iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

# RAM AND ROM DESIGN USING VERILOG

**Shreyas N[1], Tharun C[1], Thrisha Bai D[1], Shreyas D Nadiger[1], Suma V Shetty[2]**
UG Student[1], Assistant Professor[2]
Department of Electronics and Communication Engineering,
Sapthagiri College of Engineering, Bengaluru,
Affiliated to Visvesvaraya Technological University, Belagavi.

**ABSTRACT**
This project focuses on the design and simulation of Random Access Memory (RAM) and Read-Only Memory (ROM) modules using Verilog HDL. The RAM module supports read and write operations, while the ROM module allows only read operations with pre-initialized data. Both memory modules are designed to accommodate an 8-bit data width and handle up to 8 memory allocations. A comprehensive testbench was developed to verify the functionality of the RAM and ROM designs. The testbench simulates memory operations, including address selection, data read/write, and output validation. Simulation waveforms were analyzed to confirm the correct behavior of the memory modules. This project was implemented on EDA Playground, utilizing its simulation environment to debug and verify the design. The work provides a foundational understanding of digital memory design, Verilog HDL programming and simulation methods, with the potential for future expansion and integration into more advanced systems, including FPGA-based applications. The project successfully shows how RAM and ROM components can be efficiently modeled, tested, and validated using hardware description languages, contributing to the development of robust and scalable digital systems.

**Keywords:**
Random Access Memory, Read-Only Memory, Verilog HDL, memory allocations, Simulation waveforms.

## INTRODUCTION

Memory is a fundamental component of any digital system, providing storage and retrieval capabilities for data and instructions. In digital design, memory units are categorized as volatile (RAM) and non-volatile (ROM). Random Access Memory (RAM) allows both read and write operations, making it suitable for temporary data storage during program execution. In contrast, Read-Only Memory (ROM) is a non-volatile storage medium used to store fixed data or instructions that remain constant throughout the system's operation. This project aims to design and simulate 8x8 RAM and ROM modules using Verilog HDL (Hardware Description Language). Verilog is widely used in digital design for modeling hardware components and simulating their behavior before hardware implementation. The RAM module in this project supports read and write operations synchronized with a clock signal, while the ROM module stores pre-initialized data that can only be read. The design is verified through testbenches and simulation waveforms using EDA Playground, an online HDL simulation environment. This project provides a hands-on experience with digital memory design, Verilog coding, and functional simulation. It forms a crucial step toward understanding the role of memory in complex systems and prepares for future exploration of more advanced memory architectures and FPGA-based implementations

## OBJECTIVES

The main goal of this research is to explore and design efficient memory architectures, including single-port RAM, dual-port RAM, and ROM, using Verilog for hardware implementation on FPGA platforms. The study aims to investigate optimization techniques to enhance the performance of these memory modules in terms of speed, resource utilization, and power efficiency. An important focus is on analyzing the impact of simultaneous memory access in dual-port RAM and identifying its advantages in high-performance systems compared to the simpler single-port RAM. Additionally, the research seeks to assess the efficiency of ROM for static data storage, highlighting its role in embedded systems. The project emphasizes evaluating the complexity, design trade-offs, and resource requirements of each memory type. Through simulation, synthesis, and hardware testing

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

using Xilinx Vivado, the research aims to validate the functionality and performance of the designed modules. A comparative study between single-port RAM, dual-port RAM, and ROM will be conducted to understand their suitability for different applications based on factors like speed and memory efficiency. Addressing challenges such as contention management in dual-port RAM and resource limitations on FPGA is a key aspect of the research. Ultimately, this project seeks to provide insights into the design of memory systems for embedded applications and offer recommendations for future advancements in memory design and optimization techniques.

## LITERATURE SURVEY

### A. Energy-Efficient Design of RAM and Its Implementation on FPGA

B. Pandey et al., [1] There are many systems which use RAM as its primary memory for example spacecraft, our home computer etc. Energy efficient RAM is even more important in outer space exploration as the power source is very limited and the environmental condition is extreme. Energy efficiency is not only based on the architecture of the system but also on the environmental condition. For that reason, low power RAM will increase the performance of the system even in extreme conditions. In this project, we are advancing towards a design of RAM that will consume less power even in extreme conditions. Four contemporary FPGAs are used for implementation of our RAM design and we find the least power consumer among these four FPGAs architectures. We find a 20.24% reduction in power consumption at 5pF, when we migrate our RAM design from Kintex-7 architecture to Artix-7 Architecture. The results are gathered by using Xilinx Vivado 19.1 tool with VERILOG hardware description language.

