

Assignment No. 1

Good Luck | Page 1
Data

Define computer organization and architecture

q1 1. Computer Organization

Computer organization refers to the operational units and their interconnections that realize the architectural specifications. Examples are things that are transparent to the programmer:

1. control signals

2. Interfaces between computer and peripherals

3. The memory technology being used

It deals with low level design and low level components & connection between adders, gates, flip-flops.

2. Computer Architecture

Computer Architecture refers to those attributes of a system that have a direct impact on the logical execution of a program. Examples :

1. The instruction set

2. The number of bits used to represent various data types.

3. I/O mechanisms

4. memory addressing techniques

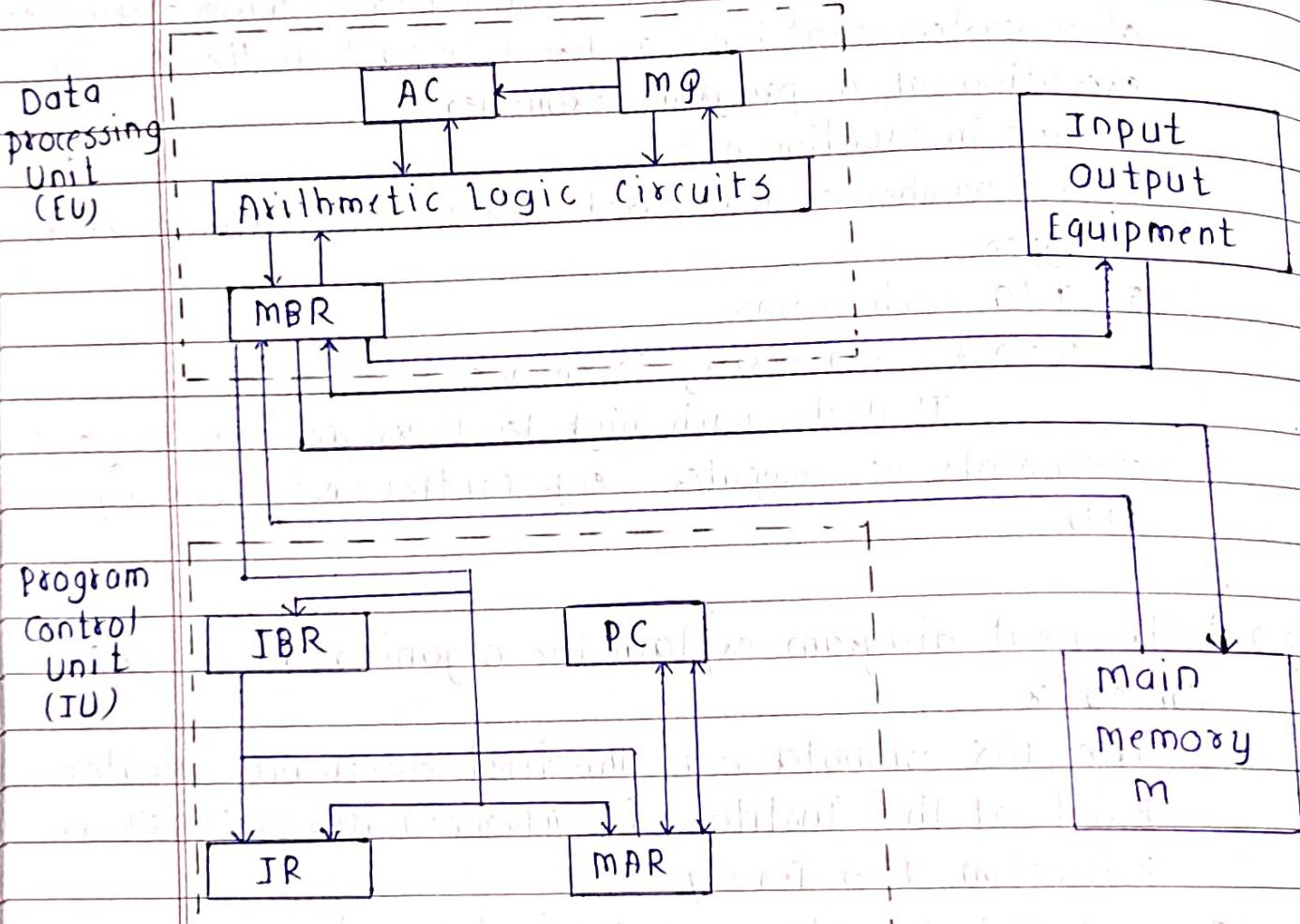
It deals with high level design and design of components of computer. e.g. control unit, memory, ALU.

q2. With neat diagram explain the organization of IAS computer.

