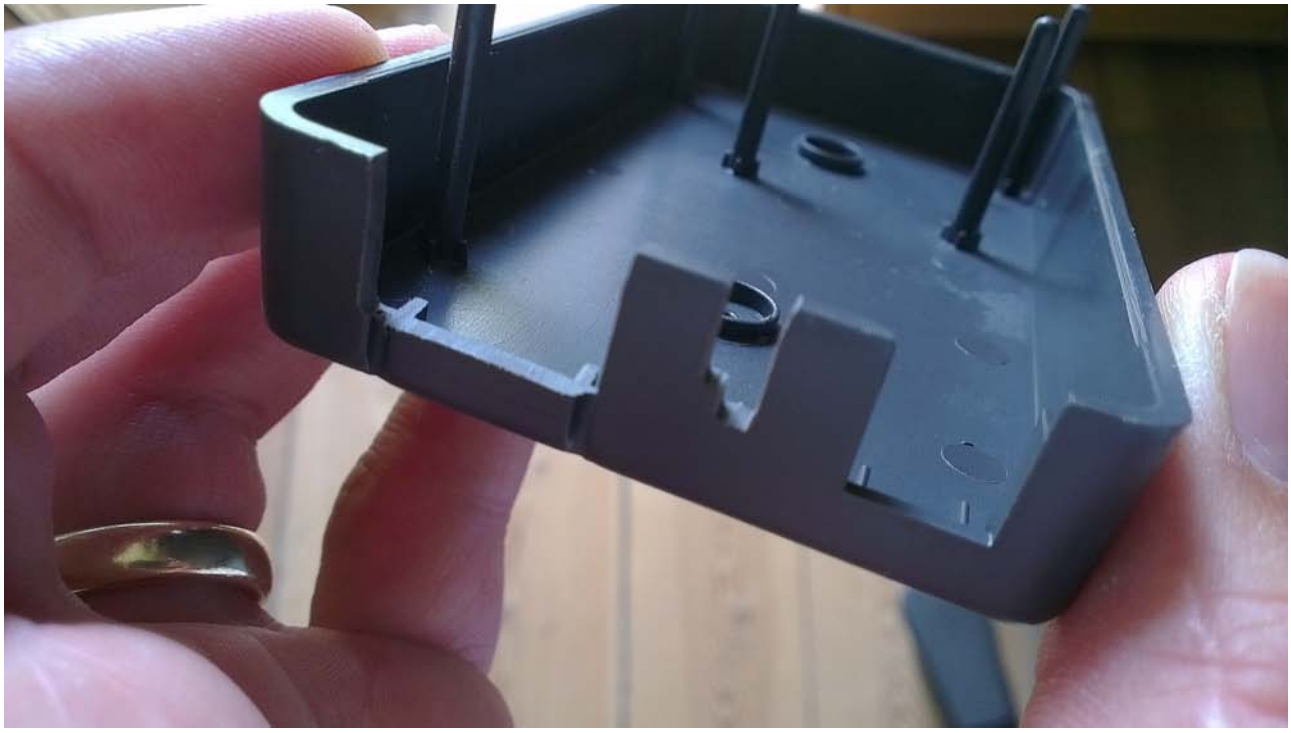
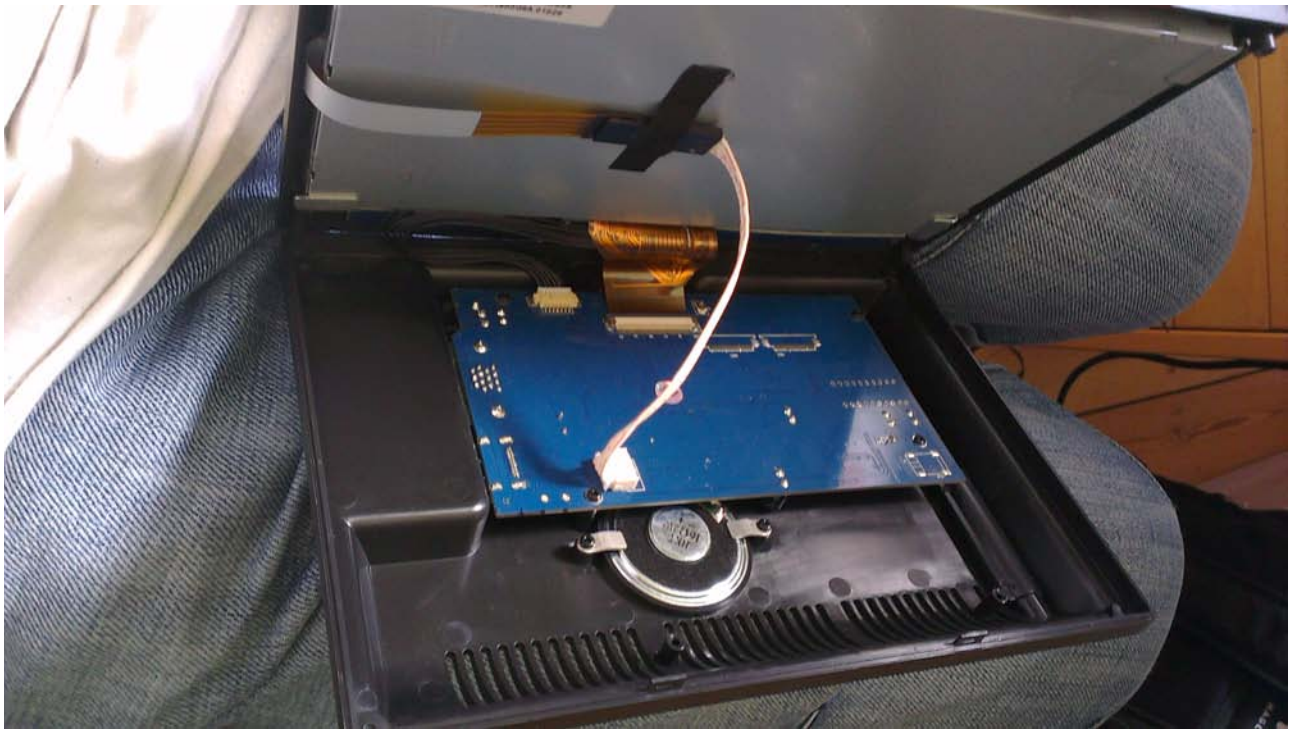
DIY instructions for the Arduino/ATEM Touchscreen

