POSITION MOVEMENT AND SPEED CONTROL BY STEPPER MOTOR WITH MICROCONTROLLER PIC16F877

Oleh

Kang Pin Chye

Disertasi ini dikemukakan kepada
UNIVERSITI SAINS MALAYSIA

Sebagai memenuhi sebahagian daripada syarat keperluan
untuk ijazah dengan kepujian

SARJANA MUDA KEJURUTERAAN (KEJURUTERAAN ELEKTRIK)

Pusat Pengajian Kejuruteraan
Elektrik dan Elektronik
Universiti Sains Malaysia

Mei 2006

ABSTRACT

The usage of the stepper motor in our daily life has been increasing and can be found easily in our daily life. It shows the important and the benefit of precise movement and speed control of stepper motor. In such, this project has been carry out. This Project mainly objective is to develop a digital microcontroller to control the position and speed of the stepper motor. The input of the motor is control by using keypad while the program display by using the LCD screen. The hardware implementation is done by using the belt that couple with the roller to determine the position movement of the stepper motor .The belt is painted by white dot to represent the pointer and the casing will be number accordingly to the number of the keypad to show the accuracy of the position control. Simulation of the circuit had been done by using PSIM with the stepper motor is replace by the equivalent resistance and inductance according to the motor datasheet. The simulation is carries out to determine the correct sequence is generate to the stepper motor so that it can rotate in the right manner. The wrong step sequence will cause the motor rotate in jerky manner. This Project consist of three main parts include PIC microcontroller, Translator Circuit and Motor Driver Circuit. The PIC microcontroller mainly purpose is to receive input from the keypad and develop output for the purpose of generating the pulse and direction control. The pulses which are sent from the PIC controller will use to triggered the transistor that will act as switch to supply the motor require voltage. The translator circuit will responsible to generate the stepping sequence that needed by the motor. Finally, the driver circuit is to handle the current drawn by the motor windings to avoid any overloaded. In Sum, it can proved that the stepper motor can develop a precise position movement and speed control by using PIC16F877 and can be practice in daily life.

ABSTRAK

Penggunaan motor pelangkah dalam kehidupan harian kita semakin meningkat dan dapat ditemui dengan senang dalam kehidupan seharian.Ini menunjukkan kepentingan untuk mengawal ketepatan dan kelajuan motor pelangkah untuk tujuan pelbagai kegunaan harian. Oleh itu, projek ini dijalankan bagi tujuan tersebut. Tujuan utama projek ini dijalankan adalah bagi tujuan penjanaan suatu mikropengawal berdigit bagi mengawal kedudukan dan kelajuan motor pelangkah. Masukan bagi motor pelangkah ini dikawal oleh papan kekunci manakala keluaran pula dipaparkan menerusi skrin LCD. Perkakasan bagi projek ini diimplementasikan dengan menggunakan tali getah yang disambungkan kepada suatu pemutar untuk menentukan kedudukan motor tersebut. Getah yang disambungkan itu ditanda dengan warna putih supaya jelas kelihatan kedudukan asal dan kedudukan pergerakan seterusnya motor tersebut. Selain itu, getah yang ditanda dengan menggunakan tanda putih itu akan membantu mengurangkan rintangan terhadap pergerakan motor ke tahap minima.

Simulasi bagi litar dilakukan dengan mewakili motor dengan perintang dan kearuhan yang sesuai mengikut helaian data motor yang digunakan. Dalam projek ini, ia dapat dibahagikan kepada tiga bahagian utama iaitu Mikropengawal PIC , Litar penukar dan Litar pemandu motor. Tujuan mikropengawal ialah untuk menerima masukan daripada papan kekunci dan menjanakan denyut dan mengawal arah gerakan motor pelangkah. Litar penukar pula bertanggungjawab untuk menjanakan susuanan langkah yang tepat manakala litar pemandu motor mengawal arus yang melalui gegelung motor supaya lebihan beban tidak berlaku. Kesimpulannya, ini dapat dibuktikan bahawa motor pelangkah dapat dikawal kedudukan dan kelajuannya menggunakan mikropengawal PIC16F877.

ACKNOWLEDGEMENT

I would like to take this opportunity to express my deepest appreciation and thanks to those who helped me in accomplishing my final year project. Without their help, the completion of this project will not be possible.

First and foremost, I would like to thank to my supervisor Dr.Ir.Syafrudin Masri for his guidance, his assistance and most of all his patience with me during the project .

Besides, I also wish to express my appreciation to all the staff in PCB lab and Power lab for helping me in my PCB fabrication and stepper motor understanding.

For the industrial partner, I would like to thank you Gates Components (M) Sdn. Bhd.(The Malaysia distributor for Microchip Products), especially to Mr.A.K Khoo for helping me get some samples of dsPIC and PIC microcontroller.

Last but not least , I would like to express my deepest gratitude to my parents and all my friends especially Chan Chee Pin who had give me advice and providing moral supports to me for completing this project successfully.

TABLE OF CONTENTS **PAGES**

CHAPTER 3: PROJECT DESIGN

CHAPTER 4 : ANALYSIS RESULT AND SIMULATION

CHAPTER 5 : CONCLUSION

REFERENCE

LIST OF FIGURES

LIST OF TABLES

CHAPTER1 : INTRODUCTION

1.1 Background

The usage of the stepper motor in our daily life has been increasing and can be found easily in our daily life .The application of the stepper motor mainly use in the area like Computer Peripherals, Business Machines, Process Control and also Machine Tool. Some example of the usage in Computer peripherals is in printer where it is use for the paper feed and also for ribbon wind/rewind. In the Business machine such as copy machine the stepper motor is use in paper feed and also lens positioning. Others example of are main drive for conveyor belt in process control and X-Y-Z table positioning for milling machine in machine tools.(by Thomas .E Kissell)

The usage of the stepper motor in the area that had been mention above need to control the position movement of the stepper motor and also the speed. In such, this project which entitled as position movement and speed control by stepper motor with microcontroller PIC16F877 is carry out. After do some study on the controller, PIC microcontroller has been chosen to be implemented in my project. The PIC microcontroller is friendly user and easy to write the program to control the stepper motor by using PIC basic program. Through this project, the position movement and the speed of the stepper motor can be control by using the controller and the result can be display in the LCD.

1.2 Objective

1. To control the position movement of the stepper motor by using PIC microcontroller and the input control by keypad.
2. To determine the motor speed and display the function that is running by using LCD.

1.3 Methodology

This project mainly purpose is to prove the precise and accurate movement of the stepper motor and control the stepper motor speed by using PIC controller. Therefore, in the first step, stepper motor and also the PIC microcontroller had been study. Determination about how motor will work and also how the PIC microcontroller can be work to control the position and speed of the stepper motor had been carry out.

Secondly, after study about the motor and also the controller a circuit that mainly contain a digital controller, a translator circuit and a driver circuit had been developed. Digital controller is use to generate the train pulse while the translator circuit is used to convert the pulse of train to stepping sequences that can be used by motor driver.

