I n s p i r i n g E x c e l l e n c e

Ishtiaq Hafiz Akash

Supervisor: **Dr. Amina Hasan Abedin**

A thesis submitted in partial fulfillment for the degree of Bachelor of Science in the Department of Electrical and Electronics Engineering, BRAC University

August 2018

This thesis report is submitted in partial fulfillment of the requirements for the degree of Bachelor of Science in Electrical and Electronic Engineering, 2018.

DECLARATION

This is to declare that the thesis titled "Implementation Of Microstepping in Stepper Motor Using ATmega32 Microcontroller" is submitted to the Department of Electrical and Electronics Engineering of BRAC University in partial fulfillment of the Bachelor of the Bachelor of Science degree. This is my original report.

Signature of Supervisor

Dr. Amina Hasan Abedin

Assistant Professor

EEE Department

Brac University

Ishtiaq Hafiz Akash (EEE)

ID: 14121007

ACKNOWLEDGEMENTS

This paper is the production of Ishtiaq Hafiz Akash Of Electrical and Electronic Engineering(EEE) Department Of Brac University that was forged with diligence and the years of sheer knowledge that was acquired specially during the last three semesters of thesis and the last few years of my undergraduate program, which addresses the motor system, power electronics components and its various properties.

Firstly, I would like to thank the most Merciful and The Most Omnipotent for giving me the power of endurance and strength during the rise and fall, ups and downs of our study and research. I am sincerely thankful to our honorable thesis supervisor, Dr. Amina Hasan Abedin for trusting in me and for her abysmal support. Gratefulness is bounded, not only to the thesis supervisor of Electrical and Electronics engineering (EEE) department but also to each and every faculty, who blessed me in the form of knowledge and encouragement. Last but not the least, I would like to give special thanks to my beloved parents, brothers and sisters for their love and support, and my friends for their constant support and encouragement.

ABSTRACT

Unlike other motors, stepper motor cannot give a continuous rotation. Hence, stepper motor is different from other motors. Stepper motor is a brushless DC motor that divides its full rotation into a number of equal steps, where position can be commanded to move and hold at each of these steps without any sensor for feedback. Stepper motor is widely used in electrical and mechanical applications where at low speed, positioning accuracy, and high speed dynamic are determining factors. Full drive and half drive modes are general modes of stepper motor. Here, stepper motor undergoes microstepping using ATmega32 microcontroller.

Table of Contents

Chapter 1

1.1 Introduction 1

1.2 Scope 2

1.3 Thesis Organizations 2

Chapter 2

DC Motors Theory 3

2.1 DC Motor 3

2.2 Working Principle of DC Motor 3

2.3 Brushed DC Motor 5

2.4 Brushless DC Motor 6

2.5 Stepper Motor 7

2.5.1 Variable Reluctance Motor (VRM) 8

2.5.1 Permanent Magnet Stepper Motor (PMSM) 9

2.5.3 Hybrid Stepper Motor (HSM) 10

2.6 Mode of Stepper Motor 10

2.6.1 Single Coil Excitation 11

2.9.1 Full Step Sequence 11

2.9.2 Half Step Sequence 11

2.9.3 Micro Step Sequence 12

Chapter 3

Proposed Circuit and Simulation

3.1 Proposed System Diagram 13

3.3.1Atmega32 13

3.1.2 R-2R DAC 14

3.1.2.1 Resolution 16

3.1.2.2 Speed 16

3.1.2.3 Linearity……………………………………………………………………………………………16
3.1.2.4 Setting Time…………………………………………………………………………………………17
3.1.2.5 Reference Voltage…………………………………………………………………………………17
3.1.3 DAC Interfacing with Microcontroller... ...17
3.1.4 Relay Driver 19
3.2 Rotational Angle and Voltage Assignment Relationship 20
3.3 Circuit Synopsis 22

Chapter 4
Discussion 26
4.1 Limitations of Simulation 26
4.2 Future Scope 26
4.3 Conclusion 27

References 28
Appendix I 30

List of Figures

Figure 2.1: Fleming's Left Hand Rule....4
Figure 2.2: Principles of DC Motor 5
Figure 2.3: 3 Coil, 2 Pole Brushed Permanent DC Motor....6
Figure 2.4: BLDC Motor with one coil energized 7
Figure 2.5: BLDC Motor with two coils energized....7
Figure 2.6: Permanent Magnet Stepper Motor 9
Figure 2.7: Variable Reluctance Stepper Motor....9
Figure 2.8: Hybrid Synchronous Stepper Motor 10
Figure 2.9: Full Step Drive 11
Figure 2.10: Half Step Drive 12
Figure 3.1: Circuit Layout 13
Figure 3.2: Atmega32 Microcontroller.... 14
Figure 3.3: N bit R/2R Ladder DAC 14
Figure 3.4: Linearity Specifications 16
Figure 3.5: Setting Time Specifications of DAC 17
Figure 3.6: DAC interfacing with Microcontroller 18
Figure 3.7: Internal Circuit diagram of ULN 2003A(Darlington Pair) 19
Figure 3.8: ULN 2003A.... 20
Figure 3.9: Angle and Voltage Relationship Plot.... 20
Figure 3.10: Stepper motor and their respective coil arrangements in proteus....21
Figure 3.11: Design of the proposed circuit in proteus 23

List of Table

Table 3.1 25

Chapter 1

Introduction

Stepper motor is a brushless DC motor that divides a full rotation into a number of determined steps. In other words, stepper motor moves in particular increments rather than in continuous fashion, where stator coils are energized one step at a time, making the rotor bound to move in steps. The rotor has no rotating coils. Increment size can vary, depending upon the desired application of the stepper motor

A stepper motor internal structure permits a very simple control system; step motor has several special features that make it an ideal candidate for using as a positioning control device.

i. It is possible to achieve very low speed synchronous rotation with a load that is directly coupled to the shaft

ii. The motor has full torque at standstill (if the windings are energized) iii. Precise positioning and repeatability of movement since good stepper motors have an accuracy of 3 – 5% of a step and this error is non-cumulative from one step to the next.

iv. Stepper motor has a lower output and efficiency as compared to other motors.

v. Speed is proportional to the frequency of the input pulses. Hence, a wide range of rotational speeds can be put forward to the stepper motor.

One of the disadvantages of stepper motor is that, resonance occur as speed of stepper motor is varied. Step accuracy and resolution are directly dependent to the device structure and can be a limiting factor for various applications where high resolution is required.

Micro-stepping [1-3] is a method for driving stepper motor where input pulses are converted to discrete estimations of the sinusoidal functions. Microstepping increases the position resolution and smoothness, thus minimizes the disadvantages of stepper motor as discussed above.

1.1 Applications

Computer controlled stepper motors are a type of motion-control positioning system. They are typically digitally controlled as part of an open loop system for use in holding or positioning applications.

In the field of lasers and optics they are frequently used in precision positioning equipment such as linear actuators, linear stages, rotation stages, goniometers, and mirror mounts. Other uses are in packaging machinery, and positioning of valve pilot stages for fluid control systems.

Commercially, stepper motors are used in floppy disk drives, flatbed scanners, computer printers, plotters, slot machines, image scanners, compact disc drives, intelligent lighting, camera lenses, CNC machines and, more recently, in 3D printers.

1.2 Scope

This thesis presents a stepper motor position control system design with the hope of acquiring micro-stepping abilities. Stepper motors are analyzed and different driving methods including several possible control approaches are discussed in the theory section. A proteus simulation circuit is designed in order to control the position of the rotor in microsteps using micro controller, Atmega32.

1.3 Thesis Organizations

This thesis contains four chapters. Chapter 2 talks about DC motor and the modes of stepper motor. Chapter 3 describes the proposed driver circuit of the microstepping. Chapter 4 puts forward conclusive limitations and conclusion.

Chapter 2

DC Motor Theory

2.1 DC Motor

A DC motor is a machine that converts direct electrical energy into mechanical energy. The most general types depend on the forces produced by magnetic fields. The magnetic fields are generated either by supplying direct current to electromagnetic windings or with permanent magnets. The rotor rotates on its axis as a result of electromagnetic interactions with the stator. During motor operation, the magnetic fields repel and attract each other, and the rotor continues to rotate. Normally, all types of DC motors have some internal mechanism, either electromechanical or electronic; to periodically change the direction of current flow in part of the motor. DC motors are categorized into three types: brushed DC motors, brushless DC motors, and stepper motors.

2.2 Working Principle of DC Motor

The most basic construction of a DC motor lies in a current carrying armature which is connected to the supply end through commutator parts and brushes. The armature is placed in between north and south poles of a permanent or an electromagnet. When direct current is supplied in the armature, a mechanical force acts on it due to the electromagnetic effect of the magnet. Applying Fleming's left hand rule determines the direction of the force acting on the armature conductors of DC motor as shown in the figure 2.1. The physics behind the working of a DC motor is that "*whenever a current carrying conductor is placed in a magnetic field, it experiences a mechanical force".* The magnitude of force is given by $F = BIL$. Where, B = magnetic flux density, I = current and L = length of the conductor within the magnetic field.

