

Implementation of Low Power Pipelined 64-bit RISC Processor with Unbiased FPU on CPLD

*J. Vijay Kumar, *B. Naga Raju, **M. Vasu Babu' ***T. Ramanjappa

*Dept of Physics, SKUCET, Sri Krishnadevaraya University, Anantapur, India

**Dept of Applied Sciences, St. Ann's College of Engineering Technology, Chirala, India

***Dept of Physics, Sri Krishnadevaraya University, Anantapur, India

Article Info

Article history:

Received Jan 2, 2016

Revised Mar 23, 2016

Accepted Apr 11, 2016

Keyword:

Altera Max V

Low power

Modelsim

RISC

ABSTRACT

This article represents the implementation of low power pipelined 64-bit RISC processor on Altera MAXV CPLD device. The design is verified for arithmetic operations of both fixed and floating point numbers, branch and logical function of RISC processor. For all the jump instruction, the processor architecture will automatically flush the data in the pipeline, so as to avoid any misbehavior. This processor contains FPU unit, which supports double precision IEEE-754 format operations very accurately. The simulation results have been verified by using ModelSim software. The ALU operations and double precision floating point arithmetic operation results are displayed on 7-Segments. The necessary code is written in Verilog HDL.

Copyright © 2016 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

J. Vijay Kumar,

Departement of Physics,

Sri Krishnadevaraya University, College of Engineering and Technology.

Anantapur-515003.

Email: jvkphysics@gmail.com

1. INTRODUCTION

Today RISC CPUs (and microcontrollers) represent the vast majority of all CPUs in use. The RISC design technique offers power in even small sizes and thus has come to completely dominate the market for low-power "embedded" CPUs. Embedded CPUs are by far the largest market for processors. RISC had also completely taken over the market for larger workstations [1]. Over many years, RISC instructions sets have tended to grow in size. Thus, some have started using the term "load/store" to describe RISC processors, since this is the key element of all such designs. Instead of the CPU itself handling many addressing modes, load/store architecture uses a separate unit dedicated to handling very simple forms of load and store operations [2].

The floating point operations have found intensive applications in the various fields for the requirements for high precious operation due to its great dynamic range, high precision, and easy operation rules. High attention has been paid on the design and research of the floating point processing units. With the increasing requirements for the floating point operations for the high-speed data signal processing and the scientific operation, the requirements for the high-speed hardware floating point arithmetic units have become more and more exigent. The implementation of the floating point arithmetic has been very easy and convenient in the floating point high-level languages, but the implementation of the arithmetic by hardware has been very difficult [3]. With the development of the very large scale integration (VLSI) technology, a kind of devices like CPLD and FPGAs have become the best options for implementing floating hardware arithmetic units because of their high integration density, low price, high performance and flexible applications requirements for high precious operation [4].

Low power has emerged as a major principle theme in today's electronics industry. The need for low power has caused a major paradigm shift where power dissipation has become an important consideration as performance and area. Low-power embedded processors are used in a wide variety of applications including cars, phones, digital cameras, printers, and other such devices. There are lots of techniques like Clock Gating, Supply Voltage Reduction, Multi-Vdd, Dynamic Voltage Frequency Scaling etc to reduce the power.

In the present work, CPLD based 64-bit RISC processor with a high-speed floating point double precision is designed using pipelined architecture. This can improve the speed of the operation as well as overall performance [5]. The processor contains to implement 4-stage pipelining including double precision floating point unit. The 4 stages are Fetch, Decode, Execute, Memory Read / Write Back. In this design, all the arithmetic, branch, logical and floating point operations (add, sub, mul and div) are performed and the resultant value is stored in the memory/register and retrieved back from memory when required.

This is a general purpose 64-bit RISC processor with pipelining architecture which gets instructions on a regular basis using dedicated buses to its memory executes all its native instructions in stages with pipelining. It will have short (8-bit) and long (16-bit) instructions. For all Arithmetic and logical operations 8-bit instructions are used and for all memory transactions and jump instructions 16-bit instructions are used and also have special instructions to access external ports. For all the jump instruction, the processor architecture will automatically flush the data in the pipeline, so as to avoid any misbehavior [6].

