

## Doku zum STM32F429-Board von UB

**Inhaltsverzeichnis**

Einleitung.....	3
Bilder.....	4
Hardware Layout : .....	5
1. Modul-Beschreibungen.....	6
1.1. CPU.....	6
1.2. OSC-HSE.....	6
1.3. RESET.....	6
1.4. BOOT.....	7
1.5. VCAP, PDR, BYPASS.....	7
1.6. UBAT, Backup-Batterie.....	8
1.7. Block-C.....	8
1.8. Spannungs-Regler.....	9
1.9. Analog-Spannung.....	9
1.10. Power-Pinhead.....	10
1.11. SWD/JTAG-Pinhead.....	10
1.12. GND-Pins.....	11
1.13. Testpads.....	11
1.14. Mechanik.....	11
1.15. SDRAM.....	12
1.16. USB-OTG.....	13
1.17. I2C-EEProm.....	14
1.18. I2C-Pinhead.....	14
1.19. SPI-Flash.....	15
1.20. SPI-Pinhead.....	15
1.21. CAN.....	16
1.22. RS232-UART.....	17
1.23. TFT-Pinhead.....	18
1.24. GPIO-Pinhead (Digital).....	19
1.25. GPIO-Pinhead (Analog).....	19
1.26. DAC-Pinhead.....	19
1.27. SD-CARD.....	20
1.28. Status LEDs.....	20
1.29. UART-Pinhead.....	21
1.30. RTC-Quarz.....	21
2. Aufbau und Inbetriebnahme.....	22
2.1. Tipps und Hinweise.....	22
2.2. Minimal Bestückung.....	23
2.3. Adapterkabel für ST-Link/V2.....	24
2.3. Erster Funktionstest.....	24
2.4. Buttons.....	26
2.5. Spannungs-Regler.....	27
2.6. Status LEDs.....	28
2.7. SDRAM.....	29
2.8. USB-OTG.....	30

2.9. I2C.....	31
2.10. SPI.....	32
2.11. CAN.....	33
2.12. RS232-UART.....	34
2.13. SD-CARD.....	35
2.14. Sonstige Header.....	36
2.15. RTC und Backup-Batterie.....	37
3. Anhang.....	38

## Einleitung

Diese Doku bezieht sich auf das STM32F429-Board in der Version 1.0 von UB.

Das F429-Board ist ein Eigenbau Entwicklungsboard mit einer 32bit STM32F429 CPU.

Die Platine ist so ausgelegt, das sie möglichst flexibel in der Anwendung und möglichst klein in der Abmessung ist. Der User kann sich je nach Einsatzzweck die notwendigen Module auf der Platine bestücken und die nicht notwendigen weglassen.

Im Schaltplan und in der Doku sind aus diesem Grund die einzelnen Module in logische Gruppen unterteilt um einen einfacheren Überblick zu haben welche Bauteile zu welcher Funktion gehören.

Es gibt eine Anzahl von Bauteilen die zum Betrieb vom F429-Board zwingen notwendig sind, diese befinden sich alle auf der Seite-1 vom Schaltplan. Zur Inbetriebnahme wird empfohlen zuerst nur diese Bauteile zu bestücken und die Platine dann zu testen. Beschrieben ist das ganze unter „2.2. Minimal Bestückung“

Diese Doku ist zweigeteilt. Der erste Teil zeigt grob alle Module und welche Bauteil dazugehören. Der zweite Teil dreht sich um den Aufbau und die Inbetriebnahme. Bitte zuerst alles komplett durchlesen bevor mit dem Aufbau begonnen wird.

Quelle : [www.Mikrocontroller-4U.de](http://www.Mikrocontroller-4U.de)      [www.Mikrocontroller.bplaced.net](http://www.Mikrocontroller.bplaced.net)  
EMail : [MC-4U@T-Online.de](mailto:MC-4U@T-Online.de)

