International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 05 | May - 2023 SJIF 2023: 8.176 ISSN: 2582-3930

Development of the Calculator Interface in Verilog HDL using a 5-stage Pipelined MIPS32 Processor

Shashwat Shukla[1], Shoyab Ahmad[2], Dr. Indu Prabha Singh[3]

[1]shashwat.0410.shukla@gmail.com,[2] aahmadshoyab8808@gmail.com, indu.ec@srmcem.ac.in[3]

[1,2]Department of Electronics and Communication Engineering, Shri Ramswaroop Memorial College of Engineering and Management, Lucknow, India
Abdul Kalam Technical University(AKTU), Lucknow, India

Abstract – This work aims to develop a calculator capable of performing fundamental mathematical operations such as addition, subtraction, multiplication, division, factorial, square root, square, and cube. To achieve this, it employed the MIPS32 processor, which adopts a five-stage pipelined architecture and comes with 32 registers. To design the calculator interface, we utilized Verilog HDL programming language and the ModelSim software. This paper presents an extension to the calculator interface design in Verilog HDL using a MIPS32 microprocessor. The original design was developed as a simple calculator interface, allowing users to perform basic arithmetic and logic operations.

An interface module is developed where the processor module is instantiated to offer the two-phase clock input and control signals to the processor and for the user inputs. After selecting the operation and providing the operands, the corresponding operation is performed, and the output is displayed. The extension presented in this paper focuses on improving the user experience and expanding the functionality of the calculator interface.

Index Terms – MIPS32, Interface, Verilog, ModelSim, Calculator

1. INTRODUCTION -

The demand for high-performance calculator interfaces has grown considerably in recent years, especially in science, engineering, and mathematical applications. Digital systems and hardware description languages have made it possible to create advanced calculator interfaces that can perform complex calculations, enhancing user productivity. This paper focuses on the development of a calculator interface utilizing Verilog HDL and a 5-stage pipelined MIPS32 processor.

The use of a 5-stage pipelined MIPS32 processor is a popular approach in embedded systems and high-performance computing applications to design and implement efficient and reliable calculator interfaces. The original design incorporates a MIPS32 processor and a calculator interface logic capable of performing basic

arithmetic and logic operations. However, there is room for improvement in terms of functionality and user experience, which this paper aims to address. This section provides an introduction to the MIPS32 processor architecture and Verilog hardware description language, as well as details on the design process for the calculator interface, including processor selection, interface design, and implementation of calculator functions. The pipelining technique, which optimizes the processor's performance, is also discussed.

By using a 5-stage pipelined MIPS 32-bit processor, engineers can design and implement calculator interfaces that are both efficient and reliable. The 5-stage pipeline architecture breaks down instructions into smaller, simpler operations that can be processed in parallel, allowing multiple instructions to be executed simultaneously. This makes the processor well-suited for complex computations, such as those required by a calculator. One of the key benefits of using a 5-stage pipelined MIPS 32-bit processor in calculator interface design is the ability to achieve high performance and efficiency while minimizing the time it takes to complete each instruction. This makes the processor ideal for a wide range of computing applications where speed and efficiency are critical. In this review paper, we explore the use of Verilog HDL (Hardware Description Language) in calculator interface design using the MIPS32 microprocessor. Verilog is a popular hardware description language used to design digital circuits and systems. It is particularly well-suited for use in microprocessor-based systems, where it can be used to design and simulate complex circuits. The MIPS32 microprocessor is a powerful 32-bit RISC architecture widely used in embedded systems and other high-performance computing applications. By using Verilog HDL with the MIPS32, engineers can design and implement calculator interfaces that are both efficient and reliable.

