

Adaptive Inverse Neural Network Based DC Motor Speed and Position Control Using FPGA

Abbas H. Issa, Aula N. Abd

Department of Electrical Engineering, University of Technology
30050@uotechnology.edu.iq, aula_najm@yahoo.com

Abstract

In this research two types of controllers are designed in order to control the speed and position of DC motor. The first one is a conventional PID controller and the other is an intelligent Neural Network (NN) controller that generate a control signal DC motor. Due to nonlinear parameters and movable laborers such saturation and change in load a conventional PID controller is not efficient in such application; therefore neural controller is proposed in order to decreasing the effect of these parameter and improve system performance. The proposed intelligent NN controller is adaptive inverse neural network controller designed and implemented on Field Programmable Gate Array (FPGA) board. This NN is trained by Levenberg-Marquardt back propagation algorithm. After implementation on FPGA, the response appear completely the same as simulation response before implementation that mean the controller based on FPGA is very nigh to software designed controller. The controllers designed by both m-file and Simulink in MATLAB R2012a version 7.14.0.

Keywords: DC motor, speed control, position control, PID, ANN, Inverse Neural Controller, FPGA.

Paper History: Received: (16/3/2017), Accepted: (2/5/2017)

1.Introduction

DC motors is very common devices in different industrial applications and autonomous systems, such as disk drives, automatic doors, robotic arms and numerous applications. For all that motors are widespread, and various controllers have been suggested and executed to get better efficiency, the nonlinear parameters still has effect on overall efficiency.

There are two output of DC motor can be controlled which are position and speed. Suitable control of these two outputs lead increasing system's efficiency. In several applications, such as robotic and disk drives, control on position is more substantial than control on speed [1].

The FPGA technology has a lot of benefits such as quick response, flexibility, reliability and ability of reprogramming. As well, it can be used for complicated control system. In motor control and robot control systems, FPGA used as a hardware platform for controller [2].

There are number of previously published researches within the controllers and DC motor system. In 2009, M.Fallahi, et al proposed an adaptive NN sliding mode controller that is efficient method for controlling DC servo motor. NN are used to mimic model and control of sophisticated systems due its ability to processing complex input/output mapping [3]. In 2011, Dr. Muna H. Saleh, et al proposed design, and simulate fuzzy mimic PD controller for autonomous mobile robot which has two DC servo motors to perform the matter of safety movement of the robot [4]. In 2014, Ibrahem S. Fatah, use particle swarm optimization (PSO) with speed PID controller for DC motor then compare response with conventional PID utilize by genetic algorithm [5]. In 2014, A. Medina-Santiago, et al., proposed the NN controller for mobile robot performing detection of obstacle by using multilayer perceptron trained by MATLAB, and implementation neural control on microcontroller (Arduino), DC motors and ultrasonic sensors [6]. In 2015, Wisam Najm al-din Abed proposed state feedback control technique in order to control on speed of the separately excited DC motor by tuning gain for controller using technique called bacterial foraging optimization (BFO) [7].

The aim of this work is design a classical PID and adaptive inverse neural network controllers in order to control on speed and position of DC motor. After comparison between them, the adaptive inverse neural network controller detracton the effect of nonlinear parameter and produce efficient response for control system. So this type was chosen to implementation on FPGA Spartan 3A from Xilinx. The schematic block diagram of complete controlled system of DC motor shown in Figure 1.

This paper is split into: Part:1 display the introduction, Part:2 explain the DC motor modeling, Part:3 presents PID controller,

Part:4 present the adaptive direct inverse NN controller, part: 5 explain FPGA design. The simulation results are displayed in Part: 6, Part:7 discuss the hardware implementation and part: 8 civilize the conclusion.

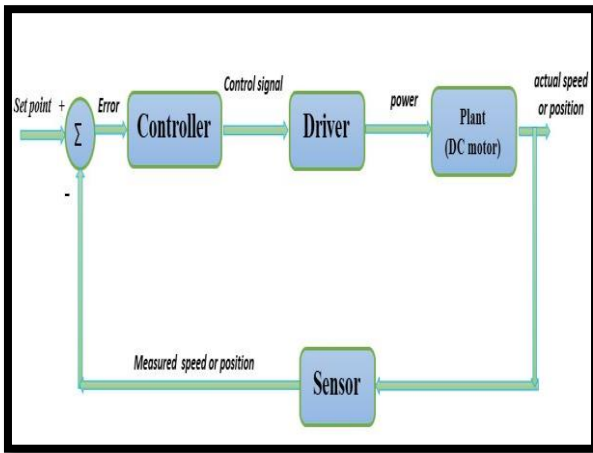


Figure 1: block diagram for control system

2. DC Motor Modeling:

In control systems DC motor consider a common actuator that provides rotary motion, the translational motion can provide by coupled motor shaft with wheels. Figure 2 explain the electric circuit that equivalent armature and the free-body diagram of the rotor. This system take source voltage (V) as the input that applied on the armature and translate this voltage in to the shaft rotational speed at the output.