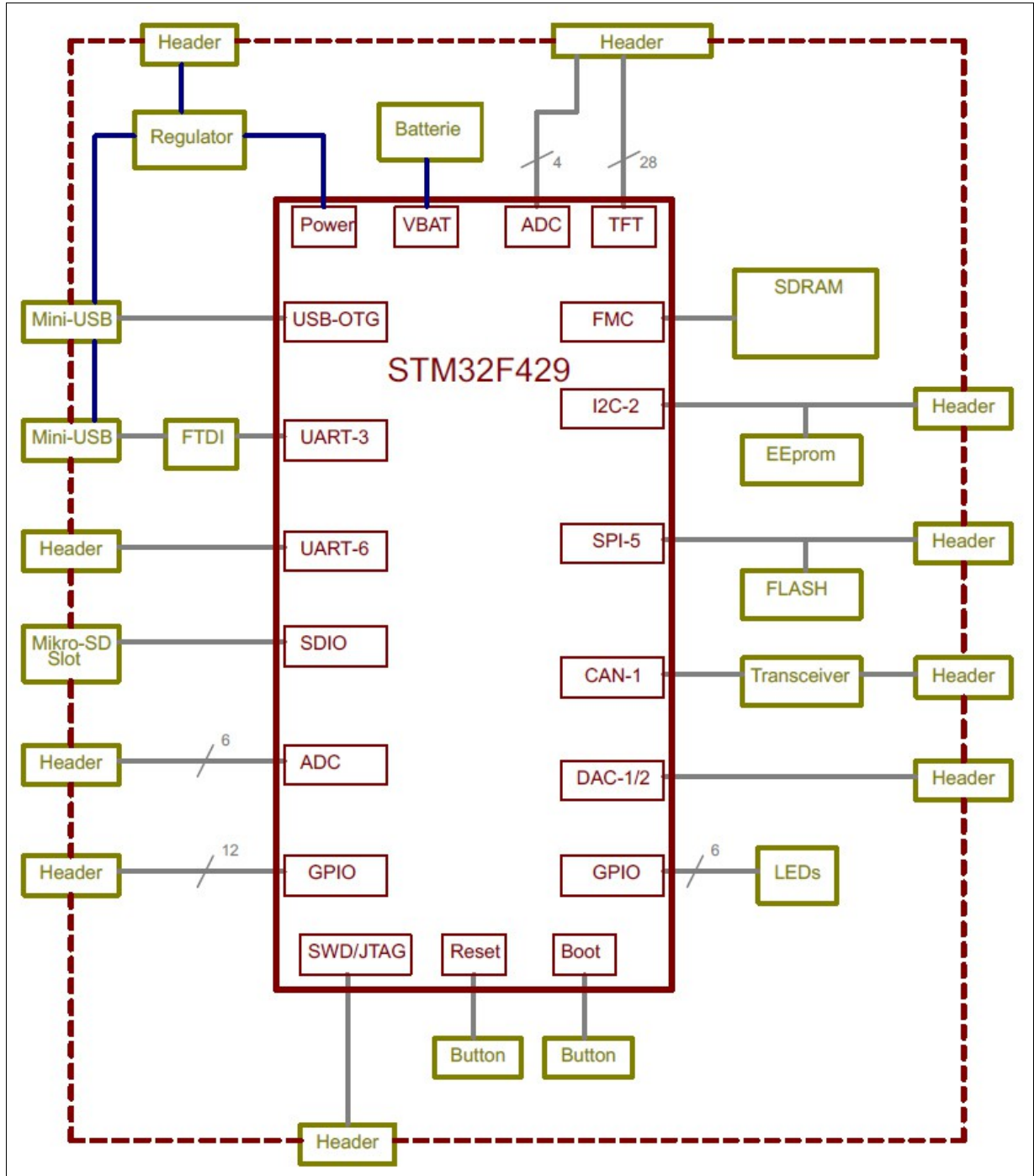
**Alle Dokumente, Software und Hardware von mir sind nicht frei von Fehlern und werden auf eigene Gefahr benutzt. Eine Gewährleistung für eventuell auftretende Fehler kann nicht übernommen werden. Wer Fehler findet, kann diese gerne per Email an mich senden, damit ich diese beheben oder zumindest dokumentieren kann.**

Jetzt viel Spaß mit dem der Platine

Uwe



### Hardware Layout :



## 1. Modul-Beschreibungen

### 1.1. CPU

Als CPU (IC1) muss ein STM32F429IGT6 bestückt werden.

