ANN Architecture for Signal Processing Applications

S N Prasad [# 1]

#1.Research Scholar, Jain University,
Assistant Registrar (Evaluation), School of ECE,
REVA University, Bangalore, India
snprasad.aithal@gmail.com

S.Y.Kulkarni [#2]

#2.Principal Director
REVA University, Bangalore, India
kulkarni_sy@reva.edu.in
sy_kul@yahoo.com

Abstract - ANN finds its applications in various signal processing applications such as image recognition(image processing techniques),pattern recognition, system identification, different types of filters(FIR,IIR) and other control problems. In this paper, a multilayer perceptron with more than one hidden layers is considered and for the image processing application such as image recognition, A design of an architecture for multilayer perceptron neuron network is achieved using FPGA. The design is implemented by using the different activation functions such as linear activation function, hard limiter activation function, piecewise linear activation function etc.. A neural network was implemented by using VHDL hardware description Language codes and XC3S250E-PQ 208 Xilinx FPGA device. The results were presented using Xilinx Foundation 9.2i.

Index terms - Multilayer Perceptron, Back propagation, Activation functions, FPGA, Neuron Plan approximation.

INTRODUCTION

Artificial Neural network (ANN) is a computation intelligence system that elaborates the abilities of biological brain. It is used to simulate the functions and architecture of human brain. It is a parallel and distributed signal processing system; which helps in solving various problems related to pattern recognition, image processing, filter applications (FIR & IIR), prediction, associative memory and bio-medical signal processing applications [1].

Many methods were invented for developing such applications but most of them couldn't fetch the appropriate results. ANN provides great alternatives and most of them are application oriented. One among the signal processing applications are the parallel system with interconnected processors. [2].

ANN can be implemented in both hardware and software. In software they are trained and then modulated to the hardware neural models using general purpose computers. Some of the learning algorithms can be implemented and used to get the appropriate results .i.e, machine learning approach could be used. Hardware implementations are necessarily meant for taking the benefit of parallelism and a specific purpose ANN hardware provide high performance and compactness for the real time applications [3].

The parallelism attached with ANN can be fully realized by the hardware implementation. The speed of the general purpose computers limits the realization of ANN and hence such parallel architectures are implemented in VLSI [4] .Digital and Analog architectures were utilized for ANN development. However the analog method may save the resource cost but lacks the design flexibility and accurate computations [1].

The entire architecture is realized by considering a simple neuron model, developing that by considering the multi-layer perceptron model, enhancing the network with the aid of

some activation functions and finally realizing any of the signal processing applications.

ARCHITECTURE:

ANN is made up of many processing units such as neurons or perceptrons. These are capable of learning the functional dependencies based on the data. The data processing can be done parallelly. ANN resembles the brain through the learning process where knowledge is acquired by the networking environment and uses synaptic weights to store them.

Neuron consists of inputs, weights, activation function unit and the outputs. Figure 1 shows the architecture of the simple neuron. It contains P1, P2 & P3 as inputs and weights W1, W2 & W3 corresponding to the inputs. An activation function unit used to generate the output function. Weights corresponding to the inputs are multiplied parallely and summed to form the input for the activation unit. Then the activation unit checks to satisfy the training algorithm's specification that the neural network is attempted to run. The most commonly used activation functions are the linear activation functions like hard limit activation function, saturating linear activating function and hyperbolic tangent sigmoid activation function [5].

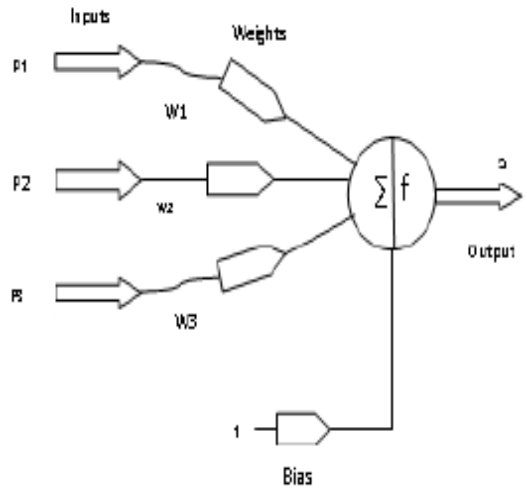


Figure 1: Simple neuron architecture [5]

The hardware implementation of the above model can be done using ASIC's or FPGA's. The latter is developed by the authors[4].The use of FPGA's extends more flexibility to the design and it maintains the high processing density with an improved performance. This helps to develop the parallel computation. The same principle can be extended to the multi layer perceptron network which clearly describes the ANN architecture for any of the signal processing applications. Any of the activation functions mentioned above could be used for the implementation.

