FPGA Modeling Of Neuron for Future Artificial Intelligence Applications

S. Sai Sree Andal[1] (M.Tech), N. Aravind[2], Asst. Prof.

[1, 2,] *Department of ECE, Teegala Krishna Reddy Engineering College/JNTU, India*

ABSTRACT:

An Artificial Neural Network, often just called a neural network, is a mathematical model inspired by biological neural networks. A neural network consists of an interconnected group of artificial Neurons. An artificial neuron is a mathematical function conceived as a crude model, or abstraction of biological neurons. This project describes a system realization of translating data from electrochemical sensor for neuron to process on FPGA. The structure of a neuron is split into various sub blocks and these blocks will be implemented individually first and then they are integrated to form the entire neuron. This project will be implemented in three stages. First we have to convert the analog signal coming from the analog circuitry using an ADC (analog to digital converter). In this project the 12 bit ADC chip will be used to convert analog signal from 4 channel analog circuitry to digital. The next module is the design of mathematical operation. This includes issues relating to data structure, design of Multiplier Accumulator (MAC) and activation function implementation. The final module is displaying the result from the data that have been accumulated by the neuron. The proposed architecture is simulated using Modelsim and synthesized using Xilinx ISE and it will be implemented on FPGA board for hardware implementation and testing. The Xilinx Chip scope tool will be used to test the FPGA inside results while the logic running on FPGA.

KEYWORDS: *Synaptic weights, Activation function, Electrochemical sensors, Bias*

1. INTRODUCTION

Electrochemical sensors are often used to determine concentrations of various analytes in testing samples such as fluids and dissolved solid materials. Electrochemical sensors are frequently used in occupational safety, medical engineering, process measuring engineering, environmental analysis.ANN is known to be able to improve electrochemical sensor this signal interpretation. In general, hardware realization requires a good compromise between accuracy and complexity of the processing units to allow a low cost effective device. In our project we are going to implement an artificial neuron with four inputs. The artificial neuron which we are implementing in our project is a prototype of the biological neuron. Basically a neuron consists of N inputs coming from dendrites get multiplied by the synaptic weights and then they are processed by soma .Depending on the strength of the input signals the neuron gets fired. Similarly we are going to implement an artificial neuron with 4 inputs. The Inputs are provided by an analog circuitry which has four channels. This analog circuitry can provide the Analog voltages for 4 channels .This acts as input source for our artificial neuron. Since our FPGA cannot deal with the analog voltages, we convert the analog input into digital output. For this purpose we are going to use 2 ADC modules in our project. The digital platform is Field Programmable Grid Array (FPGA).The approach for this project can be represented in block diagram as shown in the Fig.1. The key issue in designing this system is modular design for reconfigurability. The first issue is to convert the signal from an analog to a digital form, by sampling it using an analog-to-digital converter (ADC), which turns the analog signal into a stream of numbers.

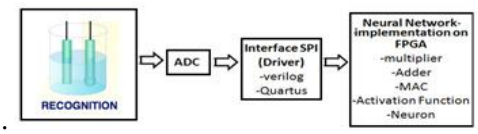

Fig.1.General flow of the project system linking applied chemical sensor to digital processing

The next module is the design of mathematical operation. This includes issues relating to data structure, design of Multiplier Accumulator (MAC) and activation function implementation. The final module is displaying the result from the data that have been accumulating by the neuron. Chipscope tool is used to view the Results after dumping the bit file into the FPGA.

II.DESIGN AND METHODOLOGYS

This section presents the design of the sub modules in implementing Fig 1. This covers the interfacing issues such as analog to digital implementation, data structure and the neuron architecture topology.

2.1.Analog to Digital Interfacing

In this project the 12 bit ADC 7476 were used to convert analog signal from electrochemical sensor to digital. It is a successive approximation 12-bit A/D converter with onboard sample and hold circuitry. As we know that the analog to digital converter is used to convert the analog signal into the digital samples. And we are using the 12 bit A/D converter means at the ADC output we get the 12 bit sample values of the giving a message signal (Modulating signal) through ADC. It is 2-channel ADC means we can give 2 inputs at a time and here we are connecting the input to the ADC from the Function Generator by giving the frequency levels and selecting a wave(Sine or Sawtooth, Square..etc) and constant voltage levels.. Communication with the device is done using a simple serial interface compatible with the SPI protocol. SPI is an interface that allows one chip to communicate with one or more other chips and in this case is ADC 7476 with the FPGA-Spartan 3E board.

