Volume 6, Issue 4, April 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Efficient VLSI Architecture for Modulo $2^n + 1$ Multiplier

Ahana Mishra, Swapnil Jain, Jyoti Dangi
Electronics & Comm. Department, NIIST,
Bhopal, India

Abstract*— Efficient modulo $2^n + 1$ multipliers is proposed. According to our algorithm, the resulting partial products are reduced by an inverted and carry save adder to two operands, which are finally adder by a 2-stage inverted n-bit adder. By using the 2-stage inverted n-bit adder, the new multipliers reduce the number of the partial product to n/2 for even and (n+1)/2 for odd expect for one correction term. The analytical and experimental result indicates that the new modulo $2^n + 1$ multipliers, offer enhanced operation among all the efficient existing solutions.

Keywords*— 2-Stage Inverted n-Bit Adder, Modulo Multiplier, Residue Number System (RNS).

I. INTRODUCTION

Residue number systems (RNS) [1]-[2] reduces the delay of carries propagation, thus suitable for the implementation of high-speed digital signal processing devices. Some arithmetic operations, such as addition and multiplication, can be carried out more efficiently in RNS than in conventional two's complement systems. RNS has been adopted in the design of Digital Signal Processors (DSP) [3]-[4], Finite Impulse Response (FIR) filters [5], image processing units [6], Discrete Cosine Transform (DCT) processors [7], communication components [8], cryptography [9], and other DSP applications [10].

In recent years, efficient schemes for modulo multipliers have been studied intensively [11]-[13]. Generally, modulo 2^n+1 multipliers can be divided into three categories, depending on the type of operands that they accept and output:

i. the result and both inputs use weighted representation;
ii. the result and both inputs use diminished-1 representation;
iii. the result and one input use weighted representation, while the other input uses diminished-1.

For the first category, Zimmermann et al. [11] used Booth encoding to realize, but depart from the diminished-1 arithmetic, which leads to a complex architecture with large area and delay requirements. For the second category, Wang *et al.* [12] proposed diminished-1 multipliers with -bit input operands. The multipliers use a non-Booth recoding and a zero partial-product counting circuit. The main drawback in this architecture was handling of zero inputs and results were not considered.

Curiger et al. [13] proposed new modulo multipliers by using the third category. This architecture use ROM based look-up methods are competitive. The main drawback in this architecture increasing n-bit, they become infeasible due to excessive memory requirements.

Jian et al. [14] also proposed for the third category architecture and reduce the memory requirement and speed up. The new architecture is based on n-bit addition and radix-4 booth algorithm, which is efficient and regular. We are replaced diminished-1 modulo 2^n+1 adder by inverted n-bit adder.

The remainder of the paper is organized as follows: mathematical formulation of Diminished-1 number representation computation of modulo multiplier is presented in Section II. The proposed structures are presented in Section III. Hardware and time complexity of the proposed structures are discussed and compared with the existing structures in Section IV. Conclusion is presented in Section V.

II. DIMINISHED -1 NUMBER REPRESENTATION

The modulo $2^n + 1$ arithmetic operations require (n+1) bit operands. To avoid (n+1)-bit circuits, the diminished-1 number system [15] has been adopted. Let $d[A]$ be the diminished-1 representation of the normal binary number $A \in [0, 2^n$, namely

$$d[A] = |A - 1|_{2^n+1} \quad \text{(i)}$$

In (i), when, $A \neq 0, d[A] \in [0, 2^n - 1$ is an n -bit number, therefore (n+1) -bit circuits can be avoided in this case. However,

$$A = 0, d[A] = d[0] = |-1|_{2^n+1} = 2^n \quad \text{(ii)}$$

is an (n+1) -bit number. This leads to special treatment for d [0]. The diminished-1 arithmetic operations [15] are defined as

$$d[-A] = \overline{d[A]}, if\ d[A] \in [0, 2^n - 1] \quad \text{(iii)}$$
$$d[A + B] = |d[A] + d[B] + 1|_{2^n+1} \quad \text{(iv)}$$
$$d[A - B] = \left|d[A] + \overline{d[B]} + 1\right|_{2^n+1} \quad \text{(v)}$$

