IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Design and Implementation of Dynamic Partial Reconfiguration of Filters for Digital Signal Processing using FPGA Xilinx

Ms. Chaitali Suroshe
PG Student, MTech
Department of Electronics & Communication Engineering
Priyadarshini Bhagwati College of Engineering

Dr. N. K. Choudhari
Professor
Department of Electronics & Communication Engineering
Priyadarshini Bhagwati College of Engineering

Abstract: ***This paper presents an implementation of recent techniques and energy efficient architectures which is partially reconfigurable FIR filter design that employs dynamic partial reconfiguration. The system is that much flexible that it can fix the hardware and software at the time of fabrication also. Such a feature is thought as reconfiguration. Even at final stages of the tactic, this reconfiguration allows the user to contain more control over the hardware and software. Such kind of reconfiguration is feasible in Field Programmable Gate Array (FPGA). Modern FPGA is a reconfigurable device which provide advanced capabilities which enable the creation of embedded systems. It consists within the modification of form of the circuitry mapped on the FPGA while the system is running. The proposed system handles changes within the needs and operational conditions because of its ability that it allows the flexible systems expansion which makes the system more flexible. The system is to implement a low-power consumption device that the power cannot consume more and save the power. The other aim for implementation of this system is area-efficient reconfigurable digital signal processing architecture that it modifies for the conclusion of the varied response or the output of finite impulse response filters using Xilinx FPGA. The design addresses implementation area efficiency and adaptability allows the dynamically inserting and/or removing the partial modules or partial device to implement the partial reconfigurable FIR filters with different types.***

Keywords: ***FIR, FPGA, SoC, LUT***

I. INTRODUCTION

A partially dynamic reconfigurable FIR filter design of this paper aims to meet all the objectives like low-power consumption, autonomous adaptability/reconfiguration ability, fault-tolerance, etc. on the Xilinx Virtex-4 FPGAs. The FPGA allow the implementation of digital systems and the FPGAs are the programmable logic devices. FPGA has many systems which can only be statically configured not be dynamically and partially. There is a necessity to check the system execution at the time of replacement reconfiguration so that reconfigure the device it another time. Dynamic Partial Reconfiguration allows the particular a component of FPGA device which be modified or we'll say that it makes partially or minor changes while the rest of the device or system continues to regulate and without tormented by the reprogramming. As we know that the current FPGA architectures require measuring of reconfiguration times in milliseconds. FIR filters are employed the digital signal processing in major, where these filters are supported electronic systems. In static reconfiguration, the device configures completely before system executed not after that because of its property. The exposure applications which is on demand like image, audio processing and video processing and coding, sensor filtering, etc. this make it essential for the reconfigurable architectures to style it. Some limitations linked with run-time reconfigurable design of upper order tap FIR filters using standard FPGA design techniques. one amid the main limitations is that the so called reconfigurable overhead, which is that the time spent for reconfiguration. this relies on the reconfigurable device and also the tactic of reconfiguration.

The finite impulse response filter could also be a special reasonably digital filters and contains a large applicability

due to its good characteristic like linear phase and stability. This winds up in an outsized number of area (slice) usage for FPGA design. The partial reconfiguration technique is going to be employed during this case because the various taps FIR filters have such an outsized amount of similarities in their structure. Therefore, the partial reconfiguration takes guarantee to chop back reconfiguration overhead, coefficient flexibility and area efficiency for higher order FIR filters. To configure the device completely before execution is barely happens in static reconfiguration. If reconfiguration is required, it's essential to forestall system execution and reconfigure the system everywhere again. Several FPGAs support performing partial reconfiguration, which reconfigures only a given subset of internal modules but the Dynamic Partial Reconfiguration (DPR) sanctions the required module of FPGA be modified or to create minor changes while the rest of the system operates without getting affected. the partial reconfiguration was proposed by Xilinx which allows the designer to divide the complete system into modules. Multiplication may be a crucial fundamental operation and its applications like multiply and accumulate (MAC) are implemented in Digital Signal Processing (DSP) applications like DFT, Finite Fourier Transform (FFT), convolution, etc. there's necessity for prime speed multiplier, due to multiplication increases the execution time of most digital signal processing algorithms. Reducing the ability consumption and time delay are some crucial requisites for several applications. Minimizing power consumption for digital designs involves optimization at every expertise of the design. This optimization depends on the technology employed in implementing the digital circuits. the $64000 delay of processing a function is barely refers by the latency, a measure of stability of the inputs to a tool and throughput are going to be defined because the measure of what number multiplications are going to be drained a given period of some time. Multiplier isn't only a high delay structure but a big source of power dissipation. to chop back power utilization, it is important to chop back the delay by utilizing different delay optimizations.

