Design and Implementation of a Feedback Control System for a Stepper Motor

A Thesis Submitted in Partial Fulfillment for the Requirement of the Degree of M.Sc. in Mechatronics Engineering

By:
Fatma Awad Abdalhamid Mohammed

Supervisor:
Dr. Abdelfattah Bilal

January 2016
قَالَ ﺗﻌﺎﻟﻰ: {اللَّهُ الَّذِينَ آمَنُوا مِنْ كُمْ وَ الَّذِينَ أُوْتُوْا اﻟْﻌِﻠْمَاءِ ﷺ} 
{شَيْئَ ﻣَا ﻣَعِيتَ ﻋَنْ ذَٰﻟِكَ ﻦَبِيًّا ﻋَلِيمًا} 
{المجادلة: 11}
DEDICATION

I dedicate this research with love to my parents the reason of what I become today thanks for your great support and Continuous care.
My sisters & brother this humble work is a sign of my love to you!
ACKNOWLEDGEMENT

I would like to express my sincere indebtedness and gratitude to my thesis advisor Dr. Abdalfattah Bilal for the ingenious commitment, encouragement and highly valuable advice he provided me from initial to the final level. Enabled me to develop and understanding of the subject. Am highly thankful to his guiding me in this project.

My thanks also go to my mother and father, for they always love and encouragement.

Last but not least, I would like to thank all my friends for they always support during my hard time.
Abstract

Stepper motors are motors that indexes by digital pulses that turn their rotors a fixed fraction of a revolution where they will be clamped securely by their inherent holding torque. Stepper motors are a cost-effective and a reliable choice for many applications such as CNC and Robotics. They are used as open loop control systems which are in no need for feedback devices. Being open loop control systems, their major drawback is that if they were not used within their specified range of load and speed, then their positioning accuracy will be jeopardized. However, a feedback loop can improve the positioning accuracy of a stepper motor without incurring the higher costs of a complete servo system. In this research work a stepper motor position control system was designed and implemented using a computer and an Arduino microcontroller as a monitoring controller, which will detect the position of the motor shaft via an optical encoder. The computer was used to input the data and to display a message if the stepper motor`s operation was interrupted. This control system was tested to detect if the motor was stopped or delayed for a casual reason and then derive the motor to the correct required position. Also the system was tested for accuracy. A steady error of 10% was detected and was attributed to the poor resolution of the encoder used.
المستخلص

مواتر الخطوة هي مواتير تتحرك حركة تقسيم باستخدام نبضات رقمية تقوم بإدارة عضوها الدوار خلال كسر من الدورة وعندما تثبت في مكانها بحزم نتيجة للعزم القابض المؤثر بطبعها. مواتر الخطوة اختيار رخيص ومعتمد عليه وتستخدم في التحكم المفتوح من غير تغذية راجعة. كونها تعمل بالتحكم المفتوح فان عيبها في حقيقة أنه إذا لم يتم استخدامها في مدى التحميل والسرعة المخصصة لها، فإن دقته في التوضيع تتعرض للخطر. على كل حال اضافة تغذية راجعة لمواتر الخطوة سيعزد من دقة توضعها من غير تحمل تكلفه نظام سيرفو كامل. في هذا العمل البحثى تم تصميم وتنفيذ نظام تحكم في الوضع بموتور خطوة باستخدام حاسب ومتحكمة دقيقة نوع اردينو كتمحكمة مراقبة حيث تقوم بمتابعة وضع عمود الموتور عبر مشفر ضوئى. استخدم الحاسب لادخال البيانات ولاظهار رسالة في حالة إذا تمت مقاطعة عمل موتور الخطوة. اختبر نظام التحكم هذا في متابعة ما إذا كان عمود الموتور قد تم توقفه أو تأخيره لاي سبب طارئ ومن ثم قيادة الموتور للوضع الصحيح المطلوب. تم أيضا اختيار دقة النظام ووجد ان هناك خطأ ثابت بمقدار 10% وعذى ذلك للتكرير الضعيف للمشير المستخدم.
# Table of contents

