FACTA UNIVERSITATIS
Series: **Electronics and Energetics** Vol. 29, No 1, March 2016, pp. 101 - 112
DOI: 10.2298/FUEE1601101K

BINARY TO RNS ENCODER FOR THE MODULI SET $\{2^n - 1, 2^n, 2^n + 1\}$ WITH EMBEDDED DIMINISHED-1 CHANNEL FOR DSP APPLICATION

Ivan Krstić[1], Negovan Stamenković[2], Vidosav Stojanović[3]

[1]University of Priština, Faculty of Technical Science, Serbia
[2]University of Priština, Faculty of Natural Sciences and Mathematics, Serbia
[3]University of Niš, Faculty of Electronic Engineering, Serbia

Abstract: A binary-to-residues encoder (forward encoder) is an essential building block for the residue number system digital signal processing (RNS DSP) and as such it should be built with a minimal amount of hardware and be efficient in terms of speed and power. The main parts of the forward encoder are residue generators which are usually classified into two categories: the one based on arbitrary moduli-set which make use of look-up tables, and the other based on the special moduli sets. A new memoryless architecture of binary-to-RNS encoder based on the special moduli set $\{2^n - 1, 2^n, 2^n + 1\}$ with embedded modulo $2^n + 1$ channel in the diminished-1 representation is presented. Any of two channels (standard modulo $2^n + 1$, or modulo $2^n + 1$ in the diminished-1 representation) operation can be performed by using a single switch. The proposed encoder has been implemented on a Xilinx FPGA chip for the various dynamic range requirements.

Keywords: RNS system, special moduli set, forward encoder, diminished-1 encoded channel, modulo carry save adders, Virtex FPGA.

1 Introduction

Residue Number System [1, 2] is a non-weighted integer number system in which arithmetic operations are limited to the addition, subtraction and multiplication. Other arithmetic operations such as division, sign detection, overflow, scaling

Received September 24, 2014; received in revised form September 1, 2015
Corresponding author: Ivan Krstić
University of Priština, Faculty of Technical Sciences, Serbia
(e-mail: ivan.krstic@pr.ac.rs)

and magnitude comparison are non-modular and quite complex for implementation. The RNS is defined in terms of a set of relatively prime moduli called RNS basis.

Special moduli set $\{2^n-1, 2^n, 2^n+1\}$ has gained popularity and is expected to play an important role in RNS digital signal processing [3]. In comparison with the other moduli sets, special moduli set has the advantage of low-cost forward conversion, modulo reduction, and the reverse conversion. Thus, the use of this moduli set can significantly reduce hardware complexity and delay [4].

The RNS DSP consists of three major parts: binary-to-residue encoder, modular arithmetic channels and residue to binary decoder (reverse converter). The forward encoder and reverse converter are needed to achieve the RNS representation of the binary number and vice versa, respectively [5]. The above-mentioned modular operations, required by each modular arithmetic channel, are inherently carry-free addition, multiplication and borrow-free subtraction, which means that each digit of the resulting number is a function of only one digit from each operand and independent of the others. This is the most attractive feature of RNS that enables one to design highly parallel structures for computation which leads to speed improvement required for the DSP applications [6, 7, 8, 9, 10].

A binary-to-residues encoder is an essential building block for a residue number system and as such it should be built with a minimal amount of the hardware and along with that be efficient in terms of speed and power. Conceptually, binary-to-residues encoder involves computation of the remainders of the input bit stream with respect to the each modulus in the RNS moduli set. In other words, the binary-to-residues encoder maps a binary weighted number into a finite ring [11]. A finite ring is a set of finite elements over which the modular addition and the modular multiplication operations are defined.

Main parts of binary-to-residues encoder are the forward converters (residue generators). The forward converters are usually classified into the next two categories: the one based on the arbitrary moduli-sets [12, 4] which are usually built using the look-up tables, and the other based on special moduli-sets [13, 14, 15]. The use of special moduli-sets simplifies the forward conversion algorithms and such forward converters can be realized using only combinational logic.