The third step is to make sure that the circuit will work, so a simulation for the whole circuit will be carry out by using PSIM to prove the precise and accurate movement of the stepper motor. The stepper motor circuit has been substitute by equivalent resistance and inductance in the simulation circuit

After doing some analysis, if it show that a correct sequence had been send to the motor. For the fourth step, the PIC controller for the purpose of control the position movement of the motor had been study and implemented. The flow chart of the program has been created and the program is written and debug.

In the fifth step, the PCB layout will be done by using Orcad and PCB fabrication will be done in the following step. Some drilling and shouldering on the fabrication is carry out next to place the component on to the board. A special casing was prepared for the hardware implementation to show the position movement of the stepper motor. Add on to the project, a LCD display will be use to show the program that the motor is execute on that moment. The position of the stepper motor will be control by using keypad.

Finally, a complete set of hardware that contain of the PIC controller , translator circuit , driver circuit , stepper motor, keypad and also LCD in a casing will be ready to show the precise and accurate movement of stepper motor.

1.4 Reference Guide

This thesis mainly contain 5 chapter which include chapter 1 that is the introduction for this whole project. Chapter 2 contains the theory that is need to understand before can start the project . Some of the theory that show in this chapter are stepper motor theory, PIC microcontroller introduction and others. In Chapter 3, Project implementation will be explain in details include the circuit design and the PCB fabrication. Chapter 4 will than contain the analysis and experiment result that had been done and some discussion will be include also. Finally, in Chapter 5 it will contain the conclusion of the whole project with some use full suggestion and comment.

CHAPTER 2: Precise Movement And Speed Control of Stepper Motor

2.1. Types Of Stepper Motor

Stepper motors fall into two basic categories: Permanent magnet and variable reluctance. The type of motor determines the type of drivers, and the type of translator used. Of the permanent magnet stepper motors, there are several "subflavors" available. These include the Unipolar, Bipolar, and Multiphase varieties.

2.1.1. Permanent Magnet Stepper Motor

The **permanent-magnet stepper motor** operates on the reaction between a permanent-magnet rotor and an electromagnetic field. Figure 2.1 shows a basic two-pole PM stepper motor. The rotor shown in Figure 2.1(a) has a permanent magnet mounted at each end. The stator is illustrated in Figure 2.1(b). Both the stator and rotor are shown as having teeth. The teeth on the rotor surface and the stator pole faces are offset so that there will be only a limited number of rotor teeth aligning themselves with an energized stator pole. The number of teeth on the rotor and stator determine the step angle that will occur each time the polarity of the winding is reversed. The greater the number of teeth, the smaller the step angle.

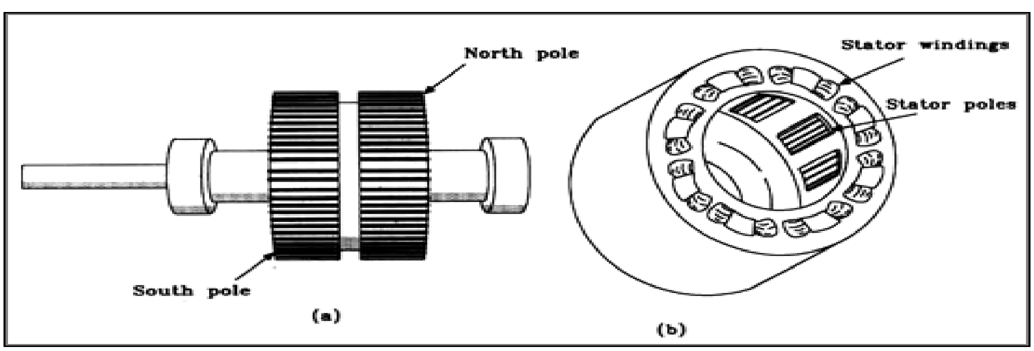


Figure 2.1 *Components of a PM stepper motor: (a) Rotor; (b) stator[3].*

When a PM stepper motor has a steady DC signal applied to one stator winding, the rotor will overcome the residual torque and line up with that stator field. The **holding torque** is defined as the amount of torque required to move the rotor one full step with the stator energized. An important characteristic of the PM stepper motor is that it can maintain the holding torque indefinitely when the rotor is stopped. When no power is applied to the windings, a small magnetic force is developed between the permanent magnet and the stator. This magnetic force is called a **residual,** or **detent torque.** The detent torque can be noticed by turning a stepper motor by hand and is generally about one-tenth of the holding torque.(by Colin D. Simpson Industrial Electronics, Prentice Hall PTR)

- **Unipolar Stepper Motors**

Unipolar motors are relatively easy to control . A simple 1-of-'n' counter circuit can generate the proper stepping sequence and drivers as simple as one translator per winding are possible with unipolar motors. Unipolar stepper motors are characterized by their center tapped windings. A common wiring scheme is to take all the taps of the center tapped windings and feed them with the voltage require to operate. A common wiring scheme is to take all the taps of the center-tapped windings and feed them +MV (Motor voltage). The driver circuit would then ground each winding to energize it. (Jones, D.W, 2004, Johnson , J, 2005).