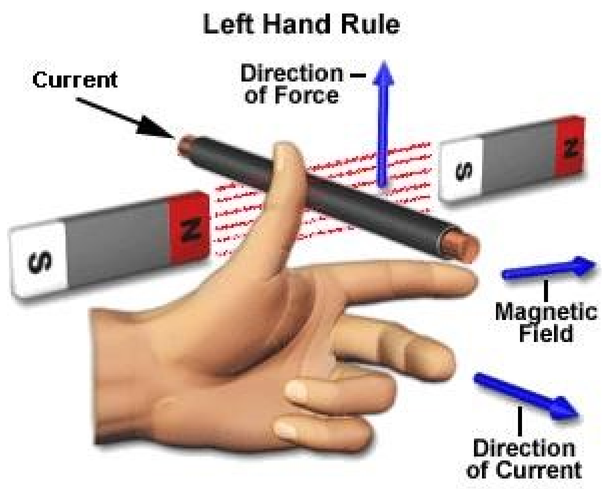

Figure 2.1: Fleming's Left Hand Rule

The stator provides a constant magnetic field and the armature, which is the rotationary part, is simply a coil. The armature is connected to a DC power source through a pair of commutator rings. When the current flows through the coil, an electromagnetic force is induced on it according to Lorentz law. So the coil would start to rotate. As it rotates the commutator rings connected with the power source of the opposite polarity will start to rotate. It results that on the left side of the coil as shown in figure 2.2, the electricity will always flow 'away' and on the right side of the coil, the electricity would always flow towards. This ensures that the torque is always in the same direction throughout the motion. So the coil will continue to rotate. But when the coil is nearly perpendicular to the magnetic flux, the torque action nears zero since Fleming's left hand rule is not being followed there. As a result, there will be irregular motion of the DC motor when during its operation. A solution to it is to add one more loop with a separate commutator pair for it. Moreover, the more such loops and the more the commutators, the smoother will be the motor rotation. In a practical motor, the armature loops are fitted inside slots with highly permeable steel layers. This will ensure stronger magnetic flux interaction. Spring loaded commutator brushes help to maintain contact with the power source. A permanent magnet stator motor is used only for very small DC motors. Most often, an electromagnet is used.

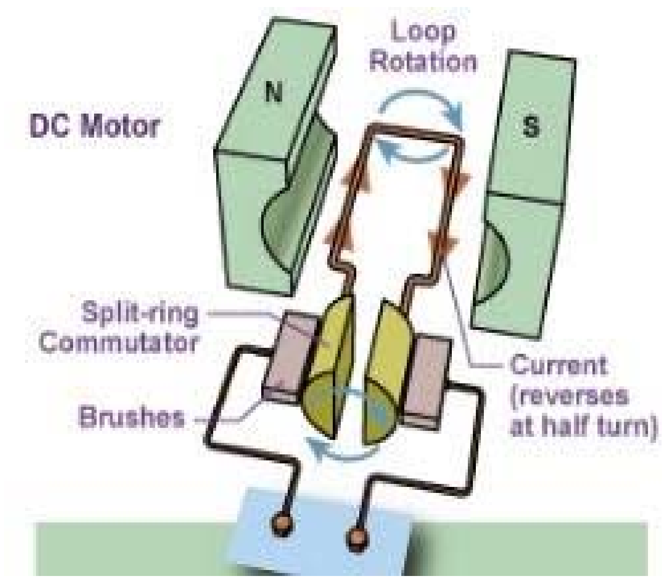

Figure 2.2: Principles of DC Motor

2.3 Brushed DC Motor

Stator generates a stationary magnetic field that surrounds the rotor. This magnetic field is generated or electromagnetic windings. Rotor(armature) contains one or more windings. When these windings are energized, they produce a magnetic field. The magnetic poles of this rotor field will be attracted by the opposite poles generated by the stator, causing the motor to turn. As the motor turns, the windings are consistently being energized in a different sequence, so the magnetic poles generated by the rotor do not overrun the poles generated in the stator. This switching in the field in called commutation. Brushed DC motor do not need a controller to switch current in the motor windings, instead it uses a mechanical commutation of the windings, a copper sleeve known as the commutator, [4-6] which resides on the axle of the rotor. As the motor turns, carbon brushes slide over the commutator coming in contact with different segments of the commutator. The segments are attached to different rotor windings; therefore, a dynamic magnetic field is generated inside the motor when a voltage is applied across the brushes of the motor.

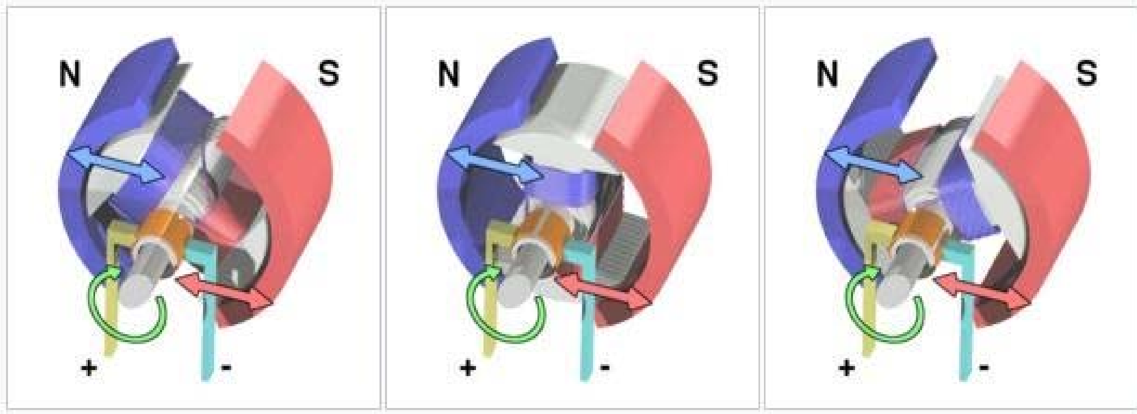

Figure 2.3: 3 Coil, 2 Pole Brushed Permanent Magnet DC motor

In the above diagram, when the coil is powered, a magnetic field is generated around the armature. The left side of the armature is pushed away from the left magnet and drawn toward the right, causing rotation. The armature continues to rotate. When the armature becomes horizontally aligned, the torque becomes zero. At this point, the commutator reverses the direction of current through the coil, reversing the magnetic field. The process then repeats.

2.4 Brushless DC Motor

In order to make the operation more reliable, more efficient and less noisy, the trend has been to use brushless DC motors. They are also lighter compared to brushed motors with the same power output. The brushes in conventional DC motor wear out over time and may cause sparking. Thus, the brushed DC motor should never be used for operations that demand long life and reliability continues to rotate. This type of rotor has only one drawback. In figure 2.4, it is seen that only one coil remains energized. The two dead coils greatly reduce the power output of the motor. It can be fixed by energizing such way that when first set of coils rotating the rotor, the next coils are pushing the rotor as shown in figure 2.5. This is done by passing same polarity current through second coil. Electronic controller and sensor are used to control coil energization process. The combined effect produces more torque and power output from the motor. The combines torque also makes sure that BLDC has beautiful constant torque nature. Due to BLDC motor construction, they have good thermal characteristics and high energy efficiency. [7] To obtain a variable speed response, brushless motors operate in an electromechanical system that includes an electronic motor controller and a rotor position feedback sensor. [8] Brushless motors can also be used to drive linear actuators. [9]

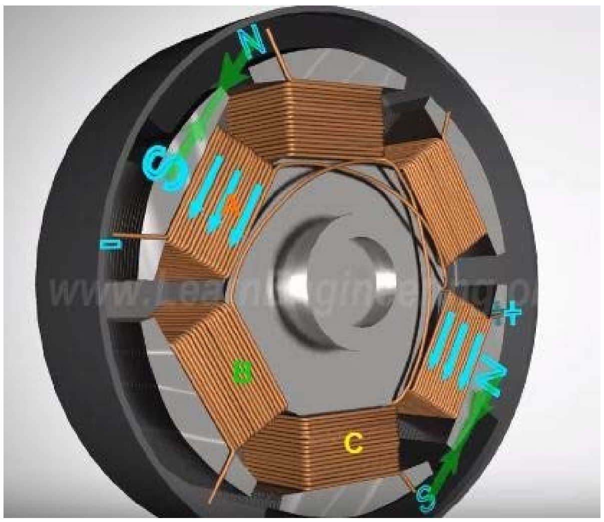

Figure 2.4: BLDC Motor with one coil energized [7]