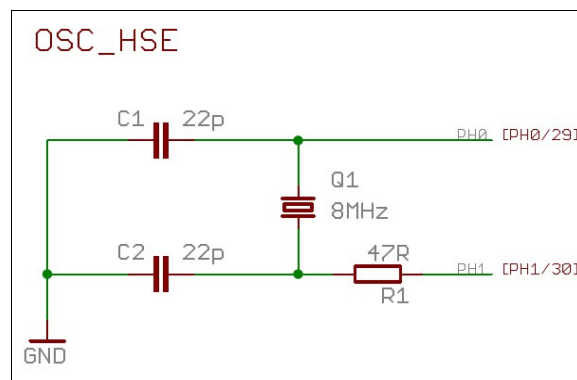
Bauform : 176pin LQFP

Flash : 1 MByte

RAM : 256 Kbyte

Core : 32bit Cortex-M4 (max 180 MHz)

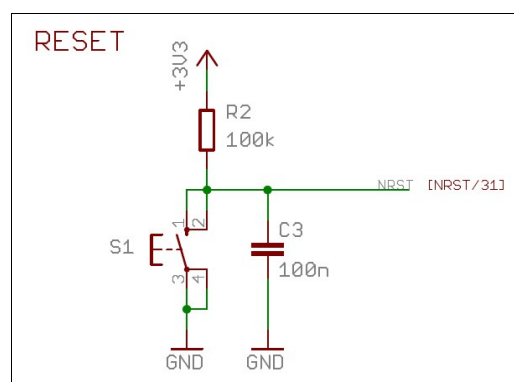
### 1.2. OSC-HSE



Falls die CPU per externem Quarz getaktet werden soll, muss Q1, C1, C2, R1 bestückt werden.

Die CPU kann auch per internem Oszillator betrieben werden, dann sind diese Bauteile nicht notwendig.

### 1.3. RESET

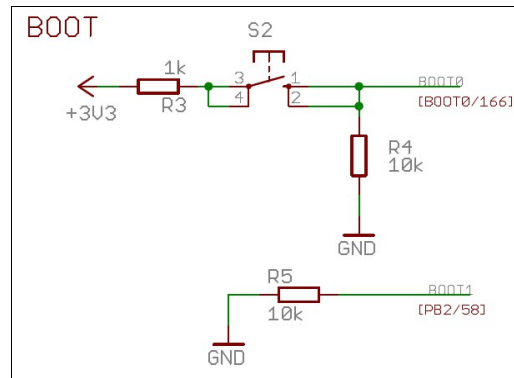


R2, C3 sind zwingend notwendig.

Um manuell einen Reset auslösen zu können, muss der Button S1 bestückt sein.

Falls ein externer Reset-Button benötigt wird, kann dieser parallel zu S1 angeschlossen werden.

## 1.4. BOOT



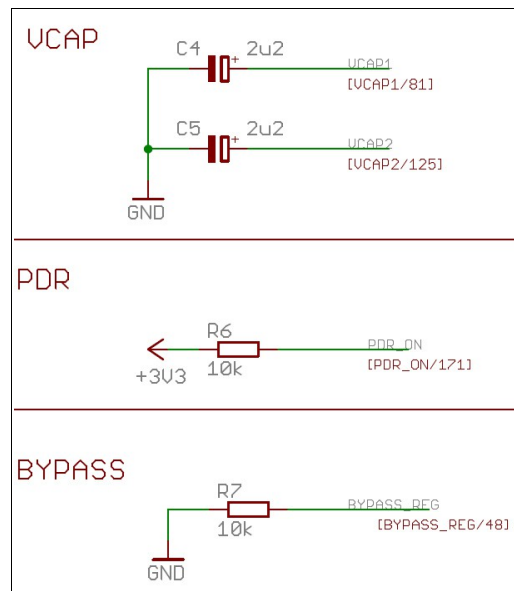
R4, R5 sind zwingend notwendig.

Um den internen Bootloader der CPU starten zu können, müssen S2,R3 bestückt sein.

