Push system for Aluminium tube by using a stepper motor with the Arduino

P P Joshi*, V S Jorapur

Department of Mechanical Engineering, Fr. Conceicao Rodrigues College of Engineering, Fr. Agnel Ashram, Bandstand, Bandra (W), Mumbai–400050, India, Affiliated to Univ. of Mumbai.

E-mail: pranavjoshi174@gmail.com

Abstract: The research content of this paper is to design a push system for an Aluminium tube to be cut on the cutting machine. The system consists of one driving pulley, one idler pulley, and a timing belt. The stepper motor will drive the pulley and the stopper attached to a timing belt will push the Aluminium tube further. This paper describes the process of developing a push system using a stepper motor driver, membrane keypad, Nokia LCD, and a microcontroller board. Different types of stepping mode and the formula for the number of pulses to be sent to a stepper motor are derived and tested. The result of the paper shows that a stepper motor along with the electronic components improves accuracy for a push application and can be used.

1. INTRODUCTION

The stepper motor is a digital electromechanical system where a user sends an electrical pulse input which results in a rotation of the rotor by a specific angle called step angle[1]. The stepper motors are used in industries worldwide for robotics and automation due to their availability in different sizes and economical price. Applications of stepper motors include CNC machines, 3-D printers, flatbed scanners, security cameras, etc. These applications require precise positioning and accurate control over their position. The stepper motor works very precisely even without feedback by the position sensor. The open-loop operation of the stepper motor makes it cheaper.

A push system is used in this project for an extruded Aluminium tube of a rectangular cross-section to be pushed on the cutting station using a stepper motor. The motive is to push an Aluminium tube up to a required length using a stopper and after reaching its stop, the human operator will cut the tube which will be of a specific length entered using a keypad. The goal here is to assist the worker to push the tube further on the cutting station up to a required length that needs to be cut. This will save time as it eliminates the process of marking and then cutting.

2. LITERATURE REVIEW

The push system for the Aluminium tube needs very precise movements which comes in very handy by using a stepper motor. A stepper motor is used where accuracy is of utmost priority such as in robotic applications. A stepper motor is the most suitable motor for this system because the motor runs with respect to the number of pulses sent to a driver. The motor is very useful due to its properties such as high torque at low speeds, low cost, and reliability[1].

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

Published under licence by IOP Publishing Ltd
Stepper motors have some problems such as jerk produced by the motor while rotating the shaft by the bigger step angle. It creates dynamic instability in the motor and results in a very noisy operation. However, the step angle can be reduced by using microstepping mode. In microstepping mode, the full step angle is subdivided into multiple small-angle steps called microsteps. G. Baluta concludes that using microstepping mode, the noise was reduced substantially, position control became more precise and the high step resolution is achieved[2].

If the load torque is stable, operating at low speed, the stepper motor can be used in an open-loop as it requires a simple control algorithm[3]. The stopping position of the system is independent of load change and only depends upon the pulse frequency and the number of pulses sends to a stepper motor driver[4]. The stepper motor coils are always excited when the power is on which induces holding torque inside the stepper motor[5]. Due to a holding torque, a self-locking function is created by a stepper motor[4]. This self-locking function is very important as it will help to hold the position of the timing belt.

The stepper motor should move the stopper from one location to another location, generally through acceleration, constant speed, and deceleration. If the speed starts by activating the frequency exceeding the limit frequency of the stepper motor the step phenomenon occurs, which results in not starting correctly. If you suddenly stop to the end, due to inertia, the motor can overshoot a few steps which will reduce positional accuracy[6]. Therefore using steady acceleration and deceleration will provide better results.

3. CALCULATION OF NUMBER OF PULSES REQUIRED TO MOVE THE STOPPER

The stepper motor is also called a pulse motor. When a single pulse is sent to the motor, the motor shaft rotates by a certain degree of angle. Most of the stepper motors have a rotation of 1.8° per step, which means 200 steps per revolution.

