An Improved RNS Reverse Converter in Three-Moduli Set

Navid Habibi*, Mohammad Reza Salehnamadi

Department of Computer Engineering, Islamic Azad University, South Tehran Branch, Tehran, Iran

Received 29 February 2016; accepted 10 April 2016

Abstract

Residue Number System (RNS) is a carry-free and non-weighed integer system. In this paper an improved three-moduli set $\{2^n-1, 2^n+1, 2^{pn}-1\}$ in reverse converter based on CRT algorithm is proposed. CRT algorithm can perform a better delay and hardware implementation in modules via other algorithms. This moduli is based on p that covers a wide range on modules and supports the whole range of its modules in dynamic range. With growth in moduli, many types of modules have been proposed. By using dynamic range we can solve many problems in Residue Number System (RNS) by just one three moduli. In proposed moduli set of this paper in Residue Number System (RNS), the internal circuit is improved and thus, complexity of circuit, energy consumption and power consumption in our proposed design is improved. These improvements are shown in evaluation in terms of CSA adders, CPA adders and delay.

Keywords: Residue Number System, Reverse Converter, Moduli Set.

1. Introduction

The RNS is integer number systems that by using the property of a carry free operations is none-weighted [1, 2].

RNS has many advantages and can be used in different filtering and cryptographies such as 1-D filtering, FIR filtering and RISK DSP and Image Processing [3]. RNS has fast calculations, so the operations like conversions can be with less latency [4].

Residue to binary in RNS system has different methods. One of its primary methods is LUT (Look Up Tables). By using ROM memories here, modules in power of two can be helpful [3]. Other methods are CRT and mixed radix conversion (MRC). These methods use their realization used carry-save adders (CSA) in CSA tree structure and carry-propagate adder (CPA) in their implementations [3].

Many modules sets have been introduced such as $\{2^n-1, 2^n, 2^n+1\}$ and $\{2^n-1, 2^n, 2^n-1-1\}$These are $\{2^n-1, 2n, 2^n+1, 2^{2n}+1\}$, $\{2^n-1, 2^n, 2^n+1, 2^{n+1}-1\}$, $\{2^n-1, 2^n, 2^n+1, 2^n+1+1\}$ $\{2^n-3, 2^n+1, 2^n-1, 2^n+3\}$ and $\{2^n-1, 2^n, 2^n+1, 2^{n-1}-1\}$ [5, 6, 7, 8, 9]. The reverse converter for these moduli set has very high latency and hardware cost because of inefficient multiplicative inverses some modules are unbalanced like moduli sets $\{2^n, 2^{n/2}-1, 2^{n/2}+1, 2^n+1, 2^{2n-1}-1\}$ [1]. In [11], an effective reverse converter in three moduli set {2n-1, 2n+1, 2pn+1-1} is proposed that has a wide range with two parameters.

In this paper, a new reverse converter based on MRC method for the moduli set $\{\mathbf{2^n-1, 2^n+1, 2^{pn}-1}\}$ is proposed. This new moduli has better latency

* Corresponding author. Email: St_n_habibi@azad.ac.ir

and hardware cost than others. In section two MRC method is described. In section three proposed reverse converter is implemented and in the last part a comparison between modules is presented.

2. Background

RNS give us some good advantages, these advantages are in complexity and hardware delay that can help us to have a better arithmetic operation and is used in cameras, Digital signal processors and many other systems. Besides this RNS also have some disadvantages. Using signed numbers and detection of overflow that could be by parity checking and this is a problem in RNS in division forms [11].

Figure 1 has shown Arithmetic operation in RNS. At first conventional binary form is presented. In the second part we should convert binary form to RNS Form which are residue numbers instead of binary form. After that we would have residue number system form and we can have our calculation in this form. At the end we should have reverse converter to reverse it in to binary form again. In here we presented a residue number in reverse converter.

