Laserentfernungsmesser

Inhaltsverzeichnis

Laserentfernungsmesser Was steckt im Laserentfernungsmesser ? Display-Ansteuerung dekodieren

Durch Zufall bin ich günstig an einen Laserentfernungsmesser gekommen. So ein Teil ist schon hochinteressant ! Es hat auch funktioniert und danach kam natürlich der Schraubendreher 🙂



So sieht das Teil von aussen aus. China-Ware und dann je nach Kundenwunsch ein Firmenlogo.



Frontansicht : Man erkennt die obere Linse wo das Auge das Objekt sieht und wo der Laserstrahl ausgesendet wird. Die Untere Linse ist mit der Empfängeroptik verbunden (vermutlich ne APD Diode).



Kurze Beschreibung der Technischen Daten :

Laser Rangefinder



TECHNICAL SPECIFICATIONS

Optical System

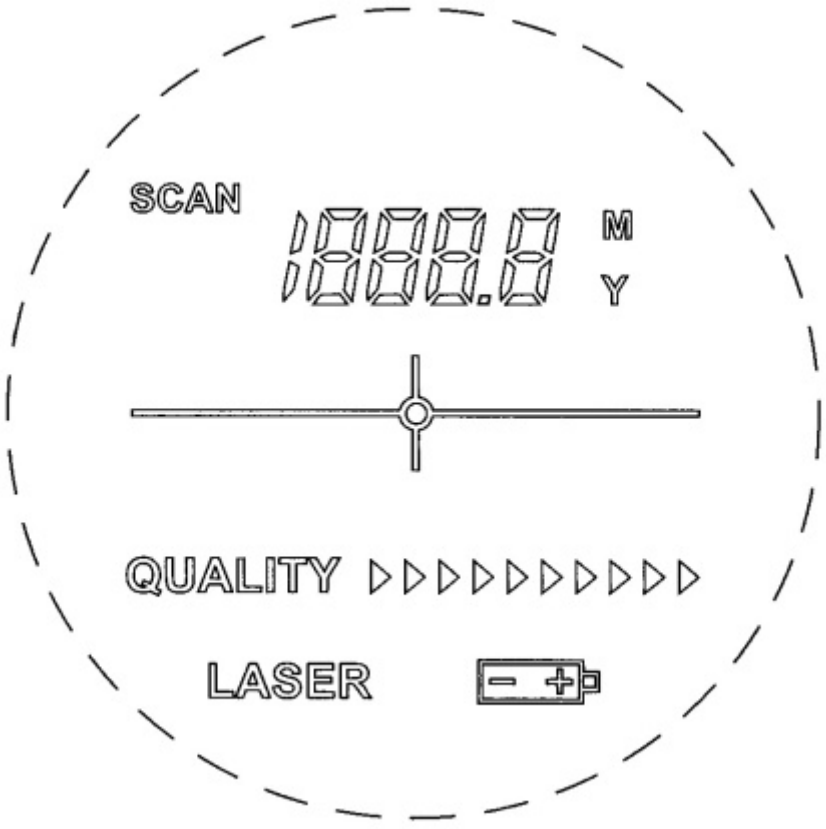
Magnification	(X)	7
Objective Lens	(mm)	26
Field of view	(°)	6
Exit pupil	(mm)	3.7
Display		LCD
Lens		Optical Glass

Laser

Class		Laser Class 1
Wavelength		905 nm
Range		15-1400 meter
Accuracy	(m)	±1
Battery life	(times)	3000
Battery type		AAA 1.5V*2
Weight	(g)	200
Dimensions	(mm)	124x48x82

