International Journal of Engineering &Technology, 7 (3.12) (2018) 826 -830

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper



Design of Reconfigurable Block FIR Filter Architecture and Implementation on Hardware

Suji S[1], Radhika P[2]

[1]MTech VLSI Design, [2]Assistant Professor,
Department of ECE, SRM Institute of Science and Technology, Chennai, India.
**Corresponding Author Email: [1]suji2309994@gmail.com, [2]radhika.p@ktr.srmuniv.ac.in*

Abstract

In this paper, a reconfigurable block FIR filter which supports variable filter length is proposed. This recon-figurable block FIR filter uses block based design. Hence, this is an algorithm free architecture. This proposed filter can be used for 5G air interface.The proposed filter produces more efficient power reduction than that of the other filter.The number of LUTs and registers are also reduced in the reconfigurable block FIR filter. The designed filter has been implemented in the ZYNQ xc7020 hardware device using the vivado 2015.4.The technique used for hardware implementation is the IP creation and debug-ging.The debugging helps in the monitoring and triggering the hardware device.

Keywords: *Reconfigurable block FIR filter, Register unit, Inner product unit(IPU), Pipelined adder, xilinx vivado 2015.4, Zynq xc7020, Cadence.*

1. Introduction

Reconfigurable FIR filter can be used for many applica-tion such as SDR channelizer, filter bank multicarrier(FBMC) transmitter, some of the medical applications.This filter can be well suitable for wireless applications.

Multiple constant multiplication (MCM) is used which per-forms multiplication of input with the set of constants[1]. The MCM block is formed by using the transpose form configura-tion of the FIR filter.MCM performance matrix multiplication of the co efficient. The register can be reduced by drawing the flow graph. The transpose form structure is an area-delay efficient for the higher order FIR filters. But the block size and the filter length is fixed.

Optimized one and two dimensional fully pipelined structures are designed [2]. This structure is used for implementation of FIR by decomposition of Distributed Arithmetic. DA based lookup table (LUT) is designed to minimize the area time problem. Thus, the input samples are derived from the convolu-tion sequence and hence, the coefficient inputs is derived from the another sequences. Therefore, the distributed algorithm is mainly used for memory-based structures. Serial in parallel out registers are to receive the input sequence, where inputs to the register is serially right shifted and transferred in parallel at every clock cycle to the bit-serial word-parallel converter.This DA implementation is not suitable for large filter orders. This filter structure is implemented in the FPGA.

A modified delay least mean square adaptive algorithm is used to achieve lower adaptation delay[3]. Implementation done by pipelined inner product computation unit for feedback error calculation.

The distributed algorithm based circular convolution of the input sample and coefficient sample has been preformed. The LUT is designed in such a way to produce 2N words. It supports higher sampling frequency but occupies more area and higher power consumption. Hence, an 1D systolic array based digital convolution is proposed. Systolic array consists of L rows and hence each row consists of P nodes. The node uses sequence of input sample which gives the address of the LUT at that node. The value in the address of the node is read and hence the bits are added using the shift and add adder. To improve the throughput the digital convolution is performed by the 2D systolic array structure.

A distributed algorithm based recnfigurable FIR filter in which its coefficient input changes in the runtime[4].This uses sharing based LUTs for store the coefficient inputs.The Distributed algorithm based systolic structure is used for ASIC imple-mentation and the distributed RAM based design is used for FPGA implementation. The reconfigurable filter is further implemented in the Xilinx virtex board and compared with the ASIC implementation.

This reconfigurable FIR filter reconfigures the input samples for the each block and which is given to the inner product unit (IPU). Hence, matrix vector product operation is performed inside the each IPCs in the IPU unit. The coefficient inputs are generated using the coefficient storage unit (CSU).The output of each inner product units are together using the pipelined adder.

The filter is implemented in the zynq evaluation board. The hardware programming is done by using debugging of respec-tive input and outputs.The steps for the hardware program-ming are IP creation,synthesis,implementation and hardware programming.

The paper further is given has follows: Section II recon-figurable block FIR filter. Section III describes simulation and results.Section IV describes the comparison of different taps of reconfigurable

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

block FIR filter.Section V describes the hardware implementation. Section VI describes the conclusion. Section VII references.