2. ARCHITECTURE OF THE DESIGN

The architecture of low power pipelined 64-bit RISC processor with Floating Point Unit is a single cycle pipelined processor as shown in Figure 1. This section presents the design of different modules like instruction fetch, instruction decode, register file, execution unit, floating point unit, memory read/write back, instruction set and low power unit along with four general purpose registers namely Register0, Register1, Register2, and Register3 [7].

2.1. Instruction Fetch

This stage consists of the program counter and branch prediction. It means the instruction present in the memory is fetched from the Program Counter (PC) and stored in the instruction register. The branch prediction part to be the most likely is then fetched and speculatively executed. This will increase flow in instruction pipeline and achieve high effective performance.

2.2. Instruction Decoder

This stage consists of the control unit, register file. The opcode fetched from the memory is being decoded for the next steps and moved to appropriate registers. This is a two-port register file which can perform two simultaneous read and one write operation. It contains four 64-bit general-purpose registers. When the Reg_Write signal is high, a write operation is performed to the register.

2.3. Instruction Execution

This stage consists of the arithmetic logic unit (ALU) and the ALU control unit. It performs the arithmetic & logical operations and also jump or branch instructions. The control unit is responsible for providing signals to the ALU that indicates the operation that the ALU will perform.

This unit also provides double precision floating point operations like addition, subtraction, multiplication, and division are performed.

2.4. Memory Unit

It means the result of the instruction execution (register-register or load instruction) is stored into the register file. The load and store instructions are used to access this module.

2.5. Low Power Unit

The input to low power unit is a global clock and gated clock is its output [8]. The input to low power unit is a global clock and its output is gated clock since the module will block the main clock in the following conditions. (i) When an instruction is a halt. (ii) When there is a continuous Nop operation. (iii) When program counter fails to increment.

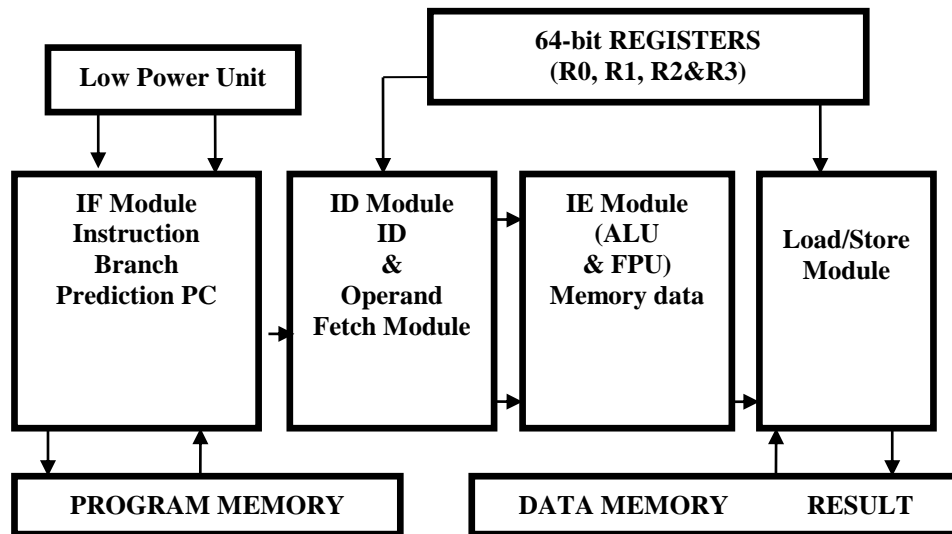


Figure 1. Architecture of the proposed design

3. HARDWARE AND SOFTWARE DETAILS

CPLDs are integrated circuits (ICs) or chips that application designers configure to implement digital hardware such as mobile phones. CPLDs are another way to extend the density of the simple PLDs. The concept is to have a few functional blocks or PLD blocks or macro cells on a single device with general purpose interconnect in between. The building block of a CPLD is the macrocell, which contains logic implementing disjunctive normal form expressions and more specialized logic operations. CPLD's predictable timing characteristics make them ideal for critical, high-performance control applications [9]. Typically, CPLDs have a shorter and more predictable delay than FPGAs and other programmable logic devices. Because they are inexpensive and require relatively small amounts of power, CPLDs are often used in cost-effective, battery-operated portable applications. The CPLD device used in the present work is MAX V (**5M2210Z**) manufactured by Altera.

