iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 3 **Issue:** VI **Month of publication:** June 2015

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

FPGA Implementation of Low Power FIR Filter using Modified Booth Algorithm

Devika.S[1], S.Lokesh[2], H. Chandrasekhar[3]
Department of Electronics and communication Engineering, Vemu Institute of Technology (Approved by AICTE, New Delhi & Affiliated to JNTU, Anantapur),P.KOTHAKOTA(P),Chittoor Dist-517112,A.P.

***Abstract*— In the implementation of many digital signal processing, image processing and multimedia algorithms multipliers and adders are widely used. In this paper we can see overview of fir filter by using modified booth multiplier. The speed of the booth multiplier is increased by reducing the delay. This paper presents radix2 and radix4 modified booth multipliers along with sixteen bit full adder which is used to implement FIR filter. So by analyzing the working of booth multiplier helps to frame a better system with less power consumption and lesser area. The code is written in VHDL and the tool used is Xilinx ISE9.1i. Fir filter is implemented on Spartan-3 FPGA with the device package XC3S400.**
Keywords*— Radix2 multiplier,Radix4 multiplier,speed ,power consumption ,Booth algorithm

I. INTRODUCTION

The FIR filter takes in input samples, processes them, and outputs the samples. A large percentage of filters implemented in the digital domain are Finite Impulse Response (FIR) filters. A filter is a sequence h(n) that operates on an input sequence x(n) to generate output sequence y(n). Since FIR filter has no feedback in its structure, it is always a stable filter and consumes less power. Multipliers are the key components in implementing FIR Filter. The early computer systems had what are known as multiply and Accumulate units to perform multiplication between two binary unsigned numbers. The multiply and Accumulate unit was the simplest implementation of a multiplier [1]. The very advanced efficient multiplier used in order to implement low power FIR filter is modified booth multiplier. This multiplier uses two methods that is radix2 and radix4. Radix4 method is referred as modified booth multiplier. FIR filters is simple to design. They are guaranteed to be bounded input-bounded output (BIBO) stable. It can be guaranteed to have linear phase. This is a desirable property for many applications such as music and video processing.FIR filters also have a low sensitivity to filter coefficient quantization errors. This is an important property to have when implementing a filter on a DSP processor or on an integrated circuit.

II. MULTIPLICATION ALGORITHM

The multiplication algorithm for an N bit multiplicand by N bit multiplier is shown below:

Y= Yn-1 Yn-2 .....Y2 Y1 Y0 Multiplicand
X= Xn-1 Xn-2 ..... X2 X1 X0 Multiplier

Multiplication Algorithm in steps

1) If the LSB of Multiplier is '1', then add the multiplicand into an accumulator.
2) Shift the multiplier one bit to the right and multiplicand one bit to the left.
3) Stop when all bits of the multiplier are zero.

From above it is clear that the multiplication has been changed to addition of numbers. If the Partial Products are added serially then a serial adder is used with least hardware. It ispossible to add all the partial products with one combinational circuit using a parallel multiplier. The multiplier used in order to boost up the speed is Booth multiplier. The tool used to simulate and synthesis is Xilinx ISE9.1i using ISE Simulator.

III.BOOTH ALGORITHM

Benefit of this algorithm is that it deals with positive and negative numbers in same way [2]. To know booth algorithm we need to know the concept of 2's complement.2's complement represents the number in negative form

A. Conditions for general 2's complement

1) Any number starting with zero is positive number in 2's complement.
2) Any number starting with one is negative number in 2's complement.

Examples: +5 = 0101(it is positive)

-5= 101 (it is negative)
+13 = 01101
-9= 01001(+9) =10111

If the number is negative, write the positive form of that number and then take the 2's complement. In Booth algorithm number is represented in 2's complement always**.**

B. Radix 2 method

This method tells that it always takes 2 bits as shown in Table I.

TABLE I.

y_i	y_{i-1}	Partial product
0	0	0
0	1	1
1	0	-1
1	1	0

Radix 2 algorithm as follows:

1.If multiplier is Zero write all 0's and extend the sign bit till 2n positions.

