Journal of Signal Processing Systems
https://doi.org/10.1007/s11265-018-1434-z

Sign Identifier for the Enhanced Three Moduli Set $\{2^{n+k}, 2^n - 1, 2^{n+1} - 1\}$

Ahmad Hiasat[1] · Leonel Sousa[2,3]

Received: 10 April 2018 / Revised: 29 October 2018 / Accepted: 25 December 2018

Abstract

The three-moduli set $\{2^n, 2^n - 1, 2^{n+1} - 1\}$ started to receive more attention lately. This moduli set is considered an arithmetic-friendly set because it avoids the demanding channel $(2^n + 1)$ of the traditional 3-moduli set $\{2^n, 2^n - 1, 2^n + 1\}$. This work considers an enhanced form of the above moduli set, $\{2^{n+k}, 2^n - 1, 2^{n+1} - 1\}$, and proposes a sign identifier for numbers within the dynamic range of the set. While the published sign identifiers have dealt with the unextended form $\{2^n, 2^n - 1, 2^{n+1} - 1\}$, this is the first sign identifier that deals with the extended form. Based on VLSI layout synthesis for the case $(k = 0)$, the proposed structure has less or similar area and power requirements, nevertheless, it achieves an improved time performance in the range of (13.0–29.6)% compared with the most recent sign identifiers. When compared with a recently published residue-to-binary converter for the moduli set $\{2^{n+k}, 2^n - 1, 2^{n-1} - 1\}$, which can function as a converter-based sign identifier, the proposed detector has on average reduced area, time, and power by 175%, 106%, and 60%, respectively.

Keywords Residue number system · Computer arithmetic · Digital circuits · VLSI design

1 Introduction

The Residue Number System (RNS) is a non-conventional representation, where numbers are represented using a group of positive and relatively prime integers called a moduli set. Using RNS, addition, subtraction, and multiplication are carried on all residue digits in parallel [1–3]. There is no carry that propagates from one RNS digit to another. Therefore, RNS is used in applications that require high computing speed such as public-key cryptography, digital signal processing, and communications systems [1, 2, 4–7], where RNS provides better time performance and energy dissipation in many real time and embedded processors.

Other arithmetic operations like division and sign identification are not easy to perform [1, 2, 8]. Many decision-making RNS-based arithmetic operations, such as division, are also dependent on the sign of a number, or on comparing two numbers. A considerable number of papers have been published so far that introduced designs for sign identifiers/detectors and magnitude comparators for 3-moduli sets [9–18]. The bulk of these papers have focused on the moduli set $\{2^n, 2^n - 1, 2^n + 1\}$ or its expanded form $\{2^{n+k}, 2^n - 1, 2^n + 1\}$. Less work was published for sign detection structures for the 3-moduli set $\{2^n, 2^n - 1, 2^{n+1} - 1\}$ [17, 18], and none dealt with its expanded form $\{2^{n+k}, 2^n - 1, 2^{n+1} - 1\}$. Sign identifiers for 4- and 5-moduli sets have been recently considered [19–22].

The moduli set $\{2^n, 2^n - 1, 2^{n+1} - 1\}$, or its enhanced form, avoids the relatively demanding channel $(2^n + 1)$ which exists in the traditional set $\{2^n, 2^n - 1, 2^n + 1\}$. A considerable number of papers showed that the arithmetic units, such as adders and multipliers, of the $(2^n + 1)$ channel are significantly more area, time, and power demanding than the $(2^n - 1)/(2^{n+1} - 1)$ channels [23–25].

Adders and multipliers modulo $(2^n - 1)$ or modulo $(2^{n+1} - 1)$ are slightly slower than modulo 2^{n+k} if $k =$

✉ Ahmad Hiasat
a.hiasat@psut.edu.jo

Leonel Sousa
leonel.sousa@inesc-id.pt

[1] Computer Engineering Department, Princess Sumaya University for Technology, Amman, Jordan

[2] Department of Electrical and Computer Engineering, Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal

[3] Instituto de Engenharia e de Sistemas de Computadores (INESC-ID), Lisboa, Portugal

Published online: 12 January 2019

0. Therefore, to balance the three channels, $(2^{n+k}, 2^n - 1, 2^{n+1} - 1)$, the value for k (greater than 0) has to be selected to make all the three channels operate at the same speed. The balancing process (i. e. selecting a specific value for k) takes into consideration the characteristics of the architectures and the technology used in their implementation. This makes the enhanced moduli set $\{2^{n+k}, 2^n - 1, 2^{n+1} - 1\}$ an arithmetic-friendly and more appealing than the traditional 3-moduli set.

This paper is organized as follows: Section 2 defines the notational convention adopted through out this paper. Section 3 introduces the analysis of the sign identification approach used for the set of moduli defined by $\{2^{n+k}, 2^n - 1, 2^{n+1} - 1\}$, where n is a positive integer and $(0 \leq k \leq n)$. It also proposes the first hardware structure dedicated to this enhanced moduli set. Section 4 provides a theoretical and an experimental evaluation for the proposed sign identifier and compares it with the available circuits of a similar moduli set.

2 Notational Convention

Defining a set of N relatively prime positive integers, $\{m_1, m_2, \cdots m_N\}$, where m_i is called a modulus, the dynamic range of this set is given by $M = \prod_{i=1}^{N} m_i$. Any integer $X \in [0, M - 1]$, is uniquely represented by N-tuple RNS-digits $(R_1, R_2, \cdots R_N)$, where:

- For each $i \in 1, 2, \cdots, N$, $R_i = \langle X \rangle_{m_i}$, that is, R_i denotes the least non-negative remainder resulting from dividing X by m_i.
- Each R_i is represented by a set of l_i bits given by $R_i = r_{(i,l-1)} \cdots r_{(i,0)}$, where $l_i = \lceil \log_2 m_i \rceil$ and $\lceil . \rceil$ is the smallest integer equal or greater than (.).
- $\hat{m}_i$ is given by the formula $\hat{m}_i = M/m_i$.
- $\left\langle m_i^{-1} \right\rangle_{m_i}$ is the multiplicative inverse of $\hat{m}_i$ with respect to m_i, where $\left\langle \hat{m}_i . \hat{m}_i^{-1} \right\rangle_{m_i} = 1$.

