International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 11 Issue: 05 | May 2024 www.irjet.net p-ISSN: 2395-0072

DESIGN OF 16 BIT RISC PROCESSOR FOR ARITHMETIC AND LOGICAL OPERATIONS IN XILINX VIVADO

K.G.Venkata Krishna[1], N.K.KamalaDevi[2], K.Vishnu Varma[3], A.Yedukondalu[4]

[1] Assistant Professor, Department of Electronics and Communication Engineering, Krishna University College of Engineering and Technology Krishna University, Machilipatnam Andhra Pradesh, India.
[2]UG Student, Dept. of ECE, Krishna University College of Eng. &Tech, Machilipatnam, A.P, India.
[3]UG Student, Dept. of ECE, Krishna University College of Eng. &Tech, Machilipatnam, A.P, India.
[4]UG Student, Dept. of ECE, Krishna University College of Eng. &Tech, Machilipatnam, A.P, India.

Abstract - *The Re prefers a more condensed, straightforward set of instructions that all execute in the same amount of time. This processor preserves functional units without compromising performance. The design makes advantage of an architecture known as Harvardd, which includes separate data and instruction memory. A word of instructions has 24 bits in total. The CPU supports three addressing modes in addition to sixteen instructions. It contains sixteen general-purpose registers. Any register has the capacity to store 16 bits of data. The procedure performs 11 arithmetic and logical operations. Every module is developed and tested separately at every stage of implementation before being properly mapped into the top-level module. The simulation results are verified using Xilinx Vivado 2023.1 once the design input and synthesis are finished using the same tool.*

Key Words: **RISC, 16-bit, VLSI, verilog**

1.INTRODUCTION

When the performance of CISC fell short of expectations and the controller design grew more challenging, people started to consider alternate approaches. It has been found that when a CPU interfaces with memory, speed is lost. Reducing the complexity of the instruction set was the only option to raise CPI. Simpler in terms of design than in terms of functioning. As a result, the CPU is not required to access memory for very many instructions in a typical RISC architecture—probably only load and store. Ultimately, pipelining increased performance. Only a few additional registers can provide a new level of performance by lowering CPU and increasing throughput. Consequently, the instruction may be successfully executed in one clock cycle. It's a common misperception that when the term "Reduced Instruction Set Computer" is used, instructions are only removed to create a smaller set of instructions. In fact, RISC instruction sets have grown in size over time, and several of them now include more instructions than many CISC CPUs.

2. LITERATURE REVIEW

2.1 S. Lad and V. S. Bendre, "Design and Comparison of Multiplier using Vedic Sutras," 2019 5th International Conference On Computing, Communication, Control And Automation (ICCUBEA), Pune, India, 2019, pp. 1-5

Fast processing units are necessary for many real-time applications in modern computerized era. The basic building elements of these units are ALU and MAC, which are necessary for quick and effective execution. Digital signal processors primarily use multipliers as their primary component. ALU and MAC performance can be improved by modifying registers, multiplier, and adder to retain correctness and speed up execution. Due to the growing delays restrictions, the design of quicker multipliers is prioritized for implementation in processors. It is crucial to create quicker multipliers in order to increase multiplication speed.

2.2 Balpande Vishwas V, Abhishek B. Pande, Meeta J. Walke, Bhavna D. Choudhari and Kiran R. Bagade. “Design and Implementation of 16 Bit Processor on FPGA.” (2015).

This project involves the design of a 16-bit RISC processor and the Verilog HDL modeling of its constituent parts. Harvard architecture is the basis of the processor. This instruction set's extreme simplicity provides an indication of the sort of hardware that should be able to correctly execute the set of instructions. More sophisticated blocks like an ALU and memory have been built and simulated in addition to the sequential and combinational processor building blocks like adders and registers. In this project, comprehensive structural ALU modeling, beginning with half adders, has been completed. Ultimately, the semi-custom layout was created just for ALU.

2.3 Seung Pyo Jung, Jingzhe Xu, Donghoon Lee, Ju Sung Park, Kang-joo Kim and Koon- shik Cho, "Design & verification of 16 bit RISC processor,"

2008 International SoC Design Conference, Busan, 2008, pp. III-13-III-14

This article presents the design and verification process for a 16-bit RISC processor. The suggested processor features a Harvard design, with internal debug logic, a 5-stage pipeline for instruction execution, and a 24-bit address. The FPGA-based processor successfully executes the SOLA algorithm and the ADPCM vocoder. Personal digital assistants (PDAs) and portable multimedia players (PMPs) are not unique human inventions. Thus, SOC level ASICs (Application Specific Integrated Circuits) are used to create compact and low power processors. The 8051 and ARM 7 processors are the most widely used SOC level ASIC processors.

