An Improved Residue to Binary Converter Based on Mixed-Radix Conversion for the Moduli Set $\{2^{2n+1}-1, 2^{2n}, 2^n-1\}$

Negovan Stamenković, Bojan Jovanović, and Vidosav Stojanović

***Abstract*— The increasing usage of Residual Number System (RNS) in signal processing applications demands the development of new and more adaptable RNS moduli sets and arithmetic units. In this paper, a new reverse converter for moduli set $\{2^{2n+1}-1, 2^{2n}, 2^n-1\}$, which can offers large dynamic range, is presented. We improved a previously introduced Mixed-Radix Converter architecture [1] for a high speed hardware design. Hardware architecture of proposed converter is based on adders and subtractors, without the need for ROM or multiplier. The presented design results in hardware saving comparison to the last reverse converter for the moduli set $\{2^{2n+1}-1, 2^{2n}, 2^n-1\}$.**

***Index Terms*— Residue Number System, Reverse Converter, Mixed Radix Conversion.**

I. Introduction

Residue Number System (RNS) architectures are typically composed of three main parts, namely, a binary-to-residue converter, residue arithmetic units, and a residue-to-binary converter [2], [3]. The residue-to-binary converter is the most complex part of any RNS architecture. Moduli set choice is also an important issue since the complexity and the speed of the resulting conversion structure depend on the chosen moduli set. Special moduli sets have been used extensively to reduce the hardware complexity in the implementation of residue to binary converters [4], [5]. Most popular three-moduli set is $\{2^n, 2^n-1, 2^n+1\}$ [5]–[7]. This moduli set has the disadvantage [8] that multiplication by powers of 2 with respect to the (2^n+1) modulus is not as simple as left circular rotation in a (2^n-1) modulus. However, larger dynamic ranges than the one provided by the moduli set proposed in [8], [9] are required. For this cases K. A. Gbolagade at al. recently proposed the moduli set $\{2^n-1, 2^{2n}, 2^{2n+1}-1\}$ which has sufficient dynamic range and avoids the modulo (2^n+1) type arithmetic [1]. In this paper K. A. Gbolagade at al. presented memoryless Chinese Remainder Theorem (CRT) based and, Mixed-Radix (MRC) based reverse converters. They showed that MRC based convertor is useful because it covers the entire dynamic range whereas CRT based convertor does not. However, multiplicative inverse proposed in this paper ($\langle m_1^{-1}\rangle_{m_2} = 2^{2n}-2^n-1$, $\langle m_2^{-1}\rangle_{m_3} = 2$ and $\langle m_1^{-1}\rangle_{m_2} = 2^{2n+1}-2^{n+1}-3$) are not best solution because two multiplicative inverse have complex values. In this paper we proposed the best values for multiplicative inverse for the same moduli set.

In this paper, we made improvement to the residue to binary converter for moduli set proposed in [1] that leads to hardware savings and improves performance of the system.

N. Stamenković is with Faculty off Natural Science, 28220 Kosovska Mitrovica, Lole Ribara 29, Serbia; e-mail: negovanstamenkovic@gmail.com.

B. Jovanovic and V. Stojanović are with Faculty of Electronic Engineering, Niš, Aleksandra Medvedeva 14, 18000 Niš, Serbia; e-mail: [bojan.jovanovic, vidosav.stojanovic]@elfak.ni.ac.rs.

The paper is organized as follows. In Section 2, we introduce the necessary background. The proposed improvements are presented in section 3. Section 4 provides hardware implementation, Section 5 is simulation, and Section 6 is conclusion.

II. Background

A. Residue Number System

A residue number system (RNS) is defined in terms of a relatively-prime moduli set $\{m_1, m_2, \dots, m_3\}$ that is $\gcd(m_i, m_j) = 1$ for $i \neq j$ [10]. The greatest common divisor (gcd) for a pair of numbers (a,b), can be calculated by the well known Euclidian algorithm. A binary number X can be represented as $X = (x_1, x_2, \dots, x_n)$, where

$$x_i = X \mod m_i = \langle X\rangle_{m_i}, \quad 0 \le x_i < m_i \tag{1}$$

such a representation is unique for any integer X in the range $[0, M-1]$, where $M = m_1 m_2 \cdots m_n$ is the dynamic range of the moduli set $\{m_1, m_2, \dots, m_n\}$. To perform the residue to binary conversion, that is to convert the residue number $(x_1, x_2, \dots, x_n)$ into the binary number X, the chinese remainder theorem (CRT) and mixed-radix conversion (MRC) are generally used.