© 2016, IJARCSSE All Rights Reserved

$$d[AB] = |d[A] \times d[B] + d[A] + d[B]|_{2^n+1}$$
$$= |d[A] \times B + B - 1|_{2^n+1} \quad \text{(vi)}$$
$$d[2^k, A] = iCLS(d[A], k) \quad \text{(vii)}$$
$$d[-2^k, A] = iCLS(\overline{d[A]}, k) \quad \text{(viii)}$$

where $\overline{d[A]}$ represents the one's complement of d [A]. In (vii) and (viii) iCLS (d[a], k) is the k -bit left-circular shift of in which the bits circulated into the LSB are complemented.

III. PROPOSED ARCHITECTURE

A proposed architecture consists of the partial products generator (PPG), the correction tern generator (CTG), the inverted end-around-carry carry save adder (EAC CSA) and 2-stage inverted n-bit adder. Based on this architecture, a solution which is more effective is proposed.

The encoding scheme accordant with the radix-4 Booth recoding [15], the partial product generator (PPG) can be constructed with the well-known Booth encoder (BE) and Booth selector (BS). The different blocks used in PPG and EAC CSA are taken from [15].

In this paper, we modified BE block which take successive overlapping triplets ($b_{2i+1}b_{2i}b_{2i+1}$) and encodes each as an element of the set {-2,-1, 0, 1 2}. Each BE block produces 3 bits: 1x, 2x and Sign. The 3 bits along with the multiplicand are used to form partial products.

The CTG produces which has the form ($.....0x_{i+1}0x_i.....0x_10x_0$) with $x_i \in \{0,1\}$. Since the 2i-th bit x_i is 1 when the BE_i block encodes 0, otherwise x_i is 0, one XNOR gate accepting the 1x and 2x bits of the block can generate the 2i-th bit x_i .

The inverted EAC CSA tree can reduce the Partial Products to two numbers. The CSA tree is usually constructed with full adders (FA).Then the final two numbers from the tree is passed through the 2 -stage inverted n-bit adder. The 2-stage inverted n-bit adder is consisting of two rows of adders. First row consist of n-bit ripple carry adder of one half adder and (n-1) full adders and the second row consist of n-bit ripple carry adder of n half adders, as shown in fig.(3).