### B. SSTL I/O Standard based environment friendly energy efficient ROM design on FPGA

M. Bansal et al., [2] Stub Series Terminated Logic (SSTL) is an input/output standard designed to match the impedance of the line, port, and device in a given design. Choosing an energy-efficient SSTL I/O standard from the available SSTL logic families in FPGA plays a crucial role in optimizing power consumption in the design under test (DUT), which in this case is ROM. As an essential component of a processor, ROM's energy-efficient design contributes to the overall efficiency of the system. Similarly, optimizing RAM plays a fundamental role in developing energy-efficient processors. For this project, we utilize Verilog hardware description language, the Virtex-6 FPGA, and the Xilinx ISE simulator. ROM is tested at the highest operating frequency supported by a 4th-generation Intel Core i7 processor to ensure compatibility with modern hardware. To improve energy efficiency, when peak performance is not required, operating the device at 1 GHz instead of 4 GHz can reduce clock power by 74.5%, signal power by 75%, and I/O power by 30.83%. While clock and signal power remain unaffected, the SSTL2_H_DCI standard exhibits significantly higher I/O power consumption—80.24%, 83.38%, 62.92%, 76.52%, and 83.03% more compared to SSTL2_I, SST18_I, SSTL2_I_DCI, SSTL2_II, and SSTL15, respectively, when operating at 3.3 GHz.N.

### C. A prototype design for microprocessor based on Verilog HDL

Hinsu et al.,[3] In modern embedded system development, microprocessors are widely used, but they often function as black boxes—where inputs are provided, and corresponding outputs are received—without visibility into their internal hardware. If developers have insight into the internal components of the processor they are working with, it can help simplify and optimize their designs. This paper presents the design and simulation results of three fundamental microprocessor modules: the Arithmetic Logic Unit (ALU), RAM, and Instruction Decoder. These modules are then integrated using structural modeling to form a basic four-bit processor. This prototype serves as a foundational model for developing more advanced processors.

### D. Design and Implementation of Synchronous Dual-Port Memory

A. Pandey et al., [4] In this article, we present the design and implementation of a dual-port Random Access Memory (RAM) module with a focus on versatility, simplicity, and efficiency. Serving as a fundamental building block in digital systems, the dual-port RAM module enables simultaneous access from multiple sources while upholding data integrity and consistency. This project utilizes Verilog Hardware Description Language (HDL) to design, implement, and verify three essential memory modules, which play a crucial role in building efficient digital systems, the memory block has been synthesized and implemented on Field-Programmable Gate Array (FPGA) platforms. Key features include simultaneous read and write operations, priority-based conflict resolution, and a port locking mechanism for single-port mode. Results demonstrate reliable operation under various scenarios, ensuring data integrity and efficiency in both single-port and dual-port modes. Our research highlights significant advantages for designers seeking dependable and effective high-speed memory subsystems, suitable for real-time signal processing and multi-channel data processing.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

**E. Design and Implementation of Power Optimized Dual Core and Single Core DLX Processor on FPGA**
S. Pandit et al., [5] The aim of the work is to design and implement a low power 32 bits RISC core on Xilinx Nexys 4 FPGA. We also look at the further reducing the dynamic power consumption by implementing a shared RAM dual core design and take advantage of parallelism offered by FPGAs to run two cores simultaneously. The design is based on 5-stage pipelined DLX architecture. The DLX architecture in a RISC core consists of Fetch, Decode, Execute, Memory access and Write-back cycle. The core is designed using Verilog HDL. The standard low power single core design consumes 0.098W at 100MHz frequency. The shared RAM dual core design consumes 0.101W at 100MHz clock frequency.

