

A 32 x 32 Booth Multiplier using 48 bit CLA Adder

T. Vimal Prakash Singh

Department of Electrical Engineering, NERIST, Nirjuli
E-mail: vimal@nerist.ac.in

Abstract—In this paper, design of a 32x32 bit multiplier using modified Booth algorithm is described. Critical path delay of a Booth multiplier consists of encoder, selector and addition path of the partial products. Booth encoder and selector were designed to minimize the delay and reduce the hardware cost. Booth selector which is used in the design has a delay of 3 gates. Designed multiplier was synthesized and layout was made using 0.11 μ m Faraday UMC CMOS technology.

Keywords: Booth algorithm, Carry Lookahead adder, Multiplier

1. INTRODUCTION

In many of the high performance audio video digital signal processors, multiplier is used as a building block. Therefore performance of the multiplier is an important parameter from the viewpoint of area, power and energy consumption of these high performance signal processors. In the past, significant efforts have been made in search of techniques for construction of efficient structures. Booth algorithm, pass transistor logic (PTL), Wallace tree have been proposed. While Booth algorithm reduces the partial products by half, full custom design method and standard cell-based method were both applied [1]-[6]. It is also found that booth selectors take up approximately one-third area in a multiplier. Booth selectors take up about one-third area in a multiplier. Inoue's multiplier saves transistors by utilizing sign-select Booth encoding algorithm [2], however, it pays more delay time than some other multipliers, such as Cho's work [6].

This paper describes the design of a booth multiplier based on efficient Booth encoder and decoder structures. Booth encoding structure was designed by looking at three input pattern. Depending upon the encoder output, each bit of the partial product are generated by a decoder. At the end of generation of partial products, a Carry Lookahead Adder (CLA) was used to produce the final product output. In order to increase the speed of the adder, carry lookahead logic was used to compute carry outputs while simultaneously computing sum outputs. In order to compensate the carry output delay for larger bit width, a four bit carry lookahead adder was designed and this 4 bit width CLA adder was used

to obtain higher bit width adder. Finally layout area of 0.11 μ m CMOS Booth multiplier is 415.17 x 401.635.

2. BOOTH ENCODER ALGORITHM

The proposed multiplier uses a Booth encoder block, a decoder block and an adder block which is similar to the conventional multiplier.

The use of Booth algorithm reduces the number of partial products. In a modified Booth encoding scheme, these partial products are even reduced by half. Recoding of Booth algorithm is performed in two stages: encoding and selection.

In the classical Booth encoding scheme, three signals $x_{2i-1}, x_{2i}, x_{2i+1}$ of multiplier bits are used for generating signals S (single), D (Double) and N (Negative) for selecting partial products of one of $0, \pm Y$ and $\pm 2Y$. Here X and Y is the multiplier and multiplicand value of m -bit width. Whether Y and $2Y$ selected depending on the bit patterns of S and D [10].

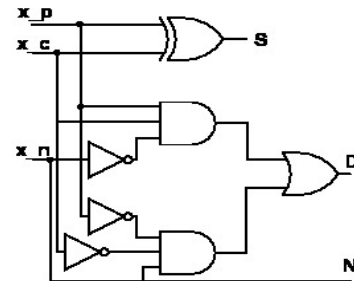


Fig. 1: Booth encoder

Booth encoding scheme used in this paper is shown in the Table 1 and in Fig. 1.

Table 1: Booth Encoder Truth Table

x_{2i+1}	x_{2i}	x_{2i-1}	PP_i	S	D	N
0	0	0	0	0	0	0
0	0	1	Y	1	0	0

0	1	0	Y	1	0	0
0	1	1	2Y	0	1	0
1	0	0	-2Y	0	1	1
1	0	1	-Y	1	0	1
1	1	0	-Y	1	0	1
1	1	1	-0(0)	0	0	1

Using these encoded signals, a Booth selector is used to select each bit of the partial product. The structure of the Booth and encoder and selector is done carefully in order to improve the speed of the encoder and selector delay line. Critical path of encoder and selector is shown in Fig. 2. Since selectors made up significant portion of of the area, they are to be sized over speed.

3. CLA ADDER MODULE

After all the partial products have been generated depending upon the encoder output signals by the Booth selectors, we need to find the sum of all the partial products. So the structure of the adder module plays an important role in terms of area and speed of the multiplier[10]. In this paper, Carry select Adder (CSA) structure is used to compute intermediate summations and finally a 48 bit CLA adder is utilized to compute the final product output.

