FAKULTI KEJURUTERAAN ELEKTRIK

LAPORAN PROJECT SARJANA MUDA

DESIGN AND DEVELOPMENT OF MOTOR CONTROL USING EMG-FORCE SIGNAL

CHOOI KAH YUNG

Bachelor of Mechatronics Engineering

May 2014

" I hereby declare that I have read through this report entitle "Design and development of motor control using EMG-force signal" and found that it has comply the partial fulfilment for awarding the degree of Bachelor of Electrical Engineering (Mechatronics Engineering)"

Signature : ....

Supervisor's Name : ....

Date : ....

DESIGN AND DEVELOPMENT OF MOTOR CONTROL USING EMG-FORCE SIGNAL

CHOOI KAH YUNG

A report submitted in partial fulfilment of the requirements for the degree of Bachelor of Mechatronics Engineering

Faculty of Electrical Engineering

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2014

I declare that this report entitle “Design and development of motor control using EMG-force signal” is the result of my own research except as cited in the references. The report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : ....

Name : ....

Date : ....

To my beloved mother and father

ACKNOWLEDGEMENT

In preparing this report, besides searching for related article and journal, I also get in contact with many people inside my University, including lecturers, researchers and colleagues. I have learned many knowledge and they have greatly contributed to me by triggering the new ideas. I wish to express my sincere appreciation to my main project supervisor, Dr. Muhammad Fahmi bin Miskon and my friend Mohammad Ihsan bin Sabri. They has helps me a lot by giving me suggestion, opinion and guidance, helping me finish this report..

I am also would like to thank my university for funding some of my material, reducing my burden. Thank shall given to all my friend who provide me assistance volunteer be the respondents on this project. Other than that, I am also grateful to all my family members.

ABSTRACT

Electromyography signal also called EMG signal, it is generated in our muscle when our muscle unit is contract or expand. The high speed progression in technology has made us possible to detect the EMG signal and use it as the control signal for other applications. This report describes the development of position control of DC motor with encoder by using EMG signal from biceps brachii muscle. EMG signal is received by using sensor, sending it into microcontroller for feature extraction and classification process. Elbow joint angle is mapped with the EMG signal and the rotation angle of DC motor is synchronized with joint angle. Experiments were conducted to analyze the relationship between joint angle, torque and EMG signal. Results showed that additional torque can increase the signal to noise ration in the equipment. For the relationship between joint angle and EMG signal, it exists 2 type of shape and the mathematics model between each respondent is different. A new mapping method has been designed in this project. It map the EMG signal to joint angle by identify the closest EMG value that represent respective joint angle. Result shows that it can classify the EMG signal but calibration needs to be done to counter the fatigue factor.

ABSTRAK

Isyarat Electromyography juga dipannggil isyarat EMG, ia dihasil dalam otot kita apabila otat kita kembang atau kontrak. Teknologi yang semakin memaju telah membuatkan kita dapat mengesan isyarat EMG dan menggunakan ia untuk aplikasi yang lain. Laporan ini menceritakan kawalan DC motor pengekod dengan mengunakan isyarat EMG dari otot bicep lengan kita. Isyarat EMG akan direkodkan mengunakan sensor, hantar ia ke mikropengawal untuk ekstrak ciri-ciri yang terdapat dalam isyarat dan klasifikasi proses. Sudut siku akan dipetakan dengan isyarat EMG dan sudut pusingan DC motor akan menyegerakan dengan sudut siku. Eksperimen akan dijalankan untuk menganalisis hubungan antara isyarat EMG, tork dan sudut siku. Keputusan menunjukkan bahawa tork akan meningkatkan nisbah isyarat kepada bunyi. Bagi hubungan antara isyarat EMG dan sudut siku, ia didapati terdapat 2 bentuk graf dan model matematik antara responden adalah berbeza. Permetaan yang baru telah dicipta dalam projek ini. Ia mempetakan isyarat EMG kepada sudut siku dengan mengklasifikasi isyarat EMG yang terdekat yang mewakili sudut siku. Keputusan menunjukkan ia boleh mengklasifikasi sudut siku tetapi penentukuran perlu dijalankan untuk menyelesaikan isu keletihan.

TABLE OF CONTENTS

LIST OF TABLE

LIST OF FIGURES

FIGURE TITLE PAGES

LIST OF SYMBOLS

N	-	Sample size
V_T	-	Terminal voltage
E_A	-	Induced voltage
I_A	-	Armature current
R_A	-	Armature resistance
k	-	Constant
ω	-	Angular velocity
ϕ	-	Magnetic flux
τ	-	Torque
F	-	Force
r	-	Radius
m	-	Mass
g	-	Gravitational force, 9.81 m/s
Θ	-	Angle (degree)

LIST OF APPENDIX

APPENDIX **TITLE** **PAGES**

CHAPTER 1

INTRODUCTION

1.1 Motivation

Stroke is a kind of brain attack that occurs when there is a disrupt in the blood supply to the brain. According to National Stroke Association of Malaysia (NASAM), stroke is the third largest cause of death in Malaysia[1]. Every year, there is estimate 40000 people suffer from stroke. One of the effects of stroke is the disability of certain part of body and it depends on which part of the brain is damaged. Based on the stroke statistics from University hospital, New Jersey, stroke is the main cause of disability among adults in US[2]. Rehabilitation needed to be done to improve patient's body function so that they can become independent to the other. Nowadays, therapist can get help from artificial device in rehabilitation process. Human exoskeleton interaction can be done in 3 methods. The first method is by brain activity or Electroencephalography (EEG), the second method is using muscle signal or Electromyography (EMG). The last method is generation of the assisting exoskeleton movements.

Robot is suitable to be used in rehabilitation because it can provide precise movement, able to collect data from the user performance[3]. Recent technologies have made it possible to use robot as the assistant to the therapist. It can assist therapist to conduct rehabilitation programs without monitored by therapist and can work for a long time. One of the example is rehabilitation robot which named PolyJBot from Tong and Hu [4]. It was developed to assist stroke subjects to actively train their wrist, elbow and ankle using their EMG signal as the intention driven signal. Figure 1 shows the exoskeleton robotics training device from Hong Kong Polytechnics University. It is targeted for stroke patients and allows them control the robotics fingers by using their own surface EMG signal.

Figure 1: Exoskeleton robot training device

1.2 Problem Statement

The characteristic of EMG signal should be studied before it could be applied to control the rotation of a motor. Information like joint angle and torque can be obtained in EMG signal. However different people will have different muscle firing frequency, ratio of slow to fast-switch fibre, fatigue status and muscle unit, hence the EMG signal can't directly compare across the subjects. To control the position of the motor, the relationship between the EMG signal, torque and position have to be study before using it to control the motor actuator.

Researched questions are described as below:

a) What are the available features that could provide the most accurate and easiest mapping between EMG-force signal and joint angle?

b) What is the relationship or mathematic model between EMG signal and joint angle?

c) How to map the EMG-force signal to the arm's joint angle in real time for different user?

d) How to synchronize the motion of motor actuator to our arm by minimum error and smoothly?

1.3 Project Objectives

The objectives of this project are defined as below:

a) To study the relationship between joint angle, torque and EMG-force signal

b) To synchronize the rotation angle of motor and elbow joint angle by mapping EMG-force signal from biceps brachii muscle.

1.4 Scope of the Project

The scopes of works are described as below

a) 4 respondents are selected for experiment 2 and 3. All EMG signal are retracted from biceps brachii muscle in right arm by using disposable surface electrode.

b) All experiments are carried under minimum fatigue condition. The experiments are focused on isometric contraction (static position).

c) The rotation angle of DC motor actuator and elbow joint angle is limit from 0 to 90 degree only.

1.5 List of Contribution

The contribution of this thesis is:

a) Development of a new mapping method that can accurate synchronizes and maps the joint angle and EMG signal in real time and suitable for different user.

b) Development of a new method to control a DC motor by using human arm motion.

1.6 Outline of Dissertation

This report contains detail information of the methodology of the project and the material we using in this project. Chapter 2 is literature review. It contains background theory and some result that related to this project by other researcher. In chapter 3, research methodology, experiment setup, the feature and characteristic of material used will be showed. The detail procedure about the conduction of experiment will also discussed in chapter 3. Chapter 4 shows the results that have been generated. Any analysis, discussion or calculation that involved will be present here. Conclusion of the result will showed in chapter 5. The list of item used, programming coding and its flow chart will be attached in appendix.

CHAPTER 2

THEORETICAL BACKGROUND AND LITERATURE REVIEW

2.1 Theory of motor control using EMG-force signal

This section introduces the EMG signal, signal processing, Surface EMG (SEMG) collection, feature extraction, pattern classification and model of human upper limb. Figure 2 shows the block diagram of actuator control system using EMG-force signal.

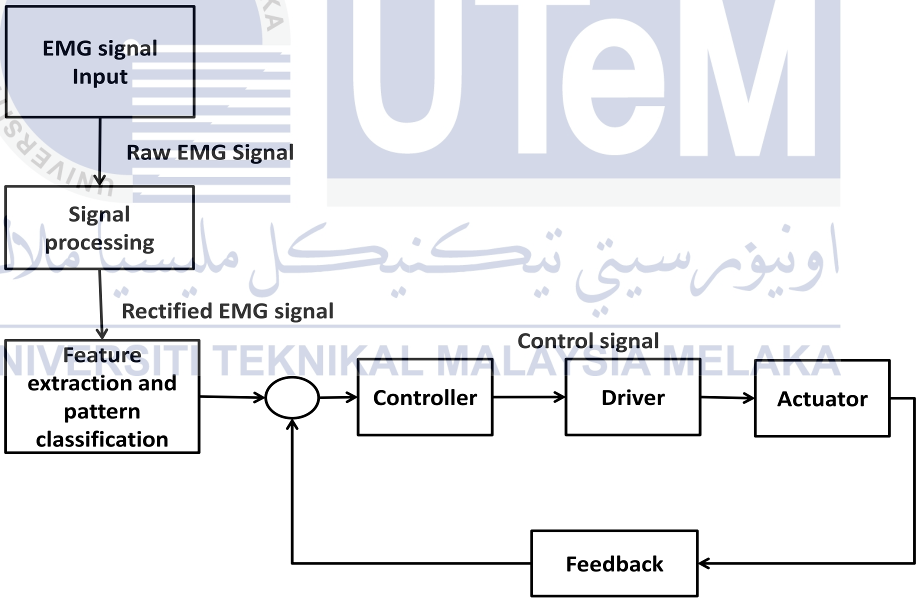


Figure 2: Block diagram of motor control using EMG-Force signal

2.1.1 Electromyography (EMG) signal

Electromyography (EMG) refers to recording of muscle's electric activities [5]. It also is an experimental technique concerned with the development, recording and analysis of myoelectric signals. Figure 3 shows the raw EMG signal during muscle contraction.

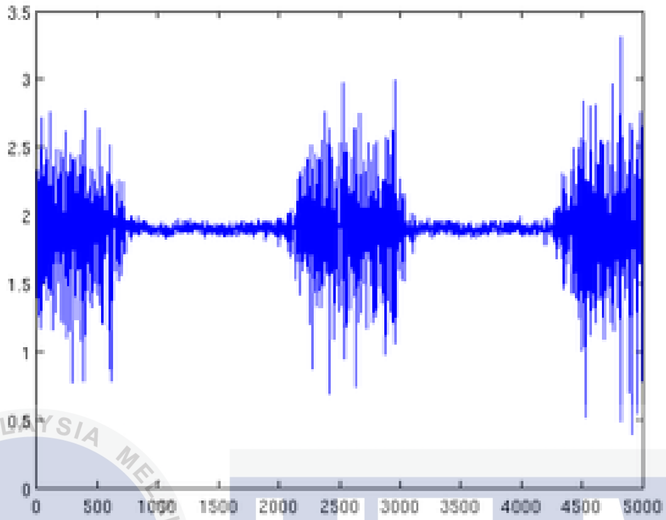


Figure 3: Raw EMG signal

Myoelectric signals are formed by physiological variations in the state of muscle fibre membranes. The amplitude of EMG signal is depends on the muscle contraction. The stronger the contraction of the muscle, the higher the amplitude of EMG produced by that muscle. Dynamic action like rotating joint angle or static action like holding the load can increase our muscle contraction. Other than that, the shape and amplitude of EMG also depends on amount of motor unit and firing rate, hence different people will have different amplitude and shape of EMG signal.

2.1.2 Signal Processing

Since raw EMG signal has very low voltage, typical EMG electric potentials is between 50 μV up to 20 or 30 mV in amplitude[6]. Practical EMG signal contains some noise due to the influence of electronic component and the electric power cable, it has to go through signal processing unit before we can process it. Raw EMG signal has to amplifier, go through band pass filter to filter the noise. Normally frequency of surface EMG ranged from 0-500Hz, but only 50-150Hz is the useable energy[7]. Controller like microcontroller and Arduino can't process negative voltage signal. To easier the analysis process, the entire negative signal have to be eliminated. There are 2 methods to eliminate the negative signal, one is by full rectification process, and the other one is by adding offset voltage into the signal to neutralize the negative signal [8].

2.1.3 Surface EMG data collection

EMG signals are measured within the muscle tissue itself or at the external derma by means of either needle or surface electrodes. Electrodes are sensors that detect electrical potential generated inside the nerves and muscles[9]. Even though EMG signal which recorded by needle is more accurate than surface electrodes, however modern researcher more prefer to surface electrodes as it is more convenient and bring no pain to respondents.

There are 2 types of electrode that can be used to capture the surface EMG signal, unipolar electrode and bipolar electrode. Unipolar electrode is a type of electrode which only requires one electrode to capture the signal. For the bipolar electrode, it requires 2 electrodes, 1 of the electrode placed in the centre of the muscle to capture the muscle signal while another electrode placed in the end of muscle to acts as the reference point. The output is the difference of voltage level between those 2 electrodes.

The high impedance between our skins and surface electrode will weaken the signal received by electrode. To improve the performance of the surface electrode, the skin surface should be watch and cleaned using alcohol. Hair should be shaved to increase the touching area between the surface electrode and our skin.[10] Electrolytic gel like silver chloride gel is used to improve the contact between the electrode and the skin, it act as the medium to easier the transfer of electrons from skin to electrode.

Position of electrode plays a main part in getting an accurate result. Different set of muscle have different locations for the electrode. Figure 4 shows the location for biceps brachii muscle.

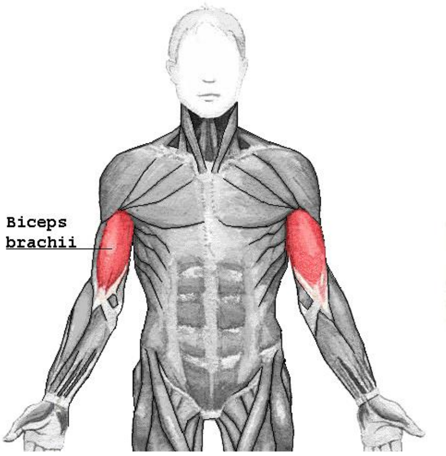


Figure 4: Biceps brachii muscle

Based on recommendation of SENIAM (Surface Electromyography for the Non-Invasive Assessment of Muscles) on biceps brachii, it is recommended that the signal electrodes to be placed on the line between the medial acromion and the fossa cubit at 1/3 from the fossa cubit, while the reference electrode placed on or around the wrist[11]. Figure 5 shows the location of the electrode for biceps bacchii muscle which recommended by SENIAM. Another method is placed the reference electrode 2cm parallel from the signal electrode[12]. This method is applied to decrease the effect of muscle belly dislocation due to the flexion activity. It also act as a standardize method when involves more than 1 respondents.

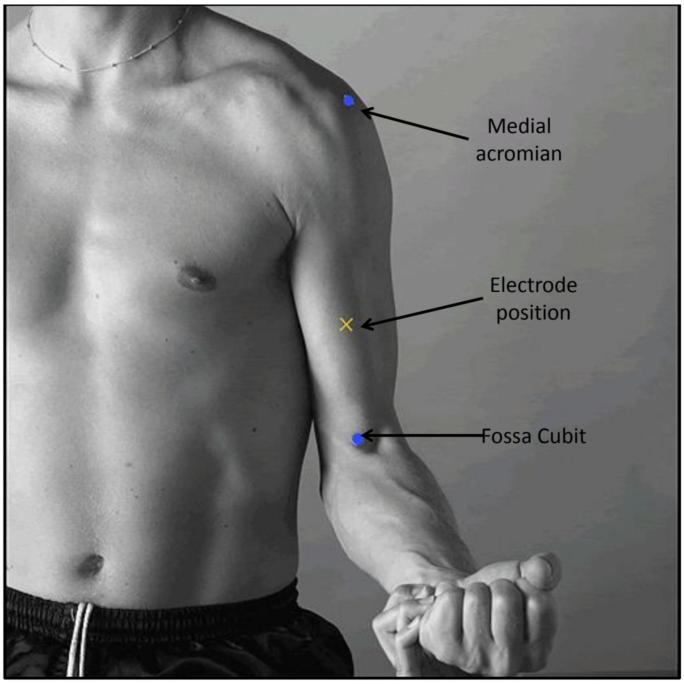


Figure 5: Location of electrodes for biceps brachii

2.1.4 Feature extraction and Pattern Recognition

Feature extraction will define the characteristic of an EMG signal. EMG signal has to go through feature extraction to extract the information from the signal. So far, there are at least total of 19 features that available to be used to classify the motion [13]. Each of them have different rate of motion classification. Some of them are Integral of absolute value (IAV), variance (VAR), Zero crossing, and energy of wavelet coefficient. For the simple motion which involves 1 set of muscle, moving average and root means square (RMS) are the most common feature used for real time analysis. For moving average, a certain amount of data are averaged using the gliding window technique [12]. It used for rectified signals and serves as an estimator of the amplitude behaviour. Below show the calculation of moving average value.

$$\text{Moving average} = \frac{\sum_{i=0}^{N-1}|x(i)|}{N} \quad (2.1)$$

N=sample size

Root Mean Square (RMS) is based on the square root calculation, the RMS reflects the mean power of the signal and is the preferred recommendation for smoothing and estimate the power of signal [6]. In practical, RMS is better than moving average as it can eliminate the noise in the signal.

$$\text{RMS} = \sqrt{\frac{\sum_{i=0}^{N-1} X^2(i)}{N}} \qquad (2.2)$$

N = Sample size

EMG signal can be classify either in time domain or in frequency domain. There are few methods to classify the pattern of EMG signal. The first method is by setting a threshold after analysis the signal. When the signal value is more or less than the threshold value, control signal will be given out to control the actuator[14].

The second method is by mathematic model. By using moving average feature, K.Ullah and J.H Kim showed that the relationship between the torque and joint angle to EMG signal is exponential in nature, a new mathematical model can be produced to match the torque to the EMG signal by using Levenberg-Marquardt Algorithm [15]. Figure 6 shows the results get by K. Ullah. As showed in figure 6, the torque and joing angle is proportional to the amplitude of moving average signal.

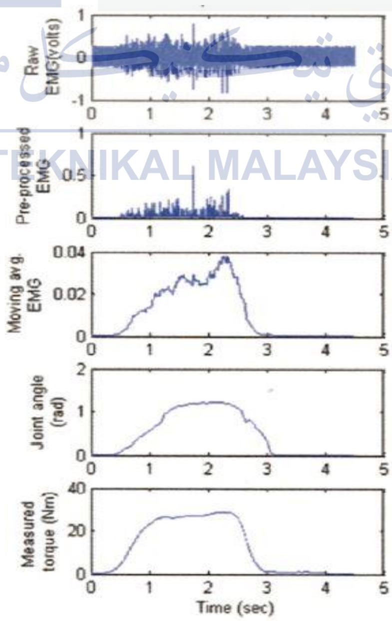


Figure 6: The relationship between EMG, joing angle and torque

The third method is by using neural network. Neural network is powerful in classification purpose, hence it is suitable to be used for the task that involved more than 1 channel of EMG signal or more than 1 set of muscle. In the rehabilitation robotic exoskeleton done by Y.J. Fun, neuro fuzzy controller is used to predict the motion of user in real time and further produce the signal to control the exoskeleton [16]. The neural network contains 5 layers and make the whole system contains flexible and non-linear adaptive characteristics. Experiments have proved that the fuzzy neural network is effective in classification and prediction of user's motion.

2.1.5 Model of Human Upper Arm

Figure 7 shows the model of human arm under different joint angle [17].

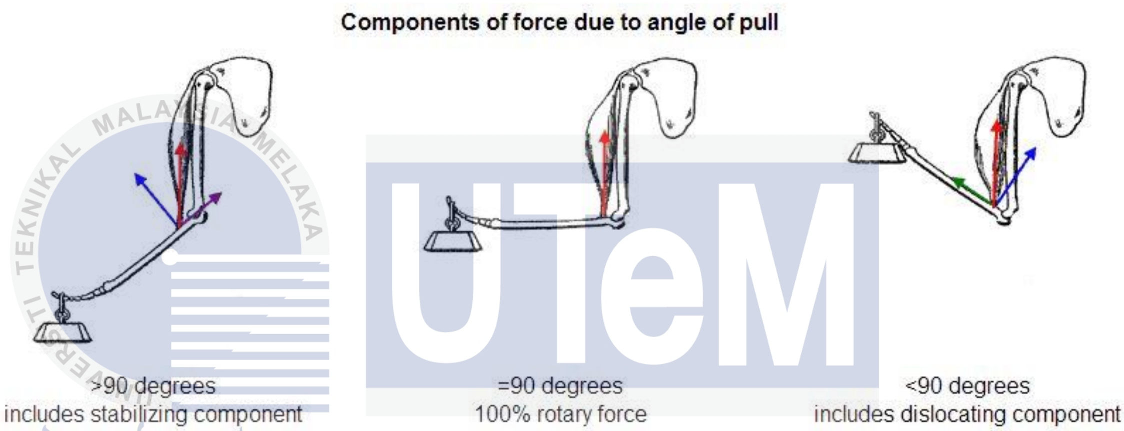


Figure 7: Model of human arm under different joint angle

Based on figure 7, it shows that there is several type of force exists on the joint when lifting up a weight. There are rotary component, stabilizing component and dislocating component.

Angle of pull : The angle between muscle and bone (red colour in figure 6).

Rotary component : Muscle's force which contribute to bone movement around joint axis (blue colour in figure 6)

Stabilizing component : Degree of parallel force generated on the bone and joint when angle is less than 90 degree (purple colour in figure 6).

Dislocating component : Degree of parallel force generated on the bone and joint when angle is more than 90 degree. (green colour in figure 6).

When the angle of elbow is less than 90 degree, only rotary component and stabilizing component is exist in the elbow. When the angle of elbow is at 90 degree, rotary component is at maximum value while stabilizing component is at minimum. When the angle of elbow is more than 90 degree, only rotary component and dislocating component exist in the elbow.

F.P Carpes showed that the the isometric torque produced in the elbow is maximum at 90 degree[18]. Figure 8 shows the result about the isometric torque in elbow vs the joint angle. After 90 degree, the torque starts decreasing. Hence the angle of this project is limited from 0 to 90 degree only.

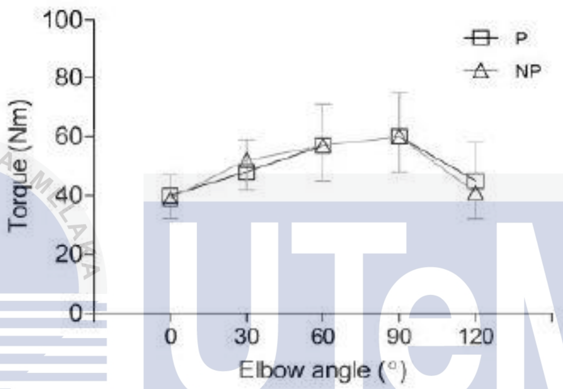


Figure 8: Isometric torque vs joint angle

In 2007, j.Y Gefen showed that anterior deltoid was assist in isometric elbow extension during a weight lifting using elbow [19]. Figure 9 shows location of anterior deltoid muscle.

Figure 9: Anterior deltoid muscle

His results showed that there is increasing activity in anterior deltoid when the elbow joint angle is over a certain degree. This means that when we lifting up an weight using the elbow, once the elbow joint angle is over a certain degree, there will be some force shift from biceps brachii to anterior deltoid, the torque exerted by biceps brachii will decreases.

2.1.6 Motor Actuator Control

Actuator is the device which perform task by linear or rotation mechanism either by receiving electrical, pneumatic or hydraulic energy. It also can be state that actuator is a device which convert energy into motion. Electric motor actuator is the most common actuator used in market due to its compact size, easily to assemble with supply power and low cost. DC motor is the type of electric motor which use DC voltage as the supply and convert it to rotational motion. Shunt DC motor is a motor whose field circuit get its supply power directly from the armature terminals of the motor [20]. Figure 10 shows the circuit of shunt dc motor while figure 11 shows the speed torque characteristic of dc shunt motor. The rotation speed of rotor is inversely proportional to the motor induced torque.

