

Documento técnico funcional

*Sumo de robots
Robotgroup Multiplo N6 Max
Expocarreras 2016*



Integrantes:

WIEILLY, Alan
MACULA, Alejandro
AMARO, Marcos
DIAZ BURCET, Nehuén

Sistemas Embebidos

Licenciatura en Sistemas

2016

CONTENIDO

I. INTRODUCCIÓN

A. Sumo de robots

II. ENTORNO DE DESARROLLO

A. Requisitos

B. Instalación de drivers

C. Entorno de desarrollo Arduino

D. Monitor serial

III. FUNCIONES PRINCIPALES DESARROLLADAS

A. Move

B. Atacar

C. Defenderse

D. Aumentar o disminuir velocidad

E. Otras funciones

F. Comportamiento del robot (led RGB)

I. INTRODUCCIÓN

En el presente documento se detallarán los principios básicos del Sumo de robots y la especificación técnica funcional del desarrollo realizado para la Expocarreras 2016. Se abarcarán tanto las reglas del juego, los algoritmos principales programados por el grupo y una guía con los pasos necesarios para preparar el ambiente de desarrollo del robot Múltiplo N6 Max utilizado para la exposición.

A. Sumo de robots

El sumo de robots es una competencia en donde se enfrentan dos robots en una pista, el objetivo es sacar completamente de la pista al robot adversario.

La pista es una circunferencia, donde en la parte central existen dos franjas grises, que es desde los robots parten en el inicio de la competencia. También cuenta con una franja exterior blanca para indicar al robot que está fuera, y el interior de la pista es negra, que es donde se desarrolla la competencia.

Se deben esperar al menos 5 segundos para iniciar el primer movimiento.

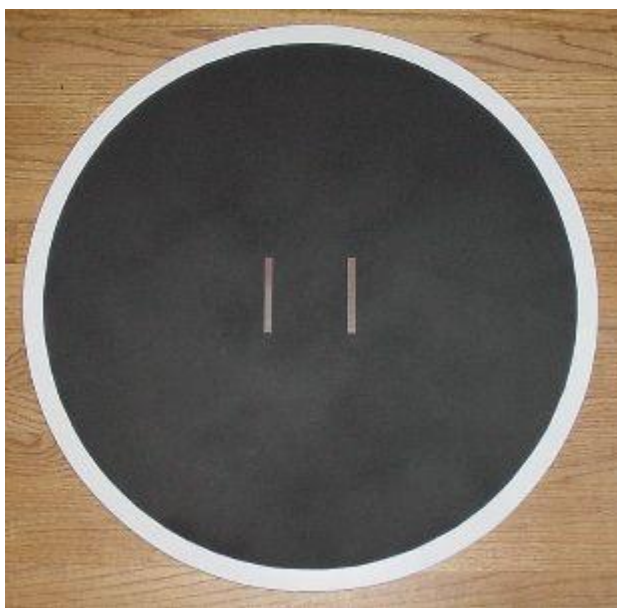


Figura 1: pista del sumo de robots.

II. ENTORNO DE DESARROLLO

A. Requisitos

Se debe contar con una computadora con un sistema operativo Windows XP o versión superior. El fabricante también provee drivers e IDE compatibles con MAC OS X.

El robot opera con 3 pilas AA y usa una conexión mediante un cable mini USB para la entrada al robot y estándar del extremo a la computadora

B. Instalación de drivers

Para empezar a desarrollar con el robot, es necesario primero instalarse los drivers provistos por Robotgroup dentro del DuinoPack del siguiente enlace: <http://www.robotgroup.com.ar/archivos/software/Duinopack-v1-3-win.zip>

Una vez descargados, descomprimir la carpeta.

Antes de encender el robot, hay que asegurarse que tenga un jumper conectado en las patitas para CDC, tal como se muestra en la figura 2, en caso contrario, no se podrá subir el código al robot.

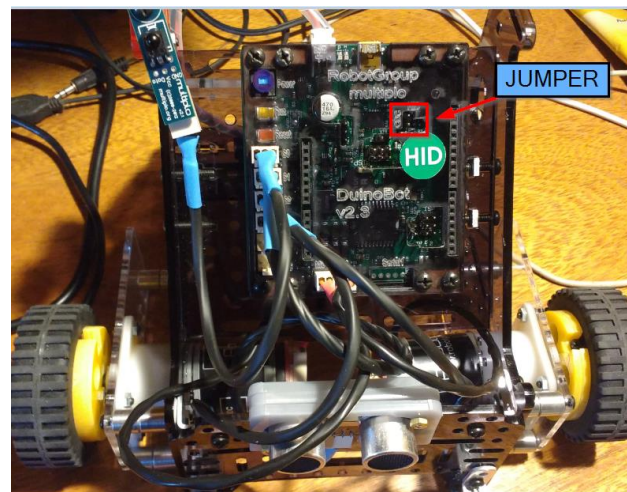


Figura 2: conexión de jumper a patas CDC

Conectamos el cable mini usb al robot y el otro al conector a la computadora.

Una vez realizado el paso anterior, esperamos que el sistema operativo busque el driver del robot y lo instale. Es posible que el sistema instale los drivers equivocados en un sistema Windows, y quede en un estado en que no pueda usarse, como se ve en la figura 3.

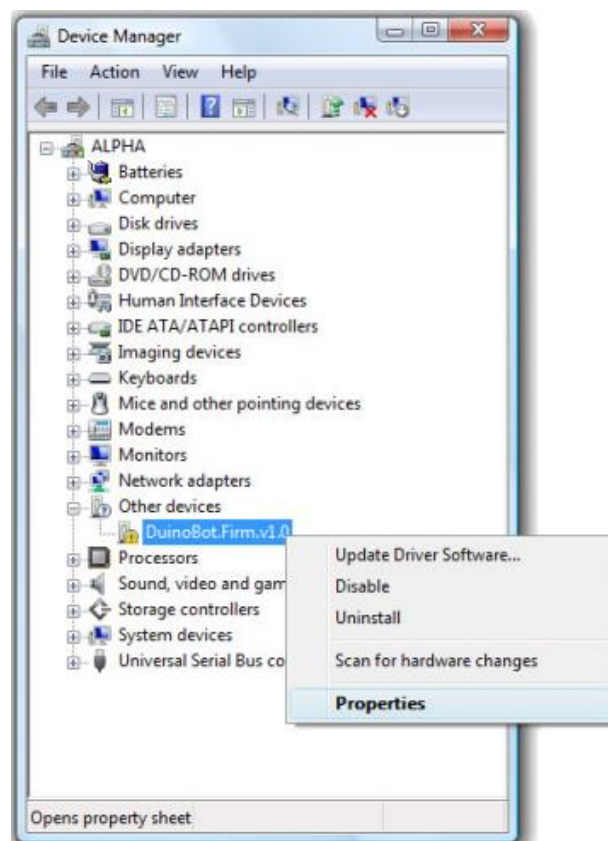


Figura 3: driver instalado incorrectamente

Para solucionar este problema, vamos al administrador de dispositivos -> Otros dispositivos, click derecho y “Actualizar controlador”. Elegir la instalación manual y buscar dentro del DuinoPack descargado anteriormente, la carpeta “Drivers”.

Importante: es posible que en sistemas Windows 8 en adelante no deje instalar los drivers debido a que no se encuentran firmados, para esto, reiniciar el sistema con opciones de arranque que permitan instalar drivers no firmados.

C. Entorno de desarrollo Arduino

Dentro del paquete DuinoPack descargado anteriormente, se encuentra el entorno de desarrollo Arduino. Para empezar a usarlo, tan solo hace falta ejecutar el archivo “arduino.exe”.

El IDE trae varios ejemplos, como estado de la batería del robot, manejo del control remoto, etc. Para acceder a ellos nos dirigimos a Archivo -> Múltiple -> Ejemplos.

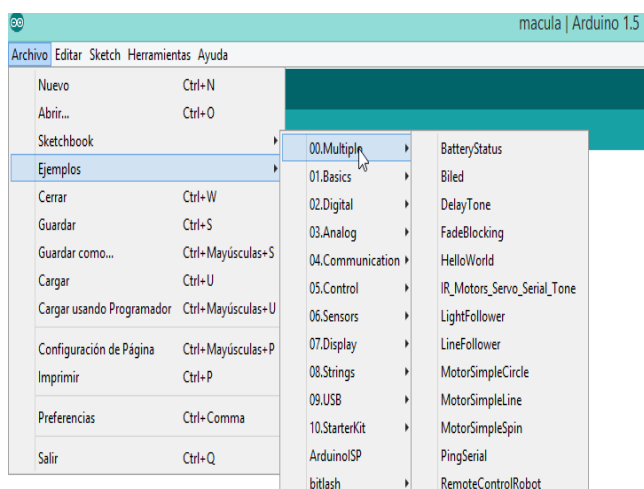


Figura 4: Ejemplos para Múltiple N6

Para poder cargar el programa tenemos que configurarlo con la placa que posee el robot. Para ello nos dirigimos a Herramientas -> Tarjeta, y elegimos la opción que dice Duinobot.v2.3.CDC W/Atmega 1284.

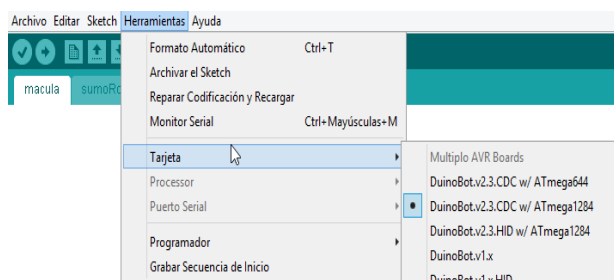


Figura 5: Elección de tarjeta

Importante: hay dos funciones principales que deben existir, la primera es **void setup()** y la segunda **void loop()**

1. **setup:** en esta función se pueden inicializar variables, setear valores, etc. al comenzar el programa, tales como los baudios a utilizar.
2. **loop:** está función, tal lo indica su nombre, es el loop en el cual el programa ejecutará las rutinas que le indiquemos, por ejemplo: hacer avanzar el robot, leer el estado del control remoto, etc. Esta función se ejecuta reiteradamente.

Luego de seguir los pasos anteriores y haber conectado el robot, vamos a Herramientas -> Puerto Serial, y nos fijamos que el puerto este tildado, en caso de que no, lo tildan.

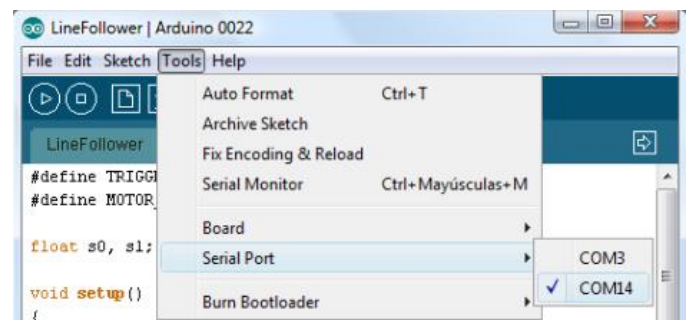


Figura 6: Elección del puerto serial

Para verificar que el código escrito compile correctamente, presionar el botón “Verificar”. En caso que tenga algún error, lo mostrará en la consola inferior.

