Efficient Hardware Architecture for Single Neuron in Artificial Neural Network

Pham Dang Lam, Hoang Trang
University of Technology-VNU-HCM, 268 Ly Thuong Kiet str., dist. 10, Ho Chi Minh City, Vietnam
*Email: {lamd.pham, hoangtrang}@hcmut.edu.vn

***Abstract*- Artificial Neural Network (ANN) schemes usefully apply to enhance the current parallel processes such as pattern recognition, system identification or adaptive control system. In pattern recognition systems, an ANN mathematic model includes many nodes known as single neurons. And each neuron comprises simple components named as Nonlinear Weighted Sum (NWS) and Activation Function (AF) structures. The state-of-art of NWS and AF hardware architecture replies on complexity and efficiency of single neuron in an ANN. Perhaps, the most challenge in implementation of ANN hardware is to be able to reduce the mean error but still satisfy the strict silicon area and speech requirements based on how to implement the NWS and AF structures efficiently. This work proposes the effective hardware architecture for single neuron in an ANN combining the NWS and sigmoid function as AF structure to get the best performance in pattern recognition applications. The new scheme is not only co-simulated on both software version in MATLAB and hardware version in Verilog hardware language but also synthesized on FPGA and 65nm technology to estimate the area, timing violation and feasible speech. In hardware implementing process, the floating point operations following the IEEE 754 standard and booth algorithm are also used to reduce the measurement uncertainty.**

Keywords-Artificial Neural Network (ANN), neuron, Application Specific Integrated Circuit (ASIC), Field Programmable Gate Array (FPGA). Activation Function (AF), Nonlinear Weighted Sum (NWS).

I. INTRODUCTION

Artificial Neural Network, each of neuromorphic engineering branches, was introduced and obtained many successes in the end of 1980 [1, 2]. However, ANN is only explored from point of view of software because of the hardware resource limitation. Nowadays, the implementation of ANN on the processing computers by software is not suitable for applications requiring real time strictly even if using the faster computer. So, the parallel hardware ANNs are approached as great potential solutions. Besides, continuing scaling of Moore's law for greater transistor densities is able to verify the neural network models in hardware environment more easily. As the result, many neuromorphic chips or hardware libraries are available for specific applications [3]. Based on the electrical pulse activities through the real neural cell components, the common mathematic model for artificial neural cell applied in many pattern recognition systems widely is shown in figure 1.

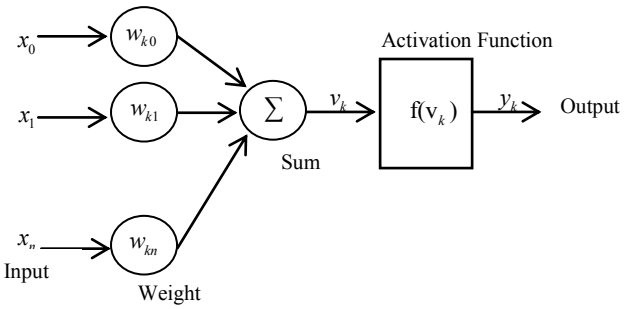

Figure 1. Artificial neural mathematic model

Where wkn and xn are two input vectors, f(vk) is the activation function, and the yk is the output of the neuron. The relation between inputs and output in one neuron corresponds to the following generalization

$$y_k = f(\mathrm{v}_k) = \mathrm{f}(\mathrm{w}_{ki} \times \mathrm{x}_i) = \mathrm{f}(\mathrm{W} \times \mathrm{X})$$

$$\mathrm{X} = \begin{bmatrix} x_0 \\ x_1 \\ . \\ . \\ . \\ x_n \end{bmatrix} \text{ and } W = \begin{bmatrix} w_0 \\ w_1 \\ . \\ . \\ . \\ w_n \end{bmatrix} \quad (1)$$

As presenting about the ANN structure, the state-of-art single neuron affects the performance of ANN directly. So, in order to get the convenient ANN hardware architecture, many approximate methods have been proposed to simplify the single neuron in ANN. Most of the previous works attended to approximate the AF in artificial neural cell to achieve the simplest hardware design. There are two principal methods in which the Look_up tables, Taylor series scheme is the first and sum-of-step, piece-wise linear or the others is the other branch taking many advantages more than. In the second direction, the sigmoid function as AF are proposed by many studies such as A-law approximation proposed by Myers and Htchinson, Alippi and Storti-Gajani approximation [4], PLAN Approximation (Piecewise Linear Approximation) introduced by Amin [5], Piecewise second-order approximation presented by Zhang et.al [6], Thamer M.Jamel et.al's approximation [8],

This work is funded by the Department of Science and Technology HoChiMinh City (DOST HCM) from 12/2012-05/2014

in which the best performance shows 0.0129 maximum error in Alippi and Storti-Gajani approximation scheme.

