INTERNATIONAL JOURNAL OF
COASTAL, OFFSHORE & ENVIRONMENTAL ENGINEERING IJCOE Vol.7/No. 2/Spring 2022 (31-38)

Available online at: www. Ijcoe.org

Fast Reverse converter Design for three moduli set $\{2^n.2^n-1.2^{n-1}-1\}$ Using CRTF

Javad Ahsan[1]. Mohammad Esmaeildoust[2*]. Amer Kaabi[3]. Vahid Zarei[4]

[1] *Faculty of Marine Engineering. Khorramshahr University of Marine Science and Technology. Khorramshahr. Iran; javad.ahsan@kmsu.ac.ir*
[2*] *Faculty of Marine Engineering. Khorramshahr University of Marine Science and Technology. Khorramshahr. Iran; m_doust@kmsu.ac.ir*
[3] *Department of Basic Sciences. Abadan Faculty of Petroleum Engineering. Petroleum University of Technology. Abadan. Iran; Kaabi_amer@put.ac.ir*
[4] *Faculty of Marine Engineering. Khorramshahr University of Marine Science and Technology. Khorramshahr. Iran; v.zarei@kmsu.ac.ir*

ARTICLE INFO

Article History:
Received: 13 Oct. 2021
Accepted: 12 Apr. 2022

Keywords:
Marine secure communication
Residue number system
CRTF
Reverse converter

ABSTRACT

Security is necessary for marine communication systems such as marine wireless sensor networks and automatic identification system which is the emerging system for automatic traffic control and collision avoidance services in the maritime transportation sector. Public key cryptography algorithms have an important role in these systems to realize secure communication systems. Public key cryptography algorithms such as RSA and Elliptic curve cryptography (ECC) have high computation costs and many works are done by researcher in order to speed up the operation. Residue number system which is a carry free system is widely used to speed up the operation in public key cryptography algorithm. In this paper. an improved RNS reverse converter for three-module set $\{2^n.2^n-1.2^{n-1}-1\}$ using chinese reminder theorem with fractional is presented. Unit gate delay and area comparison of the proposed reverse converter with literature have confirmed that the proposed reverse conversion takes fewer hardware costs and higher speed.

1. Introduction

Maritime communication systems such as Radio frequency port communication system [1]. Automatic Identification System (AIS) which is the emerging system for automatic traffic control and collision avoidance services in the maritime transportation sector [2] and marine wireless sensor networks [3] have security vulnerability and many works are done by researcher in order to provide security. Public key cryptography algorithms such as RSA [4] and ECC [5]. [6] are widely used in these systems in order to provide security. Digital signature algorithms [7] which are employed for authentication and integrity in these application [3] are also realized using public key cryptography algorithms. Public key cryptography algorithms have high computation costs and many works are done by researcher in order to speed up the operation [8]. [9]. [10]. Residue number system (RNS) [11] is a carry free number system which provides operation on parallel channel. This property led to fast arithmetic operation. Residue number system is widely used in public key cryptography algorithms in order to realize fast implementation [12]. [13].

RNS has three main parts includes forward conversion. arithmetic unit and reverse conversion. In the RNS system. the weighted numbers are converted into corresponding residues by a forward converter. The arithmetic unit consists of a modulus adder. subtractor. and a modulus multiplier for each modulus of the moduli set. This part performed on RNS numbers in parallel without carry propagation between residue channels. In order to use the result of arithmetic unit of RNS. the residues should be converted into its equivalent weighted binary number

by using a reverse converter [11]. Efficiency of the RNS system is related to form and the number of moduli. The dynamic range. the speed. and the hardware implementation of RNS systems are directly influenced by the form and the number of chosen moduli [14].

The most challenging parts of designing a RNS system is the design of reverse converter. Design of the reverse converter is critical step in terms of computational speed. Algorithms such as Chinese Remainder Theorem (CRT) and mixed-radix conversion (MRC). are used to design a reverse converter. Many works are done by researcher on different moduli set and efficient reverse converter are designed. The RNS bases has been selected with the aim of achieving optimal reverse converter. high dynamic range and simple computational architecture for implementation with the minimum hardware resources. time delay. and low power consumption. In [15]. three moduli set $\{2^n.\ 2^n\text{-}1.\ 2^n\text{+}1\}$ is presented. This moduli set is well-formed and led to simple hardware implementation for RNS design. However arithmetic operation in moduli 2^n+1 is more complex. to accelerate the speed of the RNS arithmetic unit. in [16] modulus 2^{n+1}-1 is used instead. and the RNS is designed with the moduli set $\{2^n.\ 2^n\text{-}1.\ 2^{n+1}\text{-}1\}$. In [17] the moduli set $\{2^{n+k}.\ 2^n\text{-}1.\ 2^{n+1}\text{-}1\}$is employed to design an efficient RNS arithmetic unit.

