International Journal of Scientific and Technical Advancements
IJSTA
ISSN: 2454-1532

# VHDL Design & FPGA Implementation of 16-bit Microprocessor

Surbhi Sharma[1], Meena Kohli[2], Rachit Vaid[3]
[1, 2, 3]Department of ECE, MIET, KotBhalwal, Jammu, India
Email address: surbhi.ece@mietjammu.in, meena.ece@mietjammu.in, rachit88vaid@gmail.com

***Abstract*—**One of the important aspects of modern electronic technology is embedded systems based on microcontrollers. Microcontrollers are such an important part of this digital word where everyone wants to complete ones intended task in as minimum time as possible. So this work is devoted for the design and implementation of 16-bit Microprocessor RISC Architecture and then optimizing it for high speed and low power. Presented Microprocessor RISC have instruction set of 23 instructions and each instruction is executes in a single cycle including branch instructions which is an enhancement to the previously version of the RISC.
Microprocessor is a programmable register based electronic device which accepts binary data as input, processes on data based on binary instructions and provide result as output. Microprocessor integrates huge processing power in a compact space. VLSI designing use to implement microprocessor makes it possible to program it onto a single integrated circuit using VHDL. A 16-bit microprocessor with 16 instructions is designed. The FPGA which is used for the implementation of the circuit is the Xilinx Spartan3E (device) XC3S500 (family) FG320 (package) FPGA device. The working environment/tool for the design is the Xilinx ISE Design suit 13.2. Modelling Simulator ISim is used for the functional, simulation and implementation (post-translate, post-map and post-place & route simulation) of the VHDL model.

***Index Terms*—** Arithmetic logic unit(ALU); Field programmable gate array(FPGA); Logic Gates, Microprocessor; Register transfer level (RTL); Synchronous circuit; VHDL; VLSI; Very high speed integrated circuit (VHSIC) hardware description language.

## I. INTRODUCTION

A microprocessor is a multipurpose, programmable, clock-driven, register-based electronic device that reads binary instructions from a storage device called memory accepts binary data as input and processes data according to those instructions and provides results as output. It is also called that a microprocessor is a programmable integrated device that has computing and decision –making capability similar to that of the central processing unit of a computer.

A simple 16-bit synchronous microprocessor has been designed. It consists of an instruction set of 9 bits (i.e. 5 bits for opcode & 4 bits for special instructions like shift, rotate etc.) having 20 instructions including 16 Arithmetic and logical instructions and control instructions for memory and ports. The instructions are fetched which is decoded at the control unit.

The instruction and instruction data is sent, along with system data, to the corresponding blocks. Further a 16x16 array memory, designed to store the data, can store and read sixteen 16-bit data arrays. The memory system stores the program as well as the data used by the program. The data can also be taken from the 16x16 array type memory for processor operations.

Various tools are put forth in designing of this system. In the present case microprocessor is bricked up using synthesized operations in the form of objectives and broader aspects. Fig. 1.1 shows the organization of the processor design.

*a) System overview*

The design has been represented by separate modules. The main focus amongst them for Operational Design, Software Design and Hardware Design.

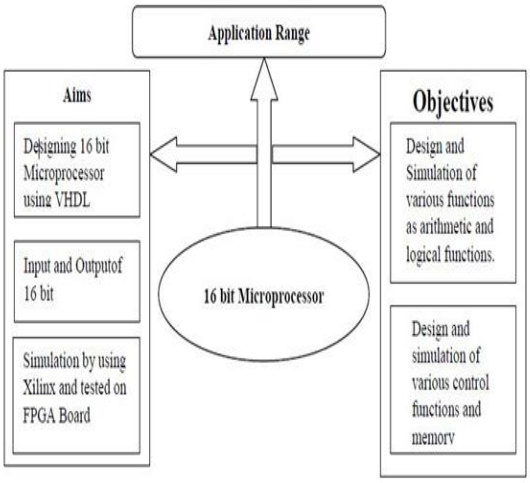


Fig. 1.1: Organization of processor design.

