ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

Implementation of 32 bit Floating Point Multiplier and Adder for FFT Processor Using VHDL

Roshan Pahune[1], Dr. Anagha Rathkanthiwar [2]

PG Student [VLSI], Dept. of Electronics Engg , PCE, Nagpur, India[1]

Associate Professor, Dept. of Elect & Tele. Engg, PCE, Nagpur, India [2]

ABSTRACT: The design of 32 bit Single Precision Floating Point Multiplier and Adder is simulated and presented in this paper.. The floating point number can support wide range of values. It is represented using three fields: sign, exponent and mantissa. In this paper floating point addition, and multiplication algorithms for IEEE-754 (single precision) is presented. The IEEE-754 converter is used to convert decimal floating point number into Binary floating point format and it is also used to verify the results. For performance measurement of this design, various parameters like area, delay and memory requirement are obtained and are presented in this paper.

KEYWORDS: Floating Point Number, IEEE-754 standards, Single Precision, Floating Point Multiplier , Floating Point Adder ,VHDL.

I. INTRODUCTION

This paper describes a methodology to design floating point multiplier and adder using VHDL. This Design follows 32 bit single precision IEEE 754 standards. Floating point number can be represented using three fields the sign, exponent and mantissa. The memory requirement and power consumption is more for floating point algorithms.

A.Floating Point Number Representation

The IEEE-754 Single precision and Double precision format is used to represents the floating point numbers. The floating point numbers can support the wide range of values. The IEEE-754 floating point forrmat have three basic field: sign, exponent, and mantissa

a) Sign
The sign bit is one bit field. If this bit is 1 then it denotes positive number and if it is 0 then it denotes a negative number.

b) Exponent
The exponent is 8 bit field for IEEE-754 single precision format and 11 bit for IEEE double precision format. For single precision format the bias value of exponent is 127 and double precision format the bias value of exponent is 1023.

c) Mantissa
The mantissa is 23 bit field for IEEE-754 single precision and 52 bit for IEEE double precision format. Mantissa is also known as significant.

Table 1. IEEE-754 Single Precision Format

1	8	23
S	E	M

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

Table 2. IEEE-754 Double Precision Format

1	11	52
S	E	M

B.Floating Point Multiplication Algorithm

The Single precision Floating Multiplier is 32 Bit. To multiply the two floating point number we use Booth algorithm and following steps is done:

1. Obtain the Sign : S1 XOR S2.
2. Add the exponents and subtract the bias value : (E1 + E2 - Bias).
3. Place the decimal point in the Mantissa Result.
4. Normalize the result ; obtain the 1 at the MSB of the results Mantissa.
5. Check Overflow or Underflow ?

C.Floating Point Addition Algorithm

The Single precision Floating Point Adder is 32 Bit . To add the two floating point number we use Ripple Carry Adder and following steps is done:

1. Compare the exponents of the two numbers and shift the smaller number to the right until its exponents matches the larger exponents.
2. Add the Mantissa of two numbers.
3. Normalize the result ; obtain the 1 at the MSB of the results Mantissa.
4. Check Overflow or Underflow ?

II .RELATED WORK

"Design andImplementation of IEEE-754 Decimal Floating Point Multiplier, Adder and Subtractor". This paper describe the synthesis and simulation of decimal floating point multiplier , adder and subtraction for 64 bit (Double Precision format. In this paper decimal floating point multiplier is compared with booth algorithm. The design is synthesize using Xilinx ISE software and simulated using ModelSim. This paper indicate that the delay requirement in decimal floating point multiplier isless as compare to booth multiplier. [1]

"Synthesis of Double Precision Floating Point Multiplier Using VHDL". This paper explain the design and simulation of 64 bit double precision floating point multiplier using VHDL tools. The pipelining technique is used for synthesis of the double precision floating point multiplier. In this paper pipelined double precision floating point multiplier is compared with decimal floating point multiplier which indicate that the speed of pipelined floating point multiplier is more as compared to decimal floating point multiplier. The double precision floating point multipliers was synthesize using Xilinx ISE 13.1 tools and simulate in ISE simulator. [2]