<table>
<thead>
<tr>
<th>Number</th>
<th>Contents</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>تاﺔ</td>
<td>I</td>
</tr>
<tr>
<td></td>
<td>DEDICATION</td>
<td>II</td>
</tr>
<tr>
<td></td>
<td>ACKNOWLEDGEMENT</td>
<td>III</td>
</tr>
<tr>
<td></td>
<td>ABSTRACT</td>
<td>IV</td>
</tr>
<tr>
<td></td>
<td>ABSTRACT IN ARABIC</td>
<td>V</td>
</tr>
<tr>
<td></td>
<td>CONTENTS</td>
<td>VIII</td>
</tr>
<tr>
<td></td>
<td>TABLE OF LIST OF FIGURES</td>
<td>VIII</td>
</tr>
<tr>
<td></td>
<td>LIST OF Tables</td>
<td>IX</td>
</tr>
<tr>
<td></td>
<td><strong>Chapter one: Introduction</strong></td>
<td></td>
</tr>
<tr>
<td>1.1</td>
<td>General</td>
<td>1</td>
</tr>
<tr>
<td>1.2</td>
<td>problem statement</td>
<td>1</td>
</tr>
<tr>
<td>1.3</td>
<td>objectives</td>
<td>2</td>
</tr>
<tr>
<td>1.4</td>
<td>problem statement</td>
<td>2</td>
</tr>
<tr>
<td>1.5</td>
<td>literature review</td>
<td>3</td>
</tr>
<tr>
<td>1.6</td>
<td>layouts</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td><strong>Chapter Two: Theoretical back ground</strong></td>
<td></td>
</tr>
<tr>
<td>2.1</td>
<td>Introduction</td>
<td>5</td>
</tr>
<tr>
<td>2.2</td>
<td>Control system</td>
<td>5</td>
</tr>
<tr>
<td>2.2.1</td>
<td>Open loop control</td>
<td>6</td>
</tr>
<tr>
<td>2.2.2</td>
<td>Closed loop control</td>
<td>6</td>
</tr>
<tr>
<td>2.3</td>
<td>stepper motor</td>
<td>7</td>
</tr>
<tr>
<td>2.3.1</td>
<td>Stepper motor advantage and disadvantage</td>
<td>8</td>
</tr>
<tr>
<td>2.4</td>
<td>Applications</td>
<td>8</td>
</tr>
<tr>
<td>2.5</td>
<td>Fundamentals of operation</td>
<td>8</td>
</tr>
<tr>
<td>2.6</td>
<td>Types of stepper motor</td>
<td>10</td>
</tr>
<tr>
<td>2.6.1</td>
<td>Permanent magnet</td>
<td>11</td>
</tr>
<tr>
<td>2.6.2</td>
<td>Variable reluctance</td>
<td>11</td>
</tr>
<tr>
<td>2.6.3</td>
<td>Hybrid stepper motor</td>
<td>13</td>
</tr>
<tr>
<td>2.7</td>
<td>Types of windings</td>
<td>13</td>
</tr>
<tr>
<td>2.7.1</td>
<td>Unipolar</td>
<td>13</td>
</tr>
<tr>
<td>2.7.2</td>
<td>Bipolar</td>
<td>15</td>
</tr>
</tbody>
</table>
### 2.8 Comparison between drivers

### 2.9 Compression between types of stepper motor

### 2.10 Step angle

### 2.11 Modes of excitation

#### 2.11.1 Full step

#### 2.11.2 Half step

#### 2.11.3 Micro step

### 2.12 Stepper Motor Specifications and Speed-Torque Characteristics

<table>
<thead>
<tr>
<th>Chapter three: Hardware components</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>3.1</strong></td>
</tr>
<tr>
<td><strong>3.2</strong></td>
</tr>
<tr>
<td><strong>3.2.1</strong></td>
</tr>
<tr>
<td><strong>3.3</strong></td>
</tr>
<tr>
<td><strong>3.4</strong></td>
</tr>
<tr>
<td><strong>3.5</strong></td>
</tr>
<tr>
<td><strong>3.6</strong></td>
</tr>
<tr>
<td><strong>3.7</strong></td>
</tr>
<tr>
<td><strong>3.8</strong></td>
</tr>
<tr>
<td><strong>3.9</strong></td>
</tr>
<tr>
<td><strong>3.10</strong></td>
</tr>
<tr>
<td><strong>3.11</strong></td>
</tr>
</tbody>
</table>

### Chapter Four: Results and Conclusion

#### 4.1 Results

#### 4.2 Conclusion

<table>
<thead>
<tr>
<th>Chapter Five: Conclusion and recommendations</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>5.1</strong></td>
</tr>
<tr>
<td><strong>5.2</strong></td>
</tr>
<tr>
<td>Appendixes</td>
</tr>
</tbody>
</table>
Chapter I
Introduction

1.1 General
Stepper motors are characterized by being indexed by digital pulses that turn their rotors a fixed fraction of a revolution where they will be clamped securely by their inherent holding torque. Stepper motors are cost-effective and reliable choices for many applications such as CNC and Robotics. They are simple, cheap, durable and compatible with digital control systems. They are used as open loop control systems which are in no need for feedback devices. Being open loop control systems, their major drawback is that if they were not used within their specified range of load and speed, then their positioning accuracy will be jeopardized. That is if the robot or CNC machine was erroneously over loaded due to any casual reason, their response will be unknown. However, a feedback loop can improve the positioning accuracy of a stepper motor without incurring the higher costs of a complete servo system.

