A MRC Based RNS to Binary Converter Using the Moduli Set \( \{2^{2n+1} - 1, 2^{n-1}, 2^{2n} - 1\} \)

H. K Bello, K.A Gbolagade

Abstract— This paper presents a new reverse converter for the moduli set \( 2^{2n+1} - 1, 2^{n-1}, 2^{2n} - 1 \). We used Mixed Radix conversion to convert the residues of the moduli to binary number. It is a 5-bit dynamic range and when compare to the literature state of the art reverse converter of the same dynamic range, it is considered to be slightly faster.

Index Terms— Dynamic Range, Reverse converter, Mixed Radix conversion, Moduli set, Residues, Binary number, RNS

I. INTRODUCTION

For many decades in computer arithmetic, Residue Number System (RNS) has been an interesting research field. It usually uses positional bases that are relatively prime to each other e.g. 7, 11, 13, 15 etc. It is used in Digital Signal Processing (DSP), Cryptography system and other systems where we need high speed and low-power hardware implementation [1]. This is possible because of the following advantages of RNS properties: carry free and borrow free of addition and subtraction respectively and error correcting properties [2,3]. These properties provide RNS with a high performance computing architecture [4]. RNS is capable of performing high speed of arithmetic, though, not popular in general arithmetic and computing due to its difficulties in division, magnitude detection and most especially error correction [5].

In this paper, a novel and efficient reverse converter for the \( \{2^{2n+1} - 1, 2^{n-1}, 2^{2n} - 1\} \) moduli set is proposed using a Mixed Radix Conversion method for the reverse conversion. The rest of this paper is organized as follows: Section II presented literature review; In section III, we discussed the proposed architecture of the system; In Section IV, we presented the hardware realization; In section V, performance evaluation was discussed and lastly section VI we presented the conclusion of the paper.

II. LITERATURE REVIEW

A. The Residue Number System (RNS)

RNS is an integer that capable of supporting high speed

Manuscript received July, 2017.
H.K Bello, Department of Computer Science, Federal Polytechnic, Offa, Kwara State, Nigeria. +2348067651314(e-mail: hassan.bello@fedpolyoffaonline.edu.ng)
K.A Gbolagade, Department of Computer Science, Kwara State University, Malete, Kwara State, Nigeria (e-mail: kazeem.gbolagade@kwazu.edu.ng)

concurrent automatic [6]. It may be defined as a set of relatively prime integer called moduli which can be denoted as \( m_1, m_2, m_3, \ldots m_n \) where \( m_i \) is the \( i \)-th modulus [7] and \( \gcd(m_i, m_j) = 1 \) for \( i \neq j \). Each \( X \) can be represented as a set of smaller integers called \( x_1, x_2, x_3, \ldots x_n \)

where \( X \mod m_i = x_i; x_i = [X]_{m_i} \)

RNS has the capabilities to support parallel carry free addition, borrow free subtraction and single step multiplication without partial fraction [8][9].

Many moduli sets are in used in RNS with different Dynamic Range (DR). The most popular 3n-bit DR moduli set known as traditional moduli set is \( \{2^{n-1}, 2^n, 2^{n+1}\} \) [10]. Others are \( \{2^{n-1}, 2^n-1, 2^n+1\} \) [11]. The dynamic range of \( 3n \)-bits produced by this moduli set will not be able to be used by any application with higher DR. 4n-bit DR 4 moduli set such as \( \{2^n-1, 2^n, 2^n+1, 2^{n+1}\} \) [12] were suggested. In order to raise parallelism in RNS arithmetic, 5n-bit moduli set were suggested \( \{2^n, 2^{2n}-1, 2^{3n}+1\} \), \( \{2^n-1, 2^n, 2^n+1, 2^{2n-1}\} \) etc. were also proposed.

1) Selection of Moduli in RNS

In RNS system, the Dynamic Range (DR) is the product of all the moduli such that the interval can be uniquely represented in RNS [13], this is an important aspect to be considered in the choice of moduli in RNS. Proper choice of moduli when designing RNS system is very essential; this is because the moduli choice affects the complexity of forward conversion, reverse conversion and RNS arithmetic circuits. Also, the speed of resulting conversion depends on selected moduli [14], therefore, moduli selection and data conversion are very critical in RNS to binary conversion [15]. However, it is a common fact that as the number of moduli increases, the speed of residue arithmetic units increases and the conversion from residue-binary becomes slower and complex [16]. Abdullahi and Skavantzos stated that the moduli set \( m_i \) and \( m_j \) satisfy the following criteria:

They must be pairwise prime (i.e. \( \gcd(m_i, m_j) = 1 \))

Each moduli \( m_i \) should be as small as possible so that operation modulo \( m_i \) require minimum computation time

The moduli \( m_i \), \( m_j \) should imply simple binary to Residue Number system and Residue Number System to binary conversion.

