Journal of Scientific & Industrial Research
Vol. 80, November 2021, pp. 971-978

Implementation of Area Efficient Multiple Passband FIR Filter for 5G Applications

S N Raju Kalidindi[1,2]*, Sudheer Kumar Terlapu[2] and M Vamshi Krishna[3]

[1]Centurion University of Technology & Management, Odisha 761 211

[2] Shri Vishnu Engieering College for Women, Bhimavaram 534 202

[3] Dhanekula Institute of Engineering & Technology, Vijayawada 521 151

Received 03 August 2021; revised 06 October 2021; accepted 28 October 2021

In television, mobile and digital signal processing applications, efficient multiband filters have great usage. The proposed architecture gives the Reconfigurable Finite Impulse Response (FIR) filter with multiple pass bands. Implementation of architecture utilizes FIR filter with control logic and frequency selection circuit. By adjusting the parameters of the filter, proper bandwidth of the pass band can be achieved and the ripple content in the pass band and stop band can be controlled. The efficient way to adjust the bandwidth is to choose the effective value of the filter length and coefficients. The area efficient multiple passband FIR filter using control logic has been proposed with order (n = 4 and 11). Target device that has been selected for implementation is Field Programmable Gate Array (FPGA) Virtex 4 Device. The Look-Up Tables (LUT) utilization for the implemented architecture with length of filter (n = 11) is observed to be 6%.

Keywords: Block least mean square, Distributed arithmetic, FIR filter, Folded direct form, Unfolded direct form

Introduction

Finite Impulse Response filters are designed to use in many applications like audio and video signal processing, noise reduction and echo cancellation, etc. In 5G mobile communication applications filters with fast data rates, multiple bandwidth selections and area efficient low power consumption implementations play a crucial role. It has been proposed that the reusability and design productivity of the semiconductor technology can be improved with the help of Reconfiguration of FIR filter with parameterized IP (Intellectual Property) cores.[1] The development cost and product cycle times can be reduced with the help of IP cores and SoC design methodology. The functionality of IP cores and SoC designs are customized as per the requirements of the customer or end user. The selection of parallel or serial implementation of the FIR filters will be a downside amidst the area and speed performance of the final design.

Park, Meher presented Distributed Arithmetic (DA) design technique in order to design a FIR filter which uses the look-up tables (LUT) inorder to save up precomputed results and thereby reducing the computational complexity.[2] The DA based FIR filters require LUTs, adders and scalar units to implement the FIR filter. The Authors also presented a multiplier which includes the add and shift functions for calculation of results. This method is best suitable for less number of filter coefficients and less area overhead.

An efficient reconfigurable and fixed FIR filter with Multiple Constant Multiplications (MCM) has been proposed with less number of adders.[3] It reduces horizontal and vertical sub expression removal in a specified number of coefficient applications. The rearranged form of filter design approach is employed in the implementation to provide parallel processing. Therefore, the area efficient architecture with less power consumption has been implemented.

The FIR filter design with block least mean square (BLMS) algorithms for DA method was proposed by Mohanty and Meher.[4] It utilizes LUT technique to compute filter output in order to perform correlation and convolution. The shared LUTs minimize the significant amount of addition operations. But shared LUT architecture couldn't implement the block based structure of the filter. The BLMS algorithm can provide the block based filter's structure. A Time-multiplexed DA based FIR filter design using DRAM technique was presented with reconfigurable pipeline

*Author for Correspondence
E-mail: ksnraju@ieee.org

adder trees and partial product generators.[5] The proposed architecture is an area efficient DA based FIR filter, which saves 80% of the slice logic compared with the conventional DA and decomposed DA methods. Step Limiting Bull-Horrocks Modified (SLBHM) and Step Limiting n-dimensional Reduced Adder Graph (SLRAGn) algorithms have been proposed with the least number of adders steps in order to implement the FIR filters.[6] They contrasted the results with the existing algorithms and shown that the required number of adders steps are less at lower values of the order. The proposed algorithms are not giving optimized results in terms of adder steps as the order of the filter is increasing. But the tradeoff between area and speed is effective even for higher values of the order of the filter. The area efficient and fast multiplier techniques are presented, these multipliers are suited for the filtering operations as they are giving less computation time with less hardware utilization.[7,8]

