IACSIT International Journal of Engineering and Technology, Vol.2, No.3, June 2010
ISSN: 1793-8236

A Shannon Based Low Power Adder Cell for Neural Network Training

K.Nehru, *Member, IACSIT*, Dr.A.Shanmugam, S.Deepa and R.Priyadarshini

***Abstract*—The proposed full adders for low power and high performance neural network training circuits has been implemented using Shannon decomposition based technique for sum and carry operation. The hardware includes multiplier circuit for product term and an adder circuit to perform summation. The proposed full adder is designed using tanner EDA tools and the resulting parameters such as 25.6% improvement in power dissipation and 20% improvement in transistor count from the simulated output when compared with MC IT based adder cell.**

***Index Terms*—MCIT, Shannon adder cell, power, area, propagation delay.**

I. Introduction

The blooming development of Computer Science has led to the growth of integrated circuit (IC) devices. Most of the Very Large Scale IC (VLSI) applications, such as digital-signal processing and microprocessors, use arithmetic operations extensively [2]. In addition, among these widely used operations, subtraction and multiplication are most commonly applied. The 1-bit full adder is the building block of these operation modules. Therefore, enhancing its performance is crucial to ameliorating the performance of overall modules [3]. Meanwhile, as the widespread use of portable IC devices, such as MP3 players, mobile phones and PDAs etc., IC engineers are required to improve the performance of existing operation modules in some aspects, especially in power depletion and size. Since the battery technology available does not advance at the same rate as the microelectronics technology, IC designers have encountered more constraints: high speed, high throughput, small silicon area, and at the same time, low power dissipation. Hence, the research of establishing low power, high performance adder cells is becoming feverish. One efficient method to accomplish this task is derived from the structural level. This approach to designing and analysing an adder cell is decomposing it into smaller modules for further analysis and improvement. In this way, an optimized full adder cell can be constructed by connecting these improved smaller modules.

There are various logic styles available for high speed and low power design each logic style has its own merits and demerits in terms of power and speed. So among the logic styles available, pass transistor logic is found to enhance the circuit performance. Here gate and source propagates the signal. This logic style has a great functionality that can reduce the number of transistor counts. The PTL can be designed by either using pmos or nmos, but nmos is mostly desired. This has low internodal capacitance effects and so PTL enables low power and high speed digital circuits. The proposed full adder is designed using Sedit and simulated using Tspice. This full adder is used to design efficient neural networks.

Electronics and communication engineering, Bannari Amman Institute of Technology

II. Adder Architecture

The basic idea behind CPL is the use of nMOS network for implementing the logic functions. The complementary logic function can be achieved by passing inverted input signals in CPL. These inverted input signals are directly passed from the input node to the output node. This CPL logic style has least power dissipation per gate during logic transition which makes it ideal for memory circuits. Availability of a number of design tools for CPL makes the design flexible and reduces the design cost. The CPL has certain drawbacks due top source follower action, body effect, limited fan-out capability, high leakage power when not cross coupled. Duality principle enables logic function such as AND-OR, NAND-NOR, XOR-XNOR. By just interchanging the inputs AND, OR basic logic gates, MUX circuits can be constructed [5].

A. Multiplexing Control Input Technique(MCIT)

The MCIT technique is developed by using Karnaugh map from the Boolean expressions for the sum and carry signals from the standard truth table for the full adder circuit.

The Boolean expressions are given as:

$$C = AB + BC + CA \quad (1)$$

$$S = ABC + A'B'C + AB'C' + A'BC' \quad (2)$$

By using expressions (1) and (2), the pass transistor functions can be implemented. When the expression result =0, the pass transistor function is given by the complement of the input variable. If the expression result = 1, the pass transistor function is given by the input variable. To implement the pass transistor function for 'n' input variables, we use n-1 control input data and only one input data. The source input acts as an input of the signal.

The formulae for the ith stage are given as:

$$Ci+1 = Ai\,Bi + (Ai \oplus Bi)\,Ci \quad (3)$$

$$Si = Ai \oplus Bi \oplus Ci \quad (4)$$

The full adder pass transistor using (3) and (4) are given in Fig.1 with A, A', B and B' as the pass transistor inputs and

these are multiplexing control inputs. XOR (A, B) in Fig.2 can be obtained from the stage I (differential node) and the differential node XOR (A, B) and Ci are fed through the MCIT and forms XOR circuit for sum and XNOR for its complement. A duality of pass transistor logic trees and a multiple input logic gate are combined in a full adder to receive the logic signals from the corresponding pass transistor logic trees (differential node) [6].