See http://skaarhoj.com/wiki/index.php/Touchscreen_for_Arduino/ATEM

Comments are under the pictures

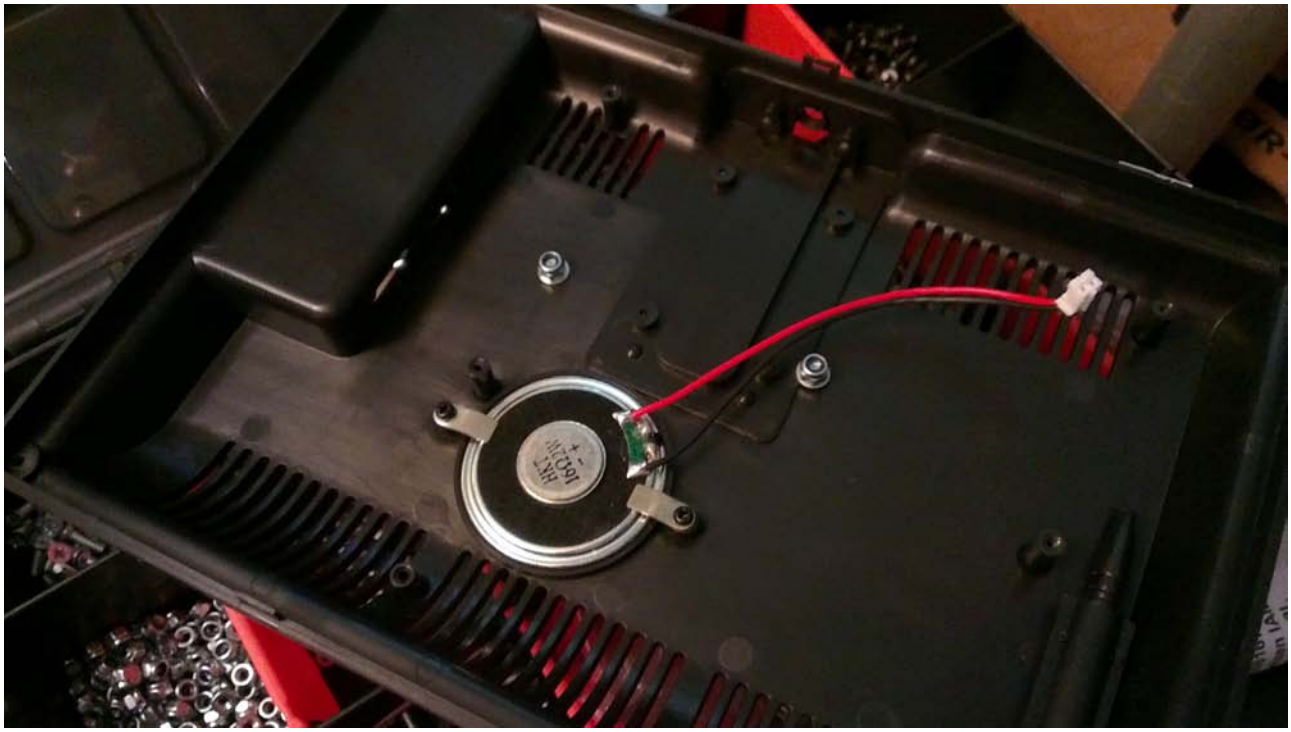


You need to modify the Arduino enclosure a bit so the **slot for the USB plug can host the Ethernet jack** on an Arduino Ethernet module. Use a sharp knive, a saw or something else, just keep your fingers safe in the process.



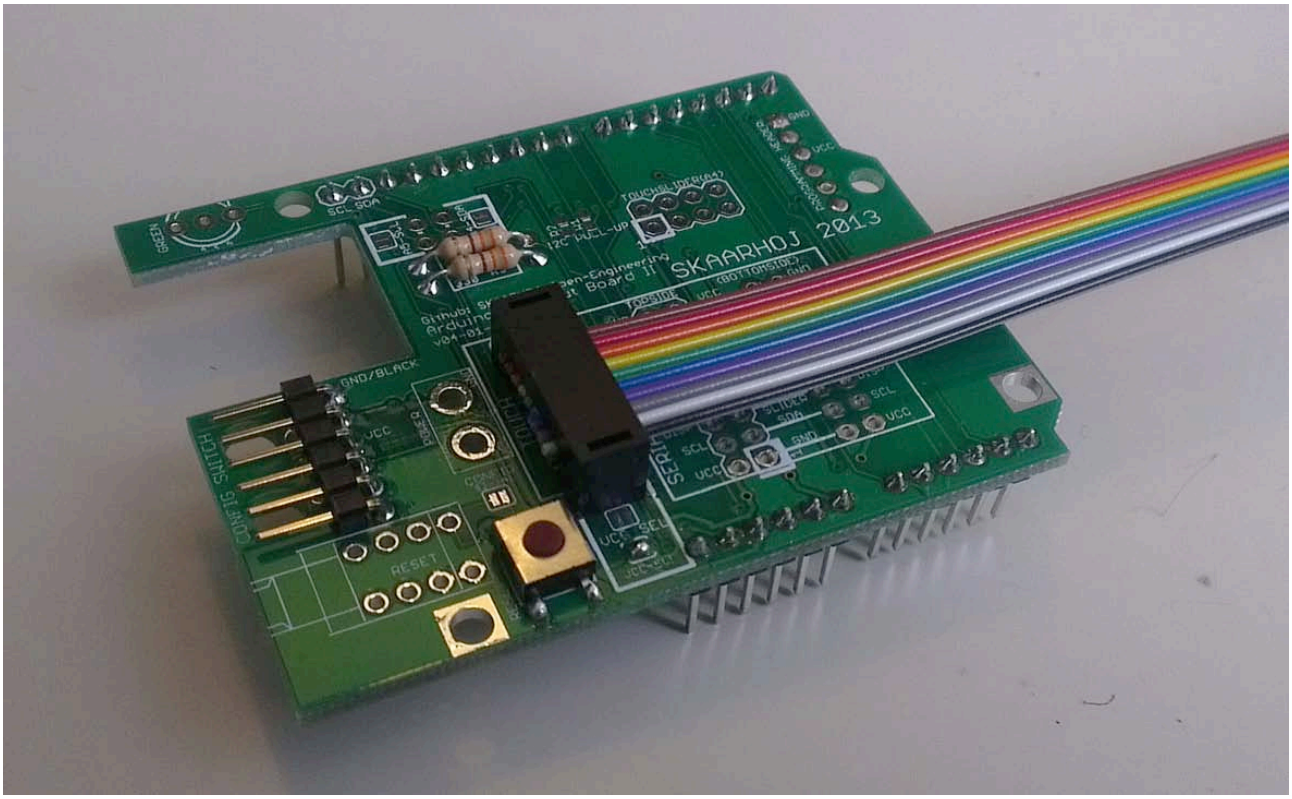
Remove the screws that hold the screen together. It looks like this when opened. You need to squeeze the plastic enclosure a bit in the top to make two small latches flip out. **Unmount the cables and the blue PCB** (printed circuit board).

Drill two holes for 3mm bolts through the bottom of the Arduino enclosure and the back of the monitor (which has the PCB removed now). Also **drill an extra hole larger** (around 6+mm) for the wires.