Der Bootloader wird dann mit dieser Sequenz gestartet :

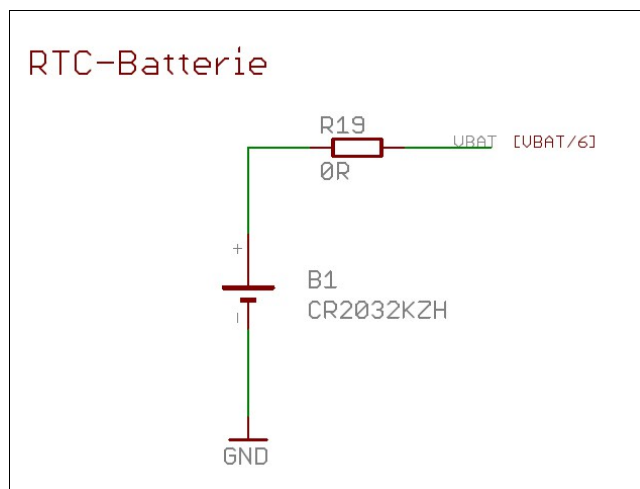
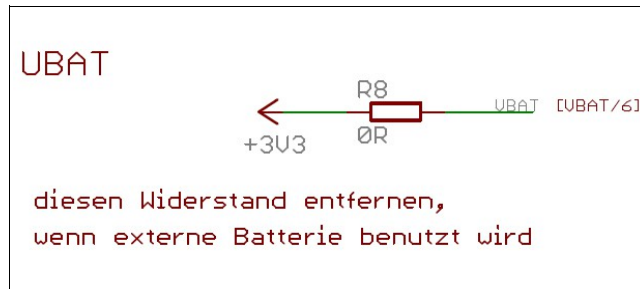
1. Power vom Board einschalten
2. Boot-Button (S2) drücken und festhalten
3. Reset-Button (S1) kurz drücken und wieder loslassen
4. Boot-Button (S2) loslassen.
5. Der Bootloader ist jetzt gestartet und die CPU wartet auf Software per RS232 (X2) oder USB (X1) oder CAN (J8)

## 1.5. VCAP, PDR, BYPASS



Die Bauteile C4, C5, R6, R7 müssen zwingend bestückt werden.

## 1.6. UBAT, Backup-Batterie



Falls keine externe Batterie benutzt werden soll, kann der UBAT-Pin mit dem 0R Widerstand R8 an VCC angeschlossen werden.

Falls eine externe Batterie benutzt werden soll (B1,R19) darf der Widerstand R8 nicht bestückt sein !

Statt den 0R Widerständen R8,R19 können auch zwei Dioden bestückt werden. Damit UBAT entweder von extern oder per B1 versorgt wird.

Hinweis : Der Batterie-Halter (B1) verdeckt einige Bauteile, diese müssen zuerst eingelötet werden !!

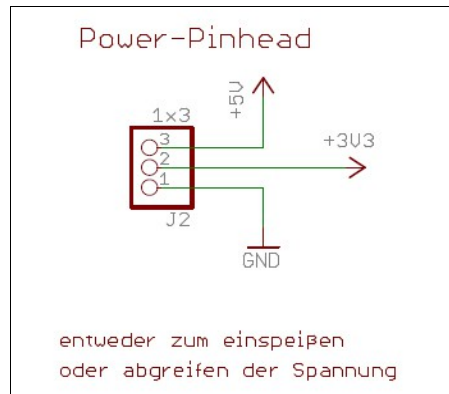
## 1.7. Block-C

Alle Kondensatoren mit „CB“ im Namen müssen bestückt werden.



