International Journal of Electronics Communication and Computer Engineering
Volume 6, Issue 5, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

A Precision Motion Control of a Circular Platform by using Stepper Motors

Baki Koyuncu, Hakan Koyuncu

***Abstract* – A computer interfaced stepper motor control unit is developed to generate motion for a circular platform. Stepper motor unit is connected to LPT parallel port of a computer and user instructions are entered through a system GUI to result circular motion around its motor axis. Control unit is constructed by using 4 power transistors in a driver circuit to drive the stepper motor continuously in 360 degrees with controlled speed and proper right or left rotations. An object oriented high level language is chosen to access the parallel port and send the proper digital pulses to rotate the stepper motor. A graphical user interface, GUI, is also developed to have the user machine interface to control the stepper motors.**

***Keywords* – LPT, GUI, Stepper Motor, Power Transistor, Darlington, DB25 Connector, Parallel Port.**

I. Introduction

Rotation of objects or platforms are carried out by using various motor controls [1]. Object rotations are carried out for many scientific reasons. One of them is laser scanners. Object is stationed at the center of a circular platform and rotated in front of a multi linear laser beams. The reflected laser beams which carry the surface information of the object are recorded and the 3D object is graphically generated in virtual media. Hence a circular rotating platform is essential for this kind of work. Rotation is obtained by electric motors which generate linear and smooth movements in time [2]. Stepper motor is another alternative for this kind of motion. Overall goal of this study is to generate a computer controlled rotational motion with a circular platform by using a stepper motor. There are 3 resulting tasks in the design. They are:

- Direction – The user must be able to rotate the platform in two directions
- Speed – The stepper motor must rotate at a fixed speed at any time interval. This speed can be varied at user request.
- Torque – The stepper motor's holding torque must be enough to sustain the weight of the motor assembly and the circular table.

Using a computer interfaced hardware circuit, positioning commands are fed into the stepper motor via a printer cable from the computer. The hardware amplified the signals through power transistors and relayed them to the stepper motor. Fig. 1 shows the block diagram of the deployed hardware.

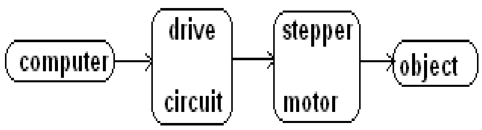

Fig.1. Block diagram of a computer interface Stepper motor and its circuit driving an object

II. Stepper Motor Theory

Stepper motors are structurally different than the standard dc motors. Instead of rotating continuously when power is applied, stepper motors move in small incremental steps. A stepper motor is constructed with several coil windings that must be energized in the proper sequence in order to achieve shaft rotation Proper energizing sequence will rotate the shaft in clockwise and reversal sequence will rotate in anticlockwise direction [3].

There are three basic types of stepper motors. They are identified as permanent magnet motor, variable reluctance motor and hybrid motor which is the combination of previous two.

The shaft permanent magnet stepper motor will turn fewer degrees as each pulse of current is received at the stator. The main feature is that a permanent magnet is used for the rotor, which means that no brushes are required. The drawback of this type of motor is that it has relatively low torque and must be used for low-speed applications. Its simple construction and low cost make it an ideal choice for non-industrial applications.

Hence unipolar permanent magnet stepper motor is deployed in this study [4]. Fig. 2 shows an example of 6 poles and 4 coils unipolar stepper motor.

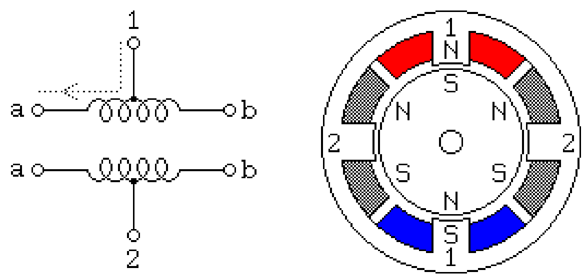

Fig.2. Simplified diagram of 6 poles and 4 coil unipolar stepper motor.

