



# Manual R8



[www.misladrillos.com](http://www.misladrillos.com)



R8

---

# Índice

## R8

- [Introducción](#)
- [Entradas y Salidas](#)
- [Conexiones sugeridas](#)
- [Instalación y configuración Arduino](#)
- [Instalación ArduBlock](#)

## Periféricos R8

- [Motores](#)
- [Led bicolor](#)
- [Sensor de luz o LDR](#)
- [Sensor de tacto o Switch](#)
- [Sensor CNY70](#)
- [Micrófono](#)
- [Sensor de distancia](#)
- [Bluetooth](#)
- [Batería](#)

## Sensores incluidos en la placa

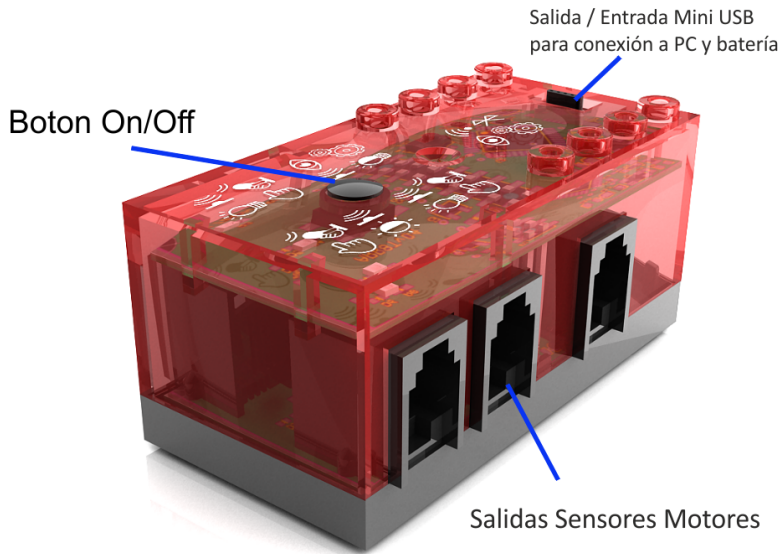
- [Receptor IR](#)
- [Mini parlante buzzer](#)
- [Cuadro resumido de las salidas del chip Atmel y los puertos del R8](#)

## [Batería](#)

## R8

# R8

El ladrillo llamado R8 es el “cerebro” del kit robótico. Salida / Entrada Mini USB para conexión a PC y batería. Tiene conectores laterales para motores y sensores.

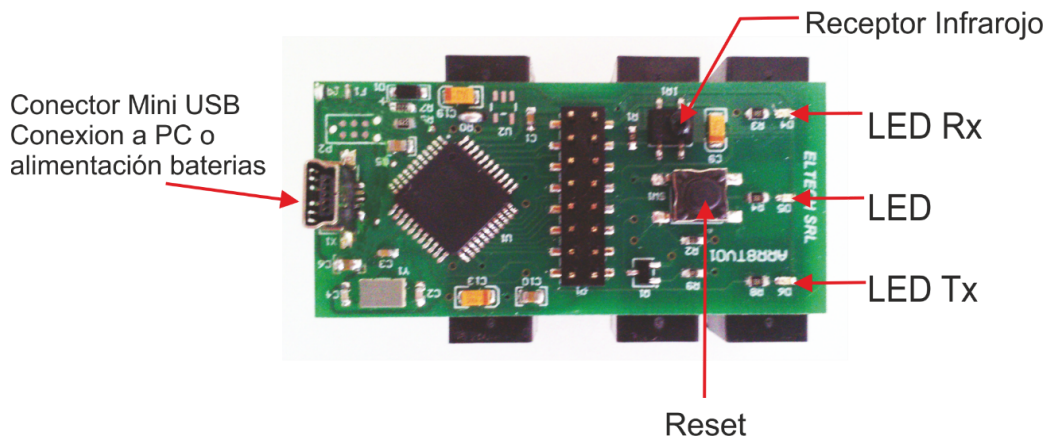


Su corazón es una Placa simil Arduino Genuino con agregados.

Un agregado es un sensor infrarrojo para detectar un control remoto y un mini parlante para reproducir melodías y sonidos básicos.

Cuenta con 3 led indicadores y se alimenta y/o conecta a la PC por medio de una entrada de conexión mini USB. El software de Arduino detecta esta placa como Arduino Genuino micro.

El chip “cerebro” del conjunto es un Chip ATMEL Atmega32U 1628E.



## R8

El ladrillo tiene varias conexiones. Por cada conector tiene alimentación +5V, masa y salidas/entradas digitales y/o analógicas

### ENTRADAS y SALIDAS DIGITALES:

Estos se llaman digitales porque sólo pueden manejar valores 0 o 1 (apagado-encendido).

### ENTRADAS Y SALIDAS ANALÓGICAS:

Una señal eléctrica analógica es aquella en la que los valores de la tensión o voltaje varían constantemente y pueden tomar cualquier valor. Un sistema de control (como un microcontrolador) no tiene capacidad alguna para trabajar con señales analógicas, de modo que necesita convertirlas en señales digitales para poder trabajar con ellas.

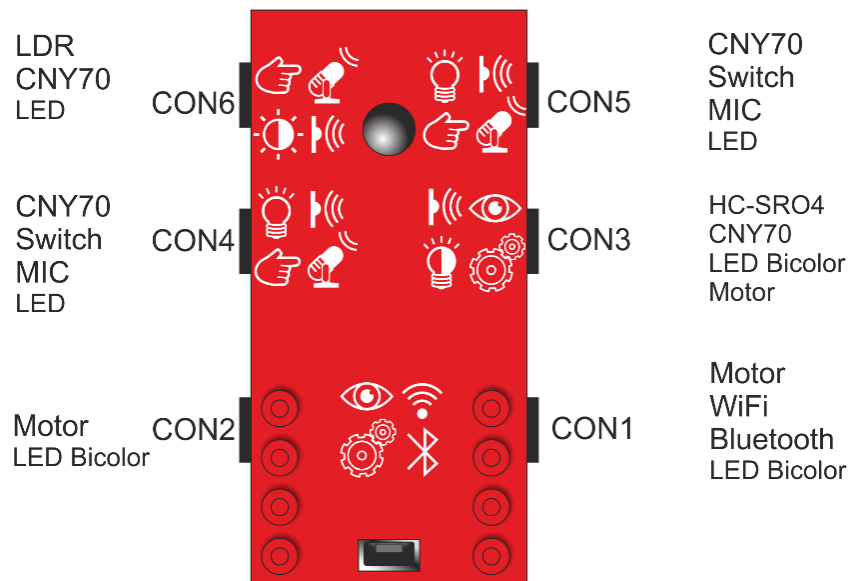
En el caso de un Arduino, el valor de 0 voltios analógico es expresado en digital como 0 y el valor de 5V analógico es expresado en digital como 1023.

Arduino tiene una resolución de 10 bits, es decir, unos valores entre 0 y 1023.

## Conexiones sugeridas

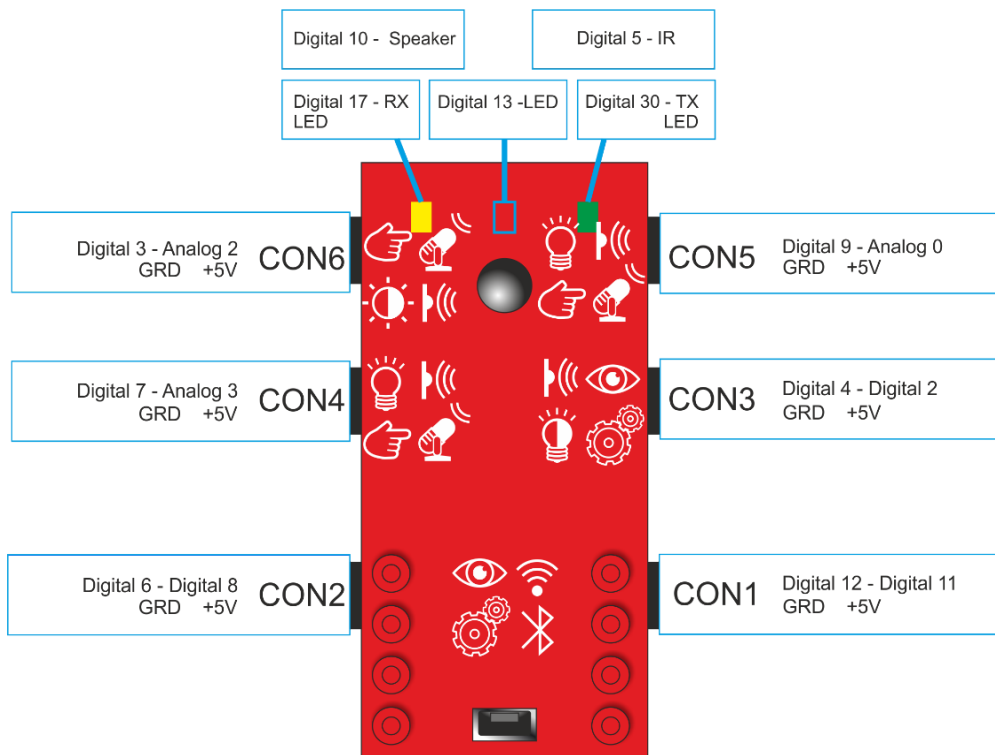
Su corazón es una Placa simil Arduino Genuino con agregados: un sensor infrarrojo para detectar un control remoto y un mini parlante para reproducir melodías y sonidos básicos.

HC-SRO4 = SENSOR DE DISTANCIA POR ULTRASONIDO  
 CNY70 = SENSOR DE DISTANCIA CORTA POR INFRAROJO



Este es un esquema de las salidas PIN por cada conector. Se necesitan en el caso de usar código de Arduino. Si se usa nuestro programa recomendado "Ardublock para Mis Ladrillos" sólo se debe poner directamente el número de conector.

Los íconos impresos son sugerencias de conexión pero las posibilidades de conexión en algunos casos son más amplias (ver abajo)



## Instalación y configuración Arduino

Arduino es una plataforma de código abierto que se utiliza para la construcción de proyectos de electrónica. Una placa Arduino consiste placa de circuito físico programable, que tiene un microcontrolador y un programa que se puede desarrollar en el IDE de Arduino (Entorno de Desarrollo Integrado) que se ejecuta en un ordenador, se utiliza para escribir y cargar el código de la computadora a la tarjeta física. <https://www.arduino.cc/>

Recomendamos instalar estas versiones de Arduino que son las más compatibles con el Software Ardublock Para Mis ladrillos.

Arduino 1.6.9

WINDOWS: [https://www.arduino.cc/download\\_handler.php?f=/arduino-1.6.9-windows.exe](https://www.arduino.cc/download_handler.php?f=/arduino-1.6.9-windows.exe)

MAC OS: [https://www.arduino.cc/download\\_handler.php?f=/arduino-1.6.9-macosx.zip](https://www.arduino.cc/download_handler.php?f=/arduino-1.6.9-macosx.zip)

LINUX: [https://www.arduino.cc/download\\_handler.php?f=/arduino-1.6.10-linux32.tar.xz](https://www.arduino.cc/download_handler.php?f=/arduino-1.6.10-linux32.tar.xz)

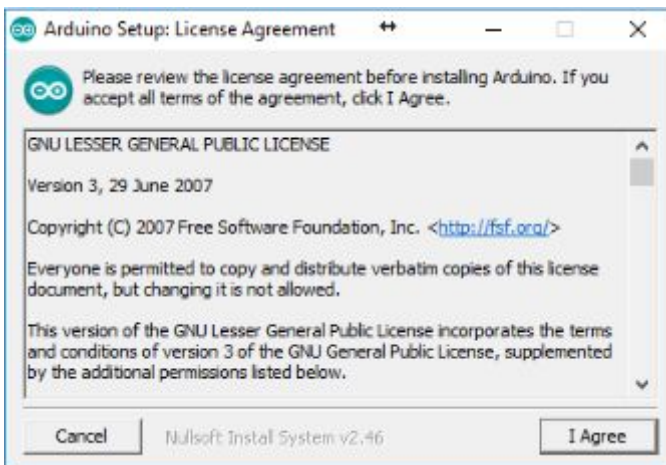
Una vez descargado hay que instalarlo como cualquier software común, según tu sistema operativo.

## Ejemplo en Windows:

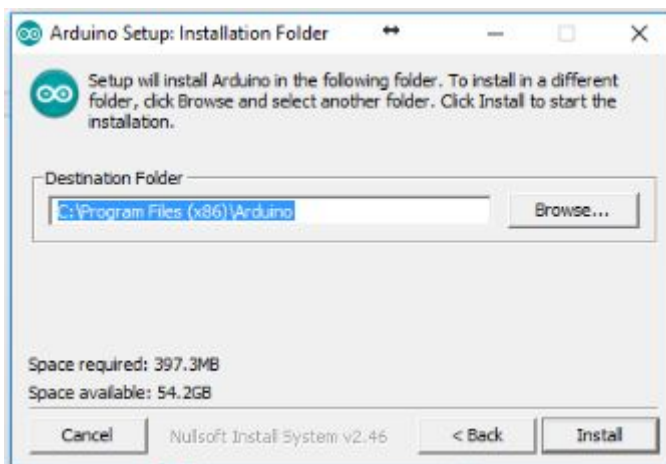
Hacer doble click en el archivo descargado:



Aceptar



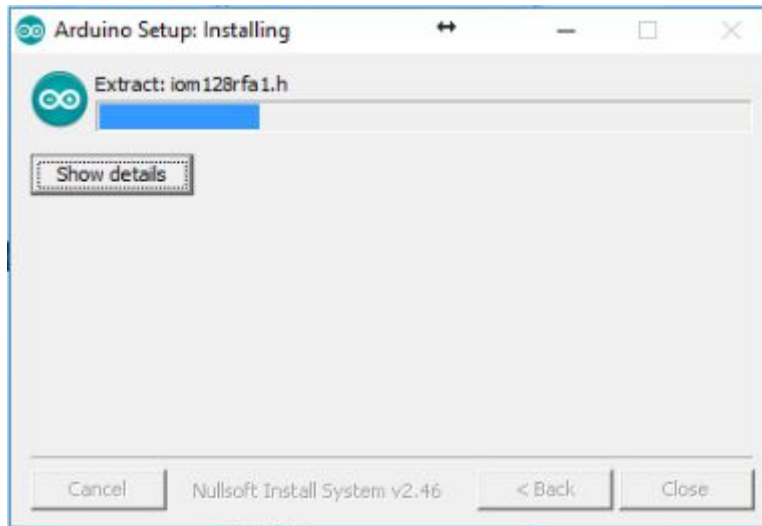
Instalar:





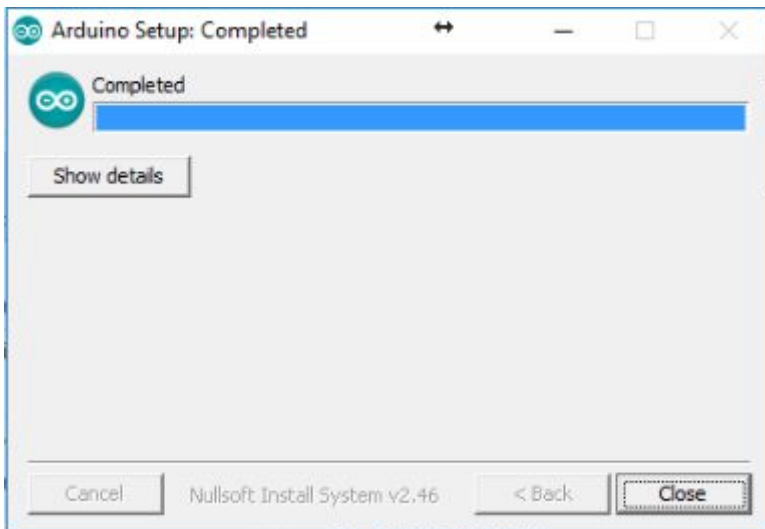
R8

---

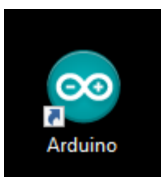


Esperar.....

Cerrar



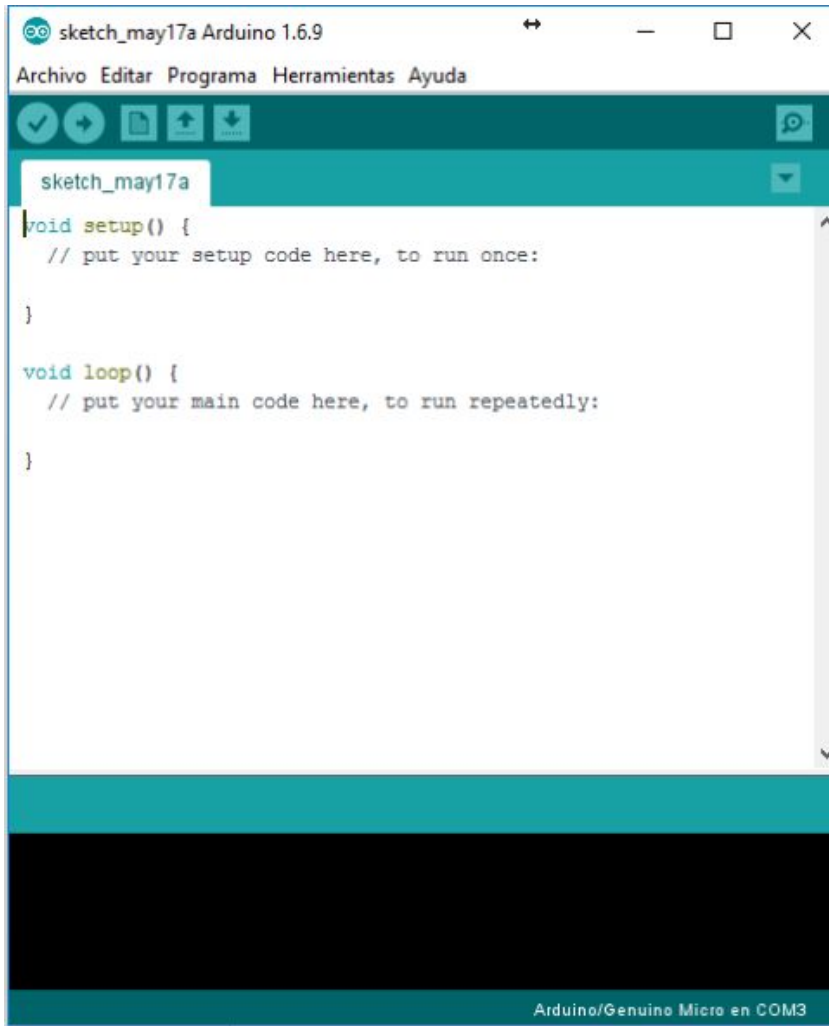
Ya se instaló el software de Arduino, para abrirlo hacer doble click en el ícono del Escritorio (o en su defecto en inicio/programas)



## R8

---

Se abre una ventana como esta:



Ahora deberás conectar el ladrillo R8 al puerto usb de la computadora.



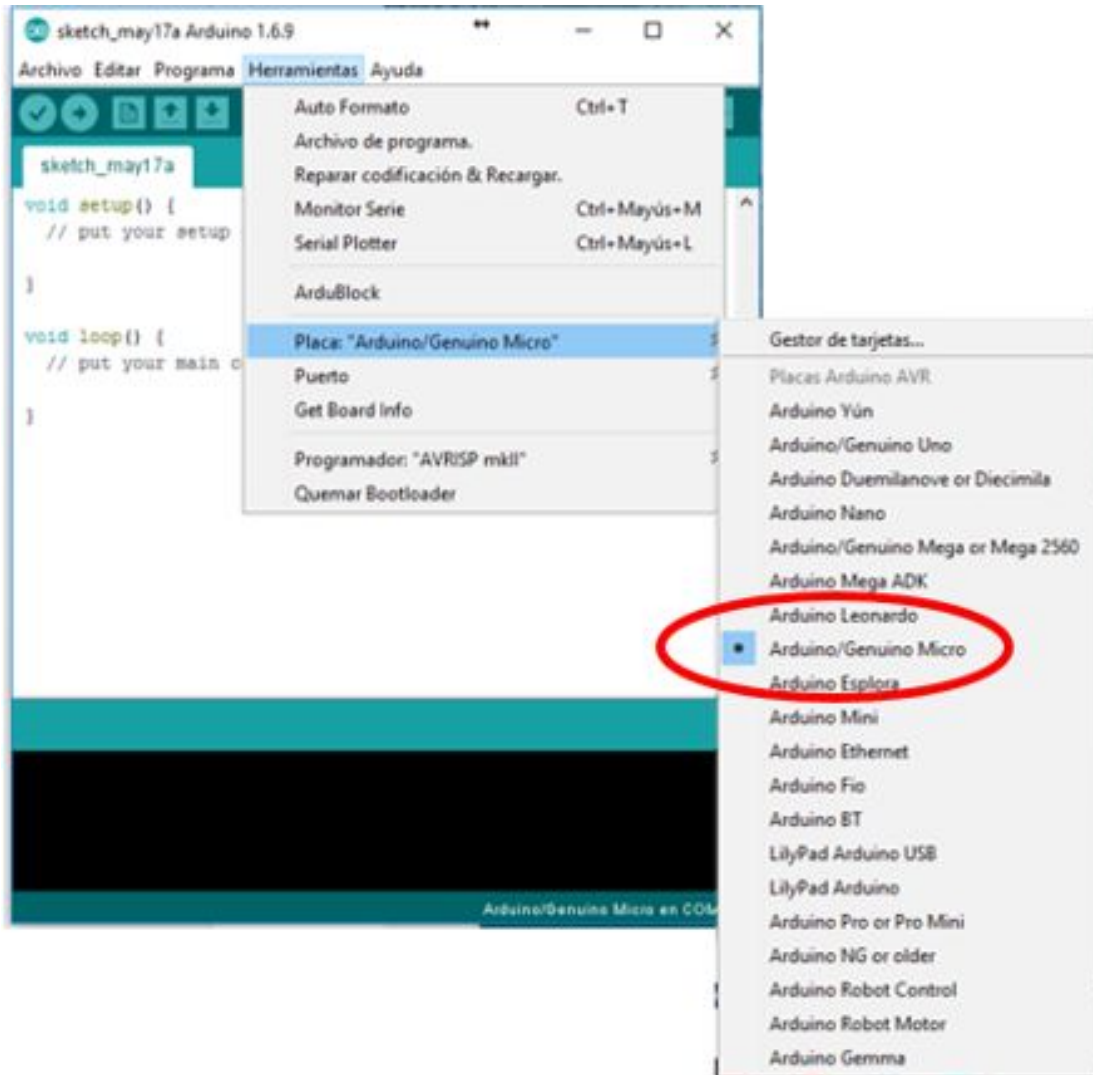
La primera vez deberás esperar unos instantes para que la computadora instale los Drivers.

Si el ladrillo está apagado después de enchufarlo, prenderlo.



R8

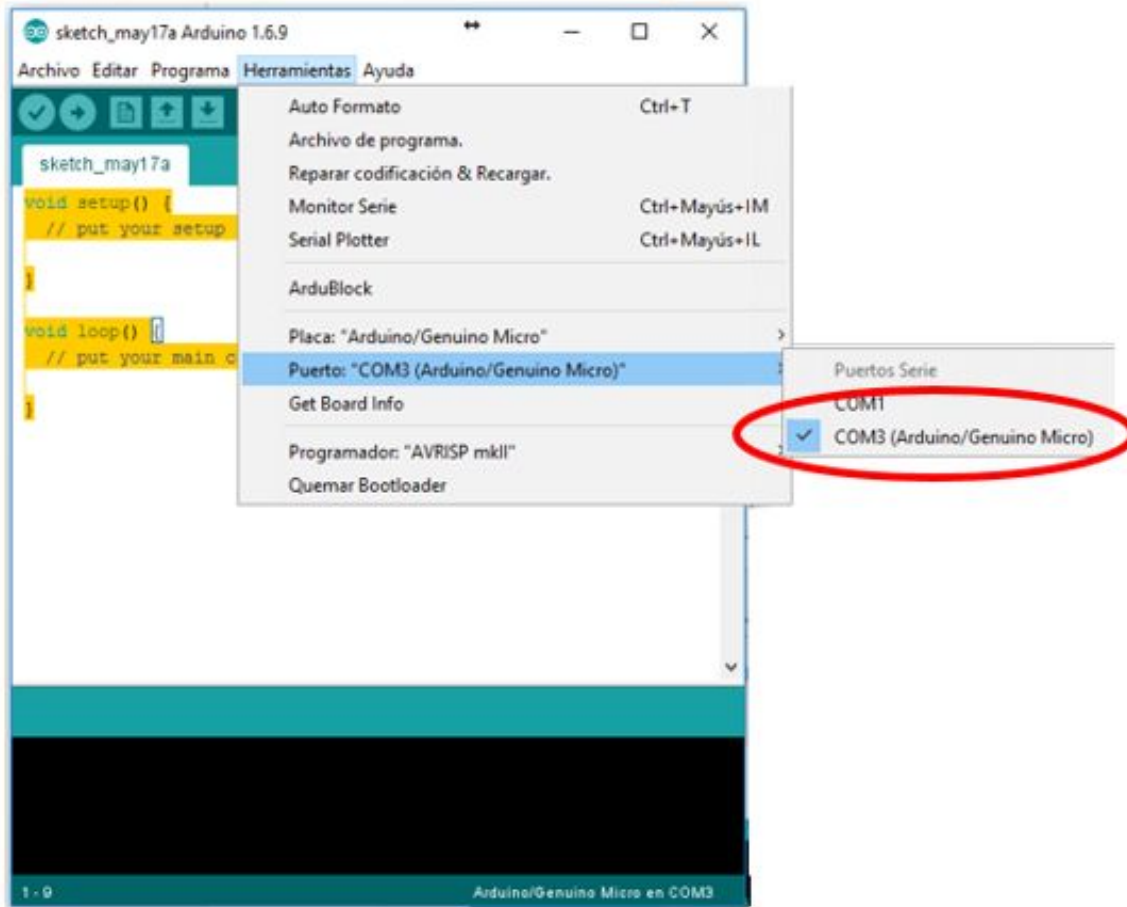
Después seleccionar: Herramientas/Placa/Arduino Genuino Micro



Ya casi concluimos, sólo nos falta verificar el puerto con al que se conecta.

## R8

Abrir Herramientas/Puerto/ y seleccionar el puerto donde aparezca la placa Genuino micro, si es que no está seleccionado ese puerto.



Listo! Está listo para que se le puedan cargar códigos de arduino.



Para más información sobre como programar en Arduino:

<https://www.arduino.cc/en/Tutorial/HomePage> .También hay otros sitios en español que se pueden consultar en internet.

Como este es un producto dirigido a chicos, el código es muy complejo y por eso optamos por una interfaz más gráfica en donde adaptamos el programa Ardublock a la estructura de piezas de Mis ladrillos.

## Instalación ArduBlock

ArduBlock es un entorno de programación gráfica para Arduino.

Es un entorno visual muy intuitivo y sencillo donde los distintos bloques con los que se realizan los programas encajan entre sí como si estuviésemos encastrando piezas de puzzles.

Una de las ventajas más significativas de ArduBlock es que además de programar mediante bloques, podemos conocer el código real, que se ejecuta en segundo plano en lenguaje Arduino.

Esta característica nos ayuda a aprender código Arduino para cuando queramos dar el salto de los entornos de programación gráficos a entornos de programación con código fuente.

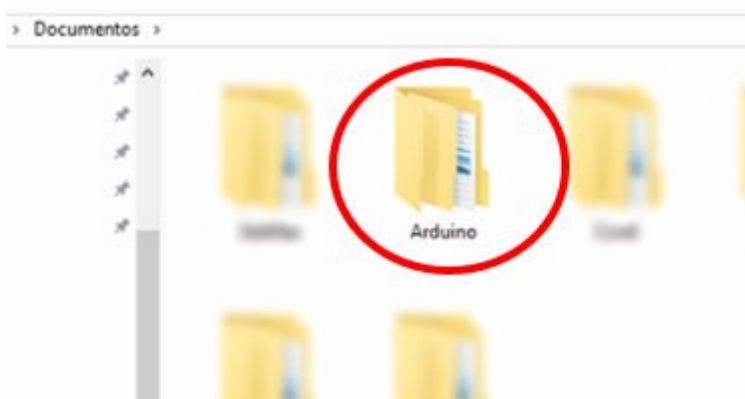
Es una muy buena herramienta para dar los primeros pasos en el mundo de la programación.

### Instalación de Ardublock

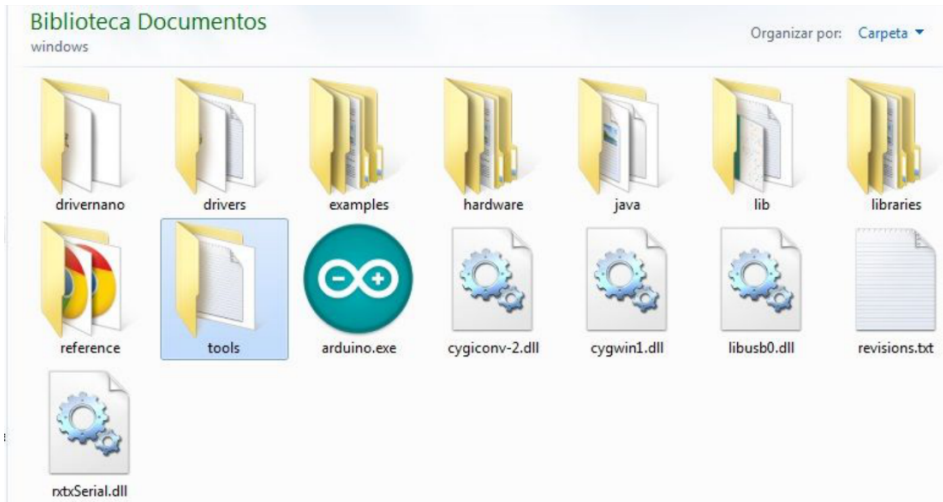
A continuación cerraremos el programa de arduino. El siguiente paso es descargar el archivo ardublock-all.jar que contiene la versión ArduBlock para Mis Ladrillos desde el siguiente link:

<http://www.edubots.com.ar/ardublock/ardublock-all.jar>

Una vez que tenemos el archivo .jar, buscamos la carpeta de Arduino que tenemos en nuestra computadora, normalmente dentro de “Documentos”.



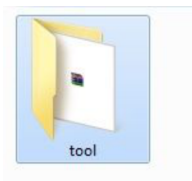
Dentro de ésta, ingresamos a una subcarpeta llamada "tools", como se muestra en la siguiente imagen.



Dentro de la carpeta "tools", creamos una carpeta con el nombre "ArduBlockTool".



Dentro de esta última carpeta creada, "ArduBlockTool", creamos una subcarpeta con el nombre "tool".



En el interior de esta última carpeta, guardamos el archivo .jar.

R8

---

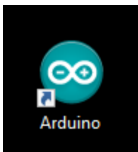


En resumen:

...../documentos/Arduino/tolos/ArduBlockTool/tool/ardublock-all.jar

En este punto ya tenemos instalado Ardublock.

Ahora procedemos a abrir el software de Arduino haciendo doble click en el ícono del Escritorio (o Inicio /programas)



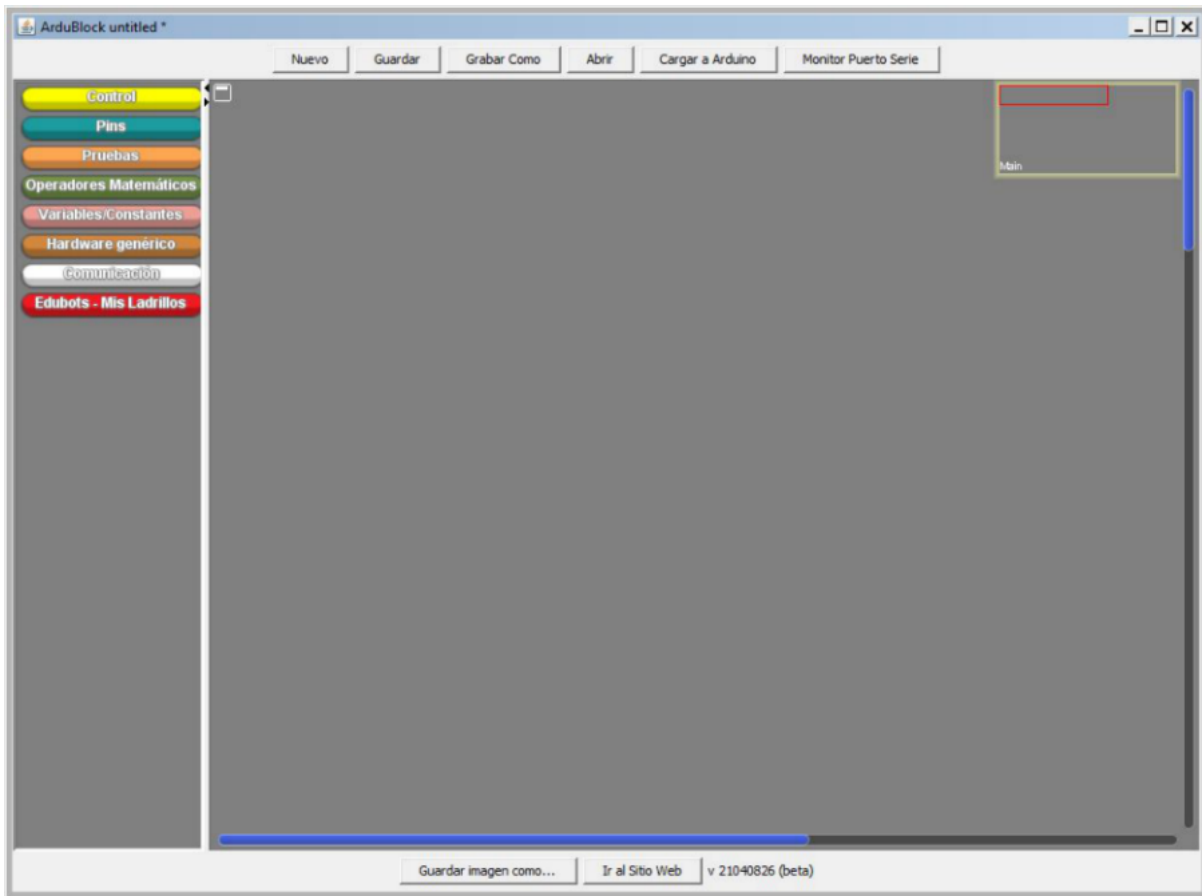
Al abrir el IDE de Arduino y hacer click sobre "Herramientas", aparecerá ArduBlock, como se muestra a continuación.



## R8

---

Hacemos click sobre ArduBlock y ya se abre el entorno de programación gráfico.



Para más información sobre cómo usar el ArduBlock, leer el manual de ArduBlock ([Tutorial ArduBlock - Kit R8.pdf](#))

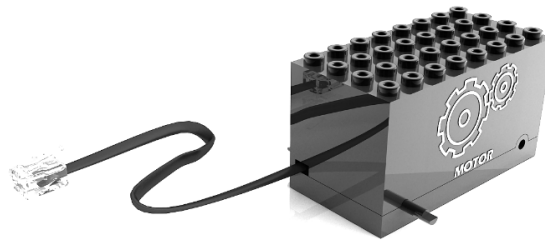


## Periféricos R8

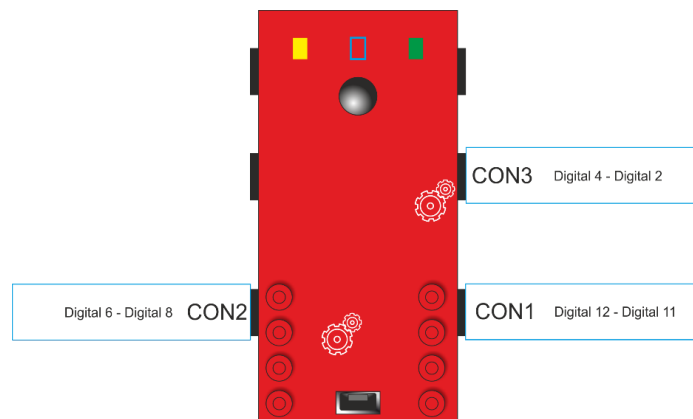
Ahora explicamos los periféricos del ladrillo R8. El kit R502 y R503 tienen varios periféricos.



### Motores



Se pueden conectar en cualquier salida con doble salida digital. Recomendamos las salidas CON1 (Digital 11 y 12) CON2 (digital 6 y 8). Eventualmente se puede conectar otro motor en CON3 (digital 4 y 5) pero no suele usarse ya que el kit viene con 2 motores.



Los motores se alimentan siempre por medio de un mosfet, poniendo un puerto digital en alta y otro en baja se logra un sentido, e invirtiendo estos puertos se cambia el sentido. Hay que evitar poner los dos puertos en alta para evitar posibles roturas. Para apagarlo se ponen los dos puertos en baja. En el caso de usar ArduBlock esto está resuelto más fácilmente, ver Tutorial ArduBlock - Kit R8.pdf

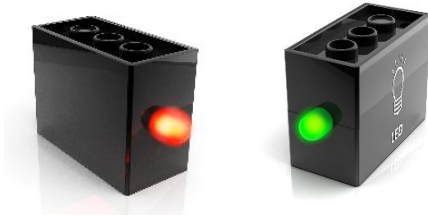
Otra recomendación para los motores es que si se programan cambios bruscos, es decir 100% de velocidad para adelante y 100% velocidad para atrás, pueden surgir esporádicamente picos de tensión que pueden resetear el ladrillo. Para filtrar esos picos

R8

hace falta -en el caso de usar código Arduino puro- agregarle una pequeña espera (unos 50 milisegundos) que sirve para drenar el pico y no se nota en el funcionamiento del kit. Si se usa ArduBlock de Mis Ladrillos no hace falta poner nada.



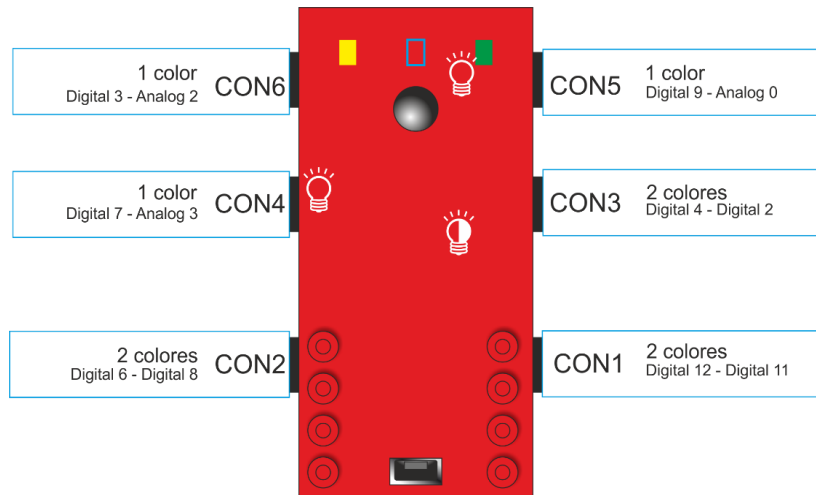
## Led bicolor (ó 1 sólo color)



El led bicolor tiene la posibilidad de tener color verde, color rojo o mezclar ambos colores, que da una tonalidad naranja. Para conectarlo usando las salidas digitales recomendamos usar el CON3 para tener dos colores (para dos colores también sirven CON1 CON2).

En los puertos CON4, CON5, CON6 va a funcionar a un color como encendido digital.

Básicamente consiste en encender una salida digital u otra. ( `digitalWrite(2 , HIGH); digitalWrite(4 , LOW);` )

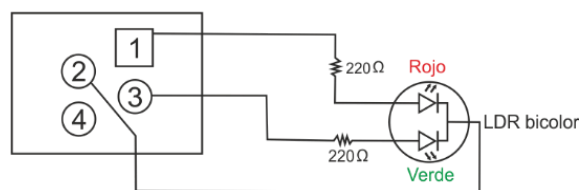


A continuación mostramos un esquema de cómo está armada la electrónica del sensor

### LUZ LED BICOLOR



### Rj9



R8

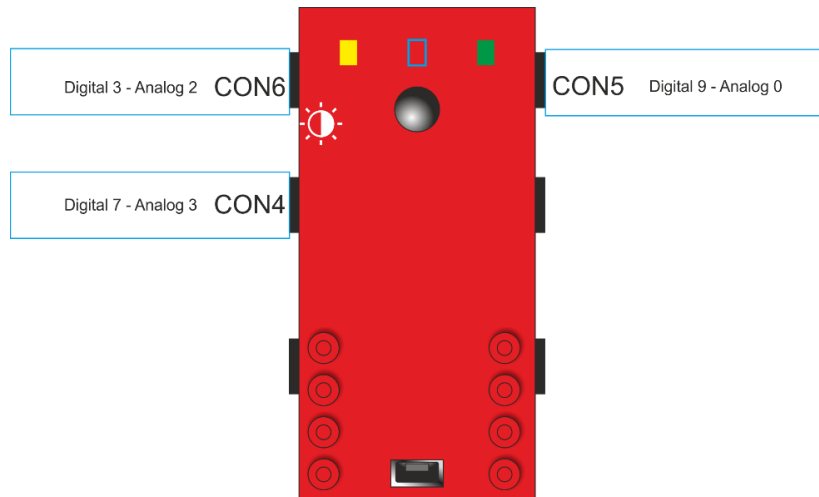


## Sensor de luz o LDR

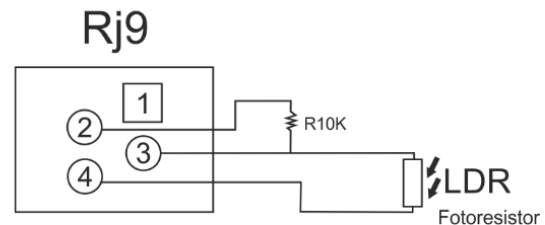


El Sensor de luz LDR o fotoresistor sirve para detectar y medir la intensidad de luz del medio ambiente. Es un sensor muy pequeño y sencillo de utilizar. Una fotoresistencia LDR es sensible a la luz que recibe y ofrece una resistencia mayor o menor en función de la cantidad de luz que recibe. Son sensores ideales para proyectos de iluminación o cualquier proyecto que necesite detectar la luz ambiente.

Se utiliza conectado a los puertos analógicos CON 5, CON4, CON6 del KIT.



Esquema de la electrónica del sensor  
SENSOR DE LUMINOSIDAD LDR



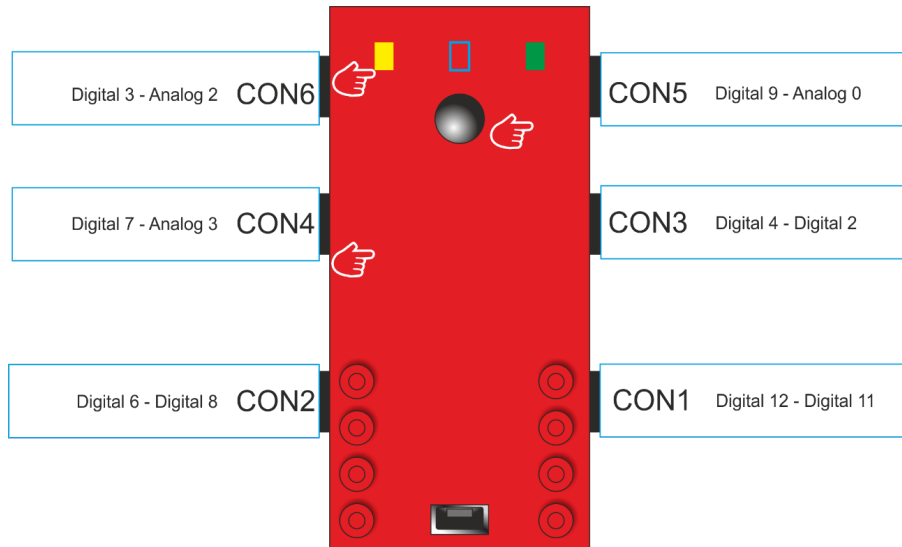


## Sensor de tacto o Switch



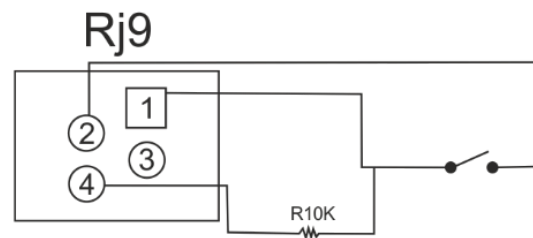
El llamado "sensor de tacto" es un pulsador que se activa presionando la ruedita hacia el lado de la carcasa.

Se puede conectar en cualquier puerto ya que usa una sola salida digital.




Esquema de la electrónica del sensor

### SWITCH



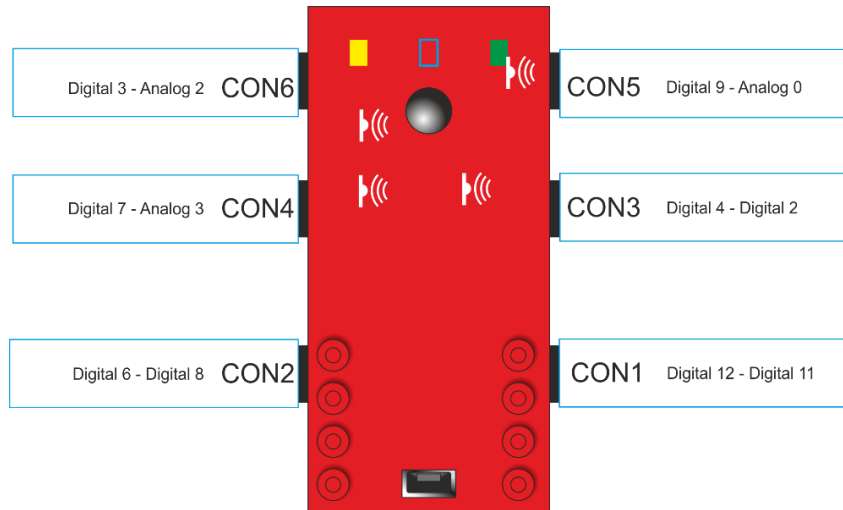
R8



# Sensor CNY70

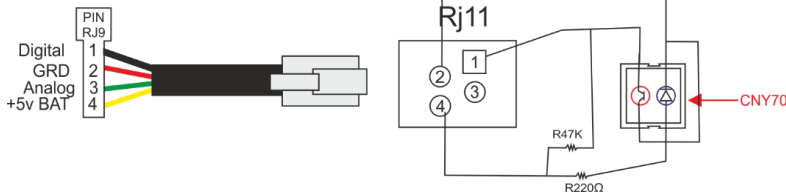


El CNY70 es un sensor de infrarrojos de corto alcance basado en un emisor de luz y un receptor, ambos apuntando en la misma dirección, y cuyo funcionamiento se basa en la capacidad de reflexión del objeto, y la detección del rayo reflejado por el receptor. El CNY70 devuelve por la pata de salida correspondiente, según el montaje, un voltaje relacionado con la cantidad de rayo reflejado por el objeto. Se leerá del emisor un '1' cuando se refleje luz y un '0' cuando no se refleje. Este sistema es el que se emplea para distinguir entre blanco y negro, en la conocida aplicación del robot seguidor de línea. Se puede conectar en cualquier salida que tenga un pin digital