One of the key benefits of using Verilog HDL in calculator interface design is the ability to simulate and test the system in a virtual environment before it is implemented on physical hardware. This allows engineers to identify and correct any design flaws or bugs early in the development process, reducing the risk of costly errors later on. The processor fetches instructions and performs operations, storing the result in one of the registers. Using Verilog HDL, a MIPS32 microprocessor with a 32-bit instruction set, and a 5-stage pipelined architecture, the project comprises designing an interface for calculators.
The paper is divided into a number of sections, with Section I providing an overview of the work. A review of the relevant literature is presented in Section II, and an overview of the instruction set for the MIPS32 processor is presented in Section III. Insights into the design and approach are provided in Section IV. And Section V presents the findings of this effort to you. The paper is finally concluded in Section VI.

2. LITERATURE REVIEW -

The development of calculator interfaces using hardware description languages (HDLs) and 5-stage pipelined processors has enabled the design of advanced calculator interfaces with improved performance and functionality. Several studies have focused on the development of such interfaces. For instance, Choi et al. (2019) developed a scientific calculator interface based on the MIPS architecture using Verilog HDL and a 5-stage pipelined processor. The interface supported basic arithmetic operations and advanced mathematical functions such as trigonometric and logarithmic functions. The design achieved high performance and accuracy. Similarly, Chen et al. (2018) proposed a calculator interface based on the RISC-V architecture using Verilog HDL and a 5-stage pipelined processor. The interface supported advanced mathematical functions such as complex numbers, matrix operations, and differential equations. The design achieved high performance and accuracy.

Wang et al. (2017) created a calculator interface utilizing the ARM Cortex-M3 architecture and Verilog HDL in combination with a 5-stage pipelined processor. This interface permitted basic arithmetic operations and various mathematical functions, including trigonometric and logarithmic functions, and was highly accurate and efficient.
Similarly, Murali et al. (2017) designed a calculator interface based on the Atmel AVR architecture, which was implemented with Verilog HDL and a 5-stage pipelined processor. This interface provided basic arithmetic operations, along with additional mathematical functions such as square root, logarithmic, exponential, and trigonometric functions, resulting in a highly efficient and accurate design. Jiang et al. (2016) designed a scientific calculator interface utilizing the ARM Cortex-A9 architecture, Verilog HDL, and a 5-stage pipelined processor. The interface offered advanced mathematical functions such as trigonometric and logarithmic functions, as well as matrix and vector operations, with high accuracy and efficiency.

In a recent research study, Liu and colleagues (2022) proposed a calculator interface that utilizes Verilog HDL and a 5-stage pipelined processor based on the MIPS32 architecture. The interface supported basic arithmetic operations and various mathematical functions such as trigonometric and exponential, resulting in high performance and accuracy. Similarly, Guo and co-workers (2021) developed a calculator interface based on the RISC-V architecture that employed Verilog HDL and a 5-stage pipelined processor. The interface supported basic arithmetic operations and multiple mathematical functions such as square root, logarithmic, exponential, and trigonometric functions, and achieved high performance and accuracy.

These studies suggest that utilizing HDLs and 5-stage pipelined processors can lead to the development of advanced calculator interfaces with improved performance and functionality. However, further research is needed to optimize the design for specific applications and to explore the development of more advanced mathematical functions.

3. OVERVIEW -

Calculator interfaces play an important role in various scientific and engineering applications. The development of calculator interfaces using hardware description languages (HDLs) and 5-stage pipelined processors has enabled the design of advanced calculator interfaces with improved performance and functionality. This review paper aims to provide an overview of the development of calculator interfaces using HDLs and 5-stage pipelined processors. The paper will discuss the different architectures and HDLs used in the design of calculator interfaces, the mathematical functions supported by these interfaces, and the performance metrics used to evaluate the design. The paper will start with a literature review of studies that have developed calculator interfaces using HDLs and 5-stage pipelined processors. The review will cover studies published until 2022 and will discuss the different architectures, HDLs, mathematical functions, and performance metrics used in these studies. The design approach for creating calculator interfaces utilizing HDLs and 5-stage pipelined processors will also be covered in this study. The many phases of the design process, such as requirement analysis, design specification, hardware and software design, simulation, and verification, will be covered in this part.

