ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 3, Issue 8, August 2015

Implementation of a 32 bit RISC processor with memory controller by using VHDL

Sd. Ameerunnisa Begum[1], Dr. M. Sailaja[2]

M.Tech Student, Department of ECE, Jawaharlal Nehru Technological University Kakinada, Kakinada, India[1]

Professor, ECE, Jawaharlal Nehru Technological University Kakinada, Kakinada, India [2]

Abstract: The design of a RISC processor with memory controller is done in this paper. For the best use of memory, this processor contains a memory module and control unit which are included in the processor design. This Processor embodies 15 basic instructions involving Arithmetic, Logical, and Data Transfer and control instructions. To implement these instructions the design incorporates various design blocks like Control Unit (CU), Arithmetic and Logic Unit (ALU), Accumulator, Program Counter (PC), Instruction Register (IR), Memory and additional logic. A new architecture is implemented for the proposed RISC processor with 32 bit input. The processor has small instruction set and control logic design is very much simplified. It is basically designed in order to achieve faster executions and the processor can execute each instruction within one clock cycle. All individual logic blocks are simulated using ModelSim Simulator and top module is obtained by connecting all the blocks in an order.

Keywords: 32 bit RISC processor, memory (RAM and ROM), Memory module and control unit, simple instruction set, faster executions.

I. INTRODUCTION

Now a day's RISC processors are used in many fields such as industrial applications, electronic gadgets, and tablet computers to world's fastest super computers like K computers. Reduced instruction set architecture includes simple instruction set rather than a complex instruction set. Simple instruction set increases the speed of the processor by keeping this point highlighted in this paper a RISC [3]. [7] processor is designed with 15 basic instructions to perform arithmetic, logical, data transfer and control operations.

A simplified memory [1], [4] controller module is implemented with different control signals to control the data flow from and to memory unit. While an integrated memory controller has the potential to increase the system's performance. The memory controller design includes both RAM and ROM integrated as one memory unit. This reduces the cost per memory unit. Simplified control unit with different control signals to control the data flow, and to control the operation of remaining modules. These types of processors can be used for general purpose applications.

II. RISC PROCESSOR DESIGN

A new RISC processor is designed with memory controller [3], [5]. The processor is a 32 bit processor [6] and has a simple instruction set. This Processor embodies 15 basic instructions involving Arithmetic, Logical, Data Transfer and control instructions. To implement these instructions the design incorporates varies design blocks like control unit (CU), Arithmetic and logical unit (ALU), Accumulator, Program counter (PC), Instruction register (IR), Memory and additional logic. This processor is designed in order to achieve higher performance with parallel processing and pipelining. Each instruction is executed by using a single clock cycle.

The total processor is designed with various logic blocks. The input to this processor is a Data-in of 32 bit. Executable clock is a positive edge trigger clock input, reset and y are one bit inputs and fetch is a clock input given to the processor. Data input is given to instruction register and memory unit. Instruction register stores the instruction currently being executed. The inputs to the Instruction register are Data in, reset, load instruction register (ldir).

The output of instruction register is 4 bit opcode (MSB bits of 32 bit input) and the remaining 28 bits of given input data are obtained as irout, this irout is given as input to the multiplexer. Program counter stores the address of the instruction being executed at the current time.

Whenever the fetch and incpc are HIGH then PC increments its value by '1'. The outputs of Instruction register and Program counter are fed to Multiplexer as inputs. Depending on the fetch input, multiplexer generates output, which is taken as input address for memory. Memory unit write/read the data to the given/from input address. The data read from memory is given as one of the inputs to ALU and another input to ALU is the output of accumulator.

The ALU performs different operations on the given two inputs. According to the opcode, it produces output as Acc of 32 bits which is stored in accumulator. Control unit design is very much simplified. It sends command signals to other modules for different opcode values. when 'y' input of buffer is HIGH, the data out is obtained from buffer.

All the individual modules in RISC processor are synthesized and simulated using Xilinx ISE software and ModelSim simulator.

