International Journal of Emerging Engineering Research and Technology
Volume 3, Issue 8, August 2015, PP 90-95
ISSN 2349-4395 (Print) & ISSN 2349-4409 (Online)

A Modified Radix2, Radix4 Algorithms and Modified Adder for Parallel Multiplication

Chakka Supriya[1], G.Vijaya Krishna[2]

[1]*Department of ECE, Audisankara Institute of Technology, Gudur, India (PG Scholar)*
[2]*Department of ECE, Audisankara Institute of Technology, Gudur, India (Associate Professor)*

ABSTRACT

A novel architecture of parallel multiplier using modified Booth's recoding unit for 2's complement numbers is presented in this paper. The basic Booth's recoding algorithm requires add and shift operations for multiplication, these steps make this multiplier sequential. Parallel multiplication can be achieved using Booth's recoding algorithm and simple Brown's array of adders, but it requires more number of adders to get correct output. Other parallel multiplication techniques are available using Booth's recoding algorithm. However, these array multiplier also requires add, shift and extra control unit. The proposed design has two major features; first is modified Booth's recoding unit which produces partial products second is modified array of adders. Modified array of adder block designed, which uses less number of adders than conventional Booth's recoding multiplier. Multiplexers are basic unit used for Booth's recoding unit we are designed modified Booth's recoding unit radix2 and radix4 for 4 bit, 8bit architectures .the proposed design has been simulated and synthesized with Xilinx 13.2 tool

Keywords: Booth's recoding unit, parallel multiplier, radix2, radix4.

INTRODUCTION

Multipliers are fundamental blocks in today's Digital signal processing units. Several multiplication algorithms are used to achieve multiplication, some of them are sequential multiplication and others are parallel multiplication. Parallel multiplication is always advantageous over sequential multiplications. Typically, Brown's array multiplier can do binary multiplication. However, it is not suitable for 2's complements numbers. So, to achieve parallel multiplication, Baugh-Woolly's 2's complement multiplication algorithm is widely used [1, 2]. Booth's algorithm [3, 4] is also used frequently for 2's complement numbers. Booth's algorithm uses radix recoding to achieve high speed. Increase in radix produces reduced number of partial products. Higher radix recoding has significant use in high-speed digital arithmetic. The basic multipliers based on booth algorithm are sequential multipliers. However, Booth's array multiplication is another method to achieve parallel multiplication. These types of array multipliers consist of a basic unit of add, subtract and shift with controlled input. It also requires small controller to control all the operation in the multiplier depending on input vectors. Lots of work is being done on parallel multipliers using Booth's recoding algorithm in this paper, a multiplexer based modified Booth's recoding circuit has been designed to generate efficient partial products. These partial products always have larger number of bits than the input number of bits. This width of partial product usually depends on radix scheme used for recoding. These generated partial products are added using novel array of adders to achieve parallel multiplier output. This scheme uses less number of gates so this circuit consumes less power and area.

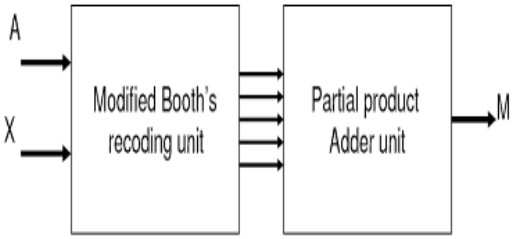
Proposed Novel Parallel Multiplier Using Modified Booth's Recoding Unit

Parallel multiplication using basic Booth's recoding algorithm is discussed in section 2. Since this technique requires lot of adders as a result it requires more power & area. In proposed multiplier design, we have reduced number of adders required in partial product addition. Mainly correction bits are reduced. This is done without compromising correctness of multiplication of 2's complement numbers. Also, we have used multiplexer based Booth's recoding scheme. The output recoding unit

Address for correspondence:

chpriya241@gmail.com

has been changed. This change results in partial products which after recoding are always greater than input bit length b y one bit in radix 2 scheme. Also in radix 4 schemes, it is always greater by two bits. These extra added bits work as correction bits to get proper output of multiplier. Also, at hardware realization of Booth's recoding scheme, we can remove extra select line, which is used at the time of recoding. Because of this extra select lines multiplexer size become large. We have observed that if we do not consider this extra bit at the time of hardware realization we can reduces size of one multiplexer .So, in radix 2 LSB decides first partial product. Also, in radix4 two LSB bits decides first partial product. The working of this novel design has been explained in following sections.



