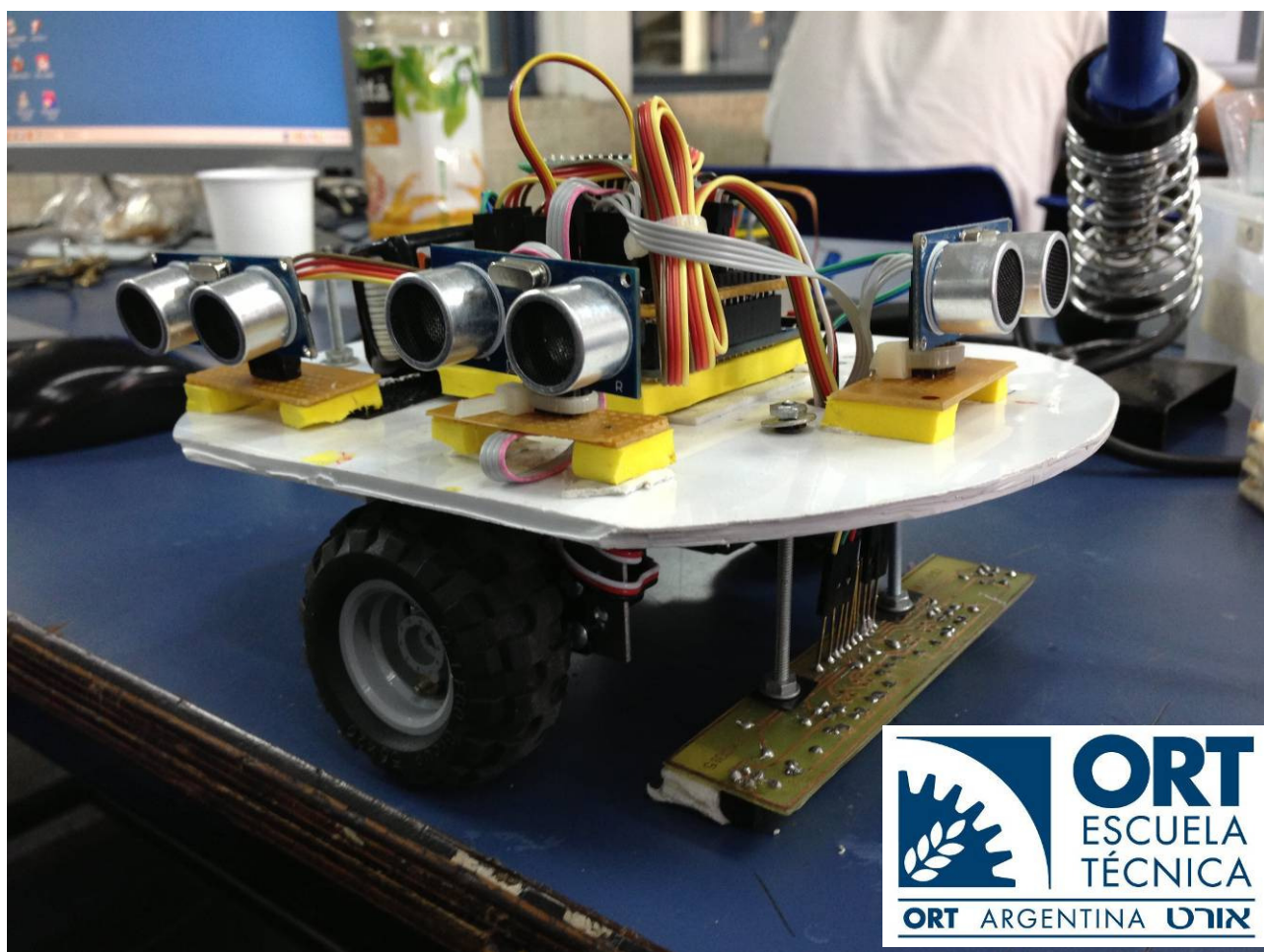
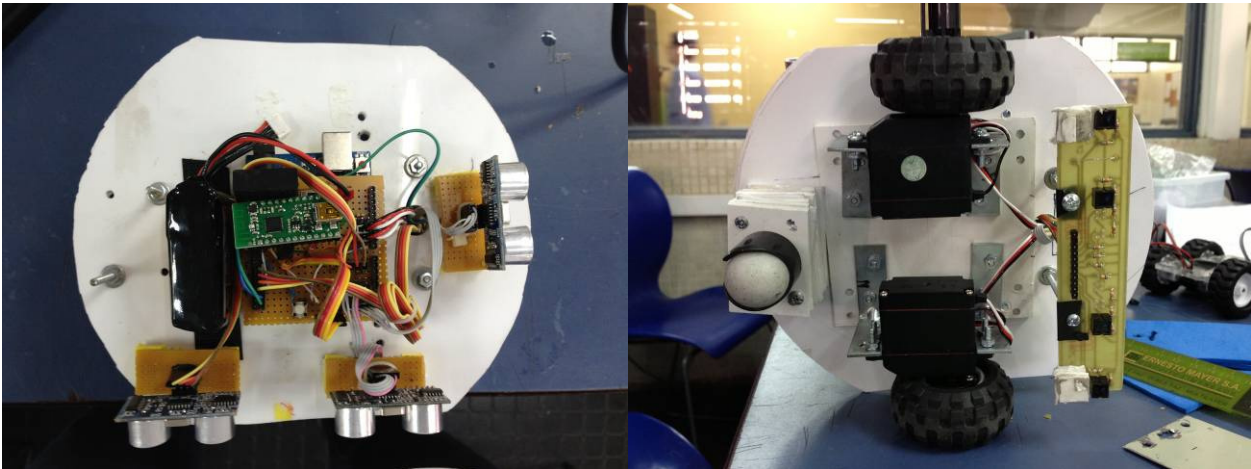