2.4 Chandni N. Naik, Vaishnavi M. Velvani, Pooja J. Patel, Khushbu G. Parekh, "VLSI Based 16 Bit ALU with Interfacing Circuit", International Journal of Innovative and Emerging Research in Engineering Volume 2, Issue 3, 2015.

This project uses VHDL to construct a 16-bit ALU that is interfaced with ROM and RAM. One of the most crucial CPU modules, the ALU allows for modifications to be made during the majority of instruction executions. Therefore, further ALU operation is a crucial duty. After that, Xilinx is used to implement this design. After creating an ALU, interfaced RAM and ROM with it. Waveforms are displayed for each result in the Xilinx program. The CPU is sped up by this project.

2.5 Pushpalata Verma, K. K. Mehta, "Implementation of an Efficient Multiplier based on Vedic Mathematics Using EDA Tool", International Journal of Engineering and Advance Technology, Vol.1, no. 5, June, 2012

An optimized area efficient multiplier is built in this project. Due to the increasing growth of integration, many significant signal processing systems are designed on VLSI platforms. Systems and applications for signal processing need a lot of processing power, which means they consume a lot of energy. Performance and area are two crucial design factors for VLSI systems. In general, the multiplier element's performance determines the system's performance.

3. PROPOSED SYSTEM

The processor's job is to efficiently carry out each and every instruction provided in accordance with machine language. Arithmetic and Logical Unit, or ALU, is a combinational circuit. This unit is made to execute different integers with different sets of instructions. Operation code (opcode) plus a few operands make up the instruction (machine word) that an ALU receives from a processor. Thus, the operands are employed in the operation after the opcode instructs the ALU on which and what operation to carry out. A little collection of data storage facilities is referred to as the Register Bank. The ALU verifies the bits and signals whether the operation was successful by storing the result of the operation in an accumulator, which is then stored in a storage register. If the operation is unsuccessful, a status message also referred to as a status register or Z-Flag will be shown. Its job is to run programs and provide effective operation for the data kept in memory. A set of instructions is all that a processor needs to carry out a task in a computer. The command to be carried out is stored in the control unit. The address register, data register, and instruction register are among the registers found in a CPU. The CPU's function is to retrieve, decode, and carry out memory operations in accordance with the registers. Decoding the op-code, identifying the instruction, figuring out which operands are in memory, getting the operands from memory, and giving a processor an order to carry out the instruction are all part of the IR task. This is accomplished with the aid of a control unit, which produces the timing signals needed to regulate the several processing components involved in carrying out the command.

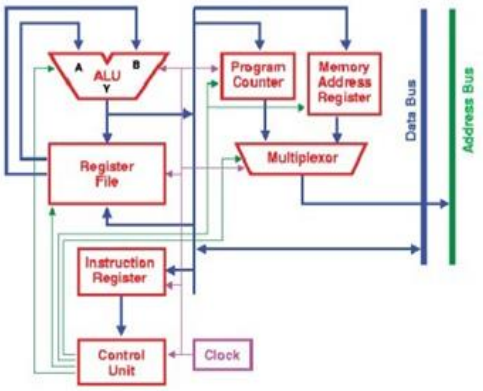

Fig -1: Proposed Architecture

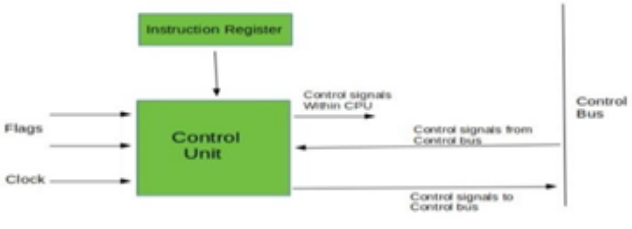

Fig -2: Control unit

The portion of the central processing unit (CPU) of a computer that controls how the processor operates is called the control unit. It was a component of John von Neumann's Von Neumann Architecture. The control unit is in charge of instructing the computer's memory, input and output devices, and arithmetic/logic unit on how to react to commands given to the processor.

