Journal of Advances in Science and Technology
Vol. 12, Issue No. 25, (Special Issue) December-2016, ISSN 2230-9659

Implementation of Riscprocessorusing Verilog

Prof. Pramod V. Patil[1]*, Miss. Namrata R. Patil[2], Miss. Priyanka R. Jadhav[3]

[1]Professor, Department of ECE, Hirasugar Institute of Technology, Nidsoshi, India

[2]B.E, Students Department of ECE, Hirasugar Institute of Technology, Nidsoshi, India

[2]B.E, Students Department of ECE, Hirasugar Institute of Technology, Nidsoshi, India

Abstract – RISC is a design philosophy to reduce the complexity of instruction set that in turn reduces the amount of space, cycle time, cost and other parameters taken into account during the implementation of the design. The intent of this paper is to design and implement 16 bit RISC processor using Project Navigator tool. This processor design depends upon design specification, analysis and simulation. It takes into consideration very simple instruction set. The momentous components include Control unit, ALU, shift registers, IDU, PC, Clock generation unit and accumulator register.

----------♦----------

I. INTRODUCTION

Now a days, Computers are mainstream in quotidian activities. RISC Processor is a CPU design strategy that uses simplified instructions for higher performance with faster execution of instruction. It also reduces the delay in execution. It uses general instructions rather than specialized instructions. They are less costly to design, test and manufacture. This has helped in implementation of RISC in technological field. Its range of application includes signal processing, convolution application, supercomputers such as K computers and wider base for smart phones. In this work, an 16 bit RISC processor is presented with higher performance and efficiency being the main aim. This processor comprises of Control unit, general purpose registers, Arithmetic and logical unit, shift registers. Control unit consists of programcounter and IDU. According to the instruction to the control unit generate signal to decode the instruction. The architecture supports 16 instructions for arithmetic, logical, shifting and rotational operations

II. LITERATURE SURVEY

Table 2.1

Year	Author name	Tool used	Work done	Result/Conclusion
2011	P.Ganesh Kumar	Xilinx ISE 10.1	Pipelining, efficient cache implementation, power design consideration and speculative branching are some of the features that are used to improve the performance of the RISC architectures.	It can be concluded that the RISC philosophy, in general has proved superior to the CISC.
2013	Anand Nandakumar Shardul	Xilinx ISE 14.2	16-bit non-pipelined RISC processor, which is used for signal processing applications	The simulation result shows that the processor is capable of implementing the given instruction in single clock cycle.
2015	Niklaus Wirth	Xilinx ISE 10.1	Pipelining, efficient cache implementation, power design consideration and speculative branching are some of the features that are used to improve the performance of the RISC architectures.	It can be concluded that the RISC philosophy, in general has proved superior to the CISC.

III. BLOCK DIAGRAM OF RISC PROCESSOR

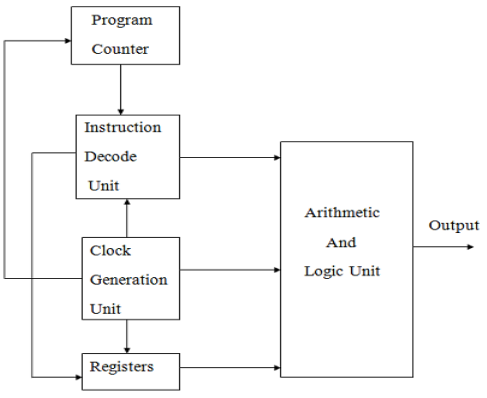

Fig3.1. Risc processor

- **Arithmetic Logic Unit (ALU)**

The arithmetic/logic unit (ALU) executes all arithmetic and logical operations. Arithmetic operations either take two registers as operands. The result is stored in a third register. The arithmetic/logic unit can perform arithmetic operations or mathematical calculations like addition, and subtraction and also performs logical operations include Boolean comparisons, such as AND, OR, XOR, NAND, NOR and NOT operations.

- **Registers**

Holds values of internal operation, such as the address of the instruction being executed and the data being processed i.e. Program Counter Register, Status Register. We have designed A,B,C,D,E,F,G,H,I,J Registers. If register content is 0 then it is source, if 1 then it is destination.

- **Instruction decode unit**

The main function of the instruction decode unit is to use the 9-bit instruction provided from the previous instruction fetch unit to index the register file and obtain the register data values. The instructions opcode field bits are sent to a control unit to determine the type of instruction to execute. The type of instruction then determines which control signals are to be set and function that Execute unit is to perform, thus decoding the instruction.