El “tratorcito”

Escuela Técnica ORT, sede Almagro



Mecánica



Vista superior

Vista Inferior

- Dimensiones: 170 x 200mm. Altura 110mm
- Ruedas de 55mm de diámetro
- Estructura: Placa circular de alto impacto de 3mm (Diámetro:200mm)
- Tracción diferencial (dos motores en el centro del robot)
- Punto de apoyo, tipo ball-caster
- Peso: 620 gramos
- Motores: 2 servos trucados, generando servos de rotacion continua.

Electrónica

- Controlador Arduino UNO (circuito esquemático en anexo)
- Regulador de tensión
- Sensores reflectivos CNY70 (circuito esquemático en anexo)
- Sensores De proximidad (ultrasonido) Hc-sr04
- Modulo UART inalámbrico (WIXEL).

Funcionamiento básico

Este robot, para encontrar la salida, busca doblar a la derecha siempre que tenga la posibilidad. Si no, se moverá en línea recta hasta que no tenga otra posibilidad más que doblar a la izquierda. Por lo tanto el giro de la izquierda se utilizara únicamente cuando tenga paredes a la derecha y a la izquierda.

El funcionamiento esta armado básicamente por una maquina de estados, la cual tiene 3 estados. El estado de doblar a la derecha, el de doblar a la izquierda y de moverse para adelante.

El programa entrara al estado de doblar a la izquierda cuando detecte paredes a la derecha y a la izquierda a menos de 20cm. En ese momento girará a la izquierda por un tiempo ya establecido, luego de ese tiempo el robot continua girando hasta que el sensor derecho de proximidad detecte una pared a una distancia pequeña.

El programa entrara al estado de doblar a la derecha cuando no detecte una pared a la derecha a una distancia corta. En ese momento girará a la derecha por un tiempo ya establecido, luego de ese tiempo el robot avanza hasta que el sensor derecho de proximidad detecte una pared a una distancia pequeña.

El estado de moverse para adelante se “encarga” de mantener el robot lo mas derecho posible. El robot tiene dos sensores en el costado derecho del robot. El que esta mas adelante es el que funciona en los otros dos estados. En este estado el robot cambia la velocidad de uno de los dos motores, dependiendo de la distancia a la que se encuentra el sensor derecho que esta mas adelante. Con el que esta mas atrás se evita que el robot gire demasiado, ya sea para de derecha como para la izquierda. (Este es uno de los problemas que tuvimos, ya que nos fue difícil lograr que el robot siga derecho la pared).