3 Sign Detection

The motivation of this work is to design a significantly faster sign identifier for the three moduli set $\{2^{n+k}, 2^n - 1, 2^{n+1} - 1\}$, without imposing additional area or power requirements. Therefore, in this section, the sign identification analysis of this enhanced moduli set is introduced first. Then the analysis is customized for the specific case of $k = 0$ for comparison with other similar sign identifiers.

3.1 Enhanced Moduli Set Analysis

Based on the new mixed-radix Chinese-Reminder-Theorem proposed in [10], the sign detection theorem was introduced in [17]. It states that for a moduli set given by $\{m_1 = 2^p, m_2, \ldots\ldots, m_N\}$, where p is a positive integer, the variable α is an indicator of the sign of X, where:

$$\alpha = \left\langle \sum_{i=1}^{N} \frac{m_1 \left\langle \hat{m}_i^{-1} \right\rangle_{m_i}}{m_i} R_i - \frac{1}{\prod_{i=2}^{N} m_i} R_N \right\rangle_{2^p} , \tag{1}$$

For the case $p = n + k$, the work in [17] shows that:

$$\text{sign}(X) = \begin{cases} 0, & \text{when } \alpha \in [0, 2^{n+k-1} - 1] \\ 1, & \text{when } \alpha \in [2^{n+k-1}, 2^{n+k} - 1] \end{cases} . \tag{2}$$

The most significant bit of α represents the value of sign(X), where the binary representation of α is: $\overbrace{\alpha_{n+k-1} \cdots \alpha_1 \alpha_0}^{p=n+k}$. Therefore,

$$\text{sign}(X) = \alpha_{n+k-1} . \tag{3}$$

Equivalently, Eqs. 2 and 3 can be written as:

$$\text{sign}(X) = \begin{cases} \text{Positive}, & \text{when } \alpha_{n+k-1} = 0 \\ \text{Negative}, & \text{when } \alpha_{n+k-1} = 1 \end{cases} . \tag{4}$$

For the enhanced 3-moduli set under consideration, $m_1 = 2^{n+k}$, $m_2 = (2^n - 1)$, and $m_3 = (2^{n+1} - 1)$. The multiplicative inverses of the moduli set $\{2^{n+k}, 2^n - 1, 2^{n+1} - 1\}$, are:

$$\left\langle \hat{m}_1^{-1} \right\rangle_{m_1} = \langle (3)2^n + 1 \rangle_{2^{n+k}} ,$$

$$\left\langle \hat{m}_2^{-1} \right\rangle_{m_2} = \left\langle 2^{n-k} \right\rangle_{(2^n-1)} ,$$

$$\left\langle \hat{m}_3^{-1} \right\rangle_{m_3} = \left\langle -2^{n-k+3} \right\rangle_{(2^{n+1}-1)} . \tag{5}$$

The proofs of the multiplicative inverses of Eq. 5 are:

- Proof of $\left\langle \hat{m}_1^{-1} \right\rangle_{m_1} = \langle (3)2^n + 1 \rangle_{2^{n+k}}$.
  $\left\langle (2^{n+1} - 1)(2^n - 1) \right\rangle_{2^{n+k}} = \langle (-3(2^n) + 1) \rangle_{2^{n+k}}$.
  Therefore,
  $$\left\langle ((-3)2^n + 1)((3)2^n + 1) \right\rangle_{2^{n+k}} = \left\langle (-9)2^{2n} + 1 \right\rangle_{2^{n+k}} , = 1.$$
- Proof of $\left\langle \hat{m}_2^{-1} \right\rangle_{m_2} = \left\langle 2^{n-k} \right\rangle_{2^n-1}$.
  $\left\langle (2^{n+k})(2^{n+1} - 1) \right\rangle_{2^n-1} = \left\langle (2^k)(1) \right\rangle_{2^n-1}$.
  Therefore,
  $$\left\langle (2^k)(2^{n-k}) \right\rangle_{2^n-1} = \left\langle 2^n \right\rangle_{2^n-1} = 1.$$

- Proof of $\left\langle \hat{m}_3^{-1} \right\rangle_{m_3} = \langle -2^{n-k+3} \rangle_{2^n-1}$.
  $\langle (2^{n+k})(2^n - 1) \rangle_{2^{n+1}-1} = \langle -2^{k-2} \rangle_{2^{n+1}-1}$.
  Therefore,

$$\left\langle -2^{k-2}(-2^{n-k+3}) \right\rangle_{2^{n+1}-1} = \left\langle 2^{n+1} \right\rangle_{2^{n+1}-1} = 1.$$

Substituting the moduli m_1, m_2, and m_3 and the corresponding multiplicative inverses in Eq. 1 results in:

$$\begin{aligned} \alpha = \Bigg\langle \Bigg\lfloor & ((3)2^n + 1)R_1 + \frac{2^{n+k}(2^{n-k})}{(2^n - 1)} R_2 \\ & + \frac{2^{n+k}(-2^{n-k+3})}{(2^{n+1}-1)} R_3 - \frac{1}{(2^{n+1}-1)(2^n-1)} R_3 \Bigg\rfloor \Bigg\rangle_{2^{n+k}} . \end{aligned} \tag{6}$$

considering the second term on the right-hand side of Eq. 6,

$$\begin{aligned} \frac{2^{n+k}(2^{n-k})}{(2^n - 1)} R_2 &= \frac{2^{2n}}{(2^n - 1)} R_2 \\ &= \underbrace{(2^n + 1)R_2 + \frac{R_2}{2^n - 1}}_{1} \end{aligned}$$

