Global Journals Inc.

Global Journal of Computer Science and Technology
Interdisciplinary
Volume 12 Issue 10 Version 1.0 Year 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

Modeling a Secured Digital Image Encryption Scheme Using a Three Moduli Set

By B. A. Weyori, P. N. Amponsah & P. K. Yeboah

Catholic University College of Ghana, Fiapre-Sunyani, Ghana

Abstract - This paper proposes a new digital image coding scheme that uses a three moduli set with a common factor. The proposed scheme is specific to a particular three moduli set {2n+2, 2n+1,2n}. The design of the scheme is based on the residue to binary converter which achieves in terms of area and critical path delay as compared to the state of the art. This scheme offers high-speed processing because in the reverse converter the computation of the multiplicative inverse is eliminated, and it achieves low-power VLSI implementation for image processing such as digital image transform and digital image filtering.

Keywords : RNS, VLSI, image coding, CRT, multiplicative inverse, binary number system, converters, moduli set, critical path delay.

GJCST-E Classification: E.3

Modeling a Secured Digital Image Encryption Scheme Using a Three Moduli Set

B. A. Weyori [α], P. N. Amponsah [α] & P. K. Yeboah [α]

Abstract - This paper proposes a new digital image coding scheme that uses a three moduli set with a common factor. The proposed scheme is specific to a particular three moduli set {2n+2, 2n+1,2n}. The design of the scheme is based on the residue to binary converter which achieves in terms of area and critical path delay as compared to the state of the art. This scheme offers high-speed processing because in the reverse converter the computation of the multiplicative inverse is eliminated, and it achieves low-power VLSI implementation for image processing such as digital image transform and digital image filtering.

Keywords : RNS, VLSI, image coding, CRT, multiplicative inverse, binary number system, converters, moduli set, critical path delay.

I. Introduction

Data encryption is important to the security and integrity of information to be transmitted through a network. The need for a secured communication is more profound than ever, recognizing the fact that the conduct of almost all our business and personal matters are carried out today by computer networks [1]. Hence, in an environment where data encryption applications are fast-evolving, an algorithm that offers efficient and low-complexity encryption can provide security for information against intrusion and sophisticated threats that abound now. Moreover, the information that has to be transmitted must be encrypted to reduce the size of the data and increase processing speed.

As part of the information transmission, images are extensively used in such fields as desktop publishing, medical imaging, military target analysis, manufactured automation control, machine vision, geo-physical imaging, graphic arts and multimedia [1]. The application of these image features is so dependent on the type of hardware required for high performance that very large scale integration (VLSI) technology becomes vital for digital image processing [2].

One method of designing high-speed and low power VLSI digital systems is by using the residue number system (RNS) [2]. The RNS has such inherent features as parallelism modularity, fault tolerance and carry-free propagation. These features make RNS widely used in Digital Signal Processing (DSP) application such as digital filtering, convolution, Fast Fourier Transform (FFT) and image processing [3],[4],[5]. Thus, RNS is the best tool to employ for a secured, fast and successful image data or image pixels or image elements encryption method.

A two-dimensional image function can be viewed as f(x,y), where x and y are spatial (plane) coordinates and f is the amplitude at any pair coordinate (x,y), called the intensity or gray level of an image at that point. When x,y and the amplitude values are all finite or discrete quantities, the object so formed is a digital image [6],[7].

In this paper, we present an image coding scheme which is based on the residue to binary converter for the three moduli set {2n+2,2n+1,2n} presented in [3]. The image coding scheme achieves speed and area in terms of security.

To clarify this framework, this study gives some background to the image processing scheme, which leads to the proposed coding techniques. The study concludes from an analysis of the encoding and decoding process.

II. Backround

Images are coded and processed to produce results that are more secured and protected than the original images for specific applications. Hence the best approach for achieving a secured, high-speed and low-power VLSI implementation for digital image coding and processing is by the use of the residue number system (RNS) [1],[2]. A residue number system is defined in terms of a relatively prime moduli set $\{m_i\}_{i=1,\ldots n}$ such that $\gcd(m_i, m_j) = 1$ for ≠j, where gcd means greatest common divisor of m_i and m_j, while $M = \Pi_{i=1}^{n} m_i$, is the dynamic range. The residues of a decimal number can be obtained as $x_i = \left|X\right|_{m_i}$ thus X can be represented in RNS as X= $(x_1, x_2, x_3, \ldots x_n)$, $0 \le x_i < m_i$. This representation is unique for any integer in RNS, $X \in [0, M-1].$ In this paper, $X \bmod m_i$ will be represented as $\left|X\right|_{m_i}$.

