Q.1 a. Discuss about error detection and correction using parity checkers and generators.

Answer:

An error detection code is an binary code that detects digital errors during transmission. The most common error is parity bit. A parity bit is an extra bit included with a binary message to make the total number of 1's either odd or even. The parity bit is chosen in such a way that the sum of 1's is odd for odd parity and the sum of 1's is even for even parity. During data transfer at the sending end the message bits are applied to a parity generator and transmitted along with the parity bit. At the receiving end all the incoming bits are applied to a parity checker .An error is detected if the checked parity does not conform to the adopted parity. An xor gate is used as an odd parity generator.

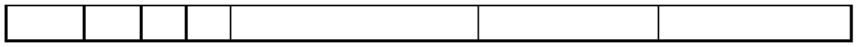
Diagram of odd parity detector with xor gate. ----------(2 +2 marks)

b. Compare Horizontal Micro Code & Vertical Micro Code.

Answer:

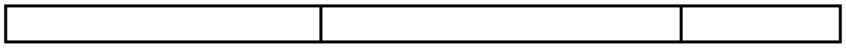
The concept of microprogramming is based on reading stored control programs. The micro operations are executed when the corresponding control signals are made active. The control signals are stored in a control memory. By fetching control memory words one by one the control signals can be generated. Each control memory word is known as micro instruction. Having an individual bit for each control signal in the micro instruction format is known as horizontal micro code. In a vertical microcode a single field can produce an encoded sequence. This technique takes more time for generating the control signals and more micro instructions are needed.

Horizontal micro code



Control bits **branch conditions** **next address**

Vertical micro code



Micro code **branch conditions** **next address**

Explanation of the formats with difference ----------(4 marks)

c. What is meant by overflow in binary addition and how is it detected?

Answer:

When 2 numbers of n digits are added and the sum occupies n+ 1 digit, an overflow has occurred. If both numbers are positive or both numbers are negative, overflow occurs. When 2 numbers are added in sign magnitude form an overflow can be easily detected from the carry out of the number of bits.

If the 2 numbers are added in 2's complement form the sign bits are part of the number and hence overflow cannot be detected. Therefore the overflow bit changes the sign bits in 2's complement addition.

An overflow occurs if the 2 numbers are positive and the sum contains a negative sign or vice versa. This can be detected using the carry in to the sign bit position and the carry out of the sign bit position.

If the 2 carries are not equal an overflow condition will occur. An xor gate can be used to detect the overflow with both the carry in and carry out bits as inputs and the output will be 1.

4 points or XOR gate diagram with cin and cout ----------(4 marks)

d. Write short notes on Assemblers.

Answer:

An assembler is a system program that translates an assembly language program to an equivalent binary machine language .It converts the source program into an object program. It operates on character strings and produces an equivalent binary interpretation. A 2 pass assembler scans the entire program twice. During 1st pass it generates a table that correlates all user defied address symbols with their binary equivalent value. The binary translation is done during the 2^{nd} pass. During 2^{nd} pass a table look up procedure is used which is a search of table entries to determine whether a specific item matches one of the items stored in the table. The important task of the assembler is to check for possible errors in the object program called as error diagnostics.

Definition of assembler -----(1 mark)
Characteristics of assemblers---- (3 marks)

e. Name few high performance techniques to improve system performance.

Answer:

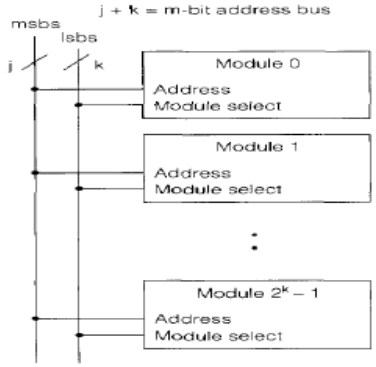
Small memory access time, using cache memory ii. Fetching an instruction in advance-prefetch so that 2 consecutive instructions are overlapped iii. Organising main memory into 2 independent banks one containing odd addresses and the other even addresses known as memory interleaving. The CPU can be divided into different sections so that each stage of an instruction cycle can be done by an independent section. This technique called as instruction pipelining. Superscalar architecture works on multiple adders in single CPU. The overall performance can be improved by system design techniques called multi processing and multi programming.