User manual & Operation Guide

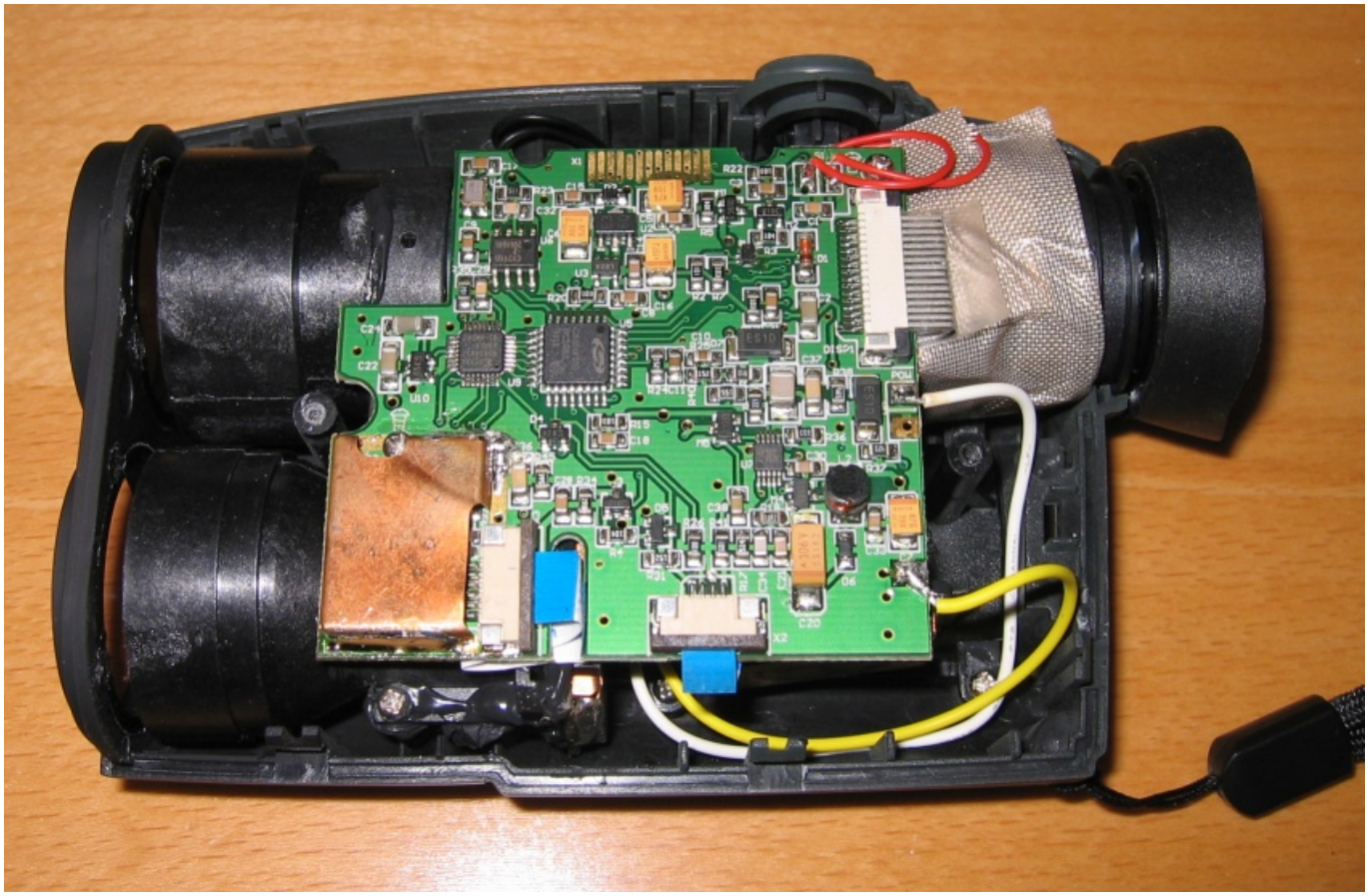
Und hier noch die möglichen Innenansichts-Segmente :



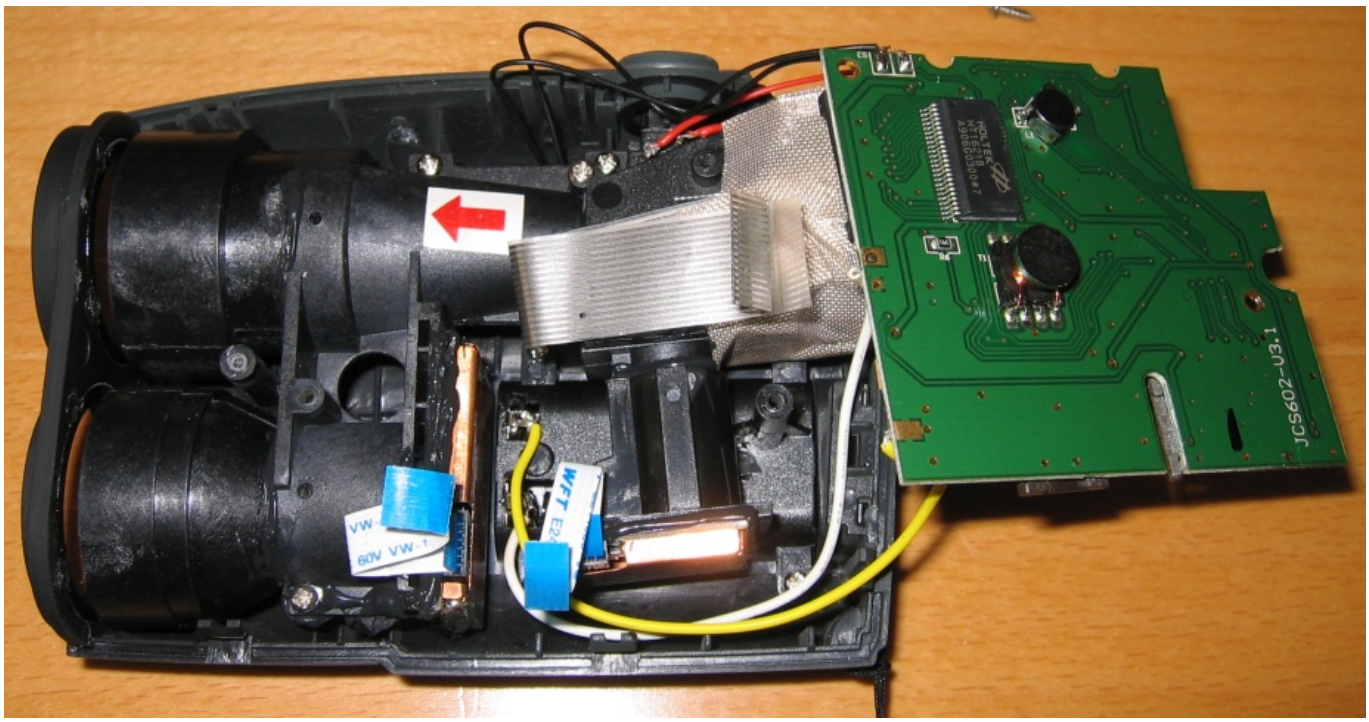
So sieht das ganze in Aktion aus :