"Implementation of 32 Bit Binary Floating Point Adder Using IEEE 754 Single Precision Format". This paper shows the design and simulation of the 32 bit single precision floating point multiplier using VHDL. In this paper pipelined architecture is used to increase the speed and performance of the adder. In this paper IEEE -754 single precision format is used. The floating point adder is synthesize using Xilinx ISE software and simulated in ISE simulator. [3]

"Design of 32 Bit Floating Point Addition and Subtraction Units Based On IEEE 754 Standard". In this paper 32 bit floating point adder and subtractor is designed using Verilog code. The single precision floating point adder and subtractor is synthesize forCyclone III Family and simulated in Quartus II Simulator. [4]

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

III. RESULT AND DISCUSSION

D. Multiplication of Floating Point Number

Input :
The input is taken in the form of Decimal Floating Point Number.
A and B , where A = - 18.5 and B = 9.5
Using IEEE-754 converter the Decimal Floating Point value is converted into Binary IEEE format.
The IEEE Binary Representation of - 18.5 and 9.5 is
-18.5 = 11000001100101000000000000000000
9.5 = 01000001000110000000000000000000
Output : A* B = -18.5 * 9.5 = - 175.75
The IEEE Binary Representation of this output value is
-175.75 = 11000011001011111100000000000000
The output value is verify using IEEE-754 converter.

E.Addition of Floating Point Number

Input :
The input is taken in the form of Decimal Floating Point Number.
A and B , where A = - 18.5 and B = 9.5
Using IEEE-754 converter the Decimal Floating Point value is converted into Binary IEEE format.
The IEEE Binary Representation of - 18.5 and 9.5 is
-18.5 = 11000001100101000000000000000000
9.5 = 01000001000110000000000000000000
Output : A + B = - 18.5 + 9.5 = - 9
The IEEE Binary Representation of this output value is
- 9 = 11000001000100000000000000000000
The output value is verify using IEEE-754 converter.

F. RTL View of 32 bit Floating Point Multiplier and Adder

The RTL view of Floating Point Multiplier and Adder consists of two 32 bit inputs and one 32 bit output.

