International Journal of VLSI System Design and Communication Systems

ISSN 2322-0929
Vol.04, Issue.10,
October-2016,
Pages:1062-1065

www.ijvdcs.org

Design and Implementation of 16-Bit MAC Ternary Multiplier

K. SEETA RAMA RAJU[1], P.SURYA KUMARI[2]

[1]PG Scholar, Dept of ECE, Godavari Institute of Engineering and Technology(A), Rajahmundry, AndhraPradesh, India, Email: seetaramaraju.4@gmail.com.

[2]Assistant Professor, Dept of ECE, Godavari Institute of Engineering and Technology(A), Rajahmundry, AndhraPradesh, India, Email: suryakumari.peesapati@gmail.com.

Abstract: This Paper Proposed for 16 bit Ternary Multiplier using MAC Unit. The efficiency in terms of area and speed of proposed MAC unit architecture is observed through reduced area, low critical delay and low hardware complexity. The proposed MAC unit reduces the area by reducing the number of multiplication and addition in the multiplier unit. Increase in the speed of operation is achieved by the hierarchical nature of the Ternary multiplier unit. The proposed MAC unit is implemented on a field programmable gate array (FPGA) device, XC3S200-TQ144-4 (Spartan 3). The performance evolution results in terms of speed and device utilization are compared with earlier MAC architecture. The total operation is coded with Verilog, synthesized and simulated using Xillinx ISE 10.1.

Keywords: Ternary Numbers. Multiplier, FPGA, MAC Unit.

I. INTRODUCTION

The general MAC architecture consists of a conventional multiplier, adder and an accumulator. Where the output is added to the previous MAC output result by an accumulate adder. The Multiply-Accumulate (MAC) unit is extensively used in microprocessors and digital signal processors for data-intensive applications, such as filtering, convolution, and inner products. Most digital signal processing methods use nonlinear functions such as discrete cosine transform (DCT) or discrete wavelet transform (DWT) or FFT/IFFT computations that can be efficiently accelerated by dedicated MAC units. Because they are basically accomplished by repetitive application of multiplication and addition, the speed of the multiplication and addition determines the execution speed and performance of the entire computation. As the multiplier exhibits inherently long delay among the basic operational blocks in digital system, the multiplier determines the critical path. In order to improve the speed of the MAC unit, there are two major bottlenecks. The first is the partial products reduction network that is used in the multiplication block and the second is the accumulator. Both of these stages require addition of large operands that involve long paths for carry propagation. The main key to the proposed architecture is using the Ternary multiplier to design the MAC unit and compare the performance with the conventional MAC units in terms of area, speed and number of resources.

It is well known fact that the speed of MAC is governed by the speed of the multiplier The remainder of this paper is organized as follows, SectionII describes Overview of Ternary Mathematics. SectionIII Review of Multipliers. SectionIV describes the MAC architecture unit. SectionV describes the Results and Discussion. SectionVI presents the Conclusion drawn in this work.

II. MAC UNIT

Multiply-accumulate operation is one of the basic arithmetic operations extensively used in modern digital signal processing (DSP). The MAC unit provides high-speed multiplication, multiplication with cumulative addition, multiplication with cumulative subtraction, saturation, and clear-to-zero functions. Therefore, the main motivation is to investigate various pipelined MAC architectures and circuit and the design techniques which are suitable for the implementation of high through put signal processing algorithms. Hence, a high-speed MAC that is capable of supporting multiple precisions and parallel operations is highly desirable. MAC is composed of an adder, multiplier and an accumulator. The implementation of the multiplier is in the form of Booth Multiplier. The adder used is Ripple Carry Adder. The layout of this adder is simple which allows for fast design time. The Parallel in Parallel out (PIPO) shift register is used as the accumulator. The inputs for the MAC are to be fetched from memory location and fed to the multiplier block of the MAC, which will perform multiplication and give the result to the adder which will accumulate the result and then will store the result into a memory location. This entire process is to be achieved in a single clock cycle in the architecture of the MAC unit. The MAC design consists of one 8-bit booth multiplier, one 16-bit ripple carry adder & a 17-bit accumulator using PIPO shift register. To multiply the values of A and B, Booth Multiplier is used instead of conventional multiplier because this is simple to design.

