ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Implementation of New Modified Booth Recoder Architecture for Efficient Design of Add-Multiply Operator

A.Rama.V.S.Gupta
Department of ECE,
Swarnandhra College of Engineering & Technology,
Seetharampuram, Narsapur, A.P, India.

J.E.N.Abhilash
Department of ECE,
Swarnandhra College of Engineering & Technology,
Seetharampuram, Narsapur, A.P, India.

I.V.Ravi Kumar
Department of ECE,
Swarnandhra College of Engineering & Technology,
Seetharampuram, Narsapur, A.P, India.

Abstract:

Large numbers of multifaceted arithmetic operation are used in many digital signal processing (DSP) relevance. The multiplier reduces within power and area consumption and plays important role in high performance of any digital indication processing system. within this paper, mainly centre of attention on optimizing and increased performance by reduction in power consumption in propose of the fused Add-Multiply (FAM) operator. This implements a new technique by straight recoding of sum two numbers in Modified Booth (MB) form. In this paper implemented a new and efficient structured technique by straight recoding of sum of two numbers by considering existing modified booth (MB) technique. The new technique is implemented by three new dissimilar schemes by integrating them within existing FAM plans. The performance of the proposed three different schemes gives reduction in conditions of critical delay, hardware complication and power utilization while comparing with the existing AM design.

KEYWORDS: ***Digital Signal Processing (DSP), Modified Booth (MB), Add-Multiply (AM) operation***

INTRODUCTION

In several design Digital Signal Processing (DSP) appliance multiplier plays an significant role. The DSP arrangement uses in modern electronics and make extensive use of custom accelerators for multimedia, communication etc,. In the transverse filter, Fast Fourier transform (FFT), implementation of recursive and discrete Fourier transforms, the multiplier is used in their implementation. The performance of DSP arrangement is individually precious by propose concerning the structural design of arithmetic units.

In the ground of arithmetic optimization the recent research activities have shown that the invent of arithmetic mechanism combining operations which share data and which gives significant performance improvements. Based on the observation that an addition can often be subsequent to a multiplication. While introducing the multiply-accumulator (MAC) and multiply-add (MAD) units leading to high efficient implementations of DSP algorithms when compared to the conventional ones, which use only primitive resources. To optimize the performance of the MAC operations in reduction of area, critical delay and power consumption more number of architectures has been proposed. The most of the DSP applications are used Add-Multiply (AM) operations when compared with MAC/MAD operations. The MAC/MAD operation performs the multiplication on given inputs and then the result is given to the adder unit. Simply multiplication then addition is processed. In case of Add-Multiply (AM) unit, firstly the inputs are added and then the output of adder is pushing to the input of a multiplier. The AM unit increases the significant area and critical path delay and power consumption of the circuit when compared with the MAC/MAD unit. Fusion techniques are employed supported on the direct recoding of the sum of two numbers into its modified booth (MB) form to reduce the design of AM operators. Thus the carry propagate adder of the

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

conventional AM design is eliminated resulting in increases performance of the system. A new signed bit MB Recoder which transforms redundant binary inputs to their MB recoding form. A special expansion of the pre-processing step of the Recoder is needed in order to handle operands in carry save representation.

In this proposes a two stage Recoder which converts a number in carry put away form to its MB representation. The first step transforms the carry save form of the input number into signed number form which is after that recoded inside the second phase so that it matches the form that the MB digits request. This technique has been used for the design of high performance flexible coprocessor architectures targeting the computationally intensive DSP applications. In the recoding of a redundant input from its carry keep form to the corresponding borrow set aside form keeping the critical pathway of multiplication operation fixed. When compared to the conventional AM, the direct recoding of the summation of two numbers during its MB form leads to more resourceful performance of the Fused Add Multiply (FAM) component, existing recoding schemes are based on complex multiplications in bit level, which are implemented by dedicated circuits at gate level. In this paper, focuses on efficient design of FAM operator, targeting the optimization of the recoding method for direct determining of the MB form of the sum of two numbers. Specifically in this propose a new recoding technique which decreases critical path delay and reduces power consumption.

During conventional design the multiplicand is formed by adding the inputs A and B, the adder inserts significant delay and this leads to increases in area, power consumption and critical delay. In the proposed design the sum is directly recoded as the MB digit and improves in power consumption, area and delay in existing intend. In order to be apply either in signed (in 2's complement representation) or unsigned numbers, which comprise of odd or even number of bits the proposed NS-MB algorithm is structured, simple and can be easily modified. In this proposed NSMB approach the three alternative schemes are analyzed using conventional and signed bit half adders (HA's) and Full adders (FA's) as basic building blocks for that proposed NSMB algorithm.

The presentation of the planned NS-MB is appraised by three alternative schemes by comparing with state-of-the-art recoding techniques. The critical path delay, power estimation has been used to provide accurate measurements are estimated regarding various bit-widths of input numbers. Regarding various bit-widths of the input numbers, the industrial tools for RTL synthesis and power estimation have been used to provide accurate measurements of area utilization, critical path delay and power dissipation. For large range of frequencies, the adoption of the proposed recoding technique delivers optimized solution for FAM design enabling the targeted operation to be timing functional. Under the same timing constraints the proposed FAM design deliver improvements in both area occupation and power consumption.

The remaining of the paper is organised while in the coming sections we discuss about motivation and present technical background for the implementation of FAM design and the proposed NS-MB recoding scheme is presented and experimental evaluation are given and then clearly identifying the advantages of the proposed NS-MB schemes with respect to critical delay and power dissipation and last section concludes the paper.

