3 AXIS CNC MACHINE CONTROLLERS

MOHD AZRI BIN ABDUL RAHMAN

Report submitted in partial fulfillment of the requirements
for the award of Bachelor of Mechatronic Engineering

Faculty of Manufacturing Engineering
UNIVERSITI MALAYSIA PAHANG

JUNE 2013

SUPERVISOR’S DECLARATION

I hereby declare that I have checked this project and in my opinion, this project is adequate in terms of scope and quality for the award of the degree of Bachelor of Mechatronic Engineering.

Signature : ……………………………………….
Name of Supervisor :MR. KHAIRUL FIKRI BIN MUHAMAD
Position : ……………………………………….
Date : 19 JUN 2013

STUDENT'S DECLARATION

I hereby declare that the work in this project is my own except for quotations and summaries which have been duly acknowledged. The project has not been accepted for any degree and is not concurrently submitted for award of other degree.

Signature : ………………………………..
Name : MOHD AZRI B. ABDUL RAHMAN
I.D Number : FB09058
Date : 19 JUN 2013

DEDICATION

I specially dedicate to my beloved parents

and those who have guided

and motivated me for this project

ACKNOWLEDGEMENTS

First and foremost, the deepest sense of gratitude to the ALLAH, who guide and gave me the strength and ability to complete this final year project successfully. Infinite thanks I brace upon Him. I would like to express my sincere gratitude to my supervisor Mr. Khairul Fikri B. Muhamad for his continuous guidance, support and encouragement, which gave me huge inspiration in accomplishing this research. His practice of professional ethics and conducts which encourages me to become confident and competent person to work individually as well as in group with minimal supervision.

Besides that, I also took this opportunity to thank Mr. Suhaimi B. Puteh who helped me in preparing the measurement and electronic components. He also very supportive in exploring the features and utilization of components besides allows to use the laboratory until late night for research purpose. Not forgot to thank the panels which is Mr. Ismail B. Mohd Khairuddin who critics the outcome of this research besides providing some suggestions to improve the discussion and conclusion as well. I would also like to express my deepest appreciation to my parents Mr. Abdul Rahman B. Basaruddin and Mrs. Rafiah Bt. Abd. Aziz which whom always support and motivate me to complete this research and thesis. I also owe a depth of gratitude to all my friend who shared their knowledge and ideas that lead to the completion of this thesis. Thanks for the friendship and memories.

Finally, to individuals who has involved neither directly nor indirectly in succession of this research with thesis writing. Indeed I could never adequately express my indebtedness to all of them. Hope all of them stay continue support me and give confidence in my efforts in future.I am grateful to University Malaysia Pahang for their support during the period of this work. Thank you.

ABSTRACT

This study focuses on the programming of 3 Axis CNC machines where user can key in the data through Visual Basic software and as the results, CNC 3 axis machine will move according to the user data input. These project required 3 stepper motor as the motor is use to move the machine into X, Y and Z directions. The motor then will connect to the output of Darlington IC (ULN2803A). The inputs of Darlington IC are connecting to the parallel port that will connect to the computer. The Darlington IC also known as controller of this project. The most important part in this project is the programming. The movement is control by the program that will be written on C programming that was built in Visual Basic software. This program will determine how many rotation that the motor will rotate as the rotating motor will cause the distance of the machine move according to its axis. The programs also will activate the address on the parallel port so that parallel port will send the signals to the input controller. As a results, the motor will move and the machine also will go to its locations depends on the user data input on the Visual Basic. This proposal includes the introduction of the project, literature review and research methodology for further planning of investigation.

ABSTRAK

Kajian ini memberi tumpuan kepada pengaturcaraan 3 paksi CNC mesin di mana pengguna boleh memasukkan data melalui perisian Visual Basic dan sebagai keputusan, CNC mesin 3 paksi akan bergerak mengikut input data pengguna. Projek ini diperlukan 3 motor utama sebagai motor itu digunakan untuk menggerakkan mesin ke X, Y dan Z. Motor kemudian akan menyambung kepada output Darlington IC (ULN2803A). Input daripada Darlington IC sedang menghubungkan kepada port selari yang akan menyambung ke komputer. Darlington IC juga dikenali sebagai pengawal projek ini. Bahagian yang paling penting dalam projek ini adalah program ini. Pergerakan atau sebab motor berputar adalah kawalan oleh program yang akan ditulis pada pengaturcaraan C yang dibina dalam perisian Visual Basic. Program ini akan menentukan berapa banyak putaran motor akan berputar sebagai motor berputar akan menyebabkan jarak langkah mesin mengikut paksinya. Program-program juga akan mengaktifkan alamat di pelabuhan selari supaya port selari akan menghantar isyarat kepada pengawal input. Sebagai keputusan, motor akan bergerak dan mesin juga akan pergi ke lokasi bergantung kepada input data pengguna pada Visual Basic. Cadangan ini termasuk pengenalan projek, kajian literatur dan metodologi penyelidikan untuk perancangan lagi penyiasatan.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

π : 3.142

° : Degree

Pps : Pulse/second

LIST OF ABBREVIATIONS

GUI	: Graphical User Interface
VB	: Visual Basic
CNC	: Computer numerical control
FYP	: Final Year Project
FKP	: Faculty of Manufacturing
UMP	: University Malaysia Pahang
ULN	: Upper Limits of Normal
IC	: Integrated Circuit
LED	: Light Emitting Diode
WPF	: World Population Foundation
NC	: Numerical Control
ISO	: International Standard Operation

CHAPTER 1

INTRODUCTION

1.1 PROJECT BACKGROUND

Computer Numerical Control or CNC machine is a conventionally machine where an operator decides and adjusts various machines parameters like feed, depth of cut and etc. depending on type of job, and controls the slide movements by hand. It also is a specialized and versatile form of a Soft Automation and its applications cover many kinds, although it was initially developed to control the motion and operation of machine tools. A CNC machine takes codes from a computer and converts the code using software into electrical signals. The signals from the computer are then used to control motors. Since the motors can turn very small amounts the machine is able to move in highly precise movements over and over again. (Chana Raksiri and Manukid Parnichkun, 2004)

In University Malaysia Pahang (UMP), especially at laboratory in Manufacturing Engineering Faculty, there are a lot of machines can use CNC 3 axis programs for examples drilling machine and cutting machine, but those machine not yet using the technology. So that, staff and students has to do it manually, like marking and do some calculation. It wills easily the lecturers, staff and student works by added programs into the machine. With the implement of CNC many people can shorten their task and works. For the improvements purpose, the machine has to be added with CNC programs and converts the physicals machine a little bit.

1.2 PROBLEM STATEMENT

The FKP laboratory currently have problem with the small product to be machining using CNC machine. Therefore whether the staff and students like it or not, they don't have any choices and have to use CNC machine that are already have at the laboratory. The problem is, the machine is too big for the tiny or small pieces of product. The aimed of this project is to build a small CNC machine purposely for a small product to be machining.

1.3 RESEARCH OBJECTIVES

There are several objectives to be achieved, which are:

1. To design and develop CNC 3 axis machine.
2. To verify the controller validity.

1.4 RESEARCH QUESTIONS

There are several questions have to be solved, which are:

1. How does the motor move
2. How to do the algorithm so that the motor will move to the data input by user.

1.5 PROJECT SCOPE

This project is developed to FKP machining Laboratory. This machine will be using by staff and student of University Malaysia Pahang (UMP) Campus Pekan. The purpose of this project is to study the movement of CNC machine when user key in the data through Visual Basic software whether in X, Y and Z direction. This study also will use stepper motor connect with parallel port. And for the controller, ULN 2803 IC which is it is known as Darlington will be use.

As the results the movement or rotation of motor will be produced based on the program algorithm written in C programming. The results then will be analyzed using Visual Basic software which is it can control the movement of the motor thru the programming written on it.

1.6 DEFINITION OF TERM

CNC is a short name of Computer Numerical Control. CNC is the motions of a machine tool are controlled by means of a prepared program containing coded alphanumeric data. CNC can control the motions of the workpiece or tool, the input parameters such as feed, depth of cut, speed, and the functions such as turning spindle on/off, turning coolant on/off. While axis is known as a straight line about which a body or geometric object rotates or may be conceived to rotate. In mathematics, axis is called an unlimited line, half-line, or line segment serving to orient a space or a geometric object, especially a line about which the object is symmetric or is also known as a reference line from which distances or angles are measured in a coordinate system. (www.technologystudent.com)

Meanwhile axis in CNC is a little bit different. Each major line of the number scale is called an axis. This old principle, when applied to CNC programming, means that at least two axes- two number scales – will be used. This is the mathematical definition of an axis. An axis also is a straight line passing through the center of a plane or a solid figure, around which the parts are symmetrically arranged. The definition can be enhanced by a statement that an axis can also be a line of reference. In CNC programming, an axis is used as a reference all the time. (Sotiris L.Omirou, AndreasC.Nearchou, 2007)

1.7 EXPECTED OUTCOMES

The expected result for the project is well work and able to use by FKP machining laboratory in UMP especially for staff and student. The systems are able to give the staff and students more easily to machining the small or tiny product precisely. The student also can learn more using both big and small CNC machine to produce output, which is the product. This system is integrated with software and hardware through Visual Basic program. The system is friendly user and can be using both staff and student as well.

1.8 SIGNIFICANT OF STUDY

After this project has been done, the Faculty of Manufacturing staff and students will much easier do their job to produce small machining product by using this small type of CNC machine more accurate and precise. With this small CNC machine also, it can avoid the unexpected problem when fabricate a small product. All of the knowledge could be apply either in programming, mathematical function or both of it. In addition, when something knowledge apply into several application it would give us a lot of good implications such as to create a self-confidence .Other than that, by invention may give a lots of attraction to ourselves compared just learning the theoretical of some knowledge. Hence, it also creates our mind to think positively and become more creative. There are several method that is important on this project:

1. C programming
2. Mathematical algorithm

Hopefully this idea can be implemented in the long term to facilitate better and standard and provide effective testing infrastructure for achieving better results in future.

1.9 ORGANIZATION OF THESIS

This thesis consists of five chapters. Chapter 1 presents introduction. Literature review on article and journal are on the Chapter 2. Meanwhile Chapter 3 presenting the methodology of the project. Chapter 4 will discuss about the results that are taking from the experiment. Finally in Chapter 5 concluded the project and provide with recommendation for the project.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

In this chapter, the findings and previous studies regarding to this project title will be presented. Most of the finding materials are taken based on the journals, article and also from the books. From the findings, the general information regarding to the project can be gathered more easily before the experiment began. In section 2.2, the study on the type of programming will be explained. In section 2.3, part of CNC has been told. Lastly in section 2.4, the advantages and disadvantages will be discussed.

2.2 BACKGROUND OF STUDY - WHAT IS CNC?

CNC stands for Computer Numerical Control and has been around since the early 1950's in the United States, it is use by the US Air Force by metalworking machine tool builders. Before this, it was called NC, for Numerical Control. Its basic method of controlling movement. In the early 1970's computers were introduced to these controls, hence the name changes to Computer Numerical Control. It was a major advance in the ability of machines to faithfully reproduce complex part machining steps more accurately without human intervention or variability. CNC machine uses a stream of digital information (code) from a computer to move motors and other positioning systems in order to guide a spindle over raw material.

A CNC machine uses mathematics and coordinate systems to understand and process information about what to move, to where, and how fast. Most CNC machines are able to move in three controlled directions at once. These directions are called axes. The axes are given simple names such as X, Y and Z (based on the Cartesian coordinate system). The X axis is always the longest distance a machine or a part of a machine must travel. X may be the movement from front to back, Y the movement from left to right, and the Z is almost always vertical movement (normally the spindle's positioning movement up and down). A CNC machine must be able to communicate with itself to operate. A computer numeric control unit sends position commands to motors. The motors must talk back to the control that, indeed, they have acted correctly to move the machine a given distance. The ability of CNC machines to move in three (or more) directions at once allows them to create almost any desired pattern or shape. All of this processing happens very fast. While people in most walks of life have never heard of this term, CNC has touched almost every form of manufacturing process in one way or another. Manufacturing engineering is the most field that were dealing with CNC on regular basis.

The earliest CNC machines received code instructions through hard-wired controllers, which meant that the programming format could not be altered. However, later models were programmed via mainframe cables and floppy disks, which permitted variations in programming. At present, most CNC machines are tied into a network of computers and receive operating and tooling instructions via a software file containing the ".NC" extension. Today Modern CNC machines are also capable of running overnight or for several days without human supervision. In fact, CNC machines are now so sophisticated that they can dial the chief operator’s cell phone to notify him or her when a tool part breaks, and still complete other parts of the program. These features make it possible to produce thousands of parts while the machine shop is closed or while the operator is performing other tasks.

2.3 DEMANDS ON CNC

Nowadays there are many technologies growth. With this growing competitive, there are many industries demanded want to use new, cheap, high quality and faster's machine. In the industries at this millenniums century, especially in manufacturing field, CNC machine are the major machines use to produce the output according to its advantages (precise, faster, etc.) thus many type of CNC machine are build to do multi jobs like cutting, roughing, pocket, slotting, drilling, threading and etc. But CNC developer industries always build large scale of CNC machine because of its purpose. Today, there are still not yet developer build CNC machine in small scale, simple and easy to use. Because of this, these projects are made. Demands on the CNC nowadays also this idea are produce. Usually the small items are fabricates and produce by using simple machine and techniques like molding, injection molding, blowing and etc. but with the demand on the CNC and demand from the user, these tiny little things also now are produce using CNC machine as there are many advantages using these CNC machine in terms of its quality and results.

2.4 TYPE OF PROGRAM

2.4.1 Parametric Program

Parametric programming, as a feature of modern Computer Numerical Control (CNC) machines, has the potential to bring higher efficiency to manufacturing industries. The application of parametric programming to CNC operations is possible in several ways. These include generating a single CNC program for parts with similar design, inventing macros for machining custom design features, and developing subprograms for a group of parts that are not similar in design but require similar machining operations. In all these applications, parametric programming can significantly reduce the part programming time and effort which in turn leads to shorter throughput and product development times. These applications particularly fit group technology manufacturing in which similar parts are grouped into part families and then processed by a number of machine tools within a cell or by a single multi-tasking machining center. The two common approaches in group technology are grouping based on design similarity, and grouping based on similar machining requirements. Parametric programming can be applied to part families formed by either of the two grouping approaches, as illustrated in this study.

As one of the less frequently utilized features of CNC machines, parametric programming has the potential to increase the efficiency of CNC operations. This feature is particularly beneficial to companies with group technology manufacturing where parts with similar design or operational requirements are processed within a machine cell. Parametric programming is a G/M code programming (see figure 2.1) in which axis position (x,y,z, a, etc.), feed and speed functions can be specified by a parametric expression. Most CNC machines provide a parametric programming feature that allows the user to load a part program or a subprogram to the machine controller. The part program is then called up whenever a similar part is machined or similar operation has to be performed on one or more parts. For example, several parts may require machining a pocket with different sizes or drilling a hole circle pattern of various diameters and number of holes. A single parametric subprogram can be called up from the main program to machine these part features. Upon loading the main program, the values of parameters associated with a feature such as length, width and depth of a pocket are entered; then, these val6es are antonmtically transferred from the main program to the parametric subprogram. This approach eliminates the redundant codes in part program and reduces the size of the program and programming time. (Manocher Djassemi, 1998)

```
O100
P10=.75 (DIAMETER OF END-MILL)
        (CUTTER)
P11=2.5 (X POSITION OF HOLE CENTER)
P12=1.  (Y POSITION OF HOLE CENTER)
P13=.5  (DEPTH OF HOLE)
P14=1.5          (DIAMETER OF HOLE)
P15=20           (FEEDRATE IN IPM)
P16=1000         (SPINDLE SPEED RPM)
G92 X0 Y0 Z1     (SET INITIAL TOOL )
                 (POSITION)
S[P16] M03
G00 X[P11]  Y[P12]  (RAPID TO HOLE )
                    (CENTER)
M06 T01
G43 H1 Z.1               (RAPID TO Z=.1")
G01 Z[-P13]  F[P15/2]
Y[P12+[P14/2]-[P10/2]]   F[P15]
G02 J[-[P14/2]-[P10/2]]
G01  Y[P12]
G00 Z.1
M30
```

Figure 2.1: Parametric program for machining a hole

Source : Manocher Djassemi, 1998

2.4.2 An Epitrochoidal Pocket

An epitrochoid is a curve traced by a point P attached to a circle of radius r rolling around the outside of a fixed circle of radius R, where the point is at distance h from the center C of the exterior circle (see figure 2.2). Usually A machining strategy for milling a particular set of pockets with epitrochoidal boundary is proposed. The method is suitable to be integrated into the controller of a CNC milling machine and is particularly useful for machining chambers of rotary internal combustion engines. Motion generation is achieved by an algorithm which utilizes real-time CNC interpolation providing the highest possible accuracy, of which the milling machine is capable. The surface quality is controlled by applying roughing and finishing passes. The whole machining task can be programmed in a single block of the part program.