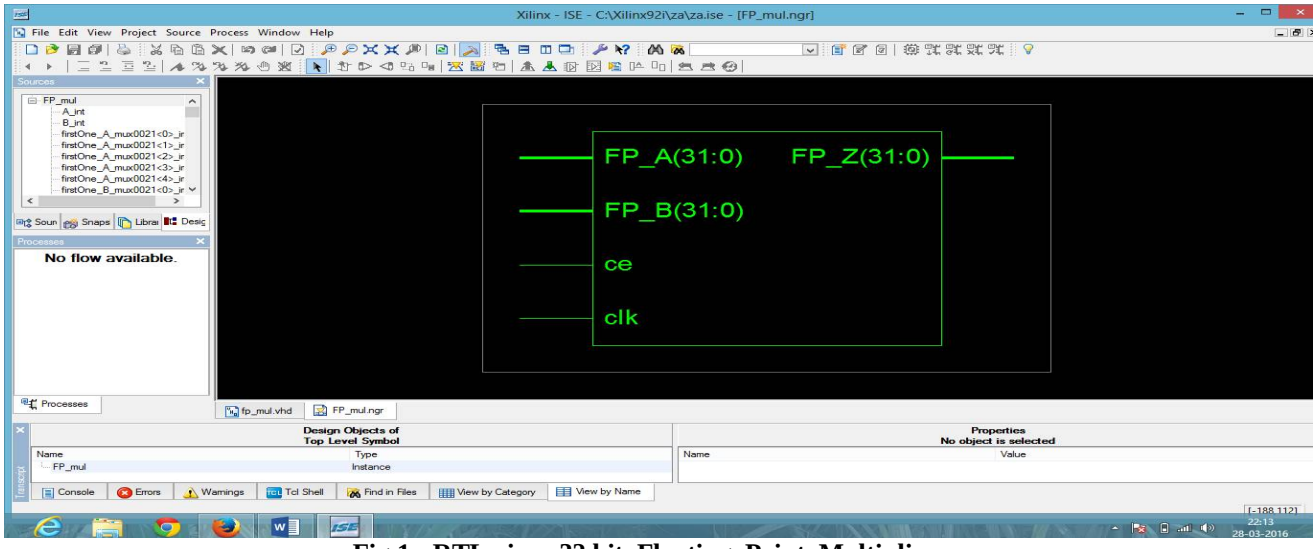

Fig 1: RTL view 32 bit Floating Point Multiplier

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

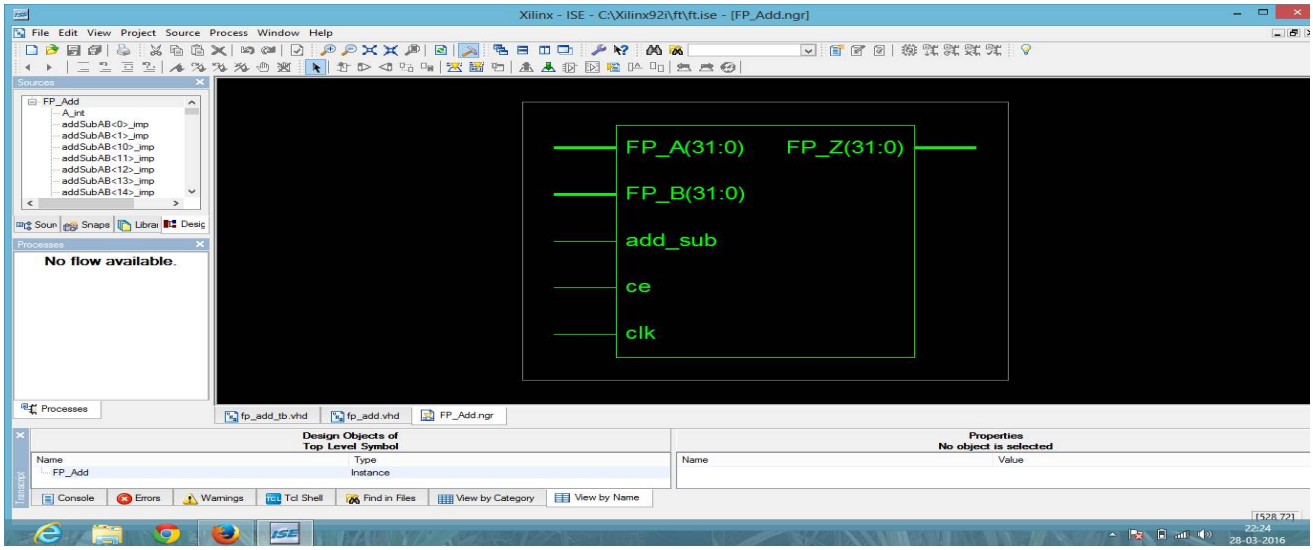

Fig 2: RTL view 32 bit Floating Point Adder

G.Simulation Result of 32 Bit Floating Point Multiplier and Adder

The output of floating point Multiplier and Adder in the form of IEEE Binary Floating Point Number. The Simulation result is verify using IEEE- 754 converter.