1.2 Problem Statement
In control practice, many alternative solutions are available and many theories are established. The challenge of the control designer is to choose a suitable effective and economic solution for his control problem. position control systems for the movable machine parts of industrial robots and machine tools using open loop control systems implemented through stepper motor requires programmers of high technological skills in order to command these devices with in their
specified speed load limitations. Otherwise these machines may show a low accuracy response.

1.3 Objectives

The main purpose of this work is to design and implement a stepper motor position control system using a computer and electronic circuit as a monitoring controller, which will detect the position of the motor shaft via an optical encoder and computer. The purpose of the control system is to detect if the motor was stopped or delayed for a casual reason and then derive the motor to the correct required position.

1.4 Methodology

In this research a control circuit will be designed and an experimental work will be carried out to check the accuracy of the suggested control circuit. The control circuit will consist of a stepper motor which shaft is connected to an optical encoder via a gearbox (n turns of the encoder shaft for each turn of the stepper motor’s shaft, so as to increase the resolution of the control system). The pulses generated by the encoder will be fed back to the control circuit which is interfaced to a PC equipped with a control program and a user interface screen. A block diagram of the proposed control system is shown in Figure (1).
1.5 literature review

Sahar Ahmed Altom [1] in her M.Sc. research titled. "motor control using microcontroller 2012 " using at mega 16 to provide high positive resolution for a stepper motor control system due to its capability of controlling stepper motors with very precise resolution.

Mohammed Khider Khalifa [2] in his M.Sc. research titled. "Designing a remote control system for stepper motor using micro controller", the micro controller is used to control the speed direction and step angle of the stepper motor by entering different commands through the key pad.

Hussein Sarhan [3] in paper titled " PLC- controlled stepper motor drive for NC positioning system "A PLC used as an indexer to perform full stepping, half stepping and micro stepping mode of operation of stepper motor. The drive system has been implemented in two axes NC positioning system. The linear displacement for each machine axis has been evaluated and compared for the three stepping modes of stepper motor. The experimental and calculation results show that the response per pulse (Resolution) for full stepping mode is 8.35000μm, for half stepping mode is 4.17500μm and for micro stepping mode is 1.04375μm, which means that the micro stepping mode is the best one for precision control of NC positioning system.

In paper "Design and Development stepper motor position control system using Atmel 85c51 Microcontroller" [4]. A closed loop position control. A system is described by microcontroller program which can control the position of stepper motor properly. The microcontroller is
interfacing with ADC 0804 IC. This ADC interface is provided to set the position of the stepper motor angle. Drive the stepper motor for each step and read the output of opto-slot for movement of each step. Count the step value and display in LCD.

1.5 Layout

This research consists of five chapters. Chapter one literature review and the proposal approach of the project, chapter two presents the theoretical background review of the project, Chapter three illustrates the control system feedback control of stepper motor and illustrate the Arduino software, Chapter four include the results and conclusion, and Chapter five include the recommendations and future work.

Chapter two
Theoretical background

2.1 Introduction

On Power Electronics Specialists Conference 1989 [5]. A paper titled "A DC motor position control using sliding mode and disturbance estimator", introduced a robust DC motor position control system, incorporating a disturbance estimator for a second-order system. The
control law is numerically implemented for position control. The proposed algorithm is easy to design and implement. Simulation results have shown its robustness toward parameter variations, good dynamic response to a step in load torque, and a null static error.

Martindale, Shawnee, KS (2005) [6]. In his work titled "Controlling Antennas Powered by AC or Large DC motors with the RC1500 or RC2000 antenna controllers", introduces The RC1500A, RC2000A, and RC2000C antenna controllers, which are designed to control satellite antennas, powered by 36 volt DC motors which use pulse type position sensors. The RC1500A is designed for use with single axis polar mount antennas.

Rehab [7], in her M.Sc. research titled." Building a Computerized Position Control System Using DC Motor". Managed to build a model for the position control of a DC motor using an optical encoder.

2.2 Control system

A control system is an interconnection of components forming a System configuration that will provide a desired system response. The basis for analysis of a system is the foundation provided by linear system, which assumes a cause–effect relationship for the components of a system. A component or process to be controlled can be represented by a block diagram as shown in Figure (2.1).
There are two types of control system

2.2.1 Open loop control

Also known as non-feedback controls system because it does not use feedback to determine if its output has achieved the desired goal of the input. An open loop control system cannot correct any errors that it could make, and it often used in simple processes because of its simplicity and low cost especially in system that feedback is not necessary. This type of control system utilizes a control actuator to desired response as shown in Figure (2.2).

