

# *PiKoder/ PPM wRX*

## Construction Manual

---

Version 1.0  
Released 08/2018

Gregor Schlechtriem  
[webmaster@pikoder.de](mailto:webmaster@pikoder.de)

[www.pikoder.de](http://www.pikoder.de)

## Table of Contents

<b>Helpful Hints</b>	<b>3</b>
<b>Contents of the Kit and Tools needed</b>	<b>4</b>
<b>Construction</b>	<b>6</b>
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
<b>Commissioning</b>	<b>13</b>
Programming of the WiFi radio ESP8266-01 .....	13
Programming of the PiKoder/PPM.....	18
Testing of the PPM output with an FC .....	19
<b>Receiver pinning</b>	<b>21</b>
<b>Receiver schematic</b>	<b>22</b>
<b>Wi-Fi protocol</b>	<b>23</b>

---

# 1

---

## *Helpful Hints*

Before you start the construction of your PiKoder/PPM 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/PPM 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](mailto:gregor@pikoder.com). I am looking forward to your feedback.

---

# 2

---

## ***Contents of the Kit and Tools needed***

The kit in front of you contains all the necessary components needed to build a PiKoder/PPM 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/PPM wRX**

#### **Resistors**

4k7, 0,25W (yellow/purple/red/gold) R1

#### **Capacitors**

22p C1,C2

100n C3,C5

2.2  $\mu$ F, 16 V C4

#### **Semiconductors**

PIC16F628A IC1

LF 33CV IC2

#### **Misc.**

Crystal 4 MHz Q1

IC socket 18 pins

Header female 4 pins (2x)

Header male 2 pins (2x) P1

Header male 1x3 pins

ESP8266-01 Wi-Fi radio

PCB PiKoder/PPM wRX

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. Power supply 4.8 – 6 Volts (e.g. 4 AA batteries)
2. Serial to USB converter with 3.3V signal level for changing WLAN settings and application specific parameters (optional)
3. PC with PiKoder Control Center (PCC, downloaded from [www.pikoder.com](http://www.pikoder.com)) for use with USB cable
4. Flight Controller (FC) w/ PPM input such as e.g. an APM for testing (optional)

---

# 3

---

## ***Construction***

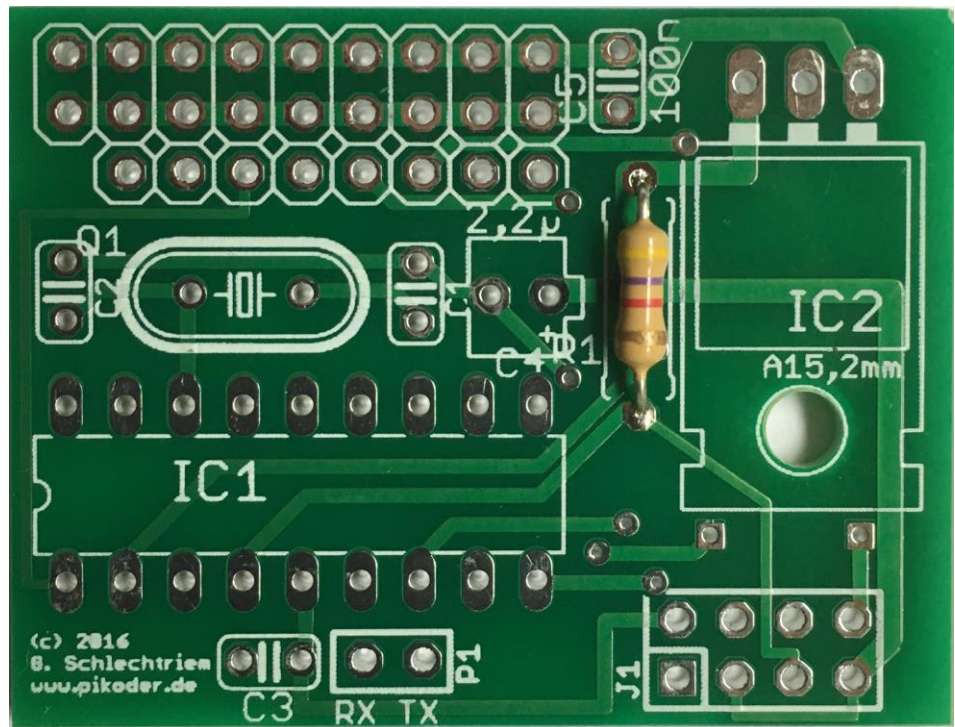
The following paragraphs will describe how to populate the components on the circuit board. The silk print on the circuit board will support this process. All parts are populated on the top side.

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**

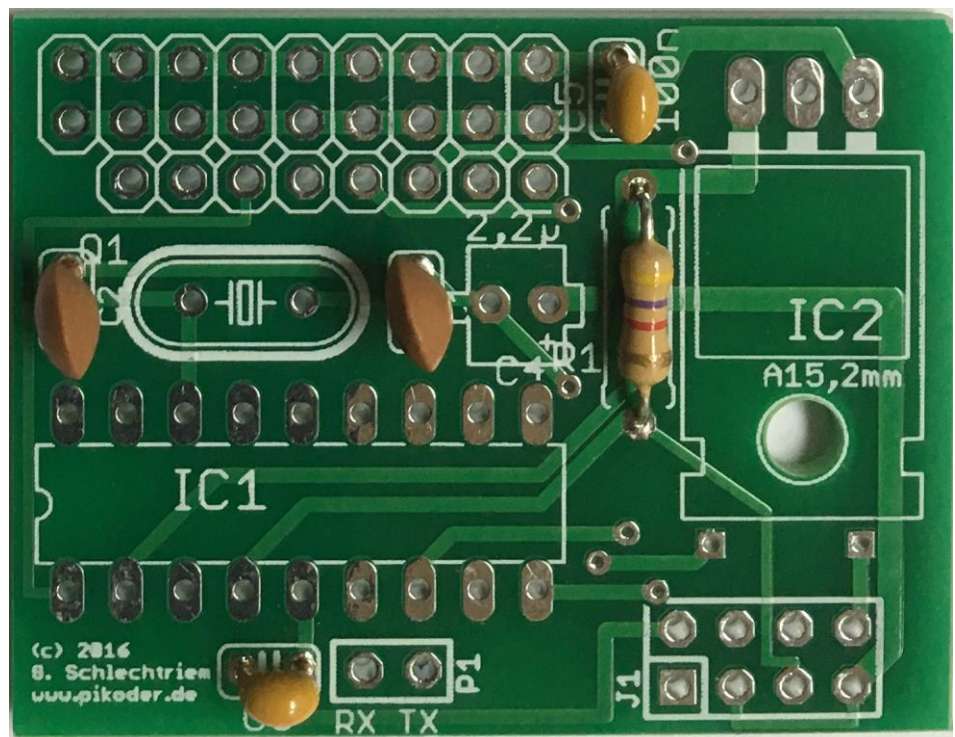
The resistor R1 will be equipped first. In order to do this, bend the connecting wires at a 90 ° angle based on the grid dimensions and then place the component in the designated holes (refer to the image below)

