PiKoder/SSC RX User's Guide

Content

1

Overview

The PiKoder/SSC RX represents an eight channel bluetooth module based on the PiKoder/SSC which has been designed as a replacement of the receiver in an R/C car. In concert with the picCAR-App you can easily control your model with your Android smart phone. Using the Arduino Digital RC for a broader and more professional feature set. The PiKoder/SSC RX features a wide range of power supply (4.0 – 6.5 V) allowing you to simply replace your models current receiver.

This User's Guide describes in section 2 the hardware setup for the PiKoder/SSC RX. The following sections 3, 4, and 5 describe how to establish the Bluetooth connection between the receiver and the controlling computer.

Section 3 outlines details as to how to connect to the receiver from a BT enabled PC running Windows 7 (either with build-in BT or using a USB dongle) for the customization of your receiver to your application.using the PiKoder Control Center (PCC), a graphical user interface for reviewing and setting PiKoder parameters . This step would be optional since the default parameters cover a number of standard applications.

Section 4 then describes how to communicate with an Android tablet with build-in BT. In section 5 you will learn how to connect your PiKoder/SSC to an Arduino using a Bluetooth Shield. Please refer to the exhibits for the pinning and the schematic.

The receiver is based on a PiKoder/SSC programmed with the latest firmware. This means that you can design your own remote control using the protocols and interfaces featured by the SSC. Please refer to the PiKoder/SSC User's Guide for more information.

This User's Guide is based on the most recent hard- and firmware version 2.3 available for the PiKoder/SSC and the related programming software

"PCC PiKoder Control Center". Please check for updated information and new software releases on www.pikoder.com.

Hyperlinks were integrated into the text for convenience. You would also find all downloads referenced on the PiKoder/SSC webpage.

Please share with me any comments, improvement ideas or errors you will find or encounter in working with your PiKoder/SSC RX. I can be reached at webmaster@pikoder.com. Thank you very much!

2

Hardware setup

The PiKoder/SSC RX is based on a single chip solution for implementing a serial servo controller (PiKoder/SSC) which gives you full control of up to eight servos or electronic speed controls with a resolution of 1 µs. To meet the requirement of simple receiver replacement, the PiKoder/SSC has been combined with a power regulator and a Bluetooth TX/RX module.
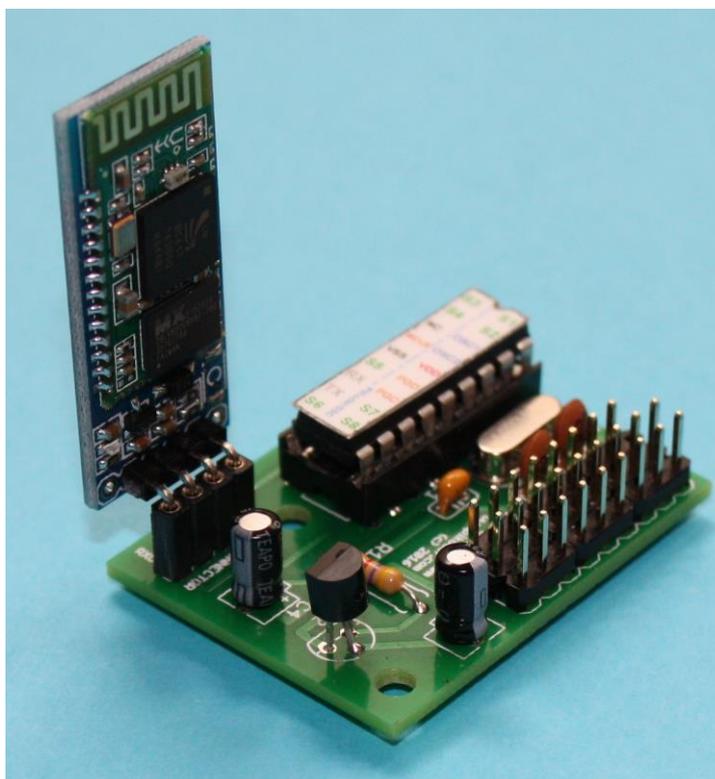