The FIR filter with LUT cascaded based implementation using weighted-sum (WS) functional decomposition has been proposed.[9] For serial multiplication of coefficients and delayed input sample signals requires more hardware functional blocks and takes more time to compute the result. To avoid the shortcomings of the serial multiplication the authors proposed parallel multiplication with the help of cascaded LUT. The implementation requires less number of LUTs and memory, when compared to the RAM based implementation. The FIR filters with intellectual property cores has been proposed with data path and controller block.[10,11] Where coefficients are supplied from the controller block and data samples are getting from datapath unit. For each selection of bandwidth the user has to give the control information, therefore utilizing these core structures in the real time applications will be difficult. The parameterization of datapath for order 2, 4, 6, and 8 has taken and presented the results with 33% & 48% less power utilization and 10% increase in the area utilization.

An MCM direct form Block FIR filter with lesser power and greater speed utilization implementation was presented.[12] The entire function is divided into several blocks and each block will be executed parallel to increase the speed of the operation. A 5 way 6 tap filter presented by the authors, requires more area due to additional circuits like Serial In Parallel Out (SIPO) and Parallel In Serial Out (PISO) at the input and output of the filter respectively. The block FIR filter is suitable for fixed set coefficients and less number of taps. If the filter order increases, it increases the overhead of additional circuits like SIPO and PISO, which in turn increases the overall area of the filter.

The high speed and low complexity multipliers have been proposed with the canonical signed digit (CSD) coefficients.[13] Pipelined Registers are inserted within multiplier's critical path to reduce the overall delay, thereby increasing the speed of computations. The length of the critical path is estimated to decrease around 33% of the existing critical path without pipeline registers. It indicates the pipelined registers would consume the logic overhead, which increase the utilization of the silicon area during the full custom implementation of the design.

The low complexity reconfigurable digital bandwidth filters are proposed with cascading of subband filters.[14,15] These reconfigurable digital filters can be tunable over audio frequency range. The coefficients are represented by CSD format. Using Farrow structure and CSD coefficients, the single bandwidth filters are used to generate variable bandwidth. The values of the coefficients are slightly adjusted to get the new bandwidth of the filter. The authors proposed a low complexity variable bandwidth filter with multiplier less architecture. The proposed implementation is simulated for digital hearing aid requirements. The reduction of hardware complexity of the FIR filters with the help of CSD representation has been discussed.[16] The authors also mentioned the errors involved due to the non zero CSD coefficients. Due to increase of the non zero CSD coefficients, the value of the error is prone in the stopband extrema. In order to reduce the errors in the stopband the minimum non zero CSD reductions should be 3 non zero digits and it gives an optimal solution.

For semicustom implementation of reconfigurable filters requires the target device. There are several semicustom ICs available with large amount of logic utilization and network interconnect resources, such as FPGAs. The crossbar architectures with 16 bit granularity is suitable for the DSP applications.[17] The central crossbar and multiple crossbar fabrics are used in the crossbar architectures with routing mappings, these fabrics and mappings allow the user to implement the reconfigurable architectures in the semicostum implementation. The internal architecture

of the Virtex FPGA consists of configurable logic blocks (CLBs), input-output blocks (IOBs), clock resources, memory blocks and configurable routing. The bitstream structure of the implementation is organized as arrays or frames. Each frame corresponding to an atomic reconfiguration unit. The frames can be accessed individually for read or write operations, which allows Virtex devices to use for dynamic reconfiguration and partial reconfiguration applications.[18] The Virtex4 devices are also intended for the structures for reconfigurable architectures, therefore Virtex4 device is selected for the implementation.

