8-Bit Radix-4 Booth Multiplier using GDI Technique

Ankita Dhankar, Satyajit Anand

***Abstract* — An 8-bit radix-4 Booth Multiplier is implemented that demand high speed and low energy operation. It is a good approach if we implement the multiplier as a hybrid architecture of the radix-4/-8 because the radix-8 mode has low power consumption capability, occupying less area and number of partial products obtained in this mode are less(N/3). But the detection of the 3B term while computing the partial products is very difficult and it is difficult to implement it on the FPGA board. So by comparing the performances of the two multipliers we suggest to go with the radix-4 multiplier. Compared to a conventional CMOS Multiplier, the proposed multiplier's power delay product is 10% less with the use of only 1656 transistors in comparison to conventional CMOS circuit, which uses 2782 transistors.**

***Index Terms* - encoder, multiplier, gate-diffusion input (GDI), power consumption, PPG**

I. INTRODUCTION

Multipliers are essential components in signal-processing for multimedia applications. While many previous works focused on implementing high-speed multipliers, recently there have been many attempts to reduce power consumption [1]. This is due to the increased demand for portable multimedia applications, which require low power consumption as well as high-speed operation.

However low-power multipliers without any consideration for high-speed are not the appropriate solutions of low-energy embedded signal processing for multimedia applications [2]-[4].

Previously, a hybrid radix-4/-8 modified Booth encoded (MBE) multiplier was proposed for low-power and high-speed operation [2]-[8]. It is a good approach if we implement the multiplier as a hybrid architecture of the radix-4/-8 because the radix-8 mode has low power consumption capability, occupying less area and number of partial products obtained in this mode are less(N/3) compared to the partial products of the radix-4 mode(N/2). But the detection of the 3B term while computing the partial products is very difficult and it is difficult to implement it on the FPGA board.

So by comparing the performances of the two multipliers we suggest to go with the radix-4 multiplier and the Radix-4 encoder with the partial product generation block is implemented here.

The paper is organized as follows. Section II introduces radix-4 encoder. Section III introduces radix-4 partial product generation block. Section IV introduces GDI circuit methodology. In Section V, we can see proposed radix-4 booth multiplier. Experimental results summarized in Section VI. Finally, conclusion of this paper done in Section VII.

II. RADIX-4 ENCODER BLOCK

Radix-4 multiplication obtains an improvement in the multiplication algorithm due to the less number of partial products entering the Wallace tree to be reduced. This can be achieved by the application of the multiplier recoding, changing from a 2's complement format to a signed-digit representation from the set {0, ±1, ±2}.

Now let us assume that we are giving a triplet of multiplier one at a time in the form of Yi-1, Yi, Yi+1 and gets the encoded output in the form of Neg, X1_b, X2_b, Z. In which 'X2_b' means that the multiplicand is not to be doubled, 'X1_b' means multiplicand is to be doubled and 'Neg' means neither to be doubled nor to be inverted. Fig. 1 shows the block diagram of the radix-4 encoder.

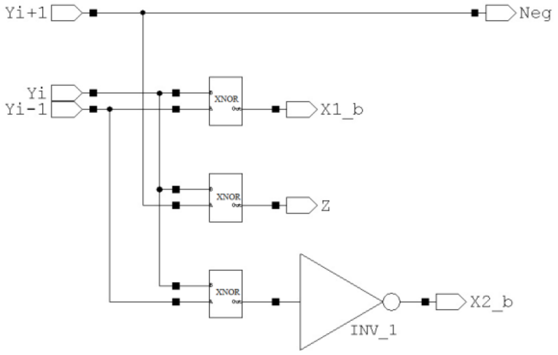


Fig.1 Radix-4 Encoder Block Diagram

III. RADIX-4 PARTIAL PRODUCT GENERATOR

In this block, multiplier bits encoded from the encoder and the multiplicand bits get multiplied and the partial products are generated. Fig. 2 shows the block diagram of 1-bit partial product generator block where the input X1_b, X2_b, Neg, Z are the encoded multiplier bits from the encoder and X_j and X_{j-1} are the multiplicand bits which are to be multiplied by the multiplier bits coming from the encoder [9].

