

Area and power efficient fir filter Based on residue number system

¹Chippy Elsa Thomas and ²Mrs. R.P Meenaakshi Sundhari

¹M.E VLSI, Department of ECE, Sasurie College of Engineering, Vijayamangalam

²Head of the Department, Department of ECE, Sasurie College of Engineering, Vijayamangalam

ABSTRACT

In digital systems, filter occupies a major role. This paper describes the design of FIR filter based on residual number system using modified version of booth multiplier which requires minimum number of multipliers and adders. The modified booth multiplier uses modified booth algorithm or modified booth encoder. The modified booth encoder is used to avoid variable size partial arrays. In FIR filters multiplier consumes more power. To reduce power the number of partial products in the multiplier is to be reduced. One of the solutions of realizing high speed multipliers is to enhance parallelism, which helps to decrease the number of subsequent stages. The RNS is carry free hence the booth recoding method does not propagate the carry to the next stages. Finally the proposed FIR filter based on RNS structures are beneficial in terms of area and power when compared to the existing FIR based on RNS using Wallace multiplier.

1.INTRODUCTION

Digital filters are essential in DSP applications. Finite Impulse Response (FIR) filters and Infinite Impulse Response (IIR) filters are the two basic filters. Both filter implementations includes binary-to-residue converter at its input, which converts the input data into equivalent residues. In this paper architecture for the FIR filters implementation is proposed.

2.FIR FILTER

FIR filters are digital filters with finite impulse response and also called as non-recursive digital filters. The filters with any arbitrary magnitude

response can be tackled using fir response. The basic FIR filter is represented by the following equation

$$Y(n) = \sum_{k=0}^{N-1} h(k)x(n-k)$$

$h(n)$ and $x(n)$ are finite duration sequences, hence their convolution is also in finite duration. The duration of $y(n)$ is $L+M-1$.

3.RESIDUE NUMBER SYSTEM

A residue number system is defined by a set of N integer constants,

$$\{m_1, m_2, m_3, \dots, m_N\},$$

Is referred to as the moduli. Let M be the least common multiple of all the m_i .

Any arbitrary integer X smaller than M can be represented in the defined residue number system as a set of N smaller integers

$$\{X_1, x_2, x_3, \dots, x_N\}$$

With $x_i = X \text{ mod } m_i$; Represents the residue class of X to that modulus m_i . Note that for maximum representational efficiency it is imperative that all the moduli are coprime; that is, no modulus may have a common factor with any other. M is then the product of all the m_i .

The Residue Number System (RNS) is an alternative arithmetic approach which supports operations performed in parallel independent channels. The residue number system is a non-weighted number system which speeds up arithmetic operations by dividing them into smaller parallel operations.

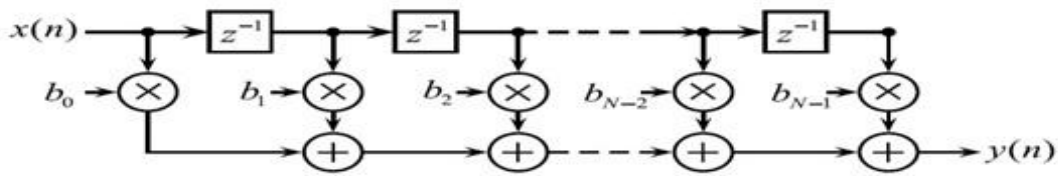


Figure 1: Basic Structure of FIR Filter

A FIR tap is a coefficient or delay pair in which the number of FIR taps is the amount of memory required to complement the filter and the number of calculations. The implementation of the Nth-order tapped delay-line FIR filter, using an RNS with base $B = \{m_1, m_2, \dots, m_n\}$. It presents the organization of one modulo channel, out of the N required for the complete filter implementation. The modulo 2^n is performed as conventional arithmetic operation by discarding the higher order bits positioned after the bit position n . The modulo $2^n + 1$ are much more complex, and involve addition of correction factors and carry save addition involves end-around carries.