Listing of 3 methods + Explanation ----------(3 +1 marks)

f. Elaborate the concept of memory interleaving.

Answer:

In order to carry out 2 or more simultaneous accesses to memory the memory must be partitioned into separate modules which are independent of each other. In an interleaved memory consecutive addresses are assigned to different memory modules. In a 2 module memory system the even addresses are in one module and the odd addresses in the other. With m way interleaving m consecutive instructions can be fetched from m different modules. Hence by staggering the memory access effective memory cycle time can be reduced .When CPU is accessing more than one consecutive locations both odd and even modules can perform simultaneous read and write operations and hence access time is reduced by half.



(a) Consecutive words in consecutive modules (interleaving)

Diagram + definition ----------(2 + 2 marks)

g. How do you explain the computer system from the following different views?
(i) Programmer's view
(ii) Computer architect's view **(7 × 4)**

Answer:

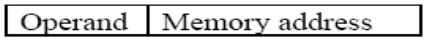
The assembly language level programmer would have to focus not only on the algorithm and data structures but also on details such as which hardware register to use , where to store the program and what instructions are needed for operations etc. Therefore a programmer's view can be logic designer's view when assembly language is used or an architect's view when high level language is used. An architect is directly involved with i.CPU registers , memory map, I/O map, Device controllers, Instruction set, DMA channels .There are buses communicating with the peripherals, and memory.These operations are controlled by the control unit which interacts with the processor.How the devices interact with each other and how the speed mismatches can be solved and the result of simultaneous accessing of memory are solved by the designer or architect.

----------**(2 + 2 marks)**

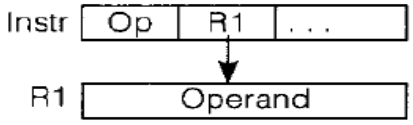
Q.2 a. Discuss the following addressing modes with functional diagrams: **(9)**
(i) Register direct addressing
(ii) Relative addressing
(iii) Index addressing

Answer:

Register direct addressing : Operand is in a register ;the register address is given in the instruction; Since the operand address is directly available in the instruction there is no need to calculate the operand address and hence instruction cycle time is reduced.[memory address] = operand

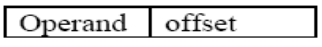


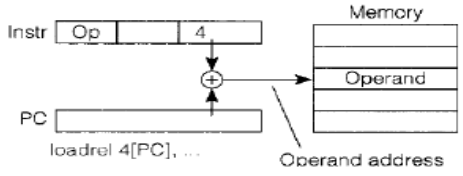
Load R1, X : Loads the contents of memory location in register R1.



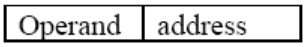
Relative addressing: Instead of using GPR the effective address is determined by the contents of program counter (PC) . The instruction specifies the operand address as the relative position of the current instruction address i.e contents of PC. This mode is used to specify the branch address in the branch instruction.

operand address = [PC] + offset
[PC] = 825 PC <- PC +1 = 826
[825] = 24 Therefore effective address = 826 + 24 =850

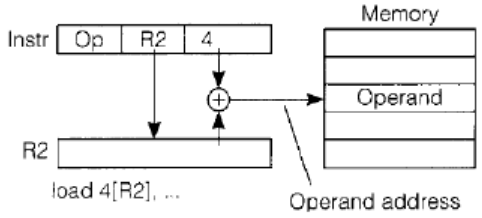




Index addressing : An index register contains an offset or displacement. The instruction contains the address that should be added to the offset in the index register, to get the effective operand address. The address field in the instruction gives the start address of an array in memory. The index register contains the index value for the operand i.e the difference between the start address and the operand address.
Effective address= contents of base referenced + value in displacement field



Operand address = Address + [Index register]



Explanation with diagrams ----------(3 x 3 marks)

b. What is the philosophy of RISC based machine? How is it different from a CISC Based machine? Discuss briefly. (9)

Answer:
CISC –
i. large number of powerful instructions taking multiple clock cycles. An average CPI is between 2 and 15. ii. Most instructions refer memory. iii. Memory reference addressing modes and variable format instructions iv. Slower instruction execution format with single register set non pipelined operation with micro programmed control. v. Due to powerful instructions in the instruction set of a CPU, the main memory size and the cost are reduced.vi. As the number of instructions is lesser execution time is reduced with a highly efficient compiler. vii. The control unit design becomes very complex and due to lot of hardware circuitry as the CPU is complex, the hardware cost and power requirement are increased.