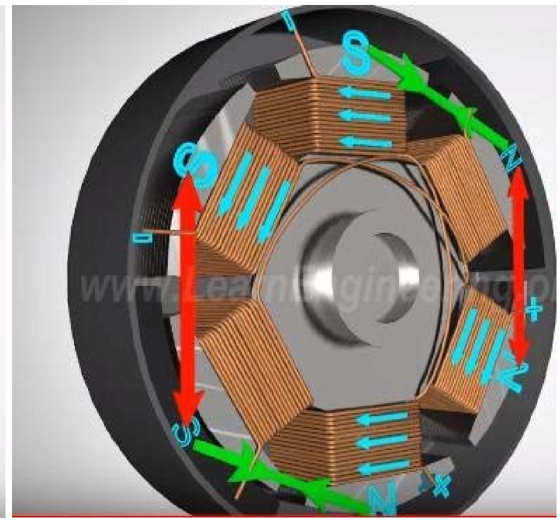

Figure 2.5: BLDC Motor with two coils energized

2.5 Stepper Motor

Stepper motors are DC motors that rather move in steps but not continuously. Multiple coils are organized and aligned in groups named "phases". Prompting the energization of each of the phases sequentially makes the motor to rotate one step at a time and hence, stepper motors are used to fulfill the requirements where the needs of precision motion control applications are required for the likes of printers, compact disc drives, CNC machines, plotters, image scanners, floppy disc drives, camera lenses, slot machines, conveyers, process automation and more recently, in 3D printers. The below shares some main reasons for the choice of using a stepper motor.

a.) Positioning – Since steppers move in precise repeatable steps, they shine better in applications requiring precise positioning such as 3D printers, CNC, Camera platforms and X, Y Plotters. Some disk drives also use stepper motors to position the read/write head.

b.) Speed Control – Precise increments of movement also allow for best control of rotational speed for process automation and robotics.

c.) Low Speed Torque - Normal DC motors have low torque at low speeds. A Stepper motor has maximum torque at low speeds, so they are a better option for applications requiring low speed with high precision.

The operating principle of stepper motor is very simple as the rotor rest position adjusts with a fixed angle rotation by using excitation switches from one section to any other. This is the basic concept for all the types of stepper motor. However, based on the machine structure and operation principle stepper motor has three types-

i. Variable Reluctance Motor (VRM)

ii. Permanent Magnet Stepper Motor (PMSM)

iii. Hybrid Stepper Motor (HSM)

2.5.1 Permanent Magnet Stepper Motor

Rotor comprises of a permanent magnet and operates on the attraction or repulsion between the rotor PM and the stator electromagnets.

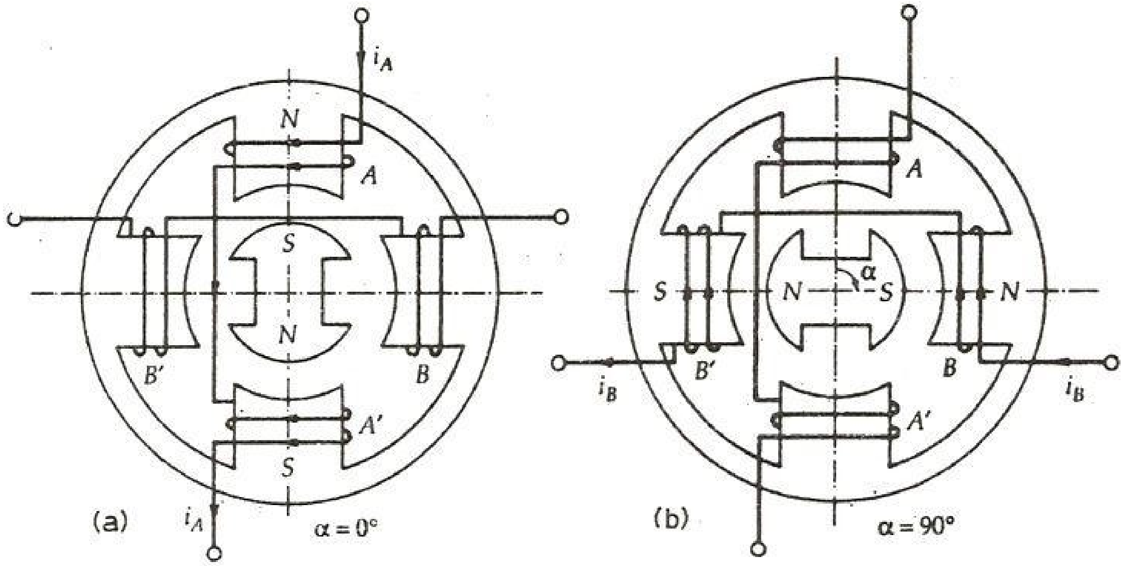

Figure 2.6: Permanent Magnet Stepper Motor

2.5.2 Variable Reluctance Stepper Motor

Variable reluctance (VR) motors have a plain iron rotor and operate based on the principle that minimum reluctance occurs with minimum gap, hence the rotor points are attracted towards the stator magnet poles.

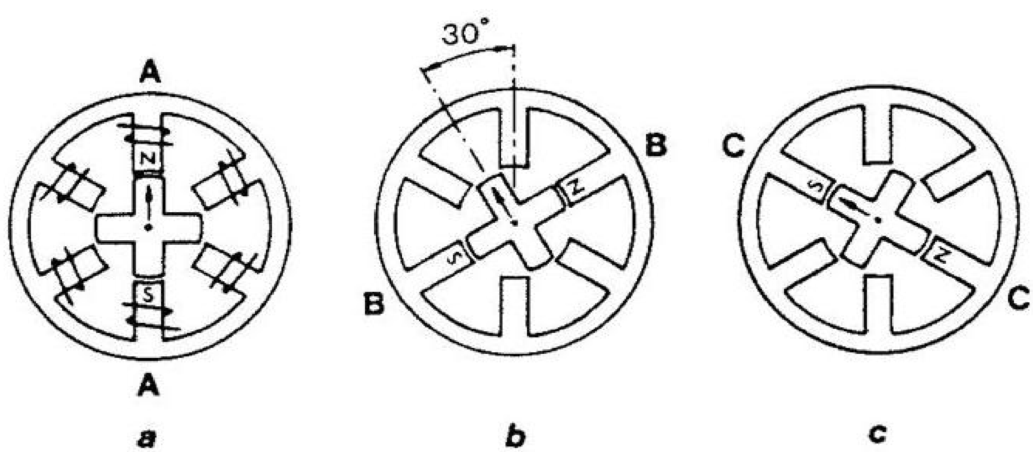

Figure 2.7: Variable Reluctance Stepper Motor

2.5.3 Hybrid Synchronous Stepper Motor

Hybrid Synchronous Stepper Motor uses a combination of permanent magnet (PM) and variable reluctance (VR) as to achieve maximum power in a small package size.

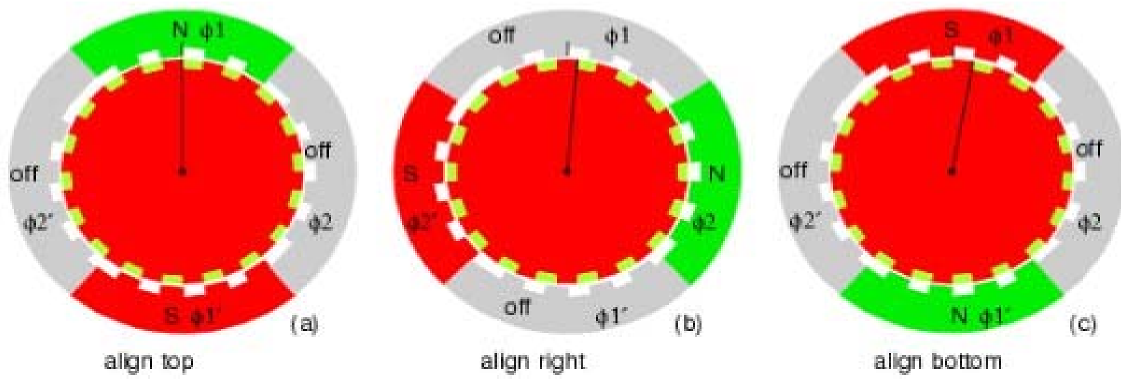

Figure 2.8: Hybrid Synchronous Stepper Motor

The rotor shaft holds one or more pairs of stacked laminations containing many teeth along its outside diameter. A permanent magnet between each laminated stack pair creates a north and south pole oriented along the motor shaft axis. Compared with servomotors, hybrid steppers have the advantage of rugged, simple construction; reliability with little maintenance; high torque at low speeds; and no need for position or velocity feedback devices. Comparative disadvantages include diminished torque at high speeds; resonance and noise; and high consumption of current. Where precise positioning is necessary, hybrid steppers best suit applications characterized by stable loads and speeds under 1,000 rpm.

2.6 Modes of Stepper Motor

The running modes determine how a stepper motor should be running, there are three commonly used excitation modes-single coil excitation, full step, half step and microstepping. Each of these excitation mode affects torque the motor delivers and also the running properties.

2.6.1 Single Coil Excitation

Each of the coils are energized one at a time, and the rotor moves to align itself with the coils.

