

Bluetooth

- [BLUETOOTH un poco de teoría](#)
- [Sólo con la microbit: BLUETOOTH programa en Makecode](#)
- [BLUETOOTH programa Serial Bluetooth Terminal](#)
- [Avanzado: BLUETOOTH con App Inventor. Extensiones](#)
- [Avanzado: BLUETOOTH App Inventor programa. Sólo con la micro:bit](#)
- [Ahora con la granja](#)

BLUETOOTH un poco de teoría

ONDAS

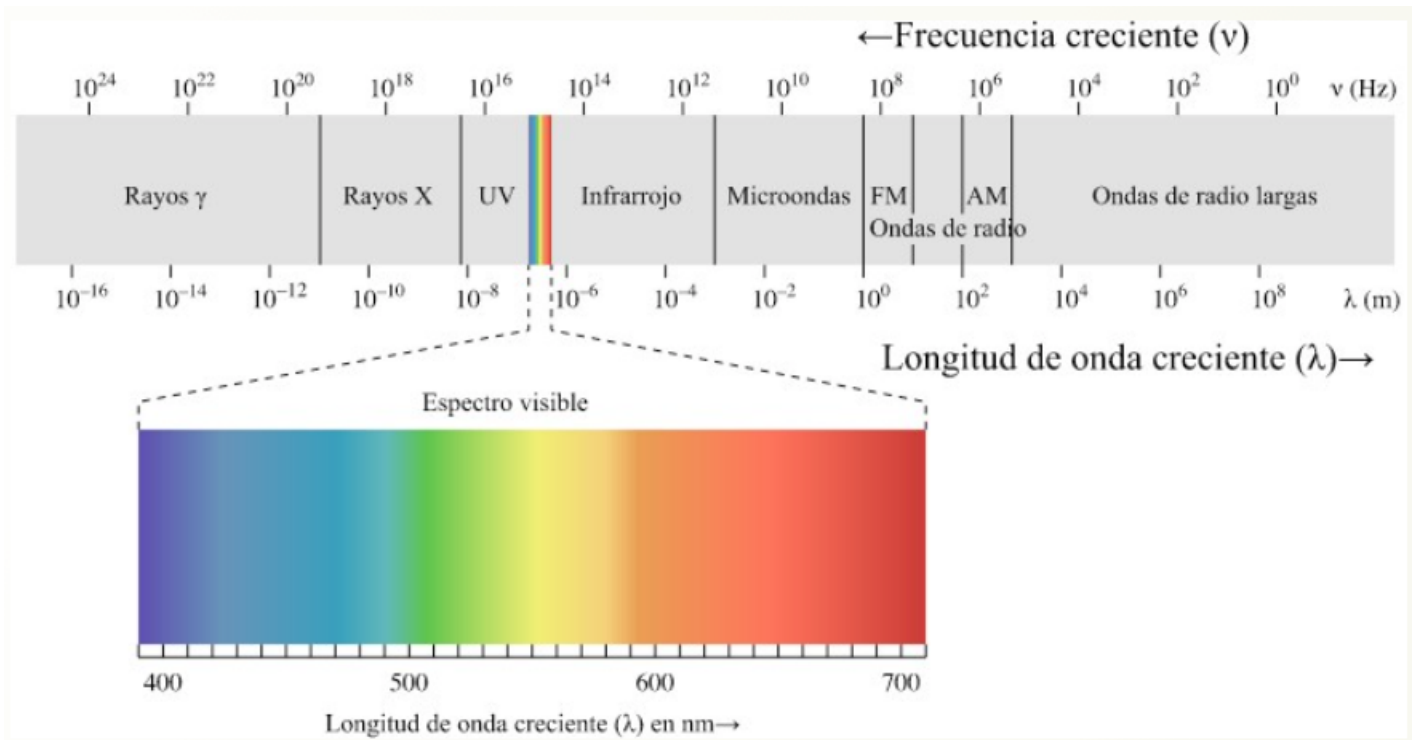
Una onda es una señal que se propaga por un medio. Por ejemplo el sonido, que es una onda mecánica que viaja usando el aire o cualquier otro material. Pero en el caso de las señales eléctricas pueden ser enviadas por el cable o a través del vacío (no necesitan un medio para transmitirse).

Dependen de 3 parámetros principalmente:

- **Amplitud:** altura máxima de la onda. Hablando de sonido representaría el volumen. Si nos referimos a una onda eléctrica estaríamos representando normalmente el voltaje.
- **Longitud de onda λ :** distancia entre el primer y último punto de un ciclo de la onda (que normalmente se repite en el tiempo).
- **Frecuencia f :** Número de veces que la onda repite su ciclo en 1 segundo (se mide en hertzios).
- **Periodo T** es simplemente es la inversa de la frecuencia. $T=1/f$

La relación entre ellas es muy fácil pues las ondas electromagnéticas viajan a la velocidad de la luz c y si velocidad es espacio/tiempo luego $c = \lambda/T$ luego **$c = \lambda * f$**

Dentro del espectro electromagnético encontramos diferentes tipos de señales dependiendo de las características de su onda.



TRANSMISIÓN INALÁMBRICA: BLUETOOTH.

- Hoy en día, este grupo está formado por miles de empresas y se utiliza no sólo para teléfonos sino para cientos de dispositivos.
- Bluetooth es una red inalámbrica de corto alcance pensada para conectar pares de dispositivos y crear una pequeña red punto a punto, (sólo 2 dispositivos).
- Utiliza una parte del espectro electromagnético llamado "**Banda ISM**", reservado para fines no comerciales de la industria, área científica y medicina. Dentro de esta banda también se encuentran todas las redes WIFI que usamos a diario. En concreto funcionan a 2,4GHz. (Un G son 10^9) luego entre FM y Microondas.

¿Sabías que?

Su curioso nombre viene de un antiguo rey Noruego y Danés, y su símbolo, de las antiguas ruinas que representan ese mismo nombre.

