International Journal of Reconfigurable and Embedded Systems
Vol.8, No.2, July 2019, pp. 145~150
ISSN: 2089-4864, DOI: 10.11591/ijres.v8.i2.pp145-150

Neuronal logic gates realization using CSD algorithm

Lakshmi Kiran Mukkara, K.Venkata Ramanaia.
Department of Electronics and Communication Engineering, YSR Engineering College of Yogi Vemana University,Proddatur, India

Article Info

Article history:

Received Feb 17, 2019
Revised Apr 19, 2019
Accepted May 21, 2019

Keywords:

CSD (Canonic Signed Digit)
FPGA (Field Programmable Gate Array)
VLSI (Very Large Scale Integrated circuit)

ABSTRACT

Any digital circuit is made with fundamental building blocks i.e. logic gates. Artificial neural networks (ANN) became an emerging area in various applications such as prediction problems, pattern recognition, and robotics and system identification due to its processing capabilities with parallel architecture. Realization of Boolean logic with neural networks is referred as neuronal logic. ANN computes faster as it requires of low and simple precision computations. Also, it requires economic and low precision hardware. Neural network contains more number of addition and multiplication processes. It is known that CSD algorithm computes faster than conventional or standard multipliers. In this paper, VLSI implementation of neuronal half adder with CSD algorithm is proposed and implemented in FPGA. The results are compared with that of conventional and vedic multiplier. It is observed that CSD algorithm provides lowest delay and low power consumption in comparison with vedic algorithm and conventional method but at the expense of minimum area.

Corresponding Author:

Lakshmi Kiran Mukkara,
Department of Electronics and Communication Engineering,
YSR Engineering College of Yogi Vemana University,
Proddatur, Kadapa 516360, India.
Email: ijres@iaesjournal.com

1. INTRODUCTION

Computation theory is mainly realized with logic gates and digital circuits. Lot of research has been doing for complexity reduction of embedded systems i.e to achieve low power system. Hence logic gates such as OR, AND, NOT become back bone of any digital design in current days. These Boolean gates combined in various ways to form buffers, half adders, full adders, inverters and so on, which act as basic building components of any complex digital circuit and they are available in various package styles. Gate delay of each gate affects overall delay of the system, in this paper a novel approach is considered to have low propagation delay [1].

A human brain is consisting of infinite number of neurons with very complexity structure. In the nervous system, neuron is the key element which passes information both internally (within the brain) and externally (out of the brain and within the body). An artificial neural network mimic the function of biological neuron and has entered into many applications such as pattern recognition, robotics, cyber security, digital image processing, GPS systems, medical diagnosis and so on. Some problems can be solved using ANN with proper training, which are not solved by human beings and normal computers. ANN has some attractive properties like parallel processing, adaptability and self-organization. Table 1 Doses evaluation of administration anti-hypertensive drugs [2].

Multiplier is a key component in the ALU design of different processors. They may be image and signal processors, general purpose microprocessors, special purpose micro controllers, FPGAs, CPLDs and so on. Multipliers must be fast for designing effective architectures. Several fast multipliers are designed using CSD algorithm in literature. Normal multiplier contains partial products with is the root cause for

power consumption and computation delay. CSD algorithm reduces number of partial products so that computation become faster and power consumption become lower.

Implementation of Boolean logic with ANN is referred as Neuronal logic. Efficient design of single neuron affects the efficiency of neuronal logic system. FPGAs are suitable for Neuronal logic implementation due to its flexibility and re-configurability. But the issue here is to implement the total neuronal logic system on FPGA.

In this paper, characteristics of ANN and features of CSD multiplication algorithm are referred for realization of half adder (AND & XOR logic gates) to determine the performance of Conventional neuronal half adder, Vedic neuronal half adder and CSD neuronal half adder Section 2 shows NN structures for basic logic gates, Section 3 contains design of NN logic gates with vedic algorithm, Section.4 discusses design of NN logic gates with CSD algorithm, Section 5 presents results and discussion of the carried out work and Section 6 presents conclusion and future scope of the proposed work.