The FIR filters with IP Cores and reconfigurable FIR (RFIR) filters discussed till now, mainly concentrates on the area reduction using the effective multiplication techniques and increasing the speed of computations using LUTs. These methods are capable of generating the filters, single bandwidth. The reusable IP core FIR filters give the adjustable bandwidths, but require the user to send coefficients every time to change the filter bandwidth through the control unit. The auto adjustment of filter bandwidth with control logic is proposed in the paper.

The Proposed architecture is implemented with Multiply and Accumulate (MAC) based implementation of FIR filter, controller and frequency selection circuit. The folded direct form of FIR filter implementation is used to implement the linear phase FIR filter with less number of multiplications because every linear phase finite impulse response requires coefficients that are symmetrical around the middle coefficient.

Implementation

The Reconfigurable FIR filter with multiple band of frequencies can be implemented with the help of coefficient generation unit. For a specific band of frequency, the selection of coefficients must be done properly. The width of the pass band and transition band depends on the number of coefficients and their precision. The design of a reconfigurable FIR filter with adjustable bandwidths can be done with multiple set of coefficients.

It is proposed that, RFIR filter with multiple pass bands depending upon the frequency of the input signal and the selection of pass band depends on the input signal frequency. In the design of RFIR filter, a frequency selection unit, control unit and FIR filter are used as shown in Fig. 1. The frequency selection unit receives two input signals i.e., clock signal and the reference clock signal. The frequency selection unit generates the decision based on the condition that, whether the input signal frequency is above or below the reference clock signal. The output of the frequency selection unit is FS.

The controller unit is having a finite state machine which will generate the required set of coefficients for the FIR filter. The finite state machine consists of four states, state 0 is the initial state, state 1 is the conditional state, state 2 is the coefficients selection of first bandwidth and state 3 is responsible for coefficients of second bandwidth selection. The controller receives the input signal sample, clock, enable and FS. Based on the decision made by frequency selection unit, the controller generates the coefficients. The FIR filter is implemented with the MAC FIR filter. The amount of multiplication logic can be further reduced with the help of a Distributed Algorithm (DA) technique which uses LUTs to store the partial products of the multiplications. Therefore, the overall computational logic is reduced and results in reduction of power consumption also.

The multiply and accumulate structure is used in the design of FIR filter. The FIR filter output for an N tap filter is given by Eq. 1.

$$y[n] = \sum \square[k]x[n-k] \quad k = 0,1,\dots\dots,N-1 \qquad \dots (1)$$

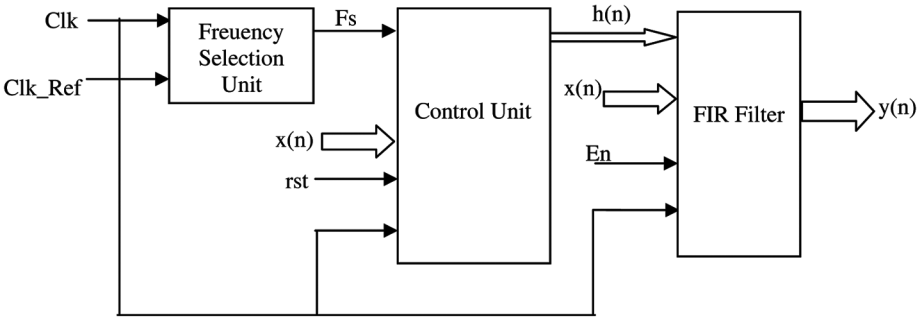

Fig. 1 — Block Diagram of the Proposed RFIR Filter with control logic

It gives the convolution of the latest N samples with the coefficients h(k). The conventional tapped delay realization of the filter is shown in the Fig. 2.[2] The computation of y[n] needs a N-1 number of additions and N number of multiplications and structure implemented is in the form of multiply and accumulate (MAC). But it needs 'N' number of MAC cycles before sampling the following input sample. Therefore, the implementation of parallel MAC structures increases the speed of operation by N times.