Furthermore, RNS is a carry-free system for addition, subtraction and multiplication operations. Given two integer numbers K and L, RNS is represented

Author α : Faculty of Information and Communication Sciences and Technology, Catholic University College of Ghana, Fiapre-Sunyani, Ghana.

by $K=(k_1,k_2,k_3,...k_n)$ and $L=(l_1,l_2,l_3,...l_n)$ respectively. In this case, we use the operator Θ for addition, subtraction and multiplication. Thus, we calculate $W=K\Theta L$ as $W=(w_1,w_2,w_3,...w_n)$, where $w_i = |k_i\Theta l_i|_{m_i}$, for i=1,n. The complexity of the calculation of this operation Θ is determined by the number of bits required to represent the residue and not by the one required to represent the input operands [3], [7], [8].

This RNS system achieves high speed computation because of its parallel computing nature. In order to convert numbers from binary to residue numbers, a residue-to-binary converter is required at the front end. Then, to convert back from residue to binary a residue-to-binary converter is required at the back end. The residue-to-binary converter usually consists of a lot of moduli operations; the computation of which is tedious. The reverse converter (residue-to-binary) is a crucial part of the RNS system. To perform the conversion of residue-to-binary, that is convert the residue number $(x_1,x_2,x_3,...x_n)$ into the binary number X, the traditional CRT is used [1][2][9],[10]. The traditional CRT is shown in equation (1):

$$X = \left|\sum_{i=1}^{n} M_i \left|M_i^{-1} x_i\right|_{m_i}\right|_M \qquad (1)$$

Where $M=\prod_{i=1}^{n} m_i$, $M_i = \frac{M}{m_i}$, and M_i^{-1} is the multiplicative inverse of M_i with respect to m_i. The moduli set, $\{m_i\}_{i=1,...n}$, must be pairwise and relatively prime for the equation (1) to be used. In this case, the moduli set {2n+2, 2n+1,2n} has a common factor. This simply implies that for equation (1) to be used in the conversion back to binary the moduli set must be mapped to a set of relatively prime moduli. Hence the decimal conversion of $(x_1,x_2,x_3,...x_n)$ for the moduli set which are not pairwise relatively prime can be computed as follows [3], [6]:

$$X = \left|\sum_{i=1}^{n} \alpha_i x_i\right|_{M_L} \qquad (2)$$

Where M_L is the Lowest Common Multiple (LCM) of $\{m_i\}_{i=1,...n}$, the set of moduli sharing a common factor, X is the decimal equivalent of $\{x_i\}_{i=1,...n}$, α_i is an integer such that $|\alpha_i|_{\frac{M_L}{\mu_i}} = 0$ and $|\alpha_i|_{\mu_i} = 1$, and $\{\mu_i\}_{i=1,...n}$ is a set of integers and such that $M_L = \prod_{i=1}^{n} \mu_i\mu_i$ and divides m_i. It should be taken into consideration that α_i may not exist for some i. The modified form of the equation (2) is shown as equation (3):

$$|X|_{M_L} = \left|\sum_{i=1}^{n} \beta_i \left|\beta_i^{-1}\right|_{\mu_i} x_i\right|_{M_L} \qquad (3)$$

Therefore, a software-based RNS image coding scheme has been proposed as a good tool in image data coding [1][2]. This paper codes the entire image data and so makes it more detailed and achieves a high-speed and low-power VLSI implementation.

III. Proposed Image Encription Scheme

RNS can serve three goals, namely, to increase the speed of transmission, reduce the area of image data, and increase the security level of transmission through computer networks.

a) New Method for Image Data Coding

The image data coding system consists of an encoder and a decoder. The moduli set {2n+2, 2n+1,2n}, which has a common factor is used for the image coding scheme. The encoder is built by a R/B converter, which requires an RNS image processor of small wordlength. The decoder is used to recover the encrypted bitstream according to the moduli set and the proposed conversion technique in [3]. The modified RNS-to-Binary conversion method does not require the computation of a multiplicative inverse and also reduces the problem of the large modulo M as compared to the conversion using the traditional CRT. Considering the reduction in the large **mod-M** to **mod-n** and the elimination of the computation of the multiplicative inverse the proposed image coding scheme achieves reduced area, increased speed and decrease in internal delay of the conversion from RNS to binary.

b) Security of the Proposed Coding Scheme

Compared to the binary image coding, the proposed RNS image coding scheme has an encoder and a decoder, which is designed based on the operation of a three moduli set with a common factor. The end results of the RNS image encoder in this new scheme are in small-wordlength and are arranged into a certain encrypted order. An intruder who breaks into the network does not know the moduli set and the order of the encrypted bitstream that are computed in parallel. Only the designed decoder with the correct R/B converter and moduli set can recognize and decode the encrypted bitstream back to the processed and transformed digital image data according to the way they are arranged.

