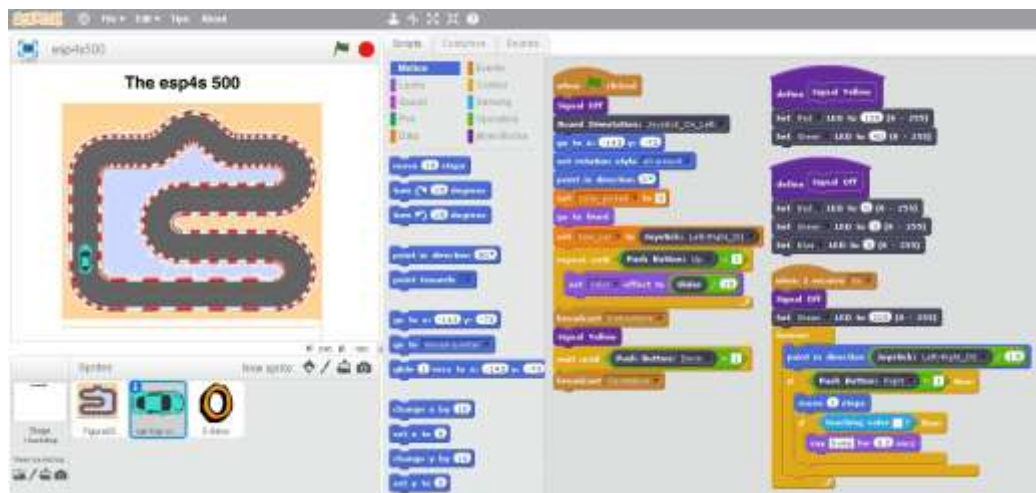


**esp4s**

# Una aplicación para control desde Scratch/Snap! para **Arduino Explora**



**Copyright © 2015 Alan Yorinks. All rights reserved.**

**This manual is distributed WITHOUT ANY WARRANTY, without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE**

Traducción y Adaptación al Español: José Manuel Ruiz Gutiérrez  
j.m.r.gutierrez@gmail.com

# Tabla de Contenidos

Introducción

Una visión general de la Plataforma Hardware

Sensores que incorpora la tarjeta Explora.

Una visión general de la Plataforma Software.

Instalación

esp4s

Python

pySerial

Arduino Sketch

Ejecutando esp4s

Usando Scratch

Proyectos Scratch Incluidos

Usando Snap!

Descripción de Bloques.

Configurara el Led de la tarjeta

Seleccionar y Configurar el brillo de Led RGB

Tocar Nota

Tocar Tono Continuo

Configura Nivel salida de TinkerKit

Orientación de la Tarjeta

Unidades d Temperatura.

Slider

Luz

Temperatura

Sonido

Pulsador

Joystick

Acelerómetro

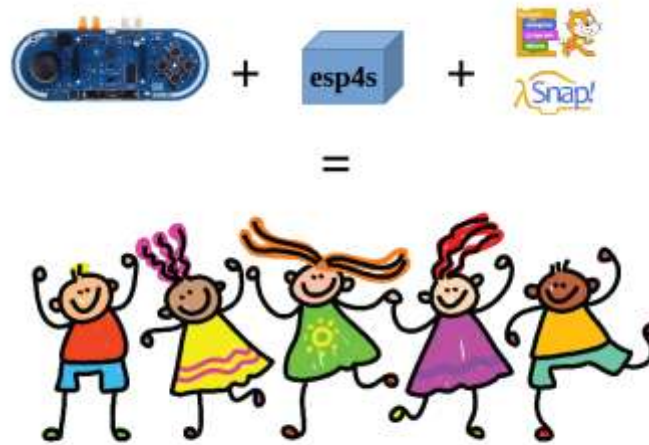
Entrada TinkerKit

Si necesita más Ayuda

Solución de Problemas

Otros Proyectos Open Source de Mr. Y's Lab

# Introducción



Para muchos, nada puede ser más divertido que escribir programas que controlan y supervisan los dispositivos físicos como joysticks, LED, sensores de sonido, sensores de temperatura y similares.

Pero antes que la primera línea de código pueda ser escrita, los circuitos deben ser diseñados, tienen que adquirirse los componentes y necesitan ser montados y probados.

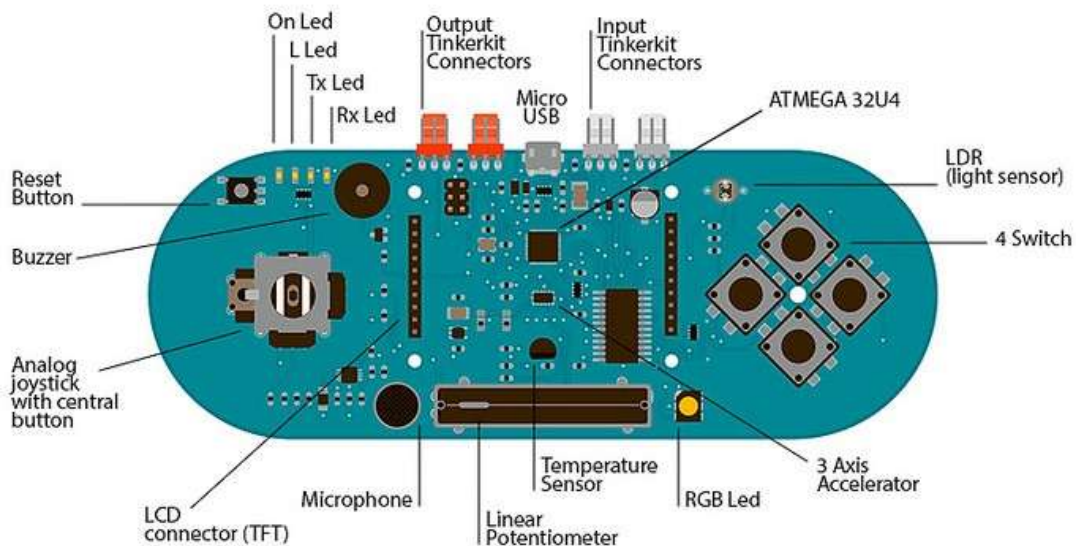
Además, a menudo se requiere un software personalizado para proporcionar una interfaz de programación para el hardware. Este software, como el hardware, tiene que ser diseñado, codificado y probado antes de su integración con el hardware.