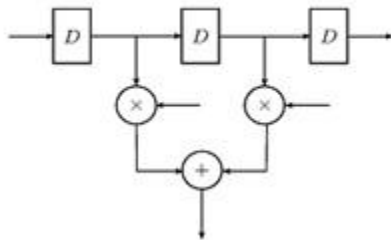


Figure 2: Modulo-mi two tap filter

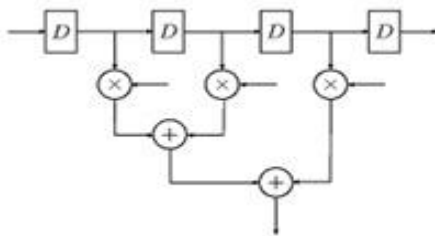
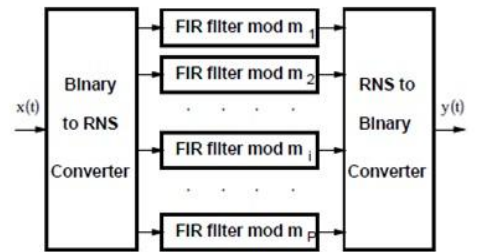


Figure 3: Modulo-Mi Three Tap Filter

In this paper a two and three tapped delay line based fir filters are designed based on residue number system as the proposed system. The existing system is the binary multiplier. In the filter the basic moduli set $\{2^n, 2^n+1, 2^n-1\}$ is used by the Wallace tree multiplier. The RNS based fir filter and the fir filter based on binary multiplier are compared. The proposed method reduces the



power.

Figure 4: RNS based FIR filter

5. MODULAR MULTIPLICATION

The modular multiplication is normally very complicated. In RNS the modulo set $\{2^n, 2^n-1, 2^n+1\}$ is used in the fir filter design. The modulo multiplication offers simple and efficient implementation there by reduces the cost and high speed data rates. Here the basic moduli set multiplication is based on the Wallace tree. The moduli 2^n-1 plays an essential role in RNS application in fault tolerant computer system, in error detection in computer systems and in floating point arithmetic. The modulus 2^n+1 is complex and bottleneck for RNS arithmetic unit. It has long delay than other modulus. For the moduli 2^n+1 the $n+1$ bit is wide and modulo operation is difficult.

