Computer Architecture Lab Project Documentation

Name: Aavas Roy

Roll: 2207099

Group: B2

Course No: CSE 2114

Course Title: Computer Architecture Laboratory

Project Title: Developing a tiny 22-bit Computer

Date of submission: 03/08/2025

Submitted to

Dr. M.M.A. Hashem

Professor , Head of the Department

Department of Computer Science and Engineering

Khulna University of Engineering & Technology, Khulna

Md. Sakhawat Hossain

Lecturer

Department of Computer Science and Engineering

Khulna University of Engineering & Technology, Khulna

Objectives:

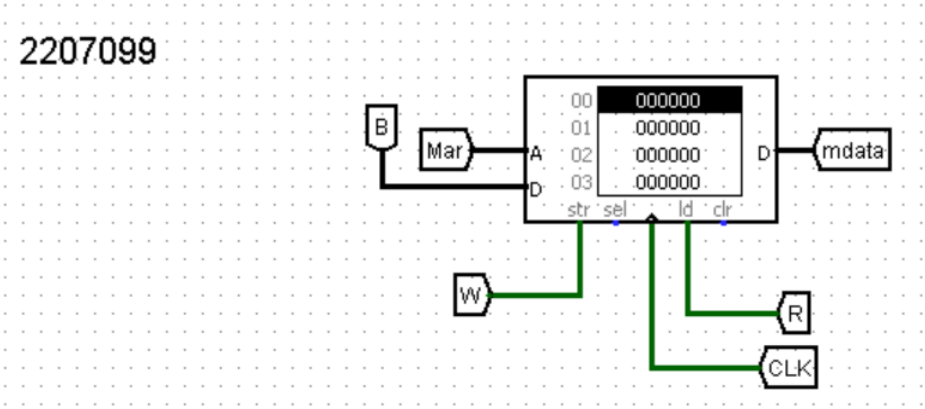
- To know about **Basic Working Principle of a CPU**.
- To learn about **Control Unit and Processing Unit** of a Computer.
- To know about components inside CPU.
- To know about **RAM , Registers , Counters** and its classifications.
- To design a CPU which is able to perform **basic operations**.

Introduction:

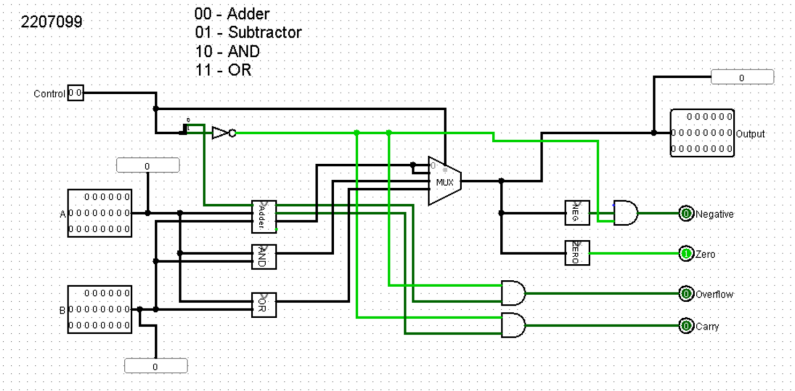
The **Central Processing Unit (CPU)** is the primary component of a computer that acts as its “control center.” The CPU, also referred to as the “central” or “main” processor, is a complex set of electronic circuitry that runs the machine's operating system and apps .Here the CPU is designed using a **22 bit RAM** with **address bit width of 5**. So the RAM has total of **2^5=32 addressable unit.** As the word width is 22 bits , so the **total size of the RAM is 2^5 x22 bits**. The ALU performs all the arithmetic and logical operation.The ALU operations are addition, subtraction,multiplication,OR as well as AND. Also the CPU can perform Load ,Store ,Branch ,Jump, Halt.

Here, only 4 bits are required to represent each operation and 5 bits are required to represent address. The least significant 5 bits(0-4) of instruction are used as **address** and 16-19 bits are used as **opcode** and the other bits are used as don’t care. But all the 22 bits are used as data.