FPGA has many systems which can only be statically configured not be dynamically and partially. In static reconfiguration, the device configures completely before system executed not after that because of its property. The exposure applications which is on demand like image, audio processing and video processing and coding, sensor filtering, etc. all this is essential to form the reconfigurable architectures. Some limitations linked with run-time reconfigurable design which uses the standard FPGA design techniques that uses upper order tap FIR filters. one amid the main limitations is that the so called reconfigurable overhead, which is that the time spent for reconfiguration. this relies on the reconfigurable device and also the tactic of reconfiguration.

II. The Proposed Work

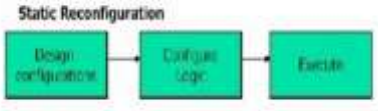
This proposed system contains of five different modules that are as follows: 1) Finite impulse response filter which is dynamically reconfigurable. 2) The multiplier of multi-precision; 3) The features a task features by the input operands scheduler (IOS) which is to reorganize the input file stream into a buffer and thereby decrease the vo transitions; 4) the operating frequency which is desired for the system and that of multiplier is produced by the frequency scaling unit; 5) The multi-precision multiplier has razor flip-flops and it might be report timing errors which is associated with not satisfied high voltage supply levels. the most objective of the proposed system is to style and implement a minimum delay reconfigurable filter design that mixes multi-precision multiplier with an error-lenient DFS technique supported razor flip-flops. The finite impulse response filters architecture's design relies on partial reconfiguration and output of finite impulse response filters calculates from a bunch of input samples. The group of input samples is multiplied by a bunch of coefficients then added together to get the output. The finite impulse response filters recognition will be exhausted the hardware in addition as in software. An implementation of software would involve sequential execution of the functions of filters while implementation of hardware of FIR filters allows the functions of filters to be processed or worked in an exceedingly parallel custom, that increase the processing speed but less flexible for changes. Thus, the proposed system offers the bendability of computer software in addition because the capability to make manner high performance systems.

III. Partial Reconfiguration

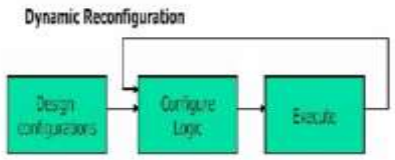
To reconfigure the logic of particular part of the system Partial Reconfiguration is used, where as other parts are operating normally. The partial reconfiguration is effective and it allows the designer to makeover or change variety of the devices of the system and thus it reduce power and improve system upgradability. To program the FPGA, we'll must change the bit-file. The file of bits is greater than 1MB sometimes it is greater than that also. so while reconfiguration we have to vary the bit-file, this is too hectic job so we decide partial reconfiguration for this system.

There are two kinds of reconfiguration which is classified into two static and dynamic reconfiguration.

- The Static reconfiguration is additionally called Compile time reconfiguration. it is the best form and most typically used sort of reconfiguration. The resources will remain from the initializing till the tip of operation. Throughout the operation the hardware resources remain static. So it's completely called static reconfiguration.



- The Run time reconfiguration is additionally called dynamic reconfiguration. It is one in every of the commonly used sort of reconfiguration and this system uses dynamic allocation of resources at run time. This will increase the performance of the system and use of optimized circuits which are loaded and unloaded dynamically during the operation. during this manner we are going to maintain the pliability of the system and by this functional density also increased.