In this work, the effective hardware architecture for artificial neural cell in ANN is proposed. In the new structure, using the floating point format IEEE 754, Booth algorithm and new Piecewise Linear Approximation (new PLAN) of nonlinear sigmoid function gain the efficiency results compared with the other methods.

The rest of paper is organized follows. Section 2 gives the overview of approaching algorithms, standards, basic hardware architectures of ANN related to artificial neuron, some approximate methods for activation function. Next, section 3 proposes an efficient approximate method for AF in single neural cell. The ANN architecture and its detailed operation are given in section 4. Section 5 shows the experiment results. Finally, section 6 presents the conclusion.

II. OVERVIEW

In this section, the approaching algorithms, standards and the basic hardware architecture of ANN related to artificial neuron is presented firstly. Next, some approximate methods for activation function are collected and analyzed to estimate their performances.

a. Floating point IEEE 754 format

The floating point IEEE 754 format is the most widely used in many CPUs or FPUs of computer systems [12, 13]. And the floating point number flowing IEEE 754 format includes three distinct components as sign digit, exponent digit and significant digit (mantissa) basically as in figure 2.

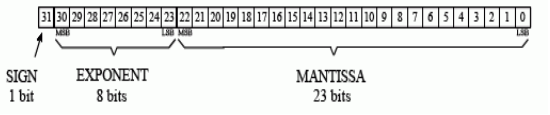

Figure 2. Floating point IEEE 754 format

Because of the advances as accuracy and wide range of floating point number, many researches in the pattern recognition systems using neural network method apply the floating point format as the effective method to enhance the performance [14, 15]. Hence, this paper also utilizes the single floating point IEEE 754 format as one of conveniences to increase the accuracy and reduce the measurement uncertainty.

b. Booth algorithm

The Booth algorithm, invented by Andrew Donald Booth, is multiplication algorithm that attends to reduce calculation steps for large sign binary numbers in two's complement notation. In multiplication operation between two significant digits following the floating point IEEE 754 format, applying the Booth algorithm presented by Puneet Paruthi [9] or Shaifali [11] confirms the high performance compared with array multiplication. Besides, Gokul Govindu et.al's survey also confirms the interpretation of Booth algorithm for matrix multiplication application on FPGA base architecture [10]. Then, the Booth algorithm is applied to promote the proposed neural hardware architecture.

c. Basic hardware architecture

From point of view of hardware design, the single neural cell can be imagined as a simple architecture including NWS and the AF structures as in figure 3. The NWS component implements the total sum of all multiplications of inputs and weight values and the AF component takes the role of activation function.

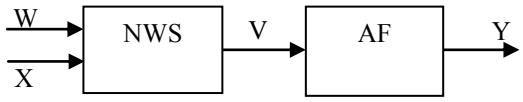

Figure 3. Hardware architecture for artificial neural cell

In pattern recognition system using ANN as one of effective solution, back-propagation algorithm is adopted as leaning process gaining many successes to find golden weight values following the training rules. One of steps of propagation algorithm is to get the result of derivable function to apply the gradient decent optimization algorithm. In many activation functions are proposed as Steep, Hardlim, Purelin, sigmoid, or tan-sigmoid, the sigmoid function is chosen because of the advance formula for calculating the derivable function.

$$f(\mathrm{x}) = \frac{1}{1+e^{-x}}; f'(\mathrm{x}) = \frac{e^{-x}}{1+e^{-x}} \\ \Rightarrow f'(\mathrm{x}) = \mathrm{f(x)(1-f(x))} \tag{2}$$

Another main reason of choosing the sigmoid function as AF in pattern recognition machines is its wide variety gaining higher performance compared with the other functions.

However, an obstacle to approach the hardware view point for sigmoid function is how to approximate this nonlinear function to more simple formula. As the result, many researchers try to prove the high performance approximate methods such as Alippi, Amin, Zhang, and Thamer M.Janel… Following the bellow approximate formulas, the authors use the fix point standard attending to simple the hardware design to reduce the area more than to decrease the measurement uncertainty. The survey result for input values in range [-8; 8] with 0.1 step value on software (MATLAB) confirms their efficiency compared together.