*b) Operational overview*

The operational view deals with various kinds of operation which a microprocessor can perform. The CPU (Central Processing Unit) is the "brain" of computer. It is composed of several parts like data path, control path and memory units. At each clock cycle, Control Unit is needed to generate the control signals automatically for operating the data path. It is based on the finite state machine concept. The control unit for a processor basically cycles through three main steps, usually referred to as the instruction cycle i.e. fetch an instruction, Decodes the instruction and Executes the instruction. Second part deals with the Arithmetic Logic Unit which perform



Surbhi Sharma, Meena Kohli and Rachit Vaid, "VHDL Design & FPGA Implementation of 16-bit Microprocessor," *International Journal of Scientific and Technical Advancements*, Volume 2, Issue 4, pp. 287-292, 2016.

arithmetic computations such as addition, subtraction, multiplication, division, Increment, decrement and logical functions such as AND, OR, XOR, left shift, right shift etc.

*c) Software overview*

Interfacing with VHDL software used in this system, reduces the complexity and also provide the graphic presentation of the system, VHDL is advantageous when used for systems design is that it allows the behavior of the required system to be described (modeled) and verified (simulated) before synthesis tools translate the design into real hardware (gates and wires).This not only indulge in compilation but also produces waveform results. For performing compilation and simulation of any logic circuit design, few sophisticated Computer Aided Design (CAD) tools such as Altera's II and Xilinx web pack are used

*d) Hardware overview*

The Fundamental building block of microprocessor instantiates components control unit and memory and specifies the necessary signals to connect the components. Component memory is a memory device and contains the instructions and data for the CPU to execute. Component control unit is an RTL implementation of the CPU device that is simulated for correctness and synthesized to implement the design. It includes clock signal, reset valid data in, IR, address register, ready signal and data is required for operation. A final point is that when a VHDL model is translated into the "gates and wires" that are mapped onto a Programmable logic device such as a CPLD or FPGA, and then it is the actual hardware being configured, rather than the VHDL code being "executed" as if on some form of a processor chip. Instruction register is for storing the instruction being fetched from the memory. The program counter holds the address of the memory of the current instruction. After the execution of instruction, the program counter move to the next instruction. If there is branch instruction, the program counter is loaded with the address of the next instruction. Then the value of the program counter is copied by the control unit to the address register, which gives the new address in address bus .The process of storing data into memory is called writing and retrieving data or op-code from the memory is called reading.

## II. IMPLEMENTATION

The basic block showing various input and output is as:

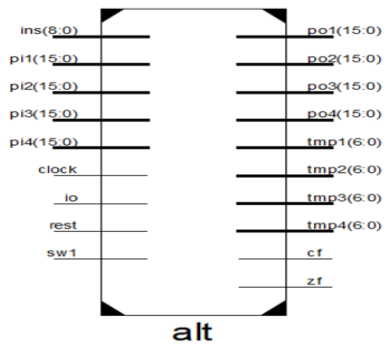


Figure 2.1: RTL of microprocessor.

The microprocessor consists of four basic blocks. These are:

- Port Registers
- Control Unit
- 16X16 Array Type Memory
- Arithmetic & Logic Unit

In port registers, there are four input ports and four output ports. Enable pin (sw1) is kept high to order to select the port registers. In each port data of 16 bits can be fed via input port (p1, p2, p3, p4). Similarly, the 16-bit output data can be taken from output ports (p1o, p2o, p3o, p4o).

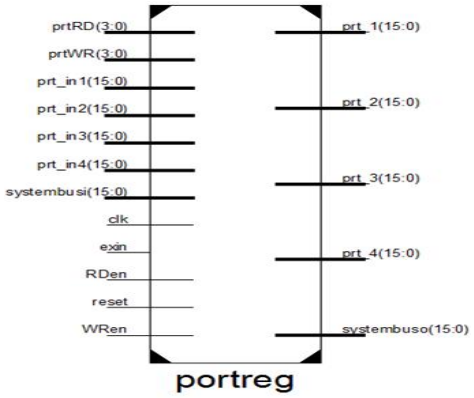


Fig. 2.2: RTL of port registers.