A complex logic operation can be expressed using this pass transistor function, by which the number of stages in pass transistor logic tree gets reduced and the operation speed gets enhanced. The output expression Ai, Bi and Ci and its complement are obtained at the restoration node. As a result of balancing the adder circuit, the critical path is minimized and bit rate conversion and the operation speed gets improved. Complementary pass transistor logic is pass transistor logic with complementary inputs and outputs available simultaneously.

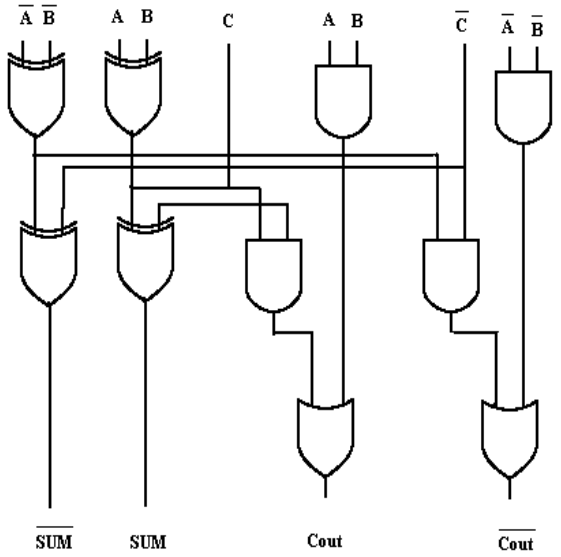

Fig.1. MCIT for a full adder circuit using logic gates

It has solved pass transistor logic threshold voltage loss problem by just complementing the inputs and outputs and adding an inverter at the output. Usually, pull-up pmos transistors are necessary for swing restoration. To minimize the static current due to the incomplete turn-off of the nmos in the output inverter, a weak pmos feedback device added in the CPL circuits in order to pull the pass transistor output to full supply voltage level. The sum circuit of this adder cell is designed by using MCIT as shown by the broken encircled area in Fig.2.

B. *Shannon Theorem*

According to this theorem any logic expression is divided into two terms. One with a particular variable set to 1and multiplying it by a variable and then set the variable to 0 and multiplying it by the inverse. The fullest reduction can be obtained by continuously repeating the Shannon theorem. This method is useful especially to multiplier and pass transistor circuit design. The Shannon's theorem in a generalised way can be stated as a function of many variables, $f(b_0, b_1, b_2, y, b_i, y, b_n)$ can be written as the sum of two terms, say one with a particular variable a_i, set to 0, and one with it set to 1.

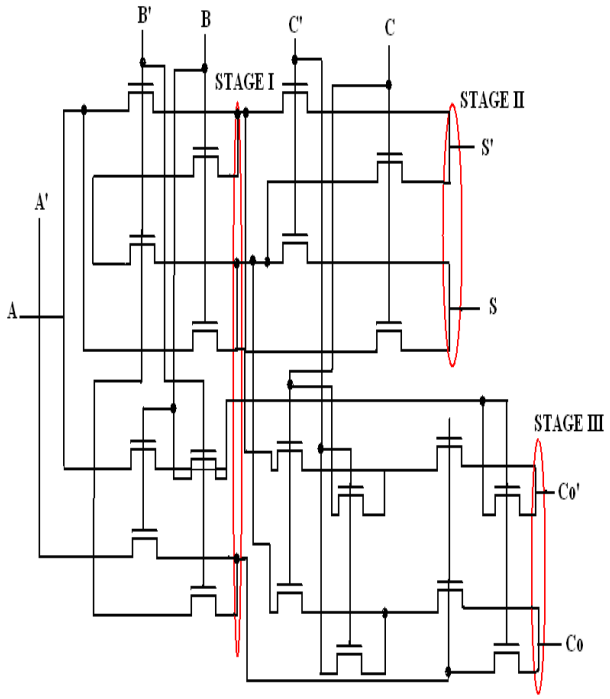

Fig.2. MCIT for a full adder circuit using pass transistor

$$f(b_0, b_1, b_2, \ldots\ldots, b_i, \ldots\ldots y, b_n) = b_i' f(b_0, b_1, b_2, \ldots\ldots, 0, \ldots\ldots y, b_n) + b_i f(b_0, b_1, b_2, \ldots\ldots, 1, \ldots\ldots y, b_n) \quad (5)$$

