ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Special Issue 3, April 2017

Efficient Implementation of VLSI on Competitive Neural Network

M. Uma Maheswari [1], M. Hari Hara Bhuvanesh[2], M.G. Aravind Sai[3]

Assistant Professor, Dept. of ECE, Meenakshi College of Engineering, Chennai, India[1]

B.E, Dept. of ECE, Meenakshi College of Engineering, Chennai, India [2,3]

ABSTRACT-The power and usefulness of artificial neural networks have been demonstrated several applications including speech synthesis, diagnostic problems, medicine, signal processing, computer vision and problems that fall under the category of pattern recognition. VLSI implementation of Learning Vector Quantization, a type of neural network based on competition is analysed. Learning Vector Quantization is a pattern classification method in which each output unit represents a particular class. The similarity of input vectors with the weight vectors is checked using squared Euclidean distance computation and weight vectors are adjusted through supervised methodology. Here it is mainly focused on reducing the hardware utilization during the implementation of Learning Vector Quantization. The techniques used for carrying out multiplication are radix-4 booth multiplication and the Distributed Arithmetic Accumulation .Compare to Distributive Arithmetic technique, Booth multiplication provides better area and delay for Learning Vector Quantization implementation.

KEYWORDS-Neural Network, Competitive Learning, LVQ, Booth Multiplier, Distributed Arithmetic

I. INTRODUCTION

An artificial neuron network (ANN) is a computational model based on the structure and functions of biological neural networks. Information that flows through the network affects the structure of the ANN because a neural network changes - or learns, in a sense - based on that input and output. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. This is true of ANNs as well. A Pipelined architecture is used for competitive learning and the architecture is implemented by FPGA. It is used as hardware accelerator in a system on a programmable chip for reducing the computational time. To implement large and effective software neural networks, considerable processing and storage resources need to be committed– which can consume vast amounts of computer memory and hard disk space. Furthermore, the designer of neural network systems will often need to simulate the transmission of signals through many of these connections and their associated neurons – which must often be matched with incredible amounts of CPU processing power and time.Booth multiplier and Distributive Arithmetic technique is used instead of pipelined architecture which reduces the no of slices and area occupied in the Xilinx. Area occupied by the Xilinx is reduced .Hardware Utilization is reduced .No of slices in the Xilinx is reduced so that proportionally delay time is reduced.

II. RELATED WORK

In [1] An efficient pipeline architecture is designed based on the code word swapping scheme for enhancing the throughput. The CPU time of the NIOS processor executing the CL training with the proposed architecture as an accelerator is measured. Experiment results show that the CPU time is lower than that of other hardware or software implementations running the CL training program with or without the support of custom hardware. Keywords-*k*-winners-take-all; competitive learning; on-chip learning; reconfigurable computing; FPGA. In [2], We are using 180nm CMOS VLSI technology for implementing circuits which performs arithmetic operations and for implementing Neural Network. The arithmetic circuits presented here are based on MOS transistors operating in subthreshold region. The basic blocks of artificial neuron are multiplier, adder and neuron activation function. The functionality of designed neural network is verified for analog operations like signal amplification and frequency multiplication. The network designed can be adopted for digital operations like AND, OR and NOT. The network realizes its functionality for the

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Special Issue 3, April 2017

trained targets which is verified using simulation results. The schematic, Layout design and verification of proposed Neural Network is carried out using Cadence Virtuoso tool. In [3] Adaptive Competitive Neural Network not only groups similar input feature vectors together but also determines the appropriate number of groups of these vectors. This algorithm uses a new proposed criterion referred to as the ACL criterion. This criterion evaluates different clustering structures produced by the ACL neural network for an input data set. Then, it selects the best clustering structure and the corresponding network architecture for this data set. The selected structure is composed of the minimum number of clusters that are compact and balanced in their sizes. The selected network architecture is efficient, in terms of its complexity, as it contains the minimum number of neurons. Results show that the ACL algorithm is more accurate and robust in both determining the number of clusters and allocating input feature vectors into these clusters than the other algorithm especially with data sets that are sparsely distributed. In [4] The proposed algorithm, called quantum hamming neural network (QHNN), is capable to recognize incomplete patterns, as well as, increase the probability of recognizing patterns on the account of undesired patterns. Moreover, these undesired patterns could be used as new patterns for training the algorithm in subsequent steps. The proposed algorithm is testified via a case study and a classification experiment, where promising results, reaches to 100%, are given and compared favorably with other reported quantum competitive algorithms.