El diseño y el montaje tanto el hardware como el software de juego puede ser bastante lento y no está exento de obstáculos.

¿Hay manera más rápida de escribir código? ¡Sí! Al combinar un microcontrolador Arduino Esplora con la librería esp4s y, o bien Scratch o bien Snap!. De este modo la *primera línea de código* puede ser escrita en minutos!

## Una visión general de la Plataforma Hardware.

La tarjeta Arduino Esplora integra sensores, actuadores y un microcontrolador todo en una sola tarjeta, por lo que es ideal para ponerse rápidamente en marcha. No hay nada que diseñar o probar! Sólo tiene que sacarlo de la caja, alimentarlo y listo. Esplora viene con un conjunto completo de sensores y actuadores que se enumeran a continuación.



## Sensores que incorpora la tarjeta Explora

- Joystick con pulsador central.
- 4 pulsadores.
- Control deslizante (potenciómetro lineal).
- Micrófono para el seguimiento de la amplitud del sonido del ambiente circundante.
- Sensor de luz medir los niveles de luz ambiente.
- Sensor de temperatura para controlar la temperatura ambiente.
- 3 ejes Acelerómetro para controlar el movimiento del dispositivo.
- 2 conectores TinkerKit para añadir sensores adicionales.

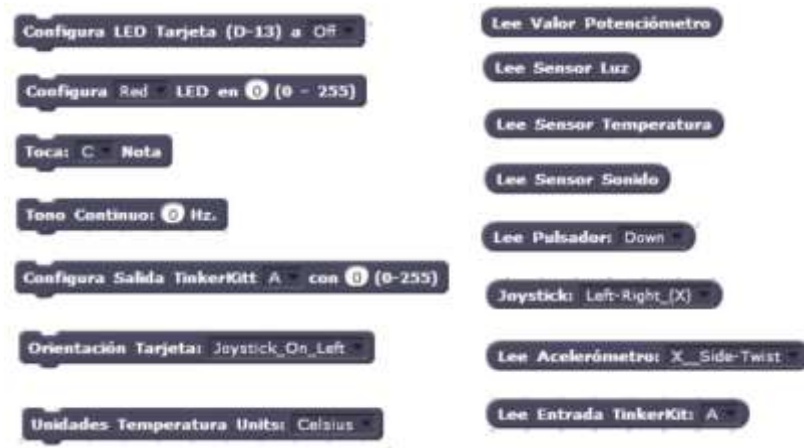
## Una visión general de la Plataforma Software

Tanto Scratch como Snap! son maravillosos lenguajes de programación gráfica que simplifican el trabajo de programación. Ambos lenguajes evitan la necesidad de hacer frente a problemas de sintaxis problemáticos. Como resultado, los programadores de forma rápida e intuitiva pueden emprender la tarea de la elaboración de programas. Pero todavía hay un problema a resolver. Ni Scratch ni Snap! vienen equipados con bloques de programación para controlar o supervisar la tarjeta Explora.

Y así nació **esp4s**. Es una extensión para Scratch y Snap! que agrega varios bloques de programación personalizados para Explora integrados en las librerías que incluye el editor de programas y también incluye un puente de comunicación para controlar y supervisar Explora.

A continuación se muestra una captura de pantalla de la serie completa de bloques de programación esp4s. Todos estos bloques se describen en detalle en la sección Descripción de bloque de este

documento.



Bloques de la librería esp4s para Scrtach

## Instalación

### esp4s

Descargue y descomprima la distribución **esp4s** en el directorio que desee.

### Python

Antes de utilizar la librería esp4s, es necesario tener instalado Python 2.7 en el equipo. Asegúrese de que usted tiene la versión 2.7 y no la versión 3. Versión 3 no es compatible con la librería. Puede descargar python desde: <http://python.org/> .

### pySerial

Después de instalar Python, el siguiente paso es instalar pySerial. Puede descargar la última versión de pySerial aquí: <http://pyserial.sourceforge.net/>

Siga las instrucciones de la página web para la instalación en su sistema operativo.

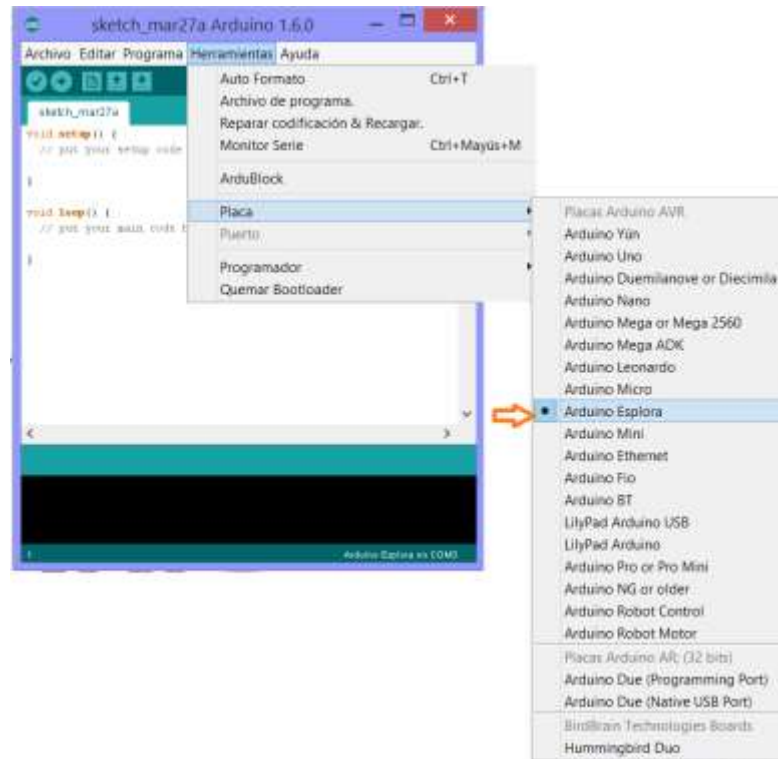
**Nota: es posible que tenga privilegios de administrador para realizar la instalación.**

