ISSN No:-2456-2165

Arduino Uno and Bluetooth based Speed Control of DC Stepper Motor with Reduced Jerking Level

Soumyendu Banerjee
Department of Electrical Engineering
University of Engineering and Management Kolkata
Kolkata, India

Dibyajyoti Das
Department of Electrical Engineering
University of Engineering and Management Kolkata
Kolkata, India

Sanjay Bhadra
Department of Electrical Engineering
University of Engineering and Management Kolkata
Kolkata, India

Debayan Ghosh
Department of Electrical Engineering
University of Engineering and Management Kolkata
Kolkata, India

Abstract:- This work suggests controlling the speed of a stepper motor using an Arduino Uno. The HC-05 Bluetooth module and an Android application were used to implant the control scheme, providing the stepper motor with wireless control. The main goal of this study, along with speed control, was to lessen the stepper motor's jerking while changing each step. The stepper motor's jerking is a serious issue since it occasionally prevents the desired step count from being reached and makes noise while the motor is running. The jerking intensity was therefore reduced in this work by using a sinusoidal pulse amplitude modulation (PAM) to lessen the rapid change in input signal. Using the accelerometer ADXL345 to test the jerking level, it was determined that sinusoidal PAM could lessen the stepper motor's jerking even when the step input signal was given directly.

Keywords:- *Stepper Motor; Arduino Uno; Jerking Level; Sinusoidal PAM*

I. INTRODUCTION

Stepper motors are a kind of electromechanical transmission device, which converts digital pulse signal into angular or linear displacement. Driving mobile or moving objects from one place to another is the primary function of a stepper motor. Three stages, acceleration, and constant-speed deceleration are part of the process. More advanced, customized, and regulated stepper motors are now available on the market thanks to control systems. Stepper motors are primarily used in controlling of various types of joints in robotics [1] and mechanized assembly lines in many sectors such as open and closing doors etc. [2-3].

Now, as implied by its name, a stepper motor operates in steps, mostly dependent on input pulse signals supplied by a microcontroller unit to each phase wire of the stator component [4]. By modifying the appropriate logic to run the motor as necessary, the use of a microcontroller unit makes the provision of input pulses much simpler [5]. This unit requires low power consumption components, but their main drawback is that they frequently fail to supply the motor with the required level of current, necessitating the attachment of an external driver module [6]. The driver increases the current level to provide necessary amount of power to drive the motor.

The step angle, rotor speed, and produced torque in the rotor shaft are taken into account and used for managing the further rotation of the rotor shaft, however, in the controlling strategy of stepper motor employing microcontroller unit like Arduino, Raspberry Pi [7]. Additionally, a crucial role is played by time-optimized stepper motor control mechanisms to modify the driving mechanism of stepper motors [8].

In addition, rotor shaft acceleration and deceleration control is a crucial field of research using optimization approaches. Additionally, it improves the topology of the interface and lowers the noise level for the motor system. By offering a closed loop control technique, MATLAB simulating software [9-11] played a crucial contribution in the field of stepper motor speed control. Therefore, better precession and increased current level are demonstrated using vector control drive for stepper motors [12].

The stepper motor now exhibits jerking during each step of rotation during the transition phase of the input signal, which breaks down the system and produces undesired noise [13]. Although the amount of jerking cannot be directly detected, an accelerator is a sensor that can measure acceleration, static and dynamic vibration, and other things [14-15]. An accelerometer will henceforth be utilized for measuring earthquakes, motion detection, and servomotor control. In essence, the accelerometer output signal detects the amount of vibration whenever a vibration occurs. In this study, a stepper motor's jerking was measured using an accelerometer, model ADXL335. To lessen vibration and resonance in the system, research is also being done on hybrid stepper motors to minimize damping, noise, and jerking levels.

To lessen the jerking level in this work, we employed sinusoidal PAM signal in place of a direct step input signal to the stepper motor [16]. Wave drive, full step, and half step modes were used to operate the steeper motor (micro stepping were neglected). Using an accelerometer to generate a normal step input signal, the amount of jerk in each step was measured. The same driving modes were employed in the

ISSN No:-2456-2165

following phase's sinusoidal PAM, and the jerking level was once more measured. When comparing these two measurements of jerking, it can be shown that the average amount of jerk is significantly less for sinusoidal PAM signals.