MULTI-LAYER PERCEPTRON NETWORK

There may be many types of neural networks such as Probabilistic Neural Networks, General Regression Neural Networks, Radial Basis Function Networks, Adaline Networks, Hebb Networks, Hetero-associative Networks, Recurrent Networks, Hybrid Networks etc. The neural network without any qualification is called as Multilayer Perceptron network (MLP).It is one of the feed-forward artificial neural network models which map the several inputs to several outputs. In a MLP, the nodes are situated as multilayered and each layer is fully connected to the next layer. The processing element neuron or perceptron is connected to any of the activation function. For training the network, the structure utilizes a supervised learning technique called the back propagation. Figure(2) illustrates the 3-layered multilayer perceptrons.

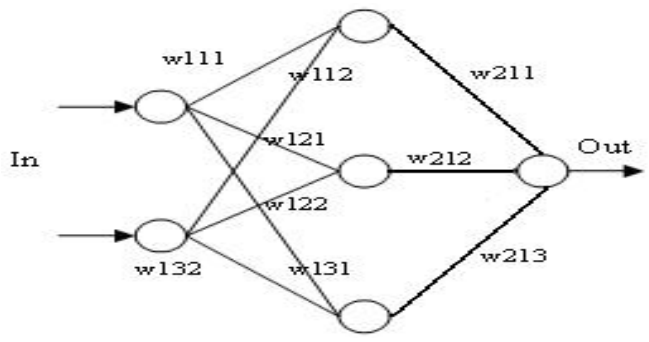


Figure 2 : MLP Network.

This network has 2 input layers,3 hidden layers and 1 output layer. The figure explains three-layer back propagation multilayer perceptrons. This concept is elaborated to three-layer MLP network structure as follows; Figure 3 represents the 3-layer MLP network structure [5].

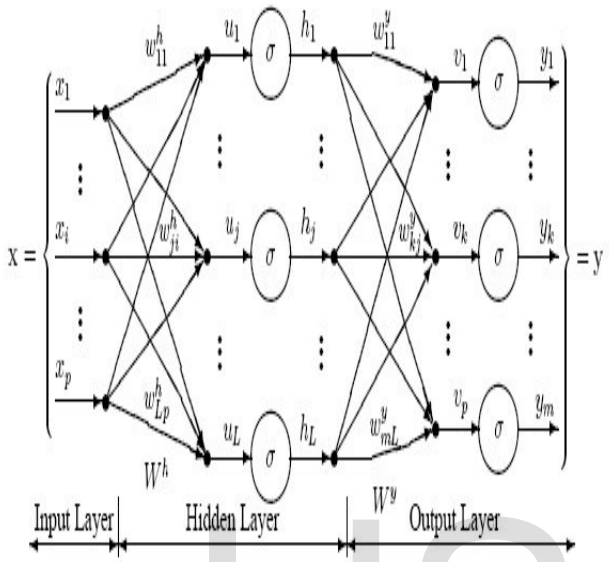


Figure 3: Three-layer Multilayer Perceptron [5].

For each input pattern, only one winning node is active and this is valid in solving most of the classification problems. Following assumptions were made:

- No connection between non-adjacent layers.
- No recurrent connections.
- The connections are defined by associated weights.
- The weighted sum of the neurons is computed by applying suitable activation functions.
- High or low neuron output will be generated.

ACTIVATION FUNCTIONS

Activation function is a function used to transform the activation level of a neuron into an output signal. There are many types of activation functions such as Identity function, bipolar step function, Binary step function, Sigmoidal function, Ramp function etc. Three types of non linear activation functions were implemented in this work using FPGA. Those are hyperbolic tangent sigmoid activation function, Saturating linear activation function, hard limiting activation function.

Hyperbolic tangent sigmoid activation function or tansig function is given by the following equation;

$$\alpha = e^{n}-e^{-n}/ e^{n}+e^{-n}$$

More commonly, it is used in MLP's since it is differentiable. The complexity is observed in implementation using digital hardware since it is an exponential series.

The mathematical model for a piecewise linear activation function is given by,

$$Y=f(u)=\begin{cases} -1 & \text{for } u<-1 \\ u & \text{for } -1<u<1 \\ 1 & \text{for } u>=1 \end{cases}$$

Also, the hard limiting activation function is given by,

$$Y=f(u)=\begin{cases} 0 & \text{if } u<0 \\ 1 & \text{if } u>0 \end{cases}$$

This type of function is used to create neurons and it is referred to as McCulloch-Pitts model.

RESULTS

Figure 4 represents the simulation results of ANN with all the activation functions used. ie,with tangent sigmoid activation function, piecewise linear function & hard limiting function. Also, the result without activation function is shown at the end.

