

**ACADEMY OF TECHNOLOGY
AEDCONAGAR, HOOGLY - 712121**



**Subject Name: Computer Organization
Subject Code: PCC-CS 302**

**Discipline: CSE/ IT
Semester: 3RD**

**GROUP-A
Multiple Choice Questions (each question carries 1
mark)**

1. A source program is usually in _____
a) Assembly language
b) Machine level language
c) High-level language
d) Natural language

2. Which memory device is generally made of semiconductors?
a) RAM
b) Hard-disk
c) Floppy disk
d) Cd disk

3. The ALU makes use of ____ to store the intermediate results.
a) Accumulators
b) Registers
c) Heap
d) Stack

4. The 8-bit encoding format used to store data in a computer is _____
a) ASCII
b) EBCDIC
c) ANCI
d) USCII

5. The ____ format is usually used to store data.
a) BCD
b) Decimal
c) Hexadecimal
d) Octal

6. The basic principle of Von Neumann computer is
(a) storing program and data in separate memory
(b) using pipeline concept
(c) storing both program and data in the same memory
(d) using a large number of registers

7. What is called the von Neumann bottleneck?
(a) CPU (b) Memory **(c) Single Bus** (d) ALU

- (a) Microprocessor's throughput
(b) Speed with which it performs I/O operations
(c) Time required to execute a basic instruction
(d) Time required to execute a small operation.

14. A ripple carry adder requires time.
(a) Constant time
(b) Linear time ($O(N)$)
(c) $O(\log(N))$
(d) $O(N \log(N))$

15. The largest integer that can be represented in signed-2's complement representation using n bits
(a) 2^{n-1}
(b) 2^n
(c) $2^{n-1}-1$
(d) $2^n - 1$

16. With 2's complement representation, the range of values that can be represented on the data bus of an 8-bit microprocessor is given by
(a) - 128 to + 127
(b) - 127 to +128
(c) - 128 + 128
(d) 0 - 255.

17. Equivalent hexadecimal of $(76575372)_8$ will be
(a) FAFAFF
(b) FAFAFA
(c) FFFAAA
(d) FAAFAF

18. An arithmetic shift-right
(a) multiplies a signed binary number by 2
(b) divides a signed binary number by 2
(c) multiplies a binary number by 2
(d) divides a binary number by 2

19. Subtractor can be implemented by using
(a) Adder
(b) Complement
(c) Both (a) and (b)
(d) none of these.

20. If a positive number is added to a negative number, then there is a possibility of
(a) Underflow
(b) Overflow
(c) Both a and b
(d) none of these.

21. Overflow occurs when
(a) data is out of range
(b) data is within range
(c) no output
(d) None of these

22. Booth's algorithm gives procedure for multiplying binary integers in
(a) Signed magnitude representation
(b) 2's complement representation
(c) Unsigned representation
(d) None of the above

23. The least negative value that the product of two 8-bit two's complement numbers can take is
(a) -2^{14}
(b) -2^{15}
(c) -2^{16}
(d) -2^{20}

(a)00

(b)01

(c)10

(d)11

37. Microoperation used to shift register contents without loss of information
is called (a) arithmetic shift
(b) logical shift (c) circular shift
(d) not available

38. An overflow occurs after an arithmetic shift left of n-bit register, if initially
(a) R_{n-1} is not equal to R_{n-2} (c) R_{n-1} is equal to 1
(b) R_{n-1} is equal to R_{n-2} (d) R_{n-1} is equal to 0

39. An arithmetic shift-left
(a) multiplies a signed binary number by 2
(b) divides a signed binary number by 2
(c) multiplies a binary number by 2
(d) divides a binary number by 2

40. The rule for Sign Extension in case of negative number is
(a) Move the sign bit to new left most position only
(b) Fill in with ones
(c) Fill in with zeros
(d) Fill in with copies of the sign bit

GROUP-B

Multiple Choice Questions (each question carries 2 marks)

1. The Sign Extension 16-bit representation of $-20 = 11101100$ (8-bit representation) is (a) 111111111101100 (c) 1111111100010100
(b) 1000000011101100 (d) 0000000011101100
2. A decimal number has 35 digits. Approximately how many digits would the binary representation have?
(a) 30 (b) 60 (c) 90 (d) 120
3. Multiplication of a pair of Floating-Point numbers $X = m_x * 2^a$ and $Y = m_y * 2^b$ is represented as
(a) $X * Y = (m_x * m_y) * 2^{a+b}$
(b) $X * Y = (m_x / m_y) * 2^{a+b}$
(c) $X * Y = (m_x * m_y) * 2^{a-b}$
(d) $X * Y = (m_x / m_y) * 2^{a-b}$
4. If the multiplier Q is 0011 (+3 in decimal) and 4-bit registers are in use, then in Booth's technique for multiplication, specify the cycle(s) where only right shift are required
(a) 1st cycle & 2nd cycle (c) 1st cycle & 3rd cycle
(b) 2nd cycle & 4th cycle (d) 4th cycle only

5. A general algorithm for multiplication of Floating-Point numbers consists of three basic steps:

- I. Compute the exponent of the product by adding the exponents together.
II. Normalize and round the final product.
III. Multiply the two mantissas.

The order of occurrences of the above three steps is

- (a) I, II, III (b) I, III, II (c) II, III, I (d) II, I, III

6. In the following statement

$$P + Q: R1 \leftarrow R2 + R3, R4 \leftarrow R5 \vee R6$$

the + between R2 and R3 and + between R5 and R6 indicate

- (a) OR operation for both the cases (c) add microoperation for both the cases
(b) OR operation & add microoperation (d) add microoperation & OR operation

7. If two numbers are added, and they are both positive or both negative, the overflow occurs if and only if

- (a) there is a carry (c) there is no carry
(b) result has opposite sign (d) result has same sign

8. A general algorithm for division of Floating-Point numbers consists of three basic steps:

- I. Divide the mantissa and determine the sign of the result.
II. Normalize and round the resulting value, if necessary.
III. Compute the exponent of the result by subtracting the exponents.

The order of occurrences of the above three steps is

- (a) I, II, III (b) I, III, II (c) II, III, I (d) III, I, II

9. The following numerical example clarifies the

microoperation: 1010 A (before)

1100 B (logic operand)

1110 A (after)

- (a) Selective-Complement (c) selective-set
(b) selective-clear (d) Mask

10. The result of division of the two Floating Point numbers X = 1.0000 * 2⁻² and Y = - 1.0100 * 2⁻¹ is

- (a) 0.1100 * 2⁻² (c) -1.0010 * 2⁻³
(b) -0.1101 * 2⁻¹ (d) 1.0110 * 2⁻¹

11. In the following options which one is an example of Overflow

- condition: (a) 0010+1001 (c) 1010+1100
(b) 1100+0100 (d) 1011+1110

12. In non-restoring algorithm of Binary Division, either addition or subtraction is performed when

- (a) LSB of dividend becomes 0 & 1, respectively
(b) LSB of dividend becomes 1 & 0, respectively
(c) MSB of dividend becomes 0 & 1, respectively
(d) MSB of dividend becomes 1 & 0, respectively

Question Bank (MCQ PART-II)

Academy of Technology

COMPUTER SCIENCE & ENGINEERING AND INFORMATION TECHNOLOGY

Sub: Computer Organization (PCC CS 302)

1. When generating physical addresses from a logical address the offset is stored in _____
a) Translation look-aside buffer
b) Relocation register
c) Page table
d) Shift register
2. In a system, which has 32 registers the register id is _____ long.
a) 16 bit
b) 8 bits
c) 5 bits
d) 6 bits
3. The instructions like MOV or ADD are called as _____
a) OP-Code
b) Operators
c) Commands
d) None of the mentioned
4. The order in which the return addresses are generated and used is _____
a) LIFO
b) FIFO
c) Random
d) Highest priority
5. In memory-mapped I/O _____
a) The I/O devices and the memory share the same address space
b) The I/O devices have a separate address space
c) The memory and I/O devices have an associated address space
d) A part of the memory is specifically set aside for the I/O operation
6. The advantage of I/O mapped devices to memory mapped is _____
a) The former offers faster transfer of data
b) The devices connected using I/O mapping have a bigger buffer space
c) The devices have to deal with fewer address lines
d) No advantage as such
7. The method which offers higher speeds of I/O transfers is _____
a) Interrupts
b) Memory mapping
c) Program-controlled I/O
d) DMA
8. The signal sent to the device from the processor to the device after receiving an interrupt is _____
a) Interrupt-acknowledge
b) Return signal

- c) Service signal
 - d) Permission signal
9. An interrupt that can be temporarily ignored is _____
- a) Vectored interrupt
 - b) Non-maskable interrupt
 - c) Maskable interrupt**
 - d) High priority interrupt
10. In the case of, Zero-address instruction method the operands are stored in _____
- a) Registers
 - b) Accumulators
 - c) Push down stack**
 - d) Cache
11. The addressing mode which makes use of in-direction pointers is _____
- a) Indirect addressing mode**
 - b) Index addressing mode
 - c) Relative addressing mode
 - d) Offset addressing mode
12. The addressing mode, where you directly specify the operand value is _____
- a) Immediate**
 - b) Direct
 - c) Definite
 - d) Relative
13. Whenever the data is found in the cache memory it is called as _____
- a) HIT**
 - b) MISS
 - c) FOUND
 - d) ERROR
14. LRU stands for _____
- a) Low Rate Usage
 - b) Least Rate Usage
 - c) Least Recently Used**
 - d) Low Required Usage
15. In _____ mapping, the data can be mapped anywhere in the Cache Memory.
- a) Associative**
 - b) Direct
 - c) Set Associative
 - d) Indirect
16. The transfer between CPU and Cache is _____
- a) Block transfer
 - b) Word transfer**
 - c) Set transfer
 - d) Associative transfer
17. What is the high speed memory between the main memory and the CPU called?
- a) Register Memory
 - b) Cache Memory**
 - c) Storage Memory
 - d) Virtual Memory

18. CISC stands for _____
- a) Complex Information Sensed CPU
 - b) Complex Instruction Set Computer**
 - c) Complex Intelligence Sensed CPU
 - d) Complex Instruction Set CPU
19. Which of the following is the fastest means of memory access for CPU?
- a) Registers**
 - b) Cache
 - c) Main memory
 - d) Virtual Memory
20. The reason for the implementation of the cache memory is _____
- a) To increase the internal memory of the system
 - b) The difference in speeds of operation of the processor and memory**
 - c) To reduce the memory access and cycle time
 - d) All of the mentioned
21. The effectiveness of the cache memory is based on the property of _____
- a) Locality of reference**
 - b) Memory localisation
 - c) Memory size
 - d) None of the mentioned
22. The temporal aspect of the locality of reference means _____
- a) That the recently executed instruction won't be executed soon
 - b) That the recently executed instruction is temporarily not referenced
 - c) That the recently executed instruction will be executed soon again**
 - d) None of the mentioned
23. The spatial aspect of the locality of reference means _____
- a) That the recently executed instruction is executed again next
 - b) That the recently executed won't be executed again
 - c) That the instruction executed will be executed at a later time
 - d) That the instruction in close proximity of the instruction executed will be executed in future**
24. The correspondence between the main memory blocks and those in the cache is given by _____
- a) Hash function
 - b) Mapping function**
 - c) Locale function
 - d) Assign function
25. The algorithm to remove and place new contents into the cache is called _____
- a) Replacement algorithm**
 - b) Renewal algorithm
 - c) Updation
 - d) None of the mentioned
26. The bit used to signify that the cache location is updated is _____
- a) Dirty bit**
 - b) Update bit
 - c) Reference bit
 - d) Flag bit

27. The write-through procedure is used _____
a) To write onto the memory directly
b) To write and read from memory simultaneously
c) To write directly on the memory and the cache simultaneously
d) None of the mentioned
28. The approach where the memory contents are transferred directly to the processor from the memory is called _____
a) Read-later
b) Read-through
c) Early-start
d) None of the mentioned
29. During a write operation if the required block is not present in the cache then _____ occurs.
a) Write latency
b) Write hit
c) Write delay
d) Write miss
30. In _____ protocol the information is directly written into the main memory.
a) Write through
b) Write back
c) Write first
d) None of the mentioned
31. The technique of searching for a block by going through all the tags is _____
a) Linear search
b) Binary search
c) Associative search
d) None of the mentioned
32. The number successful accesses to memory stated as a fraction is called as _____
a) Hit rate
b) Miss rate
c) Success rate
d) Access rate
33. The number failed attempts to access memory, stated in the form of a fraction is called as _____
a) Hit rate
b) Miss rate
c) Failure rate
d) Delay rate
34. If hit rates are well below 0.8, then they're called as speedy computers.
a) True
b) False
35. The extra time needed to bring the data into memory in case of a miss is called as _____
a) Delay
b) Propagation time
c) Miss penalty
d) None of the mentioned
36. SIMD represents an organization that _____.
a) refers to a computer system capable of processing several programs at the same time.

- b) represents organization of single computer containing a control unit, processor unit and a memory unit.
- c) includes many processing units under the supervision of a common control unit
- d) none of the above.
37. When an instruction is read from the memory, it is called
- a) Memory Read cycle
- b) Fetch cycle
- c) Instruction cycle
- d) Memory write cycle
38. _____ register keeps track of the instructions stored in program stored in memory.
- a) AR (Address Register)
- b) XR (Index Register)
- c) PC (Program Counter)
- d) AC (Accumulator)
39. The DMA transfers are performed by a control circuit called as _____
- a) Device interface
- b) DMA controller
- c) Data controller
- d) Overlooker
40. The technique where the controller is given complete access to main memory is _____
- a) Cycle stealing
- b) Memory stealing
- c) Memory Con
- d) Burst mode
41. Which representation is most efficient to perform arithmetic operations on the numbers?
- a) Sign-magnitude
- b) 1's complement
- c) 2'S complement
- d) None of the mentioned

Group-B

Considerer the following statement upto question 8

Non pipelined system takes 100 ns to process an instruction. A program of 1000 instructions is executed in non pipelined system. Then same program is processed with processor with 5 segment pipeline with clock cycle of 25 ns/stage.

1. The total time of execution in pipeline processor is
- a) 15050ns
- b) 25100ns
- c) 30000ns
- d) None of these
2. For the above statement the speedup ration of the pipeline processor is.....

- a) 3.984
 - b) 4.337
 - c) 4.136
 - d) 4.373
3. MIPS of the no pipeline processor is
- a) 40
 - b) 30
 - c) 20
 - d) 10**
4. MIPS of the pipeline processor is
- a) 50
 - b) 39.84**
 - c) 60
 - d) 45
5. Efficiency of the pipeline processor is....
- a) 0.99**
 - b) 0.75
 - c) 0.89
 - d) 1.02
6. For the abovementioned statements if the time required to process the five stages of pipeline are respectively 15ns, 20ns, 20ns, 25ns and 30ns. (Consider this statement upto question no) then the time to execute 1000 tasks in pipeline processor is
- a) 30130
 - b) 30120**
 - c) 40120
 - d) 35120
7. MIPS of the pipeline processor is
- e) 43.20
 - f) 33.20**
 - g) 23.20
 - h) None of these
8. Efficiency of the pipeline processor is
- a) 0.99**
 - b) 0.75
 - c) 0.89
 - d) 1.02
9. A block-set associative cache memory consists of 128 blocks divided into four block sets . The main memory consists of 16,384 blocks and each block contains 256 eight bit words. Consider the system is byte addressable. (Consider this statement upto question number 12).
- How many bits are required for addressing the main memory?
- a) 22**
 - b) 25
 - c) 19
 - d) 32
10. How many bits are needed to represent the TAG.

- a) 9
- b) 11
- c) 8
- d) 13

11. How many bits are needed to represent the SET.

- a) 5
- b) 4
- c) 6
- d) 8

12. How many bits are needed to represent the WORD.

- a) 8
- b) 10
- c) 7
- d) 11

13. Calculate the average access time if cache memory access time is 130ns and memory access time is 900 ns and we have the hit ratio for cache memory as $h=0.9$.

- a) 110 ns
- b) 220ns
- c) 130ns
- d) 140ns

14. For question no. 13 what is the efficiency of the memory system?

- a) 0.75
- b) 0.85
- c) 0.90
- d) 0.65

15. If the average access time is 330 ns and if cache memory access time is 150ns and memory access time is 900 ns then hit ratio for cache memory is?

- a).95
- b)0.96
- c)0.80
- d)0.90

16. If the average access time is 300 ns and if cache memory access time is 150ns then find the memory access time if hit ratio for cache memory is 0.8.

- a)850 ns 
- b)780ns
- c)1000ns
- d)880 ns

17. If the average access time is 256 ns and if cache memory access time is 160ns and the memory access time is 960ns then miss ratio for cache memory is?

- a)0.9
- b)0.8
- c)0.2
- d)0.1

18. A disk pack has 19 surfaces. Storage area of each surface has an inner diameter of 22cm and outer diameter of 33 cm. Maximum storage density on any track is 2000bits/cm and minimum spacing between tracks is 0.25mm.(Consider this statement upto question number 24.) Minimum storage density on any track is ?
- a) 1333.33
 - b) 1313.33
 - c) 1133.13
 - d) 1311.13
19. The number of track per surface is.....
- a) 210
 - b) 230
 - c) 220
 - d) 240
20. Storage capacity of each surface is....
- a) 34100.6Kbits
 - b) 33000.6Kbits
 - c) 30410.6Kbits
 - d) None of these
21. Storage capacity per track is....
- a) 133.23Kbits
 - b) 113.23Kbits
 - c) 123.23Kbits
 - d) 138.23Kbits
22. Storage capacity of the pack is....
- a) 72.225Mbytes
 - b) 62.225Mbytes
 - c) 92.225Mbytes
 - d) None of these
23. Data Transfer in bytes per second at a rotational speed of 3600rpm?
- a) 8.29Mbits/sec
 - b) 9.29Mbits/sec
 - c) 7.29Mbits/sec
 - d) 6.29Mbits/sec
24. Data Transfer in bytes per second at a rotational speed of 3600rpm?
- a) 9.68Mbits/sec
 - b) 9.29Mbits/sec
 - c) 7.29Mbits/sec
 - d) 6.29Mbits/sec
25. To construct a RAM of size 1Kx8 using same size smaller RAM chips of each size 256x2 the arrangement will be.....
- a)2x8
 - b)8x2
 - c)4x4

- d) None of these
26. If each register is specified by 3-bits and instruction ADD, R1,R2, R3 is two byte long; then what is the length of co-code field?
- a) 6 bit
 - b) 8 bit
 - c) 7 bit
 - d) None of these
27. What is the maximum number of 0-address instructions if the instruction size is 32 bit and 10-bit address field?
- a) 2^{22}
 - b) 2^{10}
 - c) 2^{32}
 - d) 2^{12}
28. What is the maximum number of 1-address instructions if the instruction size is 32 bit and 10-bit address field?
- a) 2^{22}
 - b) 2^{10}
 - c) 2^{32}
 - d) 2^{12}
29. What is the maximum number of 1-address instructions if the instruction size is 32 bit and 10-bit address field?
- a) 2^{22}
 - b) 2^{10}
 - c) 2^{32}
 - d) 2^{12}
30. LOAD R1, C is address instruction and LOAD C is address instruction.
- a) One, one
 - b) One, two
 - c) Two, one
 - d) Two, two

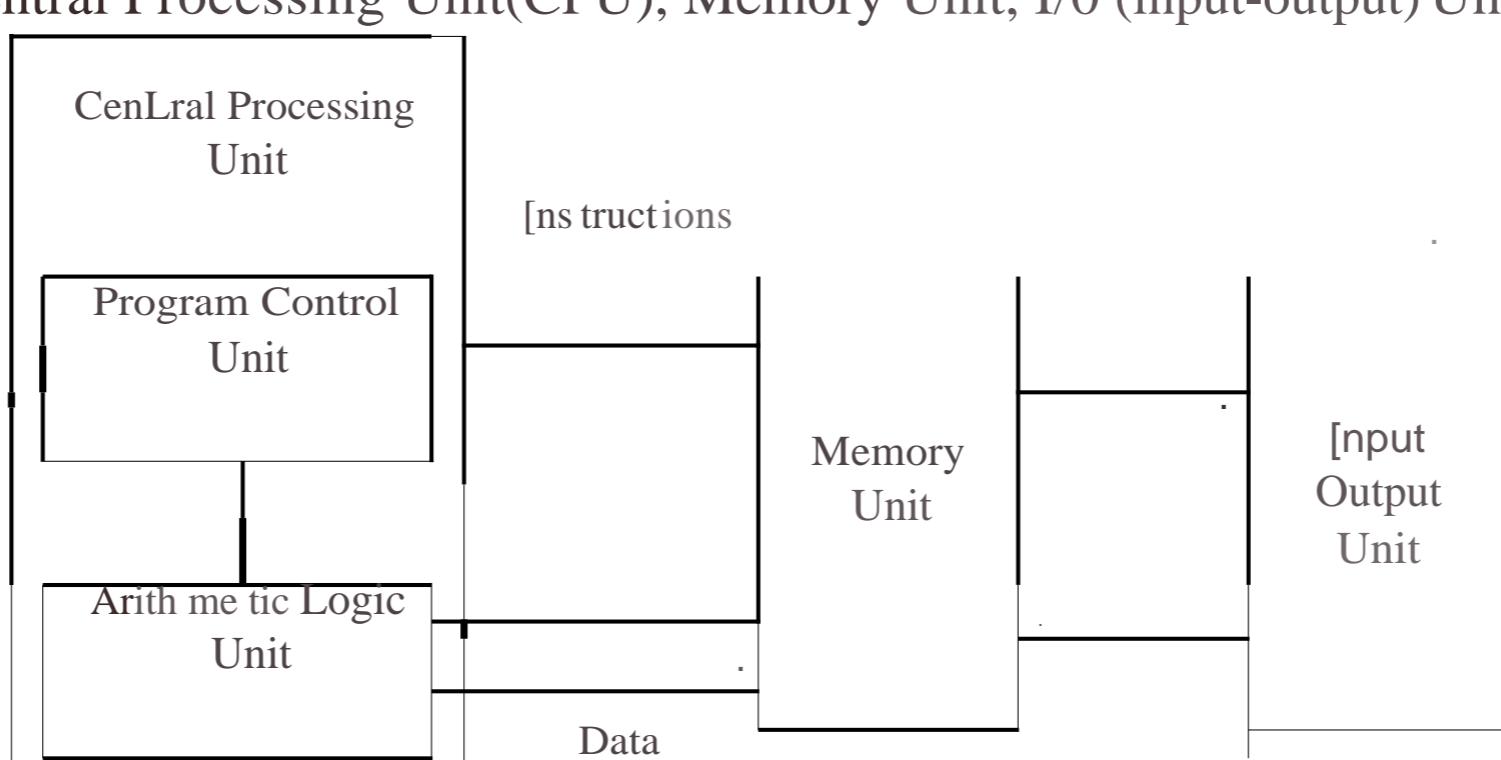
COMPUTER ORGANISATION

Introduction	2
Computer Arithmetic	11
Instruction Set	43
Memory Organization	61
Control Unit	103
Bus Structure	113
Input-Output Organization	122

INTRODUCTION

Chapter at a Glance

- Computer Organization is concerned with the way the hardware components operate and the way they are connected together to form the computer system. It refers to the operational units & their interconnections that realize the architectural specifications. Organization is basically the designer view of the computer hardware i.e. as a designer, one must know, how the different hardware elements are designed and implemented, how they are to be interconnected, how they operate. It basically deals with the in-depth detailed view of the computer hardware and also verifies whether the computer parts do operate as intended.
- Computer Architecture is the study of the structure and behaviour of the various functional modules of digital computers as seen by a programmer and also how they interact to provide the processing needs of the user. Architecture includes the instruction formats, instruction sets and addressing modes. It refers to those attributes of the system that have a direct impact on the logical execution of the program i.e. what are the basic hardware needed, what functions they do, what are the elements that are needed for the direct execution of the program etc. It is basically the higher-level or top-level functional view of the computer hardware. Architecture does not provide any information regarding the detailed implementation of the hardware elements.
- Parts of a digital computer: A digital computer consists of the following main parts:
Central Processing Unit(CPU), Memory Unit, I/O (input-output) Unit.



Block Diagram of a Digital computer

- Structure and Functions of the Different Units of a Digital Computer
 - (a) **Memory Unit:** Its purpose is to store both instructions & data. It is also called the Random-Access Memory(RAM) because the CPU can access any memory location at random.
 - (b) **CPU:** It is the brain of the computer and performs the bulk of data processing operations in computer. The two main units of a CPU are the Arithmetic Logic Unit and the Program Control Unit. The important parts of CP are:
 - i) **Arithmetic Logic Unit (ALU):** It performs instructions related to arithmetic operations like ADD, SUB, MUL etc. and logical operations like AND, OR, NOT etc.

(i) Program control unit : It interprets & sequences instructions i.e., interprets & sequences which instruction in a program is to be executed first.

(ii) Registers: These are collections of registers that store data.

(c) I/O Unit: This unit provides an efficient mode of communication between the central system (computer) & the outside environment. Through the I/O unit, programs & data must be entered into computer memory for processing & results obtained from computations must be recorded or displayed to the user.

- Operating System :

An Operating System is a program (or system software) that acts as an intermediary between a user of a computer & the computer hardware. Its purpose is to provide an environment in which a *user* can execute programs conveniently. So, an O.S. helps to use the computer hardware in an efficient manner.

- Functions of an Operating system: O.S. has the following functions.

(a) O.S. coordinates the efficient use of the hardware:

Operating System controls & coordinates the use of the hardware among the various application programs (like compilers, database systems, games etc.) for the various users (i.e. people, machines, and other computers).

(b) O.S. provides an environment within which user programs can do useful work:

Operating System provides the means for the proper use of the resources (like hardware, software & data) of a computer system in the meaningful & smooth operation of the computer.

(c) O.S. acts as a resource allocator:

O.S. manages the various resources (hardware and software) of a computer system & allocates them to specific programs & users as necessary for their tasks.

(d) O.S. acts as a control program:

As a control program O.S. focuses on the need to control the operations of the various input-output devices & user programs i.e. it controls the execution of user programs to prevent errors & improper use of the computer.

- von Neumann concept:

Von Neumann proposed the idea, known as the stored-program concept, which deals with making the programming process easier by representing programs in a form such that they can be suitably stored in memory alongside the data. So, a computer could get its instructions by reading them from memory & also a program could be set or altered depending on the memory values. Thus Von Neumann introduced the key concept of stored programs (i.e. programs & their data were located in the same memory) in the first generation computers.

Von Neumann published the idea in 1945 while proposing a new computer the EDVAC (Electronic Discrete Variable Computer) and in 1946.

Multiple Choice Type Questions

1. The basic principle of the von Neumann computer is

[WBUT 2007]

- Storing program and data in separate memory
- Using pipelining concept
- Storing both program and data in the same memory
- Using a large number of registers

Answer: (c)

- 2 From a Source Code, a compiler can detect [WBUT 2010]
a) Run-time error
b) Logical errors
c) Syntax error
d) None of these
A11swcr: (c)
3. How many minimum, NAND gates are required to make a flip-flop? [WBUT 2010]
a) 4
b) 3
c) 2
d) 5
A11swc1: (c)
4. The basic principle of a Von Neumann computer is [WBUT 2013]
a) storing program and data in separate memory
b) using pipeline concept
c) storing both program and data in the same memory
d) using a large number of register
Answer: (c)
6. The logic circuit In ALU is [WBUT 2013]
a) entirely combinational
b) combinational cum sequential
c) entirely sequential
d) none of these
Answer: (a)
6. The Von...Neumann bottleneck Is a problem, which occurs due to [WBUT 2014]
a) small size main memory
b) speed disparity between CPU and main memory
c) high speed CPU
d) malfunctioning of any unit in CPU
Answer: (b)
7. The circuit used to store one bit of data is known as [WBUT 2016]
a) Register
b) Encoder
c) Decoder
d) Flip-flop
Answer: (d)
8. SIMD represents an organization that [WBUT 2016]
a) refers to a computer system capable of processing several programs at the same time
b) represents organization of single computer containing a control unit, processor unit and a memory unit
c) includes many processing units under the supervision of a common control unit
d) none of these
Answer: (c)
9. The ALU makes use of to store the intermediate results. [WBUT 2017]
a) Accumulators
b) Registers
c) Heap
d) Stack
Answer: (a)

10. A source program is usually in

[WBUT 2019]

- a) Assembly language
- b) High-level language
- c) Natural language
- d) Machine level language

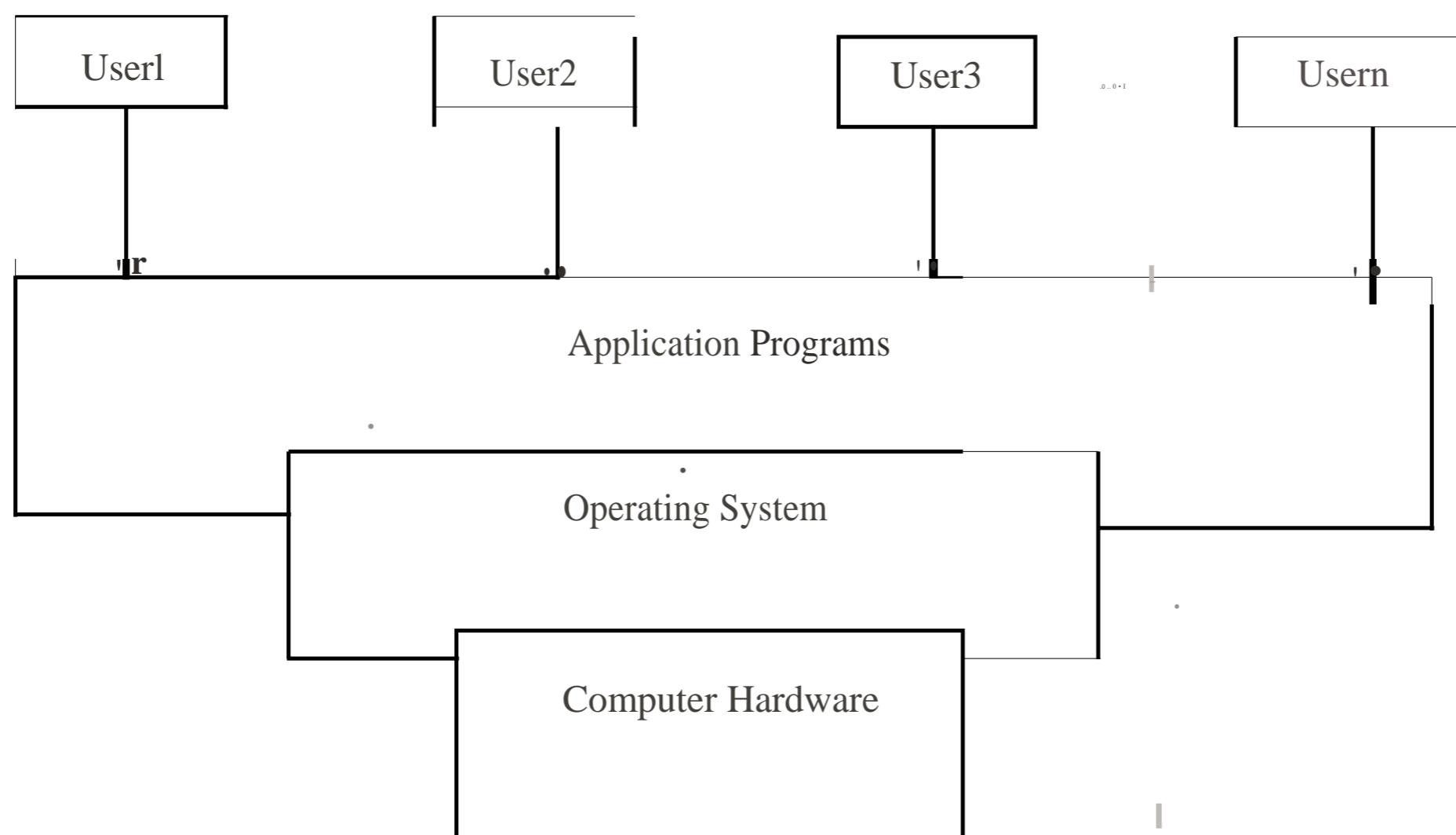
n , r: (c)

- b) Machine level language
- c) Natural language

Short Answer Questions : J

1. What is the role of operating system? [WBUT 2002, 2003 2005 2006 2008 2011]
In order to use computer hardware in an efficient manner each computer must have an operating system in it. An Operating System is a program (can also be considered as a system software) that acts as an intermediary between a user of a computer & the computer hardware.

Block Diagram



Abstract view of the components of an Operating System

O.S. has the following functions.

(a) **O.S. coordinates the efficient use of the hardware**

Operating System controls & coordinates the use of the hardware among the various application programs (like compilers, database systems, games etc.) for the various users (like people, machines, and other computers).

(b) **O.S. provides interface between user programs and devices**

Operating System provides the means for the proper use of the resources (like hardware & software & data) of a computer system in the meaningful & smooth operation of the computer.

Q) What is O.S.?

O.S. acts as a resource allocator

O.S. manages the various resources (hardware and software) of a computer system, allocates them to specific programs & users and uses them for their tasks.

POPULAR PUBLICATIONS

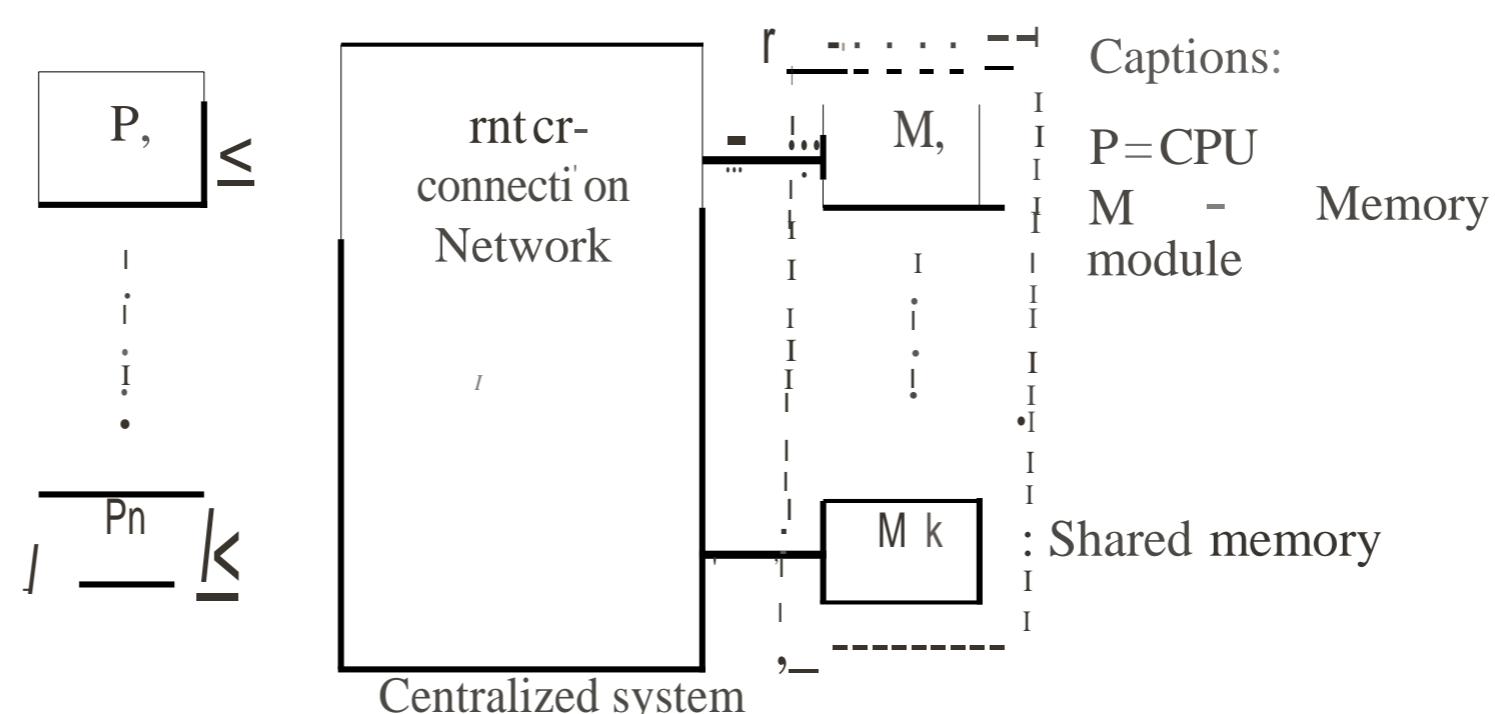
(d) O.S. acts a control program

As a control program O.S. focuses on the need to control the operations of the various input-output devices & user programs i.e. it controls the execution of user programs to prevent errors & improper use of the computer.

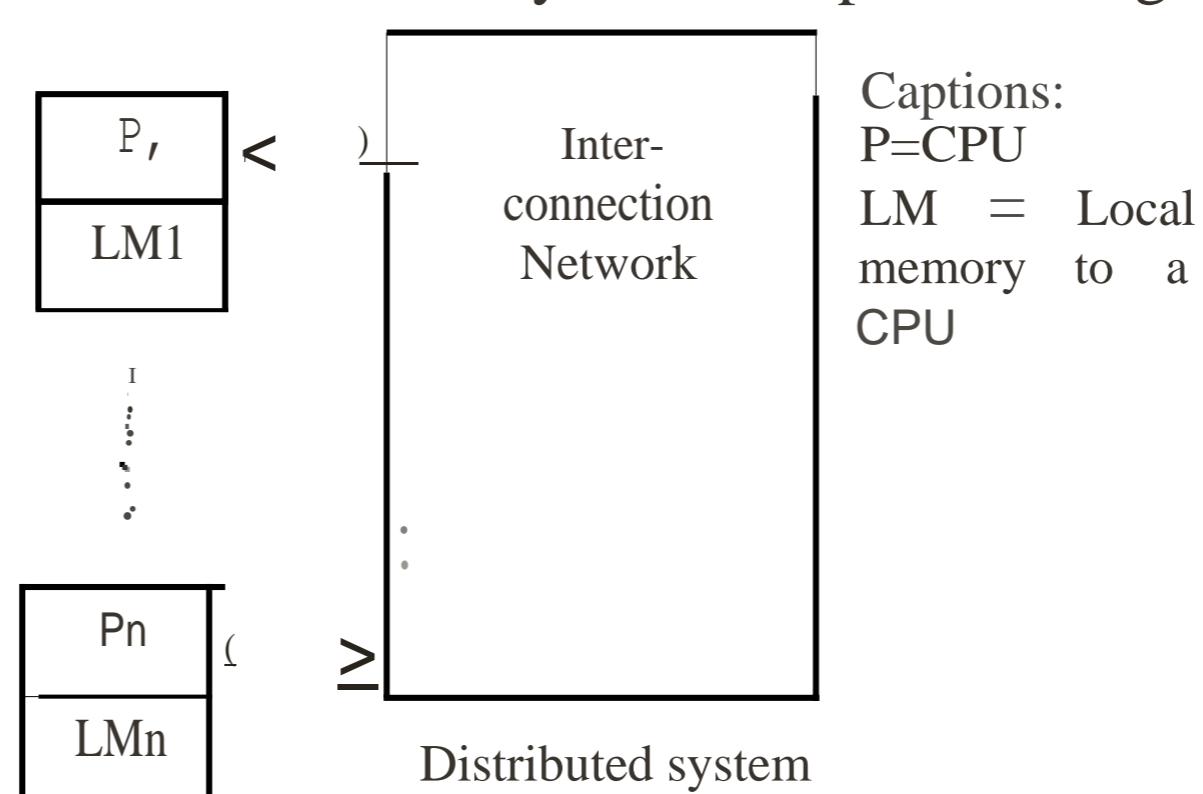
Q2 Compare between centralized and distributed architecture. Which is the best architecture among them and why? [WBUT 2014]

Answer:

In centralized architecture, all the processors access the physical main memory uniformly. All processors have equal access time to all memory words. The degree of interactions among tasks is high. Thus probability of bus conflicts is *high*, because of frequent sharing of codes between two processors. The architecture is shown in the following figure:



In distributed system, a local memory is attached with each processor. All local memories distributed throughout the system from a global shared memory accessible by all processors. A memory word access time varies with the location of the memory word in the shared memory. The degree of interactions among tasks is less. Thus probability of bus conflicts is also less. The distributed system is depicted in figure.



It is faster to access a local memory with a local processor. The access of remote memory attached to other processor takes longer due to the added delay through the interconnection network. Therefore, the distributed system is faster and in this regard it is better.

COMPUTER ARITHMETIC

(ii) Chapter at a Glance

Arithmetic Logic Unit: The Arithmetic Logic Unit or ALU performs arithmetic and logical operations on data in a digital computer. The various other units and devices of the computer system actually bring data into the ALU for its processing and then takes the results out from the ALU. The ALU operation is mainly based on the use of simple digital logic devices that store binary digits to perform various simple Boolean logic operations.

So to tell briefly, ALU is an important component in the CPU of the digital computer that performs various arithmetic and logical operations. ALU actually gets data and supplies the processed data from and to the various other units.

The various registers (temporary storage locations within the CPU) connected by signal paths to the ALU actually hold the data sent to the ALU for computations and also the results of the computations. The various flags (whose values are stored in CPU registers) are set by the ALU as the result of an operation. The operation of the ALU as well as the data movement into and out of the ALU is controlled by the signals provided by the control unit.

Diagrams:

Figure I, shows the ALU inputs and outputs, whereas figure 2, shows the CPU and its various components (control unit is not shown here).



Fig. I: ALU inputs and outputs

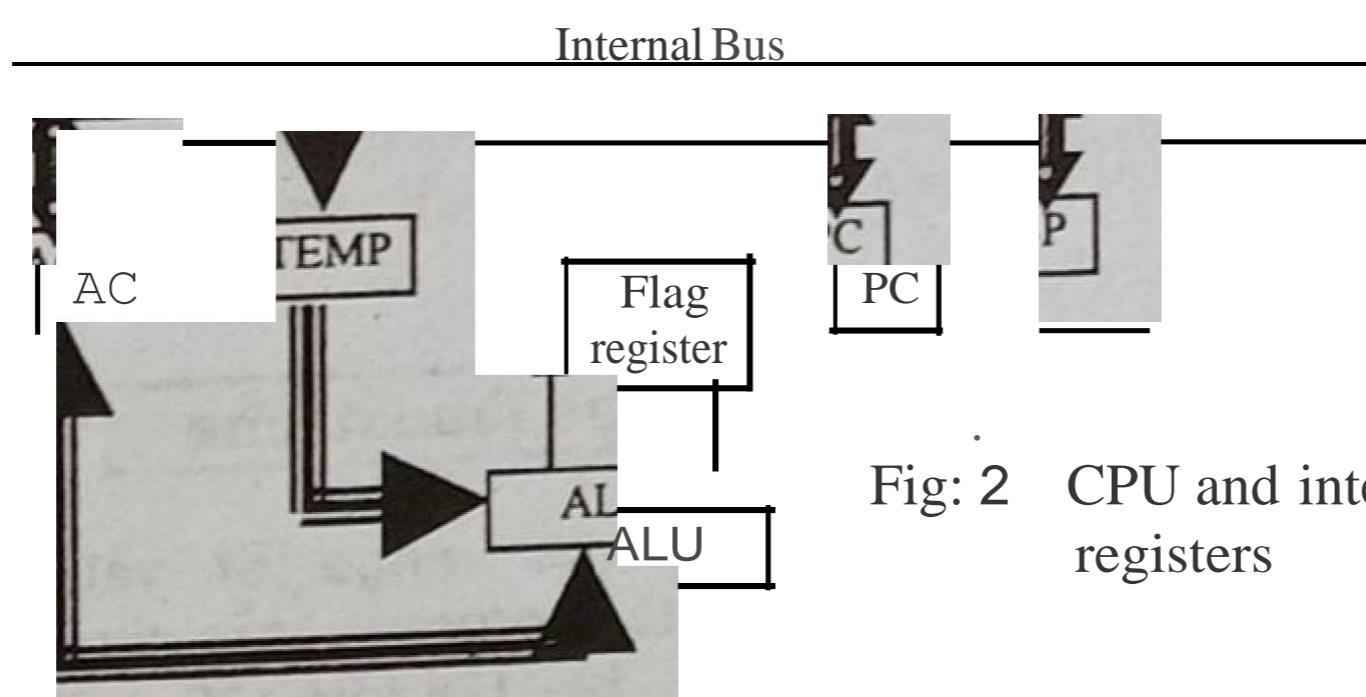


Fig. 2 CPU and internal registers

- **IEEE representation of floating point numbers:** All modern computers use the floating-point representation that was specified in IEEE standard 754. Here as was discussed before numbers are represented by a mantissa and an exponent.

Out of the multiple number of bit widths specified by IEEE standard 754 for floating-point numbers, single-precision and double-precision widths are the most commonly used widths. Figure 3 shows the two formats. Single-precision numbers are 32-bits long with 8-bits of exponent, 23-bits of fraction and 1 sign-bit, whereas double-precision numbers are 64-bits long with 11-bits of exponent, 52-bits of fraction and 1 sign-bit.

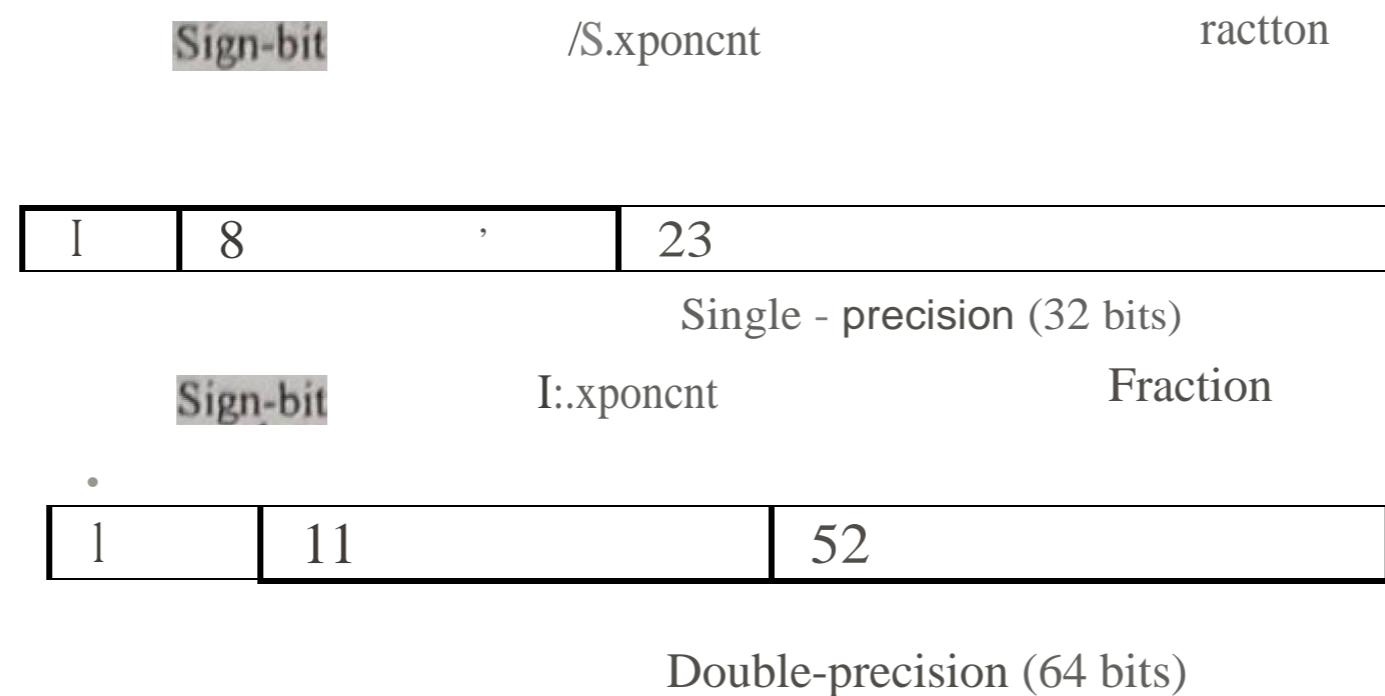


Fig: 3 J EE 754 Floating-pointformats

- Carry look-ahead adder:** This circuit basically speeds up the generation of carry signals. The logic expressions for sum (si) and carry-out (ci) of stage i are:

$$s_i = A_i \oplus B_i \oplus c_i, \text{ and } c_{i+1} = A_i B_i + A_i c_i + B_i c_i = A_i B_i + (A_i + B_i)c_i = G_i + P_i c_i$$

where $G_i = A_i B_i$ and $P_i = A_i + B_i$.

The expressions G_i and P_i are called the generate and propagate functions for stage i respectively. So if the generate function for stage i is equal to 1 i.e. if $G_i = 1$, then $c_{i+1} = 1$ (when both A_i and B_i are equal to 1).

The propagate function (P) means that an input carry will produce an output carry when either of A or B is 1. So all G and P functions can be formed independently and in parallel in only one logic gate delay. Now, if the propagate function can be realized as $P_i = A_i \oplus B_i$, then a simple circuit can be derived using a cascade of two 2-input XOR gates (to realize 3-input XOR function).
 - Booth's multiplication algorithm:** Booth's algorithm provides a procedure by which binary integers in signed-2's complement representation (i.e. multipliers can be positive or negative) can be multiplied.
 - Division Algorithm:** Division of two fixed-point binary numbers represented in signed-magnitude form is done by successive compare, shift and subtract technique. However in division, it may give rise to an overflow result i.e. if the expected quotient is of n-bits but the actual quotient comes as n+1 bits then that condition is an overflow condition, which must be taken care of.

| Multiple Choice Type Questions)

1. With 2's Complement representation, the range of values that can be represented on the data bus of an 8 bit micro-processor is given by:

 - a) -128 to + 127
 - b) -128 to + 128 [WBUT 2003, 2012]
 - c) -127 to 128
 - d) -256 to + 256

Answer: (a)

2. When signed numbers are used in binary arithmetic, then which one of the following notations would have unique representation for zero?

 - a) Sign Magnitude
 - b) 1's complement
 - c) 2's complement
 - d) None

[WBUT 2003, 2007₁, 2008₁, 2009]

An wcr: (<.:J

3. Subtractor can be implemented using
a) adder
b) complementer
c) both (a) & (b)
d) none of these [WBUT 2006, 2012, 2015]
Ans, ver: (c)
4. Maximum n bit 2's complement number is [WBUT 2007, 2009, 2015, 2018, 2019]
a) $2^n - 1$
b) $2^n + 1$
c) $2^{n-1} - 1$
d) Cannot be said
Ans, ver: (c)
5. Adding 011011012 to 10100010₂ in 8-bit 2's complement binary will cause an overflow:
a) True
b) False [WBUT 2007, 2009]
Answer: (b)
6. The conversion of (FAFAFA)₁₆ into octal form is [WBUT 2008, 2013]
a) 76767678
b) 76575372
c) 76737672
d) 76727672
Answer: (b)
7. Which logic gate has the highest speed?
a) ECL
b) TTL
c) RTL
d) DTL [WBUT 2009]
Ans, ver: (c)
8. Booth's algorithm for computer arithmetic is used for [WBUT 2009, 2011]
a) multiplication of numbers in sign magnitude form
b) multiplication of numbers in 2's complement form
c) division of numbers in sign magnitude form
d) division of numbers in 2's complement form
Answer: (b)
9. The conversion (FAFA.FB)₁₆ into octal form is [WBUT 2009]
a) 76767676
b) 76575372
c) 76737672
d) None of these
Answer: (d)
10. A decimal no. has 30 digits. Approximately, how many would the binary representation have? [WBUT 2009]
a) 30
b) 32
c) 60
d) 90
Answer: (d)
11. The logic circuit in ALU is [WBUT 2009]
a) Entirely combinational
b) Entirely sequential
c) Combinational cum sequential
d) None of these
Answer: (c)
12. Equivalent hexadecimal of (76575372)s will be [WBUT 2011]
a) FAFAFF
b) FAFAFA
c) FFFAAA
d) FAAFAF
Answer: (b)

POPULAR PUBLICATIONS

13. If you convert (+46.5) into a 24 bit floating point binary number following IEEE convention, what would be the exponent? [WBUT 2013]

- a) 00011100 b) 00000011 c) 1100010 d) none of these

Answer: (d)

14. The maximum number of additions and subtractions are required for which of the following multiplier numbers in Booth's algorithm {WBUT 2014, 2015}

- a) 010001111 b) 01111000 c) 0000 1111 d) 0101 0101

Answer: (d)

15. By logical left-shifting the content of a register once, its content is [WBUT 2015]

- a) doubled b) halved
c) both (a) and (b) d) no such decision can be made

Answer: (d)

16. Floating point representation is used to store [WBUT 2016]

- a) Boolean values b) Whole numbers
c) Real numbers d) Integers

Answer: (c)

17. A given memory chip has 12 address pins and 4 data pins. It has the number of locations. [WBUT 2016]

- a) 2^4 b) 2^{12} c) 2^{48} d) 2^{16}

Answer: (b)

18. (2FAOC)16 [WBUT 2016]

- a) (195084)10 b) (0010111101000001100)2
c) Both (a) and (b) d) None of these

Answer: (b)

19. In a normal n-bit adder) to find out if an overflow has occurred, we make use of [WBUT 2017]

- a) AND gate b) NAND gate c) NOR gate d) XOR gate

Answer: (d)

20. For which of the following multiplier numbers in Booth's algorithm maximum no. of additions and subtractions are required? [WBUT 2018]

- a) 01001111 b) 01111000 c) 00001111 d) 01010101

Answer: (c)

21. In straight binary code, N-bits or N binary digits can represent different values. [WBUT 2019]

- a) 2^N b) 2^{N+1} c) $2A(N-1)$ d) 2^{N-1}

Answer: (a)

INSTRUCTION SET

• Cl1apte1..at a Glance

- 11: (111 tio11 s t: l11slru tic.in • ct is the s•t o instril • lio,18 lhnl ll 1 in•lil u: ul>lt, l◊ ,,, r< 111c.
1n 11 ti,rti ttlmr 111 lltl1' It.!- ils \vil 8 • 1 Or inst uclion i,-, 1111 11,11uutlc 111 ..c:I, w11ith t<Ht'; 1 I•
Or ntl tl1c i11 l11lCli 0 lls d ill 111\1 !)llltic1lur (.)ll1(>Ul c1 vn i '/\ lrc1111 (.'1 1\1 1 111t:J \0 (.'()1111)11 1
de{ cnd11g 11 tl1c•JJ irl orgn11iJ'1tio11 nncl rcllitcclu1 of lhl. 01111>ut • 1..
[rt. tr11 ti 11 r r111nt: 111. lrucli 11 for-111•t clcnl. willt tl1, looks 0 ll l1n 1c i11 tru elien , 1 11t 1
in. t ru ction 11 tl1 • 1,urt s. 'f 11c • p ocl • ' pn1 t, lite lncl,lrcss 11! 111c)clc '1111 nnd Lil <>JJCl" H 1cl'i <>1
the • ddrc (j • • (J m11ct nddr s) pnrl. l'hc op rnction fiulcl is en11 • cl t11 • '0l>ccldt ' <Jr 1 Hc:
opcra lion ode'• 11 op r 11d/ndc.lrcss (i Ids conlni11 ill1cr the opc1• 11ds thc111s 1 vcn <>1 tl1c oddrcs
c of tor & lo tion of tl1c ,t 1 or r u111cnls (i.e. udclr cs. •s l' upcrl1ncls) in 111n11 1\1
niory or i11 tl1 proccs or dcpc11dir1g on thc vnricJ11S nclc.lr •s :i 113 1110d 11 (1) as sp • sj fi c1 in a
pnrticulnr instructio11.

• Tl1rcc, T\vo one nnd zero nddre instructlo1ls:

(i) 'f11rcc-naddr i11strt1 tion :

In the `c` type of instruction, all operand addresses are explicitly defined. The instruction format is as follows, specifying a memory or processor register operand run-length, index, and offset:

- (i) It suits in short program while evaluating mathematical expressions.
 - (ii) as condition in if.

(ii) T, o-address ill tructions:

Here the instruction format has two different address fields, each specifying either a memory location or a processor register operand.

JJ,alltage : Less execution time compared to onc address instruction.

(iii) One-address instructions:

such instruction format has a single explicit address field and uses an implied accumulator (AC) register for all data manipulation.

dva, tages:

- (i) Much less number of bits is required to specify the single operand address.
 - (ii) Less complicated decoding and processing circuit is needed.

(i) Zcro-addrcs Instructions:

Such instructions do not contain any explicit addresses (except for P II and P J> instructions). As the operands are stored in a pushdown stack (the operands required must be there in the top positions in the stack), hence no addresses are required.

Advantage. Do not contain any explicit addresses. Instructions are simple.

- Different types of addressing modes:

- (i) [mə] lied Mode

- #### (ii) Immediate Addressing Mode

- (iii) Register Mode or Register Direct Mode

- (iv)

- ## Register Indirect Mode

- #### (v) Auto-increment Mode

- (vi) Auto-decrement Mode

- #### (vii) Direct Address Mode

POPULAR PUBLICATIONS

- (iii) Indirect address mode
- (i) Relative address mode
 - (a) PC (Program Counter) Relative Addressing Mode
 - (b) Index addressing Mode or Index Register Relative Addressing mode
 - (c) Base Register Addressing Mode
 - (d) Lack Addressing mode

Multiple Choice Type 'A' Questions

1. Instruction cycle is [WBUT 2006, 2007, 2011, 2012, 2015, 2018]
a) fetch-decode-execution
b) decode-fetch-execution
c) fetch-execution-decode
d) none of these
Answer: (a)
2. Micro instructions are kept in [WBUT 2007, 2011, 2015, 2018]
a) Main memory
b) Control memory
c) Cache memory
d) None of these
Answer: (b)
3. Which of the following addressing modes is used in the instruction PUSH B? [WBUT 2008, 2011]
a) Immediate
b) Register
c) Direct
d) Register Indirect
Answer: (b)
4. Which of the following addressing modes is used in instruction RAL [WBUT 2012]
a) immediate b) implied c) direct d) register
Answer: (b)
5. Which of the following address modes is used in the instruction 'POP S'? [WBUT 2013]
a) immediate b) register c) direct d) register indirect
Answer: (d)
6. A computer uses words of size 32-bit. The instruction [NBUT 2014]
a) may or may not be one byte length
b) must always be fetched in one cycle with 2 bytes in the cycle
c) must always be fetched in two cycles with one byte in each cycle
d) must be of 2 bytes length
Answer: (c)
7. The CPI value for RISC processor is [WBUT 2014]
a) 1 b) 2 c) 3 d) none of these
Answer: (c)

- . In t1 roe or. tt1 ddr~~oss~~ of tt10 next instruction to be executed is stored in
a) stack pointer register b) index register [WSUT 0, S]
c) base register d) program counter register

Answer: (c)

- A stack-organized computer uses instruction of [WBUT 2016]

9. a) Indirect addressing
b) Two addressing
c) zero addressing
d) Index addressing

Answer: (c)

When performing a looping operation, the instruction gets stored in the [WBUT 2017]

10. a) Registers
b) Cache
c) System heap
d) System stack

nsw : (b)

In case of Zero-address instruction method the operands are stored in [WBUT 2017]

11. a) Registers
b) Accumulators
c) Stack
d) Cache

nsw : (c)

The addressing mode(s), which uses the PC instead of a general purpose register is [WBUT 2017]

12. a) Indexed with offset
b) Relative
c) Direct
d) Both (a) and (b)

n".."cr : (b)

How many memory locations can be addressed by a 32-bit computer? [WBUT 2018]

13. a) 64 KB
b) 32 KB
c) 4 GB
d) **4 MB**

Answer: (a)

The addressing mode of an instruction is resolved by [WBUT 2018]

14. a) ALU
b) OMA controller
c) CU
d) program

Answer: (b)

The addressing mode, where you directly specify the operand value is [WBUT 2019]

15. a) Immediate
b) Direct
c) Definite
d) Relative

Answer: (a)

How is the effective address of base register calculated? [WBUT 2019]

16. a) By addition of base register contents to the partial address in instruction
b) By addition of implied register contents to the partial address in instruction
c) By addition of base **register** contents to the complete address in instruction
d) By addition of implied **register** contents to the complete address in instruction

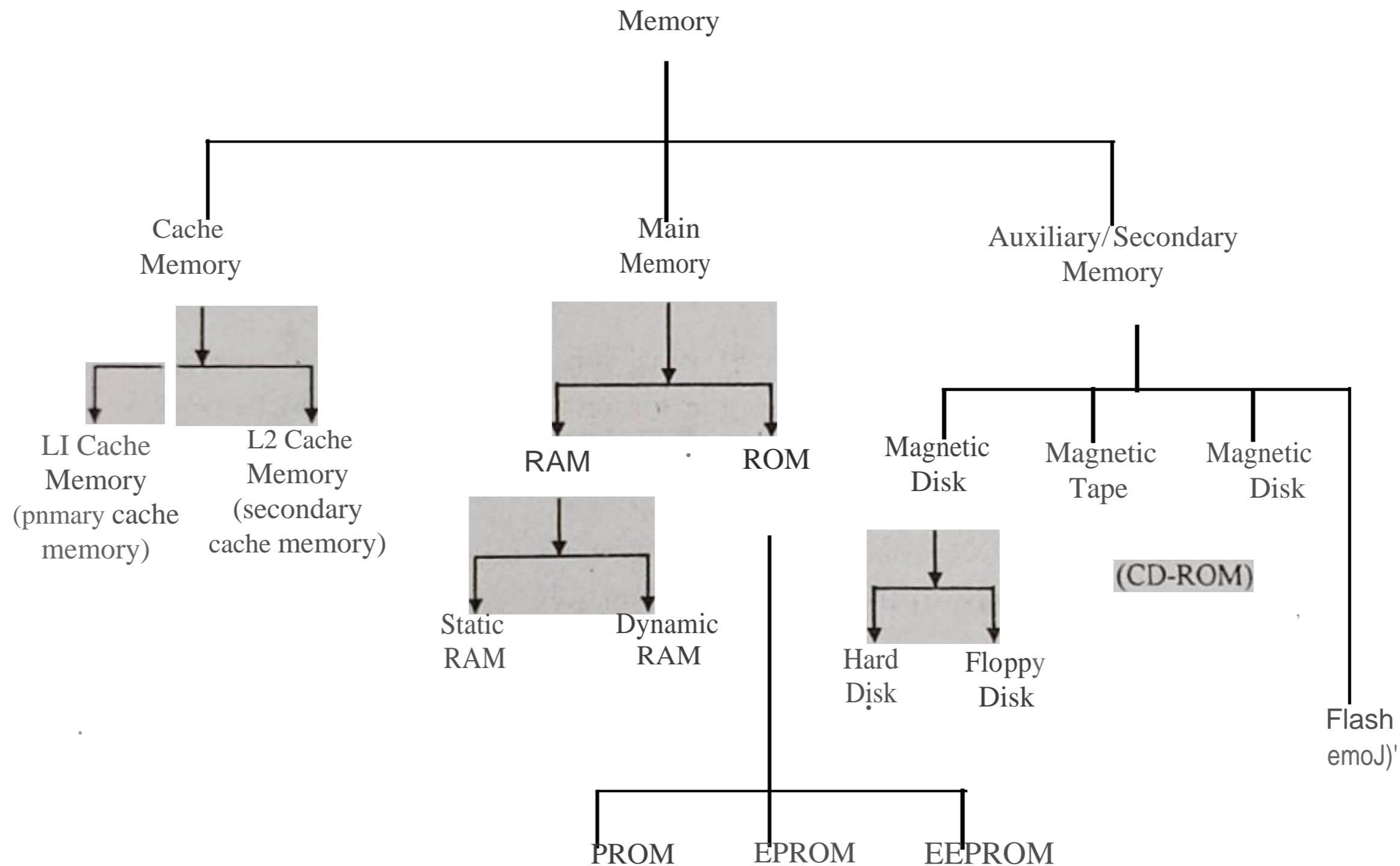
Answer: (a)

MEMORY ORGANIZATION

w Chapter at a Glance

- 1 'P of MO : MO can be of three following 3 types:
 - (a) **P 10** : The p-channel (+ve) MOS which depends on the flow of holes.
 - (b) **N10S**: NMOS (-ve) MOS which depends on the flow of electrons. These are commonly used in circuits with only one type of MOS transistor.
 - (c) **CMOS** : CMOS (Complementary MOS) technology uses both PMOS and NMOS transistors connected in a complementary fashion in all circuits.
- **Memory Classification and hierarchy:** Memory can be broadly classified into three main parts: the cache memory, the main memory and the auxiliary memory. The cache memory lies between the CPU and the main memory and is the fastest, smallest and the most expensive of all the memory units. The auxiliary or the secondary memory unit is the slowest, largest and least expensive of all the memory units. The main memory lies in between the two.