In [18]. efficient reverse converter is designed for moduli set $\{2^n.\ 2^n\text{-}1.\ 2^{n-1}\text{-}1\}$ using Chinese Reminder Theorem with fractional (CRTF). In order to have fast and efficient implementation of the converter. in this paper an improvement over converter reported in [18] is presented. The improved version has spare one 4*n*-bit Kogge-stone's prefix adder [19] from the critical path. Therefore. faster reverse converter design is resulted.

This paper is organized as follows: Section 2 presents the basic RNS mathematics. The reverse converter for moduli set $\{2^n.\ 2^n\text{-}1.\ 2^{n-1}\text{-}1\}$ proposed in [18] is detailed in section 3. Section 4 presents the proposed reverse converter. The implementation and performance evaluation results are presented in Section 5. Finally. Section 6 concludes the paper.

2. Related Background

2.1. RNS Background

The RNS is defined in terms of relatively prime moduli set $\{m_1.m_2.\ldots.m_n\}$ that is $\gcd(m_i.m_j) = 1$ for $\neq j$. A weighted number *X* can be represented as $X = (x_1.\ x_2.\ldots.x_n)$. Where

$$X = X\ mod\ m_i = |X|_{m_i}.\quad 0 \le m_i < x_i. \tag{1}$$

Such a representation is unique for any integer X in the range $[0.M-1]$. where M is the dynamic range of the moduli set $\{m_1.m_2.\ldots.m_n\}$. which is equal to the product of m_i terms $(M = m_1.m_2.\ldots.m_n)$ [11].

2.2. Chinese Remainder Theorem

The Chinese Remainder Theorem (CRT) may rightly be viewed as one of the most important fundamental results in the theory of residue number systems. Computing weighted number X form its RNS representation. i.e.. $(x_1.\ x_2.\ldots.x_n)$. based on the moduli set $\{m_1.m_2.\ldots.m_n\}$ is as follows:

$$X = \left|\sum_{i=1}^{n}\left|M_i^{-1}\right|_{m_i} M_i x_i\right|_M . \tag{2}$$

where $M_i = M/m_i$. $\left|M_i^{-1}\right|_{m_i}$is the multiplicative inverse of $M_i\ \ i\ =\ 1.2.\ldots.n$ [20].

2.3. Approximate CRT

The modification of the CRT using fractional values. namely approximate CRT. was introduced for the first time in [21] to perform sign detection and division in RNS. Considering a number X in RNS with the moduli set $\{m_1.m_2.\ldots.m_n\}$. if it is divided by M. the following formula can be obtained:

$$\tilde{X} = \frac{X}{M} = \left|\sum_{i=1}^{n}\frac{\left|M_i^{-1}\right|_{m_i}}{m_i}x_i\right|_1 = \left|\sum_{i=1}^{n}k_i x_i\right|_1 . \tag{3}$$

Where.

$$k_i = \frac{\left|M_i^{-1}\right|_{m_i}}{m_i}.\quad i = 1.2.\ldots.n. \tag{4}$$

The value $\tilde{X}$ can be considered as a positional characteristic of *X*. and the number *X* can be obtained by

$$X = \tilde{X}M. \tag{5}$$

Block diagram of CRTF for general reverse conversion is shown in figure 1.

2.4. Chinese Remainder Theorem with Fractions

In [22] presents a modified version of the approximate CRT. named CRTF to perform residue-to-binary conversion. First. number of the fractional bits required for an accurate residue-to-binary conversion as follows:

$$N = \lceil \log_2 M\mu - 1 \rceil. \tag{6}$$

Where

$$M = \prod_{i=1}^{n} m_i \,.\, \mu = \sum_{i=1}^{n} (m_i - 1). \tag{7}$$

Operations on real numbers in computing devices results in high hardware and delay. Hence. it is necessary to replace fraction computations with integer ones. This can be achieved by making some modifications in the algorithm. To have integer calculations. each number k_i should be multiplied by 2^N. The integer part of each of the resulting numbers plays the role of fractional part in the original method. while bits exceeding order N. should be discarded and not considered in calculations. In other words. calculations must be performed in modulus 2^N.

