“IMPLEMENTATION OF HIGH-SPEED OPERATING FIR FILTER WITH DA ALGORITHM COMPARING RESULTS WITH MAC ALGORITHM AND SIMPLE FIR FILTER RESULT”

Ekta H. Jain^{*1}, Chandu.N. Bhoyar²

^{*1}Department of Electronic (VLSI) Engineering RTMNU University, India

²Department of Electronic Engineering, RTMNU University, India

^{*}ektahjain@gmail.com, ²cnbhoyar@yahoo.com

***Corresponding Author: -**

Email: ektahjain@gmail.com

Abstract: -

Recent years there has been an increasing trend to implement digital signal processing functions in Field Programmable Gate Array (FPGA). therefore, we need to put great effort in designing efficient architectures for digital signal processing functions such as FIR filters, which are widely used in audio and video signal processing, telecommunications etc. We are going to present a method for implementing high speed Finite Impulse Response (FIR) filters using MAC (MULTIPLY AND ACCUMULATE) and Distributed Arithmetic (DA) method. MAC is a conventional FIR filter In these method adders, multipliers and delay elements are used. Distributed Arithmetic (DA) has been used to implement a bit-serial scheme of a general symmetric version of an FIR filter due to its high stability and linearity by taking optimal advantage of the look-up table (LUT) based structure of FPGAs. The performance of the DA technique for FIR filter design is analyzed and the results are compared to the MAC design technique.

Keyword: - “FIR Filter”, “DA Algorithm”, “MAC Algorithm”, “LUT”

Distributed under Creative Commons CC BY-NC 4.0 OPEN ACCESS

I INTRODUCTION

Due to the increasing use of FIR filters in video and communication systems, high performance in area, speed and power consumption is needed. FPGA based implementation of FIR filters on traditional method costs considerable hardware resources, which goes against the decrease of circuit scale and the increase of system speed. A new design and implementation of FIR filters using Distributed Arithmetic is provided in this paper to solve this problem. Distributed Arithmetic structure is used to increase the resource usage while pipeline structure is also used to increase the system speed. In addition, the divided LUT method is also used to decrease the memory units. The simulation results show that FIR filters using Distributed Arithmetic can work stable with high speed and can save almost 50 percent hardware resources to decrease the circuit scale, and can be applied to many areas for its great flexibility and high reliability. Noise which can degrade the quality of the desired portion. In such system we can remove or filter out the useless samples. For example, if we consider the case of the telephone system, there is no need to transmit very high frequencies since most speech falls within the band of 400 to 3,400 Hz. Therefore, all frequencies up and below that band are filtered out. The frequency band within 400 and 3,400 Hz which is not filtered out is known as the pass band, and the frequency band that is blocked is known as the stop band. This equation can also be expressed as a convolution of the coefficient sequence with the input signal:

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

That is the filter output is a weighted sum of the current and a finite number of previous values of the input. The process of selecting the coefficients and filter's length is called filter design. The goal is to set those parameters such that certain desired stop band and pass band parameters will result from running the filter.

II BLOCK DIAGRAM OF FIR FILTER

Fig1-.The logical structure of an FIR filter

Fig 2-.The logical structure of 4 TAP FIR Filter

Fig.3.- RTL view of 4 tap FIR filter

Fig 4-.simulation result of 4 tap FIR Filter

Table I.- Synthesis report of 4 tap fir filter

Device Utilization Summary (estimated values)				
Logic Utilization	Used	Available	Utilization	
Number of Slices	27	960	2%	
Number of Slice Flip Flops	52	1920	2%	
Number of 4 input LUTs	42	1920	2%	
Number of bonded IOBs	25	66	37%	
Number of MULT18X18SIOs	3	4	75%	
Number of GCLKs	1	24	4%	

III.INTRODUCTION TO MAC ALGORITHM

In computing mainly digital signal the multiply–accumulate operation is a common step that computes the product of two numbers and adds that product to an accumulator. The hardware unit that performs the operation is also called as a multiplier–accumulator (MAC unit) the operation itself is also often called a MAC or a MAC operation.

$$a \leftarrow a + (b \times c).$$

The MAC unit provides hardware support for a limited set of Digital Signal Processing (DSP) operations used in embedded code while supporting the integer multiply instructions in the Cold Fire microprocessor family. Modern computers may contain a dedicated MAC consisting of a multiplier implemented in combinational logic followed by an adder and an accumulator register that stores the result. The output of the register is feedback to one input of the adder though on each clock cycle the output of the multiplier is added to the register. The Combinational multipliers needed a large amount of logic but can also compute a product much more quickly than the method of shifting and adding typical of earlier computers.

