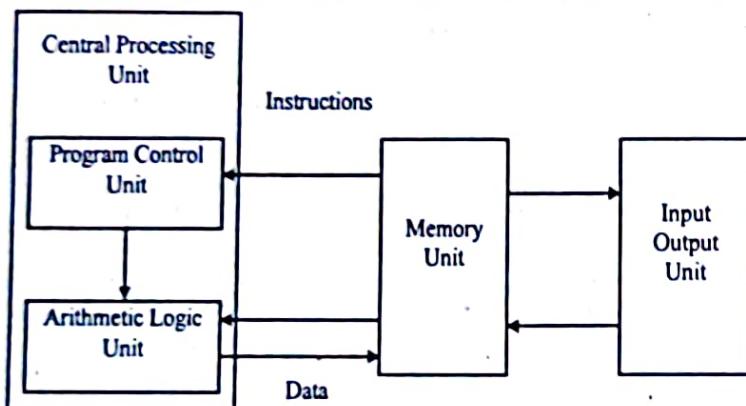


INTRODUCTION

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 programs, 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 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.

COMPUTER ORGANISATION

(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 within 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 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:

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. 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]

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

Answer: (c)

CO-3

POPULAR PUBLICATIONS

2. From a Source Code, a compiler can detect

- a) Run-time error
- b) Logical errors
- c) Syntax error
- d) None of these

Answer: (c)

[WBUT 2010]

3. How many minimum, NAND gates are required to make a flip-flop? [WBUT 2010]

- a) 4
- b) 3
- c) 2
- d) 5

Answer: (c)

4. The basic principle of a 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 register

Answer: (c)

[WBUT 2013]

5. The logic circuit in ALU is

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

Answer: (a)

[WBUT 2013]

6. The Von-Neumann bottleneck is a problem, which occurs due to

- 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)

[WBUT 2014]

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

- a) Register
- b) Encoder
- c) Decoder

Answer: (d)

[WBUT 2016]

d) Flip-flop

8. 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

Answer: (c)

[WBUT 2016]

9. The ALU makes use of

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

Answer: (a)

CO-4

10. A source program is usually in

[WBUT 2019]

- a) Assembly language
- c) High-level language

- b) Machine level language
- d) Natural language

Answer: (c)

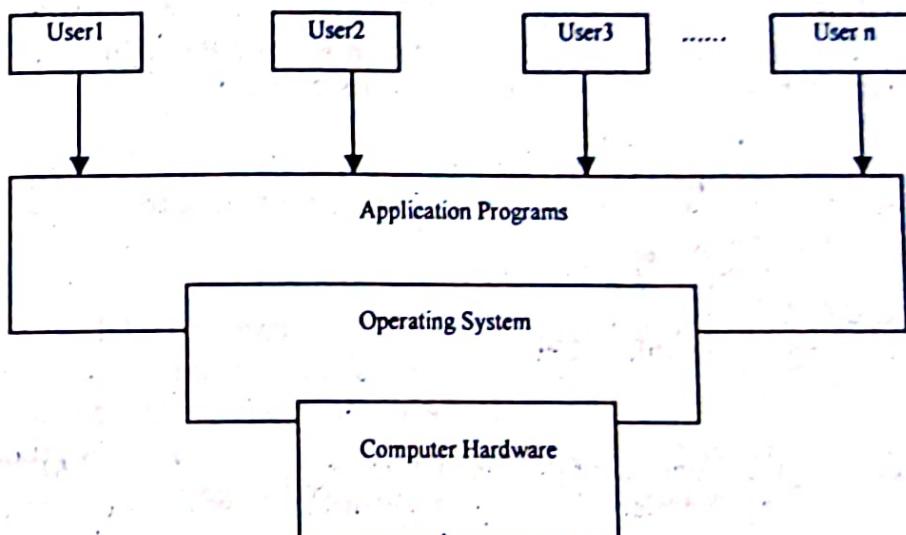
Short Answer Type Questions

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, games etc.) for the various users (like people, machines, and other computers).

(b) *O.S. provides an environment within 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.

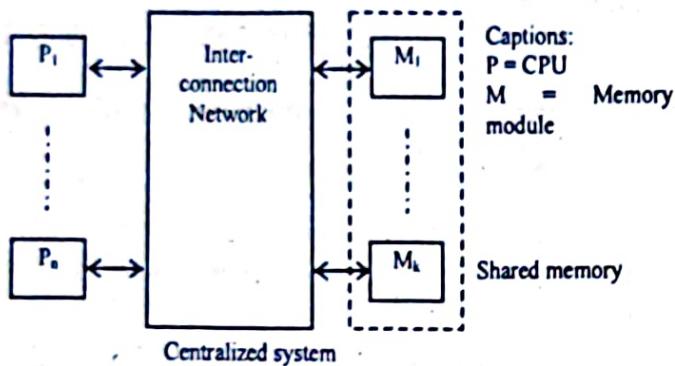
(d) O.S. acts a control program

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.

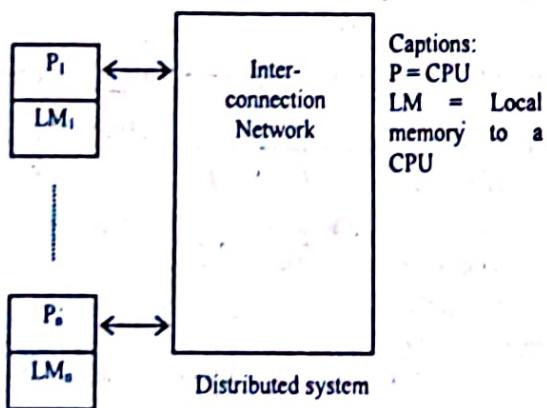
2. Compare between centralized and distributed architecture. Which is the best architecture among them and why? [WBUT 2014]

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 ORGANISATION

3. Explain the role of operating system in a computer system.

[WBUT 2017]

Answer:

An operating system has three main functions:

- (1) manage the computer's resources, such as the central processing unit, memory, disk drives, and printers,
- (2) establish a user interface, and
- (3) execute and provide services for applications software.

Long Answer Type Questions

1. 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]

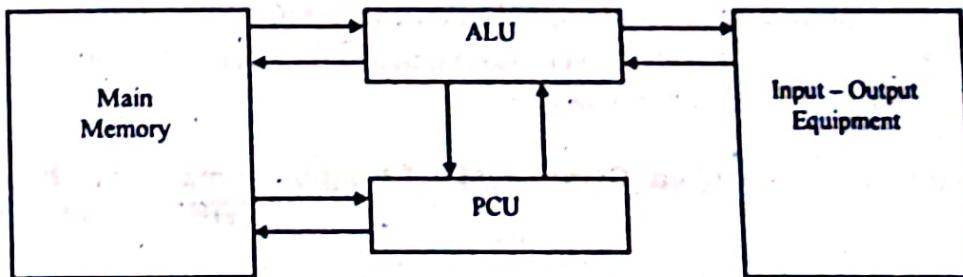
Answer:

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.

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.

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

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.

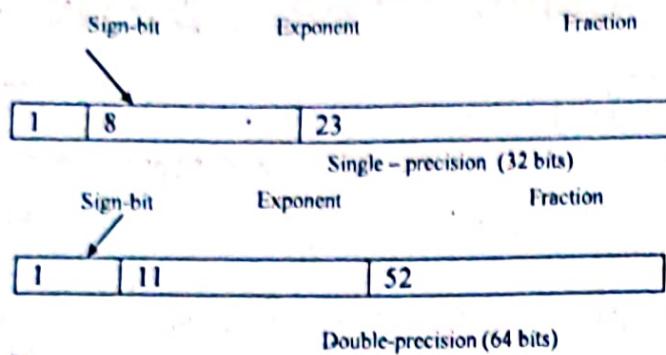


Fig. 3 IEEE 754 Floating-point formats .

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

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

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 $P_i = A_i B_i$, 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?

[WBUT 2003, 2007, 2008, 2009]

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

Answer: (c)

CO-12

3. Subtractor can be implemented using

[WBUT 2006, 2012, 2015]

- a) adder
- c) both (a) & (b)

- b) complementer
- d) none of these

Answer: (c)

4. Maximum n bit 2's complement number is [WBUT 2007, 2009, 2015, 2018, 2019]

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

Answer: (c)

5. Adding 01101101_2 to 10100010_2 in 8-bit 2's complement binary will cause an overflow:

b) False

[WBUT 2007, 2009]

- a) True

Answer: (b)

6. The conversion of $(FAFAFA)_{16}$ into octal form is [WBUT 2008, 2013]

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

- d) 76727672

Answer: (b)

7. Which logic gate has the highest speed?

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

[WBUT 2009]

- d) DTL

Answer: (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 $(FAFAFB)_{16}$ 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
- c) Combinational cum sequential
- b) Entirely sequential
- d) None of these

Answer: (c)

12. Equivalent hexadecimal of $(76575372)_8$ will be

[WBUT 2011]

- a) FAFAFF
- b) FAFAFA
- c) FFFAAA

- d) FAAFAF

Answer: (b)

CO-13

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) 0000011 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) 01000 1111 b) 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 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 [WBUT 2016]
..... number of locations.
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) $2^{(N-1)}$ d) 2^{N-1}

Answer: (a)

Short Answer Type Questions

1. Explain the relative advantages & disadvantages of parallel adder over serial adder.
 [WBUT 2002, 2003, 2006, 2012]

Answer:

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

Disadvantages of parallel adder: main disadvantages of parallel adder is the propagation delay of carry bit one full adder to next higher position full adder. Sufficient time must be allowed so that carry bit 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 {-1,1} for the quotient digits instead of {0,1}.

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 drawback of the restoring division technique can be avoided in the non-restoring division technique.

3. What are guard bits?

[WBUT 2004, 2008, 2009, 2011, 2015]

OR,

What is the necessity of guard bits?

[WBUT 2017]

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.

CO-15

4. a) Briefly explain the IEEE 754 standard format for floating point representation.
[WBUT 2007, 2009, 2017, 2019]

Answer:

All modern computers use the floating-point representation that was specified in IEEE standard 754.

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

31	30 29 ... 24 23	22 21 1	0
S	E	M	
Sign bit	Exponent	Mantissa	

Bit 31 → sign bit

Bit 30 to 23 → exponent

22 to 0 → 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 means 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: NaN or Not a Number is the symbol for any invalid operation result. For example, dividing 0 by 0 or subtracting an infinite value from another would produce invalid results, which would be represented by NaN. A NaN result would allow an user to re-check a decision and figure out the problem.

Infinity: An exponent of all 1s and a fraction of all 0s are used to denote the values of +infinity and -infinity. The sign bit is used to distinguish between negative and positive infinity.

5. Write $+7_{10}$ in IEEE 754 floating point representation in double precision.

[WBUT 2009]

OR,

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

[WBUT 2015]

Answer:

Bit 31 (sign bit) for the given number is 0 (positive sign).

Bits 23 to 30 (exponent field): 1000000.

Bits 0 to 22 (significant): 1.1100000000000000000000.

Hence, converting to hex, the result becomes 40E00000.

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 must propagate or ripple through this cascade, this configuration is called an *Ripple Carry Adder*.

A fast adder circuit must speed up the generation of the carry signals. 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.

7. Using Booth's algorithm multiply (-12) and (+6).

[WBUT 2011]

OR,

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

[WBUT 2017]

Answer:

The binary representation of 00110 = 00110 (multiplier)

The binary representation of 12 = 01100

The binary representation of -12 = 2's complement of 01100 = 10100 (multiplicand)

$Q_n Q_{n+1}$	$BR = 01100 \quad \overline{BR} + 1 = 10100$	AC	QR	Q_{n+1}	SC
	Initial	0 0 0 0 0	00110	0	1 0 1
0 0	ashr AC, QR (Including Q_{n+1})	10000	00011	0	
1 0	Subtract BR	00100	00011	0	1 0 0
		10010	00001	1	0 1 1
1 1	ashr AC, QR (Including Q_{n+1})	11001	00000	1	0 1 0
0 1	Add BR	00101			
		10010	10000	0	0 0 1
0 0	ashr AC, QR (Including Q_{n+1})	11001	01000	0	0 0 0

The answer is: 1100101000

8. Apply Booth's algorithm to multiply the two numbers $(6)_{10}$ and $(-9)_{10}$. Assume the multiplicand and multiplier to be 5 bits each.

