ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

FIR Filter Implementation Using Novel Modulo 2^n-2^k-1 Adder for RNS

V.Nandini
PG Scholar,
Department of ECE,
Dhanekula Institute of Engineering and Technology,
Vijayawada, India.

A.Sivannarayana
Assistant Professor,
Department of ECE,
Dhanekula Institute of Engineering and Technology,
Vijayawada, India.

Abstract:
A design of high performance modular adder with an application of FIR filter is implemented in this paper. Moduli set in the form of 2^n-2^k-1 ($1 \le k \le n-2$) can offer excellent balance among RNS channels for multi-channel RNS processing. Modular adder can be used in RNS addition, multiplication and division. In this paper, a Finite Impulse Response (FIR) filter using a novel algorithm and its Very Large Scale Integration (VLSI) implementation structure are proposed for modulo 2^n-2^k-1 adder. In the proposed algorithm, parallel prefix operation and carry correction techniques are adopted to eliminate the re-computation of carries. Any existing parallel prefix structure can be used. So we can get better trade-off between area and delay and power can be reduced. The total design is coded with verilog-HDL and the synthesis is done using cadence RTL compiler using typical libraries of TSMC 180nm and 45nm technology. The area, power and delay are compared for both technologies.

Keywords:
RNS, modulo adder, parallel prefix, FIR filter.

I. INTRODUCTION:

RNS is a non-weighted numerical representation system and has carry-free property in multiplication and addition operations. Recent days, it has received intensive study in the VLSI circuits design for digital signal processing (DSP) systems with high speed and low power consumption [1]-[3]. For integers A and B, which are of n-bit width, addition is performed as shown below

$$C = A + B \qquad A+B+T < 2^n$$
$$C = \langle A+B+T \rangle_{2^n} \qquad A+B+T \ge 2^n \quad (1)$$

In the above equation, one of the outputs is selected by given condition. The effective modulo adders in RNS are 2^n-1, 2^n, 2^n+1. These 2^n-1 and 2^n+1 adders are based on parallel prefix and carry correction respectively. Some modulo 2^n-2^{n-2}-1 adder based on the technique of carry offset, which is only required to obtain the carry information of A+B. In order to find the carries for addition each carry is modified to the utmost carry [3]. By using carry computation, the block corrects the carries further in the proposed modulo adder. In this paper, the modulo adder 2^n-2^k-1 is based on the carry correction and parallel prefix addition is proposed. This modulo adder is divided into four units pre-processing unit, the prefix computation unit, the carry correction unit, and the sum computation unit. This paper is organized as follows: Section III describes the RNS background and arithmetic operation. Section IV describes the novel modulo 2^n-2^k-1 adder. Section V describes the application of modulo adder in FIR filter. In section VI simulation results and comparisons are presented. Finally, we conclude this paper.

II. RELATED WORK:

Shang Ma, Jian-Hao Hu and Chen-Hao Wang proposed "A Novel Modulo 2^n-2^k-1 Adder for Residue Number System". In this system the addition operation is performed using the modulo 2^n-2^k-1. The sum computation operation is done in the adder, which consists of four units, which eliminates re-computation of carries.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

This system provides excellent performance and reduces the delay and area, thereby reducing the power. This paper explains the implementation of FIR filter using the novel modulo 2^n-2^k-1 adder to show the better performance compared to standard adders used in the filter.

III. RNS BACKGROUND AND ARITHMETIC OPERATION

An RNS is defined by a set of relatively prime integers called the moduli. Each integer X can be represented as a set of smaller integers called the residues [4]. This relation can be notationally written based on the congruence:

$$X \bmod m_i = r_i \qquad (2)$$

The RNS is capable of uniquely representing all integers X that lie in its dynamic range. The dynamic range is determined by the moduli-set $\{m_1, m_2,\ldots,m_n\}$ and denoted as M where,

$$M=\prod_{i=1}^{N} m_i \qquad (3)$$

Algebraic Operations:

