Design of Neural Architecture in 0.35µmTechnology Using Analog VLSI.

[1] Mr.Maulik B.Rami , [2] Prof.H.G.Bhatt , [3]Prof.Y.B.shukla

[1]PG student(EC), Electronics and Communication Department S.V.I.T., Vasad-Gujarat.

[2], [3]Associate professor, Electronics and Communication Department, S.P.C.E., Visnagar-Gujarat.

Abstract

Artificial neural network are attempts to mimic, at least partially, the structure and function of brains and nervous systems. The human brain contains billion of biological neurons whose manner of interconnection allow us the reason, memorize and compute. Advances in VLSI technology and demand for intelligent machines have created strong resurgence of interest in emulating neural system for real time applications. Such an artificial neural network can be built with help of simple analog components like MOSFET circuits and basic circuits with help of operational amplifier. This paper gives information about neuron behavior and how it takes intelligent decision.

Index Terms - Introduction, Architecture of neuron, Analog Neural components, Simulation Results, Conclusion& future work.

I. INTRODUCTION

Neural networks are used when there is no algorithmic solution to a problem or a problem is too complicated to be solved by known algorithms Neural networks can be used when the definition of the problem does not exist, but the samples of inputs and corresponding outputs are available.if we are ever going to understand intelligence and develop artificially intelligent machines or computers, we need to study the brain and its neurons, and how neurons work together to solve problems.It is not useful consider neural networks to solve problems for which the analytical solution can be easily found and implemented. In that case, the corresponding neural implementation will be generally larger and less accurate than the direct algorithmic solution.[2]

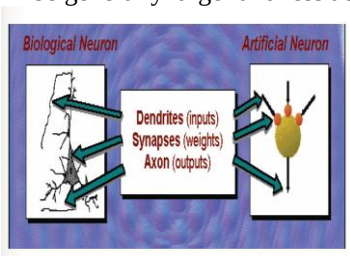

Fig.1 Comparison Of Neurons[7]

Here we compare a biological neuron with a typical artificial neuron. Similarly both take inputs, use weights and generate an output.Neural Networks are composed of a large number of computational nodes operating in parallel. Computational nodes, called neurons, consists in processing elements with a certain number of inputs and a single output that branches into collateral connections, leading to the input of other neurons.[2]

Normally they perform a nonlinear function on the sum (or collection) of their input. The neurons are highly interconnected via weight strengths these interconnections are typically called synapses and control the influence of neurons on the others neurons.[2]

The synaptic processing is typically modelled as multiplication between a neuron outputs and synaptic weight strengths.[3] Each neuron's output level depends therefore on the outputs of the connected neurons and on the synaptic weight strengths of the connections.[3]

Digital technology has advantages of mature fabrication techniques, weight storage in RAM, and arithmetic operations exact within the number of bits of the operands and accumulators. Digital chips are easily embedded into most applications. Digital operations are usually slower than in analogue systems, specially in the weight x input multiplication, and analogue input must first be converted to digital.[2] Analog neural networks can exploit physical properties to perform network operations and thereby obtain high speed and densities. Analog design can be very difficult because of the need to compensate for variations in manufacturing, in temperature, etc Creating an analog synapse involves the complication of analog weight storage and the need for a multiplier being linear over a wide range. Hybrid design attempts to combine t he best of analog and digital techniques. The external inputs/outputs are digital to facilitate integration into digital systems, while internally some or all of the processing is done in analog. There are three types of neural networks.

1. Non-learning neural networks:
2. Off-chip learning networks:

3. On-chip learning networks:[3]

On-chip learning networks is the neural network performs both the feed forward phase and the learning one The advantages are the high learning speed due to the analog parallel operations and the absence of the interface with a host computer for the weight update. On-chip learning networks are suited to implement adaptive neural systems, i.e. systems that are continuously taught while been used.

