ISSN (Online): 2349-7084
GLOBAL IMPACT FACTOR 0.238
I2OR PIF 2.125

INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 2, ISSUE 10, OCTOBER 2015, PP 674-678

# Implementation of a Fast Sign Detection Algoritm for the RNS Moduli Set $\{2^{N+1}-1, 2^N-1, 2^N\}$, N = 16, 64

**[1] GARNEPUDI SONY PRIYANKA, [2] K.V.K.V.L. PAVAN KUMAR**

*[1](M.Tech) DECS Branch, Department of ECE*
*[2] Asst.Professor, Department of ECE*
*Vignan's Nirula Iinstitute of Technology & Science for Women*
*Pedapalakaluru, Guntur, Andhra Pradesh, India*

**Abstract: -** In this dissertation a fast sign detection algorithm for the residue number system (RNS) moduli set $\{2^{N+1}-1, 2^N-1, 2^N\}$, N =16,64 is illustrated. This algorithm allows parallel implementation and completes the implementation with simply modulo 2n additions. This unit can be implemented using one carry save adder, one comparator, one carry generation unit and post processing unit. The results of this algorithm confirm that the area and delay are reduced compared to mixed radix conversion and Chinese reminder theroms. This algorithm is implemented using verilog language tool in Xilinx 13.2 version.

**Keywords:** residue number system (RNS), restricted moduli set, sign detection.

## I.Introduction

The RNS is a unique, non-weighted, carry free number system that provides parallel, high speed and fault tolerant arithmetic operations.The sign information is concealed in each residue digit in an RNS, so sign detection in an RNS is more difficult than that in the weighted number system, where the sign bit is the most significant bit (MSB).sign detection plays an important role in magnitude comparisons, overflow detection, and in branching operations.

There are quite a lot of researches for a sign detection in RNS moduli. A general theorem is derived by establishing necessary conditions for sign detection [1]. There are two categories based on algorithms. They are ROM based algorithms and specialized algorithms. In the ROM based algorithm one method detects sign for a preferred class of RNS is carried out as a sum modulo 2 of digits in the associated mixed radix system (MRS) [2], in [3] another method a sign detection procedure based on fractional illustration is proposed to lessen the sum modulo M in the conversion formula to a sum modulo 2. In the specialized algorithms one [4] detects the sign based on the new Chinese reminder theorem (CRT) II, the modulo operations are enclosed by size $\sqrt{M}$ , another one [5] uses the nth mixed radix conversion (MRC) for sign detection. A [6] combinational logic is implemented based on $\{2^{n-1}, 2^n, 2^{n+1}\}$ for sign detection in RNS, and it is not extended for other moduli sets. Recently the moduli set $\{2^{N+1}-1, 2^N-1, 2^N\}$ only having 2n and 2n-1terms used for sign detection because of its efficiency for modular operations (such as addition, subtraction and multiplication), and reverse conversion [7].

In this phenomenon, a sign detection for the moduli set $\{2^{N+1}-1, 2^N-1, 2^N\}$ is obtainable. This unit consists of a carry save adder (CSA), a comparator and a carry generation and post processing unit, and it only requires addition of the modulo 2n. This proposed method achieved better efficiency compared to the algorithms based on ROM technology [2], [3] and specialized algorithms [4], [5].

## II. PROPOSED SIGN DETECTION ALGORITHM

A standard RNS is completely distinct for positive integers in the range [0, M). An implicit signed number system may be considered to be split into a positive half and negative half of the range to hold negative integers. The dynamic range M of the moduli set {m1, m2,…., $m_{N-1}$, $m_N = 2^n$} is even. An implicit symbol of the

ISSN (Online): 2349-7084
INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 2, ISSUE 10, OCTOBER 2015, PP 674-678

actual result Y can be obtained in its range [-M/2, M/2-1) ,after conversion from residue number to the weighted number ensuing the noninteger X in the interval [0,M/2) as follows:

$$Y=\begin{cases}X, & if\ 0 \le X < M/2\\ X-m, & if\ M/2 \le X < M.\end{cases}$$

The mixed radix CRT obtainable in [8] is as follows.