Generally, the model above composed by two parts according to the main parameters form that part: armature resistance (R) and inductance (L) form the electrical part, but moment of inertia of the rotor (J) and friction of the mechanical system (b) form the mechanical part. By a constant factor K_t the generated torque (T) is proportional only with the armature current (i) when the magnetic field is constant as given in equation 1, this is called motor with armature-controlled.

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

The back electromotive force (e), is proportional by a constant factor K_b to the angular velocity (ω) of the shaft as given in equation 2.

$$e = K_b \omega = K_b \frac{d\theta}{dt} \tag{2}$$

By applying Newton's law and KVL on the circuit of Figure 2 and convert resulting equations in to Laplace transform then re-order those equations, the input voltage $V(s)$ to output position $\Theta(s)$ transfer function and the input voltage $V(s)$ to output speed $\omega(s)$

transfer function appear in equation 3 & 4 respectively [8].

$$\frac{\theta(s)}{V(s)} = \frac{K}{s[(Ls+R)(Js+b)+K^2]} \tag{3}$$

$$\frac{\omega(s)}{V(s)} = \frac{K}{[(Ls+R)(Js+b)+K^2]} \tag{4}$$

These equations used to show the block diagram for DC motor as in Figure 3.

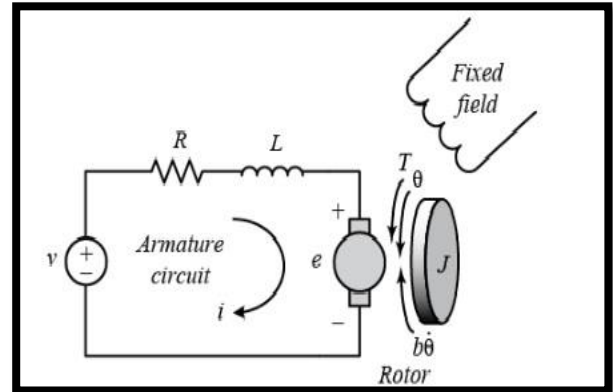


Figure 2: DC motor equivalent circuit and free-body diagram of the rotor [8]

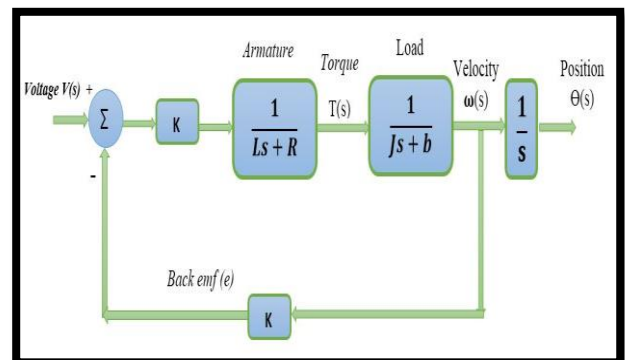


Figure 3: A block diagram of DC motor

3. Pid Controller:

One type of the conventional controllers called the Proportional-Integral-Derivative (PID) controller. The block diagram of a PID controller with plant is shown in Figure 4. The tracking error $e(t)$ represents the difference between reference input $r(t)$ and actual output $y(t)$. After that $e(t)$ become input to the PID controller. The control signal equation in time domain look as:

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \tag{5}$$

After take laplace transform for both sides of equation above and then re-order resulting equation, the transfer function of PID controller is given by:

$$\frac{u(s)}{e(s)} = K_p + \frac{K_i}{s} + K_d s \tag{6}$$

Where: K_p , K_i and K_d are the proportional, integral and derivative gains respectively for the PID controller [9].

