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

A Modified Partial Product Generator for Redundant Binary Multipliers

Pinjari Subhan Basha
M.Tech (VLSI Design),
Department of ECE,
VIF College of Engineering and Technology, Hyderabad, T.S, India.

Mr.B.Kotesh
Assistant Professor,
Department of ECE,
VIF College of Engineering and Technology, Hyderabad, T.S, India.

Imthiazunnisa Begum
Assistant Professor & HOD,
Department of ECE,
VIF College of Engineering and Technology, Hyderabad, T.S, India.

ABSTRACT:

Due to its high modularity and carry-free addition, redundant binary (RB) representation can be used when designing high performance multipliers. The conventional RB multiplier requires and additional RB partial product (RBPP) row, because an error-correcting word (ECW) is generated by both the radix-4 Modified Booth encoding (MBE) and the RB encoding. This incurs in an additional RBPP accumulation stage for the MBE multiplier. In this paper, a new RB modified partial product generator (RBMPPG) is proposed; it removes the extra ECW and hence, it saves one RBPP accumulation stage. Therefore, the proposed RBMPPG generates fewer partial product rows than a conventional RB MBE multiplier. Simulation results show that the proposed RBMPPG based designs significantly improve the area and power consumption when the word length of each operand in the multiplier is at least 32 bits; these reductions over previous NB multiplier designs incur in a modest delay increase (approximately 5%). The power-delay product can be reduced by up to 59% using the proposed RB multipliers when compared with existing RB multipliers.

I. INTRODUCTION:

Digital multipliers are widely used in arithmetic units of microprocessors, multimedia and digital signal processors. Many algorithms and architectures have been proposed to design high-speed and low power multipliers. A normal binary (NB) multiplication by digital circuits includes three steps. In the first step, partial products are generated; in the second step, all partial products are added by a partial product reduction tree until two partial product rows remain. In the third step, the two partial product rows are added by a fast carry propagation adder. Two methods have been used to perform the second step for the partial product reduction. A first method uses 4-2 compressors, while a second method uses redundant binary (RB) numbers. Both methods allow the partial product reduction tree to be reduced at a rate of 2:1. The redundant binary number representation has been introduced by Avizienis to perform signed-digit arithmetic; the RB number has the capability to be represented in different ways. Fast multipliers can be designed using redundant binary addition trees.

Alternatively, a high-radixBooth encoding technique can reduce the number of partial products. However, the number of expensive hard multiples (i.e., a multiple that is not a power of two and the operation cannot be performed by simple shifting and/or complementation) increases too [14-16]. Besliet al. [16] noticed that some hard multiples can be obtained by the differences of two simple power-of-two multiplies. A new radix-16 Booth encoding (RBBE-4) technique without ECW has been proposed in [14]; it avoids the issue of hard multiples. A radix-16 RB Booth encoder can be used to overcome the hard multiple problem and avoid the extra ECW, but at the cost of doubling the number of RBPP rows. Therefore, the number of radix-16 RBPP rows is the same as in the radix-4 MBE.

However, the RBPP generator based on a radix-16 Booth encoding has a complex circuit structure and a lower speed compared with the MBE partial product generator [10] when requiring the same number of partial products.

II. RB PARTIAL PRODUCT GENERATOR:

As two bits are used to represent one RB digit, then a RBPP is generated from two NB partial products. The addition of two N-bit NB partial products X and Y using two‘s complement representation can be expressed as follows: where is the inverse of, and the same convention issued in the rest of the paper. The composite number can be interpreted as a RB number. The RBPP is generated by inverting one of the two NB partial products and adding -1 to the LSB. Each RB digit _ belongs to the set _1, 0, 1_; this is coded by two bits as the pair. Note that 1 = −1. RB numbers can be coded in several ways. Table II shows one specific RB encoding, where the RB digit is obtained by performing Both MBE and RB coding schemes introduce errors and two correction terms are required:

1) when the NB number is converted to a RB format, -1 must be added to the LSB of the RB number; 2) when the multiplicand is multiplied by -1 or -2 during the Booth encoding, the number is inverted and +1 must be added to the LSB of the partial product. A single ECW can compensate errors from both the RB encoding and the radix-4 Booth recoding. The conventional partial product architecture of an 8-bit MBE multiplier is shown in Fig. 1, where b_ represents the bit position, and is generated by using an encoder and decoder (Fig. 2). An N-bit CRBBE-2 multiplier includes N/4 RBPP rows and one ECW; the ECW takes the form as follows: where i represent the ithrow of the RBPPs. In a correction term is always required by RB coding. If also corrects the errors from the MBE recoding, then the correction term cancels out to 0. That is to say that if the multiplicand digit is inverted and added to 1, then is 0, otherwise is -1. The error-correcting digit is determined only by the Booth encoding, no negative encoding1, negative encoding.

