



# Madrid-bot



## MADRIDBOT 2007

### NEXUS 4 y NEXUS 5

### SOCCER PLAYERS

Nerea de la Riva Iriepa y Silvia Santano Guillén

[nereairiepa@gmail.com](mailto:nereairiepa@gmail.com) y [nani.complubot@gmail.com](mailto:nani.complubot@gmail.com)

[www.complubot.org](http://www.complubot.org)

#### Resumen

Nexus 4 y Nexus 5 son dos robots diseñados para poder participar en la prueba de SOCCER de la RoboCup Junior en la modalidad de 2+2. En esta prueba, dos equipos formados por dos robots cada uno (portero y delantero) tienen que jugar al fútbol (SOCCER) con una pelota especial (emisora de infrarrojos) y en un campo delimitado por paredes de 1,2 x 1,8 m (aprox) y cuyo suelo es un degradado de grises (del negro al blanco). Estos robots están limitados tanto de peso (2500gr. máximo) como en tamaño (tiene que entrar en un cilindro de 220mm de diámetro y una altura máxima de 220mm)



#### 1 Introducción

NEXUS es toda una familia de robots de SOCCER que parten de la primera plataforma experimental (NEXUS 3) y que han evolucionado a lo largo de las siguientes versiones (Nexus 4 y Nexus 5) añadiendo

nuevas prestaciones y mejoras.



Todos los robots de la serie NEXUS están basados en un plataforma omnidireccional, de dos o tres ruedas, que les hace tener una gran movilidad. La forma cilíndrica de pisos apilados es otra de las características comunes de estos robots que buscan siempre la máxima estabilidad y resistencia así como un centro de gravedad bajo y un peso "comedido".

#### 2 Arquitectura de los robots

Dispone de una unidad central de control que se comunica con un conjunto de módulos periféricos mediante un bus I2C.

Esta estructura nos permite añadir y/o cambiar módulos con relativa sencillez. Cada uno de los distintos módulos dispone de un pequeño procesador para las comunicaciones I2C.



## 3 Estructura de los robots

Nexus 4 y Nexus 5 no son dos robots idénticos. Esto es así por varios motivos:

- Nexus 4 está diseñado para jugar como portero y Nexus 5 como delantero.
- Las limitaciones de peso nos obligan a prescindir de algunos de los elementos en función del comportamiento de cada robot.
- Al ser la primera vez que hemos trabajado con “plataformas omnidireccionales” hemos preferido probar distintas soluciones.

Aún así Nexus 4 y Nexus 5 comparten muchos elementos tanto a nivel de Hardware como de Software lo que nos permite compartir experiencia y recursos (diseños, repuestos...) entre ellos.

### 3.1 Estructura mecánica

La estructura de pisos apilados nos permite distribuir los distintos elementos del robot en función de su funcionalidad, peso o mejor ubicación.

Cada una de estas plataformas está formada por un fino panel (1,8mm de espesor) de material compuesto del tipo carbono-epoxy. Las ventajas de este tipo de material para nuestra aplicación son las siguientes:

- Es ligero
- Muy rígido
- Muy resistente

Por otra parte están los inconvenientes:

- Es caro
- El proceso de fabricación es complejo
- Es difícil de mecanizar
- Es conductor

Esto último no nos afecta mucho pero es un inconveniente que hay que tener en cuenta y que también ocurriría con otros materiales como el aluminio.



Las placas de material compuesto las hemos fabricado nosotros mismos siguiendo el siguiente proceso:

1. En primer lugar hemos fabricado unas plantillas en madera con la forma deseada.
2. A continuación hemos cortado con la ayuda de un “cutter” las láminas de tejido de carbono pre-impregnado con la resina epoxy.
3. Las distintas láminas se han ido apilando hasta formar ocho capas compactando con un rodillo para eliminar todo el aire posible entre las mismas.
4. Una vez terminadas, nos las han procesado en un laboratorio que disponían del equipamiento necesario.

Dado que la resina epoxy de este material solo endurece a alta temperatura (unos 180 grados) hemos podido manipular el material sin problemas.



Cada una de las placas que forman los pisos tienen forma circular, esto es así por los siguientes motivos:

- Con esta forma es mucho más fácil que no se que atascado en las paredes del campo.



