Mixed Radix Conversion based RSA Encryption System

Salifu Abdul-Mumin
Department of Computer Science
University for Development Studies,
Navrongo, Ghana

Kazeem A. Gbolagade
Department of Computer Science
College of Information and Communication Tech
Kwara State University
Malete, Nigeria

ABSTRACT
Information security is a critical issue in data communication networks. This is more important in wireless communications due to the fact that the transmitted signal could go beyond the communicating participants. Any person with the right equipment could intercept the transmitted information with ease. It is therefore paramount to encrypt information before transmission to prevent intruders from making meaning to intercepted signals.

In this paper, an improved Rivest Shamir Adleman (RSA) cryptosystem based on Residue Number System (RNS) is implemented. There are two stages of encryption. The first stage is the traditional RSA and the second stage is to further encrypt the cypher text obtained from RSA using smaller moduli. The first stage of the decryption process is to obtain a partial result through Mixed Radix Conversion (MRC). The final stage of decryption is the RSA decryption process. This is to allow a message m, for which $m^e < n$ to be able to be encrypted. The private key length is also enhanced by adding the moduli set to the RSA private key component. It is observed that the proposed system outperforms the existing algorithm in terms of security.

General Terms

Security, Encryption, RSA, RNS

Keywords
Information Security, Encryption, RSA, RNS, MRC, Forward Conversion, Backward Conversion

1. INTRODUCTION
Information security is one of the fundamental issues in ensuring data confidentiality and data integrity which are some of the key factors in improving the quality of service in data communication. There are so many ways of securing data in an unsecured communication channel; some of which include network intrusion detection system and by way of encryption. There are two types of encryption; Symmetric (Private-key encryption) which relies solely on the secrecy of the private key and Asymmetric (Public-key encryption) which has a public key for encryption and a private key for decryption. An example of such a public-key encryption is the RSA public-key cryptosystem.

Rivest Shamir Adleman (RSA) public-key cryptosystem was invented at MIT in 1977 by Ronald Rivest, Adi Shamir and Leonard Adleman. The public key in this cryptosystem consists of the value n, which is called the modulus, and the value e, which is called the public exponent. The private key consists of the modulus n and the value d, which is called the private exponent [1].

RSA is the most widely used public-key cryptosystem [2].

Most public-key cryptosystems currently in use, such as RSA, rely on the intractability of factoring and computing discrete logarithms. However, in 1994, Shor proposed efficient quantum algorithms to solve these problems [3]. Hence, should quantum computing become viable, currently-in-use cryptosystems will be broken. As such, research on efficient post-quantum public-key cryptosystems is most valuable [4].

In some cases also, it is easy to compute modular roots without knowledge of the prime factors. For instance, if m is known to be very small, such that c = me < n, then m can be recovered from c by taking eth roots over the integers, which is easy [1].

In this work, Residue Number System (RNS) is used to enhance the RSA public-key cryptosystem by further passing through the cypher text obtained from RSA into very smaller moduli set such that smaller messages that could not have been encrypted by RSA would have the opportunity to be encrypted. The key length is also enhanced as the chosen moduli set are part of the private key. Also this cryptosystem will not entirely rely on the intractability of factoring and computing discrete logarithms. The proposed system outperforms the existing algorithm in terms of security

The rest of the paper is organized as follows; Section 2 gives background of RNS, Section 3 presents the proposed scheme where the second level of encryption (backward conversion), the first level of decryption (Reverse conversion) and the hardware realization are presented. Section 4 presents the performance analysis with the traditional RSA encryption and the paper is concluded in Section 5.

2. BACKGROUND OF RNS
RNS is defined in terms of a set of relatively prime moduli. If M denotes the moduli set, then

P = {m1, m2, ,,,,,,,, mk}, GCD (mi, mj) = 1,for i ≠ j. Any integer X in the range [0, M) where M= ∏ki=1mi can be uniquely and unambiguously represented by the residue sequence: X↔ (x1, x2 …xk) where X1 = Xmodmi, i = 1, 2, …….k is the residue modulus mi of X. The range [0, M) is called dynamic range or the legitimate range of X. [5]

Residue Number System (RNS) is an integer number system with the capabilities to support parallel, carry-free addition, borrow-free subtraction and single step multiplication without partial product. These features enable RNS utilization in Digital Signal Processing (DSP) applications such as digital filtering, convolution, fast Fourier transform and image processing

[6], [7], [8]

However, RNS has not found a wide spread usage in general purpose computing due to the following difficult RNS arithmetic operations: magnitude comparison, sign detection,

overflow detection, moduli selection, reverse and forward conversions.[9]

The conversion from a conventional number system to a residue number system is referred to as Forward conversion and the conversion from a residue number system to a conventional number system is known as Reverse/Backward conversion which is achieved by using the Chinese Remainder Theorem or Mixed Radix Conversion. There are other variations of the two that can be used to obtain reverse conversion.