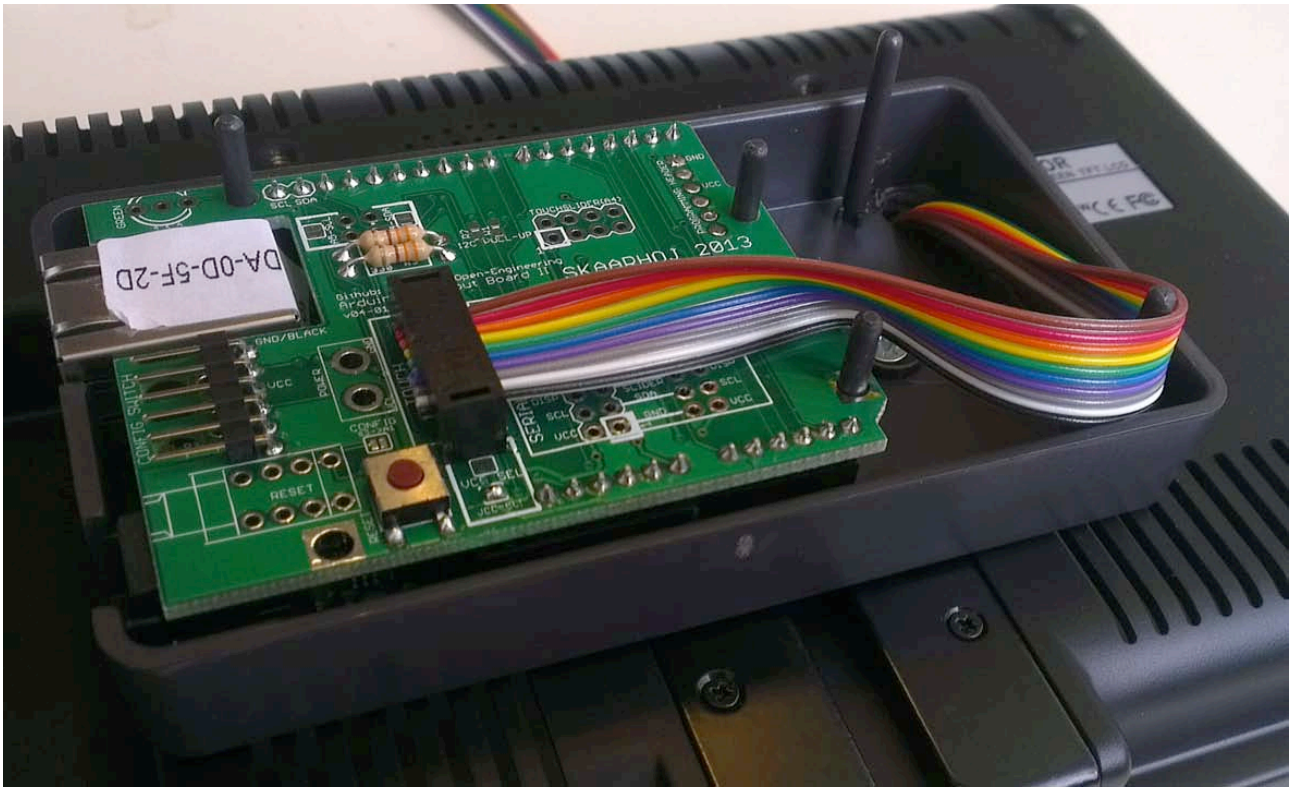


The two M3 bolts (10mm long) are tightened with **locking nuts.**

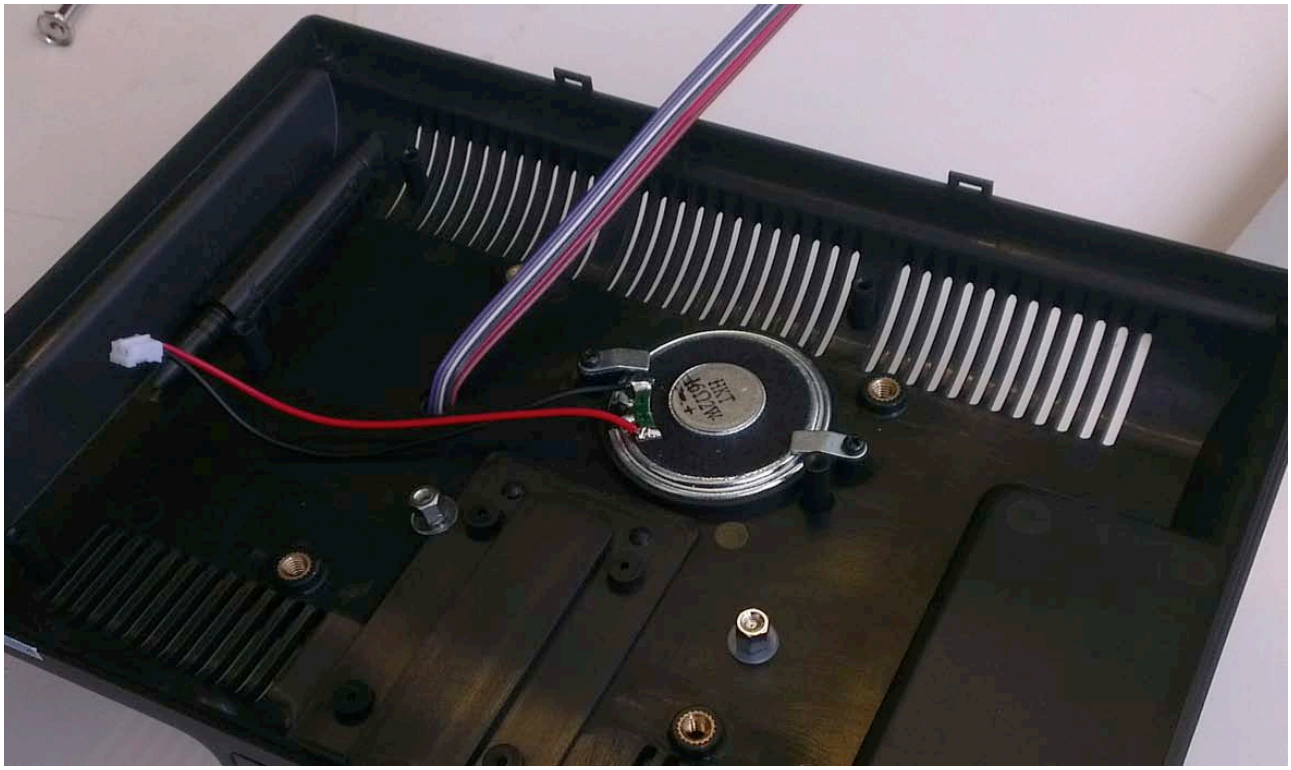
The nuts from the arduino side. Notice the 6mm hole for wires.

