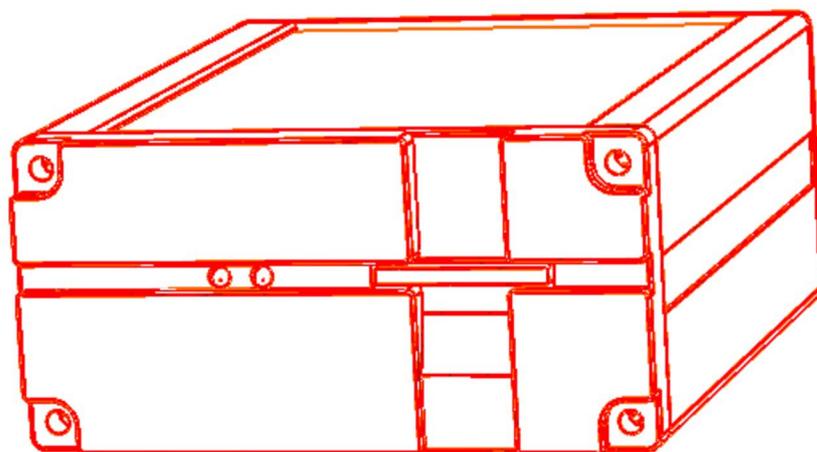


*Christoph Egretzberger*

# **ND158I und EgretzMB Floppy Buch**

sd2iec und iecata für Anfänger



***EIN DATA EGRETZ BUCH***

Laufwerks und Kontaktplatine für den C64 DTV  
und/oder zwei Laufwerke für den C64/C128  
Copyright Christoph Egretzberger

Beschreibung Version: 2.1 (27.10.2015)  
Platinenversion: 2.35 (Prototyp2) und 2.36

# Inhaltsverzeichnis

<b>Kapitel 1: Einleitung</b>	<b>5</b>
<b>Kapitel 2: Software</b>	<b>7</b>
2.1 Arbeiten mit dem sd2iec Laufwerk .....	7
Verzeichnisse wechseln (CMD-Syntax) .....	7
D64 benutzen (am Beispiel Inhaltsverzeichnes aufrufen) .....	7
Firmwareversion abfragen (nur direkt nach dem Einschalten möglich) .....	8
D64 Diskettenverzeichnis erzeugen .....	8
D81 Diskettenverzeichnis erzeugen .....	8
D71 Diskettenverzeichnis erzeugen .....	8
<b>Kapitel 3: Platinenaufbau</b>	<b>11</b>
3.1 Anschlüsse und Jumper .....	11
<b>Kapitel 4: Aufbauanleitung</b>	<b>13</b>
4.1 SMD Bauteile auf der Unterseite .....	13
SMD Löten einfach so... .....	14
4.2 Kontaktstifte .....	15
4.3 SD Card Reader .....	16
4.4 IC Fassungen und Quarz .....	17
4.5 Stiftleisten verlöten .....	18
<b>Kapitel 5: Aufbau für Baugruppen</b>	<b>19</b>
5.1 DTV Funktionen .....	19
5.2 SD2IEC .....	19
5.3 IEC ATA .....	21
5.4 Lumafix (Spiff) .....	21
<b>Kapitel 6: Gehäuse</b>	<b>25</b>
6.1 Kleinspannung .....	25
6.2 Mechanische Abmessungen der Platine .....	26
6.3 DTV-Gehäuse .....	27
6.4 ND1581 Gehäuse .....	27
6.5 Egretz 15Mx81 – Gehäuse (nur konstruiert Stand 03.09.2013) .....	27
<b>Kapitel 7: NLQ-HD-Firmware</b>	<b>29</b>
7.1 Kurzanleitung .....	29
7.2 Ausführliche Anleitung .....	30
Zitat von »NLQ« .....	33
<b>Kapitel 8: Fehlerbehebung</b>	<b>35</b>
Device not Present .....	35
IECATA lässt sich nicht flashen .....	35
<b>Kapitel 9: Sonstiges...</b>	<b>37</b>

---

9.1 Stifte .....	38
<b>Kapitel 10: BASIC</b>	<b>39</b>
10.1 .....Datei für den Diskettenwechschalter am sd2iec 39	
<b>Kapitel 11: Literaturverzeichnis</b>	<b>41</b>

# 1 Einleitung

Eine Laufwerks und Kontaktplatine für den DTV.

Zwei Laufwerke für den C64/C128.

Die Idee entstand schon vor einigen Jahren als der DTV noch im Handel angeboten wurde und die Szene noch die Möglichkeiten auslotete. Ich habe mich damals wie viele andere auch daran gemacht den DTV zu zerlegen und wollte ihn mit den möglichen Features ausstatten. Da ich zu diesem Zeitpunkt an meinem neuen Arbeitsplatz viel mit kleinen Prüfspitzen zu tun hatte, kam mir die Idee einfach eine Art Mainboard zu basteln und den DTV wie eine CPU auf diese Platine zu stecken, als Kontakte sollten die alten Prüfspitzen aus der Firma dienen. Die Positionen waren schnell ausgemessen, ein danach gebohrtes Plättchen aus Kunststoff simulierte die Platine, damit konnte man zumindest mal die Bohrpositionen kontrollieren.

Gescheitert bin ich am Platinenlayout, und so blieb das ganze Projekt mit zerlegten DTV in einer Schachtel einige Jahre liegen, bis ca. Juni 2011. Nachdem ich für den DTV ein neues Aluminium Gehäuse gefertigt hatte und meinen alten Prototyp mühevoll mit vielen Kabeln zumindest die wichtigsten Funktionen (Joystick1/2, IEC, und PS2 Port) hinzugefügt hatte wollte ich dieses Projekt wieder in Angriff nehmen. Ich habe mich in ein Layoutprogramm eingelernt, und sogar den DTV mit seinen abgreifbaren Punkten als Bauteil integriert. Die dafür notwendige Platine war relativ groß und bot noch genügend Platz für Ideen, also entschloss ich mich dazu noch ein sd2iec und ein iecata zu verbauen. Das war dann eigentlich doch zu viel für die Platine und an Aufwand, und brachte das Projekt fast ein zweites mal zum kentern.

Mit der Unterstützung einiger Arbeitskollegen und Freunden, gelang es mir die Schaltung etwas zu verbessern, ich wollte möglichst wenig heikle SMD Bauteile verwenden und alle IC's sollten austauschbar sein. Also mussten alle IC's gesockelt werden, das war vorteilhaft für verschiedene Prüfspitzen. Der erste Prototyp machte natürlich nur Schwierigkeiten und brachte mich fast um den Verstand. Ich musste mich erst mit der Mosfet Schaltung auseinan-

dersetzen, um zu verstehen, das die Signale in der Software invertiert erwartet werden. Mit einer zusätzlichen Lochrasterplatine, die als Zwischensockel für den AT-Mega diente, konnte ich die Signale invertieren. So funktionierte die Schaltung dann auch mit der original Software. Leider war eine Befestigungsbohrung für die DTV Platine falsch, deswegen konnte ich die Platine nur mit dem C64 testen, dort erwies sie sich aber dann als sehr brauchbar.

Der zweite Prototyp musste natürlich etliche Bauteile mehr unterbringen, den zusätzlichen IC, einige pullup Widerstände die ich vergessen hatte, und einen colorfix den ich auch noch machen wollte. Natürlich schaffte das der Autorouter schon lange nicht mehr, deswegen musste ich alle Leitungen mühselig selbst per Hand verlegen, das dauerte natürlich auch wieder sicher 1-2 Monate. Als ich das hatte, habe ich versucht noch die Leitungen möglichst dick zu machen, um die Lebensdauer der Leiterbahnen zu erhöhen (es könnten sich ja irgendwann der Lack ablösen, und Korrosion die Leiterbahnen zersetzen, oder Haarrisse entstehen), immerhin soll die Platine ewig halten. So gesehen ist auch die Platine nie fertig da ich immer etwas zu verbessern finde. Dennoch habe ich mich irgendwann entschlossen mal 2 Prototypen fertigen zu lassen, die Laufwerke funktionierten diesmal auf Anhieb, doch die Qualität der Platinen war sehr schlecht (Zeit für einen Herstellerwechsel) Auch das Positionieren der neuen Kontaktstifte, und deren Anzahl machen jetzt Probleme, deswegen ist es vielleicht doch besser die Stifte durch kurze Kabel zu ersetzen und fest zu verlöten, Nach dem zusammen schrauben der Platinen kann dieser Verband dann noch immer leicht in verschiedene Gehäuse verbaut werden, ein Versuch läuft diesbezüglich noch (Stand 29.3.2012).