Dann mal auf das gute Stück :



Klappt man die Platine weg, kommt die Optik zum Vorschein :



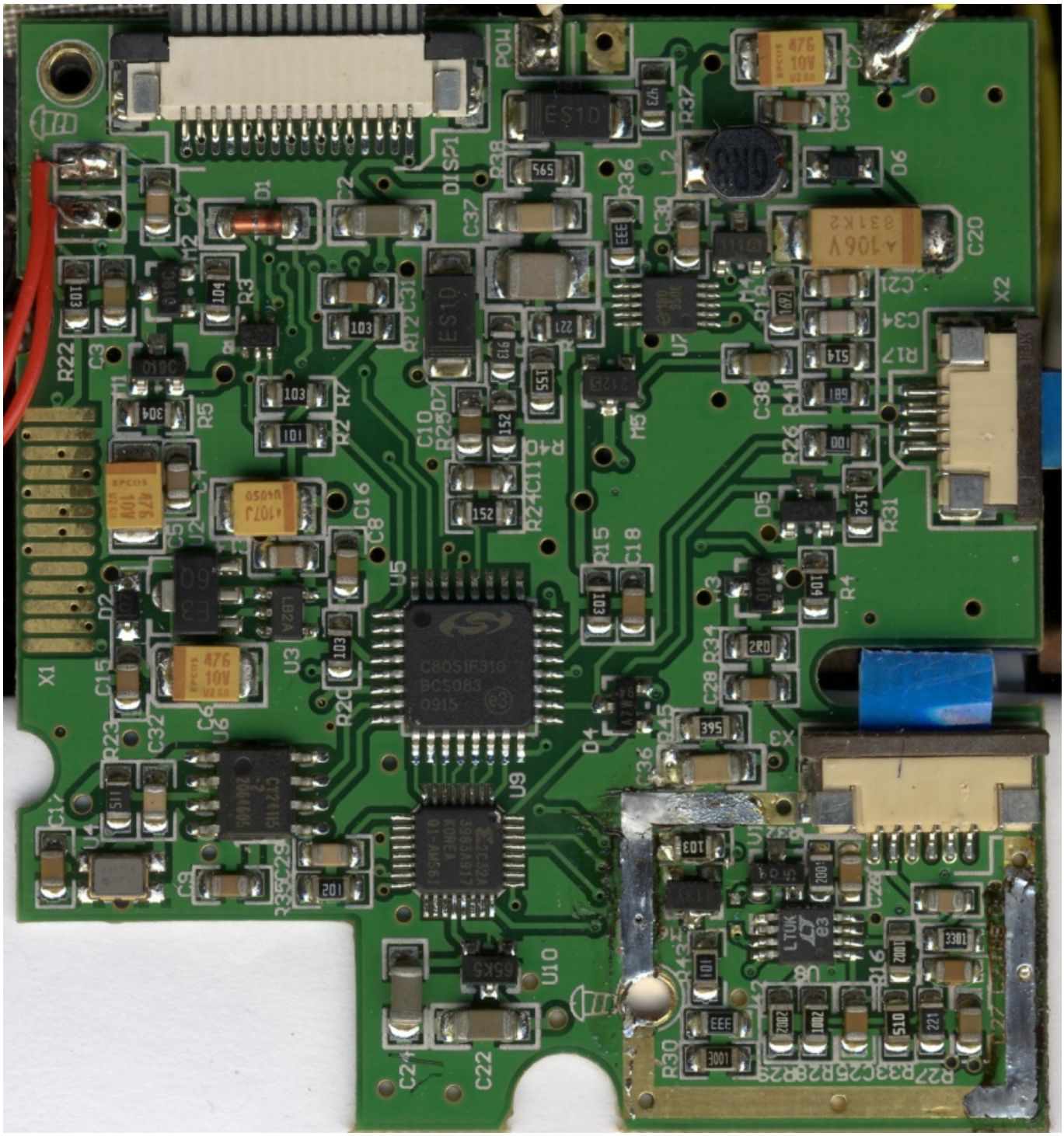
Was haben wir auf der Top-Lage ?

```
Bezeichnung     Typ     Aufgabe
C8051f31    uC     8051 Clone Microcontroller zur Steuerung
CY24115     PLL+Clock     Clockgenerator
```