B. Chinese Remainder Theorem

The binary number X is computed by

$$X = \left\langle \sum_{i=1}^{n} \langle x_i N_i\rangle_{m_i} M_1 \right\rangle_M \tag{2}$$

where $M_i = M/m_i$ and $N_i = \langle M_i^{-1}\rangle_{m_i}$ is the multiplicative inverse of M_i modulo m_i. The main drawback of this approach is that it requires multiplication by the M_is, which are large numbers, and modulo M operations

C. Mixed-Radix Conversion

The number X can be computed by

$$X = a_n \prod_{i=1}^{n} M_i + \cdots + a_3 m_1 m_2 + a_2 m_1 + a_1 \tag{3}$$

where a_is are called the mixed-radix coefficients and they can be obtained from the residues by

$$a_n = \Big\langle \big(((x_n - a_1)\langle m_1^{-1}\rangle_{m_n} - a_2)\langle m_2^{-1}\rangle_{m_n} - \cdots - a_{n-1}\big)\langle m_{n-1}^{-1}\rangle_{m_n} \Big\rangle_{m_n} \tag{4}$$

where $n > 1$ and $a_1 = x_1$. For MRDs a_i, $0 \le a_i < m_i$, any positive number in interval $[0, M-1]$ is uniquely represented. The Mixed Radix Conversion is a strictly sequential process. There is no need for final modulo reduction.

III. The proposed improvements

Suppose that we have residue number $\{x_1, x_2, x_3, \}$, $0 \leq x_i < m_i$, for the moduli set $\{m_1, m_2, m_3, \}$. The binary equivalent X of the residues can be computed as follows [10]

$$X = a_1 + a_2 m_1 + a_3 m_1 m_2 \tag{5}$$

were 1, m_1 and $m_1 m_2$ are numerical base, a_1, a_1 and a_3 are mixed radix digits. In (5), a_1, a_2 and a_3 are represented as an sequential algorithm

$$\begin{aligned} a_1 &= x_1 \\ a_2 &= \left\langle (x_2 - a_1) c_{12} \right\rangle_{m_2} \\ a_3 &= \left\langle ((x_3 - a_1) c_{13} - a_2) c_{23} \right\rangle_{m_3} \end{aligned} \tag{6}$$

where $c_{i,j}$ for $1 \leq i \leq j < 3$ is the multiplicative inverse of m_i modulo m_j, or $\langle c_{ij} \times m_i \rangle_{m_j} = 1$. If the mixed-radix digits are given, any number in the interval $[0, M-1]$ can be uniquely represented.

Well known block diagram of MRC Converter for three moduli set is in Figure 1 displayed.

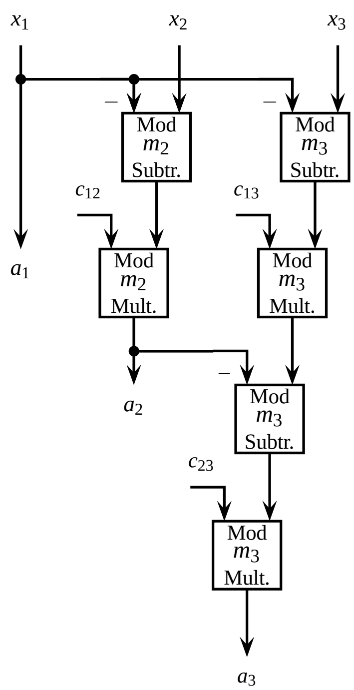

Fig. 1: MRC Converter for three moduli set.

