International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 04 Issue: 02 | Feb -2017 www.irjet.net p-ISSN: 2395-0072

Review on 32 bit single precision Floating point unit (FPU) Based on IEEE 754 Standard using VHDL

Sayali A. Bawankar[1], Prof. G. D. Korde[2]

M-tech (VLSI), E&T Department, BDCOE, Sewagram[1]

Assistant Professor (Sr. Scale), E&T Department, BDCOE, Sewagram[2]

Abstract - *Floating point unit (FPU) is a part of computer system specially designed to carry out operation on floating point number. This paper shows review of IEEE floating point unit (FPU) which will perform multiplication, addition, subtraction and division function on 32bit operand that uses the IEEE-754 standard. Floating point numbers representation can support a much wider range of values than fixed point representation. The work is to implement and analyses floating point unit operation and hardware module were implemented using VHDL and synthesized using Xilinx ISE suite.*

***Key Words*: *Floating point unit, IEEE 754, VHDL, Xilinx ISE*.**

1. INTRODUCTION

The digital arithmetic operations are very important in the design of digital processors and application-specific systems. An arithmetic circuit plays an important role in digital systems with the vast development in the very large scale integration (VLSI) circuit technology; many complex circuits have become easily implementable today. Algorithms that are seemed to be impossible to implement now have attractive implementation possibilities for the future. This means that not only the conventional computer arithmetic methods, but also the unconventional ones are worth investigation in new designs. The motion of real numbers in mathematics is convenient for hand computations and formula manipulations. However, real numbers are not well-suited for general purpose computation, because their numeric representation as a string of digits expressed in say, base 10 can be very long or even infinitely long. Examples include pie, e, and 1/3. In practice, computers store numbers with finite precision [8]. Numbers and arithmetic used in scientific computation should meet a few general criteria

- Numbers should have minimum storage requirements.
- Arithmetic operations should be efficient to carry out.
- A level of standardization, or portability, is desirable–results obtained on one computer should closely match the results of the same computation on other computers.

An arithmetic circuit which performs digital arithmetic operations has many applications in digital coprocessors, application specific circuits, etc. Because of the advancements in the VLSI technology, many complex algorithms that appeared impractical to put into practice, have become easily realizable today with desired performance parameters so that new designs can be incorporated [11]. The standardized methods to represent floating point numbers have been instituted by the IEEE-754 standard through which the floating point operations can be carried out efficiently with modest storage requirements. An arithmetic unit is a part of computer processor (CPU) that carries out arithmetic operations on the operands in computer instructions words. Generally arithmetic unit performs arithmetic operations like addition, subtraction, multiplication, division. Some processor contains more than one AU - for example one for fixed operations and another for floating point operations. To represent very large or small values large range is required as the integer representations is no longer appropriate. These values can be represented using IEEE -754 Standard based floating point representation. In most modern general purpose computer architecture, one or more FPUs are integrated with the CPU; however many

embedded processors, especially older designs, do not have hardware support for floating point operations. Almost every languages has a floating point data types; computers from pc's to supercomputers have floating point accelerators; most compliers will be called upon to compile floating point algorithm from time to time; virtually every operating system must respond to floating point exceptions such as overflow [6].

1.1 FLOATING POINT NUMBER

The term floating point is derived from the meaning that there is no fixed number of digits before and after the decimal point, that is, the decimal point can float. There was also a representation in which the number of digits before and after the decimal point is set, called fixed-point representations. In general floating point representations are slower and less accurate than fixed-point representations, but they can handle a larger range of numbers. Floating Point Numbers are numbers that consist of a fractional part. For e.g. following numbers are the floating point numbers: 35, -112.5, ½, 4E-5 etc. Floating point arithmetic is considered a tough subject by many peoples. This is rather surprising because floating-point is found in computer systems. Almost every language supports a floating point data type. A number representation (called a numeral system in mathematics) specifies some way of storing a number that maybe encoded as a string of digits. In computing, floating point describes a system for numerical representation in which a string of digits (or bits) represents a rational number. The term floating point refers to the fact that the radix point (decimal point, or more commonly in computers, binary point) can "float"; that is, it can be placed anywhere relative to the significant digits of the number [7].