In many applications, microstepping can increase system performance, and lower system complexity and cost, compared to full-step and half-step driving techniques. Microstepping can be used to solve noise and resonance problems, and to increase step accuracy and resolution. Microstepping modes are ranging from 1/2 to 1/32 of the full step. Higher the microstepping, lower the vibration and noise[7].

By using EasyDriver as a stepper motor driver, the microstepping up to 1/8th of the full step can be achieved[8]. The EasyDriver will help to drive the stepper motor in 1/8th microstepping mode. It is set as default settings.

The stepper motor runs on the pulses sent to it by a stepper motor driver. But the EasyDriver by default runs on microstepping mode. So either computer or PLC has to do the calculations for the required number of pulses.

3.1 Factors affecting the number of pulses

3.1.1 Stepping mode

Each pulse will result in a certain angle of rotation of the shaft. In this case, due to the use of an EasyDriver as the stepper motor driver, one full step will become 8 microsteps. The stepper motor takes 200 full steps to complete one revolution. It will need 1600 pulses or microsteps to complete one full revolution.

3.1.2 Pitch profile of the pulleys and the timing belt

The pulley and the timing belt will also be the factor in calculating the number of pulses. The pulley and the timing belt need to have the same tooth profile to interconnect with each other. The timing belt and pulleys have standard tooth profiles such as GT 2, GT 3, or GT 5 where the number indicates the pitch length. That means GT 2 pulley or timing belt will have a 2 mm pitch length as shown in figure
3.1.3 Number of teeth on the pulleys
The number of teeth and the pitch diameter needs to be the same for driving pulley and idler pulley to push the Aluminium tube linearly on the cutting machine. The length of the timing belt will depend upon the diameter of the pulley and the distance between the driving pulley and the idler pulley.

Let,

- The pitch of the timing belt and pulley = 2 mm i.e. GT 2 profile
- Number of teeth on both pulleys = 20 teeth
- Angle per full step taken by stepper motor = 1.8°
- Total full steps to complete 1 revolution = $360°/1.8° = 200$ full steps
- Stepping mode = $1 / 8^{th}$ step per pulse
- Pulses need to move the rotor one full step = 8 pulses
- Pulses need to move rotor one full revolution = 200 steps x 8 pulses = 1600 pulses

Now one revolution of the rotor is one revolution of the driving pulley as it will be interconnected

Hence,

One revolution of the driving pulley = Number of teeth x Pitch length

$= 20 \times 2 \text{ mm} = 40 \text{ mm}$

Now the relation between length traveled by the stopper attached to the timing belt and the pulses is

$1600 \text{ pulses} = 40 \text{ mm}$

Equation (1)

Equation (1) is completely dependent upon the three factors i.e. stepping mode, pitch profile of the pulleys, and the number of teeth on the pulleys.

By using Equation (1), the formula for the number of a required pulse will be

$$\text{Required number of pulses for entered distance} = \left(\frac{Z \times 1600}{40}\right),$$

Where, $Z = \text{Entered distance}$

4. CALCULATION OF TORQUE REQUIRED TO PUSH ALUMINIUM TUBE

A stepper motor should provide an output torque larger than load torque and be required to start and stop at a proper step rate against load inertia.

4.1 Types of torque
4.1.1 Detent torque
When a hybrid stepper motor is in a state of rest and not energized, a small torque must be applied to the rotor to break through the static equilibrium state which is called detent torque[5].

4.1.2 Holding torque
When the windings are powered and the rotor is at the balanced position, an external torque should be
applied to a hybrid stepper motor to rotate the rotor one full step. This torque is called holding torque. Holding torque maintains the rotor position at equilibrium irrespective of load. And it depends upon the magnitude of the current\[5]\.

So holding torque is a very important characteristic in the stepper motor. Also, the stepper motors are selected by calculating the holding torque required by the load.

In this case, the load is Aluminium tube, and the timing belt which is made of neoprene synthetic rubber.