Selection of port number is based on the selection lines corresponding to pin 3 down to 0 of ins for control unit. When RDen is kept high, the data input to the port registers is fed to the control unit via sb1 as data-in. Similarly, when WRen is kept high, the 16-bit data from the control unit is fed to the port registers as system bus input for port registers, which is then taken as output from the output port registers. The simulation result for Port Register block shows the inputs and corresponding outputs with certain signals getting high in the process.

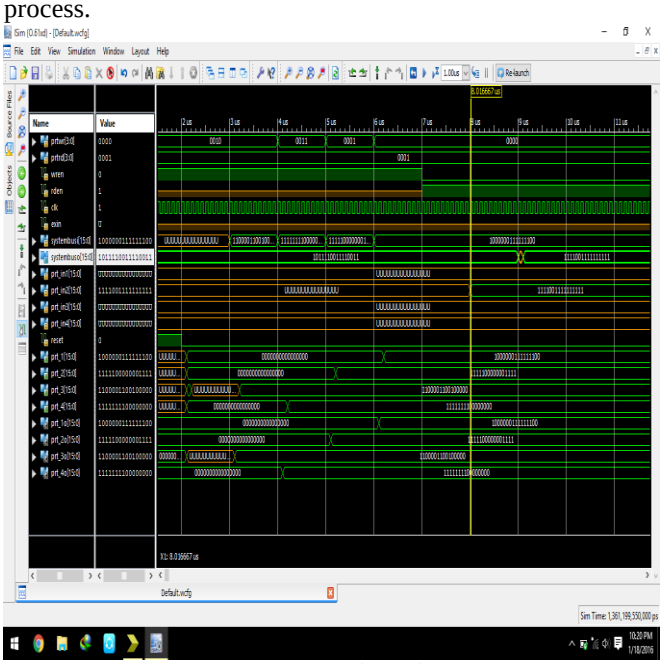

Fig. 2.3: Simulation waveform of port registers.



Surbhi Sharma, Meena Kohli and Rachit Vaid, "VHDL Design & FPGA Implementation of 16-bit Microprocessor," *International Journal of Scientific and Technical Advancements*, Volume 2, Issue 4, pp. 287-292, 2016.

The Control Unit (CU directs operation of the processor. It tells the computer's memory, Arithmetic/logic unit and input and output devices about the proper respond to a program's instructions. It directs the operation of the other units by providing timing and control signals. Most computer resources are managed by the CU. It directs the flow of data between the devices.

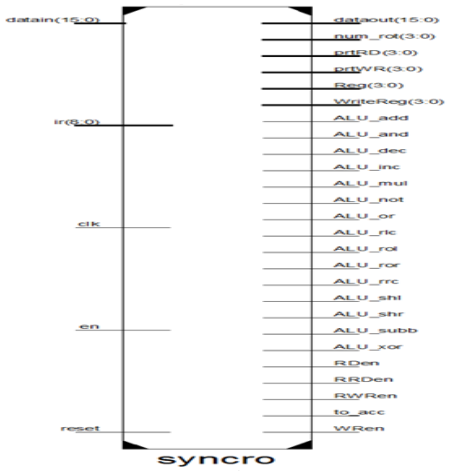

Fig. 2.4: RTL of control unit.

The simulation using Isim shows the decoding of instructions in the control unit. It shows either of the 20 instructions being activated corresponding to their opcode which is encoded under bit 8 down to 4 of ir of control unit.

