COPY RIGHT

Title: IMPLEMENTATION OF EFFICIENT MODIFIED BOOTH RECODER USING S-MB TECHNIQUES
Volume 06, Issue 09, Pages: 8– 15.
Paper Authors

M.IMTHIAZ BASHA, S.NAZNEEN SHAGUFA

Gates Institute of Technology, Gooty, Anantapur

USE THIS BARCODE TO ACCESS YOUR ONLINE PAPER

To Secure Your Paper As Per UGC Guidelines We Are Providing A Electronic Bar Code

IMPLEMENTATION OF EFFICIENT MODIFIED BOOTH RECODER USING S-MB TECHNIQUES

M.IMTHIAZ BASHA[1], S.NAZNEEN SHAGUFA [2],

[1] Asst Professor, ECE Department, Gates Institute of Technology, Gooty, Anantapur
[2] Asst Professor, ECE Department, Gates Institute of Technology, Gooty, Anantapur

ABSTRACT:

Complex arithmetic operations are widely used in Digital Signal Processing (DSP) applications. In this work, we focus on optimizing the design of the fused Add-Multiply (FAM) operator for increasing performance.Modern consumer electronics make extensive use of Digital Signal Processing (DSP) providing custom accelerators for the domains of multimedia, communications etc. Typical DSP applications carry out a large number of arithmetic operations as their implementation is based on computationally intensive kernels, such as Fast Fourier Transform (FFT), Discrete Cosine Transform (DCT), Finite Impulse Response (FIR) filters and signals' convolution. We investigate techniques to implement the direct recoding of the sum of two numbers in its Modified Booth (MB) form .We introduce a structured and efficient recoding technique and explore three different schemes by incorporating them in FAM designs. Comparing them with the FAM designs which use existing recoding schemes, the proposed technique yields considerable reductions in terms of critical delay, hardware complexity and power consumption of the FAM unit.

Index Terms—Add-Multiply operation, arithmetic circuits, Modified Booth recoding, VLSI design.

I INTRODUCTION

MODERN consumer electronics make extensive use of Digital Signal Processing (DSP) providing custom accelerators for the domains of multimedia, communications etc. Typical DSP applications carry out a large number of arithmetic operations as their implementation is based on computationally intensive kernels, such as Fast Fourier Transform (FFT), Discrete Cosine Transform (DCT), Finite Impulse Response (FIR) filters and signals' convolution. As expected, the performance of DSP systems is inherently affected by decisions on their design regarding the allocation and the architecture of arithmetic units.Recent research activities in the field of arithmetic optimization have shown that the design of arithmetic components combining operations which share data, can lead to significant performance improvements.Based on the observation that an addition can often be subsequent to a multiplicationthe Multiply-Accumulator (MAC) and Multiply-Add (MAD) units were introduced [3] leading to more efficient implementations of DSP algorithms compared to theconventional ones, which use only primitive resources [4]. Several architectures have been proposed to optimize the performance of the MAC

operation in terms of area occupation, criticalpath delay or power consumption [5]–[7]. As noted in [8], MAC components increase the flexibility of DSP data path synthesis as a large set of arithmetic operations can be efficiently mapped onto them. Except the MAC/MAD operations, many DSP applications are based on Add-Multiply (AM) operations. The straightforward design of the AM unit, by first allocating an adder and then driving its output to the input of a multiplier, increases significantly both area andcritical path delay of the circuit. Targeting an optimized design of AM operators, fusion techniques are employed based on the direct recoding of the sum of two numbers (equivalently a number in carry-save representation [14]) in its Modified Booth (MB) for [15]. Thus, the carry-propagate (or carry-look-ahead) adder [16] of the conventional AM design is eliminated resulting in considerable gains of performance. A special expansion of the preprocessing step of the recoder is needed in order to handle operands in carry-save representation. In [12], the author proposes a two-stage recoder which converts a number in carry-save form to its MB representation. The first stage transforms the carry-save form of the input number into signed-digit form which is then recoded in the second stage so that it matches the form that the MB digits request. Recently, the technique of [12] has been used for the design of high performanceflexible coprocessor architectures targeting the computationally intensive DSP applications [17]. Zimmermann and Tran [13] present an optimized design of [10] which results in

