Proceedings of the World Congress on Engineering 2014 Vol I,
WCE 2014, July 2 - 4, 2014, London, U.K.

Motor Speed Control Using FPGA

Ahmed Telba *Member, IAENG, Member, IEEE*

***Abstract*- DC motor had been used in many applications. In some applications the control of DC Motor speed is a deal breaker. These applications require a very tight speed controlling to avoid serious problems. There are various ways to control the speed of motor. The process of developing any solution to a certain problem should go through three steps. The first step is to simulate the problem and try to find the solution. The second one is to verify that your solution is really working before you try it on real-time problems. The last step is to validate your solution on real-time measurements. In this paper we studied the problem, analyzed it, and we found the solution and did simulation to check its outcomes. Our goals in this paper are to verify our solution and implement it using Field-Programmable Gate Arrays (FPGAs). FPGAs must be programmed using Hardware Description Language (HDL). Xilinx had been used to control speed the simulation done using real time measurements using FPGA for step response of the system using MATLAB/SIMUKLINK and PSIM.**

***Index Terms*—DC Motor, speed control, FPGA, modeling and simulation**

I. Introduction

Motors is very popular devices that can be used in every house, computers and cars. The principle of controlling AC motor is not different from AC motors to DC motors. DC motors are seldom used in ordinary applications because all electric supply companies furnish alternating current. However, for special applications such as in steel mills, mines and electric trains, it is advantageous to convert alternating current into direct current in order to use DC motors. The reason is that the speed/torque characteristics of dc motors are much more superior to that of AC motors. Therefore, it is not surprising to note that for industrial drives, DC motors are as popular as three-phase induction motors [1].

In this paper using and implement our solution using Field-Programmable Gate Arrays (FPGAs), where FPGAs is an integrated circuit designed to be configured by a customer or a designer using .Xilinx and Altera are the current FPGA market leaders and long-time industry rivals. Together, they control over 80 percent of the market. Both Xilinx and Altera provide free Windows and Linux design software which provides limited set of devices. Other competitors include Lattice Semiconductor, Actel, Silicon Blue Technologies , Achronix, and QuickLogic [2_3].

Manuscript received March 20, 2014 revised March 28, 2014. Ahmed Telba is working as Researcher in Electrical Engineering Department, College of Engineering, King Saud University, KSA, (corresponding author to provide phone: 966-14678800; fax: 966-14676757; e-mail: atelba@ ksu.edu.sa).

II. Theoretical Analysis and Design

In this section discussing System Modeling for DC Motor Speed and Physical setup .A common actuator in control systems is the DC motor. It directly provides rotary motion and, coupled with wheels or drums and cables, can provide translational motion. The electric equivalent circuit of the armature and the free-body diagram of the rotor are shown in figure.1 [4]:

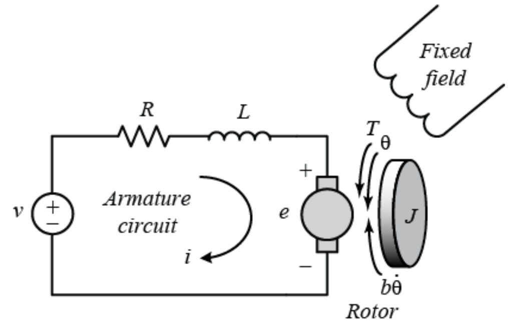


Fig 1: The equivalent circuit and free-body of DC motor.[3]

The input of the system is the voltage source (V) applied to the motor's armature and will be the output is the rotational speed of the shaft(θ/dt). The rotor and shaft are assumed to be rigid. There is a viscous friction model, which is the friction torque. The friction torque is proportional to shaft angular velocity. The physical parameters involved are: (J) Moment of inertia of the rotor,(b) Motor viscous friction constant,(Ke) Electromotive force constant,(Kt) Motor torque constant,(R) Electric resistance,(L) Electric inductance,(T) Torque,(i) Armature current,(e) Back electromotive force, and ($\dot{\theta}$)Angular velocity.