IV. Block diagram

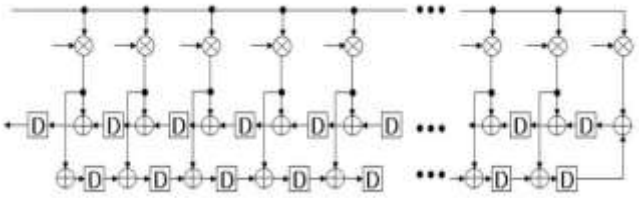


Fig.1. N-tap transposed FIR filter

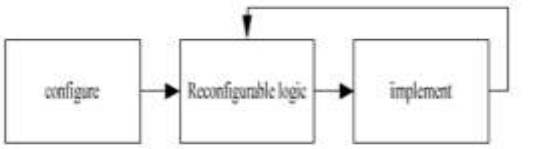


Fig 2. Dynamic partial reconfiguration

V. Operation

On adaptive systems, a limited factor for the unstipulated system Performance is usually the speed of which the system is in a position to transmute to perform a particular task. In this operation section describes how to implement the 20-tap FIR filter, which may partially reconfigure from 8-tap to maximum 20-tap FIR filter and the meaning of this is that this type of filter can extended from 8-tap to 20-tap. the full system is implemented on a Xilinx Virtex-4 FPGA. A HDL diamond unravelment and Synthesis Partial reconfiguration requires a hierarchical diamond tideway that has to be strictly followed during the HDL coding process. The initial step of the precision reconfiguration design flow is to define three types of HDL design description (i.e. Top-level design module, Base design module, partial reconfiguration design module) then integrate those HDL descriptions separately and these HDL design descriptions are combined to following three design modules. * Top-level design module: during this step, we must consider each sub-module interconnection using bus macro and area assignment. Top-level description must only contain I/0, Clock primitive, Base design, partial reconfiguration module, bus-macro instantiations and signal declarations. * Base design module: Base design module is static design module which suggests that base design module won't reconfigure whether or not partial reconfiguration is performed and hence this step is very similar to traditional HDL design method. * partial reconfiguration design module: partial reconfiguration module is additionally same to traditional HDL design method. However, partial reconfiguration module can exist two or more. Therefore, designer must describe multiple description of HDL.

VI. Design Constrains

After the HDL design description is synthesized, the subsequent step is to position constraint on the planning for place and route (PAR) and timing constraint to enhance the planning performance. Design constraints must have zone group, reconfiguration mode and bus macro location constraints. Implement Base Design First, the bottom design must be implemented. the data generated by implement base design is employed for partial reconfigurable modules implementation phase. Implement Partial Reconfigurable Modules After the bottom design is implemented; each PR modules must even be implemented. Each of the PR modules must be implemented separately. Merge the ultimate step within the PR design flow is to merge the highest, base, and PR modules. During the merge step, an entire design is made from each PRM and therefore the base design. during this step, many partial bit streams for every PRM and initial full bit streams are created to configure the FPGA. All FPGA devices is considered as stuff well-balanced of two layers: the configuration memory layer and therefore the hardware logic layer FPGAs succeed their uncommon or the special re-programmability and flexibility thanks to this composition. The hardware logic layer accommodates the computational hardware resources, together with the diamond up tables, flip-flops, digital signal processing blocks, memory blocks, transceivers, and other components. This layer also contains the routing resources and switch boxes that allow components to be connected to create a circuit. The configuration memory layer stores the FPGA configuration information through a computer file called a configuration file or bit stream. This computer file contains all the data that determines the implemented circuit, like the values stored within the search table, initial set and reset status of flip-flops, initialization values for memories, voltage standards of the input and output pins, and routing information for the programmable interconnect to enable the resources to create the described circuit. The hardware logic layer implementing function is thus completely determined by the values stored within the configuration memory. Most modern devices have volatile memory thanks to SRAM based configuration. to alter the circuit implemented within the FPGA, a user modifies the contents of the configuration memory by loading a brand new bit stream. The operation explained above is termed FPGA configuration or reconfiguration and is usually performed through the external FPGA interfaces like JTAG, or Select-Map on Xilinx devices. the whole configuration

memory is reloaded and therefore the FPGA remains inactive/inaccessible during this era. FPGAs built using non-volatile technologies don't seem to be designed to support such dynamic loading of the configuration memory. Partial reconfiguration refers to the modification of 1 or more portions of the FPGA logic while the remaining portions don't seem to be altered. However, the terms dynamic reconfiguration and partial reconfiguration are frequently used interchangeably within the literature, they'll diverge. The partial reconfiguration operation can either be static or is dynamic, this suggests that the reconfiguration operation can occur while the FPGA logic is in a very reset state (static) or running (dynamic). it's also not necessary that everyone dynamic reconfigurations are partial in nature. Partial reconfiguration is supported through external FPGA interfaces furthermore as special internal interfaces like the interior Configuration Access Port (ICAP) on Xilinx devices.

