ISSN 2320-9984 03 >
9 772320 998022

Volume 2 Issue 3 | September 2014 ISSN: 2320-9984 (Online)

International Journal of Modern Engineering & Management Research

Website: www.ijmemr.org

An Efficient Adaptive RNS Filter Design on High Speed FPGA

Divyansh Namdev
Research Scholar
Takshshila Institute of Engineering & Technology,
Jabalpur (M.P.) [INDIA]
Email: samarthdivyansh@gmail.com

Rakesh Patel
Associate Professor
Department of Electronics and Communication Engg.
Takshshila Institute of Engineering & Technology
Jabalpur (M.P.) [INDIA]
Email: rakeshpatel@takshshila.org

Abstract*—Digital filters are important component in Very Large Scale Integration (VLSI) design. The available Finite Impulse Response (FIR) filter provide a long transient response, which is their major limitation. To overcome this drawback, Residue Number System (RNS) based FIR filters are **proposed** here in this paper. High-speed is obtained by using the residue arithmetic way that allows the computation for filter output by using N FIR sub filters for reduced dynamic range working in parallel form. Total three moduli 2^n-1, 2^n, 2^n+1 sets are used in proposed RNS based Filter. 4-tap Low Pass Filter (LPF) for FIR filter, and RNS based FIR filter along with 4-tap LPF are designed. Verilog HDL language is used for RTL entry and for analysis in this paper. The simulation is done using Xilinx EDA tool Integrated Simulation Environment ISE-12.2*

Keywords:— *ISE, RNS, Verilog, LPF, FIR, EDA, DSP, moduli, RTL.*

1. INTRODUCTION

In recent years, there has been significant development in the field for Digital Signal Processing (DSP) along with the advancement in VLSI technology. Various applications for DSP include audio, image and video processing and consumer electronics[1] . FIR digital filters are widely used in digital signal processing by virtue for stability and easy implementation. The major drawback for FIR filters is the increased amount for calculation needed to process a signal through the FIR filter. The advent for VLSI technology and DSP processors provides an opportunity for significantly increased efficiency for RNS to minimize the delay. RNS approaches are becoming famous for designing high efficient DSP processors because of its ability to carry free arithmetic computation. This carrying of free computations lead to lump-sum execution of arithmetic operation on its residues. But in RNS, selection of moduli is one of the most important aspect that determines bit area, power consumption, efficiency, speed etc.

In RNS, arithmetic operations on large integers are done, by splitting them into smaller residues and performing the operations, independently and parallel form, thereby speeding up the whole operation referred in paper [2, 3]. In proposed paper, an attempt has been made, to design and simulate LPF for RNS based FIR filters, with the consideration for 4 tap using Verilog.

2. FIR FILTER

The output say (y) for any LTI system is determined by performing convolution of its input signal say (x) with its impulse response say (b). For a discrete time FIR filter, the output is weighted sum of the present input and some finite number for previous values for the input. The working is described by the following equation, which shows the output sequence y[n] in terms for its input sequence x [n]

$$y(n) \sum_{i=0}^{N} b\ x(n-i) \quad \text{..... (1)}$$

Here,

y(n) is output signal

x(n) is input signal

b_i are the coefficients of filter, also called as tap weights, that develops the impulse response and N is the order of filter. An Nth-order filter has shown on the right-hand side in equation (1). The x(n-i) in these terms are referred to as taps, basis on the structure for a tapped delay line. There are many implementations or blocks which provide the delayed inputs to the multiplication computations.

FIR digital filters have exactly linear phase response and suffer less from the effects for finite word length as compared with IIR digital filters. The main components for FIR filter are adder, multiplier and delay. The carry propagation delay is a limiting factor for the adder and multiplier. The delay for FIR filters is reduced by incorporating RNS based modulo adder and modulo multiplier in the simple FIR filter. Prior to simulation for FIR filter, the coefficients are scaled by the following rules given below.

3. RNS BASED FIR FILTER

Researchers have discussed about the optimized RNS based FIR filter model [4, 5]. It is defined by a set for relatively prime integers called the moduli. The moduli set are denoted as $\{m_1,m_2,.....m_n\}$ where $m_1,m_2,.....m_n$ are the modulus. Each integer can be represented as a set for smaller integers called the residues. The residue-set is denoted as $\{r_1,r_2,.....r_n\}$ where r_i is i^{th} residue. The residue is defined as the least positive remainder, when X is divided by the moduli m_i [5]. This relation can be notationally written (based on the congruence and the equation) as given below:

$$Mod\ mi = r_i \quad \text{..... (2)}$$

The same congruence can be written in an alternative notation as:

$$|X| mi = r_i \quad \text{..... (3)}$$

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 {m1, m2, .., mn} and, denoted as M where:

$$M = \prod_{i=1}^{n} m_i \quad \text{..... (4)}$$

The RNS provides unique representation for all integers in the range between 0 and M-1. If the integer X is greater than M-1, the RNS representation repeats itself. Therefore, more than one integer might have the same residue representation. It is important to emphasize that the moduli have to be relatively prime to be able to exploit the full dynamic range M.

