

PiKoder/SSC wRX

Construction Manual

Version 1.0i
Released 05/2023

Gregor Schlechtriem
webmaster@pikoder.de

www.pikoder.de

Table of Contents

Helpful Hints	3
Contents of the Kit and Tools needed	4
Construction	6
Step 1: Equipping of the resistor R1	6
Step 2: Equipping of the Capacitors C1 – C3, and C5	7
Step 3: Equipment of the crystal.....	8
Step 4: Equipment of the IC socket.....	8
Step 5: Equipment of the voltage regulator IC2	9
Step 6: Equipment of the headers	10
Step 7: Equipping of the Capacitor C4.....	10
Commissioning	13
Programming of the WiFi radio ESP8266-01	13
Programming of the PiKoder/SSC	18
Testing of the servo channels with real servos.....	19
Receiver pinning	21
Receiver schematic	22
Wi-Fi protocol	23

Helpful Hints

Before you start the construction of your PiKoder/SSC wRX, it is recommended that you review these instructions completely. Then you would know what matters and would avoid mistakes which will be hard to fix afterwards.

A basic understanding of electronic parts, their handling, and soldering is required for the construction and the commissioning of the PiKoder/SSC wRX.

Conduct the soldering and the wiring in an orderly and conscientious manner; don't use acidic solder of any kind. Make sure that there are no cold solder joints. Keep these things in mind, because an unclean or bad joint, a defective contact or a bad construction cause a time-consuming search for faults and could possibly cause a destruction of the components.

The possibility that something won't work after the assembly, can be drastically diminished by working conscientiously and orderly. Check every step before continuing. Follow the instructions! Only do the things written in the manual and do not skip any steps! Check every step twice: once for building and once to check.

Please take the time it needs to build this kit. Tinkering is not task work and should be enjoyable!

Please share with me any errors, unclear instructions or ideas for improvements. You can reach me at gregor@pikoder.com. I am looking forward to your feedback.

Contents of the Kit and Tools needed

The kit in front of you contains all the necessary components needed to build a PiKoder/SSC wRX including the pre-programmed PIC16F628A and a pre-configured Wi-Fi radio ESP8266-01. Please verify the completeness with the help of the attached checklist.

Parts list: PiKoder/SSC wRX

Resistors

4k7, 0,25W (yellow/purple/red/gold)	R1	<div style="text-align: center;">√ <input type="checkbox"/></div>
-------------------------------------	----	---

Capacitors

22p	C1,C2	<input type="checkbox"/>
100n	C3,C5	<input type="checkbox"/>
2.2 µF, 16 V	C4	<input type="checkbox"/>

Semiconductors

PIC16F628A	IC1	<input type="checkbox"/>
LF 33CV	IC2	<input type="checkbox"/>

Misc.

Crystal 4 MHz	Q1	<input type="checkbox"/>
IC socket 18 pins		<input type="checkbox"/>
Header female 4 pins (2x)		<input type="checkbox"/>
Header male 2 pins	P1	<input type="checkbox"/>
Header male 1x8 pins		<input type="checkbox"/>
Header male 2x9 pins		<input type="checkbox"/>
ESP8266-01 Wi-Fi radio		<input type="checkbox"/>
PCB PiKoder/SSC wRX		<input type="checkbox"/>

You also need the following tools:

1. Electronic Soldering Iron
2. Electronic Solder
3. Side cutters to shorten the component connections

Additionally, to commission your receiver you will need:

1. Serial to USB converter and four jumper cables (female – female)
2. PC with downloaded terminal application and the PiKoder Control Center (PCC)
3. [optional] Power supply 4.8 – 6 Volts (e.g., 4 AA batteries)
4. [optional] Standard hobby servo (analog)

Construction

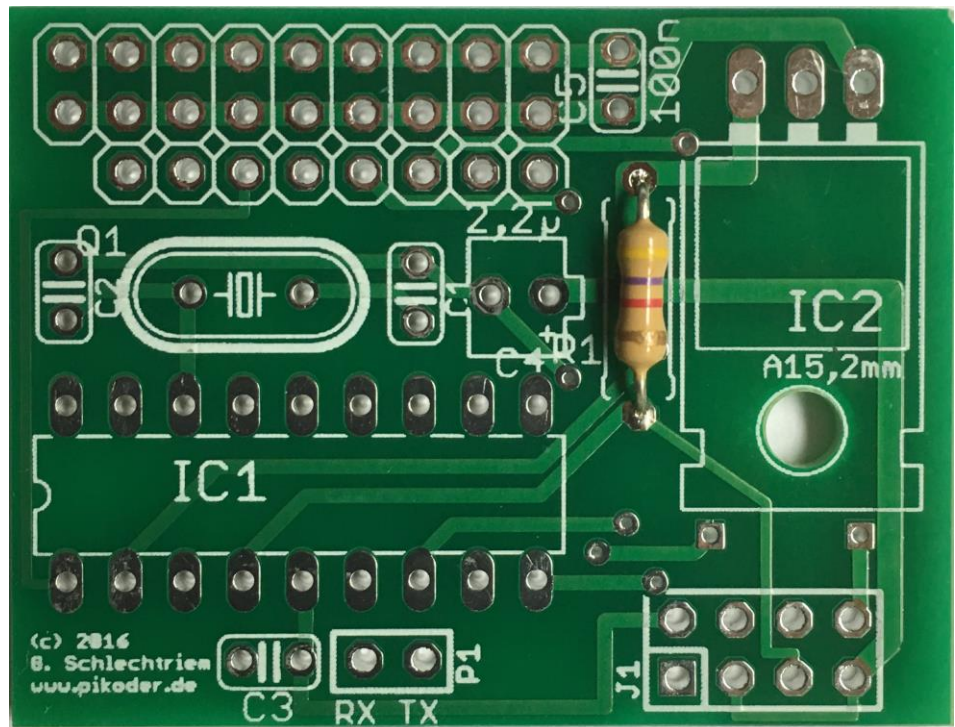
The following paragraphs will describe how to populate the components on the circuit board. The silk print on the circuit board will support this process. All parts are populated on the top side.

The order of the placement of the components depends on their height; generally the lower pieces will be placed first.