MOTIVATION AND FUSED AM IMPLEMENTATION

MOTIVATION

This paper mainly focuses on AM units which realize the process Z= X. (A+B). The addition operation is performed on the inputs A and B by using adder and then input X and the adder output i.e., figure Y=A+B are determined to a multiplier within charge to acquire the result (Z) within the conventional design of AM operator (Fig 1(a)). In order to decrease the delay

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

which is occurred due to carry signals propagated inside the adder a Carry-Look-Ahead (CLA) adder can be worn which on the other hand add to the area profession and power indulgence. The union of the adder and MB programming unit into a solo data path block represents an optimized design of AM operator. In the Fused AM operator the direct recoding of the sum Y=A+B to its MB representation is taken. The fused Add-Multiply (FAM) parts have just one adder at the end. As a effect of using one adder at the end major area investments and significant path delay of the recoding process is concentrated. In this paper presented a innovative technique for straight recoding of two numbers in the MB representation.

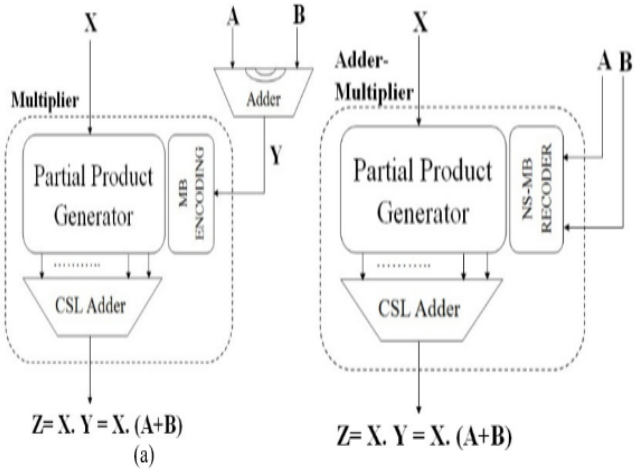


Fig 1. AM operator stand on the (a) existing plan and (b) Fused propose with direct recoding of the sum of A and B in its MB representation

Review of the Modified Booth Form:

Modified Booth (MB) is a extensive form used in multiplication. The MB encoding uses redundant signed digit radix-4 programming method. The essential benefit of this method is that it decreases the amount of partial products by half in multiplication process comparing to any other radix-2 representation.

Let us judge the multiplication of 2's complement statistics X and Y with every number consisting of n=2k spot. The multiplicand Y can be correspond to in MB form as

$$Y = \langle y_{n-1}y_{n-2}\cdots y_1y_0\rangle_{2's} = -y_{2k-1}\cdot 2^{2k-1} + \sum_{i=0}^{2k-2} y_i \cdot 2^i$$

$$= (y_{k-1}^{MB}y_{k-2}^{MB}\ldots\ldots y_1^{MB}y_0^{MB})_{MB} = \sum_{j=0}^{k-1} y_j^{MB}.2^{2j} \quad (1)$$

$$y_j^{MB} = -2\,y_{2j+1} + y_{2j} + y_{2j-1} \quad (2)$$

Digits $y_j^{MB} \epsilon \{-2,-1,0,+1,+2\}$, $0 \leq j \leq k-1$, communicate to the three successive bits $y_{2j+1}, y_{2j}, y_{2j-1}$ with one bit extend beyond and allowing for to $y_{-1} = 0$. the table I shows how the MB digits are formed by summarizing the MB encoding technique. Each figure is symbolize by three bits given name s, one and two. The sign spot (s) represents the number sign either negative (s=1) or optimistic (s=0). Signal one representing the complete value of a numeral is equal to 1 (one=1) or not (one=0). sign two representing the complete value of a number is equal to 2 (two=1) or not (two=0). By means of these three signals (s, one, two) the MB digit (y_j^{MB}) is formed and it's represented by following equation

$$y_j^{MB} = (-1)^{S_j}.[one_j + 2.two_j] \quad (3)$$

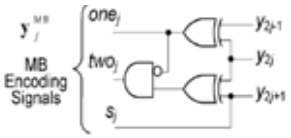
Fig 2(a) shows the Boolean equations on which the implementation of the MB encoding signals is based (Fig 2(b)).

TABLE I
MODIFIED BOOTH ENCODING TABLE

Binary			y_j^{MB}	MB Encoding			Input Carry
Y_{2j+1}	Y_{2j}	Y_{2j-1}		Sign=s_j	x 1=one_j	x 2=two_j	$C_{in,j}$
0	0	0	**0**	0	0	0	0
0	0	1	**+1**	0	1	0	0
0	1	0	**+1**	0	1	0	0
0	1	1	**+2**	0	0	1	0
1	0	0	**-2**	1	0	1	1
1	0	1	**-1**	1	1	0	1
1	1	0	**-1**	1	1	0	1
1	1	1	**0**	1	0	0	0

$$one_j = y_{2j-1} \oplus y_{2j}$$
$$two_j = (y_{2j+1} \oplus y_{2j}).\overline{one_j}$$
$$s_j = y_{2j+1}$$

(a)



(b)

Fig. 2. (a) Boolean equations and (b) gate-level schematic for the implementation of the MB encoding signals

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

FAM Implementation:

The proposed FAM design represented in fig 1 (b)
The multiplier is a parallel one based on the MB algorithm.
Let us consider X,Y, the term Y= { y_{n-1} y_{n-2}.....$y_1y_0\}_{2's}$ is prearranged stand on the MB algorithm and multiply with X= { x_{n-1} x_{n-2}.....$x_1x_0\}_{2's}$. Mutually X and Y consists of n=2k bits and in 2's complement form. Equation (4) explains the production of the k partial commodities.

$$PP_j = X.y_j^{MB} = \overline{p_{j,n}}\ 2^n + \sum_{i=0}^{n-1} p_{j,i}.2^i \quad (4)$$

The partial product PP_j is generated from *i*- th bit of $p_{j,i}$ and is stand on the subsequently logical appearance while fig (3) demonstrate its execution at gate intensity

$$p_{j,i} = \left((x_i \oplus s_j) \wedge one_j\right) \vee (((x_{i-1} \oplus s_j) \wedge two_j \quad (5)$$

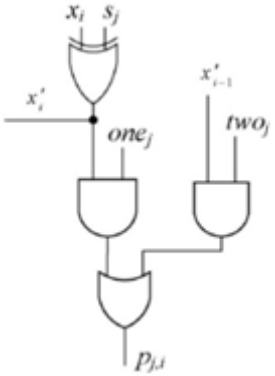


Fig 3. Creation of the i-th bit $P_{j,i}$ of the partial product PP_j meant for the conventional MB multiplier

We consider x_{-1} =0 and $x_n = x_{n-1}$ for the working out of the slightest and most considerable bits of partial product respectively. The quantity of ensuing prejudiced products are [n/2] +1=k+1 in case of n=2k+1. Based on sign conservatory of the preliminary 2's complement digit the most significant MB digit is formed.

