INTEGRATION, the VLSI journal 55 (2016) 22–29

Contents lists available at ScienceDirect

INTEGRATION, the VLSI journal

journal homepage: www.elsevier.com/locate/vlsi

Method for designing two levels RNS reverse converters for large dynamic ranges

Hector Pettenghi [a,1,*], Ricardo Chaves [b,*], Roberto de Matos [c], Leonel Sousa [d]

[a] Department of Electrical and Electronic Engineering (CTC/UFSC) 88040 Florianópolis, Brazil
[b] INESC-ID, IST, Universidade de Lisboa, 1000-029 Lisbon, Portugal
[c] Department of Electrical and Electronic Engineering (CTC/UFSC) 88040 Florianópolis, Brazil
[d] INESC-ID, IST, Universidade de Lisboa, 1000-029 Lisbon, Portugal

ARTICLE INFO

Article history:
Received 21 September 2015
Received in revised form
23 February 2016
Accepted 23 February 2016
Available online 2 March 2016

Keywords:
Residue Number Systems
RNS-to-Binary converters
Adder-based converters

ABSTRACT

In the last years, research on Residue Number Systems (RNS) has targeted larger dynamic ranges in order to further explore their inherent parallelism. In this paper, we start from the traditional 3-moduli set $\{2^n, 2^n-1, 2^n+1\}$, with an equivalent $3n$-bit dynamic range, and propose horizontal and vertical extensions to scale the dynamic range and enhance the parallelism according to the requirements. Two different methods to design general reverse converters for extended moduli sets to the desired dynamic ranges are introduced. Previous converters require complex weight selection of the inputs or complex final conversion steps. In this work the weight selection of the multiplicative terms associated to the inputs is reduced to additions of $2n$-bit length and the final conversion step requires only one comparison. Experimental results suggest that the proposed approaches achieve significant area reductions, up to 61% lower area reductions, in comparison with the state-of-the-art for generic DR purposes. Despite having identical delay metrics as the existing generic state of the art, Area-Delay-Product efficiency metrics improvements up to 2.7 times can be achieved. The obtained results also validate the improved scalability of the proposed approaches, allowing for better results with the increase of n and the DR.

1. Introduction

Residue arithmetics, based on Residue Number Systems (RNS), have been in use in digital processing systems for many years [1]. RNS is a carry-free arithmetic system with modular characteristics offering the potential for high-speed and parallel computation. Arithmetic operations, such as addition, subtraction, and multiplication, can be carried out more efficiently than in the conventional binary systems [1], independently and concurrently, in several residue channels. The adoption of RNS has provided significant efficiency improvements for different types of Digital Signal Processing (DSP) applications [1], while allowing for an easier scaling for applications with larger dynamic ranges requirements, such as in adaptive filtering and cryptography [2].

The choice of the moduli set is of key importance in order to obtain balanced moduli sets. Moduli sets with a large number of channels can improve the arithmetic computation at the cost of reverse conversion performance.

With efficient reverse converters, capable of supporting large moduli sets, it is possible to compensate this extra cost, especially when several arithmetic operations have to be performed, such as in cryptographic or signal processing systems. In these cases the use of multiple arithmetic moduli channels can lead to better performance metrics.

To support these moduli sets, reverse converters need to be devised. Consequently, reverse conversion structures are usually presented whenever a novel moduli set is proposed. To devise these conversion structures the Chinese Remainder Theorem (CRT) [1], the mixed-radix conversion (MRC) [3] and the New CRT-I [4] algorithms are considered. Each of the moduli sets presented below have an associated conversion structure.

As mentioned before, in applications such as in cryptography [5], very large operands are used for. However, the level of parallelism and the achievable Dynamic Range (DR) provided by the traditional three-moduli set $\{2^n, \overbrace{2^n+1, 2^n-1}^{2^n\pm 1}\}$, with a DR of around $3n$-bit [6,7], are not enough.

* Corresponding author.
E-mail addresses: hector@eel.ufsc.br (H. Pettenghi), ricardo.chaves@inesc-id.pt (R. Chaves), roberto@eel.ufsc.br (R.d. Matos), las@inesc-id.pt (L. Sousa).
[1] Tel.: +55 48 3721 2359; fax: +55 48 3721 9280.

http://dx.doi.org/10.1016/j.vlsi.2016.02.004
0167-9260/

In these cases, horizontal extensions can be used in order to add more moduli to the moduli set. This approach has been considered and proposed in the state-of-the-art, such as the four-moduli sets with a DR of about 4n-bit: $\{2^n, 2^n\pm 1, \mathbf{2^{n+1}+1}\}$ and $\{2^n, 2^n\pm 1, \mathbf{2^{n+1}-1}\}$ [8,9], $\{2^n, 2^n\pm 1, \mathbf{2^{n-1}+1}\}$ and $\{2^n, 2^n\pm 1, \mathbf{2^{n-1}-1}\}$ [10]. Horizontal extensions of five-moduli sets with a DR of about 5n-bit have also been proposed: $\{2^n, 2^n\pm 1, \mathbf{2^{n\pm 1}-1}\}$ [11], $\{2^{n+1}, 2^n\pm 1, \mathbf{2^{n+1}\pm 1}\}$ [12], and $\left\{2^n, 2^n\pm 1, \mathbf{2^n \pm 2^{\frac{(n+1)}{2}}+1}\right\}$ [13] that is composed of co-prime moduli for n odd and has been revisited by Hiasat in [14]. The moduli considered in [12] are co-prime numbers for n even, however, complex multiplicative inverses are required, resulting in expensive reverse conversion structures. In [15] the authors propose a full RNS using the 8 moduli set $\{2^{n-5}-1, 2^{n-3}-1, 2^{n-3}+1, 2^{n-2}+1, 2^{n-1}-1, 2^{n-1}+1, 2^n, 2^n+1\}$. The proposed moduli set is not regular, presenting channels with n to $n-5$ bits with non-co-prime moduli, resulting in a lower DR. As in [12], complex multiplicative inverses are required, resulting in costly and complicated hierarchical reverse converter structures. In addition, vertical extensions of channels have also been proposed in order to increase the DR, such as $\{\mathbf{2^{n+\beta}}, 2^n\pm 1\}$ [7], where $0\le\beta\le n$ is used to increase the DR up to 4n-bits with a 3-moduli set. This is achieved towards a more balanced moduli set, since the performance difference between the 2^n units and the $2^n\pm k$ arithmetic units. Therefore the overloading of the 2^n channel up to 2^{2n} can be done without affecting the delay in the arithmetic channels.

Moduli sets with both vertically and horizontally extensions have also been recently proposed $\{\mathbf{2^{2n}}, 2^n\pm 1, \mathbf{2^{2n+1}-1}\}$ [16], $\{\mathbf{2^{2n}}, 2^n\pm 1, \mathbf{2^{2n}+1}\}$ [17], and $\left\{\mathbf{2^{n+\beta}}, 2^n\pm 1, \mathbf{2^n\pm 2^{\frac{n+1}{2}}+1}, \mathbf{2^{n+1}+1}\right\}$ and $\left\{\mathbf{2^{n+\beta}}, 2^n\pm 1, \mathbf{2^n\pm 2^{\frac{n+1}{2}}+1}, \mathbf{2^{n-1}+1}\right\}$ with $-\frac{(n-1)}{2}\le\beta\le 3n$ [18]. The proposals [16,17] provide a DR of $\simeq$ 6n-bits at a cost of unbalancing the moduli set. In contrast, the proposal [18] provides a more balanced moduli set with a maximum DR of $(8n+1)$-bit.

In the paper [19] a method based on New CRT-I for designing RNS reverse converter that uses generic hybrid extended moduli sets of the form $\{\mathbf{2^{n+\beta}}, 2^n\pm 1, \mathbf{2^n\pm k_1}, \mathbf{2^n\pm k_2}, \ldots, \mathbf{2^n\pm k_f}\}$ is presented, with k_j being odd values and $0\le\beta\le n$. However, this method imposes a complex modular weight selection of the multiplicative terms, V_{ji}, associated to the residue inputs R_i, which is a substantial drawback. Moreover, the modular addition of these weighted inputs requires a large number of comparisons, and consequently a dedicated circuitry is used in the architecture to reduce the complexity of the Final Conversion step (FC). In this work a method is proposed to accommodate the generic moduli set horizontal and vertical extended presented in [19], by reducing the modular values used as the multiplicative terms, V_{ji}, and requiring only a single comparison in the final conversion operation.

