2581-4575

International Journal For Recent Developments in Science & Technology

Crossref

A Peer Reviewed Research Journal

STATE MACHINE AND ITS DESIGN MODEL FOR MODULO ADDERS

[1]KAMADI DURGAPRASAD, [2] B.V.V.S.R.K.K.PAVAN M.Tech,(Ph.D.)

[1]M.TECH VLSI, DEPT OF E.C.E, KAKINADA INSTITUTE OF ENGINEERING AND TECHNOLOGY, KORANGI, ANDHRAPRADESH, INDIA, 533461

[2]ASSOCIATE PROFESSOR, KAKINADA INSTITUTE OF ENGINEERING AND TECHNOLOGY, KORANGI, ANDHRAPRADESH, INDIA, 533461

ABSTRACT:

Reversible logic is a computing worldview that has pulled in noteworthy consideration as of late because of its properties that lead to ultra-low power and dependable circuits. Reversible circuits are key, for instance, for quantum computing. Since expansion is a central task, structuring productive adders is a foundation in the examination of reversible circuits. Residue Number Systems (RNS) has been as an incredible asset to give parallel and flaw tolerant executions of calculations where augmentations and duplications are overwhelming. Modular adders are very crucial components in the performance of residue number system-based applications. Most of the work published so far has been restricted to modulo (2n± 1) adders or modulo-specific adders. Less work has been dedicated to modulo-generic adders. This work presents new designs for modulo (2n±K) adders, where K is any integer in the range of $3 \leq K < 2^{n-1}$. The proposed structure merges two binary adder structures and maximizes sharing of components, wherever possible. This merger permits shorter cell-interconnections, which results in space wastage reduction. Additionally, tristate-based multiplexers (MUXs) are used in lieu of the more demanding gate-based MUX's.In our design model, we propose the mix of RNS and reversible logic. The parallelism of RNS is utilized to build the execution of reversible computational circuits ensuring the correct design for power optimization.

INTRODUCTION:

The residue number system (RNS) is a non-weighted representation. This representation is based on expressing any number using relatively prime positive integers, known as moduli. The product of all moduli defines the dynamic range in which numbers within this range are uniquely represented [1]. When it comes to additions, subtraction and multiplication, each modulus is independent from other moduli in processing its computations [1].The data path in which computations take place with respect to any modulus is referred to as a channel. For an RNS representation that uses L moduli, there are L parallel channels. The relatively independent and parallel channels reduce considerably the time needed for all computationally demanding arithmetic applications that depend mainly on the above listed operations. When a certain dynamic range is specified, the level of parallelism achieved is higher if the number of moduli L is increased. This implies dividing the total number of bits of a dynamic range over more channels, thus, having shorter word-lengths for each channel, where a shorter word-length can be processed faster than a longer one [1]. The level of parallelism achieved using three-moduli sets proved to be reasonable for some digital signal processing (DSP) applications. Other DSP and cryptography applications require higher levels that cannot be attained using three-moduli sets [1–6].When considering the number of moduli, three and four moduli sets have been heavily researched [1, 7–11]. To achieve higher levels of parallelism, five-moduli sets have been introduced such as{22n, 2n− 3, 2n+ 3, 2n− 1, 2n+ 1}, {2n, 2n− 1, 2n+ 1, 2n+ 1−1,2n− 1− 1}, and {2n, 2n− 1, 2n+ 1, 2n−2(n+ 1/2)+ 1, 2n+ 2(n+ 1/2)+1} [12–14].An application-specific RNS-based processor uses mainly adders and multipliers. The area, time, and power needed to perform modular additions are very critical factors in applications that use RNS. A modular multiplier is basically seen as a process of successive binary and modular additions [1, 15, 16]. Examining the form of different moduli introduced in three- four-, and five-moduli sets reveals that most of these moduli are of the form(2n± 1) [13], (2n± 3) [12], (2n± 2(n+ 1/2)+ 1) [13, 14] or more general forms such as (2n±K), where 3 ≤K< 2n− 1 [1]. Most of the published research in designing RNS-based modular adders has-been dedicated to the moduli of the form 2n and (2n± 1) [17–23].Significant improvements have been achieved in the proposed designs, where the delay requirements of modulo (2n± 1) adders were getting closer to the delay of a modulo 2n binary adders. Such an advancement is crucial in improving the overall performance of any RNS-based processor. On the other hand, very limited work has been dedicated to general modular adders [24–32], or modulo-specific adders, other than modulo (2n± 1) [33–35].This work is intended to propose modulo (2n±K) adders that can serve any moduli, where any moduli can be expressed using one of the two given forms. The intention is to reduce the area and time requirements of such adders to approach those of modulo (2n± 1). Any improvement in this direction would make higher-order moduli sets, such as{22n, 2n− 1, 2n+ 1, 2n−2(n+ 1/2)+ 1, 2n+ 2(n+ 1/2)+ 1}, more appealing in DSP and cryptographic applications that require high-level of parallelism.

