

---

# Tabla de contenido

0.-Introduction	1.1
1.-¿Qué es PRIBRELLA?	1.2
2.-Kit de préstamo	1.3
3.-El pensamiento computacional	1.4
4.-Configuración	1.5
5.-Salidas LED	1.6
6.-Semáforo	1.7
7.-Pulsador	1.8
8.-Buzzer	1.9
9.-Entradas	1.10
10.-Hasta ahora...	1.11
11.-Salidas	1.12
Grupo robótica	1.13
Muro	1.14
Créditos	1.15

## PIBRELLA

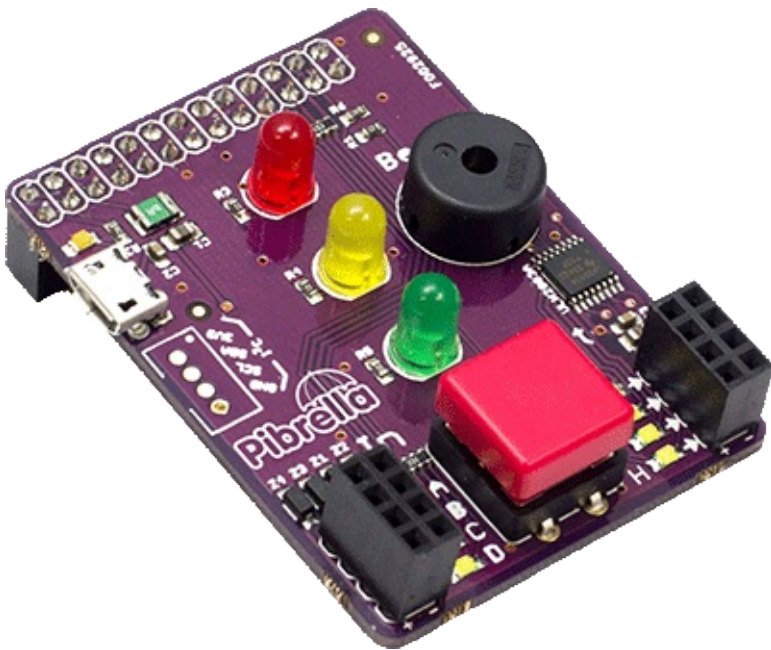


### En este curso vas a aprender:

- Manejar la **RASPBERRY** remotamente
- Programar la RASPBERRY con **SCRATCH**
- Con SCRATCH y **PIBRELLA** haremos unos ejercicios sencillos de robótica

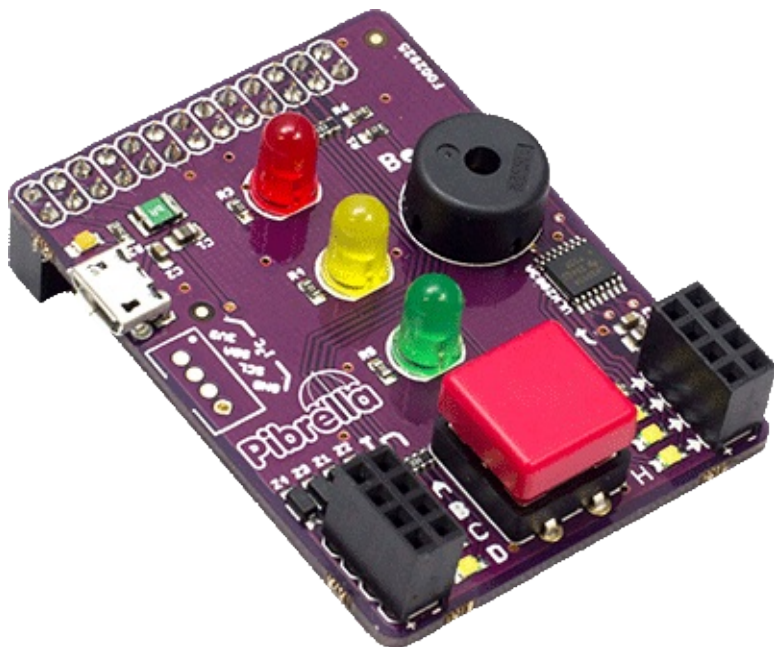
### Conocimientos previos

- Conocimientos BASICOS RASPBERRY [aquí](#)
- Scratch básico lo tienes [aquí en este curso de Aularagón](#).



## ¿Qué es PIBRELLA?

Es una Shield muy sencilla que se acopla a la **Raspberry** en los **GPIO** y trae esta placa al mundo real con sencillas entradas y salidas. Esta placa se puede conseguir [por unas 10 £](#).



¿Qué es una Shield? Mira [esta página de la Shield Echidna](#) ¿Qué es GPIO? pues eso es que no te has leído [este capítulo](#) ¿No podría pasar de la Shield y conectar los leds directamente? Ni pensarlo, se ve que no te has leído [este capítulo](#)

## ¿Qué tiene?

- Salidas tipo LED rojo, naranja y verde
- Salida un pequeño buzzer
- Entrada un interruptor
- 5 Conectores de entradas ABCD
- 5 Conectores de salida EFGH

## Precauciones

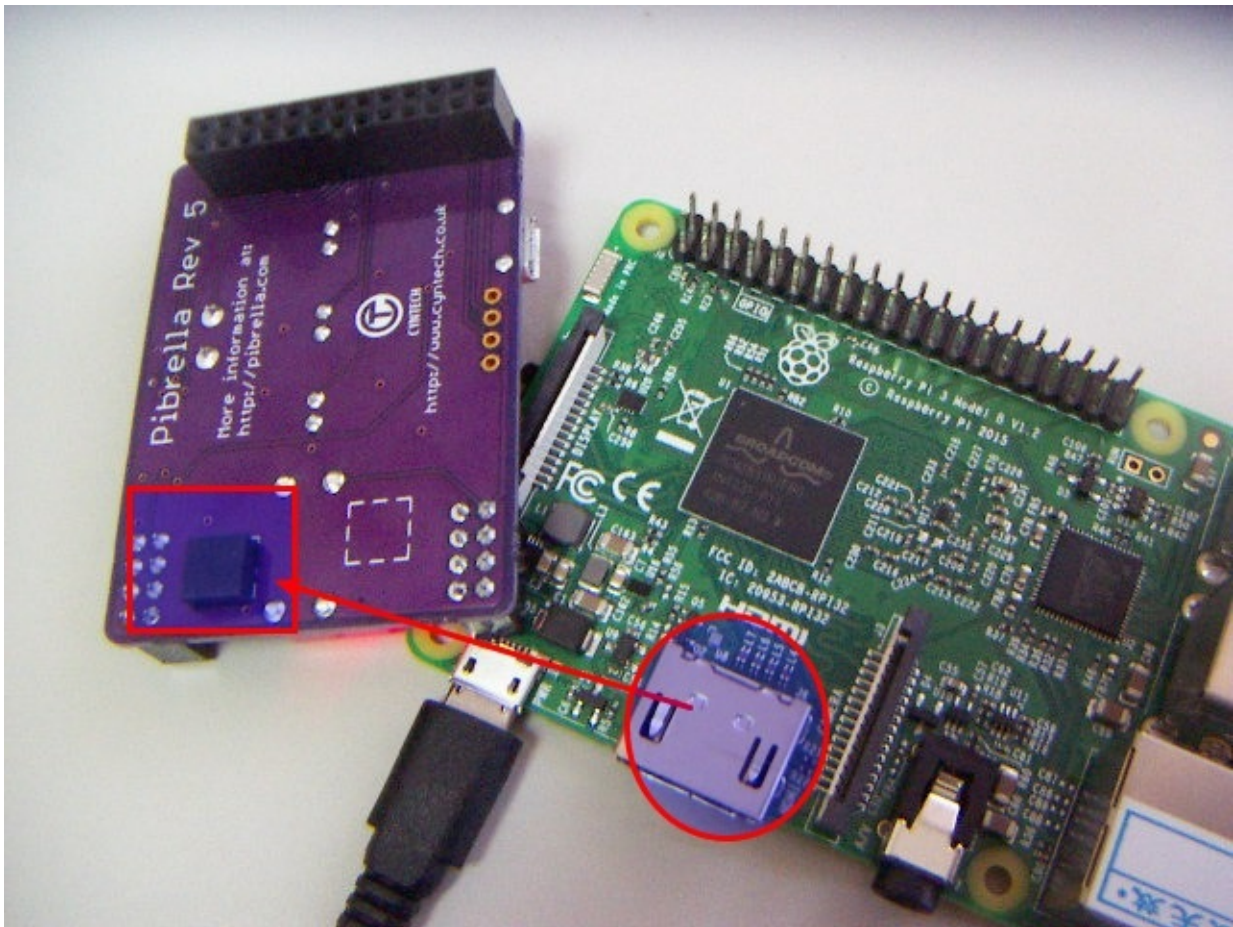
Raspberry no tiene de forma nativa protecciones en sus GPIO ([ya lo sabes](#)), Pibrella ayuda a prevenir algún daño y por eso podemos conectar cosas en las entradas y salidas, no obstante:

- No podemos mezclar las entradas y las salidas.
- No podemos cortocircuitar las salidas
- No podemos conectar ninguna potencia superior a 5V.
- Y por supuesto, como en todas las placas: No utilices las tomas de corriente (osea los enchufes 220V de la pared), excepto para conectar las fuentes de alimentación, claro.

