19/4/23

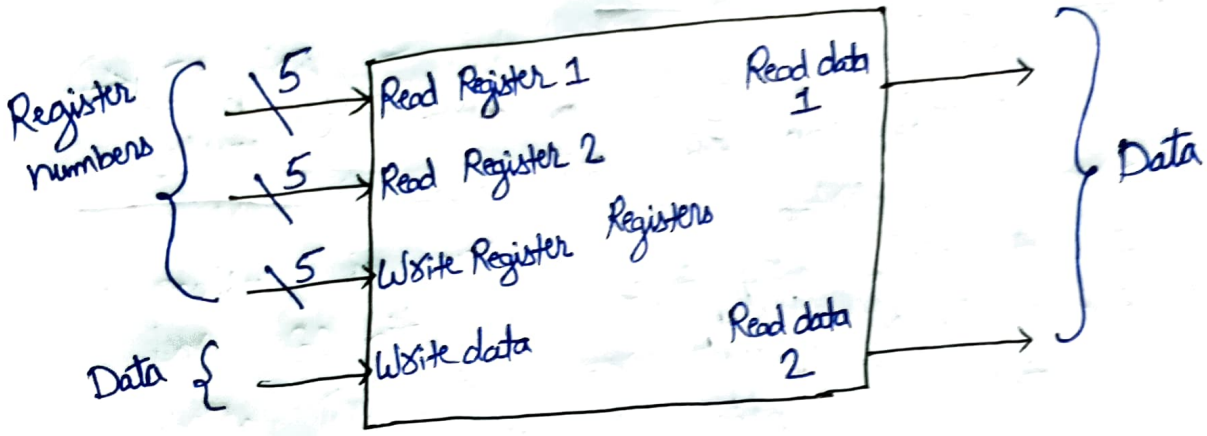
Assignment - 6

Q1) Write short notes on MIPS register file.

Soln

In MIPS the register file contains 32 registers each having size 32 bits.

→ The block diagram of a register file is shown in figure below.



→ In MIPS the register file have 5 i/p port, 4 i/p port representing for data input and 1 i/p port representing for control signal input.

→ In 4 data inputs 3 representing for the address of register and one representing the data to be written in the register file. Since MIPS has 32 registers, the address inputs are 5 bit each.

→ The size of the write data port is 32 bit each.

→ The control i/p port is a 1 bit port which tells when the register will be written.

→ There are two output ports, each of 32 bits for reading the data from the register file.

Q2.] Write short notes on instruction memory and data memory.

Soln Instruction memory:-

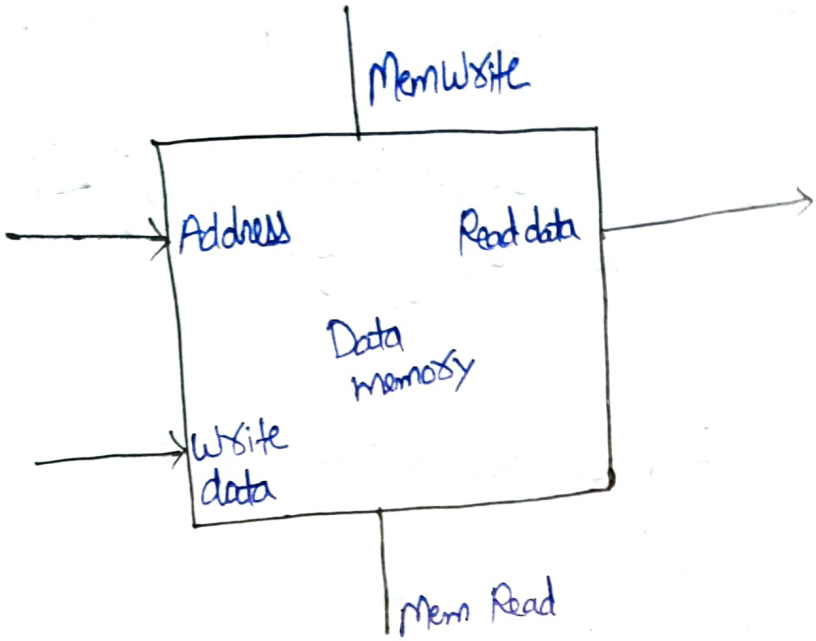
→ It is a memory where instructions are stored. It has one i/p port having size 32 bits, it is an address port which takes the address of instruction to be executed. It contains 1 o/p ports having size 32 bits and it provides the 32 bit instructions as per the 32 bit instruction address.