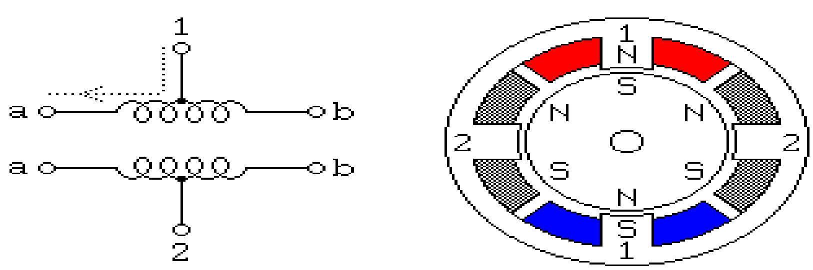


Figure 2.2 *A two phase Unipolar Stepper Motor Schematic[3]*

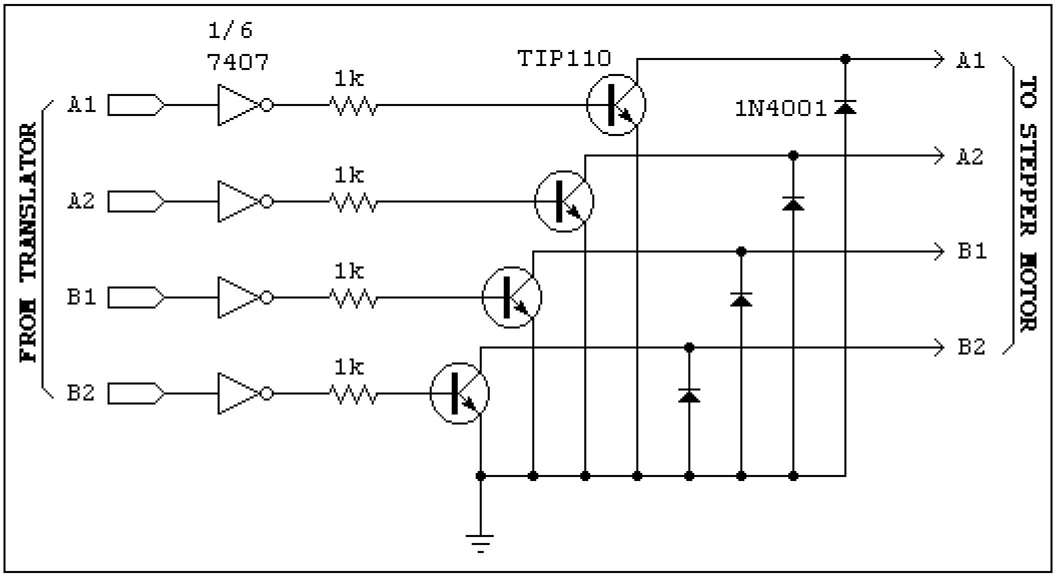
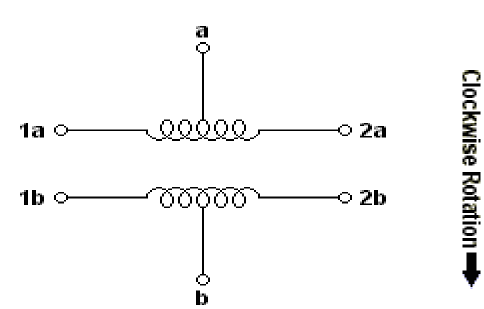


Figure 2.3 - *A typical unipolar stepper motor driver circuit. Note the 4 back EMF protection diodes [3].*

Unipolar stepper motors are recognized by their center-tapped windings. The number of phases is twice the number of coils, since each coil is divided in two. So the diagram below (Figure2.4), which has two center-tapped coils, represents the connection of a 4-phase unipolar stepper motor.



Index	1a	1b	2a	2b
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1
5	1	0	0	0
6	0	1	0	0
7	0	0	1	0
8	0	0	0	1

Figure 2.4 - *Unipolar stepper motor coil setup (left) and 1-phase drive pattern (right)[3].*

In addition to the standard drive sequence, high-torque and half-step drive sequences are also possible. In the high-torque sequence, two windings are active at a time for each motor step. This two-winding combination yields around 1.5 times more torque than the standard sequence, but it draws twice the current. Half-stepping is achieved by combining the two sequences. First, one of the windings is activated, then two, then one, etc. This effectively doubles the number of steps the motor will advance for each revolution of the shaft, and it cuts the number of degrees per step in half.

Clockwise Rotation

Index	1a	1b	2a	2b
1	1	0	0	1
2	1	1	0	0
3	0	1	1	0
4	0	0	1	1
5	1	0	0	1
6	1	1	0	0
7	0	1	1	0
8	0	0	1	1

Alternate Full Step Sequence
(Provides more torque)

Clockwise Rotation

Index	1a	1b	2a	2b
1	1	0	0	0
2	1	1	0	0
3	0	1	0	0
4	0	1	1	0
5	0	0	1	0
6	0	0	1	1
7	0	0	0	1
8	1	0	0	1
9	1	0	0	0
10	1	1	0	0
11	0	1	0	0
12	0	1	1	0
13	0	0	1	0
14	0	0	1	1
15	0	0	0	1
16	1	0	0	1

Half Step Sequence

Figure 2.5 - *Two-phase stepping sequence (left) and half-step sequence (right)[3].*

- **Bipolar Stepper Motors**

Bipolar motors are known for their excellent size/torque ratio and provide more torque for their size than the unipolar motors. Bipolar motors are designed with separate coils that need to be driven in either direction for proper stepping to occur. Bipolar stepper motors use the same binary drive pattern as a unipolar motor , only the 0 and 1 signals correspond to the polarity of the voltage applied to the coils and not simply on/off. (Jones, D.W, 2004, Johnson , J, 2005).

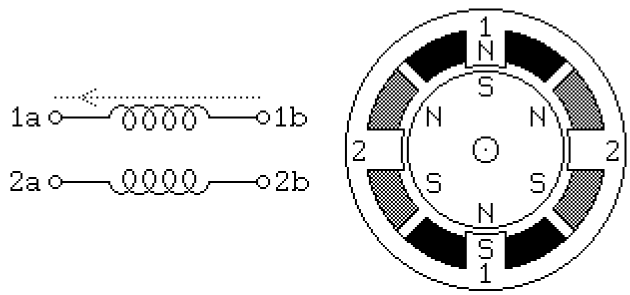
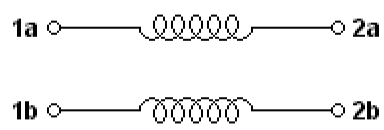


Figure 2.6 *A Bipolar 2 Phase Stepper Motor Schematic[3]*

Unlike unipolar stepper motors, Bipolar units require more complex driver circuitry. Bipolar motors are known for their excellent size/torque ratio, and provide more torque for their size than unipolar motors. Bipolar motors are designed with *separate* coils that need to be driven in either direction (the polarity needs to be reversed during operation) for proper stepping to occur. This presents a driver challenge. Bipolar stepper motors use the same binary drive pattern as a unipolar motor, only the '0' and '1' signals correspond to the polarity of the voltage applied to the coils, not simply 'on-off' signals. Figure 2.7 shows a basic 4-phase bipolar motor's coil setup and drive sequence.



Index	1a	1b	2a	2b
1	+	–	–	–
2	–	+	–	–
3	–	–	+	–
4	–	–	–	+
5	+	–	–	–
6	–	+	–	–
7	–	–	+	–
8	–	–	–	+

Figure 2.7 *- Bipolar stepper motor coil setup (left) and drive pattern (right)[3]*

A circuit known as an "H-bridge" (shown below) is used to drive Bipolar stepper motors. Each coil of the stepper motor needs its own H-bridge driver circuit. Typical bipolar steppers have 4 leads, connected to two isolated coils in the motor. ICs specifically designed to drive bipolar steppers (or DC motors) are available (Popular are the L297/298 series from ST Microelectronics, and the LMD18T245 from National Semiconductor). Usually these IC modules only contain a single H-bridge circuit inside of them, so two of them are required for driving a single bipolar motor. One problem with the basic (transistor) H-bridge circuit is that with a certain combination of input values (both '1's) the result is that the power supply feeding the motor becomes shorted by the transistors. This could cause a situation where the transistors and/or power supply may be destroyed. A small XOR logic circuit was added to keep both inputs from being seen as '1's by the transistors.

Another characteristic of H-bridge circuits is that they have electrical "brakes" that can be applied to slow or even stop the motor from spinning freely when not moving under control by the driver circuit. This is accomplished by essentially shorting the coil(s) of the motor together, causing any voltage produced in the coils by during rotation to "fold back" on itself and make the shaft difficult to turn. The faster the shaft is made to turn, the more the electrical "brakes" tighten.