Canned cycles provide a programming method of a CNC machine to accomplish repetitive machining operations using the G/M code language. Drilling, counter-boring, peck drilling, pocket or slot machining are all examples of standard canned cycles. However, the standard canned cycles are limited in number and capability, being unable to accommodate the increasing needs of applications with complex geometries. The machining of an epitrochoidal pocket constitutes a characteristic case that the standard canned cycles cannot handle. Despite its important machining applications, modern CNC systems lack a similar dedicated canned cycle. Pocket milling is a common machining operation used for removing the material inside a closed boundary on the flat surface of a workpiece to a fixed depth (see figure 2.3). Rectangular and circular forms of pockets are the standard types of pockets found as canned cycles in the programming capabilities of modern CNC systems. Today, epitrochoids can be found in important mechanical parts such as gears with epitrochoidal tooth profile, cams and epitrochoidal-shaped housings for rotary internal combustion engines and rotary piston pumps. (Sotiris L.Omiroua, AndreasC.Nearchoub,2007)

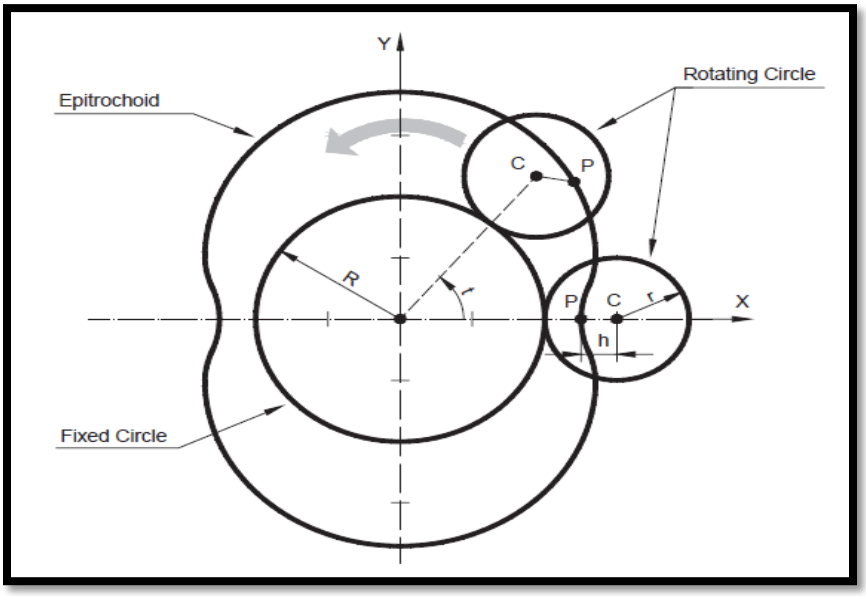


Figure 2.2: Definition of an epitrochoid curve

Soure : Sotiris L.Omiroua, AndreasC.Nearchoub,2007

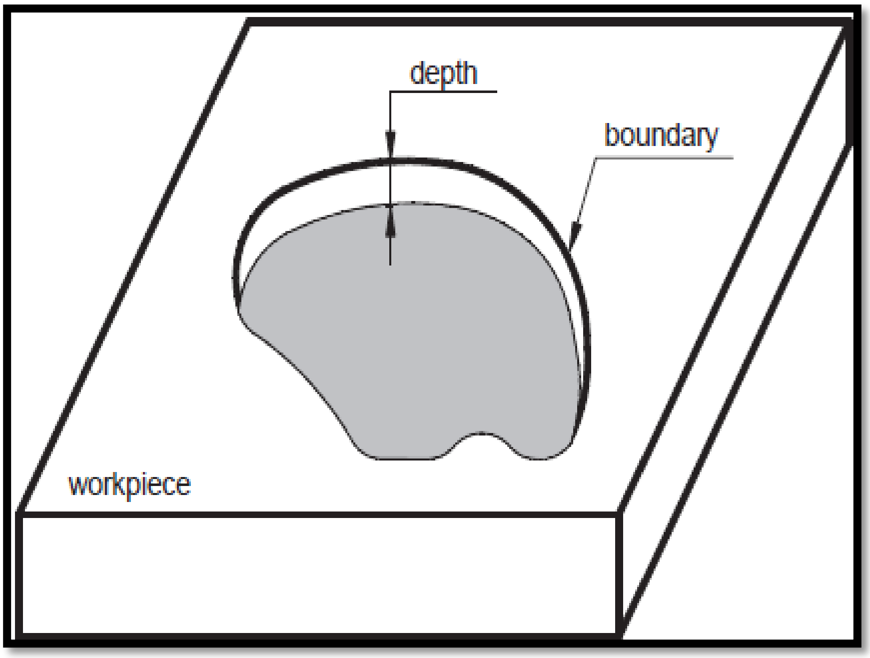


Figure 2.3: Pocket Illustration

Source : Sotiris L.Omiroua, AndreasC.Nearchoub,2007

2.4.3 Station Software Program for CNC Profile Cutting

A programming software station is a graphic workstation with a special-purpose graphic editor that allows the user to draw, edit and nest profiles in any configuration on screen. The actions of the user are simultaneously translated into commands in the desired CNC language, such as ISO NC language. At the end of a session, the program is saved to disk and is ready to be transmitted to the CNC controller. The programming station can be used to both generate and verify NC data graphically. The platform for a programming station can be any suitable computer or terminal capable of displaying reasonably good graphics. The IBM PC and compatibles can serve adequately as platforms for developing and implementing programming station software. Programming stations are used to automate program generation and achieve tool-path verification in CNC profile cutting.

With today's software tools, it is well worth the effort to develop one's own programming station software rather than to depend on expensive or unsuitable alternatives. The advantages of applying programming stations can be summarized as follows:

(i) The visual feedback involved in generating CNC programs on programming stations eliminates many types of mistakes possible with manual coding.

(ii) Program generation and verification are contained in the same package. A variety of tools (such as tool-path generation, dimension check etc.) are available for verification.

(iii) Program coding time is drastically reduced because of a variety of productivity features available on programming stations.

(iv) Profiles can be nested better, with resultant material savings.

(v) Programming can be done entirely off-line, thus saving valuable machine time

Specifically designed CAM tools, incorporating special facilities, are necessary to produce efficient CNC code. Such tools, called programming stations, can be developed within a man-year by an experienced programmer using software tools and techniques available today.

2.4.4 Workshop Programming

The programming language developed for this purpose includes special machining cycles which can be used on this of the engineering drawing. A special conversation program is a further facilitation of the part-programming, which can even be implemented while the machine tool is in operation.

The term "workshop programming system" is understood to mean a system entailing a functional link between the programming and the performance of a machining task. Workshop programming systems are characterized by the fact that program inputs can be carried out at or directly into the machine on the basis of engineering dimensions without the need for the prior computation of coordinates. This means that the program input is based on workpiece data rather than on tool movements. Depending on the sophistication of the part spectrum there is a choice between playback method, teach-in method, a possibly extended form of control data programming, an NC programming language, or part-family programming. CNC controls with an integrated NC programming system are workshop programming systems characterized by the structural break-down into programming and control (see figure 2.4).This clear division of labor rakes it possible to carry out programming simultaneously with the machine tool in operation. Both systems do constitute a functional unit, but they can run more or less independently **of** each other. (G. Spur (1), H. Meier, 1979) (G. Spur (1), H. Meier, 1979)

The workshop programming system described here allows the operate to prepare in a straightforward, rapid and reliable way all the part-programs in conversational mode at the computer screen, in *so* doing using a programming language adapted to the user. (G. Spur (1), H. Meier, 1979)

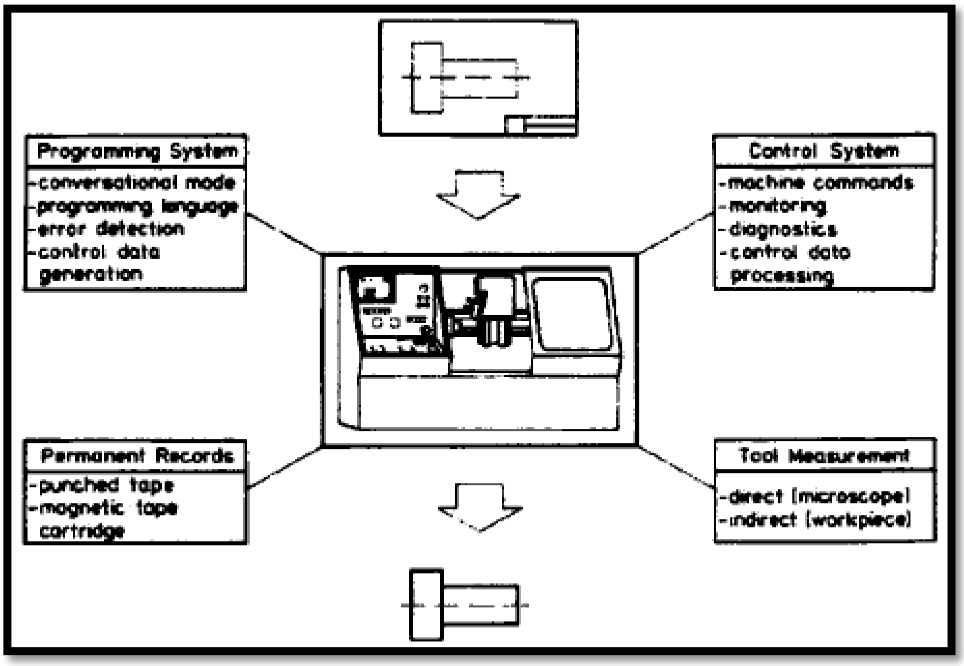


Figure 2.3: Functions of a Workshop Programming system

Source : G. Spur and H. Meier, 1979

2.4.5 Comparison between the Programs

After the research has been read trough the journals that had been written by the professionalism, the summary from its method can be determined to see the pros and cons in the research. The comparisons also have been made on the table 2.4 to see clearly the target and focus on their research. (Seungkil Son, Taejung Kim, Sanjay E. Sarma and Alexander Slocum, 2009)

Table 2.1: The advantages and disadvantages of each program

Programs	Advantages	Disadvantages
Parametric Program	Produce a lot of similar models by one program. Also, can do multitasking jobs in one time.	Suitable for the industry only where in the industry they have to produce many models with a very similar design and spec.
An Epitrochoidal Pocket Program	Very useful to do a complex pockets milling like in the combustion engine. Also, it useful for the standard canned cycled cannot do.	Suitable for the pocket canned cycle only.
Station Software Program	Allows the user to draw, edit and nest profiles in any configuration on screen. Then it can be translated into commands in the desired CNC language, such as ISO NC language	Need the station software to do the programs part before it can be translated to the CNC machine, otherwise, it can't be use.
Workshop Program	With workshop programming, cutter lengths are measured on the machine tool itself. It also rapid and very precise.	Program input is based on workpiece data rather than on tool movements
Visual Basic	Simple, easy to programs and use. This type of	It is not suitable to use for produce a product as the

	program also very familiar among the programmers.	functions is only for a simple canned cycle.

2.5 PART OF CNC

In Industry it is not efficient or profitable to make everyday products by hand. On a CNC machine it is possible to make hundreds or even thousands of the same items in a day. First a design is drawn using design software, and then it is processed by the computer and manufactured using the CNC machine. Below, there are basic parts of every CNC machine must have (see figure 2.5). (S.T. Newman and A. Nassehi, 2008)

1. The VICE - This holds the material to be cut or shaped. Material must be held securely otherwise it may 'fly' out of the vice when the CNC begins to machine. Normally the vice will be like a clamp that holds the material in the correct position.

2. The GUARD - The guard protects the person using the CNC. When the CNC is machining the material small pieces can be 'shoot' off the material at high speed. This could be dangerous if a piece hit the person operating the machine. The guard completely encloses the dangerous areas of the CNC.

3. The CHUCK - This holds the material that is to be shaped. The material must be placed in it very carefully so that when the CNC is working the material is not thrown out at high speed.

4. The MOTOR - The motor is enclosed inside the machine. This is the part that rotates the chuck at high speed.

5. The LATHE BED - The base of the machine. Usually a CNC is bolted down so that it cannot move through the vibration of the machine when it is working.

6. The CUTTING TOOL - This is usually made from high quality steel and it is the part that actually cuts the material to be shaped.

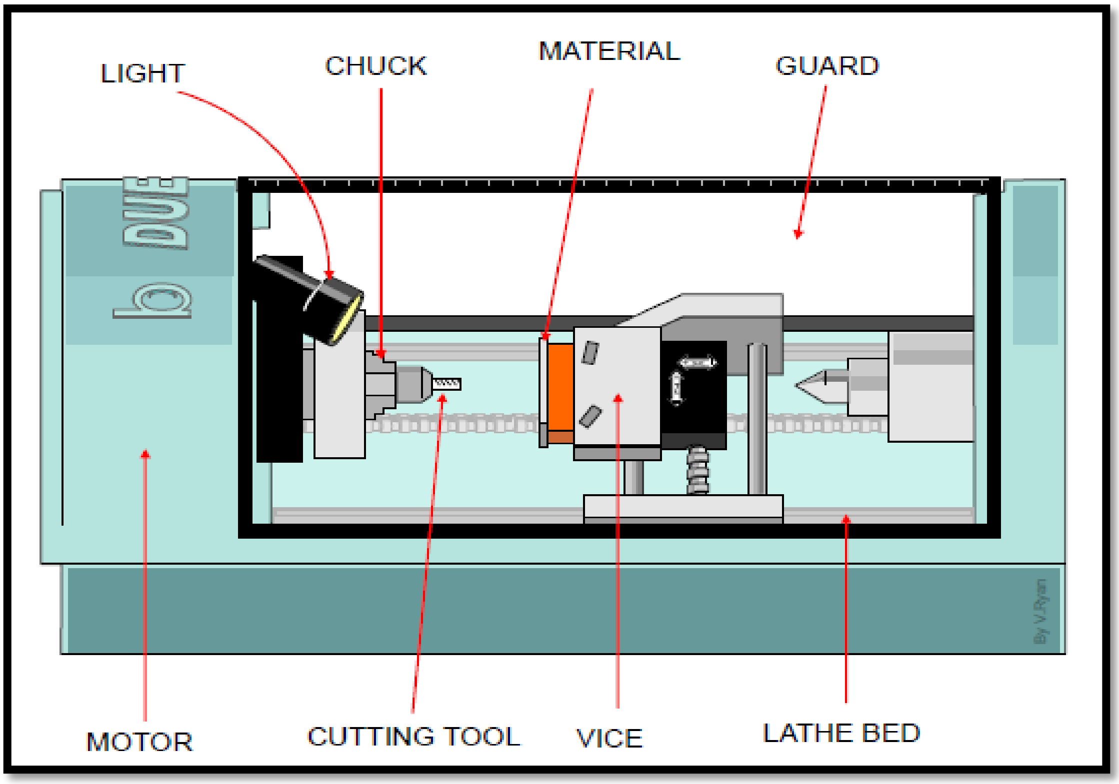


Figure 2.5: Parts of CNC

Source : Maciej Staniak and Cezary Zieli, 2010

2.6 BENEFITS AND DRAWBACK OF CNC MACHINES

CNC Computer Numerical Control machines are widely used in manufacturing industry. Traditional machines such as vertical millers, centre lathes, shaping machines, routers etc. operated by a trained engineer, in many cases, been replaced by computer control machines.

Table 2.2: Advantages and Disadvantages of CNC machine

Advantage	Disadvantage
CNC machines can be used continuously 24 hours a day, 365 days a year and only need to be switched off for occasional maintenance.	CNC machines are more expensive than manually operated machines, although costs are slowly coming down.
CNC machines are programmed with a design which can then be manufactured hundreds or even thousands of times. Each manufactured product will be exactly the same.	The CNC machine operator only needs basic training and skills, enough to supervise several machines. In years gone by, engineers needed years of training to operate centre lathes, milling machines and other manually operated machines. This means many of the old skills are been lost.
Less skilled/trained people can operate CNCs unlike manual lathes / milling machines etc. which need skilled engineers.	Fewer workers are required to operate CNC machines compared to manually operated machines. Investment in CNC machines can lead to unemployment.
CNC machines can be updated by improving the software used to drive the machines.	Many countries no longer teach pupils / students how to use manually operated lathes / milling machines etc. Pupils / students no longer develop the detailed skills required by engineers of the past. These include mathematical and
Training in the use of CNCs is available	

through the use of 'virtual software'. This is software that allows the operator to practice using the CNC machine on the screen of a computer. The software is similar to a computer game.	engineering skills.
CNC machines can be programmed by advanced design software such as Visual Basic, enabling the manufacture of products that cannot be made by manual machines, even those used by skilled designers / engineers.	
Modern design software allows the designer to simulate the manufacture of his/her idea. There is no need to make a prototype or a model. This saves time and money.	
One person can supervise many CNC machines as once they are programmed they can usually be left to work by themselves. Sometimes only the cutting tools need replacing occasionally.	

A skilled engineer can make the same component many times. However, if each component is carefully studied, each one will vary slightly. A CNC machine will manufacture each component as an exact match.	

Source : M. Kovacic, M. Brezocnik, I. Pahole, J. Balic and B. Kecelj,2005

2.7 CONCLUSION

3 axis CNC machine in a small scale size are suitable for all places like in the industries, universities and companies as well. It can generate a product in a small scale size that have to fabricate using CNC machine. With this machine around, all the works will become more easily. Demand on CNC nowadays make the industries have to use CNC machine even for a small scale product as the user now very demanding on what they want.

CHAPTER 3

METHODOLOGY

3.1 INTRODUCTION

The main objective of this project is to design and develop the 3 axis CNC machine. This project is aim purposely for a small scale product that use CNC machine which is commonly use in the industries specifically for a small product. This section will present about the methodology design and development of CNC 3 axis machine to fulfill the main objective.

Methodology is a guideline for a developer to plan the structure and control development process. This chapter will continue to discuss the method for the design and development of the 3 axis CNC machine. Design and development of this CNC 3 axis machine consist of several parts which is the mechanical parts, electrical parts and programming. Each part will be design stage by stage according to its priority. In this project, Nanotech Hollow shaft stepper motor will be use. While Darlington IC is the controller for the machine. The data will be programmed using C language on Microsoft Visual Basic and the input will be sent to the driver and the output will be seen on the rotation of the motor.

For the computer control, C++ and C programming (C language) will be used as the fundamental to build the complex algorithm as the move or rotation of the motor. The data will be program in the C language on Visual Basic and the output will be sent to the motor so that it will move based on the user data input. Software integration required to combine signals from the sensors and the controller.

All these parts will be combined together as every parts of the machine is relating between each other. The output of the development CNC machine will be taken and analyze to get better results and precise. Further investigation will be established to implants the most optimum solution for the CNC 3 axis machine.

3.2 METHODOLOGY FLOWCHART

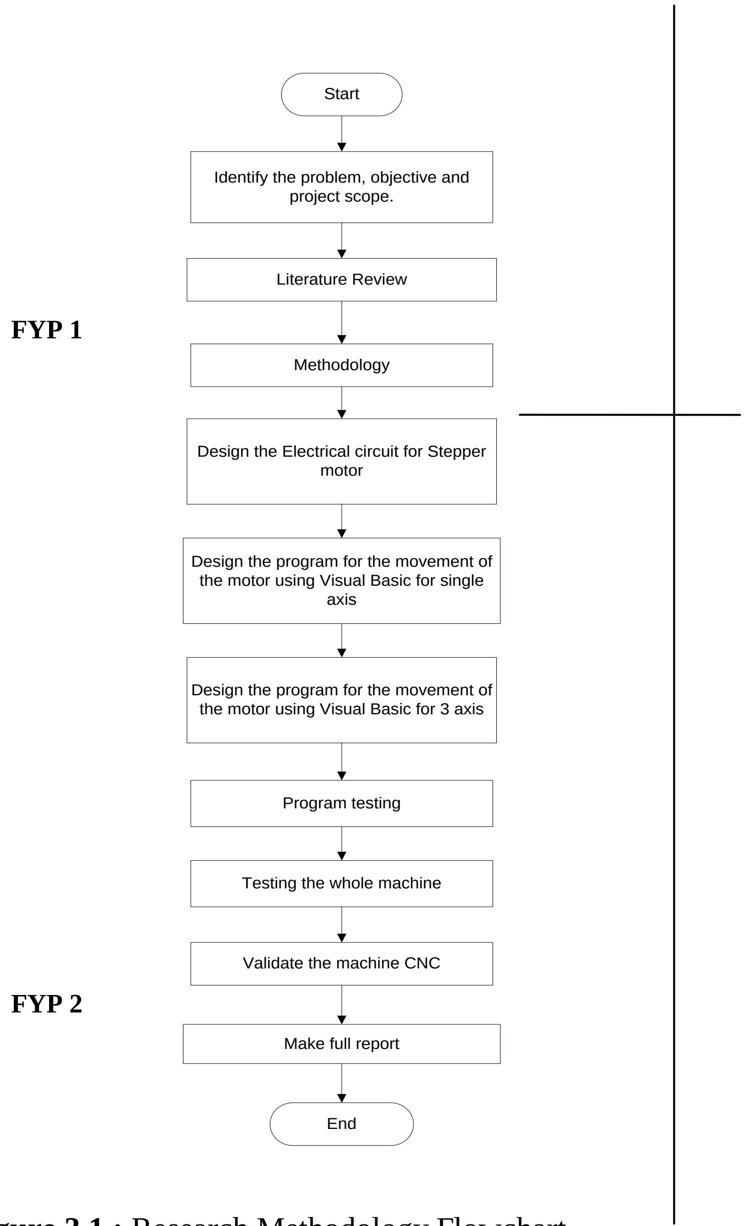


Figure 3.1 : Research Methodology Flowchart

3.3 DESIGN METHODOLOGY FOR CNC 3 AXIS MACHINE

For this CNC 3 axis machine, the operation is all same like the other CNC machines. First, user will key in the data into user interface that have been made on the computer, and then data will be sent to the controller. The controller then read the data and sent it to the stepper motor. And the result, the motor will move according to the user input data. Controller will sent the data to the motor by using number of bit. Whereby these number of bit will determine the rotation of the motor. The master mind of this project is depending on the program in the Visual Basic.