The figure 1 below shows the general structure for RNS based FIR filter. X(t) and Y (t) are the input and output for this figure. The forward and reverse conversion is based on the special moduli set and the New Chinese Remainder Theorem (NCRT) and the three FIR filter blocks are used here to speed up the processes.

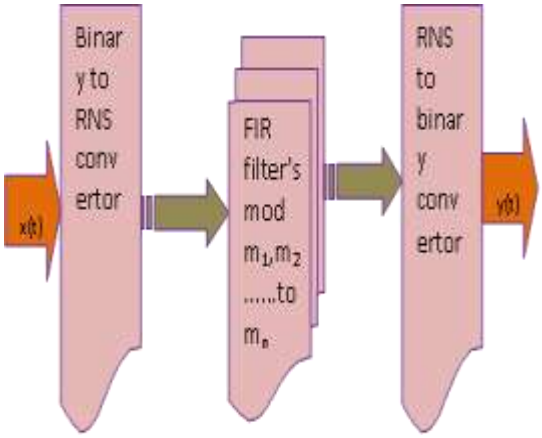

Figure 1: RNS based FIR filter

Choice for Moduli: The moduli should satisfy the following conditions. They should be relatively prime. The moduli should be as small as possible so that, operations modulo require minimum computational time. The moduli should imply simple weighted to RNS

and RNS to weighted conversions as well as simple RNS arithmetic. The moduli set should be of the forms 2^k+1,$2^k-1$and 2^k for simple conversions, and simple arithmetic in RNS system. The product for the moduli should be large enough in order to implement the desired dynamic range. The moduli should create a balanced decomposition for the dynamic range.

4. DESIGN METHODOLOGY

Proposed work shows a new approach for forward conversion. The forward conversion stage is of paramount importance as it is considered as an overhead in the overall RNS. Forward converters are usually classified into two categories based on the moduli used. The first category includes forward converters based on arbitrary moduli-sets. These converters are usually built using look-up tables. The second category includes forward converters based on special moduli-sets. The use of special moduli-sets simplifies the forward conversion algorithms and architectures referred in[6] Usually, the special moduli-sets are referred to as low-cost moduli-sets. In this section, special moduli-set $\{2^n+1,2^n,2^n-1\}$ is focused, as it is the most commonly used moduli-set for this design. Researchers have discussed about the importance for special moduli-set forward converters. A typical architecture, for the implementation for a forward converter, from binary to RNS representation, using the special moduli-set, is shown in Figure 2. One way for implementing a residue adder, for modulo m structure is composed, for one n-bit adder referred in [7].

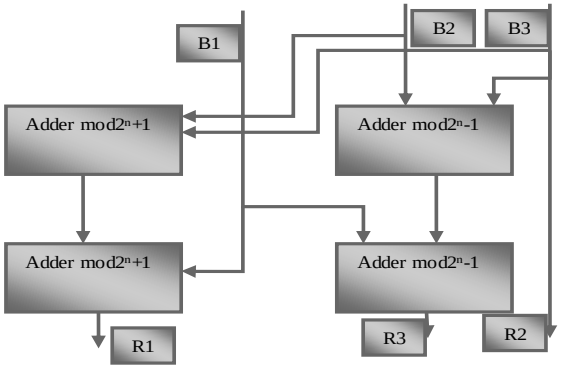

Figure 2: Forward $\{2^n+1, 2^n, 2^n-1\}$ Converter

The modulo 2^n-1 adder structure is shown in figure 3.This adder adds two n-bit numbers, X and Y in the first step. In the second step, the result is given to the multiplexer for 2^n-1 adder. In 2^n+1adder, the first stage is same as that for 2^n-1 adder but the second step is done with the two's complement for modulo m. The final result is selected between the two outputs according to the two output carries. Modulo 2^n+1 adder in series method is shown in Figure 4. The delay for modulo 2^n+1 structure is equal to the delay for two (n+1)- bit adders as well as the delay for one (n+1)-bit 2×1multiplexer referred in[8].