REVERSIBLE GATES

Reversible circuits give a coordinated connection among data sources and yields; accordingly, information sources can be recuperated from yields. This fascinating component results in huge power saving in advanced circuits [20]. Established advanced entryways are not reversible, reversible doors ought to be planned as fundamental segments to structure coherent reversible circuits. Understood reversible doors are Feynman, Peres and HNG [20, 21]. The square charts of these entryways are introduced in Fig. 1

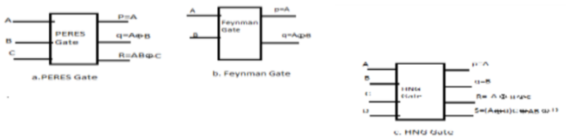

Figure 1: Representing Reversible gates

MODULO ADDER CONFIGURATION UTILIZINGREVERSIBLE CIRCUITS

This segment exhibits the reversible usage of three particular viper structures that are much of the time connected to RNS.A. The CSA with EAC the CSA is a 3-to-2 pressure unit that is extremely mainstream for customary number arithmetic just as in RNS designs because of its speed and cost. ACSA can be worked by utilizing n FAs for including three n-bit operands. As per [21], the HNG reversible door can be utilized to understand a FA by setting the fourth contribution of HNG to the zero-rationale level, as appeared in Fig. 2

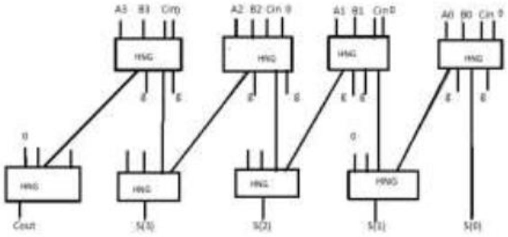

Figure 2: The RCA-based Modulo Adder

The RCA with EAC for modulo 2n -1 expansion of two n-bit numbers, requires n FAs and n HAs in the first and second dimensions, individually. Like CSA, FAs can be acknowledged with HNG entryways. Furthermore, the Peres reversible door can be utilized to actualize a HA, where the third info bit is set to zero,thelast quantum cost of the RCA with EAC for two n-bit operands is 6n+4n=10n, since the individual quantum cost and profundity of a Peres entryway is 4. In addition, the all-outquantum profundity of the RCA with EAC is ((3×(n-1)+4+(3×(n1)+5))×Δ. Besides, the all-outconsistent sources of info and rubbish yields are 2n and 3n, individually, since one of the contributions of HNG and Peres entryways is zero, and furthermore two and one yields of HNG and Peres doors, separately, are not utilized.

