

CONSTRUCCIÓN MICRORROBOT CONTROLADO POR PICAXE

CARACTERÍSTICAS

Se trata de un microrrobot capaz de moverse aleatoriamente esquivando obstáculos, seguir líneas oscuras dibujadas sobre fondo blanco y moverse hacia un foco de luz o alejarse del mismo.

ESTRUCTURA

La estructura del microrrobot va a ser una placa de aluminio de de 2 mm de espesor. Se ha elegido este material debido a su ligereza, la facilidad de mecanizado y por ser un material fácil de conseguir.

Con respecto a la forma, ésta puede ser cualquiera, siempre y cuando se puedan acoplar las tres ruedas (dos motrices y una loca) formando un triángulo equilátero. El tamaño será algo mayor del mínimo necesario para no tener problemas en el caso de realizar alguna modificación sobre lo inicialmente previsto.

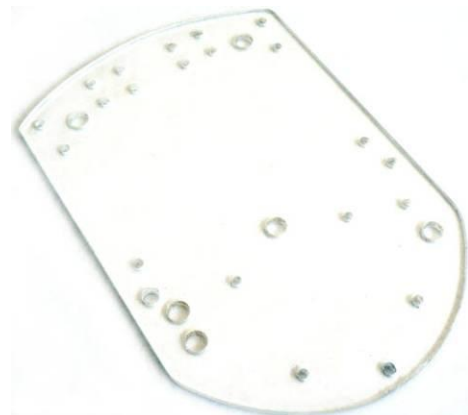
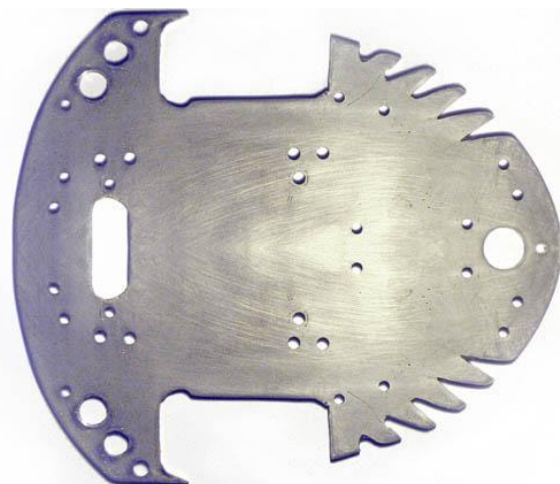
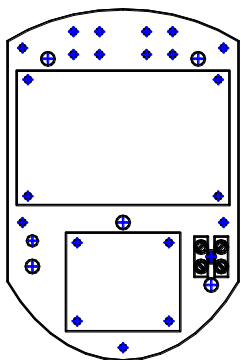
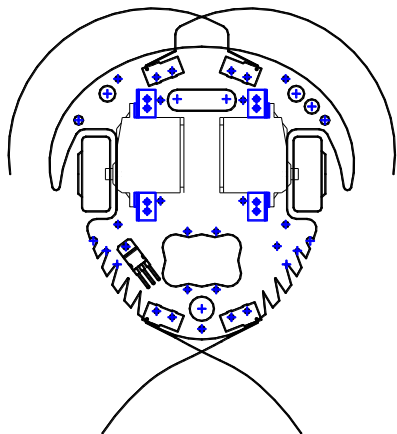
Como forma, teniendo en cuenta el tamaño y posición de los distintos elementos que irán acoplados a la base, se ha elegido una foto de un trilobites para darle un aspecto atractivo.

El diseño de esta base se ha realizado con un programa de CAD y se han incluido más taladros de los necesarios para no tener problemas a la hora de incluir nuevos elementos.

Sobre esta base irán colocados dos motores con sus ruedas, la rueda trasera loca, la batería, los sensores, el regulador de tensión y los soportes que nos permitirán colocar una segunda plataforma que servirá de soporte para la circuitería.

La plataforma superior será de metacrilato de 3 mm de espesor, se fijará a la base mediante 5 separadores de latón y alojará los circuitos, el interruptor general, algunos indicadores luminosos y los orificios necesarios para el paso de los cables.

Por último, para proteger las partes más sensibles, se colocará una tercera lámina de metacrilato, montada sobre separadores.

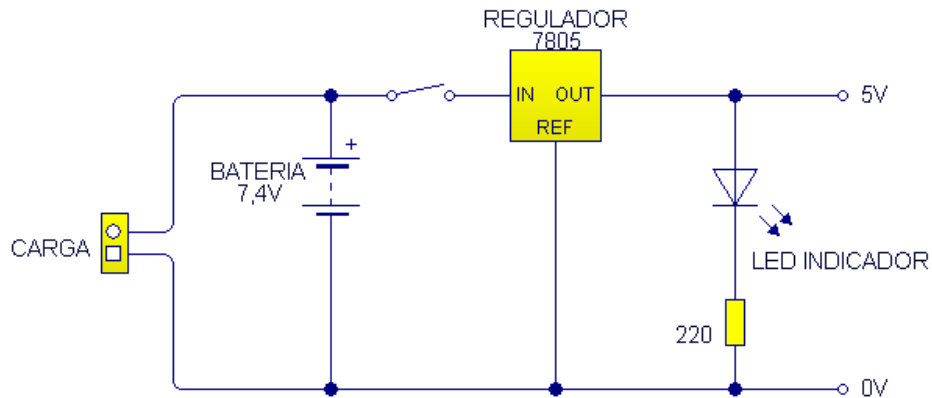
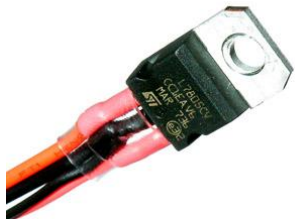
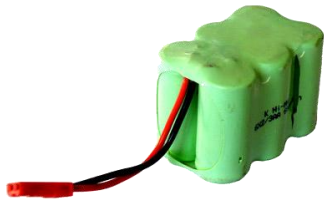


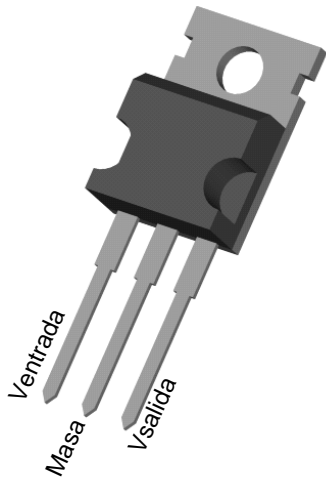
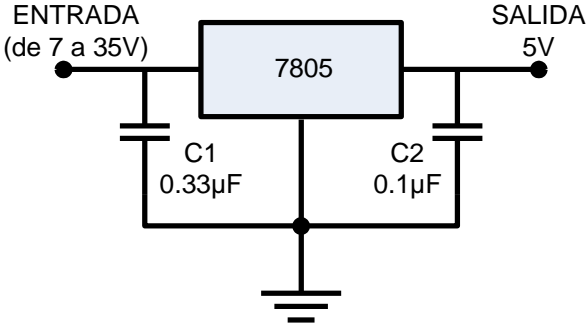
ENERGÍA

Como fuente energética se ha elegido una batería de níquel-metal hidruro de 6 elementos, con una capacidad de 650 mAh a 7,4 voltios. Del cable de este elemento se ha sacado una derivación para dejar libre el terminal de salida y no tener que desconectar el dispositivo para proceder a su recarga.