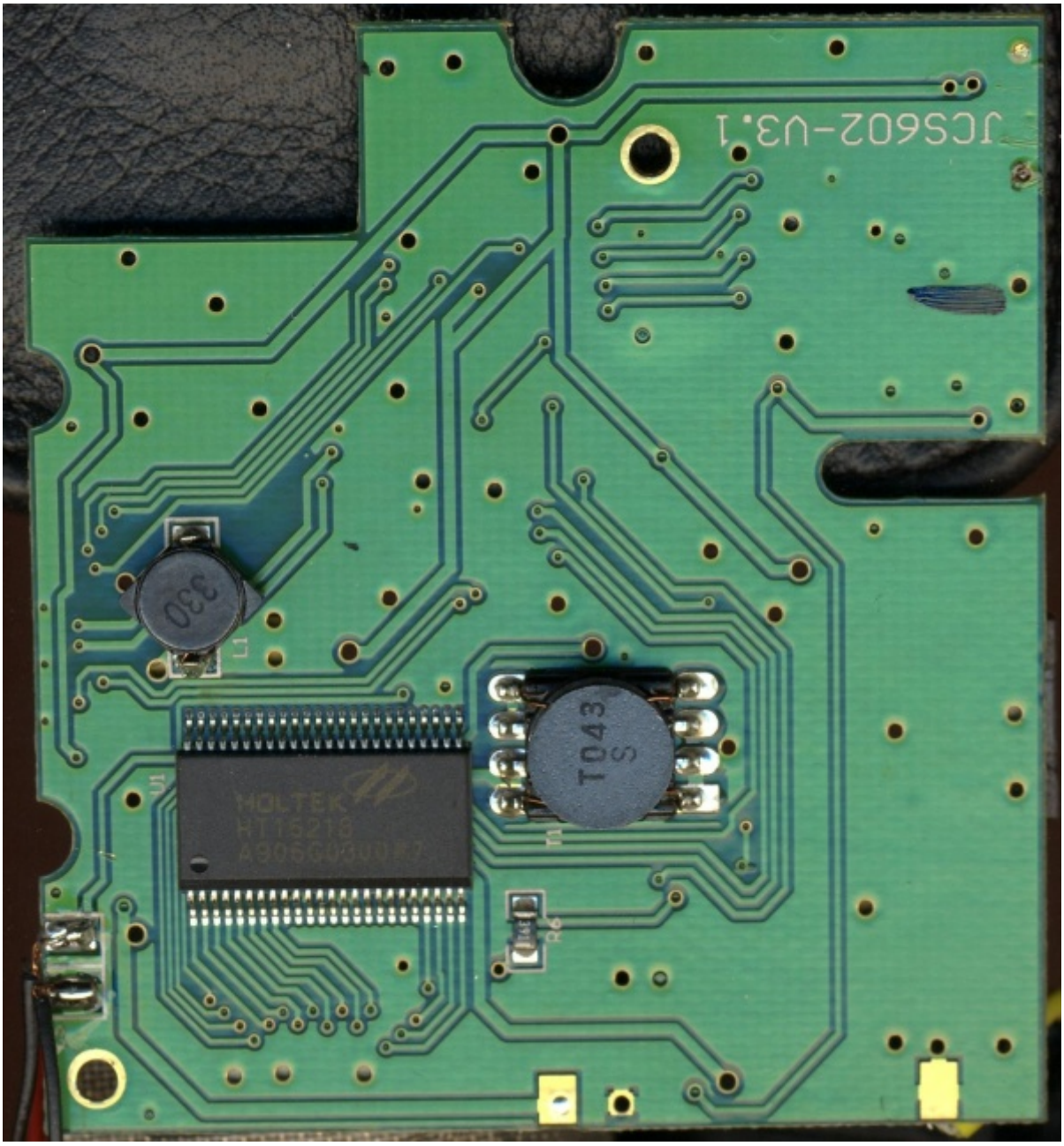
```
CPLD XC2C32A    CPLD

Coolrunner CPLD für die Timingkritische Distanzmessung ?
LT1713_LTUK     Komparator

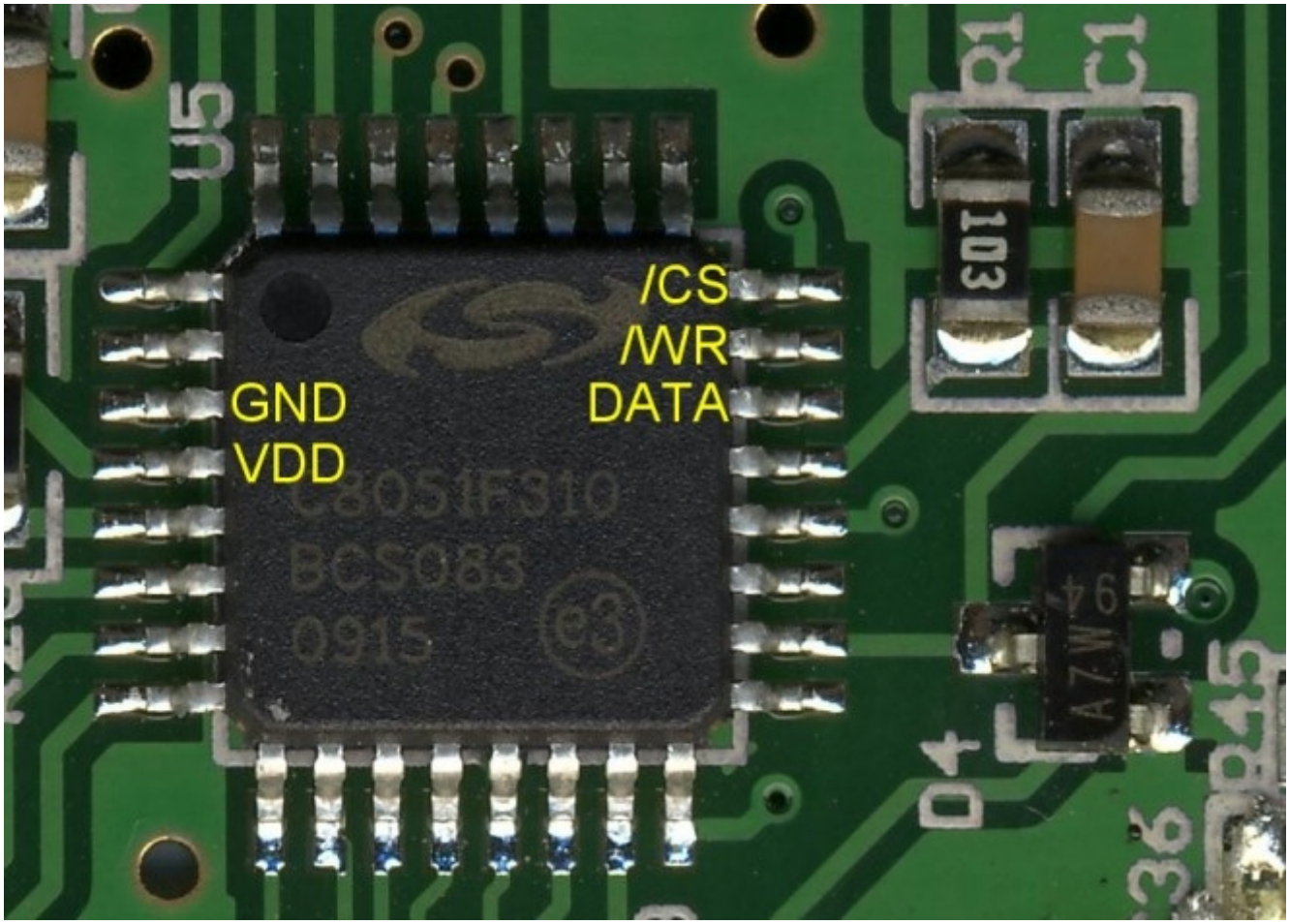
Komparator fürs Empfangssignal der APD Diode
```



