

Beschreibung der USB-Baugruppe mit  
Soundkarte für die SDR-Console  
(Ansteuerung „Adalm Pluto“ über Ethernet mit  
*Mode SSB und Mode CW*)

(c) Andreas, Lindenau DL4JAL

12. Mai 2021

# Inhaltsverzeichnis

<b>1</b>	<b>Vorwort</b>	<b>3</b>
<b>2</b>	<b>Hardware</b>	<b>4</b>
2.1	Leiterplatte USB-Platine . . . . .	4
2.1.1	Die ICs auf der USB-Platine . . . . .	4
2.1.1.1	IC4 UA78M33CDCY . . . . .	4
2.1.1.2	U2 TUSB2046B . . . . .	4
2.1.1.3	IC3 FT232RL . . . . .	4
2.1.1.4	U3 PC2906C . . . . .	4
2.1.2	Layout-Fehler Beseitigung auf der USB-Platine . . . . .	4
2.1.3	Aufbau der USB-Platine . . . . .	5
2.1.3.1	Schritt 1, USB-Buchse und Spannungsregler 3,3 V . . . . .	5
2.1.3.2	Schritt 2, TUBS2046B . . . . .	6
2.1.3.3	Schritt 3, FT232RL . . . . .	7
2.1.3.4	letzter Schritt, PCM2906C . . . . .	8
2.1.4	Die Steckverbindungen der USB-Platine . . . . .	9
2.1.4.1	X1, USB Printbuchse Typ B . . . . .	9
2.1.4.2	J1, Audio Output . . . . .	9
2.1.4.3	J2, Audio Input . . . . .	9
2.1.4.4	J3, PTT über RS232 . . . . .	9
2.1.4.5	J4, Stromversorgung . . . . .	9
2.1.5	USB-Platine Abstandsbolzen . . . . .	9
2.2	Zusatz-Platine . . . . .	10
2.2.1	Die ICs der Zusatz-Platine . . . . .	10
2.2.2	Layout-Fehler auf der Zusatzplatine-Platine . . . . .	10
2.2.2.1	IC1 PIC18F26K22 oder PIC18F25K22 . . . . .	11
2.2.2.2	IC2 TPA4861 . . . . .	11
2.2.2.3	IC3A AD823 . . . . .	11
2.2.2.4	IC3B AD823 . . . . .	11
2.2.3	Die Steckverbindungen der Zusatz-Platine . . . . .	11
2.2.4	Steckverbindungen Zusatz-Platine und USB-Platine . . . . .	13
2.2.5	Die Einstellregler der Zusatz-Platine . . . . .	16
2.3	Display-Platine . . . . .	17
2.3.1	LCD-Anzeige, DOG104SPI . . . . .	17
2.3.2	Drehgeber, Taste und LED . . . . .	18
2.3.3	Verbindungen zur Zusatz-Platine . . . . .	18

<b>3</b>	<b>Firmware, Software</b>	<b>20</b>
3.1	Bedienung der Baugruppe	21
3.1.1	Drehgeber	21
3.1.2	Einzeltaster	21
3.1.3	Mode SSB	21
3.1.4	Mode CW	21
3.2	Menu-Funktionen	22
3.2.1	[1] MTon, zum Frq. abstimmen	22
3.2.2	[2] SSB/CW, Mode umschalten	22
3.2.3	[3] P.x5, 5 Punkte RX–TX	22
3.2.4	[4] P.x5, 5 Punkte TX–RX	22
3.2.5	[5] SETUP, Zugang zum SETUP	23
3.3	SETUP-Funktionen	23
3.3.1	[1] Abbruch, SETUP Abbrechen	23
3.3.2	[2] MTon, Mithoerton einstellen	23
3.3.3	[3] Timer, RX–TX einstellen	23
3.3.4	[4] Tim.SSB, TX–RX einstellen	24
3.3.5	[5] TimerCW, TX–RX einstellen	25
3.3.6	[6] Keyer, Mode einstellen	25
3.3.7	[7] Keyer, Punktsp einstellen	25
3.3.8	[8] Keyer, Strichsp einstellen	26
3.3.9	[9] Display, drehen 180°	27
3.4	Einstellungen in der SDR-Console	27
<b>4</b>	<b>Schlusswort</b>	<b>29</b>

# Kapitel 1

## Vorwort

Dieses Projekt darf nicht kommerziell vermarktet oder genutzt werden. Alle Rechte liegen bei DL4JAL (Andreas Lindenau).

# Kapitel 2

## Hardware

### 2.1 Leiterplatte USB-Platine

#### 2.1.1 Die ICs auf der USB-Platine

##### 2.1.1.1 IC4 UA78M33CDCY

Die Stromversorgung erfolgt über die USB-Verbindung zum PC. Der PC liefert 5 Volt. IC4 ist ein Spannungsregler 3,3 Volt. Er stellt die Spannung 3,3 Volt für alle Platinen bereit.

##### 2.1.1.2 U2 TUSB2046B

U2 ist ein USB-HUB. U2 kann bis zu 4 USB ICs ansteuern. In unserem Fall sind 2 USB ICs am TUSB2046B angeschlossen.

##### 2.1.1.3 IC3 FT232RL

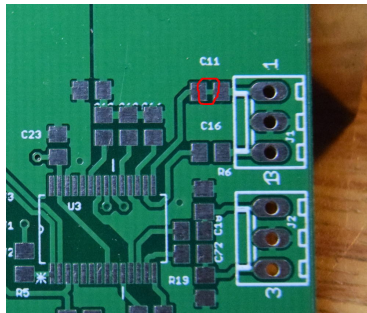
Als erstes ist der FT232RL am USB-HUB angeschlossen. Für die Umsetzung der PTT brauchen wir einen RS232 Eingang am PC. Der FT232RL stellt im PC unter Windows eine COM-Schnittstelle zur Verfügung. Genutzt werden können verschieden Eingänge am FT232RL. Ich nutze den CTS-Eingang und habe die entsprechende SMD Lötbrücke gesetzt.

##### 2.1.1.4 U3 PC2906C

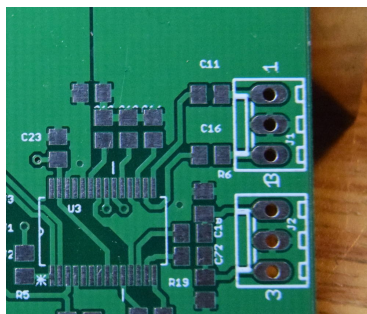
Als zweiter IC ist der PCM2906C am USB-HUB angeschlossen. Der PCM2906 ist ein Audio-Codec Stereo IC. Er bildet die Devices einer Soundkarte im PC. Den Input-Device und den Output-Device können wir am PC auswählen.

#### 2.1.2 Layout-Fehler Beseitigung auf der USB-Platine

Leider hat sich ein Fehler auf der Platine eingeschlichen, den ich jetzt erst bemerkt habe.



Beide Pads von C11 sind mit irrtümlicherweise einem Leiterzug verbunden.  
Dieser Leiterzug muss durchtrennt werden.



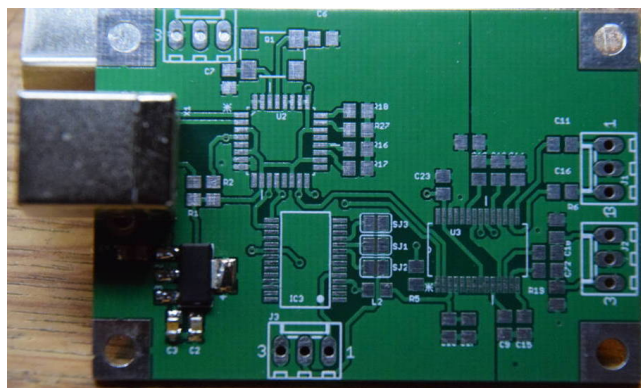
Im Bild habe ich den Leiterzug entfernt.

