IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 2 Issue 5, May 2015.
www.ijiset.com
ISSN 2348 – 7968

Design and Implementation of Partial Reconfigurable Fir Filter Using Distributed Arithmetic Architecture

Vinaykumar Bagali[1], Deepika S Karishankari[2]

[1]Asst Prof, Electrical and Electronics Dept, BLDEA'S College of Engineering, Bijapur, Karnataka, India

[2]Electrical and Electronics Dept, BLDE'S College of Engineering, Bijapur, Karnataka, India

Abstract

The design of partial reconfigurable FIR filter using systolic distributed arithmetic architecture. A two dimensional fully pipelined structure is used to implement low power, high speed, computationally efficient FIR Filter. Look-Up-Table (LUT) a new architecture in distributed arithmetic is planned to reduce the partial reconfigurable time. In partial reconfiguration module, by changing the filter coefficients the FIR filter is dynamically reconfigured to realize the low pass and high pass filter characteristics. Using XUP Virtex 5 LX110T kit the design is implemented. FIR Filter design illustrates improvement efficiency and time

Keywords*-Distributed Arithmetic, FIR, Dynamic Partial Reconfiguration, FPGA, Systolic architecture*

1. Introduction

The most fundamental component in digital signal processing is Finite Impulse Response (FIR) those are mainly implemented on hardware rather than software for high speed computation. It is necessary to design reconfigurable filter architecture based on power or resources consideration, the low power high speed implementation of FIR Filter design in various embedded application is required.

One of the challenges in partial reconfigurable architectures is the reconfigurable overhead, which is the time spent for reconfiguration on the fly because the computational complexity and reconfiguration time of the FIR filter increases with the increase in filter order and type of arithmetic used. In the literature several efficient hardware architectures are developed using non reconfigurable/reconfigurable architectures. Several multiplier with less schemes have been proposed in the literature [2, 3.4,7], As multipliers consume more power in Multiply and Accumulate (MAC) operation. One of the method i.e. multiplier less technique uses memories (RAMs, ROMs) or LUTs to store pre-computed values of coefficient operations is Distributed arithmetic. Although DA algorithm show a good set of characteristics with respect to speed and chip area the LUT of arithmetic used. Complexity increases with respect to number of filter coefficients thus the memory requirement increases which greatly reduce the speed. All these designs have a common feature that they consist of output network adders which will not support pipelining. The architecture in [8] proposed a DA architecture which uses the concept of systolic arrays and avoids the output adder network by introducing pipelining. A reconfigurable FIR filter has been proposed earlier in [7,8] uses Xilinx reconfiguration tools on Virtex FPGA to achieve reconfiguration. But the design uses output adder network for the architecture in the static region of the FPGA. But the reconfiguration head (analogy to the size of the partial bit stream file in KB) is almost same in both cases. It is also shown in [8] that reconfiguration time depends on the size of the bit stream file. It utilizes the approach of dynamically reconfiguring the coefficients via LUTs using modular reconfiguration scheme. Self reconfigurable adaptive FIR filter in partial re-configurable platform is also proposed in the literature [8]. A systolic based DA architecture with dynamic reconfigurable module to reconfigure the filter coefficients is proposed. It is based on dynamically reconfiguring at the finest possible level, the LUTs that store the coefficients, with a small dynamic reconfiguration area. Here, we propose a new architecture for a LUT. in DA. By using this architecture the size of the *.BIT* file needed to upload is reduced to a great extent as compared to

ISSN 2348 – 7968

[9]. The further section is structured as follows. Section II explains the basics of DA and systolic Architecture for FIR filter. Section III presents the proposed Partial Reconfiguration Module and in section IV Implementation results and discussion. Thus conclusions with section V.