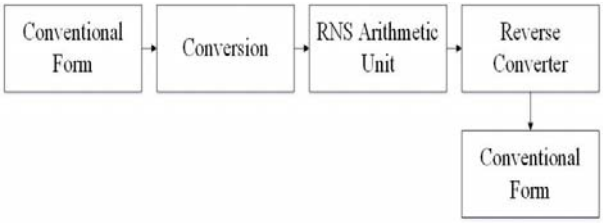

Fig. 1. RNS Construction

Most converters in RNS are usually based on two algorithm; CRT and MRC [3].Given a moduli set $\{m_i\}_{i=1,k}$, the residues (x_1, x_2, ..., x_k) can be converted into the corresponding decimal number X in the following ways: First, by the use of the well-known CRT, which is given as [10]:

$$X = |\sum_{i=1}^{k} m_i |M_i^{-1}|_{m_i} x_i|_M \quad (1)$$

Where $M = \prod_{i=1}^{k} m_i$, $M_i = \frac{M}{m_i}$ and M_i^{-1} is the multiplicative inverse of M_i with respect to m_i.

The MRC can also be used. Suppose we have a residue number representation (x_1, x_2, ..., x_k) with respect to the moduli set $\{m_i\}_{i=1,k}$ and Mixed Radix Digits (MRDs), $\{a_i\}_{i=1,k}$, the decimal equivalent of the residues can be computed as follows [3]:

$$X = a_1 + a_2 m_1 + a_3 m_1 m_2 + \cdots + a_n m_1 m_2 \ldots m_{k-1} \quad (2)$$

Where the MRDs are given as (10)

$$\boldsymbol{a_1 = x_1}$$

$$\boldsymbol{a_2 = |(x_2 - a_1)|m_1^{-1}|_{m_2}|_{m_2}}$$

Up to

$$a_k = |(((x_k - a_1)|m_1^{-1}|_{m_k} - a_2)|m_2^{-1}|_{m_k} - \cdots |(-a_{k-1})|m_{k-1}^{-1}|_{m_k}|_{m_k} \quad (3)$$

Given the moduli set $\{m_1, m_2, m_3\}$ with $m_1 = 2n$, $m_2 = 2n+1$, and $m_3 = 2n - 1$, the decimal equivalent of the residue numbers (x_1, x_2, x_3) is computed as (10)

$$A = m_1 |\frac{A}{m_1}| + x_1 \quad (4)$$

Implementation of the proposed moduli set with this algorithm is presented in next section.

3. Reverse Converter's Design

A residue number system in three moduli set {2n-1, 2n+1, 2pn+1-1} is proposed in [11] that its p is even in it. A Reverse Converter with CRT method is presented in [11]. With this moduli, it could achieve a better Speed and delay in its hardware design. It this article, we have improved this moduli and we could get a better speed and delay. This Improved Three moduli set is calculated in CRT algorithm and its hardware is presented. A comparison between ref [11] and our propose moduli is given in evaluation which shows a better hardware delay and a better speed in Revers Converter. CRT algorithm in moduli set $\{\boldsymbol{2^n - 1, 2^n + 1, 2^{pn} - 1}\}$ is given bellow.

Theorem: $\{\mathbf{2^n-1, 2^n+1, 2^{pn}-1}\}$ are Prime numbers.

Proof: based on the theorem on gcd we have the following equation that represents the common biggest divisor between numbers. In this, if the answer is equal to one, numbers are prime together. The equation is gcd(a,b) = (b,a mod b) in great common divisor.

At first for these residue numbers we have bellow equation.

$$\gcd(2^{pn}+1-1, 2^n-1) = \gcd(2n-1,1) = 1 \quad (5)$$

$$\gcd(2^{pn}+1-1, 2^n+1) = \gcd(2n+1,1) = 1 \quad (6)$$

$$\gcd(2n+1-1, 2n-1) = \gcd(2n+1,1) = 1 \quad (7)$$

So we can proof that these modules are prime to each other.