2.6.2 Full Step Drive

Consequently, two neighboring coils are energized one step at a time, but making the resolution of stepper motor the same as that of the single coil excitation.

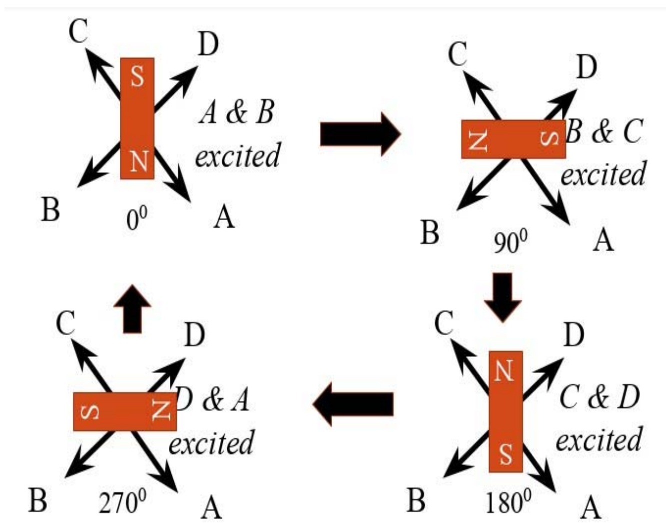

Figure 2.9: Full Step Drive

2.6.3 Half Step Drive

Consequently, two neighboring coils are energized followed by a single coil being energized in the direction of the rotation of the stepper motor one step at a time, making the resolution of stepper motor to be double than that of the single coil excitation or full step drive.

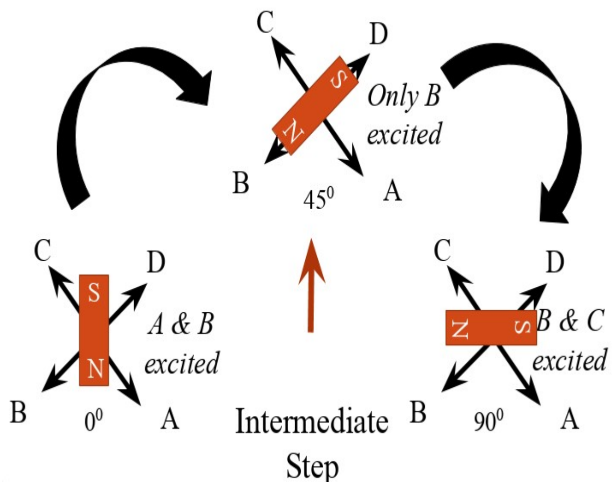

Figure 2.10: Half Step Drive

2.6.4 Microstepping

Applying full voltage to two coils makes the rotor to align in the middle of the two axes. Applying full voltage to one coil and zero voltage to adjacent coil makes the rotor to align to the magnetic axes of the first coil. Reducing the applied voltage gradually to the first coil and increase the applied to the second coil gradually from zero, makes the rotor to move from first to the second coil gradually. Gradual increase or decrease of voltages in several steps is termed microstepping. Microstepping yields precision that is more accurate over its position.

This chapter converses about the concept of DC motor and the fabrics of Stepper motor control theory, which is essential for readers to read before continuing to the next chapter. Next chapter will discuss about the proposed control system design for running the stepper motor in micro-stepping mode.

Chapter 3

Proposed Circuit and Simulation Synopsis

3.1 Proposed System Diagram

For our proposed design and development of a stepper motor position control system in microstepping mode using Atmega32 micro-controller, we follow the below diagram:

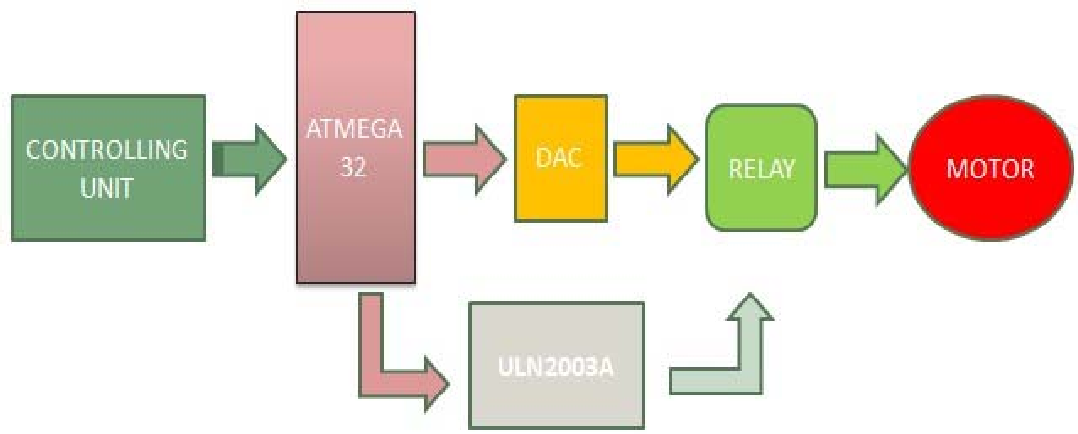

Figure 3.1: Circuit layout

3.1.1 ATmega32

There are 4 ports that provide parallel I/O interfaces to outside world: PortA, PortB, PortC, PortD. Each port provides 8 bidirectional digital I/O lines which are connected to ATmega32 pins provided that alternate functions are not selected on that port. Even though bidirectional, at any time the I/O line can either be Input and Output. PD2, PD3 and PB2 are used as interrupt enable pins, meaning that when these pins are enabled, the microcontroller pauses its current task just to perform another task with higher priority. The AVCC is the supply voltage pin for ADC, which should be externally connected to VCC. Normally 5V is applied to AVCC. AREFF is the analog reference pin for ADC. It is connected to suitable voltage if external ref is chosen.

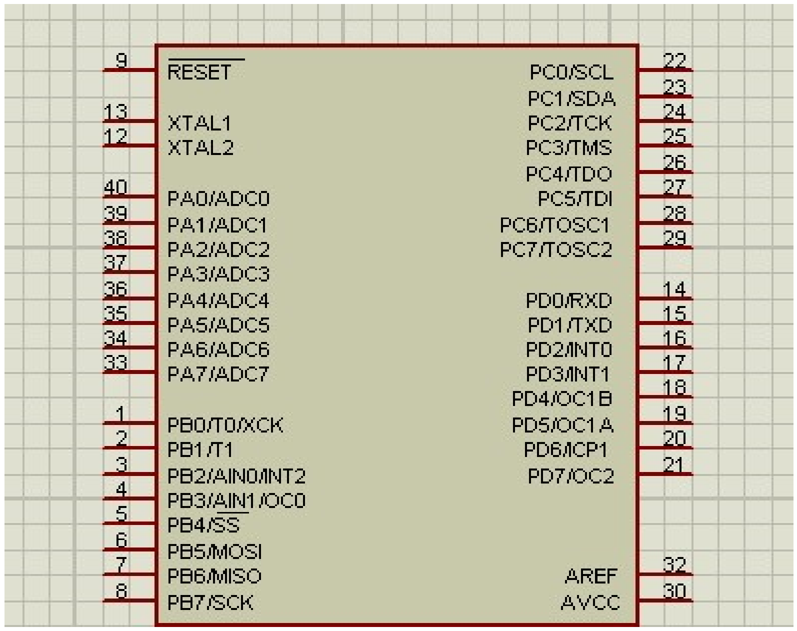

Figure 3.2: ATmega32 microcontroller

3.1.2 R-2R ladder DAC

DAC where only the resistor of R and 2R is used in the so-called R/2R DAC.

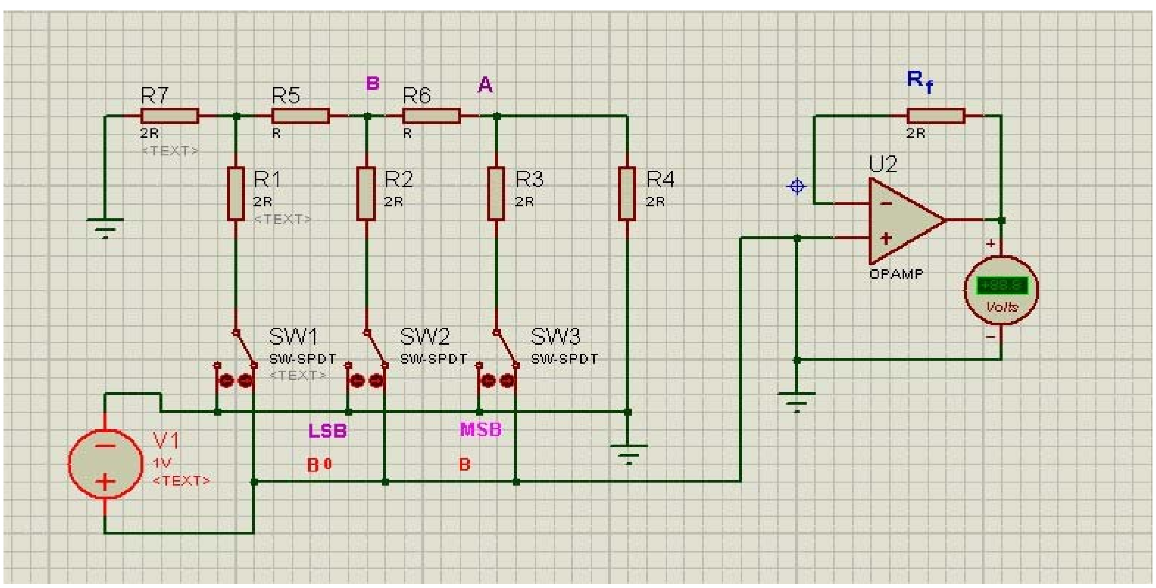

Figure 3.3: N bit R/2R Ladder DAC

$V_O = -V_R \sum_{n=1}^{N} B_n/2^{(n-1)}$ R where B_1 Is MSB; considering N is the number of bits $\sum I_n = V_R/2R \sum_{n=1}^{N} B_n/2^{(n-1)}$ where B_1 is MSB