VII. Conclusion

In this paper, we present a dynamic partial reconfiguration for the FIR filter design. This is area efficient, flexible, low power consumption system whose configuration time sway allowing dynamically inserting and/or removing the partial modules ant time. In future, automatic partial reconfiguration of digital circuit is promising solution within the run-time environment when self-reconfigurable hardware platform using microcontroller unit and configuration memory for the proposed system. Today Reconfiguration chip will be utilized in satellites, modems, computer PCs etc. Electronic Odometer, Engine Monitor etc. are some ASIC products suitable for automobile applications. Engine Monitor and red light Controller is another ASIC product that monitors different parameters like temperature, voltage etc. of a vehicle. ASICs will be widely used for industrial application. in the field of technology based on FPGA, the partial reconfiguration plays an eminent role and the different partial reconfigurations are present to make a FIR filters but here we implement a dynamic partial reconfiguration for the system which is partially reconfigured from 8-tap to maximum 20-tap FIR filter and the full system is implemented on a Xilinx Virtex-4 FPGA. Different techniques employed in dynamic partial reconfiguration is discussed with each technique used for that particular application or the purpose.

REFERENCES

[1]. K. Anandan and N. S. Yogaananth, "VLSI Implementation of Reconfigurable Low Power Fir Filter Architecture", IJIRCCE, Vol.2, Special Issue 1, pp no 3514-. 3520, March 2014.

[2]. S. Karthick, Dr. S. Valarmathy and E. Prabhu," RECONFIGURABLE FIR FILTER WITH RADIX-4 ARRAY MULTIPLIER", jatit, Vol. 57 No.3, pp.326-336, Nov.2013.

[3]. Keinert, J., Deprettere, E.F.: Multidimensional data flow graphs. In: S.S. Bhattacharyya, E.F. Deprettere, R. Leupers, J. Takala (eds.) Handbook of Signal Processing Systems, second edition. Springer (2013)

[4]. Karol Desnos and Francesca Palumbo 22. Fradet, P., Girault, A., Poplavko, P.: Spdf: A schedulable parametric data-ow moc. In: 2012 Design, Automation & Test in Europe Conference & Exhibition (DATE), pp. 769{774. IEEE (2012)

[5]. Ersfolk, J., Roquier, G., Jokhio, F., Lilius, J., Mattavelli, M.: Scheduling of dynamic dataow programs with model checking. In: Signal Processing Systems (SiPS), 2011 IEEE Workshop on, pp. 37{42. IEEE (2011)

[6]. S.-R. Kuang and J.-P. Wang, "Design of power-efficient configurable booth multiplier," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 57, no. 3, pp. 568–580, Mar. 2010.

[7]. Martin Kumm, Konrad M¨oller and Peter Zipf "Dynamically Reconfigurable FIR Filter Architectures with Fast Reconfiguration", ieee journal of solid-state circuits, vol. 41, no. 4, April 2006.

[8]. Yeong-Jae Oh, Hanho Lee, Chong-Ho Lee, "A Reconfigurable FIR Filter Design Using Dynamic Partial Reconfiguration", IEEE, vol-06, pp. 4851–4854, ISCAS 2006.

[9]. Xilinx Inc.: Development System Reference Guide. www.xilinx.com.

[10]. Xilinx Inc.: XAPP 290: Two flows for Partial Reconfiguration: Module Based or Difference Based.www.xilinx.com, Sept. (2004).

[11]. Pramod Kumar Meher, Shrutisagar Chandrasekaran and Abbes Amira, "FPGA realizasion of FIR Filters by Efficient and Flexible Systolization Using Distributed Arithmetic", ieee transactions on signal processing, pp no-1-9.


IJCRT