III. LITERATURE SURVEY:

In conventional digital computers, integers are represented as binary numbers of fixed length. There are several other number systems that are useful for certain applications. These include the Redundant Signed Digit Number System. In a redundant signed digit representation with radix r each digit is allowed to take more than r-values. Redundancy in representation makes faster addition and subtraction in which each sum or difference digit is a function of only the digits in the adjacent portions of the operands. In this project work we emphasize on the floating point representation of XLU numbers, which is based on signed digits. And various rounding schemes for them, namely truncate, nearest to zero, nearest to even, nearest to positive and negative infinity are discussed. Redundancy in the number system allows a method for fast addition and subtraction. Carry free addition is an attractive property of redundant signed digit number. The requirement for totally parallel addition and subtraction determines the minimum redundancy (r 2 values) which is necessary in the representation of one digit.

Low Power and High Speed Multiplication Design through Mixed Number Representations

A low power multiplication algorithm and its VLSI architecture using a mixed number representation is proposed. The reduced switching activity and low power dissipation are achieved through the Sign-Magnitude (SM)notation for the multiplicand and through a novel design of the Redundant Binary (RB) adder and Booth decoder .The high speed operation is achieved through the Carry-Propagation-Free (CPF} accumulation of the Partial Products (PP) by using the RB notation. Analysis showed that the switching activity in the PP generation process can be reduced on average by 90%. Compared to the same type of multipliers [I, 2, 31, the proposed design dissipates much less power and is 18% faster on average.

Posibits, Negabits, and Their Mixed Use in Efficient Realization of Arithmetic Algorithms

Positively weighted and negatively weighted bits (posibits, negabits) have been used in the interpretation of 2‘scomplement, negative-radix, and binary signed-digit number representation schemes as a way of facilitating the development of efficient arithmetic algorithms for various application domains. In this paper, we show that a more general view of posibits and negabits, along with their mixed use in any combination (using inverse encoding for negabits), unifies a number of diverse implementation schemes, while at the same time making the resultant designs more efficient by avoiding custom or modified hardware elements and restricting the implementation to the use of standard arithmetic cells. Such standard cells have been highly optimized and are continually improving due to their wide applicability.

High Speed 16×16-bit Low-Latency Pipelined Booth Multiplier

This paper presents a high-speed 16×16-bit CMOS pipelined booth multiplier. Actually in an n-bit modified Booth multiplier, because of the last sign bit, n/2 +1 partial product rows are generated rather than n/2. The extra row not only increases the delay and power consumption of Wallace tree, but also it leads to irregularity and complexity of Wallace tree designing. In this multiplier the last sign bit is removed by using a simple high-speed approach. This causes 4% reduction in power consumption and 5.2% reduction in transistor count. Also by using new partial product generation and booth encoder circuits and a novel adder, speed of pipelined multipliers is improved. By these new architectures, final adder performs 25 bit addition in only two cycles with high speed (1.6 GHz). Due to lower number of cycles (5 clock cycles), delay of the overall circuit is only 3.1ns and besides power consumption is decreased so that at a data rate of 1 GHz and under the supply voltage of3.3V, power consumption is 169mW.

Arithmetical Operations in Quaternary System Using

While performing the several arithmetic operations such as addition, subtraction and multiplication the speed of modern computers are limited because of carry propagation delay. A carry-free arithmetic operation can be achieved using higher radix number system such as Quaternary Signed Digit(QSD). In QSD, each digit can be represented by a number from-3 to 3. Using this number system any integer can be represented in multiple ways. With constant or fix delay and less complexity carry free addition, multiplication and other operations can be implemented on large number of digits. This paper deals with the implementation of QSD based arithmetic operations. The designs are simulated and synthesized using VHDL software, Model sim software is used for simulation. Arithmetic operations are widely used and play an important role in various digital systems such as computers and signal processors.

High-Speed Booth Encoded Parallel Multiplier Design

This paper presents a design methodology for high-speed Booth encoded parallel multiplier. For partial product generation, we propose a new modified Booth encoding (MBE) scheme to improve the performance of traditional MBE schemes. For final addition, a new algorithm is developed to construct multiple-level conditional-sum adder (MLCSMA). The proposed algorithm can optimize final adder according to the given cell properties and input delay profile. Compared with a binary tree-based conditional-sum adder, the speed performance improvement is up to 25 percent. On average, the design developed herein reduces the total delay by 8 percent for parallel multiplier. The whole design has been verified by gate level simulation.IN various computing and signal processing applications, parallel multiplier has been a basic building block for many algorithms. Many high performance algorithms and architectures have been proposed to accelerate multiplication.