Modified Booth's Recoding Unit Radix 2 Method

Here, we have this recoding unit using multiplexers. Select lines to multiplexer are input bit sequence of multiplier and outputs are according to modified table given in table. So, in this scheme, partial products are always one bit more than input vector. If our multiplier is of n bit then partial products are always n+1.

MODIFIED BOOTH'S RECODING TABLE FOR RADIX 2

X_i	X_{i-1}	Y	Partial Product Explanation
0	0	0	All 0's
0	1	1.A	$[A_{(n-1)}, A]$
1	0	-1.A	$\overline{[A_{(n-1)}, (-A)]}$
1	1	0	All 0's

This can be explained with simple example.

A= 1100 (-4)

X= 1010 (-6)

So, partial products obtained for these inputs using recoding scheme are shown in table.

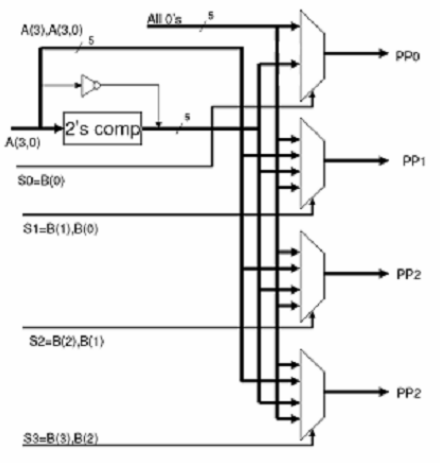
PP0= 00000

PP1= 00100

PP2= 11100

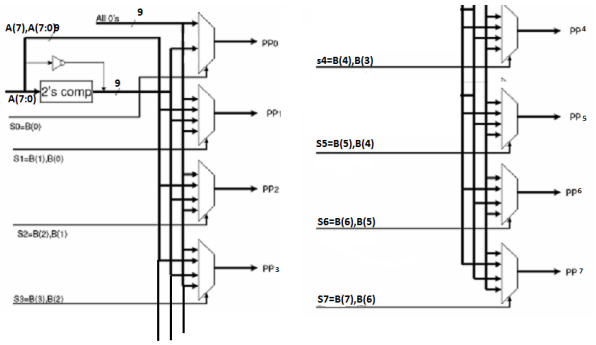
PP3= 00100

So, the hardware realization for this recoding unit is based on multiplexers and includes 2's complement unit. At the time of recoding, we assume extra '0' before LSB of multiplier, and this LSB & extra '0' bit decides Partial product according to recoding table 1. However, we have observed that at time of hardware realization only LSB is sufficient for getting partial product, because of this multiplexer becomes 2x1 rather than 4x1 and others multiplexers will remain same as per their input select lines depending upon recoding scheme. So, multiplexers become important hardware for Booth's recoding unit. Architecture of modified Booth's recoding unit



In this architecture, select lines for multiplexers are multiplier input bits taken according to recoding scheme given in table. Which are similar to General Booth's recoding scheme. We have used 4 multiplexer out of which 3 are 4x1 and one is 2x1.Each input vector is of n+1 of input bits of multiplier. 0thand3rdinput vector are always zero and 1stposition input vector is multiplier input (A) appended with extra zero at MSB to increase its width to n+1, it does not change original value. 2nd position input vector is taken 2's complement of input (A) with appending inverted MSB bit of input vector. This architecture generates four partial products vector according to table vector. This architecture generates four partial products vector according to table

Proposed 8 Bit Multiplier Using Radix2



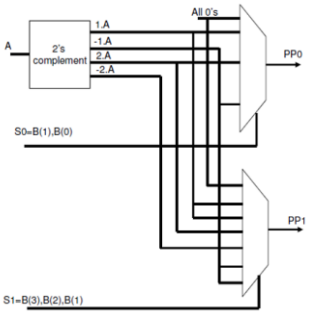
Modified Booth's Recoding Unit Radix 4 Method