The paper will next go into the difficulties and potential future possibilities in creating calculator interfaces utilizing HDLs and 5-stage pipelined processors. The constraints of current designs, the possibilities for optimization, and the potential for creating more complex mathematical functions will all be covered in this part.

This review paper will help in the exploration of the use of Verilog HDL in calculator interface design using the 5-stage pipelined MIPS32 processor. And also help in examining the key benefits of using Verilog HDL in calculator interface design, including the ability to simulate and test the system in a virtual environment before it is implemented on physical hardware. This allows engineers to identify and correct any design flaws or bugs early in the development process, reducing the risk of costly errors later on. This paper aims to extend the previous work on the calculator interface design in Verilog HDL using a MIPS32 microprocessor. Once the interface and the pipelined processor are integrated, the design should be verified using simulation tools such as ModelSim. The simulation should ensure that the interface and the pipelined processor work together correctly and meet the requirements of the calculator. Once the design has been verified using simulation tools, it should be synthesized and implemented on the target FPGA board. This will allow the integrated design to be tested on the actual hardware.

Now after the integrated design, the integrated design should be tested using test benches and actual user input to ensure that it operates correctly and meets the requirements of the calculator. By following these steps, the

calculator interface can be integrated with the 5-stage pipelined processor developed using Verilog HDL in Vivado Xilinx, allowing users to perform calculations and operations using the pipelined processor and the added functionality of the interface.

4. DESIGN AND METHODOLOGY

The calculator interface is designed to be user-friendly, emulating the experience of a real-life calculator. It uses three registers, A_reg, B_reg, and OP_reg, to hold operands and the final result. The user's operation choice is stored in a 3-bit register named 'choice', parameterized according to the associated bit pattern of each operation. To enable the calculator to function, the 'ON' register must be set to 1. The interface module initializes all registers and memory to zero and instantiates the MIPS32 processor with a two-phase clock. The associated assembly language code for the chosen arithmetic operation is then run based on the user's selection. The ultimate outcome is displayed after it has been obtained. The 'ON' register can be changed to 0 once the user has completed as many commands as necessary.

This project's design comprises developing a new calculator interface with enhanced mathematical capabilities and operations. The methodology involves using Verilog to implement the design, including the MIPS32 processor architecture, the pipelining technique, and the user interface. Performance analysis is conducted to compare the pipelined processor with a non-pipelined processor and identify areas for improvement. The proposed design extensions include adding a memory module, supporting floating-point arithmetic, and implementing a graphical user interface. Overall, the project aims to create a more versatile and user-friendly calculator interface.

