

Design and Implementation of Neural Network Based circuits for VLSI testing

*K.P. Sridhar, B. Vignesh, S. Saravanan,
 M. Lavanya and V. Vaithiyathan*

School of Computing, SASTRA University, India

Abstract: Artificial Neural Network (ANN) plays a vital role in biologically inspired microcircuits, which is also known as the new trend of Very Large Scale Integration (VLSI) involvement of silicon based neurons. This paper proposes a new design methodology in VLSI testing using neural network. Hardware based circuit is constructed by utilizing the features of neural network and that circuit is also tested for fault free method. Experimental results are targeted to ISCAS85 Combinational Benchmark circuit. Implementation of proposed fault free circuit is written in VHDL and synthesised with XILINX Spartan III FPGA.

Key words: Artificial Neural Network • VLSI • VHDL • Combinational circuits • Fault free circuit • FPGA

INTRODUCTION

In basic, biological neuron structure consists of three different parts. First part is named as soma body portion of neuron and second part is known as dendrites, which connect the different neuron to soma for transfers action potential and finally axon is used to transmit the action potential from soma to other neuron or cells or nodes. Hence, soma receives input from others neurons across the different dendrites and perform addition of input at certain point and generate spike signals. This spike should propagate to other neuron through axon. The addition process takes places based on their weight of different dendrites. Higher weight of dendrites will transfer very first. The neuron which receives input from two dendrites $d=1$ and $d=2$, at the moment that a spike is received are named as postsynaptic. As the spike from the second presynaptic neuron arrives, another postsynaptic is generated that is added to the first one. Finally, after other postsynaptic are generated by the arrival of other spikes, the addition of the input potentials is over the threshold and as a consequence, an action potential is triggered and produce large positive spikes.

From Very Large Scale Integration (VLSI) circuit point of view, spikes are considered as pulses. Spikes are short electrical pulse, which is having amplitude and time duration based on clock pulses. Dendrites and Axon are taken as input and output of node. Signals are continuous

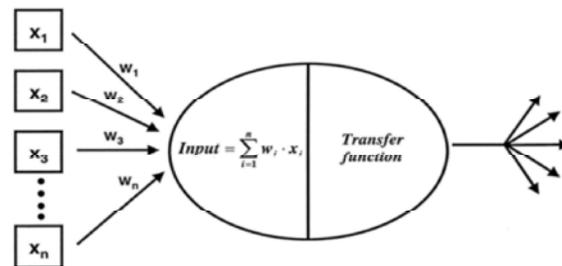


Fig. 1: Artificial neuron with inputs $(x_1, x_2, x_3, \dots, x_n)$ and Weights $(w_1, w_2, w_3, \dots, w_n)$

values that represent the neuron firing rate. Neurons implement a saturating non-linearity transfer function on the input weighted sum, the synapse implements a multiplication between the neuron's input signal and its corresponding synaptic weight are as shown in figure 1. Neuron will be activated when it has output of logic '1' and in other remaining cases it tends to '0'. Dendrite of neuron is considered as fan-in for node and axon is considered as fan-out. Node will burst firing patterns and reproduces spiking pulses in ANN based combinational circuit. Proposed neuron method is targeted to ISCAS85-C17 benchmark circuit [8], which is one of the scientific communities to test and compare ANN networks.

Definition: Fan-Out-Different neuron sharing one source neuron which means that a net propagates a signal from one source to "n" destinations.

Fan-In-The fan in node of a neuron x are the source nodes of edges sharing “ x ” as destination.

In section 2 an overview of existing methods is discussed. In section 3, proposed method is focused on VLSI based single neuron architecture with multiple inputs and one output. ISCAS85-C17 benchmark circuit is also constructed using NN for testing. Experimental results were produced in section 4. This section also shows how proposed design is targeted to XILINX Spartan III FPGA and its report is tabulated. In section 5 conclusions is provided.

MATERIALS AND METHODS

Various research works are focused in development of achieving neural network based circuits in VLSI. In paper [1], back propagation firing is presented with help of neuron inputs and it is multiplied with its corresponding input and weight inputs. Final result are summed together to get a multiple output. Each output signal is compared with the target signal which is present in error generator. This also back propagates the matched output signal weight to the previous layer of neuron weight.

ANN illustrated by electrical circuit functionality is discussed in paper [2], where modelling with Cadence tool using Verilog is done. In this method the current circuit is represented as an axon where a transfer parameter of each neuron and voltage is represented as dendrites of a cell. In paper [3], development of very efficient and versatile hamming ANN is proposed. This design is targeted to 0.35 μm and double poly silicon CMOS technology. It's feed forward neural network has the ability to classify the neuron based on hamming distance of previously stored input neuron.