The formula for torque is,
\[ T = F \times r \times \sin \theta \]

Where 
- \( T \) = Torque 
- \( r \) = Radius to apply the force 

The stepper motor will be pushing the Aluminium tube parallel to the timing belt hence \( \theta \) will become 90° and \( \sin 90^\circ = 1 \) as shown in figure 2.

![Figure 2. The direction of force acting on Aluminium tube](image)

\[ F = (\text{Weight of Aluminium tube (Kg)} + \text{Weight of timing belt (Kg)}) \times 9.8 \text{ m/s}^2 \]

The weight of the Aluminium tube and timing belt is calculated in designing software, which approximately comes 2 Kg for Aluminium tube and 0.7 Kg for the timing belt of pitch 5 mm.

- Weight of Aluminium tube = 2 Kg
- Weight of timing belt = 0.7 Kg

Hence,
\[
F = (2 \text{ Kg} + 0.7 \text{ Kg}) \times 9.8 \text{ m/s}^2 = 26.46 \text{ Kg.m/s}^2 = 26.46 \text{ N}
\]

\( r \) = radius of pulley (m)

The diameter of the pulley depends upon the number of teeth and pitch.

Let us select a bigger pulley with a 5 mm pitch and 40 numbers teeth.

Diameter of pulley = Circumference / \( \pi \)
\[ = 40 \times 5 / 3.142 = 63.69 \text{ mm} \]

Radius of pulley = 31.85 mm = 0.03185 m

Now multiplying Force with radius,
\[
T = F \times r \\
= 26.46 \text{ N} \times 0.03185 \text{ m} \\
= 0.842751 \text{ N.m}
\]

4.2 Stepper motor selection based on torque ratings

Stepper motor torque will be needed to be more than the calculated torque as there are some factors which motor torque needs to overcome such as friction between pulley and timing belt, and friction between Aluminium tube and cutting machine. If the mechanical load increases above maximum electromagnetic torque, step loss will occur\[9]\.

Also, the holding torque mentioned by the manufacturers is specified, when the full rated current is supplied.

Generally, the stepper motor torque is shown in Kg.cm for ease of understanding.

So, converting N.m to Kg.cm
0.842751 N.m = 8.6 Kg.cm
Stepper motors are available in a very wide range of holding torque which is around from 1 Kg.cm to 500 Kg.cm. For the safety of a pushing system, selecting a Stepper motor twice the power will be suitable for this project i.e. 17.2 Kg.cm. NEMA 23 stepper motors have much higher holding torque than NEMA 17. Here, 23 indicate faceplate dimensions of stepper motor i.e. 2.3in x 2.3 in. Stepper motors are available in the standard faceplate dimensions such as NEMA 17, NEMA 23, NEMA 34, etc. The larger the faceplate, the larger the core will be, inducing higher torque.

5. SYSTEM DESIGN

The system design consists of software and hardware. The software implementation is done using Arduino IDE software which is based on the C programming language. The hardware consists of the components mentioned in table 1. The figure 3 shows the connection between various components

5.1 List of components

| Sr. No. | Components |
|---------|------------|
| 1       | Arduino MEGA 2560 R3 |
| 2       | 4 x 4 Membrane Keypad |
| 3       | Nokia 5110 Display (84 x 48 pixels) |
| 4       | EasyDriver Version 4.4 |

5.1.1 Arduino MEGA 2560 R3 Microcontroller Board
The Arduino is open-source hardware designed by the Arduino team in Italy. There are two types of Arduino which are mainly used viz. Arduino UNO, Arduino MEGA, etc. The only feature which apart MEGA from UNO is the processor. Arduino MEGA uses ATmega 2560 processor while UNO uses ATmega 328. ATmega 2560 is a 100 pin surface mount 8-bit microcontroller[10].

The processor can operate at 16 MHz that is 16 million transactions per second. The flash memory is 256K byte which is much more than the UNO. Flash memory is used for storing a program. Also, the number of ports on the MEGA is more than the UNO.

5.1.2 4 x 4 Membrane Keypad
The membrane keypad is a package with multiple switches. There can be multiple combinations of many switches on the keypad. Here 4 x 4 represents 4 rows and 4 columns. The keypad has flat cable lines for each row and each column. In this case, there are eight numbers of ports four for rows and four for columns.