2. Reconfigurable Block Fir Filter

The proposed re-configurable architecture is block based algorithm-free architecture. Re-configurable FIR Filter widely used in applications where the coefficients are changing dynamically at run time.

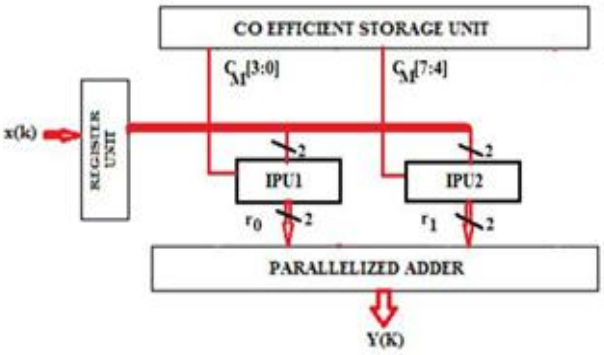

Fig. 1: Block diagram of reconfigurable filter

Here the filter length of L=2 is proposed and hence co-efficient varies from c_{M-1} to c0 where M is the number of IPU units which depends on the filter length L.

A. Register Unit

Register unit receives the inputs samples $X_k=(x_0, x_1....x_k)$ from the k number of blocks. This register block reconfigures the input samples using the (L-1) D flip flop.
For example, if input samples is 8 bit then the output will be 16 bits. The output bit size of the register unit is 2B where B is the number of bits in the input sample. The output of register unit is connect to the single inner product unit(IPU).

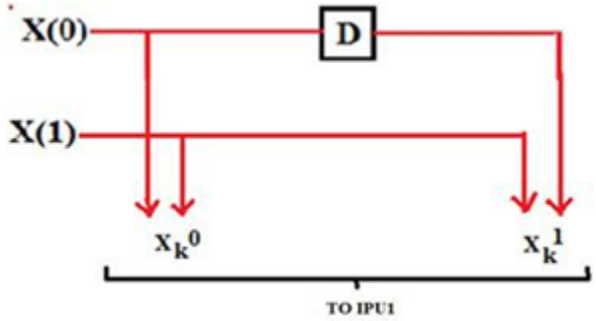

Fig. 2: Block diagram of register unit

B. Coefficient Storage Unit

The coefficient storage unit is used to store coefficient inputs [h(0); h(1):::h(k)] for any reconfigurable application. It is a ROM LUT. The coefficient input(h) and input samples x(k) can be received at the same clock period.

C. Inner Product Unit

The inner product unit receives reconfigured output from the register unit and also the coefficient input from the coefficient storage unit [h(0); h(1):::h(k)]. The number of inner product unit is equal to the size of the blocks L. Each IPU unit consist of L number of inner product computation (IPC)s. Each inner product computation unit receives B bits of input sample which is computed with the B bits of coefficient input.

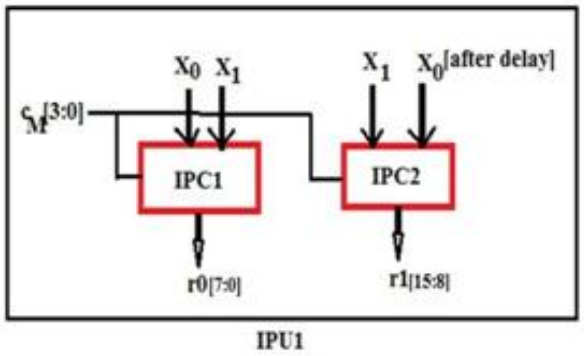

Fig. 3: Block diagram of inner product unit

Each IPC performs matrix vector product operation. Hence, each bit of input sample $[B_0,B_1....B_n]$ is multiplied with the respective coefficient input and added together.
r(k)=x(0)*h(0)+x(1)*h(1)+x(2)*h(2)+::::+x(k)*h(k).
where k varies from 0 to B-1.