3.4 STEPPER MOTOR

For this project Hollow shaft stepper motors made from Nanotec, (Figure 3.2) will be used, with 4.2 A Current per winding, 0.92Nm holding torque, 0.36 Ohm resistance per winding 1.10mH inductance per winding, and 230gcm2 rotor moment of inertia. With this impressive specification of stepper motor, it is very suitable for lightweight mechanism such as CNC 3 Axis machine as the machine is for small scale products and required high torque to generate and rotate the motor. With the weight only 0.6 kg, this stepper motor also offers high resolution for high precision speed and positioning. It also produces the highest possible torque with the compact size which is only 56mm length, 56mm width and 51mm tall. In this project, 3 stepper motor will be use as to move in the X, Y and Z direction.

The connection of stepper motor to the Darlington is shown in figure 3.4 where phase A, B, C and D are all the wire of stepper motor and P4, P5, P6 and P7 are the address from parallel port. As shown in Figure 3.4, stepper motors are connected to the output of the Darlington’s while address from parallel port are connected to the input of the Darlington IC.

Figure 3.2: DC Gear Motor

Source : https://www.motionusa.com/node/2970

3.4.1 Wiring Stepper motor

In this project, 6 wire stepper with 4 phase unipolar motor were use. These motors contain 6 wires where 2 wires are common. To determine the common wires, tap every wire with multimeter. The resistance was measure. The highest value of resistance is the common wire. On the six wires also the were two groups of three wires in which all three wires are electrically connected to each other. These wires are connected to the same coil, one in the center of the coil, two at their ends. Accordingly, two of the possible pairings have a lower resistance; one pair has a higher one. Take the pair with the higher resistance. Figure 3.4 shows the wiring connection for the stepper motor with 6 wires connections.

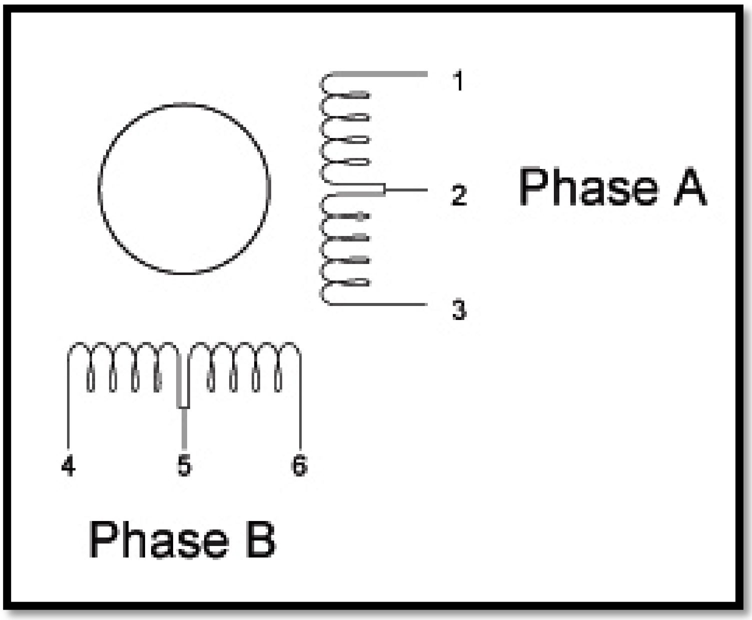


Figure 3.3 : Stepper motor wiring connection

Source : http://www.linengineering.com/resources/wiring_connections.aspx

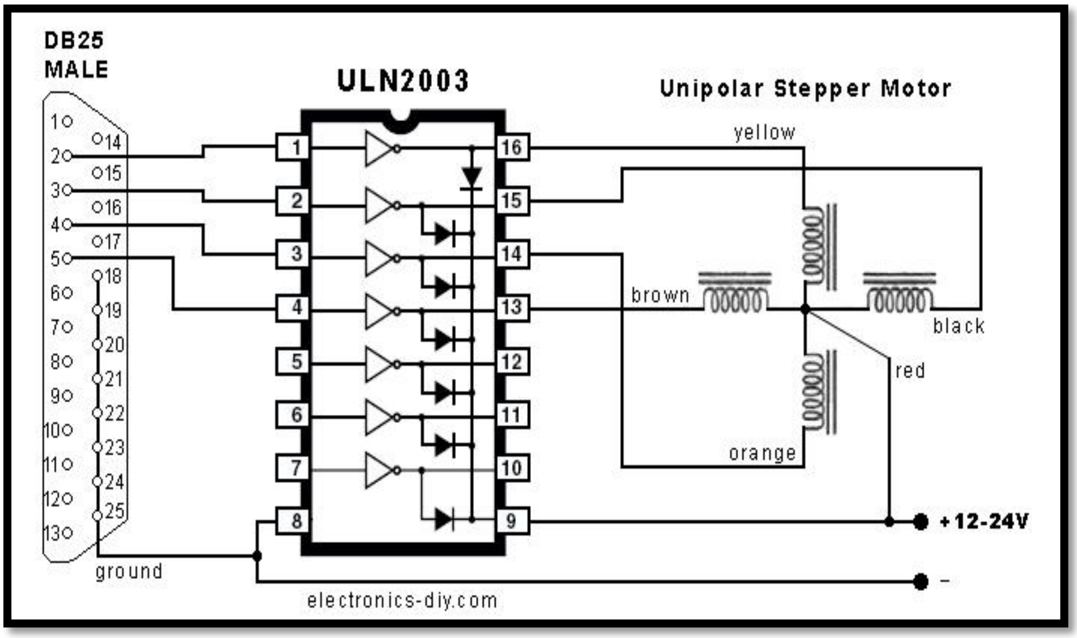


Figure 3.4: Stepper motor connection

Source : http://electronics-diy.com/stepper_motors.php

3.5 CONTROLLER

For the controller, ULN2803A (Figure 3.5) will be use, this integrated chip or IC consist of 18 pins whereby pin 1 to 8 is for the input and from pin 11 to 18 for its output. However pin 10 is common pin for all input and lastly pin 9 for the common pin for the IC (Figure 3.6). A Darlington's is an IC where 2 transistors pairs up together, in such a way that the total gain is the product of then individual transistor gain. In other word, Darlington can work with high current whereby it is very suitable to control motor using microcontrollers. The encoder from motor will be connected to the output of Darlington transistor, and the IC will be mounted on PCB board and the input pin will be connected to the parallel port.

For this project also, 3 Darlington will be used as the Darlington is to connect the three motor to the parallel port thru Darlington.

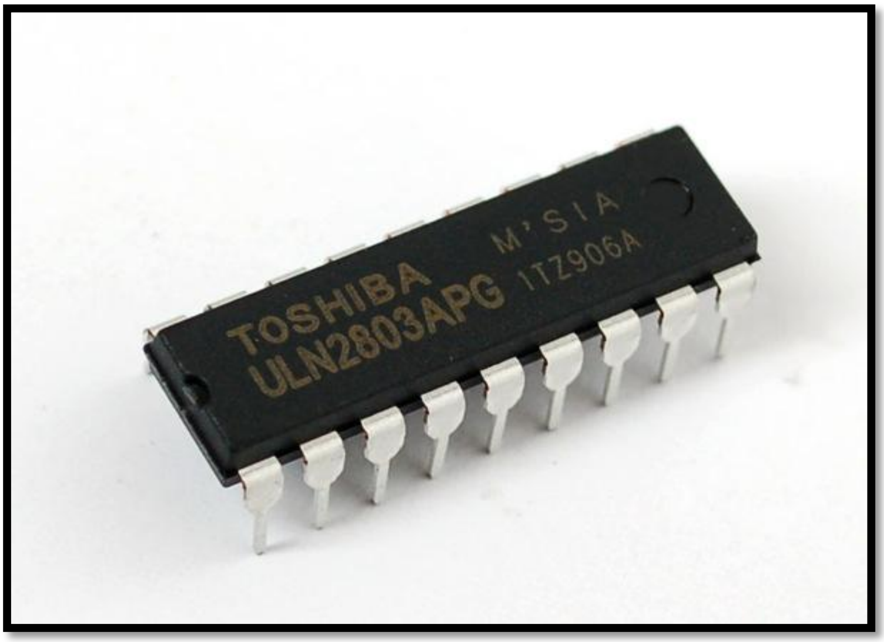


Figure 3.5: Darlington IC

Source : (http://www.alpha-crucis.com/en/power-components/4045-8-channel-darlington-driver-solenoid-unipolar-stepper-uln2803a-.html)

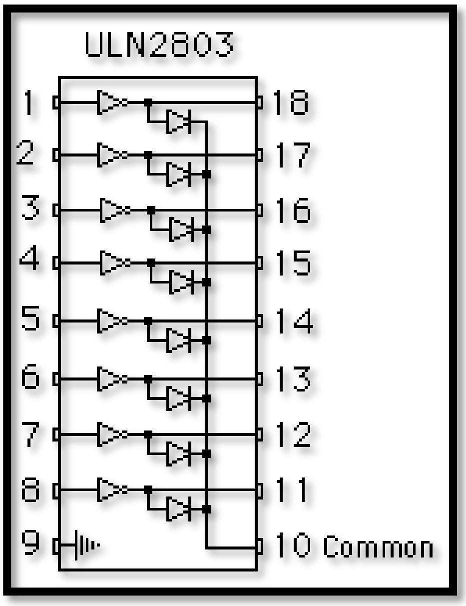


Figure 3.6: Darlington pin Connection

Source : http://freedatasheets.com/datasheetblog

3.6 SOFTWARE DEVELOPMENT

Programming is the main part or master mind of this CNC 3 axis machine project. The programming will be written to determine the stepper motor rotation of the machine. All the motor rotation will be determine by using C language that are written in the Visual Basic software. C language is use because it can be read, write and rewrite so that if anything happened it can be corrected. This program is part of the completion of the project. It will do complex algorithm so that the motor will rotate based on the pulse and bit given. User will key in the data on the C language than the data will be send to the driver, from driver, pulse and bit then send to the motor and the motor will start to move based on the data user input on the Visual Basic. Based on the different functionality, the software can be divided in to two basic parts: user interface and control. (see Figure 3.7)

Figure 3.9 below shows the flow chart for the C programming developed. As the program starts, first user will enter the GUI or user interface in the Visual

Basic software (see Figure 3.7), then user can key in the data on the X,Y and Z axis as he chooses. After user has satisfied with his options, he must hit the start button to start the movement of CNC 3 axis machine or program will start execute. From that the signal will be sent to the parallel port. Then parallel port will sent the signals to the Darlington as the IC is the controller that connects to the stepper motor. From the signals, numbers of bit will be sent to the stepper motor to start the motor rotation. Algorithm and calculation that has written and program in C programming will be functional. This program will actuate how far the motor will move and how many numbers of bits needed to move the distance that have been entering. After all the cycle has completed, user can choose whether he want to continue the cycle or just terminate the process. If the user wants to continue the cycle, it will start with the user key in the data in the user interface.

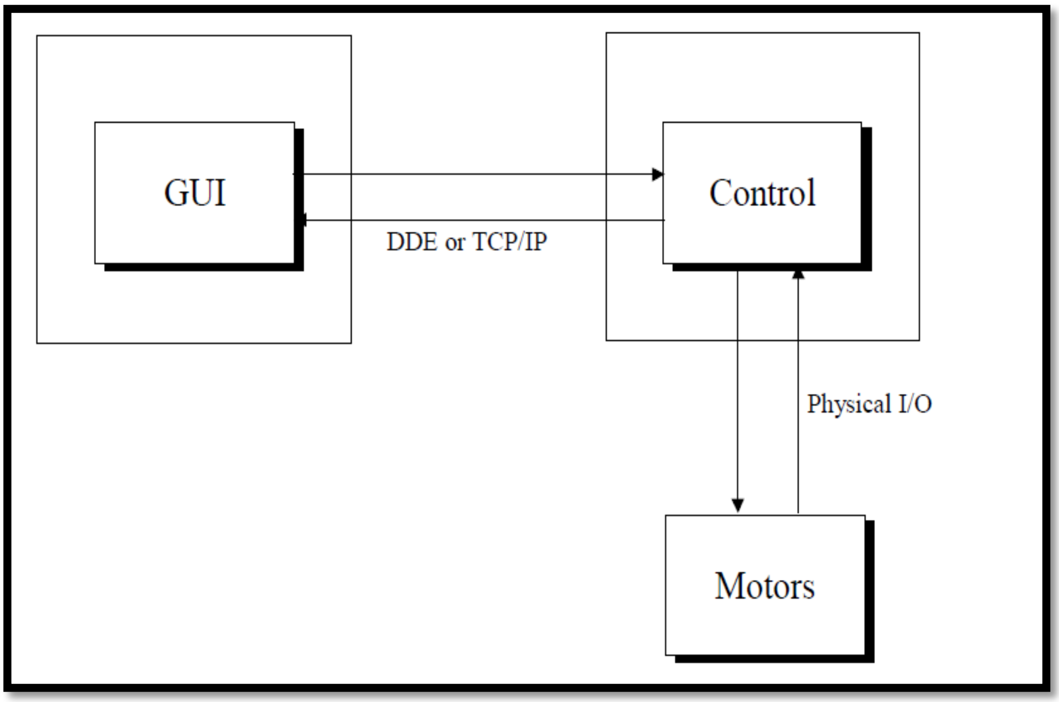


Figure 3.7: User interface and software

Source : Yitao Duan, Pete Retondo and Robert Hillaire, 1998

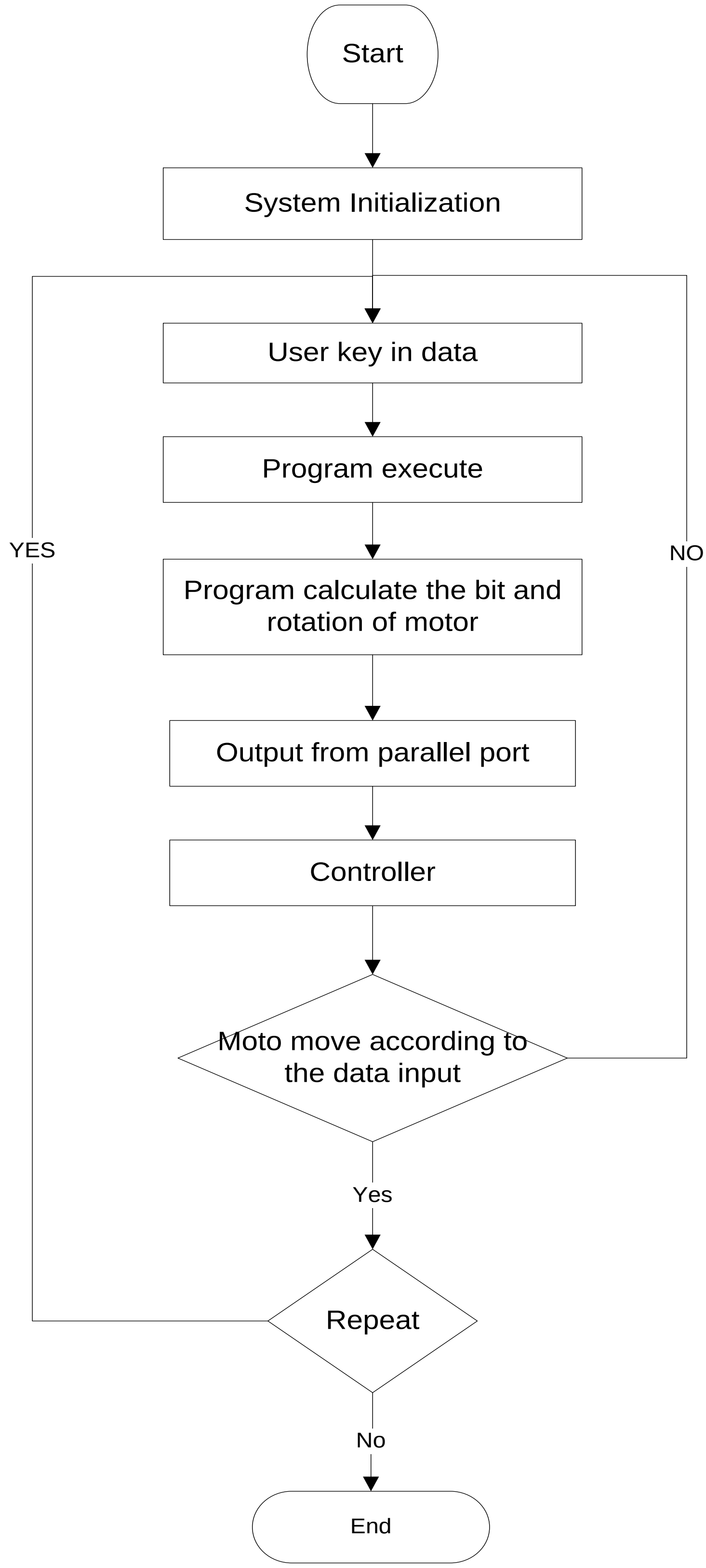


Figure 3.8: Research Methodology Flowchart

3.6.1 Parallel Port Connection

Parallel port will be use in this project as these projects is related with software and hardware that use computer programming as the main part of the project. Figure 3.9 below shows the pins out for the parallel both whereas table 3.1 shows the address and the number of bit for each pins. In this projects, only pins 2 to 9 will be use, the other pins is call dummy pins. Stepper motor consist only 6 wire to attach to the parallel port, so either these 2 to 9 pins out will be use to connect the steeper motor to the parallel port.

Address and bit of parallel port is very important to specify the motor movement. Therefore in this project, calculation of bit cannot be overstated. To activate the bit at the pin outs, simply write the value on the program according to its bit to activate it. Let say bit D7, D6 and D3 wants to be activate, what the user should do is, just write on the program the value of the bit. For example D7 its value is 128, meanwhile D6 its value is 64 and 8 is the value of D3. These bits will determine the motor rotation as the user key in the data as they choose.

