UR-147

An 8-bit Digital Computer Design & Implementation

Abstract

This project focuses on understanding modern computer architecture through the design and implementation of an 8-bit general-purpose computer using NI Multisim. Built on the von Neumann load-store architecture, the system incorporates an automatic Fetch-Decode-Execute (FDE) cycle and a hardwired instruction decoding control unit, with limited addressing modes. The computer, comprising only a CPU and memory, is capable of performing arithmetic, logical, data movement, and program flow control operations. For demonstration, an assembled program implementing Euclid's algorithm is used to find the greatest common divisor (GCD). This practical exploration offers valuable insights into modern computer architecture, showcasing how fundamental digital logic elements work together to solve computational problems.

Introduction

In today's digital world, understanding foundational computer architecture is essential for grasping how complex systems operate. This project explores these principles by designing and implementing an 8-bit general-purpose computer in NI Multisim. Using a von Neumann load-store architecture, it incorporates a Fetch-Decode-Execute (FDE) cycle and a hardwired control unit for instruction decoding, with limited addressing modes for simplicity.

The computer's architecture consists of a CPU and memory, yet it is fully capable of executing various operations, including arithmetic, logical, data movement, and program flow control. To demonstrate these capabilities, we implemented a program to compute the greatest common divisor (GCD) of two numbers using Euclid's algorithm. This example illustrates the machine's operational flow and key architectural concepts in practice. Through this project, we gain insights into how basic digital logic components interact to create the structured functionality of a digital computer, emphasizing the effectiveness of fundamental design in solving computational tasks.

Research Question(s)

1. How does the von Neumann architecture influence the computer's performance and design?
2. What are the challenges and specifications in designing an 8-bit computer?
3. How does an 8-bit computer perform program flow control, arithmetic and logic operations, and data movement?
4. How can we verify and validate the functionality of an 8-bit computer?

Materials and Methods

1. Planning and Design:
We started by studying the von Neumann architecture to understand how a basic computer processes instructions and data. We then designed the layout of our computer, identifying the key components such as the registers (Accumulator, Memory Buffer Register, Instruction Register), ALU (Arithmetic Logic Unit), memory, and control unit.

2. Simulation in NI Multisim:
Using NI Multisim®, a digital circuit simulation tool, we built each component of the computer. We created the registers, memory units, and ALU, and connected them using data and address buses. Each component was tested individually to ensure it worked correctly before integrating them into the full system.

3. Testing and Debugging:
We wrote simple programs to test the computer's ability to perform basic operations like loading data, adding numbers, and storing results. The Fetch-Decode-Execute cycle was monitored closely, and any errors were debugged by revisiting the design and simulation in NI Multisim.

4. Visualization:
We used block diagrams and flowcharts to represent the system's architecture and the flow of instructions. These visuals helped us understand and explain the process clearly. Screenshots from NI Multisim were also used to show the circuit layout and simulation results.

Results

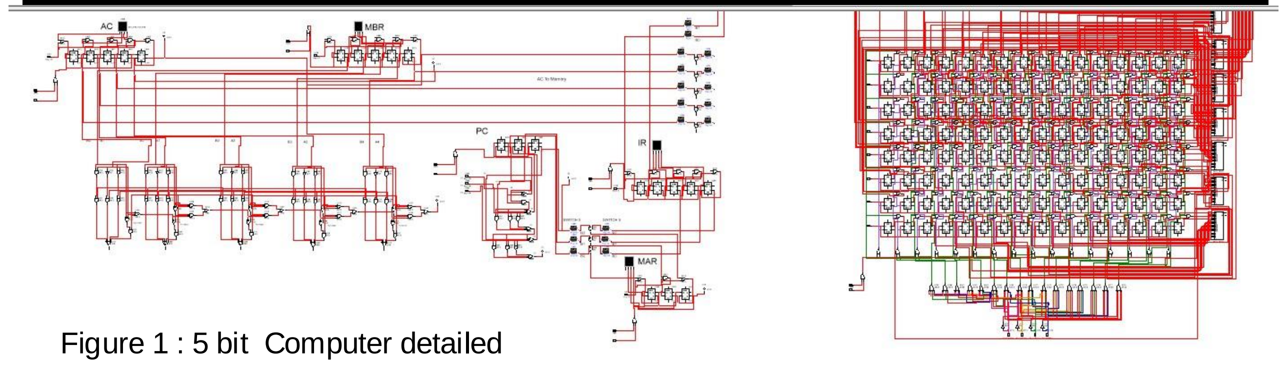

Figure 1 : 5 bit Computer detailed design

Highlights

1. 16x8b byte-addressable memory, 4-to-16 decoder, 16-to-1 multiplexer.