However to gain better performance, parallel multipliers are used as they are the fastest but the designs are much more complex. Hence, when regularity, high performance & low power are primary concerns, Booth Multipliers tend to be the primary choice. Apparently, together with the utilization of Booth multiplier approach, ripple carry adder as the adder and PIPO shift register as the accumulator, this MAC design can enhance the MAC unit speed so as to gain better system performance. The product of Xi x Yi is always fed back into the 17-bit accumulator and then added again with the next product Xi x Yi. This MAC unit is capable of multiplying and adding with previous product consecutively. Hence,

$$\text{output} = \sum XiYi \quad (1)$$

The design of 16 bit multiplier unit is carried out that can perform accumulation on 17 bit number. This MAC unit has 33 bit output and its operation is to add repeatedly the multiplication results. The total design area can be inspected by observing the total count of transistors. Several other parameters can be calculated as well. Figure 1 shows the basic block diagram of the MAC unit.

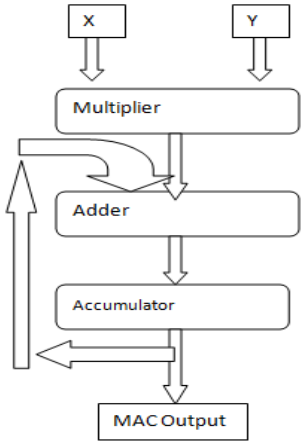

Fig1. MAC Unit.

III. PROPOSED TERNARY MULTIPLIER

Multiplication of n-bit ternary number requires generation of partial product, shifting operations & finally addition of partial product. Implementation of multiplier blocks as a combination of 1-trit ternary multiplier, half and full T-adders. Table (a) shows rules for multiplication, (b) truth table for 1-bit multiplier along with K-Map for 1-trit multiplier. Figure 5 Shows implemented ternary multiplier blocks. Truth table for the same is given in Appendix II from K-map, the output equation of multiplier is:

$$F_{mul} = A_0^2+B_0^1+B_0^2A_0^1+1.(A_0^1B_0^1+A_0^2B_0^2)$$

$$F\ carry = A_0^2+B_0^2$$

Table1: Rules for multiplication

A	B	Product	Carry
0	0	0	0
0	1	1	0
0	2	2	0
1	1	1	0
1	2	0	1
2	2	1	1

Table2. 1-Bit Ternary multiplication truth table

A	B	Product	Carry
0	0	0	0
0	1	0	0
0	2	0	0
1	0	0	0
1	1	1	0
1	2	2	0
2	0	0	0
2	1	2	0
2	2	1	1

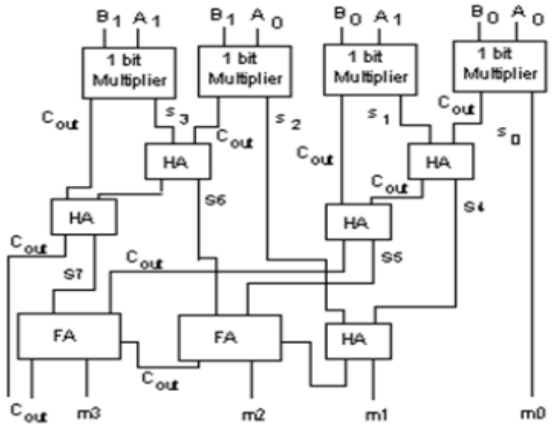

Fig2. Bit Ternary Multiplier Circuit.

A. Ternary Half Adder

Ternary half adder is a circuit for the addition two 1 trit numbers is referred to as a half adder. Circuit does not consider a carry generated in the previous addition. Table 3 expresses the addition process in ternary logic system. Here A and B are two inputs and sum(S) and carry (C) are two outputs. Since no grouping of 2's and 1'is possible, the output equation is as below.

$$\mathbf{Sum} = \mathbf{A^2B^1 + A^1B^1 + A^0B^2 + 1 * (A^1B^0 + A^0B^1 + A^2B^2)}$$