PiKoder/SSC RX fully assembled

In order to commission your PiKoder/SSC RX you would connect a battery or other power source providing a voltage between 4.0 and 6.5 Volts to your module (see below). After turning the power on, the red LED on your Bluetooth module should start to blink indicating that power is provided and that the module is ready for binding to another Bluetooth transmitter/receiver module.
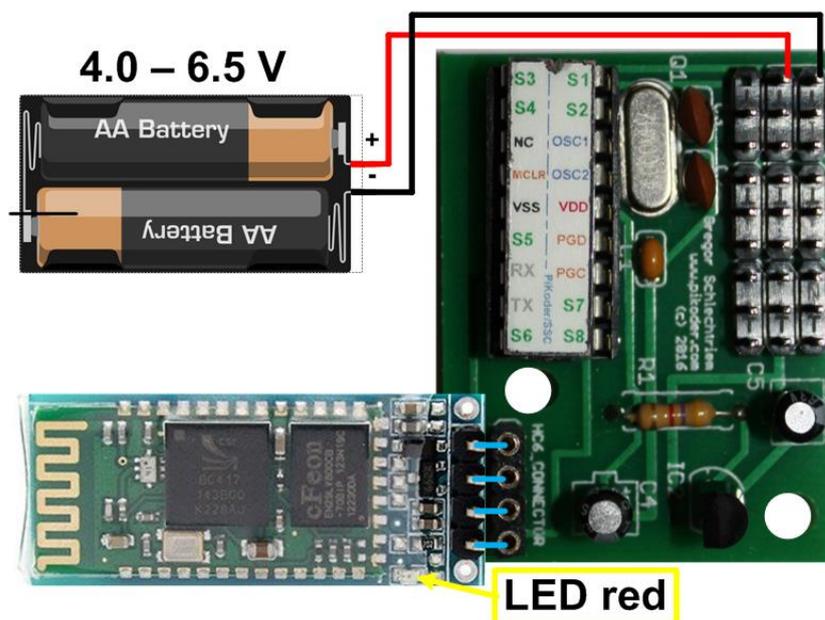
PiKoder/SSC RX wiring

If you were to use your PiKoder/SSC RX with an electronic speed controller (ESC) featuring a battery elimination circuit (BEC) then you would only connect your ESC to one of the channels as shown in the image below (please refer to exhibit A for the pcb pinning). **In this configuration do not connect a separate power supply because this could damage your boards!**
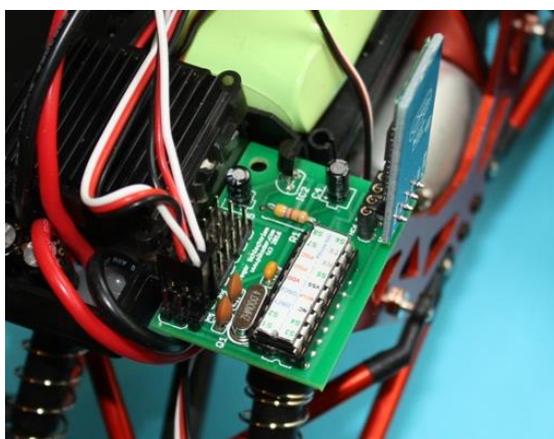
PiKoder/SSC RX wiring using BEC

3

Connecting the PiKoder/SSC to a Windows 7 PC

There are three steps required to establish the Bluetooth based communication between your PC respectively the PiKoder Control Center Application (PCC) for controlling and programming the PiKoder/SSC and the SSC itself:

    a. Installing the Windows driver for the USB Bluetooth dongle (optional)

    b. Paring your doungle to the Bluetooth transmitter/receiver

    c. Connecting the PCC Control Center application to the PiKoder/SSC using a virtual COM port

This chapter will guide you through these steps.

Installing the Windows driver for your USB Bluetooth dongle (optional)

Insert your USB Bluetooth dongle into one available USB port. Windows 7 will recognize the new hardware and install the respective drivers.

The automatic installation does include the components shown below and after the completion you should see a window similar to the one below.
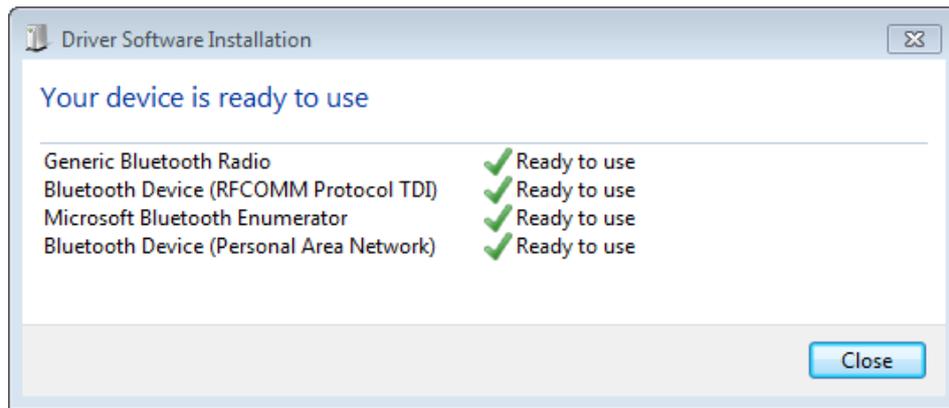
Pairing your dongle to the Bluetooth transmitter/receiver

