Control Practices of Permanent Magnet DC Motor Using Arduino as Low Cost Hardware

A.SAIRAMC.V.KARTHIK G.SHARANYA G.SANDHYA RANI ANIL KUMAR.M.S

*EEE&SVCN NELLORE EEE&SVCN NELLORE EEE&SVCN NELLORE EEE&SVCN NELLOR EEE&SVCN NELLORE*

[sairamalavuru@gmail.com](mailto:sairamalavuru@gmail.com) [c.v.karthik137@gmail.com](mailto:c.v.karthik137@gmail.com) [sharanyagoda@gmail.com](mailto:sharanyagoda@gmail.com) [sandhyarani6796@gmail.com](mailto:sandhyarani6796@gmail.com)

**Abstract**—  **this work describes a Simulink lab practices using ARDUINO as low cost hardware. These ARDUINO boards are used to perform a specific task by means of embedded C programming in ARDUINO IDE and also be programmed by using Simulink in MATLAB. By using ARDUINO boards we can implement the modern control techniques like P, PI, and PID. This can be further implemented to electrical appliances to get an effective control in devices. By using developer boards we can have communication with mobile using Bluetooth modules or by WIFI or Ethernet cable to connect to internet also. By applying this technology to machines, we can achieve smooth control on machines. These boards are capable of doing projects. In this project we are going to implement the PID control technique for a permanent magnet DC motor (because it is more efficient, more reliable and easy to control compared to the field wound motor). By using MATLAB software we can get the exact results in this project. It is very useful to students and non technical persons. By using this software we can easily write the program with basic knowledge of C and embedded C languages. In this paper by using PWM technique we control the speed of permanent magnet DC motor.**

**Keywords - ARDUINO controller, Direction, PID controller, Permanent magnet DC motor, PWM technique, Simulink in MATLAB, Speed.etc.,**

# Ⅰ. INTRODUCTION

The main goal of this work is to use low cost hardware to-connect Simulink with a real system for lab practices. Now-a-days, the practical prototypes are connected to a computer through expansive DAQs, when computers or DAQs are updated; it implies a very high cost. In the case internal DAQ’s, connected on an internal bus of the computer. Those difficulties are bigger. New Arduino based DAQ must be the lab prototypes, in order to avoid hardware changes or modifications on it. For the DC motor speed control analog circuits can be used which are complex to design. And digital circuits (microcontrollers) can be used to control motor, but it requires embedded C coding or C-code etc. To overcome these problems we can go through the ARDUINO boards. ARDUINO is a low cost hardware by using the ARDUINO with Simulink model we can obtain the control practices [4].

**Ⅱ. IDENTIFICATION OF PROBLEMS**

The problem of control practices using analog circuits is difficult. For example usage of IC555 timers in control practices. In IC555 timers the availability of RC combination circuits are difficult to achieve a required frequency pulse. So we cannot get the effective control of speed from these circuits and designing of circuits is also difficult. In digital control practices microcontrollers and microprocessors can be used for effective control, but the problem lies in coding of microcontrollers and microprocessors, which required a knowledge of embedded C or python, etc. and a depth knowledge of architecture of microcontrollers.

Thus in this paper we are concerned about speed control of permanent magnet DC motor as main objective [5].

## Ⅲ. AIMS AND OBJECTIVES

In this paper a feasible solution for the speed control of DC motor using armature voltage control technique is developed using Arduino MEGA2560 which is used to solve some of the problems in DC motor, they are as follows

* We can reach required variable speed with better performance.
* Reduce the cost occurred for the controlling as well as operating of permanent magnet DC motor
* For controlling purpose it is easy and consuming less time
* Operation is flexible
* Based on PWM pins we can controlled any number of permanent magnet DC motors simultaneously [5].

**Ⅳ. DEVELOPED MODEL**

Fig:4.1 Kit Developed for Speed and Direction Control

The speed of the permanent magnet DC motor varied by using chopper by varying the duty ratio. This is exactly what armature voltage control technique tries to achieve.

In the above diagram the pin connections are Arduino 5v pin is connected to encoder vcc and ground pins are shorted. 2nd pin of Arduino is connected to encoder channel A, in the same way 3rd pin is connected to channel B, the pulse width modulation pins of Arduino mega 4 and 5 are connected to the motor driver. The motor driver is supplied with 12V.