$$\mathbf{Carry} = \mathbf{A^2B^1 + A^1B^2 + A^2B^2}$$

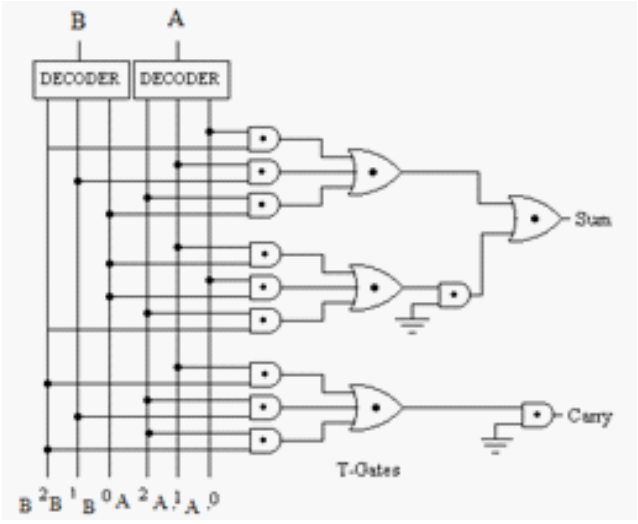

Fig3. Half Adder Circuit.

Table 3. Half adder Truth table

A	B	SUM	CARRY
0	0	0	0
0	1	1	0
0	2	2	0
1	0	1	0
1	1	2	0
1	2	0	1
2	0	2	0
2	1	0	1
2	2	1	1

B. Ternary Full adder

A ternary full adder is a circuit that adds two inputs and previous carry generated.

$$\begin{aligned}\mathbf{Sum} = {} & \mathbf{A^2B°Cin° + A^1B°Cin^1 + A°B°Cin^2 + A^1B^1Cin°} \\ & \mathbf{+ A°B^1Cin^1 + A^2B^1Cin^2 + A°B^2Cin°} \\ & \mathbf{+ A^2B^2Cin^1 + A^1B^2Cin^2 + 1 * (A^1B°Cin°} \\ & \mathbf{+ A°B°Cin^1 + A^2B°Cin^2 + A°B^1Cin°} \\ & \mathbf{+ A^2B^1Cin^1 + A^1B^1Cin^2 + A^2B^2Cin°} \\ & \mathbf{+ A^1B^2Cin^1 + A°B^2Cin^2+)}\end{aligned}$$

$$\begin{aligned}\mathbf{Carry} = {} & \mathbf{A^2B^2Cin^2 + 1 * (A^2Cin^2 + A°B^2 + A^2B^2} \\ & \mathbf{+ A^1Cin^1 + A^2Cin^1 + B^2Cin^1 + B^1Cin^2} \\ & \mathbf{+ B^2Cin^2 + A^1B^1Cin^1)}\end{aligned}$$

Table4. Full adder truth table

A	B	Cin	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	0	2	2	0
0	1	0	1	0
0	1	1	2	0
0	1	2	0	1
0	2	0	2	0
0	2	1	0	1
0	2	2	1	1
1	0	0	1	0
1	0	1	2	0
1	0	2	0	1
1	1	0	2	0
1	1	1	0	1
1	1	2	1	1
1	2	0	0	1
1	2	1	1	1
1	2	2	2	1
2	0	0	2	0
2	0	1	0	1
2	0	2	1	1
2	1	0	0	1
2	1	1	1	1
2	1	2	2	1
2	2	0	1	1
2	2	1	2	1
2	2	2	2	2

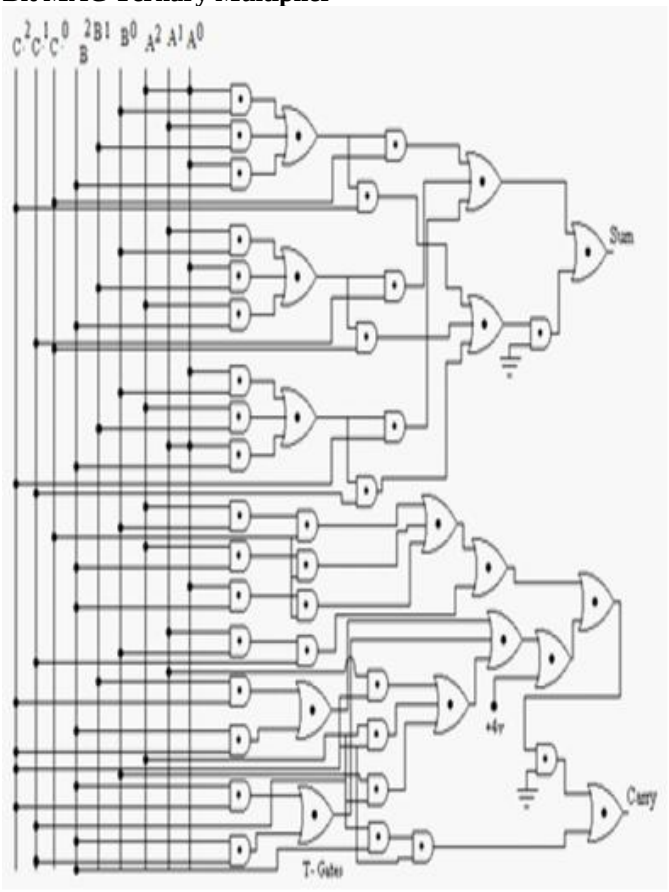

Fig4. Full adder Circuit Design.