Fig 5.Possible Hardware using MAC

Fig 6. Hardware & Technology view using mac:-

Fig.7 simulation result for FIR filter using MAC

Table II.- Synthesis report of MAC

Device Utilization Summary (estimated values)				
Logic Utilization	Used	Available	Utilization	
Number of Slices	36	960	3%	
Number of Slice Flip Flops	21	1920	1%	
Number of 4 input LUTs	62	1920	3%	
Number of bonded IOBs	27	66	40%	
Number of GCLKs	1	24	4%	

A. DISADVANTAGES OF MAC

- 1) MAC is an old technique which cannot be destroyed in data path circuit designing.
- 2) Area required for MAC in Digital Signal Processing hardware designs is too large.
- 3) Efficiency of computing mechanism is low.

The second type of multiplier-less method involves use of memories (RAMs, ROMs) or Look-Up Tables (LUTs) to store pre-computed values of coefficient operations. These memory-based methods involve Constant Coefficient Multiplier method and the very-well known Distributed Arithmetic method as examples.

IV. DISTRIBUTED ARITHMETIC (DA)

DISTRIBUTED ARITHMETIC (DA) is so named because the arithmetic operations that appear in signal processing (e.g., addition, multiplication) are not “lumped” in a comfortably familiar fashion, but are distributed in an often-unrecognizable fashion. Distributed Arithmetic (DA) plays a key role in embedding DSP functions in the Xilinx 4000 family of FPGA devices. The most-often encountered form of computation in digital signal processing is a sum of products (or in dot-product, inner-product generation, or vector analysis parlance).. The advantages are best exploited in circuit design, but off-the-shelf hardware often can be configured effectively to perform DA. By careful design one may reduce the total gate count in a signal processing arithmetic unit by a number seldom smaller than 50% and often as large as 80%.

Fig 8. Block diagram of DA

Fig 9. : LUT-based DA implementation of a 4-tap FIR filter

Fig.10 Technology view using DA

Fig.11 DA Simulation waveform

V. APPLICATIONS OF DA

- 1) Whenever the performance/cost ratio is critical (especially in custom designs), DA should be seriously considered as a contender.
- 2) FIR filters are used in digital communications in the receiver.
- 3) This is used in software radio It easily adjustable filters, with good rejection, without replacing hardware.
- 4) FIR filter is used in digital cameras.
- 5) FIR filter can be directly connected to memory because output of memory is in digital form.
- 6) This is used in software radio It allows easily adjustable filters, with good rejection, without changing hardware.

VI. CONCLUSION

This technique reduces the delay, area, power consumption. The performance can be further improved by pipelining all the partial tables. This architecture provides an efficient area-time power implementation which involves significantly less latency and less area-delay complexity when compared with existing structures for FIR Filter. Hence the power consumption is 50% less than that of MAC. As the order of the fir filter is increased the precision capability of the filter gets improved. As in this paper we have implemented 4-tap FIR filter it gives 4 bit precision value for 8 bit input data. So in future, we're going to design 8-tap and 16-tap fir filter based on same filter.

REFERENCE

- [1].chen," Low-cost fir filter designs based on faithfully rounded truncated multiple constant multiplication/accumulation", in proceedings of IEEE Transaction on circuit and systems-II: vol.60, no. 5 may 2013.
- [2].Sang yoon park, pramod kumar meher," low power, high throughput, and low area adaptive fir filter based on distributed arithmetic", in proceedings of IEEE Transaction on circuit and system 1549-7747- feb 14,2013.
- [3].Bahram rashidi, majid pourormazd," design and implementation of low power digital fir filter based on low power multipliers and adders on Xilinx fpga," in proceeding of IEEE Transaction 978-1-4244-8679-3 ,2011.
- [4]. " High-speed FIR Digital Filter with CSD Coefficients Implemented on FPGA" by "Mitsuru Yamada Akinori Nishihara".
- [5]. "Applications of distributed arithmetic to digital signal processing: a Tutorial review," ieee assp magazine, july, 1989.