Hay 3 clases de bluetooth que nos indican la máxima potencia a la que emiten y por tanto la distancia máxima que podrán alcanzar:



CLASE	POTENCIA	DISTANCIA
Clase 1	100 mW	100 m
Clase 2	2,5 mW	10 m
Clase 3	1 mW	1 m

También es muy importante la velocidad a la que pueden enviarse los datos con este protocolo:

Versión	Velocidad
1.2	1 Mbps
2	3 Mbps
3	24 Mbps
4	24 Mbps

Mbps : Mega Bits por segundo. MBps: Mega Bytes por segundo.

kb = 1.024 b M = 1.024 k G = 1.024 M

¿Te atreves a calcularlo ?

¿Cuántos ciclos por segundo tendrán las ondas que están en la **Banda ISM**? ¿Cuál es el periodo de esas ondas?

Solución

a) $f = 2.4\text{G}$

b) $\lambda = c/f = 12.5\text{cm}$ o sea, las antenas tendrían que ser de esta longitud. Hay muchos trucos para reducirla, una de ellas es la forma de serpiente que puedes ver en el HC-06

¿Te atreves a calcularlo...?

¿A qué distancia y cuanto tiempo tardarían en enviarse los siguientes archivos por Bluetooth?

1. Un vídeo de 7Mb usando versión 2 clase 2
2. Una imagen de 2.5Mb usando versión 3 clase 1
3. Un archivo de texto de 240KB usando versión 1.2 clase 1

Solución

1) $7\text{Mb} / 3\text{Mbps} = 2.3 \text{ seg.}$

2) $2.5\text{Mb} / 24\text{Mbps} = 0.1 \text{ seg.}$

3) $240 \text{ kB } 8\text{b/B} = 1.920 \text{ kb}$ $1.920 \text{ kb} / 1.024 = 1.875 \text{ Mb}$ $1.875\text{Mb} / 1\text{Mbps} = 1.875 \text{ seg.}$

¿Bluetooth clásico o Bluetooth Low Energy = BLE?

Es un protocolo similar al clásico Bluetooth pero diseñado a consumir menos potencia manteniendo funcionalidad. Su popularidad ha crecido en multitud de dispositivos

En robótica, el clásico device que utiliza BLE es la **Micro:bit**. Aunque la Micro:bit no tiene Wifi integrada, posee una radiofrecuencia que podemos configurar para Bluetooth (hay que elegir, o utilizar sus comandos de Radio o utilizar comandos de Bluetooth)

Por eso a la hora de elegir la APP tienes que tener en cuenta:

- Si acepta Bluetooth clásico o BLE
- Que la APP acepte leer datos desde el robot como enviar

Nosotros hemos elegido uno sencillo que cumple las dos condiciones (hay muchas APPs) [Serial Bluetooth Terminal](#)

Serial Bluetooth Terminal

Kai Morich

Compras en la aplicación

Terminal para los dispositivos conectados en serie con Bluetooth Classic / LE

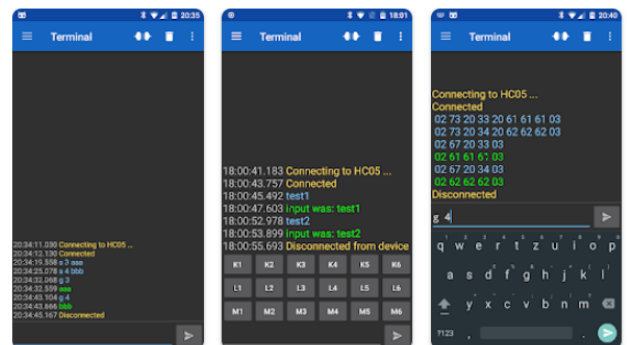
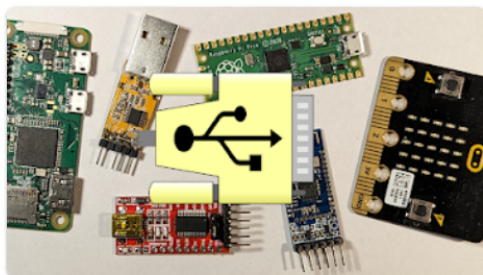


4,6★
3,31 mil reseñas

1 M+
Descargas

PEGI 3

Descargar

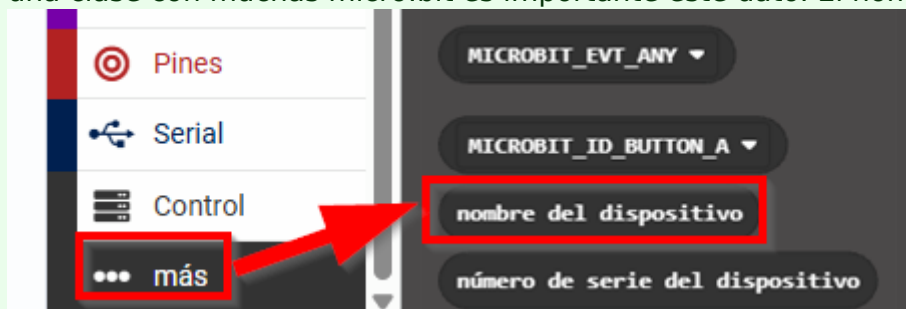


Sólo con la microbit: BLUETOOTH programa en Makecode

Realizamos un programa que :

- En **inicio**
 - se active el servicio UART para el envío y recepción de mensajes,
 - muestra un mensaje del nombre de la micro:bit, ver más abajo
- **Al conectar Bluetooth** que muestre un check
- **Al desconectar Bluetooth** que muestre X
- **Al recibir datos**, hasta # (puede ser otro carácter) que muestre la frase recibida
- **Al presionar el botón A**
 - Que muestre un mensaje
 - Que muestre la temperatura

¿Para qué mostrar el nombre de la micro:bit? Para saber a qué micro:bit conectarte. En una clase con muchas micro:bit es importante este dato. El nombre del equipo está en