The SPI algorithm is required to be implemented in hardware description language (HDL) on FPGA.Fig.2. demonstrates SPI interfacing that allows one chip to communicate with one or more other chips. As shown in the figure above the wires are called SCK, MOSI, MISO and SSEL, and one of the chip is called the SPI master, while the other the SPI slave. A clock is generated by the master, and one bit of data is transferred each time the clock toggles. Data is serialized before being transmitted, so that it fits on a single wire. There are two wires for data, one for each direction. The master and slave know beforehand the details of the communication (bit order, length of data words exchanged, etc...). The master is the one who initiates communication. Because SPI is synchronous and full-duplex, every time the clock toggles, two bits are actually transmitted (one in each direction). In term of performance, SPI can easily achieve a few Mbps (mega-bits-per-seconds) [4]. For this module, the approach taken is hardware implementation of existing technique, tailored to 12-bit environment.

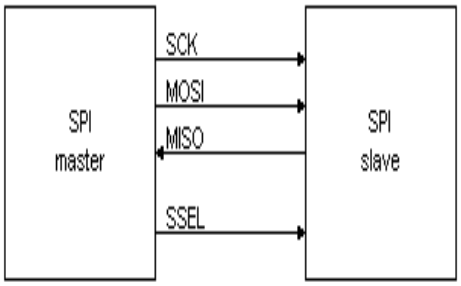

Fig.2.SPI Interfacing applied to signal

2.2. Digital Design: Data Structure and Modules for Neuron

In this section, there are 2 major parts: data structure and digital modules for neuron design on FPGA. For design tools, Modelsim to simulate the design at multiple stages throughout the design process and Quartus to program the board are used. Generally, a data structure is a particular way of storing and organizing data in a computer so that it can be used efficiently. Data structures are generally based on the ability of a computer/chip to fetch and store data at any place in its memory, specified by an address that can be manipulated by the program. For this project, the data computed from ADC will be converted into fixed-point number representation.

Fixed-point DSPs use 2's complement fixed-point numbers in different Q formats. Among the major issues in data structure is the conversion technique of fixed-point number from a Q format to an integer value so that it can be stored in memory and recognized by simulator. It is also required to keep track of the position of the binary point when manipulating fixed- point numbers in writing verilog codes The DSP (Digital Signal Processing) flows throughout the conversion to Q format representation are shown in the Fig. 3. As shown in the flowchart, a fractional number is converted to an integer value that can be recognized by a DSP assembler using the Q15 format .The number is first normalize then scaled down by 2 to the appropriate value that can be accommodated by the bits number. Finally, the value will be rounded (truncated) to integer value and be represented in binary number. A neuron can be viewed as processing data in three steps; the weighting of its input values, the summation of them all and their filtering by a activation function.

The Neuron can be expressed by the following equation:

$$y_j = f\left(\sum_i w_{ij}\, x_i - \theta_j\right) \qquad (1)$$

where y is the output of the neuron, w is the synaptic weight, x is the input and θ is the bias. The subscript i denotes the preceding neuron and j the neuron considered. The neuron computes the product of its inputs, with the corresponding synaptic weights, and then the results are added. The result is presented to a comparison unit designed to represent an appropriate activation function such as linear, sigmoid or hyperbolic tangent. The equation is shown in block diagrams in Fig.4. For the weighted inputs to be calculated in parallel using conventional design techniques, a large number of multiplier units would be required. To avoid this, multiplier/Accumulator architecture has been selected. It takes the input serially, multiplies them with the corresponding weight and accumulates their sum in a register.