*A. H-BRIDGE OPERATION*

A single chopper is designed for the armature voltage control of a permanent magnet DC motor as shown in figure.

Fig:4.2 Circuit Diagram of BJT

Here PWM signal from Arduino is given to the base of the BJT. Here BJT acts as an electronic switch. By turning ‘ON’ & ‘OFF’ of the BJT, average voltage across the armature terminal is controlled. The speed of the motor is proportional to the average voltage across its terminals. So, by controlling the average voltage, speed is control. Voltage is controlled by varying the duty ratio ‘δ’ of the converter

i.e. V= (δ)\*Vdc [since δ=t/T]

here δ is controlled by PWM signal produced from microcontroller.

By using four BJTs we can implement the H-bridge. This is used to control the direction also. The operation of H-bridge as follows

(a) (b)

Fig:4.3 H-bridge operation

The H-bridge derives its name from the full-bridge. This circuit is used to control the speed and direction of the motor by means of current flows through the motor and the direction determined by the position of the switches in the H-bridge. Whenever switches A&D are closed the motor will operate in clockwise direction as shown in fig(a), switches B&C are closed the motor will operate in anticlockwise (counterclockwise) direction as shown in fig(b) [15].

**Ⅴ. IMPLEMENTED TECHNOLOGY**

The proposed system consists of PC with Simulink (Simulink support packages for Arduino) used to program Arduino Mega 2560. And a microcontroller board Arduino Mega 2560 powered with ATMega 2560 microcontroller. We have designed a H-bridge with TP41C transistors for speed and direction control of permanent magnet DC motor. We also used opto-coupler 4N35 for optical isolation with Arduino board and to provide PWM signals to the base of the transistors.

For the demo purpose we have a permanent magnet DC motor with Hall-effect sensor based quadrature encoder used to detect speed and direction of motor. Here we used the feedback to calculate only the speed and direction reversal is shown by using voltmeter in DMM (Digital Multi Meter).

*A. ARDUINO HARDWARE*

ARDUINO as a low cost hardware compared to FPGA boards and NI DAQ etc. Arduino board cost less that is less than 1000 rupees, and easily available in market. But with limitted functionalities, which is enough for lab practices.

In this paper we have to use Arduino Mega 2560 microcontroller board the figure as shown in below

Fig:5.1 Arduino Controller Along with ATMEGA-2560

It has 54 digital I/O pins in which pins 15 pins are PWM pins. 16 analog inputs, 4 UARTs (serial ports). A flash memory 256KB, a clock frequency of 16MHZ an USB port to connect to any computer. In this board the operating voltage is 5V and the recommended voltage is 7V to 12V, the voltage limit of this board is 6V to 20V. If the voltage is less than 7V. However the 5V pin may supply less than 5V the board become unstable . If we can use more than 20V the voltage regulator may over heat and damage the board. In this project we can mainly used the PWM pins. In this board the PWM pins are 2 to13 and 44 to 46. These pins provide 8-bit PWM output.

*B. SIMULINK SUPPORT PACKAGES FOR ARDUINO*

As we are well known about the MATLAB is a mathematical software. It is used to development environment with its own high-level language programming. These are used in the field of control practices. Simulink is an useful tool for the system analysis, model and simulation. By using Simulink Arduino can be programmed directly and graphically, by performing the code generator. Simulink has large number of blocks which are used to implement the models which reduces complexity of programming. So that the less skilled person can implement real time projects. This point of view it is very useful working tool for any academic field [6].

In order to use Simulink blocks for Arduino. The free library is available that is “Arduino support from Simulink”, It has several number of blocks these blocks are specifically designed for Arduino boards. By using these blocks we can easily implemented and transferred to the Arduino in a transparent manner for the user, without any mistakes in the conversion between Simulink blocks and Arduino code. The different blocks in Simulink support packages for Arduino as shown in below figure [6].

Fig:5.2 Simulink Support Package for Arduino Hardware

Among the blocks provided for Arduino the most relevant for this work are

**Arduino Analog Input:** By this module the voltage that is being applied by a particular pin can be read. The output accuracy provided in this block is 10-bits [6].