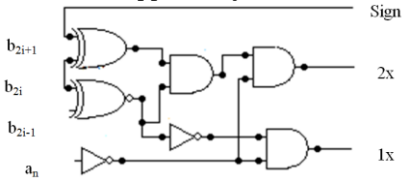


Figure 1: Booth encoder

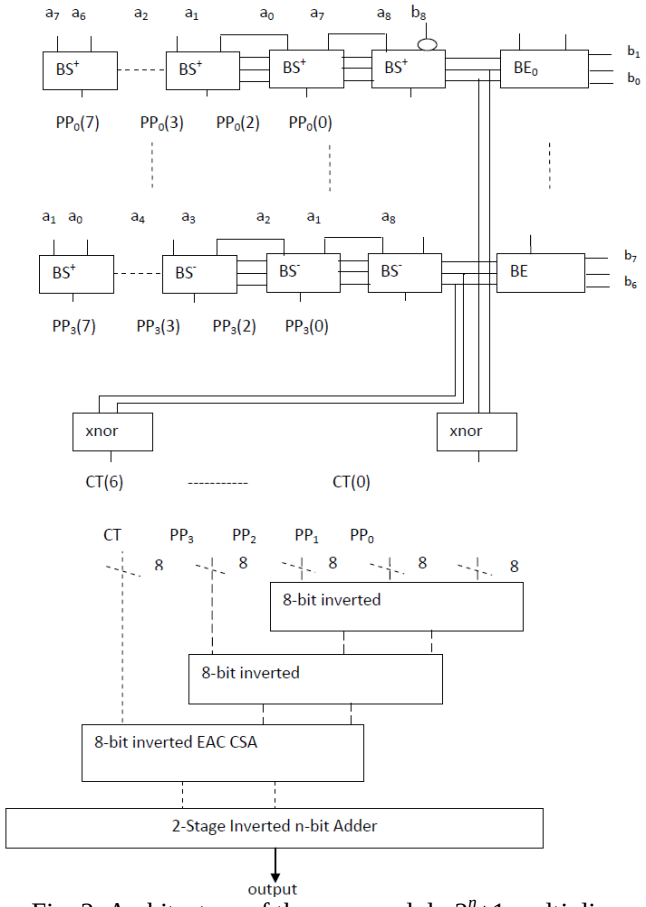


Fig. 2. Architecture of the new modulo 2^n+1 multiplier

© 2016, IJARCSSE All Rights Reserved

The proposed architecture in fig.1 take two input of n bit as d[A] and B and gives result as $P = |A \times B|_{2^n+1}$.Where d[A] is the diminished-1 representation of A.

Input			Output			Code
b_{2i+1}	b_{2i}	b_{2i-1}	Sign	2x	1x	
0	0	0	0	0	0	0
0	0	1	0	0	1	1
0	1	0	0	0	1	1
0	1	1	0	1	0	2
1	0	0	1	1	0	-2
1	0	1	1	0	1	-1
1	1	0	1	0	1	-1
1	1	1	1	0	0	-0

Figure 2(b): Truth table

Example: When, n=8,Let A=$(227)_{10}$, B=$(157)_{10}$, then d[A]=$(226)_{10}$, $|A \times B|_{2^8+1} = (173)_{10}$.

Example

n=8,d[A]=$(11100010)_2$,B=$(10011101)_2$,$a_8=0$,$b_8=0$

Encode Partial Products

$(b_8 \vee (b_7 \oplus b_1))b_0(b_8 \vee b_7)$ --- 011---PP_0---11111111

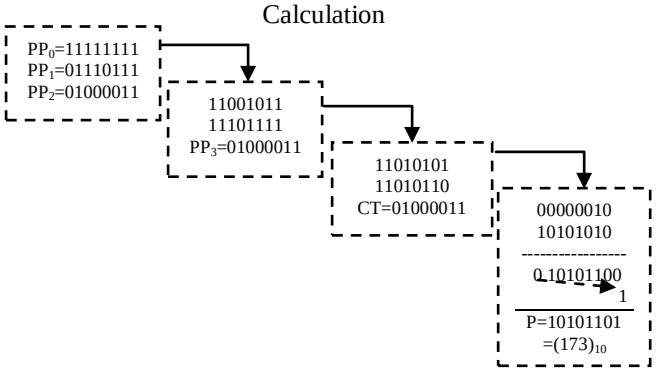
b_3 b_2 $b_1 \cdot \overline{b_7}$ ---110---PP_1---01110111

b_5 b_4 b_3 ---011---PP_2---01000011

b_7 b_6 b_5 ---100---PP_3---11110001

CT=00000001

Calculation



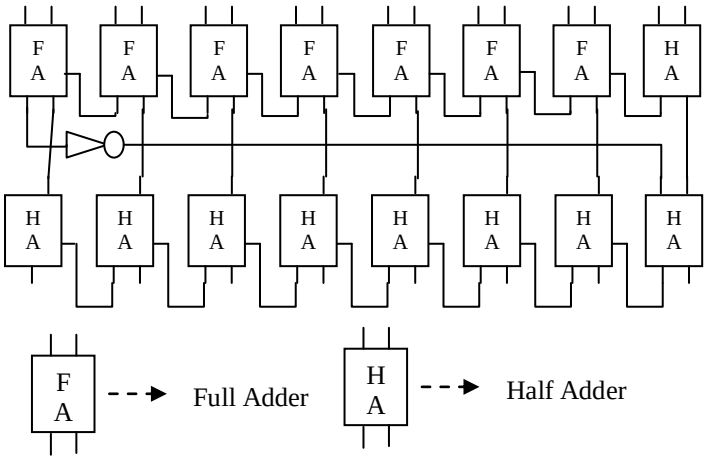


Fig.3. 2-Stage Inverted n-bit Adder

IV. RESULT AND SIMULATION

The proposed architecture has very low hardware complexity compared to [15], which consist of modulo $2^n + 1$ adder. In the proposed architecture, we use the 2-stage inverted n-bit adder. And calculate the output for 8, 12bit.