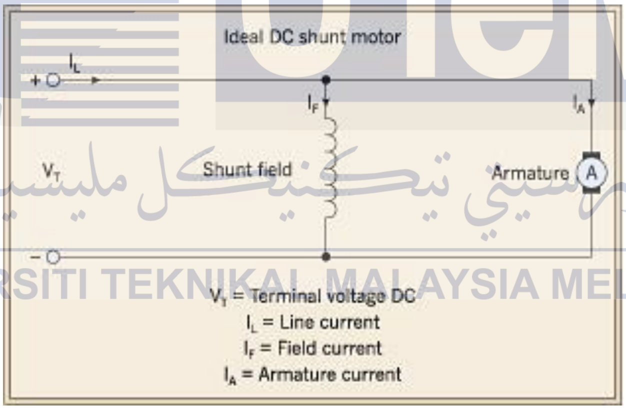


Figure 10: Circuit of shunt dc motor

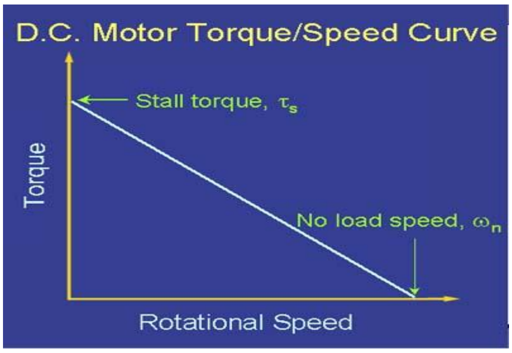


Figure 11: Speed torque characteristics

The terminal voltage, V_T is expressed as the total of generated voltage in rotor due to rotational motion, E_A, and $I_A R_A$ = voltage drop in armature circuit.

$$V_T = E_A + I_A R_A \tag{2.3}$$

When the load on the motor is increase, the rotor speed will decrease and further decrease the generated voltage as:

$$E_A = K\omega\phi \tag{2.4}$$

K= constant, ω = angular velocity, ϕ = magnetic flux

The decreasing in E_A will cause the $I_A R_A$ in equation 2.3 to be increase. Since R_A is constant, so I_A will increase. As the current increase, the induced torque will also increase as

$$\tau = k\phi I_A \tag{2.5}$$

τ= Induced torque

The increasing in induced torque will increase the rotor rotation speed. As the rotation speed increase, the increase in E_A in equation 2.3 will reduce I_A and further reduce the induced torque and thus rotation speed of the rotor will decrease again. These conditions will repeat and finally the induced torque will equal the load torque at a lower rotation speed, hence it can be conclude that as the torque increase, some of the power in rotational speed form will transfer to torque to counter the increasing load torque by decreasing its rotation speed.

Rotary encoder is existed is some kind of DC motor, these motor is called DC motor with encoder. Rotary encoder is a electro-mechanical device which convert angular position of the rotor to digital code. In position control of DC motor, at least 2-bit binary of encoder is used to identify the location of rotor.

2.2 Problems of DC motor control using EMG-force signal

The biggest problem facing in controlling an actuator by using EMG signal is the EMG signal to joint angle mapping problem. In time domain analysis, researcher can extract the feature of torque and angle, however based research of Ameri, the neural network performance of position estimation was poor if compare to force estimation by using EMG signal in time domain[21]. Without any load attach to the user, the torque produced is low and hence it lower the amplitude of EMG signal. The low amplitude of raw EMG signal will cause neural network hard to discriminate the angle and other information. Furthermore the lower amplitude of EMG signal will also decrease the signal to noise ration in the equipment.

K. Ullah did show the shape of EMG signal to the joint torque and joint angle in his result [15]. Based on the result, the relationship between joint torque and EMG signal is in exponential shape and not linear. It is hard to obtain an accurate exponential equation based on a graph and data obtained. Hence the features used in information extraction have to be select carefully to obtain a more linear graph.

As the project involved biological signal which is impossible to obtained 100% same signal for same motion, hence standard procedure of raw EMG data recording has to be followed to obtained a more accurate and consistent result. Other environment factor like room temperature & humidity will produce sweat which will affect the skin impedance and further effect the accuracy of collected EMG data. Human factor like fatigue conditions has to be controlled. Fatigue can increases the amplitude of EMG signal. It is hard to predict the changing of the signal because fatigue is depend on the workload such like the torque generated, the speed of when lifting up the load and human metabolism. To reduce the fatigue factor, the resting time for the muscle should be controlled well so that the experiment won't consume too much time and ensure the muscle doesn't damage during the experiment. The accumulated heat generated by the flow of current in the electrode cable will also affect the voltage level of EMG signal, hence the experiment should be conducted in the shortest time. As the voltage pass through in the electrode cable is very low, hence the motion of electrode cable should keep as low as we can.

The motor encoder which used in the project also may be the problem. If the speed of microcontroller is not fast enough to pick up all the encoder value, then the final destination of rotor will not be accurate. The inertia problem also may cause the rotor turning even though the voltage supplies to it have been cut off.

2.3 Performance Indices

To be considered success in this research, there are several critical requirements that must be full fill.

a) Good feature extraction and classification from EMG signal:

-The method chosen in extract the feature of EMG signal must be good in maximum class separability to prevent overlap of the feature. It also must have good robustness preserve the cluster separability in a noisy environment. The feature extraction method should have low complexity to allow other procedure can be running in real time[13].

b) The angle of rotor for dc motor should synchronize with the arm motion

-The speed of the rotor and arm motion should be synchronizing to prevent any noticeable delay. The time taken in processing the signal in Arduino board should be as short as possible to prevent Arduino board spent too many time in process the signal and further causing delay.

c) The position control of DC motor should have high accuracy.

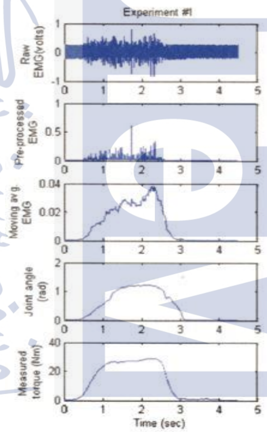
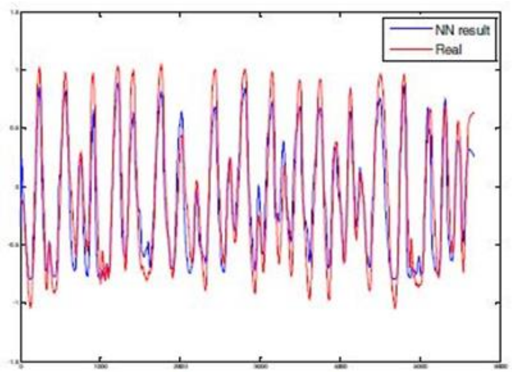
-The rotor of dc motor can stop in the right angle without high error. The inertia caused by the rotation of motor should be controlled.

2.4 Comparison among available solutions – trade off

This section discuss about available solutions to solve the problem and achieve the objective of this project. Based on the researches that have been done by other researcher, it can be concluded that basically there are 3 type of methods that can be used to control the actuator by using EMG signal. Method and solution from other researcher and other similar project title are list down in table 1.

Table 1: Comparison among variable solution

Title	DC Motor Control using EMG Signal for Prosthesis	Mathematical Model for Mapping EMG Signal toJoint Torque for the Human Elbow Joint using Nonlinear Regression[15]	Continuous position control of 1 DOF manipulator using EMG signals[22]
Arthur	C.Mohan, V.K.Giri	K.Ullah, J.H.Kim	W. Ryu, B.Han, J. Kim
Year	2011	2009	2008
Objective	Control a DC motor moving forward and backward by using EMG signal	Study the relationship between joint torque and EMG signal	Continuous position control of wrist
Set of muscle	flexor-pronator v. extensor supinatorgroup, more than 1 set	1 set	4 set
Experiment Setup	Collect EMG signal from forearm under single and double action. Data is processed in microcontroller	Respondents perform non-fatiguing and variable force maximal voluntary contraction.	Collect EMG signal under various condition like different stiffness and speed.
Data Collected	Raw EMG data for single action	Joint torque, joint angle, raw EMG signal at	Raw EMG under Sampling size =

	and double action of muscle	sampling frequency of 1Khz	200HZ, wrist joint angle
Feature extraction and pattern reorganization	Count the transition pulse of EMG signal. Forward or reverse the motor direction according to the number of count	Moving average feature is applied. Levenberg-Marquardt Algorithm is applied to calculate the parameter for mathematic model. Neural network is used to get the best mathematic model.	Moving average feature is applied for every 5 data, FIR filter is applied to get quasi tension (Filtered EMG signal). A supervised multi-layer neural network with a back propagation algorithm is used to estimate joint angle from quasi
Results and Limitation	Able to control direction of motor by using EMG signal, but unable control its rotor position at certain point	A mathematic model can be produced. It is in non-linear exponential form. Successful map EMG to joint torque  Figure 6	High correlation between measured and estimate joint angle, able to continuous position control 

Based on table 1, there are 3 type of method that can be used by researcher to control the actuator by using EMG signal. The first type is rely on a threshold value that set by researcher after EMG signal go through features extraction and analysis. This type of method is represented by C. Mohan in table 1. This method is suitable to be used when the number of classification or the number of operation is less, for example C. Mohan only need to run the motor in forward or backward only, it has only 2 number of classification. After go through analysis, a threshold value of EMG signal that separates the 2 conditions can be found. Researcher can declare what the actuator do if the EMG signal over or below that threshold value. This advantage of this method is easy and robust because the number of classification is less, hence the EMG signal can be classified easily as the range of EMG signal that represent the 2 type of condition is large. The disadvantage of this method is the performance of classification process will be decreasing if the number of classification is increasing. For example position control that require researcher classify the EMG signal from angle 0 degree to 90 degree, if the interval of angle is 10 degree, then it need to classify 10 set of conditions. As the EMG signal is variant and not consistent, sometimes the system can classify the angle wrongly when the 2 threshold values are close to each other.

The second type of method is by using mathematic model. It is represented by K. Ullah in table 1. K. Ullah has successfully found out a mathematic model that can used to represent the relationship between EMG signals to torque. Even though the research done by him is relate the EMG signal to torque, however based on the figure in table 1 or figure 5 in this thesis, the shape of joint angle and measured torque is almost similar, if the mathematic model between EMG signal and torque can be calculated, then the mathematic model between EMG signal and joint angle also possible be calculated. The advantage of this type of method is it can used in conditions which the number of classification is large such as continuous position control. However the disadvantage of this method is the mathematic model can be easier effect by other factor. This method is too depends on mathematic model, slightly change in the mathematic will effect the performance of whole system. Factor like fatigue issue during the real situation will surely affect the mathematic model. Other than that, in theory the EMG signal for each person is different, so it is not suitable used by variant user unless there is similar ration between their mathematic models.

The third method showed in table 1 is using neural network for the classification process. This method is presented by W. Ryu. Neural network is used by him to analysis and classify the pattern in the EMG signal. The advantage of this method is the performance of classification is high and it is suitable used for continuous position control. This method also suitable to be used for complex classification process which involves more than 1 set of muscles. Currently most of the researcher are using neural network for the EMG classification process no matter in time domain or frequency domain. This is because neural network has been proved useful in complex analysis and classification process, it also can self adapt to the changing of environment. The disadvantage of this method is the equipment requires large memory for the programming of neural network. Normal microcontroller does not have enough memory for the neural network.

2.5 Summary of literature review

EMG signal is generated by the activity of muscle, different people has different amplitude for their EMG signal[3]. There are many features available to extract feature from EMG signal, different features have different relationship with EMG signal and different performance [13]. For the recording of EMG signal, it is suggested to do it against resistance or holding some weight to increase the amplitude of signal and easier the classification. Moving average and RMS feature can be used to extract the information from EMG and then determine which is more suitable to be used in this position control.

For pattern classification, neural network has been proved useful and is more widely used by other researcher, However it is not recommend for this project due to the limitation of equipment. The first method that rely on threshold value or second method that rely on mathematic model may be suitable be used in this project. As the design of method must be suitable be used for variant user, hence the mathematic model for variant respondents need to be analysed to find out the similar characteristics before we can decide which method should be used.

Figure 12 shows the block diagram of this project together with its material while. Arduino Uno will be the main controller. The reading of the EMG signal will be processed and shows its angle value to LCD display. The moving average signal will be display on laptop to easier the analysis process. The motor actuator is a DC motor with encoder. The encoder will act as the feedback mechanism.

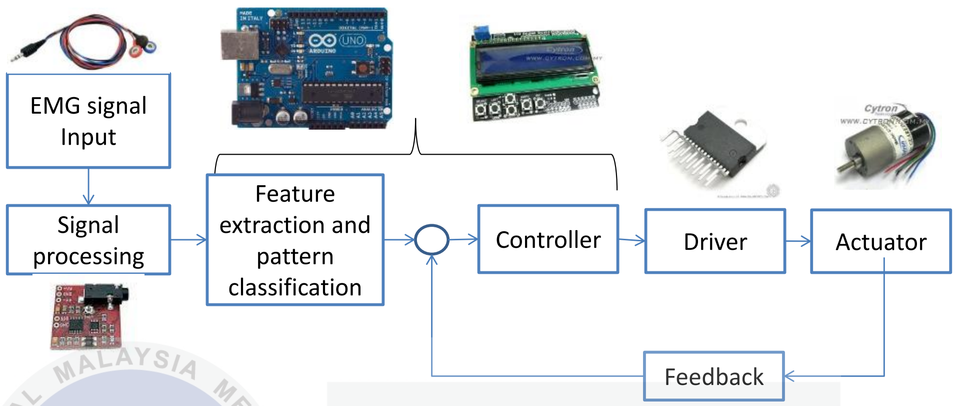


Figure 12: Block diagram with material

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Mapping of joint angle and surface EMG signal

Few experiments need to done before we can decide what type of classification method to be used in the programming. Based on the theory, the contraction of muscle or change in elbow joint angle will increase the frequency and amplitude of EMG signal. Experiment 1 will be conducted to identify the performance of sensor (signal to noise ration) by analysis the effect of load on the EMG signal for different joint angle. The weight selected in this experiment is based on the percentage of maximum voluntary contraction (MVC) of respondent. MVC represent the largest amount of torque or strength that can be given up by muscle[23]. 30% of MVC is selected because based on borg scale, 30% MVC is classified as moderate torque given out by muscle[24]. Experiment 2 will be conducted to further analyse the relationship between joint angle and EMG signal in higher angle resolution. A mathematic model that represent the EMG signal and joint angle will be calculated by using trendline function in Microsoft excel. In this experiment, the weight hold by respondent is decided based on the result from experiment 1. By referring the data collected from experiment 2, the classification method will be decided and an arduino program will be written to classify the joint angle from the collected EMG data. Experiment 3 will be run to test the performance of arduino program in term of error.

3.2 Validation of idea

This chapter will discuss about how to prove the concept and theory by setup the experiments. The target muscle is right arm biceps brachii. Experiment will be conduct to collect surface EMG signal from different joint angle. Experiment 1 is involving 1 respondent only while Experiment 2 and 3 are involving 4 respondents. Exoskeleton Mechanism is used in experiment 2 and 3 to make sure the upper arm of respondents is always straight and not moving during the whole experiment. Moving average and RMS features will be applied to convert the EMG reading to usable value. Microcontroller can't store number value that are very high, hence the RMS features need to modified. The amplitude of signal is divide by 5 before it go through RMS feature extraction, no modification on signal for moving average feature extraction. To standardize the experiment, the location of reference electrode is put 2 cm below the signal electrode for every respondent and the weight hold by respondent is based on their own MVC. To smooth the signal, the sampling time will be set as 0.2 seconds. This value is decided based on a study shows that 0.2 to 0.3 seconds is the clinically recognized maximum delay that be tolerated by the users[25].

3.2.1 Objectives

The objectives of experiments are listed as below:

a) To investigate the performance of sensor and effect of load on EMG signal for position control

b) To compare the characteristics of moving average and RMS feature in extracting the joint angle information

c) To investigate the relationship between EMG signal and joint angle, then create a program that can used to classify the angle based on EMG signal

d) To improve the performance of DC motor with encoder by eliminate any potential error

3.2.2 Experiment Setup

This section discuss about how to setup the experiment. There are total of 4 experiments. Experiment 1 is to investigate the performance of sensor and the effect of load on EMG signal for different elbow joint position, comparison among moving average feature and RMS will done here. The weight hold by following experiment 2 and 3 is based on this experiment result. Experiment 2 is to investigate the relationship between

EMG signal and joint angle in higher angle resolution. 4 respondents will be selected to conduct the experiment. Exoskeleton mechanism is used to increase the accuracy of data reading by fixing the upper arm. The target muscle is right arm biceps brachii. The results will go through analysis using Microsoft Excel and a program will be created to be used in experiment 3. Experiment 3 is to test the performance and accuracy of the created program for arduino under non-fatigue condition. 4 respondents will be chosen to run the experiment. Experiment 4 is to improve the performance of DC motor with encoder. In experiment 4, the performance of DC motor will be analysis and to minimize the error by finding out any potential error factor.

3.2.2.1 Experiment 1: The relationship between EMG signal and elbow joint angle with load and without load.

1 healthy respondent (respondent A) without muscle damage will be selected for the experiment. To determine the MVC of the respondent, respondent are required to lift up a 3kg weight to 90 degree of his elbow joint and hold it for 4 seconds. Increase the weight until the respondent unable to lift up or can't hold for 4 seconds. The maximum weight that can be lifted by the respondent will be the the weight for 100% MVC. The skin of respondent is washed and clean using alcohol pad, hair of the sensor contact area is shaved. The length of the upper arm from medial acromion to fossa cubit is measured. Figure 13 and figure 14 shows the placement of disposable electrode. Signal disposable electrode (red colour) is placed on 1/3 of the length of upper arm from the fossa cubit. The second disposable electrode or reference electrode (blue colour) is placed 2cm center to center below the first electrode. The ground disposable electrode (black colour) will be placed in the bony area in upper arm. The selected location is the bony area near the elbow. Respondent is required to wait for 3 minutes to stable the chemical reaction in disposable electrode.

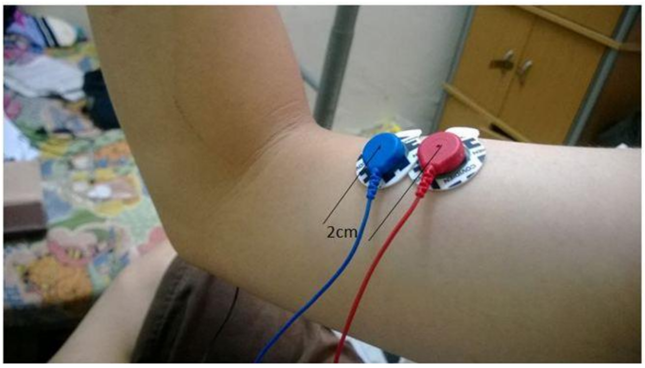


Figure 13: Position of signal and reference electrode

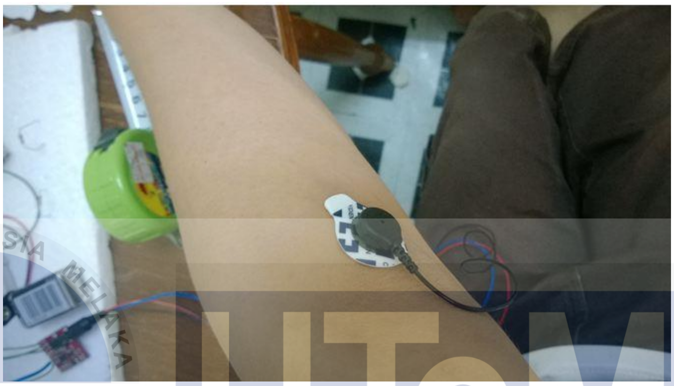


Figure 14: Position of ground electrode

Respondent is required sitting straight on a chair. Upper arm and forearm is kept in straight position downward (0 degree). The sensor-arduino circuit is showed in figure 15. Red LED will be used as the indicator of starting recording. First, without holding any weight, once the red LED is light up, surface EMG signal is recorded by using CoolTerm software for 10 second. The upper arm and forearm are fixed during the process. Once pass through 10 seconds, “end” will showed in laptop monitor and the recording will be stopped. This step is repeated by another 2 times.

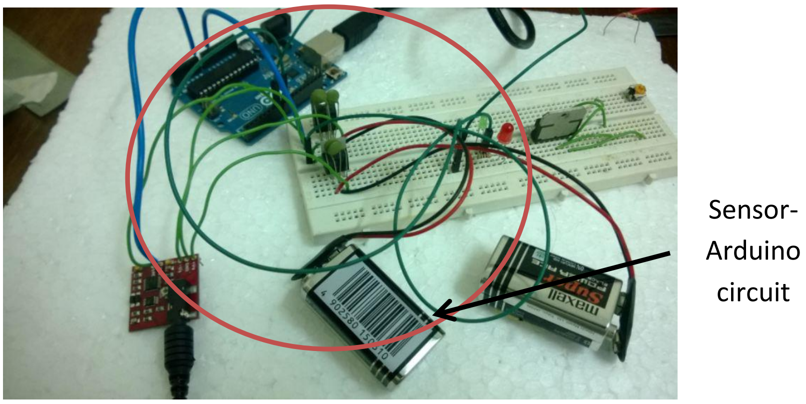


Figure 15: Sensor arduino circuit

Respondent is required to rotate the elbow from 0 degree to 20 degree within 2 seconds, hold for 4 seconds and return to 0 degree within 4 seconds. Time from 2 to 6 seconds represents the isometric contraction, green LED will light up to indicate this duration. 1 board as showed in figure 16 which marked with angle position is used to guide the respondent for position control. To minimize the parallel error, stick will be placed in the target angle. The elbow joint will rotate until the center of our palm is touched with the stick. Another stick is placed near the elbow to make sure the upper arm doesn't move forward during muscle contraction. Figure 17 shows the posture when holding the load on desired angle.

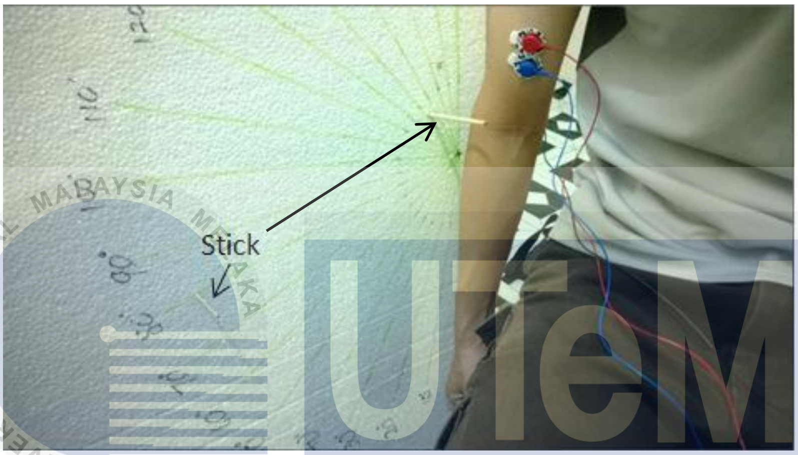


Figure 16: Board for position guiding

Figure 17: Isometric contraction

Once the red LED is light up, respondent start to lift up his arm to the desired degree within 2 seconds. As the green LED is light up, respondent will hold their arm in desired degree position. When the green LED is turn off, respondent return their arm to 0 degree or normal position. The recording is end once "end" is showed on monitor. The target degree is set as 20, 40, 60, and 80 degree. For each recording at certain degree, the process is repeated for another 2 times. There will be 1 minute resting time between each recording. The whole process will be repeated again with the 30% of MVC weight equipped on respondent.

3.2.2.2 Experiment 2: The relationship between EMG signal and joint angle for 4 respondents

4 respondents are selected to conduct the experiment. A exoskeleton mechanism as showed in figure 18 is used to fix the upper arm of respondents so it upper arm is not moving during the whole experiment. The moving of upper arm will effects the torque produced in joint to counter the load. A mah-jong paper which labels the angle from 0 to 90 degree is used to identify the joint angle. A stick as showed in red circle in figure 16 is positioned with the weight and it is point to the mah-jong paper which label with angle from 0 to 90 degree. It is used to assist respondent identify the joint angle during the experiment and minimize parallel error

Figure 18: Exoskeleton Mechanism

This experiment is similar with experiment 1 except that it is higher in angle resolution, involves more respondents and weight hold by respondents doesn’t change. Based on the result from experiment 1, 30% of MVC will be used in whole experiment. MVC of respondents in this experiment are determined by identify the maximum load that can be lift up to angle 90 degree for 4 seconds.

After skin preparation like hair shaving and clean the skin by using alcohol pad, disposable electrode is placed on 1/3 of the length of upper arm from the fossa cubit. The second disposable electrode or reference electrode is placed 2cm center to center below the first electrode. The ground electrode will be placed in the bony area in upper arm. Respondents are required to wait for 3 minutes to stable the chemical reaction in disposable electrode.

Same circuit in figure 15 is used to run the experiment. The experiment will run from 90 degree to 0 degree. Once the red LED which indicator the start of reading process is light up, respondent start lift up the load to angle 90 degree in 2 seconds. Another green LED will light up after 2 seconds, starting the isometric contraction recording process. Weight is hold for 4 seconds. After 4 seconds, green LED will turn off, respondents start return the load to 0 degree within 4 seconds. The recording process is repeated for another 2 times for the same degree. There are 1 minute rest between each recording. The whole experiment is repeated from 90 degree to 0 degree with interval of 10 degree. Figure 19 shows the setup of equipment together with respondent while figure 20 shows the posture when carry isometric contraction on 70 degree.

Figure 19: Setup of equipment together with respondent

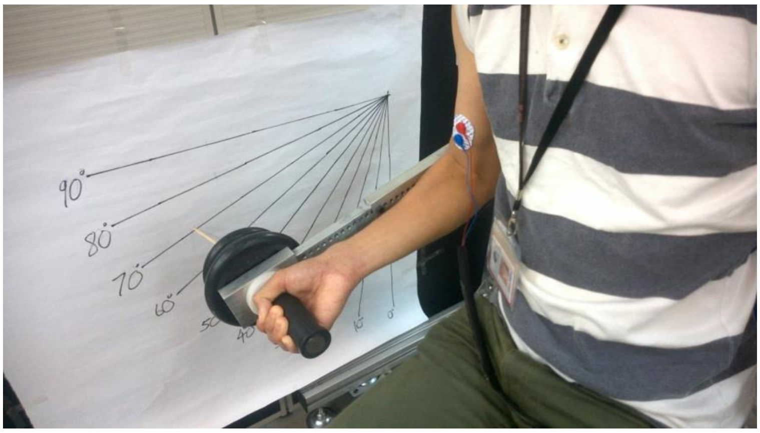


Figure 20: Isometric contraction on 70 degree

3.2.2.3. Experiment 3: Testing the performance and absolute error of designed Arduino program under non-fatigue condition.

An arduino program is designed based on the result on experiment 2. 4 respondents is selected to test the performance of the program. Exoskeleton mechanism is used to assist the respondents during the whole experiment. Same skin preparation and equipment setup in experiment 1 and 2 is used in this experiment except that LCD keypad shield is added in this experiment to show the achieved degree. Figure 21 shows the setup of circuit together with LCD.