[WBUT 2012]

Answer:

$$\begin{array}{r}
 A \quad \quad \quad 0 \ 0 \ 1 \ 1 \ 0 \quad \quad \quad 6 \\
 X \quad \times 1 \ 0 \ 1 \ 1 \ 1 \quad \quad \quad -9 \\
 Y \quad \quad \quad -1 \ 1 \ 0 \ 0 \ -1 \quad \quad \quad \text{recoded multiplier} \\
 \hline
 \text{Add-A} \quad + 1 \ 1 \ 0 \ 1 \ 0 \\
 \text{Shift} \quad \quad 1 \ 1 \ 1 \ 0 \ 1 \ 0 \\
 \text{Shift Only} \quad 1 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0 \\
 \text{Shift Only} \quad 1 \ 1 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0 \\
 \text{Add A} \quad + 0 \ 0 \ 1 \ 1 \ 0 \\
 \hline
 \quad \quad \quad 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0 \\
 \text{Shift} \quad \quad 0 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0 \\
 \text{Add-A} \quad + 1 \ 1 \ 0 \ 1 \ 0 \\
 \hline
 \quad \quad \quad 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \\
 \text{Shift} \quad \quad 1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ -54
 \end{array}$$

9. Represent (-9.5) in 64 bit IEEE floating point representation.

[WBUT 2013]

Answer:

The IEEE-754 format is as follows:

Bit 63: sign bit

Next 11-bit biased exponent represented in excess-1023 form

Next 52-bit normalized mantissa (magnitude), where the decimal point is assumed to lie just on the right of the most significant 1 bit in the real number (integer + fraction).

Now, $-9.5 = -1001.1 = -1.0011 \times 2$ to the power +3

So, bit 63 = 1

Exponent part = $1023 + 3 = 1026D = 100\ 0000\ 0010B$

Mantissa part = 00110....0 (0011 followed by 48 0's)

Hence the representation is C02300000000000H.

10. Explain IEEE single precision formats for representing – 10.5.

[WBUT 2014]

Answer:

$$10.5 = 1010.10_2 = 1.01010 \times 2^3$$

$$s = 1 e = 3 + 127 = 130 = 10000010 f = 01010$$

The single-precision representation is:

$$1\ 1000010\ 010100000000000000000000$$

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

[WBUT 2015]

Answer:

The given number in IEEE 32-bit format is 40400000_{16}

$$= 0100\ 0000\ 0100\ 0000\ 0000\ 0000\ 0000_2$$

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

Next higher order 8-bit indicates the biased exponent (E') and it is $(1000\ 0000)_2 = 128$

$$\text{Therefore, the original exponent } E = E' - 127 = 128 - 127 = 1$$

The leading bit in mantissa (after binary point) is 1, so the actual mantissa is $(1.1)_2$

$$\text{Thus, the decimal number is } +(1.1)_2 \times 2^1 = +(11)_2 = +3_{10}$$

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 worst 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. Use restoring method to divide 10100011 by 1011.

[WBUT 2016]

Answer:

(Unsigned numbers division)

Divide 163 by 11 using restoring division method.

Dividend, Q = 163 = 10100011

COMPUTER ORGANISATION

Divisor, $M = 11 = 00001011$, $M_{1C} = 11110101$

Iteration	Step/Action	Accumulator (A)	Dividend/Quotient (Q)	Divisor/Remark (M)
0	Initial values	00000000	10100011	00001011
1	Shift left A, Q Subtract A-M	00000001 11110101	0100011_	
	Restore A+M	11110110 00001011	01000110	$S_A = 1; Q_0 = 0$
2	Shift left A, Q Subtract, M	00000001 00000010 11110101	01000110 1000110_	
	Restore A+M	11110111 00001011	10001100	$S_A = 1; Q_1 = 0$
3	Shift left A, Q Subtract, M	000000010 00000101 11110101	10001100 0001100_	
	Restore A+M	11110110 00001011 00000101	00011000	$S_A = 1; Q_2 = 0$
4	Shift left Sub: A-M	00001010 11110101	0011000_	
	Restore A+M	11111111 00001011	00110000	$S_A = 1; Q_3 = 0$
5	Shift left Sub: A-M	00010100 11110101	0110000_	
		00001001	01100001	$S_A = 1; Q_4 = 0$
6	Shift left Sub: A-M	00010010 11110101	1100001_	
		00000111	11000011	$S_A = 1; Q_5 = 0$
7	Shift left Sub: A-M	00001111 11110101	10000111	
		00000100	10000111	$S_A = 1; Q_6 = 0$
8	Shift left Sub: A-M	00001001 11110101	0000111_	
	Restore A-M	11111110 00001011 00001001	00001110 00001110	$S_A = 1; Q_7 = 0$
		00001001	00001110	

Remainder = 9

Quotient = 14

Hence $163 \div 11$ gives, $Q = 14$ and $R = 9$

POPULAR PUBLICATIONS

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

[WBUT 2016]

OR,

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

[WBUT 2019]

Answer:

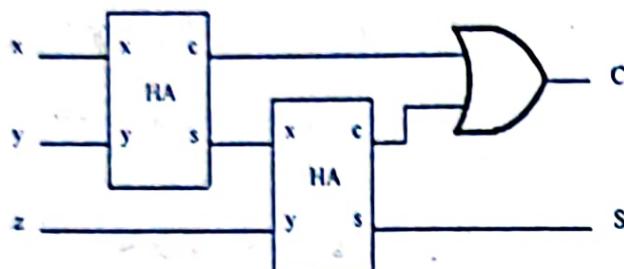


Fig: Block diagram of full-adder implementation via a pair of half-adders

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 sum 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 half-adders produces a carry-out of 1. This corresponds to the OR of the two partial carry bits.

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

[WBUT 2017]

Answer:

A system 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 (C_{in}) and a carryout (C_{out}) bit, and the adders are connected by connecting C_{out} on step k to C_{in} 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 we 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 = $O(n^2)$ which is much larger

COMPUTER ORGANISATION

than no. of gates used in carry propagation adder which is $O(n)$. So cost of carry lookahead adder is obviously larger.

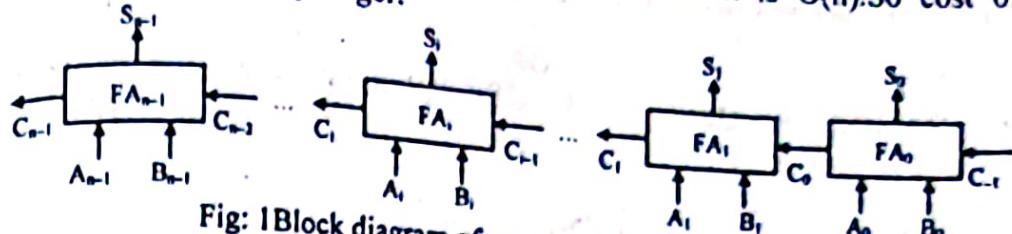


Fig: 1 Block diagram of an n-bit ripple-carry adder

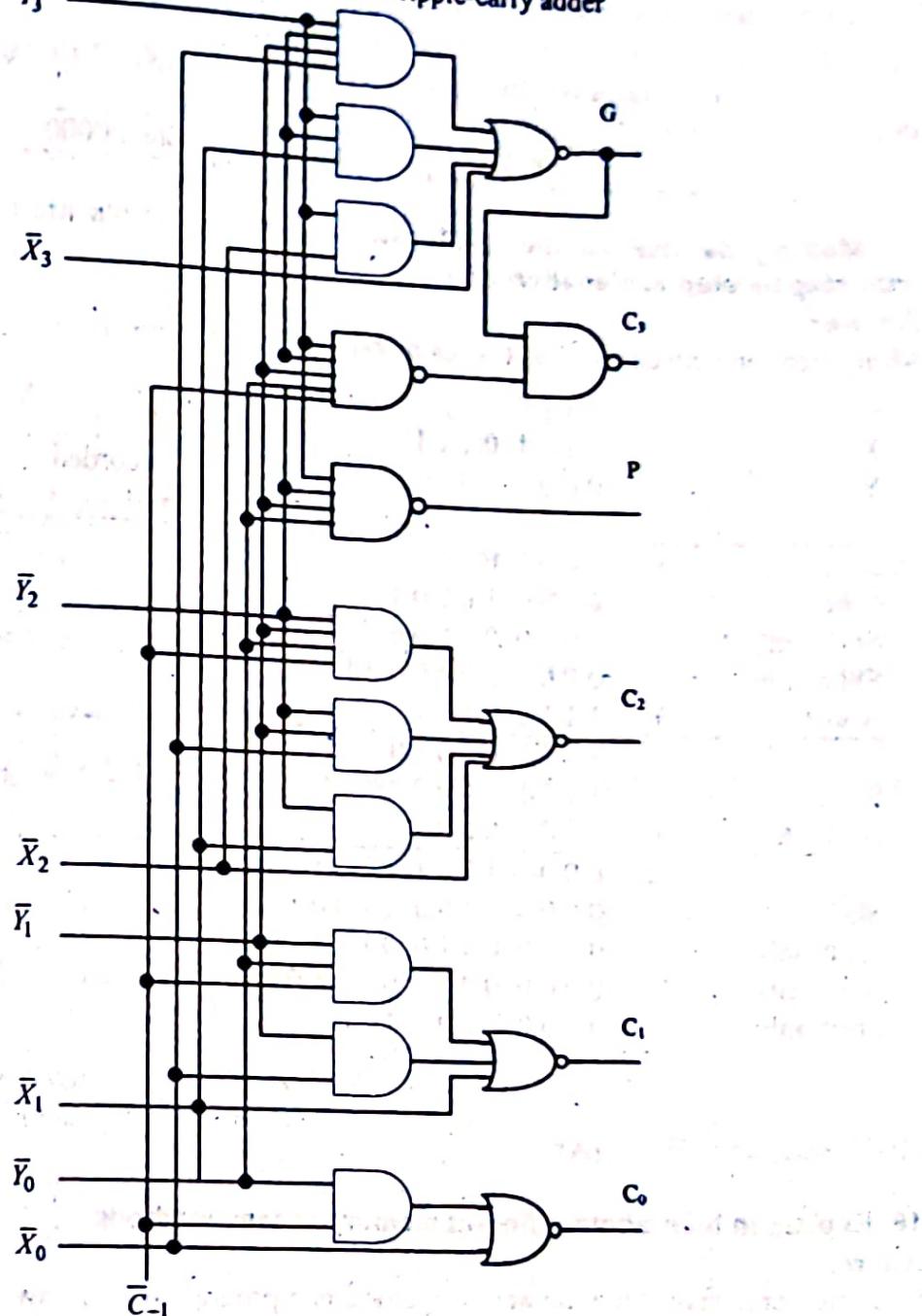


Fig: 2 Logic diagram of a 4-bit carry look ahead circuit

POPULAR PUBLICATIONS

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

[WBUT 2017]

OR,

Represent the decimal value - 7.5 in IEEE-754 single precision floating point format.

[WBUT 2018]

Answer:

The decimal number - 7.5 = - 111.1 in binary = - 1.111 × 2²

The 23-bit mantissa M = 0.111000 000000 000000 0000

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

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

1	10000001	111000 000000 000000 00000
---	----------	----------------------------

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

[WBUT 2018]

Answer:

Multiplication between -17 and -9 using booth's Algorithm:

A	1110 1111	-17
X	× 1111 0111	-9
Y	000-1 100-1	Recorded multiplier
<hr/>		
Add - A	+ 0001 0001	
shift	0000 10001	
Shift only	0000 010001	
Shift only	0000 0010001	
Add A	+ 1110 1111	
	1111 0001001	
Shift	1111 10001001	
Add - A	+ 0001 0001	
	0000 10011001	
Shift	0000 010011001	
Shift only	0000 0010011001	
Shift only	0000 00010011001	
Shift only	0000 000010011001	

$$(10011001)_{10} = 153 \quad (\text{Ans.})$$

18. Explain in brief about different memory access methods.

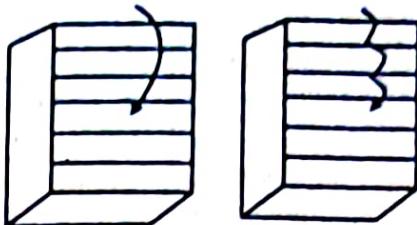
[WBUT 2018]

Answer:

In computer organisation, an access method is a program or a hardware mechanism that moves data between the computer and an outlying device such as a hard disk or a display terminals.

CO-22

There are 2 types of access method. They are-
i) Random access and ii) Sequential access.



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

Random access is better than sequential access.