### 2.1.3 Aufbau der USB-Platine

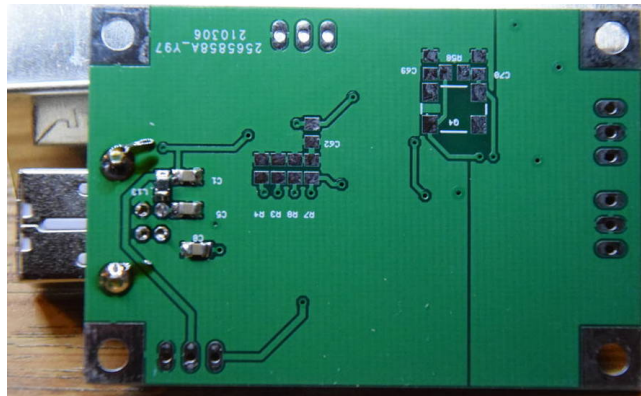
Zuerst bestücken wir die USB-Platine. Die Platine bestücken wir der Reihe nach und testen nach jedem Schritt.

#### 2.1.3.1 Schritt 1, USB-Buchse und Spannungsregler 3,3 V

Als erstes löten wir die USB-Buchse ein und den Spannungsregler IC4, mit allen dazu gehörigen Bauelementen.



Leiterplatte von oben. Links die USB-Buchse B.

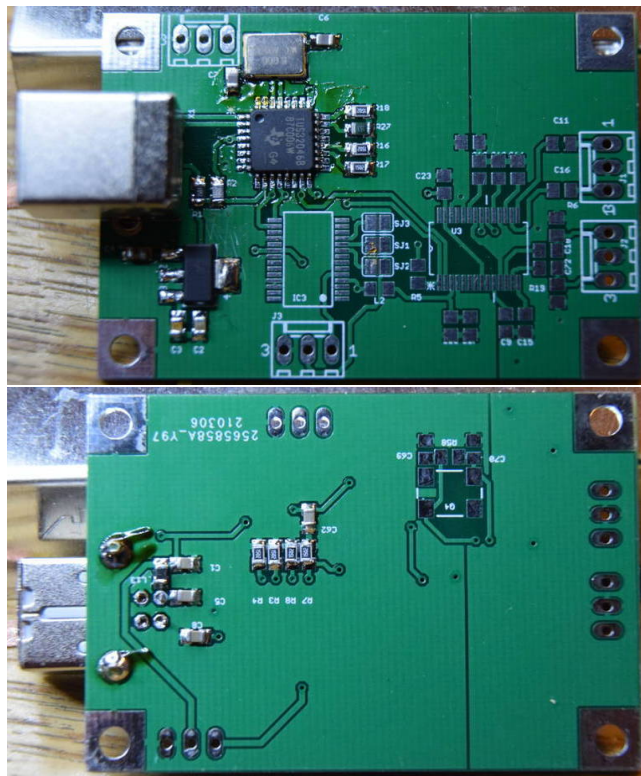


Die Bauelemente unter der Platine.

**Test nach Schritt 1** Die USB-Buchse mit einem USB-Kabel mit dem PC-Verbinden. Am Ausgang von Spannungsregler (Pin3) messen wir die Spannung. Es müssen 3,3 Volt anliegen.

### 2.1.3.2 Schritt 2, TUBS2046B

Wir löten IC U2 TUBS2046B ein mit allen seinen Bauelementen Rs und Cs und dem Quarz 6 MHz.



Wieder Platine von oben und von unten.

## Test nach Schritt 2

```
dl4jal@funkraum:~$ lsusb
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 005: ID 093a:2510 Pixart Imaging, Inc. Optical Mouse
Bus 001 Device 004: ID 04e8:4001 Samsung Electronics Co., Ltd
Bus 001 Device 003: ID 03f0:1e17 Hewlett-Packard LaserJet 1160 series
Bus 001 Device 002: ID 04d8:900a Microchip Technology, Inc. PICKit3
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
dl4jal@funkraum:~$ lsusb
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 005: ID 093a:2510 Pixart Imaging, Inc. Optical Mouse
Bus 001 Device 004: ID 04e8:4001 Samsung Electronics Co., Ltd
Bus 001 Device 003: ID 03f0:1e17 Hewlett-Packard LaserJet 1160 series
Bus 001 Device 002: ID 04d8:900a Microchip Technology, Inc. PICKit3
Bus 001 Device 006: ID 0451:2046 Texas Instruments, Inc. TUSB2046 Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
dl4jal@funkraum:~$
```

Am einfachsten geht der Test mit Linux. Mit „lsusb“ listen wir auf, was an USB-Devices vorhanden ist. Zuerst einmal ohne unsere Baugruppe. Unter Windows muss man im Gerätemanager suchen, ob etwas neues hinzu gekommen ist.

**Linux Foundation 3.0 root hub** Das ist intern im PC der Hub für USB 3.0

**Optical Mouse** Das ist meine PC-Maus

**Samsung Electronics** Das ist meine externe Backup SSD 500 GByte

**LaserJet 1160** Das ist mein Laserdrucker

**PICKit3** Programmier für mein PICs

**Linux Foundation 2.0 root hub** Das ist intern im PC der Hub für USB 2.0

Anschließend habe ich meine Platine angesteckt und siehe da jetzt ist noch eine Zeile hinzu gekommen. Unter Windows muss man im Gerätemanager suchen, ob etwas neues hinzu gekommen ist.

**Texas Instruments TUSB2046 HUB** Das ist der aufgelötet IC auf der Platine

Das sieht doch schon mal gut aus.

### 2.1.3.3 Schritt 3, FT232RL

Jetzt ist der FT232RL mit Auflöten an der Reihe. Zusätzlich wird noch die SMD-Lötbrücke SJ3 (CTS) geschlossen.

## Test nach Schritt 3

```
dl4jal@funkraum:~$ lsusb
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 005: ID 093a:2510 Pixart Imaging, Inc. Optical Mouse
Bus 001 Device 004: ID 04e8:4001 Samsung Electronics Co., Ltd
Bus 001 Device 003: ID 03f0:1e17 Hewlett-Packard LaserJet 1160 series
Bus 001 Device 002: ID 04d8:900a Microchip Technology, Inc. PICKit3
Bus 001 Device 008: ID 0403:6001 Future Technology Devices International, Ltd FT232 USB-Serial (UART) IC
Bus 001 Device 007: ID 0451:2046 Texas Instruments, Inc. TUSB2046 Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
dl4jal@funkraum:~$
```

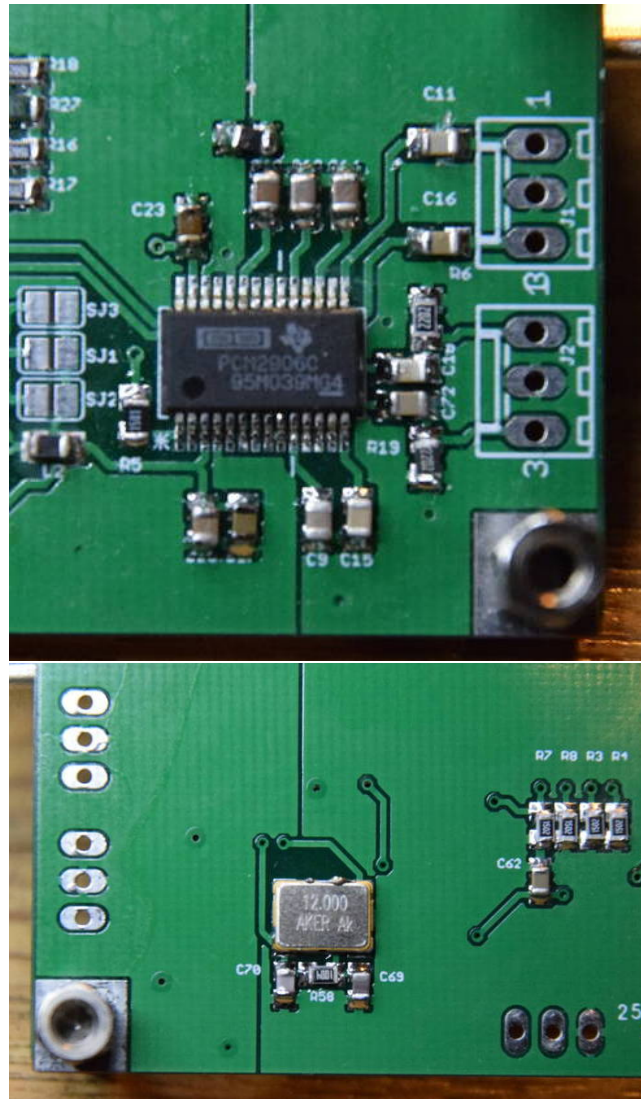


Wieder ist eine Zeile hinzu gekommen beim Befehl „lsusb“.

