

FPGA Implementation of MIPS RISC Processor

S. Suresh¹ and R. Ganesh²

¹CVR College of Engineering/PG Student, Hyderabad, India

²CVR College of Engineering/ECE Department, Hyderabad, India

Abstract

The arithmetic and logical computational operations in applications like mobile computing, automobiles, industrial process controls etc can be done by using processing element called as processor. Two commonly used processor architectures are CISC (Complex Instruction Set Computer) and RISC (Reduced Instruction Set Computer) processor. The RISC processor uses large number of simple instructions and performs complex operations by using pipeline concept with improved performance.

This paper describes an implementation of Microprocessor without Interlocked Pipeline Stages (MIPS) based RISC processor for 10-bit address bus and an 8-bit bi-directional data bus. This processor is designed by using Verilog HDL, simulated, synthesized and implemented by using Xilinx 12.4 tool, targeted for Xilinx Virtex-5 XUPV5LX110T FPGA board.

Index Terms—RISC, MIPS, Pipeline, FPGA.

I. INTRODUCTION

The computational operations like arithmetic, logical etc on data elements can be done by using a processing element called as processor. The processors are used in different applications like home appliances, automobiles, cellular phones and industrial process control etc.

The selection of processor architecture for particular applications can be done by using the type of architecture, constraints on data size, address size and memory size etc. Two commonly used processor architectures are CISC (Complex Instruction Set Computer) and RISC (Reduced Instruction Set Computer) processor. The CISC processors use less number of instructions that performs complex operations by compromising in speed of operation. The RISC processors use large number of simple instructions that performs complex operations by improving speed of operation compared to CISC processors.

The basic architecture of a processor is shown in Figure 1. In RISC processors the instructions can be executed with the help of stages by using clock cycles. The instruction can be divided into four stages i.e. fetch, decode, execute and write back. The general processor requires one clock cycle for one stage and operates on less freq, more clock cycles to execute an instruction. The processor execution time can be reduced by using pipelining concept.

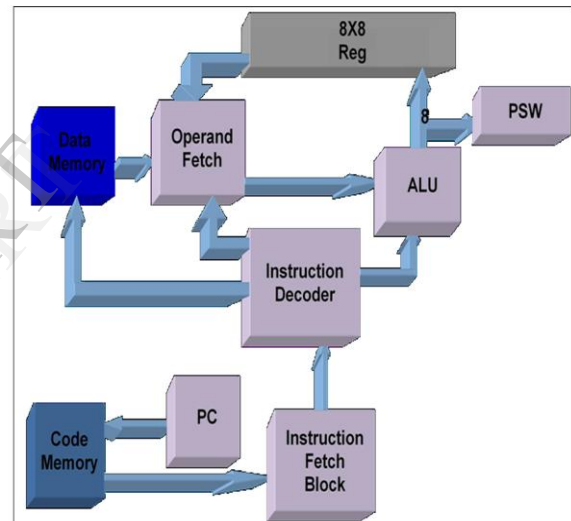


Figure 1: Basic architecture of a processor

The pipelining concept can be implemented by using number of pipelining stages, instruction unit etc. The Microprocessor without Interlocked Pipeline stages (MIPS) is a pipelining concept that can be used in a processor. The pipelined processor has four stages i.e. Instruction fetch, decode, execute, write back. The block diagram for these four stages of processor is shown in Figure 2.

The RISC processor with the concepts of MIPS is designed and implemented for 10-bit address bus and an 8-bit bi-directional data bus. The MIPS (Microprocessor without Interlocked Pipeline stages) based RISC processor is designed for both single-cycle and pipelined concepts.