After generation of partial products they are further properly weighted throughout a carry select adder (CSL) and which is prearranged by following equation

$$Z = X.Y = \sum_{j=0}^{k-1} PP_j.2^{2j} \quad (6)$$

The output of the carry select adder (CSL) gives the result Z=X.Y as shown in fig 1(b).

III. NEW SUM TO MODIFIED BOOTH RECODING TECHNIQUE (NS-MB):

Defining signed bit full adders and half adders for structured signed Arithmetic

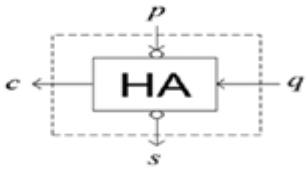
The recoding in this New sum to modified booth Recoder is recorded by considering the two consecutive bits of the input A (a_{2j} , a_{2j+1}) with two consecutive bits of the input B (b_{2j}, b_{2j+1}) into one MB digit y_j^{MB}. As from eq.(2) , the MB digit is formed by including the three bits. The most considerable of them is negatively slanted whereas two least considerable of them have positive weight. Use signed spot calculation in order to make over the two aforementioned couple of bits in MB appearance. In this paper presented a set of bit stage half adders (HA) and Full adders (FA) considering their inputs and outputs to be signed.
Specifically mention here is in this work developed two types of signed Half adders which are referred as HA^* and HA^{**}. Tables II – IV are their truth tables and their corresponding Boolean equations are represented in fig (4).

HA^*

$$c = p \vee q$$
$$s = p \oplus q$$

(a)

HA^{**}



$$c = \bar{p} \wedge q$$
$$s = p \oplus q$$

(b)

Fig.4. Structures for signed (a) HA^* (b) HA^{**}

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

The HA* which equipment the relative 2.c-s =p+q everywhere the sum s is considered negatively signal (Table II, Fig 4(a)), by considering that p,q are binary contribution and c,s are the outputs (the carry and sum correspondingly) of a HA*. The output gives one of the values { 0, +1,+2}. In table III described the dual implementation of HA* which is formed by inversing the signs of all inputs and outputs and consequently, changed the output values to {-2,-1,0}. The relation 2.c-s =-p+q shown in fig 4(b) & table IV is implemented by HA** and it shows the operation and schematic of HA**. The result manipulates a negative (p) and a positive (q) input resulting in the output values {-1, 0, +1}.

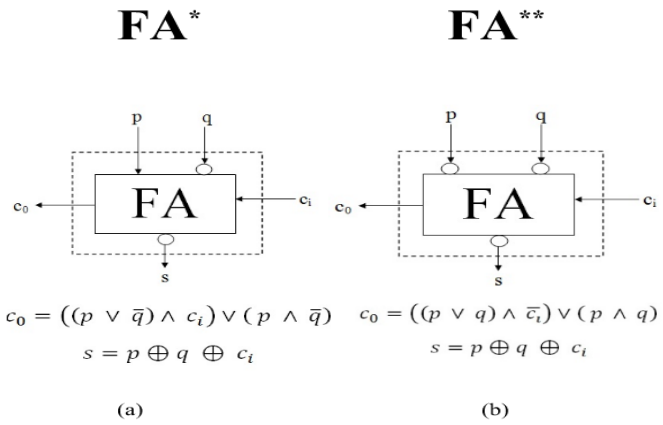


Fig.5. structures for signed (a) FA* and (b) FA**

Also within this paper represented two types of signed FAs (Full Adders) which are presented in table V and VI and fig 5. In the fig 5(a) and 5(b) represents the schematic and shows the relation of FA* and FA** with the conventional FA. The FA* apparatus the relative $2.c_0\text{-}s = p - q + c_i$ everywhere the fragment s and q be considered negatively indication (Table V, Fig 5(a)) by assuming p,q and c_i are the binary participation and c_0 , s be the output carry and sum correspondingly. Output values of FA* are {-1,0,+1,+2} and they are shown in table V (truth table of FA*). In case of FA** equipment the relative $c_0 + s = -p - q + c_i$ (Table VI Fig 5(b)) where p,q are negatively signs. The output values become {-2,-1,0,+1}. The Fig.5 shows the signed FAs implemented using conventional FA with the negative inputs and outputs inverted.

TABLE II
HA* BASIC OPERATION

INPUTS		OUTPUT VALUE	OUTPUTS	
p (+)	q (+)		c (+)	s (-)
0	0	0	0	0
0	1	+1	1	1
1	0	+1	1	1
1	1	+2	1	0

Output value = 2.c-s = p+q

TABLE III
HA* DUAL OPERATION

INPUTS		OUTPUT VALUE	OUTPUTS	
p (-)	q (-)		c (-)	s (+)
0	0	0	0	0
0	1	-1	1	1
1	0	-1	1	1
1	1	-2	1	0

Output value = -2.c+s = -p-q

TABLE IV
HA OPERATION**

INPUTS		OUTPUT VALUE	OUTPUTS	
p (-)	q (+)		c (+)	s (-)
0	0	0	0	0
0	1	+1	1	1
1	0	-1	0	1
1	1	0	0	0

Output value = 2.c-s =- p+q

TABLE V
FA* OPERATION

INPUTS			OUTPUT VALUE	OUTPUTS	
p (+)	q (-)	C_i (+)		C_0 (+)	s (-)
0	0	0	0	0	0
0	0	1	+1	1	1
0	1	0	-1	0	1
0	1	1	0	0	0
1	0	0	+1	1	1
1	0	1	+2	1	0
1	1	0	0	0	0
1	1	1	+1	1	1

