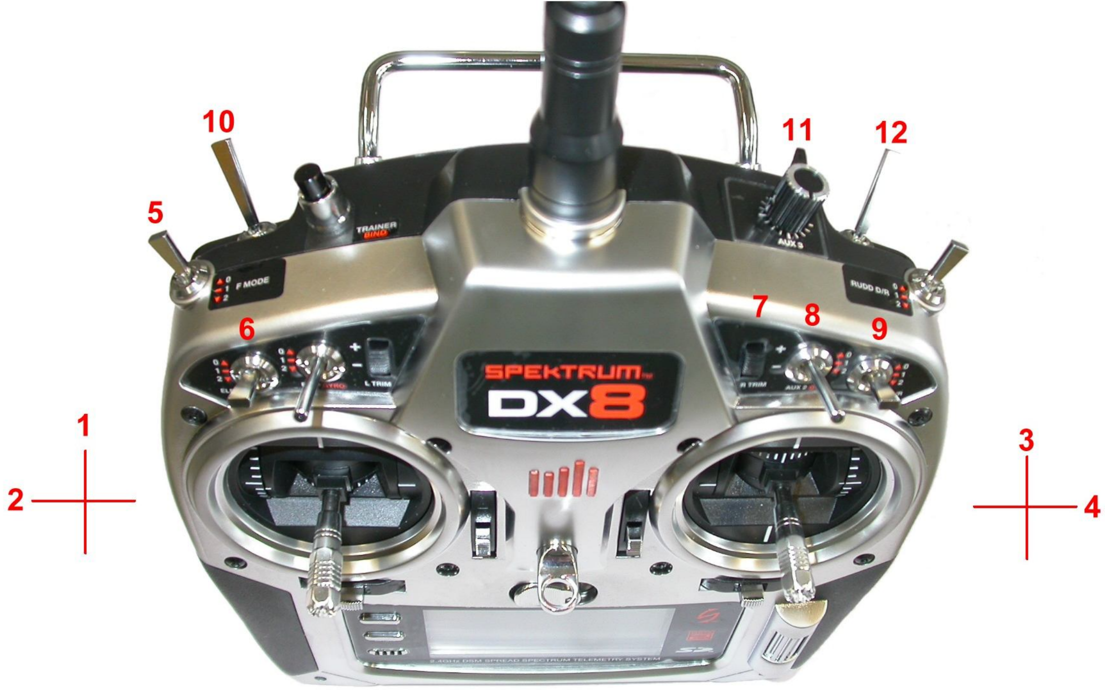
SEXPANDER – A small Spektrum DX8 Channel Expander

A normal DX8 (with the little Satellite Receiver) can only send on 8 Channels (4 Channels for Throttle, Yaw, Nick & Roll), 3 Switch-Channels and a poti. There are much more switches, but their output is not send to the air
I build this little “Modulator” which reads these additional switches and transmits them over a single channel which is normally used for the Flight-Mode switch. On FlightControl this signal will be decoded “on the fly” using a firmware patch.
So you get 12 (twelve!) fully usable channels!

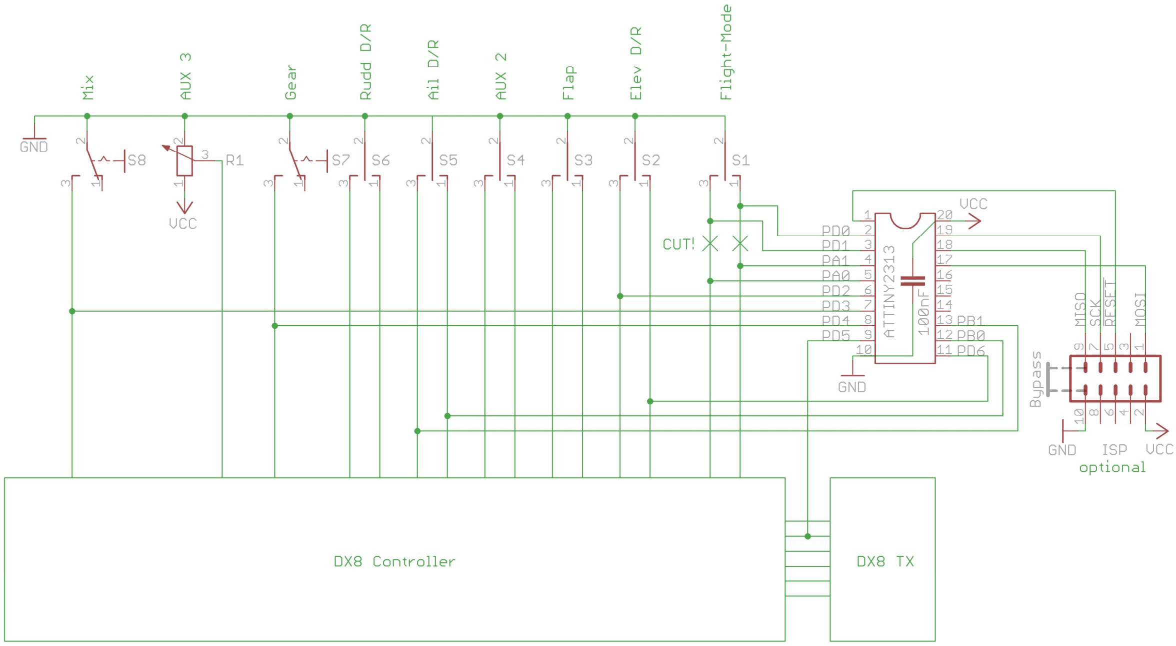
The data uplink runs at ~80 bit/sec (including Sync- and Parity-Bit). The response time is between 100 and 150 ms which is totally fast enough for a switch.
Using these additional switches is not destructive, that means you can use them in DX8 itself, such as mixing Elevator on Throttle for example.

If you want to use your DX8 temporarily in normal mode (un-modded), just use a Jumper between GND and MISO (Pin 9+10) on the ISP-Connector (or if you don't have a ISP-Connector, pull down Pin 18 of Tiny2313 directly to GND - via a Switch or something).