❖ **Alippi method**

$$y = \begin{cases} \dfrac{1-1/2+FRAC(-\mathrm{x})/4}{2^{|\mathrm{INT(X)}|}} & for \quad x > 0 \\ \dfrac{1/2+FRAC(\mathrm{x})/4}{2^{|\mathrm{INT(X)}|}} & for \quad x \le 0 \end{cases} \\ \mathrm{INT(x)}: \text{Intergral part of x} \\ \mathrm{FRAC}(x): \mathrm{x} + |\,\mathrm{INT(x)}\,| \tag{3}$$

❖ **PLAN method**

$$y=\begin{cases}1 & for \ \ |x|\geq 5\\ 0.03125|x|+0.84375 & for \ \ 2.375\leq|x|<5\\ 0.0125|x|+0.625 & for \ \ 1\leq|x|<2.375\\ 0.25|x|+0.5 & for \ \ 0\leq|x|<1\end{cases} \quad (4)$$

❖ **Zhang method**

$$y=\begin{cases}\frac{1}{2}\left(\frac{x}{2^2}-1\right)^2 & for \ \ -4<x<0\\ 1-\frac{1}{2}\left(\frac{x}{2^2}+1\right)^2 & for \ \ 4>x\geq 0\end{cases} \quad (5)$$

❖ **Thamer M.Janel method**

$$y=\frac{1}{2}\left(\frac{x}{1+|x|}+1\right) \quad (6)$$

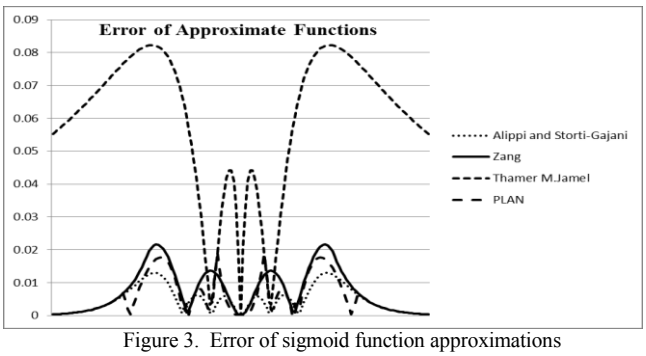

Figure 3. Error of sigmoid function approximations

TABLE I. THE PERFORMANCES OF APPROXIMATE METHODS

Activation Function	Maximum Error	Mean Error
Allipi	0.0129	0.0048
PLAN	0.0189	0.0058
Zhang	0.0215	0.0076
Thamer M.Jamel	0.0822	0.0596

Analyzing table I shows that the PLAN scheme is more suitable than others for following advances. Firstly, the survey confirms the effective maximum error (0.0189) and mean error (0.0058) compared with the other schemes. Next, the PLAN method only includes the multiplication and addition operators which are also used in the NWS structure being inconvenient to approach the hardware view point.

However, PLAN method proposed by Amin et.al is used fix point standard for operations. As the result, the error and corresponding mean error still have large measurement uncertainty, 0.007 compared with the original sigmoid function around 3.5 or -3.5 of input values. In some pattern recognition applications, the accuracy is one of strict requirement. So, PLAN method need to improve for more convenience.

III. PROPOSED APROXIMATE METHOD

As presenting about the lack of PLAN method proposed by Amin et.al, this paper presents the new PLAN to tackle this issue. Following the PLAN scheme of Amin, the linear interval from 0 to 5 of approximate function is still chosen. But in the new PLAN, the comparative thresholds are changed to get more accuracy as the bellow generalization

$$y=\begin{cases}1 & x\geq 5\\ 0.014217245\times x+0.924396007 & 3.6\leq x<5\\ 0.04963536\times x+0.800778912 & 2.3\leq x<3.6\\ 0.136783431\times x+0.606760369 & 1\leq x<2.3\\ 0.231058579\times x+0.504540715 & 0\leq x<1\end{cases} \quad (7)$$

Because the sigmoid function valuation for x < 0 is the complement of it at x >0, the completed approximate function is expanded.

$$y=\begin{cases}1 & x\geq 5\\ 0.014217245\times x+0.924396007 & 3.6\leq x<5\\ 0.04963536\times x+0.800778912 & 2.3\leq x<3.6\\ 0.136783431\times x+0.606760369 & 1\leq x<2.3\\ 0.231058579\times x+0.504540715 & 0\leq x<1\\ 1-f(x) & x<0\end{cases} \quad (8)$$

In many approximate methods, the range of input value can larger or smaller than 5. In proposed method, the survey confirms that the measurement uncertainty, around 0.00017%, of activation function for the input values being larger than 5 or smaller than -5 is very low. Hence, 5 is chosen as the threshold value in new method same as the PLAN method.