Addition and subtraction of different numbers in the RNS representation is done by individually adding or subtracting the residues with respect to the corresponding moduli. Consider the moduli-set S = $\{m_1,m_2,\ldots m_n\}$ and the numbers A and B are given in RNS representation: X = $\{A_1,A_2,\ldots A_n\}$ and Y = $\{B_1,B_2,\ldots B_n\}$ Then,

$$Z = A \mp B = \{Z_1,Z_2,\ldots Z_n\} \qquad (4)$$

Where $Z_i = (Ai \mp Bi) \bmod m_i$

This property may be applied to subtraction as well, where subtraction of B from A is considered as the addition of Bbar. The modulo operation is distributive over addition or subtraction:

$$|A \mp B|_m = ||A|_m \mp |B|_m|_m \qquad (5)$$

IV. MODULO 2^N-2^K-1 ADDER:

The novel modulo 2^n-2^k-1 structure is shown in figure (1), which consists of four modules, pre-processing unit, carry generation unit, carry correction unit, and sum computation unit.

The modulo adder is divided into two binary adders A1 and A2 with a carry correction and sum computation model to get correction carries c_i^{real} from the process carries in preprocessing unit. At last we get the final modular addition with c_i^{real} and partial sum information [5].

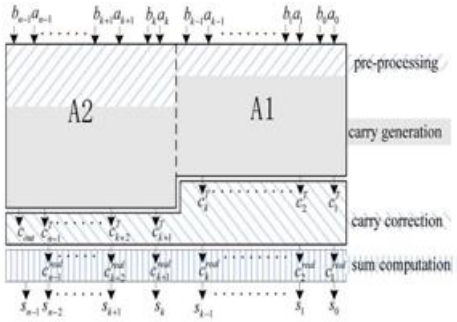

Fig.1: Modulo 2^n-2^k-1 adder structure

A. Pre-Processing Unit:

The preprocessing units are used for generating carries and carry propagation bits. The computation can be performed by A1 and A2 where A1 and A2 are used for lower k- bits and higher n-k bits addition respectively [5].

For lower k-bits carry propagation is calculated as

$$(g_0, p_0) = (a_0 + b_0, \overline{a_0 + b_0})i = 0$$
$$(g_i, p_i) = (a_i b_i, a_i \oplus b_i)i = 1,2,..k-1 \qquad (6)$$

For A2 (n-k bits) first find by simple carry save adder process

$$(g_i', p_i') = (a_i b_i, a_i \oplus b_i)$$

This $g'$and $p'$are inputs to next stage in A2 part addition. Then the output carry propagation for A2 is

$$(g_k, p_k) = (p_k', \overline{p_k'})i = k$$
$$(g_i, p_i) = (p_i' g_{i-1}', p_i' \oplus g_{i-1}')i = k+1,..n-1$$

The carry out for first unit is C_{SCSA}

$$C_{SCSA} = a_{n-1} b_{n-1} = g'_{n-1} \qquad (9)$$

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

B. Carry Generation:

In the carry generation unit, carry is generated by carry look-ahead adder.

$$C_{i+1}' = G_i + P_i C_i \qquad (10)$$

C. Carry Correction Unit:

The carry correction unit is used to get the real carries C_{i+1}^{real} for each bit needed for final sum generation from equation below.

$$\begin{aligned} c_{i+1}^{real} &= \{c_{i+1}^{T}(c_{out} + \overline{P_{i:0}})i = 0,1,..k-1 \\ &= c_{i+1}^{T}(c_{out} + \overline{P_{k-1:0}}(p_k \oplus c_k^T))i = k \\ &= c_{i+1}^{T}(c_{out} + \overline{P_{i:k+1}} + \overline{P_{k-1:0}}(p_k \oplus c_k^T))\}i = k+1,..n-2 \end{aligned} \qquad (11)$$

D. Sum Computation:

The final sum is calculated by the real carries generated from the carry correction unit. The final sum is given by

$$\begin{aligned} s_i &= c_{out} \oplus \overline{p_0} i = 0 \\ s_i &= c_k^{real} \oplus c_{out} \oplus \overline{p_k} i = k \\ s_i &= c_i^{real} \oplus p_i i = 1,..k-1, k+1 \ldots, n-1 \end{aligned} \qquad (12)$$