For easier configuration I changed the channel assignment as follows:

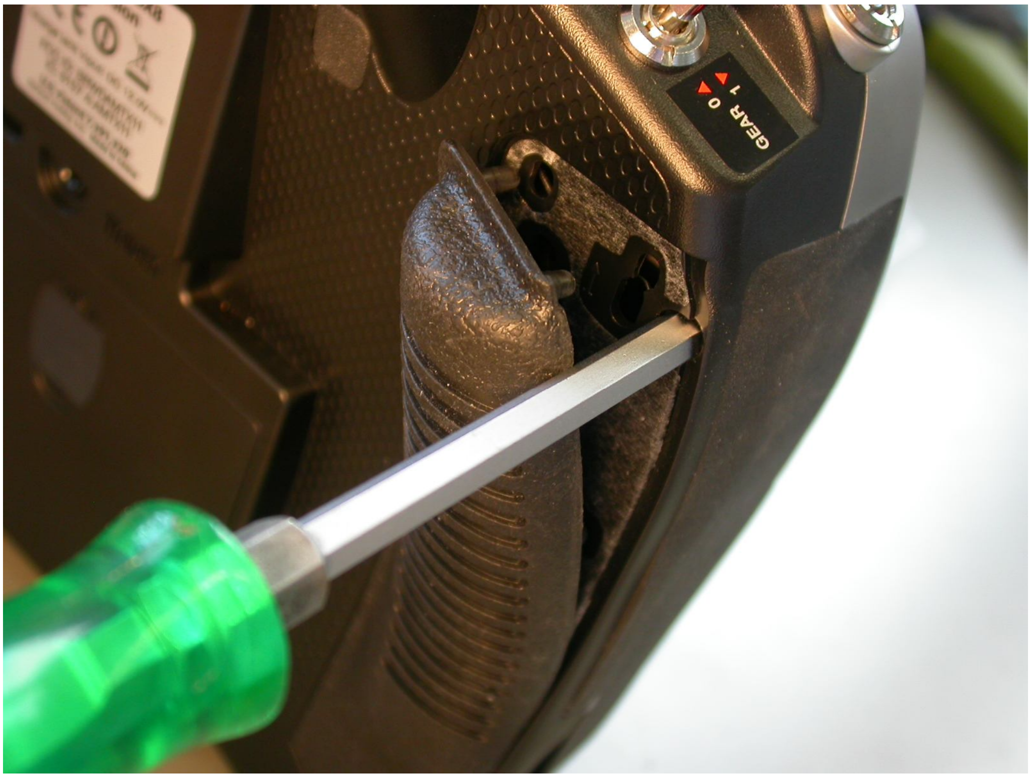


The schematic itself is very simple as you see. The only thing needed is an Atmel ATtiny2313, a capacitor and a few wires. The ISP connector (3,3 Volts!) is only needed once for programming the code when you don’t have a parallel programmer. It's hardware-compatible to the old DX7-Sexpander!

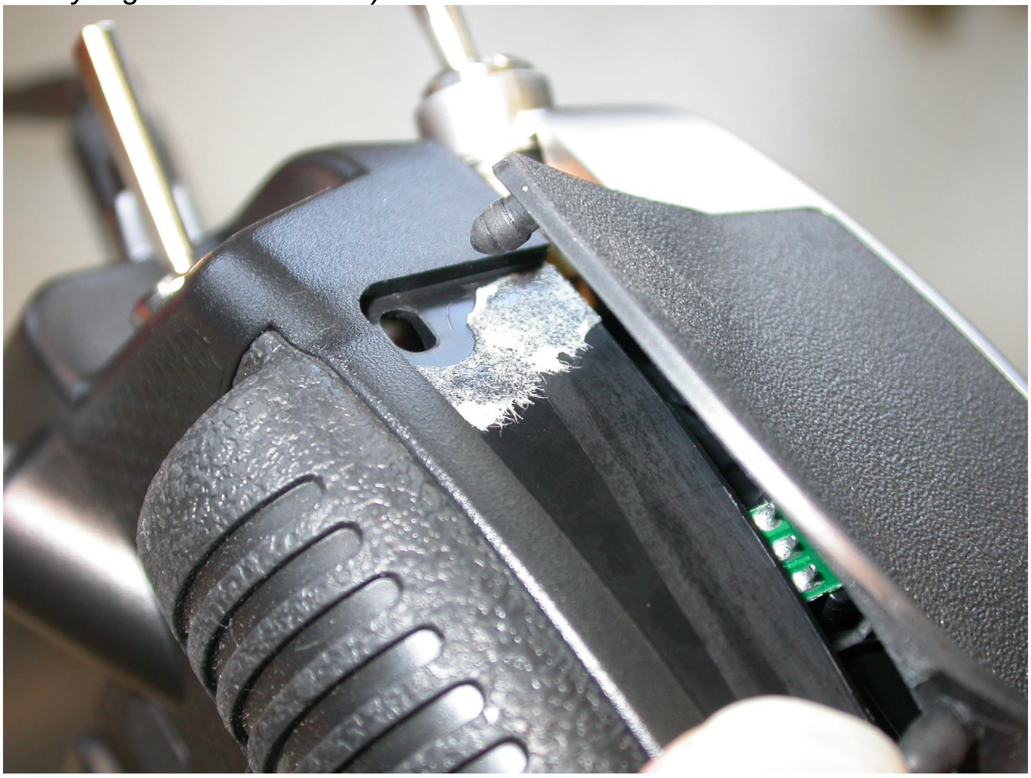


If you want to build your own, take these pictures as help:

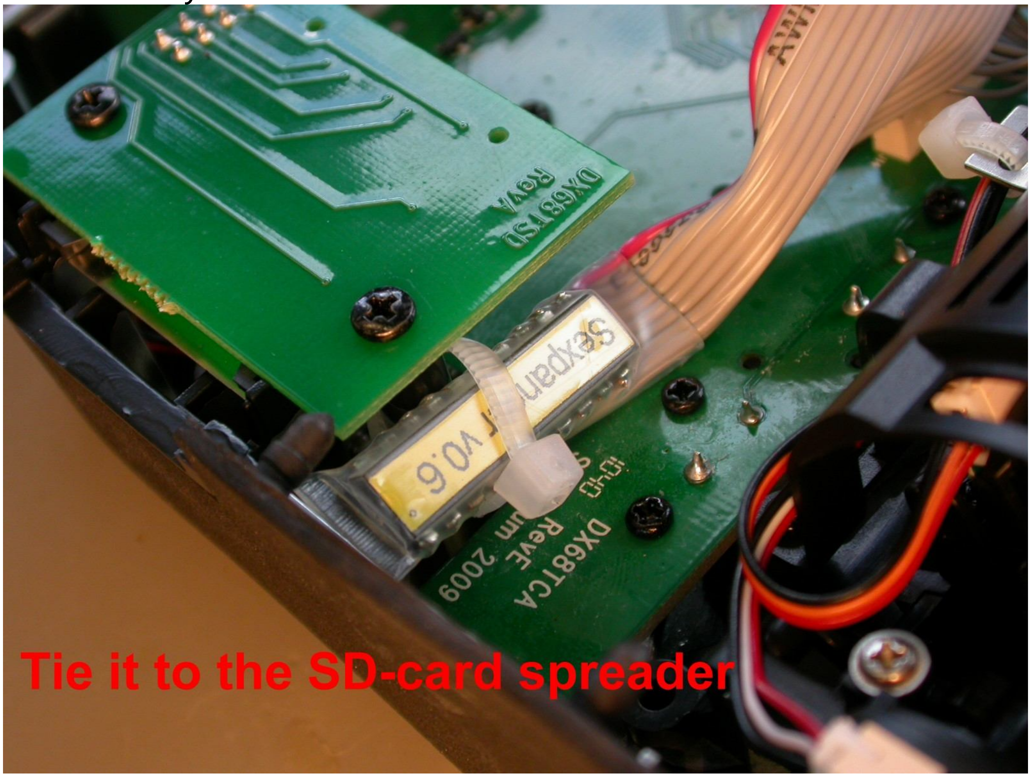
You have to lift up the gum back-things a bit, to get to screws #5 and #6 on both sides:



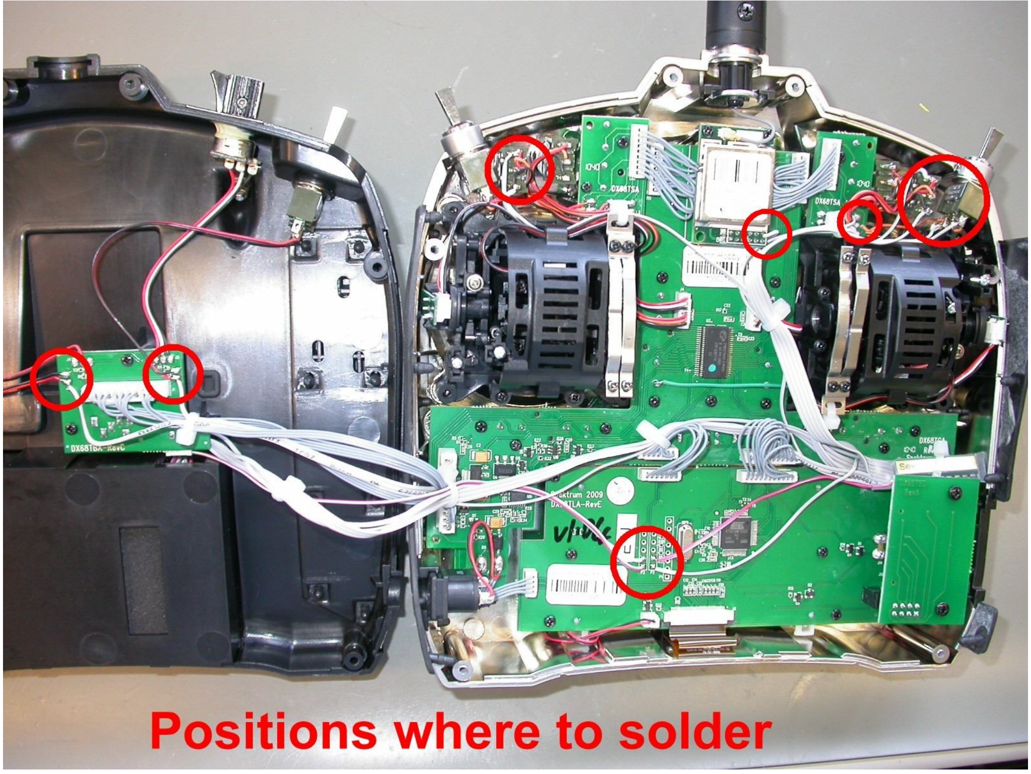
The gum side-things have to be released also a bit: (it's no problem to get them nicely together afterwards)



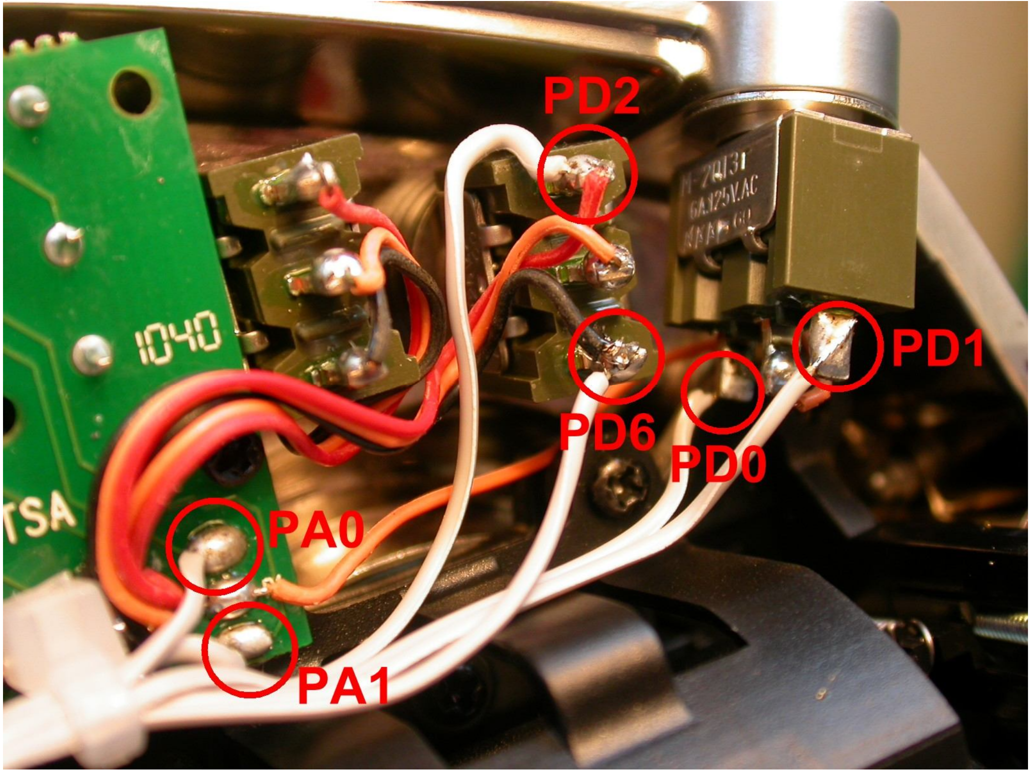
Fix the ATtiny first:

