Yaesu FT-8800

Inhaltverzeichnis

Yaesu FT-8800 Umbau Display-Farbe Protokoll der Datenübertragung von Panel-Unit zu Main-Unit Protokoll der Datenübertragung von Main-Unit zu Panel-Unit Yaesu FTDEC - Dekodiert das Display-Protokoll



Gelb ist an sich eine schöne Farbe, aber im Auto möchte man gerne eine einheitliche Farbe aller Anzeigen geniessen. Das wären dann zur Zeit Blau oder Rot. Also los gehts !

Zuerst nimmt man das Bedienteil vom Funkgerät oder der Haltung ab.



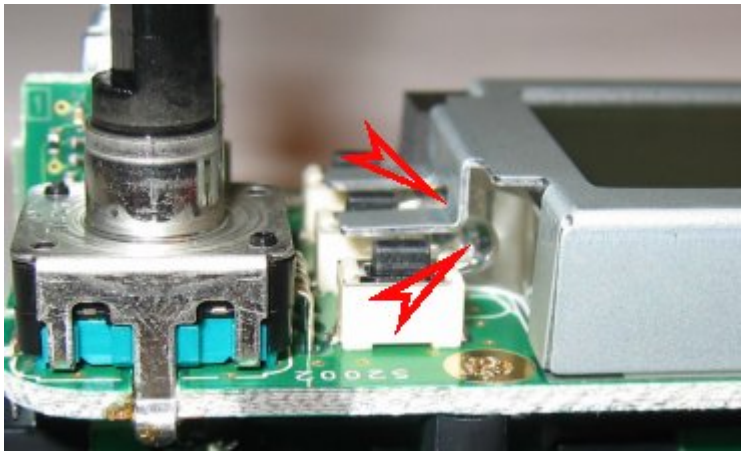
Auf der Rückseite 4 Schrauben entfernen.



Vorne dann die Knöpfe abziehen und die Muttern von den unteren Reglern abdrehen.



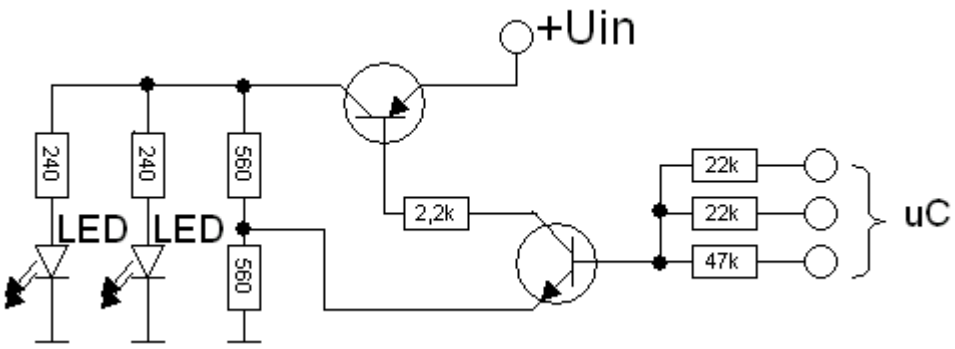
An der rechten Seite unter den Metall-Laschen kann man die beiden LEDs erkennen. Es handelt sich dabei laut Schaltplan um klare 3mm LEDs in Gelb von Toshiba (Nach Toshiba TLYH262 suchen. Typisch 2,1V mit 20mA)



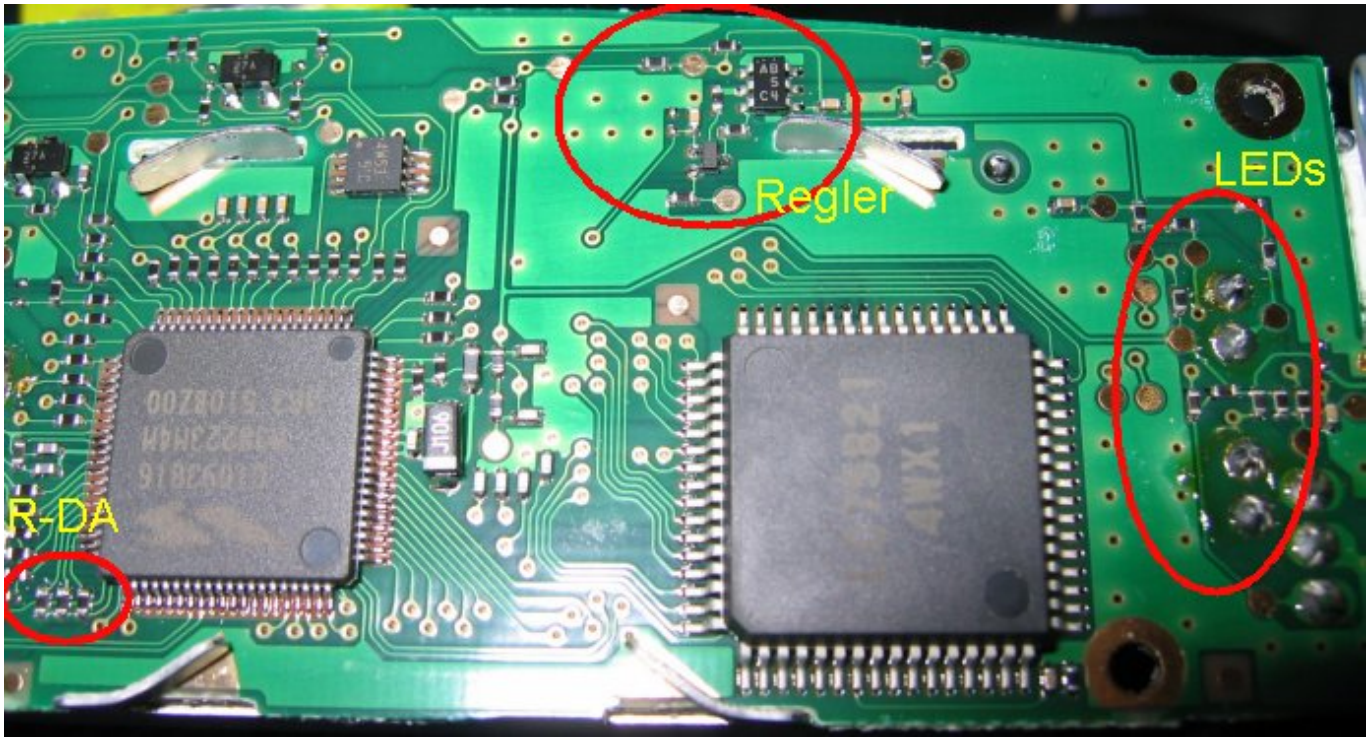
Dann stellt sich die Frage wie die Schaltung dahinter funktioniert. Schließlich gibt es ja auch noch eine

