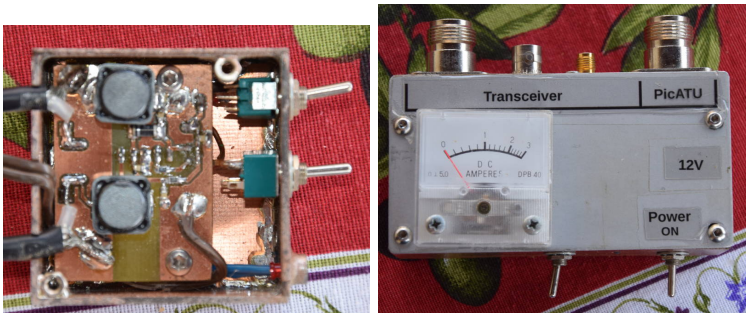
PicATU100/20 Remote Hardware

Andreas Lindenau DL4JAL

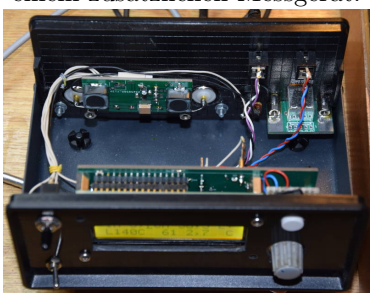
11. September 2022

Zusammenfassung

Für die Fernbedienung des PicATU100 und PicATU20 ist es notwendig eine Fernbedienung/Remote-BG aufzubauen. Im laufe der Jahre habe ich 3 verschiedene Fernsteuerbaugruppen konstruiert. Von ganz einfach bis komfortabel. Die Baugruppe hat die Aufgabe die 12V Stromversorgung in das Koaxkabel einzuspeisen. Mit kurzen 12V-Unterbrechungen (im mSek.-Bereich) werden die Funktionen im PicATU gestartet, Rückmeldungen erfolgen per CW über das Koaxkabel zum TRX.



Links die ganz einfache Variante mit Power EIN/AUS-, Remote-Schalter und LED für Rückmeldung während des Sendens. Rechts die gleiche Variante mit einem zusätzlichen Messgerät.



Zusätzlich zur Einspeisung der 12 Volt Stromversorgung habe ich eine Steuerung per RS232/TTL-Pegel konstruiert. Ganz komfortabel mit Display, Drehgeber und Taste wird der PicATU fern bedient. Zwischen Fernsteuerung und PicATU muss aber ein zusätzliches RS232-Kabel verlegt werden.

Inhaltsverzeichnis

1 Aufbau Fernbedienung, einfach **2**
1.1 Schaltbild Fernbedienung, einfach 3
1.2 Leiterplatte Fernbedienung, einfach 4

2 Aufbau Fernbedienung, einfach #2 **5**

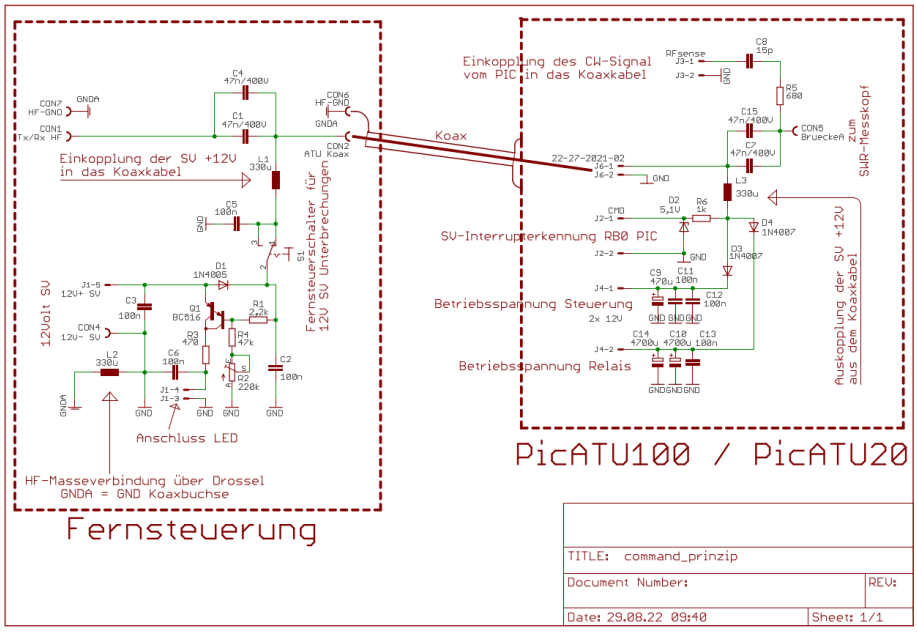
3 Aufbau Fernbedienung, komfortabel mit Display **6**
3.1 Leiterplatte SV 12V und Interrupt-Schalter 7
3.2 Schaltbild Fernbedienung komfortabel 8
3.3 Die Leiterplatte Fernbedienung komfortabel 9
3.3.1 Die ICs auf der Leiterplatte 10
3.3.2 Die Stecker auf der Platine 11
3.4 Inbetriebnahme der Baugruppe 13
3.4.0.0.1 Spannung an J3, ohne Verbindungskabel zum PicATU 14
3.4.0.0.2 Spannung an J3, mit Verbindungskabel zum PicATU 14
3.4.1 CAT-Verbindung zum TRX herstellen 14
3.4.2 CAT-Verbindung mit Monitor testen 15
3.5 Bilder der Fernbedienung 15