19. Represent the decimal value-12.5 in IEEE single precision format. [WBUT 2019]

Answer:

$$12.5 = 1100.10_2 = 1.1001 \times 2^3$$

$$s = 1 e = 3 + 127 = 130 = 10000010 f = 01010$$

The single-precision representation is:

1 10000010 0101000000000000000000000

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. [WBUT 2017]

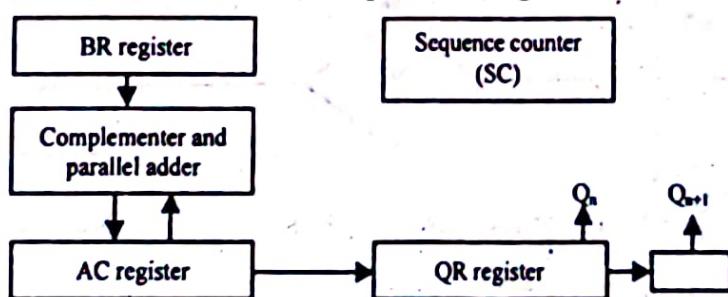
Illustrate this with an example by multiplying $(-9) \times (-13)$. [WBUT 2010]

Answer:

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.

Hardware configuration for Booth's multiplication algorithm

Diagram



The figure shows the hardware implementation for Booth's algorithm.

- (viii) Indirect Address Mode
- (ix) Relative Address Mode
 - (a) PC (i.e. Program 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 Type Questions

1. Instruction cycle is
 - a) fetch-decode-execution
 - b) decode-fetch-execution
 - c) fetch-execution-decode
 - d) none of these

Answer: (a)
2. Micro instructions are kept in
 - 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?
 - a) Immediate
 - b) Register
 - c) Direct
 - d) Register Indirect

Answer: (b)
4. Which of the following addressing modes is used in instruction RAL
 - a) immediate
 - b) implied
 - c) direct
 - d) register

Answer: (b)
5. Which of the following address modes is used in the instruction 'POP B'?
 - a) immediate
 - b) register
 - c) direct
 - d) register indirect

Answer: (d)
6. A computer uses words of size 32-bit. The instruction
 - 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
 - a) 1
 - b) 2
 - c) 3
 - d) none of these

Answer: (a)
8. In the processor, the address of the next instruction to be executed is stored in
 - a) stack pointer register
 - b) index register
 - c) base register
 - d) program counter register

Answer: (c)

9. A stack-organised computer uses instruction of [WBUT 2016]
a) Indirect addressing
b) Two addressing
c) Zero addressing
d) Index addressing

Answer: (c)

10. When performing a looping operation, the instruction gets stored in the [WBUT 2017]
a) Registers
b) Cache
c) System heap
d) System stack

Answer: (b)

11. In case of Zero-address Instruction method the operands are stored in [WBUT 2017]
a) Registers b) Accumulators c) Stack d) Cache

Answer: (c)

12. The addressing mode(s), which uses the PC instead of a general purpose [WBUT 2017]
register is
a) Indexed with offset
b) Relative
c) Direct
d) Both (a) and (b)

Answer: (b)

13. How many memory locations can be addressed by a 32-bit computer? [WBUT 2018]
a) 64 KB b) 32 KB c) 4 GB d) 4 MB

Answer: (a)

14. The addressing mode of an instruction is resolved by [WBUT 2018]
a) ALU b) DMA controller c) CU d) program

Answer: (b)

15. The addressing mode, where you directly specify the operand value is [WBUT 2019]
a) Immediate b) Direct c) Definite d) Relative

Answer: (a)

16. How is the effective address of base-register calculated? [WBUT 2019]
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)

Short Answer Type Questions

1. Explain the difference between three-address, two-address, one-address instructions & zero-address instruction with suitable examples.

[WBUT 2007, 2011, 2013]

OR,

Evaluate the following arithmetic expression into three-address, two-address, one-address, zero-address instruction format.

[WBUT 2016, 2012]

$$X = (A + B) * C$$

Answer:

Three Address Instructions:

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:

$$\text{ADD } T, A, B \Rightarrow T \leftarrow A + B$$

$$\text{MULTIPLY } X, C, T \Rightarrow X \leftarrow C * T$$

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:

$$\text{MOVE } T, A \Rightarrow T \leftarrow A$$

$$\text{ADD } T, B \Rightarrow T \leftarrow T + B$$

$$\text{MULTIPLY } C, T \Rightarrow X \leftarrow C * T$$

Use:

Two-address instructions are used in all commercial computers.

One Address Instruction:

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:

LOAD A \Rightarrow transfer certain memory content to accumulator

ADD B \Rightarrow AC \leftarrow AC + B

STORE T \Rightarrow transfer AC content to memory location T

LOAD C \Rightarrow transfer C to accumulator

MULTIPLY T \Rightarrow AC \leftarrow AC * T

STORE X \Rightarrow transfer result to memory location X

Use:
In Intel 8085 machines.

Zero Address Instruction:

Zero-address Instructions do not contain any explicit addresses (except for PUSH and POP 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. Evaluating $X = (A+B)*C$ in a two-address machine will result to:

```
PUSH A
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, 2019]

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 (e.g. → add R ₁ , R ₂ , R ₃)	ii) One or two register operands allowed per instruction (e.g. → add R ₁ , R ₂)
iii) Parameter passing through efficient on chip register windows.	iii) Parameter passing through inefficient off-chip memory.
iv) Single cycle instructions (except for load and store).	iv) Multiple cycle instruction.
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 memory.
xi) Few addressing modes.	xi) Many addressing modes.

4. What are the advantages of relative addressing mode over direct address mode?
[WBUT 2011, 2017]

Answer:

In such modes 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 mode 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:

i) Memory has $256K$ words = 2^{18} 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 18bits address input in memory.

7. Explain Indirect address mode. How is the effective address calculated in this case?
[WBUT 2016]

Answer:

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

8. Explain with example: Register Direct, Register Indirect and Base register addressing mode.
[WBUT 2018]

Answer:

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

9. Evaluate the following arithmetic expression into (i) three address, (ii) two addresses, (iii) one address and (iv) zero address instruction format $X = (A + B) * (C + D)$.
[WBUT 2019]

Answer:

(i) Three-address machine:

ADDITION: ADD X, A, B $\Rightarrow X \leftarrow A + B$

ADDITION: ADD Z, C, D $\Rightarrow Z \leftarrow C + D$

MULTIPLY: MUL Z, X, Z $\Rightarrow Z \leftarrow Z * X$

(ii) Two-address machine:

MOV	R1, A	R1 = M[A]
ADD	R1, B	R1 = R1 + M[B]
MOV	R2, C	R2 = C
ADD	R2, D	R2 = R2 + D
MUL	R1, R2	R1 = R1 * R2
MOV	X, R1	M[X] = R1

(iii) One-address machine:

LOAD A	$\Rightarrow AC \leftarrow A$
ADD B	$\Rightarrow AC \leftarrow AC + B$
STOR X	$\Rightarrow X \leftarrow AC$
LOAD C	$\Rightarrow AC \leftarrow C$
ADD D	$\Rightarrow AC \leftarrow AC + D$
MUL X	$\Rightarrow AC \leftarrow X * AC$

(iv) Zero-address machine:

PUSH A	TOP = A
PUSH B	TOP = B
ADD	TOP = A+B
PUSH C	TOP = C
PUSH D	TOP = D
ADD	TOP = C+D
MUL	TOP = (C+D)*(A+B)
POP X	M[X] = TOP

Memory Hierarchy: The block diagram of memory hierarchy is as follows:

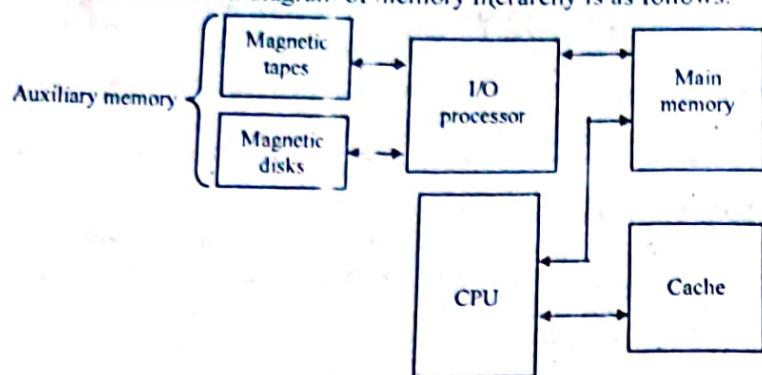
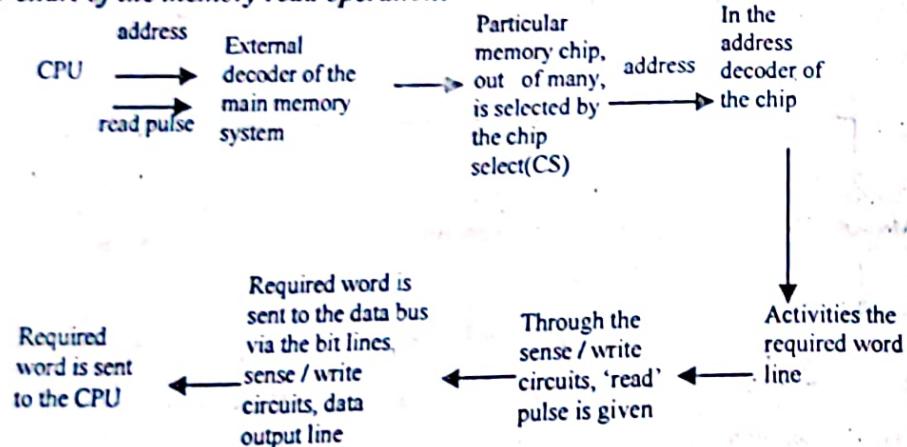


Fig: Memory hierarchy in a computer system

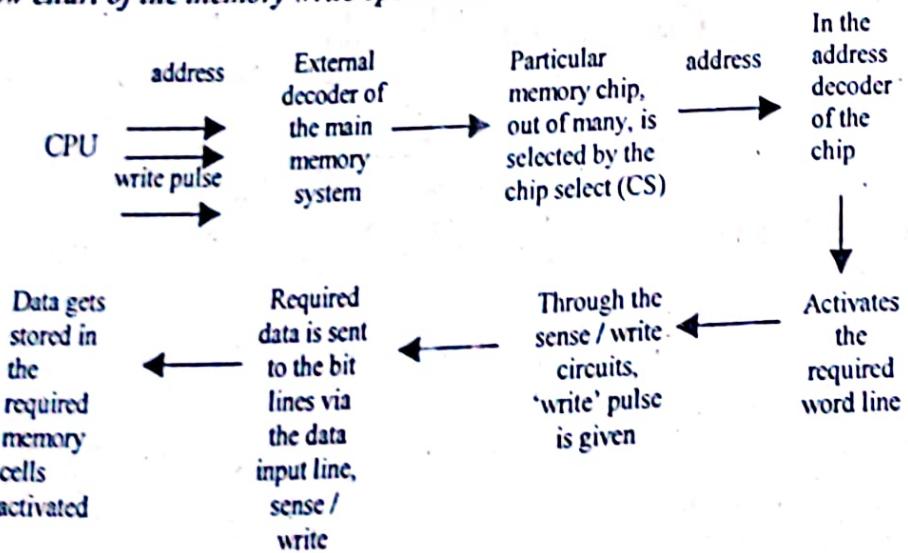
Factors on which Memory Hierarchy depends: There are various factors on which the basic hierarchy of memory depends. These are cost, storage capacity (size), and speed and access time.

- **Memory read/write access:**

Flow chart of the memory read operation:



Flow chart of the memory write operation:



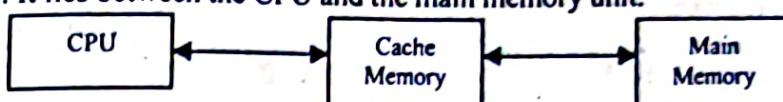
• **Static & Dynamic RAM:**

(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.

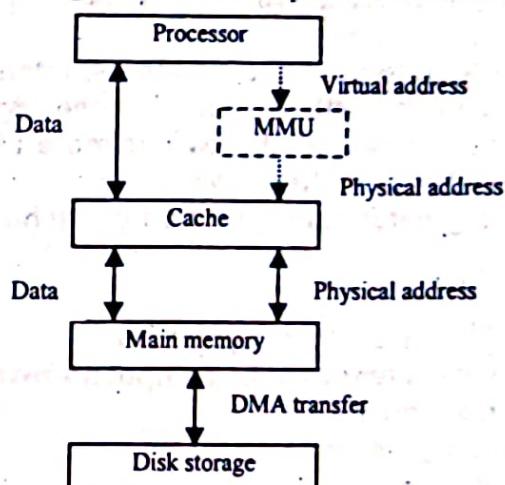


Fig: Virtual memory organization

Multiple Choice Type Questions

1. The technique of placing software in a ROM semiconductor chip is called

[WBUT 2003, 2008]

a) PROM

b) EPROM

c) FIRMWARE

d) Microprocessor

Answer: (c)

2. Cache memory

a) increases performance
c) machine cycle increases

[WBUT 2006, 2008, 2011, 2013]

b) reduces performance
d) none of these

Answer: (a)

POPULAR PUBLICATIONS

3. Associative memory is a

- a) very cheap memory
- b) content addressable memory

Answer: (c)

[WBUT 2006, 2008, 2011, 2013]

- b) pointer addressable memory
- d) slow memory

Answer: (c)

4. How many RAM chips of size (256 K × 1 bit) are required to build 1 M Byte memory?

- a) 8
- b) 10

- c) 24
- d) 32

Answer: (d)

[WBUT 2006, 2009, 2015, 2018]

- a) 1024
- b) 5

- c) 10

[WBUT 2007, 2011, 2013]

- d) None of these

Answer: (c)

6. The principle of locality justified the use of

[WBUT 2007, 2012, 2015, 2018]

- a) Interrupt
- b) DMA

- b) Polling
- d) Cache memory

Answer: (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

Answer: (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 0's in memory

Answer: (a)

9. Which one does not possess the characteristics of a memory element?

- a) A toggle switch
- c) An AND gate

- b) A lamp
- d) None of these

[WBUT 2010]

Answer: (c)

10. Data from memory location after fetching is deposited by memory in

- a) MAR
- c) IR

- b) MDR
- d) Status Register

[WBUT 2010]

Answer: (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? [WBUT 2010]

- a) $2^{36}, 2^{24}$ b) $2^{12}, 2^{36}$

- c) $2^{24}, 2^{34}$

- d) $2^{34}, 2^{38}$

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 is [WBUT 2012]

- a) 1 K b) 2 K c) 1 M

- d) none of these

Answer: (c)

14. Periodic refreshing is needed [WBUT 2012]
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 Instruction 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)

20. The user's view of memory is supported by
a) paging b) segmentation c) both

[WBUT 2014]
d) none of these

Answer: (a)

21. The largest delay in accessing data on disk is due to
a) seek time b) rotation time
c) data transfer time d) none of these

Answer: (a)

22. Address of memory location for fetching data needs to be deposited in memory
in
a) MAR c) MBR c) IR d) status register

Answer: (a)

23. Size of virtual memory is equivalent to the size of
a) hard disk b) CPU c) floppy disk

[WBUT 2014]
d) none of these

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

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
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?

a) 1 b) 2 c) 3

d) 4

Answer: (d)

27. A computer's memory is composed of 8K words of 32 bits each. How many bits are required for memory address if the smallest addressable memory unit is a word?

a) 13 b) 8 c) 10

d) 6

Answer: (a)

COMPUTER ORGANISATION

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

Answer: (b)

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

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

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

Answer: (b)

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

- a) Memory access time
- b) Memory search time
- c) Memory cycle time
- d) Instruction delay

Answer: (c)

32. A 24 bit address generates an address space of locations

[WBUT 2017, 2019]

- a) 1024
- b) 4096
- c) 2^{48}
- 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

- [WBUT 2019]
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) 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 n-address bit processor?

a) 0 b) 2^n

c) infinite d) decided by manufacturer

Answer: (c)

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.
- 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 $1 \text{ MB} = 2^{20}$ 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 = $2^{28} / 2^7 = 2^{21}$. 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

21 7

Main Memory Address

Size of the sub-fields for direct mapping cache scheme::

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 $28 - (13 + 7) = 8$.

Block Tag Word

13 8 7

Main Memory Address

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^{13} .
So number of sets in the cache memory are $2^{13} / 8 = 2^{13} / 2^3 = 2^{10}$.

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 = $28 - (10 + 7) = 11$.

Set	Tag	Word	Main Memory Address
10	11	7	

2. What is dirty bit?

[WBUT 2006, 2010]

Answer:

To cope up with the storage capacity of the main memory, pages are swapped in and out of the main memory and the secondary memory. When a particular page is required in the main memory and if that page is not there in the main memory it is swapped in from the secondary storage area. Dirty bits are used in the page table to keep track of individual pages, whether the particular page is modified ever since it is brought in the main memory. Whenever the contents of a page is modified (i.e. something is written on the page), its respective dirty bit is set. If a particular page needs to be swapped out of the main 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 2006]

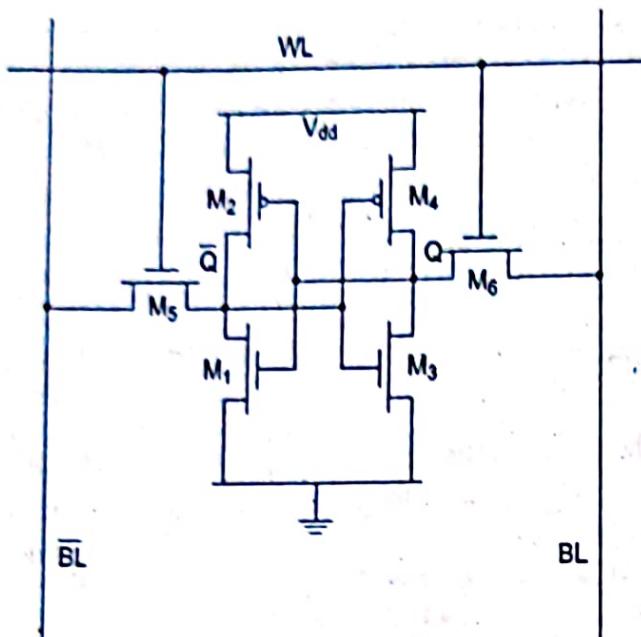
OR,

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 remanence, but is still *volatile* in the conventional sense that data is eventually lost when 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 memory 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 more than one (read and/or write) port, which may be useful in certain types of video memory and register files implemented with multi ported SRAM circuitry.

Generally, the fewer transistors needed per cell, the smaller each cell can be. Since the cost of processing a silicon wafer is relatively fixed, using smaller cells and so packing more bits on one wafer reduces the cost per bit of memory.

Memory cells that use fewer than 6 transistors are possible — but such 3T or 1T cells are DRAM, not SRAM.

Access to the cell is enabled by the word line (WL in figure) which controls the two access transistors M₅ and M₆ which, in turn, control whether the cell should be connected to the bit lines: BL and BL̄. They are used to transfer data for both read and write operations. Although it is not strictly necessary to have two bit lines, both the signal and its inverse are typically provided in order to improve noise margins.

During read accesses, the bit lines are actively driven high and low by the inverters in the SRAM cell. This improves SRAM bandwidth compared to DRAMs—in a DRAM, the bit line is connected to storage capacitors and charge sharing causes the bitline to swing upwards or downwards. The symmetric structure of SRAMs also allows for differential signaling, which makes small voltage swings more easily detectable. Another difference with DRAM that contributes to making SRAM faster is that commercial chips accept all address bits at a time. By comparison, commodity DRAMs have the address multiplexed 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 m address lines and n data lines is 2^m words, or $2^m \times n$ bits.

4. What is concept of virtual memory.

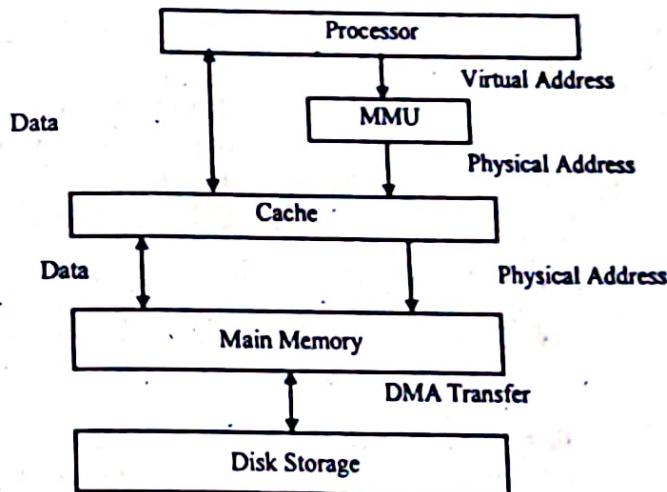
[WBUT 2007]

OR,

What is virtual memory? Why is it called virtual? [WBUT 2008, 2010, 2012, 2014]

Answer:

In a memory hierarchy system, program and data are first stored in auxiliary memory. Portions of a program or data are brought into main memory as they are needed by the CPU. Virtual memory is a concept used in some large computer systems that permit the user to construct programs as though a large memory space is available, equal to the totality of auxiliary memory. Each address that is referenced by the CPU goes through an address mapping from the so-called virtual address to a physical address in main memory. 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. A virtual memory system provides a mechanism for translating program-generated address into correct main 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 mapping table.



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 = $4 * 10 \text{ mm} / .1 \text{ mm} = 400$.

Storage capacity of the entire disk drive = number of surfaces in the drive * number of tracks / surface * number of sectors / track * number of bytes / sector = $8 * 400 * 20 * 4000 \text{ bytes} = 256 \text{ MB}$

Number of bytes transferred from each surface during one revolution of the disk = number of bytes / track = $20 * 4000 \text{ bytes} = 80,000 \text{ bytes}$.

Time per revolution = $60 / 3600 \text{ sec} = 1 / 60 \text{ sec}$

So, data transfer rate from each surface of the disk drive = $60 * 80 \text{ kb} = 4.8 \text{ mbytes / sec.}$

6. What do you mean by Stack memory?

OR,

Explain stack based CPU.

[WBUT 2012]

[WBUT 2018]

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 implemented in a random-access memory attached to a CPU. This is known as memory stack. The stack in digital computers is essentially a memory unit with an address register that can count only. The register that holds the address for the stack is called the stack pointer (SP) because its value always points at the top item of the stack.

The two basic operations that can be performed on a stack are the insertion and deletion of items. In the insertion operation, which is also known as push, information is stored in the top of stack (TOS) position in the stack and in the deletion operation, known as pop, information from 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 $\text{FULL} \leftarrow 1$ (FULL is a flip-flop which is 1 when the stack is full and EMTY is a flip-flop which is 1 when the stack is empty). The first element is pushed in the stack at the $\text{SP} \leftarrow 1$ location i.e. the first location. So when $\text{SP} \leftarrow 0$, it means that the stack is full and hence FULL is 1. So EMTY is 0 i.e. stack not empty.

The steps for the PUSH operation are as follows:

Initially, SP is cleared to 0, EMPTY is set to 1 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 $\text{FULL}=0$), a new item is inserted with a PUSH operation. The PUSH operation is implemented with the following sequence of microoperations:

- | | | |
|------------------------------------------------------------|---------------|-------------------------------------------------|
| $\text{SP} \leftarrow \text{SP} + 1$ | \Rightarrow | Stack pointer is incremented. |
| $\text{M}[\text{SP}] \leftarrow \text{DR}$ | \Rightarrow | Item, from data register, is stored on the TOS. |
| If ($\text{SP} = 0$) then ($\text{FULL} \leftarrow 1$) | \Rightarrow | To check if stack is full. |
| $\text{EMTY} \leftarrow 0$ | \Rightarrow | Mark the stack is not empty. |

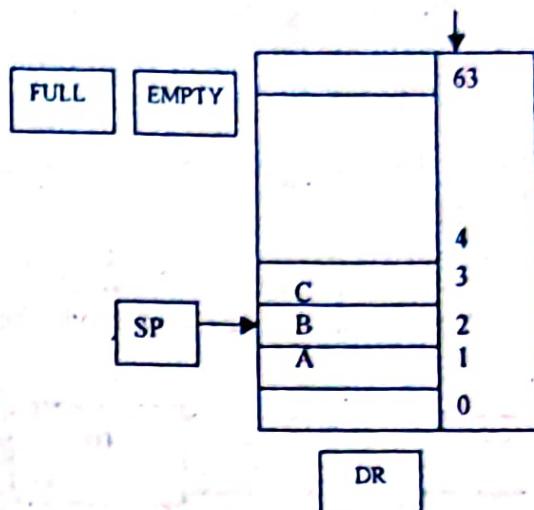
The steps of POP operation are as follows:

A new item is deleted from the stack if the stack is not empty (If $\text{EMPTY}=0$). The POP operation consists of the following sequence of microoperations:

- | | | |
|------------------------------------------------------------|---------------|----------------------------------------------------------|
| $\text{DR} \leftarrow \text{M}[\text{SP}]$ | \Rightarrow | Item from TOS is popped and stored in the data register. |
| $\text{SP} \leftarrow \text{SP} - 1$ | \Rightarrow | Stack pointer is decremented. |
| If ($\text{SP} = 0$) then ($\text{EMTY} \leftarrow 1$) | \Rightarrow | To check if stack is empty. |

FULL $\leftarrow 0$

\Rightarrow Mark the stack is not full.

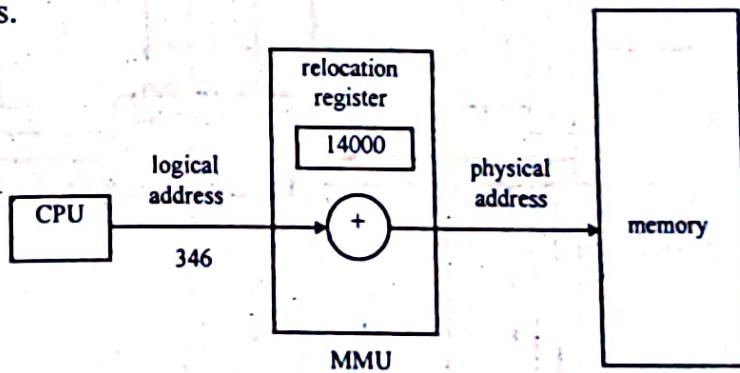


7. What do you mean by Logical address space and Physical address space?

[WBUT 2014]

Answer:

- An 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 memory-address 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. 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. [WBUT 2015, 2016]

Answer:

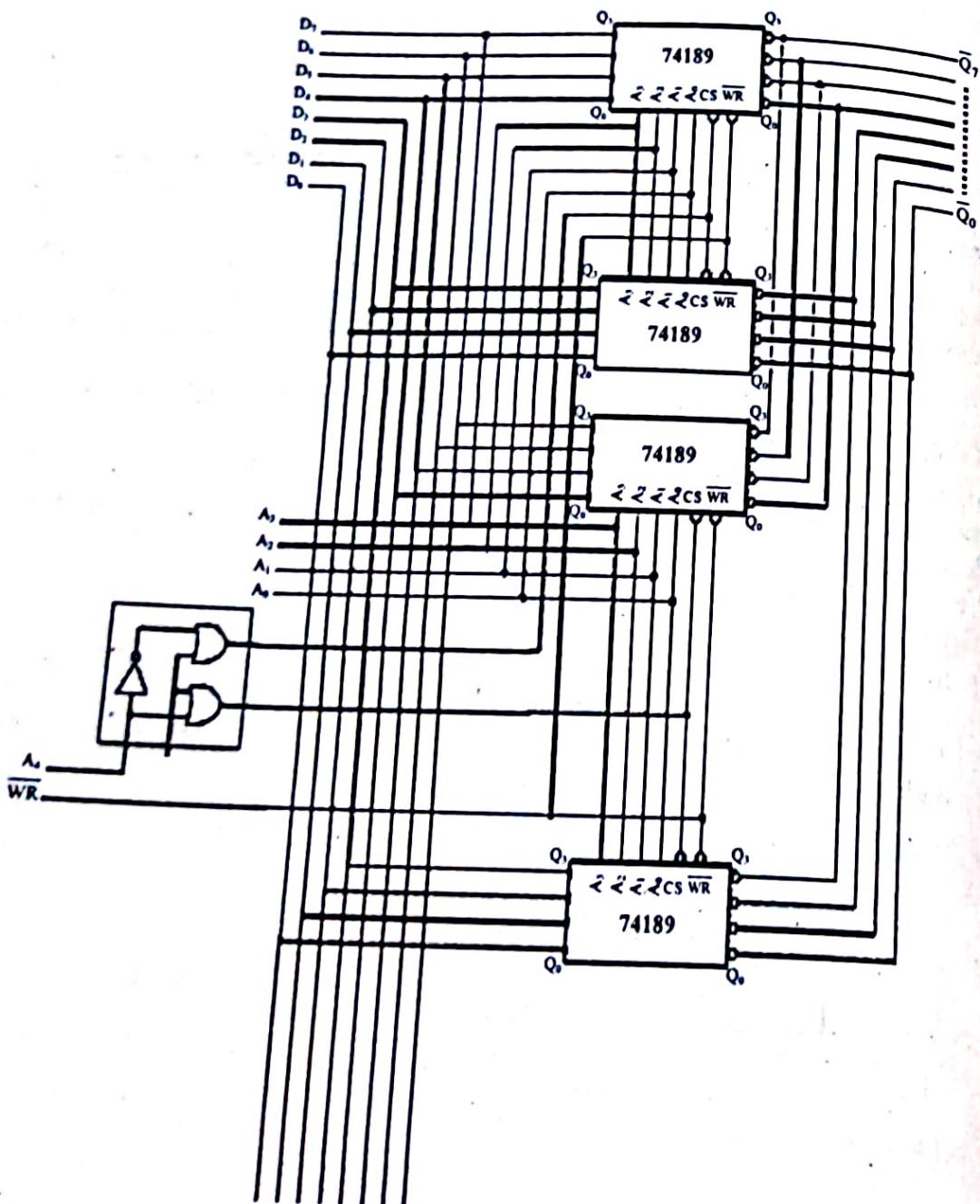
$$2KB = 2^{11} \cdot 8 \text{ bytes}$$

We have RAM chips of 256×4 which means there are 256 rows in the RAM with 4 bits in each row that is one nibble of data.

$$(2^{11} \cdot 8) / (2^8 \cdot 4) = (2^{11-8}) \cdot 2 = 2^3 \cdot 2 = 8 \cdot 2.$$

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-74

9. 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>

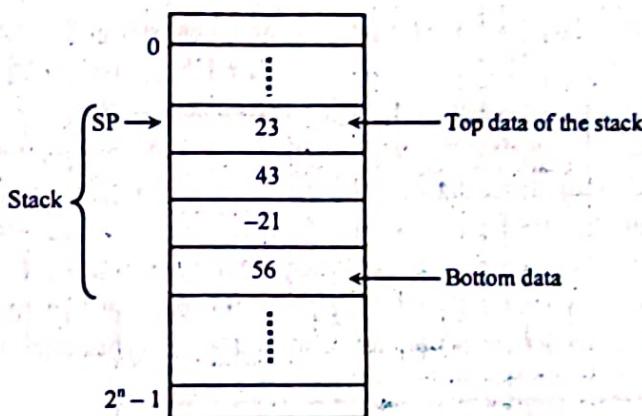


Fig: A stack of words in memory

The PUSH instruction inserts the data word at specified address to the top of the stack.

The POP instruction has the format:

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

$SP \leftarrow SP - 1$: decrement the SP by 1

$SP \leftarrow <\text{memory address}>$: store the content of specified memory address into SP,
i.e., at top of stack

The POP operation can be implemented as

$<\text{memory address}> \leftarrow SP$: transfer the content of SP (i.e., top most data) into
specified memory location

$SP \leftarrow SP + 1$: Increment the SP by 1

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. $2^{10} = (\text{Number of sets}) * 2 * 2^5$

$$\text{Number of sets} = 2^{10} / (2 * 2^5) = 2^{13}$$

When an address is mapped to a set, the direct mapping scheme is used and then associative mapping is used within a set.

14. Average memory access time depends on which factors?

[WBUT 2018]

Answer:

The average memory access time depends on three parameter. It is a common metric to analyze memory system performance. It depends on, hit time or hit 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 miss penalty is the cost of a cache miss in terms of a time.

15. 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?

[WBUT 2019]

Answer:

a) In a direct mapped cache,

Hit rate = 95%

Hit time = 4 ns

Miss penalty = 100 ns

$$\therefore \text{Miss Rate} = 1 - \text{Hit rate} = 1 - 95\% = 0.05$$

$$\begin{aligned}\therefore \text{Average memory access time} &= \text{Hit time} + (\text{Miss rate} \times \text{Miss penalty}) \\ &= 4 + (0.05 \times 100) = 9 \text{ ns}\end{aligned}$$

b) Average Memory access time

$$= \text{Hit time}_{L1} + \text{Miss rate}_{L1} \times (\text{Hit time}_{L2} + \text{Miss rate}_{L2} \times \text{Miss Penalty}_{L2})$$

$$= 4 + 0.05 \times (20 + 0.5 \times 100)$$

$$= 7.5 \text{ ns}$$

16. 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?

[WBUT 2019]

Answer:

i) Bit length for each field in direct mapping = x (left)

\therefore Block size = 4 words = 4×16 bytes

$\therefore x = 64$ bytes = 2^6 = 6 bit

ii) Block size of 4 words = 16 bytes = 2^4 bytes

Therefore, Number of bits in the Word field = 4

Cache size = 2K-byte = 2^{11} bytes, Number of cache blocks per set = 2

Number of sets = Cache size / (Block size * Number of blocks per set)
 $= 2^{11} / (2^4 * 2) = 2^5$

Therefore, Number of bits in the Set field = 5

Total number of address bits = 32

Therefore, Number of bits in the Tag field = $32 - 4 - 5 = 23$

iii) Block size of 4 words = 16 bytes = 2^4 bytes

Therefore, Number of bits in the Word field = 4

Cache size = 2K-byte = 2^{11} bytes, Number of cache blocks per set = 4

For 4 way set-associative mapping,

Number of sets = (Cache size / Block size * Number of blocks per set)
 $= (2^{11} / 2^4 * 4) = 2^5$

Therefore, Number of bits in the Set field = 5

Total number of address bits = 32

Therefore, Number of bits in the Tag field = $32 - 4 - 5 = 23$

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, 2019]

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.

CO-79

e) **Paging:**

Paging is a concept of transfer of pages between main memory and an auxiliary store like 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 O.Ss come with built-in paging files. Page 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.

CONTROL UNIT

Chapter at a Glance

- **Instruction cycle:** An Instruction Cycle consists of one or multiple machine cycles.
Example: A sequence of operations involved in processing an instruction constitutes an instruction cycle. 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 any required operands, performing the operation specified by the instruction's opcode.
- **Hardwired control unit:** In a hardwired control unit, all control signals are generated by means of hardware using conventional simple 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) An encoder.
Each micro program comprises of a sequence of micro operational steps. Each micro operational steps comprise 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 string of 1's and 0's. Each of the bits in different control words generates different microoperations related to the instruction. Control words are generally stored within control memory.

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.

Microinstructions: Microinstruction is an instruction whose bits carry out a set of microoperations at the same time.

Microprogram: A meaningful sequence of microinstructions constitutes a **microprogram**.

Multiple Choice Type Questions

1. In a Micro-processor, the address of the next instruction to be executed, is stored in
[WBUT 2003, 2011]
a) Stack pointer
b) Address hatch
c) Program counter
d) General purpose register

Answer: (c)

POPULAR PUBLICATIONS

2. A UART is an example of

- a) serial asynchronous data transmission ship
- b) PIO
- c) DMA controller
- d) none of these

[WBUT 2010]

Answer: (a)

3. Control program memory can be reduced by

- a) Horizontal format
- b) Vertical format micro-program
- c) Hardwired control unit

[WBUT 2010]
d) None of these

Answer: (b)

4. The cylinder in a disk pack is

[WBUT 2018]

- 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

Answer: (c)

Short Answer Type Questions

1. What are the advantages of microprogramming control over hardwired control?

[WBUT 2008, 2011, 2014]

Answer:

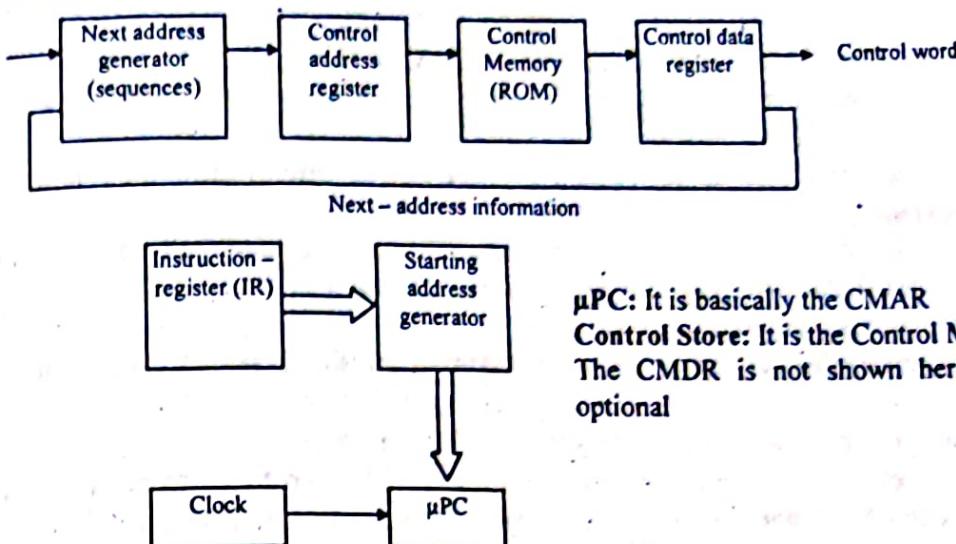
It should be mentioned that most computers today are micro-programmed. The reason is basically one of flexibility. Once the control unit of a hard-wired computer is designed and built, it is virtually impossible to alter its architecture and instruction set. In the case of a micro-programmed computer, 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, we notice that its four-bit op-code permits up to 16 instructions. Therefore, we 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 still used. If speed is a consideration, hard-wiring may be required since it is faster to have the hardware issue the required control signals than to have a "program" do it.

2. Draw the block diagram and explain the functionality of micro-programmed control unit?

[WBUT 2010]

Answer:



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 microinstructions 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 1st microinstruction from the starting address location in the control memory is sent to CMDR.

The CMDR now holds the 1st microinstruction. Simultaneously, the CMDR asks for the next address information to the next address generator.

While asking for the next address information, the CMDR executes the present control 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 control memory & sent it to the CMAR.

This cycle 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.

Answer:

Each microinstruction defines a set of datapath control signals that must be asserted in a given 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 "microprogramming".

We 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 mnemonics 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 Reg.	PCWrite Control	Sequencing
-------------	------	------	-----------	--------	-------------	-----------------	------------

Mapping from instruction code to microinstruction format:

The format for an instruction is shown below:

I	Opcode	Address
---	--------	---------

Now the format for a microinstruction is also the same as the above instruction format:

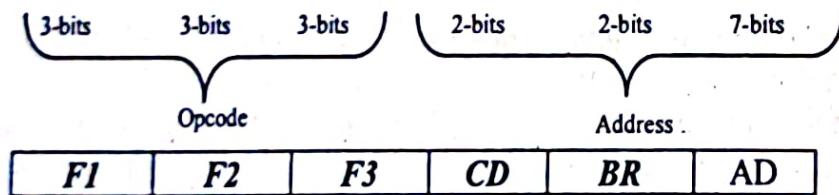


Fig: Microinstruction code format (20 bits)

4. What do you mean by instruction cycle.

[WBUT 2012]

Answer:

Instruction Cycle:

An Instruction Cycle consists of one or multiple machine cycles.

Example: A sequence of operations involved in processing an instruction constitutes an instruction cycle. 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 any required 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 $R_1 \leftarrow R_2$ else $R_1 \leftarrow R_3$, where a and b are control variables.

[WBUT 2008]

Answer:

The question (particularly the meaning of ' $ab' = 1$ ') is not very clear. It looks like the circuit is a MUX which has 2 inputs, r_2 and r_3 , and one output r_1 . If the MUX control input ab' is a 1, then content of r_2 is transferred to r_1 and, otherwise, content of r_3 is transferred to r_1 .

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 Cycle:

One or multiple clock cycles make a machine cycle.

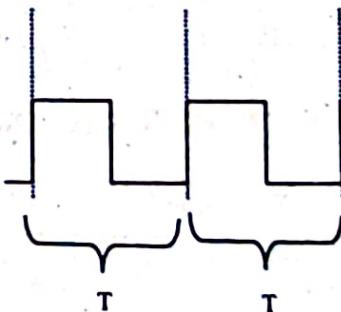
Example: Generally one memory read (i.e. CPU placing address on address bus and then memory sending 'data' back to the CPU via the data bus) or one memory write (i.e. CPU placing 'data' to be written on the data bus, the address of the memory location where data is to be written on the address bus and then sending 'write' pulse on the control bus) cycle 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-107

- **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 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 *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

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

Answer: (c)

[WBUT 2008]

2. A single bus structure is primarily found in

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

Answer: (d)

[WBUT 2008, 2011]

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

- a) k and $n \times 1$
- b) n and 2^k
- c) n and $k \times 1$

[WBUT 2014]

- d) k and 2^n

Answer: (c)

[WBUT 2017]

4. 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

Answer: (c)

[WBUT 2017]

5. The main advantage of multiple bus organisation over single 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

Answer: (a)

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

- a) Δ
- b) $\Delta * n$
- c) $6 * \Delta$

[WBUT 2018]

- d) n

Answer: (b)

Short Answer Type Questions

1. a) 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) What size of multiplexers are needed?

iii) How many multiplexers are there in the bus?

b) Why do most computers have a Common bus system? [WBUT 2010]

OR,

Explain the importance of a common bus system in a computer. [WBUT 2011, 2017]

Answer:

a) i) Number of selection lines = 4 (as $2^4 = 16$, since there are 16 registers).

ii) The size of multiplexers will depend on the number of input lines, Number of data input lines = number of registers. Hence 16 data input lines must be there. So, the size of multiplexers will be 16:1.

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 modules namely CPU, Memory, I/O etc.

Early microcomputer bus systems 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 wire 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. [WBUT 2010]

Answer:

Consider the diagram below. A, B, C & D are the 4 registers connected through the 4-lines bus system. Both the inputs (D-ends of the flip-flops) & outputs (Q-end of the flip-flops) are connected through the bus system. Tristate buffers are there at both the sending & receiving ends of the registers.

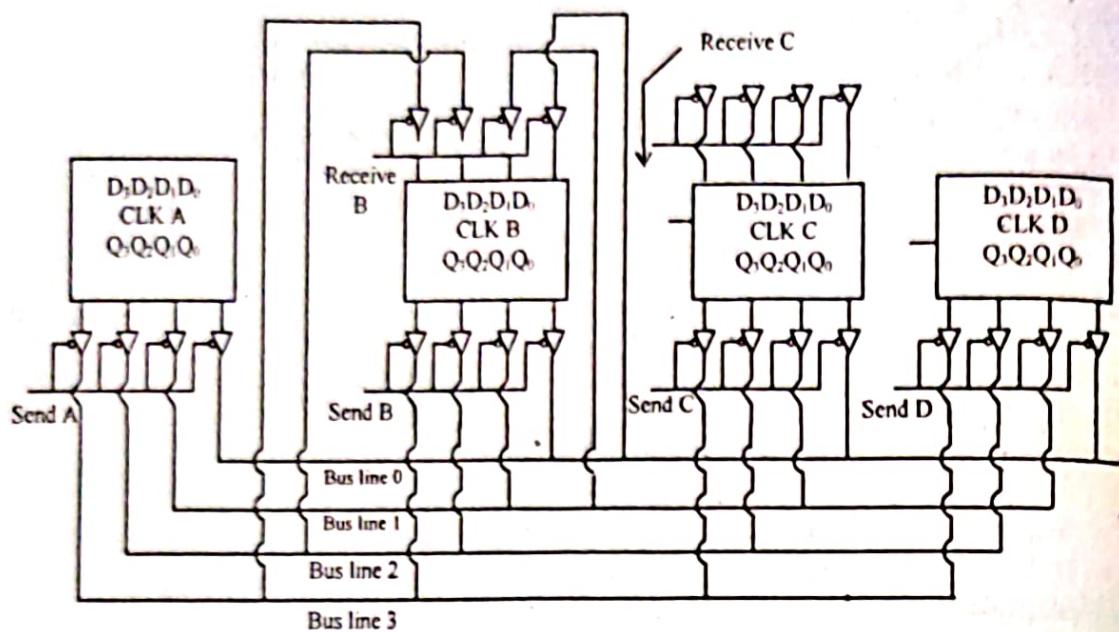


Fig: Bus system constructed with Tristate Buffers

3. What is tri-state buffer? Construct a single line common bus system using tri-state buffer. [WBUT 2015]

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.

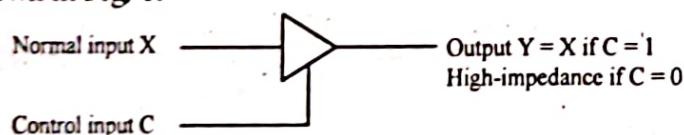


Fig: 1 Graphic symbol for a tri-state buffer gate

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 S_0, S_1 of the decoder 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 S_1S_0 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 A₀, 0th bit of A register.

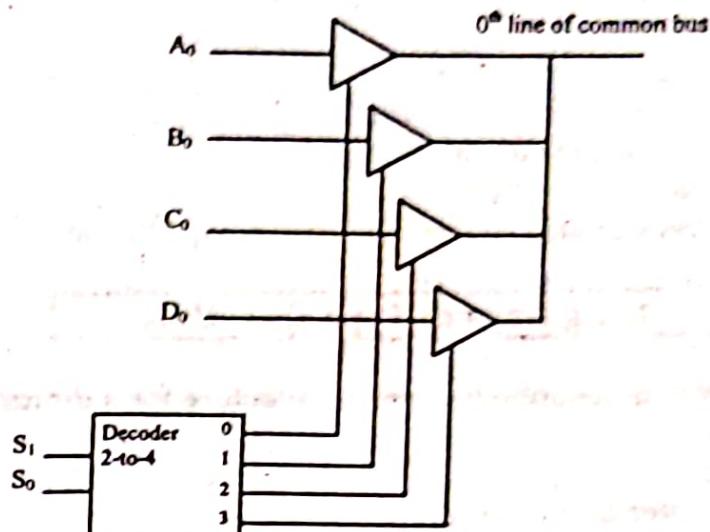


Fig: 2 A single line of a bus system with tri-state buffers

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. 1(a) of Short Answer Type Questions.

5. Why I/O bus is different from a system bus?

[WBUT 2017]

Answer:

Computers have two major types of buses:

1. 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 processors 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 4 blocks sets. The main memory contains 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? [WBUT 2018]

Answer:

The main memory size = 4096 blocks = 2^{12}

Hence, the tag and set field combine will have 12 bits.

Each Block consists of 128 words = 2^7 words.

Hence, the word field length will be 7 bits.

Hence, total size of main memory in word length = $2^{12} \times 2^7 = 2^{19}$

Hence there will be 19 bits in the main memory.

i) There will be 19 bits in a main memory

ii) There are 12 bits in each of the TAG and SET fields and 7 bits in word field length.

Long Answer Type Questions

1. What is bus? Draw and describe the bus architecture for a digital computer.

[WBUT 2012]

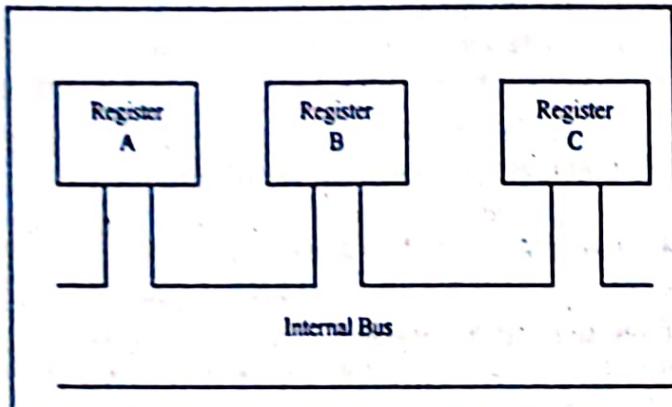
Answer:

1st Part: Refer to Chapter at a Glance.

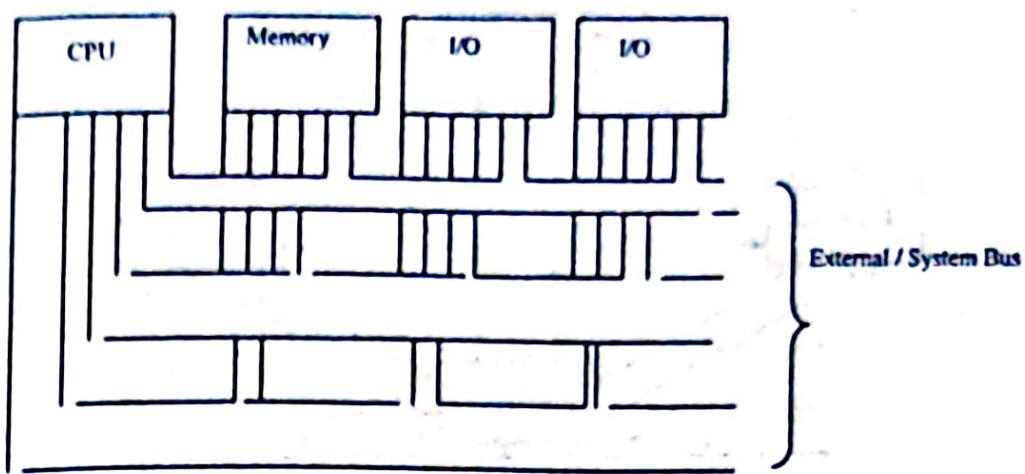
2nd Part:

Bus can be of 4 types,

Internal bus: This runs within the CPU. Internal bus connects all the registers within the 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.



Shared Bus

Here several registers can be connected via a single common bus with the restriction that only one register can send data at any time.

So, here the number of buses and thus the number of cables required are much less. Cost of cables required, 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.

INPUT-OUTPUT ORGANIZATION

Chapter at a 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*.

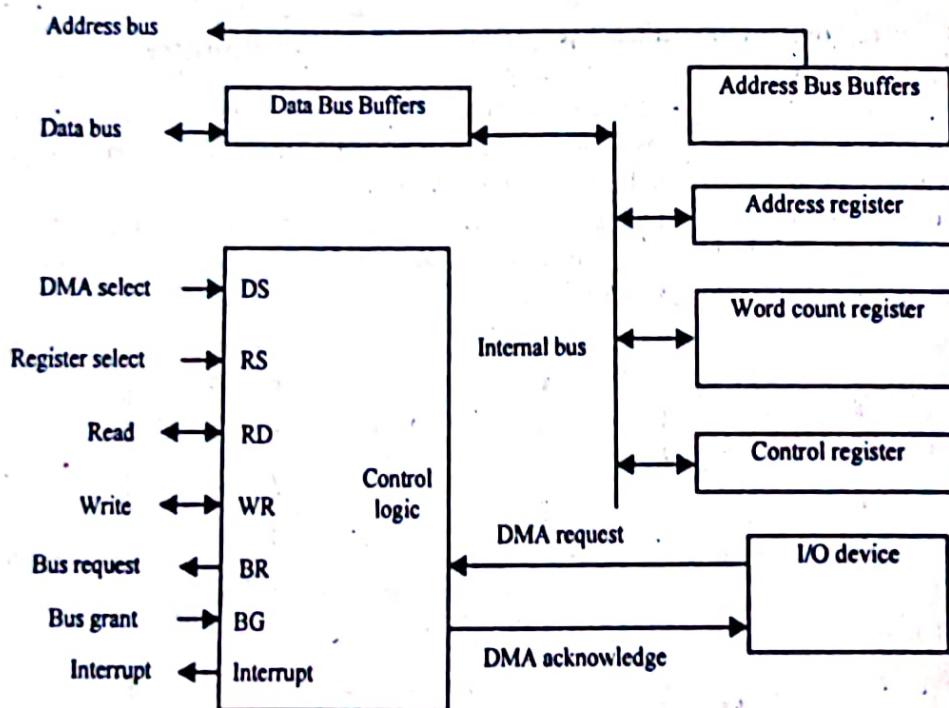
Some of the I.O devices that are commonly used in a computer are:

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

Output Devices: Printer, Monitor etc.

- **Direct memory Access (DMA):** In programmed I/O data transfer 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:



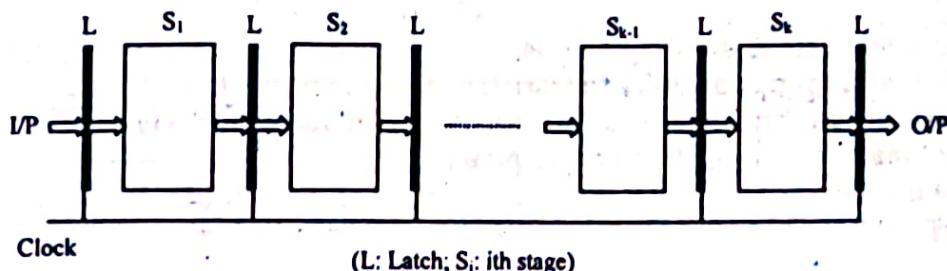
- **Interrupt:** Interrupt is a signal from an I/O device to let the CPU know that it is ready to transfer data and that it is requesting service from the CPU. As soon as CPU receives this signal, it leaves its current unfinished task as it is and branches to the interrupting device's interrupt service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

COMPUTER ORGANISATION

- **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 sources like, from I/O devices when they are ready to transfer data, from a timing device to signify that the time of an event is over, it may occur due to some power failures etc.
 - (b) **Internal Interrupts:** 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 want 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).

- **Non-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.



Multiple Choice Type Questions

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)

2. Memory mapped I/O scheme is used for the allocation of address to memories and I/O devices is used for [WBUT 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? [WBUT 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

[WBUT 2010]

- a) Polling
- b) Daisy chain
- c) Parallel method of priority interrupt
- d) All of these

Answer: (a)

5. BIOS is

[WBUT 2013]

- 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

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 use endlessly
- c) 100 bytes are allowed to be transferred
- d) none of these

Answer: (a)

7. The contention for the usage of a hardware device is called

[WBUT 2019]

- a) structural hazard
- b) stall
- c) deadlock
- d) none of these

Answer: (a)

8. The periods of time when the unit is idle is called as

[WBUT 2019]

- a) Stalls
- b) Bubbles
- c) Hazards
- d) Both Stalls and Bubbles

Answer: (d)

COMPUTER ORGANISATION

9. The DMA transfer is initiated by

[WBUT 2019]

- a) Processor
- b) I/O devices

- c) The process being executed
- d) OS

Answer: (c)

Short Answer Type Questions

1. What are the different hazards in Pipeline?

[WBUT 2007, 2019]

OR,

Explain Pipelining and Hazards

[WBUT 2017]

Answer:

Pipeline hazards are situations that prevent the next instruction in the instruction stream from executing during 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.

2nd Part:

Vectored Interrupt

1. 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.
2. The address of the service routine is hard-wired.

Non-Vectored Interrupt

1. 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.
2. The address of the service routine needs to be supplied externally by the device.

4. a) Where does DMA mode of data transfer 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

COMPUTER ORGANISATION

access to the bus for the duration of the transfer, during which time the program is effectively shut down. Burst mode DMA can transfer data very rapidly indeed.

(iii) Flyby DMA:

Flyby DMA, something that is not supported on many controllers, is a beast of a different color. The DMA controller gains access to the bus and puts the source or destination address out. 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 2012]

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 rate.

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.

Simple to design.

Wastage of CPU cycle degrades the performance of the computer.

Interrupt-initiated I/O

Medium in speed.

Medium cost.

Slightly complicated.

CPU cycle is not wasted.

7. What is meant by "pipeline architecture"? How does it improve the speed of execution of processor? [WBUT 2013]

Answer:

An instruction pipeline is a technique used in the design of computers to increase their instruction throughput (the number of instructions that can be executed in a unit of time). The basic instruction cycle is broken up into a series called a pipeline. Rather than processing each instruction sequentially (one at a time, finishing one instruction before starting the next), each instruction is split up into a sequence of steps so different steps can be executed concurrently (by different circuitry) and in parallel (at the same time). The ideal speedup from a pipeline is equal to the number of stages in the pipeline.

Time per instruction on unpipelined machine

Number of pipe stages

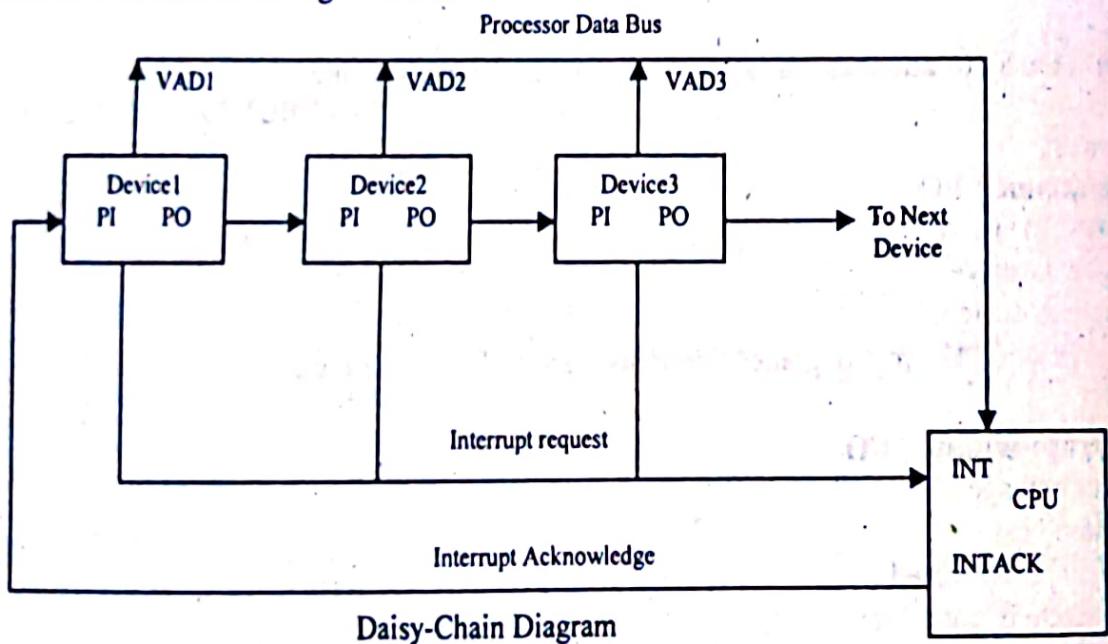
8. Explain with diagram the daisy chaining priority interrupt technique.

[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 figure below.



COMPUTER ORGANISATION

The main advantage of the daisy chain is its simplicity. Another advantage is scalability. The user can add more 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 a 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 does the updating part, 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.

10. What is pipelining? Describe pipeline hazards.

[WBUT 2014]

Answer:

1st Part: Refer to Chapter at a Glance.

2nd Part: Refer to Question No. 1 of Short Answer Type Questions.

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

[WBUT 2019]

Answer:

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 $f0, $f2, $f4  
mul.d $f6, $f8, $f0  
add.d $f0, $f10, $f12
```

The use of register \$f0 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 (WAW) hazards, also known as output dependences
- Write after Read (WAR) hazards, also known as anti-dependences

The naming of these hazards is based on what is supposed to happen. That is, a WAR hazard occurs when an instruction that writes to a register follows soon after an instruction that reads from the same register. If the write precedes the read then the first instruction (the read) is working with the wrong data value.

There are two ways to deal with hazards

- 1) Removal – 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”.

12. Explain Cache Coherence.

[WBUT 2019]

Answer:

In a shared memory multiprocessor system, all processors share a common memory. Also, each processor may have a local memory, part or all of which may be a cache. Now presence of multiple caches in the system means that copies of shared data may reside in several caches and also in the common memory. If any of the processors make any changes in its own copy of the shared variable (i.e. in its own cache), all other cache memories (in other processors) containing a copy of that variable must be informed about that change so that all the copies of that shared variable (in all other caches) get updated at a time. This type of situation in which all cached copies of the shared variable (data) have the same value at all times, is termed as the cache coherence concept.

Probable Solutions to the Cache-Coherence Problem:

The cache coherence concept as discussed is a problem in the multiprocessor system. The probable solutions to this problem may be:

- (a) To disallow local cache memories for each processor and have a shared cache memory associated with main memory. Every data access is made to the shared cache. However this scheme increases the average memory access time.
- (b) To allow only non-shared and read-only data (called as cachable data) to be stored in the local caches. The shared writable data (called as non-cachable data) are to be stored in main memory. So measures must be taken to identify whether a data is cachable or non-cachable.
- (c) Allowing writable data to exist in at least one cache. This method employs a centralized global table that stores the status of memory blocks. Each block is identified as read-only (RO) or read and write (RW). All caches can have blocks identified as RO. Only one cache (which stores writable data) can have a copy of a RW block. So if the data are updated in the cache with the RW block, the other caches are not affected because they do not have a copy of this block.

- (d) Software and hardware means are used to solve the problem also. Methods 'b' and 'c' are software-based schemes. The snoopy-cache protocol uses the hardware-based schemes.

Long Answer Type Questions

1. What is interrupt? What are the differences between vectored and non-vectored interrupt?
[WBUT 2009]

Answer:

Refer to Question No. 3 of Short Answer Type Questions.

