

# *PiKoder/ PPM wRX*

## Bauanleitung

---

Version 1.0b  
Stand 11/2020

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

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

# Inhaltsverzeichnis

<b>Allgemeine Hinweise</b>	<b>3</b>
<b>Bausatzumfang und Hilfsmittel</b>	<b>5</b>
<b>Aufbau</b>	<b>7</b>
Widerstand R1.....	7
Bestückung der Kondensatoren C1-C3 und C5 .....	8
Bestückung des Quarzes.....	9
Bestückung des IC Sockels .....	9
Bestückung des Spannungsreglers IC2.....	10
Bestückung der Stecker- und Buchsenleisten.....	11
Bestückung des Kondensators C4.....	12
<b>Inbetriebnahme</b>	<b>13</b>
Parametrierung des WiFi Controllers ESP8266-01.....	13
Parametrierung PiKoder/PPM wRX.....	18
Test des PPM Ausgangssignals mit einem FC.....	19
<b>Anschlussbelegung Empfänger</b>	<b>21</b>
<b>Empfänger Schaltung</b>	<b>23</b>
<b>WiFi Protokoll</b>	<b>25</b>

## **Allgemeine Hinweise**

Bevor Sie mit dem Nachbau des *PiKoder/PPM wRX* beginnen, empfiehlt es sich, dass Sie diese Bauanleitung erst einmal bis zum Ende durchlesen. Sie wissen dann, worauf es ankommt und vermeiden dadurch von vornherein Fehler, die manchmal nur noch mit viel Aufwand zu beheben sind.

Für den Nachbau des *PiKoder/PPM wRX* werden Grundkenntnisse über elektronische Bauteile und Löten vorausgesetzt.

Führen Sie Lötungen und Verdrahtungen absolut sauber und gewissenhaft aus, verwenden Sie kein säurehaltiges Lötzinn, Lötfett o.ä. Vergewissern Sie sich, dass keine kalte Lötstelle vorhanden ist. Denn eine unsaubere Lötung oder schlechte Lötstelle, ein Wackelkontakt oder schlechter Aufbau bedeuten eine aufwändige und zeitraubende Fehlersuche und unter Umständen eine Zerstörung von Bauelementen.

Die Möglichkeit, dass nach dem Zusammenbau etwas nicht funktioniert, lässt sich durch einen gewissenhaften und sauberen Aufbau drastisch verringern. Kontrollieren Sie jeden Schritt, bevor Sie weiter gehen. Halten Sie sich an die Bauanleitung! Machen Sie den dort beschriebenen Schritt nicht anders und überspringen Sie nichts! Haken Sie jeden Schritt doppelt ab: einmal fürs Bauen, einmal fürs Prüfen.

Nehmen Sie sich auf jeden Fall Zeit: Basteln ist keine Akkordarbeit und soll in erster Linie Spaß machen!



---

# 2

---

## ***Bausatzumfang und Hilfsmittel***

Der vorliegende Bausatz enthält alle zum Aufbau eines *PiKoder/PPM wRX* benötigten Komponenten einschließlich des bereits programmierten Micro Controllers PIC16F628A und eines vorkonfigurierten WLAN Moduls ESP8266-01. Bitte überprüfen Sie zunächst die Vollständigkeit anhand der folgenden Stückliste.

### **Stückliste: PiKoder/PPM wRX**

<b>Widerstände</b>		<b>v</b>
4k7, 0,25W (gelb/lila/rot/gold)	R1	<input type="checkbox"/>
<b>Kondensatoren</b>		
22p	C1,C2	<input type="checkbox"/>
100n	C3,C5	<input type="checkbox"/>
2,2 $\mu$ F, 16 V	C4	<input type="checkbox"/>
<b>Halbleiter</b>		
PIC16F628A	IC1	<input type="checkbox"/>
LF 33CV	IC2	<input type="checkbox"/>
<b>Sonstiges</b>		
Quarz 4 MHz	Q1	<input type="checkbox"/>
IC Fassung 18 pol.		<input type="checkbox"/>
Buchsenleiste 4 pol. (2x)		<input type="checkbox"/>
Stiftleiste 2 pol.	P1	<input type="checkbox"/>
Stiftleiste 3 pol.		<input type="checkbox"/>
ESP8266-01 WiFi Modul		<input type="checkbox"/>
Leiterplatte PiKoder/SSC wRX		<input type="checkbox"/>

Darüber hinaus werden für den Aufbau dieses Bausatzes folgende Hilfsmittel benötigt:

1. Elektroniklötkolben
2. Elektroniklötzinn
3. Seitenschneider zur Kürzung der Bauteilanschlüsse

Für die Inbetriebnahme dieses Bausatzes sind außerdem erforderlich:

1. Spannungsversorgung 4.8V .. 6V (z.B. 4 AA Batterien oder Akkus)
2. Serial auf USB Schnittstellenwandler mit 3,3 V Signalpegel
3. PC mit Anwendungssoftware (Download aus dem Internet)
4. Flight Controller (FC) mit PPM Eingang z.B. ein APM (optional)

---

# 3

---

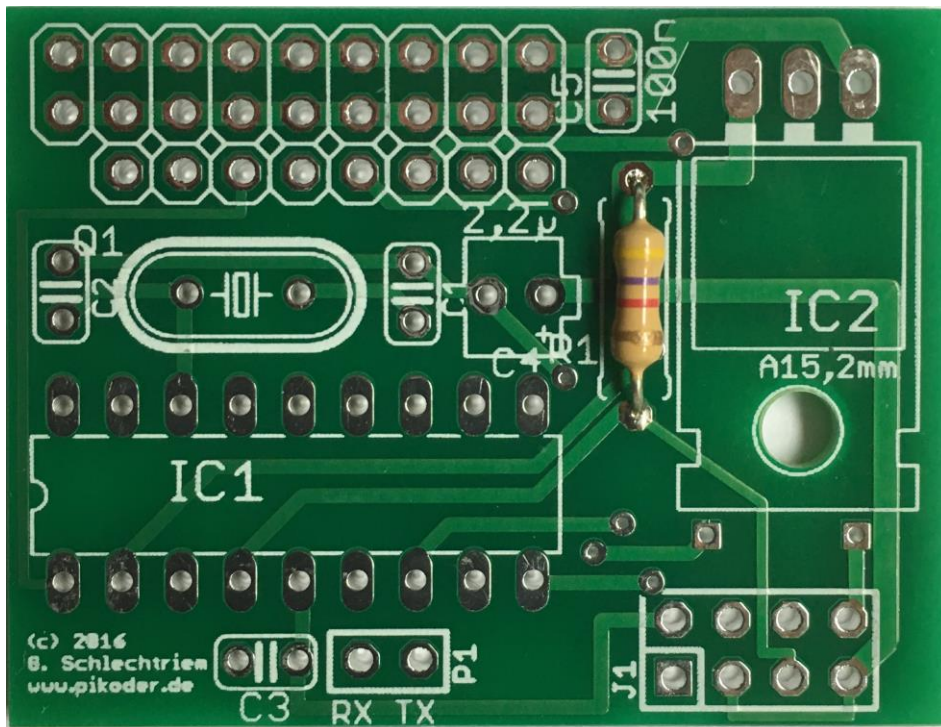
## ***Aufbau***

Die folgenden Abschnitte beschreiben die Montage der Bauelemente auf der Platine. Alle Bauteile befinden sich auf der Bestückungsseite und der Bestückungsvorgang selbst wird durch den Bestückungsaufdruck der Leiterplatte unterstützt.

Die Reihenfolge der Bestückung hängt von der Höhe der Bauteile ab – es gilt grundsätzlich, dass zunächst die flachsten Bauteile bestückt werden.