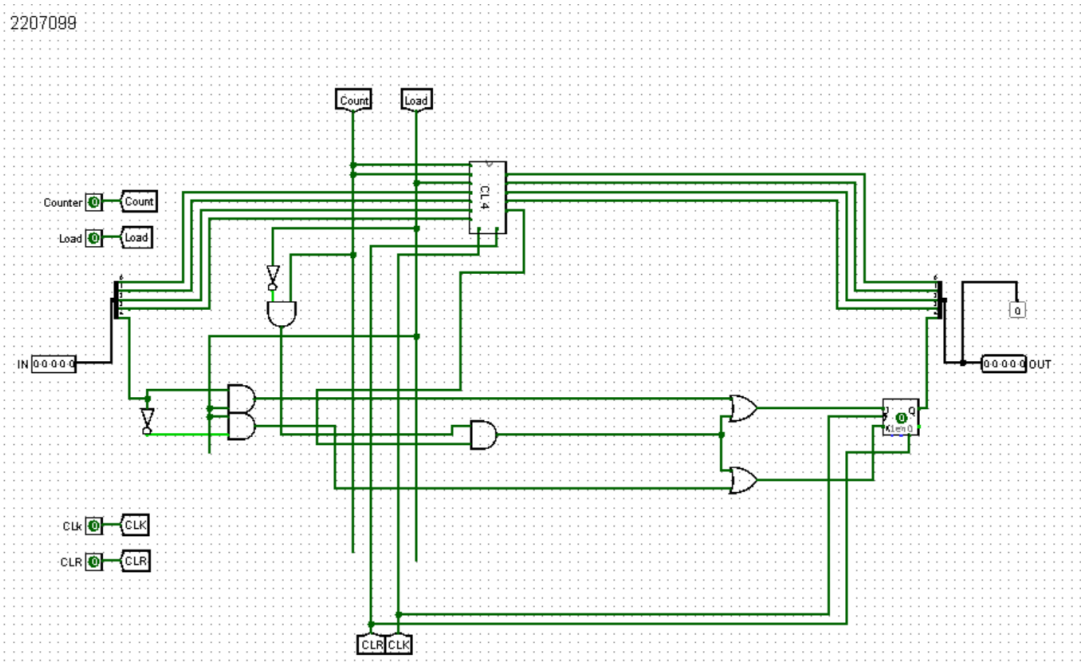
RAM: Random-access memory(RAM) is a form of electronic computer memory that can be read and changed in any order, typically used to store working data and machine code. A random-access memory device allows data items to be read or written in almost the same amount of time irrespective of the physical location of data inside the memory, in contrast with other direct-access data storage media.



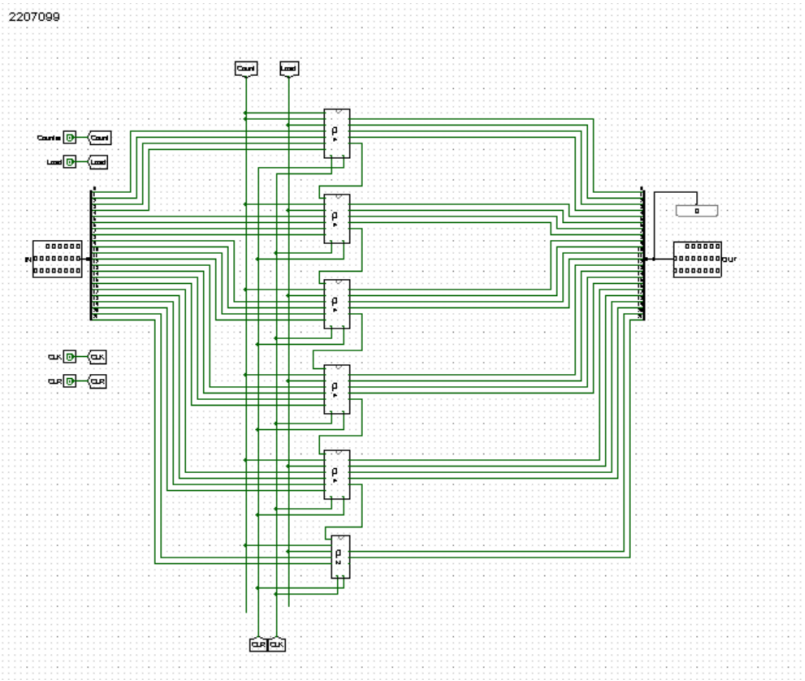
ALU: An ALU is a circuit component of a computer's central processing unit (CPU) responsible for performing **arithmetic and logical operations**. It performs tasks like addition, subtraction, bitwise operations, and comparisons. The ALU receives inputs from registers and produces outputs based on the instruction it receives. It executes calculations using binary numbers and manipulates them using logic gates. By combining and manipulating these inputs, the ALU generates the desired output, which is then stored back in the registers for further processing.