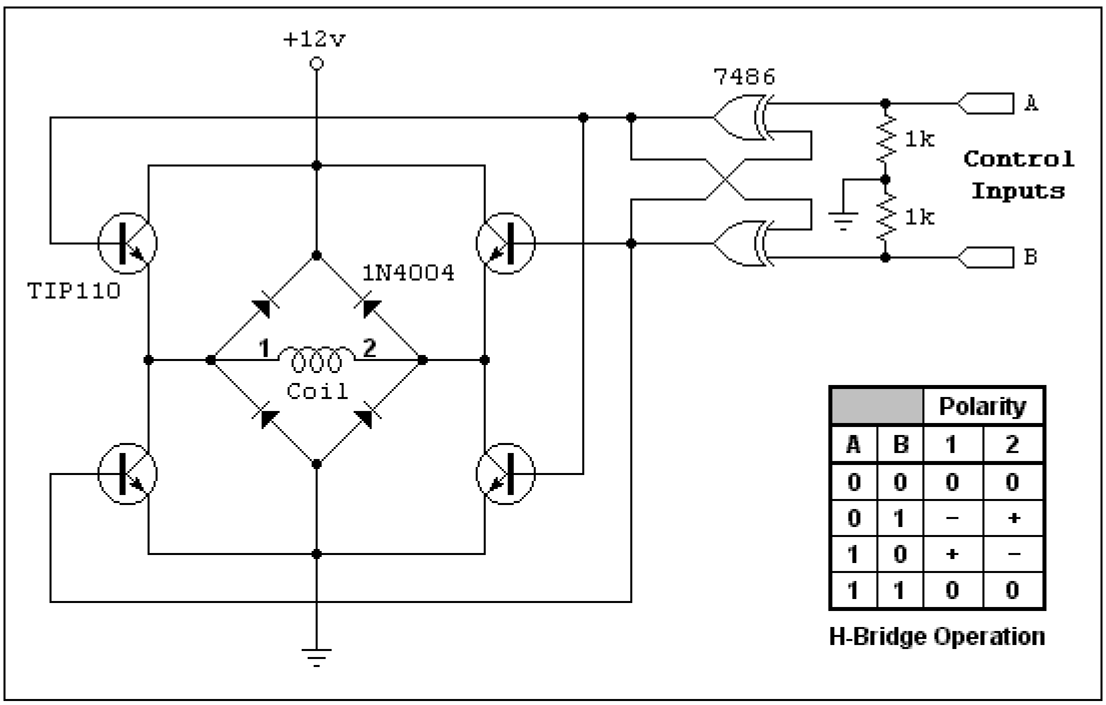
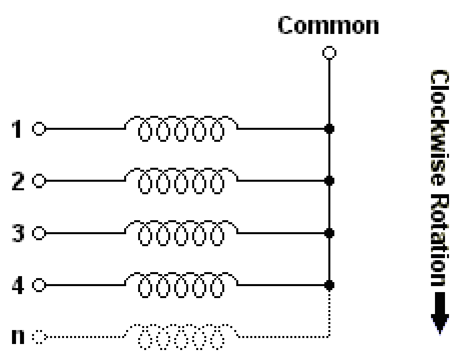


Figure 2.8 - *A typical H-Bridge circuit. The 4 diodes clamp inductive kickback [3].*

2.1.2 Variable Reluctance Stepper Motor

The **variable-reluctance** (VR) **stepper motor** differs from the PM stepper in that it has no permanent-magnet rotor and no residual torque to hold the rotor at one position when turned off. When the stator coils are energized, the rotor teeth will align with the energized stator poles. This type of motor operates on the principle of minimizing the reluctance along the path of the applied magnetic field. By alternating the windings that are energized in the stator, the stator field changes, and the rotor is moved to a new position. (by Colin D. Simpson Industrial Electronics, Prentice Hall PTR) .Figure 2.9 show the variable reluctance stepper motor coil set up and the drive pattern



Index	1	2	3	4
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1
5	1	0	0	0
6	0	1	0	0
7	0	0	1	0
8	0	0	0	1

Figure 2.9 - *Variable reluctance stepper motor coil setup (left) and drive pattern (right)[3]*.

2.1.3 Hybrid Stepper Motor

The **hybrid** step motor consists of two pieces of soft iron, as well as an axially magnetized, round permanent-magnet rotor. The term *hybrid* is derived from the fact that the motor is operated under the combined principles of the permanent magnet and variable-reluctance stepper motors. The stator core structure of a hybrid motor is essentially the same as its VR counterpart. The main difference is that in the VR motor, only one of the two coils of one phase is wound on one pole, while a typical hybrid motor will have coils of two different phases wound on one the same pole. The two coils at a pole are wound in a configuration known as a **bifilar** connection. Each pole of a hybrid motor is covered with uniformly spaced teeth made of soft steel. The teeth on the two sections of each pole are misaligned with each other by a half-tooth pitch. Torque is created in the hybrid motor by the interaction of the magnetic field of the permanent magnet and the magnetic field produced by the stator. (by Colin D. Simpson Industrial Electronics, Prentice Hall PTR)

2.2 Stepper Motor Translator Circuit

Figure 2.10 illustrates the simplest solution to generating a one-phase drive sequence. For unipolar stepper motors, the circuit in Figure 2.3, or for bipolar stepper motors, the circuit in Figure 2.7 can be connected to the 4 outputs of this circuit to provide a complete translator + driver solution. This circuit is limited in that it cannot reverse the direction of the motor. This circuit would be most useful in applications where the motor does not need to change directions.