Similarly, considering the third term on the right-hand side of Eq. 6,

$$\begin{aligned} \frac{2^{n+k}(-2^{n-k+3})}{(2^{n+1} - 1)} R_3 &= \frac{-2^{2n+3}}{(2^{n+1} - 1)} R_3 \\ &= \underbrace{-2(2^{n+1} + 1)R_3 - \frac{2R_3}{2^{n+1} - 1}}_{2} \end{aligned}$$

Substituting the above quantities 1 and 2 into Eq. 6:

$$\begin{aligned} \alpha = \Bigg\langle & ((3)2^n + 1)R_1 + (2^n + 1)R_2 + -2(2^{n+1} + 1)R_3 \\ & + \left\lfloor -\frac{2R_3}{2^{n+1}-1} + \frac{R_2}{2^n-1} - \frac{R_3}{(2^{n+1} - 1)(2^n-1)} \right\rfloor \Bigg\rangle_{2^{n+k}} \end{aligned} \tag{7}$$

The fractional part in Eq. 7 is defined to be f and simplified as follows:

$$\begin{aligned} \mathrm{f} &= \left\lfloor -\frac{2R_3}{2^{n+1}-1} + \frac{R_2}{2^n-1} - \frac{R_3}{(2^{n+1}-1)(2^n-1)} \right\rfloor , \\ &= \left\lfloor \frac{-2(2^n-1)R_3 + (2^{n+1}-1)R_2 - R_1}{(2^{n+1}-1)(2^n-1)} \right\rfloor , \\ &= \left\lfloor \frac{R_2 - R_3}{(2^n-1)} \right\rfloor . \end{aligned} \tag{8}$$

Since $R_3 = 2^n r_{(3,n)} + R'_3$, where $R'_3 = \overbrace{r_{(3,n-1)} \cdots r_{(3,0)}}^{n}$, then Eq. 8 can be rewritten as:

$$\begin{aligned} \mathrm{f} &= \left\lfloor \frac{R_2 - 2^n r_{(3,n)} - R'_3}{(2^n-1)} \right\rfloor , \\ &= \left\lfloor \frac{R_2 - r_{(3,n)} - R'_3}{(2^n-1)} \right\rfloor - r_{(3,n)} , \\ &= u - r_{(3,n)} . \end{aligned} \tag{9}$$

where:

$$u = \begin{cases} 0, & \text{when } R_2 \geq R'_3 + r_{(3,n)} \\ -1, & \text{otherwise} \end{cases} . \tag{10}$$

Substituting Eq. 9 into Eq. 7 produces:

$$\begin{aligned} \alpha = \Big\langle & -2(2^{n+1} + 1)R_3 + (2^n + 1)R_2 \\ & + ((3)2^n + 1)R_1 + u - r_{(3,n)} \Big\rangle_{2^{n+k}} . \end{aligned} \tag{11}$$

Defining the two variables $\widetilde{R}_3 = \langle 2(2^{n+1} + 1)R_3 + r_{(3,n)} \rangle_{2^{n+k}}$, and $\widetilde{R}_2 = \langle (2^n + 1)R_2 \rangle_{2^{n+k}}$, then:

$$\begin{aligned} \widetilde{R}_3 &= \left\langle 2(2^{n+1} + 1)R_3 + r_{(3,n)} \right\rangle_{2^{n+k}} \\ &= \left\langle 2^{n+2}R_3 + 2R_3 + r_{(3,n)} \right\rangle_{2^{n+k}} . \end{aligned} \tag{12}$$

Since $R_3 = \overbrace{r_{(3,n)} \cdots r_{(3,0)}}^{n+1}$, then

$$2^{n+2}R_3 = \overbrace{r_{(3,n)} \cdots r_{(3,0)}}^{n+1} \overbrace{0 \cdots \cdots 0}^{n+2} .$$

When applying modulo 2^{n+k} to the last expression, only the least significant $(n + k)$ bits are considered,

$$2^{n+2}R_3 = \overbrace{r_{(3,k-3)} \cdots r_{(3,0)}}^{k-2} \overbrace{0 \cdots \cdots 0}^{n+2} .$$

Similarly

$$2R_3 + r_{(3,n)} = \overbrace{r_{(3,n)} \cdots r_{(3,1)} r_{(3,0)} r_{(3,n)}}^{n+2} .$$

Therefore, Eq. 12 can be rewritten as a single binary word:

$$\widetilde{R}_3 = \overbrace{r_{(3,k-3)} \cdots r_{(3,0)}}^{k-2} \overbrace{r_{(3,n)} \cdots r_{(3,0)} r_{(3,n)}}^{n+2} . \tag{13}$$

Following a similar approach, $\widetilde{R}_2 = \langle (2^n + 1)R_2 \rangle_{2^{n+k}}$ can be written in binary format as:

$$\begin{aligned} \widetilde{R}_2 &= \left\langle (2^n + 1)R_2 \right\rangle_{2^{n+k}} , \\ &= \overbrace{r_{(2,k-1)} \cdots r_{(2,0)}}^{k} \overbrace{r_{(2,n-1)} \cdots r_{(2,0)}}^{n} . \end{aligned} \tag{14}$$

Substituting Eqs. 13 and 14 in Eq. 11 produces:

$$\alpha = \langle -\widetilde{R}_3 + \widetilde{R}_2 + R_1 + 2^{n+1}R_1 + 2^n R_1 + u \rangle_{2^{n+k}} \tag{15}$$