$V_O = -R_f I_O$

$V_O = -R_f [I_1 + I_2 + I_3 + I_4 + I_n]$

$$V_O = -V_R \frac{R_f}{R} \left[\frac{B_1}{2^1} + \frac{B_2}{2^2} + \frac{B_3}{2^3} + \ldots + \frac{B_n}{2^n} \right]$$

$$V_O = -V_R \frac{R_f}{R*2^n} [B_1 2^{n-1} + B_2 2^{n-2} + B_3 2^{n-3} + \ldots + B_n 2^{n-n}]$$

The operation of the above ladder type DAC is explained with the binary word (B1B0= 01). The above circuit can be drawn as given in figure 3.3.

Applying the node analysis at A

$$\frac{V_A}{\frac{2}{3}R} + \frac{V_A - V_B}{R} = 0$$

Therefore, the above equation yields $V_B = \frac{5V_A}{2}$

Applying the nodal analysis concept at point (B), we get following equation-

$$\frac{V_A}{R} + \frac{V_A - (-V_R)}{R} + \frac{V_A - V_B}{R}$$

$$V_A = \frac{V_R + 4V_B}{2}$$

Substitution the equations of V_B in the nodal equations at node A and node B gives $V_A = -\frac{V_R}{8}$

The output voltage of the complete set up is as follows

$V_0 = -(2R/R)\ V_A$

$V_0 = -(2R/R)\ (-V_R/8)$

$V_0 = V_R/8$

$V_0 = V_R/2^n$; where N is number of bits

Resolution$= -V(1 - 1/2^n)$; where N is number of bits

Advantages of R/2R Ladder DAC

i. Only two resistor values (R and 2R)

ii. Does not require high precision resistors

Resolution, Speed, Linearity, Setting Time and Reference Voltage are some of the specifications of DAC.

3.1.2.1 Resolution

It is the smallest analog increment corresponding to 1 LSB change. A N-bit resolution can provide distinct analog value of 2N times.

Normally common DAC has a resolution of 8-16 bit.

3.1.2.2 Speed

It is the rate of conversion of a single digital input to its analog equivalent.

Conversion rate mainly depends on:

i. Clock speed of input signal

ii. Settling time of converter

While the input changes rapidly, DAC speed must be high.

3.1.2.3 Linearity

It is the difference between the actual output and the desired analog over the full range of expected values.

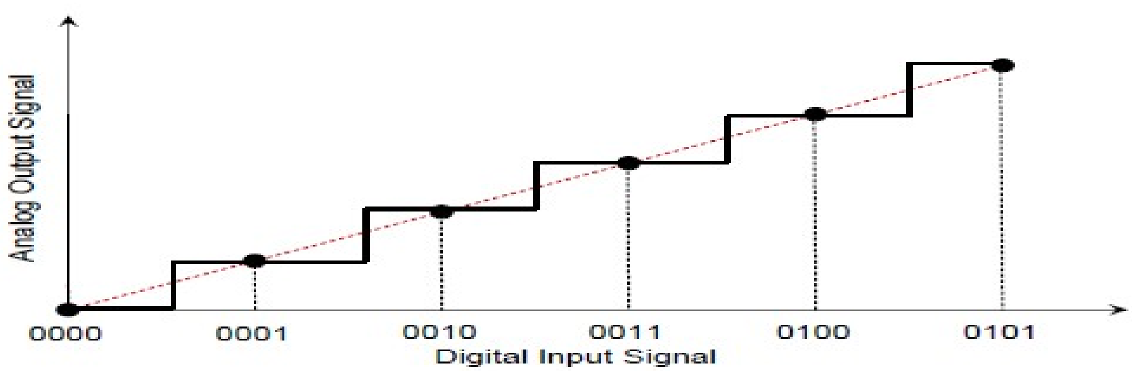

Figure 3.4: Linearity Specifications

3.1.2.4 Settling Time

Time is needed to settle the output signal within +/- ½ LSB of its final value after a given change in input scale limited by slew rate of output amplifier. Normally, a change in analog voltage would occur while adding a new binary word into DAC.

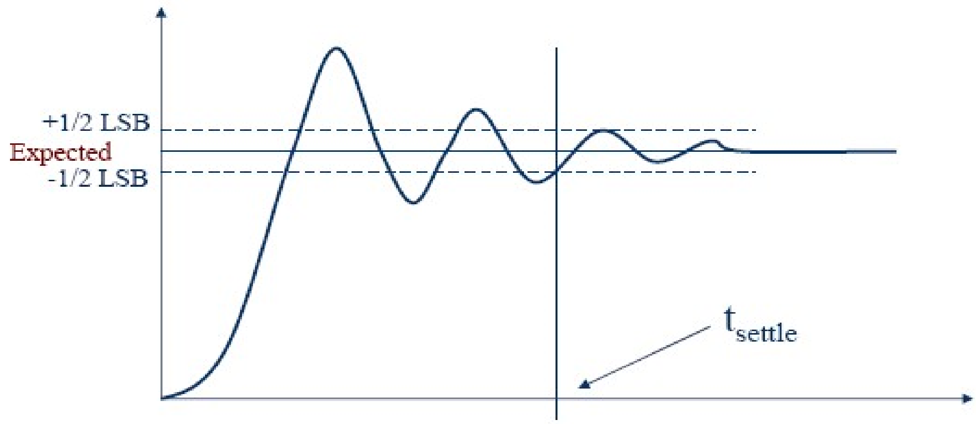

Figure 3.5: Settling Time Specifications of DAC

3.1.2.5 Reference Voltages

It is used for finding out how each digital input will be assigned to each voltage division

Types includes:

i. Non-multiplier DAC: fixed Vref.

ii. Multiplier DAC: external source provide Vref .

3.1.3 DAC Interfacing with Microcontroller

The DAC output value is derived using a resistor ladder with each end of the ladder tied to a positive and negative voltage reference input source. If the voltage of either input source fluctuates, a similar fluctuation will result in the DAC output value.

In these systems microcontroller generates output which is in digital form but the controlling system requires analog signal. In the figure shown, we use 8-bit DAC 0808 which converts digital data into equivalent analog current. For converting digital pulses to analog digital to analog converter is as device used widely. For creating DAC two methods are followed as binary weighted and R-2R ladder. DAC 0808 uses the R-2R method since it can achieve a high degree of precision. The first criterion for judging a DAC is its resolution, which is the function of the number of binary inputs. The common ones are 8, 10 and 12 bits. The resolution is decided by the number of data bit inputs since the number of analog output levels is equal to 2n, where n is the number of data inputs. DAC 0808 provides 256 discrete voltage or current levels of output. In DAC 0808, the digital inputs are converted into current, $I_{OUT}$and by connecting a resistor to I_{OUT} pin, we convert the result to voltage. The total current provided by I_{OUT} pin is a function of binary numbers at the A1-A6 pins inputs of DAC 0808, where A1 is the LSB, A6 is the MSB for the inputs. The inputs pins of DAC are connected to microcontroller pin of C_0-C_7 from where user selected the binary inputs. Hence, for converting the current into equivalent voltage I to V converter is needed and so we use an op-amp .We connect I_{OUT} to inverting terminal of op-amp. The diagram of microcontroller to DAC connection, followed by connection of op-amp to that of the DAC is as follows:

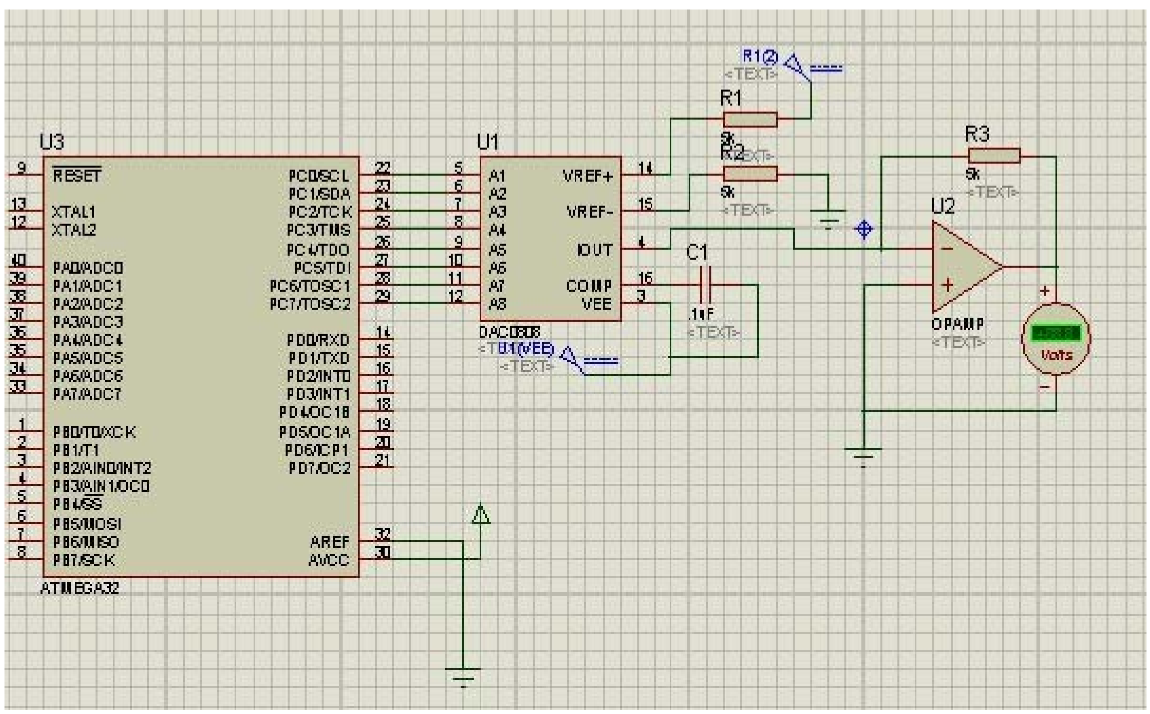

Figure 3.6: DAC interfacing with microcontroller

According to theory of DAC Equivalent, analog output is given as:

$$V_o = V_{ref} \left[\frac{A_1}{2^1} + \frac{A_2}{2^2} + \frac{A_3}{2^3} + \ldots + \frac{A_8}{2^8} \right]$$

3.1.4 Description of Relay Driver

ULN2003A [37] is a relay driver IC made of a Darlington array. It is made up of seven open collector Darlington pairs with common emitter. Moreover, ULN2003A has a capability of handling seven different relays simultaneously. A single Darlington pair made of two bipolar transistors which operate in the range of 500mA to 600mA current.

Relay Drive (ULN2003A) has 16 pins in total out of which there are:

1. 7 Input pins (Pin # 1 to Pin # 7) as in figure 3.9
2. 7 Output pins (Pin # 10 to Pin # 16)
3. 1 Ground pin (Pin # 8)
4. 1 COM pin (Pin # 9)

The ULN2003A is a Darlington transistor array which is made of seven NPN Darlington pairs that has a high-voltage outputs with internal suppression diodes which insure from problems associated with inductive loads. The high breakdown voltage and internal suppression diodes capable of inductive loads switching. A single Darlington pair collector-current rating is 500 mA.

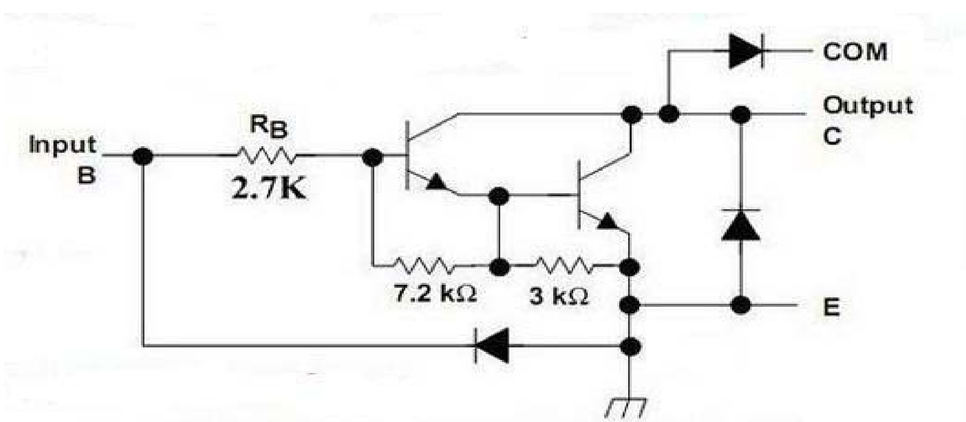

Figure 3.7: Internal circuit diagram of ULN 2003A (Darlington Array) [10]

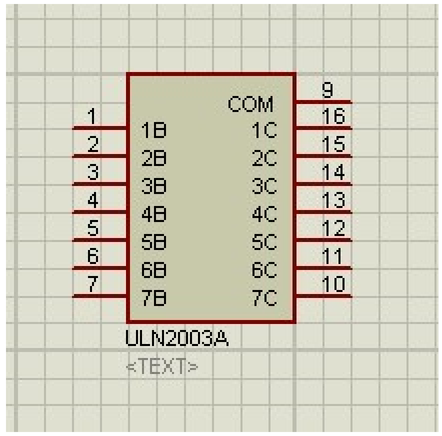

Figure 3.8: ULN 2003A

In real life, the relay circuit driver has a wide range of applications in real life. Major applications which are associated with ULN-2003A are given below.

i. Logic buffers.

ii. Line drivers.

iii. Relay drivers (for driving different loads).

iv. Lamp drivers.

v. LED display drivers (display devices).

vi. Motor (stepper and DC brushed motor) drivers.

3.2 Rotational Angle and Voltage Assignment Relationship

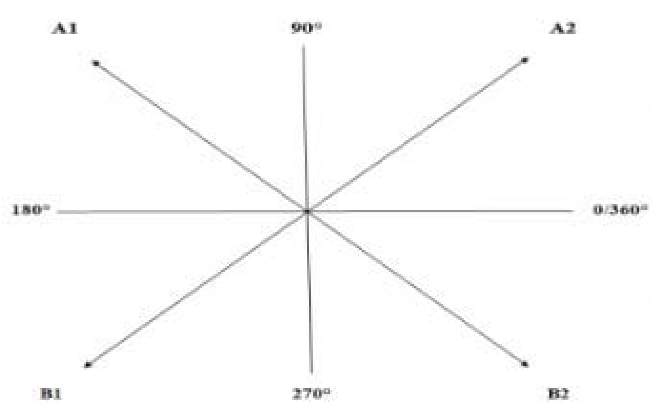

Figure 3.9: Angle and voltage relationship plot

The above figure shows what voltage should be applied to the necessary coils in order to generate desired stepper motor angles. Since 45 degree represents an angular displacement of 0, ° so to make the stepper motor to rotate to 0, ° 5cos45=3.536V must be applied to coil B2 and 5sin45=3.536V must be applied to A2. Similarly, (45+5.625) =50.625 degree represents an angular displacement of 5.625. ° So to make the stepper motor to rotate to 5.625, ° 5cos50.625=3.172V must be applied to coil B2 and 5sin50.625=3.865V must be applied to A2. Similarly, (45+11.25) =56.25 degree represents an angular displacement of 11.25. ° So to make the stepper motor to rotate to 5.625, ° 5cos56.25=2.778V must be applied to coil B2 and 5sin56.25=4.157V must be applied to A2. Similarly, (45+16.875) =61.875 degree represents an angular displacement of 16.875.° So to make the stepper motor to rotate to 16.875, ° 5cos61.875=2.357V must be applied to coil B2 and 5sin61.875=4.410V must be applied to A2. The same process is applied to all the cases to generate the desired angles.