Output value = $2.c_0\text{-}s = p\text{-}q+c_i$

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

TABLE VI
FA OPERATION**

INPUTS			OUTPUT VALUE	OUTPUTS	
p (-)	q (-)	C_i (+)		C_0 (-)	s (+)
0	0	0	0	0	0
0	0	1	+1	0	1
0	1	0	-1	1	1
0	1	1	0	0	0
1	0	0	-1	1	1
1	0	1	0	0	0
1	1	0	-2	1	0
1	1	1	-1	1	1

Output value = $-2.c_0+s = -p-q+c_i$

Proposed NS-MB Recoding Techniques:

In order to design and explore new three alternative schemes of the New Sum to Modified recoding (NS-MB) technique used both conventional and signed HAs and FAs of section III.A. The three new alternative techniques can be easily either in signed (2's complement representation) or unsigned numbers which consists of odd or even number of bits.

In all three schemes considered that both inputs A and B are consist of 2k bits in case of even and (2k+1) bits in case of odd bit-width. Consider the bits a_{2j}, a_{2j+1} and b_{2j}, b_{2j+1} as the inputs of the j-recoding cell to transform the sum of A and B in order to get at its output the three bits that need to form the MB digit y_j^{MB} according to equation (2).

NS-MB1 Recoding Scheme:

The first scheme in three alternative schemes is demote as the NS-MB1 and is demonstrate in factor in fig (6) for together even (fig 6(a)) and odd (fig 6(b)) bit size of input information. As seen in fig.6 the addition of A and B is specified by

$$Y = A + B = y_k.2^k + \sum_{j=0}^{k-1} y_j^{MB}.2^{2j}$$

$$\text{Where } y_j^{MB} = -2s_{2j+1} + s_{2j} + c_{2j,2} \qquad (7)$$

The programming of the MB digit y_j^{MB}, $0 \le j \le k-1$ of (7) is support on the investigation of section II.B. We consider the early ideals $c_{0,1} = 0$ along with $c_{0,2} = 0$. The bits s_{2j+1} and s_{2j} are pulling out starting the j recoding cell of fig.6. A conservative FA with inputs a_{2j}, b_{2j} and $c_{2j,1}$ produces the carry $c_{2j+1} = (a_{2j} \wedge b_{2j}) \vee (c_{2j,1} \wedge (a_{2j} \vee b_{2j}))$ and sum $s_{2j} = a_{2j} \oplus b_{2j} \oplus c_{2j,1}$. A bit $c_{2j,1}$ is the output carry of a predictable HA which is component of the (j-1) recoding cell and have the inputs a_{2j-1}, b_{2j-1}.

A HA* (Basic operation- Table II, Fig. 4(a)) output sum is s_{2j+1} and which is produced by driving c_{2j+1} and the twisted by a predictable HA with the fragment a_{2j+1}, b_{2j+1} the same as inputs. Within direct to produce the negatively signal sum s_{2j+1}, the HA* is used and its outputs are given by

$$c_{2j+2,2} = c_{2j+1} \vee (a_{2j+1} \oplus b_{2j+1})$$
$$s_{2j+1} = a_{2j+1} \oplus b_{2j+1} \oplus c_{2j+1} \qquad (8)$$

When we outline the most considerable number (MSD) of the NSMB3 recoding scheme, distinguished the two suitcases, in the first casing the bit width of A and B is even (Fig 6(a)) whereas in the second case both A and B consist of of odd amount of bits (Fig 6(b))

Within casing of even number of bits, the MSD $y_{k,even}^{SD}$ is warning sign digit and is agreed by the relative

$$y_{k,even}^{SD} = -c_{2k,1} + c_{2k,2} \qquad (9)$$

In case that the integer of spot of inputs A and B is odd, the MSD $y_{k,odd}^{SD}$ is a MB digit that is formed based on c_{2k+1}, s_{2k} and $c_{2k,2}$. The carry c_{2k+1} (-) and the sum s_{2k} are produced by a FA** with inputs a_{2k} (-), b_{2k} (-) and c_{2k} (Table VI, Fig 5(b)).

The essential pathway delay of NS-MB1 recorder method is calculated as

$$T_{NSMB1} = T_{HA,Carry} + T_{FA,Carry} + T_{HA^*,Sum} \qquad (10)$$

Where $T_{HA,Carry}$ and $T_{FA,Carry}$ be the hold-up of determining the output carry of a predictable HA and FA correspondingly and $T_{HA^*,Sum}$ is delay of structure the sum of a indication HA*.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