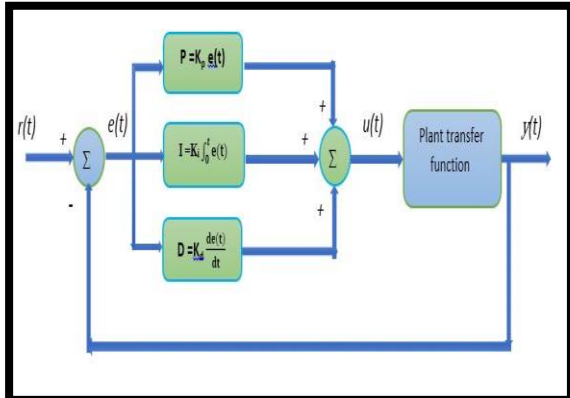


Figure 4: Structure of PID controller with plant TF

4. Adaptive Direct Inverse NN Controller:

The PID controller system could not give the desired control impact because that controlled system as yet suffers from unsuitable parameters such as slow response and nonlinearity, which make that system with low efficiency [10]. Therefore, NN controller for DC motor was proposed. An Artificial Neural Networks (ANNs) are an information processing models that inspired by the human brain cells. Each single ANN consist of a many number of strongly interconnected processing elements called neurons, these neurons action in congruence to solve specific problems [11]. There are different types of NN each has a special architecture and various training methods used to train these NN. Typically, a multilayer Feed Forward Neural Network (FFNN) composed by one input layer, one or more than one hidden layers and one output layer, Figure 5 represent architecture of a three layer FFNN. Any random function could be closer by using multilayer FFNN therefore, NN with the linear (purelin) transfer function at the output layer and one or more than one hidden layer with linear (purelin) or nonlinear (tansig) transfer function which trained to obtain better control than conventional PID by adjusting weights of NN [12].

Direct inverse neural network control is special type of adaptive control. The basic idea of this type of controller is offer an open loop control system by using a series controller has transfer function nearly inverse transfer function of the plant. This type of neural network control is also called series neural network control because this controller connected in series with the plant. The significant of utilization the

inverse of plant as a controller is to make output response of the plant closer to reference input [13]. The block diagram of direct inverse neural control shown in Figure 6. The training set of this type of controller is input/output pairs for plant, the NN receives output of plant and trying to regenerate plant input strictly.

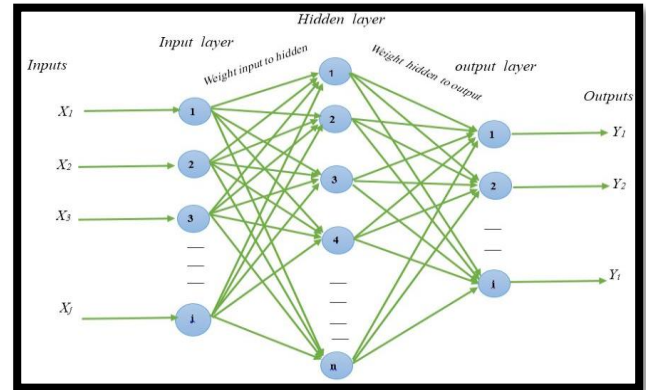


Figure 5: Architecture of a three layer feed forward Neural Network

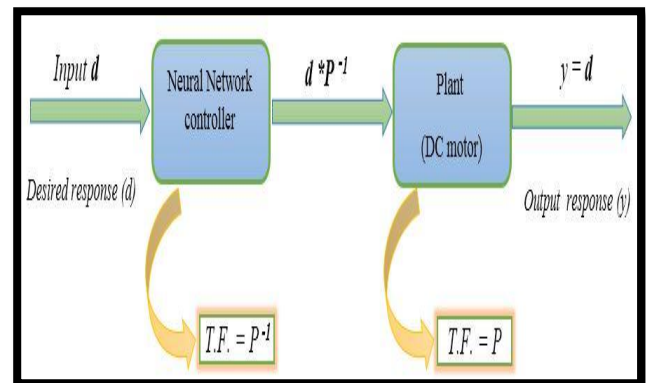


Figure 6: block diagram of inverse NN controller with plant

5. Field Programmable Gate Array:

FPGAs are re-programmable integrated circuits (ICs) and utilized to execution any logic function such as addition, subtraction and any complex function [14]. Configurable Logic Blocks (CLBs), programmable interconnection and Input Output Blocks (IOBs) are the basic components of FPGA as shown in Figure 7. The large FPGA can has tens of thousands of CLBs as well as memory and other resources [15].