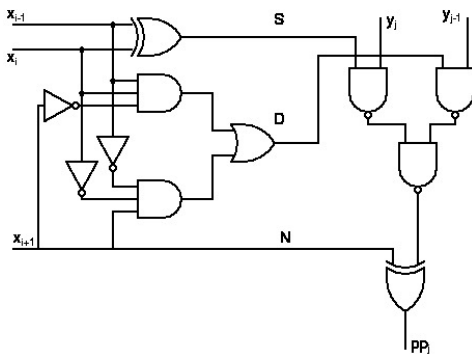


Fig. 2: Booth encoder and selector

The main idea behind the CLA structure is to generate all carry inputs in parallel to avoid waiting for the carry propagated from a Full adder module. The carry propagated at the ith stage is given by the following equation:

$$c_{i+1} = x_i y_i + (x_i \oplus y_i) c_i \quad (1)$$

The generate and propagate terms at the ith stage are as follows:

$$G_i = x_i y_i ,$$

$$P_i = x_i \oplus y_i , c_{i+1} = G_i + P_i c_i \quad (2)$$

A 4 bit CLA adder first design using the equations (1) , (2) and its structure is shown in Fig. 3. Using this 4 bit CLA structure, first a 12 bit CLA adder is designed which is shown

in Fig. 4 and four such CLA module is used to construct 48 bit CLA adder to compute the final product output of the multiplier. While designing 48 bit CLA adder due care is taken to minimize the delay of final carry generation.

4. DESIGN OF BOOTH MULTIPLIER

Classical Booth multiplier is designed in three stages: (1) Booth encoding, (2) Booth selector and (3) Adder. Gate level model of each modules of the multiplier are developed in Verilog HDL and simulated using Cadence NCSim. Gate level HDL simulation of all the stages of Booth multiplier is shown in Fig. 5.

Once it is verified, the working of the model of the Booth multiplier, verilog HDL was synthesized using Cadence RC

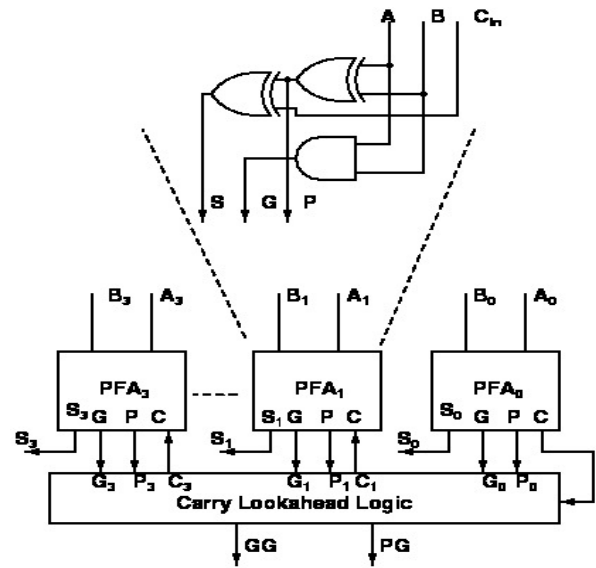


Fig. 3: 4 bit Carry Lookahead adder

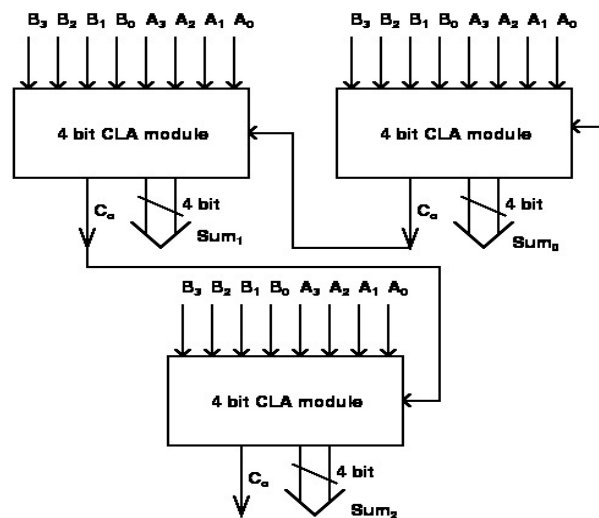


Fig. 4: 12 bit Carry Lookahead structure

compiler using the 0.11 μm UMC CMOS technology. Synthesis tool produces a verilog netlist. It is required to verify the netlist produced is correct for the golden HDL code. Cadence Conformal Logic Equivalence Checker is used to check the equivalence of the golden HDL model and verilog netlist produced by the synthesis tool and result is shown in Fig. 7.