The moduli product should be large enough in order to implement the decision dynamic range.

However, in a case where a large dynamic range (DR) is required, large moduli might be in a better performance.

In addition, another fundamental area where attention should be taken is the hardware selection, which is also a determining factor for RNS performance. The simplified conversion equation of the system is computed by using adders like Carry

A MRC Based RNS to Binary Converter Using the Moduli Set \( \{2^{2n+1} - 1, 2^{n-1}, 2^{2n} - 1\} \)

www.ijarce...
Propagation Adder (CPA), Carry Save Adder (CSA) [17] etc.

2) Data Conversion method

The three main parts of RNS systems are binary to residue (forward) conversion, arithmetic operation and residue to binary (reverse) conversion [10]. Data conversion in RNS is either through Chinese Remainder Theorem (CRT) or Mixed Radix Conversion (MRC) which is also either forward or reverse conversion.

Forward Conversion: This is the conversion of a binary or decimal number to its RNS equivalent

Reverse Conversion: This is the conversion of RNS into its binary or decimal number.

Reverse conversion is somehow difficult, many algorithms have used various moduli sets to perform reverse conversion e.g. \(\{2^n, 2^n - 1, 2^n + 1\}\) [18], \(\{2^n, 2^{n-1}, 2^{n+1}\}\) [19], \(\{2^n, 2^n - 1, 2^{n+1}\}\) [20]. The Traditional methods used in reverse conversion are the Chinese Remainder Theorem (CRT) and the Mixed Radix Conversion (MRC) [21].

a) Chinese Reminder Theorem

The Chinese Remainder Theorem (CRT) is one of the methods used in reverse conversion, i.e conversion from RNS to binary or decimal numbers [8]. Each number in the dynamic range (DR) will have a unique representation in RNS if the moduli of RNS are correctly chosen [22].

Given a set of pairwise relatively prime moduli \(\{m_1, m_2, m_3, \ldots, m_n\}\) with residues \(\{x_1, x_2, x_3, \ldots, x_n\}\) in the system of X i.e. \(x_i\equiv x_{i\%m_i}\) then that number and its residue are related as follows

\[
x \equiv \sum_{i=1}^{n} x_i M_i^{-1} m_i \mod M
\]

Where \(M\) is the product of \(m_i\) and \(M_i^{-1} = \frac{M}{m_i}\) and \(\gcd(m_i, m_j) = 1\) (for \(i \neq j\))

\[
X = a_1 + a_2 m_1 + a_3 m_1 m_2 + \ldots + a_n m_1 m_2 \ldots m_{n-1}
\]

From the above diagram,

\[
X = a_1 + a_2 m_1 + a_3 m_1 m_2 + \ldots + a_n m_1 m_2 \ldots m_{n-1}
\]

where \(a_1 = x_1\)

\[
a_2 = \left|(x_2 - a_1) m_1^{-1}\right|_{m_2}
\]

\[
a_3 = \left|(x_3 - a_1) m_1^{-1} - a_2 m_2^{-1}\right|_{m_3} \ldots
\]

\[
a_k = \left|(x_k - a_1) m_1^{-1} - a_2 m_2^{-1} - \ldots - a_{k-1} m_{k-1}^{-1}\right|_{m_k}
\]

Mixed Radix numbers are very significant because they are weighted number system and facilitate implementation of operations such as magnitude comparison [23] which posed a lot of problems in RNS.

\[
\begin{align*}
\text{mod } m_1 & \rightarrow m_1 \\
\text{mod } m_2 & \rightarrow m_2 \\
\text{mod } m_3 & \rightarrow m_3 \\
\text{mod } m_n & \rightarrow m_n \\
+ & \\
\text{mod } M & \rightarrow M
\end{align*}
\]

\begin{table}
\begin{align*}
m_1 & \quad m_2 & \quad m_3 \\
x_1 & \quad x_2 & \quad x_3 \quad a_1 m_1^{-1} \quad a_2 m_2^{-1} \quad a_3 m_3^{-1} \\
\end{align*}
\end{table}

\[X = a_1 + a_2 m_1 + a_3 m_1 m_2 + \ldots + a_n m_1 m_2 \ldots m_{n-1}\]

