**FIRE ALARM AND AUTO BRAKE SYSTEM**

int flame\_sensor\_pin = 10 ;// initializing pin 12 as the sensor output pin  
int flame\_pin = HIGH ; // state of sensor  
const int lm35\_pin = A1;  
int motor1pin1 = 2;  
int motor1pin2 = 3;  
int motor2pin1 = 4;  
int motor2pin2 = 5;  
#define echoPin 9  
#define trigPin 8  
long duration; // variable for the duration of sound wave travel  
int distance;  
void setup() {  
  // put your setup code here, to run once:  
  pinMode(motor1pin1, OUTPUT);  
  pinMode(motor1pin2, OUTPUT);  
  pinMode(motor2pin1, OUTPUT);  
  pinMode(motor2pin2, OUTPUT);  
  pinMode(13,OUTPUT);  
  pinMode(12,OUTPUT);  
  pinMode ( flame\_sensor\_pin , INPUT );  
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an OUTPUT  
  pinMode(echoPin, INPUT);  
  Serial.begin(9600); // // Serial Communication is starting with 9600 of baudrate speed  
}  
  
void loop() {  
  int temp\_adc\_val;  
  float temp\_val;  
  temp\_adc\_val = analogRead(lm35\_pin);  /\* Read Temperature \*/  
  temp\_val = (temp\_adc\_val \* 4.88); /\* Convert adc value to equivalent voltage \*/  
  temp\_val = (temp\_val/10); /\* LM35 gives output of 10mv/°C \*/  
  Serial.print("Temperature = ");  
  Serial.print(temp\_val);  
  Serial.print(" Degree Celsius\n");  
Serial.print("Pin A0: ");  
flame\_pin = digitalRead ( flame\_sensor\_pin ) ;  
  if(temp\_val > 70 || flame\_pin == LOW  ){  
  digitalWrite(motor2pin1, HIGH);  
  digitalWrite(motor2pin2, LOW);  
   digitalWrite(12, HIGH);  
     digitalWrite(motor1pin1, LOW);  
  digitalWrite(motor1pin2, LOW);  
    delay(30000);  
  }  
  else{  
  digitalWrite(trigPin, LOW);  
  delayMicroseconds(2);  
  // Sets the trigPin HIGH (ACTIVE) for 10 microseconds  
  digitalWrite(trigPin, HIGH);  
  delayMicroseconds(10);  
  digitalWrite(trigPin, LOW);  
  // Reads the echoPin, returns the sound wave travel time in microseconds  
  duration = pulseIn(echoPin, HIGH);  
  // Calculating the distance  
  distance = duration \* 0.034 / 2; // Speed of sound wave divided by 2 (go and back)  
  // Displays the distance on the Serial Monitor  
  // put your main code here, to run repeatedly:  
    if(distance<30){    
  digitalWrite(motor1pin1, LOW);  
  digitalWrite(motor1pin2, LOW);  
    }  
    else{  
  digitalWrite(motor1pin1, HIGH);  
  digitalWrite(motor1pin2, LOW);  
    }  
  }  
}