Shannon's theorem is applied to the logical function using n-1 variables as control inputs and three data lines set to a logical '1'. These source inputs are then connected to the V_{DD} lines (logical '0'), which are connected to the ground. The remaining n^{th} variable is connected from the data input to the source input. The data signals flow horizontally and the control signals flows vertically. Remove pairs of transistors when they cancel each other. The Shannon expression output depends upon the pass logic '1' or logic '0'. If it has logic'0' then the connection input is given by 0 and by '1' for the connection input '1'[5].

C. *Existing Shannon Based Adder Cell*

The existing full adder was designed based on "(1)" and "(6)" by combining the MCIT technique for sum and Shannon operation for carry.

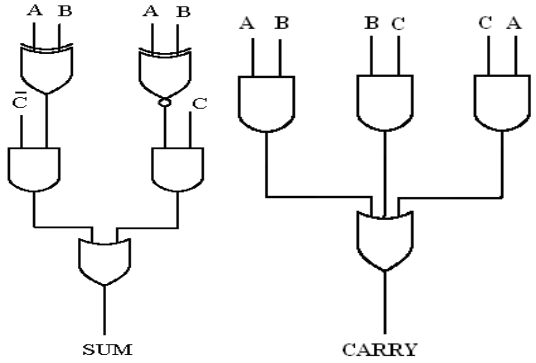

Fig.3. Shannon based full adder using logic gates

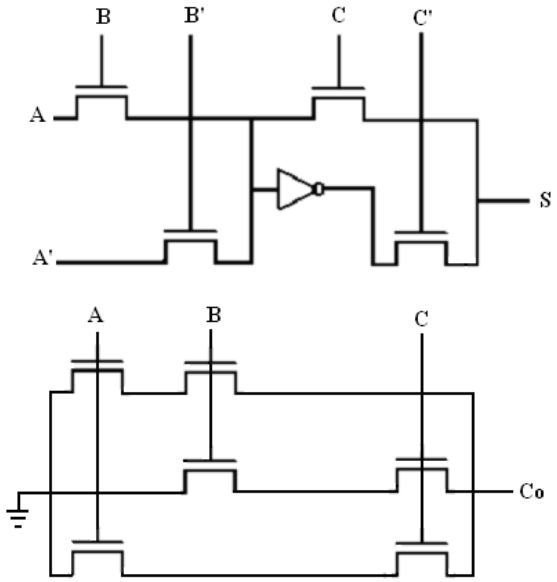

Fig.4. Shannon based adder using pass transistor logic

An input B and B' are used as the control signal of the sum circuit, which is depicted in Fig.3 and Fig.4.

$$S = (A \oplus B)\bar{C} + \overline{(A \oplus B)}C \qquad (6)$$

The equation (2) is modified as (6) and the adder is designed to reduce the transistor count to six in the sum circuit. The C and C' are the differential nodes of the circuit. The carry circuit is designed using the Shannon complementary pass transistor logic, and uses only the inputs A, B, and C. It has been designed using the fundamental Shannon equation "(5)", the source inputs are connected with logic'1', yielding an always 'ON' condition for the transistor. The actual inputs AB, BC, and CA are connected in parallel to give the output C= AB + BC + CA. In this circuit, all of the pass inputs are connected to VDD line so that the pass gates are always 'ON' [1].

D. Proposed Shannon Based Adder Cell

The proposed full adders have the modified expressions for the carry as:

$$C = \overline{(A \oplus B)}B + (A \oplus B)C \qquad (7)$$

By using Shannon's theorem the sum and the carry expressions are condensed and thereby the transistor count has decreased. In the existing design of full adder the carry was generated using six transistors where as the proposed full adder design uses only two transistors. Thus the total chip area gets minimized; hence the power has also reduced to a considerable amount.