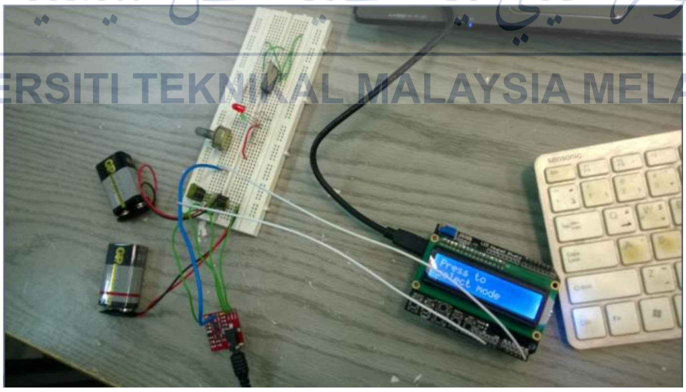


Figure 21: Setup of circuit with LCD

30% of MVC is used for all respondents. There are 3 mode in the program, calibration 1, calibration 2 and running mode. First of all, respondent is required go through calibration 1 which require them lift up the load to 90 degree, then the arduino will read the EMG reading for 3 seconds and save the average value. This process is repeated for 0, 20, 40, 60 and 80 degree. There are 30 seconds rest during the calibration 1 process. Then running mode is select to test the absolute error of program. Respondent is required to lift up the load from 0 degree to 10 degree, each degree is repeated for 3 times. The angle is repeated from 10 degree to 90 degree with interval of 10 degree. The resting time between each test in same degree is 30 seconds. 1 minute resting time is given when shifting the testing process to another degree. The LCD keypad shield is monitored for 4 seconds after the joint reaching the desired degree. The closest degree to the desired joint degree which showed in the LCD keypad shield is recorded. Among the 3 test, if there are 2 test which can't achieved the desired joint degree or have error, that joint degree is recorded down and respondent is required go through calibration 2 mode after the whole experiment. The process of calibration 2 is same like calibration 1 except that once the new EMG value is obtained and replace in the program, running mode is entered immediately to test that joint degree for 3 times with 30 seconds rest between each test. All result will be recorded in table.

3.2.2.4 Experiment 4: Identify and improve the error in DC motor with encoder

An arduino program is written to run the DC motor with encoder in 2 different speeds. 12V 2A adapter is used to power the DC motor. Higher running speed is applied when the difference between target position and located position is large. The DC motor and protractor is set up as showed in figure 22.

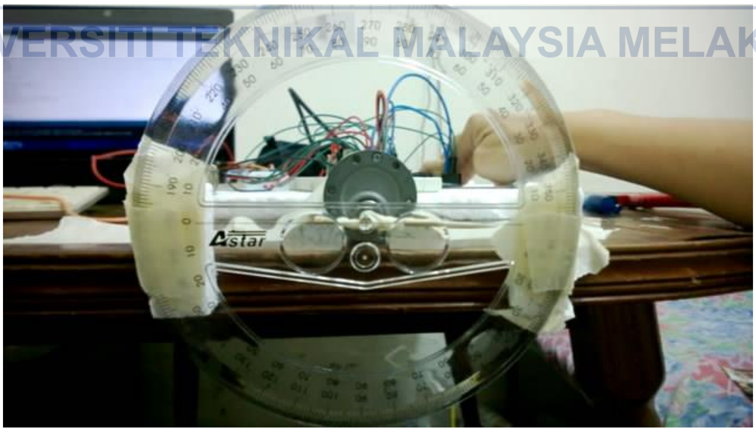


Figure 22: Setup of DC motor and protractor

The DC motor is test run, the encoder count is recorded and the notice the performance of DC motor. Calibration will be done on encoder count if there is error.

For the calibration process, run the motor for 20 degree. If there is position error at the steady state, increase or minus the percentage of pulse required by using trial and error method until it reaches the desired angle. The percentages of error are recorded in Table. The process is repeated for 40, 60 and 90 degree. The graph of percentages of error and angle are plot. The mathematic model is calculated using trendline function in Microsoft excel. The mathematic model is applied into the program. Number of pulse will be minus based on the percentage of error. Finally, run 3 tests from angle 0 degree to 10 degree, the error is recorded down. The angle is repeated from 10 degree to 90 degree with interval of 10 degree.

3.2.3 Development of prototype for validation

This section will discuss the item used in this research. It can divide into several parts: Sensor, processor, driver & actuator. The list of item is attached in Appendix A while the overall circuit is attached in Appendix B.

The electrode used in this project is Nihon Kohden, Vitrode L series G203 disposable electrode. This type of electrode is suitable used for adult, it has low impedance and 35 mm as the diameter. The gel contains in the electrode enable electrode solute and flow smoothly from skin to electrode. Figure 203 shows the figure of G203 disposable electrode.

Figure 23: G203 disposable electrode

The sensor used in this project is Muscle Sensor V3 Kit from Advancer Technologies[26]. Figure 24 shows the figure of the sensor and its circuit connection with voltage regulator in this project.

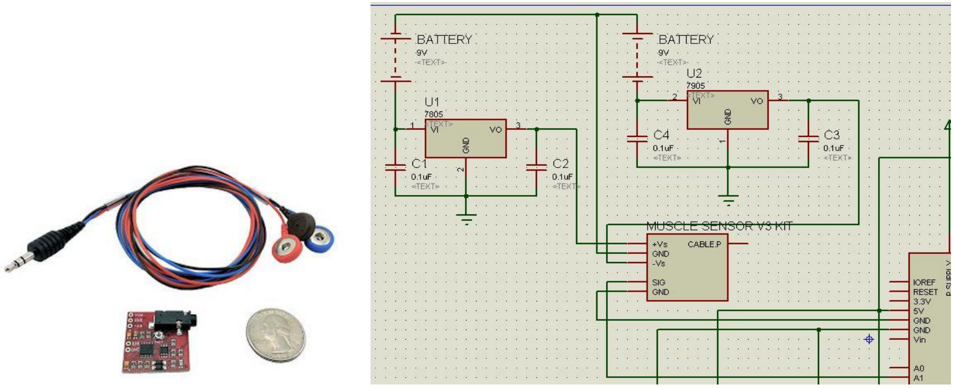


Figure 24: Muscle sensor v3 kit and its circuit connection

This sensor can measure our muscle raw EMG signal. Muscle sensor V3 kit has size about 1 inch x 1 inch. The advantage in size makes it suitable to be used in robot, medical device equipment and powered exoskeleton suits. It is equipped with adjustable gain, so by adjusting the gain, it can be used for anybody who has different amplitude in their EMG signal. However this device should be handle carefully as it is a electrostatics damage product. It is designed to be used with microcontroller, hence this sensor includes amplifier, filter (range around 2 Hz to 110 Hz) and full wave rectification to produce a smooth rectified signal that can be analysed by microcontroller. The amplitude of the output voltage depends on the activity of the muscle, the maximum limit of the output voltage is equal to the voltage supply to the device. The output of Figure 25 shows the example of output signal coming out from this sensor.

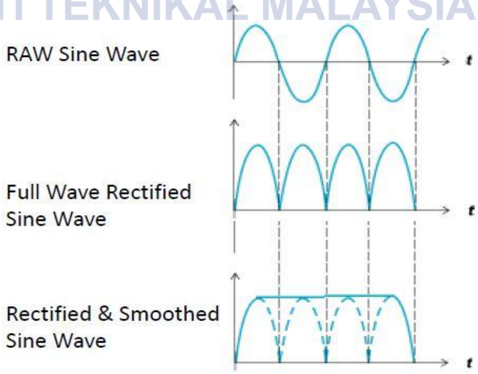


Figure 25: Example of output signal from sensor

As showed in figure 24, there are 5 pins in the sensor: 2 ground pins, +v pin, -v pin and signal pin. 1 ground pin is connected to the ground of power supply, 1 ground pin is connected to the ground of arduino board, +v and –v pin is the positive and negative voltage supply to the sensor. Based on the datasheet, even though the device has typical supply voltage of +/- 5V and can support input voltage up to 30V, but there is a case where the device burn by supply it with +/- 9V, furthermore the signal will be analysed in arduino board by using analog to digital (ADC) function which the voltage limit is 5V, hence it is better supply +/- 5V to the sensor.

To extract 5V from a 9V battery, 7805T 5V 1A positive regulator and 7809T 5V 1A negative regulator is used. Figure 26 shows the Figure of voltage regulator 7805T and 7809T.

Figure 26: Voltage regulator

Based on the datasheet, capacitor should connect between the input & ground pins and output & ground pins as showed in figure 24[27]. The function of the capacitor is to filter the noise coming out from the battery and the noise exists in DC voltage line. It is important to use a capacitor produce a precise voltage as it is connected to the Muscle Sensor V3 kit which contains a lot of sensitive IC chip inside it. In this project, the capacitor that going to use is 0.1uF ceramic capacitor.

The processor unit in this project is Arduino Uno board. Figure 24 shows the picture of Arduino Uno board

Figure 27: Arduino Uno board

Arduino Uno is a microcontroller which uses ATmega328[28]. Arduino Programming Language and Arduino Pprogramming Environment is used to program the microcontroller. ATmega328 has 6 analog pins and 14 digital pins. Among the 14 digital pins, 6 of them can be used for pulse width modulation (PWM) function. Arduino Uno can either get its supply from power adapter or USB port which connected to laptop.

LCD is used in this project to easier the analyse process, hence LCD keypad shield is used in this project. Figure 28 shows the figure of LCD keypad shield.

Figure 28: LCD keypad shield

LCD keypad shield is suitable used with Arduino board[29]. It can be direct stack above the Arduino board without any extra wire connection. LCD keypad shield consist of 2 row and 16 column display with an adjustable brightness switch. It gets its 5V power supply from the Arduino board.

Driver is used to control the speed and direction of motor rotation. It can used to supply large amount of current to the motor without interface with microcontroller because microcontroller can't handle high current. The driver used in this project is L298. Figure 29 shows the figure of driver L298.

Figure 29: Driver L298

L298 has 2 H-bridge with each bridge can support up to 2 ampere current[30]. It can support voltage up to 46 volt. Pin 1 and Pin 15 in L298 is the current sensing pin. A high power small resistance resistor which called sense resistor can be connected as showed in figure 30. User can measured the current supplied to the load by measuring the voltage drop across the sense resistor. The value of voltage can further feedback to microcontroller by using ADC to control the driver.

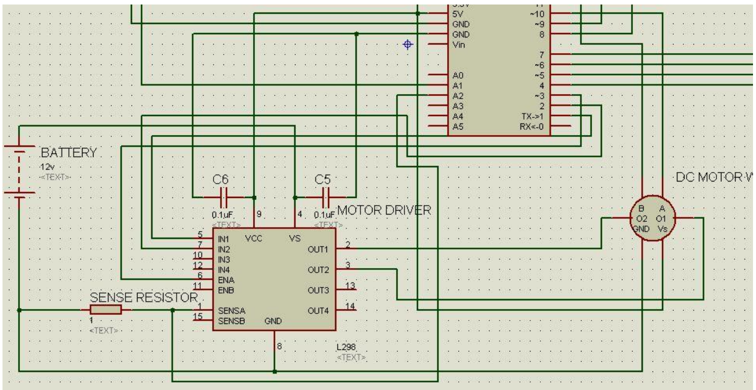


Figure 30: Driver circuit

If the driver is used to handle load with large current, 2 H-bridge can be use simultaneously to supply the current, so the current will distribute into 2 without too stressing one of the bridge. For this experiment, 1 bridge is used because the current of the load is not big.

The actuator used in this project is DC gear motor with encoder, SPGE30E-300K from Cytron[31]. The gear ratio for this motor is 1:300. It has 12 rotations per minute and 1.176 Nm as the torque. The encoder equipped on this motor has the resolution of 3 pulses per revolution and 900 pulses per main shaft revolution. The encoder can be act as the feedback by identify the direction of rotation and also the angle position of the angle. The figure of motor and the sequence of encoder A and B is showed in figure 31.

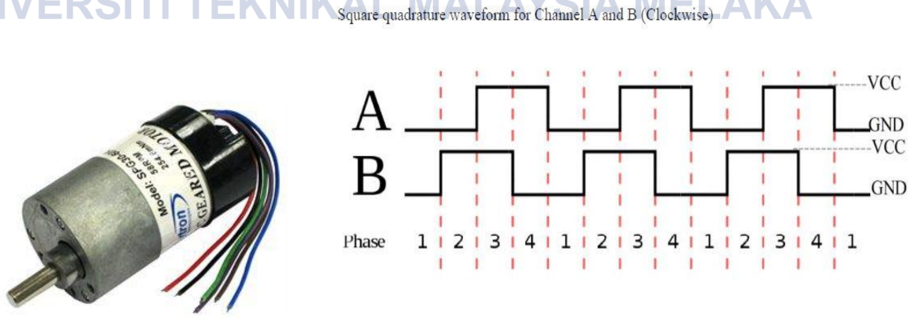


Figure 31: DC gear motor and encoder state

3.2.4 Method of Analysis

This section describes how to analysis the data get from the experiment.

For experiment 1, the data between 2-6 seconds which represent the isometric contraction of the experiment will be focused. The input signal will go through moving average and RMS feature extraction. The average value between moving RMS & Moving average vs joint angle will be plotted. Based on the graph, mathematic model will be produced by using trendline function in Microsoft Excel. This process will be repeated for the reading that involved 30% MVC load. The result of EMG with load and without load will be compared. In real application, the EMG signal will fluctuate due its natural characteristics or to the noise, hence the range of EMG value between each joint angle will be obtained and analyse. The calculation of mean, standard deviation and range is showed in below

$$\text{Mean} = \frac{\sum_{i=0}^{N-1}|x(i)|}{N} \tag{3.1}$$

$$\text{Standard deviation} = \sqrt{\frac{1}{N}\sum_{i=0}^{N-1}[x(i) - mean]} \tag{3.2}$$

$$\text{Upper range} = \text{mean} + \text{standard deviation} \tag{3.3}$$

$$\text{Lower range} = \text{mean} - \text{standard deviation} \tag{3.4}$$

For experiment 2, the data in isometric contraction will be focused. The range of EMG and its mean value for every degree will be obtained and compares with every respondent. A mathematic model will be calculated by using trendline function in Microsoft excel and the relationship between angle and EMG value will be analysed to find the similarities characteristics for all respondents.

For experiment 3, the absolute error of the program for calibration 1 and calibration 2 will be analysis. The absolute error is calculated by:

Absolute error = | Obtained degree – desired degree |

The factor that effect the error of the program will be analysis based on the measured EMG value in calibration 1 and calibration 2.

For experiment 4, the performance of motor actuator is evaluated. The mathematic model which relates the percentage of encoder error and rotation angle will be analysed to determine the relationship between encoder error and rotation angle. Another graph of encoder pulse number vs time will be plotted to determine the setting time and the percentage of overshoot.

Chapter 4

RESULT, ANALYSIS AND DISCUSSIONS

4.1 Experiment 1: The relationship between EMG signal and elbow joint angle with load and without load.

The program flow chart and coding for experiment 1 and 2 is attached in appendix C. All data such as MVC or length from medial acromion to fossa cubit Isotonic for all respondents involves in this thesis is attached in appendix as appendix D. Figure 32 shows 1 of the moving average (MA) and RMS features for unloaded 40 degree of respondent A. The reading of all EMG vs time (unloaded and loaded) for all angle are attached in the CD under the name of "result experiment 1". In medical field, Isotonic means the changing of the length of the muscle (moving) while isometric means the length of the muscle doesn't change (fix).

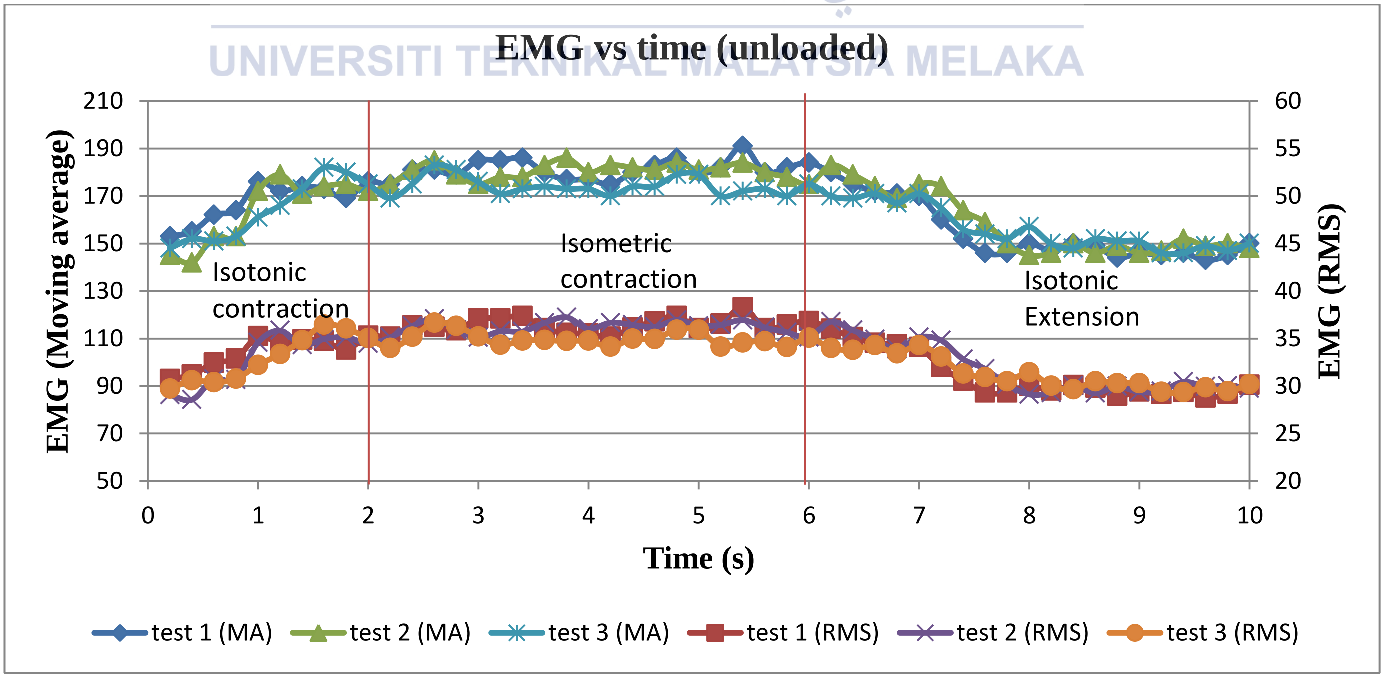


Figure 32: Moving average and RMS vs time for unloaded 40 degree

When the arm is lift up or isotonic contraction (0-2 seconds), torque is produced in the muscle, hence the moving average and RMS value will start to increase. When the respondent holds his arm or isometric contraction (2-6 seconds), the torque remains constant in the muscle, the moving average and RMS value will be near to a straight line. After 6 seconds, respondent return his arm to normal position or isotonic extension, the torque produced by the muscle stat to decrease, hence the moving average and RMS value decreases.

By comparing the result of moving average and RMS feature, both feature shared the same shape, the only difference is just the amplitude level. The different of amplitude between moving average and modified RMS is by factor 5. Since the amplitude of EMG has been divided by 5 before entering modified RMS feature extraction, if we multiply the modified RMS value by value of 5, we will get the value almost same with moving average value. For example, by referring figure 29, test 3, the moving average and modified RMS value is 176 and 35.22. If we multiply 35.22 by 5, we will get 176.1 which are almost same with moving average value. The graph about moving average & modified RMS vs time under unloaded and load condition for different degree is attached in the CD under name of “experiment 1”.

Figure 33 and 34 shows the moving average and modified RMS value for unloaded condition under different angle while figure 35 and figure 36 shows the moving average and modified RMS value for loaded condition under different angle. Calculation of mean and standard deviation is focus on the isometric contraction (2 to 6 seconds) only.

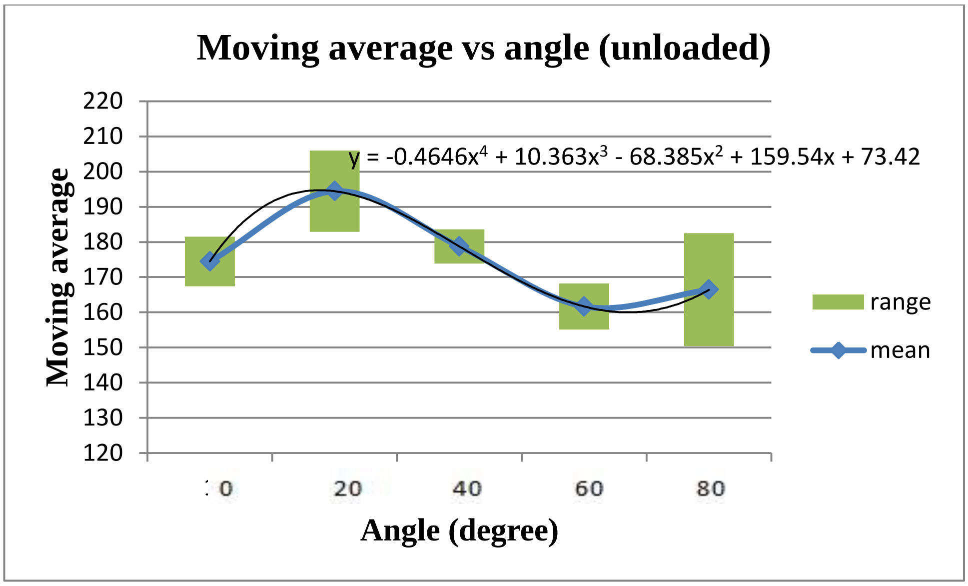


Figure 33: Moving average vs angle for unloaded condition

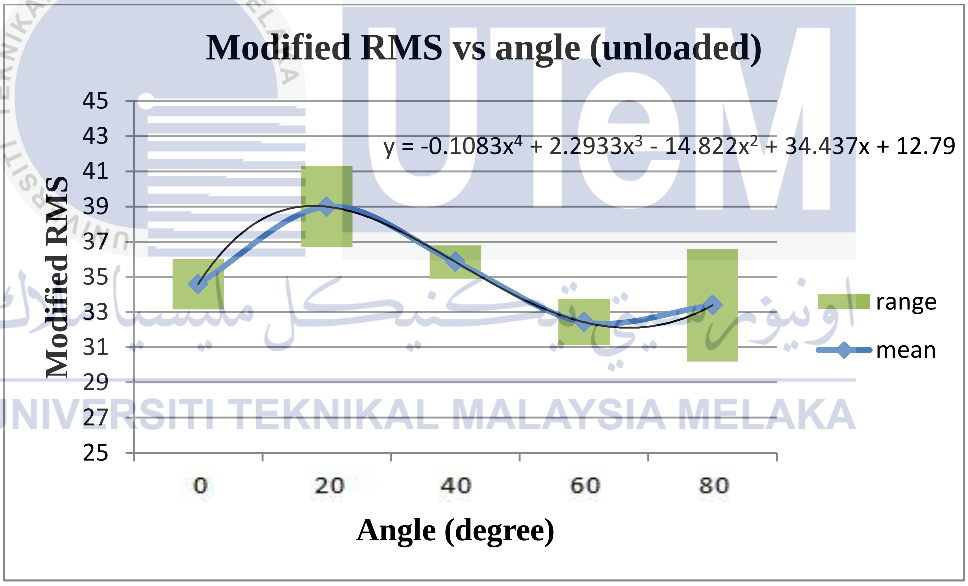


Figure 34: Modified RMS vs angle for unloaded condition

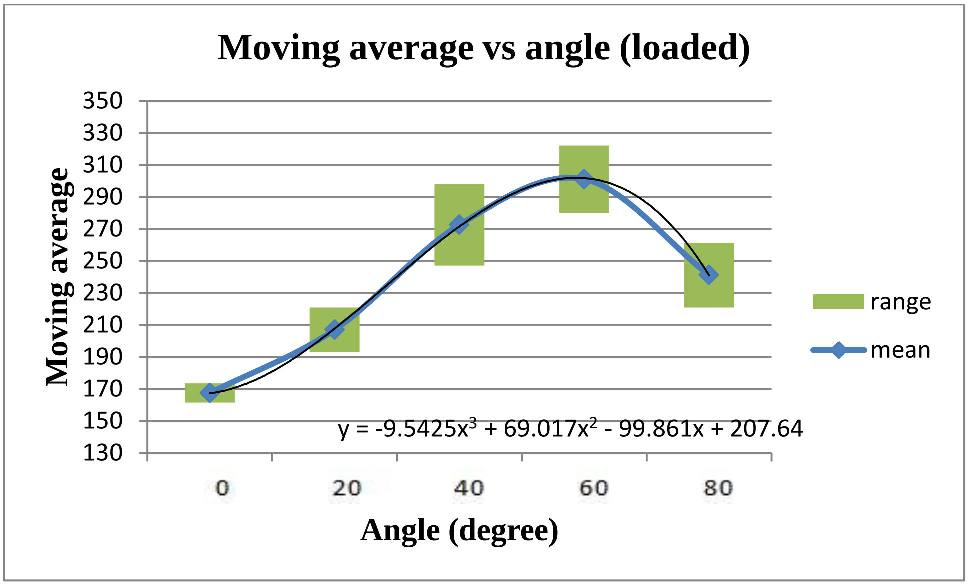


Figure 35: Moving average vs angle for loaded condition

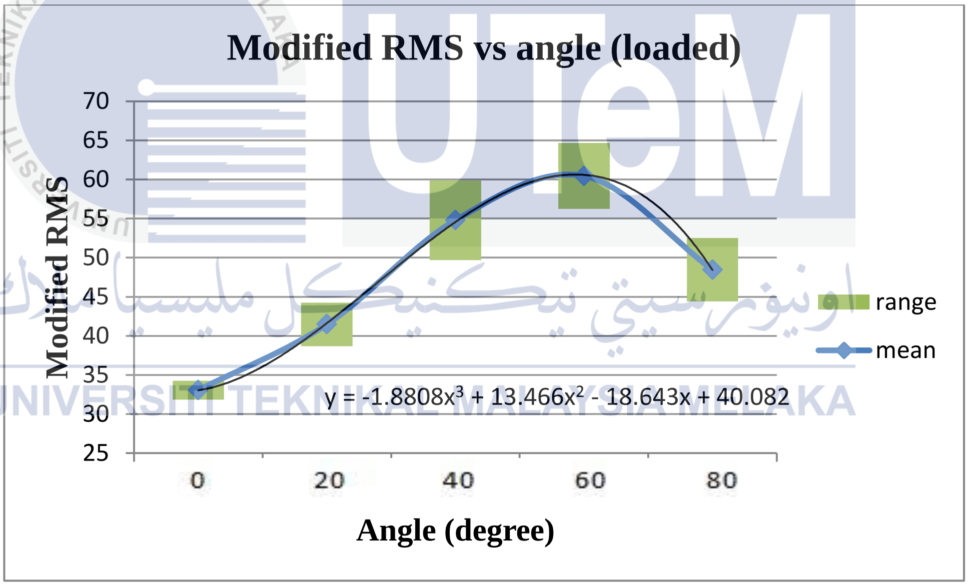


Figure 36: Modified RMS vs angle for loaded condition

The mathematic model for the mean value of moving average and modified RMS in figure 33 and 34 shows that the both mathematic model can be used to represent the relationship between EMG value and angle by using polynomial equation with power of 3. Both models are almost close to each other if the mathematic model in figure 31 is multiply by 5, however they still sharing the same shape, the only different is the amplitude level. Since the moving average and modified RMS almost share the same characteristics and shape, so moving average is selected as the classification features in the following experiment 2 and 3 because the processing time for moving average is shortest than modified RMS.

