CSE221 (UG2023)

Logic Design and Computer Organization

Phase 2

Name	ID
Omar Ahmed ElKhadragy	**23p0085**
Youssef George Nabil	**23p0227**
Youssef Ahmed Farouk	**23p0054**
Abdelrahman Mohamed Bahig	**23p0390**
Mohammed Osama Mondy	**23p0374**

1. Datapath

1.1 Definition and Functionality

The datapath is the central component of the CPU that performs all arithmetic, logical, and data transfer operations. It connects different units of the processor, enabling the execution of instructions.

1.2 Components

- **Registers**: Temporary storage locations for data, typically implemented as part of the register file.
- **Arithmetic Logic Unit (ALU)**: Executes arithmetic and logical operations such as addition, subtraction, AND, OR, etc.
- **Multiplexers (MUX)**: Used to select inputs or outputs based on control signals.
- **Data Paths**: Lines that connect different components, transferring data between them.
- **Shift Units**: Handle shift operations such as logical or arithmetic shifts, often required by certain instructions.

1.3 Operation

- The datapath takes inputs from the register file and other units like instruction memory.
- It processes these inputs using the ALU and other components.
- Results are written back to the register file or sent to the data memory based on the instruction type.

2. Control Unit

2.1 Definition and Functionality

The control unit manages the flow of data within the CPU. It generates signals that coordinate the actions of the datapath, instruction memory, and data memory.

2.2 Components

- **Instruction Decoder**: Interprets the opcode from the instruction and determines the type of operation.
- **Control Signals**: Signals such as ALUOp, MemRead, MemWrite, RegWrite, and others that guide the execution process.
- **Finite State Machine (FSM)**: Ensures proper sequencing of operations.

2.3 Operation

- The control unit decodes the fetched instruction and determines the necessary operations.
- It activates the appropriate control signals to direct data through the datapath and memory.
- Based on the instruction type (R, I, or J), it adjusts signals to ensure proper execution.

3. Instruction Memory

3.1 Definition and Functionality

Instruction memory stores the instructions to be executed by the CPU. It is read-only during program execution.

3.2 Components

- **Instruction Address**: Provides the address of the instruction to be fetched.
- **Instruction Register**: Holds the fetched instruction for decoding.

3.3 Operation

- The Program Counter (PC) provides the address of the next instruction to be fetched.
- The instruction memory retrieves the instruction at the specified address and passes it to the control unit for decoding.

4. Data Memory

4.1 Definition and Functionality

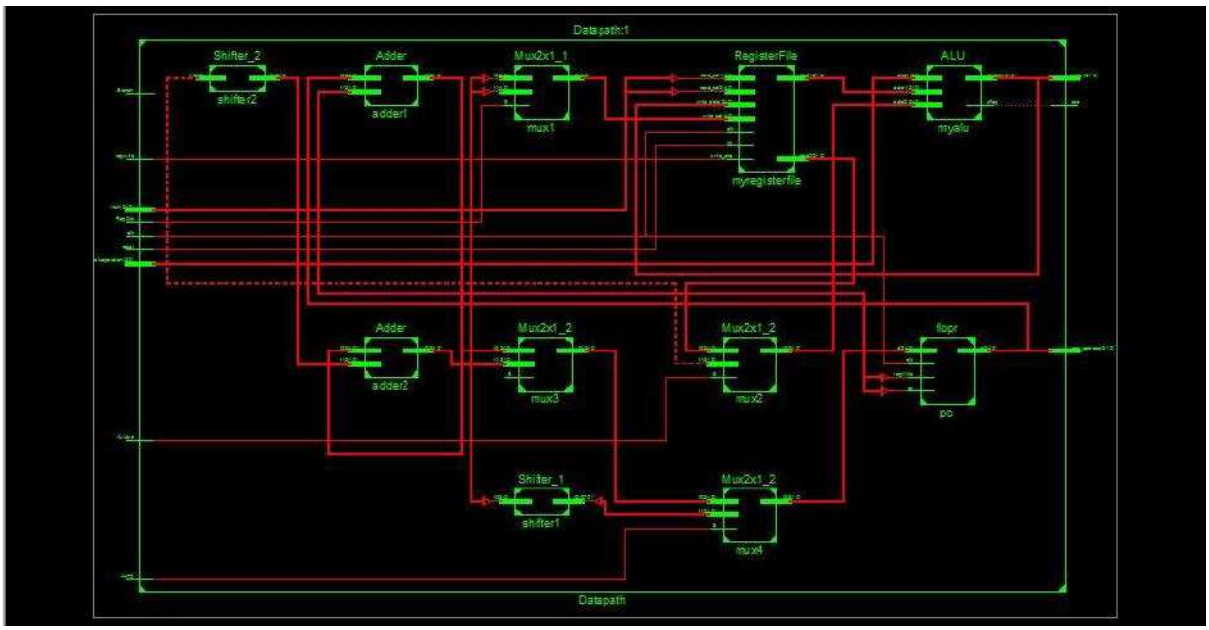
Data memory stores data used during program execution. It is accessible for both reading and writing by the CPU.