III. PROPOSED ALGORITHM

Competitive learning is a form of unsupervised learning in artificial neural networks, in which nodes compete for the right to respond to a subset of the input data. A variant of Hebbian learning, competitive learning works by increasing the specialization of each node in the network. It is well suited to finding clusters within data.

There are three basic elements to a competitive learning rule: [1] A set of neurons that are all the same except for some randomly distributed synaptic weights, and which therefore respond differently to a given set of input patterns. [2]A limit imposed on the "strength" of each neuron. [3].A mechanism that permits the neurons to compete for the right to respond to a given subset of inputs, such that only one output neuron (or only one neuron per group), is active (i.e. "on") at a time. The neuron that wins the competition is called a "Winner-Take-All" Neuron.

Accordingly, the individual neurons of the network learn to specialize on ensembles of similar patterns and in so doing become 'feature detectors' for different classes of input patterns .The fact that competitive networks recode sets of correlated inputs to one of a few output neurons essentially removes the redundancy in representation which is an essential part of processing in biological sensory systems.

For example consider,Let a set of sensors all feed into three different nodes, so that everyone is connected to every sensor.Let the weights that each node gives to its sensors be set randomly between 0.0 and 1.0. Let the output of each node be the sum of all its sensors, each sensor's signal strength being multiplied by its weight.When the net is shown an input, the node with the highest output is deemed the winner. The input is classified as being within the cluster corresponding to that node.The winner updates each of its weights, moving weight from the connections that gave it weaker signals to the connections that gave it stronger signals.Thus, as more data are received, each node converges on the center of the cluster that it has come to represent and activates more strongly for inputs in this cluster and more weakly for inputs in other clusters.The concept of multiplying the weight with the input is carried out using Booth Multiplication and and adding the sum of the products is carried out using Distributive Arithmetic.During training, the output unit that provides the highest activation to a given input pattern is declared the winner and is moved closer to the input pattern, whereas the rest of the neurons are left unchanged.This strategy is also called winner-take-all since only the winning neuron is updated.It is usually implemented with Neural Networks that contain a hidden layer which is commonly known as “competitive layer”.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Special Issue 3, April 2017