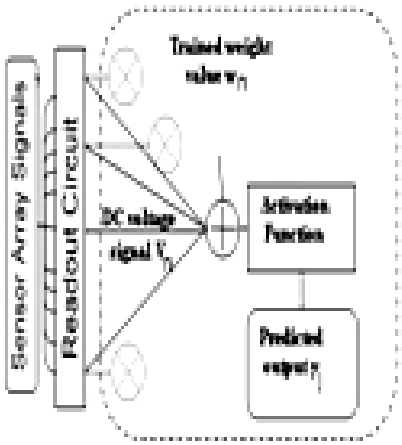

fig.4., Structure of neuron

The block diagram and flow of the hardware implementation is shown in Fig. 5 and 6. The accumulator unit is composed of a bit-serial adder and 16 bit register. The design of multiplier accumulator consists of adder and multiplier. MAC are frequently used in general computing and are especially critical to performance of digital signal processing applications. The MAC typically operate on a digital, and usually binary, multiplier quantity and a corresponding digital multiplicand quantity and generate a binary product. The design of multiplier accumulator proposed in this project consists of adder and multiplier that can accommodate or handle 4 channel of input (array of sensor).

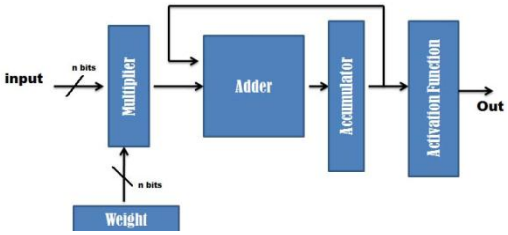

Fig.5:.The flow of proposed neuron architecture

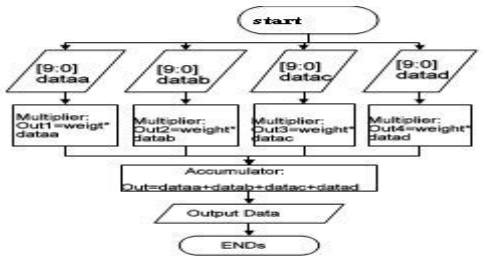

Fig.6:.Signal handling of multiplier accumulator

The architecture for the MAC is shown in Fig. 6. With tree configuration as shown in Fig.7, the use of tile logic is quite uneven and less efficient than with a chain. The idea of this configuration is that the 2 value from multiplier were added separately. The partition of the computation is then added at adder4 for the final output. Activation function in a back propagation network defines the way to obtain output of a neuron given the collective input from source synapses. The back propagation algorithm requires the activation function to be continuous and differentiable. It is desirable to have an activation function with its derivative easy to compute. The mathematical algorithm for tanh approximation using Taylor's Series expansion that is used in the hardware calculation is provided by equation 2: The design flow is presented in Fig 8.

$$y = x - x^3/3 + 2x^5/15+... \qquad (2)$$

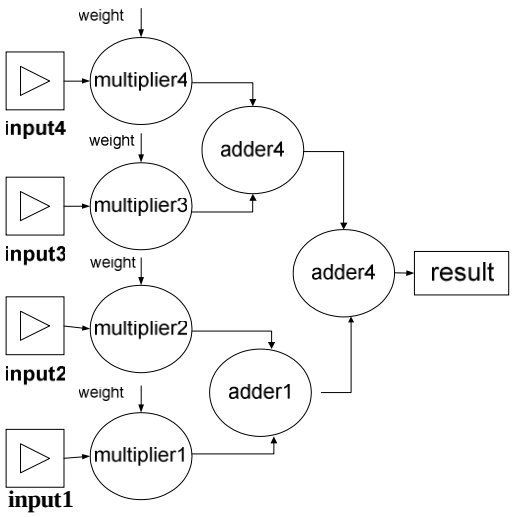

Fig.7:Tree configuration of Multiplier Accumulator

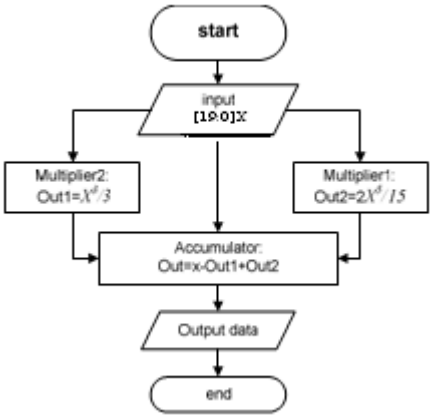

Fig.8:.Design flow of activation function

III. RESULTS AND DISCUSSION

3.1 : Simulation Results:

The following chapter consists of all the software and hardware results observed in the project. The results include snapshots of top module with the inputs, outputs and intermediate waveforms**.**

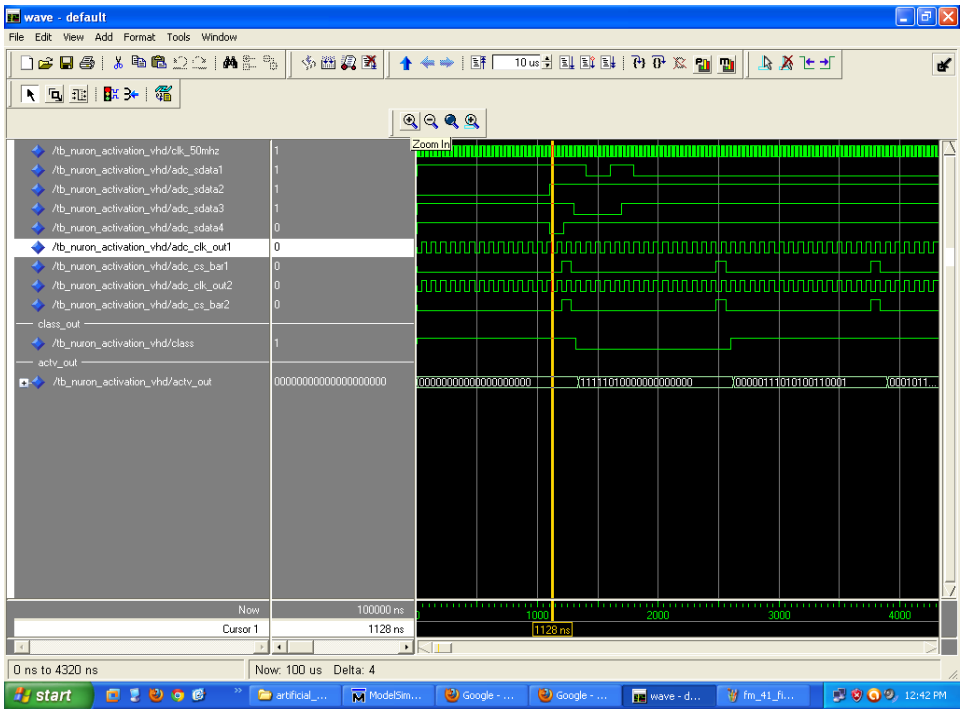

Fig.9.:Top Module simulation Results

3.2 . Chipscope Results:

Chipscope tool is used to view the Results after dumping the bit file into the FPGA. We need ICON(integrated Controller) and ILA(integrated logic analyzer) cores in order to run the chip scope tool. Chip Scope is an embedded, software based logic analyzer. By inserting an "integrated controller core" (icon) and an "integrated logic analyzer" (ila) into your design and connecting them properly, you can monitor any or all of the signals in your design. Chip Scope provides you with a convenient software based interface for controlling the "integrated logic analyzer," including setting the triggering options and viewing the waveforms.