Pairing is required to establishing a connection between two Bluetooth devices. Your Bluetooth would be in "Discoverable" mode after turning it on and allowing the dongle to discover the module. Once both modules would agree on a key the pairing is completed and the devices would recognize each other and establish a connection.
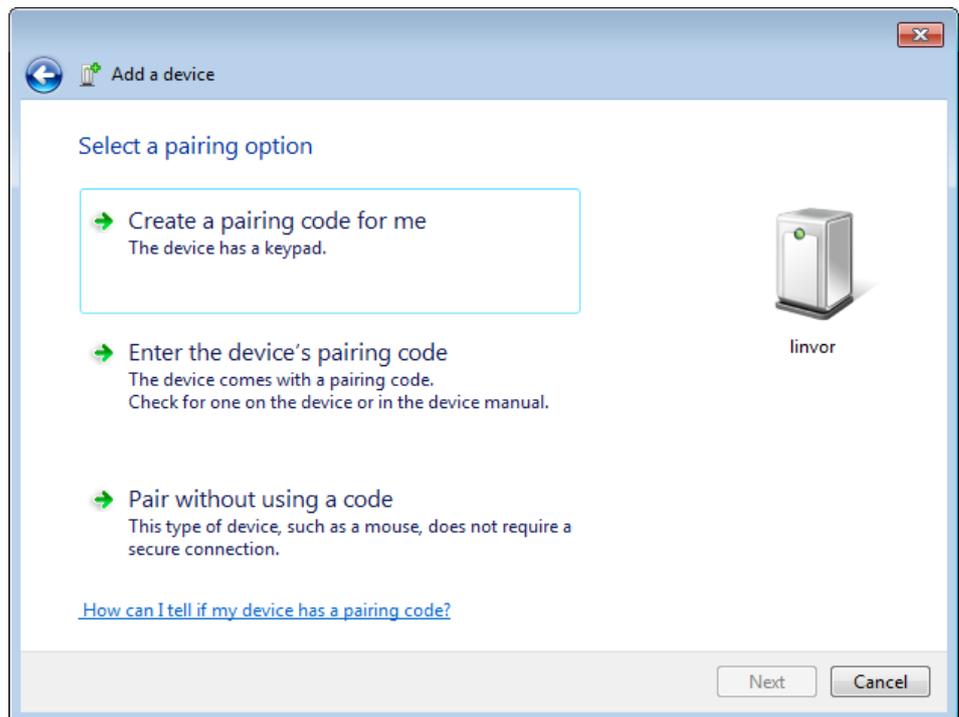
You would initiate this process by supplying power to the PiKoder/SSC RX module and select 'Add a device' at your computer's control panel.
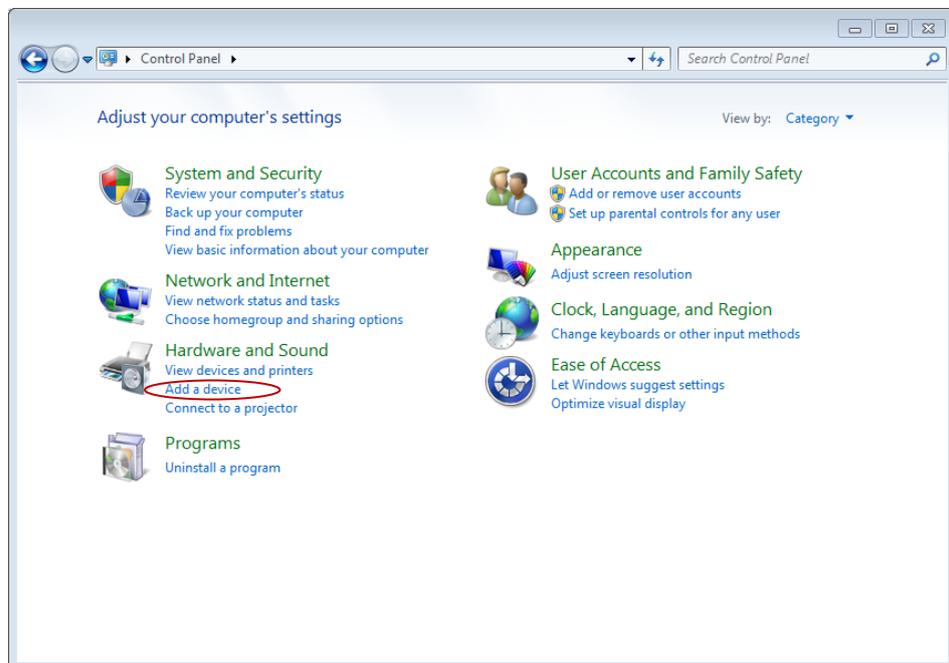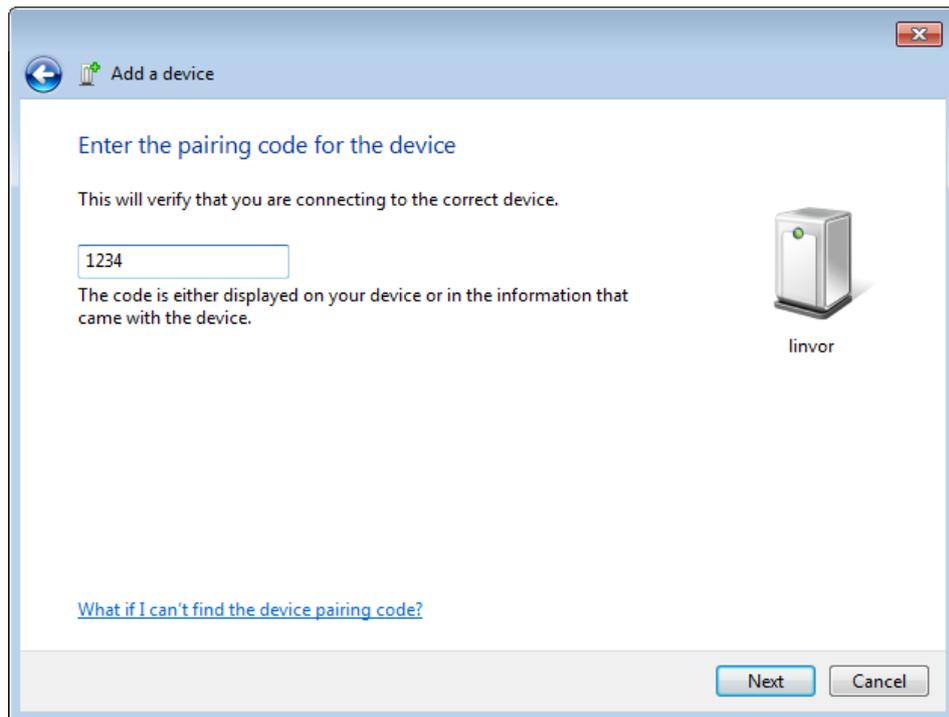