$$k_i^* = \left\lceil \frac{\left|M_i^{-1}\right|_{m_i}}{m_i} . 2^N \right\rceil .\ i = 1.2. \ldots . n. \tag{8}$$

The value of X^* obtained as follows:

$$X^* = \left| \sum_{i=1}^{n} k_i^* x_i \right|_{2^N} . \tag{9}$$

The X^*can be used to perform residue-to-binary conversion of X by the following relation where $\lfloor A \rfloor$ means the floor of A.

$$X = \left\lfloor \frac{X^* M}{2^N} \right\rfloor . \tag{10}$$

Block diagram of the general reverse converter suing CRTF is shown in Figure 1.

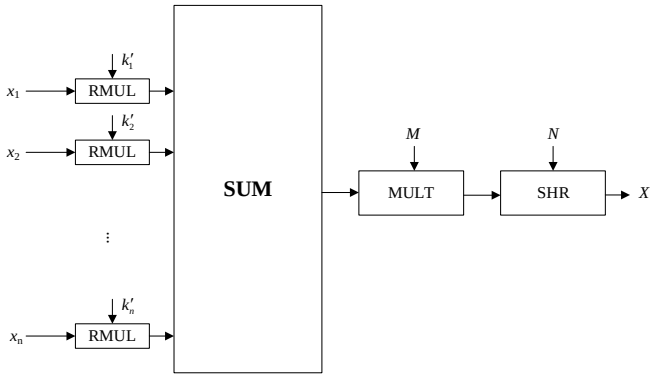

Figure 1. Block diagram of general reverse converter using CRTF [21]

3. Reverse converter for moduli set $\{2^n . 2^n - 1 . 2^{n-1} - 1\}$ proposed in [18]

In [18]. efficient reverse converter has been proposed for the moduli set $\{2^n . 2^n - 1 . 2^{n-1} - 1\}$ using CRTF. In the following the reverse converter reported in [18] is detailed.

Consider three moduli $\{m_1. m_2. m_3\} = \{2^n . 2^n - 1 . 2^{n-1} - 1\}$ with dynamic range M = $2^n(2^n\text{-}1)(2^{n\text{-}1}\text{-}1)$.

The values of the multiplicative inverse M_i^{-1} are:

$$M_1^{-1} \equiv 1 + 2^{n-1} (mod\ 2^n) \tag{11}$$

$$M_2^{-1} \equiv 2^n - 3 (mod\ 2^n - 1) \tag{12}$$

$$M_3^{-1} \equiv 2^{n-2} (mod\ 2^{n-1} - 1) \tag{13}$$

Consider the cases where $N = 4n$. i.e.. where $n > 4$. The coefficients $k_1 . k_2 . k_3$ in Eq. 4 are presented in [18] as follows:

$$k_1 = \frac{\left|M_1^{-1}\right|_{m_1}}{m_1} = \frac{2^{n-1} + 1}{2^n} = \frac{1}{2} + \frac{1}{2^n} \tag{14}$$

$$k_2 = \frac{\left|M_2^{-1}\right|_{m_2}}{m_2} = \frac{2^n - 3}{2^n - 1} \tag{15}$$

$$k_3 = \frac{\left|M_3^{-1}\right|_{m_3}}{m_3} = \frac{2^{n-2}}{2^{n-1} - 1} \tag{16}$$

The coefficients $k_1^* . k_2^* . k_3^*$ in Eq. 8 are calculated as follows:

$$k_1^* = \left\lceil \frac{\left|M_1^{-1}\right|_{m_1}}{m_1} . 2^N \right\rceil = \lceil 2^N . k_1 \rceil \tag{17}$$

$$k_1^* = \lceil 2^N . k_1 \rceil = \left\lceil 2^N . \left(\frac{1}{2} + \frac{1}{2^n}\right)\right\rceil = \left\lceil 2^{4n} . \left(\frac{1}{2} + \frac{1}{2^n}\right)\right\rceil =$$

$$= \lceil 2^{4n} . (2^{-1} + 2^{-n}) \rceil = \lceil 2^{4n-1} + 2^{3n} \rceil = 2^{4n-1} + 2^{3n}$$

The numerical value of k_1^* is:

$$k_1^* = \underbrace{010 \dots 001}_{n-bits} \underbrace{00 \dots 000}_{n-bits} \underbrace{00 \dots 000}_{n-bits} \underbrace{00 \dots 000}_{n-bits} \quad (18)$$