II. ARCHITECTURE OF NEURON

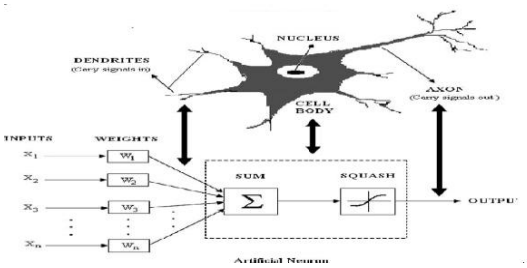

Fig.2 Architecture of Neuron.

Neural network produces a forecast by taking the weighted average of the predictors. This is just what a regression equation does Where neural networks go further is to layer this procedure The first layer of weighted averages (regression equations) produces a "hidden layer" of intermediate forecasts.

These forecasts are then used as predictors in another regression equation to produce the final forecast Note that It is possible to have more than one hidden layer, each one taking the forecasts of the previous layer as it's predictors Each layer is made up of nodes Each node takes the weighted average of the predictors generated by the previous layer The first layer is an "input layer" consisting of the predictors It feeds the first layer of averaging nodes Each of these averaging nodes feeds a corresponding sigmoid to produce a forecast You can stack as many of these forecasting layers as you want, each one taking the forecasts of the previous layer as its predictors The final forecast layer produces the forecast of the quantity you were seeking. This type of neural network is called "feed forward" because the information feeds forward from the predictors, through the layers, and on to the final prediction It is said to be "fully connected" because every node in the one layer connects to every node in the layer above and below it Feed forward networks with sigmoid squashing functions are sometimes called "perceptrons"

$$u_i = \sum_j w_{ij} x_j \qquad (1)$$

Where x_j s the jth predictor, w_{ij} is the weight for that predictor for node i, and u_i is the weighted average coming out of the ith node These weighted averages are then "squashed" by a non-linear sigmoid (s-shaped) function in order to prevent the occurrence of extreme values

$$y_i = \frac{1}{1+e^{-u_i}} \qquad (2)$$

Where, y_i is the forecast generated by the ith node

III. ANALOG NEURAL COMPONENTS

The inputs to the neuron as shown in figure 2 are multiplied by the weight matrix, the resultant output is summed up and is passed though an neuron activation function (NAF). The output obtained from the activation function is taken through the next layer for further processing. The multiplier block, adder block and the activation function model the artificial neural network. Blocks to be used are as follows

1. Multiplication block
2. Adders
3. NAF (Neuron Activation Function)block

A. Analog Multiplier

Here I have used Gilbert multiplier is named for Barrie Gilbert who designed the circuit in 1968 .The circuit combines diode-connected transistors, current mirrors, summing junctions, and differential pairs to multiply two differential signals. Consider two differential pairs that amplify the input by opposites gain

$$v_{out1}/v_{in} = -g_m R_d \qquad (3)$$

$$v_{out2}/v_{in} = g_m R_d \qquad (4)$$

$$V_{out} = V_{out1} + V_{out2} = A_1 V_{in} + A_2 V_{in}, \qquad (5)$$

where A_1 and A_2 are gain which are controlled by V_{ount1} and V_{count2}, respectively. If I1 is zero then $V_{out} = +g_m R_d V_{in}$. V_{count} is used to vary currents monotonically. if V_{count1}-V_{count2} is large then Vout will be most positive or most negative. $V_{out}=V_{in}*f(V_{count})$ and $f(V_{count})$is taylor expansion. $V_{out}=\propto V_{in} V_{count}$ which is multiplication of two inputs.

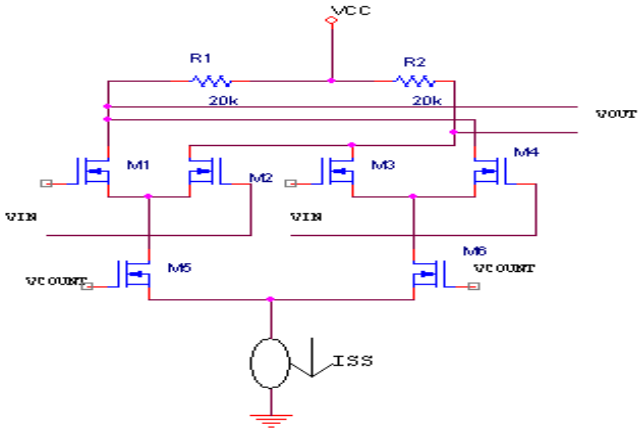

Fig.3 Gillbert Multiplier.