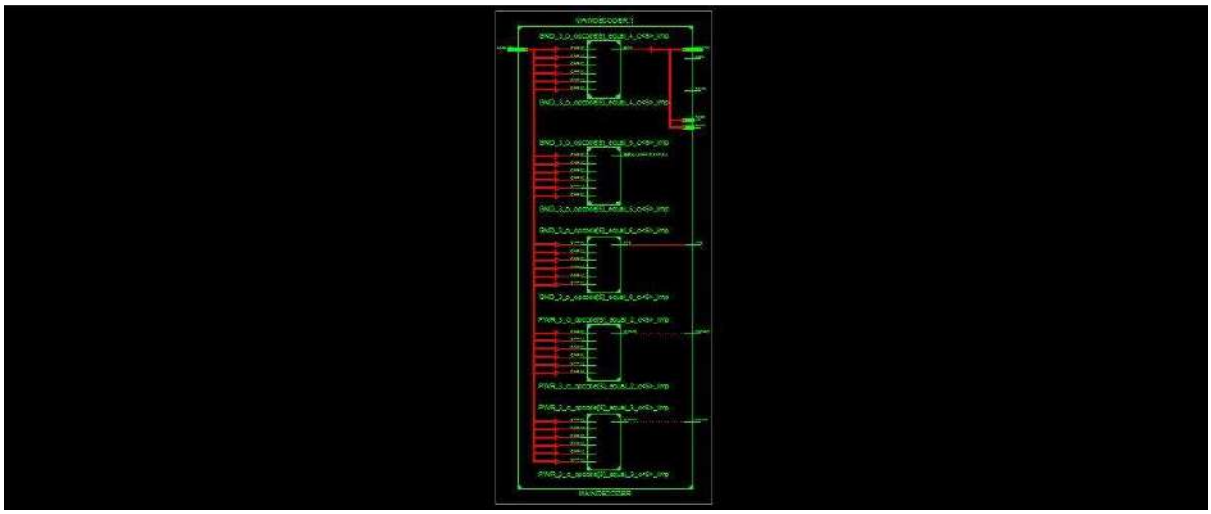
4.2 Components

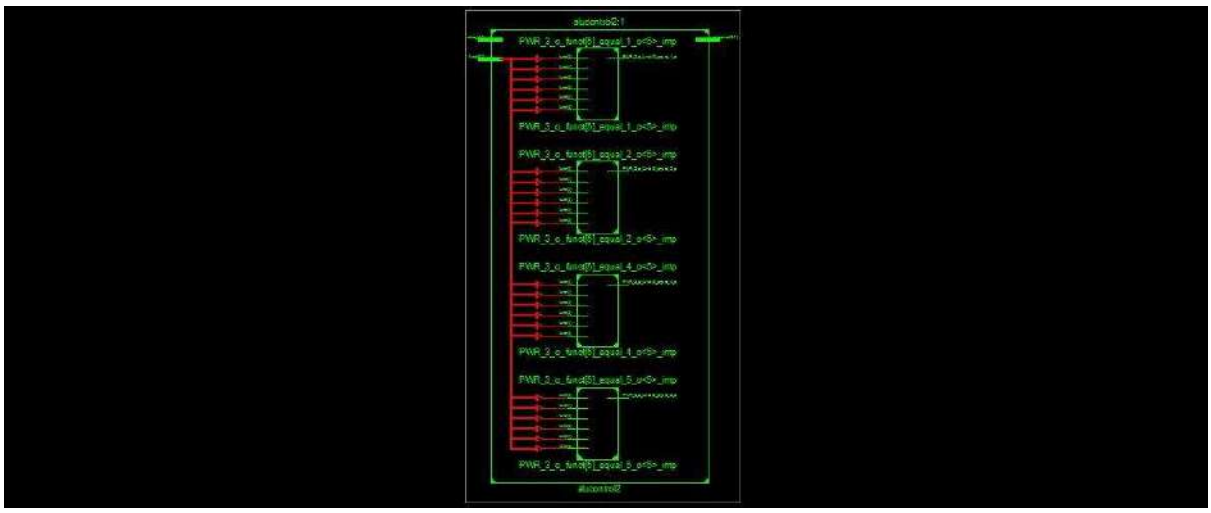
- **Address Lines**: Specify the memory location to access.
- **Data Lines**: Transfer data to and from memory.
- **Read/Write Control**: Determines whether the memory is being read or written.