For $\{m_1, m_2, \ldots, m_N\}$, the magnitude of a residue number $X=(x_1, x_2, \ldots, x_N)$ is as follows:

$$X=\sum_{j=1}^{N-2}\left(\alpha_{j+1}\prod_{i=1}^{j+1}m_i\right)+\alpha_1 m_1+\alpha_0$$

where $\alpha_{j+1}=\left|\sum_{i=1}^{j+2}\gamma i\ xi / \prod_{i=2}^{j+1} mi\right|_{m_{j+2}}$, $\alpha 1=|\gamma_1 x_1+\gamma_2 x_2|_{m_2}$, $\alpha_0=x_1$, $N>1$, $\gamma_1=(N_1|N_1^{-1}|m_1-1)/m_1$, $\gamma i=M|N_i^{-1}|mi/m_1 mi$, $M=m_1 m_2 \cdots m_N$, $N_i=M/m_i$, and the multiplicative inverse $|N_i^{-1}|_{m_i}$ is defined by $||N_i^{-1}|_{m_i} N_i|_{m_i}=1$, for $i=1, 2, 3, \ldots, N$. The floor function is denoted by $\lfloor\cdot\rfloor$.

It converts residue numbers to weighted numbers and it requires only modulo mi operations. The calculation procedure for each mixed radix $\alpha_j$ is independent of others, so the mixed radix coefficients can be computed in a fully parallel method. From this we can say that

For the moduli set $\{m_1, m_2 \ldots m_{N-1}, m_N = 2^n\}$, the value of $\alpha_{N-1}$ is equivalent to 2n-1 when the integer X is M/2.

i.e $\alpha_{N-1}(M/2) = 2^{n-1}$.

In the moduli set $\{m1, m2 \ldots m_{N-1}, m_N = 2^n\}$, for a residue representative number $(x_1, x_2 \ldots x_N)$, $\alpha_{N-1}$ is

$$\alpha_{N-1}=\left|\left\lfloor\frac{\gamma_1 x_1+\gamma_2 x_2+\cdots+\gamma_N x_N}{m_2 m_3 \cdots m_{N-1}}\right\rfloor\right|_{2^n}$$

Then the projected sign detection function is

Based on this the sign output is the MSB of $\alpha_{N-1}$. Therefore, this needs only one carry generation circuit to get nth bit of $\alpha_{N-1}$.

## III.SIGN DETECTION FOR THE MODULI SET $\{2^{N+1}-1, 2^N-1, 2^N\}$

A high efficiency sign detection entity for the moduli set $\{2_{N+1}-1, 2_N-1, 2_N\}$ is presented here. Based on the projected algorithm the sign detection unit is concurrent and suitable for VLSI implementation.

The sign detection of X = (x1, x2, x3) for the moduli set $\{2^{N+1}-1, 2^N-1, 2^N\}$ is

$$Sgn(x_1, x_2, x_3)=MSB\left(\left|\lfloor -2x_1+x_2+x_3+(x_2-x_1)/2^n-1\rfloor\right|_{2^n}\right).$$

In binary illustration, the words $x_1$, $x_2$, $x_3$ are n+1, n, n bit respectively. We symbolize $x_1$,n as the (n+1)th bit of x1, and denote $x'_1$ as the least n bits of x1.Because the word x1 is with one more bit than x2 and x3.

Take

$$W=1-\overline{W}=1+\left\lfloor\frac{x_2-x'_1-x_{1,n}}{2^n-1}\right\rfloor$$

$$=\begin{cases}0, & \text{if } x_{1,n}=0 \text{ and } x_2 < x'_1, \text{ or } x_{1,n}=1 \text{ and } x_2 \leqslant x'_1\\ 1, & \text{if } x_{1,n}=0 \text{ and } x_2 \geqslant x'_1, \text{ or } x_{1,n}=1 \text{ and } x_2 > x'_1\end{cases}$$

We denote, x''1 as a n-bit digit that equals 2x1,n-2:0+x1,n, which is concatenated by the least n-1 bits f x1 and x1,n. From this we can engrave that

$$\alpha_2=|-2x_1+x_2+x_3-x_{1,n}-\overline{w}|_{2^n}$$

$$=|-(2x_{1,n-2:0}+x_{1,n})+x_2+x_3-\overline{w}|_{2^n}$$

$$=|2^n-1-x''_1+x_2+x_3+1-\overline{w}|_{2^n}$$

$$=|\overline{x}''_1+x_2+x_3+w|_{2^n}$$

From the above equation we find MSB of $\alpha_2$ by the hardware realization using the figure (1). The circuit contains CSA [9], carry generation [10], comparator and post processing units.