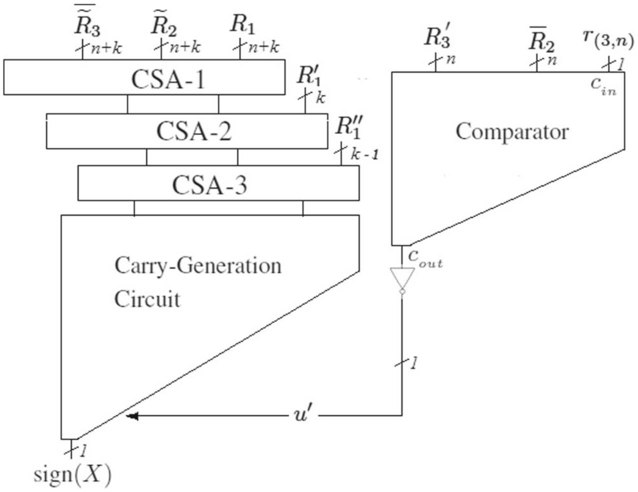

Figure 1 Block diagram of the proposed sign detector. The carry-generation circuit is given in Fig. 2 and the comparator circuit is given Fig. 3.

Defining $R_1' = \langle 2^n R_1 \rangle_{2^{n+k}} = \overbrace{r_{(1,k-1)} \cdots r_{(1,0)}}^{k} \overbrace{0 \cdots 0}^{n}$, and $R_1'' = \langle 2^{n+1} R_1 \rangle_{2^{n+k}} = \overbrace{r_{(1,k-2)} \cdots r_{(1,0)}}^{k-1} \overbrace{0 \cdots 0}^{n+1}$, and observing that: $\langle -\widetilde{R}_3 + u \rangle_{2^{n+k}} = \langle \overline{\widetilde{R}}_3 + 1 + u \rangle_{2^{n+k}} = \langle \overline{\widetilde{R}}_3 + u' \rangle_{2^{n+k}}$, where $\overline{\widetilde{R}}_3$ is the one's complement of $\widetilde{R}_3$, and $u' = u + 1$, then:

$$u' = \begin{cases} 1, & \text{when } R_2 \geq r_{(3,n)} + R_3' \\ 0, & \text{otherwise} \end{cases}. \tag{16}$$

Substituting the above definitions in Eq. 15:

$$\alpha = \left\langle \overline{\widetilde{R}}_3 + \widetilde{R}_2 + R_1 + R_1' + R_1'' + u' \right\rangle_{2^{n+k}}. \tag{17}$$

The implementation of Eq. 17 is shown in Fig. 1. Three Carry-Save Adders (CSAs) are used with different sizes. CSA-1 consists of $(n+k)$ Full-Adders (FA). CSA-2 consists of k FAs, while CSA-3 consists of $(k-1)$ FAs. The comparator circuit is a binary n-bit adder that has been stripped down to compute the output carry (c_{out}) resulting from computing $(R_3' + \overline{R}_2 + r_{(3,n)})$, where $r_{(3,n)}$ is considered an input carry (c_{in}) to the adder. Figures 2 and 3 show the gate level implementation of the carry-generation circuit and the comparator of Fig. 1, for the case $n = 8$. The carry-generation circuit of Fig. 1 is a modulo 2^{n+k} binary adder that has also been stripped down to compute sign(X) adopting a similar approach to that in [16]. The delay of this proposed sign detector is the delay of the CSA network plus the delay of the carry-generation circuit.

Numerical Example It is intended to determine the sign of $X = (R_1, R_2, R_3) = (57, 12, 9)$, where the moduli set is $(64, 15, 31)$, $n = 4$ and $k = 2$. Since $R_3 = 01001$, using (12) $\overline{\widetilde{R}}_3 = 101101 \rightarrow 45$. Since $R_2 = 1100$, using (14) $\widetilde{R}_2 = 001100 \rightarrow 12$. Since $R_1 = 11\ 1001$, then $R_1' = 01\ 0000 \rightarrow 16$, $R_1'' = 1\ 00000 \rightarrow 32$. Using (16), $u' = 1$. Applying (17), $\alpha = \langle 45+12+57+16+32+1 \rangle_{64} = 35 \rightarrow \underline{1}0\ 0011$. Thus, $\alpha_{n+k-1} = 1$. X is negative, where $X = 16377 \in \left[\frac{M}{2}, M-1\right]$.

3.2 Unextended Moduli Set Analysis

Considering the specific case of $k = 0$ (unextended set), then: $\widetilde{R}_2 = R_2$ and $R_1' = R_1'' = 0$. Hence, Eq. 17 reduces to

$$\alpha_{k=0} = \left\langle R_3'' + R_2 + R_1 + u' \right\rangle_{2^n}. \tag{18}$$

where:

$$R_3'' = \overbrace{\overline{r}_{(3,n-2)} \cdots \overline{r}_{(3,0)} \overline{r}_{(3,n)}}^{n} \tag{19}$$