2.2.2 Close loop control

This system compares the actual output with the desired output response. The measure of the output is called the feedback signal. A feedback control system that tends to maintain a relationship of one system variable to another by comparing functions of these variables and using the difference as a means of control as shown in Figure (2.3).
2.3 Stepper motor

Stepper motor is a permanent magnet motor with many poles, designed to rotate through a specific angle for each electrical pulse received from the driver unit. Therefore the motor's position can be controlled precisely without any feedback mechanism, as long as the motor is carefully sized to the application. This type of control eliminates the need for expensive sensing and feedback devices such as optical encoders. The position is known simply by keeping track of the input step pulses. It is one of the most versatile forms of positioning systems. They are typically digitally controlled as part of an open loop system, and are simpler and more rugged than closed loop servo systems.

2.3.1 stepper motor advantages and disadvantages

There are many advantages and disadvantages of stepper motors for example.
a. Advantages

1. Simple
2. Low cost
3. Rugged construction that operates in almost any environment
4. High Reliability
5. High torque at low speed

b. Disadvantages

1. Resonances can occur if not properly controlled.
2. Not easy to operate at extremely high speeds.

3.4 Applications of stepper motor

Stepper motor used in many applications such as:

1. Control
2. Instrumentation
3. CNC machines
4. Computer peripherals devices
5. Robotics

2.5 Fundamentals of operation

A stepper motor converts electrical pulses into specific rotational movements. The movement created by each pulse is precise and repeatable, which is why stepper motors are so effective for positing applications. Permanent Magnet stepper motors incorporate a
permanent magnet rotor, coil windings and magnetically conductive stators energizing a coil winding creates an electromagnetic field with a north and South Pole. The magnetic field created by the winding will cause the magnetized rotor to align itself with the magnetic field, since unlike poles attract. The direction of the magnetic field can be altered to create rotation of the rotor.

Figure 2.4: Step sequence for a two-phase motor (Wave Drive)

Figure (2.4) Illustrates a typical step sequence for a two phase motor. In Step 1, phase A is energized; it locks the rotor in the position shown. In Step 2, phase A is turned off and phase B is turned on, the rotor rotates 90° clockwise. In Step 3, phase A is turned on again but with reversed
polarity and in Step 4, phase B is turned on with reversed polarity. This sequence completes a full turn of the rotor. Repeating this sequence causes the rotor to rotate clockwise in 90° steps.

![Figure 2.5: Step sequence for a two-phase motor (Full Step Drive)](image)

Figure 2.5. Shows a more common “two phases on” stepping where both phases are always energized. The rotor in this stepping mechanism alights itself between the poles. This stepping method gives 41.4% more
torque than “one phase on” stepping but requires twice the input power. [8]

2.6 Types of stepper motor

There are three types of stepper motor classified according to machine structure and principles of operation.

2.6.1 Permanent Magnet

Called permanent magnet because the rotor made from permanent magnet they are widely used in printers, copies, and scanners, among other applications. The PM stepper motor requires no “teeth” as are typically found in the variable reluctance (VR) stepper motor. This makes the permanent magnet stepper an extremely popular choice for many motor applications. Figure (3.6) shown a PMSM which employs a cylindrical permanent magnet as the rotor and possesses four poles in its stator. Two overlapping windings are wound as one winding on poles A and A' and these two windings are separated from each other at terminals to keep them as independent windings. The same is true for poles B and B'. The terminals denotes "common" to be connected to the positive terminal of the power supply When the windings are excited in the sequence A – B – A’ – B’ --- the rotor will be driven in a clockwise direction. The step length is 900 in this machine. If the number of stator teeth and magnetic poles on the rotor are both doubled, a two-phase motor with a step length of 450 will be realized.
2.6.2 Variable-reluctance (VR) step motor

It consists of a soft iron multi-toothed rotor and a wound stator. When the stator windings are energized with DC current, the poles become magnetized. Rotation occurs when the rotor teeth are attracted to the energized stator poles.
In this type stator has 6 poles and rotor has four poles as shown in figure (2.7b), which each winding wraps around two opposite poles. When winding number 1 is energized, the x teeth are attracted to poles numbered 1. When winding 1 is turned off and winding 2 is turned on, the rotor will rotate 30 degrees so that the y teeth will line up with poles numbered 2 simply applying power to 3 windings in the sequence will make this rotor rotate continuously. Based on the difference in construction methods.

2.6.3 Hybrid step motor
Besides the two step motor types discussed previously, hybrid step motors are the most widely used in all variety of industrial applications. Hybrid motors combine the best characteristics of the variable reluctance and permanent magnet motors. They are constructed with multi-toothed stator poles and a serrated permanent magnet rotor. When energized, electrical current in the coils around each stator creates an electromagnetic pole in the stator and the teeth in the rotor line up with the serrated teeth in the stator. Compared with the other two types, hybrid step motors offer finer resolution. Standard hybrid motors have 200 to 400 rotor teeth and rotate at 0.9-1.8 step angles. They also exhibit high static and dynamic torque.

