Assignment
AFC

Q.1) What is an algorithm ?
→

* An algorithm is a step-by-step procedure or set of rules to solve a specific problem or perform a task.
* It takes an input, processes it through a series of well-defined steps, and produces an output.

* Key characteristics of an Algorithm :→
1) Definiteness :→ Each step is clear and unambiguous.
2) Finiteness :→ It should end after a finite number of steps.
3) Input :- It takes zero or more inputs.
4) output :- It produces at least one output.
5) Effectiveness :- Each step is basic enough to be carried out using known methods.

* Example :- Algorithm to Add Two Numbers :-
1) start
2) Take two numbers as input
3) Add the numbers.
4) Display the sum
5) End

* Algorithms are used in computer programming, data structures, artificial intelligence, and many real-life applications like sorting, searching, and decision-making processes.

Q.2) Explain need of an algorithm ?
→

* Need for an Algorithm :-
* Algorithms are essential in computing and problem-solving because they provide a structured approach to solving problems efficiently.
* Below are some key reasons why algorithms are needed :-

1) Efficiency :-
* Algorithms help optimize performance by reducing time and space complexity.
* for example, searching for an item in a sorted list is much faster using Binary search (O(log n)) compared to Linear search (O(n)).

2) Correctness & Reliability :-
* A well-designed algorithm ensures accurate results every time.
* Example :- Banking systems use secure algorithms for transactions to avoid errors.

3) Scalability :-
* Algorithms help manage large datasets efficiently.
* Example :- Social media platforms use sorting and recommendation algorithms to handle millions of user's data.

4) Reusability :-
* Once an algorithm is created, it can be reused in multiple applications with slight modifications.
* Example :- The Merge sort algorithm is used in databases, operating system and AI applications.

5) Optimization :-
* Algorithms help optimize resources like memory & CPU time.
* Example :- Dijkstra's Algorithm is used in Google Maps to find the shortest path between locations.

6) Automation :-
* Algorithms enable automation of repetitive tasks.
* Example :- AI-powered chatbots follow predefined algorithms to respond to users.

* In short, algorithms are the back-bone of computing, ensuring efficiency, correctness, and reliability in solving real-world problems.

Q.3) Write an algorithm to find average age of a group of 10 players?

→

* Algorithm to find the Average Age of a Group of 10 players :-
* step-by-step Algorithm :-

1) start.
2) Initialize a variable SUM = 0 to store the total age.
3) Repeat for i = 1 to 10 :
   Take the age of player i as input.
   Add the age to SUM.
4) Calculate the average age :
   average = SUM / 10
5) Display the average age.
6) End.

* Example Execution :-
- Input : 10 player ages → 20, 22, 21, 24, 25, 23, 22, 26, 24, 21.
- SUM = 20 + 22 + 21 + 24 + 25 + 23 + 22 + 26 + 24 + 21 = 228
- Average = 228 / 10 = 22.8
- output : Average Age = 22.8

* This algorithm efficiently calculates the average age using basic arith-metic operations.

Q.4) Write algorithm to this problem :- Ramshewak goes to market for buying some fruits and vegetables. He is having a currency of Rs. 500 with him for mar-keting. From a shop he purchases 2.0 kg Apple priced Rs. 50.0 per kg, 1.5 kg Mango priced Rs. 35.0 per kg, 2.5 kg Potato priced Rs. 10.0 per kg; and 1.0 kg Tomato priced Rs. 15 per kg. He gives the currency of Rs. 500 to the shopkeeper. Find out the amount shop-keeper will return to Ramshewak and also tell the total item purchased.

→

* Algorithm to Calculate shopkeeper's Return Amount & Total Items Purcha-sed :-
- step-by-step Algorithm :-

1) start.
2) Initialize total_Money = 500 (Ramshewak's Money).

3) Define item details :-
Apple : 2.0 kg x Rs. 50/kg = Rs. 100
Mango : 1.5 kg x Rs. 35/kg = Rs. 52.5
Potato : 2.5 kg x Rs. 10/kg = Rs. 25
Tomato : 1.0 kg x Rs. 15/kg = Rs. 15

4) Calculate total cost :
total_cost = 100 + 52.5 + 25 + 15 = 192.5

5) Compute remaining amount :
return_amount = total_money - total_cost
= 500 - 192.5
= 307.5

6) Compute total items purchased :
total_items = 2.0 + 1.5 + 2.5 + 1.0 =
7.0 kg

7) Display return_amount and total_items

8) End.

* output :
Total items purchased : 7.0 kg
Amount to be returned by shopkeeper
:- Rs. 307.5

* This algorithm efficiently calculates both the total quantity of purchased items and the change Ramshewak should receive.

Q.5) Find factorial of N ?
→

* Algorithm to find factorial of N :-

- The factorial of a number N (denoted as N!) is the product of all positive integers from 1 to N.
- Formula :-
N! = N x (N-1) x (N-2) x .... x 1
5! = 5 x 4 x 3 x 2 x 1 = 120

* Algorithm :-
1) Start
2) Input N (number whose factorial is to be found).
3) Initialize fact = 1.
4) Repeat for i = 1 to N.
fact = fact x i
5) Display fact.
6) End.

* Example Execution :-
- Input :-
N = 5
- Steps :-

fact = 1 x 1 = 1
fact = 1 x 2 = 2
fact = 2 x 3 = 6
fact = 6 x 4 = 24
fact = 24 x 5 = 120

- output :-
  Factorial of 5 = 120

Q.6) Explain steps involve in drawing of a flowchart.

→ * steps involved in Drawing a Flowchart:

- A Flowchart is a graphical representation of an algorithm using symbols to illustrate the sequence of steps required to solve a problem.
- Below are the key steps involved in drawing a Flowchart :-

1) Define the Problem :-
- clearly understand the problem that the flowchart will represent.
- Identify the input, process, and expected output.
- Example :- Find the factorial of a number.

2) Identify Inputs and outputs :-
- Determine what data (inputs) is required.
- Identify what result (output) should be produced.

3) Determine the Process steps :-
- Break down the problem into sequential steps.
- use decision-Making conditions where necessary.

4) select Flowchart Symbols :-
- oval (Terminator) :- start & End (start, End).
- Parallelogram (Input / output) : used for input & output.
- Rectangle (Process) :- for calculations and operations.
- Diamond (Decision) :- for conditions (if statements).
- Arrows :- show the flow of execution.

5) Arrange symbols in logical order :-
- Place the symbols in a top-to-bottom or left-to-right sequence.
- Ensure the flow of control follows a logical order.

6) connect symbols with Arrows :-
- use arrows to indicate the flow of execution from one step to another.
- Ensure clarity and avoid unnecessary crossings.

7) Review and Test the flowchart :-

- check if the flowchart correctly represents the algorithm.
- Ensure it covers all possible cases and is free of errors.

* This structured approach ensures that the flowchart is clear, logical, and easy to understand.

Q.7) Explain uses of flowchart.

→

* A flowchart is a graphical representation of an algorithm or process. It helps visualize the sequence of steps involved in problem-solving.
* Here are some key uses of flowcharts:

1) understanding and Analyzing a process :-
- flowcharts help in breaking down complex processes into simple, easy-to-understand steps.
- Example:- understanding how a billing system works in a store.

2) Algorithm and Program Development :-
- used by programmers to design algorithms before writing actual code.

3) Debugging and Error Identification :-
- Helps identify logical errors in an algorithm before implementation.

4) Documentation of Processes :-
- flowcharts are used in technical documentation to describe system workflows.

5) communication and Training :-
- Helps explain processes to team members, new employees, or students in an easy way.

6) Decision-Making and Problem-solving :-
- Helps in making logical decisions by visualizing different scenarios.

7) Business Process Modeling :-
- used in industries to design efficient workflows and improve productivity.

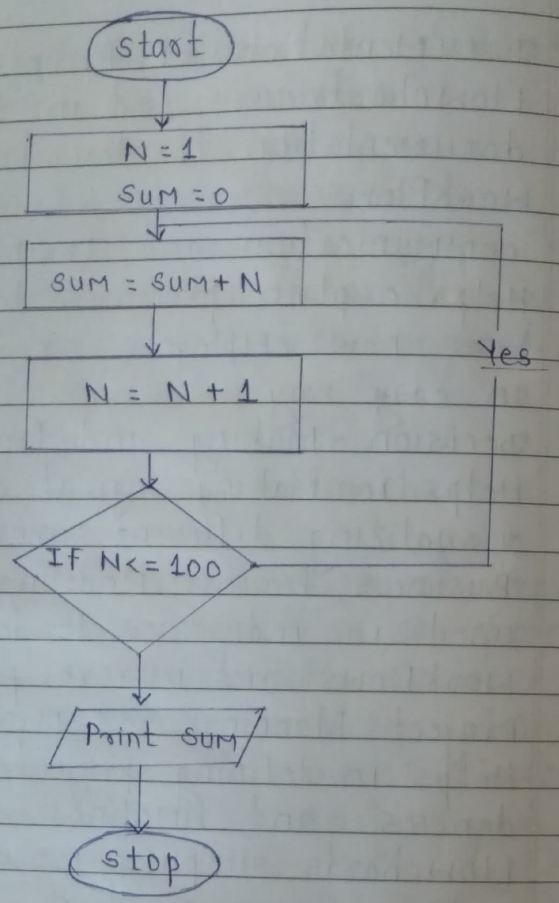
8) Project Planning and Management :-
- Helps in defining project steps, dependencies, and timelines.

* flowcharts simplify problem-solving, enhance communication, and improve efficiency across various fields like programming, business, healthcare, and project management.

Q.8) Draw a flowchart to find the sum of first 100 natural numbers.

→

* flowchart to print sum of first 100 natural numbers :-



Q.9) Draw a Flowchart to find the largest of three numbers x, y and z.

→ • Firstly we will start with reading all three numbers.

• consider the three numbers be A, B and C.

• Then in the next step, we will compare first two numbers A > B ?

• If A > B is true we will compare A with C.

• If A > C is true then A is the largest number.

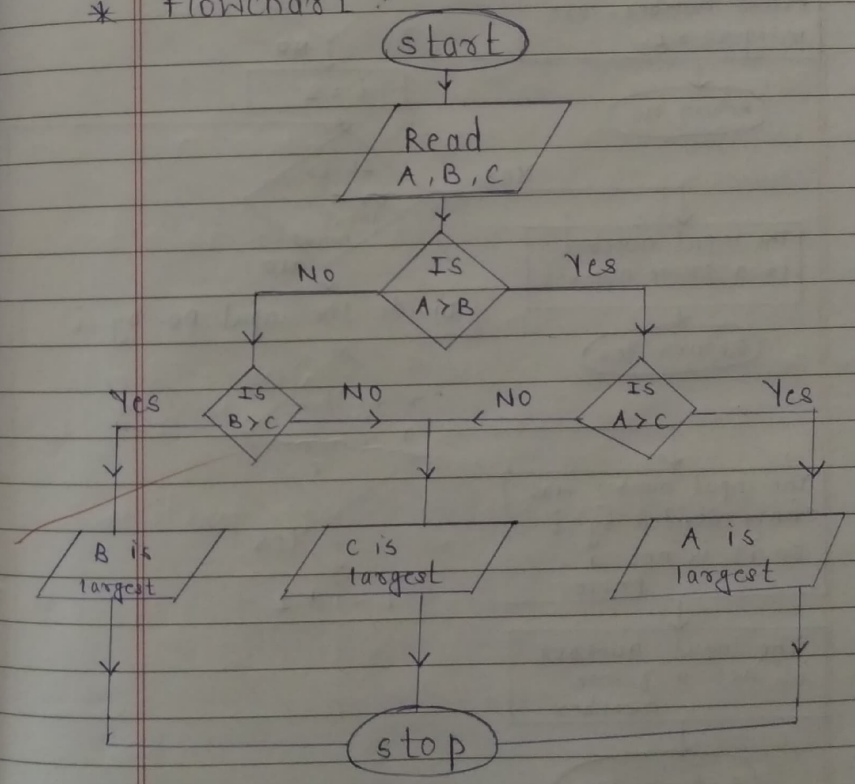
• If A > C is False then C is the largest number.

• If A > B is False we will compare B with C.

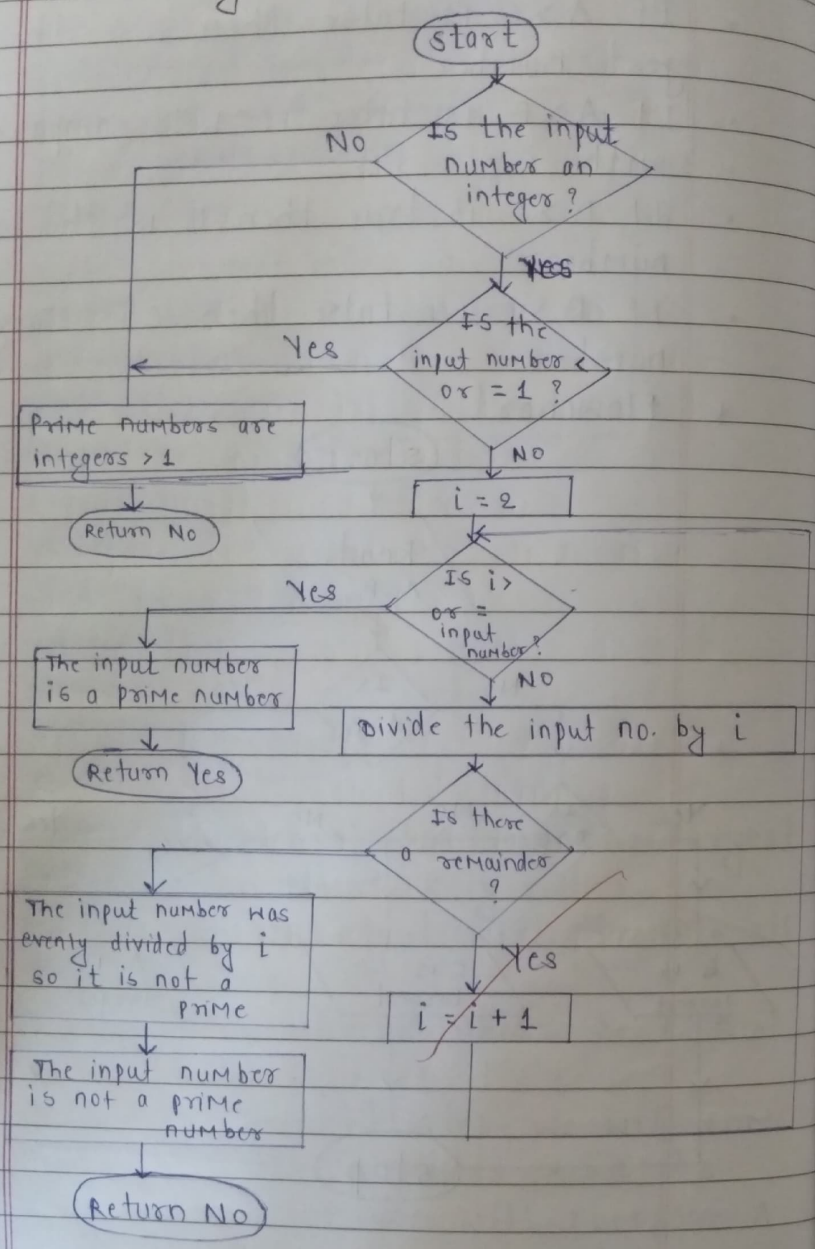
• If B > C is true then B is the largest number.

• If B > C is False then C is largest number.

* Flowchart :-

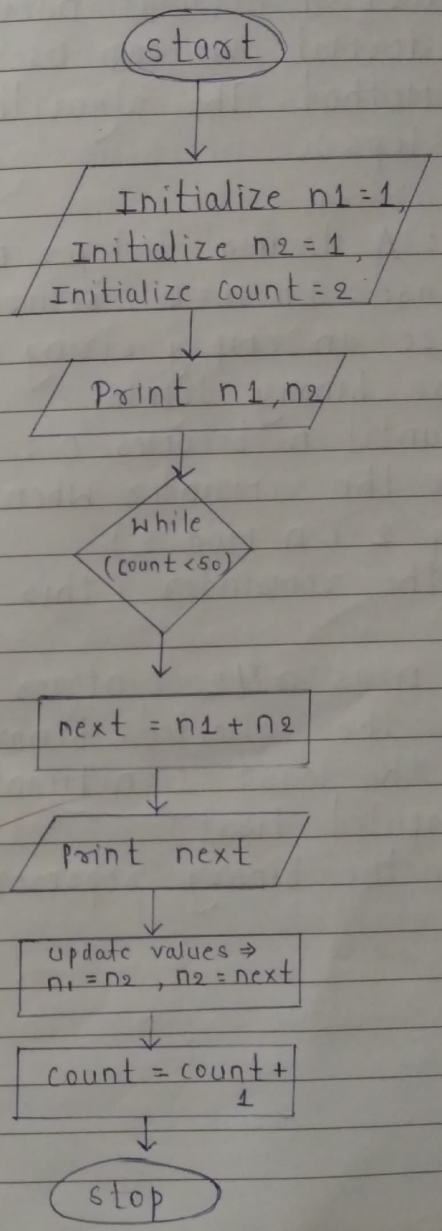


Q.10) Draw flowchart for the problem of determining prime number ?



