Neural Network on FPGA for PID Controller

Ashlesha Panbude[1], Dr. Manish Sharma[2]

[1]M.E. VLSI & ES, Department of E&TC, D.Y. Patil College of Engineering, Akurdi, Pune, Maharashtra, India

[2]Department of E&TC, D.Y. Patil College of Engineering, Akurdi, Pune, Maharashtra, India

Abstract: ***Evolution of bio-inspired concepts like neural networks learning have attracted the attention of complex and intelligent systems because of a growing interest in automatic design, capable of adaptation and fault tolerance. Artificial neural network have found widespread deployment in different areas of classification, perception, control application etc. Control Objects are becoming more complex, especially for the unknown parameters or slow changes in large delay, time-varying, nonlinear complex systems, with delay or random interference. PID controller is well known industrial used controller. Combining the artificial neural networks, genetic algorithms or other intelligent control theories with conventional PID control, can overcome the disadvantages of conventional PID control. This paper presents the hardware implementation of neural network using FPGAs. Digital system architecture is presented using Very High Speed Integrated Circuits Hardware Description Language (VHDL) and is implemented in FPGA .***

Keywords: PID controller; Artificial neural network, FPGA

1. Introduction

The concept of ANNs is emerged from the principles of brain that are adapted to digital computers. Artificial Neural Networks (ANNs) can solve great variety of problems in areas of pattern recognition, image processing and medical diagnostic. The biologically inspired ANNs are parallel and distributed information processing systems. Neural networks can be "trained" to solve problems that are difficult to solve by conventional computer algorithms. Training refers to an adjustment of the connection weights, based on **a** number of training examples that consist of specified inputs and corresponding target outputs.

The simple processing elements (PE, artificial neurons) that are interconnected by weighted connections. The predominantly used structure is a multilayered feed-forward network (multilayer perception),i.e., the nodes (neurons) are arranged in several layers (input layer, hidden layers, output layer), and the information flow is only between adjacent layers. An artificial neuron is a very simple processing unit. It calculates the weighted sum of its inputs and passes it through a nonlinear transfer function to produce its output signal system.

Implementation of ANNs falls into two categories: Software implementation and hardware implementation. ANNs are implemented in software, and are trained and simulated on general-purpose sequential computers for emulating a wide range of neural networks models. Software implementations offer flexibility. However hardware implementations are essential for applicability and for taking the advantage of ANN's inherent parallelism. VLSI implementations of ANNs provide high speed in real time applications and compactness.

Digital implementation of ANNs can make use of full custom VLSI, semi custom, ASICs (application specific integrated circuits) and FPGAs. Particularly, FPGA implementation of ANNs is very attractive because of the high flexibility that can be achieved through the re-programmability nature of these circuits. Also, FPGA is concurrent, which supports the massively parallel calculation of neural network.

2. Artificial Neural Network

The biologically inspired ANNs are parallel and distributed information processing systems. A Neural Network is a powerful data-modeling tool that is able to capture and represent complex input/output relationships. The motivation for the development of neural network technology stemmed from the desire to develop an artificial system that could perform "intelligent" tasks similar to those performed by the human brain. Neural networks resemble the human brain in the following two ways:

1. A neural network acquires knowledge through knowledge.
2. A neural network's knowledge is stored within inter-neuron connection strengths known as synaptic weights

The concept of ANNs is emerged from the principles of brain that are adapted to digital computers. The first work of ANNs were the models of neurons in brain using mathematics rule. Each neuron in ANNs takes some information as an input from another neuron or from an external input. This information is propagated as an output that are computed as weighted sum of inputs and applied as non-linear function as shown in figure1

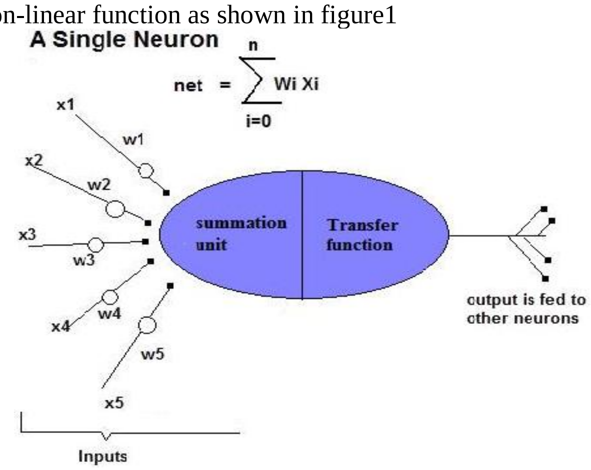


Figure 1: Block diagram for single neuron

Neural network are a form of multiprocessor computer system with:

- Simple processing element,
- High degree of interconnection,
- Adaptive interaction between elements.