In paper [4], testing of ANN is implemented with help of neural network and it is represented as hop field network. It has feedback loop from the output to the input, which gathers associate neuron in the brain which reduce time when compared to back propagation method. VLSI hardware emulation in paper [5] shows emulation in parallel algorithm for test pattern generation based neural network. Mapping is done with neural network to VLSI array and programmable cells with shift register. Delays and phase shift in neural network can be reduced by optimization modelling which leads to effective network as in paper [6]. Implementing ANN in FPGA by using multiplexing helps us to reduce the cost of resource requirement with moderate overhead of speed as discussed in paper [7].

In paper [8], shows the detection of fault in power distribution networks using probabilistic neuron logic. The presence of artificial neural network in science and management are shown in paper [9,10,11]. Thus various existing system shows that biological circuits promises for linking VLSI domain with neural network. This paper is also one of the attempts to strengthen VLSI testing based on neural network.

Proposed Method: One of the important characteristics of ANN network is inducing the modification behaviour with existing neuron connections. The process of modification behaviour implies some values which changes in that connection. The use of weights is very important because ANN learns modifying the values of the weights of the network. Here weights are associated to the synapses and can increase or decrease the signals that arrive to a synapse. When a synapse arrives, the neuron has basically two functions, to generate the potential action according to the input and to compare if the addition of all potential actions in this instance of time is over a threshold, in that case, it will have to generate a pulse through the axon.

Proposed design is divided into four major units such as Input Unit, Firing Unit, Control Unit and Output Unit as shown in figure 2. Our proposed design consists of 5 basic inputs. They are IN1 and IN2 referred as inputs, W1 and W2 referred as dendrites from others neurons along with relation weight and finally a clock signal. All the input signals are in digital, so that it is recommended to our overall system for synchronization. One output is referred to the axon. When the output would be fired, the output will have to generate a pulse to propagate it to other neuron through the axon. The aim of the processor unit of a neuron is to generate a potential according to the arrival of spikes and evaluate these potential to generate an output potential.

Firing Unit-This unit is determinate by the exponential function as $f(x) = x \times e^{-x}$, where x is known as input. Hence the neuron is said to be fired or potentially activated.

Control Unit-Inputs IN1 and IN2 were summed together and as a result, pulse is generated. If the potential is over the threshold, then voltage returns a value and the resting potential is activated for refractory time.

Input Unit-Initial unit consists of input IN1 and IN2 with control clock. Timer concept is also used in this unit. During refractory time, the neuron ignores the arrival of other spikes from other cells. To do this work, this timer wants to wait for arrival of pulses in 2 mille second.

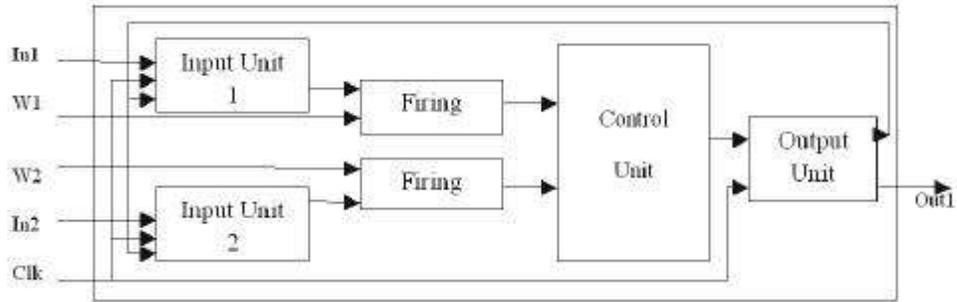


Fig. 2: VLSI based Single neuron network

In1- Input1 (dendrites d=1 from the cell 1)
 In2- Input2 (dendrites d=2 from the cell 2)
 Out1- Output1 (axon to other cells)