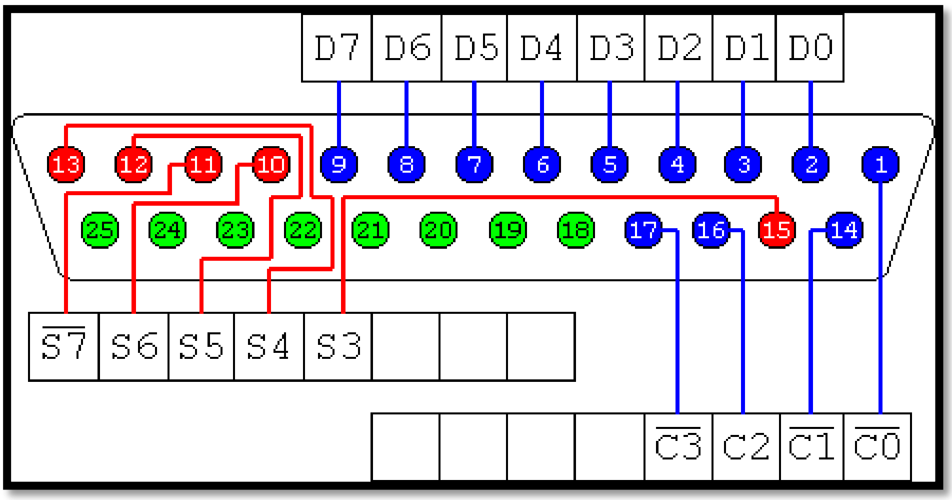


Figure 3.9: Parallel port output

Source : http://www.planet-source-code.com/vb/

Table 3.1: Parallel port address and its value

Pin No (D-Type)	Directions (In/Out)	Value	Bit	Address
1	In/Out		C0	257 (Control Port)
2	Out	1	D0	255 (Data Port)
3	Out	2	D1	255 (Data Port)
4	Out	4	D2	255 (Data Port)
5	Out	8	D3	255 (Data Port)
6	Out	16	D4	255 (Data Port)
7	Out	32	D5	255 (Data Port)
8	Out	64	D6	255 (Data Port)
9	Out	128	D7	255 (Data Port)
10	In		S6	256 (Status port)
11	In		S7	256 (Status port)
12	In		S5	256 (Status port)
13	In		S4	256 (Status port)
14	In/Out		C1	257 (Control Port)
15	In		S3	256 (Status port)
16	In/Out		C2	257 (Control Port)
17	In/Out		C3	257 (Control Port)
18-25	Gnd			

3.6.2 Calculation of Stepper Motor

As stated earlier, activated the bits will cause the rotation of the stepper motor. As the motor starts its rotation, some distances will produce. This is in line with the user desire where when the user key in the distance in X, Y and Z directions, the motor will go to that distance precisely. For the distance movement, some mathematical algorithm should be examined.

Terminology:

1. Step = Pulses:
   Distance, where 1 pulse could be 1.8° if full-stepping, and 0.9° if half-stepping.

2. Hz = pps (pulses/sec) = frequency = steps/sec:
   Speed of rotation in terms of steps or pulses.

3. Microstepping
   Microstepping Dividing the motors natural full step by smaller increments, for example, a 1.88° step motor microstepped at 64x will mean that 1 pulse is now 1.8° /64 = 0.0288°. Now, speed is 64 times slower and must send the motor extra steps just to have it perform the same as when it was in full step mode.

This terminology explained about the steps in the stepper motor which stepper motor is very synonym to its steps and rotation. For 1.8° stepper motor have 200 steps/revolution, meanwhile for 0.9° stepper motor have 400 steps/revolution and lastly for 0.45° stepper motor has 800 steps/revolution. This is because; 360° make the full rotational cycle. So to get the 1.8° number of step to achieve 360°, what to do is divide 360°/1.8° then the result will get 200 steps/revolution.

Finished with the steps, now move on with the calculation of distance. The distance calculations a little bit easy where all the diameter of the wheel attaches to the motor are taken. From that the diameter of the wheels times 3.14 (∏), the results numbers of circumferences will generated

Let say if the wheels diameter is 20cm,

$$20 \text{ cm} \times 3.14 = 62.8\text{cm}$$

From the equation below, knowing 20cm of wheels will rotate 62.8cm per revolution. If the user wants to move 10 cm, further mathematical equations have to be defined. 1 revolution consists of 200 steps. For make the full cycle, all number of bits has to activate, so that 10 cm length, only certain number of bits must be activated.

This all algorithm then will be write in the C programming as the program to make the motor move according to the user data input on the axis. For the conclusion, the mathematical algorithm closely related with the bits calculation. They very related to each other to make this project successful. Further investigation and experiment will be analyzed and the results will discuss later on Chapter 4.

The real 3 Axis CNC machines that will use in this project is shown below on figure 3.11

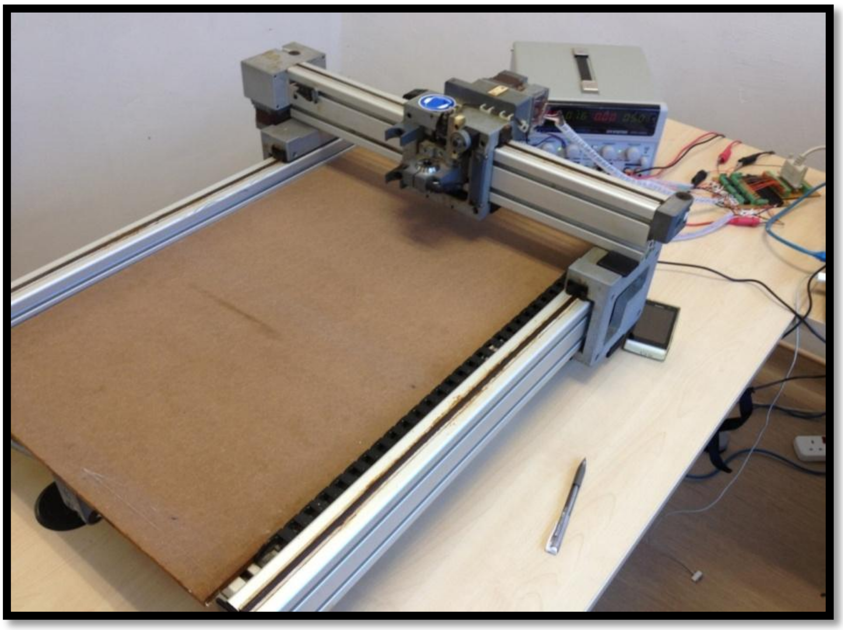


Figure 3.10: The 3 Axis machine use in this project

3.7 STEPPER MOTOR STEPS

`The positioning resolution of stepper motor required by the application may have an effect on the type of transmission used, and/or selection of the stepper motor driver. For example: A lead screw with 5 threads per inch on a full-step drive provides 0.001 cm/step; half-step provides 0.0005 mm/step; a microstep resolution of 25,400 steps/rev provides 0.0000015 cm/step. Stepper motors are driven by waveforms which approximate to sinusoidal waveforms. There are three excitation modes commonly used with stepper motors, which is full-step, half-step and microstepping.

In full-step operation, the stepper motor steps through the normal step angle, e.g. with a 200 step/revolution the motor rotates 1.8° per full step, while in half-step operation the motor rotates 0.9° per full step.

Table 3.2 : Full steps sequence for Stepper motor

Winding A	Winding B	Winding C	Winding D
1	1	0	0
0	1	1	0
0	0	1	1
1	0	0	1

CHAPTER 4

RESULT AND DISCUSSION

4.1 INTRODUCTION

After completing all the methods from previous chapter, this machine programs and interface will be analyze. In this chapter shows all the results obtained from this study. Tables of results, interfaces, and figures are included. Detailed explanation of interfaces and figures are also provided. This chapter also present and discuss about stepper motor configuration and the pulses that must be given to run a specific distance, and lastly the affected of programming on the CNC machine movement. The program and the interface will be program using Visual Basic 2008.

The optimization method usage and interpretation of its results are obtained based on detailed study of the usage of the software involved. First, make the circuit function. Then make the Graphical User Interface (GUI) by using software. This interface is going to use as the controller of the CNC machine. In this case the Visual Basic 2008 was used. This software is user friendly and reliable. After successfully make the interface, it will connect to the circuit to see whether the circuit functions or not. Analysis and discussion were focused more on the programming such as coding and mathematical function. The result is extracted from the trials, based on the CNC machine movement. Lastly, the results will be test with a conformation program to determine the movement of CNC machine based on the coding written are true or not according to the real CNC machine.

4.2 Visual Basic Graphical User Interface (GUI)

In order to control the stepper motor movement and angle, user interface from Visual Basic 2008 software has to be made. It can make any interfaces, program and style based on the project requirement. This software also has many function and shape which is each button and boxes that created can be program. Figure 4.1 below show the graphical user interface (GUI) of CNC machine controller.

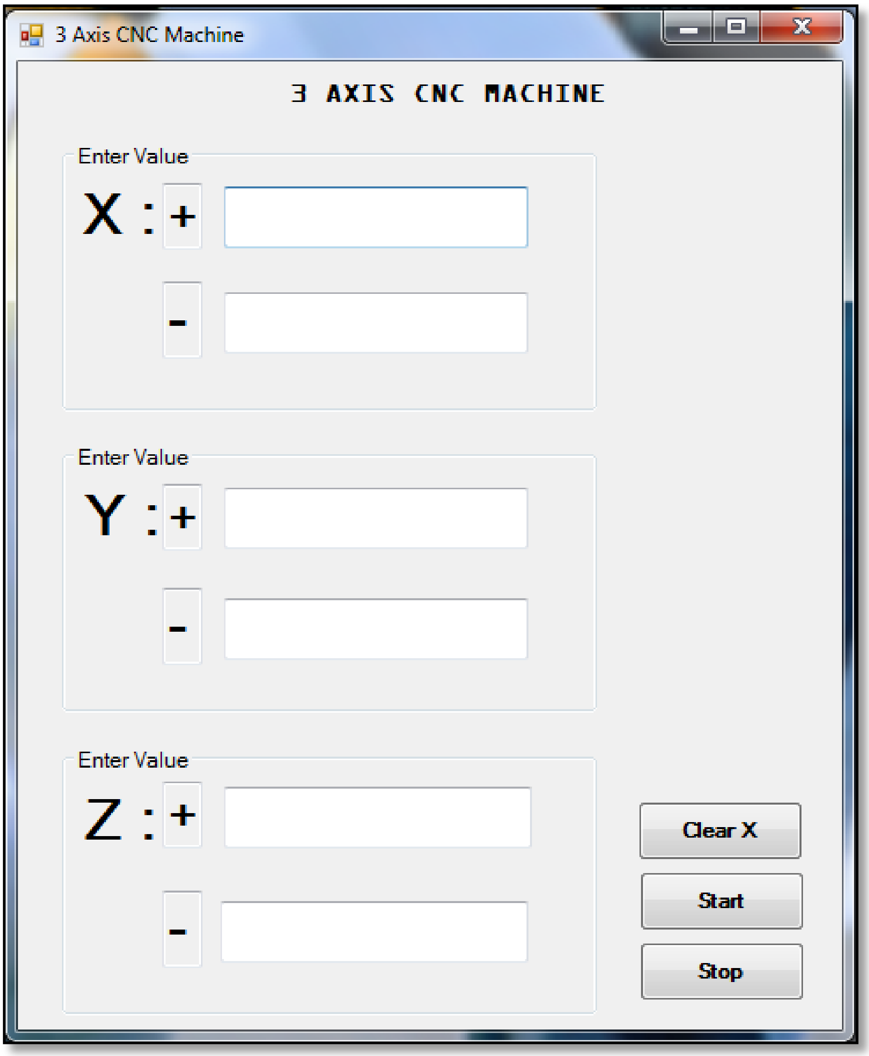


Figure 4.1 : Interface of CNC machine controller

Like the other CNC machine, this machine consist of three axis which is X, Y and Z axis where this three axes, then divide to the positive value and also negative value. The three buttons, Clear, Start and Stop functions are for run the motor on CNC machine. The interfaces act like a controller where user has to key in the values at every axis either positive value or negative value (according to the sign) then user have hit the Start button. It should be noted that all boxes must be filled with values even though the value is zero, as shown on the Figure 4.2 below or otherwise there will be an error running the software.

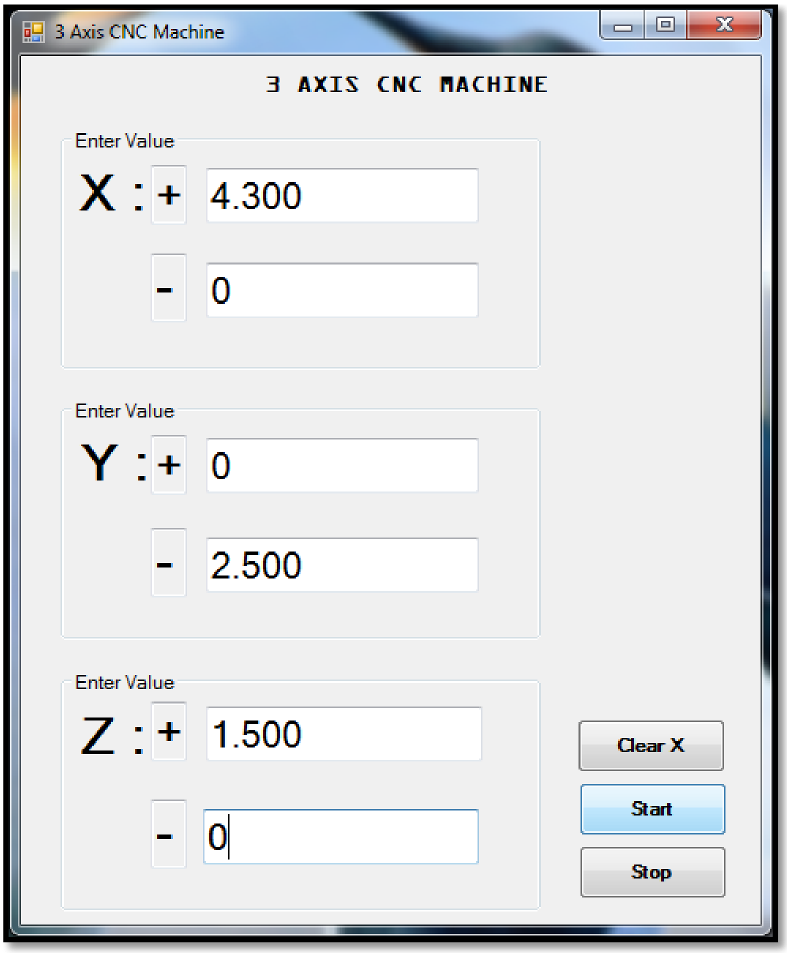


Figure 4.2 : All boxes must be filled with values.

The function of Clear button is for clear all the value at once. User also can clear the values using backspace button on the computer keyboard, but backspace button clear the values one by one. In fact, clear button and backspace button give the same result, it’s depend on the user situation where want to clear all the values or certain values only. And lastly the Stop button is for stop the motor movement immediately. User can press the Stop button while the machine operates. This Stop button also acts like the emergency button.

4.2.1 Design the Interfaces

Visual Basic software is one of the easily program the language not only allows programmers to create simple GUI applications, but also to develop complex applications. Programming in Visual Basic is a combination of visually arranging components or controls on a form, specifying attributes and actions of those components, and writing additional lines of code for more functionality. Since default attributes and actions are defined for the components, a simple program can be created. Figure 4.3 below shows the Start page of Microsoft Visual Basic 2008.

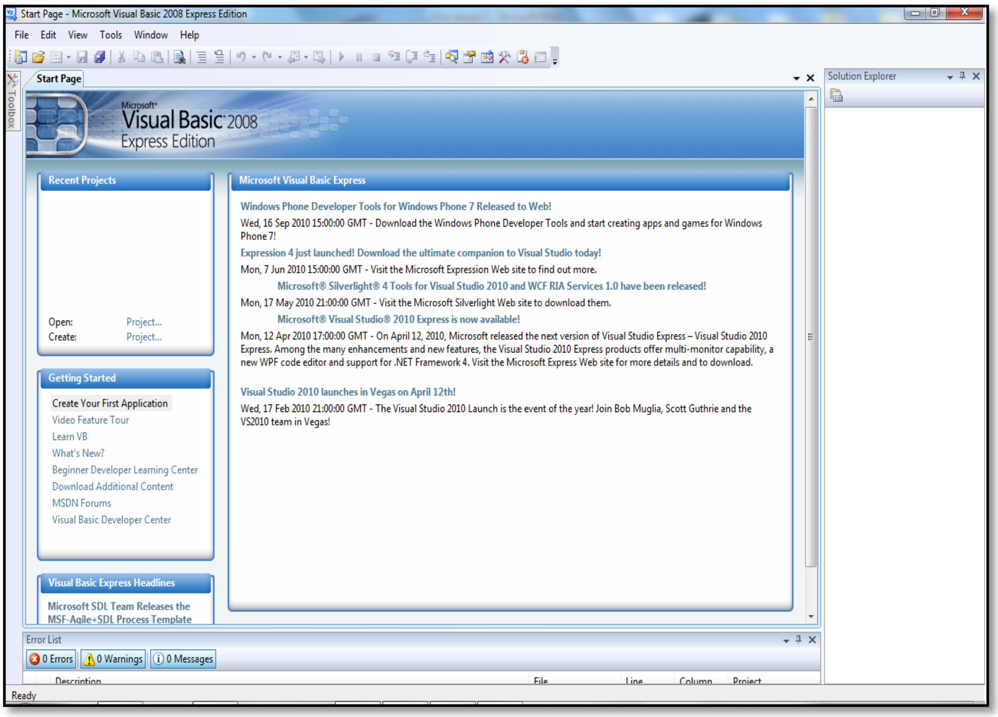


Figure 4.3 : Start Page of Microsoft Visual Basic 2008

After installing Microsoft Visual Basic 2008, open the software and the start page of this software will appear. Choose File from menu bar and then choose New Project, as shown at Figure 4.4

