

//Este ejemplo es un programa para arduino que utiliza la librería de arduino
//que aparece A continuación, y que está diseñada para la utilización de un
// servo motor. Además utiliza un sensor de ultrasonidos PING)))™.

#include <Servo.h>

//En esta ocasión utilizamos la placa Arduino AT168 Duemilanove la cual es
//suficiente para lo que necesitamos; es decir para el uso de dos motores DC,
//un servo motor y un sensor de ultrasonidos.

//Nuestra intención es que el robot se desplace por la estancia sin ningún
//recorrido marcado
//sin chocarse con ningún elemento de la misma:

//El primer bloque del programa va a consistir en un bucle a modo de
//Temporizador en el que está contenido todo el resto del programa, y que cada
//cierto tiempo se pare y compruebe que dirección seguir.
//Para la segunda fase del programa vamos a relizar el mismo sistema pero que
//se activará cuando la distancia a un objeto sea menor a un número de
//centímetros. Cuando se encuentre ese obstáculo realizará una comprobación a
//Cada lado el sensor de ultrasonidos, y decidirá continuar en la dirección en
//la que haya más espacio libre.

//Definimos los pines motores que van a controlar la velocidad y la dirección

int dirA = 10;
int dirB = 9;
int speedA = 12;
int speedB = 11;

//Definimos el valor del pin de control sobre el ping del sensor de
//ultrasonidos

const int pingPin = 3;

Servo myservo; //creamos un objeto servo para controlar el servo motor
//un máximo de 8 objetos servo se pueden crear

//Ahora iniciamos la parte de configuración

void setup()
{

Serial.begin(9600); // establecemos la librería Serie a 9600 bps
Serial.println("No Me Txoko!"); // escribimos en pantalla ese texto
pinMode (dirA, OUTPUT); // ponemos el pin de la dirección A como salida
pinMode (dirB, OUTPUT); // ponemos el pin de la dirección B como salida
pinMode (speedA, OUTPUT); // ponemos el pin de la velocidad A como salida
pinMode (speedB, OUTPUT); // ponemos el pin de la velocidad B como salida
myservo.attach(2); // vinculamos el servo del pin 2 con el objeto servo
}

//Tal y como hemos explicado en la introducción pretendemos que realice varias
//acciones, como son la de moverse con los motores DC, comprobar la distancia
//que existe por delante del robot con el sensor de ultrasonidos continuamente
//Pararse en caso de encontrarse un
//obstáculo, girar el servo motor a izquierda y derecha para que el sensor de
//ultrasonidos compruebe la distancia a cada lado, decidir a que lado ir, y
//por último pararse cada cierto tiempo y realizar comprobación de distancia a
//ambos lados nuevamente.

```

void loop()
{

// Envolvemos todo nuestro programa con este bucle for para que se repita cada
//1000 iteraciones para que después realice la comprobación de distancia a
//ambos lados
for(int i=0;i=1000;i++)
{

    long duration, cm;    // definimos la variable que va a estar en centímetros

    // Vamos a lanzar un pulso por el sensor de ultrasonidos para después
    // recogerlo y así poder calcular la distancia que existe entre el sensor y
    // lo que tiene delante (max. 3m)
    pinMode(pingPin, OUTPUT); // ponemos el pin de control del pulso del
    // sensor como salida
    digitalWrite(pingPin, LOW); // mandamos el comienzo de nivel bajo del
    // pulso de 2ms

    delayMicroseconds(2);
    digitalWrite(pingPin, HIGH); // mandamos la parte central de nivel alto del
    // pulso de 5ms

    delayMicroseconds(5);
    digitalWrite(pingPin, LOW); // mandamos el final de nivel bajo del pulso

    pinMode(pingPin, INPUT); // ponemos el pin del sensor como entrada

    duration = pulseIn(pingPin, HIGH); // calculamos el tiempo de vuelta
    // del pulso

    cm = microsecondsToCentimeters(duration); // convertimos a distancia
    // ese tiempo

    // Mostramos en pantalla el valor obtenido
    Serial.print(cm);
    Serial.print("cm");
    Serial.println();

    delay(100);

    if (cm < 50)
    {

        // Para los motores DC para realizar la comprobación de distancia a
        // ambos lados
        Serial.println(" parada");
        digitalWrite (speedA, 0);
        digitalWrite (speedB, 0);

        delay(100); // mantiene el motor parado durante 100ms

        // PARTE 1: LADO IZQUIERDO
        // Movemos el servo motor para la izquierda
        Serial.print(" Servolzquierda");
        myservo.write(145); // movemos el motor servo hacia la izquierda
        delay(1500); // esperamos 1500ms para que el servo alcance la posición

        // Ahora vamos a comprobar la distancia que hay a la izquierda del robot
        // y la vamos a guardar de igual manera que hemos explicado anteriormente
        // para tomar distancia con el sensor de ultrasonidos
        long durationLeft, cmLeft; // definimos la variable que va a estar
        // en centímetros

        pinMode(pingPin, OUTPUT); // ponemos el pin del pulso del sensor