Q.11) Draw a flowchart which generates first 50 items of the Fibonacci series:- 1, 1, 2, 3, 5, 8, ... ?

→



Q.12) Design an algorithm to convert a decimal number, n, to binary format?

→

* To convert a decimal number n to binary format, we can use the Division by 2 Method. The algorithm follows these steps :-

1) Input : A decimal integer n.
2) Edge case : If n = 0, return "0".
3) Initialize an empty string or stack to store binary digits.
4) Repeat until n becomes 0 :-
   - compute the remainder when n is divided by 2 (n Mod 2).
   - store the remainder (this is a binary digit).
   - update n as n//2 (integer division).
5) Reverse the collected binary digits (since the least significant bit (LSB) is computed first).
6) output the binary representation.

Assignment
* AFC :-

Q.1) Enlist the applications of C++.

→

* C++ is a powerful and versatile programming language used in various fields due to its efficiency, performance, and object-oriented features.
* Here are some key applications of C++ :-

1) System Programming :-
   - operating systems (e.g., windows, Linux components).
   - Device drivers.
   - Embedded systems.
2) Game Development :-
   - Game engines (e.g., Unreal Engine).
   - Graphics rendering (e.g., DirectX, openGL).
3) Software Development :-
   - Desktop applications (e.g., Adobe photoshop, Microsoft office components).
   - GUI applications (e.g., Qt, GTK, WxWidgets).
4) Competitive Programming & Algorithms :-
   - used in coding competitions (e.g., codeforces, LeetCode) due to its speed and STL (standard Template Library).
5) Embedded systems & IoT :-
   - Automotive software (e.g., self-driving car systems).

- Medical devices.
- consumer electronics.

6) Financial and Banking Systems :-
- Trading applications.
- High-frequency trading (HFT) systems.
- Risk analysis tools.

7) Web Browsers and Rendering Engines :-
- Google chrome (chromium).
- Mozilla Firefox (Gecko Engine).

8) Database Management Systems (DBMS) :-
- MySQL
- PostgreSQL

Q.2) Explain types of variables used in C++.

→

* In C++, variables store data of different types, and each type determines the size and nature of the data that can be stored.
* Below are the main types of variables used in C++ :-

1) Local variables :-
- Declared inside a function or block.
- only accessible within that function or block.
- Example :-

C++ :-

```
void example () {
    int x = 10;     // Local variable
    cout << x;
}
```

2) Global variables :-
- Declared outside all functions.
- Accessible throughout the program.
- Example :- C++ :-

```
int x = 100;   // Global variable
void display () {
    cout << x;   // Accessible here.
}
```

3) static variables :-
- Retains its value between function calls.
- Initialized only once.
- Example :-

```
void example () {
    static int count = 0;   //static variable
    count++;
    cout << count << endl;
}
```

4) Constant variables (const) :-
- value cannot be changed after initialization.
- used for fixed values.

- Example :-
  const Float PI = 3.14 ;

5) Volatile Variables :-
- Tells the compiler not to optimize the variable as it may change unexpectedly.
- used in embedded systems or hardware programming.
- Example :-
  volatile int sensorValue ;

6) Register Variables :-
- stored in CPU registers for faster access.
- Example :-

```
void example () {
    register int fastVar = 5 ; // stored
    in CPU register
}
```

Q.3) Describe 10 keywords of C++.
→ * Here are 10 important keywords in C++ along with their descriptions :-

1) int → Defines an integer variable that stores whole numbers (e.g., int a = 10 ;).

2) Float → Declares a Floating-point variable for decimal values with single precision (e.g., Float pi = 3.14 ;).

3) double → similar to Float but with higher precision for storing decimal numbers (e.g., double num = 3.141592 ;).

4) char → stores a single character enclosed in single quotes (e.g., char letter = 'A' ;).

5) bool → Represents a boolean value, either true or false (e.g., bool isOn = true ;).

6) if → used for decision-making ; executes a block of code if a condition is true (e.g., if (x > 0) { cout << "positive" ; }).

7) else → works with if ; runs a block of code if the if condition is false (e.g., else { cout << "Negative" ; }).

8) for → A loop that repeats code a fixed number of times (e.g., for (int i = 0 ; i < 5 ; i++) { cout << i ; })

9) While → A loop that runs as long as the condition remains true (e.g., while (x>0) { x--; }).

10) return → Ends a function and optionally returns a value (e.g., return 0; in main() signals successful execution).

Q.4) Elaborate variable scope in C++.
→

* Variable scope in C++ defines the region of a program where a variable can be accessed or modified. It determines the lifetime and visibility of a variable. There are four main types of scopes in C++ :-

1) Local scope (Block Scope) :-
- A variable declared inside a function or block { } is local to that block.
- It cannot be accessed outside the function / block.
- The variable exists only until the block executes.

* Example :-

```
#include <iostream>
using namespace std;
void display () {
    int x = 10;   // Local variable
    cout << "x inside function :" <<
  x << endl;
}
int Main () {
    display ();
    // cout << x ;  // ERROR : x is not
 accessible here.
    return 0;
}
```

2) Global scope :-
- A variable declared outside all functions is a global variable.
- It can be accessed and modified from any function in the same file.
- It remains in memory throughout the program execution.

* Example :-

```
# include <iostream>
 using namespace std;
 int globalVar = 50;   // Global variable
 void display () {
     cout << "Global variable :" <<
   globalVar << endl;
 }
int Main () {
     display ();
     cout << "Accessing Global
 variable in Main :" << globalVar << endl;
```

```
    return 0;
}
```

* Note :- Global variables should be used carefully to avoid unwanted Modifi-cations.

3) Function scope :-

- A function's name is visible within its defined file (or scope).
- A function variable (declared inside a function) cannot be accessed outside it.

* Example :-

```
#include <iostream>
using namespace std;
void show () {
    int funcVar = 20;  // function-scope variable
    cout << "Inside function : " << funcVar << endl;
}

int Main () {
    show ();
    // cout << funcVar;  // ERROR : funcVar is not accessible here
    return 0;
}
```

4) Static scope (Persistent storage scope) :-

- A variable declared with static inside a function retains its value between function calls.
- The value is initialized only once and persists throughout program execution.

* Example :-

```
#include <iostream>
using namespace std;

void counter () {
    static int count = 0;  // static variable.
    count++;
    cout << "count : " << count << endl;
}

int Main () {
    counter ();  // output : count : 1
    counter ();  // output : count : 2
    counter ();  // output : count : 3

    return 0;
}
```

Q.5) Explain the following operators used in C++ with example :-

- Arithmetic operators.
- Assignment operators
- comparison operators.
- logical operators
- Bitwise operators.

→

* Operators in C++ perform operations on variables and values.

1) Arithmetic operators :-

* used to perform basic mathematical operations.

operator	Description	Example (a = 10, b = 5)	output
+	Addition	a + b	15
-	subtraction	a - b	5
*	multiplication	a * b	50
/	Division	a / b	2
%	Modulus (remainder)	a % b	0

* Example code :-

```
# include <iostream>
using namespace std ;
int Main () {
    int a = 10 , b = 5 ;
    cout << " Addition : " << (a+b)
<< endl ;
    cout << "subtraction : " << (a - b)
<<endl ;
    cout << " Multiplication : " << (a * b)
<<endl ;
    cout << " Division : " << (a / b) <<
endl ;
    cout << " Modulus : " << (a % b) <<
endl ;
    return 0 ;
}
```

2) Assignment operators :-

* used to assign values to variables.

Operator	Description	Example (a = 10)	Equivalent To
=	Assign	a = 10	a = 10
+=	Add & assign	a += 5	a = a + 5
-=	Subtract & assign	a -= 3	a = a - 3
*=	Multiply & assign	a *= 2	a = a * 2
/=	Divide & assign	a /= 2	a = a / 2
%=	Modulus and assign	a %= 3	a = a % 3

* Example code :-
* C++ :-

```
# include <iostream>
using namespace std;

int Main () {
    int a = 10;
    a += 5;
    cout << "After += : " << a <<
endl;
    a -= 3;
    cout << "After -= : " << a <<
endl;
    return 0;
}
```

3) Comparison operators :-

* Used to compare values (returns true or false).

operator	Description	Example (a = 10, b = 5)	output
==	Equal to	a == b	False
!=	Not equal to	a != b	true
>	Greater than	a > b	true
<	less than	a < b	False
>=	Greater than or equal to	a >= b	true
<=	Less than or equal to	a <= b	False

* Example Code :-

```
# include <iostream>
using namespace std;

int Main () {
    int a = 10 , b = 5;
    cout << (a > b) << endl;
    // output : 1 (true)
    cout << (a < b) << endl;
    // output : 0 (false)
    return 0;
}
```

4) Logical operators :-

* used to perform logical operations (true or False).

Operator	Description	Example (a = true, b = false)	output
&&	Logical AND	a && b	False
||	Logical OR	a || b	true
!	Logical NOT	!a	false

* Example code :-

```
#include <iostream>
using namespace std;

int Main () {
    bool a = true , b = false ;
    cout << (a && b) << endl;
    // False (0)
    cout << (a || b) << endl;
    // true (1)
    cout << !a << endl;
    // false (0)
    return 0;
}
```

5) Bitwise operators :-

* used for bit-level operations.

Operator	Description	Example (a=5, b=3)	Binary calculation	output
&	Bitwise AND	a & b	0101 & 0011 → 0001	1
|	Bitwise OR	a | b	0101 | 0011 → 1000	8
^	Bitwise XOR	a ^ b	0101 ^ 0011 → 0110	6
~	Bitwise NOT	~a	~0101 → 1010	-6
<<	Left shift	a << 1	0101 << 1 → 1010	10
>>	Right shift	a >> 1	0101 >> 1 → 0010	2

* Example Code :-

```
#include <iostream>
using namespace std;

int Main () {
    int a = 5 , b = 3 ;
    cout << "Bitwise AND :" << (a &
 b) << endl;
    cout << "Bitwise OR :" <<(a | b)
 << endl;
    cout << "Bitwise XOR :" <<(a ^ b)
 << endl;
    cout << "Bitwise NOT :" << (~a)
 << endl;
    cout << "Left shift :" <<(a << 1)
 << endl;
    cout << "Right shift :" <<(a >>
 1) << endl;
    return 0;
}
```

Q.6) Write a CPP to calculate compound interest on "x" principal amount.

→

* formula for compound Interest :-

$$A = P \times \left(1 + \frac{r}{100}\right)^t$$

- A = final Amount
- P = Principal Amount
- r = Rate of Interest (per annum)
- t = Time (in years)
- CI = compound Interest = A - P

* C++ Program :-

```
#include <iostream>
#include <cMath> // for pow() function
using namespace std;

int Main () {
    double principal, rate, time, amount,
    compoundInterest;
    // Input principal amount, rate,
and time
    cout << "Enter principal amount:";
    cin >> principal;
    cout << "Enter annual interest
rate (in %):";
    cin >> rate;
    cout << "Enter time (in years):";
    cin >> time;

    // compound Interest Formula
    amount = principal * pow((1 +
rate/100), time);
    compoundInterest = amount -
principal;
```

```
    // Display Results
    cout << "Final Amount (A):"
<< amount << endl;
    cout << "compound Interest (CI):
" << compoundInterest << endl;
    return 0;
}
```

Q.7) Write a C++ program to find the sum of individual digits of a positive integer.

→

* C++ Program :-

```
#include <iostream>
using namespace std;

int Main () {
    int num, sum = 0, digit;

    // Input number
    cout << "Enter a positive inte-
ger:";
    cin >> num;

    // Ensure the number is positive
    if (num < 0) {
        cout << "Please enter a
positive integer!" << endl;
        return 1; // Exit program
```

```
    // calculate sum of digits
    while (num > 0) {
        digit = num % 10 ;   // Extract
last digit
        sum += digit ;       // Add to sum
        num /= 10 ;          // Remove
last digit
    }

    // display result
    cout << "sum of digits : " << sum
<< endl;
    return 0;
}
```

Q.8) Write a C++ Program to generate first n terms of Fibonacci sequence.

→

F(0) = 0 , F(1) = 1

* C++ Program :-

```
# include <iostream>
using namespace std;

int Main () {
    int n, first = 0 , second = 1,
next;
    // Input number of terms
    cout << "Enter the number of
terms : " ;
    cin >> n ;
    // validate input
    if (n <= 0) {
        cout << " Please enter a
positive integer ! " << endl;
        return 1 ;
    }
    cout << " Fibonacci sequence : " ;

    // Generate Fibonacci series
    for (int i = 0 ; i < n ; i++) {
        cout << first << " " ;
    // Print the current term
        next = first + second ;
    // calculate the next term
        first = second ;  // update first
        second = next ;  // update second
    }

    cout << endl;
    return 0;
}
```

* Example output :-

Enter the number of terms : 7
Fibonacci sequence : 0 1 1 2 3 5 8

Q.9) Write a C++ Program to find both the largest and smallest number in a list of integers.

→

* C++ Program :-

```
# include <iostream>
 using namespace std;

int Main () {
    int n, num, Min, Max ;

    // Input number of elements
    cout << "Enter the number of
 elements : " ;
    Cin >> n;
    // Validate input
    if (n <= 0) {
        cout << " Please enter a valid
 positive number of elements ! " <<
 endl ;
        return 1 ;
    }

    // Input first element and
 initialize Min & Max
    cout << " Enter number 1 : " ;
    Cin >> num ;
    Min = Max = num ;
```

```
    // Read remaining elements
    for (int i = 2 ; i <= n ; i++) {
        cout << " Enter number " << i
 << " : " ;
        cin >> num ;
        if (num > Max ) Max = num;
    // update Max if needed
        if (num < Min ) Min = num ;
    // update Min if needed
    }
    // Display results
    cout << "Smallest number : " <<
 Min << endl ;
    cout << " Largest number : " <<
 Max << endl ;

    return 0 ;
}
```

Q.10) Write a C++ program to sort a list of numbers in ascending order.

→

* C++ Program :-

```
# include <iostream>
 using namespace std ;
void bubblesort (int arr[] , int n) {
```

```
    for (int i=0; i<n-1; i++){
        for (int j=0; j<n-i-1;
j++) {
            if (arr[j] > arr[j+1]){
                // swap elements if they
                are in wrong order
                swap (arr[j], arr
                                [j+1]);
            }
        }
    }
}

int Main () {
    int n;
    // Input number of elements
    cout << "Enter the number of
elements: ";
    cin >> n;
    int arr[n];
    // Input array elements
    cout << "Enter " << n << " num-
bers: ";
    for (int i=0; i<n; i++) {
        cin >> arr[i];
    }
    // Sorting the array
    bubbleSort (arr, n);
    // Display sorted array
    cout << "Sorted numbers
in ascending order: ";
    for (int i=0; i<n; i++){
        cout << arr[i] << " ";
    }
    cout << endl;
    return 0;
}
```

* Example output:-
Enter the number of elements : 5
Enter 5 numbers : 12 5 9 22 7
Sorted numbers in ascending order:
5 7 9 12 22.

Q.11) Write a Program Illustrating class Declarations, Definition, and Accessing class Members.

→

* Key concepts :-

1) Class Declaration :- Defines a class with attributes (variables) and Methods (functions).

2) Class Definition :- Member functions are defined inside or outside the class.

3) Accessing Members :- objects of the class access its attributes and Methods.

