
  lcd.begin(16, 2);
  lcd.print("testing...");
  pinMode(pwm1,OUTPUT);
```

```
    pinMode(dir1,OUTPUT);
    pinMode(pwm2,OUTPUT);
    pinMode(dir2,OUTPUT);
    pinMode(buttonPin, INPUT);
    start();

}

void loop()
{
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Height: ");
  lcd.print(height);
  lcd.setCursor(0, 1);
  lcd.print("Distance: ");
  lcd.print(ultrasonic.Ranging(CM));
  lcd.print("cm");

 // check which button is pressed.
  buttonState = digitalRead(buttonPin);
  buttonState2 = digitalRead(buttonPin2);
  int pwm_value=0;

  // To increase the height is the desired height
  if (buttonState == HIGH)
  {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Moving Up ");
    delay (200);

    range =0 ;
    out =0;
```

```
do{
 buttonState = digitalRead(buttonPin);
 buttonState2 = digitalRead(buttonPin2);

  if (buttonState == HIGH){
    range = range + 5;
    hits = hits + 5;
    height = hits;

    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("UP By ");
    lcd.print(range);
    lcd.print("cm");
    delay (200);
         }

  if (buttonState2 == HIGH) {

      if (height > 50) {
            height=50;
            hits = height;
    }

    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("READY ");
    lcd.setCursor(0, 1);
    lcd.print("New Height:  ");
    lcd.print(height);
    lcd.print("cm");
    delay (500);

    out=1;
      }

  } while(out==0);
```

```
  buttonState = digitalRead(buttonPin);
  buttonState2 = digitalRead(buttonPin2);

   delay (500);
   moveup();
}

  // To decrease the height is the desired height
if (buttonState2 == HIGH)
{
 lcd.clear();
 lcd.setCursor(0, 0);
 lcd.print("Moving Down ");
 delay (200);

 out =0;
 range =0;

do{

  buttonState = digitalRead(buttonPin);
  buttonState2 = digitalRead(buttonPin2);

   if (buttonState2 == HIGH){
     range = range - 5;
     hits = hits - 5;
     height = hits;

     lcd.clear();
     lcd.setCursor(0, 0);
     lcd.print("Down By ");
     lcd.print(-range);
     lcd.print(" cm ");
     delay (200);


   }
```

```
        if (buttonState == HIGH) {
            if (height < 10){
                    height = 10;
                    hits = height;
         }

         lcd.clear();
         lcd.setCursor(0, 0);
         lcd.print("READY ");
         lcd.setCursor(0, 1);
         lcd.print("New Height:  ");
         lcd.print(height);
         lcd.print("cm");
         delay (500);

         out=1;
        }



        } while(out==0);

      buttonState = digitalRead(buttonPin);
      buttonState2 = digitalRead(buttonPin2);

       delay (500);
       movedown();

        }
delay (200);
 }

void start()
{
  ultrasonic.Ranging(CM);

if(ultrasonic.Ranging(CM) > 11 )
  { delay (30);
```

```
do{

 lcd.clear();
 lcd.setCursor(0, 0);
 lcd.print("Distance: ");
 lcd.print(ultrasonic.Ranging(CM));
 lcd.setCursor(0, 1);
 lcd.print("initializing.. ");

 digitalWrite(dir1,HIGH);
 digitalWrite(dir2,HIGH);

  pwm_value=150;
 analogWrite(pwm1,pwm_value);

 pwm_value=180;
 analogWrite(pwm2,pwm_value);

    }

   } while(ultrasonic.Ranging(CM) > 11);
 int pwm_value=0;
 analogWrite(pwm1,pwm_value);
 analogWrite(pwm2,pwm_value);

}

 lcd.clear();
 lcd.setCursor(0, 0);
 lcd.print("READY TO GO!");
 lcd.setCursor(0, 1);
 lcd.print("                ");
 delay(5000);

}
```

```
void movedown(){
  if (ultrasonic.Ranging(CM) > height)
  {
  do{


      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("New Height: ");
      lcd.print(height);
      lcd.setCursor(0, 1);
      lcd.print("Distance: ");
      lcd.print(ultrasonic.Ranging(CM));
      lcd.print("cm");

      digitalWrite(dir1,HIGH);
      digitalWrite(dir2,HIGH);

      pwm_value=150;
     analogWrite(pwm1,pwm_value);

     pwm_value=180;
     analogWrite(pwm2,pwm_value);

       }



    }    while(ultrasonic.Ranging(CM) > height);
    int pwm_value =0;
    analogWrite(pwm1,pwm_value);
    analogWrite(pwm2,pwm_value);
    delay (1500);

  }
}

void moveup(){
  if (ultrasonic.Ranging(CM) < height)
```

```
 {
 do{
   lcd.clear();
   lcd.setCursor(0, 0);
   lcd.print("New Height: ");
   lcd.print(height);
   lcd.setCursor(0, 1);
   lcd.print("Distance: ");
   lcd.print(ultrasonic.Ranging(CM));
   lcd.print("cm");

   digitalWrite(dir1,LOW);
   digitalWrite(dir2,LOW);

   pwm_value=150;
   analogWrite(pwm1,pwm_value);

   pwm_value=215;
   analogWrite(pwm2,pwm_value);

  }   while(ultrasonic.Ranging(CM) < height);
  int pwm_value=0;
  analogWrite(pwm1,pwm_value);
  analogWrite(pwm2,pwm_value);
  delay (1500);

 }
}
```