1. The IAS computer was the first electronic computer built at the Institute for Advanced study (IAS) in Princeton, New Jersey.

2. The IAS computer was the first to store the program in the same memory as the data.

3. The logic of the TAS machine was implemented with vacuum tubes.
4. The memory of the TAS machine was implemented with cathode ray tubes. The word size is 40 bits.
5. Memory is word addressed (not byte addressed)
6. Numbers were stored as fractions, not as integers.
7. The distinction between fractions and integers only affects multiplication and division.
8. Instructions are fixed size (20 bits) with a single address, stored 2 per word. 8 bits are allocated for the opcode.
9. It allows 256 different operations out of them only 21 were implemented.



1. Data processing unit is also called as execution unit
2. Program control unit is also called as instruction unit
3. Program control unit consists of IBR, PC, IR, MAR
 - i. IBR (Instruction Buffer Register) - IBR is employed to hold temporarily the right-hand instructions from a word in memory
 - ii. IR (Instruction Register) - IR contains the 8-bit opcode instruction being executed
 - iii. PC (Program Counter) - PC is an counter that contains the address of the next instruction-pair to be fetched from memory to be executed
 - iv. MAR (Memory Address Register) - MAR specifies the address in memory of the word to be written from or read into the MBR

4. Data processing unit consists of MBR, AC, MQ, and arithmetic logic circuits.
 - i. MBR (Memory Buffer Register) - MBR is a two-way register that holds the data fetched from memory and ready for the CPU to process or the data waiting to be stored in memory
 - ii. AC and MQ (Accumulator and multiplier quotient) AC and MQ are employed to hold temporarily operands and results of ALU operations.

Q.3. Explain the organization of personal computer system

1. Personal computer system uses the microprocessor chips.
2. Cache memory is used to speedup the execution time of microprocessors.
3. It contains 3 performance benchmarks

1. Software - The efficiency with which the programs written and compile into object code influence N no. of instructions to be executed.
2. Architecture - The efficiency with which individual instruction are processed directly affects CPI (Clock cycle per instruction).
3. Hardware - The speed of processor circuits determines f clock frequency.
4. It supports the pipelining. Pipelining means fetching the next instruction while executing the current instruction.
5. The organization of microprocessor is as follows:
 - A. 1971- 4004
 - i. 4004 is the first microprocessor invented in 1971.
 - ii. In that all components on single chip.
 - iii. The 4004 can add two 4-bit numbers & can multiply only by repeated addition.
 - B. 1972- 8008
 - i. 8008 is the 8-bit microprocessor.
 - ii. The 4004 and 8008 are designed for specific applications.
 - C. 1974- 8080
 - i. It is intel's first general purpose microprocessor.
 - ii. 8 bit data path.
 - D. 8086
 - i. It is 16-bit microprocessor.
 - ii. It consists of instruction cache, prefetch few instructions.

Differentiate between RISC and CISC

Q4

RISC

1. RISC is reduce instruction set computer
2. The no. of instructions is less as compared to CISC.
3. The addressing modes are less.
4. It works in a fixed instruction format.
5. The RISC consumes low power.
6. Requires more RAM
7. Pipelining is used
8. For control unit design hardware components are used e.g. flip-flop, gates etc.

CISC

1. CISC is complex instruction set computer
2. The number of instructions is more as compared to RISC.
3. The addressing modes are more.
4. It works in a variable instruction format.
5. The CISC consumes high power
6. Requires less RAM
7. Pipelining is not used
8. For control unit design by writing microprograms programs written in octal format.