2. Systolic DA architecture

We briefly outline here the conventional distributed arithmetic approach for inner-product computation, and thereafter derive a decomposition scheme for flexible DA based systolic FIR filters. An FIR filter can be described by the following equation

$$Y[n] = \sum_{i=0}^{n-1} x[n-1]c[i] \quad (1)$$

This is nothing but the inner product of inputs delayed each by a specific value with filter coefficients. The input value x(n) can be expressed in the form of corresponding bits .

$$X[n] = \sum_{b=0}^{B-1} X_b[n]X2^b \quad (2)$$

Substitution equation (1) in equation (2)

$$Y=\sum_{n=0}^{N-1} c[n] \times \{\sum_{b=0}^{B-1} 2^b \times x_b(n)\} \quad (3)$$

Equation (3) is rearranged as,

$$Y=\sum_{b=0}^{B-1} 2^b \times \{\sum_{n=0}^{N-1} c[n] \times x_b\ (n)\} \quad (4)$$

The term inside the braces in equation (4) is the sum of products of filter coefficients with the bits of inputs. If the number of input bits is N, then the sum of products can have 2N values. All the possible values are stored in a LUT (look up table). The corresponding bit vector from the input act as input to the LUT. The outputs from the LUT are taken for all the *b* bits and are shift added to get the output. This can be represented in below equation.

$$Y=\sum_{b=0}^{B-1} 2^b \times \{f\{x_b[0], x_b[1], x_b[2] \ldots, x_b\ [n]\}\}$$

where the term *f(xb[.])* in above equation is the value from LUT, according to input bits. So instead of admders and multipliers and registers (to hold temporary results) we use a LUT (a memory unit typically a ROM) and a set of shift adders equal to the number of bits that are used for representation. The computation of MAC operation is very fast. The LUT should be of size 2^N where N is the order of filter. As the order of filter increases the LUT size increases exponentially and the time taken for memory fetch also increases which affects the operational frequency of the FIR filter. So we used a systolic decomposition technique to reduce the size of LUT by using multiple LUTs.

Suppose if the order of the filter N is a composite number which can be obtained by product of two other numbers M and P, then we decompose the 2^N LUT into P LUTs of size 2^M . The equation (4) becomes,

$$Y= \sum_{b=0}^{B-1} 2^b \{\sum_{p=0}^{p-1}[\sum_{m=0}^{M-1} c(n) \times x_b(n)]\} \quad (5)$$

Using systolic array implementation for efficient mapping of equation (5) onto FPGA hardware. Systolic arrays (SA) are examples of VLSI special purpose processor networks that directly implement computationally expensive algorithms in hardware [2]. Systolic arrays are a systematic arrangement of small cells called processing elements where each processing element performs a simple task such as addition, multiplication, memory fetch etc and passes the data. The processing elements are simple finite state machines which generally perform simple tasks that generally does not consume more than a few clock cycle [8]. For the DA FIR filter our PE should contain a LUT and a memory fetch unit. The LUT results as per concept are to be added with the LUT fetches from other LUTs. So adder comes into the requirement. And finally there is a shift addition operation on the results from different bit positions.

3. Proposed DA LUT architecture

The partially reconfigurable FIR filter is designed using systolic DA architecture. The block diagram of the proposed 9 tap FIR filter is shown Fig 3.The trapezium shapes indicate shift adder, while the triangle is adder and DA LUTs are present. The reconfiguration partition in [8] consists of LUT used in DA architecture. Here we have come with an architecture in which the reconfiguration part consists of only filter coefficients but not the entire LUT as in [8]. By this we reduce the area of reconfigurable partition. We use FPGA resources for further computation of LUT entries. The time taken for

ISSN 2348 – 7968

computation, is less compared to reconfiguration time.