**Ltd FT232 USB Serial** Das ist der aufgelötete FT232RL

#### 2.1.3.4 letzter Schritt, PCM2906C

Jetzt löten wir noch den PCM2906 auf und die restlichen Bauelemente. Unter der Platine den Quarz 12 MHz mit Bauelementen.



Test nach letzten Schritt

```
dl4jal@funkraum:~$ lsusb
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 005: ID 093a:2510 Pixart Imaging, Inc. Optical Mouse
Bus 001 Device 004: ID 04e8:4001 Samsung Electronics Co., Ltd
Bus 001 Device 003: ID 03f0:1e17 Hewlett-Packard LaserJet 1160 series
Bus 001 Device 002: ID 04d8:900a Microchip Technology, Inc. PICKit3
Bus 001 Device 011: ID 08bb:29c6 Texas Instruments PCM2906C Audio CODEC
Bus 001 Device 010: ID 0403:6001 Future Technology Devices International, Ltd FT232 USB-Serial (UART) IC
Bus 001 Device 009: ID 0451:2046 Texas Instruments, Inc. TUSB2046 Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
dl4jal@funkraum:~$
```

Wieder ist eine Zeile hinzu gekommen. Unter Windows muss man im Gerätemanager suchen, ob etwas neues hinzu gekommen ist.

**PCM2906C Audio CODEC** Das ist unser Soundkarten-IC auf der Platine.

Die digitalen Komponenten scheinen zu funktionieren. Mehr Tests sind vorerst nicht möglich.

## 2.1.4 Die Steckverbindungen der USB-Platine

### 2.1.4.1 X1, USB Printbuchse Typ B

Dazu gibt es nicht viel zu erklären. An diese Buchse wird das Verbindungskabel zum PC angeschlossen.

### 2.1.4.2 J1, Audio Output

J1 ist ein 3 poliger Verbindungsstecker zur Zusatzplatine.

### 2.1.4.3 J2, Audio Input

J2 ist ein 3 poliger Verbindungsstecker zur Zusatzplatine. Es wird nur Pin 2 und Pin 1 (linker Kanal) benutzt.

### 2.1.4.4 J3, PTT über RS232

J3 ist ein 3 poliger Verbindungsstecker zur Zusatzplatine. Es wird nur Pin 2 und Pin 3 benutzt. Ich nutze „CTS“ als PTT-Funktion.

### 2.1.4.5 J4, Stromversorgung

J4 ist ein 3 poliger Verbindungsstecker zur Zusatzplatine.

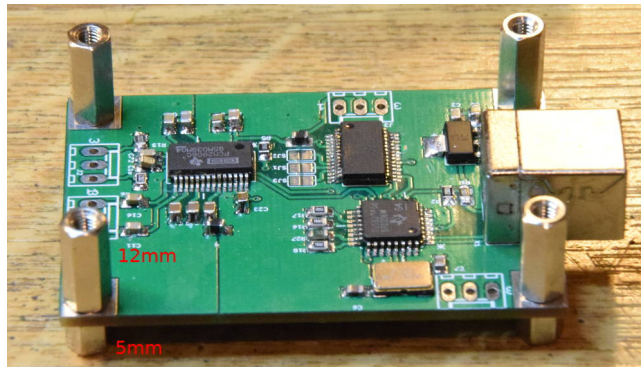
**Pin 1** GND

**Pin 2** 5 Volt vom USB-Anschluss

**Pin 3** 3,3 Volt vom Spannungsregler

## 2.1.5 USB-Platine Abstandsbolzen

An die 4 Befestigungslöcher schrauben wir Abstandsbolzen. Unten welche mit 5 mm Länge und oben welche mit 12 mm Länge.

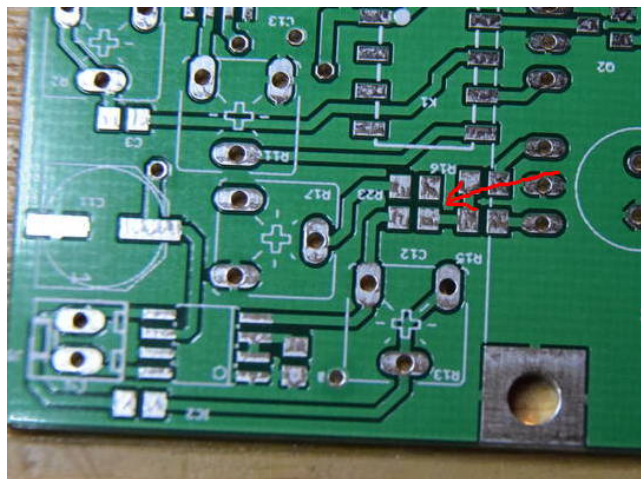


## 2.2 Zusatz-Platine

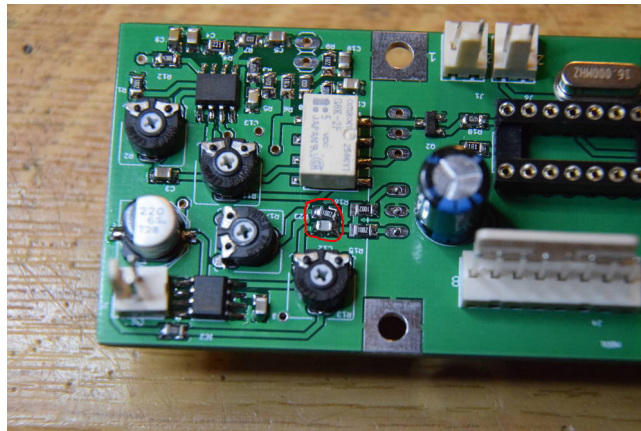
### 2.2.1 Die ICs der Zusatz-Platine

### 2.2.2 Layout-Fehler auf der Zusatzplatine-Platine

Leider hat sich auch hier ein Fehler auf der Platine eingeschlichen, den ich jetzt erst bemerkt habe.



Da wo der rote Pfeil hin zeigt hängt das Pad des Widerstandes R23, 10k in der Luft.



Ich habe den R23 etwas schräg eingelötet, so das das Pad von R23 mit dem Pad von C21 verbunden ist. Fehler ist behoben.

#### 2.2.2.1 IC1 PIC18F26K22 oder PIC18F25K22

Eigentlich reicht der PIC18F25K22 völlig aus für unseren Zweck. Ich habe den PIC18F26K22 eingesetzt, da er vorhanden war. Beide sind in dieser Anwendung zueinander kompatibel. Die Stromversorgung erfolgt mit den 3,3 Volt der USB-Platine. Der PIC steuert alle Vorgänge und ist mit der Displayanzeige verbunden.

#### 2.2.2.2 IC2 TPA4861

Der TPA4861 ist ein Audioverstärker, der auch mit geringer Versorgungsspannung noch arbeitet. Die Ausgangsleistung beträgt 1 Watt.

#### 2.2.2.3 IC3A AD823

IC3 ist ein Doppel-Operationsverstärker. IC3A OPV ist ein Impedanzwandler mit Verstärkungsfaktor 1. Er soll das sehr hochohmige DAC-Wandler-Signal aus dem PIC18F25K22 niederohmig machen.

#### 2.2.2.4 IC3B AD823

Der zweite OPV dient als Mikrofon Verstärker. Die Verstärkung wird mit dem Einstellregler R2, 22kOhm eingestellt.

### 2.2.3 Die Steckverbindungen der Zusatz-Platine

#### J1 Mike

Die Steckbuchse ist 2 polig.

PIN	Beschreibung
Pin 1	Mikrofon
Pin 2	GND-Masse

## J2 Programmer/PTTextern

Die Steckbuchse ist 5 polig. Diese Buchse ist zum Programmieren des PIC18F25K22 in der Schaltung.

Leiterplatte J2 Pin-Nummer	Leiterplatte J2 Beschreibung	PicKit3 Pin-Nummer	PicKit3 Beschreibung
PIN1	Daten PGC	PIN5	PGC (ICSPCLK)
PIN2	Daten PGD	PIN4	PGD (ICSPDAT)
PIN2	PTT extern		
PIN3	GND	PIN3	Vss (ground)
PIN4	VCC 3,3 Volt	PIN2	Vdd Target
PIN5	MCLR VPP	PIN1	/MCLR /Vpp

**Achtung!! die Nummern der Pins von „J2“ auf der Leiterplatte und die Nummern der Pins am PicKit3 sind gegenläufig . Meine Steckerbelegung war früher anders.**

Ab FW 1.02 ist PB7 vom PIC als zusätzliche Ausgabe für externe PTT-Funktionen gedacht. HIGH=PTT ein und LOW=PTT aus. Ideal ist diese Lösung nicht. Über die PC-SW „SDR-Console“ wird nicht überwacht, ob der Pluto auch wirklich auf SENDEN umgeschaltet hat. Die „PTT extern“ wird also *blind* ein- und ausgeschaltet.

