International Journal of Future Generation Communication and Networking
Vol.14, No. 1, (2021), pp. 4602 - 4607

Floating Point ALU Simulation Using VHDL: A Survey

S.K.Patil[1], S. N. Magdum[2], Nikita Gaikwad[3], Neha Hattalli[4], Anil Patil[5]
Department of E&TC Engineering, SKNCOE, SPPU, Pune, India
[1]skpatil_skncoe@sinhgad.edu
[2]*shraddha.magdum_skncoe@sinhgad.edu*
[3]*gaikwadnikita230@gmail.com*
[4]*nehahattali@gmail.com*
[5]*anilkpatil96@gmail.com*

Abstract: *VHDL environment is implemented for floating point arithmetic and logic unit design using pipelining; the novelty in the pipelining ALU design offers a high-performance ALU to simultaneously execute several instructions. Four arithmetic modules, addition, subtraction, multiplication and division, are combined in the top-down design approach to form a floating point ALU unit. In order to select a specific operation, each module is divided into sub-modules with two selection bits combined. Each module is mutually independent. In the Xilinx12.1i programmed, the modules are realized and tested by VHDL simulation.*

Keywords: *Arithmetic Logic Unit, Top-Down design, Validation, Floating point, Test-Vector*

I. INTRODUCTION

In this paper for representing numbers that would be too large or too small to be represented as integers is defined by floating point. Compared to fixed point representation with small word size, floating point representation will maintain its resolution and accuracy. Numbers in this form are expressed in binary base multiples in scaled form. The number representation is of a significant digit multiplied with exponent power by its origin. The IEEE 754 standard is used in digital systems to represent floating point numbers. The Arithmetic Logic Unit (ALU) is a digital circuit that performs arithmetic and logical operations. The ALU is a basic building block of a computer's central processing unit (CPU) and the inputs to the ALU are the data to be worked on and a control unit code that specifies which operation to perform. The product of the computation is its performance. Cases such as carry-in or carry-out, overflow, divide-by-zero, etc. are indicated by these codes. The Floating Point Unit also performs arithmetic operations between two values, but in floating point representation, they do so for numbers. And the ALU is called an FPU with floating point operations. A pipeline is a technique used to improve teaching throughput in the design of computers and other digital electronic devices (number of instructions that can be executed in a unit of time). The basic concept is to split a machine instruction's processing into a series of independent steps, with storage at the end of each step. This enables the control circuitry of the computer to issue instructions at the slowest phase processing rate, which is much quicker than the time taken to execute all steps at once. The pipeline term refers to the fact that each step carries data at once, and each step is linked to the next, so that the system is idle at no time and output is available at each clock execution after a few steps. The implementation of the pipeline involves the separation and pipelining of different phases of floating point operations into sequential phases. We also proposed a VHDL.

Environment for the design and simulation of floating point ALU for faster units. This has further helped to ease the classification, simulation of verification and realization of hardware.The VHDL

ISSN: 2233-7853 IJFGCN

standard is commonly accepted and has various capabilities suitable for designs of this type. In particular, the use of VHDL for modeling is attractive because it offers a structured system definition and enables the use of specific description styles to cover the different abstraction levels used in this design. For high speed and multiple executions, the synthesis of the proposed design is carried out to assist in broad data size management.

II.LITERATURE SURVEY

AutoRARE: An Automated Tool For Generating FPGA-Based Multi-Memory Hardware Accelerators For Compute-Intensive Applications[1]In this paper, a Java-based automatic design tool to generate hardware accelerators based on the Field Programmable Gate Array (FPGA). All VHDL models required to build/synthesize a processor specifically customized for each application are automatically created by Auto RARE. The user only needs to provide a special-purpose floating point Arithmetic Logic Unit (ALU) or function core with the VHDL definition.The tool produces a VHDL definition for the memory interface, the memory controller, the interface of the host processor, and the particular processor of the application. We also present details of the FPGA-based multi-memory hardware accelerator, generated using Auto RARE, for accelerating computationally intensive applications. The multi-memory hardware accelerator is heavily pipelined and capable of reading and writing multiple floating point values from multiple memories simultaneously**.**The multi-memory architecture is the secret to having faster 10X-100X executed hardware accelerators than traditional multi-core processors. As an application to show the merits of the multi-memory hardware accelerator, the Taylor Series expansion of the sine/cosine functionality is used. We performed the Taylor Series in software in our experiments and compared execution times with an implementation of FPGA-based hardware. Our tests show that the multi-memory Taylor Series hardware accelerator based on FPGA is 481X faster than the Taylor Series running software on a standard serverenvironment for the design and simulation of floating point ALU for faster units.
IMPLEMENTATION OF THE STANDARD FLOATING POINT MAC USING IEEE 754 FLOATING POINT ADDER [2] The standard floating point MAC was implemented in this work using the IEEE 754 floating point adder. This can be used to design, via the standard floating point MAC, all floating point DSP processors.