Como los motores y la circuitería funcionan a 5 voltios, es preciso colocar después del interruptor general un regulador de tensión 7805 para obtener una tensión de 5 voltios estabilizada. Además, este circuito integrado tiene protección contra sobrecargas y cortocircuitos, por lo que evitará problemas y averías al resto del sistema. Este elemento irá atornillado a la base de aluminio de modo que ésta haga la función de disipador. Si no conectamos eléctricamente nada a la base, no será preciso colocar una lámina aislante entre el regulador y la base.

A continuación se muestra el circuito de alimentación y un resumen de las características del regulador de tensión 7805:



CARACTERÍSTICAS BÁSICAS	TERMINALES
<ul style="list-style-type: none"> ▪ Corriente de salida superior a 1 A ▪ Protección contra sobrecalentamiento ▪ Protección contra cortocircuitos 	
<p>CONEXIÓN</p>  <ul style="list-style-type: none"> ▪ C1 es necesario si el regulador se encuentra muy alejado de la fuente de alimentación. ▪ C2 sirve para mejorar la estabilidad 	

MOTORES

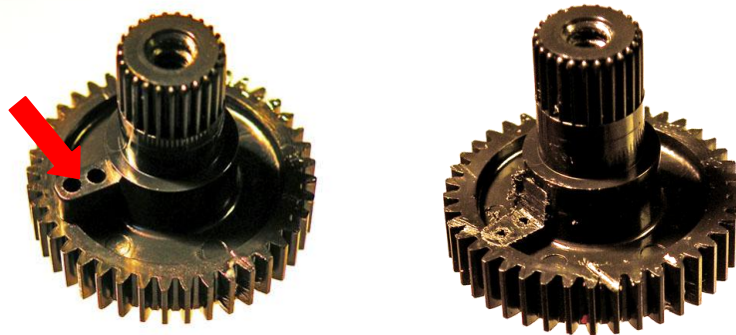
Existen en el mercado muchos tipos de pequeños motores eléctricos con reductora que podrían servir para nuestro robot. A la hora de seleccionar uno tenemos que tener en cuenta lo siguiente:

- Que gire a una velocidad moderada o lenta.
- Que el consumo, el tamaño y el peso sean pequeños.
- Que no produzca muchas interferencias.
- Que sea económico.

Teniendo en cuenta lo dicho anteriormente, se ha optado por una pareja de servomotores de radiocontrol que aunque no pueden girar de forma continua, pueden ser modificados para funcionar como cualquier otro motor. La ventaja de estos motores es su elevada calidad, precio moderado (unos 10 €), buen par, poco peso (35 g) y velocidad adecuada para nuestro diseño.

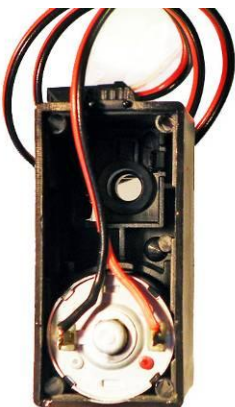
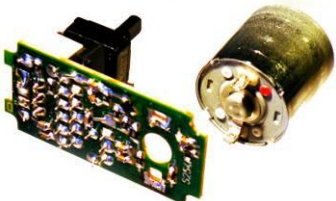
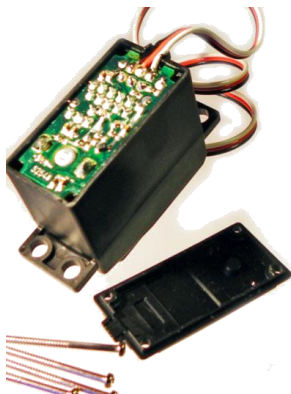
Los pasos a seguir para modificar un servomotor Futaba 3003 son los siguientes:

1. Retiramos la cruceta del eje, los cuatro tornillos de la base y desmontamos el servomotor.
2. Desoldamos el motor y separamos el controlador (circuito impreso). Esta parte ya no nos hará falta.
3. Extraemos el eje secundario y con cuidado, limamos el tope que impide el giro completo.



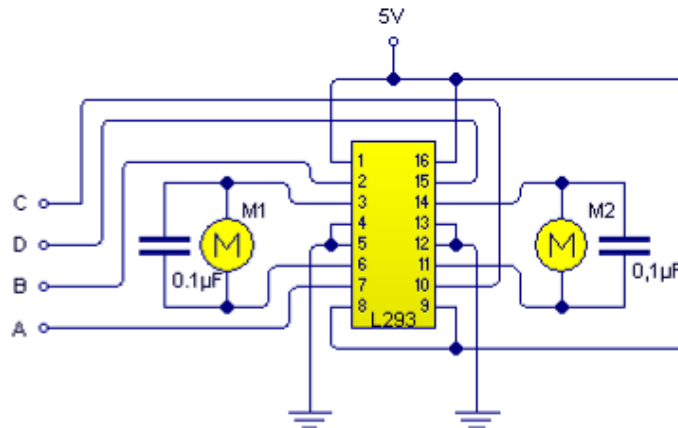
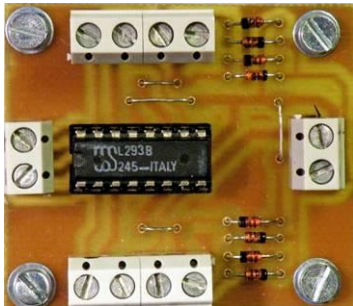
4. Soldamos los cables rojo y negro al motor, teniendo en cuenta las referencias de color (punto rojo para el cable rojo) lo colocamos en su sitio y volvemos a montar todas las piezas.

Como rueda, se ha empleado un disco de metacrilato de 40 mm de diámetro y 3mm de espesor pegado a la cruceta. Para que el robot no patine, se ha practicado una ranura al disco y se ha encajado un anillo tórico.





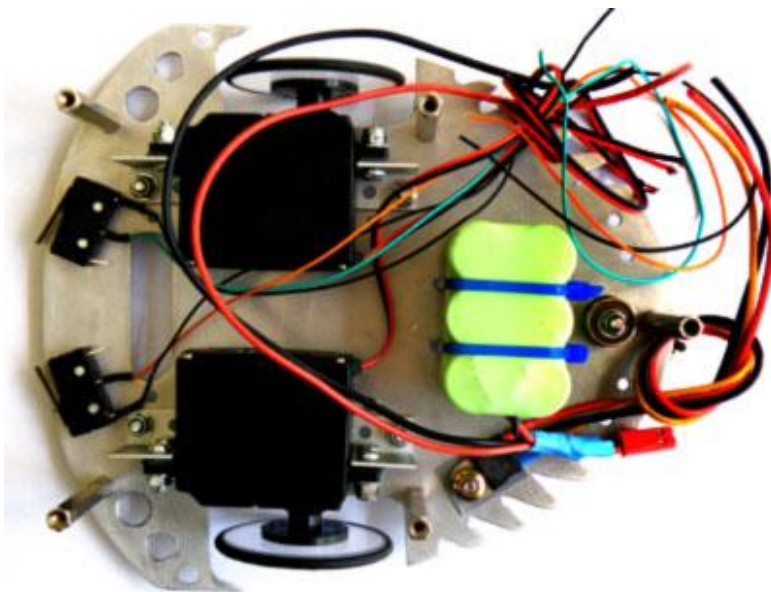
Como estos motores absorben una intensidad relativamente alta, no se pueden conectar directamente al microcontrolador, que sólo puede suministrar unos 25 mA. Por este motivo, habrá que intercalar entre el microcontrolador y los motores un elemento que se encargue de suministrar a los motores la intensidad y polaridad adecuada en función de las señales que recibe del microcontrolador. Este elemento es el circuito integrado L293D que permite gobernar dos motores mediante cuatro entradas de control AB para el motor 1 y CD para el motor 2. Estas entradas las conectaremos a cuatro salidas del microcontrolador.



A	B	M1
0	0	PARO
1	1	PARO
0	1	IZQUIERDA
1	0	DERECHA

C	D	M2
0	0	PARO
1	1	PARO
0	1	IZQUIERDA
1	0	DERECHA

0= nivel bajo (0voltios)
1= nivel alto (>2,3voltios)



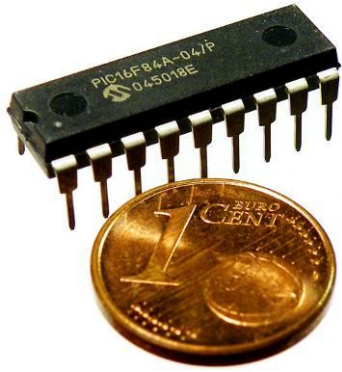
Para evitar las interferencias que pueden producir las escobillas de los motores eléctricos, es necesario colocar un condensador de 0,1 µF en paralelo con los bornes del motor para que actúe a modo de filtro.

La fijación de los motores a la base se puede hacer mediante escuadras, adhesivo o bridas. En este caso hemos fabricado un par de escuadras de aluminio y las hemos fijado a la base y al motor mediante tornillos M3.

El aspecto de la primera capa del robot se puede ver a la izquierda.

CEREBRO

El cerebro de la máquina es un microcontrolador Picaxe 18x. Se ha elegido este circuito por disponer de suficientes entradas y salidas y por admitir programas de hasta 600 líneas aproximadamente.

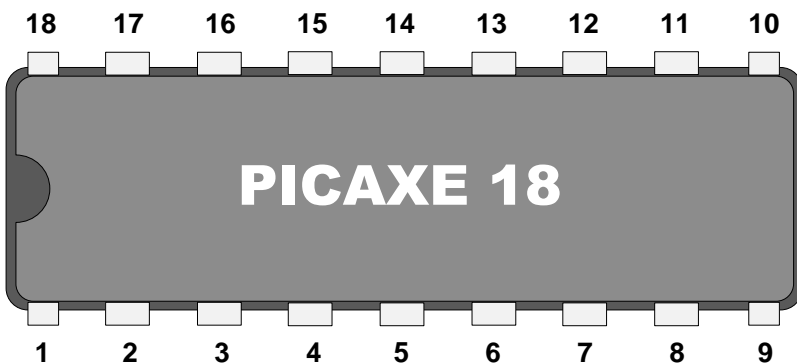


Características del Picaxe18

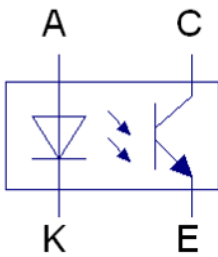
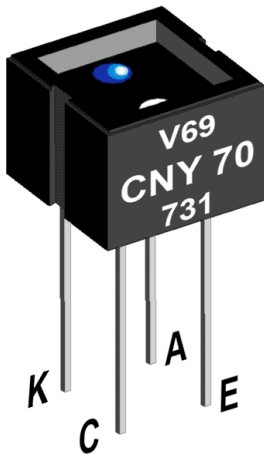
El microcontrolador Picaxe18 posee 4 entradas y 8 salidas y tiene una memoria con capacidad de hasta 600 líneas de programa en la versión 18X. Los modelos 18A y 18X admiten interrupciones y poseen puerto de infrarrojos. El tipo 18X tiene puerto i2c y salida PWM para el control de la velocidad de motores. Las características de este circuito integrado son las siguientes:

Modelo	Nº de pines	Memoria (líneas de programa)	A/D	Entradas	Salidas	Memoria Datos	Interrupciones
18	18	40	3	5	8	128	no
18A	18	80	3	5	8	256	si
18X	18	600	3	5	9	256+i2c	si

Distribución de terminales



1. Entrada 2 (analógica o digital)
2. Salida serie
3. Entrada serie
4. Reset
5. Masa
6. Salida 0
7. Salida 1
8. Salida 2
9. Salida 3
10. Salida 4
11. Salida 5
12. Salida 6
13. Salida 7
14. Alimentación +
15. Entrada 6 (digital)
16. Entrad 7 (digital)
17. Entrada 0 (analógica o digital)
18. Entrada 1 (analógica o digital)



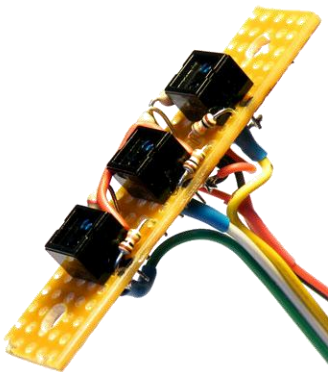
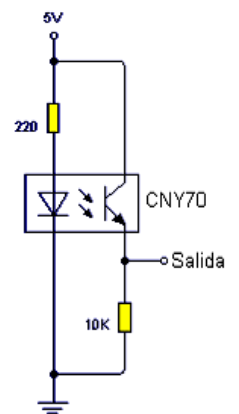
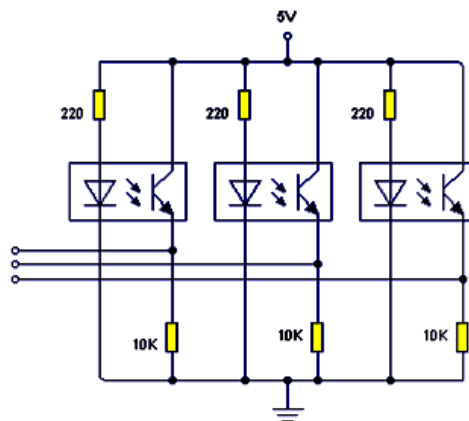
- **Dispositivo óptico emisor/receptor CNY70**

Constan de un emisor de infrarrojos y un fototransistor y detectan objetos de distinto color (por reflexión de infrarrojos) a pequeñas distancias, generalmente inferiores a un centímetro. Si se coloca delante de este sensor un objeto de color claro, la radiación infrarroja emitida por el fotodiodo, se reflejará e incidirá sobre el fototransistor haciendo que éste conduzca.

Aunque estos sensores producen una señal analógica, proporcional a la radiación infrarroja que incide sobre el fototransistor, en este caso los utilizaremos como sensores digitales. El microcontrolador Picaxe, considera que una entrada digital está a nivel alto cuando su tensión supera los 2,5 voltios.

Estos sensores los emplearemos para que el robot pueda seguir una línea negra pintada sobre fondo blanco. Para ello, colocaremos tres sensores alineados y separados unos 10 mm. De este modo, cuando el sensor central detecte color negro el robot se moverá hacia delante. Si el sensor izquierdo detecta negro y los otros no, nos estamos desviando hacia la izquierda y habrá que actuar en consecuencia y si es el sensor derecho el único que detecta negro, la desviación es hacia la izquierda. Como en el caso anterior, es necesario colocar una resistencia "push down" de 10k para tener nivel bajo a la salida cuando no conduce el fototransistor.

El esquema de conexión se muestra a continuación.

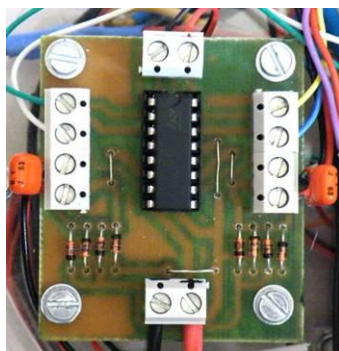
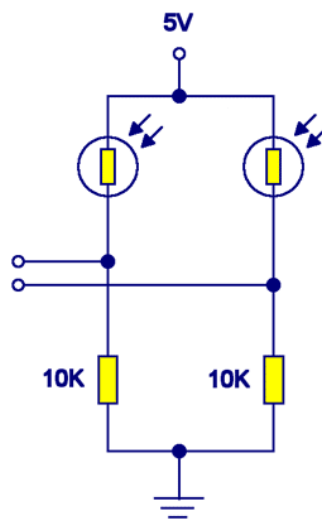
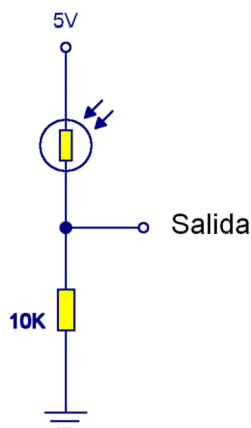
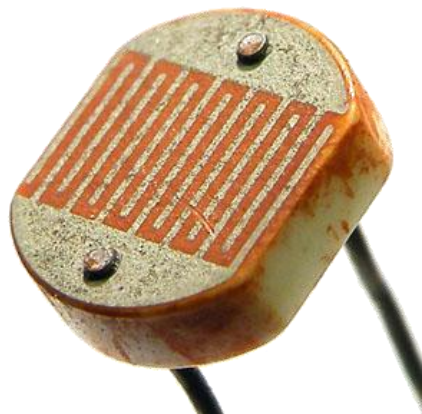


Si necesitamos que estos sensores tengan un mayor alcance podemos cambiar las resistencias limitadoras de 220 Ω por otras de 100 Ω.

•LDR

Mediante estas resistencias podemos obtener una señal analógica proporcional a la cantidad de luz que incide sobre su superficie activa. La variación de la resistencia en estos sensores es muy drástica y suele oscilar entre varios megaohmios cuando no incide luz sobre su superficie activa a unos 10 ohmios cuando está expuesta a una elevada iluminación.

Nuestro prototipo dispondrá de dos LDR colocadas en el frontal, mirando hacia delante y formando un ángulo de unos 120 grados. La finalidad de estos sensores es producir una señal analógica proporcional a la iluminación que reciben. El microcontrolador compara estas señales y actúa sobre los motores para que el robot se mueva hacia la zona más iluminada o a la zona más oscura, según programemos.



CONEXIONES

La forma de de conectar los elementos vistos anteriormente es la siguiente:

•Motores

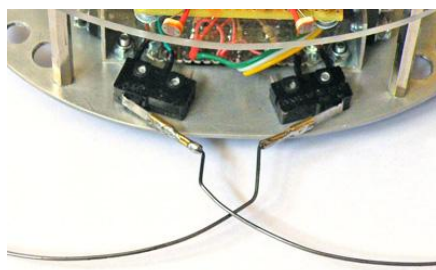
La placa de control de motores se conectará a las cuatro primeras salidas del microcontrolador (A, B, C y D con las salidas 0,1,2,3) y los terminales de alimentación se unirán a la salida del regulador de tensión (5v), respetando la polaridad.

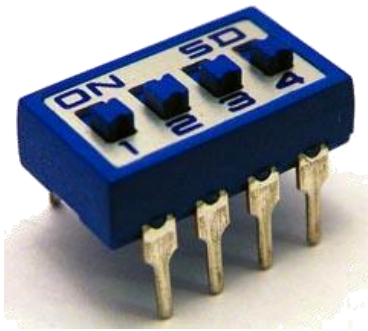
•Sensores CNY50

Los sensores CNY70 se alimentan a 5V, por lo que han de conectarse al circuito de alimentación. Las salidas obtenidas de los sensores derecho, central e izquierdo irán conectadas a las entradas 1, 0 y 2.

•Antenas (microrruptores)

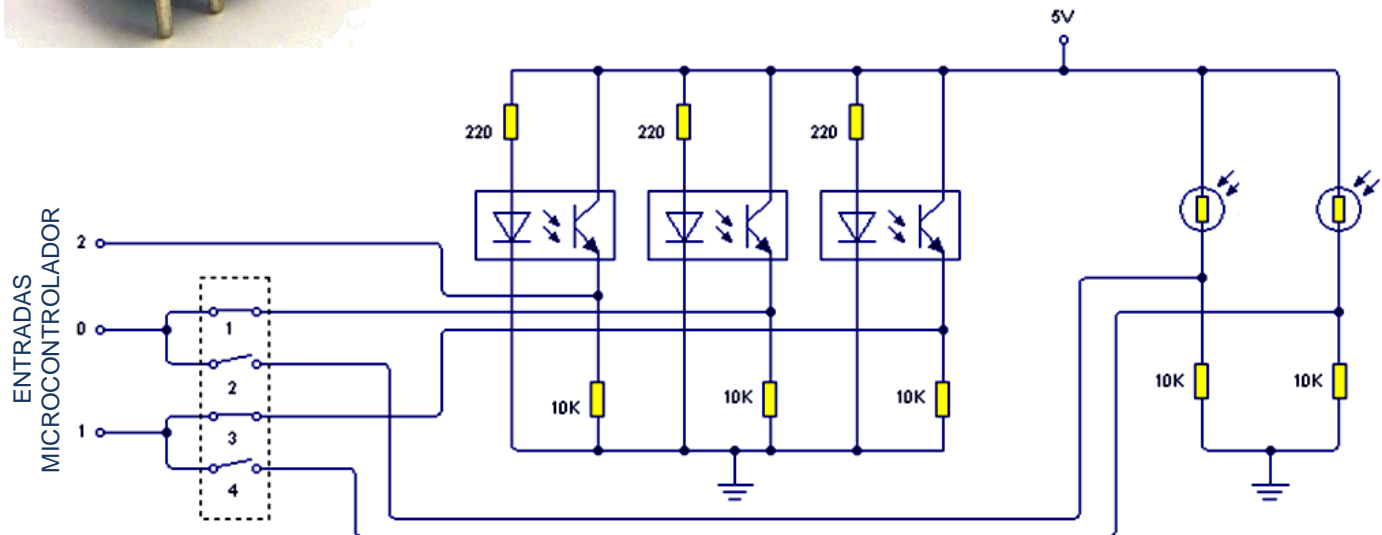
Irán conectados a la entrada 7 el derecho y a la entrada 6 el izquierdo. Si tienen tres terminales hay que tener la precaución de conectar al microcontrolador las patillas correspondientes al contacto normalmente abierto (NO). El común irá al positivo de la alimentación (+5v). Además de para detectar obstáculos, las antenas nos servirán para seleccionar el programa que queramos ejecutar.



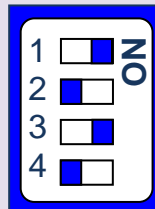
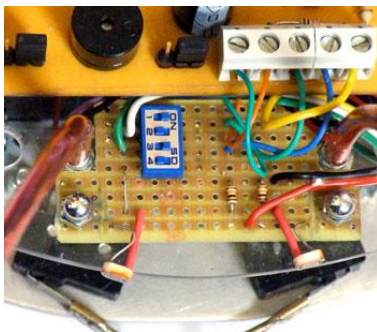


• Sensores LDR

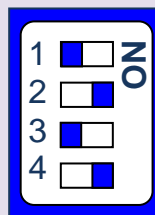
Estos sensores han de ir conectados a dos de las tres entradas analógicas/digitales del microcontrolador (0 y 1). Como sólo disponemos de 5 entradas y tenemos 7 elementos que conectar, las dos LDR han de compartir entrada con dos CNY70. Mediante una matriz de cuatro de microinterruptores seleccionaremos el par de sensores que queremos conectar a las dos entradas analógicas (0 y 19)



Según la combinación en el selector, conectaremos a las entradas 0 y 1 la LDR o los sensores CNY70 central y derecho atendiendo al siguiente esquema:



CNY70 central y derecho conectados a las entradas 0 y 1



LDR izquierda y derecha conectadas a las entradas 0 y 1



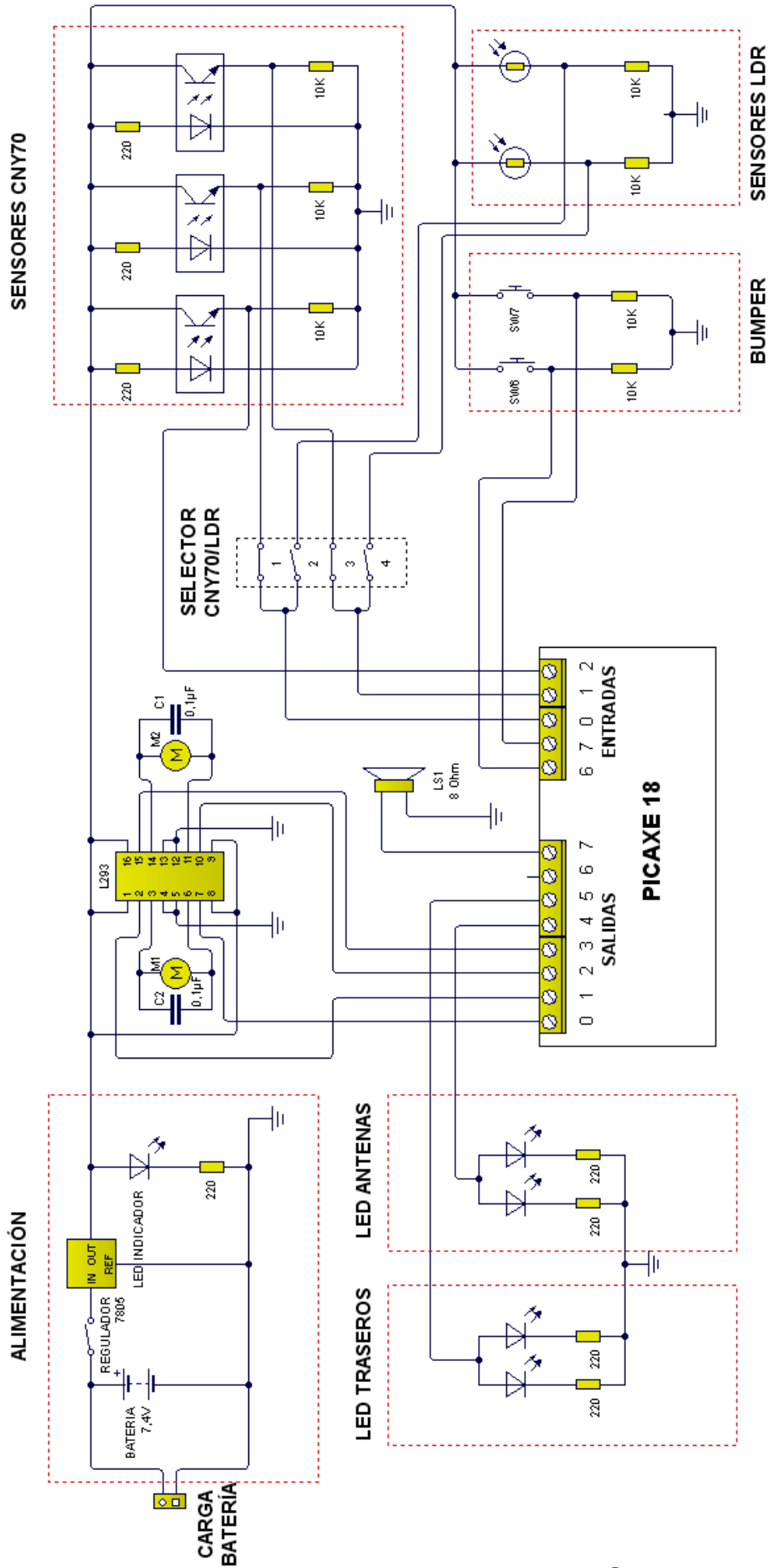
• INDICADORES LUMINOSOS

Los indicadores luminosos (LED) se conectarán a las salidas 4 y 5. Cada indicador está formado por dos LED. Uno se halla en las antenas superiores o cuernos y el otro en la parte trasera del prototipo.

• ALTAVOZ

Se trata de un microaltavoz piezoeléctrico. Va montado en la placa de control y conectado a la salida 7

ESQUEMA GENERAL



SOFTWARE

El diseño del software es la parte más apasionante del proyecto. Un software bien planteado puede mejorar enormemente la calidad del resultado. El modo de proceder es realizar pequeños programas, probarlos y una vez tengamos todos los bloques necesarios, agruparlos en un único programa.

- **Conexiones:**

Para facilitar la programación, tenemos que tener a mano un esquema con las conexiones que hemos realizado previamente. En este caso las conexiones realizadas a las entradas y salidas del microcontrolador son las siguientes:

ENTRADAS:

- 7 Microrruptor Izquierdo (Antena derecha)
- 6 Microrruptor Derecho (Antena derecha)
- 7 Microrruptor Izquierdo (Antena derecha)
- 1 CNY70 Derecho/LDR Derecha (Entrada analógica)
- 0 CNY70 Central /LDR Izquierda (Entrada analógica)
- 2 CNY70 Izquierdo

SALIDAS:

- 0 y 1 Entradas AB del L293 (Control del motor derecho)
- 2 y 3 Entradas CD del L293 (Control del motor izquierdo)
- 4 Indicadores luminosos (Cuernos)
- 5 Luz trasera

- **Programas para comprobación de los motores.**

Para comprobar el correcto funcionamiento de los motores usaremos los siguientes programas:

Movimiento hacia delante:

Adelante:

high 0 'pone la salida 0 a nivel alto
low 1 'pone la salida 1 a nivel bajo

high 2 'pone la salida 2 a nivel alto
low 3 'pone la salida 3 a nivel bajo

Como las salidas 0 y 1 están conectadas a las entradas A y B del L293, éstas gobernarán el motor derecho. Las salidas 2 y 3 controlan el motor izquierdo. Este programa hace girar indefinidamente los dos motores. Si no giran en el mismo sentido habrá que invertir las conexiones de uno de ellos. Si ambos motores giran hacia atrás, haremos el siguiente cambio:

Adelante:

low 0 'pone la salida 0 a nivel bajo
high 1 'pone la salida 1 a nivel alto

low 2 'pone la salida 2 a nivel bajo
high 3 'pone la salida 3 a nivel alto

Giro a la derecha parando un motor:

Giraderecha:

low 0 'pone la salida 0 a nivel bajo
low 1 'pone la salida 1 a nivel bajo

high 2 'pone la salida 2 a nivel alto
low 3 'pone la salida 3 a nivel bajo

Las dos primeras líneas hacen que se detenga el motor derecho y las dos siguientes hacen que el izquierdo gire hacia delante. De este modo se consigue que el robot gire a la derecha.

Giro a la izquierda invirtiendo el sentido de marcha de un motor:

Giraizquierda:

low 0 'pone la salida 0 a nivel bajo
high 1 'pone la salida 1 a nivel alto

high 2 'pone la salida 3 a nivel alto
low 3 'pone la salida 2 a nivel bajo

En este caso, el motor derecho, controlado por las dos primeras líneas, gira hacia delante mientras que el izquierdo gira hacia atrás. De esta forma se consigue un giro más cerrado que deteniendo uno de los motores.

Movimiento hacia delante durante un tiempo:

Adelante:

low 0 'pone la salida 0 a nivel bajo
high 1 'pone la salida 1 a nivel alto

low 2 'pone la salida 2 a nivel bajo
high 3 'pone la salida 3 a nivel alto

wait 10 'pausa de segundos

low 0 'pone la salida 0 a nivel bajo

low 1 'pone la salida 1 a nivel bajo
low 2 'pone la salida 2 a nivel bajo
low 3 'pone la salida 3 a nivel bajo

Las primeras cuatro líneas de programa hacen que los dos motores giren a derechas. La siguiente línea detiene el programa 10 segundos (los dos motores siguen girando durante este tiempo). Las siguientes líneas ordenan la detención de los dos motores y el programa concluye.

Bucle:

vaiven:

low 0 'pone la salida 0 a nivel bajo
high 1 'pone la salida 1 a nivel alto
low 2 'pone la salida 2 a nivel bajo
high 3 'pone la salida 3 a nivel alto

wait 10 'pausa de segundos

high 0 'pone la salida 0 a nivel alto
low 1 'pone la salida 1 a nivel bajo
high 2 'pone la salida 2 a nivel alto
low 3 'pone la salida 3 a nivel bajo

wait 5 'pausa de segundos

goto vaiven

Este programa comienza con los dos motores girando a derechas durante 10 segundos. Transcurrido ese tiempo, los motores invierten su marcha durante otros 10 segundos y la última línea del programa hace que volvamos al inicio, de modo que el programa se estará repitiendo indefinidamente.

- **Programas básicos de entrenamiento:**

Primer programa: cuadrado

Una trayectoria cuadrada se compone de cuatro trayectorias rectas iguales y cuatro giros de noventa grados. Para poder programar con precisión, es necesario conocer de antemano la velocidad a la que se mueve el prototipo. Para ello, basta con cronometrar el tiempo que tarda en recorrer un metro y el tiempo necesario para completar un giro de 360° con un motor parado. En este caso hemos obtenido los siguientes datos:

Tipo de movimiento	Cantidad	Tiempo
Línea recta	1 m	15 s
Giro	360°	10 s

A partir de estos datos, podemos crear un programa que haga que el movimiento del robot describa un cuadrado perfecto del tamaño que queramos. El programa siguiente define una trayectoria cuadrada de 40 cm de lado. Para ello se ha tenido en cuenta, además de los tiempos la distancia entre ejes que es de 10 cm.

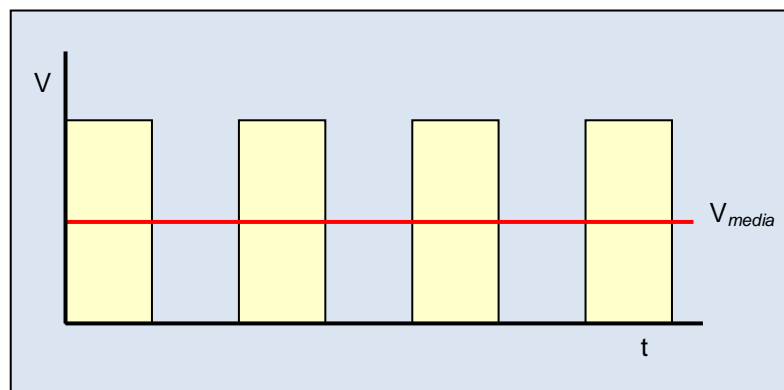
cuadrado:

high 0 low 1 high 2 low 3	}	Trayectoria recta durante 3 s (20 cm)
wait 3		
high 0 low 1 low 2 low 3	}	Giro durante 2,5 s con un motor parado (90° y 10 cm)
pause 2500		

goto cuadrado

Variación de la velocidad:

Una manera muy interesante de reducir la velocidad de un motor de una forma precisa, sin perder mucho par y sin tener que modificar la tensión que llega a los motores, es la modulación por ancho de pulso (PWM). Esta técnica consiste alimentar a los motores de forma intermitente mediante un tren de ondas rectangulares similares a los mostrados a continuación:



En este caso, los impulsos de tensión duran lo mismo que los tiempos de reposo por lo que la tensión media es la mitad y por tanto, el motor girará también a la mitad de la velocidad que tendría si le llegase el cien por cien de tensión.

Si queremos reducir más aún la velocidad tenemos que reducir la duración de los impulso y hacer mayor el tiempo de reposo.

La regulación de velocidad PWM requiere que la frecuencia de los pulsos enviados al motor ronde los 10 Kilohercios y el uso de motores de buena calidad.

A continuación se muestra un ejemplo de regulación PWM:

`pwm1:`

`high 0`
`low 1`
`high 2`
`low 3` } Los dos motores en marcha durante
1 milisegundo

`pause 1` }

`low 0`
`low 1`
`low 2`
`low 3` } Los dos motores en reposo durante
1 milisegundo

`pause 1` }

`goto pwm1`

Círculo:

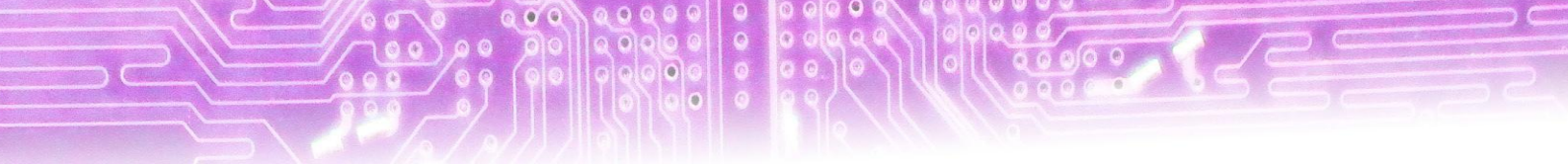
Aplicando una regulación PWM a uno de los motores, podemos hacer que nuestro prototipo siga una trayectoria circular. El tamaño del círculo dependerá de la relación entre el tiempo que está parado y girando el motor que gira de forma discontinua.

`circulo3:`

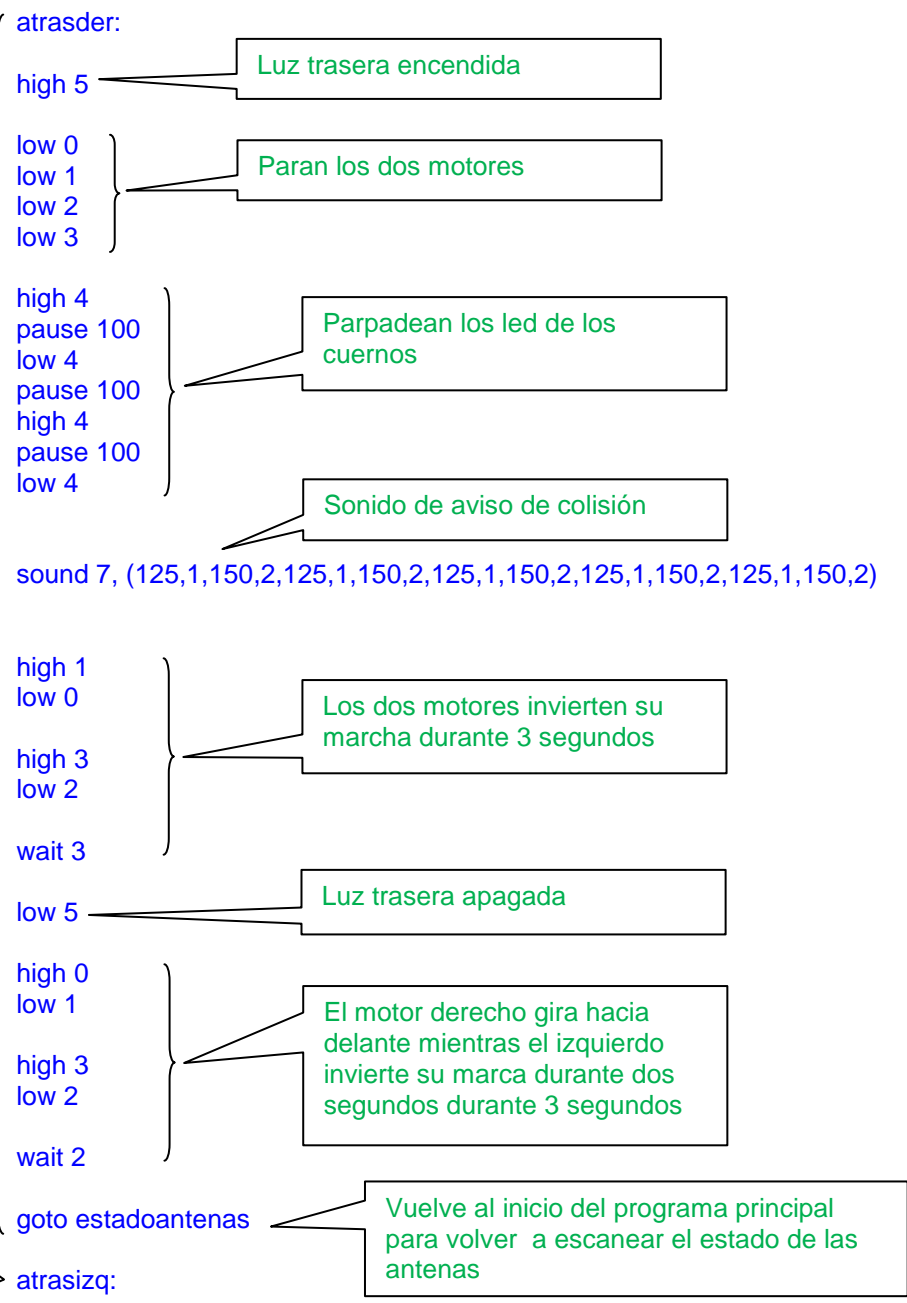
`high 0`
`low 1`
`high 2`
`low 3`
`pause 2` } Los dos motores giran en el mismo
sentido durante 2 milisegundos

`high 0`
`low 1`
`low 2`
`low 3`
`pause 1` } El motor 1 gira y el motor 2 se detiene
durante 1 milisegundo

