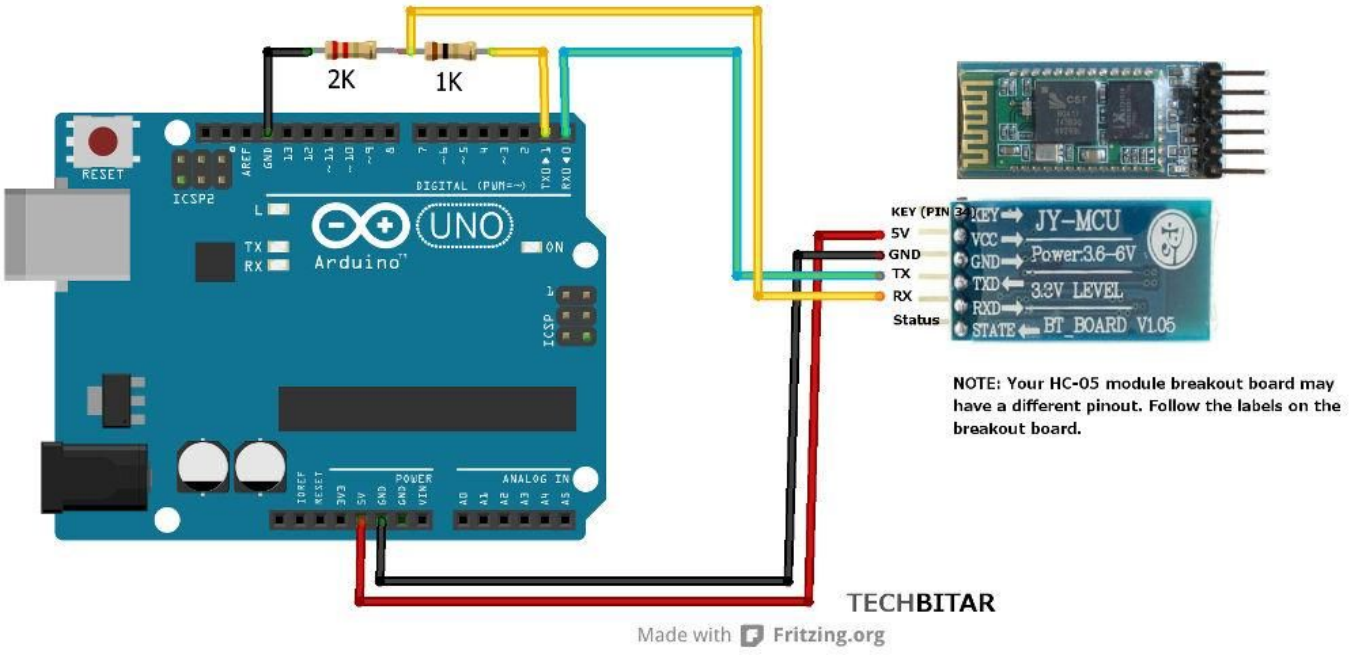
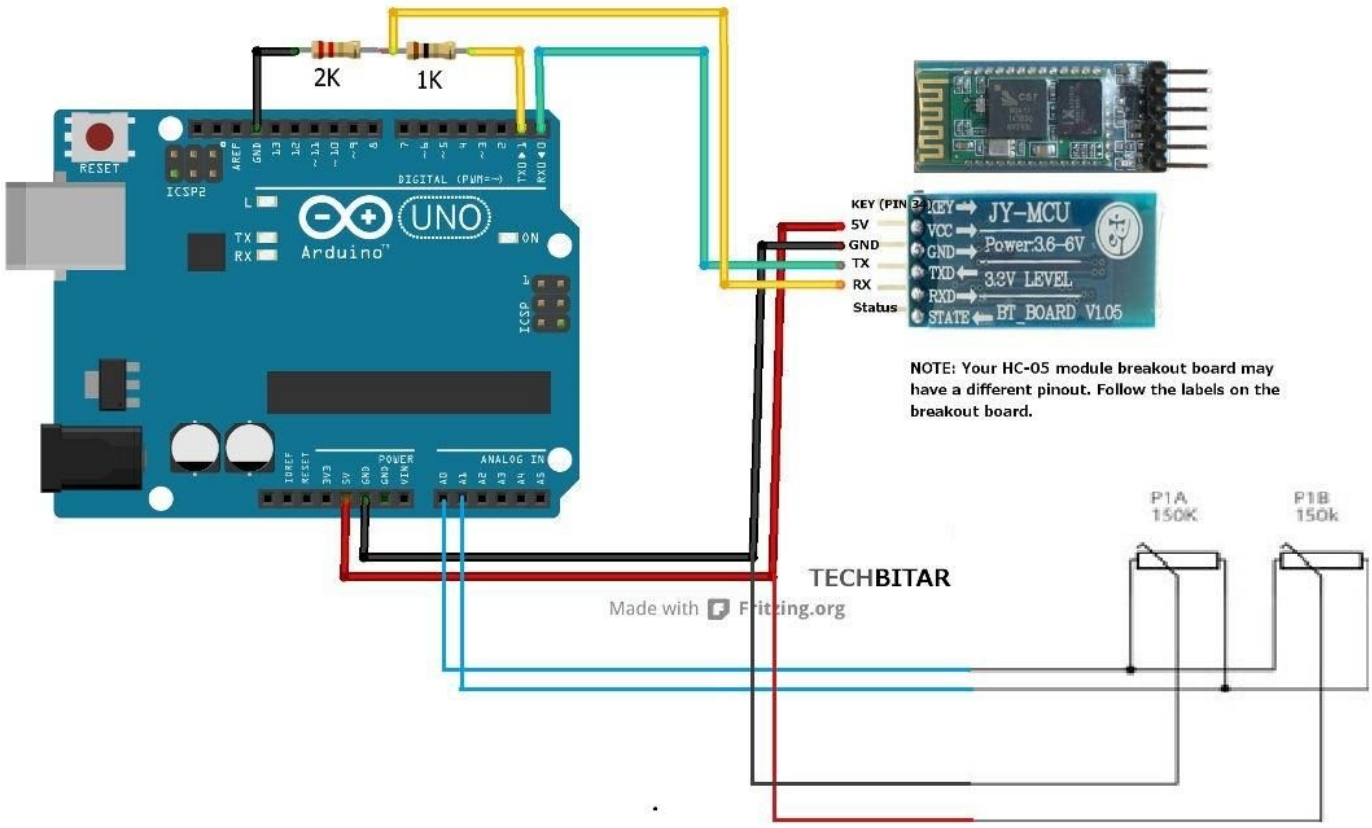
Bluetooth Joystick Example

