Indian Journal of Science and Technology, Vol 9(48), DOI: 10.17485/ijst/2016/v9i48/94148, December 2016
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

A Novel Architecture of FBMC Transmitter using Polyphase Filtering and its FPGA Implementation

M. Ramesha[1*] and T. Venkata Ramana[2]

[1]Department of ECE, ACS College of Engineering, Bangalore - 560074, Karnataka, India; rameshmalur037@gmail.com
[2]Department of ECE, GITAM University, Visakhapatnam - 530045, Andhra Pradesh, India; teppala@gmail.com

Abstract

Background/Objectives: In most of the Cognitive radio and frequency spectrum access we are using Filter Bank Multicarrier (FBMC) architecture; the Cognitive radio changes its parameters according to the environmental changes to operating the data in transmitting and receiving. This design makes use of FBMC to process according to parameter changes**. Methods/Statistical Analysis:** Here we design an efficient Architecture for FBMC Transmitter using Polyphase filter technique. The polyphase filter contains low power FIR Filters, Modified Booth multipliers, Carry save adders and it is flexible to fixed-point and floating point arithmetic. We are using Pipelined architecture to design an IFFT Module in the transmitter. **Findings:** The proposed FBMC Transmitter is better in terms of performance, complexity overhead compared to conventional methods. The FBMC Transmitter is designed and implemented on Artix 7 FPGA and compared with different FPGA's and Results are verified on Chip Scope pro tool. **Applications/Improvement:** FBMC as a new concept in cognitive radio applications for dynamic access spectrum management purpose.

Keywords: Cognitive Radio, Chip Scope-Pro Tool, FBMC, IFFT, FPGA, Polyphase Filtering

1. Introduction

Most of the FPGA's and DSP hardware kit is having high clock signal rate in terms of MHz and the ultra scale High end devices work at GHz speed. To reduce the Clock frequency in most of the hardware architecture, we need to go with Multiplexing Techniques and at low clock frequency it sorts out the data processing. Filter Bank is the one of the best ways to split the large amount of incoming data into sub-band carrier signals with low bandwidth and sampling rate. We have seen in most of the research works FBMC dominates over OFDM in all the constraints. To reduce the hardware complexity and high Adjacent Leakage Power Ratio (ACLR) for Television White Space (TVWS) they are using FBMC approach and it is flexible to produce good ACLR[1]. A most application like communications, Digital Radio, and wireless networks uses relaxed synchronization for Future generation. To reduce signaling overhead we need to use non orthogonal multi-carrier waveforms[2,3]. This technique is unconventional to Orthogonal Frequency Division Multiplexing (OFDM). Hence Filter bank Multicarrier (FBMC) is a better approach and it is non orthogonal multicarrier in nature.

For 5-G communications, hardware prototyping of FBMC/OQAM Baseband system is designed in paper[4]. From algorithm specification to on-board verification and demonstration is done and during implementation, software and hardware co-simulation step used. The system description of OFDM and OQAM/FBMC are analyzed in detail. However the Hardware complexity is equivalent to OFDM System. In many advanced communication systems, the best medium access technique is Power Line Communications (PLC) and also it is one of the multi carrier approaches. For PLC in FBMC system implementation, they are using three different architectures which are applied for poly-phase filtering. By using different Filter forms, they analyzed quantization errors for both FBMC Transmitter and receiver[5]. The parallel

*Author for correspondence

algorithms and FBMC Systems are designed based on Normalized prototype filter. The coefficients are pre-computed and stored in memory. The transmitter module here uses a parallel architecture which includes Fast Fourier Transformation (FFT). This achieves better performance over OFDM. Unified structure and parallel algorithm are better than the conventional serial algorithms[6,7].

The overview of FBMC techniques for both Transmitter and receiver using polyphase filtering and the FFT algorithm implementation for WIMAX communication systems is described[8,9]. The pipelined FFT Architecture is reconfigurable and it supports upto 1024 –point and which is used in OFDM systems[10]. Reconfigurable OQAM based filter bank multicarrier modulation system is implemented[11]. Implementation allows reconfiguring the number of subcarrier from 64 to 1024 and dynamic allocation of subcarriers between primary and secondary users. The results and analysis shows that the filter bank multicarrier system is better than conventional OFDM system for free spectrum sharing application.