3. PROPOSED CRYPTOSYSTEM

Our proposed cryptosystem has two stages of encryption; the first stage is the traditional RSA encryption and the second stage is the encryption using the proposed moduli set from RNS. Forward conversion in RNS is the second stage encryption and reverse conversion is the first stage decryption and finally, the traditional RSA decryption will serve as the second stage deciphering process. The proposed system has inherent features of symmetric key encryption system embedded in the asymmetric key encryption system. The same key used for the second stage encryption is the same key used for the first stage decryption process. However the problem with regards to sharing that secrete key between the sender and the receiver is avoided.

In this paper, we consider only the second stage encryption and the first stage decryption process.

3.1 Implementation of Stage-2 Encryption Scheme

Given the moduli set above, let $m_1 = 2^n - 1,\ m_2 = 2^n$, and $m_3 = 2^{n+1} - 1$ where $m_i, i = 1, 2, 3$ are the moduli representing the channel-order of the moduli set.

Now, in order to convert a number X from binary/decimal to its residues equivalent (forward conversion) using the moduli set; we compute the $r_{i,\{i=1,2,3\}}$, the residue set by performing $r_i = |X|_{m_i}$ which is the modulus operation on X with respect to each modulus. $M = \prod_{i=1}^{3} m_i = [2^{3n+1} - 2^{2n} + 2^{2n}]$ (1)

Which indicates that X is a $(3n+1)$-bit number and represented in binary as:

$$X = x_{3n}x_{3n-1}x_{3n-2}\ldots x_1x_0 \quad (2)$$

It follows therefore that the r_i's can be computed as follows;

- r_2 is the n least significant bit (LSB) of X in binary.
- For r_1 and r_3, we partition X into two n-bit blocks B_1, B_2, and one $(n+1)$-bit block B_3 where;

$$\left.\begin{aligned} B_3 &= \sum_{j=2n}^{3n} x_j 2^{j-2n+1} \\ B_2 &= \sum_{j=n}^{2n-1} x_j 2^{j-n} \\ B_1 &= \sum_{j=0}^{n-1} x_j 2^j \end{aligned}\right\} \quad (3),$$

This implies

$$X = B_1 + 2^n B_2 + 2^{2n} B_3 \quad (4)$$

Therefore,

$$r_1 = |X|_{2^n-1} = |B_1 + 2^n B_2 + 2^{2n} B_3|_{2^n-1}$$

$$= ||B_1|_{2^n-1} + |B_2 2^n|_{2^n-1} + |2^{2n} B_3|_{2^n-1}|_{2^n-1}$$

$$= |B_1 + B_2 + B_3|_{2^n-1}$$

Similarly,

$$r_3 = |X|_{2^{n+1}-1} = |B_1 + 2^n B_2 + 2^{2n} B_3|_{2^{n+1}-1}$$

$$= ||B_1|_{2^{n+1}-1} + |B_2 2^n|_{2^{n+1}-1} + |2^{2n} B_3|_{2^{n+1}-1}|_{2^{n+1}-1}$$

$$= |B_1 + 2^n B_2 + 2^{n-1} B_3|_{2^n-1}$$

Example1:

Given the moduli set $\{2^n - 1, 2^n, 2^{n+1} - 1\}$, take $n = 2$ and a number, $X = 50$. Then the conversion process is as follows;

$50 = 110010 = 0110010$ (7-bits, since X is a $(3n+1) - bit$ number)

Since $n = 2$, we partition X into two 2-bits blocks and one 1-bit block.