<https://makecode.microbit.org/S60585-58735-21378-05922>



<https://makecode.microbit.org/#pub:S60585-58735-21378-05922>

BLUETOOTH programa Serial Bluetooth Terminal

Entramos con el móvil a Google Play e instalamos esta aplicación

https://play.google.com/store/apps/details?id=de.kai_morich.serial_bluetooth_terminal

Serial Bluetooth Terminal

Kai Morich

Compras en la aplicación

4,6★
3,28 mil reseñas ⓘ

1 M+
Descargas

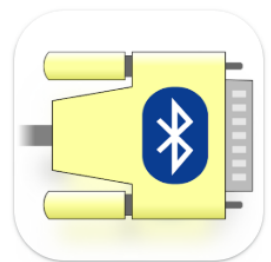
PEGI 3 ⓘ

Descargar

Compartir

Añadir a la lista de deseos

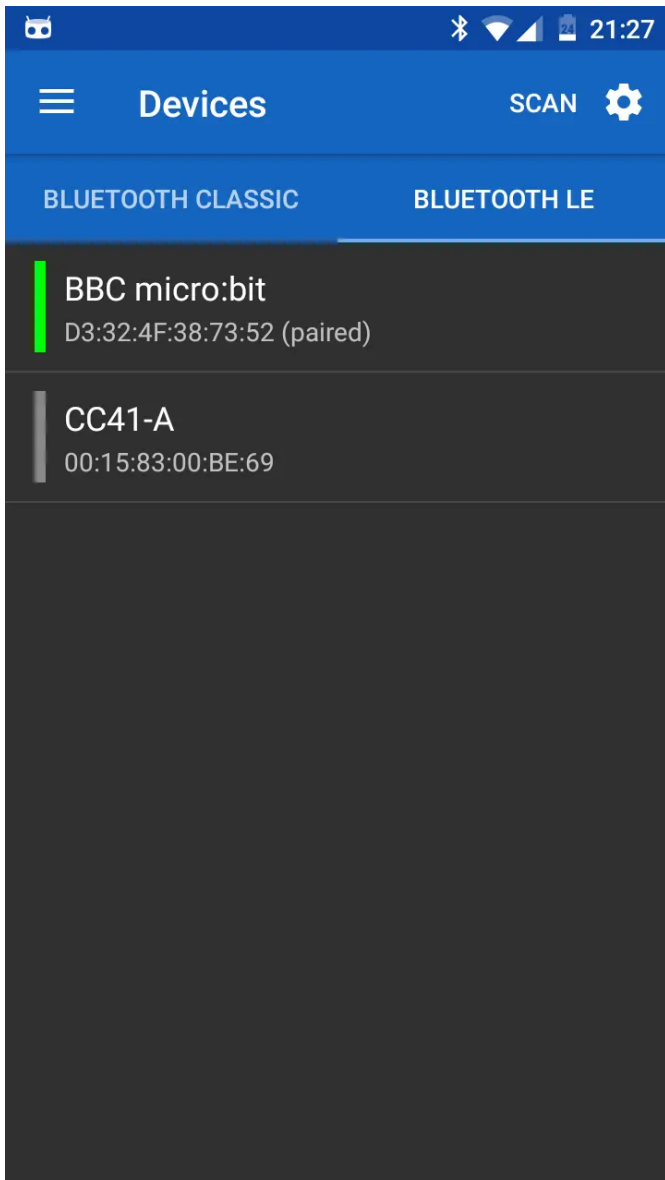
Esta aplicación está disponible para tu dispositivo



Tiene la ventaja de

- **Enviar** mensajes
- **Recibir** mensajes
- Permitir conexiones **BLUETOOTH LE** (Low emission) **que es lo que utiliza MICRO:BIT**

Entramos en **Devices** y en **Bluetooth LE** y nos conectamos a la Micro:bit



Una vez conectado, podemos:

- enviar un mensaje, que como hemos definido anteriormente en Makecode tiene que ir entre #
- recibir un mensaje, se visualizará lo que nos envíe la micro:bit que en Makecode lo hemos programado al apretar el botón A

<https://www.youtube.com/embed/H0HDVPmX-tE>

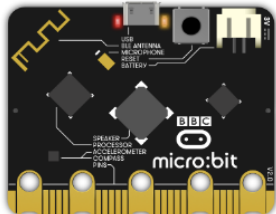
Avanzado: BLUETOOTH con App Inventor. Extensiones

Descargas e instalación

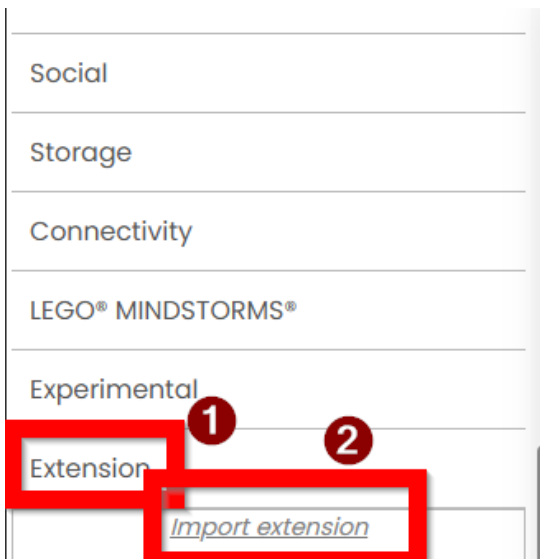
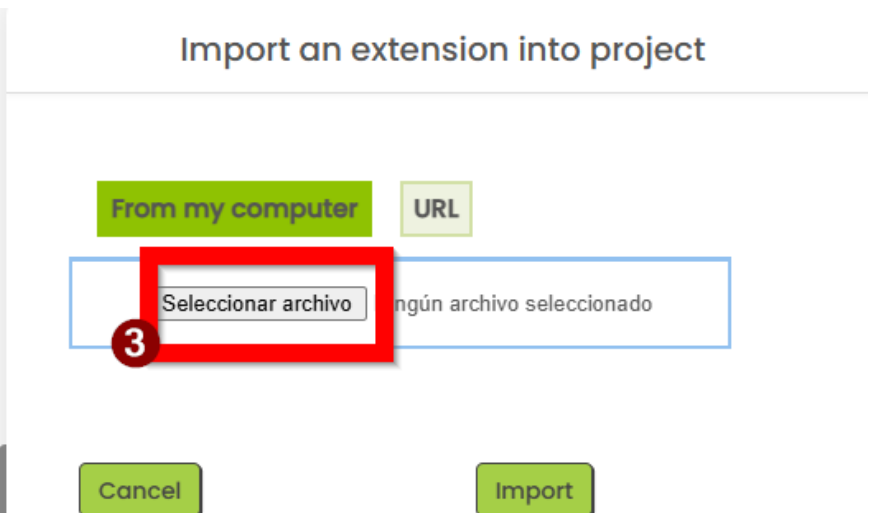
Necesitamos estas extensiones para poder crear una APP que pueda enviar y recibir con nuestra micro:bit