Memory Classification: Memory can be classified accordingly:



Memory Applications

In the block diagram of memory hierarchy is as follows:

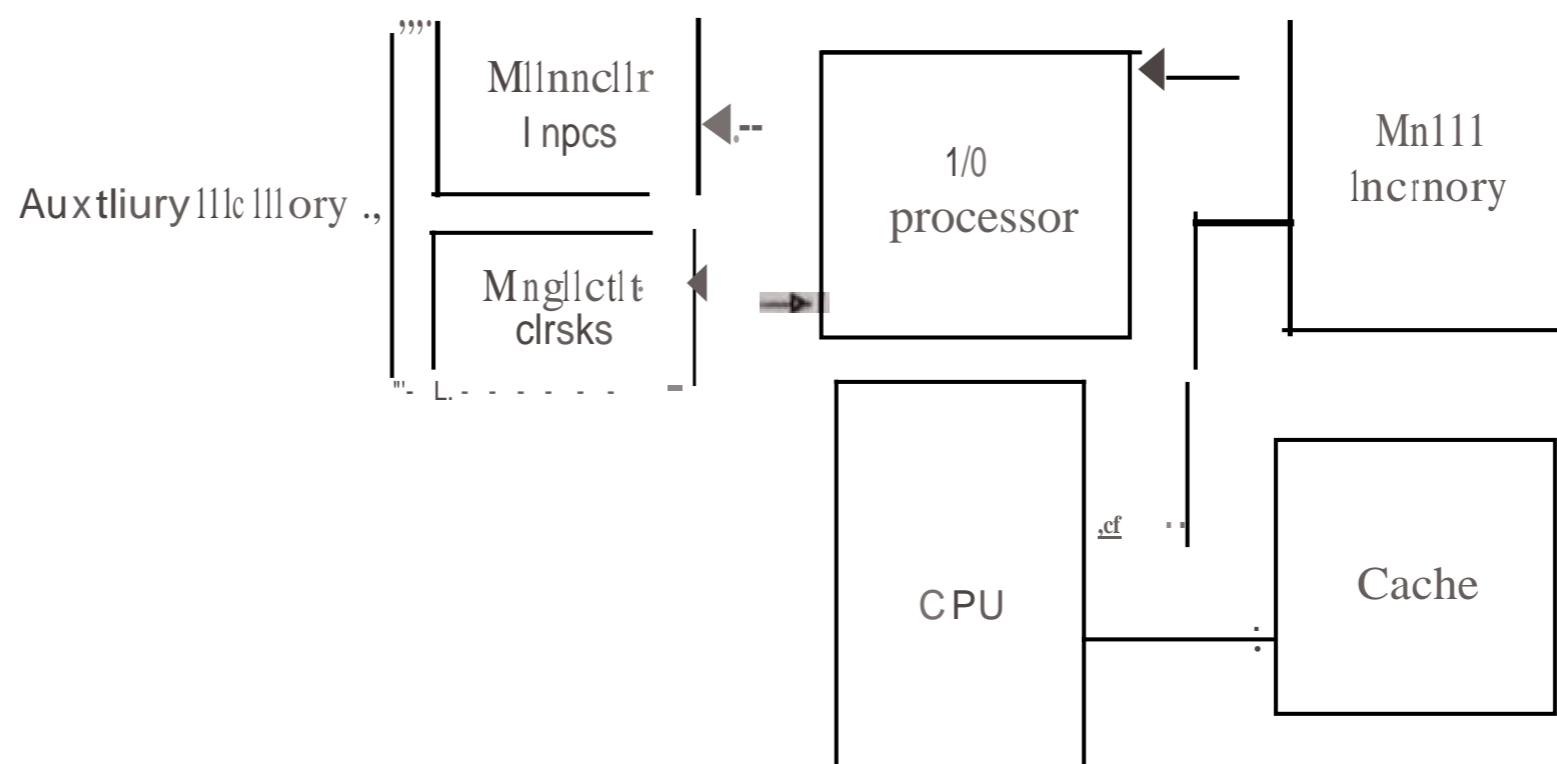
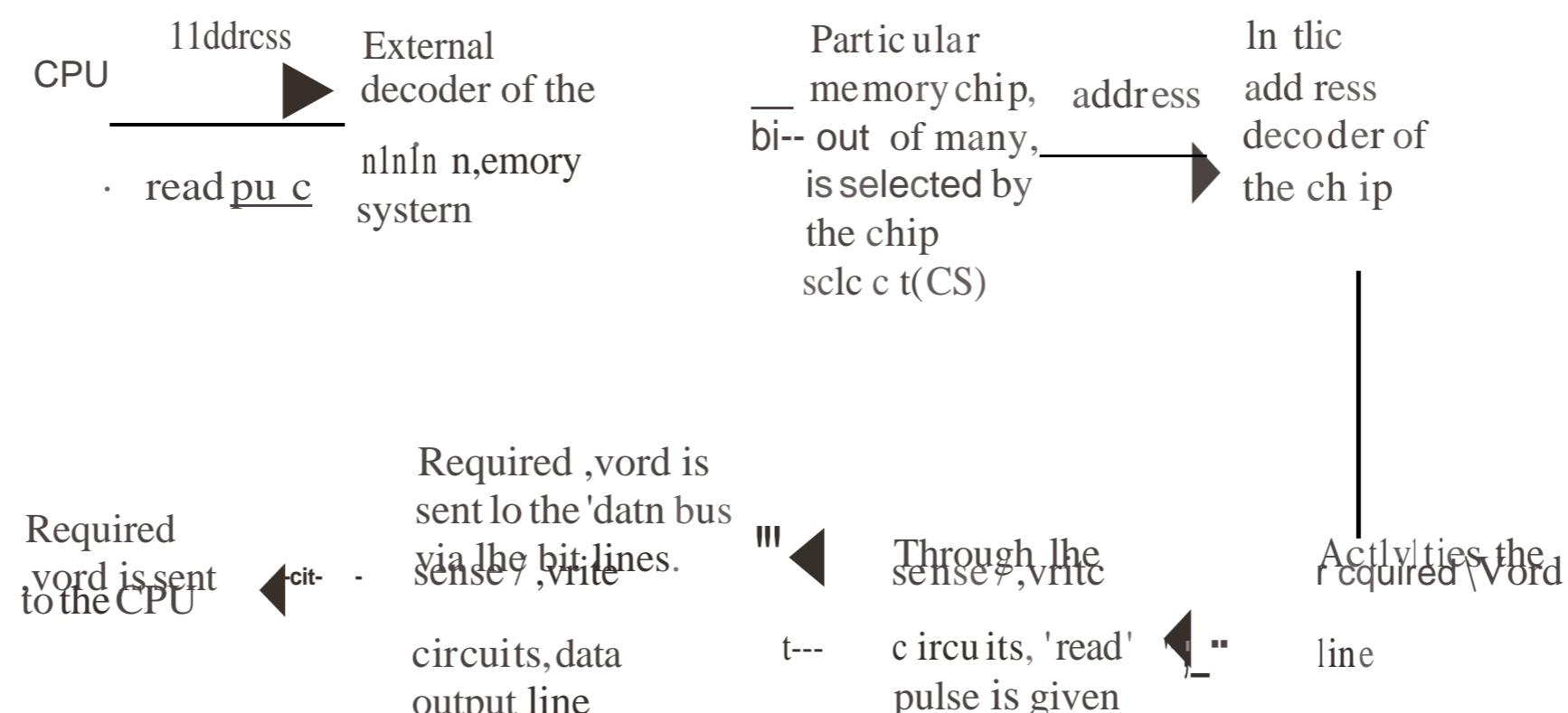


Fig: Memory hierarchy in a computer system

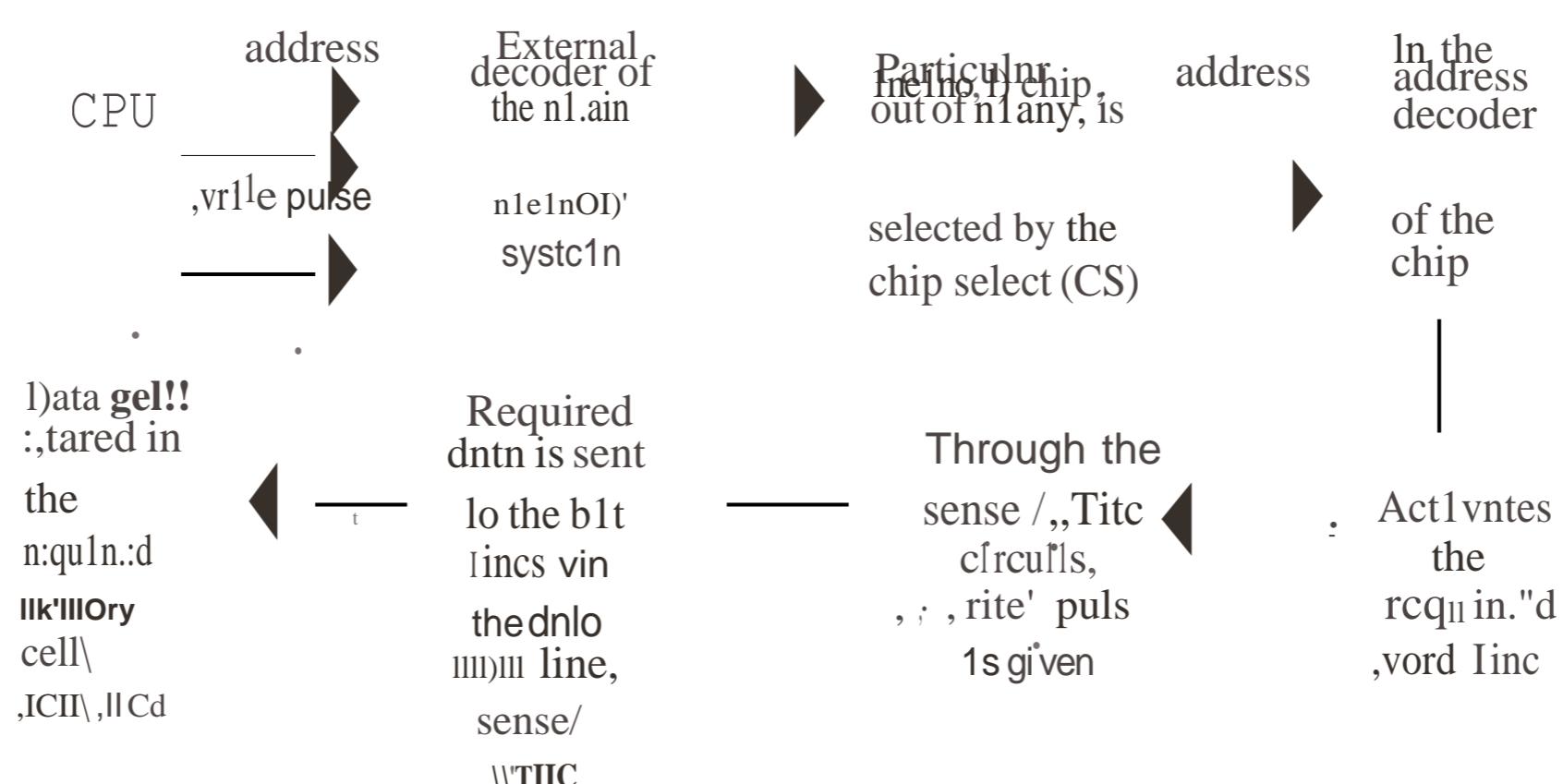
actors on which memory hierarchy depends: There are various factors on which the basic hierarchy of memory depends. These are cost, storage capacity (size), and speed and access time.

- Mention the read/write access:

Flowchart of read/write operation:



Flowchart of write operation:



- **t** **tic** . **D** ·na mic **RAM**:
(i) *tatic* *JI:*

It consists of internal fltp-fl.ops that store the binary information. Stored information remains, valid as long as power is applied to the unit.

ii., *D*,, *a* *iC RAi.t*: dynamic RA loses its stored information in a very short time (a few milliseconds even though the power supply is on).

ach m rnol")' Cache memory is a very small but very fast memory. t is the smallest but fastest among all other memory units. A very-high-speed memory, it Is sometimes used to increase the speed of processing by making current programs and data available to the CPU at a rapid rate. t lies behveen the CPU and the main memory unit



- Virtual memory: It is a technique that allows the execution of processes that may not be completely in main memory. This concept used in some large computer systems permit the user to construct programs as though a large memory space is available. Virtual memory is used to give programmers the illusion that they have a very large memory at their disposal, even though the computer actually has a relatively small main memory.

• *Diaaram:*

The below figure shows the organization that implements virtual memory.

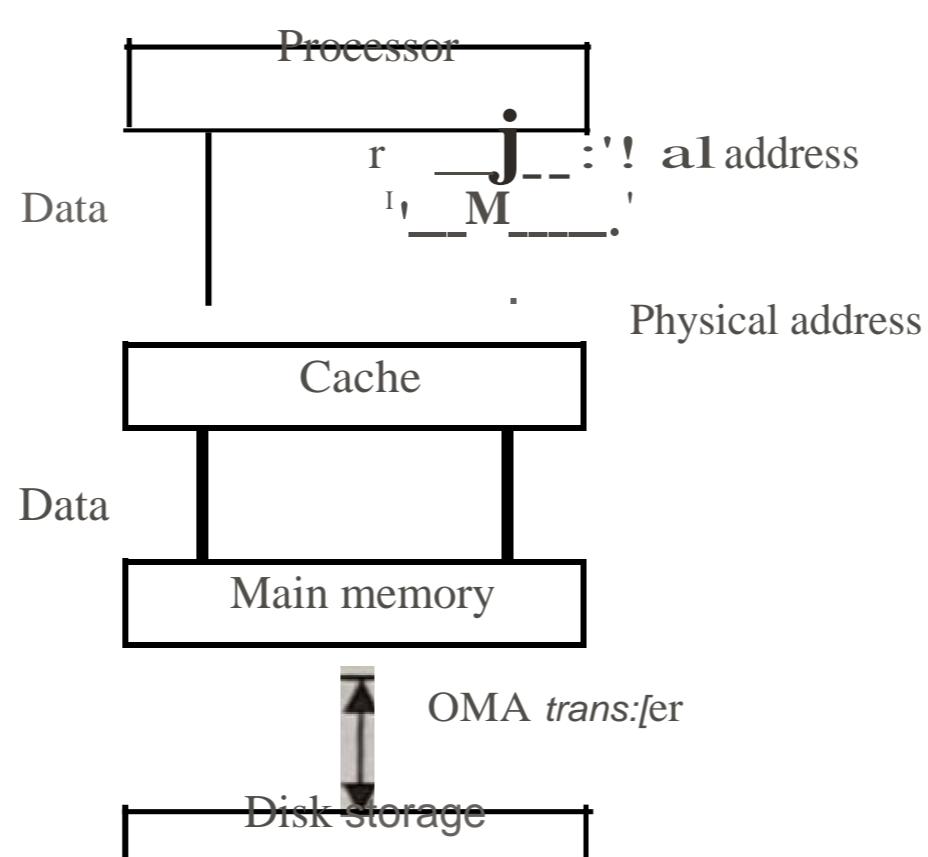


Fig: Virtual memory organization

Multiple Choice Type Questions

- The technique of placing software in a ROM semiconductor chip is called [WBUT 2003, 2008]

 - a) PROM
 - b) EPROM
 - c) FIRMWARE
 - d) Microprocessor

A..n "er: c

- 2 Cache memory [WBUT 2006, 2008, 2011, 2013]

 - a) increases performance
 - b) reduces performance
 - c) machine cycle increases
 - d) none of these

POPULAR PUBLICATIONS

3. Assertion is a
a) primary memory
b) pointer addressable memory
c) computer addressable memory
d) slow memory [WBUT 2006, 2008, 2011, 2013]
4. How many RAM chips of size (256 K x 1 bit) are required to build 1 M Byte memory?
a) 8 b) 10 c) 24 d) 32 [WBUT 2006, 2009, 2016, 2018]
Ans.: (b)
5. How many address bits are required for a 1024 x 8 memory?
[WBUT 2007, 2011, 2013]
a) 1024 b) 5 c) 10 d) None of these
Ans.: (c)
6. The principle of locality justified the use of
a) Interrupt b) Polling
c) OMA d) Cache memory [WBUT 2007, 2015, 2018]
Ans.: (d)
7. A major advantage of direct mapping of a cache is its simplicity. The main disadvantage of this organization is
[WBUT 2007]
a) It does not allow simultaneous access to the intended data and its tag
b) It is more expensive than other types of cache organizations
c) The cache hit ratio is degraded if two or more blocks used alternately map onto the same block frame in the cache
d) Its access time is greater than that of other cache organizations
Ans.: (c)
8. The purpose of ROM in a Computer System is [WBUT 2010]
a) to store constant data required for computer's own use
b) to help reading from memory
c) to store application program
d) to store OS in memory
Ans.: (a)
9. Which one does not possess the characteristics of a memory element?
a) A toggle switch b) A lamp [WBUT 2010]
c) An AND gate d) None of these
Ans.: (c)
10. Data from memory location after fetching is deposited by memory in
a) MAR b) MDR [WBUT 2010]
c) IR d) Status Register
Ans.: (b)

11. Virtual memory system allows the employment of [WBUT 2010]

- a) More than address space
- b) The full address space
- c) More than hard disk capacity
- d) None of these

Answer: (a)

12. A system has 48-bit virtual address, 36-bit physical address and 128 MB main memory. How many virtual and physical pages can the address space support? [YYBUT 2010]

- a) 2^{24} , 2^{24}
- b) 2^{21} , 2^{36}
- c) $2^{24}, 2^{34}$
- d) $2^{134}, 2^{36}$

Answer: (b)

13. Maximum number of directly addressable locations in the memory of a process or having 10 bits wide control bus, 20 bits address bus and 8 bit data bus is [WBUT 2012]

- a) 1 K
- b) 2 K
- c) 1 M
- d) none of these

14. Periodic refreshing is needed

- a) SRAM
- b) DRAM
- c) ROM
- d) EPROM

Answer: (b)

15. Physical memory broken down into groups of equal size is called [WBUT 2012]

- a) page
- b) tag
- c) block/frame
- d) index

Answer: (c)

16. Bi-directional buses use

[WBUT 2012]

- a) tri-state buffers
- b) two tri-state buffers in cascade
- c) two back to back connected tri-state buffer in parallel
- d) two back to back connected buffers

Answer: (c)

17. Micro Instructions are kept in

[WBUT 2012]

- a) main memory
- b) control memory
- c) cache memory
- d) none of these

Answer: (b)

18. Size of virtual memory is equivalent to the size of

[WBUT 2013]

- a) main memory
- b) cache memory
- c) secondary memory
- d) both (a) and (c)

Answer: (c)

19. The associative access mechanism is followed in

[WBUT 2014]

- a) main memory
- b) cache memory
- c) magnetic disk
- d) both (a) and (b)

Answer: (d)

POPULAR PUDLI ATfON

20. The slowest part of memory is up to 10¹⁰ ns by [WBUT 2014]
a) paging b) segmentation c) both
d) none of these
Ans., cr: (a)
21. The largest delay in accessing data on disk is due to [WAUT 2014]
a) seek time! b) rotation time
c) data transfer time d) none of those
Ans., er: (t)
22. Address of memory location for fetching data needs to be deposited in memory [WBUT 2014]
in
a) MAR c) MBR c) IR d) status register
Ans., er: (a)
23. Size of virtual memory is equivalent to the size of [WBUT 2014]
a) hard disk b) CPU c) floppy disk d) none of these
Ans.,\cr: (a)
24. If k be the number of registers and n be the size of each register, then in order to construct n-line common bus system using tri-state buffers, the total number of tri-state buffers and the size of decoder would be [WBUT 2015]
a) n^k and 2-to-4 b) n^k and log₂ k-to-k
c) k and log₂ n-to-n d) n^k and log₂ n-to-n
Answer: (b)
25. RAM is called DRAM (Dynamic RAM) when [WBUT 2016]
a) it is always moving around data
b) it requires periodic refreshing
c) it can do several things simultaneously
d) none of these
Answer: (b)
26. In order to execute a program instructions must be transferred from memory along a bus to the CPU. If the bus has 8 data lines, at most one 8 bit byte can be transferred at a time. How many memory accesses would be needed in this case to transfer a 32 bit instruction from memory to the CPU? [WBUT 2016]
a) 1 b) 2 c) 3 d) 4
Answer: (d)
27. A computer's memory is composed of 2K words of 32 bits each. How many bits are required for memory address if the smallest addressable memory unit is a word? [WBUT 2016]
a) 13 b) 8 c) 10 d) 6
Answer: (a)

28. cache memory refers to [WBUT 2016]
- a) cheap memory that can be plugged into the mother board to expand main memory
 - b) fast memory present on the processor chip that is used to store recently accessed data
 - c) a reserved portion of main memory used to save important data
 - d) a special area of memory on the chip that is used to save frequently used data

Ans,ver: (b)

29. **Write Through** technique is used in which memory for updating the data?

- a) Virtual memory
- b) Main memory [WBUT 2016]
- c) Auxiliary memory
- d) Cache memory

Ans,ver: (d)

- 30..... is generally used to increase the apparent size of physical memory.

- a) Secondary memory
- b) Virtual memory [WBUT 2017]
- c) Hard disk
- d) Disks

Ans,ver: (b)

31. The time delay between two successive initiations of memory operation is

- a) Memory access time
- b) Memory search time [WBUT 2017]
- c) Memory cycle time
- d) Instruction delay

Answer: (c)

32. A 24 bit address generates an address space oflocations

[WBUT 2017, 2019]

- a) 1024
- b) 4096
- c) 2^{14}
- d) 16,777,216

Answer: (d)

33. To get the physical address from the logical address generated by CPU we use

[WBUT 2017, 2019]

- a) MAR
- b) MMU
- c) Overlays
- d) TLB

Answer: (b)

34. During transfer of data between the processor and memory we use

[WBUT 2017]

- a) Cache
- b) TLB
- c) buffers
- d) Registers

Answer: (d)

35. The return address of the Sub-routine is pointed to by

[WBUT 2017]

- a) IR
- b) PC
- c) MAR
- d) Special memory registers

Answer: (b)

36. The instruction, Add R1, 45 does [WflUT2019]

- a) Adds the value of 45 to the address of R1 and stores 46 in register R1
- b) Adds 45 to the value of R1 and stores it in R1
- c) Finds the memory location 45 and adds that content to that of R1
- d) None of these

Answer: (b)

37. What could be the maximum size of on chip cache memory for an nMaddr . hit processor? [WBUT 2019]

- a) 0
- b) 2^n
- c) Infinite
- d) decided by manufacturer

Answer: (c)

Short Answer Type Questions 1

1. Given the following determine the size of the sub fields (In bits) in the address for Direct Mapping, associative and set associative mapping cache schemes:

- We have-256 MB main memory and 1 MB cache memory.
- The address space of this processor is 256 MB.
- The block size is 128bytes.
- There are 8 blocks in a cache set.

[WBUT 2004, 2007, 2012, 2013]

Answer:

As the size of the main memory is 256 MB hence there are 28 bits (as $256 = 2^8$ and 1 MB = 2^{20} bytes and hence $256 \text{ MB} = 2^8 \times 2^{20} = 2^{28}$) in the main memory address or the address size of main memory is 28 bits.

Size of the sub-fields for associative mapping cache schemes:

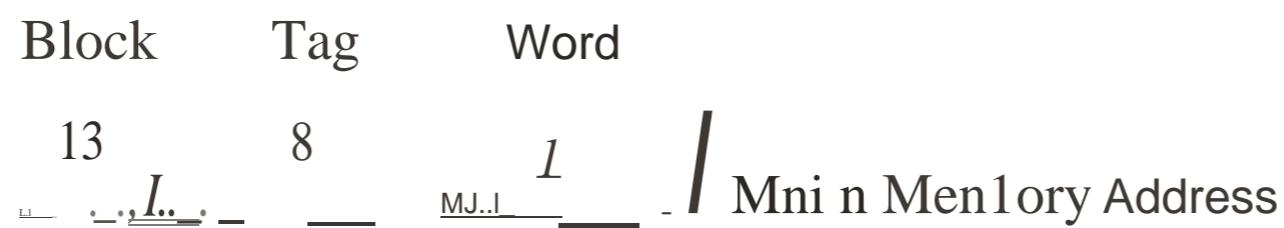
Each block size is 128 bytes or 2^7 bytes. Hence number of main memory blocks = 2^{21} . So number of bits in the tag field is 21 and that in the word field is 7 (as block size is 128 bytes).

Tag Word



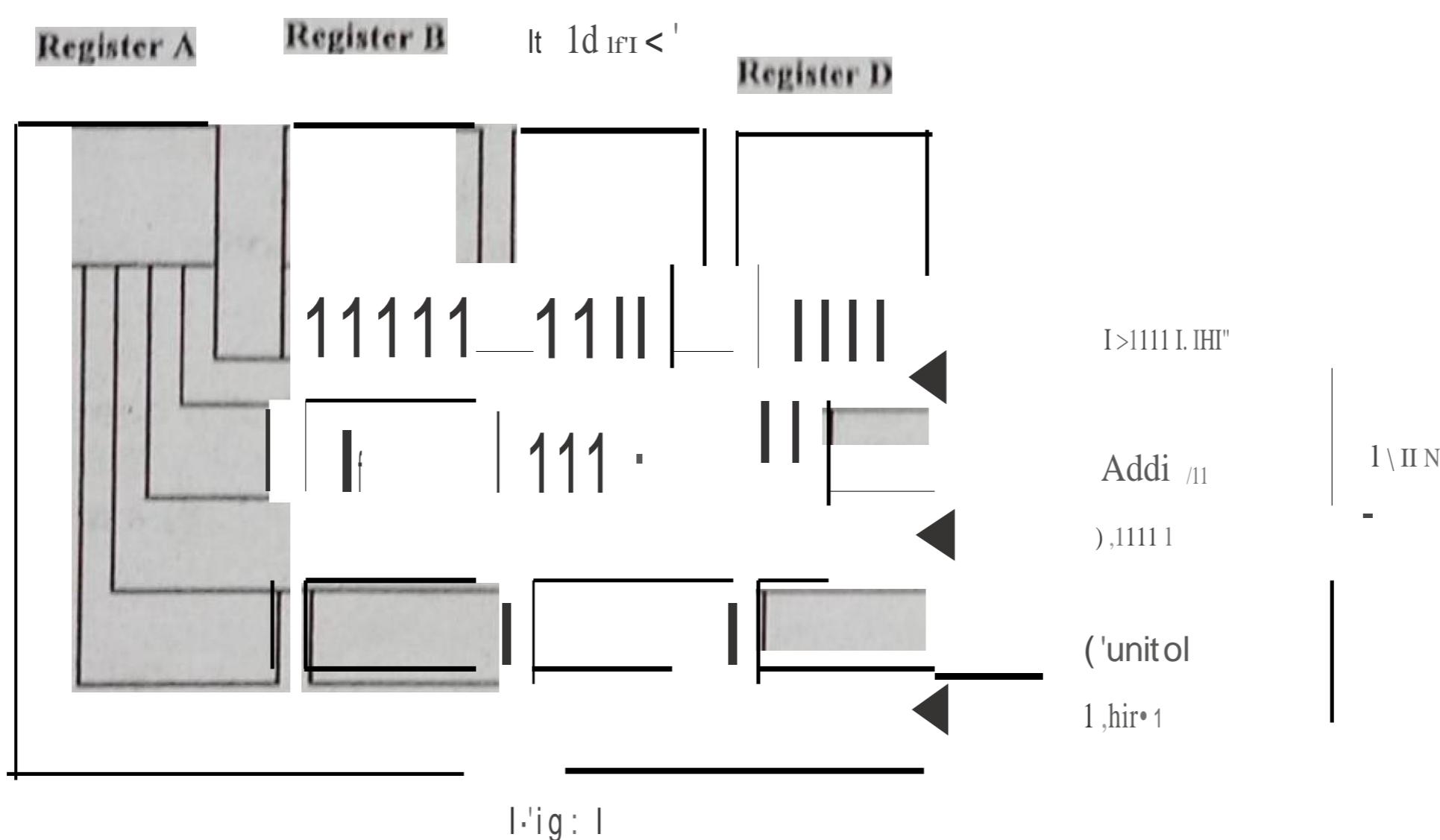
Size of the sub-fields for direct mapped cache schemes:

Now size of cache memory is 1 MB = 2^{20} bytes. Hence number of cache memory blocks = $2^{20} / 2^7 = 2^{13}$. So number of bits in the block field = 13. Now out of the total main memory address size of 28 bits, word field contains 7 bits and the block field contains 13 bits. Hence, the number of bits in the tag field is $28 - (13 + 7) = 8$.



BUS STRUCTURE

σ Chapter at a Glance



- f)ntu, A(Jdr • s .111<,I .0 11t•ol IJ,ac ,,)f JI 11, :
J)atu J..luc.s: 'JJ1cs Ji110H provic1 n JJ[1111 f<JJ'111<>v11p, dul I b 1wc• 11 l't'H It I,' ' l'hc\ t 11rc tic H 1>1 ltti
 to tran s nit ufl bits <JI u1\ 11..1lit wore! i,1 J1Hl'all t1, 8 11, Iii y t lhc, t'0II I,1 111 1,V,, c Ii 1D1' 11
 unidirectional fñ ric; or a :i1111c :IGI (>f 1-bi..J il'eclikulttl l11ch.
**J)ala line· arc colfccli vclv c:nJlc d t11c 1/1/11 /11,1, 'J'11IH ln1 cif il.c• ^o I, 11.1111y 11 1111dli1111• t11' tt,
 with n 8, 16 or 3:2 etc. S1.Jptu·afol ncs,**

POPULAR PUBLICATIONS

- **Tri-state buffer:** Bus system constructed with multiplexers. has some serious disadvantages_ If 111orc 11t11nbc of registers are connected by the bus, multiplexers cannot support that load. R11d voltages drop. Also as all registers (connected to the bus) draw some current from the bus even if they are not transmitting at that time, the bus voltage drops and thus the transfer becomes unreliable.
 - o to counter these problems a device (or a digital circuit) called *tri-state buffer* is used. It exhibits three states, two inputs and one output. The output can be in one of the three states (signal values) namely, *logic 0*, *logic 1* and a '*/igl1- i111peda11ce*' state.

I Multile Choice Type Questions ≡

1. The logic circuitry in ALU is

[WBUT 2008]

- a) entirely combinational
- b) entirely sequential
- c) combinational cum sequential
- d) none of these

Answer: (c)

2. A single bus structure is primarily found in

[WBUT 2008, 2011]

- a) main frames
- b) super computers
- c) high performance machines
- d) mini and micro-computers

Answer: (d)

3. To construct an n-line common bus using MUX fork registers of n bits each, the number of MUXs and size of each MUX are

[WBUT 2014]

- a) k and nx1
- b) n and 2k
- c) n and kx1
- d) k and 2n

Answer: (c)

4. The main purpose for using single Bus structure is

[WBUT 2017]

- a) Fast data transfer
- b) Cost effective connectivity and speed
- c) Cost effective connectivity and ease of attaching peripheral devices
- d) none of these

Answer: (c)

5. The main advantage of multiple bus organisation over single bus is

[WBUT 2017]

- a) Reduction in the number of cycles for execution
- b) Increase in size of the registers
- c) Better connectivity
- d) none of these

Answer: (a)

6. The maximum propagation delay for n-bit CLA is

[WBUT 2018]

- a) I_i
- b) $\bullet n$
- c) $"6"$
- d) n

Answer: (b)

INPUT-OUTPUT ORGANIZATION

Chapter Glance

- **Input-output organisation:** The Input-Output (*I/O*) organization of a computer provides the mode of communication between the computer and the outside world i.e. with the help of the I/O devices, users can interact with the computer. The I/O devices that are attached to the computer are also known as *peripherals*.

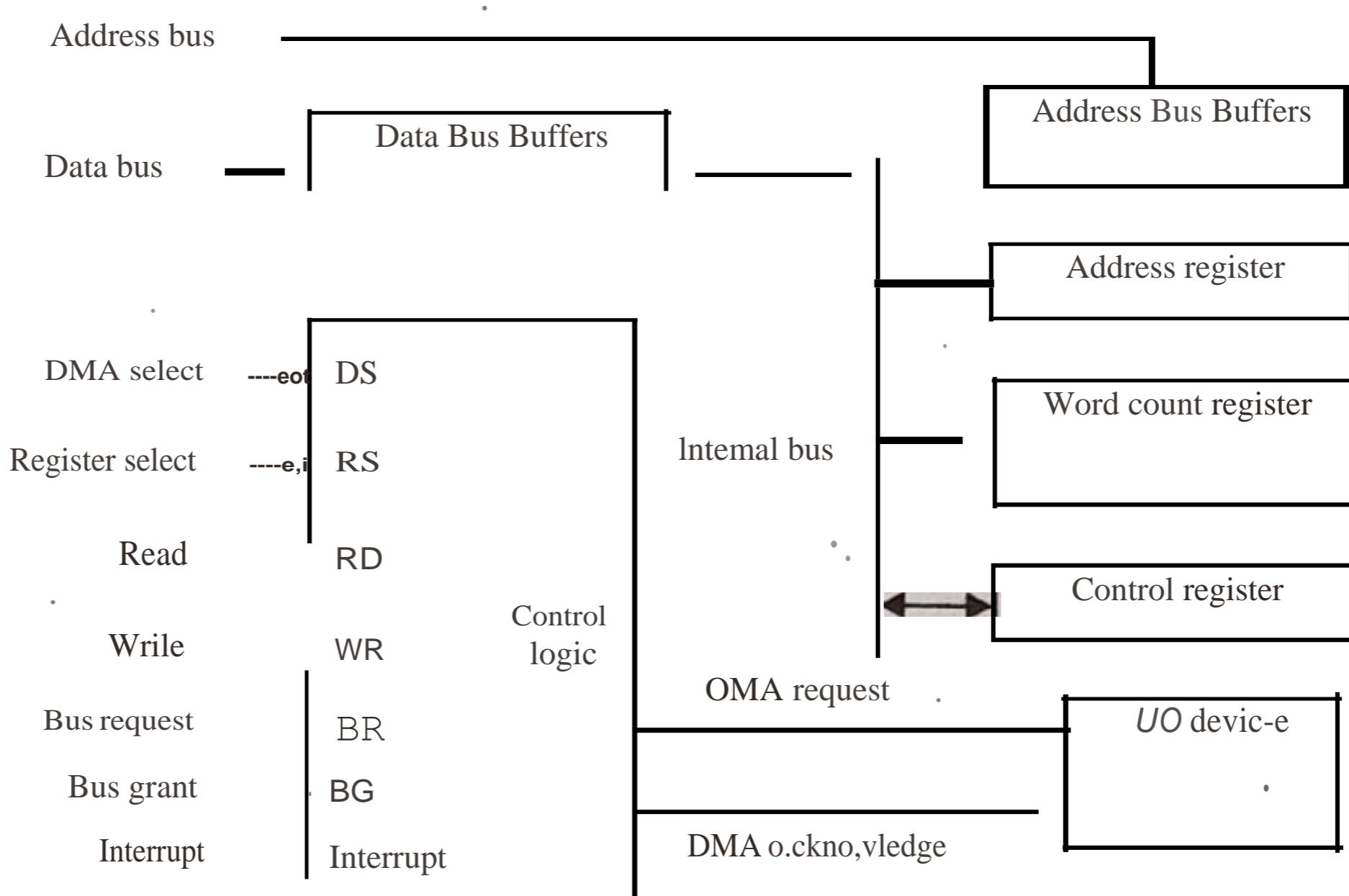
Some of the I/O devices that are commonly used in a computer are:

Input Devices: Keyboard, Mouse, Joystick, Scanner etc.

Output Devices: Printer, Monitor etc.

- **Direct memory Access (DMA):** In programmed I/O data transfer occurs between CPU and the peripherals. However in DMA, data transfer occurs between I/O devices and the memory unit without direct intervention by the CPU. CPU only initiates the transfer by supplying the starting address of the memory location from where data is to be transferred and the number of words (bytes) to be transferred.

DMA Controller:



- **Interrupt:** Interrupt is a signal from an I/O device to let the CPU know that it is ready to transfer data and that it is requesting service from the CPU. As soon as the CPU receives this signal, it leaves its current unfinished task as it is and branches to the interrupting device's interrupt service routine and comes back to its unfinished job to resume the transfer. When finished, CPU again resumes the task and continues with it.

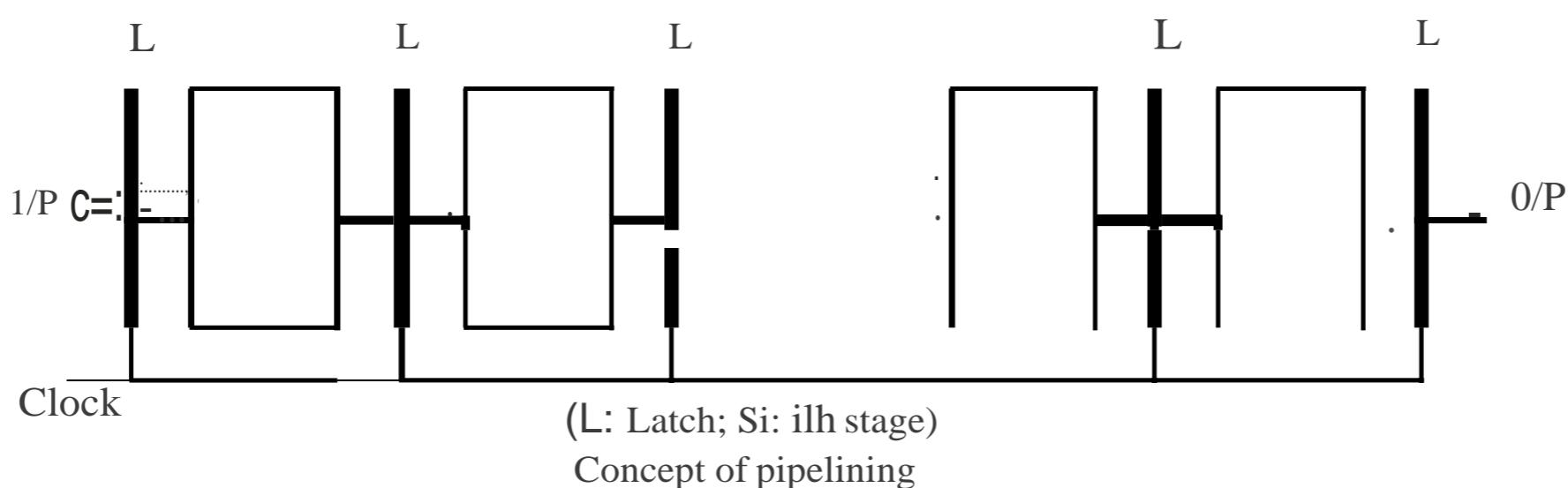
Different types of Interrupts: There are generally three types of interrupts, which are: .

- (a) *External Interrupts*: Such type of interrupts may come from any external source i.e., from I/O devices when they are ready to transfer data., from a *logging device* to signify that the time of an event is over it may occur due to some *power failures* etc. .
- (b) *Internal Interruptions*: Such type of interrupts, called traps, may occur due to some illegal or erroneous conditions in the program (i.e. illegal or erroneous use of instructions or data in the program).
- (c) *Software Interrupts*: Such type of interrupts may be incorporated or embedded in the program as an instruction by a programmer and are thus initiated by executing that instruction (interrupt instruction). So if the programmer wants to initiate any sort of interrupt procedure at any desired point in the program, he may write an interrupt instruction at that point in the program.

Example! of software interrupt is: INT 32 (say). On execution of this interrupt instruction, control branches to the ISR of the number 32 interrupt (i.e. 32 number interrupt line).

- *on-Vectored Interrupt*: In this method, the branch address (i.e. address of the ISR) is always assigned to a fixed location in the memory and the processor always branches to that particular location.
- *Vectored Interrupt*: In this method the branch address (i.e. address of the ISR) is supplied by the interrupting I/O device itself and the processor branches accordingly.
- Pipelining: Pipelining is a technique of decomposing a sequential task into subtasks, with each subtask being executed in a special dedicated stage (segment) that operates concurrently with all other stages. Each stage performs partial processing dictated by the way the task is partitioned. Result obtained from a stage is transferred to the next stage in the pipeline. The final result is obtained after the instruction has passed through all the stages.

All stages are synchronized by a common clock. Stages are pure combinational circuits performing arithmetic or logic operations over the data stream flowing through the pipe. The stages are separated by high-speed interface latches (i.e., collection of registers). Figure below shows the pipeline concept with k stages.



f Multiple Choice 1-type Questions :J

1. In a micro-processor, the address of the next instruction to be executed, is stored in [WBUT 2008]

- | | |
|--------------------|-----------------------------|
| a) stack pointer | b) address latch |
| c) program counter | d) general purpose register |

Answer: (c)

POPULAR PUBLICATION

2. **Memory** rn ppod 1/0 ct1 n₁ 1 t1..1, d f r tll • 111 c ti 11 f .,ddr~~oss~~ t I11 IIIOT
and 1/0 dovlcos ls usod for [WBUI ODs]
a) n1all y tom
b) l;1r[l y t 11
c) both largo r1d 0111 y tolrl
d) v ry l r y t r11
A11s,vc1·:(t1)

3. For BIOS (Basic I/Oput/output yst rt1) nd IOC (Inpllt/oLlput ontrol Y tom)
which ono of tl10 following I tr11 ? [WBUT 009]

- a) BIOS and IOCS aro same
- b) BIOS controls all dovlces and IOCS control only c rt.. n v c
- c) BIOS Is not a part of oportng Y t r11 and IOC I " P rt of operating system
- d) BIOS is storod in ROM and IOCS Is storod n RAM

Ans ,vcr: (c)

4. A priority Interrupt may bo accompllsl1od by [WBUT 2010]
a) Polling
b) Daisy chain
c) Parallel mothod of priority Interrupt
d) All of those

Answer: (n)

6. BIOS Is [WBUT 2013]
a) a collection of 1/0 driver programs
b) part of OS to perfrom I/Ooporation
c) **firmware** consisting of 1/0 driver programs
d) a program to control one of the 1/0 peripherals

Answer: (b)

6. In DMA the term cycle stealing means [WBUT 2014]
a) controller gets opportunity to transfer only one word in a timeslot
b) CPU releases the bus and DMA controller can uso endlessly
c) 100 bytes are allowed to be transferred
d) none of these

Answer: (a)

7. The contention for the usago of a hardware device is called [WBUT 2019]
a) structural hazard
b) stalk
c) deadlock
d) nono of tt10s0

Answer: (a)

- 8 The periods of time when tho unit Is Idle Is called as [WBUT 2019]
a) **Stalls**
b) Bubbles
c) Hazards.
An wcr: (d)

g, The pMA transfer is Initiated by

[WBUT 2019]

- a) processor
- c) I/O devices

- b) The proco boing executed
- d) OS

A.nswct: (c)

Short Answer Type Questions

1. What are the different hazards in Pipeline?
OR,

[WBUT 2007, 2019]

Explain Pipelining and Hazards

[WBUT 2017]

Answer:
Pipeline hazards are situations that prevent the next instruction in the instruction stream from executing its designated clock cycle. There are three types of pipeline hazards:

- i) Control hazards
- ii) Structural hazards
- iii) Data hazards

Control Hazards: They arise from the pipelining of branches and other instructions that change the content of program counter (PC) register.

Structural Hazards: Structural hazards occur when a certain resource (memory, functional unit) is requested by more than one instruction at the same time.

Data Hazards: Inter-instruction dependencies may arise to prevent the sequential (in-order) data flow in the pipeline, when successive instructions overlap their fetch, decode and execution through pipeline processor. This situation due to inter-instruction dependencies is called data hazard.

2. What are the different types of interrupt? Give examples.

[WBUT 2008]

Answer:

Refer to Chapter at a Glance.

3. What is interrupt?

[WBUT 2009, 2011]

What are the differences between vectored and non-vectored interrupt?

[WBUT 2009, 2011, 2018]

Answer:

1st Part:

Interrupt is a signal from an I/O device to let the CPU know that it is ready to transfer data and that it is requesting service from the CPU.

As soon as CPU receives this signal, it leaves its current unfinished task as it is and branches to the interrupting device's interrupt service routine and executes it to process the transfer. When finished, CPU again comes back to its unfinished task and continues with it.

QUE-STION 2015

- Choose the correct answer for the following questions.**
- (Multiple choices for the following questions)
- 1) If C = 1 - \$, what algorithm?
- a) 01001111 b) 01111000 c) 00001111 d) 010101

- i) In the processor, the address of next instruction to be executed is stored in
- a) stack pointer register
 ✓c) base register

3. Left-shift the content of a register by its content
- a) double
 ✓d) such decision can be made
- b) both a & c
 ✓d) k and log2 k-to-k

- number of registers and number of each register total number of tr
- the common bus system state
- and 2-to-4
 ✓b) k and log2 k-to-k
 c) n k and log2 n-to-n
- J : J X ✓d) cache memory

- v) Instruction cycle is
- ✓a) fetch-decode-execution
 b) fetch-execution
 decode-fetch-execution
 d) none of these

- Subtractor can be implemented using
- a) adder
 ✓c) both (a) and (b)
 b) complements
 d) none of these

- many RAM chips of size (256 K \times 1 bit) are required to build a 1 M byte cache?
- b) 10 c) 32 ✓d) 8

- x) Maximum number of 2's complement numbers
- a) 2^n b) 2^{n-1} ✓c) $2^{n-1} - 1$ d) cannot be said

- x) Microprocessor is used for
- main memory / b) cache memory CJ cache memory d) none of these

POPULAR PUBLICATIONS

Group - B

(Short Answer Type Questions)

2. a) What is tri-state buffer? Construct a single line common bus system using tri-state buffer.
b) What are guard bits?

a) See Topic: **BUS STRUCTURE**, Short Answer Type Question No. 3.

b) See Topic: **COMPUTER ARITHMATIC**, Short Answer Type Question No. 3.

3. Describe stack based CPU.

See Topic: **MEMORY ORGANIZATION**, Short Answer Type Question No. 9.

4. a) Write $+7_{10}$ in IEEE 32 bit format.

b) Convert IEEE 32-bit format 40400000_{16} in decimal value.

a) See Topic: **COMPUTER ARITHMATIC**, Short Answer Type Question No. 5.

b) See Topic: **COMPUTER ARITHMATIC**, Short Answer Type Question No. 11.

5. Evaluate the following arithmetic expression into three-address, two-address, one-address, zero-address instruction format. $X = (A + B) * C$

See Topic: **INSTRUCTION SET**, Short Answer Type Question No. 1.

6. a) Explain the difference between full associative and direct mapped cache memory mapping approaches.

b) What are "write back" and "write through" policies in cache?

a) See Topic: **MEMORY ORGANIZATION**, Long Answer Type Question No. 1(b).

b) See Topic: **MEMORY ORGANIZATION**, Long Answer Type Question No. 1(a).

Group - C

(Long Answer Type Questions)

7. a) Suppose register A holds the 8-bit number 11011101. Determine the sequence of binary values in A after an arithmetic shift-right, followed by a circular shift-right and followed by a logical shift-left. 3

b) Describe Booth's multiplication method and use this to multiply decimal numbers -23 and 9.

c) Suppose we are given RAM chips each of size 256×4 . Design a $2K \times 8$ RAM system using this chip as the building block. Draw a net logic diagram of your implementation.

a) & b) See Topic: **COMPUTER ARITHMATIC**, Long Answer Type Question No. 9.

c) See Topic: **MEMORY ORGANIZATION**, Short Answer Type Question No. 8.

8. a) For Booth's algorithm, when do worst case and best case occur? Explain with example.

b) What are the advantages of Interrupt I/O over programmed I/O?

c) Discuss the concept of associative memory unit using suitable example.

a) See Topic: **COMPUTER ARITHMATIC**, Short Answer Type Question No. 12.

b) See Topic: **INPUT-OUTPUT ORGANIZATION**, Short Answer Type Question No. 6.

c) See Topic: **MEMORY ORGANIZATION**, Long Answer Type Question No. 2.

9. a) Write a program to evaluate the arithmetic statement $Y = (A - B + C) / (G + H)$.

(i) Using an accumulator type computer with one address instruction.

- (ii) Using a stack organized computer with zero-address instructions
- b) What is van Neumann bottleneck? How can this be reduced?
- c) A digital computer has a memory unit of **64K × 18** and cache memory of 1KB to 16 bits. The cache uses direct mapping with a block size of 4 words. How many bits are there in the tag, index and address fields of the address format? Also calculate address format for associative mapping for 4-way set associative mapping.

- a) See Topic: INSTRUCTION SET, Long Answer Type Question No. 3.
- b) See Topic: INTRODUCTION, Long Answer Type Question No. 1.
- c) See Topic: **MEMORY ORGANIZATION**, Long Answer Type Question No. 13.(c).

10. a) Differentiate between hardwired control and micro programmed control. Draw the block diagram of a basic hardwired control organization with two decoders and a sequence counter and a number of control logic gates.

b) A digital computer has a common bus system for 16 registers of 32 bits each. The bus is constructed with multiplexers?

- (i) How many selection inputs are there in each multiplexer?
- (ii) How many multiplexers are there in the bus?
- c) Explain the basic OMA operations for transfer of data between memory and processor.

- a) See Topic: **CONTROL UNIT**, Long Answer Type Question No. 3.
- b) See Topic: **BUS STRUCTURE**, Short Answer Type Question No. 4.
- c) See Topic: **INPUT-OUTPUT ORGANIZATION**, Long Answer Type Question No. 3(b).

11. a) Explain different hazards in pipelining.

b) What are vector interrupts? How are they used in implementing hardware interrupts?

c) What is speedup, throughput and efficiency of a pipelined architecture?

See Topic: **INPUT-OUTPUT ORGANIZATION**, Long Answer Type Question No. 6.

QUESTION 2016

Group-A

(Multiple Choice Type Questions)

1 Choose the correct alternatives for the following:

Q RAM is called DRAM (Dynamic RAM) when

- a) it is always moving around data
- b) it requires periodic refresh
- c) it can do several things simultaneously
- d) none of these

Q Floating point representation is used to store

- a) Boolean values
- b) Whole numbers
- c) Real
- d) Integers

Q A given memory chip has 12 address pins and 4 data pins. It has 2^{12} locations.

a) 2^4

✓ b) 2^{12}

c) 2^{48}

d) 2^{16}

POPULAR PUBLICATIONS

iv) In order to execute a program instructions must be transferred from memory along a bus to the CPU. If the bus has 8 data lines, at most one 8 bit byte can be transferred at a time. How many memory accesses would be needed in this case to transfer a 32 bit instruction from memory to the CPU?

computer's memory is composed of aK words of 32 bits each. How many bits are required¹ to form a memory address if the smallest addressable memory unit is a word?

vi) Cache memory refers to

- a) cheap memory that can be plugged into the mother board to expand main memory
 - b) fast memory present on the processor chip that is used to store recently accessed data
 - c) a reserved portion of main memory used to save important data
 - d) a special area of memory on the chip that is used to save frequently used data

vii) SIMD represents an organization that

- a)** refers to a computer system capable of processing several programs at the same time
b) represents organization of single computer containing a control unit, processor unit and a memory unit
c) includes many processing units under the supervision of a common control unit
d) none of these

viii) The circuit used to store one bit of data is known as

- a) Register b) Encoder c) Decoder d) Flip-flop

ix) (2FAOC)16

- a) (195084)10
 - b) (001011110100001100)2
 - c) Both (a) and (b)
 - d) None of these

x) The addressing mode used in an instruction of the form ADDX Y is

- a) absolute b) indirect c) index d) none of these

xi) Write Through technique is used in which memory for updating the data?

- a) Virtual memory
 - b) Main memory
 - c) Auxiliary memory
 - d) Cache memory

x ii A stack-organised computer uses instruction of

- a) Indirect addressing
 - b) Two addressing
 - c) Zero addressing
 - d) Index addressing

Group-8

(Short Answer Type Questions)

2 Explain indirect address mode. How is the effective address calculated in this case?

ee 'ɒp ɪ : I S'J'R|JCI |ON SE'f . Short Answer Type Question No. 7.

CO-150

3 Design a 4-bit combinational circuit to decrement a 5-bit number.

Set Topic: COMPUTER ARITHMETIC, Short Answer Type Question No. 4(b).

4 A digital computer has a common bus system constructed with multiplexers. It has four 16-bit registers of 32 bits each. The bus is

i) How many selection inputs are there in each multiplexer?

ii) How many multiplexers are there in the bus?

Set Topic: BUS STRUCTURE Short Answer Type Question No. 10(b).

5. Write a program to evaluate the arithmetic statement

$$Y = (A - B + C) / (G + H)$$

i) Using an accumulator type computer with one address structure

ii) Using a stack organization computer with zero-addresses structure

Set Topic: i) STR CI'IO SET Long Answer Type Question No. 3.

5 Show how to implement a full adder, by using half adders.

Set Topic: COMPUTER ARITHMETIC, Short Answer Type Question No. 14.

Group-C

(Long Answer Type Questions)

7. a) A computer uses a memory unit with 256 K words of 32 bits each. A binary instruction code is stored in one word of memory. The instruction has four parts: an indirect bit, an operation code, a register code part to specify one of 64 registers, and an address part-

Q How many bits are there in the operation code, the register code part, and the address part?

iQ Draw the instruction word format and indicate the number of bits in each part.

iiQ How many bits are there in the data and address inputs of the memory?

b) Use restoring method to divide 10100011 by 1011.

c) Suppose we are given RAM chips each of size 256x4. Design a 2Kx8 RAM system using this chip as the building block. Draw a net logic diagram of your implementation.

a) See Topic: INSTRUCTION SET, Short Answer Type Question No. 6.

b) See Topic: COMPUTER ARITHMETIC, Short Answer Type Question No. 13.

c) See Topic: MEMORY ORGANIZATION, Short Answer Type Question No. 8.

8. a) For Booth's algorithm, when do worst case and best case occur? Explain with example.

b) What are the advantages of Interrupt I/O over Programmed I/O?

c) Draw the logic diagram of the cell of one word in associative memory including the read and Write logic.

1) See Topic: COMPUTER ARITHMETIC, Short Answer Type Question No. 12.

2) See Topic: MEMORY ORGANIZATION, Short Answer Type Question No. 6.

3) Set Topic: MEMORY ORGANIZATION, Long Answer Type Question No. 16.

9 a) Explain the various phases of instruction cycle in a basic computer.

b) What is Von Neumann bottleneck? How can this be reduced?

POPULAR PUBLICATIONS

- c) A two-way set associative cache memory uses blocks of four words. The cache can accommodate a total of 2048 words from the main memory. The main memory Kxa,2
i) How many bits are there in the tag indexblock and word fields of the address?
ii) What is the size of the cache memory?
a) See topic: CONTROL UNIT, Long Answer Type Question No. 2
b) See Topic: INTRODUCTION, Long Answer Type Question No. 1.
c) See Topic: MEMORY ORGANIZATION, Long Answer Type Question No. 17.

1.0 a) Differentiate between hardware control and micro programmed control. Draw the block diagram of a basic hardware control organization with two decoders, a sequencer and a number of control logic gates.

b) A hierarchical Three -Level Memory (Cache, Main memory, Hard Disc) system has the following specifications:

- i) Cache Memory Access Time is 10nsec
- ii) Disc Access Time is 150nsec
- iii) Hit ratio of Cache Memory is 0.97
- iv) Hit ratio of Main Memory is 0.9

What should be the Main Memory access time to achieve an overall access time of 20nsec?

c) Explain the basic OMA operations for transfer of data between memory and peripherals

a) See Topic: CONTROL UNIT, Long Answer Type Question No. 3.

b) See Topic: MEMORY ORGANIZATION, Long Answer Type Question No. 17

c) See Topic: INPUT-OUTPUT ORGANIZATION, Long Answer Type Question No. 3b.)

- i) a) What are the hazards of instruction pipelining? How are these taken care of?
b) Explain the strobe Control method of Asynchronous data transfer. What are the advantages of this method?
c) What do you understand by the term 'Program Interrupt'? Explain with the help of suitable diagrams.

See Topic: INPUT-OUTPUT ORGANIZATION, Long Answer Type Question No. 7.