## Esquema electrónico

DISTANCIA INFRAROJO CNY70



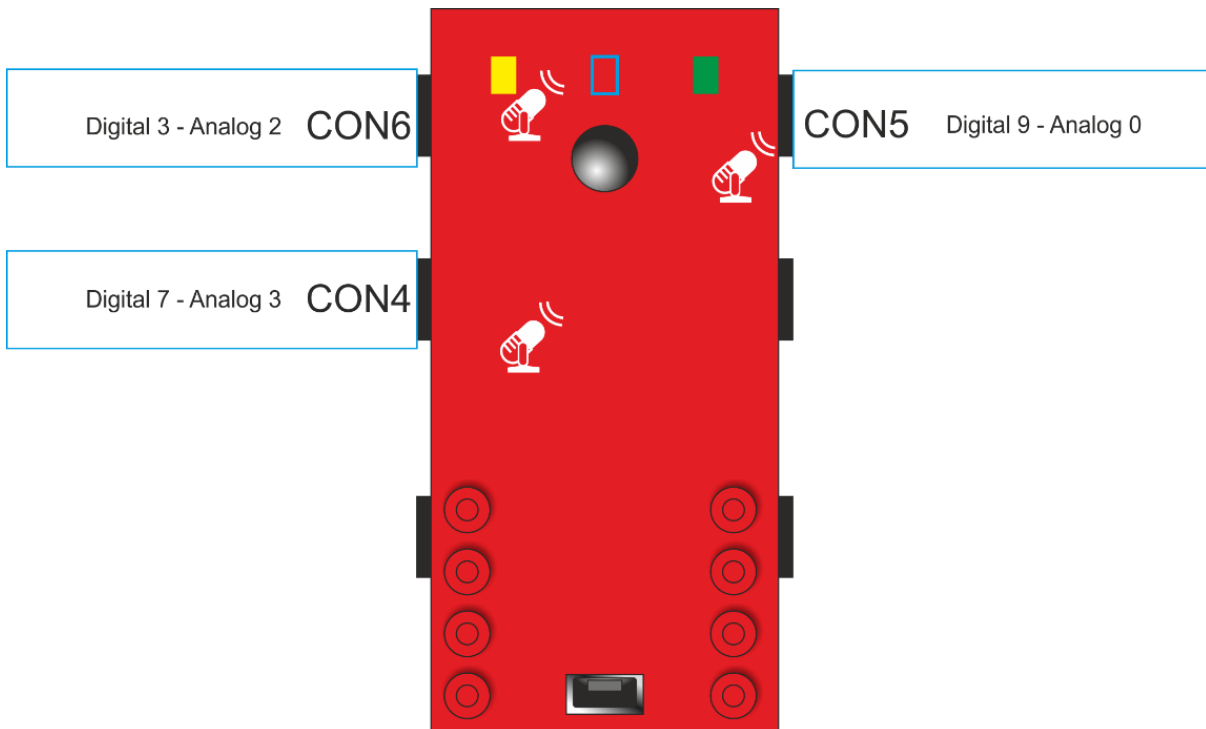


## Micrófono



El Kit tiene un sensor “micrófono” que detecta sonidos bruscos como aplausos, chasquido de los dedos etc. Tiene un filtro que minimiza sonidos “ruido” del entorno. Sirve para transmitir una orden al robot mediante este tipo de sonidos “secos”.

La lectura se hace por la entrada analógica, por lo tanto se enchufa en cualquier puerto con conexión analógica: CON4, CON5, CON6. Da lecturas analógicas entre 0 y 1023 donde suele dar un valor mínimo (alrededor de 50, depende del entorno) al estar activo por los ruidos del entorno. Cuando detecta el sonido seco pega un salto considerable y detectable por el programa. Por nombrar un ejemplo, de una lectura de 50 pasa a una lectura de 300 o más con algún golpe seco, silbido aplauso etc.







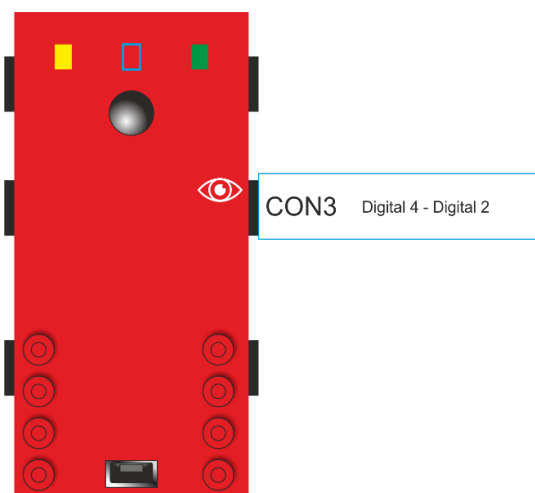
## Sensor de Distancia



Este es un sensor de distancia por ultrasonido basándose en un HC-SR04 usualmente usado en Arduino. El sensor de distancia ultrasónico es una excelente opción si estás buscando un módulo ultrasónico potente pero económico que sea compatible con Arduino y otras plataformas de microcontroladores. Su rendimiento estable es incomparable: la resolución es tan baja como de 3mm, tiene alta precisión, funciona con 5V (DC), es altamente compatible con las tarjetas Arduino.

### Características:

- Voltaje de funcionamiento: 5V (DC)
- Ángulo Eficaz: <math><15^\circ</math>
- Distancia de detección: 2 cm - 450 cm
- Resolución: 0,3 cm
- Medición de ángulo:  $30^\circ$
- Disparo de la señal de entrada: TTL pulso  $10\mu\text{s}$
- Echo señal de salida: señal PWL de TTL

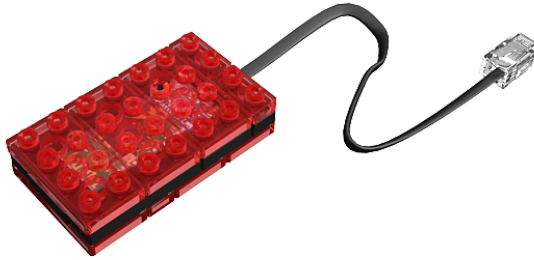


R8

---



# Bluetooth

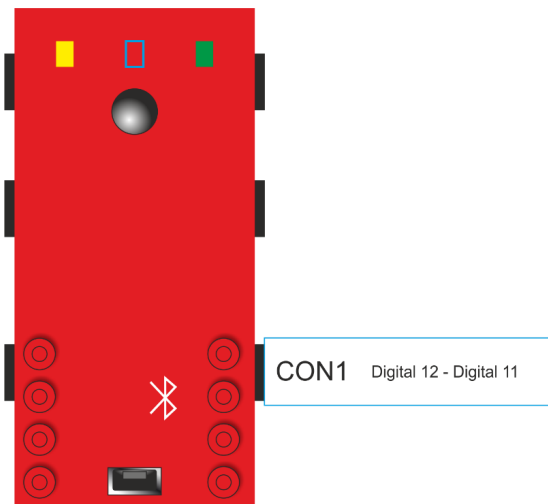


Con el módulo bluetooth HC-05 se puede conectar el ladrillo R8 a un teléfono, tablet, Pc, etc. El módulo bluetooth HC-05 viene configurado de fábrica para trabajar como maestro o esclavo. En el modo maestro puede conectarse con otros módulos bluetooth, mientras que en el modo esclavo queda a la escucha de peticiones de conexión. Agregando este módulo a tu proyecto podrás controlar a distancia desde un celular o una laptop todas las funcionalidades que desees.



El módulo BlueTooth HC-05 utiliza el protocolo UART RS 232 serial.

Es ideal para aplicaciones inalámbricas. Para conectar este módulo, enchufe en CON 1





## R8

---

Descargar alguna app para bluetooth al teléfono o la tablet.

Recomendamos para empezar usar esta app: Bluetooth Terminal HC-05

<https://play.google.com/store/apps/details?id=project.bluetoothterminal&hl=es-419>

Abra el software de Arduino y descargue el siguiente código al Ladrillo R8:

```
#include <SoftwareSerial.h> // Incluimos la librería SoftwareSerial  
SoftwareSerial BT(11,12); // Definimos los pines RX y TX del Arduino conectados al  
Bluetooth  
void setup()  
{  
  BT.begin(9600); // Inicializamos el puerto serie BT (Para Modo AT 2)  
  Serial.begin(9600); // Inicializamos el puerto serie  
}  
void loop()  
{  
  if(BT.available()) // Si llega un dato por el puerto BT se envía al monitor serial  
  {  
    Serial.write(BT.read());  
  }  
  if(Serial.available()) // Si llega un dato por el monitor serial se envía al puerto BT  
  {  
    BT.write(Serial.read());  
  }  
}
```

Ahora abre la consola serie en arduino (lupa arriba a la derecha)

# R8

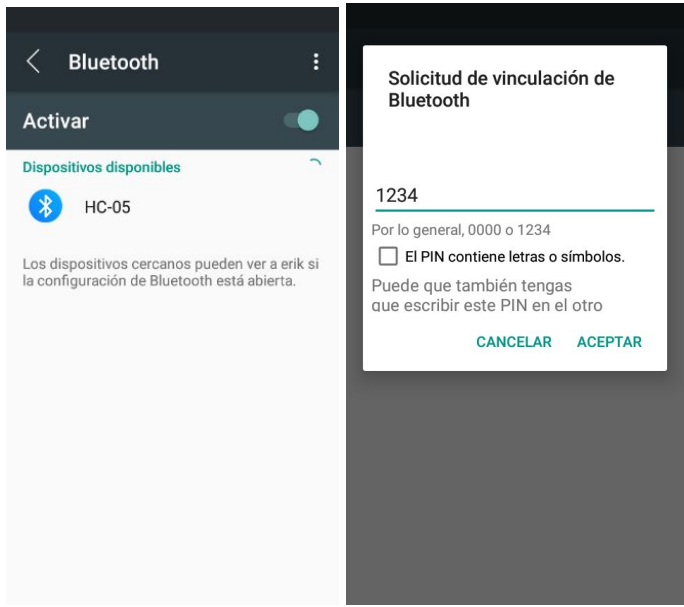
```
BT_conf_AT_HC-05
Archivo Editar Programa Herramientas Ayuda

BT_conf_AT_HC-05
1 #include <SoftwareSerial.h> // Incluimos la libreria SoftwareSerial
2 SoftwareSerial BT(11,12); // Definimos los pines RX y TX del Arduino conectados al Bluetooth
3
4 void setup()
5 {
6   BT.begin(9600); // Inicializamos el puerto serie BT (Para Modo AT 2)
7   Serial.begin(9600); // Inicializamos el puerto serie
8 }
9
10 void loop()
11 {
12   if(BT.available()) // Si llega un dato por el puerto BT se envia al monitor serial
13   {
14     Serial.write(BT.read());
15   }
16
17   if(Serial.available()) // Si llega un dato por el monitor serial se envia al puerto BT
18   {
19     BT.write(Serial.read());
20   }
21 }
22

Subido
avrdude done. Thank you.

Arduino/Genuino Micro en COM4
```

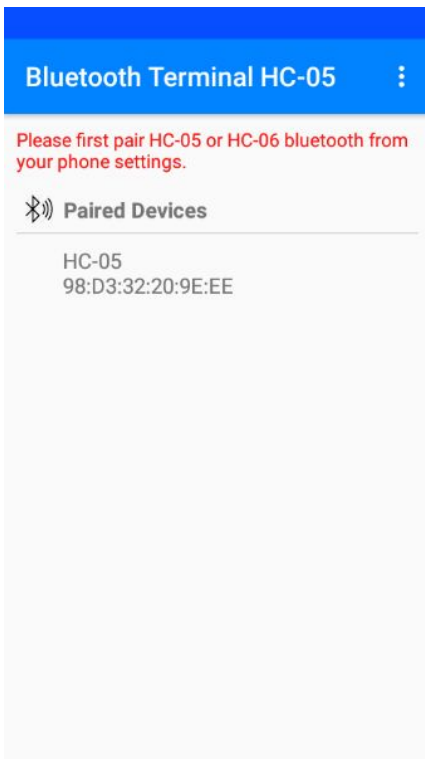
Vaya al teléfono / tablet para vincularlo con el dispositivo. Dentro del teléfono entra en el ícono Bluetooth y pon “Buscar dispositivos”. Cuando aparezca el dispositivo llamado HC-05, conectarse. Va a pedir la contraseña que es 1234 por defecto.



