Speed Control for Brushless DC Motors using PID Algorithm

Whitepaper

Contents

Abstract ... 3

Keywords ... 3

Introduction ... 4

Three Phase Motor Commutation ... 4

Methods for Speed Control of Motor ... 6

1. Open-loop Method ... 6
2. Closed-Loop Method ... 6
    A. PID Algorithm ... 7

Implementation ... 9

1. Hall Sensor Decoder ... 10
2. Speed Computation ... 10
3. PID Controller ... 10
4. PWM Generator ... 10
5. Implementation Results ... 11
    A. Simulation Results ... 11
    B. Chipscope Pro Analyser Results ... 12
    C. Comparison with mathematical model ... 12

Conclusion ... 13

References ... 13

About Authors ... 14

Abstract

Field Programmable Gate Arrays (FPGA) has gained attraction in the field of Industrial automation and is becoming more popular in motor control applications due to its robustness and customizability. The use of FPGA provides outstanding hardware design flexibility. This further enables implementation of a fully customizable peripheral set and also helps to meet the increasing performance requirements. The paper covers the use of Xilinx Kintex 7 FPGA based implementation of close loop control system. The close loop system controls the speed of Three-Phase Brushless Direct Current (BLDC) motor, by using feedback from three hall sensors. Xilinx Kintex 7 FPGA based speed controller of brushless DC motor system is designed and implemented using PID algorithm. The design is implemented using Verilog Hardware Description Language (HDL). As FPGA supports reconfigurable computing and provides a facility of on-chip parallelism, good operational performance was achieved.

Keywords

BLDC, PID, PWM, RPM, FPGA

Introduction

Nowadays, brushless DC motors are widely used in various applications, right from motor vehicle, and industrial application to aircrafts for the control application. The primary reason of increasing usage of brushless DC motor can be attributed to good weight/size to power ratio, excellent acceleration performance, little or no maintenance and less acoustic and electrical noise than Brushed DC motors. This has led to significant researches for the speed control on the BLDC motors.

In BLDC motor, the commutation of a BLDC motor is controlled electronically. To rotate the BLDC motor the stator windings should be energized in a sequence. It is important to know the rotor position in order to understand which winding will be energized following the energizing sequence. Rotor position is sensed using Hall-effect sensors embedded into the stator. Mostly BLDC motors have three Hall sensors inside the stator on the non-driving end of the motor. Whenever the rotor magnetic poles pass near the Hall sensors they give a high or low signal indicating the N or S pole is passing near the sensors. Based on the combination of these three Hall sensor signals, the exact sequence of commutation can be determined.

The remainder of the paper is arranged as follows. Section 2 describes the three phase commutation. Section 3 explains different methods for speed control of BDLC motor and provides detailed explanation of PID algorithm. The implementation of FPGA based speed control of motor using three feedback hall sensors is described in section 4. While Section 5 shows the simulation and validation results of the implemented design. Finally, conclusions are drawn in Section 6 and references are listed in Section 7.

Three Phase Motor Commutation

As compared to Brushed motors, the commutator and brushes in Brushless motors are replaced by six Transistor/MOSFETs/IGBTs to form three-phase Bridge as shown in figure1. Three phase windings either in Star/Delta fashion is connected in motor with each winding 120 degrees apart. Windings are energized by providing a constant voltage level at the transistors on one branch of bridge (as indicated by signals PWM A, PWM B& PWM C); while on the other branch the PWM pulses are applied (as indicated by signals PWM A, PWM B & PWM C).