For simplicity the zero bits in the most significant bits can be removed:

$$k_1^* = \underbrace{10 \dots 001}_{(n-1)-bits} \underbrace{00 \dots 000}_{n-bits} \underbrace{00 \dots 000}_{n-bits} \underbrace{00 \dots 000}_{n-bits} \quad (19)$$

The number k_1^* has a $(4n-1)$-bits width.
Find the value k_2^*:

$$k_2^* = \lceil 2^N . k_2 \rceil = \left\lceil 2^{4n} . \left(\frac{2^n - 3}{2^n - 1}\right)\right\rceil \quad (20)$$

Since the number k_2^* is taken rounded up. therefore k_2^* can be rewritten as:

$$k_2^* = \underbrace{11 \dots 101}_{n-bits} \underbrace{11 \dots 101}_{n-bits} \underbrace{11 \dots 101}_{n-bits} \underbrace{11 \dots 101}_{n-bits} \quad (21)$$

Since this number contains more ones than zeros. in [18] the technique which introduced in [23] is employed in order to simplified the required operation:

$$|(-X).(-C)|_{2^\alpha} = \left|\bar{X}.\left(\bar{C} + 1 - 2^f\right) + \Delta_{COR}\right|_{2^\alpha} \quad (22)$$

The correction factor is calculated by the formula:

$$\Delta_{COR} = \left|(1 - 2^g).\left(\bar{C} + 1 - 2^f\right)\right|_{2^\alpha} \quad (23)$$

where g is the bit width of the number X. f is the bit width k_i^* . and α is the bit width N. Let us find the correction factor by the formula (23):

$$\Delta_{COR} = \left|(1 - 2^n).\left(\overline{k_2^*} + 1 - 2^{4n}\right)\right|_{2^{4n}} = |2|_{2^{4n}} \quad (24)$$

After inversion the value k_2^*. we have:

$$\overline{k_2^*} = \underbrace{00 \dots 010}_{n-bits} \underbrace{00 \dots 010}_{n-bits} \underbrace{00 \dots 010}_{n-bits} \underbrace{00 \dots 010}_{n-bits} \quad (25)$$

Remove the zero bits in the most significant bits results:

$$\overline{k_2^*} = \underbrace{10}_{2bits} \underbrace{00 \dots 010}_{n-bits} \underbrace{00 \dots 010}_{n-bits} \underbrace{00 \dots 010}_{n-bits} \quad (26)$$

We get that the number $\overline{k_2^*}$ has a bit width $(3n+2)$-bits and $\Delta_{COR} = 2_{10} = 10_2$ is 2 bits.
Find the value k_3^*:

$$k_3^* = \lceil 2^N . k_3 \rceil = \left\lceil 2^{4n} . \left(\frac{2^{n-2}}{2^{n-1} - 1}\right)\right\rceil \quad (27)$$

Since the number k_3^* s taken rounded up. We get:

$$k_3^* = \underbrace{10 \dots 000}_{(n-1)-bits} \underbrace{10 \dots 000}_{(n-1)-bits} \underbrace{10 \dots 000}_{(n-1)-bits} \underbrace{10 \dots 000}_{(n-1)-bits} \underbrace{1001}_{4bits} \quad (28)$$

We get that the number k_3^* has a bit width 4*n*-bit.

4. Improved Reverse converter for moduli set $\{2^n . 2^n - 1 . 2^{n-1} - 1\}$

In order to realize the fast implementation of the reverse converter. some part of the reverse converter for the moduli set $\{2^n . 2^n - 1 . 2^{n-1} - 1\}$ proposed in [18] are redesigned which is detailed as follows.
The first step will be the calculation of value X^*. which obtained by taking modulo N (4*n*-bit) as

$$X^* = \left|\sum_{i=1}^{n} x_i k_i^*\right|_{2^N} \quad (29)$$

After applying the compression technique reported in [18]. four terms are resulted which is shown in figure 2. The last term x_{30}occupies 1 bit - the most significant bit of the number. Let's discard it for further summation. Let the first three terms be equal to $v_1 . v_2$ and v_3. In [18]. in order to realize the hardware implementation of summation of partial product shown in figure 2. one 4n-bit CSA and XOR followed by one KSA adder are employed. In order to achieve faster implementation. in the improved version. KSA adder is eliminated and two 4n-bit variable named X_1^* and X_2^* will send to next level of the design. It should be noted that one KSA adder are spare from the critical path. Therefore. in the next step. the value of $(X_1^* + X_2^*) \times M$ should be calculated. Since M the dynamic range is equal to $2^n(2^n - 1)(2^{n-1} - 1) = 2^{3n-1} - 2^{2n} - 2^{2n-1} + 2^n$. we have