RISC – Lesser number of simple instructions in the instruction set of CPU. Ii. Equal instruction length for all instructions iii. Large number of registers iv. Load/Store architecture v. faster instruction execution time vi. Few addressing modes vii. Multiple register sets with register to register addressing----------**(9 marks)**

Q.3 a. Classify computers based on the instruction formats of CPU organization. Explain each with an example. (9)

Answer:
Most computers fall into one of 3 CPU organizations i. Single accumulator organization ii. General register organization iii. Stack organization.
In single accumulator type, all operations are performed with implied accumulator register. It uses one address field. ADD X where X is the address of the operand.

AC <- AC + M [X].

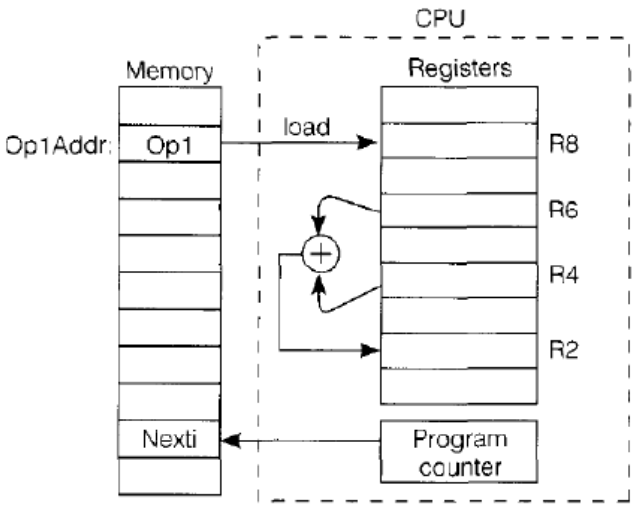
AC is the accumulator register and M[X] denotes the memory word located at address X.
In general register type of organization 3 address register fields are needed.

ADD R1,R2, R3 to denote the operation **R1 <- R2 + R3.** The number of address fields can be reduced from 3 to 2 if the destination register is the same as that of the source register.

ADD R1,R2 **R1<- R1 + R2**

Each address field may specify a processor register or a memory word.

ADD R1, X **R1<- R1 + M[X].** It has 2 address fields one for R1 and the other for memory address X.

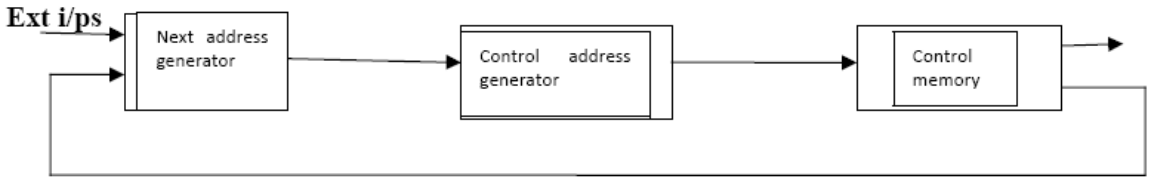


Computers with STACK would require PUSH and POP operations with an address field. PUSH X will push the word at address X to the top of the stack. The stack pointer is automatically updated. These types of instructions do not need an address field .The instruction formats can also be classified according to number of address fields as single address, 2 address and 3 address instructions.