Q5. Write a note on addressing modes

1. The way the operands are chosen during execution of an instruction is determined by the addressing mode

Opcode	mode	Address
--------	------	---------

2. Purpose of addressing modes

- i. To provide programming flexibility to the user like pointers to memory, counters for loop control, indexing of data etc.
- ii. To reduce the no. of bits in the addressing field of the instruction

3. Types of addressing modes

- i. Implied Mode - Operands are specified implicitly in the instruction itself.
e.g. "Increment Accumulator".
- ii. Immediate mode - The address field contains the operand itself instead of the address of the operand.
- iii. Register mode - The address field contains the address of a CPU register which contains the operand.
- iv. Register Indirect mode - The address field contains the address of a register which holds the memory address of the operand.
- v. Direct Address mode - The address field contains the memory address of the operand.
- vi. Indirect Address mode - The address field contains the address of the memory location that contains the operand.
- vii. Indexed Addressing mode - The content of an index register is added to the address part of the instruction in order to obtain the effective address.
- viii. Base Register Addressing mode - The content of a base register is added to the address part of the instruction in order to obtain the effective address.
- ix. Relative Addressing mode - The address of the program counter is added to the address field to get the address of the operand.

Define the terms user mode, supervisor mode

1. User Mode

- i. The system is in user mode when the operating system is running a user application such as handling a text editor.
- ii. The transition from user mode to supervisor mode occurs when the application requests the help of operating system or an interrupt or a system call occurs.
- iii. The mode bit is set to 1 in the user mode. It is changed from 1 to 0 when switching from user mode to supervisor mode.

2. Supervisor Mode

- i. The system starts in supervisor mode when it boots and after the operating system is loaded, it executes applications in user mode.
- ii. There are some privileged instructions that can only be executed in supervisor mode.
- iii. These are interrupt instructions, input output management etc.
- iv. If the privileged instructions are executed in user mode, it is illegal and a trap is generated.

Q.4.

Differentiate between RISC and CISC

RISC

1. RISC is reduce instruction set computer
2. The no. of instructions is less as compared to CISC.
3. The addressing modes are less.
4. It works in a fixed instruction format.
5. The RISC consumes low power.
6. Requires more RAM
7. Pipelining is used
8. For control unit design hardware components are used e.g. flip-flop, gates etc.

CISC

1. CISC is complex instruction set computer
2. The number of instructions is more as compared to RISC.
3. The addressing modes are more.
4. It works in a variable instruction format.
5. The CISC consumes high power
6. Requires less RAM
7. Pipelining is not used
8. For control unit design by writing microprograms programs written in 0's 1 format

Q.5. Write a note on addressing modes

1. The way the operands are chosen during execution of an instruction is determined by the addressing mode

Opcode	mode	Address
--------	------	---------

2. Purpose of addressing modes

- i. To provide programming flexibility to the user like pointers to memory, counters for loop control, indexing of data etc.
- ii. To reduce the no of bits in the addressing field of the instruction

3. Types of addressing modes:

i. Implied Mode - Operands are specified implicitly in the instruction itself.
e.g. "Increment Accumulator".

ii. Immediate mode - The address field contains the operand itself instead of the address of the operand.

iii. Register mode - The address field contains the address of a CPU register which contains the operand.

iv. Register Indirect mode - The address field contains the address of a register which holds the memory address of the operand.

v. Direct Address mode - The address field contains the memory address of the operand.

vi. Indirect Address mode - The address field contains the address of the memory location that contains the operand.

vii. Indexed Addressing mode - The content of an index register is added to the address part of the instruction in order to obtain the effective address.

viii. Base Register Addressing mode - The content of a base register is added to the address part of the instruction in order to obtain the effective address.

ix. Relative Addressing mode - The address of the program counter is added to the address field to get the address of the operand.

Q.6 Define the terms user mode, supervisor mode

i. User Mode

- The system is in user mode when the operating system is running a user application such as handling a text editor.
- The transition from user mode to supervisor mode occurs when the application requests the help of operating system or an interrupt or a system call occurs.
- The mode bit is set to 1 in the user mode. It is changed from 1 to 0 when switching from user mode to supervisor mode.

ii. Supervisor Mode

- The system starts in supervisor mode when it boots and after the operating system is loaded, it executes applications in user mode.
- There are some privileged instructions that can only be executed in supervisor mode.
- These are interrupt instructions, input output management etc.
- If the privileged instructions are executed in user mode, it is illegal and a trap is generated.