$$X_1^* \times M = X_1^* \times \left(2^{3n-1} - 2^{2n} - 2^{2n-1} + 2^n\right) \quad (30)$$

$$= X_1^* 2^{3n-1} - X_1^* 2^{2n} - X_1^* 2^{2n-1} + X_1^* 2^n \qquad = X_{11}^* + X_{12}^* + X_{13}^* + X_{14}^*$$

	4n-1	4n-2	...	3n+1	3n	3n-1	...	n+3	n+2	n+1	n	n-1	n-2	...	5	4	3	2	1	0
$V_1 =$	$x_{1(n-1)}$	$x_{1(n-2)}$	...	x_{11}	x_{10}	$\overline{x_{2(n-2)}}$	...	$\overline{x_{22}}$	$\overline{x_{21}}$	$\overline{x_{20}}$	$\overline{x_{2(n-1)}}$	$\overline{x_{2(n-2)}}$	$\overline{x_{2(n-3)}}$	...	$\overline{x_{24}}$	$\overline{x_{23}}$	$\overline{x_{22}}$	$\overline{x_{21}}$	$\overline{x_{20}}$	x_{30}
$V_2 =$	x_{10}	$\overline{x_{2(n-3)}}$	...	$\overline{x_{20}}$	$\overline{x_{2(n-1)}}$	$x_{3(n-2)}$	...	x_{31}	x_{30}	$x_{3(n-2)}$	$x_{3(n-3)}$	$x_{3(n-4)}$	$x_{3(n-2)}$	...	x_{35}	x_{34}	x_{33}	x_{32}	1	
$V_3 =$	$\overline{x_{2(n-2)}}$	$x_{3(n-2)}$	...	x_{31}	x_{30}								$x_{3(n-5)}$	...	x_{32}	x_{31}	x_{30}		x_{31}	
	x_{30}																			

Figure 2. Partial product for calculation of $X^* = |\sum_{i=1}^{n} x_i k_i^*|_{2^N}$ [18]

Where

$$X_{11}^* = \overline{X_1^*}\,\underbrace{11\ldots1}_{2n}$$
$$X_{12}^* = \overline{X_1^*}\,\underbrace{11\ldots1}_{2n-1}$$
$$X_{13}^* = X_1^*\,\underbrace{00\ldots0}_{n-2}\,10$$
$$X_{14}^* = X_1^*\,\underbrace{00\ldots0}_{3n-1}$$

$$X_2^* \times M = X_2^* \times \left(2^{3n-1} - 2^{2n} - 2^{2n-1} + 2^n\right)$$

$$= X_2^* 2^{3n-1} - X_2^* 2^{2n} - X_2^* 2^{2n-1} + X_2^* 2^n \qquad (31)$$

$$= X_{21}^* + X_{22}^* + X_{23}^* + X_{24}^*$$

Where

$$X_{21}^* = \overline{X_2^*}\,\underbrace{11\ldots1}_{2n}$$
$$X_{22}^* = \overline{X_2^*}\,\underbrace{11\ldots1}_{2n-1}$$
$$X_{23}^* = X_2^*\,\underbrace{00\ldots0}_{n-2}\,10$$
$$X_{24}^* = X_2^*\,\underbrace{00\ldots0}_{3n-1}$$

Hardware implementation of the improved reverse converter is shown in Figure 3. In order to realize Eq. (30) and (31). four sage CSA followed by one KSA adder is employed. Hardware details of the proposed reverse converter included in Table 1.

Table 1. Detailed of each component for the first level design of the reverse converter

Component	FA	XOR	AND	OR
CSA (1)	4*n*	-	-	-
CSA (2)	6*n*	-	-	-
CSA (3)	6*n*-2	-	-	-
CSA (4)	6*n*	-	-	-
CSA (5)	7*n*-1	-	-	-
CSA (6)	7*n*-1	-	-	-
CSA (7)	7*n*-1	-	-	-
KSA	-	14*n*-2	21*n*-3	7*n*-1

5. Performance Evaluation

In this section. the performance of the proposed reverse converter has been compared with other converters for the moduli set $\{2^n.2^n-1.2^{n-1}-1\}$ introduced in [18]. The conversion delay and hardware cost of the proposed converters and converter [18] are illustrated in Table 2.