QUESTION 2017

Group-A

(Multiple Choice Type Questions)

C) Choose the correct alternatives for any ten of the following:

- i) The main purpose for using single Bus structure is
 - a) Fast data transfer
 - b) Cost effective connectivity and speed
 - c) Cost effective connectivity and ease of attaching peripheral devices
 - d) None of these
- ii) The ALU makes use of to store the intermediate results
 - a) Accumulators
 - b) Registers
 - c) Stack
 - d) Stack

- iii) Is generally used to Increase the app rent size of physcalmemo ry.
- a) Secondary memory
 - b) Vlrtual memory
 - c) Hard disk
 - d) Disks
- iv) The time delaybe twn two successive Initiations of memory operation Is
- a) Memo ry accessi me
 - b) Memory search time
 - c) Memo ry cycle time
 - d) Instruction delay
- v) The main advantage of multiple bus organisation over slngle bus Is
- a) Reduction in the number of cycles for execution
 - b) Increase in size of the registers
 - c) Better connectivity
 - d) none of these
- vi) When performing a looping operation, the Instruction gets stored in the
- a) Registers
 - b) Cache
 - c) System heap
 - d) **System st ck**
- vii) In case of Zero-address instruction method the operands are stored in
- a) Registers
 - b) Accumulators
 - c) Stack
 - d) Cache
- viii) The addressing mode(s), which uses the PC instead of a general purpose register is
- a) Indexed with offset
 - b) Relative
 - c) Direct
 - d) Both (a) and (b)
- ix) In a normal n-bit adder, to find out if an overflow has occurred, we make use of
- a) AND gate
 - b) NANO gate
 - c) NOR gate
 - d) XOR gate
- x) A 24 bit address generates an address space of locations
- a) 1024
 - b) 4096
 - c) 2^{14}
 - d) 16,777,216
- xi) To get the physical address from the logical address generated by CFIU we use
- a) MAR
 - b) MMU
 - c) Overlays
 - d) TLB
- xii) During transfer of data beM,een the proces\$or and memory we use
- a) Cache
 - b) TLB
 - c) buffers
 - d) Registers
- xiii) The return address of the Sub-routine is pointed to by
- a) IR
 - b) PC
 - c) MAR
 - d) Special memory registers

Group-B

(Short Answer Type Questions)

2. a) Briefly explain the IEEE 754 standard format for floating point number representation.
 b) Represent the decimal value (-7.5) in IEEE single precision format.
a) See Topic: COMPUTER ARITHMETIC, Short Answer Type Question No. 4(a).
b) See Topic: COMPUTER ARITHMETIC, Short Answer Type Question No. 16.

POPULAR PUBLICATIONS

3. Two 1024×4 bits RAM chips are given. Design a memory of size 2048×4 bits.

Sec 1st part: **MEMORY ORGANIZATION**, Short Answer Type Question No. 10.

4. What is the difference between carry look ahead adder and ripple carry adder? Explain the working of operating system in a computer system. 010

1st Part: Sec Topic: COMPUTER ARITHMETIC, Short Answer Type Question No. 15.

2nd part: Sec Topic: INTRODUCTION, Short Answer Type Question No. J.

5. Explain Pipelining and Hazards. Define latency time of a memory.

1st part: Sec Topic: INPUT-OUTPUT ORGANIZATION, Short Answer Type Question No. I.

2nd part: Sec Topic: MEMORY ORGANIZATION, Short Answer Type Question No. J I.

6. a) What are the advantages of associative mapping over direct mapping?

b) Consider a series of address references given: 2, 3, 11, 16, 21, 13, 64 and 48. Assuming a direct mapped cache with 8 one-word blocks that is initially empty, label each reference in the list as a hit or a miss and show the final contents of the cache.

Sec Topic: MEMORY ORGANIZATION, Short Answer Type Question No. 12.

Group-C

(Long Answer Type Questions)

7. a) Present the Booth's algorithm for multiplication of signed 2's complement number in a flow chart and explain.

b) Multiply (-12) and (+6), using Booth's multiplication algorithm.

c) Divide (-15) by (-3) using Restoring & Non-restoring Division algorithm.

a) See Topic: COMPUTER ARITHMETIC, Long Answer Type Question No. 1.

b) See Topic: COMPUTER ARITHMETIC, Short Answer Type Question No. 7.

c) Sec Topic: COMPUTER ARITHMETIC, Long Answer Type Question No. 10.

8. Discuss in detail the various factors that need to be considered while designing the ISA of a processor. Compare and contrast of RISC and CISC architecture.

1st part: See Topic: INSTRUCTION SET, Long Answer Type Question No. 4.

2nd part: See Topic: INSTRUCTION SET, Short Answer Type Question No. J.

9. Explain in detail the Bus Arbitration techniques in DMA.

Set Topic: INPUT-OUTPUT ORGANIZATION, Long Answer Type Question No. 3(b).

1a. a) Can ROM be also a RAM? Justify your answer.

b) What is speed up? Prove that maximum speed up will be K.

c) A disk pack has 20 surfaces. Storage area on each surface has an inner diameter of 22cm and outer diameter of 33cm. Maximum storage density on each track is 2000 bits/cm and maximum spacing between tracks is 0.25mm.

r) What is the storage capacity of the pack?

ii) What is the data transfer rate in bytes per second at a rotational speed of 7200r.p.m.?

d) What is the necessity of guard bits?

S 1st part: M INPUT-OUTPUT ORGANIZATION, Long Answer Type Question No. 8(a).

1st part: fc, pr: INPUT-OUTPUT ORGANIZATION, Long Answer Type Question No. 8.

COMPUTER ORGANISATION

c) See Topic: **MEMO y, ORGANIZATION**, Long Answer Type Question No. 14(b).

d) See Topic: **COMPUTER RITIIMETIC**, Short Answer Type Question No. 3.

11

a) What are the advantages of relative addressing mode over direct addressing mode?
b) compare and contrast Memory mapped I/O and I/O mapped I/O.

c) Explain the importance of a common bus system in a computer. Why I/O bus is different from a system bus?

R) See Topic: **INSTRUCTION SET**, Short Answer Type Question No. 4.

b) See Topic: **INPUT-OUTPUT ORGANIZATION**, Long Answer Type Question No. 4.

c) I part: See Topic: **BUS STRUCTURE**, Short Answer Type Question No. 1(b).

2 D part: See Topic: **DUS STRUCTURE**, Short Answer Type Question No. 5.

12 Write short notes on any three of the following:

- a) Addressing modes
- b) Static and dynamic memory
- c) Instruction pipelining
- d) Concept of programmed I/O
- e) Bus organization using tri state

s) See Topic: **INSTRUCTION SET**, Long Answer Type Question No. S(a).

b) See Topic: **MEMORY ORGANIZATION**, Long Answer Type Question No. 20(d).

c) See Topic: **INPUT-OUTPUT ORGANIZATION**, Long Answer Type Question No. 9(d).

d) See Topic: **INPUT-OUTPUT ORGANIZATION**, Long Answer Type Question No. 9(e).

c) See Topic: **BUS STRUCTURE**, Long Answer Type Question No. 2.

QUESTION 2018

Group-A

(Multiple Choice Type Questions)

1. Choose the correct alternatives of the following:

i) The principle of locality justifies the use of

- a) Interrupt
- b) Polling
- c) OMA
- d) Cache memory

if) Instruction cycle is

- a) fetch-decode-execute
- b) fetch-execution-decode
- c) decode-fetch-execution
- d) decode-execution-fetch

-) How many RAM chips of size (256K x 1 bit) are required to build 1M Memory? d) 8
aj 10 C) 32 .

iv) Maximum value on n bit 2's complement number is ...

- a) 2^n
- b) $2^n - 1$
- c) 2^{n-1}
- d) Cannot be said

v) Micro instructions are kept in

- a) Main memory
- b) Control memory
- c) Cache memory
- d) Secondary storage

POPULAR PVQLIQATIQNS

- vi) The maximum propagation delay for n-bit CLA is o) 6 A
a) 6. b) A 11 d) 11
- vii) The cylinder in a disk pack is
a) collection of all tracks in a surface different surfaces of disks
b) logical view of same radius tracks on
c) collection of all sectors in a track
d) collection of all disks in the pack
- viii) For which of the following multiplier numbers in Booth's algorithm maximum no. of additions and subtractions are required?
a) 01001111 b) 01111000 c) 00110111 d) 01010101
- x) How many memory locations can be addressed by a 32-bit computer?
a) 64 KB b) 32 KB c) 4 GB d) 4 MB
- x) The addressing mode of an instruction is resolved by
a) ALU b) DMA controller c) CU d) program

Group-B

(Short Answer Type Questions).

Multiply decimal number (-17) and (-9) using Booth's multiplication method with step by step explanation.

Sec Topic: COMPUTER ARITHMETIC, Short Answer Type Question No. 17

3. Explain stack based CPU.

Sec Topic: MEMORY ORGANIZATION, Short Answer Type Question No. 6.

4. What are the limitations of direct-mapped cache? Explain with an example, how it can be improved into set-associative cache?

Sec Topic: MEI\10RY ORGANIZATION, Short Answer Type Question No. 13.

5. Define speedup, efficiency and throughput of a pipelined processor. Design a 4-bit Combinational circuit decremener using four full adders.

I¹ Part: Stt Topic: INPUT-OUTPUT ORGANIZATION, Long Answer Type Question No. 6(c).

2. Part: Sec Topic: COMPUTER ARITHMETIC, Long Answer Type Question No. 4(b).

6. Explain with example Register Direct, Register Indirect and Base register addressing mode
Set Topic: INSTRUCTION SET, Short Answer Type Question No. 8.

Group-C

(Long Answer Type Questions)

arally, q, blic: block diagram or Computer System Why do peripherals need interface?"

1. H: S... -pkIN' I ODU "ION, 1.001 Answer Type Question No. 2.

n, -it, INPIJ -OffPUf ORG NilATION, Long An ..., Typ, Qut tioaNo-

b) A block set-associative cache consists of a total of 64 blocks divided into 4 blocks sets. The main memory can have 4096 blocks, each consisting of 128 words.

- i) How many bits are there in a main memory address?
- ii) How many bits are there in each of the TAG, SET and Word fields?

cc Topic: BUS STRUCTURE, Short Answer Type Question No. S.

c) What are 'write through' and 'write back' policies in cache memory?

See Topic: MEMORY ORGANIZATION, Long Answer Type Question No. 1 (a).

S a) Evaluate the arithmetic statement $X = (A * B) / (C + D)$ in one, two and three address machines.

b) Represent the decimal value - 7.5 in IEEE-754 single precision floating point format. Explain in brief about different memory access methods.

c) Explain Instruction Cycle with suitable flow chart.

a) See Topic: INSTRUCTION SET, Long Answer Type Question No. 2.

b) 1st Part: See Topic: COMPUTER ARITHMETIC, Short Answer Type Question No. 16.

2nd Part: See Topic: COMPUTER ARITHMETIC, Short Answer Type Question No. 18.

c) See Topic: CONTROL UNIT, Long Answer Type Question No. 2.

ga) If a CPU has 16 bit address bus and 8 bit data bus draw the connection diagram for this CPU with four 256x8 RAM and one 512x8 ROM.

b) Design a 4-bit ALU capable of performing 14 different micro operations including logical, arithmetic and shifting operations.

c) What is the difference between vectored and non-vectored interrupt? Average memory access time depends on which factors?

a) See Topic: MEMORY ORGANIZATION, Long Answer Type Question No. 10(a).

b) See Topic: COMPUTER ARITHMETIC, Long Answer Type Question No. 8.

c) 1st Part: See Topic: INPUT-OUTPUT ORGANIZATION, Short Answer Type Question No. 3(2^d Part).

2nd Part: See Topic: MEMORY ORGANIZATION, Short Answer Type Question No. 14.

10 a) Explain memory-hierarchy.

b) Can a Read Only Memory be also a Random Access Memory? Justify your answer.

c) Discuss the concept of associative memory unit using suitable example.

d) Define latency time in a memory.

a) & b) See Topic: MEMORY ORGANIZATION, Long Answer Type Question No. 19(a)& (b).

c) See Topic: MEMORY ORGANIZATION, Long Answer Type Question No. 2.

d) See Topic: MEMORY ORGANIZATION, Short Answer Type Question No. 11.

11. Write short notes on *any three* of the following:

- a) Instruction Format
- b) Carry Look-ahead Adder
- c) Design of 4-bit ALU
- d) RISC
- e) Overflow in Fixed-point Representation

POPULAR PUBLICATIONS

- a) See Topic: INSTRUCTION SET, Long Answer Type Question No. 5(c).
- b) See Topic: COMPUTER ARITHMETIC, Long Answer Type Question No. 12(d).
- c) See Topic: COMPUTER ARITHMETIC, Long Answer Type Question No. 12(e).
- d) See Topic: INSTRUCTION SET, Long Answer Type Question No. 5(d).
- e) See Topic: COMPUTER ARITHMETIC, Long Answer Type Question No. 12(f).

QUESTION 2019

<group A

(Multiple choice type Question)

- 1 Choose two correct alternatives for any one of the following:
 - i) A source program is usually in
 - a) Assembly language
 - b) Machine level language
 - c) High level language
 - d) Natural language
 - ii) In straight binary code, N bits or N binary digits can represent... different values.
 - a) 2^N
 - b) $2^{(N+1)}$
 - c) $2^{(N-1)}$
 - d) $2^N - 1$
 - iii) Maximum n bits 2's complement number is
 - a) ?An
 - b) $2^n - 1$
 - c) $2^{(n-1)} - 1$
 - d) Cannot be said
 - iv) The addressing mode, where you directly specify the operand value is
 - a) Immediate
 - b) Direct
 - c) Definite
 - d) Relative
 - v) Low is the effective address of base-register calculated?
 - a) By addition of base register contents to the partial address in instruction
 - b) By addition of implied register contents to the partial address in instruction
 - c) By addition of base register contents to the complete address in instruction
 - d) By addition of implied register contents to the complete address in instruction
 - vi) The instruction, Add R1, 45 does
 - a) Adds the value of 45 to the address of R1 and stores 45 in that address
 - b) Adds 45 to the value of R1 and stores it in R1
 - c) Reads the memory location 45 and adds that content to that of R1
 - d) None of these
 - vii) An 11-bit address generates an address space of locations
 - a) 1024
 - b) 4096
 - c) 2^{11}
 - d) $16,777,216$
 - viii) What is the maximum size of on-chip cache memory for a 32-bit processor?
 - a) " " " "
 - b) 2^{12}
 - c) Infinite
 - d) decided by manufacturer
 - ix) Which of the following is responsible for generating physical address from the logical address generated by CPU?
 - a) MMU
 - b) Cache
 - c) Overlays
 - d) 111

COMPUTER ORGANISATION

- x) The contention for the usage of a hardware device is called
a) structural hazard b) stalk c) deadlock d) none of these
- xi) The periods of time when the unit is idle is called as
a) Stalls b) Bubbles c) Hazards d) Both (a) & (b)
- xii) The OMA transfer is initiated by
a) Processor b) The process being executed
c) I/O devices d) OS

Group-B

(Short Answer Type Questions)

How can a full adder be implemented using half adders? Explain with proper circuit diagram.

See Topic: COMPUTER ARITHMETIC, Short Answer Type Question No. 13.

3. Explain the different kinds of data hazards in pipelining with suitable examples.

See Topic INPUT-OUTPUT ORGANIZATION, Short Answer Type Question No. 11.

4. (a) Briefly explain IEEE 754 format for floating point representation of numbers.

(b) Represent the decimal value -12.5 in IEEE single precision format.

a) See Topic: COMPUTER ARITHMETIC, Short Answer Type Question No. 4(a).

b) See Topic: COMPUTER ARITHMETIC, Short Answer Type Question No. 19.

5. (a) If a direct mapped cache has a hit rate of 95%, a hit time of 4ns, and a miss penalty of 100ns, what is the average memory access time?

(b) If an L2 cache is added with a hit time of 20ns and a hit rate of 50%, what is the new average memory access time?

See Topic MEMORY ORGANIZATION, Short Answer Type Question No. 15.

Group-C

(Long Answer Type Questions)

6. Write short notes on the following:

a) Resorting Division Algorithm

b) Direct Memory Access

c) IEEE double precision format

a) See Topic: COMPUTER ARITHMETIC, Long Answer Type Question No. 12(g).

b) See Topic: INPUT-OUTPUT ORGANIZATION, Long Answer Type Question No. 9(f).

c) See Topic: COMPUTER ARITHMETIC, Long Answer Type Question No. 12(h).

7. a) Explain the difference between full associative and direct mapping technique.

b) Explain the write back and write through policies in cache.

c) Cache memory has 2K blocks. Block size is of 4 words = 16 bytes. 32 bit address is provided. The machine is byte addressable.

? What is the bit length for each field in Direct Mapping?..

What is the bit length for each field in 2-way set associative mapping?

|||) What is the bit length for each field in 4-way set associative mapping?

a) S .1 rgi : 1 , f(IJR(.#, \f, 1z i, f IO , IA g **Ans** er T₁ pe Question o. I(b) (In Part).

b) t'f<JfJi: 1 f(Jf{\ fJJ r, rz \| fJO.'.. l.ftJng n v. er T) pe Qu tion 'i"o. 1(a).

c) S . ur,ic. J .., 1(J Ji< ..;\. JZ,\' f I,Q . hort ns. er T) pt Question a. 16.

8.) V a ., t rJ "h'i e-- oe. 1tt.;5., s,c, a e^ '0 artl rre,rrO"Yf'1apoed 110?

b1 zpt., r C, r,r Sc..r erE:"::h
r,J xp'aJn r: ffc::rrJ"t raza.ds :. p i:,e "·"9

J **ei**' opic:; r pr .r_r;r PL r RG :---IZ T IO." Long answerType Question o. 4.

hJ S ,1 ,pl:r::J tp .r r;r PL ORG '\JZ,\ IO!', hort n er T peQu esion. o. 12.

i) Se .r ,ni ; J .p 1-) . p(T (JRG A IZA N O:- hort ns er T peQ uestio: o. 1.

9 a) I /rat are tre dfferent addres-s r g rrodes?

b) E1plarn ,: 1.1f8b e1amp1es of each of e modes.

r.:, E,arua tr': fol & 11mg arithrre -c expression ;, o (Q three address , 0i) tv,o addresses, {ir,) addrE:\$5 ard ti 11 zero address instruction format X =(A+ B) (C + D).

aJ & hJ 5ee 'f opic: !"-STR c f1O. •...,-T, Long answer T pe Question o. 1.

c;<:e.c-opic.: 1-., TRr;c.-/O.iSf , Short answer T pe Question No. 9,

10. a) What are the differences ber,een RISC and CISC?

b) D1rded 43 b:t 11 using t lor-restoring algorithm (The tracing table must be shovm clearty)

CJ Nhat rs meant by 0 1erflo11 and undetlo 11in signed magnitude representation of numbers?

t.1 J S-ee. opic.: ;,,,-S-/-R' C'f IQ.-; SET, Short answer Type Question o. 3.

hJ& cJ \ee .rr,pic: C.O.\.1.?CTER ARITIIMETI , Long answer T pe Question No. ll(a) & (b).

- (b) CPU: It acts as the brain of the computer and performs the bulk of data processing operations in a computer. The two main units of a CPU are the Arithmetic Logic Unit and the Program Control Unit. The important parts of CPU are:
 - (i) Arithmetic Logic Unit (ALU): It performs instructions related to arithmetic operations like ADD, SUB, MUL etc. and logical operations like AND, OR etc.
 - (ii) Program Control Unit (PCU): It interprets & sequences instructions i.e., interprets & sequences which instruction in a program is to be executed first.
 - (iii) Register Sets: These are collections of registers that store data.
- (c) **Input-Output** (I/O) Unit: This unit provides an efficient mode of communication between the central system (computer) & the outside environment. Through the I/O unit, programs & data must be entered into computer memory for processing & results obtained from computations must be recorded or displayed to the user. • Operating Systems:
An Operating System is a program (or system software) that acts as an intermediary between a user of a computer & the computer hardware. Its purpose is to provide an environment in which a user can execute programs conveniently. So, an O.S. helps to use the computer hardware in an efficient manner.
- Functions of an Operating System: O.S. has the following functions.
 - (a) O.S. coordinates the efficient use of the hardware:
Operating System controls & coordinates the use of the hardware among the various application programs (like compilers, database systems, games etc.) for the various users (like people, machines, and other computers).
 - (b) O.S. provides an environment "in which other programs can do useful work":
Operating System provides the means for the proper use of the resources (like hardware, software & data) of a computer system in the meaningful & smooth operation of the computer.
 - (c) O.S. acts as a resource allocator:
O.S. manages the various resources (hardware and software) of a computer system & allocates them to specific programs & users as necessary for their tasks.
 - (d) O.S. acts as a control program.
As a control program O.S. focuses on the need to control the operations of the various inputoutput devices & user programs i.e. it controls the execution of user programs to prevent errors & improper use of the computer.
- on von Neumann Concept:
Neumann proposed the idea, known as the stored-program concept, deals with making the programming process easier by representing programs in a form such that they can be suitably stored in memory alongside the data. So, a computer could get its instructions by reading them

then in front memory & also a program could be set or altered depending on the memory values. Thus Von Neumann introduced the key concept of stored programming (i.e. Programs & their data were located in the same memory) in the first generation computers. Neumann published the idea in 1945 while proposing a new computer, the EDV (Electronic Discrete Variable Computer) and in 1946.

CO-3

Multiple Choice Type Questions

[WBUT 20071

1. The basic principle of the von Neumann computer is

- a) Storing program and data in separate memory
- c) Storing both program and data in the same memory
-) Using a large number of registers

Answer: (c)

(WBUT
20101

2. From a Source Code, a compiler can detect

None of these

- b) Logical errors
- a) Run-time error
- d) Syntax error

Answer: (c)

3. How many minimum, NANO gates are required to make a flip-flop? (WBUT 20101

Answer: (c)

4. The basic principle of a Von Neumann computer is (WBUT 20131

- a) storing program and data in separate memory
- b) using pipeline concept
- c) storing both program and data in the same memory using a large number of register

Answer: (c)

circuit in ALU is

- a) entirely combinational

[WBUT 20131

- c) entirely sequential

- b) combinational cum sequential

- d) none of these

Answer: (a)

5. The logic

POPULAR PUBLICATIONS

6. The Von-Neumann bottleneck is a problem, which occurs due to [WBUT 20141]

- a) small size main memory
 - b) speed disparity between CPU and main memory
 - c) high speed CPU
 - d) malfunctioning of any unit in CPU

Answer: (b)

7. The circuit used to store one bit of data is known as

8 SIMD a) refers represents to a computer an organization system that capable of processing several programs [WBUT 2016] at the **same time**

- b) represents organization of single computer containing a control unit, processor unit and a memory unit
 - c) includes many processing units under the supervision of a common control unit
 - d) none of these

Ans"cr: (c)

9 The ALU makes use of to store the intermediate results. [WBUT 2017]

Accumulators b) Resistors c) Heap d) Stack

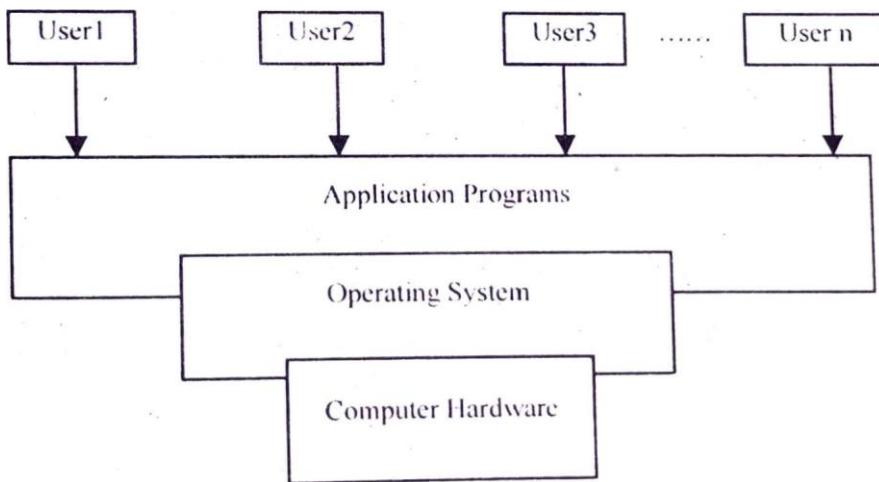
Answer: (a)

Short Answer Type questions

1. What is the role of operating system? [WBUT 2002, 2003, 2005, 2006, 2008, 2011]

Answer:

In order to use computer hardware in an efficient manner, each computer must have an operating system in it. An Operating System is a program (can also be considered as a system software) that acts as an intermediary between a user of a computer & the computer hardware.

Block Diagram

Abstract view of the components of an Operating System

O.S. has the following functions.

(a) O.S. coordinates the efficient use of the hardware

Operating System controls & coordinates the use of the hardware among the various application programs (like compilers, database systems, etc.) for the various users (like people, machines, and other computers).

CO-5

(h) O.S. provides an environment within which other programs can do useful work
Operating System provides the framework for the proper use of the resources (like hardware, software & data) of a computer system in the meaningful & smooth operation of computer.

allocates O.S. manages the specific various resources (hardware and software) of a computer system to programs & users as necessary for their tasks.

O.S. acts as a control program

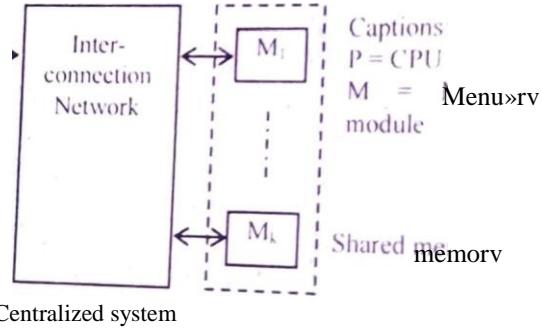
As a control program O.S. focuses on the need to control the operations of the input-output devices & user programs i.e. it controls the execution of user programs, prevent errors & improper use of the computer.

2. Compare between centralized and distributed architecture. Which is

POPULAR PUBLICATIONS

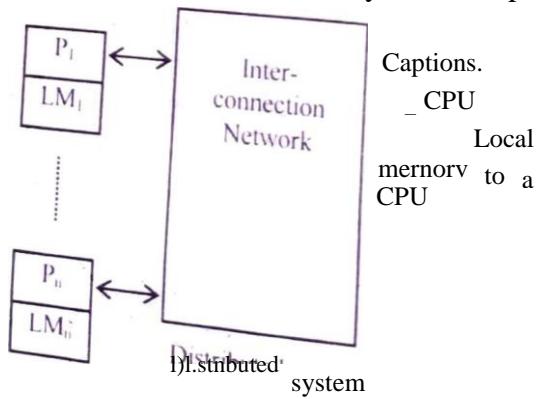
architecture among them and why? [WBUT the 2014]best Answer:

In centralized architecture, all the processors access the physical memory uniformly. All processors have equal access time to all memory words. The degree of interactions among tasks is high. Thus probability of bus conflicts is high, because of frequent sharing of codes between two processors. The architecture is shown in the following figure:



Centralized system

In distributed system), a local memory is attached each processor. All local memories distributed throughout the a global shared accessible by all processors. A memory word access time varies with the location of the memory in the shared memory. The degree of interactions among tasks is less. Thus probability Of bus conflicts is also less. The distributed system is depicted in figure.



CO-6

...te

faster to access processor a local memory takes with longer a local due processor. to the
~~interconnection~~
 added 1 lie access delay through of retoote the memory intertachCd to net Otl

1 the (list tibuted is taster and in this regard, it is Ntter.

3 Explain the role of operating system in a computer system. (WBUT 20171

~~AN~~ operating sy stelli three null n functions:

- (1) the ~~computer's~~ resources, such as the central processing unit, memory, disk ~~drives~~, and printers,
- (3) establish a user interface, and
- ~~an~~ selecte and prox ide services for applications software.

Long Answer Type Questions

What is Von Neumann architecture?

[WBUT 2002, 2003, 2004, 2007, 2008, 2009, 2011, 2012]

What is Von Neumann bottleneck?

[WBUT 2002, 2003, 2004, 2006, 2007, 2008, 2009, 2011, 2012]

OR,

What is von Neumann bottleneck? How can this be reduced? [WBUT 2015, 2016]

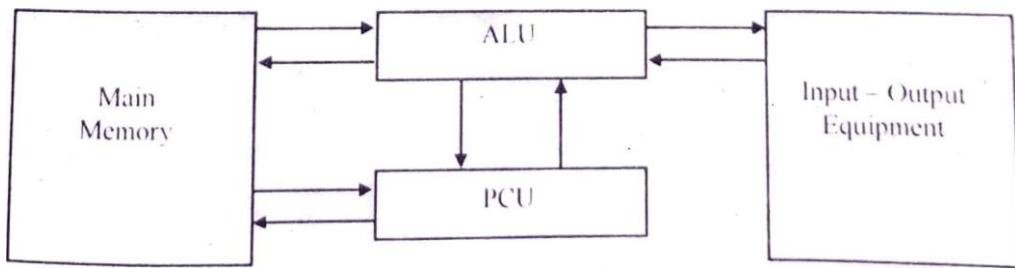
Answer:

1st Part:

John Von Neumann was a mathematician who was a consultant on the ENIAC project (Electronic Numerical Integrator and Computer), the world's first general-purpose electronic digital computer.

Von Neumann proposed the idea, known as the stored-program concept, which makes programming easier by representing programs in a form such that they can be suitably stored in memory alongside the data. So, a computer could get its instructions by reading them from memory & also a program could be set or altered depending on the memory values.

The computer designed based on the idea of stored-program concept proposed by Von Neumann is known as the IAS computer. It was designed at the Princeton Institute for Advanced studies in 1952.



Structure of the IAS computer

CO-7

POPULAR PUBLICATIONS

Main Features of the IAS Computer

- (a) A main memory, which stores both data & instructions.
(b) An arithmetic-logical unit (ALU) capable of operating on binary data.
(c) A control unit, which interprets the instructions in memory & causes them to be executed.
(d) Input & output (I/O) unit operated by the control unit.
(e) The memory is read-write in nature.
(f) The contents of this memory are addressable by location.
(g) Execution generally occurs in a sequential fashion from one instruction to the next.

Illus All of today's computers, (on generally, having the same general structure & function are referred to as Von Neumann machines and this design is referred to as the von Neumann architecture.

2nd Part:

Bottleneck in Von Neumann 'architecture':

Since the CPU has much higher speed than the main memory (RAM), the CPU has to wait longer to obtain a data-word from the memory. This CPU-memory speed disparity is referred to as Von Neumann Bottleneck.

This performance problem is reduced by using a special type fast memory called cache memory between the CPU and main memory. The speed of cache memory is almost same as the CPU, for which there is almost no waiting time of the CPU for the required data-word to arrive. Another way to reduce the problem is by using special type computers known as Reduced Instruction Set Computers (RISC). The intention of the RISC computer is to reduce the total number of memory references made by the CPU; instead it uses large number of registers for same purpose.

- Bandwidth between CPU and memory is very small in comparison with the amount of memory.
- The address modification scheme in the IAS computer was inefficient and so, to restart a program, the original undefined program must be reloaded into memory.
- No facilities were provided for structuring e.g. instructions to link modules such as subroutines (procedures) that implement frequently used

- Floating-point was not implemented due to the cost of hardware
- As in each word instructions were stored, hence the program control unit

program
program

POPULAR PUBLICATIONS

needed
hit and the .

2 Describe the function of Major Components of a digital computer with diagram.
[WBUT 2003, 2007, 2011]

OR,

Draw a block diagram to illustrate the basic organization of computer system and explain the function of various units. OR, [WBUT 2006, 2013]

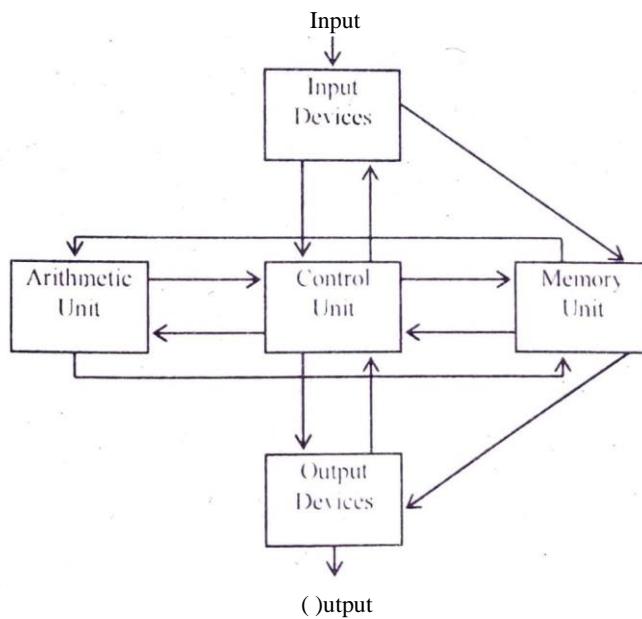
Draw a diagram for digital computer. [WBUT 2012] OR,

Explain the basic block diagram of Computer System. [WBUT 2018] MIS\$cr:

A digital computer system consists of an interconnected system of processor, memory and input output devices.

The processor is also called central processing unit. It is the heart of any computer, which actually does the whole job of program execution. The central processing unit used in micro computers is also called microprocessor. Main memory and central processing unit are mounted on a single board called the mother board. For the program executing the source program that is the program written by the user and the data required for it to be stored in the computer memory. Only then central processing unit runs the program.

The second part of the computer component is main memory which is referred to as memory, primary storage or common storage. It holds the source code or program. A program is defined as an ordered set of instructions which can be used to solve a problem. It holds data which may be input data. The memory unit consists of many thousand storage locations or cells. Each cell is a tiny device which stores the state of binary digits as zero or one. This state of the cell represents a single bit. The word bit is derived from binary digits.



POPULAR PUBLICATIONS

CO-9

Important parts of the CPU are:

- Arithmetic and logic unit: It performs instruction related arithmetic OPeration
- Program control unit: It interprets and sequences which instructions in a Program is to be executed first.
- Register set: these are collections of registers that store data.

3. Write short notes on the following:

[WBUT 20051]

a) Von Neumann architecture

OR,

IAS computer

[WBUT 2011]

b) Data flow architecture

[WBUT 20091]

Answer:

a) IAS computer:

Refer to Question No. I (I/ Part) of Long Answer Type Questions.

b) Data flow architecture:

Dataflow architectures do not have any program counter and conceptually depend on whether input arguments to instructions are available (i.e. only if an operand is available instructions are executed), the used executability Von Neumann of instructions architectures is determined or the control flow is in

contrast to the conventional architectures. Dataflow architectures find their usage in specialized hardware used in digital signal processing, data warehousing, network routine, parallel computing and graphics processing purposes.

COMPUTER ARITHMETIC

Chapter at a Glance

Arithmetic Logic tJnit: The Arithmetic Logic Unit or ALU performs arithmetic and logical operations on data in a digital computer.

The various other units and devices of the computer system actually bring data into the ALU for its processing and then takes the results out from the ALU. The ALU operation is mainly based on the use of simple digital logic devices that store binary digits to perform various simple Boolean logic operations.

So to tell in brief, ALU is an important component in the CPU of the digital computer that performs various arithmetic and logical operations. ALU actually gets data and supplies the processed data from and to the various other units.

The various registers (temporary storage locations within the CPU) connected by signal paths to the ALU, actually hold the data sent to the ALU for computations and also the results of the computations. The various flags (whose values are stored in CPU registers) are set by the ALU as the result of an operation. The operation of the as well as the data movement into and out of the ALU is controlled by the signals provided by the control unit.

- Diagrams:

Figure 1, shows the ALU inputs and outputs, whereas figure 2, shows the CPU and its various components (control unit is not shown here).

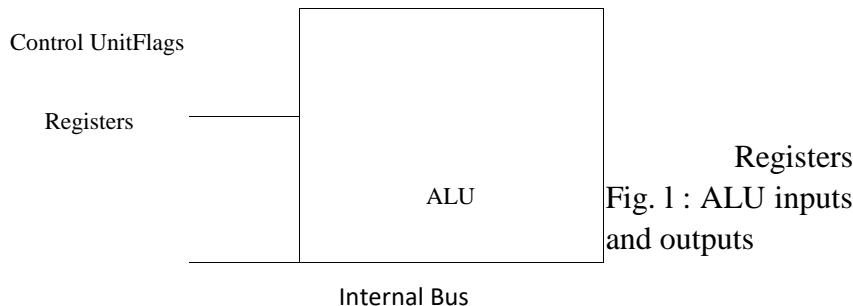


Fig. 1 : ALU inputs
and outputs

POPULAR PUBLICATIONS

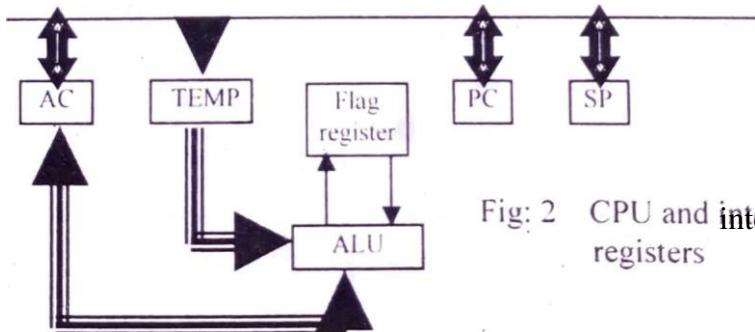


Fig: 2 CPU and internal registers

- IEEE representation of floating point numbers: All modern computers use the floatinoPoint representation that was specified in IEEE standard 754. Here as was discussed before numbers are represented by a mantissa and an exponent.

Out of the multiple number of bit widths specified by IEEE standard 754 for floating-point numbers, single-precision and double-precision widths are the most commonly used widths.

CO-II

Figure 3 shows the two formats. single-precision numbers are 32-bits long with 8-bits exponent, 23-bits of fraction and 1 sign-bit. whereas double-precision numbers are long with 11-bits of exponent, 52-bits of fraction and 1 sign-bit.

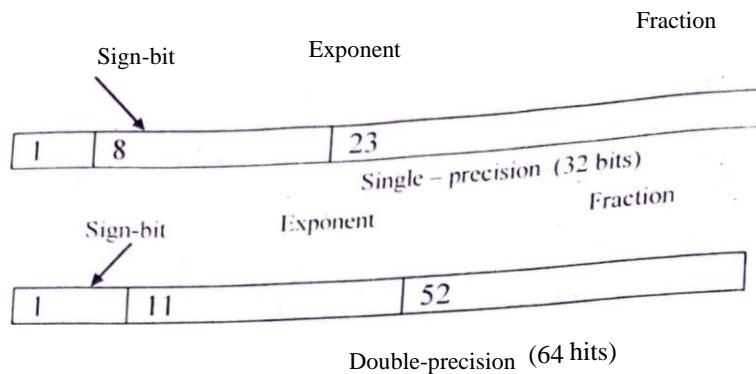


Fig: 3 IEEE 754 Floating-point formats

- Carry look-ahead adder: This circuit basically speeds up the generation of carry signals. The logic expressions for (s_i) and carry-out (C_i) of stage i are:

$$s_i = A_i \oplus B_i \oplus C_{i-1} \text{ and } C_i = G_i + P_i A_i B_i$$

where $G_i = A_i B_i + A_i C_{i-1} + B_i C_{i-1}$ and $P_i = A_i + B_i$.

The expressions G_i and P_i are called the generate and propagate functions for stage i respectively. So if the generate function for stage i is equal to 1 i.e. if $(G_i = 1)$, then $C_i = 1$ (when both A_i and B_i are equal to 1).

- The propagate function (P) means that an input carry will produce an output carry when either of A or B is 1. So all G and P functions can be formed independently and in parallel in only one logic gate delay. Now, if the propagate function can be realized as $P = A_1 B_1 + A_1 C_{i-1} + B_1 C_{i-1}$ then a simple circuit can be derived using a cascade of two 2-input XOR gates (to realize 3-input XOR function).

Booth's multiplication algorithm: Booth's algorithm provides a procedure by which binary integers in signed-2's complement representation (i.e. multipliers can be positive or negative) can be multiplied.

Division Algorithm: Division of two fixed-point binary numbers represented in signed-magnitude form is done by successive compare, shift and subtract technique. However in division, it may give rise to an overflow result i.e. if the expected quotient is of n -bits but the actual quotient comes as $n+1$ bits then that condition is an overflow condition, which must be taken care of.

POPULAR PUBLICATIONS

CO-N

Multi le Choice e uestions

1. With 2's Complement representation, the range of values that can be represented on the data bus of an 8 bit micro-processor is given by•

- a) -128 to + 127
- b) -128 to + 128 [WBUT 2003, 2012]
- c) -127 to 128
- d) -256 to + 256

Ans»er: (a)

2. When signed numbers are used in binary arithmetic, then which one of the following notations would have unique representation for zero?

[WBUT 2003, 2007, 2008, 2009]

- a) Sign Magnitude
- b) I's complement
- c) 21s complement
- d) None

Ans»er: (c)

3. Subtractor can be implemented using [WBUT 2006, 2012, 2015]

- a) adder
- b) completer
- c) both (a) & (b)
- d) none of these

Answer: (c)

4. Maximum n bit 2's complement number is [WBUT 2007, 2009, 2015, 2018]

- c) $2^n - 1$
- d) Cannot be said

Ans»er: (c)

5. Adding 011011012 to 101000102 in 8-bit 2's complement binary will cause an overflow:

Answer: (c)

- a) True
- b) False

Answer: (b)

6. The conversion of (FAFAFA)₁₆ into octal form is

- a) 76767676
- b) 76575372
- c) 76737672

Answer: (b)

7. Which logic gate has the highest speed?

- a) ECL
- b) TTL
- c) RTL

8. Booth's algorithm for computer arithmetic is used for

- a) multiplication of numbers in sign

magnitude form

- b) multiplication of numbers in 2's complement form
- c) division of numbers in sign magnitude form
- d) division of numbers in 2's complement form

Answer: (b)

9. The conversion (FAFAFB)₁₆ into octal form is

- a) 76767676
- b) 76575372
- c) 76737672

Answer: (d)

co-13

PNBUT 2007, 20091

[WBUT 2008, 20131

- d) 76727672

[WBUT 20091 d) DTL

PNBUT 2009, 2011)

[WBUT 20091 d) None of these

POPULAR PUBLICATIONS

representation have?

- a) 30 b) 32

Answer: (d)

c) 60

[WBUT
20091

11. The logic circuit in ALU is

- a) Entirely combinational
c) Combinational cum sequential

Answer: (c)

b) Entirely sequential

d) None of these

[WBUT 20111

12. Equivalent hexadecimal of (76575372)₈ will c) be

d) FAAFAF

10. A decimal no. has 30 digits. Approximately, how many would d) 90the binary

FFFAAA

- a) FAFAFF b) FAFAFA

Answer: (b)

13. If you convert (+46.5) into a 24 bit floating point binary number following IEEE convention, 00011100what would b) be 0000011the exponent? c) 1100010 d) none [WBUT of these20131

a)

Ansm•r: (d)

14. The maximum number of additions and subtractions are required for which of the following multiplier numbers in Booth's algorithm [WBUT 2014, 20151

- a) 01000 1111b) 0111 1000 c) 0000 1111 d) 0101 0101

Answer: (d)

15. By logical left-shifting the content of a register once, its content is [WBUT 20151

- a) doubled b) halved

c) both (a) and (b)

d) no such decision can be made

Answer: (d)

16. Floating point representation is used to store

- a) Boolean values b) Whole numbers [WBUT 2016]
- c) Real numbers
- d) Integers

Answer: (c)

17. A given memory chip has 12 address pins and 4 data pins. It has the number of locations.

- a) 2^4 b) 012 [WBUT 2016]

Answer: (b)

18. (2FAOC)16

- a) (195084)₁₀ [WBUT 2016]
- c) Both (a) and (b)

b) (0010111101000001100)₂
d) None of these

Answer: (b)

19. In a normal n-bit adder, to find out if an overflow has occurred, we make use of

a) AND gate

b) NAND gate

Answer: (d)

c) NOR gate

[WBUT 2017]

d) XOR gate

20. For additions which of and the subtractions following multiplier are required?numbers in Booth's algorithm [WBUT maximum2018]

no. of a) 01001111

b) 01111000

c) 00001111

d) 01010101 Answer: (c)

Short Answer e questions

1. Explain the relative advantages & disadvantages of parallel 2002, adder 2003, 2006, over 2112]serial adder.

ins" er:

Advantages of parallel adder: It is faster than the serial adder.

Disadvantages q/parallel adder: nnain disadvantages of parallel adder is the propagdion delay ofcarry bit one full adder to next higher position full adder. Sufficient time mu:t be llowed so that carry bit

POPULAR PUBLICATIONS

produced by the adder of the LSB will be Propagate through the adder and be available at the next higher position full adder before the addition is performed..

Circuit complexity is more than serial adder.

2. Compare Restoring & Non-Restoring Division algorithms.

[WBUT 2003, 2005, 2006, 2007]

Answer:

Restoring division.

Restoring division operates on fixed-point fractional numbers and depends on the following assumptions:

$$D < N$$

$$0 < N, D < 1.$$

The quotient digits q are formed from the digit set {0, 1}. Non-restoring division

Non-restoring division uses the digit set } for the quotient digits instead of }.

Restoring division technique is the hardware method of performing division operations. Here after each division step, the partial remainder obtained, restored by adding the divisor to the negative difference. This is done to get back the original AC value or to restore the value after every division step.

Non-restoring division technique, if the difference is negative then the divisor is not added directly to the partial remainder. It is added only after shifting the negative difference to the left i.e. suppose while performing division by restoring division technique, subtraction of the divisor content in D from that of A leads to a negative result (i.e. an unsuccessful subtraction). Still the value of A is to be restored. This is however time consuming and can be seen as an unnecessary overhead. This of the restoring division technique can be avoided in the non-restoring division technique

CO-15

co-16

What is the necessity of guard bits?

Answer:

Guard bits are additional or extra bits present in the ALU registers that load the exponent and significant of each operand prior to a floating point operation. Guard bits are basically used to pad out (i.e. to add or place) the right end of the significant with extra 0's to keep the length of the numbers fixed.

3. a) Briefly explain the IEEE 754 standard format for floating point representation. [WBUT 2007, 2009, 2017]

3. What are guard bits?

20171

4 a) Briefly explain the IEEE 754 standard format for floating point representation.

All modern computers use the floating-point representation that was specified in IEEE

The floating-point number can be stored in this format in any type of computer.

30 29 04 23

Mantissa

Bit 3— sign bit

Bit 3 to 23 exponent

22 top mantissa

S is used to represent sign of a number.

S = 0 for positive number

S = 1 for negative number

M is the Mantissa which is a fraction and 23 bit long. The Mantissa is normalized that it does not contain zero in its leading bits.

b) How NaN (Not a Number) and Infinity are represented in this standard.

[WBUT
2007]

Answer

NaN: NN or Not a Number is the symbol for any invalid operation result. For example, dividing by 0 or subtracting an infinite value another would produce invalid results, which

POPULAR PUBLICATIONS

would be represented by NaN. A NaN result would allow an user to recheck a ccision and figure out the problem.

Infinity: In exponent of all Is and a fraction of all Os are used to denote the values Of +infinitynd —infinity. The sign bit is used to distinguish between negative and positive liltinity.

5. write in IEEE 754 floating point representation in double precision.

Write +71in IEEE 32 bit format. OR,

[WBUT 20091

[WBUT 2015]

Alls" cr:

Bit • 31 (sign bit) for the given number is () (positive sign).

Bits 23 to 30 (exponent field):~~1000000~~

Bits 0 to 22 (significant): 1.1 IOOOOOOOOOOOOOOOOO.

Hence. converting to hex, the result becomes 4()E()()00.

6. What are the advantages of CLA over ripple carry adder? [WBUT 2011] Answer:

A cascaded connection of n full adder blocks, can be used to add two n-bit numbers. Since the carries will propagate or ripple through this cascade, this configuration is called an Ripple ('Ripple' Adder).

A fast adder circuit will speed up the generation of the carry signals. In case of a carrylook ahead adder, the carry does not have to depend explicitly on the preceding one and can be expressed as functions of relevant addend and augend bits. So, the overall delay is much lesser than the conventional parallel adder.

7. Using Booth's algorithm multiply (—12) and (+6). [CWBUT 2011]

OR,

Multiply (—12) and (+6), using Booth's multiplication algorithm. [WBUT 2017]

Answer:

The binary representation of (—12) = 1001 10 = 00110 (multiplier)

The binary representation of 12 = 01 100

The binary representation of -12 = 2's complement of 01 100 = 10100 (multiplicand)

Q _n Q _{n+1}	BR=01000 BR4 1 = 10100	AC		Q _{n+1}	
	Initial	0 0000	001 10	0	1 0 1
0 0	ashr AC,QR (Including 0 1)	10000	0001 1	0	

1 0	Subtract BR	00100	00011	0	I O O
		10010	00001		O i l
I 1	ashr AC,QR (Including	11001	00000		0 1 0
0 1	Add BR	00101			
		10010	1 0000	0	O O I
oo	ashr AC,QR (Including Q)•1)	11001	01000		0 0 0

[WBUT 2012]

The answer is: **110010**

P8. Apply BPUoBoth's TailgooNrlthm to multiply the two numbers (6)₁₀ and (-9)₁₀.
ASSUme

the multiplicand and multiplier to be 5 bits each. 6 Answer: 0

0 1 1 0 x * 1 0 1 recoded multiplier

$$\begin{array}{r} -1100-1 \\ \times 1010 \\ \hline \end{array}$$

$$\begin{array}{r}
 \text{Shift Only} \quad | \quad 110\ 01010 \\
 \text{Shift} \\
 \text{Add A} \quad + \quad 0110 \\
 \hline
 010\ 010 \\
 \circ 1 \\
 \circ 001\ 1\ 0 \\
 \hline
 0 \quad 10
 \end{array}$$

Add—A + I 1 0 1 Shift Only
 0

[WBUT
20131

9. Represent (-9.50) is 64 bit IEEE floating point representation.

Answer:

The IEEE-754

Bit 63: sign bit

Next I I-bit Shift
1023 fornn

| 1 1 0 0 1 0 1 0
| 1 0 0 1 0 1 0 -

54 biased exponent represented in excess-

Next 52-bit normalized Inantissa (magnitude), where the decinnal point is assumed to lie just on the right of the most significant I bit in the real number (integer + fraction). NOW, $-9.5 = -1001.1 = -1.0011 \times 2^3$ so, bit 63 = 1

$$\text{Exponent} = 1023 + 3 = 100000010B$$

Mantissa part = 001 10....0 (001 1 followed by 48 0's)

Hence the representation is C02300000000000H.

10. Explain IEEE single precision formats for representing — 10.5. (WBUT
20141

Answer:

$$10.5 = 1010.102 = 1.01010 \times 2^3 s = 1 e = 3 + 127 =$$

130 = 10000010 f= 01010 The single-precision representation is:

1 10000010 01010000000000000000000000000000

11. Convert IEEE 32-bit format 4040000016 in decimal value.

Answer:

[WBUT 20151

The given number in IEEE 32-bit format is 404000001

= 01 90 0000 0100 0000 0000 0000 0000

6

Since the leading bit is 0, the number is positive.

Next higher order 8-bit indicates the biased exponent

and it is (1000 00000): - 128

Therefore, the original exponent $E = E' - 127 = 128 - 127 = 1$

The leading bit in the binary point is 1, so the actual decimal number is $+(1.1)_2 \times 2^1 = +(11)_2 = +3_{10}$

The leading bit in the binary point is 1, so the actual mantissa is (1.1)

the actual mantissa is (1.1)

12. For Booth's algorithm, when do worst case and best case occur? Explain with example. [WBUT 2015, 2016]

Answer:

Worst case is one when there are maximum number of pairs of (01)s or (10)s in the multipliers. Thus, maximum number of additions and subtractions are encountered in the case.

Best case is one when there is a large block of consecutive 1s in the multipliers, requiring minimum number of additions and subtractions.

13. Show how to implement a full adder, by using half adders. [WBUT 2016]

Answer:

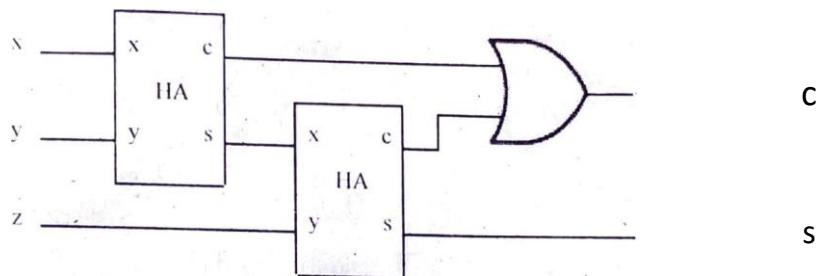


Fig: Block diagram of full-adder implementation via a pair of half-adders

A full-adder can be constructed from two half-adders and an OR gate, as shown in Figure below. The explanation of why this works is as follows. (In this paragraph, + denotes addition, not the OR operation.) Consider the addition of $x + y + z$. This can be grouped as $(x + y) + z$ where $(x + y)$ represents the output of the half-adder that receives x and y . This partial sum is added to z by the other half-adder, yielding the complete sum bit S . As for C , consider that there are two possible ways to make $C = 1$: first, if $x + y = 2$, then adding z can only make the total sum 2 or 3, and either way $C = 1$. In this case, the first half-adder's carry-out is a 1. Second, if $x + y = 1$, then C will be 1 only if $z = 1$ to make the total 2. In this case, the second half-adder's carry output will be 1. Thus we see that $C = 1$ if and only if at least one of the halfadders produces a carry-out.

Of 1. This corresponds to the OR of the two partial carry bits.

- 14 1J:p restorng method to divide 10100011 by 1011. [WBUT 2016] Answer:

(Unsigned numbers division)

Divide 163 by 11 using restoring division method.

Dividend, Q = 163 - 101 0001 1

Divisor, M = 11 = 0000101 1, wc = 111 10101

R PUBLICATIONS

Iteration	S' tpe/Act1011	I vldend/Quottent (Q)	Divisor/Relnark (M)
0	Initial values	Accumulator (A) 00000000	00001 O 1 I
	Shift left A. Q	00000001	$S_A = 1; Q_0 = 0$
	Subtract A-NI	00000001 1 1 10101 I 101 10 00001011	01000110 1000110 10
	Restore	00000001 1 1 10101 I 101 10 00001011	(2)
	Shift len A. Q	00000001	$S_A = 1; Q_1 = 0$
	Subtract, M	00000010	
	Restore	11110111 1 1 10101 00000010	10001100 0001 100 IL) Q _
	Shift left A. Q	00000101	$S_A = 1; Q_2 = 0$
	Subtract, M	00001011	
	Restore	1 1 10101 1 1 1010 00001011 00000101	0011 0 0 0 OOOI IQ OQ
4	Shift left sub A-M	00001010 1 1 10101	$S_A = 1, Q_3 = 0$
	Restore A+M	00001011	
		00001010	OOI IQOQQ
5	Shift left sub A-M	00010100 1 1 10101 00001001	01 10000 _
		00001010	$S_A = I (21 = ()$
6	Shift left Sub A-M	00010010 1 1 10101 00000111	10000 L _
		00000111	IOOQQQLL
7	Shift left Sub. A-M	00001111 1 1 10101 00000100	$S_A = 1; Q_5 = 0$
		00001111	0 0
		00000100	0
8	Shift left Sub. A—M	00001001	000 L LL
	Restore A—M	00001011 00001001	00 0 L L LO 0000 L L I O
		00001011	$S_A = 1; Q_7 = 0$
		00001001	QQO OL II O

POPULA

$$\text{Renulinder} = 9 \quad \text{Quotient} = 14$$

Elence 163 I 1 gives, Q = 14 and R = 9

15. Represent the decimal value (-7.5) in IEEE single precision format.

Represent format. the decimal value — 7.5 in IEEE-754 single precision floating [WBUT 2017] point

[WBUT
2018] co-20

COMPU

x nssyrcr:

The decimal number • 111.1 in binary 1 . 1 1 1 0 22 ' R-bit

The Inantissa M O. 1 1 1000 000000 000000 0000

The biased exponent $E' = E + 127 = 129$ 1000 0001

Since the number is negative. the sign bit S = 1 here fore. the IEEE single-precision (32bit) representation is

1	10000001	111000 000000 000000 000000
---	----------	-----------------------------

16. What is the difference between carry look ahead adder and ripple carry adder?

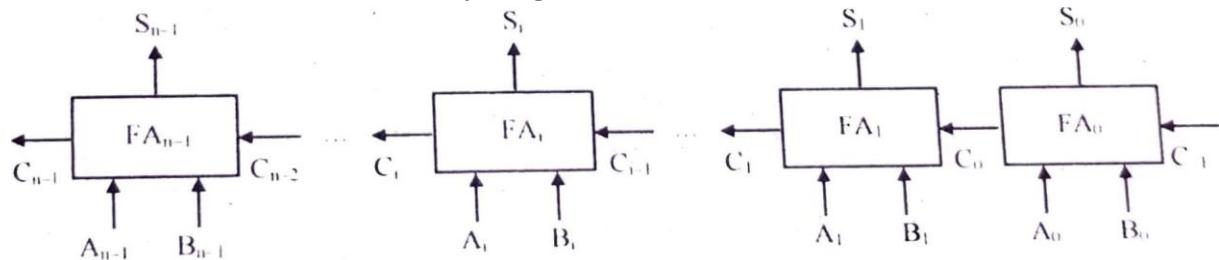
LWBUT 20171

Ans»cr:

A of ripple-carry adders is a sequence of standard full adders that makes it possible to add numbers that contain more bits than that of a single full adder. Each full adder has a carrying (Cin) and a carryout (Cout) bit, and the adders are connected by connecting Cout on step k to on step k+1.

Carry lookahead adder is faster than ripple carry adder (also known as carry propagation adder) since it consists of carry lookahead circuit and all its inputs given by G(x) and P(x) generator functions are calculated simultaneously.

But cost of Carry Lookahead Adder should be more since cost in digital logic means how many gates are using. what is the FAN-IN of those gates. So in carry lookahead adder we have carry lookahead circuit that contains many gates compared to carry propagation. To be precise no. of gates used in carry lookahead circuit = O(n²) which is much larger than no. of gates used in carry propagation adder which is O(n). So cost of carry lookahead adder is obviously larger.



Fio: I Block diagraml of an n-bit ripple-carry adder

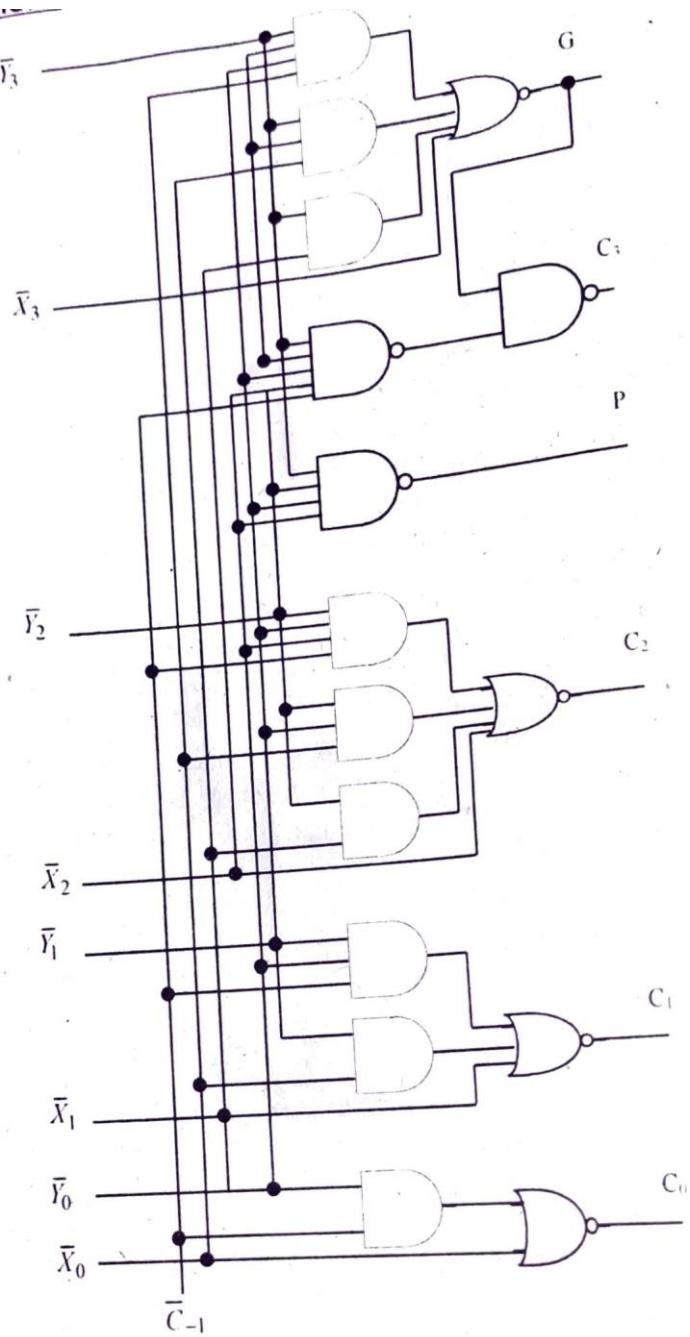


Fig: 2 Logic diagram of a 4-bit carry look ahead circuit

17. Multiply decimal number (-17) and (-9) using Booth's multiplication method with step by step explanation. CWBUT 20181

Ans"cr:

Multiplication betwuen —1 7 and —9 using booth's Algoritlun:

$$\begin{array}{r} 111011 \quad 11 \\ \times \quad \quad \quad -17 \end{array}$$

$\begin{array}{r} \times 111 \quad 1 \quad 11 \\ \hline \end{array}$

0 0 0 -1 1 0 0 -

Recorded

multipuer

In u Iti lier

Add

Add - A	+	0 0 0 1 0 0 0 1	
shift		0 0 0 0 1 0 0 0 1	
Shift only		0 0 0 0 0 1 0 0	
Shift only		0 1	
		0 0 0 0 0 0 1 0	
		0 0 1	

Add A

Shift	I 1 I 1 0 0 0 1 0 0 1
	1 1 1 1 1 0 0 0 1 0 0 1

$$\text{Add} - A \quad + 0 0 0 1 0 0 0 1$$

Shift	0 0 0 0 1 0 0 1 1 0 0 1
Shift	0 0 0 0 0 1 0 0 1 1
Shift only	0 0 1
Shift only	0 0 0 0 0 0 1 0 0 1
Shift only	1 0 0 1
	0 0 0 0 0 0 1 1 0 0 1

POPULAR PUBLICATIONS

0 0 0 0 0 0 0 1 0 0 1

1 0 0 1



(1001 1001) -- 153 (Ans.)