improvements in both area and critical path. In [23], the authors propose the recoding of a redundant input from its carry-save form to the corresponding borrow-save form keeping the critical path of the multiplication operation fixed.Although the direct recoding of the sum of two numbers in its MB form leads to a more efficient implementation of the fused Add-Multiply (FAM) unit compared to the conventionalone, existing recoding schemes are based on complex manipulations in bit-level, which are implemented by dedicated circuits in gate-level. This work focuses on the efficient design of FAM operators, targeting the optimization of the recoding scheme for direct shaping of the MB form of the sum of two numbers (Sum to MB – *S-MB*). More specifically, we propose a new recodingtechnique which decreases the critical path delay and reduces area and power consumption. The proposed *S-MB* algorithm is structured, simple and can be easily modified in order to be applied either in signed (in 2's complement representation) or unsigned numbers, which comprise of odd or even number of bits. We explore three alternative schemes of the proposed *S-MB* approach using conventional and signed-bit Full Adders (FAs) and Half Adders (HAs) as building blocks. We evaluated the performance of the proposed *S-MB* technique by comparing its three different schemes with the state-of the-art recoding techniques [12], [13], [23]. Industrial tools for RTL synthesis [18] and power estimation [19] have been used to provide accurate measurements of area utilization, critical path delay and power

dissipation regarding various bit-widths of the input numbers. We show that the adoption of the proposed recoding technique delivers optimized solutions for the FAM design enabling the targeted operator to be timing functional (no timing violations) for a larger range of frequencies. Also, under the same timing constraints, the proposed designs deliver improvements in both area occupation and power consumption, thus outperforming the existing *S-MB* recoding solutions

II MODIFIED BOOTH FORM

Modified Booth (MB) is a prevalent form used in multiplication . It is a redundant signed-digit radix-4 en the number of partial products in multiplication comparing to any other radix-2 representation. Let us consider the multiplication of 2's complement numbers and with each number consisting of. The multiplicand can be represented in MB form as:

$$Y = \langle y_{n-1} y_{n-2} \cdots y_1 y_0 \rangle_{2's} = -y_{2k-1} \cdot 2^{2k-1} + \sum_{i=0}^{2k-2} y_i \cdot 2^i$$

$$= \langle \mathbf{y}_{k-1}^{MB} \mathbf{y}_{k-2}^{MB} \cdots \mathbf{y}_1^{MB} \mathbf{y}_0^{MB} \rangle_{MB} = \sum_{j=0}^{k-1} \mathbf{y}_j^{MB} \cdot 2^{2j} \quad (1)$$

$$\mathbf{y}_j^{MB} = -2y_{2j+1} + y_{2j} + y_{2j-1}. \quad (2)$$
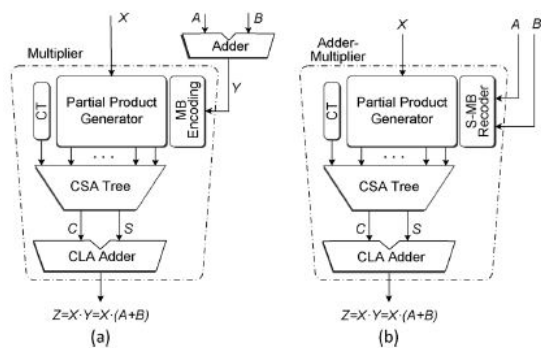

Fig. 1. AM operator based on the (a) conventional design and (b) fused design

with direct recoding of the sum of A and B in its MB representation. The multiplier is a basic parallel multiplier based on the MB algorithm. The terms CT, CSA Tree and CLA Adder are referred to the Correction Term, the Carry-Save Adder Tree and the final Carry-Look-Ahead Adder of the multiplier.

III.SUM-PRODUCT IMPLEMENTATION

The direct recoding of the sum of two numbers in its MB form leads to a more efficient implementation of the fused Add-Multiply (FAM) unit compared to the existing. The efficient design of FAM operators are used to attain the optimization of the recoding for direct shaping of the MB form of the sum of two numbers (Sum to MB). The Sum- Modified Booth algorithm is structured, simple and can be easily modified in order to be applied in signed or unsigned numbers, which consist of odd or even number of bits. Sum-Modified Booth algorithm using unsigned and signed-bit Full Adders and Half Adders. The recoding technique reveals efficient solutions for the FAM design to obtain the targeted operator and timing functional for a larger range of frequencies. It obtains a significant improvement in both area and power consumption.The fused Add-Multiply (FAM) contains only one adder at the end . As a result, significant area savings are observed and the critical path delay of the recoding process is reduced and decoupled from the bit-width of its inputs. We present a new technique for direct recoding of two numbers in the MB representation of their sum. Compare to radix-2 representation