III. System Equations

The torque generated in DC motor is proportional to the armature current and the strength of the magnetic field. Choose that the magnetic field is constant and, therefore, that the motor torque is proportional to only the armature current i by a constant factor K_t as shown in the equation (1). This is referred to as an armature-controlled motor[5] .

$$T = k_t i \tag{1}$$

The back emf, e, is proportional to the angular velocity of the shaft by a constant factor K_e.

$$e = k_e \theta^{\cdot} \tag{2}$$

ISBN: 978-988-19252-7-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

In Standard International (SI) units, the motor torque and back emf constants are equal, that is, $K_t = K_e$; therefore, we will use K to represent both the motor torque constant and the back emf constant. From figure.1, we can derive the following governing equations (3) and (4) based on Newton's 2nd law and Kirchhoff's voltage law.

$$J\ddot{\theta} + b\dot{\theta} = Ki \tag{3}$$

$$L\frac{di}{dt} + Ri = V - K\dot{\theta} \tag{4}$$

IV. TRANSFER FUNCTION OF THE SYSTEM

Applying the Laplace transform to the above modeling equations can be expressed in terms of the Laplace variable S domain

$$s(Js + b)\theta(s) = KI(s) \tag{5}$$

$$(Ls + R)I(s) = V(s) - Ks\theta(s) \tag{6}$$

We get at the following open-loop transfer function by eliminating I(s) between the two above equations, where the rotational speed is considered the output and the armature voltage is considered the input.

$$P(s) = \frac{\dot{\theta}}{V(s)} = \frac{K}{(Js+b)(Ls+R)+K^2} \qquad \left[\frac{rad/sec}{V}\right] \tag{7}$$

V. STATE SPACE MODELING

In state-space form, the governing equations above can be expressed by choosing the rotational speed and electric current as the state variables.

$$\frac{d}{dt}\begin{bmatrix}\dot{\theta}\\ i\end{bmatrix} = \begin{bmatrix}-\frac{b}{J} & \frac{K}{J}\\ -\frac{K}{L} & -\frac{R}{L}\end{bmatrix}\begin{bmatrix}\dot{\theta}\\ i\end{bmatrix} + \begin{bmatrix}0\\ \frac{1}{L}\end{bmatrix}V \tag{8}$$

$$y = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}\dot{\theta}\\ i\end{bmatrix} \tag{9}$$

VI. SYSTEM TRANSFER FUNCTION

Applying the Laplace transform and the above modeling equations can be expressed in terms of the Laplace variable S.

Pulse-Width Modulation:

Pulse-width modulation (PWM) of a signal or power source involves the modulation of its duty cycle, to either convey information over a communications channel or control the amount of power sent to a load. Pulse-width modulation uses a square wave whose pulse width is modulated resulting in the variation of the average value of the waveform. If we consider a square waveform f(t) with a low value y_{min}, a high value y_{max} and a duty cycle D, the average value of the waveform is given by:

$$\bar{y} = \frac{1}{T}\int_0^T f(t)dt \tag{10}$$

The output of PWM pulse generator for a certain D is shown in Figure.2[4]

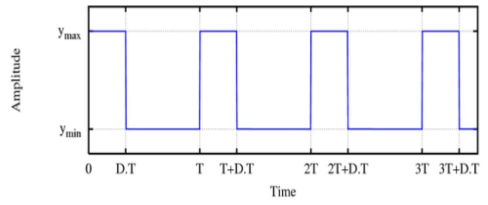


Fig.2: Basic PWM

As f(t) is a square wave, its value is y_{max} for $0 < t < D.T$ and y_{min} for $D.T < t < T$.