The proposed scheme achieves high-speed and low-power VLSI implementation for image processing such as digital image transform and digital image filtering. The design of the scheme is based on the residue to binary converter presented in [1] as shown below:

Theorem 1: Given the moduli set (m_1, m_2, m_3) and the dynamic range $M = P_1P_2P_3$, the residue number (x_1, x_2, x_3) is converted into binary number by:

$$X = (x_2 - x_1)m_1 + x_1 + m_1m_2\left|\frac{(x_1 + x_3)}{2} - x_2\right|_{\frac{m_3}{2}} \qquad (4)$$

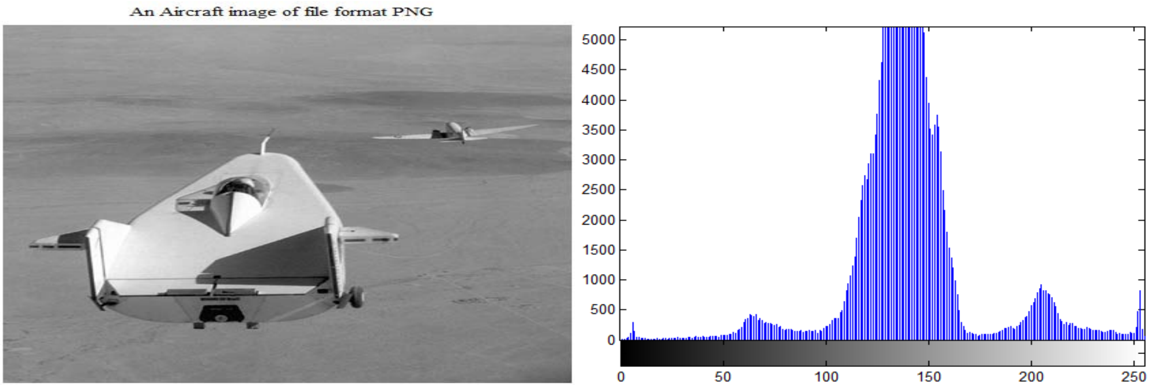


Fig. 1 : An Original image of an Aircraft and A histogram showing the distribution of the image pixels

Table 1 : A representation of a 12-by-12 image data extracted out of the aircraft image data of size 512 by 512

154	153	155	155	153	154	154	155	154	152	155	154
156	156	157	155	153	155	155	154	156	156	158	157
156	159	160	154	153	155	153	155	155	156	157	156
158	161	157	158	157	157	159	155	156	157	160	159
159	161	158	159	160	161	156	155	156	158	158	156
156	158	157	160	158	158	158	156	154	156	157	156
158	158	158	156	157	157	154	154	155	153	152	155
156	155	155	153	156	154	155	154	153	157	153	153
155	155	154	154	157	157	156	155	158	160	158	157
156	159	158	159	157	157	155	157	160	158	156	158
157	157	157	159	156	156	156	157	160	159	156	158
153	156	158	158	160	157	156	158	159	159	156	156

The conversion was done using n=3, since the dynamic range of the grayscale image is 255, using the moduli set {2n +2, 2n + 1, 2n}, the moduli set will form {8, 7, 6}.

Table 2 : Conversion of 12-by-12 decimal image pixels to RNS with moduli 2n + 2

2	1	3	3	1	2	2	3	2	0	3	2
4	4	5	3	1	3	3	2	4	4	6	5
4	7	0	2	1	3	1	3	3	4	5	4
6	1	5	6	5	5	7	3	4	5	0	7
7	1	6	7	0	1	4	3	4	6	6	4
4	6	5	0	6	6	6	4	2	4	5	4
6	6	6	4	5	5	2	2	3	1	0	3
4	3	3	1	4	2	3	2	1	5	1	1
3	3	2	2	5	5	4	3	6	0	6	5
4	7	6	7	5	5	3	5	0	6	4	6
5	5	5	7	4	4	4	5	0	7	4	6
1	4	6	6	0	5	4	6	7	7	4	4