- **Program counter**

Program counter is a register in a computer processor that contains the address (location) of the instruction being executed at the current time. As each instruction gets fetched, the program counter increases its stored value by 1. After each instruction is fetched, the program counter points to the next instruction in the sequence. When the computer restarts or is reset, the program counter normally reverts to 0.

- **Software used**

Language verilog

Operating System Windows 7

IDE Xilinx 13.1 3

IV. RISC PROCESSOR VIEWS

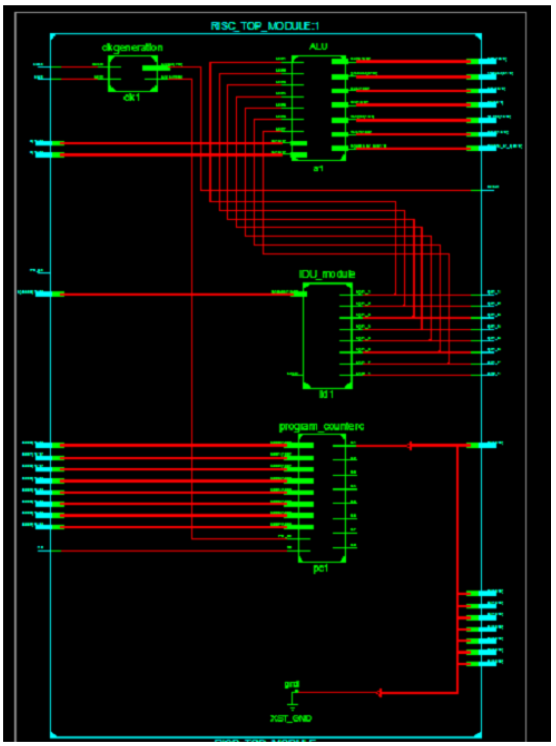

Fig4.1. processor view

V. RISC PROCESSOR TECHNOLOGICAL VIEW

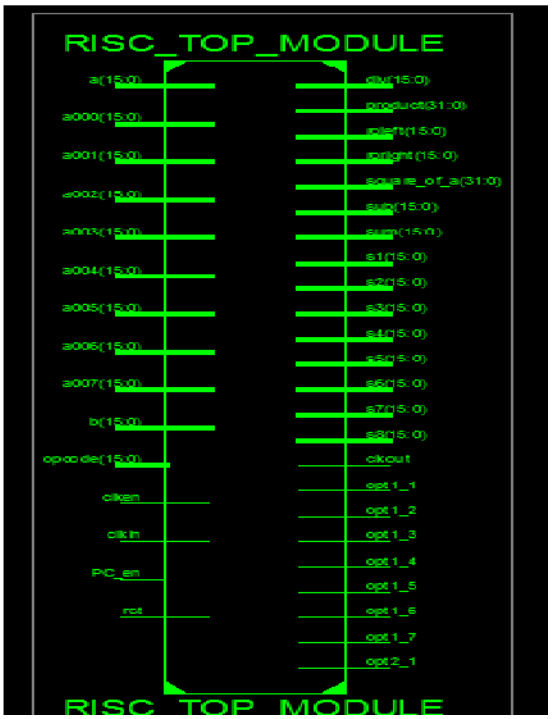

Fig5.1. technological view

VI. FLOW CHART

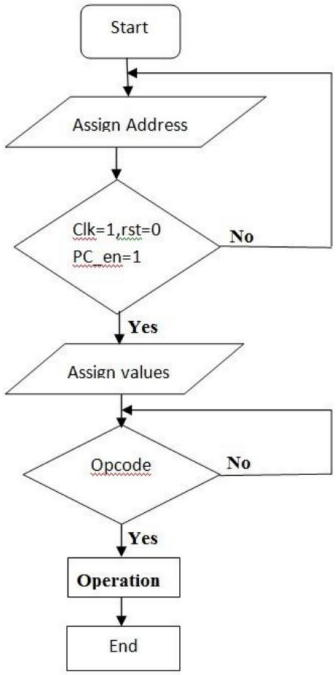

Fig 6.1. Flow chart

VII. DIFFERENT OPERATION

Table 7.1

Opcode	Operation
0000h	Addition
0080h	Product
0100h	Substraction
0180h	Division
0200h	Rotate right
0280h	Rotate left
0300h	Square