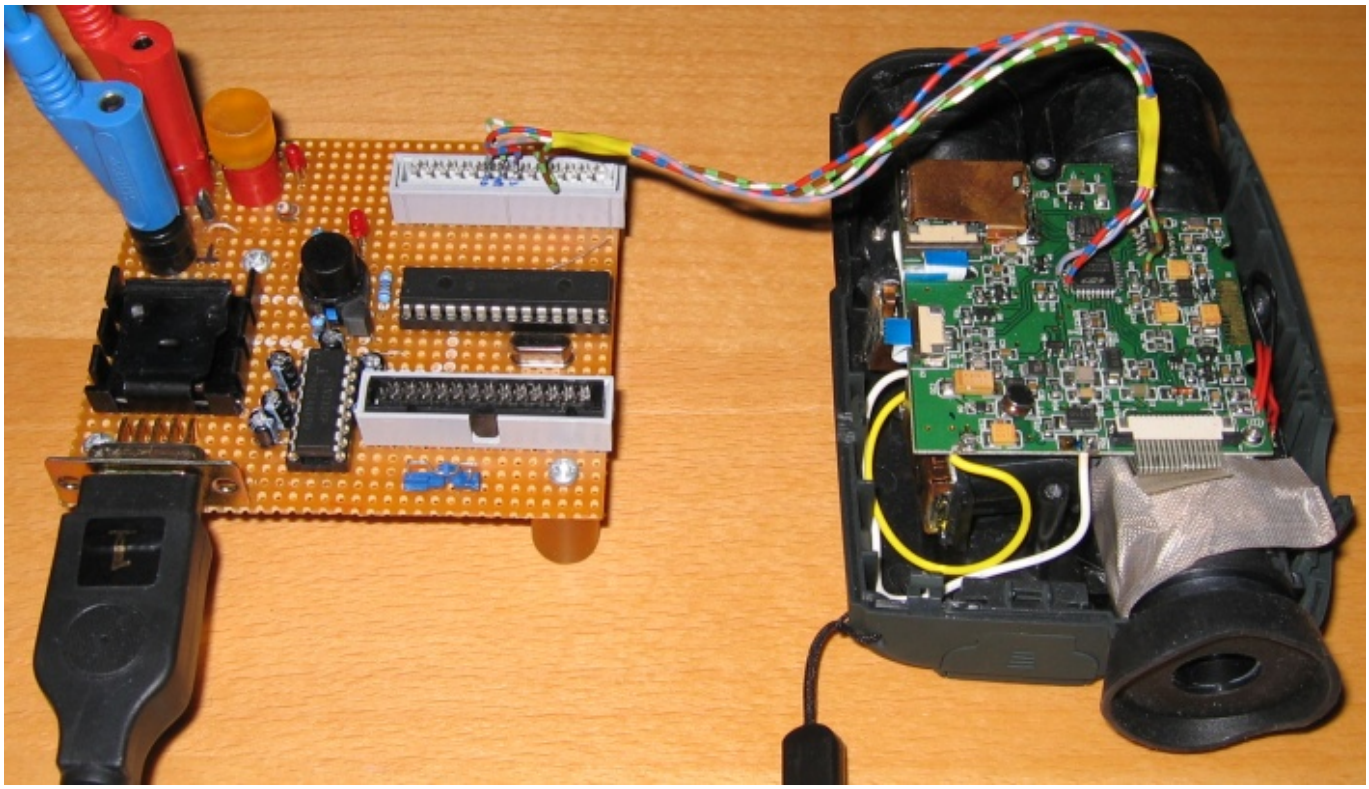
Und was gibt die Rückseite her ? Bezeichnung Typ Aufgabe 1621B Controller Display-Controller für LCD Einblendung