By comparing figure 33 and figure 35 which represent the situation for unloaded and loaded condition, we can see that the range of EMG value in figure 33 is not uniform. Its value stop increasing and start decreasing at 20 degree and start increasing back at 60 degree. This shape is conflict with theory describe in chapter 2. For figure 35 which stands for loaded condition, the EMG value stop increasing and start decreasing at 60 degree. This kind of shape is reasonable as the increasing of angle in elbow has activate anterior deltoid muscle, there are force shifting from biceps brachii to anterior deltoid, hence the torque exerted in biceps brachii is decreases, so the EMG value will decreases. The results also show that the effect of weight has increases the performance of sensor and the result. The additional weight increases the torque applied and hence increases the EMG amplitude level in the muscle. The increasing of signal amplitude increases the signal to noise ration (S/N) in the sensor. Hence it is suggested that 30% of MVC weight should be hold by respondent during the following EMG recording process in experiment 2 and 3.

4.2 Experiment 2: The relationship between EMG signal and joint angle for 4 respondents

Figure 37, 38, 39 and 40 shows moving average vs angle for respondent A, B, C and D. A detail reading for all the result in experiment 2 is attached in the CD under the name of “result experiment 2”. The moving average and range value are obtained and calculated only for the isometric contraction (2 to 4 seconds).

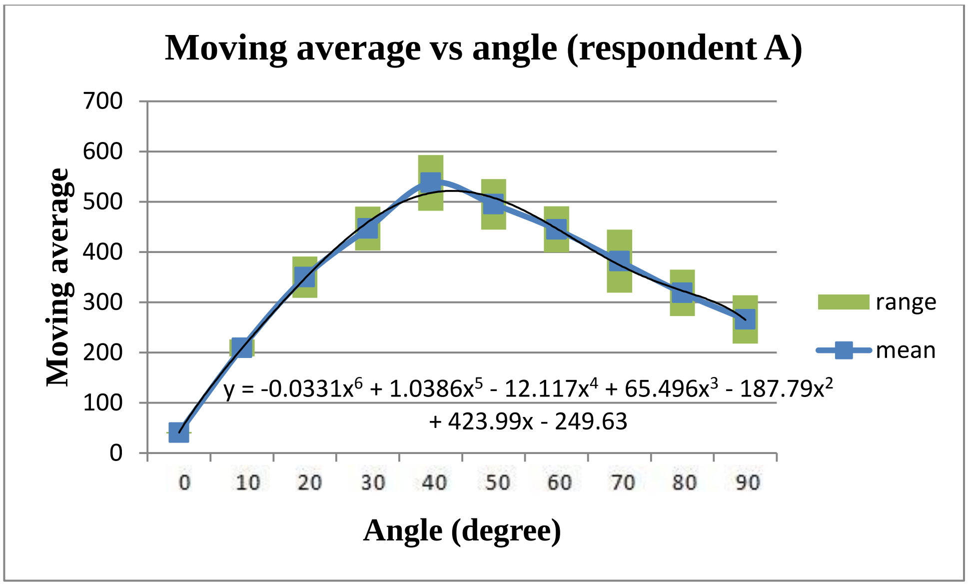


Figure 37: Moving average vs angle for respondent A

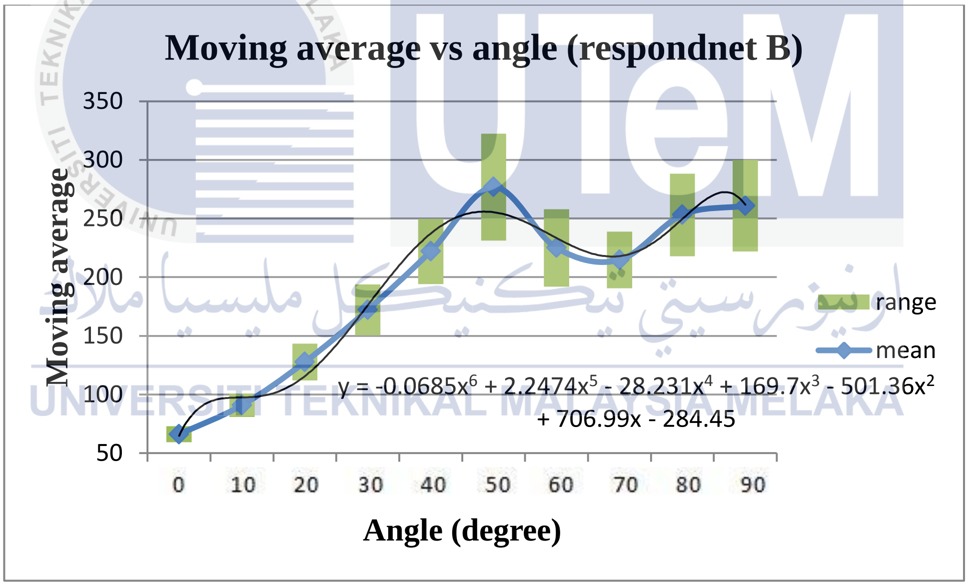


Figure 38: Moving average vs angle for respondent B

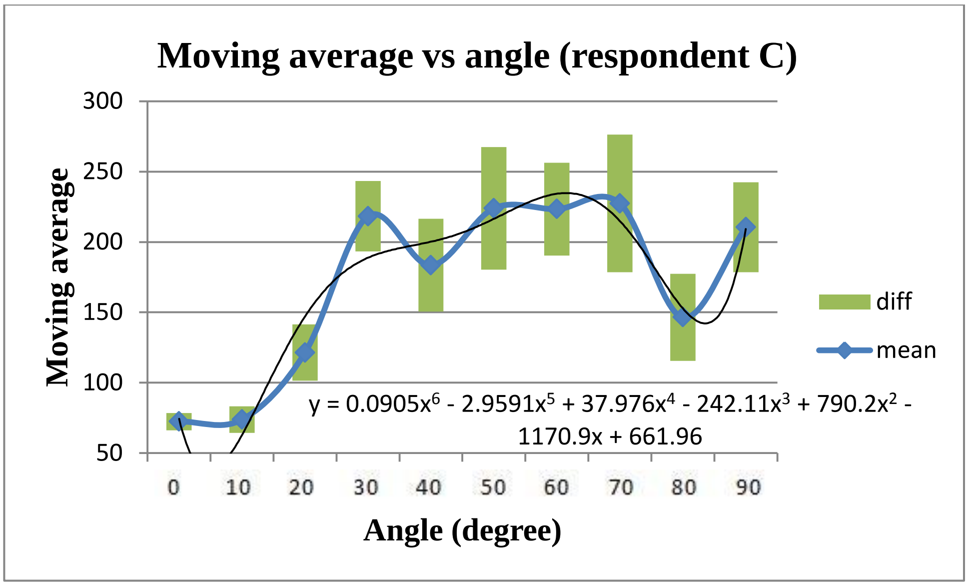


Figure 39: Moving average vs angle for respondent C

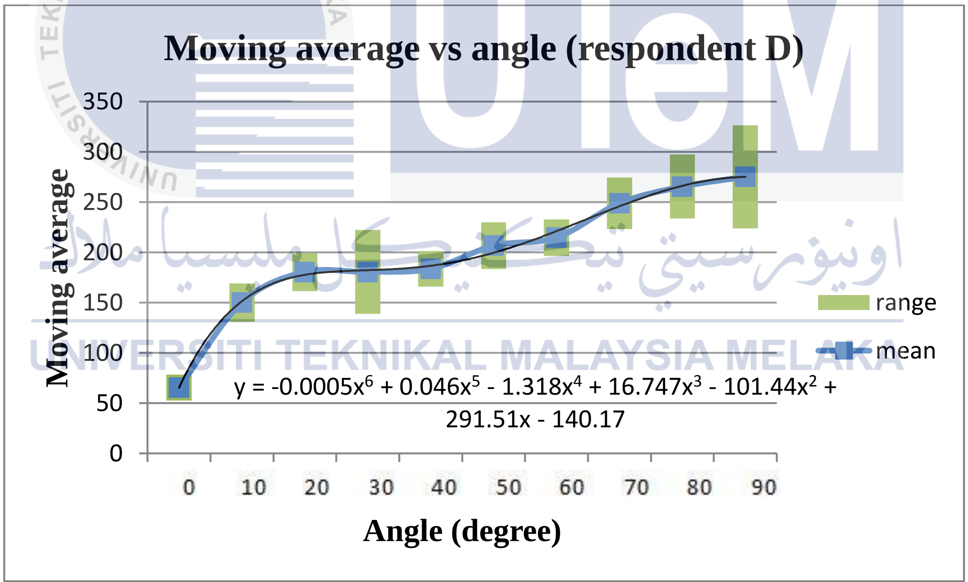


Figure 40: Moving average vs angle for respondent D

Based on the figure 37 to figure 40, we can observe that only respondent A and D showed reasonable result. For respondent B, the overall result from 0 to 70 degree are reasonable, however the mean value for 80 and 90 degree is higher than what we expected. Based on the theory, the mean value for 80 and 90 degree should be lower than the mean value for 70 degree. There should only exist 1 turning point on the shape of the graph due to shifting of force from bicep brachii to anterior deltoid. However this shifting process is affected by the length of the biceps muscle. Based on the figure, only respondent A and B has happened force shifting between biceps brachii and anterior deltoid. Respondent D doesn't have a turning point in her graph as the force shifting process doesn't occur. Hence it is assumed that the error for 80 and 90 degree of respondent B is caused by the human error in controlling the muscle. It requires some skill to control the muscle, even though our elbow reaches the desired angle, but we still can exert force from biceps without moving the elbow. The error happened in 80 and 90 degree because this 2 degree is the first and second EMG recording in experiment 2. Respondent B may not good in controlling the muscle at the beginning, but the situation improved on the following recording. For respondent C, the large number of turning point in figure 36 shows that she is not good in controlling her muscle. Respondent A has the best shape as showed in figure 34 because he has go through and practice the EMG recording many times and has experience on how to control the muscle.

If we refer to figure 37 to 40, we may observe that most of the moving average value for a degree is located in the middle of the degree below and above it. For example in figure 37, the mean value for 30 degree is 446.71. It is located in the middle of 20 degree and 40 degree which are 350.05 and 537.55. The formula is showed in below.

$$MA_n = (MA_{n-10} + MA_{n+10})/2 \qquad (4.1)$$

MA = moving average value

n = angle

Table 2, 3, 4 and 5 and figure 41 to figure 44 show the actual and calculated value for moving average based on equation 4.1 for all respondent. Only odd degree of angle is selected to perform the calculation.

Table 2: Actual and calculated moving average value for respondent A

Angle (degree)	**Actual value**	**Calculated value**	**Accuracy (%)**
0	40.68		
10	209.02	192.37	93.47
20	350.05		
30	446.71	443.8	99.35
40	537.55		
50	494.95	491.26	99.25
60	444.97		
70	381.79	381.87	97.9
80	318.77		
90	266.02		
Overall			97.49

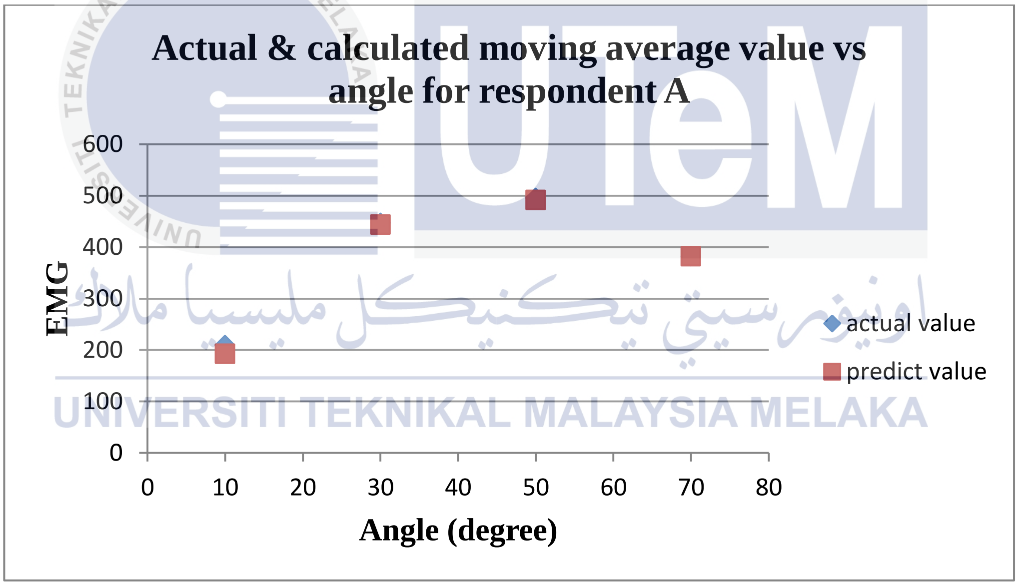


Figure 41: Actual and calculated moving average value for respondent A

Table 3: Actual and calculated moving average value for respondent B

Angle (degree)	**Actual value**	**Calculated value**	**Accuracy (%)**
0	66.09		
10	90.87	96.88	93.39
20	127.67		
30	172.17	174.82	98.46
40	221.97		
50	276.8	223.47	80.73
60	224.97		
70	214.75	239.07	88.68
80	253.17		
90	260.88		
Overall			90.315

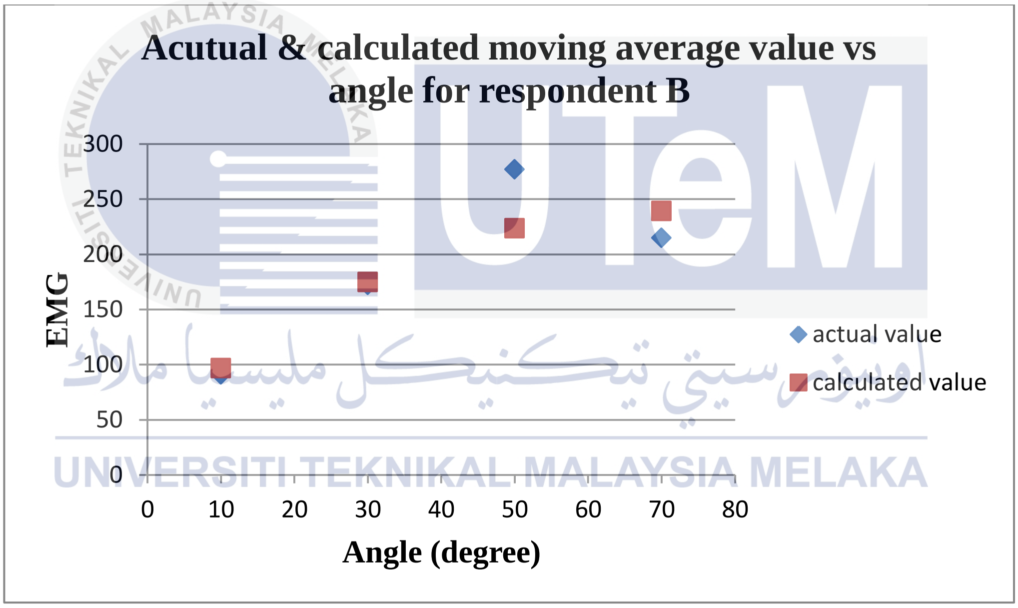


Figure 42: Actual and calculated moving average value for respondent B

Table 4: Actual and calculated moving average value for respondent C

Angle (degree)	Actual value	Calculated value	Accuracy (%)
0	72.41		
10	73.82	96.88	68.76
20	121.35		
30	218.28	152.4	69.82
40	183.44		
50	223.88	203.39	90.85
60	223.33		
70	227.34	184.9	81.33
80	146.46		
90	210.38		
Overall			77.69

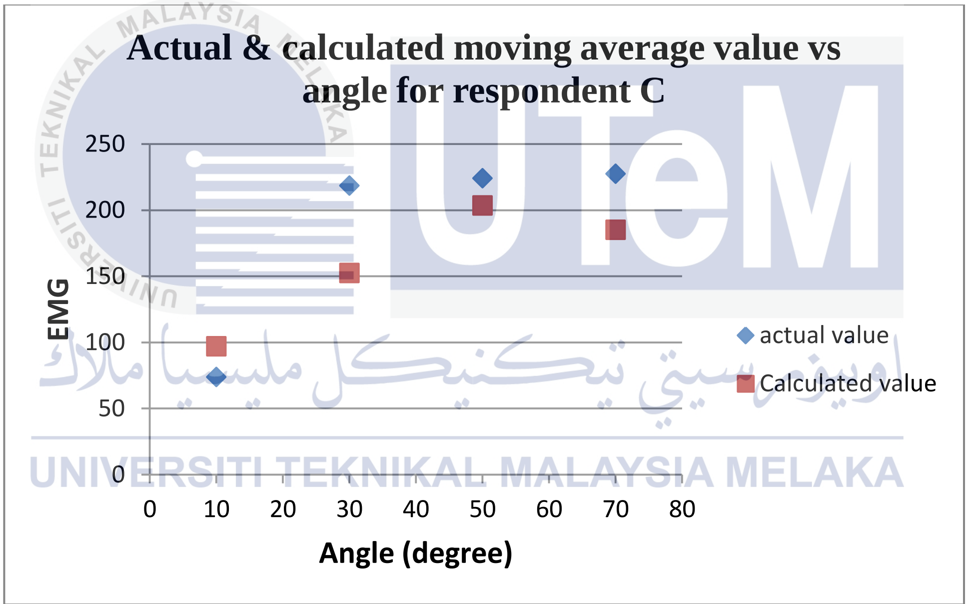


Figure 43: Actual and calculated moving average value for respondent C

Table 5: Actual and calculated moving average value for respondent D

Angle	Actual value	Predict value	Accuracy (%)
0	65.61		
10	150.12	123.02	81.95
20	180.43		
30	180.53	182.02	99.17
40	183.6		
50	206.67	199.16	96.37
60	214.72		
70	248.83	240.1	96.46
80	265.48		
90	275.12		
Overall			93.49

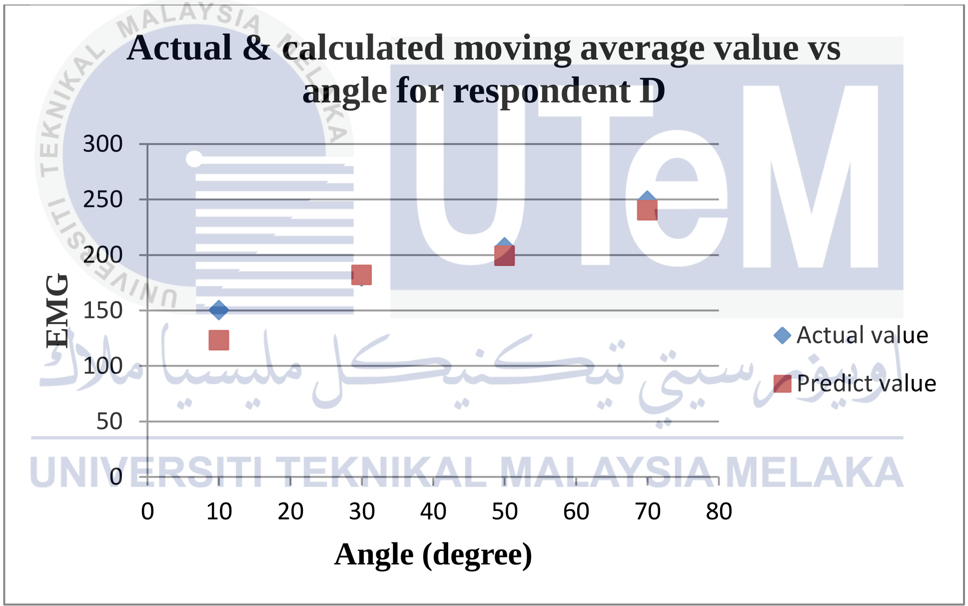


Figure 44: Actual and calculated moving average value for respondent D

Based on the overall result from table 2 to table 5, we can see that the best result which has the highest accuracy is from respondent A, table 2, 30 degree, the calculated value is 443.8 while the actual value is 446.72, the accuracy is 99.35%. As his moving average value vs angle in figure 37 has the smooth line, hence it is easier to predict and calculate other moving average value based on the formula. The lowest accuracy happened in table 4, respondent C, 10 degree, the accuracy is only 68.76%. This situation happened in respondent C because her data is chaos, so it is hard to predict any moving average value based on calculation. For the overall performance, respondent A has the highest accuracy which is 97.49% while respondent C has the lowest accuracy which is 77.69%. The result from table has proved that the moving average value can be calculated and predicted based on the formula.

Figure 42 show the graph of moving average vs angle for all respondents.

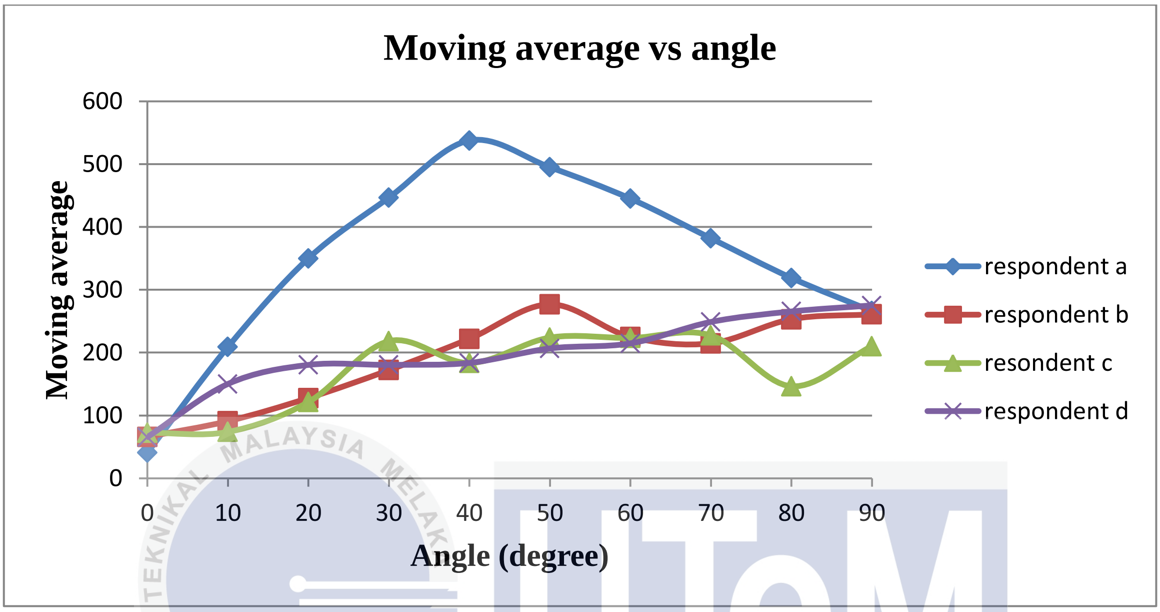


Figure 45: Moving average vs angle for all respondent

From figure 45, it been noticed that there exists 2 type of shape, one is like shape of respondent A which has a turning point while the another one has the shape of respondent D which there are no turning point on the graph. The mathematic model for all the respondents are listed in below.

Respondent A: $y= -0.00331x^6+1.0386x^5-12.117x^4+65.496x^3-187.79x^2+423.99x-249.63$

Respondent B: $y= -0.0685x^6+2.2474x^5-28.231x^4+169.7x^3-501.36x^2+706.99x-284.49$

Respondent C: $y=0.0905x^6-2.9591x^5+37.976x^4-242.11x^3+790.2x^2-1170.9x+661.96$

Respondent D: $y=0.0288x^6+0.046x^5-1.318x^4+16.74x^3-101.44x^2+291.51x-140.17$

The mathematic model for respondent C is ignored due to the result is conflict with the theory and considered as fail due to bad in muscle control. The shape owned by respondent B is considered same type with the shape of respondent A even though the last 2 degree of respondent B is not accurate. By comparing the mathematic model between respondent A and respondent B, it is noticed that slightly change in the EMG reading can cause big changing in the mathematic model. Other than that, there are no similar characteristics between the moving average of angle for every respondent. For example, the maximum point for respondent a fall in 40 degree while respondent b falls in 50 degree. Other than that, the maximum point is determined by the trigger of anterior deltoid muscle. The trigger of this muscle is different for every people because it is effect by the length of forearm and human movement behaviour. The different of gradient for the line of each respondent also mark that the ratio between every respondent in every angle is different. As the conclusion, mathematic model is suitable to be used in this project.

4.3 Experiment 3: Testing the performance and absolute error of Arduino program under non-fatigue condition

Based on the result from experiment 2, a arduino program has been produced to classify the angle. The flow chart and coding of program is attached in appendix E. To enable the program suitable to be used for every user, hence there are no pre-installed moving average value inside the program. User need to calibrate and save the moving average value into the program before using it to control a DC motor. This program contains 3 modes. There are calibration 1, calibration 2 and running mode. Calibration 1 mode is used to calibrate and save the moving average value inside the program. Calibration in this mode is only allowed for even degrees like 0, 20, 40, 60,80 degree and 90 degree. The moving average value for odd degree like 10, 30, 50 and 70 degree are calculated based on the equation 4.1. During the calibration, user needs to hole the weight in isometric contraction at desired degree for 3 seconds, then the moving average value will showed up on the LCD screen and save inside the program. Figure 46 shows the screen of LCD when calibrate 20 degree in calibration 1 mode.