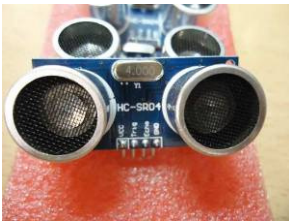
NOTA: se cuenta con la posibilidad de enviar y recibir valores por la UART para debug de forma inalámbrica (modulo WIXEL)

Componentes y precios

Costo total: \$912

Hc-sr04 Sensor Distancia Por Ultrasonido X3

<http://articulo.mercadolibre.com.ar/MLA-460898542-hc-sr04-sensor-distancia-por-ultrasonido-arduino-robotica- JM>



Costo: \$120

Wixel Pair + USB Cable

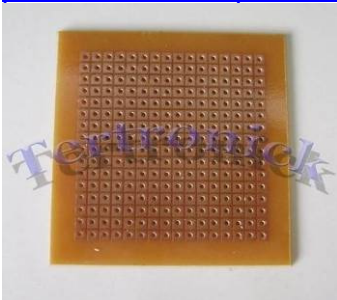
<http://www.pololu.com/catalog/product/1339>



Costo: \$212

Placa Experimental

<http://articulo.mercadolibre.com.ar/MLA-454522664-placa-experimental-pertinax-5x5cm-para-soldar-componentes- JM>



Costo: \$15

Arduino UNO

<http://articulo.mercadolibre.com.ar/MLA-463708389-arduino-uno-r3- JM>



Costo: \$320

Sensor reflectivo CNY70 X 4



Costo: \$40

SpringRC SM-S4303R Continuous Rotation Servo

<http://www.pololu.com/catalog/product/1248>



Costo: \$132

Pololu Ball Caster Variety Pack

<http://www.pololu.com/catalog/product/949>



Costo: \$132.5

Batería Lipo 11.1v

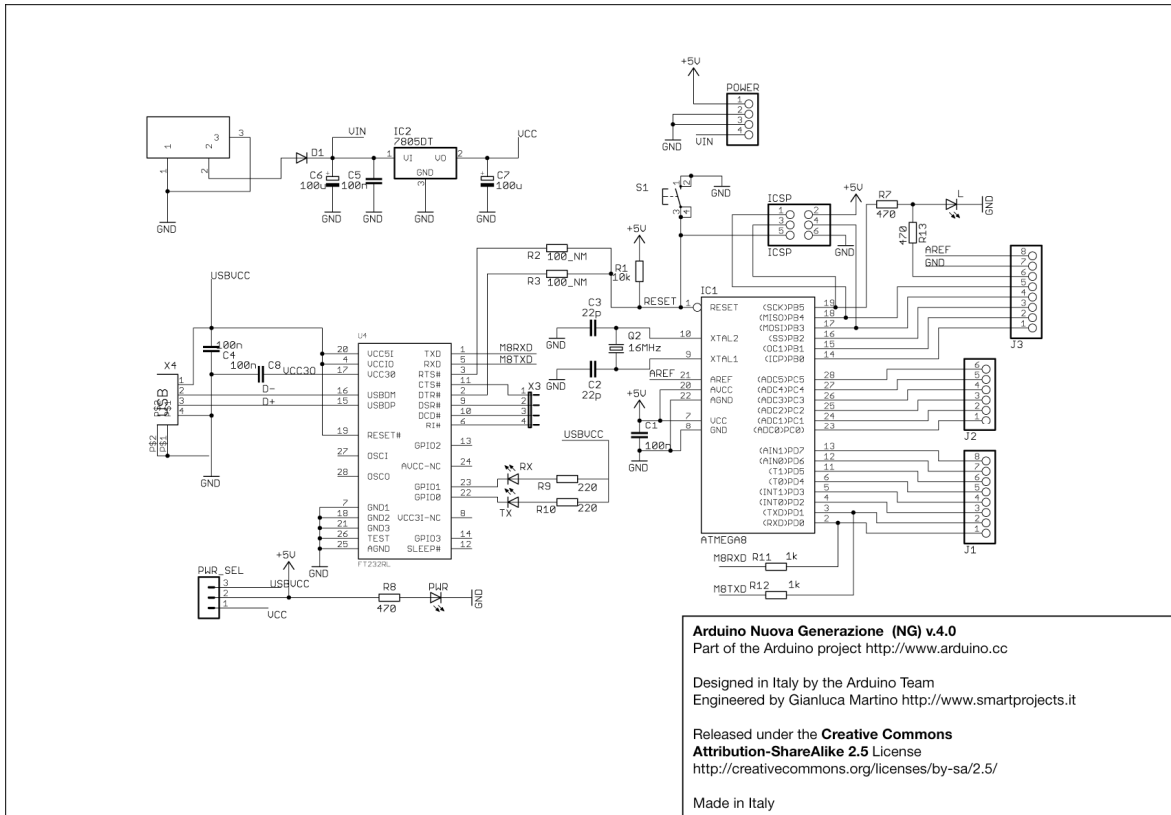
<http://dx.com/p/11-1v-850mah-20c-replacement-li-poly-battery-pack-for-trex-250-rc-helicopter-51164>