This paper is organized as follows: The proposed FBMC Transmitter is explained in detail with IFFT Module and Polyphase filtering in Section 2, Multiplication module in Section 3, the simulation results and design summary and comparative analysis are done in Section 4. Section 5 gives overall conclusion of the FBMC Transmitter.

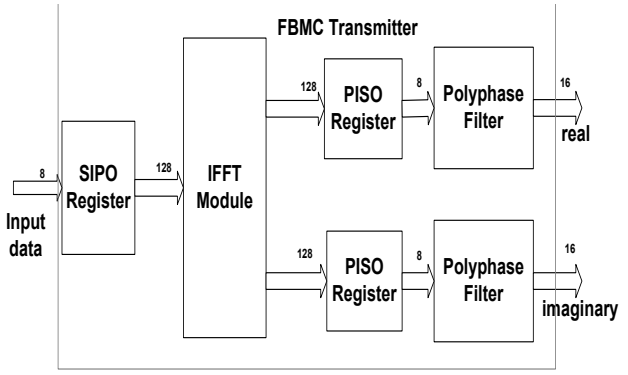

Figure 1. Hardware architecture of FBMC transmitter.

2. Proposed FBMC Transmitter

The proposed FBMC transmitter is as shown in Figure1. The input data which is serially fed to Serial in Parallel Out (SIPO) register. The parallel output data is 128 –bit which is given to IFFT Module. Once IFFT operation is done real and imaginary output data is fed to two PISO register. This gives continuous 8-bit and it is given to polyphase filter. Once filtering is done, we get the 16-bit real and imaginary transmitted output data.

2.1 IFFT Module

Internal Architecture of IFFT- Module as shown in Figure 2. The IFFT module receives the 128-bit data from SIPO register and divided to 8- data flip-flop modules. Each D-FF contains 16bits data and gives the same data as an output after one clock interval of time. The flip-flops data is fed to Processing Unit. The Processing Unit contains 8 different Butterfly Units. Each Butterfly unit receives 16-bit data from flip-flops. The basic Butterfly unit is as shown in Figure 3. The two inputs are a and b, 'W' indicating twiddle factor produces two outputs c, d. By using these 8-units of butterfly structure we build whole IFFT architectures[12]. Each butterfly unit requires 2-adders and 1-complex multiplier unit. The Architecture is in Decimal In Time (DIT) radix -2 structures. In this design input to butterfly is 16-bits and 8 Butterfly units are used. In general For N-stage, N-adders and N/2 complex multipliers are required.

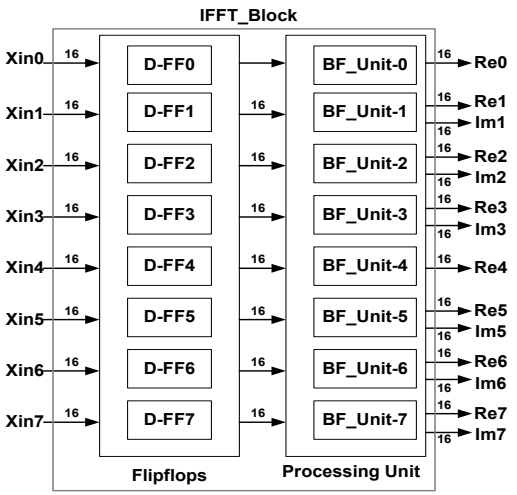

Figure 2. Internal architecture of IFFT-module.

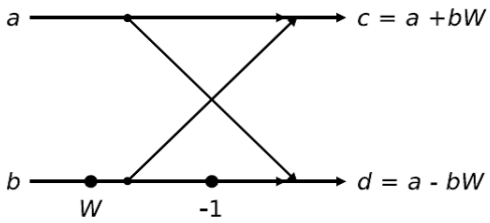

Figure 3. Radix-2 FFT butterfly structure.

2.2 Polyphase Filtering

In order to Implement Polyphase filter it is important to decide filter structure, data representation and Selection whether data it is fixed-point or floating point arithmetic[13]. To implement the design, a basic direct form filter structure is adopted, where this structure is divided into three modules, such as FIR_First, FIR_Gen, FIR_Last modules is as shown in Figure 4. The Polyphase Filter is implemented for 8-tap filter. The N(8) tap filter consist of N+1 (9) coefficients and N+1(9) multipliers. The elements in the FIR_First are delay unit (register), multiplier which is kept as a constant. In the FIR_Gen that is repeated for N-tap filter and for the last module only multiplier and adder are taken into account.