Table 3 : Conversion of 12-by-12 decimal image pixels to RNS with moduli 2n + 1

0	6	1	1	6	0	0	1	0	5	1	0
2	2	3	1	6	1	1	0	2	2	4	3
2	5	6	0	6	1	6	1	1	2	3	2
4	0	3	4	3	3	5	1	2	3	6	5
5	0	4	5	6	0	2	1	2	4	4	2
2	4	3	6	4	4	4	2	0	2	3	2
4	4	4	2	3	3	0	0	1	6	5	1
2	1	1	6	2	0	1	0	6	3	6	6
1	1	0	0	3	3	2	1	4	6	4	3
2	5	4	5	3	3	1	3	6	4	2	4
3	3	3	5	2	2	2	3	6	5	2	4
6	2	4	4	6	3	2	4	5	5	2	2

Table 4 : Conversion of 12-by-12 decimal image pixels to RNS with moduli 2n

4	3	5	5	3	4	4	5	4	2	5	4
0	0	1	5	3	5	5	4	0	0	2	1
0	3	4	4	3	5	3	5	5	0	1	0
2	5	1	2	1	1	3	5	0	1	4	3
3	5	2	3	4	5	0	5	0	2	2	0
0	2	1	4	2	2	2	0	4	0	1	0
2	2	2	0	1	1	4	4	5	3	2	5
0	5	5	3	0	4	5	4	3	1	3	3
5	5	4	4	1	1	0	5	2	4	2	1
0	3	2	3	1	1	5	1	4	2	0	2
1	1	1	3	0	0	0	1	4	3	0	2
3	0	2	2	4	1	0	2	3	3	0	0

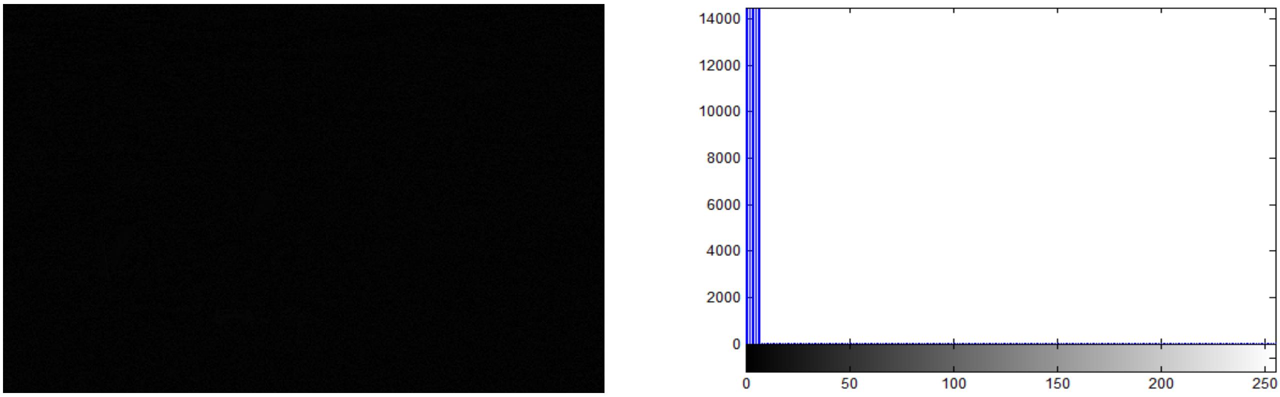


Fig. 2 : An encrypt image of an Aircraft with moduli set 2n + 2 and A histogram showing the distribution of the encrypted image pixels

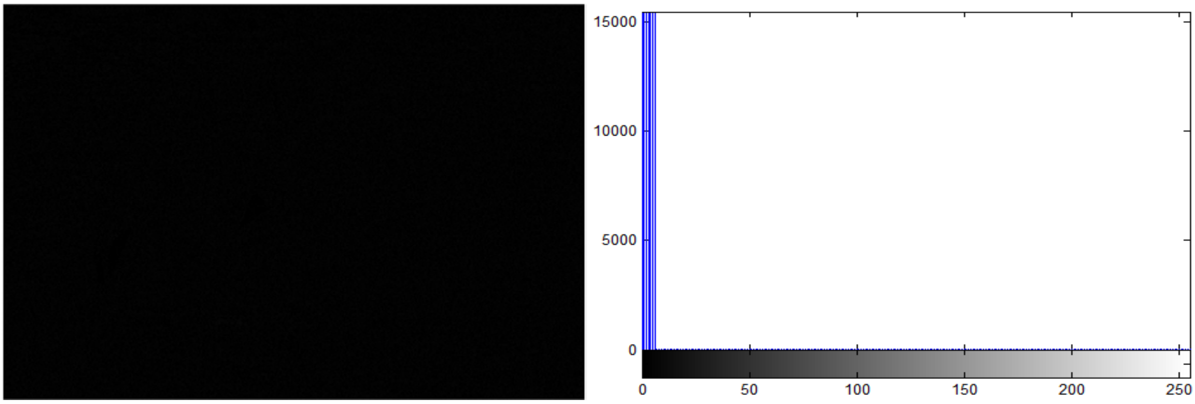


Fig. 3 : An encrypt image of an Aircraft moduli set 2n +1 and A histogram showing the distribution of the encrypted image pixels

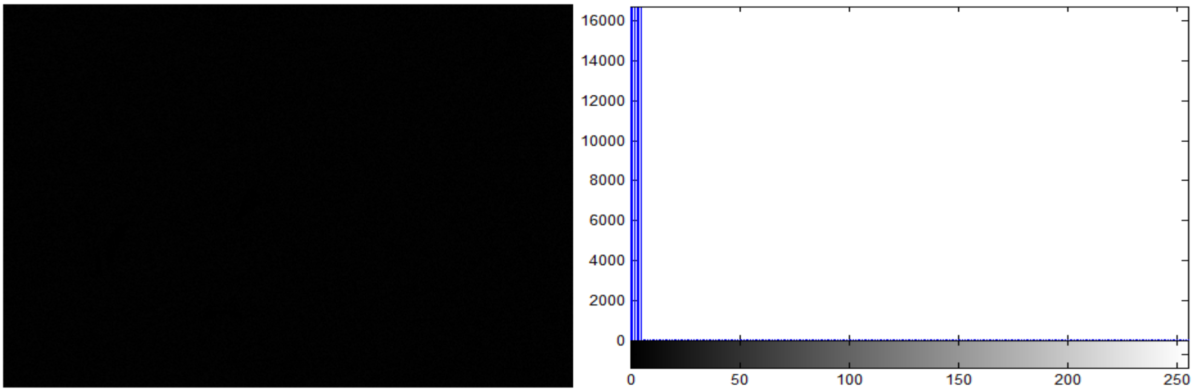


Fig. 4 : An encrypt image of an Aircraft moduli set 2n and A histogram showing the distribution of the encrypted image pixels

Example 1: Given the moduli set {2n +2, 2n +1, 2n} and modulis $(2,0,4)$. Convert the encrypted pixel bitstream from RNS-to-Binary using the proposed technique in [1]. Where n = 3.

$m_1 = 8, m_2 = 7, m_3 = 6, x_1 = 2, x_2 = 0$ and $x_3 = 4$

$$X = (0-2)8 + 2 + (8\times 7)\left|\frac{(2+4)}{2} - 0\right|_{\frac{8}{2}}$$

$$X = (-2)8 + 2 + 56\left|\frac{6}{2}\right|_4$$

$$X = -16 + 2 + 56(3)$$

$$X = 154$$

IV. Encoder and Decoder

The designed system is divided into two parts. The first part deals with the encoding that is mainly carried out by the encoder (B/R converter). The second part involves the decoding, which is implemented by the decoder (R/B converter).

a) Encoder

1. Read the original digital image signal as binary or decimal value.
2. The digital image data or elements are encrypted into bitstream in a certain order according to the moduli set.
3. The encrypted bitstream is processed by the RNS image processor and the output is sent.

b) Decoder

1. The processed encrypted bitstream (digital image data encoded with RNS) is received and recognized.
2. The decoder with the correct moduli set is used to decode the encrypted bitstream back to binary or decimal so that it is easily read by the computer.

Fig. 1 : Encoder

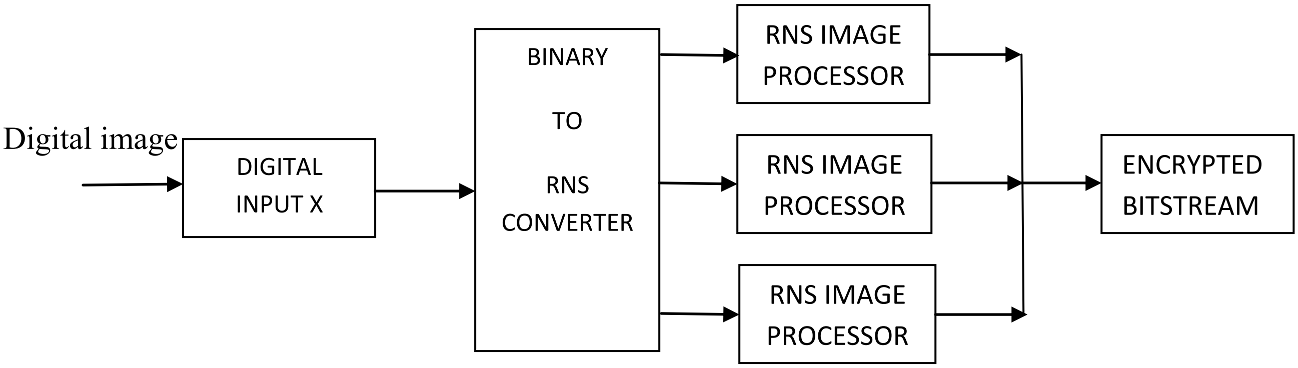
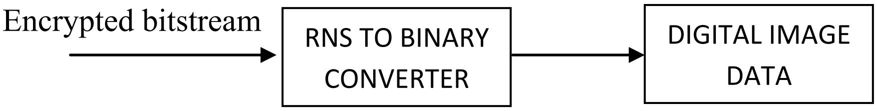


Fig. 2 : Decoder



V. Conclusion

Data encryption is the process of transforming information (referred to as plaintext) using an algorithm (called a cipher) to make it unreadable to anyone except those possessing special knowledge, usually referred to as a key. The result of the process is encrypted information (in cryptography, referred to as cipher text). The reverse process, i.e. to make the encrypted information readable again is referred to as decryption, (i.e. to make it unencrypted).

In this paper, we built an encryption and decryption scheme based on a three moduli set. We demonstrated the security strengths of the encryption scheme. The proposed scheme outperforms most of the encryption schemes in terms of area and delay due to the fact that our scheme operates on smaller magnitude operands as it requires less complex adders and multipliers, which potentially offers high-speed processing.

References Références Referencias

1. A. Ammar, A.Al Kabbany, M. Youssef and A. Emam, “A Secure image coding scheme using Residue Number System”, in proceedings of the 18th National Radio science conference, Egypt, pp. 339-405, March 2001.
2. W. Wang, M.N.S. Swamy and M.O. Ahmad, “RNS Application for Digital Image Processing”, 4th IEEE international workshop on System-on-chip for Real-time Application, pp. 77-80, July 2004.
3. A. Gbolagade, S. D. Cotofana, A residue to Binary Converter for the {2n+2,2n+1,2n} Moduli Set, Asilomar Conference on Signals, Systems, and Computers, California, USA, October 2008
4. Yuke Wang, Xiayu Song, Mostapha Aboulhamid and Hong Shem, “Adder based Residue to Binary Numbers Converters for (2^n- 1, 2^n, 2^n+ 1), IEEE Trans. On Signal Processing, Vol. 5, No. 7, pp. 1772-1779, July, 2002.
5. Wei Wang, M.N.S. Swamy, M.O. Ahmad and Yuke Wang, “A study residue-to-binary converter for three moduli RNS and a scheme of its VLSI implementation”, IEEE Trans. On circuits and systems I: Fundamental Theory and App., Vol. 50, No. 2, pp. 235-243, Feb. 2003.

6. Szabo, N. and Tanaka, R., “Residue arithmetic and its application to computer technology”, McGraw-Hill, New York 1967.
7. Rafeal C. Gonzalez and Richard E. Woods, Digital image processing; Second edition, Prentice-Hall, Inc. New Jersey, U.S.A., 2002.
8. A. Gbolagade and S.D. Cotofana, “MRC Technique for RNS to Decimal Conversion Using the Moduli Set {2n+2,2n+1,2n}”, Proceedings of the 16th Annual Workshop on Circuits, Systems and Signal Processing., pp. 318-321, Veldhoven, The Netherlands, November 2008
9. A. Gbolagade, S. D. Cotofana, Residue Number System Operands to Decimal Conversion for 3-Moduli Sets, Proceedings of 51st IEEE Midwest Symposium on Circuits and Systems (MWSCAS 08), pp. 791-794, Knoxville, USA, August 2008
10. B. Parhami, Computer Architecture: Algorithms and Hardware Designs, Oxford University Press, 2000.

This page is intentionally left blank