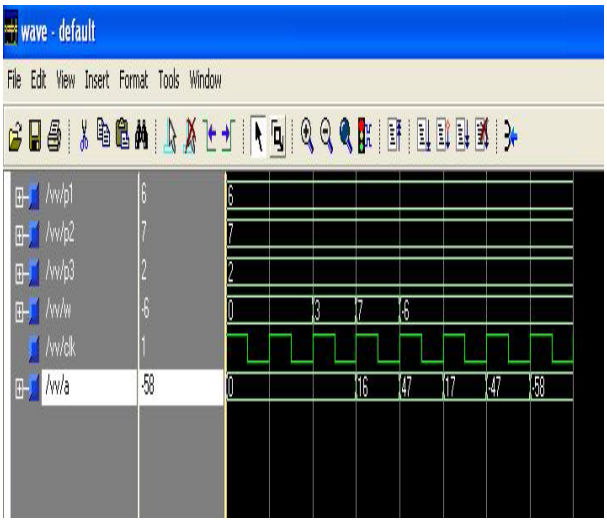
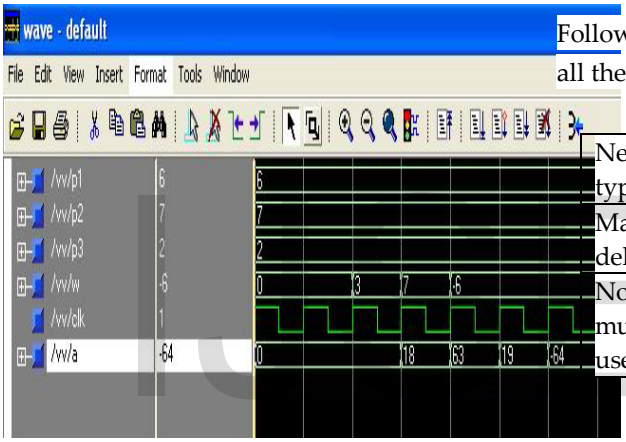


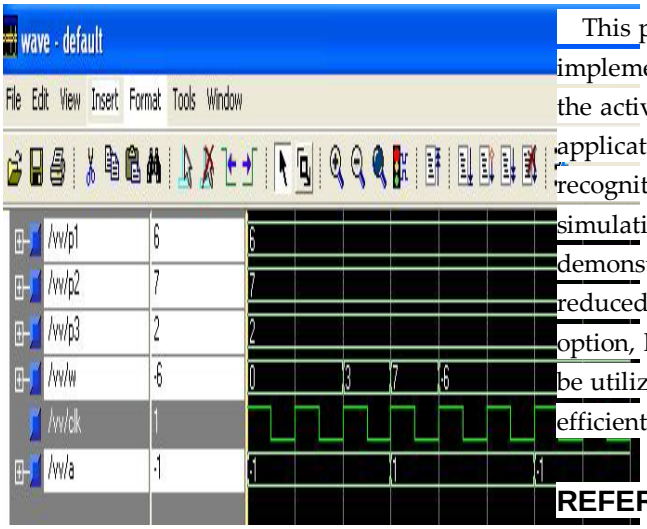
Figure 4: Results observed for various activation functions.



Following table describes the timing summary of all these activation functions.

Neuron type	Tangent Sigmoid	Piecewise linear	Hard limit
Max.Path delay	27.68ns	13.78ns	14.9ns
No. of multipliers used	3	2	2

CONCLUSION



This paper successfully explains the hardware implementation of MLP networks with any of the activation functions for the signal processing applications such as voice recognition, image recognition, and in case of adaptive filters. The simulation and comparison results were clearly demonstrated and the maximum path delay is reduced significantly. The re-programmability option, low cost and other features of FPGA can be utilized to implement ANN architecture in an efficient manner.

REFERENCES

[1]. Kalbande, Chandrashekhar, and Anil Bavaskar. "Implementation of FPGA based general purpose artificial neural network." *ITSI Transactions on Electrical and Electronics Engineering* (2013): 2320-8945.

[2]. Jain, Anil K., Jianchang Mao, and K. M. Mohinuddin. "Artificial neural networks: A tutorial." *Computer* 29.3 (1996): 31-44.

[3]. Sahin, Suhap, Yasar Becerikli, and Suleyman Yazici. "Neural network implementation in hardware using FPGAs." *Neural Information Processing*. Springer Berlin Heidelberg, 2006.

[4]. S N Prasad,S Y Kulkarni,"Low power adder based ANN", International Journal of Computer Applications (0975 – 8887) Volume 118 – No. 10, May 2015

[5]. Thilagavathy, A., and K. Vijaya Kanth. "Digital Hardware Implementation of Artificial Neural Network for Signal Processing." *International Journal of Scientific & Engineering Research*, Volume 4, Issue3, March-2013.

[6]. A. Cheang-Martinez, J. Acevedo-Davila, L. Torres-Trevino, F. Reyes Valdes, and A. Saldivar-Garcia, "Corrosion prediction and annual maintenance improvement of concrete structural components using neural networks," in *Electronics, Robotics and Automotive Mechanics Conference, 2007. CERMA 2007*, sept. 2007, pp. 202 –206.

[7]. Rafid Ahmed Khalil Sa'ad Ahmed AI-Kazzaz "Digital Hardware implementation of Artificial Neurons models using FPGA" , Springer U.S.,2012.

[8]. Neeraj Chasta , Sarita Chouhan and Yogesh Kumar, " Analog VLSI implementation of neural network architecture for signal processing" VLSICS., Vol.3.no.12,2012.