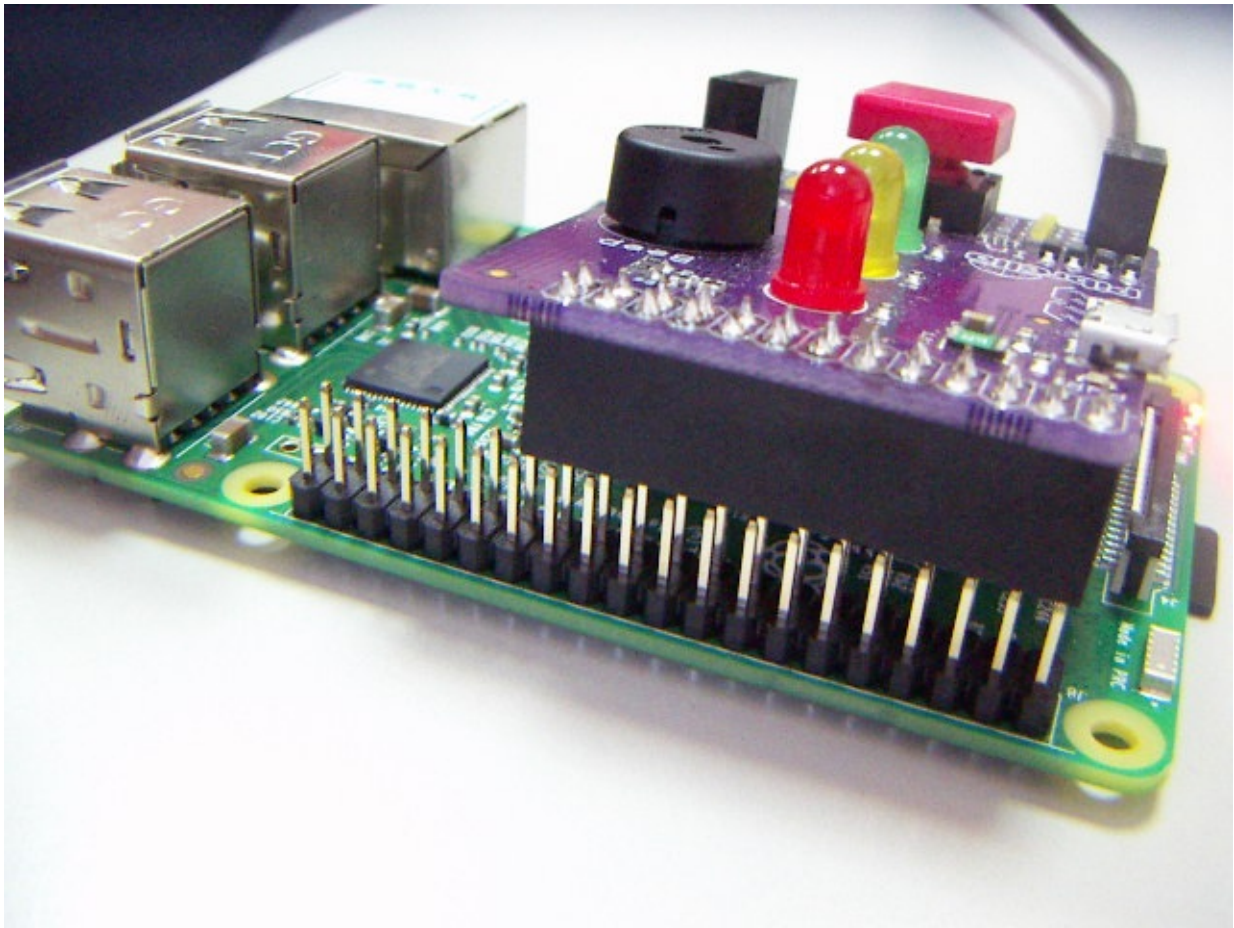
[Fuente](#)

## ¿Cómo se conecta?

Pegaremos la pequeña almohadilla negra que viene en el paquete para que descansa la Pibrella en el mismo conector de HDMI de la Raspberry (si fuese una Raspberry antigua, pues en el otro cuadrado, [ver](#)):



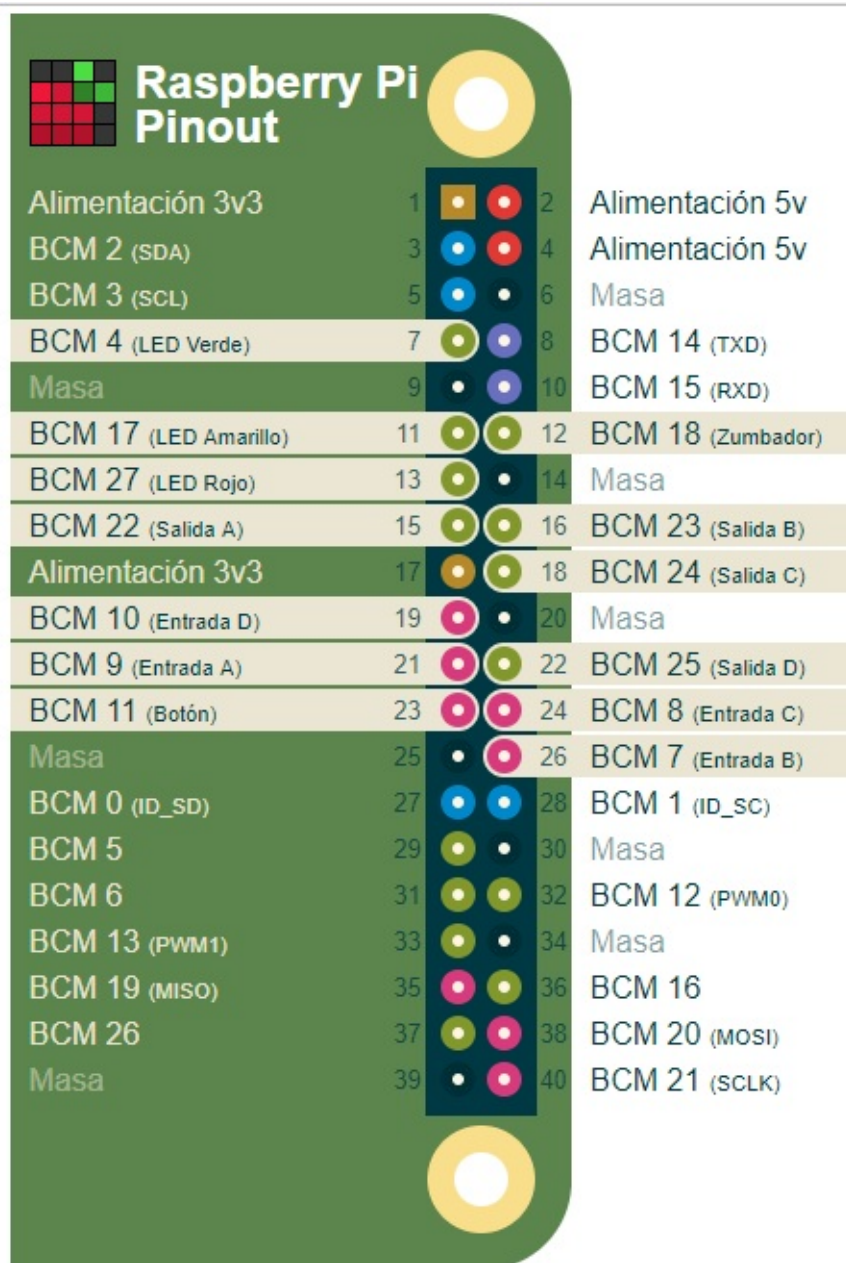
Y lo conectamos en los pines de GPIO más exteriores:



## ¿Qué pines de GPIO utiliza?

Pues estos:





Fuente: <https://es.pinout.xyz/pinout/pibrella>

## ¿Más información?

- En <http://pibrella.com/> tienes la página oficial, donde a grandes rasgos explica qué es Pibrella y la tienda oficial donde comprarlo en <https://shop.cynotech.co.uk/products/pibrella>
- En <http://guides.cynotech.co.uk/pibrella/pibrella/> tienes una página completa de documentación
- [Muro sobre Pibrella](#)

## KIT DE PRESTAMO

Si haces el curso, está disponible este préstamo para que lo puedas hacer:

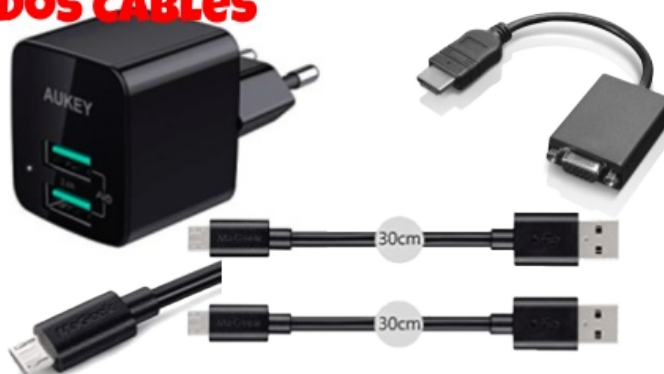


**RASPBERRY**

**CARGADOR  
DOBLE USB CON  
DOS CABLES**

**HDMI-VGA**

**MICRO SD Y ADAPTADOR**



## Pensamiento computacional

¿Dónde se encaja esta shield?¿se puede comparar este robot con otros robots de otros cursos que hacemos desde CATEDU?

Esta [es la hoja de ruta](#), no se tiene que tomar al pie de la letra, pero intenta ayudar al profesorado que tenga una visión global de tanta oferta:

Como se puede ver **P8IBRELLA** es **LA ALTERNATIVA ROBÓTICA DE SCRATCH usando RASPBERRY**. O también se puede ver como para trabajar con la Raspberry, tenemos dos alternativas, con código o de forma gráfica en bloques con Scratch y Pibrella.

## RoboTICa

Oferta de formación en Pensamiento computacional del Centro Aragonés de Tecnologías para la Educación.





# 1 CONFIGURAR TU RASBERRY

Necesitas hacer estos pasos:

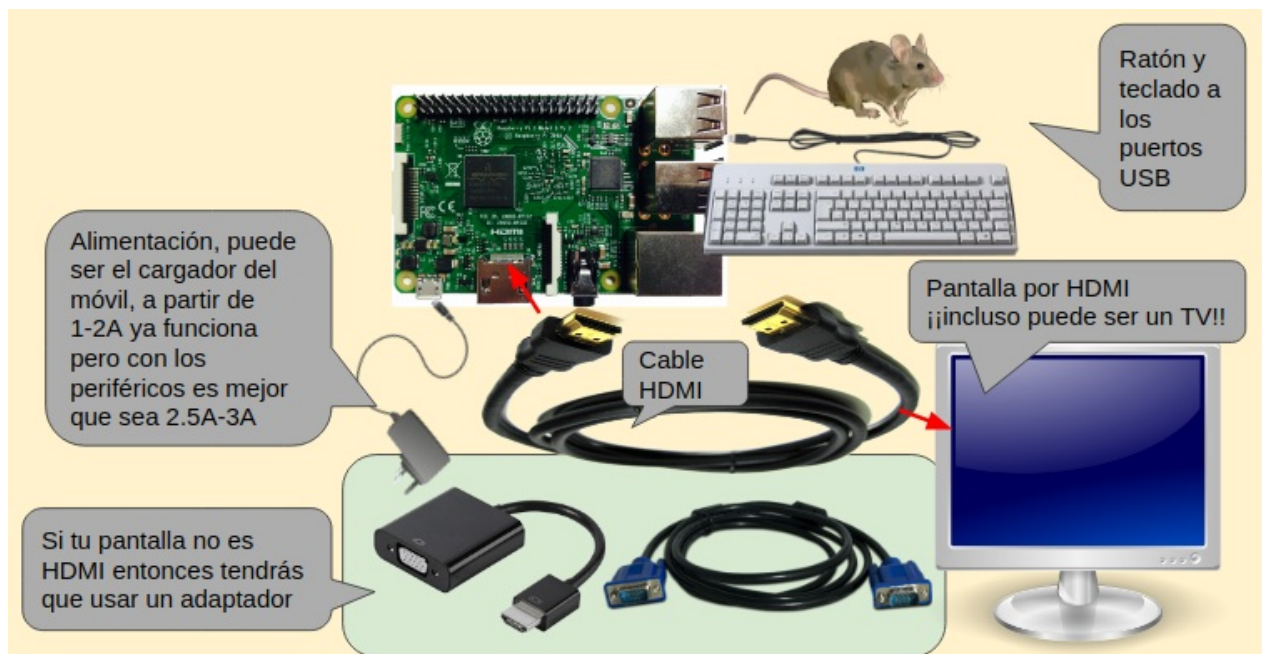
1. **INSTALAR EL SISTEMA OPERATIVO RASPBIAN** [aquí cómo hacerlo](#) y por supuesto saber apagarlo correctamente ([aquí](#)).

Los siguientes pasos **déjalos para el final**, para [la práctica final del coche](#).

1. **CONECTAR LA RASPBERRY A LA WIFI Y ASIGNARLE UNA IP FIJA** [aquí cómo hacerlo](#)
2. No obligatorio pero útil es el comunicarte vía remotamente de forma textual con SSH ([aquí](#)) cambiar usuario y contraseña ([aquí](#))
3. **CONECTARTE CON LA RASPBERRY de forma remota y gráfica con VNC** [aquí como hacerlo](#) .

## Aclaraciones: ¿Necesito los 3 pasos anteriores para conectarme via remotamente con la Raspberry para la robótica Pibrella?

Podríamos usar la Raspberry con la Pibrella conectado con un teclado, ratón y una pantalla **y no necesitamos un ordenador, usamos el mismo ordenador que es la Raspberry!!**:



## PERO.. que pasaría si ...

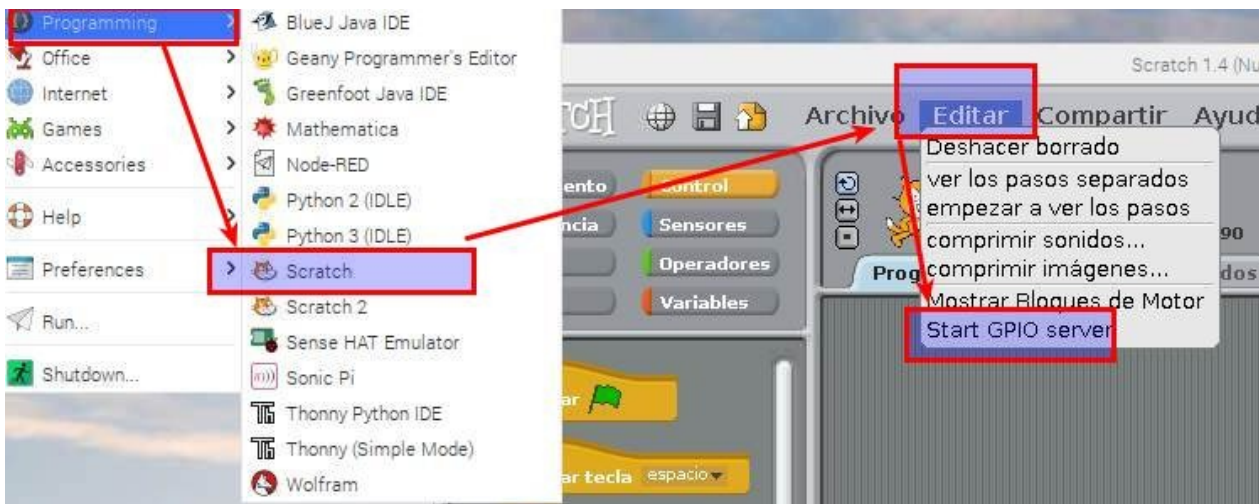
Quieres hacer una práctica de la Pibrella donde se mueva por ejemplo [el coche](#). Entonces estás obligado a comunicarte de forma remota.