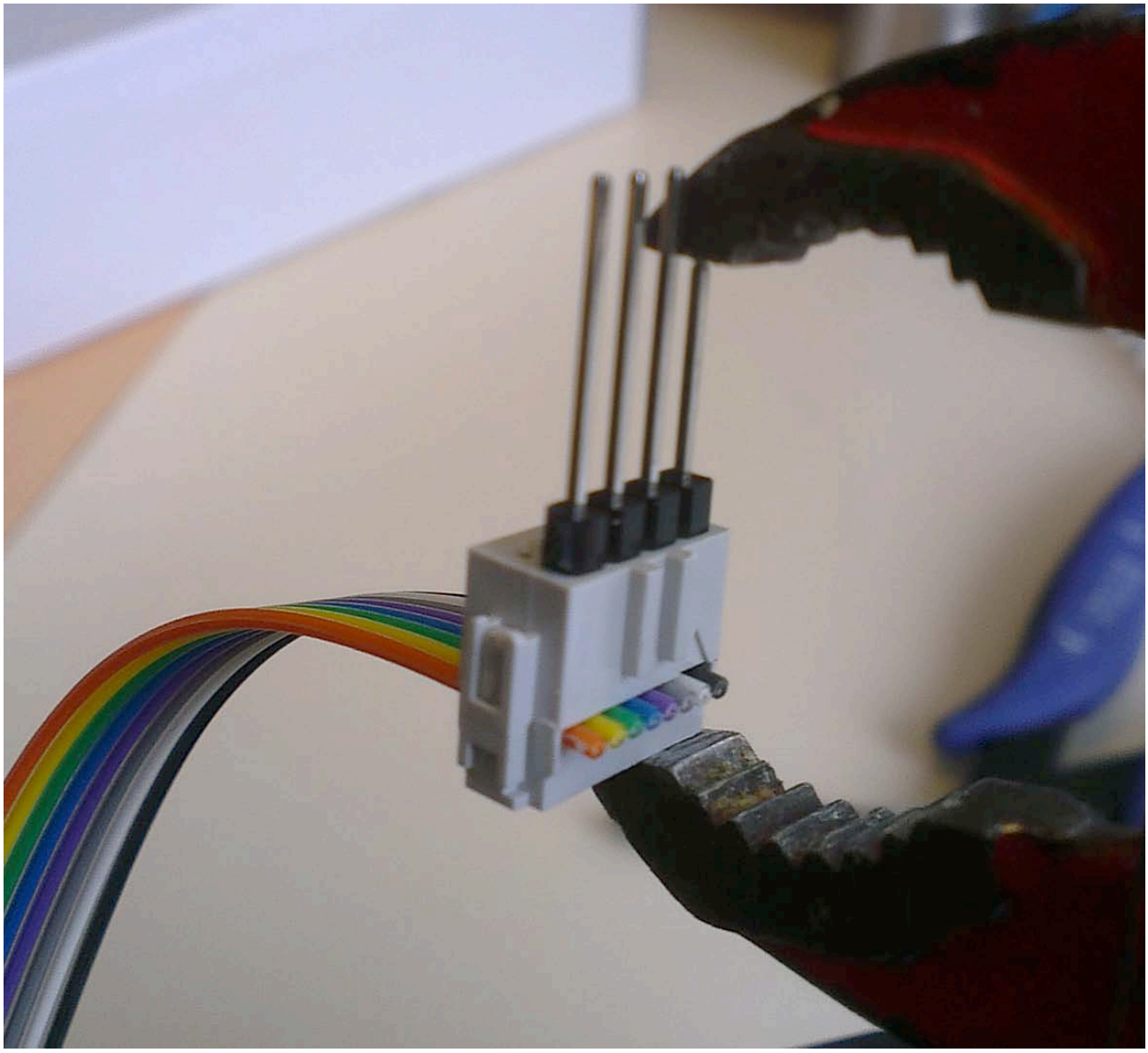


Solder the breakout shield: All header posts, resistors for bicolor LED, Reset button, 10W IDC connector with 30cm flatcable (going through the 6+mm hole into screen).

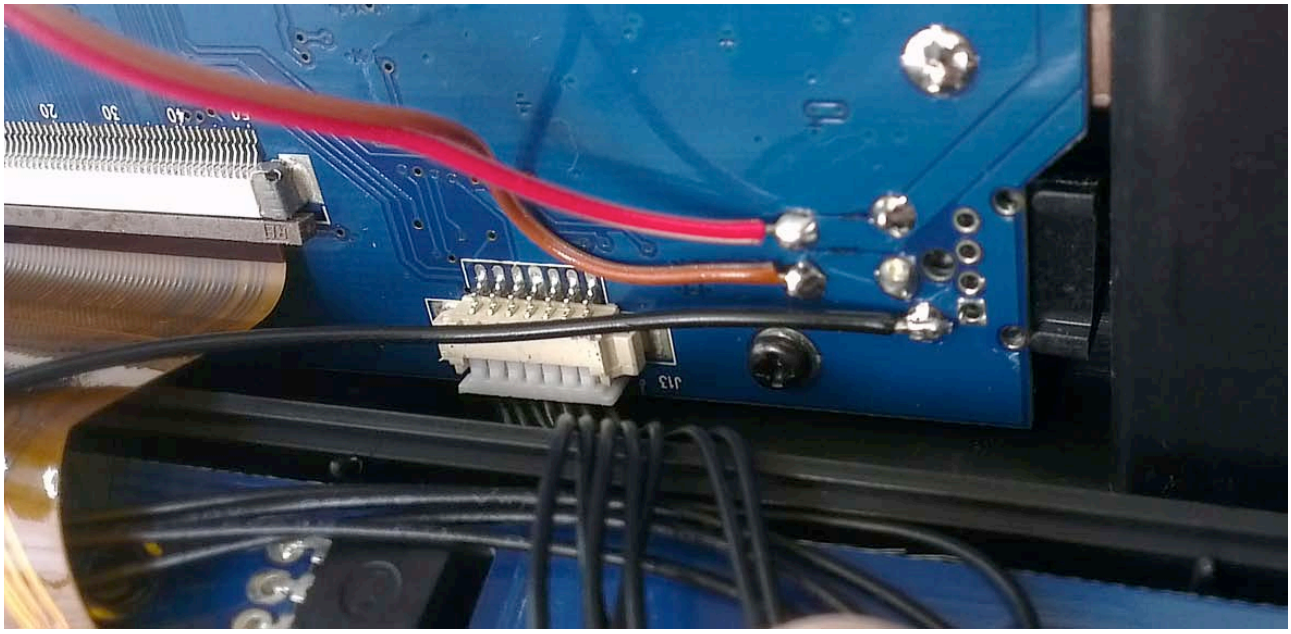


The wires going into the monitor case



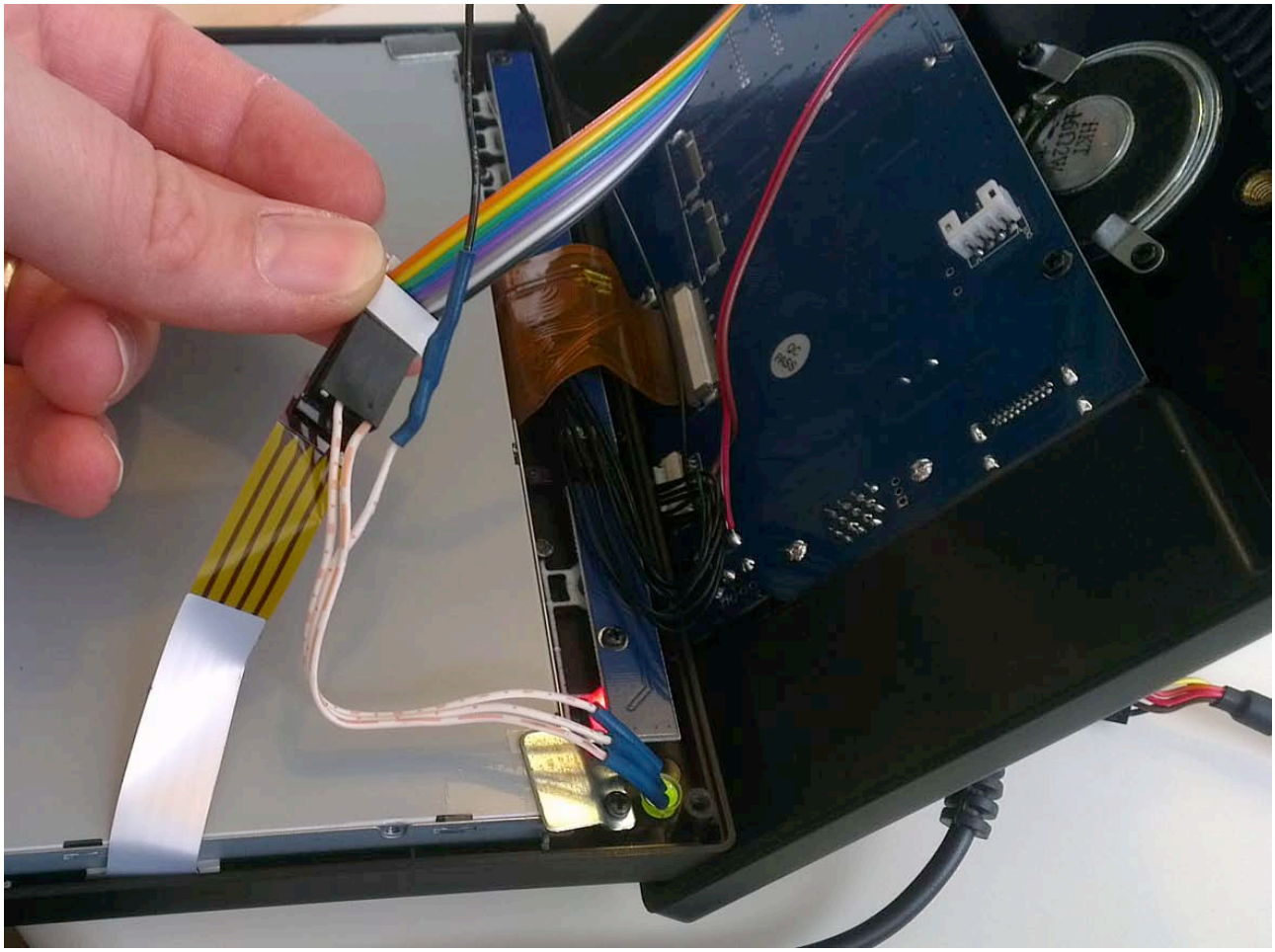


Place a **long header post** on pins 1,3,5,7 of and **IDC plug (8W)** pressed onto the flatcables **first 8 wires** (the last two are separated for power supply). With a plier**, press the pins down** into the IDC plug.