### Arduino Sketch

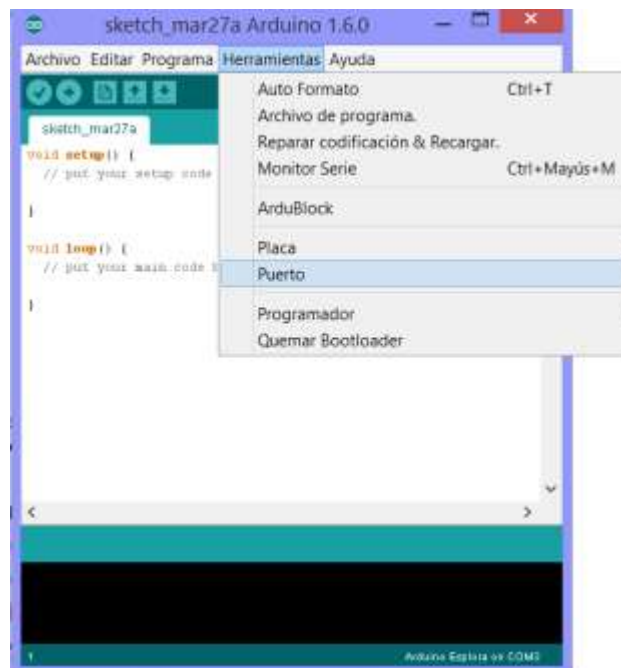
Primero tendrá que instalar el entorno de desarrollo integrado Arduino (IDE) de <http://arduino.cc/en/Main/Software>

Instale todo el software que incluye controladores USB si lo pida el programa de instalación de Arduino.

A continuación, abra la Aduino IDE. Seleccione Tools/Boards y luego seleccione Esplora en la lista desplegable.



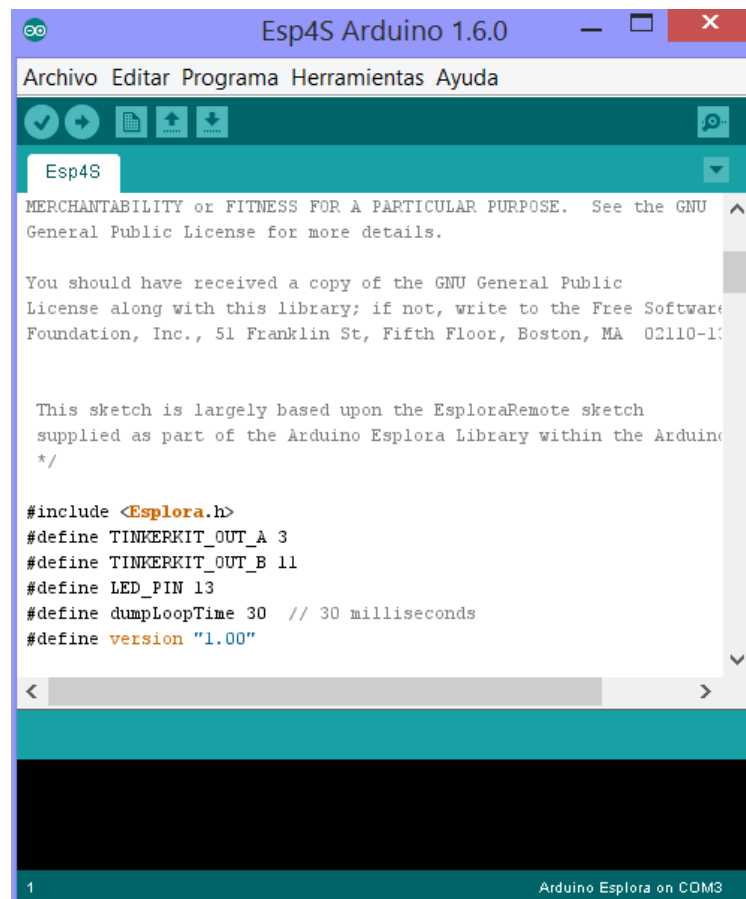
Luego, desde el menú Tools/Port, seleccione el puerto COM del ordenador. El IDE puede mostrar Leonardo como el identificador para el puerto.



Por último, descargue el sketch de Arduino al Explora haciendo lo siguiente :

- Seleccione **File/Open** y luego usando el selector de archivos, vaya al directorio en el que extrajo los archivos esp4s y seleccione Esp4s.ino en el directorio *Arduino\_Sketch /Esp4s*.
- Seleccione la (flecha hacia la derecha) botón Cargar. Cuando la carga se completa, cierre el IDE Arduino.

Sólo necesita hacer esto una vez, siempre y cuando no sobrescriba el sketch con otro sketch en algún otro momento.





# Ejecutando esp4s

**Si está en un entorno Linux.**

Abra una terminal de comandos y vaya al directorio que contiene los archivos extraídos esp4s.

Escriba: *esp4s.py python*



```
afy@afy-Z68A-D3H-B3: ~/PycharmProjects/esp4s
afy@afy-Z68A-D3H-B3:~$ cd PycharmProjects
afy@afy-Z68A-D3H-B3:~/PycharmProjects$ cd esp4s
afy@afy-Z68A-D3H-B3:~/PycharmProjects/esp4s$ python esp4s.py
esp4s version 1.0 Copyright(C) 2015 Alan Vorinks All Rights Reserved

Opening Esplora Serial port /dev/ttyACM0
Starting HTTP Server!
Use <Ctrl-C> to exit the extension

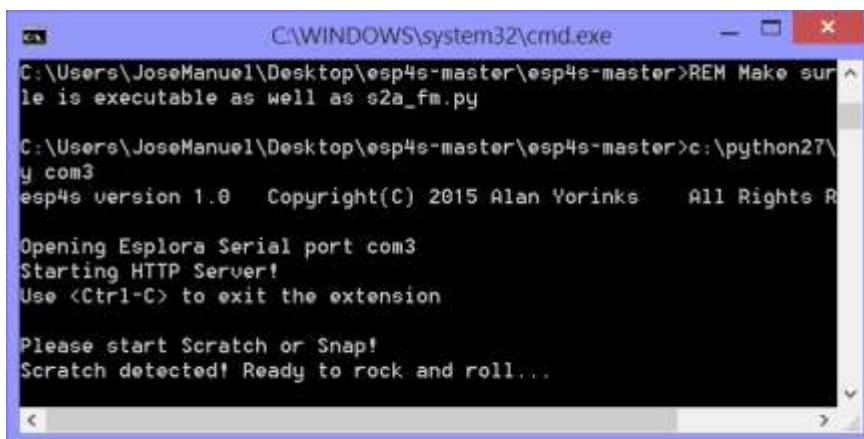
Please start Scratch or Snap!
```

**Si está en un entorno Windows.**

En este caso deberá ejecutar un fichero tipo "bat" de proceso por lotes que contenga las siguientes instrucciones:

*c:\python27\python esp4s.py com3*

Teniendo en cuenta que el parámetro "com3" se debe modificar poniendo el numero de COM por el que nos estamos comunicando con la tarjeta Arduino Explora y evidentemente también debemos poner bien el path en donde se encuentra instalado python 2.7



```
C:\WINDOWS\system32\cmd.exe
C:\Users\JoseManuel\Desktop\esp4s-master\esp4s-master>REM Make sur
le is executable as well as s2a_fm.py

C:\Users\JoseManuel\Desktop\esp4s-master\esp4s-master>c:\python27\
y com3
esp4s version 1.0 Copyright(C) 2015 Alan Vorinks All Rights R

Opening Esplora Serial port com3
Starting HTTP Server!
Use <Ctrl-C> to exit the extension

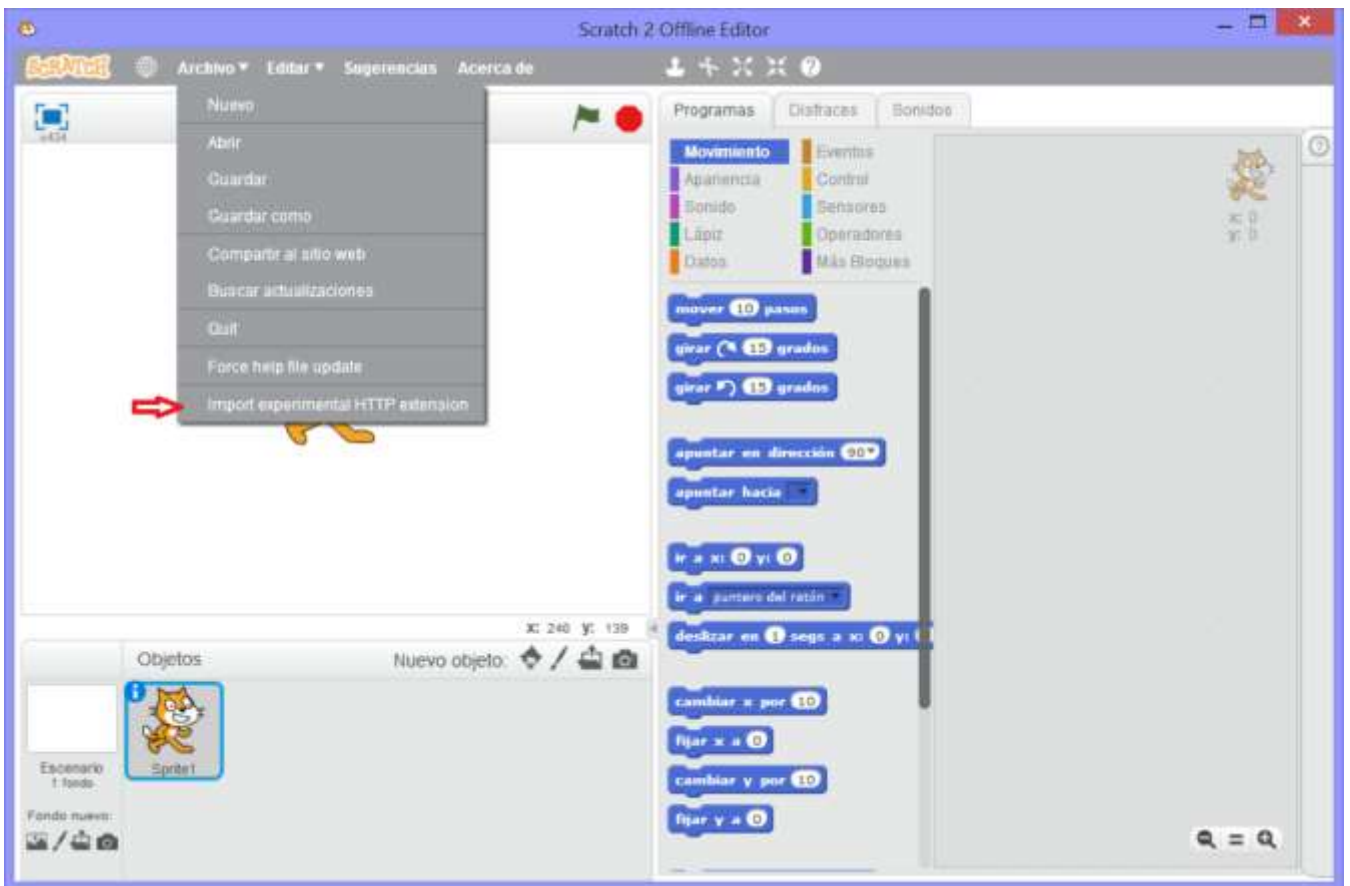
Please start Scratch or Snap!
Scratch detected! Ready to rock and roll...
```

En pocos segundos, debería ver el mensaje de bienvenida e instrucciones para iniciar la versión off-line de Scratch o la versión en línea de Snap !.

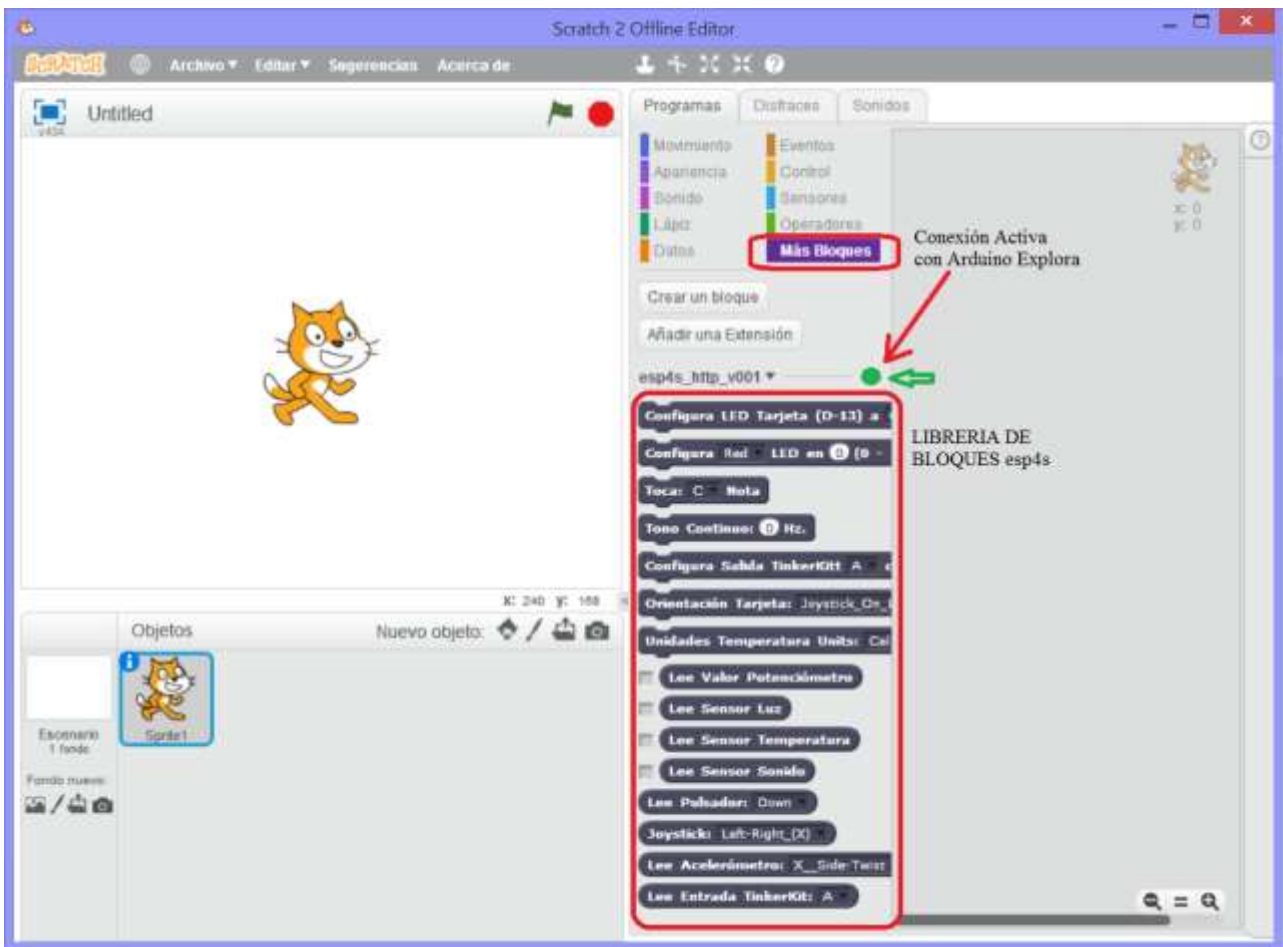
## Usando Scratch

En primer lugar, ejecutamos eps4s. Después, si usted no lo ha hecho, descargue e instale Scratch de <http://scratch.mit.edu/scratch2download/>

Inicio Scratch 2.0 fuera de línea (desde su PC). Luego, mientras mantiene pulsada la tecla **Mayús** haga clic con el ratón en **Archivo**. En la lista desplegable, seleccione "**Import experimental HTTP extension**".

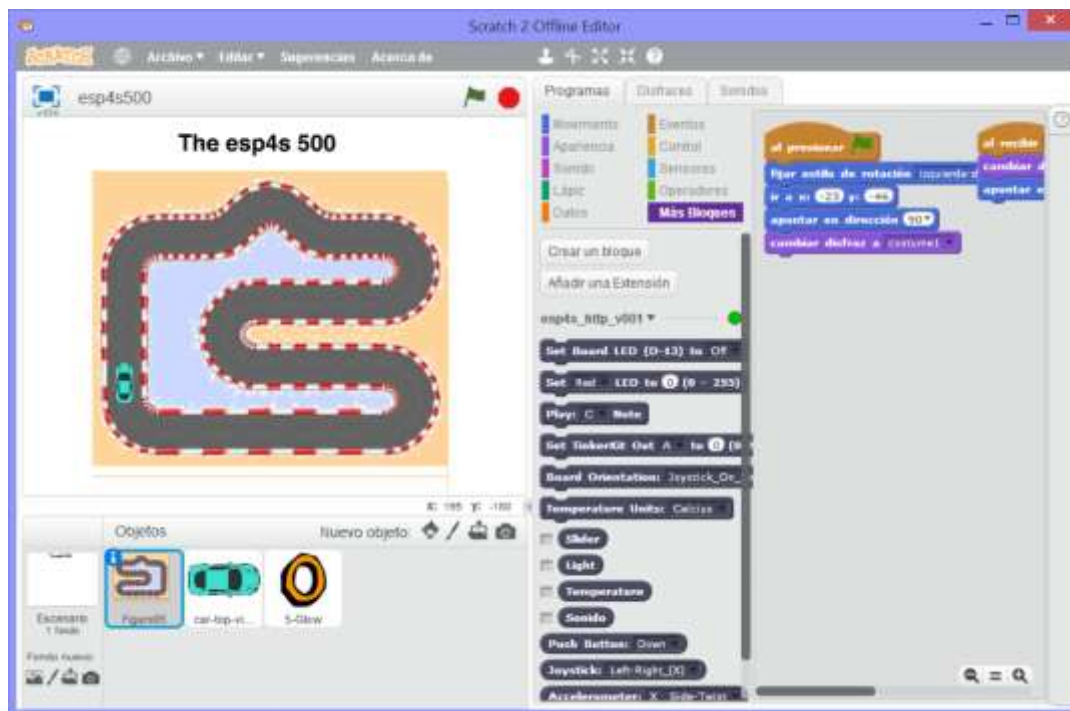


Desde el selector de archivos, vaya al directorio que contiene los archivos extraídos esp4s y seleccione **ScratchFiles/ExtensionDescriptors /ep4s.s2e**. Después, si hace clic en el selector **Más Bloques**, debería ver los bloques esp4s. El indicador de conexión debe ser de **color verde** y se puede iniciar la programación.

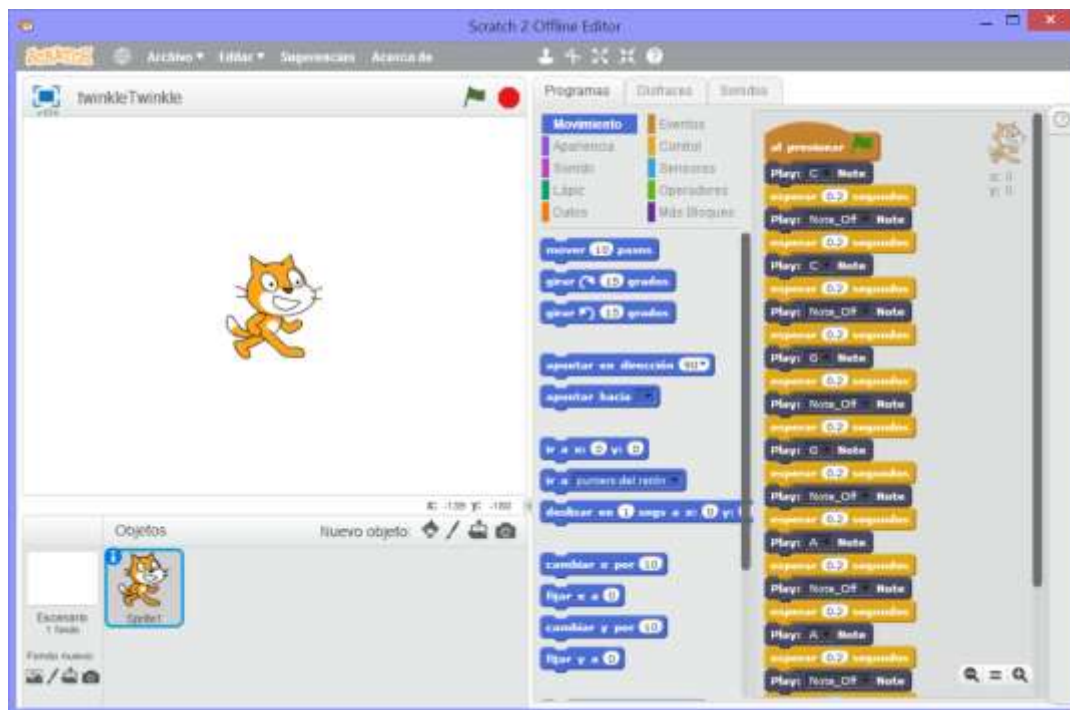


## Proyectos Sertach Incluidos

Hay dos proyectos incluidos en el directorio ScratchFiles/ScratchProjects. Uno de los proyectos es un juego de coche de carreras, el *esp4s500*, y el otro es *twinkleTwinkle* que reproduce las primeras notas de Twinkle Twinkle Little Star. Para cargar y ejecutar los programas, iniciar eps4s, y luego con Scratch abierto ir al menú Archivo/Abrir y seleccione el programa que desea ejecutar.



Ejemplo: *esp4s500*

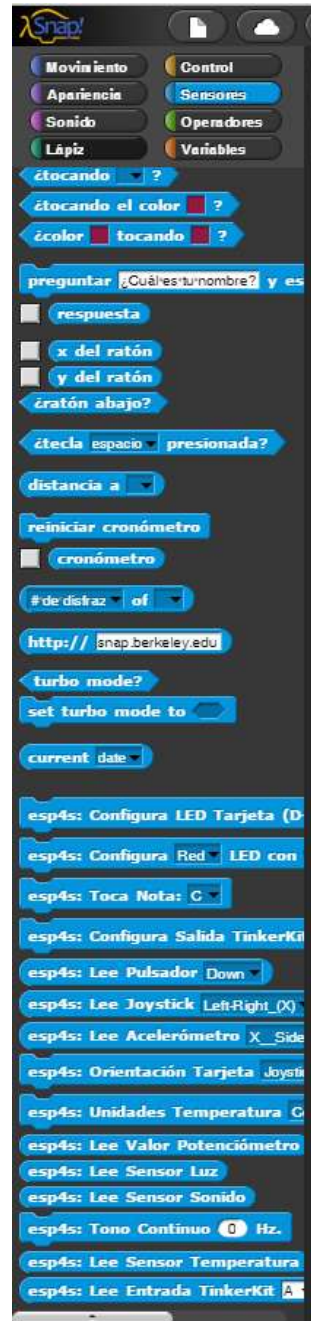


Ejemplo: *twinkleTwinkle*

# Usando Snap!

En primer lugar inicie esp4s, a continuación, utilizando su navegador (ya sea Chrome o Firefox), vaya a la Pagina Web del Editor Snap! <http://snap.berkeley.edu/snapsource/snap.html>.

Seleccione el icono de Archivo  y luego en Importar ...