2. If multiplier is one write the multiplicand as it is and extend the sign bit till 2n positions.

3. If multiplier has -1, take the 2's complement of multiplicand and extend the sign bit till 2n positions.

C. Radix 4 method

This method tells that it always takes 3 bits as shown in Table II. The problem with radix 2 method is that it is not suitable for synchronous designs[3]. Hence we go for grouping 3 bits and It is referred as Modified Booth Algorithm. In Radix 4 method multiplier bits are grouped together and for each group of three bits partial product is generated. The overlapping is necessary so as to know what was happened in the last block as the MSB of the block act as sign bit.

TABLE II

Multiplier bits			Operation On Multiplicand
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

D. Adders

In order to implement FIR Filter sixteen bit full adder is used. A full adder is a logical circuit that performs an addition operation on three binary digits. The full adder produces a sum and carries value, which are both binary digits. It can be combined with other full adders or work on its own.

Volume 3 Issue VI, June 2015
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

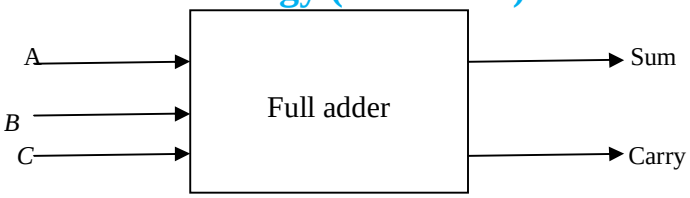


Fig1.Block diagram of Full Adder

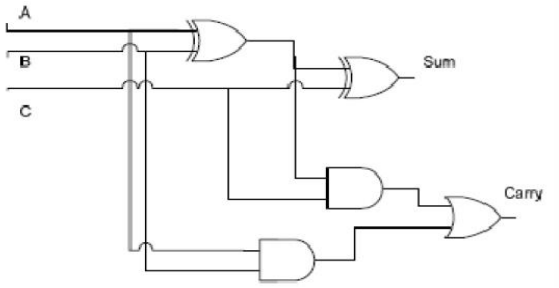


Fig 1. Gate level representation of a FULL ADDER.

Sum=A XOR B XOR C
Carry=(A OR B) AND (B OR C) AND (C OR A)

D. Finite Impulse Response (FIR) Filter

FIR filters have impulse responses of finite lengths. In FIR filters the present output depends only on the past and present values of the input sequence but not on the previous output sequences. Thus they are non recursive hence they are inherently stable.FIR filters possess linear phase response [4]. Hence they are very much applicable for the applications requiring linear phase response.

The difference equation of an FIR filter is represented as

$$y(n)= x(n) * h(n)$$

x(n) is the input sample and h(n) is the filter co-efficient. The output of FIR filter is the convolution of x(n) and h(n). FIR filters possess linear phase response. Hence they are very much applicable for the applications requiring linear phase response [5] .

IV. RESULTS

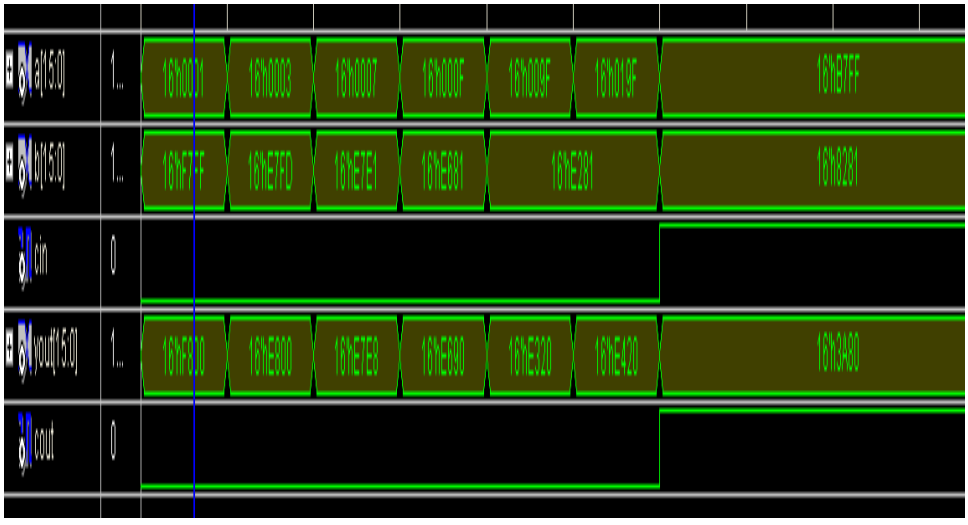


Fig 2: Simulation of 16-bit carry adder

Volume 3 Issue VI, June 2015
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