Ve a la App Bluetooth Terminal HC-05 y ábrela, conéctate al dispositivo

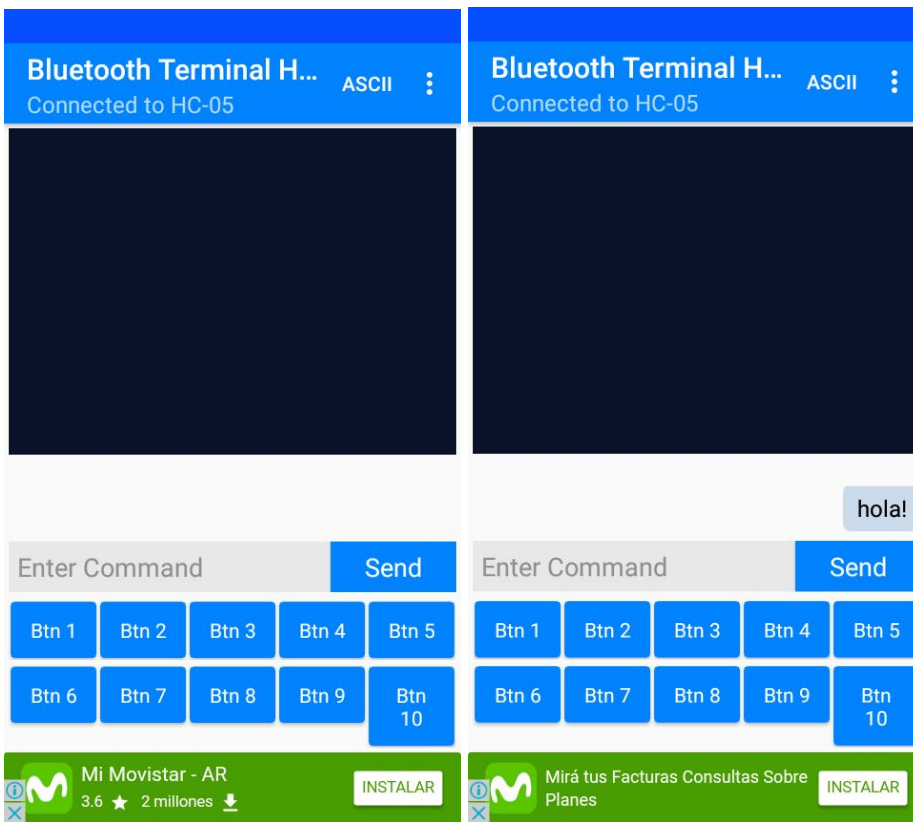


Después de elegir dispositivo:

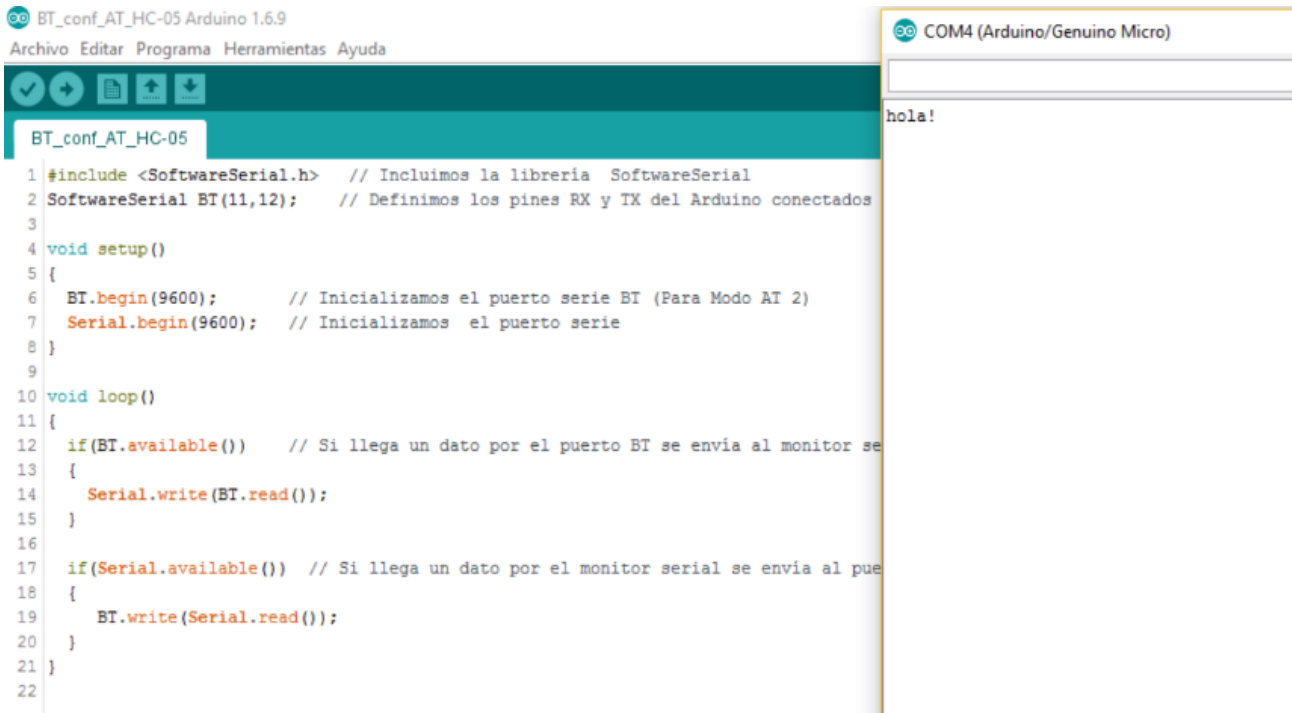


Una vez conectados tocamos a donde dice Enter Command y escribimos algún texto (poner el ejemplo: "Hola!") y apretamos send.

# R8



Este texto va a ser enviado vía bluetooth a la consola del Arduino:

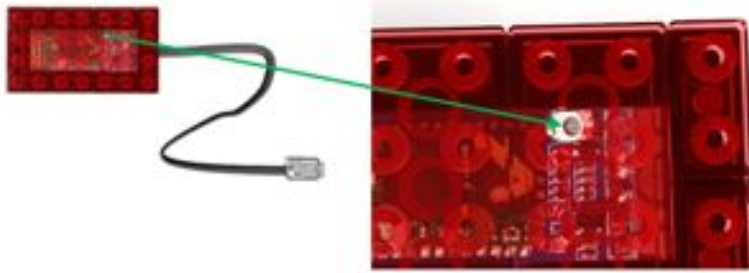


Estos “mensajes” pueden comandos que después se usan en algún programa de arduino.



### Como cambiar el nombre y contraseña del dispositivo Bluetooth

Para cambiar el nombre y contraseña del dispositivo se hace mediante comandos AT. Hay varios métodos, pero vamos a explicar ahora el método usado con la consola Serie de arduino. Desenchufa el dispositivo Bluetooth, aprieta el botón con algún objeto con punta. Enchúfalo mientras mantienes apretado el botón, espera unos segundos y después puedes soltarlo. El led va a titilar espaciadamente. Significa que está en modo AT.



Vamos ahora al software de arduino y pegamos el código que ponemos a continuación:

```
#include <SoftwareSerial.h>  
SoftwareSerial mySerial(11, 12); // RX, TX  
void setup() {  
  Serial.begin(9600);  
  delay(1000);  
  mySerial.begin(38400);  
}  
void loop()  
{  
  if (mySerial.available())  
    Serial.write(mySerial.read());  
  if (Serial.available())  
    mySerial.write(Serial.read());  
}
```

Ahora vamos a la consola Serie de arduino (lupa arriba derecha)

#### Hacemos un test de comunicación:

Lo primero es comprobar si nuestro bluetooth responde a los comandos AT

Enviar: AT

Recibe: OK

Si recibimos como respuesta un OK entonces podemos continuar, sino verificar las conexiones o los pasos anteriores.

## R8

---

Por defecto nuestro bluetooth se llama "HC-05" esto se puede cambiar con el siguiente comando AT

Enviar: AT+NAME=<Nombre> Ejm: AT+NAME=MISLADRILLOS

Respuesta: OK

Por defecto viene con el código de vinculación (Pin) "1234". Para cambiarlo hay que enviar el siguiente comando AT

Enviar: AT+PSWD=<Pin> Ejm: AT+PSWD=0000

Respuesta: OK

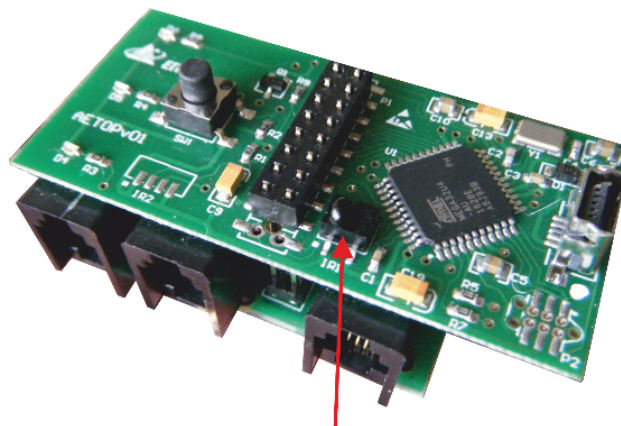
Se puede saber cuál es el pin actual de nuestro módulo, para eso hay que enviar el siguiente comando: AT+PSWD?