Based on this algorithm, we consume$\{\boldsymbol{x_1, x_2}$, …,$\boldsymbol{x_n}\}$ as residue numbers and $\{\boldsymbol{m_1, m_2}$, …, $\boldsymbol{m_n}\}$ as a moduli set. The CRT algorithm is as follow:

$$X = |\sum^{L} |X_i N_i|\, M_i|_M \quad (8)$$

$$X = \prod_{i=1}^{L} m_i \quad (9)$$

$M_i = \frac{M}{m_i}$ *and* $N_i = |M_i{}^{-1}|_{p_i}$in this is inverse multiplication of M for modules from i=1,…,n

For this we have:

$$X = |X_1 m_2 m_3 M_1{}^{-1} + x_2 m_1 m_3 M_2{}^{-1} + x_3 m_1 m_2 m_3{}^{-1}|_M \quad (10)$$

Since

$$\begin{aligned} m_1 m_2 m_3{}^{-1} &= (2^n-1)(2^n+1)(-2^{(p-2)n} - 2^{(p-4)n} - \cdots \\ &\quad - 2^{2n} - 2^0) \\ &= (2^{2n}-1)\left(-2^{(p-2)n} - 2^{(p-4)n} - \cdots - 2^{2n} - 2^0\right) \\ &= (-2^{pn}) + 1 = -m_3 + 1 \end{aligned} \quad (11)$$

By placing above formula in to the main formula we would have

$$X = |X_1 m_2 m_3 M_1{}^{-1} + x_2 m_1 m_3 M_2{}^{-1} + x_3(-m_3+1)|_M \quad (12)$$

According to our main formula we will have formula as follow:

$$\begin{aligned} &= \left[\tfrac{X}{m_3}\right] m_3 + x_3 \Rightarrow \left[\tfrac{X}{m_3}\right] = \tfrac{x-x_3}{m_3} \\ &\Rightarrow \left[\frac{X}{m_3}\right] = |X_1 m_2 M_1{}^{-1} + x_2 m_1 M_2{}^{-1} + x_3|_{m_1 m_2} \\ &\Rightarrow |2^{n-1}(x_1 x_1) + 2^{n-1}(2^n x_2 - x_2) - x_3|_{m_1 m_2} \\ &\qquad = |S_1 + S_2 + S_3 + S_3|_{2^{2n}-1} \end{aligned} \quad (13)$$

The residue that we have presented can be shown in binary form. Therefore we would have:

$$X_1 = (X_{1,n-1} X_{1,n-2} \dots X_{1,0}) \quad (14)$$

$$X_2 = (X_{2,n} X_{2,n-1} \dots X_{2,0}) \quad (15)$$

$$X_3 = (X_{3,pn-1} X_{3,pn-2} \dots X_{3,0}) \quad (16)$$

By separating Ss in different forms, four parts would have be created that are presented bellow:

$$S_1 = |2^{n-1}(x_1 x_1)|_{2^{2n}-1} = (X_{1,0} X_{1,n-1} X_{1,n-2} \dots X_{1,0} X_{1,n-1} X_{1,n-2} \dots X_{1,1}) \quad (17)$$

$$S_2 = |2^{n-1}(2^n x_2)|_{2^{2n}-1} = (X_{2,0} 0 \dots 0 X_{2,n} \dots X_{2,1}) \quad (18)$$
(Containing n-1 serial zero number)

$$S_3 = |2^{n-1}(-x_2)|_{2^{2n}-1} = (X_{2,n}{}' X_{2,n-1}{}' \dots X_{2,0}{}' 1 \dots 1) \quad (19)$$
(Containing n-1 serial one number)

$$S_4 = |-x_3|_{2^{2n}-1} = |-2^{pn}(X_{3,pn-1} \dots X_{3,(n-2)n}) - \cdots - -2^{2n}(X_{3,4n-1} \dots X_{3,2n}) - -2^{0n}(X_{3,2n-1} \dots X_{3,0})|_{2^{2n}-1} \quad (20)$$