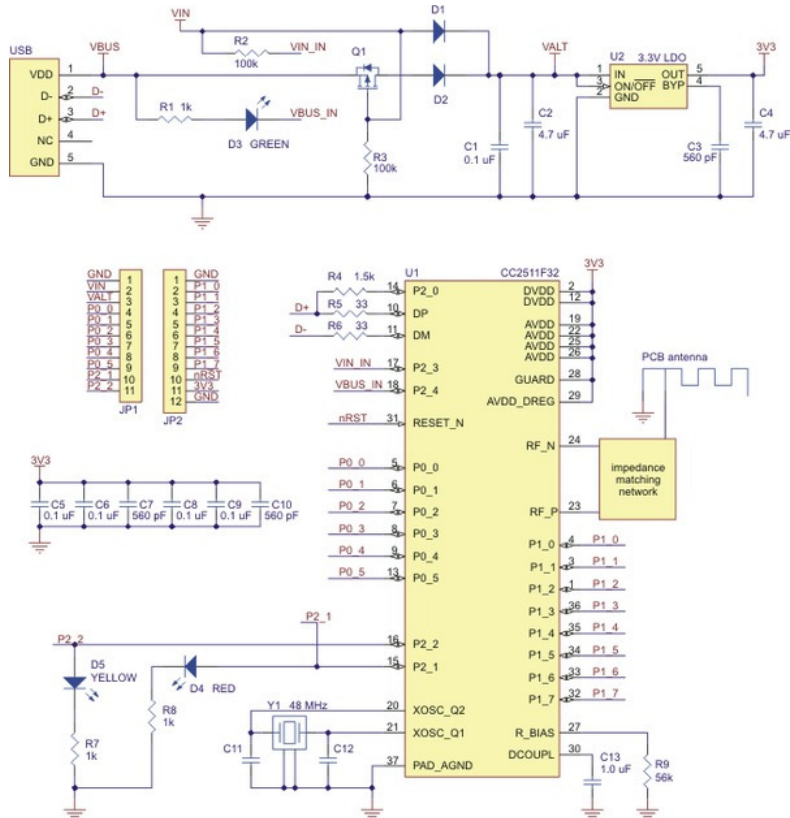
Costo: \$53

Circuitos Esquemáticos

Arduino UNO



WIXEL



Código fuente

```
Motor derecho: Pin 11
Motor izquierdo: Pin 12
Sensor reflectivo derecho: Pin A4
Sensor reflectivo izquierdo: Pin A3
Sensor proximidad 1: Pin 3 y 2
Sensor proximidad 2: Pin 5 y 4
Sensor proximidad 3: Pin 7 y 6

#include <Servo.h>

//=== DEFINICIONES DE PINES ===
const int TrigPin1 = 3;
const int EchoPin1 = 2;
const int TrigPin2 = 5;
const int EchoPin2 = 4;

const int TrigPin3 = 7;
const int EchoPin3 = 6;

const int Boton = 8;
const int Led = 13;

const int MotDPin = 9;
const int MotIPin = 10;
const int ServoDPin = 11;
const int ServoIPin = 12;

int SensorI = A3;
int SensorD = A4;

int maxnegro,minblanco,LimBlanco,LimNegro;

//=====

//--- CONSTANTES ---

#define TIME_OUT 10000// antes 6000
#define vel 50
#define derecho 0
#define izquierda 1
#define derecha 2
#define parar 3
#define atras 4
#define adelante 5
// -----

Servo MotorDer;
Servo MotorIzq;
```


Competencia de Robótica “**MEGAMAN**”. Facultad de Ingeniería
Universidad de Buenos Aires – 8 de Junio de 2013

```
long dist[3];
long ad;
long micro=0;
int inByte = 0;
int der,izq,segundos;

void setup()
{
    Serial.begin(9600);
    MotorIzq.attach(MotIPin);
    MotorDer.attach(MotDPin);

    pinMode(Led, OUTPUT);

    pinMode(TrigPin1, OUTPUT);
    pinMode(TrigPin2, OUTPUT);
    pinMode(TrigPin3, OUTPUT);

    pinMode(EchoPin1, INPUT);
    pinMode(EchoPin2, INPUT);
    pinMode(EchoPin3, INPUT);

    pinMode(Boton, INPUT_PULLUP);

    der=0;
    izq=0;
}

void loop()
{
    int sd,si,valor,valor1,i,valor2;
    int estado=0;

    delay(100);
    //while(digitalRead(Boton)==HIGH){}
    //Calibrar();
    //delay (10000);
    while(digitalRead(Boton)==HIGH){
        medirultra();
        //probarluz();
        probarultra();

        switch (estado){
            case derecha: motores(-vel,vel);
                delay(100);
                estado=parar;
                break;

            case izquierda: motores(vel,-vel);
```