The remaining of this paper is organized as follows. A novel method to design reverse converters to the extended moduli sets is presented in Section 2, and an additional technique that minimizes the number of required levels is presented in Section 3. A performance analysis of a case study is presented in Section 4. The efficiency of the state-of-the-art of reverse converters with large DRs is compared with the one achieved with our proposals, in Section 5. Section 6 concludes this paper with some final remarks.

2. Multi-level hybrid extensions of the three-moduli set $\{2^n, 2^n\pm 1\}$

To simplify the presentation of the method and the description of the architectures, the following notation is adopted [14]: (i) the symbol $*$ operates the concatenation of the binary representation of two numbers, and (ii) R_i denotes the residue for m_i.

The dynamic range is equal to the product of the N moduli of a defined set ($M=\prod_{i=1}^{N} m_i$), $\hat{m}_i = M/m_i$, and $\left|\hat{m}_i^{-1}\right|_{m_i}$ represents the multiplicative inverse of $\hat{m}_i$ with respect to modulus m_i. A value represented in RNS can be converted back to binary (X) using the CRT [1]:

$$X=\left|\sum_{i=1}^{N}\hat{m}_i\left|\hat{m}_i^{-1}\right|_{m_i}R_i\right|_M=\sum_{i=1}^{N}\hat{m}_i\left|\hat{m}_i^{-1}\right|_{m_i}R_i - MA(X), \tag{1}$$

where $A(X)$ is an integer that depends on the value of X.

As stated above, herein both horizontal and vertical extensions are considered. For the vertical extension the power of two modulus is extended, towards a more balanced moduli set, since the power of two modulus typically allows for more efficient arithmetic operations than the remaining moduli sets for the same word length [18]. This leads to the moduli $\{\mathbf{2^{n+\beta}}, 2^n\pm 1\}$, with $0\le\beta\le n$, covering DRs up to $(4n)$-bits [7].

In order to achieve arbitrarily wider moduli sets, horizontal moduli set extensions are herein considered by the addition of conjugate moduli pairs to the above moduli set in the same way as [19]. These conjugate moduli pairs are of the form $2^n\pm k_j$, $0\le j\le f$, with k_j being an odd value in the range $1\le k_j<2^{n-1}$ [20] chosen in such a way that all moduli are co-prime with each other. With this, moduli sets of the form $\{\mathbf{2^{n+\beta}}, \mathbf{2^n\pm k_f}, \ldots, \mathbf{2^n\pm k_1}, \mathbf{2^n\pm k_0}\}$ are obtained, with a DR around $(1+\frac{\beta}{n}+2\times(f+1))\times n$-bit, for any integer n.

The values of k_j can be chosen in order to obtain the highest possible DR, however a cost function can be used to obtain the most balanced moduli sets and minimizing the number of "1"s in the representation of k_j in order to derive more efficient architectures, such as the ones presented in the following. It should noted that the proposed method can also be used to derive reverse converters for moduli sets with non-conjugate moduli pairs. Herein, we only detail moduli sets with conjugate moduli pairs to simplify the explanation.

In order to illustrate the proposed moduli set extensions, we first derive the extension for the moduli set with $f=1$ and $k_0=1$, resulting in the moduli set $\{\mathbf{2^{n+\beta}}, \mathbf{2^n\pm k_1}, 2^n\pm 1\}$. Following, the derivation and discussion of the limitations of extending the moduli set with conjugate moduli pairs for different f values is also presented.

2.1. Moduli set $\{\mathbf{2^{n+\beta}}, \mathbf{2^n\pm k_1}, 2^n\pm 1\}$

As presented above, let us consider the value $0\le\beta\le n$. From now on, the values of the moduli sets are ordered in a decreasing order, excluding the 2^n+1 and 2^n-1 (placed in the before-last and last positions), resulting in $m_1=2^{n+\beta}$, $m_2=2^n+k_1$, $m_3=2^n-k_1$, and $m_4=2^n+1$, $m_5=2^n-1$, whereas $\hat{m}_1=2^{4n}-2^{2n}(k_1^2+1)+k_1^2$, $\hat{m}_2=2^{n+\beta}(2^{2n}-1)(2^n-k_1)$, $\hat{m}_3=2^{n+\beta}(2^{2n}-1)(2^n+k_1)$, $\hat{m}_4=2^{n+\beta}(2^n-1)(2^{2n}-k_1^2)$, and $\hat{m}_5=2^{n+\beta}(2^n+1)(2^{2n}-k_1^2)$.

For the proposed extension the following expression $\hat{\hat{m}}_i$ is used:

$$\hat{\hat{m}}_i=\frac{M}{\prod_{j=1}^{i} m_j}\quad \text{with } 1\le i\le 5. \tag{2}$$

The chosen k_1 needs to satisfy that the resulting moduli set $\{\mathbf{2^{n+\beta}}, \mathbf{2^n\pm k_1}, 2^n\pm 1\}$ is composed of co-prime numbers. For example, $k_1=3$ satisfies this condition for $n\ge 3$.

The values of the multiplicative inverses are integer numbers, which can be obtained by applying the condition $\left|(\hat{m}_i)(\hat{m}_i^{-1})\right|_{m_i}=1$, $1\le i\le 5$ [14].

It is important to notice that the multiplicative inverse $\left|\hat{m}_1^{-1}\right|_{m_1}$ satisfies the following equation when $0 \le \beta \le n$:

$$\left||\hat{m}_1^{-1}|_{m_1}\hat{m}_1\right|_{m_1} = \left||\hat{m}_1^{-1}|_{m_1}(\overbrace{\left|2^{4n}\right|_{m_1}}^{=0} - \overbrace{\left|2^{2n}\right|_{m_1}}^{=0}(k_1^2+1)+k_1^2)\right|_{m_1}$$

$$= \left||\hat{m}_1^{-1}|_{m_1}k_1^2\right|_{m_1} = 1. \tag{3}$$

Given that $|(\psi m_1+1)|_{m_1} = 1$, with ψ being a positive integer, the multiplicative inverse $\left|\hat{m}_1^{-1}\right|_{m_1}$ can be expressed as:

$$\left||\hat{m}_1^{-1}|_{m_1}k_1^2\right|_{m_1} = |(\psi m_1+1)|_{m_1} = 1 \Rightarrow |\hat{m}_1^{-1}|_{m_1} = \frac{\psi m_1+1}{k_1^2}. \tag{4}$$

- Therefore it is possible to reduce the modulo computation from M to $\hat{\hat{m}}_1 = \hat{m}_1$ in Eq. (1) as follows:

$$X = \left|\sum_{i=1}^{5}\overbrace{\hat{m}_i|\hat{m}_i^{-1}|_{m_i}}^{V_{0i}} R_i\right|_M = \left||\,|\hat{m}_1^{-1}|_{m_1}\hat{m}_1R_1|_M + \sum_{i=2}^{5}\left|\overbrace{|\hat{m}_i^{-1}|_{m_i}\hat{m}_i}^{V'_{0i}} R_i\right|_M\right|_M =$$

$$\left||\hat{m}_1^{-1}|_{m_1}(2^{4n}-2^{2n}(k_1^2+1)+k_1^2)R_1 + \sum_{i=2}^{5}V'_{0i}R_i\right|_M =$$

$$\left||\hat{m}_1^{-1}|_{m_1}R_1[2^{4n}-2^{2n}(k_1^2+1)]+(m_1\psi+1)R_1 + \sum_{i=2}^{5}V'_{0i}R_i\right|_M =$$

$$\left|\overbrace{[|\hat{m}_1^{-1}|_{m_1}[2^{4n}-2^{2n}(k_1^2+1)]+m_1\psi]}^{V'_{01}} R_1 + R_1 + \sum_{i=2}^{5}V'_{0i}R_i\right|_M =$$

$$\left|\sum_{i=1}^{5}V'_{0i}R_i + R_1\right|_M, \tag{5}$$

where Eq. (5) can be rewritten as:

$$X = \sum_{i=1}^{5}V'_{0i}R_i - MA(X) + R_1 = \sum_{i=1}^{5}\left(\frac{V'_{0i}}{m_1}R_i - \overbrace{\frac{M}{m_1}}^{\hat{\hat{m}}_1A(X)}\right)m_1 + R_1$$

$$= \left|\sum_{i=1}^{5}\overbrace{\frac{V'_{0i}}{m_1}}^{V_{1i}}R_i\right|_{\hat{\hat{m}}_1} m_1 + R_1 = \overbrace{\left|\sum_{i=1}^{5}V_{1i}R_i\right|_{\hat{\hat{m}}_1}}^{X_1} m_1 + R_1. \tag{6}$$

Due to X_1m_1 is a shift-left operation of $(n+\beta)$-bits, the X_1 can be concatenated to R_1 to derive X, where R_1 becomes the less significant $(n+\beta)$ bits of X.