Thus, $B_1 = 10,\ B_2 = 00, and\ B_3 = 011$

Therefore;

$r_1 = |B_1 + B_2 + B_3|_{2^{n+1}-1}$

$\Rightarrow |50|_{2^2-1} = |50|_3 = |2 + 0 + 3|_5 = 2$, and

$r_3 = |B_1 + 2^n B_2 + 2^{n-1} B_3|_{2^{n+1}-1}$

$|50|_{2^3-1} = |50|_7 = |10 + 2^2(00) + 2^1(011)|_{111} = |2 + 0 + 6|_7 = 1.$

Since r_2 is the LSB of X in binary, then we have $B_1 = 10 = 2$.

This implies that $|50|_{3,4,7} = \{2, 2, 1\}$

The hardware realisation of the forward conversion is achieved by using simple fast adders like the carry save adder (CSA) for three bits addition and carry propagate adder (CPA) for two bits addition as shown in the figure below;

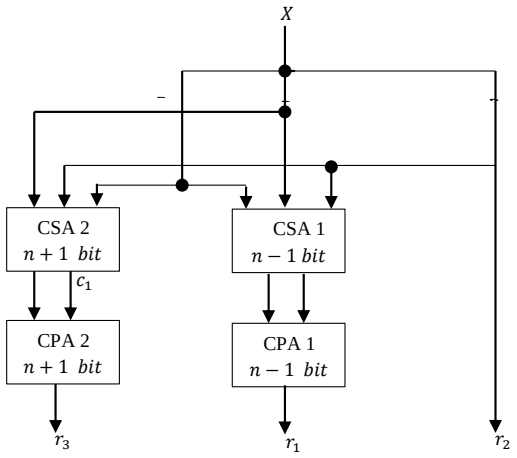

Fig 1: Hardware Architecture of stage-1 Encryption Scheme

The hardware requirements of this architecture and the delay imposed in computing the residues will be as follows:

$$Area(A) = A_{CSA1} + A_{CSA2} + A_{CPA1} + A_{CPA2} = 2(n+1)A + 2(n-1)A = 4nA \quad \text{And}$$

$$Delay(D) = D_{CSA2} + D_{CPA2} = D + 2(n+1)D = (2n+3)D$$

3.2 Implementation of Stage-1 Decryption Scheme

The Mixed Radix Conversion (MRC) is employed to convert any number X in RNS representation to its binary/decimal equivalent.

The general form of the MRC is given as

Follows;

$$X = d_1 + d_2 m_1 + d_3 m_1 m_2 + \cdots + d_n m_1 m_2 m_3 \ldots m_{n-1}$$

Where $d_i, i = 1,2,\ldots,n$ are the Mixed Radix Digits (MRDs) and computed as follows:

$$d_1 = x_1$$

$$d_2 = \left|(x_2 - d_1)|m_1^{-1}|_{m_2}\right|_{m_2}$$

$$d_3 = \left|\left((x_3 - d_1)|m_1^{-1}|_{m_3} - d_2\right)|m_2^{-1}|_{m_3}\right|_{m_3}$$

$$\vdots$$

$$d_n = \left|\left(\ldots\left((x_3 - d_1)|m_1^{-1}|_{m_n} - d_2\right)|m_2^{-1}|_{m_n} - \cdots - d_{n-1}\right)|m_{n-1}^{-1}|_{m_n}\right|_{m_n} \quad (6)$$

That is, X in the interval $[0, M]$ can be uniquely represented.

Theorem1: Given the moduli set $\{2^n - 1, 2^n, 2^{n+1} - 1\}$, where $m_1 = 2^n - 1$, $m_2 = 2^n$ and $m_3 = 2^{n+1} - 1$ for every integer $n > 1$, the following hold true:

$$|m_1^{-1}|_{m_2} = -1 \quad (7)$$

$$|m_2^{-1}|_{m_3} = 2 \quad (8)$$

$$|m_1^{-1}|_{m_3} = -2 \quad (9)$$

Proof: If it can be demonstrated that $\left|m_i^{-1} \times m_i\right|_{m_i} = 1$, then m_i^{-1} is the multiplicative inverse of m_i with respect to m_i. Thus;