## J3 Drehgeber

Diese Beschreibung ist identisch mit der Beschreibung der LCD-Platine.

Display-Platine	Zusatz-Platine	Beschreibung
Pin 1	Pin 1	GND-Masse
Pin 2	Pin 2	Drehgeber Impulse
Pin 3	Pin 3	Drehgeber Drehrichtung
Pin 4	Pin 4	Drehgeber Taste
Pin 5	Pin 5	Einzeltaste

## J4 LCD

Diese Beschreibung ist identisch mit der Beschreibung der LCD-Platine.

Display-Platine	Zusatz-Platine	Beschreibung
Pin 1	Pin 1	LCD CS
Pin 2	Pin 2	LCD SCL
Pin 3	Pin 3	LCD SID
Pin 4	Pin 4	LCD Reset
Pin 5	Pin 5	LCD 3,3 Volt
Pin 6	Pin 6	LCD HG Beleuchtung 5 Volt
Pin 7	Pin 7	LED TX
Pin 8	Pin 8	GND-Masse

## J5 LP

Die Steckbuchse ist 2 polig.

PIN	Beschreibung
Pin 1	Lautsprecher A
Pin 2	Lautsprecher B

**Achtung der Lautsprecherausgang darf nicht einseitig auf Masse gelegt werden. Der Ausgang vom TPA4861 ist eine Brückenschaltung.**

### J6 PTT

Die Steckbuchse ist 2 polig und für die PTT-Taste. Ich nutze als PTT die Taste im Eigenbau Standmikrofon.

PIN	Beschreibung
Pin 1	PTT Eingang zum PIC
Pin 2	GND-Masse

### J7 Keyer

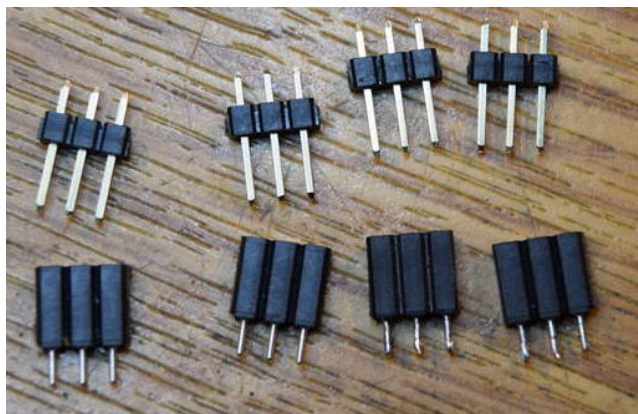
Die Steckbuchse ist 3 polig und für das Keyer-Paddel. Es kann aber auch eine einfache Morsetaste angeschlossen werden.

PIN	Beschreibung
Pin 1	Punkt oder Morsetaste
Pin 2	GND-Masse
Pin 3	Strich

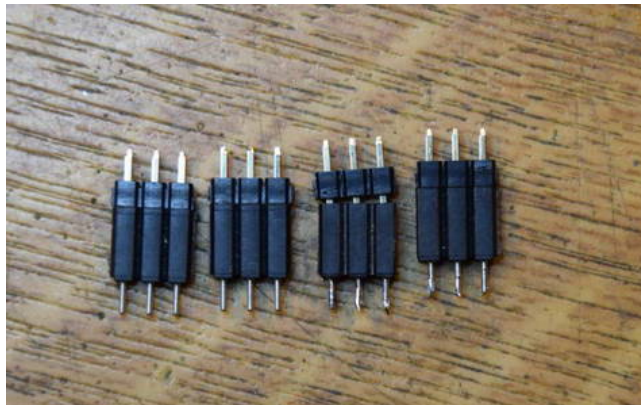
## 2.2.4 Steckverbindungen Zusatz-Platine und USB-Platine

Die 4x 3-poligen Steckverbindungen zwischen Zusatz-Platine und USB-Platine sind jetzt noch frei. Ich habe die Steckerseite der Steckverbindung unten in der USB-Platine eingelötet und die Buchsenseite in der Zusatz-Platine

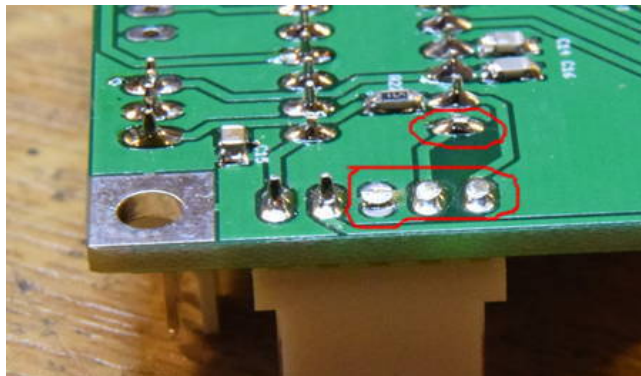
### Verbindungselemente



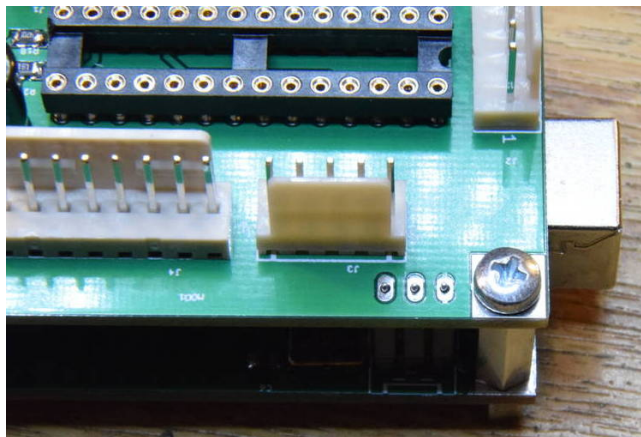
Diese Stecker und Buchsen verwende ich für die Verbindung zur USB-Platine.



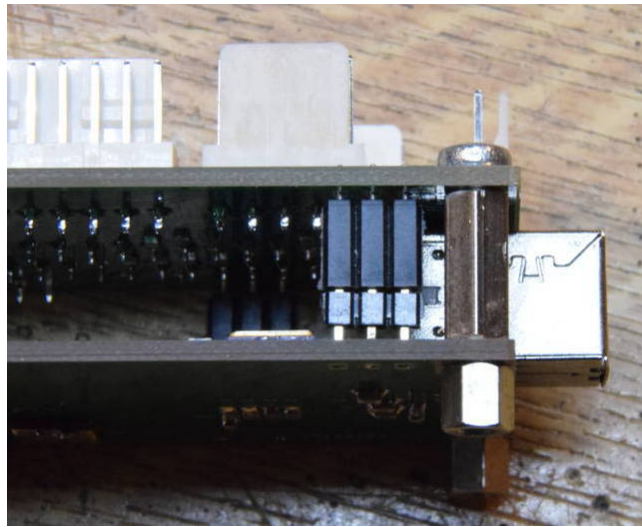
Hier zusammen gesteckt. So werden die Steckverbindungen eingelötet.



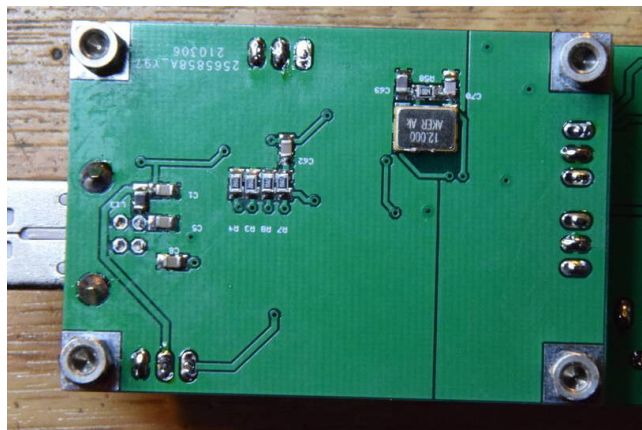
Da wir beide Platinen so nahe wie möglich montieren müssen, sonst reicht die Länge der Steckverbindungen zwischen den Platinen nicht, ist es notwendig 4 überstehenden Pins soweit wie möglich zu kürzen.