For given the RNS number $\{x_1, x_2, x_3\}$ with respect to the moduli $\{2^n - 1, 2^{2n}, 2^{2n+1} - 1\}$ the following hold true [1]

$$\begin{aligned} c_{12} &= 2^{2n} - 2^n - 1 \\ c_{23} &= 2 \\ c_{13} &= 2^{2n+1} - 2^{n+1} - 3 \end{aligned} \tag{7}$$

Definition 1: Digits in the residue number system have no ordering significance. In residue addition, subtraction, and multiplication, any particular digit of the resultant depends solely on the corresponding digits of its operands. However, Residue to Mixed-Radix Conversion depends of the digit ordering as shown in (4). Further, mixed-radix digits ordering depends of the moduli set ordering. Due to this reason we define *the form of moduli set*: the order of modules in the residue number system. For example, assuming three moduli $2^n - 1$, 2^{2n}, $2^{2n+1} - 1$ we define the first form of moduli set in descending order $\{2^{2n+1} - 1, 2^{2n}, 2^n - 1\}$, second form $\{2^{2n+1} - 1, 2^n - 1, 2^{2n}\}$, and so on. A set of three modules has six forms. Finally, the sixth form is a set of modules in ascending order. Thus, the modulo at first position is m_1, at second position is m_2, and at third position is m_3. ■

Multiplicative inverse for all six forms of given moduli set are shown in Table I.The first form of given moduli set $\{2^{2n+1} - 1, 2^{2n}, 2^n - 1\}$ provides the best solution for c_{ij}: $c_{12} = -1$ and $c_{13} = c_{23} = 1$. It can be seen that the fourth form of moduli set ($\{2^{2n}, 2^{2n+1} - 1, 2^n - 1\}$) also provides a good solution.

Using the first form of given moduli set mixed-radix digits can be represented as

$$a_1 = x_1 \tag{8}$$

$$a_2 = \langle x_1 - x_2 \rangle_{2^{2n}} \tag{9}$$

$$a_3 = \langle \langle x_3 - x_1 \rangle_{2^n - 1} - a_2 \rangle_{2^n - 1} \tag{10}$$

Operands a_1, a_2 and a_3 are $(2n+1)$-bit, $2n$-bit and n-bit, respectively. The proposed hardware realization of RNS to mixed-radix conversion is depicted in Figure 2(a). We proposed here a new modulo $(2^n - 1)$ subtraction algorithm that avoids the double representation of zero. Figure 2(b) illustrates the architecture of this new operator which requires two borrow propagate subtractor (BPS).

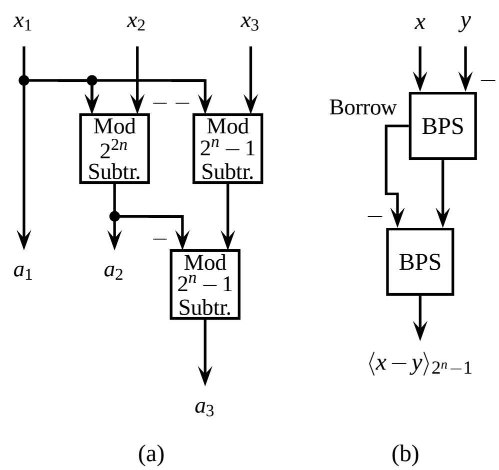

Fig. 2: Proposed MCR converter for the first form of moduli set $\{2^{2n+1} - 1, 2^{2n}, 2^n - 1\}$ (a), and modulo $\langle x - y \rangle_{2^n - 1}$ subtractor (b).

It is known, modulo $(2^n - 1)$ of a negative number is accomplished by subtracting this number from $(2^n - 1)$. This is equivalent to taking one's complement of this number. However, using subtractors we avoids ones's complement operation.

IV. Hardware implementation

Since most values that need to be processed are represented in binary, it is necessary to convert them to an RNS representation, thus binary to RNS conversion units and RNS to binary are demanded in this type of systems.

The hardware structure proposed RNS to mixed-radix numbers conversion depicted in Figure 3. Converter contains three subtractors: one modulo 2^{2n} and two moduli $2^n - 1$. This converter also contains two binary to RNS converters. The first for converting binary numbers from the $2^{2n+1} - 1$ channel to the $2^n - 1$ channel,

TABLE I: Multiplicative inverse c_{ij} of m_i and m_j for different forms of a set of modules.

Form	m_1	m_2	m_3	c_{12}	c_{13}	c_{23}
1	$2^{2n+1}-1$	2^{2n}	2^n-1	-1	1	1
2	$2^{2n+1}-1$	2^n-1	2^{2n}	1	-1	$2^{2n}-2^n-1$
3	2^{2n}	2^n-1	$2^{2n+1}-1$	1	2	$2^{2n+1}-2^{n+1}-3$
4	2^{2n}	$2^{2n+1}-1$	2^n-1	2	1	1
5	2^n-1	$2^{2n+1}-1$	2^{2n}	$2^{2n+1}-2^{n+1}-3$	$2^{2n}-2^n-1$	-1
6	2^n-1	2^{2n}	$2^{2n+1}-1$	$2^{2n}-2^n-1$	$2^{2n+1}-2^{n+1}-3$	2

and the second for converting binary numbers from the 2^{2n} channel to the 2^n-1 channel.