$$R_2 = \overbrace{r_{(2,n-1)} \cdots r_{(2,0)}}^{n} \tag{20}$$

$$R_1 = \overbrace{r_{(1,n-1)} \cdots r_{(1,0)}}^{n} \tag{21}$$

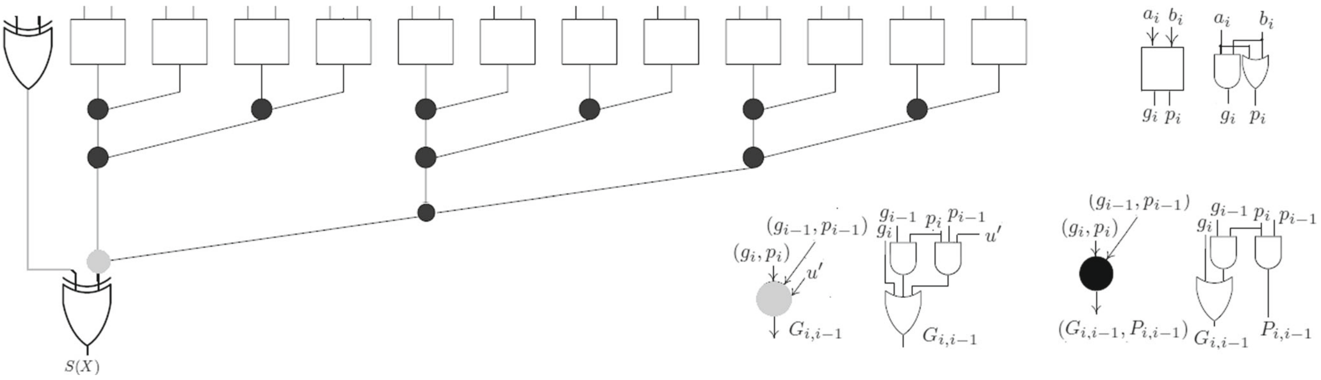

Figure 2 Gate-level implementation of the carry-generation circuit of Fig. 1 for the case $n = 8$ and $k = 5$.

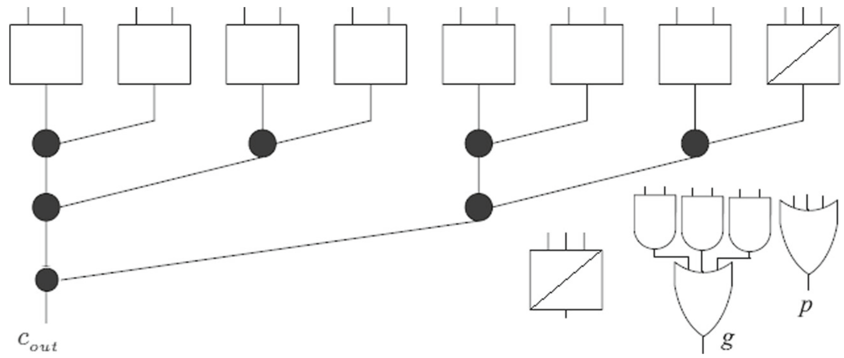

Figure 3 Gate level implementation of the comparator of Figs. 1 and 4, for the case $n = 8$ and $0 \leq k \leq n$.

The hardware implementation of Eq. 18 is shown in Fig. 4. The carry-generation circuit of Fig. 4 is shown in Fig. 5. The Comparator circuit of Fig. 4 is the same shown in Fig. 3, which works for all values of k.

4 Evaluation, and Comparison

The proposed sign identifier and the competitive work are evaluated theoretically and experimentally. The theoretical evaluation is based on using the Unit-Gate (U-G) model, while the experimental evaluation is based on VLSI circuit implementation. The competitive functionally-similar sign identifiers available in the literature are the ones introduced in [17] and [18]. Both works deal with the 3-moduli set $\{2^n, 2^n - 1, 2^{n+1} - 1\}$. The sign identifiers presented in [17] and [18] can be considered a special case of the more general one proposed in this paper. Moreover, there is no published work that presents a sign identifier for the enhanced moduli set $\{2^{n+k}, 2^n - 1, 2^{n+1} - 1\}$. However, [25] presented a residue to binary converter for the moduli set $\{2^{n+k}, 2^n - 1, 2^{n-1} - 1\}$, which can also be used as a sign identifier for this set.

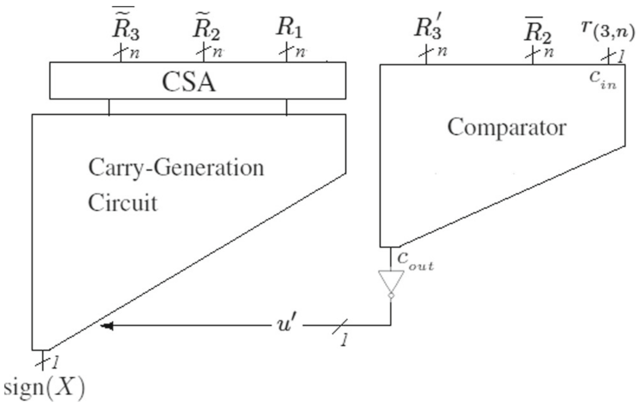

Figure 4 Block diagram of the proposed sign detector for the case $n = 8$ and $k = 0$, where the carry-generation circuit is given in Fig. 5 and the Comparator circuit is given Fig. 3.

4.1 Theoretical Model

In the U-G model [23], a two input gate (e. g. AND, NAND, OR, NOR) is considered to have an area of one unit and a delay of one unit. The Full-Adder (FA) is considered to have an area of 7 units and a delay of 4 units. A Half-Adder (HA) has an area of 3 units and a delay of 2 units. An XOR/XNOR gate has an area of 2 units and a delay of 2 units. The model ignores inverters.

The three CSAs of Fig. 1 require a total of $7(n + 3k - 1)$ units of area and 12 units of delay. The other two constituents of the sign detector are a carry-generation circuit and a comparator, both simplified structures of binary adders. The carry-generation circuit is a simplified modulo 2^{n+k} binary adder. Therefore, this circuit requires $5(n + k)$ units of area and $\left(2\lceil\log_2(n+k)\rceil + 3\right)$ units of delay. Similarly, the comparator is also a simplified modulo 2^n binary adder that requires $5n$ units of area and imposes no additional delay. Therefore, in total, the proposed sign detector requires $(17n + 26k - 7)$ area units and $\left(2\lceil\log_2(n+k)\rceil + 15\right)$ time units.