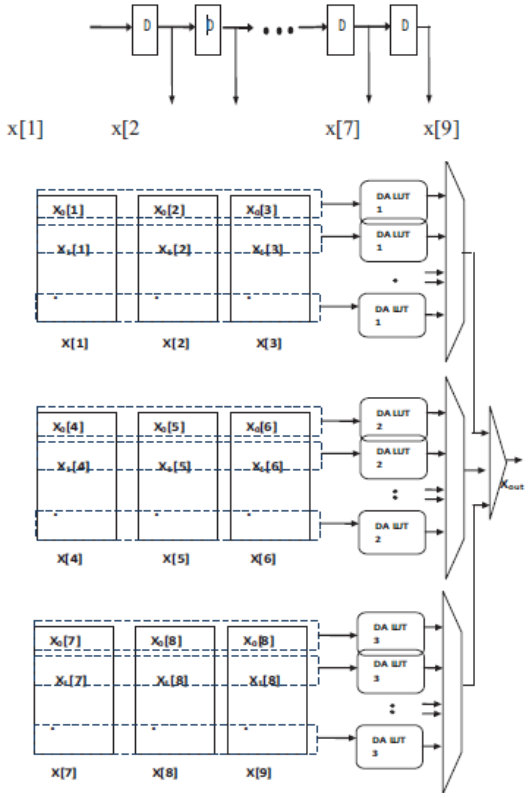

Fig 1.Systolic DA FIR Filter

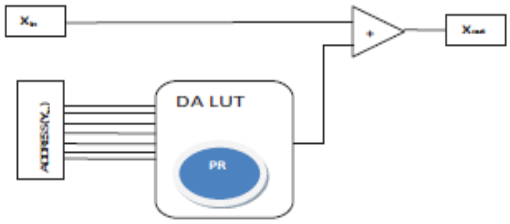

Fig 2. Processing Element 'A' architecture

Fig 2. shows the architecture of processing element A. It contains a LUT which is referred as DA LUT. The reconfigurable part is shaded in blue. The architecture of DA LUT is shown in Fig 3. This modified architecture of LUT which contains registers, adders and a memory. Memory is basically a RAM. In the design followed by D. [8] the RAM is placed in the reconfigurable partition which reconfigures the DA coefficients. In this proposed design the reconfigurable partition is modified to contain only filter coefficients rather than DA coefficients. The coefficients are obtained from the registers. The LUT values are then calculated by the network of adders and then are updated into the LUT RAM. The RAM has inputs address, write enable and read enable. This offers some advantages in speed of reconfiguration.

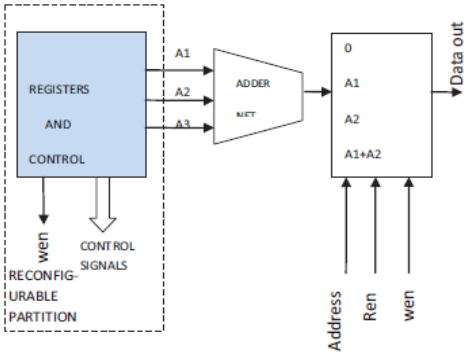

Fig 3.DA LUT architecture

Whenever the reconfigurable area is reconfigured, a set of control signals and write enable are generated. When reconfigured the some variables are assigned to their initial values**.** The control signals mainly consist of write enable and other enable signals. The write enable signal enables writing data into the RAM. Then the values in the LUT RAM are changed. The system functions now as a different filter. The advantage of this architecture over the existing architecture is that the reconfigurable area has been exponentially decreased. So, the partial BIT file has its size reduced to a great extent. Therefore the time taken to reconfigure the FPGA is greatly reduced.

4.Implementation and Results

The hardware used for developing a partial reconfigurable FIR filter is Virtex-5 FPGA on LX110T evaluation platform provided by Xilinx.inc and developed by Digilent.inc. The software used for developing and prototyping of the design are Xilinx System Generator, MODELSIM, Xilinx ISE 13.2 , PlanAhead[10]

A filter of order 9 has been designed. All the numbers are represented in fixed point representation of 12 bits and binary point at 11. Following the proposed reconfiguration scheme where only a module containing coefficients is reconfigured. Two reconfigurable modules have been created for testing.

ISSN 2348 – 7968

In one module we used the coefficients of a low pass filter and in the other module we have implemented a high pass filter. The filter coefficients are obtained from MATLAB FDA tool. The model is tested for an image. The Image is uploaded into a Read only Memory in FPGA provided by Core Generator and design is implemented and run with two different modules without switching off the device and the outputs are verified.

The size of the BIT file obtained was of 5KB, which is small when compared to the BIT file size obtained by [8]. This is shown in figure 4.

Fig 4.BIT file size

As the order of the filter increases the BIT file size obtained by [8] increases at high rate. But here the size of the file remains almost unchanged.