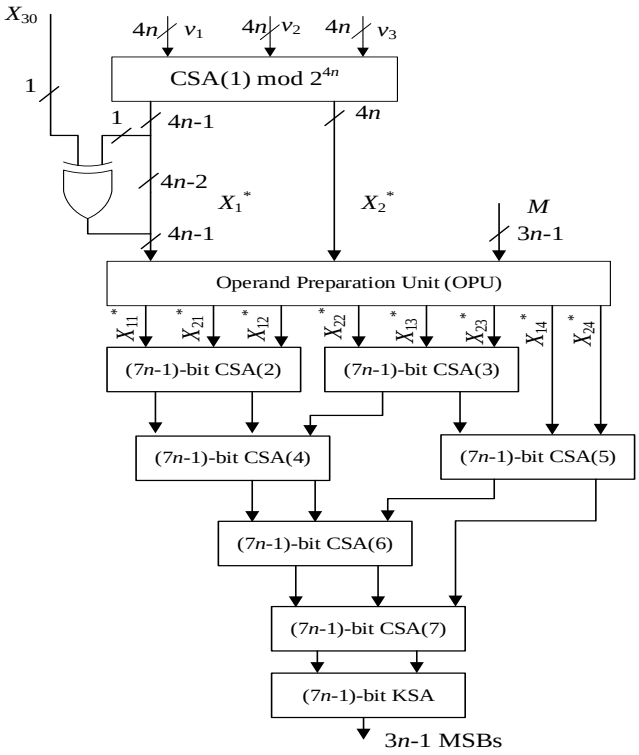

Figure 3. Hardware implementation of the proposed reverse converter

Table 2. Hardware costs and conversion delay comparison

Converter	Hardware requirement	Delay

P. Lyakhov [18]	$(3.5n^2+3n)A_{FA}$ $+(22n-4)A_{XOR}$ $+(2\alpha+\beta+n)A_{AND}$ $+(\alpha+\beta)A_{OR}$	$(n/2+1)D_{FA}$ $+2D_{KSA}$
Proposed	$(43n-5)A_{FA}$ $+(14n-2)A_{XOR}$ $+(2\gamma+n)A_{AND}$ $+(\gamma)A_{OR}$	$5D_{FA}+1D_{KSA}$

$\alpha = (7n-2)\log(7n-2)-7n+3.$
$\beta = 4nlog(4n)-4n+1.$
$\gamma = (7n-1)\log(7n-1)-7n+2.$

The Kogge-Stone structure delay is given by $[\log_2 n]$ and the computational nodes is given by $[n\log_2 n - n + 1]$ where *n* is the number of input bits [24].
To have a better comparison and deriving area and delay estimations. the unit gate model [25] is used. According to this model. each FA. half adder (HA). 2×1 MUX. XOR. XNOR. AND. OR gates considered as 7. 3. 3. 2. 2. 1. 1 gates in area and 4. 2. 2. 2. 2. 1. 1 gates in delay. respectively. Table 3 shows the unit gate delay and area comparison. According to the calculated phrase for the amount of hardware required in the converter reported in [18]. by growth of *n* the amount of hardware required will increase with a large slope compared to the proposed converter. To provide more accurate analysis. for different values of *n* the unit gate delay. area and AT are calculated and illustrated in Figure 4. It can be seen that compared to the converter reported in [18]. the proposed converters have achieved a better delay and needs less hardware. Low hardware requirements of the proposed converters. led to efficient and simple implementation on real hardware with the growth of *n*. the proposed converter has achieved 87.5% less delay and 37.6% less hardware requirements compared to converter reported in [18] for 16-bits design.