The input x(n) is applied to the filter_fir module along with the h(k), enable (en) and clock (clk). The phase detector initially compares the clock signal with the reference clock signal, and depending upon the clock signal's value, it will generate the output signal (FS). The signal FS holds a binary value true or false. Control unit consists of two sets of coefficients and selects only one set at a time based on the value of FS. There is a finite state machine in the control logic with minimum of four states. The initial state of the state machine checks for Start pulse and clear (clr) signals. If both the signals are enabled then it moves to the second state. The second state is a conditional state and selects the next state based on the value of FS. States three and four are having different set of coefficients. The second state selects only one state and therefore one set of coefficients.

Bandwidth selection of the filter relies upon the proper selection of the coefficients of the filter and also on the number of taps. The high precision of the coefficients reduces the ripple content of the filter response. For analyzing the working of the MAC FIR filter of 8 bits with five taps, the tabular form is presented in Table 1. The value of the input sample x(n) at each input clock pulse is shown in the first row of the table and coefficient values are taken from h(0) to h(4) as 1, 2, 2, 2, and 1 respectively. The output samples of the filter y(n) are given in the last column and values in the rectangular box will be available at the filter output. As the number of filter taps is five, output of y(n) is valid after fifth sample i.e., from N(5). The output values can be compared with the waveform presented in the Fig. 3.

The RFIR filter with control logic for area efficient and low power consumption has been presented with 11 taps and 16 bits in Fig. 4. The block diagram of the proposed filter has been implemented and shown in the schematic diagram. The implemented design is contrasted with simulation results of the folded direct form (FDF) and the unfolded direct form (UDF) FIR

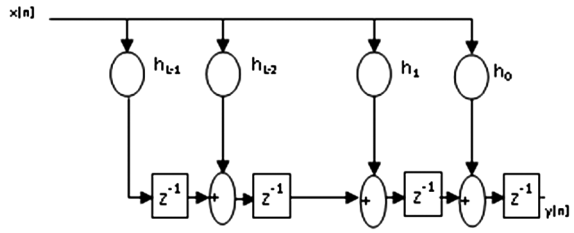

Fig. 2 — MAC based FIR filter

Table 1 — FIR filter response matrix for n=4

x(n)>	2	1	3	4	1	2	3	4	2	0	0	y(n)
n(0)	1											2
n(1)	2	1										5
N(2)	2	2	1									9
N(3)	2	2	2	1								16
N(4)	1	2	2	2	1							19
N(5)		1	2	2	2	1						19
N(6)			1	2	2	2	1					20
N(7)				1	2	2	2	1				20
N(8)					1	2	2	2	1			21
N(9)						1	2	2	2	1		20
N(10)							1	2	2	2	1	15
N(11)								1	2	2	2	8

filter of reusable SoC designs. The two sets of coefficients for the filter were chosen from MATLAB R2021a. The two passbands of the filter are selected as $\omega_{p1} = 0.15\pi$ to $\omega_{p2} = 0.35\pi$ and $\omega_{p3} = 0.55\pi$ to $\omega_{p4} = 0.7\pi$. Filter coefficients for both the pass bands are obtained from MATLAB and they are normalized to the integer values.

The function used to generate the coefficients for the speciefied band width is given in Eq. 2.

$$b = fir1(n, w_n, ftype) \quad \ldots (2)$$

where, n is the order of the filter; w_n is the bandwidth of the filter; *ftype* specifies type of the filter

The default window used in the function for generating the filter coefficients is Hamming window technique. In order to generate the coefficients of the bandpass filter, the value of w_n is given as ω_{p1} and ω_{p2}. Similarly for the next set coefficients the value of w_n is given as ω_{p3} and ω_{p4}. The generated coefficients have the default types as double and float. Therefore, all the coefficients are having the fractional numbers. Hence, it is difficult to use the coefficients directly in the filter design. In order to avoid large fractional numbers, it is advisable to normalize the fractional numbers into integer values. The normalization factor used in the implementation to convert fractional