**Para el resto de prácticas NO ES NECESARIO COMUNICARTE DE FORMA REMOTA** incluso es más rápido utilizar directamente la Raspberry con la Pibrella con pantalla, teclado y ratón.

# 2 CONFIGURAR SCRATCH para que utilice GPIO

En el Scratch de la RASPBERRY (ojo no de tu PC) tenemos que configurarlo para que interactúe con el GPIO de la Raspberry:

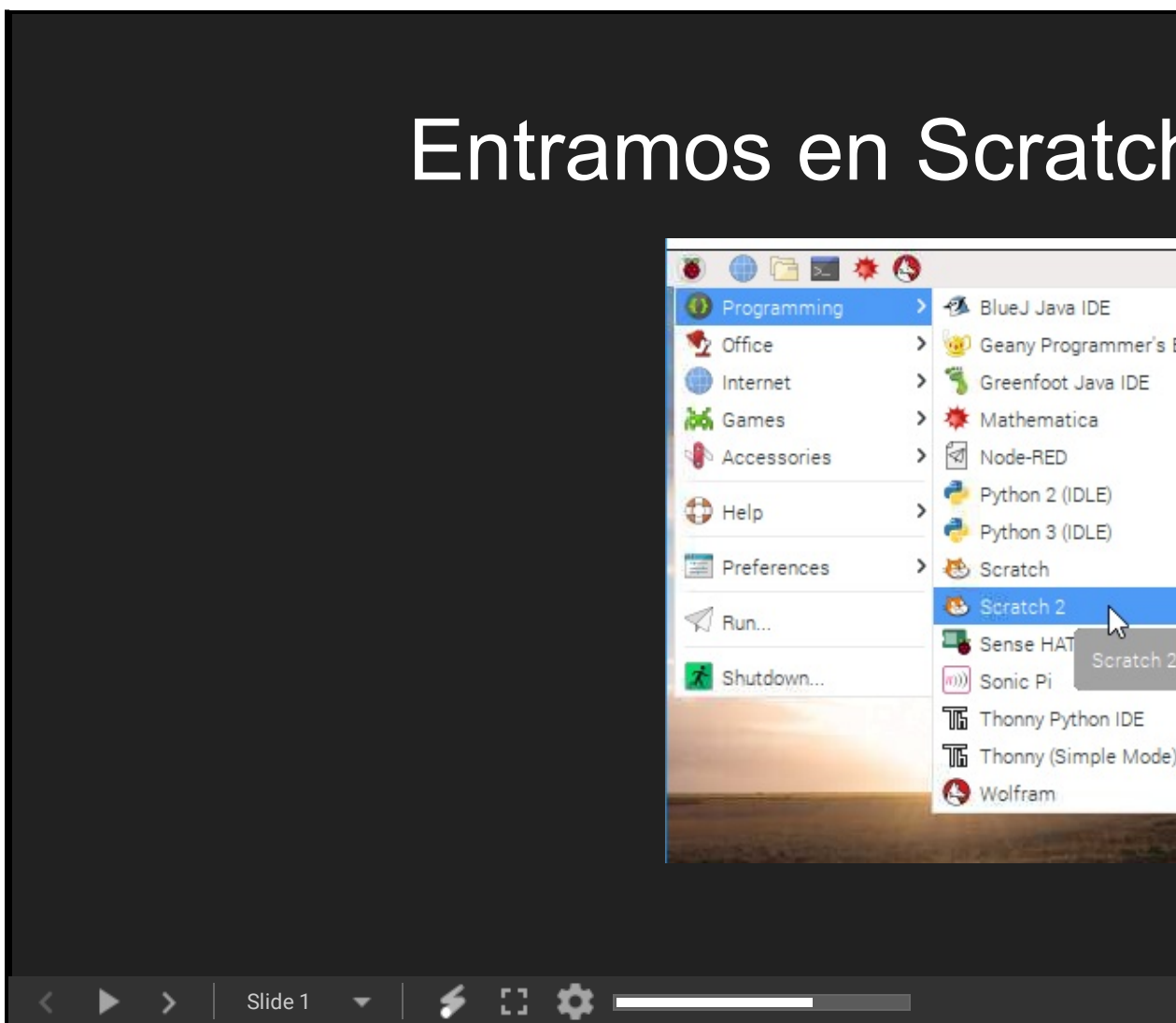
1. Hay que entrar en **SCRATCH 1.0** (ojo no la 2.0)
2. Editar - **Start GPIO server**



Para pararlo Editar - STOP GPIO Server

### 3 ¿CON SCRATCH 1.0??? ¿NO SE PUEDE CON SCRATCH 2.0?

Sí que se puede, pero como puedes ver en la siguiente presentación **SÓLO AÑADE 2 FUNCIONES** PARA LAS GPIO, con eso sí que se puede utilizar Pibrella **pero no tiene funciones específicas para Pibrella** que sí que tiene Scratch 1.0



## 4 ¿Y NO SE PUEDE CON PYTHON?

Pues sí, [aquí tienes las librerías](#) que necesitas y ejemplos.

## Salidas de PIBRELLA tipo LED

### ¿Cómo se programan las SALIDAS TIPO LED?

Muy fácil:

1. Se crea una variable con el nombre **AddOn**
2. Le asignamos el valor **Pibrella**
3. Enviamos un **mensaje** (broadcast) de la salida de Pibrella que queremos encender o apagar

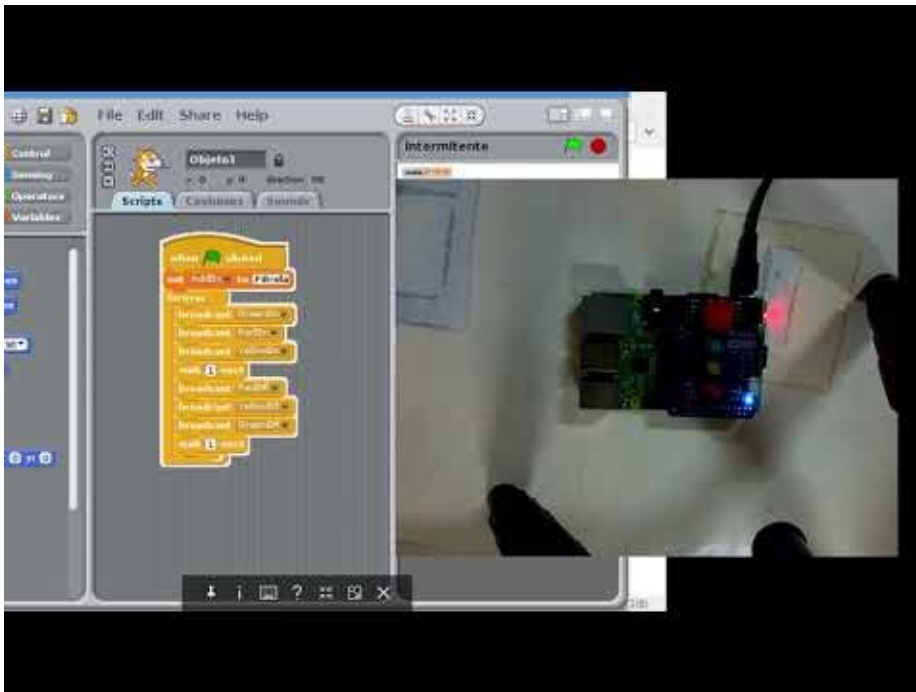
### ¿Qué mensaje hay que enviar?

Son tan intuitivos que no hace falta explicarlos:

- RedOn
- RedOff
- YellowOn
- YellowOff
- GreenOn
- GreenOff

### Un ejemplo

Vamos a realizar un intermitente:



[Video link](#)

¿Te atreves?