III. THE PROPOSED ALGORITHM

In this paper, we assumed \(m_1 = 2^{2n-1} - 1, m_2 = 2^{2n-1} \) and \(m_3 = 2^n - 1\)

**Theorem 1:** The moduli set \(\{2^{2n-1} - 1\}, \{2^{2n-1}\}, \{2^n - 1\}\) are pairwise relatively prime numbers

**Proof:** Using Euclid’s theorem \(\gcd(m_1, m_2) = \gcd(m_2, m_3) = 1\), then \(\gcd(2^{2n-1} - 1, 2^{2n-1}) = \gcd(2^{2n-1}, 2^{2n-1} - 1)\)

\[
\text{To evaluate } 2^{2n-1} - 1 = 2^{2n-1} - 1 = 1
\]

\[
\text{Also } \implies \gcd(2^{2n-1}, 1) = 1
\]
gcd \( (2^n - 1, 2^n) \) = gcd(2\(^n\cdot1\),2\(^n\cdot1\))
if \( 2^n - 1 \) is -1
then gcd(2\(^n\cdot1\),2\(^n\cdot1\)) = gcd(2\(^n\cdot1\), -1) = 1

By the above proof, it is confirmed that 2\(^{n+1} - 1\), 2\(^n\), 2\(^2n\cdot1\) are relatively prime since the greatest common divisor (gcd) are 1. Therefore, our proposed moduli set can be used in RNS and we can proceed to determine its reverse converter.

**Theorem 2:** Given the moduli set 2\(^{n+1} - 1\), 2\(^n\), 2\(^2n\cdot1\) where \( m_1 = 2^{n+1} - 1\), \( m_2 = 2^n\), \( m_3 = 2^{2n} - 1\), the following hold to be true:
\[
\begin{align*}
|m_1| & = 2^{n+1} \quad (3) \\
|m_2| & = 2^n - 1 \quad (4) \\
|m_3| & = 1 \quad (5)
\end{align*}
\]

**Proof:**
From (3), it can be demonstrated that 2\(^{n+1}\) is the multiplicative inverse of 2\(^n\).
\[
2^{n+1} \cdot 2^n = 1
\]

Implies 2\(^{n+1}\) is the multiplicative inverse of 2\(^n\) with respect to 2\(^n\).

From (4),
\[
\frac{(2^{n+1} - 1) \cdot (2^n - 1)}{2^{n+1}} = (2^n - 1) + 2^n
\]

Implies 2\(^n\) - 1 is the multiplicative inverse of 2\(^{n+1}\) with respect to 2\(^{n+1}\).

From (5),
\[
\frac{(2^{n+1} - 1) \cdot 2^n}{2^{n+1}} = 2 \cdot 2^n - 1 + 2^n
\]

Implies 1 is the multiplicative inverse of 2\(^{2n+1}\) - 1 with respect to 2\(^{n+1}\).

**Property 1:** Modulus 2\(^n\) of a number is equivalent to s least significant figure of the number.

**Property 2:** modulo 2\(^n\) multiplication of a residue number by 2\(^n\) where s and t are positive integer numbers is equivalent to t-bit circular left shifting.

**Property 3:** modulo 2\(^n\) of a negative number is equivalent to the one’s complement of the number which is obtained by subtracting the number 2\(^{n+1}\) - 1, 2\(^n\), 2\(^2n\) will have residues \( x_1, x_2, x_3 \) respectively.

Using mixed radix conversion, let the binary representation of the residue be
\[
x_1 = x_{1,2n}x_{1,2n-1}x_{1,2n-2} \cdots x_{1,1}x_{1,0}
\]
\[
x_2 = x_{2,2n}x_{2,2n-1}x_{2,2n-2} \cdots x_{2,1}x_{2,0}
\]

(n-1) bit
\[
x_3 = x_{3,3n-1}x_{3,3n-2}x_{3,3n-3} \cdots x_{3,1}x_{3,0}
\]

Given the residues \( a_n, a_{n-1}, a_{n-2}, \ldots, a_1 \), any number \( X \) can be uniquely expressed in a mixed radix as [3] \( X \equiv (z_n, z_{n-1}, z_{n-2}, \ldots, z_1) \) which can be written as
\[
X = z_n a_n + z_{n-1} a_{n-1} + z_{n-2} a_{n-2} + \cdots + z_1 a_1 + z_0
\]

Given the above association between a mixed radix representation and let \( a_i \)'s corresponds to \( m_i \)'s in equation (9) with residue representation \( x_n x_{n-1} \cdots x_1 \)
\[
X = z_n m_{n-1} m_{n-2} \cdots m_1 + z_3 m_2 m_1 + z_2 m_1 + z_1
\]