Figure 2.8: Hybrid stepper motor

2.7 Types of winding

There are different types of winding such as:

2.7.1 Unipolar motors

Unipolar stepping motors, both Permanent magnet and hybrid stepping motors with 5 or 6 wires are usually wired as shown in the figure (2.9a), with a center tap on each of two windings. In use, the center taps of the windings are typically wired to the positive supply, and the two ends of each winding are alternately grounded to reverse the direction of the field provided by that
winding. As shown in the figure, the current flowing from the center tap of winding 1 to terminal a causes the top stator pole to be a north pole whiles the bottom stator pole is a south pole.

Figure 2.9a: Unipolar stepper motor

This attracts the rotor into the position shown. If the power to winding 1 is removed and winding 2 is energized, the rotor will turn 30 degrees, or one step. To rotate the motor continuously, we just apply power to the two windings in sequence. Assuming positive logic, where a 1 means turning on the current through a motor winding, the following two control sequences will spin the motor illustrated in the figure(2.9.b).
2.7.2 Bipolar step motor

The bipolar motor is similar in construction to the Unipolar motor except that it does not have any center taps. The advantage gained in the simplicity of construction of the motor is lost in the control circuitry as that becomes more complex. A H-bridge is required for the control of the motor.

![Conceptual Model of Bipolar Stepper Motor](image)

Figure 2.10: Bipolar stepper motor

2.8 Compression between drivers

Table (2.1) Compression between drivers:

<table>
<thead>
<tr>
<th>Unipolar drive</th>
<th>Bipolar drive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Winding with a center tap, or two separate windings per phase</td>
<td>One winding per phase</td>
</tr>
<tr>
<td>Two switches per phase</td>
<td>Four switches per phase, in the form of an H-bridge</td>
</tr>
<tr>
<td>Utilizes only half the available copper volume of winding</td>
<td>Motor winding is fully energized</td>
</tr>
<tr>
<td>Incurs twice the loss of a bipolar drive at the same output power</td>
<td>Loss is minimum compared to Unipolar drive</td>
</tr>
</tbody>
</table>

2.9 comparisons between types of stepper motor:
The choice of stepper motor type depends on torque requirement, step angle, and control technique.

Table (2.2) Comparison based on motor advantages and disadvantages.

<table>
<thead>
<tr>
<th>Motor type</th>
<th>Advantage</th>
<th>Disadvantage</th>
</tr>
</thead>
</table>

Table (2.3) Comparison based on motor characteristics.

<table>
<thead>
<tr>
<th>Specifications</th>
<th>VRM</th>
<th>PMSM</th>
<th>HSM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step angle</td>
<td>0.66°-30°</td>
<td>3.75°-45°</td>
<td>0.45°-5°</td>
</tr>
<tr>
<td>phases</td>
<td>3,4,5</td>
<td>2,4</td>
<td>2,5</td>
</tr>
<tr>
<td>Drive type</td>
<td>Unipolar</td>
<td>Unipolar/Bipolar</td>
<td>Bipolar</td>
</tr>
<tr>
<td>Rotor inertia</td>
<td>Low</td>
<td>High</td>
<td>Medium</td>
</tr>
</tbody>
</table>
Table (2.4) Comparison based on different phase characteristics

<table>
<thead>
<tr>
<th>Types</th>
<th>Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 phase</td>
<td>1. Simple driver circuit with low heat dissipation.</td>
</tr>
<tr>
<td></td>
<td>2. Less step error compared to other phases.</td>
</tr>
<tr>
<td></td>
<td>3. Higher accuracy due to more number of stator Poles.</td>
</tr>
<tr>
<td>3 phase</td>
<td>1. Torque ripple is more</td>
</tr>
<tr>
<td></td>
<td>2. Poor peak torque ratio</td>
</tr>
<tr>
<td></td>
<td>3. Power transistors are less</td>
</tr>
<tr>
<td>4 phase</td>
<td>1. Low torque ripple.</td>
</tr>
<tr>
<td></td>
<td>2. Good peak torque ratio</td>
</tr>
<tr>
<td>5 phase</td>
<td>1. Lower torque ripple.</td>
</tr>
<tr>
<td></td>
<td>2. More expensive controller</td>
</tr>
</tbody>
</table>

2.10 step angle

The step angle, the number of degrees, a rotor will turn per step is calculated as follows:

Step angle ($\Theta_s$) = \(\frac{360}{S}\)

S=mNr

m=number of phases
Nr=number of rotor teeth

2.11 modes of excitation

There are three different types of excitations

2.11.1 Full Step
Full step excitation mode is achieved by energizing both windings while reversing the current alternately. Essentially one digital input from the driver is equivalent to one step. If two phases of the hybrid stepper motor are excited, the torque produced by the motor is increased, but the power supply to the motor is also increased. This can be an important consideration for applications, where the power available to drive the motor is limited.

Table (2.5.a) Full step sequence.