When considering the case $k = 0$, the requirements of the proposed sign identifier reduce to $(17n - 7)$ area units and $\left(2\lceil\log_2 n\rceil + 7\right)$ time units. Recently, two sign detector for the three moduli $\{2^n, 2^n - 1, 2^{n+1} - 1\}$ were published [17, 18]. The area and time requirements of [17] are $(19n - 4)$ units and $\left(2\lceil\log_2 n\rceil + 9\right)$, respectively. The work of [18] presented two structures. The Kogge-Stone-based (KSB) structure requires $(17n - 1)$ area units and $\left(2\lceil\log_2 n\rceil + 9\right)$ time units. However, the Ling-based (LB) structure imposes $(16n - 1)$ area units and $\left(2\lceil\log_2 n\rceil + 11\right)$, where the time of this structure was reported mistakingly in [18] as $\left(2\lceil\log_2 n\rceil + 6\right)$. The converter-based sign detector

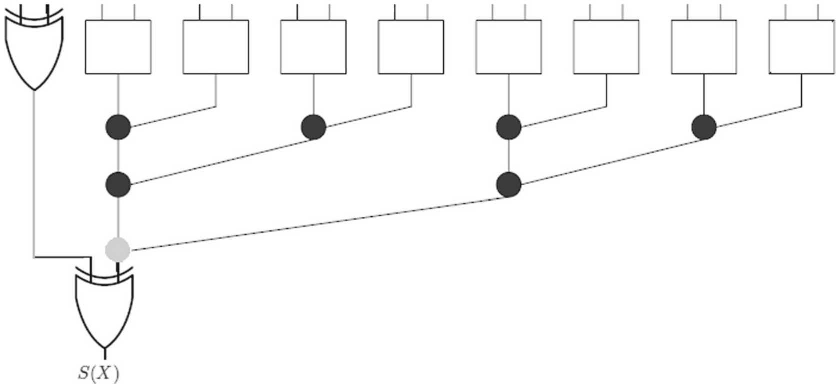

Figure 5 Gate-level implementation of carry-generation circuit of Fig. 1 for the case $n = 8$ and $k = 0$.

presented in [25] requires an area of $(9n\lceil\log_2 n\rceil + 39n + 8k)$ units and a time delay of $(6\lceil\log_2 n\rceil + 27)$ units. Table 1 shows the requirements of all sign detectors under consideration.

4.2 Experimental VLSI Realization

The proposed structure and the other state-of-the-art sign identifiers of the moduli set under consideration were modeled using Verilog Hardware Description Language. The modeling was performed for different values of n and the permissible values of k. The Synopsys Design Compiler (Version G-2012.06) was utilized to conduct the synthesis (at 1.8 V for the core voltage and 25 °C for the temperature). All modeled structures were mapped to 65 nm the Synopsys DesignWare Logic Libraries. The Synopsys Simulator was also utilized to check functionality correctness the modeled circuits. Synopsys Power Compiler was utilized to estimate the consumed power, while the Synopsys IC Compiler was also utilized to perform placement and routing. Table 2 lists the experimental results of all structures under consideration, [17, 18] and [25], for different values of n and

Table 1 Hardware and time requirements of different sign identifiers based on the U-G model.

Sign detector	Area	Time
[17] $(k = 0)$	$19n - 4$	$2\lceil\log_2 n\rceil + 9$
[18] $(k = 0)$-KSB	$17n - 1$	$2\lceil\log_2 n\rceil + 9$
[18] $(k = 0)$-LS	$16n - 1$	$2\lceil\log_2 n\rceil + 11$
Proposed $(k = 0)$	$17n - 7$	$2\lceil\log_2 n\rceil + 7$
[25] $(0 \leq k \leq n)$ converter-based	$9n\lceil\log_2 n\rceil$ $+39n + 8k$	$6\lceil\log_2 n\rceil + 27$
Proposed $(1 \leq k \leq n)$	$17n + 26k - 7$	$2\lceil\log_2 (n + k)\rceil + 15$

Table 2 VLSI experimental results of different sign identifiers.

Sign identifier	n	k	Area μm^2	Delay ps	Power μW
[17]	7	0	593	749	46.2
	11	0	878	824	58.4
	15	0	1149	829	69.7
[18]	7	0	538	735	43.4
KSB	11	0	761	801	56.3
Structure	15	0	1036	815	66.8
[18]	7	0	501	827	41.1
LB	11	0	723	905	51.6
Structure	15	0	987	922	60.4
[25]	7	0	1844	1342	74.6
Converter-based		2	1881	1375	77.4
		5	1932	1388	78.3
		7	1988	1402	79.8
	11	0	3147	1486	98.2
		4	3243	1498	99.8
		8	3361	1531	102.3
		11	3427	1547	105.1
	15	0	4468	1587	110.2
		5	4520	1623	112.0
		10	4575	1638	114.6
		15	4792	1644	117.2
Proposed	7	0	525	638	42.3
		2	695	689	47.7
		5	997	713	55.1
		7	1202	719	62.8
	11	0	750	709	53.4
		4	1124	721	61.9
		8	1492	752	66.4
		11	1874	760	69.0
	15	0	1015	720	63.1
		5	1485	762	68.2
		10	1792	771	70.4
		15	2387	793	73.2

Table 3 Relative performance of competitive sign identifiers compared with the proposed one based on experimental results.