When the key is depressed, the key’s respective row and column get activated and the pulse is identified by the pins of Arduino. Each key has its row number and column number. The combination of a row and a column assertion can be decoded to determine which key is pressed [10].

5.1.3 Nokia 5110 Display
The Nokia 5110 display is a graphic LCD useful for a lot of applications. It was intended originally to be used as a screen for cell phones and was used in lots of mobile phones during the '90s era. It uses a low powered CMOS LCD controller/driver PCD8544, which drives the graphic display of size 84x48 pixels.[11] There are four backlight LED’s fixed on the border of the display. The backlight intensity can be adjusted according to the need. The display uses a very low amount of current around 6 to 7 mA, so the use of resistors is suggested.[12]
5.1.4 EasyDriver Version 4.4 for stepper motor

EasyDriver Version 4.4 is the name of the driver for the stepper motor which is used in this project. EasyDriver can drive up to about 750mA per phase of a bipolar stepper motor. The driver by default use 1/8th step microstepping mode. This setting can be easily overridden by tying the MS1 and/or MS2 pin to the ground to set the driver to use 1/8, 1/4, or 1/2 microstep mode. The driver only needs 2 types of signals from the Arduino i.e. number of steps to rotate the shaft and the direction in which the shaft is supposed to rotate. The driver can sustain voltage from 6 volts to 30 volts. The A3967 driver chip carefully controls the current to each coil to cause the rotation to move to positions in between the normal full step positions. This allows for smoother motion, reduced mid-band resonance, and higher positional accuracy and resolution.

The Easy Driver can operate in 1/8th, 1/4, half, and full step modes. These four modes are selected by the logic levels on the MS1 and MS2 input pins as shown in table 2. Normally, the pull-up resistors on the Easy Driver hold MS1 and MS2 high, which results in a default setting of 1/8th microstep mode. By making MS1 and/or MS2 ground, the above mentioned results can be achieved[8].

| MS1   | MS2   | RESOLUTION       |
|-------|-------|------------------|
| Low   | Low   | Full Step        |
| High  | Low   | Half Step (1/2)  |
| Low   | High  | Quarter Step (1/4) |
| High  | High  | Eighth Step (1/8) |

The selection of EasyDriver makes the default resolution of step 1/8th of the full step. So, now 1 full step of the motor will be equal to 8 microsteps. And the rotation angle per step will become 1.8°/8. So by using the EasyDriver for the stepper motor, the motor will run with much less noise and vibration.

![Figure 3. Connections between components](image)

Table 2. Resolution setting of EasyDriver
6. HARDWARE IMPLEMENTATION

The above mentioned components are going to be used with the stepper motor. There will be two pulleys at both ends and the timing belt will be attached in between them. The stepper motor shaft will be attached to the driving pulley. And at the other end, the idler pulley will be placed. Both the posts, the driving pulley post, and the idler pulley post will be bolted down to the table as shown in figure 4. In the figure, the stopper attached to the timing belt is shown red. The supporters are used as guiderails to guide the Aluminium tube to the cutting machine.

![Figure 4. Rendered image of pushing system with Aluminium tube](image)

The Aluminium tube will be kept on the supporter which will constrain its motion in only one direction. The Aluminium tube will slide on the supporter when the stopper will push the tube. Both posts and the supporter are shown transparent just for the good visibility of the pushing system.

7. MINI WORKING MODEL

The mini working model here is used for obtaining results by applying various hypotheses. The written program is compiled and ran using Arduino MEGA. The Nokia display is connected to Arduino MEGA without the use of level shifters as shown in figure 5.

The EasyDriver is used to drive the NEMA 17 stepper motor which needs a low amount of current than the NEMA 23 stepper motor. The temperature of the A3967 chip installed on the driver can go up to 139°C. However, the EasyDriver has a current limit pot set to limit a current flow which will regulate the current flowing to the stepper motor. The EasyDriver has a thermal shutdown temperature of 165°C[8] and if the heat sink is used then the driver can be used at higher currents also.