4 coils should be activated in a correct sequence in order to move the stepper motor. These sequences are called stepping modes. There are two stepping modes defined as single stepping mode and half stepping mode.

A. Single Stepping

Single stepping mode is the simplest stepping mode. In this mode, each successive coil is energized by applying a current pulse and the motor moves one full step at a time. Therefore, an example motor with a step angle of 7.5 degrees will rotate through 7.5 degrees with each step. Table 1 shows the single stepping sequence of a single stepping motor.

International Journal of Electronics Communication and Computer Engineering
Volume 6, Issue 5, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

Table 1: Single stepping pulse coil sequence

Pulse	Coil1	Coil2	Coil3	Coil4
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1

B. Half Stepping

The difference between the half and single stepping for the same step rate is that, half stepping gives you half the speed but twice the resolution of a single step. For an example single stepper motor with a step angle of 7.5 degrees, half stepper motor will result 3.75 degrees of rotation. Table 2 shows the half stepping sequence of a half stepping motor.

Table 2: Half stepping pulse coil sequence

Pulse	Coil	Coil2	Coil3	Coil4
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

III. Hardware

In this study, the deployed stepper motor requires at least 1 ampere current and 5 volts voltage per phase. Parallel port of a typical computer can only deliver a few milliamp (mA) of current and 5 volts of voltage for each motor phase.

Hence a 5W power resistor can pull the necessary current to stepper motor during each phase. However, standard resistors are not capable of pulling large currents without overheating problems. Additionally, resistors cannot handle fast current switching required by the stepping sequence. Inductive loads of the coils tend to make the current reverse its direction because of the rapid on and off switching.

Darlington transistors can solve these problems. It amplifies the computer's output current at its parallel port from a few mA's into a 500mA input to the stepper motor. Two transistor are needed in parallel to supply the necessary 1A to the stepper motor. Fig. 3 show the internal schematic diagram of Darlington transistor.

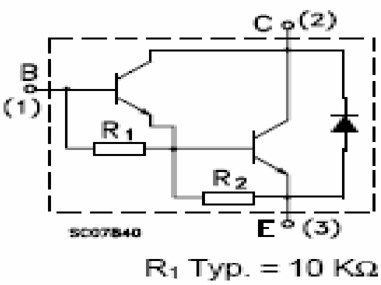

Fig.3. Internal schematic diagram of a Darlington transistor [5]

A LED between transistor base and the ground is used to see the coil which is active at any time. Stepper motor driver circuit with LEDs are constructed on a printed circuit board and each collecter of darlington transistor is connected to a phase of the motor in sequence[6]. This circuit is shown in the Fig. 4.

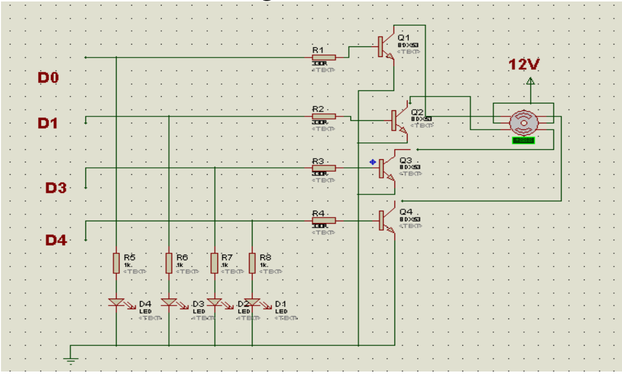

Fig.4. Complete stepper motor driver circuit diagram

Motor driver circuit is interfaced to computer through its parallel port. A DB25 connector is deployed for this purpose. Pin connections of DB25 is shown in Fig. 5

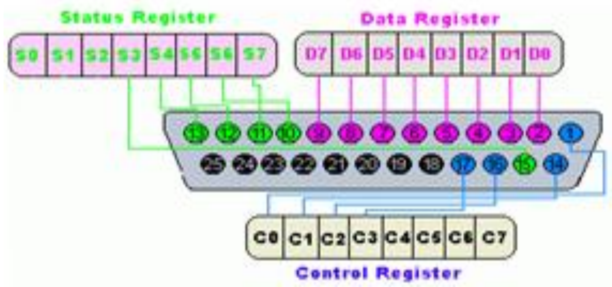

Fig.5. Pin connections of DB25 connector [7]

A parallel port contains a set of signal lines that the CPU sends or receives data with other external devices. When a PC sends data to a device using a parallel port, it sends 8 bit of data (1 byte) at a time. These 8 bits are transmitted parallel to each other [8].

