

## BITalino.com Offers



### Software and APIs for data acquisition

<http://bitalino.com/en/software>  
<http://bitalino.com/en/development/apis>



### Detailed documentation to make the most of your device

<http://bitalino.com/en/learn/documentation>  
<http://bitalino.com/en/learn/examples>

## Follow Us



[github.com/  
BITalinoWorld](https://github.com/BITalinoWorld)



[plus.google.com/  
+BITalinoWorld](https://plus.google.com/+BITalinoWorld)



[facebook.com/  
BITalinoWorld](https://facebook.com/BITalinoWorld)



[twitter.com/  
BITalinoWorld](https://twitter.com/BITalinoWorld)



## **IoT** Quickstart Guide



Have any questions? Visit:

[www.bitalino.com](http://www.bitalino.com) for software and documentation | [forum.bitalino.com](http://forum.bitalino.com) to share your thoughts and projects | [bitalino@plux.info](mailto:bitalino@plux.info) to talk with our team

## 1 Start Here

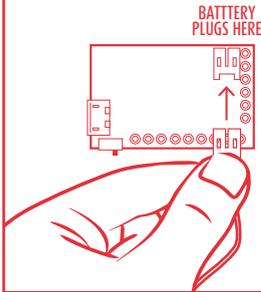
BITalino R-IoT is pre-programmed to connect to a network called "riot".



Data streams as Open Sound Control (OSC) messages to IP address 192.168.1.100 through port 8888.

## 2 Connect Battery

Plug the battery onto the socket found on the R-IoT block.



A green LED should light up on the R-IoT signaling that it is connected to the WiFi network.

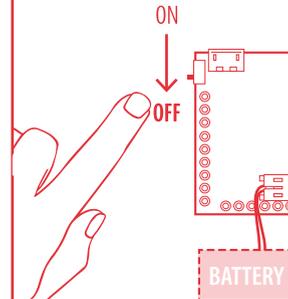
## 3 Have Fun!

At this point the BITalino R-IoT is already streaming data over UDP and ready to use.

One of the tools you can use to acquire data from the BITalino R-IoT is **OpenSignals**, (as described in the next steps), but any OSC-compatible software will work.

By default BITalino R-IoT is configured to connect to a computer with the fixed IP address 192.168.1.100 on port 8888 over a network with SSID named "riot". These settings can be changed as follows:

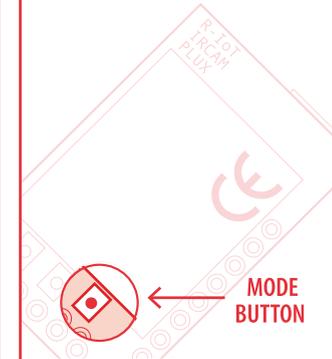
## 1 Turn Off the R-IoT



Slide the switch to the OFF position.

## 2 Set the R-IoT as Access Point

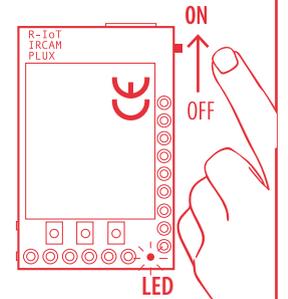
Press and hold the "mode" button.



Press the "mode" button.

## 3 Turn On the R-IoT

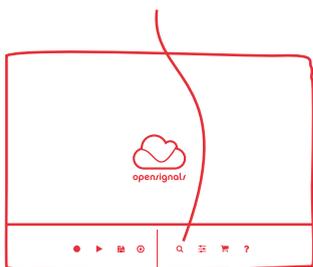
Turn on your R-IoT device; the LED will light up green and then become static.



The device is now a WiFi access point.

## A Launch OpenSignals

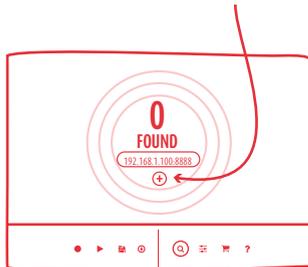
Click on the browse icon.



This will enable you to configure your BITalino R-IoT for data acquisition on OpenSignals.