```

```

// como salida
digitalWrite(pingPin, LOW); // mandamos el comienzo de nivel bajo del
// pulso de 2ms

delayMicroseconds(2);
digitalWrite(pingPin, HIGH); // mandamos la parte central de nivel alto
// del pulso de 5ms

delayMicroseconds(5);
digitalWrite(pingPin, LOW); // mandamos el final de nivel bajo del pulso

pinMode(pingPin, INPUT); // ponemos el pin del sensor como entrada

durationLeft = pulseIn(pingPin, HIGH); // calculamos el tiempo de
// vuelta del pulso

cmLeft = microsecondsToCentimeters(durationLeft); // convertimos a
// distancia ese tiempo

// Mostramos en pantalla el valor obtenido
Serial.print(cmLeft);
Serial.print("cmLeft");
Serial.println();

delay(100);

// PARTE 2: LADO DERECHO
// Movemos el servo motor para la derecha
Serial.print(" ServoDerecha");
myservo.write(35); // movemos el motor servo hacia la derecha
delay(1500); // esperamos 1500ms para que el servo alcance la
// posición

// Ahora vamos a comprobar la distancia que hay a la derecha del robot y
// la vamos a guardar de igual manera que hemos explicado anteriormente para
// tomar distancia con el sensor de ultrasonidos
long durationRigth, cmRigth; // definimos la variable que va a estar
// en centímetros

pinMode(pingPin, OUTPUT); // ponemos el pin del pulso del sensor
// como salida
digitalWrite(pingPin, LOW); // mandamos el comienzo de nivel bajo del
// pulso de 2ms

delayMicroseconds(2);
digitalWrite(pingPin, HIGH); // mandamos la parte central de nivel alto
// del pulso de 5ms

delayMicroseconds(5);
digitalWrite(pingPin, LOW); // mandamos el final de nivel bajo del pulso

pinMode(pingPin, INPUT); // ponemos el pin del sensor como entrada

durationRigth = pulseIn(pingPin, HIGH); // calculamos el tiempo de
// vuelta del pulso

cmRigth = microsecondsToCentimeters(durationRigth); // convertimos a
// distancia ese tiempo

// Mostramos en pantalla el valor obtenido
Serial.print(cmLeft);
Serial.print("cmLeft");
Serial.println();

delay(100);

```

```

// PARTE 3: VOLVER AL CENTRO
// Movemos el servo motor al centro para que vuelva a poder tomar medidas
// por delante
Serial.println(" ServoCentro");
myservo.write(85); // movemos el motor servo hacia la posición central
delay(1500); // esperamos 1500ms para que el servo alcance la posición

delay(100);

// Ahora vamos a decidir según que distancia es la mayor hacia
// que lado continuar
// GIRO IZQUIERDA
if (cmRigth < cmLeft)
{
    // motor A hacia delante y el B hacia atrás - gira a la izquierda
    Serial.print(" izquierda");
    digitalWrite (dirA, LOW);
    digitalWrite (dirB, HIGH);
    digitalWrite (speedA, 135);
    digitalWrite (speedB, 135);

    delay(500);
}

// GIRO DERECHA
if (cmRigth >= cmLeft)
{
    // motor A hacia atrás y el B hacia delante - gira a la derecha
    Serial.print(" derecha");
    digitalWrite (dirA, HIGH);
    digitalWrite (dirB, LOW);
    digitalWrite (speedA, 135);
    digitalWrite (speedB, 135);

    delay(500);
}
}

if (cm > 50)
{
    // movemos ambos motores en la misma dirección para que siga
    // para adelante
    Serial.print(" arranca");
    digitalWrite (dirA, HIGH);
    digitalWrite (dirB, HIGH);
    digitalWrite (speedA, 135);
    digitalWrite (speedB, 135);

    delay(100);
}
}

// Ahora repetimos la parte del programa que recoge los valores de distancia
// a cada lado del robot, para así poder decidir hacia que lado debe
// continuar. Esta parte es para cada cierto tiempo se pare y mire a ambos
// lados para medir la distancia, y así de esta manera evitar que el robot
// se quede muy cerca de una pared (casi en paralelo) y no la detecte por no
// tener enfrente el obstáculo