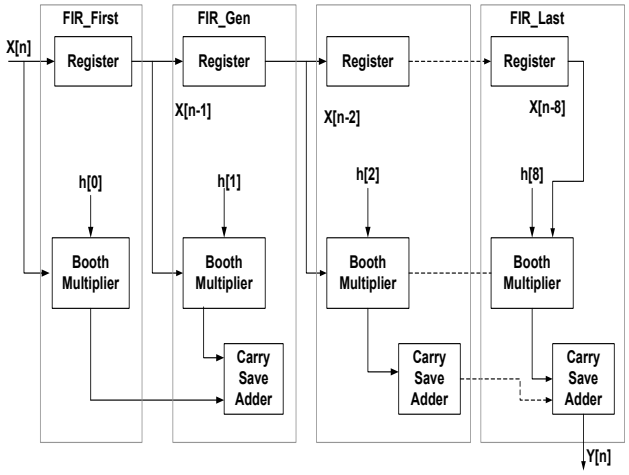

Figure 4. Polyphase filtering using direct form of FIR filter with three division modules.

3. Multiplication Module

Multiplier plays a very important role in high performance systems such as FIR Filters, Transformations and DSP. The system performance is determined by performance of the multipliers because this element slows the operation of the system. Moreover, it is area consuming and power consuming module. Hence improving the speed and area are the major design issue. However area and speed are generally conflicting constraints, so that improving the speed results in increase in area. The key objective is reduction in power with less area utilization and delay overhead. We can use a new algorithm by which it is possible to achieve both power reduction and area/delay reduction. To achieve smaller area utilization, faster operation and power optimization, a Booth algorithm is used practically. Booth multiplication is one of the methods which is allowing for the faster multiplication and data will be multiplied by recoding the number.

Due to these following merits, the Booth multiplier offers two advantages:

- Only about half of the products are required during the computation.
- Delay on the critical path is lesser than the other conventional methods.

The Booth multiplier multiplications depend on the radix, which is power of 2.As radix value increases the computation for the partial products reduces. In this paper we discuss a higher representation radix which leads to fewer digits, which means fewer partial products. Thus it reduces the number of cycles.

3.1 Radix-2Multiplier

This is one of the procedures that allows for smaller, faster multiplication circuits and data will be multiplied by the number. It allows only half the partial products to be needed during computation. But it has two main drawbacks.

- Firstly as number of add/subtraction operation becomes variable which is difficult for parallel multiplier.
- Secondly when there are isolated 1's, it is inefficient to compute.

3.2 Radix-4Multiplier (Modified Booth Multiplier)

The Radix-4 multiplier is known as modified Booth multiplier is shown in Figure 5 and this algorithm reduces the no of cycles. The reduction in the number of cycles, besides the use of recoding and carry save adders, it leads to substantial improvements in speed over basic multipliers[14]. The Booth multiplier multiplication consists of three steps:

- Booth encoder
- Generation of partial product
- Carry save adder

3.2.1 Booth Encoder

Multiplication of two's complement numbers are more complex because performing the straight forward

unsigned multiplication of the two's complement illustration of the input's does not give the accurate results. It increase the time required to perform the multiplication. To quickly convert the two's complement numbers into a format that is easily multiplied a technique is adopted which is called Booth encoding.

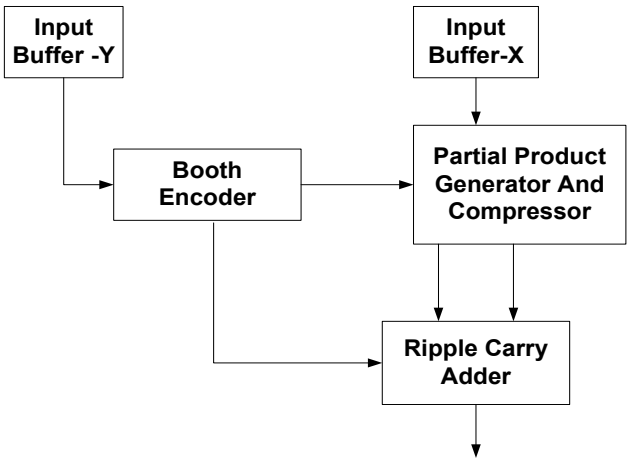

Figure 5. Block diagram of modified booth multiplier.