* C++ Program :-

```
# include <iostream>
using namespace std ;

// Class Declaration
class student {
private :
    string name ;
    int age ;
public :
    // Member Function to set Data
    void setData (string studentName,
int studentAge) {
        name = studentName ;
        age = studentAge ;
    }
    // Member Function to Display Data
    void displayData () {
        cout << "student Name : " <<
name << endl ;
        cout << "student Age : " <<
age << endl ;
    }
} ;

int Main () {
    // creating an object of student
class
    student s1 ;
    // setting values using setData()
    s1. setData ("Alice", 20) ;

    // Accessing class Members using
displayData ()
    cout << "student Details :" <<
endl ;
    s1. displayData () ;

    return 0 ;
}
```

* Example Output :-

```
student Details :
student Name : Alice
student Age : 20
```

Q.12) Write a C++ Program to illustrate default constructor, parameterized constructor and copy constructors.

→

* Explanation :-

1) Default constructor : Initializes values with default settings.

2) Parameterized constructor : Takes arguments to initialize object pr

3) Copy constructor : Creates a new object by copying values from an existing object.

* C++ Program :-

```
#include <iostream>
using namespace std;

class DeMo {
    int x, y;
public:
    // Default constructor
    DeMo () {
        x = 0;
        y = 0;
        cout << "Default constructor
called" << endl;
    }

    // Parameterized Constructor
    DeMo (int a, int b) {
        x = a;
        y = b;
        cout << " Parameterized
constructor called" << endl;
    }

    // Copy Constructor
    DeMo (const DeMo &obj) {
        x = obj.x;
        y = obj.y;
        cout << "copy constructor
called" << endl;
    }

    // function to display values
    void display () {
        cout << "x =" << x << ", y ="
<< y << endl;
    }
};

int Main() {
    DeMo obj1;  // Default Constructor
    obj1.display ();

    DeMo obj2 (10,20); // Parameterized
                          constructor

    obj2.display ();

    DeMo obj3 = obj2;
    // copy constructor
    obj3.display ();

    return 0;
}
```

* output :-

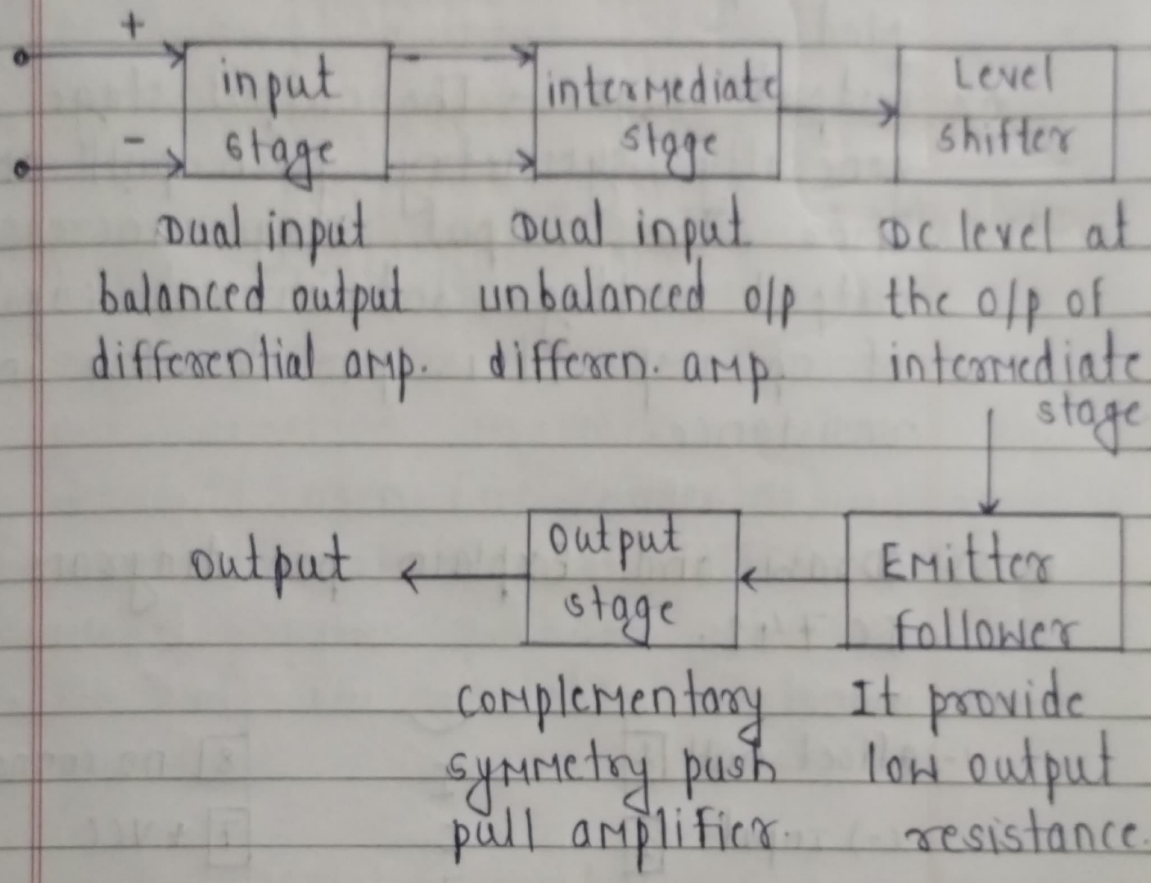
Default constructor called
x = 0, y = 0
Parameterized constructor called
x = 10, y = 20
copy constructor called
x = 10, y = 20

* This program helps illustrate the different types of constructors in C++.

★ Assignment - 1 ★

Q.1) Draw and Explain Block diagram of operational - amplifier (op - Amp).

→



1) Input stage :→ The input stage is dual input, balanced output differential amplifier. It has inputs non-inverting & inverting input.

2) Intermediate stage :→ Intermediate stage usually is another differential amplifier in single ended output. This stage is driven by output of the first stage.

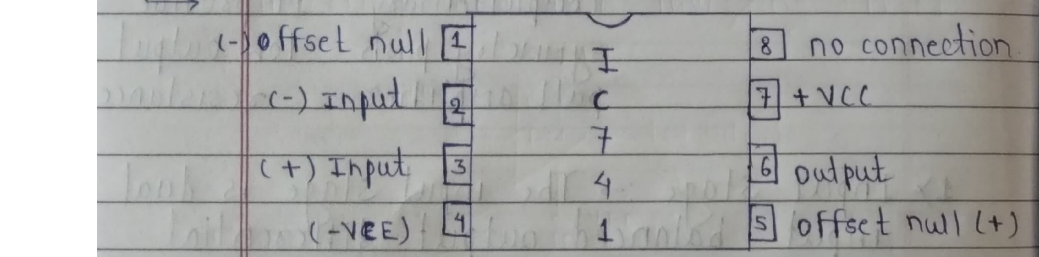
3) Emitter follower :→ The emitter follower stage provides low output resistance op-amp.

4) Level shifter :→ since large value capacitors cannot be fabricated on a chip, virtual ICs are direct coupled.

5) output stage :→ The output stage is generally symmetry push-pull amplifier, The output stage increases output voltage and current capability of op-amp. It is provides low input resistance.

Q.2) Draw and Explain pin diagram of IC 741.

→



* fig. pin diagram

★ Features of IC 741 :-

1) No external frequency compensation is required.
2) It has short-circuit protection.
3) It has offset-null capability.
4) Large common-mode and differential voltage ranges.
5) No latch up problems.
6) Power consumption is low.

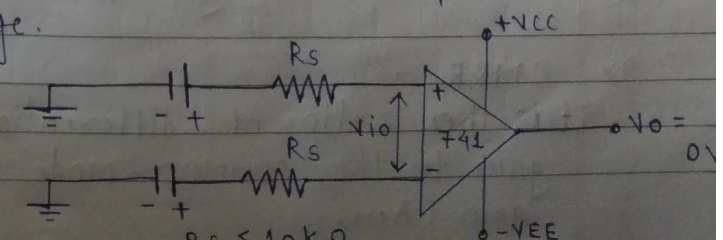
* This IC is an 8 pin IC & it is the dual in line package (DIP).
* Pin numbers 2 and 3 are inverting & non-inverting inputs respectively & we get the output voltage at pin number 6.
* Pin numbers 1 and 5 can be used to nullify the offset voltage and pin number 8 is a dummy pin which is not connected anywhere & hence should be left open (un-connected).
* Even though the Mini-DIP is the most popular package, the op-amp IC 741 is available in some other packages.

Q.3) Define the parameter of op-amp 741

→

1) Input offset voltage (Vio) :→

- Input offset voltage is the voltage that must be applied between two input terminals, balance the op-amp i.e. to null the output as shown in fig. Vio denotes the input offset voltage.



- The input offset voltage could be positive or negative.

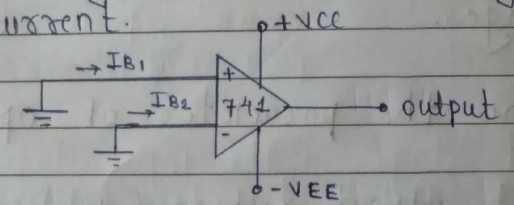
2) Input base current (I_B) :→

- Input base current is an average of currents into inverting and non-inverting input terminals of op-amp.

$$I_B = \frac{I_{B1} + I_{B2}}{2}$$

3) Input offset current (I_{io}) :→

- The difference between the current entering the two input terminals inverting and non-inverting offset current.



$I_{io} = I_{B1} - I_{B2}$, when $V_o = 0$

where, I_{B1} ⇒ current flowing into non-inverting input.

I_{B2} ⇒ current flowing into inverting input.

4) CMRR :→

- It is the ratio of differential voltage gain to the common-mode voltage gain (A_{CM}).

$$\therefore \text{CMRR} = \left|\frac{A_d}{A_{CM}}\right|$$

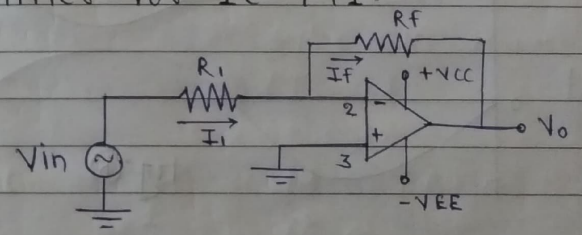
5) Slow Rate :→

- It is define as the ratio of the rate of change of output voltage w.r.t. time / the rate of change of time. (V/s → unit).

$$\therefore SR = \frac{dV_o}{dt}$$

Q.4) Derive the Derivation of inverting amplifier for IC 741.

→



* As seen in circuit diagram, AC input signal is connected to negative terminal of IC 741. That's why we called it as inverting amplifier.
* Current flowing from input power supply is given by $I_1 = I_f$.

$$\therefore I_1 = I_f$$

$$\frac{V_{in} - V_1}{R_1} = \frac{V_1 - V_o}{R_f}$$

ohm's law
V = IR
I = V/R

now, V.G.C → $V_1 = V_2 = 0$

$$\frac{V_{in} - 0}{R_1} = \frac{0 - V_o}{R_f}$$

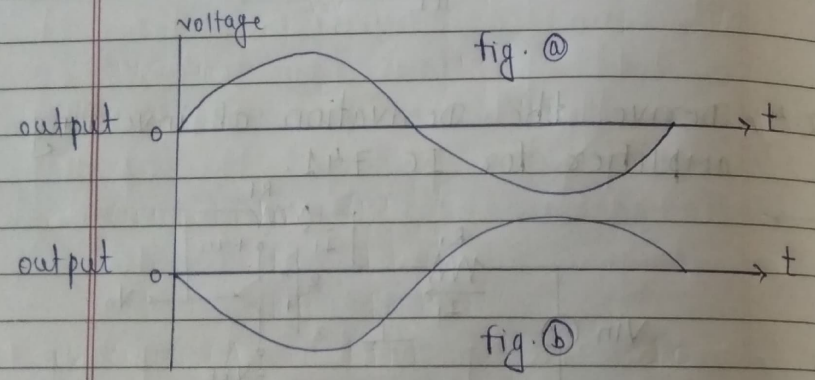
$$\therefore \frac{V_{in}}{R_1} = -\frac{V_o}{R_f}$$

$$\therefore \text{Gain } (A) = \frac{V_o}{V_{in}}$$

$$\frac{V_o}{V_{in}} = -\frac{R_f}{R_1}$$

$$\text{Gain } (A) = -\frac{R_f}{R_1}$$

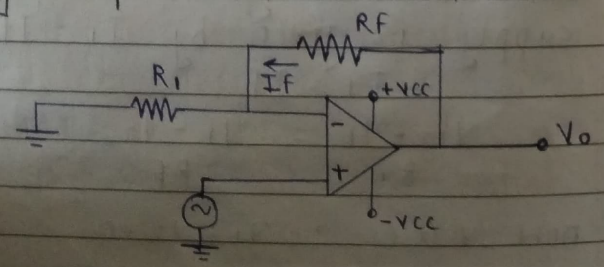
★ Waveform :→



* Negative sign indicate that output is 180° out of phase shift w.r.t input.

Q.5) Derive the Derivation of non-inverting amplifier for IC 741.

→



* As seen in circuit diagram input signal is connected to positive terminal of op-amp, that's why we called it as non-inverting amplifier.

* The current flowing from output to input is given by :

$$I_f = I_1$$

$$\frac{V_o - V_1}{R_f} = \frac{V_1 - V_G}{R_1}$$

$$V_1 = V_2 = V_{in} \rightarrow \text{V.G.C.}$$

$$\frac{V_o - V_{in}}{R_f} = \frac{V_{in} - 0}{R_1}$$

$$\therefore \frac{V_o}{R_f} - \frac{V_{in}}{R_f} = \frac{V_{in}}{R_1}$$

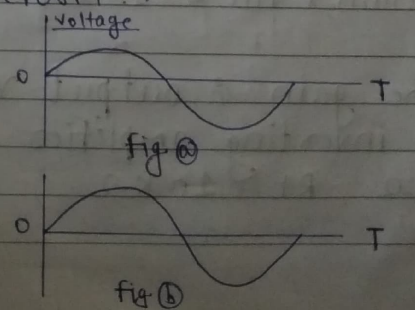
$$\therefore \frac{V_o}{R_f} = \frac{V_{in}}{R_1} + \frac{V_{in}}{R_f}$$

$$= V_{in}\left[\frac{1}{R_1} + \frac{1}{R_f}\right]$$

$$\therefore \frac{V_o}{V_{in}} = R_f\left[\frac{1}{R_1} + \frac{1}{R_f}\right] = \frac{R_f}{R_1} + \frac{R_f}{R_f}$$

$$\therefore \text{Gain} = \frac{V_o}{V_{in}} = \left[1 + \frac{R_f}{R_1}\right]$$

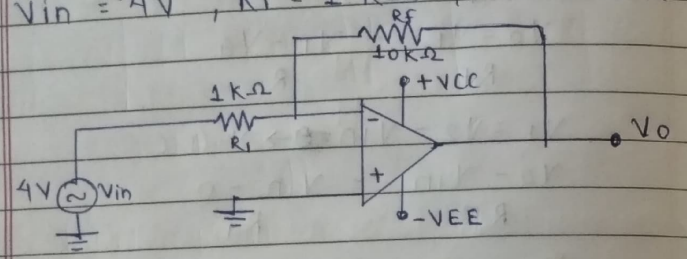
★ Waveform :→



* positive sign indicate that output are in same phase w.r.t. input.

Q.6) To find the gain and output voltage of inverting amplifier with input voltage.
$V_{in} = 4V$, $R_1 = 1\,k\Omega$, $R_F = 10\,k\Omega$

→



→ Given, $V_{in} = 4V$
$R_1 = 1\,k\Omega$
$R_f = 10\,k\Omega$

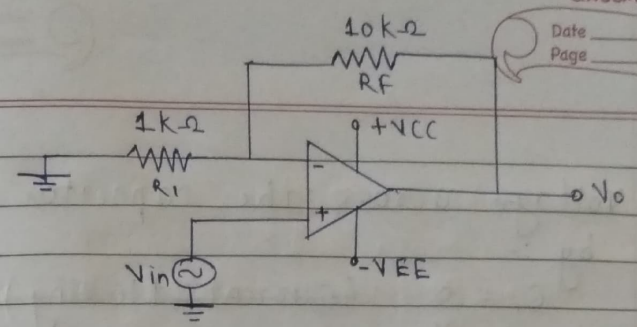
1) Gain $(A) = -\frac{R_F}{R_1} = -\frac{10}{1} = -10$

2) V_o ?
→ Gain $= \frac{V_o}{V_{in}}$

$V_o = \text{Gain} \times V_{in} = -10 \times 4 = -400\,V$

Q.7) find the gain & output voltage of non-inverting amplifier with $R_1 = 1\,k\Omega$, $R_F = 10\,k\Omega$.