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

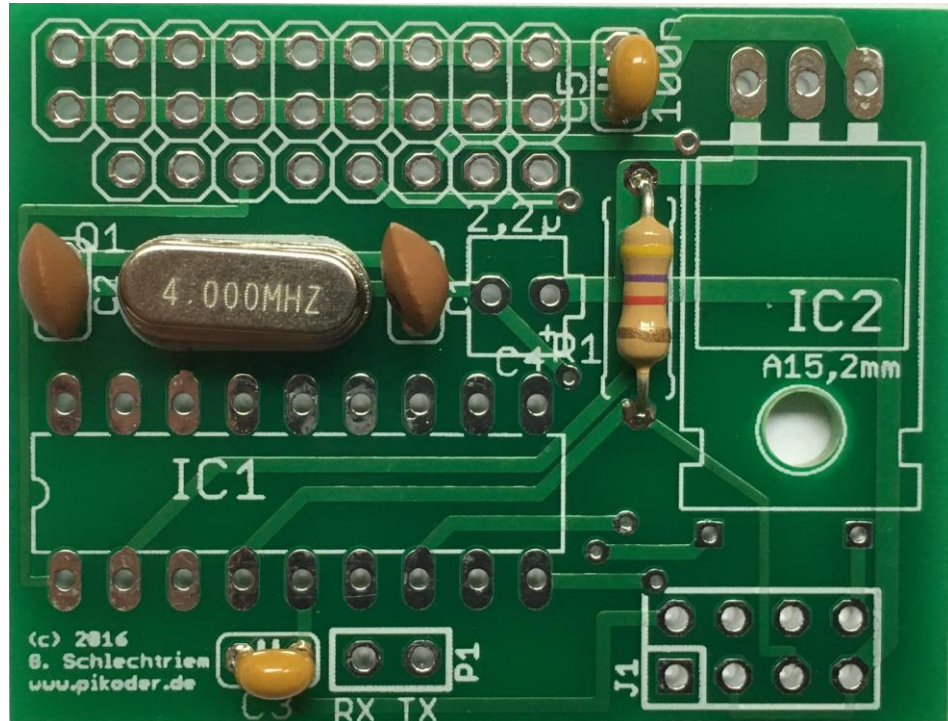
Now, the Capacitors C1 – C3 and C5 will be inserted followed by a mild angling of the connection wires in order to prevent the pieces falling out. All capacitors are non-polarized; therefore the alignment is irrelevant. Please note that C1 and C2 have a capacity of 22 p (marked 22) and C3 and C5 have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

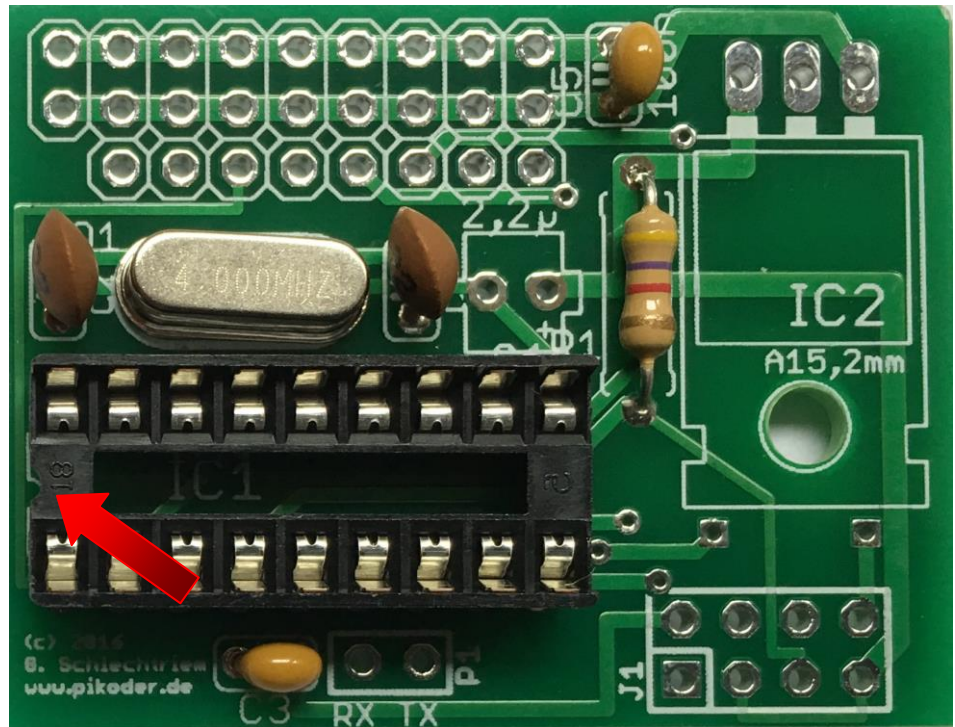


### Step 4: Equipment of the IC socket

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.

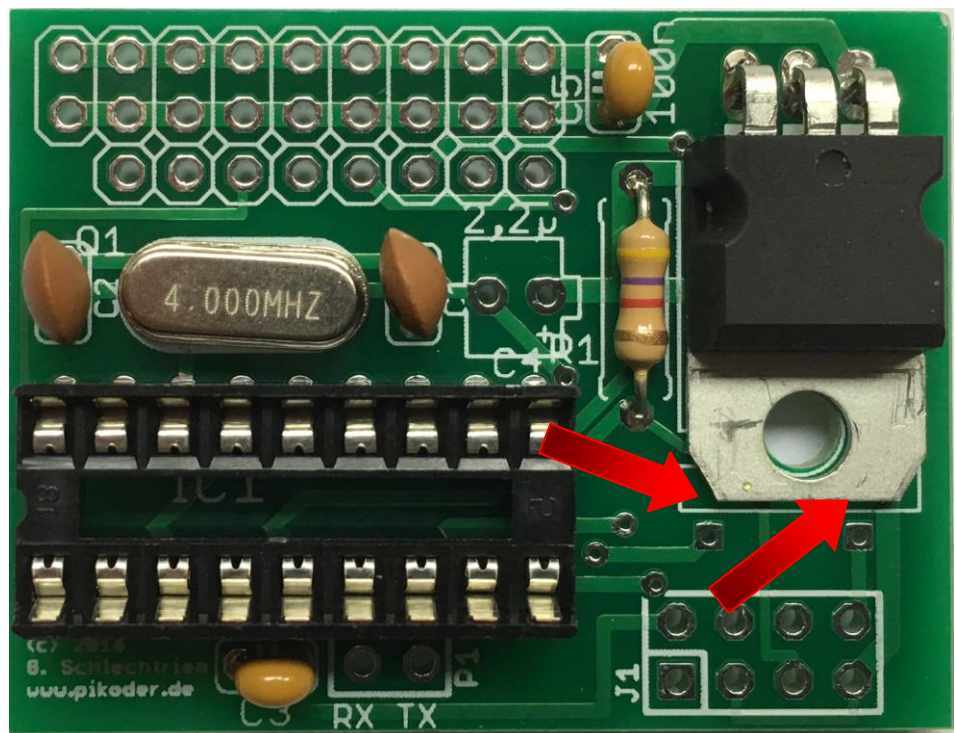
To avoid the socket from falling out of its position when turning the board slightly bend the two outer diagonal pins and then go ahead and solder all the other pins.