- In the second reduction to the modulo $\hat{\hat{m}}_2$ computation, the V_{1i} values are split into two terms, which are the divisible and the indivisible terms of the division $\frac{V_{1i}}{\hat{\hat{m}}_2}$. These divisible and indivisible terms are denoted as V'_{1i} and V''_{1i}, respectively, as presented in Eq. (7) and their values can be obtained from the division equation of $\sum_{i=1}^{5}V_{1i}$ by $\hat{\hat{m}}_2$, $\sum_{i=1}^{5}V_{1i} = \sum_{i=1}^{5}\overbrace{\left\lfloor\frac{V_{1i}}{\hat{\hat{m}}_2}\right\rfloor\hat{\hat{m}}_2}^{V'_{1i}} + \sum_{i=1}^{2}V''_{1i}$. It is important to note that $\hat{m}_j$ is divisible by $\hat{\hat{m}}_2$ for $3 \le j \le 5$ and consequently $V''_{1j} = 0$ in these cases:

$$X_1 = \left|\sum_{i=1}^{5}V_{1i}R_i\right|_{\hat{\hat{m}}_1} = \left|\left|\sum_{i=1}^{5}V'_{1i}R_i\right|_{\hat{\hat{m}}_1} + \left|\sum_{i=1}^{2}V''_{1i}R_i\right|_{\hat{\hat{m}}_1}\right|_{\hat{\hat{m}}_1}$$

$$= \left|\sum_{i=1}^{5}V'_{1i}R_i - \hat{\hat{m}}_1A(X_1) + \left|\sum_{i=1}^{2}V''_{1i}R_i\right|_{\hat{\hat{m}}_1}\right|_{\hat{\hat{m}}_1}$$

$$= \left|\sum_{i=1}^{5}\left(\overbrace{\frac{V'_{1i}}{m_2}}^{V_{2i}}R_i - \overbrace{\frac{\hat{\hat{m}}_1}{m_2}}^{\hat{\hat{m}}_2}A(X_1)\right)m_2 + \left|\sum_{i=1}^{2}\overbrace{V''_{1i}}^{\phi_{2i}}R_i\right|_{\hat{\hat{m}}_1}\right|_{\hat{\hat{m}}_1}$$

$$= \left|\overbrace{\left|\sum_{i=1}^{5}V_{2i}R_i\right|_{\hat{\hat{m}}_2}}^{X_2}m_2 + \left|\sum_{i=1}^{2}\phi_{2i}R_i\right|_{\hat{\hat{m}}_1}\right|_{\hat{\hat{m}}_1}. \tag{7}$$

- In the third reduction to the modulo $\hat{\hat{m}}_3$ computation, the V_{2i} values are divided into $\sum_{i=1}^{5}V_{2i} = \sum_{i=1}^{5}(V'_{2i}+V''_{2i})$ using the same approach as above. Note that in this case $V''_{2i} = 0$ for $4 \le j \le 5$:

$$X_2 = \left|\sum_{i=1}^{5}V_{2i}R_i\right|_{\hat{\hat{m}}_2} = \left|\left|\sum_{i=1}^{5}V'_{2i}R_i\right|_{\hat{\hat{m}}_2} + \left|\sum_{i=1}^{3}V''_{2i}R_i\right|_{\hat{\hat{m}}_2}\right|_{\hat{\hat{m}}_2}$$

$$= \left|\sum_{i=1}^{5}V'_{2i}R_i - \hat{\hat{m}}_2A(X_2) + \left|\sum_{i=1}^{3}V''_{2i}R_i\right|_{\hat{\hat{m}}_2}\right|_{\hat{\hat{m}}_2}$$

$$= \left|\sum_{i=1}^{5}\left(\overbrace{\frac{V'_{2i}}{m_3}}^{V_{3i}}R_i - \overbrace{\frac{\hat{\hat{m}}_2}{m_3}}^{\hat{\hat{m}}_3}A(X_2)\right)m_3 + \left|\sum_{i=1}^{3}\overbrace{V''_{2i}}^{\phi_{3i}}R_i\right|_{\hat{\hat{m}}_2}\right|_{\hat{\hat{m}}_2}$$

$$= \left|\overbrace{\left|\sum_{i=1}^{5}V_{3i}R_i\right|_{\hat{\hat{m}}_3}}^{X_3}m_3 + \left|\sum_{i=1}^{3}\phi_{3i}R_i\right|_{\hat{\hat{m}}_2}\right|_{\hat{\hat{m}}_2}. \tag{8}$$

At the end:

$$X = X_1m_1 + R_1; \quad X_1 = \left|X_2m_2 + \left|\sum_{i=1}^{2}\phi_{2i}R_i\right|_{\hat{\hat{m}}_1}\right|_{\hat{\hat{m}}_1};$$

$$X_2 = \left|X_3m_3 + \left|\sum_{i=1}^{3}\phi_{3i}R_i\right|_{\hat{\hat{m}}_2}\right|_{\hat{\hat{m}}_2}; \quad X_3 = \left|\sum_{i=1}^{5}V_{3i}R_i\right|_{\hat{\hat{m}}_3}. \tag{9}$$

It is important to note that $\phi_{3i} < m_3$ and $\phi_{2i} < m_2$, and consequently the multiplications of these constants by the corresponding R_i are non-modular operations. The constraints for $\phi_{3i}R_i$ and $\phi_{2i}R_i$ are:

$$\max(\phi_{31}R_1) = (m_3-1)(m_1-1) < m_3m_4m_5;$$
$$\max(\phi_{32}R_2) = (m_3-1)(m_2-1) < m_3m_4m_5;$$
$$\max(\phi_{33}R_3) = (m_3-1)(m_3-1) < m_3m_4m_5;$$
$$\max(\phi_{21}R_1) = (m_2-1)(m_1-1) < m_2m_3m_4m_5;$$
$$\max(\phi_{22}R_2) = (m_2-1)(m_2-1) < m_2m_3m_4m_5, \tag{10}$$

where the range $0 \le \beta \le n$ guarantees Eq. (10).