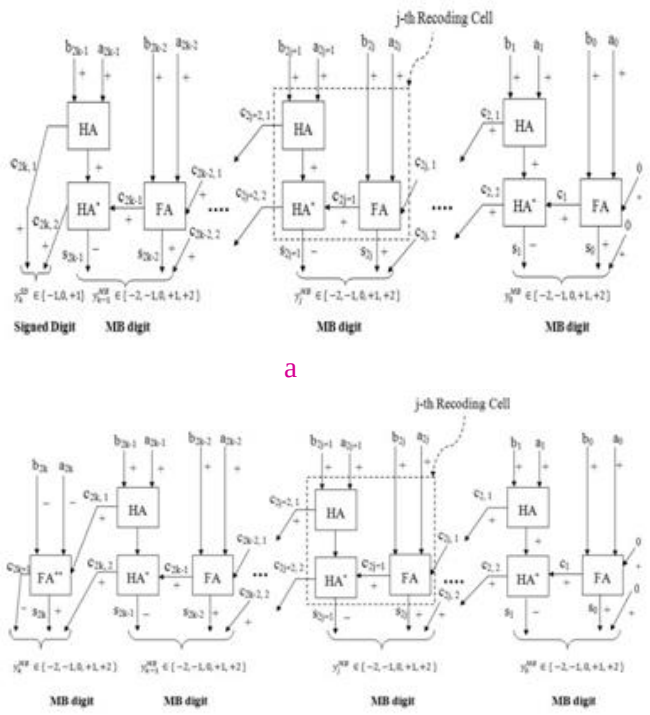


Fig. 6 Implementation of. NS-MB1 system for (a) Even and (b) odd number of bits

NS-MB2 Recoding Scheme

The second approach execute the projected recoding procedure is NS-MB2. It is demonstrate in detail in fig 7 intended for even (fig 7(a)) as well as odd (fig 7(b)) bit width of input information. Regard as the initial values $c_{0,1}$ =0 and $c_{0,2}$ =0. The digitsy_j^{MB}, 0 ≤ j ≤ k-1 stand on s_{2j+1}, s_{2j}, $c_{2j,2}$ according to (7). Again used the predictable FA to construct the carry c_{2j+1} and the sum s_{2j}. A bit $c_{2j,1}$ is the output carry of a HA* (Basic operation-Table II Fig 4(a)), fit in to (j-1) recoding chamber and have input spot a_{2j-1} , b_{2j-1}. The negatively indication bit (s_{2j-1}) formed by a HA** (Table IV, Fig 4(b)) has the inputs c_{2j+1} and the output calculation (negatively signal) of the HA* of the j recoding cell through the small piece a_{2j+1} , b_{2j+1} as contribution. The carry and sum outputs of the HA** are particular by

$$c_{2j+2,2} = c_{2j+1} \wedge \overline{(a_{2j+1} \oplus b_{2j+1})}$$
$$s_{2j+1} = a_{2j+1} \oplus b_{2j+1} \oplus c_{2j+1} \qquad (11)$$

The most significant digit (MSD) for together cases even and odd bit-width of A and B are formed as in NS-MB1 recoding scheme.

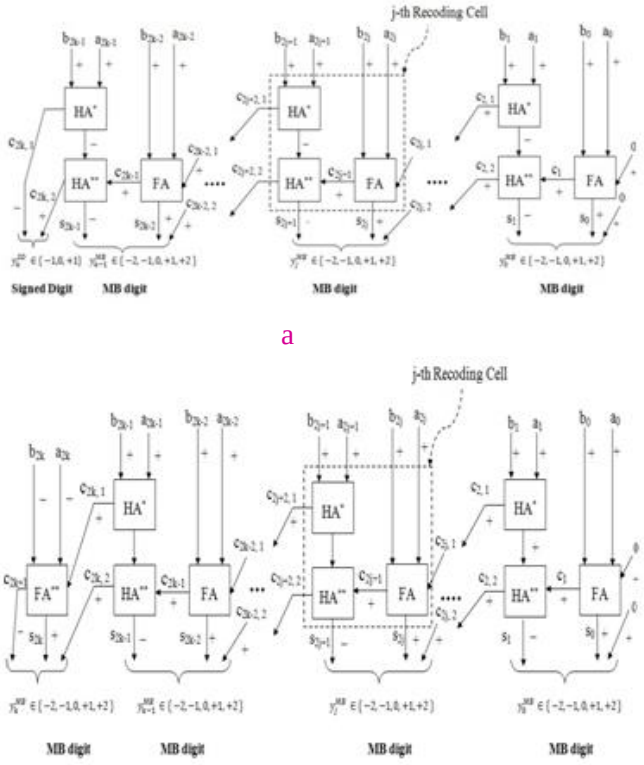


Fig. 7. Implementation of NS-MB2 system for (a) Even and (b) odd number of bits

The essential path delay of NS-MB2 recoding scheme is particular by

$$T_{NS\text{-}MB2} = T_{HA^*,Carry} + T_{FA,Carry} + T_{HA^{**},Sum} \qquad (12)$$

Where $T_{HA^*,Carry}$ *and* $T_{FA,Carry}$ are hold-up of influential the output carry of a signed HA* and FA respectively and $T_{HA^{**},Sum}$ is the hold-up of appearance the sum of a signed HA**.

NS-MB3 Recoding Scheme:

The third approach execute the projected recoding method is NSMB3 and is demonstrate in element in fig 8 for together even (Fig. 8 (a)) as well as odd (Fig.8 (b)) bit-width of input information.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