IV. SINGLE NEURAL CELL ARCHITECTURE

As introducing in section II, the simple neural cell is shown as in figure 3 including the NWS and AF structures. In order to understand how the single neural cell solves the inputs signals and weight values, the simple three layer ANN applied in the pattern recognition is given as in the figure 4.

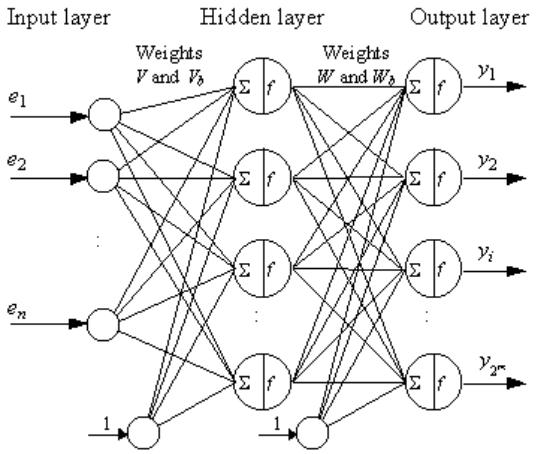

Figure 4. Three layer artificial neural network

In the figure 4, every neural cell in the hidden layer receives all the inputs from the input layers. And every hidden neural cell has to implement many addition and multiplication operations in NWS structure to sum all multiplication between input values and weight values. Next, this result is transferred to AF structure. At the AF structure, the multiplication and addition are executed again corresponding to the new PLAN scheme. This is also same as any neural cell in the ANN.

So, in order to reduce the number of operations, the combination between the state machine and loop algorithm is presented as the figure 5. In this proposed diagram, the signal processing in the artificial neural cell is controlled by state machine including NWS and ACTFUN states. In the NWS state, the pairs of input signal and weight value are multiplied by multiplication component in NWS structure step by step. After finishing every multiplication operation, the result is added with result of multiplication before by the addition component in NWS structure. And this loop is continued until that all the input values and weight values are read out from the memory outside.

Hence, the next state, ACTFUN state, is called to implement the role of activation function. Again, the multiplication and addition operations are used as the proposed new PLAN method. Inclusion, the signal processing in the single artificial neural cell proposed is the result of many multiplication and addition operations.

As hardware view point, the proposed neural cell architecture is optimized to the units known as the multiplication and addition operations. As the result, the accuracy of these operations affects to the performance of neural cell directly. In order to solve this issue, the author proposes the floating point IEEE 754 format combined to Booth algorithm for multiplication and addition operations as the efficient solutions.

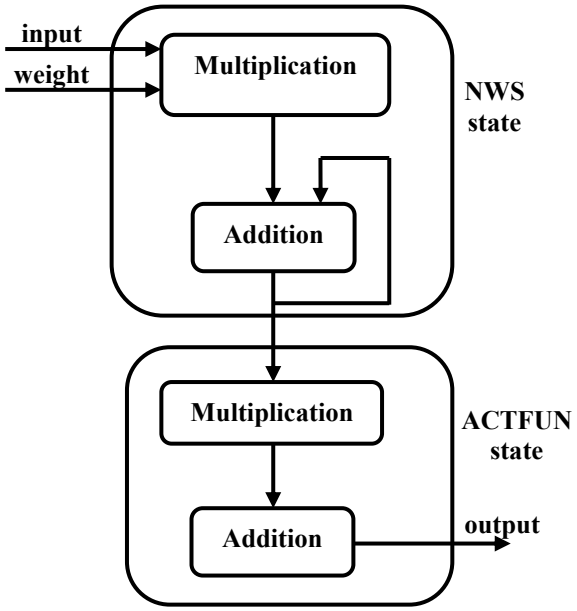

Figure 5. Signal processing in the artificial neural cell

V. EXEPERIMENTS

The first results of simulation on software (MATLAB) show the efficiency of new approximate sigmoid function compared with the other methods.