<table>
<thead>
<tr>
<th>Va</th>
<th>Vb</th>
</tr>
</thead>
<tbody>
<tr>
<td>+1</td>
<td>+1</td>
</tr>
<tr>
<td>+1</td>
<td>-1</td>
</tr>
<tr>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>-1</td>
<td>+1</td>
</tr>
</tbody>
</table>

Figure 2.11a: Full step excitation

2.11.2 Half step
In half step mode, the drive alternates between two phases ON and a single phase ON. Half step sequence is shown in Figure(2.11.b) and tabulated in table (2.5.b). This increases the angular resolution, but the motor also has less torque (approx. 70%) at the half step position (where only a single phase is ON). This may be mitigated by increasing the current in the active winding. The advantage of half step is that it reduces the vibration.

Table (2.5.b) half step sequence
The full step length of a stepper motor can be divided into smaller increments of rotor motion known as micro step by partially exciting several phase windings and micro step sequence is shown in Figure (2.11.c)

Micro step is typically used in applications that require accurate positioning and a fine resolution over a wide range of speeds. The major disadvantage of the micro step drive is the cost of implementation due to the need for partial excitation of the motor windings at different current levels.
2.12: Stepper Motor Specifications and Speed-Torque Characteristics:

When choosing a stepper motor for an application, the specifications must be understood such as: the stepper motor must be full

1- **Holding Torque**: The maximum steady torque that can be applied to the shaft of an energized motor without causing rotation.

2- **Detent Torque**: The maximum torque that can be applied to the shaft of a non-energized motor without causing rotation.

3- **Speed-Torque Curve**: The speed-torque characteristics of a stepping motor are a function of the drive circuit, excitation method and load inertia.

![Speed-Torque Curve Diagram]

Figure 2.12: Speed-torque curve

**1-Maximum Slew Rate**: The maximum operating frequency of the motor with no load applied.
2- **Maximum Starting Frequency:** The maximum pulse rate (frequency) at which an unloaded step motor can start and run without missing steps or stop without missing steps.

3- **Pull-out Torque:** The maximum torque that can be applied to the shaft of a step motor (running at constant speed) and not cause it to Lose step.

4- **Pull-in Torque:** The maximum torque at which a step motor can start, stop and reverse the direction of rotation without losing step. The maximum torque at which an energized step motor will start and run In synchronism, without losing steps, at constant speed.

5- **Slewing Range:** This is the area between the pull-in and pull-out torque curves where a step motor can run without losing step, when the speed is increased or decreased gradually. Motor must be brought up to the slew range with acceleration and deceleration technique known as ramping.
Chapter III

The Hardware Components

3.1 Introduction

This chapter explains beside the stepper motor the necessary components used in this project for full understanding of the system.

3.2 Arduino

Arduino is an open source computer hardware and software company, project and user community that designs and manufactures kits for building digital devices and interactive objects that can sense and control the physical world [8]

Arduino technology has many flavors, many boards of different types and features. Most common boards and their features are described in the following table.

Table (3.1): Arduino boards comparison
The Arduino Uno is a microcontroller board based on the ATmega328 [8] It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.
3.2.1 Schematic design

a- power:

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

**VIN**. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

**5V**. This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins by passes the regulator, and can damage your board. We don't advise it.

**3.3V**. Volt supply generated by the on-board regulator. Maximum current draw is 50 mA.

**GND**. Ground pins.

**IOREF**. This pin on the Arduino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs for working with the 5V or 3.3V.
b- Memory

The atmega328 has 32KB (with 0.5KB used for the boot loader). It also has 2KB of SRAM and 1KB of EEPROM (which can be read and written with the EEPROM library).

![Atmega328 Pin Mapping](image)

Figure 3.2: At mega 328 pins

c- Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using pin Mode, digital Write, and digital Read functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 ohms. In addition, some pins have specialized functions:

**Serial: 0 (RX) and 1 (TX):** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
**External Interrupts 2 and 3:** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.

**PWM:** 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analog Write () function.

**SPI:** 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.

**LED 13:** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the analog Reference function.

### 3.2.2 Arduino Uno specifications

The table below shown the Arduino specifications

<table>
<thead>
<tr>
<th>Microcontroller</th>
<th>ATmega328</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating Voltage</td>
<td>5V</td>
</tr>
<tr>
<td>Input Voltage (recommended)</td>
<td>7-12V</td>
</tr>
</tbody>
</table>
### 3.3 Optical encoder

A rotary encoder, also called a shaft encoder, is an electro-mechanical device that converts the angular position or motion of a shaft or axle to an analog or digital code.[9]

There are two main types: absolute and incremental (relative). The output of absolute encoders indicates the current position of the shaft, making them angle transducers. The output of incremental encoders provides information about the motion of the shaft, which is typically further processed elsewhere into information such as speed, distance, and position.
Rotary encoders are used in many applications that require precise shaft unlimited rotation—including industrial controls, robotics, special purpose photographic lenses, computer input devices (such as opt mechanical mice and trackballs), controlled stress rheometers, and rotating radar platforms.

