SGBB GOVT POLYTECHNIC COLLEGE, SIROHI

MAX TIME- 1 HR CLASS TEST-II (SESSION 2017-18) MAX MARKS-15

SUB- Microprocessor and Interfacing (EL207/CS208) DATE-30.1.2018

Name of Faculty-RAHUL SINGH RAJPUROHIT (LECTURER-EL)

Q1. Explain the addressing mode of 8085 microprocessor. **(5)**

Q2. Explain the demultiplexing of buses of 8085 with suitable diagram. **(5)**

Q3. Explain the following instruction with example (**Any two**) **{2.5+2.5=5}**

(a)LDA 16 bit address (c) XRI , 8bit data

(b)RAR (d) SUB R

Model Answer

Q 1) Explain the addressing mode of 8085 microprocessor. **(5)**

Ans- There are five addressing modes in 8085.

1. Immediate Addressing Mode: - An immediate is transferred directly to the register.

Eg: - MVI A, 30H (30H is copied into the register A)
MVI B,40H(40H is copied into the register B).

2. Register Addressing Mode: - Data is copied from one register to another register.

Eg: - MOV B, A (the content of A is copied into the register B)

MOV A, C (the content of C is copied into the register A).

3. Direct Addressing Mode: - Data is directly copied from the given address to the register.

Eg: - LDA 3000H (The content at the location 3000H is copied to the register A).

4. Indirect Addressing Mode: - The data is transferred from the address pointed by the data in a register to other register.

Eg: - MOV A, M (data is transferred from the memory location pointed by the register to the accumulator).

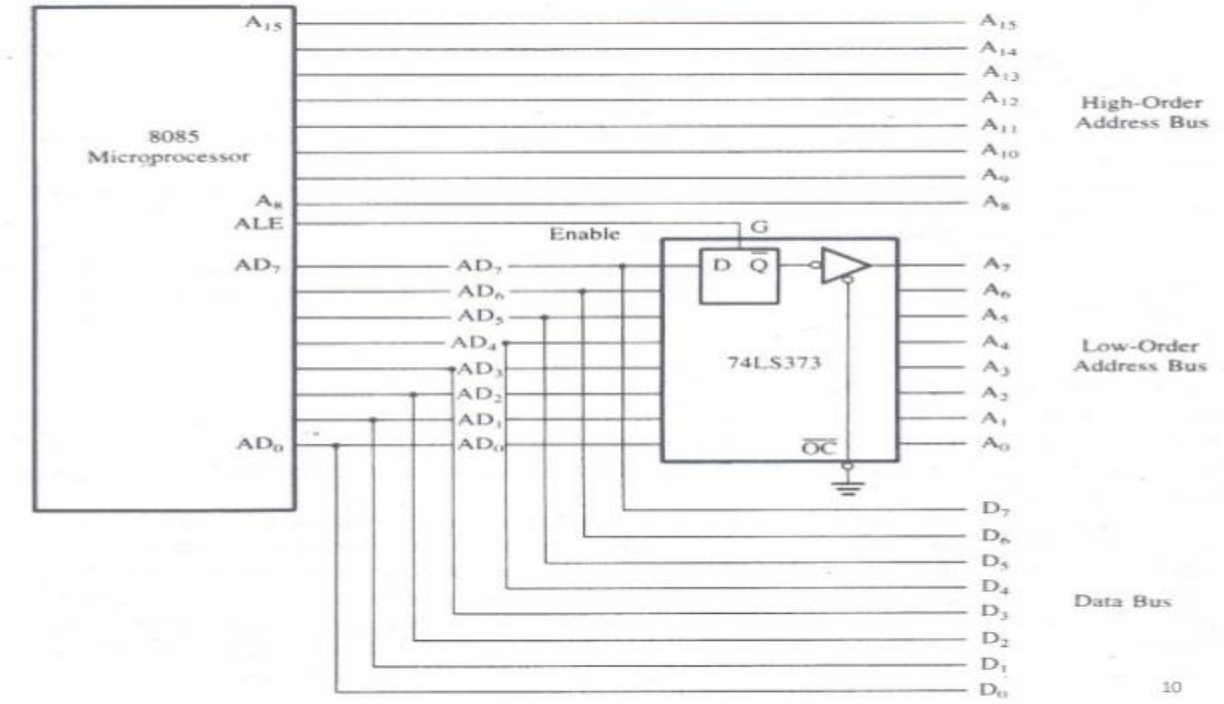
5. Implied Addressing Mode: - This mode doesn't require any operand. The data is specified by opcode itself.

Eg: - RAL

CMP

2) Explain the demultiplexing of buses of 8085 with suitable diagram. **(5)**

Ans- The AD_0-AD_7 pins can used as both address bus and data bus(only one at a time) depending upon the state of ALE. The 8085 microprocessor is used IC 74LS373 to latch the address of 8085. Basically latch is consists of 8 flip flops. Generally we used D-flip flops (Delay).The clock of these flip flops are connected together and available as a output pin called enable. The address will appear on AD0 AD7 lines. The ALE will go high and forcing Enable = 1. This will make latch enable and ready to work. Before address disappears ALE = 0. This will make latch disable. AD0 - AD7 will now be used as data bus. Hence, AD0 - AD7 (low order) address bus of the 8085 microprocessor is multiplexed (time-shared) with the data bus. The buses need to be demultiplexed. The data bus and the low order address bus on the 8085 microprocessor are multiplexed with each other. This allows 8 pins to be used where 16 would normally be required. The hardware interface is required to demultiplex the bus by latching the low order address in the first T cycle, on the falling edge of ALE. The AD0-AD7 lines in an 8085 are multiplexed to reduce the pin count of the IC.



3) Explain the following instruction with example (**Any two**) {**2.5+2.5=5**}

(a)LDA 16 bit address (c) XRI ,8bit data

(b)RAR (d) SUB R

(a)LDA 16 bit address -Load accumulator. (this instruction copies the data from a given 16 bit address to the accumulator)

Eg: - LDA 3000H (content of memory location 3000h is coppied in accumulator)

(b)RAR -

Each binary bit of the accumulator is rotated right by one position through the Carry flag. Bit D0 is placed in the Carry flag, and the Carry flag is placed in the most significant position D7. CY is modified according to bit D0.

(c) XRI ,8bit data

The content of accumulator are exclusive OR with the immediate data.

Eg: - XRI 30H

(d) SUB R

Subtract the content of a register or a memory location form the content of accumulator and the result is stored in the accumulator.

Eg: - SUB B (it subtracts the content of B register from the content of the accumulator.

SGBB GOVT POLYTECHNIC COLLEGE, SIROHI

MAX TIME- 1 HR **CLASS TEST-II (SESSION 2017-18)** MAX MARKS-15

SUB- **Digital Electronics EL/CS 205**

NOTE- ATTEMPT ALL THREE QUESTIONS. (Each Question Carry Equal Marks)

Q.1. Simplfy using K-Mape and design using gate :-

$F(A,B,C,D) = \sum m(4,5,7,8,10,11,13,14) + d(0,1,2)$

Q.2.Simplify using k mape $F(A,B,C) = \sum m(0,1,2,3,6,7)$

Q.3. What do you understand by SOP and POS form? Explain the conversion method of SOP and POS.