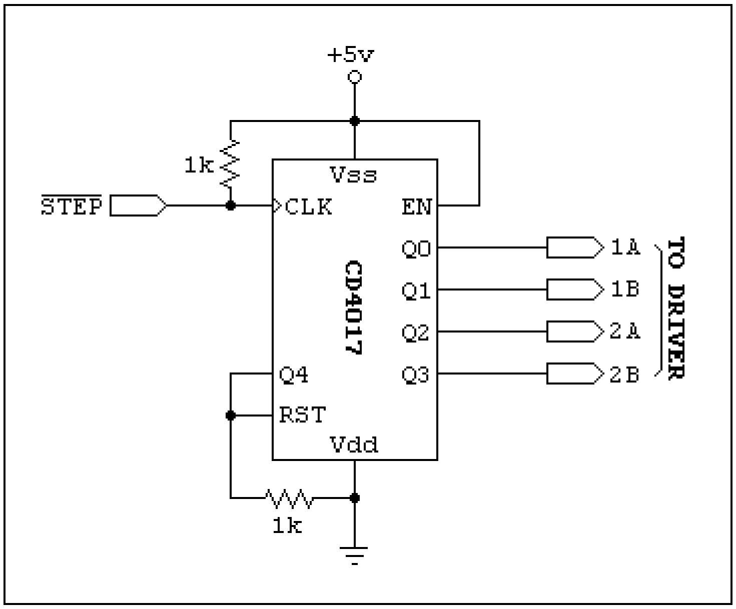


Figure 2.10 *- A simple, single direction, single phase drive translator[3].*

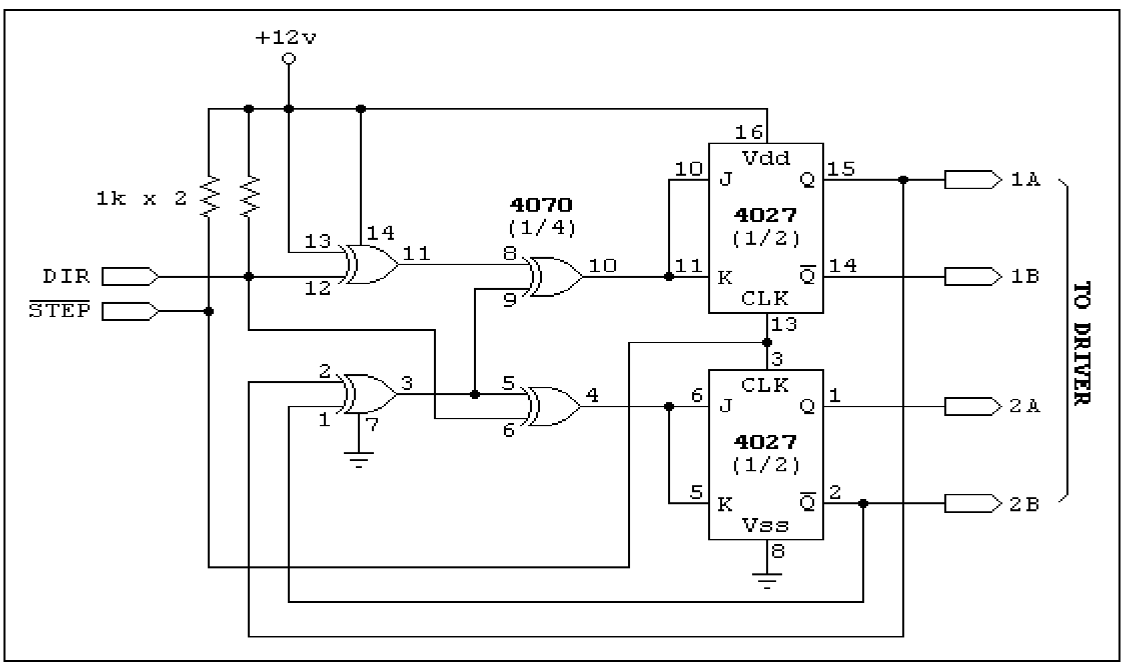


Figure 2.11 *- A simple, bidirectional, two-phase drive stepper motor translator circuit [3].*

There are several standard stepper motor translation circuits which use discrete logic ICs. Below you will find yet another one of these. The circuit in Figure 2.9 has not been tested but theoretically should work without problems.

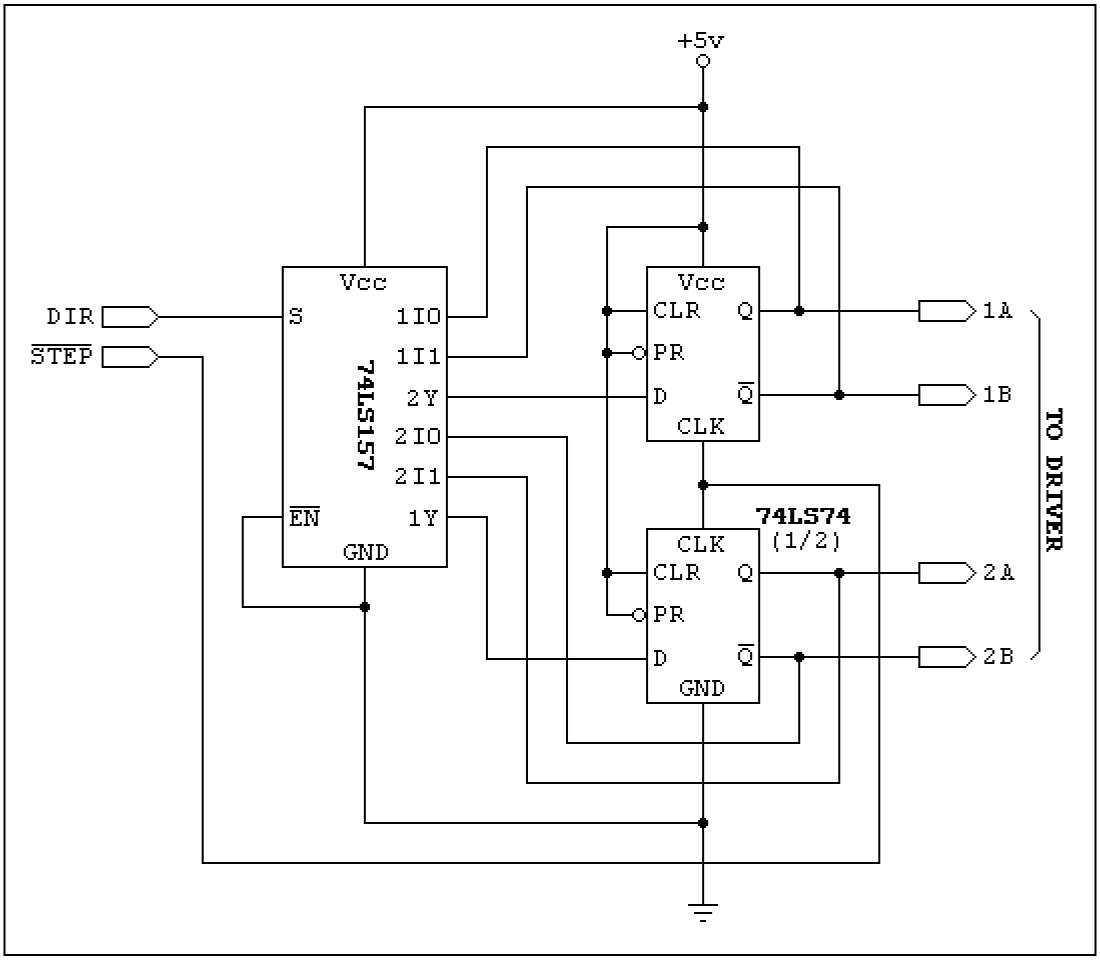


Figure 2.12 *- Another example of a two-phase drive translator circuit, this time using a multiplexer [3].*

2.3 Common Characteristic of Stepper Motor

Stepper motors are not just rated by voltage. The following elements characterize a given stepper motor and the example characteristic of the stepper motor that can be use in table2.1.

Voltage

Stepper motors usually have a voltage rating. This is either printed directly on the unit, or is specified in the motor's datasheet. Exceeding the rated voltage is sometimes necessary to obtain the desired torque from a given motor, but doing so may produce excessive heat and shorten the life of the motor.

Resistance

Resistance-per-winding is another characteristic of a stepper motor. This resistance will determine current draw of the motor, as well as affect the motor's torque curve and maximum operating speed.