Dimmer-Funktion vom Menü 9 aus.

Hier der Prinzipschaltplan der interessanten Teile:



Hier noch markiert, wo sich diese auf der Platine befinden.



In meinem Gerät lag als Uin eine Spannung von 8,2V an. Pro LED waren es etwa 25,2mA Strom. Die Spannung am Ausgangs des Längsreglers lag bei 8,1V welche sich auf 6,05V beim 240Ohm und 2,04V bei der LED aufteilen.

Der Microcontroller (hier links im Bild) kann nun mit 3 Ports und den Widerständen (eine Art einfacher DA-Wandler) verschiedene Zustände einstellen. Diese gehen dann als Steuerspannung auf die Basis des unteren Transistors und legen damit die Ausgangsspannung fest. U uC Rx ist die Spannung am Pin des Microcontrollers, Uout die Summenspannung an der Basis des unteren Transistors nach Masse.

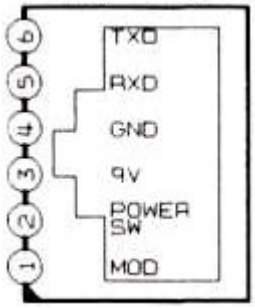
Mode	U uC Rleft [V]	U uC Rmid [V]	U uC Rright [V]	Uout [V]
Dim Off	0	0	0	0
Dim 1	4,93	4,93	4,93	4,93
Dim 2	0	4,93	4,93	2,93
Dim 2	4,93	0	0	1,99

Wenn ich mehr Zeit habe, tausche ich die LEDs aus gegen andere und gebe Bezugsquelle sowie Preis mit an.



Die Absetzbarkeit des Bedienteils ist eine schöne Sache. Ein einfaches 6pol. Western-Kabel reicht aus und kostet nicht viel. Wie funktioniert aber die Übertragung der gedrückten Tasten und der Drehregler hin zum Funkgerät ?

Die 6polige Buchse am Funkgerät ist wie folgt belegt:



Für unseren Fall sind nur GND und RXD interessant. Hier wird mit dem Protokoll der RS232 gefahren, allerdings nur mit TTL-Leveln. Die Datenrate beträgt 19200 Baud, 8n1.

Zur Umsetzung Richtung PC benötigt man nun noch einen TTL zu RS232-Wandler z.B. mit MAX232 o.ä. um folgende Logik zu erreichen : +5V TTL⇒ -15V RS232, 0V TTL ⇒ +15V RS232



In zyklischen Abständen überträgt der Microcontroller der Panel-Unit alle Informationen Richtung Funkgerät. Dies geschieht mit Paketen welche jeweils 13 Bytes enthalten. Alle Bytes sind belegt. Der Start eines Paketes läßt sich also nur erkennen, wenn vorher auf dem Bus lange Ruhe war.

Bsp.:

```
80 00 7F 7F 0E 7F 07 57 7F 7F 7F 00 00
```

Im folgenden werden alle Kommandos der Tasten und Regler an der Panel-Unit sowie dem Mikro untersucht. Die Bytes in einem Frame werden einfach durchgezählt:

```
01 02 03 04 05 06 07 08 09 10 11 12 13
```

Für das Bedienteil gilt :



Squelch und Lautstärke sind Potis, welche per A/D-Wandler im Microcontroller eingelesen werden und

den Wertebereich von 00 bis 7F haben

```
Squelch rechts             : Byte 4 offen 7F, geschlossen 00
Lautstärke rechts          : Byte 5 leise 00, laut 7F

Squelch links              : Byte 8 offen 7F, geschlossen 00
Lautstärke links           : Byte 7 leise 00, laut 7F

Hypermemory Taste 1        : Byte 13 auf 01
Hypermemory Taste 2        : Byte 13 auf 02
Hypermemory Taste 3        : Byte 13 auf 03
Hypermemory Taste 4        : Byte 13 auf 04
Hypermemory Taste 5        : Byte 13 auf 05
Hypermemory Taste 6        : Byte 13 auf 06

Linker Wahlknopf drehen    : Byte 1 normal auf 80
                             bei Linksdrehung auf FF
                             bei Rechtsdrehung auf 81

Rechter Wahlknopf drehen   : Byte 2 normal auf 00
                             bei Linksdrehung auf 7F
                             bei Rechtsdrehung auf 01

LOW links                  : Byte 10 von 7F auf 61
V/M links                  : Byte 10 von 7F auf 40
HM links                   : Byte 10 von 7F auf 1F
SCN links                  : Byte 10 von 7F auf 00

Low rechts                 : Byte 11 von 7F auf 00
V/M rechts                 : Byte 11 von 7F auf 1F
HM rechts                  : Byte 11 von 7F auf 40
SCN rechts                 : Byte 11 von 7F auf 61

SET Knopf                  : Byte 12 von 00 auf 04

VOL/SQL Taster links       : Byte 12 von 00 auf 08
VOL/SQL Taster rechts      : Ein/Ausschalter, nur elektrisch
                             als Signal auf Pin 2

Linker Wahlknopf drücken   : Byte 12 von 00 auf 02
Rechter Wahlknopf drücken  : Byte 12 von 00 auf 01

Für das Mikro gilt :

FT8800 RX Mikro

PTT am Mikro        : Byte 3 von 7F auf 1B

1 am Mikro          : Byte 6 auf 03 und Byte 9 auf 1B
2 am Mikro          : Byte 6 auf 03 und Byte 9 auf 32
3 am Mikro          : Byte 6 auf 03 und Byte 9 auf 4B
```