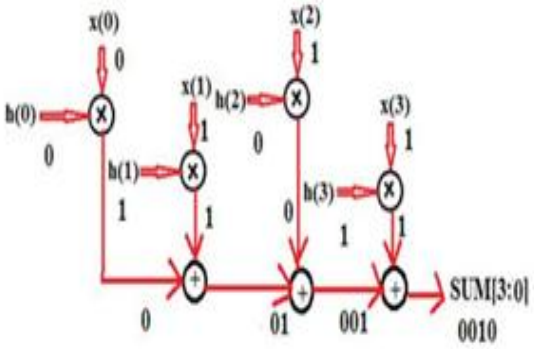

Fig. 4: Performance of inner product computation unit

D. Pipelined Adder

The pipelined adder is used to add the partial products from the IPU units .The number of D flip flop required is equal to the block size L.

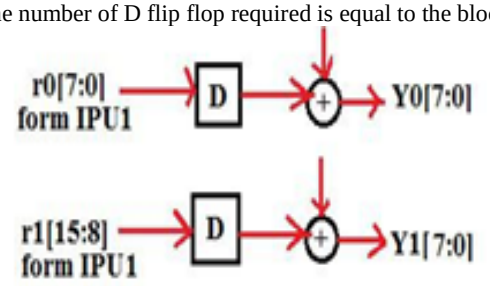

Fig. 5: Block diagram of pipelined adder

The delay associated with the architecture is given by,
$T = T_M + T_A + T_{FA} \log 2\ L$.
Where TM, TA and TFA refers to the delay due to the multiplier, half adder and full adder in single cycle.

3. Simulation and Results

The reconfigurable FIR filter was designed using the hardware description language verilog in Xilinx vivado 2016.4 and other parameters like power and area are been calculated in Cadence Encounter at 90 nano meter technology. To show the reconfigurability of the filter architecture, different filter lengths (8,16,32) has been taken.