For (7), $|(2^n - 1) \times (-1)|_{2^n} = |(-1) \times (-1)|_{2^n}$

$$= |1|_{2^n} = 1$$

Also for (8), $|(2^n) \times 2|_{2^{n+1}-1} = |2^{n+1}|_{2^{n+1}-1}$

$$= |1|_{2^{n+1}-1} = 1$$

Finally for (9), $|(2^n - 1) \times (-2)|_{2^{n+1}-1} = |(-1) \times (2^{n+1} - 2)|_{2^{n+1}-1} = |2 - 1|_{2^{n+1}-1}$

$$= |1|_{2^{n+1}-1} = 1$$

Therefore we can re-write (6) as;

$$d_1 = x_1$$

$$d_2 = |(x_2 - d_1)(-1)|_{2^n} = |x_2 + (-x_1)|_{2^n}$$

$$d_3 = |((x_3 - d_1)(-2) - d_2)(2)|_{2^{n+1}-1}$$

$$= |(2^2 x_1 - 2^2 x_3 - 2d_2|_{2^{n+1}-1} \quad (10)$$

And (5) then becomes;

$$X = x_1 + 2^n d_2 - d_2 + 2^{2n} d_3 - 2^n d_3 \quad (11)$$

3.3 Hardware Realisation

We now simplify equations (10) and (11) as follows:

$$d_1 = x_{1,n-1} x_{1,n-2} \ldots x_{1,1} x_{1,0} \quad (12)$$

$$d_2 = \left|\underbrace{x_{2,n-1}x_{2,n-2}\ldots.x_{2,1}x_{2,0}}_{n-bits} + \underbrace{x_{1,n-1}x_{1,n-2}\ldots.x_{1,1}x_{1,0}}_{n-bits}\right|_{2^n}\Big|_{2^n}$$

$$= \underbrace{d_{2,n-1}d_{2,n-2}\ldots d_{2,1}d_{2,0}}_{n-bits} \quad (13)$$

$$d_3 = \Big|\underbrace{2^2(x_{1,n-1}x_{1,n-2}\ldots.x_{1,1}x_{1,0})}_{n-bits} - \underbrace{2^2(x_{3,n-1}x_{3,n-2}\ldots.x_{3,1}x_{3,0})}_{n+1} - 2\underbrace{(d_{2,n-1}d_{2,n-2}\ldots.d_{2,1}d_{2,0})}_{n-bits}\Big|_{2^{n+1}-1}$$

$$= \Big|\underbrace{x_{1,n-3}x_{1,n-4}\ldots.x_{1,n-1}x_{1,n-2}}_{n} + \underbrace{\bar{x}_{3,n-2}\bar{x}_{3,n-3}\ldots.\bar{x}_{3,n}\bar{x}_{3,n-1}}_{n+1} + d_{2,n-2}d_{2,n-3}\ldots.d_{2,n-1}\Big|_{2^{n+1}-1}$$

$$= \underbrace{d_{3,n}d_{3,n-1}\ldots\ldots.d_{3,1}d_{3,0}}_{(n+1)-bits} \ldots\ldots(14)$$

Thus, (11) will be realised as

$$X = \underbrace{A_{2n-1}A_{2n-2}\ldots.A_0}_{2n} + \overbrace{\bar{d}_{2,n-1}\bar{d}_{2,n-2}\ldots.\bar{d}_{2,0}}^{n}$$

$$+ \underbrace{d_{3,n}d_{3,n-1}\ldots.d_{3,0}}_{n+1}\overbrace{00\ldots.0}^{2n} + \underbrace{\bar{d}_{3,n}\bar{d}_{3,n-1}\ldots.\bar{d}_{3,0}}_{n+1}\overbrace{11\ldots1}^{n}$$

$$= \underbrace{\overbrace{00\ldots0}^{n+1}\underbrace{A_{2n-1}A_{2n-2}\ldots.A_0}_{2n}}_{3n+1} + \underbrace{\overbrace{00\ldots0}^{2n+1}\overbrace{\bar{d}_{2,n-1}\bar{d}_{2,n-2}\ldots.\bar{d}_{2,0}}^{n}}_{3n+1}$$

$$+ \underbrace{\underbrace{d_{3,n}d_{3,n-1}\ldots.d_{3,0}}_{n+1}\overbrace{00\ldots.0}^{2n}}_{3n+1} + \underbrace{\overbrace{11\ldots1}^{n}\underbrace{\bar{d}_{3,n}\bar{d}_{3,n-1}\ldots.\bar{d}_{3,0}}_{n+1}\overbrace{11\ldots1}^{n}}_{3n+1}$$

$$\underbrace{\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}_{3n+1}$$

(15)