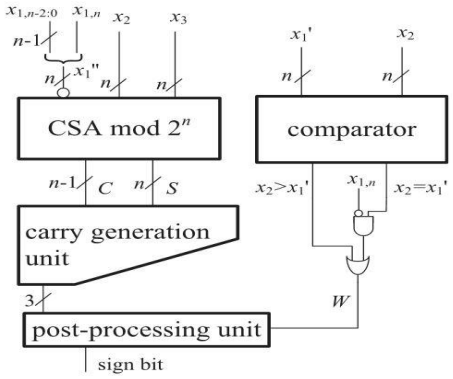



Fig 1: sign detection unit for the RNS moduli set $\{2^{N+1}-1, 2^N-1, 2^N\}$.

ISSN (Online): 2349-7084
INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 2, ISSUE 10, OCTOBER 2015, PP 674-678

The CSA mod $2^n$ is used to execute the sum of x''1+x2+x3 and the outputs obtained from this are two vectors sum (S) and carry (C) of n bit. Using carry save addition, the delay can be reduced further still. The thought is to take three inputs that we want to add together, x1 + x2+x3, and convert it into 2 outputs c + s such that x1 + x2+x3 = c + s.

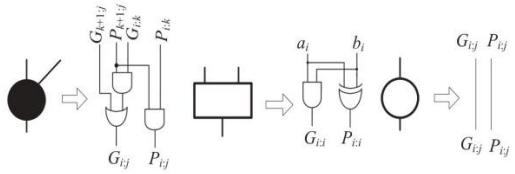


Fig 2: blocks used in carry generation unit and comparator unit.

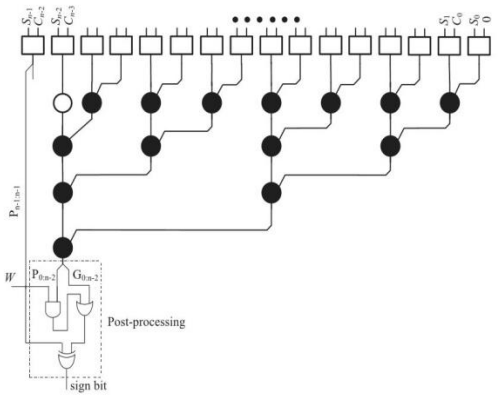



Fig 3: carry generation unit and post processing unit for n = 16, 64.

The carry generation unit and the post processing unit are used to generate nth bit of $\alpha$2 = C+S+W.

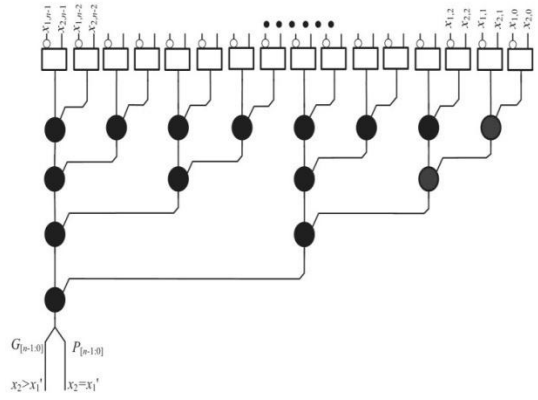


Fig 4: comparator unit for n = 16, 64.

The comparator unit is used to set up the comparison of x2 $>$ x'1 and x2 $=$ x'1.In the comparator unit a parallel implementation of the least significant bit first approach comparison algorithm [11] is used. This comparator unit and a carry generation unit are almost same apart from the addition with one input vector being set is ones compliment.

## IV. PERFORMANCE ESTIMATION

The act of the proposed sign detection unit of the moduli set $\{2^{N+1}-1, 2^N-1, 2^N\}$ is evaluated. The sign detection unit is compared with two units extended by two
Best sign detection algorithms to show the high efficiency of the new sign detection algorithm.
The outputs for the sign detection of the RNS moduli set $\{2^{N+1}-1, 2^N-1, 2^N\}$ is as follows:

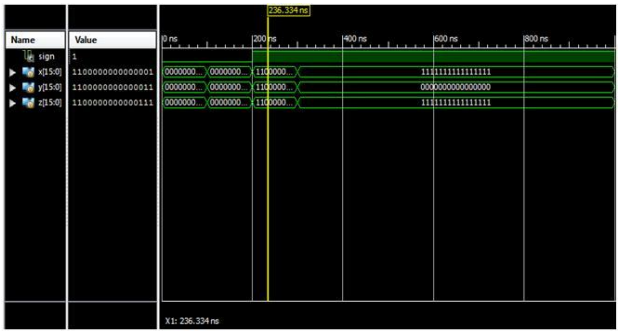



Fig 5. Output for the RNS moduli set $\{2^{N+1}-1, 2^N-1, 2^N\}$ where N=16.