IV. PROPOSED RB PARTIAL PRODUCT GENERATOR

A new RB modified partial product generator based on MBE (RBMPPG-2) is presented in this section; in this design, ECW is eliminated by incorporating it into both the two MSBs of the first partial product row and the two LSBs of the last partial product row. It is different from the scheme in Fig. 1, where all the error-correcting terms are in the last row. ECW1 is generated by PP_ and expressed as ECW_ = 0 E__ 0 F__. (7)The ECW2 generated by PP_ (also defined as an extra ECW) is left as the last row and it is expressed to eliminate a RBPP accumulation stage, ECW2 needs to be incorporated. A modified radix-4 Booth encoding and a decoding circuit for the partial product are proposed here (Fig.4); an extra 3-input OR gate is then added to the design of [10] (Fig. 2). The three inputs of the additional OR gate are, when, it is clear that, and is set to all ones.

PROPOSED RBMPPG-2:

The circuit diagrams of the modified partial product variables and are shown in Fig. 5. It is clear that has the longest delay path. It is well known that the inverter, the 2-input NAND gate and the transmission gate (TG) are faster than other gates. So, it is desirable to use TGs when designing the multiplexer [5-6]. As shown in Fig. 5 (a), the critical path delay (the dash line) consists of a 1-stage AND-OR-Inverter gate, a 1-stageinverter, and 2-stage TGs. Therefore, RBMPPG-2 just increases the TG delay by 1-stage compared with the MBE partial product of Fig. 2.The above discussion is only an example; the above technique can be applied to design any 2_-bit RB multipliers. It eliminates the extra ECWN/4 and saves one RBPP accumulation stage, i.e., three XOR gate delays, while only slightly increasing the delay of the partial product generation stage. In general, an N-bit RB multiplier has __/4_ RBPP rows using the proposed RBMPPG-2.

RADIX-4 BOOTH ENCODING

Booth encoding has been proposed to facilitate the multiplication of two's complement binary numbers. It was revised as modified Booth encoding (MBE) or radix- 4 Booth encoding. The multiplier bits are grouped in sets of three adjacent bits. The two side bits are overlapped with neighboring groups except the first multiplier bits group in which it is {b1, b0, 0}. Each group is decoded by selecting the partial product shown in Table I, where 2A indicates twice the multiplicand, which can be obtained by left shifting. Negation operation is achieved by inverting each bit of A and adding ‗1‘ (defined as correction bit) to the LSB [10-13]. Methods have been proposed to solve the problem of correction bits for NB radix-4Booth encoding (NBBE-2) multipliers. However, this problem has not been solved for RB MBE multipliers.

RB PARTIAL PRODUCT GENERATOR:

As two bits are used to represent one RB digit, then a RBPP is generated from two NB partial products [1-6].The addition of two N-bit NB partial products X and Y using two‘s complement representation can be expressed as follows Both MBE and RB coding schemes introduce errors and two correction terms are required: 1) when the NB number is converted to a RB format, -1 must be added to the LSB of the RB number; 2) when the multiplicand is multiplied by -1 or -2 during the Booth encoding, the number is inverted and +1 must be added to the LSB of the partial product. A single ECW can compensate errors from both the RB encoding and the radix-4 Booth recoding. The conventional partial product architecture

PROPOSED RB PARTIAL PRODUCT GENRATOR:

The circuit diagrams of the modified partial product variables and are shown in Fig. 5. It is clear that has the longest delay path. It is well known that the inverter, the 2-input NAND gate and the transmission gate (TG) are faster than other gates. So, it is desirable to use TGs when designing the multiplexer [5-6]. As shown in Fig. 5 (a), the critical path delay (the dash line) consists of a 1-stage AND-OR-Inverter gate, a 1-stage inverter, and 2-stage TGs.

Therefore, RBMPPG-2 just increases the TG delay by 1-stage compared with the MBE partial product of Fig. 2. The above discussion is only an example; the above technique can be applied to design any 2_-bit RB multipliers.

V. DESIGN OF RBMPPG-2-BASED HIGH-SPEED RB MULTIPLIERS:

The proposed RBMPPG-2 can be applied to any 2_-bit RB multipliers with a reduction of a RBPP accumulation stage compared with conventional designs. Although the delay of RMPPG-2 increases by 1-stage of TG delay, the delay of one RBPP accumulation stage is significantly larger than a 1-stage TG delay. Therefore, the delay of the entire multiplier is reduced. The improved complexity, delay and power consumption are very attractive for the proposed design. A 32-bit RB MBE multiplier using the proposed RBPP generator is shown in Fig. 6. The multiplier consists of the proposed RBMPPG-2, three RBPP accumulation stages, and one RB-NB converter. Eight RBBE-2 blocks generate the RBPP; they are summed up by the RBPP reduction tree that has three RBPP accumulation stages.