The computer will find the transmitter which usually is identified as 'linvor' as shown below.
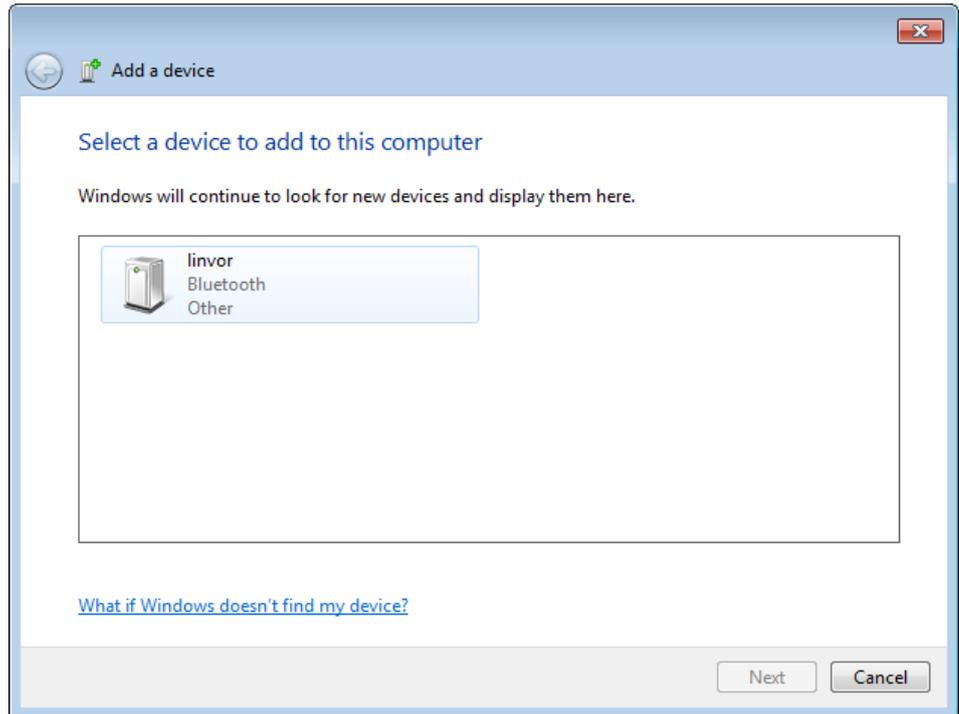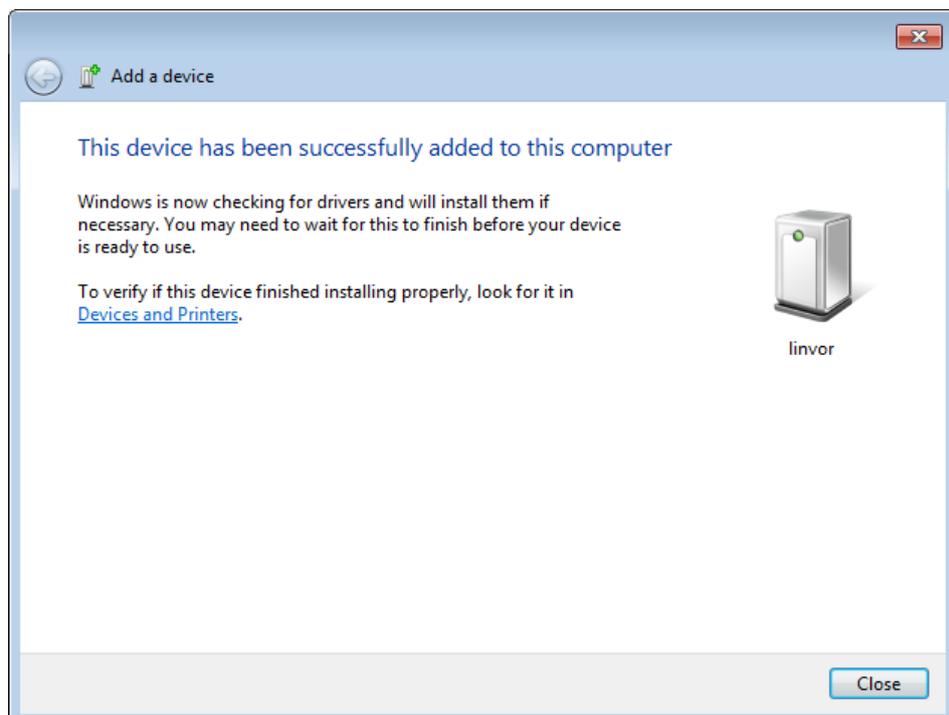
Select the device by left-clicking it and select the option: 'Enter a device's pairing code'.
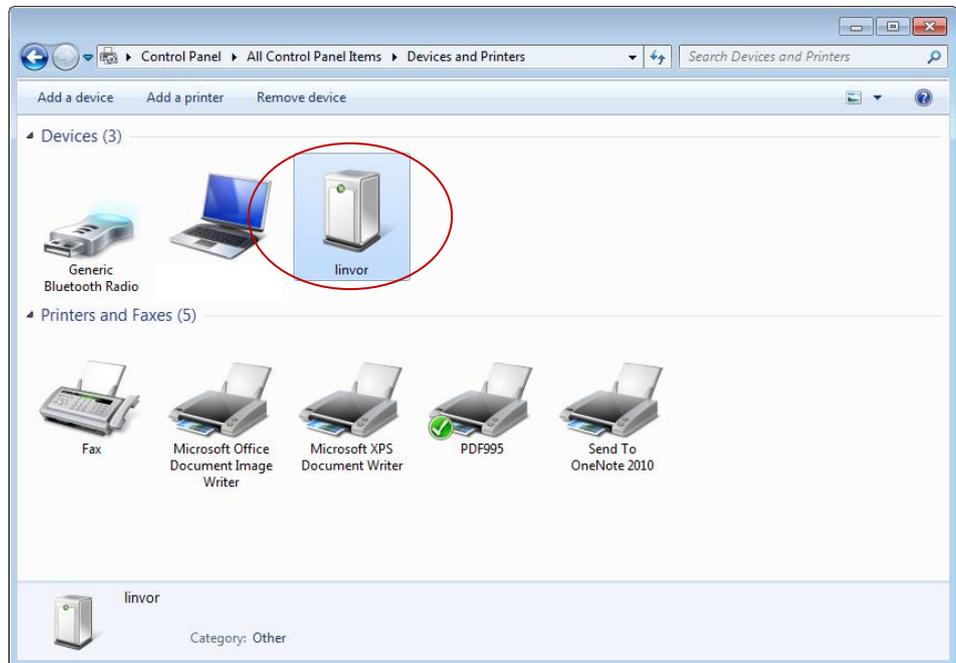