Fig 1:- Picture of an Arduino Uno board.

Working Voltage	5 V
Input voltage	7-12 V
Input voltage	6-20 V
Digital I/O Pins	14
DC current per I/O pin	40 mA
DC current for 3.3V pin	50 mA
Flash Memory	32 KB
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

Table 1:- Arduino Uno specifications

II. METHODOLOGY

Stepper motors are used in a variety of items, including robotics, telescopes, surveillance equipment, and medical devices. The key benefit of utilizing a steeper motor is that it rotates using a step signal, allowing the speed and position of the rotor to be adjusted in accordance with changes in step angle. However, step change causes the rotor to jerk during the transition time of each step input signal, resulting in undesired noise and system failure. So, the primary goal of this study was to reduce the level of jerking. Due to this, even though the motor was receiving step signals, sinusoidal PAM was still delivered with each step using an Arduino Uno, and the degree of jerking was determined using an accelerometer. Arduino Uno received the control signals via Bluetooth and an Android app. Finally, a comparison of jerking measurement between a sinusoidal PAM and a regular step signal was given. The following parts provide an explanation of the entire process.

A. Normal operation of 8-pole stepper motor

Arduino Uno is an open source platform based on ATmega328 microcontroller. It comes with a programmable circuit board i.e., microcontroller and a software to control Arduino Uno board, shown in Fig. 1. Specifications of this board is tabulated in Table I.

A 28Ybj-48 DC, 5V, 4 phase, 5 wire stepper motor with a ULN2003A Driver Board was utilized in this project. In essence, it is a unipolar stepper motor with five wires connecting to the main supply voltage and one wire for each phase. In Section III, Fig. 5 depicts the internal wiring diagram of a unipolar stepper motor. The phase pins are illuminated as necessary in accordance with the direction of rotation, and the COM pin is constantly linked to a high input signal. The phases A, B, C, and D need to be activated for rotation in the FORWARD mode, which is counterclockwise, while phase D, C, B, and A need to be energized for rotation in the BACKWARD mode. Both situations have unidirectional, or unipolar, current in each phase.

Now, the order in which each of the four phases is energized determines the step angle for rotation and the torque that is created in the rotor. The stepper motor, as its name implies, requires a step input signal to rotate. For each step rotation, one or two phases are activated, and the remaining phases are left un-powered. A stepper motor typically operates in three different modes: 1) Full Step, 2) Half step, 3) Wave drive mode. Each mode is described below

B. Wave drive mode

In this mode, each phase are energized at each time with other phases are un-energized. Fig. 2(i) shows the step input signal for wave drive mode. The main disadvantage of this mode is jerking level is very high. Also, as each phase is energized at each time, magnetic field strength is also lower, thus torque is also low, but in this mode, the motor rotates on its normal step angle.

C. Full step mode

In this mode, the next two phases in line are always energized while the other phases are de-powered. The step input signal for complete step mode is shown in Fig. 2(ii). Since two phases are activated simultaneously, this mode has the advantage of a stronger magnetic field than wave drive mode. As a result, torque is higher in this mode compared to wave drive mode, but since rotation is still based on a full step angle, jerking level is also higher, almost on par with wave drive mode.

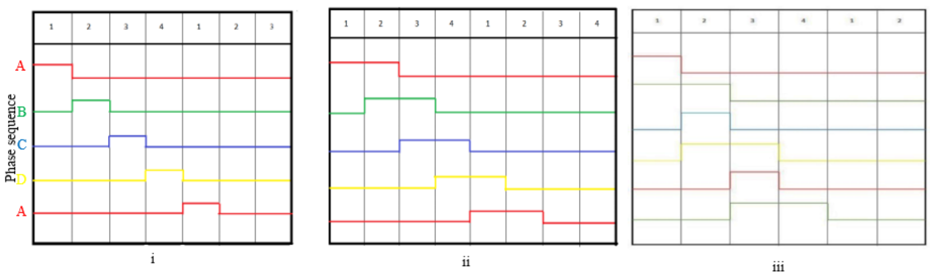


Fig 2:- Stepper motor driving modes: (i) wave drive mode; (ii) full step; (iii) half step.