### Step 5: Equipment of the voltage regulator IC2

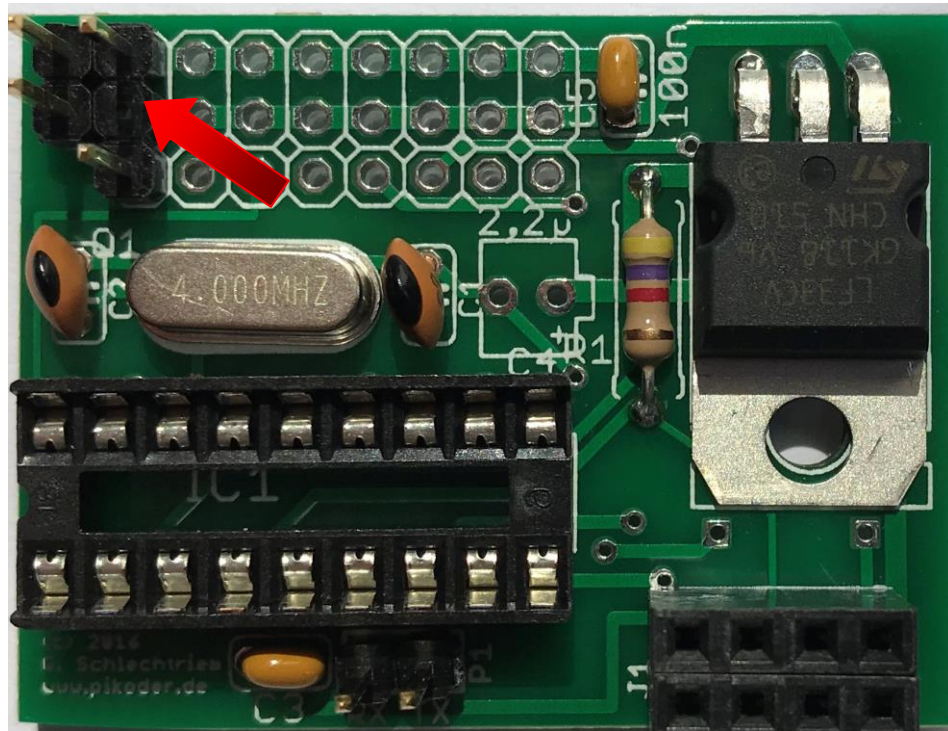
Now you would place the voltage regulator. The pins are already square bent to support a proper placement. Prior to soldering, please make sure that the whole in the PCB and the component match to avoid shorts. Note: the potentially critical points on the PCB are marked with red arrows in the following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



## Step 6: Equipment of the headers

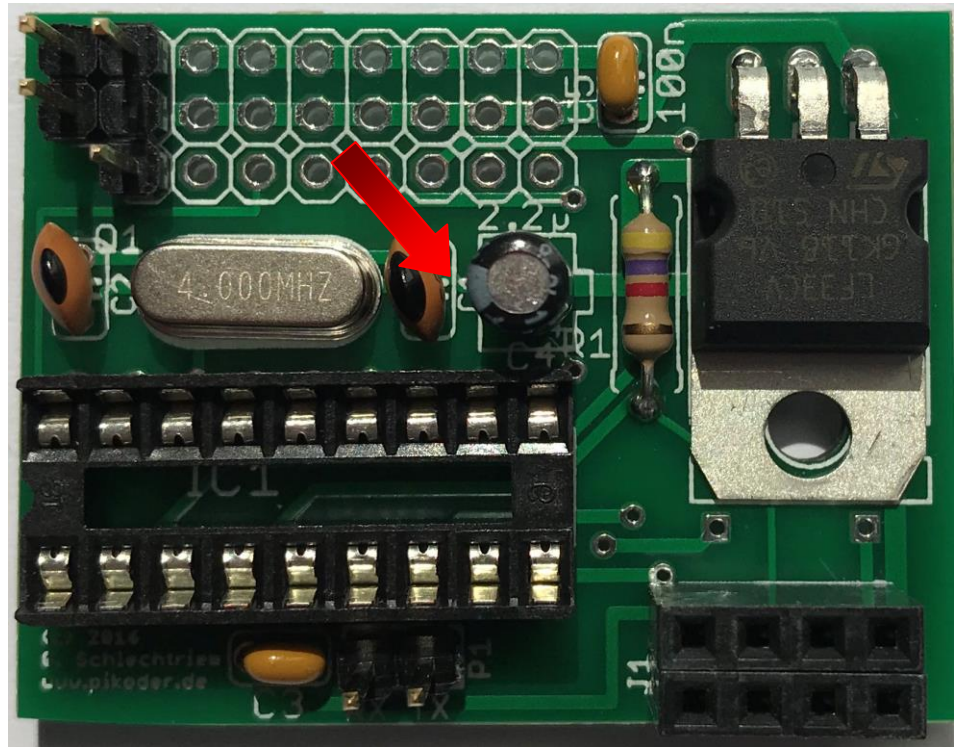
Now the two female headers (4 pins each forming an eight pin female header), the two pin header for the com interface and power supply and the PPM output header would be placed. Please refer to the image below for the correct placement.

Please note that the PPM output requires only two pins (signal and ground). The center pin would be the power supply when you connect your receiver to an FC. However, if you do not plan to supply your receiver from the FC then you should pull the center pin using a plier or a diagonal cutter to avoid a short between the different power sources as shown below. Please refer to the PiKoder/PPM wRX User's Manual for more information.