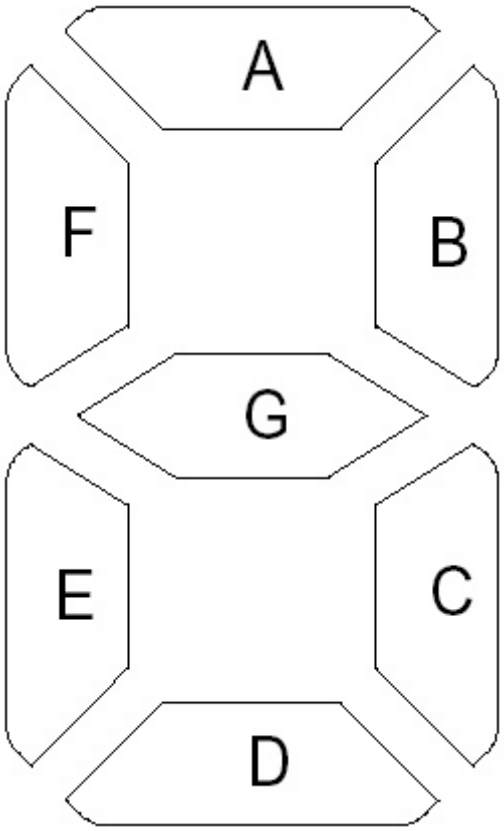
Spielt man eine Weile rum, vermisst man den USB-Anschluss um die gemessenen Werte abzuspeichern ... Nur wie kommt man da ran ? Idee war einfach den Bus anzuzapfen der zum Display-Controller geht !



Dann per uC die einzelnen Segmente auslesen und an den PC schicken zur Analyse ...



Damit kein Durcheinander herrscht noch die Segmente mit Namen belegen :



Und ob man die Segmente nicht kompakter abspeichern könnte ....



Anmerkung :

Möglich sind mit dem Display-Controller 128 Pattern (32×4).

```
Benutzt werden offensichtlich :
28 Segmente     4x7 Segment-Ziffern
2 Segmente      1x die führende 1
1 Segmente      Dezimal-Punkt
10 Segmente     Quality-Dreiecke
7 Segmente  Meter / Yard / Battery / Quality / Laser / Scan / Zielkreuz
```

Ergibt 48 Segmente die mit 4 Common und 12 Segment-Leitungen über das 16 polige Display-Kabel angesteuert werden.

```
Am Microcontroller ist die Belegung wie folgt :
Pin 3   Masse
Pin 4   VDD
Pin 22  DATA
Pin 23  /WR
Pin 24  /CS

Am Displaycontroller ist die Belegung wie folgt :
Pin 9   /CS
Pin 11  /WR
Pin 12  DATA
Pin 13  VSS
Pin 17  VDD
```

/Chipselect :

Normal High, geht auf Low für 30..35us zwischen den Übertragungen, eine Übertragung aller Daten dauert ca. 280..290us. Eine Übertragung enthält das komplette Display-RAM. ca. 3000mal pro Sekunde wird das RAM neu beschrieben.