Q.7 Evaluate i. $x = (A+B)^*(C+D)$ ii. $x = AB + CD$

$$i. x = (A+B)^*(C+D)$$

i. According to three address instruction, $x = (A+B)^*(C+D)$ can be represented as,

ADD R1, A, B

ADD R2, C, D

MUL X, R1, R2

ii. Second Address Instruction

MOV R₁, N
ADD R₁, B
MOV R₂, C
ADD R₂, D
MUL R₁, R₂
MOV X, R₁

iii. One address instruction

LOAD A
ADD B
STORE T
LOAD C
ADD D
MUL T
STORE X

iv. Zero address instruction

PUSH A
PUSH B
ADD
PUSH C
PUSH D
ADD
MUL
POP X

$$2. \quad x = AB + CD$$

i. Three address instructions for this expression are as follow:

MUL R₁, A, B

MUL R₂, C, D

ADD X, R₁, R₂

(constant
variables)

ii. Two address instructions for given expression are

MOV R1, A
MUL R1, B
MOV R2, C
MUL R2, D
ADD R1, R2
MOV X, R1

iii. One address instructions for given expression are

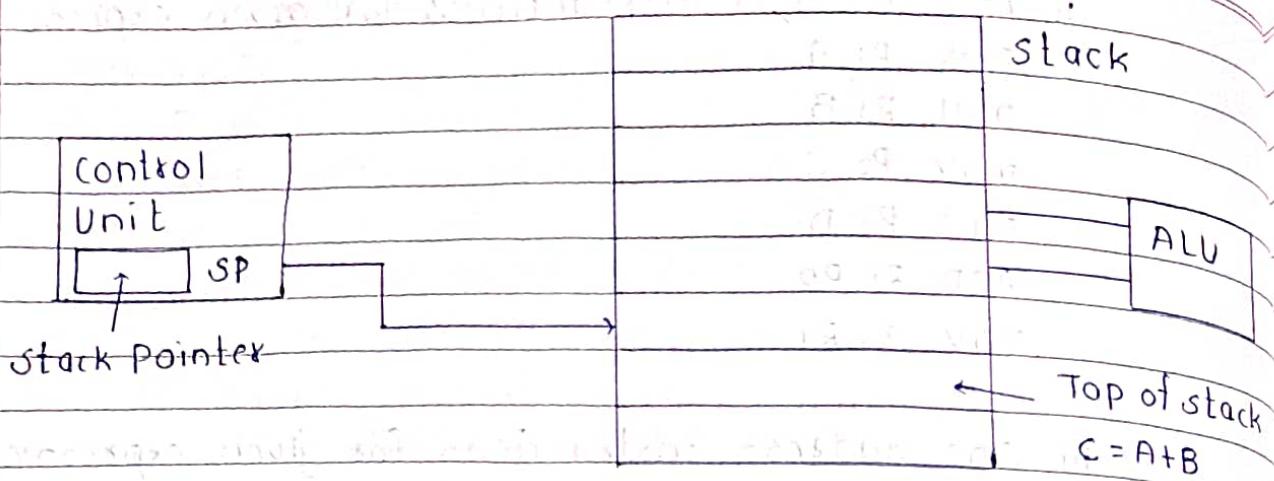
LOAD A
MUL B
STORE T
LOAD C
MUL D
ADD T
STORE X

iv. Zero address instructions for given expression are

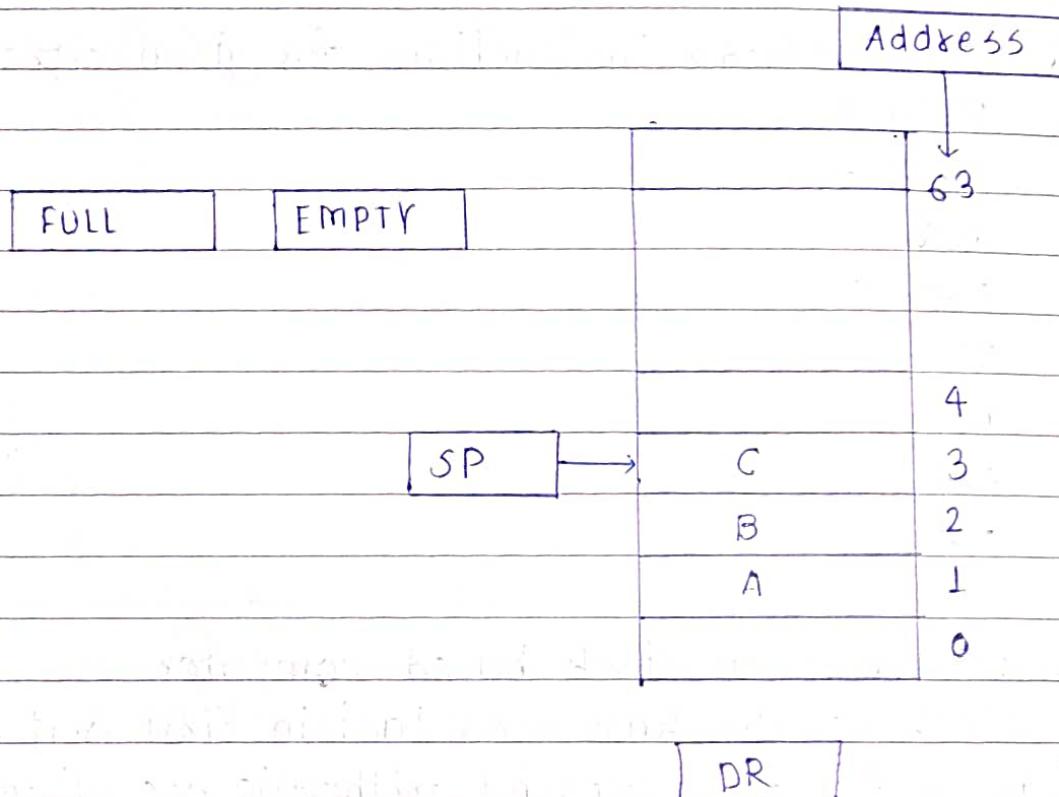
PUSH A
PUSH B
MUL
PUSH C
PUSH D
MUL
ADD
POP X

Q.8 Write a note on stack based computer.

1. Stack is also known as last in first out (LIFO). It is useful for compound arithmetic operations and nested subroutine calls.
2. It is a storage device that stores information in such a manner that the item stored last is the first item retrieved.



3. A stack pointer is a register that holds the address of the top item in the stack. It is always incremented/decremented by 2 during push & pop operations.
4. Push is the operation to insert an item into the stack.
5. Pop is the operation to retrieve an item from the stack.