Abschließend muss ich noch sagen das es sich als sehr gutes Laufwerk für den C64 erwiesen hat, und sich perfekt mit dem neuen Easyflash3 ergänzt. Das Easyflash kann mit dem eingebauten sd2iec per Fastloader aktualisiert und neue Software aufgespielt werden, andererseits können mit dem Easyflash3 einfach andere Schnelllade Kernals (z.B.: Jiffidos) eingebunden werden die wiederum die beiden Laufwerke beflügeln. Dabei kann das IECATA Als echtes Festplattenlaufwerk oder mit Adapter als Compact Flash Laufwerk verwendet werden, vor-konfiguriert als Laufwerk 9 und 10 verträgt sich das Laufwerk mit der 1541 (Adresse8)

# 2 Software

## 2.1 Arbeiten mit dem sd2iec Laufwerk

### Verzeichnisse wechseln (CMD-Syntax)

- \* OPEN1,8,15,"CD/SOMEDIR/":CLOSE1 - change to SOMEDIR (relative)
- \* OPEN1,8,15,"CD//":CLOSE1 - change to root directory
- \* OPEN1,8,15,"CD//SOMEDIR/":CLOSE1 - change to SOMEDIR (in root)
- \* OPEN1,8,15,"CD:?":CLOSE1 - go to parent directory

### D64 benutzen (am Beispiel Inhaltsverzeichnis aufrufen)

- \* OPEN1,8,15,"CD//SOMEDIR/:SOMEGAME.D64":CLOSE1 und dann LOAD"\$",8 und LIST funktioniert immer für das D64 Image "Somegame.D64" im Verzeichnis "Somedir"
- \* OPEN1,8,15,"CD:SOMEGAME.D64":CLOSE1 wenn das D64 im aktuellen Verzeichnis liegt
- \* OPEN1,8,15,"CD:?":CLOSE1 - D64 wieder verlassen
- \* @CD//SOMEDIR/:SOMEGAME.D64 und LOAD"\$",8 und LIST /bzw. jiffy Befehl fürs DIR funktioniert mit Jiffy
- \* @"CD//SOMEDIR/:SOMEGAME.D64 und LOAD"\$",8 und LIST /bzw. jiffy Befehl fürs DIR funktioniert mit S-Jiffy

**Firmwareversion abfragen (nur direkt nach dem Einschalten möglich)**

- 10 OPEN15,8,15:INPUT#15,a\$,b\$,c\$,d\$
- 20 CLOSE15
- 30 PRINT A\$,B\$,C\$,D\$
- run

```
5 REM DIRECTORY CHANGER FOR SD2IEC
```

```
10 INPUT"Target directory name";D$
```

```
20 C$="cd:"+D$
```

```
30 OPEN1,8,15,C$
```

```
40 CLOSE1
```

D64 Diskette (Diskettenverzeichnis) erzeugen.....eventuell nicht mit v0.10.3 Firmware

**D64 Diskettenverzeichnis erzeugen**

```
1 10 open1,8,3,"empty.d64,p,w"
2 20 open2,8,15,"p"+chr$(3)+chr$(255)+chr$(170)+chr$(2)
3 30 print#1,"x";
4 40 close1
5 50 print#2,"cd:empty.d64"
6 60 print#2,"n:empty,00"
7 70 print#2,"cd_"
8 80 close2
```

D81 Diskette (Diskettenverzeichnis) erzeugen

Anbei die Anpassung des Programmes für D81-Dateien (der Unterstrich in Zeile 70 muss durch den Pfeil nach links ersetzt werden):

**D81 Diskettenverzeichnis erzeugen**

```
1 10 open1,8,3,"test.d81,p,w"
2 20 open2,8,15,"p"+chr$(3)+chr$(255)+chr$(127)+chr$(12)
3 30 print#1,"x";
4 40 close1
5 50 print#2,"cd:test.d81"
6 60 print#2,"n:leer,00"
7 70 print#2,"cd_"
8 80 close2
```

Noch ein versuch.....

Das gleiche gilt natürlich auch für ein D71 Image.

Hier ein fertiges Programm für alle drei Images.

**D71 Diskettenverzeichnis erzeugen**

```
1 100 print"create an empty disk image?":print
```

```
2 110 print"press 6 for d64 image"
3 111 print"      7 for d71 image"
4 112 print"      8 for d81 image"
5 120 print"      b to basic"
6 130 print"      r to reset"
7 140 geta$:ifa$=""then140
8 150 ifa$="6"thena$="64":a=255:b=170:c=2:goto190
9 150 ifa$="7"thena$="71":a=255:b=85:c=5:goto190
10 150 ifa$="8"thena$="81":a=255:b=127:c=12:goto190
11 160 ifa$="b"thenend
12 170 ifa$="r"thensys64738
13 180 goto140
14 190 open1,8,3,"empty.d"+a$+",p,w"
15 200 open2,8,15,"p"+chr$(3)+chr$(a)+chr$(b)+chr$(c)
16 210 print#1,"x";
17 220 close1
18 230 print#2,"cd:empty.d"+a$
19 240 print#2,"n:empty,00"
20 250 rem print#2,"cd_"
21 260 close2
22 270 end
```

# 3 Platinenaufbau

## 3.1 Anschlüsse und Jumper

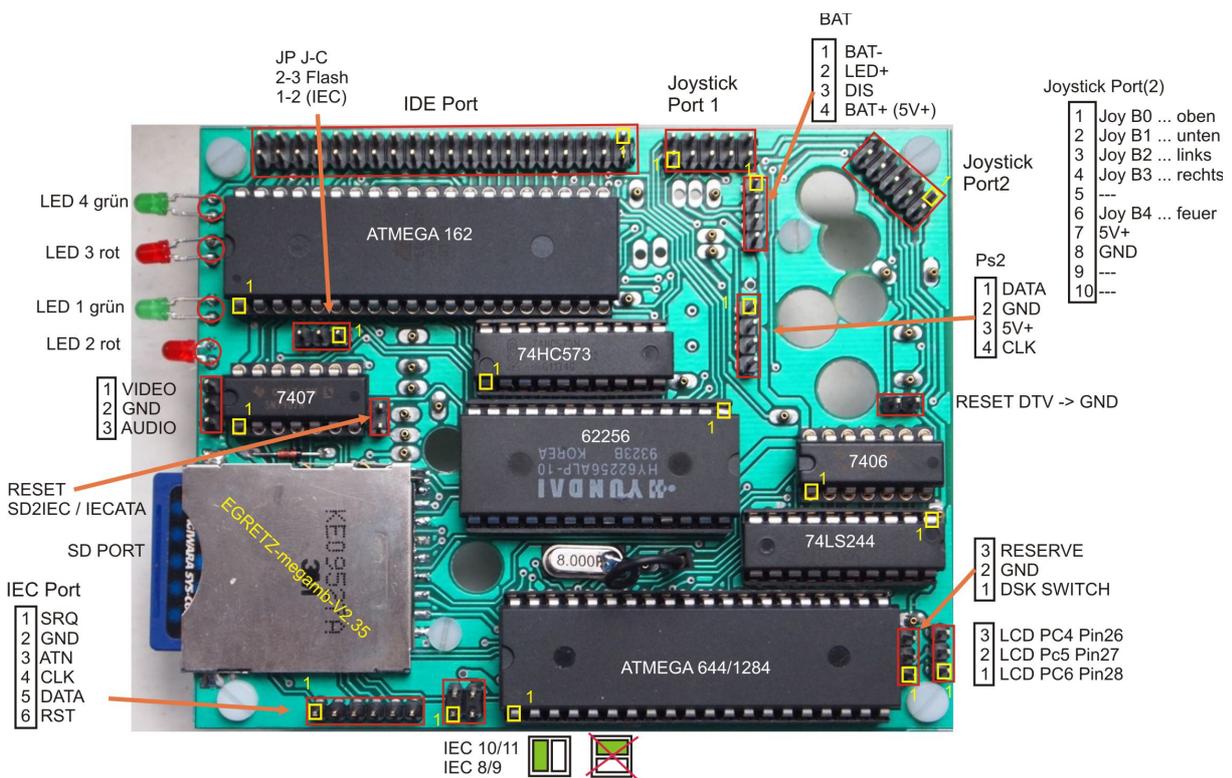


Abbildung 3.1: Anschluß und Jumper Übersicht

# 4 Aufbauanleitung

## 4.1 SMD Bauteile auf der Unterseite

Als erstes sollten die SMD Bauteile auf der Unterseite der Platine gelötet werden.

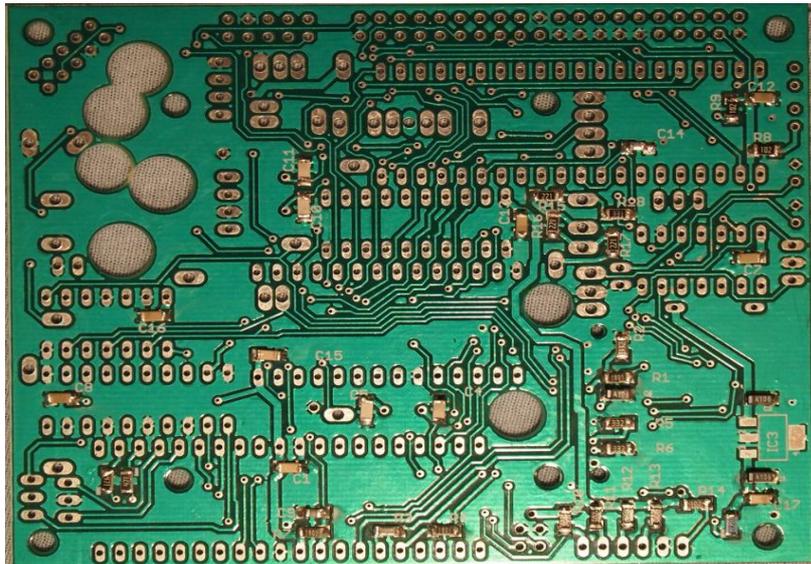


Abbildung 4.1: SMD Widerstände und Kondensatoren

Die insgesamt 39 SMD Bauteile sind:

Kondensatoren	
100nF	C1, C7, C8, C12, C13, C15, C16, C17
10yF	C2, C3, C6,
22pF	C4, C5, C10, C11,
220pF	C9, C14,

Widerstände	
1k8	R1, R3, R4, R7
3k3	R2, R5, R6
1k	R8, R9, R10, R11, R12, R13, R14
220	R15, R16, R17
330	R18
2,2	R20
470	R30, R31