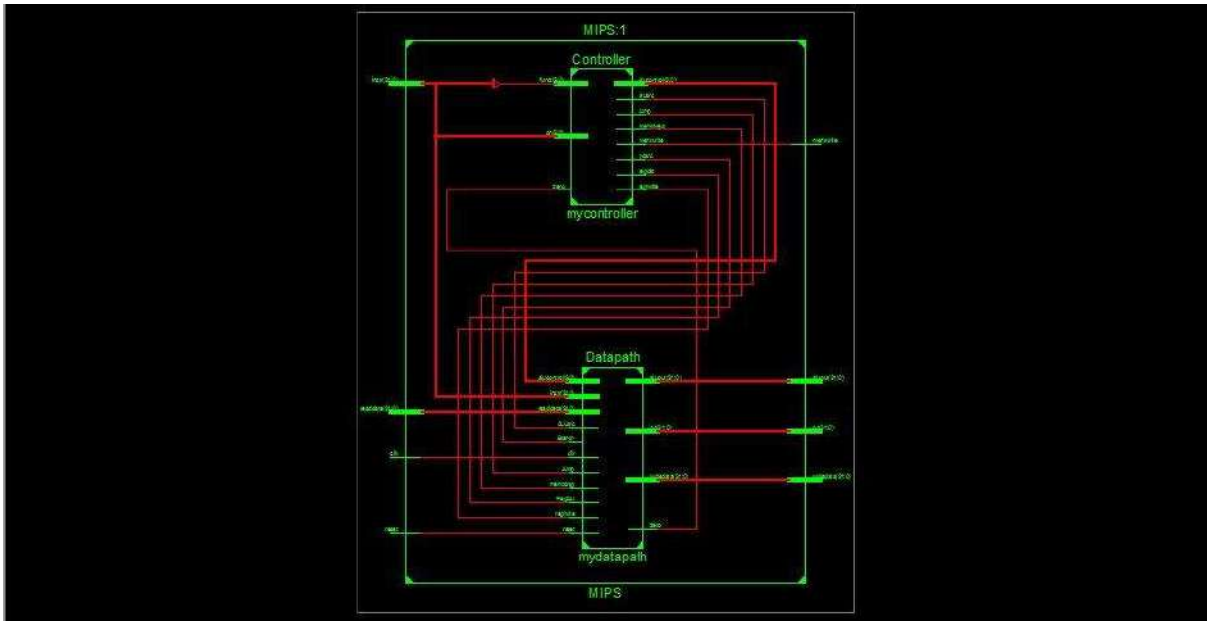
4.3 Operation

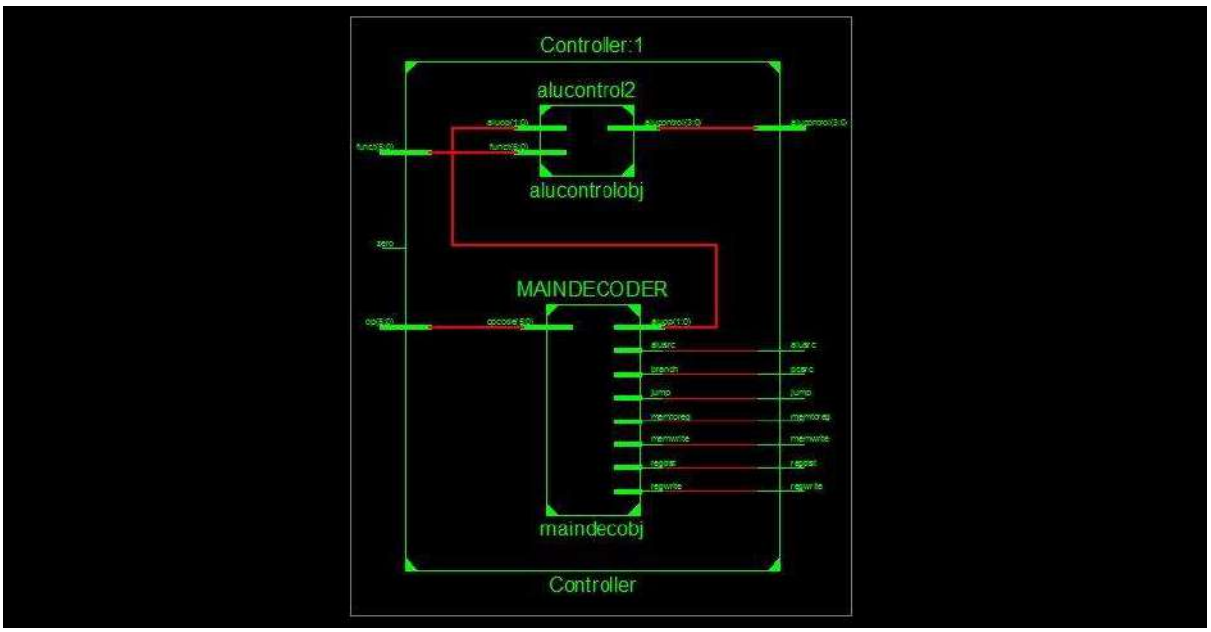
- During a load instruction (e.g., `lw`), the CPU sends a read signal along with the memory address, retrieving data to store in a register.
- During a store instruction (e.g., `sw`), the CPU sends a write signal along with the memory address and data to store.