The dynamic range of RNS system, which is equal to the product of the modulus of three moduli-set base $\{2^n-1, 2^n, 2^n+1\}$, is $M = 2^{3n} - 2^n$ i.e. corresponds to the $3n$ bits. Thus, any $3n$-bit unsigned binary integer X can be uniquely represented by its residues: $X = (x_1, x_2, x_3)$, where x_1 is the reminder when X is divided by modulo $2^n - 1$ denoted as $\langle X \rangle_{2^n-1}$, $x_2 = \langle X \rangle_{2^n}$ and $x_3 = \langle X \rangle_{2^n+1}$.

The diminished-1 number system [16] can be used to represent modulo $2^n + 1$ residue ($x_3 = \langle X \rangle_{2^n+1}$) as: $x_3' = \langle X - 1 \rangle_{2^n+1}$. Thus, the each operand is represented decreased by one, and the zero operands are not used in the computation channel.

In the diminished-1 representation, x'_3 is represented as $2^n x_{3,n} + X'_3$, where $x_{3,n}$ is zero indication bit, and X'_3 is n-bit number part. If $x'_3 > 0$, $x_{3,n} = 0$ and $X'_3 = X_3 - 1$, whereas for $x'_3 = 0$; $x_{3,n} = 1$, and $X'_3 = 0$. Thus, for the diminished-1 representation, the residue of $X - 1$ modulo $2^n + 1$ instead of the residue X modulo $2^n + 1$ is used. The results of arithmetic operations are derived alternatively when any of operands or the result is equal to zero [17, 18, 19]. In this way, the diminished-1 representation can lead to the implementations with delay and area complexity approaching that of the modulo $2^n - 1$ channel.

This paper presents a binary-to-residues encoder based on special moduli set $\{2^n - 1, 2^n, 2^n + 1\}$ with embedded diminished-1 encoded channel, which unifies the encoders architectures presented in [15]. Theoretical background of forward converters for $2^n - 1$ and 2^n channel remains the same as in [15], while new forward converter for $2^n + 1$ channel with embedded diminished-1 encoded channel has been developed. The standard and the diminished-1 forward converters for modulo $2^n + 1$ channel are implemented on the same hardware. Thus, the standard $2^n + 1$ channel or the diminished-1 channel can be activated simply, by using the single switch.

The rest of the paper is organized as follows. In Section 2 we introduce the binary-to-residues memoryless encoder for special moduli set based only on the standard combinational logic and a novel design of the binary to residue encoder with embedded diminished-1 channel. Section 3 presents the hardware implementation and performance evaluation. Our conclusion is drawn in Section 4.

2 Binary-to-residues encoder

In our approach, the $3n$-bit input is divided into three n-bit sections to obtain the corresponding three residue numbers in parallel. An $3n$-bit integer in the range $0 \leq X \leq M - 1$ can be represented in power-of-two notation as [20, 21]:

$$X = \sum_{i=0}^{3n-1} b_i 2^i = N_2 \times 2^{2n} + N_1 \times 2^n + N_0 \,, \tag{1}$$

where

$$N_0 = \sum_{i=0}^{n-1} b_i 2^i, \; N_1 = \sum_{i=n}^{2n-1} b_i 2^{i-n} \text{ and } N_2 = \sum_{i=2n}^{3n-1} b_i 2^{i-2n} \,. \tag{2}$$

In order to obtain the RNS representation of the integer X, partitioned into three n-bit parts N_0, N_1 and N_2, three residue generators are required, one for each channel.

2.1 Forward conversion for modulo 2^n and modulo 2^n-1 channel

Forward conversion for modulo 2^n channel is quite simple, i.e. the residue x_2 can be obtained by truncation of X:

$$x_2 = \langle X \rangle_{2^n} = b_{n-1}b_{n-2}\cdots b_0\,. \tag{3}$$

The calculation of x_1 can be performed as a sequence of additions [15]:

$$x_1 = \langle N_2 + N_1 + N_0 \rangle_{2^n-1}\,, \tag{4}$$

which can be performed by CSA with EAC (carry save adder with end around carry) on whose inputs three n-bit operands (N_2, N_1, N_0) are connected, followed by the CPA with EAIC (carry propagate adder with end around inverted carry) and decrementer.