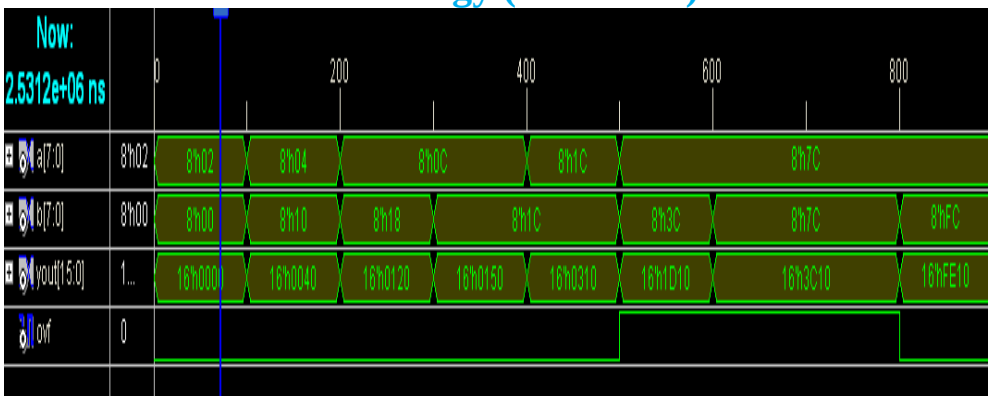


Fig 3: Simulation of Radix-4 Multiplier

Max Delay	Netname
4.020	b_3_IBUF
3.257	b_4_IBUF
3.078	b_0_IBUF
2.827	b_2_IBUF
2.820	a_4_IBUF
2.771	b_1_IBUF
2.762	b_5_IBUF
2.714	a_6_IBUF
2.651	a_2_IBUF
2.550	a_1_IBUF
2.531	b_6_IBUF
2.476	a_3_IBUF
2.445	a_5_IBUF
2.443	yout_14_OBUF
2.426	a_0_IBUF
2.414	a_7_IBUF
2.401	yout_10_OBUF
2.207	yout_12_OBUF
2.120	b_7_IBUF
2.073	u3/N11

Fig4: Delay report of Radix-4 Multiplier

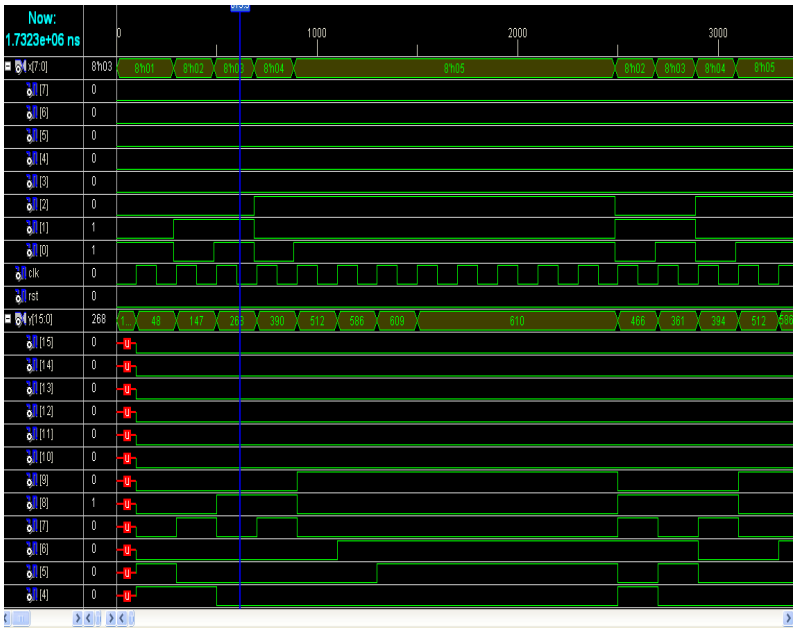


Fig5: Simulation of FIR Filter

Volume 3 Issue VI, June 2015
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Ambient Temperature (°C)	25
Junction Temperature (°C)	26.85
Case Temperature (°C)	26.51
Part Type	Commercial
Airflow (LFM)	0
Package	tq144
Total Power (mW)	56.01

