

INTELIGENCIA ARTIFICIAL APLICADA A ROBOTS

Practica 2 **Combinando Múltiples Comportamientos**



Jesús Pérez Limón
Andrés Jesús Díaz Santos
Alejandro Trujillo Caballero
Christian Ortega Machado
Cesar Antonio Enrique Ramírez

DESCRIPCIÓN DE LA TAREA

Crear un robot basado en comportamiento que se mueva por una habitación sin chocarse con los obstáculos y sea capaz de llegar a una meta mediante el seguimiento de un color en específico.

DESCRIPCIÓN DEL ROBOT

El robot que hemos usado en prácticas tiene los siguientes componentes:

- **Plataforma oruga con dos motores de tracción CC independientes.**
 - Chasis realizado en policarbonato
 - Cadenas del robot oruga: 5cm de ancho
 - Motores: 7,2V a 175rpm
- **PC Mini-ITX Intel Desktop Board DQ45EK**
- **Shield Potenciador de Motores:**
 - Voltios de entrada: 5V a 50mA
 - Voltios de salida: 15V
 - Permite 2 modos de control
 - Modo Analógico: 0V (max en un sentido), 2.5V (reposo) y 5V (max en el otro sentido)
 - Modo Analógico con dirección: 0V (derecha), 2.5V (recto) y 5V (izquierda)
 - Modo I2C (el usado en practica):

Dirección Registro	Nombre	L/E	Descripción
0	Modo	R/W	Modo de funcionamiento
1	Velocidad1	R/W	Velocidad Motor Izquierdo (modo 0,1) o Velocidad (modo 2,3)
2	Velocidad2/giro	R/W	Velocidad Motor Derecho(modos 0,1) o Giro (Modos 2,3)
3	Aceleración	R/W	Aceleración para I2C
7	Versión	Lectura	Número de Revisión del Software

- **Comunicación bus I2C-USB**
 - Responde a los comandos que se le envían desde el PC. Hay solo tres comandos primarios:
 - **I2CD_CMD (0x53)** - Es el que le permite leer o escribir a dispositivos que no tienen registrada una dirección interna como las I/O PCF8574 de la placa de expansión Philips.
 - **I2C_CMD (0x55)** - Es el que le permite leer o escribir a dispositivos con registros internos como el EEPROM's o nuestra propios módulos.
 - **USB-I2C_CMD (0x5A)** - Una serie de comandos enviados al módulo generalmente para comunicaciones seleccionadas.
 - Para hacer una escritura/lectura a través del I2C se deben pasar una secuencia de 3 bytes (Comando, Dirección del dispositivo y Datos)

- **Sensores de Distancia (Ultrasonido)**
 - Alimentación: 5V
 - Consumo medio: 4mA
 - Medidas: 24x20x17
 - Peso: 4,6g
 - Rango de medición: 15cm – 600cm
- **Sensor Brújula**
 - Alimentación: de 3.3V a 5V
 - Consumo: 25mA
 - Salida:
 - de 0 a 3599: Correspondencia directa a grados
 - de 0 a 255: Hay que hacer el 3599 / 255
- **Camara**
 - Modelo: Logitech Quickcamp Sphere/Orbit motorizada
 - Max. Photo Resolution: 1280x960
 - Max. Video Resolution: 640x480
 - Frame Rate 30fps
 - Connectivity USB 2.0

DESCRIPCIÓN DEL ENTORNO

- Habitación de 20x20m con cajas para simular los obstáculos y una meta.
- Se procuró minimizar el máximo ruido posible.
- Iluminación artificial.

DISEÑO

Para el diseño del comportamiento que tendrá el robot se ha decidido diseñar un sistema basado en estados que se ejecutará de la siguiente forma:

Inicialmente el robot en reposo realizará un barrido al entorno. Para ello realizará una vuelta sobre sí mismo capturando instantaneas, analizandolas por si detecta al objeto. Este barrido se repetirá tras un cierto numero de evaluaciones. En adelante para referirnos a esta vuelta y todo lo que implica usaremos el término “Barrido”.

Si al realizar un Barrido no se detecta el objeto, el robot realizará una lectura de los sensores de proximidad de los 3 sensores frontales para detectar posibles obstaculos en el camino y dependiendo del valor resultante decidirá si girará a la izquierda, a la derecha o avanzará hacia adelante. Al conjunto de acciones iterativas resultante de leer los sensores, decidir el movimiento a realizar y realizarlo le llamaremos “Avanzar” y será el comportamiento básico del robot.

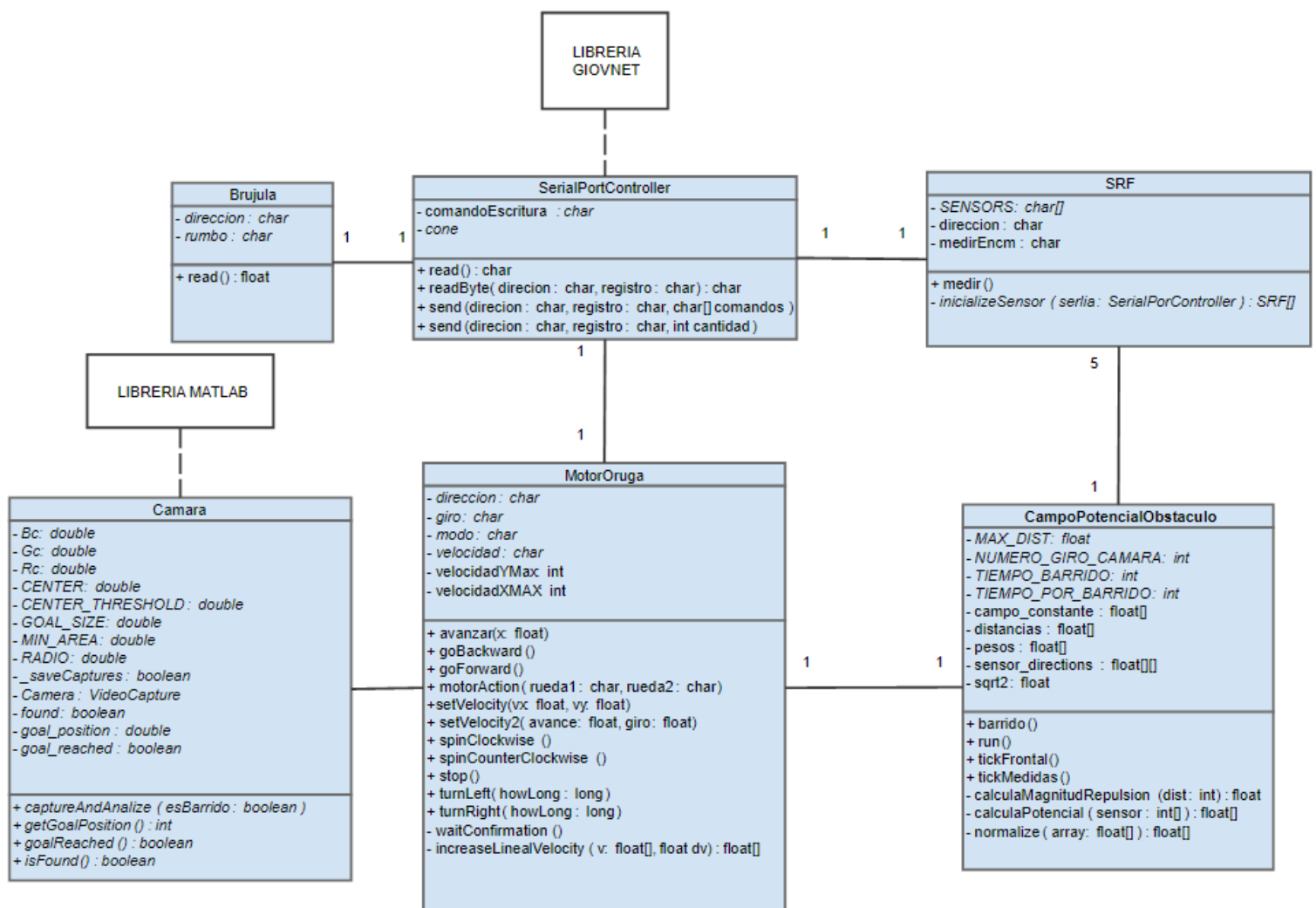
Si mientras se encuentra en el estado Avanzar, encuentra un obstaculo usará los 5 sensores para determinar hacia que lado girar. Tras esquivarlo, volverá a usar los 3 frontales.

Si al realizar un Barrido se detecta el objeto el robot tratara de centrar el objeto en el ancho de su campo de visión (En adelante “Centrado”) y avanzará hacia adelante (Avanzar). Si el objeto está localizado no se realizarán más barridos.

Si estando el objeto Centrado y el robot avazando hacia él la meta se pierde por algún motivo (Por ejemplo, sale del campo de visión al esquivar un obstaculo), el robot realizará un Barrido para intentar volverlo a localizar. En caso positivo vuelve a centrarlo, en caso negativo vuelve al comportamiento básico.

Si el robot ha alcanzado la meta (Ha llegado al objeto a buscar), realizará un par de vueltas sobre sí mismo en señal de victoria y terminará la ejecución de su codigo.

IMPLEMENTACION



DIFICULTADES ENCONTRADAS

- Los sensores ultrasonido al estar tan juntos había veces que se falseaban las medidas, un sensor cogía la señal rebote de otro y marcaba una medida mucho menor.
- La brújula se descalibraba y no daba una salida buena, por lo que se tuvo que tomar otro sistema de referencia a la hora de girar. Este sistema era el tiempo en que la rueda giraba, pero esto no es siempre exacto ya que depende de las medidas.
- El color que se seleccionó para hacer de meta es fácilmente confundible con el amarillo, por tanto, el robot a veces se confundía e iba hacia donde predomina mas el amarillo.