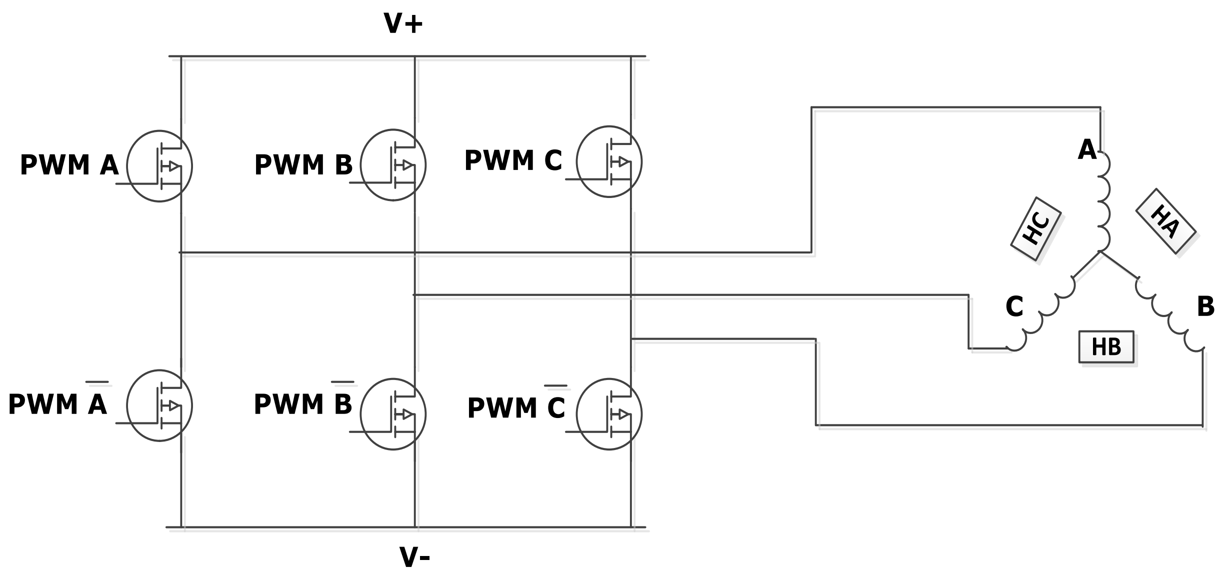

Figure 1 : Bridge Circuit of Motor Driver

At any time, one winding is connected to positive terminal through upper transistor; one winding is connected to negative terminal through lower transistor and third winding connected to ground terminal. The phase sequence at each terminal will be as per figure 2. Only two phases are active at a time. Hall sensors are used to detect the position of rotor; their outputs will appear at exactly 120 degree apart from each other as per figure 2.