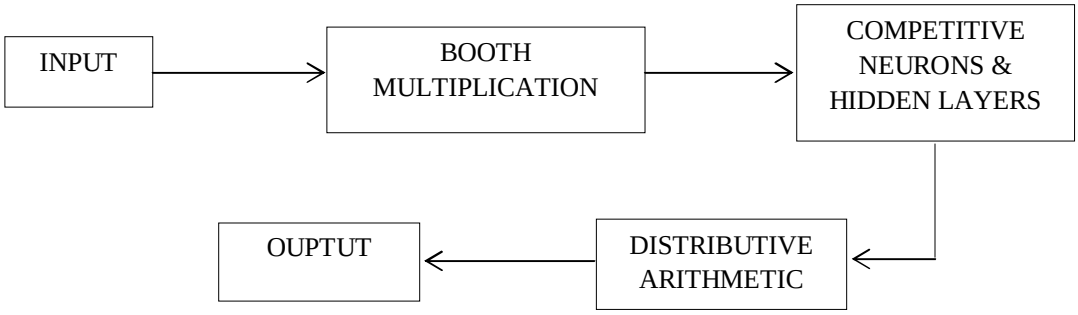

Fig.1 BLOCK DIAGRAM OF THE PROPOSED SYSTEM

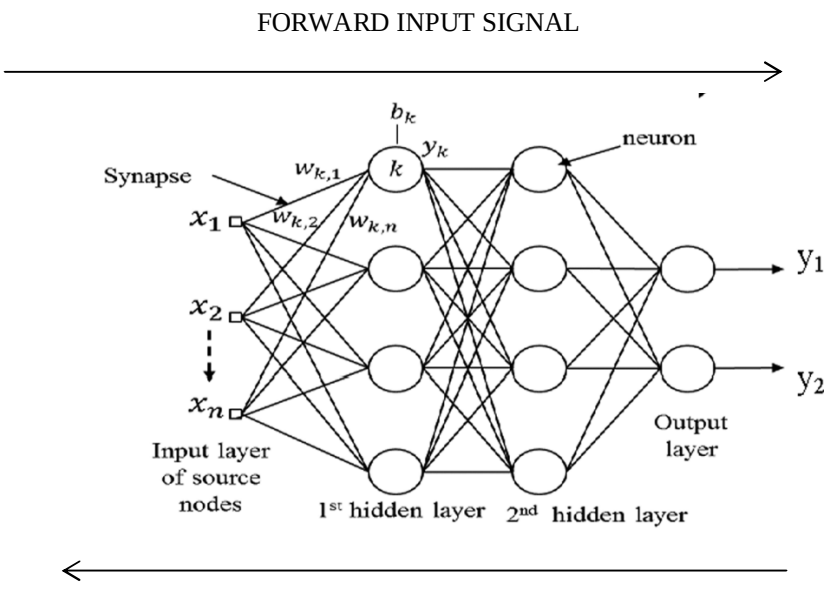

Fig. 2 Architecture of Competitive Neural Network

IX.SIMULATION RESULTS

The simulation studies involves the hardware utilization as shown in the Fig.1.The proposed system shows that the total number of slices that is used by the Xilinx is reduced as when compared with the existing system.We here instead of the pipelined architecture used the Booth Multiplier and Distributive Arithmetic to implement the competitive neural network.It is clear from the Fig.2 that the time is reduces in nanoseconds .Our result shows that the hardware is utilized and the time delay is reduce when we use Booth Multiplier and the Distributive Arithmetic instead of Pipelined Architecture.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Special Issue 3, April 2017