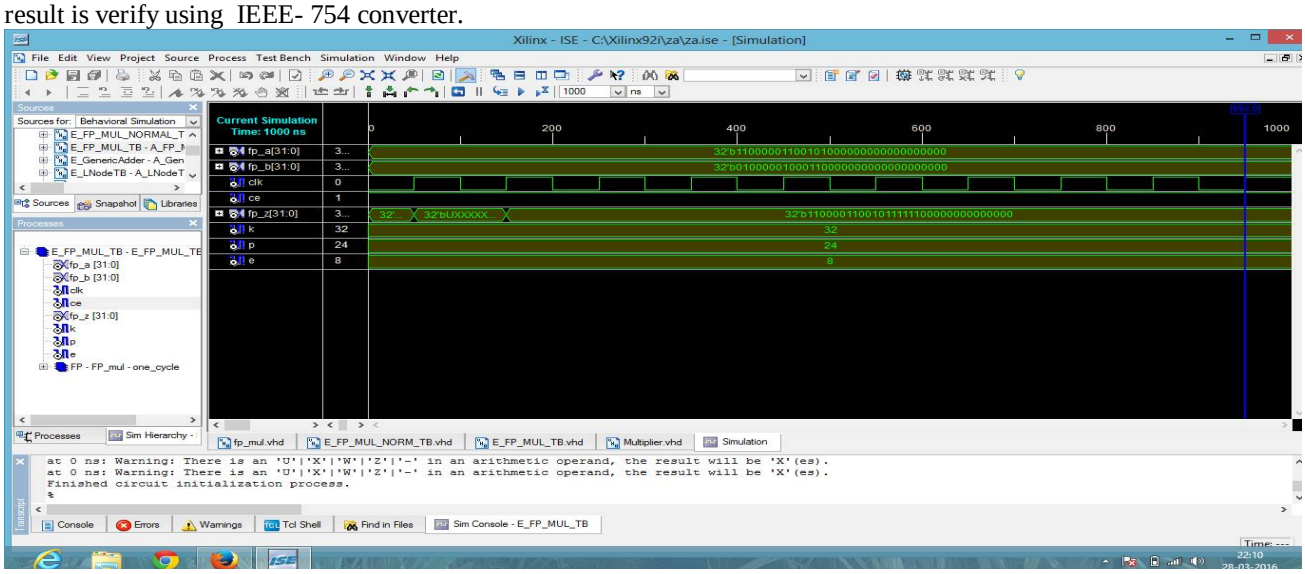

Fig 3: Simulation Result of 32 Bit Floating Point Multiplier

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

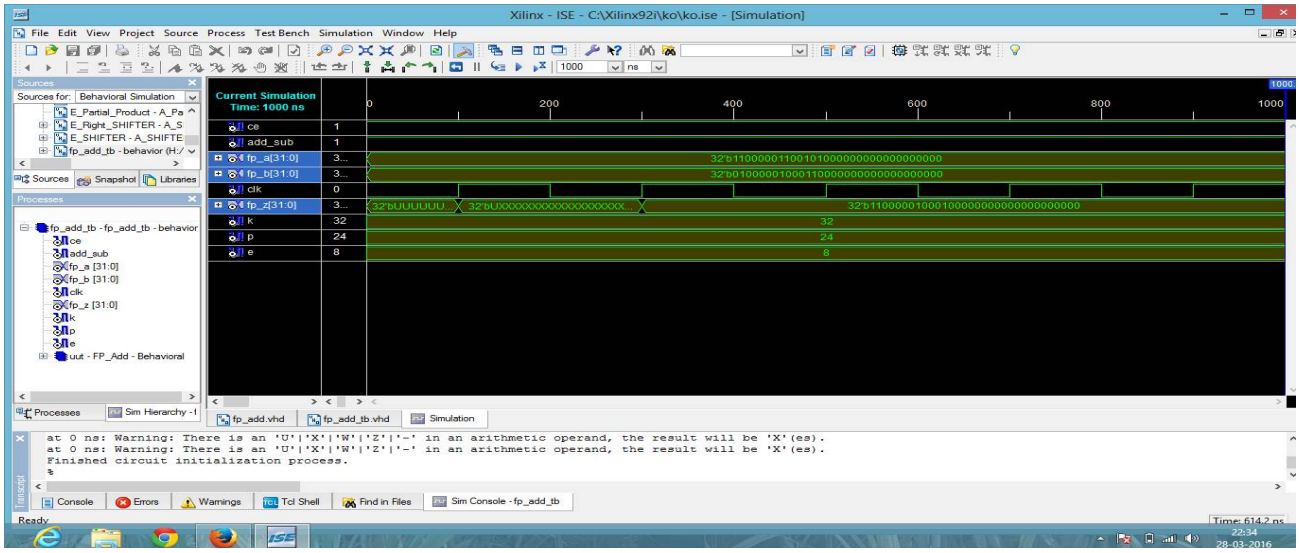

Fig 4: Simulation Result of 32 Bit Floating Point Adder

H. Device Utilization Summary of 32 Bit Floating Point Multiplier and Adder

The 32 bit Floating Point Multiplier and Adder is synthesized & Simulated using Xilinx ISE 9.2i tools. The Floating Point Multiplier achieved the operating frequency of 34.045 MHz with minimum period of 29.373 ns . As well as the Floating Point Adder achieved the operating frequency of 20.845 MHz with minimum period of 47.974 ns.

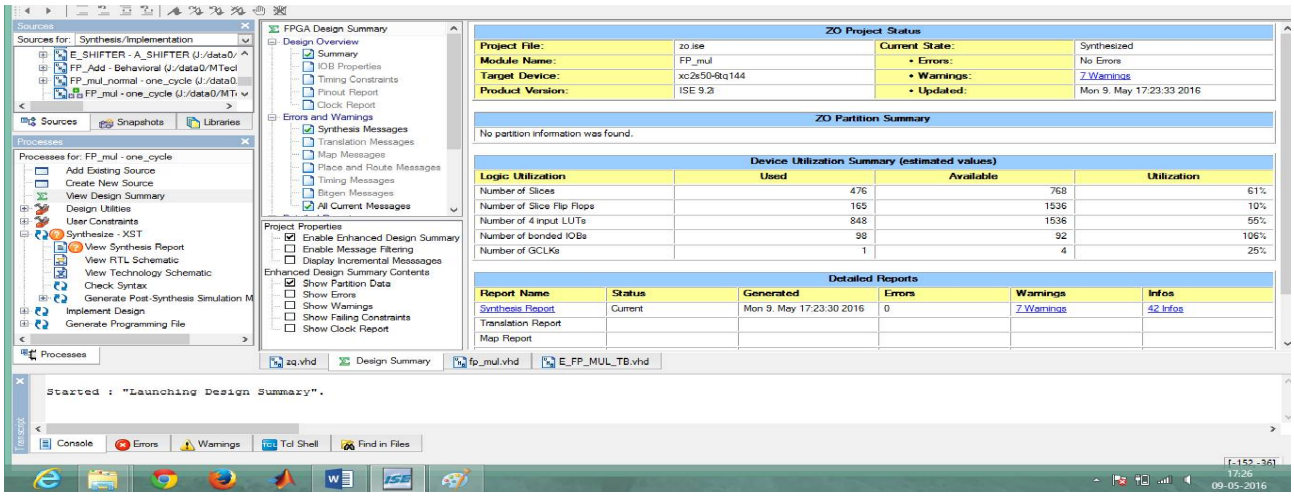

Fig 5: Design Summary of 32 bit Floating Point Multiplier

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

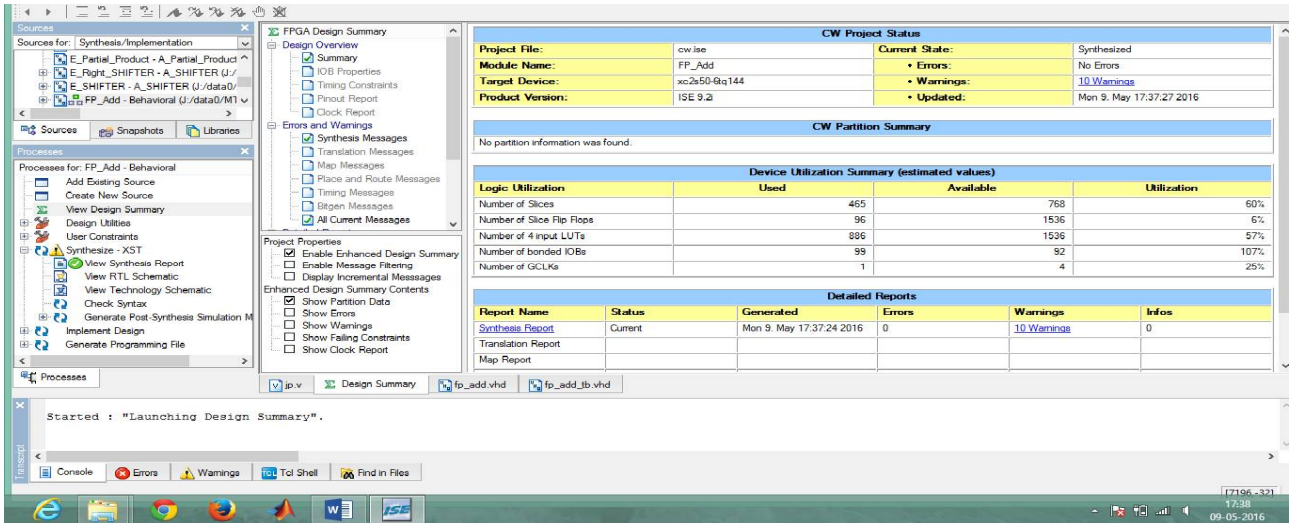

Fig 6: Design Summary of 32 bit Floating Point Adder

IV. CONCLUSION

In this paper, The 32 bit Single precision Floating Point Multiplier and Adder is synthesized and simulated using VHDL tools. The VHDL code has been successfully synthesized and simulated using Xilinx ISE 9.2i tools. The output values of 32 bit Single Precision Floating Point Multiplier and Adder is verify using IEEE-754 converter. The Floating point Multiplier achieved the maximum frequency of 34.045 MHz with delay of 29.373 ns and area of 476 slices. The Total memory requirement for Multiplier is 174628 kilobytes. As well as Floating point Adder achieved the maximum frequency of 20.845 MHz with delay of 47.974 ns and area of 465 slices. The Total memory requirement for Adder is 188456 kilobytes.

V. FUTURE SCOPE

The floating point support the wide range of values. The floating point Multiplier for double precision (64 bit) and floating point Adder for double precision (64 bit) can also be designed and simulated. Also the design can be implemented using suitable hardware platform . In this paper, we have presented design & simulation of 32 bit Single Precision Floating Point Multiplier and Adder.

REFERENCES

[1] S. Murali, B. Srinivas "Design and Implementation of IEEE-754 Decimal Floating Point Adder,, Subtractor and Multiplier, "International Journal of Engineering and Advanced Technology (IJEAT), ISSN: 2249 – 8958, Volume-4 , Issue 1 , October 2014.