```

```

// Paramos ambos motores
Serial.println(" parada2");
digitalWrite (speedA, 0);
digitalWrite (speedB, 0);

delay(100); //mantenemos los motores parados durante 100ms antes de continuar

// PARTE 1: LADO IZQUIERDO
// Movemos el servo motor para la izquierda
Serial.print(" Servolzquierda2");
myservo.write(145); // movemos el motor servo hacia la izquierda
delay(1500); // esperamos 1500ms para que el servo alcance la posición

// Tomamos la distancia a la izquierda del robot
long durationLeft, cmLeft;

pinMode(pingPin, OUTPUT);
digitalWrite(pingPin, LOW);
delayMicroseconds(2);
digitalWrite(pingPin, HIGH);
delayMicroseconds(5);
digitalWrite(pingPin, LOW);

pinMode(pingPin, INPUT);
durationLeft = pulseIn(pingPin, HIGH);

// Convertimos a distancia
cmLeft = microsecondsToCentimeters(durationLeft);

Serial.print(cmLeft);
Serial.print("cmLeft");
Serial.println();

delay(100);

// PARTE 2: LADO DERECHO
// Movemos el servo motor para la derecha
Serial.print(" ServoDerecha2");
myservo.write(35); // movemos el motor servo hacia la derecha
delay(1500); // esperamos 1500ms para que el servo alcance la posición

// Tomamos la distancia a la derecha del robot
long durationRigth, cmRigth;

pinMode(pingPin, OUTPUT);
digitalWrite(pingPin, LOW);
delayMicroseconds(2);
digitalWrite(pingPin, HIGH);
delayMicroseconds(5);
digitalWrite(pingPin, LOW);

pinMode(pingPin, INPUT);
durationRigth = pulseIn(pingPin, HIGH);

// Convertimos a distancia
cmRigth = microsecondsToCentimeters(durationRigth);

Serial.print(cmRigth);
Serial.print("cmRigth");
Serial.println();

delay(100);

```

```

// PARTE 3: VOLVER AL CENTRO
// Movemos el servo motor al centro para que vuelva a poder tomar medidas
// por delante
Serial.println(" ServoCentro2");
myservo.write(85); // movemos el motor servo hacia la posición central
delay(1500); // esperamos 1500ms para que el servo alcance la posición

delay(100);

if (cmRigth < cmLeft)
{
    // motor A hacia delante y el B hacia atrás - gira a la izquierda
    Serial.print(" izquierda2");
    digitalWrite (dirA, LOW);
    digitalWrite (dirB, HIGH);
    digitalWrite (speedA, 135);
    digitalWrite (speedB, 135);

    delay(500);
}

if (cmRigth >= cmLeft)
{
    // motor A hacia atrás y el B hacia delante - gira a la derecha
    Serial.print(" derecha2");
    digitalWrite (dirA, HIGH);
    digitalWrite (dirB, LOW);
    digitalWrite (speedA, 135);
    digitalWrite (speedB, 135);

    delay(500);
}

// movemos ambos motores en la misma dirección para que siga para adelante
Serial.print(" arranca2");
digitalWrite (dirA, HIGH);
digitalWrite (dirB, HIGH);
digitalWrite (speedA, 135);
digitalWrite (speedB, 135);

delay(100); // mantener los motores girando durante 100ms

}

//Función que convierte un valor en microsegundos a un valor en centímetros
long microsecondsToCentimeters(long microseconds)
{
    return microseconds / 29 / 2;
}

```