ISSN No:-2456-2165

D. Half step mode

In this mode, a phase is first activated, followed by that phase and its subsequent phase. The step input signal for complete step mode is shown in Fig. 2(iii). The primary benefit of this mode is that the average magnetic field strength and torque are between those of the full step and half step modes. However, because the rotor rotates in this mode at half its regular step angle, the jerking level is the lowest of these three modes.

E. Sinusoidal PAM

In each of these modes, a step input signal is sent to all phases; hence, only one phase needs a HIGH input signal, or a 5 V supply, from the controlling device for the duration of time ‘t1’. Fig. 3 is an illustration of this. Fig. 3(i), where a single-phase 5 V input signal is given for the time period ‘t1’, as step signal. The main drawback of this input signal is that, during the transition periods, from ‘a’ to ‘b’ and from ‘c’ to ‘d,’ the input voltage abruptly increases quickly from 0 to 5 V and 5 to 0 V, respectively, which causes the system to jerk excessively. This primary goal of this paper was to, despite direct step input, lower the jerking intensity at the input signal's transition time by using sinusoidal PAM signal, shown in Fig. 3(ii). It is evident that sinusoidal PAM reduces jerking since there is no abrupt or quick shift in input voltage level. We used the analogue input/output pins of the Arduino Uno to supply this sinusoidal value so that the values could be changed appropriately, which is not possible in the digital realm. Table II displays the values that were selected for a sinusoidal input signal. For the Arduino Uno, the maximum input voltage of 5 volts was scaled to an analogue output value of 1023, and this value was sent to analogue pins. In this approach, the analogue values, ranging from 0 to 1023, were scaled down from the range of input voltage levels in the digital domain, between 0 and 5 V.

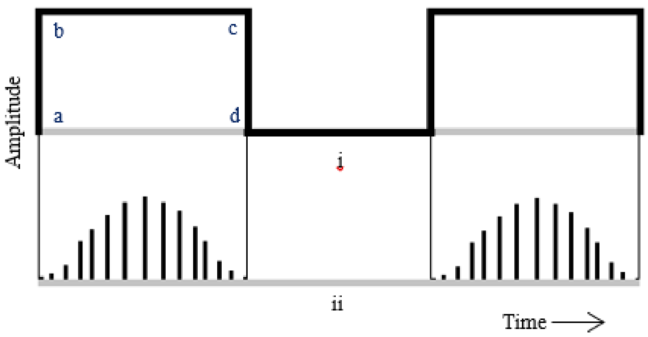


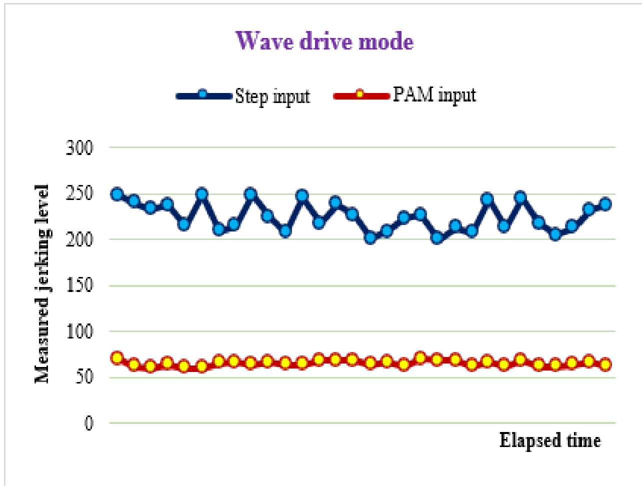
Fig 3:- Input signal pattern: (i) step input; (ii) sinusoidal PAM.