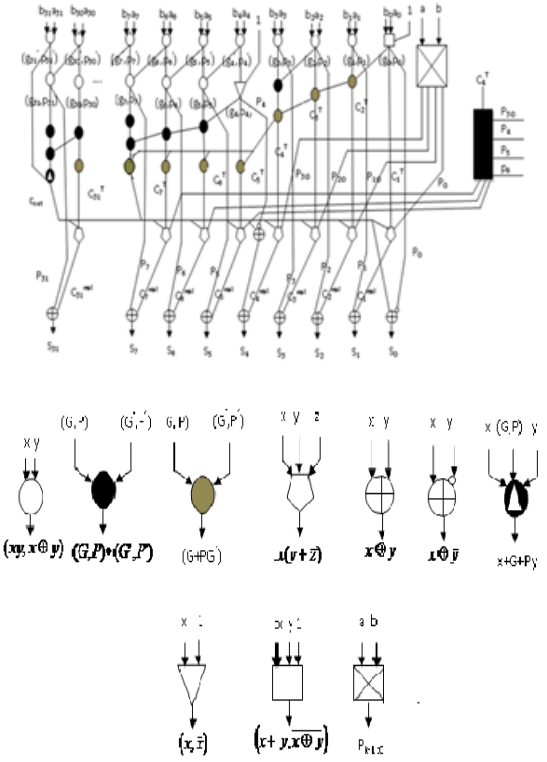

Fig.2. Proposed modulo 2^{32}-2^{16}-1 adder

V. IMPLEMENTATION OF MODULO ADDER IN FIR FILTER

In signal processing, a finite impulse response (FIR) filter is a filter whose impulse response (or response to any finite length input) is of finite duration, because it settles to zero in finite time. A FIR filter is designed by finding the coefficients and filter that meet certain specifications, which can be in time-domain or frequency-domain. In this chapter we implement FIR filter with a modulo 2^n-2^k-1 adder.

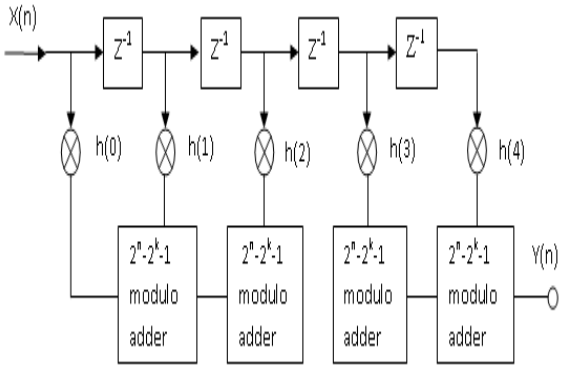

Fig.3: FIR filter with modulo 2^n-2^k-1 adder

In this, the input signal is multiplied with the coefficients and single clock delay is added through the adder. Here the adder used is modulo 2^n-2^k-1 adder. The main advantage of using this adder in the filter is that, it reduces the computation time. It also reduces area, delay, power and complexity. It improves speed of FIR filter.

VI. SIMULATION RESULTS:

We have designed the FIR filter in verilog using proposed modulo adder for bit-widths 8 and 32.All the designs are synthesized in the Cadence RTL compiler using 180nm and 45nm CMOS library. The net list file Extracted from RTL Compiler. As shown in Table.1 the area, delay and power are reduced in 45nm technology compared to 180nm technology. For modulo adder (32bit) delay is reduced to 57.1% and power reduced to 96.3%. For FIR filter (32bit) delay reduced to 52.6%, power reduced to 98%. Fig. 4 and Fig. 5 show the simulation results of modulo adder and FIR filter for 32bit.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