Ahora reiniciarlo nuevamente y trata de vincularlo al Teléfono/tablet con los nuevos datos y nuevo nombre.

## Sensores incluidos en la placa

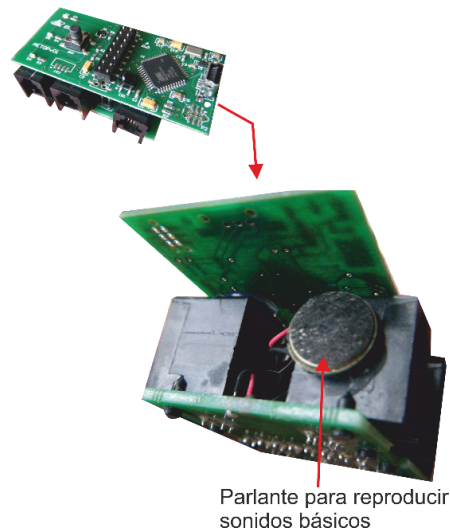
### Receptor IR

En el pin **digital 5** hay conectado un receptor infrarrojo que permite captar señales de controles remotos. Aquí mostramos al ladrillo R8 sin su carcasa a modo ilustrativo (no se puede abrir normalmente)



LDR - Fotorresistor  
Sensor que detecta Infrarojos

### Mini parlante buzzer



En el pin **digital 10** hay conectado un mini parlante que permite reproducir sonidos o melodías básicas.

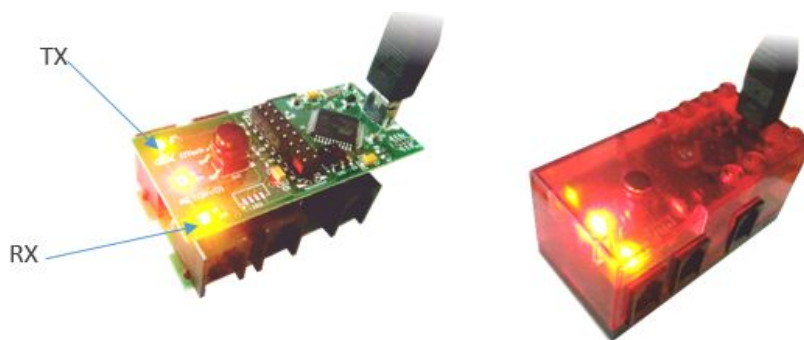
## LEDs de la placa

La placa tiene 3 Led que se iluminan con la comunicación USB.

Estos son configurables para encendido o apagado a gusto.

- El LED verde es el de comunicación TX conectado al Puerto Digital 30
- El LED amarillo es el de comunicación RX conectado al Puerto Digital 17
- El LED rojo central no tiene muestra cuando se esta reiniciando el dispositivo. Está conectado al Puerto Digital 13.

Estos 3 LED en el caso de usarse con código nativo Arduino, tienen función inversa, es decir HIGH funciona como LOW, y LOW funciona como HIGH.





## Batería recargable

El juego viene con una batería de Litio con cargador y regulación de tensión incorporado. Para cargarla hay que enchufar el conector Micro USB a algún cargador de teléfono/tablet o a la PC con el cable provisto para ello.

Tiene un LED indicador de carga:



*Verde intermitente:* Batería cargando.

*Verde fijo:* Carga completa.

*Rojo fijo:* Batería disponible.

R8

---

*Rojo intermitente: Batería descargada. Conectar a un cargador.*

**Información técnica:**

Batería interna de Li-ion 3.7V 1500mAh

Con protección electrónica y cargador incorporados.

*Salida:* 5V 2Amax. (Conector tipo Mini-USB)

*Entrada:* 5V 1Amax. (Conector tipo Micro-USB)

Rango de Temperatura:

*En uso:* -20°C a +60°C,

*Durante la carga:* 0°C a +45°C.

*Almacenamiento:* 20°C a 25°C

**Precauciones de Seguridad para baterías de Litio**

Por favor lea cuidadosamente las precauciones de seguridad en el manejo de baterías de Litio-ion. El manejo de baterías en forma incorrecta puede provocar daños o lesiones a causa de derrame de líquido electrolito, el recalentamiento, e incluso, si fallaran todas las protecciones, puede provocarse una explosión de la batería.

- Nunca caliente o incinere la batería.
- Nunca perforo, golpee o cause daños mecánicos a la batería.
- Nunca cargue la batería en altas temperaturas (mayores a 60°C).
- Nunca cargue la batería en bajas temperaturas (menores a 0°C).
- Nunca coloque en cortocircuito a la batería.
- Nunca permita que la batería se moje o quede inmersa en agua.

En caso de notar un derrame de líquido electrolito desde dentro de la batería, no tocar el líquido. Limpiar el área con un paño seco evitando el contacto con la piel. En caso de contacto con la piel, lavar con abundante agua, y si se produce irritación de la piel, consultar con un médico.

# R8

Más detalles de la placa R8

Mis Ladrillos R8 - PIN MAPPING TABLE

| Pin | Nombre del PIN              | Nombre del PIN Mapeado   | Mis Ladrillos R8        | Pin | Nombre del PIN                | Nombre del PIN Mapeado | Mis Ladrillos R8         |
|-----|-----------------------------|--------------------------|-------------------------|-----|-------------------------------|------------------------|--------------------------|
| 1   | PE6 (INT_6/AIN0)            | Digital pin 7            | CON4 - SWI2 DER - PIN1  | 23  | GND1                          | GND                    | GND                      |
| 2   | V <sub>CC</sub>             | +5V                      | USB/BAT +5V             | 24  | AVCC                          | AVCC                   | +5V                      |
| 3   | D-                          | RD-                      | USB - D-                | 25  | (ICP1/ADC8) PD4               | Digital pin 4          | CON3 - LED1 IZQ - PIN3   |
| 4   | D+                          | RD+                      | USB - D+                | 26  | (T1/#OC4D/ADC9) PD6           | Digital pin 12         | CON1 - MOTOR IZQ - PIN3  |
| 5   | V <sub>CC</sub>             | UGND                     | USB/BAT GND             | 27  | (T0/OC4D/ADC10) PD7           | Digital Pin 6 (PWM)    | CON2 - MOTOR DER - PIN1  |
| 6   | V <sub>CC</sub>             | UCAP                     |                         | 28  | (ADC11/PCINT4) PB4            | Digital pin 8          | CON2 - MOTOR DER - PIN3  |
| 7   | V <sub>USB</sub>            | V <sub>Bus</sub>         |                         | 29  | (PCINT5/OC1A/#OC4B/ADC12) PB5 | Digital Pin 9 (PWM)    | CON5 - SW1 IZQ - PIN1    |
| 8   | (SS/PCINT0) PB0             | RXLED                    | LED AMARILLO            | 30  | (PCINT6/OC1B/#OC4B/ADC13) PB6 | Digital Pin 10 (PWM)   | LED ROJO                 |
| 9   | (PCINT1/SCLK) PB1           | SCK                      |                         | 31  | (OC3A/#OC4A) PC6              | Digital Pin 5 (PWM)    | RECEPTOR INFRAROJO       |
| 10  | (PDI/PCINT2/MOSI) PB2       | MOSI                     |                         | 32  | (ICP3/CLK0/#OC4A) PC7         | Digital Pin 13 (PWM)   | PARLANTE                 |
| 11  | (PDO/PCINT3/MISO) PB3       | MISO                     |                         | 33  | (#HWB) PE2                    | HWB                    |                          |
| 12  | (PCINT7/OC0A/OC1C/#RTS) PB7 | Digital pin 11 (PWM)     | CON1 - MOTOR IZQ - PIN1 | 34  | V <sub>CC</sub>               | +5V                    | +5V                      |
| 13  | RESET                       | RESET                    | TACT SWITCH SW1         | 35  | GND2                          | GND                    | GND                      |
| 14  | V <sub>CC</sub>             | +5V                      | +5 - CON1 a CON6 - PIN4 | 36  | (ADC7/TDI) PF7                | Analog In 0            | CON5 - MIC1 IZQ - PIN3   |
| 15  | GND                         | GND                      |                         | 37  | (ADC6/TDO) PF6                | Analog In 1            |                          |
| 16  | XTAL2                       | XTAL2                    |                         | 38  | (ADC5/TMS) PF5                | Analog In 2            |                          |
| 17  | XTAL1                       | XTAL1                    |                         | 39  | (ADC4/TCK) PF4                | Analog In 3            | CON6 - LDR DER - PIN3    |
| 18  | (OC0B/SCL/INT0) PD0         | Digital pin 3 (SCL)(PWM) | CON6 - FLED DER - PIN1  | 40  | (ADC1) PF1                    | Analog In 4            | CON4 - MIC2 DER - PIN3   |
| 19  | (SDA/INT1) PD1              | Digital pin 2 (SDA)      | CON3 - LED1 IZQ - PIN1  | 41  | (ADC0) PF0                    | Analog In 5            |                          |
| 20  | (RX D1/AIN1/INT2) PD2       | Digital pin 0 (RX)       |                         | 42  | AREF                          | AEF                    |                          |
| 21  | (TXD1/INT3) PD3             | Digital pin 1 (TX)       |                         | 43  | GND3                          | GND                    | GND - CON1 a CON6 - PIN2 |
| 22  | (XCK1/#CTS) PD5             | TXLED                    | LED VERDE               | 44  | AVCC1                         | AVCC                   |                          |



## ETech AETOPv02

### PINOUT DIAGRAM

