High Speed FPGA Implementation of FIR Filter for DSP Applications

Rakhi Thakur and Kavita Khare

Abstract—Signal processing ranks among the most demanding applications of digital design concepts. It is a mature technology domain wherein the demands for enhanced performance and reduced resource utilization have risen exponentially over the years. Field Programmable Gate Array (FPGA) design technology has becoming the preferred platform for evaluating and implementing signal processing algorithms. The advantages of the FPGA approach to digital filter implementation include higher sampling rates than are available from traditional DSP chips, lower costs than an application specific integrated circuit (ASIC) for moderate volume applications, and more flexibility than the alternate approaches. Since many architectures current FPGA in-system are programmable, the configuration of the device may be changed to implement different functionality if required. This paper describes an approach to the implementation of digital filter based on field programmable gate arrays (FPGAs) which is flexible and provides performance comparable or superior to traditional approaches, lowpower, area-efficient re-configurable digital signal processing architecture that is tailored for the realization of arbitrary response Finite impulse response (FIR) filters.

Index Terms—FIR Filter, FPGA, DSP chips.

I. INTRODUCTION

A Filter is frequency selective network, which is used to modify an input signal in order to facilitate further processing. Basically there are two types of filters-analog and digital. Digital Filters are widely used in different areas, because Digital filters have the potential to attain much better signal to noise ratio than analog filters. The digital filter performs noiseless mathematical operations at each intermediate step in the transform and their precise reproducibility allows design engineers to achieve performance levels that are difficult to obtain with analog filters Digital filters operate on numbers opposite to analog filters, which operates on voltages. The basic operation of digital filter is to take a sequence of input numbers and compute a different sequence of output numbers. There exists a range of different digital filters. FIR and IIR filters are the two common filter forms. A drawback of IIR filters is that the closed-form IIR designs are preliminary limited to low pass, band pass, and high pass filters, etc. secondly FIR filters can have precise linear phase. Also, in the case of FIR

Manuscript received October 17, 2012; revised December 30, 2012.

Rakhi Thakur is with Kalaniketan Polytechnic college Jabalpur Madhya

filters, closed-form design equations do not exist and the design problem for FIR filters is much more under control than the IIR design problem. A FIR filter is a filter structure that can be used to implement almost any sort of frequency response digitally. It is usually implemented by using a series of delays, multipliers, and adders to create the filter's output. The architecture of FIR filter is shown in Fig 1.



Fig. 1. FIR Filter Architecture, (a) canonical form, (b) pipelined,(c) inverted form.

From this structure the transfer function of canonic form of the filter can be easily described in Z-domain as:

$$H(Z) = w_0 + w_1 z^{-1} + w_2 z^{-2} + \dots + w_{M-1} Z^{L-1}$$
(1)

In addition the advantages of FIR filter generally we use raised cosine pulse. The rectangular pulse occupies a large bandwidth so an alternative to rectangular pulse is a sinc pulse, which reduces the bandwidth and Inter Symbol Interference. The rectangular pulse is passed through the Root Cosine Filter a set of FIR filters to pulse shape the pulses to sinc [5]. If very high sampling rates are required, full parallel hardware must be used. Such filters can be implemented on FPGAS using combinations of the generalpurpose logic fabric, on-board RAM and embedded arithmetic hardware. Full-parallel filters cannot share hardware over multiple clock cycles and so tend to occupy large amounts of resource. Hence, efficient implementation of such filters is important to minimize hardware requirement [3].

Pradesh India(rakhi082003@yahoo.co.in) Kavita Khare is with Maulana Azad National institute of technology Bhopal India (kavitakhare1@yahoo.co.in)

II. FIR FILTER DESIGN TECHNIQUES

FIR filters are particularly useful for applications where exact linear phase response is required. The FIR filter is generally implemented in a non-recursive way, which guarantees a stable filter [2]. FIR filter design essentially consists of two parts:-

A. Approximation Problem

The approximation stage takes the specification and gives a transfer function through four steps. They are as follows:

- 1) A desired or ideal response is chosen, usually in the frequency domain.
- 2) An allowed class of filters is chosen (e.g. the length *N* for a FIR filters).
- 3) A measure of the quality of approximation is chosen.
- 4) A method or algorithm is selected to find the best filter transfer function.

B. Realization problem

The realization part deals with choosing the structure to implement the transfer function which may be in the form of circuit diagram or in the form of a program. There are essentially three well-known methods for FIR filter design namely:

- 1) The window method
- 2) The frequency sampling technique
- 3) Optimal filter design methods

III. FIR FILTER ADD/SHIFT IMPLEMENTATION

In binary arithmetic, multiplication by a power-of-two is simply a shift operation. Implementation of systems with multiplications may be simplified by using only a limited number of power-of-two terms, so that only a small number of shift and add operations are required [4].