This is also same scheme as explained above that reduces partial products so it is very helpful for fast multiplication of long input bit sequences. But, here partial product which we got from recoding unit is always 2 bit more than input bits. If our inputs are of n bit then partial products are n+2 bit. Recoding scheme is shown in table, and architecture of this recoding unit is shown in fig. In this radix scheme, select lines of multiplexers are 3 bits but first multiplexer can be of 2 select lines, which are two LSBs and remaining multiplexers have 3 input select lines

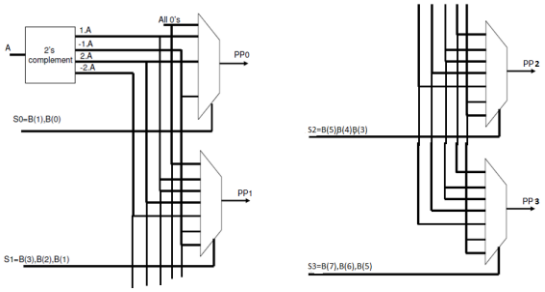
MODIFIED BOOTH'S RECODING UNIT FOR RADIX 4

X_{i+1}	X_i	X_{i-1}	Y	Partial Product Explanation
0	0	0	0	All 0's
0	0	1	1.A	$[A_{(n)}, A_{(n)}, A]$
0	1	0	1.A	$[A_{(n)}, A_{(n)}, A]$
0	1	1	2.A	$[A_{(n)}, A, 0]$
1	0	0	-2.A	$[\overline{A_{(n-1)}}, -A, 0]$
1	0	1	-1.A	$[\overline{A_{(n-1)}}, \overline{A_{(n-1)}}, -A]$
1	1	0	-1.A	$[\overline{A_{(n-1)}}, \overline{A_{(n-1)}}, -A]$
1	1	1	0	All 0's

The architecture of Booth's radix 4 recoding scheme is same as explained in section A shown in fig. Only difference is partial products are n+2 in width of input vector according to table. Now, all these partial products need to be added properly to get correct output. So, we designed modified partial product adder unit (array of adders) which moreover similar to Brown's array



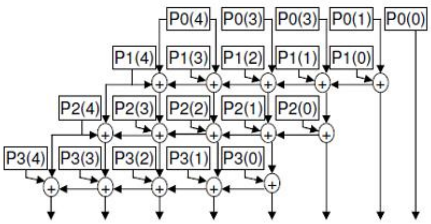
Proposed 8 Bit Multiplier Using Radix4



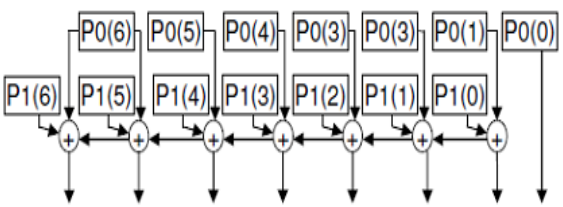
Modiefd Partial Product Adder

The partial product obtained from Booth's recoding unit needs to be added properly to get correct output of a multiplication. The addition scheme of partial product is same as Browns array multiplier except for MSBs. MSBs of partial products need to be added carefully. For that, new structure of an adder array is proposed. This modification removes the problem of large number of correction bits, which in turn require more number of adders. The proposed partial product adder arrays for 4 bit input sequence using radix 2 and radix 4 algorithms

Partial product adder unit for radix 2 recoding of 4 bit input



Partial Product Adder Unit for Radix 4 Recoding Of 4 Bit Input

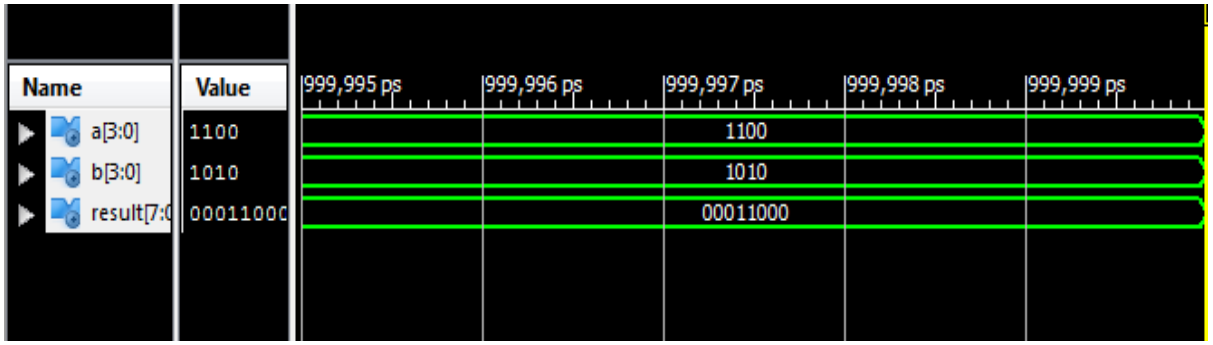


If n bit input binary input sequence is given then partial product will be Pij and multiplication output M will be of length r bit. So i =n+2 and j= (n/2).

