International Organization of Research & Development (IORD)
ISSN: 2348-0831
Vol 01 Issue 02 | 2014

Low Power and High Speed Design Of FIR Filter

Mr. Pravin Y.Kadu, Ku. Shubhangi Dhengre

Department Of electronics and communication
Abha Gaikwad-Patil College of Engineering & Technology Nagpur (MH)

ABSTRACT

Finite impulse response (FIR) filters are widely used in various DSP applications. The low-power or high speed techniques developed specifically for digital filters can be found in. Most of the applications in digital communication, seismic signal processing (noise elimination), speech processing (adaptive noise cancelation), and many other synthesis operations of signal require large order FIR filters ,since the number of multiply-accumulate (MAC) operations required per filter output increases linearly with the filter order, therefore implementation of these filters of large orders is a challenging task .In this paper, we propose designing of FIR filter using high speed low-power multiplier adopting the new implementing approach. We are using Vedic Multiplier. Vedic Multiplier is a fast processing multiplier as number of partial products are less. The carry look ahead adder will avoid the unwanted addition and thus minimize the switching power dissipation. The architecture is coded in VHDL, simulated in ModelSim and synthesize in Xlinx ISE Software .

Index Terms: Xilinx ISE, FIR, VHDL.

I. INTRODUCTION

Finite impulse response (FIR) filters are highly used in most of the DSP applications. Most of the applications, the FIR filter circuit must be able to operate at high sampling rates, while in many applications, the FIR filter circuit must be a circuit operating at moderate sample rates and uses low power. Various techniques can be applied to digital FIR filters to either increase the effective speed or reduce the power consumption of the original filter. Less work has been done that dealing with reducing the hardware complexity or power consumption of FIR filters. Since many years parallel processing applied to an FIR filter involves the hardware units replication that exist in the original filter. The choice of the multiplier and adder circuit also affects the resultant power dissipation. If the multiplier is choosen wisely with less number of calculations speed and power can be optimized.

Multipliers play an important part in digital signal processing systems. Multipliers consume considerable power have large area and long time period. Therefore, multiplier having low power design has been an important part in low-power VLSI system design. Aim of doing this work is optimization of multiplier designs produces more power-efficient solutions than optimization only at low levels. The main objective is reduction in power with small area and delay. We can optimize the area, power and speed using optimized architectures of adder and multiplier.

II. PROPOSED WORK

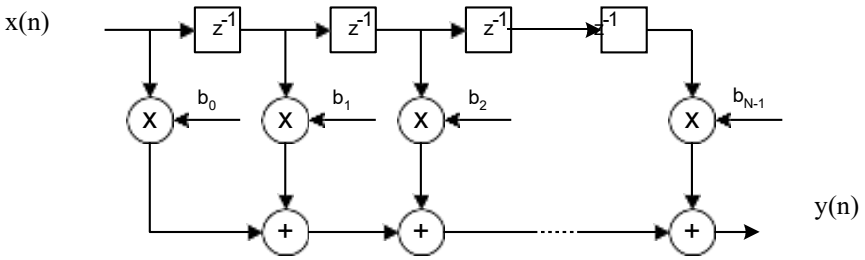

Fig. 1: Structure of n-tap FIR filter

The five steps for design of a digital filter involve:

1. Filter specification: It consists giving the type of filter, say high pass filter, the required amplitude and phase responses, the word length of the input data.and the sampling frequency.

2. Filter coefficient calculation: The choice of coefficient calculation method will be influenced by several things. The main of which are the critical requirements i.e. specification. The optimal, frequency and window sampling method are the most commonly used.

3. Realization: In this converting the transfer function into a suitable filter network or structure.

4. Analysis of finite word length effects: The result of quantizing the filter coefficients and input data as well as the effect of carrying out the filtering Start Performance specification Calculation of filter coefficients Realization structuring Finite world length effects analysis H/W or S/W implementation Stop operation using fixed word length on the filter performance is analyzed here.

5. Implementation: In this process involves producing the software code and/or hardware and performing the actual filtering.

The analysis of linear, time-invariant FIR filter is generally carried out by using the *Z* transforms. Review of the *Z*-transform is presented. The filter structures characterizing the difference equations are represented using basic elements such as multipliers and adders.