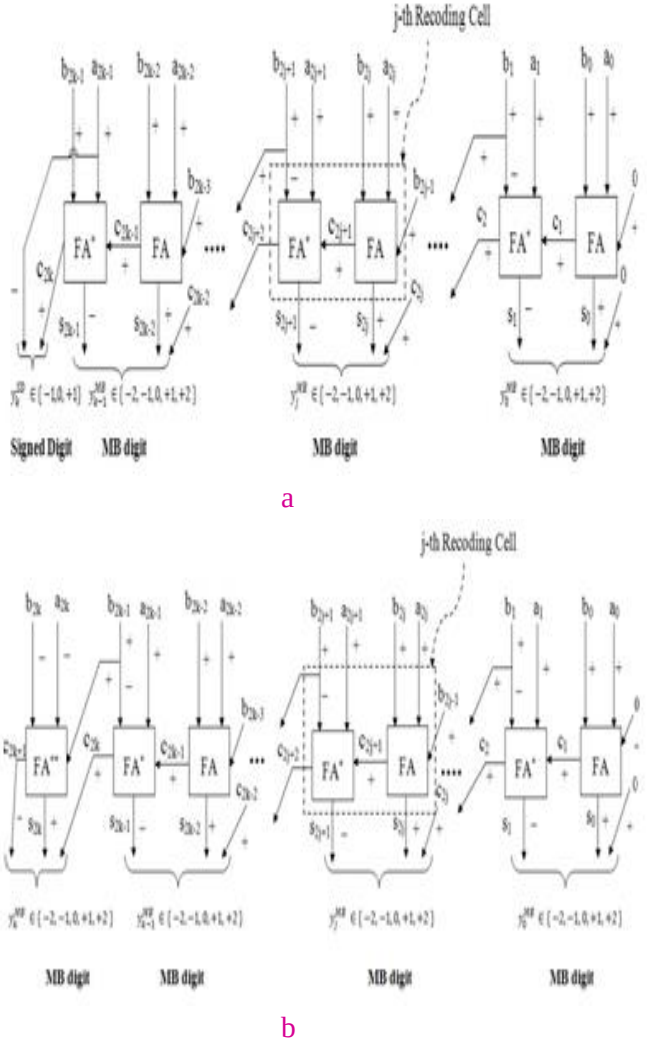


Fig.8. Implementation of NS-MB3 system for (a) Even and (b) odd number of bits

The digits y_j^{MB}, $0 \leq j \leq k-1$ are shaped stand on s_{2j+1}, s_{2j} with c_{2j} according to eq (7). Again used the straight FA to construct the carry c_{2j+1} with sum s_{2j} with participation a_{2j}, b_{2j} and b_{2j-1}. Since the bit s_{2j+1} needs to be negatively signal apply FA* (Table V, Fig 5(a)) with inputs a_{2j+1}, b_{2j+1}(-) and c_{2j-1} which construct the carry c_{2j+2} along with the sum s_{2j+1}(-)

$$c_{2j+2} = (a_{2j+1} \wedge \overline{b_{2j+1}}) \vee (c_{2j+1} \wedge (a_{2j+1} \vee \overline{b_{2j+1}}))$$
$$s_{2j+1} = a_{2j+1} \oplus b_{2j+1} \oplus c_{2j+1} \qquad (13)$$

The MSD $y_{k,even}^{SD}$ is a signed digit and is given by

$$y_{k,even}^{SD} = -a_{2k-1} + c_{2k} \qquad (14)$$

In case that the amount of bits of inputs A with B is odd, the MSD $y_{k,odd}^{SD}$ is a MB digit that is formed based on c_{2k+1}, s_{2k} and c_{2k}. The carry c_{2k+1} (-) and the sum s_{2k} are produced by a FA** with inputs a_{2k} (-), b_{2k} (-) and b_{2k-1} (Table VI, Fig 5(b)).

The essential path hold-up of NS-MB3 Recoder system is invariable in respect to the input bit measurement and is particular by

$$T_{NS\text{-}MB3} = T_{FA,\,Carry} + T_{FA^*,Sum} \qquad (15)$$

Where $T_{FA,\,Carry}$ is the delay of determining the output carry of a predictable FA and $T_{FA^*,Sum}$ is the delay of appearance the sum of a signal FA*.

Unsigned Input Numbers:

In case that the input numbers A and B are unsigned, their most significant bits are positively signed, Fig 9-11 present the modifications that have to make in all NS-MB schemes for both cases of even (The two most significant digits change) and odd (only the most significant digit change) bit width of A and B regarding signs of the most significant bits A and B. The basic recoding block in all schemes remains unchanged.

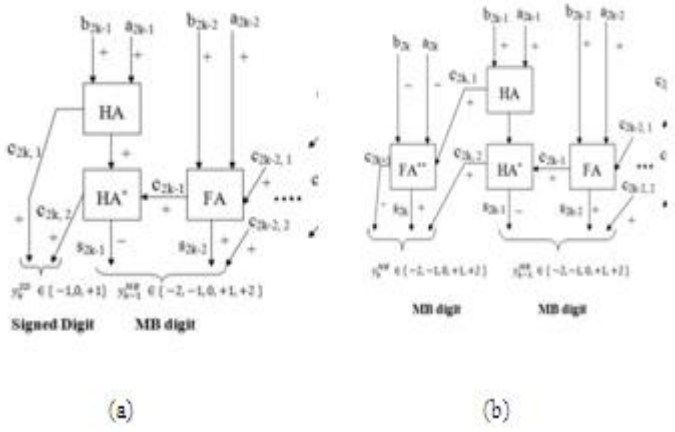


Fig. 9. Implementation of MSD of the NS-MB1 in case of unsigned input numbers (a) Even and (b) odd bit width

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

TABLE-VII
COMPARISON OF DELAY, POWER CONSUMPTION (EVEN BIT WIDTH)

S.NO	Method	Time Delay (ns)	Power consumption (mw)	Power Delay Product (pw)
Ripple Carry Adder				
1	Existing	25.347	135	3421.845
2	NSMB-1	24.811	98	2431.478
3	NSMB-2	24.811	99	2456.289
4	NSMB-3	24.811	96	2381.856
Carry Select Adder				
5	Existing	28.051	130	3646.63
6	NSMB-1	24.764	101	2501.164
7	NSMB-2	24.409	99	2416.491
8	NSMB-3	24.764	99	2451.636
In Built Adder				
9	Existing	17.508	124	2170.992
10	NSMB-1	17.025	100	1702.5
11	NSMB-2	17.025	102	1736.55
12	NSMB-3	16.937	98	1659.826

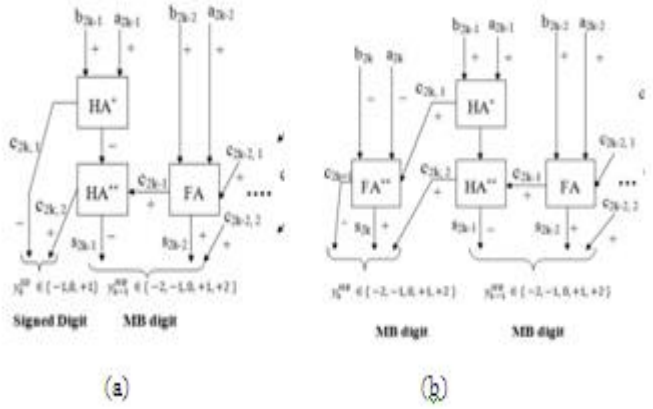


Fig. 10. Implementation of MSD of the NS-MB2 in case of unsigned input numbers (a) Even and (b) odd bit width