In order to obtain a non-modular addition of the $\phi_{3i}R_i$ and $\phi_{2i}R_i$ terms, $X = M-1$ is set as input to provide the maximum residue

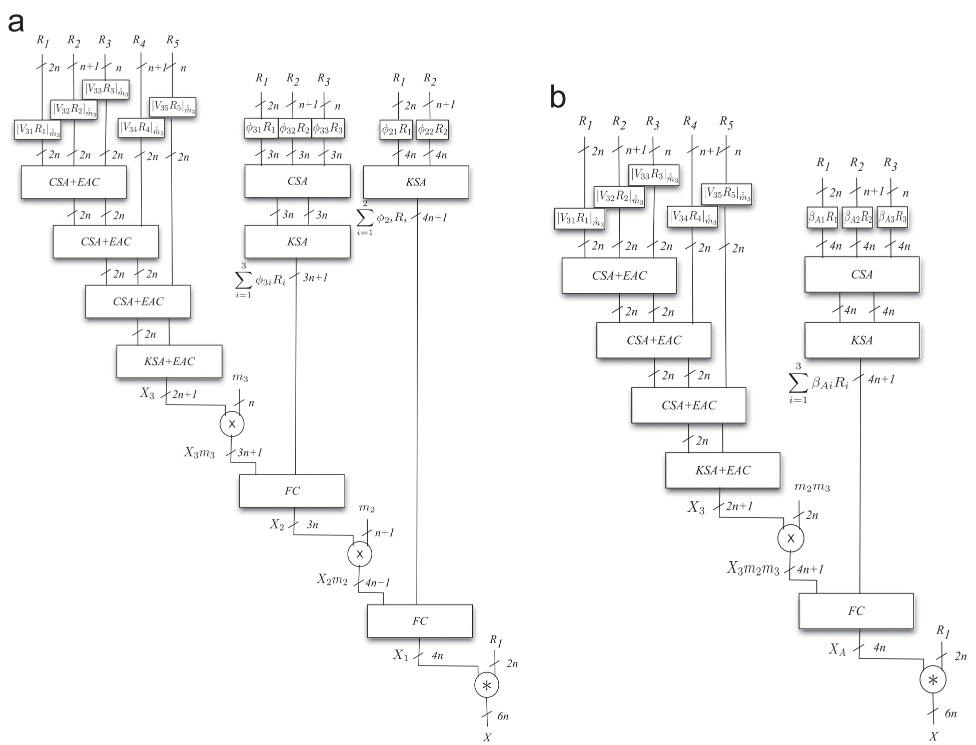

Fig. 1. Block diagram of the reverse converter $\{\mathbf{2^{2n}}, \mathbf{2^n} \pm \boldsymbol{k_1}, 2^n \pm 1\}$ for (a) Multi-level approach, and (b) two-level approach.

values in $R_i = m_i - 1$, $1 \le i \le 3$:

$$R_i = |M-1|_{m_i} = \left| \overbrace{|M|_{m_i}}^{=0} - 1 \right|_{m_i} = |-1|_{m_i} = m_i - 1. \tag{11}$$

Therefore:

$$\sum_{i=1}^{3} \phi_{3i}R_i = \overbrace{\phi_{31}}^{<m_3}(m_1-1) + \overbrace{\phi_{32}}^{<m_3}(m_2-1) + \overbrace{\phi_{33}}^{<m_3}(m_3-1) < \hat{\hat{m}}_2;$$

$$\sum_{i=1}^{2} \phi_{2i}R_i = \overbrace{\phi_{21}}^{<m_2}(m_1-1) + \overbrace{\phi_{22}}^{<m_2}(m_2-1) < \hat{\hat{m}}_1, \tag{12}$$

which is satisfied for $\beta = 0$ due to the value of $m_1 - 1 = m_5$, and consequently $\sum_{i=1}^{j+1} \phi_{(j+1)i}R_i < \hat{\hat{m}}_j$, for j=1 and j=2. However, when $\beta = n$ the value of $m_1 - 1$ is m_4m_5 and consequently, Eq. (12) cannot be satisfied for some particular cases. If the moduli selected does not satisfy Eq. (12), then the solution consists of the reduction of β to $\beta = 0$. Once the values of ϕ_{3i} and ϕ_{2i} are obtained and is assured that Eq. (12) is satisfied, it is possible to rewrite Eq. (9) as:

$$X = X_1m_1 + R_1; \quad X_1 = \left| X_2m_2 + \sum_{i=1}^{2} \phi_{2i}R_i \right|_{\hat{\hat{m}}_1};$$

$$X_2 = \left| X_3m_3 + \sum_{i=1}^{3} \phi_{3i}R_i \right|_{\hat{\hat{m}}_2}; \quad X_3 = \left| \sum_{i=1}^{5} V_{3i}R_i \right|_{\hat{\hat{m}}_3}. \tag{13}$$

It is important to note that another reduction to $\hat{\hat{m}}_4$ is possible. However, the $\hat{\hat{m}}_3$ modulo computation can be easily implemented with a Rotate-Left binary operation, ROL, based on a Carry Save Adder (CSA), and End Around Carry (EAC) trees as explained in [7], since $\hat{\hat{m}}_3 = m_4m_5 = 2^{2n} - 1$. The multiplications by ϕ_{ji} terms and m_i, $1 \le i \le 3$, are not modular operation, therefore they are implemented by conventional binary multipliers. The final conversion step to derive X_3 is implemented by using a Kogge and Stone Adder (KSA) and EAC. The Kogge Stone prefix adder structure is herein considered since it is one of the adder structures suggesting better performances [22].

The final conversion steps to derive X_1 and X_2 require only one comparison, since $\max(X_3m_3 + \sum_{i=1}^{3} \phi_{3i}R_i) < 2 \times \hat{\hat{m}}_2$ and $\max(X_2 m_2 + \sum_{i=1}^{2} \phi_{2i}R_i) < 2 \times \hat{\hat{m}}_1$. The architecture of the resulting $\{\mathbf{2^{2n}}, \mathbf{2^n} \pm \boldsymbol{k_1}, 2^n \pm 1\}$ converter by using this approach is depicted in Fig. 1(a).

2.2. Moduli set $\{\mathbf{2^{n+\beta}}, \mathbf{2^n} \pm \boldsymbol{k_f}, \ldots, \mathbf{2^n} \pm \boldsymbol{k_2}, \mathbf{2^n} \pm \boldsymbol{k_1}, 2^n \pm 1\}$

For a general extension, to which correspond the moduli set $\{\mathbf{2^{n+\beta}}, \mathbf{2^n} \pm \boldsymbol{k_f}, \ldots, \mathbf{2^n} \pm \boldsymbol{k_2}, \mathbf{2^n} \pm \boldsymbol{k_1}, 2^n \pm 1\}$, the multiplicative inverses have to satisfy the condition $|(\hat{\hat{m}}_i)(\hat{m}_i^{-1})|_{m_i} = 1$, for $1 \le i \le (3 + 2 \times f)$. The number of iterative reductions and moduli in the sets are $t = 2 \times f + 1$ and $N = 2 \times f + 3$, respectively.

To obtain the binary value of X, Eq. (13) can be extended:

$$X = \overbrace{\left| \sum_{i=1}^{N} V_{1i}R_i \right|_{\hat{\hat{m}}_1}}^{X_1} m_1 + R_1 = \left| \sum_{i=1}^{N} V_{0i}R_i \right|_{\hat{\hat{m}}_0};$$

$$X_1 = \left| \overbrace{\left| \sum_{i=1}^{N} V_{2i}R_i \right|_{\hat{\hat{m}}_2}}^{X_2} m_2 + \sum_{i=1}^{2} \phi_{2i}R_i \right|_{\hat{\hat{m}}_1} = \left| \sum_{i=1}^{N} V_{1i}R_i \right|_{\hat{\hat{m}}_1};$$

$$\vdots$$

$$X_{j-1} = \left| \overbrace{\left| \sum_{i=1}^{N} V_{ji}R_i \right|_{\hat{\hat{m}}_j}}^{X_j} m_j + \sum_{i=1}^{j} \phi_{ji}R_i \right|_{\hat{\hat{m}}_{j-1}} = \left| \sum_{i=1}^{N} V_{(j-1)i}R_i \right|_{\hat{\hat{m}}_{j-1}};$$

$$\vdots$$

$$X_{t-1} = \left| X_t m_t + \sum_{i=1}^{t} \phi_{(t)i}R_i \right|_{\hat{\hat{m}}_{(t-1)}} = \left| \sum_{i=1}^{N} V_{(t-1)i}R_i \right|_{\hat{\hat{m}}_{(t-1)}};$$

$$X_t = \left| \sum_{i=1}^{N} V_{ti}R_i \right|_{\hat{\hat{m}}_t}. \tag{14}$$

The parameters of Eq. (14) for the $(j+1)$-reduction, $1 \le j \le t-1$, can be derived as follows:

$$\left|\sum_{i=1}^{N} V_{ji}R_i\right|_{\hat{m}_j} = \left|\left|\sum_{i=1}^{N} \overbrace{\left\lfloor \frac{V_{ji}}{\hat{\hat{m}}_{j+1}} \right\rfloor \hat{m}_{j+1}}^{V'_{ji}} R_i\right|_{\hat{m}_j} + \left|\sum_{i=1}^{j+1} V''_{ji}R_i\right|_{\hat{m}_j}\right|_{\hat{m}_j}$$

$$= \left|\left|\sum_{i=1}^{N} \overbrace{\frac{V'_{ji}}{m_{j+1}}}^{V_{(j+1)i}} R_i\right|_{\hat{m}_{j+1}} m_{j+1} + \sum_{i=1}^{j+1} \overbrace{V''_{ji}}^{\phi_{(j+1)i}} R_i\right|_{\hat{m}_j}, \tag{15}$$

in the same way as in Eqs. (7) and (8).