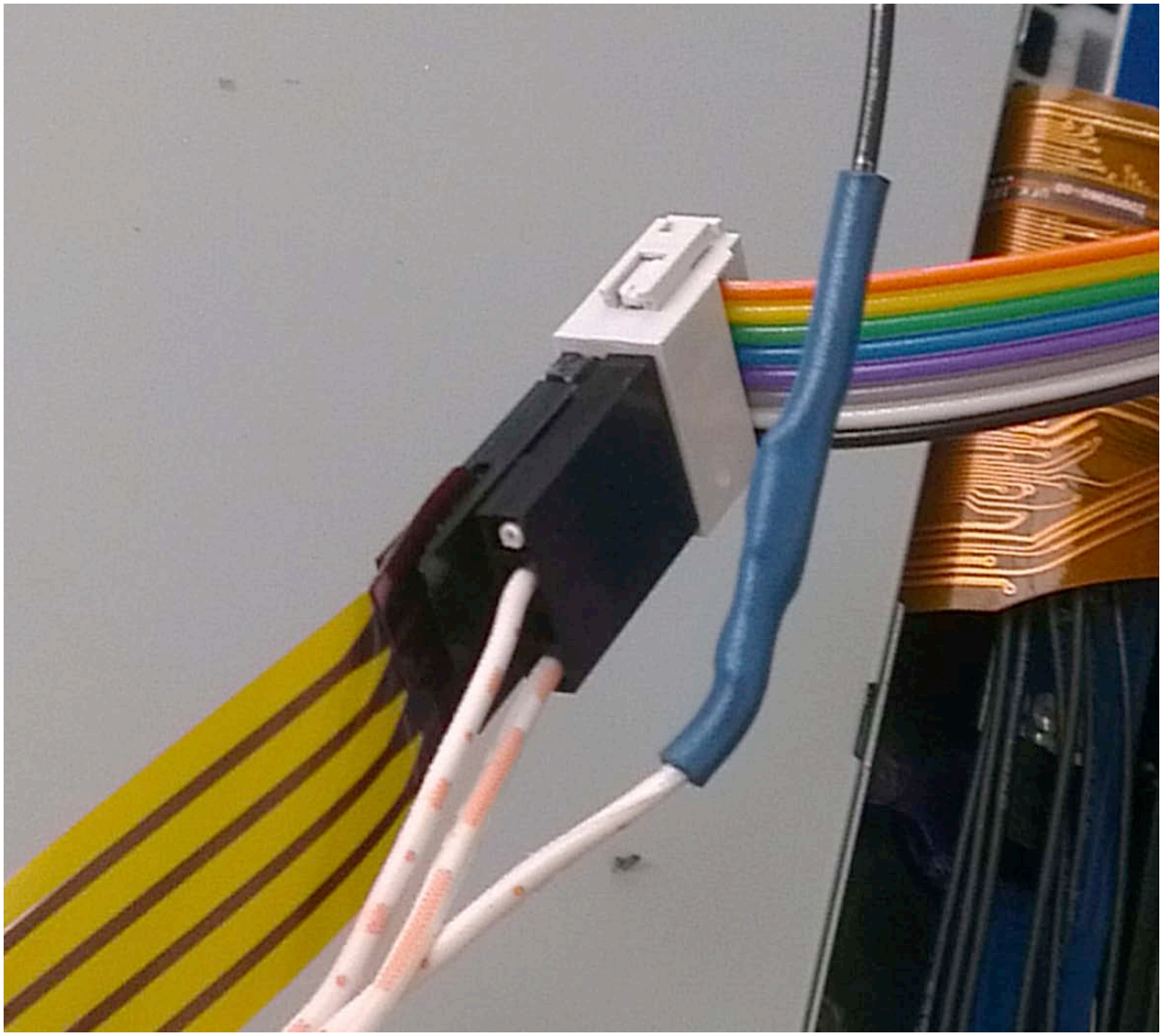


The **power wires 9 (Vin) + 10 (GND)** from the flat cable is **soldered to the DC plug** of the monitor. The black wire is for the LEDs (see next picture).

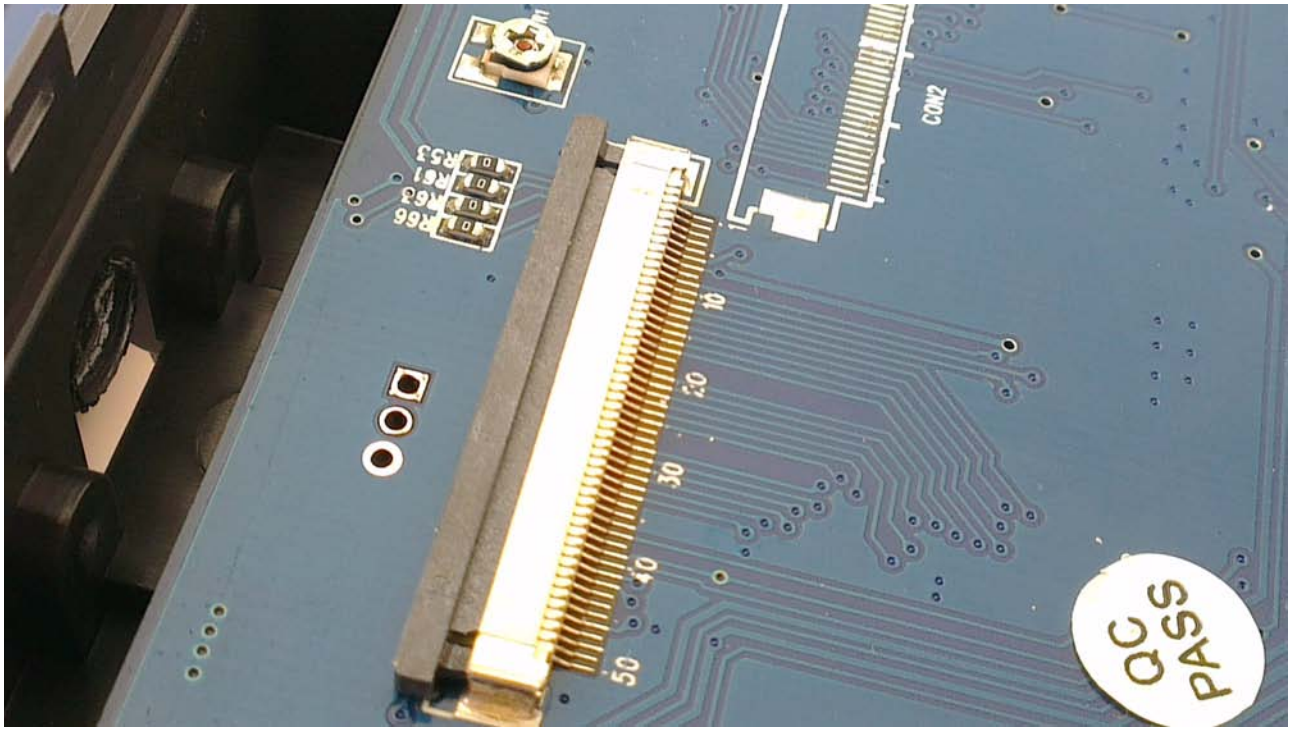
Drill a hole for a 5mm Bicolor led.