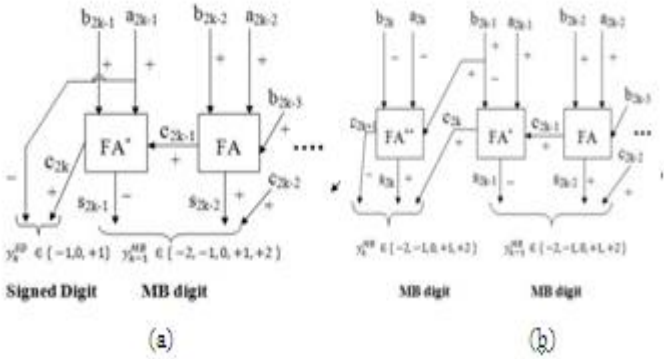


Fig. 11. Implementation of MSD of the NS-MB3 in case of unsigned input numbers (a) Even and (b) odd bit width

IV PERFORMANCE EVALUATION

The performance of the three proposed recoding schemes in a fused add-multiply operator and they are implemented using VHDL for together cases even as well as odd bit-width of the Recoder's input information. To evaluate the presentation of the projected NS-MB schemes by evaluate with the active method in terms of critical delay, power consumption is show in table- VII for Even bit width of the inputs. The inputs with odd bit width and their comparison is shown in table-VIII

The comparison of power consumption with different techniques is shown in fig. 12 for even bit width as well as for odd bit size is show in fig.13

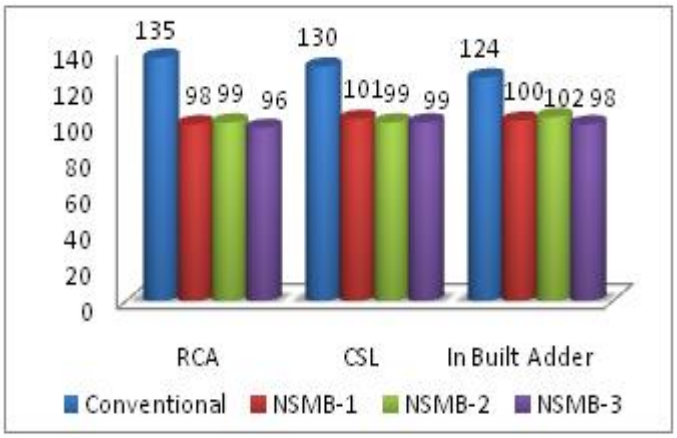


Fig 12. Comparison of power consumption Results (Even Bit Width)

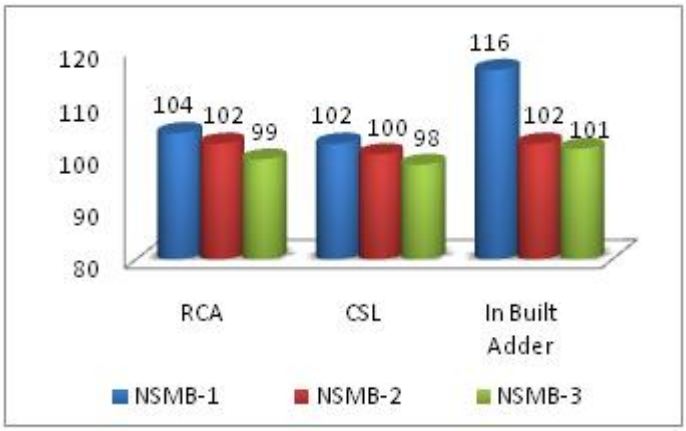


Fig 13. Comparison of power consumption Results (Odd Bit Width)

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

TABLE-VIII
COMPARISON OF DELAY, POWER CONSUMPTION (ODD BIT WIDTH)

S.NO	Method	Time Delay (ns)	Power consumption (mw)	Power Delay Product (pw)
Ripple Carry Adder				
1	NSMB-1	28.972	104	3013.088
2	NSMB-2	28.945	102	2955.144
3	NSMB-3	28.972	99	2868.228
Carry Select Adder				
4	NSMB-1	26.822	102	2735.844
5	NSMB-2	26.822	100	2682.2
6	NSMB-3	26.896	98	2635.808
In Built Adder				
7	NSMB-1	19.593	116	2272.788
8	NSMB-2	19.593	102	1998.486
9	NSMB-3	19.593	101	1978.893

The simulation results of Conventional and proposed schemes for both even and odd bit width is shown in following figures 14-16

Fig.14 Simulation result of Conventional Design

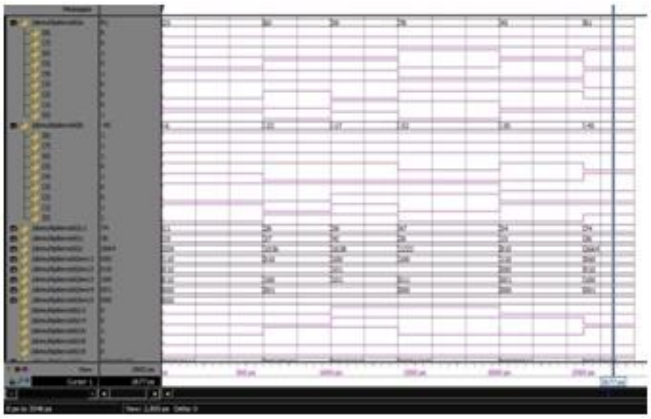


Fig.15 Simulation result of Even Bit Width

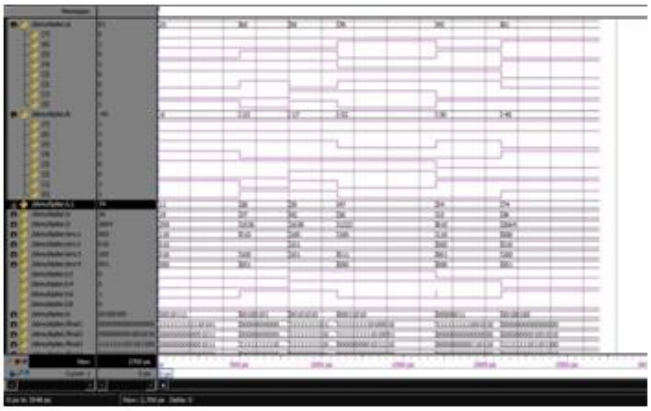


Fig.16 Simulation result of Odd Bit Width

V CONCLUSION

The design of the fused add-multiply is used to execute the straight recoding of the addition of two information in its modified booth (MB) form. This work focuses on optimizing the invent of the Fused add-multiply (FAM) machinist. In this work explored three new alternative designs of the proposed New sum to modifies booth recoding technique (NS-MB) and compared them with the existing method. The proposed recoding schemes incorporated in FAM designs and they give the performance improvements in conditions of critical delay, power expenditure comparing by way of existing method.