2. Addressing modes (direct and indirect) using dedicated/designed instruction types.

3. Learned about the von Neumann architecture and how computers execute instructions.

4. Built and simulated an 8-bit computer using NI Multisim.

5. Gained hands-on experience with digital logic design and understanding the Fetch- Decode-Execute cycle.

6. Successfully ran a sample program that demonstrated the computer's ability to perform arithmetic operations.

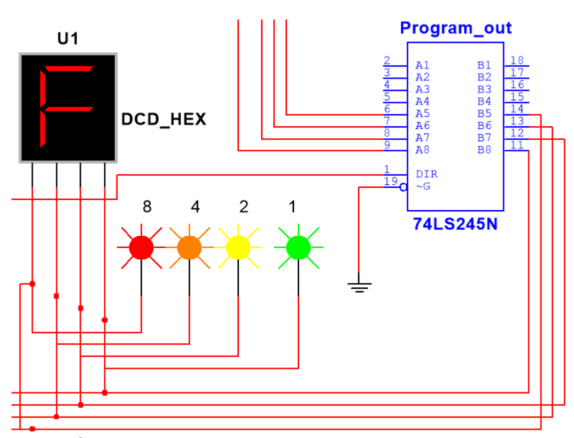

Figure 2 : Program Counter

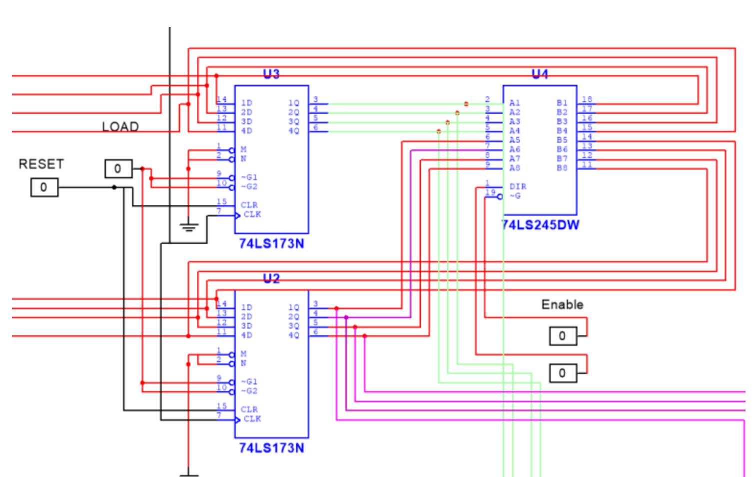

Figure 3 : A and B Register

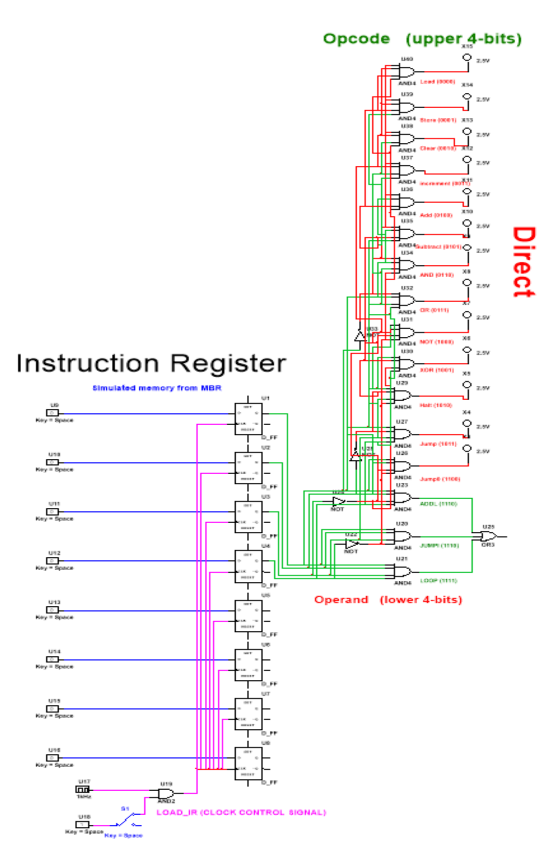

Figure 3 : Addressing Mode

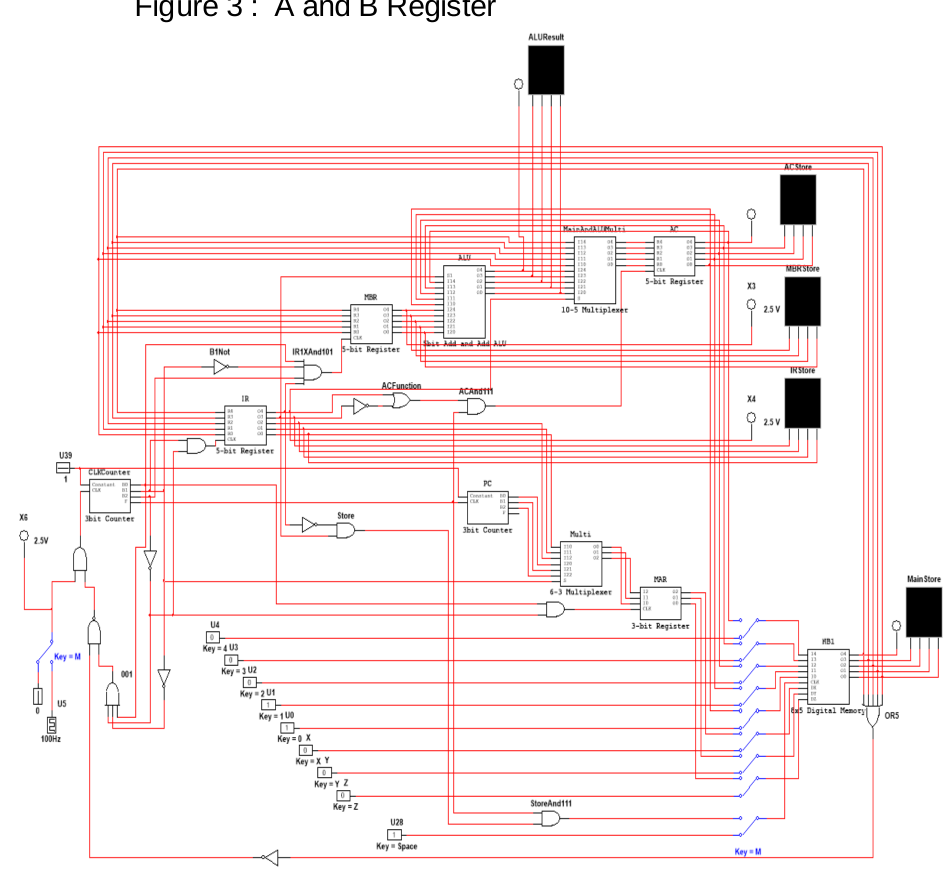

Figure 4 : Control Unit

Conclusions

In conclusion, our research has allowed us to successfully demonstrate design and implement an 8-bit computer using von Neumann architecture. Our system can perform 16 different general-purpose computations with data allocation making it a fully functional and complete representation of what we researched in computer organization and architecture.

Future Work

- Make a 16-bit version of this computer
- Increase the amount of instructions

Acknowledgments

We would like to thank **Professor Waqas Majeed** for his guidance throughout this project. His lectures and support were crucial in helping me understand the complexities of computer organization.

Experience

This project will help us both in our careers and in general as it will allow us to figure out our interest more clearly especially in computer engineering and digital system design. We will gain valuable hands-on experience while learning from our advisors and help us be better prepared for practical skills useful for future job opportunities and academic growth.

Future Career Plans

We plan to continue learning about computer systems and aim to work on more advanced projects in the future. After completing our degree, we hope to find a role in computer design or software development, and possibly further our education with advanced courses in this area.

Contact Information

Email us at following for more information:
jflore73@students.kennesaw.edu Asherar1@students.kennesaw.edu
wpitts6@students.kennesaw.edu blamsal@students.Kennesaw.edu

Adviser: wmajeed@kennesaw.edu

References

Essentials of Computer Organization and Architecture by Linda Null. Jones & Bartlett Pub (2018). 5th edition
https://www.multisim.com/
https://www.asmirvine.com/

KENNESAW STATE UNIVERSITY
COLLEGE OF COMPUTING AND SOFTWARE ENGINEERING

Author(s) Jesus Flores , Bishwash Lamsal , Adrian Sherard, William Pitts
Advisors(s) Waqas Majeed