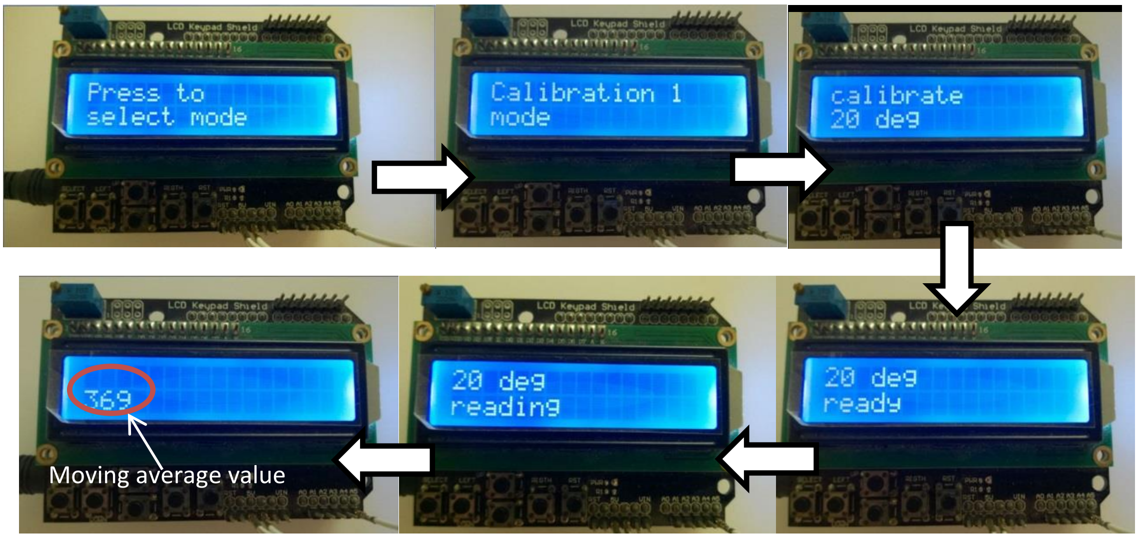


Figure 46: Screen of LCD when calibrate 20 degree in calibration 1 mode

Calibration 2 mode is used for correction. After go through calibration 1, moving average for odd degree will be calculated by program. Since the moving average is come from calculation, it is not 100% accurate compare to real time reading from the sensor, hence the function of calibration 2 is used to correct and replace any inaccurate moving average value that calculated by program. It is operating same like figure 43, however the obtained moving average value is directly replace the previous saved value without go thorough any calculation. The angle for calibration 2 is allowed for every degree. Other than that, calibration 2 also can be used to update the moving average value due to fatigue issues.

User can view the saved moving average by using LCD keypad shield. Figure 47 shows the screen of LCD when viewing the saved moving average.

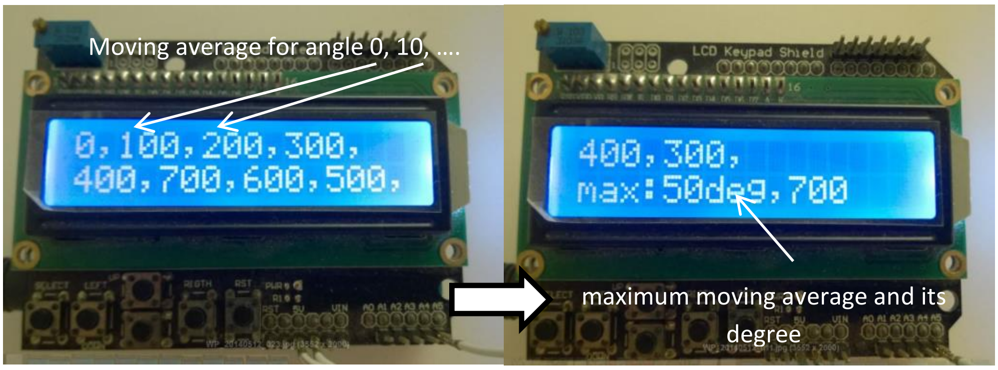


Figure 47: Screen of LCD when viewing the saved moving average

Based on the result from experiment 2, we know that there are 2 type of graph may exist for the moving average vs angle. By viewing the saved moving average data for each degree, we can classify what type of graph we have. Figure 48 showed the graph of moving average vs angle with a turning point.

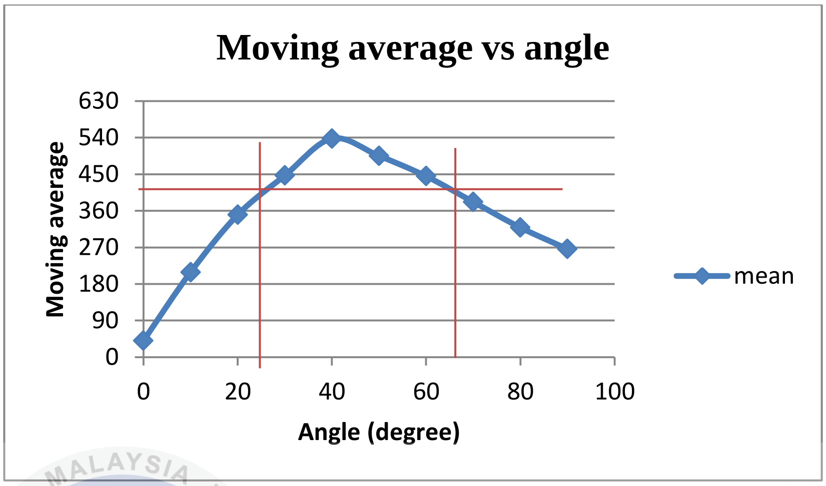


Figure 48: Graph of moving average vs angle with a turning point

As we can see on figure 48, after the moving average pass the maximum point and its value start to drop until 400, there are 2 point which can be represent by moving average equal to 400, one is near the 30 degree and the other one is 70 degree. To prevent the confusion of program or the output of program change rapidly between 30 and 70 degree, the program will only achieved 70 degree after its moving average has been classify as maximum degree at least once, this means that the program has to capture a moving average that belong to the degree that posses the maximum moving average before it can classify the degree over the maximum moving average. The program will classify the degree at the right hand side of maximum point and can't classify the degree at the left hand side of maximum point unless the moving average get by the program is near to the 0 degree, then the classification system will be reset.

In running mode, the program classify the degree by read the moving average signal from arduino and compare it with the saved moving average data to find the nearest value that represent certain degree. Figure 49 shows the screen of LCD when in running mode.

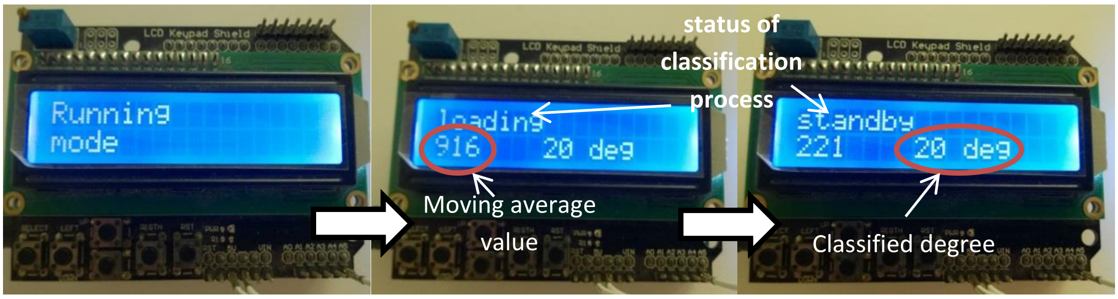


Figure 49: Screen of LCD when in running mode

To stabilize the performance of program, if the program can't classify the same degree for continuous equal or more than 3 times, "loading" will print out on the screen, the previous classify degree at the right of screen won't update. If the program can classify the moving average in the same degree for more than 3 times, "standby" will print out on the screen, the classify degree at the right will update. In the program, it took 0.1 seconds to read the EMG signal and come out with a moving average value, hence to come out with a degree output, it need at least 0.3 seconds (doesn't add up with processing time).

Table 6 to table 9 shows the performance of the program in term of absolute error for all respondents while figure 50 to figure 53 show the moving average vs angle and average absolute error for all respondents. In this experiment, respondent B and C is replaced with respondent E and F.

Table 6: Performance of the program for respondent A

Calibration 1								
angle	Moving average	test (angle) test 1	absolute error	test 2	absolute error	test 3	absolute error	average absolute error
0	19	0	0	0	0	0	0	0
10	161	10	0	10	0	10	0	0
20	303	20	0	20	0	30	10	3.33
30	381	40	10	30	0	30	0	3.33
40	460	40	0	40	0	40	0	0
50	396	30	20	50	0	50	0	6.67
60	333	60	0	30	30	30	30	20
70	310	20	50	20	50	30	40	46.67
80	287	80	0	50	30	20	60	30
90	193	90	0	80	10	80	10	6.67
Calibration 2								
angle	Moving average	test (angle) test 1	absolute error	test 2	absolute error	test 3	absolute error	average absolute error
60	304	50	10	70	10	80	20	6.67
70	345	20	50	80	10	60	10	23.33
80	328	20	60	80	0	70	10	23.33
90	252	90	0	50	40	70	20	20

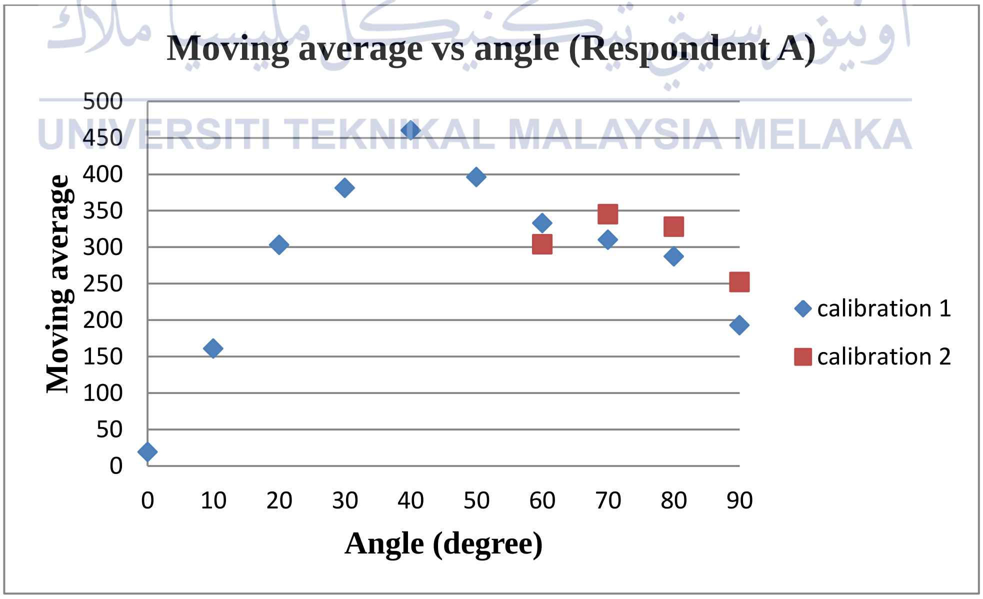


Figure 50: Moving average vs angle for respondent A

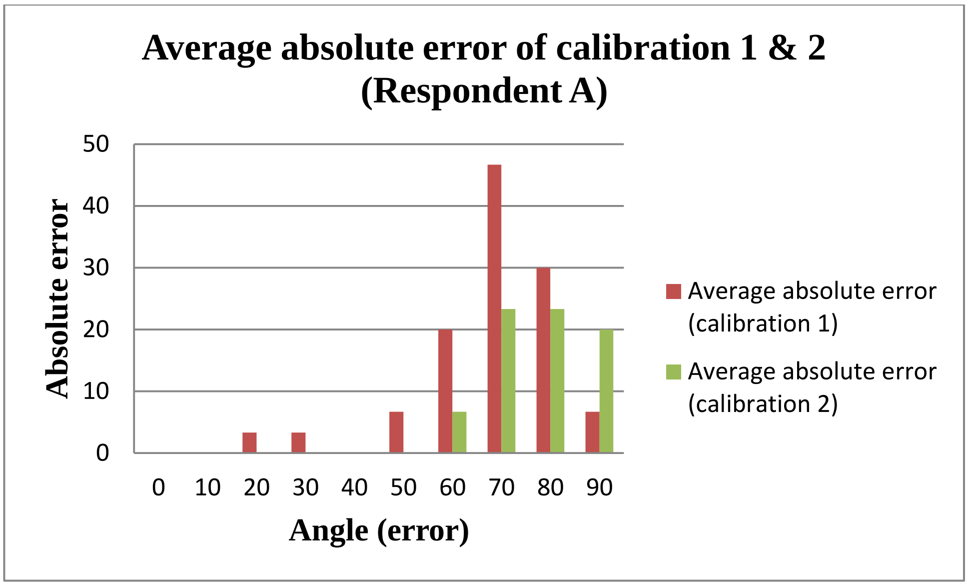


Figure 51: Average absolute error of calibration 1 & 2 for respondent A

Table 7: Performance of program for respondent D

Calibration 1								
angle	Moving average	test (angle) test 1	absolute error	test 2	absolute error	test 3	absolute error	average absolute error
0	1	0	0	0	0	0	0	0
10	60	10	0	10	0	10	0	0
20	119	20	0	20	0	20	0	0
30	164	20	10	20	10	30	0	6.67
40	209	30	10	30	10	30	10	10
50	218	60	10	60	10	60	10	10
60	228	60	0	60	0	60	0	0
70	210	70	0	60	10	60	10	6.67
80	192	90	10	80	0	90	10	6.67
90	179	90	0	90	0	90	0	0
Calibration 2								
angle	Moving average	test (angle) test 1	absolute error	test 2	absolute error	test 3	absolute error	average absolute error
30	160	20	10	30	0	20	10	6.67
40	234	30	10	40	0	40	0	3.33
50	254	50	0	60	10	60	10	6.67
70	266	60	10	70	0	60	10	6.67
80	269	70	10	60	20	70	10	13.33

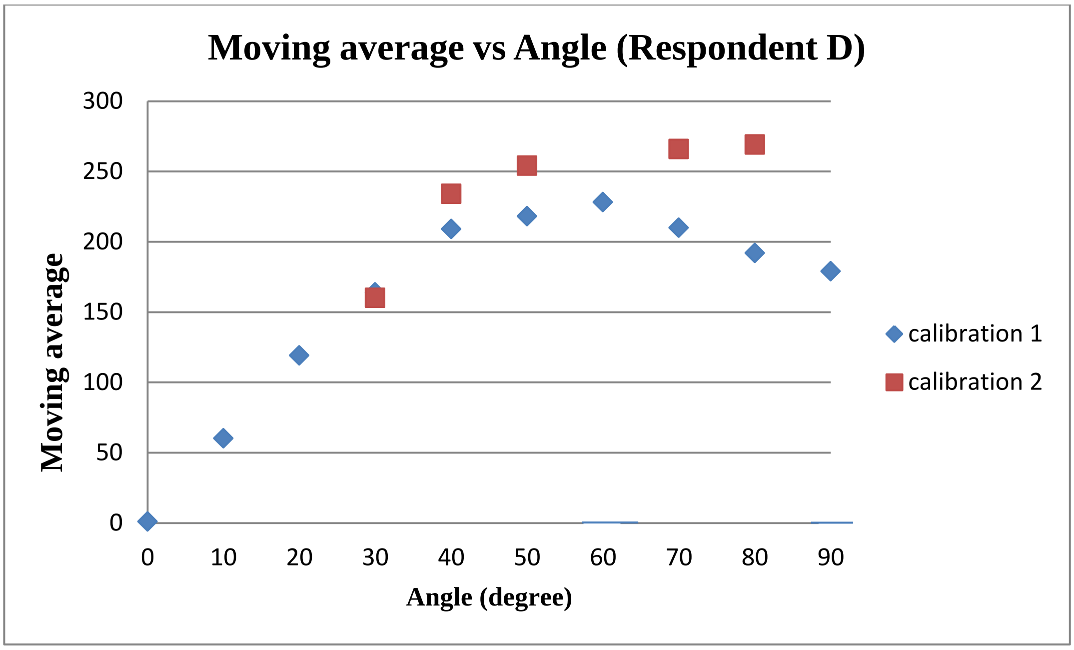


Figure 52: Moving average vs angle for respondent D

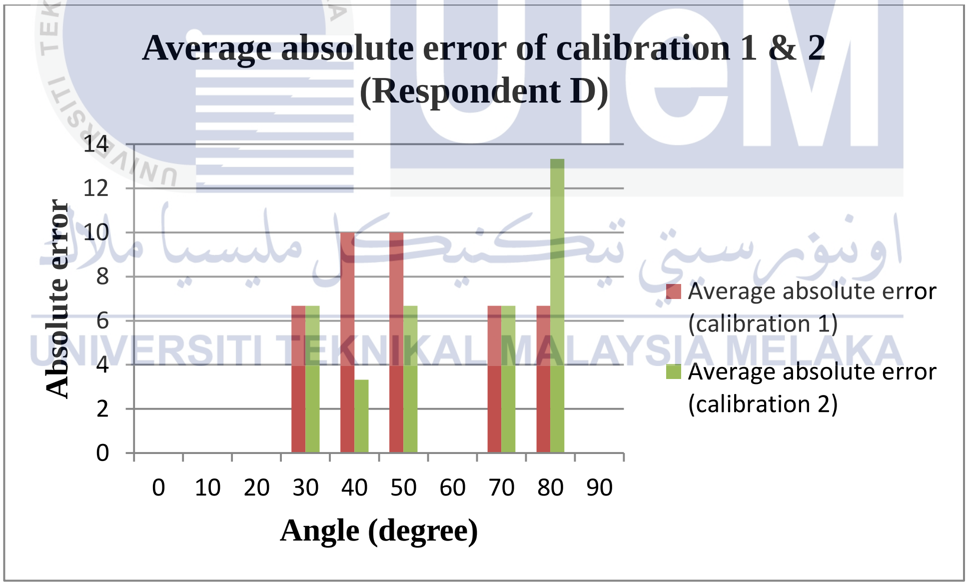


Figure 53: Average absolute error of calibration 1 & 2 for respondent D

Table 8: Performance of program for respondent E

Calibration 1								
angle	Moving average	test (angle)						average absolute error
		test 1	absolute error	test 2	absolute error	test 3	absolute error	
0	16	0	0	0	0	0	0	0
10	64	10	0	10	0	10	0	0
20	112	20	0	20	0	20	0	0
30	138	20	10	20	10	30	0	6.67
40	164	30	10	30	10	30	10	10
50	166	40	10	30	20	90	40	23.33
60	168	80	20	90	30	80	20	23.33
70	195	70	0	80	10	80	10	6.67
80	222	90	10	90	10	80	0	6.67
90	166	90	0	80	10	90	0	3.33
Calibration 2								
angle	Moving average	test (angle)						average absolute error
		test 1	absolute error	test 2	absolute error	test 3	absolute error	
30	98	30	0	30	0	30	0	0
40	106	30	10	20	20	20	20	16.67
50	146	70	20	50	0	50	0	6.67
60	159	50	10	60	0	50	10	6.67
70	175	60	10	60	10	70	0	6.67
80	192	70	10	80	0	70	10	6.67

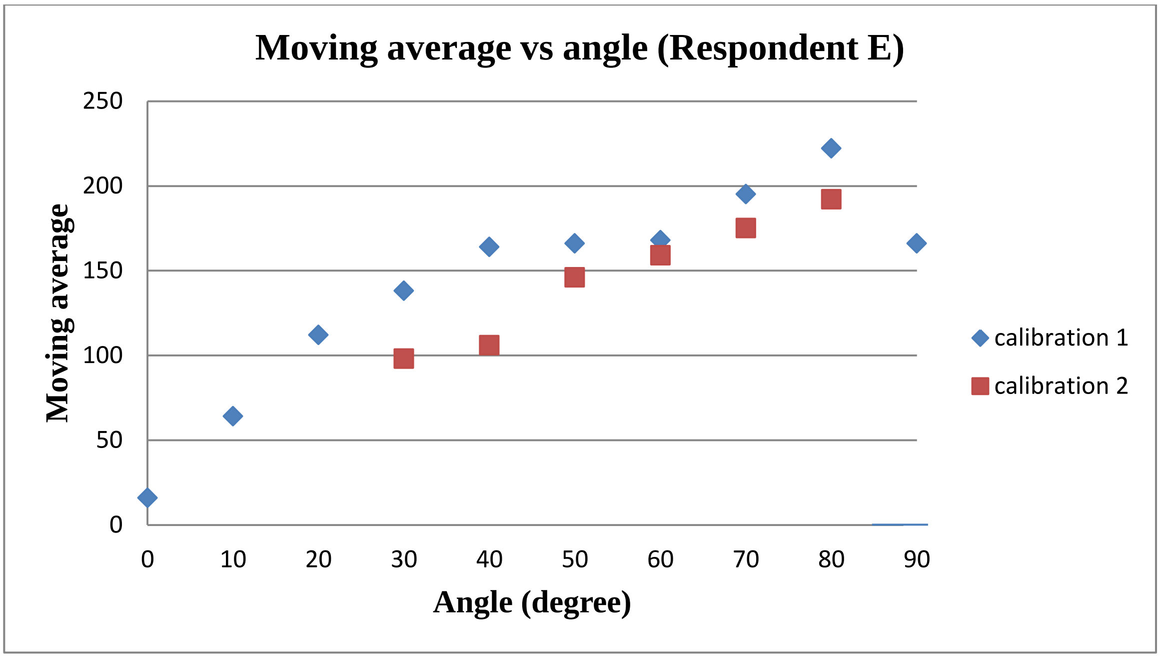


Figure 54: Moving average vs angle for respondent E

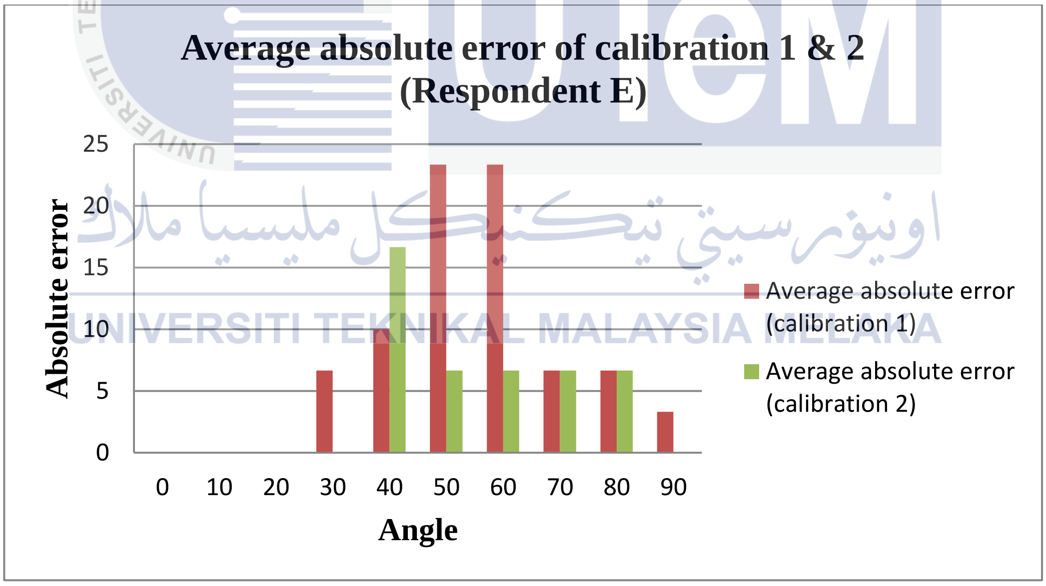


Figure 55: Average absolute error of calibration 1 & 2 for respondent E

Table 9: Performance of program for respondent F

Calibration 1								
angle	Moving average	test (angle) test 1	absolute error	test 2	absolute error	test 3	absolute error	average absolute error
0	44	0	0	0	0	0	0	0
10	77	10	0	10	0	10	0	0
20	111	20	0	20	0	10	10	3.33
30	132	30	0	20	10	40	10	6.67
40	154	40	0	40	0	40	0	0
50	146	40	10	70	20	40	10	13.33
60	139	80	20	40	20	40	20	20
70	128	40	30	60	10	40	30	23.33
80	118	80	0	70	10	80	0	10
90	133	40	50	90	0	70	20	23.33
Calibartion 2								
angle	Moving average	test (angle) test 1	absolute error	test 2	absolute error	test 3	absolute error	average absolute error
30	104	20	10	30	0	40	10	6.67
50	136	40	10	40	10	40	10	10
60	142	40	20	40	20	40	20	20
70	211	70	0	70	0	70	0	0
90	168	90	0	90	0	70	20	6.67

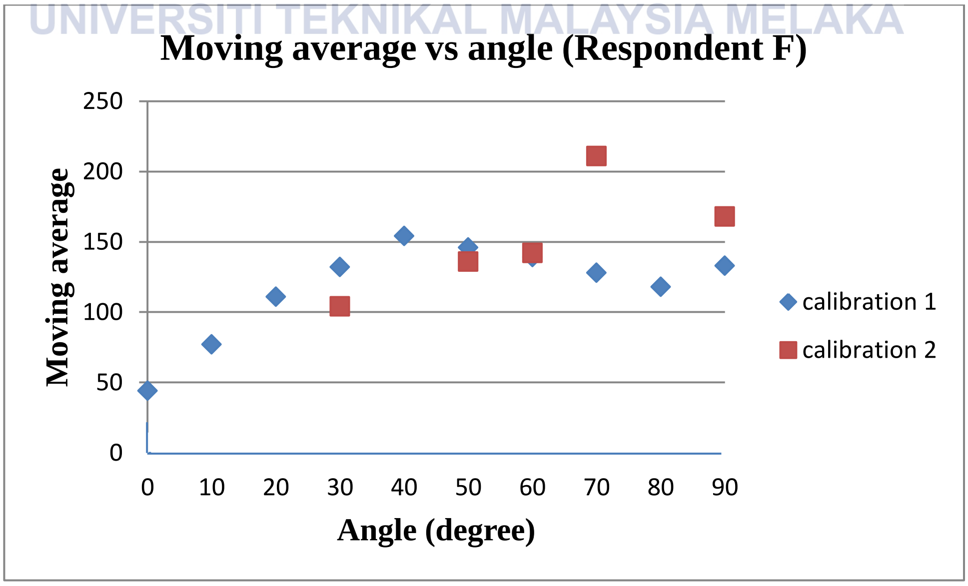


Figure 56: Moving average vs angle for respondent F

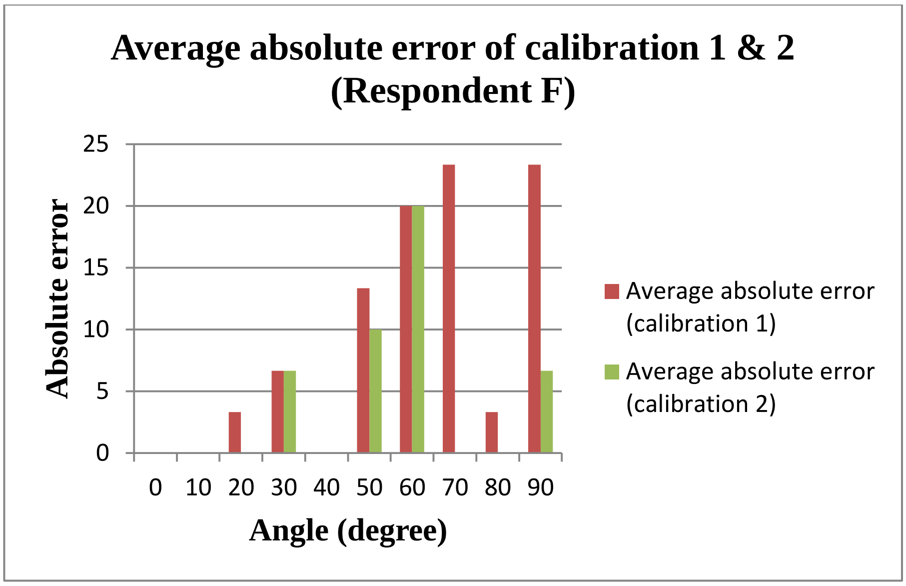


Figure 57: Average absolute error of calibration 1 & 2 for respondent F

Based on the result from table 6 to table 9, we can see that the average absolute error for the 0 degree to 20 degree is almost 0 for every respondent. This is because the separation of moving average in the initial of degree is large, hence the program can classify the degree much more easier.