And for Vs we have:

$$S_4 = S_{4,1} + S_{4,2} + \cdots + S_{4,(\frac{p}{2})} \quad (21)$$

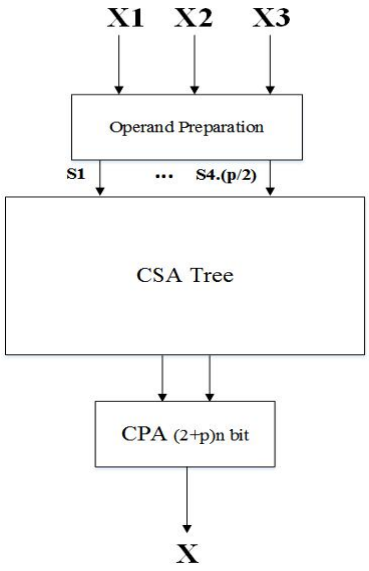

Fig. 2. Reverse Converter for proposed moduli

4. Evaluation

Hardware architecture of our presented residue base on presented formulas in section V is shown in figure 2In this architecture a CSA tree is used that is composed of CSA adders. These adders (p/2) bit CSA modules is in the CSA tree. Levels of this tree is different is number of inputs. The regular CPA adder in this tree is ((2+p)n) bit and is used to calculate a separate result of CSA tree. The delay of this tree is equal to $((6+p)n+(p/2))T_{fa}$. This delay is when the delay of full adder is equal to the delay of carry save adder. The proposed moduli set has less level of CSA tree. So it's shown that delay of the tree and number of adder are also optimized. These differences are shown that this moduli set is better in hardware delay and number of adder and gates which are used in its design. This proposed residue number is shown in figure 3A comparison of this module set with best known moduli implementationis shown in table 1. Table 2 shows proposed moduli in specific p=2 and its comparison to best known moduli implementation. It has shown that delay has been decreased. Hardware complexity of proposed module set is one more than half of other module which is remarkable.

Table 1

Hardware and Delay comparison in different modulus

Revers converter	residue	CSA tree	CPA adder	Complexity	Delay
Proposed Reverse Converter	$\{2^n-1, 2^n+1, 2^{pn}-1\}$	(p/2)	((2+p)n)	(p/2)+ ((2+p)n)	$((6+p)n+(p/2))T_{fa}$
[11], Mehdi Hoseinzadeh, Keihaneh Kia	$\{2^n-1, 2^n+1, 2^{pn+1}-1\}$	(p/2)+1	((2+p)n)+1	(p/2)+((2+p)n)+2	$((6+p)n+((p/2)+2))T_{fa}$

Table 2

Hardware and Delay comparison in moduli form (p=2)

Revers converter	residue	Complexity	Delay
Proposed Reverse Converter (p=2)	$\{2^n-1, 2^n+1, 2^{2n}-1\}$	1+ 4n	$(8n)T_{fa}$
[12], A. Hariri, R. Rastegar, K. Navi	$\{2^n, 2^{2n}-1, 2^{2n+1}+1\}$	4n bit(CSA)+4n bit (adder)	$t_{CLA(4n)}+t_{NOT}+ t_{FA}$

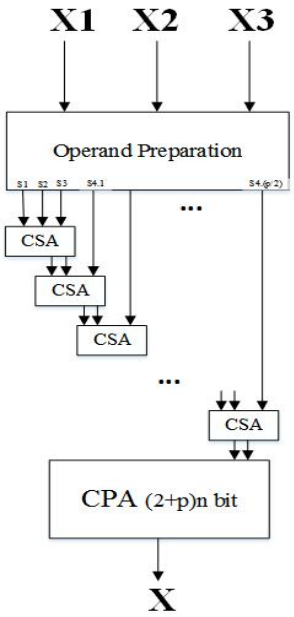

Fig. 3. Reverse Converter with CSA tree for proposed moduli

5. Conclusion

In this paper, we presented a new three-moduli set $\{\mathbf{2^n-1, 2^n+1, 2^{pn}-1}\}$ for p in reverse converter based on CRT algorithm. The proposed moduli deploys carry save adder (CSA) and carry propagate adder (CPA).This new moduli leads us to a better internal hardware implementation. The proposed moduli set has one level of designs less, Therefore In comparison to other moduli sets architectures, we could get better performance and less hardware designs which lead us to a better energy consumption and less power consumption in our new design. Which shows enhancement in both complexity and hardware delay**.**