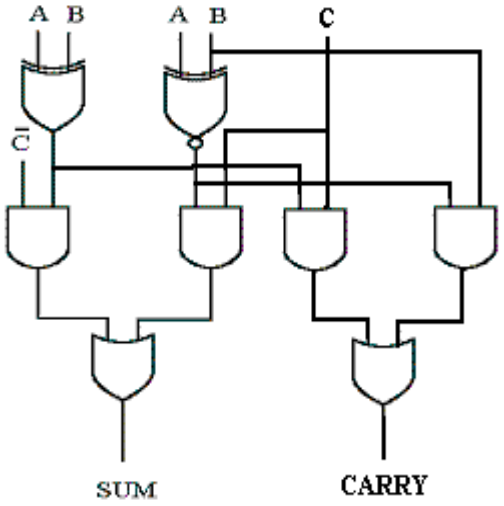

Fig.5. Proposed Shannon based adder using logic gates

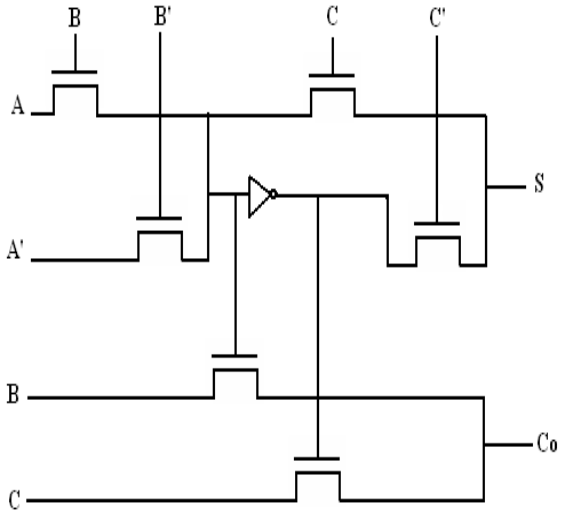

Fig.6. Proposed Shannon based adder using pass transistor logic

III. NEURAL NETWORKS

Neural network research is motivated by two desires: to obtain a better understanding of the human brain and to develop computers that can deal with abstract and poorly defined problems. Many different neural network structures have been tried, some based on imitating what a biologist sees under the microscope, some based on a more mathematical analysis of the problem. The most commonly used structure is shown in fig.7. This neural network is formed in three layers, called the input layer, hidden layer, and output layer. Each layer consists of one or more nodes, represented in this diagram by the small circles. The lines between the nodes indicate the flow of information from one node to the next. In this particular type of neural network, the information flows only from the input to the output (that is, from left-to-right). Other types of neural networks have more intricate connections, such as feedback paths [8].

The nodes of the input layer are passive, meaning they do not modify the data. They receive a single value on their input, and duplicate the value to their multiple outputs. In comparison, the nodes of the hidden and output layer are active.

The most common technique in the training method named back propagation. In this, each input should be assigned with an initial set of weight. Then each input set from the training data is supplied to the network. This corresponding output is tested and compared to the expected output. A function of the error is used to update the coefficients. After testing all of the input sets data, the network retests all of the data. This continues until the network determines that the coefficients are supplying the best output possible and it does not generate the optimal network architecture. This method can potentially required millions of iterations before the final transfer function are chosen [8].

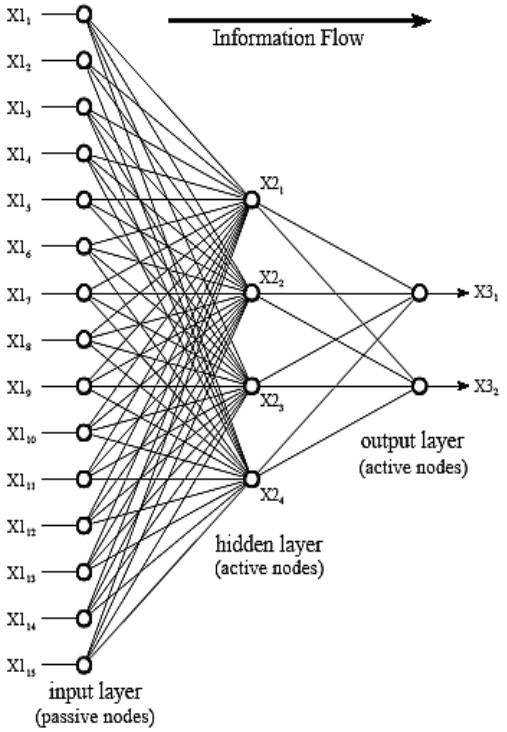

Fig 7. Neural network architecture

IV. Neural Netwoks Architecture Withadder

The hardware implementation of the neural network consists of the multiplier circuit for the product term and adder circuit for the summation term [1].

