Mohsen Mojahed
Amir Sabbagh Molahosseini
Azadeh Alsadat Emrani Zarandi

MULTIFUNCTIONAL UNIT FOR REVERSE CONVERSION AND SIGN DETECTION BASED ON FIVE-MODULI SET $\left\{\mathbf{2^{2n},\ 2^n+1,\ 2^n-1,\ 2^n+3,\ 2^n-3}\right\}$

Abstract *A high dynamic range moduli set $\left\{2^{2n},\ 2^n+1,\ 2^n-1,\ 2^n+3,\ 2^n-3\right\}$ has recently been introduced as an arithmetically balanced five-moduli set for the residue number system (RNS). In order to utilize this moduli set in applications handling signed numbers, two important components are needed: a sign detector, and a signed reverse converter. However, having both of these components results in high-hardware requirements, which makes RNS impractical. This paper overcomes this problem by designing a unified unit that can perform both signed reverse conversion as well as sign detection through the reuse of hardware. To the authors' knowledge, this is the first attempt to design a sign detector for a moduli set that includes a $\{2^n 3\}$ moduli. In order to achieve a hardware-amenable design, we first improved the performance of the previous unsigned reverse converter for this moduli set. Then, we extracted a sign-detection method from the structure of the reverse converter. Finally, we made an unsigned reverse converter-to-sign converter through the use of the extracted sign signal from the reverse converter. The experimental results show that the proposed reverse convertor and sign detector result in improvements of 31% and 28% in area and delay, respectively, as compared to the previous unsigned reverse convertor with sign output using a comparator.*

Keywords computer arithmetic, residue number system, reverse converter

Citation Computer Science 22(1) 2021: 101–121

1. Introduction

Computer arithmetic plays a significant role in modern computational systems, as designing efficient arithmetic circuits for different numbers [21, 24] is important for modern high-performance applications. The residue number system (RNS) [4] has been known as a tool that provides parallelism for the efficient implementation of arithmetic operations, including addition and multiplication [15]. Despite the traditional applications of RNS (including digital signal processing [2] and cryptography [22]), it has been used in emerging technologies such as deep learning [19] and DNA arithmetic [27].

The most fundamental part of the design of an RNS system is the moduli set selection [20]. The rate of the parallelism as well as the complexity of the inter-modulo operations and dynamic range are based on the moduli set. Due to this, various special moduli sets have been introduced for RNS, which are categorized as arithmetic-friendly (balanced) and conversion-friendly (unbalanced) moduli sets [3,7,8,10,12–14,16–18,25]. Arithmetic-friendly moduli sets are suitable for applications where the rates of internal addition and multiplication are significantly higher than the required conversions, such as cryptography and deep convolutional neural networks. Among the balanced moduli sets, $\{2^{2n},\ 2^n+1,\ 2^n-1,\ 2^n+3,\ 2^n-3\}$ [1] is one of the interesting ones due to the use of 2^{2n} together with fully balanced moduli $2n \pm 1$ and $2n \pm 3$. However, the inclusion of the $2n \pm 3$ moduli results in the increased complexity of inter-moduli operations such as reverse conversion, sign detection, and magnitude comparison. These problems limit the use of this moduli set in limited unsigned applications, while many applications such as deep learning require working on signed numbers.

In this paper, we have designed essential components for designing a signed residue number system based on moduli set $\{2^{2n},\ 2^n+1,\ 2^n-1,\ 2^n+3,\ 2^n-3\}$. First, the previous unsigned reverse converter for this moduli set is improved in order to reduce the hardware complexity and delay. Then, we extract a sign-detection algorithm for this moduli set. The proposed sign-detection algorithm has been implemented using the same reverse converter circuits, making it possible to have a unified unit that can perform two operations; namely, sign detection and signed reverse conversion. The experimental results show the effectiveness of the proposed signed components.

Previous work on multifunctional unit design for RNS was reported in [18]. The proposed work differs from [18] from two main aspects. First, moduli set $\{2^{2n},\ 2^n+1,\ 2^n-1,\ 2^n+3,\ 2^n-3\}$ is not considered in [17]. Second, the RNS comparison is not included in the proposed multifunctional unit in order to reduce the hardware area.

In the rest of the paper, related works are investigated in Section 2, the formulas and main structure of the unsigned reverse converter of moduli set $\{2^{2n},\ 2^n+1,\ 2^n-1,\ 2^n+3,\ 2^n-3\}$ are briefly reviewed in Section 3. Section 4 presents the reverse converter. Section 5 describes the proposed sign-detection algorithm and implementation. Finally, an evaluation of the experimental results is presented in Section 6.

2. Related works

One major barrier when using RNS is the absence of sign detection where, in contrast to binary number systems, there is no sign bit. The sign of a number in RNS is determined on the basis of dynamic range division and whether the number lies in the upper or lower range. This is usually done by the use of a comparator and is considered to be an arduous operation in RNS. Conventionally, the sign is detected using a comparator in the output of the reverse convertor and conditional operation [26]. Recently, a new approach has been introduced to improve and enhance the efficiency of reverse convertors with no detection where the sign can be obtained from within the convertor so as to reduce the chip-area, delay, and power consumption. Extensive studies have therefore been conducted on sign detection for popular moduli sets [5, 6, 9, 23]. However, due to the lack of a large dynamic range, such sets are not considered to be suitable in a number of applications that require high-speed computations such as encryption [11]. Nowadays, researchers have focused their attention towards finding multi-moduli sets (more than three moduli) [1]. Designing a signed reverse convertor for such sets has numerous complexities. In [26], however, a new approach was proposed for designing reverse convertors for a class of moduli sets of composite form $\left\{2^k, 2^p - 1\right\}$ known as C-class. Applying these sets result in the creation of computational channels for the easier processing of digital signals based on RNS.

The Chinese Remainder Theorem (New CRT-I) is one of the best choices for designing convertors for such sets, the formula of which is in the form of $X = x_1 + 2^k Y$. Here, by dividing the dynamic range into two parts, the upper half (which is smaller than $M/2$) has an MSB that is equal to zero and is within the range of positive numbers, whereas the lower half (which is greater than $M/2$) has an MSB that is equal to one and is within the range of negative numbers. The $M/2$ number with an MSB that is equal to zero has K states. The first half of these K states has an MSB that is equal to zero and is within the range of positive numbers, and the other half has an MSB that is equal to one and lies within the range of negative numbers. In [26], the sign is detected by designing a detection unit that is comprised of a number of logical gates and by analyzing the MSBs. In [18], another classification was made of moduli sets (known as A-class) in the form of $\left\{2^k, 2^p - 1, 2^{2n} - 1\right\}$ for sign detection. This was done by applying changes in the previous detection unit and by using Chinese Remainder Theorem 2 (New CRT-II) for designing reverse convertors. A similar approach to sign detection has been used in the present study.