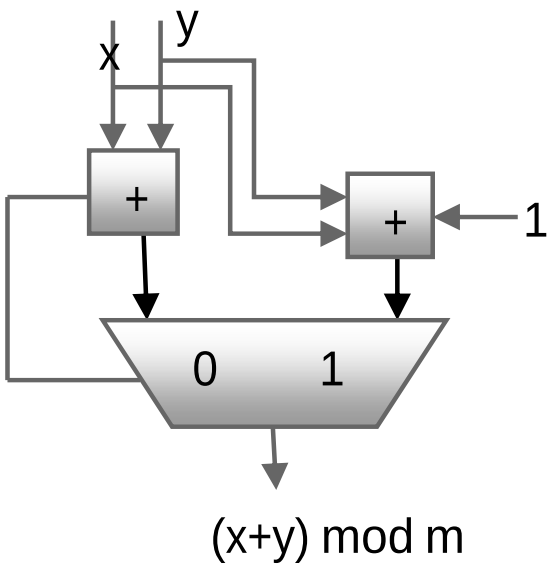

Figure 3: modulo 2^n-1 adder

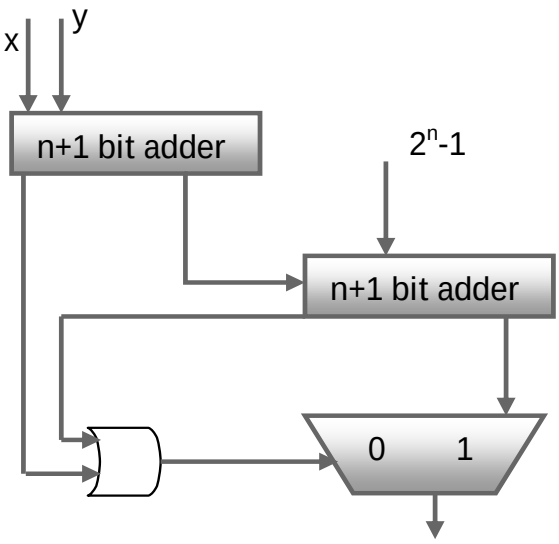

Figure 4: modulo 2^n+1 adder

5. TOOL, PLATFORM AND LANGUAGE

Language and tools used in this thesis work. Xilinx ISE 12.2 is used for simulation and Verilog has been considered as a programming language Verilog is the Hardware Description Language that may be used to design a digital system at many levels for abstraction, ranging from the algorithmic level to the minimum gate level.

The Xilinx ISE Simulator (ISim) is a HDL simulator that enables you to perform functional (behavioral) and timing simulations for VHDL, Verilog and mixed-language designs. The basic simulation flow is shown in the figure 5.

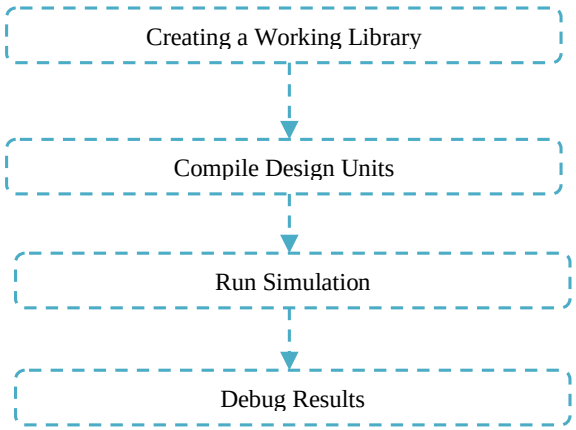

Figure 5: Simulation Flow

The target device used is Xilinx's Vertex 4 xc4vlx25-12ff668.

Figure 6: Package marking for target device

6. CONCLUSION

FIR Filter in Modern digital and discrete data communication, plays very importance roll. Existing works are themselves an achievement, but the long computation time proposed work, just gives a new approach to replace that, highly complex computation with simple residue based computations. Paper work mainly proposes, a new forward $\{2^n+1, 2^n, 2^n-1\}$ convertor, which uses modulo 2n+1 and 2n-1 adder and, that forward convertor will further be used in the design for FIR filter and RNS based FIR filter, considering 8-tap LPF, by using Verilog language. The forward (binary to RNS) and reverse (RNS to binary) conversion blocks, for RNS based FIR filters, have been designed for $\{2^n+1,2^n,2^n-1\}$ moduli and then incorporated in the RNS based FIR filter, which is simulated and analyzed .

REFERENCES:

[1] Richard Conway, John Nelson, "Improved RNS FIR filter Architecture", IEEE Transaction on Circuits and Systems, vol.51, no.1, pp.26-28, January 2004.

[2] Salvatore Pontraelli, Gian Carlo Cardarilli, Marco Re, Adelio Salsano, "Optimized implementation for RNS FIR filters based on FPGAs", Springer Journal Signal Processing System, vol. 67, no.3, pp.201-212, September 2012.

[3] Pemmaraju V. Ananda Mohan, "RNS to Binary converter for a Three moduli set {2n+1-1,2n, 2n-1}", IEEE Transaction on Circuits and Systems, vol.54, no.9, pp.775-779, September 2007.

[4] E. Grosswald, "Topics from the theory for Numbers", Collier Macmillan Ltd., New York, 1996.

[5] A. Omondi and B. Prem Kumar, "Residue Number System: Theory and implementation", Imperial College Press, 2007, (ISBN 978-1-86094-866-4).

[6] Jean-Luc Beuchat, "Some Modular Adder and Multipliers for Field Programmable Gate Arrays", Proceedings for the IEEE 17th

International Symposium on Parallel and Distributed Processing, Washington DC, USA, pp-190.2, April 2003.

[7] Somayeh Timarchi, Keivan Navi, "Improved Modulo 2n +1 Adder Design", World Academy for Science, Engineering and Technology, 2001.

[8] George Ignatius, Murali Krishna Varma U, Nitish. S. Krishna,Sachin. P.V. P. Sudheesh, Extended Kalman Filter based Estimation for Fast Fading MIMO Channels, IEEE, 2013