Sign Identifier	n	k	AR %	DR%	PR%
[17]	7	0	13.0	17.4	9.2
	11	0	17.1	16.2	9.6
	15	0	13.2	15.1	10.5
[18] KSB structure	7	0	2.5	15.2	2.6
	11	0	1.5	13.0	5.4
	15	0	2.1	13.2	5.9
[18] LB structure	7	0	−4.6	29.6	−2.8
	11	0	−3.6	27.6	−3.4
	15	0	−2.8	28.1	−4.3
[25] Converter- based	7	0	251	110	76
		2	171	100	62
		5	94	95	42
		7	65	95	27
	11	0	320	110	84
		4	189	108	61
		8	125	104	54
		11	83	104	52
	15	0	340	120	75
		5	204	113	64
		10	155	112	63
		15	101	107	60

k. Similar to the other structures, the proposed architecture works for all odd and even values of n. Table 3 lists the relative performance of [17, 18], and [25] as compared with the proposed structure, where the proposed structure is used as a reference. The percentage of reductions in area (AR), delay (DR), and power (PR) are computed using the formula: $\frac{\text{other work - proposed}}{\text{proposed}} \times 100\%$. Compared with the work of [17], the proposed sign detector (for the case $k = 0$) has reduced area, time, and power by an average of 14.4%, 16.3%, and 9.7%, respectively. The proposed structure for the same case of ($k = 0$) has almost similar area and power performance when compared with both structures of [18]. However, it has an average delay reduction of 13.8% when compared with the first structure and 28.4% when compared with the second structure of [18]. When compared with the converter-based sign detector of [25], the proposed detector has on average reduced area, time, and power by 175%, 106%, and 60%, respectively. Figure 6 shows the behavioral trend of area, time, and power of the proposed sign identifier as function of k, based on the experimental results listed in Table 2.

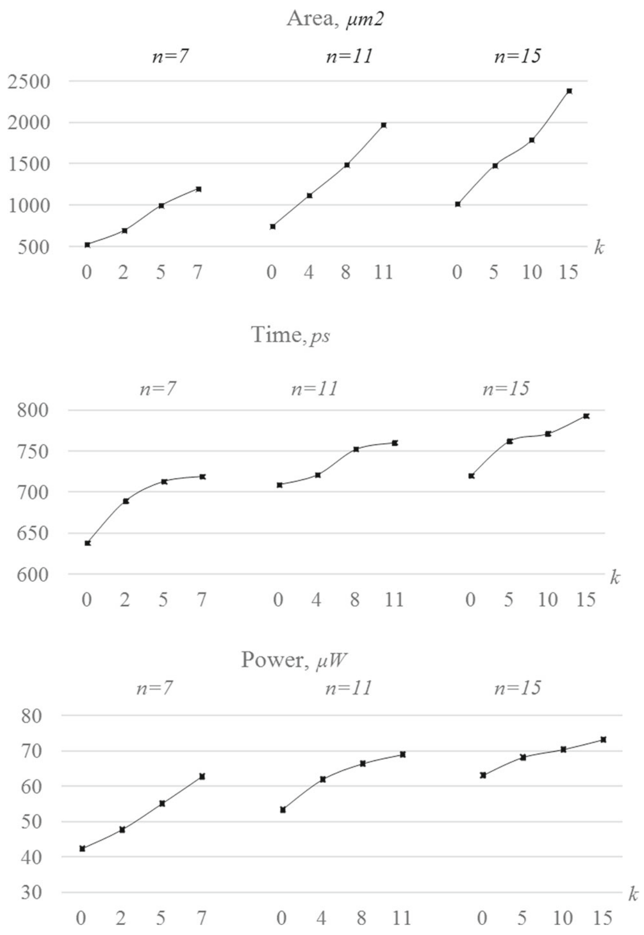

Figure 6 The behavioral trend of area, time, and power of the proposed sign identifier as function of k, based on the experimental results listed in Table 2.

5 Conclusions

This paper suggested the first sign identifier for the enhanced and arithmetic-friendly three-moduli set $\{2^{n+k}, 2^n - 1, 2^{n+1} - 1\}$, $0 \leq k \leq n$. The only sign identifiers available in the literature for this set dealt with the specific case of $k = 0$. While the proposed structure has less or comparable area and power requirements compared with the most recent published circuits, the proposed work of the unextended form showed a considerable time improvement in the range of 13% to 29.6%. However, when compared with a RNS-to-Binary converter which can perform the sign detection for the moduli set $\{2^{n+k}, 2^n - 1, 2^{n-1} - 1\}$, the proposed detector has an improved area, time, and power performance by 175%, 106%, and 60%, respectively.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