The stack can be implemented using two ways:

1. Register Stack
2. Memory Stack

1. Register stack

- i. A stack can be organized as a collection of a finite number of registers.
- ii. In a 64-word stack the stack pointer contains 6 bits.
- iii. The one-bit register FULL is set to 1 when the stack is full; empty register is 1 when the stack is empty.
- iv. The data register DR holds the data to be written into or read from the stack.

2. memory stack

- i. A stack may be implemented in a computer's random access memory.

The reverse polish notation in the stack is also known as postfix expression
 consider infix notation $A + B$. Reverse polish notation $AB+$ also called postfix.

Q.g. Describe i-format, j-format and x-format.

i. i-format

- i. The i-instruction format contains fields for two registers (typically source and destination) and for a 16-bit immediate value.
- ii. The i-format is used for arithmetic operations with an immediate operand and also for load / store instructions in which a base+ displacement memory addressing scheme is used.

Opcode	RS	RD	Immediate Address
6 bits	5 bits	5 bits	16 bits

2 J-instruction format

- i. The j-format is used for the jump instruction, which jumps to an absolute address.
- ii. Because instructions must be aligned to 32-bits, the low 3 bits of every valid address are always 0.
- iii. Thus we can jump to any one of 2^{32-6+3} addresses. The currently-executing address is used to supply the missing 3 bits.

Opcode	Address
6 bits	26 bits

- iv. The only instructions which use the J format are Jump and Jump-and-Link, which have opcodes 0x2 and 0x3.