As shown in figure.3 six N-MOSFET are used in designing of Gilbert Multiplier. Designing of MOSFET's are based on 0.35um technology. For given lengths W/L ratios are given in Tabel I.

TABEL I. W/L RATIO OF GILLBERT CELL.

MOSFET(M)	W(µm)
1	1.4
2	1.4
3	1.4
4	1.4
5	0.2
6	0.2

B. Operational Amplifier.

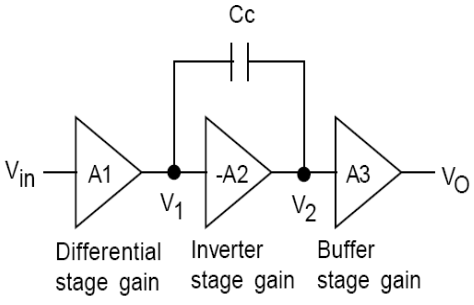

Fig.4 General Block Diagram Of OPAMP[8]

This OPAMP is a two-stage OPAMP where the first stage is a differential amplifier whose differential current output is mirrored into the next stage and converted to a single ended output through circuitry very similar to the synapse circuit. The outputs of the synapses can easily be summed. The summation is done by connecting all current outputs together. The summed current then must be converted to a voltage by a current to voltage (IV) converter.

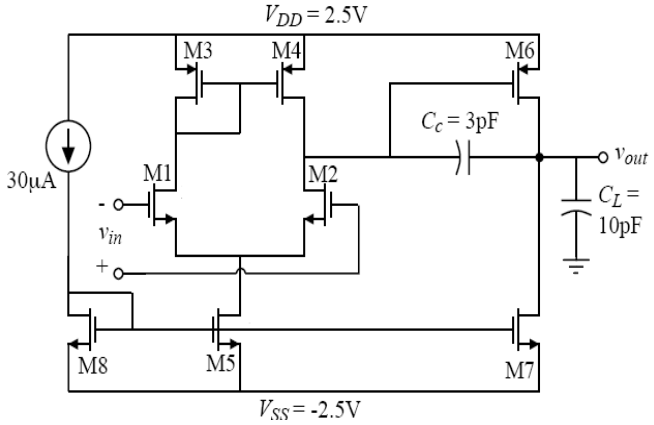

Fig.5 Schematic Of Two stage OPAMP.

TABEL II. W/L RATIO OF OPAMP.

MOSFET(M)	W/L(µm)	W(µm)
(1,2)	3	1.05
(3,4)	15	5.25
(5,8)	4.5	1.58
(6)	94	32.9
(7)	14	4.92

C. Opamp As Adder.

Adder can be design using operational amplifier. So it is necessary to implement op-amp. As shown in figure, opamp is designed with eight MOSFETs. In which five are N-MOSFET and 3 are P-MOSFET'S are designed with L=0.35um length.

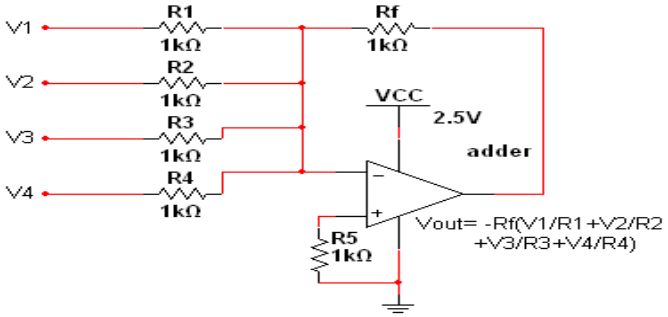

Fig.6 OPAMP As Adder Circuit.

TABEL III. SPECIFICATION OF ADDER.