Slight modifications to the architecture of the modulo 2^n-1 residue generator presented in [15] are introduced: LSB full-adder of CPA with EAIC (end-around inverted carry) has been replaced with $\overline{\text{HA}}$ (full-adder with one input driven by the logical one), while MSB half-subtractor of decrementer has been replaced with XOR gate, Fig. 1. $\overline{\text{HA}}$ has the same complexity as the standard half-adder except for an extra inverter, whose delay and area consumption can be ignored if using the unit-gate model as a means of performance evaluation.

The critical path of the binary-to-modulo 2^n-1 converter is depicted by a dashed line.

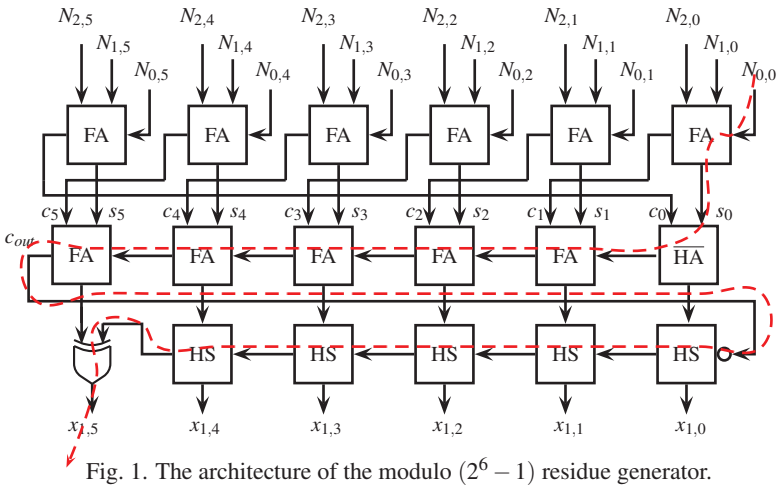

Fig. 1. The architecture of the modulo (2^6-1) residue generator.

Propagation delay of binary to modulo 2^n-1 converter, according to the unit-gate model, is $T_1 = 3n+4$. The area cost of binary to modulo 2^n-1 converter is

$A_1 = 17n - 5$.

2.2 Forward conversion for modulo $2^n + 1$ channel

Modulo $2^n + 1$ residue and modulo $2^n + 1$ residue in diminished-1 number system representation can be calculated by equations derived in [15]:

$$x_3 = \langle S + C + 1 \rangle_{2^n+1}, \tag{5}$$

$$x_3' = \langle S + C \rangle_{2^n+1}, \tag{6}$$

where S and C are n-bit partial sum and carry vectors generated by modulo $2^n + 1$ carry save adder, whose inputs are driven by N_2, $\overline{N_1}$ and N_0, where $\overline{N}_1 = 2^n - 1 - N_1$ is the one's complement of operand N_1 [22].

By introduction of a control bit d

$$d = \begin{cases} 1, & \text{if calculating } x_3 \\ 0, & \text{if calculating } x_3' \end{cases}, \tag{7}$$

equations (5) and (6) are combined to form

$$x_3'' = \langle S + C + d \rangle_{2^n+1}. \tag{8}$$

The modulo $2^n + 1$ addition of two n-bit operands $S = s_{n-1}s_{n-2}\ldots s_0$, $C = c_{n-1}c_{n-2}\ldots c_0$ and the 1-bit operand d is based on the following relation

$$x_3'' = \begin{cases} S + C + d, & S + C + d \le 2^n \\ S + C + d + (2^n - 1) - 2^{n+1}, & \text{otherwise} \end{cases}. \tag{9}$$

In order to implement (9) we can ignore the output carry (c_{out}) from 2^{n+1} position and add the constant value of $2^n - 1$ to the result of $A = S + C + d$ (in binary notation $A = a_n a_{n-1}\ldots a_0$), if $S + C + d > 2^n$. That is, the output of the residue generator should yield the value B which is obtained by adding the $(n+1)$-bit binary number $K = 0\underbrace{11\ldots1}_{n}$ to the binary number A, $B = a_n a_{n-1}\ldots a_0 + 0\underbrace{11\ldots1}_{n}$, where $a_n = c_{out}$.