3 INSTRUCTION FORMATS WITH EXAMPLE FOR EACH ---------(3X3 MARKS)

b. Compare micro programmed control versus hardwired control. (9)

Answer:



CONTROL UNIT

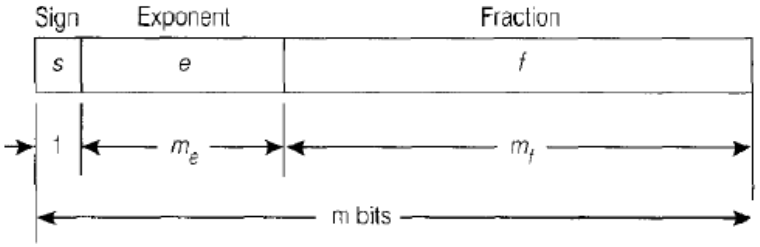
micro programmed control	hardwired control
The control units are Generated by a series of sequential steps of micro operations .It uses micro programmed control unit and the steps are initialized using control word of 0's and 1's.Each control word of memory is called micro instruction and the sequence is micro program stored in ROM.	The control functions are generated by hardware using logic techniques. It does not use a control memory.
It provides a well structured control organization into kind of programming.	The control unit is random using ffs, registers, decoders etc.
The control unit is more adaptable to changes and hence more flexible. But redesign is required, but adding new features requires only adding control memory.	Not flexible in accepting new system specific or new instructions.
More number of micro instructions are used (>100)	Less number of instructions
Used in mainframes and microprocessors.	Mostly in RISC processors.
Uses more area	Uses more area
Routines are used and hence performance is improved.	No routines are used.
The use of control memory decreases the speed .	Increases the speed.

MICRO PROGRAM CONTROL UNIT DIAGRAM ----------(2 MARKS)
COMPARISON TABLE MINIMUM 7 POINTS ----------(7 MARKS)

Q.4 a. Discuss about the special values and exceptions in floating point number system and obtain the range of numbers that can be represented using that format. (9)

Answer:

To represent very large numbers and very small numbers a computer must be able to represent them such that the binary point is variable. The binary point is said to be floating and hence called floating point number.



$1 + m_e + m_f = m$, Value$(s, e, f) = (-1)^s \times f \times 2^e$

S	E'	M
1 bit	8bits	23 bits

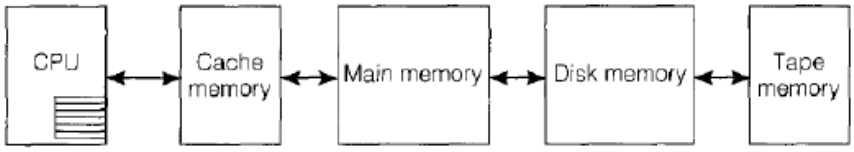
SIGN	E'BIASED EXPONENT	MANTISSA
0= + 1= -	E'= E +bias 8 bits E-signed exponent	23 bit fraction
	E' ranges between 0 to 255	
	E'= 0 –Exact zero E' =255 –α 0<E'<255 Range of numbers with actual exponent E in the range between -126<E< 127	+/- 1.M X $2^{E'-127}$ Range of numbers = +/- 2^{-126} to +/- 2^{127} = +/-10^{+-38}

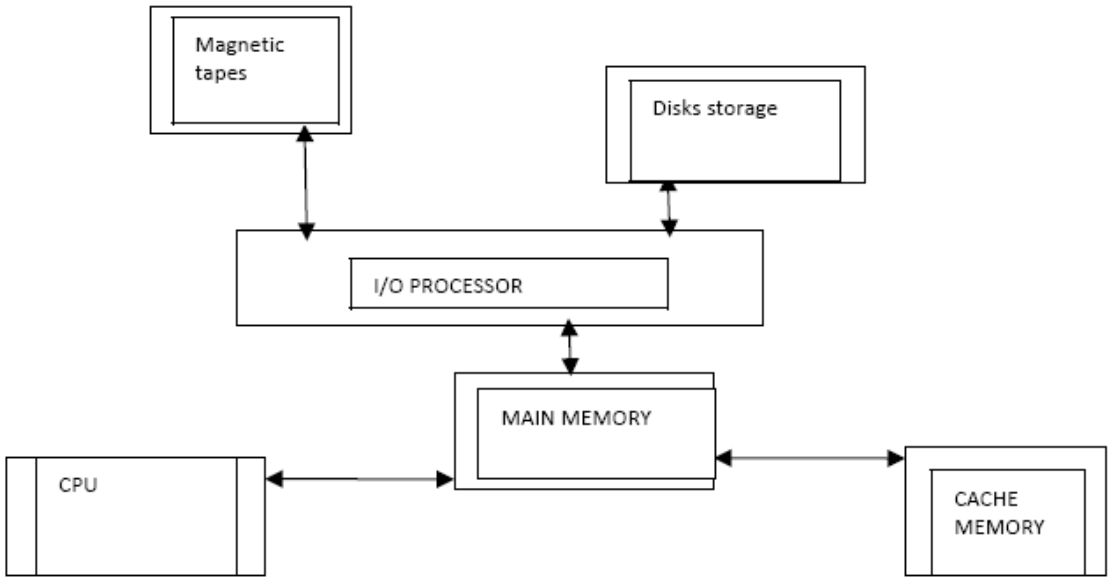
BIASED EXPONENT RANGES ----------(5 MARKS)
MANTISSA RANGES ----------(4 MARKS)