III. Methodology and Implementation

1. Studying Basic Blocks in FIR Filter

Parameters of a FIR filter depends on the basic building blocks like multiplier, delay and adder. So we have to study the effect of these blocks on FIR filter viz.

a) Adder

Addition is the most common and often used arithmetic operation on microprocessor, especially digital computers. It works as a building block for synthesis all other arithmetic operations. Hence, in context of the efficient implementation of an arithmetic unit, structures of the binary adder become a very critical hardware unit.

b) Multiplier

A binary multiplier is an electronic circuit used in digital electronics like a computer to multiply two binary numbers. Multiplier is built using binary adders. Many type of computer arithmetic techniques can be used to implement a digital multiplier.Many techniques involve computing a set of partial products and then adding the partial products together. There are various typed of multipliers, but we are discussing only Array Multiplier and Vedic Multiplier.

2. Comparison between different Adders and Multipliers

Comparison is done between different architectures of adders and multipliers.

Types of Adder:	**Types of Multiplier:**
• Ripple Carry Adder	Array Multiplier
• Carry Look Ahead Adder	Vedic Multiplier

After comparing we found that Carry Look Ahead Adder is advantageous over Ripple Carry Adder. And Vedic Multiplier is found to be more advantageous over Array Multiplier.

3. Selecting an Optimized Adder and Multiplier

3.1. Carry Look Ahead Adder

A carry-look ahead adder (CLA) is a type of adder used in digital system. A carry-look ahead adder improves speed by reducing the amount of time required to wait for the carry bits. It can be compared with the simpler, but usually slow, ripple carry adder for which the carry bit is calculated side by side with the sum bit, and each bit waits till the previous carry has been calculated to begin calculating carry bits and its own result. The CLA adder calculates one or more carry bits before doing the actual sum, which minimizes the wait time to calculate the result of the larger value bits.

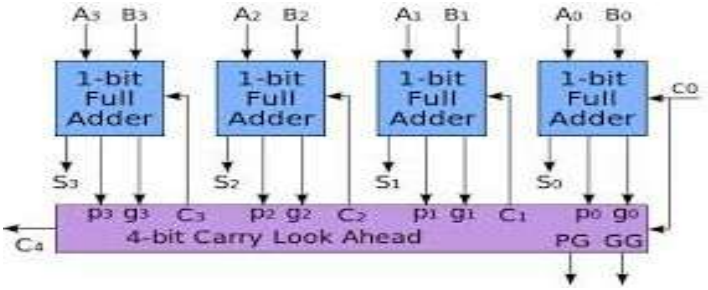

Fig. 2: 4-bit Carry Look Ahead Adder

Let ai and bi be the augends and addend inputs, ci the carry input, si and ci+1 , the sum and carry-out to the ith bit location. The pi and gi (auxillary functions) are called the propagate and generate signals, the output of adder is given by:
pi = ai + bi gi = ai bi
si = ai xor bi xor ci ci+1 = gi + pici

3.2. Vedic Multiplier

The multiplier architecture is based on this Urdhva tiryakbhyam sutra. The benefit of this algorithm is that partial products and their sums are calculated at the same time. It makes the multiplier clock independent. Regularity is the another advantage of this multiplier as compared to many other multipliers. Hence lay out design will be easy due to its modular behaviour. Architecture can be ellaborated using two 4 bit numbers i.e. the multiplier and multiplicand are 4 bit numbers. The multiplier and multiplicand and bits are divided into four bit blocks. These four bit blocks are again splitted into two bit multiplier blocks. According to the algorithm the 8 x 8 (A x B) bit multiplication will be as follows.

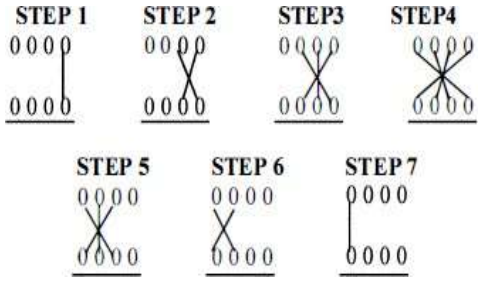

Fig. 3: 4*4 Multiplication technique using Urdhva tiryakbhyam sutra.