PERFORMANCE EVALUATION:

The performance of various 2_-bit RB multipliers using the proposed RBMPPG-2 is assessed; the results are compared with NBBE-2, CRBBE-2 and RBBE-4 [14] multipliers that are the latest and best designs found in the technical literature. All designs of RB multipliers use the RBFA and RBHA of [7]. An RB-NB converter is required in the final stage of the RB multiplier to convert the summation result in RB form to a two‘s complement number. It has been shown that the constant-time converter in [7] does not exist [19-21]. However, there is a carry-free multiplier that uses redundant adders in the reduction of partial products by applying on-the-fly conversion [22] in parallel with the reduction and generates the product without a carry-propagate adder [23-24].A hybrid parallel-prefix/carry-select adder [25] is used for the final RB-NB converter. The NBBE-2 multiplier design uses the same encoder and decoder as shown in Fig. 2. 4-2 compressors [26-28] are used in the partial product reduction tree. The extra ECW in the NB multiplier designs is also modified as proposed in [11]. Table VI summarizes the delay, area, power and power-delay product (PDP) of the NB and RB multiplier designs; the delay, area, power and PDP metrics are compared separately.

VI. FUTURESCOPE:

A new modified RBPP generator has been proposed in this paper; this design eliminates the additional ECW that is introduced by previous designs. Therefore, a RBPP accumulation stage is saved due to the elimination of ECW. The new RB partial product generation technique can be applied to any 2_-bit RB multipliers to reduce the number of RBPP rows from __/4 + 1_ to __/4_.Simulation results have shown that the performance of RB MBE multipliers using the proposed RBMPPG-2 is improved significantly in terms of delay and area. The proposed designs achieve significant reductions in area and power consumption when the word length is at least32 bits. The PDP can be reduced by up to 59% using the proposed RB multipliers when compared with existing RB multipliers. Hence, the proposed RBPP generation method is a very useful technique when designing area and PDP efficient power-of-two RB MBE multipliers.

VII. CONCLUSION:

A new modified RBPP generator has been proposed in this paper; this design eliminates the additional ECW that is introduced by previous designs. Therefore, a RBPP accumulation stage is saved due to the elimination of ECW. The new RB partial product generation technique can be applied to any 2_-bit RB multipliers to reduce the number of RBPP rows from __/4 + 1_ to __/4_.Simulation results have shown that the performance of RB MBE multipliers using the proposed RBMPPG-2 is improved significantly in terms of delay and area. The proposed designs achieve significant reductions in area and power consumption when the word length is at least32 bits.

The PDP can be reduced by up to 59% using the proposed RB multipliers when compared with existing RB multipliers. Hence, the proposed RBPP generation method is a very useful technique when designing area and PDP efficient power-of-two RB MBE multipliers.

REFERENCES:

[1]A. Avizienis, ―Signed-digit number representations for fast parallel arithmetic,‖ IRE Trans. Electron. Computers, vol. EC-10,pp. 389–400, 1961.

[2] N. Takagi, H. Yasuura, and S. Yajima, ―High-speed VLSI multiplication algorithm with a redundant binary addition tree,‖IEEE Trans. Computers, vol. C-34, pp. 789-796, 1985.

[3] Y. Harata, Y. Nakamura, H. Nagase, M. Takigawa, and N.Takagi, ―A high speed multiplier using a redundant binaryadder tree,‖ IEEE J. Solid-State Circuits, vol. SC-22, pp. 28-34,1987.

[4] H. Edamatsu, T. Taniguchi, T. Nishiyama, and S. Kuninobu, ―A33 MFLOPS floating point processor using redundant binary representation,‖ in Proc. IEEE Int. Solid-State Circuits Conf.(ISSCC), pp. 152–153, 1988.

[5] H. Makino, Y. Nakase, and H. Shinohara, ―A 8.8-ns 54x54-bitmultiplier using new redundant binary architecture,‖ in Proc.Int. Conf. Comput. Design (ICCD), pp. 202-205,

[6] H. Makino, Y. Nakase, H. Suzuki, H. Morinaka, H. Shinohara,and K. Makino, ―An 8.8-ns 54×54-bit multiplier with highspeed redundant binary architecture,‖ IEEE J. Solid-State Circuits,vol. 31, pp. 773-783, 1996.

[7] Y. Kim, B. Song, J. Grosspietsch, and S. Gillig, ―A carry-free54b×54b multiplier using equivalent bit conversion algorithm,‖IEEE J. Solid-State Circuits, vol. 36, pp. 1538–1545,

[8] Y. He and C. Chang, ―A power-delay efficient hybrid carrylookaheadcarry-select based redundant binary to two‘s complement converter,‖ IEEE Trans. Circuits Syst.I, Reg. Papers, vol.55, pp. 336–346, 2008.