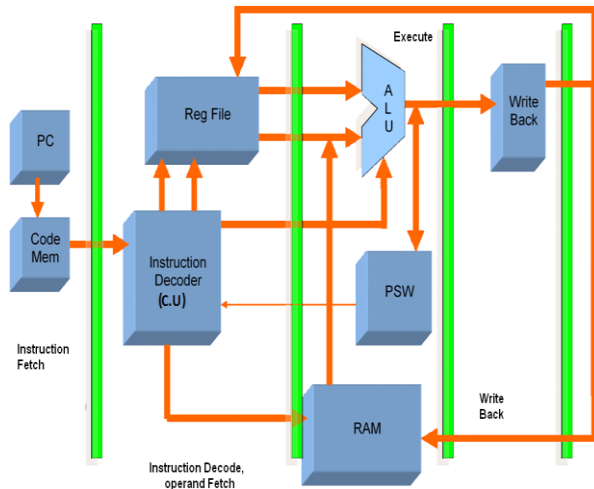


Figure 2: Block diagram for four different stages of the processor

This paper explains the FPGA implementation of MIPS based RISC processor. This paper is organized such that the chapter II gives an overview of CISC, RISC and concept of MIPS for the processor design. The chapter III describes the simulation and synthesis results of single cycle and pipelined processor design by using Xilinx ISE tools, and chapter IV describes the FPGA implementation of MIPS processor by using Xilinx virtex5 FPGA. The conclusion is presented in chapter V followed by references.

II. OVERVIEW OF MIPS PROCESSOR DESIGN

The processors can be designed by using the number of instructions, data bits, address bits, addressing modes, single cycle or multi cycles etc. The RISC processor is designed for both single cycle and multi cycles by using the concept of MIPS. The block diagram of 8-bit single cycle processor is shown in Figure 3.

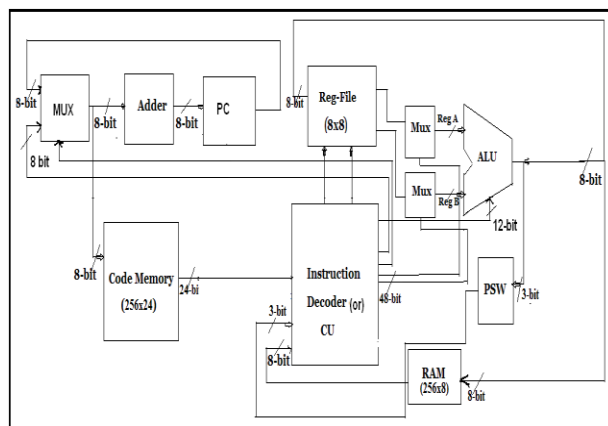


Figure 3: Single cycle Processor

The single cycle processor performs the tasks of instruction fetch, instruction decode, execution, memory access and write-back all in one clock cycle. First the PC value is used as an address to index the instruction memory which supplies a 24-bit value of the next instruction to be executed.

This instruction is next divided into opcode fields, register address fields, etc to address the register file. The register file supports two independent register reads and one register write in one clock cycle. The register file reads the requested address and outputs the data values contained in that registers. The data values are applied to the Arithmetic and Logic Unit (ALU) to perform the arithmetic and logical operations like addition, subtraction, logical AND etc. The result of ALU can be used to store in register files or to address the data memory.

The single cycle processor performance can be improved by using a concept called as Pipeline. Pipeline is a technique where by multiple instructions are overlapped in execution. In a computer pipeline, each step in the pipeline completes a part of an instruction. If the stages are perfectly balanced, then the time per instruction on the pipelined processor assuming ideal conditions is equal to

$$\frac{\text{Time per instruction on unpipelined machine}}{\text{Number of pipe stages}}$$

Number of pipe stages

The block diagram of 8-bit MIPS RISC processor is shown in Figure 4.

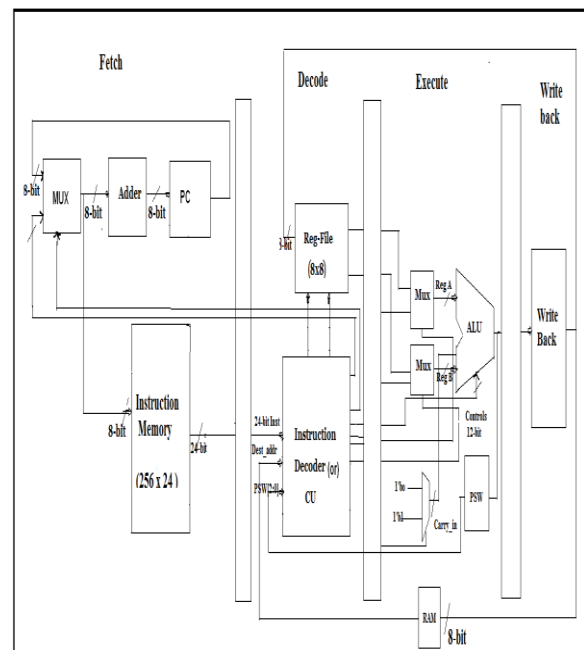


Figure 4: Block diagram of four stage pipelined 8-bit RISC Processor

The FPGA implementation of 8-BIT MIPS RISC processor can be designed by using the four stage pipelined concept with the individual blocks as explained below.