## METHODOLOGY

### A. Working Principle

This project focuses on designing and implementing three fundamental memory modules, ensuring accurate modeling and verification using hardware description languages and FPGA simulation. These modules serve as key building blocks for efficient digital systems: single-port RAM, dual-port RAM, and ROM. In single-port RAM, data can be read from or written to the memory through a single port, controlled by a clock signal, read/write enable signals, and an address bus. Only one operation—either read or write—can occur at a time, making it suitable for simple memory requirements. On the other hand, dual-port RAM allows simultaneous read and write operations through two independent ports, each with its own set of control signals and address buses. This capability enables high-performance applications requiring concurrent memory access, which improves efficiency and system throughput.
The ROM (Read-Only Memory) module is designed to store static data that cannot be modified during operation. Data stored in ROM is accessed quickly and efficiently, making it ideal for storing firmware or lookup tables in embedded systems. In this project, memory design and operations are coded in Verilog, synthesized, and simulated using Xilinx Vivado. Address decoding logic, memory read and write control, and data storage mechanisms form the core functional elements of the memory modules.

### B. Algorithm

This flowchart illustrates the iterative process of hardware design using High-Level Synthesis (HLS). It begins with developing and editing C/C++ code, which is then debugged and compiled to ensure correctness. The compiled code undergoes HLS to convert it into a hardware description language (e.g., Verilog or VHDL). After synthesis, the design is evaluated against timing and resource specifications. If the design fails to meet these requirements, adjustments are made either by modifying HLS directives or redesigning the code, and the process is repeated.
This flowchart outlines the step-by-step process for hardware design and verification using High-Level Synthesis (HLS). The process starts with creating and refining C/C++ code, which provides a high-level representation of the intended hardware functionality. This code is written and refined by the designer to ensure clarity and alignment with the intended hardware behavior. Once the code is prepared, it is subjected to debugging and compilation to identify and fix any errors, ensuring the code runs correctly and is ready for synthesis. The next step is High-Level Synthesis (HLS), where the C/C++ code is translated into a hardware description language (HDL) such as Verilog or VHDL. This step is crucial as it converts high-level functional descriptions into a low-level representation that can be implemented on hardware devices like FPGAs or ASICs. The synthesized hardware is then evaluated to determine whether it meets the required timing constraints (e.g., clock frequency) and resource utilization (e.g., logic gates, memory, or power). If the design does not meet the required specifications, the process returns to the optimization stage. The designer can adjust or apply new directives—such as loop unrolling, pipelining, or resource sharing—or modify the code to better align with performance objectives. This iterative loop continues until the design satisfies all timing and resource requirements. Once the design meets the specifications, it undergoes C/RTL co-simulation. This step verifies the functional correctness of the synthesized hardware by simulating its behavior alongside the original C/C++ code. The co-simulation ensures that the hardware implementation matches the intended behavior of the software model. If the co-simulation passes, the design is finalized, and the hardware description is exported for further implementation on the target platform. However, if the co-simulation fails, the process loops back to the code refinement stage, and the steps are repeated. This iterative process ensures that the final hardware design is functionally correct, optimized, and ready for deployment.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