International Journal for Innovative Engineering and Management Research
A Peer Reviewed Open Access International Journal
www.ijiemr.org

radix-4 is more efficient. The area saving is the most significant, since in most signal processing applications, It can be employed to achieve high throughput . Each digit is represented by three bits named s, one andtwo. The sign bit shows if the digit is negative (s=1) or positive(s=0).In MBRS recoding technique, the sum of two consecutive bits of the input A and B are converted into MB digit. The most significant digit is negatively weighted while the two least significant digits is positive weight The two pairs of bits in MB form are used signed-bit arithmetic. For this purpose, a set of bit-level signed Full Adders (FA) are consider here to calculate the carry and sum. The schematics and Boolean equation for even and odd bit half adders is shown in Fig.1 and Fig. 2.

TABLE – 1 Modified Booth Encoding Table

Binary			MMB Encoding			carry
Y_{2j+1}	Y_{2j}	Y_{2j-1}	Sign= sj	onej	twoj	
0	0	0	0	0	0	0
0	0	1	0	1	0	0
0	1	0	0	1	0	0
0	1	1	0	0	1	0
1	0	0	1	0	1	1
1	0	1	1	1	0	1
1	1	0	1	1	0	1
1	1	1	1	0	0	0

Full Adder for Odd Bit, Carry = $((\overline{p}) \wedge ci) \vee (p \wedge \overline{\ })$, Sum = p XOR q XOR ci
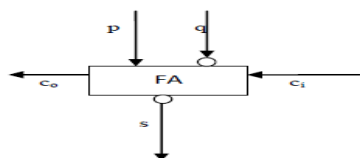

Fig. 1 Boolean equation and schematics for half adder odd bit

Full Adder for Even Bit, Carry = $((p \vee q) \wedge \overline{\ }) \vee (p \wedge q)$, Sum = p XOR q XOR ci
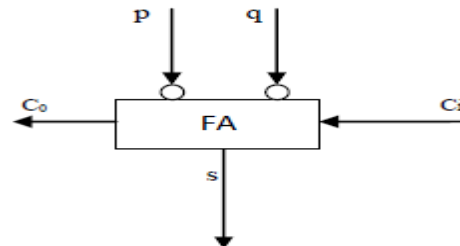

Fig. 2 Boolean equation and schematics for half adder even bit

The basic operation of full adder for odd bit and even bit is shown in the Table 4 and 5 respectively. Considering that p, q and are the binary inputs and c0, s are the outputs (carry and sum respectively) of a full adder for even bits. It implements the relation 2·c−s = p − q + where the bits s and q are considered negatively signed (Table 4). Table 5 show the operation of full adder for odd bits which implements the relation 2·c+s = −p–q+ and manipulates a negative (q) and a positive (p) inputs.In the proposed recoding technique is referred as MBRS. For the given circuits Fig 5 Full adder is modified to obtain low power and high speed. The sum of A and B is given by the next relation

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

$$\text{Where } y_j^{MB} = -2s_{2j+1} + s_{2j} + c_{2j}$$

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

Table –2 Truth Table For Full Adder Odd Parity Dual Operation

Inputs			Output Value	Output	
p(+)	q(-)	c_i		Carry, C	Sum, 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 = $2.c_0 - s = p - q + c_i$

Table –3 Truth Table For Full Adder Even Parity Dual Operation

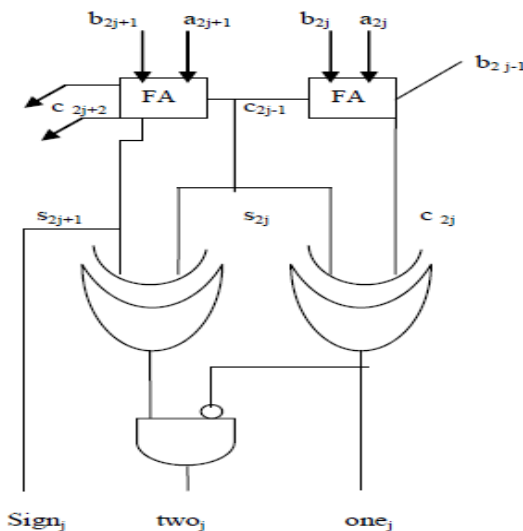| Inputs			Output Value	Output	
p(-)	q(-)	c_i		Carry, C	Sum, S
0	0	0	0	0	0
0	0	1	+1	0	1
0	1	0	-1	1	1
0	1	1	0	0	0
1	0	0	-1	1	1
1	0	1	0	0	0
1	1	0	-2	1	0
1	1	1	-1	1	1