Analog input voltage	Corresponding Arduino values
0	0
0.6265	128
1.24	254
1.83	377
2.40	493
2.93	601
3.42	700
3.85	788
4.22	863
4.52	925
4.75	973
4.91	1005
4.99	1021
5	1022

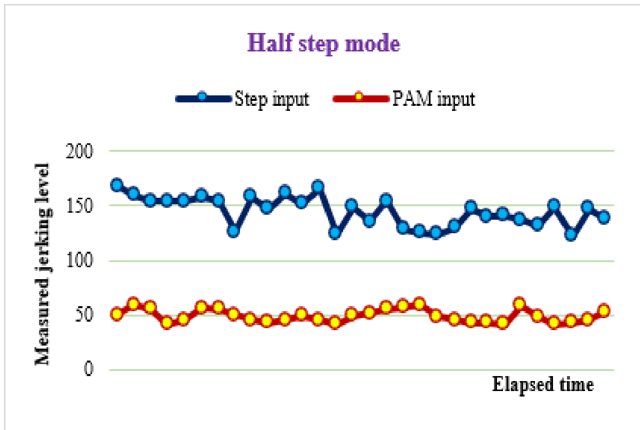
Table 2:- PAM signal values

III. RESULT AND DISCUSSION

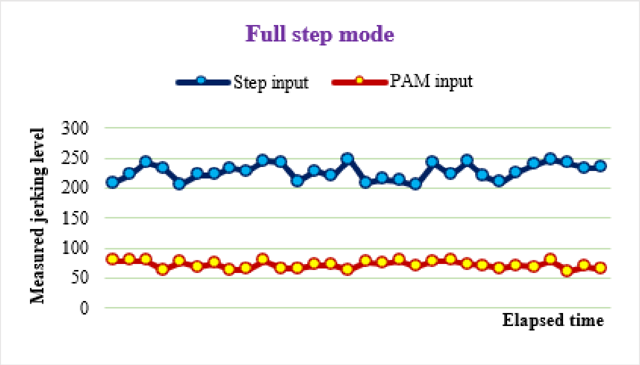
An accelerometer was fixed to the main motor to measure how much each of the stepper motor's three operating modes jerked. The degree of jerk was then shown on the serial monitor of Arduino Uno and over Bluetooth in an android application. Fig. 4 depicts the measured jerking levels for the wave drive, full step, and half step modes of operation. Each figure displays the jerking level for the step input signal in ‘blue’ and the PAM signal in ‘orange’. These statistics clearly show that, despite the use of step signals, the jerking level is significantly decreased for PAM input signal. For wave drive, full step, and half step, respectively, the jerking level was reduced by approximately 71.43%, 68.72%, and 66.34%. In Fig. 5, the circuit schematic is displayed. As an extension of this study, this technique can be evaluated using Raspberry-Pi [17-20].



i



ii



iii

Fig 4:- Jerking level measurement using step and PAM input.

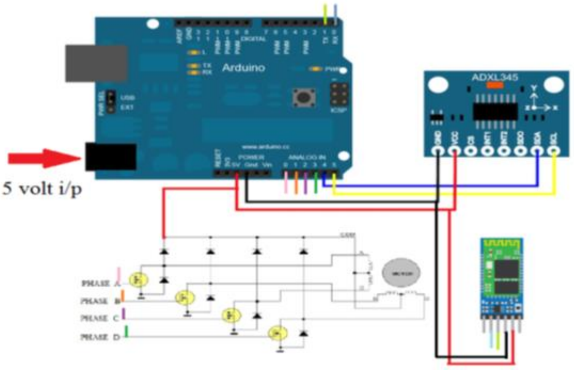


Fig 5:- Circuit connection of proposed work.

IV. CONCLUSION

This paper proposes a topology for controlling the speed of a DC stepper motor using an Arduino Uno. Along with it, the main emphasis was placed on lowering the stepper motor's jerking intensity when changing each step. In order to do this, sinusoidal PAM with the same time period was delivered to each phase despite employing a direct step input signal. An accelerometer was used to measure the jerking level reduction, and an overall reduction of 68.83% jerking level was attained.