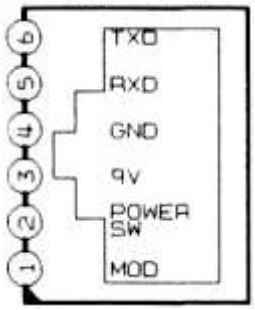
```
A am Mikro          : Byte 6 auf 03 und Byte 9 auf 66
4 am Mikro          : Byte 6 auf 19 und Byte 9 auf 1B
5 am Mikro          : Byte 6 auf 19 und Byte 9 auf 32
6 am Mikro          : Byte 6 auf 19 und Byte 9 auf 4B
B am Mikro          : Byte 6 auf 19 und Byte 9 auf 66
7 am Mikro          : Byte 6 auf 31 und Byte 9 auf 1B
8 am Mikro          : Byte 6 auf 31 und Byte 9 auf 32
9 am Mikro          : Byte 6 auf 31 und Byte 9 auf 4B
C am Mikro          : Byte 6 auf 31 und Byte 9 auf 66
* am Mikro          : Byte 6 auf 4B und Byte 9 auf 1C
0 am Mikro          : Byte 6 auf 4B und Byte 9 auf 33
# am Mikro          : Byte 6 auf 4B und Byte 9 auf 4B
D am Mikro          : Byte 6 auf 4B und Byte 9 auf 66

P1 Taste am Mikro   : Byte 6 auf 63 und Byte 9 auf 1C
P2 Taste am Mikro   : Byte 6 auf 63 und Byte 9 auf 33
P3 Taste am Mikro   : Byte 6 auf 63 und Byte 9 auf 4C
P4 Taste am Mikro   : Byte 6 auf 63 und Byte 9 auf 66

UP-Taste am Mikro   : Byte 6 auf 1D und Byte 9 auf 06
DOWN-Taste am Mikro : Byte 6 auf 34 und Byte 9 auf 06
```

Die Absetzbarkeit des Bedienteils ist eine schöne Sache. Ein einfaches 6pol. Western-Kabel reicht aus und kostet nicht viel. Wie funktioniert aber die Übertragung vom Funkgerät zum Display ?

Die 6polige Buchse am Funkgerät ist wie folgt belegt:



Für unseren Fall sind nur GND und TXD interessant. Hier wird mit dem Protokoll der RS232 gefahren, allerdings nur mit TTL-Leveln. Die Datenrate beträgt 19200 Baud, 8n1.

Zur Umsetzung Richtung PC benötigt man nun noch einen TTL zu RS232-Wandler z.B. mit MAX232 o.ä. um folgende Logik zu erreichen : +5V TTL⇒ -15V RS232, 0V TTL ⇒ +15V RS232



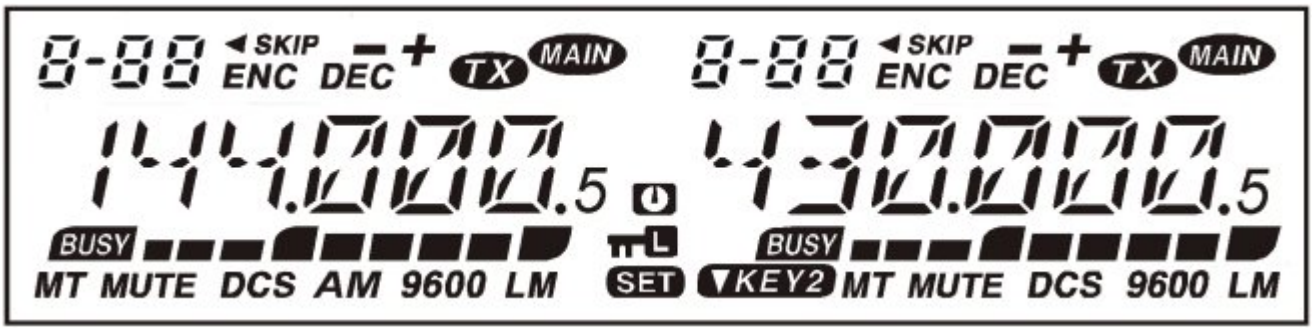
In zyklischen Abständen überträgt der Microcontroller des Funkgerätes ein Paket mit 42 Bytes. Es gibt keine Start/Ende-Kennung im Frame, so das nach einer ruhigen Lücke auf dem Bus von >1ms gesucht werden muss um den Anfang zu finden.

In den 42 Bytes sind nun alle Segmente einkodiert. Es wird also keine Frequenz oder ein ASCII-Text übertragen sondern jedes Segment wird einzeln an oder ausgeknippst !