3. Background

The residue number system is a modular number system that can perform parallel arithmetic operations without carry-propagation among the residue digits [4]. It can be designed based on some pair-wise relatively prime numbers that can form a moduli set;

i.e., $\{m_1, m_2, ..., m_n\}$. Then, the regular weighted binary numbers can be transformed to residue set $(x_1, x_2, ..., x_n)$ according to the following relationship:

$$x_i = X \mod m_i = |X|_{m_i} \tag{1}$$

The product of the moduli (i.e., $M = m_1 \times m_2 \times ... \times m_n$) defines the dynamic range (DR). The DR represents the range of integer numbers that can be represented in an RNS (that is, $[0, M)$ in an unsigned RNS). In order to realize the signed RNS, the DR is categorized into two sections (as follows) [26]:

$$when\, M\, is\, even: \begin{cases} \left[0, \frac{M}{2} - 1\right] & : Positive\ Numbers \\ \\ \left[\frac{M}{2}, M - 1\right] & : Negative\ Numbers \end{cases} \tag{2}$$

$$when\, M\, is\, odd: \begin{cases} \left[0, \frac{M-1}{2}\right] & : Positive\ Numbers \\ \\ \left[\frac{M+1}{2}, M - 1\right] & : Negative\ Numbers \end{cases} \tag{3}$$

The RNS-represented numbers can be converted back to normal weighted representations using the reverse converter. The reverse conversion can be done using New CRT-II [17]. For instance, RNS number (x_1, x_2, x_3, x_4) can be converted into its equivalent weighted number X using four-moduli set $\{m_1, m_2, m_3, m_4\}$ as follows [13]:

$$X = Z + m_1 m_2 |k_1(Y - Z)|_{m_3 m_4} \tag{4}$$

$$Z = x_1 + m_1 |k_2(x_2 - x_1)|_{m_2} \tag{5}$$

$$Y = x_3 + m_3 |k_3(x_4 - x_3)|_{m_4} \tag{6}$$

where the required multiplicative inverses can be achieved by considering the following relationships:

$$|k_1 m_1 m_2|_{m_3 m_4} = 1 \tag{7}$$

$$|k_2 m_1|_{m_2} = 1 \tag{8}$$

$$|k_3 m_3|_{m_4} = 1 \tag{9}$$

The reverse converter for five-moduli set $\{2^{2n},\ 2^n + 1,\ 2^n - 1,\ 2^n + 3,\ 2^n - 3\}$ has been designed using New CRT-II in [1] according to the two- and three-layer structures. In the next section, the proposed sign-detection algorithm is described, which is based on the reverse converter of [1]. Due to this, the main formulas and architectures of [1] are briefly reviewed here. The first stage of the converter of [1] includes two parallel two-moduli reverse conversions. First, the second and third moduli (i.e., $\{2^n - 1, 2^n + 1\}$) are combined using CRT-II as follows:

$$X_M = |x_0 + k_M(2^n - 1)(x_1 - x_0)|_{2^{2n}-1} \tag{10}$$

Then, the last two moduli (i.e., $\{2^n + 3, 2^n - 3\}$) can be combined according to the following relationship:

$$X_N = x_2 + (2^n + 3)|k_N(x_3 - x_2)|_{2^n-3} \tag{11}$$

The results of stage one (which are X_M and X_N) are used to construct the final number according to composite moduli set $F = \{2^{2n}, 2^{2n} - 1, 2^{2n} - 9\}$ with corresponding residues (x_c, X_M, X_N) as follows:

$$X_F = x_C + 2^{2n}|k_{MF}(x_N - x_C) - k_{NF}D_N(x_M - x_N)|_{D_M D_N} \tag{12}$$

where K_M, K_N, K_{MF} and K_{NF} are multiplicative inverses that are calculated and proven in [1]. After inserting the required multiplicative inverses, (12) can be calculated according to the following formulas [1] for $n = 3\alpha$:

$$X_F = x_C + 2^{2n}Y \tag{13}$$

where

$$\begin{aligned} Y =& |2^{2n}G_0 + U - 9G_0 - 7\rho U|_{D_M D_N} \\ =& \left|2^{2n}G_0 + 9\bar{G}_0 + 8\rho\bar{U} + (\rho+1)\,U + D_M\,(D_N - 8\rho - 9)\right|_{D_M D_N} \end{aligned} \tag{14}$$

$$U = X_N - X_C, \tag{15}$$

$$V = -(X_M - X_N) \tag{16}$$

$$\bar{U} = D_M - U \tag{17}$$

$$G_0 = \left|7\rho U + 2^{2n-3}V\right|_{D_M} = \left|8\rho U + \rho\bar{U} + 2^{2n-3}V\right|_{D_M} \tag{18}$$

$$\rho = \frac{2^{6\alpha} - 1}{2^6 - 1} \tag{19}$$

$$D_M = 2^{2n} - 1 \tag{20}$$

$$D_N = 2^{2n} - 9 \tag{21}$$

The overall structure of the reverse converter for moduli set $\{2^{2n},\ 2^n + 1,\ 2^n - 1,\ 2^n + 3,\ 2^n - 3\}$ is depicted in Figure 1. OPU-M and OPU-N compute X_M and X_N using carry-save adders (CSAs) and modulo carry-propagate adders (CPAs).

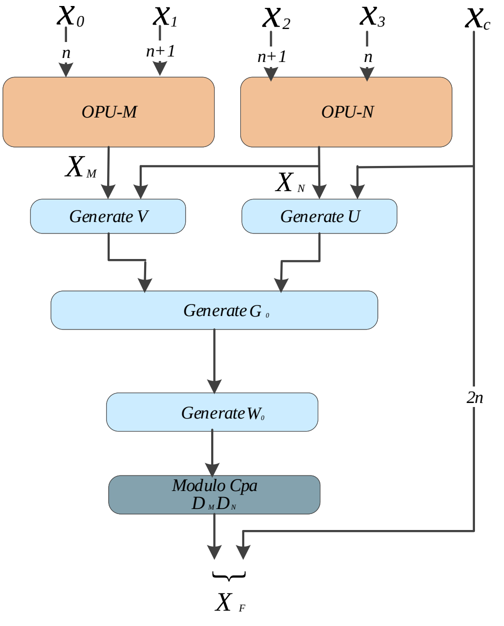

Figure 1. Unsigned reverse converter of [26]

4. Reverse converter

In this section, we describe the reverse converter of [1] with small modifications based on CRT-II for moduli set $\{2^{2n}, 2^n+3, 2^n-3, 2^n+1, 2^n-1\}$ to enable efficient sign extraction from the reverse converter. In the first step (Section 4.1), the reverse converter of subset $\{2^n+3, 2^n-3\}$ is obtained. In the second step (Section 4.2), the value of W is obtained using subset $\{2^n+1, 2^n-1\}$. In the third step (Section 4.3), the value of Z is obtained using subset $\{2^{2n}-9, 2^{2n}\}$. Finally, in the last step (Section 4.4), the value of X is obtained using subset $\{2^{2n}(2^{2n}-9), (2^{2n}-1)\}$.