Competencia de Robótica “**MEGAMAN**”. Facultad de Ingeniería
Universidad de Buenos Aires – 8 de Junio de 2013

```
    delay(100);
    estado=parar;
    break;

case parar: motores(0,0);
    break;

case atras: motores(-vel,-vel);
    delay(100);
    estado=parar;
    break;

case adelante: motores(vel,vel);
    delay(100);
    estado=parar;
    break;
}

valor=Serial.read();

if(valor=='i')estado=izquierda;

if(valor=='f')estado=adelante;

if(valor=='d')estado=derecha;

if(valor=='p')estado=parar;

if(valor=='a')estado=atras;
}

//motores(vel,-vel);
//delay(800);
//motores(0,0);
//delay(800);

while(1)
{
    sd = analogRead(SensorD);
    si = analogRead(SensorI);

    switch (estado)
    {
        case derecho: medirultra();

            /*if(valor>11){
                valor=dist[1];
                if(valor-valor1>5){
                    motores(vel,-vel);
                    delay(600);
                }
            }*/
            valor=dist[1]+1;

            /*Serial.print(" valor1: ");
```

Competencia de Robótica “**MEGAMAN**”. Facultad de Ingeniería
Universidad de Buenos Aires – 8 de Junio de 2013

```
Serial.print(valor1);
Serial.print(" valor: ");
Serial.print(valor);
probarultra();
*/

//valor1=valor;

if(valor>dist[2]+2)motores(-vel,vel);
  delay(100);
if(valor+3<dist[2])motores(vel,+vel);
  delay(100);
/*valor2=dist[2]+1;
if(valor2>7)valor2=7;
if(valor2<3)valor2=3;
valor2=map(valor2,7,3,7,3);

if(valor>10)valor=10;
if(valor<2)valor=2;*/
motores(vel*5/valor,vel);
break;

case derecha: motores(vel,vel);
  delay(900);//900
  motores(-vel,vel);
  delay(1200);//1300
  motores(vel,vel);
  while(dist[1]>16){
    medirultra();
  }
  //delay(400);
  estado=derecho;
break;

case izquierda: //valor2=16000;
  motores(vel,-vel);
  delay(700);
  /*for(i=0;i<12;i++){
    medirultra();
    if(dist[1]<valor2)valor2=dist[1];
  }*/
  while(dist[2]-dist[1]>1){
    medirultra();
    motores(-vel,vel);
  }
  estado=derecho;
break;

case parar: motores(0,0);
  break;
}

//Serial.print(" estado: ");
//Serial.print(estado);
```

Competencia de Robótica “**MEGAMAN**”. Facultad de Ingeniería
Universidad de Buenos Aires – 8 de Junio de 2013

```
medirultra());

if(dist[1]>25) estado=derecha;
if((dist[0]<7)&(dist[1]<10)) estado=izquierda;

//Serial.print(" estado: ");
//Serial.print(estado);
//Serial.println();

}
}

void probarultra(){
//medirtodo();
Serial.print(" 1: ");
Serial.print(dist[0]);
Serial.print(" 2: ");
Serial.print(dist[1]);
Serial.print(" 3: ");
Serial.print(dist[2]);

Serial.println();
//delay(50);
}

void medirultra(void)
{
delay(10);
dist[0]=medir(1);
delay(10);
dist[1]=medir(2);
delay(10);
dist[2]=medir(3);
}

long medir(int nrosensor)
{
long duration, cm;

switch(nrosensor)
{
case 1:
digitalWrite(TrigPin1, LOW);
delayMicroseconds(2);
digitalWrite(TrigPin1, HIGH);
delayMicroseconds(10);
digitalWrite(TrigPin1, LOW);
duration = pulseIn(EchoPin1, HIGH, TIME_OUT);
```