Step 1: Equipping of the resistor R1

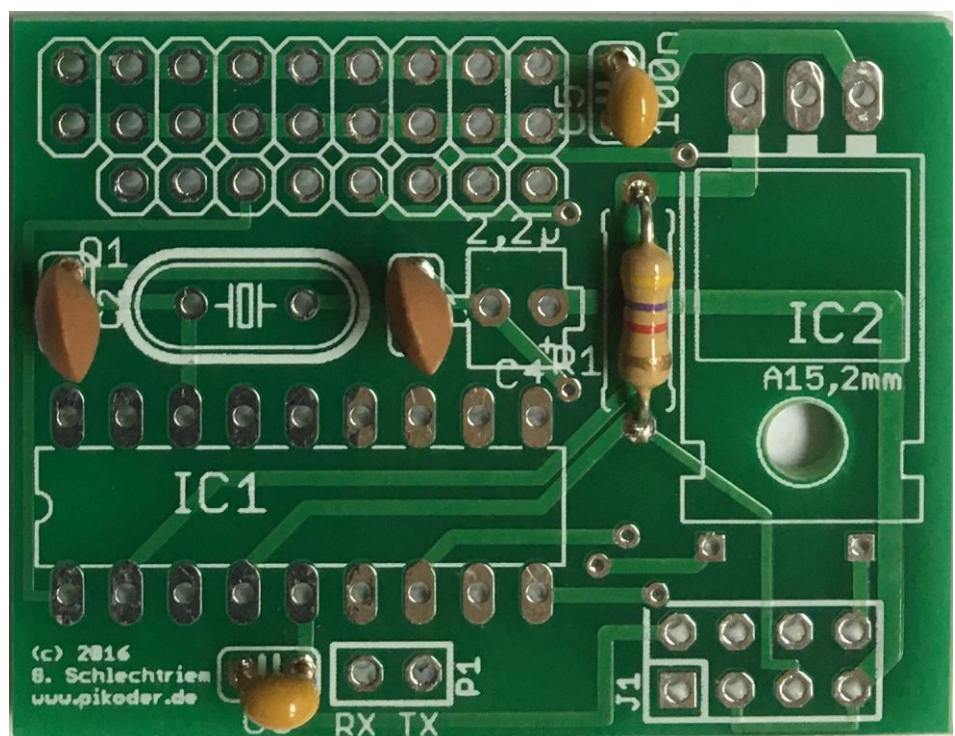
The resistor R1 will be equipped first. In order to do this, bend the connecting wires at a 90 ° angle based on the grid dimensions and then place the component in the designated holes (refer to the image below)

To ensure that the part does not fall of the circuit board when turning it around for soldering, bend the connecting wires at a roughly 45 ° angle apart and the solder them carefully with the conducting paths on the back of the circuit board. Afterwards cut off the excess wire.



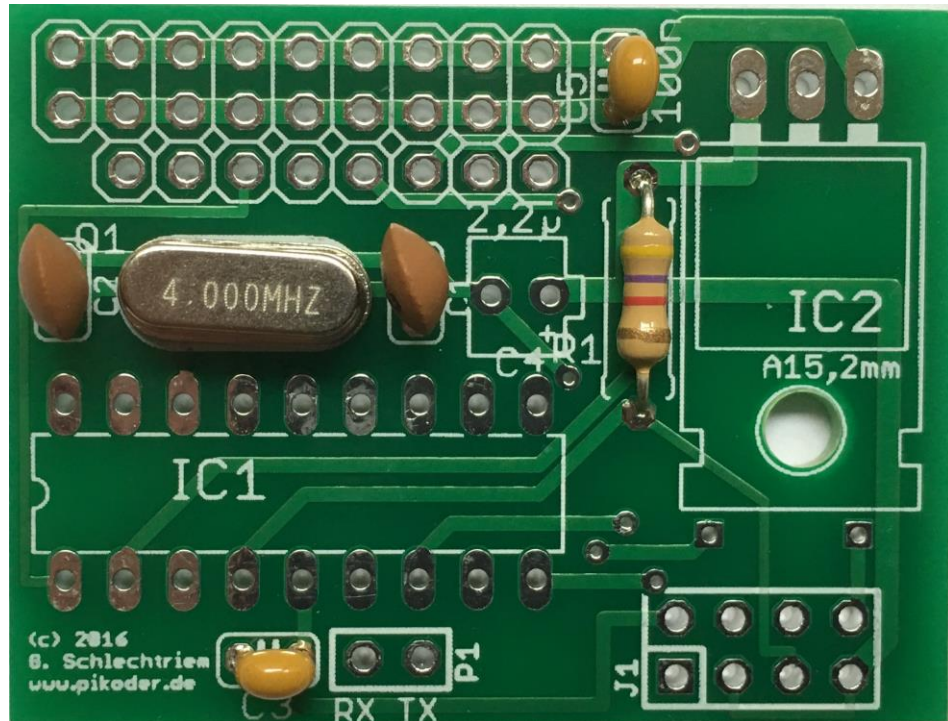
Step 2: Equipping of the Capacitors C1 – C3, and C5

Now, the Capacitors C1 – C3 and C5 will be inserted followed by a mild angling of the connection wires in order to prevent the pieces falling out. All capacitors are non-polarized; therefore the alignment is irrelevant. Please note that C1 and C2 have a capacity of 22 p (marked 22) and C3 and C5 have a higher capacity of 100 n



Step 3: Equipment of the crystal

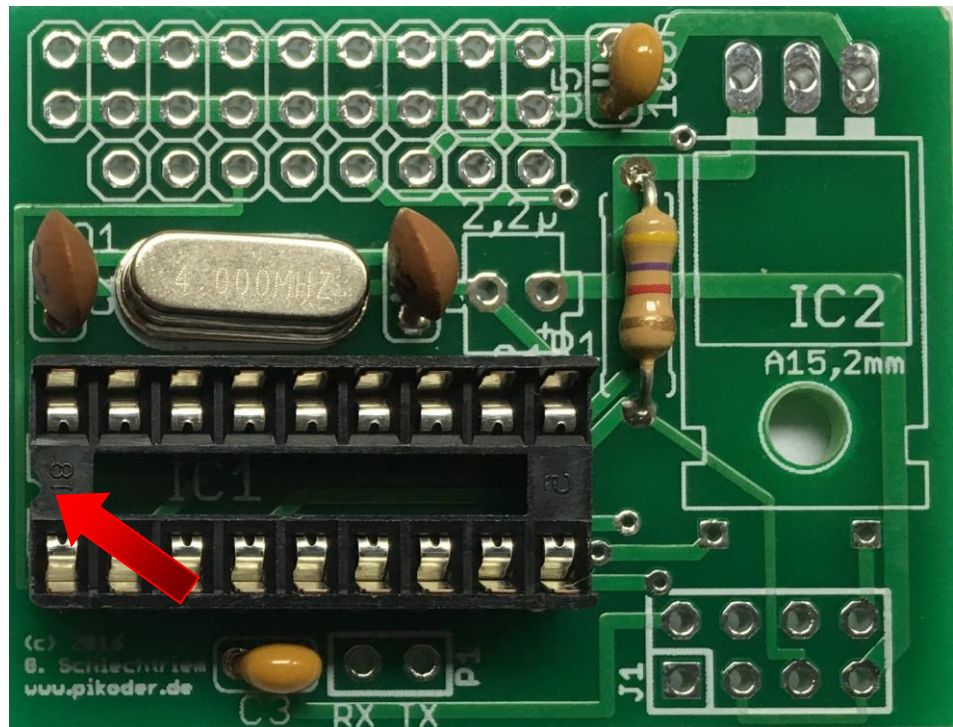
In this step, you would place the crystal, which is also a non-polarized is means that the alignment again is irrelevant.