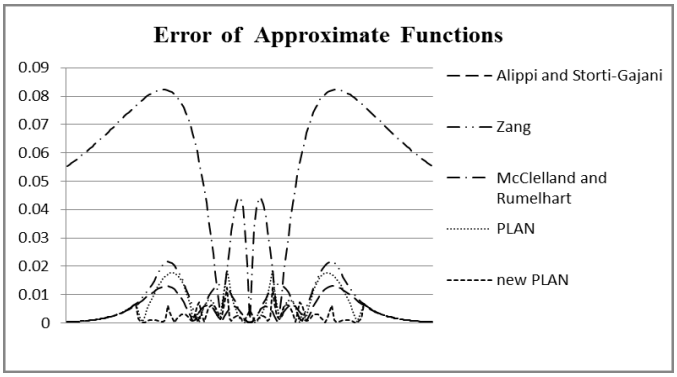

Figure 6. Error of approximate methods

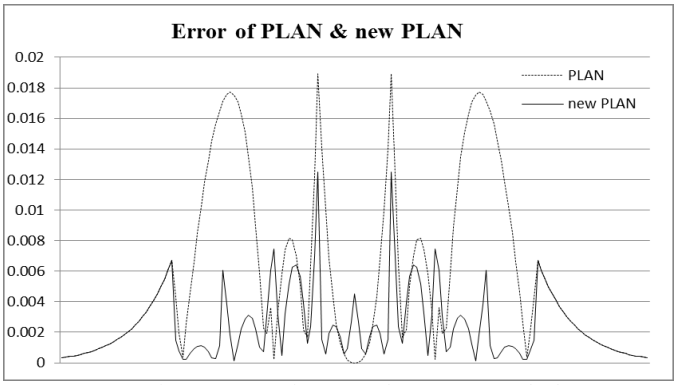

Figure 7. Error of PLAN and new PLAN methods

TABLE II. PERFORMANCES OF APPROXIMATE METHODS

Activation Function	Maximum Error	Mean Error
Allipi	0.0129	0.0048
Plan	0.0189	0.0058
Zhang	0.0215	0.0076
Thamer M.Janel	0.0822	0.0596
New PLAN	**0.0124**	**0.0024**

In our experiment, the input values in [-8; 8] range with 0.1 step are verified again. And the results are collected to calculate the maximum error and mean error showing that the new PLAN gets the best measurement uncertainty result. Next, the new PLAN for sigmoid function is implemented in Verilog language and co-simulated between software and hardware versions.