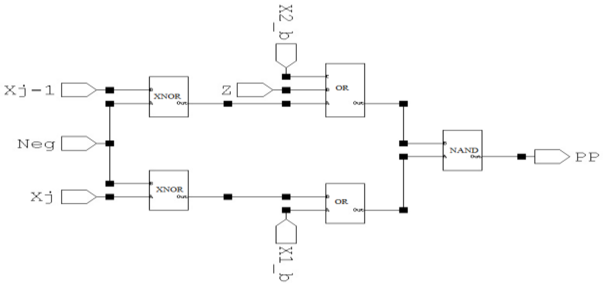


Fig.2 1-BIT Partial Product Generator Block

Fig. 3 shows 9-BIT PPG block made from nine 1-bit PPG blocks and nine half adders. Half adders are to get 2's complemented form.

Manuscript received April, 2013.

Ankita Dhankar, Electronics and Communication Department, M.Tech. VLSI Design, Mody Institute of Technology and Science, Lakshmangarh, India.

Satyajit Anand, Electronics and Communication Department, Mody Institute of Technology and Science, Lakshmangarh, India.

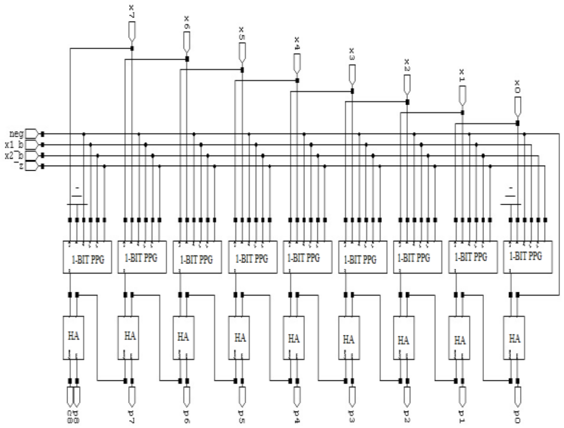


Fig.3 9-BIT PPG Block

IV. GDI CIRCUIT METHODOLOGY

The GDI method [6]–[7] is based on the simple cell shown in Fig. 4. At a first glance the basic cell resembles the standard CMOS inverter, but there are some important differences. The GDI cell contains four terminals – G (the common gate input of the nMOS and pMOS transistors), P (the outer diffusion node of the pMOS transistor), N (the outer diffusion node of the nMOS transistor) and the D node (the common diffusion of both transistors). P, N, and D may be used as either input or output ports, depending on the circuit structure. The GDI approach allows implementation of a wide range of complex logic functions using only two transistors. This method is suitable for design of fast, low-power circuits, using a reduced number of transistors (as compared to CMOS and existing PTL techniques), while improving logic level swing and static power characteristics. Multiple-input gates can be implemented by combining several GDI cells.

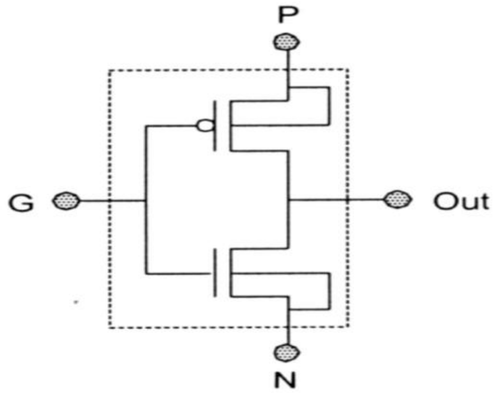


Fig. 4. GDI basic cell

TABLE 1
Logic functions that can be implemented with a single GDI cell

N	P	G	D	Function
'0'	B	A	$\overline{A}B$	F1
B	'1'	A	$\overline{A}+B$	F2
'1'	B	A	$A+B$	OR
B	'0'	A	AB	AND
C	B	A	$\overline{A}B+AC$	MUX
'0'	'1'	A	$\overline{A}$	NOT

Table I shows an example of how simple configuration changes of the inputs P, N, and G in the basic GDI cell correspond to very different Boolean functions at the output D. Most of these functions require a complex (6- 12 transistors) gate in CMOS (as well as in standard PTL implementations [8]), but are very simple (only two transistors per function) in the GDI design methodology.

