International Journal of Engineering Applied Sciences and Technology, 2020
Vol. 4, Issue 12, ISSN No. 2455-2143, Pages 459-463
Published Online April 2020 in IJEAST (http://www.ijeast.com)

SIGNED MULTIPLICATION USING QUAD PAIR RECODING

R. Sivarajan
B.E., M.E., Assistant Professor, Adhiparasakthi Engineering College,
Melmaruvathur, Tamilnadu, India – 603319

Abstract – **The main objective of this paper is to reduce the number of partial products obtained during the multiplication. Reducing the number of partial product leads to complete the multiplication process faster. In other words, the elapsed time to complete the multiplication process is reduced. This can be achieved by using Quad pair recoding technique in which multiplier (one of the input of the multiplication process) is coded by using quad pair recoding. Also this method is compared with the existing two methods using recoding and bit pair recoding technique.**

Keywords: Recoding, Bit Pair Recoding, Quad Pair Recoding

I. INTRODUCTION

Generally, multiplication can take atleast two inputs namely multiplicand and multiplier. Each bit in the multiplier is multiplied with the multiplicand and forms a partial product. The partial product is defined as the intermediate product which produced during the multiplication process. Finally all the partial products are summed together to form a final product. In an article written by Andrew. D. Booth (1951), Booth's multiplication is proposed in which multiplication of multiplicand with the bit pair 00 or 11 of the multiplier leads to all 0's as the partial product, 01 leads to multiplicand as the partial product and 10 leads to two's complement of multiplicand as the partial product.

In an article written by Macsorley. O. L (1961), the high speed arithmetic is proposed in which multiplication can be done by variable length shift. This paper the multiplicand or negative multiplicand can be shifted in variable length. This because we have various operation occurred during multiplication. This paper has done the pairing of three bits of multiplier and forms eight operation depending on code. The operations are +0, +2, +2, +4, −4, −2, −2 and 0 for 000, 001, 010, 011, 100, 101, 110, and 111 respectively. Here, +0 represent the zero as a partial product, +2 represent multiplicand without shift, +4 represent multiplicand with one shift, −2 represent negative multiplicand without shift and −4 negative multiplicand with one shift. Similarly this paper has done the pairing of four bits of multiplier and forms sixteen operation depending on code.

In an article written by Louis P. Rubinfield el al (1975), modified Booth algorithm using bit pair recoding technique is explained. In this, recoded multiplier is formed by grouping three bits and given the code for the corresponding group. In an article written by Carl Hamacher et al (2012), the two methods of signed multiplication is explained namely recoding and bit pair recoding. In this, it is clear that the multiplication done using bit pair recoding has less number of partial product compared to recoding method.

In an article written by Barun Biswas et al (2013), the classical Booth Multiplication method is proposed in which number of partial products required are 4 and 6 for 12 bit and 16 bit multiplication respectively. We know that the partial product formed during multiplication can be summed together to form a final product. Hence the adder circuit is necessary to complete the multiplication process. In an article written by Divya Govekar et al (2017), modified Booth's algorithm using bit pair recoding with two different adders namely CLA and hybrid adders was implemented. In this hybrid adder requires very less number of components and less delay compared to CLA. In an article written by Balakumaran R et al (2016), modified Booth algorithm using bit pair recoding is implemented with CLA and hybrid CLA. In this hybrid CLA takes less delay compared to CLA.

In an article written by Chiou-Yng Lee et al (2005), Low-complexity bit-parallel dual basis multipliers using the modified Booth's algorithm was proposed. This paper achieves 9% space complexity compared to other multipliers. In an article written by Nirlakalla Ravi et al (2012), a new reduced multiplication Structure is proposed using modified Booth Encoding Multiplier for Low power and Low area. This paper modifies the adder used in the multiplier. Normally, Ripple Carry adder is used in the multiplier structure, but here Carry Save adder is proposed. As a result, average power in milliwatt is reduced from 2.99 to 2.84.