3 R-instruction format

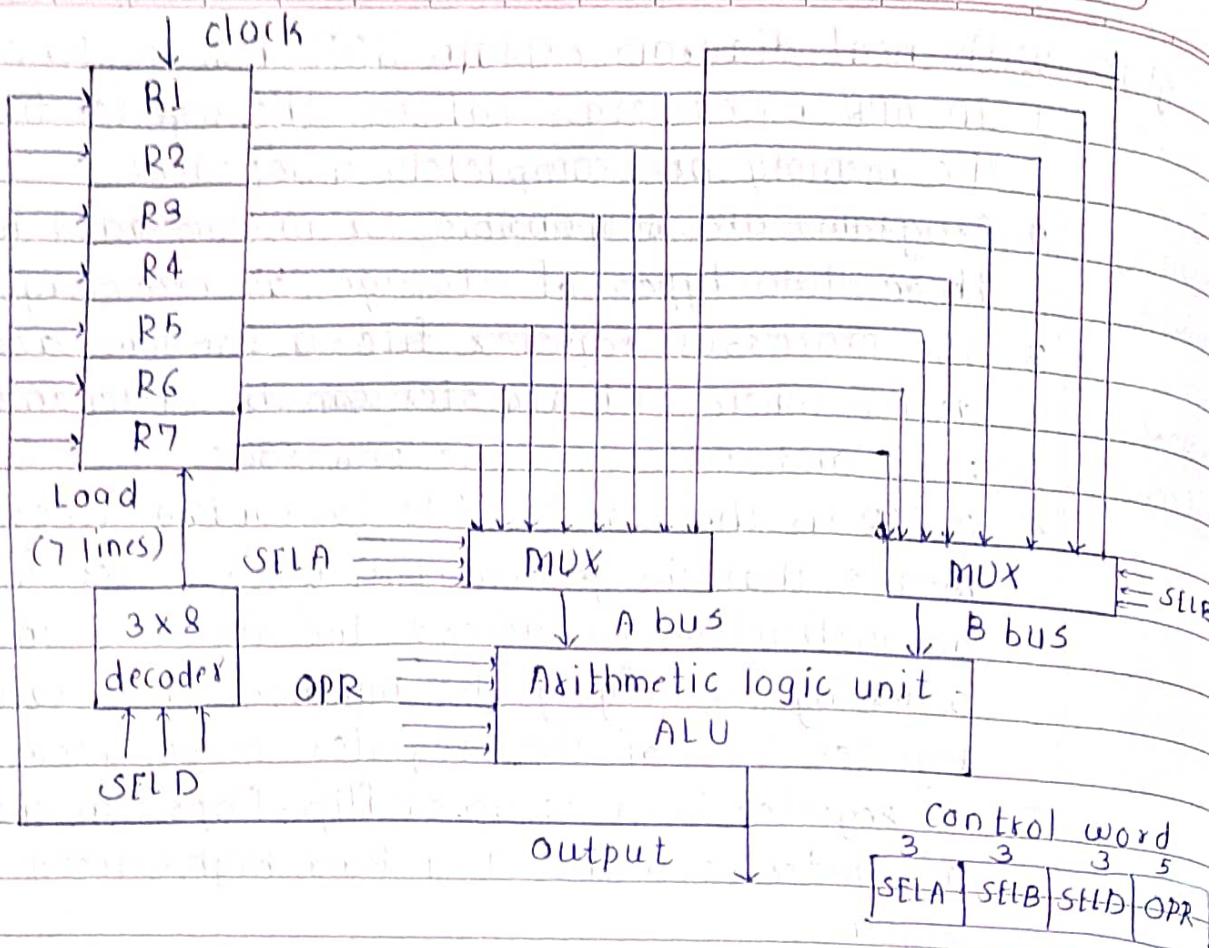
- i. The R instruction format has fields for three registers (typically, two sources and a destination), as well as shift amount (5 bits) and a function (6 bits).
- ii. It is used for arithmetic/bitwise instructions which do not have an immediate operand.

Opcode = 000000	RS	RT	RD	sh.Amt	Func.
6 bits	5	5	5	5	6

- iii. The function field specifies the actual arithmetic function to be applied to the operands given by the RS, RT (sources) and RD (destination) fields.

- iv. For example, function 32 (100000b) is addition.
- v. The left/right shift instructions use the shift amount field to specify the amount to shift.
- vi. All R-instructions have 0 opcode field.

- With neat diagram explain GPR based CPU organization.
1. In GPR based organizations, the register file and the memory are completely independent.
 2. Programs are responsible for moving data between these two types of storage as necessary.
 3. The processor register file - a random-access device where each register can be independently read and written by the processor.
 4. A GPR in the register file is random access, which means that the general-purpose register file allows an instruction to access the registers in any order by specifying the number (also called the register ID) of the registers to be accessed.
 5. A register is made up of flip-flops. In the CPU, a register is a one-of-a-kind, high-speed storage region.
 6. Combinational circuits are used to implement data processing.
 7. The following are two essential functions implemented by registers in CPU operation:
 - i. It can be used as a temporary data store site. This allows directly implementing applications to have quick access to data when needed.
 - ii. It can record the CPU's condition and information about the currently executing programme.
 8. If a CPU has some registers, these registers can be linked by a shared bus.
 9. The following diagram shows the general organization of seven CPU registers.
 10. The control unit is in charge of the CPU bus system. The control unit specifies the data flow via the ALU by selecting the ALU's function and system components.