Step 4: Equipment of the IC socket

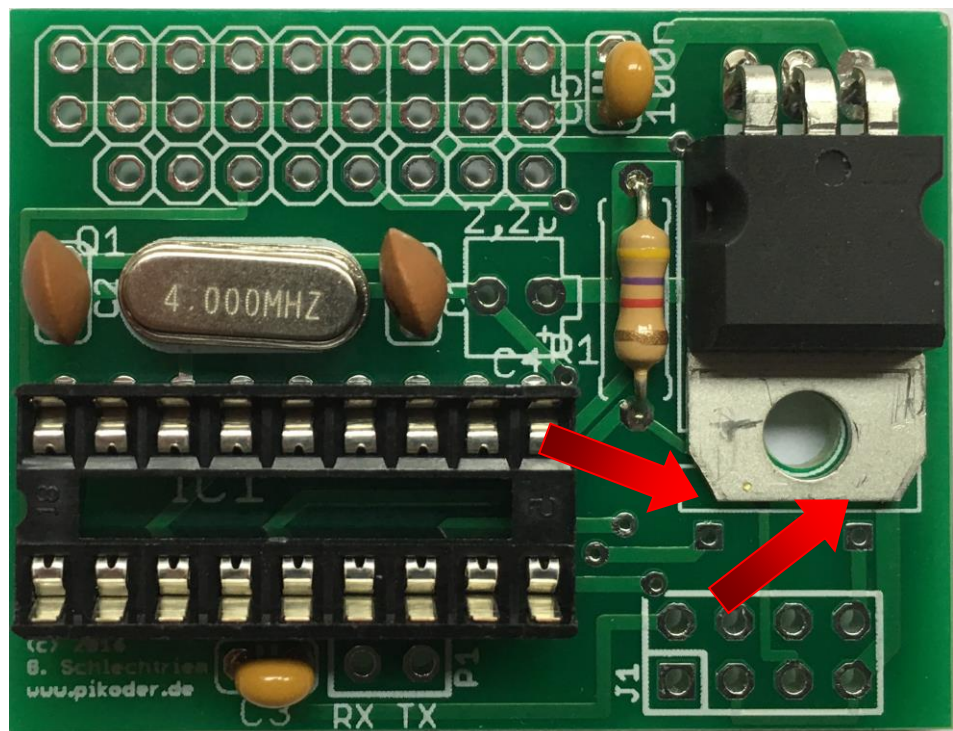
Place the socket in the position indicated on the top of the circuit board. Once again, verify the positioning: the notch (marked with a red arrow in the following image) needs to point to the left.

To avoid the socket from falling out of its position when turning the board slightly bend the two outer diagonal pins and then go ahead and solder all the other pins.



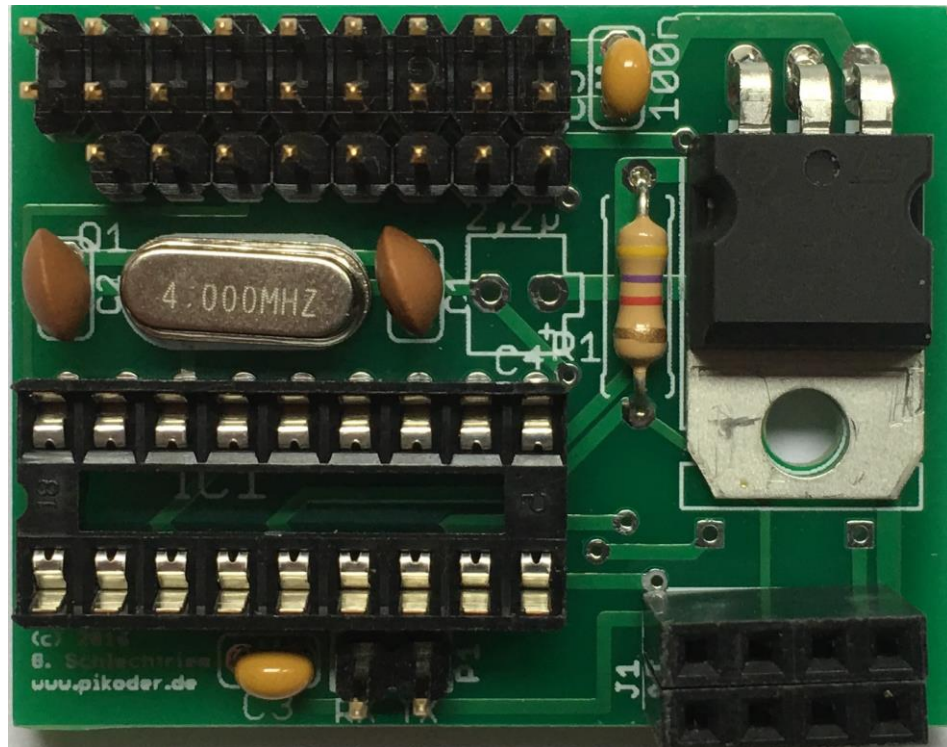
Step 5: Equipment of the voltage regulator IC2

Now you would place the voltage regulator. The pins are already square bent to support a proper placement. Prior to soldering, please make sure that the whole in the PCB and the component match to avoid shorts. Note: the potentially critical points on the PCB are marked with red arrows in the following image.