Where,

$$A = \underbrace{x_{1,n-1}x_{1,n-2}\dots x_{1,1}x_{1,0}}_{n}\overbrace{00\dots0}^{n}$$

$$\bowtie \underbrace{d_{2,n-1}d_{2,n-2}\dots d_{2,1}d_{2,0}}_{n}\overbrace{00\dots0}^{n}$$

$$= \underbrace{x_{1,n}x_{1,n-1}\dots x_{1,1}x_{1,0}d_{2,n-1}d_{2,n-2}\dots d_{2,1}d_{2,0}}_{2n}$$

$$= A_{2n-1}A_{2n-2}\dots A_1A_0$$

The proposed schematic diagram will be as follows:

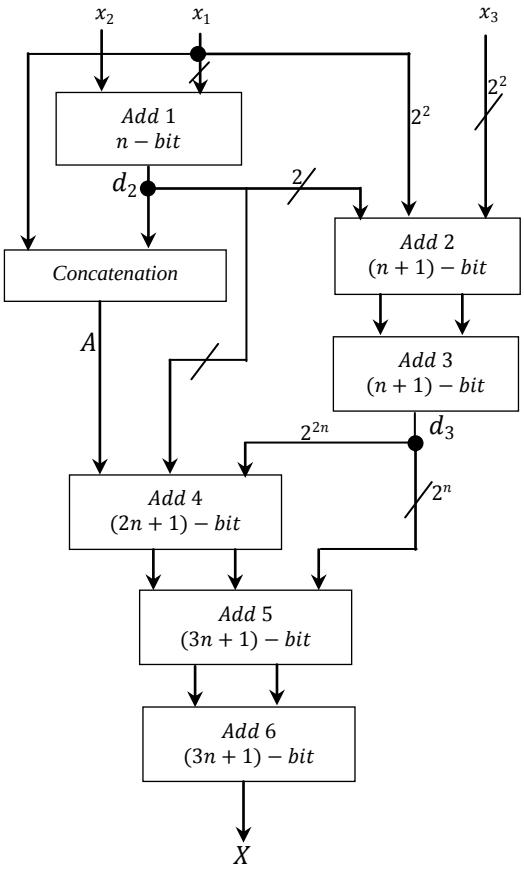

Fig 2: Hardware Architecture of Level-1 Decryption Scheme

The hardware realisation of the reverse converter involves an initial routing of the residues after which six simple fast adders – three CSAs and three CPAs are used as shown in the figure above to get the decimal/binary equivalent of the RNS number.

The area and delay requirements of the architecture will be as follows:

$$Area(A) = A_{Add\,1} + A_{Add\,2} + A_{Add\,3} + A_{Add\,4} + A_{Add\,5} + A_{Add\,6}$$

$$= nA + 2(n+1)A + (2n+1)A + 2(3n+1)A$$

$$= (11n+5)A$$

$$Delay(D) = D_{Add\,1} + D_{Add\,2} + D_{Add\,3} + D_{Add\,4} + D_{Add\,5} + D_{Add\,6}$$

$$= 2n + D + 2(n+1) + D + D + 2(3n+1)$$

$$= (10n+7)D$$

Since the delay of a CSA is *D* and that of a regular CPA is twice the area.

4. PERFORMANCE ANALYSIS

In the traditional RSA cryptosystem, for a message, m such that $c = m^e < n$, then *m* can be recovered from *c* by taking e^{th} roots over the integers, which is easy. However when $c = m^e < n$ passes through the second stage of encryption system, will cause m^e change to something else which will be presented in the form (r_1, r_2, r_3) since the moduli are very small. The idea is illustrated below;

Let p = 13, q = 17, e = 5 (e is an odd public exponent between3 and n – 1 that is relatively prime to p – 1 and q – 1)

Then $N = pq = 221, \Phi(N) = (p-1)(q-1) = 192$ which implies $e.d = 1 \bmod \Phi(N)$. Therefore d = 77. To encrypt the binary message m = 10 with respect to the key $p_k = (N = 221, e = 5)$, taking *m* as 2 (hence an element of Zn = 221) in the natural way. Computing $c = 25 \bmod 221 = 32$. We have c = 32 which is less than n. So taking e^{th} root of c = 32 will recover the original message with ease. That is $\sqrt[5]{32} = 2$.