Data memory:-

→ It is a data path element to which stores the data for the access by the ALU for the instruction execution.

→ The data memory and the instruction memory in some architecture are one unit, but some other architecture they may be seperated.

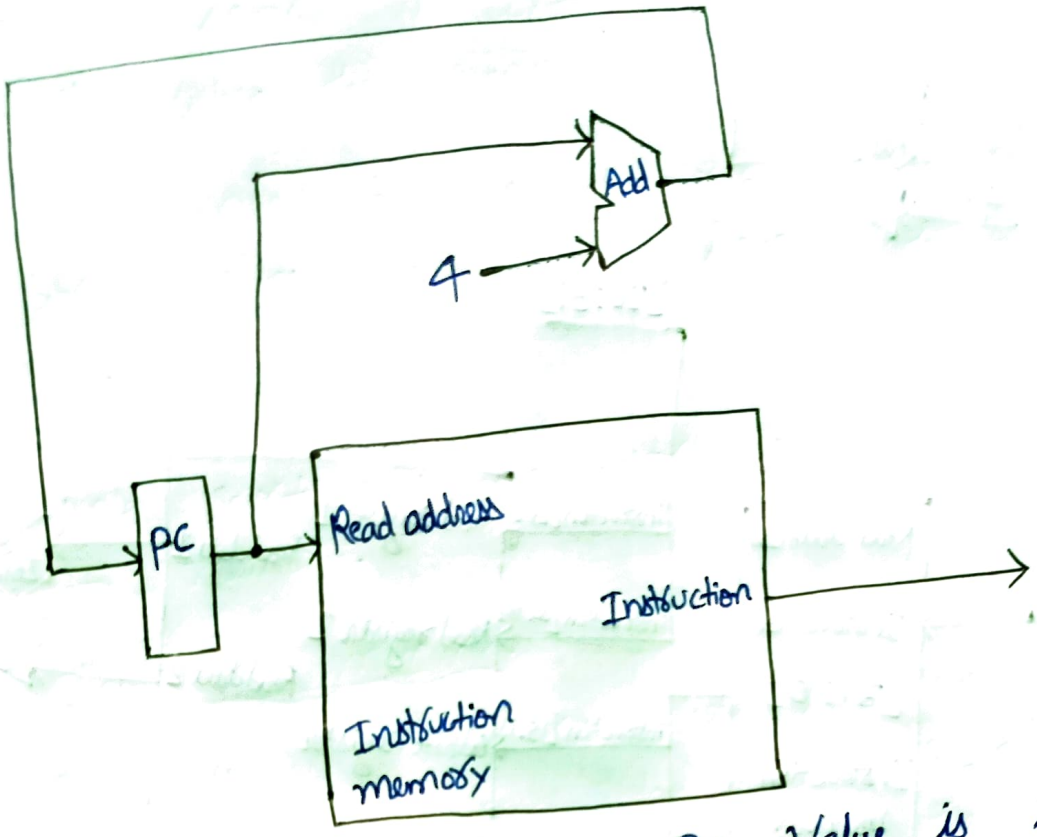
→ The figure for the data memory is shown below-



→ It has 4 i/p ports, one representing the address i/p which is 5 bit, two representing control i/p one bit each, one for read control, another is for write control. 1 i/p port for 32 bit that text the data to be written. The data memory has also 1 o/p port having size 32 bit representing the data to be read from the data memory.

Q3> Discuss the datapath design showing the program counter fetching the instruction sequentially.

