IMPLEMENTATION OF ALU ON FPGA

SWASTIK RAO[1] SAKSHI MOHIT[2] SHARMEEN SHAIKH[3] RITIKA SHARMA[4]
PROF. YOGESH KARUNAKAR[5]

[1-5] Department of Electronics and Telecommunication
K.C. College of Engineering & Management Studies & Research, Kopri, Thane (E)-400 603, India.

Abstract*—We have implemented ALU on FPGA with the help of Eda playground. The objective of ALU in digital computers is to develop appropriate algorithms using arithmetic and logic design for efficient utilization of hardware. In this paper, we have used VHDL to program the ALU on Eda Playground.

I. INTRODUCTION

ALU is the execution unit of the central processing unit (CPU), which is the core component of all central processors. CPU is composed of an arithmetic-logic unit (ALU) and control circuits. The ALU carries out basic arithmetic and logic operations, and the control section determines the sequence of operations. The inputs to an ALU are the data where we have to perform operations. ALU is capable of calculating the results of a wide variety of basic arithmetical and logical computations [1]. They are called operands. They perform the necessary operation and the result is the output of the operation we have performed. The ALU takes, as input, the data to be operated on and a code, from the control unit, indicating which operation to perform. The output is the result of the computation. The designed ALU will be capable of performing Arithmetic operations and logic operations. These operations are common to all computer systems and thus are an essential part of computer architecture. All modules described in the design are coded using VHDL which is a very useful tool with its degree of concurrency to cope with the parallelism of digital hardware [2]. The ALU consists of input or output or even both. They also contain results of previously performed operations or the current operation and also registers. Registers are used to store, fetch and process data and that is being used by the Central Processing Unit.

II. LITERATURE SURVEY

Arithmetic logic unit is the core of any CPU that can be part of a programmable reversible computing device such as a quantum computer. The major concern for ALU design, using normal gates is heavy power consumption. The main reason for power consumption the normal irreversible gates. In order to ensure low power design constraint a new type of gates called reversible gates were introduced. In reversible gates the number of inputs is equal to the number of outputs and there is a one to one mapping between the inputs and outputs. The proposed ALU is expressed in VHDL hardware description language and is implemented on FPGA.

III. DESIGN SYSTEM

III.1. BLOCK DIAGRAM

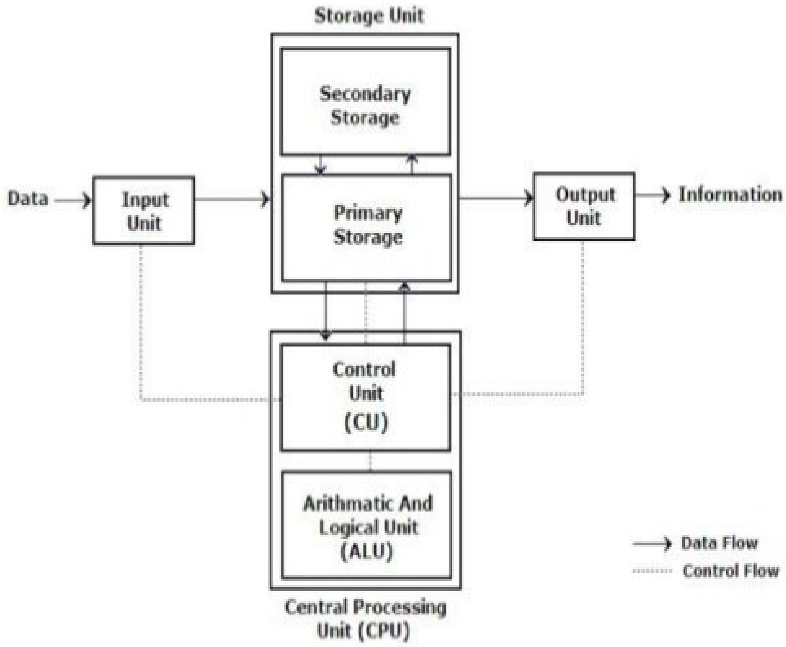


Figure 1. Block diagram of ALU

III.1.1. Input Unit:

Computers need to receive data and instruction in order to solve any problem. Therefore, we need to input the data and instructions into the computers. Keyboard is the one of the most commonly used input device. All the input devices perform the following functions.

a) Accept the data and instructions from the outside world.
b) Convert it to a form that the computer can understand.
c) Supply the converted data to the computer system for further processing.