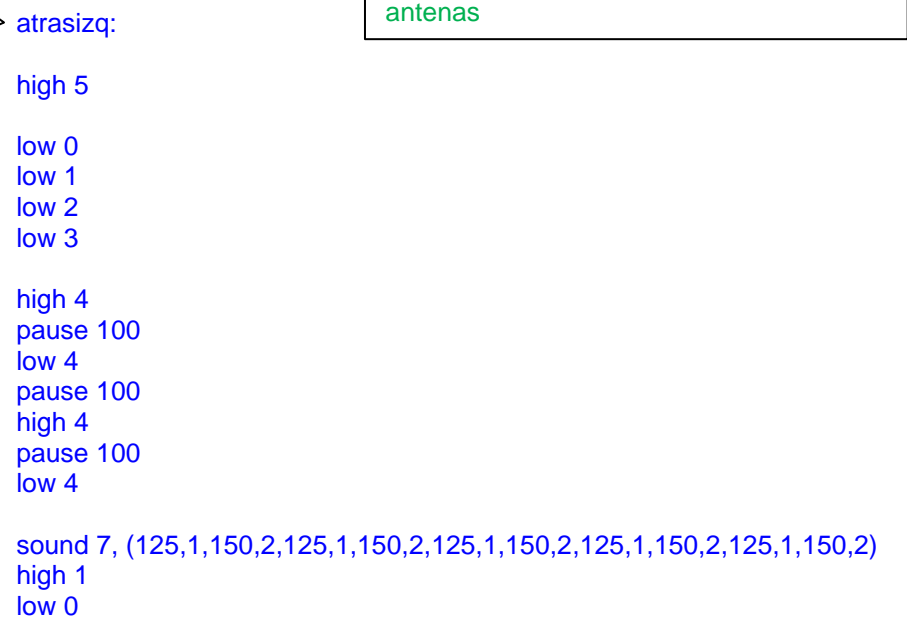
`goto circulo3`



Este programa hace que el robot se mueva hacia atrás tres segundos y después gire a la derecha unos 180 grados



Este programa hace que el robot se mueva hacia atrás tres segundos y después gire a la izquierda unos 180 grados. Sólo se diferencia del programa anterior en el sentido del último giro



high 3
low 2

wait 3

low 5

high 1
low 0

high 2
low 3

wait 2

goto estadoantenas

- Programa SIGUELINEAS

Este programa hace que el robot siga una línea negra dibujada sobre un fondo blanco. Para ello, utiliza tres sensores CNY70 que dan salida 0 cuando tienen delante una superficie oscura.

```
!*****  
!*****          SIGUELINEAS          *****  
!*****
```

```
'      CONEXIONES  
'  
'      ENTRADAS:  
'  
'      CNY70 DER / LDR DER      ENTRADA 1  
'      CNY70 CEN / LDR IZQ ENTRADA 0  
'      CNY70 IZQ                ENTRADA 2  
'  
'      SALIDAS:  
'  
'      MOTOR DERECHO           SALIDAS 0 Y 1  
'      MOTOR IZQUIERDO        SALIDAS 2 Y 3  
'      CUERNOS                 SALIDA 4  
'      LUZ TRASERA            SALIDA 5
```

SIGUELINEAS:

high 5

if input2 is off then giraizquierda

if input1 is off then giraderecha

if input0 is off and input2 is on and input1 is on then adelante

if input0 is off and input2 is off and input1 is off then giraderecha

goto siguelineas

giraizquierda:

high 0
low 1

low 2
low 3

goto siguelineas

giraderecha:

high 2
low 3

low 0
low 1

goto siguelineas
adelante:

high 0
low 1

high 2
low 3

goto siguelineas

- **Programa LDR**

Se basa en comparar los valores analógicos que aportan dos sensores LDR colocados en el frontal del robot.

```
!*****  
!*****          LDR          *****  
!*****
```

```
!      CONEXIONES  
!      ENTRADAS:  
!      BUMPER IZDO          ENTRADA 7  
!      BUMPER DER          ENTRADA 6  
!      CNY70 DER / LDR DER  ENTRADA 1  
!      CNY70 CEN / LDR IZQ ENTRADA 0  
!      CNY70 IZQ           ENTRADA 2  
!      SALIDAS:  
!      MOTOR DERECHO       SALIDAS 0 Y 1  
!      MOTOR IZQUIERDO    SALIDAS 2 Y 3  
!      CUERNOS             SALIDA 4  
!      LUZ TRASERA        SALIDA 5
```

LDR:


if input7 is off and input6 is off then SIGUELUZ

if input7 is on then atrasderecha

if input6 is on then atrasizquierda

atrasderecha:

low 0
low 1
low 2
low 3



high 4
pause 100
low 4
pause 100
high 4
pause 100
low 4

sound 7, (125,1,150,2,125,1,150,2,125,1,150,2,125,1,150,2,125,1,150,2)

high 1
low 0

high 3
low 2

wait 2

high 0
low 1

high 3
low 2

WAIT 2

goto LDR

atrasizquierda:

low 0
low 1
low 2
low 3

high 4
pause 100
low 4
pause 100
high 4
pause 100
low 4

sound 7, (125,1,150,2,125,1,150,2,125,1,150,2,125,1,150,2,125,1,150,2)

high 1
low 0

high 3
low 2

wait 2

high 1
low 0

high 2
low 3

WAIT 2

goto LDR

SIGUELUZ:

```
readadc 0,b0 'ldr izquierda
readadc 1,b1 'ldr derecha
if b0 > b1 then izquierda
if b0 < b1 then derecha
if b0 = b1 then haciadelante
```

```
goto LDR
izquierda:
```

```
high 0
low 1
```

```
low 2
low 3
goto ldr
derecha:
```

```
high 2
low 3
```

```
low 0
low 1
```

```
goto ldr
haciadelante:
```

```
high 0
low 1
```

```
high 2
low 3
```

```
goto ldr
```

- Selector de programas


```
!*****
!***** SELECCION DE PROGRAMA *****
!*****
```

SELECCIONPROGRAMA:

```
'LOS DOS CUERNOS PULSADOS AL INICIO          SIGUELINEAS
'EL CUERNO IZQUIERDO PULSADO AL INICIO        LDR
'NINGUN CUERNO PULDADO AL INICIO              LIBRE
```

```
sound 7, (105,10)
pause 100
sound 7, (105,10)
pause 100
sound 7, (105,10)
pause 100
sound 7, (105,10)
pause 100
sound 7, (105,10)
pause 100
sound 7, (105,100)
```

```
if input7 is on and input6 is on then SIGUELINEAS
if input7 is on and input6 is off then LDR
if input7 is off and input6 is off then LIBRE
```



SALUDO:
high 4
high 5
pause 100
low 4
low 5
pause 100

high 4
high 5
pause 100
low 4
low 5
pause 100

high 4
high 5
pause 100
low 4
low 5
pause 100

high 4
high 5
pause 100
low 4
low 5
pause 100

sound 7, (125,1,150,2,125,1,150,2,125,1,150,2,125,1,150,2,125,1,150,2)
PAUSE 100
sound 7, (125,1,150,2,125,1,150,2,125,1,150,2,125,1,150,2,125,1,150,2)
PAUSE 300
sound 7, (125,1,150,2,125,1,150,2,125,1,150,2,125,1,150,2,125,1,150,2)
PAUSE 100
sound 7, (125,1,150,2,125,1,150,2,125,1,150,2,125,1,150,2,125,1,150,2)

return