**Arduino PWM:** Through this block of PWM signal is sent to the selected pin. The frequency of the signal pulse is set at 490Hz. The duty cycle may be modified with a precision of 8-bits (values between 0 and 255) [6].

**Arduino Digital Input/Output:** From these two blocks digital signals can be read and written on the selected ports [6].

*C. CONNECTING ARDUINO TO SIMULINK*

To connect Arduino board to Simulink model file , the procedure is as follows

Simulink menu bar Tools Real-time Hrdware

Prepare Hardware

A dialogue box open up, there we need to select the board we are using i.e., Arduino MEGA 2560 and in host-board connection, communication board number to which Arduino is associated is mentioned, then press OK.

To change the board follow the same procedure, Simulink gives support for wide range of Arduino boards.

*D. GENERATION OF PWM SIGNALS*

PWM signal of 490Hz is generated by microcontroller MEGA 2560 here in Simulink, by using Simulink support packages for Arduino we have PWM block, by directly giving data to the PWM in the range of (0-225). Here if considers the data given to it as the duty ratio(δ)

Figure: 5.3s

It shows the PWM signal generated Arduino, when constant of (25/100)\*255 i.e., at δ=25%

Figure: 5.4

It shows the PWM signal generated Arduino, when constant of (50/100)\*255 i.e., at δ=50%

Figure:5.5

It shows the PWM signal generated Arduino, when constant of (75/100)\*255 i.e., at δ=75%

Figure: 5.6

It shows the PWM signal generated Arduino, when constant of (100/100)\*255 i.e., at δ=100%

So by simply giving a signal of value ranges (0 to 255). 5V PWM signal related to the given value is generated by Arduino.

**Ⅵ. SIMULINK MODEL EXPLANATION**

Constant block is enough to power the PWM block of Arduino. A dashboard knob is connected to the constant source block to adjust the value of it easily during run time.

*A.ENCODER BLOCK*

Encoder block is designed to read the speed of motor by using the encoder’s Hall-effect sensor output. Which is a square pulse. Which has the frequency of the square wave is proportional to the speed of the motor.

Fig: 6.1 Encoder Block

Hall-effect sensor’s data is taken as input from digital input block ‘pin 2’. A positive edge triggered system is used along with clock and delay block, to find the speed of the motor, Kalman filter is used to reduce the distortion in reading the speed of motor.

*B.PID BLOCK*

In closed loop control, the output of encoder block is compared with the set speed and error is given PID controller block.

Here P-controller is designed by simply using a gain block with a value of Kp

I-controller is designed by simply using a gain block with a value

D-controller is designed by simply using a gain block with a value of K

Fig:6.2 PID Controller Block

At the end all the output’s of P,I,D are added using a summing block to get output. The output value is directly given to the PWM block of Arduino. In between the PID&PWM blocks, logic controlled switches are used to control the direction of the motor & ON/OFF of motor related to the switches in the dashboard. In the Simulink model converter blocks are used to convert the data type of the input signal to make it suitable for the different blocks in Simulink.

*C.KALMAN FILTER*

A discrete filter is used to reduce the distortion caused by the sensor noise, which estimates the continues output’s from the discrete input signal. Kalman filter with single eye & resolution of 0.00005 played great role in reducing noise distortion in speed calculation from discrete signal.

Fig:6.3 Speed of the motor with (blue)& without filter(red)

*D.CIRCUIT DIAGRAM*

The complete circuit diagram of speed control of motor as shown in below figure.

Fig:6.4 Circuit Diagram

Here opto-couplers 4N35 are used for electrical isolation of Arduino with H-Bridge converter, it helps in protection of Arduino board.

**Ⅶ. RESULTS**

*A.OPEN LOOP CONTROL*

Open loop control can be done in two ways

1. Embedded control
2. External mode control

**Embedded control:** In embedded control, a potentiometer is connected to Arduino to set speed value of the motor. We need to dump the code to the board, we can use it stand alone.

**External mode control:** Live-time control, Simulink mode is set to external mode then run the simulation. In this Arduino is used as Host for PC, Simulink acts as master control. If we vary the parameters in the Simulink block, the changes are seen in the real-time system. In our experiment ‘δ’ is varied using knob from 20%,40%,60%,80%,100%. The speed of the motor is following the duty ratio.