Encoders may be implemented using a variety of technologies:

- **Conductive.** A series of circumferential copper tracks etched onto a PCB is used to encode the information. Contact brushes sense the conductive areas. This form of encoder is now rarely seen except as a user input in digital millimeters'.
- **Optical.** This uses a light shining onto a photodiode through slits in a metal or glass disc. Reflective versions also exist. This is one of the most common technologies.
- **Magnetic.** Strips of magnetized material are placed on the rotating disc and are sensed by a Hall-effect sensor or resistive sensor. Hall Effect sensors are also used to sense gear teeth directly, without the need for a separate encoder disc.

The encoder disk is firmly connected to the back-shaft of the motor, so that both the shaft and the encoder disk rotate at the same r.p.m. (the back-shaft is an extension of the output shaft of the motor at its back, usually present for the sole purpose of adding a shaft encoder). When this encoder disk is inserted, the rotation of the motor causes the beam of light to be periodically intercepted by the solid parts of the encoder disk creating a sequence of pulses of light, which will be translated by the photo couple's receiver into pulses of electricity. Those pulses of electricity contain all the information we need to implement a closed loop control. The frequency of those pulses is directly proportional the speed of rotation of the shaft (RPM) and the number of those pulses correspond to the angular displacement of the shaft. The more the number of holes in an encoder disk, the higher will be the resolution (the slightest angular displacement that can be detected). One important factor that affects the performance of shaft encoder and thus the overall performance of a closed loop control system is the position.
Figure 3.3: Component of an optical encoder

3.4 ULN2003 Driver
The ULN2003 is an array of seven NPN Darlington transistors capable of 500mA, 50V output. It features common-cathode for switching Inductive loads. Designed for different logic input levels. [10]
3.5: Pull up resistor

A resistor used in logic circuits to ensure a well-defined logical level at a pin under all conditions. There are three logic states: high, low and floating (or high impedance). The high impedance occurs when the pin is not pulled to a high or low logic level, but is left floating "instead". [11]
3.6 Push-button

Is a simple switch mechanism for controlling some aspect of a machine or a process. Buttons are typically made out of hard material, usually plastic or metal. As shown in Figure (3.6), the surface is usually flat or shaped to accommodate the human finger or hand, so as to be easily depressed or pushed. Buttons are most often biased switches, though even many un-biased buttons (due to their physical nature) require a spring to return to their un-pushed state.

3.7: light-emitting diode (LED)

Is a two-lead semiconductor light. It is a junction diode, which emits light when activated. When a suitable voltage is applied to the leads, electrons are able to recombine with electron holes within the device, releasing energy in the form photons.
3.8 principle of operation

This system includes hardware and software. The hardware consists of several items (Stepper motor, Arduino board, ULN2003, Pull up resistor, Push button light emitted diode). An optical encoder (incremental encoder) was designed with a certain angle determined respect to the application that needed to use. This optical encoder is associated with a Stepper motor. The motor control circuit is connected to an Arduino board, via an ULN2003A. The shaft encoders attached to the motor consist of a metal or plastic disc with holes, in this case the reflective opto switch was used. In the case of the reflective opto switch the infrared beam is only detected by the phototransistor if it is reflected by an object close to the switch. Pins (5, 4, 3) works as indicators. The digital pin connected with two 100kΩ resistance works as voltage divider and current limiter. When connected the circuit with PC the stepper move with the input value and rotate the encoder disk. if the stepper stopped with external force or delay the encoder calculate the delay or stopped `1shown in the screen. The software (written in C) has two functions: to control the motors and to read the status of the encoder.
3.9 Flow chart

The flow chart identify how the system is work.

Start

Include stepper library

Variables

Configure I/O

Get SW value

Wait input from PC

Return to Zero Level
Process input value

If stepper
Delay

Yes

Yes

Calculate delay time
Add delay to input value

No

Yes

Calculate stop time
Add stop to input value

No
3.10 Arduino Code

The Arduino code is programmed by using C language.

```c
#include <Stepper.h>

int step sPerRevolution = 64
// change this to fit the number of steps per revolution;
// for your motor

Initialize the stepper library on pins 8 through 11: //
Step permy Stepper (step sPer Revolution, 11, 9, 10, 8)
Con stint led1 = 3;
Con stint led2 = 4;
Cons tint led3 = 5;
Con stintir = A0;
Con stintth = 900;
long y;
long z;
Con stint sw1 = 6 ;
Con stint sw2 = 7 ;
int a = 0 ;
```
int b = 0;
int c = 0;
int incoming Byte;