In an article written by Vignesh Kumar. R et al (2012), high accuracy fixed width multiplier is designed using radix 4 modified Booth Algorithm. This paper result in significant improvement in area and power when compared to conventional multipliers.

II. EXISTING METHODS

From literature survey, it is clear that two methods are available to compute multiplication process between two numbers namely recoding and bit pair recoding method. This section deals with how the multiplication can be done using these two methods.

2.1 Recoding Method

In recoding method of multiplication, multiplier (second input) is recoded in which first appending of one zero at the $LSB-1$ position of the multiplier is to be done and then pair the bits from the left with overlapping. That is first pair is formed by taking LSB and $LSB-1$ bits, second pair is formed by taking $LSB+1$ and LSB bits, third pair is formed by taking $LSB+2$ and $LSB+1$ bits. After pairing the bit, each pair is assigned a code as shown in table 1.

Table 1 Code using Recoding Method

Bits	Code
00	0
01	+1
10	−1
11	+1

For example, let us assume to perform the multiplication between 60 and −15. Here Multiplicand is $60_{10}=(00111100)_2$ and Multiplier is $(-15)_{10}=(11110001)_2$.

Recoding of Multiplier is as follow:

$$\begin{array}{ccccccccc}
7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 & \mathbf{-1} \\
B_7 & B_6 & B_5 & B_4 & B_3 & B_2 & B_1 & B_0 & \boldsymbol{B_{-1}} \\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & \mathbf{0} \\
11 & 11 & 11 & 10 & 00 & 00 & 01 & 10 & \\
0 & 0 & 0 & -1 & 0 & 0 & +1 & -1 &
\end{array}$$

Now perform multiplication of multiplicand and recoded code of multiplier as follow:

$$\begin{array}{cccccccccccccccc}
 & & & & & & & & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\
 & & & & & & & & 0 & 0 & 0 & -1 & 0 & 0 & +1 & -1 \\
\hline
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & & \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & & & \\
1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & & & & \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & & & & & \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & & & & & & \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & & & & & & & \\
\hline
1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0
\end{array}$$

Since MSB bit of the result is 1, the result is the negative value and the decimal value is obtained by taking two's complement of the result and put negative sign in front of it. Hence the result is $(-0000001110000011+1)_2=(-0000001110000100)_2=(-900)_{10}$. Here we seen that the number of partial product required is 8 for 8- bit multiplication process.

2.2 Bit Pair Recoding Method

In bit pair recoding method, we first form the code for the multiplier using recoding method. After that, pair the code, resulting from the recoding, without overlapping from the left. That is $LSB+1$ and LSB code of recoded multiplier forms a first pair, $LSB+3$ and $LSB+2$ code forms a second pair and so on. Finally each pair is assigned with a code as shown in table 2.

Table 2 Code for Recoded Code Pair

Recoded Code Pair	Bit Pair Recoded Code
0, 0	0
0, +1	+1
0, −1	−1
+1, 0	+2
+1, +1	Not Possible Pairs
+1, −1	−1
−1, 0	−2
−1, +1	−1
−1, −1	Not Possible Pairs

Bit pair recoding can also be obtained directly from the multiplier without forming a recoded code. For that, we first appending of one zero at the $LSB-1$ position of the multiplier is to be done and then group the bits in three from the left with overlapping. That is first group is formed by taking $LSB+1$ to $LSB-1$ bits, second group is formed by taking $LSB+3$ and $LSB+1$ bits, third group is formed by taking $LSB+5$ and $LSB+3$ bits. After grouping the bit, each pair is assigned a code as shown in table 3.

Table 3 Code using Bit Pair Recoding Method

Bits	Code	Bits	Code
000	0	100	−2
001	+1	101	−1
010	+1	110	−1
011	+2	111	0

For example, let us assume to perform the multiplication between 60 and −15. Here Multiplicand is $60_{10}=(00111100)_2$ and Multiplier is $(-15)_{10}=(11110001)_2$.