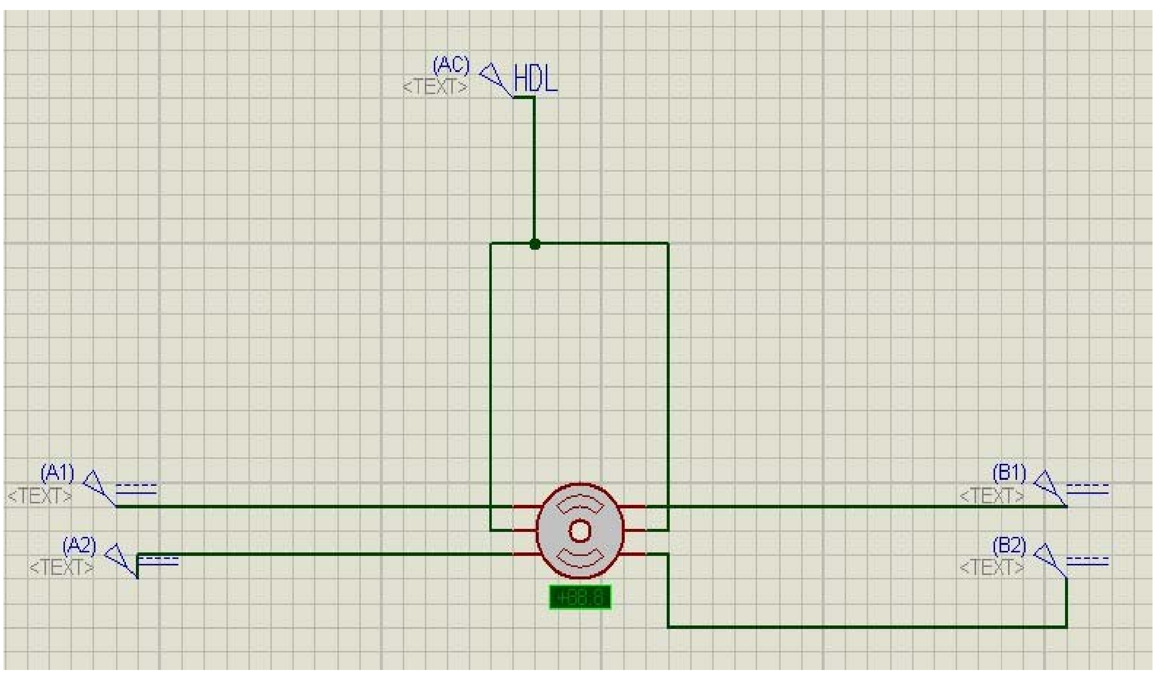

Figure 3.10: Stepper motor and their respective coil arrangements in proteus

3.3 Circuit Synopsis

Figure 3.11 shows circuit implementation in proteus to make the stepper motor move automatically and sequentially in micro-steps. There are in total four coils in unipolar stepper motor. Necessary and suitable voltages of assigned magnitudes must be applied to the appropriate and suitable coils of the stepper motor one at a time so as to make the stepper motor move sequentially and automatically in small steps.

Automatic and sequential rotation of stepper motor in micro-steps calls for the need of continuous and automatic conversion of each and every digital binary inputs in DAC (Digital to Analog Converter) one at a time to generate the correct and the right voltages one step at a time, which are to be fed to the suitable coils of the stepper motor vie relays so that it rotates in micro-step. That is why, the microcontroller, ATmega32 is connected to DAC, where automatic binary inputs are fed from PORTC and PORTB of the microcontroller one at a time to the DAC to be converted to analog ones. The whole integrations in shown in figure 3.11.

Since, there are four coils in stepper motor, four analog voltage channels are required to feed the coils with assigned voltages one at a time throughout the rotation of the stepper motor. This calls for the need of four DAC. But for the sake of reducing the number of pins in ATmega32 microcontroller, yet serving the same purpose of microstepping, two DAC are used that comes with the addition of four extra relays. But how does the relay work and manage the operations? The answer to this is in the next two consecutive pages.

Two DAC generates two separate voltage channels but the four coils of stepper motor must be fed by four separate voltage channels (one for each). This is where the concept of relay interferes out of the blue. Each DAC has got two relays, where each of the relay is connected to one corresponding coils, making each DAC to produce two separate voltage channels for every two coils via relays as shown in figure 1. Relays act as a simple switching mechanism between the DAC and the stepper motor.

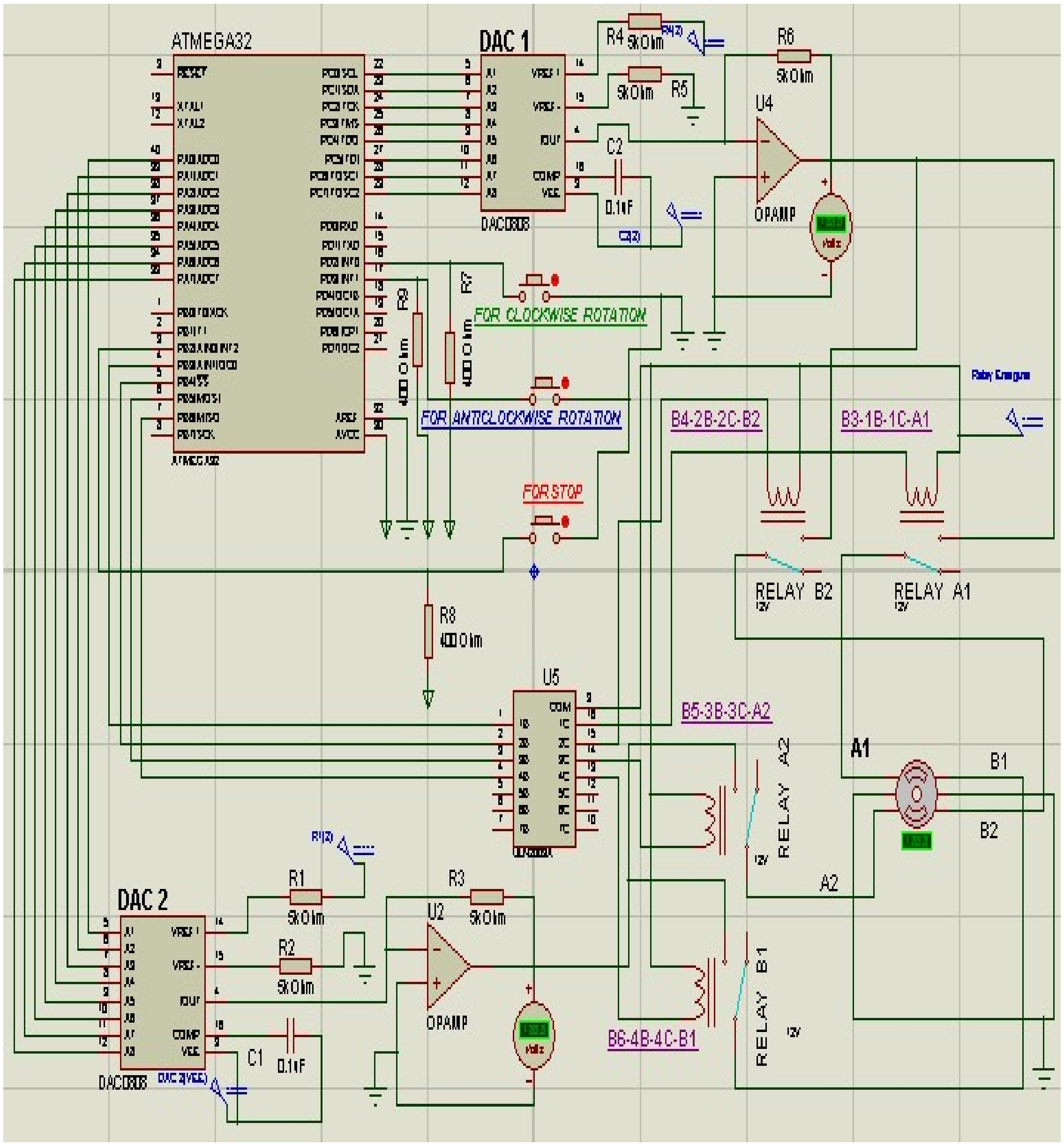

Figure 3.11: Design of the proposed circuit in proteus

Relays are programmed to operate through coding (***see appendix1***) by making use of the connection between the relay and the relay driver, ULN2003A and ATmega32 to such an extent that, when a motor wants to rotate from one angle to the other, necessary digital binary inputs are passed into the DAC and the correct analog voltages get produced , which are passed to the appropriate corresponding relays and relays will only be switched on the specific coils where the required analog voltages are needed to energize those coils to make the stepper motor move in micro-steps throughout its clockwise and/or anticlockwise rotation. Once again, figure 3.10 shows the entire connections of ATmega32, two DACs, relay driver, four relays and one unipolar stepper motor, where PORTC of ATmega32 is cascaded with DAC1 which is connected to Relay A1 and Relay B2, which further joins Coil A1 and Coil B2 respectively. PORTA of ATmega32 is cascaded with DAC2 which is connected to Relay A2 and Relay B1, which further joins Coil A2 and Coil B1 respectively.

In figure 3.10, Int0 is used for pending current task of any microcontroller and start clockwise rotation of unipolar stepper motor in micro-step angle of 5.625 degree. Int1 is used for pending current task of any microcontroller and start anticlockwise rotation of unipolar stepper motor in micro-step angle of 5.625 degree. Int2 is used for pending current task of any microcontroller and stop the rotation of unipolar stepper motor.

As shown in the diagram, each of the buttons, connecting to Int0, int1 and Int2 are connected to pull-up resisters of 400 Ohm. If pull-up resisters are not connected to those input pins, then it would be difficult to tell whether the state of the input is high or low. The state of the pin would be referred to as the floating state. In order to prevent this state, pull up resisters are connected to the input pins.