Fig. 5: Verilog simulation of Booth multiplier stages

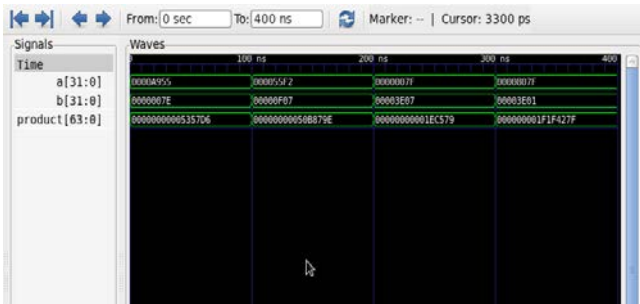


Fig. 6: Simulation waveform of Booth multiplier

Since the golden and revised netlist are same, post synthesis simulation is run again and simulation result is shown in Fig. 8.

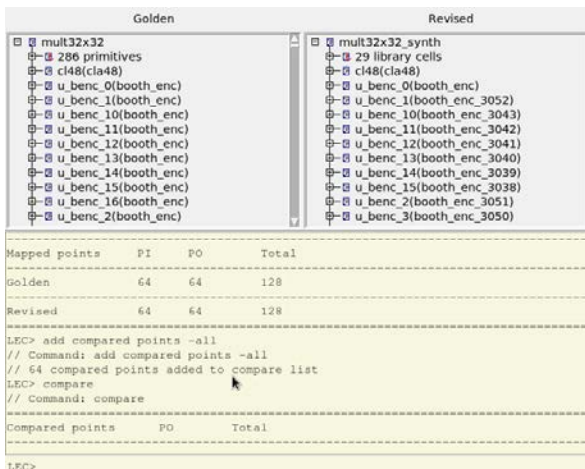


Fig. 7: Conformal LEC

5. IMPLEMENTATION AND RESULT

The verilog model thus developed for the multiplier is synthesized using RC Compiler using 0.11 μm UMC CMOS technology. It is also seen in Fig. 7 that the netlist produced by the Cadence tool is logically equivalent to the golden verilog model of the Booth multiplier. Post synthesis simulation of the multiplier is performed using NCSim and is shown in the Fig. 8.

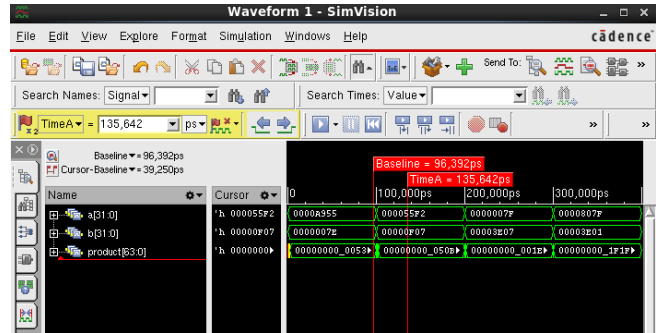


Fig. 8: Post Synthesis Simulation waveform

The area report produced by the RC Compiler is 30361 sq. units and instance power report is shown in Fig. 9. From the instance power report, it is concluded that the 48 bit CLA adder consumes 4.65% of the entire power and CSA modules at partial product levels 11th to 15th equally consume 0.13%.

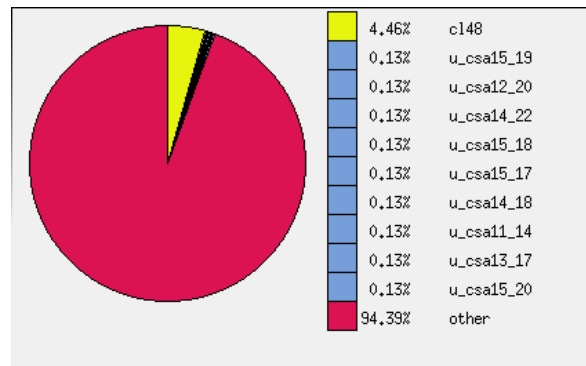


Fig. 9: Instance Power distribution

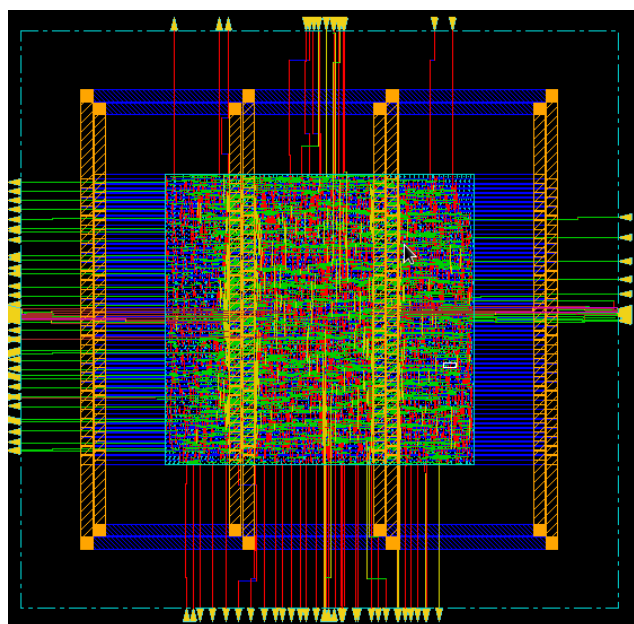
The propagation delay time of the whole multiplier is 1.853 ns at 1.1 V and area of the layout is 415.17 x 401.635 sq. unit. Fig. 10 shows the final layout of the multiplier. Comparison of the designed multiplier with available conventional multipliers in the literature are summarized in Table 2.