IV. RESULTS

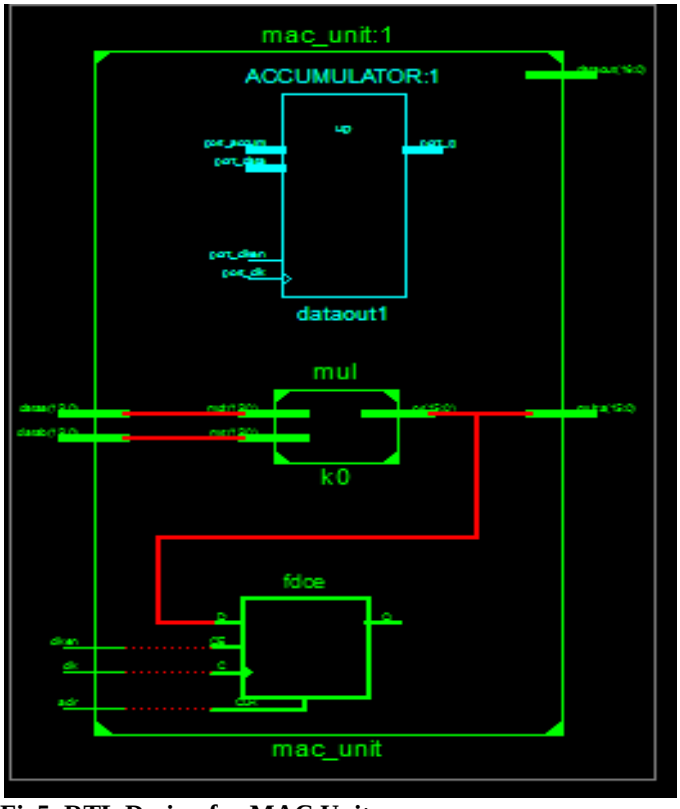

Fig5. RTL Design for MAC Unit.

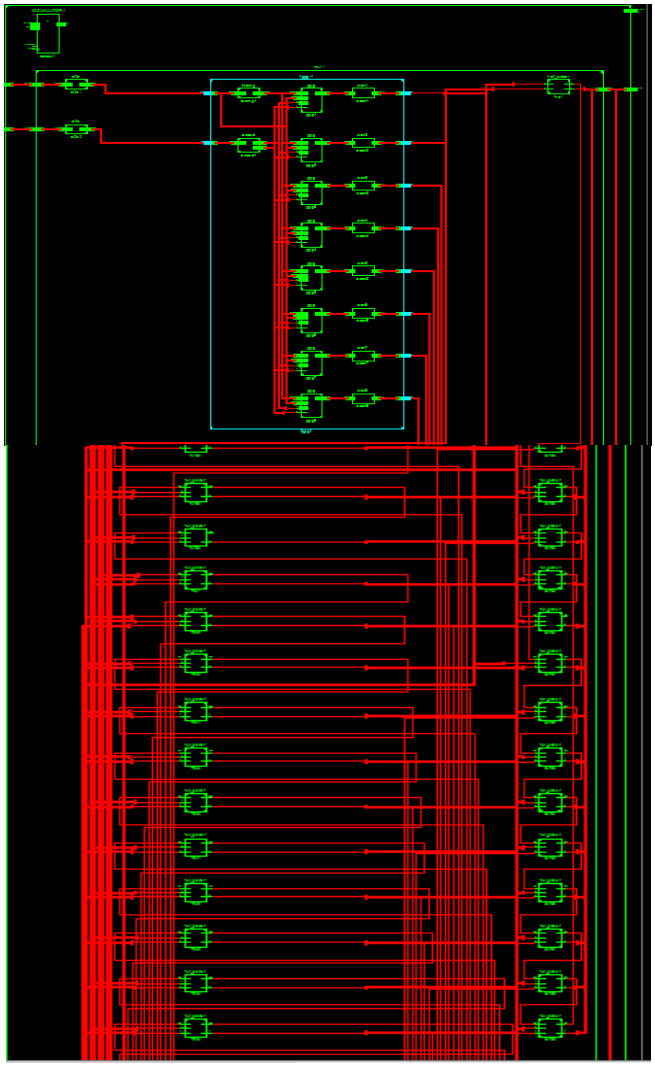

Fig6. RTL Design for Ternary Multiplier.

