CARRO CON SENSOR INFRARROJO

1. Codigo

#include <Servo.h>

#include <IRremote.h>

int izquierda\_motor\_retroceder = 9;//IN1  
int izquierda\_motor\_avanzar = 5;//IN2  
int derecha\_motor\_avanzar = 6;//IN3  
int derecha\_motor\_retroceder = 10;//IN4  
int RECV\_PIN = 8;//Control remoto  
IRrecv irrecv(RECV\_PIN);  
decode\_results results;  
int trig =A0; // Trigger p2.0  
int echo = A1; // Echo p2.1

Servo myServo;  
int pos = 0;  
int servo = 11;

/\*Sirena policial\*/  
int duracion=250; //Duración del sonido  
int fMin=2000; //Frecuencia más baja que queremos emitir  
int fMax=4000; //Frecuencia más alta que queremos emitir  
int i=0;

void setup(){  
 Serial.begin(9600);  
 irrecv.enableIRIn();  
 pinMode(izquierda\_motor\_retroceder,OUTPUT);  
 pinMode(izquierda\_motor\_avanzar,OUTPUT);   
 pinMode(derecha\_motor\_avanzar,OUTPUT);  
 pinMode(derecha\_motor\_retroceder,OUTPUT);   
 pinMode(echo,INPUT);  
 pinMode(trig,OUTPUT);  
 myServo.attach(servo);  
}

void loop(){  
 medir\_distancia();  
 if(irrecv.decode(&results)){   
 Serial.println(results.value,HEX);  
 irrecv.resume();  
 delay(5);

if(results.value==0x00FF02FD){  
 avanzar(2000);  
 }

if(results.value==0x00FF9867){  
 retroceder(2000);  
 }

if(results.value==0x00FF906F){  
 derecha(2000);  
 }

if(results.value==0x00FFE01F){  
 izquierda(2000);  
 }

if(results.value==0x00FF22DD){  
 giro\_derecha(2000);  
 }

if(results.value==0x00FFC23D){  
 giro\_izquierda(2000);  
 }

if(results.value==0x00FFA857){  
 detener(2000);  
 }

if(results.value==0x00FFA25D){  
 avanzar(2000);  
 detener(1000);  
 retroceder(2000);  
 detener(1000);  
 derecha(2000);  
 detener(1000);  
 izquierda(2000);  
 detener(1000);  
 giro\_derecha(2000);  
 detener(1000);  
 giro\_izquierda(2000);  
 detener(1000);  
 }   
 }  
}

void detener(int tiempo){  
 analogWrite(izquierda\_motor\_avanzar,0);  
 analogWrite(izquierda\_motor\_retroceder,0);  
 analogWrite(derecha\_motor\_avanzar,0);  
 analogWrite(derecha\_motor\_retroceder,0);  
 delay(tiempo);  
}

void avanzar(int tiempo){  
 analogWrite(izquierda\_motor\_avanzar,255);  
 analogWrite(izquierda\_motor\_retroceder,0);  
 analogWrite(derecha\_motor\_avanzar,255);  
 analogWrite(derecha\_motor\_retroceder,0);  
 delay(tiempo);  
}

void retroceder(int tiempo){  
 analogWrite(izquierda\_motor\_avanzar,0);  
 analogWrite(izquierda\_motor\_retroceder,255);  
 analogWrite(derecha\_motor\_avanzar,0);  
 analogWrite(derecha\_motor\_retroceder,255);  
 delay(tiempo);  
}

void derecha(int tiempo){  
 analogWrite(izquierda\_motor\_avanzar,0);  
 analogWrite(izquierda\_motor\_retroceder,0);  
 analogWrite(derecha\_motor\_avanzar,255);  
 analogWrite(derecha\_motor\_retroceder,0);  
 delay(tiempo);  
}

void izquierda(int tiempo){  
 analogWrite(izquierda\_motor\_avanzar,255);  
 analogWrite(izquierda\_motor\_retroceder,0);  
 analogWrite(derecha\_motor\_avanzar,0);  
 analogWrite(derecha\_motor\_retroceder,0);  
 delay(tiempo);  
}

void giro\_derecha(int tiempo){  
 analogWrite(izquierda\_motor\_avanzar,0);  
 analogWrite(izquierda\_motor\_retroceder,255);  
 analogWrite(derecha\_motor\_avanzar,255);  
 analogWrite(derecha\_motor\_retroceder,0);  
 delay(tiempo);  
}

void giro\_izquierda(int tiempo){  
 analogWrite(izquierda\_motor\_avanzar,255);  
 analogWrite(izquierda\_motor\_retroceder,0);  
 analogWrite(derecha\_motor\_avanzar,0);  
 analogWrite(derecha\_motor\_retroceder,255);  
 delay(tiempo);  
}

void medir\_distancia(){  
 digitalWrite(trig,LOW);  
 delayMicroseconds(2);  
 digitalWrite(trig,HIGH);  
 delayMicroseconds(2);  
 digitalWrite(trig,LOW);  
 float distancia = pulseIn(echo,HIGH);  
 distancia = distancia/58;  
 if(distancia<10){  
 detener(1000);  
 retroceder(500);  
 mover\_servo();  
 }  
 Serial.println(distancia);  
}

void mover\_servo(){  
 int izquierda=0;  
 int derecha =0;  
 for(pos=0;pos<=180;pos+=1){  
 myServo.write(pos);  
 delay(15);  
 }  
 for(pos=180;pos>=90;pos-=1){  
 myServo.write(pos);  
 delay(15);  
 }  
}