Advanced verification of Single precision floating point ALU [3] In this paper our daily life, arithmetic is the fundamental process involving operations such as addition, subtraction and multiplication. The Arithmetic Logical Unit (ALU) is used to verify the proper operation of these arithmetic operations, and it is the most important element of a system. The DUT verification of single-precision floating-point arithmetic logical unit (ALU) in the language of Machine Verilog using Questa Sims software is discussed in this paper and the objective is to achieve full functional coverage.The FPU design as format of single precision IEEE754 compliant integrated unit also can't handle only basic floating point operations but also handle operations like shifting, square root determination and operation of transcendental functions like sine, cosine and tangential function.It is a math coprocessor which is designed specially to carry out operations on floating point numbers . The FPUs will perform operations like addition, subtraction, multiplication and division. Main function of FPUs can execute different functions such like as exponential or trigonometric calculations, although these are done with software library routine in nearly all recent processors. Our FPU is basically a 32 bit (single precision) IEEE754.

ISSN: 2233-7853 IJFGCN

Testing of floating point unit using BIST with parallelism [4] In this paper parallel testing, the proposed architecture is altered by splitting the FPU into 3 separate blocks. In this step, the other block of the FPU is checked in parallel when one of the blocks is in normal operation. In Verilog HDL, the RTL code of the design is written, and Xilinx Vivado 2015 is used for simulation. Compared to the traditional method, the proposed method decreases the dynamic power by 10.47 percent.Arithmetic computations can be on integer orfloating(real)numbers. In digital systems, ALU handles arithmetic operations. However, ALU is not suitable for handling operations on real numbers as the result may not be precise and accurate. Hence to perform operations on real numbers digital systems use a dedicated unit called floating point unit(FPU). In this paper, the designed FPU is single precision and operates on IEEE - 754 - 2008 format. The available arithmetic operations on this FPU are floating point multiplication, division, addition and subtraction. The designed FPU can operate on normal(normalized) and subnormal(DE normalized) numbers present in floating point numbers. In this paper, stuck-at fault model using Built in self test(BIST) method is designed for the floating point unit to check the fault in the design. Basic idea behind the BIST is testing the device by itself. The proposed design is modified for parallel testing by dividing the FPU into 3 independent blocks. In this method when one of the blocks is in its normal operation the other block of the FPU is tested in parallel. The design's RTL code is written in Verilog HDL and Xilinx Vivado 2015 is used for simulation. The proposed method reduces the dynamic power by 10.47% compared to the conventional method.

Optimization of an Arithmetic Logic Unit using generic floating point algorithm for 12-Bit Architecture[5] In this paper proposed high speed generic floating point algorithm for 12-Bit Architecture is consist of adder, subtractor, multiplier, divisor, square root, and cube root modules. A novel algorithm was proposed for each modules using VHDL to optimize the speed and area as well as to attain the highest maximum operating frequency. A top down approach was applied for the modules and these were further subdivided into sub-modules for the two inputs to be combined. The novel algorithm was written in VHDL code, simulated in Xilinx 9.2i and 13.1 version and implemented in Xilinx FPGA Vertex boards. The results were analyzed and compared with different algorithms. As the FPGAs' (Field Programmable Gate Array) popularity increases rapidly, they are becoming more suitable in many applications that require dense computations and high frequency. The improvement in the performance of electronic systems can be traced to developments in integrated circuits which form the fundamental building blocks of modern electronics technology. As with the modern technology fast-approaching, the development of integrated circuits can also be attributed with the help of hardware description language (HDL) such as VHDL and Verilog. The need to improve the face of the ICs through the use HDL also defines the speed and area of the whole circuitry. The basis for conceptualizing the design, development, and evaluation of a generic algorithm is to provide a direct impact of improving the existing modules that was implemented on many design applications in DSP. Prior, to the development of the VHDL modules, other studies involving ALU similar to the paper were considered to determine the parameters to be observed. The enhancement of the existing VHDL modules in the other papers are implemented for the ALU and this includes the development of the new modules such as the square and cube root.

