How to use BlueController with Arduino environment

Hardware setup (only needed when not already delivered with optiboot bootloader)

1. Burn adapted optipoint bootloader (hardware/optiboot/bootloaders/optiboot/optiboot_bluec328_19200bps.hex)
2. Set fuses and lockbits (refer to hardware/optiboot/boards.txt)
3. Open the sketch “BlueController_Setup”, adapt it to your needs (e.g. change the bluetooth name), upload it and run it once. It is finished after the LEDs has blinked 8 times.

Unsupported: When you want to use a different baudrate, you have to set the parameters of the BTM-222 before flashing the bootloader, because you cannot upload anything using the bootloader when the baudrate between bootloader and BTM-222 doesn't match.

Software setup for Mac OSX

Copy the „hardware“ and demo sketches folders to `Documents\Arduino` in your home directory.

Unsupported: If you are using an ATmega88P based BlueController, you have to merge the contents of the „hardware/optiboot/avrdude/.avrduderc“ file with the “.avrduderc” in your home directory. When it doesn't exist, just copy the one from this distribution. This is a user specific avrdude config file which will be used additionally to the original one.

Software setup for Windows XP / Windows Vista / Windows 7

Copy the „hardware“ and demo sketches folders to `My Documents\Arduino.`

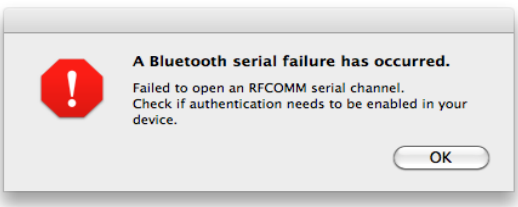
Unsupported: If you are using an ATmega88P based BlueController, you have to copy „hardware/optiboot/avrdude/.avrduderc“ as „avrdude.rc“ to `C:\Windows` or any other directory which is in the system search path. This is a user specific avrdude config file which will be used additionally to the original one.

How to activate the bootloader mode using buttons

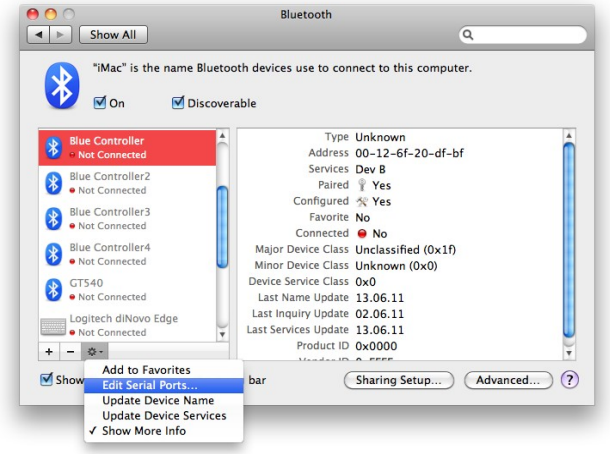
1. Press both buttons (reset + INT0)
2. Release the reset button while still holding the INT0 button
3. Release the INT0 button
4. The LED will light very dim as long as the bootloader is ready to accept uploads

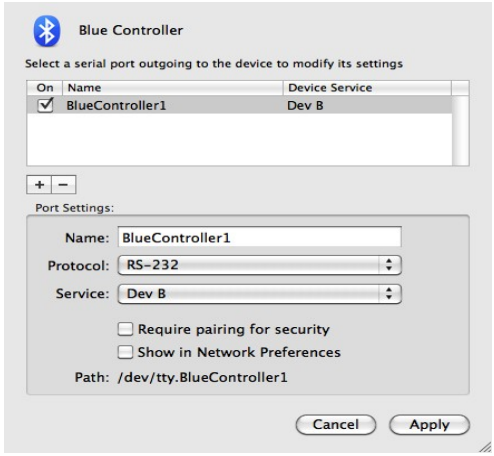
Connection problems on a Mac

Sometimes you might get a connection error box from the Arduino IDE:



Open the Bluetooth System Preferences, select your BlueController device and choose „Edit Serial Ports...“. In the dialog which appears, you can toggle the setting „Require pairing for security“:





Most of the time this cures the connection problem.

How to activate the bootloader mode from your sketch

Look at the files bootloadertools.cpp/.h in the “Serial_and_Blink_BlueController” sketch. Copy these two files to your sketch and put a `#include "bootloadertools.h"` line at the beginning of you main sketch file. Call the function `enter_bootloader()` to activate the bootloader.