One of the common error that involves the maximum moving average point is exist in table 6, respondent A, degree 60 and degree 70. As we can see from figure 47, the graph of moving average vs angle is a graph with a turning point. The moving average at 20 and 30 degree during the calibration 1 is 303 and 381 while the moving average at 60 and 70 is 333 and 310. During the testing of 70 degree, test 1 and test 2 show 20 degree while test 3 shows 30 degree. This situation happens because the moving average at 70 degree (310) is close to moving average at 20 degree (303). In theory, the program should has read a signal which belong to maximum point, 40 degree (460) domain and classify the degree to 70 degree once the moving average drop to value near to 310. However this doesn't happen as the result show that it classify it to a 20 degree, this showed that the program can't catch up a moving average that belong to 40 degree. This error may caused by the moving speed of arm. To pass through the 70 degree, the arm must move through 40 degree, however the respondent moves the elbow too quickly, the duration for the EMG at 40 degree is very short. When go through moving average features extraction which is the average of all EMG value in 0.1 seconds, the characteristics of EMG signal at 40 degree already lost in the process, hence the output of moving average is low and unable classified as 40 degree. Some of errors also may occurs due to the inaccurate calibration reading in the calibration

1 process. Calibration 1 is important and respondents need to control their muscle well, for examples do not exert extra force during isometric contraction.

In theory, since all the saved moving average value is get on real time, it is not effect by the user and his or her body conditions. However in practical, there are many factor can effect EMG signal and make it keep changing during the whole experiment. Some of the factors are user bad in muscle control, increasing of body temperature, sweat and fatigue issues. Even through there are resting time between each testing, however it is hard to control the fatigue issues because level of fatigue is depends on metabolism of respondent and workload such as speed of the arm movement. These factors will affect the error of calibration 1 as the moving average value has been changed after several testing in the beginning of the experiment. This reason supports the fact that most of the error happened in the end of experiment which is near the 90 degree. The exist of calibration 2 is to update the moving average value. The testing method of calibration 2 is different than calibration 1 as the testing process start running after update the value. Hence in theory the absolute error of calibration 2 should be less than calibration 1. Based on the graph in figure 47, 49, 51 and 53, we can observe that the moving average in calibration 1 and calibration 2 are different. For respondents A, D and F, 10 out of 14 or 71.43% of moving average value in calibration 2 is higher than the moving average value in calibration 1. These phenomena proved that the fatigue factor exist and has changed the EMG signal. When our muscle start to fatigue, the amplitude of EMG signal will increasing while its frequency will decreasing [32]. As the By referring the graph on figure 51, 53, 55 and 57, most of the absolute error of calibration 2 show decreasing sign. Figure 58 shows the comparison of average absolute error in calibration 1 and calibration 2 for all respondents.

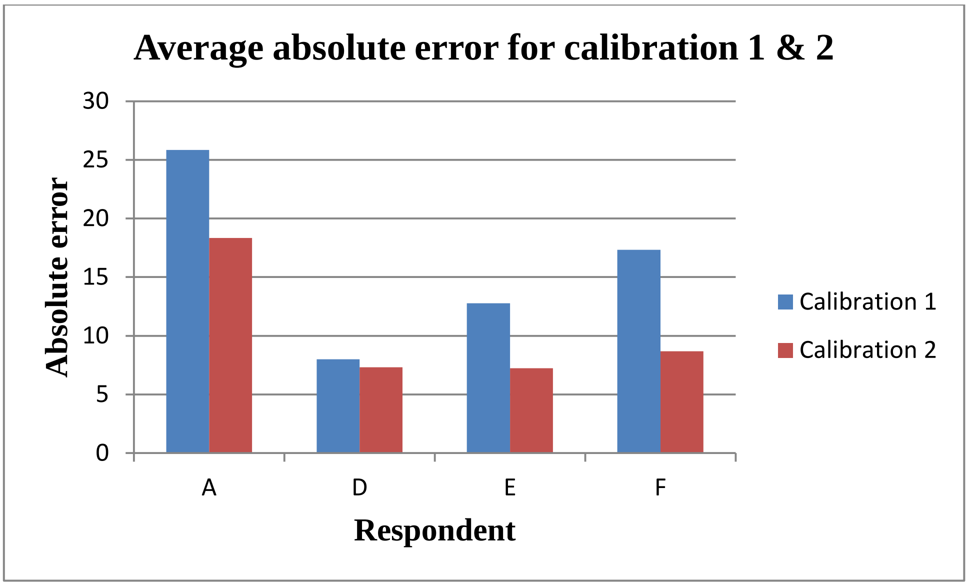
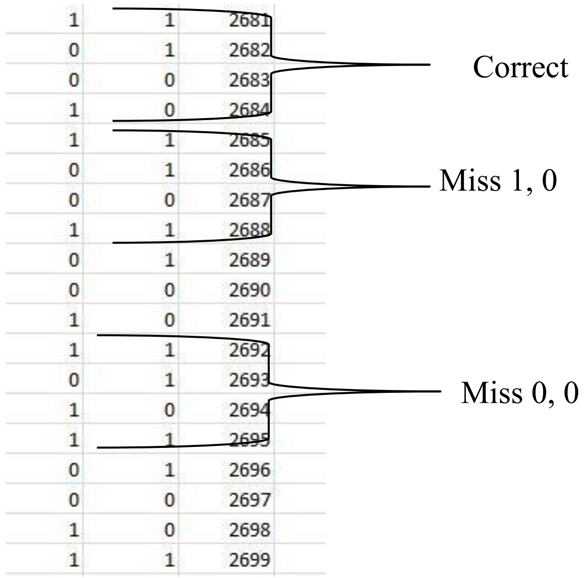


Figure 58: Average absolute error for calibration 1 & 2 for all respondents

The angle involves in figure 58 is only limit to angle that has go through calibration 1 and calibration 2 process only. The result in figure 58 showed that the absolute error of calibration 2 is less than calibration 1. The performance of program has improved after calibrate for the second time and apply directly. This also concludes that the moving average for different angle is changing during the experiment. The saved in moving average value which represent the joint angle need to be updated frequently to ensure the performance of the program.

4.4 Experiment 4: Identify and improve the error in DC motor with encoder

Figure 59 shows the encoder state of the DC motor while it running. The first column is encoder A, second column is encoder B and the third column is the count number. The normal encoder value for clockwise rotation is 11, 01, 00, 10.



1	1	2681
0	1	2682
0	0	2683
1	0	2684
1	1	2685
0	1	2686
0	0	2687
1	1	2688
0	1	2689
0	0	2690
1	0	2691
1	1	2692
0	1	2693
1	0	2694
1	1	2695
0	1	2696
0	0	2697
1	0	2698
1	1	2699

Figure 59: Encoder states of the DC motor

Based on figure 59, that there is error for the encoder has missed some of the encoder value. The error may caused by the slow processing speed of the Arduino Uno board or the slow response time inside the encoder mechanism. The error may cause the motor rotate for larger degree than what desired. During the experiment, it also can be observed that the error is proportional to the speed of motor. To reduce the error, encoder pulse is used instead of encoder count to control the position of rotor. 1 encoder pulse is equal to 4 state of encoder count. By monitor the encoder pulse instead of encoder count, the processing time for the programming in monitoring the rotor position can be saved.

The flow chart of motor controller is attached in Appendix B while Arduino programming code and flow chart is attached in Appendix F. The maximum PWM is set to 255 while the minimum PWM is set to 180. Table 10 shows the percentage of encoder error versus angle.

Table 10: Percentage of encoder error vs desired angle

Desired Angle (degree)	Percentage of encoder error (%)
20	66
40	45
60	30
90	24

A graph is plotted using Microsoft Excel. Figure 60 shows the graph of desired angle versus percentage of error. By using trendline function in Microsoft Excel, the most suitable equation for mapping is polynomial equation in power of 2. The mathematic model is:

$$y = 0.0096x^2 - 1.6653x + 95.639 \tag{4.2}$$

y=Percentage of error, x=rotate angle

Figure 57 shows that the error of encoder is decrease with the increase of pulse count or angle of rotation

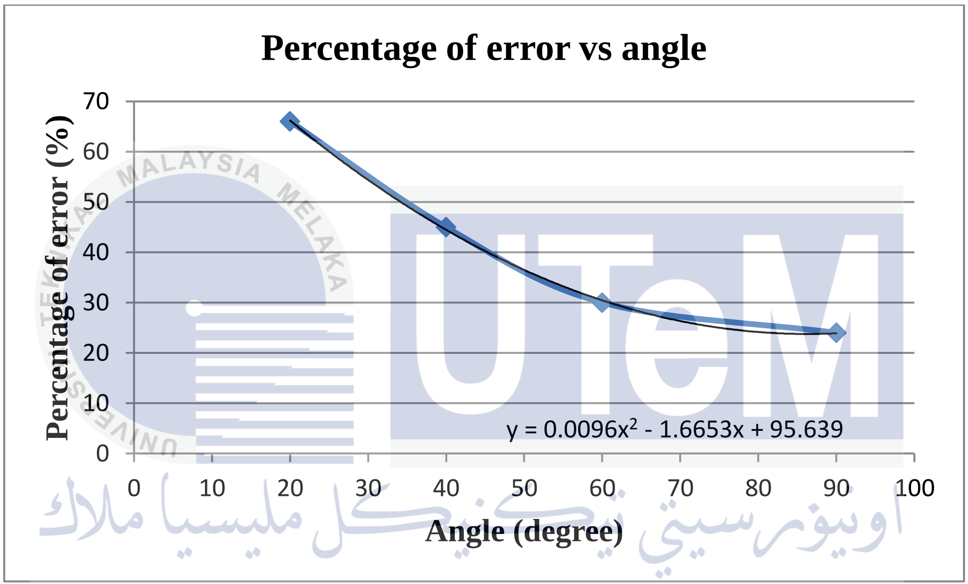


Figure 60: Percentage of error vs angle

The percentage of error is predicted using equation 4.2, and then the calculated pulse number will be decreased based on the percentage of error. The calculation is showed in below.

Based on equation 4.2,

$$\text{Percentage of error} = 0.0096(90^2) - 1.6653(90) + 95.639$$

$$= 23.522\%$$

Motor gear ratio is 3.

For 90 degree, the rotation of rear shaft, Θ_1 = gear ratio x rotate degree

= 300 x 90

= 27000 degree

Encoder has 3 pulses per shaft revolution

Total pulse required = (rotation of rear shaft/revolution)*3

= (27000/360)*3

= 225

Actual pulse required = Total pulse – error percentage*total pulse

= 225-(23.522*225/100)

= 172.07

The overall result of DC motor test run by using the formula (4.2) is showed in table 11.

Table 11: Absolute error of DC motor for different angle

The desired angle (degree)	Obtained angle (degree)						Average Absolute error
	Test 1	Absolute error	Test 2	Absolute error	Test 3	Absolute error	
10	10	0	10	0	10	0	0
20	20	0	19	1	19	1	0.67
30	30	0	30	0	31	1	0.33
40	40	0	40	0	40	0	0
50	52	2	51	1	50	0	1
60	60	0	60	0	60	0	0
70	72	2	72	2	71	1	1.67
80	84	4	82	2	85	5	3.67
90	90	0	90	0	90	0	0

The overall result for DC motor is good. The maximum error is fall among 70 and 80 degree. This is because in figure 60, the mathematic model generated by the trendline function in Microsoft excel does not completely cover or match on these 2 degree.

Figure 61 shows the time response for 90 degree. Its detail reading can found in the CD under the name of “result experiment 4”.

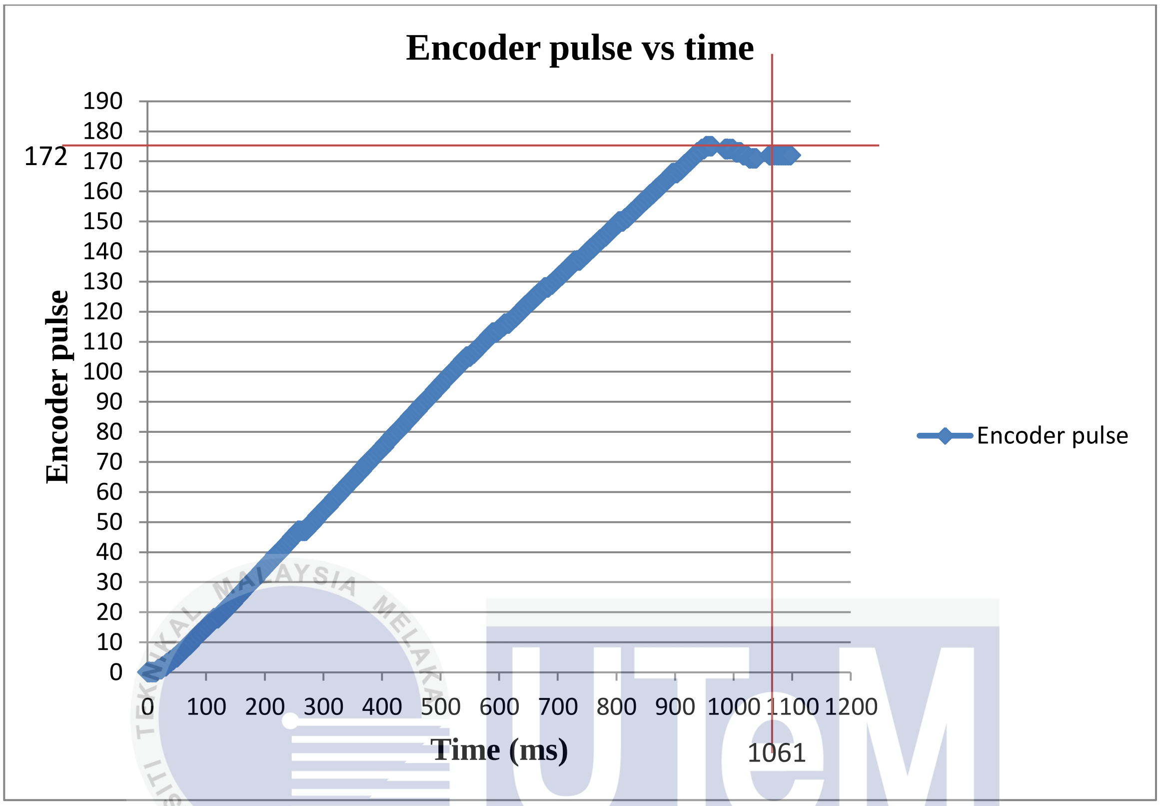


Figure 61: Encoder pulse vs time

Based on data obtained, the DC motor took about 1.06 seconds to reach 90 degree. There are some damping when it near to the desired angle. The damping is caused by the inertia of the motor. The maximum amplitude is 175, hence the percentage of overshoot can be calculated as showed in below.

Percentage of overshoot = (maximum value – steady state value)/steady state value x 100%

= (175-172)/172 *100%

= 1.744%

The percentage of overshoot is small, it is mainly due to the selection motor has high gear ratio. The high gear ratio produces high torque and slow speed performance. The motor can stop in shortest time once the power supply is cut off.

CHAPTER 5

CONCLUSIONS

5.1 Conclusions

As the conclusions, the relationship between joint angle, torque and EMG signal have been studied in experiment 1 and 2. When EMG is recorded with minimum torque applied to the muscle, the amplitude of EMG is low. The low amplitude of signal will has a low S/N, and hence it can't show out the exact feature available in the EMG signal. The moving average feature and RMS feature is almost same in nature, they share the same shape. In the result of moving average vs angle, it shows that there are 2 type of graph. The first type of graph has a turning point in the middle of the angle, its value start decreasing in the middle of the angle. This effect is caused by the force shifting from biceps brachii to anterior deltoid. The second type doesn't has a turning point, its value is proportional to the angle. For the second objective, an arduino program has been designed to map the EMG signal to joint angle. The performance of program is acceptable. However it showed that the moving average of EMG signal is increasing during the experiment due to the fatigue factor that generated inside the muscle. The saved in EMG value that represent the joint angle needs to be updated to reduce the error of the system. The program in experiment 3 and experiment 4 can combined together to create a complete system that using EMG signal to control the position of DC motor.

For the motor actuator, there is error inside the encoder of DC motor. The error is inversely proportional to the rotation angle. The results obtained through serial monitor showed that the encoder error can be eliminated by calculate its error through mathematics model and reduce it from the program. Due to the motor actuator has high gear ratio, the percentage of overshoot of motor actuator is low.

Figure 61 shows the combination of software and hardware of this project. It shows that the rotor can follow the movement of arm with little error. To prevent the interrupt of electric supply to the arduino board, the power source of arduino board and dc motor should be same.

Figure 62: Combination of software and hardware

5.2 Suggestions and future work

The project involves using of EMG signal to control a DC motor, one of the major problem is the fatigue issues which can change the amplitude of EMG signal. It is hard to predict the effect bring by fatigue to the mathematic model because the level of fatigue depends on the body metabolism, the weight exerted on muscle and the speed of the motion. It is suggested that a encoder may be placed together with the elbow. The angle given by the encoder will synchronize with the EMG moving average value to keep update the saved moving average value inside the program. The use of artificial intelligence like neural network may be used to classify the signal and continuous update the program. Neural network is possible apply in more powerful arduino board, however the number of hidden layer need to be controlled so that it won't exceed the memory limit of Arduino board. The problem about the DC motor encoder error, it may be solved by using processor with higher speed.

REFERENCES

[1] (2013, 27 October). *Storke Prevention*. Available: http://www.nasam.org/english/prevention-what_is_a_stroke.php

[2] (2013, 27 October). *Stroke statistics*. Available: http://www.uhnj.org/stroke/stats.htm

[3] R. Kai & L. Li, *An Introduction to Biomechatronics*. Singapoore: Pan Stanford Publishing Pte, Ltd, 2011.

[4] R.Kai & X.L. Hu, "Service Robotics: Robot-Assisted Training for Stroke Rehabilitation," 2008.

[5] K. Najarian & R. Splinter, *Biomedical signal and image processing*, Second edition ed. New York: CRC Press, Taylor & Fancis Group, 2012.

[6] Twentle Medical System International. B.V. (2005). *The measurement of an EMG*. Available: http://www.tmsi.com/en/applications/electromyography.html

[7] A. V. S. Reana &. R. Aggarwal, "2-d robotic arm control using EMG signal," *International Journal of Computer Applications (0975 -8887)*, vol. 72, June 2013.

[8] A. Zweig, "Controling a crane arm with EMG sensor," *Spring Term 2011*, 2011.

[9] W. S. Pease, H. L. Lew & E. W. Johnson, *Johnson's practical electromyography* vol. 4 edition: Lippincott Williams & Wikins, 2007.

[10] SENIAM. (2013, 1 November). *Preparation of the skin*. Available: http://www.seniam.org/

[11] Seniam. (2013). *Recommendations for sensor locations in arm or hand muscles*. Available: http://www.seniam.org/

[12] P. Konrad, "The ABC of EMG," in *A Practical Introduction to Kinesiological Electromyography*, ed: Noraxon INC. USA., April 2005.

[13] R. Boostani &. M. H. Moradi. Evaluation of the forearm EMG signal features for the control of a prosthetic hand [Online].

[14] Mohan. C & V. K. Giri, "DC Motor Control using EMG Signal for Prosthesis," *IJECT 2011*, vol. 2, 2011.

[15] K. Ullah and J. H. Kim, "A mathematical model for mapping EMG signal to joint torque for the human elbow joint using nonlinear regression," in *Autonomous Robots and Agents, 2009. ICARA 2009. 4th International Conference on*, 2009, pp. 103-108.

[16] F. Yuanjie and Y. Yuehong, "Mechanism design and motion control of a parallel ankle joint for rehabilitation robotic exoskeleton," in *Robotics and Biomimetics (ROBIO), 2009 IEEE International Conference on*, 2009, pp. 2527-2532.

[17] (2014, 1 May). *Mechanic in excercise: Angle of pull*. Available: http://www.exrx.net/Kinesiology/AnglePull.html

[18] F. P. Carpes, J. M. Geremia, A. P. B. Karolczak, F. Diefenthaeler & M. A. Vaz D, "Preference and torque asymmetry for elbow joint," *Motriz: rev. educ. fis*, vol. 18, pp. 319-326, 2012.

[19] J. Y. Gefen, A. S. Gelmann, G. J. Herbison, M. E. Cohen & R. R. Schmidt, "Use of shoulder flexors to achieve isometric elbow extension in C6 tetraplegic patients during weight shift."spinal cord, vol.35, p.308,1997.

[20] S. J. Chapmen, *Electric Machinery Fundamentals*, 4 ed. New York: Elizabeth A.Jones, 2003.

[21] A. Ameri, K. B. Englehart, & P. A. Parker, "A comparison between force and position control strategies in myoelectric prostheses," in *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, 2012, pp. 1342-1345.

[22] R. Wondae, H. Byungkil, & K. Jaehyo, "Continuous Position Control of 1 DOF Manipulator Using EMG Signals," in *Convergence and Hybrid Information Technology, 2008. ICCIT '08. Third International Conference on*, 2008, pp. 870-874.

[23] University of Rochester Medical. Center. (2013, 1 November 2013). *Maximum Isometric Voluntary Contraction Testing (MVICT):* . Available: http://www.urmc.rochester.edu/fields-center/protocols/documents/quantitativemyometry.pdf

[24] T.E.Bernard. (2013, 10 December). *Design of static muscle work*. Available: http://personal.health.usf.edu/tbernard/HollowHills/SMWDesignM20.pdf

[25] R. N. Scott, *An introduction to Myolectric prosthesis*. New Brunswick, Canada: Bio Engineering Institute, University of New Brunswick, Canada, 1984.