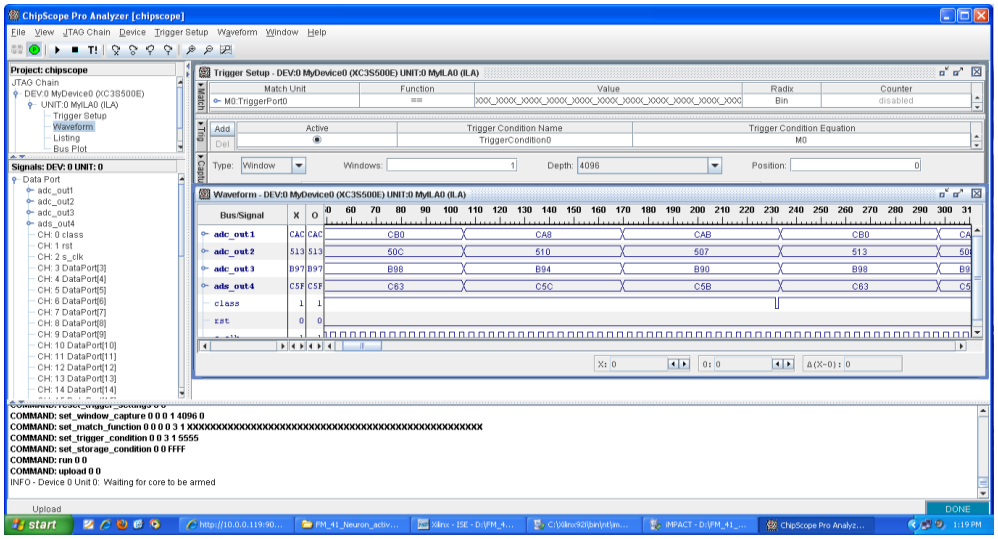

Fig.10.:Chipscope results

IV. CONCLUSION

The basic behaviour of the biological neuron can be emulated in an artificial neuron. A biological neuron with their dendrites, soma and axon can be characterized in an artificial neuron as a black box with inputs and an output. To implement the system the electronic pulses or spikes transmitted through neurons are replaced by digital signals or pulses. With all these things we get an electronic system that reproduces the behaviour of the biological neuron. The aim is to have the possibility of interconnect more of these artificial neurons to create a complete neuronal network. In this thesis work we only have focused our efforts to create an only artificial neuron. To complete the global system is not too complicate, to interconnect the artificial neuron with other neuron; the programmer should only connect the output of our neuron with the input of the next neuron, and in this way, with the other previous and next neurons. Besides, the programmer should codify a program that governs the relations between the different weights of the neuronal network, but this part is out of our analysis. With the VHDL code generated a FPGA can be programmed. Depending of the capacity of the FPGA used, a larger or less number of neuron can be programmed, depending on the same way of the number of interconnections of each neuron. Therefore, according to the results obtained, we can say that we have designed a system whose performance leads to the achievement of the objectives of the project and at the same time could work as the main base to develop future applications.

V. REFERENCES

[1] Wan Fazlida Hanim Abdullah., Masuri Othman, Mohd Alaudin Mohd Ali,and Md. Shabiul Islam. Improving ion-sensitive field-effect transistor selectivity with back propagation neural network. WSEAS Transactions of Circuits and Systems. 9(11): 700-712,2010

[2] Stradiotto, N. R., Yamanaka, H., & Zanoni, M. V. B. Electrochemical sensors: a powerful tool in analytical chemistry. Journal of the Brazilian Chemical Society 14: 159-173, 2003

[3] FPGA Implementation of a Multilayer Perceptron Neural Network using VHDL, Yamina TARIGHT, Michel HUBIN, Proceedings of ICSP '98 [page1-3]

[4] 'Overview and Use of the PICmicro Serial Peripheral Interface', Microchip(TM),[page 1-9]

[5] Poole,D., Linear algebra : a modern introduction. Belmont, CA, Thomson Brooks/Cole, 2005

[6] Polikar, R. Ensemble based systems in decision Making. IEEE Circuits and systems Magazine6(3)21-45,2006.

[7] A.Durg,W.V.Stoelker,J.P.Cookson,S.E.Umbaugh and R.H.Moss,"Identification of Variegating Coloring in Skin Tumors:Neural Network vs Rule Based Induction methods",IEEE Eng. in med. and Biol., vol.12 pp.71-74 & 98,1993

[8] FPGA Implementation of Artificial Neural Networks: An application on Medical ExpertSystems,G.P.K Economou,E.P.Mariatos, N.M.Economopoulos, D.Lymberopoulos, and C.E. Goutis,Department of Electrical Engineering University of Patras, GR 261 10,Patras, Greece

[9] A.R Ormondi and J.C Rajapakse, FPGA Implementation of Neural Network, [page 271-296],2006

[10] Simon Haykin 'Neural Networks and Learning Machines, third edition, [page 40-45]

[11] Benard Widrow, David E. Rumelhart, and Michael A. Lehr, "Neural networks: Applications in industry, business and science," Communications of the ACM,vol. 37, no. 3, pp. 93–105, Mar. 1994.

[12] Altera Corporation 'Cyclone III Device Handbook, Volume, 1',[chapter5,page 1-8], July 2007

S. Sai Sree Andal[1] (M.Tech),

N. Aravind[2], Asst. Prof