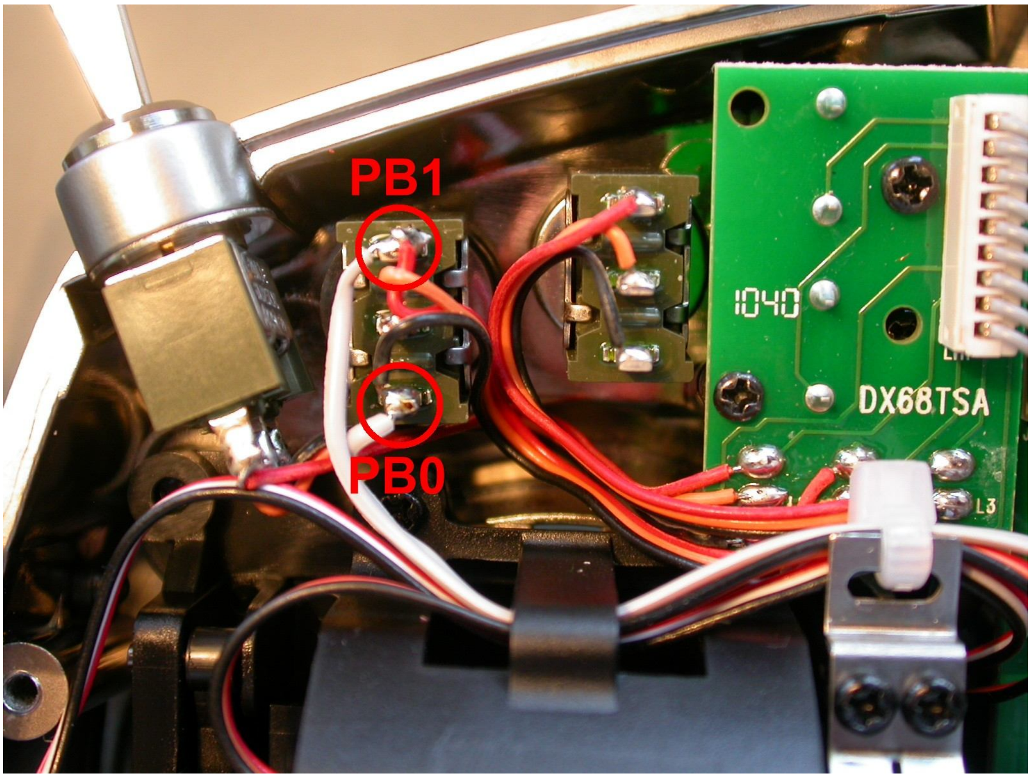


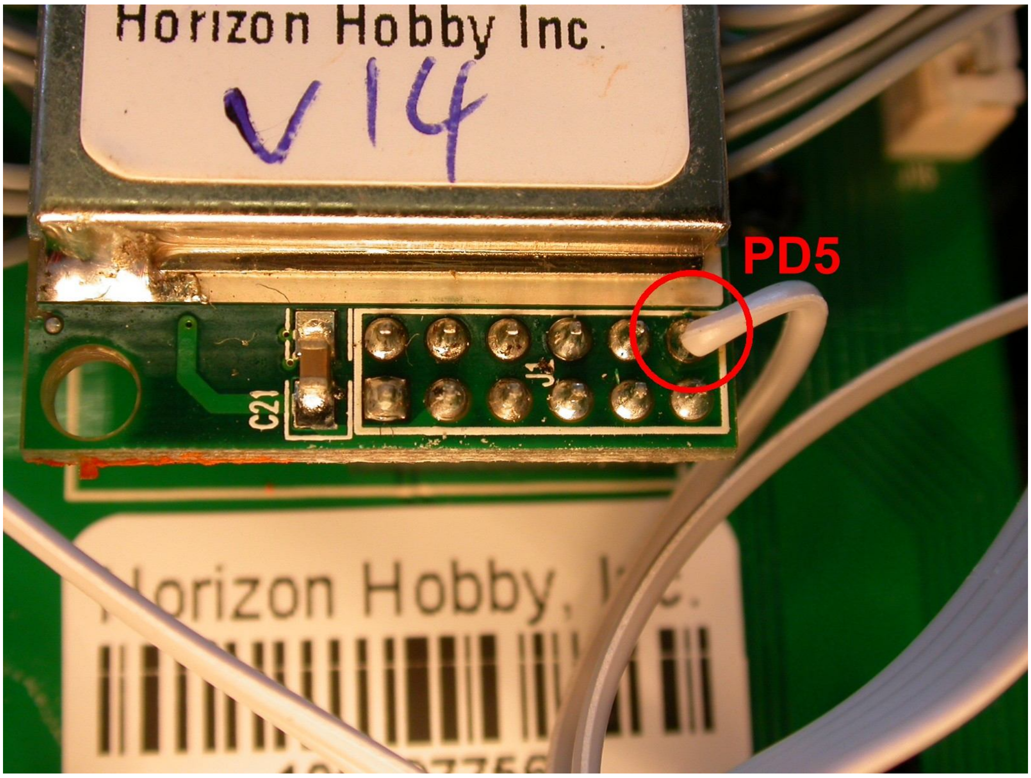
You have to solder at this points:

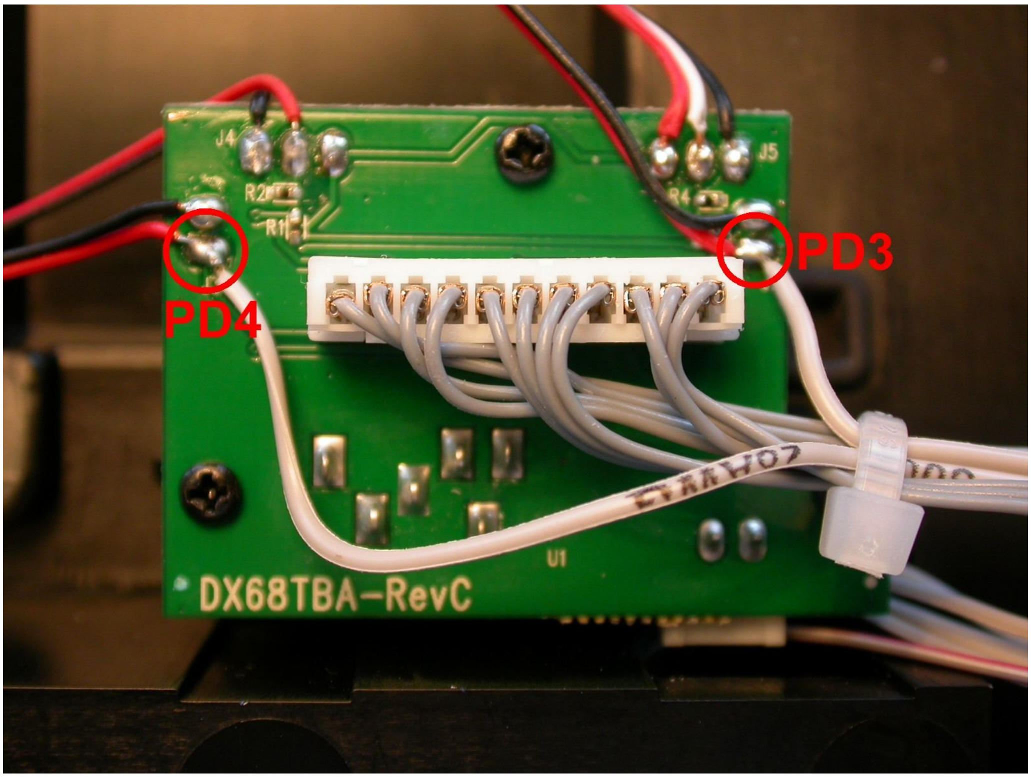


Now you connect the wires to the described pins:
(Between points PD0 and PA1, and also between PD1 and PA0 are original wires - you have to remove both first.)

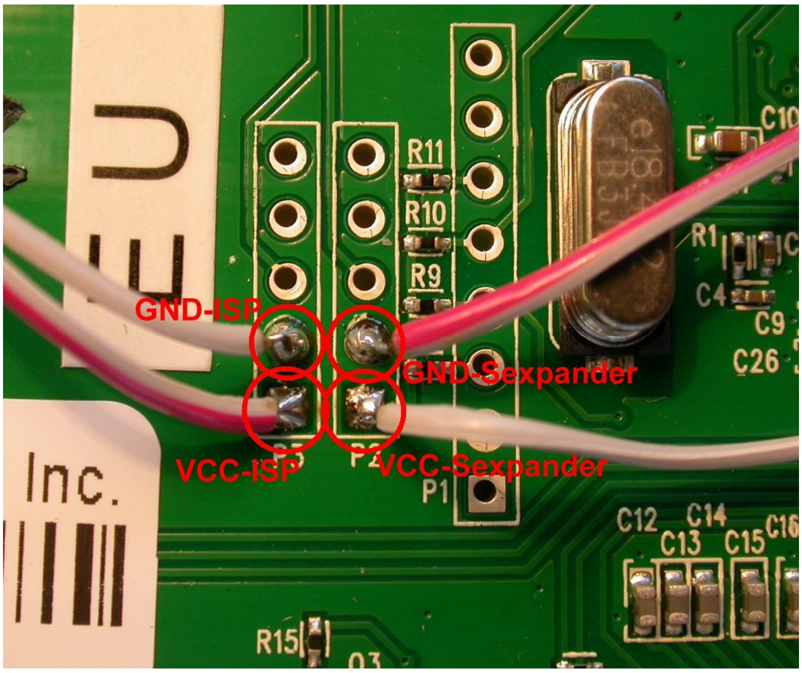



PB1
PB0
1040
DX68TSA
L3

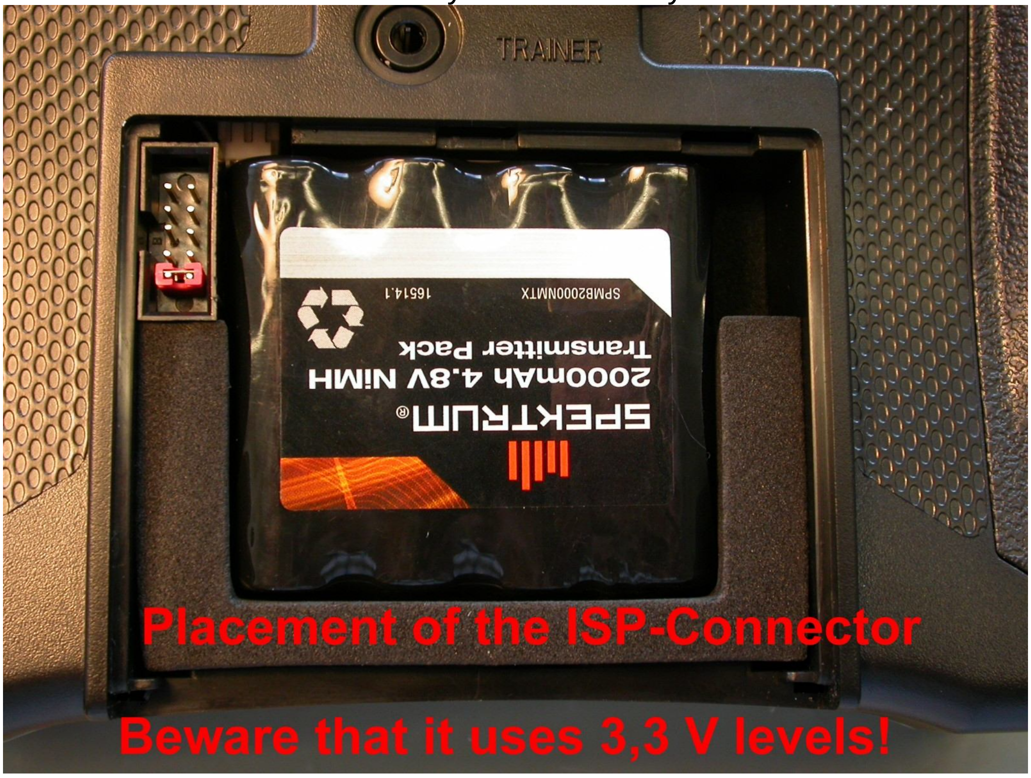

Horizon Hobby Inc.
V14
PD5
C21
J1
Horizon Hobby, Inc.