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526
INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 3, Issue 8, August 2015

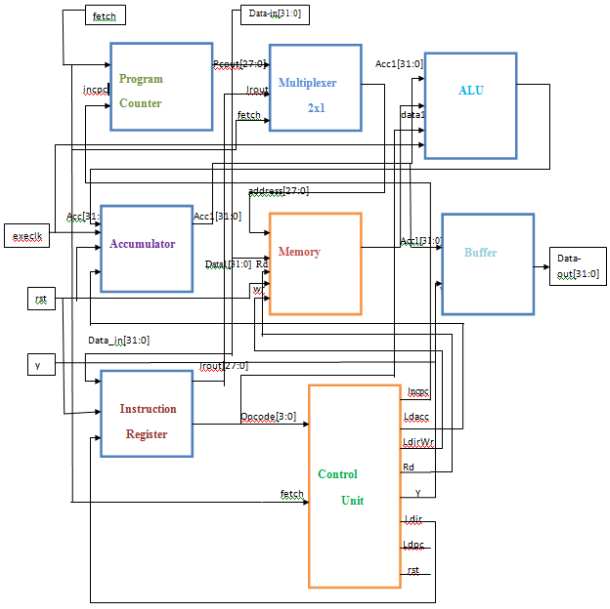

Figure1. Block diagram of RISC processor

III. SIMULATION RESULTS

A. Accumulator

In this simulation result the RISC processor is simulated using Xilinx ISE software and ModelSim simulator. As shown in Figure 3, the RTL schematic view of Accumulator is generated after the synthesis. The simulation waveform as shown in figure 4, is generated after simulation in ModelSim simulator. This waveform specifies how the data is stored in accumulator. When rst is '0' the accumulator output is set to zero otherwise if ldacc is '1' then the accumulator generates output.

Figure2. RTL schematic view of Accumulator

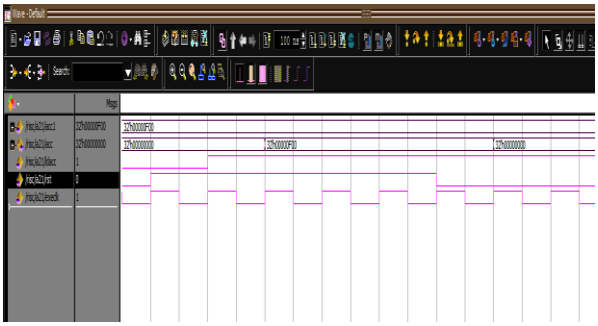

Figure3. Simulation waveform of Accumulator

B. Arithmetic and logical unit

ALU module is synthesized and simulated using Xilinx software and ModelSim simulator. The RTL view of ALU as shown in figure 6. ALU is designed to perform 15 basic operations such as arithmetic, logical and data operations. The two 32 bit inputs to the ALU are acc and data, and 4 bit input opcode and execlk is a positive edge trigger clock input. The waveform in figure 7, specifies how different arithmetic, logical and data transfer operations are performed for different opcode values.

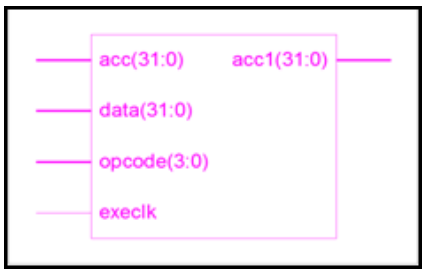

Figure4. RTL schematic view of ALU

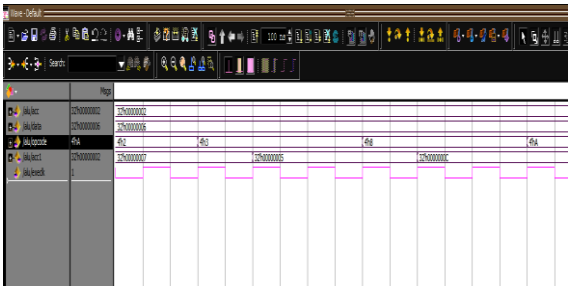

Figure5. Simulation waveform of ALU

C. Instruction Register

Instruction register stores the instruction currently being executed. This module is synthesized and simulated using Xilinx ISE software and ModelSim simulator. The RTL view of Instruction register is shown in figure 8. The input data is given as input to the instruction register and rst, ldir are '1' bit inputs. The waveform of instruction register is shown in figure 9, which specifies The output of instruction register is 4 bit opcode (MSB bits of 32 bit input) and the remaining 28 bits of given input data are obtained as irout, if rst and ldir values are '1'. Otherwise opcode will be set to all 1's and irout set to zero.