Floating point unit core for Signal Processing applications[6] A floating point unit is a feature of a computer system specifically built to perform floating point number operations. In different systems, the floating point unit has been introduced as a coprocessor rather than as an integrated unit. In the design of Digital Signal Processing and application specific systems, today's floating point arithmetic operations are

ISSN: 2233-7853 IJFGCN

very important. Arithmetic logics are quicker and more area-efficient, as Fixed-Point, but calculation using Floating-Point numbers is often desirable. Adding and multiplication is often performed in most optical signal processing applications. This paper presents a review of the Floating Point unit for a signal processing applications, which has faster rate of operations.

III.PROPOSED SYSTEM

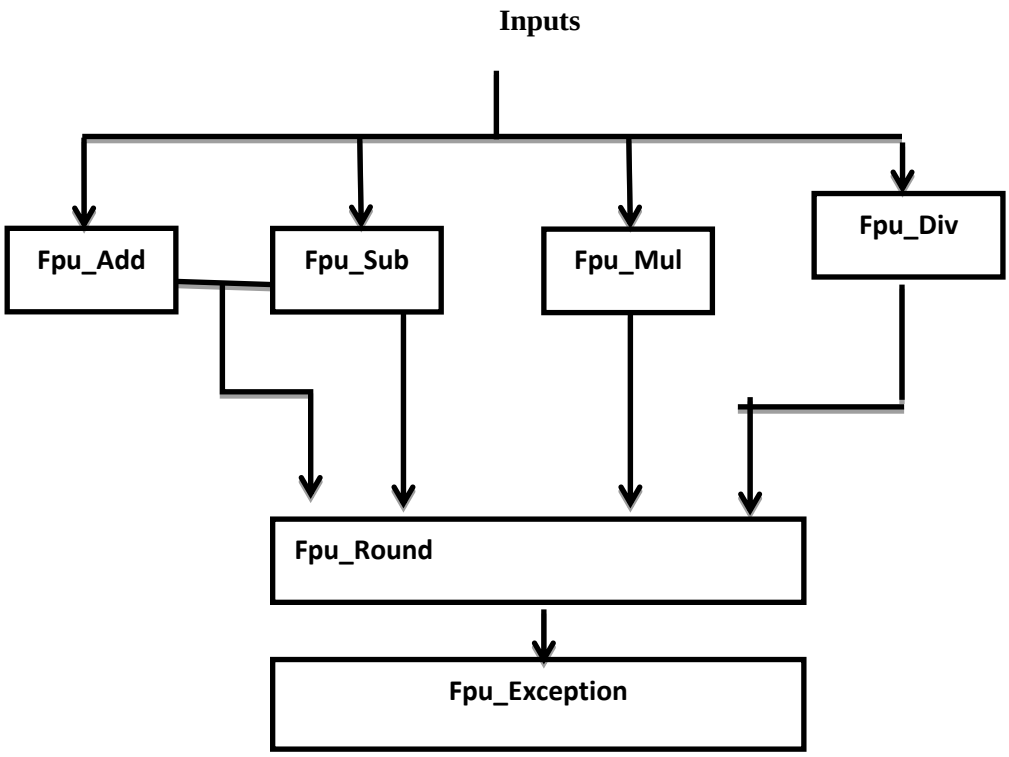

Fig. 1 System Block Diagram

A part of the central processing unit is the arithmetic unit, also called the arithmetic logic unit (ALU) (CPU). It is also referred to as the CPU "driver" because it enables mathematical operations, such as addition, subtraction, and multiplication, to be done by the computer. "The ALU also conducts operations of logic, such as "AND," "OR," and "NOT. When processing, the arithmetic unit works along with the register array, which holds data.

IV.ALGORITHM

The main objective of this paper is to introduce pipelining using VHDL in the configuration of the floating-point ALU. A 8-bit floating point ALU that operates on the IEEE 754 standard is designed for the sub-objectives. The four basic arithmetic operations are provided by floating point representations; addition, subtraction, multiplication and division are defined in this section. The second sub-objective is to use VHDL to model the actions of the ALU design. The Specifications for the implementation of a 8-bit floating-point ALU are:

i. Input A and B and output result are 8-bit binary floating point.

ISSN: 2233-7853 IJFGCN

ii. Operands A and B operate as follows

A (operation) B=results

Operation can be addition (+), subtraction (-), Multiplication (*), division (/)

iii. 'Selection' a 2-bit input signal that selects ALU operation and operate as shown in table1.

iv. Status- a 4-bit output signal work as flag a microprocessor.

v. Clock pulse is only provided to the module which is selected using demux.

vi. Concurrent processes are used to allow processes to run in parallel hence pipelining is achieved by this execution.