Página de descargas <https://iot.appinventor.mit.edu/#/>

Supported Extensions

Device	Description	Extension
	Bluetooth low energy Bluetooth Low Energy, also referred to as Bluetooth LE or simply BLE, is a new protocol similar to classic Bluetooth except that it is designed to consume less power while maintaining comparable functionality.	Download
	BBC micro:bit The micro:bit is a computing platform from the BBC. It is an open platform for developing all manner of projects and is programmable by many different editors, including a blocks editor provided by Microsoft. Learn more about the micro:bit at the Micro:bit Educational Foundation's website.	Download

Una vez descargadas, vamos al APP INVENTOR <https://ai2.appinventor.mit.edu> y las instalamos en extensiones :



Una vez instaladas, se visualizan como extensiones abajo del menú. Las dos últimas son las que utilizaremos:



Experimental

Extension

Import extension

 Microbit_Accelerometer	 
 Microbit_Button	 
 Microbit_Device_Information	 
 Microbit_Io_Pin_Simple	 
 Microbit_Led	 
 Microbit_Magnetometer	 
 Microbit_Temperature	 
 Microbit_Uart_Simple	 
 BluetoothLE	 

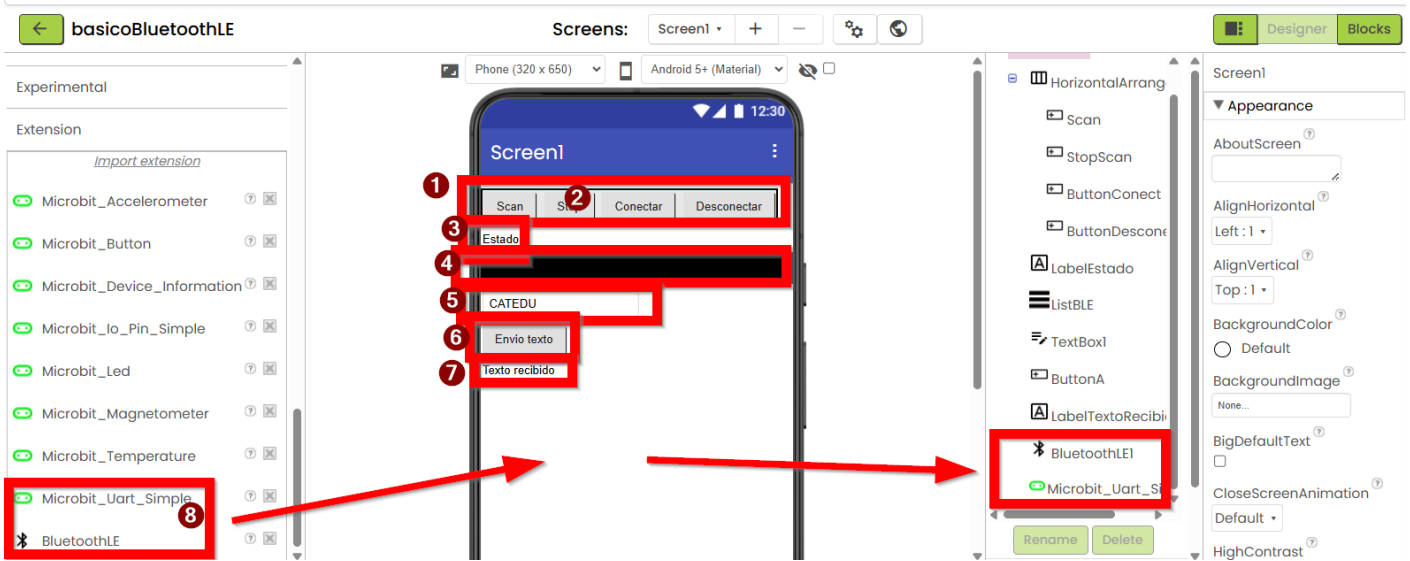
La extensión Bluetooth tiene diversas funciones que tienes su descripción [aquí](#) (English). Para poder instalarla, vamos al APP INVENTOR <https://ai2.appinventor.mit.edu>

Avanzado: BLUETOOTH App Inventor programa. Sólo con la micro:bit

En DESIGNER

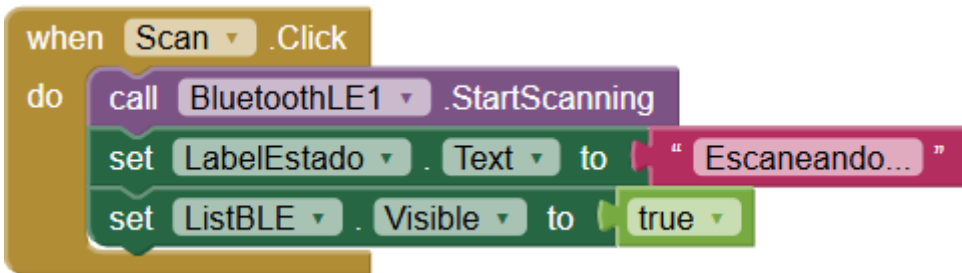
incorporamos:

1. **HorizontalArrangement** para que los botones queden alineados horizontalmente
2. **Botones**
 1. Scan
 2. Stop
 3. Conectar
 4. Desconectar
3. **Label** que dirá el estado de la conexión. Lo llamaremos **LabelEstado**
4. **ListView** que lo llamaremos **ListBLE** donde mostrará los diferentes dispositivos Bluetooth LE que detecta
5. **TextBox** para poner el texto que queramos a enviar a micro:bit
6. Un **botón Enviar** el texto anterior
7. Un **Label** que lo llamaremos **LabelTextoRecibido** que mostrará el mensaje desde micro:bit
8. Añadimos los elementos de las extensiones que hemos instalado anteriormente
 1. Microbit_UART_Simple
 2. BluetoothLE

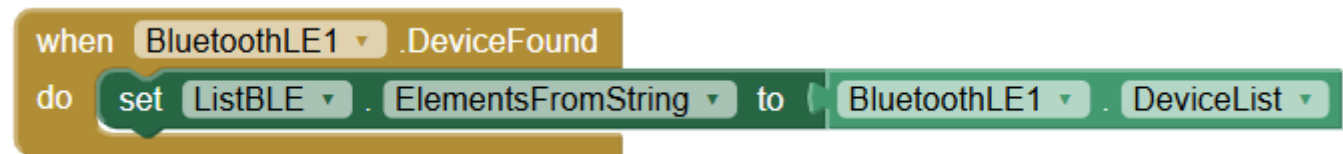


En Blocks

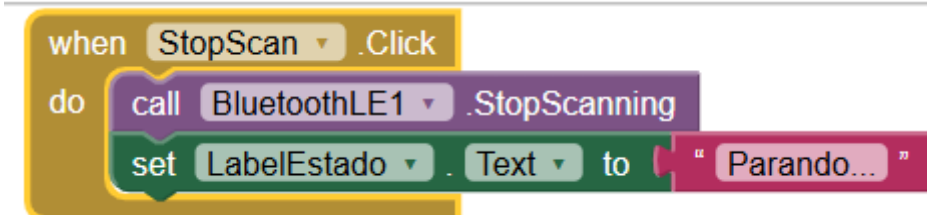
Cuando escaneemos, que el elemento empiece el escaneo y la lista se vuelva visible, además de que LabelEstado diga que esta escaneando



Si ha encontrado un dispositivo, que lo vaya añadiendo a la lista ListBLE

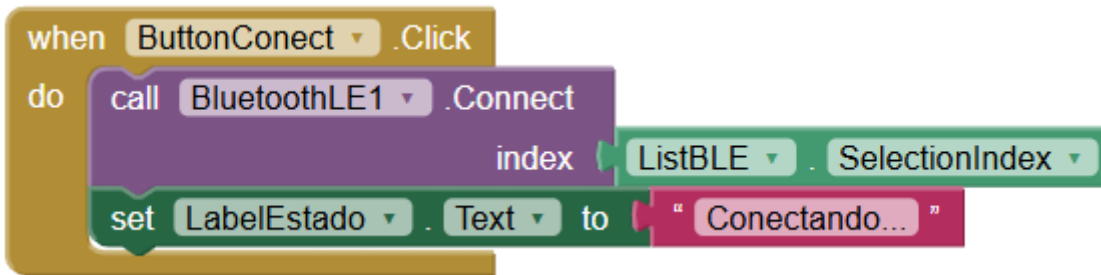


Cuando le digamos que pare, simplemente se lo mandamos al dispositivo y LabelEstado lo informa

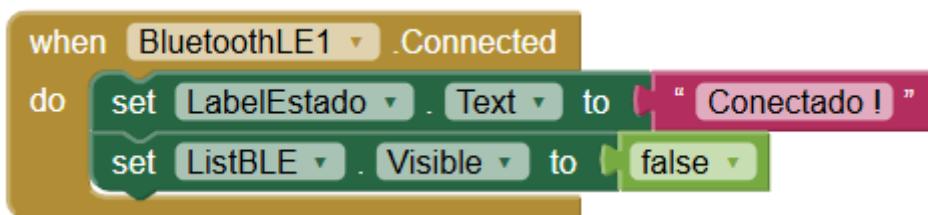




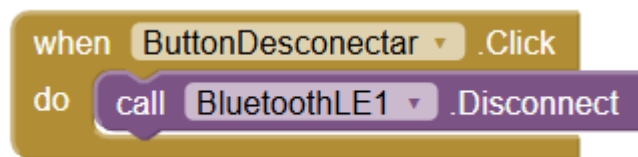
Cuando le demos a conectar, pues conecta con el seleccionado en ListBLE y LabelEstado informa



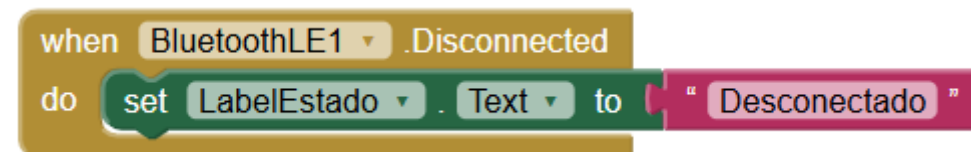
Si conecta, pues LabelEstado informa y ListBLE no es necesaria por lo tanto se oculta, pues entorpece la visión