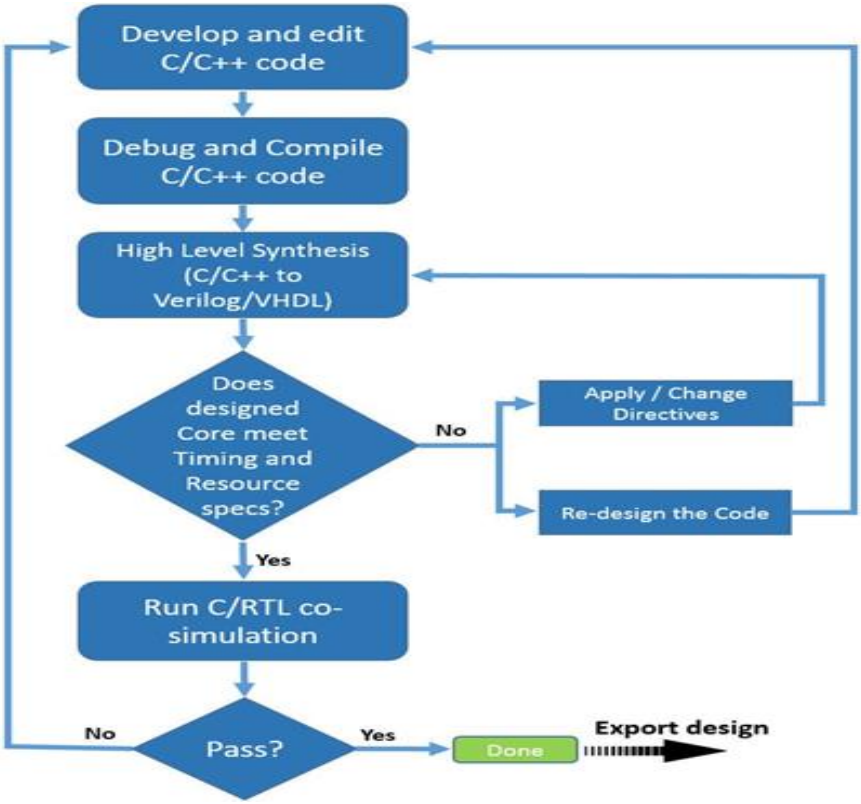


*Figure 1 Flow Diagram [6]*

**C. Implementation Details**

The implementation of this project involves the design, simulation, and testing of single-port RAM, dual-port RAM, and ROM using Verilog on the Xilinx Vivado FPGA design platform. The development process started with defining the architecture and behavioral logic for each memory module, followed by coding the modules in Verilog. Comprehensive testing and simulation were carried out to verify the functionality and efficiency of each design.

1. Single-Port RAM Implementation:

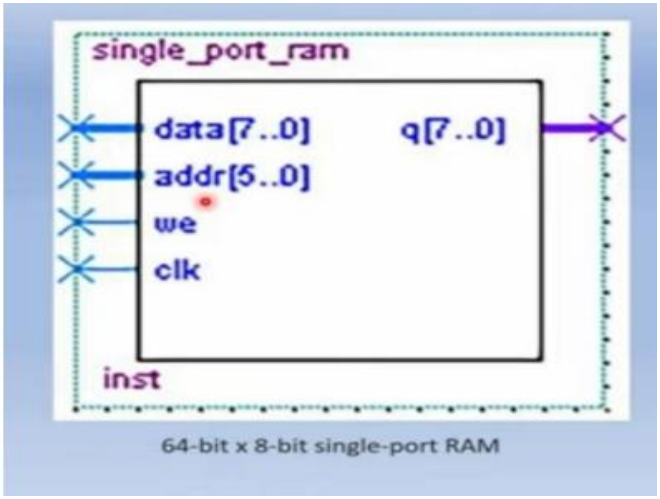


*Figure 2 Single-Port RAM [7]*

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

The single-port RAM Fig 2 module was designed to support basic read and write operations through a single access port. This port shares the same data bus for both reading and writing operations. The design included essential components such as a memory array, address bus, clock input, and read/write control signals. The operation is synchronized with the clock signal, where a read or write operation is triggered based on the control signal. Only one operation—either read or write—can occur per clock cycle. The memory module was tested by writing data to specific addresses and reading it back to ensure data integrity.

2. Dual-Port RAM Implementation

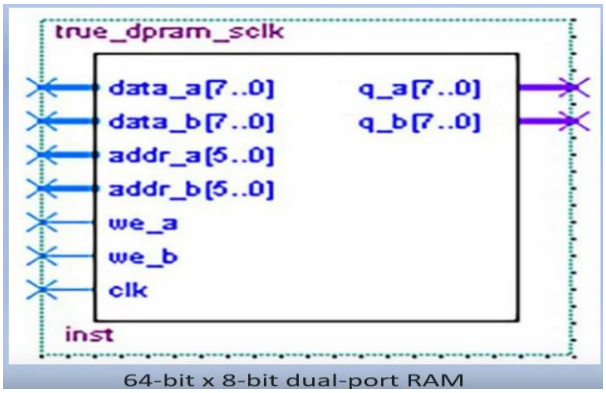


***Figure 3 Dual-Port RAM [8]***

The dual-port RAM Fig 3 design was more complex, as it allows simultaneous read and write operations through two independent access ports. Each port was assigned its own address bus, data bus, and control signals, enabling concurrent operations. Special logic was implemented to handle scenarios where both ports attempted to access the same memory location simultaneously. Address decoding and arbitration mechanisms were integrated to manage potential conflicts and ensure data consistency. This design significantly enhances system performance by enabling parallel data processing, making it suitable for high-speed computing and communication systems.