b. Explain in detail about memory hierarchy. (9)

Answer:

The total memory capacity of a computer can be visualized as being a hierarchy of components. The memory hierarchy system consists of all storage devices from the slow but high capacity auxillary memory device to a relatively faster main memory to an even smaller and very fast buffer memory accessible to the high speed processor.

Component					
Access type	Random access	Random access	Random access	Direct access	Sequential access
Capacity, bytes	64–1024	8–256 KB	8–64 MB	1–10 GB	1 TB
Latency	1–10 ns	20 ns	50 ns	10 ms	10 ms–10 s
Block size	1 word	16 words	16 words	4 KB	4 KB
Bandwidth	System clock rate	8 MB/s	1 MB/s	1 MB/s	1 MB/s
Cost/MB	High	$500	$30	$0.25	$0.02



At the bottom of the hierarchy are the slow magnetic tapes used to store removable files. Above it are the magnetic drums or disks to store backup data. The main memory occupies a central position to communicate directly with the CPU and auxiliary devices through an I/O processor. The cache memory is used in large computer systems to compensate for the speed difference between the main memory access time .

MEMORY HIERARCHY BLOCK DIAGRAM ----------(4 MARKS)
EXPLANATION ----------(4 MARKS)

Q.5 a. Discuss associative mapping technique and state its advantages over direct mapping technique. (9)

Answer:

5.a Three different types of mapping functions are in common use: associative, direct-mapped, and block-set-associative. We divide main memory addresses into fields and these fields partition the main memory address into blocks and words within the blocks. As an example, a 32-bit main memory address is partitioned into two fields, a low-order field specifying the word in a 64-byte block, and a high-order field specifying the block number. Thus there are 2^{26} or 64 M 64-byte blocks. The block number field may be further partitioned for purposes of finding the block in the cache.

In associative mapping, any block from mapping main memory can be placed anywhere in the cache. After being placed in the cache, a given block is identified uniquely by its main memory block number, referred to as the tag, which is stored inside a separate tag memory in the cache. Regardless of the kind of cache, a given block in the cache may or may not contain valid information.
The CPU address value of 15 bits is placed in the argument register and the associative memory is searched for a matching address. If the address is found the corresponding 12 bit data is read. If no match occurs the main memory is accessed for the word. The address data pair is the transferred to the associative cache memory. If the cache is full then an address data pair must be displaced to make room for a pair. Direct-mapped caches form the other extreme, where a given main memory blocks can be placed in one and only one place in the cache. The main memory needs an address that includes both the tag and the index bits. The number of bits in the index field is equal to the number of address bits required to access the cache memory.

EXPLANATION -----(6 MARKS)
FIGURE----------(3 MARKS)
SECTION 12.6 COMPUTER SYSTEM ARCHITECTURE- M MONO
FIGURE12-17,12-18 ON PG NO 504-505

b. A block set-associative cache consists of a total of 64 blocks divided into 4 block sets. The main memory contains 4096 blocks each consisting of 128 words.
(i) How many bits are there in the main memory address?
(ii) How many bits are there in each of TAG, SET, WORD fields? (9)

Answer:

5. b. A block set-associative cache consists of a total of 64 blocks divided into 4 block sets the main memory contains 4096 blocks each consisting of 128 words.
(a). How many bits are there in the main memory address? **(3 marks)**
(b). How many bits are there in each of TAG, SET, WORD fields? **(6 marks)**

64 blocks divided into 4 block sets. Main memory is 4096 blocks of 128 words
Set associative cache
4096 = 2^{12} 128 words = 2^7 hence 19 bits for main memory address.