In order to guarantee that to compute $\sum_{i=1}^{j+1} \phi_{(j+1)i}R_i$, a modular addition $\left|\sum_{i=1}^{j+1} \phi_{(j+1)i}R_i\right|_{\hat{m}_j}$ is not required, the extension of Eq. (12) to the $(j+1)$-reduction $1 \le j \le t-1$ needs to be satisfied. In this case is also used $X = M-1$ as input to reach the maximum residue values in $R_i = m_i - 1$, $1 \le i \le N$, as presented in Eq. (11). Therefore:

$$\sum_{i=1}^{j+1} \phi_{(j+1)i}R_i = \sum_{i=1}^{j+1} \phi_{(j+1)i}(m_i - 1) < \hat{\hat{m}}_j. \tag{16}$$

which is satisfied for $\beta = 0$ and mostly of cases for $\beta = n$.

3. Two-levels hybrid extensions of the three-moduli set $\{2^n, 2^n \pm 1\}$

It is possible to simplify the number of iterative reductions to two by using Lemma 1, [21], and Lemma 2:

Lemma 1.

$$|A|_p \times k = |k \times A|_{k \times p}. \tag{17}$$

Lemma 2.

$$\left||A|_q\right|_p = |A - k \times q|_p = |A|_p;\, q = k \times p. \tag{18}$$

For the five moduli set $\{\mathbf{2^{n+\beta}}, \mathbf{2^n \pm k_1}, 2^n \pm 1\}$, it is possible to avoid one iterative reduction if Lemma 1 is applied to $X_2 m_2$ in Eq. (13):

$$X_2 m_2 = \left|X_3 m_3 + \sum_{i=1}^{3} \phi_{3i}R_i\right|_{\hat{m}_2} \times m_2 = \left|X_3 m_3 m_2 + \sum_{i=1}^{3} \phi_{3i} m_2 R_i\right|_{\hat{m}_1}. \tag{19}$$

Therefore Eq. (13) can be reduced by applying Lemma 2 to:

$$X = X_A m_1 + R_1;$$
$$X_A = \left|X_3 m_2 m_3 + \sum_{i=1}^{3} \beta_{Ai} R_i\right|_{\hat{m}_1};$$
$$X_3 = \left|\sum_{i=1}^{5} V_{3i}R_i\right|_{\hat{m}_3}, \tag{20}$$

with $\sum_{i=1}^{3} \beta_{Ai}R_i = \sum_{i=1}^{3} \phi_{3i} m_2 R_i + \sum_{i=1}^{2} \phi_{2i}R_i$.

It is important to note that the terms $\beta_{Ai}R_i$ are non-modular multiplications, since $0 \le \beta_{Ai} < m_2 m_3 - 1$:

$$\max(\beta_{A1}R_1) = [(m_3-1)m_2 + (m_2-1)](m_1-1) < m_3 m_4 m_5;$$
$$\max(\beta_{A2}R_2) = [(m_3-1)m_2 + (m_2-1)](m_2-1) < m_3 m_4 m_5;$$
$$\max(\beta_{A3}R_3) = [(m_3-1)m_2 + (m_2-1)](m_3-1) < m_3 m_4 m_5, \tag{21}$$

where the range $0 \le \beta \le n$ guarantees the conditions expressed in Eq. (21).

In order to obtain a non-modular addition $\sum_{i=1}^{3} \beta_{Ai}R_i$ terms, $X = M-1$ is set as input to reach the maximum residue values in $R_i = m_i - 1$, $1 \le i \le 3$, as presented in Eq. (11), therefore:

$$\beta_{A1}(m_1-1) + \beta_{A2}(m_2-1) + \beta_{A3}(m_3-1) < m_2 m_3 m_4 m_5, \tag{22}$$

which is always set for $\beta = 0$. When the values of β_{Ai} for $\beta = n$ are obtained and the condition of Eq. (22) is satisfied we can guarantee a non-modular addition $\sum_{i=1}^{3} \beta_{Ai}R_i$ as presented in Eq. (20). If the moduli selected does not satisfy Eq. (22), then the solution consists of the reduction of β to $\beta = 0$.

The architecture of a generic $\{\mathbf{2^{2n}}, \mathbf{2^n \pm k_1}, 2^n \pm 1\}$ by using this approximation is depicted in Fig. 1(b). The main difference in comparison with the proposal shown in Fig. 1(a) is the use of only one multiplier and one FC.

For the generic moduli set $\{\mathbf{2^{2n}}, \mathbf{2^n \pm k_f}, \ldots, \mathbf{2^n \pm k_2}, \mathbf{2^n \pm k_1}, 2^n \pm 1\}$, with $t = 2 \times f + 1$ and $N = 2 \times f + 3$ we apply Lemma 1 $2(t-3)$ times and consequently Eq. (20) can be expressed as:

$$X = X_A m_1 + R_1;$$
$$X_A = \left|X_t \prod_{i=2}^{t} m_i + \sum_{i=1}^{t} \beta_{Ai}R_i\right|_{\hat{m}_1};$$
$$X_t = \left|\sum_{i=1}^{N} V_{ti}R_i\right|_{\hat{m}_t}. \tag{23}$$

In this case:

$$\sum_{i=1}^{t-1} \beta_{Ai}R_i = \sum_{i=1}^{2} \phi_{2i}R_i + \sum_{j=1}^{2}\sum_{i=1}^{t-2} \phi_{j(t+1-i)} \left(\prod_{i-1}^{t-1} m_{(t+1-i)}\right) R_j$$
$$+ \sum_{j=3}^{t-1}\sum_{i=3}^{t+3-j} \phi_{j(t+3-i)} \left(\prod_{i-1}^{t-1} m_{(t+1-i)}\right) R_j + \phi_{(t)t} \left(\prod_{i=2}^{t-1} m_i\right) R_t, \tag{24}$$

where the values of ϕ_{ji} can be derived from Eq. (15).

In order to guarantee the addition $\sum_{i=1}^{t} \beta_{Ai}R_i$ can be performed by non-modular adders, the values of β_{Ai} obtained multiplied by the maximum residue values, $R_i = m_i - 1$, $1 \le i \le N$, need to satisfy:

$$\sum_{i=1}^{t} \beta_{Ai}R_i = \sum_{i=1}^{t} \beta_{Ai}(m_i - 1) < \hat{m}_1, \tag{25}$$

which is satisfied for $\beta = 0$ and mostly of cases for $\beta = n$.