1.2 FLOATING POINT UNIT

Floating-point units (FPU**)** equally are a math co-processor which is designed specially to carry out operations on floating Point number. Typically FPUs can handle operations like addition, subtraction, multiplication and division. FPUs can also perform various transcendental functions such as exponential or trigonometric calculations, though these are done with software library routines in most modern processors. When a CPU executes a program that is calling for a floating point (FP) operation, there are three ways by which it can carry out the operation. Firstly, it may call a floating-point unit emulator, which is a floating-point library, using a series of simple fixed-point arithmetic operations which can run on the integer ALU. These emulators can save the added hardware cost of a FPU but are significantly slow. Secondly, it may use an add-on FPUs that are entirely separate from the CPU, and are typically sold as optional add-ons which are purchased only when they are needed to speed up math-intensive operations. Else it may use integrated FPU present in the system [9].The FPU designed by us is a single precision IEEE 754 compliant integrated unit. It can handle not only basic floating point operations like addition, subtraction, multiplication and division but can also handle operations like shifting, square root determination and other transcendental functions like sine, cosine and tangential function

1.3 IEEE-754 STANDARDS

IEEE-754 is a Floating point technical standard established by IEEE in 1985 and the most widely used standard for floating-point computation, followed by many hardware (CPU and FPU) and software implementations. The standard defines five basic formats, named using their base and the number of bits used to encode them. There are three binary floating-point formats (which can be encoded using 32, 64, or 128 bits) and two decimal floating-point formats (which can be encoded using 64 or 128 bits). The first two binary formats are the 'Single Precision' and 'Double Precision' formats of IEEE-754 1985, and the third is often called 'quad': The decimal formats are similarly often called 'double' and 'quad' [8].

1.3.1 SINGLE PRECISION FORMAT

The IEEE Single precision format uses 32 bit for representing floating point number. Most significant bit starts from the left exponent, and mantissa [6].That is divided into three subfields and storage layout for single precision is as shown in fig 1.

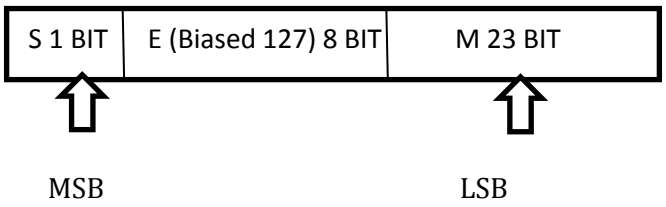

Fig 1: Single precision format

2. LITERATURE REVIEW

I.V. Vaibhav et al. in paper [1] entitled "VHDL Implementation of floating point multiplier using Vedic mathematics", The multiplier is designed in VHDL and simulated using I- Simulator. The design is synthesized using Xilinx ISE 12.1 tool targeting the Xilinx Spartan3E FPGA. A test bench is used to generate the stimulus and the multiplier operation is verified. The over flow and under flow flags are incorporated in the design in order to show the over flow and under flow cases. The result shows the Vedic multiplication method is the efficient way of multiplying two floating point numbers. The lesser number of LUTs shows that the hardware requirement is reduced, thereby reducing the power consumption.

Sushma S. Mahakalkar et al. in paper [2] entitled"Design of High Performance IEEE754 Floating Point Multiplier Using Vedic mathematics", In this paper represents an efficient single and double precision floating point multiplier design. High speed can be achieved using this Vedic multiplier and carry save adder. The design algorithm and the result show that this Vedic multiplier requires less area and high speed as compared to the conventional multiplier.

Prof J. M. Rudagi et al. in paper [3] entitled "Design and implementation of efficient multiplier using Vedic mathematics", in this paper, An efficient Vedic multiplier with high speed, low power and consuming little bit wide area was designed. It was found that the multiplier based on Vedic sutras had execution delay of almost half of that of binary multiplier (partial products method).Hence signal processing can be made faster and efficient.

Rupali Dhobale and Soni Chaturvedi in paper [4] entitled"FPGA Implementation of single precisions floating point adder", In this paper a single precision floating-point adder is implemented. The main contribution of our work is to implement and analyze floating-point addition algorithms and hardware modules were implemented using VHDL and is Synthesized using Xilinx ISE14.2 Suite. The work is to implement and analyse floating-point addition algorithms and hardware modules were implemented using VHDL and is synthesized using Xilinx ISE14.2 Suite.

Prerna Mandolin and Mr. Atush Jain in paper [5] entitled"VHDL Implementation of Addition and Subtraction unit for Floating Point Arithmetic Unit", In this paper presents the implementation of addition and subtraction unit for floating point arithmetic unit. VHDL code is written floating point adder/ Subtractor unit and it has been implemented, tested on Xilinx ISE 13.1. It can be extended to have more mathematical operation like multiplication, division, square root etc.

Somsubhra Ghosh et al. in paper [6] entitled "FPGA Based implementation of a double precision IEEE Floating point adder", In this paper presents a novel technique to implement a double precision IEEE floating-point adder that can complete the operation within two clock cycles. The proposed technique has exhibited improvement in the latency and also in the operational chip area management.