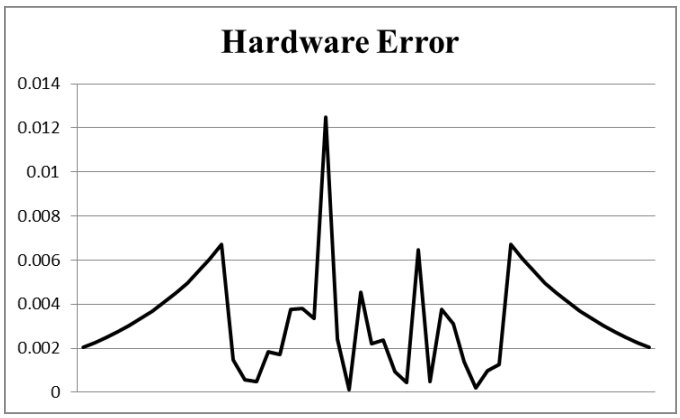

Figure 8. Hardware Error Simulation Result

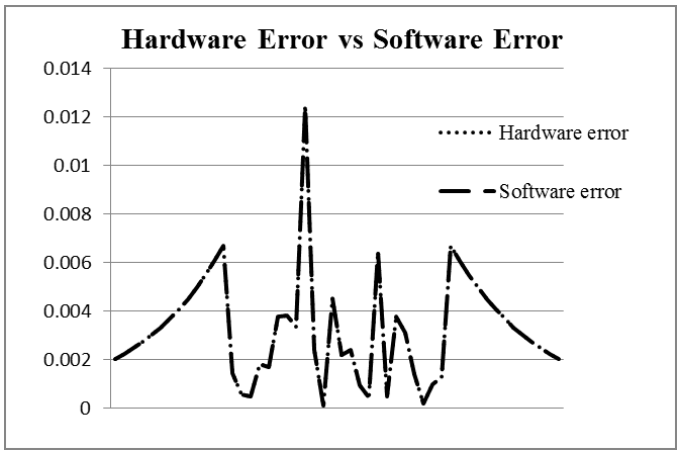

Figure 9. Approximate Sigmoid Co-Simulation

These figures 8 and 9 show that the measurement uncertainty is very low confirming the matching between hardware and software results. The following table III also collects the maximum error and mean error information for proposed approximate sigmoid function on hardware view point compared with standard sigmoid function.

TABLE III. PROPOSED APPROXIMATE SIGMOID FUNCTION PERFORMANCE

New PLAN	Maximum Error	Mean Error
Software	0.01248522236	0.00244
Hardware	0.01248522506	0.00325

Next, the hardware architecture for single neural cell is implemented base on the diagram in figure 3.2. The following results not only confirm the accuracy of the neuron behavior but also estimate the performance of neural cell on FPGA and TSMC 65nm library.

One hundred automatic test cases generated by software (MATLAB) are applied to confirm the accuracy of single neuron. In every test case, twenty six pairs of input and weight values inspired from the sound features of Mel-frequency-cepstrum method in speech recognition systems are transferred to the single neural architecture step by step. Then, the results on both software version and hardware version are collected and analyzed.

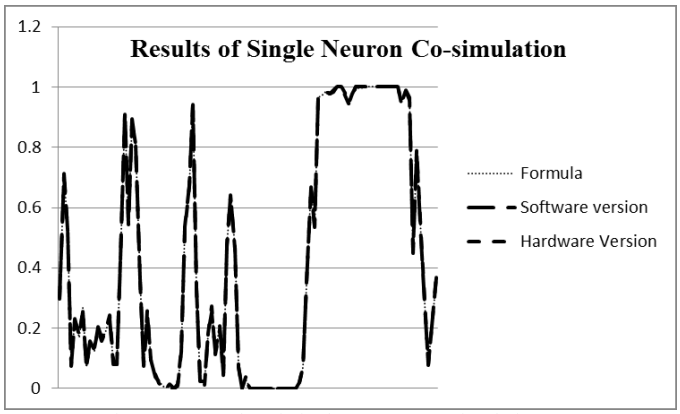

Figure 10. Results of Single Neuron Co-Simulation

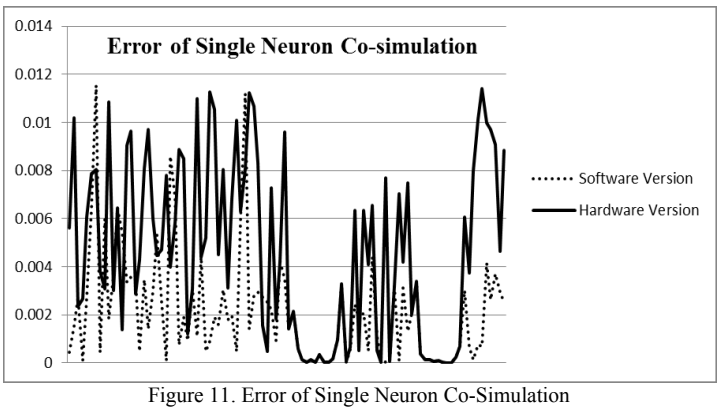

Figure 11. Error of Single Neuron Co-Simulation

Figure 11 shows the matching between software simulation and hardware simulation compared with the single neuron behavior build on formula base. Next, the figure 4.6 and the table IV point out the measurement uncertainty

between software version and hardware version in random test cases.

TABLE IV. SINGLE NEURON CO-SIMULATION PERFORMANCE

Single neuron	Maximum Error	Mean Error
Software	0.01155	0.00214
Hardware	0.01139	0.00466

Next, the proposed neuron hardware architecture is also synthesized on both FGPA (Altera - Cyclone II- C35F672C6 and Xinlin-Virtex4-xc4vsx35) and 65nm technology to estimate the resource used. Because this work is first steps to build up the completed artificial neural network, the effective structure must not use the utilities such as Memory, DSP core, or PLL core supported by FPGA conveniences. So, the results synthesized on 65nm technology inform the best frequency (250 MHz) and the feasibility to develop the completed soft IP following ASIC design flow. The hardware structure is also verified in real time on DEII kit (FPGA – Cyclone II) in 50 MHz frequency maximum to demonstrate the great potential design strongly.