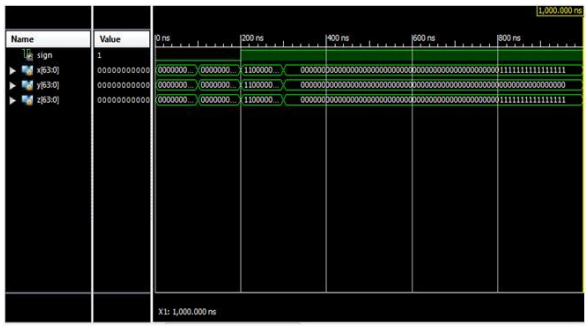
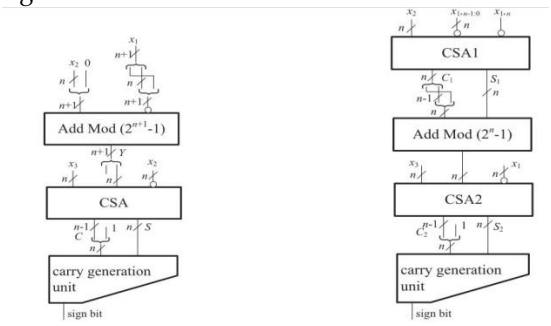



Fig 6. Output for the RNS moduli set $\{2^{N+1}-1, 2^N-1, 2^N\}$ where N=64.

From the result we can say that for the inputs x=0000000000001111, y=0000000000001111, z=0000000000001111, the sign bit is 0, and for the inputs x= 0000111111111111, y=1111111111111111, z=1111111111111111, the sign bit is 1. The results with N=64 bit also verified.
The comparison is done by comparing the proposed

ISSN (Online): 2349-7084
INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 2, ISSUE 10, OCTOBER 2015, PP 674-678

method with the two best existing methods. They are figured below:




(a) (b)

Fig 7. (a) Sign detection unit for the RNS moduli set $\{2^{N+1}-1, 2^N-1, 2^N\}$ based on [4].
(b) Sign detection unit for the RNS moduli set $\{2^{N+1}-1, 2^N-1, 2^N\}$ based on [5].

The experimental results shows that the area used for the proposed algorithm is compared to the specialized algorithms [4] and [5].

| Device Utilization Summary | | | | |
|---|---|---|---|---|
| **Logic Utilization** | **Used** | **Available** | **Utilization** | **Note(s)** |
| Number of 4 input LUTs | 84 | 7,168 | 1% | |
| **Logic Distribution** | | | | |
| Number of occupied Slices | 46 | 3,584 | 1% | |
| Number of Slices containing only related logic | 46 | 46 | 100% | |
| Number of Slices containing unrelated logic | 0 | 46 | 0% | |
| **Total Number of 4 input LUTs** | 84 | 7,168 | 1% | |
| Number of bonded IOBs | 50 | 141 | 35% | |
| **Total equivalent gate count for design** | 552 | | | |
| Additional JTAG gate count for IOBs | 2,400 | | | |

Fig 8: devise utilization summary for the specialized algorithm [4].

In this algorithm the blocks used are one add mod $(2^{n+1}-1)$, one CSA, carry generation unit. The total number of gates used for this algorithm is 552, the delay occurred is 22.450ns having levels of logic 14.

| Device Utilization Summary | | | | |
|---|---|---|---|---|
| **Logic Utilization** | **Used** | **Available** | **Utilization** | **Note(s)** |
| Number of 4 input LUTs | 47 | 7,168 | 1% | |
| **Logic Distribution** | | | | |
| Number of occupied Slices | 27 | 3,584 | 1% | |
| Number of Slices containing only related logic | 27 | 27 | 100% | |
| Number of Slices containing unrelated logic | 0 | 27 | 0% | |
| **Total Number of 4 input LUTs** | 47 | 7,168 | 1% | |
| Number of bonded IOBs | 50 | 141 | 35% | |
| **Total equivalent gate count for design** | 297 | | | |
| Additional JTAG gate count for IOBs | 2,400 | | | |

Fig 9: devise utilization summary for the specialized algorithm [5].