ALU operations are divided into smaller modules: addition, subtraction, multiplication and division and is controlled though demux and mux.

V. CONCLUSION

In this paper implementation of a floating point ALU using VHDL using Xilinx tool is done. It will be simulation and will be purely software oriented. This behavioral design can be made synthesizable and thus can be used for layout and fabrication on FPGA based digital circuits in future.By simulation with various test vectors the proposed approach of pipeline floating point ALU design using VHDL is successfully designed tested and implemented currently .we are conducting further research that consider the further reduction in the hardware complexity in terms of synthesis.

REFERENCES

[1] [1] C.Rami Reddy, O.Homa Kesav, A. Maheswara Reddy "High Speed Single Precision Floating Point Unit Implementation Using Verilog" International Journal of Advances in Electronics and Computer Science, ISSN: 2393-2835 Volume-2, Issue-8, Aug.-2015

[2] [2] Amana Yadav and Ila Chaudhary "Design of 32-bit Floating Point Unit for Advanced Processors" Int. Journal of Engineering Research and Application ISSN : 2248- 9622, Vol. 7, Issue 6, (Part -5) June 2017, pp.39-46.

[3] [3] Naresh Grover and M.K.Soni" Design of FPGA based 32-bit Floating Point Arithmetic Unit and verification of its VHDL code using MATLAB" I.J. Information Engineering and Electronic Business, 2014, 1, 1-14 Published Online February 2014 in MECS

[4] [4] Bhaskar Chittaluri "Implementation of Area Efficient IEEE-754 Double Precision Floating Point Arithmetic Unit Using Verilog" International Journal of Research Studies in Science, Engineering and Technology Volume 2, Issue 12, December 2015, PP 15-21

[5] [5] ShaikhShoaibArif and Dr. B. B. Godbole "Multi-Precision Floating Point Arithmetic Logic Unit for Digital Signal Processing" International Journal of Engineering Research in Electronics

ISSN: 2233-7853 IJFGCN

and Communication Engineering (IJERECE) Vol 5, Issue 2, February 2018 [6] YedukondalaRaoVeeranki, R. Nakkeeran “Spartan 3E

[6] [6] L.-K. Wang and M. J. Schulte, Decimal FloatingPoint Adder and Multifunction Unit with Injection-Based Rounding, in Proc. 18th IEEE Symp. Computer Arithmetic ARITH 07,2007,pp.56 68.

[7] [7] M.Vazquez, G. Sutter, G. Bioul, and J.P Deschamps, Decimal Adders/Subtractors in FPGA: Efficient 6input LUT Implementations, inProc. Int. Conf. Reconfigurable Computing and FPGAs ReConFig 09, 2009, pp.42 47.

[8] [8] Xilinx Inc. XST User G uide 12.1, v12.1 ed., Xilinx Inc., June 2009. [Online]. Available: http://www.xilinx.com

[9] [9] Xilinx Inc. Xilinx ISE D esign Suite 12.1 Software Manuals, v12.1 ed., Xilinx Inc., June 2009. [Online]. Available: http://www.xilinx.com

[10] [10] Xilinx Inc. Virtex-5 Li braries Guide for VHDL design, v12.1 ed., Xilinx Inc., June 2009. [Online]. Available: http://www.xilinx.com

[11] [11] Xilinx Inc,DS335: Floating-Point Operator v5.0June 2009

[12] [12] M. Cornea, J. Harrison, C. Anderson, P. Tang, E. Schneider, and E. Gvozdev, A software implementation of the ieee 754r decimal floating-point arithmetic using the binary encoding format, pp. 148 162, 2009, computers, IEEE Transactionson.

[13] [13] M. Cornea, C. Anderson, J. Harrison, P. T. P. Tang, E. Schneider, and C. Tsen, A software implementation of the ieee 754r decimal floating-point arithmetic using the binary encoding format, pp. 29 37, 2007, computer Arithmetic, 2007. ARITH 07.18th IEEE Symposiumon.

[14] [14]The decNumber C library, v3.68 ed., IBM UK Laboratories, January 2010.[Online].

[15] [15] M. Cowlishaw. (2009) Decimal library performance.

[16] WEBSITE:

[17] 1.www.researchgate.com

[18] 2. www.stmjournals.com

[19] 3. www.ieeexplore.ieee.org

[20] 4. www.scibey.com

[21] 5. www.dcitechnol.com

ISSN: 2233-7853 IJFGCN