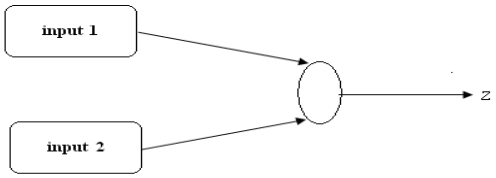

Fig.8. Basic Neuron Representation

$$Z = m_0+m_1n_1+m_2n_2+m_3n_1^2+m_4n_2^2+m_5n_1n_2 \quad (8)$$

V. Results And Discussions

Fig 9 shows the schematic of the modified Shannon based sum and carry of full adder. From the simulation results, it was shown that proposed method is better than the existing method in terms of power consumption and transistor counts.

TABLE I. Comparison of Full Adders

Type of adder cell	Power consumed (w)	Number of MOSFET's
MCIT based adder cell	1.425765e-003	80
Shannon	1.153056e-004	18
Modified Shannon	**0.896820e-004**	**14**

Table I shows the output obtained from the MCIT technique and the Shannon based adder cell by the simulation through the tanner tool.S-Edit is used for schematic design and T-Spice for simulation.

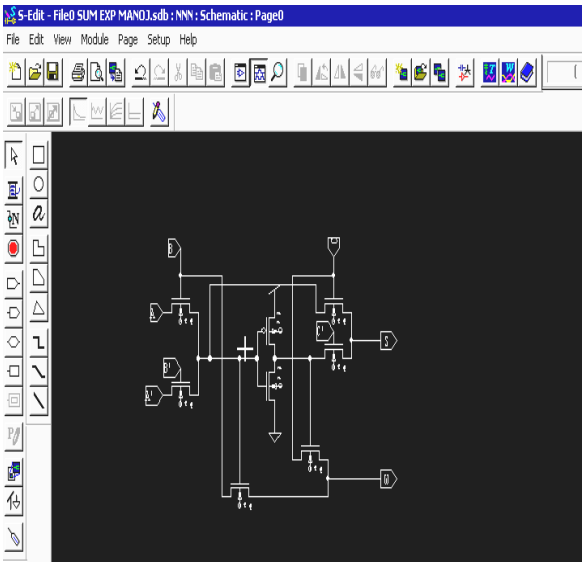

Fig.9.Schematic for modified Shannon full adder

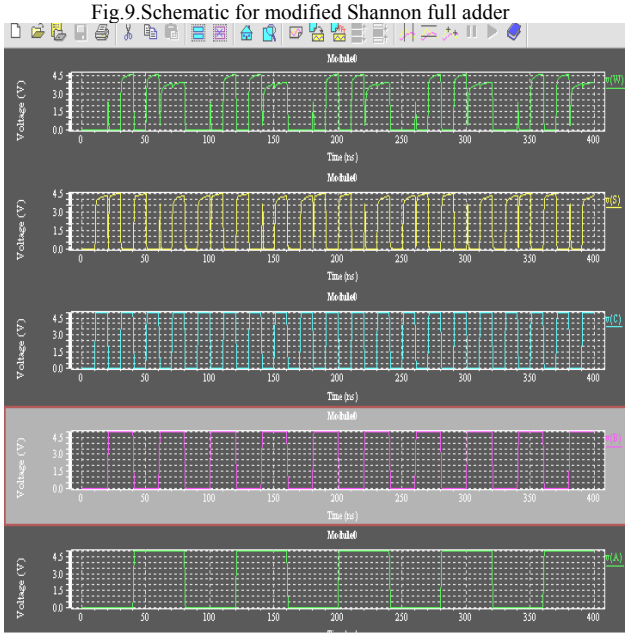

Fig.10.Modified Shannon's carry, sum, A, B, C respectively

VI. Conclusion

The adder cell designed using Shannon based technique has been simulated and results are compared with MCIT based adder cell in terms of power, transistor count. This proposed adder cell having improvement in all these parameters. Due to less number of transistors used in designing the adder cell, the power consumption and occupies area reported to be lower than MCIT adder circuits.