In der Panel-Unit ist nun ein Microcontroller der diese aufbereitet und weiter aufteilt damit die 264 Segmente angesteuert werden können:

Microcontroller : 30 Leitungen = 60 Segmente Display-Controller 1 : 51 Leitungen = 102 Segmente Display-Controller 2 : 51 Leitungen = 102 Segmente



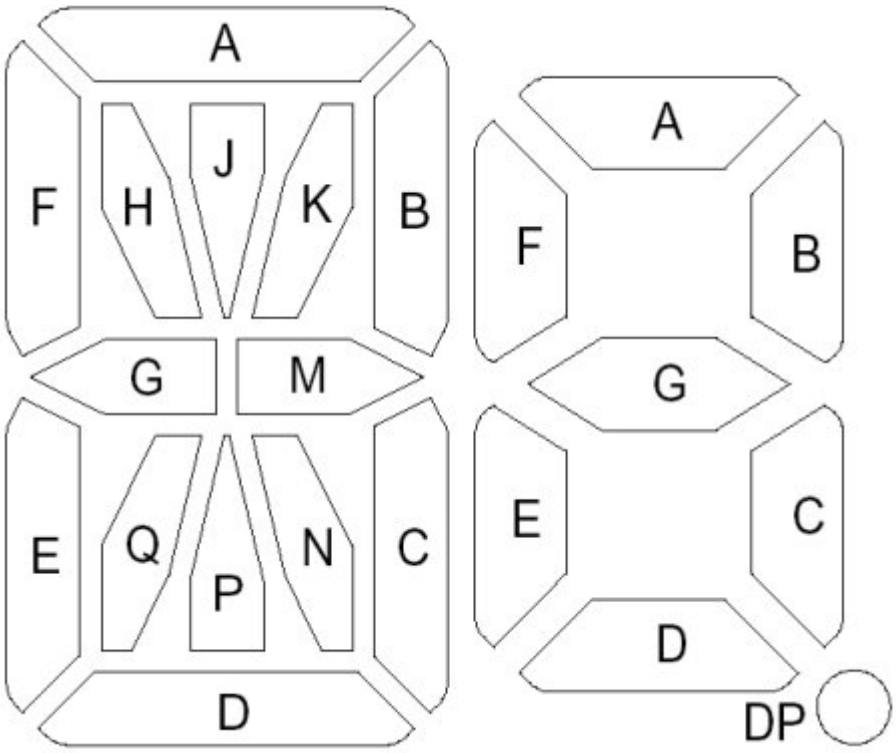
Als einzige Zusatz-Info wird noch die Helligkeit der Display-Beleuchtung im letzten Byte mitgegeben (4 Stufen) welche direkt auf die 3 Ports geht und ein Widerstands-DA Wandler ansteuert.

In der Reihenfolge des Empfangs werden die Bytes von 01 … 42 durchnummeriert.

Ein Beispiel wäre die Anzeige LUBU 500.000 (Helligkeit DIM 1) :

```
84 00 00 00 00 14 10 06 38 61 20 01 31 00 44 01 20 07 39 1D 64 75 10 07 69
21 2E 07 39 33 04 01 01 00 00 00 20 18 00 00 00 0E
```

Für die 13-Segment-Anzeige (J+P kombiniert) sowie 7-Segment-Anzeige gilt folgende Benennung:



Die Stellen der 14-Segment-Anzeige sind von links nach rechts mit 1..6 durchnummeriert.