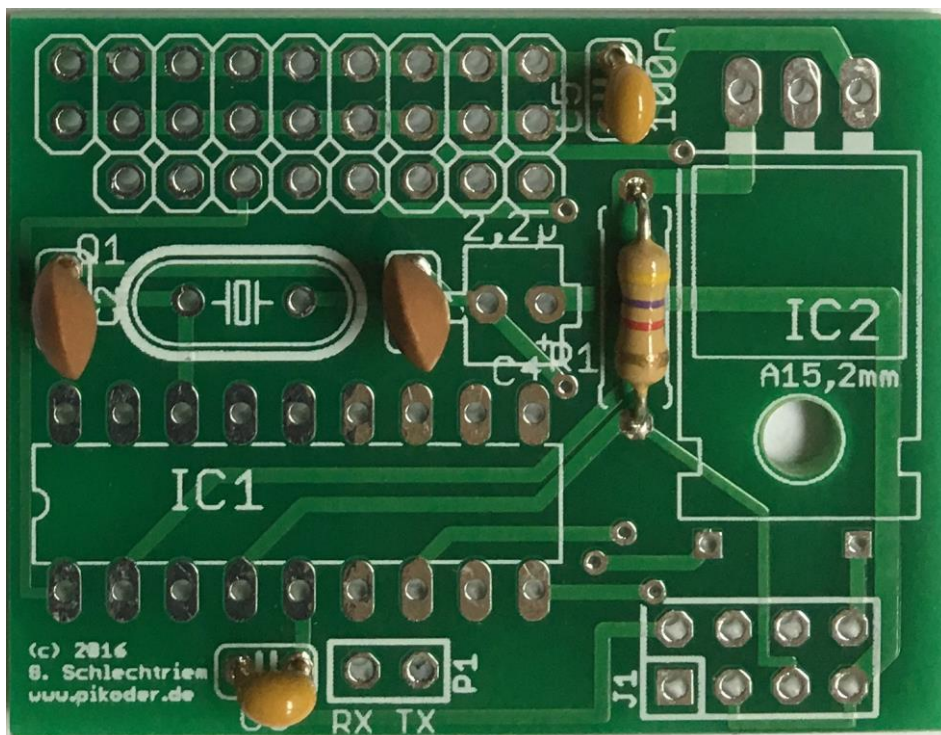
### **Widerstand R1**

Zuerst wird der Widerstand R1 bestückt. Hierzu werden die Anschlussdrähte entsprechend dem Rastermaß rechtwinkelig abgebogen und in die vorgesehenen Bohrungen laut Bestückungsplan gesteckt. Damit die Bauteile beim Umdrehen der Platine nicht herausfallen können, biegen Sie die Anschlussdrähte ca. 45° auseinander und verlöten diese dann sorgfältig mit den Leiterbahnen auf der Rückseite der Platine. Anschließend werden die überstehenden Drähte abgeschnitten.



## Bestückung der Kondensatoren C1-C3 und C5

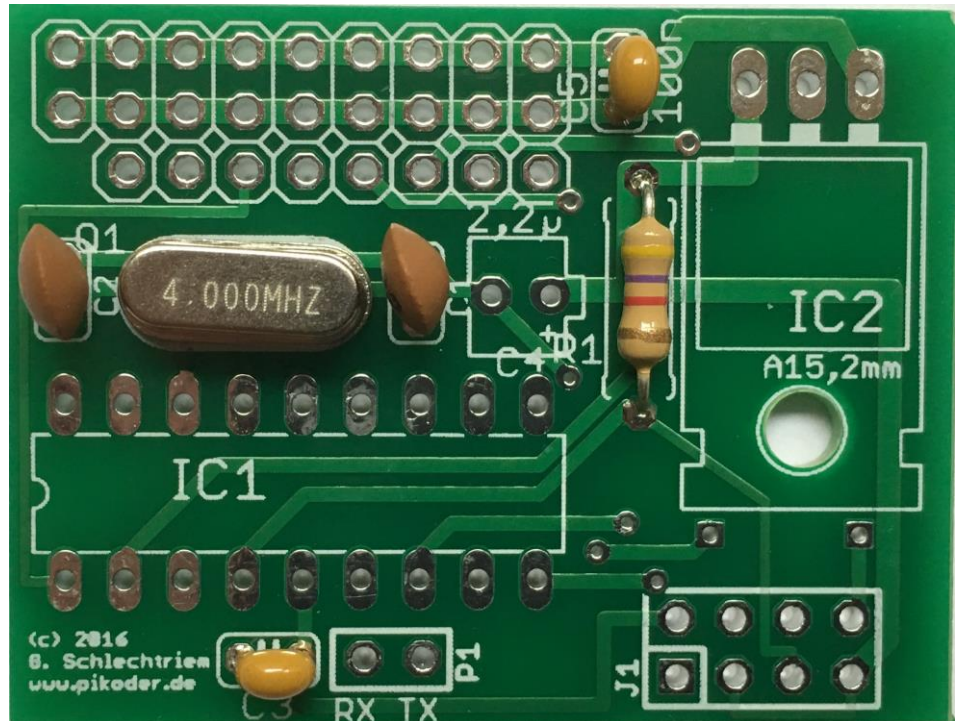
Nun werden die Kondensatoren C1 – C3 und C5 eingesetzt und die Anschlüsse wiederum leicht angewinkelt, um das Herausfallen zu verhindern. Die vier Kondensatoren sind ungepolte Keramikkondensatoren – von daher ist die Ausrichtung egal. Bitte beachten Sie, dass es sich bei C1 und C2 um Kondensatoren mit einer Kapazität von 22p (Aufdruck 22..) handelt, bei C3 und C5 um Typen mit 100n.





## Bestückung des Quarzes

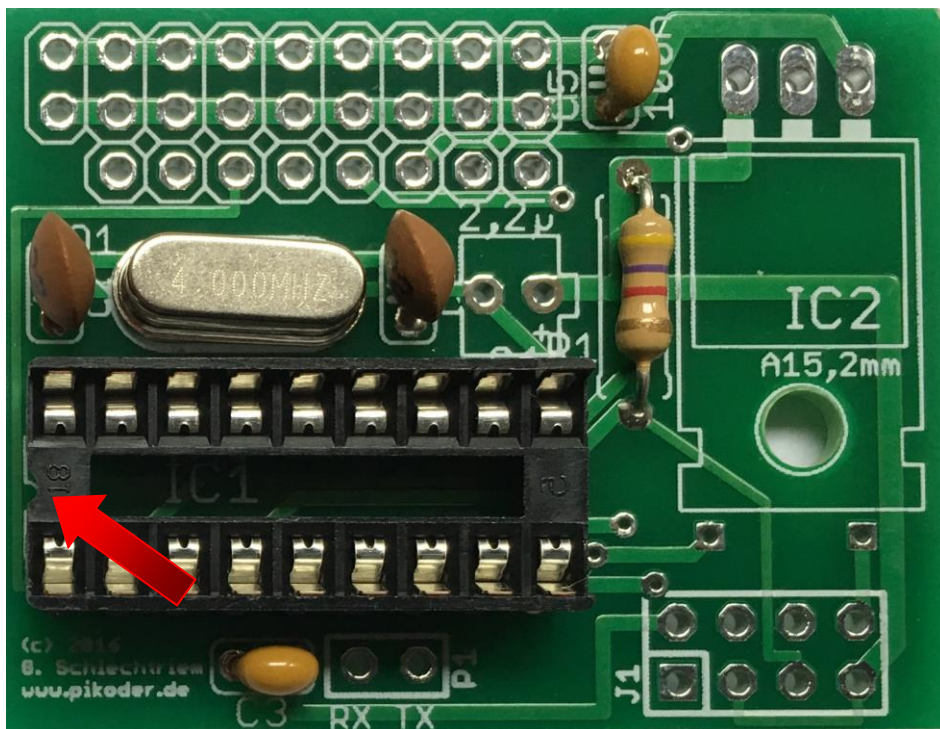
Es folgt das Einsetzen und Verlöten des Quarzes- auch hier ist die Ausrichtung unkritisch.