2. NEURONAL HALF ADDER

2.1. Neuron Model

McCulloch introduced the model for neuron after rigorous study on human nervous system. Many models are proposed after this model. As in the biological system neuron gets stimulus from its dendrites, neuron in these models gets stimulus input from nearby fields. Such inputs are combined to form a net output and that net output is send to activation function which may be linear or nonlinear activation function. If the net output is higher than threshold then output of activation function is high and is known as "fired". This can be given as input to another neuron for firing. Basic neuronal model of McCulloch is as shown in Figure 1.The neuron may be described in the following way.

$$u_k = \sum_{j=1}^{n} w_{kj} x_j \quad (1)$$

$$v_k = u_k + b_k \quad (2)$$

$$y_k = \omega(v_k) \quad (3)$$

Where x_1, x_2, x_3,…x_m are input stimuli.
w_{kj}=synaptic weight of neuron k

y_k=output of neuron k

b_k=bias value of neuron k

ω=activation function

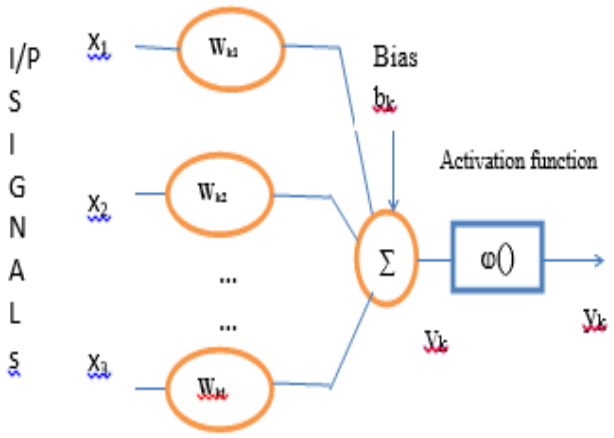

Figure1. Mcculloch Neuron model

2.2. NN model for Half Adder

The above model in Figure 2 represents neural network based Half Adder with inputs x1, x2 and outputs sum and carry. Numbers along the path are weights and numbers above the summing element are bias values. This structure contains 2 levels of neurons, first layer contains 2 neurons and second level contains one neuron for each sum and carry, where sum/carry can be obtained by sending neuron output through activation function.

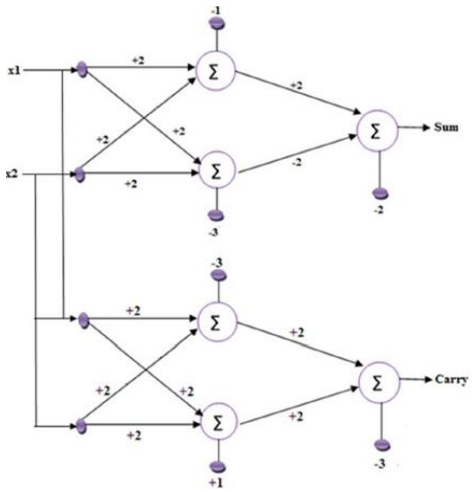

Figure 2. NN model for Half Adder

2.3. Hardware realization of a neuron

Realization of hardware for Artificial Neuron has been of keen focus since last few decades. FPGA is one of the efficient VLSI implementation methods which are used by researchers to implement artificial neuron. Implementation of single input neuron with hardware is as shown in Figure 3.

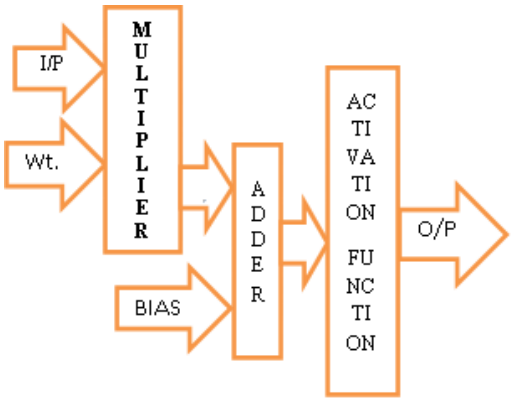

Figure3. Hardware implementation of simple neuron model

3. VEDIC NEURONAL HALF ADDER

Basic neural network is actually contains multiplier, adder and activation function along with the inputs, weights, bias inputs and output(s) as shown in Figure 3. This can be called as Vedic Neuron structure as vedic algorithm is applied in the multiplier block. Vedic algorithm contains 'UrdhvaTiryagbhyam' Mantra which improves the speed of multiplication. Following Figure 4 shows multiplication of two 4-bit numbers using this vedic mantra. Figure 5 Illustration of 4-bit Vedic Multiplicationas shown in Figure 4 Vedic Neuron Structure as shown in:

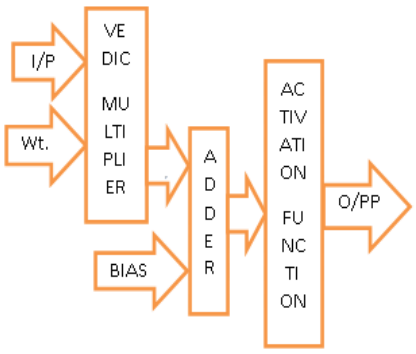

Figure 4. Vedic neuron structure

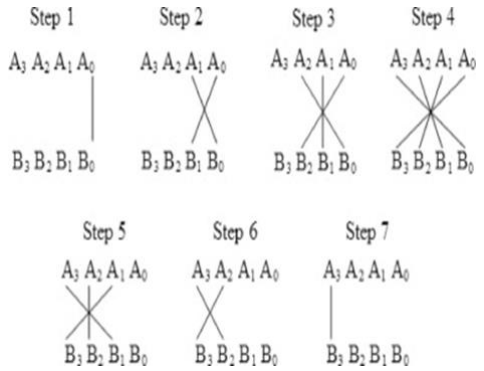

Figure 5. Illustration of 4-bit Vedic Multiplication

4. CSD NEURONAL HALF ADDER

Vedic neuron structure as discussed in the last section can become CSD neuron structure if Vedic multiplier is replaced with CSD multiplier [3]. With CSD speed of computation can be further improved. Review of CSD characteristics along with CSD multiplication algorithm is as explained in following sub sections 4.1 and 4.2 respectively.

4.1 Review of CSD ALGORITHM

In conventional multiplier the number of addition operations required is just oneless than of the number of nonzero bits in the multiplier number. To reduce the multiplier number the power consumption and area, the constant number can be coded so that it contains minimum nonzero bits. This can be done by representing the number in Canonic Signed Digit form. The characteristics of CSD representation are shown below [2, 3,4].

a A CSD number doesn't contain consecutive bits are nonzero.
b The CSD number contains minimum number of nonzero bits, hence the name canonic.
c The CSD representation is unique for a given number.
d CSD numbers cover the range -4/3 to 4/3 in which the values in the range [-1,1) are of great interest.
e The number of nonzero bits in the range of [-1,1) for W-bit CSD numbers is W/3+1/9+O(2-w).
f Therefore CSD number contains around 33% fewer nonzero bits than normal numbers.

The algorithm for converting binary number to CSD number is presented below. The binary A is represented as A=aW-1aW-2 ….a1a0 and its CSD number is C= cW-1cW-2 ….c1c0.

```
a-1=0, y-1= 0,aW=aW-1

for (i= 0 to W-1)
{
Qi=aixor ai-1

yi=yi-1 and Qi

ti=ai+1 and yi

ci=1- t i - ti    }
```

4.2. CSD Multiplication

Multiplication is the primitive arithmetic operation involved in this work and this will decide performance of the entire design. Thus we have considered CSD multiplication algorithm in which partial products are less so that power dissipation is reduced and speed of computation is improved. The Figure 7 shows the 4 bit CSD multiplication. Here, multiplier and multiplicand are taken as 4-bits each 4-bit multiplier is converted to 8-bit CSD number 2 bits to each bit of multiplier by Bit-by-Bit CSD converter module and immediately send it to multiplication block as shown in Figure 7 [3]. Internal process within the block of multiplication of. CSD form of Multiplier contains '00' or '10' or '11' representing 0 or+1 or -1 respectively. When it is '10' or '11' result is addition of earlier result and multiplicand shifted by the count value. This is repeated until last 2 bits. Final product will be stored in result.Figure 6 CSD Neuron Structure as shown in:

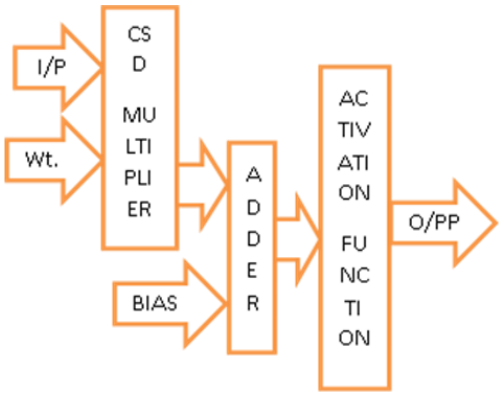

Figure 6. CSD Neuron Structure

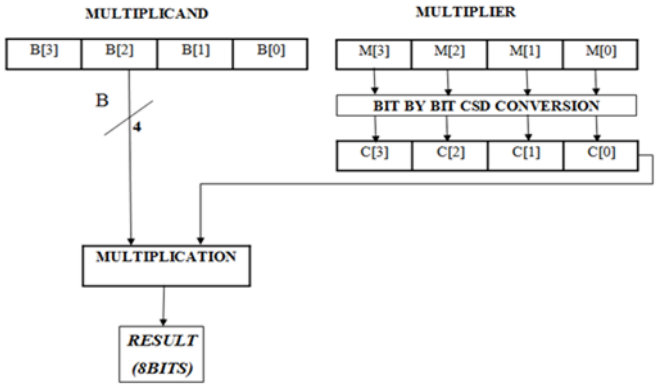

Figure 7. Illustration of 4-bit CSD multiplication

5. RESULTSAND DISCUSSION

RTL schematic for AND gate and XOR gate are as shown in Figure 8 and Figure 9 respectively below. These two gates are combined to form half adder. Performance parameters of half adder with conventional, Vedic multiplier and CSD multiplier are tabulated in Table 1 CSD multiplier based half adder has less delay than both Conventional and Vedic multiplier and small power reduction compared to those methods but at the cost of some increased resources [5,6].

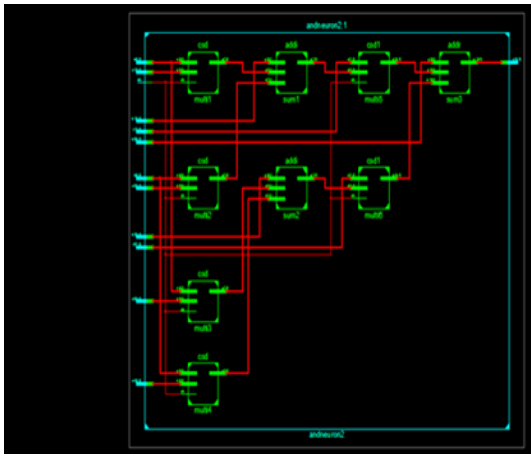

Figure 8. RTL Schematic diagram for CSD based Neuronal AND gate (carry part of half adder)

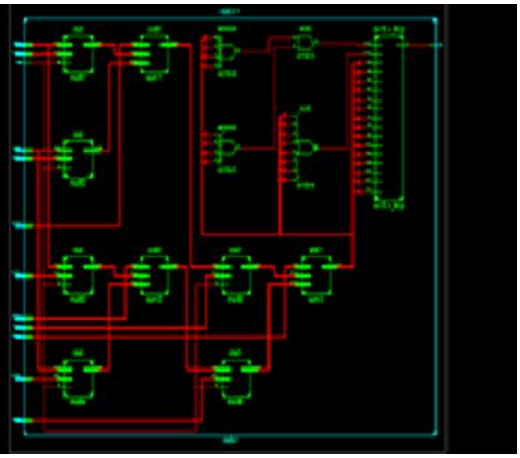

Figure 9. RTL Schematic diagram for CSD based Neuronal XOR gate (Sum part of half adder)

Table 1. Comparison of performance parameters for half adders with different multipliers

PARAMETER	CONVENTIONAL	VEDIC	CSD
Power	3.87 watts	3.85 watts	3.81 watts
Delay	32.969 ns	25.675ns	18.23 ns
No. of flip flops	54/126800	40/126800	62/126800
4 bit Slice LUT's	83/63400	70/63400	172/63400

6. CONCLUSION& FUTURE SCOPE

In this paper Neuronal CSD Half adder is proposed and implemented on FPGA. It is observed CSD half adder faster in performance with comparable low power consumption with that of other two multipliers. This work can be extended to implement any other combinational circuits.

REFERENCES

[1] Anshika, Yamuna SV, NidhiGoel and S.Indu (2015,Sep). Neural Logic Gates Realization using Vedic Mathematics. *Paper presented at the IEEE conference on Next Generation Computing Technologies*, Dehradun, India.

[2] M.Lakshmi Kiran and K.Venkata Ramanaiah. (2017). Implementation of High Speed and Low Area Digital Radix-2 CSD using Pipeline Concept.*International Journal of Electronics and Communication Engineering*. 10(1). pp. 53-61.

[3] Keshab K.Parhi.(1999). VLSI Digital Signal Processing Systems: Design and Implementation, John Wiley and sons Publishing Company Inc., India .

[4] M.LakshmiKiran and K.V.Ramanaiah. (2017). Design and Implementation of High Speed Radix-2 CSD based Floating Point Multiplier. *International Journal of scientific research in multidisciplinary studies*. 3(7). pp.17-22.

[5] Shoab Ahmed Khan. (2010). Digital Design of signal processing systems: A practical approach, John Wiley & sons Publishing Company, UK.

[6] Neelu Farhan, Ann Louisa Paul J., Naadiya Kousar L S., Devika S. and Prof. Ruckmani Divakaran.(2016). Design and Implementation of Logic Gates and Adder Circuits on FPGA using ANN. *International Journal for Research in Applied Science & Engineering Technology*. 4(5), 623-629.