Die Zuordnung zwischen Bytes und Segmenten ist damit wie folgt:

```
Byte  Bit   Bedeutung
---------------------

01    80    ???
01    40    Frequenz links Stelle 6 Balken M
01    20    Frequenz links Stelle 6 Balken C
01    10    S-Meter links Balken 9
01    08    Dezimal-Punkt kHz links
01    04    Middle TX Power links
01    02    Automatic Power Off Active
01    01    Kleine kHz 5 links

02    80    ???
02    40    Frequenz links Stelle 6 Balken N
02    20    Low TX Power links
02    10    Frequenz links Stelle 6 Balken H
02    08    Frequenz links Stelle 6 Balken G
02    04    Frequenz links Stelle 6 Balken A
```

```
02    02    Frequenz links Stelle 6 Balken K
02    01    Frequenz links Stelle 6 Balken B

03    80    ???
03    40    S-Meter links Balken 7
03    20    Frequenz links Stelle 6 Balken F
03    10    Frequenz links Stelle 6 Balken E
03    08    Frequenz links Stelle 6 Balken Q
03    04    S-Meter links Balken 8
03    02    Frequenz links Stelle 6 Balken J+P
03    01    Frequenz links Stelle 6 Balken D

04    80    ???
04    40    Frequenz links Stelle 5 Balken H
04    20    Frequenz links Stelle 5 Balken G
04    10    Frequenz links Stelle 5 Balken A
04    08    Frequenz links Stelle 5 Balken K
04    04    Frequenz links Stelle 5 Balken B
04    02    Frequenz links Stelle 5 Balken M
04    01    Frequenz links Stelle 5 Balken C

05    80    ???
05    40    Frequenz links Stelle 5 Balken E
05    20    Frequenz links Stelle 5 Balken Q
05    10    S-Meter links Balken 6
05    08    Frequenz links Stelle 5 Balken J+P
05    04    Frequenz links Stelle 5 Balken D
05    02    Frequenz links Stelle 5 Balken N
05    01    9600 bps links

06    80    ???
06    40    Frequenz links Stelle 4 Balken A
06    20    Frequenz links Stelle 4 Balken K
06    10    Frequenz links Stelle 4 Balken B
06    08    Frequenz links Stelle 4 Balken M
06    04    Frequenz links Stelle 4 Balken C
06    02    S-Meter links Balken 5
06    01    Frequenz links Stelle 5 Balken F

07    80    ???
07    40    S-Meter links Balken 4
07    20    Frequenz links Stelle 4 Balken J+P
07    10    Frequenz links Stelle 4 Balken D
07    08    Frequenz links Stelle 4 Balken N
07    04    AM links
07    02    Frequenz links Stelle 4 Balken H
07    01    Frequenz links Stelle 4 Balken G

08    80    ???
08    40    ???
08    20    ???
```

```
08    10    ???
08    08    ???
08    04    Frequenz links Stelle 4 Balken F
08    02    Frequenz links Stelle 4 Balken E
08    01    Frequenz links Stelle 4 Balken Q

09    80    ???
09    40    Frequenz links Stelle 3 Balken K
09    20    Frequenz links Stelle 3 Balken B
09    10    Frequenz links Stelle 3 Balken M
09    08    Frequenz links Stelle 3 Balken C
09    04    S-Meter links Balken 3
09    02    ???
09    01    Dezimal-Punkt MHz links

10    80    ???
10    40    Frequenz links Stelle 3 Balken J+P
10    20    Frequenz links Stelle 3 Balken D
10    10    Frequenz links Stelle 3 Balken N
10    08    DCS links
10    04    Frequenz links Stelle 3 Balken H
10    02    Frequenz links Stelle 3 Balken G
10    01    Frequenz links Stelle 3 Balken A

11    80    ???
11    40    Frequenz links Stelle 2 Balken M
11    20    Frequenz links Stelle 2 Balken C
11    10    S-Meter links Balken 1
11    08    Frequenz links Stelle 3 Balken F
11    04    Frequenz links Stelle 3 Balken E
11    02    Frequenz links Stelle 3 Balken Q
11    01    S-Meter links Balken 2

12    80    ???
12    40    Frequenz links Stelle 2 Balken N
12    20    MUTE links
12    10    Frequenz links Stelle 2 Balken H
12    08    Frequenz links Stelle 2 Balken G
12    04    Frequenz links Stelle 2 Balken A
12    02    Frequenz links Stelle 2 Balken K
12    01    Frequenz links Stelle 2 Balken B

13    80    ???
13    40    Memory Tune links
13    20    Frequenz links Stelle 2 Balken F
13    10    Frequenz links Stelle 2 Balken E
13    08    Frequenz links Stelle 2 Balken Q
13    04    BUSY links
13    02    Frequenz links Stelle 2 Balken J+P
13    01    Frequenz links Stelle 2 Balken D
```

```
14    80    ???
14    40    Frequenz links Stelle 1 Balken H
14    20    Frequenz links Stelle 1 Balken G
14    10    Frequenz links Stelle 1 Balken A
14    08    Frequenz links Stelle 1 Balken K
14    04    Frequenz links Stelle 1 Balken B
14    02    Frequenz links Stelle 1 Balken M
14    01    Frequenz links Stelle 1 Balken C

15    80    ???
15    40    Frequenz links Stelle 1 Balken E
15    20    Frequenz links Stelle 1 Balken Q
15    10    ???
15    08    Frequenz links Stelle 1 Balken J+P
15    04    Frequenz links Stelle 1 Balken D
15    02    Frequenz links Stelle 1 Balken N
15    01    ???

16    80    ???
16    40    ???
16    20    ???
16    10    ???
16    08    ???
16    04    ???
16    02    ???
16    01    Frequenz links Stelle 1 Balken F

17    80    ???
17    40    Frequenz rechts Stelle 6 Balken M
17    20    Frequenz rechts Stelle 6 Balken C
17    10    S-Meter rechts Balken 9
17    08    Dezimal-Punkt kHz rechts
17    04    Middle TX Power rechts
17    02    ???
17    01    Kleine kHz 5 rechts

18    80    ???
18    40    Frequenz rechts Stelle 6 Balken N
18    20    Low TX Power rechts
18    10    Frequenz rechts Stelle 6 Balken H
18    08    Frequenz rechts Stelle 6 Balken G
18    04    Frequenz rechts Stelle 6 Balken A
18    02    Frequenz rechts Stelle 6 Balken K
18    01    Frequenz rechts Stelle 6 Balken B

19    80    ???
19    40    S-Meter rechts Balken 7
19    20    Frequenz rechts Stelle 6 Balken F
19    10    Frequenz rechts Stelle 6 Balken E
19    08    Frequenz rechts Stelle 6 Balken Q
19    04    S-Meter rechts Balken 8
```

```
19    02    Frequenz rechts Stelle 6 Balken J+P
19    01    Frequenz rechts Stelle 6 Balken D

20    80    ???
20    40    Frequenz rechts Stelle 5 Balken H
20    20    Frequenz rechts Stelle 5 Balken G
20    10    Frequenz rechts Stelle 5 Balken A
20    08    Frequenz rechts Stelle 5 Balken K
20    04    Frequenz rechts Stelle 5 Balken B
20    02    Frequenz rechts Stelle 5 Balken M
20    01    Frequenz rechts Stelle 5 Balken C

21    80    ???
21    40    Frequenz rechts Stelle 5 Balken E
21    20    Frequenz rechts Stelle 5 Balken Q
21    10    S-Meter rechts Balken 6