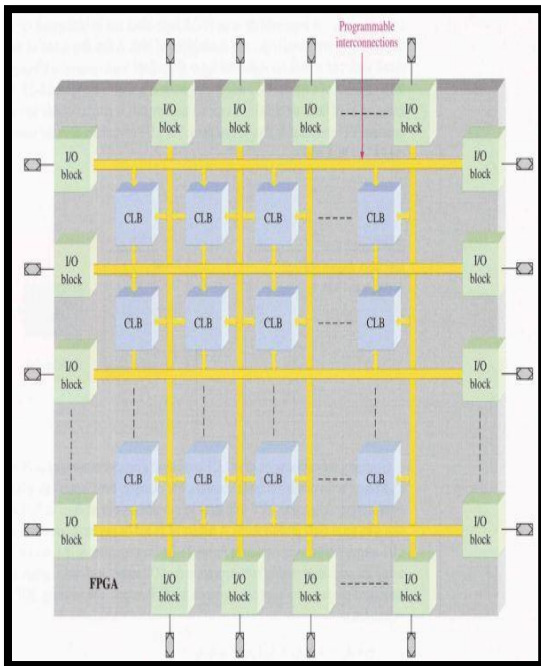


Figure (7): Basic components for FPGA [15]

6. Simulation Results:

The physical parameters of the DC motor used in this paper are given in Table 1. By substitute these values in equation 4 & 5, the T.F. of position and the T.F. of speed are shown below:

$$\frac{\theta(s)}{V(s)} = \frac{0.01}{s[(0.5s+1)(0.01s+0.1)+0.0001]} \quad (7)$$

$$\frac{\omega(s)}{V(s)} = \frac{0.01}{[(0.5s+1)(0.01s+0.1)+0.0001]} \quad (8)$$

The control of speed and position of DC motor implemented using above transfer functions. The control system simulated in MATLAB/Simulink program, a response for both speed and position is obtained with two types of controllers which are PID and adaptive direct inverse NN controller. For the designed PID controller, tuning PID parameters by using automatic tune the PID controller block from MATLAB/Simulink will obtain the values of K_p , K_i , and K_d as (17.937, 43.45 and -0.7768 respectively) (43.45, 1.335 and 16.836 respectively) for speed and position control respectively. This type of controller has overshoot and slow response as shown in simulation result for both speed and position control in Figure 8 & Figure 9 respectively.

After that, adaptive direct inverse NN controller is designed by programming using M-file then convert to Simulink block, the training data for this type of controller is both input and output for plant transfer function, the output for plant is become training input for

NN and the input of the plant become target output for NN, the actual output for NN controller compare with target output and the difference between them is the error which is back propagated to the network until reach the acceptable value for error. The structure for the neural controller has 7 neurons for first hidden layer, 4 neurons for second hidden layer, one neuron in both the input and output layers. Two response obtained, the first one with linear transfer function (*purelin*) TF and other response obtain with non-linear transfer function (*tansig*) TF for the hidden layers, in both cases linear (*purelin*) TF was used for output layer. The acceptable value for error setting to 10^{-5} , that makes the actual output to be very close to desired output. The response in this type of controller for both speed and position control shown Figure 10 & Figure 11 respectively. After comparison results, this controller is the robustness and effectiveness type over PID controller.

In order to comparison more clarified, the designed controller types compared together in the same graphic result for both speed and position control with purelin T.F. then with tansig T.F. for hidden layers as shown in Figure 12 & Figure 13. Table 2 show the more clarified comparison for speed and position controller types with purelin T.F., and Table 3 show the same comparison but with tansig T.F.

Table 1 parameters value for DC motor.

Physical parameter	Symbol	Value	Unit
moment of inertia of the rotor	J	0.01	Kg.m ²
damping of the mechanical system	b	0.1	Nms
back-electromotive force constant	K	0.01	Nm/A
electric resistance	R	1	Ω
electric inductance	L	0.5	H

