**Fire Fighting Robotic Car using IOT**

The vehicle controller programming code using Arduino IDE

#define motora1Pin 4

#define motora2Pin 5

#define motorb1Pin 6

#define motorb2Pin 7

#define trigPin 3

#define echoPin 2

#define trigrightPin 12

#define echorightPin 8

#define flamecarPin A0 #define flame2Pin A1

#define pumpPin 13

int distance = 0;

int distanceright = 0; int counter;

int coun;

int durationright; int duration;

int flamesignal1; int flamesignal2;

unsigned long pulseTime = 0; //to be used in the function of the ultrasonic sensor to calculate the distance through the high level time

void setup() { Serial.begin (9600);

pinMode(trigPin, OUTPUT); pinMode(echoPin, INPUT); pinMode(motora1Pin, OUTPUT); pinMode(motora2Pin, OUTPUT); pinMode(motorb1Pin, OUTPUT); pinMode(motorb2Pin, OUTPUT); pinMode(trigrightPin, OUTPUT); pinMode(echorightPin, INPUT); pinMode(flamecarPin, INPUT); pinMode(flame2Pin, INPUT);

}

void loop() {

if (analogRead(flame2Pin>150){ //if flame is detected in room

if (analogRead(flamecarPin)<150){ //if the flame sensor on the car cannot see the flame findDistance(); //function to calculate the distance from the front ultrasonic sensor

if (distance >= 500 || distance <= 0){ Serial.println("Out of range");

}

else{ Serial.println("sensor1"); Serial.print(distance);

}

if (distance <= 0 || distance >= 30)

{

Forward(); //moves the robot forward

}

else

{ Stop();

delay(3000); //some time

Right(); //turn right delay(450);

Stop(); delay(2000);

findDistanceSide(); //function to find the distance on the left counter=0;

while( distanceright !=0 && distanceright <= 30) {//continue until end of obstacle Forward();

counter++; //to increment the counter to be used later delay(1);

findDistanceSide(); Serial.println("sensorside"); Serial.println(distanceright);

}

Stop(); delay(3000); //some time

Backward(); //to decrease the error of the ultrasonic sensor delay(200);

Left(); //straight again delay(600);

Stop();

delay(2000); findDistanceSide();

while( distanceright !=0 && distanceright <= 50) //continue until obstacle is away

{

Forward(); findDistanceSide();

}

Forward; delay(800);

Stop(); delay(3000);

Left(); delay(600); // Stop(); delay(2000);

while(counter!=0) {Forward();

coun= counter-1; //to decrement the counter which was incremented before counter=coun;

delay(9); }

Forward(); delay(1000);

Stop();

delay(3000); //now back in place Right(); delay(500); //now back straight Stop(); delay(3000);

}

}

else {

Stop();

}

while (analogRead(flamecarPin>150){ //while the fire is on digitalWrite(pumpPin,HIGH); //pump the water to extinguish the fire

}

}

delay(50);

}

long findDistance() { //function to find the distance in front of the robot digitalWrite(trigPin, HIGH);

delayMicroseconds(1000); digitalWrite(trigPin, LOW); duration = pulseIn(echoPin, HIGH); distance = (duration/2) / 29.1;

}

long findDistanceSide() { //function to find the distance on the left side of the robot digitalWrite(trigrightPin, HIGH);

delayMicroseconds(1000); digitalWrite(trigrightPin, LOW); durationright = pulseIn(echorightPin, HIGH); distanceright = (durationright/2) / 29.1;

}

long findflamesignal1() {

flamesignal1=analogRead(flamecarPin);

}

long findflamesignal2(){

flamesignal2= analogRead(flame2Pin);

}

void Forward() { digitalWrite(motora1Pin,HIGH); digitalWrite(motorb2Pin,HIGH); digitalWrite(motorb1Pin,LOW); digitalWrite(motora2Pin,LOW);}

void Backward() { digitalWrite(motora1Pin,LOW); digitalWrite(motorb2Pin,LOW); digitalWrite(motorb1Pin,HIGH); digitalWrite(motora2Pin,HIGH);}

void Right() {

digitalWrite(motora1Pin,HIGH); digitalWrite(motorb1Pin,HIGH); digitalWrite(motora2Pin,LOW); digitalWrite(motorb2Pin,LOW);

}

void Left() {

digitalWrite(motora1Pin,LOW);

digitalWrite(motorb2Pin,HIGH); digitalWrite(motorb1Pin,LOW); digitalWrite(motora2Pin,HIGH);

}

void Stop() {

digitalWrite(motora1Pin,LOW); digitalWrite(motorb2Pin,LOW); digitalWrite(motorb1Pin,LOW); digitalWrite(motora2Pin,LOW); }

**Code for GSM Module-**

#include<SoftwareSerial.h>

SoftwareSerial gsm(2,3);

void setup()

{

delay(10000);

Serial.begin(9600);

gsm.begin(9600);

gsm.println("AT+CMGF=1");

delay(1000);

gsm.println("AT+CMGS=\"+91xxxxxxxxxx\"\r"); //replace x by your number

delay(1000);

gsm.println("hello");

delay(100);

gsm.println((char)26);

delay(1000);

}

void loop()

{

}

**Call Program for GSM**

#include<SoftwareSerial.h>

SoftwareSerial gsm(2,3);

void setup()

{

delay(10000);

Serial.begin(9600);

gsm.begin(9600);

gsm.println("ATDxxxxxxxxxx;"); //replace x by your number

delay(100);

gsm.println("ATH");

}

void loop()

{

}

**GSM Test Program**

#include<SoftwareSerial.h>

SoftwareSerial gsm(2,3);

void setup()

{

Serial.begin(9600);

gsm.begin(9600);

}

void loop()

{

if(gsm.available())

{

Serial.write(gsm.read());

}

if(Serial.available())

{

byte a=Serial.read();

if(a=='#')

{

gsm.write( 0x1a );

}

else

{

gsm.write(a);

}

}

}