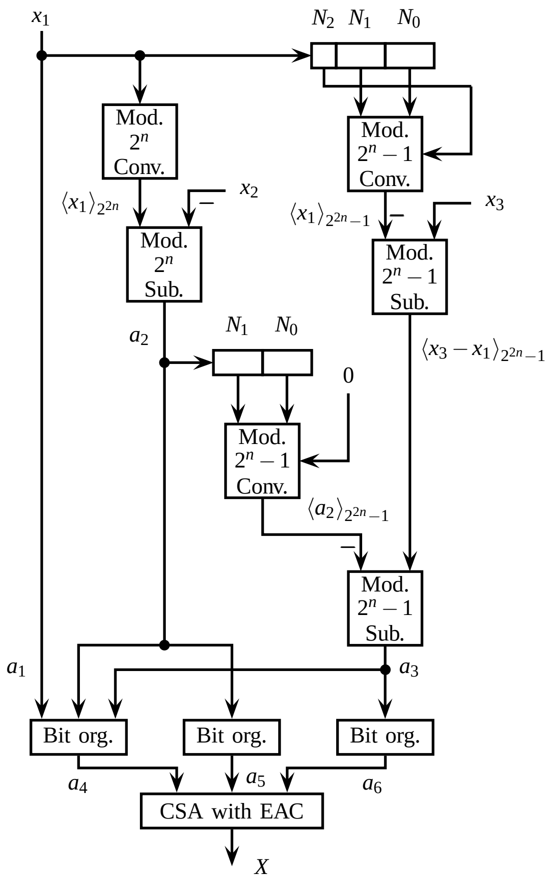

Fig. 3: Hardware realization of residue number system to mixed-radix conversion

The simplest one is the converter for the m_2 channel. The value $\langle x_1\rangle_{2^{2n}}$ can be obtained by the remainder of the division of x_1 by 2^{2n}, which can be accomplished by truncating the binary value $x_1=X_{2n}X_{2n-1}\cdots X_1X_0$. Since x_1 is binary number on $2n+1$ bits, then

$$\langle x_1\rangle_{2^{2n}}=X_{2n-1}X_{2n-2}\cdots X_1X_0.$$

For the 2^n-1 channel the calculation of the corresponding residues is more complex, since the final result of the conversion depends on the value of all the X bits. Instead of using a division operation to calculate the 2^n-1 residue, which is a complex operation and expensive both in terms of area and speed, this calculation can be performed as a sequence of additions, as described below:

$$\langle x_1\rangle_{2^n-1}=\langle N_2 2^{2n}+N_1 2^n+N_0\rangle_{2^n-1} \tag{11}$$

By taking the equation:

$$\langle 2^n\rangle_{2^n-1}=1 \tag{12}$$

equation (11) can be rewritten as:

$$\langle x_1\rangle_{2^n-1}=\langle N_2+N_1+N_0\rangle_{2^n-1} \tag{13}$$

Thus the conversion of x_1 to moduli 2^n-1 can be performed simply by modulo 2^n-1 adding the N_0 and N_0 components of x_1. For our design these two operands N_0 and N_1 are binary numbers on n bits, while N_3 is one for forward conversion x_1, or zero for a_2 forward conversion, with modulo 2^n-1.

In designing a modulo 2^n-1 adder, it is useful to distinguish among three cases, depending on the intermediate result of the addition of the two operands, N_1 and N_2, where $0\le N_1;N_2<2^n-1$ [11]:

- $0\le N_1+N_2<2^n-1$;
- $N_1+N_2=2^n-1$;
- $2^n-1<N_1+N_2<2^{n+1}-2$.

In the first case, the intermediate result is the correct modulo 2^n-1 result. In the second and third cases, we should subtract 2^n-1 in order to get the correct result; this subtraction is equivalent to subtracting 2^n and adding 1.