4.1. Reverse converter for set $\mathrm{N} = \{\mathbf{2^n+3, 2^n-3}\}$

To design the reverse converter of two-moduli set $\{2^n+3, 2^n-3\}$, the new two-channel Chinese Remainder Theorem algorithm [1] is used as follows:

$$X_N = x_1 + (2^n+3)|k_N(x_2-x_1)|_{2^n-3} \tag{22}$$

Moduli 2^n+3 and 2^n-3 are pairwise relatively prime numbers; thus, the multiplicative inverses are calculated as follows [1]:

$$|k_N \times (2^n+3)|_{2^n-3} = 1 \rightarrow k_N = |(2^n+3)^{-1}|_{2^n-3} \rightarrow$$
$$|k_N \times (2^n+3)|_{2^n-3} = |k_N \times 6|_{2^n-3} \rightarrow k_N = |\frac{1}{6}|_{2^n-3} \tag{23}$$

Therefore, (23) can be computed as follows:

$$k_N = \begin{cases} \frac{\left(2^{n-1}-2\right)}{-3} = -\left(2^{n-3}+2^{n-5}+\ldots+2^{1}\right) & : even\ n \\ \frac{\left(2^{n-1}-1\right)}{3} = \left(2^{n-3}+2^{n-5}+\ldots+2^{0}\right) & : odd\ n \end{cases} \tag{24}$$

Now, X_N in (22) can be calculated using (24). Note that (22) can be rewritten as (25) by separating the most significant bit (MSB) of x_1 from its remaining least significant bits (LSBs) [1]. Its implementation is shown in Figure 2 as Modular Addition 3 (Mod 3 Adder).

$$X_N = x_1 + (2^n+3)\left|k_N\left(x_2 - \overbrace{(3C_n+\hat{x}_1)}^{x_1}\right)\right|_{2^n-3} \tag{25}$$

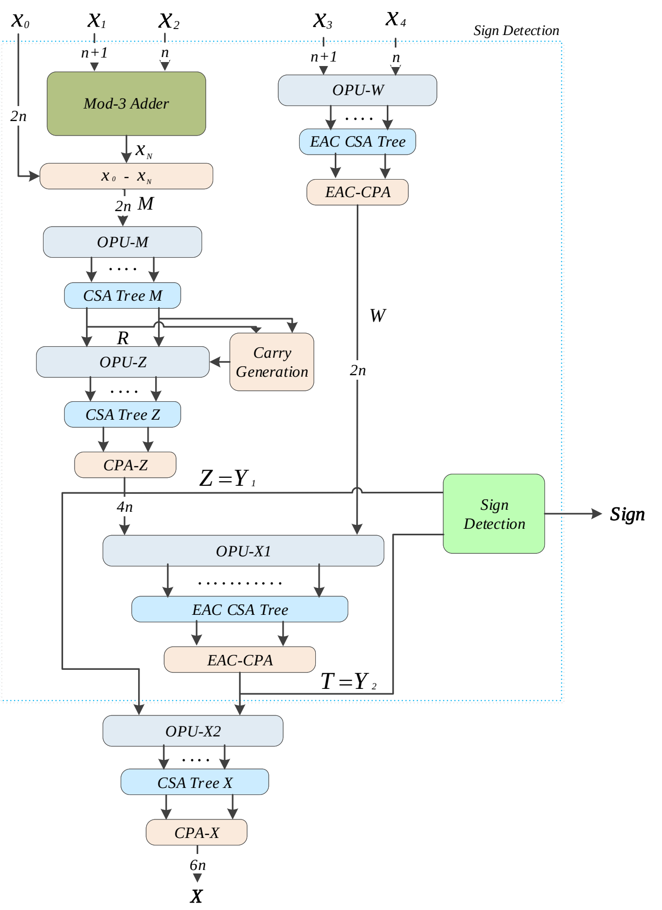

Figure 2. Proposed multifunction unit structure: reverse converter and sign detector

4.2. Reverse converter for set $\mathbf{W} = \{\mathbf{2^n + 1, 2^n - 1}\}$

The main reverse conversion formula for moduli set $\{2^n + 1, 2^n - 1\}$ is as follows [26]:

$$W = x_3 + (2^n + 1)|K_W(x_4 - x_3)|_{2^n-1} \tag{26}$$

Therefore, when considering $K_W = 2^{n-1}$ and $m_1|Z|_{m_2} = |m_1 Z|_{m_1 m_2}$ [1], we have:

$$\begin{aligned}
&W = |x_3 + K_W(2^n + 1)(x_4 - x_3)|_{2^2n-1} = \\
&|2^{n-1}(2^n + 1)x_4 + x_3 - 2^{2n-1}x_3 - 2^{n-1}x_3|_{2^{2n}-1} = \\
&|2^{n-1}(2^n + 1)x_4 + 2^{2n}x_3 - 2^{2n-1}x_3 - 2^{n-1}x_3|_{2^{2n}-1} = \\
&|2^{n-1}(2^n + 1)x_4 + 2^{2n-1}x_3 - 2^{n-1}x_3|_{2^{2n}-1} = \\
&|2^{n-1}(2^n + 1)x_4 + 2^{2n-1}x_3 + 2^{n-1}\bar{x}_3 + (2^{n-1} - 1)|_{2^{2n}-1}
\end{aligned} \tag{27}$$

4.3. Reverse converter for set $\mathbf{Z} = \{(\mathbf{2^{2n} - 9}), \mathbf{2^{2n}}\}$

The combination of the reverse converter of moduli set $\{2^n + 3, 2^n - 3\}$ (i.e., composite modulo $2^{2n} - 9$) and 2^{2n} can be done as follows:

$$Z = X_N + (2^{2n} - 9)|K_Z(x_0 - X_N)|_{2^{2n}} = X_N + (2^{2n} - 9)R \tag{28}$$

Next, multiplicative inverse K_Z can be achieved as follows [1]:

$$|K_Z \times (2^{2n} - 9)|_{2^{2n}} = 1 \to k_Z = |(2^{2n} - 9)^{-1}|_{2^{2n}} \to k_Z = |(-9)^{-1}|_{2^{2n}} \tag{29}$$

Therefore,

$$K_Z = \begin{cases} -(56\rho + 1) & if\ n = 3\alpha \\ -(56\rho + 1) & if\ n = 3\alpha + 1 \\ -(2^{2n-1} + 56\rho + 1) & if\ n = 3\alpha + 2 \end{cases} \tag{30}$$

where

$$\rho = \frac{2^{6\alpha} - 1}{2^6 - 1} \tag{31}$$

For example, by considering $n = 3\alpha$, we have:

$$K_Z = -(56\rho + 1) \tag{32}$$

Then, substituting (32) in (28) results in the following:

$$\begin{aligned}
M = (x_0 - X_N) \to R = |K_Z \times M|_{2^{2n}} = |(-(56\rho + 1)) \times M|_{2^{2n}} \\
= |(-(64 - 8)\rho M - M)|_{2^{2n}} = |-64\rho M + 8\rho M - M|_{2^{2n}}
\end{aligned} \tag{33}$$

The negative values can be computed when considering $\bar{M} = (2^{2n} - 1) - M$, which lead to $-M = \bar{M} - (2^{2n} - 1)$.

Therefore,

$$R = |-64\rho M + 8\rho M - M|_{2^{2n}} =$$
$$|64\rho(\bar{M} - 2^{2n} + 1) + 8\rho M + (\bar{M} - 2^{2n} + 1)|_{2^{2n}}$$
$$|8\rho M + 64\rho\bar{M} + \bar{M} + 64\rho - 2^{2n} \times 64\rho + 1 - 2^{2n}|_{2^{2n}} \rightarrow$$
$$R = |8\rho M + (64\rho + 1)\bar{M} + (64\rho + 1) + \delta_0|_{2^{2n}} \tag{34}$$

When $n = 3\alpha + 2$, we have $\delta_0 = -2^{2n-1}M$; otherwise, it equals zero. Therefore,

$$|\delta_0|_{2^{2n}} = |-2^{2n-1}M|_{2^{2n}} = |2^{2n-1}(\bar{M} - 2^2 n + 1)|_{2^{2n}}$$
$$= |2^{2n-1}\bar{M} + 2^{2n-1}|_{2^{2n}} = (\bar{M}_0 \overbrace{0...0}^{2n-1}) + (1\,\overbrace{0...0}^{2n-1}) \tag{35}$$

and results in

$$R = |8\rho M + (2^{2n-1} + 64\rho + 1)\bar{M} + (2^{2n-1} + 64\rho + 1)|_{2^{2n}} \tag{36}$$

Assuming that $n = 3, \rho = 1$, and then:

$$R = |8\rho(M_{2n-1}...M_0) + (64\rho + 1)(\bar{M}_{2n-1}...\bar{M}_0) + (64\rho + 1)|_{2^{2n}} =$$
$$|(\overbrace{M_{2n-4}...M_0}^{2n-3}\overbrace{0...0}^{3}) + (\overbrace{\bar{M}_{2n-8}...\bar{M}_{2n-1}}^{2n-6}\overbrace{0...0}^{6}) + (\overbrace{\bar{M}_{2n-1}...\bar{M}_0}^{2n}) + (64\rho + 1)|_{2^{2n}} \tag{37}$$

Therefore, according to (28), we have:

$$Z = X_N + (2^{2n} - 9)R = (\overbrace{0...0}^{2n}\overbrace{x_{N,2n-1}...x_{N,0}}^{2n}) + (\overbrace{R_{2n-1}...R_0}^{2n}\overbrace{0...0}^{2n})$$
$$- (\overbrace{0...0}^{2n-3}\overbrace{R_{2n-1}...R_0}^{2n}\overbrace{000}^{3}) - (\overbrace{0...0}^{2n}\overbrace{R_{2n-1}...R_0}^{2n}) \tag{38}$$

Now, (38) can be rewritten as follows:

$$Z = (\overbrace{0......0}^{2n}\overbrace{x_{N,2n-1}...x_{N,0}}^{2n}) + (\overbrace{R_{2n-1}...R_0}^{2n}\overbrace{\bar{R}_{2n-1}...\bar{R}_0}^{2n}) + (\overbrace{1...1}^{2n-3}\overbrace{\bar{R}_{2n-1}...\bar{R}_0}^{2n}\overbrace{111}^{3}) +$$
$$(\overbrace{1...1}^{2n}\overbrace{0...0}^{2n}) + 2 \tag{39}$$

C represents the carry output of non-modular adder $R_c + R_S$, which is used to avoid the deployment of the carry-propagate adder; this carry is obtained by a specific parallel prefix adder. Finally, we have:

$$Z = X_N + (2^{2n} - 9)(R_S + R_C - 2^{2n}C) =$$
$$X_N + 2^{2n}R_C + 2^{2n}R_S - 2^{4n}C - 2^3R_C - 2^3R_S + 2^{2n+3}C - R_S - R_C + 2^{2n}C =$$
$$X_N + 2^{2n}R_C + 2^{2n}R_S + 2^3(\bar{R}_C - 2^{2n} + 1) + 2^3(\bar{R}_S - 2^{2n} + 1) + (\bar{R}_C - 2^{2n} + 1) +$$
$$(\bar{R}_S - 2^{2n} + 1) - (2^{4n} - 2^{2n+3} - 2^2)C = X_N + 2^{2n}R_C + 2^{2n}R_S +$$
$$2^3\bar{R}_C + 2^3\bar{R}_S + \bar{R}_C + \bar{R}_S - \left(2^{2n+4} + 2^{2n+1} - 2^4 - 2\right) - \left(2^{4n} - 2^{2n+3} - 2^{2n}\right)C \tag{40}$$

Therefore,

$$Z = (\overbrace{0...0}^{2n}\overbrace{x_{N,2n-1}...x_{N,0}}^{2n}) + (\overbrace{R_{C,2n-1}...R_{C,0}}^{2n}\overbrace{0...0}^{2n}) + (\overbrace{R_{S,2n-1}...R_{S,0}}^{2n}\overbrace{0...0}^{2n}) - \\ (\overbrace{0...0}^{2n-3}\overbrace{R_{C,2n-1}...R_{C,0}}^{2n}\overbrace{000}^{3}) - (\overbrace{0...0}^{2n-3}\overbrace{R_{S,2n-1}...R_{S,0}}^{2n}\overbrace{000}^{3}) - (\overbrace{0...0}^{2n}\overbrace{R_{C,2n-1}...R_{C,0}}^{2n}) - \\ (\overbrace{0...0}^{2n}\overbrace{R_{S,2n-1}...R_{S,0}}^{2n}) - (2^{2n+4} + 2^{2n+1} - 2^4 - 2) - (2^{4n} - 2^{2n+3} - 2^2)C \quad (41)$$

And consequently,

$$Z = (\overbrace{0......0}^{2n}\overbrace{x_{N,2n-1}...x_{N,0}}^{2n}) + (\overbrace{R_{C,2n-1}...R_{C,0}}^{2n}\overbrace{\bar{R}_{C,2n-1}...\bar{R}_{C,0}}^{2n}) + \\ (\overbrace{R_{S,2n-1}...R_{S,0}}^{2n}\overbrace{\bar{R}_{S,2n-1}...\bar{R}_{S,0}}^{2n}) + (\overbrace{1...1}^{2n-3}\overbrace{\bar{R}_{C,2n-1}...\bar{R}_{C,0}}^{2n}\overbrace{111}^{3}) + \\ (\overbrace{1...1}^{2n-3}\overbrace{\bar{R}_{S,2n-1}...\bar{R}_{S,0}}^{2n}\overbrace{111}^{3}) - (2^{2n+4} + 2^{2n+1} - 4 - 2^4 - 2) \\ - (2^{4n} - 2^{2n+3} - 2^2)C \quad (42)$$

4.4. Reverse converter for set $\mathbf{W} = \{\mathbf{2^{2n}(2^{2n} - 9), (2^{2n} - 1)}\}$

At the final stage, the results of the previous stages should be combined as follows:

$$X = Z + 2^{2n}(2^{2n} - 9)|K_X \times (W - Z)|_{2^{2n}-1} \quad (43)$$

The multiplicative inverse of K_X can be calculated as follows [1]:

$$|k_X \times 2^{2n}(2^{2n} - 9)|_{2^{2n}-1} = 1 \to k_X = |(2^{2n}(2^n + 1))^{-1}|_{2^n-1} \to k_X = -2^{2n-3} \quad (44)$$

This can be proven as below:

$$|k_X \times (2^{2n}(2^{2n} - 9))|_{2^2n-1} = \\ |(-2^{2n-3}) \times (2^{2n}(2^{2n} - 9))|_{2^2n-1} = \\ |(-2^{2n-3}) \times (-2^3)|_{2^{2n}-1} = |2^{2n}|_{2^{2n}-1} = 1 \quad (45)$$

Now, the final value (i.e., X) can be obtained as follows:

$$X = \left(\overbrace{0...0}^{2n}\overbrace{Z_{4n-1}...Z_0}^{4n}\right) + 2^{2n}(2^{2n} - 9)|2^{2n-3}(Z - W)|_{2^{2n}-1} \quad (46)$$

The internal modulo $2^{2n} - 1$ operation can be simplified as follows:

$$T = |2^{2n-3}(Z - W)|_{2^{2n}-1} = \\ |2^{2n-3}((Z_{4n-1}...Z_{2n})2^{2n} + (Z_{2n-1}...Z_0) - (W_{2n-1}...W_0))|_{2^{2n}-1} =$$

$$|((Z_{2n+2}Z_{2n+1}Z_{2n}Z_{4n-1}\ldots\ldots Z_{2n+3}) + (Z_2Z_1Z_0Z_{2n-1}\ldots\ldots Z_3) + (\overline{W_2}\ \overline{W_1}\ \overline{W_0}\ \overline{W_{2n-1}}\ldots\ \overline{W_3}))|_{2^{2n}-1} \tag{47}$$

Therefore,

$$X = (\overbrace{0...0}^{2n}\overbrace{Z_{4n-1}...Z_0}^{4n}) + (2^{4n} - 2^{2n+3} - 2^{2n})T =$$
$$(\overbrace{0...0}^{2n}\overbrace{Z_{4n-1}...Z_0}^{2n}) + (\overbrace{T_{2n-1}...T_0}^{2n}\overbrace{0...0}^{4n}) - (\overbrace{0...0}^{2n-3}\overbrace{T_{2n-1}...T_0}^{2n}\overbrace{0...0}^{2n+3}) - (\overbrace{0...0}^{2n}\overbrace{T_{2n-1}...T_0}^{2n}\overbrace{0...0}^{2n}) \rightarrow$$
$$= (\overbrace{T_{2n-1}...T_0}^{2n}\overbrace{Z_{4n-1}...Z_0}^{4n}) + (\overbrace{1...1}^{2n-3}\overbrace{\bar{T}_{2n-1}...\bar{T}_0}^{2n}\overbrace{1...1}^{2n+3}) + (\overbrace{1...1}^{2n}\overbrace{\bar{T}_{2n-1}...\bar{T}_0}^{2n}\overbrace{1...1}^{2n}) + 2 \tag{48}$$

The reverse converter structure is presented in Figure 2. It should be noted that the details regarding each part can be found in [1].

5. Sign detection

The residue number system has some difficult operations, such as division, overflow detection, comparison, and sign detection that have made this residue number system not be universally applicable. Hence, an efficient method of sign detection for moduli set $\{2^{2n}, 2^n+1, 2^n-1, 2^n+3, 2^n-3\}$ is presented in this paper. The idea of this paper is elicited from [26], which will be discussed in the following paragraphs. In general, the residue number system is designed for unsigned numbers, the output of the reverse converter represents an unsigned value, and the output needs to be corrected to use this system for signed numbers. Early in the conventional method, the sign detection was approached by comparing the output value with half of dynamic range $M/2$, which is shown in Figure 3.

In this section, our focus is on the efficient and operational design of a unit for detecting the sign and correcting the output of the reverse converter with better performance as compared to using the comparator and multiplexer in the output of reverse converter. Here, after specifying the sign (if necessary), we will correct the output so that the output of the reverse converter will display the correct two's complement value. In order to detect the sign, the method in [26] has been applied on moduli set $\{2^{2n}, 2^n+3, 2^n-3, 2^n+1, 2^n-1\}$ in this paper. Figure 4 shows the range of positive and negative numbers in moduli set $\{2^{2n}, 2^n+3, 2^n-3, 2^n+1, 2^n-1\}$. As it is shown, although the numbers within range $[0, 2^{6n-1} - (10 \times 2^{4n-1} - 9 \times 2^{2n-1}))$ have bits with zero values and the sign is positive, it is also a zero-valued bit within range $[2^{6n-1} - (10 \times 2^{4n-1} - 9 \times 2^{2n-1}), 2^{6n-1})$. However, these numbers are yet negative in range $[2^{6n-1}, 2^6n - (10 \times 2^{4n} - 9 \times 2^{2n}))$, and they always have bits with one value.

The principles of a reverse converter permit us to achieve an effective method of detecting the sign by using formulas. We intend to detect the sign by utilizing formulas and reverse converter equations.

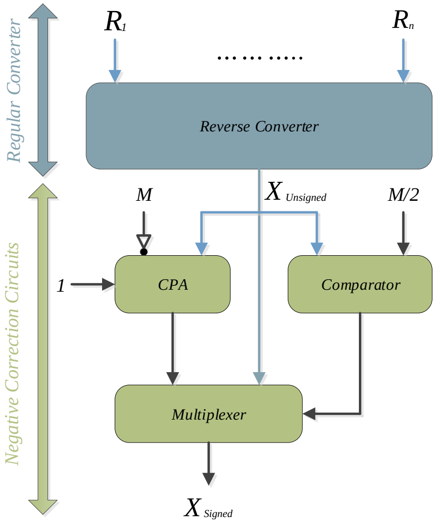

Figure 3. Traditional reverse converter with signed [26]

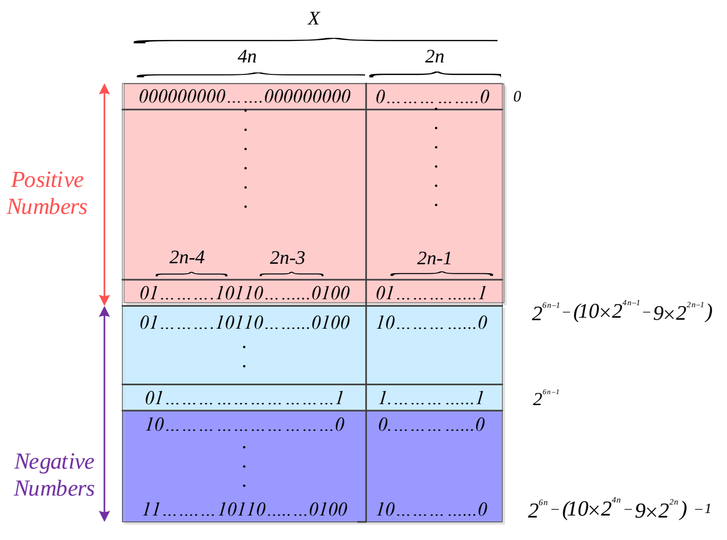

Figure 4. Dynamic range distribution

It is noticeable that one approach to obtaining the sign is to consider a dynamic range where $X \in [0, \frac{M}{2})$ or $X \in [\frac{M}{2}, M)$ regarding the range of positive and negative numbers in a residue number system. When $Y_2 \neq \frac{M_2 - 1}{2}$, the range of numbers considered for Y_2 are divided into two parts, which are positive or negative concerning the range of the numbers in the residue number system. Therefore, if the bit with a value of Y_2 equals one, X is negative, and if the bit with a value of Y_2 equals zero, X is positive. However, when $Y_2 = \frac{M_2 - 1}{2}$, we need to examine Y_1 in order to detect the sign. Therefore, when $Y_1 \geqslant \frac{M_1}{2}$, X is negative; otherwise, X is positive.

$$W = x_3 + p_3|K_W(x_4 - x_3)|_{p_4} \tag{49}$$

$$Z = \overbrace{x_1 + p_1|K_Z(x_2 - x_1)|_{p_2}}^{Y_1} \tag{50}$$

$$X = \overbrace{Z}^{Y_1} + \overbrace{p_1 p_2}^{M_1} | \overbrace{k_X(Z - W)|_{p_3 p_4}}^{Y_2} = Y_1 + Y_2 M_1 \tag{51}$$

$$M_1 = 2^{2n}(2^{2n} - 9) \tag{52}$$

$$M_2 = (2^{2n} - 1) \tag{53}$$

Therefore, the following equation is presented for the sign detection of X.

$$sign = (Y_{2,2n-1} \vee L) \rightarrow sign = \begin{cases} 0 & + \\ 1 & - \end{cases} \tag{54}$$

$$L = (L_1 \wedge (L_2 \vee L_3))) \tag{55}$$

$$L_1 = (\bar{Y}_{2,2n-1} \wedge Y_{2,2n-2} ... \wedge Y_{2,0}) \tag{56}$$

$$L_2 = (Y_{1,4n-1}) \tag{57}$$

$$L_3 = (\bar{Y}_{1,4n-1} \wedge Y_{1,4n-2} \wedge ... \wedge Y_{1,2n+3} \wedge (Y_{1,2n+2} \vee (\bar{Y}_{1,2n+2} \wedge Y_{1,2n+1} \wedge ... \wedge Y_{1,2n-1})) \tag{58}$$

Now, by specifying the sign and knowing whether the sign is negative, the output of the reverse converter needs to be corrected. If the sign-detection circuit produces a value of zero, this indicates that the number is positive and that the output does not need to be corrected. However, if the sign-detection circuit produces a value of one, this indicates that the number is negative and that there is a need for correcting the output. If $\hat{X}$ indicates the signed output, the value of $\hat{X}$ is obtained based on the following relationship:

$$\hat{X} = X - M \tag{59}$$

$$M = 2^{2n}(2^{2n} - 1)(2^{2n} - 9) = 2^{2n}(2^{4n} - 10 \times 2^{2n} + 9) = 2^{6n} - (10 \times 2^{4n} - 9 \times 2^{2n}) \tag{60}$$

The two's complement of M is represented as $6n$ bits.

$$\begin{aligned} &2^{6n} - M = 2^{6n} - (2^{6n} - (10 \times 2^{4n} - 9 \times 2^{2n})) = \\ &10 \times 2^{4n} - 9 \times 2^{2n} = 2^{2n}(10 \times 2^{2n} - 9) \end{aligned} \tag{61}$$

Now, for correcting the output of the reverse converter (provided that the number is negative), X should be added to $(10 \times 2^{2n} - 9)$. Hence, the sign detection and correcting the value of X is applicable (if necessary) by using the hardware presented in Figure 5.

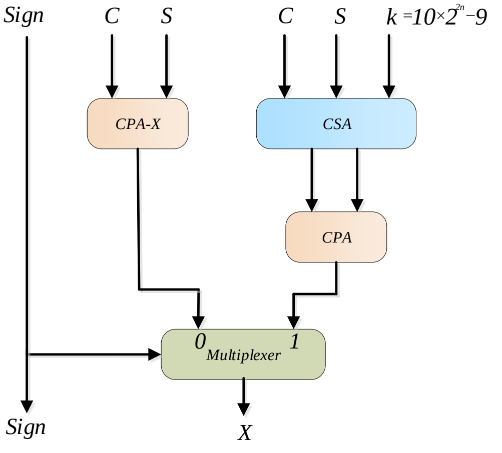

Figure 5. Sign detection and correction circuit

•Numerical examples

We provide an example to examine the operation of the proposed circuit. Suppose we intend to obtain the sign of an RNS number based on moduli set $\{2^{2n}, 2^n + 3, 2^n - 3, 2^n + 1, 2^n - 1\}$. For $n = 3$, the regarded moduli set takes a value of $\{64, 11, 5, 7, 9\}$ and the dynamic range equals 221760. Suppose that we must change weightless residual number $(32, 0, 0, 0, 0)$ (which is equivalent to a number with a value of 110880) into a binary weighted status. Therefore, the obtained values within a range of 0 to 110879 indicate the positive numbers, and the values within a range of 110880 to 221759 indicate the negative numbers. The 110880 value is located in the lower half of the dynamic range; thus, it is a negative number. Therefore, we must receive 110880–221760 = −110880 as the output of the reverse converter.

Consequently, the proposed structure of the two's complement produces the negative number of −110880, which is equivalent to 151264 instead of 110880 (for mentioned moduli sets $k = 2n = 6$ and $p = 2n = 6$, and n is odd).

$$X = 110880 \xrightarrow{RNS} (32, 0, 0, 0, 0)$$

Therefore, X_N has the following values:

$X_N = 0$

$W = 0 = (000000)$

$$Z = Y_1 = 1760 = (011011100000)$$

$$X = Z + 2^{2n}(2^{2n}-9)\overbrace{|2^{2n-3}(Z-W)|_{2^{2n}-1}}^{Y_2} = 1760 + 3520|\overbrace{14080|_{63}}^{Y_2} \rightarrow Y_2 = 31$$

$$\frac{M_1}{2} = 1760$$

$$\frac{M_2 - 1}{2} = 31$$

Given that $Y_2 = 31$, the sign of X is dependent on Y_1 , and while $Y_1 \geqslant \frac{M_1}{2}$ exists, the sign is negative; otherwise, the sign is positive. Hence, the following will be performed for investigating $Y_1 \geqslant \frac{M_1}{2}$:

$$L_1 = (\bar{Y}_{2,2n-1} \wedge Y_{2,2n-2} ... \wedge Y_{2,0}) = 1$$

$$L_2 = (Y_{1,P+K-1}) = 0$$

$$L_3 = (\bar{Y}_{1,P+K-1} \wedge Y_{1,P+K-2} \wedge ... \wedge Y_{1,K+3}$$

$$\wedge (Y_{1,K+2} \vee (\bar{Y}_{1,K+2} \wedge Y_{1,K+1} \wedge ... \wedge Y_{1,K-1})) = 1$$

$$L = (L_1 \wedge (L_2 \vee L_3))) = 1$$

$$Sign = (Y_{2,2n-1} \vee L) \rightarrow Sign = (0 \vee 1) = 1$$

If the sign is negative, we need to correct the output of the reverse converter and display the correct one. Since there is a need for output correction, the outcome of X is added to the value of $K = (10 \times 2^{2n} - 9)$; if the output sign is equal to the value of one (i.e., the number is negative) and according to Figure 5, the corrected value is inserted at the output of the multiplexer.

$$X = 100100111011100000$$

It can be concluded from the above equations that the value of the sign equals one. Therefore, the number is negative, the output must be corrected, and the corrected value is placed at the output of the multiplexer.

$$X = 110880, \;\; M = 221760 \rightarrow \hat{X} = X - M = 110880 - 221760 = -110880$$

$$\hat{X} = -110880 = -(011011000100100000)$$

$$\xrightarrow{2's\ Complement} (100100111011100000) = 151264$$

Accordingly, both the original value (110880) and the corrected value (151264) are placed at the input of the multiplexer; since the sign circuit equals one, the input of a multiplexer is placed at the output.

6. Performance evaluation

We compared the proposed design with the hardware architecture of [1] using both theoretical and experimental analysis as shown in Tables 1 and 2. In a theoretical comparison, we evaluate the time delay and the area of each design based on its basic components, which include the full adder (FA), half adder (HA), multiplexer (MUX), and simple gates (G).

Table 1

Hardware complexity comparison

Component	Delay	Area
W	N/A (it is not on the critical delay path)	$(2n+2)A_{FA} + (2n-2)A_{HA} + (4n-1)A_G$
X_N	$(3n + \lceil \log 2n \rceil + 2)D_{FA}$	$(2n^2 + 2n + 2)A_{F.A} + (n)A_{HA}$
CPA – M Generate	$(2n)D_{FA}$	$(2n)A_{FA}$
OPU-M	$(1)D_G$	$(4n-5)A_G$
CSA Tree M	$(3)D_{FA}$	$(6n)A_{FA}$
Carry Generation	$(3 + 2\lceil \log n \rceil)D_G$	$(2^{n+1} + 2^n + 2n - 3)A_G$
OPU-Z	$(1)D_G$	$(8n)A_G$
CSA Tree Z	$(4)D_{FA}$	$(20n)A_{FA}$
CPA - Z	$(4n)D_{FA}$	$(4n)A_{FA}$
OPU-X1	$(1)D_G$	$(2n)A_G$
EAC CSA Tree	$(1)D_{FA}$	$(2n)A_{FA}$
EAC CPA	$(4n)D_{FA}$	$(4n)A_{FA}$
Sign Detection	$(\lceil \log 4n + 1 \rceil)D_G$	$(4n+3)A_G$
Total Sign	$(13n + \lceil \log 2n \rceil + 10)D_{FA} + (\lceil \log 4n + 1 \rceil + 2\lceil \log n \rceil + 6)D_G$	$(2n^2 + 42n + 2)A_{FA} + (3n - 2)A_{HA} + (2^{n+1} + 2^n + 24n - 6)A_G$
OPU-X2	$(1)D_G$	$(4n)A_G$
CSA Tree X	$(2)D_{FA}$	$(12n)A_{FA}$
CPA-X	$(6n)D_{FA}$	$(6n)A_{FA}$
Total Reverse And Sign	$(19n + \lceil \log 2n \rceil + 12)D_{FA} + (\lceil \log 6n \rceil + 2\lceil \log n \rceil + 9)D_G$	$(2n^2 + 60n + 2)A_{FA} + (3n - 2)A_{HA} + (2^{n+1} + 2^n + 30n - 4)A_G$

Therefore, the time delay and area of each section are considered to be D_{FA}, D_{HA}, D_{MUX}, D_G, A_{FA}, A_{HA}, A_{MUX}, and A_G, respectively. The time delay and the level of logic gate XOR are considered $2D_G$ and $3A_G$, respectively. It should be noted that logarithmic terms refer to the number of levels or to the expression of CSA trees. Since some parts of the designed circuits are identical to the presented original reference in [1] and contain no particular variations, they are regarded with the values of original reference in terms of a hardware analysis. As shown in Figure 2, the rotational operation, shift, and the one's complement are performed in the internal structure of the OPU. The area of n-bit CSA is equal to n full adders, and its time delay is equal to one full adder; meanwhile, the area and delay of n-bit CPA is equal to n full adders.

Table 2

Experimental result comparison

Bits	n	3	4	5
Proposed signed reverse converter and sign detector	Power	12.2674	17.0864	25.2444
	Area	18958	24395	34424
	Delay	20.31	25.55	31.98
Proposed reverse converter and sign detector	Power	6.1881	8.5580	18.2859
	Area	10953	14523	24039
	Delay	14.61	18.06	26.15
Proposed reverse converter	Power	9.5837	13.2400	21.3554
	Area	15035	21371	31121
	Delay	19.31	24.53	30.98
[26] with sign output using comparator	Power	11.0941	16.4309	27.3186
	Area	16724	22353	32545
	Delay	21.61	27.24	32.91
Reverse converter of [26]	Power	9.3667	15.7674	26.4297
	Area	14150	20676	30449
	Delay	21.49	27.16	32.83

Since the main objective of this paper is to propose an approach to extract the sign from the inside of the reverse converter, we use the comparator and reverse converter of [1] to extract the sign and compare it with the extracted sign in the proposed approach. It should be noted that, in [1], two two-stage and three-stage reverse converters are designed in a way that are approximately equivalent in terms of time delay, power, and level. We thus make the comparison with the two-stage reverse converter in [1]. All in all, the area and time delay for the reference reverse converter are as follows:

$A_{Sign\_Regular} = A_{Reverse\_Converter} + A_{Comparator} + A_{CPA} + A_{MUX2\times1} =$

$A_{Reverse\_Converter} = ((3(6n) - 2)A_{AND} + (6n-1)A_{OR} + (6n-1)A_{XNOR})+$

$(6n)A_{FA} + (6n)A_{MUX2\times1}$

$A_{Reverse\_Converter} = (8n^2 + 18n + 4)A_{FA} + (7n-2)A_{HA} + (4n)A_{MUX} + (4n-1)A_G$

$D_{Sign\_Regular} = D_{Reverse\_Converter} + D_{CPA} + D_{MUX2\times1} =$

$D_{Reverse\_Converter} + (6n)D_{FA} + D_{MUX2\times1}$

$D_{Reverse\_Converter} = (11n + 3\lceil\log n\rceil + 4)D_{FA} + D_{MUX2\times1}$

The level and time delay in the sign detection in the proposed approach equals the following:

$$A_{Sign\_Proposed} = A_{X_N} + A_W + A_{Y_1} + A_{Y_2} + A_{Sign\_Detecton}$$
$$D_{Sign\_Proposed} = D_{X_N} + D_W + D_{Y_1} + D_{Y_2} + D_{Sign\_Detecton}$$

There is also $D_{Sign\_Detecton} = D_{Y_1} + D_{Y_2}$; however, given that the value of D_{Y_1} overlaps the computation of Y_2 and that its time delay is trivial compared to the computation of Y_2, we neglect it. Therefore, the area and time delay for the proposed reverse converter can be calculated as follows:

$$A_{Sign\_Proposed} = (2n^2 + 42n + 2)A_{FA} + (3n - 2)A_{HA} + (2^{n+1} + 2^n + 24n - 6)A_G$$
$$D_{Sign\_Proposed} = (13n + \lceil \log 2n \rceil + 10)D_{FA} + (\lceil \log 4n + 1 \rceil + 2\lceil \log n \rceil + 6)D_G$$

For our experimental evaluation, we used Synopsys Design Compiler software to achieve the estimated delay, area, and power-consumption using 180-nm technology. These results are presented in Table 2. In addition, the power-delay-product (PDP) evaluation is performed in Figure 6. It can be seen that the proposed multifunction architecture (i.e., the signed reverse converter plus sign detector) has better circuit parameters than a conventional reverse converter with a sign output that uses a comparator to detect the sign and then correct the output.

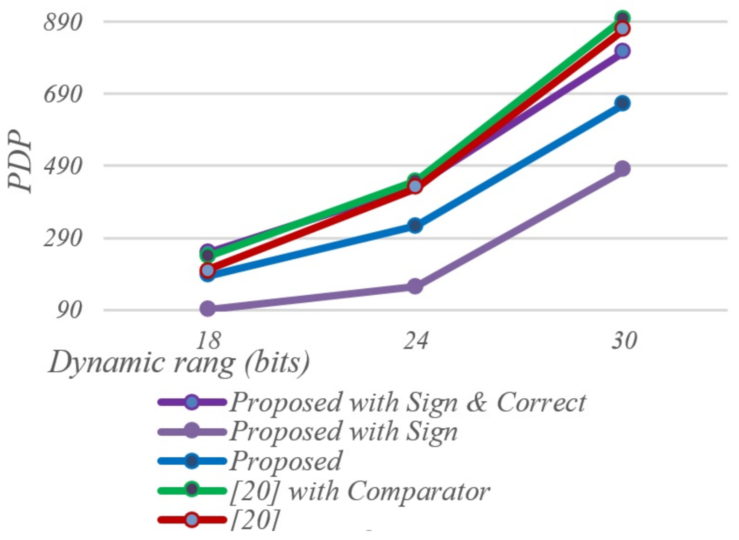

Figure 6. PDP Evaluation

7. Conclusion

This paper presents a signed reverse converter for moduli set $\{2^{2n},\ 2^n + 3,\ 2^n - 3,\ 2^n + 1, 2^n - 1\}$ with the extraction of the sign from the reverse converter and then use it to correct the sign of the reverse converter output. In addition, we also presented the first sign-detection unit for moduli set $\left\{2^{2n}, 2^n + 3, 2^n - 3, 2^n + 1, 2^n - 1\right\}$. The proposed circuits have inserted the sign-handling mechanism to this five-moduli RNS, making it practical for real applications. However, other important features such as scaling and magnitude comparison are needed in RNS systems. The integration of the RNS magnitude comparison and scaling circuits for moduli set $\left\{2^{2n}, 2^n + 3, 2^n - 3, 2^n + 1, 2^n - 1\right\}$

to a multifunctional unit is an open problem that requires further research. Having a complete multifunctional unit that can perform all of the difficult RNS operations (including reverse conversion, sign detection, scaling, and magnitude comparison) can push the RNS forward to modern applications like deep learning.