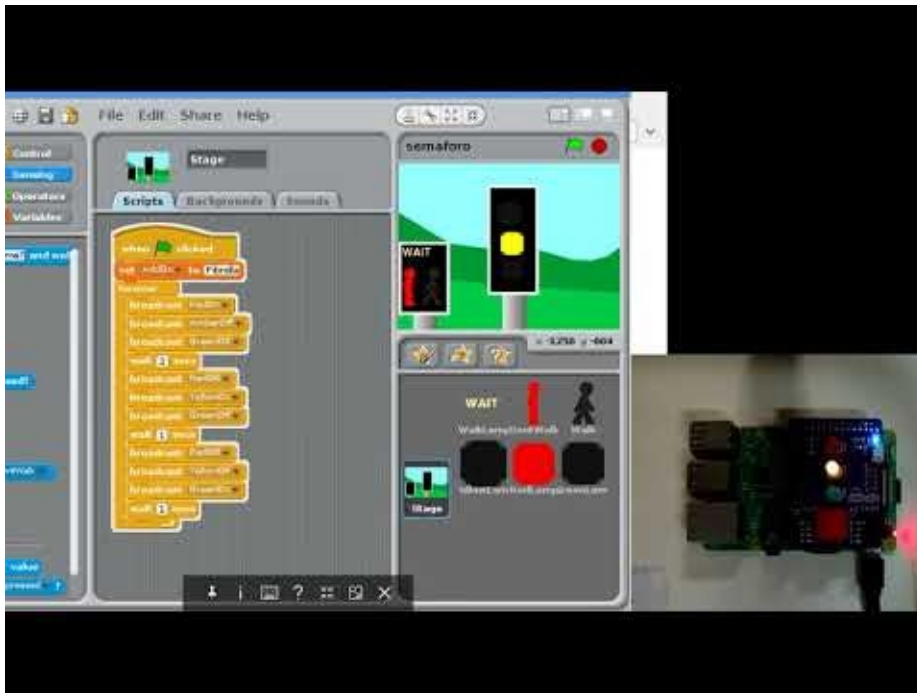
Solución





## Semáforo

Un semáforo interactuando con otro virtual en Scratch:



[Video link](#)

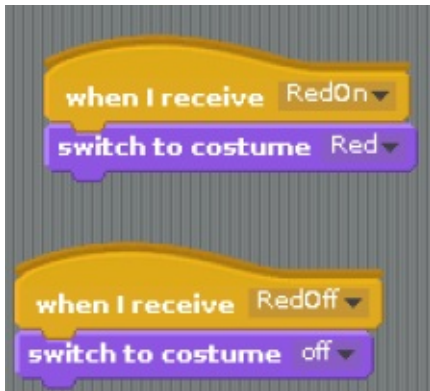
¿Te atreves?

Solución



El escenario del fondo tiene el siguiente programa:

Y cada luz tiene dos disfraces, uno encendido y otro todo negro y sus programas muestran uno de los dos disfraces según el mensaje enviado por el anterior programa:

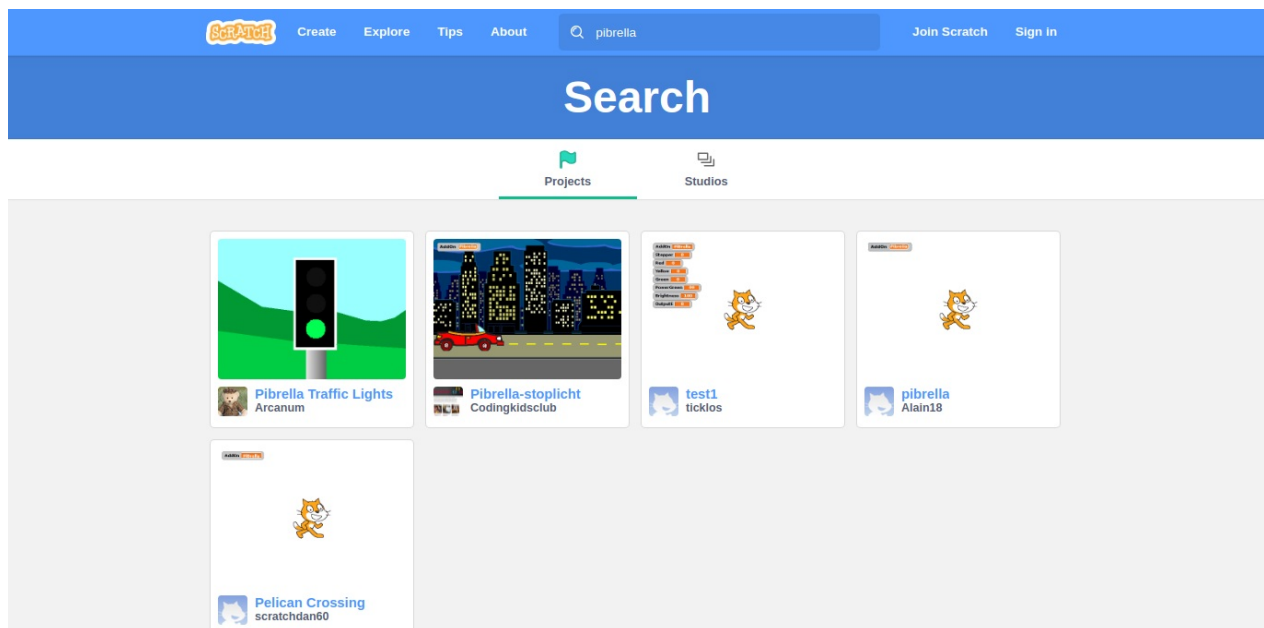


El archivo lo tienes en <https://github.com/JavierQuintana/pibrella>

## Soy un vago

Bueno, quizás nos hemos pasado con este titular, era sólo para llamar la atención, quizás el mejor título es "¡Vamos a reutilizar código!:

Si entramos en la página oficial de [Scratch](#) vemos que hay pocos proyectos realizados con Pibrella, y encontramos este ejemplo del semáforo (por cierto mucho más complejo que el explicado anteriormente):



Si nos descargamos uno de ellos, **NO NOS VAA FUNCIONAR EN LA RASPBERRY CON PIBRELLA PUES ESTÁN REALIZADOS CON SCRATCH 2.0**

Para ello necesitamos alguna herramienta que convierta proyectos Scratch 2.x a 1.x una de ellas puede ser esta:

<http://kurt.herokuapp.com/20to14>

## Kurt Tools

*tools for scratch projects —based on [kurt](#). made by [blob8108](#).*

## Retro Converter

Convert Scratch 2.0 projects ( `.sb2` ) back into Scratch 1.4 project files ( `.sb` ).

For people who prefer Scratch 1.4.

The converter can't handle anything not supported by Scratch 1.4, such as cloning and vector graphics, but it's fine with everything Scratch 1.4 supports.

Upload a Scratch 2.0 project to convert:

No file chosen

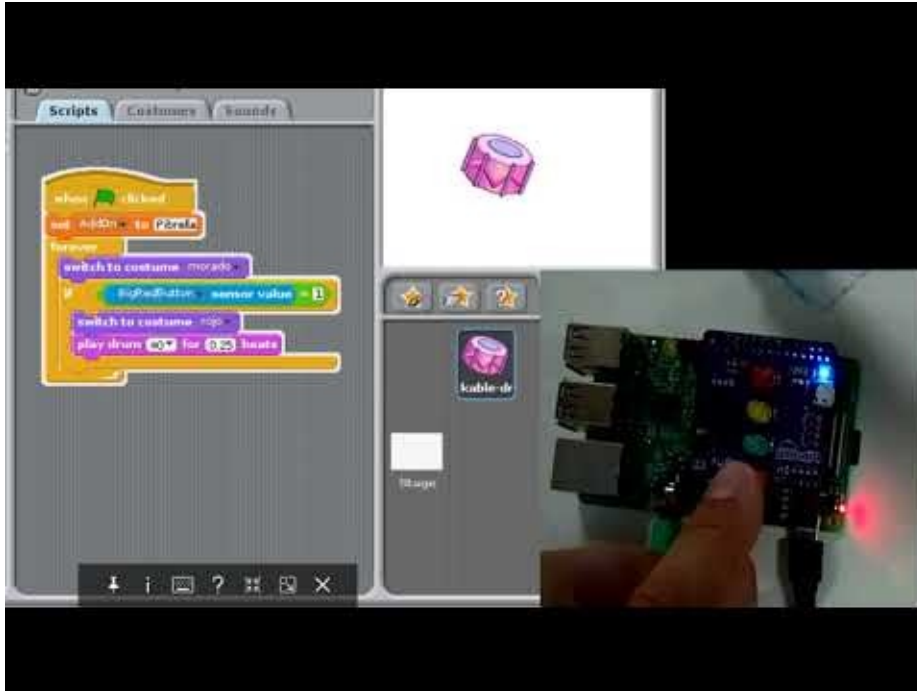
Nota la conversión o es perfecta, sobre todo en la ubicación de los Sprites en el escenario, los mueve de su situación.