Adder		
V1	0.25V	-2.0 volt
V2	0.3V	
V3	0.5V	
V4	1.0V	

D. Neuron Activation Function Block.

Neuron activation function designed here is tan sigmoid. The design is basically a variation of the differential amplifier with modification for the differentiation output. The same circuit should be able to output the neuron activation function and the differentiation of the activation function. Here three designs are considered for NAF. NAF can be operated in three regions.

(1)Linear function with adjustable threshold

The output voltage of the circuit is determined by Vo= Vi(R2/R1) - V_{ref}(1+R2/R1) and threshold is determined by q= V_{ref}(R1+R2)/R1. Lower set point is determined by V_{ref} (R1+R2)/R2-Vcc (R1/R2) and Upper set point is determined by the V_{ref}(R1+R2)/R2+Vcc(R1/R2).

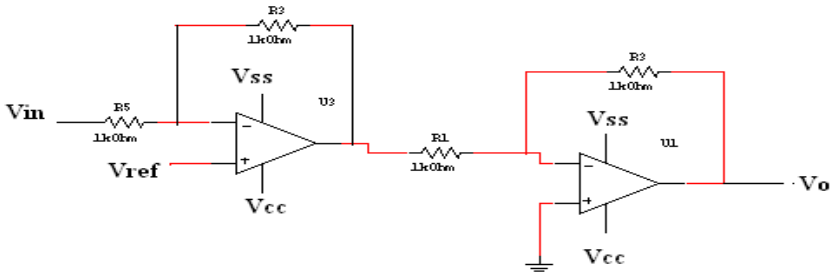

Fig.7 Schematic of Linear function with adjustable threshold

TABEL IV. PARAMETER VALUES.

Parameter Values	
R_1	1000k
R_2	2500k
R_3	1000k
V_{ref}	1 volt

(2) Sigmoid function with fixed gain control.

Another non linear function widely used is sigmoid function as shown in figure.

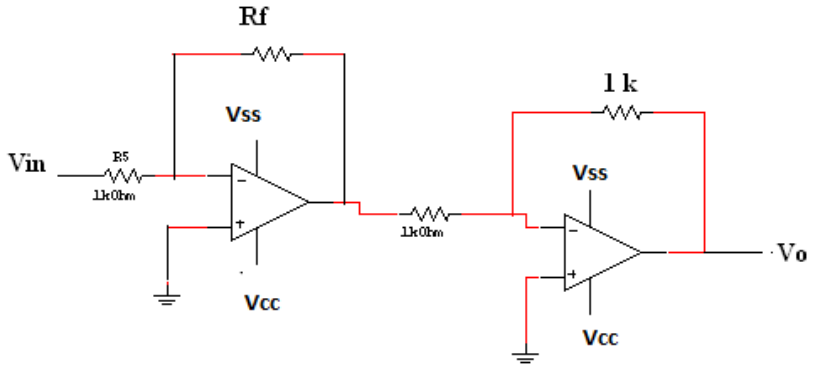

Fig.8 Schematic Of Sigmoid Function With Fixed Gain Control

This is same as Linear threshold function only V_{ref} is made to ground, so Upper limit as well as lower limit will be symmetrical around axis as shown its transfer function.

TABEL V. PARAMETER VALUES.

Parameter Values	
R_s	1000k
R_f	2500k
R_3	1k
V_{ref}	Ground

(3) Step Function

This is also same as sigmoid function but the difference is only in Upper Limit and Lower Limit is almost same. As shown in its Transfer Function it is at zero voltage.

E. Design of Neural architecture.

Artificial Neuron is combination of above developed and simulated blocks. Here, two multipliers with weight and input neuron are developed. Multiplier outputs are differentiated and amplified then applied to adder inputs. Input of multiplier is applied and amplified with respective weight and results of multiplier outputs and adder output can be identified with simulated results. Sigmoid function or Neuron activation function without derivative is used to limit the output of neuron within high and low logic. Result is shown in Fig.19.