A. Stage-1: Fetch:

The fetch unit consists of Program Counter (PC), Adder, 8-bit 2x1 Multiplexer and the Instruction Memory. The size of the instruction memory or program memory is 256x24. In this stage the instruction is fetched from the memory and stores it into the Instruction Register and increments the Program Counter.

B. Stage-2: Decode:

Decode unit consists of mainly instruction decoder or Control Unit (CU) and Register File. The register file consists of 8-general purpose registers; each one is 8-bit length.

C. Stage-3: Execution:

The Execution unit has ALU as main element and finds the desired result. It also computes branch instruction for the targeted address and provides the same for Memory Write back Block. The multiplexer chooses which operand to be fetched. Execution unit consists of mainly Arithmetic and Logic Unit (ALU), Data Memory and Program Status Word (PSW).

D. Stage-4:

The Memory Write back Block consists of the ALU register and a multiplexer with source signal. The result is fed to either to memory or to the register multiplexer. The result of which is again fed to Data Memory, this output is given to Instruction Decoder.

The MIPS single-cycle processor performs the tasks of instruction fetch, instruction decode, execution and write-back all in one clock cycle.

The MIPS processor is designed by using Verilog Hardware Description Language. The design is synthesized for generating the net list from the given specifications for the targeted FPGA board. The output of the synthesis i.e. RTL schematic view of Single-Cycle MIPS is shown in Figure 5.

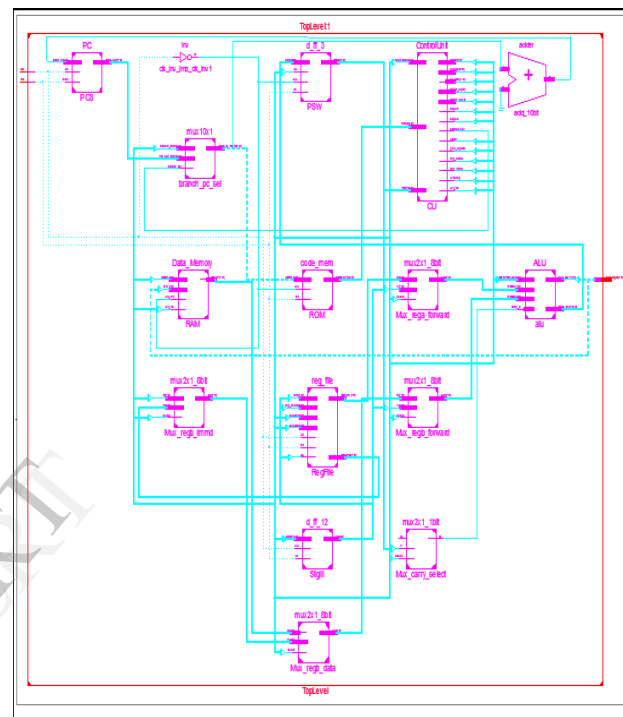


Figure 5: RTL Schematic of the Single-Cycle MIPS Processor

III. SIMULATION AND SYNTHESIS RESULTS OF MIPS PROCESSOR

The design of MIPS Processor is implemented on Xilinx ISE (Integrated Software Environment) tools with the sub tools listed in Table 1.

Table1: List of Tools under Xilinx ISE design suite

Design Action	Name of the Tool
Design Entry	HDL Editor (Verilog)
Synthesis	Xilinx Synthesis Tool
Simulator	ISim Simulator
Implementation	FPGA Editor, Plan Ahead
Device Configuration	iMPACT

The Figure 6 shows the simulation result of the MIPS Single-Cycle processor for addition and subtraction and multiplication.

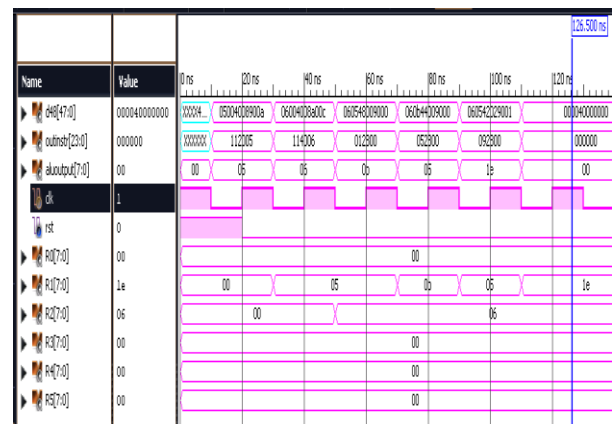


Figure 6: Simulation result of MIPS single cycle processor

The MIPS processor can be implemented by using a pipeline method in which more than one instruction can be executed simultaneously.