## Pulsador rojo

El pulsador rojo de la Pibrella es un sensor que se detecta con esta instrucción en Scratch



Vamos a hacer un tambor:



[Video link](#)

¿Te atreves?

Solución

El archivo lo tienes en <https://github.com/JavierQuintana/pibrella>

El tambor tiene dos disfraces para visualizar cuando se pulsa, y cuando se pulsa que emita un sonido:





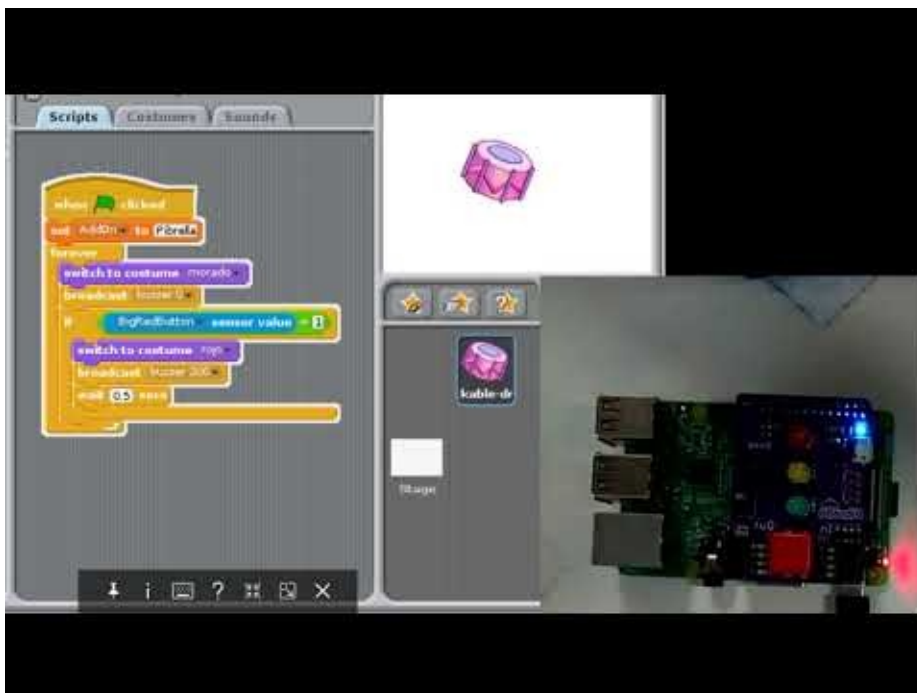


## Buzzer

El pequeño altavoz de la Pibrella se activa igual que los LEDs, enviando un mensaje seguido de la frecuencia, si la frecuencia es nula, el buzzer se apaga:



Vamos a modificar el tambor:



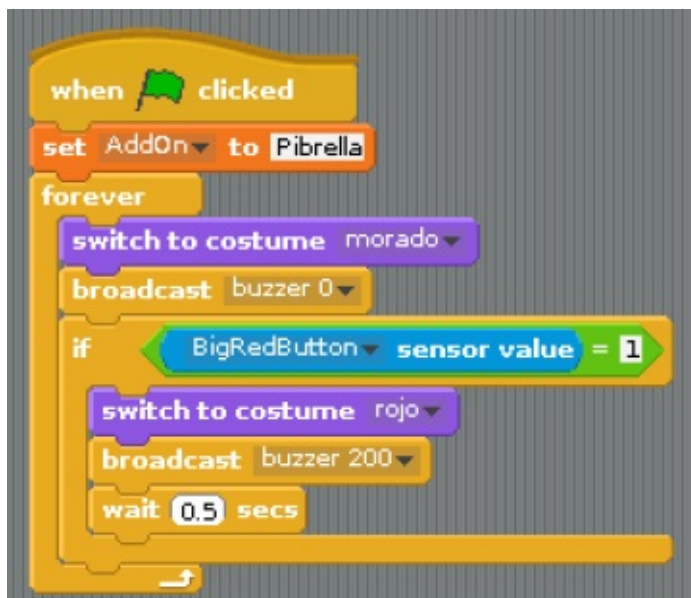
[Video link](#)

¿Te atreves?

Solución

El archivo lo tienes en <https://github.com/JavierQuintana/pibrella>

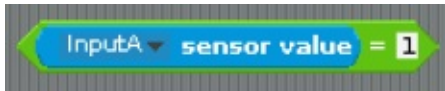
El tambor tiene dos disfraces para visualizar cuando se pulsa, y cuando se pulsa que emita el buzzer:



# Entradas

Nota: En la manipulación de las entradas y salidas **CONECTA PIRELLA CON SU PROPIA ALIMENTACIÓN USB**

Pibrella tiene 4 entradas ABCD que se controlan con la instrucción:

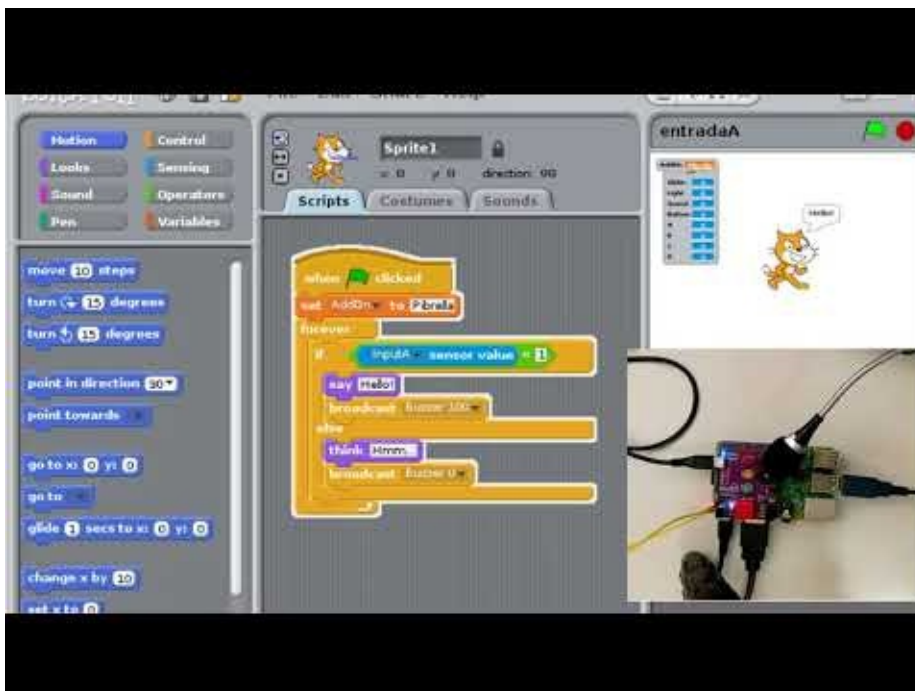


Donde A puede ser las otras letras BCD y el valor 1 es cuando están cortocircuitados los dos pines de A, por lo tanto el valor 0 corresponde cuando están en abierto.

Internamente son interruptores en configuración **pull-up** donde la masa es el pin más interno y la tensión de alimentación es 3.3V:



Vamos a modificar el tambor :



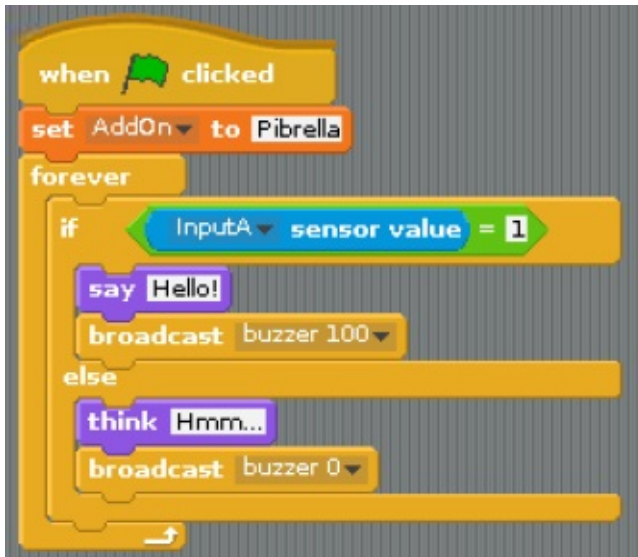
[Video link](#)

¿Te atreves?