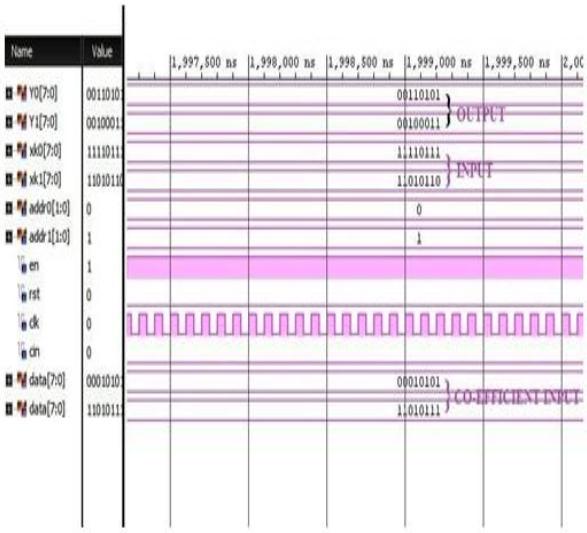

Fig. 6: Output of reconfigurable FIR filter for coefficient 8 bit

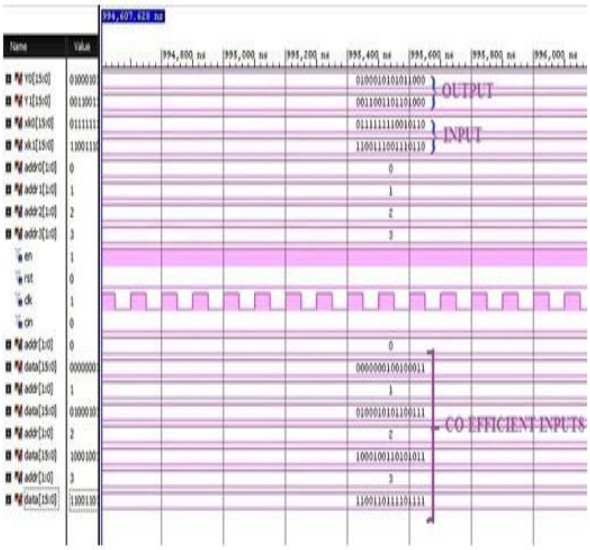

Fig. 7: Output of reconfigurable FIR filter for coefficient 16 bit

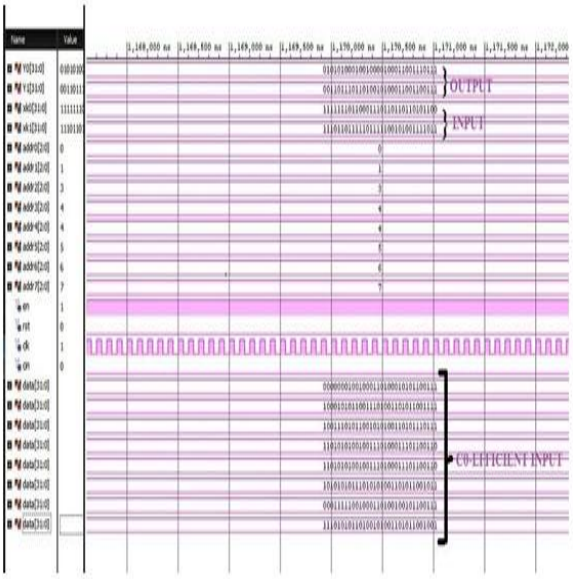

Fig. 8: Output of reconfigurable FIR filter for coefficient 32 bit

4. Comparison of Different Taps of Reconfigurable Block Fir Filter

The reconfigurable filter uses only single block architecture which supports different block size L. This tabulation gives the power, area, no of LUTs and registers required for the different taps of the filter for block size L=2.

Table 1: The Power, Area, No of Luts and Registers Required for the Different Taps of the Filter

Parameters	Power (mW)	No of LUTs	No of registers
RECONFIGURABLE FIR FILTER FOR 32 BITS	27.739	407	34
RECONFIGURABLE FIR FILTER FOR 16 BITS	17.435	254	17
RECONFIGURABLE FIR FILTER FOR 8 BITS	8.746	84	9

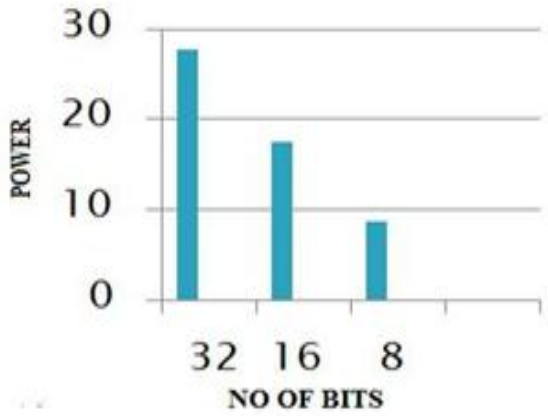

Fig. 9: Comparison of power

The graph gives the pictorial representation of power of different taps of reconfigurable FIR filters. The graph shows a linear decrease in power consumption.

5. Hardware Implementation

This reconfigurable FIR filter is programmed in the zynq xc7020 using the xilinx vivado.Zynq xc7020 is a general purpose evaluation SOC kit.The hardware manager in vivado is used for interface with the zynq device.Here, 4 tap filter has been taken for the hardware implementation. The steps involved in the hardware programming are:

A. IP Creation and Debugging

The debugging involves the creation of the IP, which cre-ates the probe for each input and output pins in the design.The debugged design helps in displaying and triggering the output form the board using hardware manager.The steps for this process is given in the form of a flow chart.