References

[1] S.Saravanan,M.Madheswaran, "Design of low power, high performance area efficient shannon based adder cell for neural network training,''Control,Automation,Communication and Energy Conversation,INCACEC 2009.
[2] J.D. Lee, Y.J. Yoony, K.H. Leez, B.-G. Park, "Application of dynamic pass- transistor logic to an 8-bit multiplier," J. Kor. Phys. Soc.38 (3) (2001) 220–223.
[3] A.Khatibzadeh, K. Raahemifar, "A novel design of a 6-GHz 8x8 bit pipelined multiplier," IEEE Proceedings of the 9th International Database Engineering and Application Symposium (IDEAS'05), 2005, pp. 1–5.
[4] S.V. Kosonocky, A.J. Bhavnagarwala, K. Chin, G.D. Gristede, A.M. Haen, W. Hwang, M.B. Ketchen, S. Kim, D.R. Knebel, K.W. Warren, V. Zyuban, "Low-power circuits and technology for wireless digital systems," IBM J. Res. Dev. 47 (2/3) (2003) 283–298.
[5] D. Markovic, B. Nikolic, V.G. Oklobdzija, "A general method in synthesis of pass-transistor circuits," Microelectr. J. 31 (2000) 991–998.
[6] C.-H. Chang, J. Gu, M. Zhang, "A review of 0.18-mm full adder Performances for tree structured arithmetic circuits," IEEE Trans. Very Large Scale Integration System. 13 (6) (2005) 686–695.
[7] A.G.Ivakhnenko, "Heuristic self-organization in problems of engineering cybernetics," Automatica, 6(2), 1970.
[8] M.C.Acock, Y.A.Pachepsky, "Estimating missing weather data for agricultural Simulations using group method of data handling," Journal of Applied meteorology, 39(2), pp.1176-1184, 2000.
[9] T.Kondo, A.S.pandya, J.M.Zurada, "logistic GMDH-type notification neural networks and their application to the identification of the X-ray film characteristic curve, "proc. of IEEE international Conference on System, Man, and Cybernetics, pp.437-442, 1999.
[10] T.Kondo, A.S.Pandya and H.Nagashino, "GMDH-type neural network algorithm with a feedback loop for structural identification of RBF neural network, "International Journal of Knowledge-Based Intelligent Engineering systems, 11(7), pp.157-168, 2007.
[11] T.Kondo, J.Ueno and, A.S.Pandya, "Multi-layered GMDH-type neural networks with radial basis functions and their application to the 3-dimentional medical image recognition of the liver," Journal of Advanced Computational Intelligence, 11(7), pp. 157- 168, 2007.
[12] T.Konda, A.S.Pandya, "GMDH-type neural network algorithm with sigmoid functions," International Journal of Knowledge-Based Intelligent Engineering systems, 7(4), pp.198-205, 2003.

K. Nehru received his B.E. Degree in Electronics and Instrumentation Engineering from Anna University, Tamilnadu, India in 2005 and M.E. Degree in VLSI design from Anna University, Tamilnadu, India in 2007. He is currently working towards the Ph.D degree in Electronics and Communication Engineering in Anna University, Coimbatore.Tamilnadu, India and working as a Lecturer in ECE Department, Bannari Amman Institute of Technology, Tamilnadu, India.

Dr. A. Shanmugam received the BE Degree in PSG College of Technology in 1972, Coimbatore and ME Degree from College of Engineering, Guindy, Chennai in 1978 and Doctor of Philosophy in Electrical Engineering from Bharathiar University, Coimbatore in 1994.From 1972–76, he worked as Testing Engineer in Testing and Development Centre, Chennai. He was working as a Lecturer Annamalai University in 1978. He was the Professor and Head of Electronics and Communication Engineering Department at PSG College of Technology, Coimbatore during 1999 to 2004. Authored a book titled "Computer Communication Networks" which is published by ISTE, New Delhi, 2000.He is currently the Principal, Bannari Amman Institute of Technology, Sathyamangalam..He is on the editorial board of International Journal Artificial Intelligence in Engineering & Technology (ICAIET), University of Malaysia, International Journal on "Systemics, Cybernetics and Informatics (IJSCI)" Pentagram Research Centre, Hyderabad, India. He is member of the IEEE, the IEEE computer society.