VII. APPLICABLE DESIGN

We have three main blocks, which are FPGA, Driver Circuit, and DC Motor. The desired duty cycle (speed) and direction of DC Motor is taken from the user through FPGA. FPGA generate PWM signal and direction according to the given value and sends it to the driver circuit. Driver circuit drives the current to meets DC Motor need for current. After the voltage and current applied to DC Motor, the Quadrature Encoder encodes the speed and the direction of the motor and send it to FPGA as signal A and signal B. Finally, FPGA decode these two signal to show the speed of the motor. Figure.3 shows the signal flow chart of the system

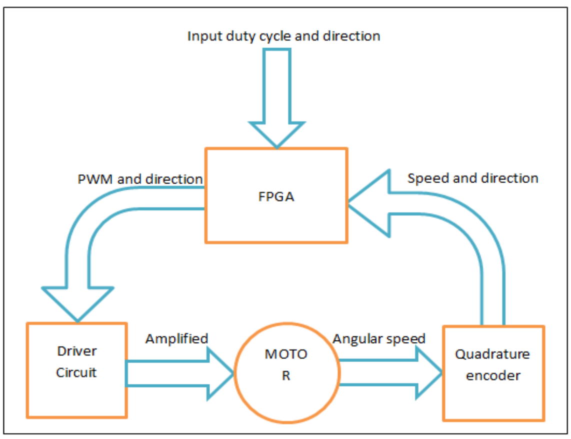


Fig.3: Signal flow chart of the system

VIII. CONTROLLER AND DECODER UNIT

There are two different approaches to implement our model. The first one is using FPGA as a controller to the plant. The second one is to use Microcontroller instead of FPGA. FPGAs are logic multithreading real-time processing chips. It doesn't have the built-in peripherals feature. Instead, peripherals can be logically programmed. In

ISBN: 978-988-19252-7-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

addition, FPGAs are made of raw of logic gates, which means you can make them do anything and that's the power of FPGA. On the other hand, Microcontroller has the feature of using of peripherals like UART, SPI, PWM, Timers, and so on, and that's make it easier to interface with other devices. We chose FPGA for the following reasons:

1. FPGAs are concurrent. You can take sequential functionality like adding soft processor core. While the microcontroller as always sequential. This makes FPGAs better suited for real-time applications such as executing DSP algorithms.

2. FPGAs are an increasingly attractive solution path for demanding applications due to their ability to handle massively parallel processing. FPGAs tend to operate at relatively modest clock rates measured in a few hundreds of megahertz, but they can perform sometimes tens of thousands of calculations per clock cycle while operating in the low “tens of watts” range of power. Compared to FPGAs, microprocessors that operate in the same power range have significantly lower processing functionality. Typically, a similarly power rated microprocessor may run at 1-2 GHz clock rate, or roughly 4 or 5 times as fast as an FPGA, but it will be much more limited in how many operations it can perform per clock cycle, with a maximum typically in the range of four or eight operations per clock. This means that FPGAs can provide 50 to 100 times the performance per watt of power consumed in a microprocessor.

3. FPGA are flexible, you can add subtract the functionality as required. This can not be done in microcontroller.

4. FPGAs are liked in military applications. There are two main reasons of that. The first is that FPGAs are hard-wired and the random attack of alpha rays can not destroy/corrupt the memory areas hence collapse the device functionality. The second reason is that the life time of FPGA based development is longer. It can be adopted for advanced chip is required. Microcontrollers change too often and there is lots re-work required to do in order to keep pace with changing technology. This is necessary to save the design from being obsolete.

5. While FPGAs may be more expensive than a single MCU, their functionality, such as embedded DSP and memory blocks and a flexible I/O ring, may offset the cost of multiple devices[5].

IX. DRIVER CIRCUIT

The two basic principles of driving DC Motor using PWM are to separate the PWM circuit from the motor circuit for safety, and amplify the signal coming from PWM. There are different arrangements that could realize these principles. We tried two circuits.

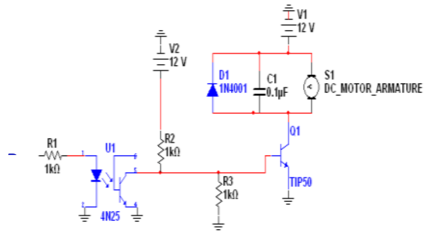


Fig 4: DC Motor Driver Circuit

Bridge Circuit :

Which is basically an electronic circuit that enables a voltage to be applied across a load in either direction. These circuits are often used in robotics and other applications to allow DC motors to run forwards and backwards. It would be useful to be able to run a motor in either direction. Reverse the DC power connections on such a motor and its shaft will rotate in the opposite direction.