Figure 6: Wallace tree 1

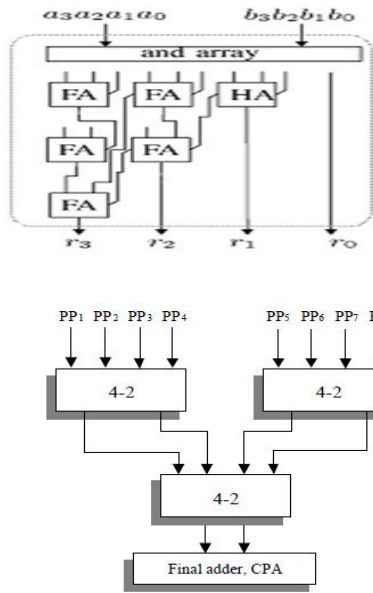


Figure 5: modulo 2^4 multiplier

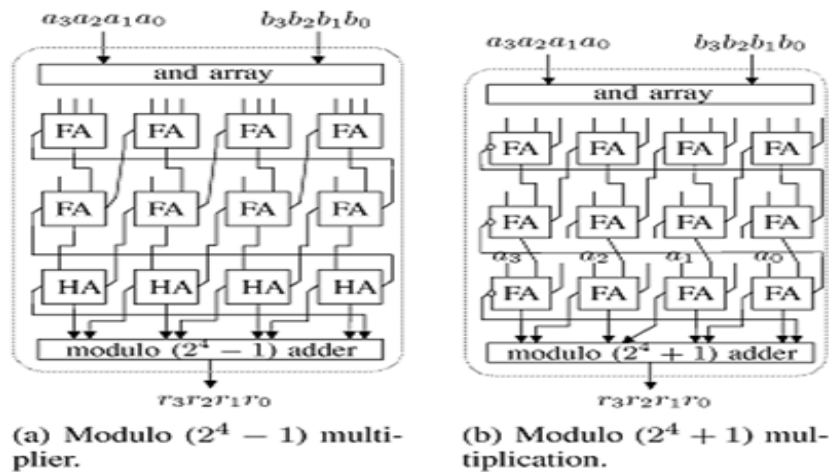


Figure 7: modulo $2^4 \pm 1$ multiplier

Due to the complexity and the error computation in the modulo 2^n+1 operation, it can be replaced by the $2^{n-1}-1, 2^{n+1}-1, 2^{2n+1}-1, 2^{2n+1}+1$ and $2^{2n}+1$. Many different modulo set have been suggested, such as $\{2^n-1, 2^n, 2^n+1\}$, $\{2^n, 2^n-1, 2^{n-1}-1\}$, $\{2^n, 2n-1, 2n+1-1\}$, $\{2^{2n}+1, 2^n+1, 2^n-1\}$, $\{2^n, 2^{2n}-1, 2^{2n}+1\}$, $\{2^{2n}, 2n+1+1, 2^{n+1}-1\}$. In this paper we have used three moduli set $\{2^n-1, 2^n, 2^n+1\}$. Nevertheless, it has disadvantaged that residue (2^n+1) requires $(n+1)$ bits to represent 2^n+1 state, which means that almost half of the states remain unused.

6. WALLACE TREE

A Wallace tree is an efficient hardware implementation of a digital circuit that multiplies two integers. It is efficient in terms of power and regularity without significant increase in delay and

area. The partial products are generated using AND gates. The parallelism in generating the partial product is realized by ANDing the first bit (LSB) of the multiplier with the multiplicand bits. The second partial product is generated by ANDing the second multiplier bit with the multiplicand bits preceded by single zero. The third partial product is obtained by ANDing the third multiplier with the multiplicand bits preceded by double zeros. The Wallace tree has three steps:

- Multiply (that is - AND) each bit of one of the arguments, by each bit of the other, yielding n^2 results,
- Reduce the number of partial products to two by layers of full and half adders
- Group the wires in two numbers, and add them with a conventional adder.

7.THE BOOTH’S MULTIPLIER

The booth multiplier use the booth encoding algorithm in order to reduce the number of partial products by considering two bits of multiplier at a time, thereby achieving a speed advantage over other multiplier architectures. Thus the algorithm is applicable for both signed and unsigned numbers. Booth multiplier can be used in different modes such as radix-2, radix-4, radix-8 etc. But we decided to use Radix-4 Booth’s Algorithm because of number of Partial products is reduced to $n/2$.

7.1MODIFIED BOOTH MULTIPLIER

The modified booth encoding (MBE) or modified booth algorithm (MBA) was proposed by O.L.Macsorely. The disadvantage of booth algorithm or radix 2 algorithm are the number of add subtract operations and number of shift operations becomes variable and becomes inconvenient in designing parallel multipliers and becomes inconvenient when there is isolated ones.

7.1.1 MODIFIED BOOTH ENCODER (MBE)

Modified Booth encoding is most often used to avoid variable size partial product array. In MBE, the multiplier B has to be converted into a Radix-4 number by dividing them into three digits respectively according to Booth Encoder Table. Prior to convert the multiplier, a zero is appended into the Least Significant Bit (LSB) of the multiplier. The multiplier has been divided into four partitions and hence that mean four partial products will be generated using booth multiplier approach instead of eight partial products being generated using conventional multiplier.

B_{n+1} , B_n and B_{n-1} are three bit binary numbers of the multiplier. B_{n+1} is the most significant bit (MSB) and B_{n-1} is the least significant bit (LSB). Z_n is representing the Radix-4 number of the 3-bit binary multiplier number. For example, if the 3-bit multiplier value is “111”, so it means that multiplicand A will be 0.And it’s the same for others either to multiply the multiplicand by -1, -2 and so on depending on 3 digit number. From the table 1, the M, 2M and 3M are the elect control signals for the partial product generator. It will determine whether the multiplicand is multiplied by 0,-1, 2

or -2. M and 2M are designed as an active low circuit which means if let’s say the multiplicand should be multiplied by 1 then the M select signal will be set to low “0” whereas if the multiplicand should be multiplied by 2 then the 2M select signal will be set to low “0”. The 3M is representing the sign bit control signal and its active high circuit which means if the multiplicand should be multiplied by -1 or -2, then the sign, 3M will be set to high “1”.