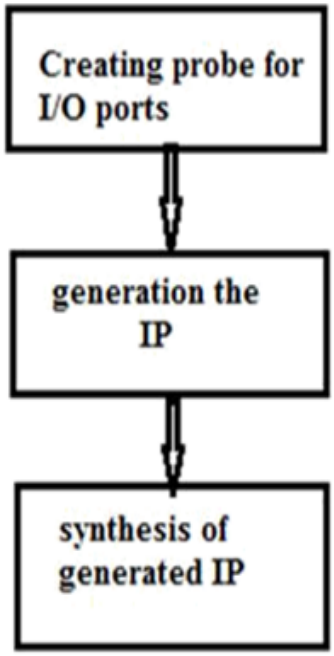

Fig. 10: Flow chart for IP creation and debugging

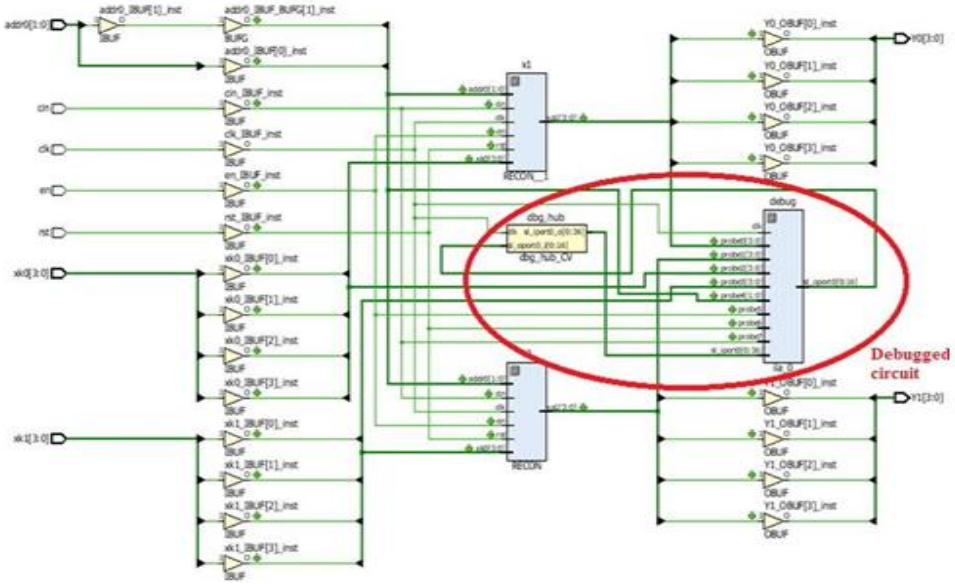

Fig. 11: Schematic of the filter with debugged circuit

The schematic shows the reconfigurable block FIR filter and the debugged circuit created by the IP creation and debugging.

B. Synthesis and Implementation

Synthesis involved in generating the netlist file.Netlist file consist of the details about input , output and wires. After synthesis, the input and output should be connected to their respective pins and should be provided with the required voltage. Inputs buttons or switches should be given with 1.8v and the output leds should be given with3.3v.
Then, the implementation has to be preformed which involve optimization, routing and placement. The main purpose of doing synthesis and implementation is to generate bit stream(.bit) file.This also gives the power, temperature produced during implementation, no of LUTs and registers used for the filter and the debug circuit.

Table 2: Implementation Result

POWER	0.127W
TEMPERATURE	26 C
THERMAL MARGIN	58 C
NEGATIVE SLACK	15.25ns

Name	Slice LUTs (53200)	Slice Registers (106400)	F7 Muxes (26600)	Block RAM Tile (140)	Bonded IOB (200)	BUFGCTRL (32)
XKL2RECON	1192	1835	25	1	22	3
dbg_hub (dbg_hub_CV)	392	622	1	0	0	1
debug (ila_0)	796	1211	24	1	0	0
x1 (RECON__1)	2	1	0	0	0	0
x2 (RECON)	2	1	0	0	0	0

Fig. 12: No of LUTs and registers after implementation

C. Hardware Programming

The generated bitstream file has been implemented into the zynq broad by using the hardware manager in vivado.The hardware device should be turned ON. Hence, the host device has been connected by connecting the USB cable to JTAG PROG connector.After programming the device,it is possible to monitor and trigger the output waveform.