Onkar Singh and Kanika Sharma in paper [7] entitled "Design and implementation of area efficient single precision floating point unit", In this paper presents the implementation of single precision floating point unit which is able to perform the four mathematical operations (addition, subtraction, multiplication, division).The whole design is captured in VHDL and simulated, placed and routed on vertex 5 FPGA from Xilinx. The proposed design of the single precision Floating point unit required less hardware than the previous designs.

Shilpa Kukati, et al. paper [8] entitled "Design and implementation of low power floating point arithmetic unit", In this paper, the low power optimizing technique multi threshold voltage (MVT) is used for reducing the power consumption of arithmetic unit. The power saving for slow High_vt was 84.4% compared to typical library. The synthesis of the

floating point unit is done using cadence RTL complier in 45nm technology. This paper proposes implementation of IEEE floating point (FP) multiplication, addition and subtraction. Arithmetic on IEEE FP numbers imposes more challenges compared to fixed-point arithmetic.

Mr. Anjana Sasidharan and Mr. P. Nagarajan in paper [9] entitled "VHDL Implementation of IEEE 754 floating point unit", In this paper, Arithmetic unit has been designed to perform pack, unpack and rounding arithmetic operations on floating point numbers. The unit has been coded in VHDL and synthesized on model sim5.5. The results of the computation will be identical, independent of implementation, given the same input data Error and error condition in the mathematical processing will be reported in a consistent manner regardless of implementation. In proposed work the pack, unpack and rounding mode was implemented using the VHDL language and simulation was verified.

Miss Pradnya, et al. in paper [10] entitled "Single Precision Floating point unit", In this paper, floating point Arithmetic unit has been designed and suitable algorithm has been developed to perform operations such as addition, subtraction, multiplication and division. The algorithm can be implemented in pipelined way to reduce the delay and increase the computation time for operation. The unit has been coded in VHDL and implemented tested on Xilinx ISE 13.1.

Prashanth B. U. V et al. in paper [11] entitled "Design and Implementation of Floating Point ALU on a FPGA Processor", In this paper, The implemented DSP modules care floating point based ALU computational systems. Finally the simulation waveforms are obtained in the FPGA simulation tools and the simulation waveforms are verified with the hardware design aspects, and matching results are obtained. The design is based on high performance FPGA "Cyclone TI" and implementation is done after functional and timing simulation. The simulation tool used is Modelsim.

Guillermo Marcus et al. in paper [12] "A Fully Synthesizable Single-Precision, Floating-Point Adder/Subtractor and Multiplier in VHDL for General and Educational Use", In this paper FP adder and FP multiplier are presented. Both are available in single cycle and pipeline architectures and they are implemented in VHDL, are fully synthesizable with performance comparable to other available high speed implementations. The design is described as graphical schematics and VHDL code and both are freely available for general and educational use.

3. OVERALL ANALYSIS OF REPORTED WORK

Floating point unit required less hardware. The low power optimizing technique multi threshold voltage (MVT) is used for reducing the power consumption of arithmetic unit. The power saving for slow High_vt was 84.4% compared to typical library. Arithmetic unit has been designed to perform pack, unpack and rounding arithmetic operations on floating point number. Error and error condition in the mathematical processing will be reported in a consistent manner regardless of implementation floating point Arithmetic unit has been designed and suitable algorithm has been developed to perform operations such as addition, subtraction, multiplication and division. The algorithm can be implemented in pipelined way to reduce the delay and increase the computation time for operation. The result between the hardware and software are matching this will clear the gap between hardware implementation and the software simulation.

4. PROPOSED DIAGRAM

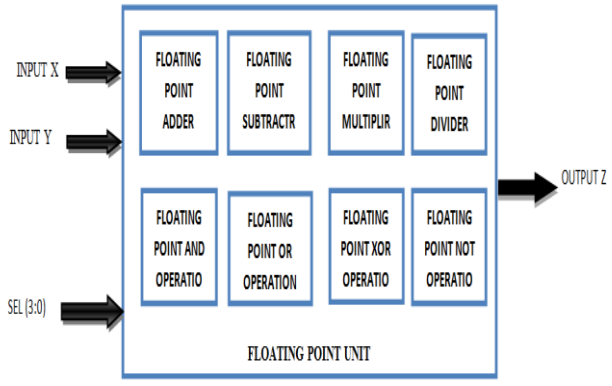

Fig 2: Proposed Block Diagram of Floating Point Unit

In the proposed methodology, the operations of the arithmetic unit are divided into various operations