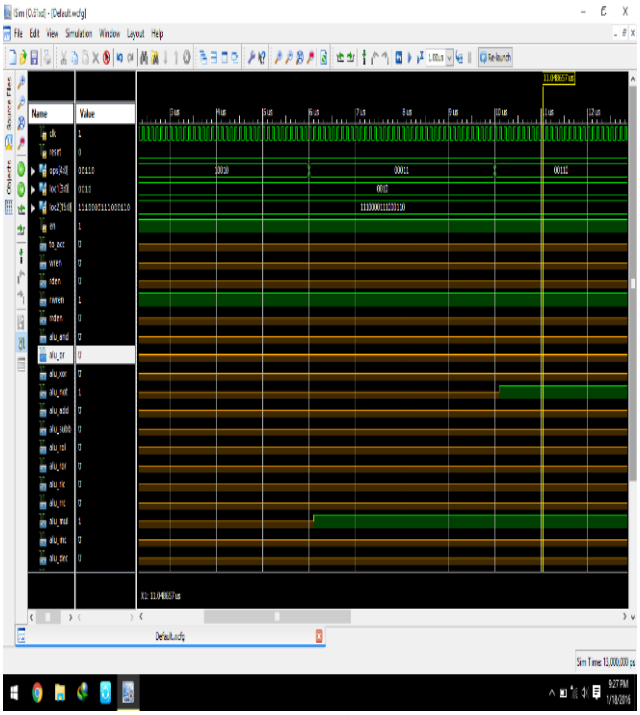
Fig. 2.5: Simulation waveform of control unit.

A 16x16 bit array type memory is designed. It has main inputs which are: -

- Reg(Read register)
- WriteReg(Write register)
- RRDen(Register read enable)
- RWRen(Write Register enable)
- System Bus input

Reg & WriteReg are taken as 4-bit Input from Control Unit where Reg is used to define array to read data from and WriteReg is used to select array from where to write data. They both decide which 16 bit is to be selected in 16X16 array type memory. The 16-bit system bus input data is also fed to the memory from the ALU. The signals RRDen and RWRen decides whether the data is to be read from or written to the memory. When RWRen is kept high, the data is fetched from ALU as system bus input to the memory. Similarly, when RRDen is kept high the system bus output of memory displays the data stored in memory. The output is fed to 7 segment displays which show result as numeric values to user. The simulation results showed data written to as well as read from one of the 16, 16-bit arrays corresponding to the value of Reg and WriteReg.

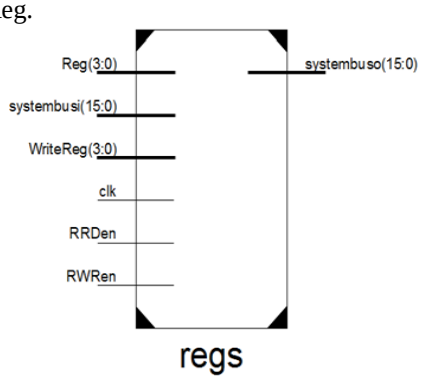

Fig. 2.6: RTL of memory.

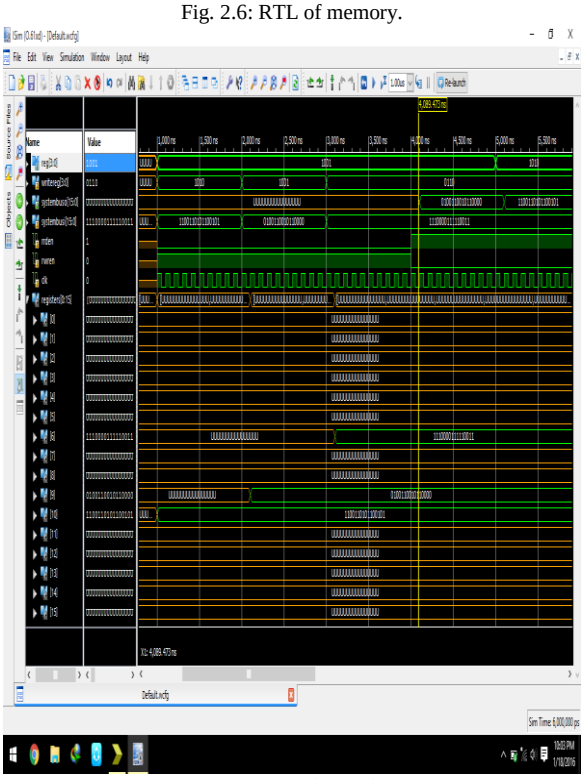
Fig. 2.7: Simulation waveform of memory unit.



Surbhi Sharma, Meena Kohli and Rachit Vaid, "VHDL Design & FPGA Implementation of 16-bit Microprocessor," *International Journal of Scientific and Technical Advancements*, Volume 2, Issue 4, pp. 287-292, 2016.