4 Schlusswort **18**

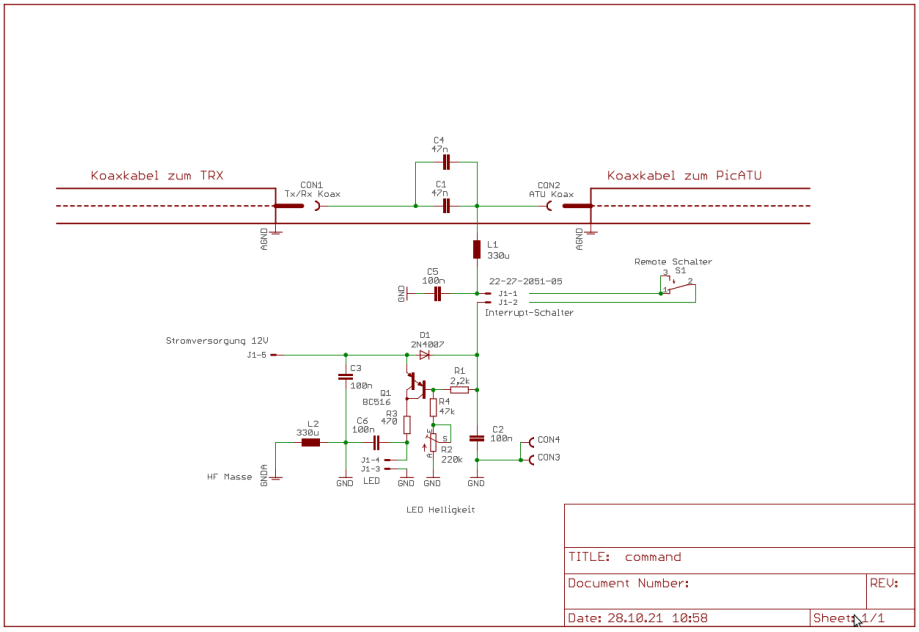
Kapitel 1

Aufbau Fernbedienung, einfach

Übersicht, Blockschaltbilder

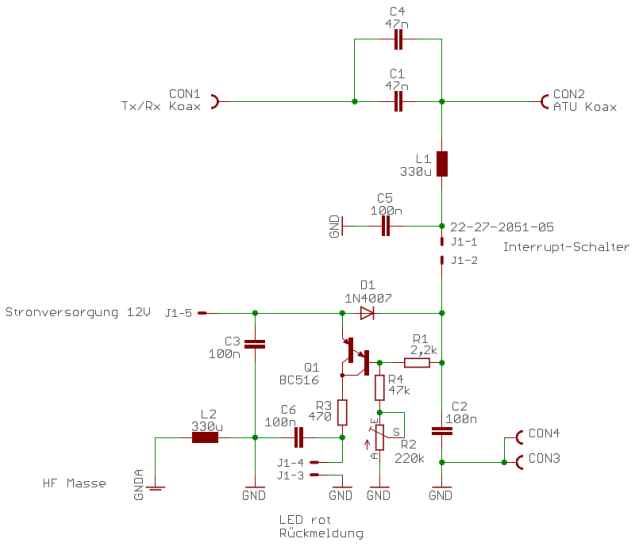


Eine Übersicht der Wirkungsweise der Fernsteuerung. Erste Aufgabe ist die 12V Stromversorgung in das Koaxkabel einzuspeisen. Die 2 Drosseln 330uH verhindern eine Beeinflussung des Sende und Empfangs-Signales. Mit dem Fernsteuerschalter werden kurze Unterbrechungen (mSek.) in der Stromversorgung erzeugt. Der Interrupteingang des PIC erfasst die Unterbrechung und reagiert darauf und beginnt seine Menüfunktionen per CW aufzuzählen. Wie das genau funktioniert beschreibe ich später.



Hier noch einmal die Remote-BG mit Prinzip-Schaltbild. Die Stromversorgung des PicATU wird hier eingespeist. Der „Remote-Schalter“ löst Funktionen im PicATU aus. Steht der TRX auf Empfang erfolgen die Rückmeldungen vom PicATU kommen per CW. Ist der TRX auf Sendung erfolgt die Rückmeldung des PicATU per LED. Blinken im 1 Sekunden-Rhythmus (z.B.: Match-Funktion wurde beendet).

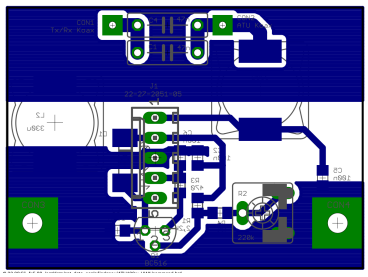
1.1 Schaltbild Fernbedienung, einfach



Der Spannungsabfall an der Diode ist je nach Strom unterschiedlich groß. Das wird für die Rückmeldung des PicATU während des Sendens genutzt. Ist der

Matchvorgang beendet werden im Sekundenrhythmus alle Relais EIN bzw. AUS geschaltet. Die LED rot beginnt zu blinken. Der Sendebetrieb kann beendet werden. GNDA ist HF-Masse am Koaxkabel. An GND und J1/5 werden die 12V Betriebsspannung eingespeist. Die beiden Drosseln L1, L2 330uH verhindert einen Rückfluss der HF in die Stromversorgung.

1.2 Leiterplatte Fernbedienung, einfach



Die kleine Leiterplatte habe ich in ein Gehäuse aus Leiterplattenmaterial eingeschraubt. Die Koaxkabel und das SV-Kabel habe ich direkt an die Platine gelötet. J1 habe ich nicht bestückt. Dadurch ist die Baugruppe sehr klein geworden.

Hier ein Bild von der kleinen Baugruppe. Der Blechwinkel wird unter den vorderen Standfuß des TRX geklemmt.

Kapitel 2

Aufbau Fernbedienung, einfach #2

Bei dieser Fernbedienung habe ich die gleiche Platine verwendet. Das Gehäuse ist auch aus Leiterplattenmaterial. Das Schaltbild ist das Gleiche. Der einzige Unterschied ist das zusätzliche Strommessgerät. Das Messinstrument ist am *Interrupt Schalter* mit eingeschleift.

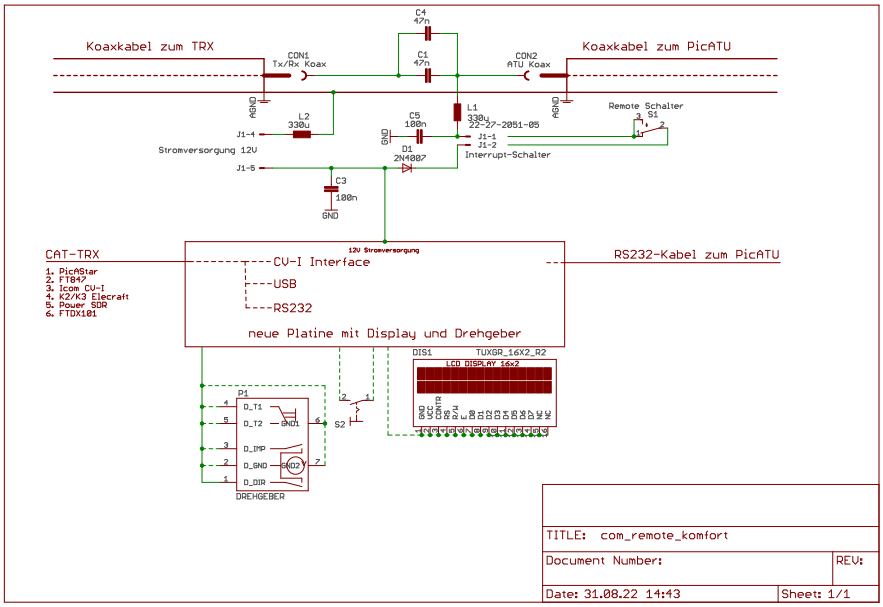
Kapitel 3

Aufbau Fernbedienung, komfortabel mit Display

Die einfache Fernbedienung und Stromversorgung des PicATU über das Koaxkabel reicht im Normalfall aus. Ein kleiner Nachteil ist das kurze Senden, damit der PicATU die VFO-Frequenz erkennt und die neue Match-Einstellung aktiviert.

Möchte man eine automatische Frequenz-Nachführung des PicATU mit der VFO-Frequenz des TRX, muss die komfortablere Fernbedienung aufgebaut werden. Zwischen Fernbedienung und PicATU muss noch ein RS232-Kabel verlegt werden.

Übersicht, Blockschaltbilder



31.08.22 14:45 f=1.07 /opt/data/opt_data_eagle/lindenau/PicATU_100W/com_remote_komfort.sch (Sheet: 1/1)

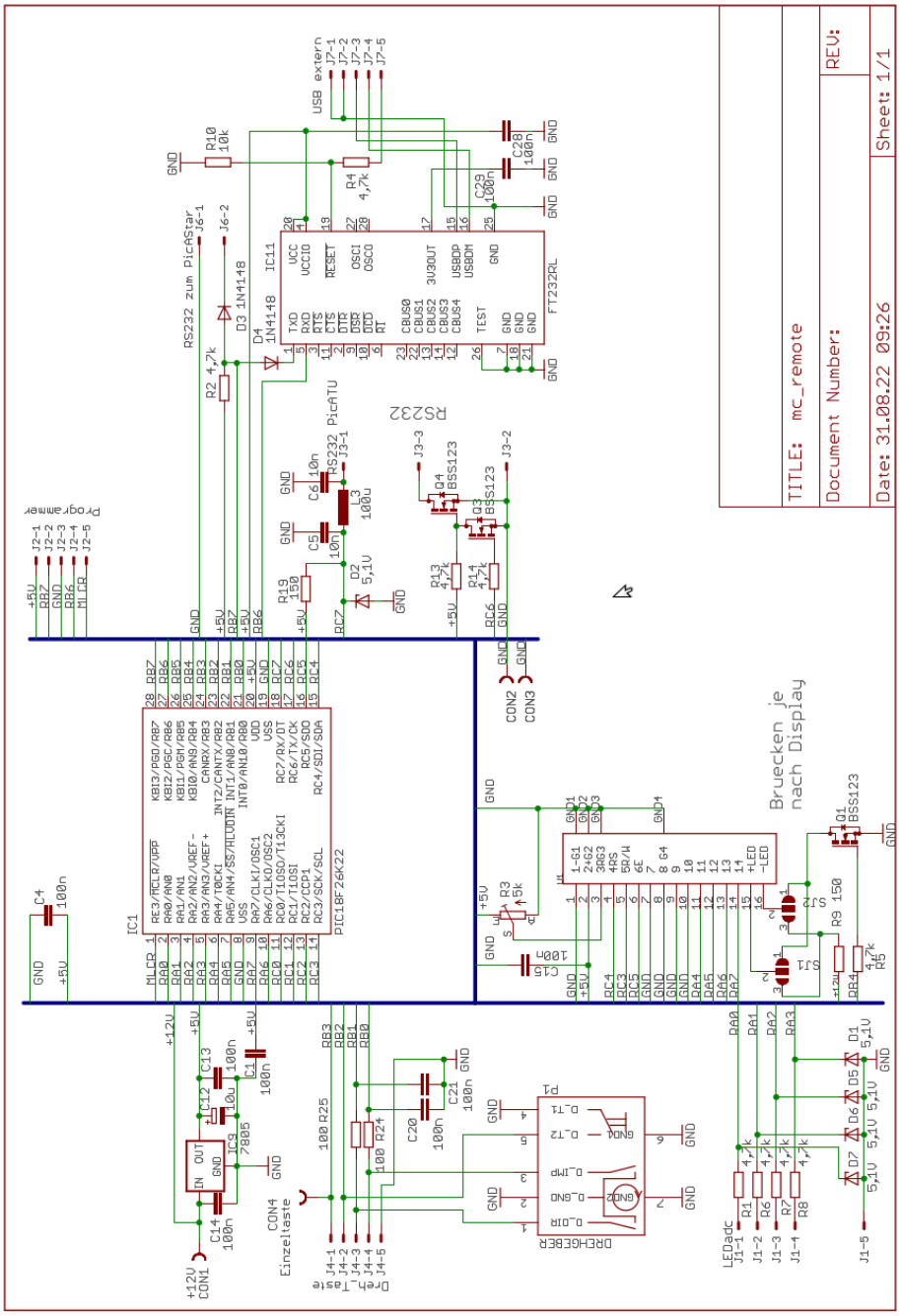
So sieht das Blockschaltbild dieser Baugruppe aus. Über die RS232-Verbindung zum PicATU wird bei einer Frequenzänderung sofort die neue Frequenz des TRX gesendet. Der PicATU hat so immer die aktuelle Match-Einstellung ohne dass ein Sendesignal notwendig ist. Die Fernbedienung verarbeitet die verschiedenen Datenformate der unterschiedlichen Transceiver zu einem einheitlichen Frequenz-Befehl.

Zusätzlich kann mit der Einzeltaste und Drehgeber der PicATU bedient werden. Der *Remote Schalter S1* ist zwar vorhanden, wird aber nicht benötigt.