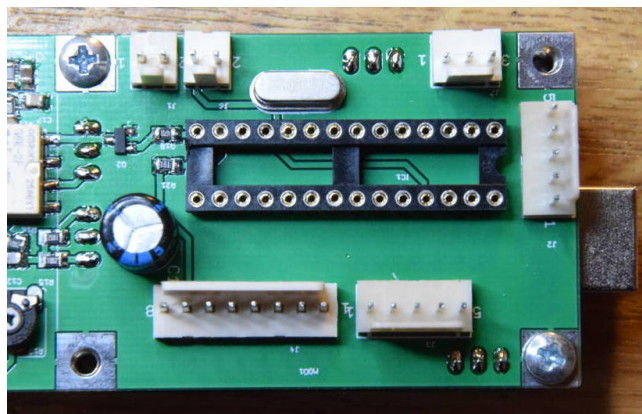
Die 4x 3-poligen Steckverbindungen werden zwischen den beiden Platinen eingefädelt.



Man sieht das die Verbindung eigentlich ein bisschen zu kurz ist. Es geht aber gerade noch so.



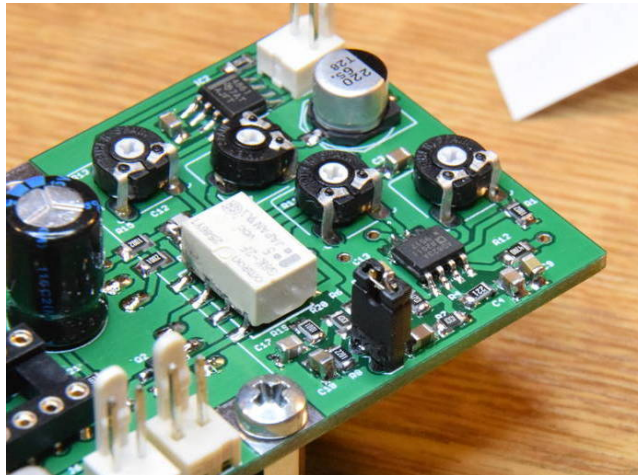
Nun verlöten wir alle Verbindungen. Hier die Sicht von unten.



Hier die Sicht von oben.



## Spannungsversorgung für das Mikrofon



Zum Schluss löten wir noch den Brückenstecker für die Spannungsversorgung des Elektret Mikrofons ein. Wenn ein dynamisches Mikro verwendet bleibt die Brücke offen.

### 2.2.5 Die Einstellregler der Zusatz-Platine

#### **R13, 100kOhm**

Mit R13 wird die Gegenkopplung (Verstärkung) des TPA4861 eingestellt. Das ist die Grundlautstärke im Lautsprecher. Der „Volume-Regler“ für die Lautstärke befindet sich im PC in der Software SDR-Console.

#### **R2, 22kOhm**

Mit R2 stellen wir die Verstärkung des Mikrofon-Verstärkers ein. Die Verstärkung richtet sich nach dem Mikrofontyp. Ich habe ein Elektret Mikro verwendet.

#### **R11, 50kOhm**

Der Modulationsgrad im Mode CW wird mit diesem Regler eingestellt. Dabei ist zu beachten, dass der TX-Equalizer ausgeschaltet wird und der Sprachprozessor auf 0 geregelt werden sollte. Die Oberwellen werden so am besten unterdrückt.

#### **R17, 22kOhm**

Lautstärke des Mithörtones regelt R17. Vorher sollte mit R13 die Grundlautstärke eingestellt werden.

## 2.3 Display-Platine

### 2.3.1 LCD-Anzeige, DOG104SPI



Hier sehen wir die LCD-Anzeige mit der Hintergrundbeleuchtung. Die Pins der LCD werden durch die Hintergrundbeleuchtung gesteckt und verlötet. Das erhöht die Stabilität und verhindert, dass die Glasplatte der LCD beim aufstecken bricht. **Achtung die LCD nur leicht andrücken vor dem Verlöten, damit die LCD nicht unter „Spannung“ steht.** Ich habe eine defekte Anzeige erzeugt, durch zu festes andrücken.

#### Display Zusammenbau

Geliefert wird die LCD-Anzeige „EA DOGS104W-A“ und die Hintergrundbeleuchtung „EA LED36X28-A“. Die Schutzfolien werden entfernt und alle PINs der LCD-Anzeige werden durch die Löcher der Hintergrundbeleuchtung gesteckt, so dass sich eine Einheit ergibt. Damit das ganze stabil wird, werden alle PINs verlötet. Verlöten wir nicht, wird es ganz schwer das Display wieder aus der Fassung zu ziehen ohne es zu zerbrechen.

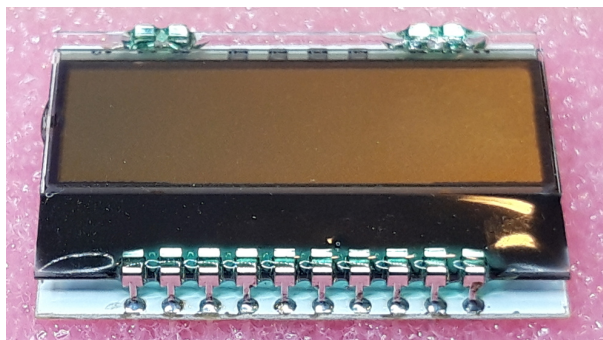


Foto vom Display. Alle Pins werden verlötet.

### Display mit Fassung in Leiterplatte einbauen

Aus der Buchsenleiste „SPL 20“ brechen wir 2x 2 Buchsen ab für die unteren beiden PINs am Display und 1x 10 Buchsen für die oberen PINs am Display. Damit am Display keine mechanischen Spannungen entstehen werden die Buchsenleisten mit gestecktem Display in die Leiterplatte eingelötet. **Achtung!! das Display ist sehr zerbrechlich.**



Hier habe ich die Buchsenleisten auf die Pins am Display aufgesteckt. Anschließend wird das Display mit Buchsenleiste in die Leiterplatte gesteckt und verlötet. So vermeidet man mechanische Spannungen am Display.

### 2.3.2 Drehgeber, Taste und LED

Wir löten noch den Drehgeber dies Einzeltaste und die LED ein. Die LED sollte erst nur mit einem Beine eingelötet werden, damit man korrigieren kann beim Einbau in die Frontplatte.

### 2.3.3 Verbindungen zur Zusatz-Platine

Damit die LCD-Anzeige und Bedienung der Baugruppe funktioniert brauchen wir noch Verbindungen mit 13x Litze 0,14mm<sup>2</sup>. Ich habe die Litzen direkt angelötet. Am anderen Ende sind 2 Buchsen-Stecker ein 5-poliger Stecker und ein 8-poliger Stecker, die auf die Zusatz-Platine aufgesteckt werden.

**J1 Dreh** 5 Litzen direkt anlöten.

Display-Platine	Zusatz-Platine	Beschreibung
Pin 1	Pin 1	GND-Masse
Pin 2	Pin 2	Drehgeber Impulse
Pin 3	Pin 3	Drehgeber Drehrichtung
Pin 4	Pin 4	Drehgeber Taste
Pin 5	Pin 5	Einzeltaste

**J2 LCD** 8 Litzen direkt anlöten.

Display-Platine	Zusatz-Platine	Beschreibung
Pin 1	Pin 1	LCD CS
Pin 2	Pin 2	LCD SCL
Pin 3	Pin 3	LCD SID
Pin 4	Pin 4	LCD Reset
Pin 5	Pin 5	LCD 3,3 Volt
Pin 6	Pin 6	LCD HG Beleuchtung 5 Volt
Pin 7	Pin 7	LED TX
Pin 8	Pin 8	GND-Masse

## Kapitel 3

# Firmware, Software

Die Firmware wurde in Assembler geschrieben. Der Quelltext ist etwa 2800 Zeile lang ohne die Gleitpunktbibliothek (2700 Zeilen).

**DAC-Wandler** Ich habe das erste Mal den DAC-Wandler im PIC aktiviert und Versuche gemacht. Der DAC-Wandler im PIC18F26K22 bzw. PIC18F25K22 hat eine Wandlerbreite von 5 Bit. Daraus ergeben sich 32 Abstufungen in der Amplitude. Im Ram habe ich 36 Werte der Sinuskurve abgelegt. Eine volle Sinuskurve sind die Sinuswerte von  $0^\circ$  bis  $360^\circ$ . Dann geht es wieder bei  $0^\circ$  los. Alle  $10^\circ$  berechne ich den Sinus und addiere ihn zur Mittellinie. Die Mittellinie befindet sich genau zwischen 0 und 31 (0x0 Hex und 0x1F Hex). Jetzt musste ich noch einen Timer programmieren der nach einer genau definierten Zeit einen Interrupt auslöst und der Reihe nach die errechneten Zahlenwerte der Sinuskurve auf dem DAC-Wandler ausgibt. Das ist einfach genial!

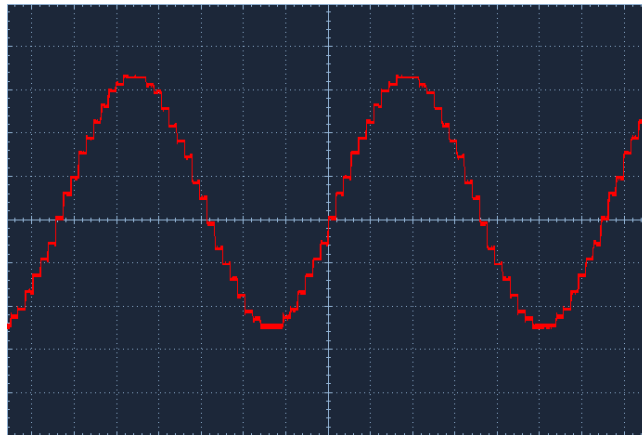
Ein kleines Rechenbeispiel. Ich möchte einen Sinus-Ton von 800 Hz erzeugen.

$$\frac{1}{800Hz} = 0,00125\text{Sekunden}$$

Ein volle Sinus-Schwingung dauert 0,00125 Sekunden. In dieser Zeitspanne müssen 36 Werte am DAC-Wandler ausgegeben werde. Also noch mal durch 36 teilen.

$$0,00125/36 = 0,034722m\text{Sekunden}$$

Bei 800 Hz muss aller 0,034722 mSek ein DAC-Wert ausgegeben werden. Durch die Änderung der Zeit kann ich die Sinus-Frequenz verstellen. Ich habe die Frequenzausgabe auf 400 Hz bis 1000 Hz eingeschränkt. Durch die hohe CPU-Taktfrequenz von 64 MHz im PIC erledigt der PIC die Ausgabe der Sinusfrequenz nebenbei und wird wenig belastet. Der DAC-Ausgang am PIC ist sehr Hochohmig und darf nur wenig belastet werden. Ich habe den OPV AD823 als Impedanzwandler zwischen geschaltet.



Damit man ein Vorstellung hat wie der Sinus aussieht, hier ein Bild mit dem Oszi gemacht. Man sieht die einzelnen Stufen in der Kurve, aber der Takt dieser Stufen ist sehr hoch hat einen großen Abstand zur Grundwelle. Bei 800 Hz ist der Ausgabetak 28,8 kHz und liegt außerhalb des Übertragungsbereiches des Audiochip (speziell in unserer Anwendung). Ein Tiefpass ist nicht nötig.

Ich nutze auch den Nullpunkt der Sinuskurve ( $0^\circ$ ) zum Ein- und Ausschalten des Tones. Ich erhoffe mir dadurch saubere CW-Signale möglichst ohne Tastenклик.

## 3.1 Bedienung der Baugruppe

Die Baugruppe hat 2 Bedienelemente. Einen Drehgeber mit Tastenfunktion und eine Einzeltaste.

### 3.1.1 Drehgeber

Der Drehgeber hat im *Mode SSB* keine Funktion. Im *Mode CW* wird mit dem Drehgeber die Keyer-Geschwindigkeit in WpM (**W**örter **p**ro **M**inute) eingestellt. Einstellbereich 9 WpM bis 40 WpM.

### 3.1.2 Einzeltaster

Der *kurze Tastendruck* ist der Zugang in das Menü. Die Beschreibung ist im Kapitel 3.2. Der *lange Tastendruck* schaltet den Mode SSB/CW um.

Der *kurze Tastendruck* ist in fast allen Funktion die *OK-Taste*.

### 3.1.3 Mode SSB

Nach dem Drücken der PTT-Taste sollte man mit dem Sprechen warten bis die LED auf der Frontplatte aufleuchtet.

### 3.1.4 Mode CW

Mit einem Punkt oder Strich startet das Umschalten vom Empfang auf Senden. Wenn die LED auf der Frontplatte leuchtet kann mit dem Senden der CW-

Zeichen begonnen werden. Die Gebegeschwindigkeit wird mit dem Drehgeber eingestellt. Die Änderung wird erst in den Gebepausen aktiv.

## 3.2 Menu-Funktionen

Die Einzeltaste oder die Taste im Drehgeber startet das Menü. Die Auswahl der Funktion erfolgt anschließend mit dem Drehgeber. Die Einzeltaste oder die Taste im Drehgeber startet die ausgewählte Funktion.

### 3.2.1 [1] M Ton, zum Frq. abstimmen

Diese Funktion schaltet den Mithörton ein. Früher nannte man das „Einpfeifen“. So kann man parallel zum zum Empfangenen CW-Signal die Empfangsfrequenz verstellen, bis das CW-Signal und der Mithörton auf der gleichen Tonhöhe sind.

### 3.2.2 [2] SSB/CW, Mode umschalten

Dieser Menüpunkt schaltet den Mode um. Immer wechselseitig Mode SSB/CW. Die Umschaltung kann man auch mit einem „langen Tastendruck“ der Einzeltaste vollführen. Das geht noch schneller und man braucht nicht erst in das Menü wechseln.

### 3.2.3 [3] P.x5, 5 Punkte RX–TX

Diese Funktion ist eine Hilfsfunktion. Es wird die Umschaltzeit von Empfang auf Senden ermittelt. Dazu nutzen wir die Empfangsanzeige im Wasserfall der „SDR-Console“. Die Software in der „SDR-Console“ geht auf Senden und gibt 5 CW Punkte ab zum TRX. Im Wasserfall der Software „SDR-Console“ sieht man die gesendeten Punkt als Echo vom QO100 zurück kommen. Mit der Anzahl der empfangenen Punkte im Wasserfall können wir die Zeitverzögerung des Übergangs von *Empfang auf Senden* sichtbar machen. Sind alle 5 Punkte im Wasserfall zusehen, gab es keine Zeitverzögerung beim Umschalten von Empfang auf Senden. Angenommen es sind nur 2 Punkte im Wasserfall zu sehen, würde die Zeitverzögerung 3x 0,2 Sekunden betragen. Das sind 600 mSek. Ein Punkt hat eine Signallänge von 100 mSek. mit einer folgenden Pause von 100 mSek. Diese 600 mSek werden im SETUP Funktion [3] *Timer, RX-- > TX* eingetragen. Damit ändern wir nichts an der Zeitverzögerung, aber mit der rote LED neben dem Display bekommen wir eine Information wann der TX Sendebereit ist. Vom drücken der PTT dauert es 600 mSek der TX bereit ist und die rote LED leuchtet. Ab jetzt kann man sicher sein, dass das SSB-Signal oder CW-Signal gesendet wird.

### 3.2.4 [4] P.x5, 5 Punkte TX–RX

Mit dieser Funktion wird die Umschaltzeit TX-- >RX ermittelt. Ich habe hier 100 mSek. im SETUP eingetragen. Damit habe ich keine Probleme. Mein TRX mit dem Adalm-Pluto ist in meinem Haus-Ethernet eingebunden. Dadurch habe ich lange Umschaltzeiten von Empfang auf Senden und auch umgekehrt.

Um die Umschaltzeiten RX-TX-RX klein zu halten haben einige OMs sich eine ganz raffinierte Sache einfallen lassen. Sie betreiben auch den Adalm-Pluto

TRX über das Ethernet, aber mit einer Punkt zu Punkt Verbindung. Das erlaubt für die Sende-Empfangsumschaltung PoE(**P**ower **o**ver **E**thernet) zu nutzen. Das wird Verzögerungsfrei vom PC zum TRX weiter geleitet. Dabei hat sich aber heraus gestellt, dass die letzten gesprochenen Worte oft nicht mit gesendet werden. Das liegt am zu schnellen Umschalten von Senden auf Empfangen. Die gesprochenen Worte werden in der Software digitalisiert und in einen Puffer geschrieben. Wird zu schnell umgeschaltet sind noch gesprochene Worte im Puffer, aber der TRX kann sie nicht mehr senden, da schon auf Empfang geschaltet wurde.