The pipelining is a technique used in the MIPS implementation to increase the number of instructions throughput the process and it does not reduce the time to execute or complete the instruction but we can execute a number of instructions at once. There are four stages to execute any instruction in the MIPS implementation process. The output of the synthesis i.e. RTL schematic view of Four Stage Pipeline MIPS is shown in Figure 7.

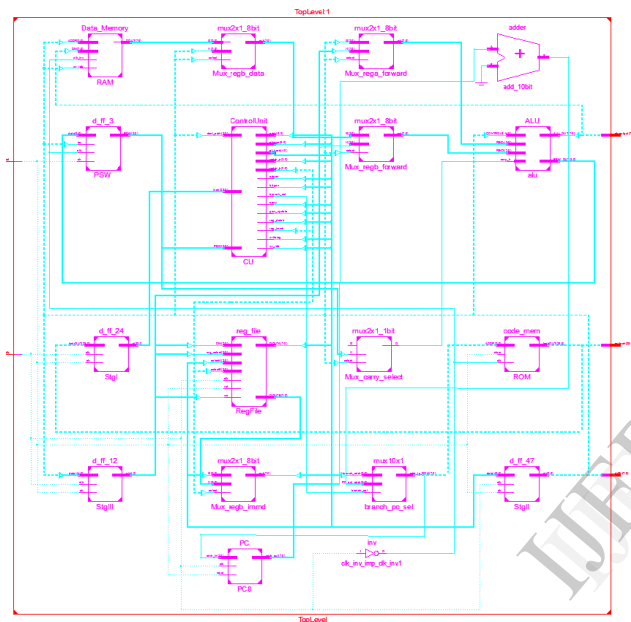


Figure 7: RTL Schematic of the Four Stage Pipeline MIPS Processor

The Figure 8 shows the simulation result of the Four Stage Pipeline MIPS processor for addition, subtraction, multiplication programs.

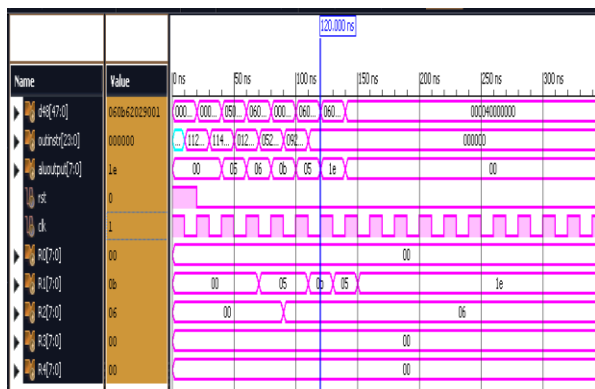


Figure 8: Simulation result of the Four Stage Pipeline MIPS processor

The Instruction Format used in the MIPS processor is a 24-bit length and it is shown Fig.9. The most significant 23-16 bits are used for opcode, the bits 15-13 are used for address generation for register A, the bits 12-10 are used for register B address generation and remaining 10 bits are used for the address, in these 10 bits 2-bits are unused and can be used for future purpose.

8-bit Opcode	Reg A Address (3-bit)	Reg B Address (3-bit)	10-bit Address (2-Bits are unused)
--------------	-----------------------	-----------------------	------------------------------------

Figure 9: Instruction Format

IV. FPGA IMPLEMENTATION OF MIPS PROCESSOR

The MIPS Processor functionality is verified by using simulation and FPGA prototyping. The FPGA prototyping can be done after the completion of a valid functional simulation and post route simulation. The design of MIPS processor is targeted for Xilinx Virtex 5 XC5VLX110T for FPGA implementation. The FPGA implementation MIPS processor using Virtex 5 FPGA board is shown in Figure 10. The arithmetic operations like addition, subtraction etc is observed using FPGA prototyping results.