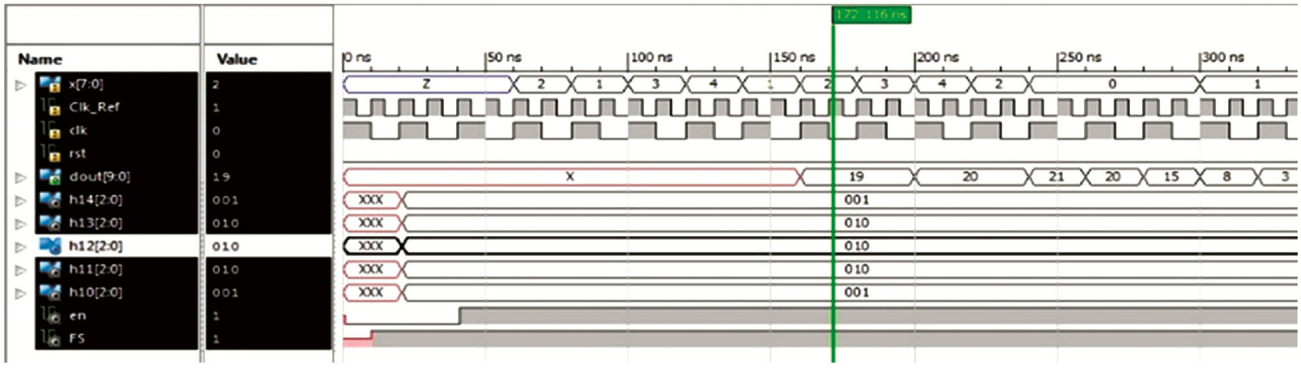

Fig. 3 — Simulation result of RFIR filter

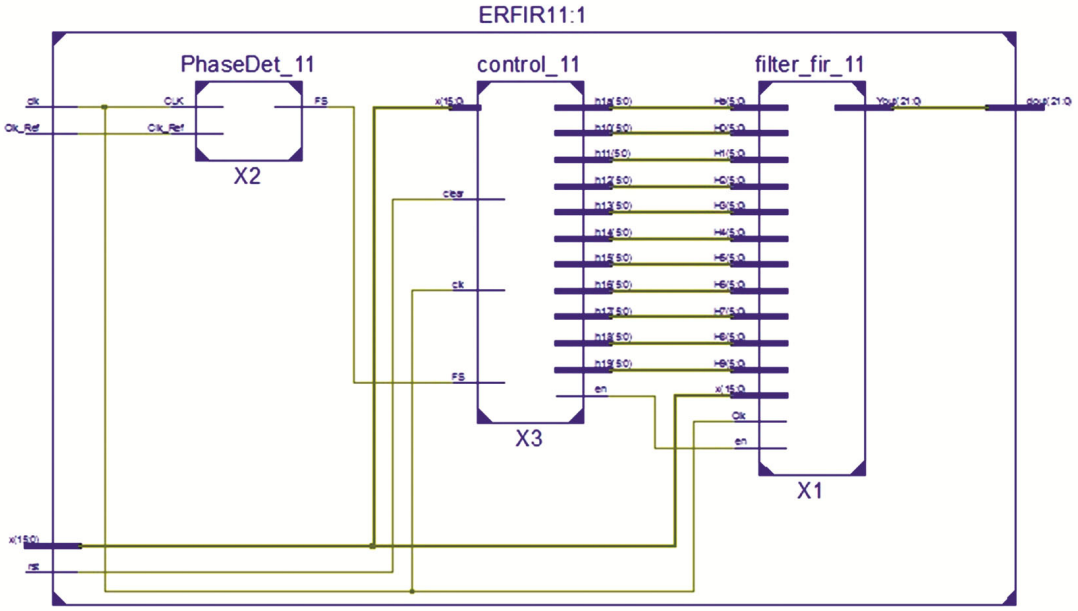

Fig. 4 — Schematic Diagram of RFIR Filter

number into integer numbers is 64 and then rounded to the nearest integer value. The range of coefficients generated after the normalization are between −32 and 31. Therefore the selected numbers of bits for the coefficients are 6 bits. To generate the sharp transition bandwidths and less amount of ripples in the stopband, it requires more number of coefficients and with high precision coefficient values. Hence the coefficient size should also be increased.