© 2016, IJARCSSE All Rights Reserved

For a more realistic comparison, we implemented modulo multipliers for the new, [11]–[14] and [15]. At first, we used VHDL language to generate hardware models for the new and the multipliers in [11]–[14] and [15] with operand sizes of 8 and 12bits.

Comparison of Synopsys result in the proposed architecture and diminished-1 modulo $2^n + 1$ architecture is given in Table 1 respectively.

These improvements are reasonable. When compared with Diminished-1 modulo $2^n + 1$ multipliers for weighted representation; the blocks of the new multipliers are based on inverted n-bit adder architecture and use efficient inverted n-bit adders.

Table 1: Result for 8 bit

Multiplier	Jian *et al*[12] 8-bit	Jian *et al*[12] 12-bit
No. of slices	2 out of 192	22 out of 192
No. Of 4 input LUTS	2 out of 384	39 out of 384
Delay	8.23 ns	19.80ns

V. CONCLUSION

In this paper, we proposed efficient architecture for modulo $2^n + 1$ multiplier. This architecture uses inverted n-bit adder Booth recoding and reduces the number of the partial products to n/2 for even and (n+1)/2 for odd, this is the least number of the partial products among all modulo multipliers published. The reduction scheme uses the well-known inverted EAC CSA tree and the final 2-stage inverted n-bit adder generates the result. The circuit to handle the zero-input case is merged into the first Booth encoder and there is no extra delay to be added. The new multipliers, compared to existing implementations, offer better power while being more compact and their regular structure allows efficient VLSI implementations.

REFERENCES

[1] P. V. Ananda Mohan, *Residue Number Systems: Algorithms and Architectures*, Kluwer, Academic Publishers, 2002.

[2] Omondi, and B.Premkumar,Residue Number System:Theory and Implementation,*Imperial College Press*,2007

[3] R. Chaves, L. Sousa, "RDSP: a RISC DSP based residue number system", *in Proc. Euromicro Symposium on Digital System Design (DSD)*, pp. 128–135, Sept. 2003.

[4] J. Ramirez, A. Garcia, S. Lopez-Buedo, and A. Lloris, "RNS-enabled digital signal processor design", *Electronics Letters*, vol. 38, no. 6, pp. 266–268, March 2002.

[5] U. Meyer-Baese,A. Garcia, and F. Taylor, "Implementation of a communications channelizer using FPGAs and RNS arithmetic", *Journal of VLSI Signal Processing*, vol. 28, no. 1-2, pp. 115-128, June 2001.

[6] J. C. Bajard, and L. Imbert, "A full RNS implementation of RSA", *IEEE Trans. Comput*., vol. 53, no 6, pp. 769–774, June 2004.

[7] Y. Liu, and E.M.-K Lai, "Design and implementation of an RNSbased 2-D DWT processor", *IEEE Trans. on Consumer Electronics*, vol. 50, no. 1, pp. 376-385, Feb. 2004.

[8] R. Zimmermann,"Efficient VLSI implementation of modulo $(2^n \pm 1)$ addition and multiplication," in *Proc. 14th IEEE Symp. Comput. Arithm.*, Adelaide, Australia, Apr. 1999, pp. 158–167.

[9] Z.Wang, G. A. Jullien, andW. C.Miller, "An efficient tree architecture for modulo multiplication," *J. VLSI Signal Process. Syst.*, vol.14, no. 3, pp. 241–248, Dec. 1996.

[10] L. Leibowitz,"A simplified binary arithmetic for the fermat number transform," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-24, pp. 356–359, May 1976.

[11] J.W.Chen, R.H.Yao and W.J.Wu,Efficient "modulo $(2^n + 1)$ multipliers," *IEEE Trans. VLSI systems*., vol. 19, no 12, pp. 2149–2157, Dec. 2011.

[12] Yuuki Tanaka and Shugang Wei, "An Efficient Diminished-1 Modulo 2n+1 Multiplier Using Signed-Digit Number Representation", 978-1-4799-8641-5/15/$31.00 c 2015 IEEE.

© 2016, IJARCSSE All Rights Reserved