Thus, if we consider the moduli set 2\(^{n+1}\) and 2\(^n\)
\( z_2 = (x,y) \) and by applying the mixed radix algorithm in (1) for the two moduli set and also equation (6).
\[
X = a_1 + a_2 (2^{n+1} - 1) + a_3 (2^n - 1)
\]

or \( N = x_3 + y_3 + y_4 + 0 \)

and \( y_3 = \frac{a_1}{x_2}2^n + 1 \)

or \( N = x_3 + y_3 + y_4 + 0 \)

\[
\begin{align*}
\text{or } N & = x_3 + y_3 + y_4 + 0
\end{align*}
\]

(15)
From equation (1)
\[ X = a_1 + a_2(2^{n+1} - 1) + a_3(2^{2n+1} - 1) \times 2^{-1} \]
\[ = a_1 + 2^{2n+1}a_2 - a_2 + 2^{3n}a_3 - 2^{n-1}a_3 \]  
(16)

Let \( X = \beta_3 + \beta_4 \)  
(17)

where, \( \beta_4 = 2^{3n}a_3 \)  
(18)
\[ \beta_3 = \beta_1 + \beta_2 \]  
(19)
\[ \beta_1 = a_1 + 2^{n+1}a_2 \]  
(20)
\[ \beta_2 = a_2 - 2^{n-1}a_3 \]  
(21)

We add 1 to \( \beta_2 \) so as to make it of the same bit with \( \beta_1 \)
\[ \beta_2 = 1a_{2,n}a_{2,n-3}a_{2,n-4} ... a_{2,0} + a_{3,2n-1}a_{3,2n-2}a_{3,2n-3} ... a_{3,0} \]  
(22)

From equation (14) and (15)
\[ X = a_{3,3n-1}a_{3,3n-2}a_{3,3n-3} ... a_{3,0} + \beta_{3,3n-1}\beta_{3,3n-2} \cdots \beta_{3,0} \]  
(23)

IV. HARDWARE REALIZATION

The hardware realization of the proposed system is based on (13), (14) or (15). This can be seen on table 1. The carry propagation Adder1 (CPA1) is used to add up \( x_2 \) and \( x_3 \). The carry save adder 1 (CSA1) is used to compute \( a_3 \). Carry Propagation adder 2 (CPA2) and Carry propagation adder 3 (CPA3) are in parallel where either of them can be used to compute (14) or (15). Carry propagation adder 4 (CPA 4) is used to add \( \beta_1 \) and \( \beta_2 \) to obtain \( \beta_3 \). 1 is appended to the concatenated result in equation (22) as to make \( \beta_2 \) a 3n-bit number.
Fig. 3: Hardware implementation of

V. PERFORMANCE EVALUATION

The performance of the proposed system is evaluated by comparing it with [8], [9] and [4] of the same DR. The author [8] compared the performance evaluation of his work with one of the best converter in literature \(2^n - 1, 2^n + 1, 2^{n+1} - 1\) [16] and found that the performance [8] is better than [16]. The performance evaluation of our proposed system is better than [8], [9] and [24].

<table>
<thead>
<tr>
<th>Converters</th>
<th>DR bits</th>
<th>Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>[9]</td>
<td>5n bits</td>
<td>(13n+4)</td>
</tr>
<tr>
<td>[24]</td>
<td>5n bits</td>
<td>(9n+6)</td>
</tr>
<tr>
<td>[8]</td>
<td>5n bits</td>
<td>(6n+3)</td>
</tr>
<tr>
<td>Proposed</td>
<td>5n bits</td>
<td>(6n+1)</td>
</tr>
</tbody>
</table>

VI. CONCLUSION

The author of [8] compared his work with one of the best converter in literature [16]. In this paper the proposed converter is compared with [8], [9] & [24] and the proposed system is slightly found faster.

REFERENCES

[1] Someyyah J. J, Amir S.M. “Hybrid RNS to Binary Converter for the Moduli set \{2^n, 2^n - 1, 2^n + 1, 2^{n+1} - 1\}” Research

[23] Leonel Sousa, Samuel Antao “ MRC based RNS reverse converters for the four moduli sets \([2n + 1, 2n-1, 2n, 22n+1 – 1]\) and \([2n + 1, 2n - 1, 22n, 22n+1 – 1]\). IEEE transactions on circuits and system 59-11(4), pp 244-248, 2012.