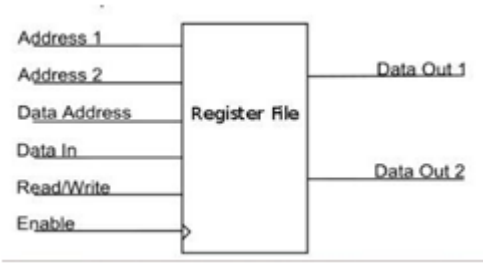

Fig -3: Block diagram of Register file

A decoder and a collection of registers make up a basic register file. A data input and an address are needed for the register function. But in a modern processor architecture, this straightforward register file is useless since there are times when we don't want to write a new value to a register. Additionally, in a single cycle, we usually wish to post back one value and read two values simultaneously.

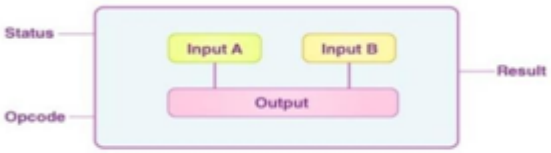

Fig -4: Arithmetic and logic Signals

3. VERILOG HDL AND XILINX VIVADO

3.1 FLOW OF VLSI DESIGN

The formal specification of a VLSI chip is the first step in the VLSI design cycle, which proceeds through several stages to generate a packaged chip.

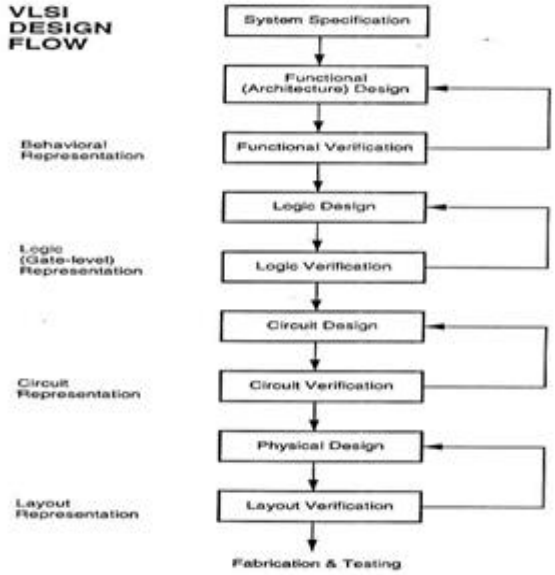

Fig -5: The flow of VLSI design

4. RESULTS

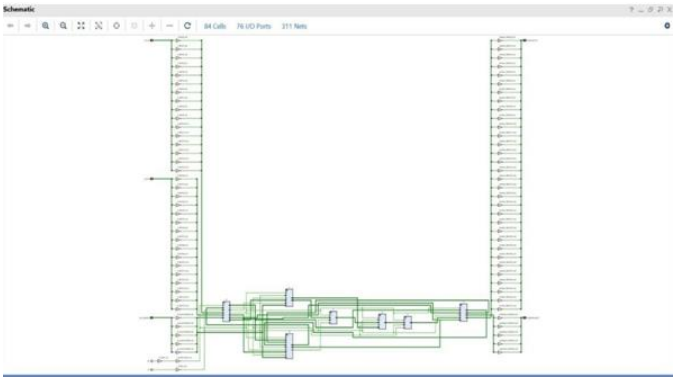

Fig -6: Schematic View

The output of a Vedic multiplier utilizing the UrdhvaTrigbhyam sutra is seen in the following 16-Bit Multiplier simulation result. The MAC procedures are carried out using two inputs, a=252 and b=846, to produce an output value of c=213192.
Compared to other multipliers, the latency and size of the gate rise relatively slowly as the number of bits increases.
As a result, the CPU's speed, power, and timing are all efficient.

Fig -6: System Utilization

Transcript window results for ALU are displayed in the following figure, which includes all matched conditions. The processor module's integrated multiplier operation is executed. The data from the transcript is compared and the multiplication process is confirmed.

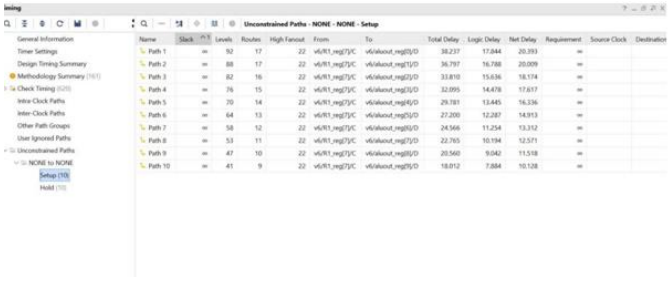

Fig -7: System Delay