DESIGN OF 32 BIT ALU AND IMPLEMENTATION IN FPGA

ROHAN JOJU, Student, Dept of Electronis&Communication Engineering, RSET

Abstract: **This paper primarily deals with the construction of arithmetic Logic Unit (ALU) using Hardware Description Language (HDL) using Xilinx ISE 9.2i and implement them on Field Programmable Gate Arrays (FPGAs) to analyze the design parameters.. ALU of digital computers is an aspect of logic design with the objective of developing appropriate algorithms in order to achieve an efficient utilization of the available hardware. The hardware can only perform a relatively simple and primitive set of Boolean & arithmetic operations and are based on a hierarchy of operations that are built by using algorithms employing the hardware. Speed, power and utilization of ALU are the measures of the efficiency of an algorithm. In this paper, we have simulated and synthesized the various parameters of ALUs by using verilog on Xilinx ISE 9.2i and SPARTAN 6 FPGA board. Keywords: FPGA, ALU, XILINX**

I.INTRODUCTION

The design and implementation of FPGA based Arithmetic Logic Unit is of core significance in digital technologies as it is an integral part of central processing unit. ALU is capable of calculating the results of a wide variety of basic arithmetical and logical computations. The ALU takes, as input, the data to be operated on (called operands) and a code, from the control unit, indicating which operation to perform. The output is the result of the computation. Designed ALU will perform the following operations:

• Arithmetic operations
• Bitwise logic operations
- shifter operations

All the modules described in the design are coded using verilog which is a very useful tool with its degree of concurrency to cope with the parallelism of digital hardware. The top level module connects all the stages into a higher level at Register Transfer Logic (RTL). RTL describes the requirements of data and control units in terms of digital logic to execute the desired operations. Each instruction from the architecture's instruction set is defined in detail in the RTL Once identifying the individual approaches for input, output and other modules, the verilog descriptions are run through a verilog simulator and then is downloaded the design on FPGA board for verification.

II .OBJECTIVE

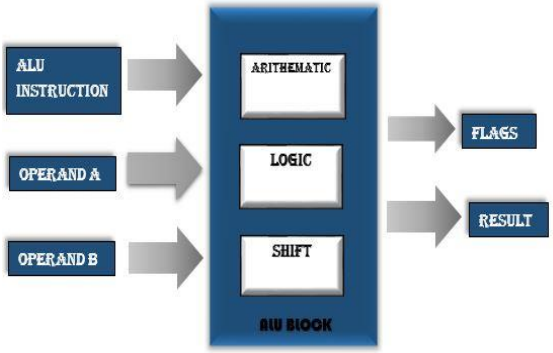
The goal of the project is to design and implement an Arithmetic and Logic Unit (ALU) for a 32 bit Microprocessor. The ALU is the functional block of the Microprocessor which handles the integer Arithmetic (Add, Subtract, Increment etc), Logic operations (AND, OR, XOR etc) and Shift operations (Shift Left, Shift Right etc) in the instruction set. An instruction of a RISC Microprocessor with a 32 bit instruction word has the format :

Opcode	Register Source 1	Register Source 2	Register Destination
8 bits	8 bits	8 bits	8 bits

The instruction is decoded and sent to the execution units, and the ALU would receive its instruction in the form:

ALU Instruction	Operand A	Operand B
6 bits	32 bits	32 bits

Depending on the Opcode, the instruction is handled by Arithmetic, Logic or Shifter sub units of the ALU and the output result of the calculation is generated.



The project team will design the sub units (Arithmetic, Logic and Shift) of the ALU using Verilog HDL, simulate and perform basic testing. The subunits will then be integrated to form the complete design, which will be extensively simulated and tested at the ALU level.

The project team will design and develop the ALU level simulation testbench and testcases. The testcases and expected results for comparison will be generated by developing a C model of the ALU.

After the design has been successfully simulated and tested, the project team will synthesize the design and map it on to the Xilinx FPGA to complete the hardware implementation. Simulation of the synthesized gate level design will also be done using the Xilinx Tools to make sure that the design behaves as expected before porting to FPGA. A bit file of the design will be generated and downloaded on to the Xilinx FPGA development board. The test cases generated using the C model will be applied to the FPGA to complete the hardware testing.

III.DESIGN OF TOP LEVEL (RTL) VERILOG MODULE OF 32- BIT ARITHMETIC LOGICAL UNIT (ALU)

High level design methodology allows managing the design complexity in a better way and reduces the design cycle. [10]. A high-level model makes the description and evaluation of the complex systems easier. RTL description specifies all the registers in a design, and the combinational logic between them. The registers are described either explicitly through component instantiation or implicitly through inference [3]. The combinational logic is described by logical equations, sequential control statements subprograms, or through concurrent statements [3]. Designing at a higher level of abstraction delivers the following benefits [10]:

- **Manages complexity:** Fewer lines of code improves productivity and reduces error.
- **Increases design reuse:** Implementation of independent designs as cell library & reuse in various models.
- **Improves verification:** Helps to run process faster.

IV. OPERATION OF ALU

There are two kinds of operation which an ALU can perform first part deals with arithmetic computations and is referred to as Arithmetic Unit. It is capable of addition, subtraction. The second part deals with the Gated results in the shape of AND, OR, XOR,XNOR, inverter, rotate, left shift and right shift, which is referred to as Logic Unit. The functions are controlled and executed by selecting operation or control bits.

Software Approach

The verilog software interface used in this design reduces the complexity and also provides a graphic presentation of the system. The key advantage of Verilog when used for required system to be described (modeled) and verified (simulated) before synthesis tools translate the design into real hardware (gates and wires). This software not only compiles the given verilog code but also produces waveform results

.