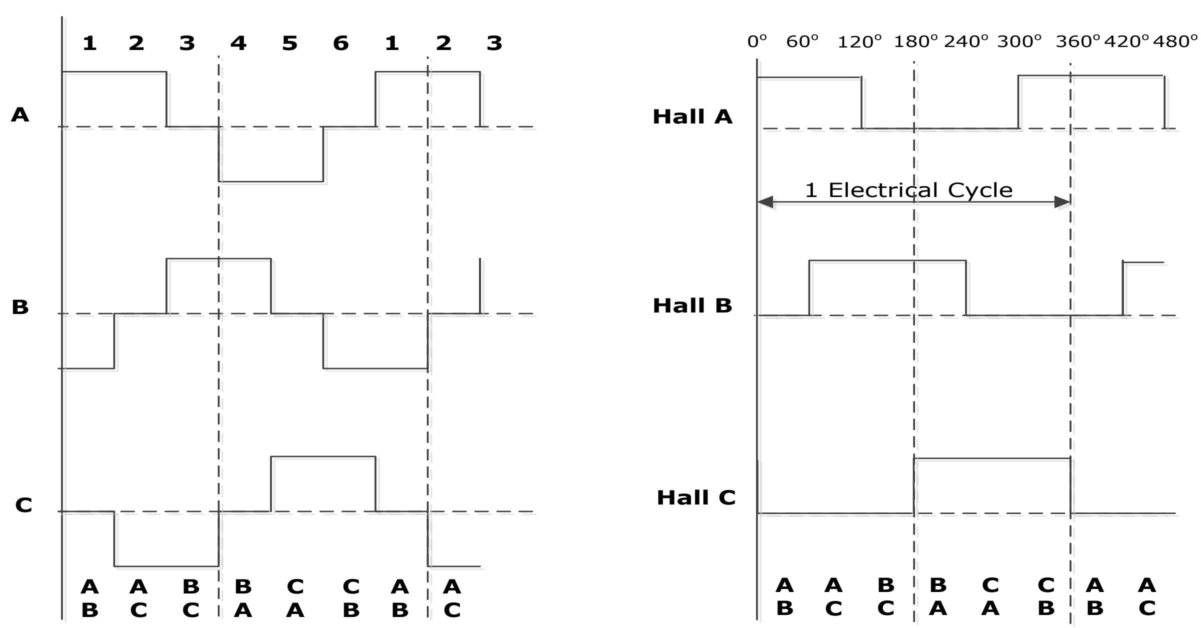

Figure 2 : Three Phase Commutation and Hall Sensor Outputs

Methods for Speed Control of Motor

For speed control of BLDC motor using sensor based method, it is required for the controller to know the rotor position for electronic commutation.

Sensor based commutation method can be implemented in two ways:

1. Open loop
2. Closed loop

1. Open-loop Method

In this method, the actual motor speed is not tracked and hence there is no feed-back mechanism. Only reference speed is used to control the motor speed by updating PWM duty cycle. The advantages of this method are simplified control algorithm and lower cost of implementation. The disadvantage is that accuracy cannot be maintained as there is no way to track the actual motor speed.

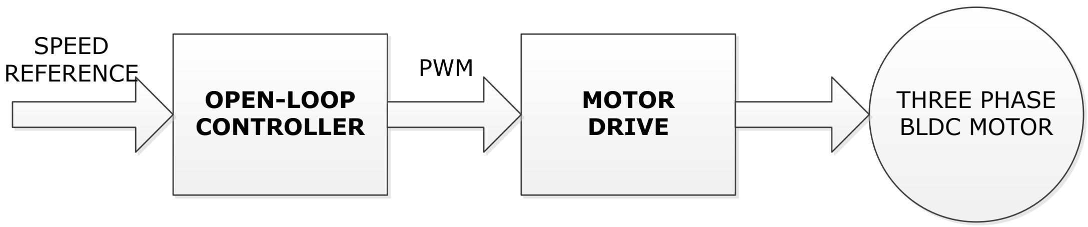

Figure 3: Open Loop Control Method

2. Closed-Loop Method

In this method, the actual motor speed is tracked and hence the feed-back mechanism is critical. Here both the reference speed and the actual speed are used to control the motor speed by updating the PWM duty cycle. The main advantages of this method are the system stability; minimal disturbances compared to open-loop control and reduced sensitivity for dynamic load variations.

P controller, PI controller, PD controller and PID controller are the controller types which can be used in feedback or feedforward systems based on the system requirements. PID controller is widely implemented in closed-loop type of feedback control system for speed control of BLDC motors.

P and PD controller is directly proportional to incoming error; hence a little change in error can cause system instability.

PI controller is an accurate and provides good system stability i.e. less steady state error response. But the integral factor in controller takes more iteration to reduce error to zero.

PID controller is the most reliable, accurate and provides better system stability i.e. less steady state error response as well as it helps to eliminate incoming error to zero with minimum iterations.