Solución

El archivo lo tienes en <https://github.com/JavierQuintana/pibrella>

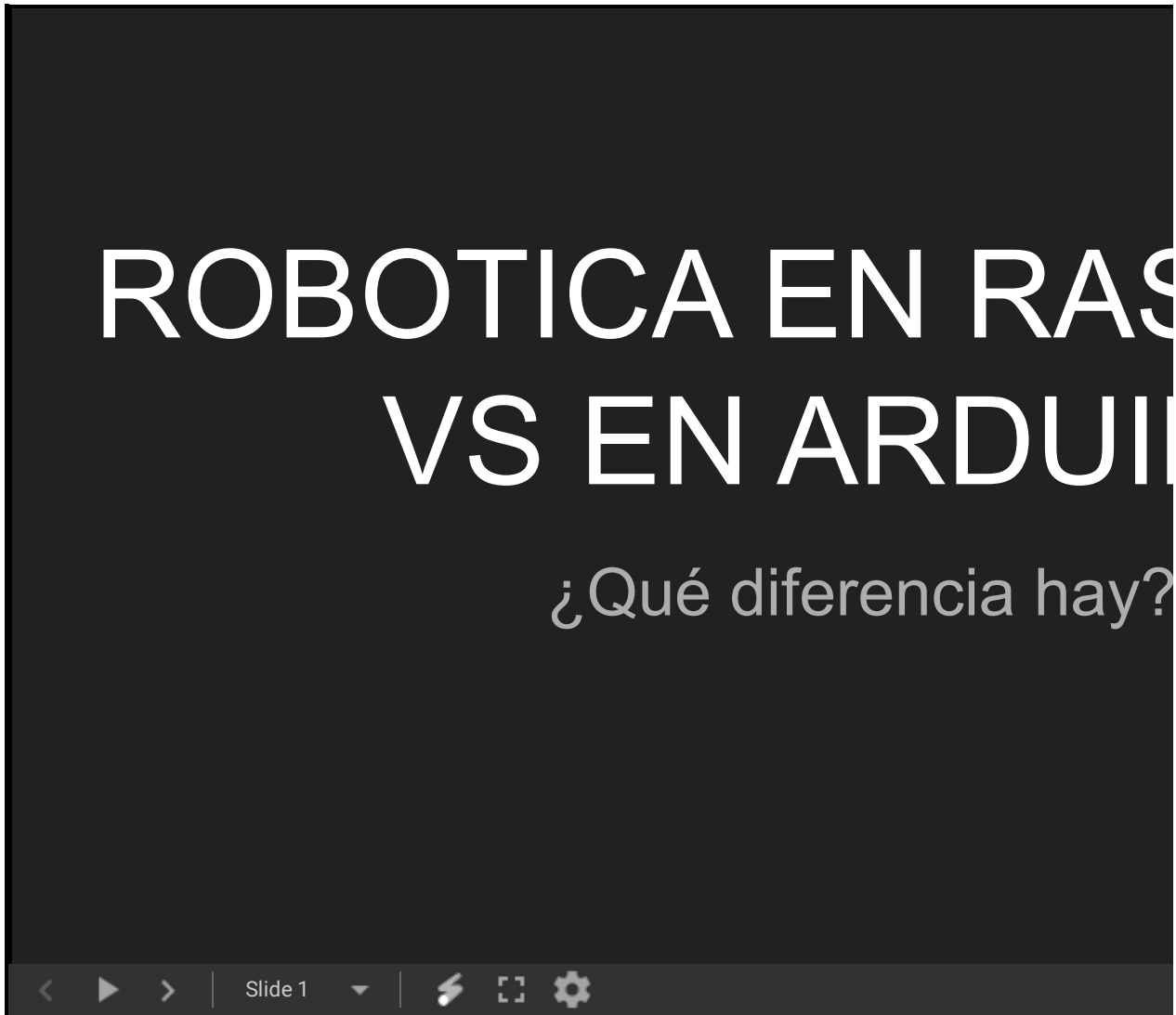
Hacemos dos actuaciones diferentes, decir *Hola* y *buzzer*, según el evento de InputA valga 0 o 1:



## Hasta ahora te hemos dejado de forma local...

Podías usar la Raspberry con la Pibrella de forma local, con una pantalla, teclado y ratón pero ... a propósito vamos a forzarte a usarlo remotamente para que aprendas esta opción

Mira la diapositiva 3 y entenderás lo que quiero decir:



¿COMO SE HACE?

Pues eso es que no has visto [ESTE TEMA DE RASPBERRY MUY BÁSICO](#)



## Salidas

La **ventaja** de Pibrella es que podemos conectar directamente un pequeño motor, led, servo... son salidas tipo interruptor, que conecta con alimentación de 5V o no conecta con alimentación 0V y cada salida tiene sus dos terminales, no comparten masa.

Las instrucciones son sencillas:

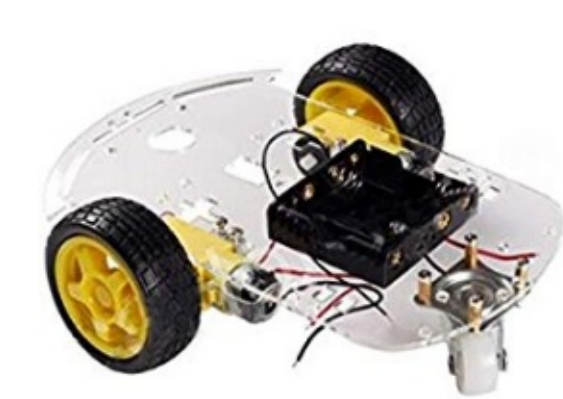
- Para encender la salida E: Enviar el mensaje **OutputEon**
- Para apagar la salida E: Enviar mensaje **OutputEoff**

Os dejamos a vuestra imaginación cuales serían los mensajes para las otras 3 salidas **F, G y H**.

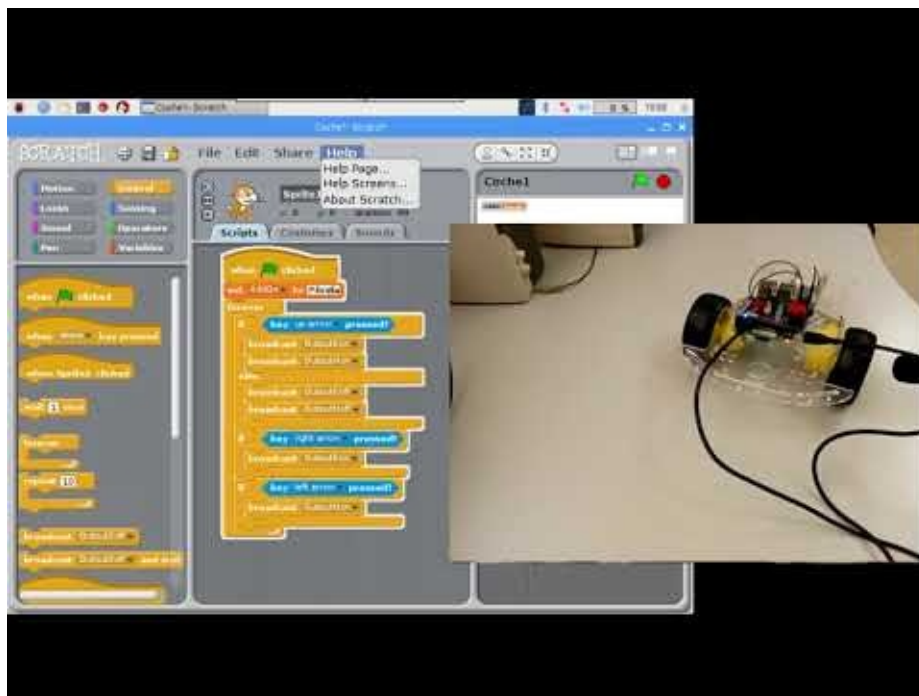
Vamos a utilizarlo para mover un pequeño coche, así damos movimiento a nuestra pibrella.

La desventaja es que no puede invertir la polaridad, luego no podemos mover nuestro coche hacia atrás.