Altera MAX V CPLDs deliver the industry's best value in low cost, low power CPLDs, offering robust new features at up to 50% lower total power when compared to competitive CPLDs. Altera's MAX V also features a unique, non-volatile architecture and one of the industry's largest density CPLDs. In addition, the MAX V integrates many functions that are previously external, such as flash, RAM, oscillators, and phase-locked loops. In many cases, it delivers more I/Os and logic per footprint at the same price as competitive CPLDs. The MAX V utilizes green packaging technology, with packages as small as 20mm. MAX V CPLDs are supported by Quartus II software v.10.1, which allows productivity enhancements resulting in faster simulation, faster board bring-up, and faster timing closure [10].

4. RESULTS AND DISCUSSION

The design is implemented on Altera MAX V CPLD on which arithmetic, branch operations and logical functions are verified. Pipelining would not flush when branch instruction occurs as it is implemented using dynamic branch prediction. Branch predictions will increase flow in instruction pipeline and achieve high effective performance. When the processor is idle, CLOCK is switched off through sleep mode by using low power technique. This design can be used for low power applications to enhance the battery life of the devices. This 64-bit RISC processor consumes only 1 instruction, whereas 32-bit RISC processor needs more than 1 instruction. This processor with floating point operations is used in many applications like signal processing, graphics and medical equipments.

Figure 2 shows the simulation results of low power unit. Figure 3 shows the simulation results of 64-bit RISC processor with FPU. Figure 4 shows the RTL schematic view of the processor which describes how the logic resources are organized inside the top level schematic view.