Booth Encoding conversion Table is shown Table-1.

Table 1. Booth encoding conversion

A	B	Y
0	0	0
0	1	1
1	0	-1
1	1	0

Apply encoding technique to the multiplier bits (By using Table 1) before the bits are used for getting partial products.

3.2.2 Partial Product

The initial step in binary multiplier is Partial products generation. In conventional multiplier the partial product consist of series of logic AND gates as shown below in Figure 6**.**

If the multiplier bit is '0', then the partial product will also zero, and if multiplier bit is '1' then the multiplicand is copied as it is. Each one of the partial product will get shifted one unit to the left. In the signed multiplication onwards, each partial product row is shifted one unit to left. Proposed Booth encoding technique is used for the reduction of the number of partial product. Here the two bits are compared and recoded. The Radix-2 has the disadvantage, but it can be eliminated by examining three bits at a time. This method has advantage as it reduces the partial product to n/2.The truth table for Modified Booth multiplier is shown in Table 2.

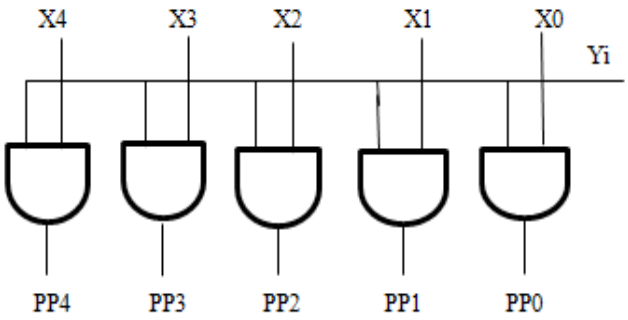

Figure 6. Partial product using and gates.

3.2.3 Carry Save Adder

One of the key methods to enhance the speed in digital circuits is the ability to add numbers using minimal carry propagation Carry Save Adders (CSA) are one of the most broadly used technique for fast arithmetic. CSA used to lessen the number of addition cycles as well as to make each cycle faster. It is not necessary to propagate the carries during each addition. Instead of that the carries generated during each addition may be saved as parallel carries added with the next operand during next addition. These adders deletes or postpones any carry after summation of two bits and substitute this carry in the next bit sum column by column basis, which saves the time.

Table 2. Modified booth multiplier truth table

Xi + 2	Xi + 1	Xi	Partial Product
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	2
1	0	0	-2
1	0	1	-1
1	1	0	-1
1	1	1	0

4. Results and Analysis

The proposed FBMC Transmitter is designed and demonstrated on FPGA Artix 7 (XC7A100T-3CSG324) and simulated using Model sim 6.5. The Simulation result of FBMC-Transmitter is shown in Figure 7. The input data value is 78 in Hexa, after 5 clock cycles will get the transmitted outputs of FBMC. Check the timing in Figure 7.

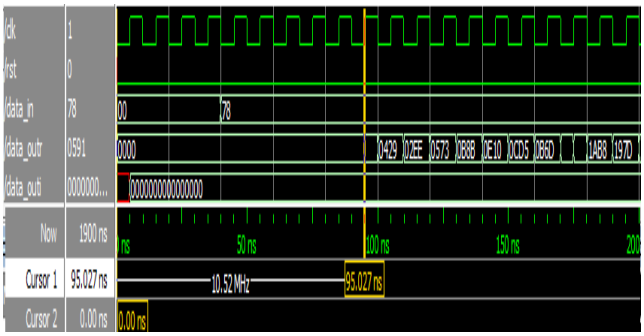

Figure 7. Simulation results of FBMC- transmitter with timings.

The hardware result of FBMC- Transmitter is shown in Figure 8 using debugging tool –chip scope pro. The input data is 0111_1000 is applied through Triggerport 1. Getting continuous data and at the end 2DIE is last output data both in simulation and on hardware tool.