Answers:-

Q.1. Simplfy using K-Mape and design using gate :-

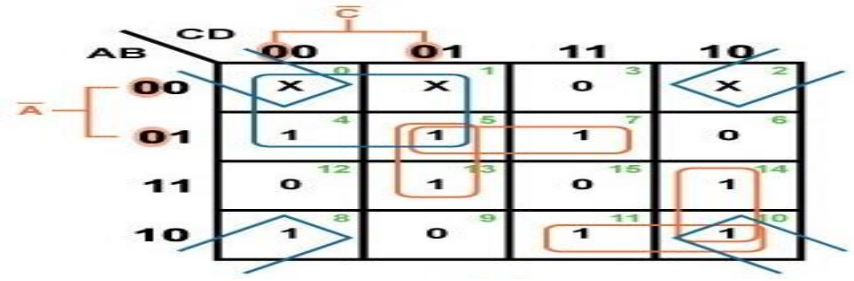
$F(A,B,C,D) = \sum m(4,5,7,8,10,11,13,14) + d(0,1,2)$

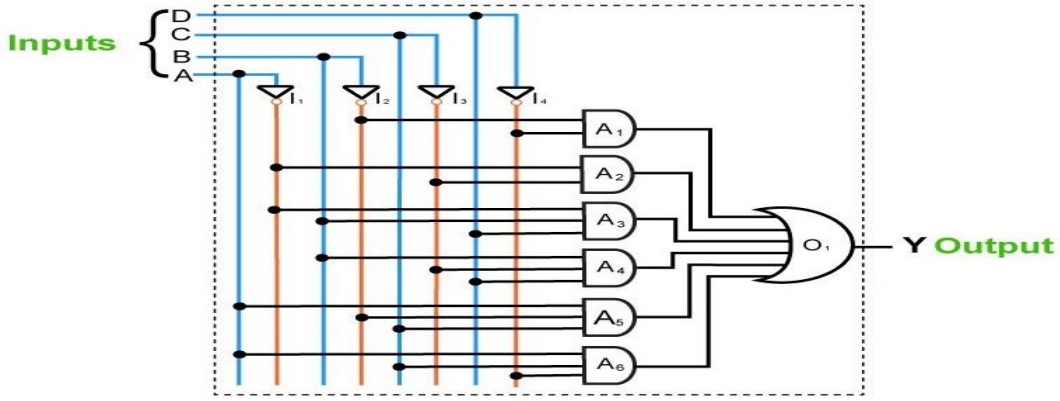
Ans:-

Truth Table:-

Decimal Number	A	B	C	D	Out Put
0	0	0	0	0	x
1	0	0	0	1	x
2	0	0	1	0	x
3	0	0	1	1	0
4	0	1	0	0	1
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	0
10	1	0	1	0	1
11	1	0	1	1	1
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	0

k-map





Circuit Design F (A,B,C,D)= A'C'+B'D'+AB'C+ACD'+A'BD+BC'D

Q.2.Simplify using k mape F(A,B,C) = ∑ m(0,1,2,3,6,7)

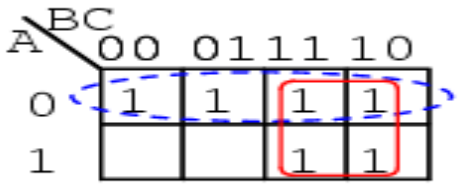
Ans:-

Truth Table:-

Decimal Number	A	B	C	Out Put
0	**0**	**0**	**0**	1
1	**0**	**0**	**1**	1
2	**0**	**1**	**0**	1
3	**0**	**1**	**1**	1
4	**1**	**0**	**0**	0
5	**1**	**0**	**1**	0
6	**1**	**1**	**0**	1
7	**1**	**1**	**1**	1

K-Mape

$$Out= \overline{A}\,\overline{B}\,\overline{C}+\overline{A}\,\overline{B}\,C+\overline{A}\,B\,C+\overline{A}\,B\,\overline{C}+A\,B\,C+A\,B\,\overline{C}$$



$$Out= \overline{A} + B$$

Q.3. What do you understand by SOP and POS form? Explain the conversion method of SOP and POS.

Ans:-

A Boolean function is an algebraic form of Boolean expression.There are two types of canonical forms: 1.Sum-of-min terms or Canonical SOP 2. Product-of- max terms or Canonical POS

1. **Sum of Product (SOP) Form**

The sum-of-products (SOP) form is a method (or form) of simplifying the Boolean expressions of logic gates. In this SOP form of Boolean function representation, the variables are operated by

AND (product) to form a product term and all these product terms are ORed (summed or added) together to get the final function.

A sum-of-products form can be formed by adding (or summing) two or more product terms using a Boolean addition operation. Here the product terms are defined by using the AND operation and the sum term is defined by using OR operation.

The sum-of-products form is also called as Disjunctive Normal Form as the product terms are ORed together and Disjunction operation is logical OR. Sum-of-products form is also called as Standard SOP.

SOP form can be obtained by

- Writing an AND term for each input combination, which produces HIGH output.
- Writing the input variables if the value is 1, and write the complement of the variable if its value is 0.
- OR the AND terms to obtain the output function.

2. Product of Sums (POS) Form

The product of sums form is a method (or form) of simplifying the Boolean expressions of logic gates. In this POS form, all the variables are ORed, i.e. written as sums to form sum terms. All these sum terms are ANDed (multiplied) together to get the product-of-sum form. This form is exactly opposite to the SOP form. So this can also be said as "Dual of SOP form".

Here the sum terms are defined by using the OR operation and the product term is defined by using AND operation. When two or more sum terms are multiplied by a Boolean OR operation, the resultant output expression will be in the form of product-of-sums form or POS form.

The product-of-sums form is also called as Conjunctive Normal Form as the sum terms are ANDed together and Conjunction operation is logical AND. Product-of-sums form is also called as Standard POS.

Decimal Number	**A**	**B**	**C**	**SOP**	**POS**
0	0	0	0	A'B'C'	ABC
1	0	0	1	A'B'C	ABC'
2	0	1	0	A'BC'	AB'C
3	0	1	1	A'BC	AB'C'
4	1	0	0	AB'C'	A'BC
5	1	0	1	AB'C	A'B'C'
6	1	1	0	ABC'	A'B'C
7	1	1	1	ABC	A'B'C'

S.G.B.B. GOVERNMERNT POLYTECHNIC COLLEGE SIROHI

SECOND CLASS TEST

Subject: C Language: CS201 dated 27.03.2018 / EL210 dated 26.03.2018

MM 15 Time 1Hour

Q1. Write difference between Array and Structure.

Q2. Write the advantages of using Pointer.

Q3. Write Short Note on:

& operator and

* operator

Ans 1.

Arrays	Structures
1. An array is a collection of related data elements of the same type.	1. Structure can have elements of different types
2. An array is a derived data type	2. A structure is a programmer-defined data type
3. Any array behaves like a built-in data types. All we have to do is to declare an array variable and use it.	3. But in the case of structure, first, we have to design and declare a data structure before the variable of that type are declared and used.
4. Array allocates static memory and uses index/subscript for accessing elements of the array.	4. Structures allocate dynamic memory and uses (.) operator for accessing the member of a structure.
5. An array is a pointer to the first element of it	5. Structure is not a pointer
6. Element access takes relatively less time.	6. Property access takes relatively large time.

Ans 2:

Major advantages of pointers are:

(i) It allows management of structures which are allocated memory dynamically.
(ii) It allows passing of arrays and strings to functions more efficiently.
(iii) It makes possible to pass address of structure instead of entire structure to the functions.
(iv) It makes possible to return more than one value from the function.

Ans 3: Address of (&) Operator in C

This operator returns address of any variable.

Consider the given example

```c
#include <stdio.h>

int main()
{
	int x=10;

	printf("Value of x: %d\n",x);
	printf("Address of x: %X\n",&x);

	return 0;
}
```

Output

```
Value of x: 10
Address of x: F0C10E3C
```

Unary plus (+) Operator

This operator does not make any effect on the operand value, it just returns operands value.

Consider the given example:

```c
#include <stdio.h>

int main()
{
	int x= +4;
	printf("x= %d\n",x);

	return 0;
}
```

Output

```
x= 4
```

Here, we assigned +4 to the variable `x` and the result is 4.

2) Unary minus (-) Operator

This operator makes the value negative. It makes positive value to negative and negative value to positive.

Consider the given example:

```c
#include <stdio.h>

int main()
{
	int x=10;
	int y=-20;

	printf("value of -x: %d\n",-x);
	printf("value of -y: %d\n",-y);

	return 0;
}
```

Output

```
value of -x: -10
value of -y: 20
```

Here, we assigned 10 to variable `x` and -20 to variable `y`, when we print the value of both variables using **Unary minus operator**, the result is **-x= 10 and -y= 20**.