## 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/PPM wRX.

- This page has been left blank intentionally -



---

# 4

---

## **Commissioning**

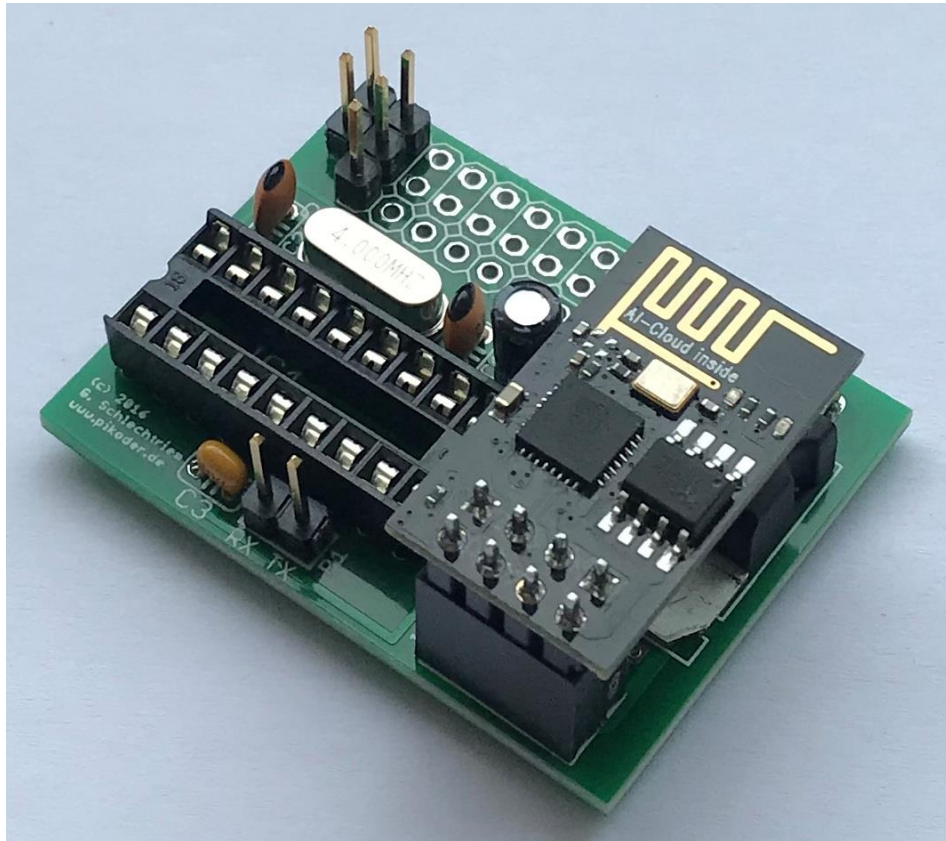
The commissioning of your PiKoder/PPM wRX starts off with programming the Wi-Fi radio to match your desired network settings. You would determine the network name (SSID) of your receiver and also determine your password. In order to simplify this process your Wi-Fi radio comes pre-programmed with the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

Then you would (optionally) program your PiKoder/PPM wRX-controller to reflect any specifics in your application. Then your receiver would be all set to be used with your Android Smart device as your new RC.

### **Programming of the WiFi radio ESP8266-01**

For the programming, please place the ESP8266 on its respective position on the PiKoder/PPM 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/PPM 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 with a 3.3 V signal level such as the one shown on the following page. These adapters are readily available on e.g. eBay or could be ordered at [www.pikoder.de](http://www.pikoder.de).

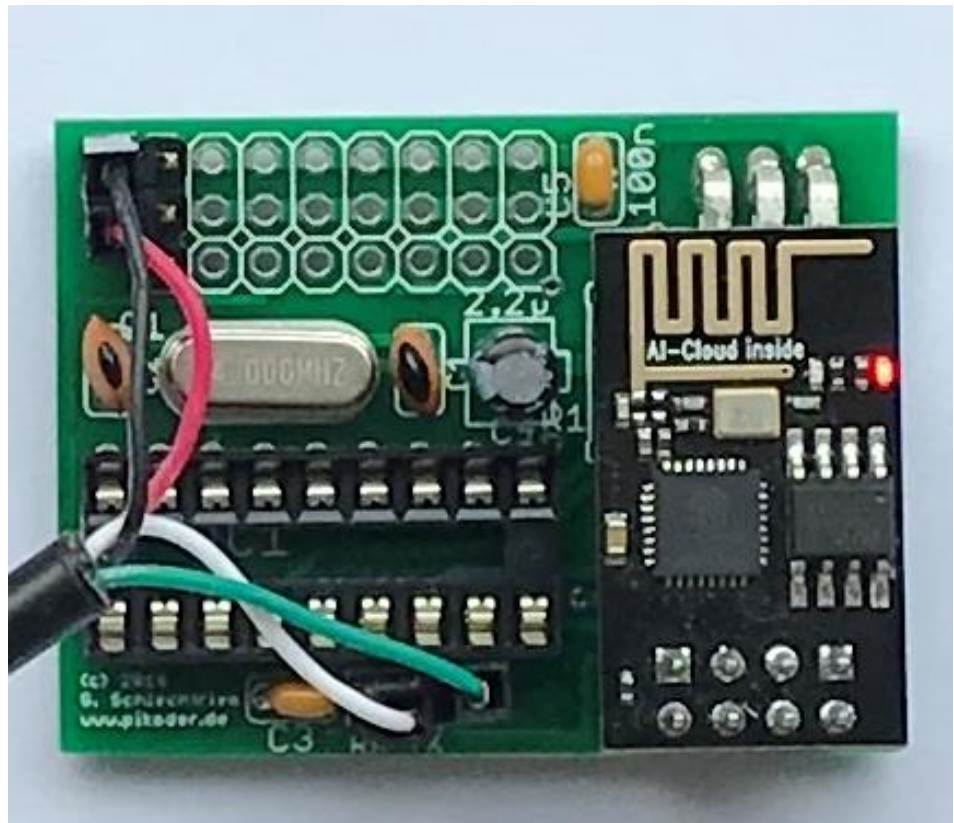


*PiKoder/PPM wRX w/o controller with ESP8266-01 ready for programming*



USB to UART adaptor

The following image shows the wiring of the PCB and the converter based on the color scheme on the previous picture. Upon connecting power 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.

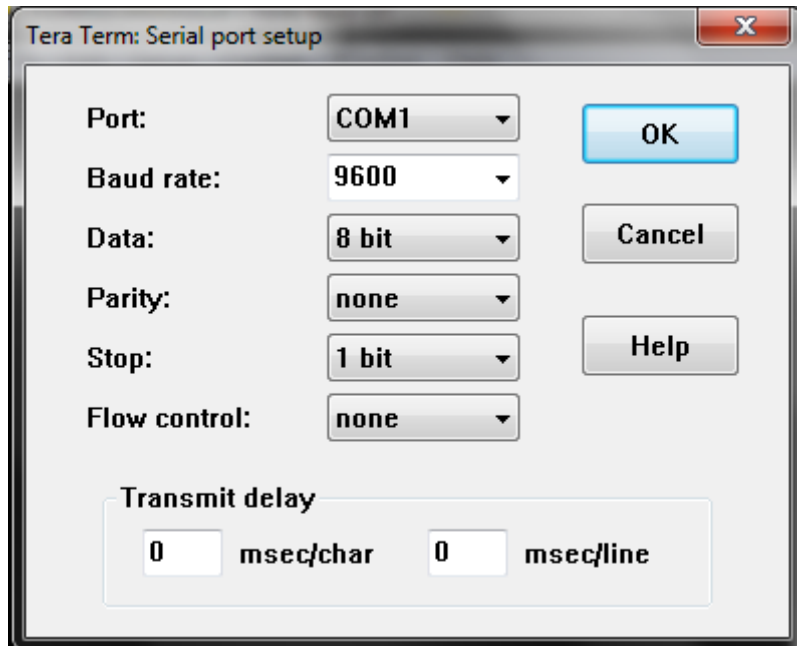


Please note that on the initial use of the USB-UART converter your PC will install driver software. Please wait until your system indicates the successful completion of the installation.

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>.

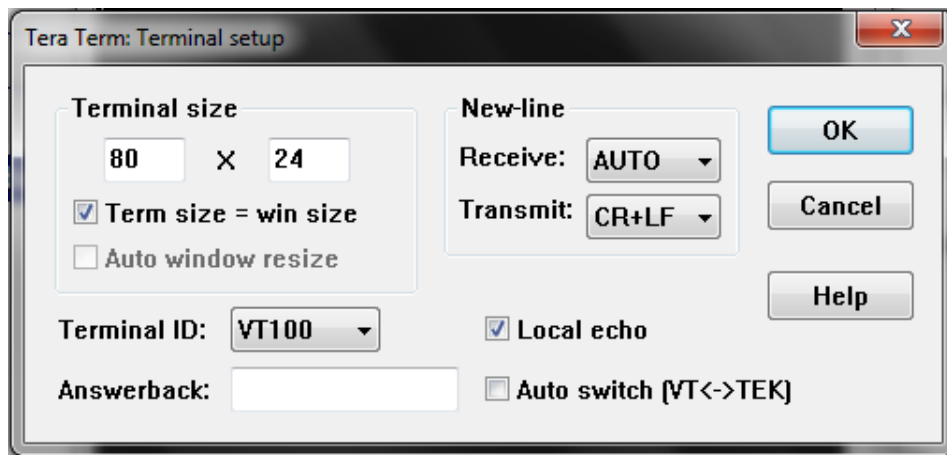
You start programming by setting the parameters of your computer's serial port. Starting Terra Term please select „Setup“, then „Serial port...“ and get the following screen:





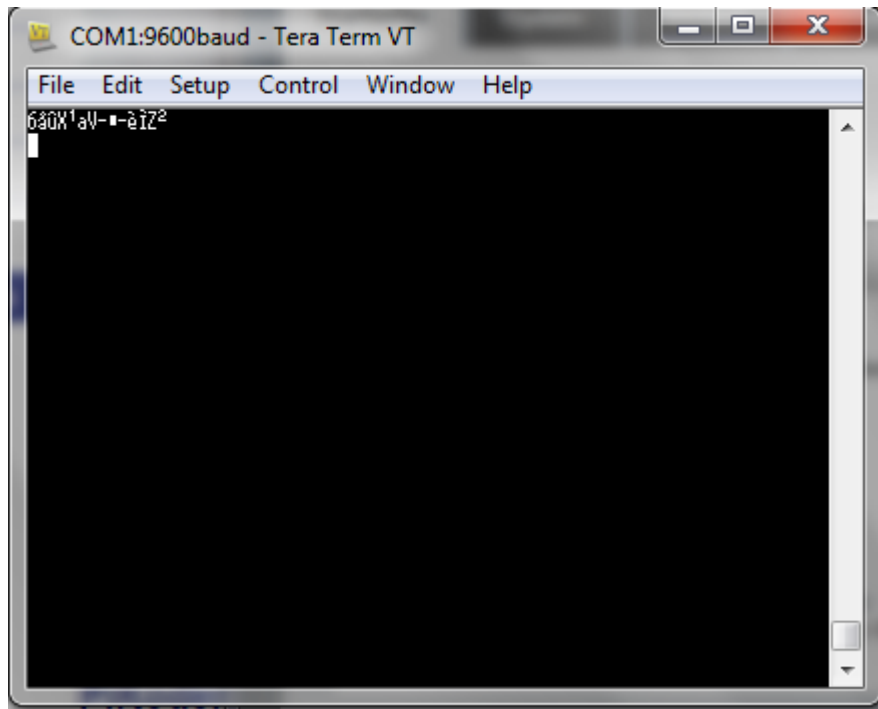
Select the COM port connected to the USB-UART converter and a Baud rate of 9600. Now hit OK.

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

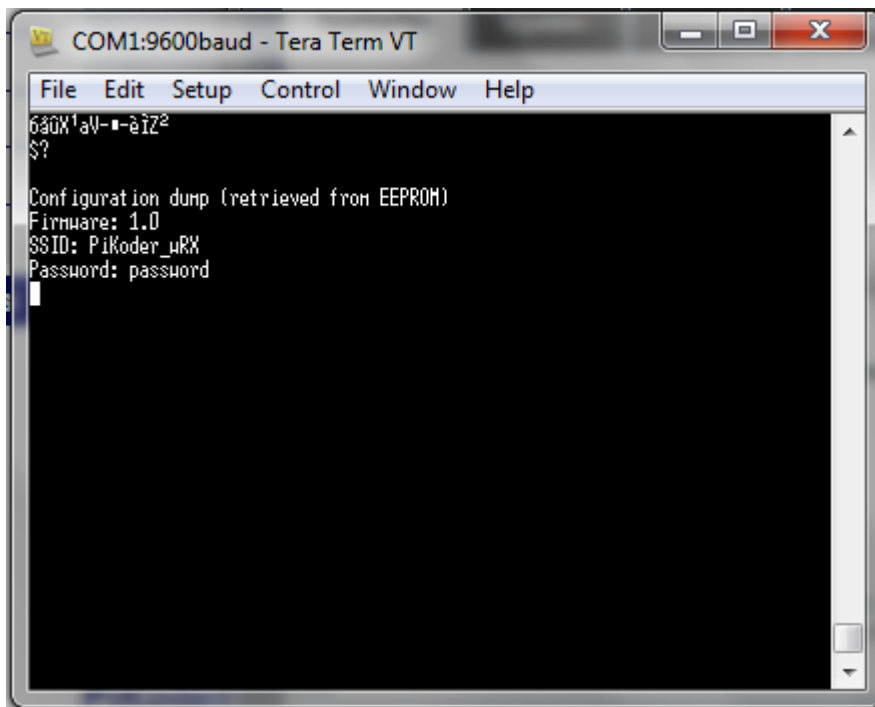


Make sure your settings match the above picture and hit OK again.