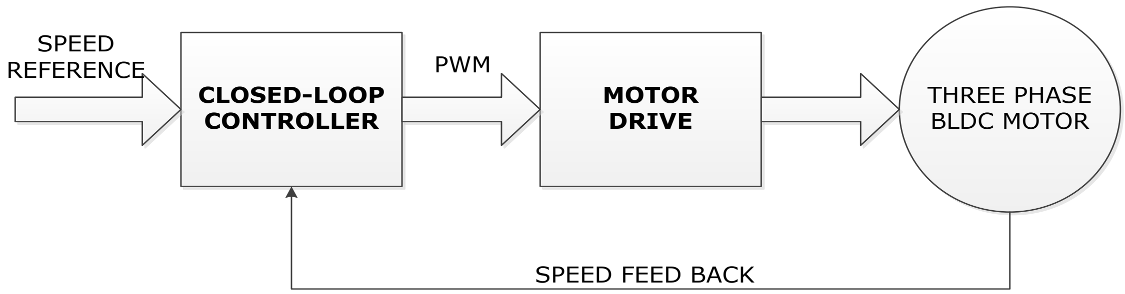

Figure 4 : Closed Loop Control Method

A. PID Algorithm

PID controller is the most widely used in closed-loop control system. The algorithm works on the error generated from the difference between the reference speed and the actual speed. PID parameters Proportional gain Kp, Integral Gain Ki, and Derivative Gain Kd affects system's overall performance. Hence choosing right parameters for a system is a difficult process and can be done by using several tuning methods which includes manual tuning, Ziegler-Nicholas tuning and Cohen-coon tuning. Normally a mathematical model of the system is designed along with PID controller and the system performance is observed with applied set of values of PID parameters to finalize the best suited values.

Following are the effects of PID parameters (Kp, Ki, and Kd)

- System Rise time will be reduced by Kp, it provides faster response in variable load condition
- Steady state error will be reduced by Ki , hence the motor speed is pushed near to reference speed
- Settling time and overshoot will be reduced by Kd, hence provides faster response.

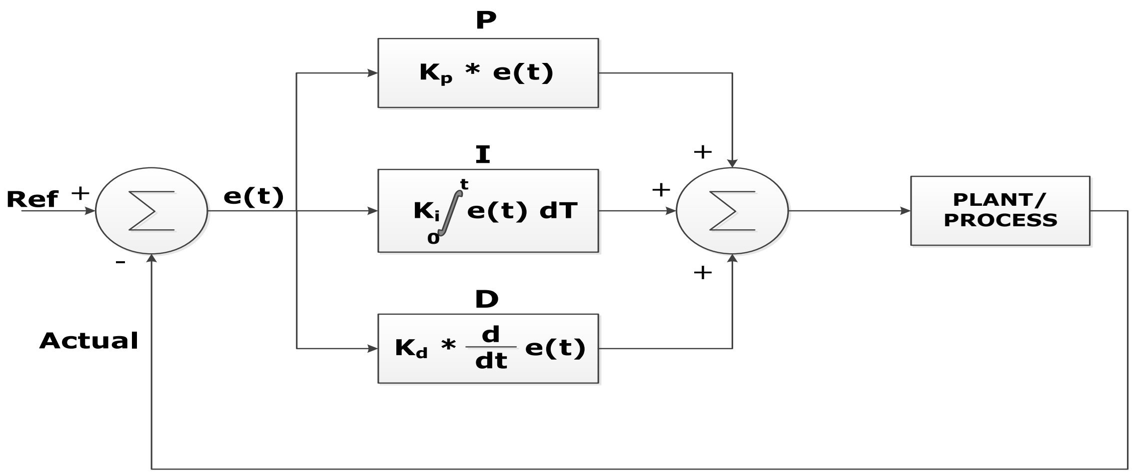

Figure 5 : Block Diagram of a System with PID Controller and Feedback loop

PID controller works on following equation:

$$u(t) = K_p * e(t) + K_i \int_0^t e(t)\, dT + K_d * \frac{d}{dt} e(t) \quad \text{.... (1)}$$

For digital implementation of above continuous form equation, discrete time transformation is required. The discrete form is achieved by first applying Laplace transform on each individual term and subsequently applying z transform on each term. The transformation for each individual term is as shown in table 1.