A **16-bit ALU** is designed, which performs 16 operations. It has 20 inputs: -

- 16-bit system bus input data
- Input for Increment/decrement operation
- Inputs for Arithmetic operation like addition, subtraction, multiplication etc.
- Inputs for special operations like rotate (with and without carry), shift (left and right)
- Inputs for Logical operations (and, ex-or, not, or).

As ALU is the most essential entity in processor since it is concerned with arithmetic, logical and decision making operations such as AND, OR, NOT, NAND, etc. It is considered to be so important as the entire processor depends on it. In ALU, two input data buses are used to provide data and the resultant output is obtained based on the desired operation. The output is send to the memory register for temporary storage of data.

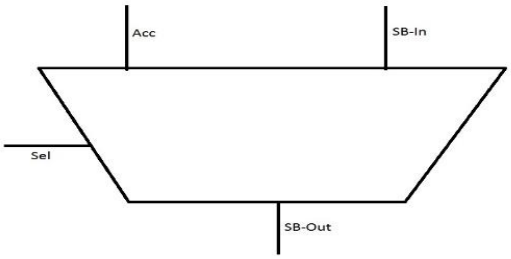


Fig. 2.8 Basic Block of ALU.

The design of an ALU decides how powerful CPU is, but at the same time it consumes large energy and generate heat. Therefore, an efficient designing of ALU is necessary to make the CPU powerful. Following is the RTL of the ALU which shows decoded instructions being fed as input to process data fed through system bus.

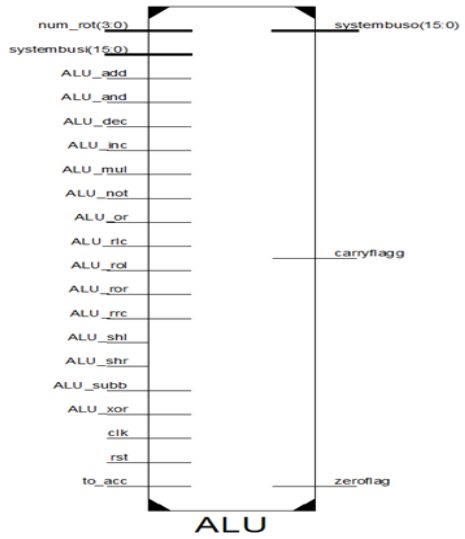


Fig. 2.9: RTL of ALU.

The simulation results show the processed output. The pin num_rot provides with additional data required for certain logical operations. It defines the number of bits for rotation as well as shift.

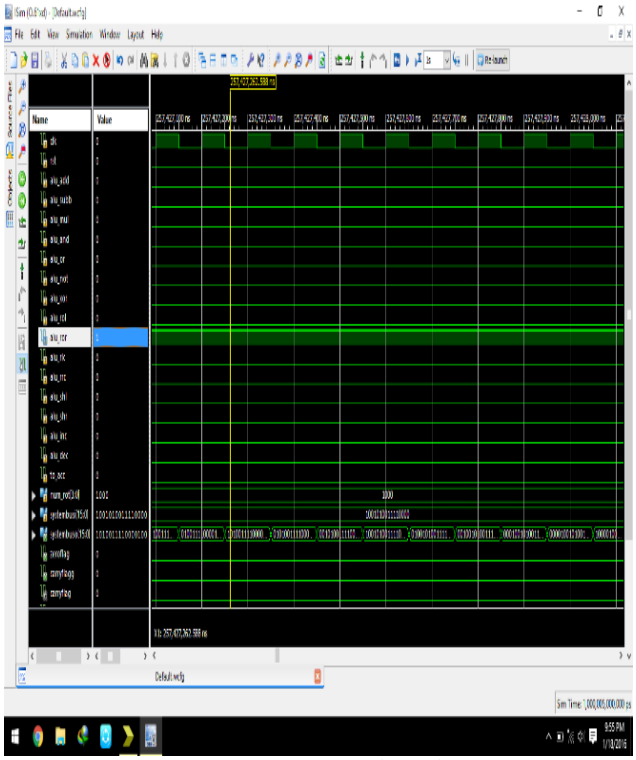

Fig. 2.10: Simulation Waveform of ALU.