The proposed system will transform $c_1 = 32$ using the moduli set as pk_2 into $c_2 = (2, 0, 4)$ and only a knowledge of the pk_2 which is the moduli set can recover the original $c_1 = 32$.

Table 1: RSA results evaluation

Message(m)	m^e	Cipher text (c)	e√c	Key-Length
12	248832	207	2.905	7-bits
5	3125	31	1.987	7-bits
2	32	32	2	7-bits

In the table above, transmitting 12 and 5 is secured because m^e is greater than n = 221. Transmitting 2 however is not secured because m^e is less than n = 221. Taking e^{th} root of c = 32 will recover the original message without the knowledge of the prime numbers.

Table 2: Proposed system results evaluation with $(2^n - 1, 2^n, 2^{n+1} - 1)$ moduli set

Message(m)	m^e	C_1	C_2	Key-Length
12	248832	207	(0, 3, 4)	14-bit
5	3125	31	(1, 3, 3)	14-bit
2	32	32	(2, 0, 4)	14-bit

In the table above, transmitting 12, 5 and 2 are secured because the second level of encryption will transform C1 into C2. This will not only cause more confusing to the hacker, but also enable the message 2 to be encrypted.

The key-length in the proposed system is enhanced because the moduli set are part of the private component of the classical RSA cryptosystem. The security of a cryptosystem is proportional to the length of the private key

Total delay for stage-2 encryption and stage-1 decryption is $(12n + 10)D$ and the total cost for both forward and reverse conversion is $(15n + 5)A$

5. CONCLUSION

Information security is very vital and a major factor in determining the quality of service in data transmission. There is no such thing as perfect security; we need to concentrate

more on making our information difficult to steal and making meaning out of it. RSA encryption system is the most widely used public key encryption system to hide information from unauthorized access and other malicious activities.

This paper designed and implemented an improved system of the tradition RSA cryptosystem by having two stages of encryption. The first stage is the traditional RSA and the second stage is to pass the cypher text obtained from RSA into smaller moduli such that smaller messages that cannot be encrypted by the traditional RSA will be able to do so by the smaller moduli. The key length is also enhanced as the moduli are part of the private key. The security of a cryptosystem is proportional to the length of the private key. This will help reduce the vulnerability to attacks like brute force.

5. REFERENCES

[1] Burt Kaliski, ―The Mathematics of the RSA Public-Key Cryptosystem‖, RSA Laboratories. April 9, 2006

[2] Rivest, R. L.—Shamir, A.—Adleman, L.: A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. Communications of ACM, Vol. 2, pp. 120–126, 1978

[3] P.W. Shor, Algorithms for Quantum Computation: Discrete Logarithms and Factoring. In IEEE Symp. on Found. of Comp. Sci., p.124-134, 1994

[4] Paulo Martins, Leonel Sousay, Programmable RNS Lattice-Based Parallel Cryptographic Decryption, IEEE 26th International Conference on Application-specific Systems, Architectures and Processors (ASAP) pp: 149-153, 2015

[5] F.J. Taylor, ―Residue Arithmetic: A Tutorial with Examples,‖ IEEE Computer Society Press Los Alamitos, CA, USA vol. 17, no. 5, pp. 50-62, May 1984

[6] K. A. Gbolagade, An Efficient MRC based RNS-to-Binary Converter for the moduli set, $\{2^{2n+1}-1, 2^n, 2^{2n}-1\}$, AIMS SA, 2011

[7] K.A. Gbolagade and S.D. Cotofana, An O(n) Residue Number System to Mixed Radix Technique, IEEE International Symposium on Circuits and Systems (ISCAS 2009), pp. 521-524, Taipei, Taiwan, China, May, 2009.

[8] H. Siewobr and K.A.Gbolagade, Modulo Operation Free Reverse Conversion in the Moduli Set $\{2^{2n+1}-1, 2^n, 2^{2n}-1\}$ *International Journal of Computer Applications (0975 – 8887) Volume 85 – No 18, January 2014*

[9] Edem K. Bankas, Kazeem A. Gbolagade, A New Efficient RNS Reverse Converter for the 4-Moduli Set, $\{2^n, 2^n + 1, 2^n - 1, 2^{2n+1} - 1\}$, International Journal of Computer, Electrical, Automation, Control and Information Engineering Vol:8, No:2, 2014