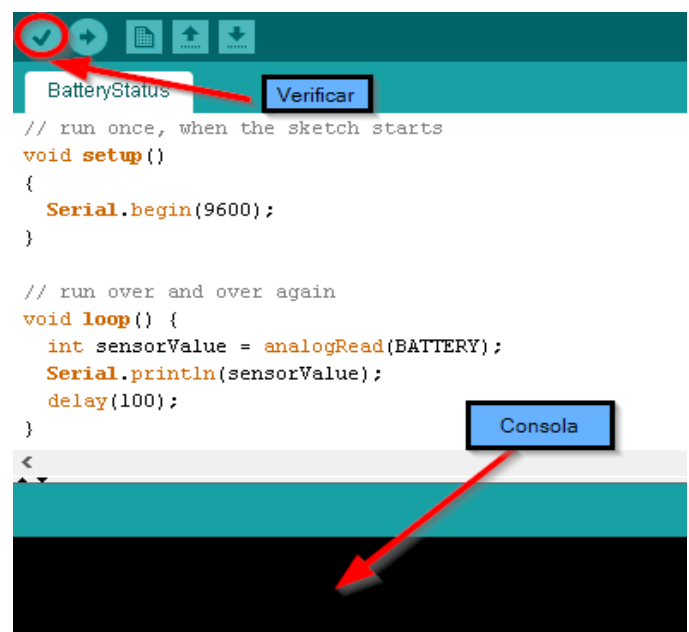


Figura 7: Verificar código

Una vez que siguieron todos pasos, elegimos la opción cargar y esperamos a que termine.

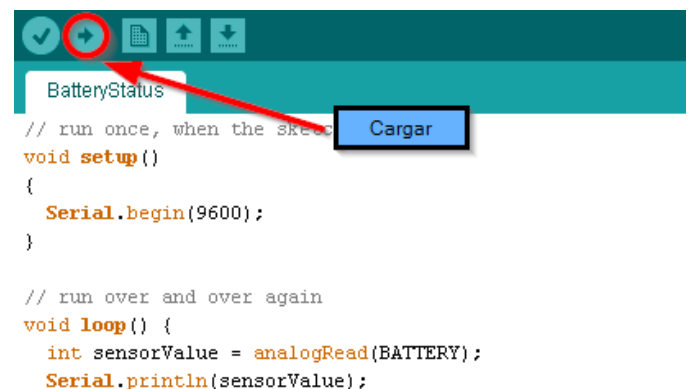


Figura 8: Cargar el código en el robot

D. Monitor serial

Dentro del entorno de desarrollo Arduino, también es posible imprimir líneas hacia el monitor que trae integrado. Esto es muy útil para depurar código, ver los valores que devuelven los sensores, etc.

Para esto, hay que cargar el código al robot introduciendo lo que se quiere imprimir con la sentencia `Serial.println("texto o variable");`

Una vez cargado el código, ir a Herramientas -> Monitor Serial. Luego de este paso, es necesario presionar el botón "Run" del robot para que empiece a correr el código cargado dentro de él.