V. PROPOSED RADIX- 4 BOOTH MULTIPLIER

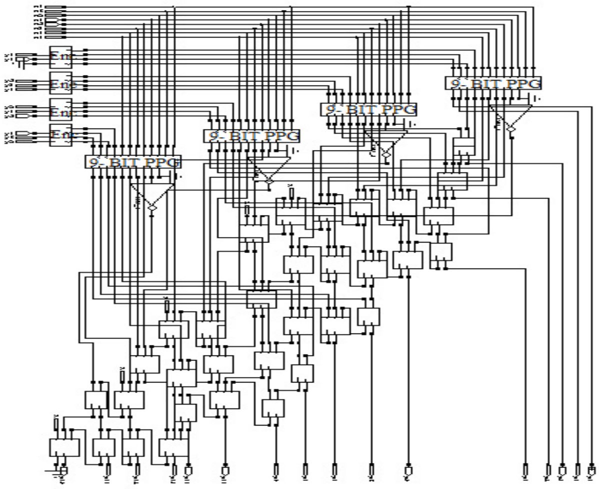


Fig. 5 shows the schematic of a Radix-4 Booth Multiplier.

Multiplier bits first goes into the booth encoder. The booth encoded multiplier bits, the multiplicand bits goes into the 9-bit PPG, the multiplier and multiplicand are multiplied, and four partial products are obtained. These partial products go into the tree of full adders and half adders. The four partial products are adder up by this tree and 16-bit output product was obtained. Fig. 6 shows the symbolic representation of radix-4 booth multiplier.

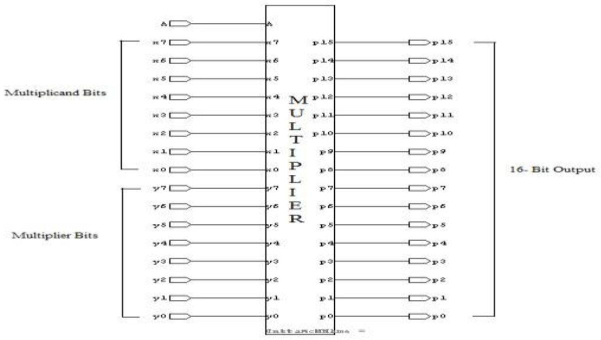


Fig.6 Symbolic Representation of Radix-4 Booth Multiplier

VI. EXPERIMENTAL RESULT

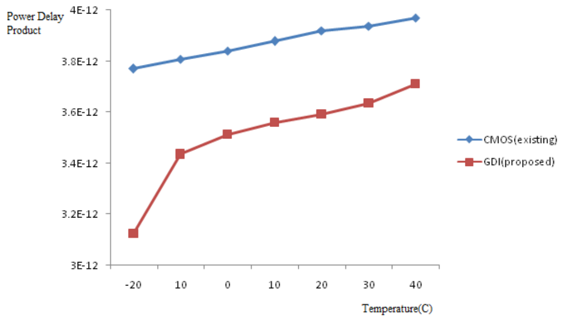
Table II shows implementation results including the proposed multiplier hardware area and power delay product and Table III & Table IV shows the number of transistors required to design the multiplier respectively as compared to previous work. The multiplication of power and delay means energy [1]-[3]-[5]. It has been observed that Power delay product is lower in proposed GDI multiplier as compared to the conventional CMOS multiplier. Hence, the proposed GDI multiplier has good performance than conventional CMOS multiplier [2]-[8].

Table II shows comparison between CMOS & GDI multiplier at different temperatures. In addition, the plot for power delay product and transistor count shows the comparison for CMOS multiplier and GDI multiplier.

TABLE II PDP for CMOS Multiplier & GDI Multiplier

Type of Gate	No. of transistor	Number needed	Transistors Est.
XOR	10	49	490
XNOR	8	128	1024
OR	6	60	360
NAND	4	81	324
3 Input OR	8	36	288
AND	6	48	288
NOT	2	4	8

PDP v/s Temperature



Transistor Count

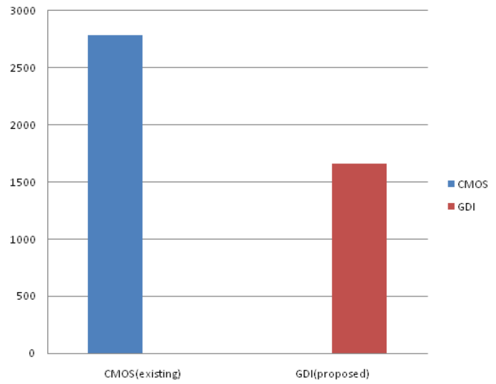


TABLE III Transistors Required For CMOS Multiplier

Type of Gate	No. of transistor	Number needed	Transistors Est.
XOR	4	49	196
XNOR	6	128	768
OR	2	60	120
NAND	4	81	324
3 Input OR	4	36	144
AND	2	48	96
NOT	2	4	8