REFERENCES

[1]. S. Banerjee, G. K. Singh, "Feedback Control Topology of n-DOF Robotic Manipulator and Optimum Positioning of End-Effector using PSO," in proc of 2nd IEEE Conf. on Applied Signal Processing, pp. 41-45, 2020, Kolkata, India.
[2]. Ł. Przeniosło and M. Hołub, "Development of microprocessor, time optimized stepper motor driving algorithm," in proc of IEEE Intern. Conf. Meth. Mode. Autom. Rob., pp.174-179, 2017, Poland.
[3]. T. Kanjo and A. Sugawara, "Stepping Motors and Their Microprocessor Controls". Oxford, U.K.: Clarendon, 1994.
[4]. D. O. Carrica, S. A. Gonzalez, M. Benedetti, "A high speed velocity control algorithm of multiple stepper motors," Mechatronics, vol. 14, no. 6, pp. 675-684, 2014.
[5]. W. Ruifeng, W. Zhe and W. Liying, "Stepper Motor Control based on AT89S51 Microcontroller," in proc of IEEE 8th Intern. Conf. Intel. Comp. Tech. Autom., pp. 844-847, 2015, China.
[6]. L. Zhang *et al.*, "Research on stepper motor motion control based on MCU", in proc of IEEE Chin. Auto. Cong., 2017, pp. 3122-3125, China.
[7]. Ł. Przeniosło and M. Hołub, "Development of microprocessor, time optimized stepper motor driving algorithm," in proc of IEEE 22nd Intern. Conf. Meth. Mod. Autom. Robot., pp. 174-179, 2017, Poland.
[8]. C. -K. Lai, J. -S. Ciou and C. -C. Tsai, "FPGA-based stepper motor vector control system design," in proc of IEEE Intern. Auto. Cont. Conf., pp. 1-5, 2017, Taiwan.
[9]. S. Banerjee and G. K. Singh, "Deep neural network based missing data prediction of electrocardiogram signal using multiagent reinforcement learning," Biomed. Sig. Proc. Cont., vol. 67, 102508, 2021.
[10]. S. Banerjee, R. Gupta, J. Saha, "Compression of Multilead Electrocardiogram Using Principal Component Analysis and Machine Learning Approach," in proc of IEEE App. Sig. Proc. Conf., pp. 24-28, 2018, Kolkata, India.
[11]. S. Banerjee and G. K. Singh, "Quality Aware Compression of Multilead Electrocardiogram Signal using 2-mode Tucker Decomposition and Steganography," Biomed. Sig. Proc. Cont., vol. 64, 102230, 2020.
[12]. TH Itkonen, *et al.*, "Trade-off between jerk and time headway as an indicator of driving style," PLoS One., vol, 12, no. 10, pp.1-19, 2017.
[13]. Z. Ling, J. Ju-min, L. Chang-an and C.Ying, "The detection system of vehicle driving simulator attitude based on ADXL335," in proc of IEEE 2nd Intern. Conf. Mech. Auto. Con. Eng., pp. 4541-4543, 2011, Hohhot.
[14]. K. W. -H. Tsui, N. C. Cheung and K. C. -W. Yuen, "Novel Modeling and Damping Technique for Hybrid Stepper Motor," IEEE Trans. Indus. Elect., vol. 56, no. 1, pp. 202-211, 2009.
[15]. M. Giriraj and P. Anvesh, "Contolling of servomotors according to pitch and yaw and roll motions of accelerometer," in proc of IEEE Intern. Conf. Ene. Effi. Tech. Sust., pp. 886-889, 2016, India.
[16]. D. R. Gaan, M. Kumar and C. G. Majumder, "Variable rate based microstepping of stepper motor using MATLAB GUI," in proc of IEEE Indian Cont. Conf., pp. 385-390, 2017, India.
[17]. S. Banerjee and G. K. Singh, "A Robust Bio-signal Steganography with Lost-data Recovery Architecture using Deep Learning," IEEE Trans. Instrum. Meas., vol. 71, pp. 1-10, 2022, 4007410, 2022.
[18]. S. Banerjee and G. K. Singh, "Quality Guaranteed ECG Signal Compression using Tunable-Q Wavelet Transform and Möbius Transform based AFD," IEEE Trans. Instrum. Meas., vol. 70, 4008211, 2021.
[19]. S. Banerjee and G. K. Singh, "Monte Carlo Filter based Motion Artifact Removal from Electrocardiogram Signal for Real-time Telecardiology System," IEEE Trans. Instrum. Meas., vol. 70, 4006110, 2021.
[20]. S. Banerjee and G. K. Singh, "A new approach of ECG steganography and prediction using deep learning," Biomed. Sig. Proc. Cont., vol. 64, 102151, 2021.