void setup ( ) {
set the speed at 60 rpm: //
; )my Step per .set Speed(100
initialize the serial port: //
Serial .begin (9600) ;
; ) pin Mode (led1, OUTPUT
; ) pin Mode (led2, OUTPUT)
; ) pin Mode (led3, OUTPUT)
; ) pin Mode (sw1, INPUT
; ) pin Mode (sw2, INPUT
; ) pin Mode (ir, INPUT{

void loop ( ) {
; ) int x = analog Read (ir ) if (x > th
digital Write(led2,HIGH)
} else;

digital Write(led2,LOW) ;
}

if (Serial. available() > 0 {
// read the incoming byte:
'0' ; int in coming Byte = Serial. read
say what you got ://
Serial .Print("I received: " ) ;
; )Serial. Print(in coming Byte .DEC
; Serial .print ln (" motor is running..")
; in coming Byte = incoming Byte * 40
{ + +)for (c = 0; c < incoming Byte; c
Z ++;
{} if (z > 7
") ; Serial .print ln (" STOP PROBLEM
; )digital Write ( led1 , HIGH( 
; )digital Write ( led2 , HIGH 
; )digital Write (led3 , HIGH 
;delay (200 )
; ) digital Write ( led1 , LOW 
; ) digital Write (led2, LOW 
digital Write(led3 , LOW) ;
; )delay(200 
;digital Write(led1 , HIGH) 
; )digital Write(led2 , HIGH
) ;digital Write(led3 , HIGH Delay (200);
digital Write (led1 , LOW)
digital Write (led2 , LOW)
digital Write(led3 , LOW) ;
delay (200)
; }break
{}if (y > 12
; "Serial .print ln ("  DELAY PROBLEM"
; ) digital Write(led1 , HIGH
; ) digital Write(led2 , HIGH
; ) digital Write ( led3 , HIGH )
delay (200)
; ) digital Write ( led1 , LOW
digital Write ( led2 , LOW)
; ) digital Write( led3 , LOW
; ) delay (200;
; ) digital Write ( led1 , HIGH
digital Write ( led2 , HIGH)
; ) digital Write ( led3 , HIGH
; ) delay (200
; ) digital Write ( led1 , LOW
digital Write (led2 , LOW)
; ) digital Write ( led3 , LOW
; delay (200
}

; int x = analog Read (ir
{)if (x > th {
   digital Write(led2,HIGH );
   ;++y
   ;z = 0
}
else
{
   ; digital Write (led2,LOW
}
; my Stepper .step (51)
}

Delay (incoming Byte);//

; y=0
; delay (10
step one revolution in one direction://
a = digital Read (sw1);
; b = digital Read (sw2;
digital Write(led1 , LOW)
; digital Write(led3, LOW)
; delay (60)
{if (a == LOW)
; digital Write(led1, HIGH)
; digital Write(led2, LOW)
; digital Write(led3, LOW)
Serial.printLn("manual clockwise");
; my Stepper.step(stepsPerRevoution
Delay (200);{
{if (b == LOW)
digital Write(led1, LOW);
; digital Write(led2, LOW)
; digital Write(led3, HIGH)
:step one revolution in the other direction//
Serial.printLn("manual counterclockwise");
; my Stepper.step(-steps PerR evolution
; delay (200)
}

z = 0
}
}
Chapter Four

Results & Conclusion

4.1 Results

Table 4.1: (comparison between the actual and the desired position)

<table>
<thead>
<tr>
<th>Actual Position</th>
<th>Command Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.9</td>
</tr>
<tr>
<td>2</td>
<td>1.8</td>
</tr>
<tr>
<td>3</td>
<td>2.9</td>
</tr>
<tr>
<td>4</td>
<td>3.85</td>
</tr>
</tbody>
</table>

Figure 4.1 (position Line Chart)
Figure 4.2 (position chart)

4.2 Conclusion

Chapter three was explained the components that are used in this project. And explained how they tied together, the flow chart and the Arduino program.

This system includes hardware and software. The hardware component is (Stepper motor, Arduino, Encoder, ULN2003Pull up resistor, Push button light emitted diode). An optical encoder was designed with a certain angle determined respect to the application that needed to use. This optical encoder is associated with a Stepper motor. The motor control circuit is connected to an Arduino board, via an ULN2003. The encoder attached to the motor shaft consists of a metal or plastic disc with holes. The reflective opto switch was used, In the case of the reflective opto switch the infrared beam is only detected by the phototransistor if it is reflected by an object close to the switch. When connected the circuit with PC the stepper move with the input value and rotate the encoder disk. if the stepper stopped with external force or delay the encoder calculate the delay or stopped shown in the screen. The software (written in C)
has two functions: to control the motors and to read the status of the encoder.

Chapter Five

Recommendations and future works

- Design of feedback control of stepper motor using encoder deserves further research and investigation.

- Stepper motor and encoder can be used in many applications and as a future work this project can be designed as a helpful unit connected with wireless circuit to make alarm or notification if there is a problem happened in the system.

- Using a proper encoder will give accurate results.