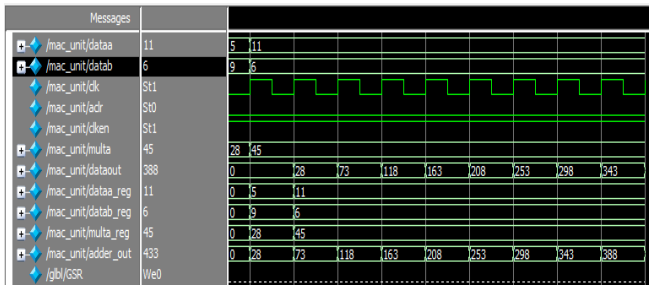

Fig7. Waveform for Ternary Multiplier using MAC.

Table5. Power Analysis

Parameter	Result
Power	1.3mW
Delay	35.602ns
LUTs	303 out of 1028

V. CONCLUSION

To develop ternary circuits & to verify their performance, tools for digital simulation, layout generation & extraction are necessary. Supported by two typical examples the proposed work describes basic procedure & establishes Verilog as potential tool for simulation of ternary circuits and systems. Further, in proposed work additional features can be incorporated to provide enough information to verify timing information. The proposed simulator can be used to synthesis & to verify the performance of ternary logic circuits.

VI. REFERENCES

[1] A.S. Kumar and A.S. Priya, "Minimization of Ternary Combinational Circuits - A Survey", International Journal of Engineering and Technology, vol. 2, pp. 3576-3589, 2010.

[2] A.P. Dhande and V.T. Ingole, "Design and Implementation of 2-Bit Ternary ALU Slice", 3rd International Conference SETIT, vol. 1, pp. 1- 11, 2005.

[3] S. Das, P. Dasgupta and S. Sensarma, "Arithmetic Algorithms for Ternary Number System", Progress in VLSI Design and Test, Springer mBerlin Heidelberg, vol. 7373, pp. 111-120, 2012.

[4] Chung-Yu-Wu, "Design & Application of Pipelined Dynamic CMOS Ternary Logic & Simple Ternary Differential Logic", IEEE Journal on Solid State Circuits, vol. 28, no. 8, 1993.

[5] D.S. Reddy, J. Venkatesh and P. Karthik, "Design and Implementation of High Performance Two's Complement Multiplier", International mJournal of Latest Trends in Engineering and Technology (IJLTET), vol. 2, pp. 441- 446, 2013.

[6] K.N. Vijeyakumar, V. Sumathy, M.G. Devi, S. Tamilselvan and R.R. mNair, "Design of Hardware Efficient High Speed Multiplier using mModified Ternary Logic", Procedia Engineering, vol. 38, pp. 2186- m2195, 2012.

[7] A.P. Dhande, R.C. Jaiswal and S.S. Dudam, "Ternary Logic Simulator musing VHDL", 4th International Conference on Sciences of Electronic m(SETIT), vol. 1, pp. 25-29, 2007.

[8] A.P. Dhande and V.T. Ingole, "Design of 3-Valued R-S & D Flip-Flops mBased on Simple Ternary Gates", International Journal Of Software Engineering & Knowledge Engineering, vol. 15, no. 2, pp. 411-417, 2005.

[9] B.V.S. Vidya and T. Kirankumar, "A Novel Approach to Ternary mmultiplication", IEEE International Conference on Computer Communication and Informatics (ICCCI), vol. 1, pp. 1-4, 2012.

[10] wikipedia.com

[11] W.C. Yeh and C.W. Jen, "High-speed Booth Encoded Parallel Multiplier mDesign", IEEE Transactions on Computers , vol. 49, no. 7, pp. 692-701, m2000.

[12] W. Pengjun, L. Kunpeng and M. Fengna, "Design of Ternary Adiabatic mMultiplier on Switch-Level", Journal of Electronics, China, vol. 28, no. m3, pp. 375-382, 2011.