Sonstiges	
Spannungswandler	LM2936MP-3.3

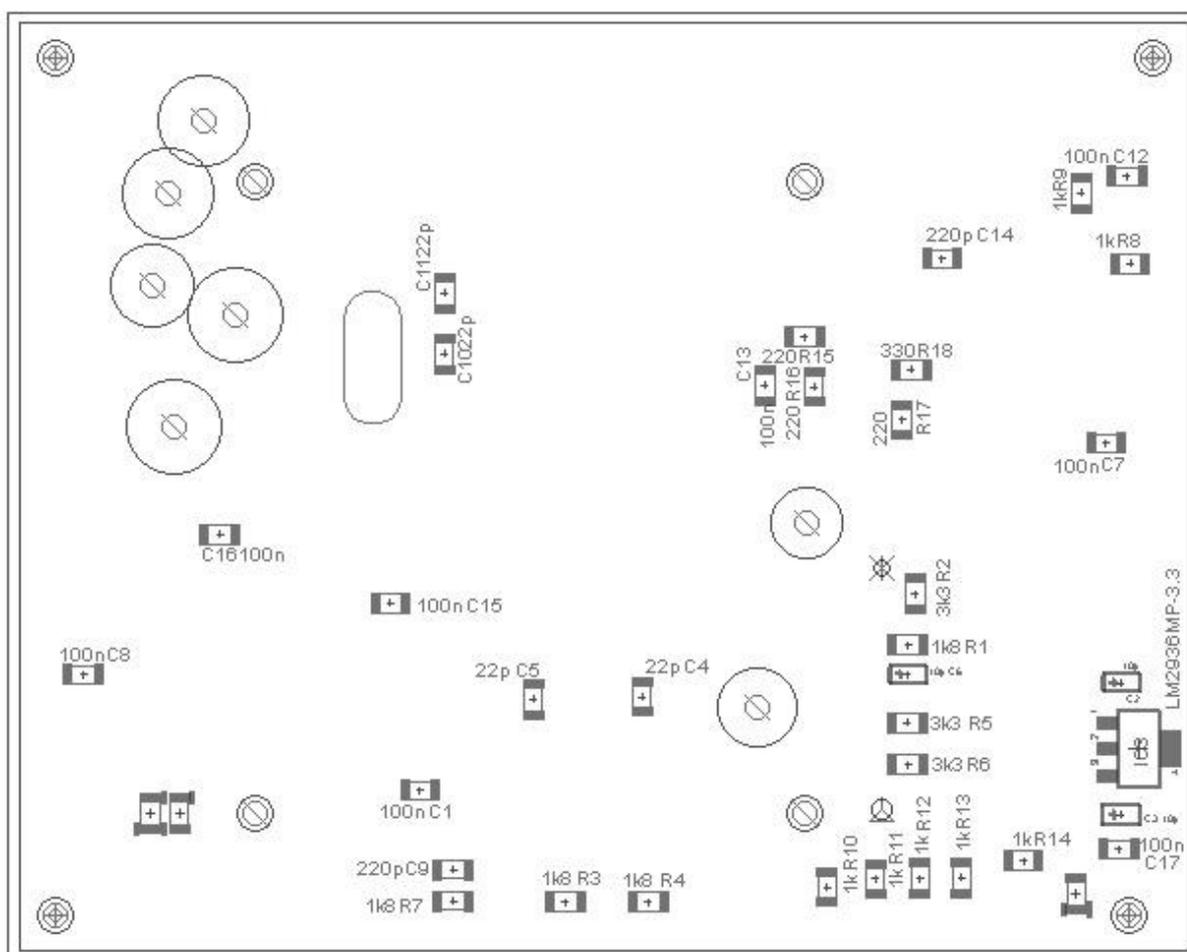


Abbildung 4.2: SMD Bauteilpositionen auf der Unterseite

### SMD Löten einfach so...

Wer sich fragt wie er denn nur die SMD Teile ohne geeignete SMD-Paste Löten soll, kann ich nur sagen wie ich das mache. Ein Pad mit etwas Lötzinn verzinnen, SMD Bauteil mit der Pinzette auf das Pad positionieren und halten. Mit dem LötKolben beides heiß machen bis sich

der SMD Bauteil mit dem Lötzinn verbindet, danach einfach die zweite Seite wie jeden anderen Bauteil verlöten

## 4.2 Kontaktstifte

Die Kontaktstifte werden ebenfalls auf der Unterseite gelötet. Da die Stifte an ihrem Ende ein Loch haben, in das kein Lötzinn gelangen darf, müssen sie etwas weiter durch gesteckt werden, als Abstandshilfen dienen die M 2,5 Schrauben, diese werden vorübergehend in die 4 Löcher ganz außen gesteckt, und mit den passenden Muttern befestigt. Dadurch entsteht ein kleiner Spalt von ca. 2mm zwischen Platine und Auflagefläche (natürlich sollte die Auflagefläche möglichst eben sein). Nun können die Stifte in die benötigten Löcher gesteckt und verlötet werden.

Das Ausrichten der Stifte ist nicht gerade die einfachste Übung, da zumindest bei meinen Prototypen der Platinenhersteller sehr ungenau gebohrt hat, deswegen müssen die Stifte am besten mit einem Druckbleistift (ohne Miene) ausgerichtet werden,

Mit dem Druckbleistift kann der Kontaktstift während des Lötvorganges gehalten werden, ohne das man sich die Finger verbrennt.

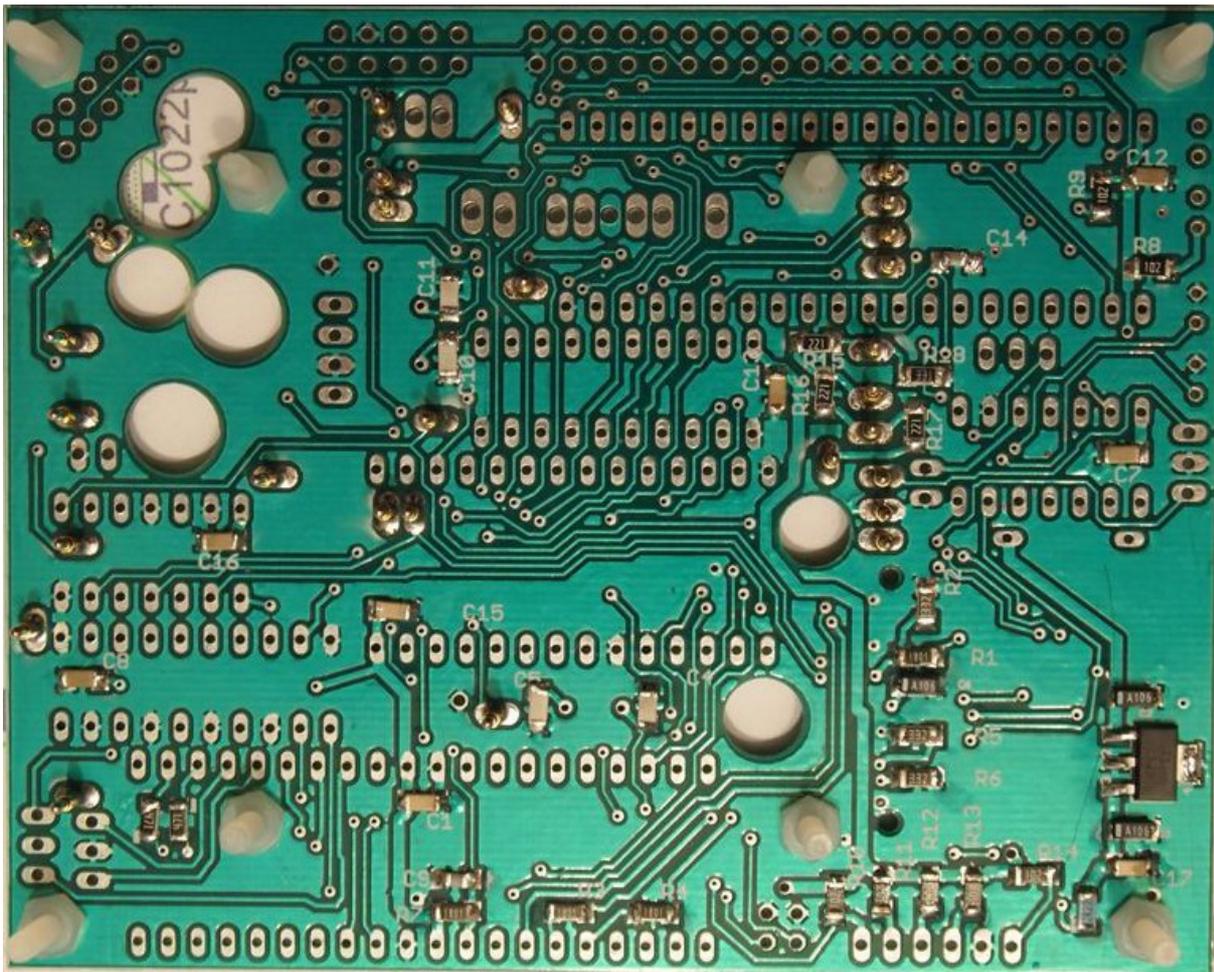


Abbildung 4.3: Kontaktstifte

Die Kontaktstiftlöcher für den Userport sind nur optional und werden nicht bestückt

### 4.3 SD Card Reader

Nachdem alle Stifte positioniert sind können die Bauteile auf der Oberseite bestückt werden, als erstes werden die Diode D1 und der SD Card Reader verlötet.