The following menue will appear and allow you to enter the pairing code which is usually '1234'.

After successful completion of the pairing you would get the following screen.

If you go back to the Control Panel you can check out the properties of the Bluetooth connection by selecting the 'View devices and printers' option and right-clicking the 'linvor' picto.


By selecting 'Hardware' you could figure out the virtual COM port used for the Bluetooth connection; in this instance COM11 is used. This COM port information is required to perform the next step of connecting your PiKoder/SSC to your computer.
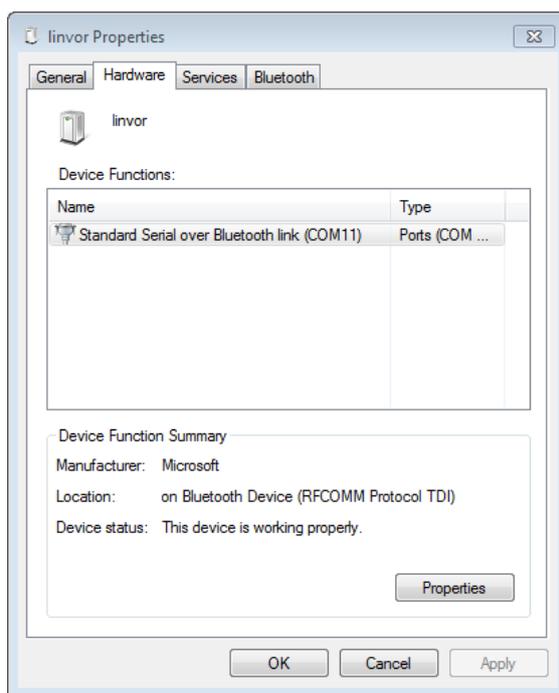
Connecting the PiKoder/SSC RX to the PCC PiKoder Control Center

Please download the latest executable version of the PCC application from www.pikoder.com and unzip the downloaded file – there is no installation required.

.

