

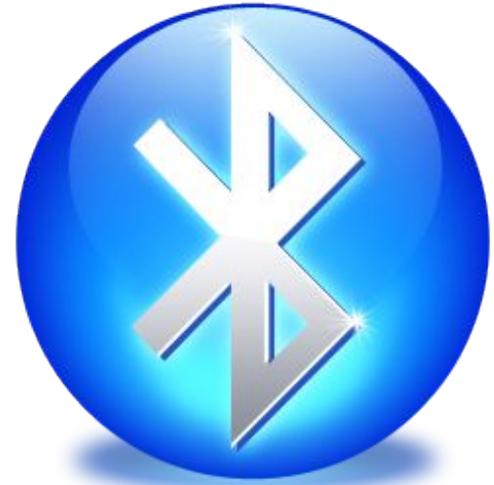


# VISUALINO

APUNTES Y PROYECTOS VI

por Aurelio Gallardo Rodríguez BY – SA – NC

BLUETOOTH



# CONTROL DEL ROBOT POR ANDROID

Para poder controlar el robot printbot mediante android, podemos seguir las instrucciones de la página...:

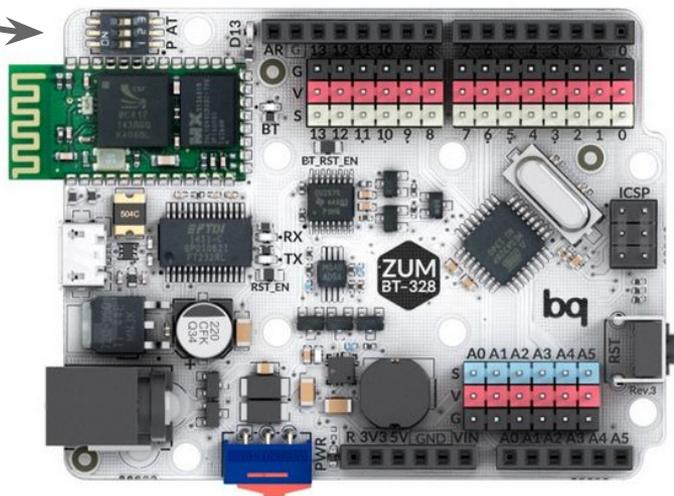
<http://diwo.bq.com/robopad-3/>

En ella, se explica cómo cargar el programa del robot para que acepte comunicaciones bluetooth y cómo usar el programa de comunicación robopad++

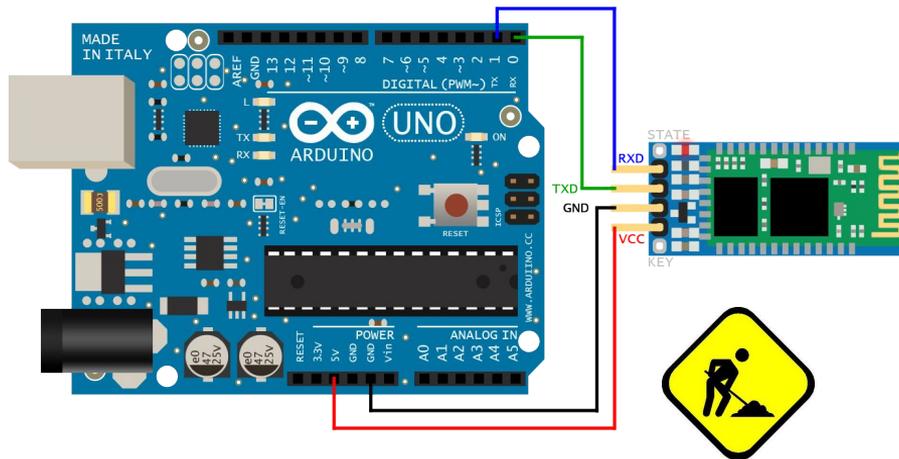
Como no es nuestro propósito el estudio de la programación ANDROID (recomiendo para quien quiera hacer un controlador APP INVENTOR), vamos a intentar controlar en el siguiente programa un servo mediante el mismo programa ROBOPAD++. Sólo programaremos la parte de nuestra placa.

# ¿QUÉ TIENE QUE TENER MI PLACA?

1 ON los demás OFF



ZUM BT - 328 con  
módulo bluetooth



ARDUINO UNO y módulo HC-05 (06)

[http://hetpro-store.com/TUTORIALES/bluetooth\\_hc-06\\_app\\_arduino/](http://hetpro-store.com/TUTORIALES/bluetooth_hc-06_app_arduino/)

# ¿QUÉ MÁS TENGO QUE TENER O HACER?

- Instala la versión más actualizada de ROBOPAD++ desde el PLAY STORE en tu dispositivo ANDROID.
- Conecta en tu android el puerto bluetooth.
- Ejecuta el programa ROBOPAD++ y elige el robot PrintBot
- Conéctate al dispositivo ZUM-BT. El código de conexión será 0000 ó 1234
- Carga el programa en la placa ZUM-BT (Ver ficha siguiente)
- Pulsa los botones que se te indique en el ROBOPAD++

# RECIBIR DATOS DE ROBOPAD++

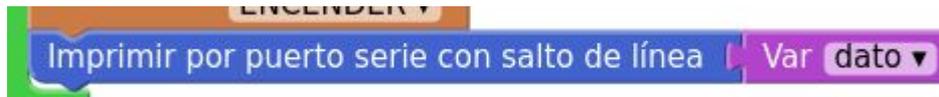
The image shows a Scratch script for controlling LEDs based on data received from a Robopad++ via Bluetooth. The script is as follows:

- Bluetooth** block: Tasa de baudios: 19200, zum?
- Bluetooth: Puerto Serie Disponible** block: ejecutar
- Declarar variable** block: dato = Bluetooth: recibir
- si** block:
  - ejecutar** block: LED PIN# Pin digital 13 ENCENDER
- de lo contrario** block:
  - ejecutar** block: LED PIN# Pin digital 13 APAGAR

Conexión con bluetooth. A 19200 baudios y activar ZUM

ROBOPAD envía las letras U,D,L,R para los movimientos; S para parar, y M,B,G,I para otras operaciones PERO codificadas según el código ASCII. En el ejemplo, al pulsar abajo "D" se envía su código ASCII 68, lo cual se aprovecha para encender o no un LED.

# POSIBLES PROBLEMAS



ARDUINO lleva “mal” el establecer más de una comunicación serie, por lo que posiblemente este comando no funcione o incluso pueda hacer que se bloquee el dispositivo. En ese caso, habrá que “desenchufar” el cable USB, hacer las modificaciones pertinentes y volver a cargar los programas. A veces, si hay una conexión serie establecida que implique los pines 0 y 1 (RX y TX), ARDUINO no carga bien los programas y puede dar error. El puerto bluetooth parece que internamente usa ambos pines.

# PROYECTOS Y AMPLIACIONES

- **Proyectos usando ROBOPAD++:**
  - Crear un programa que pulsando los botones en ROBOPAD++ arriba, izquierda, derecha, abajo, me enciendan cuatro leds diferentes.
  - Crear un programa que que pulsando los botones en ROBOPAD++ controle el avance, retroceso y giro de los servos del robot.
- **Proyecto y ampliación: manejo del robot usando programación APP INVENTOR**
  - [ai2.appinventor.mit.edu](http://ai2.appinventor.mit.edu)
  - <http://diwo.bq.com/course/curso-de-introduccion-a-mit-app-inventor/>
  - <http://diwo.bq.com/controlando-un-printbot-con-botones-en-app-inventor/>