The Control unit holds the two sets of coefficients as a look-up table and supplies only one set of coefficients to the FIR filter based on the clock signal frequency. Magnitude and phase responses of the both sets of coefficients are shown in Figs 5 and 6.The x-axis of the magnitude and phase response curves is denoted by normalized frequency in radians/cycles. The normalized frequency is usually denoted for the

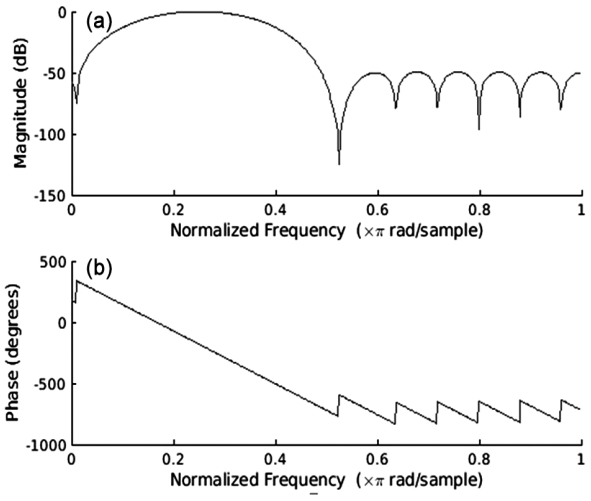

Fig. 5 — magnitude and phase response of ω_{p1} = 0.15π to ω_{p2} =0.35π

representation of digital signals. Multiple pass bands can be achieved with the help of clock gating technique. The Phase detector circuit should give its decision and depending upon the reference signal's value and the input clock signal. The phase detector is used to compare the reference clock signal with the input clock signal and the decision is made based on the values of both signals. For more pass bands, it requires to generate multiple reference signals. The width of phase detector output depends on the number of pass bands required. Control logic will select one set of coefficients stored in the LUTs. For each specific pass band, there will be a set of coefficients. State machine inside the control logic selects any one set of coefficients based on the decision made by phase detector circuit.

Simulation Results

Reconfigurable FIR filter with control logic with 11 taps and 16 bits has been simulated in a Xilinx ISIM simulator with Virtex4 XC4VFX12 device as a target device. The simulation results are compared with UDF and FDF reusable SoC designs.

The coefficients generated using MATLAB R2021a are listed in Table 2. These two sets of coefficients are stored in the control unit and one of the sets will be supplied to the FIR filter automatically based on the value of the clock signal. The schematic diagram generated after synthesizing with RTL schematic is shown in the Fig. 4. The device utilization of the implementation is verified after synthesizing the multiband reconfigurable FIR filter. The number of slices utilized out of the total resources available in the virtex 4 device is only 1%. Similarly the remaining resources utilized like BUFG, DSP48s and external IOBs are observed to be 6%, 34% and 17% of the available resources as shown in Table 3. The devices like one 16 × 7 to 22 MAC unit, nine 16 × 7 multipliers, nine 16 × 7 adders and registers utilized in the implementation are listed in the Table 4. The sizes of the MAC, multipliers and adders will vary based on the bit width of the input signal and coefficients.