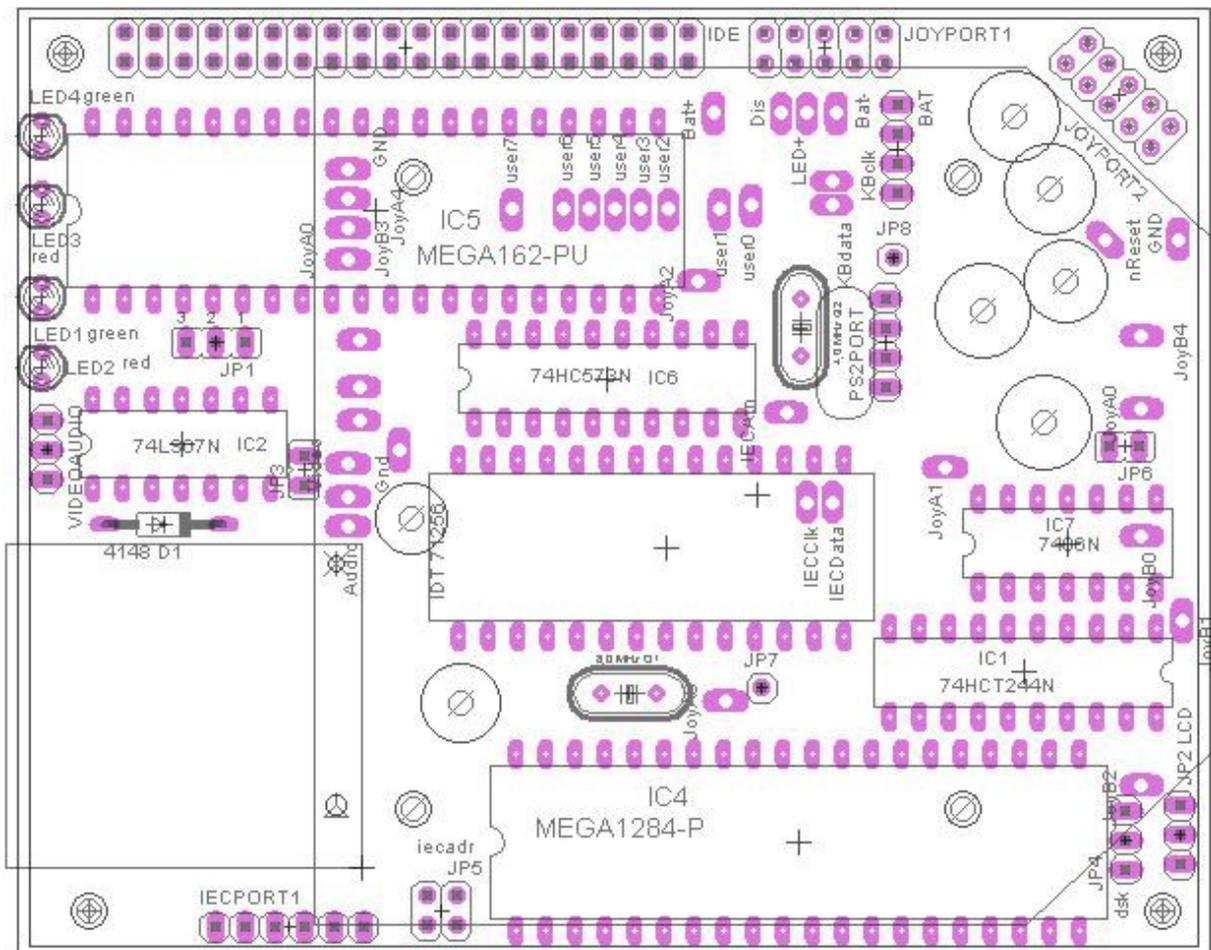


Abbildung 4.4: Bauteile Oberseite

#### 4.4 IC Fassungen und Quarz

Nachdem die Diode und der Card reader eingelötet sind können die IC Fassungen verlötet werden, diese müssen teilweise für die Stifte und Befestigungsschrauben freigestellt werden, siehe dazu die Abbildung unten.

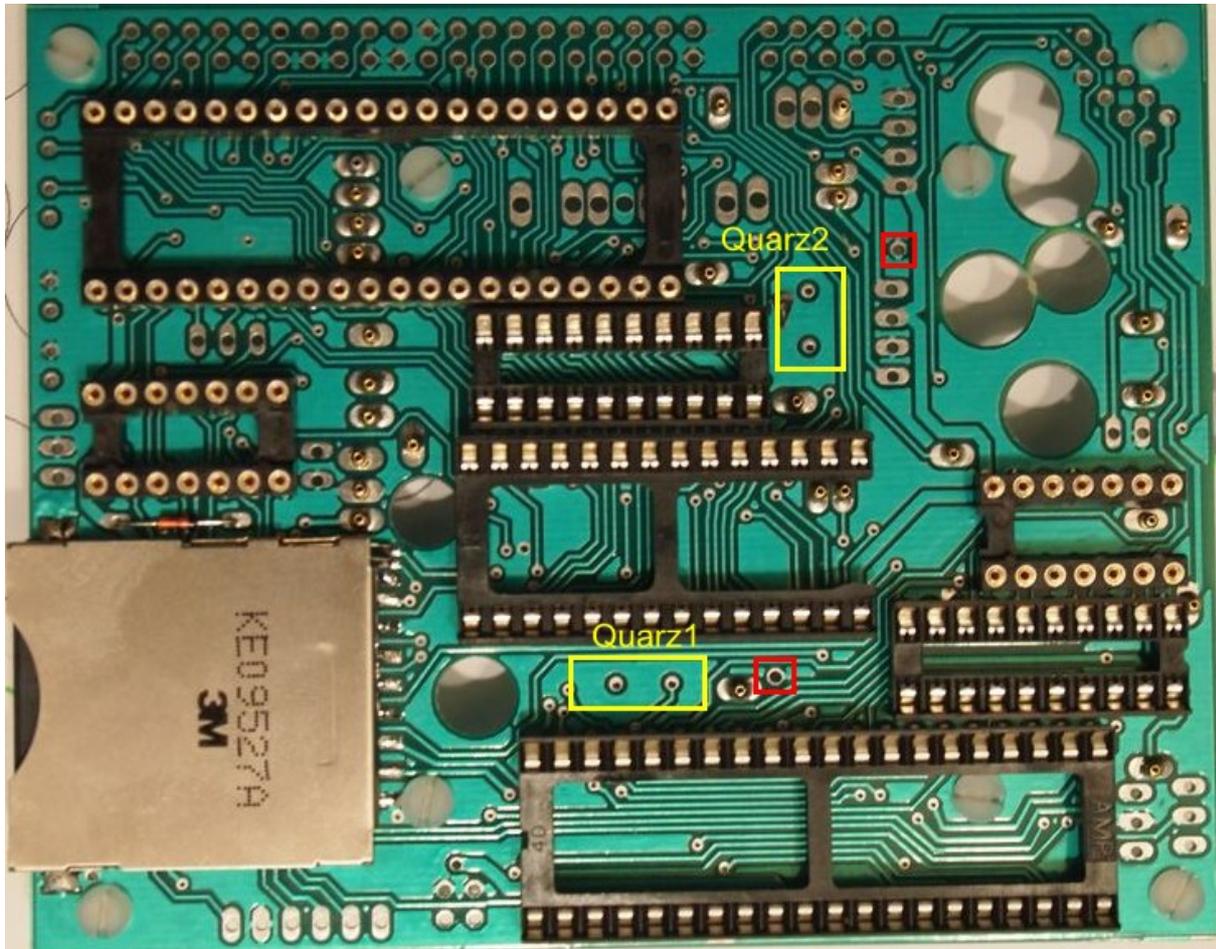


Abbildung 4.5: IC Socket Oberseite

Nach den IC Fassungen wird der 8Mhz Quarz[1] eingelötet., der für das SD2IEC notwendig ist.

Optional kann auch für das IECATA ein (8Mhz) Quarz[2] eingelötet werden. Das Quarzgehäuse kann über den freien Pin (rot markiert) mit Masse verbunden werden.



## 4.5 Stiftleisten verlöten

Als nächstes werden noch die Stiftleisten für die Ports und die Jumper eingesetzt und angelötet.

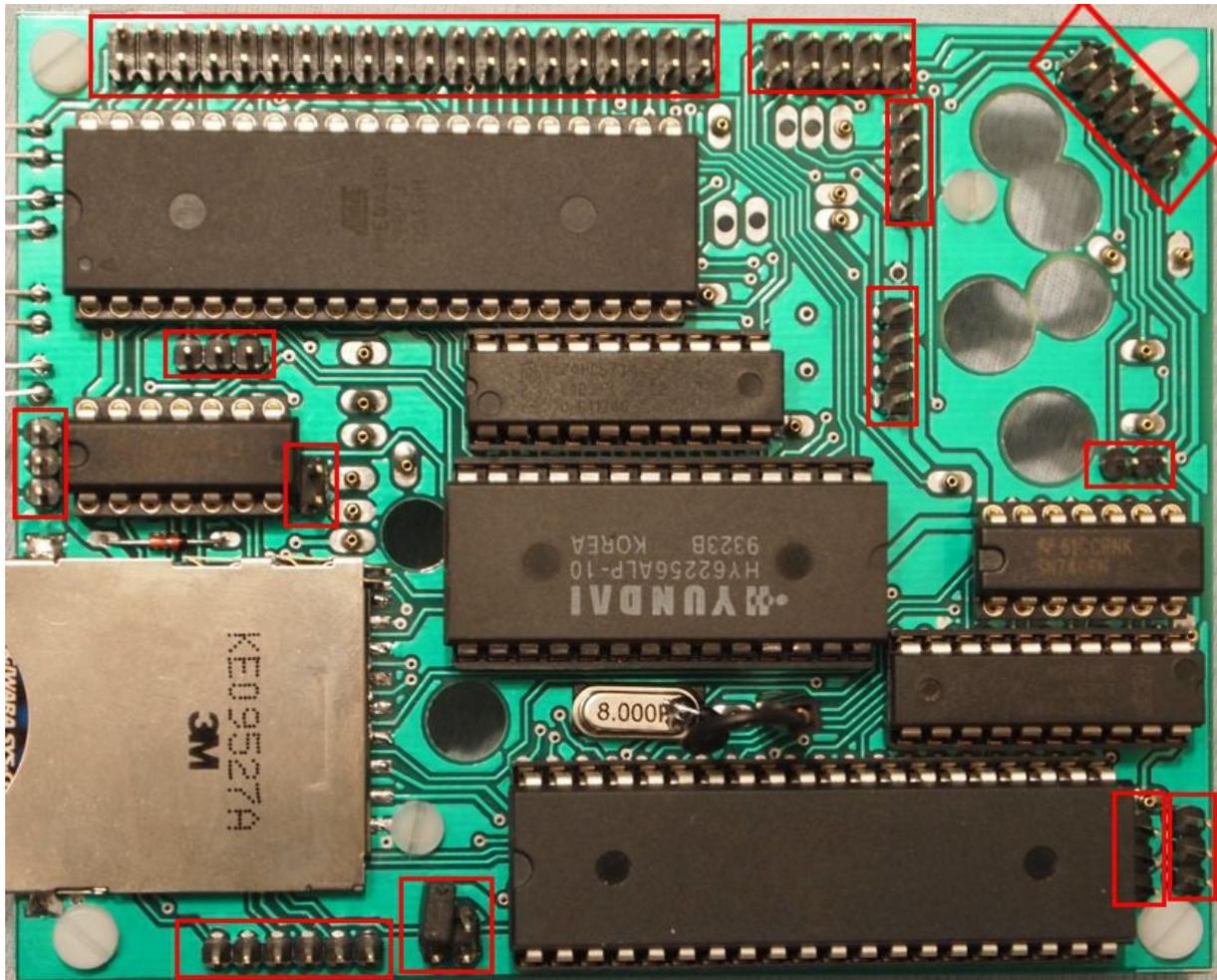


Abbildung 4.6: Stiftleisten und Ports verlöten

In alten PC Gehäusen finden sich oft passende Verbindungsleitungen die einfach auf die Stiftleisten aufgesteckt werden können.

# 5 Aufbau für Baugruppen

In diesem Kapitel möchte ich nochmal auf die einzelnen Baugruppen der Platine eingehen. Sollte es einmal zu Problemen kommen ist es so leichter möglich die Ursache ausfindig zu machen. Auch ist es möglich die Platine nur teilweise aufzubauen, wenn nicht alle Funktionen benötigt werden.

## 5.1 DTV Funktionen

Durch das Einlöten von Federkontaktstiften, oder das direkte verlöten der Kontaktplatine mit der DTV Platine können folgende Funktionen genutzt werden:

- Joystick 1
- Joystick 2
- Color fix (Version Spiff)
- Reset DTV
- PS2 Port (Tastaturanschluss)
- IEC Port (zB. Für Diskettenlaufwerke wie z.B. 1541)

Diese Funktionen können nach dem Verlöten an den Stiftleisten abgegriffen werden, die DTV Platine wird mit der Adapterplatine verschraubt, dadurch werden die Kabel gegen Zug entlastet. Weiteres löten entfällt, da die Kabel an der Adapterplatine an und abgesteckt werden können, das schon die DTV Platine.

## 5.2 SD2IEC

Das SD2IEC ist eine abgeänderte Shadowwolf Version, und ist mit dieser Softwaretechnisch voll Kompatibel. Die Änderungen betreffen nur die Kommunikation mit dem C64. Die originale Version benutzt dazu Mosfets, meine Version dagegen die gleichen IC's die auch in den

Commodore Laufwerken verwendet werden ( 7406 /7407). Das ist durch die große Platine möglich, und erschien mir als bessere Variante. Außerdem arbeitet mein SD Card reader mit 3,3V anstelle der 3,0V (Die Spezifikation der SD Karte geht von 2,7 bis 3,6V.)und der AT-Mega mit 5V das erleichtert ein anschließen eines LCD-Displays.

## Bauteilliste:

IC4	AT-Mega 1284	Optional AT-Mega 644
IC1	74HCT244N	
IC7	7406N	
IC3	LM2936MP-3,3V	
SD CON 3M	SD-Connector104G-TAAO-R	
Q1	8 Mhz Quarz	
JP2,JP4	3x1 pol Stiftleiste	LCD,dsk
IECPOR1	6x1 pol Stiftleiste	
JP5	2x2 pol Stiftleiste	lecadresse
R1,R3,R4,R7	1k8	
R2,R5,R6	3k3	
R20	2,2	
R10,R11,R12,R13,R14	1k	Pulup für C64
R30,R31	470	
D1	4148	
LED1,LED2	grün,rot	
C1,C8,C16,C17	100nF	
C2,C3,C6	10yF	
C4,C5	56pF	
C9	220pF	

### 5.3 IEC ATA

Das verbaute IEC ATA (V2) ist kompatibel zur NLQ- HD Firmware.

Bauteilliste:

IC5	ATMEGA 162PU	
IC6	74HCT573	
IC8	62256	Auf Position/Richtung achten
IC2	7407N	
Q2	Quarz	Optional
JP1	3x1 pol Stiftleiste	Flash
JP3	2x1 pol Stiftleiste	RESET
IDE	2x20 pol Stiftleiste	IDE Anschluß
R8,R9	1k (1206)	Für Led3/ Led4
C14	220p (1206)	Für Resetleitung
C10,C11	22p (1206)	Optional für Quarz
C12,C7,C13,C15	100n (1206)	Kondensatoren für ICs
LED3	3mm Led	rot
Led4	3mm Led	grün
Sockel	40pol,28pol,14pol,20pol	Würde ich dringend empfehlen (28pol in breiter Bauform)

Der Aufbau kann so wie in unter Absatz 2.0 (Aufbauanleitung) erfolgen, wichtig ist nur das die Led mit dem Massepin weiter weg vom SD-Slot sind, pos ist näher zum SD-Slot. Der 62256 und der 74HCT244N Sitzen verkehrt zu den anderen ICs!

Der ATMEGA 162PU muss mit dem C64 geflasht werden (mit dem DTV ist das nicht möglich, da diesem eine Leitung am IEC Bus fehlt).

### 5.4 Lumafix (Spiff)

Die Positionen an denen die DTV Platine Kontaktiert wird ist an den Widerständen R14, R16, R20 und R24, der Rest wird auf dem Egretzmegamb erledigt. Dadurch können auch größere SMD Widerstände benutzt werden, und man muss weniger aufpassen beim löten.

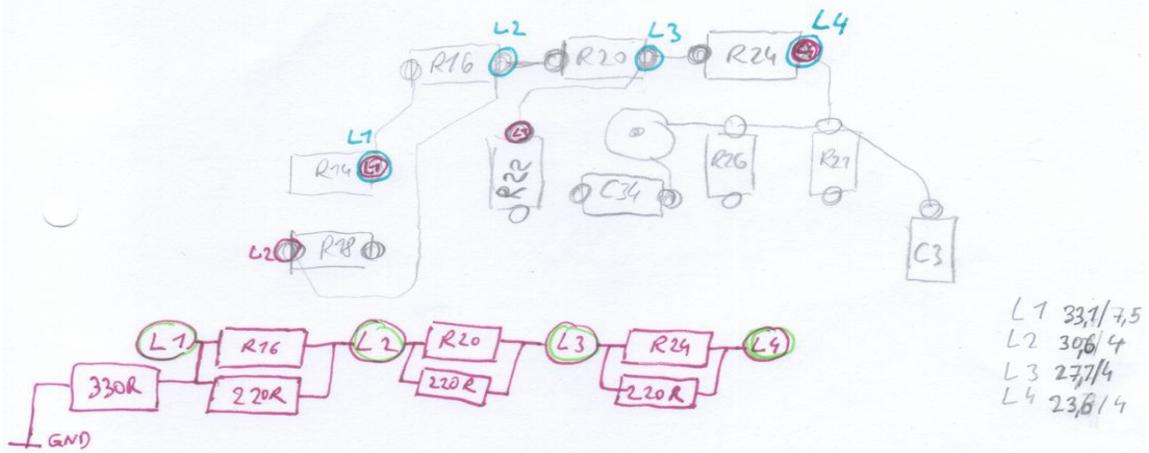
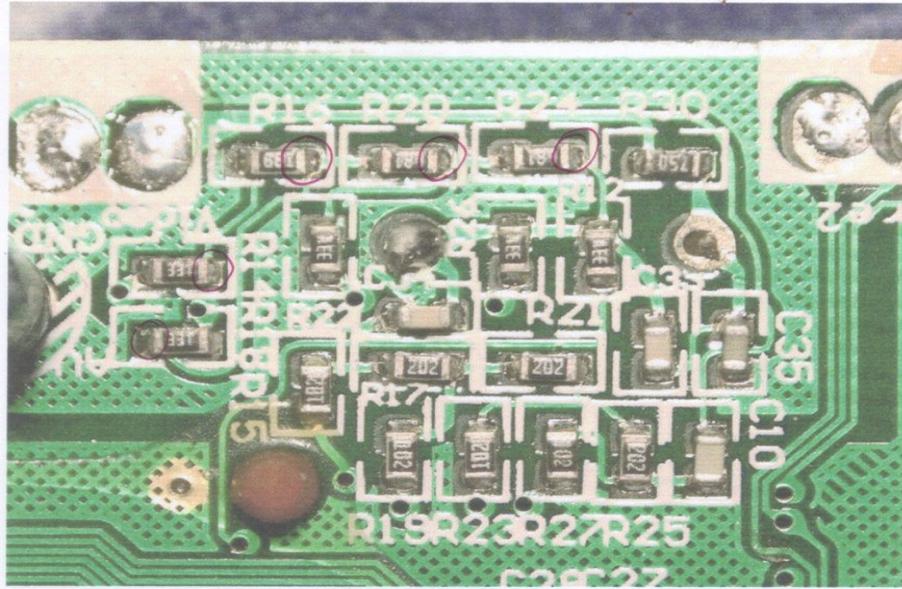