Tag	Set	word
6	6	4

Main memory address

Here there are 2 blocks per set. Therefore main memory blocks are 0,64,128,…map into cache set 0.
Hence for the given problem

Tag	Set	word
4	7	7

TAG field is 4 bits SET field is 7 bits
Word field is 7 bits

Q.6 a. Differentiate priority interrupt and daisy chain priority interrupt. (9)

Answer:

Data transfers between the CPU and an I/O device must be initiated by the CPU. The readiness of the device can be determined from the status of a control line or flag. The CPU can monitor the status of the flag but this is time consuming and hence the device interface can interrupt the CPU every time the flag is enabled. The CPU responds to the interrupt signal by storing the return address from PC into a memory stack and then branching to a service routine that processes the required transfer.

A priority interrupt is an interrupt system that establishes a priority over various sources to determine which condition is to be serviced first when 2 or more requests arrive simultaneously. It can be designed to determine which conditions are permitted to interrupt while another interrupt is being serviced. Higher priority interrupt levels are assigned to requests if delayed could have serious consequences. Establishing the priority of simultaneous interrupts can be done software or hardware. A polling procedure is used to identify the highest priority source by software means. In this method there is one common branch address for all interrupts. The common service program begins at the branch address and polls the interrupt sources in sequence. The order in which they are tested determines the priority of each interrupt request.

Daisy chain priority selection consists of a serial connection of all devices that request an interrupt from the processor. The device with the highest priority is placed in the first position followed by lower priority devices up to the device with the lowest priority which is placed last in the chain.

Explanation ----------(5 marks)

Diagrams for both ----------(4 marks)

SECTION 11.5 COMPUTER SYSTEM ARCHITECTURE MONO

PG 435-437 FIG 11-19

b. Explain superscalar, super pipelined and VLIW architecture. (9)

Answer:

To increase the performance by allowing several instructions to processed in parallel and multiple pipelined units are provided for instruction processing. Instructions can be issued simultaneously to each unit. The effective MIPS rate can be increased by replicating various processing circuits so that several instructions can be in the same processing phase. The super scalar architecture allows the execution of 2 or more successive instructions simultaneously in different pipelines. Thus the throughput of a superscalar processor is greater than that of a pipelined scalar processor by twice or more. A superscalar processor may use RISC or CISC architecture.

The goal of pipelining is to increase the productivity i.e the number of instructions executed per second. Pipelines improve the rate at which instructions can be executed as long as there are no dependencies. Super pipelining uses a lengthy pipeline with several stages operating at higher clock frequency with each stage performing a critical function.

VLIW architecture exploits the ILP in programs by simultaneously executing more than one basic instruction. The compiler translates high level language program in to basic operations that the processor can execute simultaneously. The compiler groups several operations into a very long instruction word. The processor supports multiple functional units to execute within one clock cycle. While executing the instruction the processor issues operations in parallel to the appropriate functional units. The VLIW processor fetches a very long instruction word for parallel execution.

ARCHITECTURE EXPLANATION----------(3 X 3 MARKS)

Q.7 a. Perform addition and subtraction on the operands. The format is (9)

sign	Excess 15 exponent	Fractional mantissa
1 bit	5bits	6 bits

The operands are A =

0	10001	011011

The operands are B =

0	01111	101010

Answer:

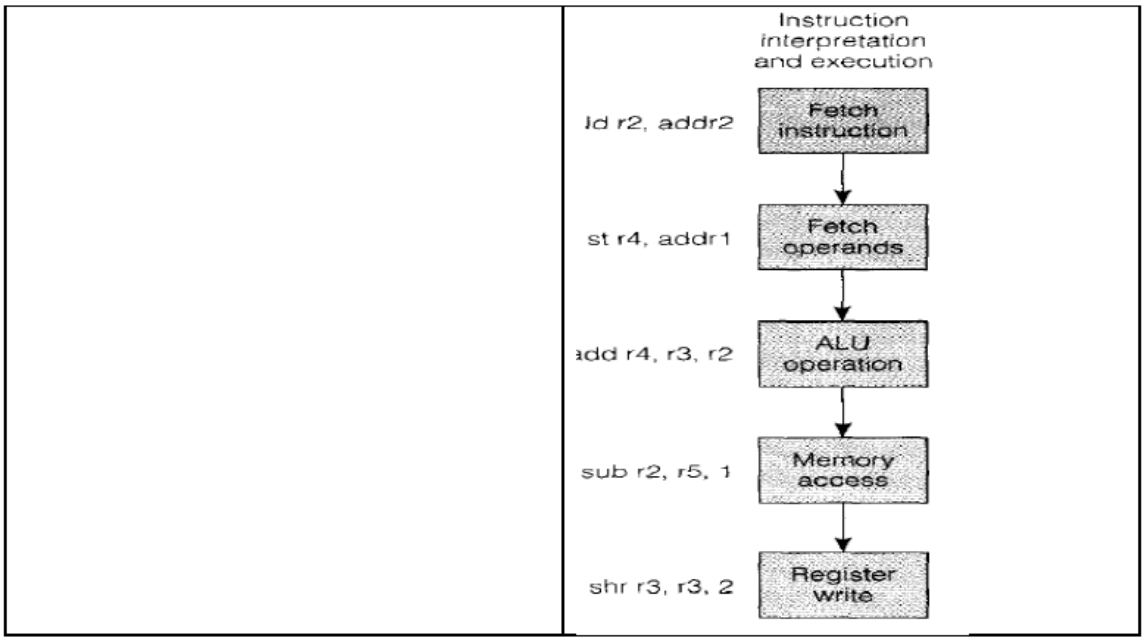
Exponent for A = 10001 =17
Actual exponent = 17-15 = 2
Exponent for B = 01111 =15
Actual exponent = 15-15 =0
Exponent calculation ----------(4 MARKS)
Therefore number B is smaller with difference 2.Hence its mantissa is shifted right by 2 bits.
Mantissa of B = 101010
Shifted mantissa = 001010
Add mantissa
A = 011011 + B = 001010 = Mantissa of result = 100101

Both numbers are positive therefore RESULT = 0 10001 100101
Mantissa calculation with final result in floating point format ----------(4 + 1 marks)

b. Differentiate parallel processing and pipeline processing. (9)

Answer:

parallel processing	pipeline processing
Parallel processing is used to denote simultaneous computations in the CPU for the purpose of increasing its computational speed. Instead of processing each instruction sequentially a parallel processor performs concurrent data processing tasks to achieve faster execution time. Hence also increase its thoroughput. The amount of processing that can be accomplished during a given time interval of time increases. The amount of hardware increases with parallel processing.	Pipeline is a decomposing a sequential process into sub process with each sub process being executed in a special dedicated segment that operates with all other segments. The result obtained in each segment is transferred to the next segment in the pipeline. The final result is obtained after it passes through all segments. Pipeline processor performs simultaneous operations in each segment. Once the pipe is full it takes only one clock pulse to obtain an output.
Parallel processing may occur in data stream or in instruction stream or both. If in a single instruction causing parallel execution of incoming data it is called SIMD or MISD which is a system with multiple instruction stream and single data stream. Parallel processing exhibited in both the instruction and data streams forming MIMD.	Any operation that can be decomposed into a sequence of sub operations can be implemented by a pipeline processor. Pipeline processing has been applied to floating point arithmetic operations.



DIAGRAMS FOR BOTH ----------(3 MARKS)
ANY 6 DIFFERENCES ----------(6 MARKS)

TEXT BOOK

I. Morris Mano-Computer System Architecture-PHI, Eastern Economy Edition-2001
II. John D.Carpinelli-Computer Systems Organization and Architecture-Pearson Education Asia-1st Edition