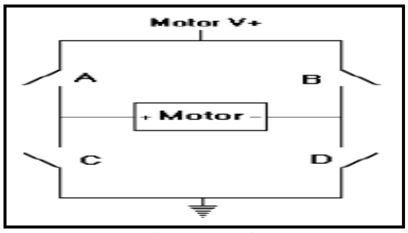
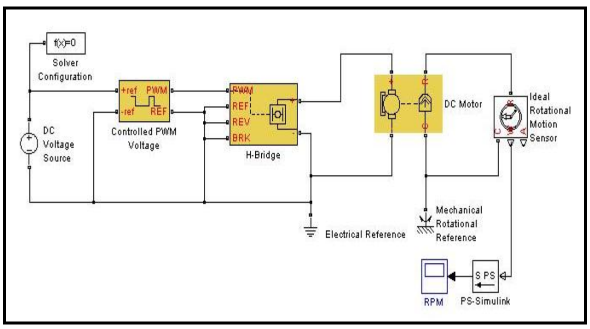


Fig.5: Basic principle of H-bridge for DC motor control [6]

X. SYSTEM SIMULATION RESULTS

We simulated our design in two different software. The first one is MATLAB/SIMUKLINK. Which is one of the most popular electrical engineering program. The other one is PSIM. Which is a common software to do power system and electrical circuits simulation. PSIM is a simulation software specifically designed for power electronics and motor drives. It provides a powerful simulation environment for power electronics, analog and digital control, magnetics, motor drives, and dynamic system studies

***Circuits of Simulation* :**
The simulation of the system using MATLAB/SIMULINK using the circuit in Figure.6 and for PSIM[10] using the circuit in Figure7 and check the system stability and step response as shown in fig 8.



Figu.6: MATLAB/SIMUKLINK Block Diagram for the system Model.

ISBN: 978-988-19252-7-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

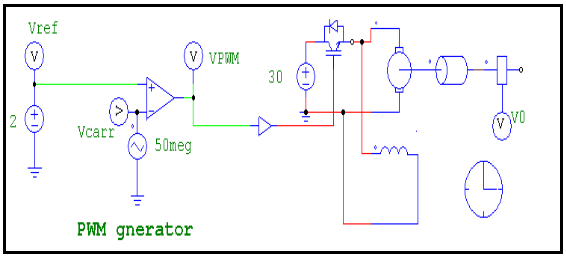


Fig.7: Circuit for DC motor system in PSIM[10]

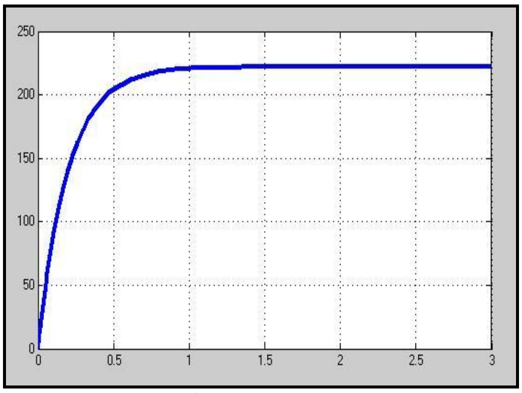


Fig.8: step response of simulated system

Figure.9 shows the experimental setup system and block diagram which contains an FPGA, Driver Circuit, and Motor. The user gives Direction and Duty Cycle to FPGA by push-buttons. After that, FPGA generate PWM and Direction signals depending on what the user specified. These PWM signal goes to Driver Circuit to amplify the power to meet the motor requirements. On the other hand, the Direction signal sets the polarity of the driving voltage as user need. once the motor run, Signal A and B will be generated to encode the Direction and Speed of the motor. Finally, FPGA takes these two signals to decoder them and show the user the Speed and Direction.