III.1.2. Storage Unit:

The storage unit of the computer holds data and instructions that are entered through the input unit, before they are processed. The various storage devices of a computer system are divided into two categories.

a) Primary Storage: This memory is generally used to hold the program being currently executed in the computer, the data being received from the two input unit, the intermediate and final results of the program. The primary memory is temporary in nature. The data is lost, when the computer is switched off.

b) Secondary Storage**:** Secondary storage is used like an archive. It stores several programs, documents, data bases etc. The programs that you run on the computer are first transferred to the primary memory before it is actually run. Whenever the results are saved, again they get stored in the secondary memory.

III.1.3. Memory Size:

All digital computers use the binary system, i.e. 0's and 1's. Each character or a number is represented by an 8-bit code. The set of 8 bits is called a byte.

III.1.4. Output Unit:

The output unit of a computer provides the information and results of a computation to outside world. Printers, Visual Display Unit (VDU) are the commonly used output devices. Other commonly used output devices are Speaker, Headphone, Projector etc.

III.1.5. Arithmetic Logical Unit:

All calculations are performed in the Arithmetic Logic Unit (ALU) of the computer. It also does comparison and takes decision. The ALU can perform basic operations such as addition, subtraction, multiplication, division, etc and does logic operations.

III.1.6. Control Unit:

It controls all other units in the computer. The control unit instructs the input unit, where to store the data after receiving it from the user. It controls the flow of data and instructions from the storage unit to ALU. It also controls the flow of results from the ALU to the storage unit.

III.1.7.Central Processing Unit:

The Control Unit (CU) and Arithmetic Logic Unit (ALU) of the computer are together known as the Central Processing Unit (CPU). The CPU is like brain performs the following functions:

a) It performs all calculations.
b) It takes all decisions.
c) It controls all units of the computer.

IV. SOFTWARE APPROACH

The VHDL software interface used in this design reduces the complexity and also provides a graphic presentation of the system. The key advantage of VHDL when used for systems design is that it allows the behavior of the required system to be modelled and simulated before synthesis tools translate the design into real hardware [3]. Eda Playground compiles the given VHDL code and also produces waveform results.

V. DESIGN OF 8-BIT ALU

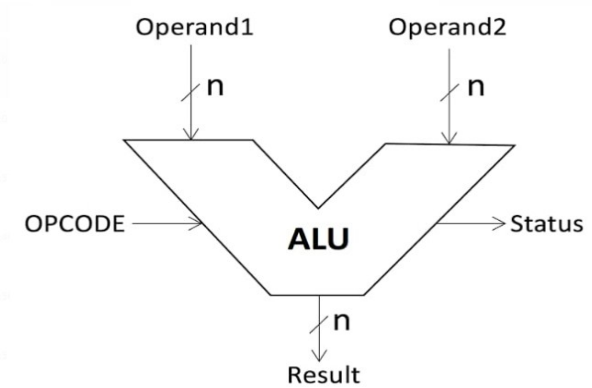


Figure 2. Arithmetic Logic Unit (ALU)

In this 8 bit ALU we have taken two 8 bit operand A (7:0) and B (7:0) and result is stored in ALU_Out (7:0). All operations are performed between these two operands. Result obtained is also 8 bit long. Since we are implementing ALU on FPGA opcodes are used.

The opcode determines what operations are to be performed by ALU. We have taken 4 bit opcode, so total 16 instruction should be created. Here we have used all 16 instructions.

All opcode as shown in Table 1.