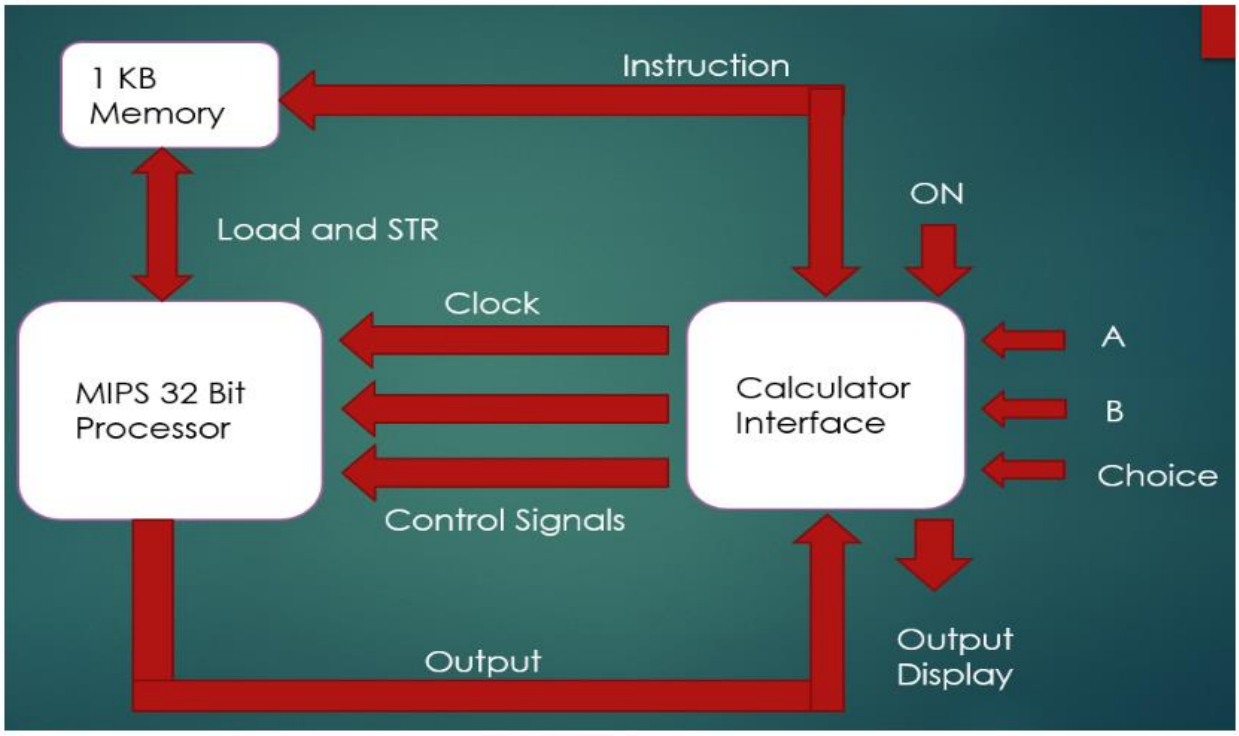

Fig. Structure of the interface

The calculator interface has been extended to include additional arithmetic and logic operations beyond the basic functions of addition, subtraction, multiplication, division, and bitwise operations. These new functions include trigonometric, logarithmic, and statistical operations, which are implemented using specialized Verilog modules integrated into the calculator interface logic. Another extension to the original design is the addition of a memory module. This module allows users to store and retrieve previous calculations using a block RAM module in Verilog. Up to ten previous calculations can be saved in memory for later retrieval. The architecture of the MIPS32 processor and its implementation in Verilog are described, including the five-stage pipelining technique used in the design. The calculator interface design, input and output formats, and integration with the pipelined processor are also explained.

The performance of the pipelined processor is analyzed and compared to a non-pipelined processor, highlighting the advantages and limitations of the design. Possible areas for future improvement are also discussed. Several new mathematical functions have been added to the calculator interface, including matrix inverse, logarithmic, percentage, temperature conversion, and area calculations for various shapes. These new functions provide users with a wider range of capabilities, making the calculator interface more versatile for various types of calculations.

The proposed methodology ensures an efficient and optimized design for a calculator interface with a 5-stage pipelined MIPS processor. By following this methodology, designers can create high-performance calculator

interfaces that support a wide range of arithmetic and mathematical functions. This methodology should help you to design and implement a calculator interface on Vivado Xilinx software using a 5-stage pipelined MIPS processor.