Make sure that your computer is turned on and that your Bluetooth dongle is placed in a USB port. Power up the PiKoder/SSC RX with the transmitter / receiver connected (the red led will blink fast). Start the PCC Control Center application.
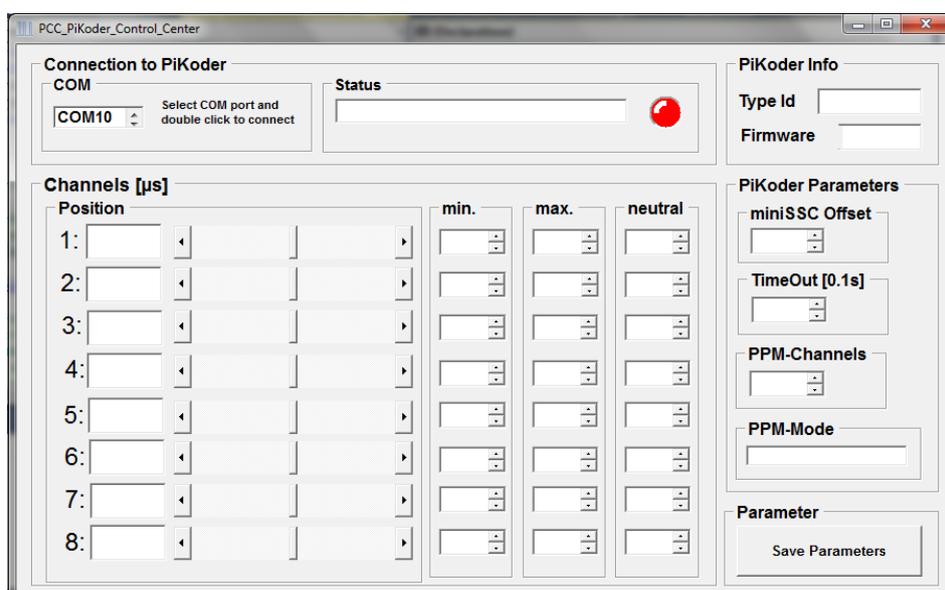
The PiKoder Control Center application would show COM10 to be the first available port in your computer. Select the COM port your PiKoder/SSC is connected to (in this example COM31, see below) and then click on the box to indicate your selection and communication to the controller will be established.
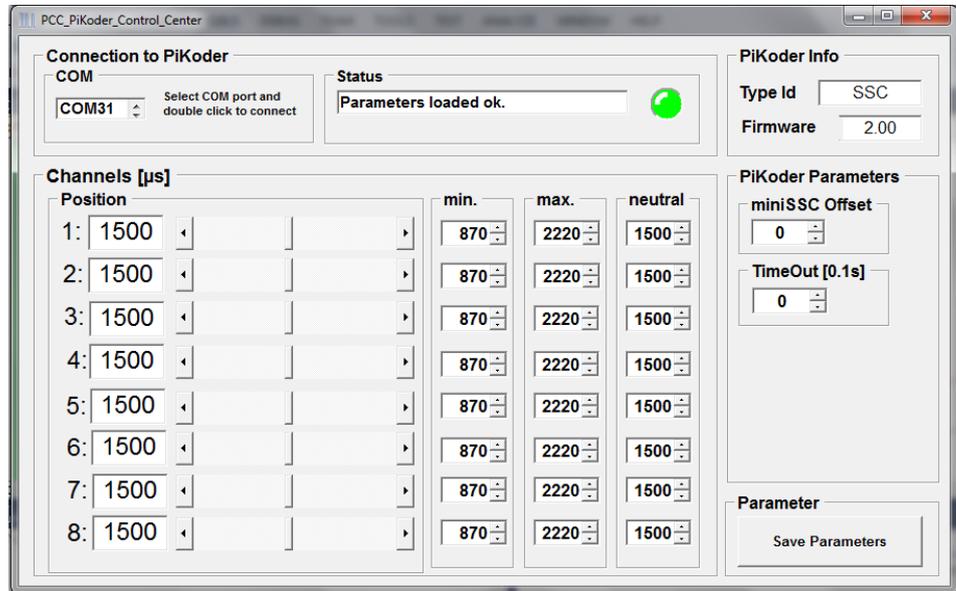
You would now have full control of your SSC: either for real-time control by the sliders or for changing the settings and save the new parameters. Please refer to the PiKoder/SSC User's Manual for a full description of all parameters.

Room for your notes

4

Connecting the PiKoder/SSC to an Android tablet

An Android tablet or smart phone can be used also for remotely controlling your PiKoder/SSC based on the PiKoder/SSC's terminal interface in combination with an Android app such as BlueTerm.

You would have to install BlueTerm on your tablet. Please make sure to set the parameters as shown on the screenshot below. Esspecially the CR+LF characters can be removed because they are not needed by the PiKoder/SSC.
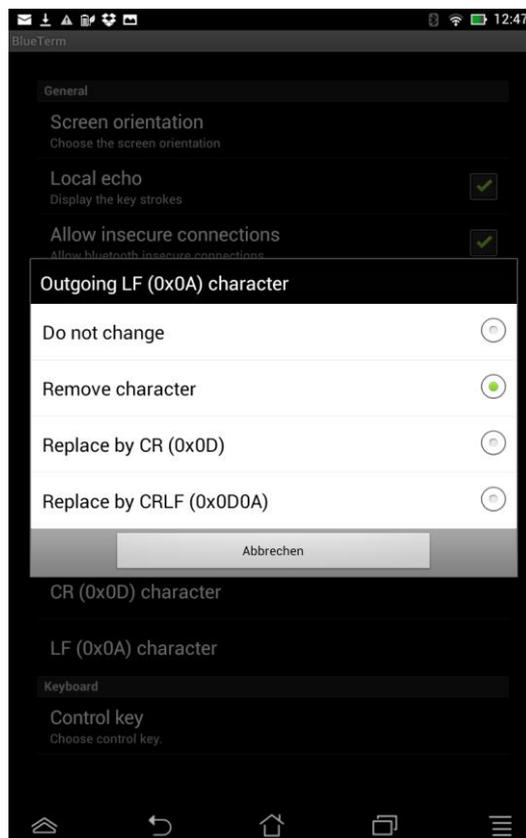
After connecting to linvor you can apply all standard commands as shown below.