Utilizando el selector de archivos, vaya al directorio en el que extrajo los archivos esp4s y



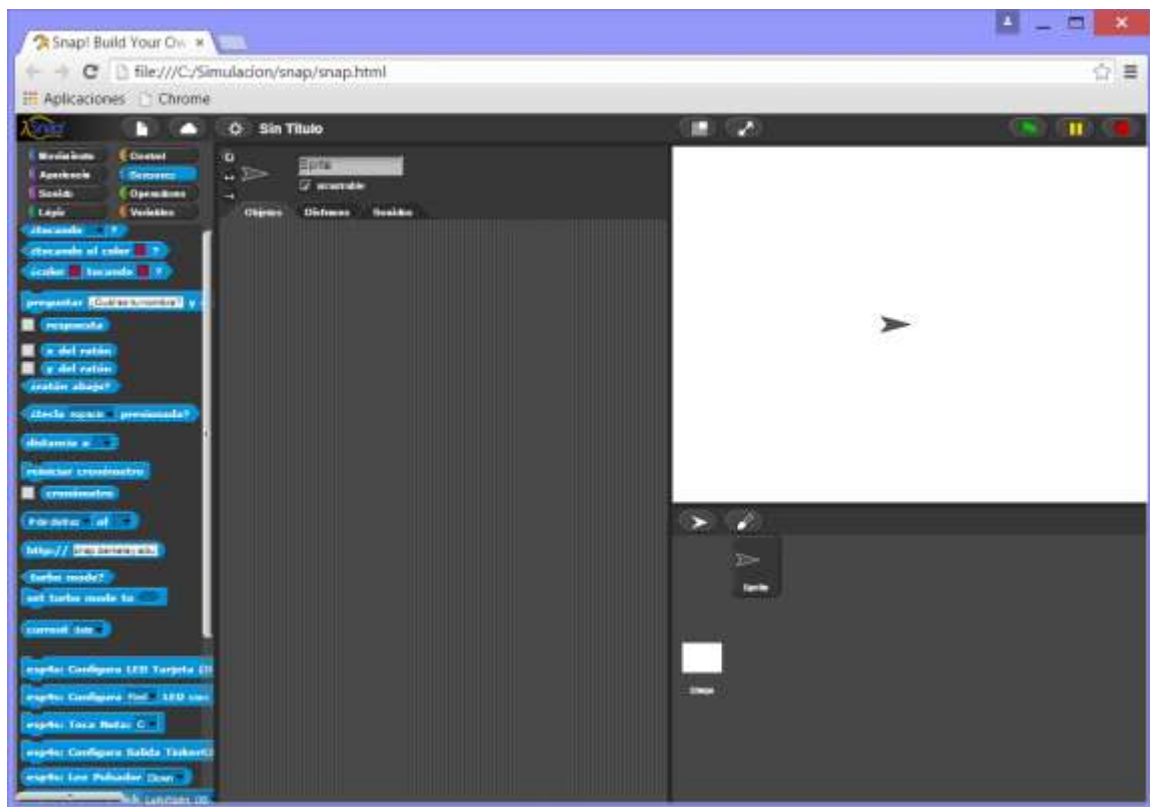
seleccione

*Snap! \_files/Esp4s\_blocks.xml.*

Haga clic sobre Sensores y verá todos los bloques identificados con "esp4s:".



Bloques de la librería esp4s para Snap!

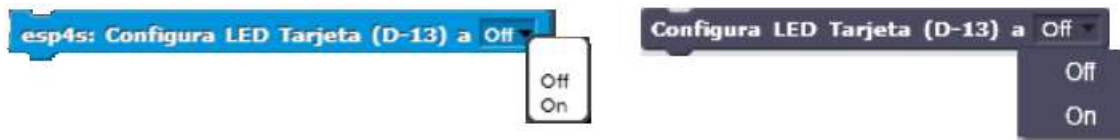


# Descripción de Bloques

Los bloques negros son los bloques de Scratch, y los bloques azules son para Snap !.

## Configurar el Led de la tarjeta

Controla el LED del Pin 13 de la Tarjeta



Este bloque controla el pequeño led amarillo "L" de la tarjeta Arduino Explora. Seleccione On u Off En la lista desplegable.

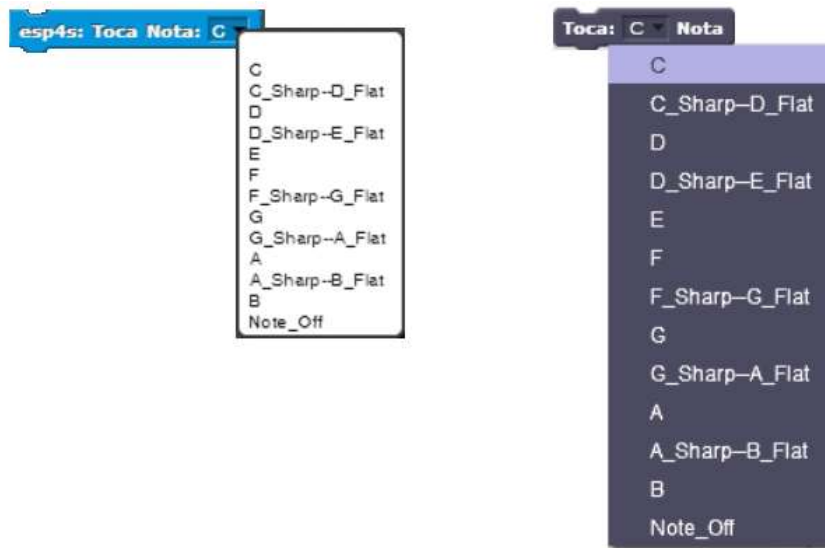
## Seleccionar y Configurar el brillo de leds RGB



Selecciona el Color del LED de la lista desplegable y luego permite introducir un nivel de brillo. 0 es apagado, y 255 es el máximo brillo. Puede activar más de un LED a la vez para mezclar colores.

NOTA: Si utiliza *Tocar Nota* o *Tocar Tono Continuo*, el LED rojo dejará de lucir hasta que se reinicie Esplora. Para obtener una explicación, véase <http://arduino.cc/en/Reference/EsploraTone>

## Tocar Nota



Seleccione la nota de la lista desplegable. La nota se reproducirá continuamente hasta que seleccionar *Note\_Off*.

## Tocar Tono Continuo



Introduzca la frecuencia del tono que desea reproducir expresada en Hertz. Para desactivar el tono apagado, ajustar la frecuencia a cero o utilizar el bloque Nota de Juego y seleccione *Note\_Off*.