21    08    Frequenz rechts Stelle 5 Balken J+P
21    04    Frequenz rechts Stelle 5 Balken D
21    02    Frequenz rechts Stelle 5 Balken N
21    01    9600 bps rechts

22    80    ???
22    40    Frequenz rechts Stelle 4 Balken A
22    20    Frequenz rechts Stelle 4 Balken K
22    10    Frequenz rechts Stelle 4 Balken B
22    08    Frequenz rechts Stelle 4 Balken M
22    04    Frequenz rechts Stelle 4 Balken C
22    02    S-Meter rechts Balken 5
22    01    Frequenz rechts Stelle 5 Balken F

23    80    ???
23    40    S-Meter rechts Balken 4
23    20    Frequenz rechts Stelle 4 Balken J+P
23    10    Frequenz rechts Stelle 4 Balken D
23    08    Frequenz rechts Stelle 4 Balken N
23    04    DCS rechts
23    02    Frequenz rechts Stelle 4 Balken H
23    01    Frequenz rechts Stelle 4 Balken G

24    80    ???
24    40    ???
24    20    ???
24    10    ???
24    08    ???
24    04    Frequenz rechts Stelle 4 Balken F
24    02    Frequenz rechts Stelle 4 Balken E
24    01    Frequenz rechts Stelle 4 Balken Q

25    80    ???
25    40    Frequenz rechts Stelle 3 Balken K
25    20    Frequenz rechts Stelle 3 Balken B
```

```
25    10    Frequenz rechts Stelle 3 Balken M
25    08    Frequenz rechts Stelle 3 Balken C
25    04    S-Meter rechts Balken 3
25    02    ???
25    01    Dezimal-Punkt MHz rechts

26    80    ???
26    40    Frequenz rechts Stelle 3 Balken J+P
26    20    Frequenz rechts Stelle 3 Balken D
26    10    Frequenz rechts Stelle 3 Balken N
26    08    MUTE rechts
26    04    Frequenz rechts Stelle 3 Balken H
26    02    Frequenz rechts Stelle 3 Balken G
26    01    Frequenz rechts Stelle 3 Balken A

27    80    ???
27    40    Frequenz rechts Stelle 2 Balken M
27    20    Frequenz rechts Stelle 2 Balken C
27    10    S-Meter rechts Balken 1
27    08    Frequenz rechts Stelle 3 Balken F
27    04    Frequenz rechts Stelle 3 Balken E
27    02    Frequenz rechts Stelle 3 Balken Q
27    01    S-Meter rechts Balken 2

28    80    ???
28    40    Frequenz rechts Stelle 2 Balken N
28    20    Memory Tune rechts
28    10    Frequenz rechts Stelle 2 Balken H
28    08    Frequenz rechts Stelle 2 Balken G
28    04    Frequenz rechts Stelle 2 Balken A
28    02    Frequenz rechts Stelle 2 Balken K
28    01    Frequenz rechts Stelle 2 Balken B

29    80    ???
29    40    KEY2
29    20    Frequenz rechts Stelle 2 Balken F
29    10    Frequenz rechts Stelle 2 Balken E
29    08    Frequenz rechts Stelle 2 Balken Q
29    04    BUSY rechts
29    02    Frequenz rechts Stelle 2 Balken J+P
29    01    Frequenz rechts Stelle 2 Balken D

30    80    ???
30    40    Frequenz rechts Stelle 1 Balken H
30    20    Frequenz rechts Stelle 1 Balken G
30    10    Frequenz rechts Stelle 1 Balken A
30    08    Frequenz rechts Stelle 1 Balken K
30    04    Frequenz rechts Stelle 1 Balken B
30    02    Frequenz rechts Stelle 1 Balken M
30    01    Frequenz rechts Stelle 1 Balken C
```

```
31    80    ???
31    40    Frequenz rechts Stelle 1 Balken E
31    20    Frequenz rechts Stelle 1 Balken Q
31    10    Keypad Lock
31    08    Frequenz rechts Stelle 1 Balken J+P
31    04    Frequenz rechts Stelle 1 Balken D
31    02    Frequenz rechts Stelle 1 Balken N
31    01    SET

32    80    ???
32    40    ???
32    20    ???
32    10    ???
32    08    ???
32    04    ???
32    02    ???
32    01    Frequenz rechts Stelle 1 Balken F

33    80    ???
33    40    Preferential Memory rechts
33    20    DEC rechts
33    10    ENC rechts
33    08    Negativ rechts
33    04    Positiv rechts
33    02    TX rechts
33    01    MAIN rechts

34    80    ???
34    40    Kanal rechts Stelle 3 Balken F
34    20    Kanal rechts Stelle 3 Balken E
34    10    Kanal rechts Stelle 3 Balken A
34    08    Kanal rechts Stelle 3 Balken G
34    04    Kanal rechts Stelle 3 Balken B
34    02    Kanal rechts Stelle 3 Balken C
34    01    SKIP rechts

35    80    ???
35    40    Kanal rechts Stelle 2 Balken E
35    20    Kanal rechts Stelle 2 Balken A
35    10    Kanal rechts Stelle 2 Balken G
35    08    Kanal rechts Stelle 2 Balken B
35    04    Kanal rechts Stelle 2 Balken C
35    02    Kanal rechts Stelle 1 Balken D
35    01    Kanal rechts Stelle 3 Balken D

36    80    ???
36    40    Kanal rechts Stelle 1 Balken A
36    20    Kanal rechts Stelle 1 Balken G
36    10    Kanal rechts Stelle 1 Balken B
36    08    Kanal rechts Stelle 1 Balken C
36    04    Kanal rechts Trennstrich
```

```
36    02    Kanal rechts Stelle 2 Balken D
36    01    Kanal rechts Stelle 2 Balken F

37    80    ???
37    40    ENC links
37    20    Negativ links
37    10    Positiv links
37    08    TX links
37    04    MAIN links
37    02    Kanal rechts Stelle 1 Balken F
37    01    Kanal rechts Stelle 1 Balken E


38    80    ???
38    40    Kanal links Stelle 3 Balken A
38    20    Kanal links Stelle 3 Balken G
38    10    Kanal links Stelle 3 Balken B
38    08    Kanal links Stelle 3 Balken C
38    04    SKIP links
38    02    Preferential Memory links
38    01    DEC links

39    80    ???
39    40    ???
39    20    Kanal links Stelle 2 Balken B
39    10    Kanal links Stelle 2 Balken C
39    08    Kanal links Stelle 1 Balken D
39    04    Kanal links Stelle 3 Balken D
39    02    Kanal links Stelle 3 Balken F
39    01    Kanal links Stelle 3 Balken E

40    80    ???
40    40    Kanal links Stelle 2 Balken F
40    20    Kanal links Stelle 2 Balken E
40    10    Kanal links Stelle 2 Balken A
40    08    Kanal links Stelle 2 Balken G
40    04    ???
40    02    ???
40    01    ???

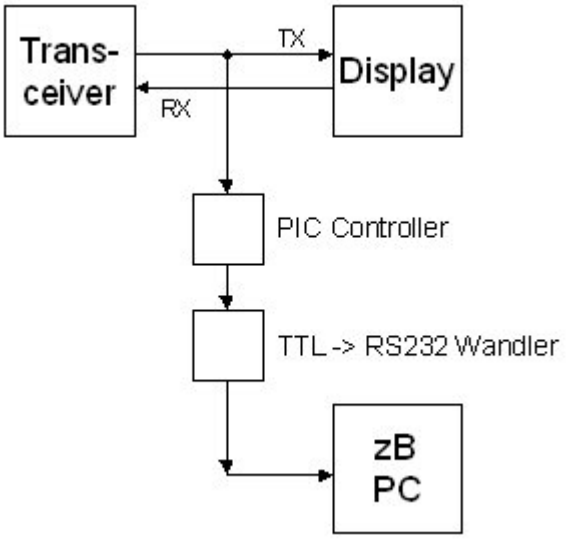
41    80    ???
41    40    Kanal links Stelle 1 Balken E
41    20    Kanal links Stelle 1 Balken A
41    10    Kanal links Stelle 1 Balken G
41    08    Kanal links Stelle 1 Balken B
41    04    Kanal links Stelle 1 Balken C
41    02    Kanal links Trennstrich
41    01    Kanal links Stelle 2 Balken D

42    80    ???
42    40    ???
```

```
42    20    ???
42    10    ???
42    08    ##
42    04    ## Helligkeit Hintergrundbeleuchtung
42    02    ##
42    01    Kanal links Stelle 1 Balken F

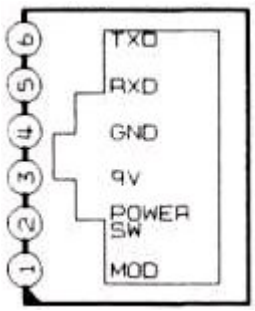
Helligkeit Hintergrundbeleuchtung:
Aus : xxxx000x
DIM1: xxxx111x
DIM2: xxxx011x
DIM3: xxxx100x
```