3.1 Leiterplatte SV 12V und Interrupt-Schalter

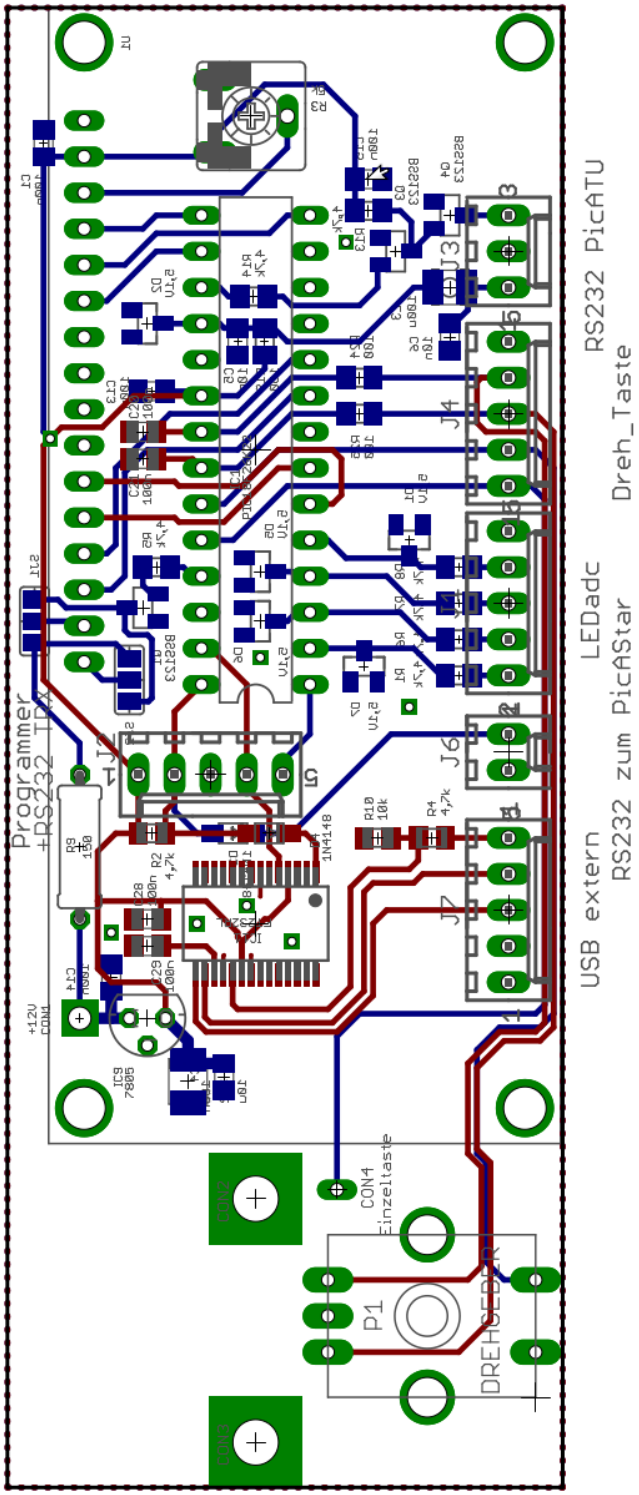
Für die Einspeisung der 12V in das Koaxkabel und die Remotesteuerung mit Schalter wird wieder die gleiche Leiterplatte der einfachen Fernbedienung verwendet. Die LED, Q1 und die Widerstände habe ich weg gelassen.

3.2 Schaltbild Fernbedienung komfortabel



In dieser Fernbedienung ist neben der Platine der Stromversorgung 12V noch eine Leiterplatte mit einem PIC-Mikrocontroller eingebaut.

3.3 Die Leiterplatte Fernbedienung komfortabel



3.3.1 Die ICs auf der Leiterplatte

Die Nummerierung der IC's geht nicht der Reihe nach, da das Schaltbild aus einer anderen Anwendung teilweise übernommen wurde.

IC1 PIC18F26K22

Der PIC18F26K22 ist der Mikrocontroller auf der Platine. Er steuert alles. Ich liste mal die Funktion der einzelnen Pins auf.

1. MCLR, Dieses ist für die Programmierung des Flash-Speichers mit einem Programmiergerät.
2. RA0, frei
3. RA1, frei
4. RA2, frei
5. RA3, frei
6. RA4, LCD Display
7. RA5, LCD Display
8. RA6, LCD Display
9. RA7, LCD Display
10. VSS, GND
11. RC0, frei
12. RC1, frei
13. RC2, frei
14. RC3, LCD Display
15. RC4, LCD Display
16. RC5, LCD Display
17. RC6, RS232#1 TX
18. RC7, RS232#1 RX
19. VSS, GND
20. VDD, +5V
21. RB0, Drehgeber Impulse
22. RB1, Drehgeber Richtung
23. RB2, Drehgeber Taste
24. RB3, Einzeltaste

25. RB4, frei
26. RB5, frei
27. RB6, PGC Programmer, RS232#2 TX
28. RB7, PGD Programmer, RS232#2 RX

Es sind einige Pin's noch frei. Ich habe aber bewusst den PIC18F26K22 eingesetzt da er zwei vollwertige RS232-Schnittstellen hat.