DB25 connector pins are divided into three groups; Data, Control and Status registers. These registers are IO mapped and each has a unique address. For a typical PC, the base address of LPT1 is 0x378[9]. In this study, LPT1 is used and the address of the data bits is 0x378 in hexadecimal form. 4 pins, D0, D1, D2, D3, in data register are deployed to transmit current pulses for 4 motor phases. D0 D1, D2 and D3 pins of DB25 male connector are connected to the bases of 4 Darlington transistors. Collectors of Darlington transistors and the cables of the four coils of the stepper motor are connected to +12 volts power supply. Additionally, anodes of the LEDs are connected to D0-D3 pins so that the user can see which coil is active when 5v is sent from parallel port of the computer via D0-D3.

DB25 male connector parallel port has 8 pins that have to be grounded. Hence, these 8 pins, cathodes of the LEDs and the common wire of the stepper motor is grounded to complete the circuit.

IV. Platform

The stepper motor driver circuit is developed to rotate an object on a circular platform. 3D object scanner as shown in Fig. 6 rotates an object 360 degree with the help of stepper motor in 3.5 degree steps. The body of the motor is fixed in a rectangle wooden box. A light weighted circular wooden platform is placed on the rotating axis (shaft) of the stepper motor at its center point. It's made sure that there is no contact between box and platform. The ideal gap is taken as 1 cm. Scanner platform diameter is considered as 30cm and light weighted objects can be placed on the platform to be scanned with maximum 500 grams.

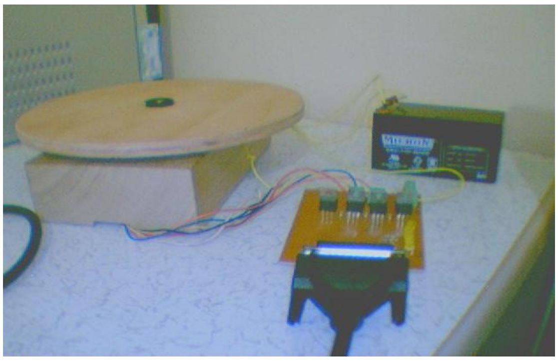

Fig.6. Scanner platform and driver circuitry

V. Software

In Windows 95 & 98, the user can access to the parallel port directly. In other versions, accessing I/O Ports in protected mode is governed by two events, The I/O privilege level in the EFLAGS register and the I/O permission bit map of a Task State Segment (TSS).

There are only two I/O privilege levels used. These are level 0 & level 3. User mode programs will run in privilege level 3, while device drivers and the kernel will run in privilege level 0. This separation allows operating system to distinguish programs with not enough privilege to access the I/O ports.

Input32.dll can work with all the windows versions without any modification in user code or the dll itself. When functions are called, input32.dll checks the operating system version. If the operating system is WIN9X, the input32.dll access directly to port. However, if the operating system version is higher, the dll installs a kernel mode driver and talk to parallel port through that driver. The user code does not aware of the OS version on which it is running.

Energizing each coil depends on stepping modes. Each coil is addressed so that the data must be sent to this address over parallel port properly. For example, if the base address is 0x378, CPU addresses D0 and sends a 1 to D0 port to make it active and to energize the coil 1.

Two algorithms are developed and implemented in the software. In first algorithm; angle, direction, stepping mode and delay time of the rotation are considered as inputs. Stepper motor is rotated according to these values. In second algorithm only stepping modes are entered.