PROPSOED MODULO ADDITION:

1. DESIGN PROCEDURE:

An RNS-based modulo (2n−K) adder is defined as

S=A+Bm1

where m1= (2n−K), 3 ≤K≤ 2n− 1−1.Equation can be rewritten as

$$S = \begin{cases} A + B & \text{if } A + B < m_1 \\ A + B - (2^n - K) & \text{if } A + B \geq m_1 \end{cases}$$

Observing in both cases of the last equation that S<m1< 2n, then applying modulus 2n to both cases of (8) results in

$$S = \begin{cases} \langle A + B \rangle_{2^n} & \text{if } c_{\text{out}} = 0 \\ \langle A + B + K \rangle_{2^n} & \text{if } c_{\text{out}} = 1 \end{cases}$$

Where cout is the output carry resulting from computing (A+B+K). It should be observed that applying modulus 2n to anyone-negative integer implies considering just the least significant nbits of the binary representation of the integer. The basic concept used in this paper for evaluating S in (9) is to compute, simultaneously, both output cases (i.e. A+B andA+B+K), determine cout and select one of the two outputs viatristate-based MUXs.The output of the first case of (9) (i.e. ⟨A+B⟩2n) is simply performed by a structure similar to Fig. 2. However, in order to compute the second case of (9), the three stages of the adder of Fig. 2 (namely preprocessing stage, parallel-prefix stage, and the sum computation stage) need to be modified as follows:

i. Preprocessing stage: The binary representations of A, B and K

$$A \rightarrow \overset{n}{\overline{a_{n-1}a_{n-2}, \ldots, a_1a_0}}$$

$$+B \rightarrow \overset{n}{\overline{b_{n-1}b_{n-2}, \ldots, b_1b_0}}$$

$$+\widetilde{K} \rightarrow \overset{n}{\overline{0\ k_{n-2}, \ldots, k_1k_0}}$$

$$A' \rightarrow \quad a'_{n-1}, \ldots, a'_1a'_0$$

$$+B' \rightarrow b'_nb'_{n-1}, \ldots, b'_2b'_10$$

ii.

$$a'_i = \begin{cases} a_i \oplus b_i & \text{if } k_i = 0 \\ a_i \odot b_i & \text{if } k_i = 1 \end{cases}$$

iii.

$$b'_{i+1} = \begin{cases} a_i \wedge b_i & \text{if } k_i = 0 \\ a_i \vee b_i & \text{if } k_i = 1 \end{cases}$$

where ⊙ refers to an exclusive NOR logic operator. Adding ai to biis performed using a half-adder circuit if ki= 0. However, addingai, bi, and ki is performed using a pseudo half-adder circuit if ki= 1.The half-adder and pseudo half-adder circuits are shown in Fig.

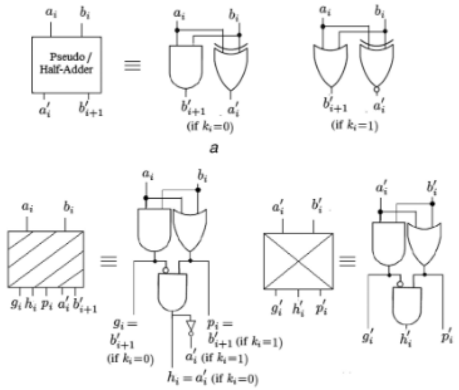

Figure3: Pseudo Half adder circuits

FSM MODEL FOR MODULO ADDITION

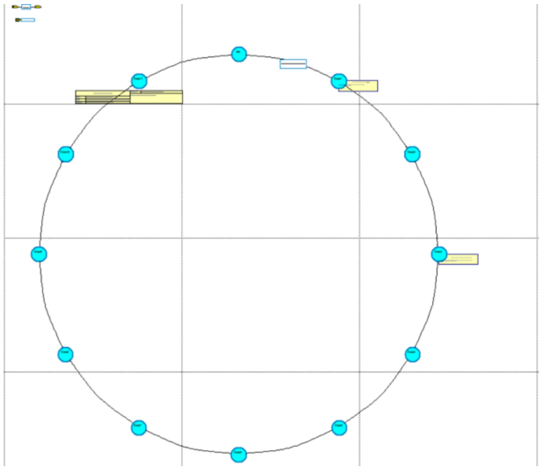

Figure 4: FSM control model for modulo addition.