Fig 6: Total estimated power of FIR Filter

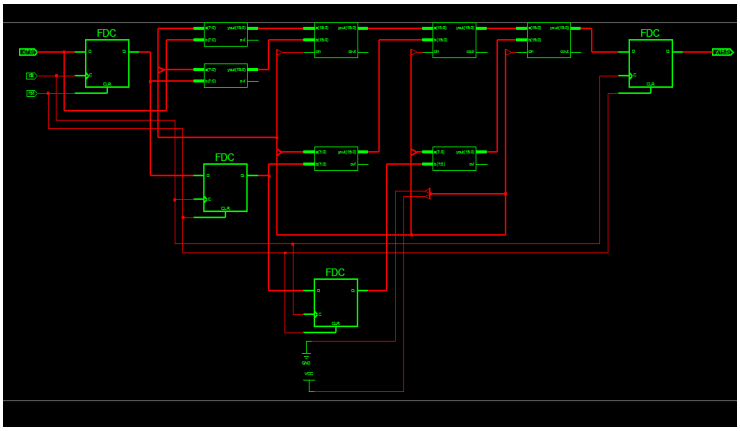


Fig 7: RTL Schematic of FIR Filter

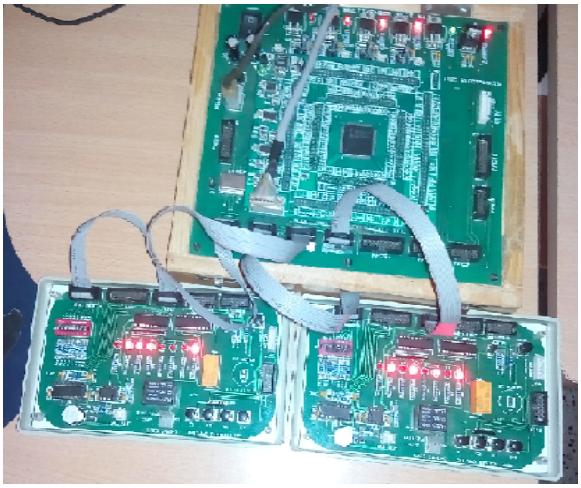


Fig.8: Fir filter Hardware implementation on Spartan-3 on FPGA

V. CONCLUSION

The modified booth multiplier is successfully implemented in tap delay Fir Filter. The code is written in VHDL and successfully simulated and synthesized using Xilinx ISE9.1 using Spartan-3 FPGA. It is shown that modified booth multiplier is fast in computation as a result speed increases. Hence when computations steps reduce FIR filter consumes less area and power. Hardware implementation of FIR Filter is done on Spartan-3 FPGA and the device used is XC3S400 with package TQ144. Hence we can conclude that radix 4 modified booth algorithm works the best.

VI. ACKNOWLEDGMENT

I sincerely thank my Project Coordinator Mr. H. CHANDRASEKHAR, M.Tech, and Associate Professor for his guidance, valuable suggestions and support in the completion of the work. I express my sincere gratitude to my Guide MR.S.LOKESH, M.Tech, Assistant Professor, and Department of ECE for his supervision and timely guidance with valuable pieces of advice during the course of my work. I extend my thanks to HOD Mr. L. RAMA MURTHY, for his support for completion of this work.