4. Performance estimation: A case study

In order to better illustrate the proposed methodology, a particular case study for the moduli set $\{\mathbf{2^{2n}}, \mathbf{2^n \pm 3}, 2^n \pm 1\}$ with $n=4$, $\{m_1, m_2, m_3, m_4, m_5\} = \{256, 19, 13, 17, 15\}$, is herein considered. The following illustrates the resulting parameters for each conversion approaches. Applying CRT [1]:

$$X = \Big| |\overbrace{3\,590\,145}^{V_{01}} R_1|_{16\,124\,160} + |\overbrace{3\,394\,560}^{V_{02}} R_2|_{16\,124\,160}$$
$$+ |\overbrace{11\,162\,880}^{V_{03}} R_3|_{16\,124\,160} + |\overbrace{15\,175\,680}^{V_{04}} R_4|_{16\,124\,160}$$
$$+ |\overbrace{15\,049\,216}^{V_{05}} R_5|_{16\,124\,160} \Big|_{16\,124\,160}. \tag{26}$$

Applying MRC [3]:

$$X = \overbrace{|8(|7(|4(|1(R_5 - V_1)|_{15} - V_2)|_{15} - V_3)|_{15} - V_4)|_{15}}^{V_5}$$
$$\times \overbrace{1\,074\,944}^{= m_1 m_2 m_3 m_4} + \overbrace{|4(|9(|1(R_4 - V_1)|_{17} - V_2)|_{17} - V_3)|_{17}}^{V_4}$$
$$\times \overbrace{63\,232}^{= m_1 m_2 m_3} + \overbrace{|11(|3(R_3 - V_1)|_{13} - V_2)|_{13}}^{V_3} \times \overbrace{4864}^{= m_1 m_2}$$

Table 1
Operations comparisons of reverse conversion algorithm approaches for $\{\mathbf{2^{2n}}, \mathbf{2^n \pm 3}, 2^n \pm 1\}$, n=4.

Operation		CRT [1]	MRC [3]	New CRT [19]	Multi-level hybrid	2-levels hybrid
Modular $\lvert V_{ji}R_i\rvert_{\hat{m}_j}$	#	5	10	5	5	5
	$\hat{m}_j$	16 124 160	[a]	62 985	255	255
Additions	#	1	11	1	3	2
	Compressor	5:1	6:1; $(2:1)\times 10$	5:1	5:1; 3:1; 2:1	5:1; 3:1
	(bits)	$(6n)$	$(6n)$; $(2n)\times 4$; $(3n)\times 3$; $(4n)\times 2$; $(5n)$	$(4n)$	$(2n)$; $(3n)$; $(4n)$	$(2n)$; $(4n)$
Multiplications	#	–	4	–	7	4
	Mult. length	–	$(n+1)\times 2n$ $n\times(3n+1)$ $(n+1)\times 4n$ $n\times(5n+1)$	–	$(2n+1)\times n$; $n\times n$; $2n\times n$; $(n+1)\times n$; $3n\times(n+1)$; $2n\times 2n$; $2n\times(n+1)$	$(2n+1)\times 2n$; $2n\times(n+1)$; $2n\times 2n$; $2n\times n$
Modular FCs	#	1	–	1	2	1
	Comparisons	1	–	1	1	1

[a] The given modular multiplication is not in the form $\lvert V_{ji}R_i\rvert_{\hat{m}_j}$. Each modular multiplication of MRC implementation is shown in Eq. (27).

$$+\overbrace{\lvert 17(R_2-V_1)\rvert_{19}}^{V_2}\times\overbrace{256}^{=m_1}+\overbrace{R1}^{=V_1}, \tag{27}$$

where the constant values can be extracted from [23].

Applying the methodology described in [19] based on New CRT-I:

$$X=\Big|\,\lvert\overbrace{14\,024}^{V_{11}}R_1\rvert_{62\,985}+\lvert\overbrace{13\,260}^{V_{12}}R_2\rvert_{62\,985}+\lvert\overbrace{43\,605}^{V_{13}}R_3\rvert_{62\,985}+\lvert\overbrace{59\,280}^{V_{14}}R_4\rvert_{62\,985}+\lvert\overbrace{58\,786}^{V_{15}}R_5\rvert_{62\,985}\Big|_{62\,985}\times 256+R_1. \tag{28}$$

Applying the methodology described in Section 2:

$$X=\Bigg|\bigg|\Big|\,\lvert\overbrace{56}^{V_{31}}R_1\rvert_{255}+\lvert\overbrace{53}^{V_{32}}R_2\rvert_{255}+\lvert\overbrace{176}^{V_{33}}R_3\rvert_{255}+\lvert\overbrace{240}^{V_{34}}R_4\rvert_{255}+\lvert\overbrace{238}^{V_{35}}R_5\rvert_{255}\Big|_{255}\times 13+\overbrace{10}^{\phi_{31}}R_1+\overbrace{8}^{\phi_{32}}R_2+\overbrace{7}^{\phi_{33}}R_3\bigg|_{3315}\times 19+\overbrace{2}^{\phi_{21}}R_1+\overbrace{17}^{\phi_{22}}R_2\Bigg|_{62\,985}\times 256+R_1. \tag{29}$$

Applying the methodology described in Section 2:

$$X=\bigg|\Big|\,\lvert\overbrace{56}^{V_{31}}R_1\rvert_{255}+\lvert\overbrace{53}^{V_{32}}R_2\rvert_{255}+\lvert\overbrace{176}^{V_{33}}R_3\rvert_{255}+\lvert\overbrace{240}^{V_{34}}R_4\rvert_{255}+\lvert\overbrace{238}^{V_{35}}R_5\rvert_{255}\Big|_{255}\times 247+\overbrace{192}^{\beta_{A1}}R_1+\overbrace{169}^{\beta_{A2}}R_2+\overbrace{133}^{\beta_{A3}}R_3\bigg|_{62\,985}\times 256+R_1. \tag{30}$$

Table 1 shows the operations comparison of different reverse conversion approaches for this case study.

The CRT [1] and New CRT-I [19] based converters are implemented by using modular $\lvert V_{ji}R_i\rvert_{\hat{m}_j}$ operations in order to obtain a single comparison at the FC. However, the modular additions and multiplications required are more complex than the rest of the existing and herein proposed approaches. The use of ROL operations to derive the modular $\lvert V_{ji}R_i\rvert_{\hat{m}_j}$ operations, in the same way as in [18], is not an efficient approach when the V_{ji} terms have a large number of logic 1's in their binary representations.

The approaches presented in [19] require complex modular $\lvert V_{ji}R_i\rvert_{\hat{m}_j}$ operations, since $\hat{m}_j$ is not of the form $\{2^n\pm 1\}$. However, the hybrid and 2-level proposals use the modulo $\hat{m}_t$, equal to $2^{2n}-1$, that allows for the simplification of the modular operations.

It is important to note that the addition of the 5 terms with $2n$ bits of length, in the proposed approach, is performed by a CSA+EAC as explained in Section 2.

Given the use of $\hat{m}_t$ is equal to $2^{2n}-1$ for the hybrid and the 2-level proposals, the additions at the end of the CSA+EAC trees can be implemented with a KSA+EAC. Additionally, the remaining final conversion stages require a single comparison. It is important to note that the implementations based on CRT [1] and New CRT-I [19] require extra hardware in the CSA tree to compute the final conversion step with a single comparison.

Regarding the scalability of the 2-level proposal, it can be seen from Eq. (23) that the complexity of the computation of X_A and X_t grows with the number of moduli. Each extra modulo j will imply the non-modular addition of $\beta_{Aj}R_j$ to X_A and the modular addition of $V_{tj}R_j$ to X_t. Note that, the computation of X_A is not in the critical path, and the additions to compute X_t can be grouped, in order to compute them in parallel. This suggests a linear increase of the area with the number of moduli and no significant delay increase. Finally, the multiplication of X_t by the constant term, $\prod_{i=2}^{t} m_i=\prod_{i=2}^{t}(2^n+k_i)$, also increases by n with each extra modulo. Nevertheless, this is a non-modular multiplication by a constant and considering small values of k_i (in comparison with 2^n). Thus, an acceptable area and delay increase with each extra modulo is expected.

Overall, the 2-level proposal seems scalable, given its expect area and delay increase with each additional modulo.

5. Experimental results

In order to assess the performance and cost of the generic reverse conversion approach herein proposed, the proposed and state-of-the-art reverse converters structures were described in a synthesizable VHDL and implemented on a 90 nm Standard Cell ASIC technology from UMC [24], using the Design Vision synthesis tool (version E-2010.12-SP4).