REFERENCES

[1] Kostas Tsoumanis, Sotiris Xydis and Kaima Pekmestzi "An Optimized Modified Booth Recoder for Efficient Design of the Add-Multiply Operator", IEEE Trans, Vol. 61, No. 4, April 2014.

[2] A. Amaricai, M. Vladutiu, and O. Boncalo, "Design issues and imple-mentations for ßoating-point divide-add fused," IEEE Trans. Circuits Syst. II–Exp. Briefs, vol. 57, no. 4, pp. 295–299, Apr. 2010.

[3] E. E. Swartzlander and H. H. M. Saleh, "FFT implementation with fused ßoating-point operations," IEEE Trans. Comput., vol. 61, no. 2, pp. 284–288, Feb. 2012.

[4] J. J. F. Cavanagh,DigitalComputer Arithmetic. NewYork: McGraw-Hill, 1984.

[5] S. Nikolaidis, E. Karaolis, and E. D. Kyriakis-Bitzaros, "Estimation of signal transition activity in

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

FIR Þlters implemented by a MAC archi-tecture," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol. 19, no. 1, pp. 164–169, Jan. 2000.
[6] O. Kwon, K. Nowka, and E. E. Swartzlander, "A 16-bit by 16-bit MAC design using fast 5: 3 compressor cells," J. VLSI Signal Process. Syst., vol. 31, no. 2, pp. 77–89, Jun. 2002.
[7] L.-H. Chen, O. T.-C. Chen, T.-Y. Wang, and Y.-C. Ma, "A multiplica-tion-accumulation computation unit with optimized compressors and minimized switching activities," in Proc. IEEE Int, Symp. Circuits and Syst., Kobe, Japan, 2005, vol. 6, pp. 6118–6121.
[8] Y.-H. Seo and D.-W. Kim, "A new VLSI architecture of parallel multiplier–accumulator based on Radix-2 modiÞed Booth algorithm," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 18, no. 2, pp. 201–208, Feb. 2010.
[9] A. Peymandoust and G. de Micheli, "Using symbolic algebra in algo-rithmic level DSP synthesis," in Proc. Design Automation Conf., Las Vegas, NV, 2001, pp. 277–282.
[10] W.-C. Yeh and C.-W. Jen, "High-speed and low-power split-radix FFT," IEEE Trans. Signal Process., vol. 51, no. 3, pp. 864–874, Mar. 2003.
[11] C. N. Lyu and D. W. Matula, "Redundant binary Booth recoding," in Proc. 12th Symp. Comput. Arithmetic, 1995, pp. 50–57.
[12] J. D. Bruguera and T. Lang, "Implementation of the FFT butterßy with redundant arithmetic," IEEE Trans. Circuits Syst. Il, Analog Digit. Signal Process., vol. 43, no. 10, pp. 717–723, Oct. 1996.
[13] W.-C. Yeh, "Arithmetic Module Design and its Application to FFT," Ph.D. dissertation, Dept. Electron. Eng., National Chiao-Tung Univer-sity, , Chiao-Tung, 2001.
[14] R. Zimmermann and D. Q. Tran, "Optimized synthesis of sum-of-prod-ucts," in Proc. Asilomar Conf. Signals, Syst. Comput., PaciÞc Grove, Washington, DC, 2003, pp. 867–872.
[15] B. Parhami, Computer Arithmetic: Algorithms and Hardware De-signs. Oxford: Oxford Univ. Press, 2000.
[16] O. L. Macsorley, "High-speed arithmetic in binary computers," Proc. IRE, vol. 49, no. 1, pp. 67–91, Jan. 1961.
[17] N. H. E. Weste and D. M. Harris, "Datapath subsystems," in CMOS VLSI Design: A Circuits and Systems Perspective, 4th ed. Read-ington: Addison-Wesley, 2010, ch. 11.
[18] S. Xydis, I. Triantafyllou, G. Economakos, and K. Pekmestzi, "Flex-ible datapath synthesis through arithmetically optimized operation chaining," in Proc. NASA/ESA Conf. Adaptive Hardware Syst., 2009, 407–414. [19] [Online]. Available: http://www.synopsys.com/Tools/Implementa-tion/RTLSynthesis/DCUltra/Pages/default.aspx
[20] [Online]. Available: http://www.synopsys.com/Tools/Implementa-tion/SignOff/PrimeTime/Pages/default.aspx [21] Z. Huang, "High-Level Optimization Techniques for Low-Power Mul-tiplier Design," Ph.D., University of California, Department of Com-puter Science, Los Angeles, CA, 2003.
[22] C. S. Wallace, "A suggestion for a fast multiplier," IEEE Trans. Elec-tron. Comput., vol. EC-13, no. 1, pp. 14–17, 1964.
[23] [Online]. Available: http://www.faraday-tech.com/main/IPon-line/category.do?method=showProcessIPList&process=90&catego-ryID=3089&categoryName=Standard%20Cell
[24] M. Daumas and D. W. Matula, "A Booth multiplier accepting both a redundant or a non redundant input with no additional delay," in Proc. IEEE Int. Conf. on Application-SpeciÞc Syst., Architectures, and Pro-cessors, 2000, pp. 205–214.
[25] Z. Huang and M. D. Ercegovac, "High-performance low-power left-to-right array multiplier design," IEEE Trans. Comput., vol. 54, no. 3, pp. 272–283, Mar. 2005.