3. PID Controller

. Proportional-Integral-Derivative (PID) control is the most commonly used control algorithm in industry and has been universally accepted in industrial control, like for industries such as chemical, petrochemical, robotics etc. The popularity of PID controllers is due to the fact that they are low cost and easy to maintain also gives robust performance in a wide range of operating conditions and its functional simplicity, which allows engineers to operate them in a simple, straightforward manner, but the conventional PID control parameters will have no changes after completion, resulting in the parameter variations of controlled objects which is difficult in real time. Today's complex system makes it difficult for PID to control non linear parameters.

4. PID Control Principle

Since computer is accessed in control field, digital computer is used to replace simulated computer adjuster to compose computer control system, thus it not only can use software to achieve PID control algorithm, but can use the computer's logic function to make PID control more flexible. Digital PID control in the production process is the most commonly used control method, widely used in metallurgy, machinery, chemical and other industries. In the simulated control system, the PID controller is controlled according to the proportion of deviation (P), integral (I) and differential (D) is the most widely used automatic controller, the control principle as shown in figure 2

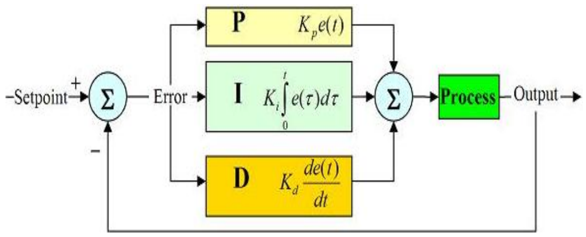


Figure 2: PID Controller

Therefore, PID controller has the following characteristics: Principle is simple, easy to achieve, its a basic controller that can meet the majority of actual needs, controller can be applied to a variety of different objects, the algorithm has strong structural robustness in many cases, its control quality is not sensitive to the structure and parameter perturbations of controlled object.

The main limitation of PID control is its dependence on the controlled object, generally needs to know in advance the mathematical model of the controlled object to design. In practical industrial control, due to this requirement the controlled object has non-linear, time variability and other characteristics, so it is difficult to establish accurate mathematical model or the characteristic parameters obtaining online , making its application limited.

5. PID Neural Network

The structure of PID neural network could be given as follows:

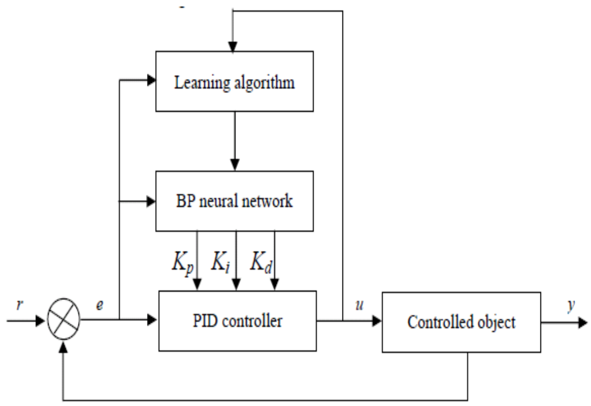


Figure 3: Structure of PIDNN

Using neural network PID controller to replace the ordinary PID controllers can make the error between the system output and expected values minimum the controller consists of two parts, namely, conventional PID control and neural networks, in which, the conventional PID directly controls the controlled object with a closed loop, and its control parameters Kp, Ki and Kd are online adjust mode; neural network is to adjust the parameters of PID controller based on the operational status of the system, to achieve a performance optimization, making the output of the output neurons corresponding to the three adjustable parameters of PID controller.

The training of the PID parameters for the system is done using matlab i.e the three parameter kp, ki, kd are tuned. Then these tuned parameter are used for any error value between the actual value and the desired value of the system. the parameter are well tuned to maintain the controller operation for the desired value.

Figure 5 shows the graph for a PID controller with Kp=100, Ki=200, Kd=10. In VHDL an LUT is then created so that according to error i.e difference between actual and the desired value the weights of neural network are adjusted accordingly to give the desired kp, ki, kd parameters.