The various list of Arithmetic as well as logical operations performed in an ALU are listed below along with the opcode bits from which they are decoded. The simulation results for main are obtained by portmapping the various blocks shown in figure 2.10. The data fed from the ports is processed by the ALU once fed from the control unit along with the decoded opcode deciding the operation to be performed. The processed output is fed to memory to be stored in one of the sixteen 16-bit arrays. The stored data can be then displayed.

Table 1: Various operations performed in the ALU.

| | |
|---|---|
| 00001 | Add |
| 00010 | Subtract |
| 00011 | Multiplication |
| 00100 | AND operation |
| 00101 | OR operation |
| 00110 | NOT operation |
| 00111 | XOR operation |
| 01000 | Rotate left |
| 01001 | Rotate Right |
| 01010 | Rotate left with carry |
| 01011 | Rotate Right with carry |
| 01100 | Shift left |
| 01101 | Shift right |
| 01110 | Increment |
| 01111 | Decrement |
| 10000 | Write enable |
| 10001 | Read enable |
| 10010 | Register WR enable |
| 10011 | Register RD enable |



Surbhi Sharma, Meena Kohli and Rachit Vaid, "VHDL Design & FPGA Implementation of 16-bit Microprocessor," *International Journal of Scientific and Technical Advancements*, Volume 2, Issue 4, pp. 287-292, 2016.

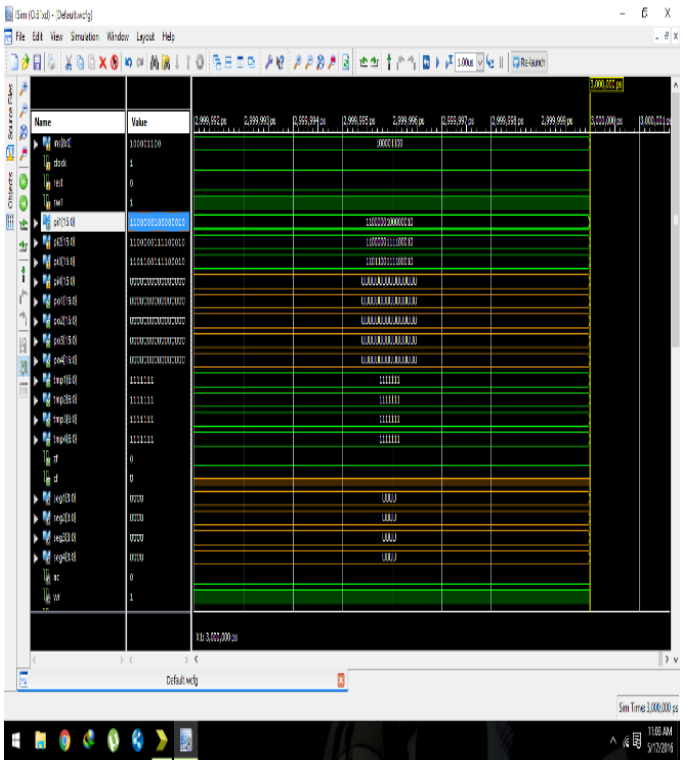

Fig. 2.11: Simulation waveform of main.

The block diagram in figure 2.12 shows the flow of signals as well as control between various blocks of designed microprocessor. The final output is fed to four 7 segment display to show 16-bit of data.

