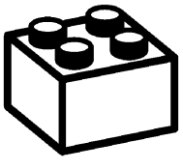
User’s Manual

Brickz Bluetooth Infrared Blaster



Introduction

Brickz Bluetooth IR Blaster is Platform Independent Bluetooth Infrared Transmitter / Blaster , compatible with GNU Linux, MS Windows, MAC OS X and Android. The module features Bluetooth SPP (Serial Port Protocol) and USB interface connection with the host system.The module design is based on powerful MCU, which offers great customization of the firmware due to its support with the Arduino Development environment. Brickz has two powerful Infrared Leds and a 3.1mm mono jack for connecting Infrared Extender cable. The code format that module uses is the widely used PRONTO HEX Infrared code format, which provides flexibility and makes it universal.

Module Features:

- Bluetooth communication with the HOST SYSTEM / MCU
- USB Connection with the host System
- Built-in high power infrared leds
- Optional infrared transmitter extender
- Infrared operates at 940nm wavelength

Example Application

- Digital signage infrared remote control & automation
- Home Automation & remote control
- Media remote control & automation systems

Device Setup & Pairing

Brickz is shipped with a USB-A to USB-B cable which by default can power Brickz by any USB port available nearby (Including USB Chargers, Computer USB Ports etc). Once you connect it via USB, the unit will bootup and it will be listening for new Infrared codes both from its USB and Bluetooth communication interfaces.

Powering the device is also possible via a standard 2.1mm DC Power Adapter with rating 9-12V DC. In case you use a DC Adapter then you dont need the USB connection for power.

If you plan to hide brickz behind a TV or any other object, you can use the infrared extender cable that comes with the package. The Infrared Extender head has a double sided 3M skotch tape which will allow you to fix the IR head next to the TV or near your target remote control device.

The Brickz bluetooth IR blaster uses a standard bluetooth SPP serial communication protocol, which enumerates on the host system as a serial port. To connect to Brickz over bluetooth enable the bluetooth connection on your computer / smartphone and open bluetooth settings. Wait for the device discovery to complete and select the Brickz bluetooth name from the list of the discovered devices. You will be asked to enter a bluetooth pairing code - the pairing code is **1234.**

Once the pairing is complete and the connection established you will be able to communicate with the device by rfcomm or via the newly created serial port on the host system (See examples below).

Android

Brickz for android example application is open source software and its source code and binary releases (APKs) can be downloaded from Irdroid's GitHub repository dedicated for the Brickz Project (links at the end of the Manual). The application provide means for connecting to brickz and sending predefined PRONTO HEX codes. These codes are freely available and can be downloaded online from the remote-central infrared database or any other infrared database on the internet. By default the application is hardcoded with Samsung TV Discreate PRONTO HEX codes, fro turning TV on and OFF etc.

MS Windows

In MS Windows you can pair and connect to Brickz via bluetooth and once the pairing procedure is completed a new virtual serial port will show up on your system (You can identify the serial port name by visting Control Panel-> Device Manages -> System). Once you identify the serial port name, you can use serial terminal applications such as RealTerm, Putty or any other serial terminal to connect to Brickz's serial port and send PRONTO HEX Codes via serial

The default serial settings are 9600 8 n 1 . These settings need to be entered into the serial terminal application that you use.

GNU Linux

In Linux, you can pair and connect to Brickz and use rfcomm in order to enumerate Brickz on your system as a serial port and be able to use any serial terminal application such as gtkterm in order to send pronto ir codes to brickz.

1. Use the bluetoothctl utility to scan for Brickz and obtain its bluetooth mac address so that you can use it to connect via rfcomm :
    a. first enter bluetoothctl
    b. scan on

    c. after the scan completes you will see a list of available devices with their MAC addresses , identify Brickz by name and copy its MAC address which will be used in the next step
2. Use the following command to bind Brickz to /dev/rfcomm0 device on your system:
    a. frcomm bind brickz_bluetooth_mac_address.
3. Once you complete the step above a new device will show up in /dev called /dev/rfcomm0
4. Use Serial Terminal application such as gtkterm to connect to /dev/rfcomm0 9600 8 n 1
5. After completion of the above step you can directly send Pronto HEX codes form the terminal (At the end of this manual you will find links to the most popur PRONTO IR Databases available on the Internet)

MAC OS X

In MAC OS X you have to go through the pairing procedure with Brickz.

1. Enable Bluetooth
2. Open Bluetooth Settimgs
3. Search for Brickz name in the list of available bluetooth devices
4. Pair / Enter Brickz default pairing code - 1234
5. Once Pairing is complete a new device will become available in your /dev folder
6. Open a terminal and type cd /dev ;ls
7. The new device file name should start with try.brickz….

At this point you can send infrared codes directly to Brickz directly from the terminal. Type :

echo -e -n "SEND YOUR PRONTO CODE\n" > /dev/tty.brickz…

And the it command will be sent to Brickz and executed.

Links

- http://www.remotecentral.com/cgi-bin/codes/ remote central pronto hex ir codes database
- https://irdb.tk A website which allow you to generate ir codes in various format depending on the protocol
- https://irdroid.eu/product/Brickz - bricks website
- https://github.com/irdroid/Brickz - Source code & Documentation