Fig. 4 shows the hardware architecture of the RNS to binary conversion for the modulo 2^n-1. For residue number x_1 each of the N_1 and N_2 is the n bits binary numbers, but N_2 is one bit binary number. On the other hand, mixed-radix coefficient a_2 is represented with $2n$ bits, i.e. only N_0 N_1 exist.

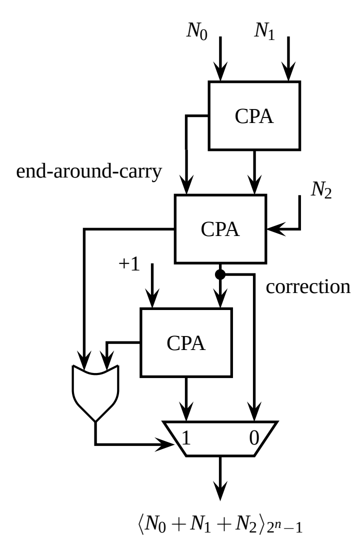

Fig. 4: Binary to RNS conversion for modulo 2^n-1.

A. Mixed-radix to binary conversion

The hardware realization of (5) can be represented as

$$\begin{aligned} X &= (a_1 + 2^{2n+1}a_2 + 2^{4n+1}a_3) - a_2 - 2^{2n}a_3 \\ &= a_4 + a_5 + a_6 \end{aligned} \tag{14}$$

were

$$\begin{aligned} a_4 =& a_1 + 2^{2n+1}a_2 + 2^{4n+1}a_3 \\ =& (\underbrace{a_{1,2n}, a_{1,2n-1}, \dots, a_{1,0}}_{2n+1}) \\ &+ (\underbrace{a_{2,2n-1}, a_{2,2n-2}, \dots, a_{2,0}}_{2n}, \underbrace{0, 0 \dots, 0}_{2n+1}) \\ &+ (\underbrace{a_{3,n-1}, a_{3,n-2}, \dots, a_{3,0}}_{n}, \underbrace{0, 0, \dots, 0}_{4n+1}) \\ =& (\underbrace{a_{3,n-1}, \dots, a_{3,0}, a_{2,2n-1}, \dots, a_{2,0}, a_{1,2n}, \dots, a_{1,0}}_{5n+1}) \end{aligned} \tag{15}$$

Operand a_5 and a_6 must be expanded to $(5n+1)$-bit number since operand a_4 is a $(5n+1)$-bit number.

$$\begin{aligned} a_5 &= -a_2 \\ &= -(\underbrace{a_{2,2n-1}, a_{2,2n-2}, \dots, a_{2,0}}_{2n}) \\ &= -(\underbrace{0, 0, \dots, 0}_{3n+1}, \underbrace{a_{2,2n-1}, a_{2,2n-2}, \dots, a_{2,0}}_{2n}) \\ &= (\underbrace{1, 1, \dots, 1}_{3n+1}, \underbrace{\overline{a}_{2,2n-1}, \overline{a}_{2,2n-2}, \dots, \overline{a}_{2,0}}_{2n}) \end{aligned} \tag{16}$$

$$\begin{aligned} a_6 &= -2^{2n}a_3 \\ &= -(\underbrace{0, 0, \dots, 0}_{2n+1}, \underbrace{a_{3,n-1}, a_{3,n-2}, \dots, a_{3,0}}_{n}, \underbrace{0, 0, \dots, 0}_{2n}) \\ &= (\underbrace{1, 1, \dots, 1}_{2n+1}, \underbrace{\overline{a}_{3,n-1}, \overline{a}_{3,n-2}, \dots, \overline{a}_{3,0}}_{n}, \underbrace{1, 1, \dots, 1}_{2n}) \end{aligned} \tag{17}$$

Hardware structure for mixed-radix to binary conversion, based on the equations (15), (16) and (17), contain only Carry-Save-Adders (CSA) with End-Around-Carry (EAC). Operand a_4 is simply obtained by concatenating mixed-radix digits a_1, a_2 and a_3 which are $(2n+1)$ bits, $2n$ bits and n bits, respectively. Operand a_5 is complemented mixed radix digit a_2 which is first expanded to $(5n+1)$ bits. Operand a_6 is one's complement of binary numbers which is obtained by left shift of mixed radix digit a_3 by $2n$ bits and then it is expanded to $(5n+1)$ bits.