SGBB GOVT POLYTECHNIC COLLEGE, SIROHI

CLASS TEST-II (SESSION 2017-18)

MAX TIME- 1 HR MAX MARKS-15

SUB- ELECTRONICS DEVICE AND CIRCUIT (EL204/CS204)

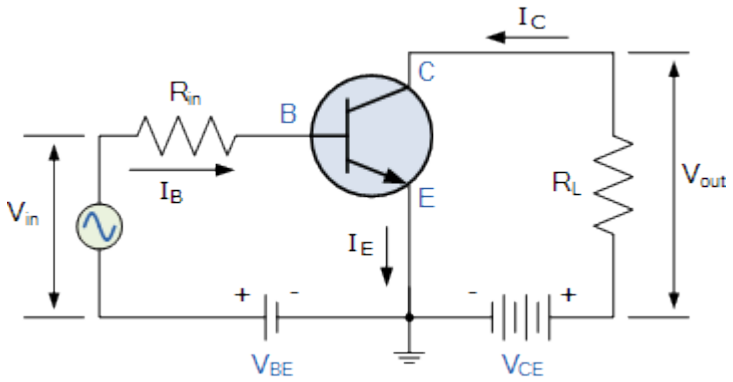
Q-1 Draw and explain circuit diagram of a BJT in common emitter (CE) configuration with output characteristics diagram.

Ans:

In the Common Emitter or grounded emitter configuration, the input signal is applied between the base and the emitter, while the output is taken from between the collector and the emitter as shown. This type of configuration is the most commonly used circuit for transistor based amplifiers and which represents the "normal" method of bipolar transistor connection.

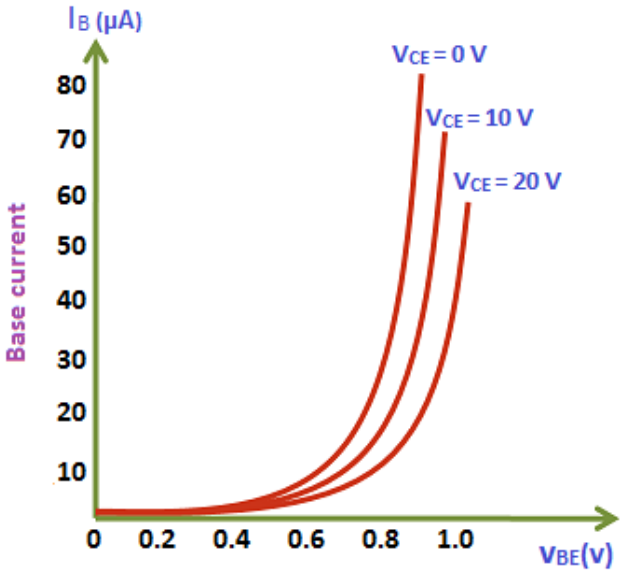
The common emitter amplifier configuration produces the highest current and power gain of all the three bipolar transistor configurations. This is mainly because the input impedance is LOW as it is connected to a forward biased PN-junction, while the output impedance is HIGH as it is taken from a reverse biased PN-junction.

The Common Emitter Amplifier Circuit



In this type of configuration, the current flowing out of the transistor must be equal to the currents flowing into the transistor as the emitter current is given as Ie = Ic + Ib.

As the load resistance (R_L) is connected in series with the collector, the current gain of the common emitter transistor configuration is quite large as it is the ratio of Ic/Ib. A transistors current gain is given the Greek symbol of Beta, (β).



I/P characteristics CE configuration

As the emitter current for a common emitter configuration is defined as Ie = Ic + Ib, the ratio of Ic/Ie is called Alpha, given the Greek symbol of α. Note: that the value of Alpha will always be less than unity.

Since the electrical relationship between these three currents, Ib, Ic and Ieis determined by the physical construction of the transistor itself, any small change in the base current (Ib), will result in a much larger change in the collector current (Ic).

Then, small changes in current flowing in the base will thus control the current in the emitter-collector circuit. Typically, Beta has a value between 20 and 200 for most general purpose transistors. So if a transistor has a Beta value of say 100, then one electron will flow from the base terminal for every 100 electrons flowing between the emitter-collector terminal.

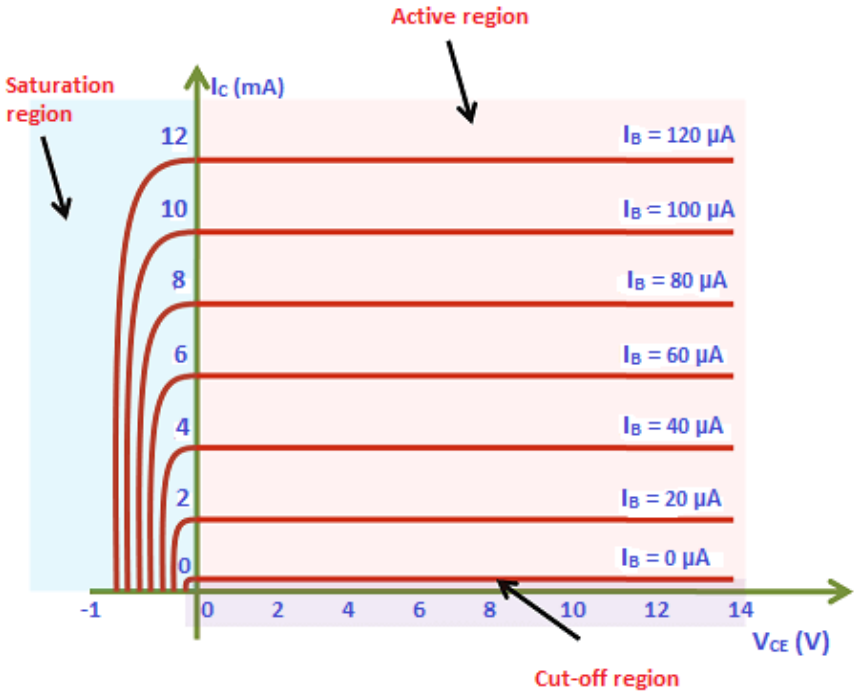
By combining the expressions for both Alpha, α and Beta, β the mathematical relationship between these parameters and therefore the current gain of the transistor can be given as:

$$\text{Alpha},(\alpha) = \frac{I_C}{I_E} \quad \text{and} \quad \text{Beta},(\beta) = \frac{I_C}{I_B}$$

$$\therefore I_C = \alpha . I_E = \beta . I_B$$

$$\text{as:} \quad \alpha = \frac{\beta}{\beta + 1} \qquad \beta = \frac{\alpha}{1 - \alpha}$$

$$I_E = I_C + I_B$$



O/P Characteristics CE Configuration

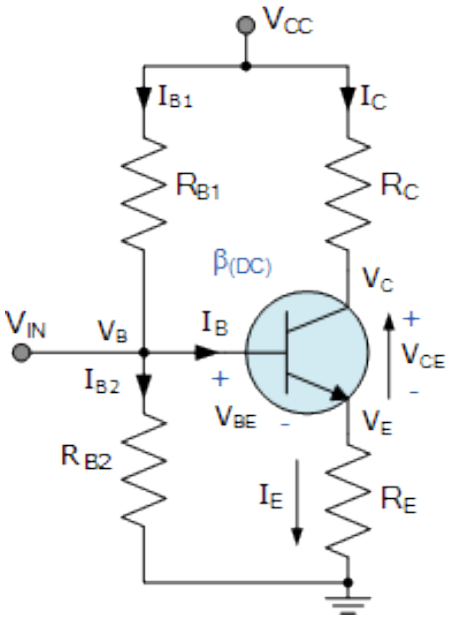
Where: "Ic" is the current flowing into the collector terminal, "Ib" is the current flowing into the base terminal and "Ie" is the current flowing out of the emitter terminal.

Then to summarise a little. This type of bipolar transistor configuration has a greater input impedance, current and power gain than that of the common base configuration but its voltage gain is much lower. The common emitter configuration is an inverting amplifier circuit. This means that the resulting output signal is 180^o "out-of-phase" with the input voltage signal.

Q-2 Explain voltage divider biasing.

Ans: **Voltage Divider Transistor Biasing**

The common emitter transistor is biased using a voltage divider network to increase stability. The name of this biasing configuration comes from the fact that the two resistorsR_{B1} and R_{B2} form a voltage or potential divider network across the supply with their center point junction connected the transistors base terminal as shown.



$$V_C = V_{CC} - R_C I_C = (V_E + V_{CE})$$
$$V_E = I_E R_E = V_B - V_{BE}$$
$$V_{CE} = V_C - V_E = V_{CC} - (I_C R_C + I_E R_E)$$
$$V_B = V_{BE} + V_E = V_{RB2} = \left(\frac{R_{B2}}{R_{B1}+R_{B2}}\right) V_{CC}$$
$$I_{B2} = \frac{V_B}{R_{B2}}$$
$$I_{B1} = I_B + I_{B2} = \frac{V_{CC} - V_B}{R_{B1}}$$
$$R_B = \frac{R_{B1} \times R_{B2}}{R_{B1}+R_{B2}} \qquad I_B = \frac{V_B - V_{BE}}{R_B + (1+\beta)R_E}$$
$$I_C = \beta_{(DC)} I_B$$
$$I_E = I_C + I_B = \frac{V_E}{R_E}$$

This voltage divider biasing configuration is the most widely used transistor biasing method, as the emitter diode of the transistor is forward biased by the voltage dropped across resistor R_{B2}. Also, voltage divider network biasing makes the transistor circuit independent of changes in beta as the voltages at the transistors base, emitter, and collector are dependant on external circuit values.

To calculate the voltage developed across resistor R_{B2} and therefore the voltage applied to the base terminal we simply use the voltage divider formula for resistors in series.

Generally the voltage drop across resistor R_{B2} is much less than for resistor R_{B1}. Then clearly the transistors base voltage V_B with respect to ground, will be equal to the voltage across R_{B2}.

The current flowing through resistor R_{B2} is generally set at 10 times the value of the required base current I_B so that it has no effect on the voltage divider current or changes in Beta.

SGBB GOVT POLYTECHNIC COLLEGE, SIROHI

MAX TIME- 1 HR CLASS TEST-II (SESSION 2017-18) MAX MARKS-15

SUB- COMPUTER SYSTEM ARCHITECTURE (CS202) DATE-29/01/2018

NOTE- ATTEMPT ALL THREE QUESTIONS.

1) Explain various addressing modes of computer.
2) Explain Timing and control unit of computer.
3) What is Stack Organization and its operations.

MODEL ANSWERS

Answer 1:-

The operation field of an instruction specifies the operation to be performed. This operation will be executed on some data which is stored in computer registers or the main memory. The way any operand is selected during the program execution is dependent on the addressing mode of the instruction. The purpose of using addressing modes is as follows:

1. To give the programming versatility to the user.
2. To reduce the number of bits in addressing field of instruction.

Types of Addressing Modes:-

Immediate Mode:-

In this mode, the operand is specified in the instruction itself. An immediate mode instruction has an operand field rather than the address field.

For example: ADD 7, which says Add 7 to contents of accumulator. 7 is the operand here.

Register Mode:-

In this mode the operand is stored in the register and this register is present in CPU. The instruction has the address of the Register where the operand is stored.

Advantages of this mode:

- Shorter instructions and faster instruction fetch.
- Faster memory access to the operand(s)

Disadvantages of this mode:

- Very limited address space

- Using multiple registers helps performance but it complicates the instructions.

Register Indirect Mode:-

In this mode, the instruction specifies the register whose contents give us the address of operand which is in memory. Thus, the register contains the address of operand rather than the operand itself.

Auto Increment/Decrement Mode:-

In this the register is incremented or decremented after or before its value is used.

Direct Addressing Mode:-

In this mode, effective address of operand is present in instruction itself. Single memory reference to access data. No additional calculations to find the effective address of the operand.

For Example: ADD R1, 4000 - In this the 4000 is effective address of operand.

NOTE: Effective Address is the location where operand is present.

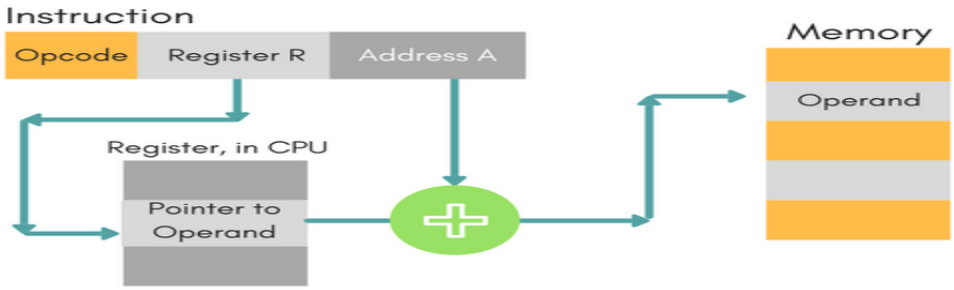
Indirect Addressing Mode:-

In this, the address field of instruction gives the address where the effective address is stored in memory. This slows down the execution, as this includes multiple memory lookups to find the operand.

Displacement Addressing Mode:-

In this the contents of the indexed register is added to the Address part of the instruction, to obtain the effective address of operand.

EA = A + (R), In this the address field holds two values, A (which is the base value) and R (that holds the displacement), or vice versa.



Relative Addressing Mode:-

It is a version of Displacement addressing mode. In this the contents of PC(Program Counter) is added to address part of instruction to obtain the effective address.

EA = A + (PC), where EA is effective address and PC is program counter.

The operand is A cells away from the current cell(the one pointed to by PC)

Base Register Addressing Mode:-

It is again a version of Displacement addressing mode. This can be defined as EA = A + (R), where A is displacement and R holds pointer to base address.

Stack Addressing Mode:-

In this mode, operand is at the top of the stack. For example: ADD, this instruction will *POP* top two items from the stack, add them, and will then *PUSH* the result to the top of the stack.

Answer 2:-

Design of Control Unit

Control unit generates timing and control signals for the operations of the computer. The control unit communicates with ALU and main memory. It also controls the transmission between processor, memory and the various peripherals. It also instructs the ALU which operation has to be performed on data.

Control unit can be designed by two methods which are given below:

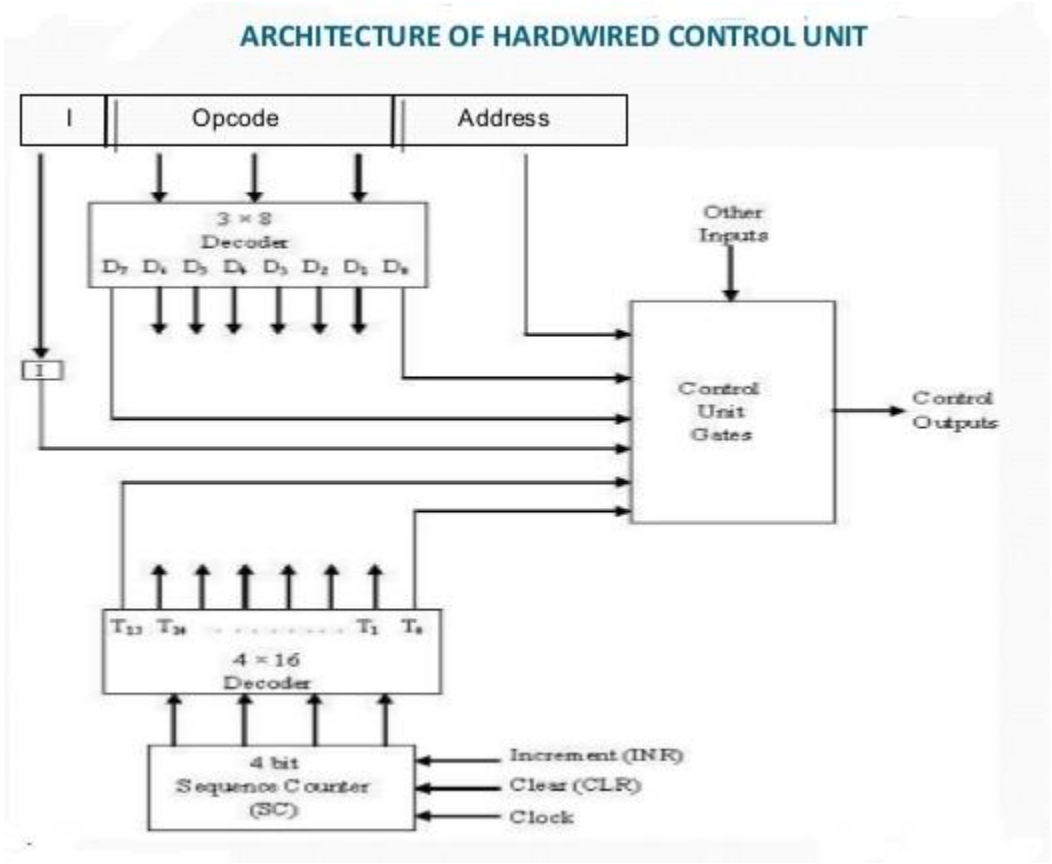
Hardwired Control Unit

It is implemented with the help of gates, flip flops, decoders etc. in the hardware. The inputs to control unit are the instruction register, flags, timing signals etc. This organization can be very complicated if we have to make the control unit large.

If the design has to be modified or changed, all the combinational circuits have to be modified which is a very difficult task.

Microprogrammed Control Unit

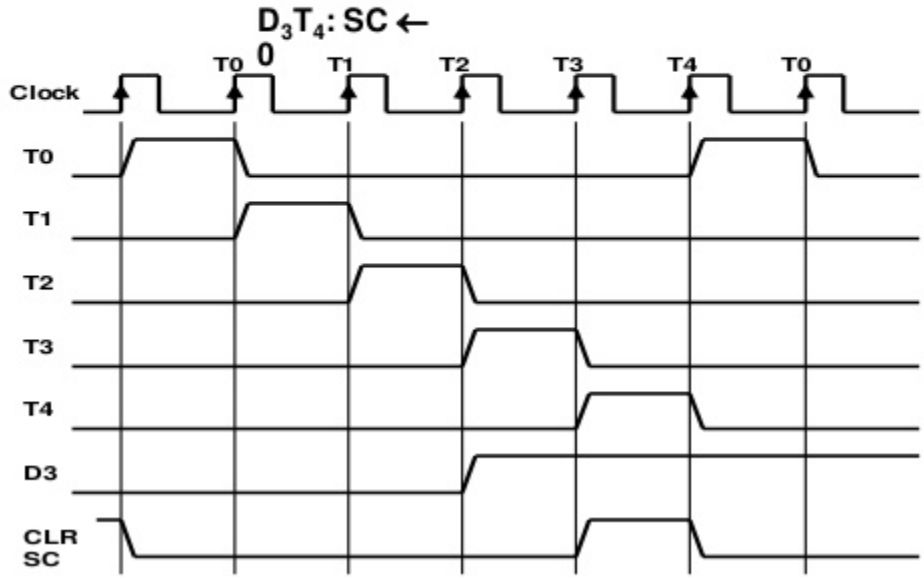
It is implemented by using programming approach. A sequence of micro operations is carried out by executing a program consisting of micro-instructions. In this organization any modifications or changes can be done by updating the micro program in the control memory by the programmer.



Timing Signals :-

- Generated by 4-bit sequence counter and 4×16 decoder
- The SC can be incremented or cleared.

- Example: $T_0, T_1, T_2, T_3, T_4, T_0, T_1, \ldots$
  Assume: At time T_4, SC is cleared to 0 if decoder output D3 is active.



Answer 3:-

To organize data we need some data structure like array stack link list etc. Program have different type of data , it can be stored in computer memory or CPU registers. data type who support different type of data and as well as easy manipulation. Stack fulfill these requirement. So we need stack.

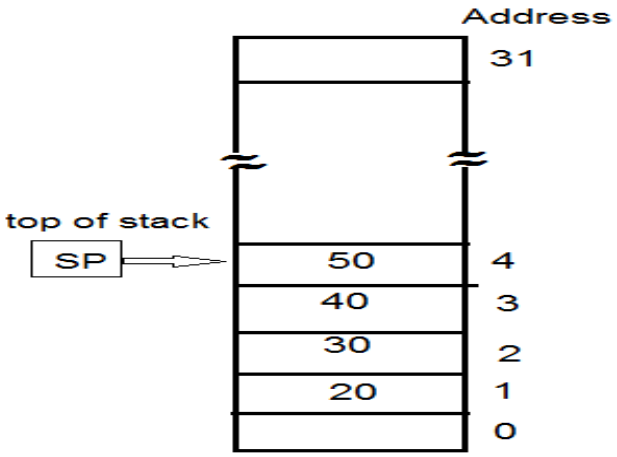
What is Stack:

1. A stack is a list of data elements, usually words or bytes, with the accessing restriction that the element can be added or removed at one end of the list only. This end is called the top of the stack and the other end is called the bottom of the stack.
2. Stack work on LIFO concept.
3. Placing new data element called PUSH
4. Removing the top data element called POP.

How computer uses stack?

1. Stack can be part of register unit or memory unit with a register that holds the address for the stack.
2. Part of register array or memory used for stack is called stack area.
3. Register which used to hold the address of stack is called stack pointer.
4. The value in the stack pointer always points at the top data element in the stack.

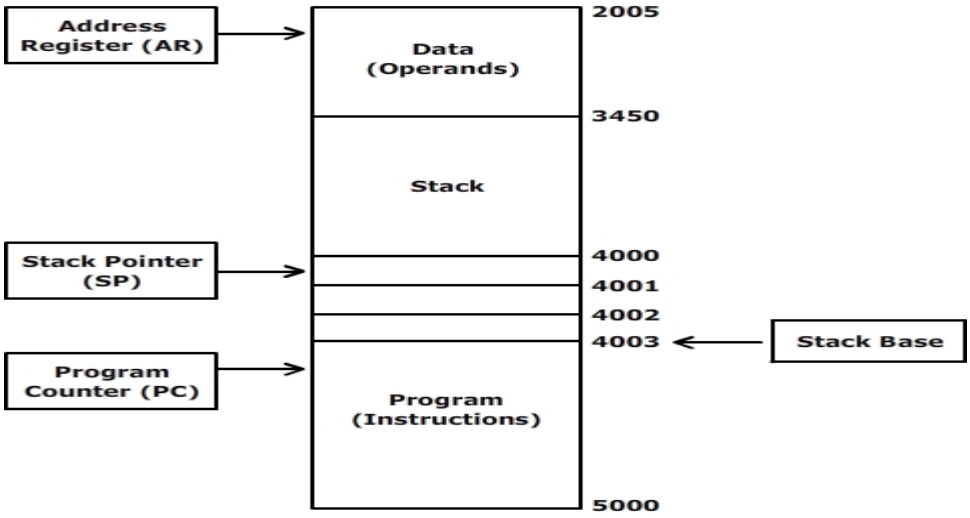
Register stack::



1. Four element store in stack
2. The data element 50 is top of stack, therefore the content Of SP is now 4.
3. The stack pointer is a 5 bit register, because 2^5=32.
4. Initialy it is clear to 0 and stack is said to be empty.
5. When data element is pushed on the stack, SP incremented.

Memory stack ::

1. Operation of memory stack is exactly similar to register stack. However it is implemented using computer memory instead of CPU register array.
2. Memory stack has an advantage of large size but the operation on it is slower than that of register stack.
3. This is because register stack is internal to the CPU and does not need any memory access.



The stack pointer SP initially points to 4003 and then on each ‘push’ operation, the value of SP is decremented to 4002, 4001, 4000 and so on. The last address available for stack is 3450.

Following operations are associated with the stack;-

Initialization Conditions:-

SP → 4003 // set stack pointer to memory address 4003

EMPTY → 1 // set EMPTY to 1 as initially stack is empty

FULL → 0 // clear FULL to 0 as stack contains no element

PUSH Operation:-

SP → SP - 1 // Stack Pointer is decremented

M[SP] → DR // Data from DR register is written at top of stack

Here, stack pointer is decremented so that it can point to the address of the next higher word. M[SP] refers to the memory location addressed by the value in SP. So the statement M[SP] ß DR writes the data from the data register DR to the top of stack.

POP Operation:-

DR → M[SP] // Read an item from top of stack

SP → SP + 1 // Increment stack pointer

For the pop operation, the data stored at the top of stack is read into the data register DR and the stack pointer is then incremented to point to the lower address word.

II Class Test Code CS203 Date 25.1.2018 MM 5x3=15

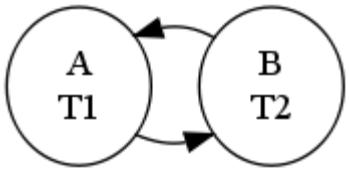
Solve All Questions.

Q1. What is a Deadlock?

A set of two or more processes are deadlocked if they are blocked (i.e., in the waiting state) each holding a resource and waiting to acquire a resource held by another process in the set.

or

A process is deadlocked if it is waiting for an event which is never going to happen.



Example:

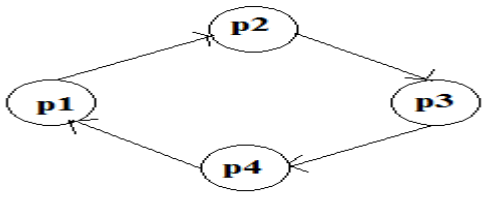
- a system has two tape drives
- two processes are deadlocked if each holds one tape drive and has requested the other

Q2. What is wait-for graph

A wait-for graph in computer science is a directed graph used for deadlock detection in operating systems and relational database systems.

Wait-for graph algorithm o track which other resources a process is currently blocking on. In a

If all resource types has only single instance, then we can use a graph called wait-for-graph,. Here, vertices represent processes and a directed edge from P1 to P2 indicate that P1 is waiting for a resource held by P2. A cycle in a wait-for-graph indicate a deadlock. So the system can maintain a wait-for-graph and check for cycles periodically to detect any deadlocks.



Q3. What is Deadlock Recovery

Once a deadlock is detected, we need to break the deadlock.

Deadlock Recovery
Traditional operating system such as Windows doesn't deal with deadlock recovery as it is time and space consuming process. Real time operating systems use Deadlock recovery.

Recovery method

1. Killing the process.

killing all the process involved in deadlock.

Killing process one by one. After killing each process check for deadlock again keep repeating process till system recover from deadlock.

2. Resource Preemption
Resources are preempted from the processes involved in deadlock, preempted resources are allocated to other processes, so that their is a possibility of recovering the system from deadlock.

Q4.What is Fragmentation:-

Fragmentation is position in memory which occurs in a dynamic memory allocation system when many of the free blocks are too small to satisfy any request.

External Fragmentation: External Fragmentation happens when a dynamic memory allocation algorithm allocates some memory and a small piece is left over that cannot be effectively used.

If too much external fragmentation occurs, the amount of usable memory is drastically reduced. Total memory space exists to satisfy a request, but it is not contiguous.

Internal Fragmentation: Internal fragmentation is the space wasted inside of allocated memory blocks because of restriction on the allowed sizes of allocated blocks.

Allocated memory may be slightly larger than requested memory; this size difference is memory internal to a partition, but not being used

Q5. What is paging? Why paging is used?

Paging is a memory management scheme

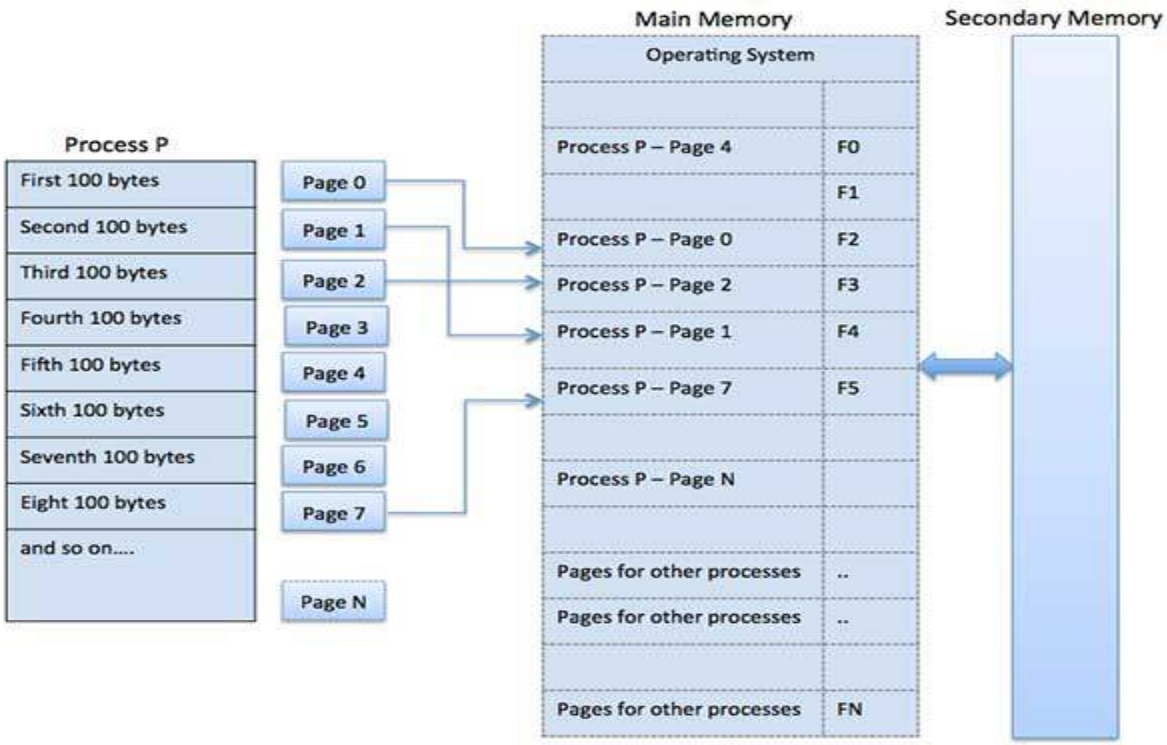
OS performs an operation for storing and retrieving data from secondary storage devices for use in main memory.

Data is retrieved from storage media by OS, in the same sized blocks called as pages. Paging allows the physical address space of the process to be non contiguous. The whole program had to fit into storage contiguously.

Paging is a memory management technique in which process address space is broken into blocks of the same size called **pages** (size is power of 2, between 512 bytes and 8192 bytes). The size of the process is measured in the number of pages.

Similarly, main memory is divided into small fixed-sized blocks of (physical) memory called **frames** and the size of a frame is kept the same as that of a page to have optimum utilization of the main memory and to avoid external fragmentation.

Paging technique plays an important role in implementing virtual memory.



Subject Code 207 Subject: - DBMS. MM-15

TEST –II

Q.1) Describe the various types of mapping constraints?

Mapping Cardinalities: express the number of entities to which another entity can be associated via a relationship. For binary relationship sets between entity sets A and B, the mapping cardinality must be one of:

1. One-to-one: An entity in A is associated with at most one entity in B, and an entity in B is associated with at most one entity in A.
2. One-to-many: An entity in A is associated with any number in B. An entity in B is associated with at most one entity in A.
3. Many-to-one: An entity in A is associated with at most one entity in B. An entity in B is associated with any number in A.
4. Many-to-many: Entities in A and B are associated with any number from each other.

The appropriate mapping cardinality for a particular relationship set depends on the real world being modeled.

Q.2) What are the various types of integrity constraints?

Integrity Constraints

1. Integrity constraints provide a way of ensuring that changes made to the database by authorized users do not result in a loss of data consistency.
2. We saw a form of integrity constraint with E-R models:
    - **Key declarations:** stipulation that certain attributes form a candidate key for the entity set.
    - **Form of a relationship:** mapping cardinalities 1-1, 1-many and many-many.
3. An integrity constraint can be any arbitrary predicate applied to the database.
4. They may be costly to evaluate, so we will only consider integrity constraints that can be tested with minimal overhead.

Various types of integrity constraints are-

1. Domain Integrity
2. Entity Integrity Constraint
3. Referential Integrity Constraint
4. Key Constraints

Q.3) What do you mean by relational algebra?

The relational algebra is a theoretical procedural query language which takes an instance of relations and does operations that work on one or more relations to describe another relation without altering the original relation(s).

The primary operations of relational algebra are as follows:

- Select
- Project
- Union
- Set different
- Cartesian product
- Rename

Projection Operator (∏): Projection operator is used to project particular columns from a relation. Syntax:

$\prod_{(Column\ 1,Column\ 2....Column\ n)}$(Relation Name)

Selection operator (σ): Selection operator is used to select tuples from a relation based on some condition. Syntax:

$\sigma_{(Cond)}$(Relation Name)

Cross Product(X): Cross product is used to join two relations. For every row of Relation1, each row of Relation2 is concatenated. If Relation1 has m tuples and and Relation2 has n tuples, cross product of Relation1 and Relation2 will have m X n tuples. Syntax:

Relation1 X Relation2

Union (U): Union on two relations R1 and R2 can only be computed if R1 and R2 are **union compatible** (These two relation should have same number of attributes and corresponding attributes in two relations have same domain) . Union operator when applied on two relations R1 and R2 will give a relation with tuples which are either in R1 or in R2. The tuples which are in both R1 and R2 will appear only once in result relation. Syntax:

Relation1 U Relation2

Minus (-): Minus on two relations R1 and R2 can only be computed if R1 and R2 are **union compatible**. Minus operator when applied on two relations as R1-R2 will give a relation with tuples which are in R1 but not in R2. Syntax:

Relation1 - Relation2

Rename(ρ): Rename operator is used to give another name to a relation. Syntax:

ρ(Relation2, Relation1)

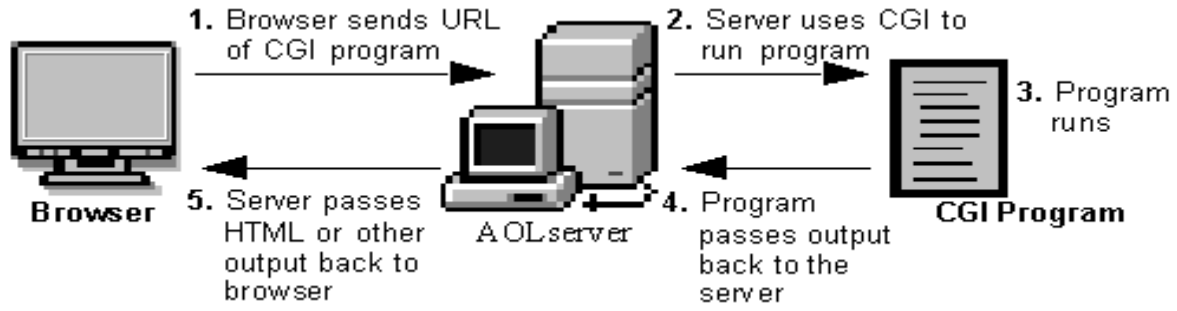
Q.4) what do you mean by concurrency control?

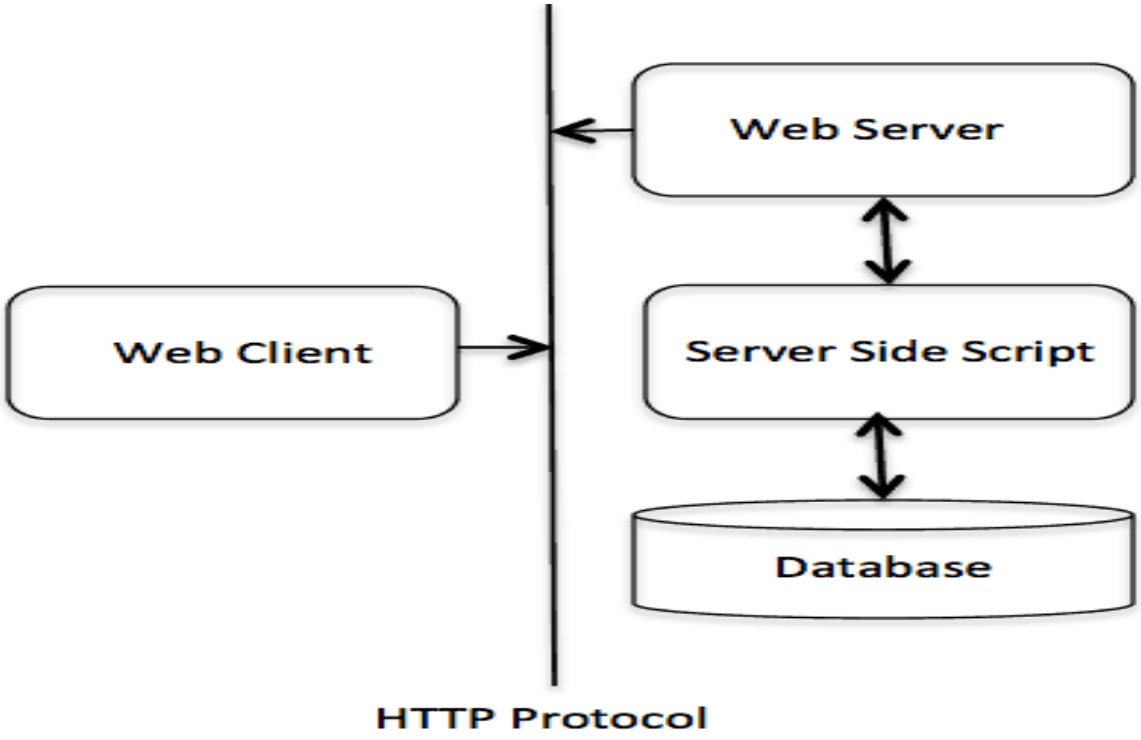
Concurrency control is a database management systems (DBMS) concept that is used to address conflicts with the simultaneous accessing or altering of data that can occur with a multi-user system. Concurrency control, when applied to a DBMS, is meant to coordinate simultaneous transactions while preserving data integrity.

Subject Code 209 Internet & Web Technology. MM-15

TEST -II

Q.1) Explain the working of CGI with GET & POST Method .





Example :.

GET Method web form uses the HTTP **GET request method** to submit data to the web server.

The GET method sends results to the web server by encoding them as part of the URL.

```
<FORM action = "/cgi-bin/hello_get.cgi" method = "GET">
   First Name: <input type = "text" name = "first_name">  <br>

   Last Name: <input type = "text" name = "last_name">
   <input type = "submit" value = "Submit">
</FORM>
```

The POST Method

. This method directs the data to the standard input port of the CGI script, allowing your CGI script to read it directly.

To use the post method, must add a POST attribute to the FORM tag that appears in your HTML file. For example:

<FORM ACTION="http://www.rebol.com/cgi-bin/test-cgi.cgi"

METHOD="post">

Q.2) Explain the File Handling in Perl

Perl File handling is used in accessing file such as text files, log files or configuration files.