Let p_{i+1} denote the carry from i-th bit position obtained while performing addition of the binary numbers A and K. It is obvious that:

$$\begin{aligned} p_1 &= a_0, \\ p_{i+1} &= p_i \vee a_i, \text{ for } i = 1, 2, \ldots, n-1, \end{aligned} \tag{10}$$

where $\vee$ corresponds to the logical OR operation.

Furthermore, the output vector B is:

$$\begin{aligned} b_0 &= \overline{a_0}, \\ b_i &= \overline{p_i \oplus a_i}, \text{ for } i = 1, 2, \ldots, n-1, \\ b_n &= p_n \oplus a_n . \end{aligned} \tag{11}$$

According to the value of the control signal sel either $A = a_n a_{n-1} \ldots a_0$ or $B = b_n b_{n-1} \ldots b_0$ should be connected to the output (if $sel = 0$ then $x''_3 = A$, else $x''_3 = B$):

$$x''_{3,k} = (b_k \wedge sel) \vee \left(a_k \wedge \overline{sel}\right) . \tag{12}$$

Considering the values of c_{out} and p_n, there are three cases to be discussed:

1. If $S + C + d < 2^n$, that is $c_{out} = 0$ and p_n can be zero or one, then $sel = 0$, which corresponds to the binary number A at the output.
2. If $S + C + d = 2^n$, that is $c_{out} = 1$ and $p_n = 0$, then $sel = 0$ and $x''_3 = A$.
3. If $S + C + d > 2^n$, that is $c_{out} = 1$ and $p_n = 1$, then $sel = 1$, which corresponds to the binary number B at the output.

According to the above discussion it can be concluded that $sel = c_{out} \wedge p_n$, where $\wedge$ corresponds to the logical AND operation.

Equation (12) can be simplified by putting (11) in (12):

$$\begin{aligned} x''_{3,0} &= a_0 \oplus sel, \\ x''_{3,i} &= (sel \wedge \overline{p_i}) \oplus a_i, \text{ for } i = 1, 2, \ldots, n-1, \\ x''_{3,n} &= a_n \wedge \overline{p_n} . \end{aligned} \tag{13}$$

Finally, the architecture of the binary to the modulo $2^n + 1$ converter with the embedded modulo $2^n + 1$ channel in the diminished-1 representation for $n = 6$ is given in the Fig. 2. Depending on the value of the control bit d, the converter gives either x_3 or x'_3. The critical path of the converter is depicted by the dashed line.

The theoretical formula for the propagation delay, i.e. conversion time, of the binary to the modulo $2^n + 1$ residue generator with the embedded diminished-1 channel is $T_3 = 2n + 10$. The area cost is equal to $A_3 = 18n$.

The validity of the modulo $2^n + 1$ channel and the diminished-1 encoded channel operation of the binary to residues encoder for the 16-th bit input number and $n = 6$ is demonstrated in the following example. Let $X = 54\,425 = 1101\,010010\,011001$. The carry save adder with end around inverted carry (EAIC) reduces the three 6-bit inputs N_0, N_1 and N_2 to the two 6-bit numbers: the partial sum sequence (S) and the partial carry sequence (C)