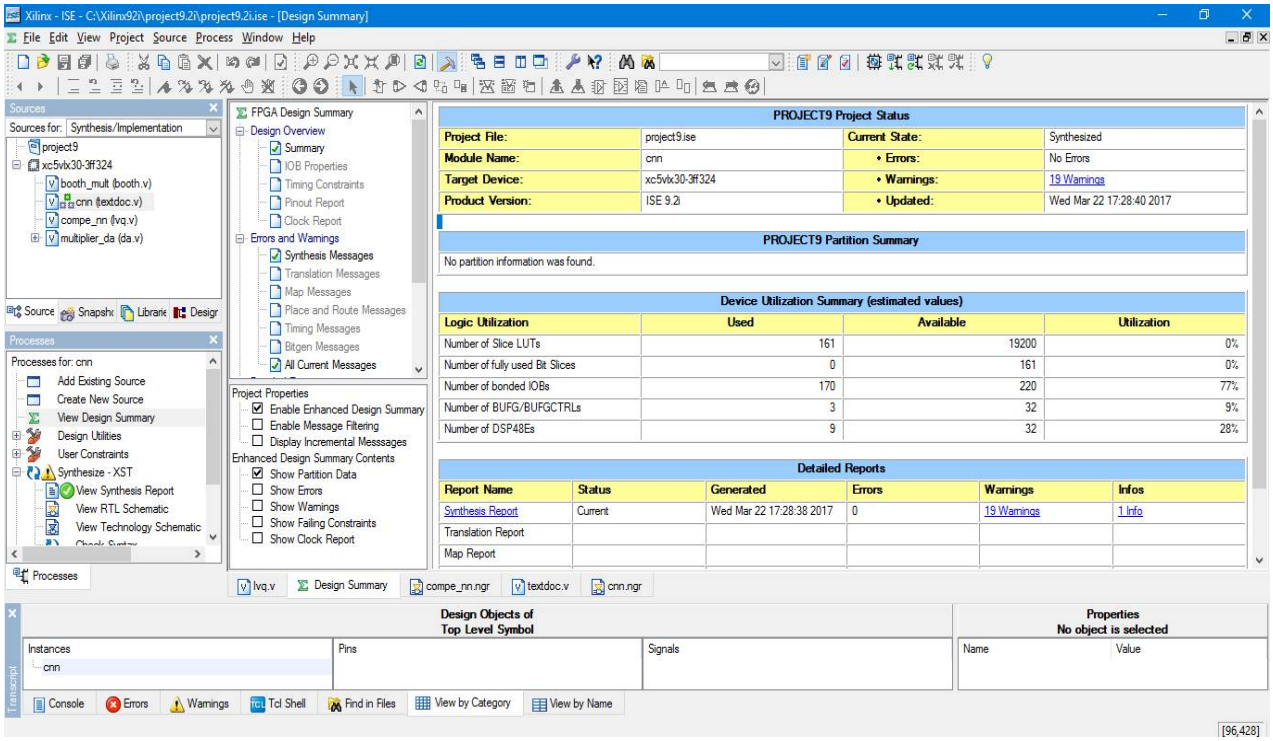

Fig. 1 Design Summary

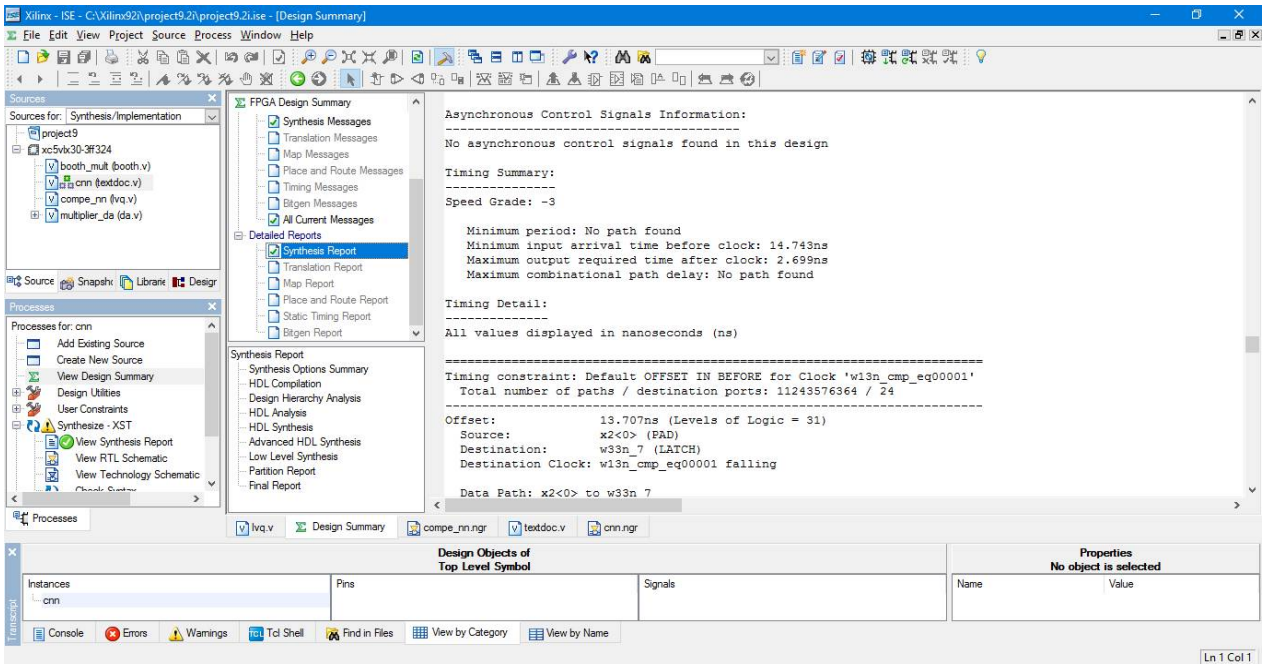

Fig. 2 Synthesis Report

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Special Issue 3, April 2017

X. CONCLUSION

A New method is proposed for implementing the VLSI on Competitve Neural Networks using Xilinx Software to reduce the hardware utilization and the area and the no.of slices occupied by it .The synthesis results confirms that the RDA design is capable of much higher order filters that it has the same time having similar equivalent gate counts and throughput. The techniques used for carrying out multiplication are radix-4 booth multiplication and the Distributed Arithmetic Accumulation for adding the products.

REFERENCES

1. Hui-Ya Li, Chien-Min Ou, Yi-Tsan Hung, Wen-Jyi Hwang, and Chia-Lung Hung (2010) "Hardware Implementation of *k*-Winner-Take-All Neural Network with On-Chip Learning" .
2. Prof.Bapuray, D.Yammenavar, Vadiraj.R.Gurunaik, Rakesh.N.Bevinagidad, Vinayak.U.Gandage (2011) "DESIGN AND ANALOG VLSI IMPLEMENTATION OF ARTIFICIAL NEURAL NETWORK" Vol.2, No.3, July .
3. Ahmad R. Abas (2013) “Adaptive Competitive Learning Neural Networks”.
4. Mohammed Ziad, Alaa Sagheer, Nasser Metwally (2015),“An Autonomous Competitive Learning Algorithm using Quantum Hamming Neural Network”.
5. Amos R. Omondi, Jagath C. Rajapakse (2006) “FPGA Implementations of Neural Networks ”.
6. A.H.Fetanat, Ezat mahmoud soleiman (2014) "Intrusion Detection System Using Supervised Learning Vector Quantization " Volume 1 Issue 10 November.