A. Implementation of First Algorithm

- Take the stepping mode, direction, angle and delay time and then calculate number of steps
- Control the stepping mode and if the stepping mode is single stepping
- Energize the coils in selected direction with single stepping mode for number of step times
- Else if the stepping mode is half stepping
- Energize the coils in selected direction with half stepping mode for number of step times
- Stop and release the stepper motor.

B. Implementation of second algorithm

- Take the stepping mode from the user.
- Control the stepping mode and if the stepping mode is single stepping
- While mouse wheel up energize coils in clockwise with single step sequence
- While mouse wheel down energize in counter clockwise with single step sequence
- Else if the stepping mode is half stepping
- While mouse wheel up energize coils in clockwise with half step sequence
- While mouse wheel down energize in counter clock wise with half step sequence
- Stop and release the stepper motor.

VI. Graphical User Interface

A graphical user interface, (GUI), is constructed to control the stepper motor movement . It provides to user 2 way control to motor. One of them is the angular control in 2 way direction. The other one is the usage of mouse wheel for rotation.

Additionally, a motor simulation is shown for the user to see the coil position during the motor rotation. The GUI is shown in Fig. 7.

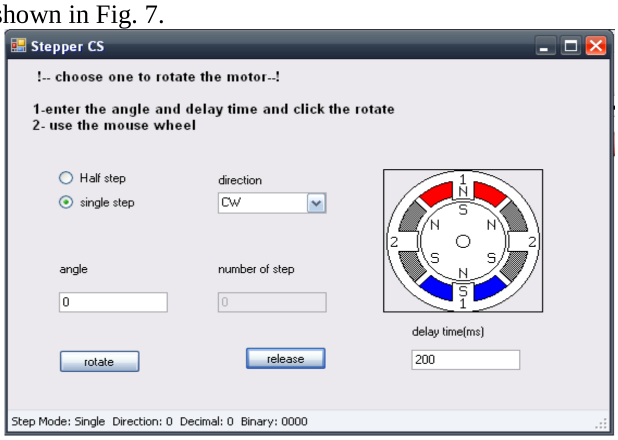

Fig. 7. Graphical user interface for developed software

VII. Conclusion

A simple an efficient platform rotator is developed for all practical purposes. Rotating the platform with intended angle, speed and direction 3D rotation of objects become easer. The circular motion of the platform is linear. Uniform motion of the steps makes the recording of the returned laser beam very stable during the 3D scanning.

Continuous motor motion introduces a sliding effect of the recordings. But with stepper motor control the motion and the recordings are snychronized and recording is carried out at the end of each step.

The graphical user interface is easy to understand for any user. Extra motional features can be added or removed.

In conclusion, controlling the stepper motor and the scanner platforms are succesfully achieved from a computer.

REFERENCES

[1] http://www.ams2000.com/stepping101.html

[2] An Introduction To Stepper Motors For The Experimenter ,David Benson

[3] http://coeleveld.com/attachments/081_How%20stepper%20motors%20work.pdf

[4] http://www.digikey.com/videos/en/v/What-is-a-Permanent-Magnet-Stepper-Motor/1646392885001

[5] https://en.wikipedia.org/wiki/Darlington_transistor

[6] Roger C. Brendemuehl, “Stepper motor control circuit”, Beloit Corporation, Beloit(1980)

[7] http://www.integration.co.nz/FAQ-Pinouts.php

[8] Parallel Port Complete, by Jan Axelson

[9] http://logix4u.net/parallel-port/43-how-to-read-parallel-serial-port-address-from-bios

AUTHOR'S PROFILE

Baki Koyuncu

is a professor in computer eng dept of Ankara university. He has received his BSc, Msc and PHD from Birmingham and Loughborough universities in united Kingdom . He is an IEEE senior member. He teaches computer hardware and software courses. His research areas are image processing , computer hardware , embedded systems , wireless sensor networks and RFIDs.

Hakan Koyuncu

is in computer science department of Loughborough University, UK. He has currently finished his PhD research in the area of RFIDs and position detection of objects in indoors. His research interests include wireless sensors, wireless networks and mobile Communication. He has completed his MSC in multimedia and wireless networks. He has developed algorithms to determine the accurate positions of the mobile and stationary objects.