ISSN: 2319-8753

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 3 , Issue 4 , April 2014

Performance improvement in FIR filter using Residue Number System with modulo adders and multipliers

Mythili.M[1], Gowrishankar.V[2], Venkatachalam.K.V[3]

PG Student, Department of ECE, Velalar College of Engineering and Technology, Erode, Tamilnadu[1]

Assistant Professor, Department of ECE, Velalar College of Engineering and Technology, Erode, Tamilnadu[2]

Professor, Department of ECE, Velalar College of Engineering and Technology, Erode, Tamilnadu[3]

***ABSTRACT*-Digital Signal Processing (DSP) systems are the core of wide range of applications like audio, video, image processing and consumer electronics.Most of the DSPs involve repetitive operations of addition, subtraction and multiplication on large integers.A digital Finite Impulse Response (FIR) filter performs the frequency shaping or linear prediction on a discrete-time input sequence {x_0, x_1, $x_{2....}$}. In this work we focused on the design of an efficient VLSI (Very Large Scale integration) architecture for FIR filters which aims at reducing the power consumption, increasing the speed and also to reduce the hardware complexity using Residue Number System (RNS). This enables simultaneous parallel processing on all the digits resulting in high speed addition and multiplication in RNS domain. It uses modulo adders and modulo multipliers to obtain high speed performance.**

***KEYWORDS*: Finite Impulse Response (FIR), Residue Number System (RNS), modulo adders, modulo multipliers, moduli set.**

I.INTRODUCTION

The aim described in [1] was the comparison of RNS-FIR filter with TCS-FIR filter in terms of delay, area and power dissipation. And the work in [1] took into account the dynamic power dissipation, which was the dominant portion of the energy consumed a few years ago. The static powerdissipation is also an important role in today's powerbudgets due to the technology scaling, and the increased transistor'sleakage due to sub-threshold current. In [2] the static and dynamic power dissipation of RNS filter was reduced compared to TCS filter, but the speed was not improved.

RNS implementation is faster than its TCS counterpart because the computations are accomplished in short word - length modulo channels without carry propagation between channels. As the carry chain that slows down Very Long Word-Length (VLWL) (hundred bits or more) addition and multiplication is effectively broken in RNS, the problem in implementing VLWL arithmetic on platforms with limited space and constrained battery specification, such as smart cards and Radio Frequency Identification (RFID) tags, is the hardware required for the conversion between TCS and RNS as well as the simultaneous computation in severalmodulo channels. As RNS is well suited for applications involving repetitive computations like repeated modulo multiplications in cryptographic algorithm and multiply-add operations in signal processing algorithm, the research emphasis has moved obviously in recent years to the area-power efficient implementation of concurrent modulo arithmetic operations in RNS.

The techniques such as multi-modulus and multi-function architectures to reduce the hardware redundancy as well as multi - threshold voltage and multi-supply voltage designs to lower the power dissipation have been proposed in [2]–[4]. Such control techniques are intended for algorithm level design space exploration and are applicable to generic modulo arithmetic architectures. For structural level simplification of specific modulo arithmetic operations like modulo multiplication, and modulo addition techniques that explore unique number theoretic properties of special moduli of the forms 2^n, 2^n+1, 2^n-1 have received wide spread attention amongst others[5]-[8].

II. BACKGROUND

Residue Number System is widely used in the implementation of application specific Digital Signal Processing (DSP) systems. Since the system has gained a remarkable importance in recent years because of the lower power consumption over their two's complement counterparts.

A. *ResidueNumberSystem(RNS)*

RNS is defined by a set of relatively prime integer. The RNS implementations can speedup addition, subtraction and multiplication. The data represented by RNS is processed in parallel with no dependence or carry propagation between the processing units. The process of converting the input data into RNS representation is called *Forward Conversion*, and the process of converting back the output data from RNS to conventional representation is called *Reverse Conversion*.A general structure of a typical RNS processor is shown in Fig 1.