7. Brown 1992, "Field Programmable Gate Arrays" Kluwer Academic Publishers,volume 8,pp 68-127.
8. Deepali Chandel, GaganKumawat (2013) "Booth Multiplier: Ease of multiplication" Volume 3, Issue 3, March.
9. Dr. UdoSeiffert, Professor Lakhmi C. Jain (2002) "Self-Organizing Neural Networks Recent Advances and Applications" Volume 5, PP 13-41,volume 78.
10. H. C. Card, D. K. McNeill, C. R. Schneider (1998) "Analog VLSI Circuits for Competitive Learning Networks", March, Volume 15, Issue 3, pp 291-314.
11. Hui-Ya Li, Wen-Jyi Hwang, Chih-Chieh Hsu, and Chia-Lung Hung "Efficient K-Means VLSI Architecture for Vector Quantization".
12. J. Choi, B. J. Sheu (1993) "A High-Precision VLSI Winner-Take-All Circuit for Self-Organizing Neural Networks", IEEE Journal of Solid-State Circuits, 28, pp. 576-584.
13. J.LUBKIN AND G.CAUWENBERGHS (2000) "VLSI Implementation of Fuzzy Adaptive Resonance and Learning Vector Quantization" March 21.
14. Manish Chaudhary, Mandeep Singh Narula (2013) "FPGA Implementation of Booth’s Multiplier Using Verilog" Volume-3, Issue-1, June .
15. M.Biehl, A. Ghosh, and B. Hammer(2007) ”Dynamics and GeneralizationAbilityofLVQAlgorithms”,Journal of Machine Learning Research,pp 323-360 .
16. N.K. Bose, P. Liang (1996) "Neural network fundamentals with graphs, algorithms, and applications" Published by McGraw-Hill.
17. Peled and B. Liu (1974) “A New Hardware Realization of Digital Filters,” IEEE Transactions on ASSP, vol. 22, no.6, pp. 456-462.