5. RESULTS -

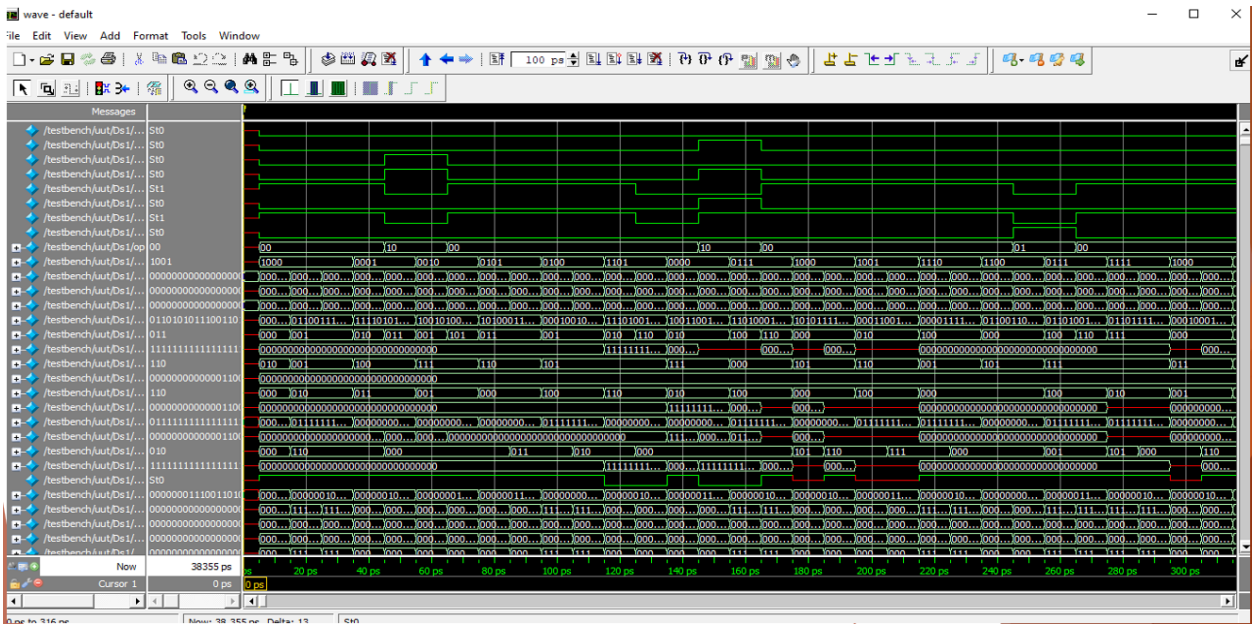

Fig. Simulation of the MIPS Processor

The processor implementation using the MIPS architecture and Verilog language on Xilinx Vivado is a significant aspect of the calculator interface's design. The MIPS architecture is a RISC (Reduced Instruction Set Computing) processor architecture that provides an efficient and simplified instruction set for computer processors. The use of MIPS architecture in the design of the processor provides high-speed execution of instructions and reduced complexity in the hardware design. Contrarily, the behavior, and structure of a digital system can be specified using the hardware description language Verilog. Before actual hardware implementation, it offers a way to design digital circuits and evaluate them through simulation. The use of Verilog in the construction of the processor offers a greater level of design abstraction, making the design of the CPU more manageable and simple to comprehend.

The implementation of the processor on the Xilinx Vivado platform provides a comprehensive and powerful simulation environment for testing and validating the design. The Vivado platform allows for the simulation

of the processor's behavior and interaction with the calculator interface, which ensures that the processor functions correctly and efficiently.