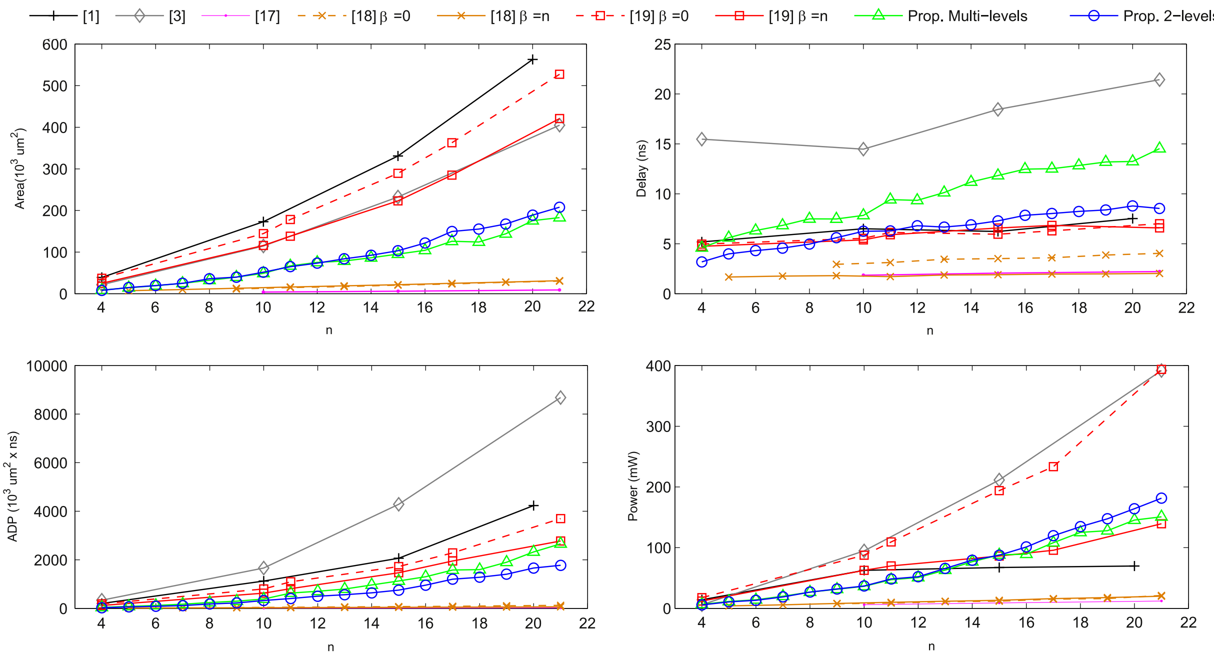

Fig. 2. Experimental results for reverse converters for the moduli set with $DR=6n$ bits, $\{\mathbf{2^{2n}},\mathbf{2^n}\pm\mathbf{3},2^n\pm1\}$ based on CRT [1], and based on MRC [3], $\{\mathbf{2^{2n}},2^n\pm1,\mathbf{2^{2n}}+1\}$ [17], $\left\{\mathbf{2^n},2^n\pm1,\mathbf{2^n}\pm\mathbf{2^{\frac{n+1}{2}}}+1,\mathbf{2^n}+\mathbf{1}+\mathbf{1}\right\}$ and $\left\{\mathbf{2^n},2^n\pm1,\mathbf{2^n}\pm\mathbf{2^{\frac{n+1}{2}}}+1,\mathbf{2^{n-1+1}}\right\}$ [18], $\left\{\mathbf{2^{2n}},2^n\pm1,\mathbf{2^n}\pm\mathbf{2^{\frac{n+1}{2}}}+\mathbf{1}\right\}$ [18], both based on New CRT-I $\{\mathbf{2^n},\mathbf{2^n}\pm\mathbf{3},2^n\pm1\}$ [19], and $\{\mathbf{2^{2n}},\mathbf{2^n}\pm\mathbf{3},2^n\pm1\}$ [19], and $\{\mathbf{2^{2n}},\mathbf{2^n}\pm\mathbf{3},2^n\pm1\}$ for our two proposals.

To evaluate the scalability of the proposed reverse converters, experimental results for $DR=6n$, $4\le n<22$, were obtained for the best existing state of the art, namely for the moduli set $\{\mathbf{2^{2n}},2^n\pm1,\mathbf{2^{2n}}+\mathbf{1}\}$ [17], presenting the most area efficient conversion structure and $\left\{\mathbf{2^{n+\beta}},2^n\pm1,\mathbf{2^n}\pm\mathbf{2^{\frac{n+1}{2}}}+\mathbf{1}\right\}$ [18], for $\beta=0$ and $\beta=n$. It is important to note that the configuration with $\beta=n$ leads to the conversion structure with the best performance for $DR=6n$. Experimental results were also obtained for the few existing extensible solutions using CRT [1] and MRC [3] implementing a reverse converter for the moduli set $\{\mathbf{2^{2n}},\mathbf{2^n}\pm\mathbf{3},2^n\pm1\}$. Moreover, the architectures presented in [19], that use the New CRT-I technique, are also implemented for the moduli set $\{\mathbf{2^{n+\beta}},\mathbf{2^n}\pm\mathbf{3},2^n\pm1\}$, for $\beta=0$ and $\beta=n$. While additional dedicated reverse conversion structures for particular moduli sets, with $DR=6n$, exist in the state of the art, these are not herein considered since they present worst results than the reverse conversion structures presented in [17,18], as shown in [25].

Fig. 2 depicts the obtained results, for the resulting area, delay, Area-Delay-Product (ADP) and dynamic power. As expected, the dedicated conversion structures [17,18] present lower delays and area requirements. However, it is important to notice that they cannot be extended to larger moduli sets than $6n$ and $8n+1$ bits, respectively, which is the main goal of this work. Considering the traditional conversion approaches based on the CRT [1], the obtained area results demonstrate that they are not able to derive efficient ADP values. The MRC [3] solution shows a poor speedup providing high ADP values. When compared with the generic and scalable reverse approaches proposed in [19] ($\beta=0$ and $\beta=n$), the herein proposed approaches suggest a significant area reduction, in the order of 54% when $\beta=n$. Delay-wise, the proposed multi-level solutions are clearly slower than the architectures presented in [19], being on average 79% slower. On the other hand, in the 2-level approach the delay increase is just of 16% regarding in [19]. This less significant delay increase is achieved given the further parallelism exploited by the 2-level approach. Despite this delay increase, the area reduction is more than enough to compensate the delay increase if the Area-Delay-Product (APD) efficiency metric is considered. For the ADP, the multi-level approach allows for a 16% efficiency improvement, while the 2-level approach allows for an improvement of 41% with regard to [19] with $\beta=n$ and 71% in comparison with the CRT technique [1]. Between the two proposed approaches, the 2-level solution is clearly faster than the multi-level one, considering the analysed range. While the multi-level approach exhibits smaller area for some particular cases, it is not enough to compensate the delay degradation when ADP is considered. In terms of power, the proposals have similar consumption values. In comparison with the DR scalable architectures [19], for $\beta=0$ and MRC the proposals have less power consumption.

Table 2
Performance of the reverse converters for $DR=10n$–bits.

	$n=7$	$n=9$	$n=11$	$n=13$
Area (10^3 μm²) [19]	74	331	475	610
Area (10^3 μm²) 2-levels	91	152	197	240
Area reduction (%)	−23	**54**	**59**	**61**
Delay (ns) [19]	5.36	6.28	6.67	7.40
Delay (ns) 2-levels	5.23	5.97	6.41	6.82
Speedup	**1.02**	**1.05**	**1.04**	**1.08**
ADP (10^3 μm²) [19]	397	2079	3168	4514
ADP (10^3 μm²) 2-levels	476	907	1263	1637
ADP reduction (%)	−20	**56**	**60**	**64**
Power (*mW*) [19]	58	141	168	206
Power (*mW*) 2-levels	64	106	131	159
Power reduction (%)	−11	**25**	**22**	**23**

Given the above results, the following analysis only considers (i) the architecture presented in [19], which has shown to be the only efficient implementation in the state-of-the-art for large DRs and n values, and (ii) the two level approach herein proposed, given the overall better results when compared with the multi-level approach.