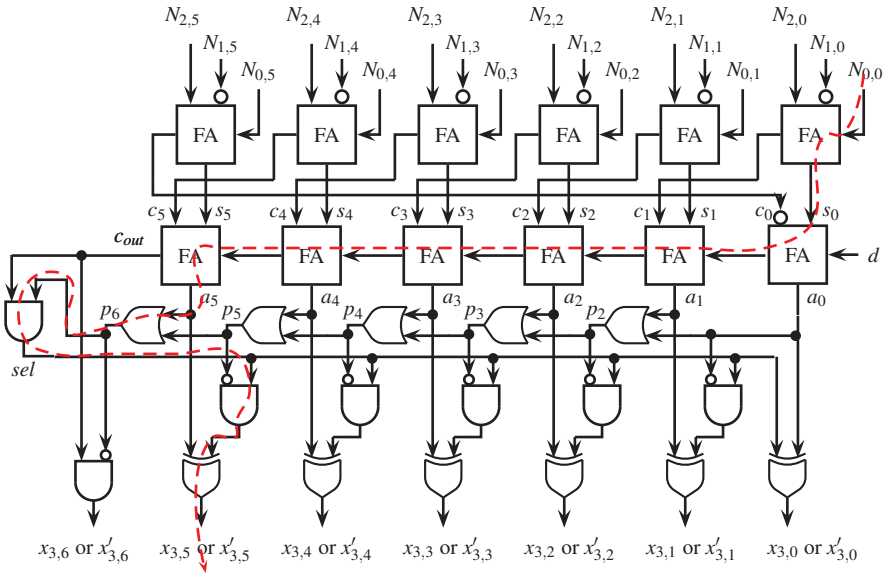

Fig. 2. The architecture of binary to modulo (2^6+1) converter with the embedded modulo (2^6+1) channel in the diminished-1 representation. For $d=1$ we have $x_3 = \langle X \rangle_{2^6+1}$, but for $d=0$ we have diminished-1 encoded channel $x'_3 = \langle X-1 \rangle_{2^6+1}$.

$$
\begin{array}{rcccccccc}
N_2 & = & & 0 & 0 & 1 & 1 & 0 & 1 \\
\overline{N}_1 & = & & 1 & 0 & 1 & 1 & 0 & 1 \\
N_0 & = & & 0 & 1 & 1 & 0 & 0 & 1 \\
\hline
S & = & & 1 & 1 & 1 & 0 & 0 & 1 \\
C & = & 0 & 0 & 1 & 1 & 0 & 1 & \\
 & & & & & & & \rightarrow 1 & \text{EAIC}
\end{array}
$$

For $d=1$ the carry input of LSB full adder of the CPA adder is equal to 1, and the CPA gives:

$$
\begin{array}{rcccccccc}
S & = & & 1 & 1 & 1 & 0 & 0 & 1 \\
C & = & & 0 & 1 & 1 & 0 & 1 & 1 \\
d & = & & & & & & & 1 \\
\hline
A & = & \mathbf{1} & 0 & 1 & 0 & 1 & 1 & 0
\end{array}
$$

A carry out $c_{out} = a_6 = 1$ is generated. Since the 6-bit vector p is equal to

$$p = [1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1]$$

and the $sel = c_{out} \wedge p_6 = 1$ (bit p_6 is MSB), the output of the converter is given by:

$$x''_3 = x_3 = [0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0]$$

or in the decimal notation $x_3 = 20$, which can be verified as true.

For $d = 0$, the CPA gives:

$$\begin{array}{cccccccc} S & = & & 1 & 1 & 1 & 0 & 0 & 1 \\ C & = & & 0 & 1 & 1 & 0 & 1 & 1 \\ \hline A & = & \mathbf{1} & 0 & 1 & 0 & 1 & 0 & 0 \end{array}$$

A carry out $c_{out} = a_6 = 1$ is generated. Since the 6-bit vector p is equal to

$$p = [1 \quad 1 \quad 1 \quad 1 \quad 0 \quad 0]$$

and the $sel = c_{out} \wedge p_6 = 1$, the output of the converter is given by:

$$x_3'' = x_3' = [0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1]$$

or in the decimal notation $x_3' = 19$, which can be verified as true.

3 Hardware implementation and performance evaluation

In this section, the propagation delay and the amount of the hardware needed for implementation of proposed encoder on an ASIC and FPGA chip, along with the comparison to the encoders presented in [15] are given.

The encoder architecture shown in Fig. 3 is based on equations (3), (4) and (9).

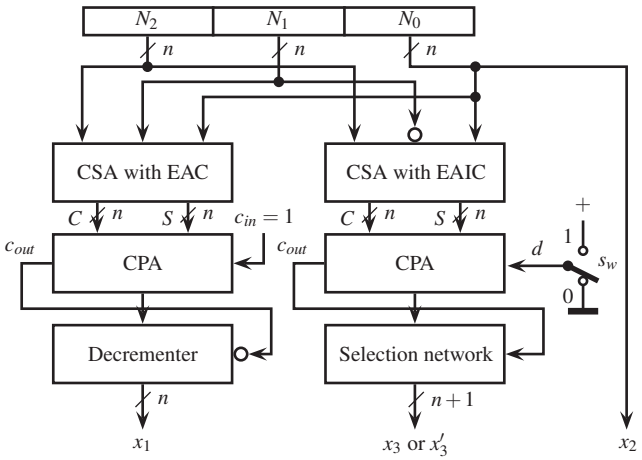

Fig. 3. The architecture of the new binary to residues encoder for moduli set $\{2^n - 1, 2^n, 2^n + 1\}$ with embedded diminished-1 channel

The carry save adder with the end around carry, which is an adder on whose inputs three n-bit operands are connected, followed by the decrementer is used for the modulo $2^n - 1$ channel.

By using the switch s_w the modulo 2^n+1 channel in the new encoder can yield either the modulo 2^n+1 residue (x_3) or the modulo 2^n+1 residue represented in diminished-1 number system (x'_3), as illustrated in Fig. 3. If the carry input of the CPA is equal to 1 (s_w in upper position) the output of modulo (2^n+1) channel is equal to x_3. On the other side, if the carry input of the CPA is equal to 0 (s_w in lower position) the output of the modulo 2^n+1 channel is equal to the modulo 2^n+1 residue in diminished-1 representation denoted as x'_3.

As is shown in Figure 3, the CPA is critical element in the both modulo (2^n-1) and modulo (2^n+1) residue generator data paths. The encoder performance can be increased if the CPA is replaced with parallel prefix adder. In the VLSI implementation, the parallel-prefix adders (or carry-tree adders) are known to have the best performance [23]. However, this performance advantage does not translate directly into an FPGA chip due to constraints on the logic block configurations and routing overhead. An FPGA chip, such as the Virtex-6, contains the number of slices, each containing a number of multiplexers, look-up tables, logic gates, flip-flops, etc. The parallel-prefix adder implementation on an FPGA chip is given in [24]. However, the implementations of the binary to residues encoder based on the parallel-prefix adders can lead to the higher hardware cost and consequently considerable the power dissipation in comparison to the carry propagate based architectures.

Since the residues are computed in parallel, the propagation delay of binary to residues encoder is $T=\max(T_1,T_3)$. That is, if $n<6$, $T=2n+10$, else $T=3n+4$. The area cost of binary to residues encoder is $A=A_1+A_3=35n-5$.

The presented algorithms were used for the description of proposed binary to residues encoder in the VHDL hardware programming language. Complete design was implemented on the Virtex 6 `XC6VCX75T` FPGA chip using Xilinx ISE Design Suite 14.2 while behavioral and post-route simulation of implemented encoder was performed using ISIM simulator.

The exact values of the area cost and propagation delay which relate to the ASIC and the FPGA-based implementations of the presented encoder architecture for different values of n, along with the comparison to architectures presented in [15], are shown in Tables 1 and 2.

Table 1. ASIC implementation - performance comparison

Design	**Delay**	**Area**
Encoder [15]	$3n$+5	$34n$
Encoder with D-1 channel [15]	$\max(2n+12,3n+5)$	$37n+7$
New encoder	$\max(2n+10,3n+4)$	$35n-5$

The encoder shown in Fig. 3 can perform the operation of the both encoders

presented in [15]. As can be seen from Table 1 there is no significant propagation delay improvement if the ASIC implementation is considered. However, the increase of performance regarding area consumption is significant in comparison to the encoder with the modulo 2^n+1 residue in the diminished-1 number system representation.

Table 2. FPGA implementation - Propagation delay [ns] / Area consumption [slices]

n	Encoder [15]	Encoder with D-1 channel [15]	New encoder
4	4.072 / 28	4.034 / 29	3.885 / 31
5	4.962 / 36	4.806 / 38	4.529 / 41
6	6.238 / 45	6.099 / 47	5.370 / 48
7	7.012 / 52	8.049 / 54	6.500 / 58
8	4.463 / 74	4.567 / 77	4.618 / 77
10	5.535 / 90	5.535 / 95	5.563 / 96

Functional simulation waveforms of the binary to residues encoder based on moduli set $\{63,64,65\}$ with diminished-1 encoded channel are shown in Fig. 4. The length of the input string X is 18 bits and it is sub-grouped into three groups of 6 bits. The output residues are 6 bits long for modulo 2^6-1 and modulo 2^6 channels, and 7 bits long for modulo 2^6+1 channel.

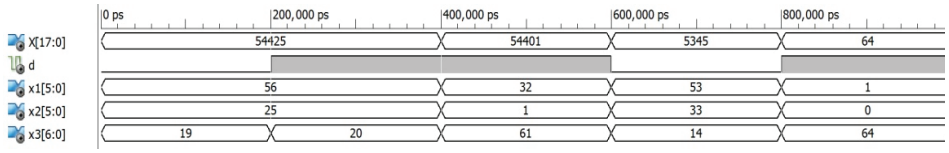

Fig. 4. Functional simulation waveforms of the binary-to-residue encoder based on moduli set $\{63,64,65\}$ with diminished-1 encoded channel.

4 Conclusion

In this paper, we investigated the binary to residues memoryless encoder, which is an important issue concerning the utilization of the RNS number system in DSP application. We proposed a new binary to residues encoder for moduli set $\{2^n-1,2^n,2^n+1\}$ with embedded diminished-1 channel which can be used instead of standard modulo 2^n+1 channel. The modulo 2^n+1 channel and the diminished-1 channel are implemented on the same hardware and the encoder can be used to perform either the modulo 2^n+1 or the diminished-1 channel operation. The single switch is used for the channel selection. Our approach avoids the initial

calculation of $X-1$ in order to compute the residue of X modulo 2^n+1 residue for the diminished-1 encoded channel and enables utilization of any modulo binary adder designs.

The speed of the binary to residues encoder can be increased by using pipelining. The encoder proposed in this paper is applicable to the ASIC and the FPGA implementations. The obtained results of the design in terms of the number of Xilinx FPGA logic elements and input-to-output propagation delays are given.

Acknowledgement

This work was supported by the Serbian Ministry of Science and Technological Development, Project No. 32009TR.

References

[1] H. L. Garner, "The residue number system," *IRE Trans. Electronic Computer*, vol. EC-8, no. Issue 2, pp. 140–147, Jun. 1959.

[2] N. Szabo and R. I. Tanaka, *Residue Arithmetic and its Application to Computer Technology*. New York: McGraw-Hill, 1967.

[3] E. Gholami, R. Farshidi, M. Hosseinzadeh, and K. Navi, "High speed residue number system comparison for the moduli set $\{2^n-1, 2^n, 2^n+1\}$," *Journal of Communication and Computer*, vol. 6, no. 3, pp. 40–46, 2009.

[4] J. Low and C.-H. Chang, "A new approach to the design of efficient residue generators for arbitrary moduli," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 60, no. 9, pp. 2366–2374, Sep. 2013.

[5] W. K. Jenkins and B. Leon, "The use of residue number systems in the design of finite impulse response digital filters," *IEEE Trans. on Circuits and Systems*, vol. CAS-24, no. 4, pp. 191–201, Apr. 1977.

[6] R. Chaves and L. Sousa, "RDSP: a RISC DSP based on residue number system," in *Digital System Design, 2003. Proceedings. Euromicro Symposium on*, Belek-Antalya, Turkey, Sep. 1–6, 2003, pp. 128–135.

[7] N. Stamenković, *Digital filter implementation using RNS-binary arithmetic*, monographies ed. LAP Lambert Academic Publishing, 2014.

[8] D. Živaljević, N. Stamenković, and V. Stojanović, "Digital filter implementation based on the RNS with diminished-1 encoded channel," in *Telecommunications and Signal Processing (TSP), 35th International Conference on*, Prague, Czech Republic, Jul. 3–4, 2012, pp. 662–666.

[9] N. Stamenković, D. Živaljević, and V. Stojanović, "The use of residue number system in the design of the optimal all-pole IIR digital filters," in *Telecommunications and Signal Processing (TSP), 36th International Conference on*, Rome, Italy, Jul. 2–4, 2013, pp. 722–726.

[10] D. Živaljević, N. Stamenković, and V. Stojanović, "FIR filter implementation based on the RNS with diminished-1 encoded channel," *International Journal of Advances in Telecommunications, Electrotechnics, Signals and Systems*, vol. 2, no. 2, pp. 56–62, 2013.

[11] K.-W. Kim and W.-J. Lee, "An efficient parallel systolic array for AB^2 over $GF(2^n)$," *IEICE Electronics Express*, vol. 10, no. 20, pp. 1–6, 2013.

[12] R. Capocelli and R. Giancarlo, "Efficient vlsi networks for converting an integer from binary system to residue number system and vice versa," *Circuits and Systems, IEEE Transactions on*, vol. 35, no. 11, pp. 1425–1430, 1988.

[13] F. Pourbigharaz and H. M. Yassine, "Modulo-free architecture for binary to residue transformation with respect to $\{2^m-1, 2^m, 2^m+1\}$ moduli set," in *Circuits and Systems, 1994. ISCAS '94., 1994 IEEE International Symposium on*, vol. 2, 1994, pp. 317–320.

[14] M.-H. Sheu, S.-H. Lin, Y.-T. Chen, and Y.-C. Chang, "High-speed and reduced-area RNS forward converter based on $(2^n-1, 2^n, 2^n+1)$ moduli set," in *Circuits and Systems, 2004. Proceedings. The 2004 IEEE Asia-Pacific Conference on*, vol. 2, 2004, pp. 821–824.

[15] I. Krstić, N. Stamenković, M. Petrović, and V. Stojanović, "Binary to RNS encoder with modulo 2^n+1 channel in diminished-1 number system," *International Journal of Computational Engineering & Management (IJCEM)*, vol. 17, no. 4, pp. 1–9, May 2014. [Online]. Available: www.ijcem.com

[16] L. M. Leibowitz, "A simplified binary arithmetic for the Fermat number transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-24, no. 5, pp. 356–359, Oct. 1976.

[17] H. Vergos and C. Efstathiou, "A unifying approach for weighted and diminished-1 modulo 2^n+1 addition," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 55, no. 10, pp. 1041–1045, Oct. 2008.

[18] C. Efstathiou, I. Voyiatzis, and N. Sklavos, "On the modulo 2^n+1 multiplication for diminished-1 operands," in *Signals, Circuits and Systems, 2008. SCS 2008. 2nd International Conference on*, Monastir, Tunisia, Nov. 7–9, 2008, pp. 1–5.

[19] E. Vassalos, D. Bakalis, and H. Vergos, "Reverse converters for RNSs with diminished-one encoded channels," in *EUROCON, 2013 IEEE*, Jul. 1–4, 2013, pp. 1798–1805.

[20] B. Vinnakota and V. V. B. Rao, "Fast conversion techniques for binary-residue number systems," *IEEE Trans. On Circuits And Systems-I: Fundamental Theories And Applications*, vol. 41, no. 12, pp. 927–929, Dec. 1994.

[21] S. J. Piestrak, "Design of residue generators and multioperand modular adders using carry-save adders," *IEEE Transactions on Computers*, vol. 423, no. 1, pp. 68–77, Jan. 1994.

[22] E. Vassalos, D. Bakalis, and H. Vergos, "On the design of modulo $2^n \pm 1$ subtractors and adders/subtractors," *Circuits Syst Signal Process*, vol. 30, no. 6, pp. 1445–1461, 2011.

[23] C. Efstathiou, H. Vergos, and D. Nikolos, "Fast parallel-prefix modulo 2^n+1 adders," *Computers, IEEE Transactions on*, vol. 53, no. 9, pp. 1211–1216, Sep. 2004.

[24] S. V. Padmajarani and M. Muralidhar, "A new approach to implement parallel prefix adders in an FPGA," *International Journal of Engineering Research and Applications (IJERA)*, vol. 2, no. 4, pp. 1524–1528, July-August 2012. [Online]. Available: www.ijera.com