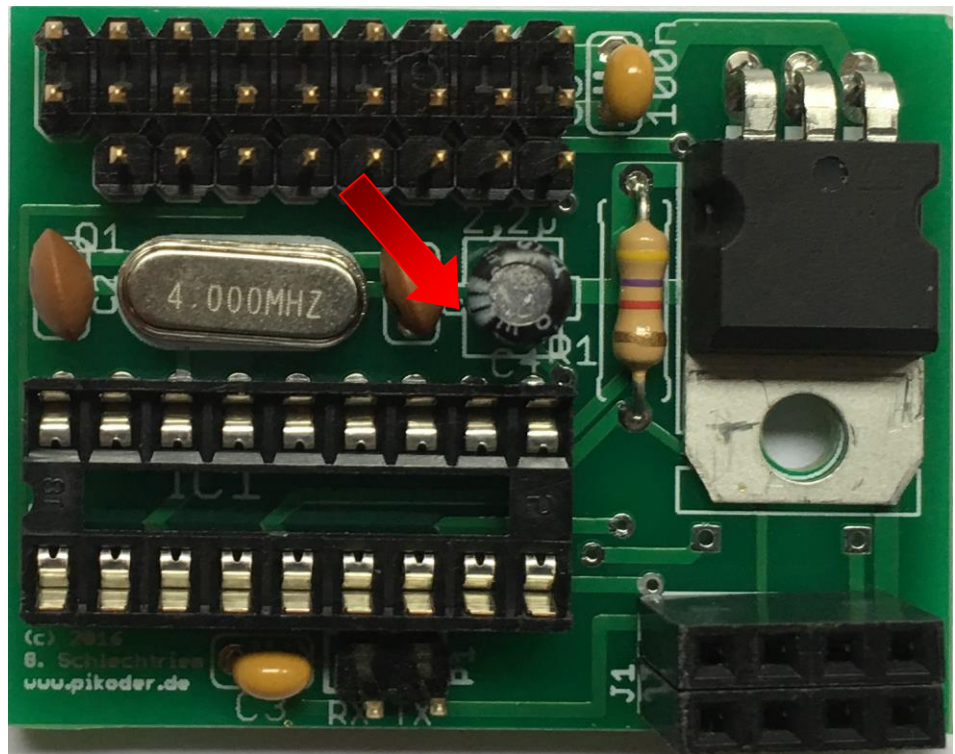
Step 6: Equipment of the headers

Now the two female headers (4 pins each forming an eight pin female header), the two pin header for the com interface and the pin headers for connecting to the servos and the power supply would be placed. The connection to the servos is comprised of a 2x9 pin and a 1x8 pin header. The 8 pin header is located at the inner position (see the image below). Please refer to the silk print for the correct placement.



Step 7: Equipping of the Capacitor C4

Place the remaining capacitor C4. This capacitor is polarized which means that you have to honor the orientation. The red arrow in the image below points to the “-” pin marked on the component.



This step completes the construction. The following section describes the commissioning of your PiKoder/SSC wRX.

- This page has been left blank intentionally -

Commissioning

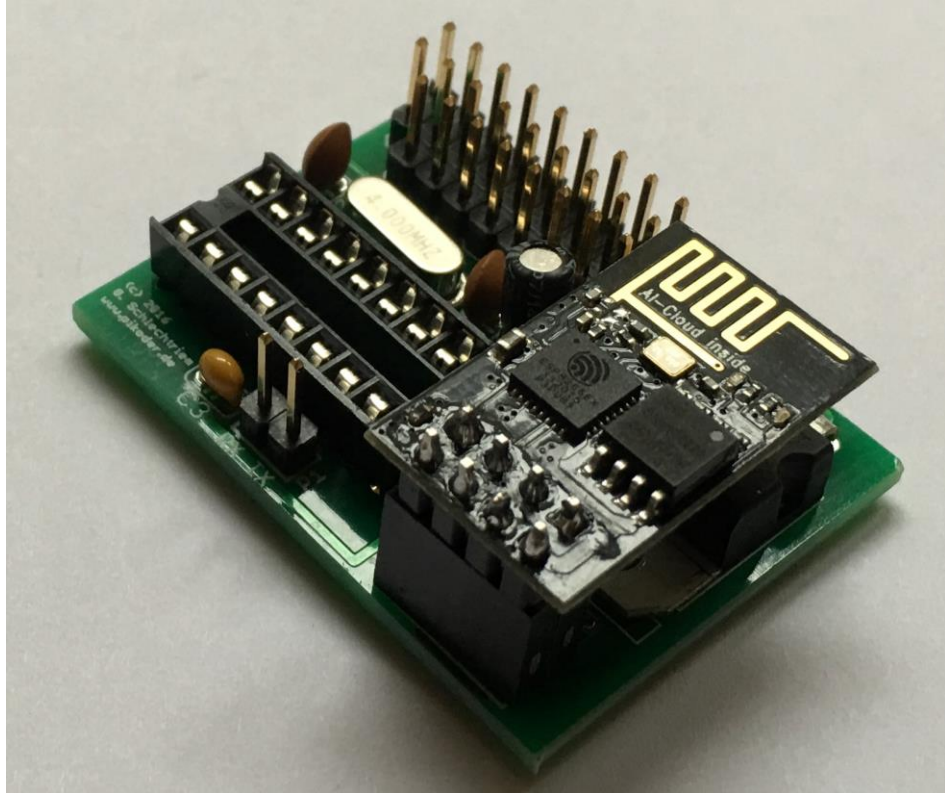
The commissioning of your PiKoder/SSC wRX starts off with programming the Wi-Fi radio to match your desired network settings. You would determine the network name (SSID) of your receiver and also determine your password. To simplify this process your Wi-Fi radio comes pre-programmed with the [ESP8266-01 udpRC sketch](#), which is freely available as Open Source.

Then you would (optionally) program your PiKoder/SSC wRX-controller to reflect any specifics in your application. Finally, you would commission your new RC system using your Android Smart device as your transmitter.

Programming of the WiFi radio ESP8266-01

For the programming, please place the ESP8266 on its respective position on the PiKoder/SSC wRX PCB (please refer to the following picture for more information). The parameters which you would set are stored in the non-volatile storage of the ESP-Controllers and would be applied every time you turn your receiver on. Please note that the PiKoder/SSC controller cannot be inserted during the programming.

The ESP8266 would be programmed using a standard terminal application on your PC applying a very limited number of simple commands. To establish the connection between the radio and the PC you would need a USB-UART converter such as the one shown on the following page and four jumper cables. These adapters and cables are readily available on e.g., eBay or could be ordered at www.pikoder.de.