3. ROM Implementation

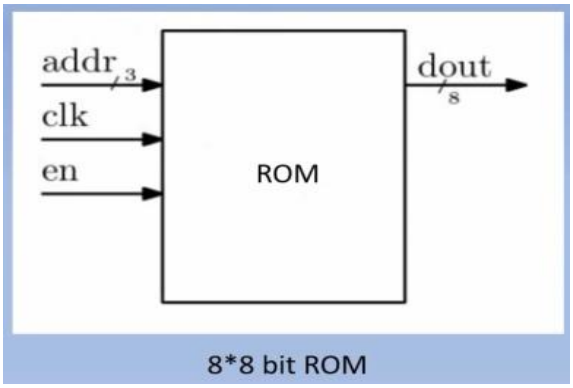


***Figure 4 ROM [9]***

The ROM (Read-Only Memory) Fig 4module was designed to store pre-defined, static data that remains unchanged during operation. The contents of the ROM were hard-coded into the Verilog design and stored during synthesis. This design is particularly useful for applications requiring fixed data storage, such as firmware or lookup tables. The ROM module only supports read operations, This ensures fast and efficient data retrieval without the complexity of write logic. Simulation and testing verified that the data stored in ROM aligned with the expected values. The designs were simulated in Xilinx Vivado to verify their functional correctness. Testbenches were developed to validate each memory module under various scenarios, including

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

sequential and random memory accesses. Waveform analysis was used to observe the read and write operations and ensure correct timing and control logic.

4. Performance Analysis

After successful simulation, the designs were synthesized and implemented on an FPGA to evaluate their real-world performance. Metrics such as timing, area (resource utilization), and power consumption were analyzed. The single-port RAM demonstrated simple and resource-efficient operation, while the dual-port RAM provided higher flexibility and better performance at the cost of increased resource usage. The ROM exhibited fast and low-power read operations, making it ideal for static data storage.

5. Challenges and Optimization

During implementation, challenges such as managing contention in dual-port RAM and optimizing resource utilization on the FPGA were encountered. These were addressed by careful design of control logic and efficient memory mapping. The project successfully demonstrated the functionality and efficiency of single-port RAM, dual-port RAM, and ROM, Offering valuable insights into memory design for embedded systems and hardware applications.