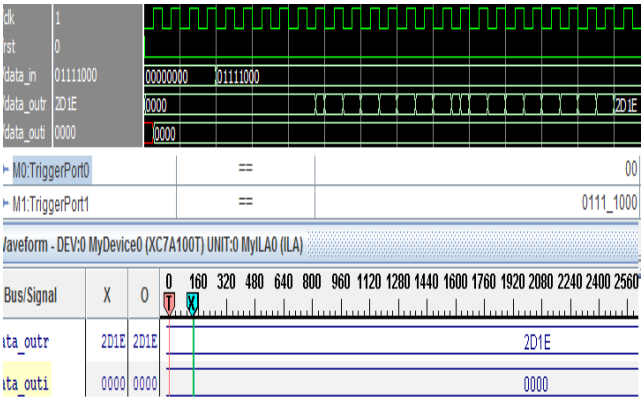

Figure 8. Hardware verified results of FBMC-transmitter using Chip scope-pro tool.

The Table 3 shows the design utilization of different FPGA's and DSP slices.

The Figure 9 shows the Simulation results of IFFT Module with timings. In Figure 9 inputs are 16-bit each. The values of inx0…inx7 are 10,20,30,40,50,60,70 and 80. We will get outputs. The 0th and 3rd outputs are having only real values and no imaginary values. For Verification purpose, Matlab tool results of IFFT Module are also shown in Figure 10.

Table 3. The Proposed FBMC-Transmitter Design summary using

Different FPGA's–Spartan-3, Viterx-5 and Artix-7

Logic Utilization	XC3S400-5 PQ208	5VLX110T-3 FF1136	7A100T-3 CSG324
Number of Slice Registers	3185	4696	440
Number of Slice LUTs	440	440	5007
Number of fully used LUT-FF pairs	5595	178	179
Number of bonded IOBs	42	42	42
Number of MULT18X18s	8	NA	NA
Number of BUFG/ BUFGCTRLs	1	1	1
Number of BUFG/ BUFGCTRLs	NA	8	8

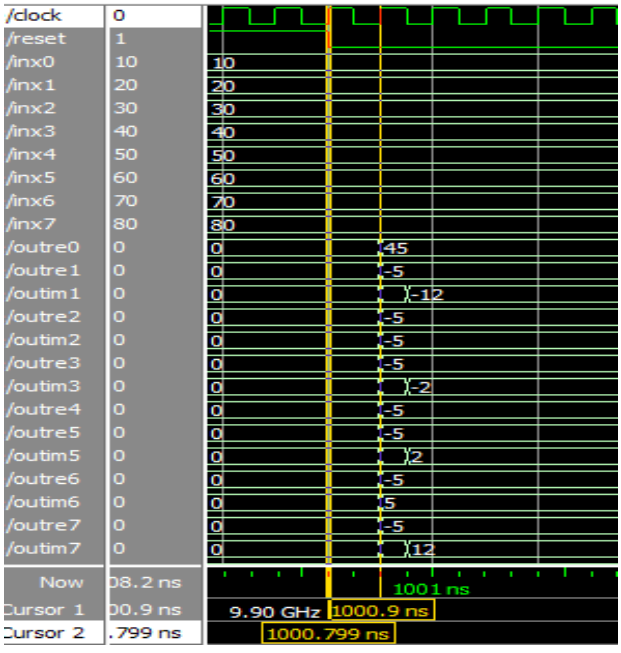

Figure 9. Simulation results of IFFT Module with timings.

The Simulation result of Polyphase filtering with timings is as shown in Figure 11. The input is 8-bit data 4E in hex value, output are generated after 1 and half cycles.

```
>>  x =[10 20 30 40 50 60 70 80]

x =

    10    20    30    40    50    60    70    80

>> y =ifft(x)

y =

  Columns 1 through 5

  45.0000           -5.0000 -12.0711i  -5.0000 - 5.0000i  -5.0000 - 2.0711i  -5.0000

  Columns 6 through 8

  -5.0000 + 2.0711i  -5.0000 + 5.0000i  -5.0000 +12.0711i
```

Figure 10. Matlab tool results of IFFT module for verification purpose.

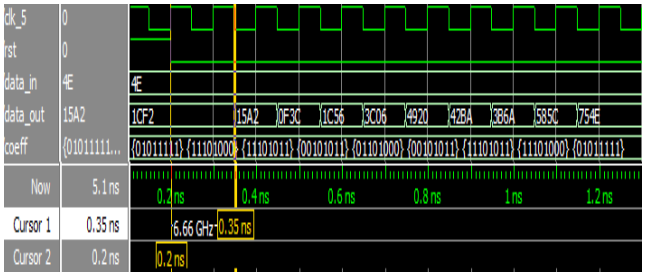

Figure 11. Simulation results of polyphase filtering with timings.