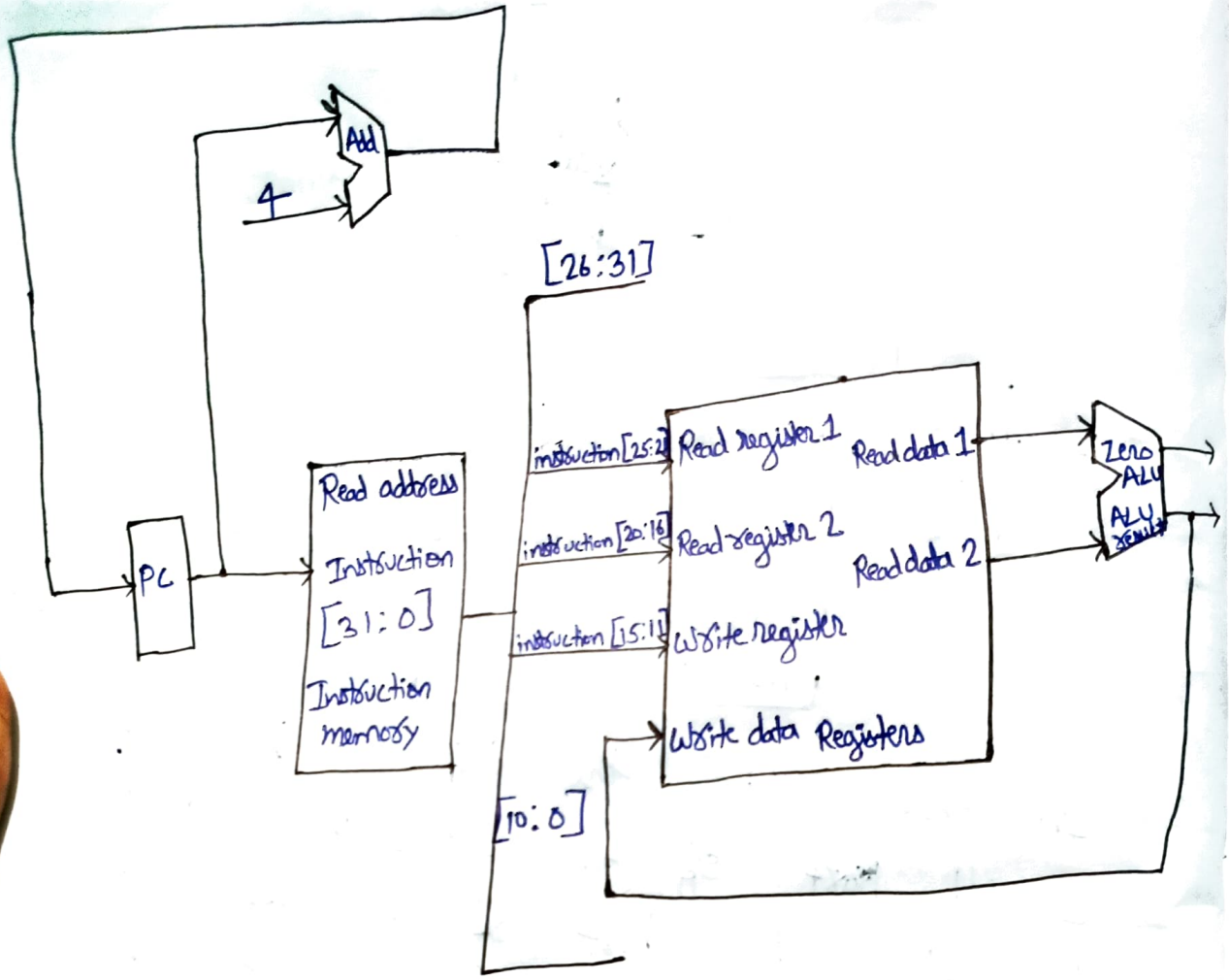
Soln → The diagram for the datapath where PC fetches the instructions sequentially is shown below.



→ In this datapath the PC value is fed to 1 i/p port of the adder whereas 4 is fed to other port of the adder or i/p port of the adder and adder give the o/p PC+4 which is again fed back to PC hence address contain of PC is PC+4 which is address of next instruction to be fetched.

Q4) Discuss and explain the design of datapath for executing R type of instruction.

Soln The diagram for the datapath for executing R type of instruction is shown below.



→ Here, the PC provides the address of the instruction which is 'R' type. Now, the instruction is fetched from instruction memory which is a 32 bit instruction and the instruction is decoded and the different field of the instruction is extracted. Here, the instruction is divided into 6 parts as per, the 'R' format instruction.

→ Instruction bit 21 to 25 and 16 to 20 represent the address of the two source register file and instruction bit 11 to 15 represents the destination address of the register in the register file.

are the address of the read register are giving the data are read from these registers as data 1 and data 2 respectively and fed to the ALU for the ALU operation. The 32 bit ALU result is fed that to the register file at the write data port.

Q5) What is datapath and control path? Why both designs are separate? Explain.

Soln → A datapath is a collection of functional units such as arithmetic logic units (ALU) or multipliers that perform data processing operations, registers, and buses.

→ Control path is the physical entity in a processor which: fetches instructions, fetches operands, decodes instructions, schedules events in the datapath which actually causes the instruction to be executed.

→ The main reason for separating the datapath and control path is to improve the modularity and flexibility of the system.

- Separating the datapath and control path also allows for easier implementation of pipelining and other performance-enhancing techniques.
- Separating the datapath and control path in CoA improves modularity, flexibility and performance making it easier to design and optimize complex systems.