/WR :

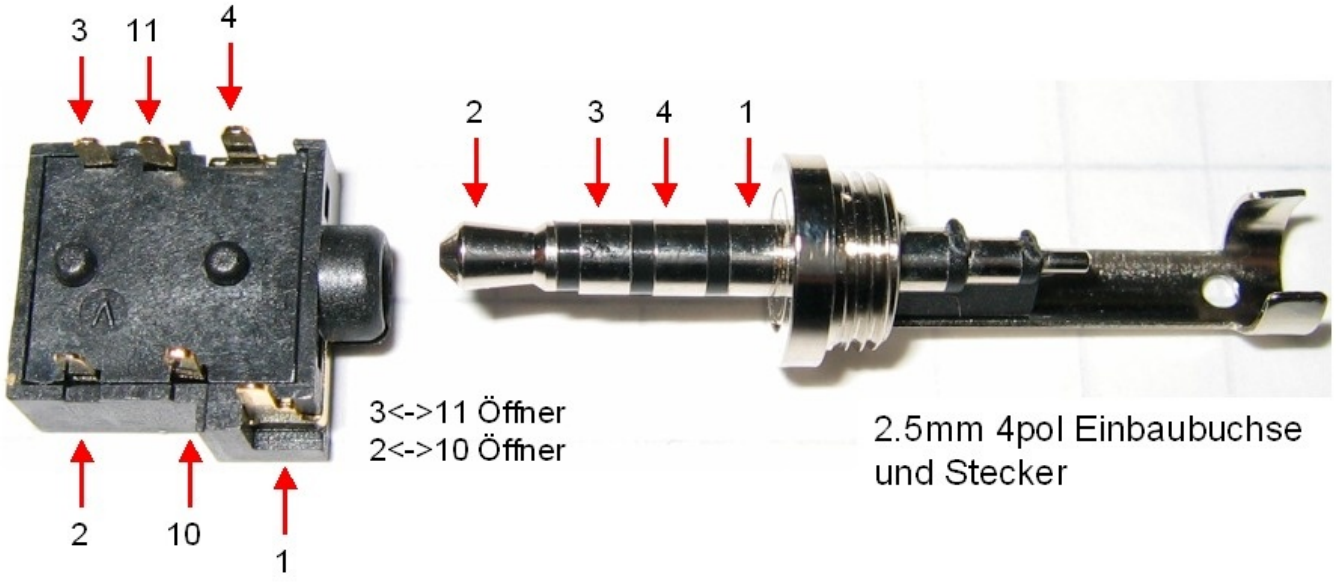
ein Bit dauert ca. 1,4 .. 3,0us. Die Datenübernahme erfolgt vom Chip bei der steigenden Flanke des /WR Signals.

DATA :

Es wird nur das Write Mode Successive Address Writing verwendet. Gesendet wird der Befehl „101“, dann „000000“ als Start-Adresse sowie 32x 4 Bit Blöcke. Es wird also jedesmal das ganze Display-RAM im IC neu geschrieben.

Zum Batteriesymbol : Bei grob 2,4V wird Low Batt angezeigt, wenns noch weniger ist zeigt das Display beim einschalten nur noch „LOU“