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

Fig. 3 Circuit diagram for proposed MBRS

The critical path delay of MBRS recoding scheme can be calculated as:

$$T_{MBRS-2} = T_{FA\ Carry} + T_{FA\ Sum}$$

Where $T_{FA,Carry}$ are the delays of shaping the output carry of a conventional even bit Full Adder respectively and $T_{HA,Sum}$ is the delay of forming the sum of a signed odd bit full adder.

IV SUM TO MODIFIED BOOTH RECODING TECHNIQUE (*S-MB*)

A. Defining Signed-Bit Full Adders and Half Adders for Structured Signed ArithmeticIn *S-MB* recoding technique, we recode the sum of two consecutive bits of the input ( ) with two consecutive bits of the input ( ) into one MB digit . As we observe from (2), three bits are included in forming a MB digit. The most significant of them is negatively weighted while the two least significant of them have positive weight. Consequently, in order to transform the two aforementioned pairs of bits in MB form we need to use signed-bit arithmetic. For this purpose, we develop a set of bit-level signed Half Adders (HA) and Full Adders (FA) considering their inputs and outputs to be signed.More specifically, in this work, we use two types of signed HAs which are referred as HA* and HA**. Tables II - IV are their truth tables and in Fig. 4 we present their corresponding Boolean equations. Considering that , are the binary inputs and , are the outputs (carry and sum respectively) of a HA* which implements the relation where the sum is considered negatively

signed (Table II, Fig. 4(a)), the output takes one of the values . In Table III, we also describe the dual implementation of HA* where we inversed the signs of all inputs and outputs and, consequently, changed the output values to . Table IV and Fig. 4(b) show the operation and schematic of HA** which implements the relation and manipulates a negative ( ) and a positive ( ) input resulting in the output values .

SMB-1 RECODING SCHEME

The first scheme of the proposed recoding technique is referred as *S-MB*1 and is illustrated in detail in Fig.4 for both even and odd bit-width of input numbers. As can be seen in Fig.4, the sum of A and B is given by the next relation:

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

Where $y_j^{MB} = -2s_{2j+1} + s_{2j} + c_{2j}$

The encoding of the MB digits y_j^{MB}, $0 \le j \le$ k -1, of is based on the analysis. Bits s2j+1 and s2j are extracted from the j recoding cell of Fig.4. A conventional FA with inputs a2j, b2j and b2j-1 produces the carry c2j+1 = (a2j $\wedge$ b2j) V (b2j-1 $\wedge$ (a2j V b2j)) and the sum s2j = a2j $\oplus$ b2j $\oplus$ b2j-1. As the bit s2j+1 need to be negatively signed, use a FA* with inputs a2j+1, b2j+1 (-) and c2j+1, which produces the carry c2j+2 and the sum s2j+1 (-):

$$C_{2j+2} = (a_{2j+1} \wedge \overline{b}_{2j+1}) \, v \, (c_{2j+1} \wedge (a_{2j+1} v \overline{b}_{2j+1}))$$

Note that, based on equation b2j+1 = 2 * b2j+1 – b2j+1, b2j+1 is driven to the FA* as negatively signed while it is also used with positive sign as an input carry of the subsequent recoding cell. It consider the

initial values b-1 = 0 and c0 = 0. When its form the most significant digit (MSD) of the *S-MB*1 recoding scheme, it distinguish two cases: In the first case, the bit-width of A and B is even, while in the second case, both A and B comprise of odd number of bits. In the first case, the MSD is a signed digit and is given by the next algebraic equation:

$$y_{k,even}^{SD} = -a_{2k-1} + c_{2k}$$

The critical path delay of *S-MB*1 recoding scheme is constant in respect to the input bit-width and is given by the equation:

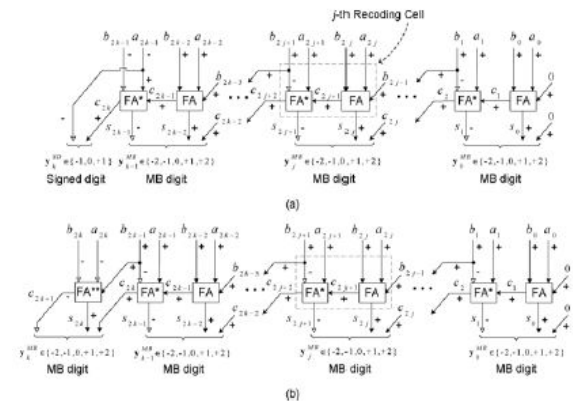TS-MB1 = TFA, carry + TFA*, sum (7)

Fig.4 *S-MB*1 recoding scheme for (a) even and (b) odd number of bits.

SMB recoding scheme 2:

The second approach of the proposed recoding technique, S-MB2, is described in Fig.5 for even and odd bit-width of input numbers. It consider the initial values c0,1 = 0 and c0,2 = 0. The digits y_j^{MB}, $0 \le j \le$ k -1, are formed based on s2j+1, s2j and c2j,2 . As in the S-MB1 recoding scheme, it uses a conventional FA to produce the carry c2j+1 and the sum s2j. The inputs of the FA are a2j, b2j and c2j,1. The bit c2j,1 is the output carry of a conventional HA which is part of the (j-1) recoding cell and has the bits a2j-1,

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

b2j-1 as inputs. The bit s2j+1 is the output sum of a HA* in which it drives c2j+1 and the sum produced by a conventional HA with the bits a2j+1, b2j+1 as inputs. The HA* is used in order to produce the negatively signed sum s2j+1 and its outputs are given by the following Boolean equations:

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

$$s_{2j+1} = a_{2j+1} \oplus b_{2j+1} \oplus c_{2j+1}$$

In case that A and B comprise of even number of bits, a2n-1 and b2n-1 are negatively weighted and the conventional HA of the (n-1) recoding cell is replaced by the dual HA* The MSD is a signed digit and is given by the relation:
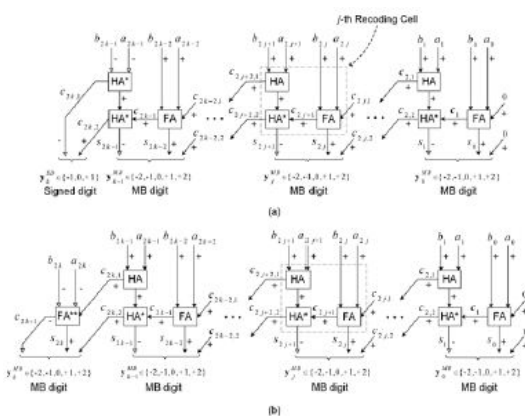
$$y_{k,even}^{SD} = -c2k,1 + c2k,2$$

Fig.5 *S-MB*2 recoding scheme for (a) even and (b) odd number of bits

SMB 3 recoding scheme:
The third scheme implementing the proposed recoding technique is *S-MB*3. It is illustrated in detail in for even and odd bit-width of input numbers. It consider that c0,1

= 0 and c0,2. It builds the digits , $0 \leq j \leq k - 1$, based on s2j+1, s2j and c2j,2 . y_j^{MB}

Once more, it uses a conventional FA to produce the carry c2j+1 and the sum s2j. The bit c2j,1 is now the output carry of a HA* which belongs to the (j-1) recoding cell and has the bits a2j-1, b2j-1 as inputs. The negatively signed bit s2j+1 is produced by a HA** in which drive c2j+1 and the output sum (negatively signed) of the HA* of the recoding cell with the bits a2j+1, b2j+1 as inputs. The carry and sum outputs of the HA** are given by the following Boolean equations:

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

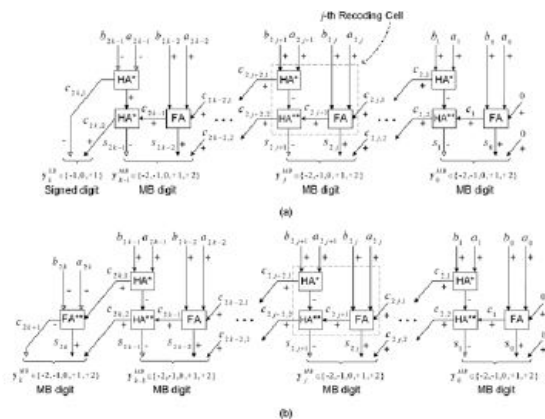$$s_{2j+1} = a_{2j+1} \oplus b_{2j+1} \oplus c_{2j+1}. \qquad (15)$$

Fig.6 *S-MB*3 recoding scheme for (a) even and (b) odd number of bits.

V. SIMULATION RESULTS

The modified booth recorder written in Verilog, compiled and simulation using modelsim. The circuit simulated and synthesized. The simulated result for modified booth recorder
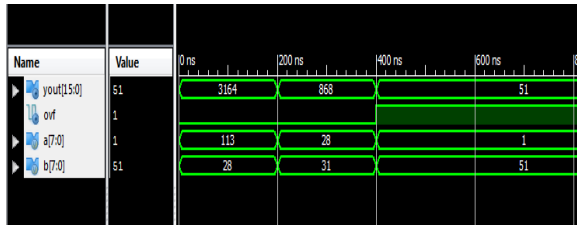
Fig. 7 Simulation Result.

VI CONCLUSION

This paper focuses on optimizing the design of the Fused-Add Multiply (FAM) operator.We propose a structured technique for the direct recoding of the sum of two numbers to its MB form. We explore three alternative designs of the proposed *S-MB* recoder and compare them to the existing ones and The proposed recoding schemes, when they are incorporated in FAM designs, yield considerable performance improvements in comparison with the most efficient recoding schemes found in literature.

REFERENCES

[1] A. Amaricai, M. Vladutiu, and O. Boncalo, "Design issues and implementationsfor floating-point divide-add fused," *IEEE Trans. CircuitsSyst. II–Exp. Briefs*, vol. 57, no. 4, pp. 295–299, Apr. 2010.

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

[3] J. J. F. Cavanagh,*Digital Computer Arithmetic*. NewYork:McGraw-Hill, 1984.

[4] S. Nikolaidis, E. Karaolis, and E. D. Kyriakis-Bitzaros, "Estimation ofsignal transition activity in FIR filters implemented by a MAC architecture,"*IEEE Trans.*

Comput.-Aided Des. Integer. Circuits Syst., vol.19, no. 1, pp. 164–169, Jan. 2000.

[5] O. Kwon, K. Nowka, and E. E. Swartzlander, "A 16-bit by 16-bitMACdesign using fast 5: 3 compressor cells," *J. VLSI Signal Process. Syst.*,vol. 31, no. 2, pp. 77–89, Jun. 2002.

[6] L.-H. Chen, O. T.-C. Chen, T.-Y.Wang, and Y.-C. Ma, "A multiplication-accumulation computation unit with optimized compressors andminimized switching activities," in *Proc. IEEE Int, Symp. Circuits andSyst.*, Kobe, Japan, 2005, vol. 6, pp. 6118–6121.

[7] Y.-H. Seo and D.-W. Kim, "A new VLSI architecture of parallelmultiplier–accumulator based on Radix-2 modified Booth algorithm,"*IEEE Trans. Very Large Scale Integer. (VLSI) Syst.*, vol. 18, no. 2, pp.201–208, Feb. 2010.

[8] A. Peymandoust and G. de Micheli, "Using symbolic algebra in algorithmiclevel DSP synthesis," in *Proc. Design Automation Conf.*, LasVegas, NV, 2001, pp. 277–282.

[9] W.-C. Yeh and C.-W. Jen, "High-speed and low-power split-radixFFT," *IEEE Trans. Signal Process.*, vol. 51, no. 3, pp. 864–874, Mar.2003.

[10] C. N. Lyu and D. W. Matula, "Redundant binary Booth recoding," in*Proc. 12th Symp. Comput. Arithmetic*, 1995, pp. 50–57.

[11] J. D. Bruguera and T. Lang, "Implementation of the FFT butterflywith redundant arithmetic," *IEEE Trans. Circuits Syst. Il, Analog Digit.Signal Process.*, vol. 43, no. 10, pp. 717–723, Oct. 1996.