## Configura Nivel salida de TinkerKit

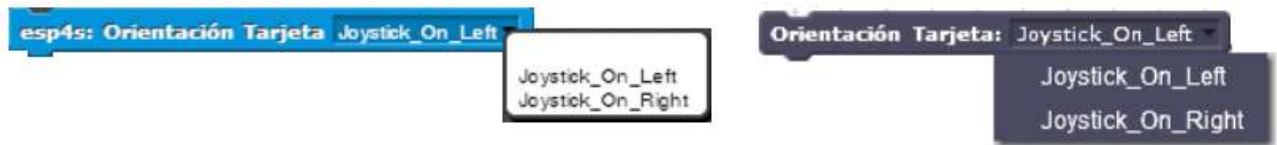


El Esplora tiene dos conectores de salida TinkerKit que permitan establecer el nivel de salida de cualquiera de los canales de salida TinkerKit. El rango de salida es de 0 a 255. Para obtener más información sobre los dispositivos TinkerKit, vaya a:

[http://store.arduino.cc/index.php?main\\_page=index&cPath=16](http://store.arduino.cc/index.php?main_page=index&cPath=16).



## Orientación de la tarjeta



Este bloque le permite seleccionar la orientación tablero de modo que el Joystick se puede colocar a la derecha o a la izquierda de la tabla. Este bloque mantiene la operación consistente para artículos como los joysticks, pulsadores y slider, con independencia de la orientación.

## Unidades de Temperatura.



Este bloque le permite tener la temperatura reportada en grados Celsius o Fahrenheit

## Slider



El bloque *Leer Valor Potenciómetro* informa la configuración actual del control deslizante. La posición extrema izquierda devuelve un cero y la posición extrema derecha devuelve 1023. Estas posiciones se mantienen cuando se utiliza el bloque de orientación del tablero.

## Luz



El bloque de Luz informa el nivel de luz actual. Dado que los niveles de luz pueden variar en gran medida dentro de una habitación, tome una lectura de la luz ambiental, y de nuevo cuando una fuente de luz se dirige hacia el sensor. Utilice baja y alta como valores de referencia para su aplicación.

## Temperatura.

Lee Sensor Temperatura

esp4s: Lee Sensor Temperatura

El bloque de temperatura informa de la temperatura ambiente. Puede seleccionar las unidades, Celsius o Fahrenheit utilizando el bloque de unidades de temperatura.

## Sonido

Lee Sensor Sonido

esp4s: Lee Sensor Sonido

El bloque de sonido informa el nivel sonoro ambiental recibida por el micrófono. El rango de valores es de 0 a 1023 (el más alto).

## Pulsador



Este bloque permite seleccionar y leer el estado de cualquiera de los cuatro botones. Cuando se pulsa un botón, se recoge un "1" en caso contrario devuelve un "0". Los botones mantienen su orientación al utilizar el bloque de Orientación.

## Joystick

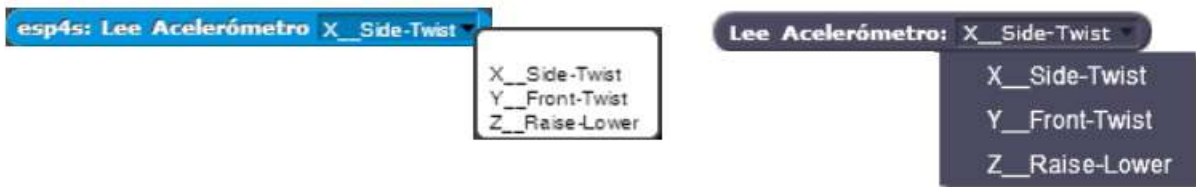


Este bloque le permite monitorizar las posiciones X e Y del joystick, además del estado del pulsador. Los valores devueltos son:

- X: -511 = máxima izquierda, 512 = máximo derecho
- Y: 511 máximo para arriba, -511 Máxima abajo
- Botón: 0 si no se presiona y 1 si se presiona.

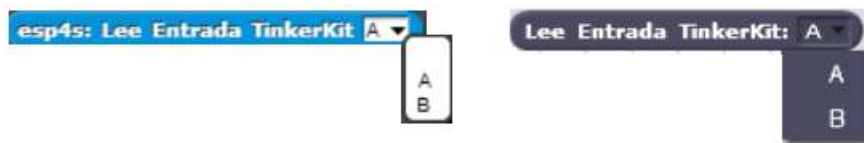
Utilizando el bloque Orientación se asegura que los valores son independientes de la orientación del tablero.

## Acelerómetro.



Este bloque le permite seleccionar y controlar el movimiento X (lado a lado), Y el movimiento (de adelante hacia atrás) o Z movimiento (que se mueve hacia el suelo o en el techo).

## Entrada TinkerKit



Este bloque informa el valor actual de un sensor de TinkerKit. Usted puede elegir cualquiera de los canales A o B, y el valor devuelto se encuentra entre 0 y 1023. Para obtener más información sobre los dispositivos TinkerKit, vaya a [http://store.arduino.cc/index.php?main\\_page=index&cPath=16](http://store.arduino.cc/index.php?main_page=index&cPath=16)

## Solución de Problemas

Los problemas más comunes se producen como consecuencia del olvido de cargar el driver de comunicación que se establece entre Arduino Explora y el PC a través del puerto USB o no enchufar la tarjeta Arduino Explora al PC con lo que se rompe la comunicación. Si necesita ayuda adicional, póngase en contacto con nosotros en el siguiente enlace.

## Si necesita más Ayuda

Si usted tiene algún problema o necesita alguna ayuda, puede ponerse en contacto con nosotros en: [MisterYsLab@gmail.com](mailto:MisterYsLab@gmail.com)

Si quieres estar al día con lo que estamos haciendo, ir a nuestro blog: <http://mryslab.blogspot.com/>  
o póngase en contacto conmigo en Twitter: Alan Yorinks@BrassFigLigee

## Otros Proyectos Open Source de Mr. Y's Lab

Esta es la lista de algunos de los enlaces a otros proyectos "open source" míos:

[https://github.com/MrYsLab/s2a\\_fm](https://github.com/MrYsLab/s2a_fm) <https://github.com/MrYsLab/xi>

<https://github.com/MrYsLab/PyMata>

Traducción y Adaptación al Español: José Manuel Ruiz Gutiérrez  
j.m.r.gutierrez@gmail.com