**Control computer system with hand gestures using arduino**

BY

S.S.V.SURYA TEJA

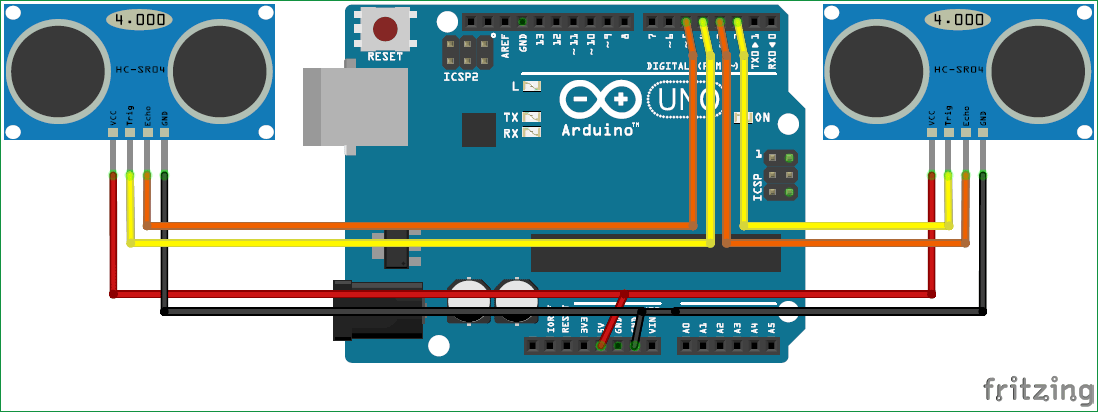
AIM: **Gesture control Laptop/Computer by combining the Power of Arduino and Python**.

**REQUIREMENTS: ULTRASONIC SENSORS,PYTHON,CONNECTING WIRES.**

### Concept behind the project:

The concept behind the project is very simple. We will place two Ultrasonic (US) sensors on top of our monitor and will read the distance between the monitor and our hand using Arduino, based on this value of distance we will perform certain actions. To perform actions on our computer we use Python **pyautogui** library. The commands from Arduino are sent to the computer through serial port (USB). This data will be then read by python which is running on the computer and based on the read data an action will be performed

### Circuit Diagram:

****

**Arduino Code:**

const int trigger1 = 2; //Trigger pin of 1st Sesnor  
const int echo1 = 3; //Echo pin of 1st Sesnor  
const int trigger2 = 4; //Trigger pin of 2nd Sesnor  
const int echo2 = 5;//Echo pin of 2nd Sesnor

long time\_taken;  
int dist,distL,distR;

void setup() {  
Serial.begin(9600);   
    
pinMode(trigger1, OUTPUT);   
pinMode(echo1, INPUT);   
pinMode(trigger2, OUTPUT);   
pinMode(echo2, INPUT);   
}

/\*###Function to calculate distance###\*/  
void calculate\_distance(int trigger, int echo)  
{  
digitalWrite(trigger, LOW);  
delayMicroseconds(2);  
digitalWrite(trigger, HIGH);  
delayMicroseconds(10);  
digitalWrite(trigger, LOW);

time\_taken = pulseIn(echo, HIGH);  
dist= time\_taken\*0.034/2;  
if (dist>50)  
dist = 50;  
}

void loop() { //infinite loopy  
calculate\_distance(trigger1,echo1);  
distL =dist; //get distance of left sensor

calculate\_distance(trigger2,echo2);  
distR =dist; //get distance of right sensor

//Uncomment for debudding  
/\*Serial.print("L=");  
Serial.println(distL);  
Serial.print("R=");  
Serial.println(distR);  
\*/

//Pause Modes -Hold  
if ((distL >40 && distR>40) && (distL <50 && distR<50)) //Detect both hands  
{Serial.println("Play/Pause"); delay (500);}

calculate\_distance(trigger1,echo1);  
distL =dist;

calculate\_distance(trigger2,echo2);  
distR =dist;

//Control Modes  
//Lock Left - Control Mode  
if (distL>=13 && distL<=17)  
{  
  delay(100); //Hand Hold Time  
  calculate\_distance(trigger1,echo1);  
  distL =dist;  
  if (distL>=13 && distL<=17)  
  {  
    Serial.println("Left Locked");  
    while(distL<=40)  
    {  
      calculate\_distance(trigger1,echo1);  
      distL =dist;  
      if (distL<10) //Hand pushed in   
      {Serial.println ("Vup"); delay (300);}  
      if (distL>20) //Hand pulled out  
      {Serial.println ("Vdown"); delay (300);}  
    }  
  }  
}

//Lock Right - Control Mode  
if (distR>=13 && distR<=17)  
{  
  delay(100); //Hand Hold Time  
  calculate\_distance(trigger2,echo2);  
  distR =dist;  
  if (distR>=13 && distR<=17)  
  {  
    Serial.println("Right Locked");  
    while(distR<=40)  
    {  
      calculate\_distance(trigger2,echo2);  
      distR =dist;  
      if (distR<10) //Right hand pushed in  
      {Serial.println ("Rewind"); delay (300);}  
      if (distR>20) //Right hand pulled out  
      {Serial.println ("Forward"); delay (300);}  
  }  
}  
}

delay(200);  
}

**Python Code:**

import serial #Serial imported for Serial communication  
import time #Required to use delay functions  
import pyautogui

ArduinoSerial = serial.Serial('com18',9600) #Create Serial port object called arduinoSerialData  
time.sleep(2) #wait for 2 seconds for the communication to get established

while 1:  
    incoming = str (ArduinoSerial.readline()) #read the serial data and print it as line  
    print incoming  
      
    if 'Play/Pause' in incoming:  
        pyautogui.typewrite(['space'], 0.2)

    if 'Rewind' in incoming:  
        pyautogui.hotkey('ctrl', 'left')

    if 'Forward' in incoming:  
        pyautogui.hotkey('ctrl', 'right')

    if 'Vup' in incoming:  
        pyautogui.hotkey('ctrl', 'down')

    if 'Vdown' in incoming:  
        pyautogui.hotkey('ctrl', 'up')

    incoming = "";