Por menos de 7€ se puede conseguir un sencillo chásis (no hace falta el soporte pilas)



Vamos a realizar un pequeño "coche teledirigido"



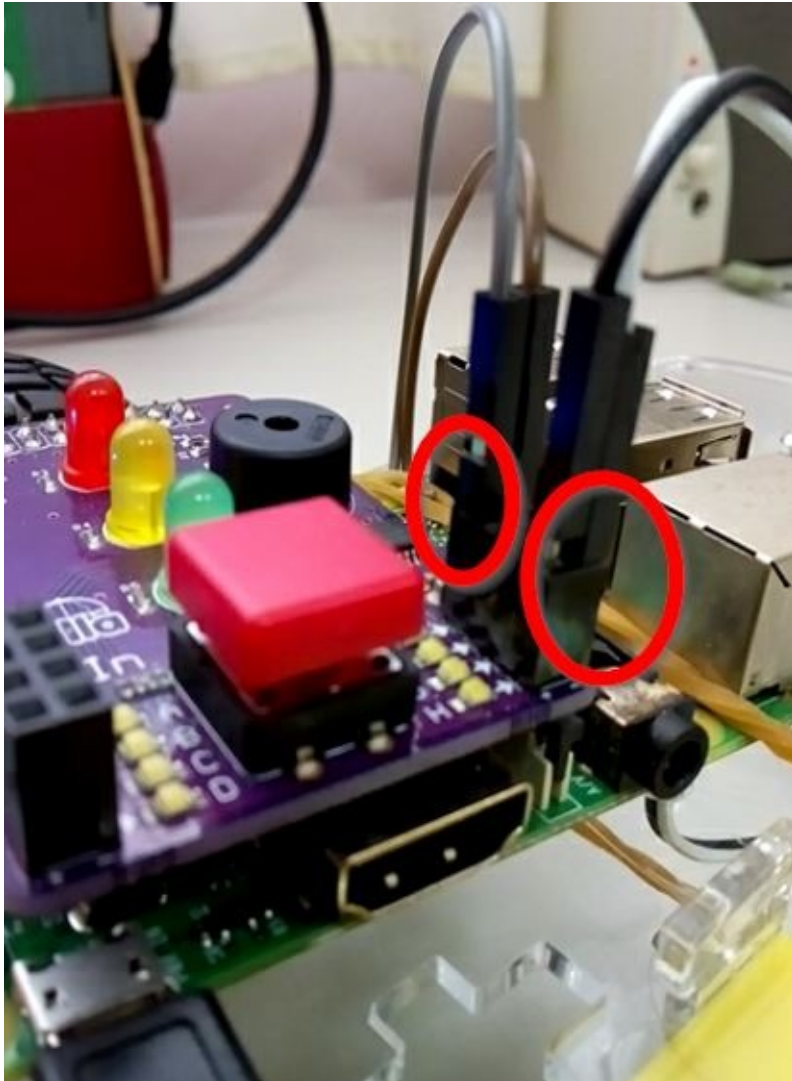
[Video link](#)

¿Te atreves?

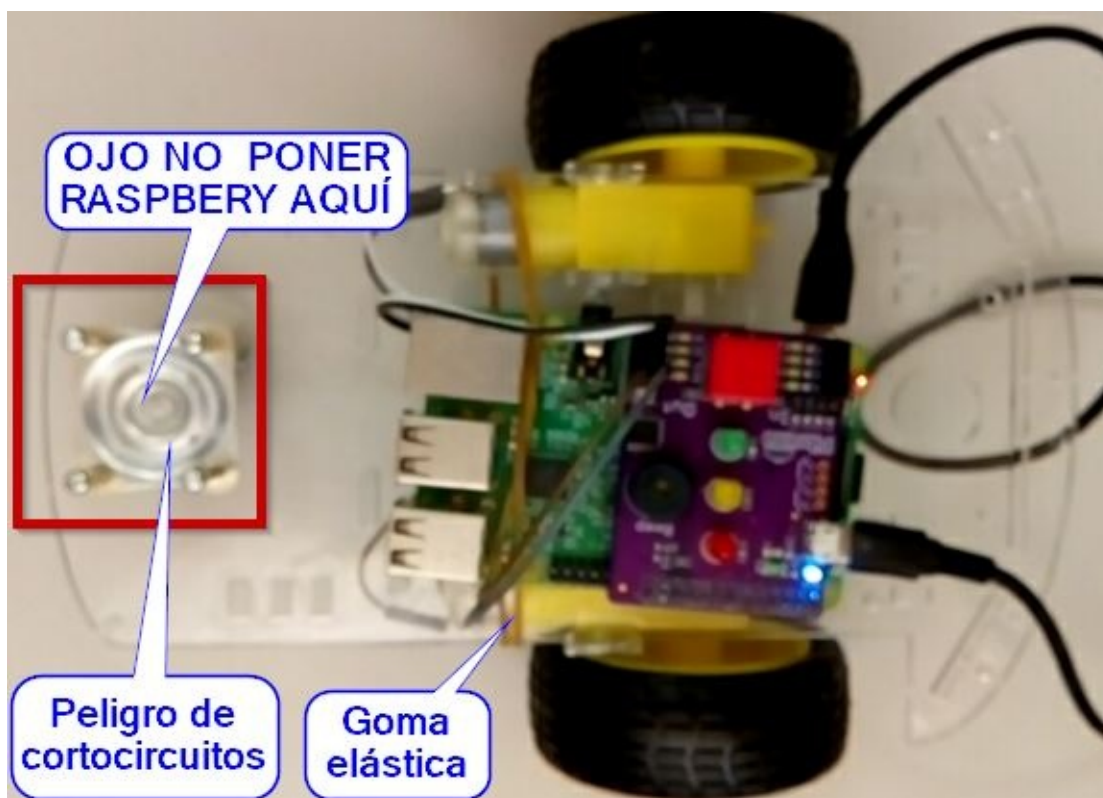
### Solución

El archivo lo tienes en <https://github.com/JavierQuintana/pibrella> como coche1

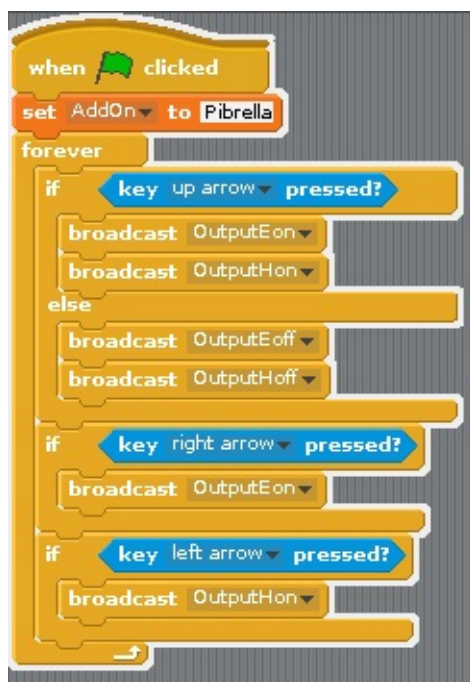
Primero conectamos cada motor en una de los conectores salida de la Pibrella, nosotros lo vamos a conectar en el E y en el H:



Ponemos la raspberry en el chasis con una goma elástica y encendemos **cuidado, no encima de la rueda loca pues provocarías cortocircuitos con los tornillos y la placa raspberry !!!**



Y el programa es sencillo:



¡¡Acuerdate de activar GPIOServer !!!

Algunos se quejan de que no es "independiente" por culpa de los cables de alimentación. SOLUCIÓN: Pon dos baterías externás de USB típicas que se usan para los móviles



## Grupo ROBOTICA EDUCATIVA EN ARAGÓN

En <http://robaragon.chatbro.com> puedes ver un chat sincronizado con Telegram que que te puedes unir con [este enlace](#) para:

- Compartir dudas y resolverlas
- Propuestas, eventos e información
- Ver lo que hacen otros compañeros





## Muro sobre robótica de Pibrella

Pon lo que consideres interesantes para este curso tuyo o de otros:



## Créditos

### Autoría

- Javier Quintana Peiro

### Colaboradores:

---

Cualquier observación o detección de error por favor aquí [sopORTE.catedu.es](mailto:sopORTE.catedu.es)

Los contenidos se distribuye bajo licencia Creative Commons tipo BY-NC-SA.



# GOBIERNO DE ARAGON

Departamento de Educación,  
Cultura y Deporte

**CATEDU**



CENTRO ARAGONÉS de TECNOLOGÍAS para la EDUCACIÓN