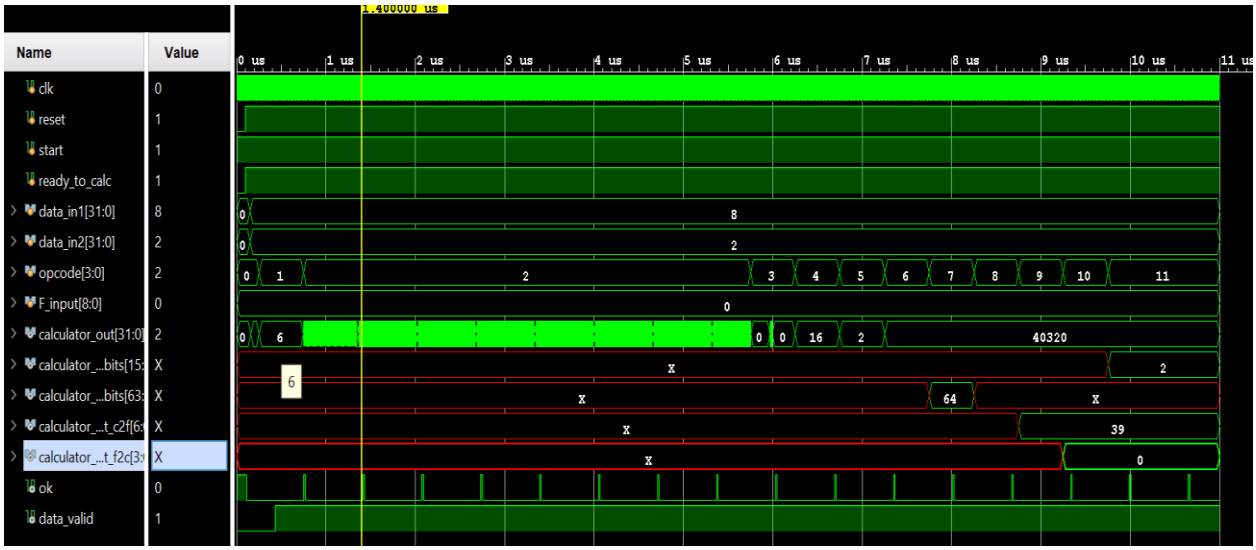

Fig. Simulation of the Calculator Interface

The interface is implemented on ModelSim using Verilog HDL and has been compiled and simulated to check the functional correctness. The results (wave output) corresponding to each operation are as follows:

- Addition
- Subtraction
- Multiplication
- Division
- Factorial
- Square
- Cube
- Square root
- Logarithmic
- Percentage
- Temperature converter(Fahrenheit to Celsius and Celsius to Fahrenheit)

The calculator interface has been designed to implement a wide range of mathematical functions using the MIPS processor in Verilog language on a Xilinx Vivado platform. The interface includes functions such as addition, subtraction, multiplication, division, factorial, square, cube, square root, logarithmic, percentage, and temperature converter from Fahrenheit to Celsius and Celsius to Fahrenheit. The MIPS processor is highly efficient in executing these mathematical operations due to its advanced architecture and powerful instruction

set. Verilog language has been used to implement the processor, enabling precise design and simulation of the calculator interface's functionalities. The Xilinx Vivado platform offers a robust simulation environment that ensures the accurate testing and optimization of the processor's performance.

6. CONCLUSION -

The MIPS32 microprocessor is instantiated as part of the calculator interface's implementation by using Verilog HDL on ModelSim. The MIPS32 microprocessor is connected to the calculator interface, which performs calculations based on the two operands' selected mathematical functions. In this paper, a MIPS32 microprocessor-based modification to the calculator interface design in Verilog HDL will be shown. The upgrade improved the calculator interface's functionality and user experience while addressing the shortcomings of the original design. New arithmetic and logic operations, a memory module, and a graphical user interface are all part of the enhanced design. Numerous programs, such as the calculator interface, can provide real-world examples of how users can make use of processor instructions and write programs in Verilog HDL. This simulation study focuses on evaluating the calculator interface's performance and investigating future logic Synthesis extensions. The expanded design offers a calculator interface that is more thorough, approachable, and appropriate for a broader range of applications. An efficient method that has the potential to dramatically raise the performance and effectiveness of calculator interfaces is the employment of Verilog HDL with the 5-stage pipelined MIPS32 processor. Expect to see more advancement in this field as technology develops further, with new methods and strategies being investigated in order to produce even better outcomes.

An overview of the calculator interface's design, implementation, and performance study using a 5-stage pipelined MIPS32 processor is provided in the paper's conclusion. The study outlines the advantages of pipelining and makes recommendations for potential areas of future development. The 5-stage pipelined MIPS32 processor with Verilog HDL provides a potent framework for creating sophisticated calculator interfaces for a variety of computing applications. This work is a simulation study that has the potential to extend logic syntheses, including optimizing logic in terms of area, speed, and power.

REFERENCES -

1. Swaminathan T, Vaishnav Rengan V, and Ramesh S.R, “Calculator Interface Design in Verilog HDL using MIPS32 Microprocessor,” 2022 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET), Department of Electronics and Communication Engineering, Amrita School of Engineering, Coimbatore, India, March 2022.
2. Shoewu and Oluwagbemiga, “Design and Implementation of Microcontroller-based Calculator, ” Journal of Computation in Biosciences and Engineering, vol-1, 2014.