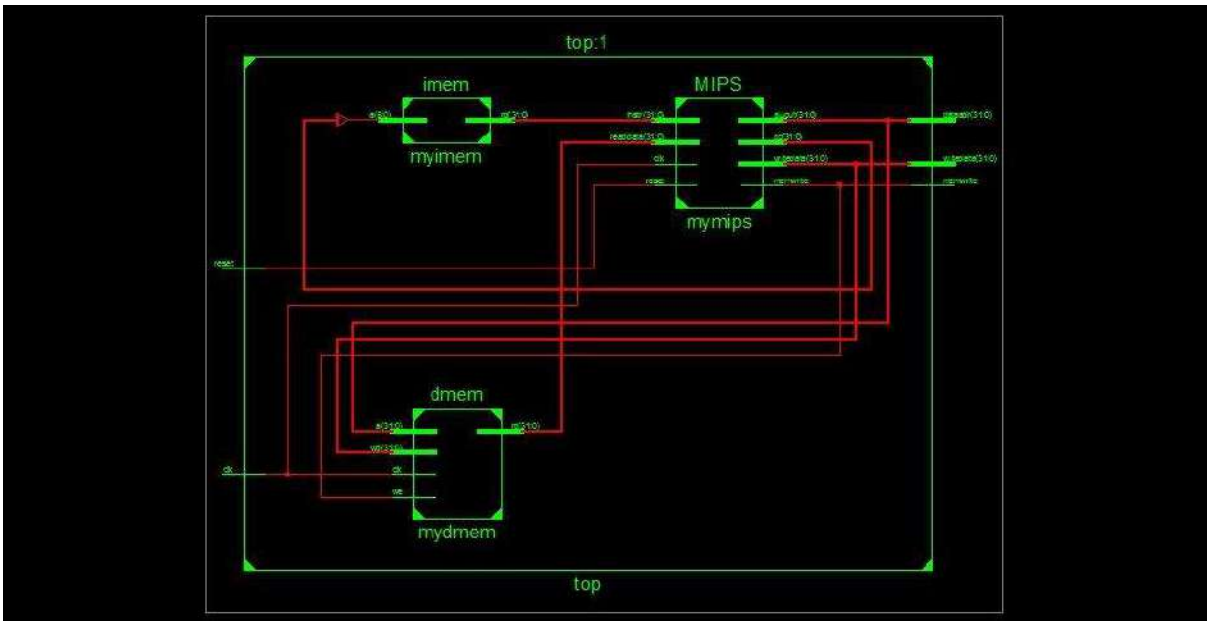

Datapath:1
Shifter_2
shifter2
Adder
adder1
Mux2x1_1
mux1
RegisterFile
myregisterfile
ALU
myalu
Adder
adder2
Mux2x1_2
mux3
Mux2x1_2
mux2
flopr
pc
Shifter_1
shifter1
Mux2x1_2
mux4
Datapath






MIPS:1
Controller
mycontroller
Datapath
mydatapath
MIPS


Controller:1
alucontrol2
alucontrolobj
MAINDECODER
maindecobj
Controller


top:1
imem
myimem
MIPS
mymips
dmem
mydmem
top

Ain Shams University
Faculty Of Engineering
CSE 221