Wie lange gewartet werden muss bis auf Empfang geschaltet werden kann, wird mit dieser Funktion getestet. Sobald die Funktion gestartet wird, geht der TRX auf Senden, wartet eine Sekunde und sendet anschließend 5 CW Punkte. Nach dem letzten Punkt wird sofort auf Empfang geschaltet. Im Wasserfall der Software sehen wir am Echo des QO100 ob Punkte fehlen. Die Anzahl der fehlenden Punkt  $\times 0,2$  Sek ist die Zeit die wir im SETUP Funktion [4] *Tim.SSB, TX-- > RX* einstellen sollten. Genau um diese Zeit muss der Sender länger an bleiben, damit die letzten gesprochenen Worte noch gesendet werden. Die rote LED neben dem Display zeigt das längere Senden an und verlischt um diesen Zeitbetrag später.

### **3.2.5 [5] SETUP, Zugang zum SETUP**

Der Menüpunkt 5 ist der Zugang zum SETUP. Die Funktionen folgen im nächsten Kapiteln.

## **3.3 SETUP-Funktionen**

Die Auswahl der Funktion erfolgt mit dem Drehgeber. Die Einzeltaste oder die Taste im Drehgeber startet die ausgewählte Funktion.

### **3.3.1 [1] Abbruch, SETUP Abbrechen**

Diese Funktion braucht man nicht weiter erklären.

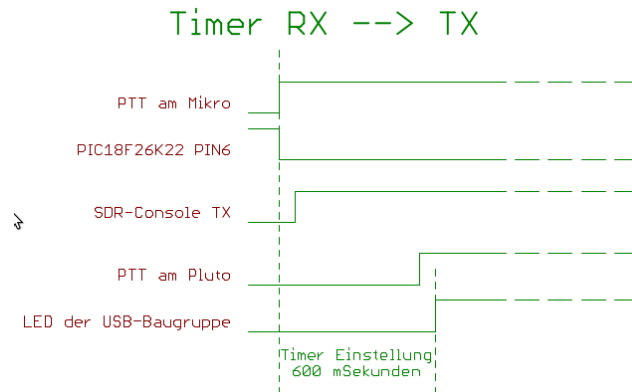
### **3.3.2 [2] MTON, Mitherton einstellen**

Der Mitherton wird zum Senden bei CW verwendet und ist ein Sinuston der mit dem DAC-Wandler des PIC18F26K22 erzeugt wird. In dieser Funktion wird die Tonhöhe eingestellt. Der Frequenzbereich geht von 400 Hz bis 1000 Hz.

### **3.3.3 [3] Timer, RX-TX einstellen**

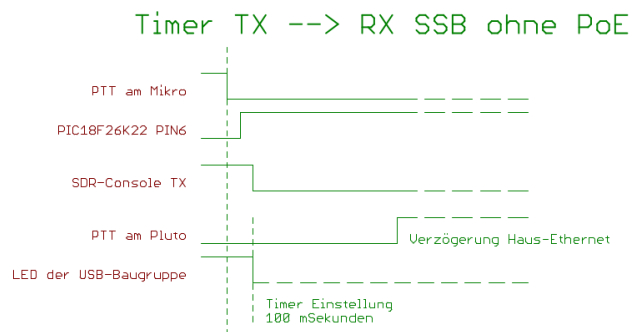
Um diese Zeit geht verzögert die LED neben dem Display an. Nach dem Drücken der PTT dauert es bei meiner TRX-Anlage etwa 600 mSek bis der Sender aktiv wird. Die 600 mSek habe ich hier eingestellt. Die Verzögerung wird mit der Menüfunktion [3] gemessen.



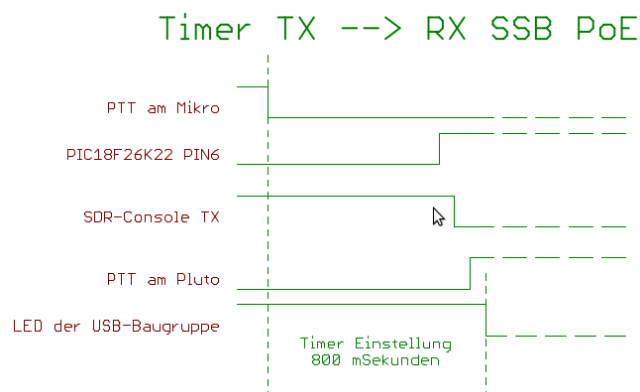


### 3.3.4 [4] Tim.SSB, TX-RX einstellen

Wie schon im Kapitel 3.2.4 auf Seite 22 erwähnt, wird hier die Zeit für das verzögerte Abschalten des Senders eingestellt. Beschreibung dazu in 3.2.4.



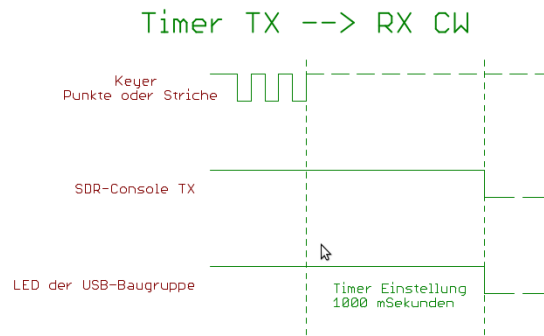
Das ist meine Konstellation mit der Ethernet Verbindung im Hausnetz. Ich habe immer eine große Verzögerung.



Bei der Konstellation mit PTT-Umschaltung über PoE sieht das etwas anders aus. So etwa könnte es sein.

### 3.3.5 [5] TimerCW, TX–RX einstellen

Für die verzögerte Abschaltung des Senders bei CW gelten andere Regeln. Die Zeit sollte noch größer sein als bei SSB. In den Gebepausen soll der Sender möglichst nicht abfallen. Maximal kann die Verzögerung bis zu 2 Sekunden betragen.



Im Mode CW sollte man immer die Abschaltzeit etwas höher wählen.

### 3.3.6 [6] Keyer, Mode einstellen

In diesem Punkt wird die Keyer-Software aktiviert.

**Keyer Norm** „Keyer Normal“ Punkte linkes Paddel und Striche rechtes Paddel.

**Keyer Rev.** „Keyer Revers“ Striche linkes Paddel und Punkte rechtes Paddel.

**Handtaste** Keyer deaktiviert. Normale Handtaste.

Diese Möglichkeiten gibt es zum Einstellen.

### 3.3.7 [7] Keyer, Punktsp einstellen

Fast jede Keyer-Software hat einen Punktspeicher. Stößt man mit dem Strichpaddel die Ausgabe eines Striches an, wartet nicht bis er zu ende ist und gibt mit dem Punktpaddel einen Punkt aus, würde der Punkt nicht ausgegeben, wenn da nicht der Punktspeicher wäre.

Ein Strich besteht normalerweise aus 4 Zeiteinheiten. 3 Zeiteinheiten das Strich-Zeichen und eine Zeiteinheit die Pause. Der Punktspeicher kann folgendermaßen eingestellt werden.

**Aus** Punktspeicher ist deaktiviert

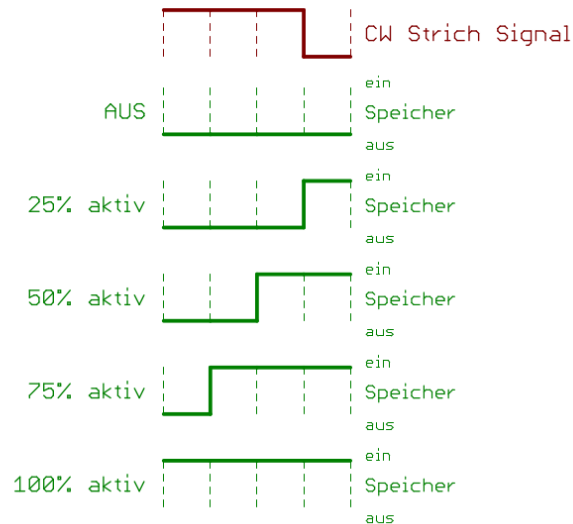
**25%** der Punktspeicher ist im letzten viertel der 4 Zeiteinheiten aktiv. Also nur während der Pause nach dem Strich-Zeichen.