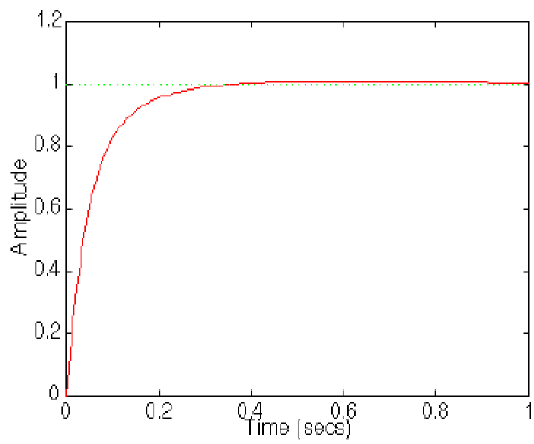


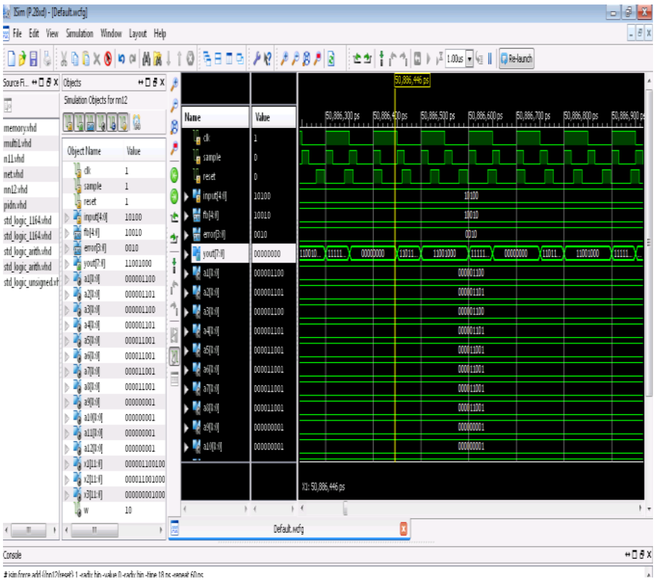
Figure 5: Tunned output of PID controller

These tuned parameters of PID are then applied to the controller to get the PID output.

$$u(t) = k_p e(t) + k_i \int e(t) + k_d \frac{de(t)}{dt}$$

6. Results

The VHDL coding is done using Xilinx 14.2 version. The figures below shows the obtained result, where according to error weights are adjusted through the look up table and then it is applied to the pid controller to give the output. In PID, actuator is used in order to obtained and maintained the précised output.



7. Conclusion

1) Irrelative of change in the difference between desired and actual input of the system, parameters of PID are maintained to obtain the desired output value.
2) The advantages of neural network for implementing PID controller is that it has strong adaptive and self-learning capability, which further improves the performance of PID controllers removing the drawbacks of conventional PID.
3) Implementing on FPGA gives the system , flexiblity, high degree of parallelism with less use of the resources

8. Acknowledgements

Authors would like to thanks to Dr. D.G Khairnar for his support.

References

[1] Yu Yong quan, Huang Ying and Zeng Bi, “A PID Neural Network Controller,” Proceeding of the International Joint Conference on Neural Net Works, IEEE Computer Society Press, California, vol. 3, pp. 1933-1938, 2003.

[2] Indranil Pan, Saptarshi Das, Amitava Gupta, “Tuning of an optimal fuzzy PID controller with stochastic algorithms for network control Systems with random time delay,” ISA Transaction, vol. 50, pp. 28-36, 2011.

[3] G.jahedi, M.M.Ardehali, “Genetic algorithm-based fuzzy- PID control methodologies for enhancement of energy efficiency of a dynamic energy system,” Energy Conversion and management, vol. 52, pp. 725-732, 2011.

[4] Saptarshi Das, Indranil Pan, Shantanu Das, “A novel fractional order fuzzy PID controller and its optimal time domain tuning based on integral performance indices,” Engineering Applications of Aritficial Intelligence, online, 2010.

[5] Yaohua Guo, Junshuang ma, Minglin Yao, “The neural network PID controller for cement Rotary kiln Temperature based on FPGA”, Information science and engineering(ISISE) December 2010.

[6] Liguo Qu, Yourui Huang and Liuyi Ling, "Design if Intelligent PID controller based on Adaptive Genetic Algorithm and Implementation of FPGA", Springer 2008.

[7] Wei-Der Chang.,”A multi-crossover genetic approach to multivariable PID controllers” Tuning [J]. Expert Systems with Applications, 2007, 33(3): 620-626166-171.

[8] Shu Huailin, “PID Neural Network Control for Complex Systems,” Processing’s of International Conference on tational Intelligence for Modelling, Control and Automation CCIMCA 99’2,10s Press, 1999, pp. 166-171.

[9] Moradi, M.H. “New Techniques for PID Controller Design,” Proceeding of 2003 IEEE Conference on Control Applications, IEEE Press, New York, vol. 2, pp. 903-908, 2003.

[10] Sangeetha T, Meenal C, “Digital Implementation of Artificial Neural Network for Function Approximation and Pressure Control Applications,” IOSR Journal of Electronics and Communication Engineering (IOSR-JECE), Volume 5, Issue 5,PP 34-39 (Mar. - Apr. 2013).