Bit pair Recoding of Multiplier is as follow:

$$\begin{array}{ccccccccc}
7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 & \mathbf{-1} \\
B_7 & B_6 & B_5 & B_4 & B_3 & B_2 & B_1 & B_0 & \boldsymbol{B_{-1}} \\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & \mathbf{0} \\
 & 111 & & 110 & & 000 & & 010 &
\end{array}$$

0	-1	0	$+1$

Now perform multiplication of multiplicand and recoded code of multiplier as follow:

								0	0	1	1	1	1	0	0
									0		−1		0		+1
0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0		
1	1	1	1	1	1	0	0	0	1	0	0				
0	0	0	0	0	0	0	0	0	0						
1	1	1	1	1	1	0	0	0	1	1	1	1	1	0	0

Since MSB bit of the result is 1, the result is the negative value and the decimal value is obtained by taking two's complement of the result and put negative sign in front of it. Hence the result is $(-0000001110000011+1)_2 = (-0000001110000100)_2 = (-900)_{10}$. Here we seen that the number of partial product required is only 4 for 8- bit multiplication process, which is nothing but only half of the partial product required when compared to recoding method.

III. PROPOSED METHOD

This section deals with how multiplication can be done using the proposed quad pair recoding method with a suitable example.

3.1 Quad Pair Recoding Method

Quad pair recoding can be obtained directly from the multiplier without forming a recoded code or bit pair recoded code. For that, we first appending of one zero at the $LSB-1$ position of the multiplier is to be done and then group the bits in five from the left with overlapping. That is first group is formed by taking $LSB+3$ to $LSB-1$ bits, second group is formed by taking $LSB+7$ and $LSB+3$ bits, third group is formed by taking $LSB+11$ and $LSB+7$ bits. After grouping the bit, each pair is assigned a code as shown in table 4.

Table 4 Code using Quad Pair Recoding Method

Bits	Code	Bits	Code
00000	0	01000	+1
00001	+1	01001	+2
00010	+1	01010	+2
00011	+2	01011	+3
00100	+1	01100	+2
00101	+2	01101	+3
00110	+2	01110	+3
00111	+3	01111	+4
10000	−4	11000	−3
10001	−3	11001	−2
10010	−3	11010	−2
10011	−2	11011	−1
10100	−3	11100	−2
10101	−2	11101	−1
10110	−2	11110	−1
10111	−1	11111	0

For example, let us assume to perform the multiplication between 60 and -15. Here Multiplicand is $60_{10} = (00111100)_2$ and Multiplier is $(-15)_{10} = (11110001)_2$.

Quad pair Recoding of Multiplier is as follow:

7	6	5	4	3	2	1	0	**−1**
B_7	B_6	B_5	B_4	B_3	B_2	B_1	B_0	$\boldsymbol{B_{-1}}$
1	1	1	1	0	0	0	1	**0**
			11100				00010	
			−2				+1	

Now perform multiplication of multiplicand and recoded code of multiplier as follow:

								0	0	1	1	1	1	0	0
											−2				+1
0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0
1	1	1	1	1	1	0	0	0	1	0	0				
1	1	1	1	1	1	0	0	0	1	1	1	1	1	0	0

Since MSB bit of the result is 1, the result is the negative value and the decimal value is obtained by taking two's complement of the result and put negative sign in front of it. Hence the result is $(-0000001110000011+1)_2 = (-0000001110000100)_2 = (-900)_{10}$. Here we seen that the number of partial product required is only 2 for 8- bit multiplication process, which is nothing but only half of the partial product required when compared to bit pair recoding method and quarter of the partial product required when compared to recoding method.

IV. RESULTS AND DISCUSSION

The performance of this paper is in the number of partial product. Table 5 shows the number of partial product occur in both existing (Recoding and Bit pair Recoding method) and proposed method (Quad Pair Recoding method). From the table 5, it is clear that the number of partial product required in quad pair recoding is half of the partial product required for bit pair recoding and quarter of the partial product required for recoding method.