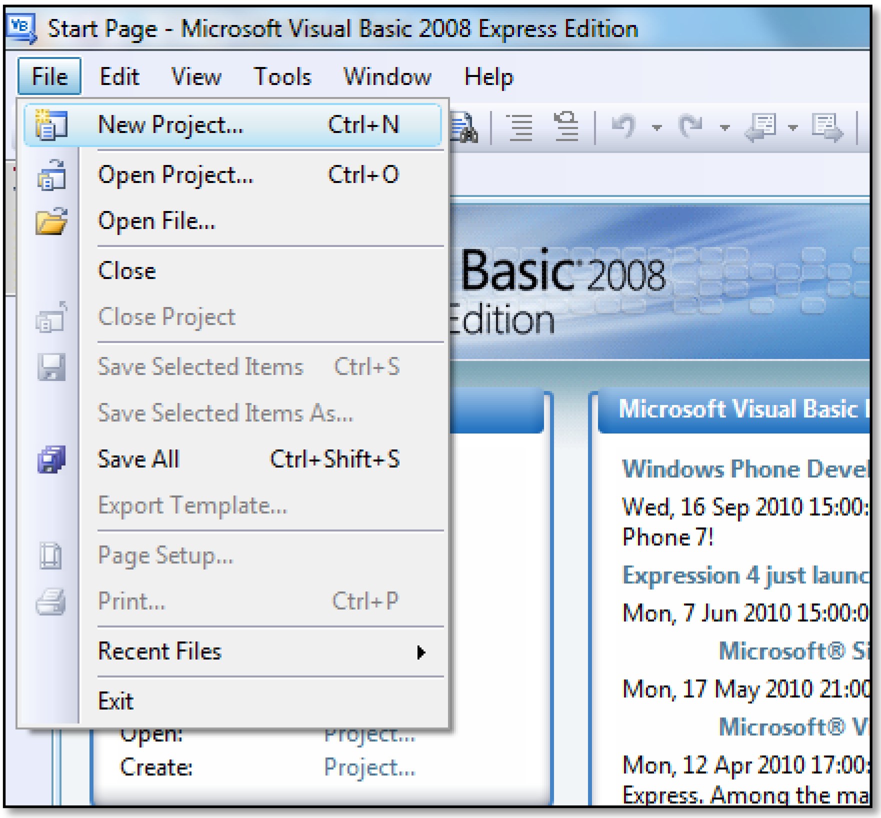


Figure 4.4 : Process of build new project form

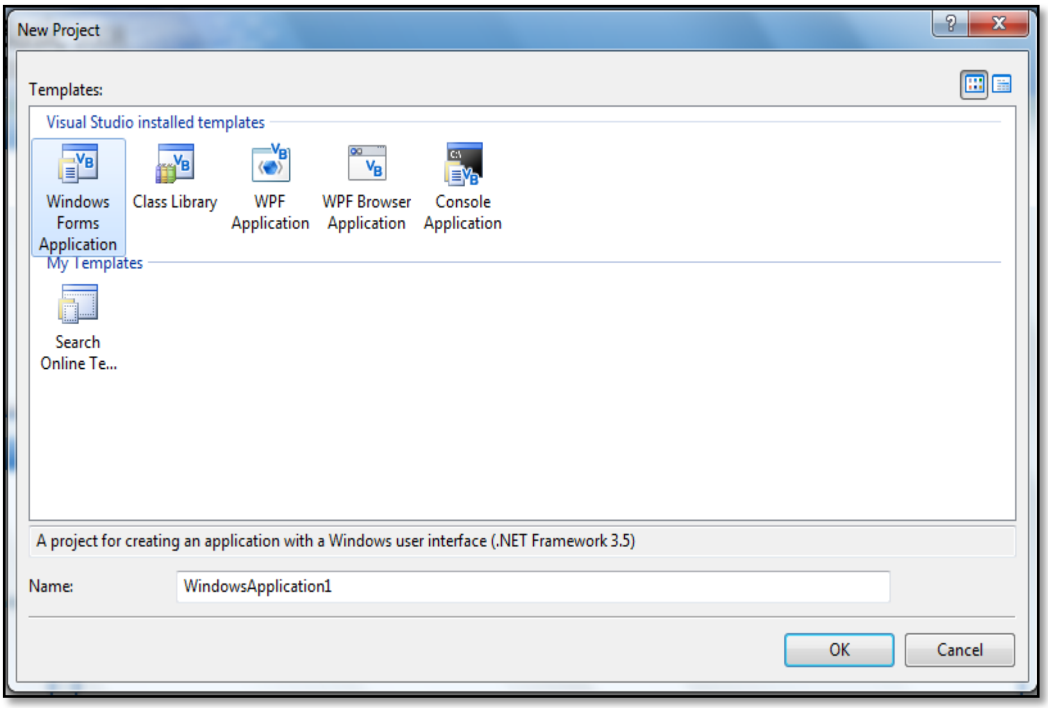


Figure 4.5 : Templates of Windows Form

After that user can see there are plenty of Templates, choose Window Form Application to start the new form and then click OK (see Figure 4.5). There are also other templates like Class Library, WPF Application, WPF Browser Application and Console Application. All of these Templates have a minor difference and its own application that suit to the project, this differences will discuss later. But for this study, GUI will be use to control the CNC machine, Windows Form Application is the best and appropriate to use according to this project.

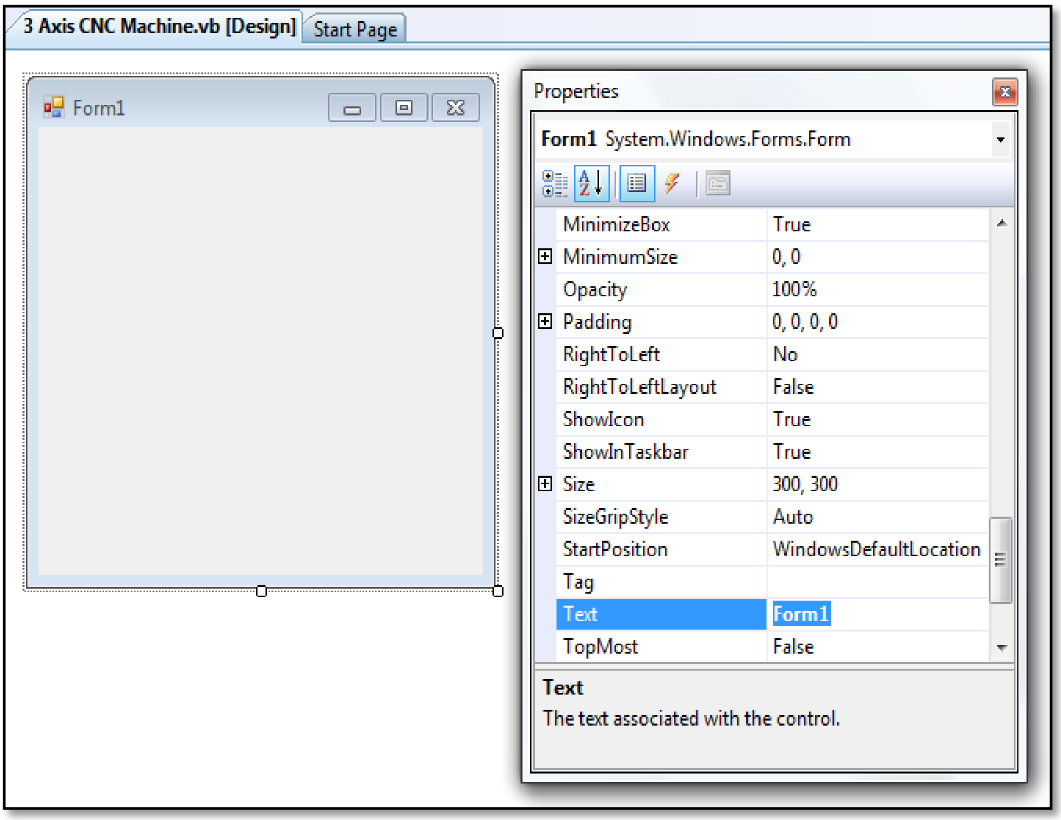


Figure 4.6 : Changing the properties at form design

After selecting Windows Form Application, Form 1 will be created. Figure 4.6 above show how the name of project can be change through the properties as well as the size of the form, the background color, background image, the text color and many more, this all kind of properties can be change under the properties of the form. The changes can be made as the users like according to the project requirement.

After the name form and the size have been changing, the next step is placing the group box on the form for placing the axes in the group. To do so, pickups a Group box from the toolbox at the top left of the software, under the container categories, click Group Box and then place it on the form as shown in Figure 4.7.

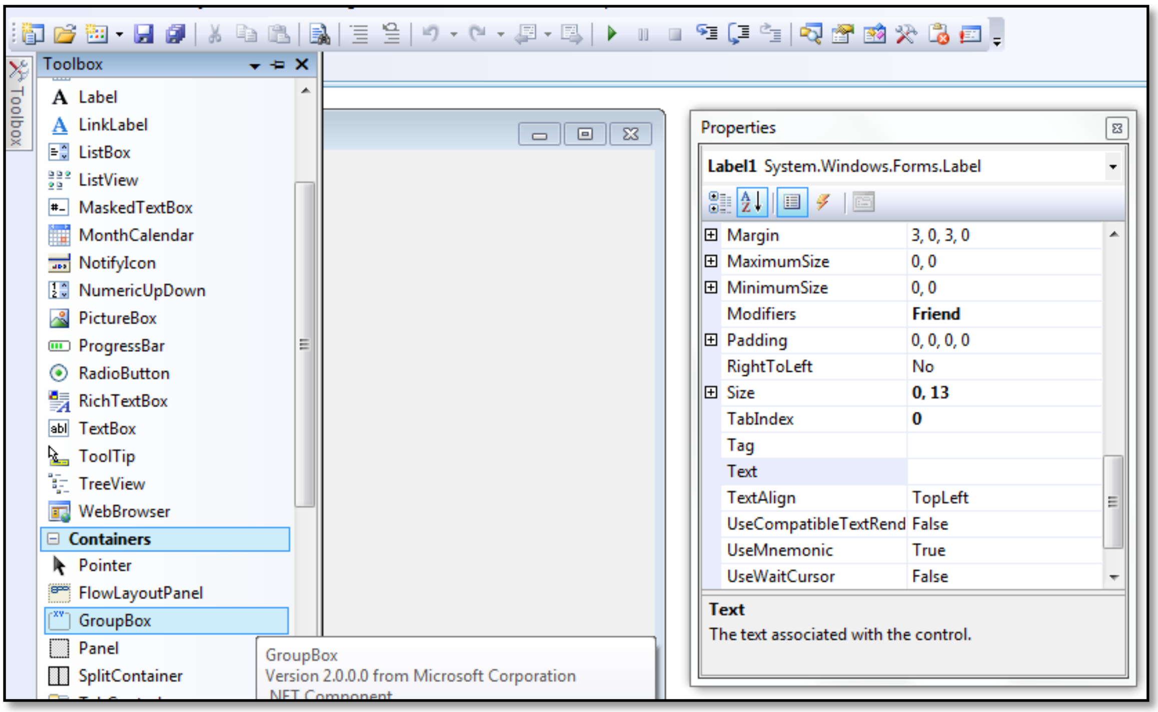


Figure 4.7 : Make a Group box

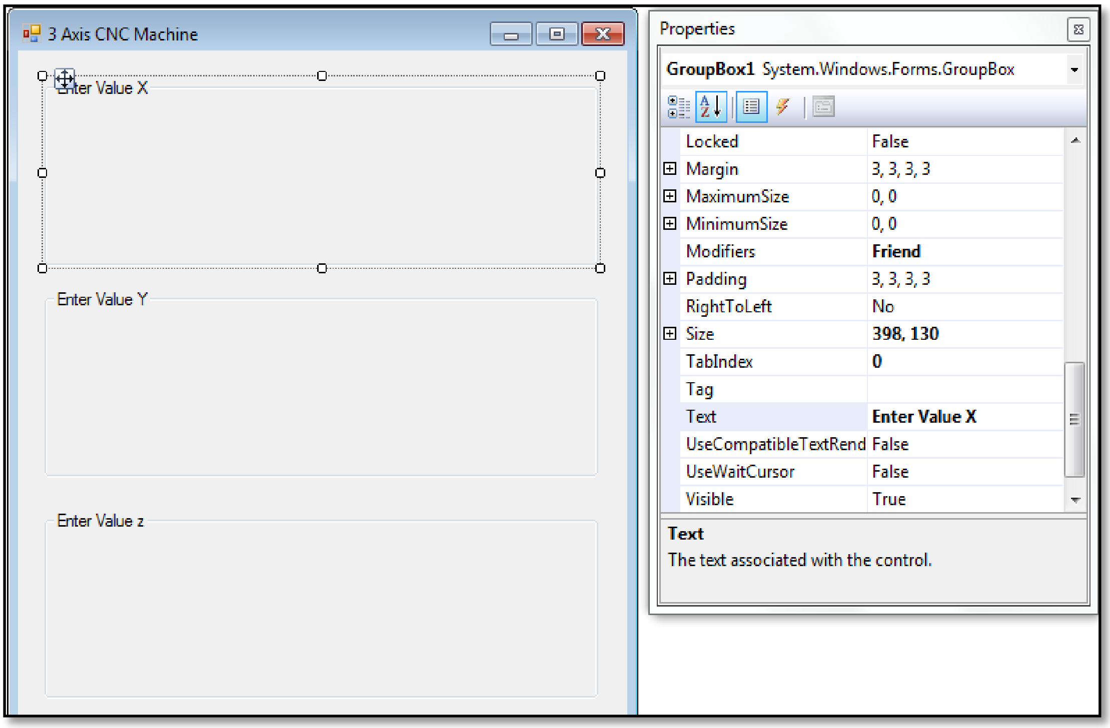


Figure 4.8 : Placing Group Box on the Form

Put the Group Box on the form and to each axes X, Y and Z as shown in Figure 4.8

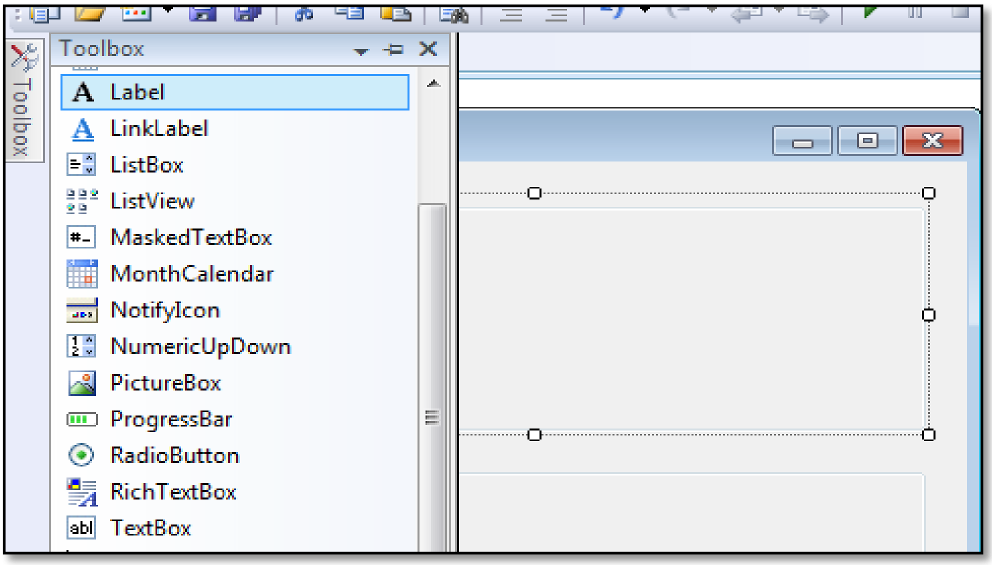


Figure 4.9 : Placing the axes on the form

First, open toolbox and then choose Label (see Figure 4.9) Place it on the form, and then change the text and also the size of the label. Place the label at every group to show the axes belong to that group as shown in Figure 4.10

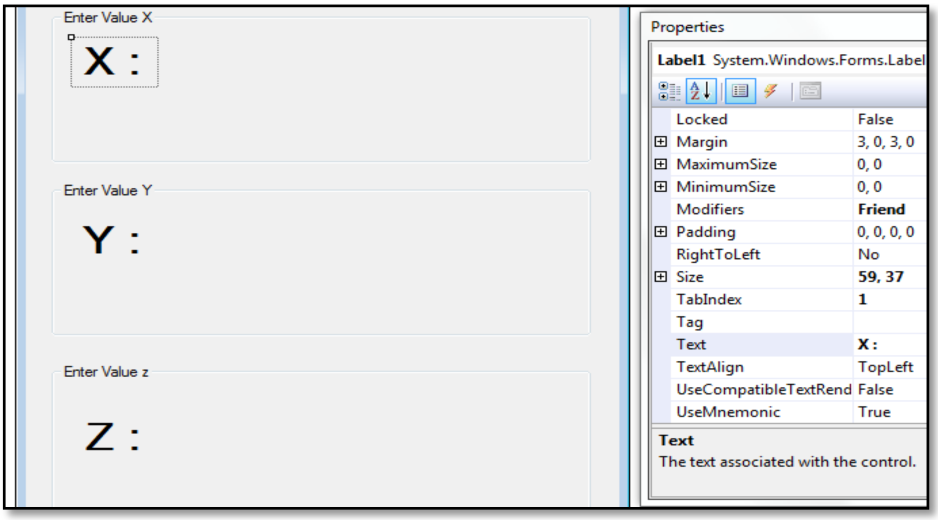


Figure 4.10 : Label the axes

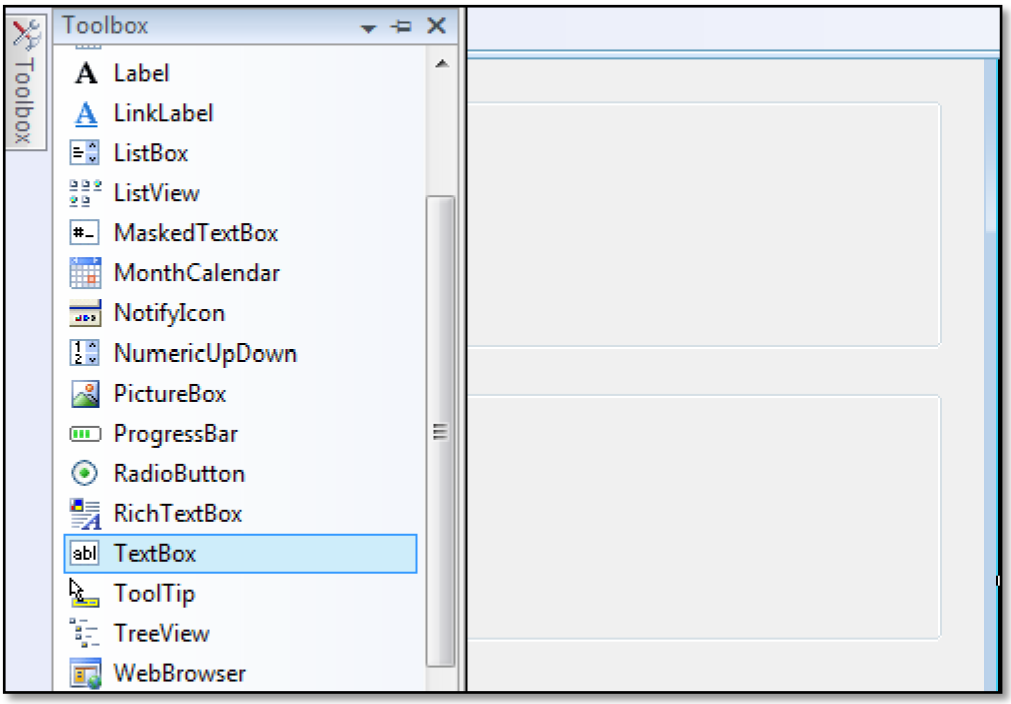
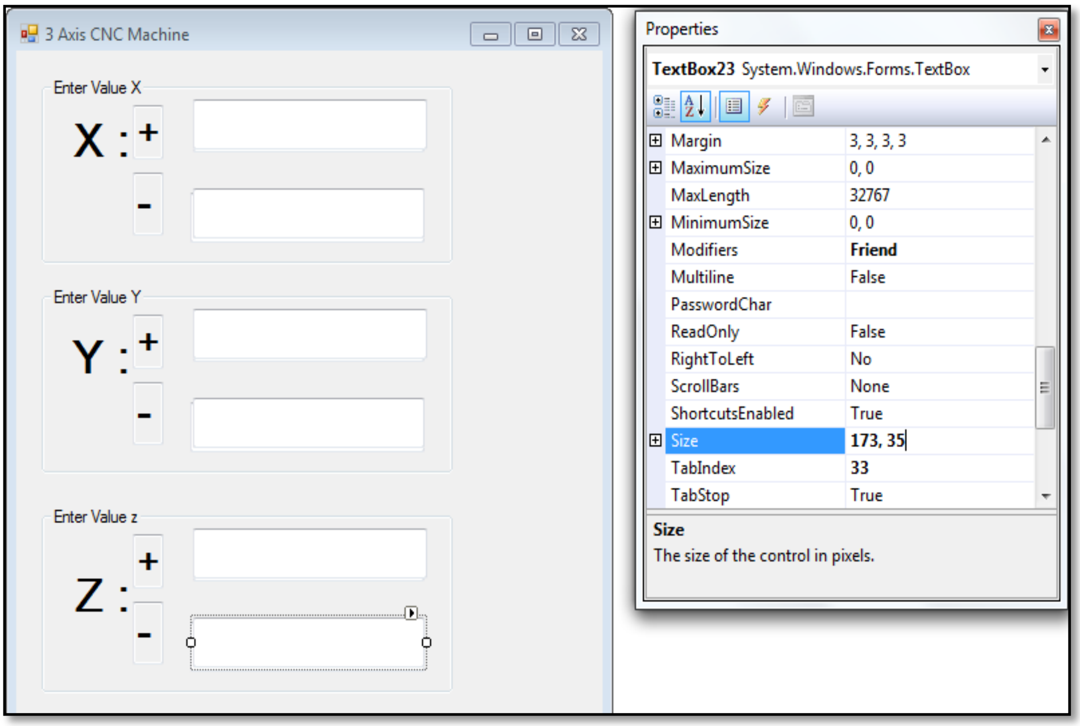


Figure 4.11 : Textbox

After placing the Label, the next step is placing the TextBox. To call upon the TextBox, click on the tool box and search the TextBox as shown in Figure 4.11. Click on the TextBox and placing it on the form for every group as shown in Figure 4.12. Once again, the size of textbox will be change due to the project compatibility.