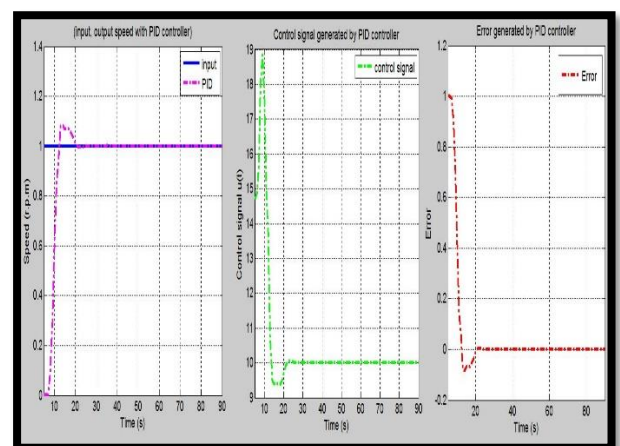


Figure 8: The system response, control signal and error for PID speed controller

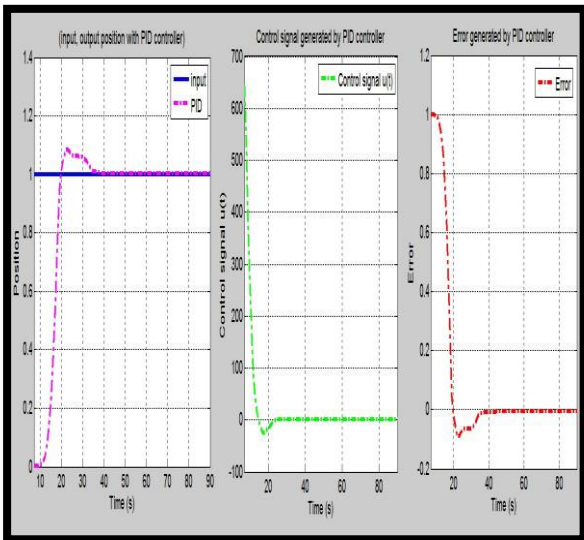


Figure 9: The system response, control signal and error for PID position controller

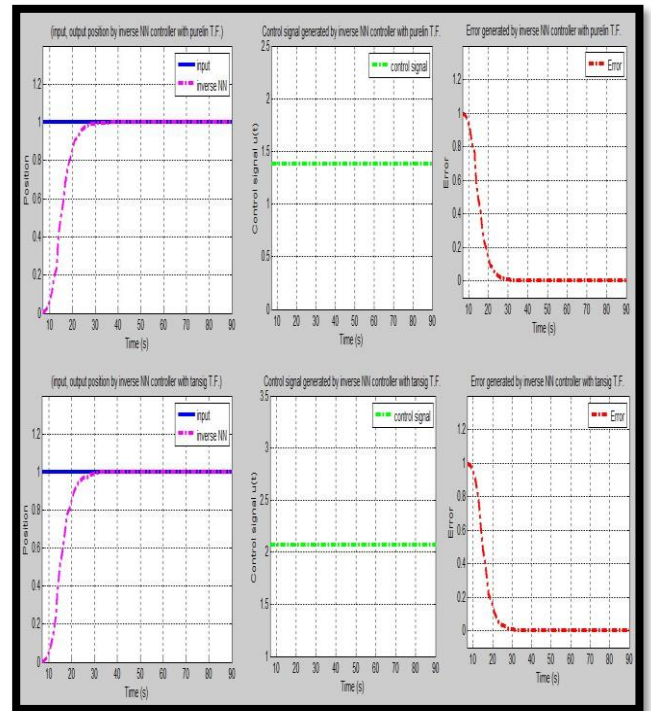


Figure 11: The system response, control signal and error for adaptive direct inverse NN position controller

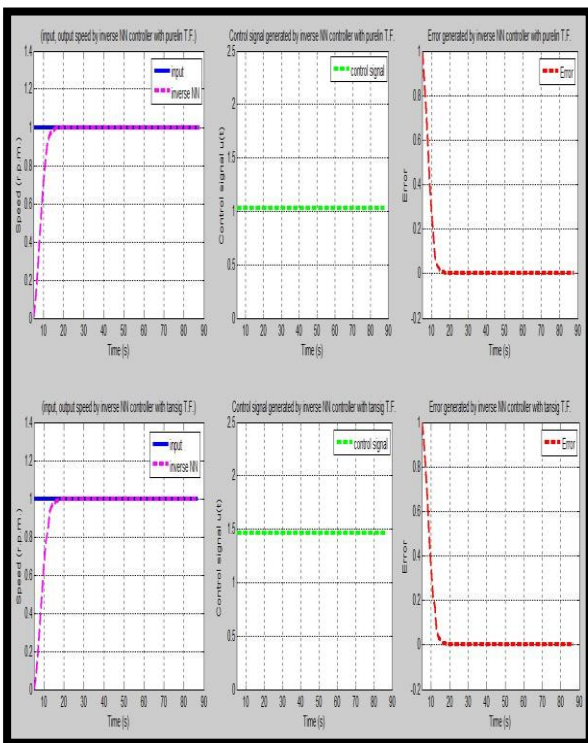


Figure 10: The system response, control signal and error for adaptive direct inverse NN speed controller