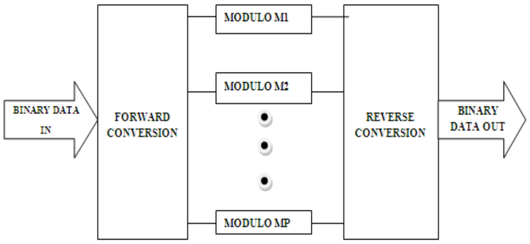


Fig.1 General block diagram of RNS processor

In RNS, a decimal number is represented by an n-tuple of its remainders with respect to each modulus in the moduli set. To illustrate an RNS number, let us consider X to be a decimal number and set {m0, m1, m2…m(n-1)} to be the moduli set for a residue number system. This RNS can be the moduli set for a residue number system. This RNS can represent any number from 0 to (M-1), where M is the product of all the moduli in the set. Number X in this system will be represented as

$$<X>_{mi} = X \bmod mi \qquad (1)$$

The operations of addition and multiplication in RNS, are done in parallel on the moduli

$$z = X\ op\ Y \rightarrow \begin{cases} Zm1 = <Xm1\ op\ Ym1> m1 \\ Zm2 = <Xm2\ op\ Ym2> m2 \\ \dots\dots\dots\dots. \\ Zmp = <Xmp\ op\ Ymp> mp \end{cases}$$

(2)

The two methods of converting residue to binary are Chinese Remainder Theorem (CRT) and Mixed Radix Conversion (MRC) method. The ultimate goal is to minimize the hardware needed for implementation while maximizing the speed of the conversion. The proposed method exploits the properties of RNS and provides parallel access. Hence the speed of the proposed method increases.Investigating new conversion schemes can lead to overcoming some obstacles in the RNS implementation of different applications.

B.ImplementationofFIRFiltersinRNS

A FIR filter of order N is described by the expression

$$y(n) = \textstyle\sum_{k=1}^{N} ak.x(n-k) \qquad (3)$$

It can be realized in RNS as

$$y(n) = \textstyle\sum_{k=1}^{N} ak.x(n-k) \rightarrow$$

$$\begin{cases} Ym1(n) = <\sum_{k=1}^{N} <Am1(k).Xm1(n-k)> m1 > m1 > \\ Ym2(n) = <\sum_{k=1}^{N} <Am2(k).Xm1(n-k)> m2 > m2 > \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots \\ Ymp(n) = <\sum_{k=1}^{N} <Amp(k).Xmp(n-k)> mp > mp > \end{cases}$$

(4)

This FIR filter can be implemented in RNS by decomposing it into p FIR filters working in parallel, as sketched in Figure 1. A key point in the design of the RNS filter is the selection of moduli. To select the set of co-prime numbers which cover the dynamic range of 20 bits, we used the tool described in [2], which selects the set of moduli giving the best delay/area/power tradeoffs according to the results of the characterization of the RNS filter composing blocks. In each tap, a modular multiplier is needed.

C.Modulo addition and modulo multiplication

Modulo addition and multiplication can be done by using special moduli set $\{2^n, 2^n+1, 2^n-1\}$.These sets of forms are preferred over the generic moduli due to the ease of hardware implementation of modulo arithmetic functions as well as system level inter-modulo operations, such as RNS to binary conversions and signed detections.

III. PROPOSED MODULO ADDER AND MODULO MULTIPLIER BASED RNS-FIR FILTER

Modular adder is one of the key components for residue number system (RNS) it can offer an excellent balance among the RNS channels for multi channel RNS processing. These adders are adopted to eliminate the recompilation of carries. It also offers better performance in delay and area.

A. *Modulus 2^n-1addition*

As *2^n-1*is congruent to zero modulo, zero can be represented by an n-bit binary string of all zeros or all ones in modulo *2^n-1*arithmetic [10]. Thus, a modulo *2^n-1*addition oftwo operands, A and B is equivalent to an n-bit addition of A, B and Cout, i.e.,

$$|A + B|2n - 1 = \begin{cases} |A + B|2n & \text{IF } A + B < 2n \\ |A + B + 1|2n & \text{IF } A + B \geq 2n \end{cases}$$

$$= |A + B + Cout|2n \qquad (5)$$

Where Cout is the carry output resulting from the addition of two inputs A and B. As Cout is added to the sum of A and B at the lsb position, modulo *2^n-1* addition is generally referred to as end-around-carry (EAC) addition.Fig. 2 represents the block diagram of modulo 2^n-1adder.