## RESULT AND ANALYSIS

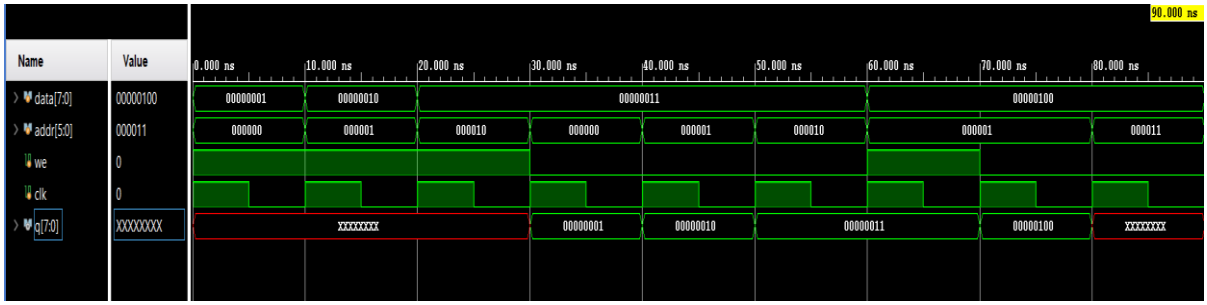


*Figure 5 Waveform of read & write of Single Port RAM*

The time scale, represented by vertical yellow markers and a horizontal time axis, indicates the progression of time in the simulation, with divisions corresponding to time intervals such as nanoseconds or microseconds. Each row in the waveform corresponds to a different signal in the circuit. For instance, the clk signal, likely the clock, toggles periodically between high (1) and low (0) to provide a timing reference. The data[3:0] signal represents a 4-bit bus displaying its binary values at each time step, while Q[3:0] is another 4-bit bus, possibly representing an output or a processed version of the data signal. Additionally, the reset (rst) signal, when high, resets the circuit's state and transitions low to allow normal operation. The behavior of the signals reflects their function in the circuit. The clk provides the timing rhythm, toggling regularly throughout the simulation. The reset (rst) is active (high) at the start, ensuring the circuit initializes properly, and becomes inactive (low) later to allow operations to proceed. The data[3:0] signal demonstrates a sequence of binary inputs (e.g., 0000, 0001, 0010), representing values applied to the circuit, while Q[3:0] seems to follow or process the data signal, potentially with a delay or through some logic, such as in a register or latch. This setup provides insights into how signals interact and evolve over time in the circuit.

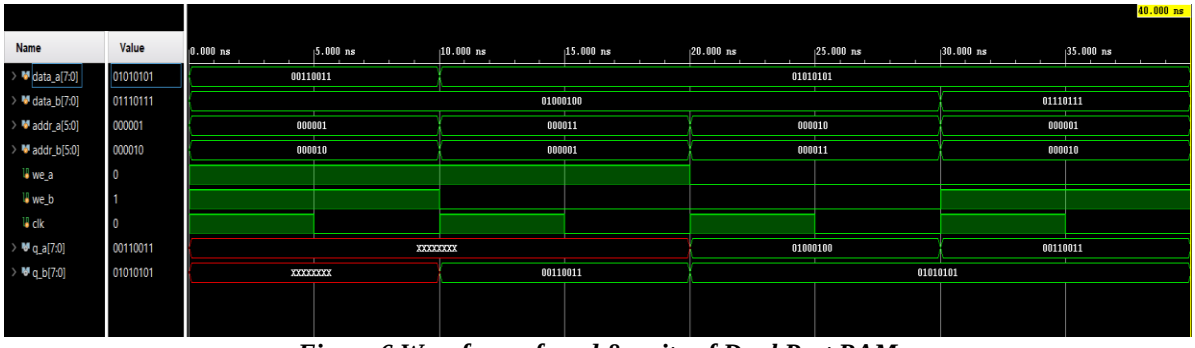


*Figure 6 Waveform of read & write of Dual Port RAM*

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

The time scale, represented on the horizontal axis, provides a clear reference for the progression of time in the simulation, measured in nanoseconds (ns). The scale spans from 0 ns to 40 ns, with major divisions marked at every 10 ns, visually emphasized by vertical yellow lines. These divisions enable precise identification of when specific events or transitions occur within the circuit, offering a structured view of time progression.
Each horizontal line in the waveform corresponds to a specific signal in the circuit, displaying its state (high or low) at various time intervals. The state of each signal dynamically changes in response to the operations or events taking place within the circuit. These transitions visually depict how signals interact and evolve over time, providing critical insights into the circuit's functionality and timing behavior. By correlating signal states with the time scale, designers can analyze and verify the behavior of the circuit under test, ensuring it operates as intended.