18. Explain in brief about different memory access methods. [WBUT 2018]

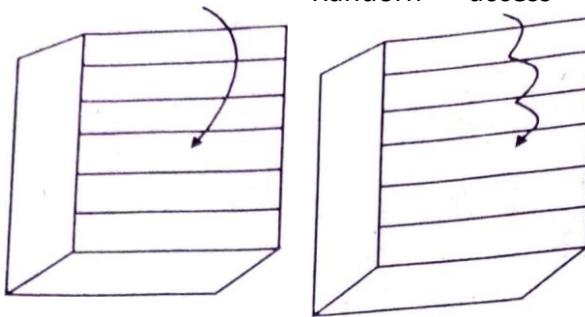
Answer:

In organisation, an access Inethod is a program or a hardssare tnechanistn that moves data between the coniputer and an outlying device such as a hard disk or a display terminals.

There are 2 ty pes of access Illethod l'hey a re-

i)

Random access and ii) Sequential access.



It is often used to describe data fields. Both type of files have advantages and disadvantages.

Rand0111 access is better than sequential access.

Long Answer Type Questions

1. Explain Booth's algorithm for multiplication of signed-2's Complement numbers using a flowchart & show how the multiplication is accomplished using a SUitable example. [WBUT 2003, 2004, 2005, 2007, 2009, 2010, 2012] OR,

Explain Booth's Algorithm with floW-chart and suitable example. [WBUT 2006, 2011]

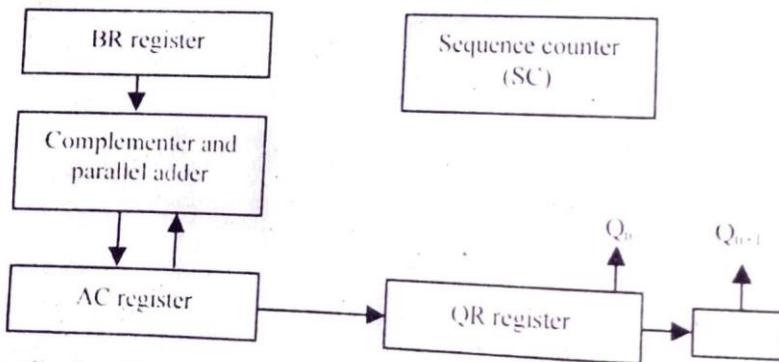
OR,

Present the Booth's algorithm for multiplication of signed 2's complement number in a flow chart and explain. [PUBUT 2017] Illustrate this with an example by multiplying $(-9) \times (-13)$. [WBUT 2010] Answer:

Booth's algorithm provides a procedure by binary integers in signed-Ys complement representation (i.e. multipliers can be positive or negative) can be multiplied.

Hardware configuration /br Booth's multiplication algorithm

Diagram



The figure shows the hardware implementation for Booth's algorithm. Register BR: It holds the multiplicand along with its sign-bit.

Register AC: It holds the partial product after each stage of multiplication. Register QR: It holds the multiplier along its sign-bit.

COM

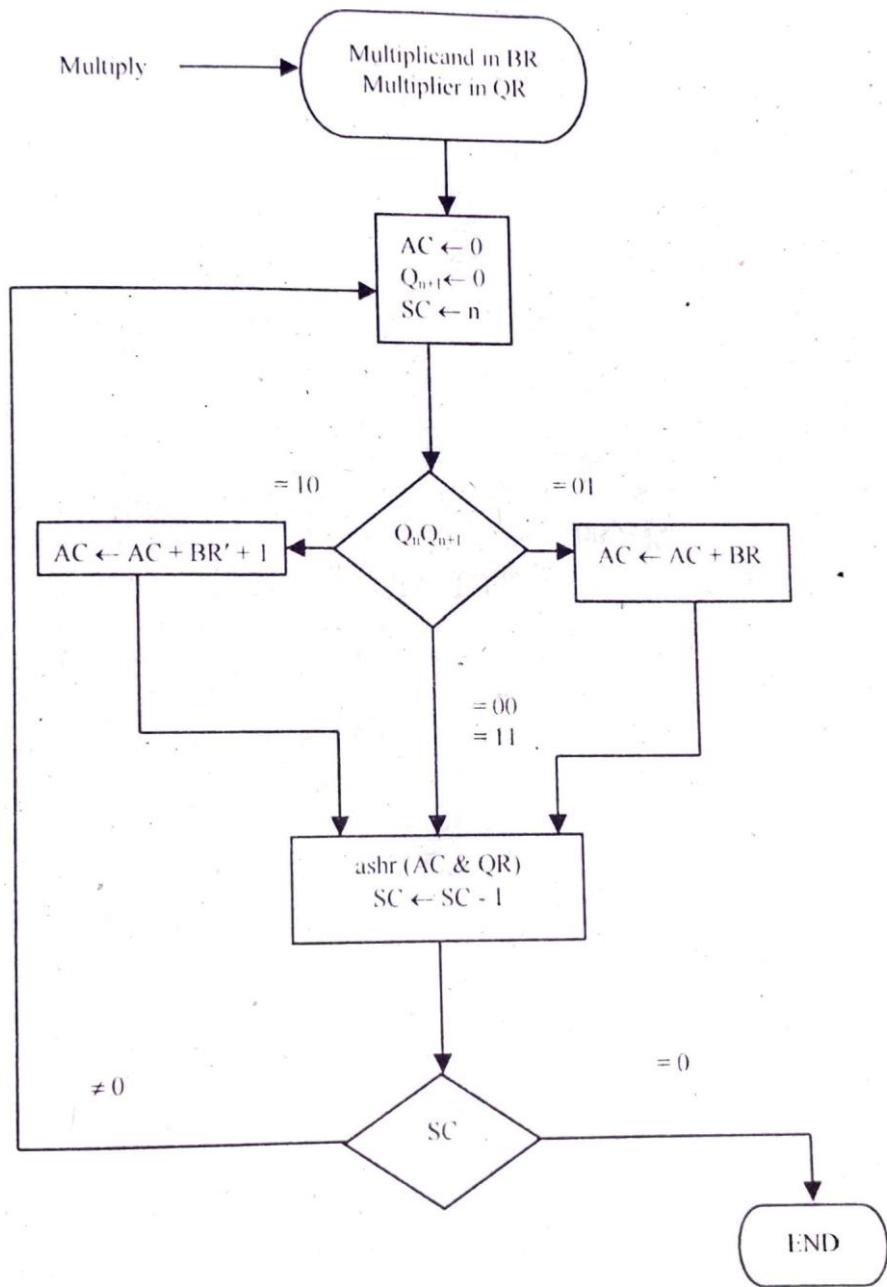
). It designates the least significant bit of the multiplier.

This is a flip-flop with the purpose of double-bit inspection of the multiplier.

Sequence Counter (SC): Keeps track of the number of bits in the multiplier and decrements by 1 after each multiplier bit is multiplied to the multiplicand.

Booth algorithm gives a procedure for multiplying binary integers in signed -2^n s complement representation.

POPULAR PUBLICATIONS



Example:

The binary representation of 9 = (01)001

The binary representation of -9 = 2's complement of 01001 = 1 1 (multiplicand) The binary representation of I

The binary representation of -13 = 2's complement of 01 101 -- 1001 1 (multiplier)

POPULAR PUBLICATIONS

POPULAR PUBLICATIONS

$Q_n Q_{n+1}$	$BR = 10111 \quad BR + I = 01001$	AC	QR	Q_{n+1}	SC
	Initial	0 0 0 0 0	1 0 0 1 1	0	1 0 1
1 0	Subtract BR	0 1 0 0 1			
	ashr AC,QR (Including Q_{n+1})	0 0 1 0 0	1 1 0 0 1	1	1 0 0
1 1	ashr AC,QR (Including Q_{n+1})	0 0 0 1 0	0 1 1 0 0	1	0 1 1
0 1	Add BR	1 0 1 1 1			
	ashr AC,QR (Including Q_{n+1})	1 1 0 0 1	1 0 1 1 0	0	0 1 0
0 0	ashr AC,QR (Including Q_{n+1})	1 1 1 0 0	0 1 0 1 1	0	0 0 1
1 0	Subtract BR	0 1 0 0 1			
	ashr AC,QR (Including Q_{n+1})	0 0 1 1 1	1 0 1 0 1	1	0 0 0

The answer is: 0001110101

[WBUT2007, 2009]

2. Multiply -5 by -3 using Booth's Algorithm.

Answer:

The binary representation of 5 - 0101—

The binary representation of -5 = 2's complement of 0101= 101 1 (multiplicand)

The binary representation of 3 = 1 101(nnulti lier)

The binar resentation of-3 2's lement of 001 1

$Q_n Q_{n+1}$	$BR = 10111 \quad BR + I = 01001$	AC	Q_{n+1}	SC	
	Initial	0 0 0 0	I 1 0 1	0	I O O
1 0	Subtract BR	0 1 0 1			
	ashr AC,QR (Includin ° l)	0 0 1 0	I I 1 0		O i 1
0 1	Add BR	1 0 1 1			
	ashr ,AC,QR (Includin ° Q_n)	1 1 0	I 1 I	o	0 1 0
1 0	Subtract BR	0 1 0 1			

POPULAR PUBLICATIONS

		O O i l			
	ashr AC,QR (Including (A)n l)				O O I
	ashr AC,QR (Includin °(On	0 0 0 0	I I I I	I	0 0 0

The answer is: 000011 II

3. Represent the decimal value - 7.5 in IEEE - 754 single precision floating-point
PNBUT 2008, 2011] Answer:

The IEEE-754 format is as follows.

Bit 31: sign bit

Bit 30-23: 8-bit biased exponent represented in excess-1 27

Bit 22-0: 23-bit normalized mantissa (magnitude), ~~to~~ i of the most significant I bit in
the real number (integer part aligned to lie just on the right
fraction).

Scti*1 Nddcr

real adder adds one bit at a time.
 serial adder is comparatively slower than
 adder.
 hardware needed for serial adder is less.
 circuit in serial adder is less complex.

Parallel Adder

1. Parallel Adder adds the whole thing at once.
2. Parallel adder is much faster than serial adder.
3. Hardware needed for parallel adder is more.
4. Circuit needed for parallel adder is more complex.

Now, $-7.5 = -111.1 = -1.111 \times 2^3$ to the power +2
 So, bit 31 = 1

$30-23 = 127 + 2 = 129D = 10000001B$
 $22-0 = 1110...0$ (3 1's followed by 20 0's)
 Hence the representation is C0F00000H.

Answer:Both are used for adding binary numbers.

1. a) compare parallel adder with serial adder. 2008, 2011)

parallel

3

4

- b) Explain and draw the 4-bit binary decrementer circuit. [WBUT 2008]
 OR,

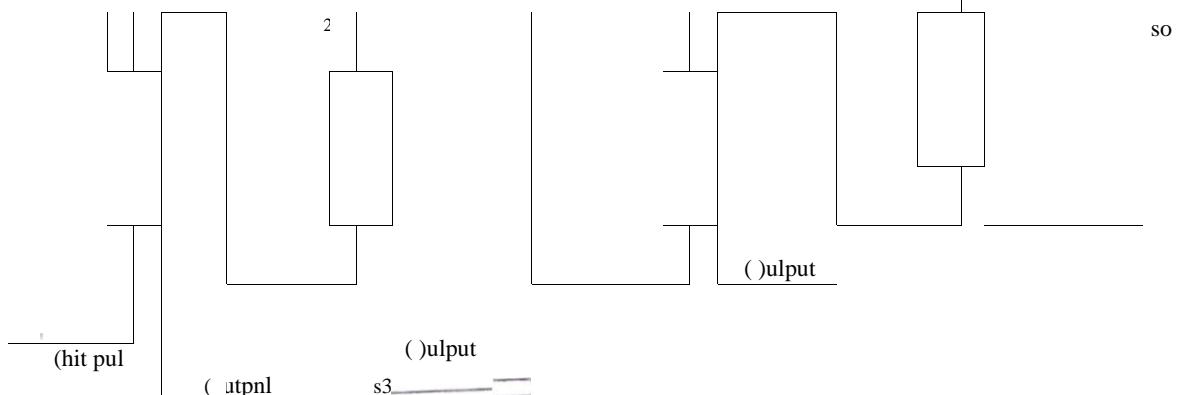
Explain and draw a binary decrement unit. PWBUT 20111 OR,

Design a 4-bit combinational circuit decrementer using four full adders.

[WBUT 2016, 2018]

Ansucr:

POPULAR PUBLICATIONS



5, a) A 32-bit floating-point binary number has a bit plus a sign for the exponent. Negative Numbers in the mantissa and exponent are in signed-magnitude representation. What are the longest and smallest positive qualities that can be represented excluding zero? Explain with example.

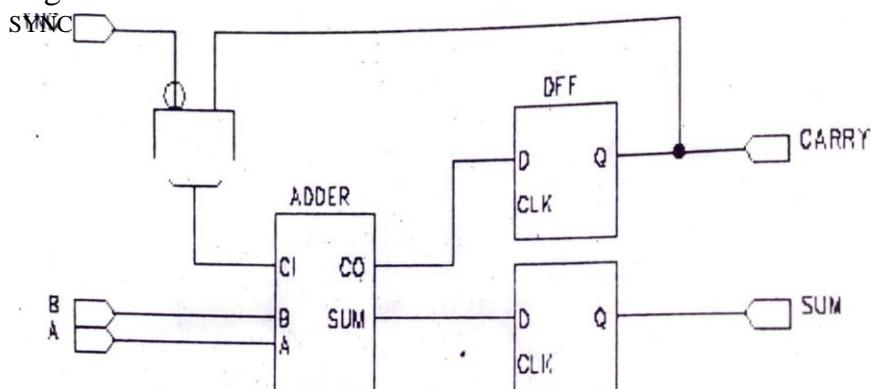
b) Explain with diagrams, Serial & Parallel Adders.

c) ADD A+B, where A = 63.11236589 & B = 0.002365991 x 1029. (WBUT 20091

a) In the 92-bit floating-point number negative numbers are represented in the leading bit is complement form, where mantissa and 8 bits are reserved for signed exponent, then the (longest) --- $11 \dots 11 * (2^{127})$ and the minimum (smallest) positive is $00 \dots 00 * (2^{-128})$.

A serial adder single is a binary adder that adds the two numbers bit-pair wise. Each bit-pairs are added in each clock pulse. The carry of each pair is propagated to the next pair.

Circuit diagram:



Serial Binary addition is done by, in simplest terms, a flip-flop and a full adder as stated above. However, there are slight nuances to the addition that can be confusing. When a serial adder performs its addition, it is

partially dependent on the clock cycle as a flip-flop is asynchronous and the full adder is not. Thus, when a timing diagram is shown, the sum output will change as the inputs are changed relative to the previous clock cycle which was used to determine the carry in bit.

Example of operation

Denary $5+9=14$

$X=5, Y=9, \text{Sum}=14$

Binary $0101+1001=1110$

Addition of each step

Inputs			Outputs	
Cin	X	Y	Sum	Cout
0	1	1	0	1
1	0	0	1	0
0	1	0	1	0
0	0	1	1	0

*addition starts from lowest

*addition starts from lowest

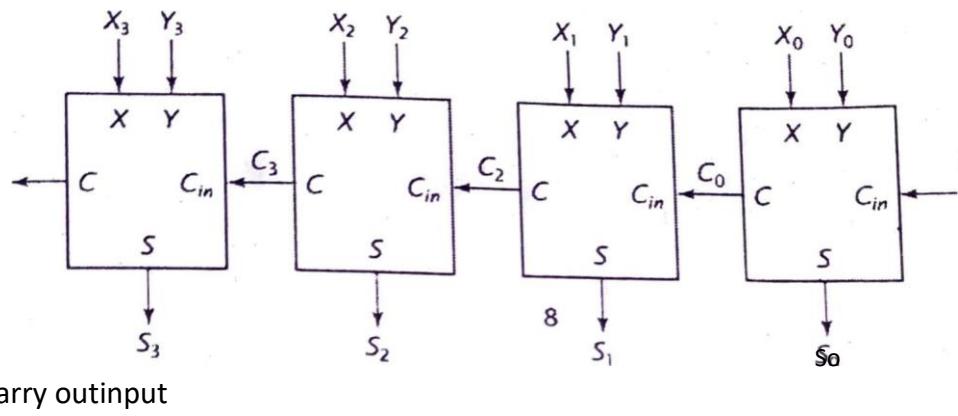
Result=1110 or 14

parallel Adder:

\ parallel adder is a binary adder that generates the arithmetic of two binary

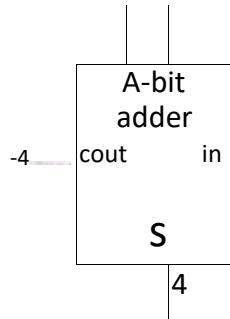
numbers of any lengths using multiple cascaded full-adder circuits. Here the output carry from one full-adder is connected to the input carry of the next high order full-adder. Here the

done in pulse. entire addition is a single



(Co)

(a)



(b)

Parallel adder is the digital circuit that generates the arithmetic sum of two binary numbers of any lengths. Multiple cascaded full-adder circuits constitute a parallel adder. The output carry from one full-adder is connected to the input carry of the next high order full-adder and so on. So an n-bit parallel (binary) adder requires n full-adders. The augend bits of X (say. n data bits) come from one register (say. Register R1) while the addend bits of B (n data bits) come from another register (say. register R2). Subscript numbers from right to left designates the augend bits and the addend bits. Subscript 0

denotes the lower-order bit while the subscript n denotes the higher-order bit. The carries are connected in a chain through the full-adders. Co is the input carry to the multiple Cascaded binary adders and Cout is the output carry. The required sum bits are generated by the S outputs of the full-adders. The sum can be stored either to a third register (say. Register R 3) or may be in any one of the source registers (replacing its previous content).

Example of operation

Here suppose that ' 1 0' i.e. 1010 is to be added to '5' i.e. 0101. So the augend bits or Ao through A; will be 0, 1, 0 and 1 (i.e. Ao will input A1 will input 1, A2 will input 0 and AN will input 1). Similarly, for B also (the addend bits). Adding the two numbers will

give the output as 1111 (i.e. 15). So the output line so will output 1 . S1 will output 1 , S will output 1 and will output 1 . In the exanil)e there are no carry-bits.

c) $A = 63.1\ 1236589 \times 10^{15} - 63\ 1\ 12365890000000\ B$
 $= 0^0 02365991\ 1\ 0-29 = 2.365991\ E-32$

Result of addition: 12622473 1 780000000

6. Explain non-restoring division algorithm and explain the hardware [WBUT diagram20101 Perform the Restoring division operation with 19 divided by 8. nswer: st

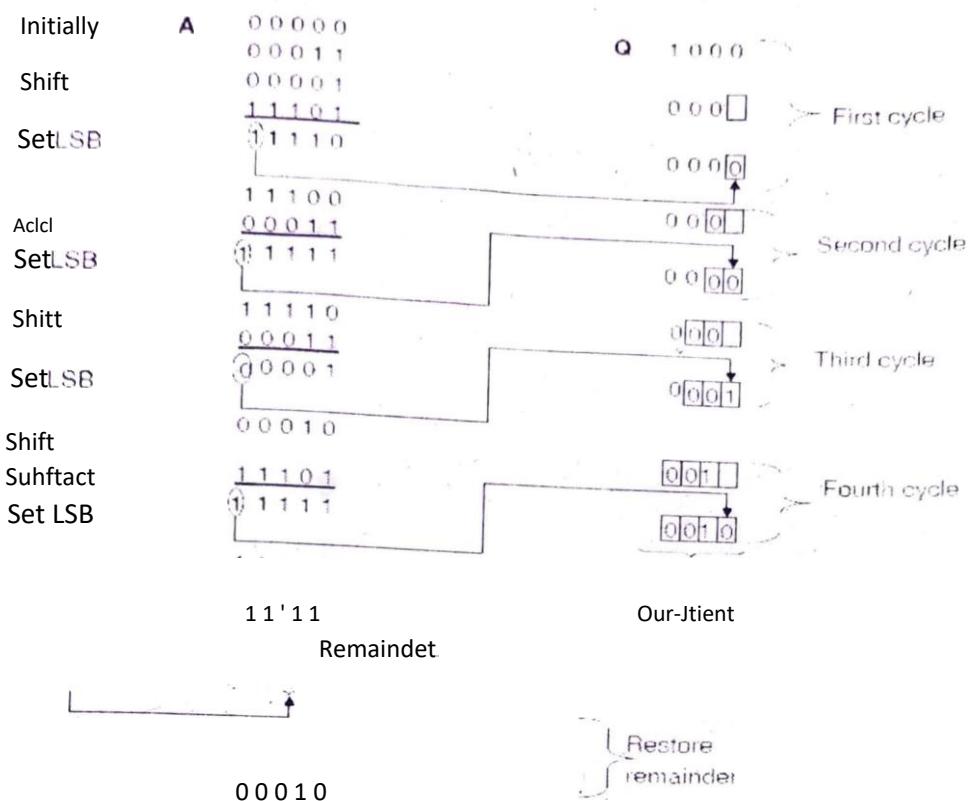
Non-restoring division algorithm:

Non-restoring division uses the digit set {-1 , 1} for the quotient digits instead of {0, 1}

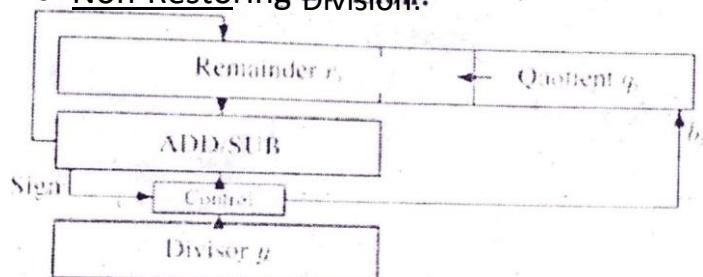
Non-restoring division technique, if the difference is negative then the divisor is not

added directly to the partial remainder. It is added only after shifting the negative difference to the left i.e. suppose while performing. division by restoring division technique, subtraction of the divisor content in D from thatof A leads to a negative result (i.e. an unsuccessful subtraction). Still the value of A is to be restored. This is however time constillling and can be seen as an unnecessary overhead. This drai\ back of the restoring division technique can b avoided in the non-restoring division technique. Algorithm:

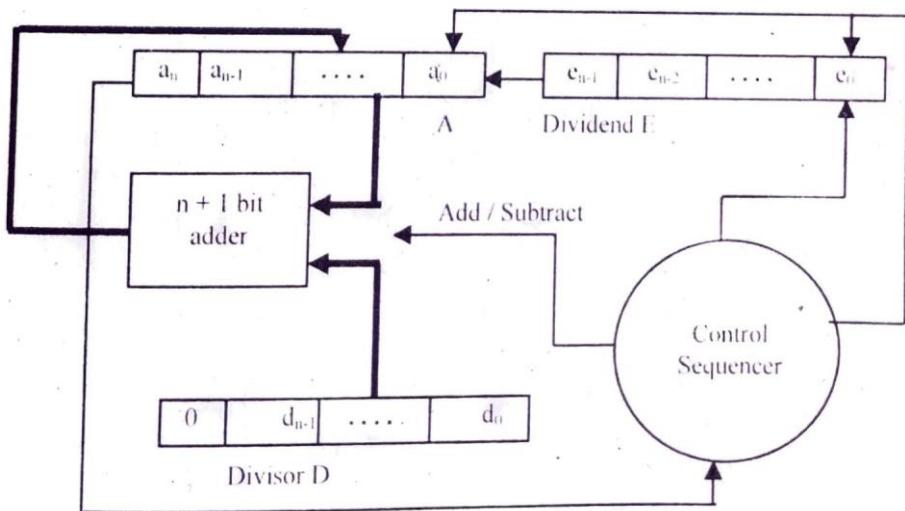
- STEP 1 : DO n TIMES.
 - IF THE SIGN OF A O, SHIFT A AND Q LEFT ONE BINARY POSITION AND SUBTRACT M FROM A;
 - OTHERWISE, SHIFT A AND Q LEFT AND ADD M TO A. IF THE SIGN OF A IS O, SET Q0 TO 1 ; OTHERWISE SET Q0 TO O.
- STEP 2: IF THE SIGN OF A I, ADD M TO A
 - The negative result is restored by adding, i.e.. R_i
 $(R_i - M) + M (1)$
 - and is followed by a shift left one (i.e., R_{i+1}) and is followed by a shift left one (i.e., multiplication by 2) and subtract:
 $2 R_i - M (2)$
 - The two operations (1) and (2)
 $2 [(R_i - M) + M] - M = 2 R_i - M$ merged into a single one: R_{i+1}



Hardwire Diagram of Non-Restoring Division:



2nd Part:



Now, as the rule of the division Fio: Circuits goes, 5-bit for restoring registers division are used technique the divisor M and the remainder A. Also, the high-order bit of M and all the bits in A are initially cleared (or made 0).

CO-31

po LAR PUBLICA ION 01000 (5-bit register) and that in in 1 1000 considered while subtracting. initial values in Q 1001 1. of M IS complement = 0000 A (5-1)it register).

-0 cleaved if at is -ve and set if +ve. sign-bit ol' A;

q,
a₁ →

The ision steps ave as e ^x plained below:

1 st Cycle:

Explanation

a₄ a₃ a₂ a₁ a₀

0 0 0 0

0	1	0	0	1	1
0	0	1	0	1	0

shifted Left as a pair (A and Q)
M is subtracted from A A is restored (Add M to A)

Q

q₄ q₃ q₂ q₁ q₀

Initial Values

1 0 0 1 0 0 1 0

1 0 0 1 1 0

0 0 0 0 1 0 1 0

A) I

0 0 0 1 0 0 1 1

0 0

I 1 0 1 0 0 1 1 0 0

0 0 0 1 0 0 1 1 0 0

0 0 0 1 0 0 1 1

0 0

cycle:Shift

↘ ↗ ↘
 1 1 0 0 0
) 1 1 0 0 0
) 1 1 0 0 0
 1 0 0 0

Left as a Pair (A and Q)

4th

Cycle:

0 1 0 0

1

1 0 0 0 0

Since result is —ve,
clear
is

2nd cycle.
and Q)

Shifted Left as a

M is subtracted from A

Since result is —ve, A is restored (Add M to A) Y^d
CIO is cleared.

0 0 0 0 1 0 0 0 0

Since result is +ve, q0 is set.

1
1 0 0 0 1

0 0 0 0

1

5th Cycle:

0 0 0 1

I 0 0 0 1 0

I 1 0 1 0 0 0 1 0

I

0 0 0 1 0 0 0 1 0

I

0 0 0 1 0 0 0 1 0

I

0 0 1 0 0 M is subtracted from A

I O O Since result is —ve, A is restored (Add M to A)

0 0 1 0 0 q0 is cleared

0 0 1 0 0 I

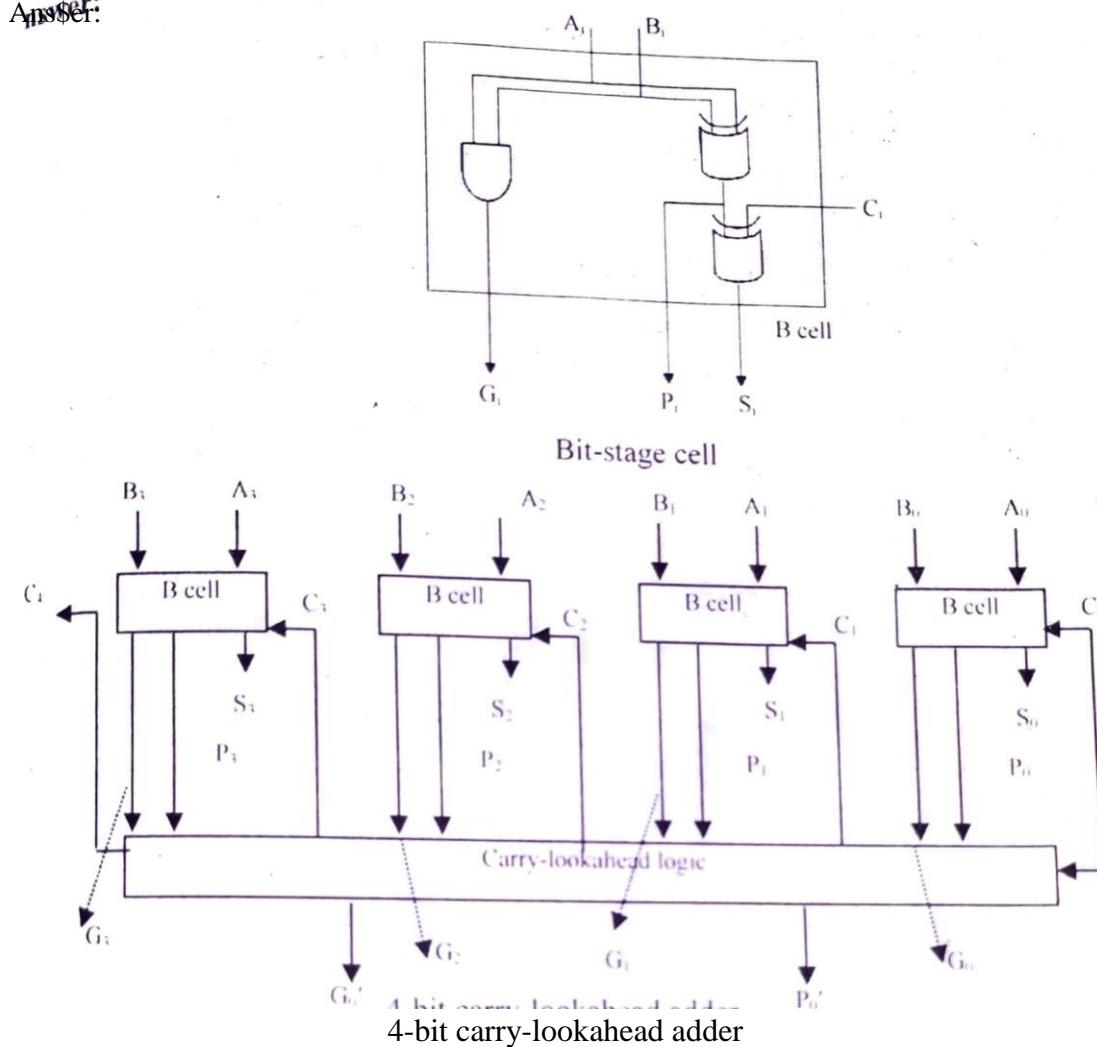
Therefore, the final value in A = 0001 1 is the remainder (the extra sign-bit is considered) and that in Q = 00010 is the quotient.

7. Draw the logic diagram and discuss the advantages of a carry look ahead a over conventional parallel adder. (WBUT

OR,

With a suitable block diagram discuss the construction and working principle of an 8-bit carry-look-ahead adder.

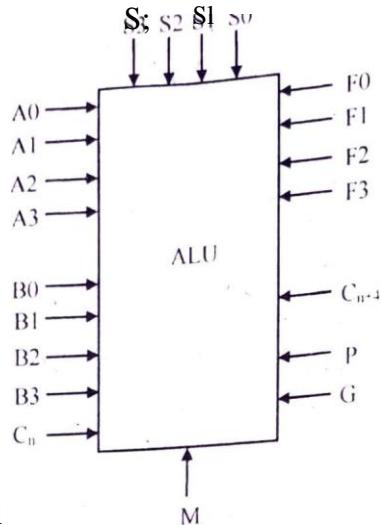
Answer:



In case of parallel adders, cascading of n-full adders are required to add two n-bit numbers together. The carry signals thus 'ripple' through the adders from right to left and all the related logic gates take a non-zero time delay (propagation delay) to respond to a change in the input. This is because in case of conventional parallel adders, the result of an addition of bits depends on the carry generated by the addition of the previous two bits. So, the sum of the most significant bit is only available after the carry signal has rippled through the adder from the least significant stage to the most significant stage. But, in case of a carry-look ahead adder, the carry does not have to depend explicitly on the preceding one and can be expressed as functions of relevant addend and augend bits. So, the overall delay is much lesser than the conventional parallel adder.

8. Design and describe a 4-bit ALU and its operations. [WBUT 20141 OR,

Design a 4-bit ALU capable of performing 14 different micro operations including logical, arithmetic and shifting operations. (WBUT 20181



Functionally, the operation of typical

ALU is represented SO
as shown in diagrtun below,

Fig: Functional representation of Arithmetic Logic Unit

Functional Description of 4-bit Arithmetic Logic Unit

- Controlled by the four function select inputs (S0 to S3) and the noode control input (M), ALU can perforll all the 16 possible logic operations or 16 difTerent arithlnetic operations on active HIGH or active LOW operands.
- When the mode control input (M) is HIGH, all internal carries are inhibited and the device performs logic operations on the individual bits. When M is LOW, the carries are enabled and the ALU performs arithlnetic operations on the two 4-bit words. The ALU incorporates full internal carry look-ahead and provides for either ripple carry between devices using the Cn+4 output, or for carry look-ahead between packages using the carry propagation (P) and carry generate (G) signals. P and G are not affected by carry in.
- For high-speed operation the device is used in conjunction with the ALU carry 100kahead circuit. One carry look-ahead package is required for each group of four ALU devices. Carry look-ahead can be provided at various levels and offers high-speed capability over extrenlely long word lengths. The comparator output (A=B) Of the device goes HIGH when all tour function outputs (F0 to F 3) are HIGH and can be used to indicate logic equivalence over 4 bits vhen the unit is in the subtract A=B is an open collector output and can be with other A=B outputs to give a comparison for more than 4 bits. The open drain output A=B should be used with an external pull-up resistor in order to establish a logic HIGH level. The A-B signal can also be used with the Cn+4 signal to indicate A > B and A < B.

POPULAR PUBLICATIONS

The function table lists the arithmetic operations that are performed without a carry in. An inconning carry adds a one to each operation. Thus, select code LHHL generates A minus B minus I (2s complement notation) without a carry in and generates A minus B when a carry is applied

subtraction is actually performed by complementary

- complement), and a no carry carry out is means generated borrow; when thus, there is a carry underflow.is generated when there is no indicated, the ALU can be used with either active LOW inputs producing active
- out Dilts or with active I in active HIGI I out uts.

Mode select in)lts jroducing			Active high in uts and out uts	
S;	S,	so	Logic (M=H)	Arithmetic(2) (M=L; C _n =I)
L	L	L		
L	L			
L			AB	
L	1-1	1-1	10 logical 0	minus I
L		L	AB	A plus A B
L	1-1			A minus B minus I
H	1-1		AB	AB minus I
L	L	L		A plus AB
L	L	H		A plus B
L	H	L	B	(A + B) plus AB
L		1-1	AB	AB minus I
1-1	L	L	logical I	A plus A ⁽¹⁾
H	L	H		(A + B) plus A
				(A + B) plus A
	1-1			A minus I

Notes to the function tables:

1. Each bit is shifted to the next more significant position.
2. Arithmetic operations expressed in 2s complement notation.

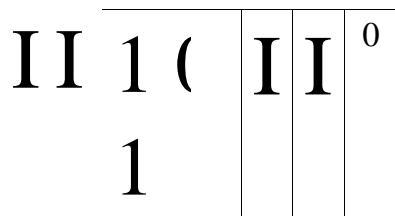
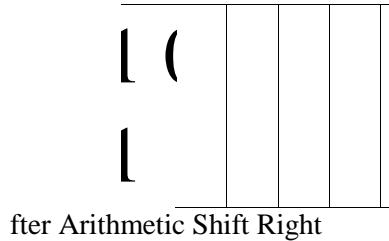
H=HIGH voltage level L=

LOW voltage level

POPULAR PUBLICATIONS

9. Suppose register A holds the 8-bit number 11011101. Determine the sequence of binary values in A after an arithmetic shift-right, followed by a circular shift-right and followed by a logical shift-left.
) Describe Booth's multiplication method and use this to multiply decimal numbers — 23 and 9.

[WBUT 2015) nsver:



After Circular Shift Right								
0	1	1	1	0	1	1	1	1

After Logical Shift Left								
1	1	1	0	1	1	1	1	0

b)

$Q_n Q_{n+1}$	$BR = 101001 \quad \overline{BR} + 1 = 010111$	AC	QR	Q_{n+1}	SC
	Initial	000000	001001	0	110
10	Subtract BR	010111			
		010111	001001	0	
	ashr(sc = sc-1)	001011	100100	1	101
01	Add BR	101001			
	ashr(sc = sc-1)	110100	100100	1	
		111010	010010	0	100
00	ashr(sc = sc-1)	111101	001001	0	011
10	Subtract BR	010111			
		010100			
	ashr(sc = sc-1)	001010	000100	1	010
01	Add BR	101001			
		110011	000100	1	001
	ashr(sc = sc-1)	111001	100010	0	
00	ashr(sc = sc-1)	111100	110001	0	000

The answer is: 111100110001

10. Divide (-15) by (-3) using Restoring & Non-restoring Division algorithm.

Answer: [WBUT 2017] There are no simple algorithms for directly performing division on signed operands that are comparable to the algorithms for signed multiplication. In division the operands can be preprocessed to transform them into positive values. After using • restoring or non restoring division method the results are transformed to the correct signed values as necessary. Here if we divide (-15) with (-3) we will get '0' as remainder and 5 quotient.

$$'0' = 00000$$

$$5 = 00101$$

$$Q = 15$$

$$A = 00000$$

$$Q = 01111$$

$$M = 00011$$

$$M' = 11100$$

POPULAR PUBLICATIONS

$M' + l = 1 \ 1 \ 101$

COMPUTER ORGANIZATION

rltethod:

gctorlng

0001 1

$$\begin{array}{r}
 00000 \\
 00000 \\
 \hline
 1 \ 1 \ 101 \\
 | \quad | \\
 | \quad 1 \\
 | \quad 101 \\
 \hline
 0001 \ 1
 \end{array}$$

Shift AQ
 $A = A - M$

$= 0$

$$\begin{array}{r}
 00000 \\
 00001 \\
 \hline
 1 \ 1 \ 101 \\
 | \quad | \\
 | \quad 1 \ 1 \\
 | \quad 10 \\
 \hline
 0001 \ 1
 \end{array}$$

110

Restore A

Shift AQ
 $A = A - M$

00001

I1IOO

0001 1
1 1 101
@oooo

I1IOO

Restore A

Shift AQ
 $A = A - M$
 $Q[0] = 1$

00001
1 1 101
1110
0001 1
00001

1001

Shift AQ
 $A = A - M$

0001 1
1 1 101
@oooo

1001 0

Restore A

00001
1 1 101
@oooo

10010

Shift AQ
 $A = A - M$

=Remainder

$Q = \text{Quotient } 111 + 1 = 1$

1101

DECODE AND APPLICATIONS

Non-restoring
method:

			Shift Aq $A = A - M$
0001 1	00000 00000 <u>1 1 101</u> 1 101 ----- 1 101 1 <u>0001 1</u> 1 1 10 ----- 1 1 101 0001 1 ----- @0000	1 1 10 ----- 1 100 1 100 1 1001 1001 1 1 10 ----- 0010 00101	Shift AQ $A = A + M$ <u>Q[0] = 0</u>
		II 100 1 100 1 1001 1001 1 1 10	Shift AQ $A = A + M$ (0101 = 1)
		10010	Shift AQ $A = A - M$ (0101 = 0)
	1 1 101 <u>0001 1</u> @0000	0010 00101	Shift AQ (0101 = 1)

A = Remainder

Q = Quotient

11. Write short notes on the following:

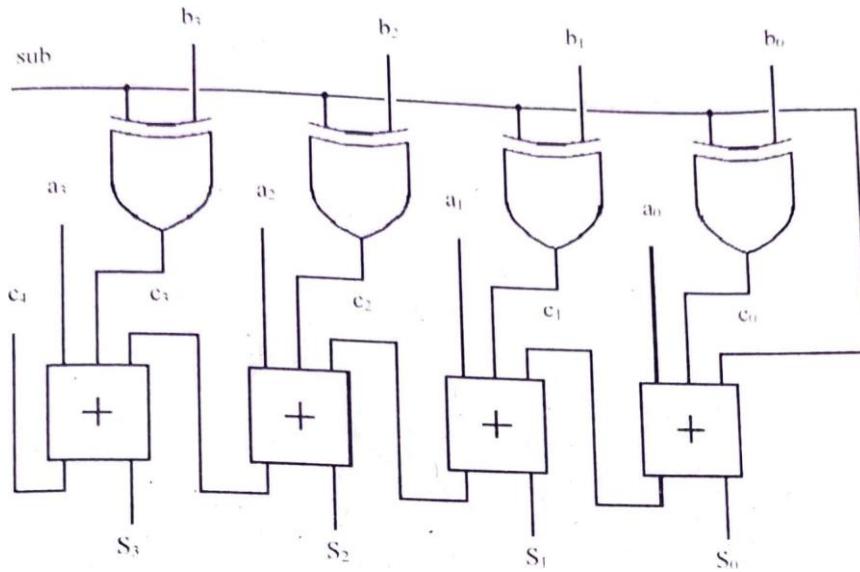
- a) Adder-subtractor circuit CWBUT 2007, 20091
- b) Non-restoring division method [WBUT 20081]
- c) Booth's algorithm [WBUT 20081]
- d) Carry Look-ahead Adder [WBUT 20181]
- e) Design of 4-bit ALU [WBUT 20181]
- f) Overflow in Fixed-point Representation CWBUT2

POPULAR PUBLICATIONS

Answer:

- a) Adder-subtractor
circuit:

The addition and subtraction operations can be combined into one common circuit by including an exclusive-OR gate with each full adder.



A 4_bit adder-subtractor circuit is shown in the above figure. -1 he mode input sub controls the operation. When sub=0 the circuit is an adder and when sub=1 the circuit becomes an subtractor. Each exclusive-OR gate receives input sub and one of the inputs of b. When sub=0, we have $b \oplus 0 = b$. The full adder receives the value of b, the input carry is 0, and the circuit performs a plus b. When sub 1, we have $b \oplus b' = 1$ and $C_0 = 1$. The b inputs are all complemented and a 1 is added through the input carry. The circuit performs the operation a plus the 2's complement of B. For unsigned numbers, this gives $(a-b)$ if $a \geq b$ or the 2's complement of $(b-a)$ if $a < b$. For signed numbers, the result is $(a-b)$ provided that there is no overflow.

b) Non-restoring division:

Refer to Question No. 6 of Long Answer Type Questions.

c) Booth's algorithm:

Refer to Question No. 1 of Long Answer Type Questions.

d) Carry Look-ahead Adder:

Refer to Question No. 7 of Long Answer Type Questions.

e) Design of 4-bit ALU:

Refer to Question No. 8 of Long Answer Type Questions.

f) Overflow in Fixed-point Representation:

POPULAR PUBLICATIONS

Overflow handling is an important consideration when implementing signal processing algorithms. Typical digital signal processing CPUs include hardware support for handling Overflow. Some RISC processors may include in these modes.

OVERFLOW MODES

Overflow with 2's complement integers OCCURS when the result Of an addition subtraction is larger than the largest integer that can be represented, or smaller than the smallest integer. In fixed point representation, the largest or smallest value depends on the format Of the number. Suppose, Assume, a 32 bit register, a CPU saturation arithmetic would set the result to -1 or +1 on an overflow; corresponding to the integer values $0 \times 8000000000000000$.

INSTRUCTION SET

Chapter at a Glance

InstrUCtion set: Instruction set is the set Of Instructions that a nnachine is able to execute. Each particular Inachine has its own set of instructions i.e. an instruction set. which consists of all the instructions used in that particular connputer, varies from computer to conoputer

depending on the specific organization and architecture of the coinputer.

Instruction format: Instruction forinat deals with the looks of a basic instruction. Each instruction has three parts. The 'opcode' part, the 'addressing Inode' part and the operands or the 'address' (i.e. operand address) part. The operation field is called the 'opcode' or the . operation code'. The operand address fields contain either the operands themselves or the addresses of storage locations of the data or arguments (i.e. addresses of operands) in main memory or in the processor depending on the various addressing modes (I) as specified in a panicular instruction.

- Three, Two, one and zero address instructions: (i) Three-address instructions:

In these type of instructions, all operand addresses are explicitly defined. Here the instruction format has three different address fields specifying a memory or a processor register operand.

Advantages:

- (i) It results in short programs when evaluating arithmetic expressions.
- (ii) Less execution tilne.
- (ii) Two-address instructions: Here the instruction format has two different address fields, each specifying either a Inenwry or a processor register operand.

Advantages: Less execution time coinpare to one-address instructions.

(iii) One-address instructions:

Such instruction format has a single explicit address field and uses an implied accumulator (AC) register for all data manipulation.

Advantages:

- (i) Much less nunnber of bits is required to specify the single operand address.
- (ii) Less complicated decoding and processing circuit is needed.

(iv) Zero-address Instructions:

Such instructions do not contain any explicit addresses (ex'cept for PUSH and POP instructions). As the operands are stored in a pushdown stack (the operands required trust be there in the top positions in the stack), hence no addresses are required.

Advantages: Do not contain any explicit addresses. So instructions are sitnple.

POPULAR PUBLICATIONS

Different types of addressing modes:

- (i) Implied Mode
- (ii) Immediate Addressing Mode

- (iii) Register Mode or Register Direct Mode
 - (iv) Register Indirect Mode
 - (v) Auto-increment Mode
 - (vi) Auto-decrement Mode
 - (vii) Direct Address Mode (Oii)
 - Indirect Address Mode
 - (i.x) Relative Address Mode
 - (a) pc (i.e. Counter) Relative Addressing Mode
 - (b) Indexed Addressing Mode or Index Register Relative Addressing Mode
 - (c) Base Register Addressing Mode
 - (x) Stack Addressing Mode

Multiple Choice Questions

1. Instruction cycle is [UT2006, 2007, 2011, 2012, 2015, 2018] fetch-decode-execution b) decode-fetch-execution

- c) fetch-execution-decode
- d) none of these

Answer: (a)

2. A digital computer has a memory unit with 24 bits per word. The instruction set consists of 150 different operations. All instructions have an operation code part (opcode) an address part (allowing for only one address). Each instruction is stored in one word of memory. Bits are needed for opcode. [WBUT 2007]

Answer: (c)

3. Micro c) a) Cache Main instructions memory memory are kept in d) Control None [WBUT of memory these 2007, 2011, 2015, 2018]

Answer: (b)

4. Which of the following addressing modes is used in the instruction PUSH B?

- a) Immediate
- c) Direct
- d) Register Register Indirect [WBUT 2008, 2011]

Answer: (b)

5. Which of the following addressing modes is used in instruction RAL

POPULAR PUBLICATIONS

a) immediate implied 2012]

Answer: (b)

c) direct

[WBUT

d) register

6. Which of the following address modes is used in the instruction ,pop 3?

a) immediate

Answer: (d) b) register c) direct [WBUT 2013] d) register indirect

co-42

1. A computer uses words of
a) may or may not be one
b) always be fetcher

SblyZtee312e-nb}thThe instruction

[WBUT 2014]

- I,) must always be fetched in one cycle with 2 bytes in the cycle
- c) must always be fetched in two cycles with one byte in each cycle
- d) must be Of 2 bytes length

\noor: (c)

The CPI value for RISC processor is

[WBUT 2014]

- \$
- d) none of these

9. In a) the stack processor, the address of the next instruction b) index register to be executed [WBUT is stored 2015)in
pointer register

- c) base register
- d) program counter register

\nswer: (c)

10 A stack-organised computer uses instruction of[WBUT 2016]

- a) Indirect addressing
- b) Two addressing
- c)Zero addressing d) Index addressing \wer: (c)

II, When performing a looping operation, the instruction gets stored in the

- a) Registers
- b) Cache
- c) System heap
- d) System stack

Answer: (b)

12. In case of Zero-address instruction method the operands are stored in

[WBUT 20171

- a) Registers
- b) Accumulators
- c) Stack
- d) Cache

Answer: (c)

13. The addressing mode(s), which uses the PC instead of a general purpose register is

[WBUT 20171

- a) Indexed with offset
 - b) Relative
 - c) Direct
 - d) Both (a) and (b)

Answer: (b)

14. How many memory locations can be addressed by a 32-bit computer?

- a) 64 KB b) 32 KB [WBUT 20181]

Answer: (a)

15' The addressing mode of an instruction is resolved by

- a) ALU [WBUT 2018] b) DMA controller d) program

nswer: (b)

POPULAR PUBLICATIONS

two- address IONS
address **Short Answer Type Questions**
 two-add

address, one-

1. Explain the difference between three-address, cwBUT 2007, 2011, 20131

OR,

two-address, one
[WBUT 2015, 2012]

instructions & zero-address instruction with suitable examples.

$X = (A + B) * C$ Evaluate the following arithmetic expression into three-address, two-address, address, zero-address instruction format. 13) * c

Three Address Instructions:

Answer:

In three-address instructions all operand addresses are explicitly defined and the instruction format has three different address fields specifying a memory or a processor register operand. For example, evaluating $X = (A+B)*C$ in a three-address machine will result to:

ADD T, A, B

$$\Rightarrow T \leftarrow A + B$$

MULTIPLY X, c, T

$$\Rightarrow X \leftarrow C * T$$

Use:

Cyber 170 is a commercial computer using three-address instructions.

Two Address Instructions:

In two-address instructions, the instruction format has two different address fields, each specifying either a memory or a processor register operand. Evaluating $X = (A+B)*C$ in a two-address machine will result to:

MOVE

T

ADD

T

$$T \leftarrow T + B$$

MULTIPLY C, T

$$X \leftarrow C * T$$

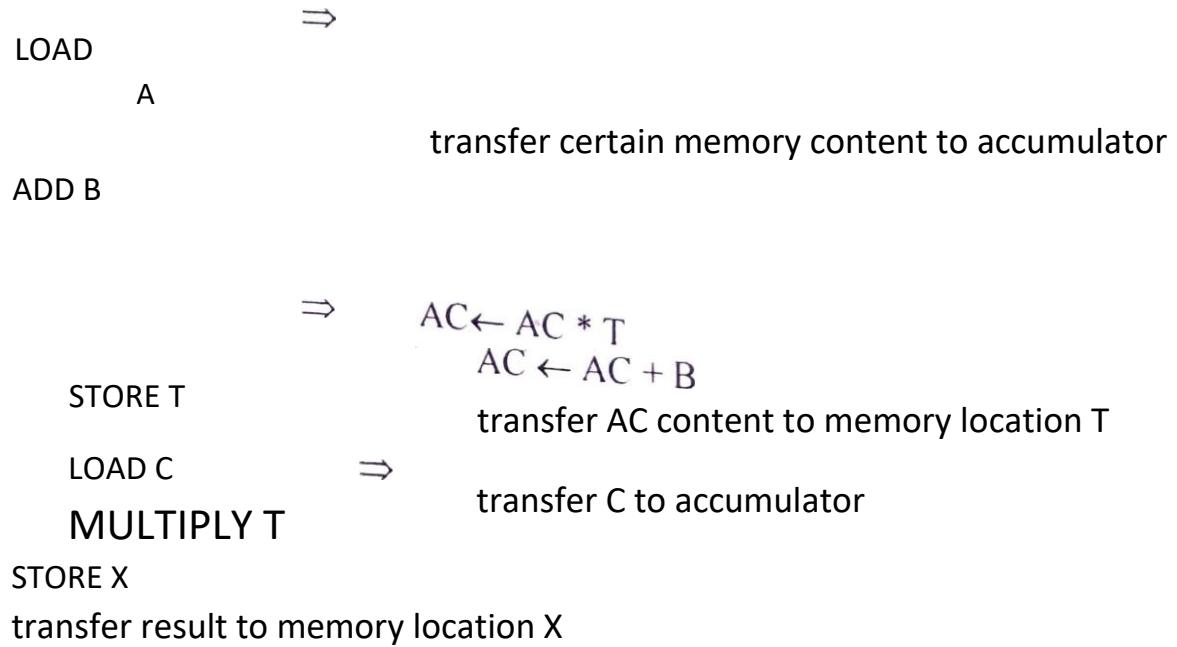
Use:

Two-address instructions are used in all commercial computers.

One Address Instruction:

POPULAR PUBLICATIONS

For one-address instructions, the instruction format has a single explicit address field and uses an implied accumulator (AC) register for all data manipulation. Evaluating $X = (A+B)*C$ in a two-address machine will result to:



U.S.

OR

Intel 8085 machines.

Address Instruction:

Micro-address instructions do not contain any explicit addresses (except for PUSH and pop instructions). As the top operands positions are stored in stack), a pushdown hence stack no addresses (the operands are required).

Evaluating $PUSH X A = (A+B)*C$ in a two-address machine will result to:

```

PUSH B
ADD
PUSH C MPY
pop X

```

Use:

In all stack-type computers.

2. Given an example and explain Base-index Addressing. [WBUT 2007, 2011, 2012] Answer:

In Base-Index addressing mode, the effective address is the sum of the contents of the specific base register and that of the specific index register. For example, such an addressing mode could be useful in accessing elements of an array. In this case, the base register would hold the starting location of the specified array and the index register would contain the location of the offset.

3. Compare and contrast RISC and CISC architecture. [WBUT 2009, 2011, 2017]

Answer:

RISC

CISC

i) Multiple register sets, often consisting of more than 256 registers. i) Single register set, typically 6 to 16 registers total.

- ii) Three register operands allowed per instruction
- ii) One or two register operands allowed per (g. -+ add RI, R2, R;) instruction (e.g. —Y add RI, R2)
- iii) Parameter passing through efficient on chip
- iii) Parameter passing through inefficient offregister windows.
- chip Inemory.
- iv) Single cycle instructions (except for load and store)
- v) Hardwired control.
- v) Micro-programmed control.
- vi) Highly pipelined.
- vi) Less pipelined.
- vii) Simple instructions that are few in number.
- vii) Many complex instructions.
- viii) Fixed length instructions.
- viii) Variable length instructions.
- ix) Complexity in compiler.
- ix) Complexity in microcode.
- x) Only load and store instructions can access memory.
- x) Many instructions can access Inemory.
- xi) Few addressing Inodes.
- xi) Many addressing Inodes.

4. What are the advantages of relative addressing mode over direct [WBUT address 2011, mode?20171

In such cases the content of the CPU register is added to the address part of the instruction to obtain the actual address of the operand (i.e. the effective address). The address part of the instruction, which is usually a signed number (either Positive or negative) on addition to the CPU register content, gives the effective address whose position in memory is relative to the address of the next instruction. Uses:

Relative addressing mode is often used with branch-type instructions.

This also results in shorter address field in the instruction format as the relative address can be specified with a smaller number of bits compared to the number of bits required to designate the entire memory address.

Whereas, in Direct addressing mode, the effective address of the operand is equal to the address part of the instruction i.e. the address part of the instruction indicates the memory location containing the operand.

5. Compare RISC and CISC architectures in brief. Explain PC-relative addressing mode with example. [WBUT 2013] Answer:

Refer to Question No. 3 & 4 of Short Answer Type Questions.

6. A computer uses a memory unit with 256 K words of 32 bits each. A binary instruction code is stored in one word memory. The instruction has four parts; an indirect bit, an operation code, a register code part to specify one of 64 registers and an address part.

- i) How many bits are there in the operation code, the register code part and the address part?
- ii) Draw the instruction word format and indicate the number of bits in each part.
- iii) How many bits are there in the data and address inputs of the memory?

[WBUT 2013, 2016]

Answer:

POPULAR PUBLICATIONS

i) Memory has 256K words 2¹⁸ words so need 18bit address bus, word size is of 32 bits • So one instruction is also of 32 bits size. There are 64 registers, so 6 bits need to represent the register part. Hence for opcode part we need $32 - (1+18+6) = 7$ bits.

ii)

indirect bit (1bit)	operation code (7 bits)	register code (6bits)	address part (18bits)
------------------------	----------------------------	--------------------------	--------------------------

iii) There are 32bits data input and 1 8bits address input in memory.

ORGANISATION

1. Explain
case?
Answer:
Refer to C

indirect address mode. How is the effective address calculated in this [WBUT 20161

Question No. I of Long /insh'er Type Questions.

addressing explain mode.with example: Register Direct, Register Indirect and Base [WBUT register20181

Answe
nswer to Question No. I of Long Answer Type Questions.

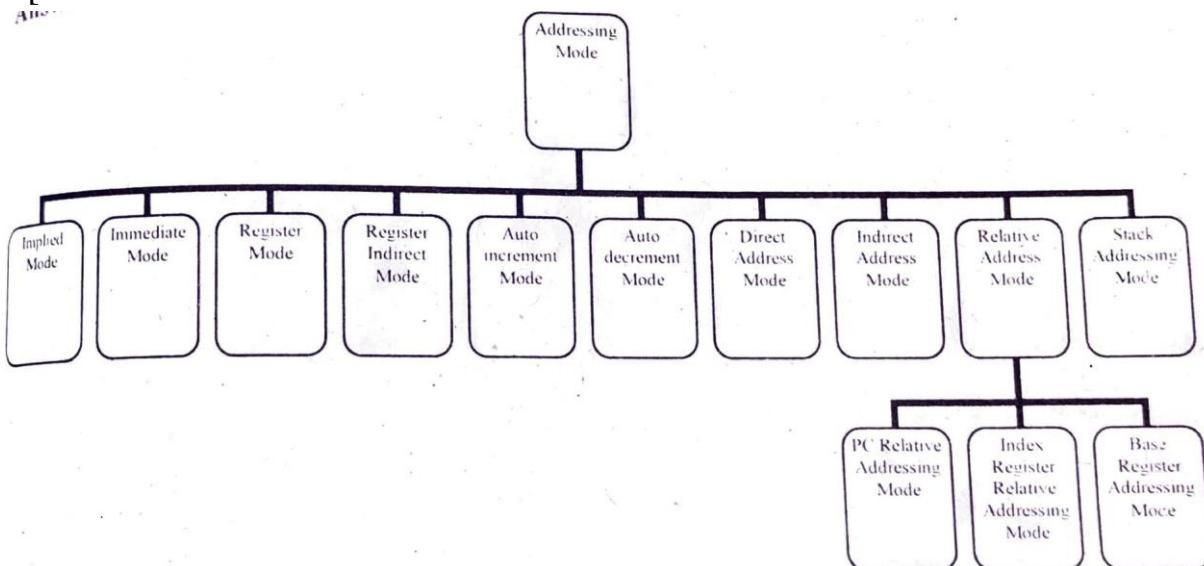
Lon Answer Type uestions

f Describe briefly different addressing modes with suitable examples.

[WBUT 2003,

20051 OR, write down the type of addressing modes. Explain with examples.

[WBUT 20131 Alls" er:



(i) Implied Mode:

In this the operands are specified implicitly' in the definition of the instruction i.e. the operands are innplied instruction definition. Operands need not be specified explicitly.

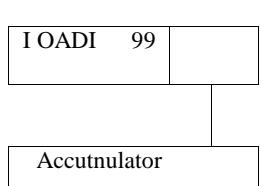
Example:

Complelment Accunnulator: The above instruction is an implied-nnode instruction because the operand in the accunnulator register is implied in the definition of the instruction. The operand need not be specified explicitly.

(ii) Imunediate Addressing Mode:

In this the operand is itself specified in the instruction operand field. Here the operand value is a constant.

Example: LOADI or LOAD_{immediate} 99



Instruction This means that 99 be (i.e. loaded the data in or the the operand) are to

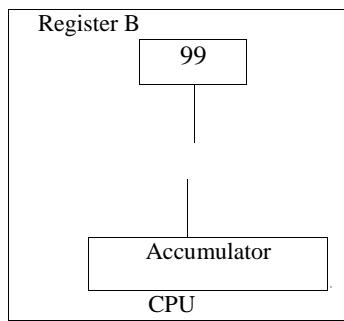
Accumulator as has been shown in the figure. In this mode the instruction has an operand field rather than an address field.