→



→ Given :-
$R_1 = 1\,k\Omega$
$R_F = 10\,k\Omega$

To find :-
1) Gain
2) V_o

1) Gain $(A) = 1 + \frac{R_F}{R_1} = 1 + \frac{10}{1} = 11$

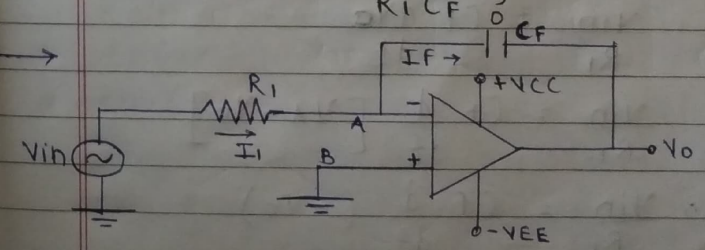
2) $V_o = ?$
→ Gain $= \frac{V_o}{V_{in}}$

$V_o = \text{Gain} \times V_{in} = 11 \times 4 = 44\,V$

Q.8) Derive the equation of integrator for IC 741 or prove that,

$$V_o = -\frac{1}{R_1 C_F}\int_0^t V_{in}\,dt$$

→



* As seen in circuit diagram, the feed-back resistance RF is normally replaced by capacitor CF to form inverting type integrator

* Voltage across the capacitor is given by :-

$$C = \frac{Q}{V}$$ ~~(current flowing)~~

* Current flowing through capacitor (IF) is given by :-

$$\left[I_F = C \frac{dV_C}{dt} \right] \quad \text{— ①}$$

$$\therefore I_F = \frac{dQ}{dt} \quad \text{— ②}$$

from eqn ① & ② :-

$$I_F = C_f \frac{dV_C}{dt} \quad \text{— ③}$$

∴ From given circuit diagram :→

$$I_1 = I_F$$

$$\frac{V_{in} - V_A}{R_1} = C_f \frac{dV_C}{dt}$$

$$V_A = V_B = 0$$

$$\frac{V_{in}}{R_1} = C_f \frac{dV_C}{dt}$$

$$\frac{V_{in}}{R_1} = C_f \frac{d}{dt}\left[V_A - V_o\right]$$

$$\therefore \frac{V_{in}}{R_1} = C_f \frac{d}{dt}(-V_o)$$

$$\therefore \frac{V_{in}}{R_1} = -C_f \frac{dV_o}{dt}$$

$$\therefore \frac{dV_o}{dt} = -\frac{V_{in}}{R_1 C_f}$$

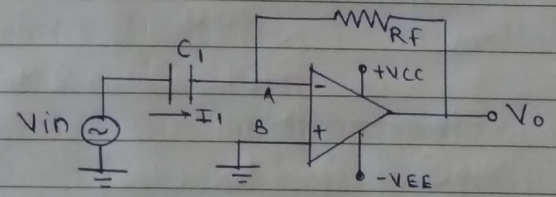
$$\therefore V_o = -\frac{1}{R_1 C_f} \int_0^t V_{in}\, dt$$

∴ Hence proved

Q.9) Derive the equation of differentiate for IC741 or prove that,

$$V_o = -R_F C_1 \frac{dV_{in}}{dt}$$

→



* As seen in circuit diagram R_1 is replaced by capacitor which is connected at negative terminal of op-amp. that's why it is called as inverting differentiation feedback.
* RF is connected with input to output.
* The current flowing from capacitor is given by :-

$$I_1 = I_F \quad \text{— ①}$$

$$I_1 = C_1 \frac{dV_C}{dt} = C_1 \frac{d}{dt}(V_{in} - V_A)$$

$$\therefore I_1 = C_1 \frac{d}{dt}(V_{in}) \qquad [\because V_A = 0]$$

* from eqn ① ⇒

$$I_1 = I_F$$

$$C_1 \frac{d}{dt}(V_{in}) = \frac{V_A - V_o}{R_F}$$

$$V_A = V_B = 0$$

$$C_1 \frac{d}{dt}(V_{in}) = \frac{-V_o}{R_F}$$

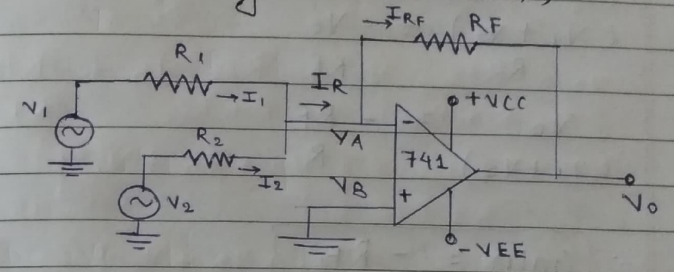
$$\left[\therefore V_o = -R_F C_1 \frac{d}{dt}(V_{in})\right]$$

∴ Hence proved.

★ Assignment No. 2 ★

Q.1) Design op-amp inverting amplifier for summing (+) (addition).

→



* As seen in circuit diagram of inverting amplifier for addition of two input signal, which is connected to negative terminal of 741 IC is called as Inverting suming amplifier.
* Now, Two input signal V_1 & V_2 are connected to negative terminal of IC 741.

∴ $V_1 + V_2$

∴ $I_R = I_1 + I_2$ ..... Addition of two current I_1 & I_2

∴ I_{RF} → feedback current.

now,

$$I_R = I_{RF}$$

$$\frac{V_1 - V_A}{R_1} + \frac{V_2 - V_A}{R_2} = \frac{V_A - V_o}{R_F}$$

∴ $V_A = V_B = 0$ [virtual Ground concept]

$$\therefore \frac{V_1}{R_1} + \frac{V_2}{R_2} = \frac{-V_o}{R_F}$$

∴ if $R_1 = R_2 = R$:→

$$\therefore \frac{V_1}{R} + \frac{V_2}{R} = \frac{-V_o}{R_F}$$

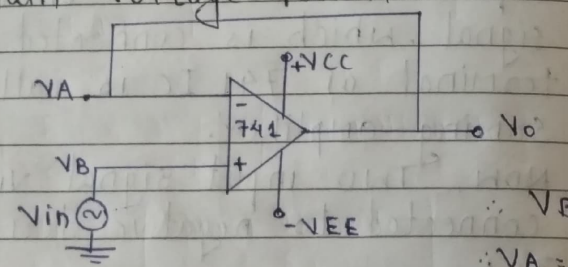
$$\frac{(V_1+V_2)}{R} = \frac{-V_o}{R_F}$$

$$\frac{V_o}{(V_1+V_2)} = -\frac{R_F}{R}$$

$$\therefore \boxed{\text{Gain} = -\frac{R_F}{R}}$$

Q.2) Explain voltage follower.

→



$\therefore V_B = V_{in}$

$\therefore V_A = V_B = V_{in}$

* Defn:-
  - It is define as the stability between output to input are always stable.
  - ∴ i.e. input must be equal to output.
    ∴ ($V_{in} = V_o$)

* As seen in circuit diagram input supply is connected to positive terminal of IC 741 is called as non-inverting amplifier.
* output voltage forming negative feedback which is connected with negative terminal of IC741 negative feedback, maintain the stability betn output to input.
* now,

$V_B = V_{in}$

$\therefore V_A = V_B = V_{in}$ [V.G.C.]

* i.e. negative feedback maintain the stability between output verses input,

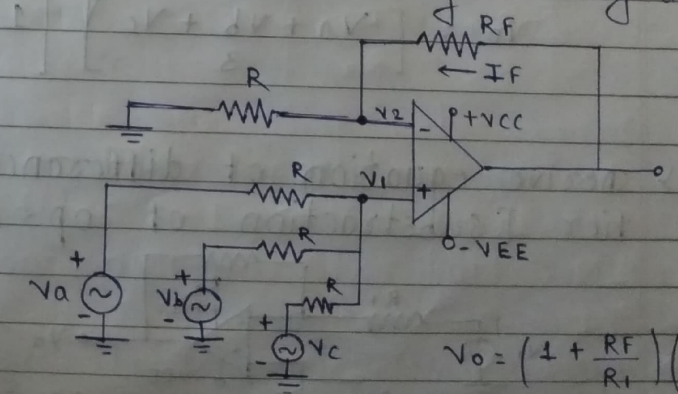
$\therefore V_{out} = V_{input}$.

$$\therefore \boxed{V_o = V_{in}}$$

Q.3) Design op-AMP for non-inverting summing amplifier.

→

* As seen in circuit diagram input signal V_a, V_b, V_c are connected to positive terminal of IC 741 that's why we called as non-inverting summing amplifier



$$V_o = \left(1 + \frac{R_F}{R_1}\right)\left(\frac{V_a+V_b+V_c}{3}\right)$$

$\therefore V_1 = V_2 \Rightarrow$ (virtual ground concept)

$\therefore$ input resistance is very high.

$$\therefore V_1 = \frac{R/2}{R+R/2} V_a + \frac{R/2}{R+R/2} V_b + \frac{R/2}{R+R/2} V_c$$

$$\therefore V_1 = \frac{R/2}{R+R/2}\left[V_a + V_b + V_c\right]$$

$$= \frac{R/2}{\frac{2R+R}{2}}\left[V_a + V_b + V_c\right]$$

$$= \frac{R/2}{\frac{2R+R}{2}}\left[V_a + V_b + V_c\right]$$

$$= \frac{R}{2} \times \frac{2}{3R}\left[V_a + V_b + V_c\right]$$

$$\therefore V_2 = V_1 = \left[\frac{V_a + V_b + V_c}{3}\right]$$

$\therefore$ According to KVL,

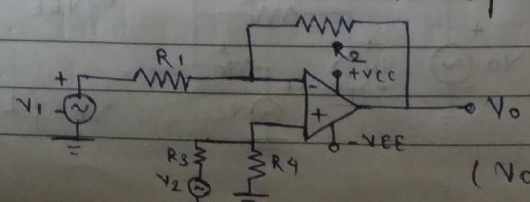
$$I_1 = I_F$$

$$V_o = V_2$$

$$\therefore V_o = \left[\frac{V_a + V_b + V_c}{3}\right]\left[1 + \frac{R_F}{R_1}\right]$$

Q.4) Derive equation of difference amplifier [substraction of op-Amp].

→



$(V_o^2 - V_o')$

* As seen in circuit diagram in difference amplifier to input signal V_1 is connected to negative terminal of IC 741, we called as inverting amplifier and input voltage V_2 is directly connected with positive terminal of IC 741, we called it as non-inverting amplifier.
* The difference between two input voltage must be equal to zero.
* Now, consider the output voltage of V_1 is given by,

$$V_oV_1 = -V_1 \frac{R_2}{R_1} \quad \text{—— (1)}$$

* Now, outpute of second input terminal V_2 is given by,

$$V_oV_2 = V_2\left(1 + \frac{R_2}{R_1}\right)\left[\frac{R_4}{R_3 + R_4}\right] \quad \text{—— (2)}$$

* Now, By taking difference output of signal V_2 and signal V_1 :-

$$V_{out} = V_oV_2 - V_oV_1$$

$$\therefore V_{out} = V_2\left(1 + \frac{R_2}{R_1}\right)\left[\frac{R_4}{R_3 + R_4}\right] - V_1\left(\frac{R_2}{R_1}\right)$$

* Now, The difference of two input signal must be equal to zero,

$$\therefore V_2 - V_1 = 0$$

$$\therefore 0 = \left(\frac{R_4}{R_3+R_4}\right)\left(1+\frac{R_2}{R_1}\right) - \left(\frac{R_2}{R_1}\right)$$

$$\left[\frac{R_4}{R_3+R_4}\right]\left[1+\frac{R_2}{R_1}\right] = \left(\frac{R_2}{R_1}\right)$$

$$\therefore \left[\frac{R_4}{R_3}+\frac{R_4}{R_4}\right] = \left(\frac{(R_2/R_1)}{(1+R_2/R_1)}\right)$$

$$\therefore \left[\frac{R_4}{R_3}+1\right] = \frac{(R_2/R_1)}{\frac{R_1+R_2}{R_1}}$$

$$\therefore \left[\frac{R_4}{R_3}+1\right] = \frac{R_2}{R_1+R_2}$$

$$\left[\frac{R_4}{R_3}+1\right] = \frac{R_2}{R_1}+\frac{R_2}{R_2}$$

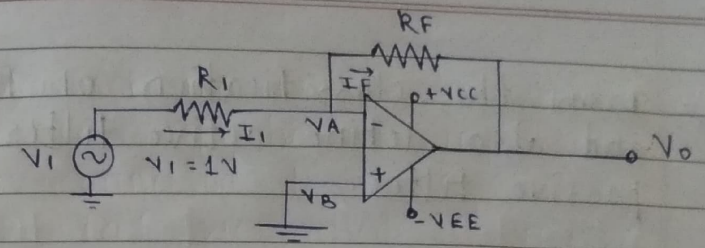
$$\therefore \left[\frac{R_4}{R_3}+1\right] = \frac{R_2}{R_1}+1$$

$$\therefore \frac{R_4}{R_3} = \frac{R_2}{R_1}+1-1$$

$$\therefore \boxed{\frac{R_4}{R_3} = \frac{R_2}{R_1}}$$

Q.5) As an inverting amplifier, let $R_F = 100\,k\Omega$, $R_1 = 10\,k\Omega$ then $V_1 = 1V$. find out the gain and current I_1.

→



→ given :-

$R_F = 100\,k\Omega$

$R_1 = 10\,k\Omega$

$V_1 = 1V$

To find :-

1) V_o 2) I_1

3) A (Gain)

$\therefore V_B = 0$, $V_A = 0$ ..... (V.G.C.)

$\therefore V_A = V_B = 0$

i) Gain :→

→ $\therefore$ Gain $= \frac{-R_F}{R_1} = \frac{-100}{10} = -10 = -10$

ii) V_o ? ⇒

→ Gain $= \frac{V_o}{V_{in}}$

$10 = \frac{V_o}{1}$

$V_o = 10$ volt

iii) I_1 = ? ⇒

→ $I_1 = \frac{V_A - V_o}{R_F}$

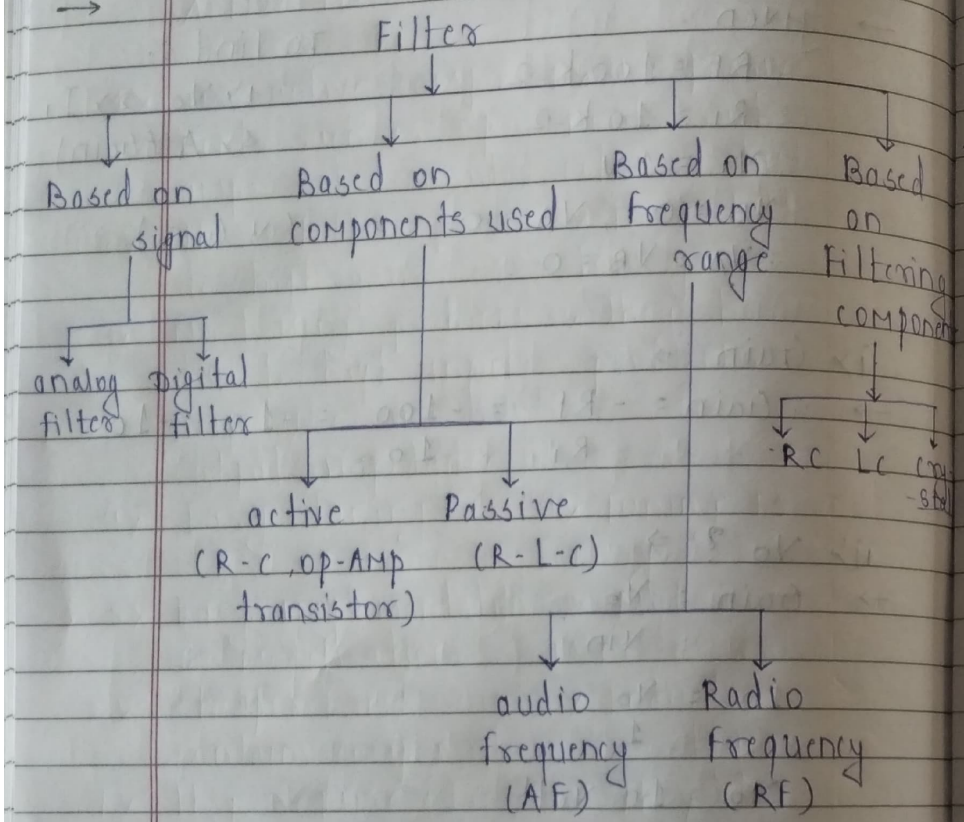
$= \frac{-V_o}{R_F} = \frac{-10}{100} \times 10^{-3}$

$\therefore I_1 = -0.1$ mA

★ Assignment - 3 ★

Q.1) Draw the classification of filter and also define active filter and passive filter.