V. Simulation

Let is give the number $X = 43210$ or in RNS notation, for $n = 3$, it is $X = \{30, 10, 6\}_{RNS\{127,64,7\}}$ or in binary form these are

x_1	0011110
x_2	001010
x_3	110

We convert this RNS number representation into the mixed-radix number representation with a_1, a_2, a_3 using Mixed-Radix convertor shown in Figure 3: $a_1 = 30$, $a_2 = 20$ and $a_3 = 5$, or in binary representation these are

a_1	0011110
a_2	010100
a_3	101

After a bit of organization, based on equations (15), (16) and (17), we get

a_4	1010101000011110
a_5	+1111111111101011
a_6	+1111111010111111
Partial sum	0101010110100101 0
Carry output	1111111010111111 0
Sum	|10|1010100011001000
End-Around-Carry	10
Final result	1010100011001010

The following holds true

$$1010100011001010_2 = 43210_{10}$$

VI. Conclusion

This paper presents an improved mixed-radix reverse converter for the recently proposed residue number system moduli set $\{2^{2n+1}-1, 2^{2n}, 2^n-1\}$. The hardware architecture of proposed converter consist of two levels. The first level is RNS to mixed-radix conversion. It is improved by using optimal choice of form of moduli set.

The second level is hardware architecture. It is composed of regular binary adders and subtractor, without the need for using modular adders. The highest number of arithmetic operations are with binary numbers of n bits. Proposed RNS reverse converter can be efficiently implemented, resulting in higher overall performance of the RNS system.

References

[1] K. A. Gbolagade, R. Chaves, L. Sousa, and S. D. Cotofana, "Residue-to-binary converters for moduli set $\{2^{2n+1}-1, 2^{2n}, 2^n-1\}$," in *2nd Int. Circuits on Adaptive Science and Technology (ICAST09)*, Accora, Ghana, Dec. 2009, pp. 26–33.

[2] W. K. Jenkins and B. Leon, "The use of residue number systems in the design of finite impulse response digital filters," *IEEE Trans. on Circuits and Systems*, vol. CAS-24, no. 4, pp. 191–201, Apr. 1977.

[3] S. Pontarelliyz, G. Cardarilliy, M. Rey, and A. Salsanoy, "Totally fault tolerant rns based FIR filters," in *14th IEEE International On-Line Testing Symposium, IOLTS'08.*, July 7–9, 2008, pp. 192–194.

[4] K. M. Ibrahim and S. N. Saloum, "The digit parallel method for fast rns to weighted number system conversion for specific moduli $(2^k-1, 2^k, 2^k+1)$," *IEEE Transactions on Circuits And Systems*, vol. 35, no. 9, pp. 1156–1158, Sept. 1988.

[5] W. Wang, M. Swamy, M. Ahmad, and W. Wang, "A study of residue to binary converters for the three-moduli sets," *IEEE Trans. on Circuits and Syst-Fundamental Theory and Applications*, vol. 50, no. 2, pp. 235–245, 2003.

[6] S. Andraos and H. Ahmad, "A new efficient memoryless residue to binary converter," *IEEE Transactions On Circuits And Systems*, vol. 35, no. 11, pp. 1441–1444, Nov. 1988.

[7] F. Pourbigharaz, "A signed-digit architecture for residue to binary transformation," *IEEE Transactions on Computers*, vol. 46, no. 10, pp. 1146–1150, Oct. 1997.

[8] A. Hiasat and A. Zohdy, "Residue-to-binary arithmetic converter for the moduly set $(2^k, 2^k-1, 2^{k-1}-1)$," *IEEE Trans. on Circuits and Systems-II: Analog and Digital Signal Processing*, vol. 45, no. 2, pp. 204–209, Feb. 1998.

[9] P. V. A. Mohan and 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 trans. on Circuits and System-I:Regular Papers*, vol. 54, pp. 1245–1254, June 2007.

[10] N. Szabo and R. I. Tanaka, *Residue Arithmetic and its Application to Computer Technology*. New York: McGraw-Hill, 1967.

[11] R. Chaves and L. Sousa, "$\{2^n+1, 2^{n+k}, 2^n-1\}$: A new rns moduli set extension," in *Proceedings of the EUROMICRO Systems on Digital System Design (DSD'04)*, Remes, France, Aug. 31/Sept. 03 2004, pp. 210–217.