
Laboratorio de Construcción de Robots Móviles

Practica No. 4

Operación de un robot móvil usando el microcontrolador Arduino

Laboratorio de Bio-Robótica, FI-UNAM

Objetivo: Controlar la operación del robot móvil usando máquinas de estados.

Desarrollo: Para cada uno de los siguientes apartados, realizar los diseños electrónicos y programas que se piden.

Duración: Tres semanas.

1. Construya el robot con el kit proporcionado, colocando dos motores de corriente directa, el portador de batería, etapa de potencia, el microcontrolador Arduino y dos sensores de contacto.

2. Configure puertos de entrada del Arduino y conecte dos sensores de contacto a dos pines de esté. Muestre el valor leído de los sensores en una terminal del IDE del Arduino como se hizo en la practica 2.

3. Escriba una función en C de Arduino que regrese el valor del sensor indicado. La función deberá tener el siguiente formato:

```
float show_sensor(char *sensor, int num_sensor)
{
    float x;
    ...
    return(x);
}
```

4. Configure puertos de salida del Arduino para controlar la etapa de potencia de dos motores como se hizo en la practica 3.

5. Escriba una función en C del Arduino que controle los movimientos de un robot móvil, en está se indicará el ángulo, en radianes, que el robot primero girará y la distancia que después avanzara. La función deberá tener el siguiente formato:

```
move_robot(float distancia, float angulo)
{
}
}
```

Para esta practica la forma de calcular el tiempo que deberán estar prendidos los motores es utilizando la función C de Arduino delay(), en las siguientes practicas se utilizarán los encoders en las llantas del robot para tener mayor precisión.

6. Modifique el programa en C de Arduino que se encuentra en el apéndice, que ejecuta el algoritmo de un robot móvil que evade obstáculos, como el que se muestra en la figura 1. En la figura 2 se muestra el algoritmo de un robot móvil que evade obstáculos, cuando los sensores de contacto S_i y S_d sienten un obstáculo sus valores son igual a uno, en caso contrario son cero.

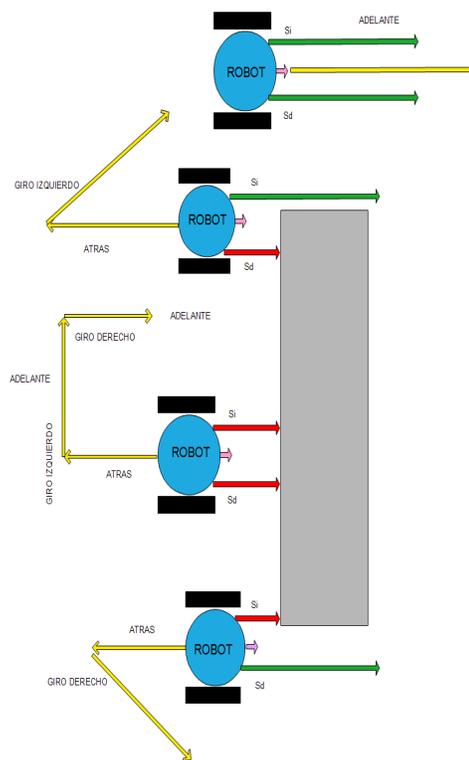


Figura 1. Robot Móvil que evade obstáculos

Apéndice

Programa en C de Arduino que ejecuta el algoritmo de un robot móvil que evade obstáculos, evade.ino

```
/******  
*           Algoritmo de Evasión de Obstaculos           *  
*           Programa en C de Arduino para un robot móvil que evade           *  
*           obstáculos.                                   *  
*                               Lab. Bio-Robótica           *  
*                               FI-UNAM                     *  
*                               *                           *  
******/  
  
// Definición de constantes  
  
#define ADELANTE move_robot(AVANCE, 0.0f)  
#define ATRAS move_robot(-AVANCE, 0.0)  
#define GIRO_IZQ move_robot(0.0, GIRO)  
#define GIRO_DER move_robot(0.0, -GIRO)  
#define ALTO move_robot(0.0,0.0)  
  
// Configuración de los pines de entrada y salida  
void setup() {  
  int i;  
  
  Serial.begin(9600); // Inicializa comunicación serial  
  
  //Configuración de los motores  
  
  // ...  
  
  //Configuración de los sensores de entrada  
  
  // ...  
}  
  
// Esta función hace que el robot primero gire un angulo y después avance una distancia  
void move_robot(float distancia, float angulo)  
{  
  
  // ....  
  
  delay(1000);  
  
}  
  
// Esta función lee el valor de un sensor, indicando su tipo y numero, y regresa su valor  
float shs(char *sensor, int num_sensor)  
{  
  float x=1.0;  
  
  // ....  
  
  return(x);  
}
```

```

// Algoritmo de evasión de obstáculos
void loop(){

int estado = 0;
float Si,Sd;
float AVANCE=.1;
float GIRO= 0.7854;

// Estado inicial
estado = 0;

// Loop infinito
while(1){

// Se leen los sensores
Si = shs("contact",1);
Sd = shs("contact",2);

Serial.print("Estado Presente: ");
Serial.println(estado);
Serial.print("Si: ");
Serial.println(Si);
Serial.print("Sd: ");
Serial.println(Sd);

switch (estado){

case 0: // est0
if (Si == 0){
if (Sd == 0){
estado = 0;
ADELANTE;
Serial.println("ADELANTE");
}
else{
estado = 1;
ALTO;
Serial.println("ALTO");
}
}
else{
if (Sd == 0){
estado = 3;
ALTO;
Serial.println("ALTO");
}
else{
estado = 5;
ALTO;
Serial.println("ALTO");
}
}
break;

case 1: // est1
estado = 2;
ATRAS;
Serial.println("ATRAS");
break;

```

```
case 2: // est2
        estado = 0;
        GIRO_IZQ;
        Serial.println("GIRO_IZQ");
        break;

case 3: // est3
        estado = 4;
        ATRAS;
        Serial.println("ATRAS");
        break;

case 4: // est4
        estado = 0;
        GIRO_DER;
        Serial.println("GIRO_DER");
        break;

case 5: // est5
        estado = 6;
        ATRAS;
        Serial.println("ATRAS");
        break;

case 6: // est6
        estado = 7;
        GIRO_IZQ;
        Serial.println("GIRO_IZQ");
        break;

case 7: // est7
        estado = 8;
        GIRO_IZQ;
        Serial.println("GIRO_IZQ");
        break;

case 8: // est8
        estado = 9;
        ADELANTE;
        Serial.println("ADELANTE");
        break;

case 9: // est9
        estado = 10;
        ADELANTE;
        Serial.println("ADELANTE");
        break;

case 10: // est10
        estado = 11;
        GIRO_DER;
        Serial.println("GIRO_DER");
        break;

case 11: // est11
        estado = 0;
        GIRO_DER;
        Serial.println("GIRO_DER");
        break;

} // end case

} // end while

} // end Main (loop)
```