6. CONCLUSION

In this paper a 32x32 bit Booth multiplier is designed which can achieve lower propagation delay and smaller area. The improved designed is implemented using 0.11 μm UMC CMOS technology. Simulation of the designed multiplier shows improvements in area as well as speed.

Table 2: Comparison of Multipliers

	Width	Gate Length	Supply Voltage	Area	Delay
Ohkubo[2]	54	0.25	2.5	3.77x3.41	4.4
Inoue[3]	54	0.25	2.5	1.04x1.27	4.1
Cho[5]	54	0.18	2.5	---	3.25
Xinyu[8]	64	0.18	1.8	1.02x1.02	2.82
This paper	32	0.11	1.1	0.4 x 0.4	1.85

**Fig. 10: Layout of the designed multiplier from Encounter****REFERENCES**

- [1] M.Suzuki, N.Ohkubo, T.Shinbo, T.Yamanaka, A.Shimizu, K.Sasaki and Y.Nakagome, "A 1.5-ns 32-b CMOS ALU in double-double pass-transistor logic", *IEEE Journal of Solid-State Circuits*, Vol. 28 , Issue: 11. pp 1145-1151, Nov. 1993
- [2] N. Ohkubo, M. Suzuki, T. Shinbo, T. Yamanaka, A. Shimizu, K. Sasaki and Y. Nakagome, "A 4.4 ns CMOS 54x54-b Multiplier Using Pass-Transistor Multiplexers", *IEEE Journal of Solid-State Circuits*, Vol 30, no. 3 , pp. 251-257, Mar. 1995.
- [3] Inoue. R. Ohe, S. Kashiwakura, S. Mitarai, T. Tsuru, T. Izawa and G. Inoue, "A 4.1 ns compact 54x65 b multiplier utilizing sign select Booth encoders.", *IEEE International Conference on Solid State Circuits 1997*, pp. 416-417, 6-8 Feb. 1997
- [4] Y Hagihara, S. Inui, A. Yoshikawa, S. Nakazato, S. Iriki, R. Ikeda, Y. Shibue, T. Inaba, M. Kagamihara and M. Yamashina, "A 2.4 ns 0.25 μ m CMOS 54x54 b multiplier.", *IEEE International Conference on Solid State Circuits 1998*, pp. 296-297, 5-7 Feb. 1998.
- [5] Ki-seon Cho, Jong-on Park, Jin-seok Hong and Goang-seog Choi, "54x54 Radix-4 Multiplier based on Modified Booth Algorithm", *GLSVLSI 2003*, 00. 233-237
- [6] V. G. Okolobdzija, D. Vileger and S. S. Liu, "A method for speed optimized partial product reduction and generation of fast parallel multipliers using an algorithmic approach", *IEEE Transactions on Computers*, Vol. 45, Issue: 3, pp. 294 -306, March 1996
- [7] Tso- Bing Juang, Jeng-Hsiun Jan, Ming-Yu Tsai and Shen-Fu Hsiao, "Partition methodology for the final adder in a tree structure parallel multiplier generator", *Asia-Pacific Conference on Circuits and Systems*, 2002, Vol. 1, 28-31 pp. 471-474 vol 1, Oct. 2002
- [8] Wu, Xinyu, Chi Huang, Jinmei Lai, and Chenshou Sun. "A 64x64-bit modified Booth multiplier utilizing multiplexer-select Booth encoder." In *ASIC, 2005. ASICON 2005. 6th International Conference On*, vol. 1, pp. 57-60. IEEE, 2005.
- [9] Weste, N. H. E. and David M. Harris., *CMOS VLSI DESIGN: A circuits and system perspective*, Addison-Wesley, 2011.
- [10] Jose, Bijoy, and Damu Radhakrishnan. "Fast redundant binary partial product generators for Booth multiplication." *Circuits and Systems, 2007. MWSCAS 2007. 50th Midwest Symposium on*. IEEE, 2007.