Table(3.1): Binary Inputs that are required to generate an Analog Voltage in DAC and thus, eventually generates theoretical rotational angles in degrees of unipolar stepper motor for microstepping for **clockwise rotation with a step angle of 5.625 degree**

DAC1 Inputs	**DAC2 Inputs**	**DAC1 Outputs**	**DAC2 Outputs**	**Voltages**				**Angle (in degrees)**
				A1	**A2**	**B1**	**B2**	
10110101	10110101	3.536V	3.536V	0V	3.536V	0V	3.536V	0
10100001	11000110	3.172V	3.865V	0V	3.865V	0V	3.172V	+5.625
10001110	11010101	2.778V	4.157V	0V	4.157V	0V	2.778V	+11.25
10011110	11100010	2.357V	4.410V	0V	4.410V	0V	2.357V	+16.875
01100010	11101100	1.913V	4.619V	0V	4.619V	0V	1.913V	+22.5
01001010	11110101	1.450V	4.785V	0V	4.785V	0V	1.450V	+28.125
00110010	11111011	0.970V	4.900V	0V	4.900V	0V	0.970V	+33.75
00011001	11111111	0.490V	4.970V	0V	4.970V	0V	0.490V	+39.375
00000000	11111111	0V	5V	0V	5V	0V	0V	+45
00011001	11111111	0.490V	4.970V	0.490V	4.970V	0V	0V	+50.625
00110010	11111011	0.970V	4.900V	0.970V	4.900V	0V	0V	+56.25
01001010	11110101	1.450V	4.785V	1.450V	4.785V	0V	0V	+61.875
01100010	11101100	1.913V	4.619V	1.913V	4.619V	0V	0V	+67.5
10011110	11100010	2.357V	4.410V	2.357V	4.410V	0V	0V	+73.125
10001110	11010101	2.778V	4.157V	2.778V	4.157V	0V	0V	+78.75
10100001	11000110	3.172V	3.865V	3.172V	3.865V	0V	0V	+84.375
10110101	10110101	3.536V	3.536V	3.536V	3.536V	0V	0V	+90

Chapter 4

Discussion

4.1 Limitations of Simulation

1.) Resolution: affects the outcome angle of the stepper motor. To theoretically rotate to 5.625 degree from 0 degree, stepper motor instead rotates to 5.66. This is all due to the facts of using 8-bit DAC which generates 4.98V and 4.08V rather than 4.976V and 4.083V respectively. The higher the resolution of the DAC, the greater the precision of the rotational angle.

2.) Hex to Binary conversion limit: to apply 5V to one of the coils of DAC, binary input of 11111111 is fed to the DAC via microcontroller. Again, to apply 4.976V to one of the coils of DAC, binary input of 11111111 is again fed to the DAC via microcontroller since 4.976V and 5V are being recognized automatically by a binary digit of 1111111. This is beyond the control of the operator. Due to this, stepper motor fails to rotate to the desired angle.

3.) No algorithm or loop: due to the systemic concept of MSB and LSB of binary inputs, no fixed algorithm or loop can be developed. All coding in AVR are done without the loop or algorithm.

4.2 Future Scope

1.) The Internet of Things (IoT) is driving a huge demand for a wide assortment of battery-operated devices. This in turn is driving the requirement for ever-increasing energy efficiency of microcontrollers and other system-level components. As a result, ultra-low power (ULP) has become an over-used marketing phrase, especially when used to describe microcontroller, ATmega32.

2.) Nowadays, microcontrollers are commonly used in many fields of industrial applications previously dominated by other devices. Their strengths such as: processing power, low cost, and small sizes enable them to become substitutes for industrial PLC controllers, analog electronic circuits, and many more. In first part of this article an overview of the Atmel AVR microprocessor family can be found, alongside with many scientific and industrial applications. [11]

3.) Adding a pad and also a wireless devices and a sensor, we can create a close loop system. Thus we can control the quality and precision of our output for the sake of reliability.

4.3 Conclusion

Though microstepping provides increased positional resolution and smoothness, simple setup and freedom from drift is not appropriate for all motion-control applications. Simple microstepping systems operate in open loop. There is no position feedback device to guarantee that the shaft position is correct. Normally this is not a problem for applications where shaft loads are relatively constant, such as *X-Y* tables, scanners, and packaging machines.

The repeatability of a positioning system is often the most important design parameter.

Accurate handling of position in terms of microstepping can be developed further by addition of higher resolution DAC.

References

[1] G. Baluta, "Microstepping mode for stepper motor control," in International Symposium on Signals, Circuits and Systems , Iasi, Romania, 2007.

[2] Sheng-Ming Yang, Feng-Chieh Lin, and Ming-Tsung Chen, "Micro-Stepping Control of a Two-Phase Linear Stepping Motor With Three-Phase VSI Inverter for High-Speed Applications," IEEE transactions on industrial applications, vol. 40, no. 5, September 2004.

[3] A. Astarloa, U. Bidarte, A. Zuloaga, and I.M. de Alegria, "Reconfigurable microstepping control of stepper motors using fpga embedded ram," in Proceedings of the 29th Annual Conference of the IEEE of the Industrial Electronics Society, Roanoke, VA, USA, pp. 2221-2226, 2003.

[4] de Silva, Clarence W, Modeling and Control of Engineering Systems. CRC Press. pp. 632–633. ISBN 1420076876, 2009.

[5] Moczala, Helmut, Small Electric Motors. London: Institution of Electrical Engineers. pp. 165–166. ISBN 085296921X, 1998.

[6] Xia, Chang-liang, Permanent Magnet Brushless DC Motor Drives and Controls. John Wiley and Sons. pp. 18–19. ISBN 1118188365, 2012.

[7] Ohio Electric Motors. DC Motor Protection. Ohio Electric Motors at WebCite, Archived November 28, 2011.

[8] Sabrie Soloman. Sensors handbook. 3rd ed. McGraw-Hill, Page 5-6, 1999.

[9] Peng Zhang. Industrial Control Technology: A Handbook for Engineers and Researchers. William Andrew, Inc., Page 91, 2008.

[10] L., Loflin. (n.d.). How to Use the ULN2003A Darlington transistor Array with Examples. from http://www.bristolwatch.com/ele/uln2003a.html Retrieved April 15, 2018.

[11] https://www.researchgate.net/publication/273379962_An_Overview_of_ATmega_AVR_Microcontrollers_Used_in_Scientific_Research_and_Industrial_Applications

Appendix 1

The following program represents the code in code vision AVR

```
#include <mega32.h>
#include <delay.h> #include
<stdlib.h>
interrupt [EXT_INT0] void ext_int0_isr(void)
{
while(1) //For infinite microstepping clockwise rotation of stepper motor using INT0 unless
INT0 is interrupted by INT2 to make the stepper motor stop or by INT1 for microstepping
anticlockwise rotation of the stepper motor
{
#asm("sei")
//for 0 degree
PORTB.3=0;
PORTB.6=0;
PORTB.4=1;
PORTB.5=1;
PORTA=0xFF;
PORTC=0xFF;
delay_ms(2000);
//for 0 degree
PORTA=0xAD;
PORTC=0xAD;
delay_ms(2000);
 //for +5.625 degree
PORTA=0x63;
PORTC=0x85;
delay_ms(2000);
```

```
//for +11.25 degree
PORTA=0XAB;
PORTC=0x71;
delay_ms(2000);
//for +16.875 degree
PORTA=0XE2;
PORTC=0x79;
delay_ms(2000);
//for +22.5 degree
PORTA=0x55;
PORTC=0x62;
delay_ms(2000);
//for +28.125 degree
PORTA=0XAF;
PORTC=0x52;
delay_ms(2000);
//for +33.75 degree
PORTA=0xDF;
PORTC=0x4C;
delay_ms(2000);
//for +39.375 degree
PORTA=0xFF;
PORTC=0x67;
delay_ms(2000);
//for +45.00 degree
PORTA=0xFF;
PORTC=0x00;
delay_ms(2000);
//for +50.625 degree
```

```
PORTB.4=0; PORTB.3=1;
PORTB.5=1;
PORTA=0xFF;
PORTC=0x67;
delay_ms(2000);
//for +56.25 degree
PORTA=0xDF;
PORTC=0x4C;
delay_ms(2000);
//for +61.875 degree
PORTA=0XAF;
PORTC=0x52;
delay_ms(2000);
//for +67.5 degree
PORTA=0x55;
PORTC=0x62;
delay_ms(2000);
//for +73.125 degree
PORTA=0XE2;
PORTC=0x79;
delay_ms(2000);
//for +78.75 degree
PORTA=0XAB;
PORTC=0x71;
delay_ms(2000);
//for +84.375 degree
PORTA=0x63;
PORTC=0x85;
delay_ms(2000);
```

```
//for +90.00
degree
PORTA=0xAD;
PORTC=0xQD;
delay_ms(2000);
… continue;
}
}
```