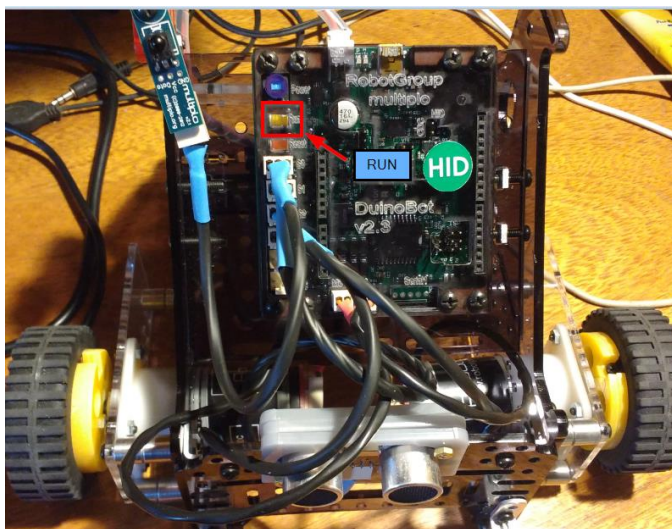


Figura 9: botón Run

III. FUNCIONES PRINCIPALES DESARROLLADAS

A. Moverse

Lo que hace esta función es recorrer la pista sin salirse del borde blanco, esto se logra con los tres sensores infrarrojos que están instalados en el robot, que miden el color cada vez que hace algún movimiento, como por ejemplo, avanzar, retroceder o girar (tiene tres giros 90, 180 y 360 grados), para que no se vaya de la misma. El botón del control remoto que permite elegir esta opción es el número 9.

B. Atacar

En este algoritmo se busca un objetivo para sacarlo de la pista, esto se logra mediante el sensor de ultrasonido instalado en el robot, que mide la distancia a la que se encuentra el adversario. La distancia máxima de cercanía entre ellos es de 3000 (cuando el oponente se encuentra pegado al sensor del robot que está atacando), la mínima indicada por nosotros para el alcance de búsqueda es de 30. El botón del control remoto que permite elegir esta opción es el número 7.

C. Defenderse

El módulo de defensa consiste en escapar cada vez que encuentra un objetivo cerca a través de la medición hecha por parte del sensor de ultrasonido mencionado anteriormente en la función de atacar. El botón del control remoto que permite elegir esta opción es el número 8.

D. Aumentar o disminuir la velocidad

Por medio de esta función se logra aumentar o disminuir la velocidad del robot cambiando el valor de la velocidad de los dos motores que posee el mismo. Al robot le agregamos tres velocidades: baja, media y alta. La velocidad baja es de 60, la media de 75, y la alta de 90. El botón del control remoto que permite elegir estas opciones son: el número 1 para la velocidad alta, el 2 para la media, y el 3 para la baja.

E. Funciones auxiliares

A continuación se listan otras funciones auxiliares desarrolladas para dar soporte a las funciones principales expuestas anteriormente:

- ✓ Girar a la derecha y a la izquierda
- ✓ Girar 360 grados
- ✓ Girar 180 grados
- ✓ Girar 90 grados
- ✓ Leer sensor de la derecha, izquierda y atrás
- ✓ Objetivo cerca
- ✓ Frenar
- ✓ Sobre la línea blanca
- ✓ Girar hasta encontrar la dirección del objetivo

F. Comportamiento del robot (led RGB)

- Al iniciar, el robot espera 5 segundos antes de hacer el primer movimiento.
- Cuando **empieza a avanzar**, el led RGB toma color **blanco**.
- Mientras está **dentro de la pista y andando**, el led RGB del robot se muestra en color **verde**.
- Si el robot se encuentra **pisando la franja blanca**, el led RGB toma el color **rojo** hasta acomodarse dentro de la pista nuevamente.
- Dentro del **modo defensivo**, si detecta un **objetivo**, el led RGB se torna de color **verde + azul**.
- En el **modo de ataque**, cuando detecta al **objetivo**, el led cambia a color **rojo + azul**.

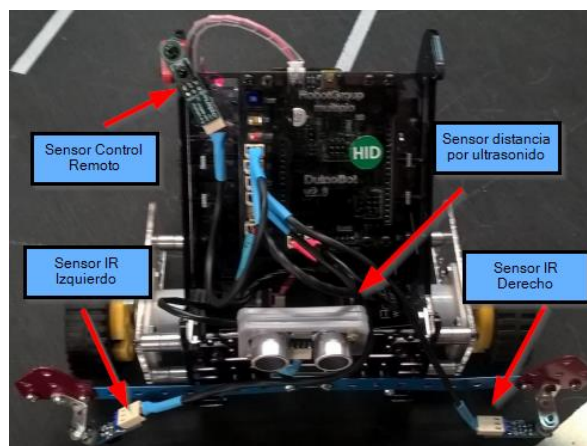


Figura 10: Sensores del robot utilizados