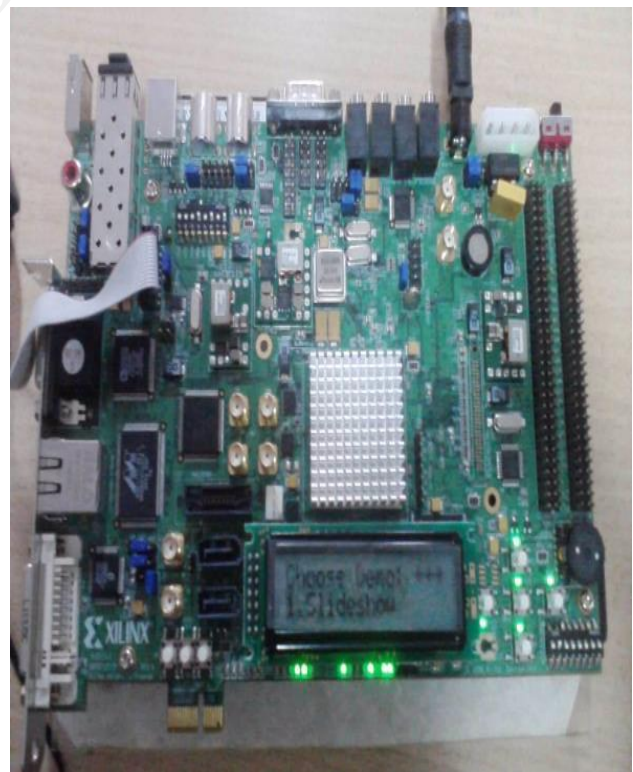


Figure 10: FPGA Prototyping result of MIPS Processor

V. CONCLUSIONS

The design of 4-stage Pipelined MIPS processor performing arithmetic, logical and data movement operation and branch instructions is designed by using Verilog HDL. The design is implemented for both single cycle and pipelining concepts using Xilinx Virtex-5 FPGA board and Xilinx ISE tools.

The single cycle MIPS processor has 410 LUTs, 332 slice register and 1BRAM. Timing analysis results show that the critical path is 38.447ns, i.e. the maximum clock frequency is 26.010MHz. The 4-state pipelined MIPS processor is utilizing 419 LUTs, 397 slice register and 1BRAM. Timing analysis results show that the critical path is 9.985 ns, i.e. the maximum clock frequency is 100.145MHz.

REFERENCES

1. Rabaey, J. M., Chandrakasan, A., & Borivoje, N. (2003). *Digital Integrated Circuits: A Design Perspective* 2nd Edition. Upper Saddle River, NJ: Pearson Education, Inc.
2. Hennessy, J. L., & Patterson, D. A. (2003). *Computer Organization and Design: The hardware/software interface* 2nd Edition. San Francisco, CA: Morgan Kaufmann Publishers, Inc.
3. D. J. Smith. (2010), „HDL Chip Design“, International Edition, Doone Publications.
4. A. S. Tanenbaum. 2000, „Structured Computer Organization“, 4th Edition, Prentice- Hall.
5. Luker, Jarrod D., Prasad, Vinod B.2001,“RISC system design in an FPGA”, MWSCAS 2001, v2, p532536.
6. Vincent P. Heuring, and Harry F. Jordan. 2003, “Computer Systems Design and Architecture”, 2nd Edition.
7. W. Stallings. 2003, *Computer Organization & Architecture. Designing for Performance*”. Prentice Hall, 6th edition.
8. Wayne Wolf. 2005, “FPGA Based System Design”, Prentice Hall.
9. Samir Palnitkar.1996,“Verilog HDL A guide to Digital Design and Synthesis”, SunSoft Pre.
10. Anderson, D.W.,F.J. Sparacio, and R. M. Tomasulo[1967]. ”The IBM 360 Model 91 Processor philosophy and instruction handling, “IBM J.Research & development 11:1 (january),8-24.
11. Bell , C. G., J. C. Mudge, and J. E. McNamara[1978]. *A DEC View of computer Engineering*, Digital Press, Bedford.
12. Mass.. Bucholtz, W. [1962]. *Planning a Computer System: Project Stretch*, McGraw-Hill, New York.
13. Darley, H. M., et al. [1989]. “Floating point/integer processor with divide and square root functions”, U.S. Patent 4,878,190 (October 31).
14. Four Stage Pipelined RISC Processor using Spartan 3A by Aboobacker Sidheeq.V.M, june 2012.