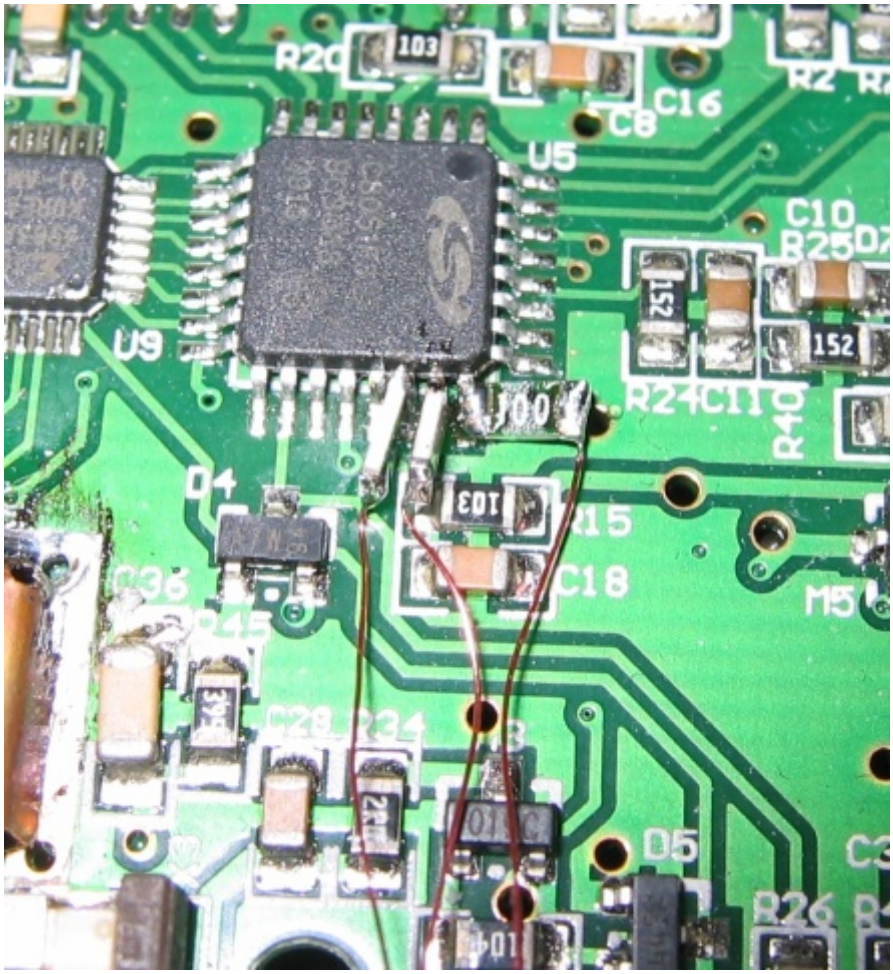
Da die Drähte auf Dauer auch nix sind, habe ich mir eine 4polie 2,5mm Buchse eingebaut, die wie folgt belegt ist :



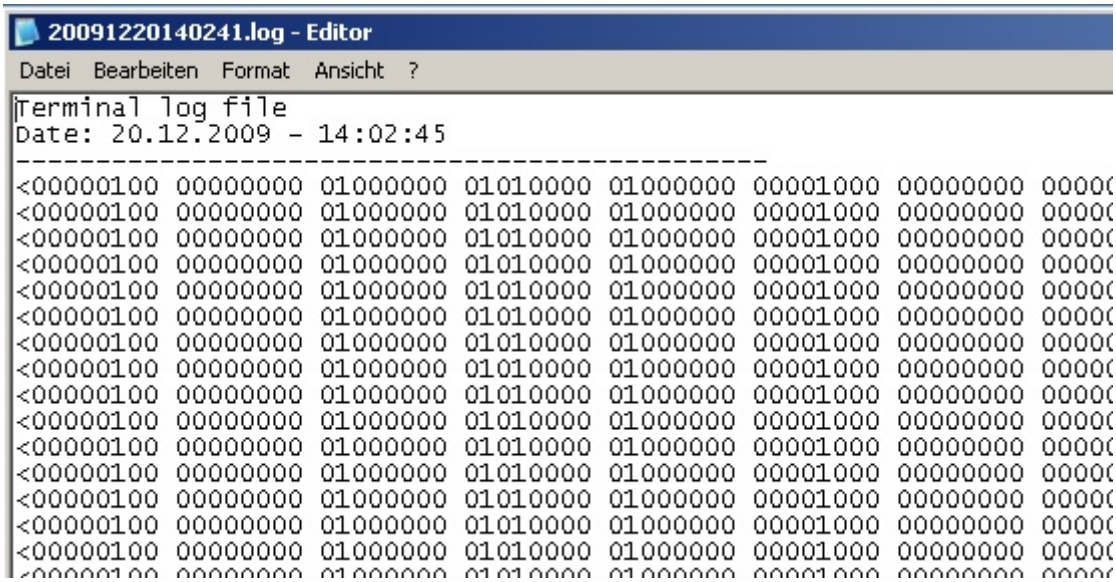
So sieht das ganze im Gerät aus :



Und die Leitungen sicherheitshalber mit 10Ohm Serienwiderständen abgreifen und auf die Buchse legen :



Der PIC-Controller emuliert nun das komplette Display-RAM des Holtek und schreibt regelmäßig den RAM-Inhalt auf der Seriellen Schnitte raus. Sieht dann so aus :



Assembler-Code und Hex-File in rudimentärem Zustand liegen hier (Ist nur zusammengestümpert, bitte keine großen Erwartungen hegen) : LRFDD.zip :

lrfdd.zip

Die Liste welches Segment was bedeutet habe ich herausgefunden : Displayspeicher.pdf :

displayspeicher.pdf

Auf PC Seite müsste jetzt ein Programm die Display-Segmente wieder zusammenbasteln und in Zahlen anzeigen. Und an dieser Stelle hab ich das Projekt aus Zeitgründen eingestellt. Wer Interesse hat, kann hier weiterarbeiten. Viel Spass !

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

Permanent link:
https://elektronikfriedhof.de/doku.php?id=elektronik:misc:laserentf

Last update: **2025/01/17 17:43**