A. *Modulus 2^n+1addition*

A modulo 2^n+1addition of two diminished-1 represented operands, A and B, is equivalent to an n-bit addition of A and B with $\overline{Cout}$, i.e.,

$$|S+1|2^n+1 = |A+B+1|2^n+1$$
$$= |A+B+\overline{Cout}|2^n+1 \qquad (6)$$

where$\overline{Cout}$is the carry output resulting from the addition of two inputs A and B. As $\overline{Cout}$ is added to

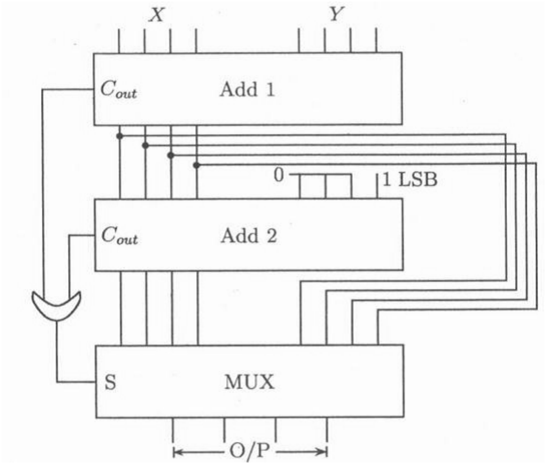
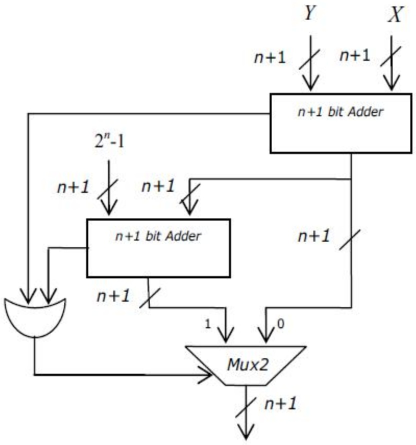


Fig.2 2^n-1 modulo adder

Where $\overline{Cout}$is the carry output resulting from the addition of two inputs A and B. As $\overline{Cout}$ is added to the sum of A and B at the lsb position, modulo 2^n+1addition is generally referred to as complementary end-around-carry(CEAC) addition.

The block diagram of 2^n+1 modulo adder can be represented in Fig.3.Arithmetic modulo 2^n+1[9]-[10] has found applicability in avariety of fields ranging from pseudorandom numbergeneration and cryptography up to convolution computations without round-off error.



.

Fig.3 2^n+1 modulo adder

Modulo 2^n+1operator are also commonly included in residue number system (RNS) applications.

C.Modulus2^n-1multiplication

For multiplication various ROM based solutions using table lookup have been proposed and compared. Sophisticated methods exist to reduce the table sizes by combining smaller table lookups with simple arithmetic operations, such as additions. For word lengths larger than eight bits, however, these solutions still require prohibitively large ROMs or many clock cycles for evaluation.

For high performance modulo multiplication, keen multipliers are necessary, that can be implemented as combinational or pipelined circuits. Solutions based ordinary integer multiplication with subsequent modulo correction using adders are proposed. A modulo (2^n-1) multiplier Architecture can be represented in Fig. 4.

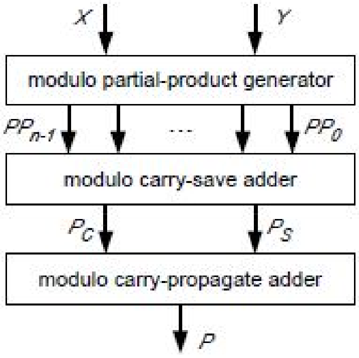

X
Y
modulo partial-product generator
PP_{n-1}
...
PP_0
modulo carry-save adder
P_C
P_S
modulo carry-propagate adder
P

Fig.4 2^n-1 modulo multiplier

Modulo multiplication can be formulated as

$$X.Y \bmod (2^n-1)$$
$$= (X.Y \bmod 2^n + X.Y \operatorname{div} 2^n)\bmod(2^n-1) \qquad (7)$$

where X.Y mod 2^ncorresponds to the low output word and X.Y div 2^n high output word of the multiplication. Therefore, modulo (2^n-1) multiplication can be accomplished by an n-bit unsigned multiplication followed by an n-bit modulo (2^n-1) addition.

D.Modulus2^n+1multiplication

Modulo (2^n+1) multiplication is considered here for the application of IDEA cipher. That is, n-bit numbers in normal representation are used for operands and result, where the value 0 is not used and the value 2^n is represented by "00….0". The presented algorithm can easily be suited for number representations with the value 0 included and the value 2^nindicated by a separate bit.