The functioning of the FIR filter is verified by testing the FIR filter on images. Fig 5 shows the original image, its low pass filtered image and high pass filtered image respectively.

Fig 5 (a) Input image (b)Low pass filtered image (c) High pass filtered image

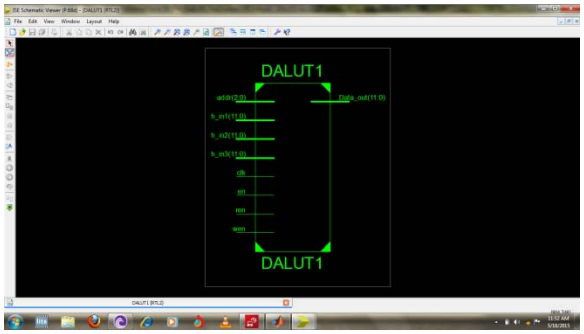

Fig 6. Synthesize of single DALUT

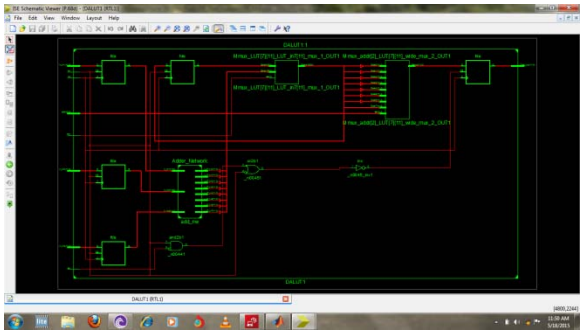

Fig 7.RTL Schematic of DALUT

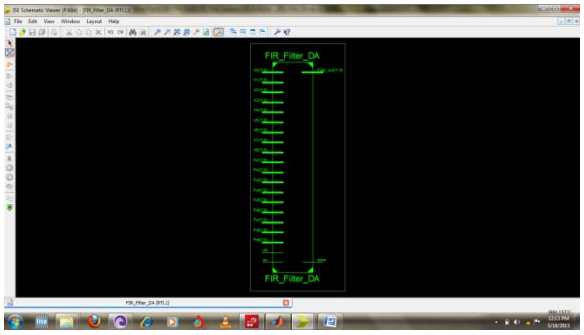

Fig 8. Main module synthesize

ISSN 2348 – 7968

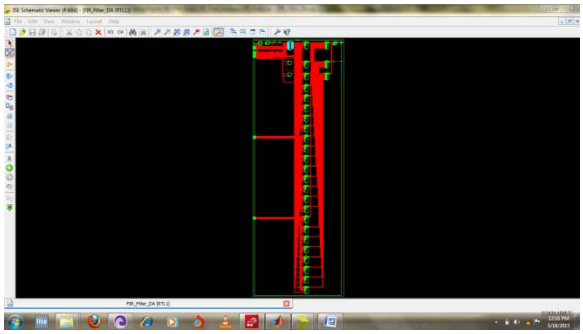

Fig 9. Main module Schematic

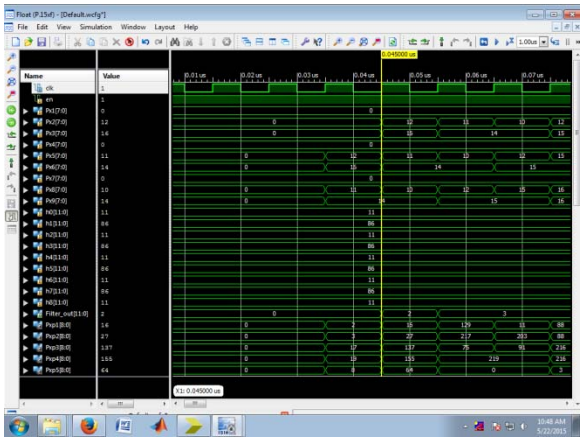

Fig 10.Low pass filter

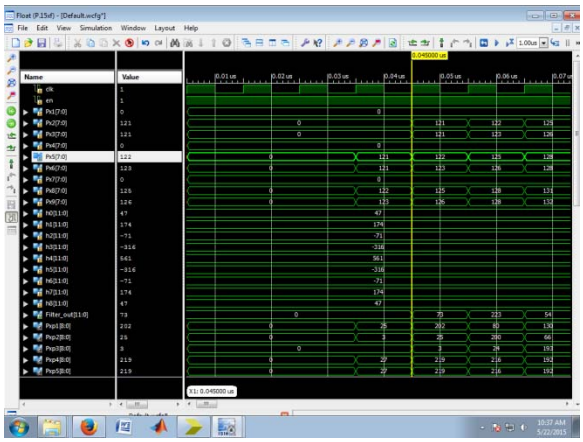

Fig 11. High pass filter

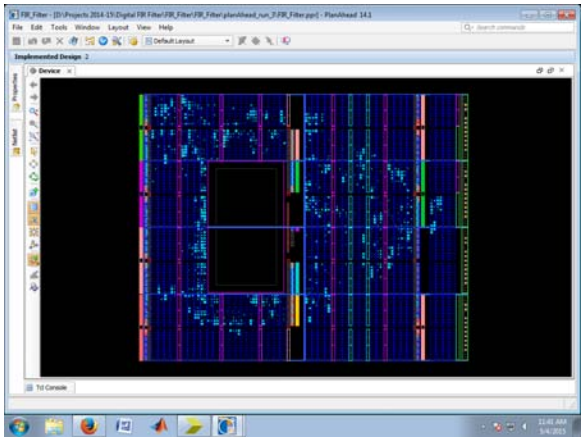

Fig 12.Floorplan Design

5. Conclusion

By Xilinx Virtex-5 FPGA a PR FIR Filter using systolized DA architecture has been implemented. With respect to speed and maximum frequency of operation the FIR Filter is optimized. The filter coefficients can be changed amid the operation using PR. The size of the .BIT file used for PR has been reduced. Since the size of .BIT is directly proportional to reconfiguration time, the time taken by PR is reduced to a great extent by using the proposed architecture.

References

[1] R.Wyrzykowski and S. Ovramenko, “Flexible systolic architecture for VLSI FIR filters,” *Proc. Inst. Elect. Eng.—Compute. Digit. Techniques*, Vol. 139, No. 2, pp. 170–172, Mar. 1992.