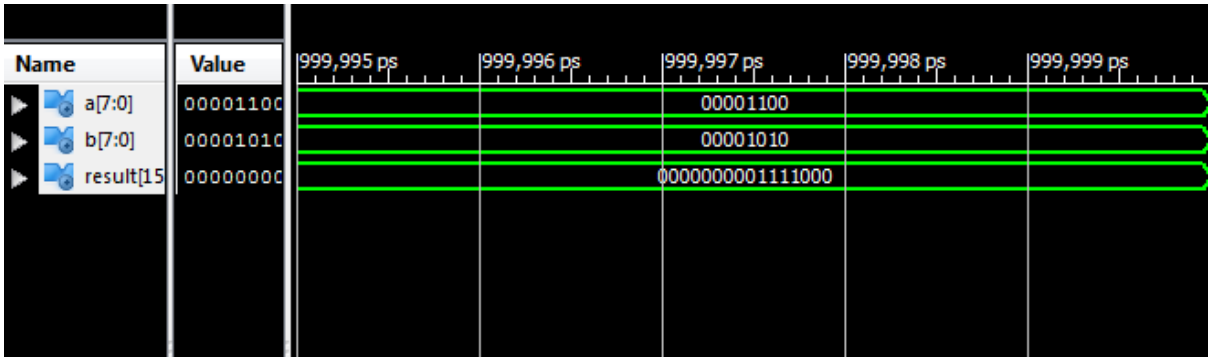
M0=> $P(0,0)$
M1=> $P(0,1)$
$P(0,2)+P(1,0)$
$P(0,3)+P(1,1)$
$P(0,4)+P(1,2)+P(2,0)$
$P(0,5)+P(1,3)+P(2,1)$
: ..... + ... + ... + ...
: ..... + ... + ... + ...
: ..... + ... + ... + ... + $P(i-1,0)$
: ..... + ... + ... + ... + $P(i-2,1)$
$P(0,j-2)+P(1,j-4)+P(2,j-6)+.....+P(i-1,2)$
$P(0,j-1)+P(1,j-3)+P(2,j-5)+.....+P(i-1,3)$
..... + ... + ... + ... + ...
..... + ... + ... + ... + ...
..... + $P(1,j-1)$ + ... + ... + ...
..... + ... + ... + ... + ...
..... + ... + $P(2,j-1)$ + ... + ...
..... + ... + ... + ... + ...
..... + ... + ... + ... + ...
M(r-1)=> $P(0,j-1)+P(1,j-1)+P(2,j-1)+.....+P(i-1,j-1)$