Hardware Approach

The verilog code which implies the hardware part of ALU is downloaded on FPGA processor.A final point is that when a verilog model is translated into the "gates and wires" that are mapped onto a programmable logic device i.e FPGA, and then it is the actual hardware being configured, rather than the verilog code being "executed" as if on some form of a processor chip.

V. IMPLEMENTATION OF ALU ON FPGA BOARD

The verilog coding of this paper design is compiled and simulated using Xilinx ISE and has been downloaded in FPGA using SpartanXC3S100E kit. The data is updated in the kit using 4 cycles of eight separate select inputs The function of FPGA is embedded on the kit along with PROM, LCD, LEDs and DIP switches. A Joint Test Action Group (JTAG) interface connects the FPGA chip with PROM and leads to PC through a serial interface.Since FPGA is a user programmable, therefore JTAG is of core significance. PROM has several postulates in the shape of data storage and debugging, permanent storage
of data, consistency of operation, low cost, high speed and compactness. PROM used in this design of ALUis“XC10S”, which is equipped with the inbuilt circuitry support and store complex functions.

VI. EXPECTED OUTCOME

The project team will deliver a working design of the ALU, and test vectors which can be used to verify its operation and the bit file for porting on to the FPGA.
The Verilog HDL source code can be used for further work in developing our own 32 bit RISC microprocessor, by adding a Control Unit and Memory Access and other blocks, which can be used as (i) an open source teaching aid for Computer Architecture courses and also as (ii) a basic processor framework on to which special purpose computational blocks can be added as required for other chip development projects.

VII. ACTION PLAN

The initial phase of the project will be dedicated to learning how to efficiently code datapath modules in Verilog HDL. The design responsibility for sub units of the ALU, the testbench design and the top level ALU block will be divided among the four team members. We plan to complete a good portion of the sub unit level design and the C model development in this semester. Next semester will be dedicated to top level simulation and porting on to the FPGA and testing.
The deliverable will be the design documents, source code in Verilog HDL, the test vectors and the bit file for porting on to the FPGA.

VII. TEST PLAN

A simulation testbench will be designed using Verilog HDL, with test vectors generated from a C model of the ALU and these will be used to simulate and verify the design. After porting the design to Xilinx FPGA, the test vectors from the C model will be applied to verify the correct operation of the hardware.

VIII. CHALLENGES FACED

FPGA implementation project may run into issues when the design size exceeds the capacity of the FPGA available. There could also be issues with the maximum frequency at which the FPGA based design can run, once the design is placed and routed. For this project, we do not anticipate any problems with these two issues as the design is small enough to fit and route the connections successfully on a Xilinx Spartan 6 development board.

ACKNOWLEDMENT

The work was supported in part by Mr. Karunakara P Menon & Ms Maleeha Abdul Azeez , Assistant Professor,Dept of Electronics & Communication Engineering, RSET.

REFERENCES

[1] Moeen Hassanalieragh, Alex Page, Tolga Soyata, Gaurav Sharma, Mehmet Aktas, Gonzalo Mateos Burak Kantarci, Silvana Andreescu, "Health Monitoring and Management Using Internet-of-Things (IoT) Sensing with Cloud-based Processing: Opportunities and Challenges", *IEEE conference on Servieces computing*, 2015.

[2] Ala Al-Fuqaha, Senior Member, IEEE, Mohsen Guizani, Fellow, IEEE, Mehdi Mohammadi, Student Member, IEEE,Mohammed Aledhari, Student Member, IEEE, and Moussa Ayyash, Senior Member, IEEE, "Internet of Things: A Survey on Enabling Technologies,Protocols,and Applications", *IEEE communication surveys & tutorials*, vol 17, No 4, Fourth quarter 2015.

[3] Alok Kulkarni, Sampada Sathe, "Healthcare applications of the Internet of Things", *IJCSIT*, Vol 5, 2014.

[4] Mattthias Thoma, Sonja Meyer, Klaus Sperner, Steffen Meissner, Torsten Braun, "On IoT service: survey , classification and enterprise integration", *IEEE International Conference on Green computing and communications,* 2012.

[5] Sherin Sebastian, Neethu Rachel Jacob, Yedu Manmadhan, Anand V. R. ,M. J. Jayashree,"Remote Patient Monitoring system", *International Journal of Distributed and Parallel Systems (IJDPS) Vol.3, No.5*, September 2012.

[6] K.M. Chaman Kumar Department of Computer Engineering Shree Rayeshwar Institute of Engineering and Information Technology , "Internet of Fitness Things - A Move Towards Quantified Health Concept, Sensor-Cloud Network, Protocols and a New Methodology for OSA Patients", *IEEE Recent Advances in Intelligent Computational Systems (RAICS),*10-12 December 2015 .

[7] Sreekanth K U PG Scholar,Department of Computer Science & Engineering Vidya Academy of Science & Technology (VAST) , Nitha K P ,Asst. Professor Department of Computer Science & Engineering Vidya Academy of Science & Technology (VAST), " A Study on Health care in Internet of Things", *International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169 Volume: 4 Issue: 2*

[8] e-Health Sensor Platform V2.0 for Arduino and Raspberry Pi [Biometric / Medical Applications], Available: https://www.cookinghacks.com/documentation/tutorials/ehealth-biometric-sensor-platform-arduino-raspberry-pi-medical

[9] Web Server, Available: http://www.fastwebhost.in/blog/what-is-

web-servers-different-types-of-web-servers

[10] Xiaomi Mi Band data extraction, analytics and Google Fit sync tools, Available:https://forum.xdadevelopers.com/general/accessories/xiaomi-mi-band-data-extraction-t3019156