The normal number representation of modulo (2^n+1) multiplication can be formulated as

$$X.Y \bmod (2^n+1)$$
$$= (X.Y \bmod 2^n - X.Y \operatorname{div} 2^n)\bmod(2^n+1) \qquad (8)$$

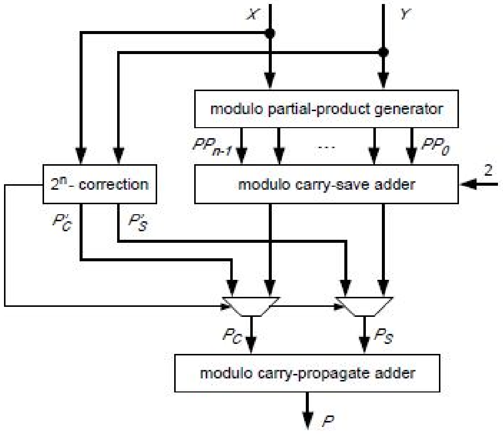


Fig.5 2^n+1 modulo multiplier

As well to modulo (2^n-1) multiplication, an n-bit unsigned multiplication followed by an n-bitmodulo (2^n+1) subtraction can be performed [3]. Again, the multiplication can be obtained by performing partial product generation and carry save addition modulo (2^n+1).

E.Implementation of FIR filter by using modulo adders and modulo multipliers

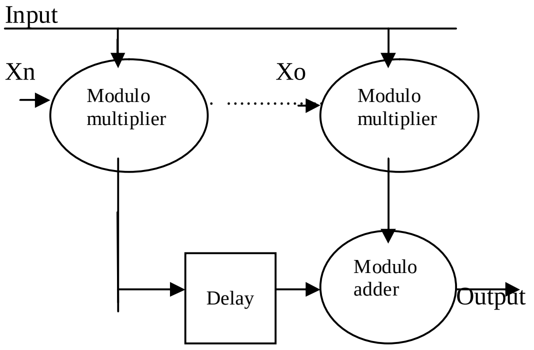


Fig.6 design of FIR filter

A main point in the design of the RNS filter is the choice of moduli. The special moduli set is considered here in the form of $\{2^n, 2^n+1, 2^n-1\}$.

FACTORS	128 TAP TCS FIR Filter	128 TAP RNS FIR Filter
Speed	350MHZ	500MHZ
Power dissipation	6190.2µW	5687.45µW

Power dissipation in µW (100MHz)

TABLE I
Results for filter implementation

The uses of modulo adder and modulo multiplier in each tapgiving the best delay/area/power tradeoffs according to the results of the characterization of the RNS filter composing blocks. Adder and subtractor based forward and reverse converters are used in the design of FIR filter. The output of forward converter is given as the input to modulo adder/multiplier based RNS-FIR. The residue output from RNS-FIR filter is applied to reverse converter, to achieve high speed of the filter output.

IV. RESULTS AND ANALYSIS

In Table I the comparison is carried out on filters executed in the 90 nm STM library of standard cells (VDD = 1:0 V , at 25 C) [3], and the power dissipation has been calculated by Synopsys Power Analyzer based on the annotated switching activity of random generated inputs. All the filters can be clocked atfmax = 500 MHz.The power dissipation is computed at a clock frequency of 100 MHz. The speed of the FIR filter is improved than the traditional TCS-FIR. Table II represents the comparison of area saving between two modulo multipliers.

The speed performance of residue number system based FIR filter is compared with the traditional Two's complement system based FIR filter. Assuredly the performance improvement factor speed is improved much better than the conventional system.

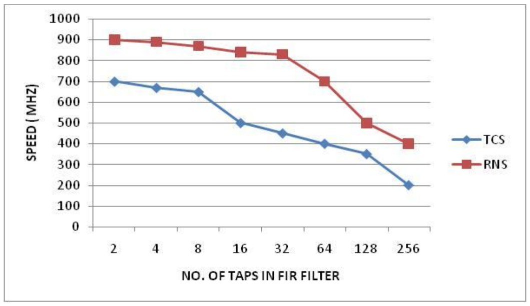


Fig.7 speed performance of RNS vs. TCS FIRFilter

n	*AREA(µm²)*		*%A REA SAVING*	
	TYPE A	*B*	*TYPE A*	*TYPE B*
32	139259	149331	8.5	7.4
36	180436	187289	3.6	4.2
40	212579	229827	6.8	7.3
48	298750	328258	7.9	10.2
52	339335	384066	9.8	8.5
64	496212	577640	14.6	13.8