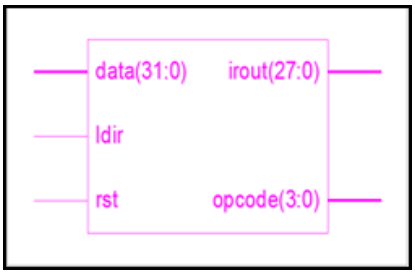

Figure6. RTL schematic view of Instruction Register.

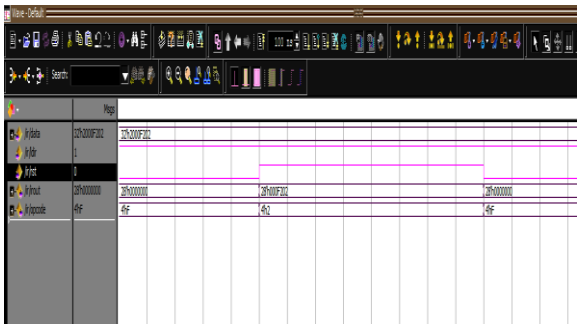

Figure7. Simulation waveform of Instruction Register

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526
INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 3, Issue 8, August 2015

D. Program counter

Program counter module is synthesized and simulated by using Xilinx ISE software and ModelSim simulator. The RTL schematic view as shown in figure 10 represents the internal architecture of program counter. As shown in figure 11, fetch input is a positive edge trigger clock applied to this module. If incpc value is set to '1' and program counter value reaches to 255 then pcout is set to zero otherwise pcout is incremented by one.

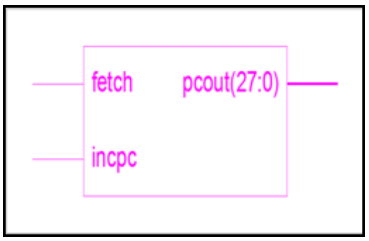

Figure8. RTL schematic view of Program Counter

Figure9. Simulation waveform of Program Counter

E. Multiplexer

The module of multiplexer is synthesized and simulated using Xilinx ISE software and ModelSim simulator. The RTL schematic as shown in figure 12 specifies the architecture of Multiplexer. The inputs of mux are irout and pcout and fetch. The waveform of Multiplexer as shown in figure 13, represents that If fetch value is '1' then multiplexer output is pcout otherwise irout.

Figure10. RTL schematic view of Multiplexer

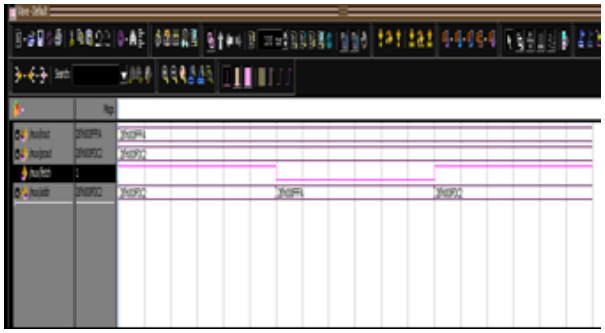

Figure11. Simulation waveform of Multiplexer

F. Memory Module

The Memory Module is designed with 256 memory locations and each memory location contains 32 bits. The memory module internal architecture is represented using RTL schematic view as shown in figure 14. This memory [2] module includes both RAM and ROM memories and both write and read operations are performed. As shown in figure 15, the waveform represents that for wr='1' and rd='0' the given data is written to given input address location, and if rd='1' and wr='0' the data is read from the given input address location. If reset is zero then the data in given input address location is set to zero.