Abbildung 5.1: Spiff Lumafix DTV Platine

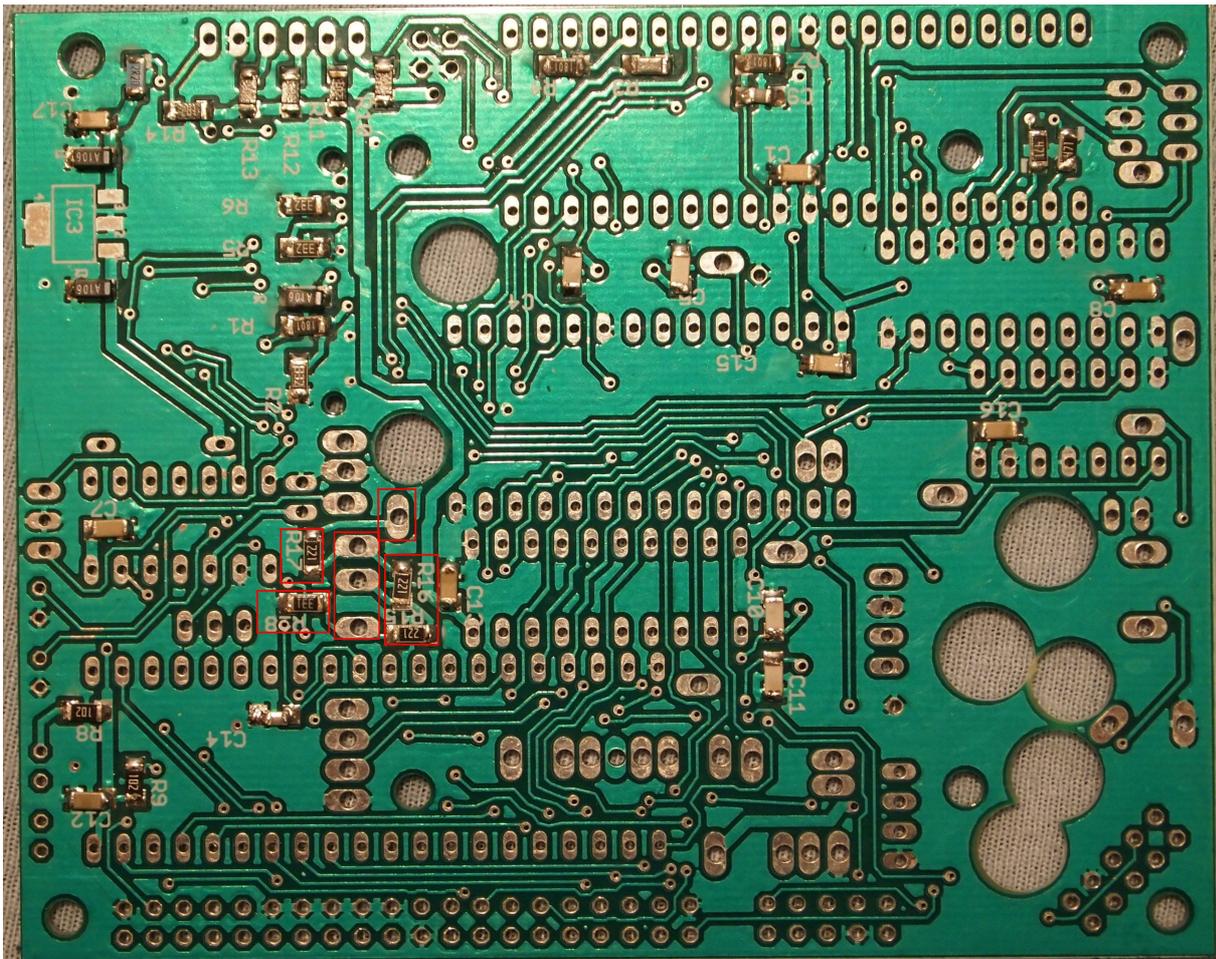


Abbildung 5.2: Lumafix Widerstände und Pinpositionen

Auf der Adapterplatine sind das die angezeichneten Widerstände und die 4 Pins gleich in deren Nähe.

R15	220
R16	220
R17	220
R18	330

# 6 Gehäuse

**E**in langlebiges elektronisches Gerät sollte als Basis ein solides Gehäuse haben, das die Platine und die anderen elektronischen Komponenten vor Umwelteinflüssen schützt.

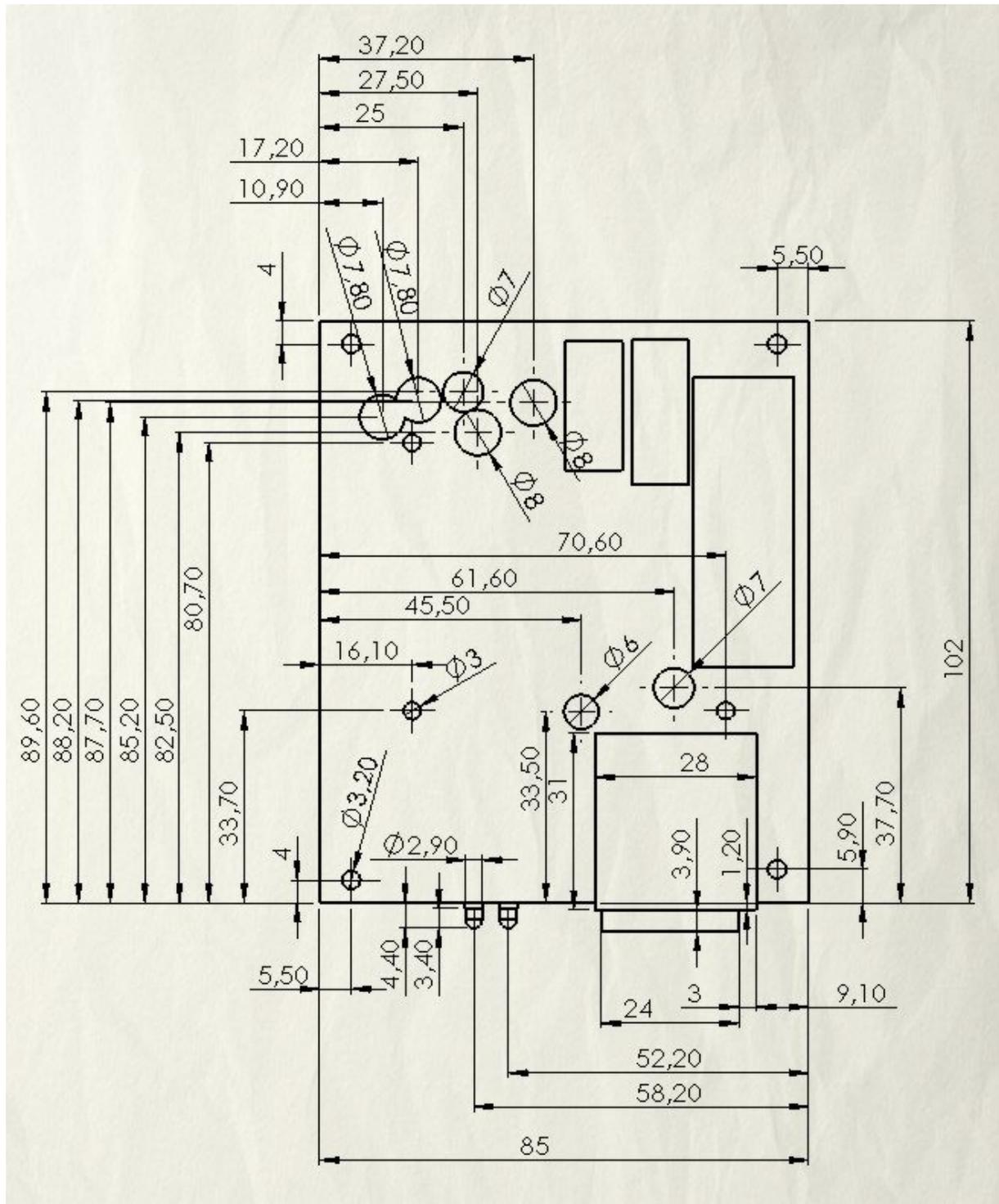
## 6.1 Kleinspannung

Als **Kleinspannung**, umgangssprachlich auch *Niedervolt* oder *Schwachstrom*, werden in der Elektrotechnik Wechselspannungen (AC) bis 50 Volt Effektivwert und Gleichspannungen (DC) bis 120 Volt bezeichnet. Sie ist ein Teilbereich der Niederspannung. Diese Spannungen gelten beim Berühren für erwachsene Menschen als nicht lebensbedrohlich und entsprechen der maximalen dauernden Berührungsspannung beziehungsweise Fehlerspannung.

Die europäische Niederspannungsrichtlinie gilt daher auch für den oberen Bereich der Kleinspannung (AC über 50 Volt Effektivwert und DC über 75 Volt).

Bei Spannungen unter 25 Volt AC oder 60 Volt DC kann gänzlich auf einen Schutz gegen Berühren verzichtet werden; diese Spannungen gelten auch für Tiere und Kinder als ungefährlich[1]. (Quelle: Wikipedia)

## 6.2 Mechanische Abmessungen der Platine



### 6.3 DTV-Gehäuse

Kühlrippengehäuse 130040  
Conrad Bestellnummer 522945

### 6.4 ND1581 Gehäuse

Aluminiumgehäuse mit Kunststofffront Design ähnlich der alten 1581, mittlerweile mit Taster.

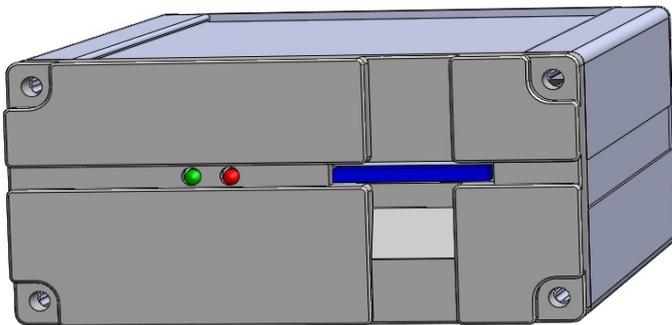


Abbildung 6.1: ND 1581 Gehäuse für C64

### 6.5 Egretz 15Mx81 - Gehäuse (nur konstruiert Stand 03.09.2013)

Eine Weitere Möglichkeit ist die Verwendung der Platine als Doppellaufwerk für den C64 bzw. C128, Dazu habe ich ein Gehäuse entwickelt, das dem 1581 ähnlich ist, Das SD2IEC ist unten verbaut, und das IECATA wird mit einer Adapterplatine als Compact Flash Laufwerk verwendet, und wird oben herausgeführt, Daneben ein 5mm Power Led und unten die Status Led's der beiden Laufwerke.