## B Enable Device

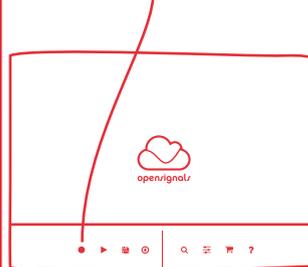
Press the button to manually add a device.



On the text box type 192.168.1.100:8888 and then press enter.

## C Acquire Data

Press (record).



You should now be able to see the data stream coming from the device

After stopping your recording, data can be saved through the button; please refer to the OpenSignals manual to learn all about its features.

## 4 Search for a R-IoT SSID

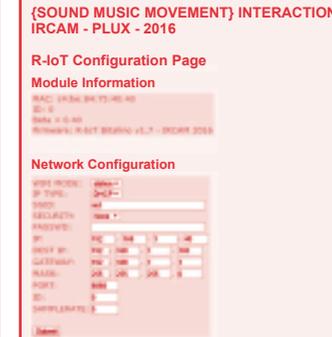
Search and connect for a WiFi network named RIOT-XXXX (xxxx are 4 random characters).



By default the R-IoT WiFi network is open.

## 5 Setup Page

The following page will appear.



Here you can change the SSID, IP and other settings.

## 6 Have Fun!

Click on "Submit" button and then restart your device.



You're ready to go!

1. Download and install Energia version 17 from [energia.nu/download](http://energia.nu/download)
2. Modify the cc3200.ld file and change the HEAP\_SIZE value to 0x00007500.
  - Windows: this file is located in C:\Program Files(x86)\energia-0101E0017\hardware\cc3200\cores\cc3200
  - Mac OS: right-click on the Energia application and select "show package contents" then go to Contents/Resources/Java/hardware/cc3200/cores/cc3200
3. Additionally, the WiFiudp.cpp source file must be modified (hardware\cc3200\libraries\Wifi), as its function parsePacket() blocks during 10ms. The code has been modified to reduce this timeout to 1ms (change timeout.tv\_usec = 1000 instead of 10000) to maintain a low latency and a 200Hz update rate of the IMU streaming.
4. We also provide a pre-configured version of Energia 17 (Windows and MacOS) for an easier setup of the environment; check out the root of the following GitHub repository: [github.com/BITalinoWorld/firmware-bitalino-riot](https://github.com/BITalinoWorld/firmware-bitalino-riot)
5. Get the SLFS library from [github.com/Ircam-R-IoT/SLFS](https://github.com/Ircam-R-IoT/SLFS) and copy it into Documents/Energia/libraries
6. Get the BITalino Energia library from [github.com/Ircam-R-IoT/bitalino-energia-library](https://github.com/Ircam-R-IoT/bitalino-energia-library) and drop it into Documents/Energia/libraries
7. Open the firmware.ino file with Energia and click the "Verify" button in the upper left corner. If it compiles without errors, you're ready to upload your own code to the BITalino R-IoT board.

To upload your firmware the following steps are needed:

### 1 Connect Battery

Plug the battery onto the socket found on the R-IoT block.

### 2 Turn On the R-IoT

Turn on your R-IoT device.

### 3 Plug USB Cable

Plug the USB cable onto the connector found on the R-IoT block.

### 4 Open Energia

Go to the **TOOLS** menu.

Connect to the R-IoT's serial port.

### 5 Press Buttons

Press the **RESET** and **FLASH** buttons simultaneously.

Keep holding both buttons - **DO NOT RELEASE**.

### 6 Upload Firmware

Click the **UPLOAD** button.

You will see a "Compiling Sketch" message. Hold both buttons until it changes to "Uploading" then release **only the RESET** button.

Keep pressing **FLASH**.

### 7 Done Uploading

You should see some normal debugging messages.

This output is expected and does not compromise the firmware uploading process.

### 8 Release Flash Button

You can now release the **FLASH** button.

Wait for 5 seconds after releasing the button and then proceed to the next step.

### 9 Reset the Device

Press the **RESET** button, wait for 5 seconds and press the **RESET** button again.

You're ready to R-IoT!