Table 1: PID Transformations

Term	**Time Domain Equation**	**Laplace Transformation**	**Z Transformation (Discrete time transformation)**
Proportional (P)	$u(t) = K_p * e(t)$	$u(s) = K_p * e(s)$	$u(z) = K_p * e(z)$
Integral (I)	$u(t) = K_i \int_0^t e(t)\, dT$	$u(s) = \frac{K_i}{s} * e(s)$	$u(z) = \frac{K_i}{(1-z^{-1})} * e(z)$
Derivative (D)	$u(t) = K_d * \frac{d}{dt} e(t)$	$u(s) = K_d * s * e(s)$	$u(z) = K_d * (1-z^{-1}) * e(z)$

By replacing values of z transformations from table 1 to equation (1), we get

$$u(z) = \left[K_p + \frac{K_i}{(1-z^{-1})} + K_d * (1-z^{-1}) \right] * e(z) \ldots. (2)$$

Further simplifying it,

$$u(z) = \left[\frac{(K_p + K_i + K_d) + (-K_p - 2K_d) * z^{-1} + K_d * z^{-2}}{(1-z^{-1})} \right] * e(z) \ldots. (3)$$

By rearranging equation (3) we have,

$$\mathbf{u(z) - z^{-1} u(z) = \left[(K_p + K_i + K_d) + (-K_p - 2K_d) * z^{-1} + K_d * z^{-2} \right] * e(z) \ldots. (4)}$$

By converting equation (4) in to difference equation, we get following equation, which can be easily implemented in digital systems by using basic digital blocks like adders and multipliers.

$$u(k) = u(k-1)+(K_p + K_i + K_d) * e (k)-(K_p + 2K_d) * e (k-1)+ K_d * e(k-2)\ldots. (5)$$

Implementation

Closed loop control method using PID algorithm is implemented on Xilinx Kintex 7 Field Programmable Gate Array (FPGA). The top level block diagram for implementation is as shown in figure 6.

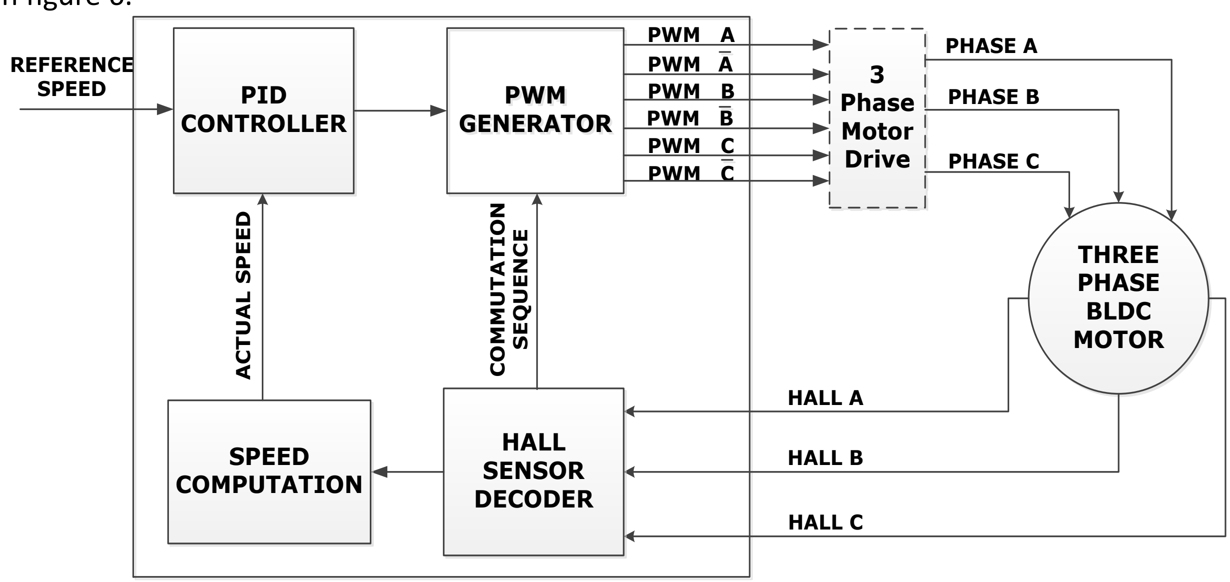

Figure 6 : Implementation Block Diagram

1. Hall Sensor Decoder

Different combination of hall inputs will be available based on rotor position. Hall input’s combinations will repeat in successive manner for an electrical cycle as shown in figure 2. By decoding the sequence of hall sensors, we will get information about change in electrical cycle as well as the commutation sequence.