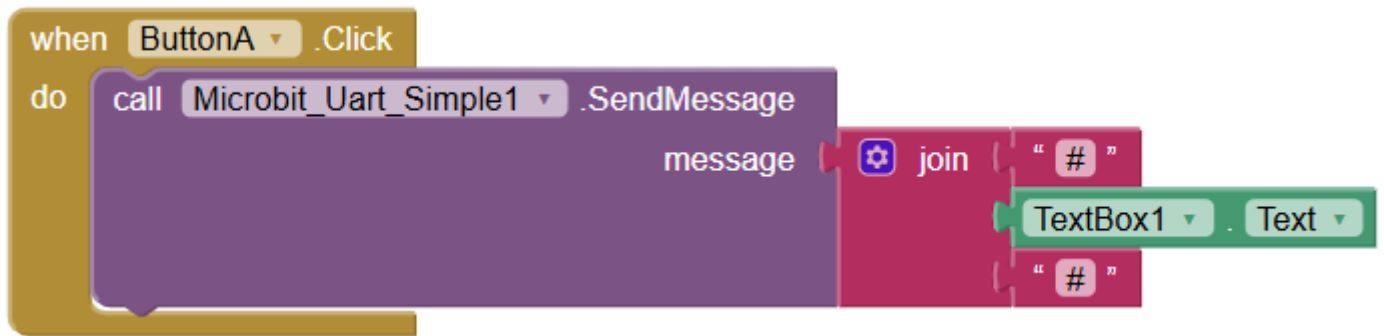
Si queremos desconectar, pues le decimos al elemento BluetoothLE que desconecte



Si se ha desconectado (voluntariamente al dar al botón anterior, o involuntariamente pues el dispositivo se ha desconectado, o esta muy lejos... etc) que informe

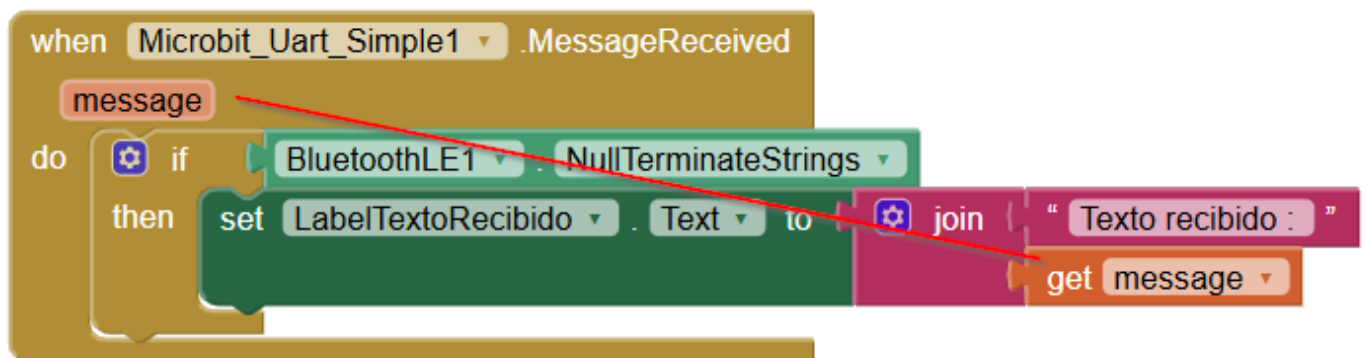


Si apretamos el botón enviar, le enviamos el texto que esta en TextBox entre "#" pues así lo hemos definido en el programa Makecode



Si se ha recibido un mensaje, pues que lo visualice, pero primero comprueba que el mensaje no este vacío

NOTA el mensaje "**message**" lo arrastras desde la instrucción "**when..**" tal y como señala la línea roja



[basicoBluetoothLE.aia](#)

La APP a tu móvil

Tienes dos opciones

- **EN VIVO CONNECT - AI COMPANION** esta opción es la más rápida, y realmente lo simula a través de la APP INVENTOR.
 - Tienes que tener instalada la APP MIT AI2 COMPANION
 - Se le pasa el código de tu APP a la APP
- **OTRAS OPCIONES**
 - Ver <https://appinventor.mit.edu/explore/ai2/setup>

OPCIÓN EN VIVO AI COMPANION

Instalas la [APP MIT AI2 COMPANION](#)

MIT AI2 Companion

MIT App Inventor

Desarrolla tus propias aplicaciones Android usando MIT App Inventor 2!