SIMULATION WAVEFORMS

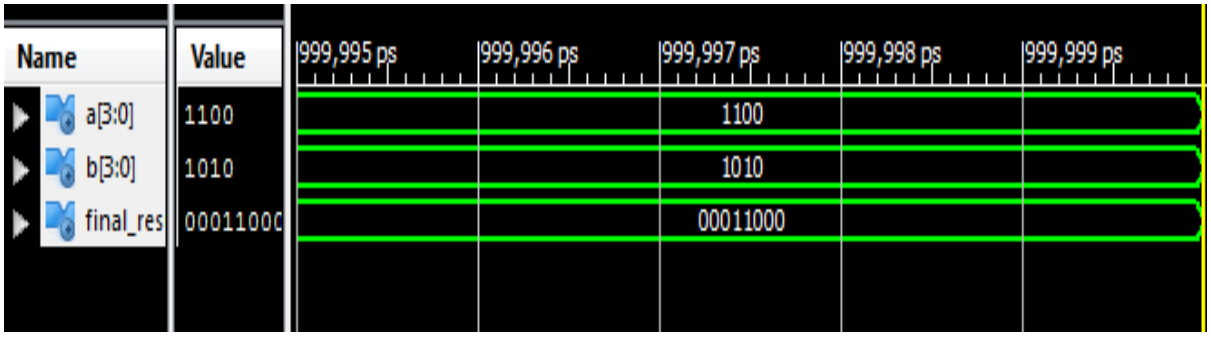
Radix2_4bit



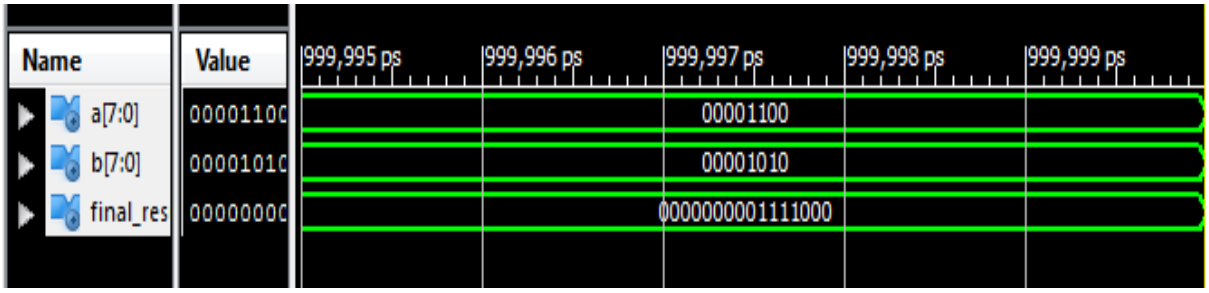
Radix2_8bit



Radix4_4bit



Radix4_8bit



CONCLUSION

We present novel design of Parallel multiplier using modified Booth's recoding unit. This multiplier based on multiplexers and novel partial product adder array. This multiplier is suitable for all signed & unsigned input vectors. We are designed modified Booth's recoding unit radix2 and radix4 for 4 bit, 8bit architectures .The proposed design has been simulated and synthesized with Xilinx 13.2 tool.

REFERENCES

[1] D. A Pucknell, K. Eshraghain, Basic VLSI Design, Prentice-Hall, ISBN 81-203-0986-3.

[2] C.R.Baugh ,B.A.Wooly, A two's complement parallel multiplication algorithm IEEE Transaction on computers, vol, c-22,no.12,December 1973.97

[3] Israel Koren, Computer arithmatics algorithms A.K.Peters Ltd. ISBN 1568811608.

[4] A.D.Booth, A signed binary multiplication technique, Quarterly Journal of Mechanics and Applied mathematics, vol-IV,pt-2-1951.

[5] Rajendra katti, A Modified Booth Algorithm for High Radix Fixed-point Multiplication, IEEE Transaction on very large scale integration VLSI systems,vol-2,no.-4 December- 1994.

[6] Wen-Chang Yeh, Chein-Wei Jen, High speed booth encoded parallel multiplier design, IEEE transaction on computers, vol-49, no-7.July-2000.

[7] Eduardo Costa, Sergio Bampi, Jos´e Monteiro, A New rchitecture for 2's Complement Gray Encoded Array Multiplier, Proceedings of the 15 th Symposium on Integrated Circuits and Systems Design (SBCCI'02).

[8] A.R. Cooper, Parallel architecture modified Booth multiplier, IEEE proceedings.vol-135.pt-G.No.3June 1988.

[9] Sanjiv Kumar Mangal, Rahul M. Badghare, Raghavendra B. Deshmukh, R. M. Patrikar, FPGA Implementation of Low Power Parallel Multiplier, 20th International Conference on VLSI Design (VLSID'07).

AUTHORS' BIOGRAPHY

Supriya Chakka has completed her B.Tech in Electronics and Communication Engineering from S.K.R College of Engineering & Technology, J.N.T.U.A affiliated college in 2013. She is pursuing her M.Tech in VLSI from audisankara Institute of Technology, J.N.T.U.A affiliated college.

Vijaya Krishna. G is an Associate Professor at Audisankara Institute of Technology, Gudur. He received his Bachelor degree in Electronics and Communication Engineering from Narayana College of Engineering & Technology, Gudur, in 2008 and Master of Technology in VLSI from Audisakara College of Engineering & Technology in 2013.