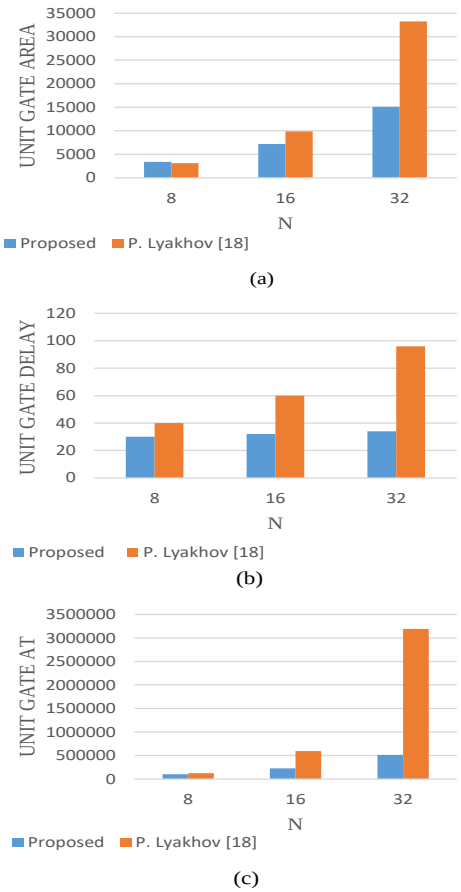

Figure 4. Unit Gate Comparison: (a) Unit gate Area. (b) Unit Gate Delay and (c) Unit Gate AT

Table 3. Unit gate delay and area evaluation

Converter	Unit Gate Area	Unit Gate Delay
P. Lyakhov [18]	$24.5n^2+66n-8$ $+3\alpha+2\beta$	$(2n+4)$ $+4(\log_2 n)+8$
Proposed	$330n-39+3\gamma$	$20+2(\log_2 n)+4$

$\alpha = (7n-2)\log(7n-2)-7n+3.$
$\beta = 4nlog(4n)-4n+1.$
$\gamma = (7n-1)\log(7n-1)-7n+2.$

6. Conclusions

Public key cryptography algorithms have an important role in marine communication systems which secure communication is needed such as marine wireless sensor networks and Automatic identification systems. Carry free property of residue number system makes it suitable to be applied in public key cryptography algorithm in order to achieve higher speed. In this paper. an improved reverse converter for three-module set $\{2^n.2^{n-1}.2^{n-1}-1\}$ using CRTF is presented. The proposed converter has achieved less hardware requirement as well as less delay. Unit gate delay and area comparison of the proposed reverse converter with literature have confirmed that the proposed reverse converter has achieved noticeable

improvements in hardware costs. conversion delay and AT metric.

8. Reference

[1] Y. J. Kim and K. Kyung. "Secured radio communication based on fusion of cryptography algorithms." in *2015 IEEE International Conference on Consumer Electronics (ICCE)*. 2015. pp. 388–389. doi: 10.1109/ICCE.2015.7066457.

[2] A. Goudossis and S. K. Katsikas. "Towards a secure automatic identification system (AIS)." *J. Mar. Sci. Technol.*. vol. 24. no. 2. pp. 410–423. 2019. doi: 10.1007/s00773-018-0561-3.

[3] L. Wei. L. Zhang. D. Huang. and K. Zhang. "Efficient and provably secure identity-based multi-signature schemes for data aggregation in marine wireless sensor networks." in *2017 IEEE 14th International Conference on Networking. Sensing and Control (ICNSC)*. 2017. pp. 593–598. doi: 10.1109/ICNSC.2017.8000158.

[4] R. L. Rivest. A. Shamir. and L. M. Adleman. "A method for obtaining digital signatures and public key cryptosystems." *Secur. Commun. Asymmetric Cryptosystems*. vol. 21. no. 2. pp. 217–239. 2019.

[5] B. N. Koblitz. "Elliptic Curve Cryptosystems." vol. 4. no. 177. pp. 203–209. 1987.

[6] V. S. Miller. "Use of Elliptic Curves in Cryptography." *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*. vol. 218 LNCS. pp. 417–426. 1986. doi: 10.1007/3-540-39799-X_31.

[7] S. Goldwasser. S. Micali. and R. L. Rivest. "Digital signature scheme secure against adaptive chosen-message attacks." *S[1] S. Goldwasser. S. Micali. R. L. Rivest. "Digital Signat. scheme Secur. against Adapt. chosen-message attacks." SIAM J. Comput. vol. 17. no. 2. pp. 281–308. 1988. doi 10.1137/0217017.IAM J. Comput.*. vol. 17. no. 2. pp. 281–308. 1988. doi: 10.1137/0217017.

[8] M. Esmaeildoust. D. Schinianakis. H. Javashi. T. Stouraitis. and K. Navi. "Efficient RNS implementation of elliptic curve point multiplication over GF(p)." *IEEE Trans. Very Large Scale Integr. Syst.*. vol. 21. no. 8. pp. 1545–1549. 2013. doi: 10.1109/TVLSI.2012.2210916.