How to upload sketches / Hex-files without the Arduino IDE

Put the BlueController in bootloader mode and use avrdude to upload your program, for example:

```
avrdude -p atmega328p -b 19200 -c arduino -P
/dev/cu.BlueController -U flash:w:myprogram.hex:i
```

When using a standard Linux or MacOS (e.g. from www.macports.com) version of avrdude, use the programmer “arduino”. If you use the avrdude version which comes together with the Arduino package[1], you have to use the programmer “stk500v1”.

The device name (`/dev/cu.BlueController`) and hex-filename (`myprogram.hex`) have to be adapted to your environment.

Differences between BlueController and Arduino

The BlueController hardware is more like the Arduino Pro than the Arduino Uno

Here the list of differences

- The BlueController onboard LED is connected to PB6 and not to PB5. The Arduino pin number is 20, so you have to change the LED pin number from 13 to 20 in your Arduino sketches or connect a additional LED (with 1 kOhm current limiting resistor) to PB6.

1/Applications/Arduino.app/Contents/Resources/Java/hardware/tools/avr/bin/avrdude on OSX

- Additional two I/O pins: PB6 and PB7.
    - PB7 is used to control the RESET line of the BTM-222 bluetooth module and cannot be used for any other purpose. The Arduino pin number is 21.
    - PB6 is connected to a LED, but can additionally be used as normal output. The Arduino pin number is 20.
- PB5 is completely free and can be used as input or output. The Arduino Uno uses this pin for its LED.
- Instead of a 16 MHz crystal, the BlueController uses the internal ATmega oscillator with a clock frequency of 8 MHz.
- The voltage is 3.3V instead of 5V
- No auto-reset feature, use the reset button + boot-loader button or the reset-magic mechanism instead
- Reset magic mechanism: Entering the boot-loader can be controlled by your sketch, so you don't need the buttons and can upload you sketch remotely, even if you don't have physical access to the BlueController board. Look at the "Serial_and_Blink_BlueController" sketch for a demo how it works.
- Serial communication speed is fixed to 19200 bps
- Bluetooth interfaces instead of USB serial interface
- Level shifters for some I/Os to make it possible to use 5V hardware
- The Arduino shields cannot be used with BlueController
- The bootloader is different, it uses an enhanced optiboot bootloader

Home of this project

The current version of this distribution can always be found here:
https://code.google.com/r/michaeldreher42-bluecontroller/

Information about the hardware can be found in the online shop:
http://lynx-dev.com/ or http://rf-store.com/

What to do when upgrading to a new Arduino version?

The pins PB6 and PB7 are not supported on a standard Arduino, therefore a different core is used. It is almost a copy of the original Arduino core directory. When you upgrade to a new Arduino version, you might need to update these files.

Actually only one files differs which you can easily adapt yourself by adding 6 lines. Copy the contents of the new Arduino core directory `Applications / Arduino.app / Contents / Resources / Java / hardware / arduino / cores / arduino` to your `Documents / Arduino / hardware / bluecontroller / cores / bluecontroller` directory and apply the following patch:

```
---
hardware/bluecontroller/cores/bluecontroller/pins_arduino.c.orig
     2010-12-24 20:56:11.000000000 +0100
+++ hardware/bluecontroller/cores/bluecontroller/pins_arduino.c
     2011-05-26 00:57:30.000000000 +0200
@@ -400,6 +400,8 @@
     PC,
     PC,
     PC,
```

```
+    PB, /* 20: BlueController LED */
+    PB, /* 21: BlueController BTM-222 RESET */
};

const uint8_t PROGMEM digital_pin_to_bit_mask_PGM[] = {
@@ -423,6 +425,8 @@
     _BV(3),
     _BV(4),
     _BV(5),
+    _BV(6), /* 20: BlueController LED */
+    _BV(7), /* 21: BlueController BTM-222 RESET */
};

const uint8_t PROGMEM digital_pin_to_timer_PGM[] = {
@@ -461,5 +465,7 @@
     NOT_ON_TIMER,
     NOT_ON_TIMER,
     NOT_ON_TIMER,
+    NOT_ON_TIMER, /* 20: BlueController LED */
+    NOT_ON_TIMER, /* 21: BlueController BTM-222 RESET */
};
#endif
```