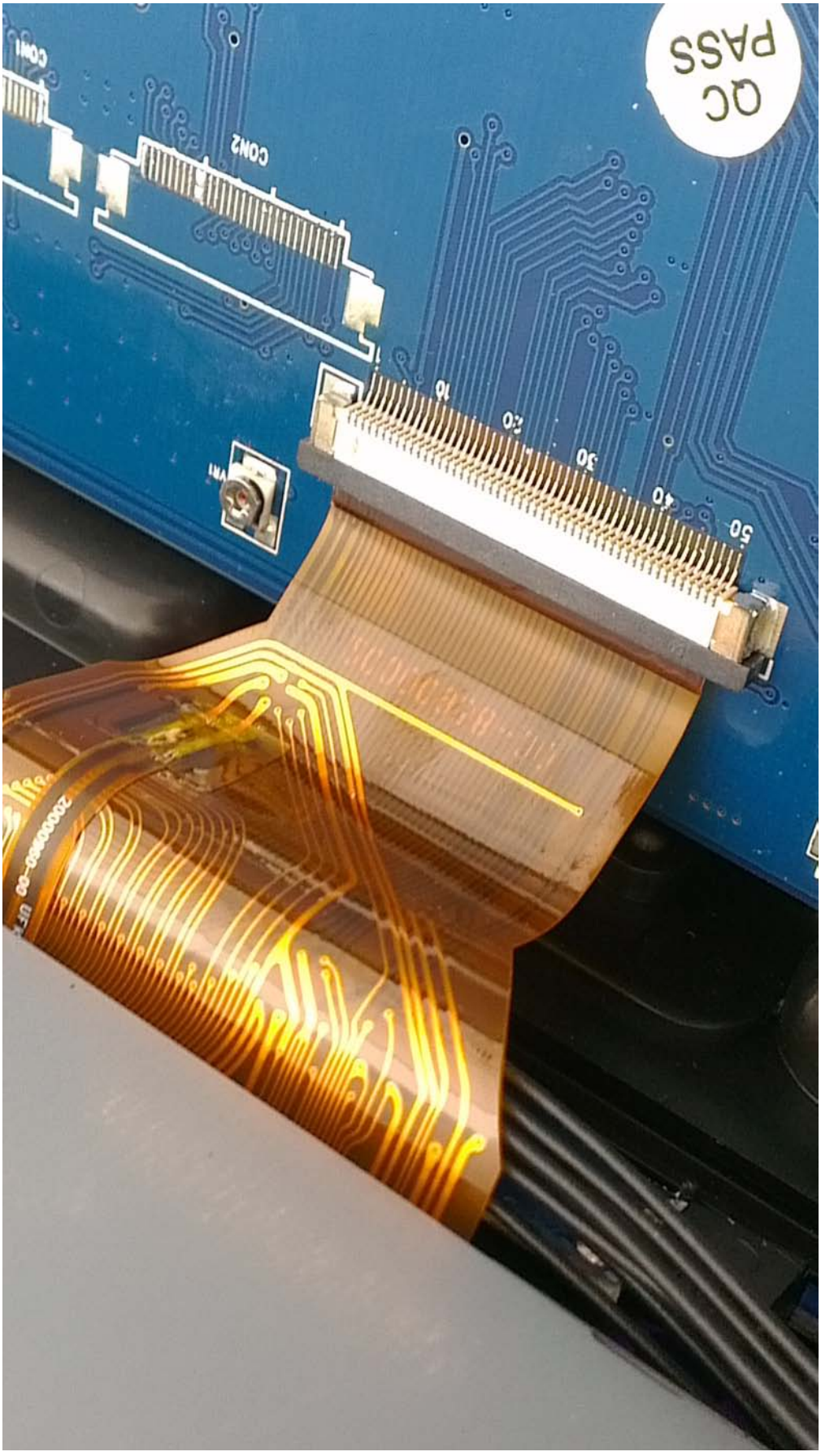
Place the **touchscreen connector on the other end of the header post pressed into the IDC plug**. Solder a **Bicolor LED onto wire** 1+3 of the wire originally used for the touchscreen, the middle pin is ground and soldered to a black wire connected to GND (see previous image). (When operation is confirmed later make sure to wrap tape around it).



(See previous description)



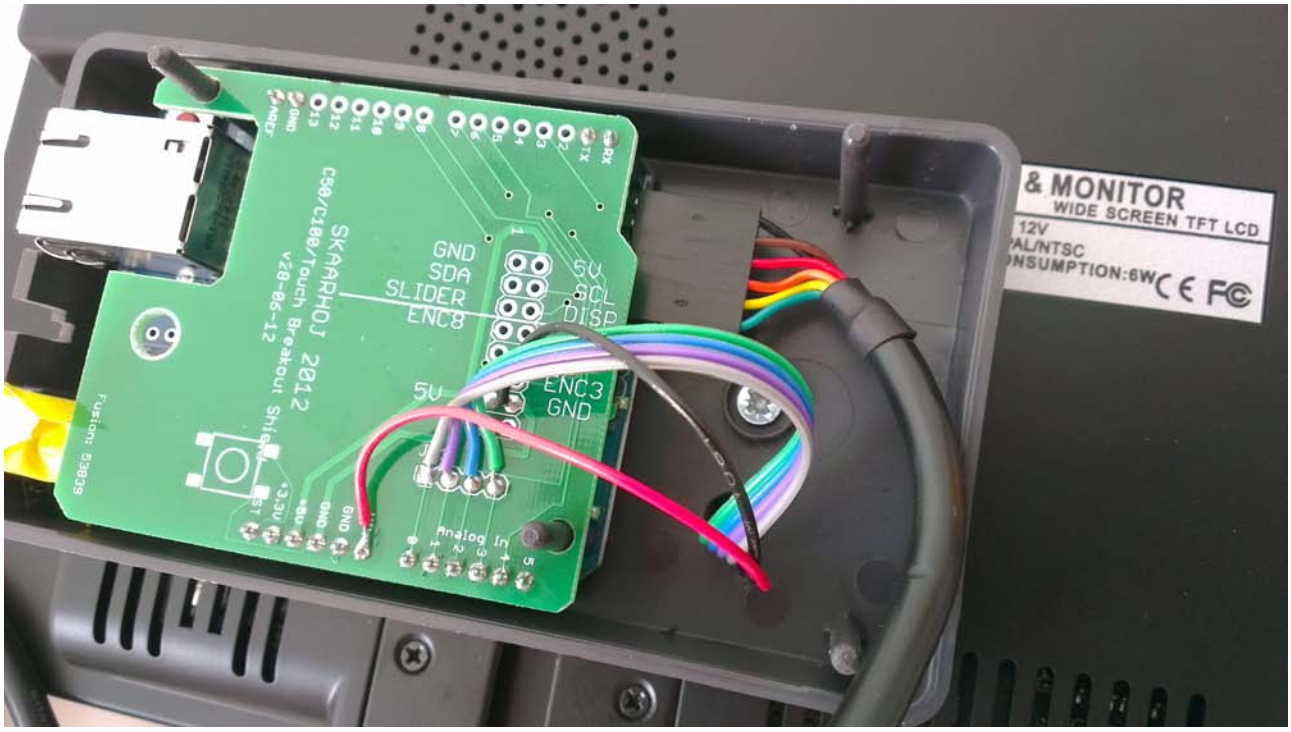
Most likely the touchscreen panel slipped out of this connector. Don’t worry. It’s easy to get it in again, just pust the brown/gray bar away from the white part. It easily moves out 1 mm or so.