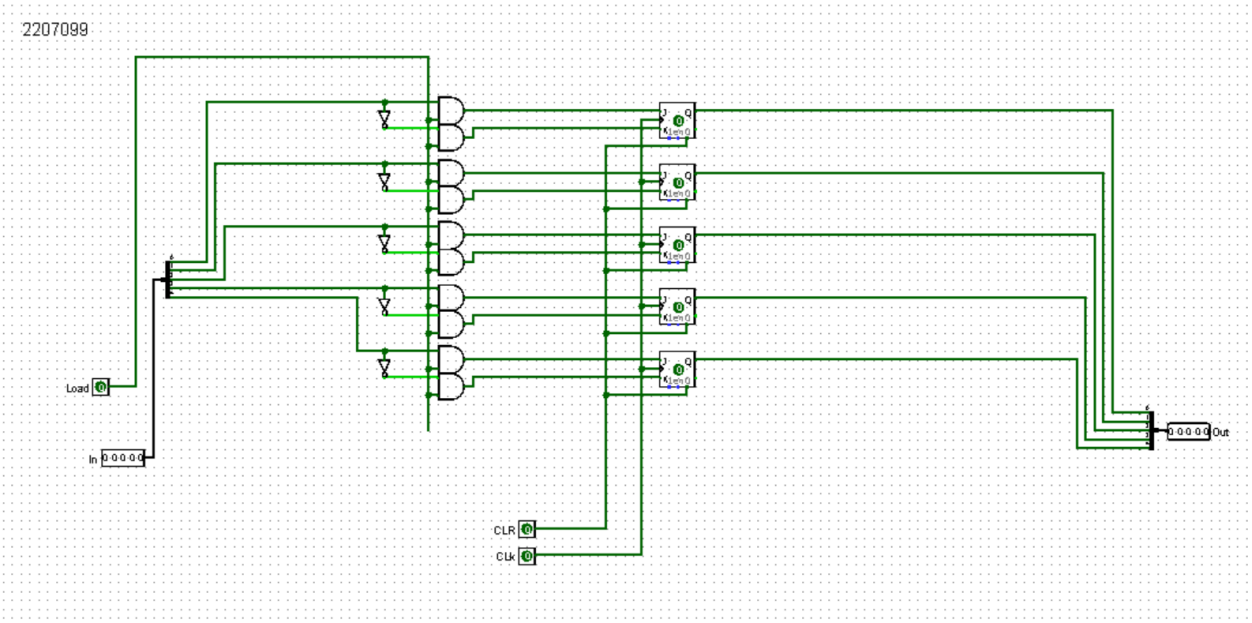
PC: The Program Counter (PC) is a crucial component of a computer's CPU, responsible for keeping track of the memory address of the next instruction to be executed in a program. By incrementing with each instruction cycle, the Program Counter ensures that the CPU processes instructions in the correct sequence, making it essential for program execution and overall system efficiency.



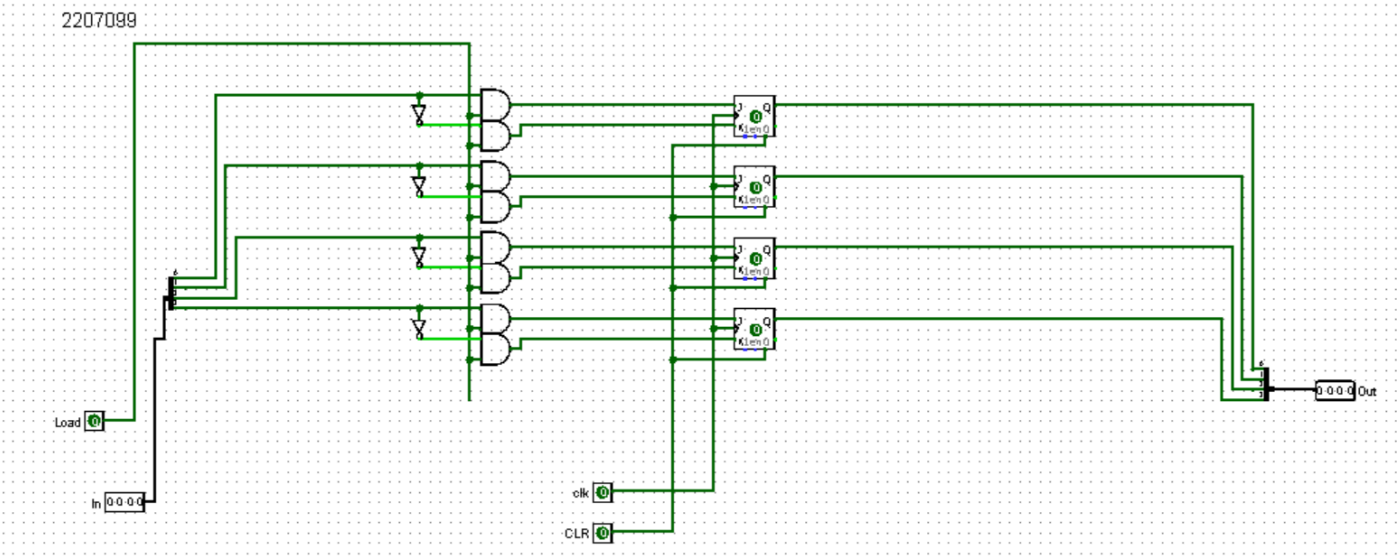
MBR: Memory Buffer Register is a register in a computer's CPU that temporarily holds data being transferred, it acts as a buffer between the CPU and memory, ensuring smooth and efficient data flow during read and write operations.