TABLE II
Comparison of area saving between multipliers

V. CONCLUSION

One of the most widely used operations in DSP is Finite Impulse Response (FIR) filtering. Thus the design of compact and high speed real-time digital filters will be necessary to find applications in radar, communications and image processing systems.So the performance improvement in FIR filter using Residue Number System with modulo adders and modulo multipliers will be designed. The implementation of the proposed FIR filter has lesser delay than the conventional method. Due to the decrement of delay the speed of the system will be increased automatically.The implementation results show that the RNS filters offer a reduction offer a total power reduction of 35% with respect to the corresponding conventional filters.

ACKNOWLEDGMENT

The authors acknowledge the contributions of the students, faculty of VelalarCollege of Engineering and Technology for helping in the design of check circuitry, and for tool support. The authors also thank theanonymousreviewers for their thoughtful commentsthat helped to improve this paper**.**The authors would like to thank the anonymous reviewers for their constructive critique from which this paper greatly benefited.

REFERENCES

[1]. Nannarelli A. and Cardarilli G.C. (May 2001) ‘Tradeoffs between Residue Number System and Traditional FIR Filters’ *Proc.of* IEEE International Symposium on Circuits and Systems, vol. II, pp. 305–308.

[2]. G. C. Cardarilli, A. D. Re, A. Nannarelli, and M. Re, “Low power and low leakage implementation of RNS FIR filters,” in *Proc. 39thAsilomar Conf. Signals, Syst. Comput.*, Pacific Grove, CA, Nov. 2005, pp. 1620–1624.

[3]. V. Paliouras and T. Stouraitis, “Multifunction architectures for RNS processors,” *IEEE Trans. Circuits Syst. II, Analog. Digit. Signal Process.* vol. 46, no. 8, pp. 1041–1054, Aug. 1999.

[4]. I. Kouretas and V. Paliouras, “RNS multi-voltage low power multiply add unit,” in *Proc. 17th IEEE Int. Conf. Electronics, Circuits Systems*, Athens, Greece, Dec. 2010, pp. 9–12.

[5]. Z. Wang, G. A. Jullien, and W. C. Miller, “An algorithm for multiplicationmodulo (2^n-1),” in *Proc. 39th IEEE Midwest Symp. CircuitsSyst.*, Ames, IA, Aug. 1996, pp. 1301–1304.

[6] R. Zimmermann, “Efficient VLSI implementation of modulo (2^n±1) addition and multiplication,” in *Proc. 14th IEEE Symp. ComputerArithmetic*, Adelaide, Australia, Apr. 1999, pp. 158–167.

[7] C. Efstathiou, H. T. Vergos, and D. Nikolos, “Modified Booth modulo (2^n-1) multipliers,” *IEEE Trans. Comput.*, vol. 53, no. 3, pp. 370–374, Mar. 2004.

[8] Z.Wang, G. A. Jullien, andW. C.Miller, “An efficient tree architecture for modulo (2^n-1) multiplication,” *J. VLSI Signal Process.*, vol. 14, no. 3, pp. 241–248, Dec. 1996.

[9] C.Efstathiou, H. T. Vergos, G. Dimitrakopoulos, and D. Nikolos, “Efficientdiminshed-1 modulo (2^n+1) multipliers,” *IEEE Trans. Comput.*, vol. 54, no. 4, pp. 491–496, Apr. 2005.

[10] Ramyamuralidharan (2012) ‘Area-power efficient modulo 2n-1 and modulo 2n+1 multipliers for {2n-1, 2n, 2n+1} based RNS’ IEEE transactions on circuits and systems-1, VOL.59, no.10.

BIOGRAPHY

Mythili.Mhas received B.E degree in ECE from Anna University, Coimbatore 2011. She is currently pursuing ME VLSI Design in Velalar College of Engineering and Technology under Anna University, Chennai. Her areas of interest in research are VLSI Signal Processing, VLSI architectures.

GowriShankar.Vhas received B.E degree from the IRTT, Erode District in 2008 and M.E in Kongu Engineering College, Perundurai in the year 2010. He is working as an Asst.Professorin the department of ECE, Velalar College of Engineering and Technology, Erode. Currently pursuing Ph.DinInformation and communication engineering under Anna University, chennai. He has published papers in various international and national journals.