References

[1] Ahmadifar H., Jaberipur G.: A New Residue Number System with 5-Moduli Set: $\{2^{2q}, 2^{2q} \pm 3, 2^{2q} \pm 1\}$, *The Computer Journal*, vol. 58(7), pp. 1548–1565, 2014.

[2] Chang C.H., Molahosseini A.S., Zarandi A.A.E., Tay T.F.: Residue Number Systems: A New Paradigm to Datapath Optimization for Low-Power and High-Performance Digital Signal Processing Applications, *IEEE Circuits and Systems Magazine*, vol. 15(4), pp. 26–44, 2015.

[3] Didier L.S., Rivaille P.Y.: A Generalization of a Fast RNS Conversion for a New 4-Modulus Base, *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 56(1), pp. 46–50, 2009. doi: 10.1109/TCSII.2008.2010157.

[4] Garner H.L.: The residue number system, *IRE Transaction Electric Computer*, vol. 8(2), pp. 140–147, 1959.

[5] Hiasat A., Sousa L.: On the Design of RNS Inter-Modulo Processing Units for the Arithmetic-Friendly Moduli Sets $\{2^{n+k}, 2^n - 1, 2^{n+1} - 1\}$, *The Computer Journal*, vol. 62(2), pp. 292–300, 2018.

[6] Hiasat A., Sousa L.: Sign Identifier for the Enhanced Three Moduli Set $\{2^{n+k}, 2^n - 1, 2^{n+1} - 1\}$, *Journal of Signal Processing Systems*, vol. 91(8), pp. 953–961, 2019.

[7] Hiasat A., Sweidan A.: Residue number system to binary converter for the moduli set $(2^{n-1}, 2^n - 1, 2^n + 1)$, *Journal of Systems Architecture*, vol. 49(1–2), pp. 53–58, 2003.

[8] Jaberipur G., Ahmadifar H.: A ROM-less reverse RNS converter for moduli set $\{2^q \pm 1, 2^q \pm 3\}$, *IET Computers & Digital Techniques*, vol. 8(1), pp. 11–22, 2013.

[9] Kumar R., Mishra R.A.: Design of Efficient Sign Detector for Moduli Set $\{2^n - 1, 2^n, 2^n + 1\}$ in Residue Number System. In: *2019 International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, pp. 1–5, 2019.

[10] Kumar S., Chang C.H.: A VLSI-efficient signed magnitude comparator for $\{2^n - 1, 2^n, 2^{n+1} - 1\}$ RNS. In: *IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1966–1969, IEEE, 2016.

[11] Kumar S., Chang C.H.: A Scaling-Assisted Signed Integer Comparator for the Balanced Five-Moduli Set RNS $\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} -, 2^{2-1} - 1\}$. In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25(12), pp. 3521–3533, 2017.

[12] Matutino P.M., Chaves R., Sousa L.: Binary-to-RNS Conversion Units for moduli $\{2^n \pm 3\}$. In: *2011 14th Euromicro Conference on Digital System Design*, pp. 460–467, IEEE, 2011.

[13] Mohan P.V.A.: 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 II: Express Briefs*, vol. 54(9), pp. 775–779, 2007.

[14] Mohan P.V.A.: New reverse converters for the moduli set $\{2^n-1, 2^n+1, 2^n-3, 2^n+3\}$, *AEU – International Journal of Electronics and Communications*, vol. 62(9), pp. 643–658, 2008.

[15] Mohan P.V.A.: *Residue Number Systems: Theory and Applications*, Springer, 2016.

[16] Mohan P.V.A., Premkumar A.B.: 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.

[17] Molahosseini A.S., Navi K., Dadkhah C., Kavehei O., Timarchi S.: Efficient Reverse Converter Designs for the New 4-Moduli Sets $\{2^n-1, 2^n, 2^n+1, 2^{2n+1}-1\}$ and$\{2^n-1, 2^n+1, 2^{2n}, 2^{2n}+1\}$ Based on New CRTs, *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57(4), pp. 823–835, 2010.

[18] Molahosseini A.S., Zarandi A.A.E., Martins P., Sousa L.: A Multifunctional Unit for Designing Efficient RNS-Based Datapaths, *IEEE Access*, vol. 5, pp. 25972–25986, 2017.

[19] Moons B., Bankman D., Verhelst M.: *Embedded Deep Learning: Algorithms, Architectures and Circuits for Always-on Neural Network Processing*, Springer, 2019.

[20] Navi K., Molahosseini A.S., Esmaeildoust M.: How to Teach Residue Number System to Computer Scientists and Engineers, *IEEE Transactions on Education*, vol. 54(1), pp. 156–163, 2011.

[21] Saračević M., Adamović S., Miškovic V., Maček N., Šarac M.: A novel approach to steganography based on the properties of Catalan numbers and Dyck words, *Future Generation Computer Systems*, vol. 100, pp. 186–197, 2019.

[22] Sousa L., Antao S., Martins P.: Combining Residue Arithmetic to Design Efficient Cryptographic Circuits and Systems, *IEEE Circuits and Systems Magazine*, vol. 16(4), pp. 6–32, 2016.

[23] Sousa L., Martins P.: Sign Detection and Number Comparison on RNS 3-Moduli Sets $\{2^n-1, 2^{n+x}, 2^n+1\}$, *Circuits, Systems, and Signal Processing*, vol. 36(3), pp. 1224–1246, 2016.

[24] Stanimirović P., Krtolica P., Saračević M., Mašović S.: Decomposition of Catalan Numbers and Convex Polygon Triangulations, *International Journal of Computer Mathematics*, vol. 91(6), pp. 1315–1328, 2014.

[25] Wang Y., Song X., Aboulhamid M., Shen H.: Adder Based Residue to Binary Numbers Converters for $\{2^n-1, 2^n, 2^n+1\}$, *IEEE Transactions on Signal Processing*, vol. 50(7), pp. 1772–1779, 2002.

[26] Zarandi A.A.E., Molahosseini A.S., Sousa L., Hosseinzadeh M.: An Efficient Component for Designing Signed Reverse Converters for a Class of RNS Moduli Sets of Composite Form $\{2^k, 2^P - 1\}$, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25(1), pp. 48–59, 2017.

[27] Zheng X., Wang B., Zhou C., Wei X., Zhang Q.: Parallel DNA Arithmetic Operation With One Error Detection Based on 3-Moduli Set, *IEEE Transactions on NanoBioscience*, vol. 15(5), pp. 499–507, 2016.

Affiliations

Mohsen Mojahed
Islamic Azad University, Department of Computer Engineering, Kerman Branch, Kerman, Iran, mojahed@iauk.ac.ir

Amir Sabbagh Molahosseini
Islamic Azad University, Department of Computer Engineering, Kerman Branch, Kerman, Iran, amir@iauk.ac.ir

Azadeh Alsadat Emrani Zarandi
Shahid Bahonar University of Kerman, Department of Computer Engineering, Kerman, Iran, a.emrani@uk.ac.ir

Received: 23.05.2020
Revised: 04.07.2020
Accepted: 04.07.2020