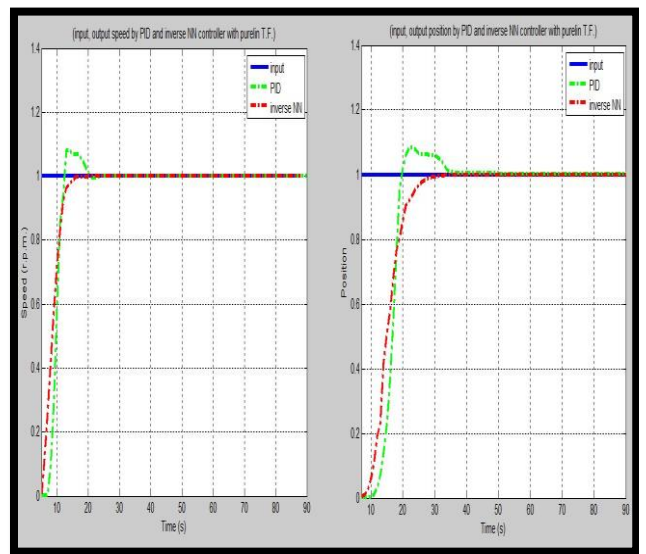


Figure 12: The system response Comparison between controller types with purelin T.F

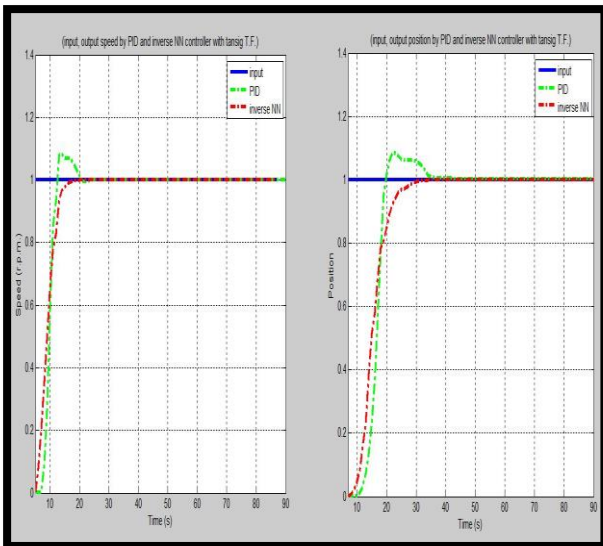


Figure 13: The system response Comparison between speed controller types with tansig T.F

Table 2 Comparison between performances of controller types with purelin T.F for hidden layer.

Speed Controller	t_d (sec)	t_r (sec)	t_p (sec)	M_p	t_s (sec)
PID	9.8	12.5	14	0.083	21
Adaptive Inverse NN	8.7	12	22	0.0006	16
Position Controller	t_d (sec)	t_r (sec)	t_p (sec)	M_p	t_s (sec)
PID	16.5	19.8	23	0.085	34
Adaptive Inverse NN	14.7	27	30	0.0036	29

Table 3 Comparison between performances of controller types with tansig T.F. for hidden layer

Speed Controller	t_d (sec)	t_r (sec)	t_p (sec)	M_p	t_s (sec)
PID	9.8	12.5	14	0.083	21
Adaptive Inverse NN	9	14	20	0.0005	18
Position Controller	t_d (sec)	t_r (sec)	t_p (sec)	M_p	t_s (sec)
PID	16.5	19.8	23	0.085	34
Adaptive Inverse NN	15	22	30	0.0068	32

7. Hardware Implementation of AINN Control Based FPGA:

After comparison between performances of the controller types, the best one is Adaptive Inverse NN (AINN) control; therefore this type with purelin T.F. was chosen for hardware implementation on FPGA. The controller converted to Very high speed integrated circuit Hardware Description Language (VHDL) code then this code compiled and downloaded on FPGA Spartan 3A board shown in Figure 14 using steps design shown in Figure 15.