4.12 : Placing the text box and change the size of font.

And the final steps are making the Clear, Stop and Start button. The steps of making these button are, first select the tool box at the left corner up on the software. Next choose 'button' and arrange it on the form. See Figure 4.13

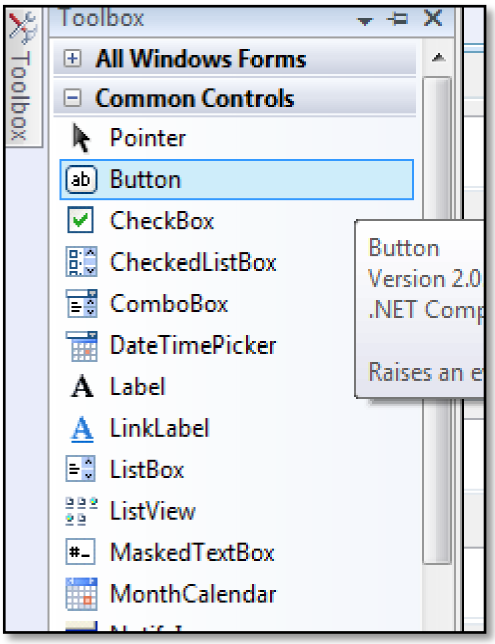


Figure 4.13: Make Start, Stop and Clear button.

The place of these buttons can put anywhere, but preferring to arrange it follow the standard place where the button should be placing. As shown in Figure 4.14. Then change the size of text, type of font, and rename it.

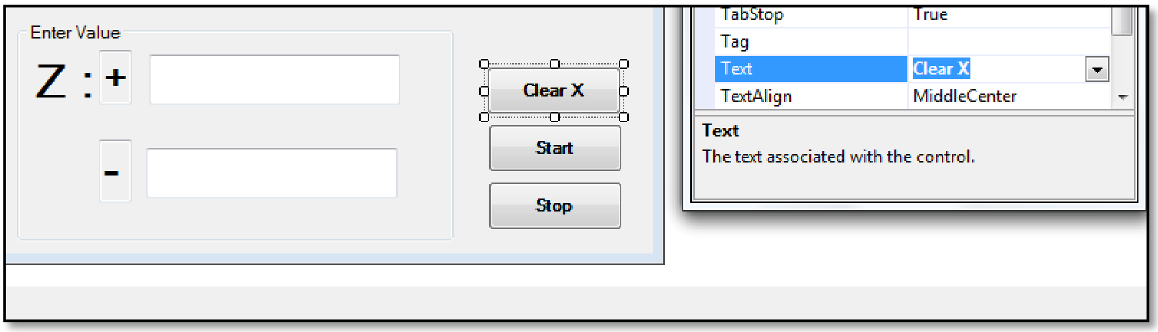


Figure 4.14 : Placing the buttons at the bottom right corner of the form.

After place the buttons, the completed form known as 3 axis CNC machine were produced (see Figure 4.14). The flowchart below conclude the flow of how to making this form as shown in Figure 4.15.

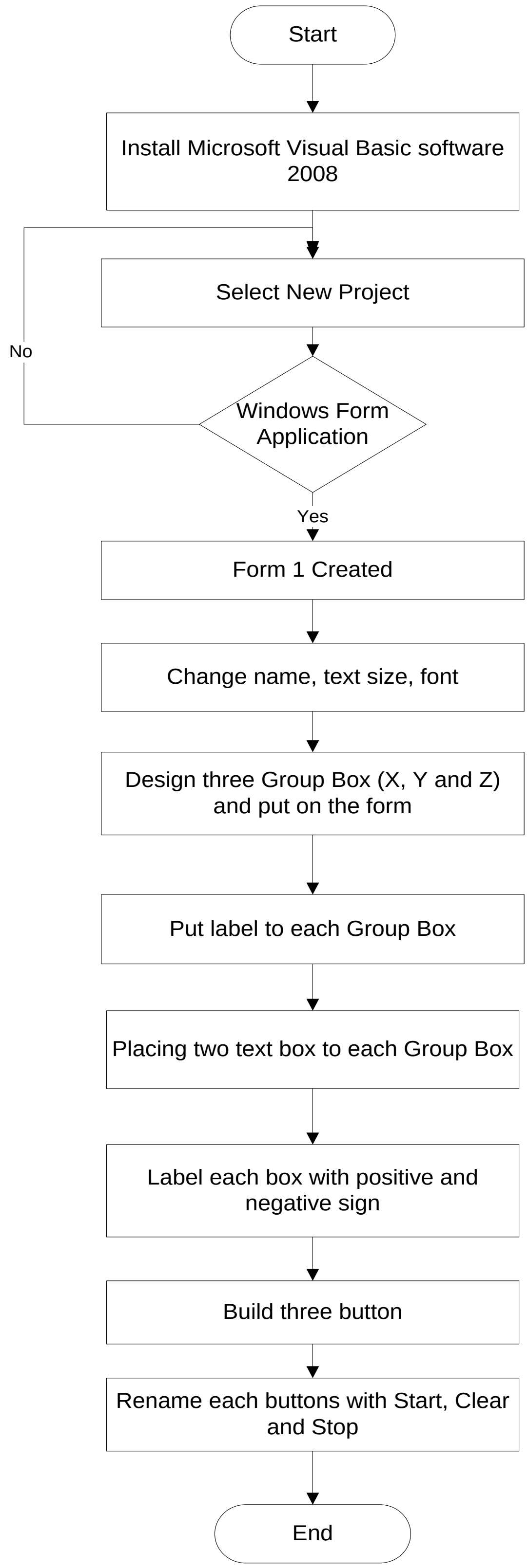


Figure 4.15 : Flowchart of making the GUI for CNC machine

This interface actually can be build using many processes, not only follow the process that has been show above but as long as the function and result are still same. The purpose of the flowchart is going to give the first idea how to build the interface. Once the user has capable to build its own interfaces, user's can use their own idea how to build it with or without following the steps above.

4.3 Circuit Development

The very important thing on circuit development is to make sure all the connections of every wiring is right. Otherwise the motor will not functioning. To make sure the connections between address from parallel port to the motor is right, first, the circuit was design using multisim software. The purpose of this software is to see whether the circuit can function or not based on the wiring on the software. The circuit of the 3 Axis machine was design using multisim software is shown on Figure 4.16 below.

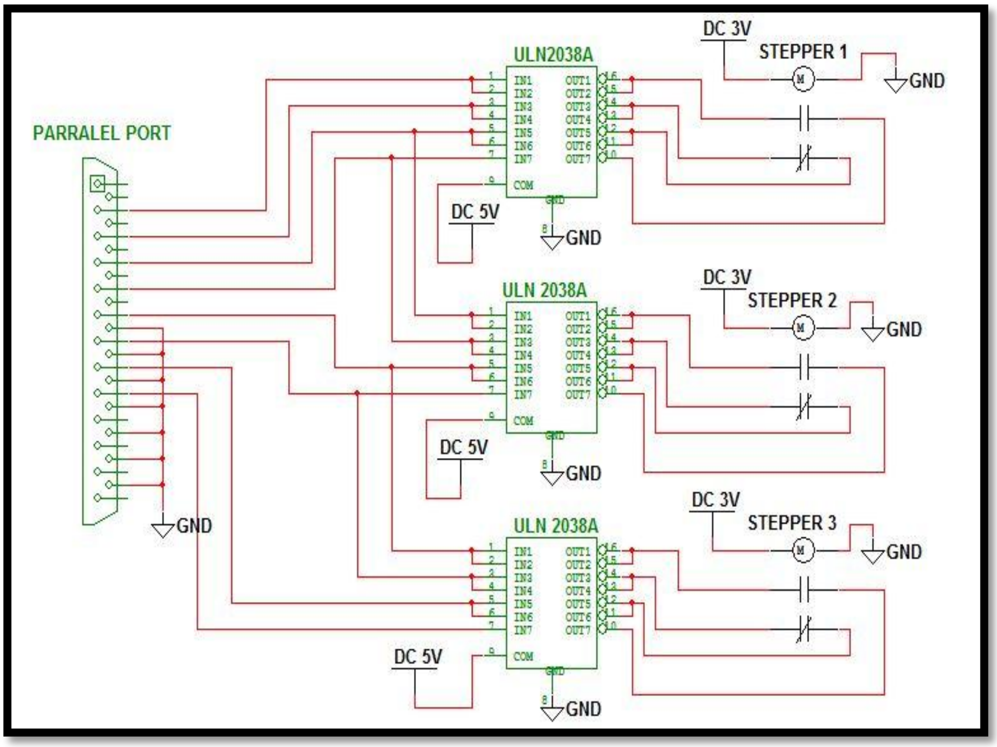


Figure 4.16: Design of 3 Axis CNC machine on multisim software

After the circuit was successfully working on multisim, the next step is test the circuit using breadboard connects to the parallel port to see whether it is function or not, before soldering on donut board or etching the circuit. Of course to trigger the LED, some simple programs are needed. The address connect to the parallel port also must be right in order to make the LED blinking sequently to show the stepper motor make a full cycle of rotation. The testing method was shown on Figure 4.17.

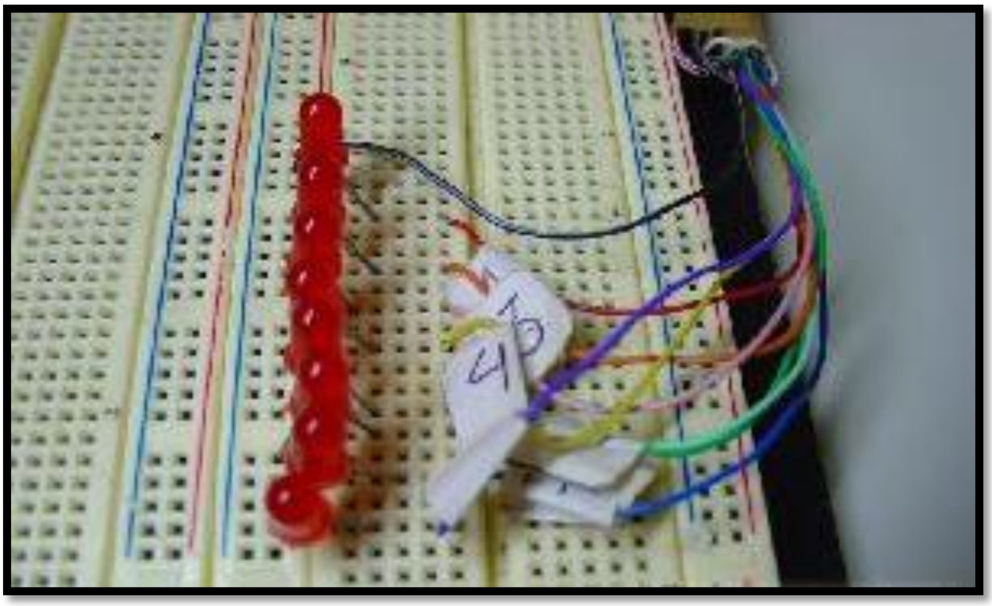


Figure 4.17 : Testing the circuit using LED and breadboard

After all methods was use to make sure the circuit fully working without any disturbance or short circuit, the last step is soldering the circuit on the donut board with the ULN 2803-A and together with the wire of each stepper motor. The final working circuit is shown on the Figure 4.18.

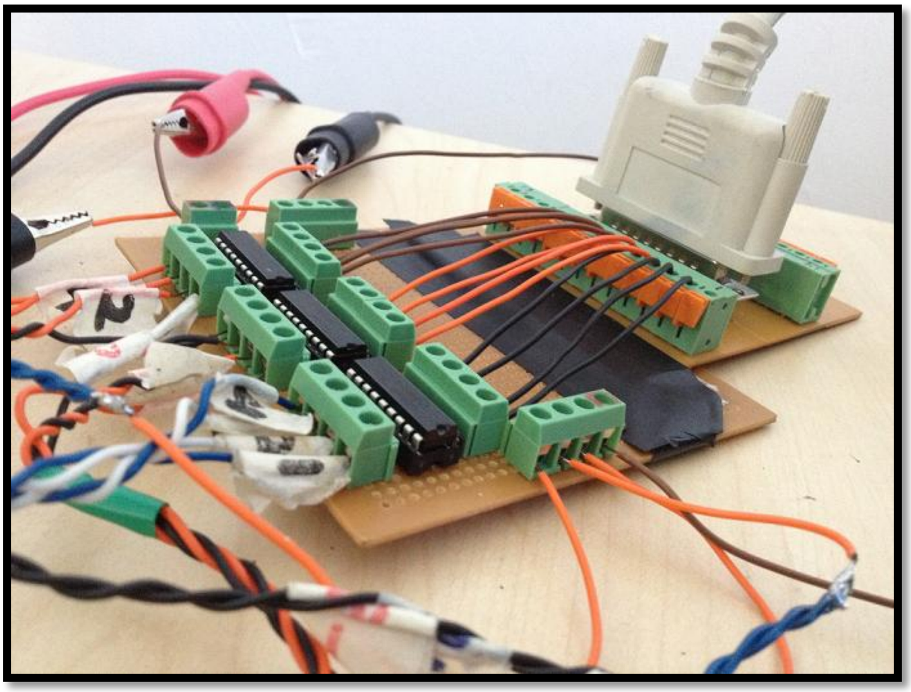


Figure 4.18 : The final Circuit that has been soldering on Donut Board

The completed circuit then being test using the interface that had been design on VB. After the test run were successful, circuit box was made to place the circuit in it, then labeling the data on the circuit box. The circuit was shown in Figure 4.19 below.

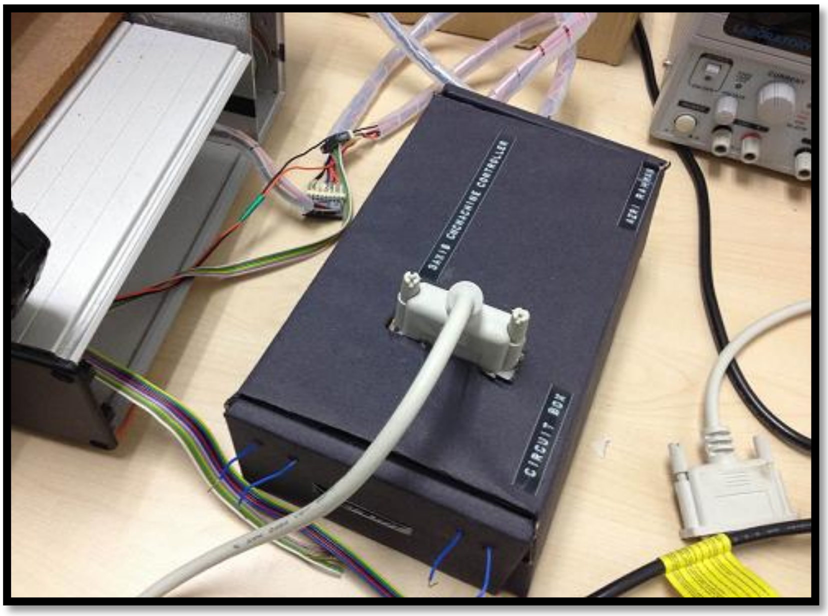


Figure 4.19 : Circuit Box for 3 Axis CNC machine

The block diagram of making the circuit was shown on the figure 4.21 below.

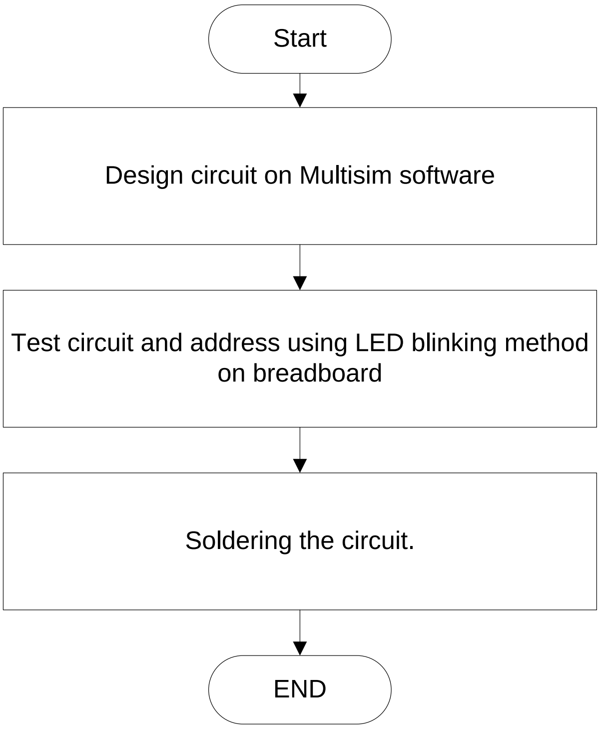


Figure 4.20 : Process of making 3 Axis CNC machine circuit

4.4 RESULT

For the results, the motor is move when the pulses are given. Thus, it makes the machine move according to the distance that the user want. When the user key in the data on GUI based on axes, the signals then will send to the output parallel port. Parallel port will receive the signals from PC and determine which address should be trigger. After that, output parallel port will send the signals to the stepper motor according to the address that had been insert on GUI. As a result, the motor will start rotate and the machine will move, following the axes and distance that had been key in.

Pulse or steps are calculated to move the stepper motor to the certain distance. For example, 1 pulse of stepper motor could be 1.8° for full stepping, 0.9° for half stepping or 0.45° for single stepping. Previous chapter state this project will use 1.8° of motor stepping which is it is full stepping method, whereby 1.8° for each stepper motor movement will made 200 steps for make a full rotation or 1 revolution. Equation 4.1 below shows how to calculate the steps/revolution when a step is given.

$$\frac{Phase\ Current\ Form\ (^\circ)}{1\ revolution\ (360^\circ)} = Steps/revolution \qquad \text{(Eqn 4.1)}$$

The distance travelled also depending on the size of the gears use. For this project, the motor gear was connected to the larger gear using belt. The purpose of the larger gear is to make the machine move faster. This can be proving by equation 4.2 below.

$$Circumference = 2\pi r \qquad \text{(Eqn 4.2)}$$

Where r is in radian.