TABLE 1: MODIFIED BOOTH ENCODER

B_{n+1}	B_n	B_{n-1}	Z_n	Partial product	1M	2M	3M
0	0	0	0	0	1	1	0
0	0	1	1	1*m	0	1	0
0	1	0	1	1*m	0	1	0
0	1	1	2	2*m	1	0	0
1	0	0	-2	-2*m	1	0	1
1	0	1	-1	-1*m	0	1	1
1	1	0	-1	-1*m	0	1	1
1	1	1	0	0	1	1	0

7.1.2 PARTIAL PRODUCT GENERATOR (PPG)

Partial product generator is the combination circuit of the product generator and the 5 to 1 MUX circuit. Product generator is designed to produce the product by multiplying the multiplicand A by 0, 1, -1, 2 or -2. A 5 to 1 MUX is designed to determine which product is chosen depending on the M, 2M, 3M control signal which is generated from the MBE. For product generator, multiply by zero means the multiplicand is multiplied by “0”.Multiply by “1” means the product still remains the same as the multiplicand value. Multiply by “-1” means that the product is the two’s complement form of the number. Multiply by “-2” is to shift left one bit the two’s complement of the multiplicand value and multiply by “2” means just shift left the multiplicand by one place.

7.1.3 SIGN EXTENSION CORRECTOR

Sign Extension Corrector is designed to enhance the ability of the booth multiplier to multiply not only the unsigned number but as well as the signed number. As shown in Table 4.2 when bit 7 of the multiplicand A(A_7) is zero(unsigned number) and B_{n+1} is equal to one, then sign E will have one value (become signed number for resulted partial product). It is the same when the bit 7 of the multiplicand A (A_7) is one (signed number) and B_{n+1} is equal to zero, the sign E will have a new value. However when both the A_7 and B_{n+1} are equal either to zero or one, the sign E will have a value zero(unsigned number). For the case when all three bits of the multiplier value B_{n+1} , B_n and B_{n-1} are equal to zero or one, the sign E will direct have a zero value independent to the A_7

value. The table for the Sign Extension Corrector is shown below.

7. RESULTS AND DISCUSSION

Thus an RNs based FIR filters are designed using wallace multiplier and modified booth multiplier inorder to reduce the power consumed and area. The reduction in power is given in the table1. The comparison tables for area and power of the existing and proposed fir filters are given

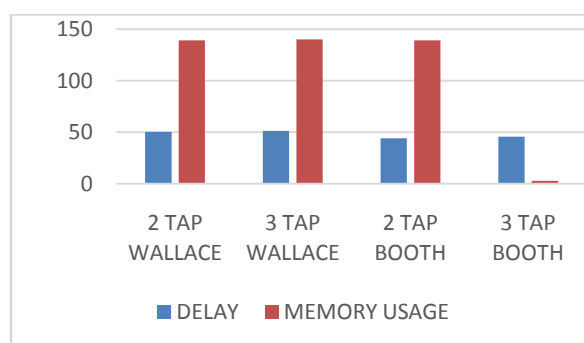
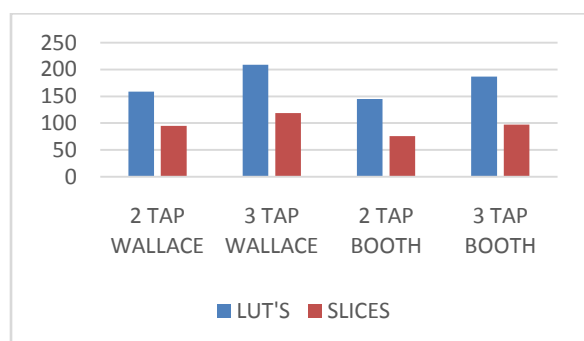
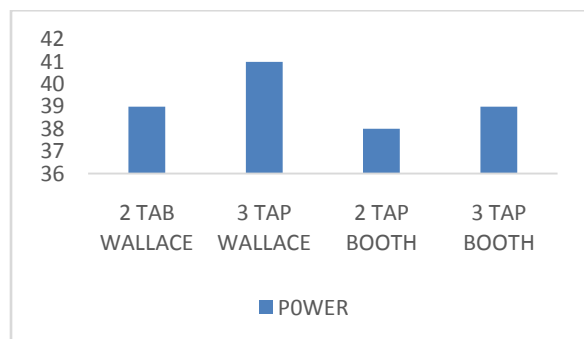