→



Q.2) Define active filter and passive filter with advantages & disadvantages of active filter.

→

★ Active filter :- The filter is called as active filter because it provide the gain and do not autonate input signal and also it use resistance, capacitance, transistor and op-amp as a active filter component.

★ Passive filter :- In passive component it does not provide the gain with input signal in the circuit and inductor as a passive filter component.

★ Advantages of active filter :-

1) Active filter use active components like op-amp or transistor and provide gain to the input signal.

2) These filters require less space as they eliminate the need of large, heavy and costly inductors.

3) They are easy to tune.

4) Active filters can be turned independently due to excellent isolation property.

5) Low cost and small in size.

6) These are easily compatible with IC's due to absence of inductors.

★ Disadvantages of active filters :-

1) Due to stability problems, active filters are limited to frequencies below 100 KHz.

2) The require power for operating the active elements.

3) selectivity and center frequency are function of gain of the op-amp.

Q.3) Difference between Active and Passive filter.

Active filter	Passive filter
1) use active components such as op-amps and transistor	1) Use passive components such as capacitor, inductor etc.
2) These filters provide high gain.	2) These filters have low gain.
3) No loading problem due to high input impedance and low output impedance	3) They has loading problems.
4) Easy tuning.	4) Tuning is not easy.
5) Easy implementation & Modification.	5) Not easy to implement & Modify.
6) cheaper and contains components.	6) Inductors are costly.
7) They require supply.	7) They do not require supply.
8) Frequency is limited to op-amp bandwidth.	8) No limitations are on frequency range.

Q.4) Define :-

1) cut off frequency :-

- The frequency for which the gain drops below 3dB of the highest value is called cut off frequency.
- At this frequency, the power falls below 50%, hence called half power point. This frequency is called as 'break frequency'.

2) Bandwidth (BW) :-

It is the range of frequencies over which the output power of the filter is above 50% for this filter.

Bandwidth = Pass band.

3) Q-factor :-

Quality factor (Q-factor) is the figure of merit of a filter.

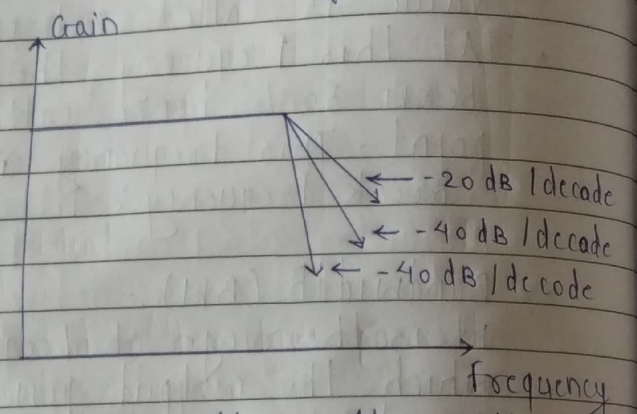
$$Q = \frac{\text{centre frequency}}{\text{Bandwidth}} = \frac{f_c}{B.W.}$$

- high quality filter gives low bandwidth i.e. low frequency response but good roll-off rate which improves the order of the filter.

4) Roll-off rate :-

In the stop band the filter response falls rapidly, the rate at which it

falls off is called roll-off rate.
Roll-off rate is the rate off change of gain with frequency.



for good (ideal) filter, the roll off rate should be ∞ dB/decade.

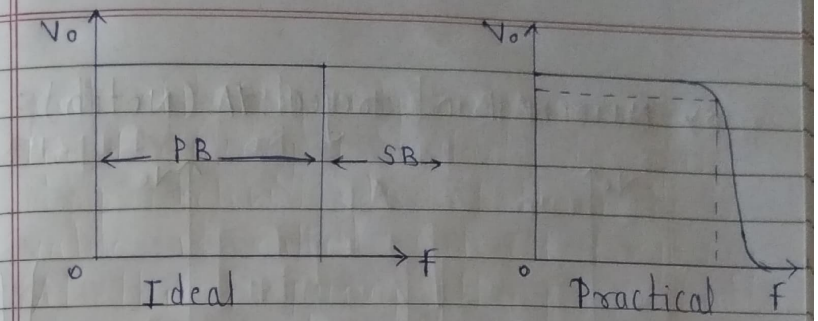
5) center frequency (f_c) :-
It is the Mid Frequency of the pass band or stop band of the filter.
center frequency is given by :

$$f_c = \sqrt{f_H - f_L}$$

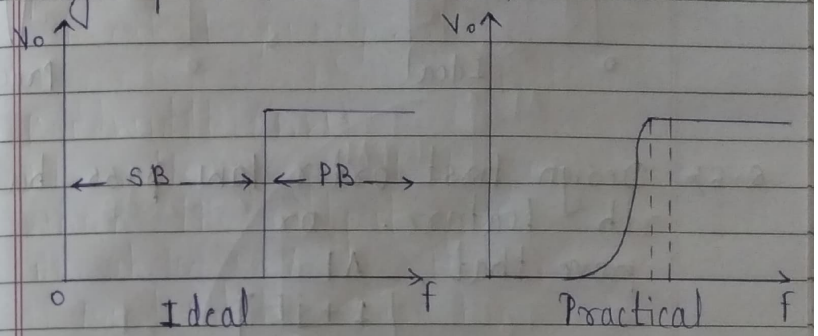
Q.5) Draw the ideal and practicle response for all filters.

→

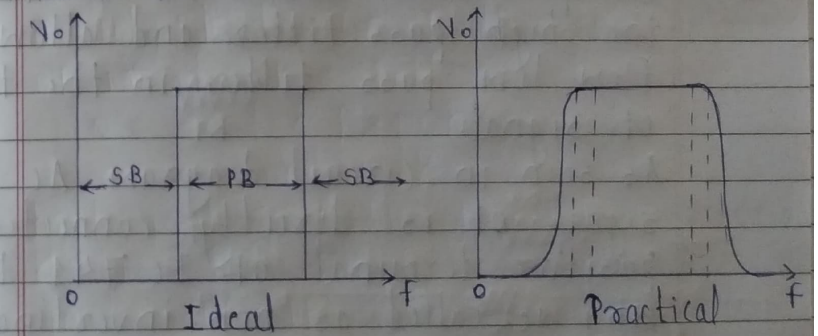
1) low Pass filter (LPF) :-



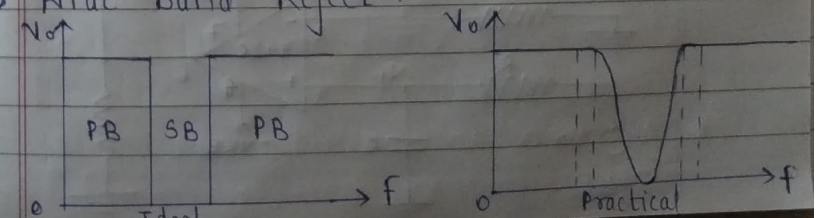
2) High pass filter (HPF) :-



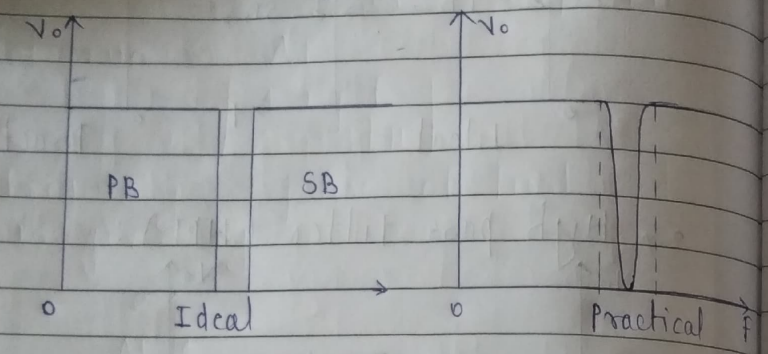
3) Band Pass filter (BPF) :-



4) Wide Band Reject :-

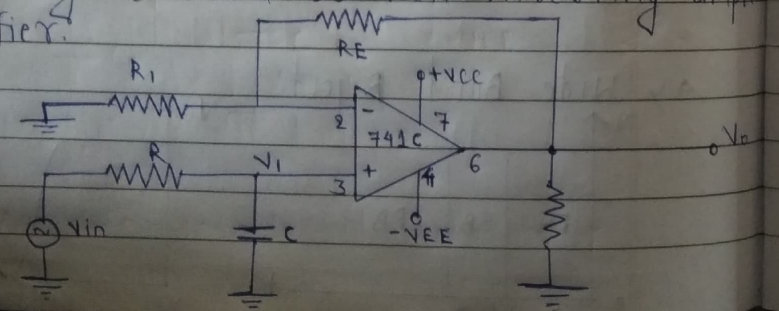


Narrow Band reject # (Notch) :-



Q.6) Design first order low pass butterworth Feature. or
prove that, $\frac{AF}{\left[1 + j\frac{F}{F_H}\right]}$

→ low pass filter define as it atonate the high pass filter and allow to flow low pass filter in the form of signal.
As seen in circuit diagram Ac input signal of is directly connected in positive terminal of IC 741 that's why we called as non-inverting amplifier.



* It consist of negative feedback with feedback resistance RF.
* In first order low pass filter consist of combination of R & C for 20 db.

$$j^2 = 1$$
$$j = \sqrt{1}$$

$$-jxc = \frac{1}{j2\pi fc}$$

$$V_1 = V_{in}\left[\frac{-jxc}{R - jxc}\right]$$

$$= \frac{1/j2\pi fc}{R + 1/j2\pi fc}$$

$$= \frac{1/j2\pi fc}{\frac{j2\pi fcR + 1}{j2\pi fc}}$$

$$V = \frac{1}{1 + jR2\pi fc} V_{in}$$

* The output voltage of non-inverting amplifier is given by,

$$V_o = \left(1 + \frac{R_f}{R_1}\right) V_1$$

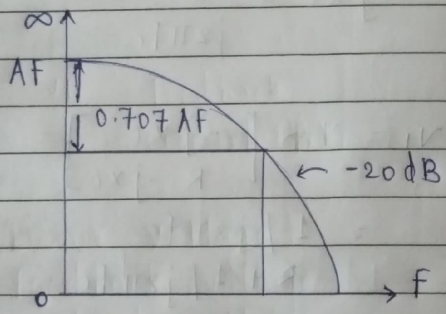
$$= \left(1 + \frac{R_f}{R_1}\right)\left(\frac{V_{in}}{1 + 2R2\pi fc}\right)$$

$$\frac{V_o}{V_{in}} = \left(1 + \frac{R_f}{R_1}\right)\left(\frac{1}{1 + j2\pi RFc}\right)$$

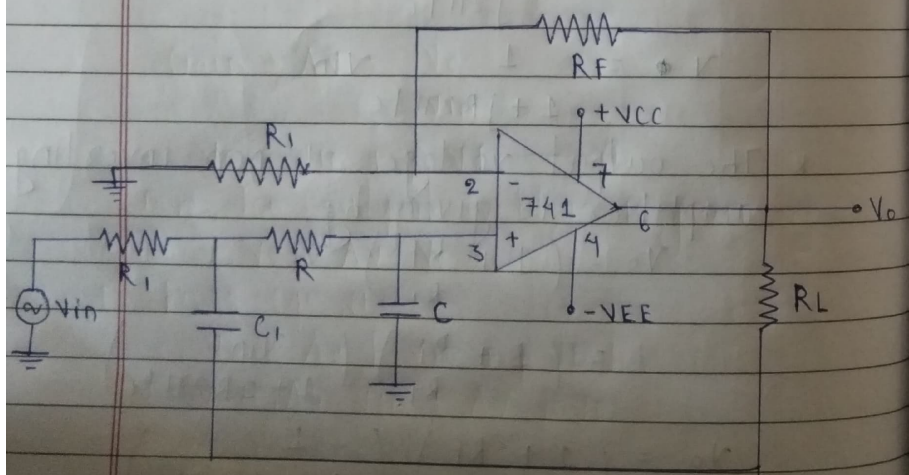
$$\text{Gain } AF = \left(1 + \frac{RF}{R_1}\right)\left(\frac{1}{1 + j2\pi RFc}\right)$$

$$\text{Gain} = \frac{AF}{\left[1 + j\frac{F}{FH}\right]}$$

* Frequency response of first order is given by :



Q.7) Design second order low pass butter-worth filter (-40dB, roll of rate).



* As seen in circuit diagram positive input signal is directly connected with positive terminal of IC 741 that's why it called as non-inverting low pass butterworth filter. In this circuit it having a combination of 2R and C it called as second order low passfilter with -40dB roll of rate.

1) $$FH = \frac{1}{2\pi\sqrt{R_1 R_2 C_1 C_2}}$$

$R_1 = R_2 = R$

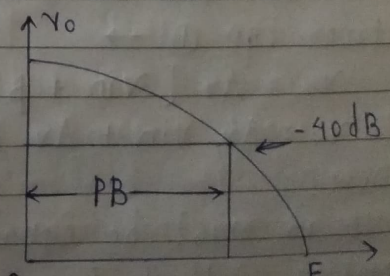
$C_1 = C_2 = C$

$$FH = \frac{1}{2\pi\sqrt{RC}}$$

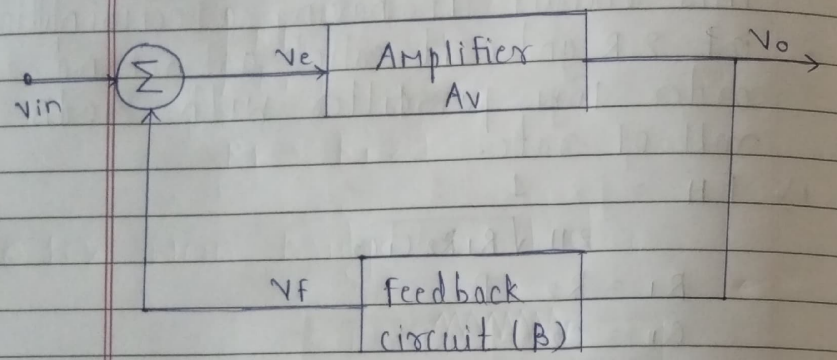
2) $$AvF = \left|\frac{V_o}{V_{in}}\right| = \frac{AF}{\sqrt{1 + \left(\frac{F}{FH}\right)^4}}$$

$$AvF = \frac{\left[1 + \frac{RF}{R_1}\right]}{\sqrt{1 + \left(\frac{F}{FH}\right)^4}}$$

frequency response of second order given by :



Q.8) Draw the block diagram of position feedback for oscillator and also eqn of oscillator.



V_e = Error voltage
V_F = feedback voltage
V_o = output voltage
A_v = Forward Gain
β = Feedback Factor.

$$\text{Gain} \Rightarrow A_V = \frac{V_o}{V_{in}}$$

$$V_e = V_{in} + \beta V_f \quad \text{—— (1)}$$

now, for output voltage is given by,

$$V_o = A_V V_e$$

from eqn (1) →

$$V_o = A_V [V_{in} + \beta V_E]$$

$$V_o = A_V V_{in} + A_V \beta V_E$$

now,

V_F is Must be equal to V_o We are Making feedback from output to input.

eqn (2) →

$$V_o = A_V V_{in} + A_V \beta V_o$$

$$V_o = A_V \beta V_o = A_V V_{in}$$

$$V_o (1 - A_V \beta) = A_V V_{in}$$

$$\frac{V_o}{V_{in}} = \frac{A_V}{(1 - A_V \beta)}$$

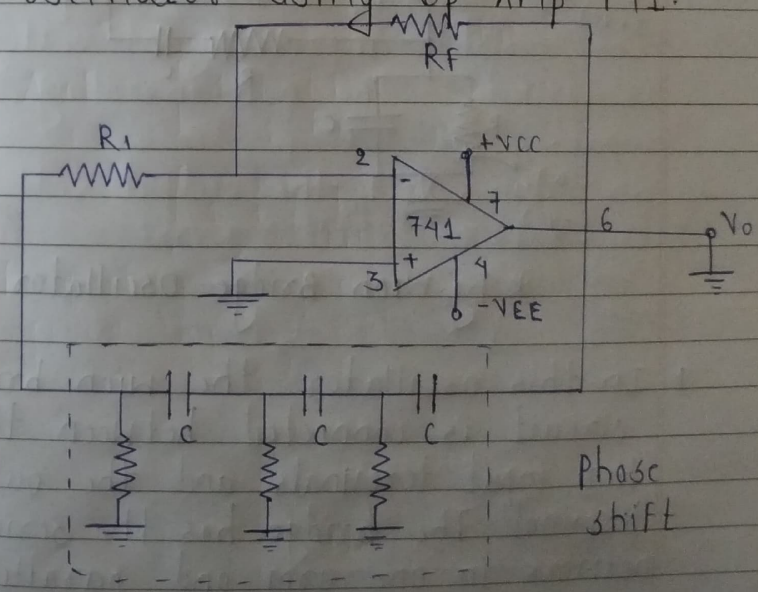
IF, $V_{in} = 0$

$$0 = 1 - A_V \beta$$

$$\therefore \boxed{A_V \beta = 1}$$

IF $A_V \beta$ is less than 1 then the polar value varies from 0 to 2π [0 to 360°].