REFERENCES

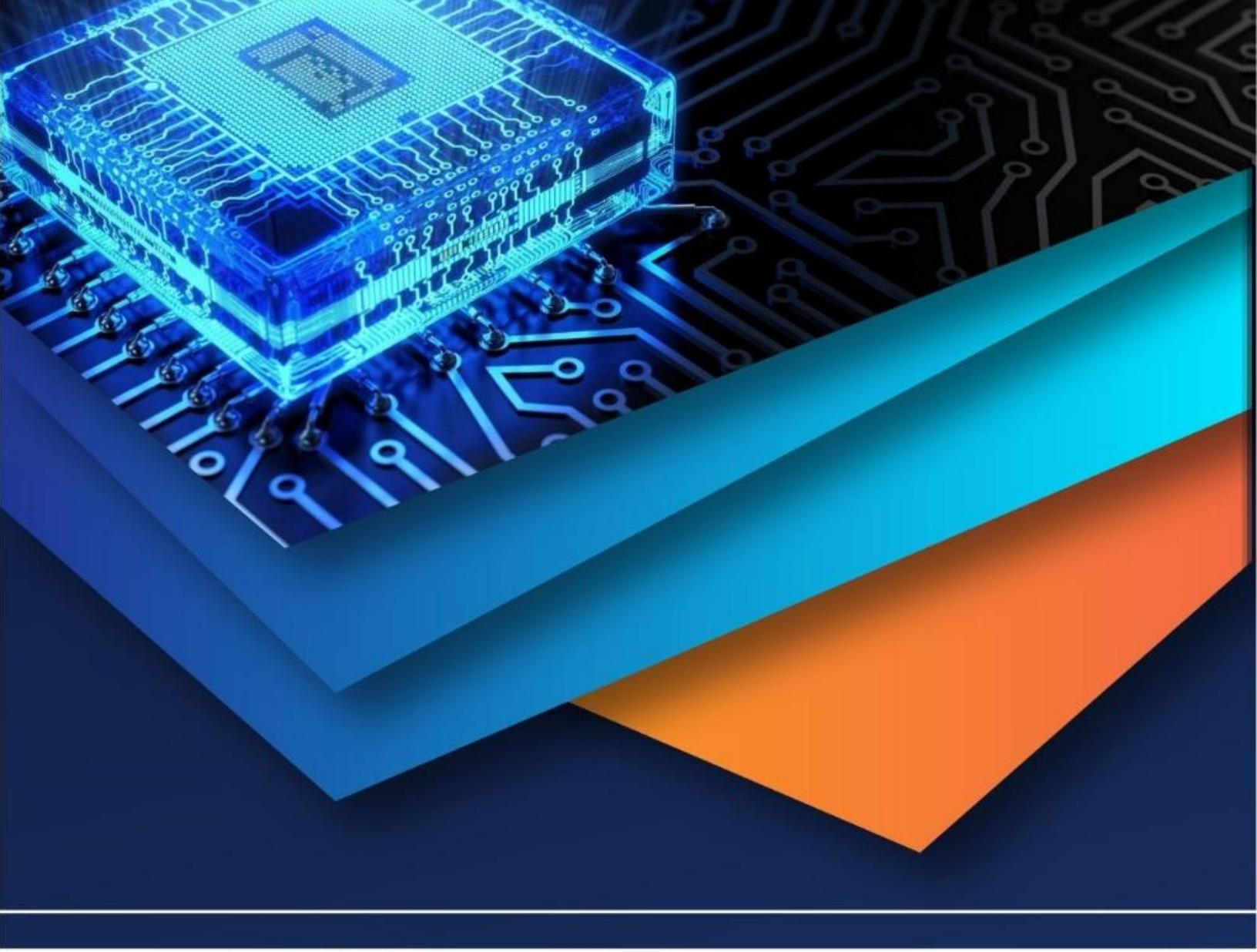
[1] T. J. Todman, G. A. Constantinides, S. J. E. Wilton, O. Mencer, W. Luk, and P. Y. K. Cheung, "Reconfigurable computing: architectures and design methods," IEE Proc. Comput. Digit. Tech., vol. 152, no. 2, pp. 193-207, Mar. 2005. International Journal of Engineering and Applied Sciences 5:2 2009 127.

[2] P.R.Aparna, and N. Thomas, "Design and Implementation of a High Performance Multiplier using HDL", International Conference on Computing, Communication an Appilcation, pp. 1-5, 2012.

[3] HeshamA. Altwaijri,"Area and performance Optimized CMOS Multipliers", PhD Thesis, Stanford University, 1997.

[4] Avatar Singh, S.Srinivasan " Digital Signal Processing"

[5] E. C. Ifeachor, and B. W. Jervis, "Finite impulse response (FIR) filter design" in Digital Signal Processing: A Practical Approach, South Asia: Prentice hall, 2002, pp. 342-440.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY