5th International Conference on Science, Technology and Management
India International Centre, New Delhi (ICSTM-16)
30th July 2016, www.conferenceworld.in ISBN: 978-93-86171-00-9

IMPLEMENTATION OF HIGH SPEED DOUBLE PRECISION FLOATING POINT UNIT ON FPGA USING VHDL

Monika Maan[1], Abhay Bindal[2]

[1,2] *ECE Department, Maharishi Markandeshwar University, Mullana , Ambala, Haryana (India)*

ABSTRACT

Floating point arithmetic unit is an important and integral part of signal and image processing applications. Many researchers have proposed many new techniques in recent years and compared their merits and demerits with the existing approaches. For floating point operations, two types of precision units i.e. single and double are defined in the IEEE-754 Standard. In this paper the basic arithmetic operations are performed. by using double precision floating point unit In the proposed technique, parallel architecture is introduced along with the high speed adder, which is shared among other operations and can perform operations independently as a separate unit. To improve the area efficiency of the unit, carry select adder is designed with the novel resource sharing technique which allows performing the operations with the minimum usage of the resources while computing the carry and sum for '0' and '1'. The design is implemented using the Xilinx Virtex-7 FPGA and the results show the 25% improvement in the speed of the designed circuit.

Keywords: Carry Select Adder, Floating Point Unit, FPGA, IEEE-754 Standard, Reversible Logic Gate.

I. INTRODUCTION

An Arithmetic circuit has multiple applications in digital coprocessors ,microprocessors and in application specific circuits because they performs the digital arithmetic operations In modern era as the size of chip is decreasing and density increasing power and speed are the most significant parameters for any VLSI design. IEEE -754 Floating-Point Standard through which the floating point operations carried out efficiently with modest storage requirements is the most popular code introduced in 1990, for representing the floating point numbers. [1].

1.1 The IEEE-754 Floating-Point Standard

There are many important operations in IEEE-754 standard which must be performed in order to get the accurate results [1]. In this section, an overall introduction to the floating-point standard is presented: 1) Floating-point number system, 2) Rounding modes, 3) Exceptions.

1.2 Floating-Point Number System

The floating-point number consists of three parts: 1) Sign, 2) Exponent, and 3) Significand.

Sign Bit : Floating point numbers are presented according to the sign magnitude representation which means a positive number is denoted by "0" a negative number is denoted by "1". By changing the value of this bit, change the sign of the number[2].

Exponent Field : Positive and Negative both the numbers are represented by the exponent field. For doing this, a bias is added to the actual exponent in order to get a stored exponent. The exponent field is 8 bits wide according to the 32-bit floating point unit and is of 11 bit wide according to the 64-bit floating point unit. This field contain an exponential with a base of 2 for binary and 10 for decimal. Since it is most commonly used format, only the binary format is covered in this dissertation.

The Significand Bit : The significand, is also called as mantissa, represent the precision bits of the number. It contained the fraction bits and an implicit leading bit. The floating point numbers are stored in normalized form in order to improve the quality of the number. The significand needs to be normalized in order to attain the form of 1.xxx, so that MSB is always "1"[3].

Precision Units	Sign	Exponent	Significand	Bias
Single Precision	**1[31]**	**8[30-23]**	**23[22-00]**	**127**
Double Precision	**1[63]**	**11[62-52]**	**52[51-00]**	**1023**

Fig. 1.1: IEEE-754 Layout for Single and Double Precision Floating Point Values.

In IEEE-754 floating point standard, the single precision format consists of 1 sign bit, 8 exponent bits and 23 significand bits. The double precision format consists of 64 bits which is divided into 1 sign bit which can be 0 or 1 as described above, 11 exponent bits and 52 significand bits.

1.3 Exceptions

In floating point operations there are four possible types of exceptions. The Overflow exception is betided in the precision format of the destination whenever the result cannot be shown as definite number. Underflow exception is occurred when the result is too small to be calculated correctly. Division by zero exception arises, when zero divides a finite nonzero number. The Invalid operation exception take place when for the operation to be performed, the given inputs are not suitable [6].To overcome from these exceptions double precision floating point unit is used.

1.4 Floating Point Unit

The operations on floating point numbers are carried out by the floating point unit, which is a part of the computer system. most of the general purpose processors such as digital coprocessors, microprocessors and application specific processors use floating point unit which is described in IEEE-754 standard. The IEEE-754 is the standard which specifies the floating point arithmetic. the floating point arithmetic has an advantage over

the fixed -point arithmetic as it cover constant relative precision over a wide dynamic range. Due to its ease of use the floating point unit is preferred by the programmers for non integer computations.

The major operations of floating point unit are Addition, Subtraction, Multiplication and Division.

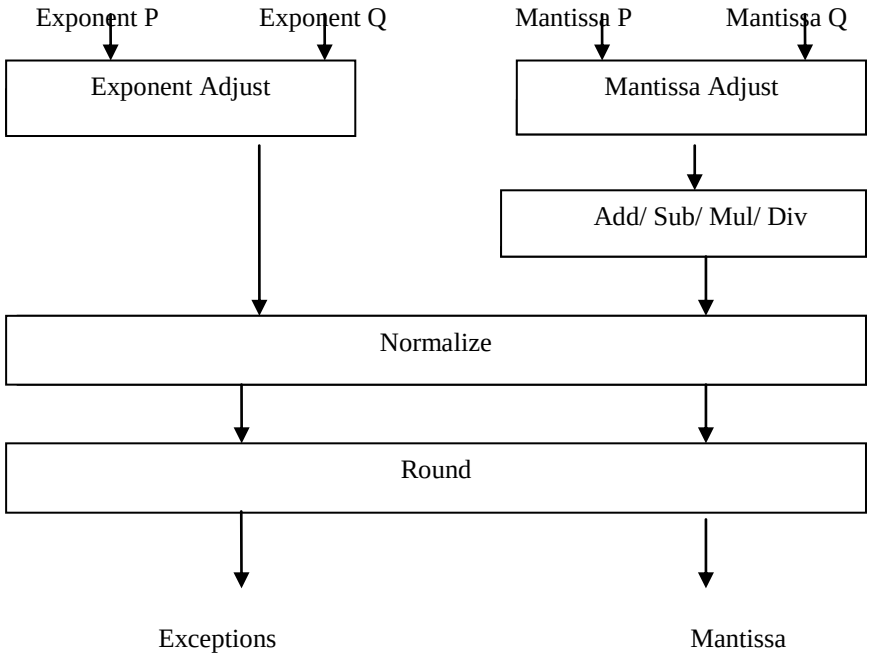

Fig. 1.2: Conceptual Overview of Floating Point Unit.

II. LITERATURE STUDY

This standard "IEEE Standard for Floating-Point Arithmetic, ANSI/IEEE Standard 754-2008, New York:" IEEE, Inc. [1], 2008 describes interchange and arithmetic methods and formats for binary and decimal floating-point arithmetic in computer programming environments. This standard specifies exception conditions and their default handling. An implementation of a floating-point system conforming to this standard may be recognized entirely in software, entirely in hardware, or in any combination of software and hardware.

Kahan et al. [2] proposed a dozen commercially vital arithmetic's boasted various word sizes, precisions, misestimating procedures and over/underflow behaviors, and additional were within the works. "Portable" software system meant to reconcile that numerical diversity had become unbearably expensive to develop. 13 years past, once IEEE 754 became official, major microchip makers had already adopted it despite the challenge it exhibit to implementers.

Ykuntam et al. [3] explained Addition is that the heart of arithmetic unit and also the arithmetic unit is commonly the work horse of a machine circuit. thus adders play a key role in planning Associate in Nursing arithmetic unit and additionally several digital integrated circuits. Carry choose Adder is one amongst the quickest adders employed in several information processors and in digital circuits to perform arithmetic operations.

Nachtigal, Michael & Nagarajan et.al. [4]. a reversible floating-point adder has been proposed in this paper which follows the IEEE754 specification for binary floating point arithmetic. The proposed architecture needed reversible designs of a controlled swap unit, a subtracter, an alignment unit, an adder, a normalization unit, and a rounding unit and many more. It can be observed that the floating-point addition is most frequently used floating point operation

Rosemin, Anuja et.al [5] explained a high speed and reduced area floating point unit(FPU) is implemented incorporating fused add subtract unit. The FPU is designed to handle numbers both in single precision and double precision formats. When compared to discrete floating point add-subtract unit, fused floating point add-subtract unit shows better performance. The FPU was designed using VHDL language and implemented on a Xilinx Virtex-II FPGA.

Jain J et. al. [6] described high speed floating point unit using reversible logic. Floating point unit is an important and essential logic design unit in various computational and research logic units. The floating point unit is analyzed in terms of cost, constant inputs, power consumption, speed, garbage outputs and area.

III. PROPOSED TECHNIQUE

In the proposed methodology, the operations of the arithmetic unit is divided into various operations utilizes the IEEE 754 double precision format. The steps used for the implementation are:

3.1 Conditional Swap

In conditional swap unit, firstly all the different parts of the IEEE 754 double precision unit are extracted i.e. significand, exponent and sign. In this unit the exponents are compared and their difference is calculated. According to the difference calculated between the exponents the significands are arranged. The significand with the higher exponent value is assigned to x and the exponent with lower value is assigned to the variable y. Now in order to perform arithmetic operations on floating point unit, they first be converted to the $1.xxxx$ format. For the conversion into this format 1 is concatenated on the MSB side of the significand.

After adding the MSB, now the significand y must be shifted to the right by the value of difference of the exponents. The output of the conditional swap unit is shifted significands x and y and the exponent and sign value which is further supplied to the arithmetic unit.

3.2 Arithmetic Operations

In the second unit the fundamental arithmetic operations are performed i.e. addition, subtraction, multiplication and division. For performing the basic operations the basic ripple carry adder is replaced by the fast and efficient carry select adder. Also the resource sharing among the operations is performed. It means that the addition unit used in the multiplication operation is also used to perform the addition and subtraction operations and also the subtraction performed in the division operation.

The carry select adder is implemented using the basic ripple carry adders which computes the value of the sum and carry on the basis of carry input taken as ‘1’ and ‘0’. The basic ripple carry adder is implemented using the reversible Peres gates. The unit computes the addition of the significand values along with the multiplication and division values. Figure 2 shows the flowchart of the basic arithmetic operations performed by the floating point unit using the reversible gates. The reversible gates present here are used to preserve the input values and can be used for further calculation.

3.3 Post Normalization

Post Normalization operation is performed after the addition operation. In this operation if there is a carry then the exponent in the third unit is also increased or decreased according to the value obtained after the operation. Normalization shift quantity is deducted just in case large cancellation happens throughout the subtraction.

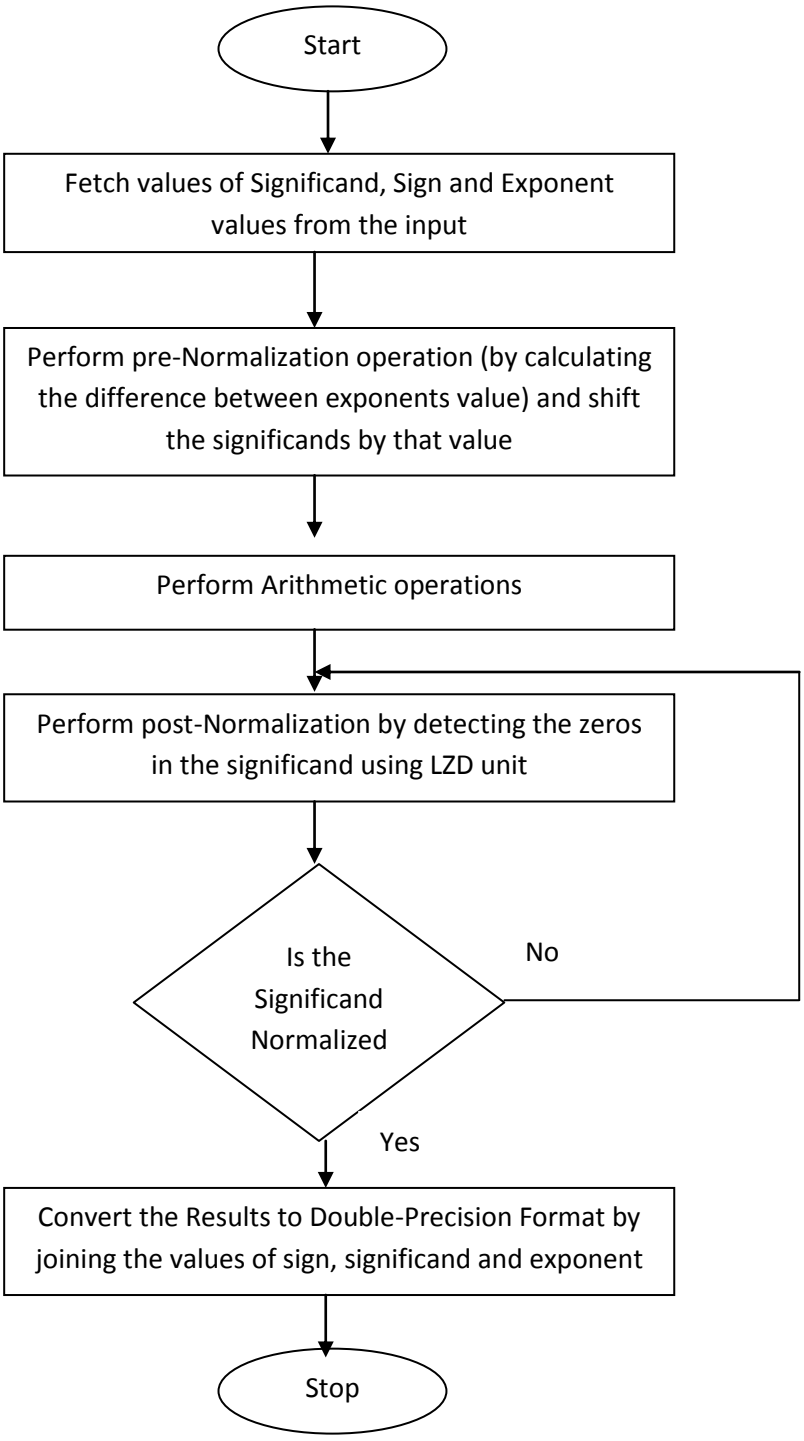

Fig. 3.1: Flow Diagram of the Proposed Methodology

IV. RESULTS AND DISCUSSIONS

The proposed methodology is implemented using the Xilinx Virtex-7 FPGA. The language used for the implementation is VHDL and the environment is Xilinx ISE. Figure 3 shows the top level module of the proposed methodology.

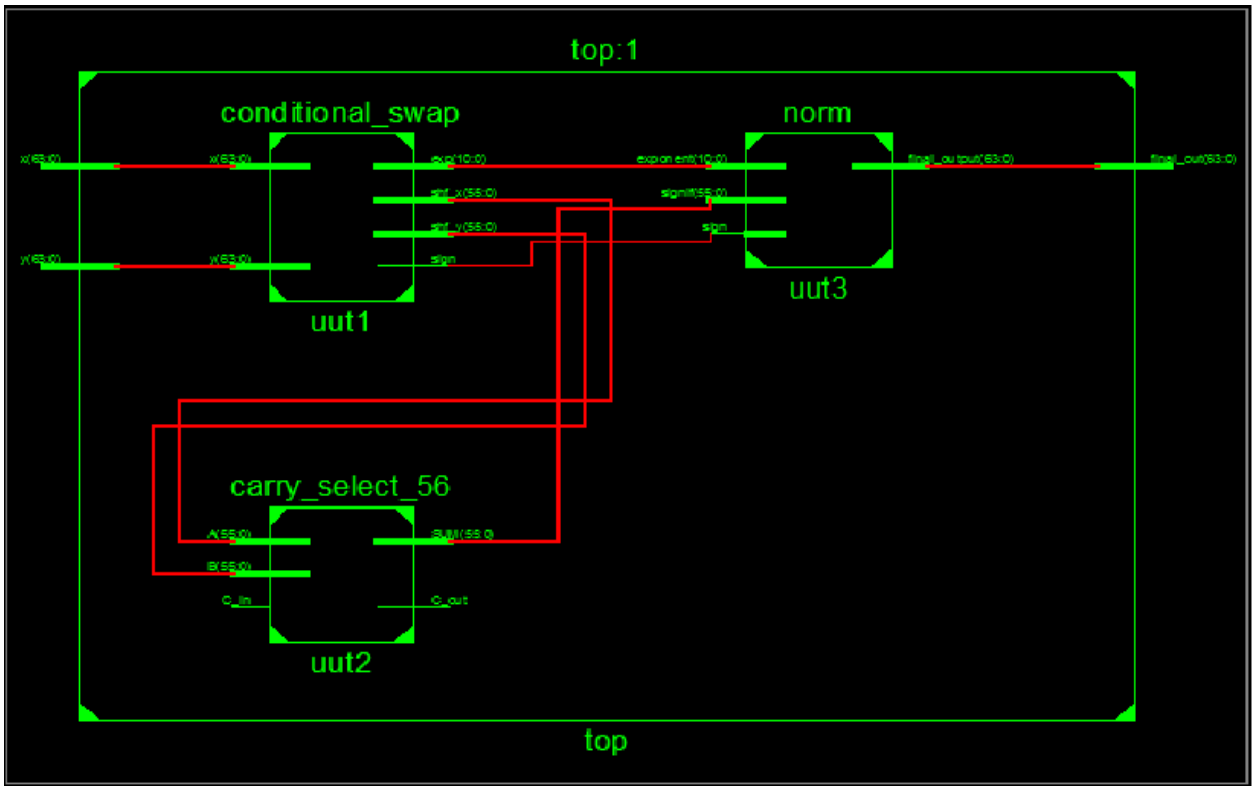

Fig. 4.1: Top Module of the Proposed Approach

Table 4.1 shows the comparison of the proposed and the basic approach.

Parameters	Basic Approach	Proposed Approach
Delay (ns)	68.45	56.297
Number of Slice LUTs	16160/437600	9876/437600
Number of DSP48A1s	9/1680	9/1680
Max Frequency (MHz)	43.30	81.62

The Delay is defined as the critical path delay calculated for the implementation. The critical path Delay is the maximum delay which a circuit must have from input to output.The proposed approach shows the decrease in delay in the proposed approach with an increase in number of devices used. Figure 4 shows the simulation waveforms of the proposed methodology.

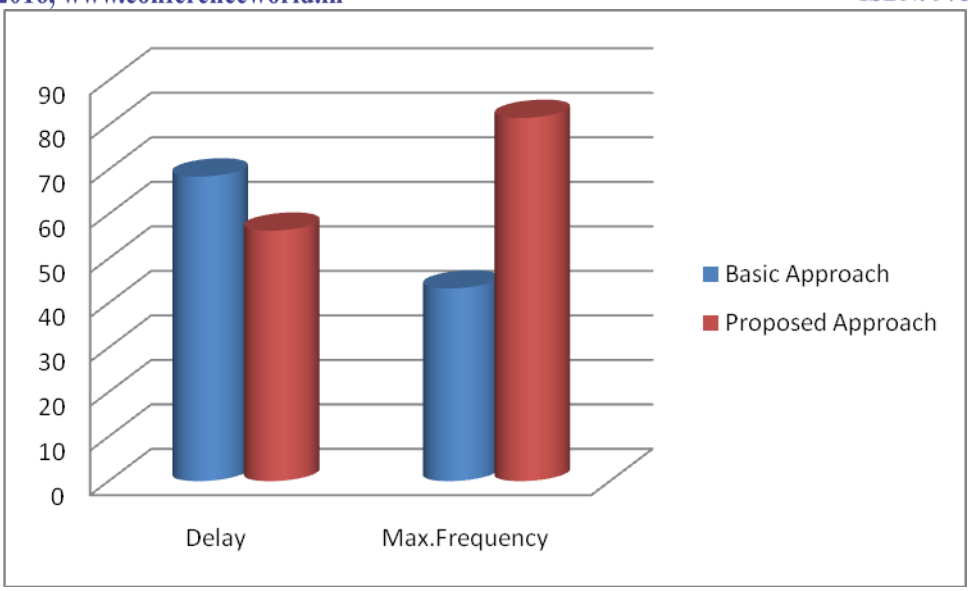

Fig. 4.2: Comparison of delay and frequency of 64-bit Floating Point unit using Virtex-7

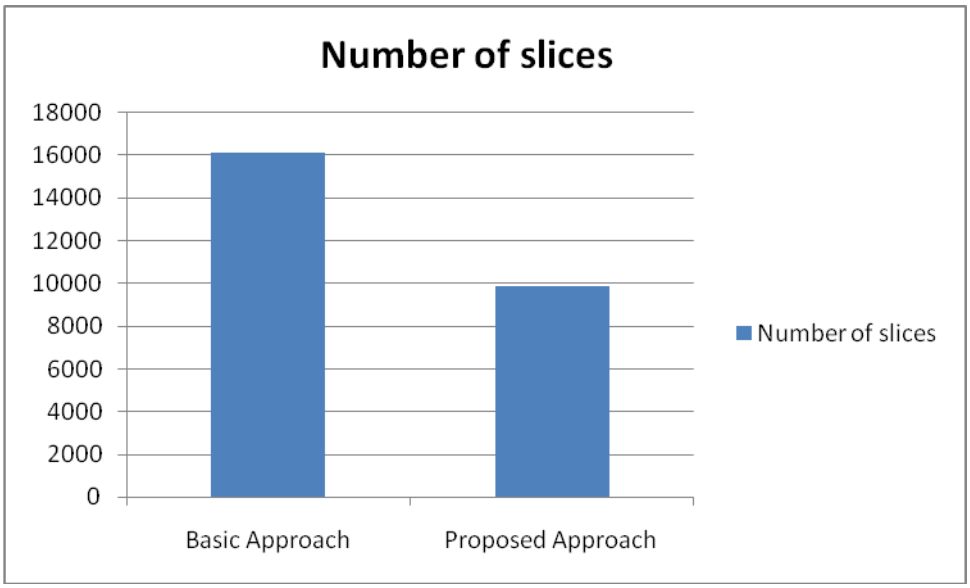

Fig. 4.3: Comparison of Number of slices of 64-bit Floating Point Unit using Virtex-7

The proposed approach shows the decrease in delay i.e. 56.297ns as compared to the basic approach, had delay of 68.45 ns. Similarly the maximum operating frequency of proposed approach is 81.62 MHz which is better than the basic approach's operating frequency i.e. 43.30MHz. Similarly in terms of resource utilization the proposed approach uses less resources as compared to the basic approach.

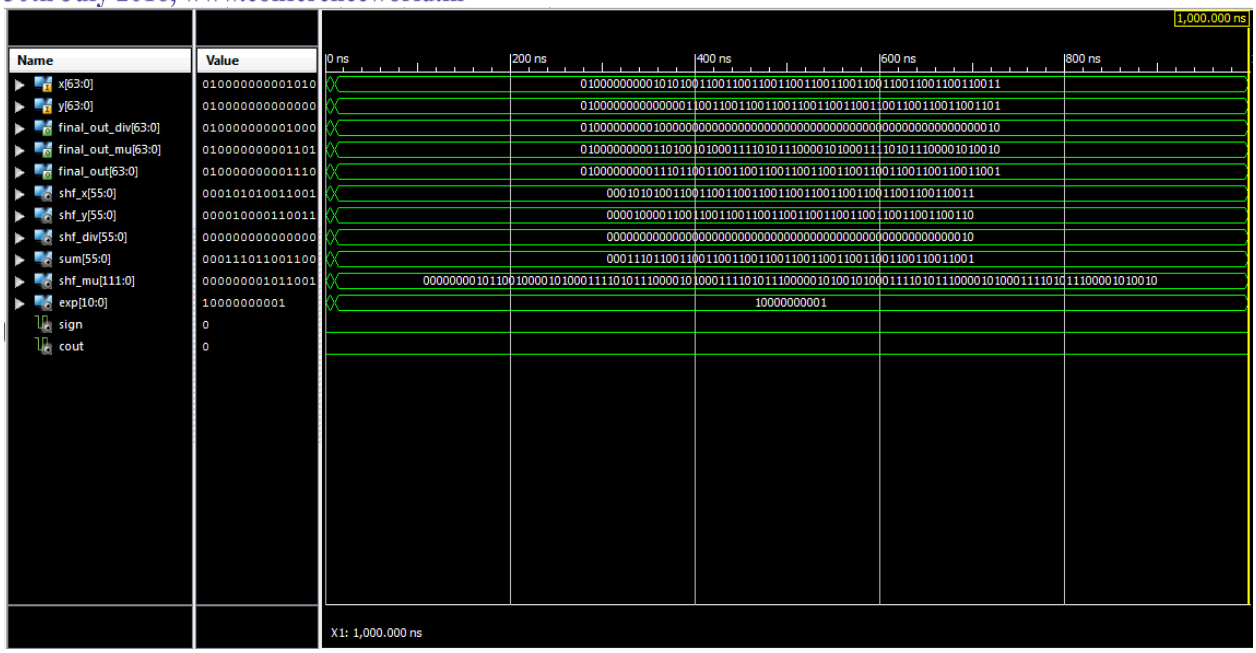

Fig. 4.4: Top Module Simulation

V. CONCLUSION

Floating point unit is an important unit in complex reduced instruction set processors and other digital signal processors. IEEE 754 standard is used for the implementation of the unit with double precision standard. In the proposed technique resources are shared among the various arithmetic blocks along with the parallel approach so as to improve the area utilization of the system . The unit is implemented using the reversible gates which preserve the input for further utilization. The combinational path delay is decreased by 25% which reduces the overall timing complexity of the proposed approach. The implementation of the proposed methodology uses the Carry Select Adder which is fastest adder. In future other units must also be optimized along with the adder to decrease the area utilization.

REFERENCES

[1] "IEEE Standard for Floating-Point Arithmetic", ANSI/ IEEE Standard 754-2008, New York:IEEE Inc., Aug. 29 2008.

[2] Kahan, William. "IEEE standard 754 for binary floating-point arithmetic."Lecture Notes on the Status of IEEE 754.94720-1776 (1996): 11.

[3] Ykuntam, Yamini Devi, MV NageswaraRao, and G. R. Locharla. "Design of 32-bit Carry Select Adder with Reduced Area." International Journal of Computer Applications, ISSN:0975 – 8887, Volume 75, Issue No.2,PP: 47-51, August 2013.

[4] Nachtigal, Michael, Himanshu Thapliyal, and Nagarajan Ranganathan. "Design of a reversible single precision floating point multiplier based on operand decomposition." In Nanotechnology (IEEE-NANO), 2010 10th IEEE Conference on, pp. 233-237. IEEE, 2010.

[5] Kavitha Sravanthi, Addula Saikumar, " A FPGA Based Double Precision Floating Point Arithmaic Unit Using Verilog", International Journal of Engineering Research and Technology, ISSN : 2278-0181, Volume-2, Issue-10, PP 576-581, October 2013.

[6] Jain, Jenil, and Rahul Agrawal. "Design And Development of Efficient Reversible Floating Point Arithmetic unit." Communication Systems and Network Technologies (CSNT), 2015 Fifth International Conference on IEEE, PP 811 – 815, DOI 10.1109/CSNT.2015.215, April 2015.

[7] Pradnya A.Shengale, Vidya Dahake, "Single precision Floating Point ALU", International Research Journal of Engineering and Technology, ISSN : 2395-0056, Volume-2, Issue-2, PP 745-748, May 2015.

[8] Rupali Dhobale, Sonu chaturvedi, "Implementation of 32-Bit Binary Floating Point Adder Using IEEE 754 Single Precision Format", IOSR Journal of VLSI and Signal Processing, ISSN : 2319-4200, Volume-5, Issue-1, PP 50-53, February 2015.