Q.9) Draw and explain phase-shift oscillator using op-AMP 741.



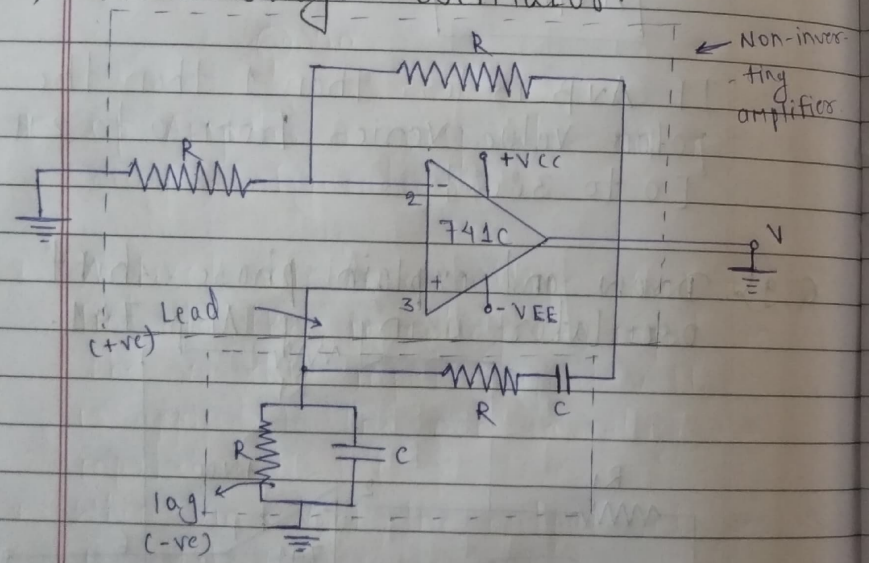
$$f_0 = \frac{1}{2\pi\sqrt{6RC}} = \frac{0.065}{RC}$$

$\therefore A = 29$

$$\left|\frac{R_F}{R_1}\right| = 29$$

$$R_F = R_1 29$$

Q.10) Wein Bridge oscillator.



* fig. Wien Bridge Oscillator *

* In this oscillator, the wien bridge circuit is connected between the amplifier input terminals and the output terminal. The bridge has the series RC network in one arm and parallel RC network in the adjoining arm. In this remaining two arms of the bridge, resistor R_1 and R_2 are connected.
* The phase angle criteria for sustain oscillation is that the total phase shift around the loop must be equal to 0°.
* The series network provides leading (+ve) phase shift hence called as lead network while the parallel network provides lagging (-ve) phase shift & therefore called as lag network. Lead & lag networks compensate phase shift at resonance and total phase shift generated by Wien Bridge will be 0°.
* The frequency of oscillation f_0 is exactly the resonant frequency of the balance wien bridge i.e.,

$$f_0 = \frac{1}{2\pi RC} = \frac{0.159}{RC}$$

The gain required for sustained oscillation is :

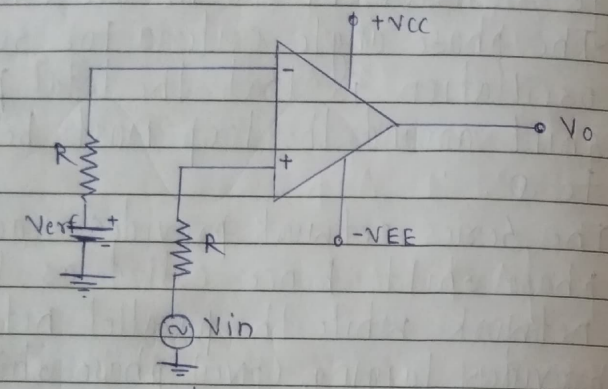
$$A_V = \frac{1}{\beta} = 3$$

$$1 + \frac{R_F}{R_1} = 3$$

$$\therefore \boxed{R_F = 2R_1}$$

★ Assignment - 4 ★

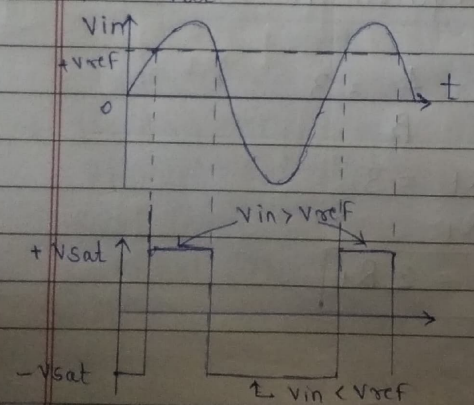
Q.1) Explain comparator from non-inverting amplifier.

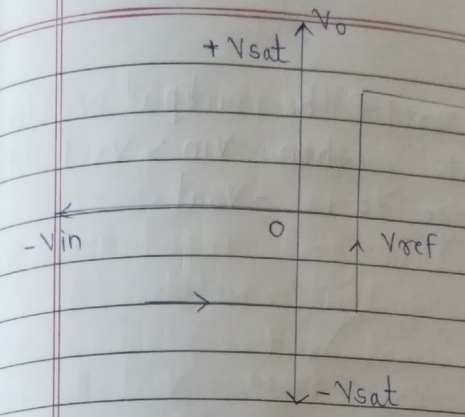


* Op-Amp used as a comparator in non-inverting configuration.

* A fixed reference voltage Vref is applied to the inverting (-) input and the variable input signal voltage Vin is applied to the non-inverting (+) input.

* When Vin < Vref the output voltage Vo is -Vat.



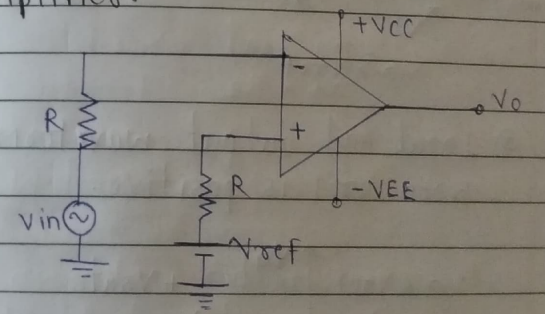


* Transfer characteristic of non-inverting comparator when Vref is positive *

* Thus Vo changes from one saturation level to another saturation level.

Q.2) Explain comparator from inverting amplifier.

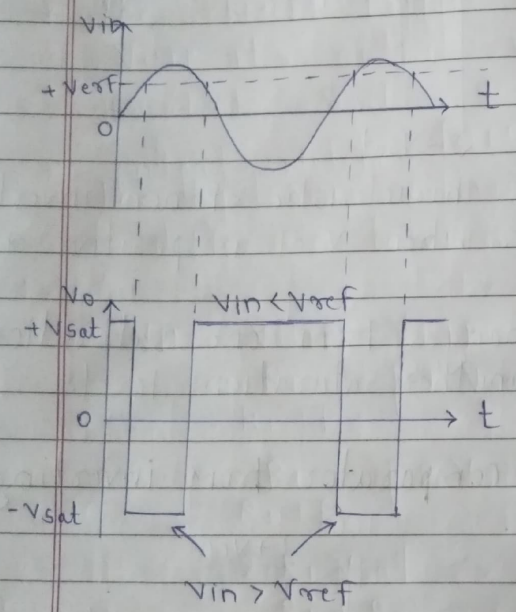
→



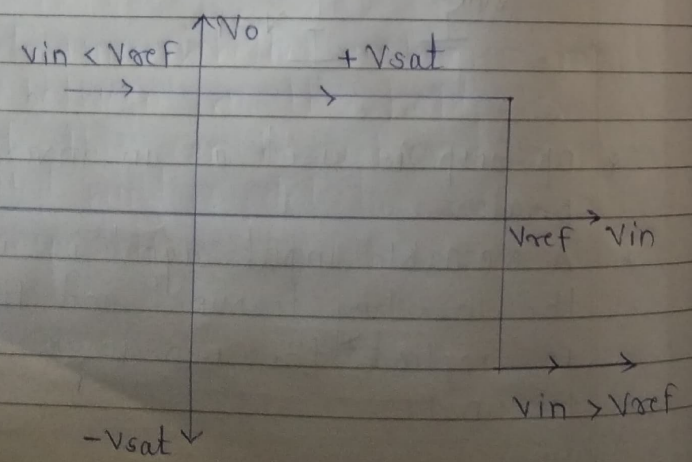
* Op-amp is used as an inverting comparators.

* A variable input signal is applied to the inverting terminal and fixed voltage is applied to the non-inverting input terminal.

* When $V_{in} < V_{ref}$, the output voltage V_o is $+V_{sat}$, when $V_{in} > V_{ref}$, the output voltage V_o is $-V_{sat}$.



ⓐ Input and output waveforms when Vref is positive.

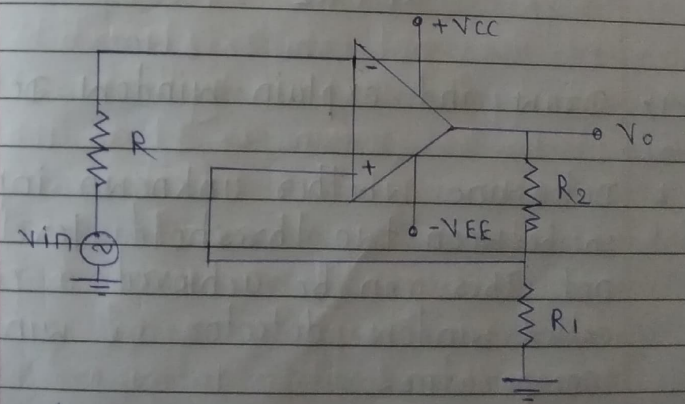


ⓑ Transfer characteristic when Vref is positive.

Q.3) Draw and Explain Schmitt Trigger. (Regenerative comparator).

→

- An inverting comparator with positive feedback, the circuit is known as schemitt trigger or squaring circuit.
- This circuit converts input waveform of any shape to a square waveform.
- The Input voltage vin is applied at inverting input terminal and feedback voltage to non-inverting terminal.
- The voltage across R_1 is feedback to the non-inverting input.



- When, $V_o = +V_{sat}$, the voltage across R_1 is UTP.

$$V_{UTP} = \frac{R_1}{R_1 + R_2}(+V_{sat})$$

$$\therefore V_{LTP} = \frac{R_1}{R_1 + R_2}(-V_{sat})$$

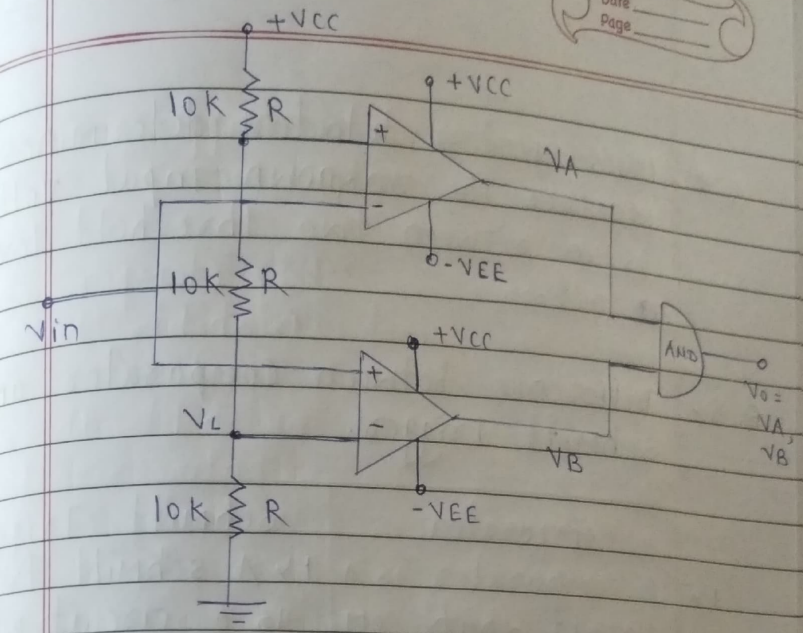
* The schmitt trigger input exceeds V_{UTP} it's output switches from $+V_{sat}$ to $-V_{sat}$.
* output switches from $-V_{sat}$ to $+V_{sat}$. The hysterisis voltage of course equal to difference between V_{UTP} and V_{LTP}.

$$V_H = V_{UTP} - V_{LTP}$$
$$= \frac{R_1}{R_1 + R_2}[+V_{sat} - (-V_{sat})]$$
$$= \frac{R_1}{R_1 + R_2}[2V_{sat}]$$
$$= \frac{2R_1 V_{sat}}{R_1 + R_2}$$

Q.4) Draw and explain window Detector.
→

* Determine whether unknown input is in between two threshold levels or not. This can be achieved a circuit called window detector or window comparator.

* Diagram :-



* Input signal is compared with two reference levels V_L and V_H by two comparators. The output of two comparators is fed to AND gate.

1) When $V_{in} \le V_L \le V_H$, V_o is at logic '0' (low). This is because comparators A output will be at logic '1' but output of comparator B will be at logic zero.

2) When $V_{in} \ge V_H \ge V_L$, V_o is at logic '0' (low), Because output of comparator B is at logic '1' and that of comparator A is at logic 0.

3) When $V_L \le V_{in} \le V_H$, V_o is at logic '1' because output of both the comparators with be at logic '1'.

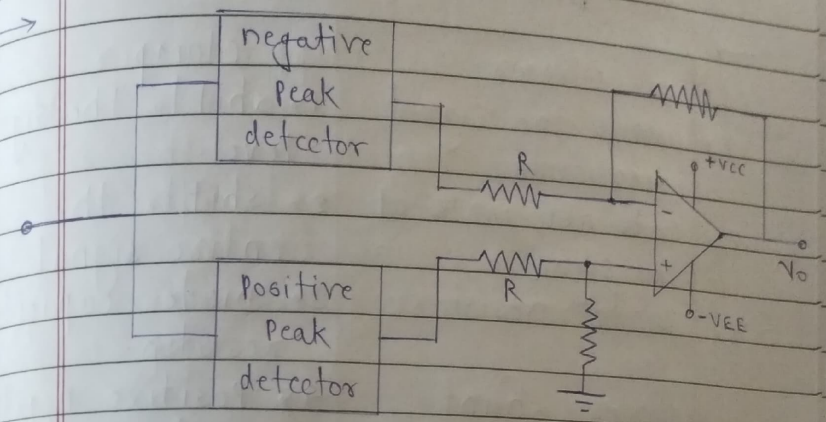
* Thus window detector determines whether the unknown input voltage is in between two threshold levels or not.

Q.5) Difference between Comparator and Schmitt Trigger.