In this algorithm the blocks used are one add mod $(2^n-1)$, two CSA, carry generation unit. The total number of gates used for this algorithm is 297, the delay occurred is 19.836ns having levels of logic 12.

| Device Utilization Summary | | | | |
|---|---|---|---|---|
| **Logic Utilization** | **Used** | **Available** | **Utilization** | **Note(s)** |
| Number of 4 input LUTs | 14 | 7,168 | 1% | |
| **Logic Distribution** | | | | |
| Number of occupied Slices | 10 | 3,584 | 1% | |
| Number of Slices containing only related logic | 10 | 10 | 100% | |
| Number of Slices containing unrelated logic | 0 | 10 | 0% | |
| **Total Number of 4 input LUTs** | 14 | 7,168 | 1% | |
| Number of bonded IOBs | 36 | 141 | 25% | |
| **Total equivalent gate count for design** | 87 | | | |
| Additional JTAG gate count for IOBs | 1,728 | | | |

Fig 10: devise utilization summary for the proposed algorithm.

In this algorithm the blocks used are one CSA, one comparator unit, one carry generation post processing unit and one. The total number of gates used for this algorithm is 87; the delay occurred is 16.016ns having levels of logic 8.

## V. CONCLUSION

A fast sign detection algorithm for the restricted moduli set $\{2^{N+1}-1, 2^N-1, 2^N\}$ is proposed with the modulo 2n. The proposed algorithm contains only modulo 2n additions and allows for parallel realization. The proposed algorithm is the first proposed for the moduli set $\{2^{N+1}-1, 2^N-1, 2^N\}$. The experimental results show that the proposed circuit achieves considerable improvements in terms of area and delay.

## VI. REFERENCES

[1] N. Szabo, "Sign detection in nonredundant residue systems," IRE Trans. Electron. Comput., vol. EC-11, no. 4,pp. 494-500, Aug. 1962.
[2] Z, UIman, "Sign detection and implicit-explicit conversion of numbers in residue arithmetic," IEEE Trans. Comput., vol. 32, no. 6, pp.590-594, Jun. 1983.
[3] T. V. Vu, Efficient implementations of the Chinese remainder thermo for sign detection and residue decoding," IEEE Trans. Comput., vol. 34, no. 7, pp. 646-651, Jul. 1985.
[4] E.AI-Radadi and P.Siy, "RNS sign detector based on Chinese remainder theorem II (CRT II)." Comput., Math. Appl., vol. 46, nos. 10-11, pp. 1559-1570, 2003.
[5] M. Akkal and P. siy, "Optimum RNS sign detection algorithm using MRC-II with special moduli set," J. Syst. Arch., vol. 54, no. 10, pp. 911-918, Oct. 2008.
[6] T. Tomczak, "Fast sign detection for RNS{2n-1,2n,2n+1}," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 55, no. 6, pp. 1502-1511, Jul. 2008.
[7] P. Mohan, "RNS-to-binary converter for a new three-moduli set {2n+1-1, 2n, 2n-1}," IEEE Trans. Cir-

cuits Syst. II, Exp. Briefs, vol. 54, no. 9, pp. 775-779, Sep. 2007.
[8] S. Bi and W. Gross, "The mixed-radix Chinese remainder theorem and its applications to residue comparison," IEEE Trans. Comput. Vol. 57, no. 12, pp. 1624-1632, Dec. 2008.
[9] S. Piestrak, "Design of residue generators and multioperand modular adders using carry-save adders," IEEE Trans. Comput., vol. 43, no. 1, pp. 68-77, Jan. 1994.
[10] R. Zimmerman, "Efficient VLSI implementation of modulo (2n ± 1) addition and multiplication," in Proc. 14th IEEE Symp. Comput. Arithmetic, 1999, pp. 158-167.
[11] K. Furuya, "Design methodologies of comparators based on parallel hardware algorithms," in Proc. 10th ISCIT, Oct. 2010, pp. 591-596.