Our design aims to full fill the design formulation as mentioned above using FSM based circuit. This design would improvise different scenario of the states of the addition for each input provision.

The input data provision with the FSM would improvise 12 states to complete the 32 bit adder and 16 bit adder.

RESULTS AND DISCUSSION:

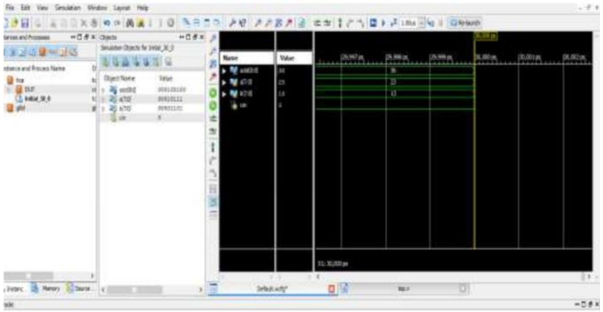

Figure 5: Representing initial phase of 32 bit modulo additions using 8 bit adder

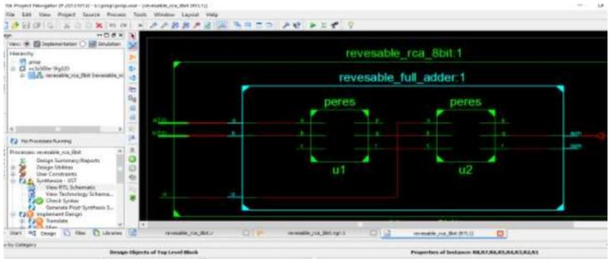

Figure 6: Representing reversible adder gates

Figure 7: Representing the final output for modulo addition using 8 bit to perform 16 bit additions

COMAPRISIONS TABLE:

SN0	PARAMETERS	Existing Modulo adder	PROPOSED Modulo FSM adder
1	AREA	9.3 %	4.17%
2	POWER	8.856W	5.03W
3	DELAY	102.87 ns	94.58 ns
5	Latency	106.85 ns	95 ns

CONCLUSIONS:

The potential of RNS in DSP and cryptographic applications can be highly enhanced by improving the performance of modular adders. This paper presented new designs for modulo-generic modular adders. The moduli can take any form of (2n±K),3 ≤K≤ 2n− 1− 1. Using VLSI tools, the proposed structures proved to be significantly more efficient than other functionally identical modular adders, where no restriction on the form of moduli are imposed. The new structures enabled improving the level of parallelism needed by many RNS-based applications and the overall computational speed.

REFERENCES:

[1]Mohan, P.V.A.: 'Residue number systems: theory and applications'(Birkhauser, Basel, Switzerland, 2016)

[2]Hiasat, A., Abdel-Aty-Zohdy, H.: 'Design and implementation of a fast andcompact residue-based semi-custom VLSI arithmetic chip'. Proc. 37thMidwest Symp. Circuits Systems, August 1994, pp. 428–431

[3]Toivonen, T., Heikkil, J.: 'Video filtering with fermat number theoretictransforms using residue number system', IEEE Trans. Circuits Syst. VideoTechnology, 2006, 16, (1), pp. 92–101

[4]Vun, C., Premkumar, A., Zhang, W.: 'A new RNS based DA approach forinner product computation', IEEE Trans. Circ. Syst. CAS-I, 2013, 60, (9), pp.2139–2152

[5]Esmaeildoust, M., Schinianakis, D., Javashi, H., et al.: 'Efficient RNSimplementation of elliptic curve point multiplication over GF(p)', IEEETrans. Very Large Scale Integr. Syst., 2013, 21, (8), pp. 1545–1549

[6]Zheng, X., Wang, B., Zhou, C., et al.: ‘Parallel DNA arithmetic operationwith one error detection based on 3-moduli set’, IEEE Trans. Nanosci., 2016,15, (5), pp. 499–507