Fig:7.1 Response of open loop control

*B.CLOSED LOOP CONTROL*

Closed loop control is implemented by taking the feedback and comparing with the reference value and error is calculated. Then error is fed to the controllers to reduce the errors. Results of P-controller with varying input.

Fig:7.2 Response of closed loop control with P-controller

Here we have steady state error, to reduce the steady state error we are implemented PI-controller with

Kp=2.5 , KI =0.15. for a variable input

Fig:7.3 Response of closed loop with PI-controller

Response of system with PI-controller with fixed input finally tuned a

Fig:7.4 Response of PI-controller with fixed input

To reducethe oscillationsD-controller is used. The speed response of the system with PID-controller with fixed input.

Fig:7.5 Response of closed loop with PID-Controller

**Ⅷ. FUTURE SCOPE**

In this paper we have implemented the developing system on a permanent magnet DC motor [5]

* In future Fuzzy based control also be implemented by using Arduino
* We can use Arduino for practical implementation of rectifiers, converters (choppers), inverters etc. using Simulink support packages.
* Since Arduino Simulink support package has support for serial communication, I2C, SPI communication blocks, many can be interfaced to it and even we connect Bluetooth, RF, ZIG-BEE modules to Arduino. Which are to control the stand alone systems wireless.

**Ⅸ. CONCLUSION**

It can be concluded that Aduino is a low cost hardware. It is very useful to students in the control practices. Here we can control the speed of permanent magnet DC motor by controlling pulses giving from the Arduino. So by using Simulink we can design controllers very easily.

**Ⅹ.REFERENCES**

[1] Arduino (2013a). <http://arduino.cc/es/Main/Software>

Last view: March,2013.

[2] Arduino. (2013b). <http://arduino.cc>

[3] Arduino. (2013c). <http://arduino.cc/es/Tutorial/Home>

Page. Last view: March,2013.

[4] R.Barber, M.Horra, J.Crespo: Control Practices using

Simulink with Arduino as Low Cost Hardware.

Preprints of the 10th IFAC Symposium Advances in

Control Education. The International Federation of

Automatic Control, August 28-30,2013 University of

Sheffield, UK.

[5] Y.V. Niranjan Kumar, P.Hima Bindu, A.Divya Sneha,

A.Sravani: A Novel Implementation of Phase Control

Technique for Speed Control of Induction Motor using

ARDUINO. ISSN 2250-2459, ISO 9001:2008 Certified

Journal, Volume 3, Issue 4, April 2013.

[6] A.Soriano, L.Marin, M.Valles, A.Valera, P.Albertos: Low

Cost Platform for Automatic Control Education Based

on Open Hardware. Preprints of The 19th World

Congress. The International Federation of Automatic

Control. Cape Town, South Africa. August24-29,2014.

[7] Khan Masoom Raza, Mohd.Kamil, Pushpendra Kumar:

Speed Control of DC Motor by using PWM.

International Journal of Advanced Research in

Computer and Commjnication Engineerig. ISSN

(online) 2278-1021, ISSN(print) 2319 5940, Vol 5,Issue4,

April 2016.

[8] Patrick Olabisi, Israel Megbowon and Isaiah Akintunde:

Programming The PIC16F877 A Microcontroller for

Speed Control of DC Motor. IOSR Journal of Electrical

And Electronics Engineering (IOSR-JEEE) e-ISSN: 2278

-1676, p-ISSN: 2320-3331, Volume 10, Issue 2 Ver.III

(March- April. 2015), PP 79-85.

[9] I G.A.P. Raka Agung, S.Huda, I W.Arta Wijaya: Speed

Control for DC Motor with Pulse Width Modulation

(PWM) Method using Infrared Remote Control Based

on ATmega 16 Microcontroller. ISBN:978-1-4799-6127-6

Bali, 5-7 November 2014, ICSGTEIS 2014.

[10] Hsin-Chuan Chen: An H-Bridge Driver using Gate

Bias for DC Motor Control. 2013 IEEE 17th International

Symposium on Consumer Electronics (ISCE).

Guided By: Mr.T.Srikanth,M.Tech, Assistant Professor,Department of ELECTRICAL AND ELECTRONICS ENGINEERING,SREE VENKATESWARA COLLEGE OF ENGINEERING,(AFFILIATED TO JNTU, ANANTAPUR)

NELLORE

T