COMPUTATIONAL COMPLEXITY OF CONVENTIOANL ARRAY MULTIPLIER AND VEDIC MULTIPLIER

Input Bit Length	Number of calculations			
	Conventional		*Vedic*	
	M	A	M	A
2	4	2	4	1
3	9	7	9	5
4	16	15	16	9
8	64	77	64	53

M: Number of multiplications, A: Number of additions

Fig. 4: Comparison between Array and Vedic multiplier

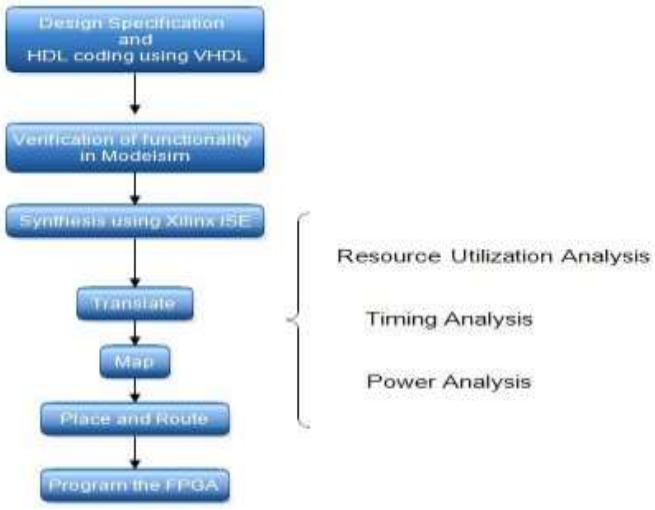

Fig. 5: Steps involved in Design & Implementation

In our project, we analyze the different architecture of adders and multipliers. In our case we are going to consider Vedic multiplier for the operation of FIR Filter. We are considering the area resource utilization, power and time delay for FIR filter architecture.

IV. SIMULATION AND RESULTS

In this work we are evaluating the performance of the proposed FIR filter using low power consumption multiplier by comparing vedic multiplier with the different multipliers. These multipliers can be implemented using VHDL coding. To get the power report and delay report we are synthesizing these multipliers using Xilinx and Modelsim.