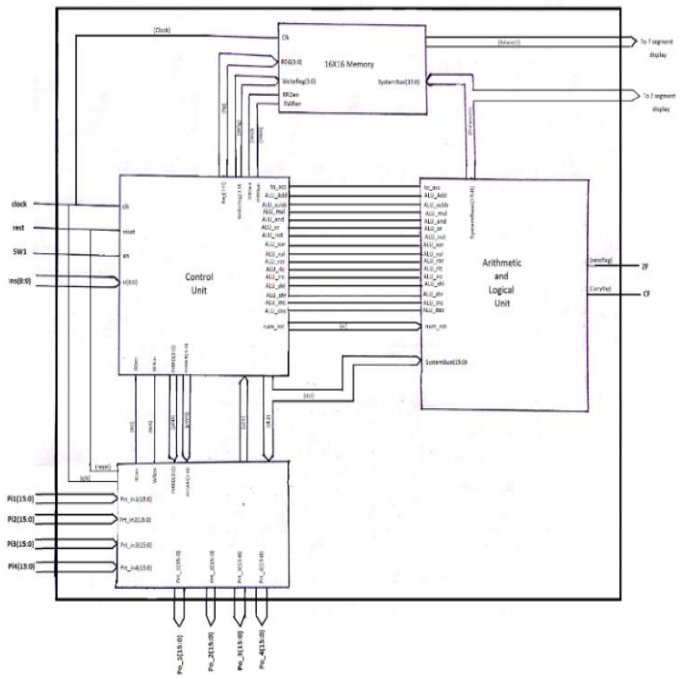


Fig. 2.12: Block diagram of microprocessor.

## III. CONCLUSION

A 16-bit microprocessor using Xilinx 13.2 is designed and simulated the instructions. The simulation shows that the processor executes for all the functional units described in the paper. There is a scope in this processor by increasing the no. functional units and Instructions with increased number of bits

Table 2: Microprocessor Design result.

| **Synthesis Results** | |
|---|---|
| 4 input LUTs used | 548 out of 7168 7% |
| IOs used | 170 |
| Number of Bonded IOBs | 170 out of 221 76% |
| Slice Flip Flops | 292 out of 7168 4% |
| Number of MULT18X18s | 1 out of 16 6% |
| Number of GCLKs | 1 out of 8 12% |
| **Timing Results** | |
| Minimum period | 10.312ns (96.974MHz) |
| Minimum input arrival time before clock | 11.046ns |
| Maximum output required time after clock | 10.916ns |
| Total REAL time to Xst completion | 5.00 secs |
| Total CPU time to Xst completion | 5.13 secs |

## REFERENCES

[1] Douglas L. perry "VHDL programming by Example" Mc Graw Hill, 4th Edition.
[2] Nazein M. Botros "HDL programming Fundamentals: VHDL & verilog"
[3] Shaily Goyal, "16 Bit Microprocessor Design and Simulation in VHDL", International Journal of Innovative Research & Development, Vol 3
[4] Martin Rosner, "Guide to Synthesis and Implementation Tools for VHDL Modeling and Design"
[5] Hall. Douglas V., "Microprocessors and Interfacing Programming and Hardware", McGraw-Hill International, Inc. 1992
[6] Xilinx User Manual, www.xilinx.com
[7] www.hdlprogramming.com
[8] Volnei A. Pedroni, "Circuit Design with VHDL"
[9] J Bhaskar, "A VHDL Primer", Third Edition
[10] Ramesh Gaonkar, "Microprocessor Architecture, Programming, and Applications with the 8085", Fifth Edition
[11] Nazeih M. Botros , "HDL Programming Fundamentals".
[12] Neeraj Jain, "VLSI Design and Optimized Implementation of MIPS RISC Processor Using Xilinx Tool", IJARCSEE, Volume 1, December 2012
[13] N. Gupta, P.Gupta, H.Bajpai , R.Singh , S.Saxena, "Analysis of 16 Bit Microprocessor Architecture on FPGA Using VHDL", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering Vol. 3, Issue 4, April 2014.



Surbhi Sharma, Meena Kohli and Rachit Vaid, "VHDL Design & FPGA Implementation of 16-bit Microprocessor," *International Journal of Scientific and Technical Advancements*, Volume 2, Issue 4, pp. 287-292, 2016.

ISSN: 2454-1532



Surbhi Sharma, Meena Kohli and Rachit Vaid, "VHDL Design & FPGA Implementation of 16-bit Microprocessor," *International Journal of Scientific and Technical Advancements*, Volume 2, Issue 4, pp. 287-292, 2016.