Perl file handles are capable of creating, reading, opening and closing a file.We are creating a file, **file1.txt** with the help of open() function.

The $fh (file handle) is a scalar variable and we can define it inside or before the open() function. Here we have define it inside the function. The '>' sign means we are opening this file for writing. The **$filename** denotes the path or file location.

Once file is open, use $fh in print statement. The print() function will print the above text in the file.

Now we are closing $fh. Well, closing the file is not required in perl. file will be automatically closed when variable goes out of scope.

```
    my $filename = 'file1.txt';
    open(my $fh, '>', $filename) or die "Could not open file '$filename' $!";
    print $fh "Hello!! We have created this file as an example\n";
    close $fh;
    print "done\n";
Example :
    print "What is your age?\n";
    $age = <STDIN>;
    if($age >= 18)
```

```
{
  print "You are eligible to vote.\n";
}
Else
 {
  print "You are not eligible to vote.\n";
}
```

OR

Q.2) what is the use of <LI> tags that can be used when designing a page ?

HTML lists are used to present list of information in well formed and semantic way. There are three different types of list in HTML and each one has a specific purpose and meaning:

Unordered list — Used to group a set of related items, in no particular order.
Ordered list — Used to group a set of related items, in a specific order.

HTML Unordered Lists
An unordered list created using the <ul> tag, and each list item starts with the <li> tag. The list items in unordered lists are marked with bullets (small black circles), by default.

```
<ul>
   <li>Chocolate Cake</li>
   <li>Black Forest Cake</li>
   <li>Pineapple Cake</li>
</ul>
```

HTML Ordered Lists
An ordered list, created using the <ol> tag, and each list item starts with the <li> tag. Ordered list contain information where order should be emphasized.
The list items in ordered lists are marked with numbers.

```
<ol>
   <li>Mix ingredients</li>
   <li>Bake in oven for an hour</li>
   <li>Allow to stand for ten minutes</li>
</ol>
```

The type attribute of the <ol> tag, defines the type of the list item marker:

Type	Description
type="1"	The list items will be numbered with numbers (default)
type="A"	The list items will be numbered with uppercase letters

type="a"	The list items will be numbered with lowercase letters
type="I"	The list items will be numbered with uppercase roman numbers
type="i"	The list items will be numbered with lowercase roman numbers

Q.3) Write a program in JavaScript to find the sum of two number .

```
<html>
<head>
<script>
function add(){
var a,b,c;
a=Number(document.getElementById("first").value);
b=Number(document.getElementById("second").value);
c= a + b;
document.getElementById("answer").value= c;
}
</script>
</head>
<body>
<input id="first">
<input id="second">
<button onclick="add()">Add</button>
<input id="answer">
</body>
</html>
```

OR

Q.3) what is the significance of Table in HTML .

The HTML table allows to arrange data -- text, t, images, links, forms,, etc. -- into rows and columns of cells
A simple HTML table, containing two columns and two rows:

```
<table>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
</table>
```

II Class Test Code CS210 Date 27.1.2018 MM 5x3=15

Solve All Questions.

Q1. What is cathode ray tube (CRT

The **cathode ray tube** (**CRT**)

Cathode Ray Tube (CRT) is a computer display screen, used to display the output in a standard composite video signal. The working of CRT depends on movement of an electron beam which moves back and forth across the back of the screen. The source of the electron beam is the electron gun; the gun is located in the narrow, cylindrical neck at the extreme rear of a CRT which produces a stream of electrons through thermionic emission. Usually, A CRT has a fluorescent screen to display the output signal

Cathode

The heater keeps the cathode at a higher temperature and electrons flow from the heated cathode.

The Control Grid

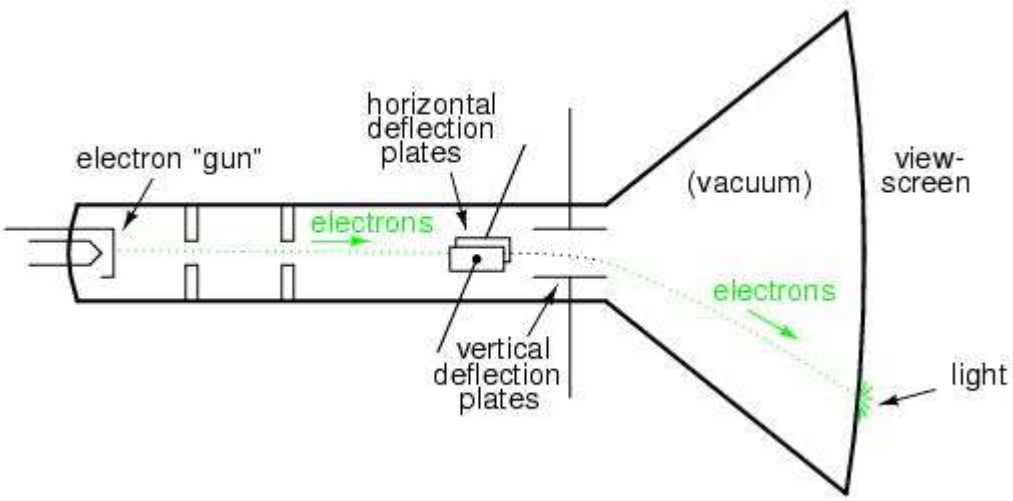
The control grid regulates the brightness of the spot on the screen.

Deflecting Plates

Two pairs of deflecting plates allow the beam of electrons. An electric field between the first pair of plates deflects the electrons horizontally, and an electric field between the second pair deflects them vertically,

Screen

This is a display device called screen. Screen is coated with special type of fluorescent material. Fluorescent material absorbs its energy and re-emits light in the form of photons when electron beam hits the screen.



Q2. What is *Flash Drive*?

Flash drive is any data storage device that holds data with non-movable parts.

It is a type of EEPROM memory chip that retains information without power.

Inside the chip data is stored in cells. Cell charges is cleared or flashed with the help of Tunneling electrons.

Flash memory uses NOR and NAND technologies.

Q3. Explain working of Wireless Key board: -

wireless keyboard is powered by battery. Keyboard also comes with a USB dongle or device which is inserted in the USB port of the computer. The dongle or the device functions as radio frequency (RF) receiver.

The keyboard has its own processor and circuitry called key matrix. The key matrix is a collection of circuits under the keyboard.. When you press any particular key, it completes this circuit, thus, enabling the processor to determine the location of the key that was pressed. This key is identified and corresponding radio signals is sent to processor via dongle.

Q4 Explain Working of *scanner*

A ***scanner*** is an input device that scans documents such as photographs and pages of text. When a document is scanned, it is converted into a digital format.

Scanners uses Charge-coupled device [CCD]. A CCD sensor is used to capture the light from the scanner and then convert it into the proportional electrons. The charge developed will be more if the intensity of light that hits on the sensor is more.

A scanner consists of a flat transparent glass bed under which the CCD sensors, lamp, lenses, filters and also mirrors are fixed. The document has to be placed on the glass bed.

Stepper motor under the scanner moves the scanner head from one end to the other.. The scanner head consists of the mirrors, lens, CCD sensors and also the filter. The scan head moves parallel to the glass bed and that too in a constant path. The scan head moves from one end of the machine to the other. When it has reached the other end the scanning of the document has been completed.

As the scan head moves under the glass bed, the light from the lamp hits the document and is reflected back with the help of mirrors angled to one another.. The CCD sensors convert the light to electrical signals according to its intensity.

The electrical signals will be converted into image format inside a computer and saved.

Q5 what is MICR?

MICR (magnetic ink character recognition) is a technology used to verify the originality of paper documents, especially checks. Special ink, which is sensitive to magnetic fields.

The ink used in the printing is a magnetic, usually containing iron oxide. The MICR text is passed before a MICR reader. The ink in the plane of the paper is first magnetized. Then the characters are passed over a MICR read head.