Then, insert the flatcable from the panel into the connector and press the brown part back against the white part. It will fix the flatcable again. Make sure the flatcable was pushed all the way in and is now nicely aligned.

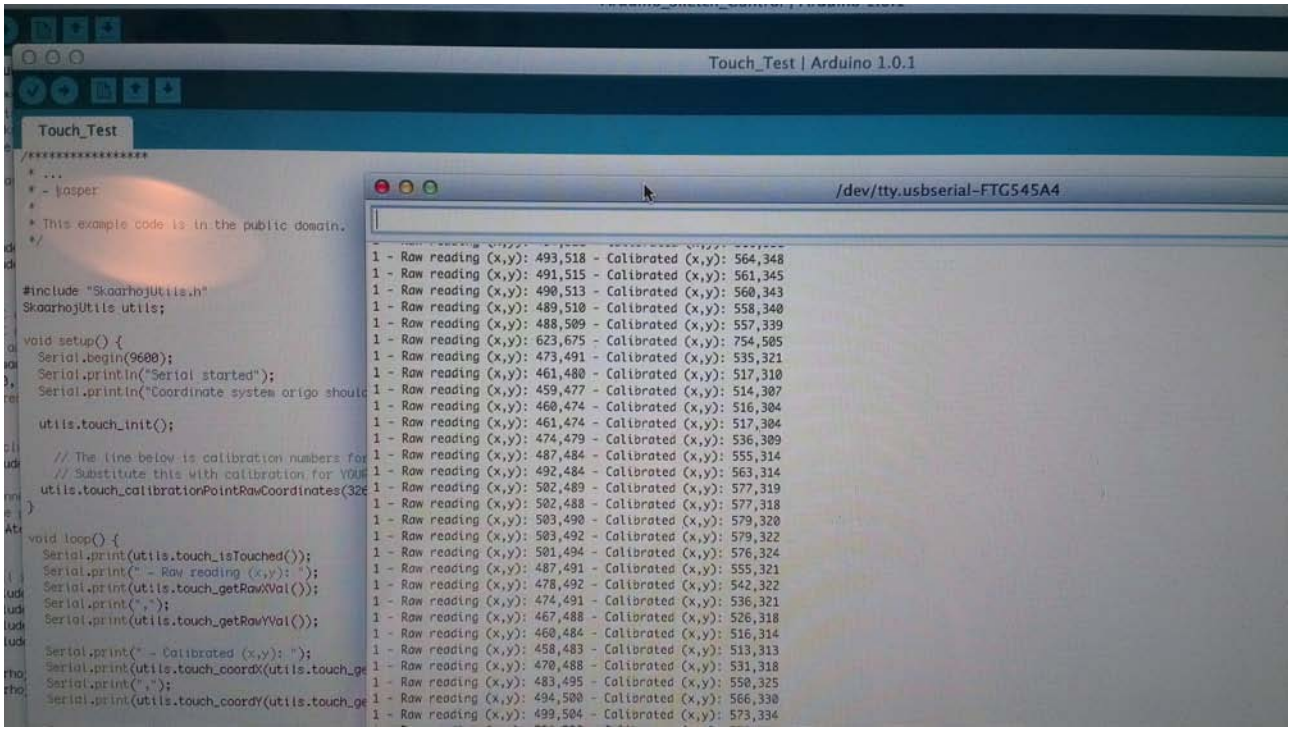
(Old photo, principle ok) If you want to make sure the power connections are right, you should be able to **measure a connection between the two DC plugs directly.**

(Old photo, principle ok) I like to **block the DC plug** of the arduino now that it is obsolete.

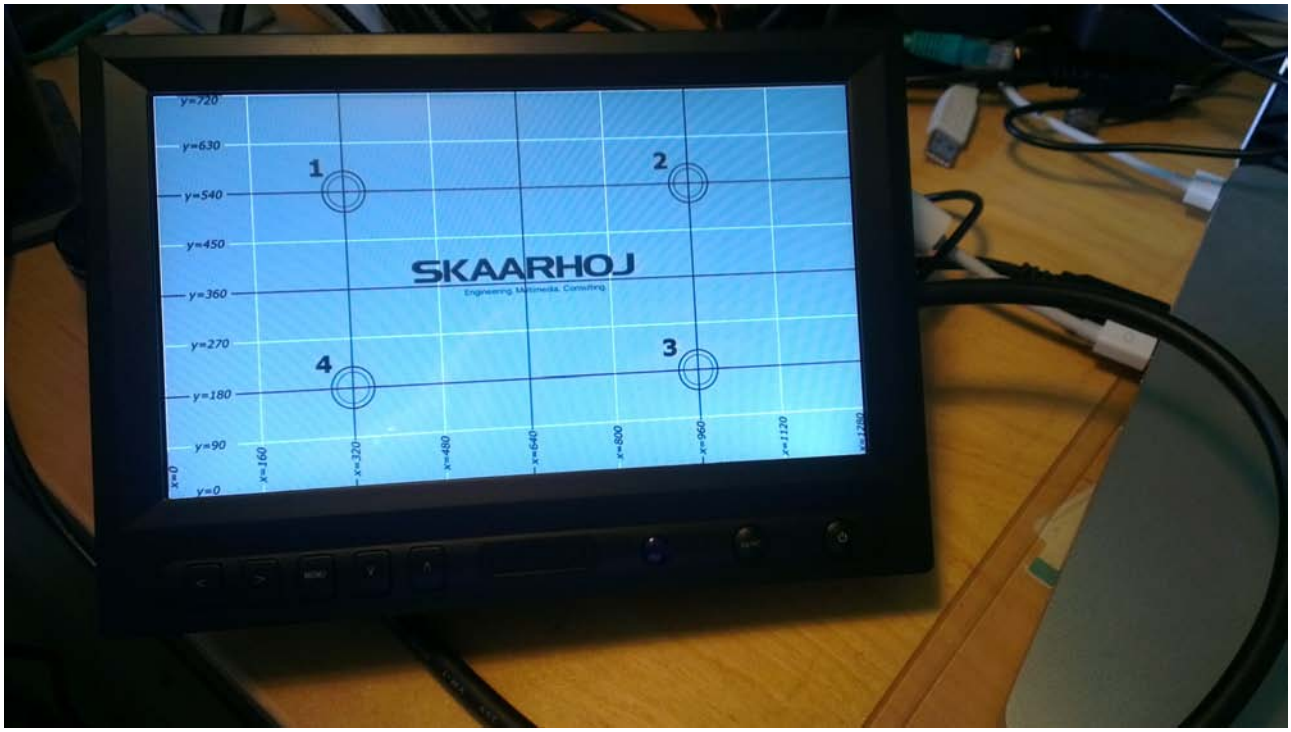


(Old photo, principle ok) The **FTDI programming cable is attached to the 6-pin header** on the Arduino Ethernet board. The black wire (out of the 6 in total) of the FTDI cable is always close to the edge. You only need this cable for programming. Notice, that even if this cable is a USB cable it needs a special driver installed before the Arduino IDE can program it. See step 3 at http://arduino.cc/en/Guide/Howto and http://www.ftdichip.com/Drivers/VCP.htm

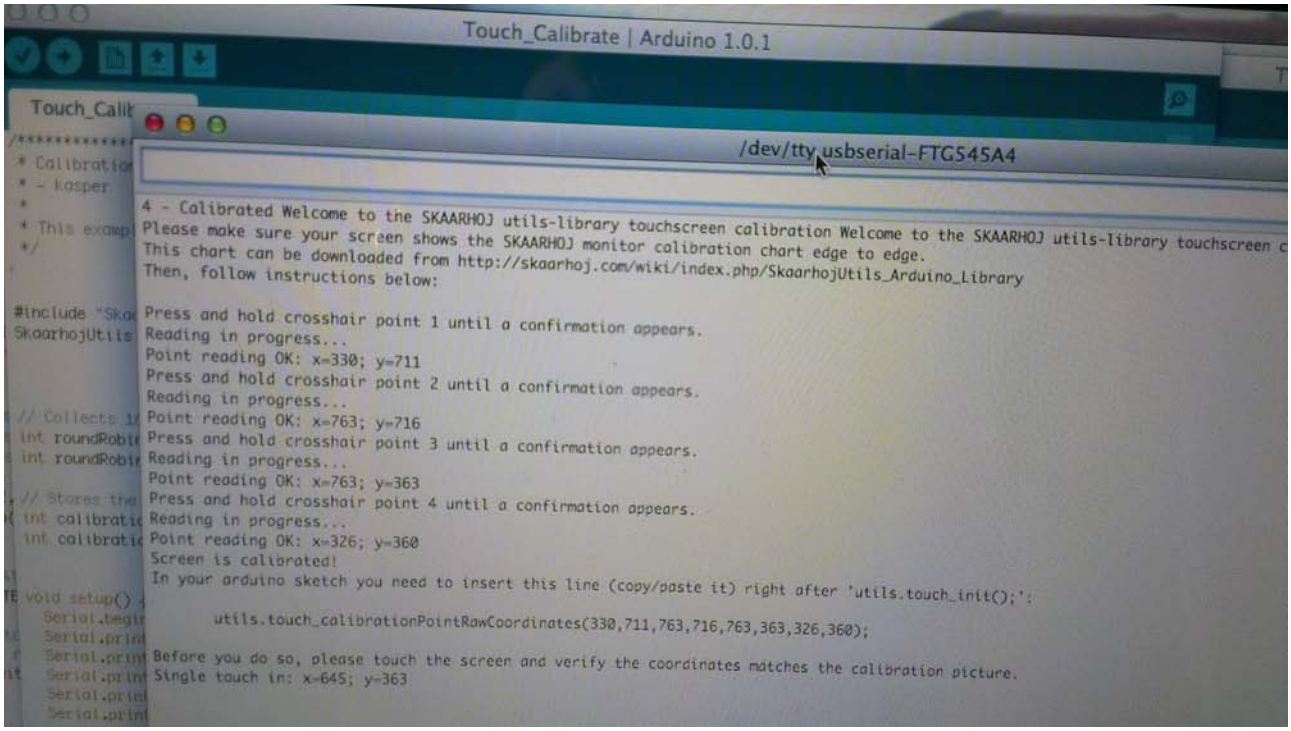
For testing the touch coordinates above you don't need to power up the screen at all. Just the Arduino needs USB power.



Upload the Touch_Test sketch and you should see coordinates when you touch the screen.

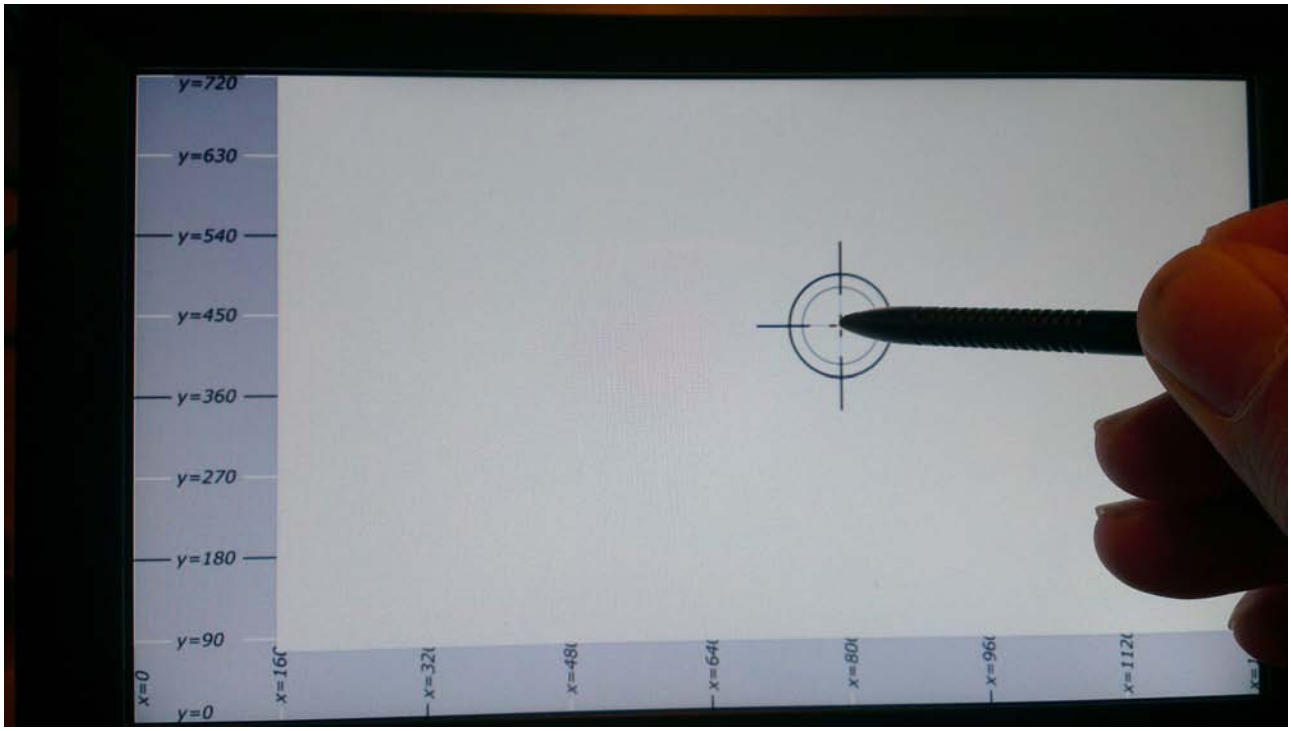


Power up the monitor and connect it to an image source. You could use a second computer or your ATEM switcher. **Display the SKAARHOJ monitor calibration chart on the screen**. It has 4 cross hairs used for calibration.



Upload the Touch_Calibrate example sketch to the Arduino. It will guide you through the calibration. This process eventually **gives you a codeline to copy/paste into other Arduino sketches** to make sure it's correctly calibrated.

Also, **take of the label from the back of the Arduino and reapply it on top**. This is the MAC address of the ethernet port. It's more useful to have this visible on top when programming. I usually reapply it to the header jack itself so I can see. In particular the last 3 hex numbers are important (the first 3 tend to be the same for Arduino boards).



Have fun with tracking a cross hair using the **ATEMtouchCrosshair** sketch should you have an ATEM 1M/E! :-)

Finalization

For actually using the screen as a touchscreen switching interface with the ATEM Multiviewer you should upload the example **ATEMtouchControl**. This will set any input source you touch in the multiviewer to Preview. Notice that it's optimized for the default layout of a 1M/E switcher. If you press and hold the Program window on the multiviewer it will change mode so that a touch of an input source is going directly to program. In order to get a visual feedback on these two modes the bi-color LED is used.

LED colors:
- When the default "Preview" mode is on, it's green.
- When "Program" mode, it's red.
- When the board is trying to connect to the ATEM switcher but has no connection yet, it's yellow (both red and green)

- kasper