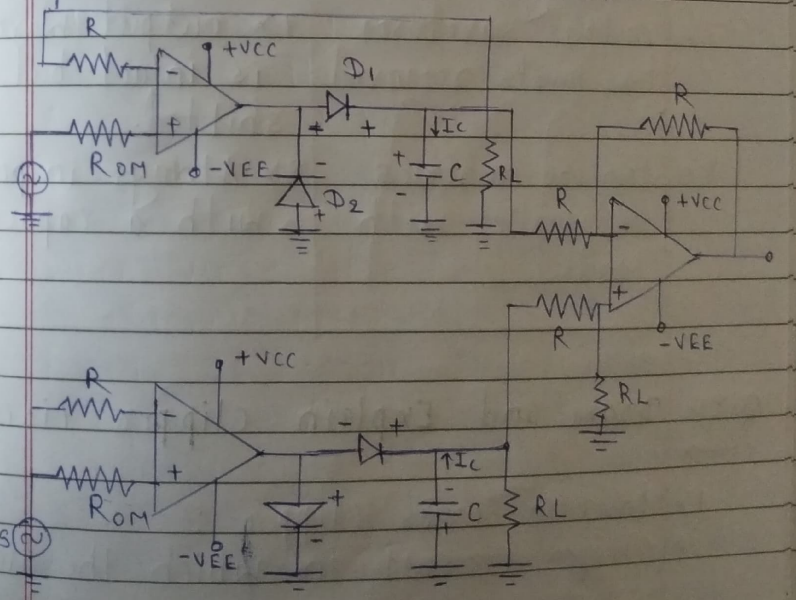
Comparator	Schmitt Trigger
1) A comparator is a circuit which compares input voltage with a single known reference voltage.	1) A schmitt trigger is a circuit which has two stable states and can be triggered with respect to two reference voltage.
2) There is no feedback.	2) It uses positive feedback.
3) It does not exhibit hysterisis.	3) It exhibits hysterisis.
4) It is less immune to noise.	4) It is immune to noise.
5) It is used as a zero-crossing detectors, sense amplifier, voltage level detector, etc.	5) It is used as a wave shaping circuit, sine wave to square wave convertor etc.
6) It requires external reference voltage to be applied.	6) It does not require external reference voltage as it is generated by saturated output feedback.

Q.6) Peak to Peak Detector. Draw and Explain.

→



* Peak to peak detector can be implemented by connecting the output of a positive peak detector and a negative peak detector to a subtractor.



Q.7) Difference between clipper and clamper.

→

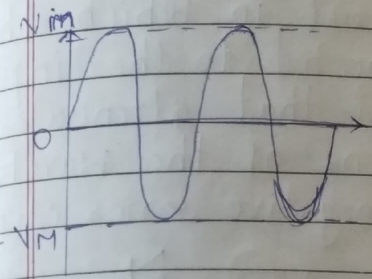
	clipper	clamper
1)	Limits the signal amplitude.	1) shifts the signal level.
2)	cuts off parts of the signal.	2) shifts the entire signal.
3)	Diode, resistor	3) Diode, capacitor, resistor.
4)	series, shunt, positive, negative.	4) Positive, negative, Biased.
5)	Does not change DC level.	5) Introduces a DC level shift.
6)	used for waveforms shaping.	6) used for signal restoration.
7)	output waveform has parts removed.	7) output waveform is intact but shifted.
8)	can be in series as parallel with load.	8) Always in parallel with a capacitor.

Q.8) Draw and Explain clipper circuit.

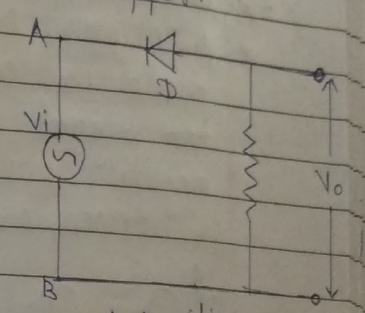
→

1) Positive Series clipper :-

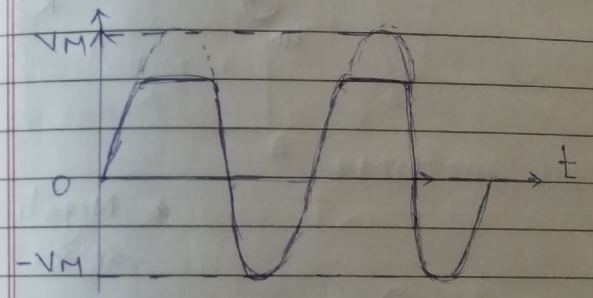
* clipper circuit in which the diode is connected in series to the input signal and that attenuates the positive portion of the waveform is termed as positive series clipper.



* Input waveform *



* positive series clipper *



* practical output waveform *

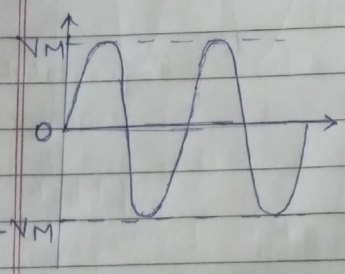
* Positive cycle of the Input :- The input voltage is applied, The voltage across the load resistor becomes zero as no current flows through it and hence VoVo will zero.

* Negative cycle of the Input :- negative cycle input makes the point A in the

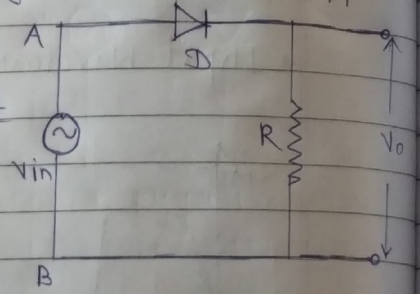
circuit negative with respect to the point. B.

2) Negative series clipper :-

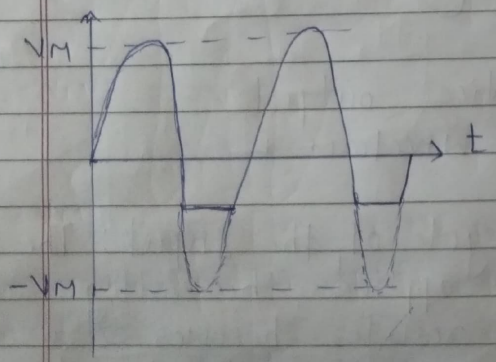
* A clipper circuit in which the diode is connected to the input ~~ten~~ signal and that attenuates the negative portions of the waveform, is termed as negative series clipper.



* Input waveform *



* Negative series clipper *



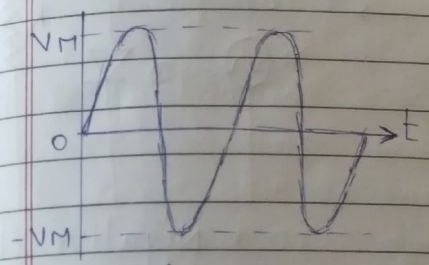
* Practical output waveform *

Q.9) Draw and Explain clamper circuit.

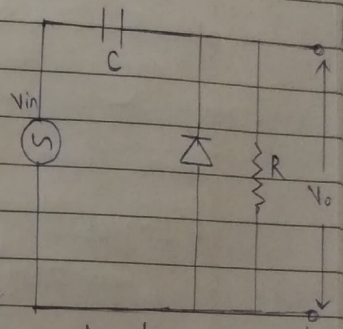
→

1) Positive clamper circuit :-

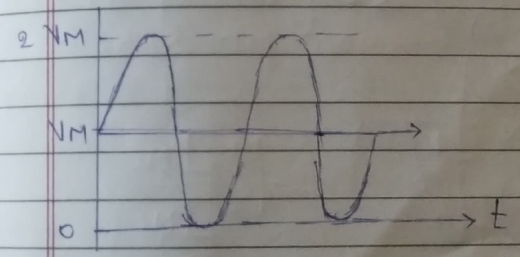
* A clamping circuit restores the DC level. When a negative peak of the signal is raised above to the zero level, then the signal is said to be positively clamped.



* Input waveform *



* Positive clamped circuit *



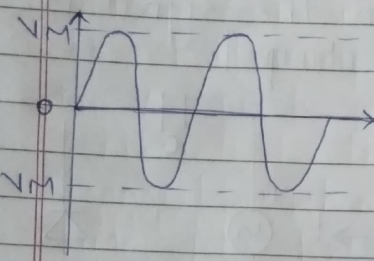
* output waveform *

* The output of the circuit at this moment will be :-

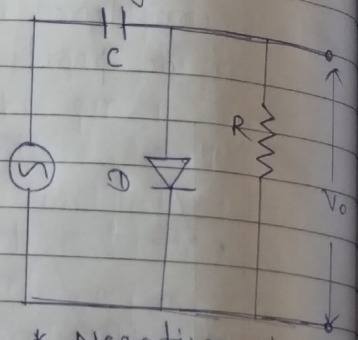
$$V_o = V_i + V_M V_o$$
$$= V_i + V_M$$

2) Negative clamper :-

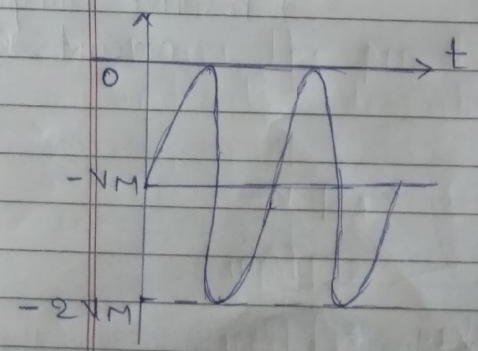
* A negative clamper circuit is one that consist of a diode, a resistor and a capacitor and that shifts the output signal to the negative portion of the input signal.



* Input waveform *



* Negative clamper circuit *



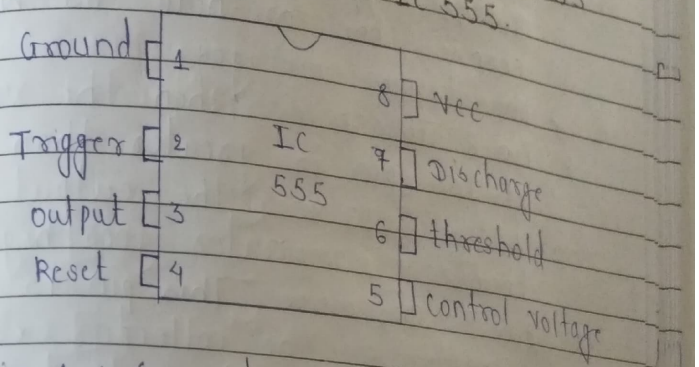
* output waveform *

- The output of the circuit at this moment will be :-

$$V_o = V_i + V_M V_o$$
$$= V_i + V_M$$

* Assignment - 5 *

Q.1) Draw the pin diagram of IC 555 and describe pin of IC 555.

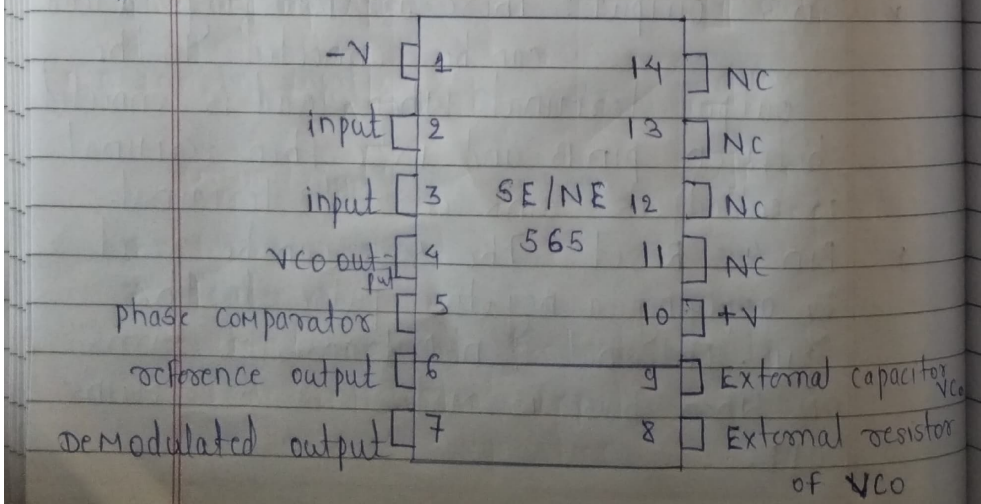


* Pin 1 :- Ground :→ All voltages are measured with respect to this terminal.
* Pin 2 :- Trigger :→ When this terminal voltage falls below the 1/3 Vcc the trigger comparator gives high output and that sets the output high.
* Pin 3 :- output :- There are two ways a load can be connected to the output terminal. When load is connected between pin 3 and +Vcc then terminal is called as 'sink current' terminal.
* Pin 4 :- Reset :- The 555 can be reset by applying a negative pulse to this pin. Internal P-n-P transistor turns ON when base is made low & remains OFF.

* Pin 5 :- control voltage :- An external voltage applied to this terminal changes the threshold as well as the trigger voltage.
* Pin 6 :- Threshold :- When the voltage at this pin is greater than threshold voltage (2/3 Vcc) the output of the threshold comparator goes high.
* Pin 7 :- Discharge :- This pin is internally connected to the collector of the transistor. When the output of the time goes low, internal transistor turns ON and it acts as a short circuit, when connects the external capacitor C to ground.

Q.2) Draw the pin diagram of IC 565 and also describe pin 565.

→

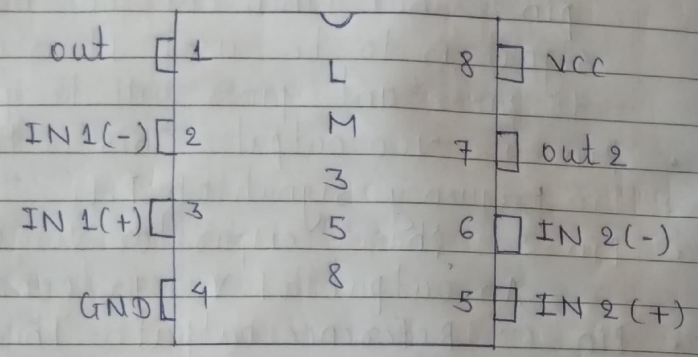


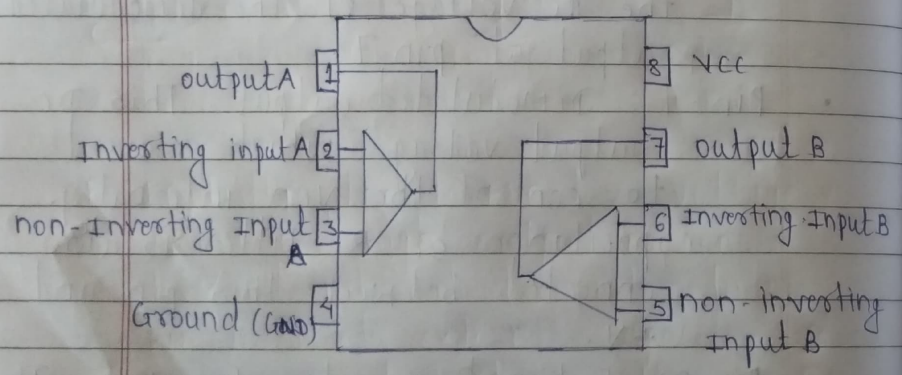
* Pin function :-
- pin-1 :- Negative power supply.
- pin-2 :- Input signal :- The frequency signal which is to be locked is connected here.
- Pin-3 :- Input signal :- It is same as pin 2, But when only one input signal is to be used. It is connected to pin 2 and pin 3 will be grounded.
- Pin-4 :- VCO output :- This is the output Frequency signal of PLL.
- Pin-5 :- Phase Comparator VCO input :- An output of VCO is connected.
- Pin-6 :- Reference output :- This is a DC reference approximately equal to the dc voltage of the demodulated output at pin 7.
- Pin-7 :- Demodulated output :- This DC output voltage which is used to control the VCO internally. Capacitance of RC filter is also connected between this pin and +V.
- Pin 8 :- External resistor for VCO :- The resistor for running oscillation of VCO is connected externally to this pin.
- Pin 9 :- External capacitor for VCO :- The capacitor, for running oscillation of VCO is connected to this pin.

- Pin 10 :- Positive power supply.
- Pin -11 -12 -13 -14 :- NC :- Not connected.

Q.3) Draw the pin diagram of LM358 and also describe pin of LM358 op-AMP.