References

[1] M.R. Taheri, A.R. Pirhoseinlo, M. Esmaeildoust, Mo. Esmaeildoust, K. Navi, "High Speed Reverse Converter for High Dynamic Range Moduli Set", International Journal of Advances in Engineering & Technology, vol.3(2), pp. 26-37, 2012.

[2] P.V. Ananda Mohan, Fellow, "RNS-To-Binary Converter for a New Three-Moduli Set $\{2^{n+1}-1, 2^n, 2^n-1\}$ ",IEEE Transactions on Circuits and Systems: Vol. 54(9), pp. 775-779, 2007.

[3] P.V. Ananda Mohan, "New reverse converters for the moduli set $\{2^n - 3, 2^n-1, 2^n + 1, 2^n + 3\}$", Elsevier, International Journal of Electrons and Communications, vol. 62, pp. 643-658, 2008.

[4] H. Siewobr, K.A. Gbolagade, Department of Computer Science, University for Development Studies, Navrongo, Ghana, "Modulo Operation Free Reverse Conversion in the Moduli Set $\{2^{2n}+1,2^n,2^{2n}-1\}$", International Journal of Computer Applications, vol. 85(18), pp. 0975 – 8887 2014.

[5] Y. Wang, X. Song, M. Aboulhamid, H. Shen "Adder based residue to binary number converters for $(2^n - 1, 2^n, 2^n + 1)$." IEEE Transactions on Signal Processing. Vol. 50(7), pp. 1772-1779, 2002.

[6] W. Wang, MNS. Swamy, MO. Ahmad, Y. Wang. A note on "a high-speed residue-to-binary converter for the moduli $\{2^k, 2^k - 1, 2^{k-1} - 1\}$ RNS and a scheme for its VLSI implementation". IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 47(12), pp. 1576-1581, 2002.

[7] B. Cao, CH. Chang, T. Srikanthan. "An efficient reverse converter for the 4-moduli set $\{2^n - 1, 2^n, 2^n + 1, 2^{2n} + 1\}$ based on the new Chinese remainder theorem". IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, vol. 50(10), pp. 1296-1303, 2003.

[8] P.V. Ananda Mohan, A.B. Premkumar. "RNS to binary converters for two four moduli sets $\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} - 1\}$ and $\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} + 1\}$. IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 54(6), pp. 1245-1254, 2007.

[9] M.H. Sheu, S.H. Lin, C. Chen, S.W. Yang. An efficient VLSI design for a residue to binary converter for general balance moduli $(2^n - 3, 2^n + 1, 2^n - 1, 2^n + 3)$", IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 51 (3), pp. 152-155, 2004.

[10] E. K. Bankas, K. A. Gbolagade, "A New Efficient RNS Reverse Converter for the 4-Moduli Set $\{2^n, 2^n + 1, 2^n - 1, 2^{2n+1} - 1\}$", World Academy of Science, Engineering and Technology International Journal of Computer, Electrical, Automation, Control and Information Engineering, vol. 8(2), pp. 328-332, 2014.

[11] M. Hoseinzadeh, K. Kia, "Effective Reverse Converter for General Tree Moduli Set $\{2^n - 1, 2^n + 1, 2^{pn+1} - 1\}$" International Journal of Image, Graphics and Signal Processin, vol. 9(6), pp. 37-43, 2012.

[12] A. Hariri, R. Rastegar, K. Navi, "High Dynamic Range 3-Moduli Set with Efficient Reverse Converter", Computers & Mathematics with Applications, vol. 55(4), pp. 660-668, 2008.