[26] A. Technologies. (2013, 15 September). *Muscle sensor v3*. Available: http://www.advancertechnologies.com/p/muscle-sensor-v3.html

[27] Futurlec. (2013, 20 September). *Technical Information - National Semiconductor 7805T Datasheet*. Available: http://www.futurlec.com/Linear/7805T.shtml

[28] Arduino. (2013, 2013). *Arduino Uno*. Available: http://arduino.cc/en/Main/arduinoBoardUno

[29] Cytron. Technologies. (2013, 22 September 2013). *LCD keypad shield*. Available: http://www.cytron.com.my/viewProduct.php?pcode=SHIELD-LCD&name=LCD%20keypad%20Shield

[30] ST. microelectronics. (2013, 20 September). *L298Dual Full Bridge Driver*. Available: http://www.st.com/web/catalog/sense_power/FM142/CL851/SC1790/SS1555/PF63147

[31] Cytron. Technologies. (2013, 22 September). *DC Geared Motor with Encoder*. Available: http://www.cytron.com.my/viewProduct.php?pcode=SPG30E-300K&name=DC%20Geared%20Motor%20with%20Encoder

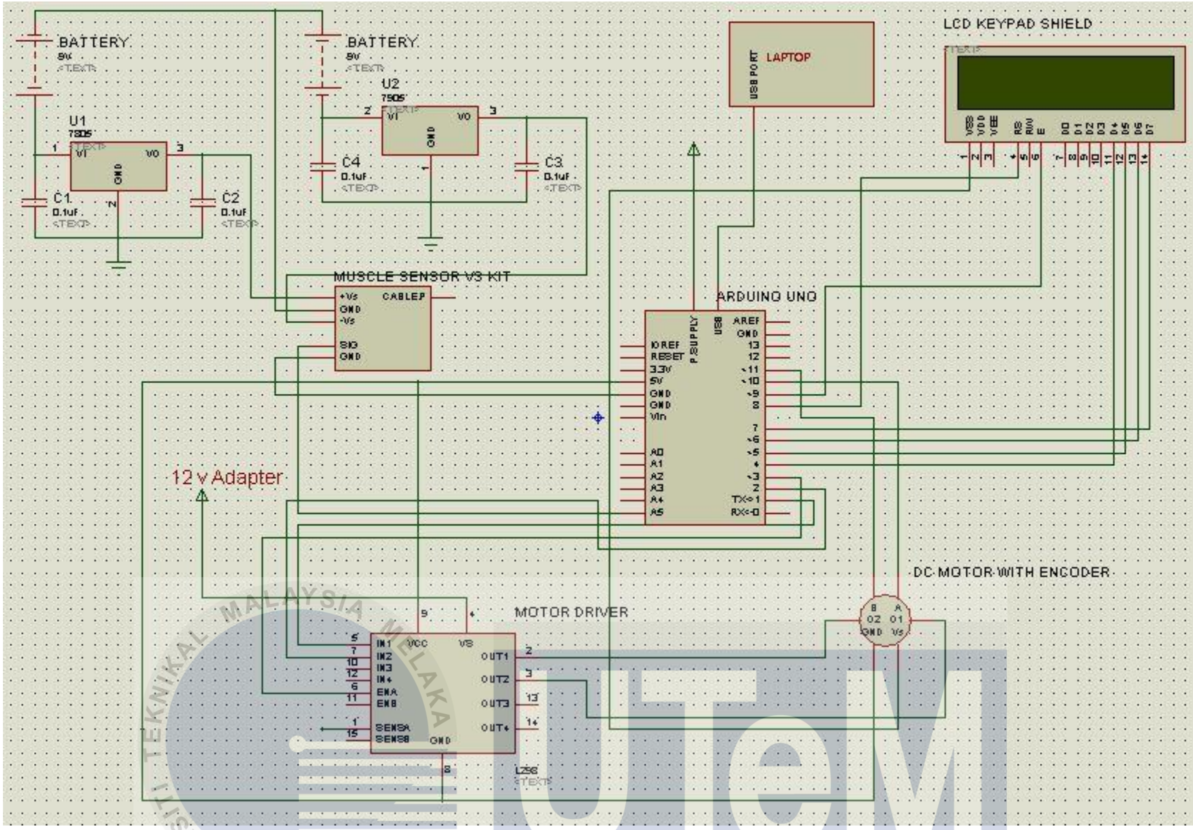
[32] T. D. Lalitharatne, Y. Hayashi, K. Teramoto, and K. Kiguchi, "A study on effects of muscle fatigue on EMG-based control for human upper-limb power-assist," in *Information and Automation for Sustainability (ICIAfS), 2012 IEEE 6th International Conference on*, 2012, pp. 124-128.

Appendices

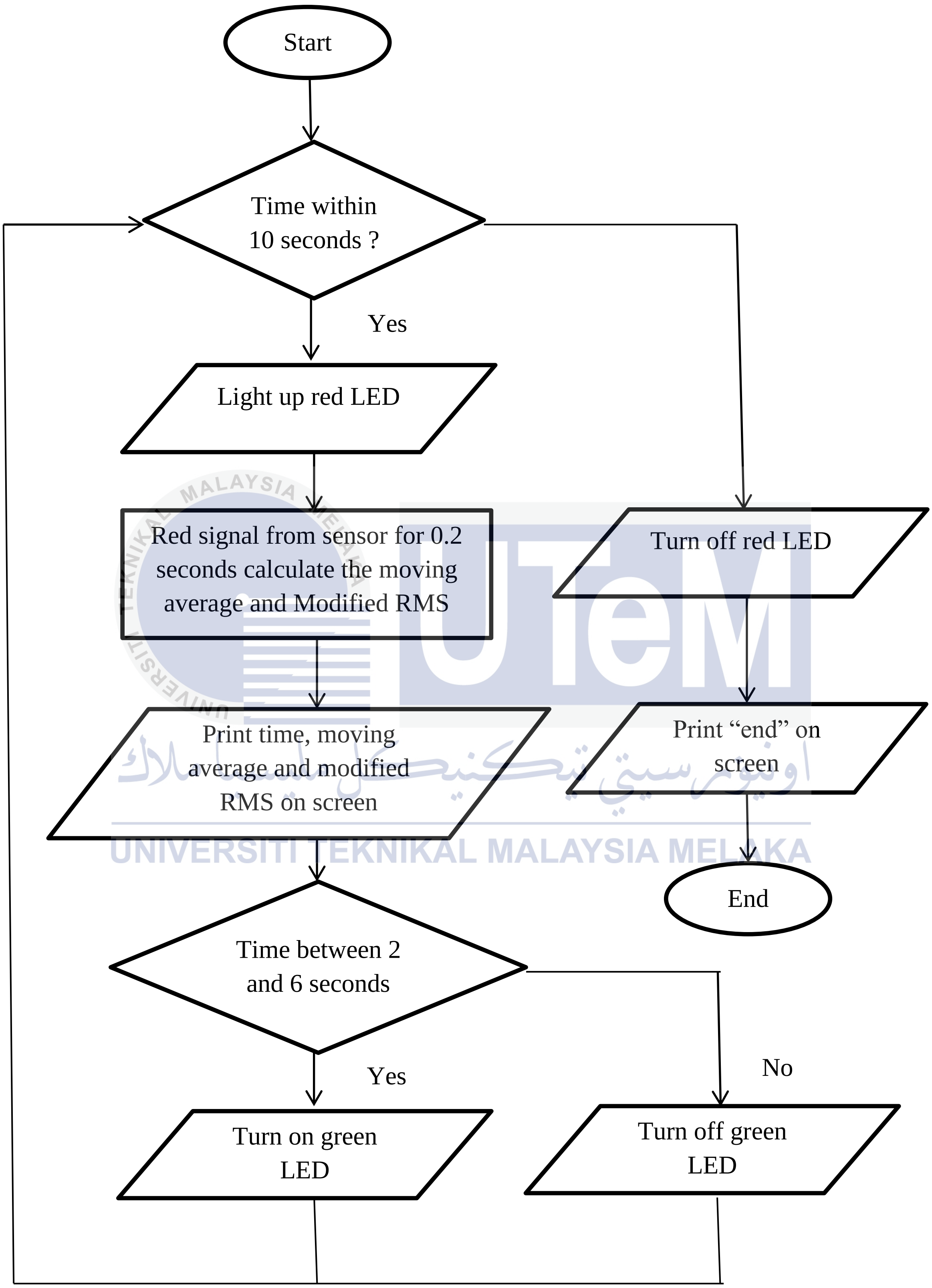
Appendix A: List of item

No	Item	Quantity	Price
1	Muscle Sensor V3 kit	1	RM280
2	G203 Disposable Electrode	-	RM0.80 per piece-
3	Arduino Uno board	1	-
4	LCD keypad Shield	1	-
5	DC gear motor with encoder	1	RM70
6	Motor driver L298	1	RM12
7	Voltage regulator 5V 1A, 7805T	1	-
8	Voltage regulator -5V 1A, 7905T	1	-
9	Capacitor, 0.1µF	2	-
10	LED	2	-
11	9V battery	2	RM10
12	9Vbattery holder	2	RM1.00
13	12V 2A Adapter	1	RM18
14	Wire		-

Appendix B: Overall Circuit



Appendix C: flow chart and programming code for experiment 1



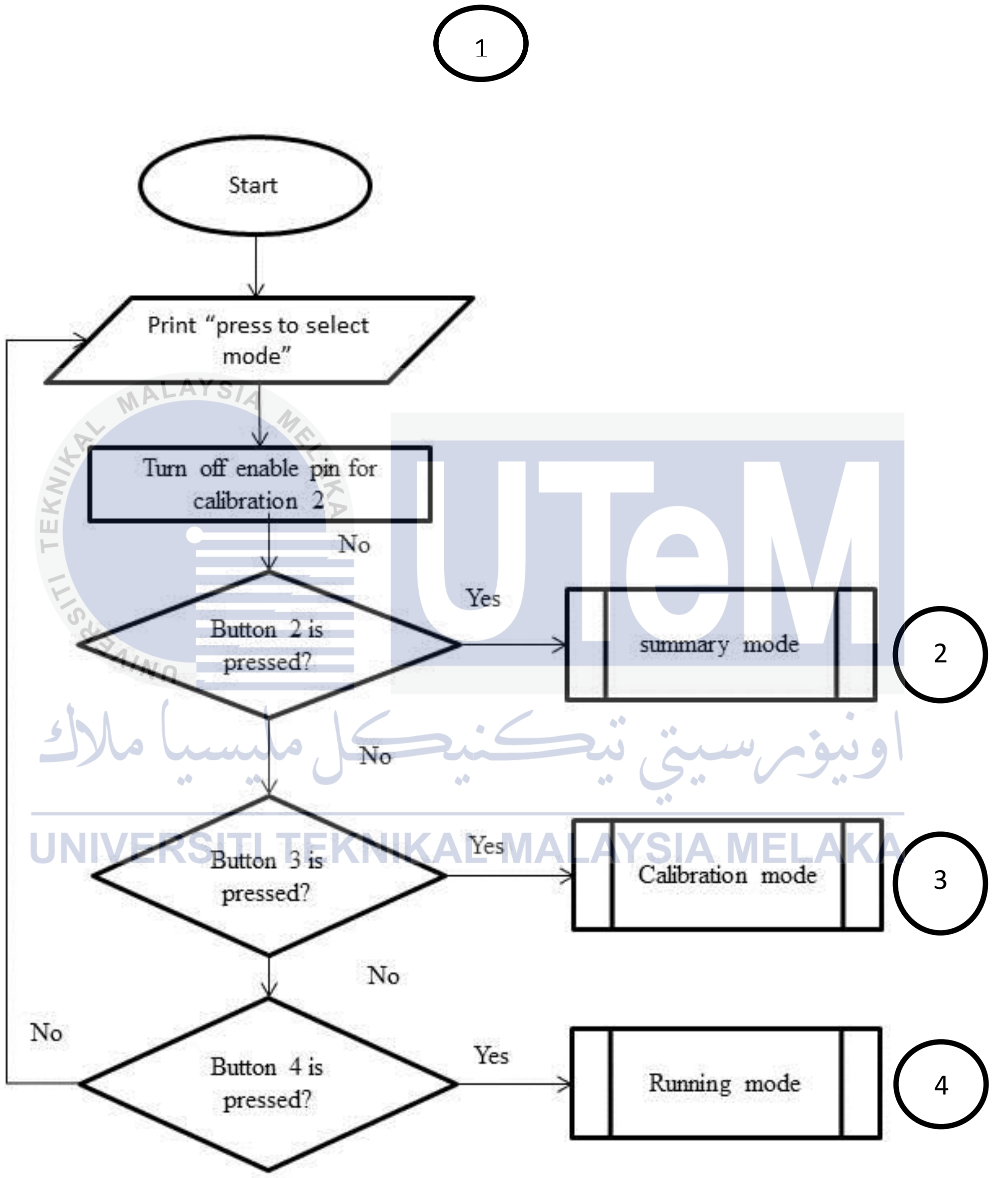
```
int i;int state;int en;int indicator;long count;float RMS;float VRMS;float TRMS;
float number; float number2;long mean;long total;long time;unsigned long time20;

void setup() {
Serial.begin(57600);
pinMode(2,INPUT);pinMode(8,OUTPUT);pinMode(7,OUTPUT);
}
void loop()
{for(i=0;i<50;i++)
{digitalWrite(7,HIGH);
time=millis();time20=time+200;
 while (millis()<time20)
{number=analogRead(A0);number2=number/5;
total=total+number;VRMS=number2*number2;TRMS=TRMS+VRMS;
count++;
   }
mean=total/count;RMS=TRMS/count;RMS=sqrt(RMS);
Serial.print(mean);Serial.print(",");Serial.print(RMS);Serial.print('\n');
total=0;count=0; VRMS=0;TRMS=0;
if (i>9){digitalWrite(8,HIGH);}
if (i>29){digitalWrite(8,LOW);}
}
Serial.print("end");Serial.print('\n');digitalWrite(7,LOW);
}
```

Appendix D: Data about respondents in this thesis

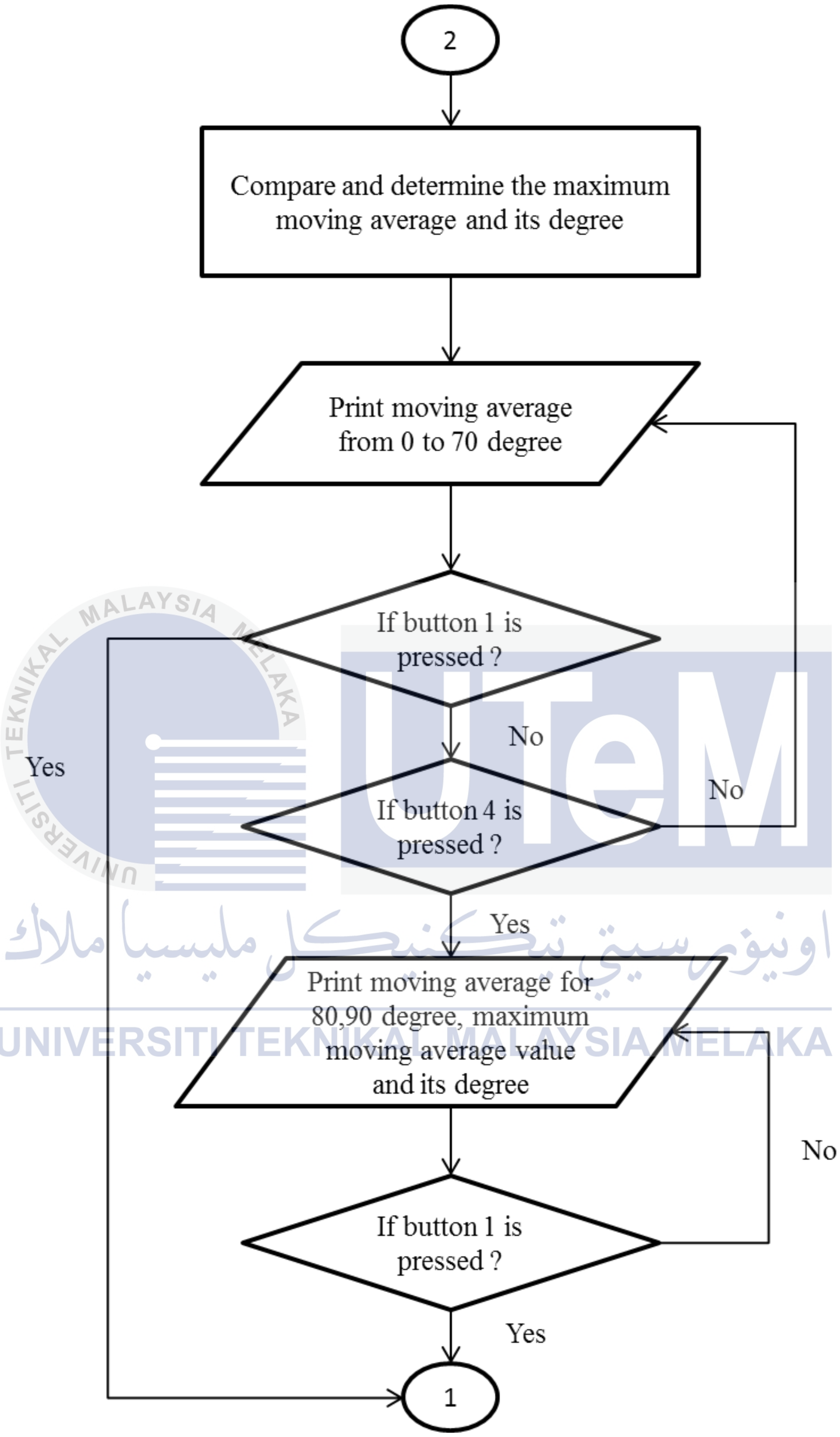
Experiment 1	Respondent	Length from medial acromion to fossa Cubit	Maximum voluntary Contraction	Selected weight
1	A	28cm	8kg	2.5kg
2	A	28cm	8kg	2.5kg
	B	27cm	7kg	2.25kg
	C	22cm	5kg	1.75kg
	D	23cm	4.5kg	1.25kg
3	A	28cm	8kg	2.5kg
	D	23cm	4.5kg	1.25kg
	E	34cm	8kg	2.25kg
	F	32cm	8kg	2.5kg

Appendix E: Flow chart and coding for program in experiment 3

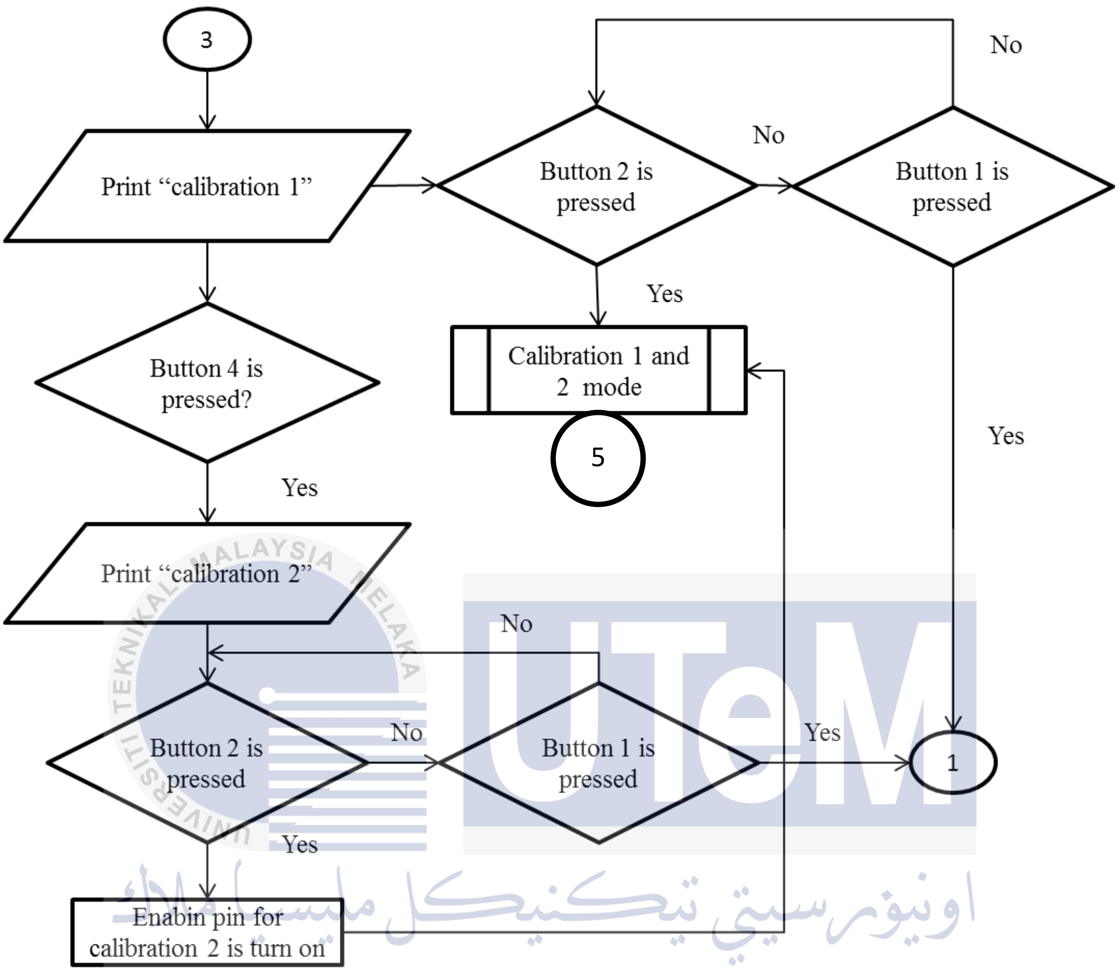
main()



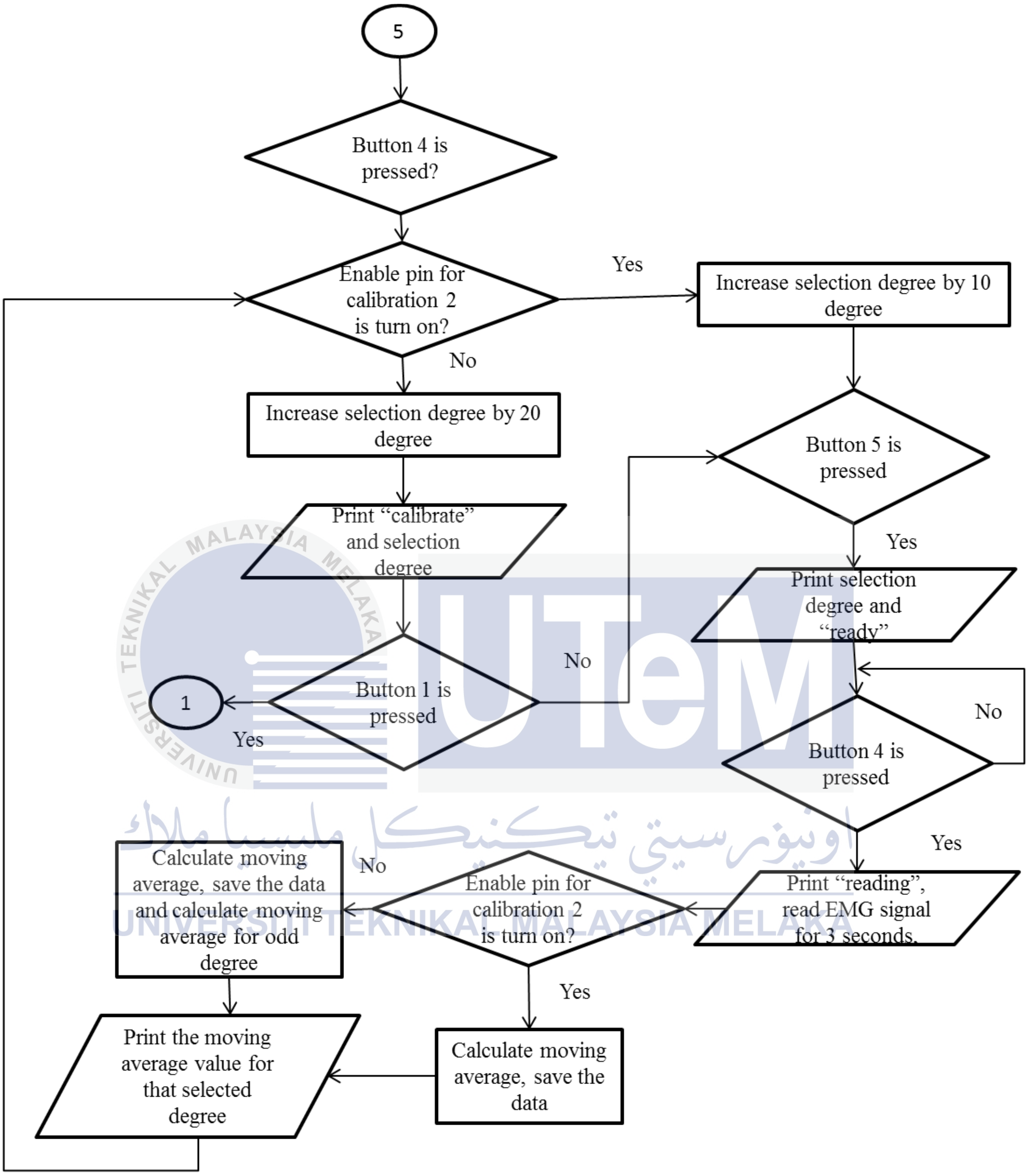
summary mode ()



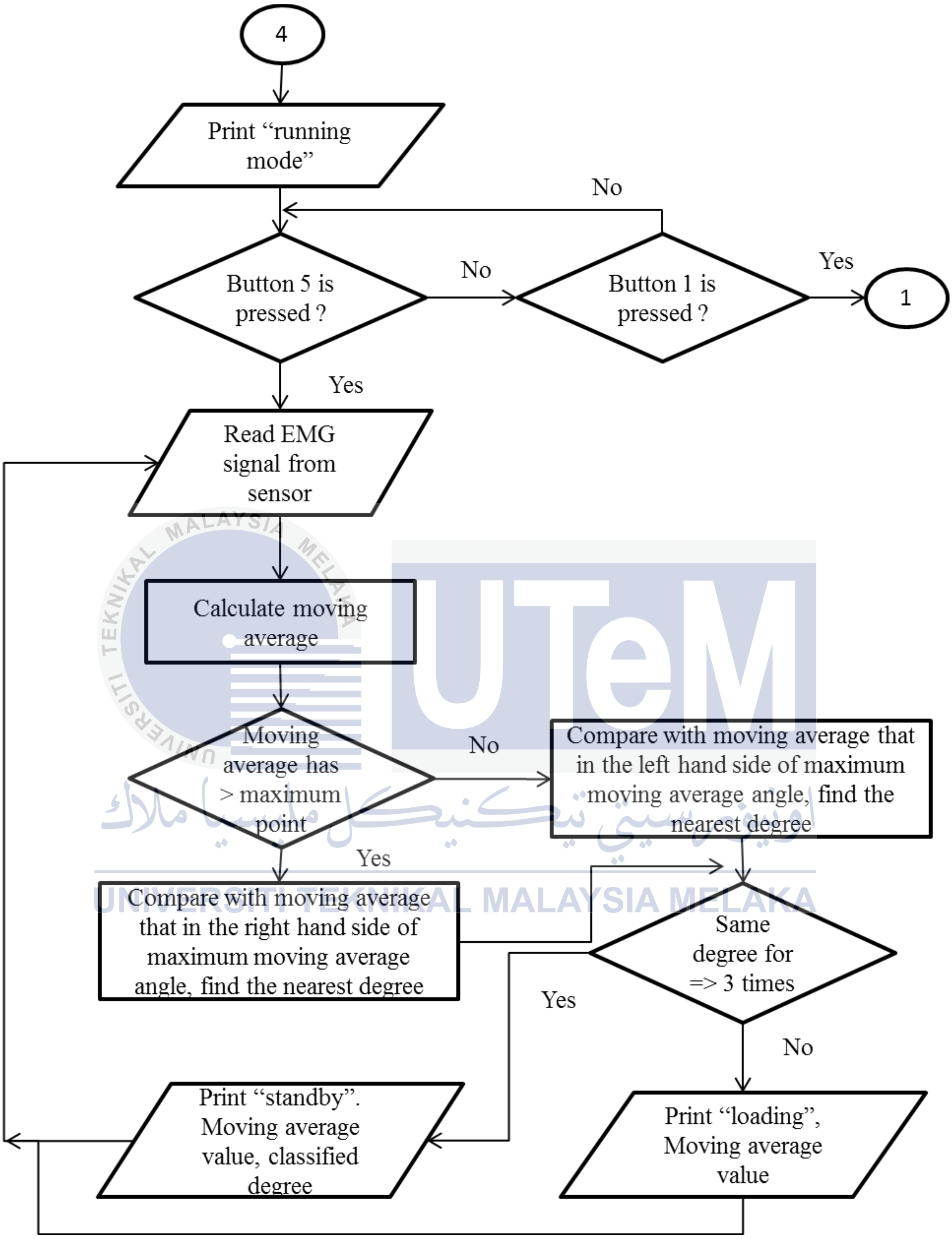
Calibrate mode ()



calibration 1 & 2 mode ()