5

Connecting the PiKoder/SSC to an Arduino Uno

This section describes how you connect an Arduino in combination with a Bluttooth Shield to your PiKoder/SSC.

In order to maintain the transparent serial communication you would have to program the Bluetooth Shield in a way that communication to the receiver is established automatically whenever the receiving unit is found. The following description is based on the ITead Bluetooth Shield. Please bear in mind that other shields may behave differently. Therefore, if you consider using another shield you would have to make sure that the automatic connect is working properly by following the user manual of this particular shield.

For programming the shield you will need your computer and a standard terminal application such as Tera Term. To begin programming you would have to stack the shield on your Arduino. Since you will be communicating directly with the shield controller using the Arduinos USB port it is recommended to make sure that your Arduino does not interfere with the communication by loading the "do nothing" sketch:

```
/* Upload this sketch into your arduino and press reset*/

void setup(){
  // do nothing...
}

void loop() {
  // do nothing...
```

```
        }
```

It is also highly recommended that you do not stack other shields onto your Arduino while programming the BT Shield since this may adversely affect pins required to be free floating.

Please set the jumpers following the manual for direct communication:

When using the connection as Figure 4, the HC-05 connects with the FT232RL chip, and the FT232RL connect to PC by USB. Whit this configuration you can use the serial software on PC to control or configure the HC-05 module.
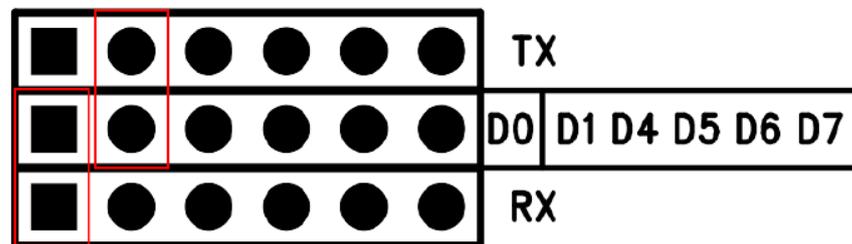

Figure 4 Connect the UART Interface as FT232

Please make sure also, that the mode switch is set to "CMD".

Start the terminal application and find your Arduino. The Bluetooth Shield uses a standard baud rate of 38400; therefore you would have to set the serial port parameter accordingly. If you use Tera Term you also want to enable local echo. Now the programming would begin. Follow the commands shown in the screen print. Please make sure that your receiver is turned on and can be found by the shield.
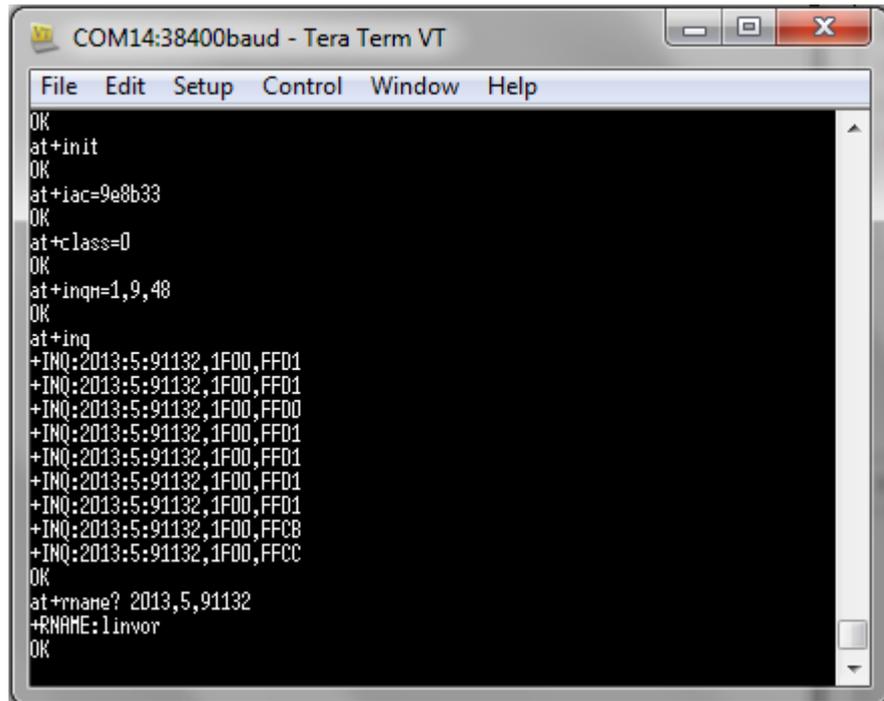
Following the "at+inq" command the Bluetooth Shield would scan for other devices and you may receive more that on device ids. (e.g. your cell phone or your computer may respond also). Therefore, you want to make sure that you bind your shield to the right device. Using the "at+rname?" command allows you to retrieve the device name based on the address. Please identify the address belonging to your "linvor" receiver modul.