(iii) Register Mode or Register Direct Mode:

In this mode the operands reside in registers that reside within the CPU i.e. the operands reside directly in the CPU registers.

Example:

A DDRegister₁' direct B



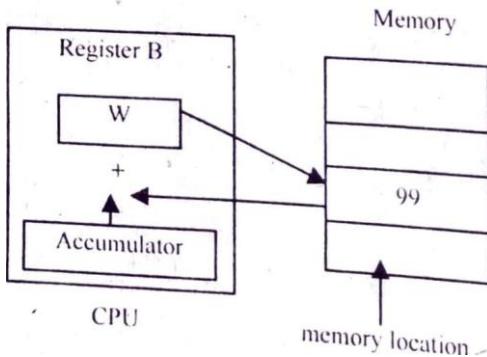
Here B is a CPU register and the content of which is 99 i.e. this content reside directly within the CPU register B (means the address of this operand is the CPU register B). Hence, this content must get added to the content of the accumulator.

(iv) Register Indirect Mode:

In this mode the instruction specifies a register in the CPU whose contents give the address of the operand in memory i.e. the content of the CPU register is the address of the operand's location in the memory.

Example:

ADDRegister_{Indirect} B



Here the content of the CPU register B (i.e. W) is the memory location that contains the operand 99. So the register w B contains the address of the operand 99. This operand (99) is to be added to the accumulator content.

two addressing modes are similar to the register indirect mode except that the These is incremented or decremented after or before its value is used to access memory ~~by~~ the content of the register (given in

the instruction) is incremented or decremented i.e. Idingly and then that tnetnory location (i.e. the resultant nwmory location) is looked acco for the actual operand.

the address stored in the register refers to table of data in metnory, it is necessary or decænnent the register after every access to the table. The increment or decrement instruction helps to achieve this.

J A D D Autotnctetnent Autodecretnent Bthe CPUJ register B has in it the location ^{Suppose} _{memory} address 2010. Let the 1st location has this address 2010 containing the operand 99. So this operand will be added to the accunllator content. Again if there is a table of memory locations starting address 2010 (and increasing downwards), then just from incrementing the register content by 1($2010+1 = 2011$) the operand in 2011 location can be accessed and added to new accurnulator content. This nnay continue until the operands in all the memory locations in the table are added to the previous accumulator content.

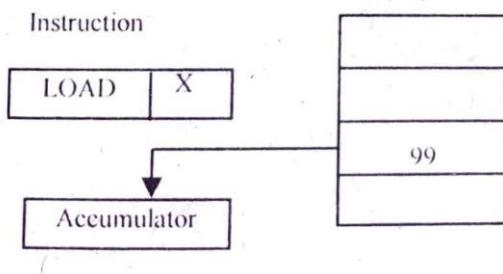
Ifhowever menwry' is increasing in descending order, then the register content needs to be decrelnented by 1 each tinne.

(vii) Direct Address Mode:

In this mode, the effective address of the operand is equal to the address part of the instruction i.e. the address part of the instruction indicates the memory location containing the operand.

Example:

LOAD



Memory Direct x

Memory

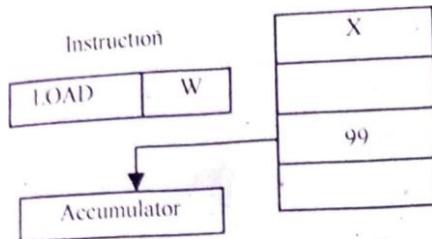
In the example, the memory location X contains the operand 99 to be loaded in x the accumulator.

(viii) Indirect Address Mode:

In this mode the address field of the instruction gives the address where the eiTective address is stored in melitory i.e. the address part of the instruction indicates the nrtnot•y location whose content is the address of the meni0t•y location containing the actual operand.

E.vatnple:

Indictet
location Menlon



In the 99. This contains I n the whose content IS the he liltinatel,s operand X accumulator.

'leniory location W

memory 1

actual operand is to
already loaded

The retraining addressing Inodes require that the address field of the instruction be add to the content of a specific register in the CPU. rhe actual operand address (i.e. t etléctive address) in these 1110des is obtained fronn the following conputation: effecti address = address part of instruction + content of CPU register. The CPU register used the coniputation may be the prograll counter, an index register, or a base regist Depending on type of CPU register, the addressing nwdes are also di tTerent. Such addressing nnodes are called:

(ix) Relative Address Mode:

In such modes the content of the CPU register is added to the adclress part of t instruction to obtain the actual address of the operand (i.e. the etTective address). Tl address part of the instruction, which is usually a signed nunnber (either positive negative) on addition to the CPU register content, gives the efTective address who position in is relative to the address of the next instruction.

Uses:

Relative addressing Inode is often used with branch-type instructions.

This mode also results in shorter address field in the instruction fornat as the relati address can be specified with a snoaller number of bits compared to the nunnber of b required to designate the entire memory address.

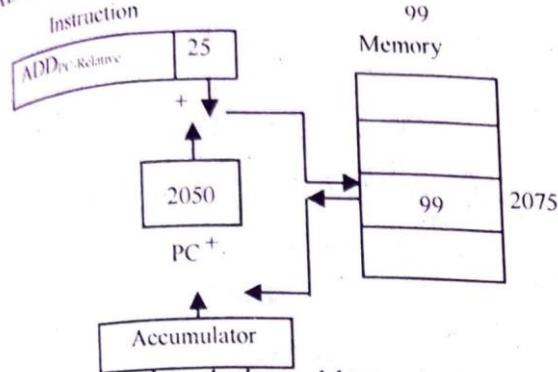
Relative addressing modes be of the lowing three types:

(a) PC (i.e. Program Counter) Relative Addressing Mode:

In this mode the content of the counter is added to the address part of instruction to obtain the actual address of the operand (i.e. the etTective address).

CO-SO

Example:
ADD_{PC-Relative} 25



Uses: Same as in relative address mode.

(b) Indexed Addressing Mode or Index R

In this mode the content of the index register is added to the address part of the instruction to obtain the effective address.

(i.e. 99) is added to the content of the the
is stored in the accumulator itself. Ret-

2075) is looked upon and its content ded
accumulator and the result after i ve addition

(b) Indexed Addressing Mode or Index Register Relative Addressing Mode:

In this mode the content of the index register is added to the address part of the instruction to obtain the actual address of the operand (i.e. the effective address). the

The example for ADDIndex Relative 25 is as follows:

ADD_I

Diagram and explanation of the example is same as the PC-Relative Mode only that the index register is now an index register instead of the PC.

CPU register

The index register can be used to access consecutive operands. This can be done by bits incrementing the register contents.

(c) Base Register Addressing Mode:

In this mode the content of the base register is added to the address part of the instruction to obtain the actual address of the operand (i.e. the effective address). the

Example:

Base Register Relative 25

ADD

Diagram and
CPU register

Uses:

explanation of the example is same as the PC-Relative Mode only that the is now base register instead of the PC.

used in computers to facilitate the relocation of programs in memory.

(x) Stack Addressing Mode:

Here the address of the operand is specified by the stack pointer (SP). The length of Instruction is the shortest as it does not include any address of the memory location or mention any register Oust like implied mode of addressing). After each stack operation,

POPULAR PUBLICATIONS

the contents of SP are automatically incremented or decremented. PUSH and POP are the most commonly used instructions of this type.

Example:

PUSH A To push the content of accumulator to the top of stack (TOS).

Uses:

Useful when PUSH and POP instructions are used in a program by the Programmer. When interrupt occurs the contents of important registers are saved into the stack. For this stack addressing is used.

2. Evaluate the arithmetic statement $X = (A * B) / (C + D)$ [WBUT in one, 2008, two 2011, and 2018] three address machines.

Answer:

Three-address

machine:

MULTIPLY T, A, B	⇒	
ADD X		X ← C + D
DIV		

Two-address

machine:

MOVE T, A	T ← A
MULTIPLY T, B	T ← T * B
MOVE X, C	X ← C
ADD X, D	⇒ X ← X + D
	DIV T

One-address machine:

LOAD A	⇒ transfer certain memory content to accumulator
MULTIPLY B	⇒ AC ← AC * B
STORE T	⇒ transfer AC content to memory location T
LOAD	⇒ transfer to ac
ADD D	⇒ AC ← AC + D
DIV T	⇒ AC ← AC / T
	⇒ transfer result

STORE X transfer result to memory location X

3. write a program to evaluate the arithmetic statement $Y = (A - B + C) + H$ '

(i) Using an accumulator type computer with one address instruction. (ii) Using a stack organized computer with zero-address instructions.

[WBUT 2015, 2016]

Answer: i) One address instruction:

LOAD A

SUB B

STORE P

LOAD C

ADD P STORE

Q

LOAD G

ADD H

DIV Q

STORE R

ii) Zero address instruction:

PUSH A

PUSH B

SUB

PUSH C

ADD

PUSH G

ADD

DIV

4. Discuss in detail the various factors that need to be considered while designing the ISA of a processor. [WBUT 20171 Answer:

The Instruction Set Architecture (ISA) is the part of the processor that is visible to the programer or compiler writer. The ISA serves as the boundary between software and hardware. The ISA of a processor can be described using 5 categories:

Basic ISA Issues

- We probably want to have some registers, so that we don't have to keep going out to slow memory, but how many registers do we want or need?
- More registers is probably better, but this will make implementing the CPU more complicated, i.e. we trade off ease of programming against ease of building the CPU. ... and we need bits in each instruction to encode the register number: 8 registers = 3 bits etc.
- It also means we have to save out more registers when we switch between programs, and we do want to support a system where multiple programs can be running at the same time.
- On the other hand, too few registers means that we will have to go out to main memory more often.

Another issue is how we deal with the special registers like the Program COUnter (PC): do we want these registers visible and accessible to the programmer?

- this would allow the **programmer** to change the PC's value by hand, which could be useful.

- or do we provide special instructions to manipulate these in only limited ways? • Most modern ISAS have 8 to 32 registers.

What size data bus do we want? This effectively sets the natural word size of the ISA.

- And because the CPU will want to fetch in word units, this also influences the size of our instructions.
 - generally, each instruction will be 1 word at minimum, or a multiple of the word

What size address bus do we want? This determines how much memory the CPU can address.

- But then we need the ability to express every address in some way.
- If the address size is too big, it can make the instruction size too big.
- For example, if the address bus is 64 bits, how will we encode the operation: load a word from address X into register R3?
 - The instruction will be at least 64 bits long to hold the address, plus bits to describe the load operation and the register which is the destination.

3 Operations

- How many operations do we want to have?
- If we have lots, this provides a rich set of operations that the programmer can perform, but it will make implementing them in silicon much more difficult. • We will also need more bits in each instruction to encode the operation: _ 5 bits => 32 operations, 6 bits => 64 operations etc.
- On the other hand, we might choose to have only a small set of simple instructions. This may force the programmer to have to combine 2 or more instructions to get something done, but the advantage is fewer bits required to encode an operation, and a simpler CPU design.

wMany Operands?

With the number of instructions decided, how many operands will each operation require?

PC to a new instruction, e.g.

POPULAR PUBLICATIONS

- If the CPU gets most of its data from registers, then we probably want to have some 3-operand instructions like
ADD RI, R2, R3, i.e. Add R2 and R3, and save the result into RI so the instruction format needs to have bits set aside to identify each of the three
- Other 3-operand instructions include instructions which compare and then divert the

BGT RI, R 2, 100, i.e. if RI > R 2, branch the CPU to instruction at current PC + 100 all instructions have 3 operations. Examples of 2-operand instructions include: LOAD R 3, i.e. get the value from memory location 4000 and load it into register R3.

SAVE R4, 5000, i.e. write R4's value out to memory location 5000.

SET R6, 23, i.e. set R6 to the literal value 23 (not the value at location 23).

And, of course, a CPU designer may think of 1-operand instructions, such as:

- INCR R4, i.e. increment the value in R4. The Java equivalent is R4++.

Later on, we will talk about the various addressing modes which a CPU designer might wish to use.

Values

Lite

Many instructions require literal values.

- e.g. in Java when we write for (i=(); 00; i++), there are two literal values: 0 and 100.
- Are we going to be able to find space in each instruction to put in literal values? If so, that will be great, but it will be wasted space if programs don't have many literal values.
- If we can't find space, then each time there is a literal, it is going to have to be stored in a register, or we are going to have to go out to memory to fetch the literal value.

Instruction Format

- Each different instruction has to be encoded differently: operation, operands, literal values, where to place the result, size of the data being operated on etc.
- What is the instruction format going to look like?
- Can we make each instruction the same size, or are some instructions going to be different sizes?

Is there going to be a single format, which makes the decoding in silicon easy, or are we going to have several different instruction types, each with a different format?

A Hypothetical Example

- Before we go any further, let's make the above concrete by doing some ISA design. Let's design an ISA with 8 registers, a 16-bit data bus and word size, and a 24-bit address bus: quite suitable for an embedded CPU, e.g. in a microwave or engine control system.
- Let's also have 3-operand instructions where the operands are all registers: Rdest=Rsrc1 op Rsrc2
- We need 3 bits to encode each register's number, so that's 9 bits out of 16 used up, leaving 7 bits.

Let's use 1 bit to encode the size of the data being manipulated: 8-bit byte or 16-bit int. That leaves 6 bits.

CO-55

- We could go for a single instruction format:

Operation	Size	Rdest	Rsrc1	Rsrc2
6 bits	1 bit	3 bits	3 bits	3 bits

- But now there's no way to copy values between the registers and the main memory, nor is there any way to put literal values into an instruction that contains a literal value.
- Let's add a second instruction format, still 10-bits wide:

Inst Type	Operation	Size	Rdest	Rsrc1	Rsrc2
0	5 bits	1 bit	3 bits	3 bits	3 bits

Inst Type	Operation	Size	Register	Literal Value
10	2 bits	1 bit	3 bits	8 bits

Inst T} pe (Operation Size Rdest Rscl 3 Rsrc2
 5 bits 1 bit 3 3 bits

 Literal Value
 Register 8 bits
 3 bits

- This allows us to do things like SET R0, 26, where the 26 is stored in the 8-bit literal section.
- But we still don't have a way to access locations in memory, so there is one final instruction format.

Inst Type	Operation	Size	Register	Memory Location
	2 bits	1 bit	3 bits	24 bits

- This instruction format is 2 words long, and the memory address is specified as a 24-bit value. This allows us to do things like LOAD R0, 4000 and SAVE R4, 500().
- We now have:
- a 1-word instruction format with 25=32 possible operations on 3 register operands, or a 1-word instruction format with 22=4 possible operations on one register and an 8-bit literal value, and or a 2-word instruction format with 22=4 possible operations on one register and

24-bit memory location

Decision Making

- Up to now, we haven't considered how we are going to modify the value in the PC so that it can deviate from the normal "next instruction" of execution.
- we need a way to skip over instructions, based on a decision, so as to implement IF statements.
- we also need to branch backward, based on a decision, so as to implement loop constructs.
- we need to jump to the start of function, and also know how to return back where we left.
- All of the above change the default PC behaviour: move to the address of the instruction.
- What decisions are we going to provide?
- Some of these we could leave out, e.g. if($R3 < R4$) is the same as if($R4 >$

co-56 is the same as if($R4 > R3$).

ORGANISATION

several choices here. We could design an instruction format which

- encodes operands to be compared,
- what type of comparison to make, change to make
 - what to the PC if the comparison is true.

5 Write short notes on the following:

a) Addressing modes [WBUT 2007, 2008, [WBUT 2012, 2011/2017]

b) Classify MRI and non-MRI instructions [WBUT 2018] c) Instruction RISC

Format [WBUT 2018]

d)

a) All types of Addressing modes: Refer to Question No. I of Long Answer Type Questions.

b) Classify MRI and non-MRI instructions:

The instruction field specifies a machine instruction or pseudo instruction. The al

instruction field in an assembly program may specify one of the following:

- A memory-reference instruction (MRI)
- A register-reference or input-output instruction (non-MRI)
- A pseudo instruction with or without an operand

A memory-reference instruction occupies two or three symbols separated by spaces. The first will be three letter symbol defining an MRI operation code such as AND,

POPULAR PUBLICATIONS

ADD, LDA, STA etc. the second is a symbolic address. The third symbol, which may or may not be present, is the letter I. If I is missing, the line denotes a direct address.

instruction. The presence of the symbol I denotes an indirect address instruction. A non-MRI is defined as an instruction that does not have an address part. A non-MRI is recognized in the instruction field of a program by any one of the three-letter symbols for the register-reference and input-output instructions. A symbol address in the instruction field specifies the memory location of an operand. This location must be defined somewhere in the program by appearing again as a label in the first column.

c) Instruction Format:

An instruction of the bits of an format must implicitly or explicitly, zero or more operands. An instruction is normally made up of a combination of an operation code and an operand most commonly by its location or address in memory. An instruction set architecture is

(Mode	Address or operand
)opcode		

format defines the layout instruction, an instruction include an opcode and

o an abstract model of a computer, it is also referred as architecture or computer architecture. The main components of an instruction are-

I) Opcode: It defines which instructions to be executed.

II) Operands: It defines the data on instruction to be executed.

d) RISC:

RISC It is a stands Illicroprocessor for Reduced that Instruction is designed set to cotnputer.perforln a smaller number of types Of COIÄÄPuter

instructions so tlvat it can operate at a higher speed. Sonne advantages Of RISC are_

- A new microprocessor can be developed and tested more quickly if one Of its aims is to be less complicated.
- Operating systems and application program who use the microprocessor, s instructions will find it easier to develop code with a smaller instruction set.
- rhe simplicity of RISC allows more freedom to choose how to use the space on a nlicroprocessor.
- Higher level language compilers produce more efficient code than fortiÄerly because they have always tended to use the smaller set of instructions to be found in a RISC coniputer.

The characteristics of RISC Architectures are-

- i) It is highly pipe lined.
- ii)It has fixed length instructions and few addressing modes. iii) it is mainly controlled by hard wired.
- iv) It has single cycle instructions, and three register operands allowed per instructions.
(Example —add R2, RA)

MEMORY ORGANIZATION

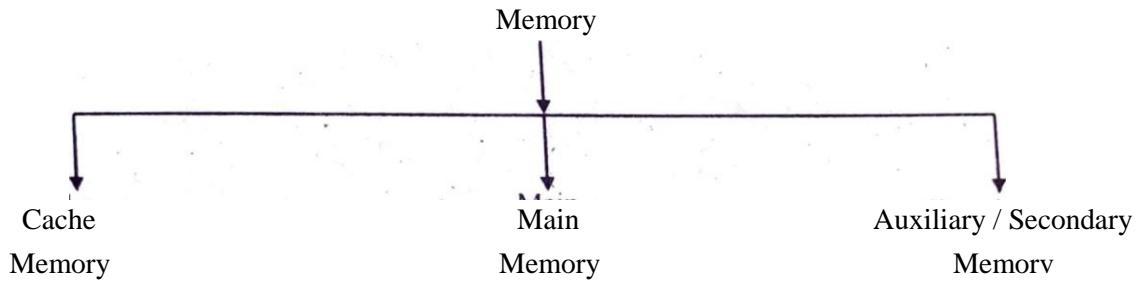
Types of MOS: MOS can be of the following 3 types:

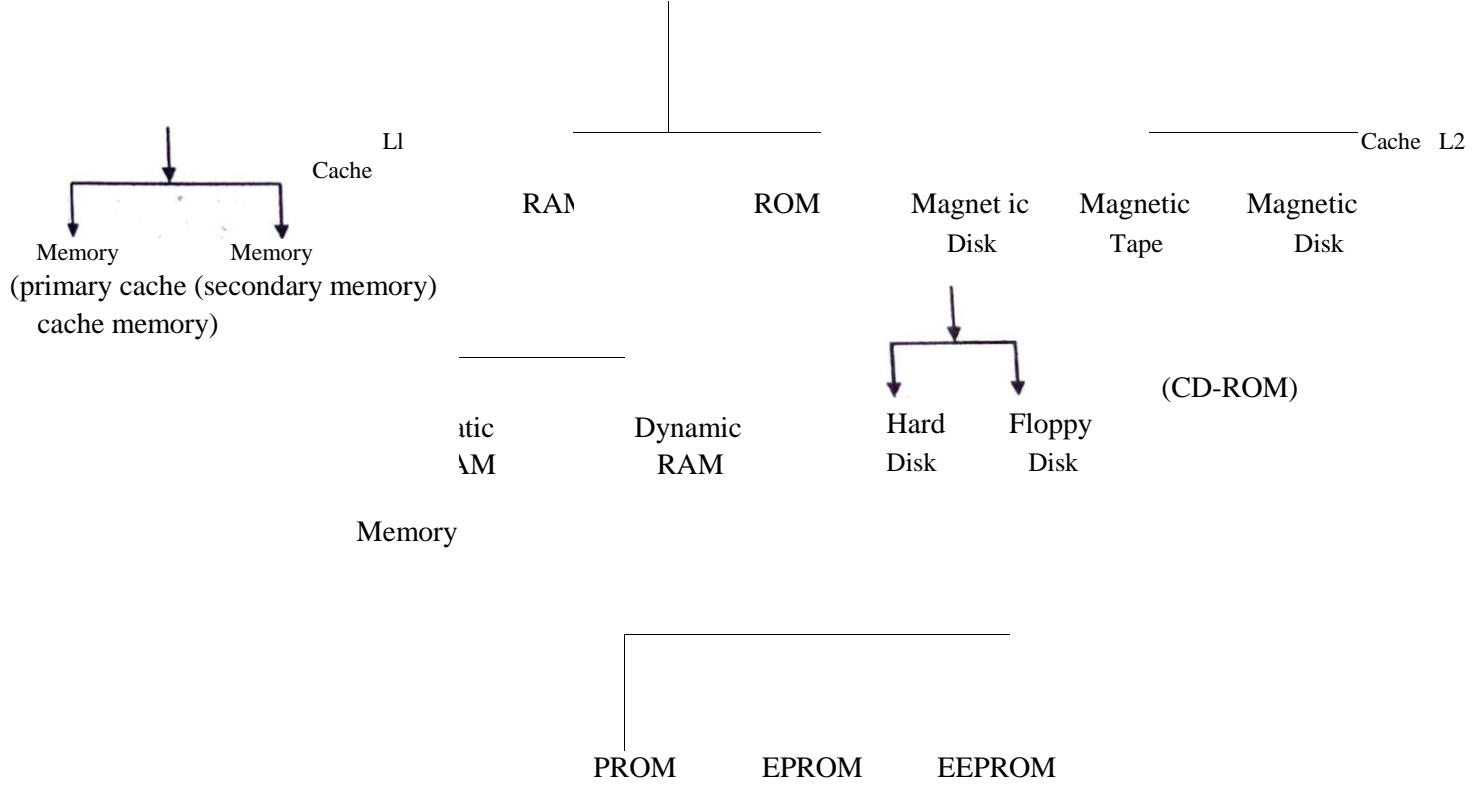
- (d) pMOS: The p-channel (+ve) MOS which depends on the flow of holes.
- (h) N MOS: The n-channel (—ve) MOS which depends on the flow of electrons. These are commonly used in circuits with only one ty pe of MOS transistor.

Chapter at a Glance

(c) CMOS: The complementary MOS (CMOS) technology uses both PMOS and NMOS transistors connected in a complementary fashion in all circuits.

Memory classification and hierarchy: Memory can be broadly classified into three main parts: the cache memory, the main memory and the auxiliary memory. The cache memory lies between the CPU and the main memory and is the fastest, smallest and the most expensive of all the memory units. The auxiliary or the secondary memory unit is the slowest, largest and the least expensive of all the memory units. The main memory lies in between the two. **Memory Classification:** Memory can be classified accordingly:





Memory Hierarchy: The block

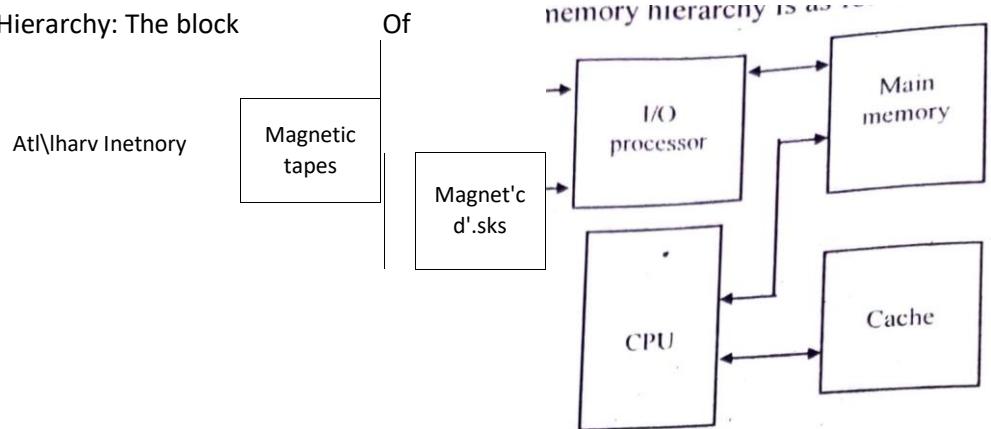


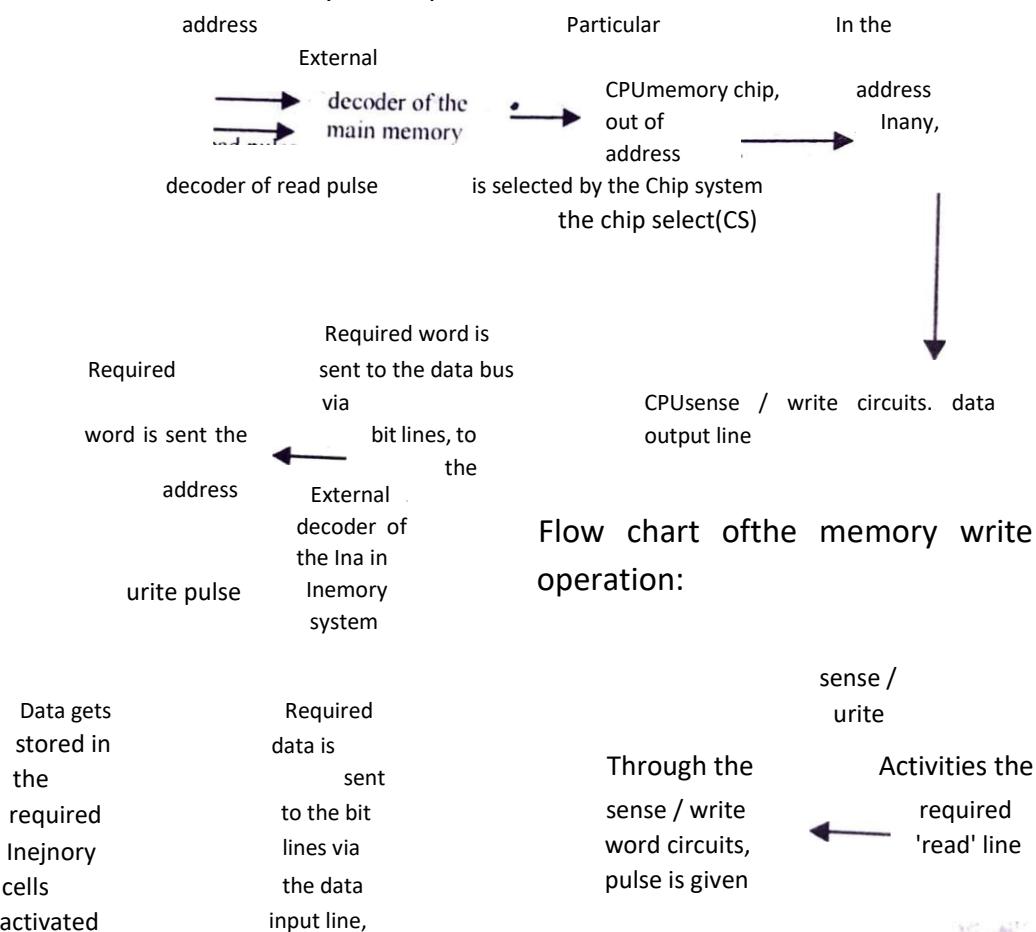
Fig: Memory hierarchy in a computer system

Factors on which Memory Hierarchy depends: There are various factors on which the basic hierarchy of memory depends. These are cost, storage capacity (size), and speed and access

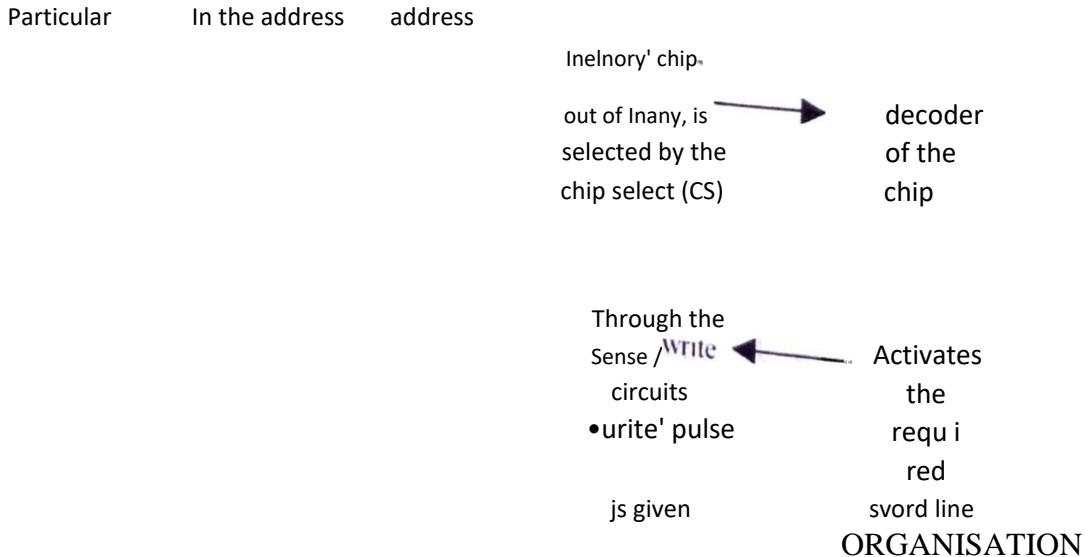
- time.

Memory read/write access:

Flow chart of the memory read operation:



COMPUTER



Storage Dynamic RAM:

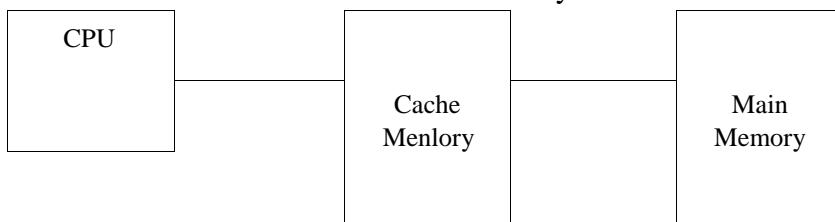
(i) static RAM:

It consists of internal flip-flops that store the binary information. Stored information remains valid as long as power is applied to the unit.

(ii) Dynamic RAM:

A dynamic RAM loses its stored information in a very short time (a few milliseconds) even though the power supply is on.

Cache memory: Cache memory is a very small but very fast memory. It is the smallest but fastest among all other memory units. A very-high-speed memory, it is sometimes used to increase the speed of processing by making current programs and data available to the CPU at a rapid rate. It lies between the CPU and the main memory unit.



- Virtual memory: It is a technique that allows the execution of processes that may not be completely in main memory. This concept used in some large computer systems permit the user to construct programs as though a large memory space is available. Virtual memory is used to give programmers the illusion that they have a very large memory at their disposal, even though the computer actually has a relatively small main memory.

- Diagram:

The below figure shows the organization that implements virtual memory.

POPULAR PUBLICATIONS

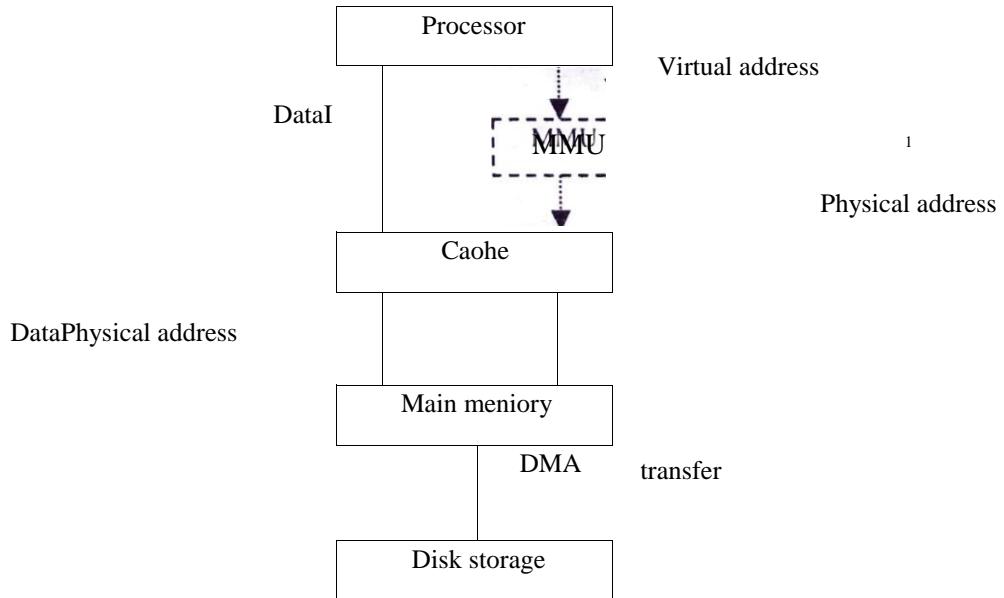


Fig: Virtual memory organization

Multiple Choice Type Questions

Technique of placing software in a ROM semiconductor chip is called [WBUT 2003, 2008] a) PROM

Answer: (c) b) EPROM c) FIRMWARE d) Microprocessor

POPULAR PUBLICATIONS

Cache memory

- a) increases performance
- machine cycle increases

Answer: (a)

[WBUT 2006, 2008, 2011, 2013]

- b) reduces performance

none of these

Associative memory is a

- a) very cheap memory
- content addressable memory

Answer: (c)

b) pointer addressable memory
slow memory

How many RAM chips of size (256 K
memory?

bit) are required to build d) 32 1 M
24

- b) 10

Answer: (d)

5. How many address bits are required for a 1024 10x 8 memory?[WBUT 2007, d)

None 2011, of these20131

- c)

- a) 1024

Answer: (c)

6. The principle of locality justified the use of b) Polling[WBUT 2007, 2012, 2015,

20181 a) Interrupt d) Cache memory

- c) DMA

Answer: (d)

7. A major advantage of direct mapping of a cache is its simplicity. The main
disadvantage of this organization that [WBUT 20071

POPULAR PUBLICATIONS

- a) It does not allow simultaneous access to the intended data and its tag
- b) It is more expensive than other types of cache organizations
- c) The cache hit ratio is degraded if two or more blocks used alternately map onto the same block frame in the cache
- d) Its access time is greater than that of other cache organizations

Answer: (c)

8. The purpose of ROM in a Computer System is [WBUT 20101]

- a) to store constant data required for computers own use
- b) to help reading from memory.
- c) to store application program
- d) to store Os in memory

Answer:

9. Which one dose not posses the characteristics of a memory element?

- a) A toggle switch
- c) An AND gate
- b) A lamp
- d) None of these

Answer: (c)

[WBUT 20101]

10.

pata MAR from memory location after fetching b) is MOR deposited by memory
 (WBUT in 2010)

- a)
- d) Status Register

Answer: (b)

11.

Virtual memory system allows the employment of (WBUT 2010)

- More than address space b) The full address space
- c) More than hard disk capacity d) None of these

Answer: (a)

12. A system has 48-bit virtual address, 36-bit physical address and 128 MB main memory. How many virtual and physical pages can the address space support? [WBUT 2010]

- a) $2^{36}, 2^{24}$
- b) $2^{12} 2^{36}$
- c) $2^{24}, 2^{34}$
- 36

Answer: (b)

13. Maximum number of directly addressable locations in the memory of a processor having 10 bits wide control bus, 20 bits address bus and 8 bit data bus

[WBUT 2012]

- a) 1 K
- b) 2 K
- c) $2^{10} \times 2^{20} \times 8$
- d) none of these

14. Periodic refreshing is needed [WBUT 2012]

- a) SRAM
- b) DRAM
- c) ROM
- d) EPROM

is

Answer: (b)

15. Physical memory broken down into groups of equal size is called [WBUT 20121

- a) page
- b) tag
- c) block/frame
- d) index

Answer:(c)

16. Bi-directional buses use [WBUT 20121

- a) tri-state buffers
- b) two tri-state buffers in cascade
- c) two back to back connected tri-state buffer in parallel
- d) two back to back connected buffers

Answer: (c)

17. Micro Instruction are kept in [WBUT 20121

- a) main memory
- b) control memory
- c) cache memory
- d) none of these

Answer: (b)

Size of virtual memory is equivalent to the size of

[WBUT 20131

- a) main memory
- b) cache memory
- c) secondary memory
- d) both (a) and (c)

Answer: (c)

19. The associative access mechanism is followed b) cache in memory

- a) main memory
- d) both (a) and (b)
- c) magnetic disk

Answer: (d)

d) [WBUT none of 20141these

20. The users view of memory is supported byc) both

- a) paging
- b) segmentation

Answer: (a)

[WBUT 2014)

21. The largest delay in accessing data on disk b) is rotation due totime

- a) seek time
- d) none of these
- c) data transfer time

Answer: (a)

22. Address of memory location for fetching data needs to be deposited [WBUT in
memory2014)

in d) status register

- a) MAR
- c) MBR

Answer: (a)

23. Size of virtual memory is equivalent to the size of

CWBUT 20141

- hard disk
- b) CPU
- c) floppy disk

d) none of these

a)

Answer: (a)

24. If k be the number of registers and n be the size of each register, then in
order to construct n-line common bus system using tri-state buffers, the total
number of tri-state buffers and the size of decoder would be [WBUT 2015]

- a) $n*k$ and 2-to-4
- b) $n*k$ and $\log_2 k$ -to- k
- c) k and $\log_2 n$ -to- n
- d) $n*k$ and $\log_2 n$ -to- n

Answer: (b)

25. RAM is called DRAM (Dynamic RAM) when [WBUT 2016]

- a) it is always moving around data
- b) is requires periodic refreshing

POPULAR PUBLICATIONS

- c) it can do several things simultaneously
- d) none of these

Answer: (b)

26. In order to execute a program instructions must be transferred from memory along a bus to the CPU. If the bus has 8 data lines, at most one 8 bit byte can be transferred at a time. How many memory accesses would be needed in this case to transfer a 32 bit instruction from memory to the CPIJ?

[WBUT 2016]

Answer: (d)

27. A required computer's for memory memory is address composed if the of smallest 8K words addressable of 32 bits each. memory HOW many unit is bitsa are

(WBUT 2016)

word?

- a) 13
- c) 10

Answer: (a)

28 Cache memory refers to

20161

- a) cheap memory that can be plugged into the mother board to expand main memory
- b) fast memory present on the processor chip that is used to store recently accessed data
- c) a reserved portion of main memory used to save important data
- d) a special area of memory on the chip that is used to save frequently used data

Ans»er: (b).

29. Write Through technique is used in which memory for updating the data?

- a) Virtual memory
- b) Main memory
- c) Auxiliary memory
- d) Cache memory

[WBUT 2016]

Answer: (d)

30. _is generally used to increase the apparent size of physical memory.

- a) Secondary memory
- b) Virtual memory
- c) Hard disk
- d) Disks

[WBUT 20171]

Answer: (b)

Answer: (c)

32. A 24 bit address generates an address space of locations
a) 1024 b) 4096 [WBUT 20171 C] c) 2^{48} d) 16,777,216

Answer: (d)

Answer: (b)

34' During transfer of data between the processor and memory we use

Answer: (d)

[WBUT 20171

35. The return address of the Sub-routine is pointed to bY

- a) IR
 - b) Address bus
 - c) MAR
 - d) Special memory registers

Answer: (b)

Short Answer Type Questions

1. Given the following determine the size of the sub fields (in bits) in the address for Direct Mapping, associative and set associative mapping cache schemes: • We have 256 MB main memory and 1 MB cache memory.

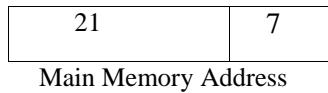
- The address space of this processor is 256 MB.
 - The block size is 128 bytes. [WBUT 2004, 2007, 2012, 2013]
 - There are

As the size of the main memory is 256 MB hence there are 28 bits (as $256 = 2^8$ and 1 MB = 2²⁰ bytes and hence $256 \text{ MB} = 2^8 \times 2^{20} = 2^{28}$) in the main memory address or the address size of main memory is 28 bits.

POPULAR PUBLICATIONS

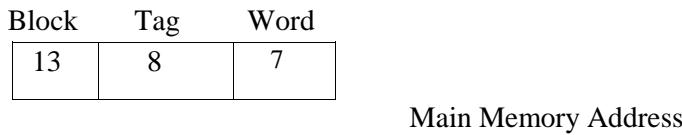
Size of the sub-fields for associative mapping cache schemes:

Each block size is 128 bytes or 27 bytes. Hence number of main memory blocks = $2^{28}/2^7 = 2^{21}$. So number of bits in the tag field is 21 and that in the word field is 7 (as block size is 128 bytes). Tag Word



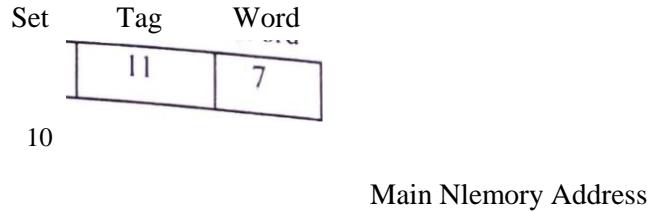
Size of the sub-fields for direct mapping cache schemes:

Now size of cache memory is 1 MB = 2^{20} bytes. Hence number of cache memory blocks = $2^{20}/2^7 = 2^{13}$. So number of bits in the block field = 13. Now out of the total main memory address size of 28 bits, word field contains 7 bits and the block field contains 13 bits. Hence, the number of bits in the tag field is $28 - (13 + 7) = 8$.



Size of the sub-fields for set-associative mapping cache schemes:

In this case, there are 8 blocks per cache set and the total numbers of cache blocks are 2^3 . So number of sets in the cache memory are $213 / 8 = 213 / 2^3 = 210$. Hence, number of bits in the set field is 10 and that in the word field is 7. So number of bits in the tag field = $28 - (10 + 7) = 11$



COMPUTER ORGANISATION

What is dirty bit?

2006, 20101

sns»cr:cope \S'ith the storage capacity Of the tmain Inemory, pages are swapped in and out

{the main and if and that the page secondary is not there Inenu»ry. in the When main memory a particular it is page swapped is required in from in the the

secondary storage the area. particular Dirty bits page are used modified in the ever page since table to it keep is brought track of in individual the main lee'S. whenever the contents of a page is modified (i.e. something is written on the its respective dirty bit is set. If a particular page needs to be swapped out of the

In memory, the O.S checks its dirty bit to see whether the page is regularly used (or is in use). If it is regularly used or is in use currently, the particular page is not swapped out.

3. Discuss with suitable logic diagram the operation of an SRAM cell. [WBUT 20061

OR,

Draw the internal cell diagram of SRAM cell.

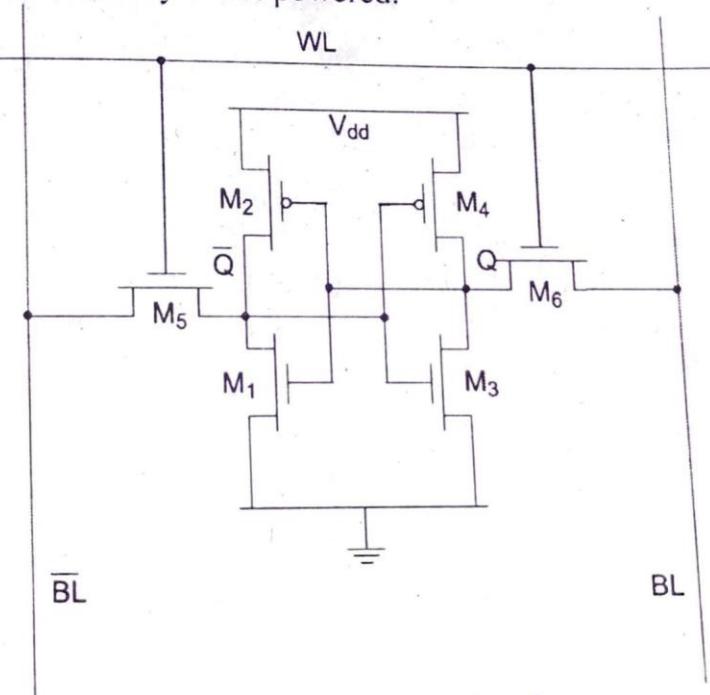
[WBUT 2013)

Answer:

static random access memory (SRAM) is a type of semiconductor memory where the word static indicates that, unlike dynamic RAM (DRAM), it does not need to be periodically refreshed, as SRAM uses bistable latching circuitry to store each bit. SRAM exhibits data

POPULAR PUBLICATIONS

remanence, but is still volatile in the conventional sense that data is eventually lost when the memory is not powered.



the memory is not powered.

Each bit in an SRAM is stored on four transistors that form two cross-coupled inverters. This storage cell has two stable states which are used to denote 0 and 1. Two additional access transistors serve to control the access to a storage cell during read and write operations. A typical SRAM uses six MOSFETs to store each bit. In addition to such 6T SRAM, other kinds of SRAM chips use 8T, 10T, or more transistors per bit. This is sometimes used to implement 11T than one (read and/or write) port, may be

co-G7

POPULAR PUBLICATIONS

useful in certain types

SRAM circuitry.

of video

files implemented

and files

with multi

Generally, the fewer silicon transistors needed per cell, fixed, the smaller using smaller each but such cell cells can be IT socell
f pro

ignore bits on one wafer reduces the cost per bit of memory.

Memory cells that use fewer than 6 transistors are possible DRAM, not SRAM.

Access to the cell is enabled by the word line (WL in figure) which controls the access transistors Ms and which, in turn, control whether the cell should be to the bit lines: BL and BL. They are used to transfer data for both read and operations. Although it is not strictly necessary to have two bit lines, both the signal

•ts inverse are t} pically provided in order to ilnprove noise margins.

During read accesses, the bit lines are actively driven high and low by the inverters in t SRAM cell. This ilnproves SRAM bandwidth compared to DRAMs—in a DRAM, the line is connected to storage capacitors and charge sharing causes the bitJine to swirl upwards or downwards. The synunetric structure of SRAMs also allows for differentik signaling, which Inakes Sinall voltage swings more easily detectable. Another differenc with DRAM that contributes to Inaking SRAM faster is that conunercial chips accept al address bits at a time. By comparison, comn10dity DRAMs have the address In ultiplexed in two halves, i.e. higher bits followed by lower bits, over the same package pins in order to keep their size and cost down.

The size of an SRAM with address lines and n data lines is 2^m words, or $m \times n$ bits.

4. What is concept of virtual memory.

[WBUT 2007]

What is virtual memory? Why is it called virtual?

WBIJT 2008, 2010, 2012,

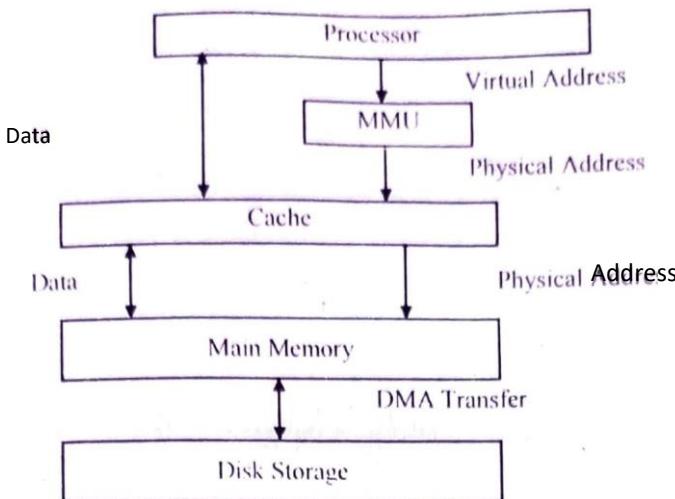
20141

In a memory hierarchy system, program and data are first stored in auxiliary

Portions of a or data are brought into main as they are needed by CPU. Virtual memory is a concept used in some large computer systems that pernlit user to construct programs as though a large meniory space is available equal to totality of auxiliary memory. Each address that is referenced by the CPU goes through an

JDelDory. Virtual is used to give programmers the illusion that the'y have program]a very address mapping from the so-called virtual address to a physical address in large at their disposal, even though the computer actually has a relatively small main Inemory. A virtual mejnory systelll provides a mechanism for translating generated address into correct IDain memory locations. This is done dynamically, while programs are being executed in the CPU. The translation or mapping is handled

~~automatically by the hardware by means of a mappin~~



5. A disk drive has 20 sectors/track, 4000 bytes/sector, 8 surfaces all together. Outer diameter of the disk is 12 cm and inner diameter is 4 cm. Inner-track space is 0.1 mm. What is the no. of tracks, storage capacity of the disk drive and data transfer rate there from each surface? The disk rotates at 3600 rpm.
 [WBUT 2012] Answer:

Radial distance covered by all the tracks lying on a surface = $12 - 4/2 = 4$ cms.

So. number of tracks/surface of the entire disk = $4 * \pi * (12^2 - 4^2) / (2 * 0.1) = 400$.
 Number of tracks/surface of the entire disk = $4 * \pi * (12^2 - 4^2) / (2 * 0.1) = 400$.
 Number of bytes transferred from each surface during one revolution of the disk = $20 * 4000 = 80,000$ bytes.

Time per revolution = $60 / 3600 = 1 / 60$ sec

So, data transfer rate from each surface of the disk drive = $60 * 80,000 / 60 = 80,000$ bytes/sec.

6. What do you mean by Stack memory?

[WBUT 2012]

OR,

Explain stack based CPU. [WBUT 20181 Answer:

A useful feature that is included in the CPU of most computers is a STACK or last-in, first-out (LIFO) list. A stack is a storage device that stores information in such a manner that the last item stored in a stack is the first one to be retrieved. Stack is a one-way list and only one information can be accessed at a time. A set of registers constitutes a stack. A stack can be

placed in a portion of a large memory or it can be organized as a Collection of a finite number of memory words or registers. this is known as register stack• A stack can exist as a stand-alone unit or can be innplelnented in a random-access memory attached to a CPU. This is known as memory stack. The stack in digital

Computers register that is holds essentially the address a melnory for the unit stack with is an called address the register stack pointer that can (SP) count because only. Theits value always points at the top item of the stack.

co-69

POPULAR PUBLICATIONS

The two basic operations that can be performed on a stack are the insertion and deletion. In the insertion operation, which is also known as push, information is stored at the top of stack (TOS) position in the stack and in the deletion operation, known as popping, the TOS is retrieved.

Figure shows a 64-word stack organization. The stack grows upward from location 0 to location 63, which is the final TOS. On addition of every new element, the TOS value is incremented by one until the stack is full, marked by FULL = 1 (FULL is a flip-flop which is 1 when the stack is full and EMPTY is a flip-flop which is 1 when the stack is empty).

The first element is pushed in the stack at the SP = 1 location i.e. the first location. So when SP = 0, it means that the stack is full and hence FULL is 1. So EMPTY is 0 i.e. stack is not empty.

The steps for the PUSH operation are as follows:

Initially, SP is cleared to 0, EMPTY is set to 0 and FULL is cleared to 0, so that SP points to the word at address 0 and the stack is marked empty and not full. If the stack is not full (if FULL=0), a new item is inserted with a PUSH operation. The PUSH operation is implemented with the following sequence of microoperations:

$SP \leftarrow SP + 1$ \Rightarrow Stack pointer is incremented.

$M[SP] \leftarrow DR$ Item, from data register, is stored on the TOS. If ($SP = 0$) then ($FULL \leftarrow 1$) To check if stack is full.

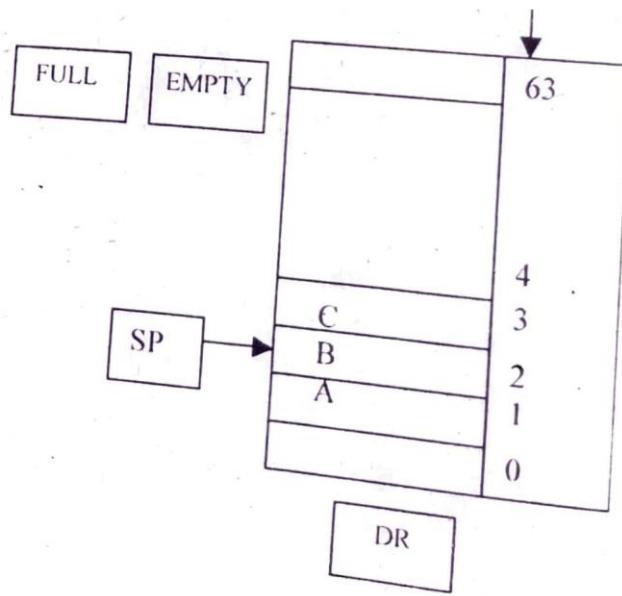
$EMTY \leftarrow 0$ \Rightarrow Mark the stack is not empty.

The steps of POP operation are as follows:

A new item is deleted from the stack if the stack is not empty (if EMPTY=0). The POP operation consists of the following sequence of microoperations:

$e \Rightarrow$ Item from TOS is popped and stored in the data register.
 $SID \leftarrow 1$ \Rightarrow Stack pointer is decremented.
If($SP = 0$) then ($EMTY \leftarrow 1$) \Rightarrow To check if stack is empty. \Rightarrow
 $DR \leftarrow M[SP]$ \Rightarrow Mark the stack is not full.

POPULAR PUBLICATIONS



cogo

COMPUTER

1. what do you mean by Logical address space and Physical address space? [WBUT 20141]

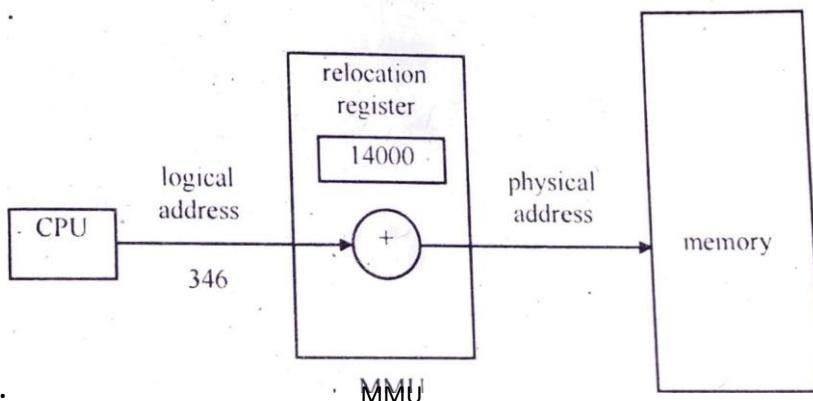
\noscr:address generated by the CPU is commonly referred to as a logical address, whereas an address seen by the memory unit -that is, the one loaded into the register of the memory- is commonly referred to as a physical address.

The compile-time and load-time address-binding methods generate identical logical and physical addresses.

However, the execution-time address-binding scheme results in differing logical and physical addresses. In this case, we usually refer to the logical address as a virtual address.

The run-time mapping from virtual to physical addresses is done by a hardware device called the Memory-Management unit (MMU).

- In MMU, the value in the relocation register is added to every address generated by a user process at the time it is sent to memory.
- The user program deals with logical addresses; it never sees the real physical



addresses.

8. Describe stack based CPU. [WBUT 2015] Answer:

Answer:

Stack-based computer operates instructions, based on a data structure called stack. A stack is a list of data words with a Last-In, First-Out (LIFO) access method that is included in the CPU of most computers. A portion of memory unit used to store operands in successive locations can be considered as a stack in computers. The register that holds the address for the top most operand in the stack is called a stack pointer (SP). The two operations performed on the operands stored in a stack are the PUSH and POP. From one end only, operands are pushed or popped. The PUSH operation results in inserting one operand at the top of stack and it decreases the stack pointer register. The POP operation results in deleting one operand from the top of stack and it increases the stack pointer register.

For example, Figure shows a stack of four data words in the memory. PUSH and POP instructions require an address field each. The PUSH instruction has the format: PUSH <memory address>

POPULAR PUBLICATIONS

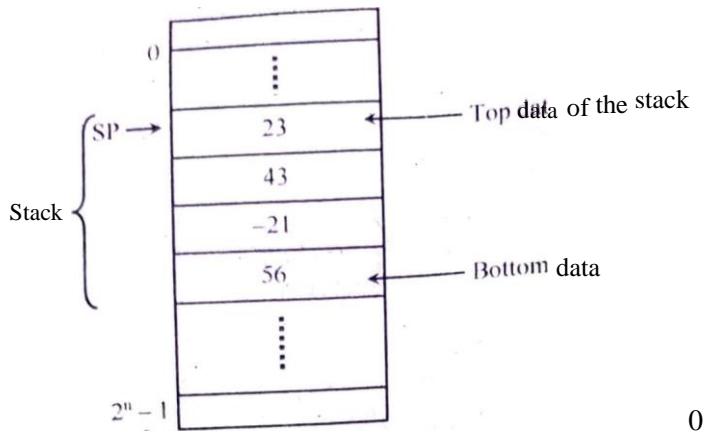


Fig: A stack of words in memory

The PUSH instruction has the format: the data word at specified address to the top of the stack

The POP instruction

POP <memory address>

The POP instruction deletes the data word at the top of the stack to the specified address. The stack pointer is updated automatically in either case. The PUSH operation can be implemented as

SP ← SP - I

decrement the SP by I

SP⁴ ← <Memory address>

store the content of specified memory address into Sp

i.e., at top of stack

The POP operation can be implemented as

<memory address>⁴ ← SP transfer the content of SP (i.e., top 11-bit data) into specified memory location SP ← SP + 1 Increment the SP by I

There are three main advantages of general-register based CPU organizations over stack based CPU organizations.

- In general-register based CPU organizations, reading a register does not affect its content, whereas, in stack based CPU organizations reading value from the top of the stack removes the value from the stack.
- In general register-based CPU organizations, any register from register file can be chosen to keep values while a program; whereas, in stack based CPU organizations, accessing values is limited by the LIFO (last-in first-out) nature of the stack.
- Since, fewer memory references are made by programs written in general register based CPU organizations, the effective execution is faster than that in stack based CPU organizations, where generally stack is implemented by memory locations and locations are accessed in LIFO nature.

POPULAR PUBLICATIONS

9. Suppose we are given RAM chips each of size 256 x 4. Design a 2K * 8 RAM system using this chip as the building block. Draw a net logic diagram Of implementation.