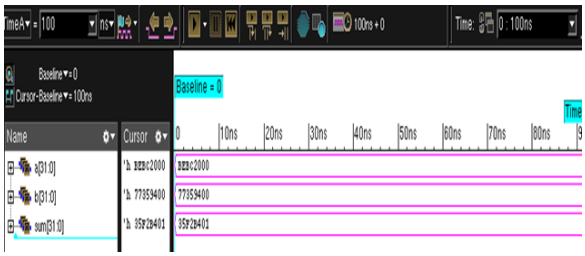

Fig.4. Simulation results of modulo adder

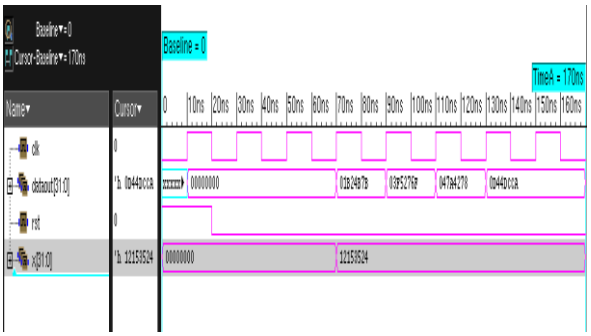

Fig.5. Simulation results of FIR filter using modulo adder

Comparison:

DESIGN	AREA		DELAY(ns)		POWER(µW)	
Technology (nm)	180 nm	45nm	180 nm	45 nm	180nm	45nm
Modulo adder (8bit)	1271	133	2.03	0.67	72.61	2.76
Modulo adder (32bit)	5096	534	3.62	1.55	267.94	9.77
FIR filter with modulo adder (8bit)	1610	38	0.64	0.27	49.11	2.93
FIR filter with modulo adder (32bit)	16390	170	0.57	0.27	690.59	13.43

Table.1. Comparison of area, delay and power for modulo adder and FIR filter

VII. CONCLUSION:

In this paper, a new class of modulo 2^n-2^k-1 adder is proposed. The novel modulo adder structure consists of pre-processing unit, carry generation unit, carry correction unit and sum computation unit. By using this novel modulo adder an FIR filter is implemented. The comparison shows that the proposed algorithm can construct a new class of general modular adder with better performance in delay. We got fine trade-off between area and delay. Hence power dissipation also reduced.

REFERENCES:

1. Shang Ma, Jian-Hao Hu, and Chen-Hao Wang "A Novel Modulo 2^n-2^k-1 Adder for Residue Number System" IEEE Transactions on circuits and systems-I, vol.60, NO.11,NOVEMBER 2013.
2. P. Patronik, K. Berezowski, S. J. Piestrak, J. Biernat, and A. Shrivastava, "Fast and energy-efficient constant-coefficient FIR filters using residue number system," in Proc. Int. Symp. Low Power Electronics and Design (ISLPED), 2011, pp. 385–390.
3. J. C. Bajard, L. S. Didier, and T. Hilaire, "ρ-direct form transposed and residue number systems for filter implementations," in proc. IEEE 54th Int. Midwest Symp. Circuits and systems, 2011, pp. 1-4.
4. M. Bayoumi, G. Jullien, and W. Miller, "A VLSI implementation of residue adders," IEEE Trans. Circuits Syst., vol. CAS-34, no.3, pp. 284-288, Mar. 1987.
5. G. Jaberipur, B. Parhami, and S. Nejati, "On building general modular adders from standard binary arithmetic components," in Proc. 45thAsilomar Conf. Signals, Systems, andComputers, 2011, pp. 6–9.
6. C. Efstathiou, H. T. Vergos, and D. Nikolas, "Fast parallel-prefix modulo 2^n+1 adders," IEEE Trans. Comput., vol. 53, no. 9, pp.1211-1216, sep. 2004.
7. P. M. Matutino, R. Chaves, and L. Sousa, "Arithmetic units for RNS moduli 2^n-3 and 2^n+3

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

operations," in proc. 13th Euro micro conf. Digital system design: Architecture, Methods and Tools (DSD), 2010, pp. 243-246.

8. A. A. Hiasat, "High-speed and reduced-area modular adder structures for RNS," IEEETrans. Comput,, vol. 51, no. 1, pp. 84–89, Jan. 2002.
9. R. A. Patel, M. Benaissa, and S. Boussakta, "Fast modulo 2^n- (2^{n-2}+1) addition: A new class of adder for RNS,"IEEE Trans. Comput., vol. 56, no. 4, pp. 572–576, Apr. 2007.