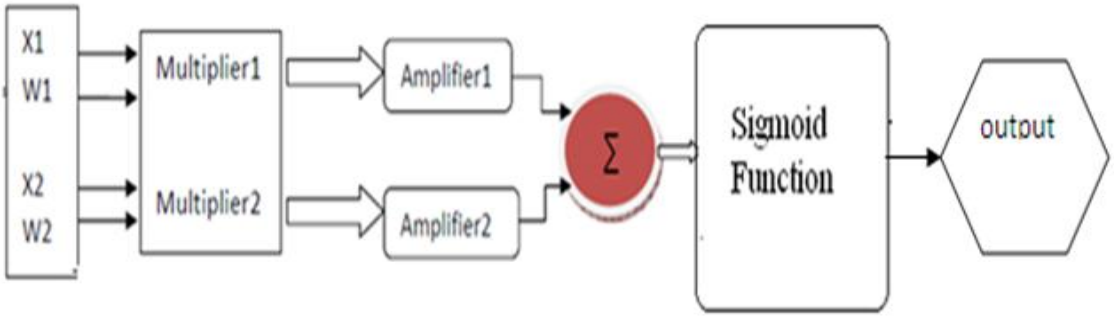

Fig.9 Design of Neural Architecture

IV. SIMULATION RESULTS.

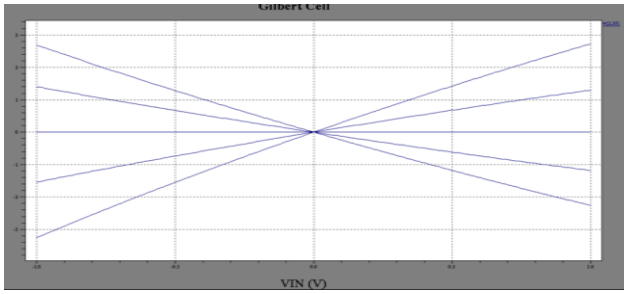

Fig.10 DC Transfer Characteristics

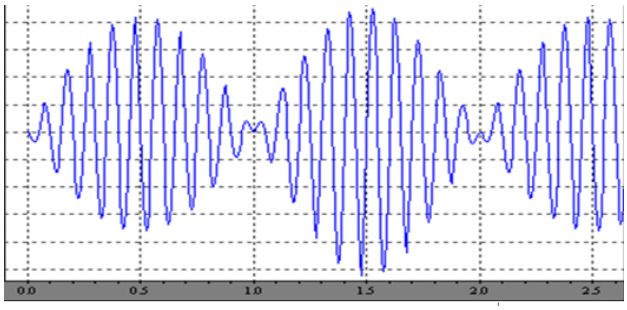

Fig.11 Modulated Waveform for Multiplier.

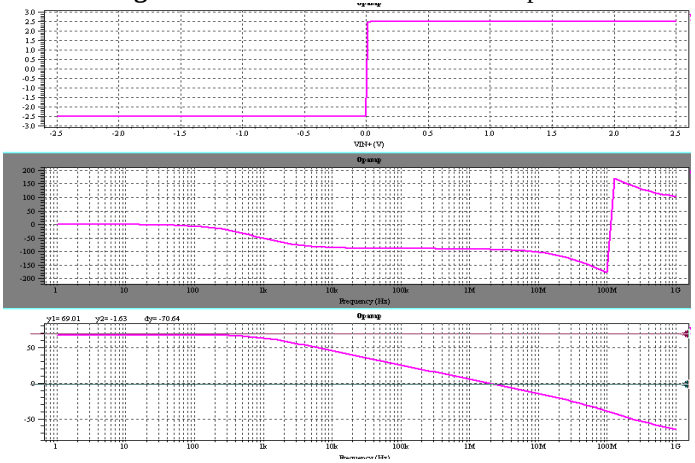

Fig.12 Transfer characteristic ,Gain and frequency response Of OPAMP.