You can grab the needed power (3,3 Volts) for the ATtiny and for the optional ISP-Port here:



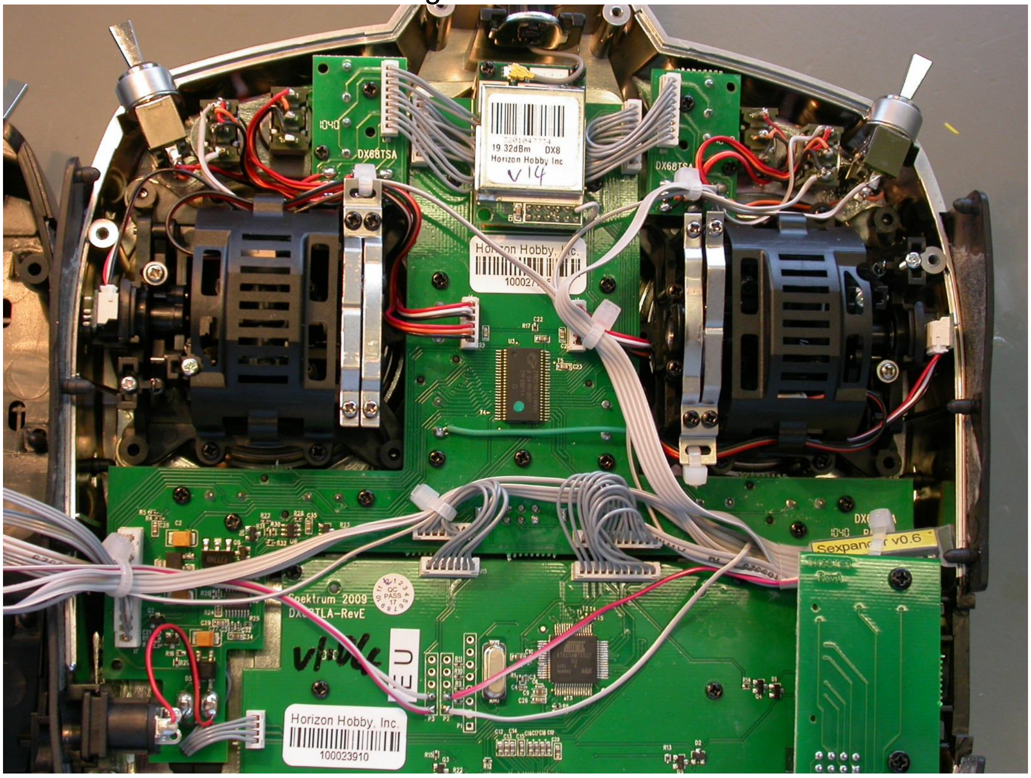
The ISP-Port can be fitted nicely into the battery-chamber:



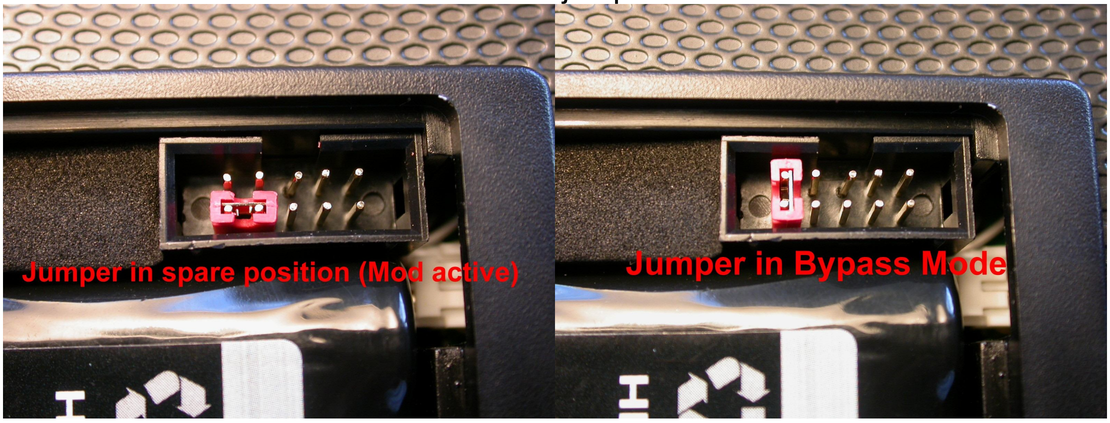
Placement of the ISP-Connector

Beware that it uses 3,3 V levels!

Take care that no cables can get into the sticks!

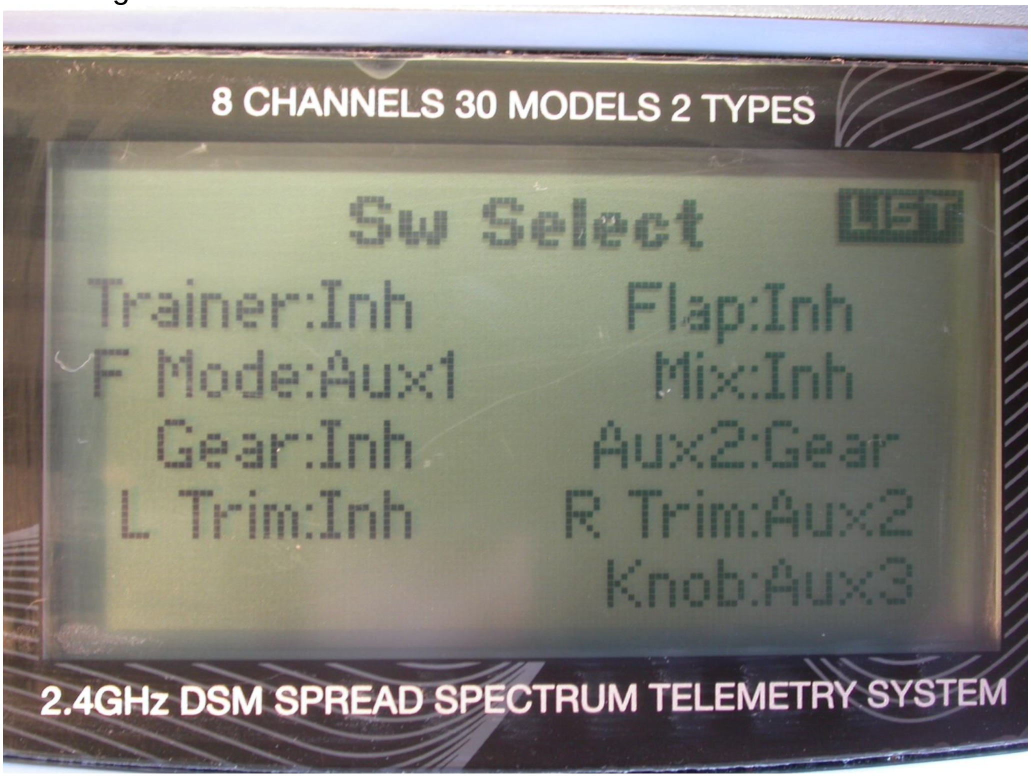


You can enable/disable the mod with a jumper:



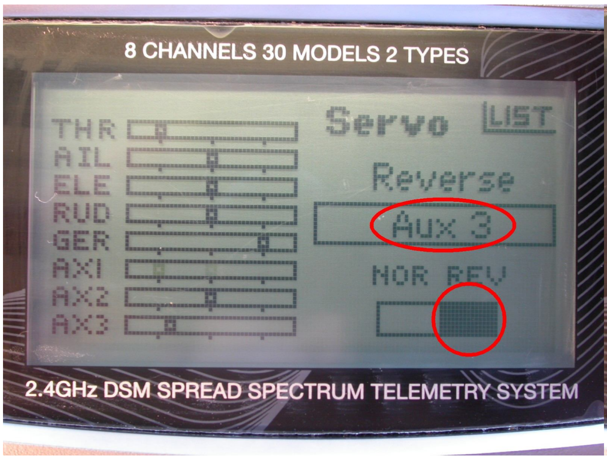
Use this settings in “System Menu” (hold down scrollwheel while power on):

Re-assign some channels:


8 CHANNELS 30 MODELS 2 TYPES
2.4GHz DSM SPREAD SPECTRUM TELEMETRY SYSTEM

```
         Sw Select                LIST
Trainer:Inh              Flap:Inh
F Mode:Aux1               Mix:Inh
   Gear:Inh             Aux2:Gear
 L Trim:Inh           R Trim:Aux2
                        Knob:Aux3
```

Reverse AUX3 and GEAR!

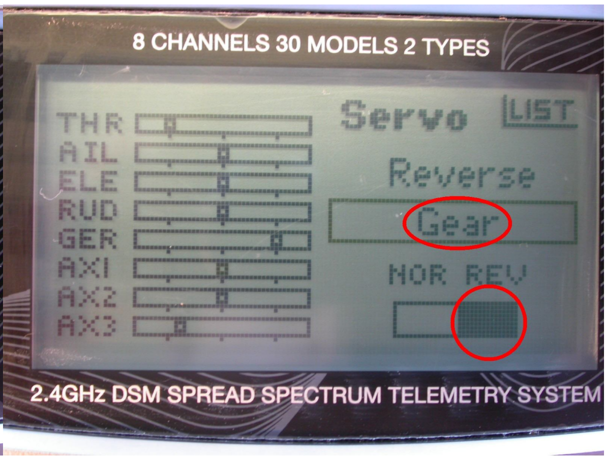

8 CHANNELS 30 MODELS 2 TYPES
THR
AIL
ELE
RUD
GER
AX1
AX2
AX3
2.4GHz DSM SPREAD SPECTRUM TELEMETRY SYSTEM

Servo LIST

Reverse

Aux 3

NOR REV


8 CHANNELS 30 MODELS 2 TYPES
THR
AIL
ELE
RUD
GER
AX1
AX2
AX3
2.4GHz DSM SPREAD SPECTRUM TELEMETRY SYSTEM

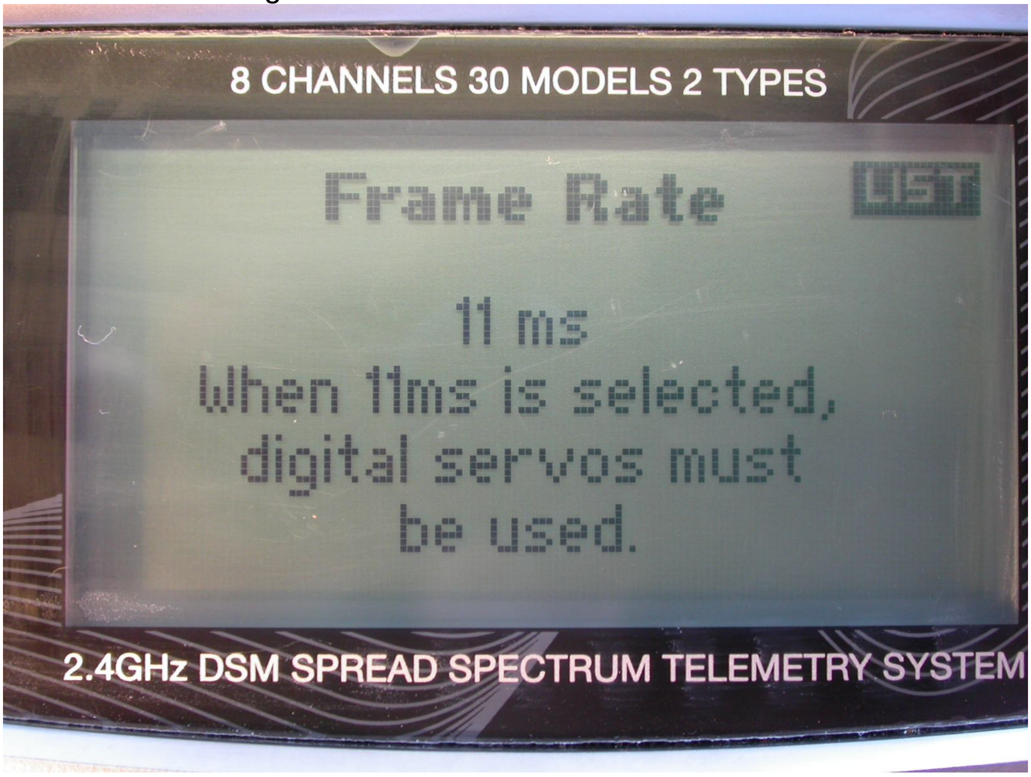
Servo LIST

Reverse

Gear

NOR REV

When some problem occurs, check that the frame-rate equals the rate which was shown after binding.