→





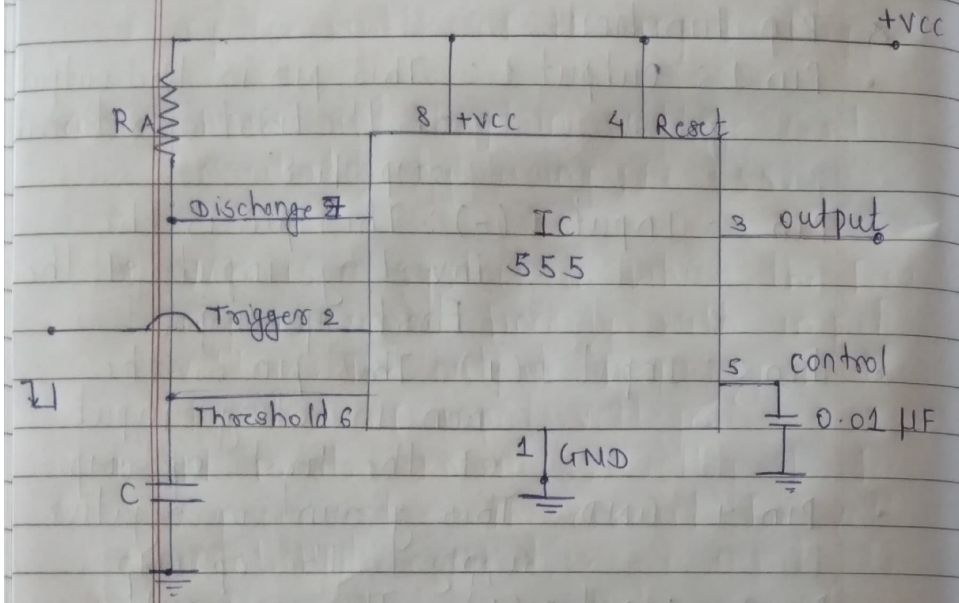
★ Features :-
- Short circuit Protected outputs
- True Differential Input stage
- single supply operations : 3.0 V to 32 V
- Low input Bias currents
- Internally compensated
- COMMON Mode Range Extends to Negative supply.
- single and split supply operation.

★ Pin functions :-
- Pin 1 :- Output 1 :- This pin is the output of the first operational amplifier.
- Pin 2 :- Input 1 (-) :- This pin is the inverting input of the first op-amp.
- Pin 3 :- Input 1 (+) :- This pin is the non-inverting input of the first op-amp.
- Pin 4 :- GND :- This ground or negative supply to the op-amp.
- Pin 5 :- Input 2 (+) :- This pin is the non-inverting input of the second op-amp.
- Pin 6 :- Input 2 (-) :- This pin is the inverting input of the second op-amp.
- Pin 7 :- output 2 :- This pin is the output of the second op-amp.
- Pin -8 :- This pin is the positive voltage supply to both op-amp.

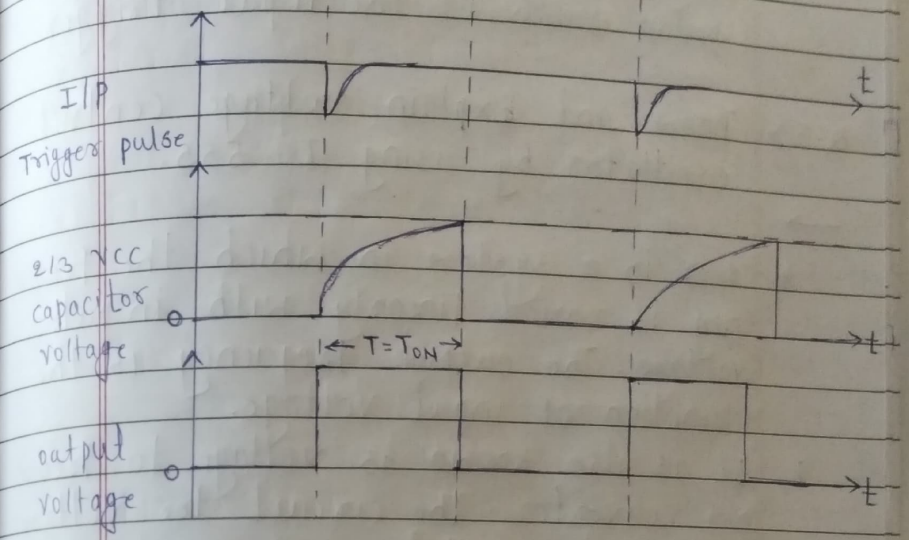
Q.4) Draw and Explain Monostable Multivibrator by using IC555.

→

* Monostable is a circuit which has one stable state and one quasi-stable state.



* Initially, the terminal 2, voltage is maintained high output is considered low and internal transistor turns ON which shorts external capacitor. $V_C = 0V$
* Now capacitor changes through RA towards VCC.
* When capacitor voltage reaches to 2/3 VCC the threshold comparator resets the Flip Flop and output goes low. This turns ON internal transistor.
* Now capacitor discharges suddenly through the transistor. Thus it remains OFF till next trigger.
* The pulse width of the input must be smaller than the expected pulse width of output waveform.



* The capacitor charging voltage is given by $V_C = V_{CC}(1 - e^{-t/RC})$

∴ T_{ON} period is calculated till $V_C = 2/3\ V_{CC}$

$$\frac{2}{3} V_{CC} = V_{CC}(1 - e^{-t/RC})$$

$$\frac{2}{3} = 1 - e^{-t/RC}$$

$$\frac{-1}{3} = -e^{-t/RC}$$

$$\log_e\left(\frac{1}{3}\right) = \log_e\left(e^{-t/RC}\right)$$

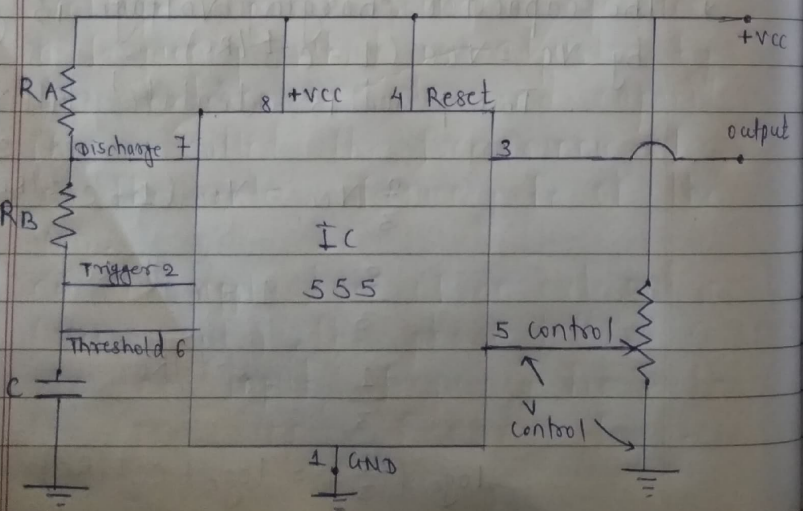
$$-1.1 = -t/RC$$
$$t = 1.1\,RC$$

* Since, the output pulse width of the Monostable Multivibrator is of longer duration than the negative pulse width of the input trigger.

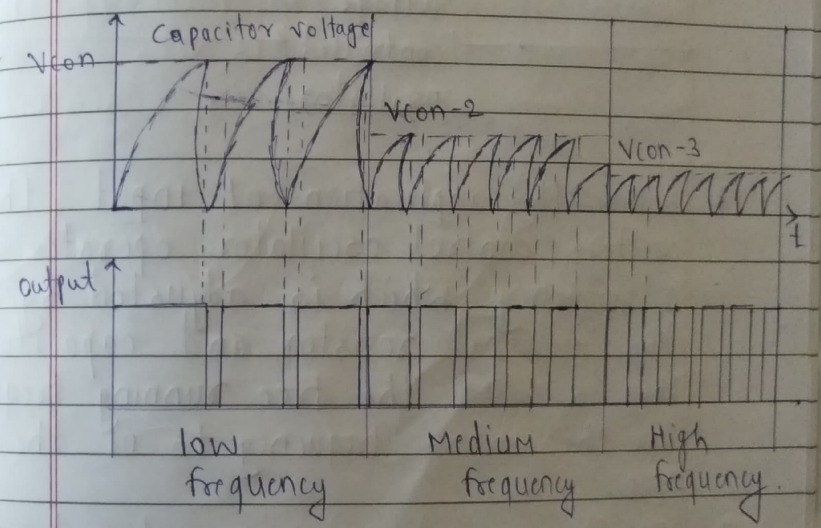
Q.5) Draw and Explain voltage control oscillator by using IC 555.

→

* VCO is a system in which the frequency is varied linearly with input voltage. The output frequency is the function of input voltage.
* It is also known as voltage to frequency convertor. The circuit for voltage controlled oscillator.



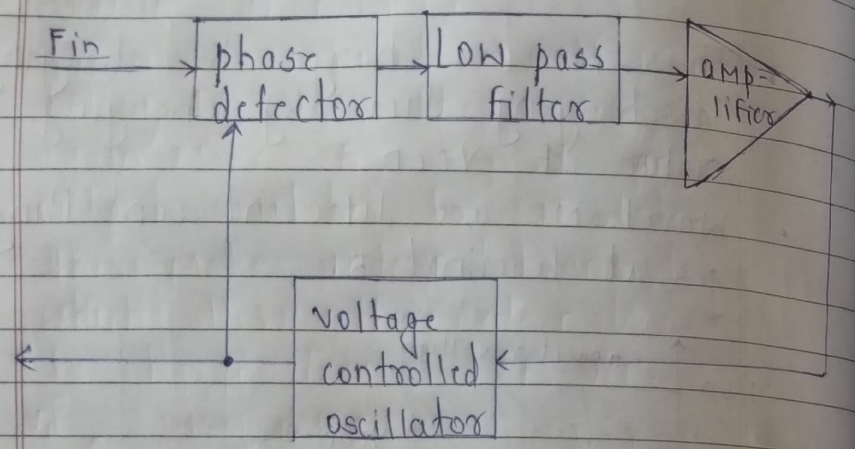
* The timer is connected in 'Astable Mode'.
* Since the reference voltage is varied, the charging and discharging voltages of the astable multivibrator changes.
* Now the capacitor charges from V_{con} to V_{con} and discharges from V_{con} to $\frac{V_{con}}{2}$.
* Thus we can change the charging and discharging time by changing the value of V_{con}.
* Waveforms are different controlling voltages at terminal 5.



Q.6) Draw the block diagram of phase locked loop and explain working of PLL.

→ * Definition :-
It is defined as a control Mechanism by which the frequency and phase of the incoming signal is synchronized with the frequency and phase of the VCO.

* Block Diagram :-



* Before the application of input, the voltage control oscillator operates at a frequency which is adjusted by the external resistor and capacitor. This is called the free running frequency or the centers frequency f_0 of the VCO.

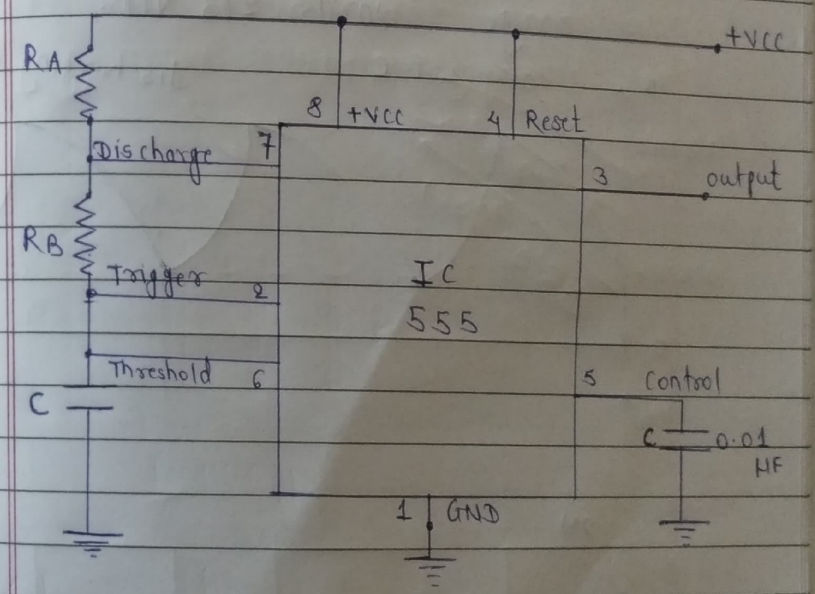
* On the application of an external input ($V_M \sin \omega t$), the phase detector compares the input frequency fin with the free running frequency fout.

* The output voltage of the phase detector is proportional to a dc voltage and often referred as the error voltage.

* When the two inputs to the phase detector are equal, the output of the phase detector has an error voltage which is just sufficient to keep the VCO operate at the frequency of the incoming signal fin. This mode of operation is the phase locked mode.

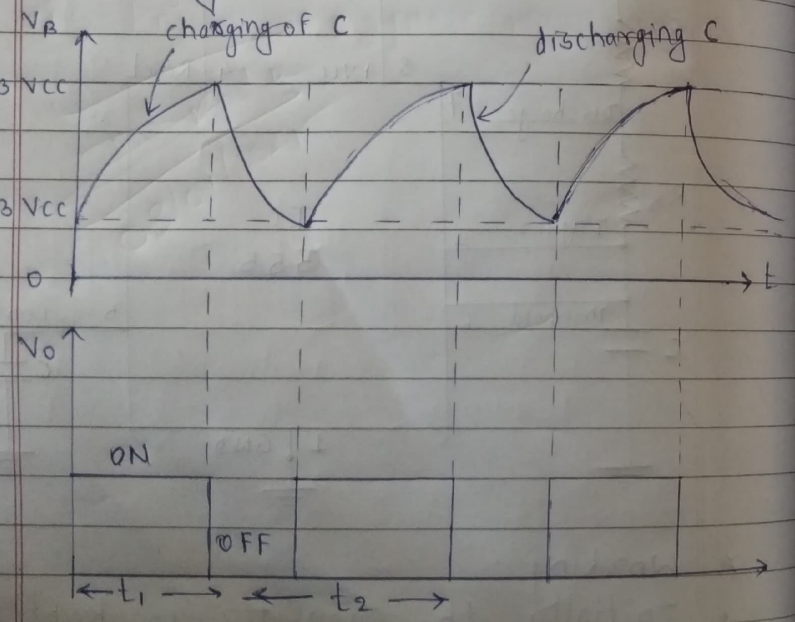
Q.7) Draw & Explain Astable Multivibrator by using IC 555.

→



* Working :-
• Initially the capacitor is kept unch-arged.

- When the Vcc is Applied, the internal transistor is held off and the output goes high.
- When capacitor voltage falls below 2/3 Vcc, the threshold comparator gives logic zero of at the output which makes R, S = 0, 0 and output remains 'unchanged'.
- When the capacitor voltage falls below 1/3 Vcc, the lower comparator again makes R, S = 0, 1 and the output switches to ON state.
- This makes internal transistor off which allows the external capacitor to charge towards 2/3 Vcc.



* Capacitor changes and discharge between 2/3 Vcc and 1/3 Vcc hence capacitor voltage is given by,

$$V_c = \text{D.C level} + \text{charging voltage of capacitor}$$

$$\therefore V_c = \frac{1}{3} V_{cc} + \frac{2}{3} V_{cc} \left(1 - e^{-t_1/RC}\right)$$

$$= \frac{1}{3} V_{cc} + \frac{2}{3} V_{cc} \left(1 - e^{\frac{-t_1}{(R_A+R_B)C}}\right)$$

* T_1 is calculated when,

$$\therefore V_c = \frac{2}{3} V_{cc}$$

$$\therefore \frac{2}{3} V_{cc} = \frac{1}{3} V_{cc} + \frac{2}{3} V_{cc} \left(1 - e^{\frac{-t_1}{(R_A+R_B)C}}\right)$$

Dividing by $\frac{2}{3} V_{cc}$ on both the sides :-

$$1 = \frac{1}{2} + \left[1 - e^{\left(\frac{-t_1}{(R_A+R_B)C}\right)}\right]$$

$$\therefore \frac{1}{2} = e^{-t_1/(R_A+R_B)C}$$

* Taking natural log on both the sides,

$$\ln\left(\frac{1}{2}\right) = \ln e^{\frac{-t_1}{(R_A+R_B)C}}$$

$$\therefore t_1 = 0.693 (R_A + R_B) C$$

* While discharging, the capacitor discharges through R_B,

$$\therefore t_2 = 0.693 (R_B) C$$

$\therefore T = t_1 + t_2$

$= 0.692\,(R_A + 2R_B)\,C$

* The frequency of oscillation is,

$$f = \frac{1}{T} = \frac{1.44}{(R_A + 2R_B)\,C}$$

* The duty cycle of the output is given by,

$$\%D = \frac{100}{T} = \frac{t_1}{T} = \frac{R_A + R_B}{R_A + 2R_B} \times 100\%$$

It is always greater than 50%, If the duty cycle of Astable MV is adjusted to 50% the circuit is known as square wave generator.

03/04/2021