Degrees per step

This is often the most important factor in choosing a stepper motor for a given application. This factor specifies the number of degrees the shaft will rotate for each full step. Half step operation of the motor will double the number of steps/revolution, and cut the degrees-per-step in half. For unmarked motors, it is often possible to carefully count, by hand, the number of steps per revolution of the motor. The degrees per step can be calculated by dividing 360 by the number of steps in 1 complete revolution Common degree numbers include: 0.72, 1.8, 3.6, 7.5, 15, and even 90. Degrees per step is often referred to as the *resolution* of the motor.

Characteristic		RS 440-307
Number of phases		4
Step angle	degree	7.5
Resistance per phase	ohm	26.7
Inductance per phase	mH	40
Current per phase	A	0.48
Absorbed Power	W	12.5
Holding Torque	mH.m	240
Voltage at motor Terminals	V	12.7
Inertia of rotor	gcm2	180
Max.detent torque	Degree celcius	16
Max.motor temperature	Degree celcius	120
Standard length of leads	mm	250
weight	g	250

Table 2.1 *Characteristic of Stepper motor*

2.4 Stepper Motor And its Advantage

There are numerous types of motion control systems, including; Stepper Motor, Linear Step Motor, DC Brush, Brushless, Servo, Brushless Servo and more. This document will concentrate on Stepper Motor technology.

In Theory, a Stepper motor is a marvel in simplicity. It has no brushes, or contacts. Basically it's a synchronous motor with the magnetic field electronically switched to rotate the armature magnet around.

A Stepping Motor System consists of three basic elements, often combined with some type of user interface (Host Computer, PLC or Dumb Terminal) as shown in figure 2.13:

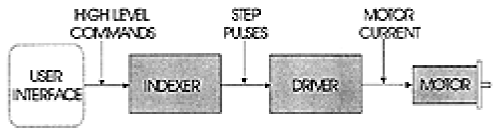


Figure2.13 *Stepping Motor System[4]*

The Indexer (or Controller) is a microprocessor capable of generating step pulses and direction signals for the driver. In addition, the indexer is typically required to perform many other sophisticated command functions.

The Driver (or Amplifier) converts the indexer command signals into the power necessary to energize the motor windings. There are numerous types of drivers, with different current/amperage ratings and construction technology. Not all drivers are suitable to run all motors, so when designing a Motion Control System the driver selection process is critical.

The Stepper Motor is an electromagnetic device that converts digital pulses into mechanical shaft rotation. Advantages of step motors are low cost, high reliability, high torque at low speeds and a simple, rugged construction that operates in almost any environment. The main disadvantages in using a stepper motor is the resonance effect often exhibited at low speeds and decreasing torque with increasing speed.

2.5 PIC16F877 Microcontroller Introduction

PIC is a stand alone chip and can work without computer after the program is burn into the PIC by using the EPIC Programmer. The PIC will receives the input signals from a keypad which will than determine the position of the pointer. When there is an input from the keypad, it will compare with the previous pointer position and determine the direction and amount of the pulses is generated by the PIC.

The PIC that used in this project is PIC 16F877A which contain of 40 pins. There are 33 I/O pins which mean it work in bidirectional way. The port A has 6 pins , port B has 8 pins, port C has 8pins, port D has 8 pins and the port E has 3pins as shown in the figure2.14. The operating speed for the clock input or the crystal frequency that used for this PIC is 20 MHz. This PIC has 8K x 14 words of FLASH Program Memory, 368 x 8 bytes of Data Memory(RAM) and 256 x 8 bytes of EEPROM Data Memory.

The PIC 16F8X fits perfectly in applications ranging from high speed automotive and appliance motor control to low power remote sensors, electronic locks, security devices and smart card. The Flash / EEPROM technology makes customization of application program fast and convenient . The small footprint packages make this microcontroller series perfect for all applications with space limitation. Low- cost , low power , high performance, ease of use and the I/O flexibility make the PIC 16F8X very versatile even in areas where no microcontroller use has been considered before such as serial communication , compare and PWM functions. Figure 2.15 show the block diagram for PIC16F877 and table 2.2 show the PIC16F877 pin out description. (microchip PIC16F877)

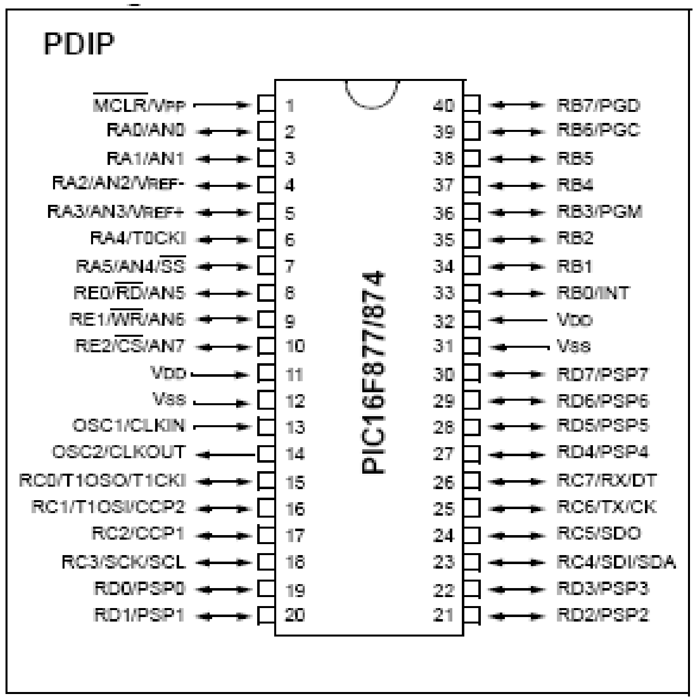
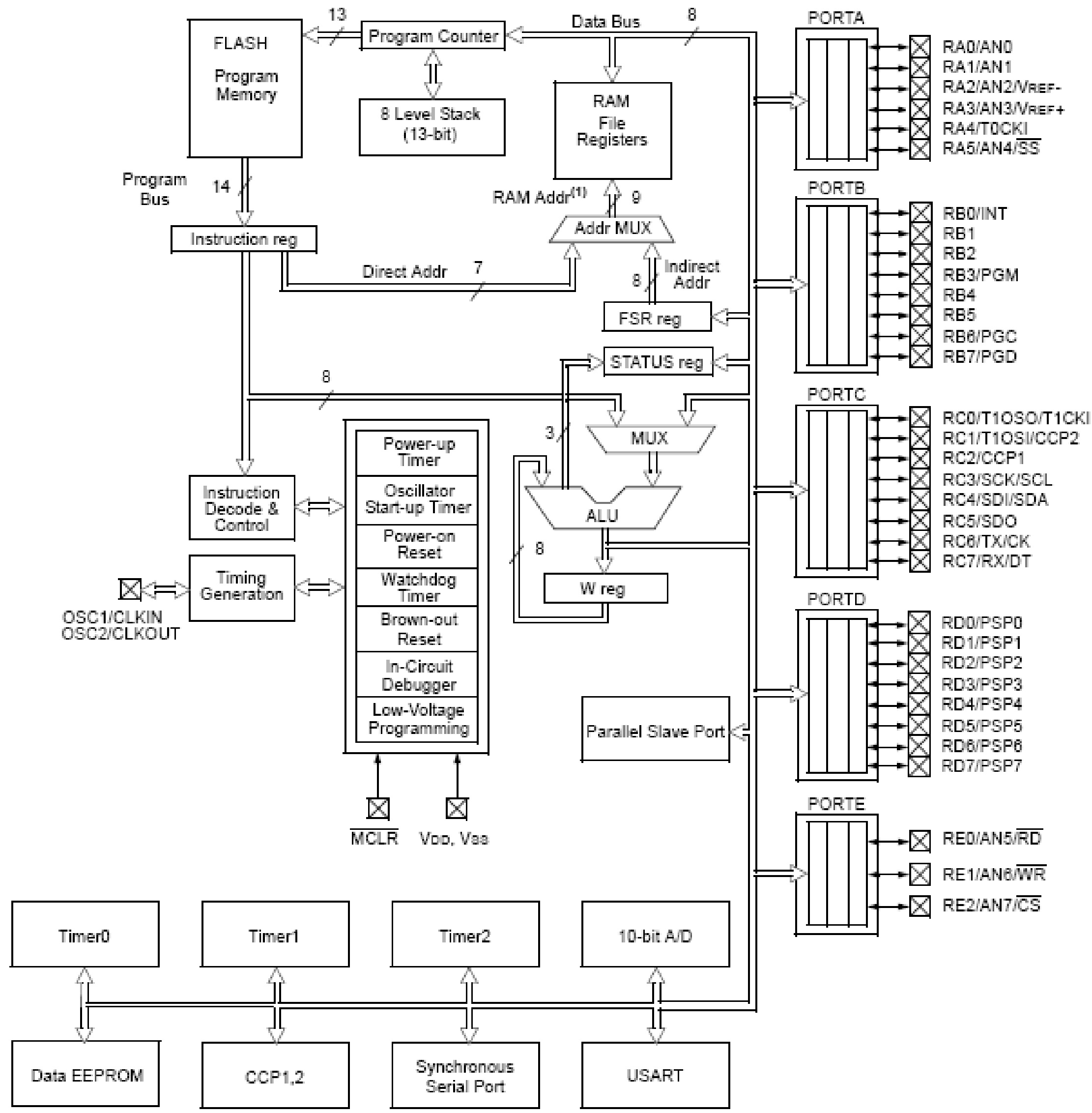