## Bestückung des IC Sockels

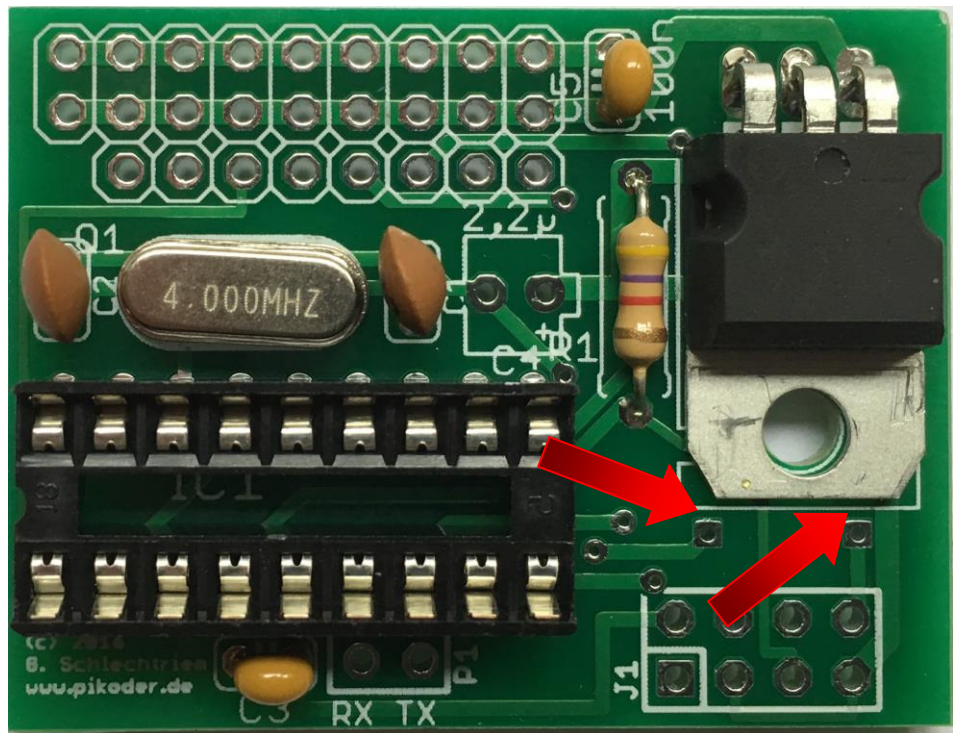
Stecken Sie die Fassung für den Controller in die entsprechende Position auf der Bestückungsseite der Leiterplatte. Beachten Sie hierbei die Einkerbung (im Bild unten auf der linken Seite) oder sonstige Kennzeichnung an der Stirnseite der Fassung. Dies ist die Markierung (PIN1) für das IC, welches später einzusetzen ist.

Um zu verhindern, dass beim Herumdrehen der Platine zum Lötten die Fassung wieder heraus fällt, werden zwei schräg gegenüberliegende Pins der Fassung umgebogen und danach werden alle Anschlüsse verlötet.



## Bestückung des Spannungsreglers IC2

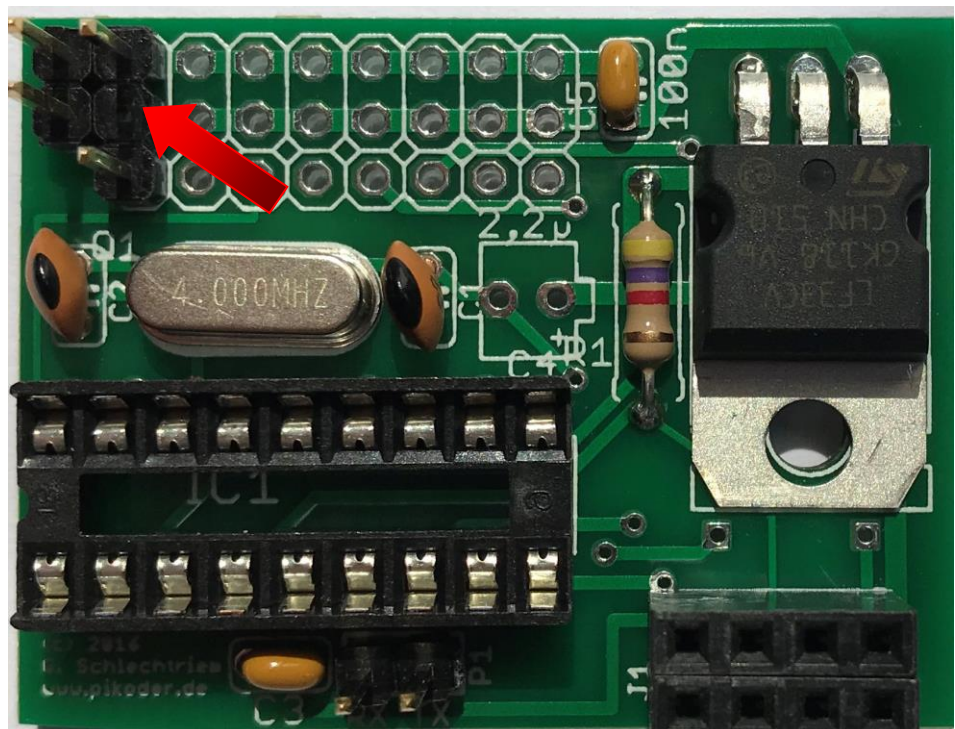
Als nächstes wird der Spannungsregler IC2 bestückt. Die Anschlussbeine sind bereits rechtwinklig abgebogen, um eine sichere Bestückung zu ermöglichen. Überzeugen Sie sich vor dem Verlöten, dass die Bohrungen möglichst sauber übereinander liegen, damit Kurzschlüsse vermieden werden. Hinweis: die möglicherweise kritischen Durchkontaktierungen sind im folgenden Bild mit roten Pfeilen gekennzeichnet.



## Bestückung der Stecker- und Buchsenleisten

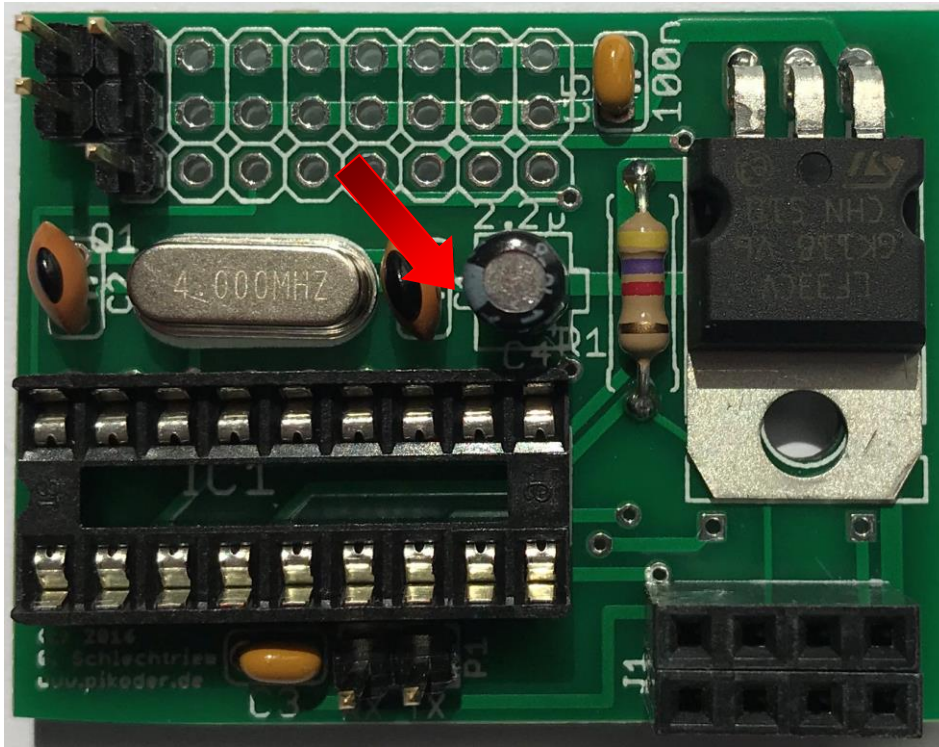
Bestücken Sie nun die Platine mit der achtpoligen Buchsenleiste (bestehend aus 2 vierpoligen Buchsenleisten) und der zweipoligen Stiftleiste.

Für das PPM Signal werden nur zwei Pins benötigt (Signal und GND). Der mittlere PIN dient der Spannungsversorgung, wenn Sie den Empfänger mit dem FC verbinden. Wenn Sie den Receiver nicht über den FC versorgen wollen, dann empfiehlt es sich, den mittleren PIN zu entfernen, um mögliche Kurzschlüsse zu vermeiden. Weitere Informationen finden Sie auch im User Manual.



## Bestückung des Kondensators C4

Schließlich wird noch der Kondensator C4 eingesetzt und verlötet. Hierbei ist zu beachten, dass es sich bei diesem Kondensator um einen gepolten Typen handelt. Daher ist der mit einem Minus gekennzeichnete und kürzere Anschlussdraht in die Bohrung auf der Seite des Quarzes einzusetzen (roter Pfeil im nachfolgenden Bild).



Damit ist die Bestückung der Leiterplatte abgeschlossen. Nun erfolgt die schrittweise Inbetriebnahme.

## ***Inbetriebnahme***

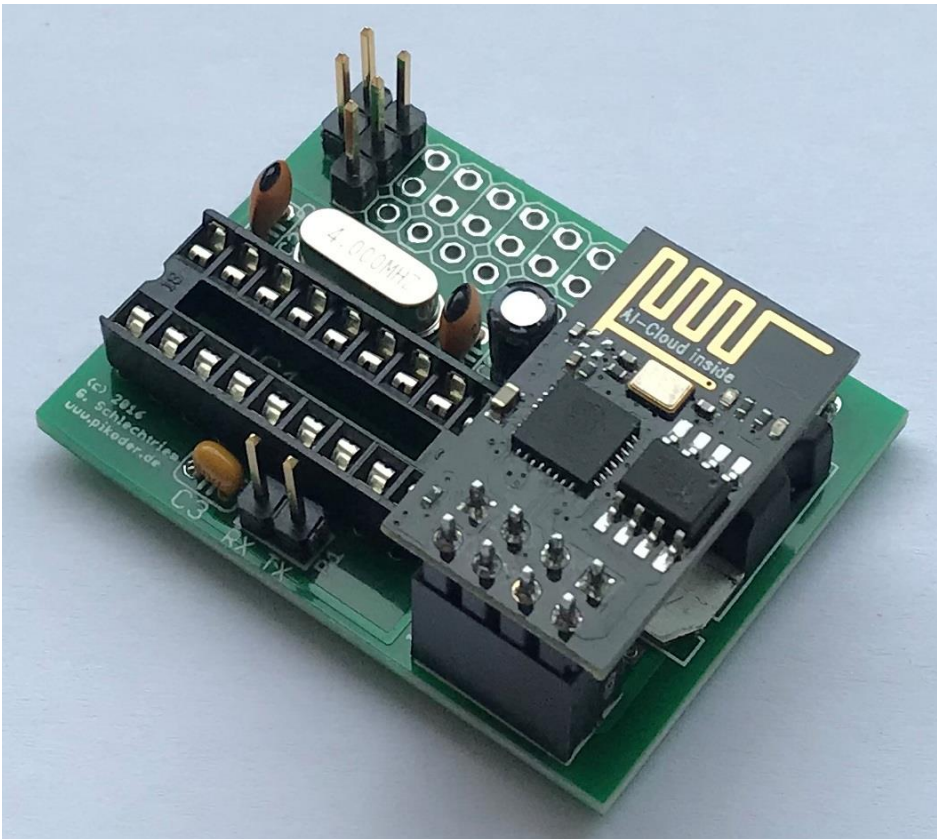
Die Inbetriebnahme der Leiterplatte erfolgt in zwei Schritten. Zunächst wird die Kennung (SSID) und das Password für den WiFi-Baustein ESP8266 konfiguriert. Zur Vereinfachung der Parametrierung ist der Controller bereits mit einer weitgehend vorkonfigurierten Firmware programmiert.

Im folgenden, optionalen Schritt wird der PiKoder/PPM wRX-Controller parametrierung und im Zusammenspiel mit einem Flight Controller (FC) getestet.

### **Parametrierung des WiFi Controllers ESP8266-01**

Die Parametrierung wird durchgeführt, indem der ESP8266 auf die *PiKoder/PPM wRX* Leiterplatte aufgesteckt wird (s. Bild auf der folgenden Seite). Die Werte, die wir im Folgenden einstellen, werden im Flash-Speicher des ESP-Controllers abgelegt und stehen dann bei jedem Neustart zur Verfügung. Der PiKoder darf während der Programmierung noch nicht in seinen Sockel eingesetzt sein.

Der ESP8266 wird über eine Terminalanwendung auf dem PC unter Verwendung von einigen wenigen Kommandos programmiert. Um die Verbindung zwischen dem WLAN Controller und dem PC herzustellen ist ein USB-UART-Adapter mit einem 3,3 Volt Signalpegel erforderlich wie beispielsweise das im Bild auf der folgenden Seite dargestellte Kabel. Solche Adapter sind bei eBay erhältlich bzw. können auch als Zubehör zum *PiKoder/PPM wRX* über [www.pikoder.de](http://www.pikoder.de) bezogen werden.

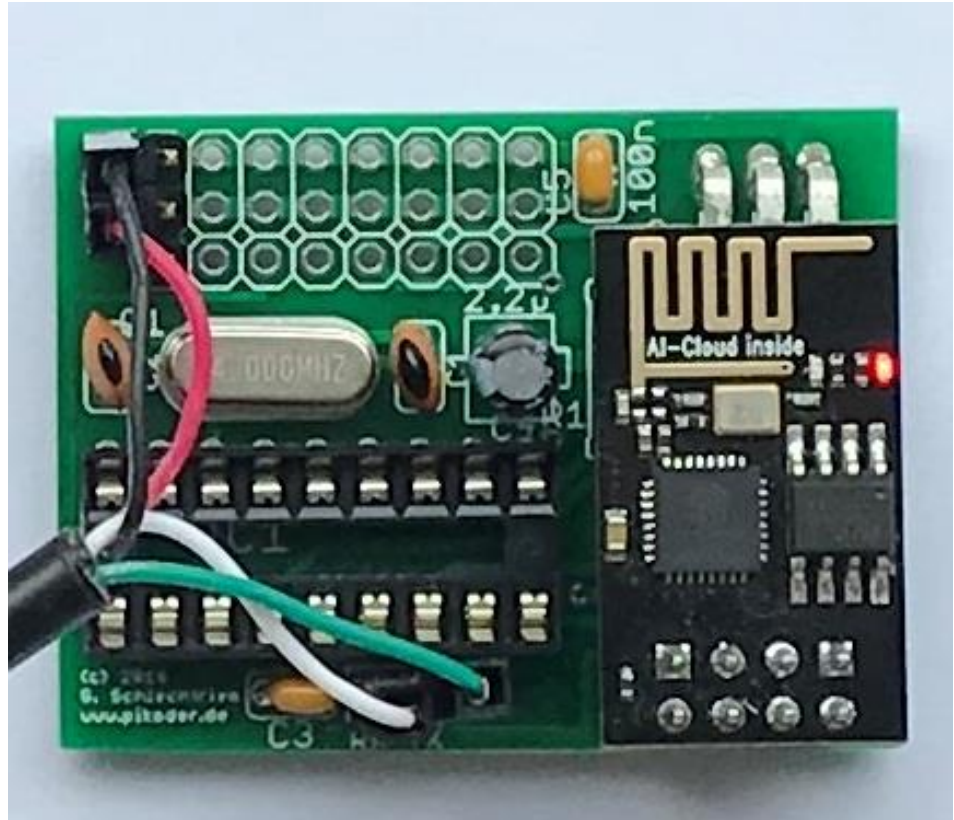


**PiKoder/PPM wRX ohne Controller mit aufgestecktem ESP8266-01**



**Adapter USB to UART zur Verbindung PC mit PiKoder/PPM wRX**

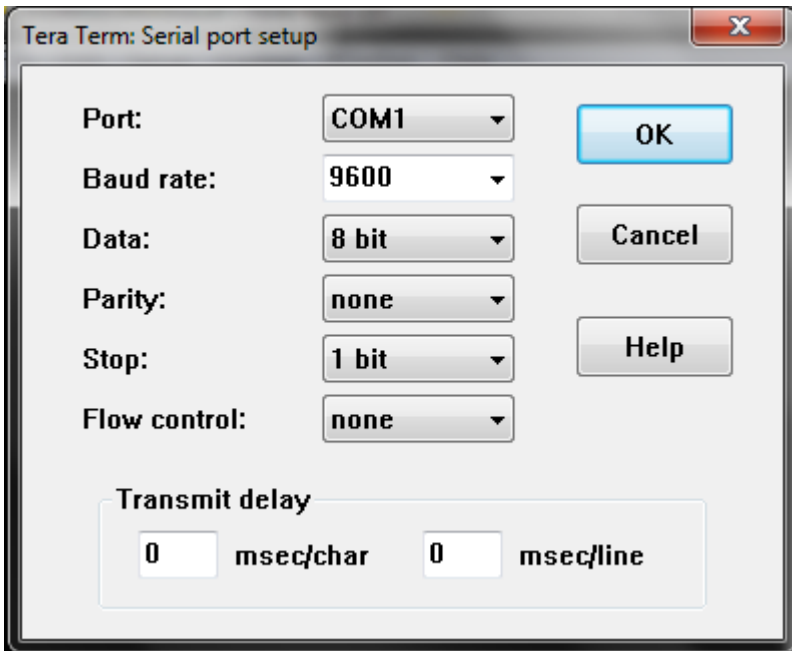
Im nachfolgenden Bild ist die Verbindung zwischen dem Adapter und der Platine dargestellt. Beim Einstecken des USB Steckers leuchten eine rote und eine blaue LED auf. Die blaue LED zeigt Aktivitäten auf der seriellen Schnittstelle an und verlöscht nach sehr kurzer Zeit wieder; die rote LED leuchtet, so lange Betriebsspannung anliegt.



Wenn Sie den USB-UART-Adapter zum ersten Mal einsetzen, wird Ihr PC zunächst die notwendigen Treiber installieren. Bitte warten Sie in diesem Fall zunächst ab, bis das Betriebssystem meldet, dass die neuen Komponenten jetzt genutzt werden können.

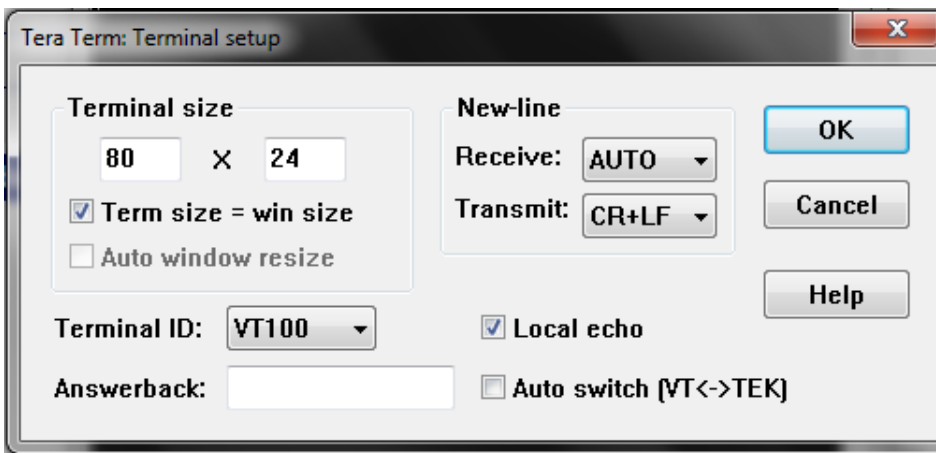
Zur Programmierung des ESP8266 benötigen Sie ein Terminalprogramm. Die folgenden Schritte werden auf Basis von Tera Term beschrieben, einem kostenlosen, Open Source Terminalprogramm, das Sie einfach herunterladen und installieren können (<https://tssh2.osdn.jp/index.html.en>).

Zunächst ist die serielle Schnittstelle auf die Übertragungsparameter des ESP8266 einzustellen. Hierzu wählen Sie „Setup“, dann „Serial port...“ und erhalten ein Fenster, wie im folgenden Bild:



Wählen Sie die serielle Schnittstelle, an die Sie Ihr USB-Kabel angeschlossen haben und stellen die Baud rate auf 9600. Klicken Sie dann OK.

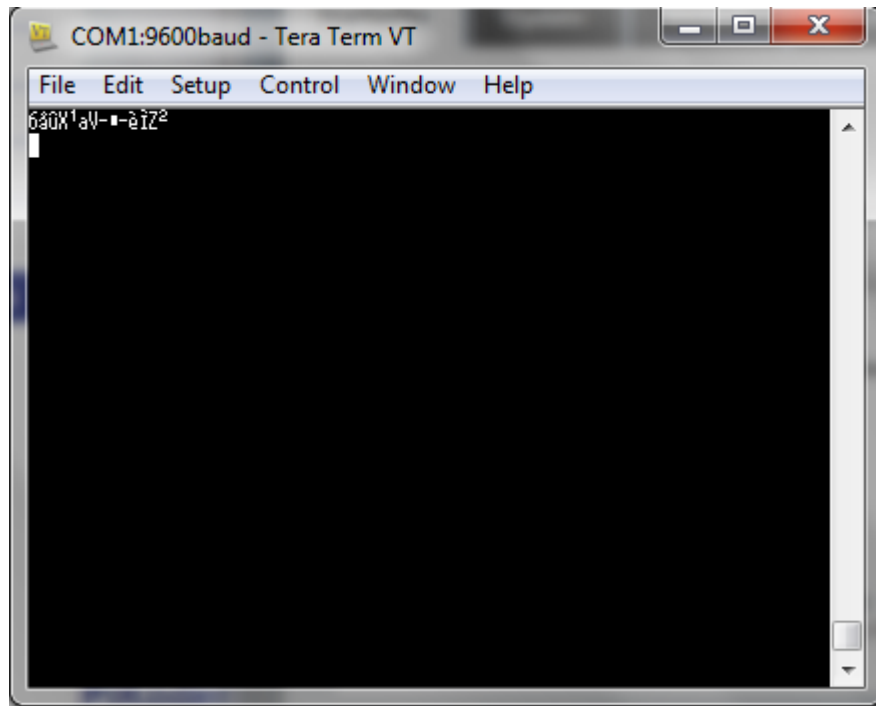
Die nächste Einstellung betrifft das Terminal selber. Nachdem Sie erneut „Setup“ und dann „Terminal...“ gewählt haben, erscheint folgendes Fenster:



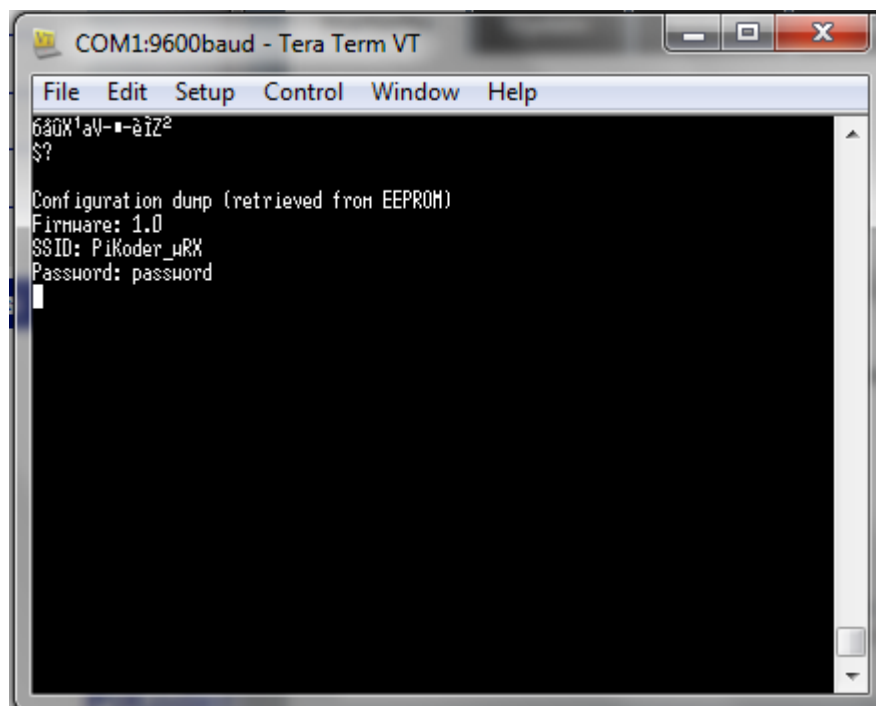
Übernehmen Sie die Einstellungen gemäß Bild und bestätigen die Auswahl mit OK.

Nun ist das Terminal für die Kommunikation mit dem ESP8266 bereit und Sie können die Programmierung nach einem Reset des ESP8266 (rotes Kabel kurz abziehen und wieder aufstecken) starten. Als Bestätigung des Reset erhalten Sie eine nicht lesbare Datenausgabe des ESP8266 (s.u.), mit der dieser seine Betriebsbereitschaft anzeigt.





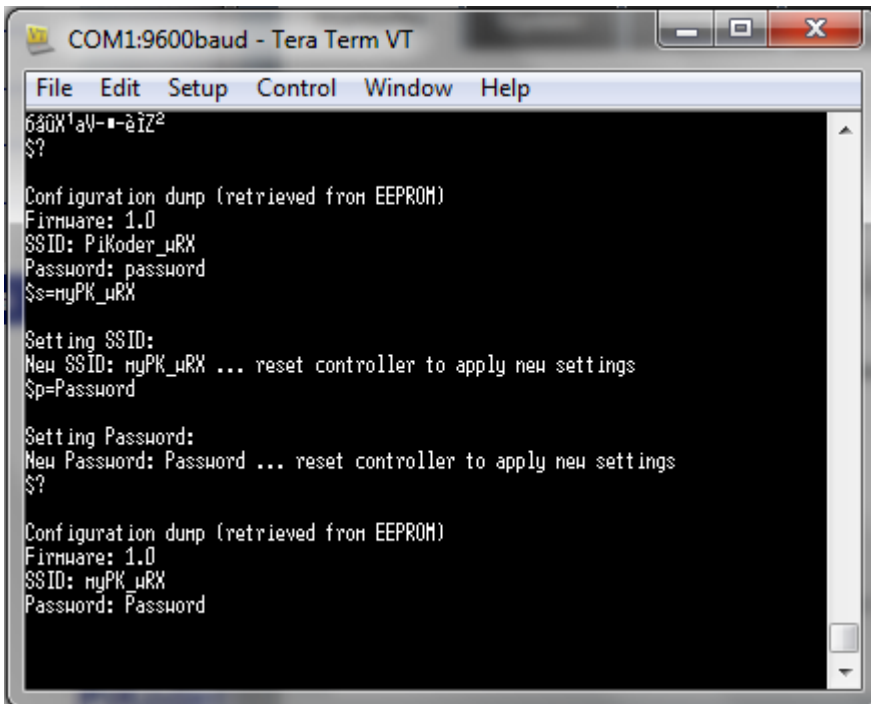
Im nächsten Schritt verifizieren Sie die aktuellen Einstellungen durch Eingabe des Befehls „\$?“ <ENTER>.



Die Ausgabe zeigt die SSID, also den Namen des Access-Points an, mit dem Sie sich später von Ihrem Smartphone aus verbinden (im konkreten Fall lautet der Name „PiKoder\_wRX“). Außerdem ist das Passwort angegeben, das Sie für die Anmeldung benötigen (hier: „password“). Wenn Sie diese Einstellungen übernehmen wollen, dann ist die Parametrierung des ESP8266 bereits abgeschlossen (Hinweis: Sie können die Einstellungen auch später noch anpassen).

Mit dem Kommando „\$s=“ passen Sie die SSID an. Im folgenden Bild sehen Sie den Ablauf. Der von Ihnen eingegebene Wert wird ausgegeben und im EEPROM des Controllers abgespeichert. Bitte beachten Sie, dass die neue SSID erst nach einem Neustart des ESP8266 wirksam wird.

Zur Änderung des Passwords verwenden Sie das Kommando „\$p=“; der Ablauf ist ebenfalls im nachfolgenden Screenprint nachvollziehbar. Sinnvollerweise überprüfen Sie nach Abschluss der Parametrierung erneut die aktuellen Einstellungen mit „\$?“.



```
COM1:9600baud - Tera Term VT
File Edit Setup Control Window Help
640X140-1-21Z²
$?

Configuration dump (retrieved from EEPROM)
Firmware: 1.0
SSID: PiKoder_wRX
PasswOrd: passwOrd
$s=myPK_wRX

Setting SSID:
Neu SSID: myPK_wRX ... reset controller to apply new settings
$p=PasswOrd

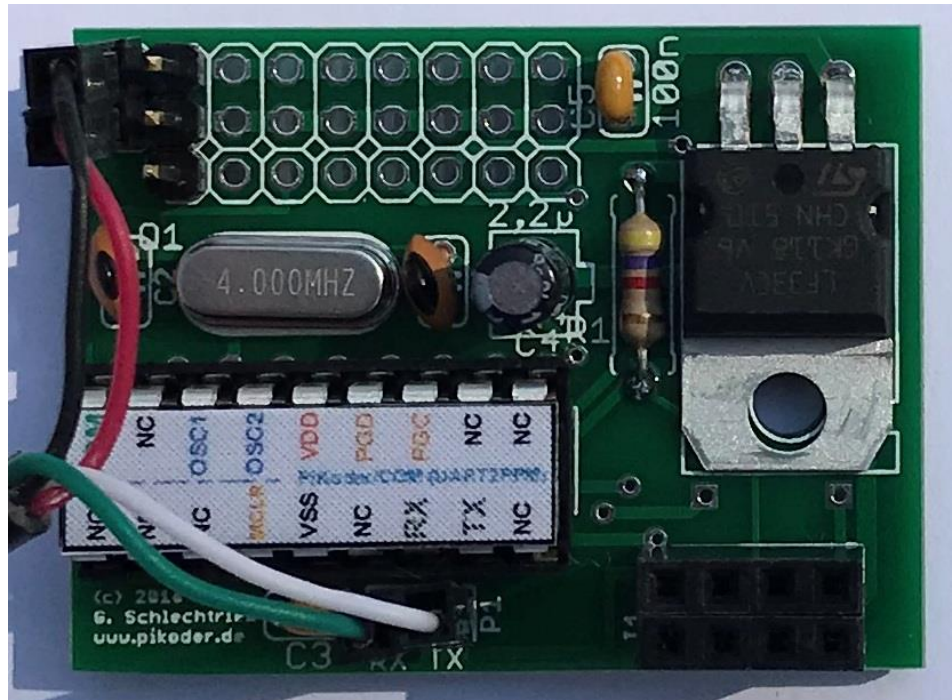
Setting Password:
Neu Password: PasswOrd ... reset controller to apply new settings
$?

Configuration dump (retrieved from EEPROM)
Firmware: 1.0
SSID: myPK_wRX
PasswOrd: PasswOrd
```

## Parametrierung PiKoder/PPM wRX

Im nächsten Schritt können nun optional die Anwendungsparameter des *PiKoder/PPM wRX* eingestellt und die einzelnen Servokanäle getestet werden. Hierbei ist beispielsweise an die Maximal- und Minimalwerte oder die Fail Safe Werte je Kanal zu denken.

Zur Durchführung der Parametrierung wird der ESP8266 abgezogen und der PiKoder/PPM eingesetzt. Bitte achten Sie hierbei auf die richtige Polung (s.a. das Bild auf der folgenden Seite). Ausserdem muss für die Kommunikation zwischen PC und PiKoder die TX- und die RX- Leitung des USB-UART-Umsetzers vertauscht werden (grünes und weißes Kabel), da wir ja jetzt quasi in die andere Richtung kommunizieren (s. Bild).



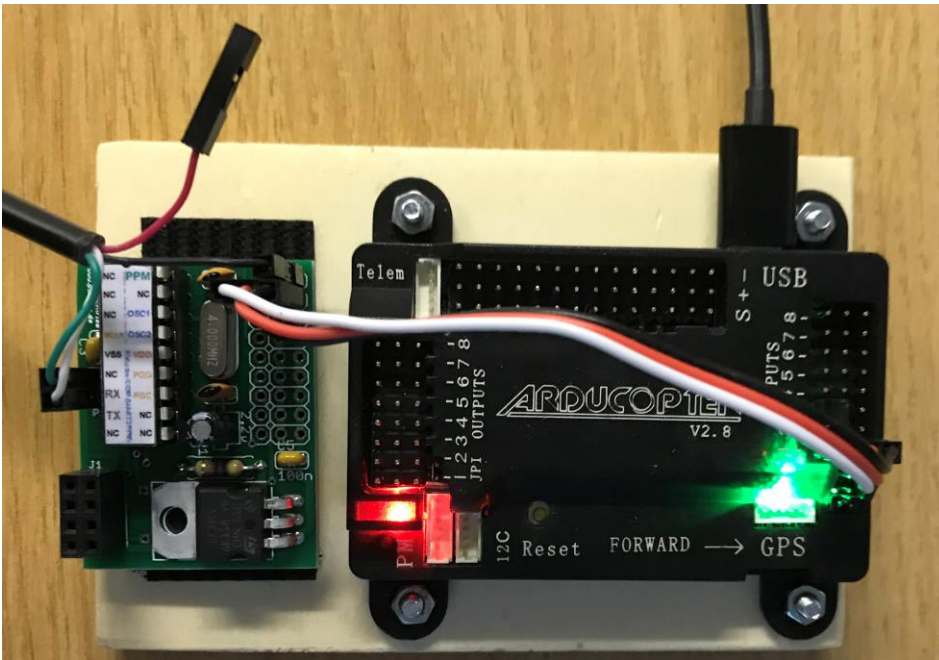
Die reine Parametrierung des PiKoder/PPM wRX kann nun entweder über die Terminal-Schnittstelle erfolgen oder über das [PCC \(PiKoder Control Center\)](#) mit seiner grafischen Bedienoberfläche. Der [PiKoder/PPM User's Guide](#) enthält eine detaillierte Beschreibung der verfügbaren Kommandos.

## Test des PPM Ausgangssignals mit einem FC

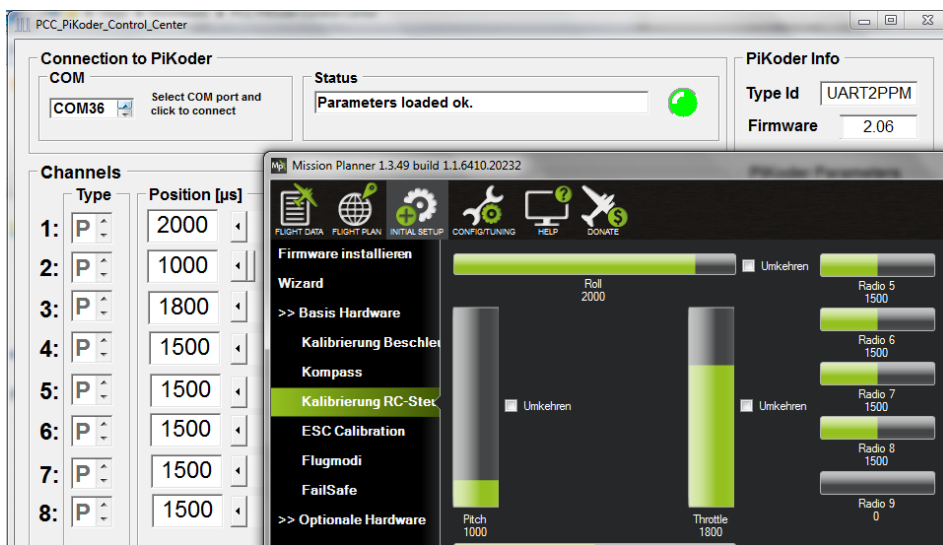
Das PPM Signal am Ausgang des PiKoder/PPM kann nicht direkt mit einem Servo überprüft werden, sondern Sie benötigen entweder ein Oszilloskop oder Sie verwenden die RC-Calibration-Funktion des Mission Planers, den Sie kostenfrei aus dem Internet herunterladen können.

Zum Test verbinden Sie den Flight Controller mit einem USB Ihres Computers und stellen mit einem dreipoligen Servoverlängerungskabel die Verbindung zwischen dem PPM-Ausgang des PiKoder/PPM wRX und dem PPM Eingang des FC her. Das dreipolige Kabel versorgt den PiKoder gleichzeitig mit Spannung. Bitte stellen Sie außerdem sicher, dass der APM für ein PPM-Eingangssignal konfiguriert ist (für den APM ist hierzu eine Kurzschlussbrücke zwischen Kanal 2 und 3 erforderlich)

Jetzt verbinden Sie Ihren PiKoder/PPM wRX mit einer weiteren USB-Schnittstelle Ihres Computers. **Bitte beachten Sie, dass das rote Kabel des USB-Adapters auf keinen Fall angeschlossen werden darf, um eine Zerstörung der USB Schnittstelle oder anderer Komponenten zu vermeiden (s. Bild auf der folgenden Seite).**



Nachdem Sie dann den APM mit dem Mission Planner und den PiKoder/PPM mit dem PCC verbunden haben, können Sie die Kanalwerte im PPC verändern und das Ergebnis im Mission Controller überprüfen.

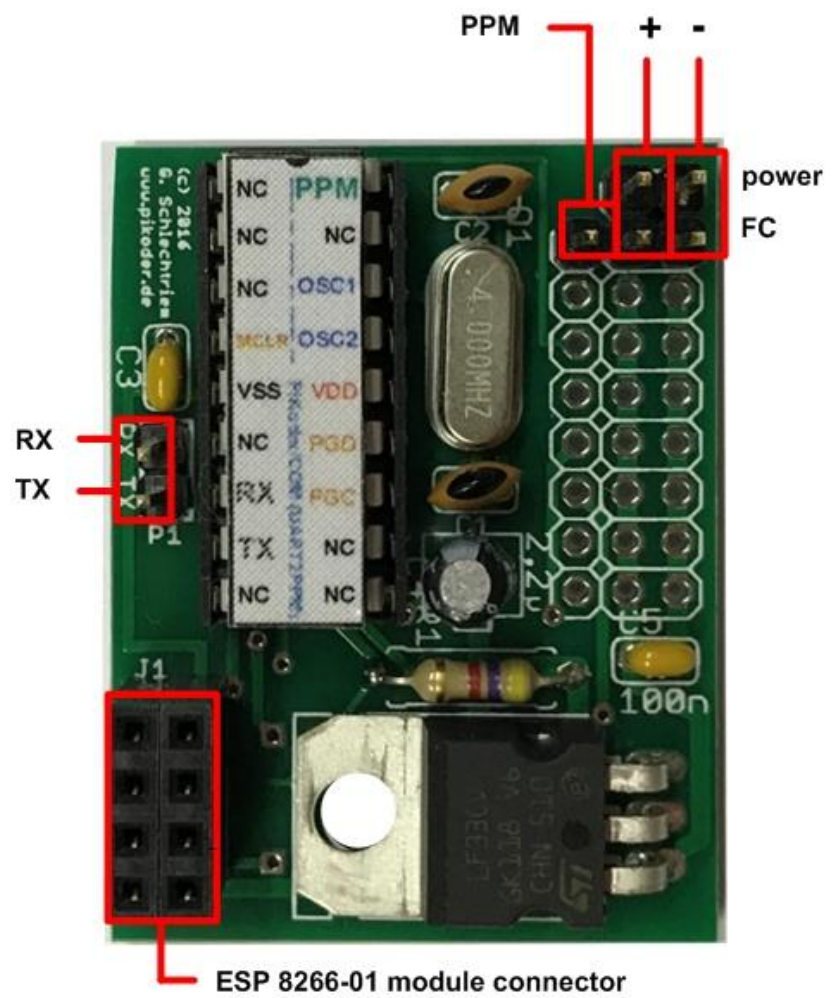


---

# A

---

## *Anschlussbelegung Empfänger*











---

# C

---

## **WiFi Protokoll**

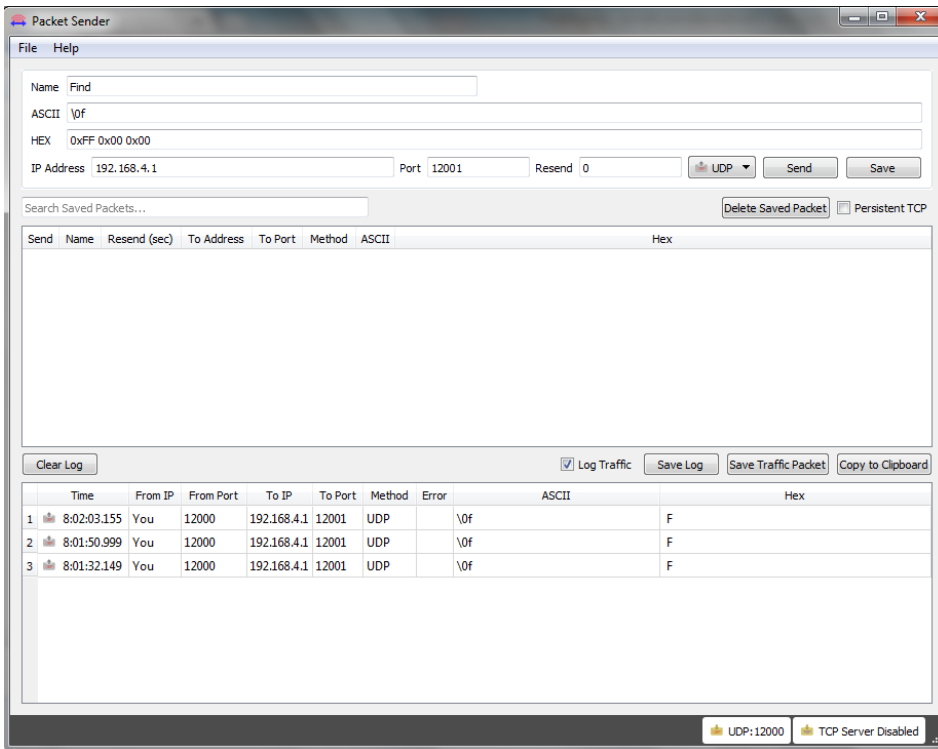
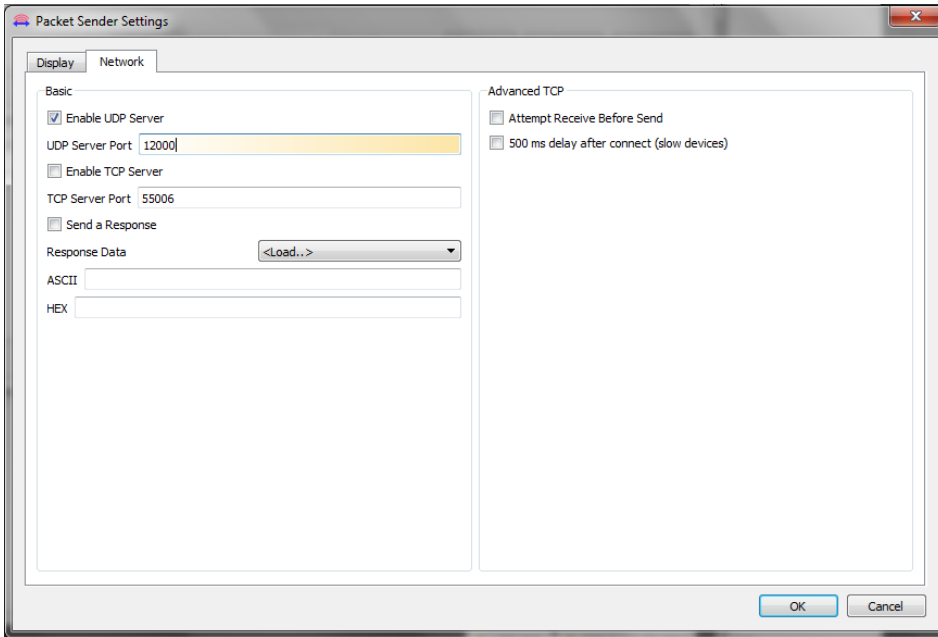
Für den Datenaustausch zwischen Sender und Empfänger wird das sogenannte UDP (User Datagram Protocol – daher der Name udpRC) verwendet, das einen effizienten Austausch geringer Datenmengen, wie bei einer Modellfernsteuerung erforderlich, unterstützt.

Dieser Abschnitt gibt Hinweise zum Aufbau der Kommunikation und zu weitergehenden Tests der WiFi-Kommunikation des PiKoder/PPM wRX. Alle Tests setzen voraus, dass der Empfänger mit Spannung versorgt wird (ev. angeschlossene Servos sind unkritisch) und die Programmierung des ESP8266 entsprechend der Vorgaben in der Bauanleitung erfolgt ist.

Die Tests setzen auf dem Programm *Packet Sender* von *Dan Nagle* auf, das Sie unter dem Link <https://packetsender.com/> downloaden können. Nach dem Laden und Installieren können Sie Ihren PC mit dem Access Point „PiKoder\_wRX“ verbinden. Achten Sie darauf, dass die Firewall das Netzwerk als Heimnetzwerk erkennt, damit keine Daten geblockt werden. Ihr Rechner sollte nun die IP 192.168.4.2 bekommen haben. (Eine genaue Beschreibung, wie Sie beispielsweise in Windows 7 die IP Adresse ermitteln können finden Sie u.a. im folgenden Beitrag: <http://www.heimnetzwerk-und-wlan-hilfe.com/ip-adresse-anzeigen-lassen.html>)

Starten Sie das Programm Packet Sender, stellen Sie unter Settings->Network den UDP-Server-Port auf 12000 und klicken die Checkbox „Enable UDP Server“ an (s. folgendes Bild). Bei der Rückkehr zum Hauptfenster sollte unten rechts dann im linken Ausgabefeld UDP:12000 stehen (s. zweites Bild auf der folgenden Seite). Falls nicht, müssen Sie die Software erneut starten.

Um einen Befehl an den Empfänger zu senden, ist im oberen Teil der Maske zunächst die IP Adresse des Empfängers einzugeben (192.168.4.1). Der vom Empfänger für die Kommunikation verwendete Port 12001 ist ebenfalls einzutragen.



Nun erfolgt die Eingabe des eigentlichen Befehls in der darüber liegenden Zeile im HEX-Format. Um die Servokanäle anzusteuern, verwenden Sie am einfachsten das miniSSC-Protokoll. Es hat folgendes Format:

0xFF 0xII 0xPP

Hierbei bedeuten:

0xFF: Startkennzeichen für einen neuen Befehl

0xII: II steht für die Kanalnummer beginnend mit 0

0xPP: PP steht für die Position des Servos (Bereich 0 – 0xFE)

Wollen Sie also beispielsweise den Wert für Kanal 1 auf die Minimalposition einstellen, dann geben Sie ein: 0xFF 0x00 0x00 (wie im vorherigen Bild dargestellt). Nach Betätigen des Sendeknopfes wird dieser Kanalwert ausgegeben. Mit der Eingabe 0xFF 0x00 0x7F bringen Sie das Servo wieder auf die Mittelstellung zurück.