1. Soderstrand, M.A., Jenkins, W.K., Jullien, G., Taylor, F. (Eds.) (1986). *Residue number system arithmetic: modern applications in digital signal processing*. New York: IEEE Press.
2. Mohan, P.V. (2016). *Residue number systems: theory and applications*. Birkhauser: Switzerland.
3. Hiasat, A. (2004). A suggestion for a fast residue multiplier for a family of moduli of the form $(2^n - (2^p \pm 1))$. *Computer Journal*, *47*(1), 93–102.
4. Sousa, L., Antao, A., Martins, P. (2016). Combining residue arithmetic to design efficient cryptographic circuits and systems. *IEEE Circuits and Systems Magazine*, *16*(4), 6–32.
5. Hiasat, A., & Al-Khateeb, A. (1998). Efficient digital sweep oscillator with extremely low sweep rates. *IEE CD&S*, *145*(6), 409–414.
6. Dutta, C.B., Garai, P., Sinha, A. (2012). Design of a reconfigurable DSP processor with bit efficient residue number system. *International Journal of VLSI Design & Commmunication System (VLSICS)*, *3*(5), 175–189.
7. Wei, J., Guo, W., Liu, H., Tan, Y. (2013). A unified cryptographic processor for RSA and ECC in RNS. In *Communications in computer and information science* (pp. 19–32). Berlin: Springer.
8. Hiasat, A., & Zohdy, H. (1997). Design and implementation of an RNS division algorithm. In *IEEE symposium on computer arithmetic* (pp. 240–249).
9. Hiasat, A. (1993). New designs for a sign detector and a residue to binary converter. *IET Proceedings - Circuits, Devices and Systems*, *140*(4), 247–252.
10. Bi, S., & Gross, W. (2008). The mixed-radix Chinese remainder theorem and its applications to residue comparison. *IEEE Transactions on Computers*, *57*(12), 1624–1632.
11. Tomczak, T. (2008). Fast sign detection for RNS $(2^n - 1, 2^n, 2^n + 1)$. *IEEE Transactions on Circuits and Systems I*, *55*(6), 1502–1511.
12. Pettenghi, H., Chaves, R., Sousa, L. (2004). $(2^n + 1, 2^{n+k}, 2^n - 1)$: a new RNS moduli set extension. In *Proceedings of the EUROMICRO systems on digital system design (DSD'04)* (pp. 210–217).
13. Xu, M., Yao, R., Luo, F. (2012). Low-complexity sign detection algorithm for RNS $\{2^n - 1, 2^n, 2^n + 1\}$. *IEICE Transactions on Electronics*, *95*(9), 1552–1556.
14. Kumar, S., & Chang, C.-H. (2016). New fast and area-efficient adder-based sign detector for RNS $2^n - 1, 2^n, 2^n + 1$. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, *24*(7), 2608–2612.
15. Sousa, L., & Martins, P. (2016). Sign detection and number comparison on RNS 3-moduli sets $\{2^n - 1, 2^{n+x}, 2^n + 1\}$. *Circuits, Systems, and Signal Processing*, *36*(3), 1224–1246.
16. Hiasat, A. (2016). A sign detector for a group of three-moduli sets. *IEEE Transactions on Computers*, *65*(12), 3580–3590.
17. Xu, M., Bian, Z., Yao, R. (2015). Fast sign detection algorithm for the RNS moduli set $\{2^{n+1} - 1, 2^n - 1, 2^n\}$. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, *23*(2), 379–383.
18. Niras, C.V., & Kong, Y. (2016). Fast sign-detection algorithm for residue number system moduli set $\{2^n - 1, 2^n, 2^{n+1} - 1\}$. *IET Computers and Digital Techniques*, *10*(2), 54–58.
19. Chang, C.-H., & Kumar, S. (2017). Area-efficient and fast sign detection for four-moduli set RNS $(2^n - 1, 2^n, 2^n + 1, 2^{2n} + 1)$. *IEEE International Symposium on Circuits and Systems (ISCAS)*, 1540–1543.
20. Hiasat, A. (2017). A reverse converter and sign detectors for an extended RNS five moduli set. *IEEE Transactions on Circuits and Systems TCAS-I*, *64*(1), 111–121.
21. Hiasat, A. (2018). A sign detector for the extended four moduli set $\{2^n - 1, 2^n, +1, 2^{2n} + 1, 2^{n+k}\}$. *IET Proceedings - Computers and Digital Techniques*, *12*(2), 39–44.
22. Kumar, S., Chang, C.-H., Tay, T.F. (2017). New algorithm for signed integer comparison in $\{2^{n+k}, 2^n - 1, 2^n + 1, 2^{n\pm1} - 1\}$ and its efficient hardware implementation. *IEEE Transactions on Circuits and Systems TCAS-I*, *64*(6), 1481–1493.
23. Zimmerman, R. (1999). Efficient VLSI implementation of modulo $(2^n 1)$ addition and multiplication. In *Proceedings of the 14th IEEE symposium on computer arithmetic* (pp. 158–167).
24. Piestrak, S., & Berezowski, K. (2008). Design of residue multipliers-accumulators using periodicity. In *Proceedings of the IET Irish signals and systems conference (ISSC)* (pp. 380–385).
25. Sheu, M.-H., Siao, S.-M., Hwang, Y.-T., Sun, C.-C., Lin, Y.-P. (2016). New adaptable three-moduli $\{2^{n+k}, 2^n, -1, 2^{n-1} - 1\}$ residue number system-based finite impulse response implementation. *IEICE Electronics Express*, *13*, 1–9.

Ahmad Hiasat received the BSc and MSc degrees in electrical engineering from the University of Jordan, Amman, Jordan. He received the PhD degree in systems engineering from Oakland University, MI, USA in 1995. He then joined Princess Sumaya University for Technology (PSUT). Seconded from PSUT, he served as the Chairman and CEO of Telecommunications Regulatory Commission of Jordan during 2006-2010 and the Chairman and CEO of Energy Regulatory Commission of Jordan during 2011-2012. He has organized and/or participated in several international conferences on ICT, energy and education, and chaired a few of them. His research interests include; computer arithmetic, residue number system, digital median filters, and VLSI design.

Leonel Sousa received a Ph.D. degree in Electrical and Computer Engineering from the Instituto Superior Tecnico (IST), Universidade de Lisboa (UL), Lisbon, Portugal, in 1996, where he is currently Full Professor. He is also a Senior Researcher with the R&D Instituto de Engenharia de Sistemas e Computadores (INESC-ID). His research interests include VLSI architectures, computer architectures, parallel computing, computer arithmetic, and signal processing systems. He has contributed to more than 200 papers in journals and international conferences, for which he got several awards - such as, DASIP'13 Best Paper Award, SAMOS'11 'Stamatis Vassiliadis' Best Paper Award, DASIP'10 Best Poster Award, and several Honorable Mention Awards from Universidade Técnica de Lisboa/Santander Totta (2007, 2009) and Universidade de Lisboa/Santander (2016) for the quality and impact of his scientific publications. He has contributed to the organization of several international conferences, namely as program chair and as general and topic chair, and has given keynotes in some of them. He has edited four special issues of international journals, and he is currently Associate Editor of the IEEE Transactions on Multimedia, IEEE Transactions on Circuits and Systems for Video Technology, IEEE Access, IET Electronics Letters, Springer JRTIP, and Editor-in-Chief of the Eurasip JES. He is Fellow of the IET, a Distinguished Scientist of ACM and a Senior Member of IEEE.