[2] S. A. White, “Applications of the distributed arithmetic to digital signal processing: tutorial review,” *IEEE ASSP Mag.*, Vol. 6, No. 3, pp. 5–19, Jul. 1989.

[3] K. K. Parhi, *VLSI Digital Signal Processing Systems: Design and Implementation.* New York: Wiley, 1999.

[4] S.-S. Jeng, H.C. Lin, and S.M. Chang, “FPGA implementation of FIR filter using M-bit parallel distributed arithmetic,” in *Proc. IEEE Int. Symp. Circuits Systems (ISCAS)*, May 2006, p. 4.

[5] A. Peled and B. Liu, “A new hardware realization of digital filters,” *IEEE Trans. Acoust. Speech, Signal Process.*, Vol. 22, no. 6, pp. 456–462, Dec. 1974

[6] H. T. Kung, “Why systolic architectures?,” *IEEE Computer, vol. 15*,no. 1, pp. 37–45, Jan. 1982.

ISSN 2348 – 7968

[7] A. Croisier, D. J. Esteban, M. E. Levilion, and V. Rizo, "Digital filter for PCM encoded signals," U.S. Patent 3 777 130, Dec. 4, 1973.

[8] Daniel Llamocca,1 Marios Pattichis,1 and G. AlonzoVera2, "Partial Reconfigurable FIR Filtering System Using Distributed Arithmetic"International Journal of Reconfigurable Computing Volume 2010 (2010),.

[9] Pramod Kumar Meher, Shrutisagar Chandrasekaran, Abbes Amira, "FPGA Realization of FIR Filters by Efficient and Flexible Systolization Using Distributed Arithmetic", *IEEE Trans on Signal Processing*, Vol. 56, No. 7, July 2008.

[10] Partial Reconfiguration User Guide, www.xilinx.com