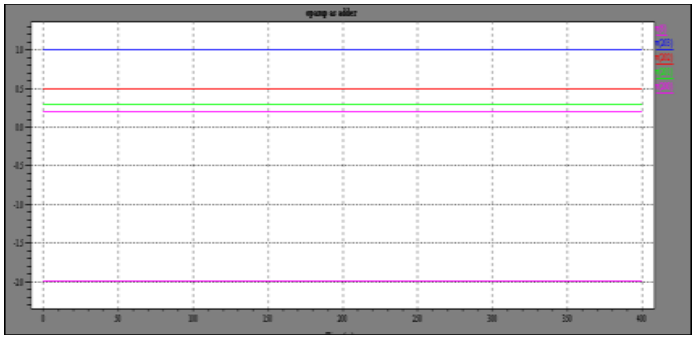

Fig.13 Adder Circuit and Simulation Result.

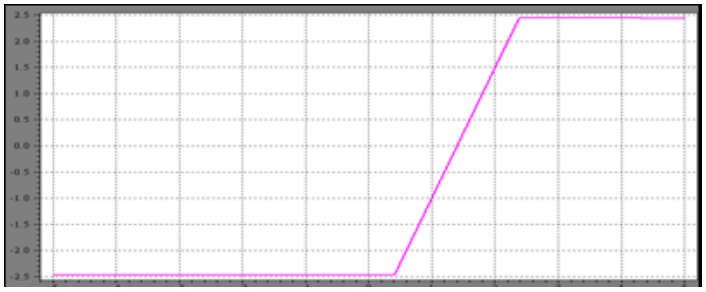

Fig.14 Transfer Function Of Linear Function With Adjustable Threshold

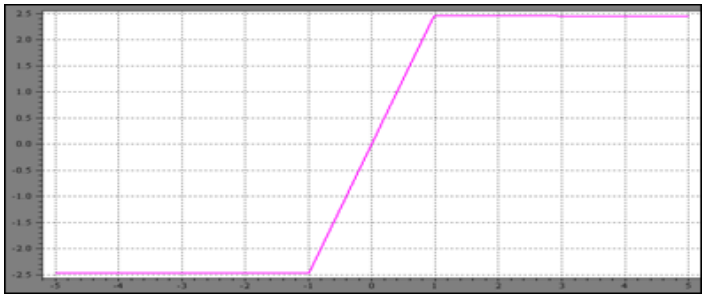

Fig.15 Transfer Function Of Sigmoid function with Fixed Gain Control

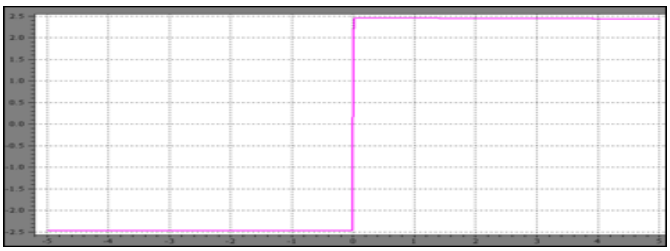

Fig.16 Transfer Function Of Step Function.