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



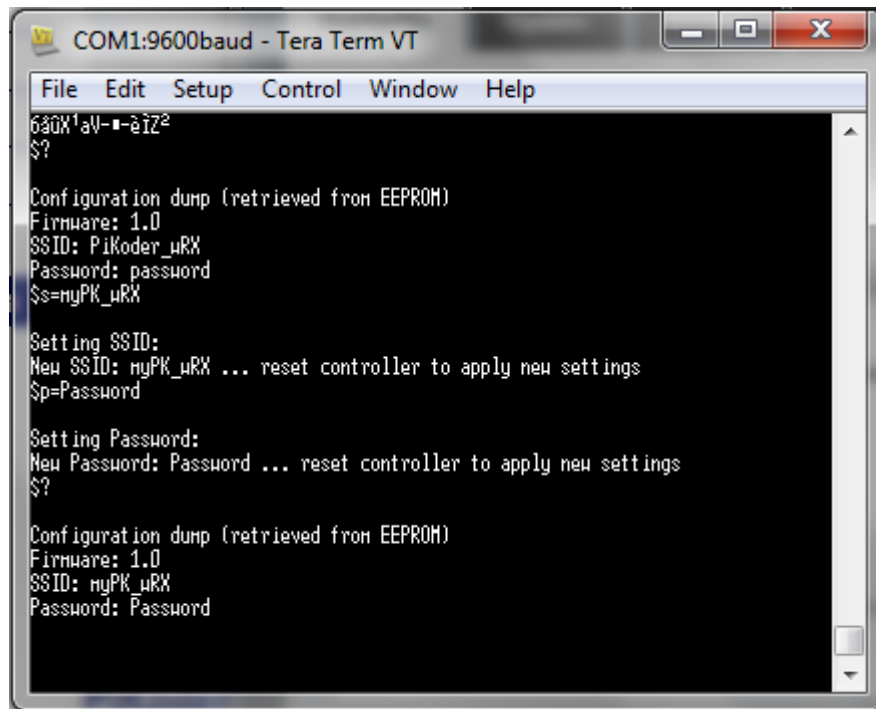
Start of by verifying the current settings of your radio. Therefore, please enter „\$?“ <ENTER>.



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.



```
COM1:9600baud - Tera Term VT
File Edit Setup Control Window Help
6ã0X1aV-■-ãîZ²
$?

Configuration dump (retrieved from EEPROM)
Firmware: 1.0
SSID: PiKoder_uRX
Password: password
$s=nyPK_uRX

Setting SSID:
New SSID: nyPK_uRX ... reset controller to apply new settings
$P=Password

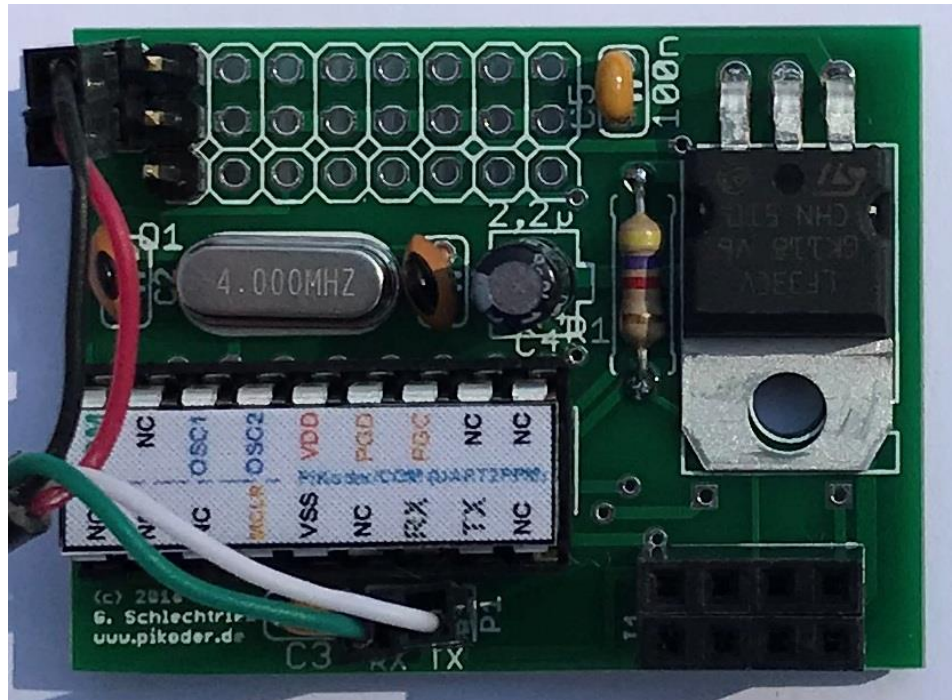
Setting Password:
New Password: Password ... reset controller to apply new settings
$?

Configuration dump (retrieved from EEPROM)
Firmware: 1.0
SSID: nyPK_uRX
Password: Password
```

## Programming of the PiKoder/PPM

In the next step you would program the PiKoder/PPM to reflect any specifics in your application such as values for max, min and fail safe per channel.

For connecting between 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 white cable) as shown in the following image.