Pre-requisites : Please don't start using this Asset by reading this tutorial. You should first understand the "Basic demo" and the "High Bit Rate demo".

1) connect your Arduino UNO to the HC-05 . the one in the picture is the JY-MCU HC-05 Bluetooth, which will ease the connection between the HC-05 board and Arduino UNO.



2) connect the Joystick, [read more about Arduino and joysticks.]



3) Upload the sketch in **BluetoothJoystick** folder to Arduino.

- when uploading the sketch you need to disconnect Rx and Vcc pins of the HC-05.

Arduino Sketch [BluetoothJoystick.ino sketch]

We will make Arduino sends the Joystick readings as Array of Bytes.

The sketch reads each Joystick value using **analogRead()** which returns a value between 0 and 1023, to represent (0 to 1023) we need 10 bits because of (2^10 =1024). Because we have to use bytes, we will need 2 bytes to save the returned value of **analogRead().** Because we have two axis/readings and 2 bytes for each axis/reading, we need a byte array of size 4.

We also need a Byte that can separate the different packets, That byte must be unique and different from the JoyStick readings. We can choose 255 (= 11111111) after doing the following trick :

By exploiting the the values in the range 0 .. 1023 :

- 0 to 1023 needs 2 bytes and 10 bits of those bytes will be used (2^10 =1024) :

  Int Value = xx xx xx xx | xx 00 00 00

  Where x is a used bit (either take 0 or 1) and the others are always 0s.

  Now the first byte = xx xx xx xx , and The second byte = xx 00 00 00

- By shifting the variable 'Value' 3 bits, it will become :

  Value = 00 0x xx xx | xx xx x0 00

  We can notice that neither the first or the second byte can have the value 255 or (11 11 11 11). There will be always 3 zeros (000) in each byte. Of course we need to shift them back in Unity.

Unity Scripting [JoystickHandler.cs script]

We need to read the bytes coming from Arduino.

After connecting to the Bluetooth Module using one of the connect methods, you will be able to start reading the data that are coming from Arduino. Please check the source code under the Name "**JoystickHanlder.cs**" in this folder, I added very detailed comments.

The **JoystickHanlder** class reads the Joystick axis values from Arduino and converts each one into the range -1 to 1, similar to the returned value of Input.GetAxis in Unity.

So check the followings :

https://docs.unity3d.com/ScriptReference/Input.GetAxis.html

https://unity3d.com/learn/tutorials/topics/scripting/getaxis