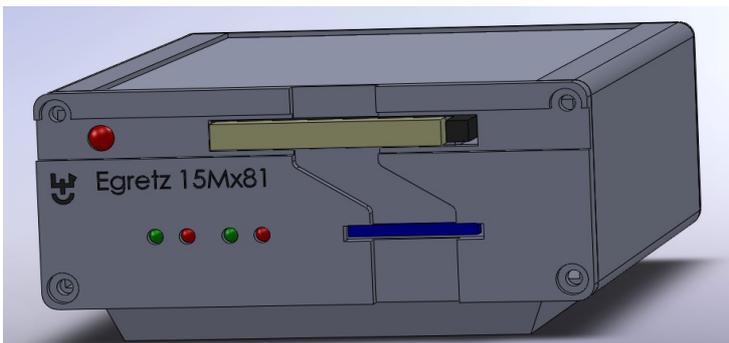


Abbildung 6.2: Egretz 15Mx81 Gehäuse für C64

# 7 NLQ-HD-Firmware

**D**ie NLQ-HD-Firmware ist Freeware. Die Files sind angehängt. Sie sind für die IEC-A-TA-V2-Hardware und dürfen nicht auf eine andere Hardware geflasht werden. Die Installation ist etwas fummelig:

## 7.1 Kurzanleitung

Zuerst musst du C-NLQ-FLASHER starten und NH1EAT0.BIN flashen.

Danach musst du irgendeine HD anschließen und C-NLQ-CAL&CON starten. Falls das Calibrationbyte falsch ist, wählst du am besten 'A' für automatische Berechnung. Im Hauptmenü tippst du später am besten 'W' für Write EEPROM und lädst ein Default-Config-File mit der gewünschten Devicenummer von Disk und schreibst es ins EEPROM. Danach muss vermutlich wieder das Calibrationbyte mit 'A' für automatisch berechnet werden. Sodann drückst du 'E' für EEPROM-configuration und speicherst den soeben berechneten Wert ins EEPROM, damit er beim nächsten Einschalten stimmt. Wenn du willst, dann kannst du die Konfiguration mit 'E' anschauen und bei Bedarf die emulierte Geräteadresse auf einen Wert zwischen 8 und 30 einstellen.

Dann kommt es darauf an, ob du eine HD (528MB-128GB) hast, die bereits eine FAT32-Partition hat oder ob nicht.

Wenn du eine HD mit FAT32-Partition hast, dann müsste die HD sofort benutzbar sein.

Wenn du keine so formatierte HD hast, dann musst du am C64 C-NLQ-PAR&FOR starten. Vorhandene Partitionen evtl. mit 'D' löschen, dabei noch 'destroy' eintippen (!! alle Daten gehen verloren!!). Danach mit 'C' eine neue Partition erzeugen, wobei du am besten die Vorgaben mit 'ok' übernimmst.

Dann solltest du eigentlich schon auf die HD zugreifen können.

Wenn du Unterverzeichnisse anlegen willst, dann geht das mit C-NLQ-MD&RD.

Die NLQ-Firmware hat den Vorteil, dass die HD im FAT32-Format beschrieben wird. Dadurch können die am C64 gespeicherten Files später am IBM-PC weiter benutzt werden. Es ist ein Hobbyprojekt und die Benutzung erfolgt auf eigene Gefahr (Datenverlust).

NLQ-HD kann:

-selektives Directory (open2,8,0,"\$\*=b" ('b' = branch = directorys))

-load (load"abc",8)

-save (save"abc",8)

-es kommt auch mit den Zeichen <Leerzeichen> <Punkt> '/' '<' '>' <Pfund> im Filenamen klar

-Files zum Lesen oeffnen (open2,8,2,"abc":get#2,A\$)

-Files zum Schreiben oeffnen (open2,8,2,"abc,p,w":print#2,"a";)

-initialize (open15,8,15,"i")

-scratch (open15,8,15,"s:abc")

-rename (open 15,8,15,"r:xyz=abc")

-change directory (open15,8,15,"cd:abcd" bzw "cd<- oder cd:<- fuers Parentdirectory)

-ui, uj = reset (open15,8,15,"ui") (das aktuelle Directory bleibt erhalten)

-aui = additional ui = reset (open15,8,15,"aui") (das aktuelle Directory wird zurueckgesetzt)

Die FW unterstützt JiffyDos, d.h. dass durch den Einbau eines JD-Eprom in den C64/C128 das Ganze viel schneller wird.

Falls du Probleme hast, dann schau mal bei [www.forum64.de](http://www.forum64.de) vorbei, stell dort eine Frage oder frag mich direkt.

## 7.2 Ausführliche Anleitung

### **1. FLASHEN (OBLIGATORISCH)**

*Zuerst musst du C-NLQ-FLASHER starten und NH1IEAT0.BIN flashen.*

Das macht man wie folgt:

- C64 und Floppy ausschalten

- IEC-ATA an Strom und seriellen Port anschließen (ohne HDD)

- Jumper J-C auf 1-2 stecken, Jumper RESET offen lassen

- C64 und Floppy anschalten

- "C-NLQ-FLASHER" von Diskette laden und starten

- Dem Bildschirmdialog folgen und die Datei "NH1IEAT0.BIN" auswählen und laden

- Jetzt Jumper J-C auf 2-3 stecken, Jumper RESET schließen

- Die Sicherheitsabfragen des Programms bestätigen

- Bei der Auswahl zwischen "C" und "D" beim ersten Bespielen "C" wählen, um alles neu aufzusetzen.

- Den Flashvorgang bestätigen und zuschauen, bis es fertig ist

- Jetzt Jumper J-C auf 1-2 stecken und Jumper RESET wieder öffnen
- Das Programm verlassen

## **2. KALIBRIEREN (OBLIGATORISCH)**

*Danach musst du irgendeine HD anschließen und C-NLQ-CAL&CON starten. Falls das Calibrationbyte falsch ist, wählst du am besten 'A' für automatische Berechnung.*

Das Kalibrationsbyte ist fast immer falsch, also

- Das Programm "C-NLQ-CAL&CON" laden und starten
- Die Abfragen beantworten (pal/ntsc, iecata, Oszi j/n) - normalerweise mit p, y, n abzuschließen
- Es erfolgt ein erster Kalibrationslauf (der Bildschirm flashed dabei), der mit dem Ergebnis endet, dass das Ding kalibriert werden muss
- "A" drücken, um die automatische Kalibrierung zu starten
- Das Programm durchläuft nun mehrere Kalibrierungsläufe, bis ein guter Wert gefunden wurde und das Programm stoppt
- Jetzt noch einmal bestätigen und es wird ein letzter Kalibrierungslauf gemacht
- Das Programm geht nun in den Menü-Modus

## **3. KONFIGURIEREN (OBLIGATORISCH)**

*Im Hauptmenü tippst du später am besten 'W' für Write EEPROM und lädst ein Default-Config-File mit der gewünschten Devicenummer von Disk und schreibst es ins EEPROM.*

Wir laden nun ein Default-Config-File von der Diskette, um die Einstellungen nicht alle von Hand machen zu müssen:

- "W" eingeben, um in den Ladedialog zu gelangen
- Falls die Diskette nicht in Laufwerk 8 liegt, "D" drücken und den entsprechenden Laufwerksbuchstaben eingeben
- Jetzt "\$" drücken, dann die vier Zeichen löschen und RETURN drücken
- Das Verzeichnis wird angezeigt. Prüfen, ob die DNxx.NH1 Dateien dabei sind
- Standardmäßig werden fünf Default-Config-Files mitgeliefert für die Laufwerksnummern 8 bis 12 (DN08 bis DN12)
- "F" drücken und den Namen des gewünschten Files eingeben (z.B. "DN12")
- Das Default-ConfigFile wird nun geladen und in den AVR geschrieben
- Das Programm geht nun wieder in den Menü-Modus

*Danach muss vermutlich wieder das Calibrationbyte mit 'A' fuer automatisch berechnet werden.*

- Nach dem Schreiben des Default-Config-Files prüft das Tool die Kalibrierung des AVR erneut, da der Wert durch das Default-File wieder auf 255 gesetzt wurde
- Also wieder mit "A" einen automatischen Kalibrierungslauf starten, wie oben beschrieben
- Das Programm geht nun wieder in den Menü-Modus

**Sodann drückst du 'E' für EEPROM-configuration und speicherst den soeben berechneten Wert ins EEPROM, damit er beim nächsten Einschalten stimmt.**

- Als erstes wird der Kalibrierungswert angezeigt, der gerade ermittelt wurde - mit RETURN bestätigen.
- Jetzt alle weiteren Punkte mit "N" überspringen bis zur letzten Abfrage
- Hier den Punkt "C" auswählen, um die Konfiguration ins RAM zu speichern.
- Das Programm geht nun wieder in den Menü-Modus

**Wenn du willst, dann kannst du die Konfiguration mit 'E' anschauen und bei Bedarf die emulierte Geräteadresse auf einen Wert zwischen 8 und 30 einstellen.**

Das geht wie folgt:

- Erneut alle Menüpunkte mit "N" überspringen bis Du zu dem Punkt "emulated device entry number 0" kommst
- Jetzt "Y" eingeben
- "E" für Enable eingeben
- Bei der "emulated devicenumber" die gewünschte LW-Nummer eintragen (8 bis 30) und mit RETURN bestätigen
- Alle weiteren Menüpunkte wieder mit "N" für No überspringen
- Am Ende erneut mit "C" bestätigen, um die Konfiguration wieder in das RAM zu speichern
- Das Programm geht nun wieder in den Menü-Modus

#### **4. STATUSMELDUNG ANPASSEN (OPTIONAL)**

Nun können wir noch die Statusmeldung des IEC-ATAv2 verändern (dies ist allerdings optional). Dazu müssen die Fehlermeldungen manuell geladen werden. Eigentlich sind diese bereits in den Default-Konfigurationsdateien enthalten, daher ist dies wirklich nur notwendig, wenn man die Statusmeldung ändern will:

- Dazu wählt man den Menüpunkt "M"essages aus.
- Im ersten Schritt kann man nun die Statusmeldung des IEC-ATA anpassen (z.B. "MEIN IEC-ATAv2")
- Dann wird die Datei "NH-ERRORMESSAGES.SEQ" geladen und in den AVR geschrieben
- Das Programm geht nun wieder in den Menü-Modus

#### **5. KONFIGURATION SPEICHERN (EMPFOHLEN)**

Man sollte zu diesem Zeitpunkt die nun erfolgte Konfiguration abspeichern, um beim nächsten Mal diese als Default-Wert laden zu können:

- Menüpunkt "S" auswählen
- Laufwerk zum Speichern mit "D" auswählen
- Mit "F" den Dateidialog öffnen und einen Namen vergeben (z.B. "X11")
- Die Konfigurationsdatei wird nun unter dem Namen X11.NH1 gespeichert und kann beim nächsten Mal anstelle der Defaultkonfiguration geladen werden (s.o.)