2. Speed Computation

Actual speed of motor is calculated based on electrical cycles. An electrical cycle has direct relation with RPM (Revolution Per Minute) of motor, as an example for single pole motor, one electrical cycle represent one revolution (1 RPM) of motor.

3. PID Controller

As described in section 3.2.1, PID algorithm is simplified to Equation (5) and is digitally implemented with use of multipliers and adders. PID processes the incoming error per electrical cycle. For implementation of this motor control application, the tuning of the PID parameter values (Kp, Ki and Kd) are derived from a mathematical model.

4. PWM Generator

The speed of motor is controlled by varying the motor torque. Motor Torque can be varied by changing average voltage across the motor windings. For varying average voltage across the windings the PWM method is used where the duty cycle of wave is varied according to require speed. The variation of applied PWM wave is based on hall sensor decoder output to recognize which phase is to be energized. Here, on one branch; level signal is applied (PWM $\bar{A}$, PWM B & PWM C) and on another branch PWM signal is applied (branch PWM A, PWM B and PWM C) which is shown in figure 7.

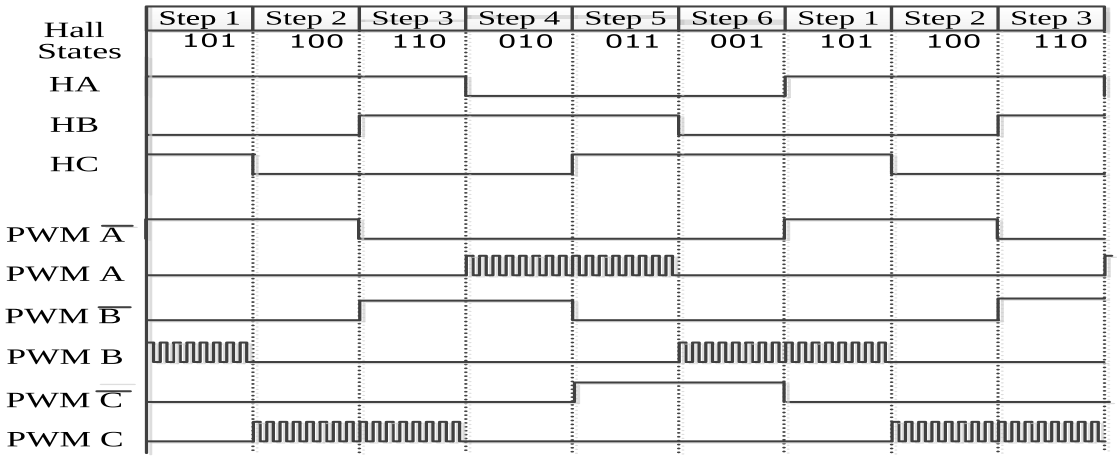

Figure 7 : Commutation sequence based on Hall Sensor

5. Implementation Results

A. Simulation Results

Figure 8 represents simulated waveform for FPGA control logic integrated in system. Ha, Hb and Hc the hall sensor inputs and reference speed are the test inputs provided through test bench. PID continuously processes the input error at every electrical cycle and based on PID output PWM waves are generated for the three phase windings.

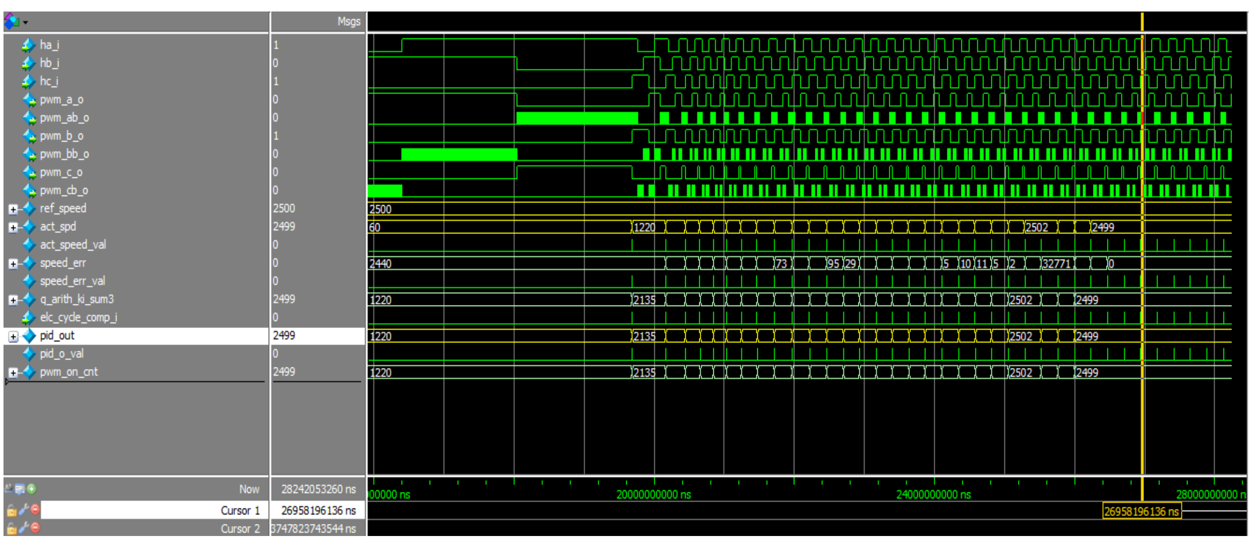

Figure 8 : Simulation Results

B. Chipscope Pro Analyser Results

Figure 9 represents in chip Xilinx Kintex 7 FPGA debugged waveform using Chip Scope Pro Analyzer for PID control logic. PID output matches (pid_o) with set reference (ref_speed) after certain electrical cycles.

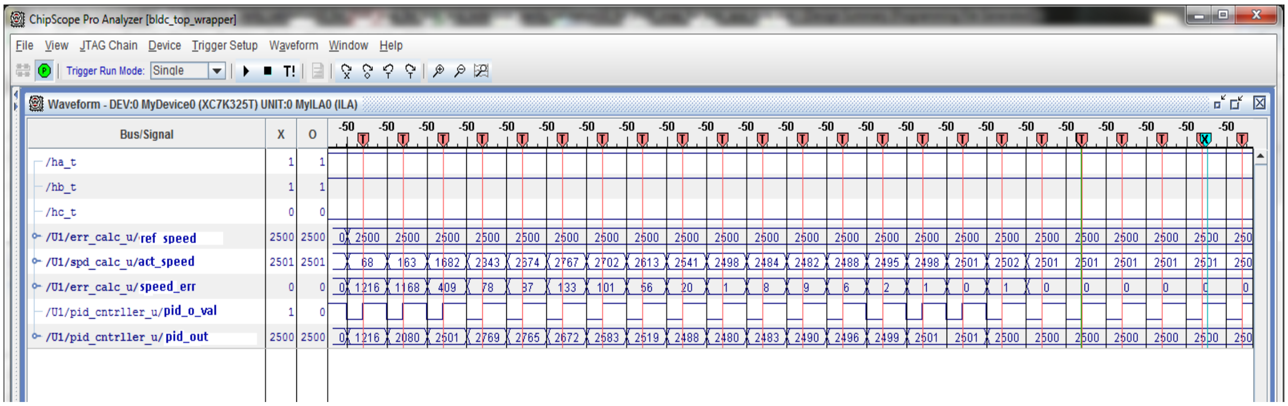

Figure 9 : Chip scope Pro Analyzer Results

C. Comparison with mathematical model

Digital FPGA based implementation with integer arithmetic produces response very close to mathematical model response as shown in figure 10.