[2]Sukhvir Kaur, Parminder Singh Jassal, " Synthesis of Double Precision Floating Point Multiplier Using VHDL," Journal of Research in Electrical and Electronics Engineering" (ISTP-JREEE). ISSN: 2321-2667, Volume 3, Issue 2, March 2014.
[3] RupaliDhobale, SoniChaturvedi " Implementation of 32 Bit Binary Floating Point Adder Using IEEE 754 Single Precision Format," IOSR Journal of VLSI and Signal Processing (IOSR-JVSP) ,e-ISSN: 2319 – 4200, P-ISSN No. : 2319 – 4197.Volume 5, Issue 1, Ver. PP 50-53 ,I (Jan - Feb. 2015).
[4] Ajay Rathor, LalitBandil "Design Of 32 Bit Floating Point Addition And Subtraction Units Based On IEEE 754 Standard," International Journal of Engineering Research & Technology (IJERT,)ISSN: 2278-0181 ,Vol. 2 Issue 6, June 2013.
[5] AddankiPurna Ramesh, Rajesh Pattimi "High Speed Double Precision Floating Point Multiplier,"International Journal of Advanced Research in Computer and Communication Engineering ISSN (Print) : 2319-5940 ISSN (Online) : 2278-1021.Vol. 1, Issue 9, November 2012.
[6]Michael Nachtigal, NagarjanRanganathan,"Design of single precision floating point multiplier based on operand decomposition", 10th IEEE International Conference on nanotechnology joint symposium with Nano Korea, August 2010.
[7] Mohamed Al-Ashrafy, Ashraf Salem and WagdyAnis, "An efficient implementation of Floating Point Multiplier,978-1-4577-0069-9, IEEE-2011.
[8] Liang-Kai Wang and Michael J. Schulte "Decimal Floating-Point Adder and Multifunction Unit with Injection-BasedRounding"
[9] Sheetal A. Jain "Low-Power Single-Precision IEEE Floating-PointUnit".
[10] M.Jenath, V Nagrajan, "FPGA implementation on Reversible floating Point Multiplier, "IJSCE, March 2012.
[11] ParthSarthyMohanty ,"Design and implementation of low power fast multipliers", thesis report NI Rourkela, 2009.
[12] J.Bhaskar A VHDL PRIMER,Third Edition.