IC9 7805

Der Spannungsregler für die 5 Volt Betriebsspannung.

IC11 FT232RL

Das ist auch ein sehr hochwertiger IC. Der IC wandelt eine RS232-Schnittstelle im TTL Hardwareformat in eine echte USB-Schnittstelle um. Ich setze gern diesen IC ein da eine Treiberanpassung im PC bei den verschiedenen BS, wie Windows10 und Linux, automatisch erfolgt und fast immer ohne Problem klappt.

3.3.2 Die Stecker auf der Platine

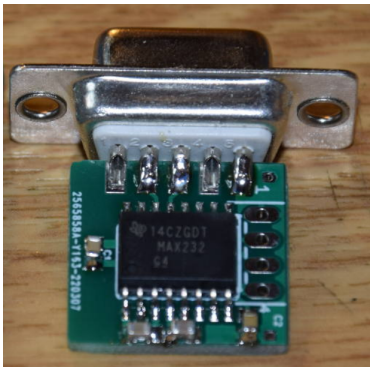
Stecker J1, LEDadc

Der Stecker J1 wird derzeit nicht benutzt. Die AD-Wandler wollte ich ursprünglich für Auswertungen der Betriebsspannung verwenden.

Stecker J2, Programmer + RS232 zum TRX

J2 ist vorgesehen für das Programmieren des PIC in der Leiterplatte. Der PIC braucht nicht aus der Fassung gehebelt werden.

Die zweite Verwendung von J2 ist der Anschluss einer echten RS232 für die Verbindung zum TRX, z.B.: FTDX101 oder FT847. Dazu ist ein kleine Leiterplatte vorgesehen, die direkt an die SUB-D9-Buchse gelötet ist. Auf der Platine ist ein SMD-MAX232 mit 4x 10uF aufgelötet. Das IC wandelt den 5V-TTL-Pegel in den RS232-Pegel um.



J2 als RS232_SUBD9

Pin1 SV 5 Volt

Pin2 RX RS232, Diode Katode

Pin3 GND

Pin4 TX RS232

Pin5 bleibt frei

Verbindung J2 zur RS232-Leiterplatte Pin's

J2 Pin1	5V			5V	RS232-LP, Pin4
J2 Pin2	RX	Katode-1N4148	1N4148-Anode	TX	RS232-LP, Pin3
J2 Pin3	GND			GND	RS232-LP, Pin1
J2 Pin4	TX			RX	RS232-LP, Pin2
J2 Pin5		bleibt frei			

Achtung die RX Leitung muss mit einer Diode 1N4148 entkoppelt werden.

J2 als Programmer

Pin1 SV 5 Volt

Pin2 PGD

Pin3 GND

Pin4 PGC

Pin5 MCLR/VPP

Stecker J3, RS232 zum PicATU

An J3 wird die Steckverbindung für das RS232-Verbindungskabel zum PicATU angeschlossen. Ich habe eine Stereo-3mm-Klinkenbuchse verwendet.

Pin1 TX-TTL

Pin2 GND

Pin3 RX-TTL

Stecker J4, Dreh_Taste

J4 ist für eine zusätzliche kleine Platine für Drehgeber und Einzeltaste. Hat der Drehgeber eine lange Achse kann auch der vorgesehene Platz, Bauteil P1, direkt auf der Platine genutzt werden. An CON4 wird die Einzeltaste angeschlossen. Ich habe die Einzeltaste auf eine kleine Lochrasterplatine gelötet.

Pin1 GND

Pin2 Einzeltaste

Pin3 Drehgeber DIR (Drehrichtung)

Pin4 Drehgeber Impulse

Pin5 Drehgeber Taste

Stecker J6, CV-I oder PicAStar-RS232-TTL

J6 ist nur 2-polig und wird für die TTL-Signal vom Icom-CV-I verwendet oder von meinem PicAStar.

Pin1 GND

Pin2 RX-RS232-TTL

Auch hier habe ich eine Stereo-3mm-Klinkenbuchse in die Rückwand eingeschraubt.

Stecker J7, USB extern

Für eine Verbindung zu einem PC und der PC-Software „PowerSDR“ habe ich eine USB-Buchse vorgesehen. IC11 FT232RL ist ein USB-RS232-Wandler. Die Steckerbelegung von J7 ist wie folgt.