2. How does polling work?

[WBUT 2012]

Answer:

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 interrupts. 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. The highest priority source is tested first, and if its interrupt signal is on, control branches to a service routine for this source. Otherwise, the next-lower-priority source is tested, and so on. Thus the initial service routine for all interrupts consists of a program that tests the interrupt sources in sequence and branches to one of many possible service routines. The particular service routine reached belongs to the highest priority device among all devices that interrupted the computer. The disadvantage of the software method is that if there are many interrupts, 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 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.

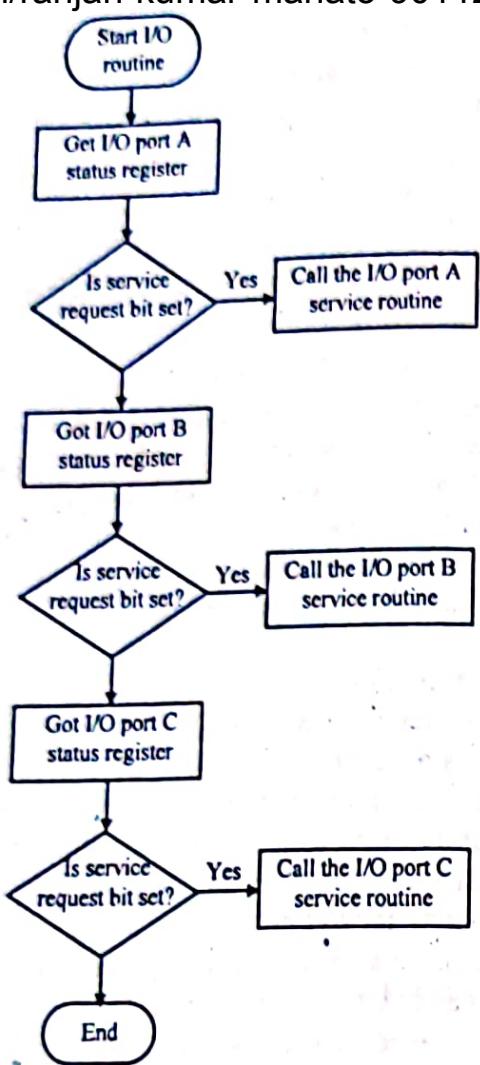


Fig: Flowchart for I/O service routine

f) Direct Memory Access: *Refer to Chapter at a Glance.*

QUESTION 2015

Group - A

(Multiple Choice Type Questions)

1. Choose the correct alternatives for the following:

i) The maximum number of additions and subtractions are required for which of the following multiplier numbers in Booth's algorithm?

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

ii) In the processor, the address of the next instruction to be executed is stored in

- a) stack pointer register b) index register
✓c) base register d) program counter register

iii) By logical left-shifting the content of a register once, its content is

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

iv) 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

- 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

v) The principle of locality justifies the use of

- a) interrupt b) polling c) DMA ✓d) cache memory

vi) Instruction cycle is

- ✓a) fetch-decode-execution b) fetch-execution-decode
c) decode-fetch-execution d) none of these

vii) Subtractor can be implemented using

- a) adder b) complementer
✓c) both (a) and (b) d) none of these

viii) How many RAM chips of size (256 K × 1 bit) are required to build 1M Memory?

- a) 24 b) 10 c) 32 ✓d) 8

ix) Maximum n bit 2's complement number is

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

x) Micro instructions are kept in

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

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.

CO-148

- (ii) Using a stack organized computer with zero-address instructions.
b) What is von Neumann bottleneck? How can this be reduced?
c) A digital computer has a memory unit of $64K \times 16$ 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.

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.(a).

10. a) 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.

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 DMA operations for transfer of data between memory and peripherals.

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:

i) RAM is called DRAM (Dynamic RAM) when

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

ii) Floating point representation is used to store

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

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

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

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?

- a) 1 b) 2 c) 3 ✓d) 4

v) A computer's memory is composed of 8K words of 32 bits each. How many bits are required for memory address if the smallest addressable memory unit is a word?

- ✓a) 13 b) 8 c) 10 d) 6

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
c) Both (a) and (b) ✓b) (00101111101000001100)
d) None of these

x) The addressing mode used in an instruction of the form ADD X 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
c) Auxiliary memory b) Main memory
✓d) Cache memory

xii) A stack-organised computer uses instruction of

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

Group - B

(Short Answer Type Questions)

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

See Topic: INSTRUCTION SET, Short Answer Type Question No. 7.

CO-150

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

See Topic: COMPUTER ARITHMATIC, Long Answer Type Question No. 4(b).

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?

See Topic: BUS STRUCTURE, Short Answer Type Question No. 4.

5. 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 instruction.

See Topic: INSTRUCTION SET, Long Answer Type Question No. 3.

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

See Topic: COMPUTER ARITHMATIC, 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 –

- 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?

b) Use restoring method to divide 10100011 by 1011.

c) Suppose we are given RAM chips each of size 256×4 . Design a $2K \times 8$ RAM system using this chip as he 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 ARITHMATIC, 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.

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. 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?

<https://www.linkedin.com/in/ranjan-kumar-mahato-90112424b>

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 size is 128Kx32.

- i) How many bits are there in the tag, index block and word fields of the address format?
- 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.

10. a) Differentiate between hardwared control and micro-programmed control. Draw the block diagram of a basic hardwared control organization with two decoders, a sequence counter 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 DMA 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. 3.b).

11. 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.

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

QUESTION 2017

Group - A

(Multiple Choice Type Questions)

1. 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) Heap d) Stack

CO-152

COMPUTER ORGANISATION

- iii) is generally used to increase the apparent size of physical memory.
a) Secondary memory ✓b) Virtual memory
c) Hard disk d) Disks
- iv) The time delay between two successive initiations of memory operation is
a) Memory access time b) Memory search time
✓c) Memory cycle time d) Instruction delay
- v) The main advantage of multiple bus organisation over single 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 stack
- 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) NAND gate c) NOR gate ✓d) XOR gate
- x) A 24 bit address generates an address space of locations
a) 1024 b) 4096 c) 2^{48} ✓d) 16,777,216
- xi) To get the physical address from the logical address generated by CPU we use
a) MAR ✓b) MMU c) Overlays d) TLB
- xii) During transfer of data between the processor 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.

See Topic: MEMORY ORGANIZATION, Short Answer Type Question No. 10.

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

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

2nd part: See Topic: INTRODUCTION, Short Answer Type Question No. 3.

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

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

2nd part: See Topic: MEMORY ORGANIZATION, Short Answer Type Question No. 11.

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.

See 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) See 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. 3.

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

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

10. 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.

i) What is the storage capacity of the pack?

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

d) What is the necessity of guard bits?

a) See Topic: MEMORY ORGANIZATION, Long Answer Type Question No. 8(a).

b) See Topic: INPUT-OUTPUT ORGANIZATION, Long Answer Type Question No. 8.

COMPUTER ORGANISATION

- c) See Topic: **MEMORY ORGANIZATION**, Long Answer Type Question No. 14(b).
d) See Topic: **COMPUTER ARITHMETIC**, 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?
a) See Topic: **INSTRUCTION SET**, Short Answer Type Question No. 4.
b) See Topic: **INPUT-OUTPUT ORGANIZATION**, Long Answer Type Question No. 4.
c) 1st part: See Topic: **BUS STRUCTURE**, Short Answer Type Question No. 1(b).
2nd part: See Topic: **BUS 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
a) See Topic: **INSTRUCTION SET**, Long Answer Type Question No. 5(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).
e) 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) DMA ✓d) Cache memory
- ii) Instruction cycle is
✓a) fetch-decode-execution b) fetch-execution-decode
c) decode-fetch-execution d) decode-execution-fetch
- iii) How many RAM chips of size (256K × 1 bit) are required to build 1M Memory?
a) 24 b) 10 ✓c) 32 d) 8
- iv) Maximum value on n bit 2's complement number is
a) 2^n b) $2^n - 1$ ✓c) $2^{n-1} - 1$ d) Cannot be said
- v) Micro instructions are kept in
a) Main memory ✓b) Control memory c) Cache memory d) Secondary storage

- vi) The maximum propagation delay for n -bit CLA is
a) Δ ✓b) $\Delta \cdot n$ c) $6 \cdot \Delta$ d) n
- vii) The cylinder in a disk pack is
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
- 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) 00001111 d) 01010101
- ix) 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)

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

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

3. Explain stack based CPU.

See 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?

See Topic: MEMORY ORGANIZATION, Short Answer Type Question No. 13.

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

1st Part: See Topic: INPUT-OUTPUT ORGANIZATION, Long Answer Type Question No. 6(c).

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

6. Explain with example: Register Direct, Register Indirect and Base register addressing mode.

See Topic: INSTRUCTION SET, Short Answer Type Question No. 8.

Group - C

(Long Answer Type Questions)

7. a) Explain the basic block diagram of Computer System. Why do peripherals need interface circuits with them?

1st Part: See Topic: INTRODUCTION, Long Answer Type Question No. 2.

2nd Part: See Topic: INPUT-OUTPUT ORGANIZATION, Long Answer Type Question No. 5.

COMPUTER ORGANISATION

b) A block **set-associative cache** consists of a total of 64 blocks divided into 4 blocks sets. The main memory contains 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?

See Topic: **BUS STRUCTURE**, Short Answer Type Question No. 5.

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

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

8. 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.

9. a) If a CPU has 16 bit address bus and 8 bit data bus draw the connection Diagram for this CPU with four 256×8 RAM and one 512×8 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(2nd 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

- 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 Questions)

1. Choose the correct alternatives for any ten of the following:
- i) A source program is usually in
a) Assembly language
 c) High-level language
b) Machine 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) 2^n 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) How 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) Finds the memory location 45 and adds that content to that of R1
d) None of these
- vii) A 24 bit address generates an address space of locations.
a) 1024 b) 4096 c) 2^{48} d) 16,777,216
- viii) What could be the maximum size of on chip cache memory for an n-address bit processor?
a) 0 b) 2^n c) infinite d) decided by manufacturer
- ix) To get the physical address from the logical address generated by CPU we use
a) MAR b) MMU c) Overlays d) TLB

COMPUTER ORGANISATION

- x) The contention for the usage of a hardware device is called
✓ a) structural hazard b) stall 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 DMA transfer is initiated by
a) Processor b) The process being executed
✓ c) I/O devices d) OS

Group - B

(Short Answer Type Questions)

2. 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) Rounding 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.

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

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

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

POPULAR PUBLICATIONS

- a) See Topic: MEMORY ORGANIZATION, Long Answer Type Question No. 1(b) (1st Part).
- b) See Topic: MEMORY ORGANIZATION, Long Answer Type Question No. 1(a).
- c) See Topic MEMORY ORGANIZATION, Short Answer Type Question No. 16.

8. a) What is the difference between isolated I/O and memory mapped I/O?

b) Explain Cache Coherence.

c) Explain the different hazards in pipelining.

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

b) See Topic: INPUT-OUTPUT ORGANIZATION, Short Answer Type Question No. 12.

c) See Topic: INPUT-OUTPUT ORGANIZATION, Short Answer Type Question No. 1.

9. a) What are the different addressing modes?

b) Explain with suitable examples of each of the modes.

c) Evaluate the following arithmetic expression into (i) three address, (ii) two addresses, (iii) one address and (iv) zero address instruction format $X = (A + B)^*(C + D)$.

a) & b) See Topic: INSTRUCTION SET, Long Answer Type Question No. 1.

c) See Topic: INSTRUCTION SET, Short Answer Type Question No. 9.

10. a) What are the differences between RISC and CISC?

b) Divided 43 by 11 using Non-restoring algorithm (The tracing table must be shown clearly)

c) What is meant by overflow and underflow in signed magnitude representation of numbers?

a) See Topic: INSTRUCTION SET, Short Answer Type Question No. 3.

b) & c) See Topic: COMPUTER ARITHMETIC, Long Answer Type Question No. 11(a) & (b).

MATHEMATICS - III

Sequence and Series	2
Multivariable Calculus (Differentiation)	30
Multivariable Calculus (Integration)	70
Ordinary Differential Equation	92
Graph Theory	133

NOTE:

MAKAUT course structure and syllabus of 3rd Sem has been changed from 2019. Present syllabus of MATHEMATICS III [for CS & IT] has been designed and structured with selected topics from other MATHEMATICS of different years and branches. Taking special care of this matter we are providing the relevant MAKAUT university solutions as model questions & answers alongwith the complete solutions of new university papers, so that students can get an idea about university questions patterns.