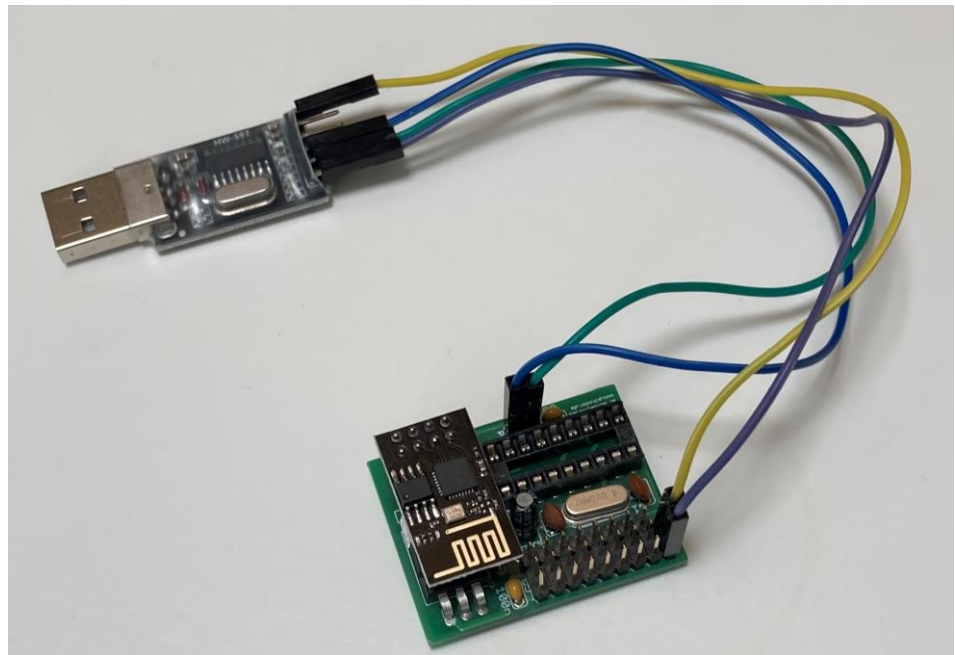
PiKoder/SSC wRX w/o controller with ESP8266-01 ready for programming



USB to UART adaptor for connecting your PC with the ***PiKoder/SSC wRX***

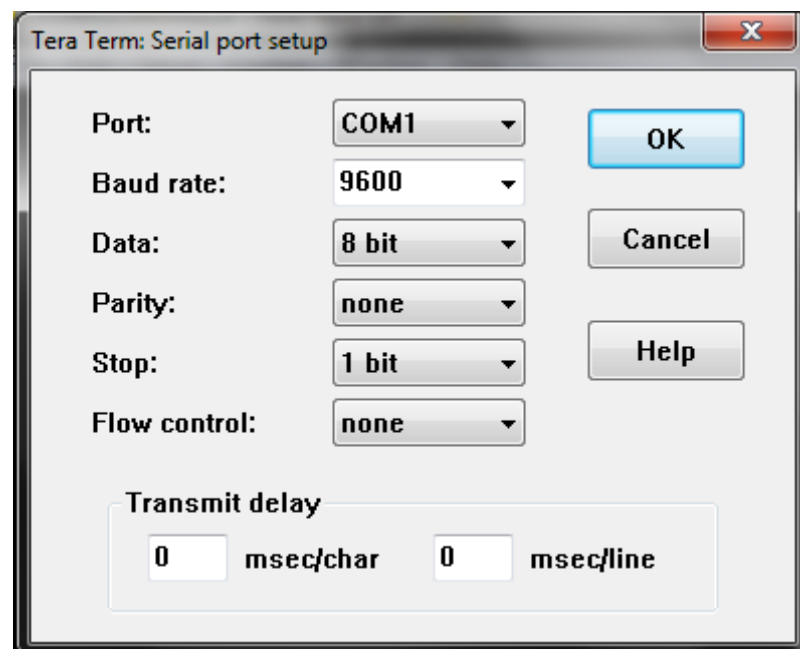
The following image shows the wiring of the PCB and the converter. Upon connecting to your PC's USB connector, a red and a blue LED will lit. The blue LED indicates communication activity and will turn off after a very short while; the red LED as a power indicator will stay on.

The driver of the USB-UART converter is included in Windows so that no additional installation is needed.



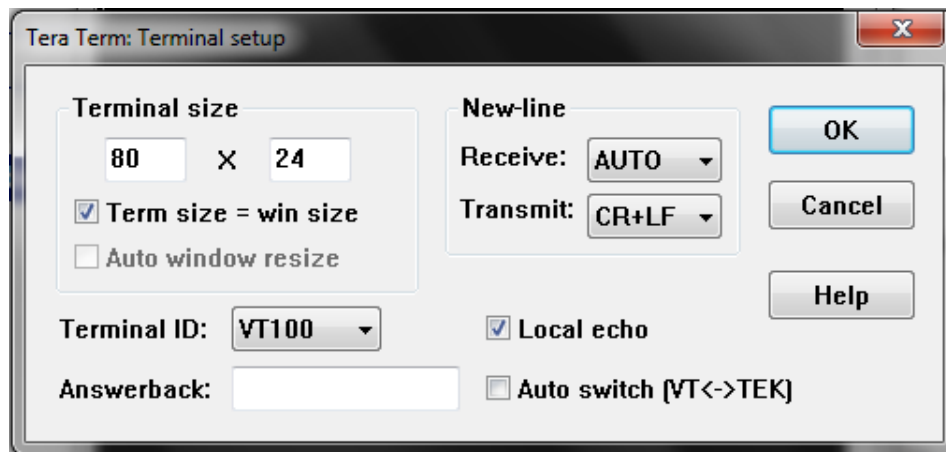
For programming the ESP8266 you would need a standard terminal program. The following steps are described employing Tera Term, a free downloadable, Open-Source terminal program available at <https://ttssh2.osdn.jp/index.html.en>.

You start programming by setting the parameters of your computer's serial port. Starting Terra Term please select „Setup“, then „Serial port...“ and get the following screen:



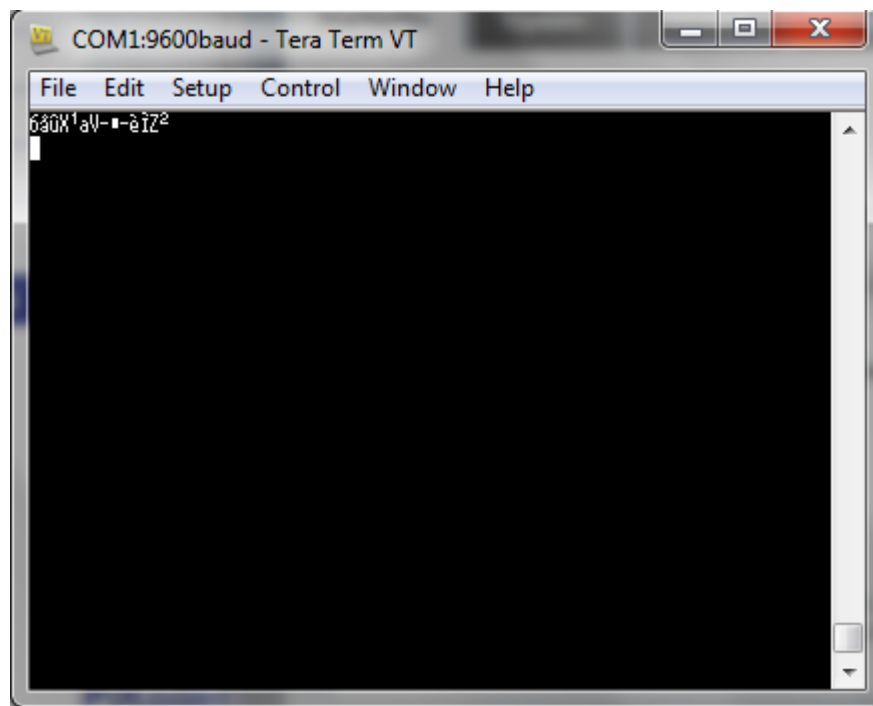
Select the COM port connected to the USB-UART converter and a Baud rate of 9600. Now hit OK.