which utilize the IEEE 754 Single precision format. It supports four arithmetic operations like addition, subtraction, multiplication, division and logarithmic operations. Firstly all the different parts of the IEEE754 Single precision unit are extracted i.e. significand, exponent and sign. In this unit the exponent are compared and their difference is calculated. According to the difference calculated between the exponent the significand are arranged. The significand with the higher value is assigned to variable x and the exponent with the higher value is assigned to variable y. Now in order to perform arithmetic operations on floating point unit, they first be converted to the single precision format. For The conversion into this format variables are concatenated on the MSB side of significand. After adding MSB, now the significand y must be shifted to the right by the value of difference of the exponent. The conditional swap output is shifted to significand x and y and the exponent and sign value which is further supplied to the arithmetic unit. After that fundamental arithmetic operations are performed i.e. addition, subtraction, multiplication and division. For performing the basic operations the basic ripple carry save adder is replaced by the fast and efficient carry select adder. Also the resource sharing operations are 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. At last logarithmic operations are performed. Operations are done by using select lines viz. if select lines are "000" ten floating point addition will perform, if select lines are "001" then floating point subtraction will perform and so on.

5. CONCLUSIONS

In proposed work, we will use floating point representation concept which have large range of values as well as accuracy and also we are using Vedic mathematics sutra. Hence hardware requirement is reduced, thereby reducing power consumption and delay. So, designing of power efficient 32-bit single precision Floating point unit (FPU) based on IEEE-754 standard using Vedic mathematics will be the probable outcome of this research work.

REFERENCES

[1] I. V. Vaibhav, K.V. Saicharan, B. Sravanthi, D. Srinivasulu "VHDL Implementation of floating point multiplier using Vedic mathematics", International Conference on Electrical, Electronics and communications- ICEEC (2014).

[2] Sushma S. Mahakalkar, Sanjay L. Haridas "Design of High Performance IEEE754 Floating Point Multiplier Using Vedic mathematics", 2014 Sixth International Conference on Computational Intelligence and Communication Network.

[3] Prof J M Rudagi, Vishwanath Ambli, Vishwanath Munavali, Ravindra Patil, Vinay kumar Sajjan "Design and implementation of Efficient multiplier using Vedic mathematics", Proc. of 1nt. Con/, on Advances in Recent Technologies in Communication and Computing 2011

[4] Rupali Dhobale, Soni Chaturvedi, "FPGA Implementation of single precisions floating point adder", International Journal of Innovative Research in Computer and Communication Engineering Vol. 3, Issue 6, June 2015

[5] Prerna Mandloi, Mr. Atush Jain, "VHDL Implementation of Addition and Subtraction unit for Floating Point Arithmetic Unit", International Journal for Research in Technological Studies Vol. 1, Issue 10, September 2014 ISSN (online): 2348-1439.

[6] Somsubhra Ghosh, Prarthana Bhattacharyya and Arka dutta, "FPGA Based implementation of a double precision IEEE Floating point adder", seventh international conference on intelligent system and control (ISCO) @2012 IEEE

[7] Onkar singh, Kanika sharma "Design and implementation of area efficient single precision floating point unit", IOSR journal of electronics and communication engineering (IOSR-JECE) Jan 2014

[8] Shilpa Kukati, Sujana D.V, Shruti Udaykumar, Jayakrishnan "Design and implementation of low power floating point arithmetic unit ",2013 international conference on green computing, communication and conservation of energy(ICGCE).

[9] Mr. Anjana Sasidharan, Mr. P. Nagarajan "VHDL Implementation of IEEE 754 floating point unit",

ISBN No.978-1-4799-38346/14/$31.00 @ 2014 IEEE

[10] Miss Pradnya A. Shengale, Prof Vidya Dahake, Prof Mithilesh Mahendra "Single Precision Floating point unit", IRJET Volume:02 issue:02 May-2015

[11] Prashanth B.U. V, Anil Kumar, G. Sreenivasulu "Design and Implementation of Floating Point ALU on a FPGA Processor", International Conferences on Computing, Electronics and Electrical Technologies (ICCEET) 2012 IEEE

[12] Guillermo Marcus, Patricia Hinojosa, Alfonso Avila and Juan Nolazco-Flores", A Fully Synthesizable Single-Precision, Floating-Point Adder / Subtractor and Multiplier in VHDL for General and Educational Use", Proceedings of the Fifth IEEE International Caracas Conference on Devices, Circuits and Systems, Dominican Republic, Nov.3-5,