By Xilinx system generator the VHDL code files are pulled to black box from matlab /Simulink blocks which uses for download process on FPGA through the JTAG (Joint Test Action Group) to work in loop simulation by using USB cable. The black box and JTAG block with I/O ports for speed and position controller shown in shown in Figure 16.

The realization of implemented hardware AINN controller on FPGA done by connecting the PC with FPGA board through USB cable and supply the board of FPGA with power then running the simulation for control system in PC with FPGA as controller, the response results obtained and compared with the response before implementation as shown in Figure 17 which shows that the two response are completely the same and that mean the hardware is very nigh to software designed controller.

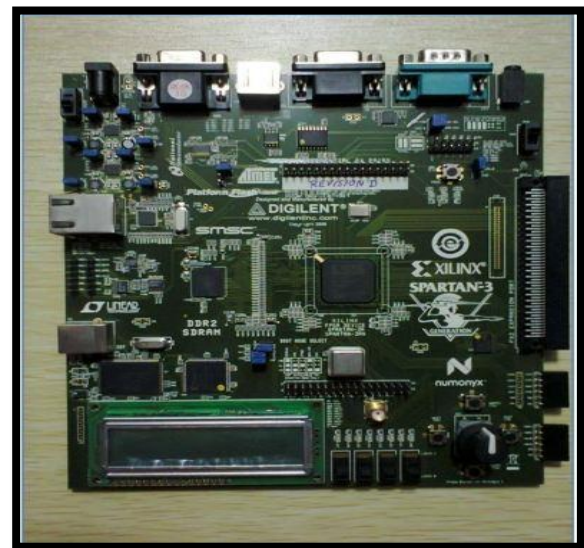


Figure 14: FPGA board

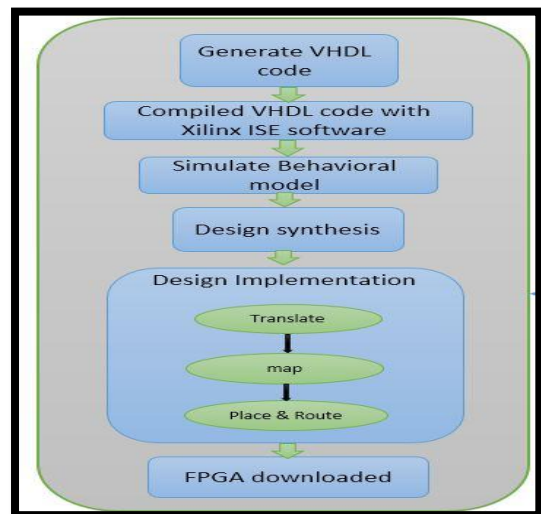


Figure 15: Design steps for FPGA

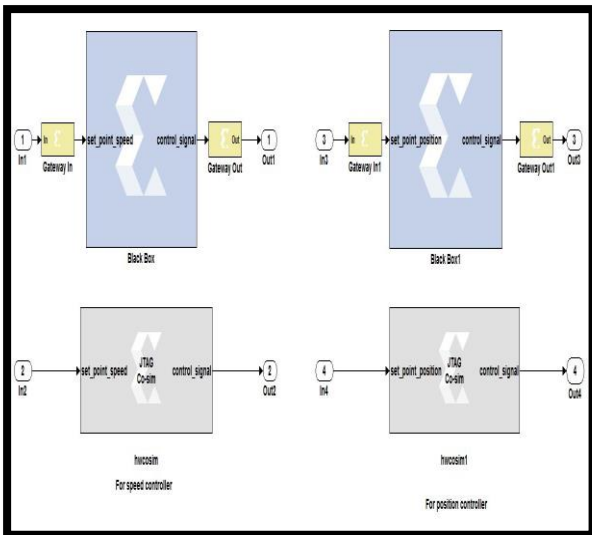


Figure 16: The black box and JTAG block with I/O ports for speed /position controller

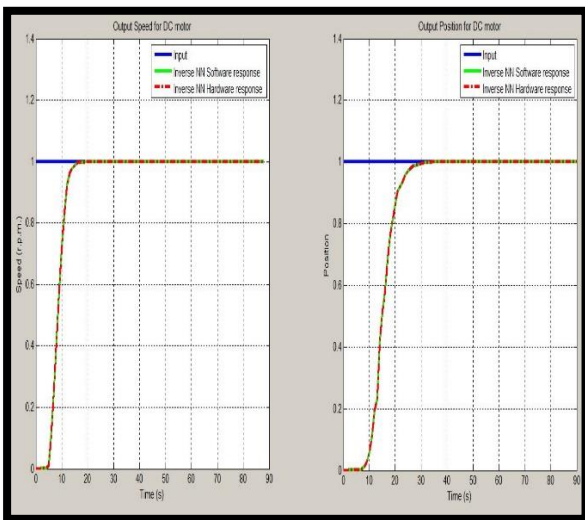


Figure 17: The control system response comparison for AINN controller before and after implementation on FPGA