An FIR filter tap as shown in Fig.2 can be implemented in two array columns of Xilinx series FPGAs. Because of the high degree of spatial and temporal locality, most signal routing delays are not critical, as they are with typical high performance FPGA designs. Each of the bit slices for the tap requires two combinational logic blocks (CLBs) in the array for implementation. The extensive local routing capability of typical FPGAs can be used for the majority of signals within and between taps.

The implementation is based on a Xilinx board . This board has the following features, relevant to the presented implementation [5]:

- 1) Spartan3 FPGA with 500000 equivalent gates (XC3S5500E),
- 2) 50MHz crystal oscillator,
- 3) Asynchronous serial port, with RS232 drivers,
- 4) Expansion connector with 100 I/O pins,
- 5) Flash memory for bit stream storage,
- 6) USB port for FPGA configuration and memory programming.

The scheme for the implementation of FIR filter to FPGAs is shown in Fig. 2. The main components of the implemented circuit are as follows [1]:

A. Memory

Prepared for storage of past position data of bunches. z⁻¹

in Fig. 1means 1-turn delay.

B. Adder

Adder is made to reduce the number of stages and is a key for stable operation of the FPGA used in the board. This reduction of the stage is effective to avoid errors by clock skew in the FPGA and to reduce the power consumption and a circuit area(or number of gates) on FPGA.

C. Multiplier

Build-in multipliers are used to fulfill the requirements of high-speed operation; therefore this number of build-in multipliers is one of the constraints to the number of taps of FIR filter.

D. Shift-Register

It is used for additional delay for adjust latency to one or two revolution period.



Fig. 2. Implementation of FIR filter.

V. FPGA IMPLEMENTATION

Advances in field programmable gate array technology have enabled FPGAs to be applied to a variety of problems. In particular, FPGAs prove particularly useful in data path designs, where the regular structure of the array can be utilized effectively. The programmability of FPGAs adds flexibility not available in custom approaches, while retaining relatively high system clock rates. The disadvantages of FPGAs are primarily related to the limited number of logic operations that can be implemented on a particular device, and the limited signal routing options that are available for connecting logical operators on the array.

The hardware description language VHDL is very popular among designers. One reason is good support of integrated circuit design by offering many integrated circuit related function and Data types. There is also large number of libraries available. It supports the behavioral modeling of hardware necessary when implementing a Fir filter generation program.

Reducing flip-flop count through minimizing multiplier logic depth has instead been shown to yield the lowest area solutions. The results presented establish a clear low area. Total memory usage is 147920 kilobytes and Minimum period is 4.255ns (Maximum Frequency: 235.026MHz)



Fig. 3. RTL view of FIR Filter

TABLE I: FPGA RESOURCE UTILIZATION OF MULTIPLIER BLOCK

Resource	Used	Available	Utilization
Slice	144	9312	1%
LUTs	92	9312	1%
Occupied Slices	84	4656	1%
Related Logic	84	84	100%
I OBs	46	92	50%
BUFGMUXs	1	24	4%

VI. CONCLUSION

There is a constant requirement for efficient use of FPGA resources where occupying less hardware for a given system. In this paper we presented a multiplier less technique, based on the add and shift method. It can perform at sample rates greatly exceeding those of a state-of-the-art programmable DSP. The FPGA solution offers complete

flexibility in the design. By reducing chip count, it improves the overall reliability of the system, provides low area, low power and high-speed implementation of FIR filters. It also reduces filter latency.

REFERENCES

- [1] S. Mirzaei, A. Hosangadi, and R. Kastner, FPGA Implementation of High Speed FIR Filters Using Add and Shift Method, 2006
- [2] K. N. Macpherson, "Low FPGA area multiplier blocks for full parallel FIR filters," in *Proc. IEEE International Conference on Field-Programmable Technology*, pp. 247 – 254, Issue Date: 6-8 Dec. 2004
- [3] M. Mehendale, S. D. Sherlekar, and G. Venkatesh, "Techniques for Low Power Realization of FIR _flters," ASP-DAC 1995.
- [4] H. Samueli, "An improved search algorithm for the design of multiplierless FIR filters with powers-of-two coefficients," *IEEE Transactions on Circuits and Systems*, pp. 1044-1047, July 1989,
- [5] P. V. Rao, Cyril Raj Prasanna, and S. Ravi, "Design and ASIC Implementation of Root Raised Cosine Filter," *European Journal of Scientific Research*, ISSN 1450-216X, vol. 31, no. 3, pp. 319-328, 2009, Xilinx Inc.,



Rakhi Thakur completed her graduation in Electronics and Tele-communication in 2002, and post graduation in Microwave Engineering in 2005 from R.G.P.V. University. She is a research scholar in MANIT, Bhopal. Her research interests are VLSI and Embedded System for Mixed applications. Earlier she was HOD of EC department in SRIST, Jabalpur but since April 2010 she in Govt. Polytechnic College Jabalpur.