[WBUT 2015, 20161

co-72

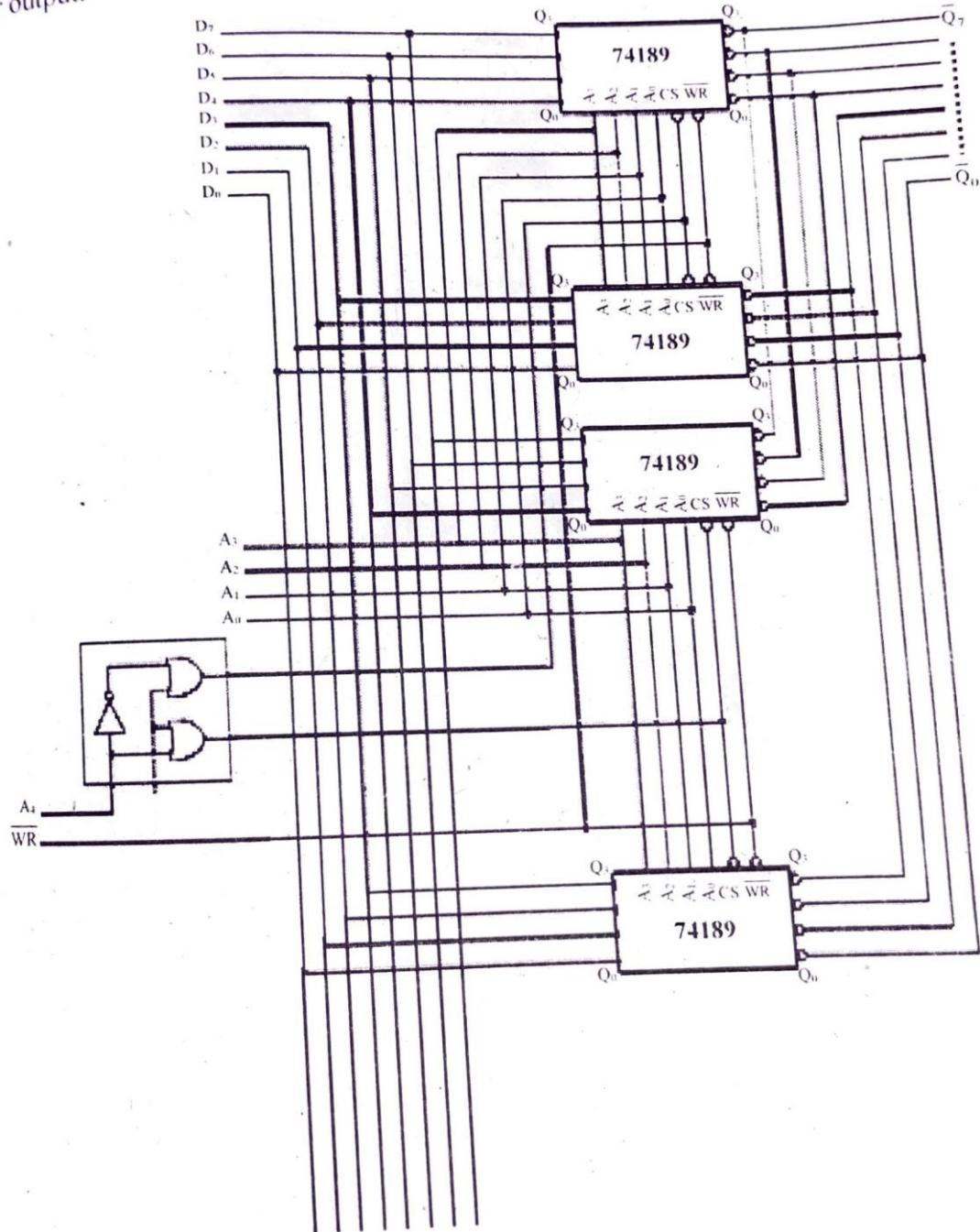
ORGANISATION

Answer: $2^{11} * 8$ bytes

Answer:
 $2KB = 2^{11} * 8 \text{ bytes}$
 We have RAM chips of $256 * 4$ which means these are 256 rows in the RAM with 4 bits in each row that is one nibble of data.
 $(2^{11} * 8) / (2^8 * 4) = (2^{11-8}) * 2 = 2^3 * 2 = 8 * 2$.
 Using 8 rows and 2 columns of $256 * 4$ chips.

$$\text{each row that } \frac{1}{2} * 8 / (2^8 * 4) = (2^{11-8}) * 2 = 2^3 * 2 = 8 * 2.$$

So we will be needing 8 rows and 2 columns of $256 * 4$ chips.
 Below we show two rows, rest will be same and the chips will be selected on the basis of decoder output.



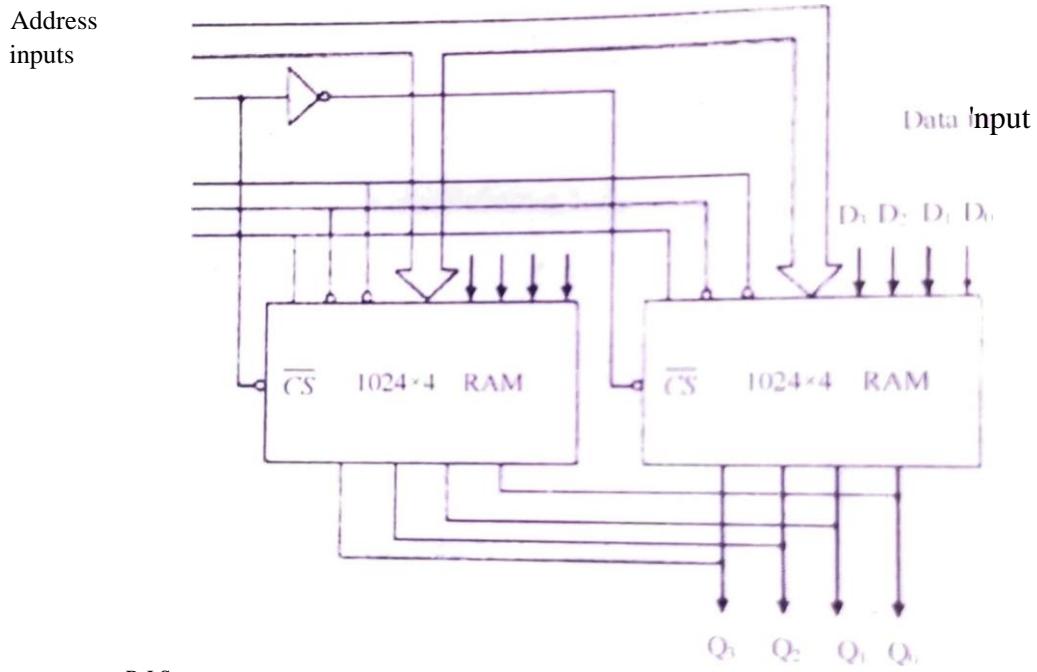
co-73

[WBUT 2017]

10. Two 1024×4 bits RAM chips are given. Design a memory of size 2048×4 bits

Answer:

In a memory has 1024 word capacity and it can store $1024 \times 4 = 4096$ bits. To expand the memory size from bits to 2048×4 bits, two RAM ICs are required. Figure I shows the connection of two RAM to develop a 2048×4 RAVI Ten-address lines Ao to A9) are directly connected with memory IC. The chip select line is connected with most significant bit address line A10) and inverted M_{SEN} connected with chip select line of the next IC. So that memory addresses from 0 to 1023 are located first memory IC1 and memory addresses from 1024 to 2047 are also located in memory IC2. One memory will be selected at a time and data out from one only. Therefore, corresponding output terminals are connected together for output.



R.J.S

POPULAR PUBLICATIONS

Fig: 1024/4 RAMS expanded to design a 2048
x4RAM

11. Define latency time of a memory. 211k!

Answer:

Random access memory latency (RAM latency) is the delay that occurs in data transmission as data moves from computer RAM and the time to fetch data from RAM than it takes to retrieve it from cache memory.

12. a) What are the advantages of associative mapping over direct mapping? Initially
- b) Consider a series of address references given: 2, 3, 11, 16, 21, 13, 64 and Assuming a direct mapped cache with 8 one-word blocks that is empty' label each reference in the list as a hit or a miss and show the final Contents of the

[WBUT 2017]

CO-a

g) Associative be mapped anywhere 1) YIPPING is in more cache flexible Inemory.than direct mapping. An Main memory block

b) index = address % 8 = address / 8

Address	Index		Hit/Miss
2	2		Miss
3	3	0	Miss
	3		Miss
16	0	0	Miss
21	5	2	Miss
13	5		Miss
64	0	8	Miss
48		6	Miss

Index	Hit/Miss
0	48
2	2
3	11
4	
5	13
6	
7	

13. What is the limitations of direct-mapped cache? Explain with an example, how it can be improved into set-associative cache? [WBUT 2018] Answer:

1st Part:

The limitations of direct-mapped cache:

Each block of main memory maps to a fixed location in the cache, therefore if two different blocks map to the same location in cache and they are continually referenced the two blocks will be continually swapped in and out, which is known as Thrashing.

2nd Part:

We are dividing both main memory and cache memory into blocks of same size i.e. 32 bytes.

Therefore, cache size = (Number of sets) * (size of each set) * (cache line size) so' using the above formula we can find out the number of sets in the cache memory i.e.

$$\text{Number of sets} = \frac{\text{Cache Size}}{\text{Size of each set} \times \text{Cache Line Size}}$$
$$\text{Number of sets} = \frac{219}{(2 \times 25)} = 2^{13}$$

co-75

the direct mapping scheme is used when an address is mapped to a set, mapping is limited within a set.

[WBUT
201

14. Average memory access time depends on which factors?

Ans: It is a common metric as memory access time depends on three parameters.

average system performance. It depends on, hit or miss latency, miss rate and miss penalty provide a quick analysis of memory systems.

Hit latency is the time to hit in the cache. Miss rate is the frequency of cache misses while average penalty is the cost of a cache miss in terms of time.

Long Answer Type Questions

1. a) Briefly explain the two 'write' policies: write through and write back for cache design. What are the advantages and disadvantages of both the methods?

[WBUT 2004, 2006, 2007, 2010]

OR,

Briefly explain write-through and write-back policies. [WBUT 2009, 2011, 2012] OR,

What are 'write through' and 'write back' policies in cache memory?

[WBUT 2013, 2014, 2015, 2018]

Answer:

Write Through: The simplest and most commonly used procedure is to update main memory with every memory write operation, with cache memory being updated in parallel if it contains the word at the specified address. This is called the write-through policy.

Advantage:

1. Simple and easy to implement and hence it is most commonly used cache write method.
2. Main memory and cache memory always contain the same data in them.
3. Effective in DMA transfers as the I/O device communicating with main memory always receives the most recent updated data from the main memory.

Disadvantage:

1. Slow (time consuming), as always two memories(cache and main memory) need to get updated simultaneously.
2. The policy will not work if the specified address location in the cache memory does not hold the required word to be updated.

Write Back: In this only the cache location is updated during a write operation. The location is then marked by a tag so that later when the word is removed from the cache it is copied into main memory.

ql dediltage:than the Previous policy as cache and nnain memory locations do not get updated
the final copy of the updated word gets stored at the main memory•

pivddvalltage:

In a back policy, data (Inodified or not) is written to the main memory finally.
suppose if the data is not Inodified at all, then the same data (unmodified) will be
to the Inain Inemory i.e. sanne data will get overwritten in the main memory.
this is tinw consti11ing and hence acts as an overhead.

b) Explain the difference between full associative and direct mapped cache mapping approaches. [WBUT 2004, 2007, 2009, 2011, 2014, 2015] Write the advantage of virtual memory. [WBUT 2008, 2010] Answer:

1st Part:

Differences between full associative (i.e. set-associative) and direct mapped cache mapping approaches are as follows:

Direct ma in

Full-Associative ma in

- | | |
|--|---|
| 1. suffers from contention problem as Choice of block replacement is nmore and hence rovides few choice of block re lacenlent. suffers much less from contention roblem. | |
| 2, Slow process. Much faster compared to direct mapping techni ue. | |
| 3. Less ex ensive (hardware). | Much less ex ensive than direct ma ing. |

2nd Part:

Advantage:

1. It provides efficiently the total available nnemory space to shared by different users.
2. The programer is not required to take care of storage allocation while writing programs.
3. The access rate is high.
4. The program execution is made independent of capacity and configuration of memory.

Discuss the concept of associative memory unit using suitable example.

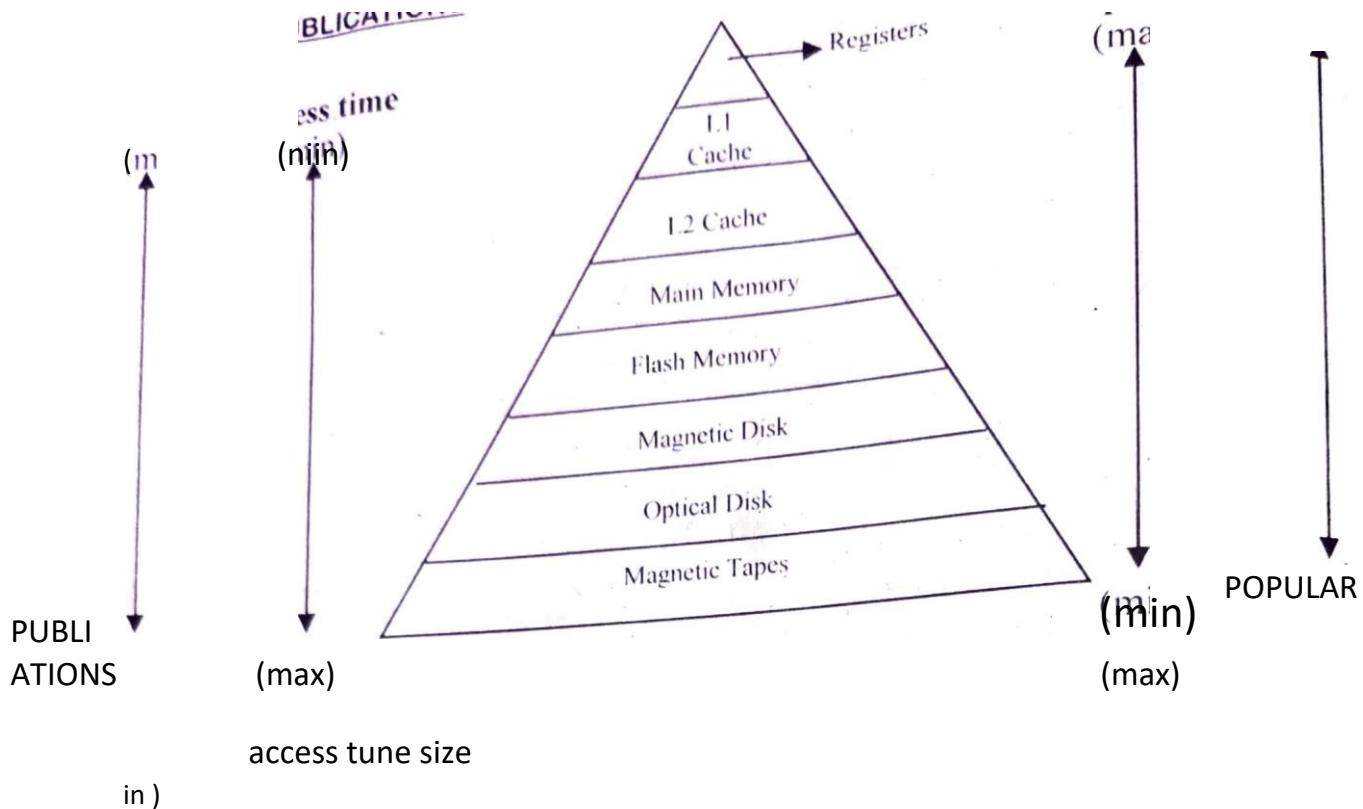
[WBUT 2005, 2015, 20181

Answer:

Based On the need of the hierarchical Inemory organization, the different Inennories are arranged in the form of a pyramid which gives a clear vision of the existing relationships between the dilTerent factors (like cost, speed, storage capacity, access time etc.) annong

the top different of which memories are the registers in the hierarchical while at the pyralnid. bottom The lie the menum•y nnagnetic hierarchy tapes, pyramid, is as shownon

speed cost



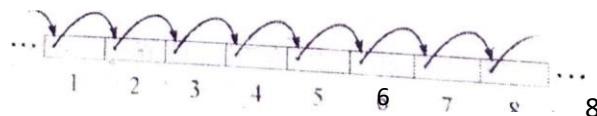
(max) On a.x)

The pyramid shows the different memories in a memory hierarchy. The relationship cost, speed and capacity are as follows: Cost Of the memories decreases downwards the pyramid with the registers being the costliest and the magnetic tapes the cheapest. Speed decreases downwards the pyramid with the registers being the fastest and the magnetic tapes the slowest. Capacity (size) increases downwards with the registers being the smallest and the magnetic tapes having the highest capacity. Apart from these, access time (inversely proportional to speed) increases downwards with the registers having the minimum access time and the magnetic tapes the maximum. A Read Only Memory can be used as a virtual Random Access Memory.

A ROM cannot work as a RAM. Although, it is possible to write data in ROMs e.g. EEPROM, Flash ROM etc., they are not RAMs (which is a volatile memory').

RAM or random access memory refers to the method by which the data is accessed from memory. Random Access means data can be accessed from memory in a random manner, i.e. in any order irrespective of the order in which we are storing the data. The opposite term for RAM is Sequential access memory or SAM . i.e. data can be accessed from that kind in a sequential only. Tape drives are classic examples of SAM.

Sequential access



Random access

In Read only Memory or ROM data is accessed in sequential way manner. RAM is much

faster but expensive also. data can be well as written in RAM which is not possible for ROM as it can be read only. So in both the a ROM can not be RAM. (CAM) is a special type of computer memory used in high speed searching applications. It is also known as associative memory.

content_addressed is accessed or by associative its content in memory (as opposed to an explicit organization address). Thus, in which reference the

clues are found. Production actual systems memory are obvious contents examples until a of desirable systems that noach employ (or set such of m
Associative stands as the most likely model for cognitive memories, a memory retrieve information best when it can be linked to other related

This linking is fast, direct and labyrinthine in the sense that the memory map is -to-many and homomorphic.

content-addressable a network switch memory receives is often a data used frame in computer from one network'ing of its ports, devices. it updates For an example,internal

table up with destination the frame's MAC source address MAC in the address table and to determine the port it what was port received the frame on. It needs then lcooksto be the ed to, and sends it out on that port. The MAC address table is usually implemented with a binary CAM so the destination port can be found very quickly, reducing the switch's latency.

3. Give examples of non-destructive read out memory and destructive read out memory. [WBUT 20051 OR,

Explain destructive read out & non-destructive read out of memory system.

[WBUT 20101

Answer:

Example of non-destructive read-out memory: Static Semiconductor Inemories like Static Random Access Memory (SRAM).Static random access metnory (SRAM) is a type of semiconductor menwry where the word static indicates that, unlike dynamic RAM (DRAM), it does not need to be periodically refreshed, as SRAM uses bistable latching circuitry to store each bit. SRAM exhibits data remanance, but is still volatile in the conventional sense that data is eventually lost when the memory is not powered.

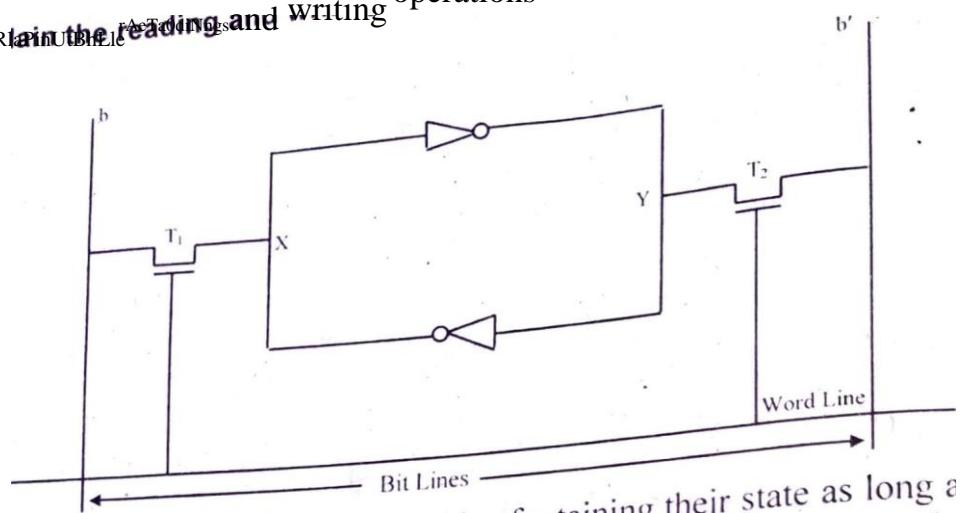
Example of destructive read-out memory: These are Inagnetic-core memories or ferritecore memories.

The term "core" comes from conventional transformers whose windings surround a magnetic core. In core Illetnory the wires pass once through any given core—they are Single-turn devices. The magnetic material for a core memory requires a high degree of magnetic remanance, the ability to stay highly magnetized, and a low coactivity so that less energy is required to change the magnetization direction. Magnetic-core mennory was the Predominant form of random-access cotnputer Inetnory for 20 years. It uses tiny magnetic toroids (rings), the cores, through which wires are threaded to write and read Information. Each core represents one bit of information.

Ansuer:

Explain the reading and writing operations of a basic static MOScell. [WBUT 2007]

4P.



Memories that consists of circuits that capable of retaining their state as long as power is applied are known as static memories. Figure shows a static MOS cell. Two inveners are cross connected to form a latch. The latch is connected to two bit lines by transistors T₁ and T₂. These transistors act as switches that can be opened or closed under control of the word line. When the word line is at the ground level, the transistors are turned off, and the latch retains its state.

Read Operation:

In order to read the state of the static MOS cell, the word line is activated to close switches T₁ and T₂. If the cell is in state 1, the signal on bit line b is high and the signal in bit line b' is low. The opposite is true-if the cell is in state 0. Thus b and b' are complements of each other. Sense/write circuits at the end of the bit lines monitor the state of b and b' and set the output accordingly.

Write Operation:

The state of the cell is set by placing the appropriate value on bit line b and its complement on b', and then activating the word line. This forces the cell into the corresponding state. The required signals on the lines are generated by the Sense/Write

- b) How many 128 x 16 RAM chips are needed to construct a memory capacity of 4096 words (16 bit is one word)? How many lines of the address bus must be used to access a memory of 4096 words? For chip select, how many lines must be

[WBUT 2007]

As, 4096 = 2¹², hence 12 lines address bus should be used.

For chip select, 6 lines in the horizontal and 6

vertical direction should be

5 a) classify memory system in a digital computer according to their use•

[WBUT 2007]

Answe
...llawin are the types of nnennory•,

1) Cl'tJ registers:

The high speed registers in the CPU serve as the working memory for temporary storage instructions and data. They usually forin a general purpose register file for storing data as it is processed. A capacity of 32 data word is typical of a register file and each register can be accessed that is, read from or written into, within a single clock cycle.

2) Main (pirnary) memory:

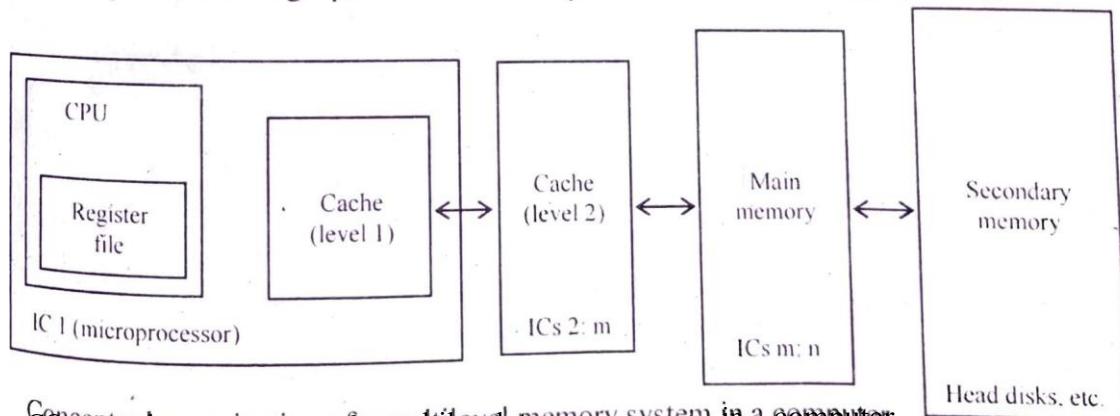
This large, fairly fast external memory stores programs and data that are in active use storage locations in main tnenmory are addressed directly by the CPU's load and store instructions. Main mennory capacity is typically between 1 and 210 MB 230 bytes is referred to as a gigabyte (1 GB). Access times of five or more clock cycles usual.

3) Secondary memory:

This memory type is much larger in capacity but also much slower than main meniory. secondary memory stores system programs. Large data files and the like that are not continuously required by the CPU. It also acts as an overflow memory when the capacity of main memory is excused. Information in secondary storage is considered to be an on line but it is accessed indirectly via input/output programs that transfer information between main and secondary Inemory. Representative technologies for secondary Illemory are magnetic hard disks and CD-ROMs, both of which have relatively slow electromechanical access times are measured in milliseconds.

4) Cache: Most computers now have another level of IC memory-sometimes several such levels called cache noemory, which in positioned logically between the CPU registers and main memory. A caches storage capacity is less than that of main memory, but with an access tilne of one to three cycles, the cache is much faster than main Illemory

ory because some of all of it can reside on the same IC as the CPU. Caches are esse ntial components of high-performance computers that aim to make CPI ≤ 1 .



Conceptual organization of a multilevel memory system in a computer
because sonne

in

- b) A random access memory module of capacity 2048 bytes is to be used COnputer and mapped between 2000 H and 27FF H. Explain with the help of a diagram the address decoding schema assuming 16 bit address bUS• 200} }

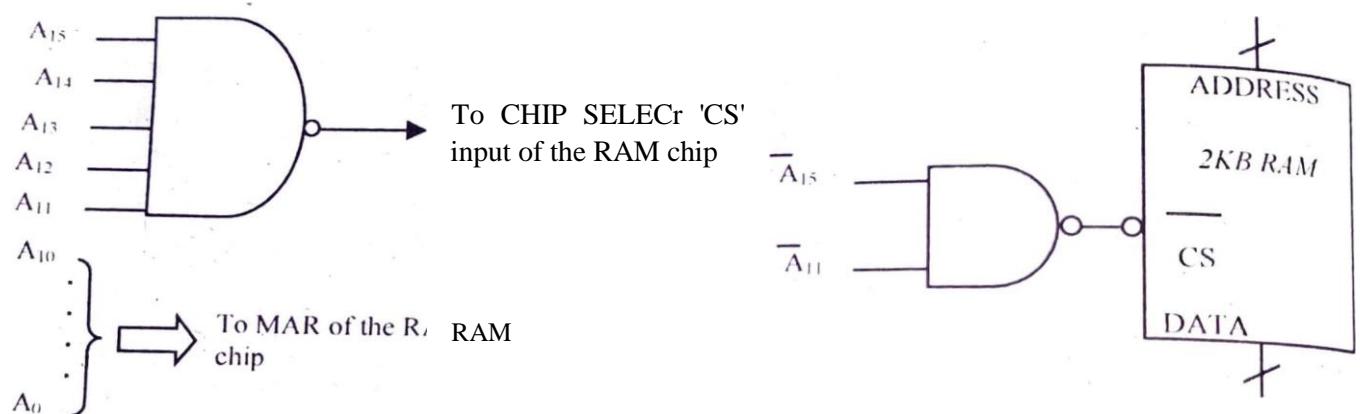
RANI capacity = 2048 bytes 21 bytes.

Thus the RAM has an I I-bit internal decoder which decodes the RAM address O through 211— I (i.e. 2047).

It is desired to select this RAM chip (using its CS input) with an external decoder such that the RAM occupies the 21 addresses, 2000H through 27FFH.

Clearly, the highest 5 bits of the address, nannely, A 1 5, A 14. A1-3- A12, All should be chosen equal to the base address 00100B, so that the 2K byte RAM chip occupies the desired addresses (i.e. 2000M through 27FFH) in the 64 bytes.

Diagram:



- c) How do the following influence the performance of a virtual memory system?

Size of page

- b) Replacement policy [WBUT 2007] Answer:

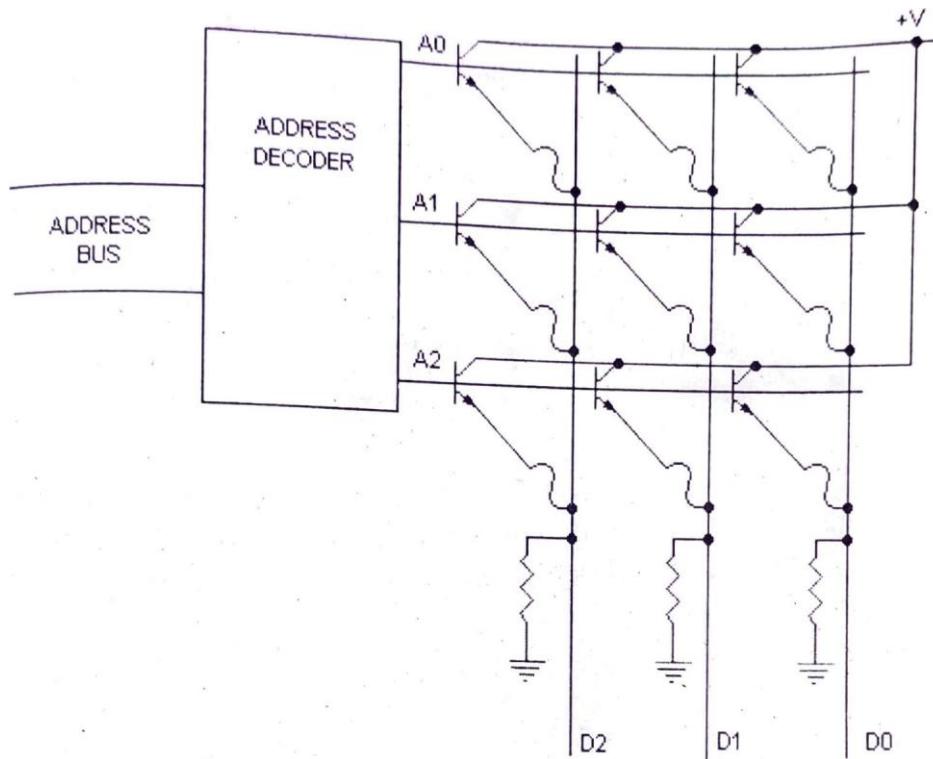
(a) Size of a page: If the size of a page is large then less number of page faults will occur. But this will lead to increase in the page transfer time.

(h) Replacement policy: The least recently used frame (i.e. the main memory block k) or the least frequently referenced frame must be replaced.

Because if a frame that is the most recently used or the most frequently used, is replaced then it may lead to more faults.

6. a) Draw the internal cell diagram of PROM and explain its functionality•

[WBUT 2008]



Programmable Read Only Memory (PROM)

A programmable read-only memory (PROM) or field programmable read-only memory (FEPROM) or one-time programmable non-volatile memory (OTP NV M) is a type of digital memory where the setting of each bit is locked by a fuse or antifuse. Such PROMs are used to store programs permanently. The key from a standard ROM is that the programming is applied after the device is constructed.

A typical PROM comes with all bits reading as 1. Burning a fuse during programming causes the bit to read as 0. The memory can be programmed just once after manufacturing by "blowing" the fuses, which is an irreversible process. Blowing a fuse opens a connection while programming an antifuse closes a connection (hence the name). The bit cell is programmed by applying a high-voltage pulse not encountered during normal operation across the gate and substrate of the thin oxide transistor (around 6V for a 2nm thick oxide, or 30MV/cm) to break down the oxide between gate and substrate.

The Positive voltage on the transistor's gate forms an inversion channel in the substrate below the gate, causing a tunneling current to flow through the oxide. The current produces additional traps in the oxide, increasing the current through the oxide and ultimately melting the oxide and forming a conductive channel from gate to substrate.

The Current required to form the conductive channel is around 10nm^2 and the breakdown occurs in approximately or less.

b) What is cache memory? How does it increase the performance of a COnPuten

[WBUT
20091

What is locality of reference? Explain the concept of cache memory with it.

OR,

[WBUT 20121

What is Cache memory? Why is it needed?

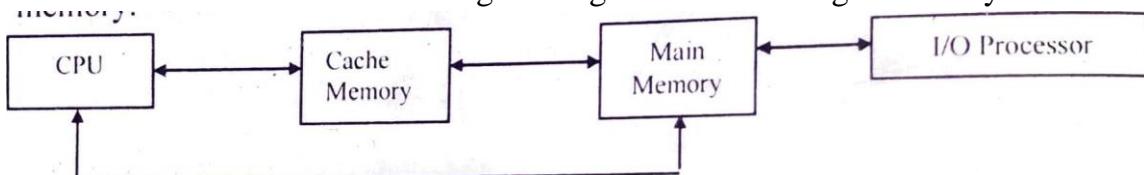
[WBUT 2013)

What is cache memory? What do you mean by hit ratio 75%?

Answer:

Locality of reference is the property that shows for large number of programs, references to memory at any given interval of time tend to be confined within a few localized areas in memory.

Cache memory is access memory (RAM) that a computer microprocessor can access more quickly than it can access regular RAM. As the microprocessor processes data, it looks first in the cache memory and if it finds the data there (from a previous reading of data), it does not have to do the more time-consuming reading of data from larger memory.



A special very high-speed memory called a cache is sometimes used to increase the speed of processing by taking current programs and data available to the CPU at a rapid rate. The cache memory is employed in computer systems to compensate for the speed differential between main memory access time and processor logic. CPU logic is usually faster than main memory access time, with the result that processing speed is limited primarily by the speed of main memory. A technique used to compensate for the mismatch in operating speeds is to employ an extremely fast, small cache between the CPU and the main memory whose access time is close to processor logic clock cycle time. The cache is used for storing segments of programs currently being executed in the CPU and temporary data frequently needed in the present calculations. By making programs and data available at a rapid rate, it is possible to increase the performance rate of the computer.

Cache memory is the fastest, smallest but the most expensive among the all devices. It stores the program segments and data that are needed frequently by CPU in current executions.

POPULAR PUBLICATIONS

Hit ratio: The performance of memory is frequently measured in terms of quantity is called hit ratio when the CPU needs to find the word in the cache, if the word is found ill the cache then its produces a hit if the word is not found in the cache, it is in main

memory as counted miss the ratio of number of hits is divided by the total CPU reference of memory is called hit ratio.
 hit ratio = number of hits / (number of hits + number of misses)
 So if hit ratio is 75% then 75% of the total times the CPU accesses memory is hit.
 For example if CPU accesses memory for 12 times then CPU will get the desired data for 9 times and won't find it 3 times.

- A three level memory system having cache access time of 5 nsec and disk access time of 40 nsec, has a cache hit ratio of 0.96 and main memory hit ratio of 0.9. What should be the main memory access time to achieve an overall access time of 16 nsec? [WBUT 2008, 2012] ill's cr:

Obviously, there is some mistake in the problem. A disk can never have a 40 nsec access time. However, let us first solve the problem with the given data.

Let the main memory access time be t nsec.

Considering a particular memory access request, the probability that it has a hit in the cache is 0.96, a hit in the main memory is $(1 - 0.96) * 0.9 = 0.036$ and, finally, a hit in the disk is $(1 - 0.96 - 0.036) 0.004$. Note that the sum of the probabilities is 1.

$$\text{Hence, } 16 = 0.96 * 5 + 0.036t + 0.004 * 40.$$

$$\text{So, } t = 306.6 \text{ nsec}$$

Clearly, this is impossible because in the memory hierarchy, the main memory occupies the middle level and not the end level. So 40 nsec must be the access time for the main memory and the access time for the disk has to be found out. This is done below.
 $16 = 0.96 * 5 + 0.036 * 40 + 0.004t$

$$\text{Hence } t = 2440 \text{ nsec}$$

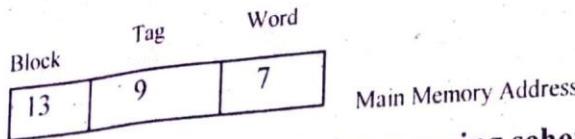
7. Given the following, determine the size of the sub-fields in the address for direct mapping, associative mapping and set-associative mapping cache schemes: [WBUT 2008]

Main memory size	512 MB
Cache memory size	
Address space of processor	512 MB
Block size	128 B
8 blocks in cache set.	

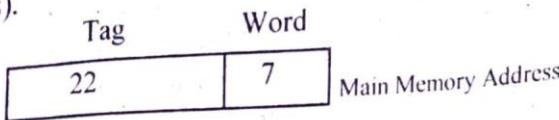
Answer:

As the size of the main memory is 512 MB hence there are 29 bits (as $512 = 2^9$ and 1 MB $= 2^{20}$ bits; hence $512 \text{ MB} - 2^9 \times 2^{20} = 2^9$) in the main memory address i.e. the address size of main is 29 bits.

Size of the sub-fields for direct mapping scheme:
Now size of cache memory is 1 MB = 2²⁰ bytes.
Hence number of cache blocks = 2²⁰/2²⁰ = 1.



Size of the sub-fields for associative mapping scheme:
Each block size is 128 bytes or 2^7 bytes. Hence number of main memory blocks = $2^{29}/2^{22} = 2^7$. So number of bits in the tag field is 22 and that in the word is 128 bytes).



Size of the sub-fields for set-associative mapping scheme:
In this case, there are 8 blocks per cache set and the total numbers of sets in the cache memory is $2^{13}/8 = 2^{13}/2^3 = 2^{10}$.

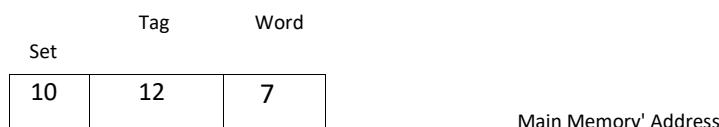
number of bits in the block field = $2^{10}/27 = 2^3$. So number of bits in the block field = 13.

Now Out Of the total main memory address size of 29 bits, word field contains 7 bits (as block size is 128B) and the block field contains 13 bits. Hence, the number of bits in the tag field is $29 - (13 + 7)$ = 9.

field is 7 (as block

In this case, there are 8 blocks per cache set and the total numbers of cache blocks are in the cache memory is

Hence, number of bits in the set 7 field = 12 and that in the word field is 7. So number of bits in the tag field = $29 - (12 + 7)$ = 10.



8. a) Can a ROM be also a RAM? Justify your answer. [WBUT 2009, 2017]
b) Explain the memory hierarchy pyramid, also explain the relationship of cost,

POPULAR PUBLICATIONS

speed and capacity. [WBUT 2009, 2013] Answer:

Refer to Question No. 2 of Long Answer Type Questions.

c) A hierarchical cache-main memory subsystem has the following specification:

i) Cache access item of 160ns ii)

Main memory access time 960n iii)

Hit ratio of cache memory is 0.9

Calculate the following:

a) Average access time of the memory system

b) Efficiency of the memory system.

Answer:

[WBUT 2009]

a) Given: Hit ratio (h) = 0.9, cache memory access time (t_{cache})

access time (t_{avg}) = 960 ns = 160 ns, main memory

Now, to access a word, the average required access time (t

$t_{avg} = h \cdot t_{cache} + (1 - h) \cdot t_{main\ memory}$ is given by: So $t_{avg} = 0.9 \times 160 + 0.1 \times 240 = 240$ ns.

1.) Avg. ifa access cache time nnetnot•y presence is not

Then ..
 $t_{\text{average}} = 0 \times t_{\text{cache}} + (1-h) \times t_{\text{main}} = t_{\text{main}}$
 of there, cache
 nnemory 240 ns.
 $\therefore \text{system efficiency} = \left(\frac{960 - 240}{960} \right) \times h$

= 960ns

$$\times 100\% = \left(\frac{720}{960} \right) \times 100\% = 75\%$$

- S) stelli efficiency 75%

9. a) State L1 and L2 cache policies with suitable figure.

[WBUT 2009]

Answer:

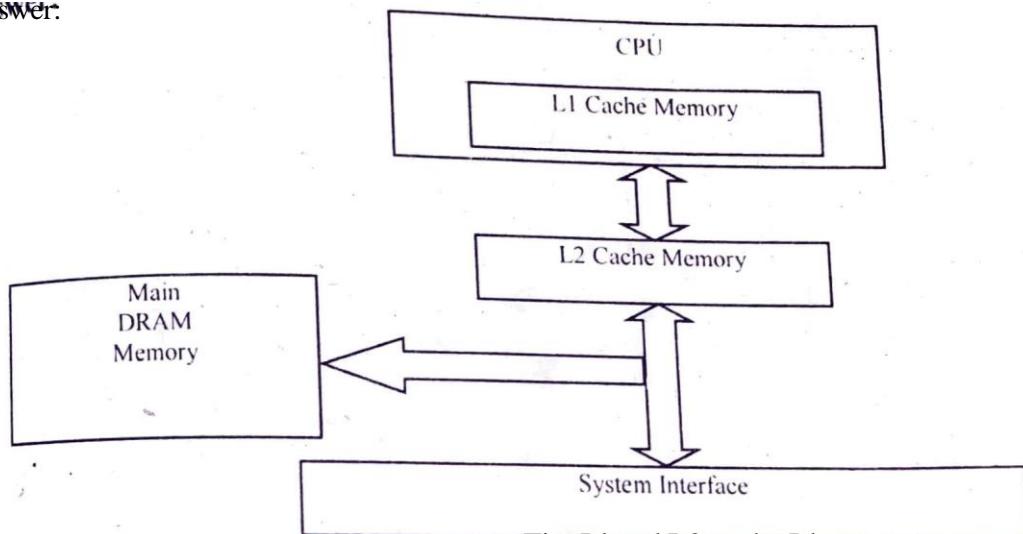


Fig: L1 and L2 cache

L1 Cache:

L1 cache lies on the same chip as the processor. It is smaller in size and faster. It has lower hit rate and higher number of misses as size is small.

L2 Cache:

L2 cache lies external to the processor chip. It is larger in size and comparatively slower. It has higher hit rate and low number of misses as size is comparatively large.

b) How many ~~350~~ 512x4 RAM chips are needed to provide a memory capacity of ~~2048~~ 8 bytes? [WBUT 2009] Answer:

Answer:
 The number of RAM chips needed are: $2048 \text{ bytes} / 256 \times 4 = 16 \text{ chips}$.

10a) If a CPU has 8-bit data bus & 16-bit address bus draw the connection diagram for this CPU with four 256x8 RAM & one 512x8 ROM. [WBUT 2010, 2018]

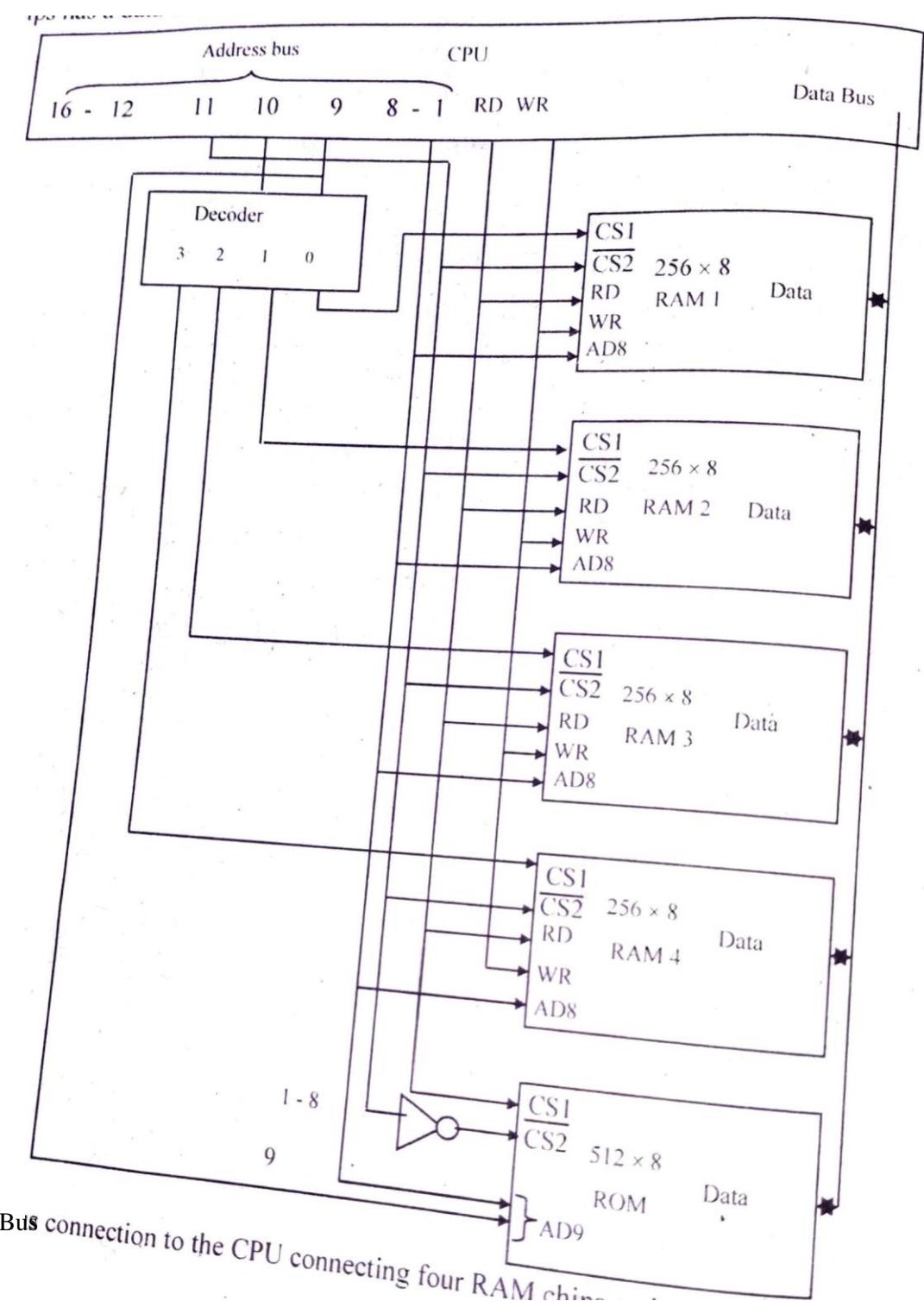
b) Show the bus connection with a CPU to connect four RAM chips bits each and a ROM chip of 512 x 8 bit size. Assume the CPU has and 16-bit address bus. Clearly specify generation of chip select sig

- b) Show the bus connection with a CPU to connect four RAM chips of size 256×8 bits each and a ROM chip of 512×8 bit size. Assume the CPU has 8-bit data bus and 16-bit address bus. Clearly specify generation of chip select signals.

[WBUT 2010, 2013, 2014]

Answer:

The figure below shows the required bus connection between CPU and four 256 RAM chips and one 512 x 8 bit ROM chip. The decoder has four bits. The IRA IV have 8 bit address bus each whereas the ROM chip has 9 bit address bus. Each of] chips has a data bus of 8 bits.



Bus connection to the CPU connecting four RAM chips

CSI and CS2 are select lines.

C0-88

11. a) Explain the difference between associative & set associative cache mapping technique. b) With the help Of following information, determine the size of sub-fields (in bits) the address for Direct mapping. Associative mapping & Set associative in mapping:

512 MB main memory & 2 MB cache memory

The address space of this processor is 256 MB

- The block size is 256 bytes

- There are 16 blocks in a cache set. [WBUT 201 OJ Ans" cr:

a) The difference between associative & set associative cache mapping technique:

Associative Mapping:

In associative cache mapping. the data from any location in RAM can be stored in

- any location in cache.

When the processor wants an address, all tag fields in the cache as checked to

- determine if the data is already in the cache.

- Each tag line requires circuitry to compare the desired address with the tag field.

- All tag fields are checked in parallel.

- The lower $\log_2(\text{line size})$ bits define which byte in the block .The renuining upper bits are the tag field.

- For a 4 GB address space with 128 KB cache and 32 byte blocks:

Tag (27bits)	Offset (5bits)
--------------	----------------

Set Associative Mapping:

- Set associative mapping is a mixture of direct and associative mapping.
- The cache lines are grouped into sets.
- The number of lines in a set can vary from 2 to 16.
- A portion of the address is used to specify which set will hold an address.
- The data can be stored in any of the lines in the set.

Tag (19bits)	Set (7 bits)	Offset (6 bits)
--------------	--------------	-----------------

- b) As the size of the main memory is 512 MB hence there are 29 bits (as $512 = 2^9$ and 1 MB = 2²⁰ bits; hence $512 \text{ MB} = 2^9 \times 2^{20} = 2^{29}$) in the main memory address i.e. the address size of main memory is 29 bits.

Size of the sub-fields for direct mapping scheme:

Now size of cache memory is 2 MB = 2²¹ bytes. Hence number of cache memory blocks = $2^{21} / 2^8 = 2^{13}$. So number of bits in the block field = 13.

Now out of the total main memory address size of 29 bits, word field contains 8 bits (as block size is 256B) and the block field contains 13 bits. Hence, the number of bits in the tag field is $29 - (13 + 8) = 8$.

Block	Tag	Word
13	8	8

Main Memory Address

Size of the sub-fields for associative mapping scheme:

Size of the sub-fields for associative mapping scheme:
Each block size is 256 bytes or 2^8 bytes. Hence number of main memory blocks = $2^{29} / 2^8 = 2^{21}$. So number of bits in the tag field is 21 and that in the word field is 8 (as block size is 256 bytes).

Tag	Word
21	8

Main Memory Address

Size of the sub-fields for set-associative mapping scheme:

In this case, there are 8 blocks per cache set and the total numbers of cache blocks are 213. So number of sets in the cache memory is $213 / 8 = 213 / 2^3 = 2^7$

Hence, number of bits in the set field is 10 and that in the word field is 8. so number bits in the tag field = $29 - (10 + 8) = 12$.

Set	Tag	Word

Main Memory Address

12. What is Belady's anomaly for page replacement technique? [WBUT Explain 2010] With example.

Answer:

Belady's anomaly, introduced in 1969, is a very common concept discussed in context of page faults. It proved that while dealing with page faults, it is possible to have more page faults when increasing the number of page frames if a first-in first-out (FIFO) method of frame management is used. The explanation is as follows:

In a computer memory, information is loaded in the main memory in the form of pages, which are specific sized storage chunks. It is possible to load only a limited number of pages at a time in the memory. For each page to be loaded, an equal sized frame is needed in the main

POPULAR PUBLICATIONS

memory. If a required page is not found in the main memory, a page fault occurs and that page is brought from the disk or secondary memory. It might happen that there is no free or empty frame in the main memory at the time of occurrence of the page

fault to accommodate the new page in the main memory. In such instances, it is required to free a frame to accommodate the new page. Before the introduction of Belad's anomaly, it was acceptable that the common page replacement algorithm producing acceptable results was the FIFO one. But, the anomaly proved that wrong.

Page Requests

Newest Page	3	1	-	4	3	2	1	0	4
	2	2	0	3	-	4	4	-	1
	3			2			4		0
Oldest Page	3				2	2	4		
		2		-		3	-	2	
			0	3		-		4	1

Page Requests

Page Requests	3	2	1	0	~	4						
Newest Page	3	2		0	~	4	~	1	0	4		
Oldest Page	3	2		0	0	4	2	1	0	4	0	0

COMI'tJl+R

c,fltnple Of Belndy's anotnnly: Using three page frames. 9 page faults occur.
Incß,asing to four page fraihes caukses **page faults** to occur. page faults are in bold

- a) A computer has a main memory of 64K 16 and a cache memory of 1K words. The cache many uses bits are direct there mapping in tag, with index, a block block size and word of 4 words,fields?

• How

What is the size of one cache word?

- How many blocks can be accommodated in the cache? (WBUT 2013) OR,

A digital computer has a memory unit of 64K x 16 and a cache memory of 1K words. The cache uses direct mapping with a block size of 4 words. How many bits are there in the tag, index, block and words fields of the address format. Also calculate address format for associative mapping and for 4-way set associative mapping. [WBUT 20151 Answer:

Direct Mapping

- 16 bit addresses (can address 64K words menoory)
- 4 words block (offset is 2 bits)
- 1K words of cache
- Nunlber of lines = $1\text{ K} / 2^2 = 2^8$
- Bits to specify which line = $\log_2(2^8) = 8$

6bits	8bits	2bits
Tag	Line/Index	Offset

In v\ord field there is 16 bits.

In this case, there are 4 blocks per cache set and the total numbers of cache blocks are $4=2^2$ So number of sets in the cache memory is $2^8/4 = 2^8/2^2 = 2^6$.

Hence. number of bits in the set field is 6 and that in the word field is 2. So nutnber of bit^q in the tag field = $16 - (6 + 2) = 8$.

8bits	6bits	2bits
Tag	Set	Offset

b) A processor's TLB has a hit ratio of 80% and it takes 20 ns to search the TLB and 100 ns to access main memory. What will be the effective access time?

[WBUT2013]

Answer:

$$\text{Effective access time} = 20 \text{ ns} + (100 \times 20\%) \text{ ns} = 40 \text{ ns}$$

14' a) Write down difference between Dynamic RAM and Static RAM. (WBUT 20141 Answer:

Dynamic RAM is the most common type of memory in use today. Inside a dynamic RAM chip, each memory cell holds one bit of information and is made up of two parts: a transistor and a capacitor. These are, of course, extremely small transistors and capacitors so that millions of them can fit on a single memory chip. The capacitor holds

the bit of information - a 0 or a 1. The transistor acts as a switch that lets the control circuitry on the memory chip read the capacitor or change its state-

in the

A capacitor is like a small bucket that is able to store electrons. To Store a I in the memory cell, the bucket is filled with electrons. To store a 0, it is emptied. The problem with the capacitor's bucket is that it has a leak. In a matter of a few milliseconds a bucket becomes empty. Therefore, for dynamic memory to work either the CPU or the memory controller has to come along and recharge all of the capacitors holding a I before they discharge. To do this, the memory controller reads the memory and then writes it right back. This refresh operation happens automatically thousands of times second.

This refresh operation is where dynamic RAM gets its name. Dynamic RAM has to be dynamically refreshed all of the time or it forgets what it is holding. The downside of all of this refreshing is that it takes time and slows down the memory.

Static RAM uses a completely different technology. In static RAM, a form of flip-flop holds each bit of memory (see How Boolean Gates Work for detail on flip-flops). A flip-flop for a memory cell takes 4 or 6 transistors along with some wiring, but never has to be refreshed. This makes static RAM significantly faster than dynamic RAM. However because it has more parts, a static memory cell takes a lot more space on a chip than a dynamic memory cell. Therefore you get less memory per chip, and that makes static RAM a lot more expensive.

So static RAM is fast and expensive, and dynamic RAM is less expensive and slower. Therefore static RAM is used to create the CPU's speed-sensitive cache, while dynamic RAM forms the larger system RAM space.

- b) A disk pack has 20 surfaces. Storage area on each surface has an inner diameter of 20 cm and outer diameter of 30 cm. Maximum storage density on any track is 2000 bits/cm and minimum spacing between tracks is 0.50 mm.

[WBUT 2014, 2017]

Answer:

Given, no. of surfaces = 20

Inner track diameter = 20cm

Outer track diameter = 30cm

So, total track width = $(30 - 20)/2 = 5\text{cm}$

Track separation = 0.50mm

Thus, no. of tracks/surface = $(5 * 10)/0.50 = 100$

Minimum track circumference = $20 * 3.14 = 62.8\text{cm}$

Maximum track storage density = 2000 bits/cm, which will be on innermost track. So Data storage capacity/track = $62.8 * 2000 \text{ bits}$

POPULAR PUBLICATIONS

Disk speed = 3600 rpm 125.6 Kbits

So, rotation time $1/3600$ minute = 16.67 msec

i) storage capacity = 100×125.6 Kbits = 251.2Mbits = 31.4 Mby^{tes} ii) Data transfer rate = 125.6 Kbits/16.67msec = 7.53 Mbits/sec

This is the maximum data transfer rate excluding seek time and rotational latency.

COMPUTER .a) A cache has 64 KB capacity, 128-Byte lines and is 4-way Set-Associative. 15. CPU generates 32-bit address for accessing data in the memory• The(i) How many Lines and Sets does the cache have?

(ii) How many entries are required for tag?

(iii) How many bits of tags are required in each entry in the tag array?

b) A hierarchical Cache-Main Memory subsystem has the following specifications:

(i) Cache Memory Access Time 80 ns (ii)

Main Memory Access Time 150 ns (iii)

Hit ratio of Cache Memory is 0.9.

Calculate the following:

(a) Average access time of the memory system

(b) Efficiency of the memory system

(c) Define addressing mode. [CWBUT 2014] Ans" er:

a) (i) No. of lines = $64K B / 128\text{bytes} = 512$ So No. of

tag entries = 512

No. = 128

(ii) Since one tag entry is required for each line, the tag array needs 512 entries

(iii) As there are 128 sets, 7 bits are needed to select the set.

As the line size is 128 bytes other 7 bits are needed for selecting word in a line. So Tag bits = $32 - 7 - 7 = 18$

b) (a) Given: Hit ratio (h) = 0.9, cache memory access time (t_{cache}) = 80 ns, main memory access time (t_{main}) = 150 ns

Now, to access a word, the average required access time (tmerage) is given by: $t_{\text{merage}} = h.tcache + (I \cdot n \cdot u)$

$$\text{So } t = 0.9 \times 80 + 0.1 \times 150 = 87 \text{ IIS.}$$

(b) Avg. access time presence of cache memory 87 ns.

Now if a cache memory is not there,

$$\text{Then } h = 0, \therefore t_{\text{average}} = Ox + (1 - h) \times t_{\text{nunn}}$$

$$= t_{\text{nunn}} - t_{\text{cun}}$$

150-87 63

$$\text{system efficiency} \quad \frac{63}{150} \times 100 \% / 0 \quad \% - 42 \% / 0$$

$$\therefore \text{System efficiency} = 42\%$$

(c) Addressing modes are an aspect of the instruction set architecture in most central Processing unit (CPU) designs. The various addressing modes that are defined in a given Instruction set architecture define how machine language instructions in that architecture identify the operand (or operands) of each instruction.

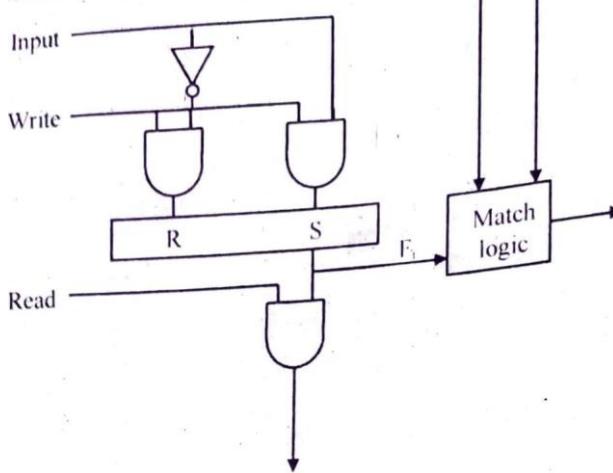
16. Draw

cell of one word
including the

Ans "cr. •

One cell of

Associative Memory



the logic diagram of the
in associative memory
read and write logic.

Associative Memory

To MI

Output

17. A two-way set associative cache memory uses blocks of four words. The cache can accommodate a total of 2048 words from the main memory. The main [WBUT memory 2016] size is 128Kx32.

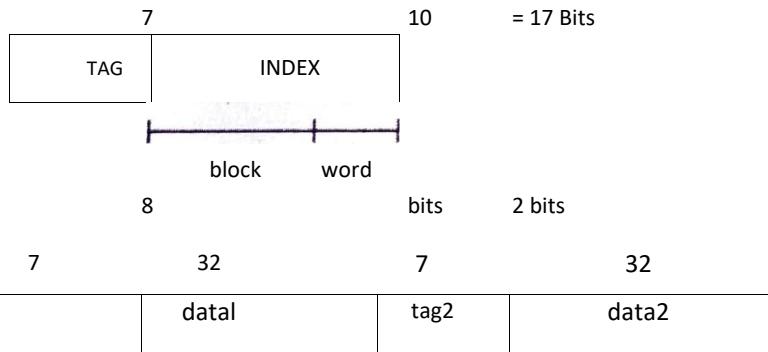
- How many bits are there in the tag, index block and word fields of the address format?

POPULAR PUBLICATIONS

ii) What is the size of the cache memory?

Answer:

$128 = (2 \text{ power } 7)$; for a set size of 2, the index address has 10 bits to accommodate $2048/2 = 1024$ words of cache.



$$\text{size of the cache is } s = 1024 * 2^{(7+32)} = 1024 * 78$$

18. A hierarchical Three-Level Memory (Cache, Main memory, Hard Disc) system

- i) Cache Memory Access Time is 10nsec
- ii) Disc Access Time is 150nsec
- iii) Hit ratio of Cache Memory is 0.97
- iv) Hit ratio of Main Memory is 0.9

What should be the Main Memory access time to achieve an Overall access time of

[WBUT 2016] co-

snswcr:

$$\text{Tetrhl} + (\lfloor 1 + 0.03 \cdot 4 \cdot t \rfloor) \cdot 0.9 - h \cdot t^2 \rfloor \cdot h^3 + 0.03 \cdot (t_i \cdot 0.4 \cdot t^2 \cdot 0.1 + * \cdot t^2 + 160) \\ + t^2 + .003(t^2 + 160)$$

10.3

$$103 - 0.27 + 0.027 t^2 + 0.003 + .48$$

= 318.33 nsec

19. a) Explain memory-hierarchy.

[WBUT 2018]

b) Can a Read Only Memory be also a Random Access Memory? Justify your answer. Answer:

a) Memory Hierarchy is an enhancement to organize the memory such that it can minimize the access time. The memory Hierarchy was developed based on a program known as locality of references. The figure clearly demonstrates the different levels of memory hierarchy.

This memory hierarchy design is divided into 2 main types:

- I. External Memory: Comprising of magnetic disk, optical disk, Magnetic Tape which are accessible by the processor via I/O Module.
2. Internal Memory: Comprising of main memory, cache memory and CPU registers. This is directly accessible by the processor.

The characteristics of the memory Hierarchy design are:

- i) Capacity: It is the global volume of information the memory can store.
- ii) Access Time: It is the time interval between the read/write request and the availability of the data.
- iii) Cost per bit: We move from bottom to top in the Hierarchy, the cost per bit increases

i.e.