But a bigger stepper motor such as NEMA 23 will need a bigger driver. This EasyDriver has a limit of 750mA/Phase. So it is recommended to use a bigger version of the EasyDriver.
Figure 5. A miniature model with electronic components

The SparkFun website provides a bigger version of the EasyDriver which can withstand current up to 2A/phase. The Big Easy Driver, designed by Brian Schmalz, is a stepper motor driver board for bipolar stepper motors up to a max 2A/phase. It is based on the Allegro A4988 stepper driver chip. Also, the advantage of using this driver is it runs by default on 16 steps microstepping mode which means more step resolution[13].

8. RESULTS
1) By using a stepper motor amongst all types of motors high holding torque and precision can be achieved.
2) The holding torque value depends upon the amount of current per phase.
3) Microstepping mode offers a low noise operation and high step resolution.
4) EasyDriver helps to drive the stepper motor in microstepping mode that is 1/8th of a full step. And it is by default action taken by the EasyDriver
5) The gradual acceleration and deceleration can be achieved using the AccelStepper library in Arduino programming.
6) The calculated number of pulses rotates the stepper motor shaft precisely to its required location.

9. CONCLUSION
It is concluded from the above results, by using a stepper motor with the help of EasyDriver, effective control is achieved over stepper motor speed. Microstepping gives more step resolution. While the gradual acceleration and deceleration of the stepper motor speed help to remove jerk and noise. The calculated number of pulses provides the required precision to attain the required position by a stopper. The results are sufficient to conclude that this whole system can be used as a push system for an Aluminium tube.

ACKNOWLEDGMENT
I am grateful to Mr. Vishal Dakshini and Mr. Manthan Dakshini for always supporting and motivating me with their immense knowledge and for letting me do this project.

REFERENCES
[1] B. Aranjo, P. K. Soori, and P. Talukder, “Stepper motor drives for robotic applications,” 2012 IEEE Int. Power Eng. Optim. Conf. PEOCO 2012 - Conf. Proc., no. June, pp. 361–366, 2012.
[2] G. Baluta, “Microstepping mode for stepper motor control,” ISSCS 2007 - Int. Symp. Signals, Circuits Syst. Proc., vol. 2, pp. 609–612, 2007.
[3] J. W. J. Ngoc Quy Le, “An Open-loop Stepper Motor Driver Based on FPGA,” Int. Conf.
[4] X. Wang, “Research on Stepper Motor Control System based on Single Chip Microcomputer,” vol. 50, no. Iceecs, pp. 105–108, 2016.

[5] Y. Fan, “3D finite element solution of a hybrid stepper motor,” Powerconversion Intell. Motion, vol. 16, no. 1, pp. 19–20, 22, 2019.

[6] X. Wei, “Acceleration and deceleration control design of step motor based on TMS320F240,” Procedia Eng., vol. 15, pp. 501–504, 2011.

[7] Japan Radio Corporations, “Microstepping,” p. 11, 2015.

[8] “Easy Driver Stepper Motor Driver.” [Online]. Available: https://www.schmalzhaus.com/EasyDriver/.

[9] J. De Viaene, S. Derammelaere, and K. Stockman, “Load angle estimation for dynamic stepping motor motion applications,” Mechatronics, vol. 53, no. July 2018, pp. 229–240, 2018.

[10] S. Barrett, Arduino Microcontroller Processing for Everyone!, Third Edit. Laramie, Wyoming: Morgan & Claypool Publishers, 2013.

[11] D. T. Htwe, “ARDUINO BASED UV CARE SYSTEM WITH GSM MODULE.”

[12] “Interface Nokia 5110 Graphic LCD Display with Arduino.” [Online]. Available: https://lastminuteengineers.com/nokia-5110-lcd-arduino-tutorial/.

[13] Sparkfun, “Big Easy Driver.” [Online]. Available: https://www.sparkfun.com/products/12859.