Based on the linvor module address you can now bind the two modules using the "at+bind" command:
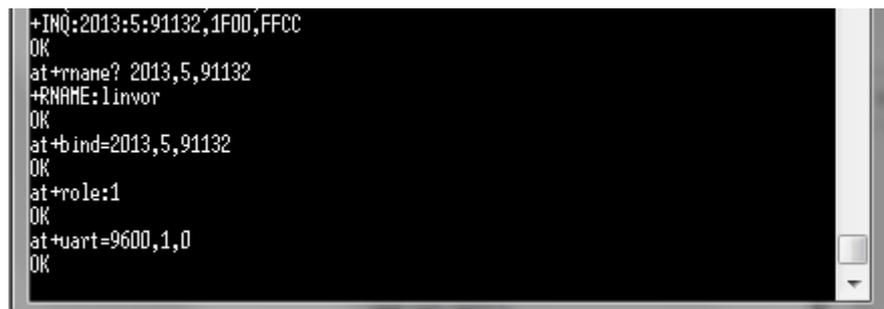
The linvor module can only act as a Bluetooth slave – therefore you have to program the shield to act as a master ("at+role" command). Set the baud rate to the PiKoder/SSC standard of 9600,1,N and finalize programming with making sure that the shield would only connect to the specified address:

This completes the programming. Set the mode switch on your shield to "DAT" and reset the shield. Please watch the red led on your receiver module. After blinking fast for a few seconds the led should go to an "on"-state indicating that the connection has been established. The status led on the shield will also change blinking rhythm to show that a connection has been established.

When using the PiKoder/SSC RX in Combination with the Arduino Digital RC sketch *arduinodtx* then please make sure to adjust the jumper setting on the BT shield as shown below.

A

PiKoder/SSC RX Pinning
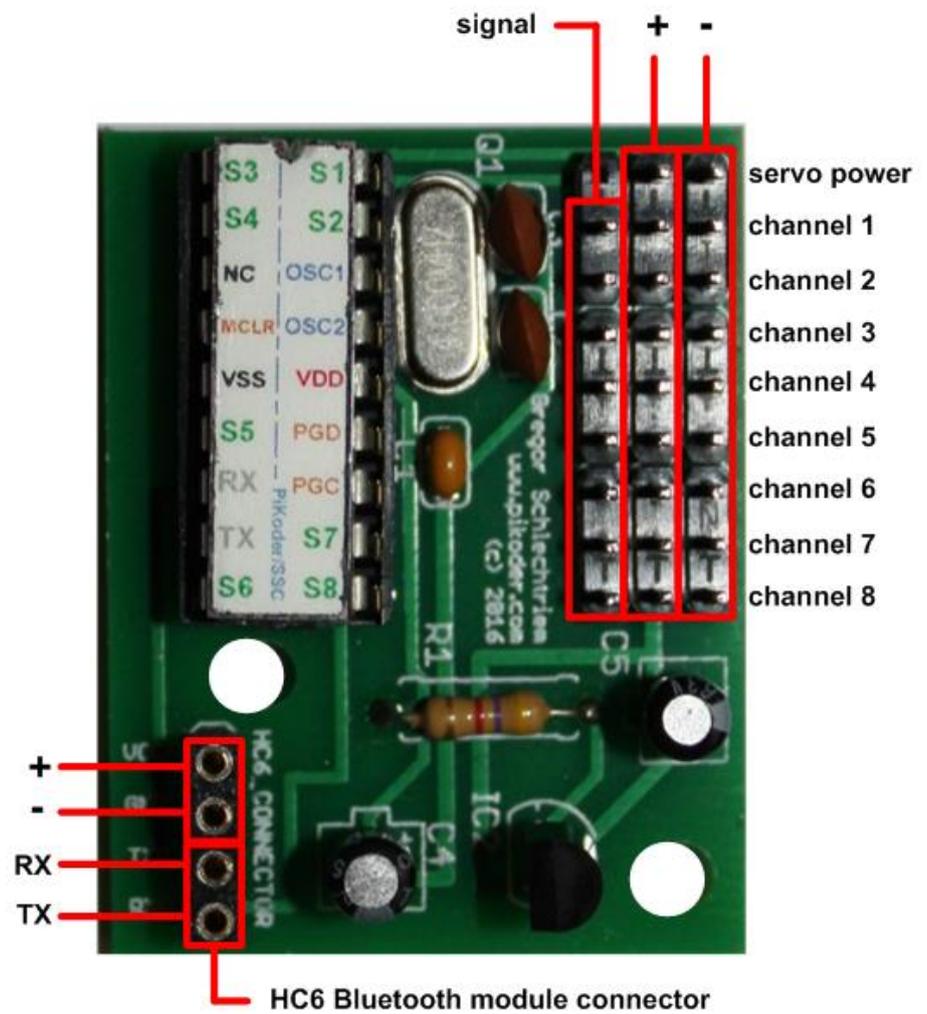
Room for your notes

B

PiKoder/SSC RX Schematic