[9] D. M. Schinianakis. A. P. Fournaris. H. E. Michail. A. P. Kakarountas. and T. Stouraitis. "An RNS implementation of an Fp elliptic curve point multiplier." *IEEE Trans. Circuits Syst. I Regul. Pap.*. vol. 56. no. 6. pp. 1202–1213. 2009. doi: 10.1109/TCSI.2008.2008507.

[10] L. Imbert. J. Bajard. L. Imbert. J. B. A. Full. and R. N. S. Implementation. "A Full RNS Implementation of RSA To cite this version : HAL Id : lirmm-00090366 A Full RNS Implementation of RSA." 2006.

[11] K. Navi. A. S. Molahosseini. and M. Esmaeildoust. "How to teach residue number system to computer scientists and engineers." *IEEE Trans. Educ.*. vol. 54. no. 1. pp. 156–163. 2011. doi: 10.1109/TE.2010.2048329.

[12] S. Asif. M. S. Hossain. Y. Kong. and W. Abdul. "A Fully RNS based ECC Processor." *Integration*. vol. 61. pp. 138–149. 2018. doi: https://doi.org/10.1016/j.vlsi.2017.11.010.

[13] S. Asif and Y. Kong. "Highly Parallel Modular Multiplier for Elliptic Curve Cryptography in Residue Number System." *Circuits. Syst. Signal Process.*. vol. 36. no. 3. pp. 1027–1051. 2017. doi: 10.1007/s00034-016-0336-1.

[14] W. Wang. M. N. S. Swamy. and M. O. Ahmad. "Moduli selection in RNS for efficient VLSI implementation." in *Proceedings of the 2003 International Symposium on Circuits and Systems. 2003. ISCAS '03.*. 2003. vol. 4. pp. IV–IV. doi: 10.1109/ISCAS.2003.1205945.

[15] Y. Wang. X. Song. M. Aboulhamid. and H. Shen. "Adder based residue to binary number converters for (2/sup n/-1. 2/sup n/. 2/sup n/+1)." *IEEE Trans. Signal Process.*. vol. 50. no. 7. pp. 1772–1779. 2002. doi: 10.1109/TSP.2002.1011216.

[16] A. Hiasat. "An Efficient Reverse Converter for the Three-Moduli Set ($2^{n+ 1}-1. 2^{n}. 2^{n}-1$)." *IEEE Trans. Circuits Syst. II Express Briefs*. vol. 64. no. 8. pp. 962–966. 2016.

[17] A. Hiasat and L. Sousa. "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}." *Comput. J.*. vol. 62. no. 2. pp. 292–300. 2019.

[18] P. Lyakhov. M. Bergerman. N. Semyonova. D. Kaplun. and A. Voznesensky. "Design Reverse Converter for Balanced RNS with Three Low-cost Modules." no. 3. pp. 1–7. 2021. doi: 10.1109/meco52532.2021.9460200.

[19] P. M. Kogge and H. S. Stone. "A parallel algorithm for the efficient solution of a general class of recurrence equations." *IEEE Trans. Comput.*. vol. 100. no. 8. pp. 786–793. 1973.

[20] A. Omondi and B. Premkumar. *Residue Number Residue Number*. 1951.

[21] C. Y. Hung and B. Parhami. "An approximate sign detection method for residue numbers and its application to RNS division." *Comput. Math. with Appl.*. vol. 27. no. 4. pp. 23–35. 1994. doi: 10.1016/0898-1221(94)90052-3.

[22] N. I. Chervyakov. A. S. Molahosseini. P. A. Lyakhov. M. G. Babenko. and M. A. Deryabin. "Residue-to-binary conversion for general moduli sets based on approximate Chinese remainder theorem." *Int. J. Comput. Math.*. vol. 94. no. 9. pp. 1833–1849. 2017. doi: 10.1080/00207160.2016.1247439.

[23] H. Nakahara and T. Sasao. "A High-speed Low-power Deep Neural Network on an FPGA based on the Nested RNS: Applied to an Object Detector."

in *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*. 2018. pp. 1–5.

[24] S. G. R and R. Kalaimathi. “Design and Analysis of Kogge-Stone and Han-Carlson Adders in 130nm CMOS Technology.” no. March 2018. 2020.

[25] M. Obeidi Daghlavi. M. R. Noorimehr. and M. Esmaeildoust. “Efficient two-level reverse converters for the four-moduli set {2n−1. 2n–1. 2n−1–1. 2n+1–1}.” *Analog Integr. Circuits Signal Process.*. vol. 0123456789. 2020. doi: 10.1007/s10470-020-01749-z.