8 CHANNELS 30 MODELS 2 TYPES
2.4GHz DSM SPREAD SPECTRUM TELEMETRY SYSTEM

Frame Rate LIST

11 ms

When 11ms is selected, digital servos must be used.

You can use -THIS FILE- to import all necessary settings via SD-card to your DX8.

Telemetry Module TM1000

You can use the telemetry module as well (without AR8000 main-receiver).
But you will only see voltage, temperature and optional the RPM-Input. There are no receiving-stats because you don't fly with the AR8000.
The Module just needs some voltage on the Data-Connector (same voltage range like the AR8000 itself). You can connect it easily to the Spektrum Diversity Module with a normal satellite-cable for getting power.
Firstly put the satellite(s) into bind mode and power up your TM1000 while pressing the small bind button.
After satellites and telemetry module are in bind mode, start your DX8 with pressed bind button.

Notes

- Do not reverse the “Flight-Mode” channel – That’s our data-uplink.
- FlightControl firmware patches are available for several versions. But be sure to use the old WinAVR-20060421 compiler, which does not have the performance-problem like newer versions. (For FC 2.1 you additionally need the MK-patch for WinAVR)
- Precompiled builds are available in /SVN/Projects/Spektrum-Expander/
- When using an old satellite receiver (like from DX7), the analogue "Aux 3" channel is dead. (11 channels only)
- If you want to reverse some of the "emulated" channels, you can change the sourcecode or you simply reverse the wire(s) on the switch. (additional wires only)

Flashing the code to the tiny2313

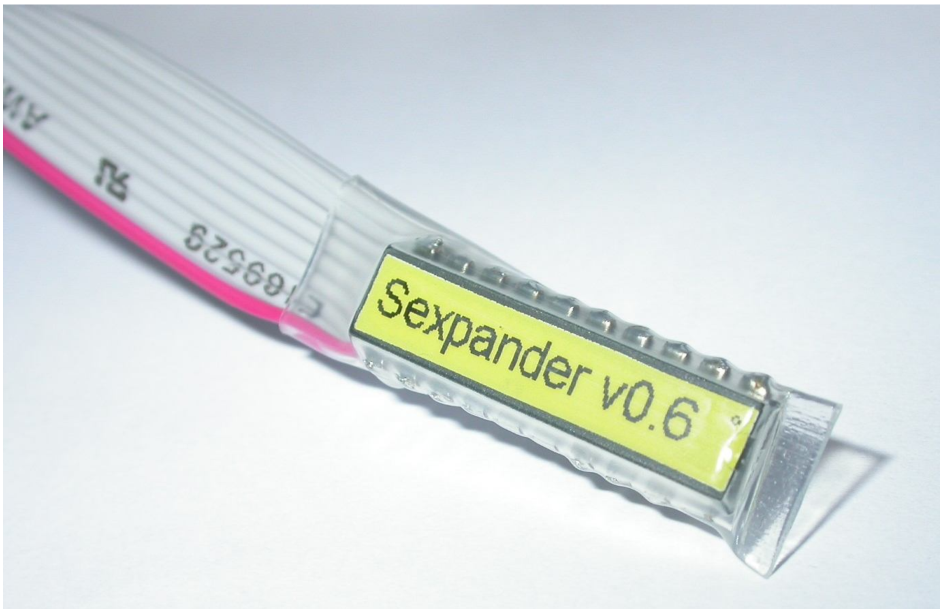
You can use a ISP-programmer to flash the code. The Fuse-Bits stay at default (1 MHz internal RC-Oscillator)
I used avrdude to flash code+fuses at once:

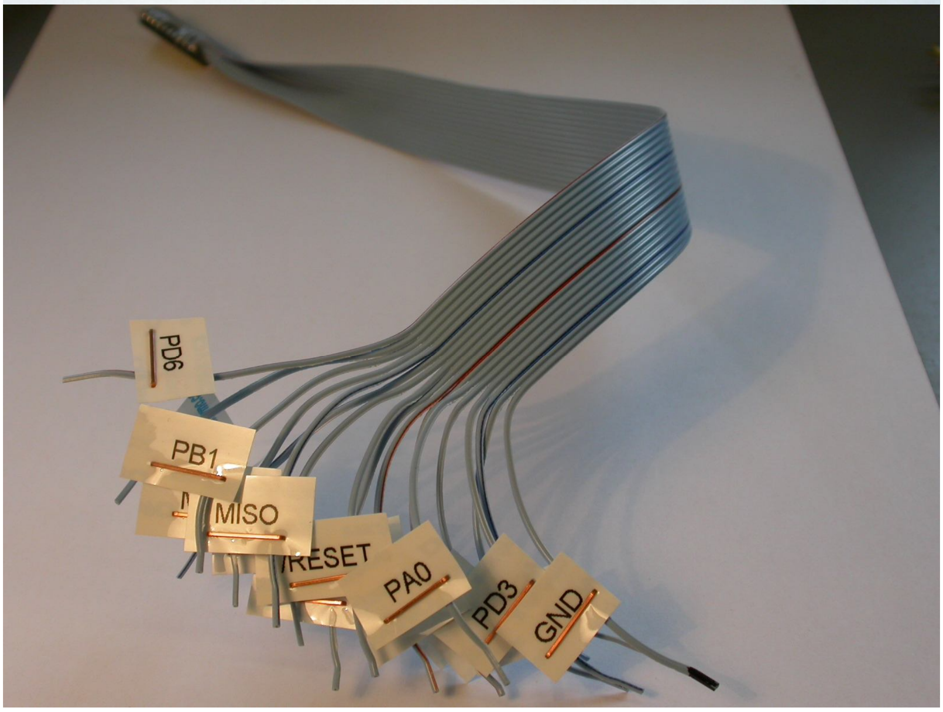
```
avrdude -p t2313 -P COM1 -c ponyser -U lfuse:w:0x64:m -U hfuse:w:0xDF:m -U efuse:w:0xFF:m -U flash:w:sexpander.hex
```

Be sure that your programmer supports 3,3 Volt levels when you flash him in the DX8!

You can DOWNLOAD code, patch, and precompiled FC firmwares from the SVN-Repository.

Now for sale





Flashed and soldered ATtiny2313 with long and labeled breakout cable for easy identify before cut-off to needed length.

Complete Set (like on pictures + Jumper + ISP-Connector): **EUR 20.-** (USD $29) (including worldwide shipping)
Single flashed ATtiny2313: **EUR 8.-** (USD $12) (including worldwide shipping)

Thank you for your attention.