Competencia de Robótica “**MEGAMAN**”. Facultad de Ingeniería
Universidad de Buenos Aires – 8 de Junio de 2013

```
if (duration==0) duration=TIME_OUT;  
cm = microsecondsToCentimeters(duration);  
break;
```

```
case 2:  
digitalWrite(TrigPin2, LOW);  
delayMicroseconds(2);  
digitalWrite(TrigPin2, HIGH);  
delayMicroseconds(10);  
digitalWrite(TrigPin2, LOW);  
duration = pulseIn(EchoPin2, HIGH,TIME_OUT);  
if (duration==0) duration=TIME_OUT;  
cm = microsecondsToCentimeters(duration);  
break;
```

```
case 3:  
digitalWrite(TrigPin3, LOW);  
delayMicroseconds(2);  
digitalWrite(TrigPin3, HIGH);  
delayMicroseconds(10);  
digitalWrite(TrigPin3, LOW);  
duration = pulseIn(EchoPin3, HIGH,TIME_OUT);  
if (duration==0) duration=TIME_OUT;  
cm = microsecondsToCentimeters(duration);  
break;
```

```
}
```

```
return cm;
```

```
}
```

```
//===== MOTORES =====
```

```
void motores(char d,char i)
```

```
{
```

```
int velder,velizq;
```

```
velder=map(d,100,-89,1150,1850);
```

```
velizq=map(i,-89,100,1150,1850);
```

```
MotorDer.writeMicroseconds(velder);
```

```
MotorIzq.writeMicroseconds(velizq);
```

```
/*
```

```
Serial.print(" VelDer:");
```

```
Serial.print(velder);
```

```
Serial.print(" VelIzq:");
```

```
Serial.print(velizq);
```

```
Serial.println();
```

```
*/
```

```
}
```

```
//=====
```

```
long microsecondsToCentimeters(long microseconds)
```

```
{
```

Competencia de Robótica “**MEGAMAN**”. Facultad de Ingeniería
Universidad de Buenos Aires – 8 de Junio de 2013

```
// The speed of sound is 340 m/s or 29 microseconds per centimeter.  
// The ping travels out and back, so to find the distance of the  
// object we take half of the distance travelled.  
return microseconds / 29 / 2;  
}
```

```
void probarluz()  
{  
int sd,si;  
  
sd= analogRead(SensorD);  
si= analogRead(SensorI);  
  
Serial.print("SensI: ");  
Serial.print(si);  
Serial.print("  SensD: ");  
Serial.print(sd);  
Serial.println();  
delay(50);  
  
}
```

```
void Calibrar()  
{  
int sd,si;  
int i;  
boolean estado;  
  
maxnegro=0;  
minblanco=1023;  
estado=false;  
  
motores (-40,40);  
  
for (i=0;i<125;i++)  
{  
sd= analogRead(SensorD);  
si= analogRead(SensorI);  
  
if (sd>maxnegro) maxnegro=sd;  
if (si>maxnegro) maxnegro=si;  
if (si<minblanco) minblanco=si;  
if (sd<minblanco) minblanco=sd;  
  
if (estado==false)  
{estado=true; digitalWrite(Led,HIGH);}  
else  
{estado=false; digitalWrite(Led,LOW);}  
delay(10);  
}  
//motores (0,0);  
//delay (100);  
motores (40,-40);  
for (i=0;i<250;i++)
```

Competencia de Robótica “**MEGAMAN**”. Facultad de Ingeniería
Universidad de Buenos Aires – 8 de Junio de 2013

```
{
sd= analogRead(SensorD);
si= analogRead(SensorI);

if (sd>maxnegro) maxnegro=sd;
if (si>maxnegro) maxnegro=si;
if (si<minblanco) minblanco=si;
if (sd<minblanco) minblanco=sd;

if (estado==false)
  {estado=true; digitalWrite(Led,HIGH);}
else
  {estado=false; digitalWrite(Led,LOW);}
  delay(10);
}
//motores (0,0);
//delay (100);
motores (-40,40);
for (i=0;i<125;i++)
{
sd= analogRead(SensorD);
si= analogRead(SensorI);

if (sd>maxnegro) maxnegro=sd;
if (si>maxnegro) maxnegro=si;
if (si<minblanco) minblanco=si;
if (sd<minblanco) minblanco=sd;

if (estado==false)
  {estado=true; digitalWrite(Led,HIGH);}
else
  {estado=false; digitalWrite(Led,LOW);}
  delay(10);
}

motores(0,0);

LimBlanco=((maxnegro-minblanco)*4)/10;

LimNegro=((maxnegro-minblanco)*6)/10;

/*
Serial.println();
Serial.print("MIN: ");
Serial.print(minblanco);
Serial.print(" MAX: ");
Serial.print(maxnegro);
Serial.println();
*/
}
```