The Table 4 gives comparison results of proposed FBMC Transmitter design and previous design[1] .For comparison they are using KINTEX-7 series FPGA. The complexity overhead also calculated and listed under Table 4.

Table 4. Comparison table of proposed design and previous design[1]

Proposed Design			
TX-Modules	**Slice registers**	**LUT's**	**DSP48E1**
IFFT Module	2864	2731	8
Poly phase filtering	80	1108	0
FBMC Transmitter	4957	3840	8
Previous Design[1]			
IFFT Module	5131	3805	13
Poly phase filtering	577	677	16
FBMC Transmitter	10981	7665	29
Complexity Overhead	2%	2%	65%

5. Conclusion

The Proposed FBMC Transmitter is designed and implemented on Artix7 FPGA. The pipelined Radix 2 IFFT module initiates to speed up the whole transmitter process. The polyphase filtering uses 8-Tap FIR filters including Modified Booth multipliers and Carry save adders. By using booth multiplier, we are reducing the half of the computation time in polyphase filtering. The comparison results prove that our IFFT Module, Polyphase filter and Overall Transmitter Modules gives better area results in terms of slices (2%), LUT's (2%) and DSP slices (65%). The hardware complexity is reduced and a performance is improved in terms of area.

6. References

1. Berg V, Dore JB, Noguet D. A flexible radio transceiver for TVWS based on FBMC. Microprocessors and Microsystems Journal. 2014 May; 38(2014):743–53.
2. Berg V, Dore JB, Noguet D. A multiuser FBMC receiver implementation for asynchronous frequency division multiple access. 17th Euromicro Conference on Digital System Design; 2014 Aug. p. 16–21.
3. Kang AS, Vig R. Performance analysis of filter bank multicarrier cognitive radio for physical layer under binary symmetric radio fading channel. International Journal of Computer Applications. 2014 May; 93(6):0975–8887.
4. Hardware prototyping of FBMC/OQAM base band for 5G mobile communication. 2016. Available from: https://perso.telecom bretagne.eu/amerbaghdadi/publications/index.php
5. Poudereux P, Mateos R, Hernandez A, Cruz-Roldan F. Study of suitable filter architectures for FBMC techniques applied to PLC communications. IEEE 20th Conference on Emerging Technologies and Factory Automation (ETFA); 2015. p. 1-6.
6. Zeng Y, Liang YC, Chia MW, Peh ECY. Unified structure and parallel algorithms for FBMC transmitter and receiver. IEEE 24th Annual International Symposium on Personal, Indoor and Mobile Radio Communications; 2013. p. 922–6.
7. Sanal R, Schariah C, Shibu RM. End to end latency analysis of FBMC using FPGA. IJERT. 2014 Aug; 3(8):1-6.
8. Ramesha M, Raman TV. Analysis of polyphase filtering and FFT in FBMC Transceiver. IJECT. 2014; 3(2):40.
9. Harikrishna K, Rao TR. FPGA based FFT algorithm implementation in WiMAX communications system wireless communication. 2nd International Conference Vehicular Technology, Information Theory and Aerospace and

Electronics Systems Technology (Wireless VITAE); 2011 Feb. p. 1-6.
10. Jhingan A, Kansal L. Performance assessment of OFDM utilizing FFT/DWT over Rician channel affected by CFO. Indian Journal of Science and Technology. 2016 Mar; 9(12):1-7.
11. Vijayakumar P, Malarvizhi S. Reconfigurable filter bank multicarrier modulation for cognitive radio spectrum sharing - A SDR implementation. Indian Journal of Science and Technology. 2016 Feb; 9(1):1-6.
12. Sarada T, Vigneswaran V. Low power 64 point FFT processor. Indian Journal of Science and Technology. 2016 Jan; 9(4):1-6.
13. Varga L, Kollar Z. Low complexity FBMC transceiver for FPGA implementation Radio elektronik. 23rd International Conference; Pardubice. 2013. p. 219–23.
14. Babulu K, Parasuram G. FPGA realization of radix-4 booth multiplication algorithm for high speed arithmetic logics. International Journal of Computer Science and Information Technologies. 2011; 2(5):2102-7.