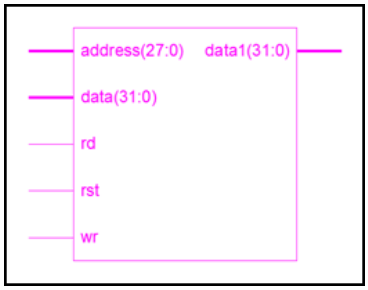

Figure12. RTL schematic view of Memory Module

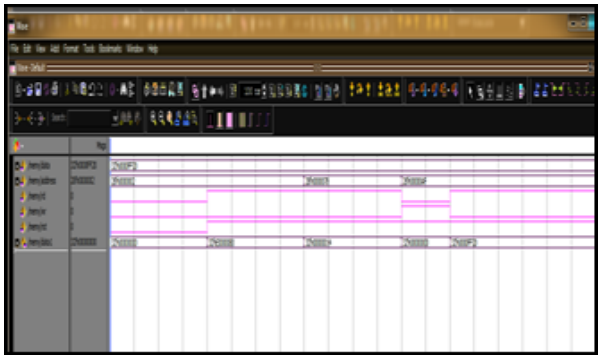

Figure13. Simulation waveform of Memory Module

G. Control Unit

In this processor the control unit logic is very much simplified. The control unit module is synthesized using Xilinx ISE software and Simulated using ModelSim simulator. The RTL schematic view as shown in figure 16, represents the architecture of control unit with fetch and opcode as inputs. The waveform in figure 17 is generated using ModelSim simulator represents that for different opcode values the control unit produces different controlling signals to remaining modules in the processor. For opcode="1011" the value incpc<='1'which is fed to program counter, for opcode="1110" the values are assigned as y<='1', rd<='0' and wr<='1'.

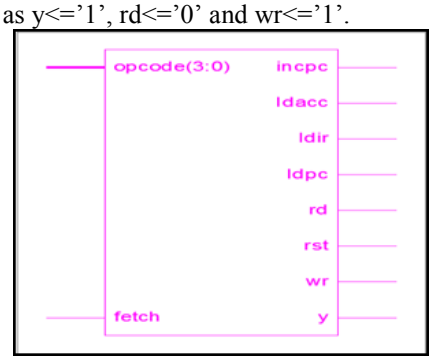

Figure14. RTL schematic view of Control Unit

DOI 10.17148/IJIREEICE.2015.3824

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526
INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 3, Issue 8, August 2015

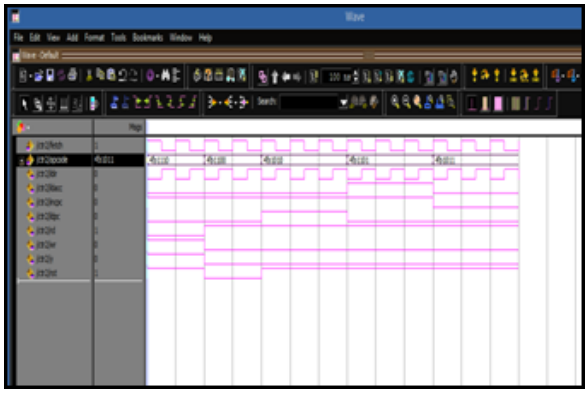

Figure15. Simulation waveform of Control Unit

H. Buffer

The buffer module is synthesized and simulated using Xilinx ISE software and ModelSim simulator. The RTL schematic in figure 18 specifies the internal structure of buffer with Acc1 and y inputs. The waveform in figure 19, shows that, if y='1' then buffer gives output otherwise the buffer does not generate output.

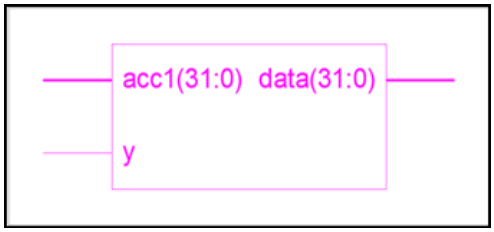

Figure16. RTL schematic view of Buffer

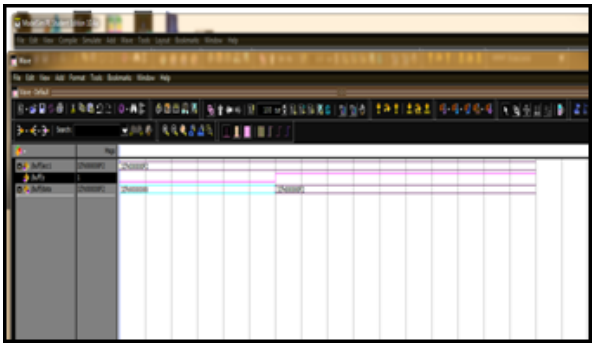

Figure17. Simulation waveform of Buffer

I. Top Module

The top Module represents the internal architecture of the processor. The RTL schematic view of top module is shown in figure 20, which specifies that the processor design is very much simplified with fewer instructions. The inputs to the processor are Data in of 32 bits, execlk which is a positive edge triggered clock input, fetch is also a clock input, reset and y are one bit inputs. As shown in simulation result in figure 21, the waveform represents that the data is applied as input to both instruction register and memory module, the data is read and written to memory for logic values of rd and wr inputs. Then after the data read from memory is fed to ALU as data1 and acc1 is another input to ALU, the operations like arithmetic, logical and data transfer are performed in ALU for different opcode values. The output from ALU is stored in accumulator, after that the final output is obtained through buffer from accumulator. The total process runs in a cyclic order with different logical blocks like ALU, CU, Memory, IR, and PC and so on.