Table 1. Opcode Table

OPCODE	**INSTRUCTION**
0000	Addition
0001	Subtraction
0010	Multiplication
0011	Division
0100	Logical shift left
0101	Logical shift right
0110	Rotate left
0111	Rotate right
1000	Logical AND
1001	Logical OR
1010	Logical XOR
1011	Logical NOR
1100	Logical NAND
1101	Logical XNOR

1110	Greater comparison
1111	Equal comparison

VI. TOOLS TO BE USED

VI.1 SOFTWARE

- Electronic design automation (EDA), also referred to as electronic computer-aided design (ECAD is a category of software tools for designing electronic systems such as integrated circuits and printed circuit boards.
- The tools work together in a design flow that chip designers use to design and analyze entire semiconductor chips. Since a modern semiconductor chip can have billions of components, EDA tools are essential for their design; this article in particular describes EDA specifically with respect to integrated circuits (ICs).
- EDA Playground is a free web application that allows users to edit, simulate (and view waveforms), synthesize, and share their HDL code. Its goal is to accelerate the learning of design and testbench development with easier code sharing and with simpler access to simulators and libraries.
- EDA for electronics has rapidly increased in importance with the continuous scaling of semiconductor technology. Some users are foundry operators, who operate the semiconductor fabrication facilities ("fabs") and additional individuals responsible for utilising the technology design-service companies who use EDA software to evaluate an incoming design for manufacturing readiness.
- EDA tools are also used for programming design functionality into FPGAs or field-programmable gate arrays, customisable integrated circuit designs.

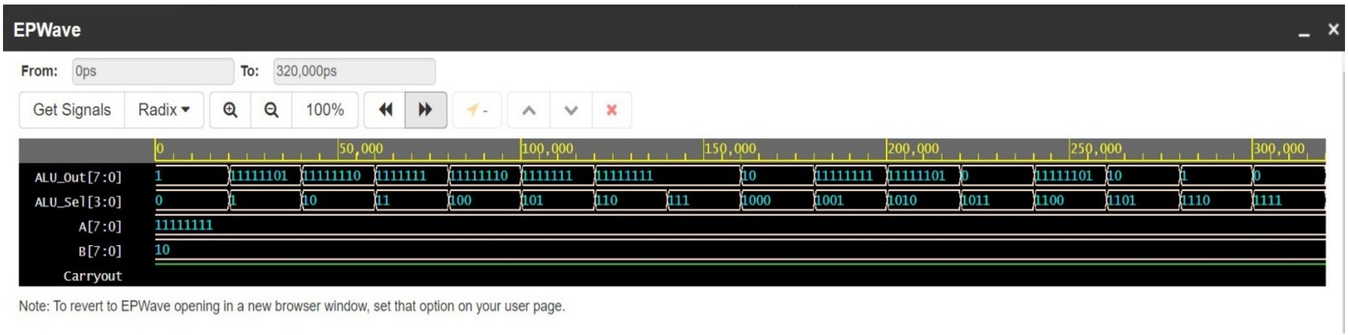
VII. SIMULATION

Example: A = 11111111, B = 00000010

Opcode=0001

Result=A-B

Result=11111101



VIII. CONCLUSION

This project helps us to know how to implement ALU on the build microcontroller on FPGA using Eda playground. After programming ALU we have to check and match the result with correct one. If the results are correct and the ALU performs the needed operations then this indicates that ALU is successfully built.

IX. FUTURE SCOPE

- In future works, we can try adding extra bits and showing over 16 bit ALU.
- We can try to minimize the propagation delay.
- In future we can also increase the number of operations as per need.

REFERENCES

[1] G. Karthik Reddy, "Low Power-Area Pass Transistor Logic Based ALU Design Using Low Power Full Adder Design, " IEEE Sponsored 9th (ISCO), 2015.

[2] Sakshi Samaiya and Anupreksha Jain, "A Review Article of FPGA ALU Unit Design Based on FGA" International Journal of Scientific Research & Engineering Trends, Volume 4, Issue 4, 2018.

[3] Shikha Khurana and Kanika Kaur, "Implementation of Alu Using Fpga," International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), Volume 1, Issue 2, 2012.

[4] Geetanjali and Nishant Tripathi,"VHDL Implementation of 32-Bit Arithmetic Logic Unit (ALU)," IJCSCE Special issue on "Emerging Trends in Engineering" ICETIE, 2012.

[5] B.Stephen Brown, V.Zvonko, "Fundamentals of digital logic with VHDL Design," 2nd Edition, Mc Graw Hill International Edition, 2005.

[6] S.Kaliamurthy, R.Muralidharan, "VHDL Design of FPGA Arithmetic Processor," International Conference on Engineering and ICT, 2007.