Figure 1 shows that the variation of number of bits versus number of partial product in all the methods such as recoding, bit pair recoding and quad pair recoding. From the figure 1 also, it is clear that the number of partial product required in quad pair recoding is half of the partial product required for bit pair recoding and quarter of the partial product required for recoding method.

Table 6 shows that the time required to complete the multiplication process. It is clear that time required to complete the multiplication process is less in quad pair recoding compared to both recoding and bit pair recoding method.

Table 5 Number of Partial Product in each method

Number of bits	**Number of Partial Products**		
	Existing Methods		**Proposed Method**
	Recoding Method	**Bit pair Recoding Method**	**Quad Pair Recoding Method**
8	8	4	2
12	12	6	3
16	16	8	4
20	20	10	5
24	24	12	6

Table 6 Elapsed Time in each method

Number of bits	**Number of Partial Products**		
	Existing Methods		**Proposed Method**
	Recoding Method	**Bit pair Recoding Method**	**Quad Pair Recoding Method**
8	0.358981	0.014963	0.023797
12	0.001238	0.001006	0.000979
16	0.001837	0.000813	0.000866
20	0.001654	0.001194	0.000976
24	0.001864	0.001534	0.000836

Figure 2 shows the variation of Elapsed time to complete the multiplication using recoding, bit pair recoding and quad pair recoding. From the figure 2, it is clear that the elapsed time is very less in quad pair recoding compared to bit pair recoding and recoding method when number of bits is larger than 12 bit.

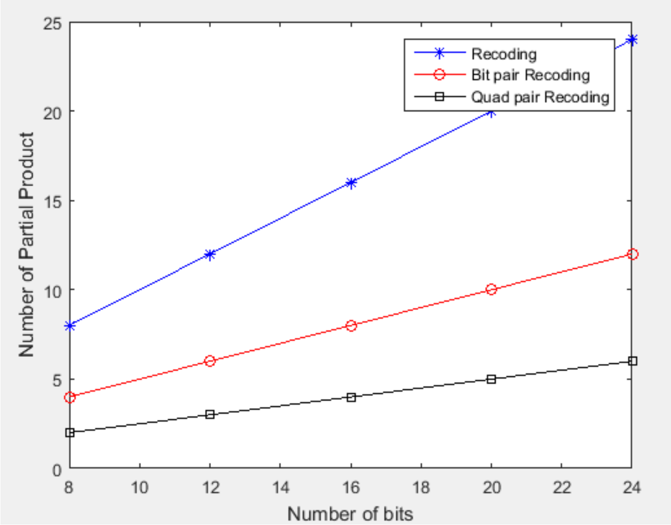

Figure 1 Variation of Number of Partial Products

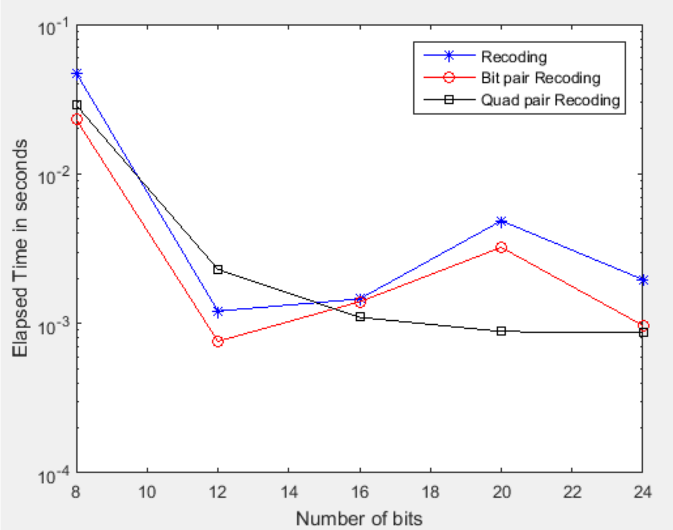

Figure 2 Elapsed Time to complete the multiplication using recoding, bit pair recoding and quad pair recoding