The simulation results shown in the Fig. 3 is the multiband reconfigurable FIR filter with 16 bits and 11 taps was implemented with the same architecture of the MAC based filter design. The area utilization comparision is always made based on the number of LUTs occupied in the design from the existing methods. Here it is observed from the comparison

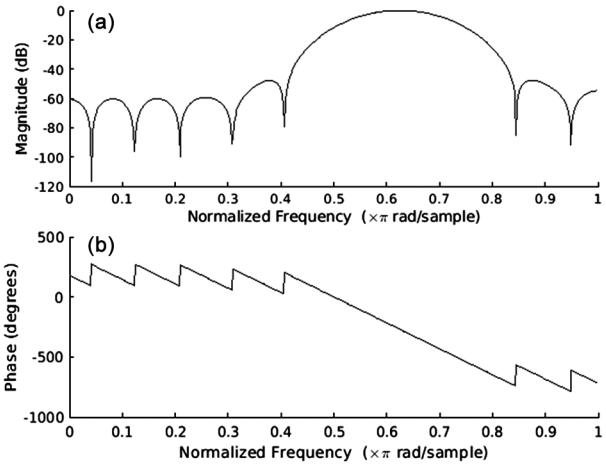

Fig. 6 — magnitude and phase response of ω_{p3} = 0.55π to ω_{p4} =0.70π

Table 2 — Coefficients for selected bandwidths

Coefficient	h0	h1	h2	h3	h4	h5	h6	h7	h8	h9	ha	hb
First Set	0	−2	−6	−5	7	21	21	7	−5	−6	−2	0
Second Set	0	−2	6	3	−18	12	12	−18	3	6	−2	0

Table 3 — Device Utiliation Summary

Name of the Device	Resource utilization out of Available resources	% utilization
Number of BUFGs	2 out of 32	6%
Number of DSP48s	11 out of 32	34%
Number of External IOBs	41 out of 240	17%
Number of Slices	82 out of 5472	1%
Number of SLICEMs	0 out of 2736	0%
Number of LOCed IOBs	0 out of 41	0%

Table 4 — Advanced HDL Macro Statistics

Device	Number of Units for Filter with Order(n) 11
16×7-to-22-bit MAC	1
16×7-bit multiplier	9
16×7-bit registered multiplier	1
16-bit adder	9
Flip-Flops	133
1-bit latch	1
6-bit latch	10

Table 5, that there are a very few number of 4 input LUTs utilized for the proposed implementation. It is observed that the proposed implementation requires 79.8% less number of 4 input LUTs than the existing UDF architecture and 83.3% less number of LUTs than FDF architecture. The UDF and FDF are the methods used in the design of reconfigurable FIR filter design for multiple pass band. The number of

Table 5 — Comparison Table

Parameter	UDF 16bit 11 taps[1]	FDF 16bit 11 taps[1]	Proposed RFIR 16bit 11 taps
Slice Register	405	420	137
Flip Flop	85	132	133
Latches	320	288	4
4 input LUTs	752	913	152

Table 6 — Operating Frequency & Timing Report

Parameter	Filter with order(n) 4	Filter with Order(n) 11
Min. time period	1.353ns	1.353ns
Max. operating frequency	738.962MHz	738.962MHz
Min. I/P arrival time before clock	5.606ns	15.269ns
Max. O/P required time after clock	4.997ns	5.405ns

slice registers for the proposed implementation would utilize around 33% of both the existing architectures. The operating frequency and timing responses are presented in the Table 6.

Conclusions

The implementation of reconfigurable FIR filter using control logic is proposed in the paper, which is suitable for allowing multiple band of frequencies with the help of single FIR filter and multiple set of coefficients. The implemented design requires less amount of silicon space when compared to the existing architectures for multiple passband filters. The automatic selection of coefficients has been used to get multiple band of frequencies. The high data rates, area efficiency and multiple passband filters are the key factors for 5G communications. The area efficient reconfigurable FIR filter has been implemented with MAC based FIR architecture. The simulation result of the proposed architecture shows that, it can save 79.8% to 83.3% of the 4 input LUTs and requires only 33% slice registers compared to the existing UDF and FDF FIR filter architectures. The total power consumed by the device is 177.7 mW and static power consumed is around 166.6 mW. The implemented device can be operated at a maximum frequency of 738 MHz.

The speed of the filters can be further improved by incorporating the LUT based designs to avoid the multiplication operations. For the design of linear multiple passband FIR filters, the coefficients are similar over center coefficient. Therefore to reduce the overall area of the multiplier and to improve the speed of multiplications, folded structures can be used.