# Madrid-bot



- Es la mejor manera de utilizar todo el espacio permitido.

El “chaflán” en la placa inferior está pensado para que con dos palitos incorporados en esta zona, tengamos algo de control sobre la pelota (esto con un borde totalmente circular es imposible).

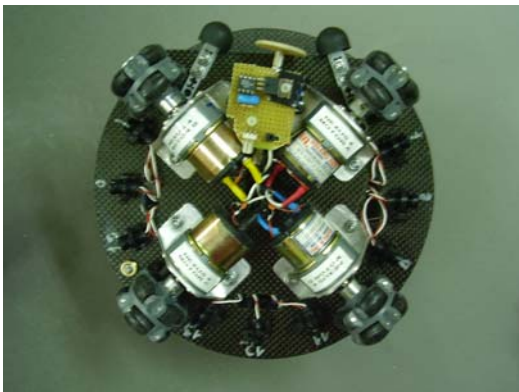
La distribución del peso está pensada para que estos robots no pierdan nunca el equilibrio (el peso se concentra abajo). La electrónica importante está en la parte superior ya que consideramos que esta es la parte que menos peligro corre. La brújula se encuentra lo más alejada posible del resto de dispositivos especialmente de los motores y el chutador ya que estos producen un campo electromagnético que le puede afectar.

La tornillería estructural es de acero de alta calidad pero toda la demás, debido a que no podemos exceder en el peso, es de nylon.

Dado que el robot está sujeto a golpes y vibraciones como parte de su funcionamiento normal, hemos usado tuercas auto-frenables para garantizar que no se puedan aflojar con facilidad.

## 3.2 Sistema de movimiento

Esta es la principal diferencia entre Nexus 4 y Nexus 5:



Nexus 4 está basado en una plataforma omnidireccional de tres ruedas motrices colocadas a 120 grados.



Nexus 5 está basado en una plataforma omnidireccional de 4 ruedas motrices colocadas a 90 grados. Nótese que las ruedas no son paralelas a la dirección de avance.

Las ruedas utilizadas son omnidireccionales de 4 rodillos.

Los componentes que integran el sistema de movimiento son:



- **Motores:** Son de corriente continua (DC) de 7,2 V de tensión nominal y unas 300 rpm. Son bastante pesados lo que nos ha dado algún problema en el caso de poner 4 motores.



- **Controladores de motores:** Hemos usado controladores dobles (pueden manejar dos motores cada uno), que se comunican vía I2C con la CPU. Cada robot lleva dos de estos controladores

## 3.3 Sistema de alimentación

Está compuesto por dos baterías de LI-PO de 2200 mA/h y con una capacidad de descarga de 12C, esto es, son capaces de darnos una intensidad de



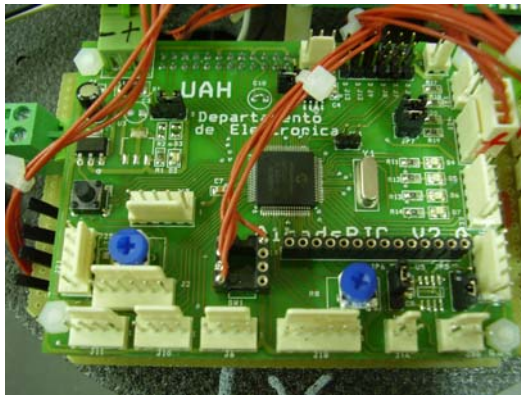
corriente de hasta  $2200 \times 12 = 26,4$  Amperios de forma sostenida.

Una de las baterías alimenta a la electrónica (cpu, sensores...) y la otra alimenta al sistema de potencia (motores y chutador).

Aunque en la actualidad ambas baterías son de 7,4 V, en la versión definitiva la batería que alimenta el sistema de potencia será de 11,1 V.

### 3.4 CPU

El microcontrolador que utilizamos en la CPU es un dsPIC 30F6010A, montado sobre una placa desarrollada por la UAH, llamada ALCADSPIC.



Sus principales características son:

- Arquitectura de 16 bits.
- Hasta 30 MIPS (millones de instrucciones por segundo) de velocidad de operación.
- Convertidor analógico digital de 10bits.
- Hasta 16 entradas analógicas.
- Controlador I2C.
- 5 contadores
- Una gran cantidad de entradas/salidas digitales.

### 3.5 Chutador

El chutador se usa para mover la pelota de una forma rápida dándole un golpe fuerte pero sin romperla.



El chutador es un electroimán de 12 V que trabaja con un "factor de servicio del 5%" (esto es, que para que nos proporcione más fuerza le damos una fuerte "excitación" durante un corto intervalo de tiempo y luego hay que esperar un tiempo a que se enfríe). Para que tenga efectividad le hemos colocado un disco circular de fibra de vidrio en el extremo para chutar la pelota.

Para activarlo, usamos una salida digital del dsPIC. Esta salida activa un multivibrador monoestable ajustable (montado con un 555) que mantiene activa la señal durante un tiempo de unos 200ms. Este tiempo se puede modificar mediante un potenciómetro de forma que el "chute" sea lo más eficaz posible en función del estado de las baterías o del tamaño de la pelota, sin necesidad de tocar el programa.

La salida del 555 activa un driver de potencia formado por un MOSFET IRF530 y un controlador MIC5011.

### 3.6 Sensores

Los robots de la serie NEXUS, por su función, llevan 4 tipos de sensores:

#### 3.6.1 Sensores de IR

Se necesitan para ver la pelota emisora e infrarrojos (roball).





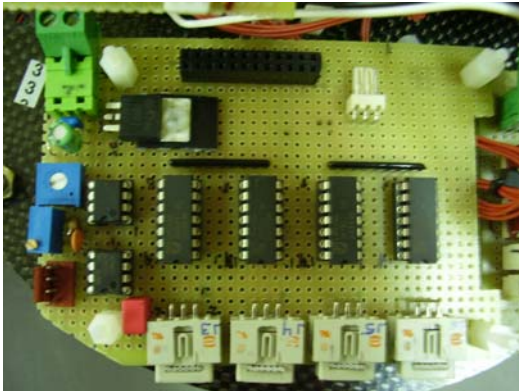
# Madrid-bot



Cada robot dispone de 16 fototransistores (SFH 313FA) colocados de forma radial en la base del robot. Para aumentar la direccionabilidad de los sensores, están protegidos con un “capuchón” de forma que sólo reciben la luz que les entra perpendicularmente.

Foto de detalle de los sensores

Cada uno de los fototransistores dispone de un amplificador y un filtro para eliminar la radiación IR que le pueda llegar del alumbrado (50 o 60 Hz).



Las 16 señales de los sensores se llevan a una placa donde se encuentra 16 comparadores que se encargan de determinar el umbral de visión de cada sensor. El nivel de comparación es único para que de esta forma resulte sencillo el ajuste.

Las salidas de los comparadores se conectan a 16 entradas digitales de la CPU. De esta forma conseguimos información de forma muy rápida sobre la localización de la pelota.

### 3.6.2 Sensores de ultrasonidos

Cada robot dispone de 4 sensores de ultrasonidos del tipo SRF08 que se comunican mediante bus I2C.



En la prueba de soccer, es muy recomendable usarlos para poder detectar las paredes del campo, al oponente o incluso al otro robot de tu equipo.

Tienen un alcance máximo de 11m, pero como estos sensores permiten variar su alcance los hemos ajustado en función del tamaño máximo del campo.

### 3.6.3 Brújula electrónica

Ambos robots utilizan una brújula electrónica del tipo CMPS03.



Esta brújula se conecta mediante bus I2C con la CPU. Tiene una precisión de unos 4 grados que resulta más que suficiente para esta aplicación.

La principal misión de esta brújula es orientar al robot adecuadamente para que marque gol en la portería contraria y no en la propia.

Antes de usar la brújula es necesario calibrarla. Esto se puede hacer mandando comandos mediante el bus I2C o directamente mediante un pequeño pulsador tal y como se muestra en la fotografía.

Esta brújula dispone de un sistema para corregir el efecto del campo generado por la red eléctrica. Para ello hay colocado un pequeño puente en función de la frecuencia de la misma : 50 Hz en Europa y 60 Hz en Japón y USA. Esto es



# Madrid-bot



especialmente importante para nosotros ya que nuestros robots tienen que competir tanto en Europa como en USA.

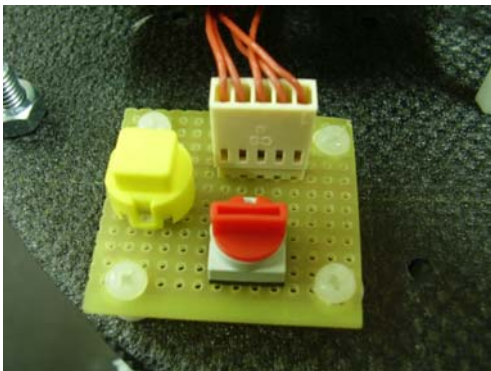
### 3.6.4 Sensor de suelo

Este sensor se encuentra en fase de desarrollo en este momento. Su misión será determinar la posición del robot en el campo en función del nivel de gris del suelo.

Para la realización de este sensor usaremos un emisor de luz y un fotodiodo con su respectivo amplificador. Estará conectado a una entrada analógica de la CPU.

### 3.7 Selector de programas

Está formado por un selector con codificación decimal y un pequeño pulsador denominado "GO":



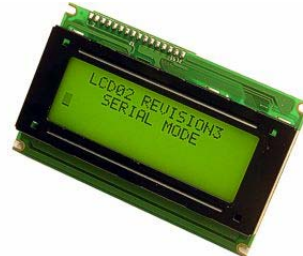
Este circuito se conecta a cuatro entradas digitales de la CPU y combinado con un pequeño programa de gestión podemos tener en la memoria del dsPIC hasta un total de 8 programas distintos que podemos ejecutar en función de la posición del selector.

De esta manera, además de tener el programa principal del robot, podemos acceder en todo momento a los programas de ajuste y verificación del robot sin necesidad de descargar programas nuevos.

Los 3 primeros bits son usados para seleccionar el número de programa y el cuarto controla un pulsador "GO" que hace que comience el programa que se encuentra seleccionado en ese momento.

### 3.8 Display y teclado

Ambos robots utilizan un display retro-iluminado (LCD03) de 4 filas de 20 caracteres.



Este display se conecta a la CPU mediante el bus I2C. Además dispone de una entrada para conectarle un teclado matricial que también se puede leer por el mismo bus I2C.



El conjunto Display + teclado constituyen el interface de comunicación entre el robot y el usuario. Se usa principalmente en los programas de ajuste o monitorización y nos permite visualizar información del estado del robot, de sus sensores e incluso actuar de forma manual los motores o el chutador.

## 4 Programación

La programación de los robots está realizada íntegramente en lenguaje C. Antes de programar microcontroladores hemos adquirido experiencia, a lo largo de dos años, programando el RCX de LEGO en NQC (un lenguaje con una sintaxis muy parecido al C, aunque un poco más limitado) y en BrickOS, un sistema operativo "empotrado" en el que se programa en C estándar.

### 4.1 Entorno de desarrollo

El Entorno de desarrollo utilizado es el MPLAB de microchip que junto con el compilador C30 nos permite programar el dsPIC en lenguaje C con toda comodidad.

### 4.2 Programador - depurador

Para descargar los programas en el dsPIC usamos, inicialmente, un pequeño programa "Bootloader"



# Madrid-bot



que nos permitía descargar los programas por el puerto serie del ordenador (o el USB con un adaptador). Este procedimiento resultaba algo engorroso al tener que cambiar continuamente de aplicación para compilar o descargar (hay que cerrarla y abrirla cada vez).



En la actualidad usamos un dispositivo de microchip llamado ICD2 que es un programador /depurador que nos permite descargar los programas desde el propio entorno MPLAB. E incluso depurar el funcionamiento del mismo, aunque esta característica no la hemos usado aún.

## 5 Programas

En realidad no podemos hablar de un único programa. Para cada robot hemos desarrollado un conjunto de programas de acuerdo con la siguiente estructura:

- Módulos básicos, son los que se encargan del funcionamiento del robot al más bajo nivel como las funciones de comunicación I2C, temporización, conversión A/D, manejo del display...
- Drivers de dispositivos. Son funciones básicas para hacer funcionar cada uno de los dispositivos. Cada uno de los drivers dispone de su propio programa de prueba.
- Módulos de prueba. Son programas para verificar el correcto funcionamiento del robot, por ejemplo, el movimiento, medidas de los sensores... Cada uno de estos módulos tiene su propio programa de prueba.

Mediante el uso del selector de programas y de un pequeño gestor, podemos tener instalados en el robot hasta un total de 8 programas distintos que nos permiten verificar el funcionamiento del robot o realizar los ajustes necesarios. Naturalmente, uno de los 8 programas será el programa principal.

## 6 Comportamientos

La principal misión del programa principal de cada robot es dotarle a este de un determinado comportamiento en función del tipo de robot (portero o delantero) y de la información recibida por los sensores.

Estos son los comportamientos que estamos implementando en los robots de la serie Nexus

- Localización de la pelota.
- Posicionamiento en función de sensores de ultrasonidos.
- Orientación mediante brújula
- Modo búsqueda cuando no ve la pelota.

## 7 Conclusiones

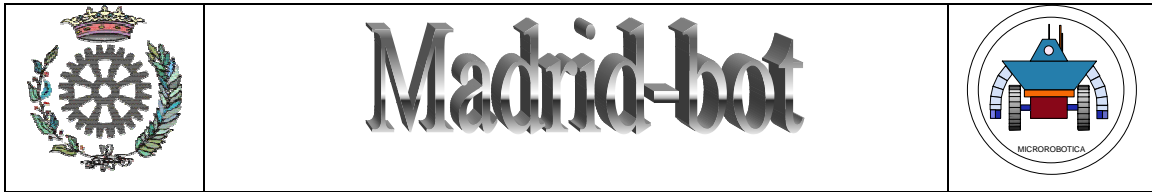
Nostras comenzamos construyendo robots usando piezas y controladores de LEGO. Para construir nuestros primeros robots de SOCCER modificamos los sensores de luz estándar de LEGO, pero seguimos usando el RCX como controlador y las ruedas (dos) y motores de LEGO.

Tras participar en distintas competiciones (tanto nacionales como internacionales) decidimos abandonar la tecnología de LEGO (con la que tanto aprendimos) para diseñar y montar robots muchos más potentes, robustos y capaces de manejar una gran cantidad de sensores y actuadores.

Estos robots de la serie Nexus son el resultado de 8 meses de intenso trabajo con los que hemos aprendido desde como se aprieta un tornillo (pero bien) hasta localizar fallos en el software o en el Hardware.

También ha sido un intensivo curso de electrónica práctica que nos ha permitido llevar a la práctica una gran cantidad de ideas nuevas.

A Nexus 4 y Nexus 5 aún le quedan unos meses para ser robots 100% operativos y lo que es más importante, necesitan competir para poder depurar sus fallos y poder implementar nuevas funciones. A pesar de que estos no estén aún terminados ya



estamos diseñando su siguiente evolución Nexus 6 que tendrá toda la experiencia que hemos adquirido en estos tres modelos.

## **8 Agradecimientos**

La lista de agradecimientos, de personas y entidades que hace posible este proyecto es muy larga, y corremos el riesgo de cometer algún olvido. Aún así, nuestro agradecimiento para:

Departamento de Electrónica de la UAH, en especial a Julio, Marcelo, Sergio... Por su ayuda con el dsPIC y el I2C

Departamento de Robótica de la URJC, en especial a Vicente y a José María. Por su apoyo y orientación.

EducaMadrid, por alojar nuestro sitio web y por todas las herramientas que nos hacen posible el tele-trabajo.

BPI y en especial a Ignacio, por esos estupendas baterías de LIPO.

ARRAY EP, por su ayuda con los componentes

Departamento de Materiales Compuestos del INTA, en especial a José María y María Antonia por su ayuda con todo lo referente a los materiales compuestos.

PRODEL, en especial a Juanjo y Jesús, por ayudarnos con el tema de los sensores, motores, controladores...