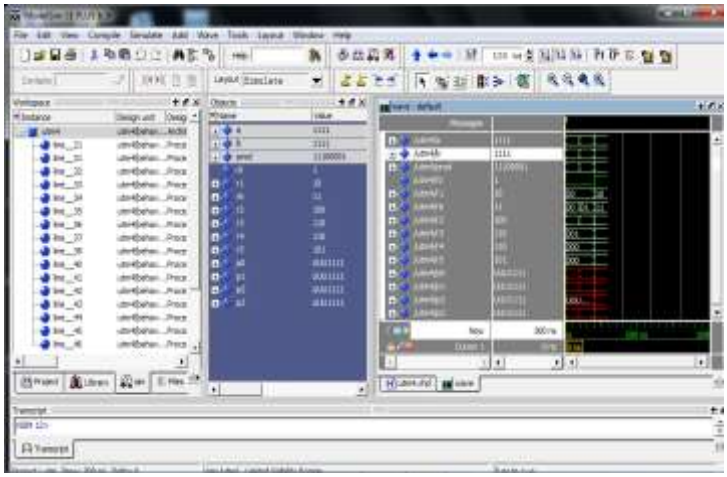

Fig. 6: Simulation Result for Vedic Multiplier

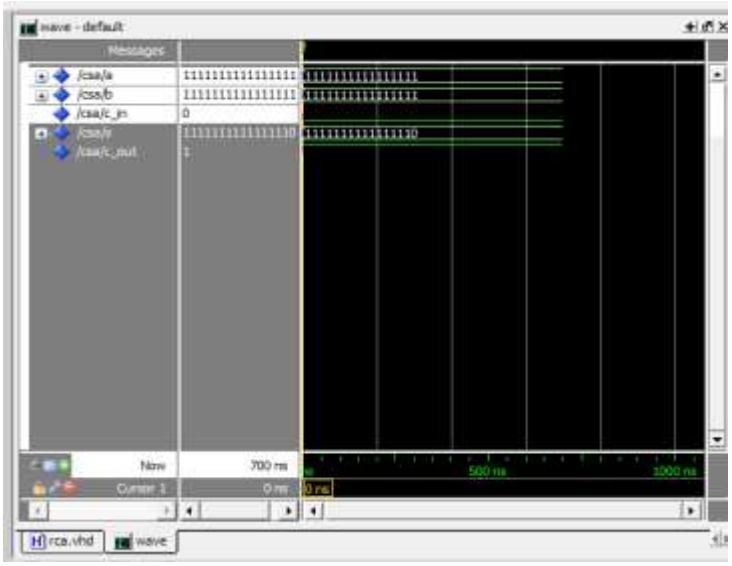

Fig. 7: Simulation Result of Carry Look Ahead Adder

Fig 8: Simulation Results of FIR filter

```
Device utilization summary: For CLA
---------------------------
Selected Device : 3s500efg320-4
 Number of Slices:                        4  out of   4656     0%
 Number of 4 input LUTs:                  8  out of   9312     0%
 Number of IOs:                          14
 Number of bonded IOBs:                  14  out of    232     6%
```

Fig 9: Device utilization Summary for 4-bit CLA

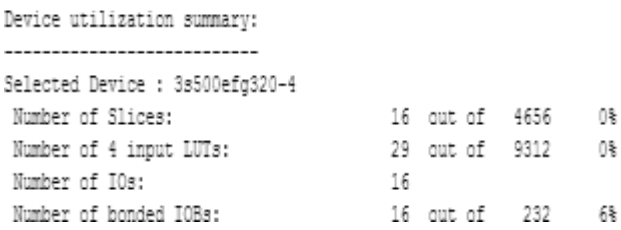

Fig 10: Device utilization Summary for 4-bit Vedic Multiplier

V. COMPARISIONS

Component	Type	Delay (ns)	Power Dissipation (mw)
Adder	Ripple Carry Adder	24.92	68.33
	Carry Look Ahead Adder	9.926	68.33
Multiplier	Array Multiplier	37.38	68.55
	Vedic Multiplier	24.77	68.52

Table 1. Comparison between adders and Multipliers

VI. CONCLUSIONS

This paper presents an efficient FIR filter based on Vedic multiplier and carry look ahead adder. Vedic Multiplier gives us method for hierarchical multiplier design and clearly indicates the computational advantages offered by Vedic methods.

The Computational path delay for proposed 4*4 bit Vedic multiplier and Carry look ahead Adder is found to be lesser then the conventional multiplier and ripple carry adder respectively. Hence our motivation to reduce delay is finally fulfilled. Vedic multiplier has less number of gates required for given 4*4 bit multiplier so its power dissipation is very small as compared to other multiplier architecture. Similarly in case of Carry look ahead Adder less number of gates are required. An awareness of Vedic mathematics can be effectively increased it must be included in engineering education which may lead to improvement significantly in many areas where fast arithmetic computational are critical such as real time DSP applications.

VII. REFERENCES

[1] Bahram Rashidi, Majid Pourormazd and Bahman Rashidi “Design and Implementation of Low Power Digital FIR Filter based on low power multipliers and adders on xilinx FPGA” IEEE 2011.
[2] R.Raja Sulochana, S Nagakishore Bhavanam Vasujadevi Midasala, and Jeevan Reddy K “Design and Implementation of FPGA based Low Power Digital FIR Filter” IOSR-JECE 2012.
[3] Sarita Chouhan and Yogesh Kumar “LOW POWER DESIGNING OF FIR FILTERS” IJATER 2012.
[4] S. Prem Kumar, G. Damodharan S. Sivaprakasam, and V. Ellappan submitted “OPTIMIZATION OF LOW POWER USING FIR FILTER” IJCSE 2011.
[5] Ramalatha, M Dayalan, K D Dharani, P Priya, and S Deborah, "High speed energy efficient ALU design using Vedic multiplication techniques", ICACTEA, 2009.
[6] Shahnam Mirzaei, , Ryan Kastner Anup Hosangadi “FPGA Implementation of High Speed FIR Filters Using Add and Shift Method” IEEE 2006.