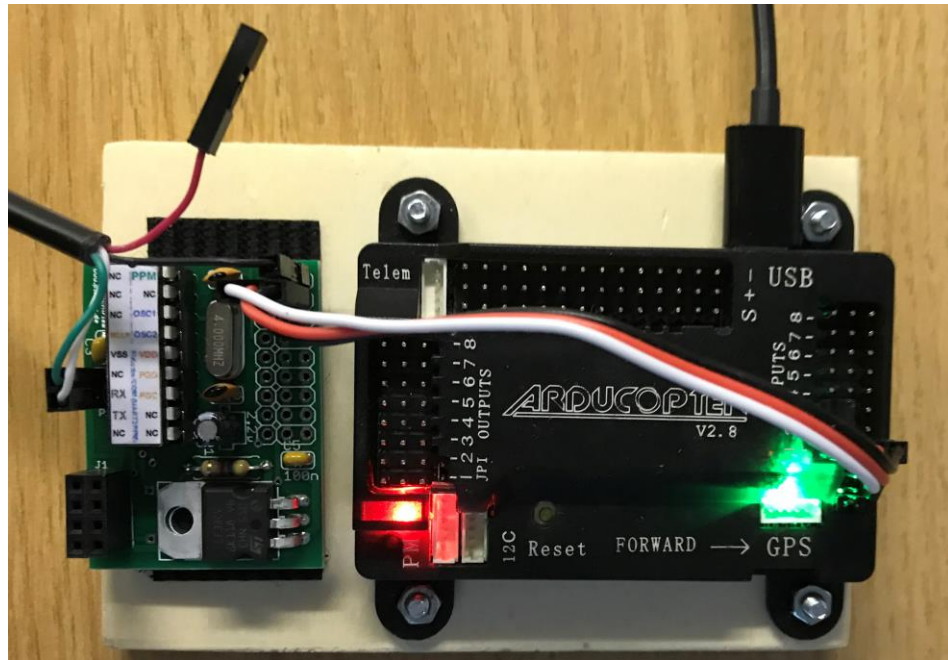
The programming requires again a terminal application or the [PCC \(PiKoder Control Center\)](http://www.pikoder.de/PiKoder_COM.html) offering a graphical user interface. Please refer to the PiKoder/PPM User's Guide on [http://www.pikoder.de/PiKoder\\_COM.html](http://www.pikoder.de/PiKoder_COM.html) for a detailed description of the available settings.

## Testing of the PPM output with an FC

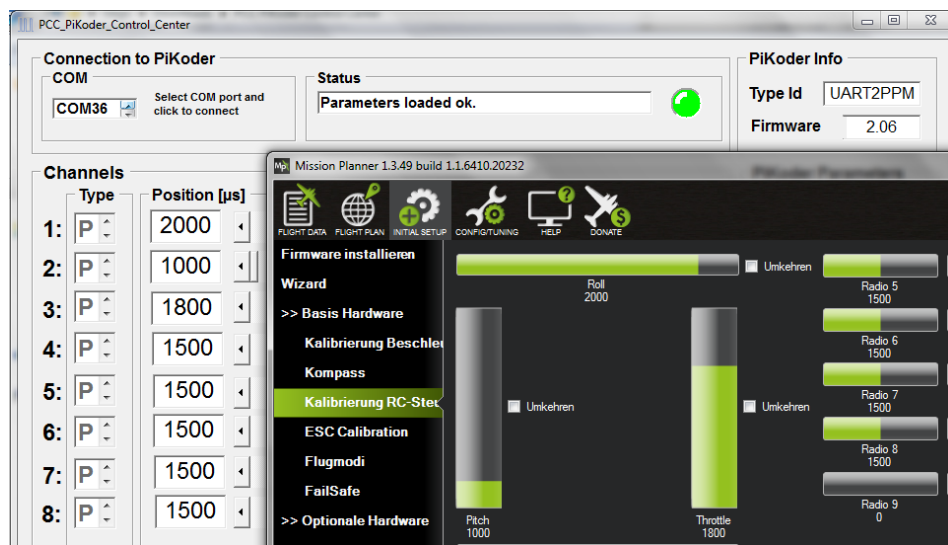
The PPM frame generated by the PiKoder/PPM can be verified using a scope but cannot be directly tested with a servo. If you wanted to test your PiKoder/PPM wRX before integrating it into your application you could use e.g. the RC calibration function of the Mission Planner.

You would connect to the FC via a USB cable and connect it to your PiKoder/PPM wRX using a standard servo cable extension. This cable will also provide for the power supply of the PiKoder. Please make also sure to configure the APM for PPM by placing a jumper between the signal pins of input channel two and three (please refer to the APM documentation).

Your PiKoder/PPM wRX would also be connected via the USB adapter to a second USB port of your computer. Please refer to the following picture regarding the cabling and **make sure that the red cable of the USB adapter is not connected to the receiver to avoid a short which might destroy your computer's USB port.**



After connecting the APM with the Mission Planner and the PiKoder/PPM with the PCC again, you can control the channel values by moving the sliders of the respective channel in the PCC as shown in the picture below.

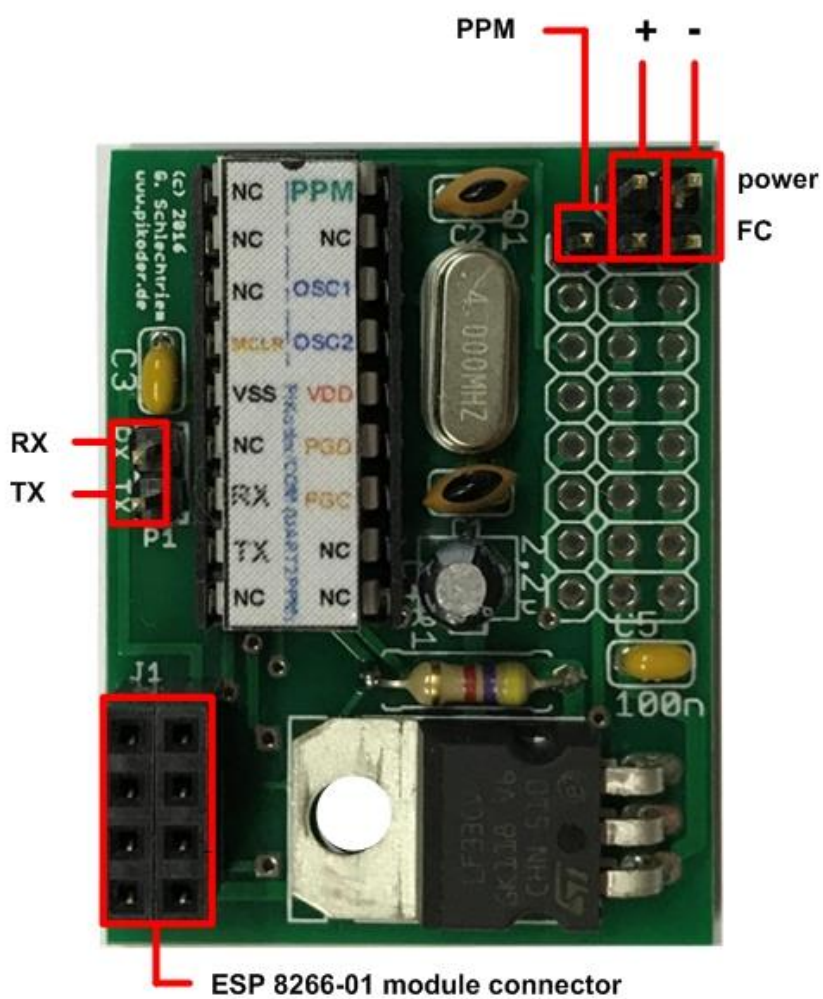


---

# A

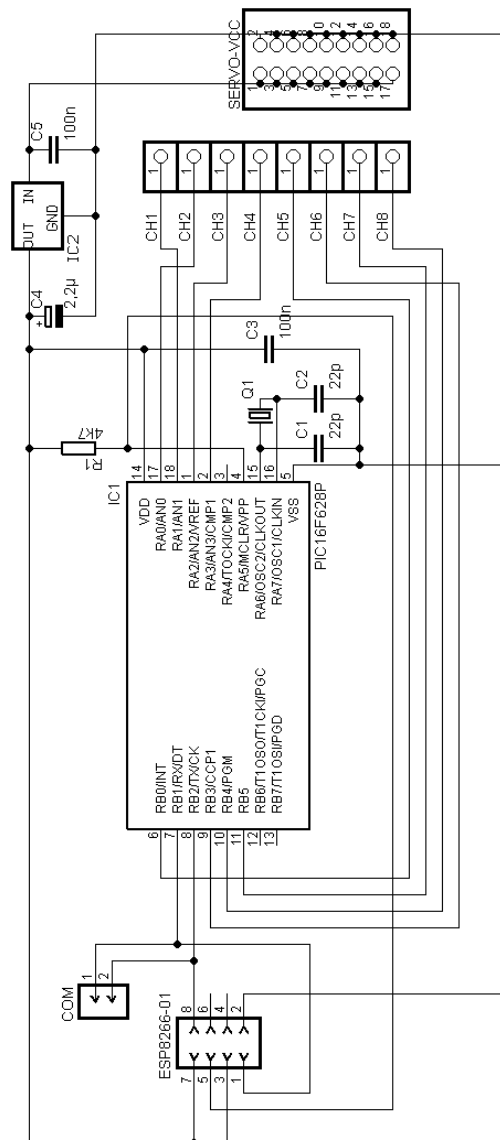
---

## Receiver pinning



# B

## Receiver schematic





---

# C

---

## ***Wi-Fi protocol***

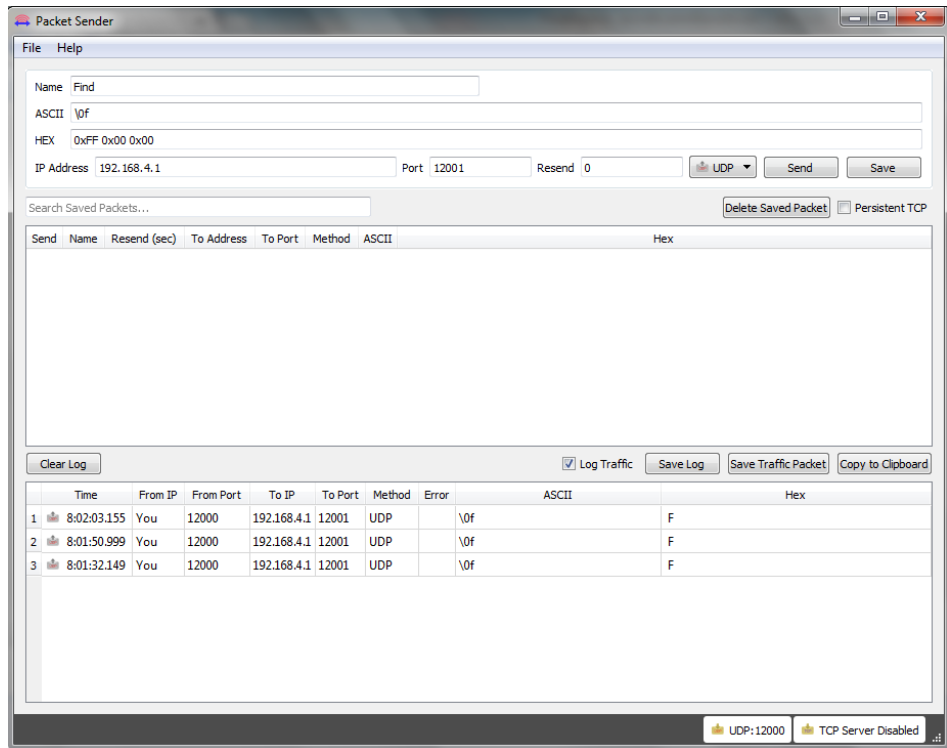
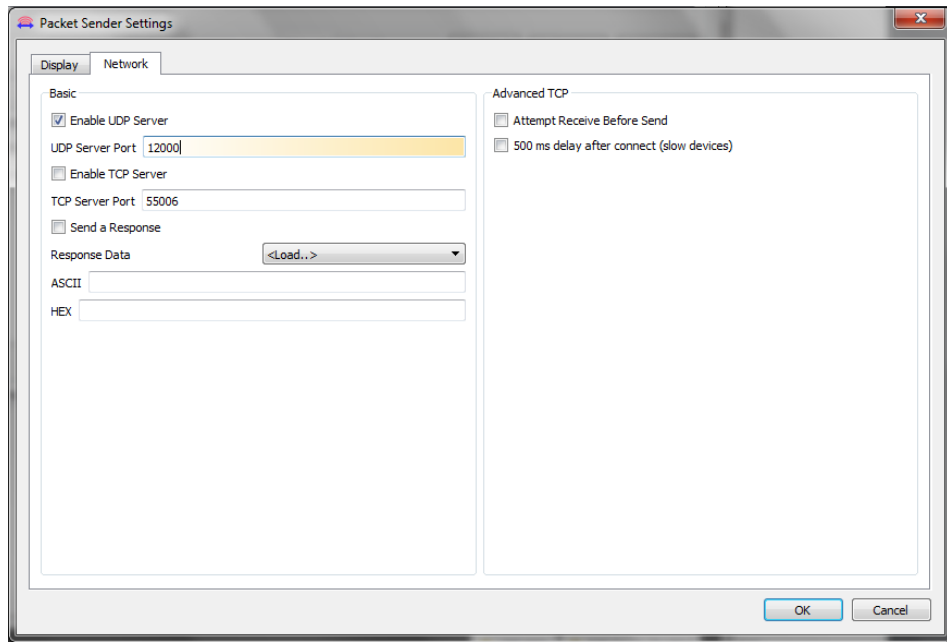
The communication between the PiKoder/PPM wRX and the Android smart device is based on the so called UDP (User Datagram Protocol – leading to the app’s name udpRC). This protocol allows the efficient exchange of small messages and thereby accommodating the typical requirements of radio controlled models.

This sections details the communication and offers additional tests of the Wi-Fi communication of the PiKoder/PPM 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.