TABLE III Transistors Required For GDI Multiplier

Temperature(C)	CMOS Multiplier	GDI Multiplier
-20	3.7692×10^{-12}	3.1229×10^{-12}
-10	3.8085×10^{-12}	3.4354×10^{-12}
0	3.8408×10^{-12}	3.5134×10^{-12}
10	3.8780×10^{-12}	3.5592×10^{-12}
20	3.9191×10^{-12}	3.5917×10^{-12}
30	3.9381×10^{-12}	3.6348×10^{-12}
40	3.9702×10^{-12}	3.7109×10^{-12}

VII. CONCLUSION

Simulation of this circuit is being done on Tanner EDA 13.0v Tool at 90nm technology and find out the following results. A GDI multiplier architecture which is proposed in this paper is an appropriate solution for multiplier because it has both low-power and high-speed and less area. After comparison, we have found out that the modified radix-4 booth multiplier has 7%-15% less power delay product with the use of only 1656 transistors in comparison to CMOS multiplier, which uses 2782 transistors. Hence, performance and area of modified multiplier is better than the existing one.

REFERENCES

1. B.S. Chekauer and E.G. Friedman, “A Hybrid Radix-4/Radix-8 Low 75Power Signed Multiplier Architecture”, IEEE Trans. Circuits and Systems-II: Analog and Digital Signal Processing, vol. 44, no.8, pp656-659, Aug. 1997.
2. Gyeonghoon Kim, Seungjin Lee, Junyoung Park, and Hoi-Jun Yoo, “A Low-energy Hybrid Radix-4/-8 Multiplier for Portable Multimedia Applications”, 978-1-4244-9472-9/11/$26.00 ©2011 IEEE.
3. A.P. Chandrakasan, S.Sheng, and R.W. Broderson, “Low-power CMOS digital design”, IEEE J. Solid-State Circuits, vol.27, pp. 473- 483, Apr. 1992.
4. G. N. Sung, Y. J. Ciou, and C. C. Wang, “A power-aware 2- dimensional bypassing multiplier using cell-based design flow,” IEEE International Symposium on Circuits and Systems, May 2008
5. H. B. Bakoglu, Circuits, Interconnections, and Packaging for VLSI. Reading, MA: Addison-Wesley, 1990.
6. A. Morgenshtein, A. Fish, and I. A. Wagner, “Gate-diffusion input (GDI) – A novel power efficient method for digital circuits: A detailed methodology,” in Proc. 14th IEEE Int. ASIC/SOC Conf., Sept. 2001, pp. 39–43.
7. “Gate-diffusion input (GDI) – A power efficient method for digital combinatorial circuits,” IEEE Trans. VLSI Syst., vol. 10, pp. 566–581, Oct. 2002.
8. Ankita Dhankar, Satyajit Anand, “Radix -4/-8 Dual Encoder Block for Multiplier Architecture using GDI Technique,” ISSN: 2278 – 1323 International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 1, Issue 9, November 2012.
9. Shiann-Rong Kuang, and Jiun-Ping Wang, “Modified Booth Multipliers With a Regular Partial Product Array”, IEEE Trans. vol.
10. 56, no.5, pp404-408, May. 2009.

AUTHORS PROFILE

Ankita Dhankar has received her B.Tech. degree in Electrical Engineering at Engineering College Bikaner, India. She is pursuing her M.Tech. degree in VLSI Design from Mody Institute of Technology and Science, Lakshmangarh, India. Till now her three papers have been published in International Journals. Field of interest is VLSI, Digital Design.

Satyajit Anand has received his B.Tech. degree in Electronics & Communication Engineering from Anna University, Chennai. He pursued his M.Tech degree in VLSI Design from Mody Institute of Technology and Science, Lakshmangarh, India. He is currently working as Assistant Professor at Mody Institute of Technology & Science, Lakshmangarh, India.