Running mode



```
#include <LiquidCrystal.h>

#include <LCD_Key.h>

LiquidCrystal lcd(8, 9, 4, 5, 6, 7);

LCD_Key keypad;

int localKey = 0; int i;int j;int degree;int correct_en;int k;int en3;int l;int midpt;

long time;long time_20;long count;long number;long total_t;long total;

int mean;int mean_t;int mean_0d;int mean_10d;int mean_20d;int mean_30d;int
mean_40d;int mean_50d;int mean_60d;int mean_70d;int mean_80d;int mean_90d;

int compare;int over;int max_90;int en4;

int mean_deg[10]={0,100,200,300,400,700,600,500,400,300};int max_deg;int
max_mean;int mean_diff[10];int min_meandiff;int min_diffdeg;int previous_deg;

String keyString = "";

void setup()

{ Serial.begin(9600);

  lcd.begin(16, 2);

  lcd.clear();

  keypad.setRate(10);

  mode_select();

}


void loop()

{ }


void mode_select()

 {degree=0;correct_en=0;j=0;previous_deg=0;

  lcd.setCursor(0, 0);

  lcd.print("Press to");

  lcd.setCursor (0,1);

  lcd.print("select mode");

  localKey = keypad.getKey();
```

```
 if (localKey == 3)
   {lcd.clear();calibration_mode();}
  else if (localKey == 4)
   { lcd.clear();running_mode();}
  else if (localKey==2)
  {max_value();lcd.clear();
   summary_1();}
  else {mode_select();}
 }

 void summary_1()
 {lcd.setCursor(0,0);
    for (i=0;i<4;i++)
      {lcd.print(mean_deg[i]);lcd.print(",");}
    lcd.setCursor(0,1);
    for (i=4;i<8;i++)
       {lcd.print(mean_deg[i]);lcd.print(",");}
     localKey = keypad.getKey();
     if (localKey==4)
    {lcd.clear();
     summary_2();}
     else if (localKey==1)
    {lcd.clear();mode_select();}
    else {summary_1();}
 }

void summary_2()
{lcd.setCursor(0,0);
    for (i=8;i<10;i++)
```

```
        {lcd.print(mean_deg[i]);lcd.print(",");}
lcd.setCursor(0,1);lcd.print("max:");lcd.print(max_deg);lcd.print("deg,");lcd.print(max_me
an);
  localKey = keypad.getKey();
 if (localKey==1)
 {lcd.clear();mode_select();}
 else {summary_2();}
}

void calibration_mode()
{ lcd.setCursor(0,0);
  lcd.print("Calibration 1");
  lcd.setCursor(0,1);lcd.print("mode");
  localKey = keypad.getKey();
  if (localKey == 2)
  { lcd.clear();
    calibrate_deg();
  }
  else if (localKey==4)
  {lcd.clear();
   correction_mode();}
  else if (localKey == 1)
  {lcd.clear();
   mode_select();}
  else
  {calibration_mode();}
}

 void calibrate_deg()
 {lcd.setCursor(0,0);
```

```
 lcd.print("calibrate");
 lcd.setCursor(0,1);
 lcd.print(degree);lcd.print(" deg");
 localKey = keypad.getKey();
 if (localKey==4)
 { select_deg();}
 else if (localKey==1)
 {lcd.clear();
 mode_select();}
 else if (localKey==5)
 {lcd.clear();
  readvalue();}
 else {calibrate_deg();}
 }

void select_deg()
{localKey=keypad.getKey();
if (localKey==0)
   {if (degree==80)
   {degree=degree+10;j=j+1;}
   else {degree=degree+20;j=j+2;}
   calibrate_deg();}
else {select_deg();}
}

void select_deg1()
{localKey=keypad.getKey();
if (localKey==0)
{degree=degree+10;j=j+1;correct_deg();}
```

```
else {select_deg1();}
}

 void readvalue()
 {lcd.setCursor(0,0);
 lcd.print(degree);lcd.print(" deg");
 lcd.setCursor(0,1);
 lcd.print("ready");
 localKey = keypad.getKey();
 if (localKey==4)
 {lcd.setCursor(0,1);lcd.print("reading");
   for(i=0;i<30;i++)
   {
   time=millis();time_20=time+100;
   while (millis()<time_20)
     {number=analogRead(A5);
      total_t=total_t+number;count++;
      }
   mean_t=total_t/count;total=total+mean_t;
    }
  mean=total/30;
  mean_deg[j]=mean;lcd.clear();
   lcd.setCursor(0,1);lcd.print(mean_deg[j]);lcd.print("
");mean_t=0;total=0;total_t=0;count=0;
   delay(2000);
   if (correct_en==1)
   {max_value();select_deg1();}
   else {estimate_value();max_value();select_deg();}
 }
 else if (localKey==1)
```

```
 {if (correct_en==1)
  {select_deg1();}
  else {select_deg();}}
 else {readvalue();}
 }

void correction_mode()
{lcd.setCursor(0,0);lcd.print("Calibration 2");
lcd.setCursor(0,1);lcd.print("mode");
localKey=keypad.getKey();
if (localKey==2)
{correct_deg();}
else if (localKey==1)
{lcd.clear();mode_select();}
else {correction_mode();}
}

void correct_deg()
{correct_en=1;
  lcd.setCursor(0,0);
  lcd.print("correct");
  lcd.setCursor(0,1);
  lcd.print(degree);lcd.print(" deg");
  localKey = keypad.getKey();
  if (localKey==4)
  { select_deg1();}
  else if (localKey==1)
  {lcd.clear();
  mode_select();}
```

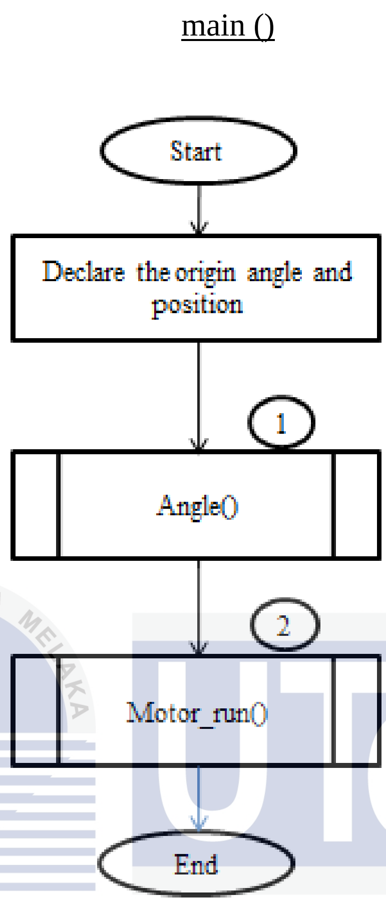
```
 else if (localKey==5)
 {lcd.clear();
  readvalue();}
 else {correct_deg();}
}

void estimate_value()
{if (j==2) {mean_deg[1]=(mean_deg[0]+mean_deg[2])/2;}
 else if (j==4) {mean_deg[3]=(mean_deg[2]+mean_deg[4])/2;}
 else if (j==6) {mean_deg[5]=(mean_deg[4]+mean_deg[6])/2;}
 else if (j== 8) {mean_deg[7]=(mean_deg[6]+mean_deg[8])/2;}
}

void max_value()
{l=0;
 max_mean=mean_deg[0];
 for (i=0;i<10;i++)
 {if (mean_deg[i]>=max_mean)
  {max_mean=mean_deg[i];l=i;}
  else {}}
  max_deg=l*10;
  midpt=(mean_deg[l]+mean_deg[l-1])/2;
  if (max_deg==90){max_90=1;}
  else {max_90=0;}
}

 void running_mode()
{
   lcd.setCursor(0,0);
```

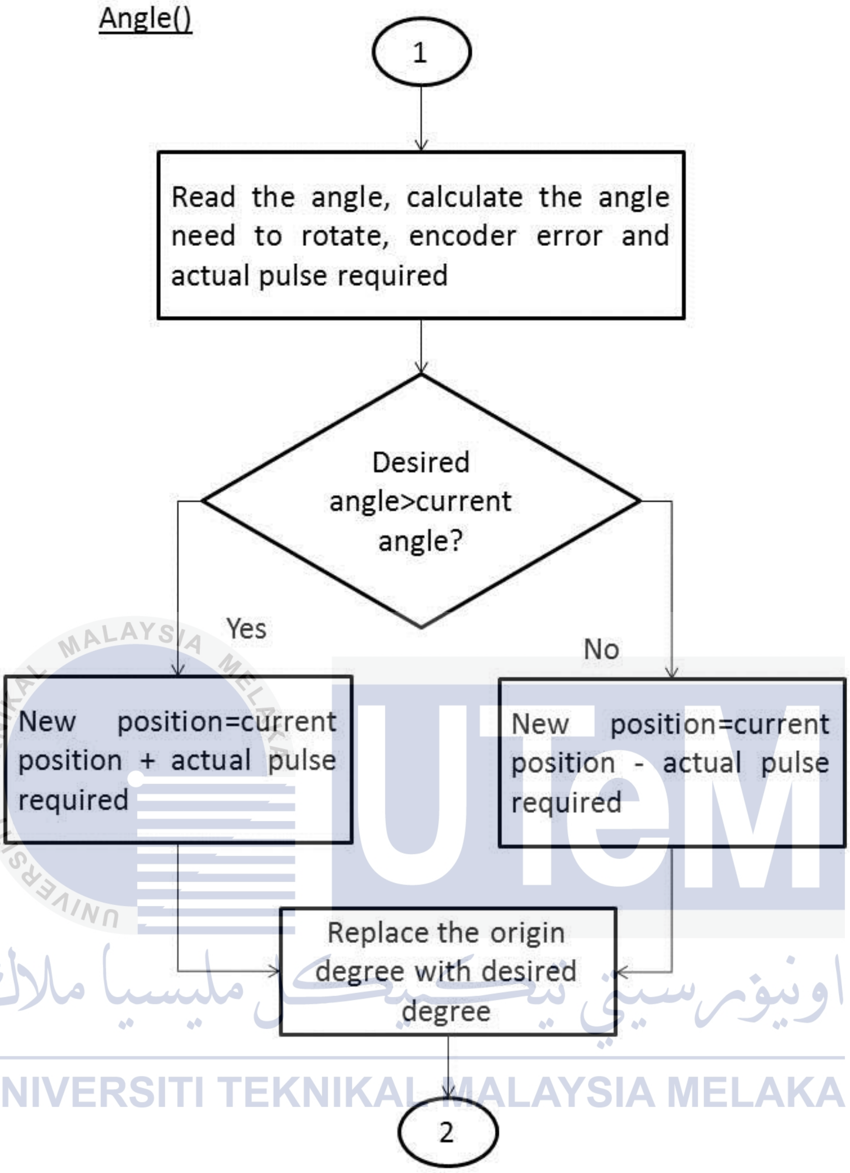
```
  lcd.print("Running");
  lcd.setCursor(0,1);
  lcd.print("mode");
   localKey = keypad.getKey();
 if (localKey == 5)
 {
 lcd.clear();
 lcd.setCursor(0,0);lcd.print("Running");
 motor_running();}
 else if (localKey == 1)
 {lcd.clear();
  mode_select();}
 else
 {running_mode();}
}

void motor_running()
{ localKey = keypad.getKey();
 if (localKey==1){lcd.clear();mode_select();}
 else
 { time=millis();time_20=time+100;
  while (millis()<time_20)
   {number=analogRead(A5);
    total_t=total_t+number;count++;
    }
   mean_t=total_t/count;
   if (mean_t>midpt){over=1;}
   lcd.setCursor(0,0);
  maxpt_seperate();
```

```
   degree_det();
   total_t=0;total=0;count=0;
   motor_running();
  }
}

void maxpt_seperate()
{k=max_deg/10;
 l=0;
 if (over==1)
     { for (i=k;i<10;i++)
     {mean_diff[i]=mean_t-mean_deg[i];if (mean_diff[i]<0) {mean_diff[i]=-mean_diff[i];}}
     min_meandiff=mean_diff[k];l=k;
     for (i=k;i<10;i++)
       {
       if (mean_diff[i]<min_meandiff)
       {min_meandiff=mean_diff[i];l=i;}
       else {}
        }
    min_diffdeg=l*10;
    if ((mean_t-mean_deg[0])<min_meandiff) {over=0;}}
   else {
   for (i=k;i>=0;i--)
    {mean_diff[i]=mean_t-mean_deg[i];if (mean_diff[i]<0) {mean_diff[i]=-mean_diff[i];}}
   min_meandiff=mean_diff[k];l=k;
   for (i=k-1;i>=0;i--)
     {
       if (mean_diff[i]<min_meandiff)
       {min_meandiff=mean_diff[i];l=i;}
```

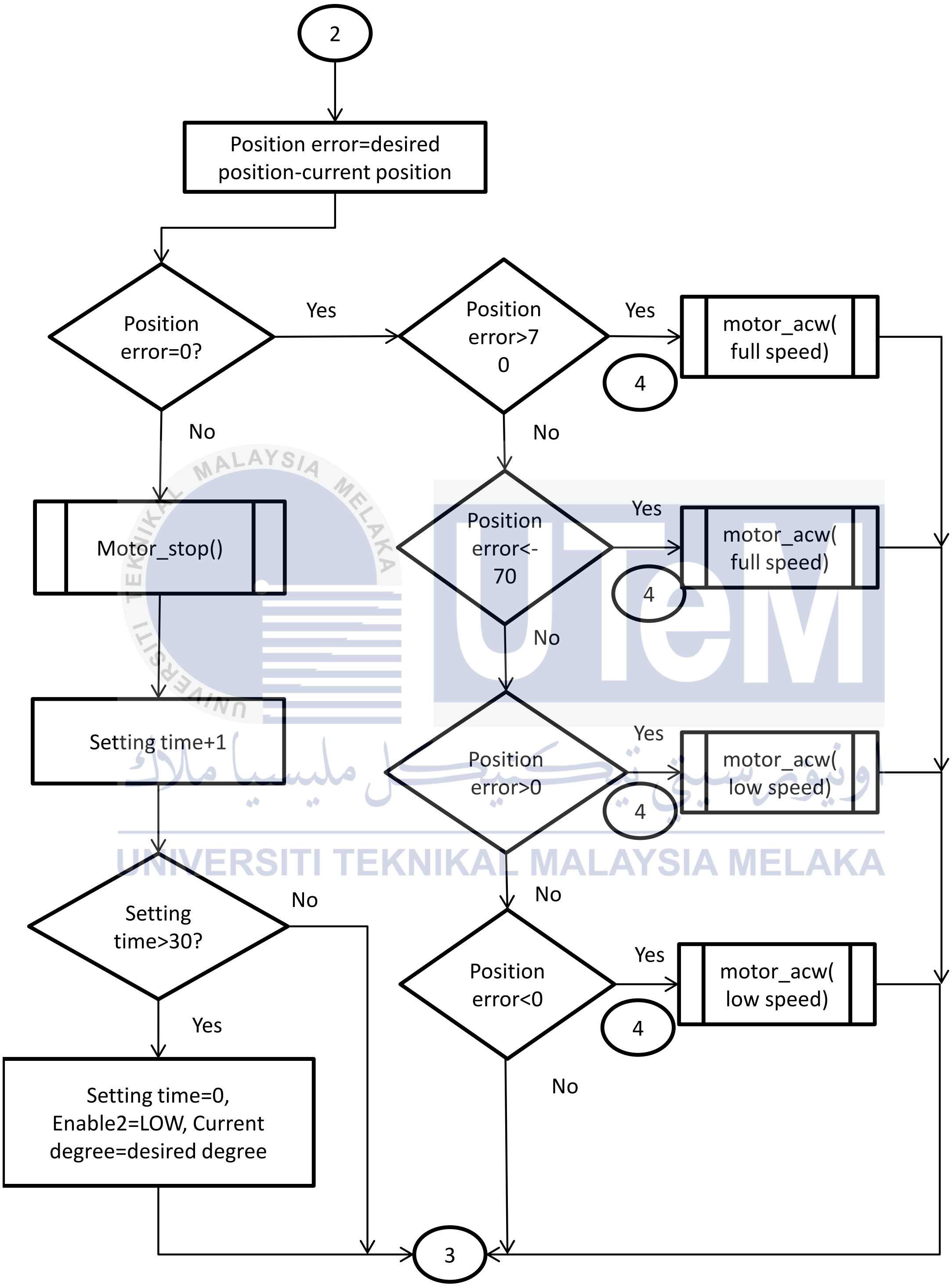
```
      else {}
     }
     min_diffdeg=l*10;}}


void degree_det()
 {
   if (min_diffdeg==previous_deg)
      {en3=en3+1;
       if (en3>=3)
       {en4=0;
lcd.clear();lcd.setCursor(0,0);lcd.print("standby");lcd.setCursor(0,1);lcd.print(mean_t);lcd.
setCursor(7,1);lcd.print(min_diffdeg);lcd.setCursor(10,1);lcd.print("deg");en3=0;}}
   else {
           en3=0
previous_deg=min_diffdeg;lcd.setCursor(0,0);lcd.print("loading");lcd.setCursor(0,1);lcd.pr
int(mean_t);    }
 }
```

Appendix F: Flow chart and coding for DC motor with encoder



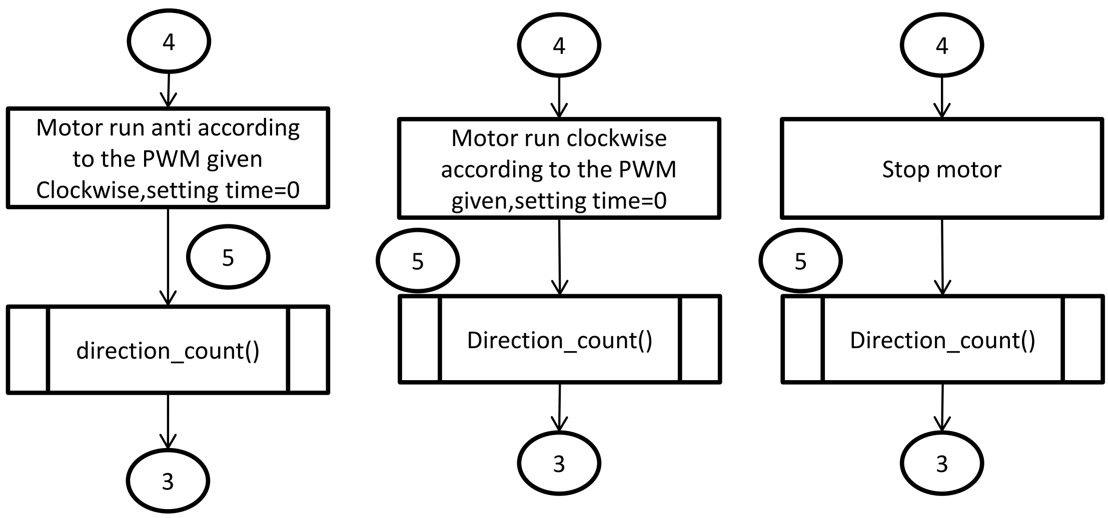

Angle()
1
Read the angle, calculate the angle need to rotate, encoder error and actual pulse required
Desired angle>current angle?
Yes
No
New position=current position + actual pulse required
New position=current position - actual pulse required
Replace the origin degree with desired degree
2

motor_run()

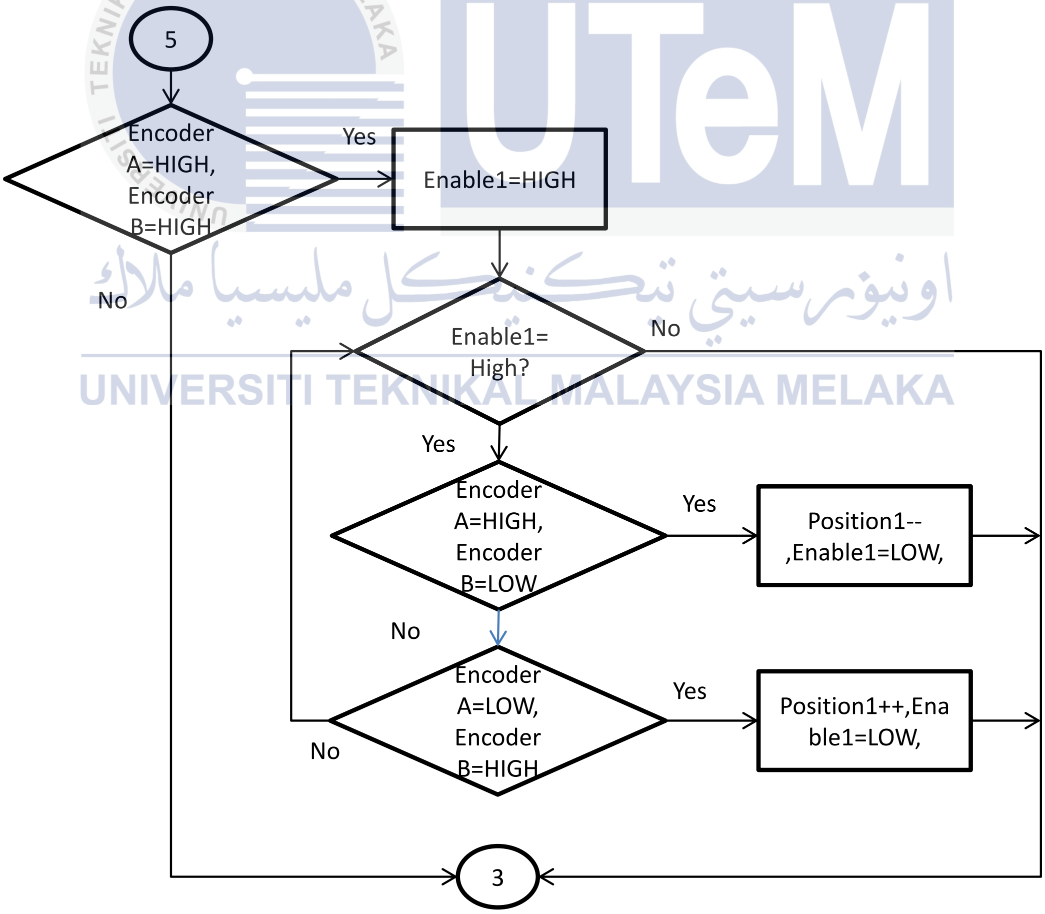


2
Position error=desired position-current position
Position error=0?
Yes
No
Position error>7 0
Yes
motor_acw(full speed)
4
No
Motor_stop()
Position error<- 70
Yes
motor_acw(full speed)
4
No
Setting time+1
Position error>0
Yes
motor_acw(low speed)
4
No
Setting time>30?
No
Position error<0
Yes
motor_acw(low speed)
4
Yes
Setting time=0, Enable2=LOW, Current degree=desired degree
No
3

motor_acw(PWM) motor_cw(PWM) motor_stop()



direction_count()



```
int desired_degree;
int initA;int initB;
int en1;int en2;
int pwm;
float e;
signed int Error0; signed int angle_diff; int origin_degree;
int origin_position; int desired_position;int countd;int count;int tset;long time;long time2;

void setup()
{ Serial.begin(57600);
  pinMode(4,INPUT);
  pinMode(5,INPUT);
  pinMode(6,OUTPUT);
  pinMode(7,OUTPUT);
  origin_degree=0; origin_position=0;
  angle(90);
  while(en2==HIGH)
  {motor_run();}
}

void loop()
{
}

 void angle(int desired_degree)
 {  angle_diff=desired_degree-origin_degree;
    if (angle_diff<0) {angle_diff=-angle_diff;}
    if (angle_diff==0){motor_run();}
```

```
  count=angle_diff*300/120;
  e=(0.0096*angle_diff*angle_diff)-(1.6653*angle_diff)+95.64;
  countd=count-(count*e/100);
 if (desired_degree>origin_degree){desired_position=countd+origin_position;}
 else if (desired_degree<origin_degree){desired_position=origin_position-countd;}
 en2=HIGH;origin_degree=desired_degree;
}

void motor_run()
{Error0=desired_position-origin_position;
 if (Error0==0)
   {motor_stop();tset++;
    if (tset>30)
    {en2=LOW;tset=0;}
    }
 else if (Error0>70)
 {motor_acw(255);}
 else if (Error0<-70)
 {motor_cw(255);}
 else
 {if (Error0>0){motor_acw(180);}
 else {motor_cw(180);}}
}

void direction_count()
{
 if ((digitalRead(4)==HIGH)&&(digitalRead(5)==HIGH))
    {  en1=HIGH;}
  while(en1==HIGH)
```

```
  {if ((digitalRead(4)==HIGH) && (digitalRead(5)==LOW))
    {origin_position--;en1=LOW;}
   else if ((digitalRead(4)==LOW) && (digitalRead(5)==HIGH))
   {origin_position++;en1=LOW;}}
}

void motor_acw(int PWM)
{analogWrite(3,PWM);
 digitalWrite(6,HIGH);
 digitalWrite(7,LOW);
  tset=0;direction_count();
}

 void motor_cw(int PWM)
 {analogWrite(3,PWM);
 digitalWrite(6,LOW);
 digitalWrite(7,HIGH);
 tset=0; direction_count();
 }

 void motor_stop()
 {analogWrite(3,0);
 digitalWrite(6,LOW);
 digitalWrite(7,LOW);
 direction_count();
 }
```