W1- Weight 1 for cell 1
 W2- Weight 1 for cell 2

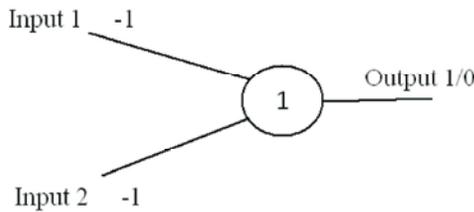


Fig. 3: Neural network with 2-input NAND gate

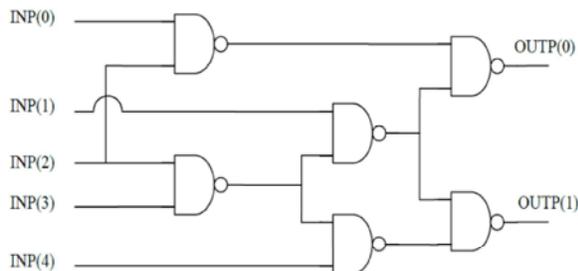


Fig. 4: ISCAS85-C17 Benchmark circuit

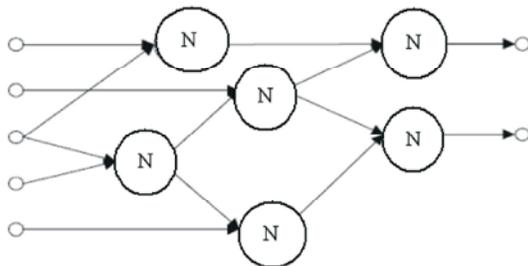


Fig. 5: Graphical representation of ISCAS85-C17 neuron

Output Unit-It performs two functions. In the first function, it produces the output for our global design and in the next function it is based on the threshold value and it predicts the output value of enable signal, which is one of the inputs from timer block.

Two input NAND gate's neural network and its corresponding pseudo code is given in figure 3.

Figure 4 shows ISCAS85-C17 benchmark circuit [8], which consists of 6 numbers of 2-input NAND gate. Its corresponding neural network graph representation is shown in figure 5.

Pseudo Code:

```

Fire= 2 +Σ;
If(Fire)
Begin
{
Neuron = 1;
Else
Neuron =0;
}
End
End if
    
```

RESULTS AND DISCUSSION

Proposed hardware based neuron design is implemented with VHDL code. Two different types of neurons are experimented for simulation. In the first type-two input neuron is constructed with hardware code and its simulation result is produced in figure 6. Here two neurons are taken as input (neuron1 and neuron2) and output pulse is generated based on the result of threshold value of control unit block. Weights were calculated through weight1 and weight2 signals.

Now we are going to simulate the result for ISCAS85-C17 [8] neuron Architecture using VHDL code as shown in figure 7. In this architecture five neurons are taken as input along with clock which possesses two output pulse named as pulse 1 and pulse 2 which looks like an ISCAS85-C17 [8] NAND benchmark circuit. The output pulse is generated when control unit block has the threshold voltage which is more than 10 micron volt.

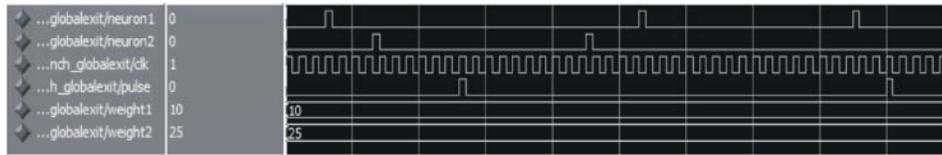


Fig. 6: Simulation Result for 2 input Neuron

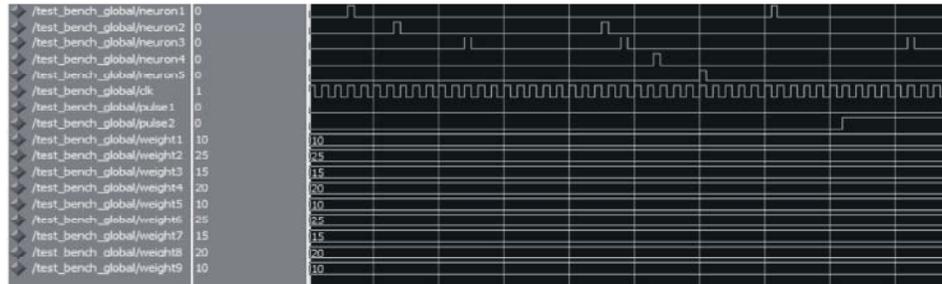


Fig. 7: Simulation Result of ISCAS85-C17 Benchmark circuit neuron

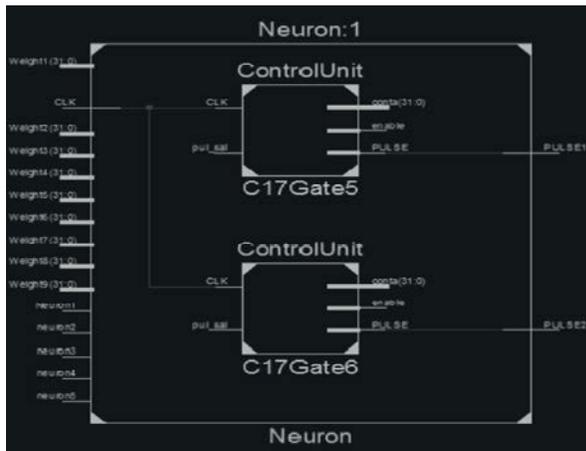


Fig. 8: RTL Schematic view of ISCAS85-C17 neuron architecture

The advantage of this architecture is by using VHDL component instantiation, it is possible to increase the input neuron for maximum level. By this result, we can be able to implement the benchmark circuit using 5 neuron (neuron1 to neuron5) and 9 weight (weight1 to weight9) values.