The next step matches the terminal parameters to the ESP requirements. Press „Setup“ again and then „Terminal...“ bringing up the following screen:

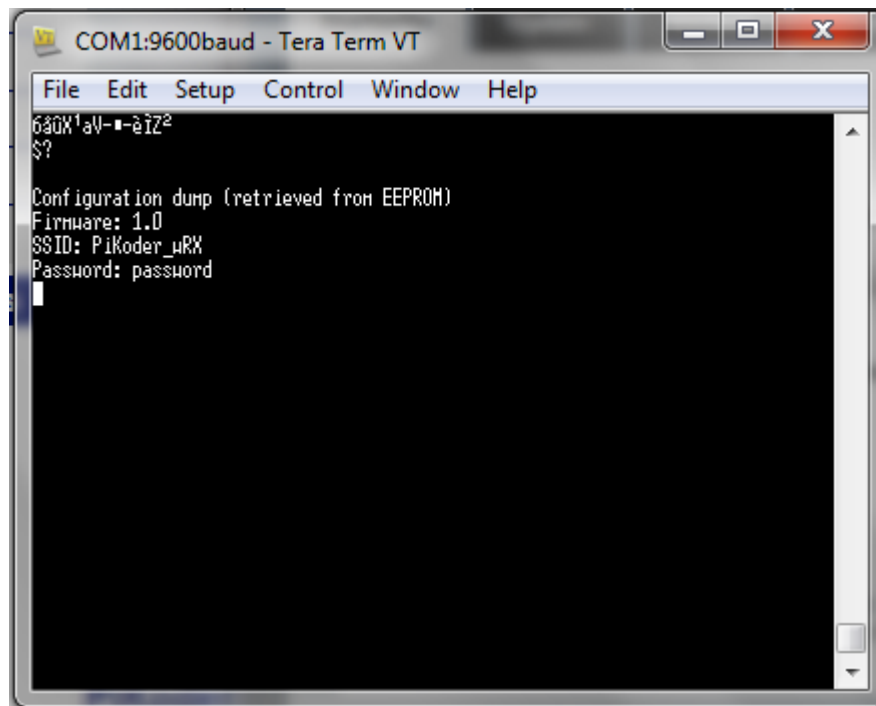


Make sure your settings match the above picture and hit OK again.

Now the terminal would be ready to communicate and program the ESP8266. Please reset the radio by briefly pulling and the reconnecting the yellow power cable. The radio would send a few unreadable characters indicating readiness (see below).



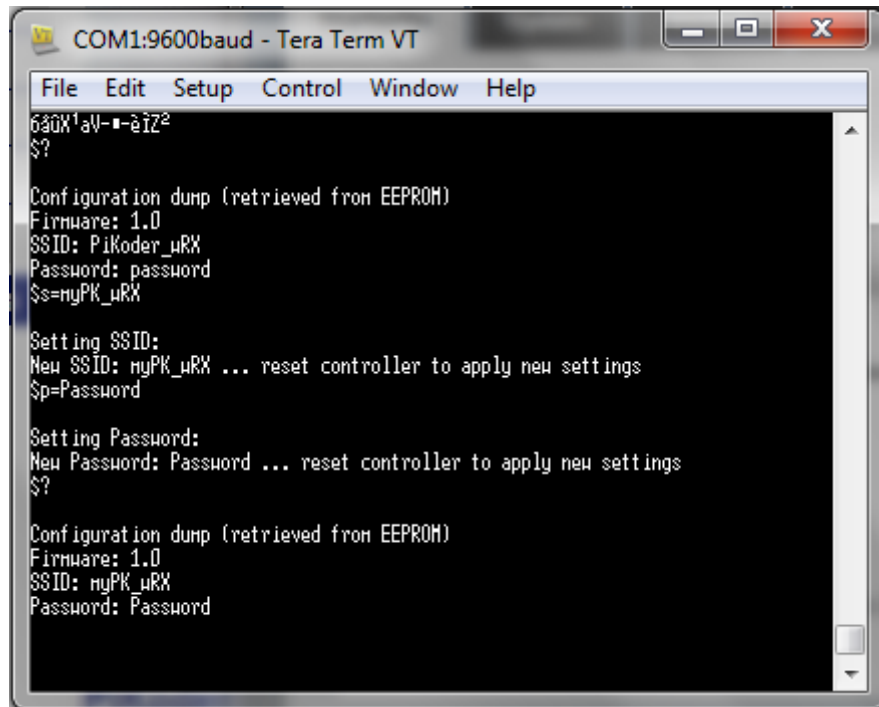
Start of by verifying the current settings of your radio. Therefore, please enter „\$?“ <ENTER>.



The parameter dump shows the SSID which is the name of the receiver's access point you would later connect your smart device to (in this particular instance the SSID would be „PiKoder_wRX“). Additionally, the password which you would need later also to connect to your receiver is echoed (in this example: „password“). If you accept these default values then you completed the programming of the ESP8266. Please note: if you would change your mind later you could still re-program the radio.

By entering the command „\$s=“ you would change the SSID. The following image shows the steps required. The new SSID you entered would be echoed and stored in the non-volatile memory of the radio. Please note that revised settings only take effect after restarting the ESP8266.

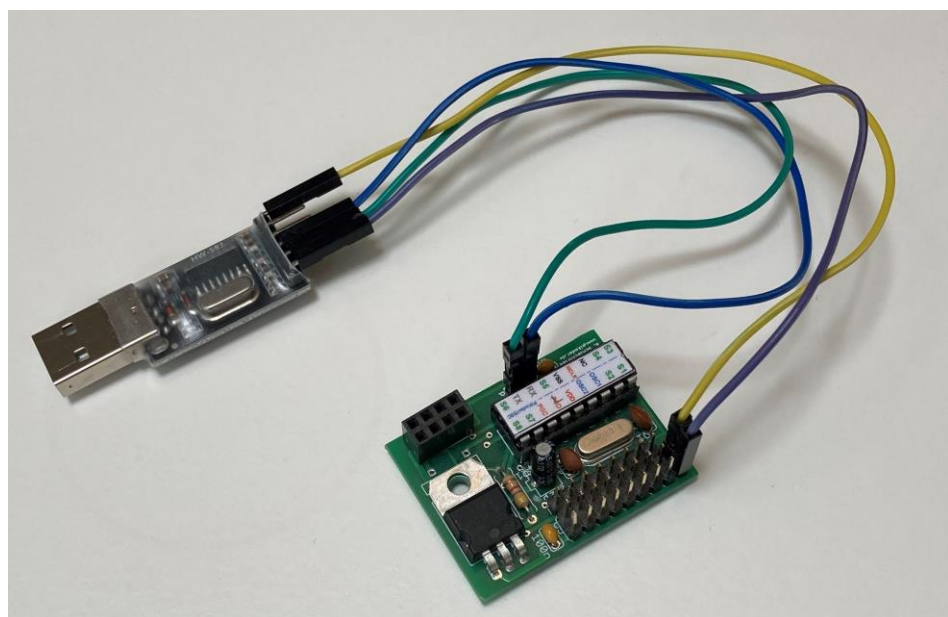
For changing your password you would use the command „\$p=“; the following picture also reflects the steps required. It might be a good idea to verify all settings prior to moving on to the next step by using „\$?“ once again.



Programming of the PiKoder/SSC

In the next step you would program the PiKoder/SSC to reflect any specifics in your application. Also, you could test all servo channels and set max, min and fail-safe values per channel.

To connect the PC and the controller you would have to remove the radio, insert the controller into its socket (please refer to the following image for the correct orientation) and switch the RX and TX line (green and blue cable) as shown in the following image.



The programming requires again a terminal application or the [PCC \(PiKoder Control Center\)](#) offering a graphical user interface. Please refer to the [PiKoder/SSC User's Guide](#) for a detailed description of the available settings.

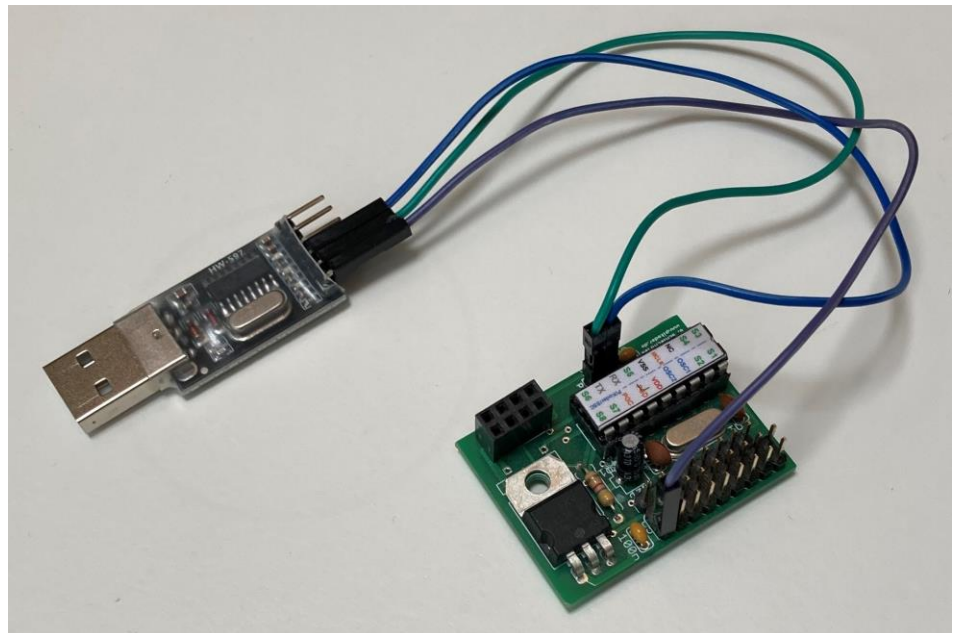
Testing of the servo channels with real servos

For the (optional) test of the servo channels with real servos you would need an additional power supply since the output of the USB adapter is most likely not sufficient to power servo movement. The voltage needed depends on the servo you select. The receiver supports a voltage range from 4.8 – 16 Volts. You would connect the additional power to the pins marked with „+“ and „-“. Please refer to exhibit A for more details.

Please note that you would also have to change the wiring of the USB adaptor as shown in the following picture – make sure that the yellow power wire is not connected. The ground connection to your PC is established by using one of the servo connector ground pins (in the following image the ground connector of channel 8 would be used).

After this, you can move a servo by moving the respective slider in the PCC.

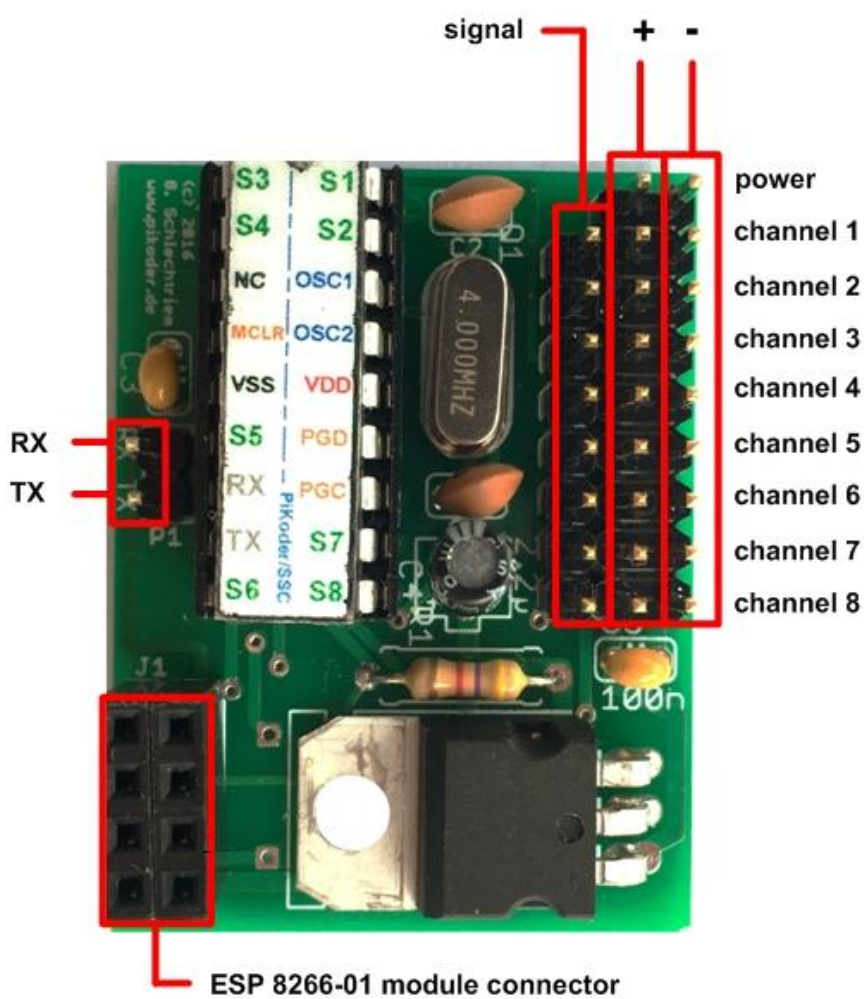
After completing the programming and the tests you would remove the USB-UART converter and plug in the radio again. This completes the commissioning of the PiKoder/SSC wRX. Your receiver is now ready for the commissioning of your new RC system.



- This page has been left blank intentionally -

A

Receiver pinning





Wi-Fi protocol

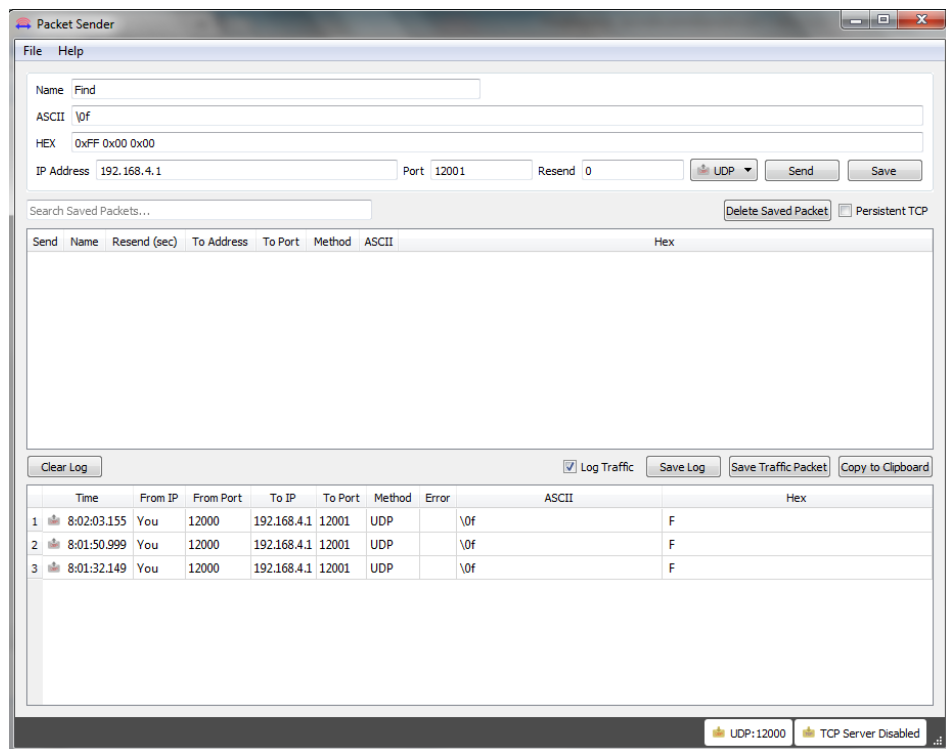
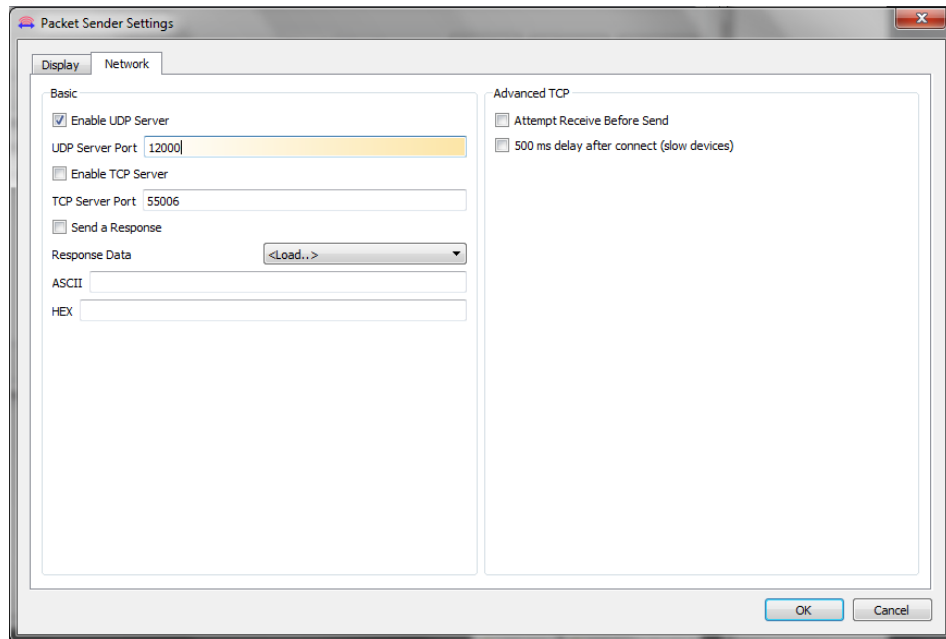
The communication between the PiKoder/SSC wRX and the Android smart device is based on the so called UDP (User Datagram Protocol – leading to the app’s name udpRC). This protocol allows the efficient exchange of small messages and thereby accommodating the typical requirements of radio controlled models.

This sections details the communication and offers additional tests of the Wi-Fi communication of the PiKoder/SSC wRX. All tests assume that the receiver is powered up (servos which are connected also are not critical) and that the programming followed the steps outlined in this manual.

The tests are based on the application *Packet Sender* by *Dan Nagle*, which you can download on <https://packetsender.com/>. After installing please connect your PC on system level to the access point offered by your receiver e.g. to „PiKoder_wRX“. Please make sure that your firewall recognizes this network as a home network to allow for data transmission. Your computer should have received the IP address 192.168.4.2.

Start *Packet Sender*, select Settings->Network, set the UDP-Server-Port to 12000 and click the checkbox „Enable UDP Server“ (see the following image) When returning to the main screen you would notice in the lower right corner UDP:12000 as shown on the second image on the next page. If not, you would have to restart the software.

In order to send a command to the receiver, you would have to enter the IP address of the receiver first (192.168.4.1) and then the receiver’s assigned communication port 12001.



After taking care of the appropriate addressing you would enter the actual command in the line above the address fields in hex format. The servo channels are controlled by the miniSSC protocol which has the following structure:

0xFF 0xII 0xPP

meaning:

0xFF: Start of a new command

0xII: II represents the channel number starting with 0

0xPP: PP indicates the position of the servo (range 0 – 0xFE)

If you wanted to move the servo on channel 1 to the minimal position, then you would enter: 0xFF 0x00 0x00 (as shown in the picture above). After hitting the Send-button, the servo will move as requested. Entering: 0xFF 0x00 0x7F brings the servo back to its neutral position.