General Purpose Register Organization

Q.11 Explain IEEE754 format for representation of floating point numbers.

1. The IEEE standard for floating-point arithmetic is a technical standard for floating-point computation which was established in 1985 by the Institute of Electrical and Electronics Engineers.
2. There are several ways to represent floating point numbers but IEEE 754 is the most efficient in most cases. IEEE 754 has 3 basic components.

i. The sign of mantissa

This is as simple as the name. 0 represents a positive number while 1 represents a negative number.

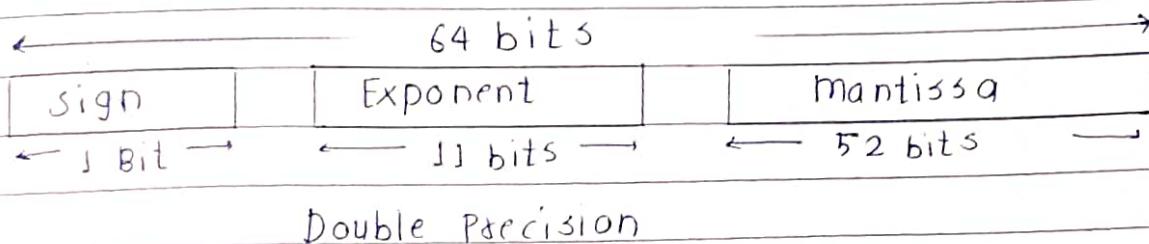
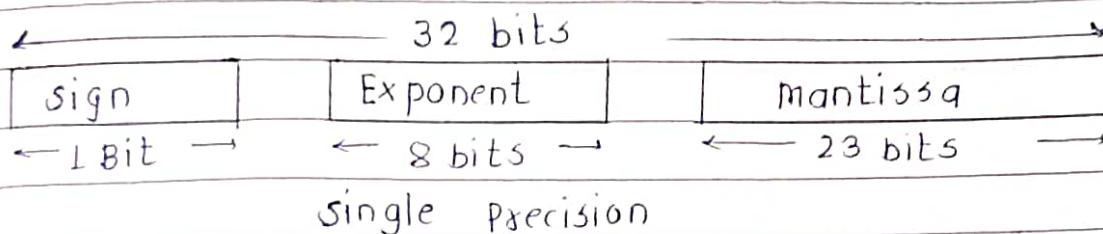
ii. The biased exponent

The exponent field needs to represent both positive and negative exponents. A bias is added to the actual exponent in order to get the stored exponent.

iii. The normalised mantissa

The mantissa is part of a number in scientific notation or a floating-point number, consisting of its significant digits.

3. IEEE754 numbers are divided into two based on the above three components: single precision & double precision



4. Floating point number is represented by scientific notation: $N = m \times B^E$ where m : Mantissa

B = Base E = Exponent

5. Decimal number system uses $B=10$ and Binary number system uses $B=2$

6. Representation of floating point number is not unique.

7. To get unique presentation, the fractional part is in its normalized form

8. It is normalized by adjusting radix point position such that there is only a single non-zero digit before the radix point

9. Adjusting exponent

- moving radix point to left, increment exponent.
- moving radix point to right, decrement exponent.

10. IEEE754 standard

$$\text{i. } N = (-1)^S \times 2^{E-127} \times 1.m$$

ii. If $E=255$ and $m_1=0$ then $N=\text{NaN}$

iii. If $E=255$ and $m\neq 0$, then $N=\text{infinity}$

iv. If $0 < E < 255$ then $N = (-1)^S \times 2^{E-127} \times 1.m$

v. If $E=0$ and $m_1=0$, then $N = (-1)^S \times 2^{E-128} \times 0.m$
(Overflow)

vi. If $E=0$ and $m\neq 0$, then $N = (-1)^S + 0 \cdot (Underflow)$