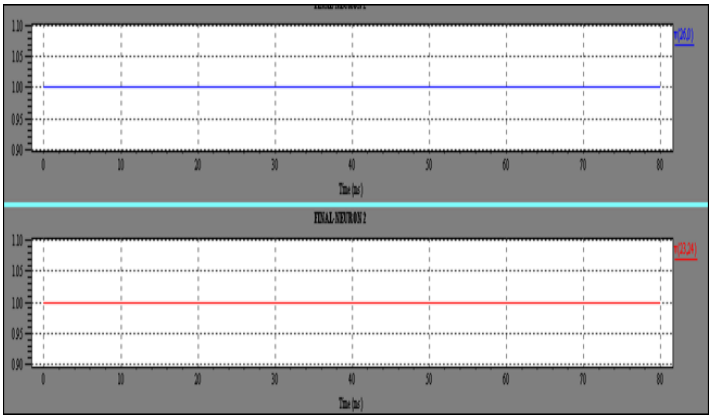

Fig.17 Input signal with weight signal for multiplier 1

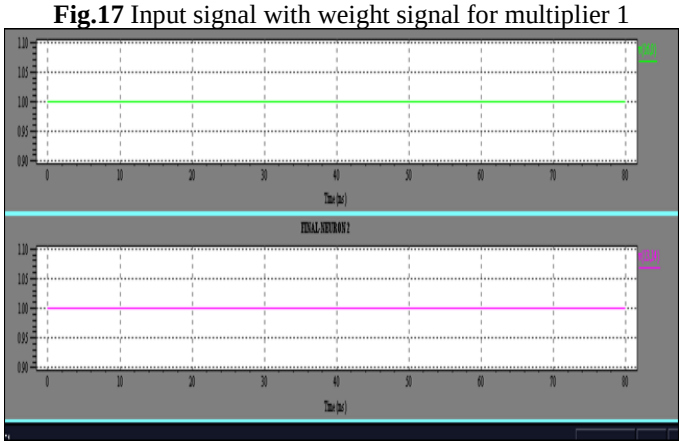

Fig.18 Input signal with weight signal for multiplier 2

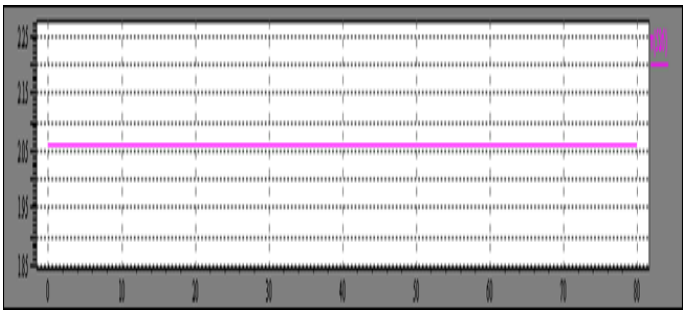

Fig.19 Output of Neural architecture

CONCLUSION & FUTUREWORK

Neural architecture is widely used in Real world interface & it can be implemented using different methods but we choose analog VLSI because it is very fast compared to digital VLSI & no need of A/D or D/A converter. One important is that it can be directly interface with physical sensors & actuators so used in pattern recognition like ECG, image processing and many applications.

Here we designed analog neural components like (Gillebert cell) multiplier or mixer, CMOS OPAMP, adder using OPAMP, activation function sigmoid, step function & linear threshold function with help of simulation TSPICE Software. After design of neural components I have combined all this blocks and designed neural architecture and got the simulation results.

In future we will design winner takes all circuit which is used to identify maximum active input than combine all neural components with WTA circuit and apply neural architecture with WTA for X-RAY image segmentation.

REFERENCES

Journals:

[1] B. Gilbert, "A precise four quadrant multiplier with subnanosecond response," *IEEE*

[2] Ismet Bayraktaroglu,"Circuit Level Simulation Based Training Algorithms For Analog Neuralnetworks", Electrical and Electronic Engineering, M.S. Thesis, 1996

[3] Bo Gian Marco,"Microelectronic Neural Systems:Analog Vlsi For Perception And Cognition",Thesis November 1998

[4] Bose N. K.Liang P.,"Neural Network Fundamentals with graphs, algorithms and Application", Tata McGraw hill, New Delhi, 2002, ISBN 0-07-463529-8

[5] Cyril Prasanna Raj P. ,"Design and Analog VLSI Implementation of Neural Network Architecture for Signal Processing"

[6] P.E. Allen,"Cmos Analog Circuit Design",2005

[7] A PPT On Fundamentals Of Neural Network.

Books:

[8] Razavi Behzad, "Design of Analog CMOS Integrated Circuits", Tata McGrawhill, New Delhi, 2002, ISBN 0-07-052903-5

[9] Mohammad Rashid "Pspice Using Orcad for Circuits and Electronics, PHI learning

[10] R. Jacob Baker, Harry W. Li and David E. Boyce "CMOS Circuit Design, Layout, and Simulation" Department of Electrical Engineering Microelectronics Research Center.

[11] Douglas R. Holberg, Phillip E. Allen, "CMOS Analog circuit design OXFORD University Press