TABLE 2: Result Comparison

PARAMETERS	WALLACE TREE		BOOTH MULTIPLIER	
	2TAP	3TAP	2TAP	3TAP
No.of 4 input LUT's	159	209	145	187
Gate Count	1240	1636	981	1324
Delay (ns)	50.17	51.12	44	45.6
Power (mW)	39	41	38	39
Memory Usage (MB)	139	140	139	140

8. CONCLUSION

In this paper one two- and three-tap RNS filters using the three moduli base of the form $\{2^{n1}, 2^{n2}-1, 2^{n3}+1\}$ using Wallace tree and booth multiplier are designed. The filters are equivalent in terms of SNR. The RNS structures demonstrate power reduction in comparison to the binary structures. RNS bases have quantitatively been proved to achieve from up to 59% normalized delay variation savings when compared to the equivalent binary structures. The area and power of the proposed multiplier is reduced.

REFERENCES

- [1] F.Taylor," Residue arithmetic: A tutorial within examples," IEEE Computer, pp. 50-62, May 1984.
- [2] H.Mogal, H, H, H. Qian, S. S.Sapatnekar, and K. Bazargan, "Fast and accurate statistical criticality computation under process variations."IEEE Transaction on CAD of Integrated Circuits and Systems, vol. 28, no.3 pp. 350-363, 2009.
- [3] I. Kouretas and V. Paliouras, "Delay Variation Tolerant FIR Filter Architecture Based On The Residue Number System," in 2013 IEEE International Symposium on Circuits and Systems (ISCAS), may 201, pp. 2223-2226.
- [4] I. Kouretas and Paliouras, "Variation-tolerant design using Residue Number System," in Euro micro Conference on Digital System Design(DSD), Aug. 2009, pp.157-163.
- [5] I.Kouretas and Paliouras "Residue arithmetic for variation-tolerant design of multiply-add units," in PATMOS, 2009, pp.26-35
- [6] I. Kouretas and V. Paliouras, "A low-complexity high-radix RNS multiplier, "IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 56, no. 11, ov. 2009
- [7] I. Kouretas and V. Paliouras, "Residue arithmetic for designing multiply-add units in the presence of non-Gaussian variation," in 2012IEEE International Symposium on Circuits and Systems (ISCAS), may 2012, pp. 1231-1234
- [8] I.Kouretas and Paliouras "Residue arithmetic for variation-tolerant design of multiply-add units," in PATMOS, 2009, pp.26-35
- [9] K. Bowman,S. Duvall, and J.Meindl, "Impact of die-to-die and with indie Parameter fluctuations on themaximum clock frequency distribution for giga scale integration," IEEE Journal of Solid-State Circuits, vol.37, no.2, Feb2002.
- [10] O.Michael, N.Sani, and Deboning, Design for Manufacture ability and Statistical Design: A Constructive ApproachNJ, USA: Springer-Verlag New York, Inc., 2006.
- [11] Jian Wen ,Rou He Yao and Wei Jing Wu,"Efficient Modulo $2n+1$ Multiplier ", IEEE transaction on vlsi systems vol.19 no.12,dec 2011.
- [12] Ahmad A Hisat,"A New Efficient Structure of a Modular Multiplier for RNS" IEEE transactions on computers vol. 49 ,no.2, feb 2000.
- [13] Gian Carlo Cardarilli, Andea Del Re, Alberto Nannarelli & Marco Re,"Low Power and Low Leakage Implentation of RNS FIR filter",IEEE Transaction on Department of Informatics & math Modelling.
- [14] Riyaz A Patel, Mohammad Benaissa,"Novel Power-Delay -Area EfficientTo Generic Modular Multiplication"IEEE transaction on circuits and Circuits,vol 57,no.6,june 2007.