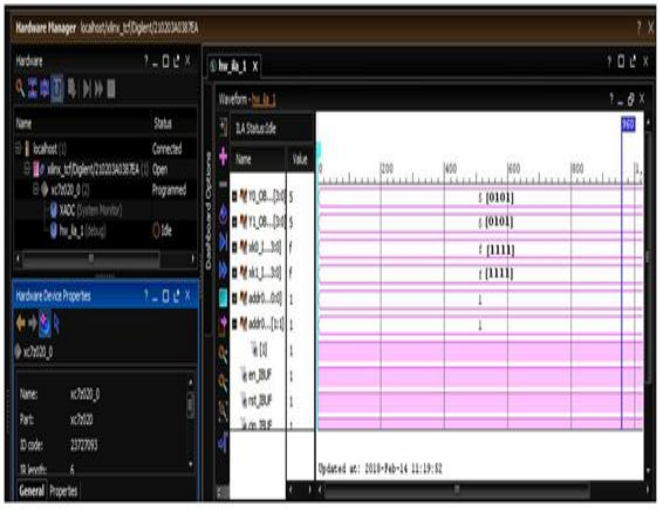

Fig. 13: Output after hardware implementation for 4 tap RFIR filter for block size L=2

6. Conclusion

In this paper, the reconfigurable FIR filter has been designed for a block size of L=2 and for different taps of 8,16,32 to show the reconfigurable nature of the filter.This filter is well suitable for wireless applications such as filter bank multi carrier transmitter.The main purpose of designing this reconfigurable block FIR filter for its variable filter length and its coefficient inputs can be changed dynamically at runtime. Thus, simulation results have been taken from the Xilinx vivado 2016.4. The filter is successfully implemented in the zynq xc7020 board using hardware manager. Thus the reconfigurable block FIR filter gives the better performance than that of the other filter.

References

[1] Basant Kumar Mohanty, Senior Member, IEEE, and Pramod Kumar Meher, Senior Member, IEEE High-Performance FIR Filter Architecture for Fixed and Reconfigurable Applications IEEE transactions on very large scale integration (vlsi) systems, vol. 24, no. 2, February 2016.

[2] Pramod Kumar Meher, Senior Member, IEEE, Shrutisagar Chandrasekaran, Member, IEEE, and Abbes Amira, Senior Member, IEEE. FPGA Realization of FIR Filters by Efficient and Flexible systolization Using Distributed Arithmetic. IEEE transactions on signal processing, VOL. 56, NO. 7, JULY 2008.

[3] Pramod K. Meher Department of Embedded Systems Institute for Info-comm Research, Singapore,IEEE. Megha Maheshwari School of Com-puter Engineering Nanyang Technological University, IEEE. A High-Speed FIR Adaptive Filter Architecture using a Modified Delayed LMS Algorithm 2011 IEEE.

[4] Sang Yoon Park, Member, IEEE, and Pramod Kumar Meher, Senior Member, IEEE Efficient FPGA and ASIC Realizations of a DA-Based Reconfigurable FIR Digital Filter IEEE transactions on circuits and systemsii: express briefs, VOL. 61, NO. 7, JULY 2014.

[5] Wonsuk Chung, Student Member, IEEE, Chanhong Kim, Sooyong Choi, Member, IEEE,and Daesik Hong, Senior Member, IEEE Synchronization Sequence Design for FBMC/OQAM Systems IEEE transactions on wireless communications, VOL. 15, NO. 10, OCTOBER 2016

[6] Kuan-Hung Chen and Tzi-Dar Chiueh, Senior Member, IEEE ”A Low-Power Digit-Based Reconfigurable FIR Filter” IEEE transactions on circuits and systemsII: EXPRESS BRIEFS, VOL. 53, NO. 8, AUGUST 2006

[7] Jongsun Park, Woopyo Jeong, Hamid Mahmoodi-Meimand, Student Member, IEEE, Yongtao Wang, Hunsoo Choo, and Kaushik Roy, Fellow, IEEE, ”Computation Sharing Programmable FIR Filter for Low-Power and High-Performance Applications” IEEE journal of solid-state circuits, VOL. 39, NO. 2, FEBRUARY 2004.

[8] Indranil Hatai, Indrajit Chakrabarti, Member, IEEE, and Swapna Banerjee, Senior Member, IEEE An Efficient Constant Multiplier Architecture Based on Vertical-Horizontal Binary Common Sub-expression Elimination Algorithm for Reconfigurable FIR Filter Synthesis IEEE transactions on circuits and systems I: Regular Papers, VOL. 62, NO. 4, APRIL 2015