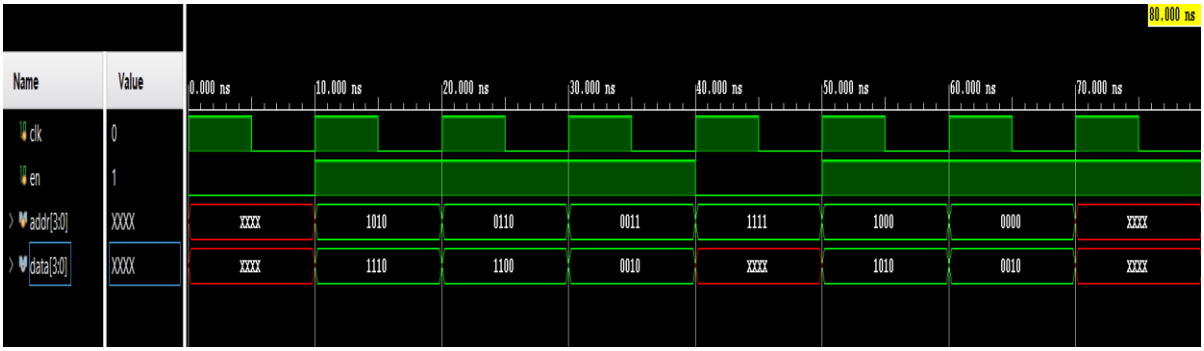


***Figure 7 Waveform of read of ROM***

The waveform includes several key signals, each playing a distinct role in the circuit's operation. The addr[7:0] signal represents an 8-bit address bus, which changes at specific clock edges to indicate a new memory or register address being accessed. This signal is crucial for identifying the location of data within the system. The clk signal, a periodic clock, serves as the synchronization backbone for all circuit operations. Transitions in data and address signals are typically aligned with the rising or falling edges of the clock, ensuring coordinated timing across the system.
The data[7:0] signal represents an 8-bit data bus that facilitates the transfer of data between components, such as memory or registers. This signal can be either read from or written to, depending on the state of control signals within the circuit. Lastly, the en signal, or enable signal, acts as a control mechanism, determining when certain operations are active. When this signal is high, it often indicates that a memory read or write operation is in progress. Together, these signals provide a comprehensive view of the circuit's address, data flow, and control mechanisms, enabling detailed analysis and verification of its functionality.

## CONCLUSION

In this project, we explored the design and implementation of single-port RAM, dual-port RAM, and ROM using Verilog and Xilinx Vivado. Each memory type has its unique advantages and applications depending on the system requirements. Single-port RAM offers a simple and resource-efficient solution for systems that need only one memory access at a time. Dual-port RAM, while more complex, provides high flexibility and performance for applications requiring simultaneous read and write operations. ROM, being read-only, is best suited for storing static data, offering a simple design with low power consumption and fast data retrieval.
The selection of a memory design depends on system requirements, including data access patterns, resource limitations, and power efficiency. Understanding the strengths and limitations of each memory type allows for better optimization of hardware designs and system performance. This project highlights the importance of carefully considering memory architecture in embedded systems, offering insights into how different memory types can be effectively utilized for varying requirements.
Through the design and simulation of these memory modules, the project demonstrated how the choice of memory type impacts system performance, resource usage, and power consumption. Dual-port RAM, while offering greater flexibility and performance, introduces added complexity and consumes more resources, which must be carefully managed. On the other hand, ROM's simplicity and efficiency in static data storage make it a

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

valuable option for many embedded systems but limits its applicability in dynamic environments where data modification is necessary.

Overall, this project emphasizes the importance of selecting the right memory design based on specific system requirements. Understanding the trade-offs between different memory types is essential for optimizing hardware designs, ensuring better performance, and making efficient use of resources. The insights gained from this work contribute to a deeper understanding of memory design in FPGA-based systems and provide a foundation for future work in embedded systems and hardware design.

## REFERENCES

[1] B. Pandey, A. Ratnakar, D. Joshi and A. K. Gupta, "Energy-Efficient Design of RAM and Its Implementation on FPGA," 2021 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT), Bhopal, India, 2021.
[2] M. Bansal, N. Bansal, R. Saini, B. Pandey, L. Kalra and D. M. A. Hussain, "SSTL I/O Standard based environment friendly energy efficient ROM design on FPGA," 3rd International Symposium on Environmental Friendly Energies and Applications (EFEA), Paris, France, 2014.
[3] N. Hinsu and D. Suryavanshi, "A prototype design for microprocessor based on Verilog HDL," International Conference for Convergence for Technology-2014, Pune, India, 2014.
[4] Pandey, B. Suhas A, B. P and S. Madhavan, "Design and Implementation of Synchronous Dual-Port Memory," 2024 Third International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE), Ballari, India, 2024.
[5] S. Pandit and P. Sikka, "Design and Implementation of Power Optimized Dual Core and Single Core DLX Processor on FPGA," 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Bengaluru, India, 2018.
[6] RAM and ROM Design www.researchgate.net.
[7] 64-bit x 8-bit Single-port RAM www.intel.com.
[8] 64-bit x 8-bit Dual-port RAM www.intel.com.
[9] Dual Port Rom digitalsystemdesign.in.