References

1 Farooq U, Saleem M & Jamal H, Parameterized FIR filtering IP cores for reusable SoC design, *Third Int Conf on Informat Technol: New Generat (ITNG'06)*, (2006) 554–559.

2 Mirzaei S, Hosangadi A & Kastner R, FPGA implementation of high speed FIR filters using add and shift method, *Int Conf Comput Des*, (2006) 308–313.

3 Trimale M B & Chilveri P G, FIR filter implementation on FPGA using MCM design technique, *Int Conf Circuits, Controls, Commun (CCUBE)*, (2017) 213–217.

4 Mohanty B K & Meher P K, A high-performance energy-efficient architecture for FIR adaptive filter based on new distributed arithmetic formulation of block LMS algorithm, *IEEE Trans Signal Process*, **61(4)** (2013) 921–932.

5 Bhagyalakshmi N, Rekha K R & Nataraj K R, Design and implementation of DA-based reconfigurable FIR digital filter on FPGA, *Int Conf Emerg Res Electronics, Computer Science and Technology (ICERECT)*, (2015) 214–217.

6 Kang H-J & Park I-C, FIR filter synthesis algorithms for minimizing the delay and the number of adders, in *IEEE Trans Circuits Syst II: Analog and Digital Signal Processing*, **48(8)** (2001) 770–777.

7 Naga Sravanthi V & Terlapu S K, Design and Performance analysis of rounding approximate multiplier for signal processing applications, in *Smart Intelligent Computing and Applications. Smart Innovation, Systems and Technologies* edited by S Satapathy, V Bhateja, J Mohanty, S Udgata, **160** (2020) 395–403.

8 Dake L, Jyothi & Terlapu S K, Low complexity digit serial multiplier for finite field using redundant basis, *Indian J Sci Technol*, **9(1)** (2016) 1–5.

9 Sasao T, Iguchi Y & Suzuki T, On LUT cascade realizations of FIR filters, *8th Euromicro Conf Digit Syst Design* (DSD'05), (2005), 467–474.

10 Wang C H, Erdogan A T & Arslan T, High throughput and low power FIR filtering IP cores, *IEEE Int SOC Conf*, (2004) 127–130.

11 Meher P K & Pan Y, MCM-based implementation of block FIR filters for high-speed and low-power applications, *IEEE/IFIP 19th Int Conf VLSI System-on-Chip*, (2011) 118–121.

12 Park J, Jeong W, Mahmoodi-Meimand H, Wang Y, Choo H & Roy K, Computation sharing programmable FIR filter for low-power and high-performance applications, *IEEE J Solid-State Circuits*, **39(2)** (2004) 348–357.

13 Tamada M & Nishihara A, High-speed FIR digital filter with CSD coefficients implemented on FPGA, in Proc the *Asia and South Pacific Design Automat Conf*, (2001) 7–8.

14 Amir A & Bindiya T S & Elias E, Design and implementation of reconfigurable filter bank structure for low complexity hearing aids using 2-level sound wave decomposition, *Biomed Signal Process Control*, **43** (2018) 96–109.

15 Haridas N & Elias E, Design of reconfigurable low-complexity digital hearing aid using Farrow structure based

variable bandwidth filters, *J Appl Res Technol*, **14(2)** (2016) 154–165.

16 Hewlitt R M and Swartzlantler E S, Canonical signed digit representation for FIR digital filters, *IEEE Workshop on SiGNAL PROCESS SYST. SiPS 2000. Design and Implementation (Cat. No.00TH8528)*, (2000)416–426.

17 Hartenstein R, A decade of reconfigurable computing: a visionary retrospective, *Proc Design, Automat Test Europe*, *Conf Exhibit 2001*, (2001) 642–649.

18 Mesquita D, Moraes F, Palma J, Moller L & Calazans N, Remote and partial reconfiguration of FPGAs: tools and trends, *Proc Int Parallel Distribut Process Sympos*, (2003) 8–17.