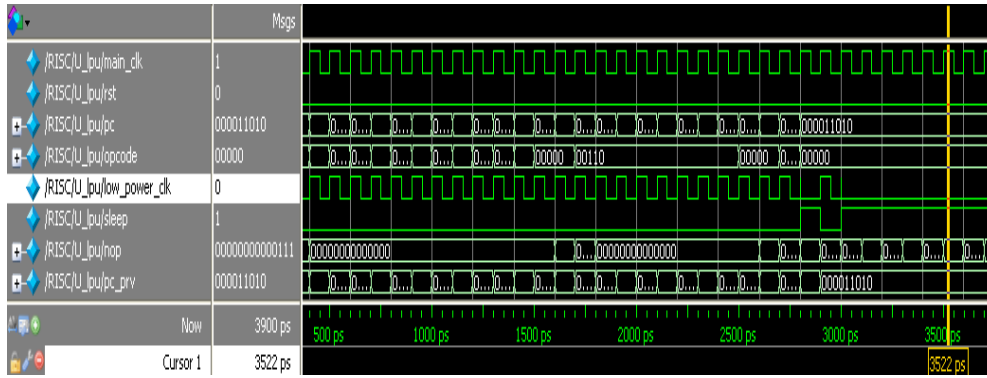


Figure 2. Simulation result of Low Power Unit

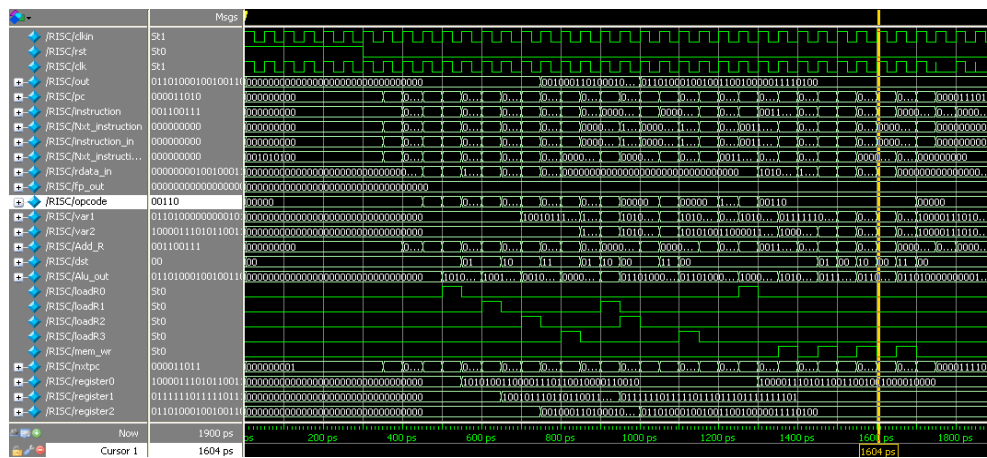


Figure 3. Simulation result of 64-bit RISC processor with FPU

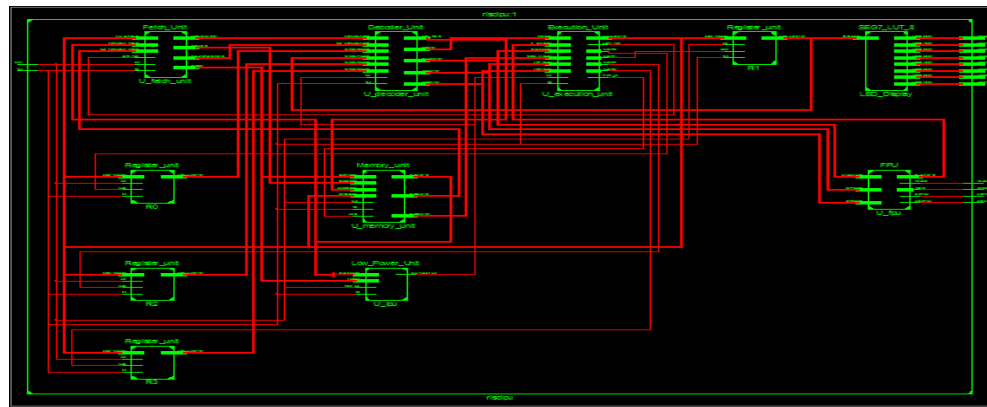


Figure 4. RTL Schematic view of proposed processor

Table 1 shows the results of the RISC processor which performs ALU operations for binary and hexadecimal values

Input var1 (Src) =64'b1010_0111_0110_1000_1111_0101_0011_1110_1011_1001_1101_0001_0010_0000_0110_0000

Input var2 (Dst) =64'b1111_0111_1010_0011_1100_1110_1101_0101_1011_1000_0010_0011_0000_0001_1101_0100

REFERENCES

- [1] D. Patterson and D. Ditzel, "The case for the reduced instruction set computer", ACM SIGARCH Computer Architecture News, 1980.
- [2] Marie Elma Domingo, "A Presentation of the RISC and CISC Instruction Set Architectures", University of the Philippines Diliman.
- [3] http://www2.Hsiao-Fen Fu_Tutorial_IEEE754floatingpoint.pdf.
- [4] Kavitha Sravanthi And Addula Saikumar, "An FPGA Based Double Precision Floating Point Arithmetic Unit Using Verilog", *International Journal Of Engineering Research & Technology*, Pp.576-581, 2013.
- [5] Aboobacker Sidheeq, "Four Stages Pipelined 16 bit RISC on Xilinx Spartan 3AN FPGA", *International Journal of Computer Applications*, pp.29-38, 2012.
- [6] Preetam Bhosle and Hari Krishna Moorthy, "FPGA Implementation of Low Power Pipelined 32-Bit RISC Processor", *International Journal of Innovative Technology and Exploring Engineering*, pp.66-71, 2012.
- [7] J. Vijay Kumar, B. Naga Raju, C. Swapna and T. Ramanjappa, "Design and Implementation of Low power pipelined 64-bit RISC processor using FPGA", *International Journal of Advanced Research Engineering and Technology*, pp.61-69, 2014.
- [8] J. Ravindra and T. Anuradha, "Design of Low power RISC Processor by Applying Clock Gating Technique", *International Journal of Engineering Research and Applications*, pp. 94-99, 2012.
- [9] https://www.altera.com/en_US/pdfs/literature/hb/max-v/mv51001.pdf
- [10] J. Vijay Kumar, B. Naga Raju, M. Vasu Babu and T. Ramanjappa "CPLD based design and implementation of low power pipelined 64-Bit RISC processor", *International Journal of Emerging Technology and Advanced Engineering*, pp.274-277, 2015.