Pin1 GND

Pin2 GND

Pin3 USBDP

Pin4 USBDM

Pin5 5V von der USB-Schnittstelle

Verbindung J7 zur Leiterplatte USB extern

J7 Pin1	bleibt frei	USB ext.	Pin1
J7 Pin2	GND	USB ext.	Pin2
J7 Pin3	USBDP	USB ext.	Pin3
J7 Pin4	USBDM	USB ext.	Pin4
J7 Pin5	5V vom PC	USB ext.	Pin5

CON1, +12V Betriebsspannung

Die Betriebsspannung erhält die Baugruppe von der SV-Platine direkt vom *Schalter-PowerON.*

CON4, Einzeltaste

Hier habe ich einen Pin der Einzeltaste angelötet. Den Drehgeber mit langer Achse habe ich direkt in die Platinen eingelötet und die Einzeltaste auf eine Lochraster-Zwischenplatine.

3.4 Inbetriebnahme der Baugruppe

Nach *PowerON* kommt der Initialisierungs-Text auf dem Display. Anschließend steht in Zeile 1 „Ready!“. Alle 5 Sekunden wird Datensatz wird zum PicATU gesendet.

3.4.0.0.1 Spannung an J3, ohne Verbindungskabel zum PicATU

Pin1 TX-TTL, 0,5 Volt

Pin2 GND

Pin3 RX-TTL, 4,8 Volt

3.4.0.0.2 Spannung an J3, mit Verbindungskabel zum PicATU

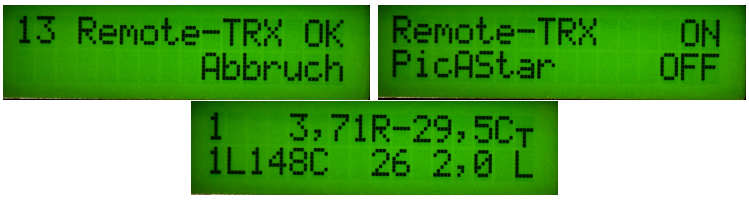
Pin1 TX-TTL, 3,6 Volt

Pin2 GND

Pin3 RX-TTL, 4,8 Volt

Die 3,6 Volt an Pin1 kommen vom PicATU zurück.

Am PicATU100/20 müssen wir jetzt nur noch die Remote RS232 aktivieren.



Im Display vom PicATU100/20 ändert sich die Frequenzangabe von „3,71M“ in „3,71R“, bedeutet Remote ist jetzt aktiv. Die Frequenzmessung im PicATU hat jetzt nur noch die Aufgabe die Richtigkeit der Frequenz am Anfang jeder *Matchfunktion* zu kontrollieren.

Mit dem Drehgeber an der Fernbedienung können wir jetzt die neue Frequenz zum PicATU senden. Der PicATU holt sich entsprechend der gesendeten Frequenz die Match-Einstellung aus dem Eeprom, ohne das wir ein Sendesignal anlegen müssen.

3.4.1 CAT-Verbindung zum TRX herstellen

Die neue Arbeitsfrequenz mit dem Drehgeber einstellen funktioniert ganz gut, aber noch besser wäre es eine CAT-Verbindung zum Transceiver herzustellen und die VFO-Frequenz des Transceivers zur Fernbedienung übermitteln.

Folgende HW-Anschlüsse habe ich vorgesehen:

J3, 5V RX-TTL Diesen Anschluss nutze ich für meinen PicAStar. Aber auch die Icom Transceiver mit einem CI-V-Anschluss.

J6, RS232 Eine normale RS232 mit einer zusätzlichen kleinen Platine mit RS232-IC (z.B.:MAX232).

J7, USB extern Auf der MC-Leiterplatte ist ein IC FT232RL aufgelötet für eine USB-Verbindung zu PC. Die USB-Buchse muss extra montiert werden.

Die 3 Stecker J3, J6 und J7 arbeiten parallel. Es darf jeweils nur eine Möglichkeit benutzt werden, sonst gibt es „Datensalat“.

In der Software habe ich folgende TRX berücksichtigt:

CAT-PicAStar(DL4JAL), J3 Der CAT-Anschluss ist ein Ausgang mit 5V TTL-Pegel. Bei jedem VFO-Frequenzwechsel wird sofort die neue Frequenz ausgegeben.

CAT-FT847, J6 Den FT847 habe ich auch angepasst. Das ist aber schon einige Jahre her. Ob das noch zuverlässig funktioniert weiß ich nicht.

CAT-Icom CI-V Remote, J3 Der Icom TRX funktioniert auch an der Fernsteuerung.

CAT-K2/K3 Elecraft, J6 Über die RS232. Baudrate ist fest eingestellt 4800 Baud.

CAT-PC PowerSDR, J7 Die Windows-Software „PowerSDR“ funktioniert auch mit der CAT-Anbindung. Entweder über die einfache RS232 oder über USB.

CAT-FTDX101 Yaesu, J6 Über die RS232. FTDX Einstellung 9600Baud.