**50%** der Punktspeicher ist in der letzten Hälfte der 4 Zeiteinheiten aktiv.

**75%** der Punktspeicher ist im letzten Dreiviertel der 4 Zeiteinheiten aktiv.

**100%** der Punktspeicher ist immer, während des ganzen Striches aktiv. Mit anschließender Zeiteinheit der Pause.

### Diagramm des Punktspeichers



#### 3.3.8 [8] Keyer, Strichsp einstellen

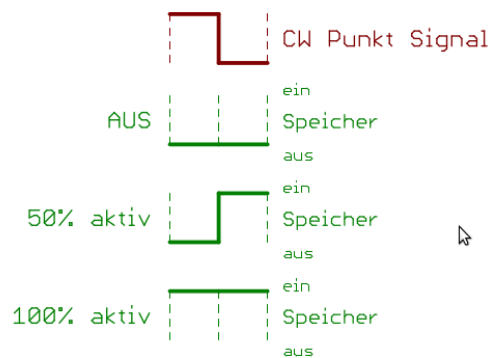
Es gibt in meiner Software auch einen Strichspeicher, der während der Punkt-  
ausgabe aktiv ist. Der Punkt besteht dem Zeichen des Punktes (eine Zeiteinheit)  
und einer anschließenden Pause (eine Zeiteinheit). Also 2 Zeiteinheiten. Folgen-  
de Einstellmöglichkeiten habe wir:

**Aus** Strichspeicher ist deaktiviert

**50%** der Strichspeicher ist in der letzten Hälfte der 2 Zeiteinheiten aktiv.

**100%** der Strichspeicher ist immer, während des ganzen Punktes mit anschlie-  
ßender Pause. aktiv.

### Diagramm des Strichspeichers

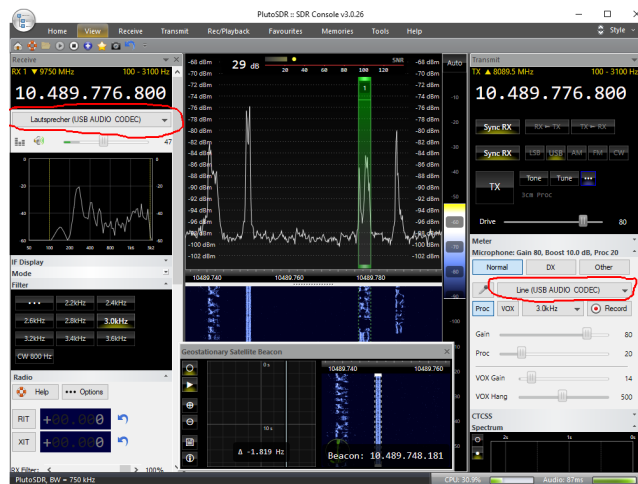


### 3.3.9 [9] Display, drehen 180°

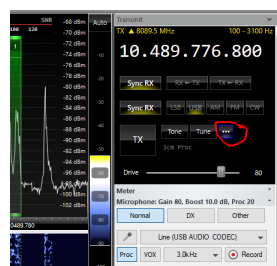
Im Display kann per Software die Zeichenausgabe um 180° gedreht werden. Das LCD-Display hat nur aus einem bestimmten Blickwinkel den vollen Kontrast. Wurde das Display symmetrisch in die Frontplatte eingebaut und die Frontplatte ist um 180° drehbar, kann getestet werden wie der beste Kontrast erzielt wird. Nach bestätigen mit der Einzeltaste führt der PIC einen Warmstart aus und das Display wird neu initialisiert. Die Schrift steht auf dem Kopf!

## 3.4 Einstellungen in der SDR-Console

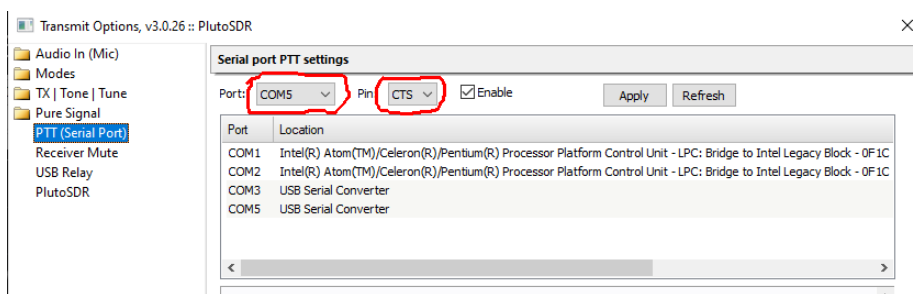
Damit alles einwandfrei funktioniert muss der passende Sounddevice eingestellt werden. Sowohl im RX-DSP als auch im TX-DSP. Bei den Einstellungen ist nicht viel zu beachten.



Links im RX-DSP und rechts im TX-DSP stehen jetzt die „USB Audio-Codec“ Devices zur Auswahl.

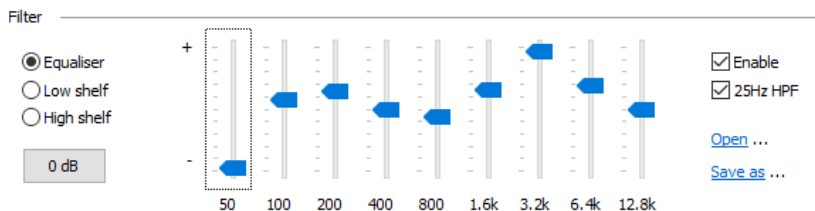
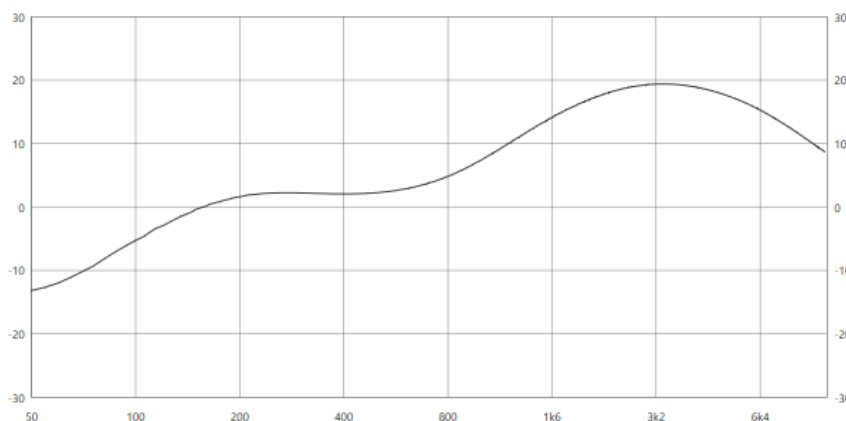


Die drei Punkte sind der Button für den Zugang zu den Einstellungen.

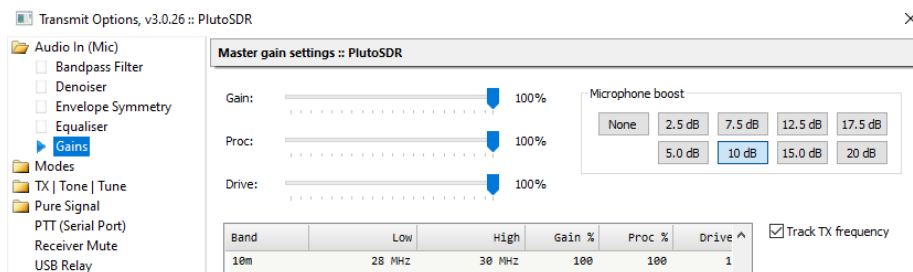


Die RS232 der Platine ist die „COM5“ und das PIN für die PTT ist „CTS“. Mit „Refrehs“ und „Apply“ wir die Auswahl bestätigt. Die PTT funktioniert jetzt.

**Graphic Equaliser settings (transmit audio) - Profile 1**



Ich benutze eine Elektret-Mikrofonkapsel. Passend dazu noch meine Einstellung im TX-Equaliser.



Zusätzlich habe ich die Verstärkung noch etwas erhöht (10dB), in diesem Fenster sichtbar.

## 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](mailto:DL4JAL@t-online.de)