[7]Hiasat, A.: ‘New designs for a sign detector and a residue to binaryconvertor’, IEE Proc. G Circ. Dev. Syst., 1993, 140, (4), pp. 247–252

[8]Wang, W., Swamy, M., Ahmad, M., et al.: ‘A study of the residue-to-binaryconverters for the three-moduli sets’, IEEE Trans. Circuits Syst. I, 2003, 50,(2), pp. 235–243

[9]Hiasat, A.: ‘An efficient reverse converter for the three-moduli set (2n+1−1,2n, 2n−1)’, IEEE Trans. CAS-II, 2017, 64, (8), pp. 962–966

[10]Molahosseini, A., Navi, K., Dadkhah, C., et al.: ‘Efficient reverse converterdesigns for the new 4-moduli sets (2n−1, 2n, 2n, +1, 22n+1 − 1) and (2n−1,22n, 2n, +1, 22n +1) based on new CRTs’, IEEE Trans. Circuits Syst. I, 2010,57, (4), pp. 823–835

[11]Hiasat, A.: ‘A residue-to-binary converter for the extended four-moduli set{2n−1, 2n + 1, 22n + 1, 22n+p}’, IEEE Trans. Very Large Scale Integ. Syst.,2017, 25, (7), pp. 2188–2192

[12]Ahmadifar, H., Jaberipur, G.: ‘A new residue number system with 5-moduliset: (22q, 2q ± 3, 2q ± 1)’, Comp. J., 2015, 58, (7), pp. 1548–1565

[13]Pettenghi, H., Chaves, R., Sousa, L.: ‘RNS reverse converters for moduli setswith dynamic ranges up to (8n + 1)-bits’, IEEE Trans. Circuits Syst. I, 2013,60, (6), pp. 1487–1500

[14]Hiasat, A.: ‘A reverse converter and sign detectors for an extended RNS fivemoduli se’, IEEE Trans. Circ. Syst. TCAS-I, 2017, 64, (1), pp. 111–121

[15]Hiasat, A.: ‘A suggestion for a fast residue multiplier for a family of moduliof the form (2n−(2p ± 1))‘, Comp. J., 2004, 47, (1), pp. 93–102

[16]Pettenghi, H., Cotofana, S., Sousa, L.: ‘Efficient method for designingmodulo (2n ± k) multipliers’, J. Circ. Syst. Comp., 2014, 23, (1), pp. 1–20

[17]Zimmermann, R.: ‘Efficient VLSI implementation of modulo (2n ± 1)addition and multiplication’. Proc. 14th IEEE Symp. ComputationalArithmetic, April 1999, pp. 158–167

[18]Kalamboukas, L., Nikolos, D., Efstathiou, C., et al.: ‘High-speed parallel-prefix modulo 2n − 1 adders’, IEEE Trans. Comput., 2000, 49, (7), pp. 673–680

[19]Vergos, H.T., Efstathiou, C., Nikolos, D.: ‘Diminished-one modulo 2n+ 1adder design’, IEEE Trans. Comput., 2002, 51, (12), pp. 1389–1399

[20]Dimitrakopoulos, G., Nikolos, D.: ‘High-speed parallel-prefix VLSI lingadders’, IEEE Trans. Comput., 2005, 54, (2), pp. 225–231

[21]Patel, R.A., Benaissa, M., Boussakta, S.: ‘Fast parallel-prefix architectures formodulo 2n − 1 addition with a single representation of zero’, IEEE Trans.Comput., 2007, 56, (11), pp. 1484–149

2[22]Jaberipur, G., Parhami, B.: ‘Unified approach to the design of modulo- 2n ± 1adders based on signed-LSB representation of residues'. Proc. 19th IEEESymp. Computer Arithmetic, April 2009, pp. 57–64

[23]Vergos, H.T., Dimitrakopoulos, G.: 'On modulo 2n + 1 adder design', IEEETrans. Comput., 2012, 61, (2), pp. 173–186