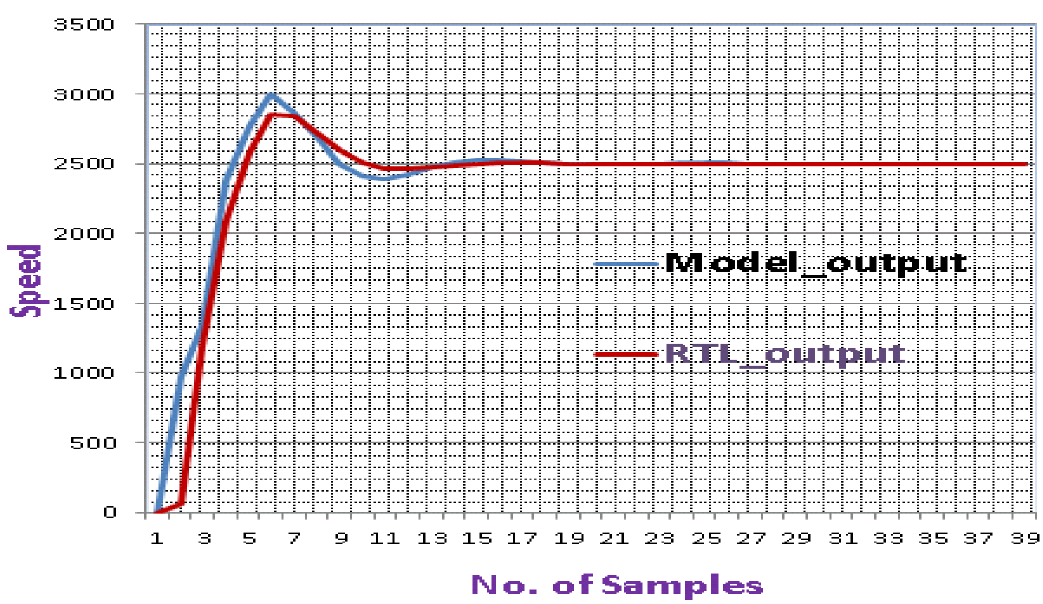

Figure 10 : Model output Vs RTL Implementation output

Conclusion

Through this paper, Hall sensors based closed loop implementation to control the speed of the motor is discussed. Simulation results and chipscope validation results of implemented PID based BLDC motor controller shows accurate results. Use of PID in the closed loop implementation adds efficiency and stability to systems. The FPGA based implemented motor controller can be used for system critical applications where the system stability and motor speed accuracy are the determining factors.

References

I. slaa503: Texas Instruments Application Report on "Sensored 3-Phase BLDC Motor Control Using MSP430".

II. Ishnatek Application Note: "Implementing a Brushless DC Motor Controller on an IGLOO FPGA".

III. Dr.Varodom Toochinda, "Digital PID controllers", June 2011

About Authors

1. **Mr. Sandeep Masade**, Technical Lead – ASIC has 11 years of experience in eInfochips. His expertise is in to Design and verification of FPGA and ASIC. He has worked on more than 12 System on Chips (SOC)/FPGA products in semiconductor industry from Micro Architecture definition to system level verification and validation of complex devices in various computing and data transfer applications. He is well versed with different technologies, protocols, algorithm and their implementations and having good exposer on the standards like European Aviation Safety Agency (EASA) and DO-254 which he has accumulated through his years of experience.

2. **Mr. Shubham Parmar** works at eInfochips as Senior Engineer. Shubham has experience in ASIC /FPGA design and verification cycle. He has worked on complete FPGA system implementation flow covering micro-architecture design, RTL coding, gate level simulation, synthesis, PAR, timing closure and board validation. He has designed and implemented various Algorithms for Communication systems and has good exposure on the standards like DO-254. He is well versed with different technologies, protocols, algorithm and their implementations. He has worked on variety of projects covering applications in domains like Satellite communication, Automation, and DSP.

3. **Mr. Atul Jaisen Bhanushali**, ASIC/FPGA Engineer has 4 years of experience in eInfochips. He has worked on complete FPGA system implementation flow covering micro-architecture design, RTL coding, gate level simulation, synthesis, PAR. He is well versed with different protocols, algorithm and their implementations and having good exposer on the standards like DO-254 which he has accumulated through his years of experience.