Bei Elecraft ist die default-Baudrate 4800 Baud. Alle anderen TRX arbeiten mit 9600 Baud.

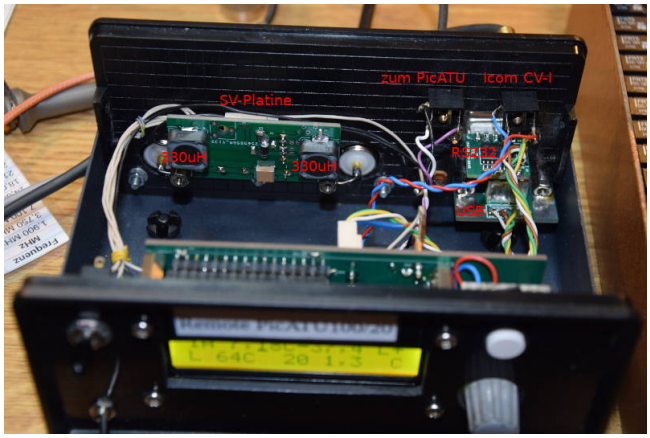
3.4.2 CAT-Verbindung mit Monitor testen

Zum Testen ob Daten über die Schnittstellen vom TRX zur Fernbedienung kommen habe ich in der Fernbedienung 2 Monitore programmiert.

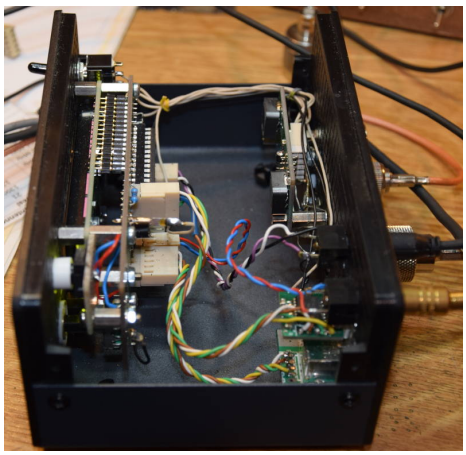
Monitor mit HEX-Ausgabe Jedes Byte was ankommt wird als hexadezimale Zahl angezeigt.

Monitor mit Ascii-Ausgabe Jedes Byte was ankommt wird als Char in einem String dargestellt. Der Buchstabe *A* ist zum Beispiel die HEX-Zahl 41 Hex.

3.5 Bilder der Fernbedienung

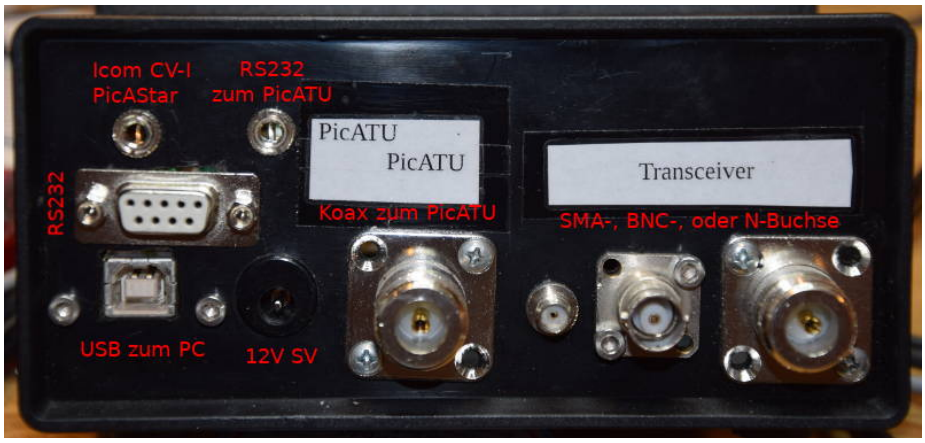


Blick in das Gehäuse. Jetzt habe ich alle HW-Schnittstellen angeschlossen und auch alle getestet.



Links die neue Leiterplatte für die Bedienung und Anzeige. Rechts hinten die SV Platine mit der Einspeisung der 12V Betriebsspannung und Anschluss für Interruptschalter.

Blick auf die neue Leiterplatte mit PIC18F26K22 und den Steckerbuchsen. Rechts oben der Schalter *PowerON* und darunter der Schalter für *12V-Interrupt*.



Ich habe noch ein Foto von hinten. Alle Anschlüsse habe ich beschriftet.

Kapitel 4

Schlusswort

Dieses Projekt darf nicht kommerziell vermarktet oder genutzt werden. Alle Rechte liegen bei DL4JAL (Andreas Lindenau). Ich wünsche viel Spaß beim Basteln.

vy 73 Andreas DL4JAL

✉ DL4JAL@t-online.de
☞ www.dl4jal.de