Then, use the same formula (Eqn 4.2) to get the larger gear circumference. After the small and big gear circumference had been get, use the equation 4.3 below to determine the distance of 1 pulse motor will move.

$$\frac{Pulse}{Second} = \frac{revolutions}{second}\ x\ \frac{pulses}{revolution} \qquad \text{(Eqn 4.3)}$$

There are two method can be used to determine the distance of machine move if 1 pulses are given. The first method as shown in equation 4.1, 4.2 and 4.3, another method is by calibration process. In this project, the second method was use to determine the distance. The method will be discussed later on analysis.

4.5 ANALYSIS

In this analysis, calibration process was use to determine the distance of 1 steps in stepper motor. To determine the distance of 1 pulse, the simple program has to be made. The program and GUI was design using Visual Basic as shown in Figure 4.21 below.

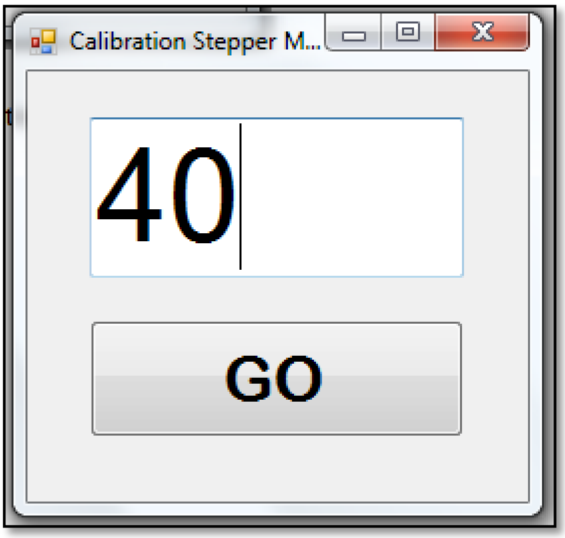


Figure 4.21 : Graphical User Interface for Calibration process

Button “GO” will be press after pulses was key in on the textbox. And as a result motor will be move to the certain distance. The programs of the calibration motor are shown on Figure 4.22 below.

```
Dim Xn As Integer
Dim X2 As Integer

        Xn = TextBox1.Text

        For 0 To Xn
            System.Threading.Thread.Sleep(value)
            Out(888, 6)
            System.Threading.Thread.Sleep(value)
            Out(888, 12)
            System.Threading.Thread.Sleep(value)
            Out(888, 24)
            System.Threading.Thread.Sleep(value)
            Out(888, 48)
        Next
```

Figure 4.22 : Program for calibration process

The value of Xn will be change to determine the distance of the motor. Xn is the value of pulses that was key in on the text box as shown in the Figure 4.21. When presses the buttons ‘GO’, motor will receive signals from the ULN 2803. The addresses that connect to the output parallel port are important because if the address was wrong, the motor will not move smoothly. After the machine was looping to the certain distance according to the textbox, by using a pen that clamp to the chuck, the linear line will be produce as the machine move as shown in Figure 4.23

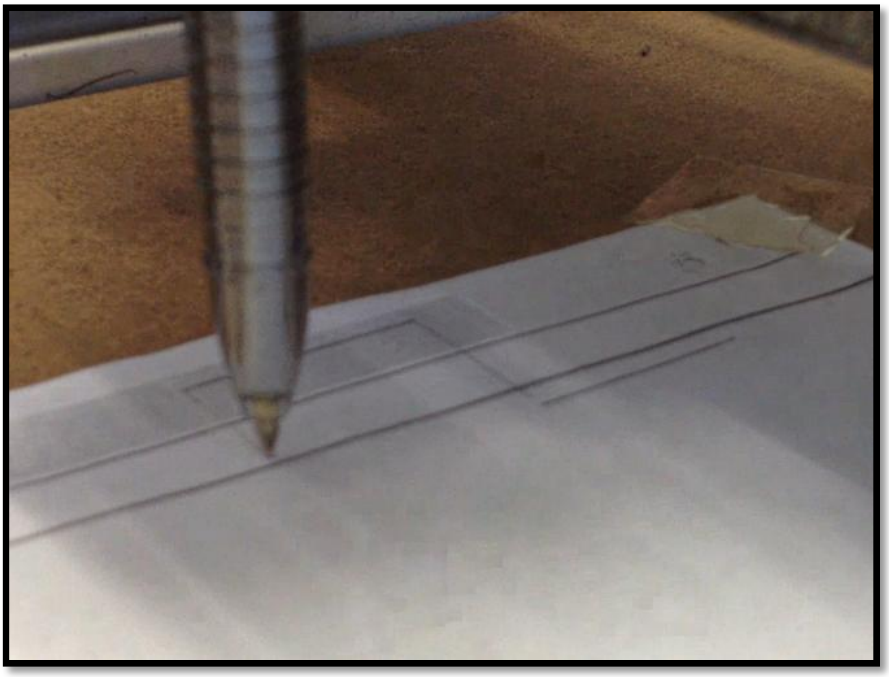


Figure 4.23 : The linear motion was produce when the calibration process being done

Table 4.1: Calibration result from various pulses

Steps	Distance Motor Move (mm)	Steps/mm
20	35	5.72
30	55	5.45
40	75	5.33
50	95	5.26
60	11	5.45
70	135	5.21

Table 4.1 show the result from the analysis that have been made, the purpose of this calibration is to determine the distance of the machine move when Xn value or pulses is given. This result will be use in the program to turn the motor distance that required. When the distance for 1 pulse was determine, the value from the axes will be multiply by the average value that get from the calibration process. The equation is shown below.

$$Motor\ move = Distance\ required\ x\ 5.4005 \qquad \text{(Eqn 3.4)}$$

Value 5.4 is the average values that were get from the data, this value will use in the in the program to make the machine move. Figure 4.24 shows how the values were use.

```
Dim Yp As Integer
Dim Y1 As Integer

        Yp = TextBox2.Text
        Y1 = Yp * 5.4
        value = 1


        For Yp = 0 To Y1

            System.Threading.Thread.Sleep(value)
            Out(888, 24)
            System.Threading.Thread.Sleep(value)
            Out(888, 48)
            System.Threading.Thread.Sleep(value)
            Out(888, 96)
            System.Threading.Thread.Sleep(value)
            Out(888, 192)

        Next

Dim Yn As Integer
Dim Y2 As Integer

        Yn = TextBox5.Text
        Y2 = Yn * 5.4
```

Figure 4.24 : 5.4 were multiply by the distance that will be key in by user on textbox2 and 5

For the Y and Z axes, this same methods was use to calibrate the data as those stepper motor have specifications.

4.6 DISCUSSION

Based on the results obtained, it can be conclude that 5.4 mm or 0.54 cm is the value of the machine will move for 1 pulse motor trigger. This value is not very accurate if compare with the real CNC machine, because the real CNC machine is very exact and precise as the machine purpose is to do the accurate things.

The values that get from the experiment are not very accurate because of some factor. The main factor of the inaccuracies is the types of waves that were use. In this project, full waves were use, that's mean 1.8° motor will trigger when 1 pulse are given. Compared with microstepping, where 1 pulse trigger will make the machine move 0.45° only. But this machine cannot use microstepping waves because the torque produce is not high, so it's probably will make the motor cannot support the CNC machine. So whether like it or not, full waves methods must use.

The distance were key in by the user will produce plus minus 1 mm. To make it the distance accurate, user have to try first whether the machine was over or less than 1mm. From that, user can key in plus or minus of the real value from the test run earlier.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 INTRODUCTION

The significant findings of the research were concluded in this chapter. The outcome of the research gave overall description of this case study. The limitation or problems encountered during conducting this research also notified together with the recommendations for future research purpose.

5.2 CONCLUSION

During the process to complete this project, there are some problems here and there especially on determine the motor stepper connection. But this problem finally overcome through a hard work, research and helping from friends. The experienced and skilled during four years learning in this engineering field also helps to overcome whatever problem faced.

This project starts with understanding the objectives of the project. From the objectives, the methods on how the machine works can be identify. It followed by research and understanding the function of each component use in this project. The wiring of stepper motor is the most challenging problem that has to be faced because the stepper motor connection is not same with the other motor connection. Finally, the user interface and program that had been learned also was implementing to make this project successful.

The stepper motors that were use in this project was imported from Nanotech Technologies where this stepper motor consist of six wires. Two wires are common wires and it is connected to the positive power supply. The other four wires have to test and connected sequently to make one rotation of motor. Usually, stepper motor comes with five or eight wires where actually the methods to control the steps of the motors are all same.

By using the programs on VB, it can determine the distance of motor travel. Base on the equation in 4.1, the value of the motor traveled can be determined. For full wave movement, it takes 1.8° of one pulse. So it takes 200 steps to complete 360° or one full rotation.

In calibration process, the programs to determine the motor traveled are written on the programs. By giving and changing the pulses on the program, the motor distance can be determined. The user graphical interfaced make the programs hidden behind it. This method can make the machine is easy to handle and friendly user. Users have to key in the data on the GUI. The programs then will calculate and analyze the signals and send it to the parallel port. From parallel port, the data will transmit to the ULN 2803, finally from the ULN, the signals will send to the stepper motor. And the results, the stepper motor will moving according to the desired distance that user had key in.

For the conclusion, this study has achieved the objectives where the controller for 3 Axis CNC Machine was successfully design. And the second objective is to validate the machine controller also achieved by the calibration experiment.

5.2 RECOMMENDATION

For current project, this machine has successfully run in 3 axes and also has achieved the objectives, but from the results and observation that had been obtained in the previous chapter, there are some future works can be recommended. For now, the 3 Axis CNC machine can move only in linear motion. Some additional research has to be done to make this machine move in curvilinear motion.

There also some recommendation can be done to attach to this machine to make it works as usual CNC machine. For example:

1. Cutting tools
2. Motor for tools
3. Spindle speed
4. Feed rate
5. Clamping tools

REFERENCES

1. B.S.V. Prasad (Jan, 1992). "Designing programming station software for CNC profile cutting." *Computers in Industry* pp. 67-76
2. Chana Raksiri and Manukid Parnichkun (April, 2004). "CNC Machine and the Tool Parts" *Machine tools and Manufactured* 12(9), pp. 1283-1291 Available: www.sciencedirect.com [April14, 2005]
3. G. Spur and H. Meier (March, 1979). "Workshop Programming with CNC Turning Machines" *CNC Control Programming* 12(3), pp. 482-486
4. http://electronics-diy.com/stepper_motors.php
5. http://freedatasheets.com/datasheetblog
6. http://www.alpha-crucis.com/en/power-components/4045-8-channel-darlington-driver-solenoid-unipolar-stepper-uln2803a-3700386509704.html
7. https://www.motionusa.com/node/2970
8. http://www.planet-source-code.com/vb/
9. http://4circuits.com/motor-controls/stepper-motor-uln2803/
10. https://www.technologystudent.com
11. Linjian Yang and Jinchun Feng (November, 2011). "Research on Multi-axis CNC Programming in Machining Large Hydraulic Turbine's blades Based on UG". *Advance in Engineering* 33(3), pp. 768-772 Available: www.sciencedirect.com
12. Manocher Djassemi (June, 1998). "Efficient CNC Machining Operations." *Computers and Industrial Engineers* 21(3), pp. 33-36
13. M. Kovacic, M. Brezocnik, I. Pahole, J. Balic and B. Kecelj (May, 2005). "Evolutionary programming of CNC machines". *Materials Processing Technology* pp. 1379-1387 Available: www.elsevier. .com/locate/jmatprotec
14. Seungkil Son, Taejung Kim, Sanjay E. Sarma and Alexander Slocum (October, 2009). "A hybrid 5-axis CNC milling machine". *Precision Engineering* 12(2), pp. 430-466 Available: www.elsevier.com/locate/precision
15. Sotiris L.Omirou and Andreas C.Nearchou (Jan, 2009). "An Epitrochoidal Pocket – A new canned cycle for CNC milling machines". *Robotics and Computer-Integrated Manufacturing* 44(2), pp. 73-80

Available: www.elsevier.com/locate/rcim

16. Sotiris L.Omirou and Antigoni K. Barouni (June, 2005). "Integration of new programming capabilities into a CNC milling system". *Robotics and Computer-Integrated Manufacturing* 44(2), pp. 518-527
   Available: www.elsevier.com/locate/rcim
17. S.T. Newman and A. Nassehi (April, 2008). "Strategic advantages of interoperability for global manufacturing using CNC technology". *Robotics and Computer Integrated Manufacturing* 12(2), pp. 669-708

APPENDIX A

PROGRAMMING FOR CONTROL 3 AXIS CNC MACHINE

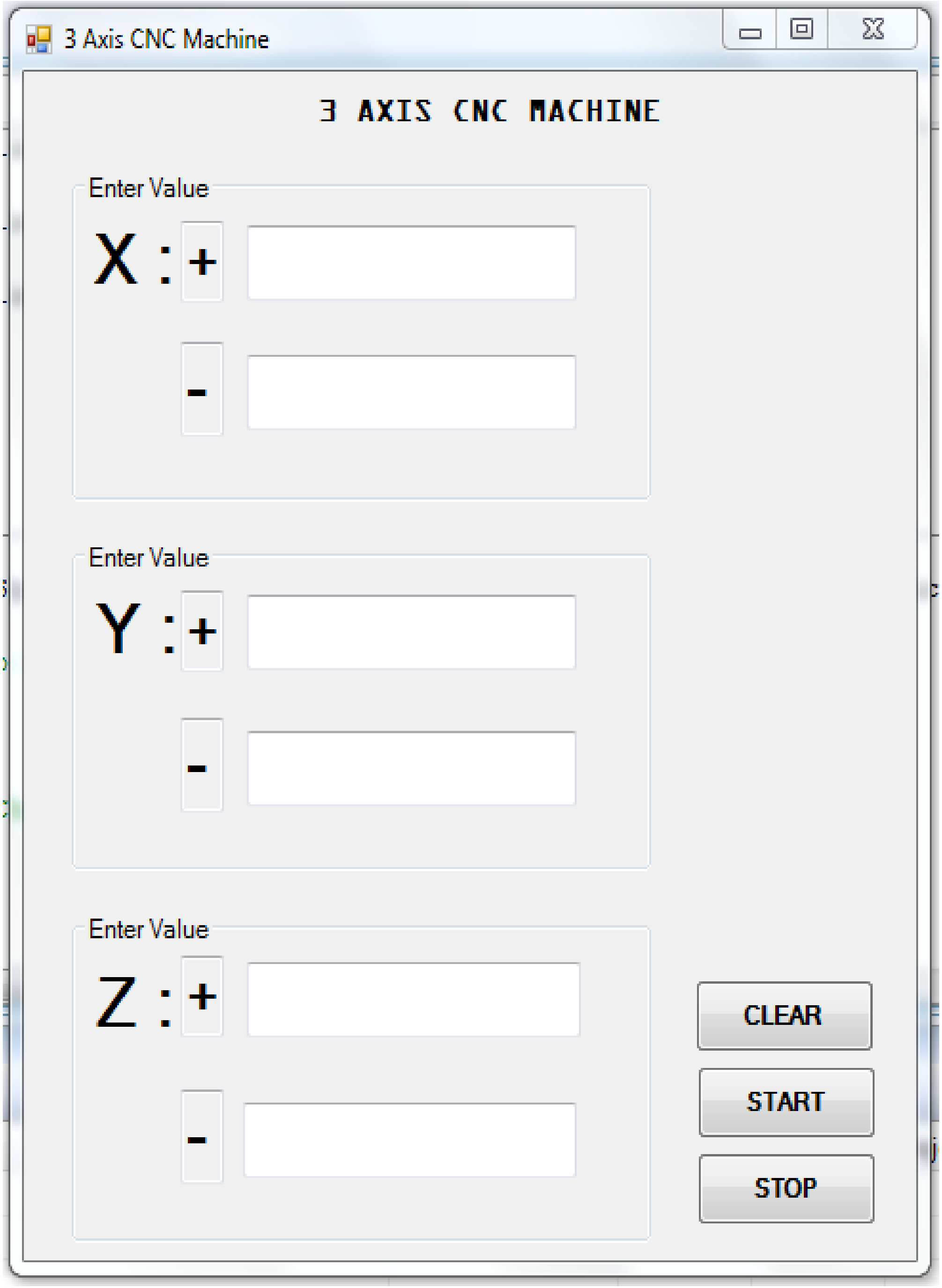
```
Public Class Form1

Public Declare Sub Out Lib "inpout32.dll" Alias "Out32" _
(ByVal PortAddress As Integer, _
ByVal Value As Integer)

Private Sub Button_ClearText_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles Button_ClearText.Click
        TextBox1.Text = ""
        TextBox2.Text = ""
        TextBox3.Text = ""
        TextBox4.Text = ""
        TextBox5.Text = ""
        TextBox6.Text = ""
    End Sub

Private Sub Button_Stop_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles Button_Stop.Click
        Out(888, 0)
    End Sub

Private Sub Button_Start_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button_Start.Click

        Dim Xp As Integer
        Dim X1 As Integer
        Dim value As Integer

        Xp = TextBox1.Text
        X1 = Xp * 5
        value = 1

        For Xp = 0 To X1

            System.Threading.Thread.Sleep(value)
            Out(888, 48)
            System.Threading.Thread.Sleep(value)
            Out(888, 24)
            System.Threading.Thread.Sleep(value)
            Out(888, 12)
            System.Threading.Thread.Sleep(value)
            Out(888, 6)

        Next

        Dim Xn As Integer
        Dim X2 As Integer

        Xn = TextBox4.Text
        X2 = Xn * 5

        For Xn = 0 To X2
            System.Threading.Thread.Sleep(value)
```

```
        Out(888, 6)
        System.Threading.Thread.Sleep(value)
        Out(888, 12)
        System.Threading.Thread.Sleep(value)
        Out(888, 24)
        System.Threading.Thread.Sleep(value)
        Out(888, 48)
    Next

    Dim Yp As Integer
    Dim Y1 As Integer

    Yp = TextBox2.Text
    Y1 = Yp * 5
    value = 1

    For Yp = 0 To Y1

        System.Threading.Thread.Sleep(value)
        Out(888, 24)
        System.Threading.Thread.Sleep(value)
        Out(888, 48)
        System.Threading.Thread.Sleep(value)
        Out(888, 96)
        System.Threading.Thread.Sleep(value)
        Out(888, 192)

    Next
    Dim Yn As Integer
    Dim Y2 As Integer

    Yn = TextBox5.Text
    Y2 = Yn * 5

    For Yn = 0 To Y2

        System.Threading.Thread.Sleep(value)
        Out(888, 192)
        System.Threading.Thread.Sleep(value)
        Out(888, 96)
        System.Threading.Thread.Sleep(value)
        Out(888, 48)
        System.Threading.Thread.Sleep(value)
        Out(888, 24)
    Next

    Dim Zp As Integer
    Dim Z1 As Integer

    Zp = TextBox3.Text
    Z1 = Zp * 1
    value = 1
    For Zp = 0 To Z1

        System.Threading.Thread.Sleep(value)
        Out(888, 24)
        System.Threading.Thread.Sleep(value)
        Out(888, 48)
        System.Threading.Thread.Sleep(value)
        Out(888, 96)
```

```
            System.Threading.Thread.Sleep(value)
            Out(888, 192)

        Next

        Dim Zn As Integer
        Dim Z2 As Integer

        Zn = TextBox6.Text
        Z2 = Zn * 1

        For Zn = 0 To Z2

            System.Threading.Thread.Sleep(value)
            Out(888, 192)
            System.Threading.Thread.Sleep(value)
            Out(888, 96)
            System.Threading.Thread.Sleep(value)
            Out(888, 48)
            System.Threading.Thread.Sleep(value)
            Out(888, 24)
        Next

    End Sub

Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer1.Tick

    If Label2.Left = -300 Then
       Label2.Left = 800 'Reset label to right AFTER most left
reached

    Else

       Label2.Left = Label2.Left - 50 'Change 50 to whatever
distance
       text to scroll

    End If

        Timer1.Enabled = False
        Timer1.Enabled = True

End Sub
End Class
```