8. Conclusions

In this paper two types of controllers for DC motor were designed including the conventional PID and intelligent adaptive inverse NN controller, in which AINN controller used to mimic inverse transfer function or inverse function for plant. The simulation results explain well that the AINN controller is the best and the most suitable for systems if high efficiency controllers are required because it has excellent and efficient performance over PID controller which still suffers from unsuitable parameters such as nonlinearity, high peak overshoot and slow response. For that reasons PID controller system could not give the desired control impact and make that system with low efficiency.

After that the best one which is AINN controller implemented on FPGA board and the response shows high accuracy and efficiency for the designed controller based FPGA.

9. References

- [1]. Gutierrez F. R., Makableh Y. F., Efficient Position Control of DC Servomotor Using Back propagation Neural Network, IEEE, International Conference on Natural Computation, (2011), 653-675.
- [2]. Hasanién H. M., FPGA Implementation of Adaptive ANN Controller for Speed Regulation of Permanent Magnet Stepper Motor Drives, Elsevier, journal, (2011), 1252-1257.
- [3]. Fallahi M., Azadi S., Adaptive Control of a DC Motor Using Neural Network Sliding Mode Control Proceedings of the International Multi Conference of Engineers and Computer Scientists, 2, (2009).
- [4]. Saleh M. H., Othman M. Z., Al-Qassar A. A., Design and Simulation of Fuzzy Like PD Controller for Autonomous Mobile Robot, the Engineering Conference of Control, Computers and Mechatronics (ECCCM), Control and Systems Engineering Department, University of Technology Baghdad-Iraq, (2011), 121-127.
- [5]. Fatah I. S., PSO-Based Tuning of PID Controller for Speed Control of DC motor, Diyala Journal of Engineering Sciences, 07, (03), (2014), 65-79.
- [6]. Medina-Santiago A., Camas-Anzueto J. L., Vazquez-Feijoo J. A., Hernández-de León H. R., Mota-Grajales R., Neural Control System in Obstacle Avoidance in Mobile Robots Using Ultrasonic Sensors, Journal of Applied Research and Technology, 12, (2014), 104-110.
- [7]. Abed W. N., Design of State Feedback Controller Based Bacterial Foraging Optimization Technique for Speed Control of DC motor, Diyala Journal of Engineering Sciences, 08,(01), (2015), 134-152.
- [8]. Telba A., Motor Speed Control Using FPGA, IEEE, Proceedings of the World Congress on Engineering, 1(2014).
- [9]. Mustafa G. Y., Ali A. T., Bashier E., Elrahman M. F., Neuro-Fuzzy Controller Design for a Dc Motor Drive, University of Khartoum Engineering Journal (UofKEJ), 3, (2013), 7-11.
- [10]. Athi thilagalakshmi R., Vijay Anand L D., Simulation of Neuro-PID Controller for Pressure Process, International Conference on Innovations In Intelligent

Instrumentation, Optimization and Signal Processing (ICIIOSP), (2013), 18-21.
[11]. Maind S. B., Wankar P., Research Paper on Basic of Artificial Neural Network, International Journal on Recent and Innovation Trends in Computing and Communication (IJRITCC), 2, Issue: 1, (2014), 96-100.
[12]. Peng D., Zhang H., Huang C., Xia F., Li H., Immune PID Cascade Control Based on Neural Network for Main Steam Temperature System, IEEE, World Congress on Intelligent Control and Automation, (2011), 480-484.
[13]. Obeid Ahmed A. H., High Performance Speed Control of Direct

Current Motors Using Adaptive Inverse Control, wseas transactions on systems and control, 7, (2012), 54-63.
[14]. Kocur M., Kozak S., Dvorscak B., Design and Implementation of FPGA - Digital Based PID Controller, IEEE, 15th International Carpathian Control Conference (ICCC), (2014), 233-236.
[15]. Thomas L. Floyd, Digital Fundamental, Person Education International, Edition, (2006), 886.