Xilinx Spartan-3 was available and sufficient to our work. We used VHDL to program Spartan-3 via ISE Design Suite 14.3 from Xilinx Inc. We developed the block diagram in Figure.10 which consists of Debouncing Circuit, Quadrature Decoder, PWM Generator, and Binary to BCD Converter. The input/output pin assignment is shown at last of Appendix.

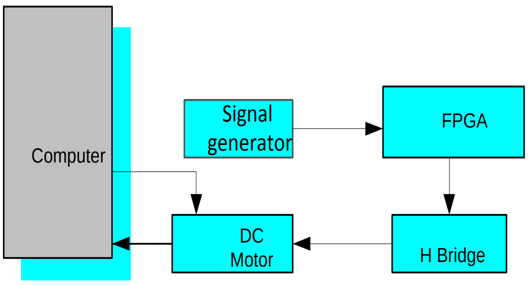


Fig.9. Experimental setup

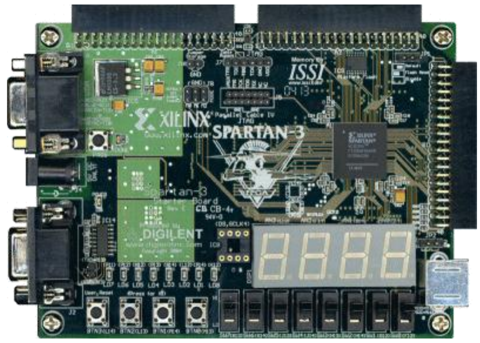


Fig 10 Real time measurements in FPGA

XII. Conclusion

This paper describes a reliable dynamic model to simulate Dc motor controller using FPGA. The experimental results validate the dynamic model. The real time measurements show the relation between the acceleration and both output speed of the motor. The model had been implemented in lab using FPGA and tools of VHDL procedure for identifying model parameters is applied using simple measurements and standard laboratory equipment.

References

[1] V. K. Mehta, Principles Of Electrical Machines, 1st Edition, S. Chand & Company LTD.,2002

[2] Norman .S .N., Control System Engineering 6th Edition, Wiley & Sons, Inc. 2011

[3] Do-Hyun Jang; Jong-Soo Won; Gyu-Ha Choe "Asymmetrical PWM method for ac chopper with improved input power factor" , IEEE, PESC Conference 1991, Cambridge, Massachusetts, USA, pp. 838–845, 1991.

[4] Lin Min Yu Hongxiang, and Ji Yanchao " An advanced harmonic elimination PWM technique for ac choppers" IEEE conf. proc., PESC'04, Aachen, Germany, 2004, pp.161-165.

[5] Ju-Sung Kang, N. A. Ahmed, Kwang-Joo Choi; Hyun Woo Lee, Mutsuo Nakaoka,"Pulse modulated ac voltage regulator using bidirectional active switches with different control strategies" IEEE conf. proc., ICEMS 2005, Nanjing, China, vol.2, Sept. 2005 pp.1107 – 1111.

[6] B. Saracoglu, M. Kale, E. Ozdemir, " A novel technique for optimal efficiency control of induction motor fed by PWM IGBT ac chopper" IEEE, Conf. Proc., PESC 04, Aachen, Germany, vol.5, June 2004, pp.3353 – 3358.

[7] K. Veszpremi, M. Hunyar, "New application fields of the PWM IGBT ac chopper " IEE Conf. Power Electronics and Variable Speed Drives, London, UK, Sept. 2000 pp. 46–51.

[8] A-R. A.M. Makky, N. A. Ahmed, E. H. El- Zohri, "Supply power factor improvement with single-phase ac voltage converter" IEEE, Conf. Proc., IEMDC'03, Madison, Wisconsin USA, vol. 2, June 2003, pp. 863 –868.

[9] Hui Li Da Zhang, and Emmanuel G. Collins " Digital Anti-Windup PI Controllers for Variable-Speed Motor Drives Using FPGA and Stochastic Theory" IEEE Trans.on PE, vol 21, no 5, pp.1496-1501, Sept.2006.

[10] PSIM6.1, available: www.powersimtech.com.