APPENDIX B

GRAPHICAL USER INTERFACE FOR 3 AXIS CNC MACHINE



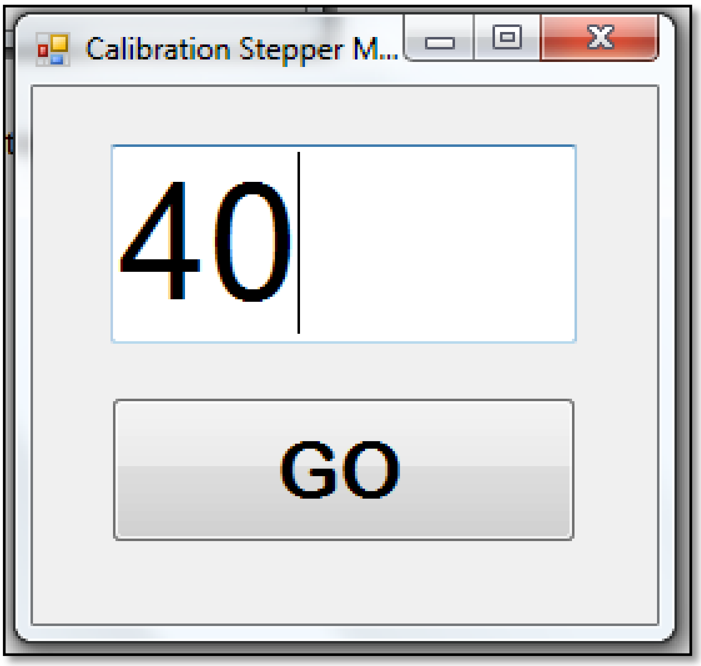
APPENDIX C

PROGRAM FOR CALIBRATION PROCESS

```
Public Class Form1 Public Declare Sub Out Lib "inpout32.dll" Alias "Out32" _ (ByVal PortAddress As Integer, _ ByVal Value As Integer)

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

Dim i As Integer i = 0
Dim x As Integer x = 10

Do Until i = 150

Out(888,9)
System.Threading.Thread.Sleep(x)
Out(888,3)
System.Threading.Thread.Sleep(x)
Out(888,6)
System.Threading.Thread.Sleep(x)
Out(888,12)
```

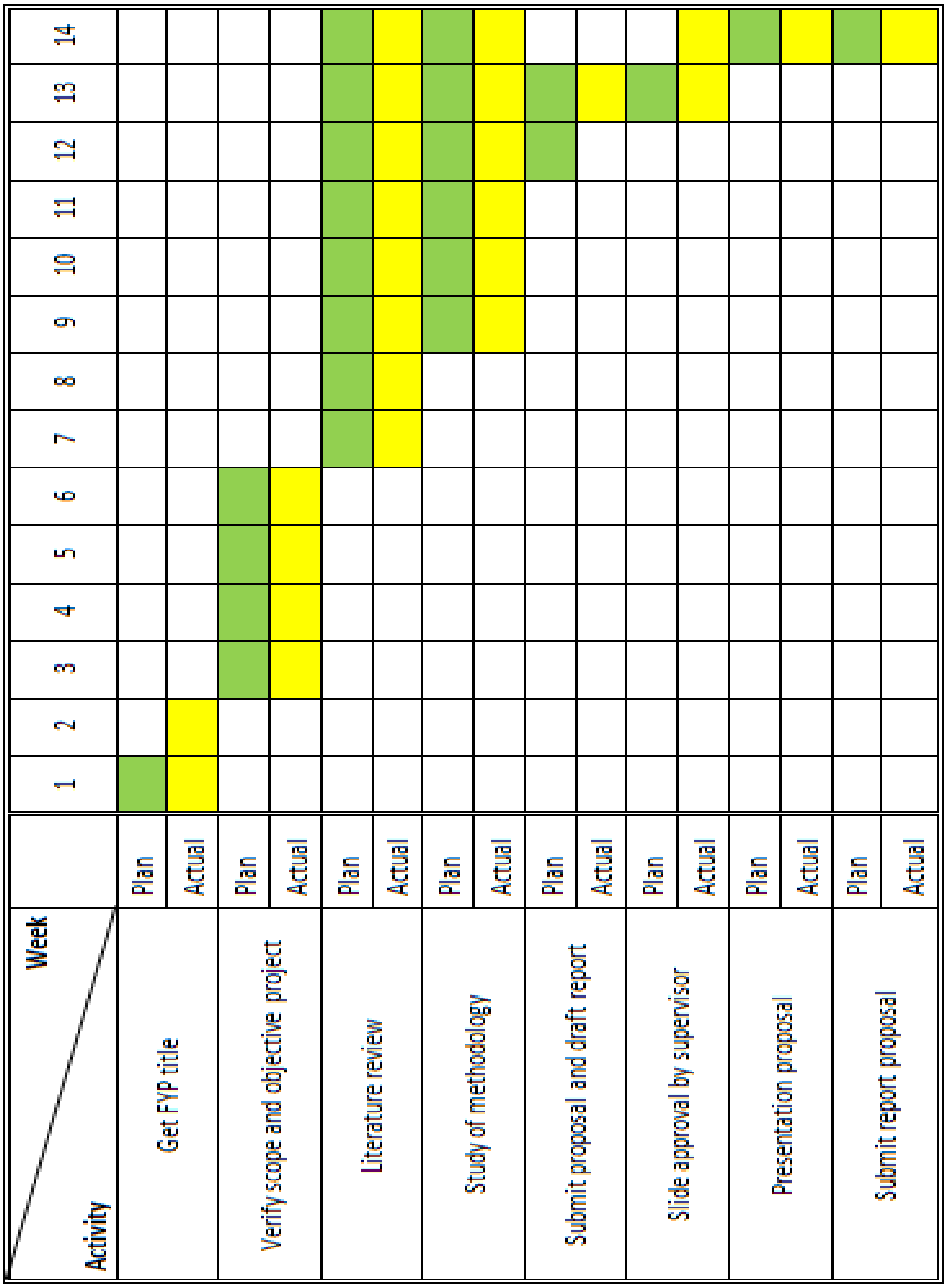
APPENDIX D

GRAPHICAL USER INTERFACE FOR CALIBRATION PROCESS



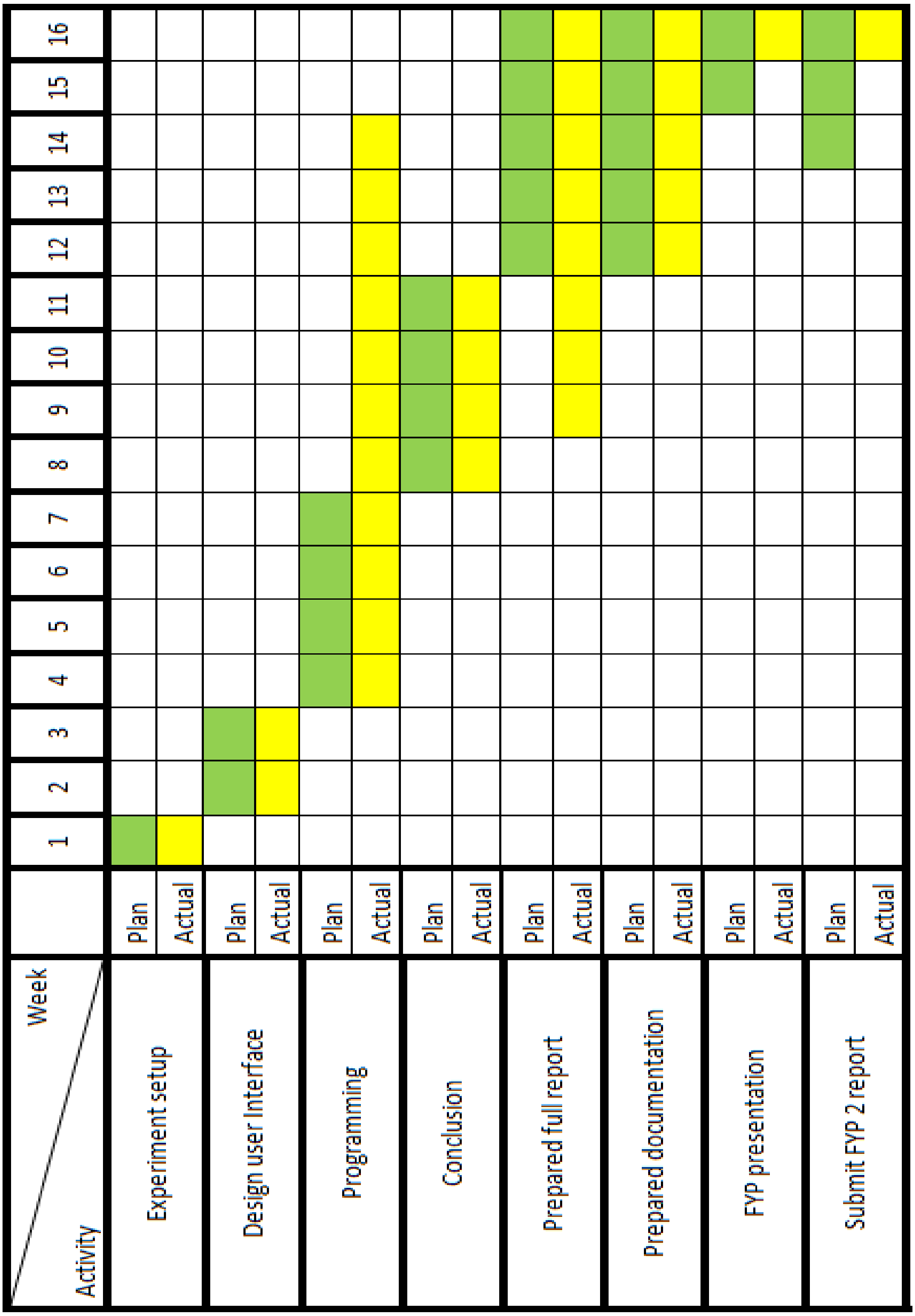
APPENDIX E1

GANTT CHART FOR PSM 1



APPENDIX E2

GANTT CHART FOR PSM 2



APPENDIX F

ULN DATASHEET

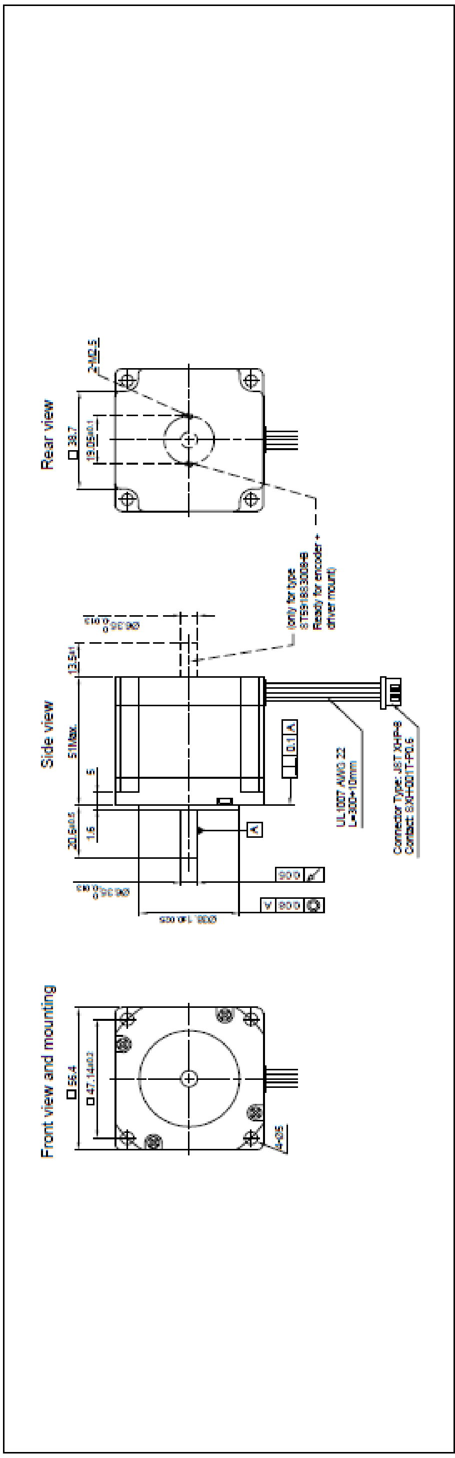
ULN2803 ULN2804

ELECTRICAL CHARACTERISTICS (T_A = 25°C, unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
Output Leakage Current (Figure 1)		I_{CEX}				µA
(V_O = 50 V, T_A = +70°C)	All Types		–	–	100	
(V_O = 50 V, T_A = +25°C)	All Types		–	–	50	
(V_O = 50 V, T_A = +70°C, V_I = 6.0 V)	ULN2802		–	–	500	
(V_O = 50 V, T_A = +70°C, V_I = 1.0 V)	ULN2804		–	–	500	
Collector–Emitter Saturation Voltage (Figure 2)		$V_{CE(sat)}$				V
(I_C = 350 mA, I_B = 500 µA)	All Types		–	1.1	1.6	
(I_C = 200 mA, I_B = 350 µA)	All Types		–	0.95	1.3	
(I_C = 100 mA, I_B = 250 µA)	All Types		–	0.85	1.1	
Input Current – On Condition (Figure 4)		$I_{I(on)}$				mA
(V_I = 17 V)	ULN2802		–	0.82	1.25	
(V_I = 3.85 V)	ULN2803		–	0.93	1.35	
(V_I = 5.0 V)	ULN2804		–	0.35	0.5	
(V_I = 12 V)	ULN2804		–	1.0	1.45	
Input Voltage – On Condition (Figure 5)		$V_{I(on)}$				V
(V_{CE} = 2.0 V, I_C = 300 mA)	ULN2802		–	–	13	
(V_{CE} = 2.0 V, I_C = 200 mA)	ULN2803		–	–	2.4	
(V_{CE} = 2.0 V, I_C = 250 mA)	ULN2803		–	–	2.7	
(V_{CE} = 2.0 V, I_C = 300 mA)	ULN2803		–	–	3.0	
(V_{CE} = 2.0 V, I_C = 125 mA)	ULN2804		–	–	5.0	
(V_{CE} = 2.0 V, I_C = 200 mA)	ULN2804		–	–	6.0	
(V_{CE} = 2.0 V, I_C = 275 mA)	ULN2804		–	–	7.0	
(V_{CE} = 2.0 V, I_C = 350 mA)	ULN2804		–	–	8.0	
Input Current – Off Condition (Figure 3) (I_C = 500 µA, T_A = +70°C)	All Types	$I_{I(off)}$	50	100	–	µA
DC Current Gain (Figure 2) (V_{CE} = 2.0 V, I_C = 350 mA)	ULN2801	h_{FE}	1000	–	–	–
Input Capacitance		C_I	–	15	25	pF
Turn–On Delay Time (50% E_I to 50% E_O)		t_{on}	–	0.25	1.0	µs
Turn–Off Delay Time (50% E_I to 50% E_O)		t_{off}	–	0.25	1.0	µs
Clamp Diode Leakage Current (Figure 6) (V_R = 50 V)	T_A = +25°C	I_R	–	–	50	µA
	T_A = +70°C				100	
Clamp Diode Forward Voltage (Figure 7) (I_F = 350 mA)		V_F	–	1.5	2.0	V

APPENDIX G

STEPPER MOTOR SPECIFICATION



SPECIFICATION \ CONNECTION	UNIPOLAR OR BIPOLAR-1 WINDING	BIPOLAR SERIAL	BIPOLAR PARALLEL
VOLTAGE (VDC)	2.16		
AMPS/PHASE	3.0	2.12	4.24
RESISTANCE/PHASE (Ohms)@25°C	0.72±10%	1.44±10%	0.36±10%
INDUCTANCE/PHASE (mH) @1KHz	1.1±20%	4.4±20%	1.1±20%
HOLDING TORQUE (Nm) [lb-in]	0.65 [5.75]	0.92 [8.14]	0.92 [8.14]
DETENT TORQUE (Nm) [lb-in]	0.03 [0.266]		
STEP ANGLE (°) ± ACCURACY	1.8±5% (NON-ACCUM)		
BACK-EMF (V) (300 U/min)		9.59 min.	
ROTOR INERTIA (Kg-m²) [lb-in²]	2.75x10⁻⁵ [0.094]		
WEIGHT (Kg) [lb]	0.65 [1.43]		

TEMPERATURE RISE: MAX.80°C (MOTOR STANDSTILL; FOR 2 PHASE ENERGIZED)
AMBIENT TEMPERATURE -10°~ 50°C [14°F ~ 122°F]
INSULATION RESISTANCE 100 MOhm (UNDER NORMAL TEMPERATURE AND HUMIDITY)
INSULATION CLASS B 130° [266°F]
DIELECTRIC STRENGTH 500VAC FOR 1 MIN. (BETWEEN THE MOTOR COILS AND THE MOTOR CASE)
AMBIENT HUMIDITY MAX. 85% (NO CONDENSATION)

PERMISSIBLE RADIAL+AXIAL FORCE

ROTOR SPRING-MOUNTED IN AXIAL DIRECTION — SPRING WASHER — BEARING

AXIAL-FORCE Fa (N)	Fa=15			
DISTANCE a (mm)	5	10	15	20
RADIAL-FORCE Fr (N)	130	90	70	52
	AXIAL		RADIAL	
SHAFT PLAY (mm)	0.08		0.02	
AT LOAD MAX: (N)	4.5		4.5	

TYPE OF CONNECTION (EXTERN): UNIPOLAR / BIPOLAR (1WINDING, SERIAL, PARALLEL)

LEADS	WINDING
BLK	A
BLK/WHT	
GRN/WHT	
GRN	A\
RED	B
RED/WHT	
BLU/WHT	
BLU	B\

FULL STEP 2 PHASE-Ex., WHEN FACING MOUNTING END (X)

STEP	A	B	A\	B\
1	+	+	-	-
2	-	+	+	-
3	-	-	+	+
4	+	-	-	+

CW / CCW

WIRING DIAGRAM: (A) BLK, BLK/WHT, GRN/WHT, (A\) GRN, (B) RED, RED/WHT, BLU/WHT, (B\) BLU

REV	DESCRIPTION	DATE	APVD

NANOTEC: ST5918S3008

SCALE FREE — X ±0.5, 1PL ±0.2, 2PL ±0.1, ANGLE ±30'

	SIGNATURE	DATE
APVD	S.Ha.	19.03.07
CHKD		
DRN	J.W.	21.11.06

STEPPING MOTOR

DWG.NO ST5918S3008