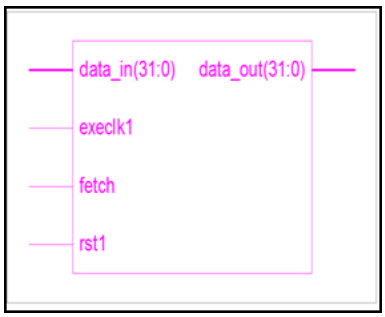

Figure18. RTL schematic of top module

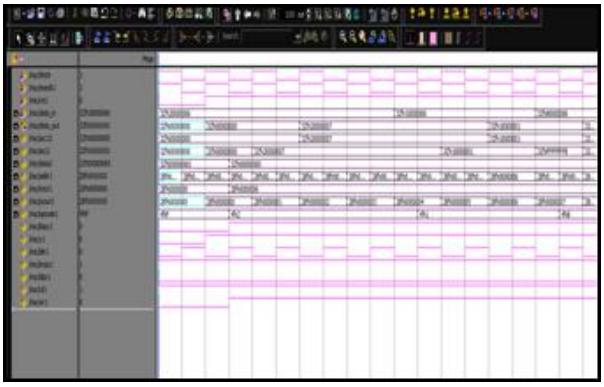

Figure19. Simulation waveform of top module

IV. CONCLUSION

In this paper RISC processor with memory controller is implemented and simulated by using ModelSim simulator. RISC processor are used in many fields such as industrial applications, electronic gadgets, and tablet computers to world's fastest super computers like K computers. For the best use of the memory the simple RISC processor is designed. The processor design incorporates various design blocks like ALU, Accumulator, CU, Memory, Program counter, Instruction register and buffer. The memory and control unit blocks are implemented with simple logic which increases the speed of the processor. The processor executes each instruction within one clock cycle, this is another advantage included in design of processor.

REFERENCES

[1]. KouhelAgarwal, Vijay Kumar Magraiya, Dr. Anil Kishore Saxena, "Verification and simulation of new designed NAND flash memory controller", *IEEE conference on Communication systems and network technologies*, page 762-766, year 2013.

[2]. Chul Lee, Sung Hoon Beak, KyunHo Park, "A hybrid flash file system based on NOR and NAND flash memories for embedded devices", *IEEE Transactions on computers*, volume 57, pages 1002-1008, year 2008.

[3]. Trivedi.P, Tripathi.R.P, "Design & Analysis of 16 bit RISC Processor Using low Power Pipelining", *IEEE international conference on Computing, Communication and automation*, pages 1294-1297, year 2015.

[4]. Hoeseung Jung, Sanghyuk Jung, "Architecture Exploration of Flash Memory Storage Controller through a Cycle Accurate Profiling", *IEEE Transactions on Customer Electronics*, Volume: 57, Page 1756–1764, 2011.

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 3, Issue 8, August 2015

[5] Navneet .K, Adesh Kumar, Lipika Gupta, "VHDL design and synthesis of 64 bit RISC processor system on chip (Soc)". *IOS journal of VLSI and Signal processing, volume 3, issue 5, 2013.*

[6]. Mr. S. P. Ritpurkar, Prof. M.N Thakare, Prof. G. D. korde, "Synthesis and simulation of a 32 bit RISC processor using VHDL", *IEEE conference on Advances in Engineering and Technology Research, 2014.*

[7]. Sakthikumaran. S, Salivahanan. S, KanchanaBhaaskaran. V. S, "16 bit RISC processor design for convolution applications", *IEEE International Conference on Recent Trends in Information Technology*, pages 394-397, year 2011.

BIOGRAPHIES

Sd. Ameerunnisa Begum, Received a B.Tech degree in the department of electronics and instrumentation engineering from Lakireddy Bali Reddy College of Engineering. Currently pursuing M.Tech in Jawaharlal Technological University Kakinada in the department of Electronics and Communication Engineering.

Dr. M. Sailaja, pursued Ph.D., M.S (U.S.A), B.Tech. Currently working as professor with 17 years of teaching experience in the department of Electronics and Communication Engineering in Jawaharlal Nehru Technological University Kakinada. Her area of interests includes Computer networks and parallel processing.