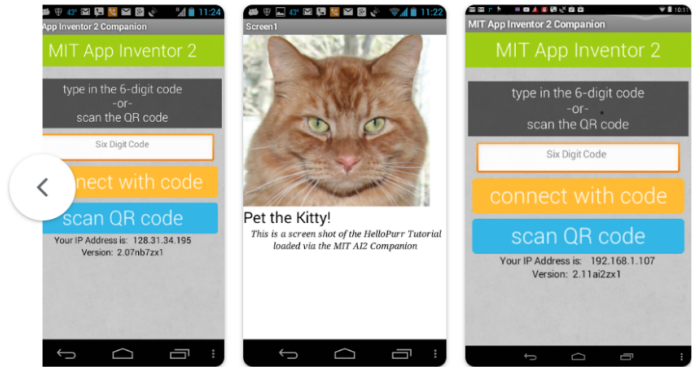


1,9★
28 mil reseñas

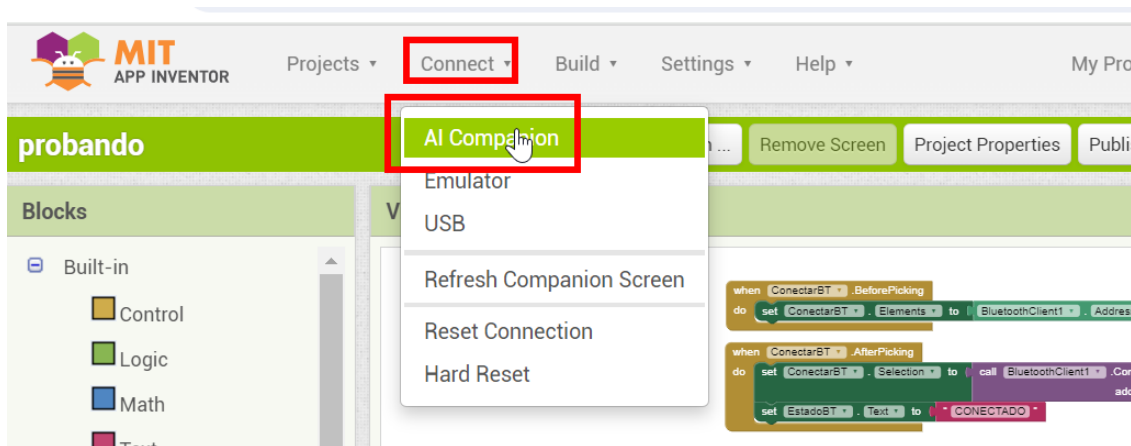
5 M+
Descargas

PEGI 3

Instalar en más dispositivos



En APP INVENTOR



Y sale un código y un QR asociado al código

Launch the MIT AI2 Companion on your device and then scan the barcode or type in the code to connect for live testing of your app.
[Need help finding the Companion App?](#)



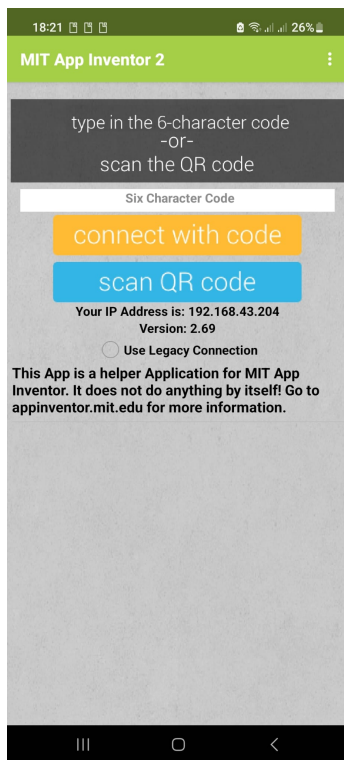
Your code is:

2.11a2zx1

Note: You are on a secure connection, legacy mode on the Companion will not work [More Information](#).

Cancel

Abrimos la [APP MIT AI2 COMPANION](#) y metemos el código anterior (o lo escaneamos con el QR)



En APP INVENTOR verás que sale una barra de progreso enviando tu APP a tu móvil. Cuando termina automáticamente lo ejecuta.

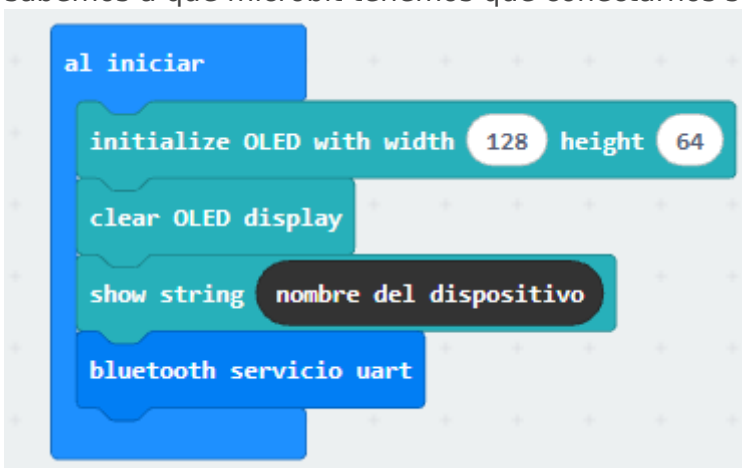
A jugar...

<https://www.youtube.com/embed/ZS5d-xrDVcA>

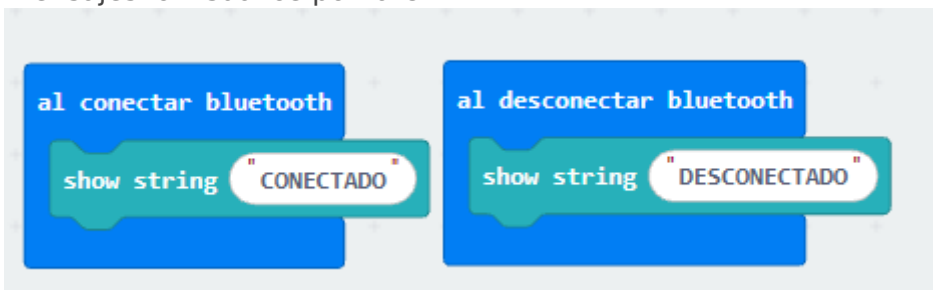
Ahora con la granja

Una vez visto cómo podemos controlar la micro:bit usando Bluetooth, vamos a hacer lo mismo con la granja

1. Empezamos instalando la **Extensión Bluetooth** que hemos visto en las páginas anteriores
2. También pondremos la Extensión **lot-environment** kit de Elecfreak para poder usar la OLED, DHT11 etc.. de la granja tal y como lo hemos trabajado en el capítulo [Programas básicos con Smart Agriculture Kit](#)
3. En el bucle **Al inicial**, inicializamos la **OLED**, y también inicializamos el **Bluetooth** pero aquí el nombre de dispositivo que lo visualice por la OLED que es más cómodo, y así sabemos a qué microbit tenemos que conectarnos si estamos en una clase con muchos:

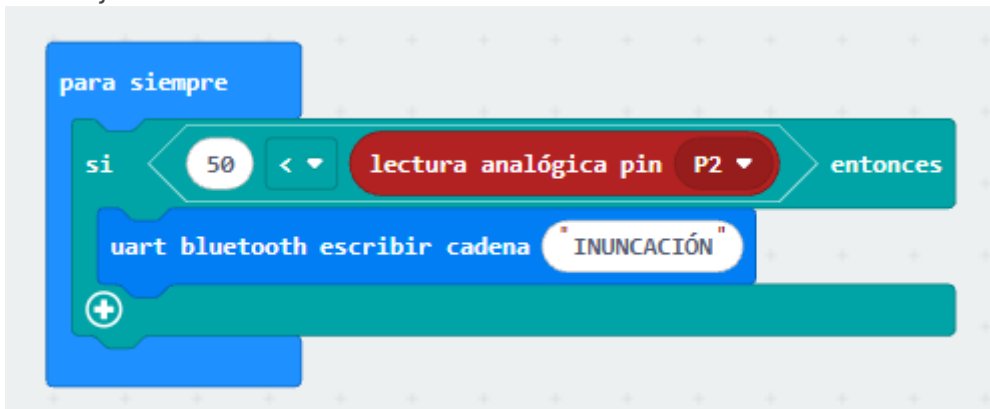


4. Añadimos el bucle **"Al conectar Bluetooth"** y **"Al desconectar Bluetooth"** que también lo hemos visto en las páginas anteriores, pero esta vez le decimos que los mensajes lo visualice por la OLED

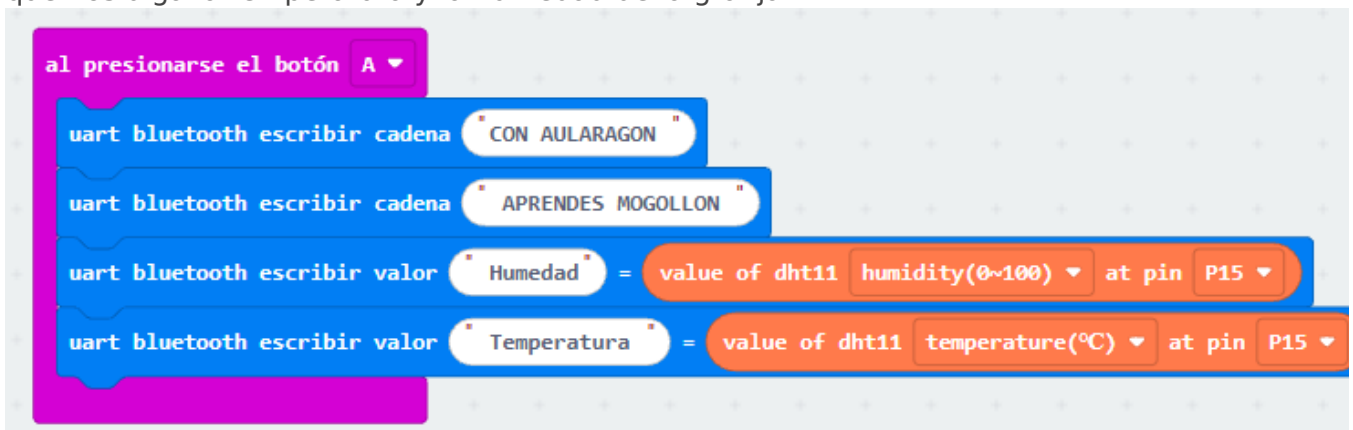




5. Vamos a jugar ahora que si detecta agua en el sensor de nivel de agua, que mande un mensaje de inundación :

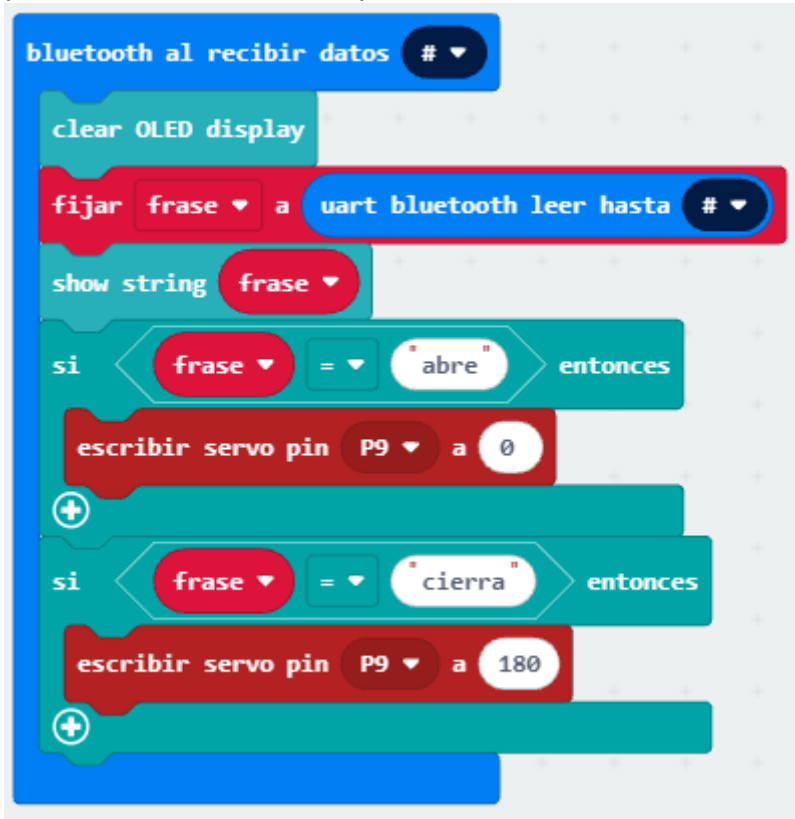


6. Vamos a jugar más y si aprieto el botón A de la micro:bit que salga un mensaje y además que nos diga la temperatura y la humedad de la granja:





7. No podemos abandonar el reto sin poner comunicación en el otro sentido, es decir de la APP a la granja: Vamos a hacer que si se envía desde la APP #abre# pues que el servo se ponga a 0º y se envía desde la APP #cierra# pues que el servo se mueva a 180º. El servo podría estar unido a una puerta, ventana, a un silo de alimentación....



El proyecto <https://makecode.microbit.org/S87280-53966-04815-61300>



<https://makecode.microbit.org/#pub:S87280-53966-04815-61300>

Y el resultado :

<https://www.youtube.com/embed/HJYtbqbrqzw>