V. CONCLUSION

This paper achieves a main objective that to reduce the number of partial product. The objective of reducing partial product is that to complete the multiplication faster within a stipulated time. This paper compares the number of partial products required to complete the multiplication using quad pair recoding with recoding and bit pair recoding method and achieves better performance. This paper also compares the elapsed time to complete the multiplication process using quad pair recoding and achieves better performance. Hence, finally it is concluded that the proposed quad pair recoding method performs better in terms of both number of partial products and elapsed time.

VI. REFERENCE

[1] Andrew. D. Booth (1951), "A Signed Binary Multiplication Technique", in the Quarterly Journal of Mechanics and Applied Mathematics, Vol. 4, Issue. 2, (pp. 236–240).

[2] Macsorley. O. L (1961), "High-Speed Arithmetic in Binary Computers", in Proceedings of the IRE, DOI: 10.1109/JRPROC.1961.287779, (pp. 67-91).

[3] Louis P. Rubinfield (1975), "A Proof of the Modified Booth's Algorithm for Multiplication", in IEEE Transactions on Computers, (pp. 1014-1015).

[4] Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Naraig Manjikian (2012), "Computer Organization and Embedded Systems", in Mc Graw Hill, ISBN 978–0–07–338065–0, MHID 0–07–338065–2.

[5] Barun Biswas and Bidyut B Chaudhuri (2013), "Generalization of Booth's Algorithm for Efficient Multiplication" in International Conference on Computational Intelligence: Modeling, Techniques and Applications, (pp. 304-310).

[6] Balakumaran R, Prabhu E (2016), "Design of high speed multiplier using Modified Booth Algorithm with hybrid carry look-ahead adder", International Conference on Circuit, Power and Computing Technologies [ICCPCT].

[7] Divya Govekar, Dr Ameeta Amonkar (2017), "Design and Implementation of High Speed Modified Booth Multiplier using Hybrid Adder", in Proceedings of the IEEE 2017 International Conference on Computing Methodologies and Communication (ICCMC), (pp. 138-143).

[8] Chiou-Yng Lee, Che Wun Chiou, Jim-Min Lin (2005), "Low-complexity bit-parallel dual basis multipliers using the modified Booth's algorithm", in ELSEVIER Computers and Electrical Engineering, doi: 10.1016/j.compeleceng.2005.09.002, (pp.444-459).

[9] Nirlakalla Ravi, SubbaRao. T, Bhaskara Rao. B, Jayachandra Prasad. T (2012), "A New Reduced Mu;triplication Structure for Low power and Low area modified Booth Encoding Multiplier" in International Conference on Modeling Optimization and Computing, doi: 10.1016/j.proeng.2012.06.324, (pp. 2767 – 2771).

[10] Vignesh Kumar. R and Kamala. J (2012), "High Accuracy Fixed Width Multipliers using Modified Booth Algorithm, in International Conference on Modeling Optimization and Computing, doi: 10.1016/j.proeng.2012.06.294, (pp. 2491 – 2498)

[11] Morris Mano. M, "Computer System Architecture", Pearson Education.

[12] Bhavya Lahari Gundapaneni, JRK Kumar Dabbakuti, (2019), "Booth Algorithm for the Design of Multiplier", in International Journal of Innovative Technology and Exploring Engineering (IJITEE), (pp. 1506-1509).

[13] Sakthivel. B, Maheshwari. K, Manojprabakar. J, Nandhini. S, Saravanapriya. A (2017), "Implementation of Booth Multiplier and Modified Booth Multiplier", in International Journal of Recent Trends in Engineering & Research (IJRTER), DOI : 10.23883/IJRTER.CONF.20170331.010.TIVW3, (pp. 45-51)