Fault free test pattern [12] is applied to this proposed method and its corresponding output result is also tabulated in Table 1. Synthesis report of the proposed method is given in Table 2. This report contains the usage of multipliers for firing unit, adder / subtractor for input unit and comparator for control unit. Utilization of register and Input Outputs (IO) are also shown in this report. CPU timing of this proposed method is also calculated.

RTL schematic view of ISCAS85-C17 [13] neuron architecture is shown in figure 8, which is targeted to XILINX tool in Spartans III FPGA.

Table 1: Fault free test pattern for ISCAS85-C17

Inputs (neuron1 to neuron5)	Outputs(pulse1 and pulse2)
10110	10
00111	00
10001	11
11011	11
01100	11
11110	10
10000	00

Table 2: Synthesis report of proposed method

Multipliers	12
Adders/Subtractors	6
Registers	384
Comparators	6
Design Statistics: IOs	296
Total CPU time to Xst completion	2.78 secs

CONCLUSION

The aim of the proposed method is to have the possibility of interconnect more number of artificial neurons to create a complete neuronal network in VLSI design testing. The emulation procedure of biological neuron was implemented for single NAND gate and ISCAS85-C17 circuit. It is also verified with fault free test patterns to prove that the circuit is tested and it is in error free. Proposed design is targeted to XILINX Spartan III FPGA and its simulation and synthesis report is tabulated. Thus these works show how neural network can be adoptable in VLSI based design testing.

REFERENCES

1. Neelmani Chhedaiya and Vishal Moyal, 2012. Implementation of Back Propagation Algorithm in Verilog, *Interantional Journal of Computer Technology and Applications*, pp: 340-343.
2. Elitsa Emilova Gieva, Rostislav Pavlov Rusev, Rosen Ivanov Radonov, Tihomir Borisov Takov and Marin Hristov Hristov, 2011. Verilog-A Behavioral Model of Hydrogen Bonding Network, *Annual Journal of Electronics*.
3. Stephane Badel, Alexandre Schmid and Yusuf Leblebici, 2003. A VLSI Hamming Artificial Neural Network with k Winner-Take-All and k Loser-Take-All, *Capability International Joint conference on Neural Networks*, Portland.
4. Priyanka Sinha, *Neural Network Automatic Test Pattern Generator*.
5. Morteza Fayyazi and Zainalabedin Navabi, 2000. Accelerating Test Generation by VLSI Hardware Emulation, *The 9th IEEE North Atlantic Test Workshop*.
6. Peter, B.L. Meijer, 2001. *Neural Networks for Device and Circuit Modelling*, *Lecture notes in computing science and engineering*, pp: 251-258.
7. Asha, R., P. Bowrna and F. Mhaboobkhan, 2013. Implementation of Feed forward Neural network using layer multiplexing for effective resource utilization in FPGA, *International journal of research in Engineering and Advanced Technology*.
8. Marizan bin Sulaiman, Adnan H. Tawafan and Zulkiflie Bin Ibrahim, 2013. Using Probabilistic Neural Network for Classification High Impedance Faults on Power Distribution Feeders. *World Applied Sciences Journal*, 23(10): 1274-1283.
9. Norizan Mohamed, Wan Muhamad, Amir W. Ahmad, Nor Azlida Aleng and Maizah Hura Ahmad, 2013. Modeling Multi Layer Feed-Forward Neural Network Model on the Influence of Hypertension and Diabetes Mellitus on Family History of Heart Attack in Male Patients. *World Applied Sciences Journal*, 21(Special Issue of Applied Math): 82-85.
10. Hossein Hakimpoor, Khairil Anuar Bin Arshad, Huam Hon Tat, Naser Khani and Mohsen Rahmandoust, 2011. Artificial Neural Networks' Applications in Management. *World Applied Sciences Journal*, 14(7): 1008-1019.
11. Ibrahiem, M.M. El Emary and S. Ramakrishnan, 2008. On the Application of Various Probabilistic Neural Networks in Solving Different Pattern Classification Problems. *World Applied Sciences Journal*, 4(6): 772-780.
12. Hamzaoglu, L. and J.K. Patel, 1998. Test set compaction algorithms for combinational circuits, *Proceedings of the CAD*.
13. HITEC/ PROOFS Fault simulator tool.