Für eine Spezial-Anwendung mussten die Display-Anzeigen incl. dem S-Meter und dem Busy-Flag vom YAESU FT-8800 ausgelesen und in ein „lesbares“ ASCII Protokoll umgewandelt werden.



Dafür benutze ich wie immer einen 16F84A PIC Microcontroller. Damit alles in 1k Speicher passt, mussten einige Tricks genutzt werden um z.B. die Umrechnung der Einzel-Segmente in ASCII-Zeichen Speicher-Sparend vorzunehmen.

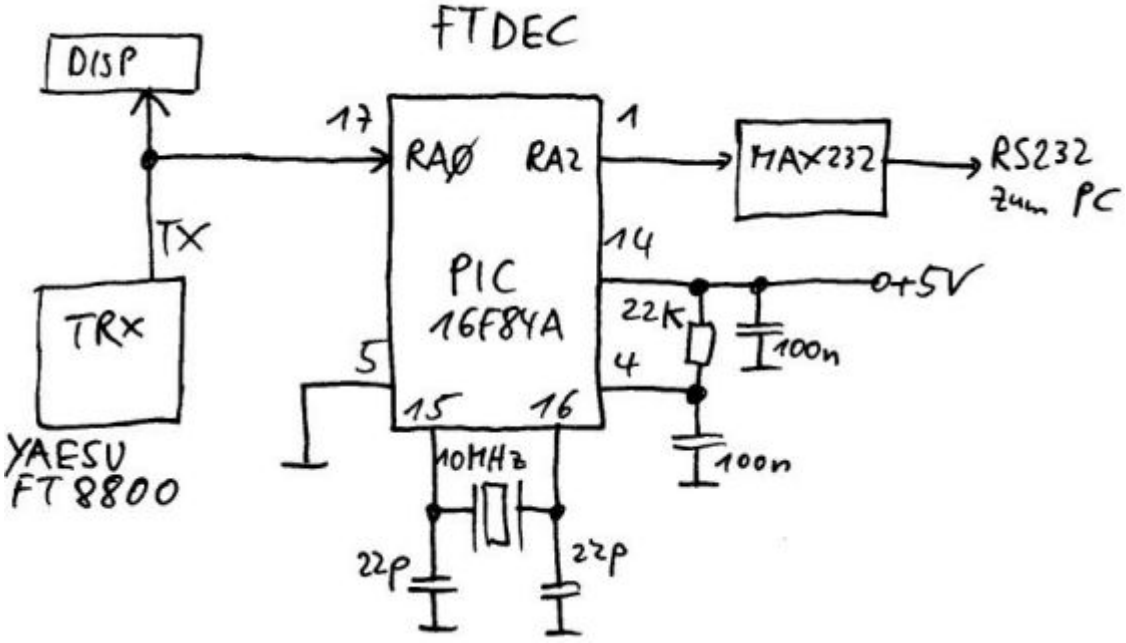
Die Daten bekommen wir aus der Verbindung zwischen Display und Transceiver. Die 6polige Buchse am Funkgerät ist wie folgt belegt:



Für unseren Fall sind nur GND und TXD interessant. Hier wird mit dem Protokoll der RS232 gefahren, allerdings nur mit TTL-Leveln. Die Datenrate beträgt 19200 Baud, 8n1.