~~MEMORY IS COSTLY THAN EXTERNAL MEMORY.~~

Internal memory is

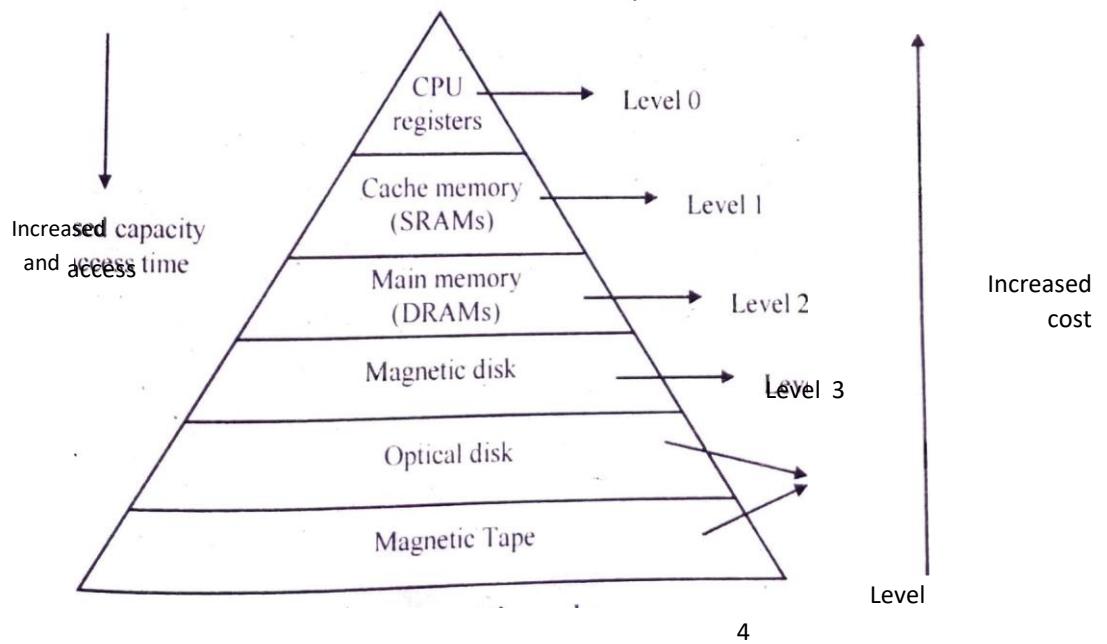


Fig: Memory Hierarchy

b) Read only memory is used for permanent data storage. access memory on the other hand, can only hold data temporarily. RAM is generally several megabytes while ROM is several gigabytes. storage. vs

20. Write short notes of the following:

[WBUT 20071]

a) Stack organization

[WBUT 20081]

b) Cache Replacement Policies

[WBUT 20091]

c) Virtual address to real address mapping

[WBUT 2011,
20171]

d) Static and dynamic memory

[WBUT 2012)

e) Paging

Answer:

a) Stack memory: Refer to Question No. 6 of Short Answer Type Questions.

b) Cache Replacement Policies:

The decision of the right block getting replaced is an important system performance factor. The objective is to keep the blocks in the cache that are likely to be referenced (needed by the CPU) in the near future. As per the property of 'locality of reference', it can be said that the blocks that have been referenced recently will be referenced again soon. So when a block is to be overwritten, it is needed to overwrite the one that has gone the longest time without being referenced. This block is called the least recently used (LRU) block and the technique is called the LRU replacement algorithm or policy.

While using the LRU algorithm, the cache controller must track references to all blocks as computation proceeds. A counter is used to track the references to each block. Depending on the number of references tracked by each counter (i.e. by the number of hits and misses) it is decided which block is used the longest time back (i.e. the least recently used block). And that block is then replaced. Though very effective LRU algorithms can lead to poor performance when accesses are made to sequential elements of an array that is slightly too large to fit into the cache.

Other replacement algorithms are however not as effective as LRU. Another algorithm that replaces the particular block in the cache set that has been in the cache for the longest period of time is called the first-in-first-out (FIFO) algorithm. The third algorithm that replaces the particular block in the cache set that has experienced the fewest references is called the least-frequently used (LFU) algorithm. Apart from these, using an algorithm that randomly chooses the blocks to be replaced, is also getting effective.

POPULAR PUBLICATIONS

c) Need for Mapping of Addresses:

To execute programs (instructions and data), they must be physically present in the main memory. If a virtual address issued by the CPU refers to a part of the program or data space that is currently in the physical memory, then those contents are accessed immediately in the main memory. Else, if the referenced address is not in the main memory, its contents must be brought into a suitable location in the main memory before they can be used.

COMPUTER

and Fraunes:

The physical size called (lilain) blocks Inetnory or /rtunes is broken while down logical (i.e. nocmory (virtual) space Inen)01'Y' is divided) is broken into down groups(i.e. space is divided) into grotIPS Of equal sizes called pages. Pages and frames must Ofequal sizes.

ed for Pages and Frantes:

di\visions are done to sinnplify the ilnplementation of memory table for address prograllls are considered to be split into pages. When a program is to be its pages frotn the auxiliary Inennory are loaded (mapped or translated) into any ailable Inetnory fratnes.

Mapping a Virtual Address to a Physical Address:

In the main memory, portions of programs and data may not be in contiguous locations and empty spaces (to bring programs from the disk) may be available in scattered locations in Inen10t•y. With help of mapping it is decided exactly where to place the news data to be brought in from the disk. The mapping hardware organization consists of a virtual address register that contains the virtual address provided, a memory mapping table that will keep track of the available frames and map them as needed to the different pages, a memory buffer register to hold the frame number where the page is to be mapped. a main memory address register that holds the physical address and the main memory. The figure below shows the hardware organization.

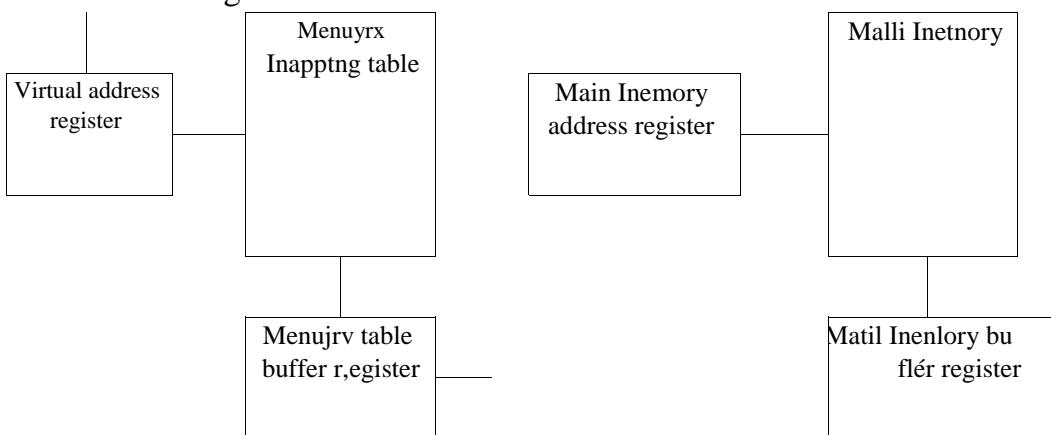


Fig: Memory Table for mapping a virtual address

The mapping from address space to memory space is facilitated if each virtual address is considered to be represented by two numbers: a page number address and a line within

the page. Suppose the address space capacity is 8K and the memory space capacity is 4K. Splitting each into groups of 1K words, 8 pages and 4 blocks/frames are obtained respectively. Now $8K = 2^{13}$ and $1K = 2^{10} = 1024$ words. So a virtual address has 13 bits (as $8K = 2^{13}$) and each page has 1024 words.

Let the high-order three bits of the virtual address specify one of the 8 pages and the low-order 10 bits give the line address within the page. The page number to block number mapping is required. As shown in figure below, the memory-page table consists of 8 words, one for each page. The virtual address has got two parts, the page number and

word gives the block number where that page is stored in main memory.

In the table pages 1, 2, 5, and 6 are shown to be available in main memory in blocks 3 0 1, and 2 respectively. The presence hit in each location, if 1, indicates that particular page has been transferred auxiliary noemory to main noemory and, if 0, indicates th particular page is not available in tmain noen)ory. The content Of the word in the memory pace table at the page nurnber address is read out into the nwmorY table buffer register. If the presence bit is a 1. the block number thus read is transferred to the two high-order bits of the Inain Internory address register. The line number from the virtual address is transferred into the 10 low-order bits of the memory address register (main memory address space has $4K = 2^{12}$ words i.e. 2 bit block number and 10 bit line number). A 'read' signal to Inain then transfers the content of the word to the main memory buffer register ready to be used by the CPU.

However, if the presence bit in the word read from the page table is 0, it signifies that the content of the word referenced by the virtual address does not reside in main memory Then a 'page fault' occurs if that page is needed in the Inain memory and it is needed to fetch that page from the disk to the main memory to resume further computation. Diagram:

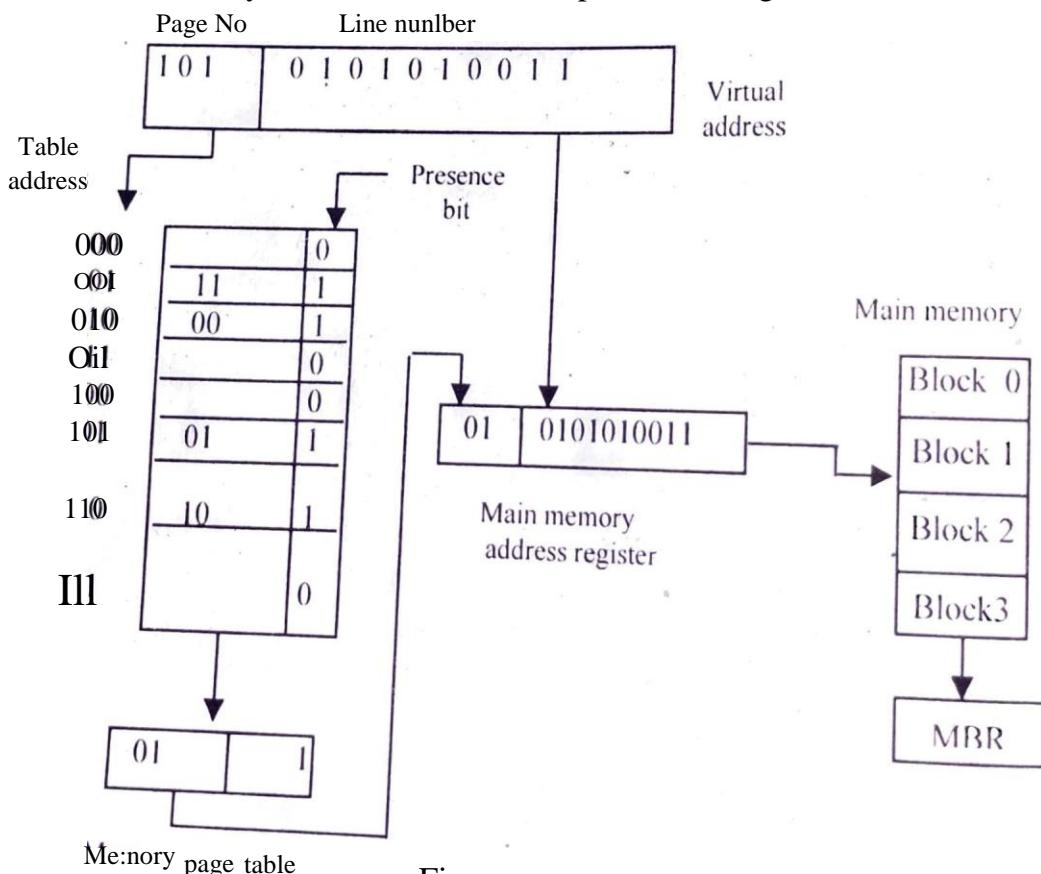


Fig: Menlory table in paged system

d) Static memory:

POPULAR PUBLICATIONS

It or flip-flops that store the binary information. stored information remains valid as long as power is applied to the {Init.

static memories, consisting of static RAM chips are capable of retaining their state as long as power is applied to them.

memories
COMPUTER

\ M loses its stored information in a very short time (a few milliseconds) unless

the supply is on.

is stored in a cell in the form of charge on a capacitor and can be maintained only for a very few milliseconds. As the charge on the leak away as a result of normal leakage, the capacitor gets turned off after the

milliseconds. So, to retain the cell information for a much longer time, the cell's must be periodically refreshed to restore the capacitor charge to its full value.

paging:

is a concept of transfer of pages between main memory and an auxiliary store, the hard disk. So, in paging, relatively inactive pages are removed from physical memory to make places for pages, which are needed by the memory for execution of an instruction. For example, nowadays all Windows OSs come with built-in paging files.

files are in megabytes, created during the Windows XP installation and reside on the hard drive. The actual size of the page file is based on how much RAM is installed in the computer. By default, XP creates a page file that is 1.5 times the amount of installed

RAM and places it on the hard drive where XP is installed.

Chapter at a Glance

one or multiple machine cycles.

- Instruction cycle: An Instruction Cycle consists of one or multiple machine cycles. ~~one or multiple machine cycles.~~ Example: A sequence of operations involved in processing an instruction constitutes instruction cycle. It has 2 major phases:

(i) Fetch Cycle: during this phase instruction is obtained from memory, the main memory fetching any required

(ii) Execution cycle: this phase includes decoding the instruction's operands, performing the operation specified by the instruction's opcode. Signals are generated by Hardwired control unit: In a hardwired control unit, a means of hardware using conventional single logic design techniques. Each step in the sequence of control signals is executed in one clock period.

The basic units of the hardwired control are: (a) A clock (b) A counter (c) A decoder (d) encoder.

Each micro program comprises of a sequence of micro operational steps. Each micro operational step consists of one or more micro operations and needs a number of control signals to be activated. The control signals needed to execute each micro operational step are generated simultaneously. So, in each clock state a micro operational step is performed i.e. in each clock state, the necessary control signals are generated & the corresponding micro operations are performed.

- Micro-programmed control unit: This is a control unit whose binary variable (i.e. the control functions that specifies a microoperation) remains stored in memory i.e. such type of control unit is software (i.e., microinstruction) based.

Control Words:

Control words are words whose bits are used to control certain specified microoperations or general operations. These are basically strings of 1's and 0's. Each of the bits in different control words generates different microoperations related to the instruction. Control words generally consist of

Routine: Meaningful sequence of instructions is called a routine (or a program).

Microroutine: Microinstructions are stored in control memory in groups. Each of these groups specifies a microroutine. So a meaningful sequence of microinstructions constitutes a microroutine. Now the microinstructions within a microroutine must be sequenced and there

must be options of branching from one routine to another.

Microinstruction: Microinstruction is an instruction

whose bits carry out a set of microoperations at the same time.

Toprogram: A meaningful sequence of microinstructions constitutes a top program.

co-loo

COMPUTER

Multi le Cho ice

e uestions

Micro-processor, the address Of
executed, is

a

[WBUT 2003, 2011]

- stored a) Stack in pointer
- b) Address hatch
- c) Program counter
- d) General purpose register

Answer: (c)

2 A UART is an example of

[WBUT 201

- a) serial asynchronous data transmission ship
- b) PIO
- c) DMA controller
- d) none of these

Answer: (a)

3. Control program memory can be reduced by

(WBUT 20101

- a) Horizontal formatcontrol unit
- b) Vertical format micro-program
- c) Hardwired
- d) None of these

Answer: (b)

4. The cylinder in a disk pack is

[WBUT2018]

- a) collection of all tracks in a surface
- b) logical view of same radius tracks on different surfaces of disks
- c) collection of all sectors in a track
- d) collection of all disks in the pack

Short Answer e uestions

1, What are the advantages of microprogramming control over hardwired control?

(WBUT 2008, 2011, 20141

It should be mentioned that most computers today are micro-programmed. The reason is basically one of flexibility Once the control unit of a hard-wired computer is designed and built, it is virtually impossible to alter its architecture and instruction set. In the case of a micro-programmed computer, however, we can change the computer's instruction set

simply by altering the micro-program stored in its control memory. In fact, taking our basic computer as an example, notice that its four-bit op-code permits up to 16 Instructions. Therefore, could add seven more instructions to the instruction set by simply expanding its micro-program. To do this with the hard-wired version of our computer would require a complete redesign of the controller circuit hardware.

Another advantage to using micro-programmed control is the fact that the task of designing the computer in the first place is simplified. The process of specifying the architecture and instruction set is now one of software (micro-programming) as opposed to hardware design. Nevertheless, for certain applications hard-wired computers are

POPULAR PUBLICATIONS

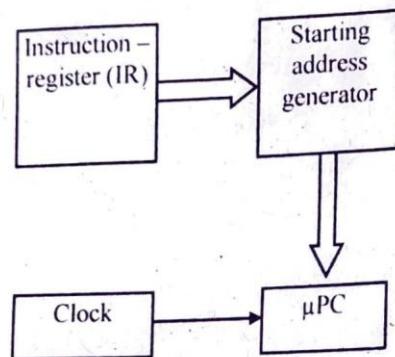
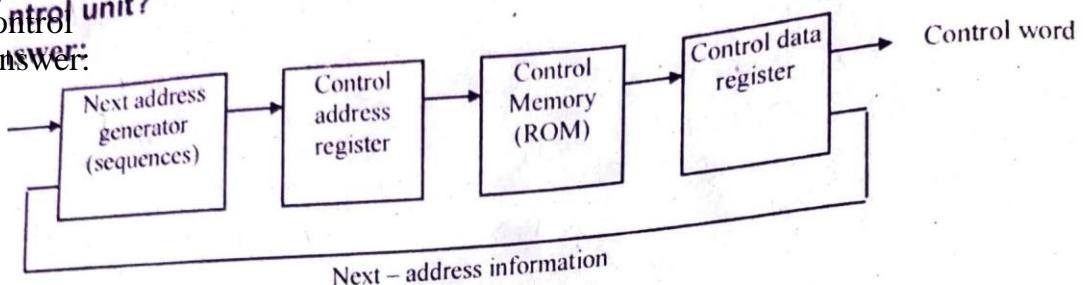
still
hard-wiring may be required since it is faster to have the "program" do it.

used. Hard-wiring is required since it is faster to provide the required control signals than to have a "program" do it.

Of micro-programmed control unit? Draw the block diagram and explain the functionality.

2. Draw the control unit?

Answer:



μPC: It is basically the CMAR
Control Store: It is the Control Memory.
 The CMDR is not shown here, as it is optional

Control Memory Address Data
Control word

Control Store: It is the Control Memory. The CMDR is not shown here, as it is optional

The parts of the microprogrammed control unit are Control Memory Address Register (CMAR), Next Address Generator(sequencer), Control Memory Data Register(CMDR) Control Memory.

Working of a Microprogrammed Control Unit:

Steps:

Instruction is fetched from the main memory and is stored in the IR.

Opcode portion of the IR, which holds the operation part of the instruction, is decoded with a decoder.

Decoding the opcode, gives the 1st address or the starting address of the microinstruction

in the control memory.

The starting address then comes to the 'next address generator'.

From there, it goes to the CMAR.

Then the control memory is accessed and the first microinstruction from the starting address location in the control memory is sent to CMDR.

The CMDR now holds the first microinstruction. Simultaneously, the CMDR sends next address information to the next address generator.

While asking for the next address information, the CMDR executes the present con

word stored in it.

on output of the microinstruction, the respective required control signal is generated. Meanwhile, the next address generator on getting 'next-address information' signal from the CMDR, generates the next location address of the next microinstruction in the memory & sent it to the CMAR.

COMPUTER ORGANISATION

This continues till the execution of the current microprogram (i.e., execution of one instruction) is over.

Once execution of one instruction is over, execution starts for the next instruction.

3 Write short note on Microinstruction. (WBUT 20121 Arms" ct•:

Each microinstruction defines a set of datapath control signals that must be asserted in a state.

Executing a microinstruction means, we assert the control signals specified

by that microinstruction.

Designing the control unit as a program composed of microinstructions is called

can choose the number of fields a microinstruction should have and which control signals should be affected by each field. In choosing the format:

- (a) simplify the representation

Ex: The mnemonics Add, Subt and Func can represent the function to be performed by ALU.

- (b) try to make it easier to write and understand microprogram.

Example: It is useful to have one field controlling the ALU, two fields to determine the two sources for the ALU, and one field to determine the destination of ALU result

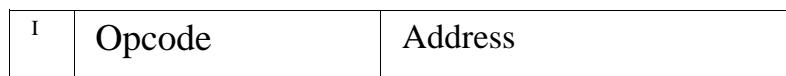
- (c) make it difficult to write inconsistent (if it requires a control signal to be set to two different values !) microinstructions.

Ex: From the three write signals RegWrite, MemWrite and IRWrite only one must be asserted in a given cycle. If the mnemonics of these three signals share the same microinstruction field, we can place only one mnemonic to that field, restricting these three signals to one at a time.

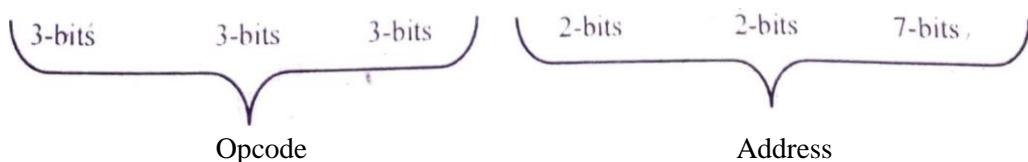
In selecting the microinstruction format for our MIPS subset multi clock cycle implementation, we can assume that signals that are never asserted simultaneously may share the same field. We can thus define the following 8 fields:

ALU Control	SRC1	SRC2	ALU Dest.	Memory	Memory Reo.	PCWrite Control	Sequencing
-------------	------	------	-----------	--------	-------------	-----------------	------------

Mapping from instruction code to microinstruction format: The format for an instruction is shown below:



Now the format for a microinstruction is also the same as the above instruction format:



			CD		AD
--	--	--	----	--	----

Fig: Microinstruction code format (20 bits)

POPULA PUBLICATIONS

4. What do you mean by instruction cycle. 20121

Answer:

Instruction Cycle:

An Instruction Cycle consists of one or multiple machine cycles •
Example: A sequence of operations involved in processing an instruction constitutes

instruction cysle. It has 2 major phases:

- (i) fetch cycle: during this phase instruction is obtained from the main memory.
- (ii) execution cycle: this phase includes decoding the instruction, fetching operands, performing the operation specified by the instruction's opcode.

5. Show the circuit diagram for implementing the following register transfer operation. If (ab- - - 1) then RI 4— R2 else RI e- R3, where a and b [WBUT are control20081 variables.

Answer:

The question (particularly the meaning of 'ab = 1') is not \ ery clear. It looks like circuit is a MUX which has 2 inputs, r2 and r3, and one output r l . If the MUX control input ab- is a 1, then content of r2 is transferred to rl and, otherwise, content of ls transferred to rl .

6. What do you mean by instruction cycle, machine cycles and T states?

[WBUT 2008,

2011 Answer:

Instruction Cycle: Refer to Question No. 4 of Short Answer Type Questions.

Machine C.rc/e. •

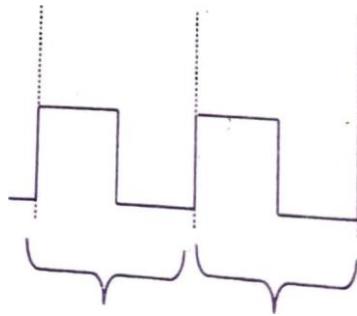
One or multiple clock cycles make a machine cycle.

Example: Generally one memory read (i.e. CPU placing address on address bus and the memory sending 'data' back to the CPU via the data bus) or one memory write (i.e. CPU placing 'data' to be written on the data bus, the address of the memory location where data is to be written on the address bus and then sending 'write' pulse on the control bus) constitute a machine cycle.

So, these sequences may occur within one clock cycle or may take multiple clock cycles. T states:

A T-State is one clock period.

A clock frequency = 1/clock period = 1/T.



CO-104

COMPUTER

Long Answer	e Questions
-------------	-------------

[WBJE 2006]

Draw time diagram for memory read

operation.

OR,

Give the timing diagrams of basic memory read and writing operations.[WBUT 2007]

OR,

Draw the time diagram for memory write operation. [WBUT 2008, 2011, 2012]

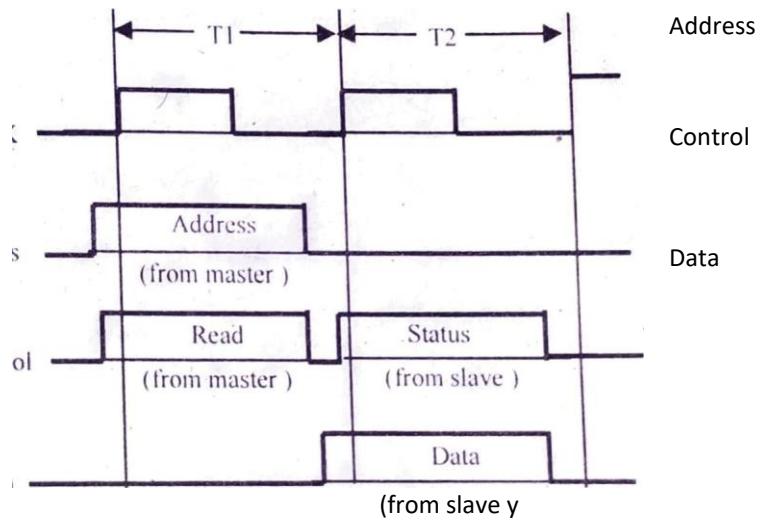
Answer:

In synchronous data transfer the timing signal is issued by the CPU clock i.e. memory operates according to the CPU clock. This type of data transfer occurs mainly between CPU & main memory.

Synchronous data transfer is mainly of two types:

(d) Memory Read (i.e. CPU reading from memory):

CLOCK



Steps:

As shown in figure above, in the P^t clock period, (T1) CPU sends the address of the memory location to be read, in the address bus.

At the same time (in T1), CPU places 'read' signal in the control line.

On getting the 'read' signal as well as the address from the CPU, memory sends back the required data to the CPU via the data bus in the next (T2) clock period. This is because, there is sufficient delay in main memory in receiving the CPU signals, searching for the required data and then placing the data in the data bus. So, data is sent back in the next clock period.

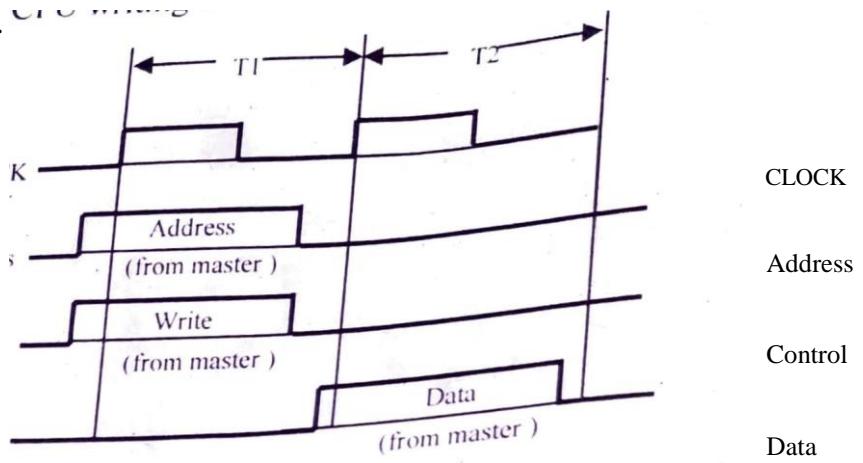
The 'status' signal is optional. It indicates whether the main memory is ready to response to the CPU request or not.

the so' control Only if bus the main on getting memory the CPU is ready request. to response, Then the it main will memory place a 'status' will place signal the data.high on

Hence if the main memory is not ready to response, it will not place the data on the data

POPULAR PUBLICATIONS

(h) Memory Write (i.e.



... to write in the 1st T-state (T₁).

CPU sends 'address' of the memory location to write, in the I T-state (T1 CPU then sends the 'write' control signal.

CPU, after some delay, sends 'data' to be written, in the next (T?) clock period / T2 state

[WBUT 20131

2. Draw and explain the instruction state cycle.

OR,

Explain the various phases of instruction OR,cycle in a basic computer. [WBUT 20161

Explain Instruction Cycle with suitable flow chart.

[WBUT 2018)

Answer:

"The basic phases of an instruction cycle (generally it consists of instruction fetch cycle and instruction execution cycle) are:

To fetch an instruction from the Inemon.

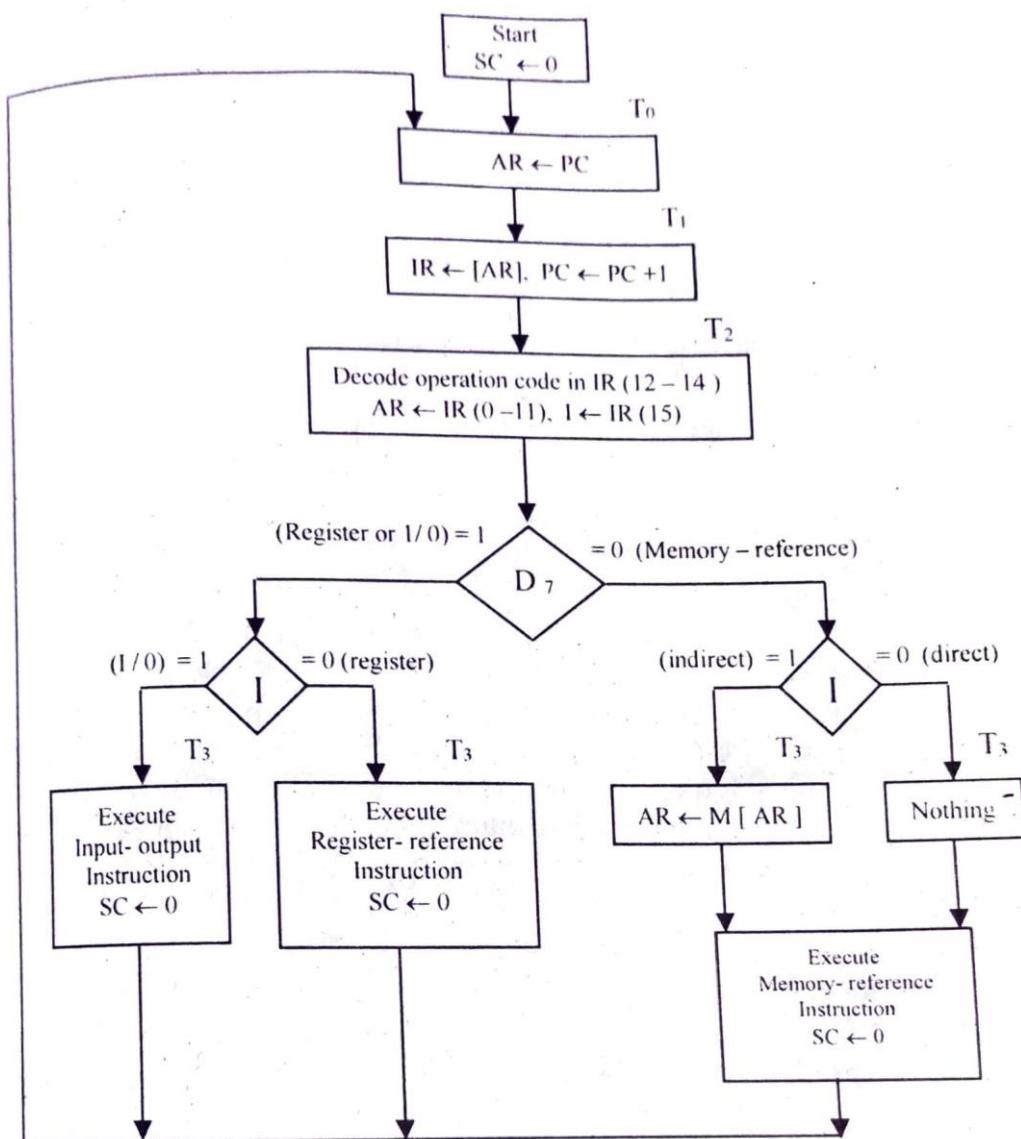
To decode the instruction.

If it is an indirect address instruction, then the effective address is to be read from the Inemon (i.e. the operands needed to be fetched from the memory one by one).

To execute the instruction.

The first three phases constitute the instruction fetch cycle and the last phase is the instruction execution cycle. On completion of step (d), the control again goes back to the step (a) to repeat the same cycle for the next instruction. This continues until a HALT instruction is encountered. The instruction cycle has got also main phases: the instruction fetch phase (this has three sub phases) and instruction execution phase. The figure shows the instruction cycle of

*— processor and instruction execution phase. The flowchart
any processor.*



The different steps of an instruction cycle are as follows:

Steps:

- The address of the first instruction in the program is loaded in the PC.
- A sequence counter (SC) is kept to track the different steps i.e. to keep track whether the steps are performed sequentially and correctly. The SC is incremented by one after each clock pulse (clock period). So, initially the SC is cleared to 0.
- In the first clock period (T₀), the contents of PC are placed to the AR (i.e. memory address register or MAR).
- In the second clock period (T₁), the content of the desired memory location (as given in the PC) is placed in the IR. Also the PC is loaded with the address of the next instruction.

- (e) In (T2), the IR content is decoded (i.e. the opcode of the instruction fetched from memory is decoded).

The flip-flop I hold the indirect bit [this specifies whether the instruction is inputoutput reference or register reference or memory reference (direct or indirect) in co-107

POPVVARpVBLICATIONS

nature]. The address part of the operands transferred to the AR.

- (f) Depending on the I-bit of the instruction, register reference or input-output reference

101

input-output me

e
(to be memory) are

it is decided whether the instruction

/) are again
instruction
reference in

input-output
code varies

and I = 1,
SC is again
goes back to

if the opcode is 1 11 then it is either a register reference or a reference instruction. else it is a reference instruction (i.e. opcode

- (g) If it is not a input-output reference instruction (i.e. if opcode is 1 11 and I then at (T0. input-output instruction is executed. After this the SC is cleared to 0 (as it completes the current instruction cycle and the control goes back

- (h) Also if it is not a register reference Instruction instruction (i.e. if opcode is 1 11 and I = 0),

cleared to 0.

(T0. the

reference is

executed. After this, similarly, the SC

at
the SC is

code varies
operand's
e directly

110 but I

- (i) On the other hand, if it is a direct memory reference instruction (i.e. if opcode from 000 through 1 10 but I = 0), then it is not needed to do anything as the effective address is already there in AR and so the instruction can be executed. Then again the SC is to be cleared to 0.

- (j) For indirect memory reference instruction (opcode varies from 000 through 1 11 but I), it is needed to fetch the effective address again from the memory. So only after fetching the actual operand from the memory, the instruction can be executed.

thus the SC is cleared to 0.

Steps (i) and (j) are performed at clock period (T3). However for indirect reference instructions it may continue till T4

- (k) Hence on completion of execution of the current instruction, SC is cleared and the address of the next instruction gets loaded in the PC and the same cycle continues.

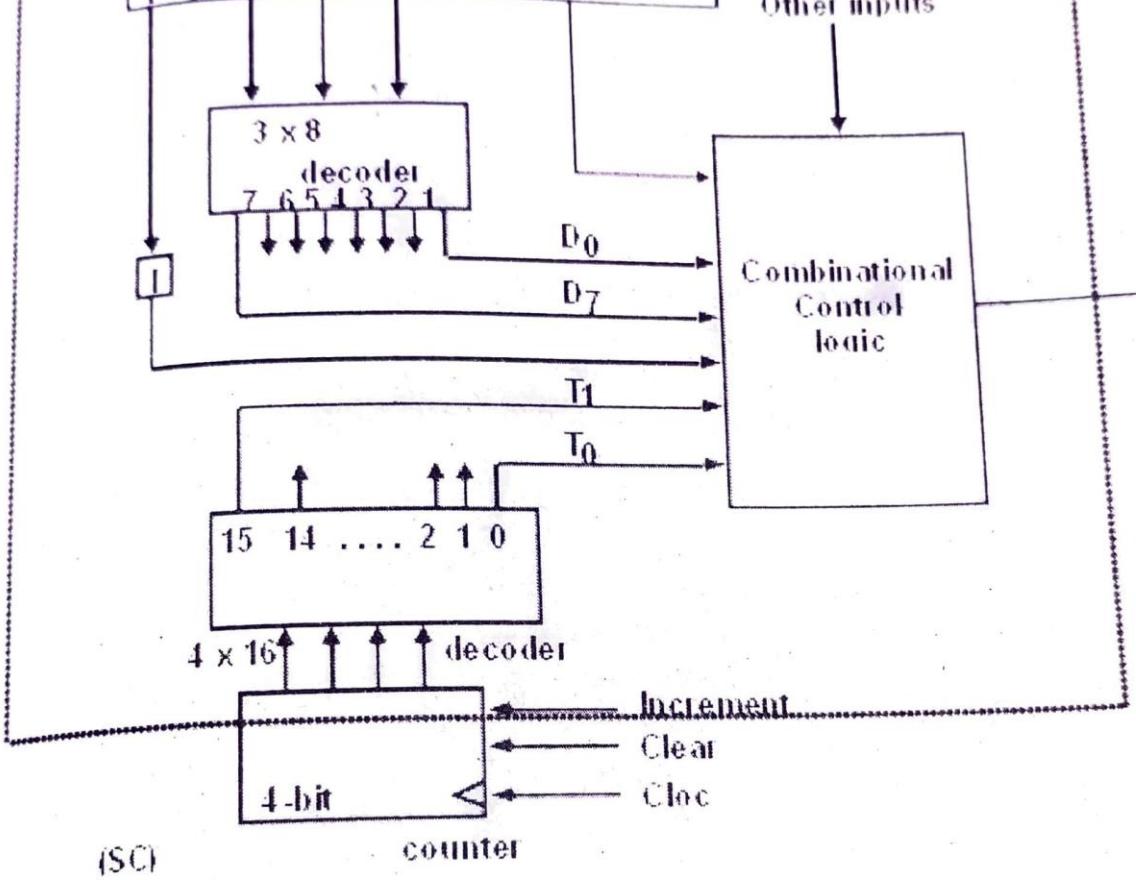
3. Differentiate between hardwired control and micro programmed control. Draw the block diagram of a basic hardwired control organization with two decoders, a sequence counter and a number of control logic gates. Answer: 2015, 2016] 1st part:

The differences are as follows:

No. Hardwired control unit

- | | |
|--|---|
| 1. Hardware implemented. | Microprogrammed control unit |
| 2. Non-programmable (done by logic implemented, design). | Programmable Software (microinstructions) |
| 3. Faster. | |
| 4. Not flexible. Slower. | |
| 5. Implementation is complex. | Flexible. |

Simple and easy to implement.



BUS STRUCTURE

Chapter at a Glance

- **Bus:** A bus is a set of wires/cables designed to transfer all bits of a w-bit word from a specified source to a specified destination. The source & destination are typically registers. A bus is a pathway connecting two or more registers within the Inodules CPU, Memory, I/O etc. A bus structure consists of a set of common lines one for each bit of a register, through which binary information is transferred one bit at a time. A bus may be unidirectional i.e. capable of transmitting data in one direction only, or it be bi-directional i.e. capable of transmitting data in both direction.

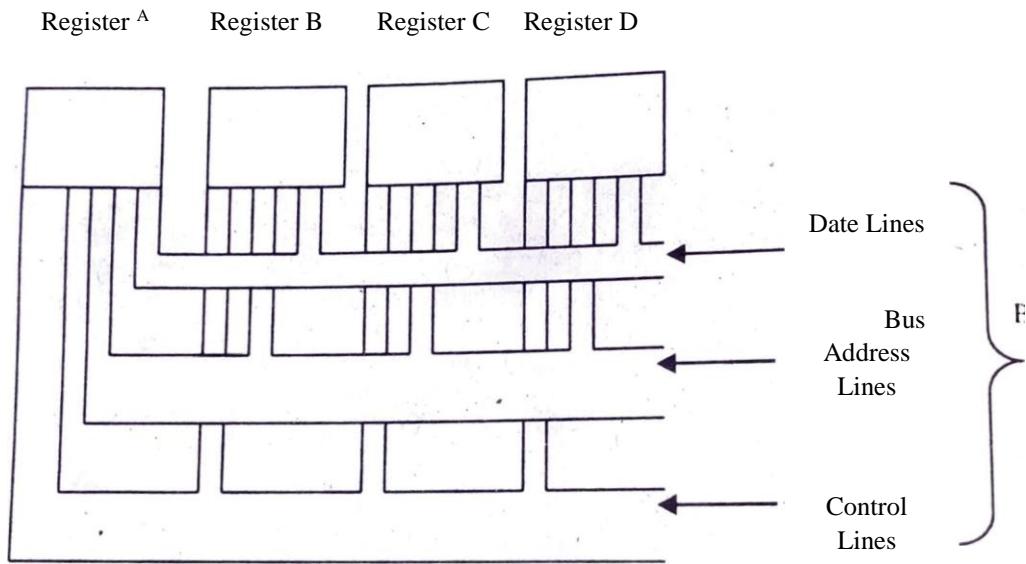


Fig: I

- Data, Address and control lines of bus:

Data Lines: These lines provide a path for moving data between registers. These are designed to transmit all bits of an n-bit word in parallel. So, they either consist of two sets of n• unidirectional lines or a single set of n-bi-directional lines.

Data lines are collectively called the data bus. This bus of size 'n' is usually a multiple Of with n = 8, 16 or 32 etc. separate lines.

Address Lines: Address lines of the bus are used to transfer the addresses of the sour(register sending the data and the destination register receiving the data. Address lines a collectively called the address bus.

Control Lines: These lines, transmitting control signals, are used to control the access to data & address lines i.e. control lines select which register, out of the multiple registers' v transmit its information through the bus.

co-110

R ORGANISATION.

Tri-state buffer: Bus system constructed with multiplexers has some serious disadvantages. If more number of registers are connected by the bus multiplexers cannot support that load and drop. Also as all registers (connected to the bus) draw some current from the bus even if they are not transmitting at that time, the bus voltage drops and thus the transfer becomes unreliable.

So to counter these problems a device (or a digital circuit) called tri-state buffer is used. It exhibits three states, two inputs and one output. The output can be in one of the three states (signal values) namely, logic 0, logic 1 and a 'high-impedance' state.

Multiple Choice Type Questions

1. The logic circuitry in ALU is

[WBUT 2008]

- a) entirely combinational
- b) entirely sequential
- c) combinational cum sequential
- d) none of these

Answer: (c)

2. A single bus structure is primarily found in [WBUT 2008, 2011]

- a) main frames
- b) super computers
- c) high performance machines
- d) mini and micro-computers

Answer: (d)

3. To construct an n-line common bus using MUX for k registers of n bits each, the number of MUXs and size of each MUX are [WBUT 2014]

- a) k and nxl
- b) n and 2^k
- c) n and kxl
- d) k and 2^n

Answer: (c)

POPULAR

4. The main purpose for using single Bus structure is [WBUT 20171

- a) Fast data transfer
- b) Cost effective connectivity and speed
- c) Cost effective connectivity and ease of attaching peripheral devices
- d) none of these

Answer: (c)

5. The main advantage of multiple bus organisation over single bus is

- a) Reduction in the number of cycles for execution [WBUT 20171
- b) Increase in size of the registers
- c) Better connectivity
- d) none of these

Answer: (a)

6, The maximum propagation delay for n-bit CLA is [WBUT 20181

sWer:(b)

1. a) A digital computer has a common bus system for 16 registers of 32-bits each i)

How many selection inputs are there in each multiplexer?

ii) What size of multiplexers are needed? [WBUT 2010] iii) How many multiplexers are there in the bus?

b) Why do most computers have a Common bus system?

Explain the importance of a common bus system in a computer.[WBUT 2011, 20171

Answer:

(as 2^{16} , since there are 16 registers.

ii) The size of multiplexers will depend on the number of input lines, Number of data input lines

iii) The number of multiplexers needed to construct the bus = number of bits in each register.
Hence 32 Multiplexers are needed.

b) A bus is a communication pathway connecting two or more registers within the system namely CPU, Memory, I/O etc.

Early microcomputers were essentially a passive backplane connected directly or through buffer amplifiers to the pins of the CPU. Memory and other devices would be added to the bus using the same address and data pins as the CPU itself used, connected in parallel. Communication was controlled by the CPU, which had read and written data from the devices as if they are blocks of memory, using the same instructions, all timed by a central clock controlling the speed of the CPU. A common bus system reduces the number of wires in the network. This creates an organized and clear connection structure between certain devices. By using latches, same bus can be used for transferring of data or instructions as well. These may lead to simple system structure.

2. Draw the logic diagram of a common bus which connects 4 registers of 4-bit each using tristate buffers.

Answer:

POPULAR PUBLICATIONS

Consider the diagram below. A, B, C & D are the 4 registers connected through [WBUT the 2010]⁴lines bus Both the inputs (D-ends of the flip-flops) & outputs (Q-end of the flipflops) are connected through the bus system. Tristate buffers are there at the sending & receiving ends of the registers.

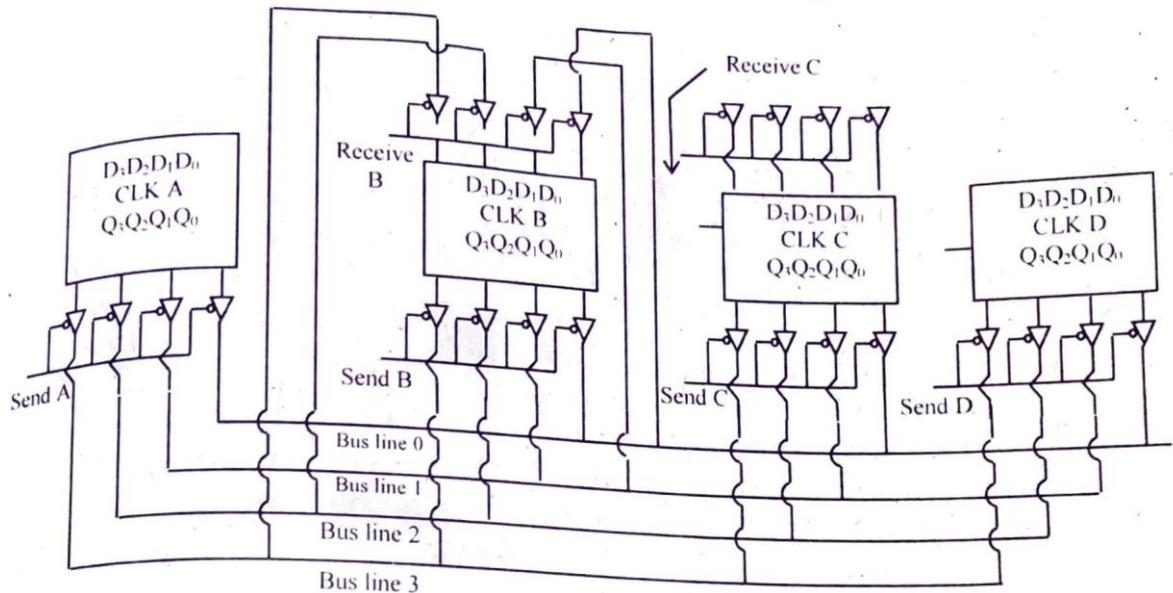


Fig: Bus system constructed with Tristate Buffers

3. What is tri-state buffer? Construct a single line common bus system using tristate buffer. [WBUT 20151 Answer: 1st part:

A tri-state gate is a digital circuit that exhibits three states out of which two states are normal signals equivalent to logic 1 and logic 0 similar to a conventional gate. The third state is a high-impedance state. The high-impedance state behaves like an open circuit, which means that no output is produced though there is an input signal and does not have logic significance. The gate is controlled by one separate control input C. If C is high the gate behaves like a normal logic gate having output 1 or 0. When C is low, the gate does not produce any output irrespective of the input values. The graphic symbol of a tri-state buffer gate is shown in Fig. 1.

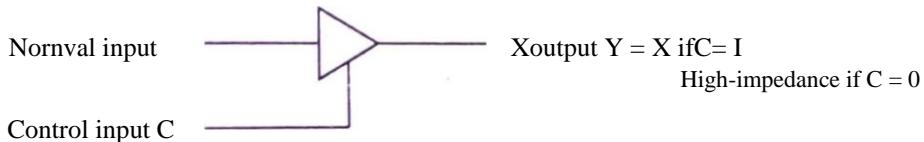


Fig: I Graphic symbol for a tri-state buffer gate

2nd Part:

A common bus system with tri-state buffers is described in Fig. 2. The outputs of four buffers are connected together to form a single line of the bus. The control inputs to the buffers, which are generated by a common decoder, determine which of the four normal inputs will communicate with the common line of the bus. Note that only one buffer may be in the active state at any given time. Because the selection lines S0, S1 of the decoder

POPULAR PUBLICATIONS

activate one of its output lines at a time and the output lines of the decoder act as the control lines to the buffers. For example, if select combination SIS() is equal to 00, then

0th output of the decoder will be activated, which then activates the top-most tri-state buffer and thus the bus line content will be currently A 0th bit of A register.

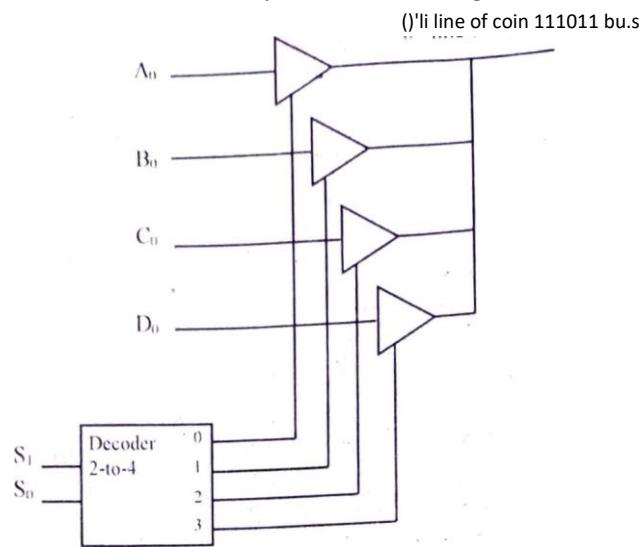


Fig: 2 A single line of a bus system with tri-state buffers

4. A digital computer has a common bus system for 16 registers of 32 bits each, The bus is constructed with multiplexers?

(i) How many selection inputs are there in each multiplexer?

(ii) How many multiplexers are there in the bus? [WBUT 2015, 2016] . Answer:
Refer to Question No. I(a) of Short Answer Type Questions.

5. Why 110 bus is different from a system bus?

[WBUT 2017]

Answer:

Computers have two major types of buses:

I. System bus: This is the bus that connects the CPU to main memory on the motherboard. The system bus is also called the front-side bus, memory bus, local bus, or host bus..

2. A number of I/O Buses, (I/O •is an acronym for input / output), connecting various peripheral devices to the CPU. These devices connect to the system bus via a 'bridge'

implemented in the processor's chipset. Other names for the I/O bus include "expansion bus", "external bus" or "host bus".

5. A block set-associative cache consists of a total of 64 blocks divided into 8 blocks sets.

The main memory contains 4096 blocks, each consisting of 12 words.

i) How many bits are there in a main memory address?

ii) How many bits are there in each of the TAG, SET and Word fields? [WB ^{UT} 201

ORGANISATION sns»cr:

The Illain Illenwry size = 4096 blocks 212 the tag and set field con)bine will have 12 bits.

Each block consist of 1 28 words 2 words.

Hence, the word field length will be 7 bits.

Hence, total size of lilain nnen)ory in word length $212 \times 27 = 219$

Hence there will be 19 bits in the main i) nemor There be 19 bits in a Iliain Inetnory

ii) There are 12 bits in each Of the TAG and SET fields and 7 bits in word field length.

Lon Answer

e uestions

What is bus? Draw and describe the bus architecture for a digital computer.

1.

[WBUT 2012]

Ans»er:

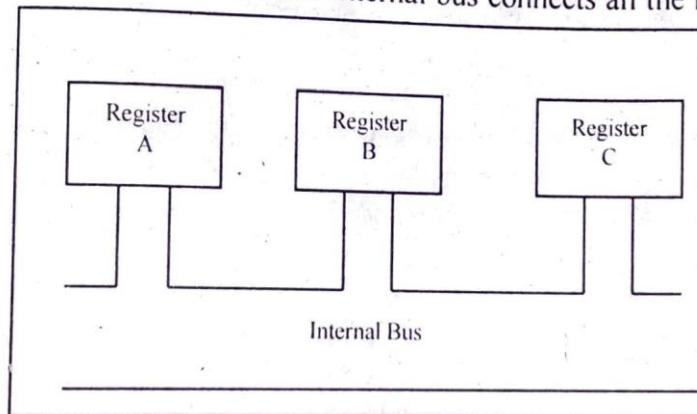
1stPart: Refer to Chapter at a Glance.

2nd Part:

Bus can be of 4 types,

Internal bus: This runs within the CPU. Internal bus connects all the registers within the

..... internal bus connects all the re

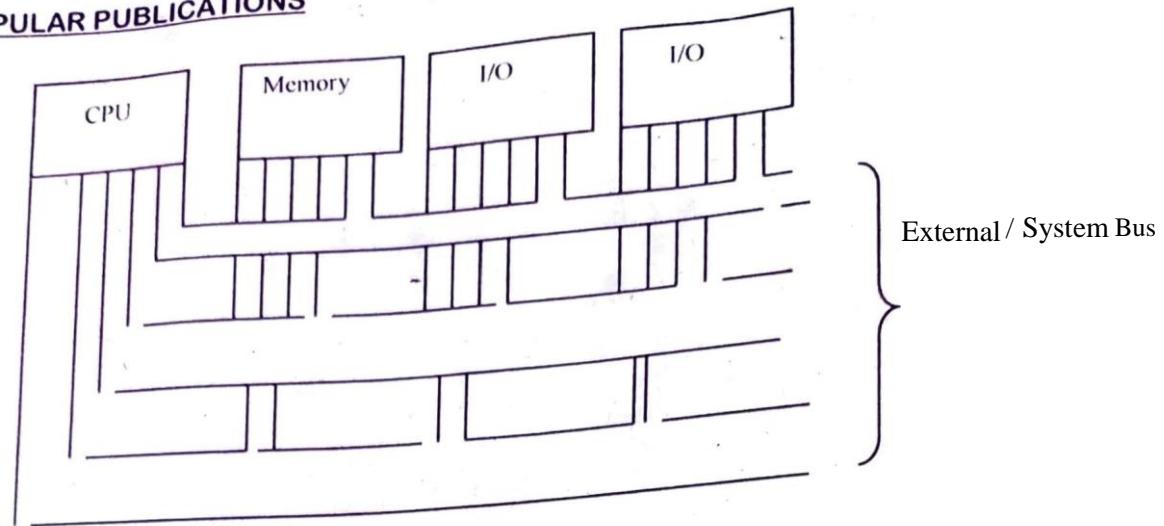


CPU.

External / System bus: This runs outside the CPU i.e. the external bus connects all the three subsystems (i.e. connects all the registers within the CPU, Memory and I/O unit) of a digital computer. This is a set of shared communication lines via which the registers inside the Input-Output processors & the CPU share a common access path to main Memory registers.

POPULAR PUBLICATIONS

POPULAR PUBLICATIONS



Shared Bus

Here several registers can be connected via a single common bus.

So, here the required number of buses and thus the number of cables required are much less. Cost of cables hence, is much less. However, shared buses do not permit simultaneous data transfers between different pair of registers, so this leads to a loss of performance compared to dedicated buses. To implement shared buses, more complicated logic circuits are needed.

Dedicated Bus

It has a unique source and a unique destination i.e. within each pair of registers there is a pair of dedicated bus for both way transmission of information.

If 'n' registers are interconnected by buses in all possible ways, then the number of dedicated buses required is ' $n(n-1)$ '.

Here the number of wires required is more. The number of wires required however varies with the number of registers to be connected. More the number of registers to be connected, more the number of wires needed.

As the number of system components and thus the number of buses to be connected increases, the number of pin requirements and the cost of cables along with the cost of complicated circuits also increases.

Here the simultaneous transfer of information between different pair of devices is possible.

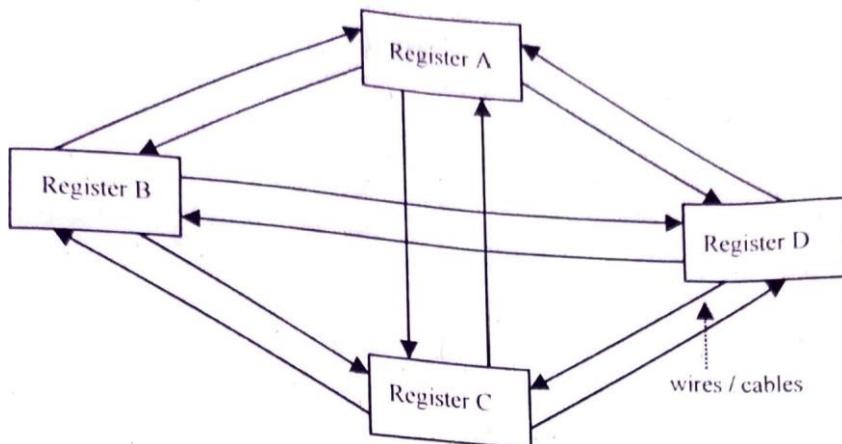


Diagram of dedicated buses connecting the registers

2. Write short note on 'Bus organization using tristate buffer'.

Answer: [WBUT 2007, 2009, 2012, 2017] Bus system constructed with multiplexers has some serious disadvantages. If more number of registers are connected by the bus, multiplexers cannot support that load and voltages drop. Also as all registers (connected to the bus) draw some current from the bus even if they are not transmitting at that time, the bus voltage drops and thus the transfer becomes unreliable.

So to counter these problems a device (or a digital circuit) called tristate buffer is used. It exhibits three states, two inputs and one output. The output can be in one of the three states (signal values) namely, logic 0, logic 1 and a 'high-impedance' state. While 0 & 1 typically correspond to two electrical voltage levels, e.g. 0 & 5 V, the 'high-impedance' state represents the state of a line that is electrically disconnected from all voltage sources i.e. an open-circuited line.

This high-impedance state is very useful in building the bus in the computer because the registers that are interconnected through the bus (i.e. communicate through the bus) do not draw any current through the bus unless they either transmit or receive data over the bus lines.

Functioning

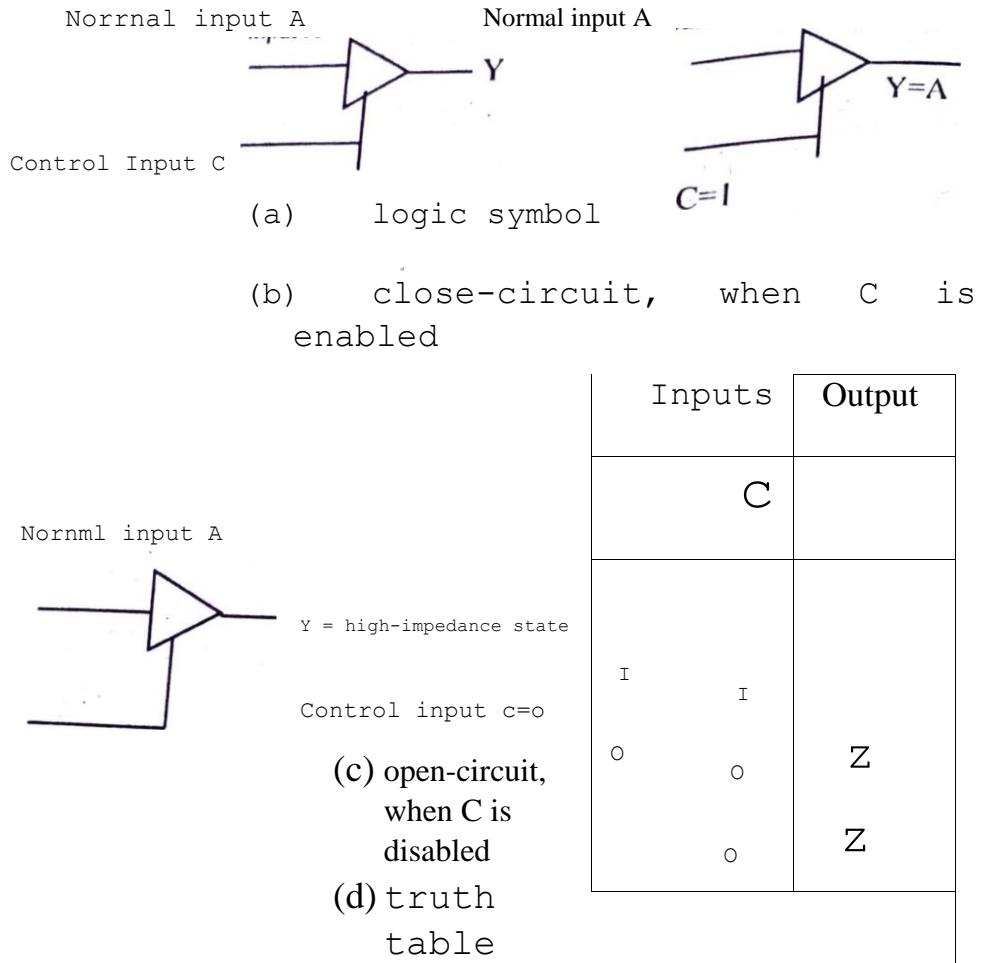
The tri-state buffer has three output states namely logic 0, logic 1 and a 'high-impedance' state. -It has two inputs (figure a). The inputs A (normal input) and C (control input) are ordinary binary signals that assume only the values 0 & 1; the output Y can assume the values 0, 1 & high-impedance.