TABLE V. PROPOSED SINGEL NEURON PERFORMANCE

Resource	Altera - Cyclone II- C35F672C6	Xinlin-Virtex4 - xc4vsx35	TSMC 65nm process
Combination Logic	4816(15%)	5369 (2%)	12616 (NAND)
Register	815(2%)	814 (17%)	1050 (DFF)
Other resource	None	None	None

TABLE VI. FREQUENCY PERFORMANCE

Allipi	Plan	Zhang	Thamer M.Janel	New PLAN
36	39	176	None	250

VI. CONCLUSION

In this paper, the hardware architecture of single artificial neuron is proposed to solve both accuracy issue and frequency issue. The obtained experimental results validated on FPGA and 65 nm technology confirm the feasibility of completed. Employing the proposed architecture in pattern recognition such as speech, human face or handwriting recognitions using complete ANN as the effective method is our future steps.

REFERENCES

[1]. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. Lang, "Phoneme recognition using time-delay neural networks," Acoustics, Speech and Signal Processing, IEEE Transactions on, vol. 37, pp. 328-339, Mar. 1989.

[2]. Robinson, "An application of recurrent nets to phone probability estimation," Neural Networks, IEEE Transactions on, vol. 5, pp. 298-305, Mar. 1994.

[3]. M.C.Miglionico, F.Parillo, "Modelling a neuron using a custom math library sfloat24 – Implementation of a Sigmoid function on a FPGA device", ISHAP Conference Sorrento Italy, pp.15-18, June 2011.

[4]. Alippi, C., and Storti–Gajani, G.: "Simple approximation of Sigmoidal functions: realistic design of digital neural networks capable of learning". Proc. IEEE Int. Symp. on Circuits and Systems, Singapore, pp. 1505–1508, 1991.

[5]. Amin, H., et all: "Piecewise linear approximation applied to nonlinear function of a neural network", IEE Proc. Circuits, Devices Sys., 144, (6), pp. 313–317J, 1997.

[6]. Zhang, M., et all.: "Sigmoid generators for neural computing using piecewise approximations", IEEE Trans. Comput.,45, (9), pp. 1045–1049, 1996.

[7]. Alin TISAN, Stefan ONIGA, Daniel MIC, Attila BUCHMAN, "Digital implementation of the Sigmoid function for FPGA circuits", ATCT Technica Napocensis, Volume 50, Number 2, 2009.

[8]. Thamer M.Jame, Ban M.Khammas, "Implementation of a Sigmoid activation function for neural network using FPGA", Scientific Conference of Al-Ma'moon University College, April 2012.

[9]. Puneet Paruthi, Tanvi Kumar, Himanshu Singh, "Simulation of IEEE 754 Standard Double Precision Multiplier using Booth Techniques", International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622, Vol. 2, Issue 5, September- October 2012.

[10]. Ling Zhuo, Seonil Choi, Prasanna, Viktor Prasanna, "Analysis of High-performance Floating-point Arithmetic on FPGAs" Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International, 26-30 April 2004.

[11]. Shaifali, Ms.Sakshi, "Comparison of IEEE-754 Standard Single Precision Floating Point Multiplier's", International Journal of Emerging Trends in Electrical and Electronics (IJETEE), Vol. 1, Issue. 3, March-2013

[12]. A.L. Thall, "Extended-precision floating-point numbers for GPU computation," Poster Session, ACM SIGGRAPH '06 Annual Conference,Boston, MA, August 2006.

[13]. Andrew Thall, "Extended-Precision Floating-Point Numbers for GPU Computation", ACM SIGGRAPH '06 NewYork, USA, Artice No.52, , 2006.

[14]. Damak, A , Krid, M., Masmoudi, D.S., "Neural Network Based Edge Detection with Pulse Mode Operations and Floating Point Format Precision", Design and Technology of Integrated Systems in Nanoscale Era, page 1-5, 25-27 March 2008.

[15]. Pedro Ferreira, Pedro Ribeiro, Ana Antunes, Fernando Morgado Dias, "A high bit resolution FPGA implementation of a FNN with a new algorithm for the activation function", Neurocomputing, Volume 71, Issues 1–3, Pages 71–77, December 2007.