- Das IEC-ATAv2 ist jetzt soweit eingerichtet
- Man kann das Programm nun mit "Q" verlassen

## **6. BETRIEB**

---

### Zitat von »NLQ«

*Wie im Tread*  
*AVR removed = Serial Port hangs not*  
*ersichtlich, ist es absolut wichtig den/die Jumper der HD auf Master zu stellen. Falls die HD irrtümlich der Slave ist, dann hängt die NLQHD-Firmware im Reset in einer Endlosschleife, die rote LED leuchtet und der Bus ist blockiert.*

NLQ  
Near Letter Quantity  
Jochen Adler  
NLQ@gmx.de

# 8 Fehlerbehebung

Hier möchte ich noch Tipps zur Behebung von Problemen geben die mir selbst untergekommen sind.

## C64 bleibt nach Load“\$“,8 hängen

... und zeigt keine Reaktion mehr, (wenn als Laufwerk 8 adressiert):  
74HCT244 falsch herum eingesetzt. Kontrollieren sie nochmal die Position für pin1

## File not found

Wenn der Fehlerkanal ausgelesen werden kann und die Softwareversion richtig über den Bus ausgelesen wird liegt der Fehler eher zwischen SD-Karte und AT-Mega.

Zb. funktioniert der „card detect“ nicht richtig oder es fehlt die Verbindung zum Mikroprozessor. Die Pins für „write protect“ und „card detect“ sind die ersten zwei Pins am SD-Sockel (3M), diese liegen eng beieinander.

Für die Kontrolle auf richtige Funktion gibt es einen kleinen Debugg-Modus, um diesen zu aktivieren muss man den RESERVE Taster beim Einschalten gedrückt halten, danach sollte die rote Led dauerhaft



leuchten, wenn die SD-Karte eingesteckt ist. Entfernt man nun die SD-Karte sollte die rote Led erlöschen, zieht man nun die Karte noch weiter heraus beginnt die grüne zu Leuchten  
Die rote Led zeigt den „card detect“ Status.  
Die grüne Led zeigt den „write protect“ Status.

Sollte nun die rote Led nie leuchten liegt ein Fehler beim „card detect“, entweder liegt es an der Lötstelle oder am Sockel, da kann nachlöten nie schaden.

### Device not Present

Nach Load“\$“,8 (,9,10) kommt nur Device not Present error

- Resettaster resetiert richtig = SD2IEC led leuchtet einige mal, eventuell vorhandene 1541 resetiert kurz.
- Wahrscheinlicher Fehler einer der 3 IEC Pins kontaktiert nicht die DTV Platine
- Durchmessen TP1 ->IEC Port Pin3 (ATN)
- Durchmessen TP8 ->IEC Port Pin4 (CLK)
- Durchmessen TP9 ->IEC Port Pin5 (Data)
- Kein Durchgang → Stifte kontrollieren, neu ausrichten mit dem LötKolben.
- Etwas Zinn auf die DTV Platine, Widerstand etwas nachlöten.

### IECATA lässt sich nicht flashen

Read wrong bytes from AVR: 0 0

Problem with AVR

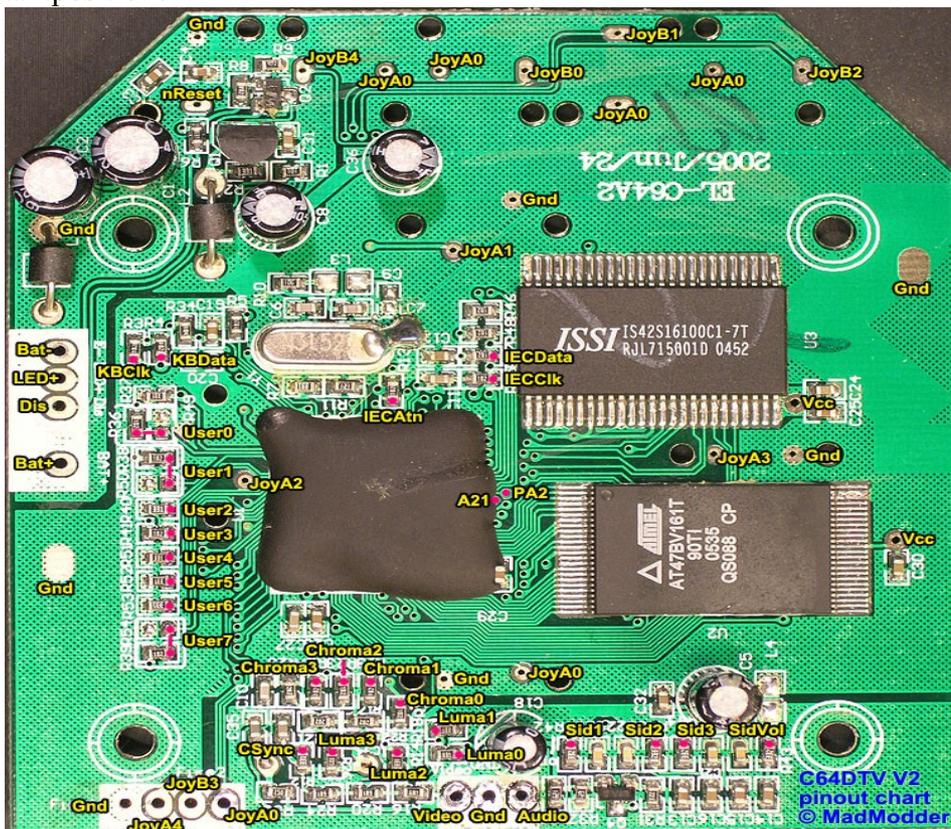
Switch power of AVR (.....) Try it again

- Der DTV hat keine SRQ Leitung, die ist aber mit Miso verbunden
- DTV entfernen (abschalten), C64 mit IEC Port verbinden und über den C64 flashen

# 9 Sonstiges...

Forschung und Entwicklung, oder einfach die Sachen die nirgends hinein gepasst haben.

DTV Kontaktpositionen



## 9.1 Stifte

Ich habe ca. 4 Arten von Stiften gekauft und getestet, und etliche weitere preislich angefragt. Die Stifte aus deutscher Produktion, bzw. von deutschen Firmen waren immer zu teuer, das ein Testen erst gar nicht in Frage kam. Sie lagen alle so bei 2 Euro je Stift, das hätte das Projekt bei ca. 35 Stiften schon preislich gesprengt, die Stifte aus China waren zwar meist günstig, allerdings gab es nie Datenblätter, und deshalb waren sie jedes Mal eine Überraschung. Leider war die DTV Platine zu dünn und hat sich zu sehr unter der Belastung gebogen. Natürlich könnte man mit einer Aluminiumplatte und Isoliermaterial die Platine abstützen, doch war mir der Aufwand mittlerweile schon zu groß.

## 9.2 Energieverbrauch

Sd2iec + SN7406	Max. 108mA (laden v. SD-Karte)
Sd2iec + SN74LS06	Max. 92mA (laden v. SD-Karte)
Sd2iec + IECATA + SN7406+SN7407 +ST1 Microdrive 5GB	540 mA (Start) 520mA (Leerlauf) max. 870mA (laden v. Microdrive)
Sd2iec + IECATA + sn74LS06 + SN74LS07 +ST1 Microdrive 5GB	470mA (Leerlauf) 770mA (laden v. Microdrive)
Sd2iec + IECATA +SN74LS06 + SN74LS07 + CF 4GB (Kingston)	~ 130mA - 260mA

Erlaubter maximaler Strom an der 5V Leitung am Kassettenport des C64: 300mA (500mA)

# 10 BASIC

## 10.1 Datei für den Diskettenwechschler am sd2iec

Der DSK-Switch Schalter dient zum Durchschalten der Diskimages zB für Programme die über mehrere Diskettenseiten gehen, oder wenn auf einer Extradiskette gespeichert wird. Sind nicht viele Programme, aber dann und wann ist das durchaus nützlich.

Damit die Knöpfe eine sinnvolle Funktion haben muss in das Aktuelle Verzeichnis (in der Regel das Rootverzeichnis der SD-Karte) eine Datei namens "Autoswap.lst" in der dann die Pfad/Dateinamen der durchzuwechselnden Images stehen.

Mehrseitige Disk-Images packt man idealerweise in ein eigenes Verzeichnis auf der SD-Karte und editiert eine Textdatei "autoswap.lst" in der die Diskseiten in der richtigen Reihenfolge eingetragen werden [disk1.d64, disk2.d64 usw... disk ist gegen den Namen der d64 auszutauschen], der Diskchanger arbeitet die Liste nur ab.

Ein kleines C64 Basic Prog. für die Erstellung einer Diskettenliste für das sd2iec.

```
1 open1,8,3,"autoswap.lst"  
2 print#1,"disk1.d64"  
3 print#1,"disk2.d64"  
4 print#1,"disk3.d64"  
(....)  
5 print#1,"disk250.d64"  
6 close1
```

# 11 Literaturverzeichnis

Hier noch eine Link liste zu einigen nützlichen Quellen:

<http://www.dienstagstreff.de/>

## Abbildungsverzeichnis

Anschluß und Jumper Übersicht.....	11
SMD Widerstände und Kondensatoren.....	13
SMD Bauteilpositionen auf der Unterseite.....	14
Kontaktstifte.....	15
Bauteile Oberseite.....	16
IC Sockel Oberseite.....	17
Stiflleisten und Ports verlöten.....	18
Spiff Lumafix DTV Platine .....	22
Lumafix Widerstände und Pinpositionen .....	23
Egretz 15Mx81 Gehäuse für C64.....	27