RIPHAH
INTERNATIONAL
UNIVERSITY

Computer Architecture

LAB 5

Fasiha Tariq - 53289

BS-CS 4

Submitted to:
Ma’am Ayesha Ikram

Date: March 13th,2025

1. Write a detailed explanation of how the Fetch-Decode-Execute cycle works.

Fetch Phase:

- The PC provides the MAR with the instruction's address.
- The MAR accesses the main memory and fetches the instruction stored at that address.
- The instruction is transferred to the MDR for temporary storage.
- The PC is incremented and jumps to the next instruction.
- The instruction is being sent to the Instruction Register (IR) for decoding.

Decode Phase:

- IR sends the instruction to the control unit.
- The Control Unit analyzes the instruction to determine the operation, location of data, and any additional information needed.
- The CPU retrieves extra information (operands) from memory or registers as needed.

Execute Phase:

- Performs arithmetic or logical operations.
- Transferring information between registers or memory locations.
- Modifying the program’s control flow (e.g., jumping to another instruction).
- Storing the result in the register or memory.

Once an instruction is executed, the PC is updated to the next instruction. The FDE cycle continues indefinitely.

---

2. Use a simple instruction as an example and describe each step.

Adding Two Numbers: ADD R1, R2, R3

Add the values in registers R2 and R3 and store the result in R1.

- **Fetch:** The instruction ADD R1, R2, R3 is retrieved from memory into the IR.
- **Decode:** The CU identifies that this is an ADD operation and that operands R2 and R3 are needed.
- **Execute:** The ALU adds the values in R2 and R3, and the result is stored in R1.

---

3. Explain the role of PC, AR, IR, AC, and DR in your own words.

Role of PC, AR, IR, AC, and DR

- Program Counter (PC): Stores the address of the next instruction and increments automatically after each instruction fetch.
- Address Register (AR): Holds the memory address the CPU needs to access (for fetching an instruction).
- Instruction Register (IR): Stores fetched instruction and is being decoded and executed.
- Accumulator (AC): Used for arithmetic and logic operations.
- Data Register (DR): Holds processed data temporarily.

4. What is the function of the Arithmetic Logic Unit (ALU) in CPU operations? How does ALU interact with registers and memory?

The **Arithmetic Logic Unit** performs mathematical and logical operations within the CPU.

Function of ALU:

- Arithmetic Operations (Addition, Subtraction, Multiplication, Division)
- Logical Operations (AND, OR, NOT, XOR)
- Comparison Operations (Greater than, Less than, Equal to)
- Bitwise Operations (Shift left, Shift right, and rotate)
- Performs operations on data stored in registers or memory.

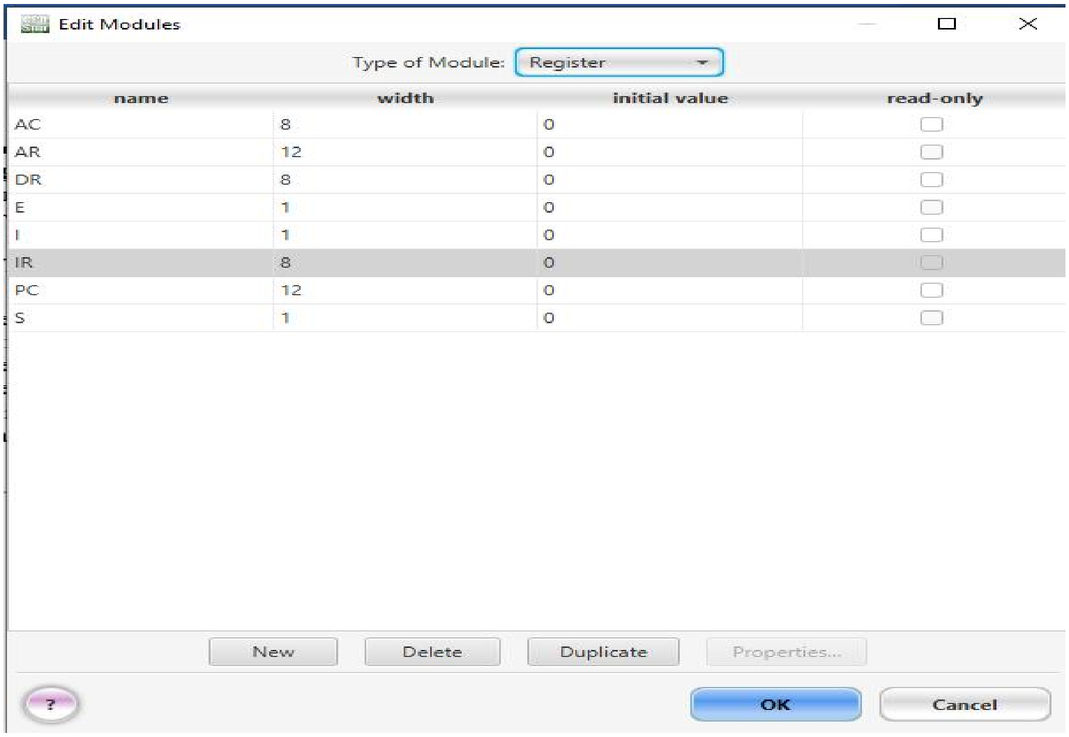
ALU plays a crucial role in the FDE instruction cycle and performs calculations in the CPU.

How ALU interacts with registers and memory:

- ALU fetches operands from registers (e.g., Accumulator Registers) and fetches data from memory through Load operations.
- ALU stores results in registers and stores data in memory through Store operations.
- ALU transfers data between registers.
- ALU generates memory addresses for data access.

ALU interacts with the register and memory to perform operations, transfer data and access memory location.

5. Create a new base machine and change the bit width of a register (e.g., make AC 8-bit instead of 16-bit)



Type of Module: Register

name	width	initial value	read-only
AC	8	0	□
AR	12	0	□
DR	8	0	□
E	1	0	□
I	1	0	□
IR	8	0	□
PC	12	0	□
S	1	0	□