Figure 2.14 *PIC 16F877 Pin Diagram[6]*

Device	Program FLASH	Data Memory	Data EEPROM
PIC16F874	4K	192 Bytes	128 Bytes
PIC16F877	8K	368 Bytes	256 Bytes



Note 1: Higher order bits are from the STATUS register.

Figure 2.15 *Block diagram for PIC16F877*

Table2.2 *Pin Out Description for PIC16F877*

Pin Name	DIP Pin#	PLCC Pin#	QFP Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	13	14	30	I	ST/CMOS(4)	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	14	15	31	O	—	Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate.
$\overline{MCLR}$/VPP	1	2	18	I/P	ST	Master Clear (Reset) input or programming voltage input. This pin is an active low RESET to the device.
						PORTA is a bi-directional I/O port.
RA0/AN0	2	3	19	I/O	TTL	RA0 can also be analog input0.
RA1/AN1	3	4	20	I/O	TTL	RA1 can also be analog input1.
RA2/AN2/VREF-	4	5	21	I/O	TTL	RA2 can also be analog input2 or negative analog reference voltage.
RA3/AN3/VREF+	5	6	22	I/O	TTL	RA3 can also be analog input3 or positive analog reference voltage.
RA4/T0CKI	6	7	23	I/O	ST	RA4 can also be the clock input to the Timer0 timer/ counter. Output is open drain type.
RA5/$\overline{SS}$/AN4	7	8	24	I/O	TTL	RA5 can also be analog input4 or the slave select for the synchronous serial port.
						PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs.
RB0/INT	33	36	8	I/O	TTL/ST(1)	RB0 can also be the external interrupt pin.
RB1	34	37	9	I/O	TTL	
RB2	35	38	10	I/O	TTL	
RB3/PGM	36	39	11	I/O	TTL	RB3 can also be the low voltage programming input.
RB4	37	41	14	I/O	TTL	Interrupt-on-change pin.
RB5	38	42	15	I/O	TTL	Interrupt-on-change pin.
RB6/PGC	39	43	16	I/O	TTL/ST(2)	Interrupt-on-change pin or In-Circuit Debugger pin. Serial programming clock.
RB7/PGD	40	44	17	I/O	TTL/ST(2)	Interrupt-on-change pin or In-Circuit Debugger pin. Serial programming data.

Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as an external interrupt.
2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
3: This buffer is a Schmitt Trigger input when configured as general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).
4: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.

Pin Name	DIP Pin#	PLCC Pin#	QFP Pin#	I/O/P Type	Buffer Type	Description
						PORTC is a bi-directional I/O port.
RC0/T1OSO/T1CKI	15	16	32	I/O	ST	RC0 can also be the Timer1 oscillator output or a Timer1 clock input.
RC1/T1OSI/CCP2	16	18	35	I/O	ST	RC1 can also be the Timer1 oscillator input or Capture2 input/Compare2 output/PWM2 output.
RC2/CCP1	17	19	36	I/O	ST	RC2 can also be the Capture1 input/Compare1 output/PWM1 output.
RC3/SCK/SCL	18	20	37	I/O	ST	RC3 can also be the synchronous serial clock input/output for both SPI and I^2C modes.
RC4/SDI/SDA	23	25	42	I/O	ST	RC4 can also be the SPI Data In (SPI mode) or data I/O (I^2C mode).
RC5/SDO	24	26	43	I/O	ST	RC5 can also be the SPI Data Out (SPI mode).
RC6/TX/CK	25	27	44	I/O	ST	RC6 can also be the USART Asynchronous Transmit or Synchronous Clock.
RC7/RX/DT	26	29	1	I/O	ST	RC7 can also be the USART Asynchronous Receive or Synchronous Data.
						PORTD is a bi-directional I/O port or parallel slave port when interfacing to a microprocessor bus.
RD0/PSP0	19	21	38	I/O	ST/TTL(3)	
RD1/PSP1	20	22	39	I/O	ST/TTL(3)	
RD2/PSP2	21	23	40	I/O	ST/TTL(3)	
RD3/PSP3	22	24	41	I/O	ST/TTL(3)	
RD4/PSP4	27	30	2	I/O	ST/TTL(3)	
RD5/PSP5	28	31	3	I/O	ST/TTL(3)	
RD6/PSP6	29	32	4	I/O	ST/TTL(3)	
RD7/PSP7	30	33	5	I/O	ST/TTL(3)	
						PORTE is a bi-directional I/O port.
RE0/$\overline{RD}$/AN5	8	9	25	I/O	ST/TTL(3)	RE0 can also be read control for the parallel slave port, or analog input5.
RE1/$\overline{WR}$/AN6	9	10	26	I/O	ST/TTL(3)	RE1 can also be write control for the parallel slave port, or analog input6.
RE2/$\overline{CS}$/AN7	10	11	27	I/O	ST/TTL(3)	RE2 can also be select control for the parallel slave port, or analog input7.
Vss	12,31	13,34	6,29	P	—	Ground reference for logic and I/O pins.
VDD	11,32	12,35	7,28	P	—	Positive supply for logic and I/O pins.
NC	—	1,17,28, 40	12,13, 33,34		—	These pins are not internally connected. These pins should be left unconnected.

Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as an external interrupt.
2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
3: This buffer is a Schmitt Trigger input when configured as general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).
4: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.

CHAPTER 3: PROJECT DESIGN

3.1 Circuit Implementation

The design of the project can be devided into three main parts as shown in the Figure 3.1. The three main part consist of translator motor driver and the motor. In order to achieve the objective of the project, the circuit as figure has been implement and control by a PIC in a positioning process and show precise movement of stepper motor.

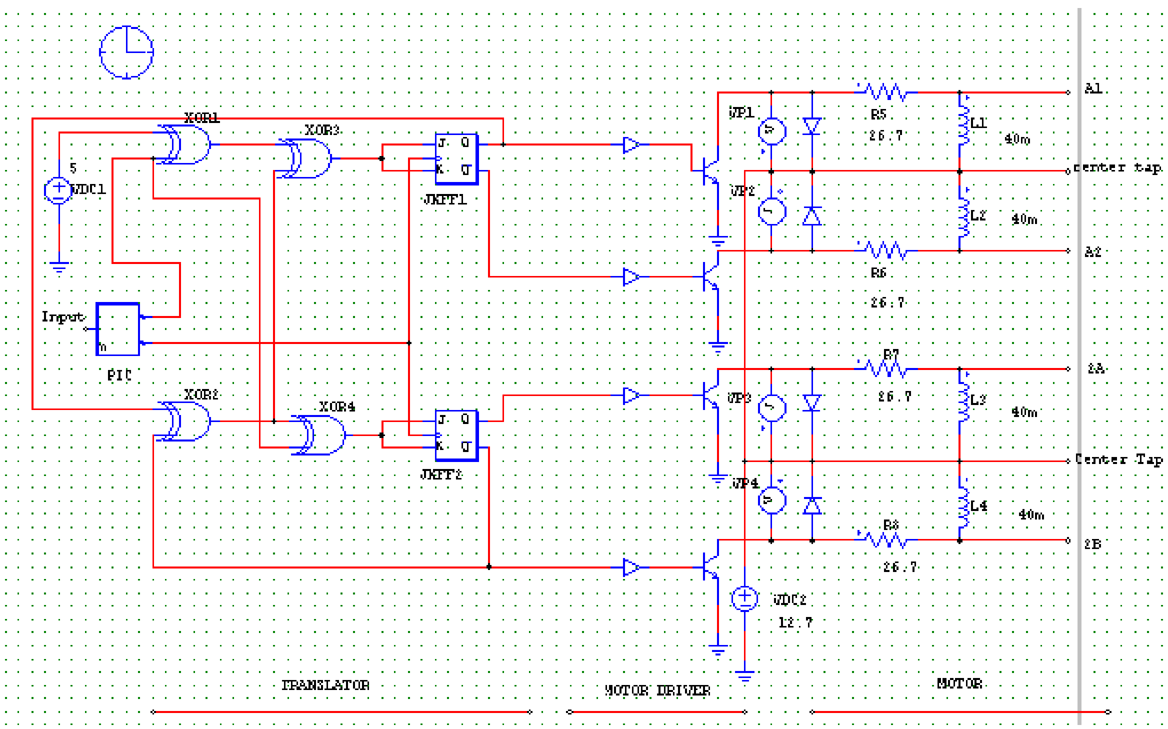


Figure 3.1 *Full Circuit Design*

PIC in this project is used to feed two pulse trains to the translator circuit . One of the pulse trains is used to control the number of steps that are needed by the motor and the other is use to determine the direction of the stepper motor. The trains of pulse that are sent to the translator circuit will then determine the step sequence to trigger the translators which will then operate the stepper motor. The motor is connect to the driver circuit in the sequence of 1A ,center tap,1B,2A, center tap and 2B.

3.2 Circuit Introduction

In this project , it is consist of three main part as shown in Figure 3.2.The three main parts are :

- PIC Microcontroller
- Translator circuit
- Motor Driver

PIC controller is used to feed two pulse trains to the translator circuit . One of the pulse is used to control the number of steps that are needed by the motor and the other is used to determine the direction of the stepper motor as shown as step and direction in the figure 3.2.

The pulse that are send to the translator circuit will determine the step sequence to trigger the transistor and operate the stepper motor while the speed will be control by the programme in the PIC controller.

The motor driver is setup to handle the current drawn by the motor windings.

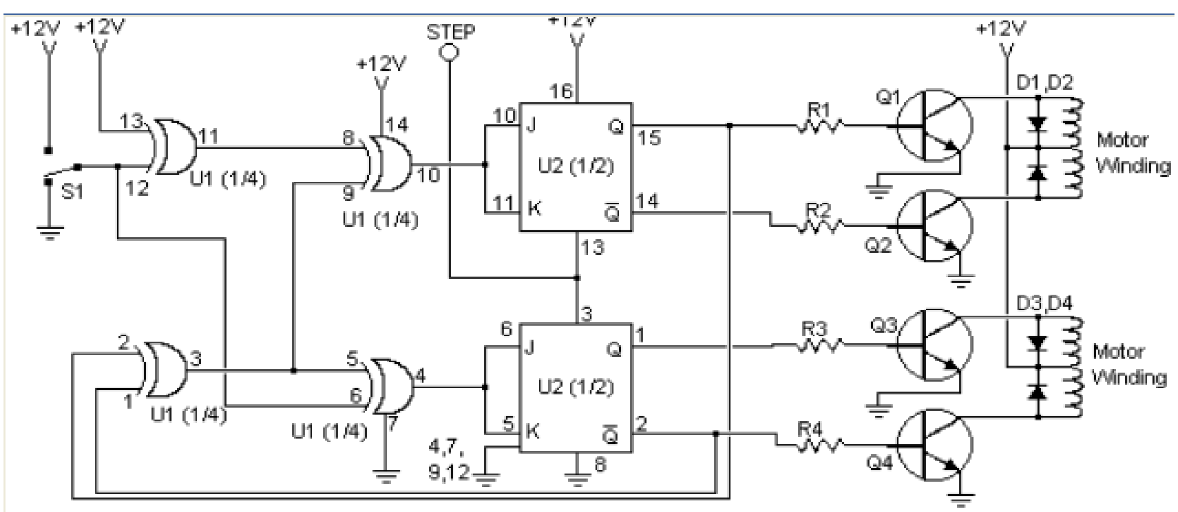


Figure 3.2 *Stepper motor with Digital Controller[3].*