Zur Umsetzung Richtung PC benötigt man nun noch einen TTL zu RS232-Wandler z.B. mit MAX232 o.ä. um folgende Logik zu erreichen : +5V TTL⇒ -15V RS232, 0V TTL ⇒ +15V RS232.

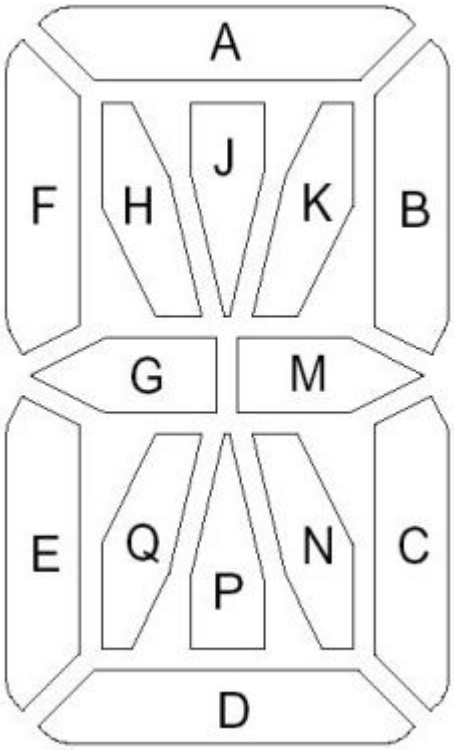
Der Schaltplan ist schnell gemalt, da ist nicht viel dran. Der PIC läuft mit 10 MHz um genug Zeit für die Dekodierung der 19200 Baud zu haben.



Ablauf in der Software :

Zuerst sucht sich der PIC eine Lücke im Protokoll und liest dann 32 Bytes ein. Der Rest bis zum Frameende (42 Bytes ingesamt) bleibt ungenutzt.

Die ersten 16 Bytes werden dann nach folgenden Informationen untersucht : S-Meter / Busy-Flag / MHz-Punkt / kHz-Punkt / 5kHz-Anzeige Die Informationen werden dann in einem Register zwischengespeichert.



Nun werden alle einzelnen Segmente (pro Segment 3 Byte) aus dem Frame ausgeschnitten und entsprechend der Abfolge M C N H G A K B F E Q J D in jeweils 2 Byte (MSB und LSB) abgespeichert.

Damit liegen nun die ersten 12 Bytes im Speicher, die die ersten 6 Zeichen des Displays darstellen.

Nun kommt der aufwendigste Teil. Aus den 2 Bytes muss die Information gewonnen werden, welches Zeichen gerade angezeigt wird und wie der dazugehörige ASCII-Code lautet.

Dazu wird im MSB die Unterscheidung unternommen ob es eine einfache Abfrage, eine Doppel-Abfrage oder eine Tabelle wird.

In den entsprechenden Unterprogrammen wird dann das LSB weiterverarbeitet und der ASCII-Code gewonnen.

Folgende Tabelle ist im PIC abgelegt (zeigt auch gleich, welche Zeichen überhaupt angezeigt werden können!): Tabelle Display-Segmente Dekodierung :

ftdec_table.pdf

Sind die ASCII-Codes gewonnen wird zunächst der String für das linke Display ausgeben nach folgendem Muster :

```
[Stelle1] [Stelle2] [Stelle3]  +
[MHz-Punkt*]  +
[Stelle4] [Stelle5] [Stelle6]  +
[kHz-Punkt*] [5kHz*]  +
[SPACE] + [S] + [0..9] +
[! fuer Busy]

*falls nötig
```

Das ganze wird dann noch für die rechte Display Seite wiederholt und mit CR+LF abgeschlossen.

Dargestellt werden also alle 6 Stellen mit Buchstaben und Zahlen, korrekte MHz und kHz Punkte sowie die 5kHz Stelle. Dazu wird noch das S-Meter ausgelesen und angezeigt ob die Rauschsperre gerade geöffnet ist oder nicht.

Anbei ein Beispiel einer Display-Anzeige mit Zeichen links und Zahlen rechts : FTDEC Logfile

```
PFORZH  S8!  432.700 S0
PFORZH  S8!  432.700 S0
PFORZH  S8!  432.700 S0
PFORZH  S8!  432.700 S0
PFORZH  S8!  432.700 S0
PFORZH  S8!  432.700 S0
```

Source-Code und HEX-File gibts hier : PIC Inhalt Display Decoder :

ftdec_source.zip

Der PIC ist zu ca. 80% ausgelastet, hier ist also noch Platz für Erweiterungen. Allerdings wurden die 8 Stack-Level komplett ausgereizt, also Vorsicht beim „einziehen“ von weiteren CALL / RETURN Ebenen.

From:
https://dg1sfj.de/ - **dg1sfj.de**

Permanent link:
https://dg1sfj.de/doku.php?id=funk:geraete:yaesuft8800

Last update: **2025/01/19 15:30**