In order to better evaluate the scalability of the proposed conversion approaches, a moduli set with a $DR=10n$ bits is considered. To the best of the authors knowledge, no dedicated conversion structure has been proposed for moduli sets above $8n$ bits.

The scalable architecture presented in [19] for $\beta = n$ has been chosen for this comparison because it has the best ADP results as presented in Fig. 2. Table 2 depicts the obtained conversion metrics for the $DR = 10n$ bits moduli set of the form $\{\mathbf{2^{2n}}, \mathbf{2^n} \pm \boldsymbol{k_3}, \mathbf{2^n} \pm \boldsymbol{k_2}, \mathbf{2^n} \pm \boldsymbol{k_1}, 2^n \pm 1\}$, $7 \le n \le 13$. k_j are odd values chosen to define the most balanced set of conjugate prime architecture presented in [19] with $\beta = n$ and the 2-level approach herein proposed, namely $k_1 = 3$, $k_2 = 9$, $k_3 = 15$ for $n = 7, 11, 13$, and $k_1 = 3$, $k_2 = 9$, $k_3 = 21$ for n=9. These results suggest that for smaller values of n, the solution presented in [19] with $\beta = n$ exhibits better area, ADP and power metrics. Nevertheless, the herein proposed solution improves with n, suggesting area and power reductions up to 61%, and 25%, respectively, and a speedup of 1.08, resulting in ADP improvements up to 2.7 times, in the analysed range. More importantly, the proposed 2-level conversion approach is able to properly scale both in terms of DR and in the channel length (n).

6. Conclusions

In this work a novel method for designing RNS reverse converters for arbitrarily long moduli sets is proposed. Two approaches are proposed that reduce the modular weight selection of the multiplicative terms associated to the inputs. The first approach is based on iterative stages used to reduce the complexity of the final converter step. The second approach minimizes the number of required iterative stages in the conversion to only two levels, with minimal area cost in comparison with the multi-level solution. Experimental results suggest that the proposed approaches allow for significant area reductions in comparison with the state-of-the-art, for generic DR reverse conversion structures. Given the similar delay metrics between the state of the art and the proposed 2-level approach, ADP improvements up to 2.7 times can be achieved, when considering a moduli set with a dynamic range of $10n$-bits. More importantly, the obtained results suggest that the proposed approaches, in particular the two level approach, are able to efficiently scale with larger moduli sets and n.

Acknowledgements

This work was supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference UID/CEC/50021/2013.

References

[1] N. Szabo, Residue Arithmetic and its Applications to Computer Technology, McGraw-Hill, New York, 1967.
[2] L. Sousa, 2^n RNS scalers for extended 4-moduli sets, IEEE Trans. Comput. (February) .
[3] B. Parhami, Computer Arithmetic: Algorithms and Hardware Designs, 2nd edition, Oxford University Press, New York, 2010.
[4] Y. Wang, Residue-to-binary converters based on new Chinese remainder theorems, IEEE Trans. Circuits Syst. II: Analog Digit. Signal Process. 47 (March (3)) (2000) 197–205.
[5] J.-C. Bajard, L. Imbert, A full RNS implementation of RSA, IEEE Trans. Comput. 53 (June (6)) (2004) 769–774.
[6] Y.-Y. Wang, X. Song, M. Aboulhamid, H. Shen, Adder based residue to binary converters for $(2^n-1, 2^n, 2^n+1)$, IEEE Trans. Signal Process. 50 (July (7)) (2002) 1772–1779.
[7] R. Chaves, L- Sousa, $\{2^n+1, 2^{n+k}, 2^n-1\}$: a new RNS moduli set extension, in: IEEE Euromicro Symposium on Digital System Design: Architectures, Methods and Tools, IEEE Computer Society, Rennes, France, September 2004, pp. 210–217.
[8] P. Mohan, A. 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. Circuits and Systems I: Reg. Pap. 54 (June(6)) (2007) 1245–1254.
[9] M. Hosseinzadeh, A. Molahosseini, K. Navi, An improved reverse converter for the moduli set $\{2^n-1, 2^n, 2^n+1, 2^{n+1}-1\}$, IEICE Trans. Electron. Express 5 (September (17)) (2008) 672–677.
[10] B. Cao, T. Srikanthan, C. Chang, Efficient reverse converters for four-moduli sets $\{2^n-1, 2^n, 2^n+1, 2^{n+1}-1\}$ and $\{2^n-1, 2^n, 2^n+1, 2^{n-1}-1\}$, IEE Proc. Comput. Digit. Tech. 152 (September (5)) (2005) 687–696.
[11] B. Cao, C.-H. Chang, T. Srikanthan, A residue-to-binary converter for a new five-moduli set, IEEE Trans. Circuits Syst. I: Reg. Pap. 54 (May(5)) (2007) 1041–1049.
[12] A. Skavantzos, T. Stouraitis, Grouped-moduli residue number systems for fast signal processing, In: Proceedings of International Symposium on Circuits and Systems, vol. 3, May 1999, pp. 478–483.
[13] A. Skavantzos, An efficient residue to weighted converter for a new residue number system, in: Proceedings of Great Lakes Symposium on VLSI, February 1998, pp. 185–191.
[14] A. Hiasat, VLSI implementation of new arithmetic residue to binary decoders, IEEE Trans. Very Large Scale Integr. Syst. 13 (January (1)) (2005) 153–158.
[15] A. Skavantzos, A. Mohammad, S. Thanos, S. Dimitrios, Design of a balanced 8-modulus RNS, in: The 16th IEEE International Conference on Electronics, Circuits, and Systems, 2009. ICECS 2009, IEEE, Yasmine Hammamet, Tunisia, 2009, pp. 61–64.
[16] L. Sousa, S. Antão, MRC-based RNS reverse converters for the four-moduli sets $\{2^n+1, 2^n-1, 2^n, 2^{2n+1}-1\}$ and $\{2^n+1, 2^n-1, 2^{2n}, 2^{2n+1}-1\}$, IEEE Trans. Circuits Syst. II 59 (April (4)) (2012) 244–248.
[17] A. Molahosseini, K. Navi, C. Dadkhah, O. Kavehei, S. Timarchi, 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 Trans. Circuits Syst. I: Reg. Pap. 57 (April(4)) (2010) 823–835.
[18] H. Pettenghi, R. Chaves, L. Sousa, RNS reverse converters for moduli sets with dynamic ranges up to $(8n+1)$-bit, IEEE Trans. Circuits Syst. I: Reg. Pap. 60 (June(6)) (2013) 1487–1500.
[19] H. Pettenghi, R. Chaves, L. Sousa, Method to design general rns reverse converters for extended moduli sets, IEEE Trans. Circuits Syst. II: Analog Digit. Signal Process. 60 (December (12)) (2013) 877–881.
[20] S. Ma, J.-H. Hu, C.-H. Wang, A novel modulo 2^n-2^k-1 adder for residue number system, IEEE Trans. Circuits Syst. I: Reg. Pap. 60 (November(11)) (2013) 2962–2972.
[21] Y. Wang, New Chinese remainder theorems, in: Conference Record of the Thirty-Second Asilomar Conference on Signals, Systems & Computers, November 1998, vol. 1, pp. 165–171.
[22] D. Giorgos, D. Nikolos, High-speed parallel-prefix VLSI ling adders, IEEE Trans. Comput. 54 (1) (2005) 225–231.
[23] R. Matos, H. Pettenghi Mixed-Radix Conversion (MRC) Equations for the Moduli Set $\{2^{2n}, 2^n \pm 3, 2^n \pm 1\}$, DEEL-UFSC Technical Report, Dez. 2015 [Online]. Available: ⟨http://tele.sj.ifsc.edu.br/roberto.matos/RNS/MRC-equations_v1.pdf⟩.
[24] Virtual Silicon Technology, Inc., UMC High Density Standards Cells Library 90 nm CMOS Process, 2010.
[25] H. Ahmadifar, G. Jaberipur, A new residue number system with 5-moduli set: $\{2^{2q}, 2^q \pm 3, 2^q \pm 1\}$, Comput. J. (September) .