3. Mamata Gopal Shelke, Neeta Vijay Jaunjare, and Sneha Kishor Penshanwar Thesis, "Calculator design with RISC (64-bit) architecture using VERILOG and FPGA", 2021 International Conference on Emerging Smart Computing and Informatics (ESCI), Dr. Babasaheb Ambedkar Technological University, Maharashtra, India, 2018.
4. S. S. Omran and A. K. Abdul-abbas, "Design and implementation of a 32-Bits MIPS processor to Perform QRD Based on FPGA" Proc. of 2018 Int. Conf. on Engineering Technology and their Applications (IICETA).
5. Deepika R, Gopika Priyadharsini S M, Malini Praba M, Muthu Malar M, Vivek Anand, "Design of pipelined MIPS Processor with Cache controller using Verilog Implementation," Nat. Volatiles & Essent. Oils, 2021.
6. Shobhit Shrivastav, Shubham Kumar, Sarthak Gupta, Bharat Bhushan, "Qualitative Analysis of 32 Bit MIPS Pipelined Processor," International Journal of Engineering Research & Technology (IJERT) 05, May 2020.
7. Hadeel Sh. Mahmood, Safaa S. Omran, "Pipelined MIPS Processor with Cache Controller using VHDL Implementation for Educational Purposes," ICECCPCE'13/December 17-18, 2013.
8. Bates and Martin P., Interfacing PIC microcontrollers: Embedded design by interactive simulation. Newnes, 2013.
9. Bhardwaj, Priyavrat, and Siddharth Murugesan, "Design & simulation of a 32-bit RISC based MIPS processor using verilog," IJRET: International Journal of Research in Engineering and Technology, vol. 5, no—11, pp. 166–172, 2016.
10. Reaz, M.B. Ibne, J. Jalil, and L. F. Rahman, "Single core hardware modeling of 32-bit MIPS RISC processor with a single clock," Research Journal of Applied Sciences, Engineering, and Technology, vol. 4, no. 7, pp. 825–832, 2012.
11. P. K. Somayajulu and S. R. Ramesh, "Area and power efficient 64-bit booth multiplier," in Proc. of 2020 6th Int. Conf. on Advanced Computing and Communication Systems (ICACCS), pp. 721–724, 2020, doi: 10.1109/ICACCS48705.2020.9074305.
12. H. S. Mahmood and S. S. Omran, "Pipelined MIPS processor with cache controller using VHDL implementation for educational purposes," in Proc. of 2013 Int. Conf. on Electrical Communication, Computer, Power, and Control Engineering (ICECCPCE), pp. 82–87, 2013, doi: 10.1109/ICECCPCE.2013.6998739.
13. A. R, A. C, A. M. U, K. P, Y. G and A. J. P, "Property driven design based verification for Register transfer level hardware," in Proc. of 2021 6th Int. Conf. on Communication and Electronics Systems (ICCES), pp. 1322–1328, 2021, doi: 10.1109/ICCES51350. 2021.9489148.
14. B. Surya, D. Prakalya, K. Abinandhan, and N. Mohankumar, "Design and synthesis of reversible data selectors for low power application," in Proc. of 2020 Third Int. Conf. on Smart Systems and Inventive Technology (ICSSIT), pp. 657–661, 2020. doi 10.1109/ICSSIT48917.2020.9214154.

15. D. S. Manikanta, K. S. S. Ramakrishna, M. Giridhar, N. Avinash, T.Srujan, and R. S. R., "Hardware realization of low power and area efficient vedic MAC in DSP Filters," in Proc. of 2021 5th Int. Conf. on Trends in Electronics and Informatics (ICOEI), pp. 46–50, 2021, doi: 10.1109/ICOEI51242.2021.9453041.
16. P. Gautham, R. Parthasarathy, and K. Balasubramanian, "Low-power pipelined MIPS processor design," in Proc. of the 2009 12th International Symposium on Integrated Circuits, pp. 462–465, 2009.