The special input line C, called the 'output enable' or 'control input' when set to 1 (figure

b), the output $Y = A$ i.e. acts as a close-circuited line with what given in the input A , comes to the output Y (i.e. if $A = 0$ then $Y = 0$ or if $A = I$ then $Y = 1$) but if $C = 0$, it disables the output line Y by placing it in the high-impedance state.

POPULAR

BLI ATIONS



INPUT-OUTPUT ORGANIZATION

Chapter at a Glance

Input_output organisation: The Input-Output (I/O) organization of a computer provides the of conununication between the computer and the outside world i.e. with the help of the I O devices. users can interact with the coluputer. The I/O devices that are attached to the c01nputer are also known as peripherals.

of the 1.0 devices that are conunonly used in a computer are:

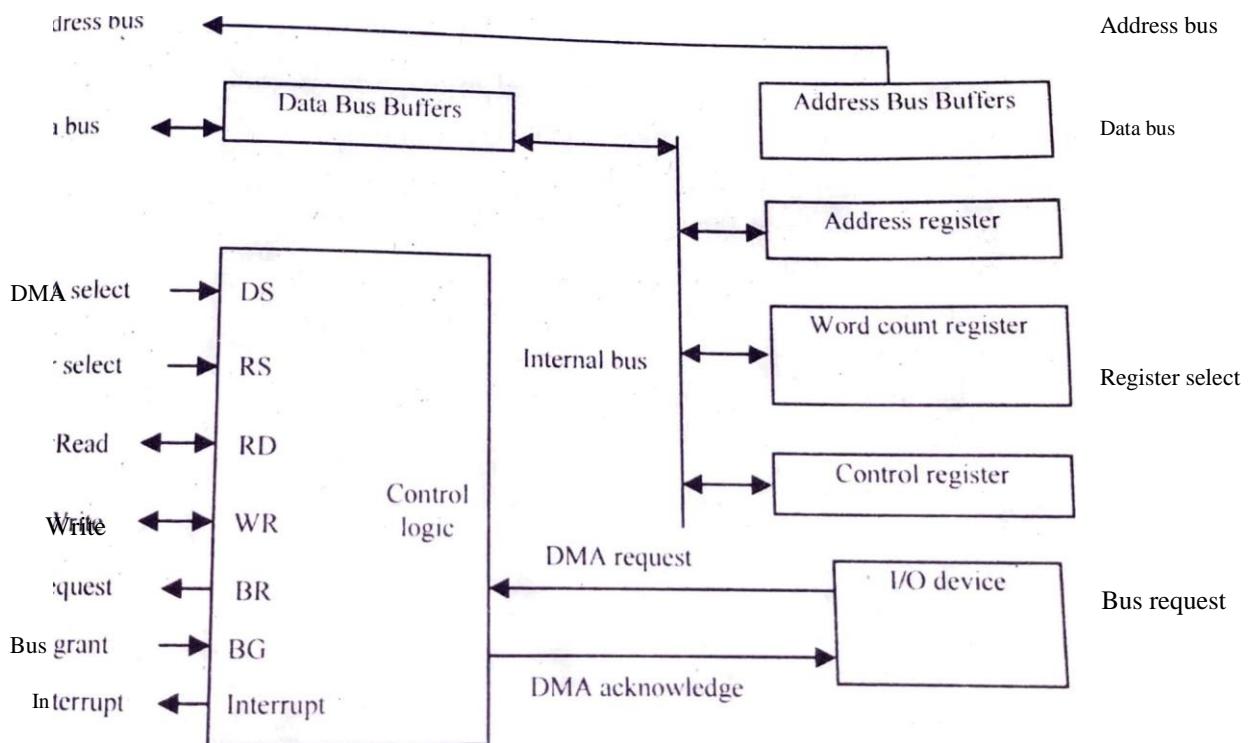
Some

Input Devices: Keyboard, Mouse, Joystick, Scanner etc.

Output Devices: Printer, Monitor etc.

Direct memory Access (DMA): In programmed I/O data transfer occur between CPU and the peripherals. However in DMA, data transfer occurs between I/O devices and the memory unit without direct intervention by the CPU. CPU only initiates the transfer by supplying the starting address of the memory location from where data is to be transferred and the number of words (bytes) to be transferred.

DMA Controller:



Interrupt: Interrupt is a signal from an I/O device to let the CPU know that it is ready to transfer data and that it is requesting service fronl the CPU. \s soon as CPU receives this Signal, it leaves

POPULAR

its current unfinished task as it is and branches to the interrupting device's interrupt service routine and executes it to process the transfer. When finished, CPU again comes back to its unfinished task and continues with it.

co-119

PUBLI

Different types Of Interrupts: There are generally three

(a) External Interrupts: Such type of interrupts may

any external ^{es like} event

from I/O device they are ready to transfer data, a timing device to Olif that the tinie of an event is overs it tuay occur due to some powerfailures etc.

(b) Internal Interrupts: Such type of interrupts, called traps, may occur due to some erroneous conditions in the program (i.e. illegal or erroneous use of instructions or in the program).

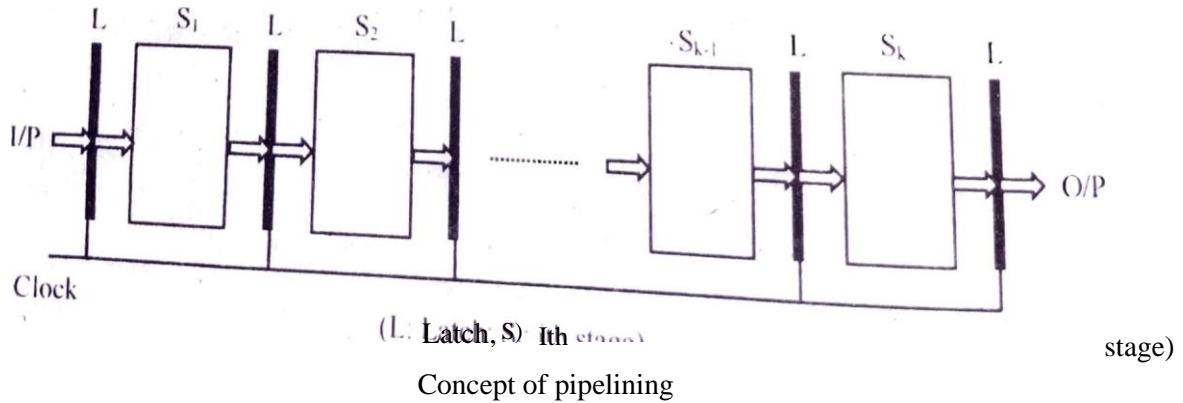
(c) Software Interrupts: Such type of interrupts be incorporated or embedded ^{in the} program as an instruction by a programmr and are thus initiated by executing th instruction (interrupt instruction). So if the programmer want to initiate any son interrupt procedure at any desired point in the program, he may write an intern, instruction at that point in the program.

Evample of software interrupt is: INT 32 (say). On execution of this interrupt instructior control branches to the ISR of the number 32 interrupt (i.e. 32 number interrupt line).

- Non- Vectored Interrupt: In this method, the branCh address . address of the ISR) is alwal. assigned to a fixed location in the memory and the processor always branches to particular location.
- Vectored Interrupt: In this method, the branch address (i.e. address of the ISR) is supplied b, the interrupting I/O device itself and the processor branches accordingly.
- Pipelining: Pipelining is a technique of decomposing a sequential task into subtasks, each subtask, with each subtask being executed in a special dedicated stage (segment) operates concurrently with all other stages. Each stage perforlns partial processing dictated b, the way the task is partitioned. Result obtained from a stage is transferred to the next stage i the pipeline. The final result is obtained after the instruction has passed through all the stage All stages are synchronized by a common clock. Stages are pure

combinational circuit performing arithmetic or logic operations over the data stream flowing through the pipe.

stages are separated by high-speed interface latches (i.e., collection of registers). Figure below shows the pipeline concept with k stages.



Multiple Choice Type Questions

1. In a micro-processor, the address of the next instruction to be [WBUT executed, 2008] is stored in
- a) stack pointer
 - b) address latch
 - c) program counter
 - d) general purpose register

Answer: (c)

2. and Memory I/O devices mapped is used I/O scheme for is used for the allocation of address [WBUT to memories 2008]
- a) small system
 - b) large system
 - c) both large and small systems
 - d) very large system

Answer: (a)

3. For BIOS (Basic input/output system) and IOCS (input/output control system), which one of the following is true? [CWBUT 2009]

- a) BIOS and IOCS are same
- b) BIOS controls all devices and IOCS controls only certain devices
- c) BIOS is not a part of operating system and IOCS is a part of operating system
- d) BIOS is stored in ROM and IOCS is stored in RAM

Answer: (c)

4. A priority interrupt may be accomplished by

- a) Polling
- b) Daisy chain
- c) Parallel method of priority interrupt
- d) All of these

[WBUT
20101]

Answer: (a)

5. BIOS is

- a) a collection of I/O driver programs
- b) part of OS to perform I/O operation
- c) firmware consisting of I/O driver programs
- d) a program to control one of the I/O peripherals

[WBUT
20131]

Answer: (b)

6. In DMA the term cycle stealing means

[WBUT
20141]

- a) Controller gets opportunity to transfer only one word in a timeslot
- b) CPU releases the bus and DMA controller can use endlessly
- c) 100 bytes are allowed to be transferred
- d) none of these

Answer: (a)

POPULAR VOLICATtONS**Short Answer Type Questions**

(WBUT 2)

1. What are the different hazards in Pipeline?

OR.

(WBUT 20)

Explain Pipelining and Hazards

Answer:

Pipeline hazards are situations that prevent the instruction in the instruction from executing during its designated clock cycle.

hazardK:

- i) Control hazards
- n) Structural hazards
- iii) Data hazards

Control hazards: They arise from the pipelining of branches and other instructions change the content of program counter (PC) register.

Structural hazard.v: Structural hazards occur when certain resource (memor, functionai unit) is requested by more than one instruction at the same time.

Data hazard: Inter-instruction dependencies may arise to prevent the sequential (in-order) data flow in the pipeline, when successive instructions overlap their fetch, ~~dec~~ and execution through pipeline processor. [This situation due to inter-instruction dependencies is called data hazard.]

2 What are the different types of interrupt? Give examples. PN BUT 20081 Answer:
Refer to Chapter at a Glance.

What 3. What are the differences between vectored and non-vectored interrupt? [WBUT 2009, 2011]

[WBUT 2009, 2011,

20181 Answer: ^{r^t} Part:

Interrupt is a signal from an I/O device to let the CPU know that it is ready to transfer data and that it is requesting service from the CPU.

As soon as CPU receives this signal, it leaves its current unfinished task branches to the interrupting device's interrupt service routine and executes IL to process the transfer. When finished, CPU again comes back to its unfinished task and continues with it.

ORGANISATION od part:

2

Vectored Interru t

In this Inethod, the branch address (i.e.the Interrupt address of the ISR) is supplied by theis interrupting I/O device itself and the processorthe branches accordingly. branches to that

2. The address of the service routine is hard-routine wired.by the device.
4. a) Where does DMA mode of data

Non-Vectoro	branch	address (i.e. assigned to a y and the at particular
1. In this method, address of the ISR)	fixed location in processor always	service
2. The address of the	be su lied externally	ine needs to ce.

find its use?

[WBUT 2009, 2014] Answer:

In interrupt-driven I/O (as well programmed I/O) transfer of data between memory and an I/O module takes place with the active intervention of the CPU. The speed of such data transfer is often limited by the speed with which the CPU services a device. For transfer of small volume of data such CPU intervention (and thus limited speed of data transfer) is ok, but transfer of large volume of data via the CPU takes a lot of time. So while transfer of large volume of data between memory and I/O module by DMA based

I/O method, CPU is removed from the path (CPU only initiates the transfer) and data flows very fast directly between the memory and the concerned I/O module(s) with the I/O device managing the memory buses..

Also during interrupt-driven I/O method, CPU, along with handling I/O tasks, is also busy handling other tasks. This to some extent hampers the I/O transfer rate. However with DMA based I/O method nothing of that sort occurs.

Hence in the above two circumstances DMA mode of data transfer find its use.

b) What are the different types of DMA controller and how do they differ in their functioning? [WBUT 2009, 2010, 2012, 2014] Answer:

There are three types of DMA controller: (i) Cycle Stealing DMA, (ii) Burst Mode DMA, (iii) Flyby DMA.

Their functioning:

(i) **Cycle Stealing DMA:**

In this mechanism, DMA controller transfers one word at a time and then returns the control of the bus to the CPU. Then again after a CPU cycle, the control comes back to the DMA controller, which again send one word, and gives back the bus control to the CPU. This carries on until the entire block of data is transferred. So, DMA transfer virtually 'steals' one memory in between every CPU cycle.

(ii) **Burst Mode DMA:**

Burst Mode DMA, in contrast, generally assumes that the destination and source

addresses can take transfers as fast as the controller can generate them. The program sets up the controller, and then (perhaps after a single ready indication from a port occurs), the entire source block is copied to the destination. The DMA controller gains exclusive

access to the bus for the duration of the transfer, during which time the Program is effectively shut off. Burst mode DMA can transfer data very rapidly indeed.

(iii) **DMA:**

flybyDMA, Flyby DMA is something controller that gains not supported access to on the Inany bus and controllers' puts the source or destination different address color. The out. DMA Then, it initiates what is in effect a read and a write cycle simultaneously. The data is read from the source address, and written to the destination at the same time.

This implies that either the source or destination does not require an address, since it is very unlikely that both would use the same. An example might be copying data from memory to a FIFO port - the source address (a pointer to memory) increments on each transfer, while the destination is always the same FIFO. Flyby transactions are very fast since the read/write cycle pair is reduced to a single cycle. Both burst and synchronous types of transfers can be supported.

5. What are differences between Serial and Parallel transmission? WBUT 20121

Answer:

In serial transmission, bits are sent sequentially on the same channel (wire) which reduces costs for wire but also slows the speed of transmission. Also, for serial transmission some overhead time is needed since bits must be assembled and sent as a unit and then disassembled at the receiver.

In parallel transmission, multiple bits (usually 8 bits or a byte/character) are sent simultaneously on different channels (wires, frequency channels) within the same cable or radio path, and synchronized to a clock. Parallel devices have a wider data bus than serial devices and can therefore transfer data in words of one or more bytes at a time. As a result, there is a speedup in parallel transmission bit rate over serial transmission bit

6. What are the advantages of Interrupt I/O over programmed I/O?

[WBUT 2013, 2015, 2016]

Answer:

Programmed I/O Slowest

in speed.

Least expensive.

POPULAR

Simple to design.

Wastage of CPU cycle degrades the performance of the computer.

Interrupt-initiated I/O

Medium in speedi

Medium cost.

Slightly complicated.

CPU cycle is not wasted.

COMPUTER ORGANISATION

is meant by "pipeline architecture"? How does it improve the speed of

BUT 20' 31

V, pipeline a technique used in the of computers Increase the number of parallel units of time. Instruction is broken up into a series called a pipeline. Rather than each instruction sequentially (one at a time, finishing one before the next), each instruction is split up into a series of stages so different can be executed concurrently (by different circuitry) and in parallel (at the same time). Ideal speedup from a pipeline is equal to the number of stages in the pipeline.

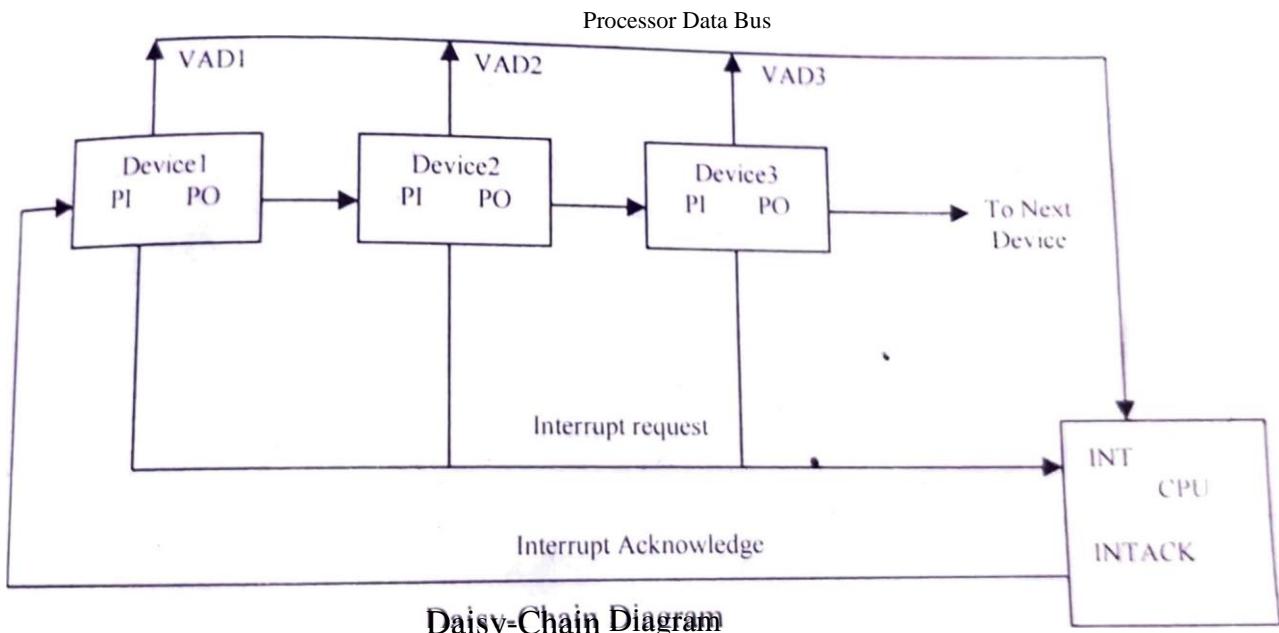
Time per instruction on unpipelined machine

Number of pipeline stages

S. Explain with diagram the daisy chaining priority interrupt technique.

[WBUT 2013, 2014]

Answer:



A daisy chain is an interconnection of computer devices, peripherals, or network nodes in series, one after another. It is the computer equivalent of a series electrical circuit. In personal computing, examples of "daisy-chainable" interfaces include Small Computer System Interface (SCSI) and FireWire, which allow computers to communicate with Peripheral hardware such as disk drives, tape drives, CD-ROM drives, printers, and scanners faster and more flexibly than previous interfaces.

The daisy-chaining method of establishing priority consists of a serial connection of all devices that request an interrupt. 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. This method of connection between three devices and the CPU is shown in the above figure.

PUBLICATIONS

The advantage of the daisy chain is its simplicity. Another advantage is scalability.

The user can add 1110 to nodes anywhere along the chain, up to a certain maximum (16 in SCSI-2 or SCSI-3, for example). A daisy-chain network can be long in terms of the distance from one end to the other, but is not well suited to situations where nodes must be scattered all over a geographic region. In such a case, the cables must zig-zag around, and the overall length of the network can become huge compared with the actual

distances between the nodes. This can cause the network to operate slowly for users near opposite ends of the chain.

9. What do you mean by memory-mapped I/O and I/O mapped I/O? [WBUT 2014]

Answer:

In memory mapped I/O, all peripherals are treated as memory locations. This means that memory addresses are assigned to I/O ports. The CPU just reads and writes data to a memory location, and the I/O controller automatically maps the port. i.e. transferring data from memory to port and from port to memory. This way of mapping allows the use of full instruction set of a CISC computer directly on peripherals, because to the CPU (or program), they are just simple memory locations. Such systems

do not have I/O specific instructions.

In I/O mapped I/O, the peripheral devices are addressed directly by the CPU using the port addresses. The length of port address is generally less than the length of a memory address. There are specific instructions for carrying out input and output tasks at the ports. This method of mapping does not facilitate the use of memory oriented complex instruction set directly on port addresses.

POPULAR

10. What is pipelining? Describe pipeline hazards.

[WBUT 2014]

Answer:

1st Part: Refer to Chapter at a Glance.

2nd Part: Refer to Question No. I of Short Answer Type Questions.

Long Answer Type Questions

I. What is interrupt? What are the differences between vectored and non-vectored interrupt?

Answer: [WBUT 2009] Refer to Question No. 3 of Short Answer Type Questions.

2. How does polling work?

Answer:

[WBUT 2012]

Establishing the priority of simultaneous interrupts can be done by 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 internal units. The program that takes care of interrupts 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.

highest priority source is tested first. Then a service routine for this source is executed. If its interrupt signal is on, control is transferred to the interrupt service routine.

on. Thus the initial service routine tests all interrupt sources in sequence and branches all to interrupts one after another. This is possible if a program service routine tests particular service routines sequentially. The disadvantage of this software method is that if there are many interrupt sources, the time required to poll them can exceed the time available to service the I/O device. In this situation a hardware priority-interrupt unit can be used to speed up the operation.

3. a) Give the main reason why DMA based I/O is better in some circumstances than interrupt driven I/O. [WBUT 2007] Answer:

In interrupt-driven I/O (as well as programmed I/O) transfer of data between memory and an I/O module takes place with the active intervention of the CPU. The speed of such data transfer is often limited by the speed with which the CPU services a device.

transfer of small volume of data such CPU intervention (and thus limited speed of data transfer) is ok, but transfer of large volume of data via the CPU takes a lot of time. So while transfer of large volume of data between memory and I/O module by DMA based I/O method. CPU is removed from the path (CPU only initiates the transfer) and data flows very fast directly between the memory and the concerned I/O module(s) with the I/O device managing the memory buses.

Also during interrupt-driven I/O method, CPU, along with handling I/O tasks, is also busy handling other tasks. This to some extent hampers the I/O transfer rate. However with DMA based I/O method nothing of that sort OCCURS.

Hence for the above two reasons DMA based I/O method is better than interrupt-driven I/O method while of large volume of data between memory and I/O module.

b) Explain the basic Direct Memory Access (DMA) operation for transfer of data bytes between memory and peripherals. [WBUT 2007, 2008, 2010, 2015, 2016] OR, Explain in detail the Bus Arbitration techniques in DMA. [WBUT 2017]

Direct Memory Access or DMA is the data transfer technique directly between the fast peripheral devices (like magnetic disk) and the memory unit through the bus without the direct intervention of the CPU. The transfer occurs as follows:

Steps:

- (a) An I/O device when ready to transfer data to the memory, it sends a DMA request to the DMA controller it is attached to.
- (b) On getting the request, the DMA controller enables the BR line to request the CPU to release the bus.
- (c) Now two cases will arise:

co-127

PUBLICATIONS

BC, = O buses are still not released by CPU and CPU will communicate with the DMA controller. CPU (with the help of the WR line of the DMA controller) into DMA address the initial address of the memory location where data is to be transferred, CPU writes the number of words to be transferred into the DMA word count register, CPU specifies the mode of control whether 'read, or write'. Also CPU reads the status of the DMA controller through the RD line.

BG = I that CPU has granted the DMA controller's request of releasing the bus and responds by enabling the bus grant line. The bus is now free and DMA controller can now control then.

- (d) the bus is under the control of the DMA controller. CPU now opts out of the transfer and carries on with other tasks.

DMA controller now places the initial address of the memory location from where the data is to be transferred into the address bus from its address register. Then the controller sends an acknowledgement to the I/O device, which has requested for the DMA transfer.

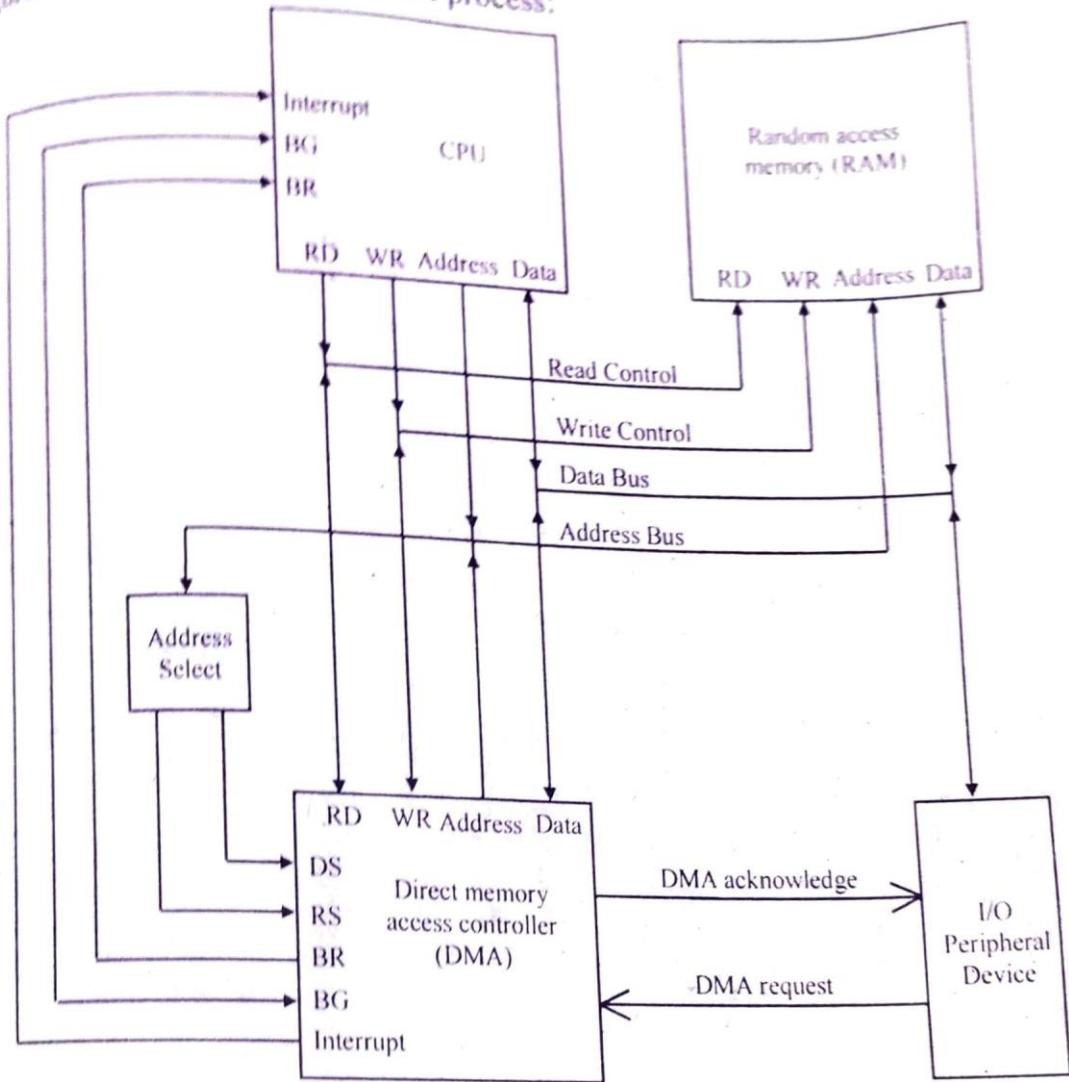
Based on the content of the control register, DMA enables the RD or WR line i.e. if data is to be transferred from I/O device to memory then it's a write operation and WR line will be enabled else the RD line will be enabled.

On transfer of each word, the word count register in the DMA controller decrements by one.

If the entire block of data is transferred or if the word count register count down to zero, the DMA stops the transfer.

It then removes the bus request by disabling the BR line. CPU then disables the BG line and again takes control of the bus. The DMA controller may also inform the CPU that the transfer is over and it has released the bus by sending an interrupt signal to the CPU. CPU then checks whether the transfer has been completed successfully by reading the count of the word count register. The zero value in the register indicates that the transfer has been done successfully.

The figure below shows the entire DMA process:



c) What is programmed I/O technique? Why is it not very useful?

[WBUT 2007, 2010]

OR,

What is programmed I/O technique? [WBUT 2008 Answer:

Programmed I/O transfers are initiated as a result of I/O instructions written in the computer program. Once the transfer is initiated, the CPU has to constantly monitor the interface units of the I/O devices to see when the transfer can be actually made i.e. when the device will be ready to transfer data.

It is not very useful, as this constant monitoring of the devices by the CPU requires lot of CPU time. In these times, the CPU remains idle, as it cannot do any other work but

to check whether the I/O device is ready to transfer data or not. Lot of CPU time is wasted (In this I/O mechanism), So for this reason this technique is the slowest compared to interrupt I/O and DMA mechanisms and thus degrades the computer performance.

between

[WBUT 2008, 2010]

4. Differentiate memory mapped I/O and I/O mapped I/O.

[WBUT

2013] Write three points to differentiate memory mapped I/O and I/O mapped I/O.

[WBUT 201?])

Compare and contrast Memory mapped I/O and I/O mapped I/O.

Mcrnol ma ed I/O

. In Isolated or 1/0 Inapped I O, 1 0 1 . In memory mapped I/O, a chunk of the are by the CPU CPU's address space is reserved for and hence occupy a separate chunk of I/O devices. addresses predetermined by the CPU for col)mnand 2. In memory mapped I/O the data transfer is and I O pott is not mentioned as a like between two memory segments. i.e. the I/O port is referred by a Inemory address.

Inlppoed 3. 10 Inapped I/O) 1 0 (also known as port 3. Memory mapped 1 0 is mapped into the address space and is accessed via a

uses a

user Inenwry, and is accessed in the satne

4. Microprocessor or microcontroller that doesn't support I/O mapped I/O.

4. Microprocessor or microcontroller supports I/O Inapped I/O.

5. In 10 Inapped I/O we use 8-bit address to interface I/O.

5. In memory mapped locations I/O we use Inenlory interfaced as an address devices.

6. In I O Inapped I/O we can use only two instructions i.e. IN and OUT.

6. In memory Inapped I/O we can

instructions for data flow.

separate, dedicated same address space as program memory and/or way

that

of to I (j use all

5. Why do peripherals need interface circuits with them?

[WBUT 2013, 2018]

I/O interface units are special hardware components that lie between the I/O devices and the processor bus. These devices serving as special communication links, between the CPU' and the peripherals, actually synchronize and supervise i.e. control all input and output transfers.

Certain differences between the characteristics (functional and behavioral) of the peripherals and the CPU (and also memory) exist. Interface units are needed to resolve the differences, which are:

- Operations of the peripherals, which are either electro-magnetic or electron-mechanical devices, are different from that of the CPU and memory. which are generally electronic devices. So a signal conversion between the two is required and is done by the interface unit between them).
- [the interface unit has to synchronize the data transfer mechanism between the slow

Interface units handle the differences between the data codes and formats in peripherals and the CPU or memory word formats.

CO-no

COMPUTER

6 g) what Explain are different vector interrupts? hazards in pipelining. How are they

used in
implementing
(WBIJT
hardware2015
(WBUT 2015)

interrupts? is speedup, throughput and efficiency of a pipelined architecture?

What
c)

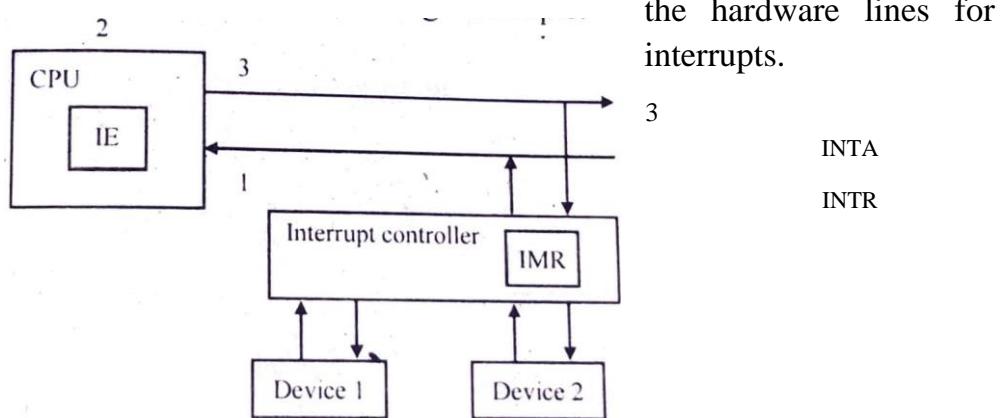
tWBUT 2015,

20181 8no•cr:

g) Refer to Question No. 10 of Short Answer Type Questions.

h) In a interrupt(I/O method, the source device that interrupts, supplies the branch information (i.e., the starting address of interrupt service routine (ISR) to the CPU. This information is called the interrupt vector, which is not any fixed memory location.

To implement interrupts, the CPU uses a signal, known as an interrupt request (INT R) to the interrupt handler or controller hardware, which is connected to each I/O device that can issue an interrupt to it. Here, interrupt controller makes liaison with the CPU on behalf of I/O devices. Typically, interrupt controller is also assigned an interrupt acknowledge (INTA) line that the CPU uses to signal the controller that it has received and begun to process the interrupt request by employing an ISR. The following figure below shows the hardware lines for implementing



1. Interrupt from interrupt controller when data transfer is needed.
2. Using IE flip-flop, CPU detects interrupt.
3. CPU branches to a respective device's ISR after enabling INT A.

The interrupt controller uses a register called interrupt-request mask register (IMR) to detect any interrupt from the I/O devices. If there is n number of I/O devices in the system, then IMR is n-bit register and each bit indicates the states of one I/O device. Let IMR's content be denoted as EOEIE, .E . When E₀ = 1 then device 0 interrupt is recognized; When E₁ = 1 , then device 1 interrupt is recognized and so on. The processor uses a flag bit known as interrupt enable (IE) in its states register (SR) to process the interrupt. When this flag bit is set the CPU responds to the presence of interrupt by enabling INTA line; otherwise not. When the INTA is accepted by a device, device puts its own interrupt vector address (V AD) to the data bus using interrupt controller.

POPULAR PUBLICATIONS

- c) a linear pipeline of k can execute n tasks in $k + (n - 1)$ clock cycles. **where** k cycles are needed to complete the execution of the very first task and $(n - 1)$ tasks require $n - 1$ cycles. Thus the total time required is

here r is the clock period. Consider an equivalent-function non-pipelined processor has a flow-through delay of kr . The amount of time it takes to execute n tasks on this non-pipelined processor is $T_1 = nkr$.

Speedup Factor: The speedup factor of a k -stage pipeline over an equivalent non-pipelined processor is defined as

$$S_k = \frac{T_1}{T_k} = \frac{nk\tau}{k\tau + (n-1)\tau} = \frac{nk}{k + (n-1)}$$

Efficiency and Throughput: The efficiency E_k of a linear k -stage pipeline is defined as

$$E_k = \frac{S_k}{k} = \frac{n}{k + (n-1)}$$

Obviously, the efficiency approaches 1 when $n \rightarrow \infty$, and a lower bound on E_k is $1/k$ when $n=1$. The pipeline throughput H_k is defined as the number of tasks (operations) performed per unit time:

$$H_k = \frac{n}{[k + (n-1)]\tau} = \frac{nf}{k + (n-1)}$$

The maximum throughput f occurs when $E_k = 1$ as $n \rightarrow \infty$.

Note that $H_k = E_k \cdot f = E_k / \tau = S_k / k\tau$

7. a) What are the hazards of instruction pipelining? How are these taken care of?
- b) Explain the Strobe Control method of Asynchronous data transfer. What are the disadvantages of this method?
- c) What do you understand by the term 'Program Interrupt'? Explain with the help of suitable diagrams.

[WBUT]

201 Answer:

a) **Structural Hazards**

Structural hazards OCCUR when two instructions in a pipeline need the same hardware resource at the same time. Structural hazards can be avoided by stalling, duplicating the resource, or pipelining the resource.

POPULAR PUBLICATIONS

For example, suppose the processor only has a single port to memory used for both data and instructions. Then there is a structural hazard between the MEM phase of a load store instruction and the IF phase of the instruction that needs to be fetched at that time. This hazard can be avoided by either stalling the instruction fetch or by having

co-132

COMPUTER

ports. Most modern processors have separate data and instruction caches which, in effect, gives them two memory ports. In addition, some modern processors can complete multiple instructions in a single long cycle. One example is - an integer multiply whose latency is approximately 10 cycles. This is because you cannot handle two successive multiplies.

Instructions deal with long latency stalling instructions, the second. The execute circuitry is generally divided up into

functional units, each handling a small number of similar instructions. These functional units can be pipelined by adding pipeline registers. This lets you start long latency instructions every cycle.

Control Hazards occur when conditional branches interfere with instruction fetches in a pipeline. The problem is that it is not known whether or not a conditional branch will be taken until sometime after the cycle fetching the next instruction. Also, the branch target address needs to be computed if the branch is taken.

A control hazard could be handled by stalling the next instruction fetch. However, this has a significant impact on performance, especially in tight loops, where many programs spend much of their time. A common technique for reducing the stalls associated with control hazards is speculative execution - guess whether or not the branch will be taken and fetch the next instruction based on the guess. To do this, the machine needs two tables containing information about recent branches:

- A branch history table records bits about recent branch history, that is, whether or not a branch was taken. The processor uses these bits to guess whether or not a branch will be taken.
- A branch target table holds target addresses for recent branches. This table reduces the time needed to determine the branch target address.

Speculative execution also requires a mechanism for backing out of instructions executed based on incorrect guesses and resuming execution of the correct instruction sequence.

Data Dependences and Data Hazards

Data hazards occur when two instructions in a pipeline refer to the same register and at least one of them writes to the register. Compiler writers use the phrase "data dependences" to cover the same kind of problem, but their terminology refers to what you can see in an instruction stream without considering the pipeline.

Also, execution circuitry is usually broken up into multiple functional units, each performing different types of operations. These functional units can be performing operations in parallel. More complex operations may take several cycles to complete. To illustrate the difficulties that result, consider the following MIPS code snippet.

```
div.d $11), $f2, $f4 mul.d $f6, $f8, $11) add.d $11), $f1 0, $f12
```

The use of register \$f1 can give rise to three different kinds of problems in this code.

POPULAR PUBLICATIONS

Read after Write (RAW) hazards, also known as true dependences write after write (WA W) hazards, also known as output dependences Write after Read (WAR) hazards, also known as anti-dependencies hazard. The naming of these hazards is based on what is supposed to happen. That is, a WAR an instruction that writes to a register follows soon after an instruction that reads from the register. If the write precedes the read then the first instruction (the read) is working with the wrong data value.

1) -- add hardware and/or, complexity to work around the hazard so it does not exist. This can be achieved by bypassing/forwarding or speculation.

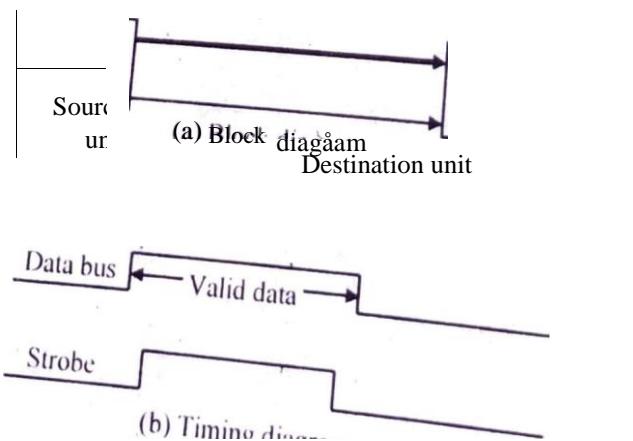
2) stall -- Sacrifice performance to prevent the hazard from occurring • Stalling causes "bubbles"

b) Strobe Control Technique A single control line is used by the strobe control method of asynchronous data transfer to time each transfer. The strobe may be activated by either the source or the destination unit. A source-initiated transfer is depicted in figure (1st) below. The source takes care of proper timing delay between the actual data signals and the strobe signal. The source places the data first, and after some delay, generates the strobe to inform about the data on the data bus. Before removing the data, the Source removes the data. By these two leading and trailing end delays, the system ensures reliable data transfer.

Similarly, the destination can initiate data transfer by sending a strobe signal to the Source unit as shown in figure(2nd) below. In response, the source unit places data on the data bus. After receiving data, the destination unit removes the strobe signal. Only after sensing the removal of the strobe signal, the source removes the data from the data bus.

The disadvantage of the strobe method is that the source unit that initiates the transfer cannot know whether the destination unit has actually received the data or what was placed in the bus. Similarly, a destination unit that initiates the transfer cannot know whether the source unit has actually placed the data on the bus.

c)



POPULAR PUBLICATIONS

Fig 1: Source-initiated strobe for data transfer

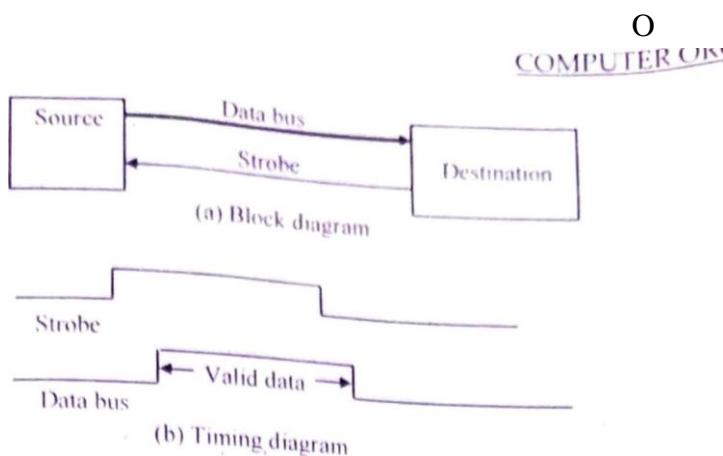


Fig 2: Destination initiated strobe for data trans,fer

interrupt. a hardware mechanism that allows events to get serviced. can interrupt execution of an ongoing task. An event can interrupt. it suspends execution of the , task and allocates resources to the event that has interrupted. provided the interrupting event is of higher priority than the one currently under execution. To facilitate on this. to the the next processor instruction. checks If for any interrupts interrupt is at pending, the end the of every processor instruction automaticallybefore

branches off to interrupted program. Saving the status (mainly the address to which the processor should return) of the interrupted program is essential for returning to interrupted program after servicing the request. This entire sequence is similar to calling a subroutine and returning from a subroutine. The only difference is that a subroutine call is programed, while the interrupt call is random from the hardware or software.

Like subroutines, servicing of interrupts can be nested. This means that at any point of time. events with higher priority can interrupt the servicing of events with lower priority. Each time a processor receives an interrupt, the interrupted program status is saved in a first-in/last-out stack and the restoration of the interrupted program takes place in reverse sequence. The RTOS manage the interrupt hierarchy.

Typical examples of hardware-generated events that can interrupt are as follows:

- Diagnostic error in functional modules
- Change of state in digital input module
- Counter becoming full in pulse input module
- Counter becoming empty in pulse output module
- Message-in buffer becoming full in communication module

- Message-out buffer becoming empty in communication module

Typical examples of software interrupt are as follows:

- Timer or time-of-day interrupts for servicing time-based event
- Programming errors, such as division by zero

Generally, all modes of task scheduling are employed in today's RTOS for different requirements.

POPULAR PUBLICATIONS

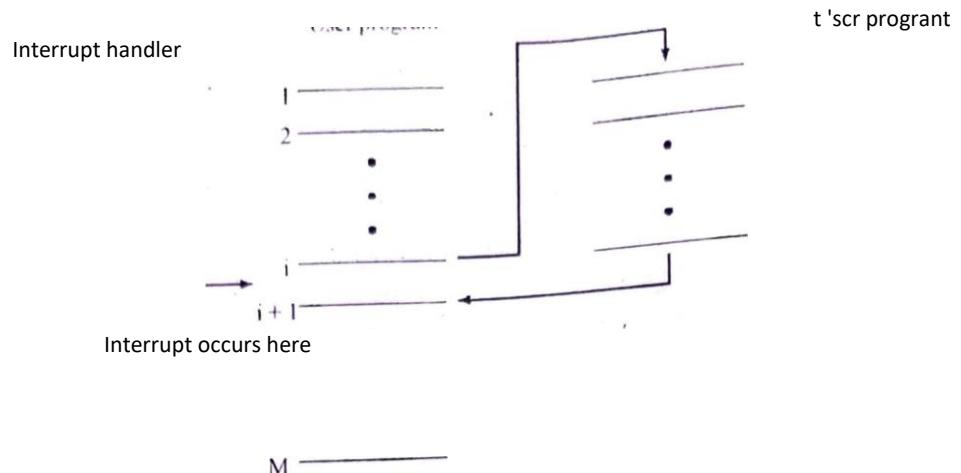


Fig: Transfer of Control via Interrupts

[WBUT 20171]

8. What is speed up? Prove that maximum speed up will be K.

Ans» er:

Speedup is the ratio of the average instruction time without pipelining to the average instruction time with pipelining.

2nd part:

Consider a 'k' segment pipeline with clock cycle time as 'Tp'. Let there be 'n' tasks to be completed in the pipelined processor. Now, the first instruction is going to take 'k' cycles to come out of the pipeline but the other 'n - 1' instructions will take only '1' cycle each.

i.e., a total of 'n - 1' cycles. So, time taken to execute 'n' instructions in a pipelined processor:

$$\begin{aligned} ET_{\text{pipeline}} &= k + n - 1 \text{ cycles} \\ &= (k + n - 1) T_p \end{aligned}$$

In the same case, for a non-pipelined processor, execution time of 'n' instructions will be:

$$ET_{\text{non-pipeline}} = n * k * T_p$$

So, speedup (S) of the pipelined processor over non-pipelined processor, when 'n' tasks are executed on the same processor is:

$S = \frac{\text{Performance of pipelined processor}}{\text{Performance of Non-pipelined processor}}$

As the performance of a processor is inversely proportional to the execution time. we have,

=non-pipeline T pipeline

$$\Rightarrow S = [n * T_p] / [(k+n-1) * T_p]$$

When the number of tasks 'n' are significantly larger than k, that is, $n \gg k$

$$S = n * k / n$$

$S = k$ where 'k' are the number of stages in the pipeline.

COMPU Write short

notes on the following.

- a) DMA controller (WBUT 20111 b) concept of hand shaking in 10 operation (WBUT 20111
pipeline hazards twBILT 20091 d) Instruction Pipeline [WBUT 20171
e) programmed I/O twBILT 20171 sns»cr:

a) I) MA Controller: Refer to Question No. 4(b) of Short Answer Type Questions.

b) Concept of hand shaking in 10 operation:

The handshaking method of asynchronous data transfer solves the problem of strobe control method. This is acknowledgment based i.e. in this mechanism, each data item being transferred is accompanied by a control signal to indicate the presence of data in the bus. Also the destination unit responds back with a control signal to acknowledge the source that the data has been received successfully. This method uses a second control signal to provide a reply to the other unit. Hence handshaking method uses two control lines.

This method is also of two types:

@Source-initiated transfer using handshaking:

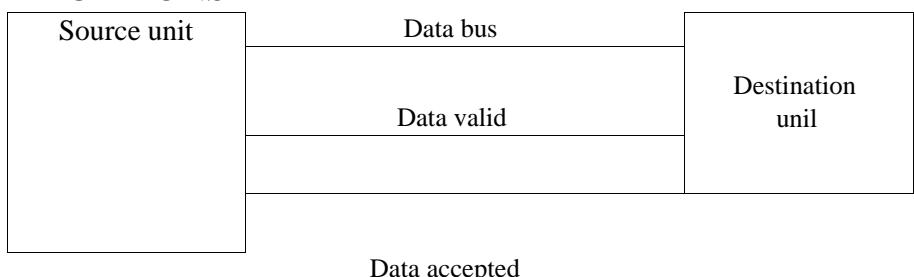
Here the source unit initiates transfer. Two handshaking lines, 'data valid' and 'data accepted' are used here.

Steps:

- (i) The source unit places data on the data bus and enables the data valid control line to indicate the destination that the data is placed on the data bus.
- (ii) The destination unit on receiving the data, enables the data accepted signal to let the source know that the data has been received successfully.
- (iii) On getting this acknowledgement from the destination, the source unit disables the data valid signal and then removes the data from the bus.
- (iv) When the destination sees that the 'data valid' signal is disabled, understanding that the acknowledgment signal has been received, it disables the 'data accepted' signal and gets ready to accept another data from the source.

Figure I (a) and (b), shows the entire process and the corresponding timing diagram.

POPULAR PUBLICATIONS



Data accepted

Fig: I (a)

PVBL'CATIONS

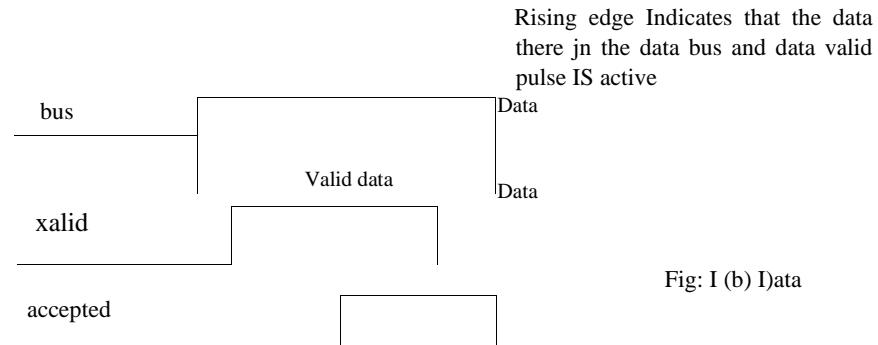


Fig: I (b) Idata

(h) Destination-initiated tran.yfer using handshaking:

Here the destination unit initiates the data transfer. In this case, the source unit places data on the data bus only after knowing that the destination unit is ready to accept dTwo handshaking lines, 'data valid' and 'ready for data' are used here. Steps:

- (i) The destination unit gets ready to accept data and thus enable the 'ready for dcontrol signal to let the source place the data on the data bus.
- (ii) The source on getting this signal, places the data on the data bus and enables the 'valid' signal to inform the placement of a valid data on the data bus.
- (iii)The destination unit then accepts the data and disables the 'ready for data' signal.
- (iv)The source understands that the data has been successfully received and thus disables

the 'data valid' signal and renooves the data fronth the data bus.

Figure 2 (a) and (b), shows the entire process and the corresponding timing diagram:

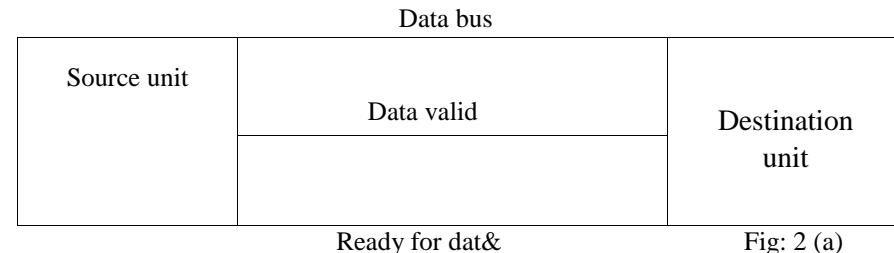


Fig: 2 (a)

POPULAR

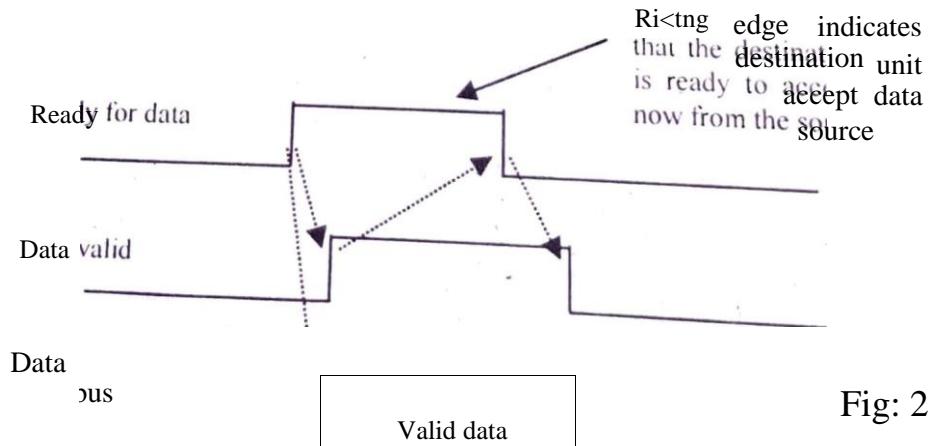


Fig: 2 (b)

COMPUTER

c) pipeline different hazards:
 Hazards (problems) that cause an instruction pipeline to deviate from its ^{The uⁿ normal} operation or delay its normal operation are the following: resource conflicts such hazards are caused when two segments tries to access memory simultaneously. suppose both instruction and data are stored in the same memory and in one segment instruction is fetched and in another segment data is fetched from the memory at a time. so this will lead to resource conflict. Using separate instruction and data memories can solve this problem.

Instruction hazards (dependency) such hazards occur when instructions tries to read or write in registers used by other instructions. These are of four types:

Read-after-read (RAR) hazards

This occurs when two instructions try to read from the same register.

Read-after-write (RAW) hazards such types of hazards are also known as data dependencies or true dependencies as these occur when an instruction needs to use the result of a previous instruction which is not yet available. Here an instruction reads a register that was written by a previous instruction. A situation called the pipeline stall or bubble occurs, as the 'read' instruction cannot proceed past the read stage of the pipeline because the result to be sent by the 'write' instruction is not available yet. So this condition delays the entire pipeline execution. Techniques to resolve such hazards are:

Hardware interlocks

Interlock is a circuit, which further delays the instruction that is stalled (for nonavailability of results) thus resolving the conflict.

Operand forwarding

This method after detecting the conflict by means of special hardware, avoids it by routing the data through special paths within the pipeline segments.

Delayed load

In such technique, compilers detect the data hazards and handle them accordingly.

Write-after-read (WAR) hazards and Write-after-write (WAW) hazards:

Such hazards, also called name dependencies, occur when an instruction's output register, has either been read or written by a previous instruction.

```

Γ 000000(AI)= · 0==0 [ 00= | = | - | 00E000 | | 0 | 00=00 | |
,==/ | -ono0 [ [ | | - 0= = | 000E00=0 | | |
=00 = | | 0 | [ [=-[ | | 0 | | 0: | 0 | 0 | |
=0=0=: | | | =0, | aa0 · =0 · | = | | | 0000 [ |
0 ~ (o)= ≈0 | | + | 00000:0 | = | 000 | 00 |
| =000 | 00 | | 0 | ~ 00 ≈ | | 000 | 1; |
=0= | 000 | | =0= [ 00= ~ | 0 | ~ 0000 ~ =0 |
≈;=0==0=0 | 0 | | 00 | [ =00= | = | 0=0, | |
0 | · 0 | 00000=00=0 | | = | | | = | 00= = [ |
; | 00 | | =00000 ~0,===: | 0 | ~ 0=0 | = | |
00 [ 00-0 | = | = | 0 | | =0000= ~ | = | 0 | =
| == ≈ | t=0= [ 0000 | =0=0= | ==0= | =
=oloJ~000=v0 | [ = | | 0= t, 0

```

alue

O | 00 | 00pupd00 | 0OA | 00 | 0 | 0 | 0= | 0= | 0= | 「00=0 | =0 | 00EOE11101j00=0 | 00 | 000 | - | 0 | 0 | 0-0 | [| 0= | 00 | 0 | =000 | | =000 | 0= | 0= | 0 | 0=000 | = | 0=0030; | 00= | . 00000 | | 0=000 | | 0=000 | |

On the other hand, if we consider the case where $\alpha = \beta$, we have $\alpha^2 = \beta^2$. Then, the expression for $\alpha - \beta$ becomes $\alpha - \beta = \alpha - \alpha = 0$. This contradicts the assumption that $\alpha \neq \beta$. Therefore, $\alpha \neq \beta$.

101ha

CC ER

needed for
instructions
results into

execution into processor registers. The execute stage (EX) executes the on the stored operand values. The last write-back stage (WB) is used to write into registers or Inetnory. All high-perforrnance computers are now equipped with

Illis pipeline.

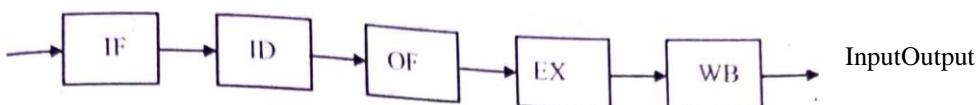


Fig: A 5-stage instruction pipeline

c) programmed I/O:

| operations will Inean a data transfer between an I/O device and memory or between an I/O device and the processor. If in any computer system I/O operations are completely controlled by the processor, then that system is said to be using 'programmed I/O'. When such a technique is used, processor executes programs that initiate, direct and terminate the I/O operations, including sensing device status, sending a read or write command and transferring the data. It is the responsibility of the processor to periodically check the status of the I/O system until it finds that the operation is complete. Let us consider the following example.

The processor's software checks each of the I/O devices every so often. During this check, the microprocessor tests to see if any device needs servicing. Figure shows the flow chart for this. This is a simple program which services I/O ports A, B and C. The routine checks the status of I/O ports in proper sequence. It first transfers the status of I/O port A into the accumulator. Then the routine block checks the contents of accumulator to see if the service request bit is set. If it is, I/O port A service routine is called. After completion of service routine for I/O port A, the polling routine moves on to test port B and the process is repeated. This test and service procedure continues until all the I/O port status registers are tested and all the I/O ports requesting service are serviced. Once this is done, the processor continues to execute the normal programs.

The routine assigns priorities to the different I/O devices. Once the routine is started, the service request bit at port A is always check'ed first. Once port A is checked, port B is check'ed. and then port C. However, the order can be changed by simply changing the routine and thus the priorities.

^When programmed I/O technique is used, processor fetches I/O related instructions from memory and issues I/O commands to I/O system to execute the instruction. The form of the instruction depends on the technique used for I/O addressing, i.e. memory Inapped I/O or mapped I/O.

POPULAR! UBLICAT/ON\$

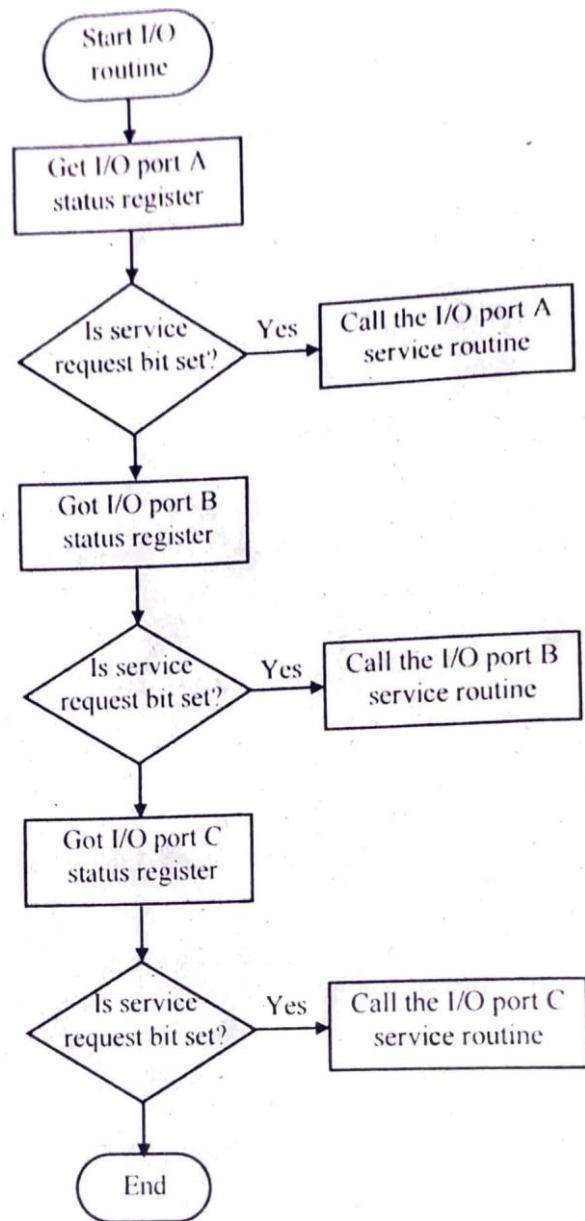


Fig: Flowchart for I/O service routine