MAR: In a computer, the memory address register (MAR) is the CPU register that either stores the memory address from which data will be fetched to the CPU registers, or the address to which data will be sent and stored via system bus.



IR: Instruction register is an element in the central processing unit (CPU) of a computer or other device that holds programming instructions that will be executed at the beginning of the next clock cycle as instructed by other parts.



Accumulator: In a computer's central processing unit (CPU), the **accumulator** is a register in which intermediate arithmetic logic unit results are stored.

Control Unit: The control unit (CU) is a component of a computer's central processing unit (CPU) that directs the operation of the processor. A CU typically uses a binary decoder to convert coded instructions into timing and control signals that direct the operation of the other units (memory, arithmetic logic unit and input and output devices, etc.).

Sequence Register: A register in the control unit of the CPU that is used to keep track of the address of the current or next instruction. Typically, the program counter is advanced to the next instruction, and then the current instruction is executed. A sequence register is made to generate t0,t1,t2 and t3. It makes arranging logic easy by studying about clock pulse.

Opcode List: The following table describes the opcode of instructions.

INSTRUCTION	OPCODE(HEX)	INSTRUCTION	OPCODE(HEX)
AND	0	LDA	6
ADD	1	HLT	7
STO	2	OR	8
ISZ	3	SUB	9
BSB	4	INC	A
BUN	5	DEC	B

MBR Representation: The following table describes the bit allocation of MBR.

NONE	OPCODE	NONE	ADDRESS
20-21	16-19	6-15	0-5

Sample Program with Instruction details:

- **06001c :** Loads Data of 0x1c location in Accumulator (Fetch)
- **00001d** : Loads Data of 0x1d location in MBR and perform AND (ALU operation)
- **02001e :** Stores value of MBR in 0x1e location (Write)
- **040005** : Jumps to location 0x05 and stores return address there (BSB)
- **01000c** : Adds value of location 0x0c and Accumulator (ADD)
- **070000** : Stops performing any operation though clock pulse remains on (HLT)

Discussion:

The designed minicomputer system comprises several interconnected components, including the CPU, memory, and control units. The CPU is responsible for fetching instructions from memory, decoding them, and executing the corresponding operations. The control unit coordinates the flow of data and instructions within the system, while the memory stores both program instructions and data. In the CPU circuit, the address part of the instruction is fetched from the memory buffer register (MBR) and transferred to the memory address register (MAR). The instruction execution proceeds sequentially, with each instruction being decoded and executed accordingly. The ALU performs arithmetic and logical operations, and the accumulator registers store intermediate results. The implemented instructions include basic arithmetic operations such as addition and subtraction, as well as load, store, jump, and halt instructions. These instructions allow the minicomputer to perform simple computational tasks and 7 manipulate data stored in memory. By simulating clock pulses and providing input data, the minicomputer executes the instructions and produces the expected output.

Conclusion:

In conclusion, the project successfully achieved its objectives of designing and implementing a minimal computer system using Logisim software. Through this project, the fundamental concepts of computer architecture and organization were explored, including the operation of CPUs, memory units, and control mechanisms. The hands-on experience gained from building the minicomputer enhanced understanding of data communication, instruction execution, and memory management within a computer system. While the implemented minicomputer supports a limited set of instructions, it serves as a foundational platform for further exploration and experimentation in computer science and engineering.