VIII. RESULTS

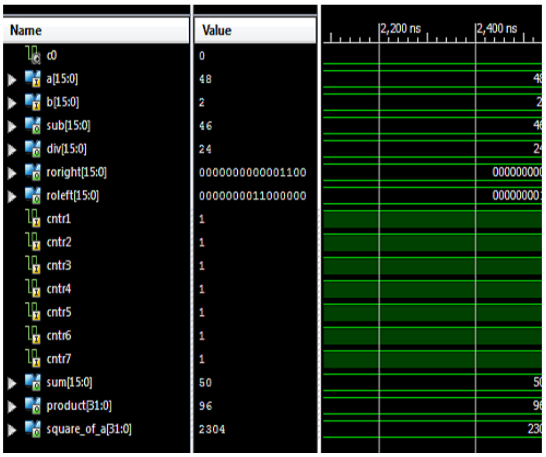

Fig8.1. ALU Result

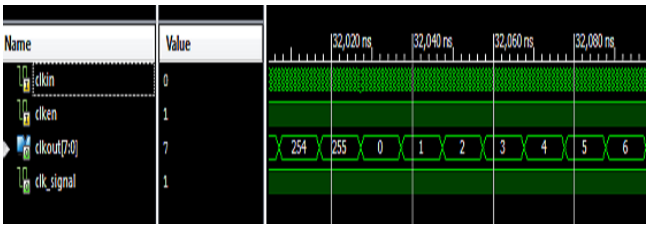

Fig8.2. CLK generation Result

Fig8.3. Register Result

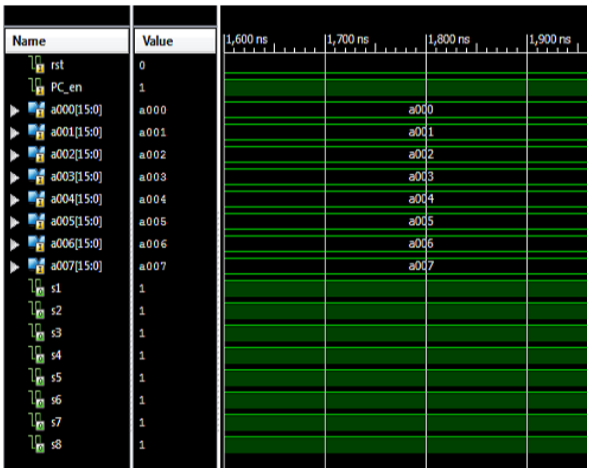

Fig8.4 .Program counter result

IX. ADVANTAGES

- RISC allows freedom of using the space on microprocessors because of its simplicity.
- Instead of using stack many RISC processors use the registers for passing arguments and holding the local variables.
- RISC functions uses only a few parameters, and the RISC processors cannot use the call instructions
- The speed of the operation can be maximized and the execution time can be minimized.

X. DISADVANTAGES

- With the increase in length of the instructions, the complexity increases for the RISC processors to execute due to its character cycle per instruction.
- The first level cache of the RISC processors is also a disadvantage of RISC, in which these processors have large memory cache on the chip itself.
- For feeding the instructions they require very fast memory systems.

XI. APPLICATIONS

- RISC processor is used to implement DSP application such as convolution and correlation.
- RISC processor is used for general purpose application.

XII. CONCLUSION

The design of a single cycle 16-Bit RISC processor has been presented. Carry select adder and Daddas multiplier structures have been employed in the RISC architecture. The processor has been designed register A to J. The processor design promises its use towards any signal processing applications. Total memory usage is 278936 kilobytes and timing required for execution is 1.092ns. 6

XIII. REFERENCES

R. Uma, Mar-Apr 2012, “Design and Performance Analysis of 8-bit RISC Processor using Xilinx Tool”.

Galani Tina G., RiyaSaini and R.D.Daruwala, July-2013, “Design and Implementation of 32 – bit RISC Processor using Xilinx”.

GalaniTina R.D.Daruwala, February 2013, “ Performance Improvement of MIPS Architecture by Adding New Features”

SamiappaSakthikumaran,S.Salivahananand V.S.KaanchanaBhaaskaran , June 2011, “16-Bit RISC Processor Design For Convolution Application”,IEEE International Conference on Recent Trends In Information Technology, pp.394-397.

Corresponding Author

Prof. Pramod V. Patil*

Professor, Department of ECE, Hirasugar Institute of Technology, Nidsoshi, India

E-Mail – patil.pramod080@gmail.com