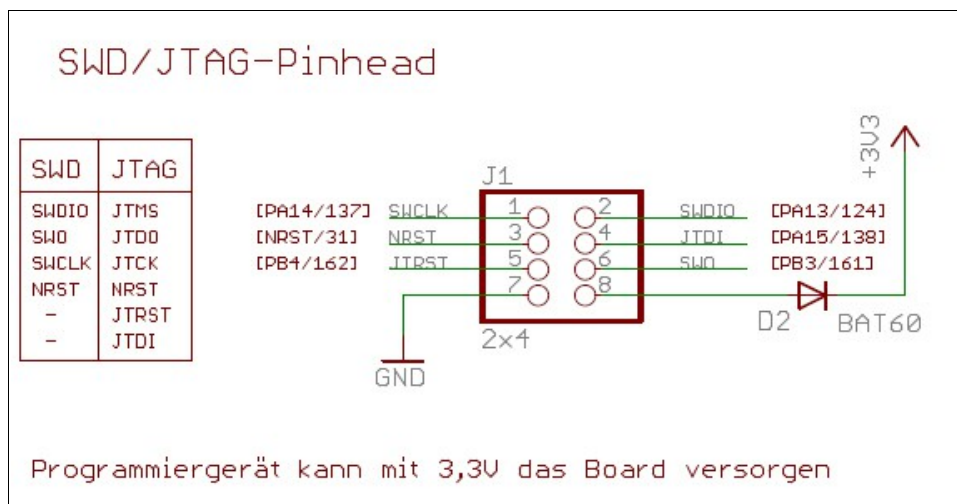


### 1.10. Power-Pinhead



An diesem Pinhead kann entweder das Board mit Spannung versorgt werden (entweder mit 3V oder mit 5V) oder der Stecker kann benutzt werden um 3V oder 5V abzugreifen (z.B. wenn das Board per USB-Buchse versorgt wird)

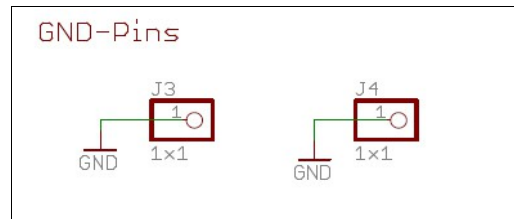
### 1.11. SWD/JTAG-Pinhead



Über diesen Pinhead kann die CPU entweder per SWD oder per JTAG programmiert werden.  
 Für SWD sind 5 Leitungen notwendig (Pin 1+2+3+6+GND)  
 Für JTAG sind 7 Leitungen notwendig (Pin 1+2+3+4+5+6+GND)  
 Der Pin-8 kann genutzt werden um das Board mit 3,3V zu versorgen (\*1)  
 (dann muss die Diode D2 so eingelötet sein wie gezeichnet)  
 Oder falls das Programmiergerät über Pin-8 vom Board versorgt werden soll, muss die Diode umgepolt eingelötet werden.

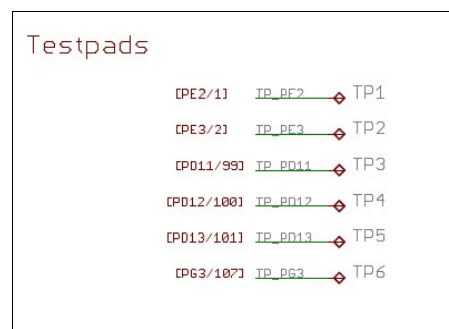
(\*1) Falls ein ST-Link/V2 von einem Discovery-Board zum programmieren benutzt wird, kann dieses so umgebaut werden, das es das STM32F429-Board mit Spannung versorgt. (Dann benötigt man keine zweite Spannungsquelle oder zweites Kabel)

## 1.12. GND-Pins



Auf dem Board kann man zwei GND-Pins bestücken um z.B. die Masse vom Oszi anzuklemmen.

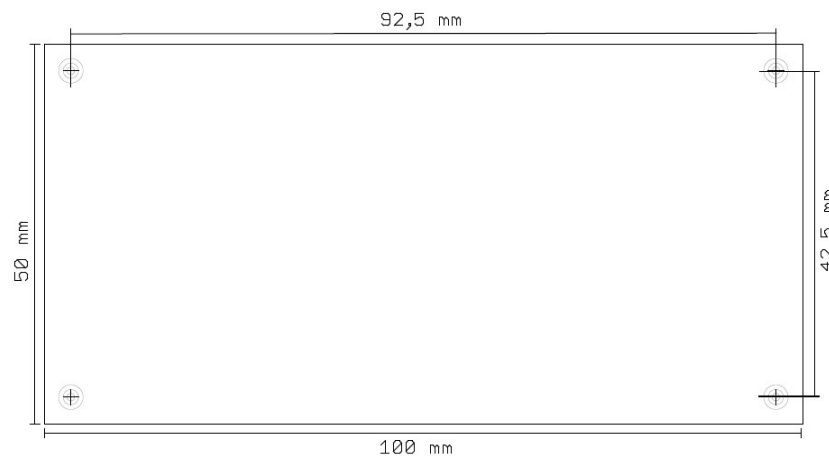
## 1.13. Testpads



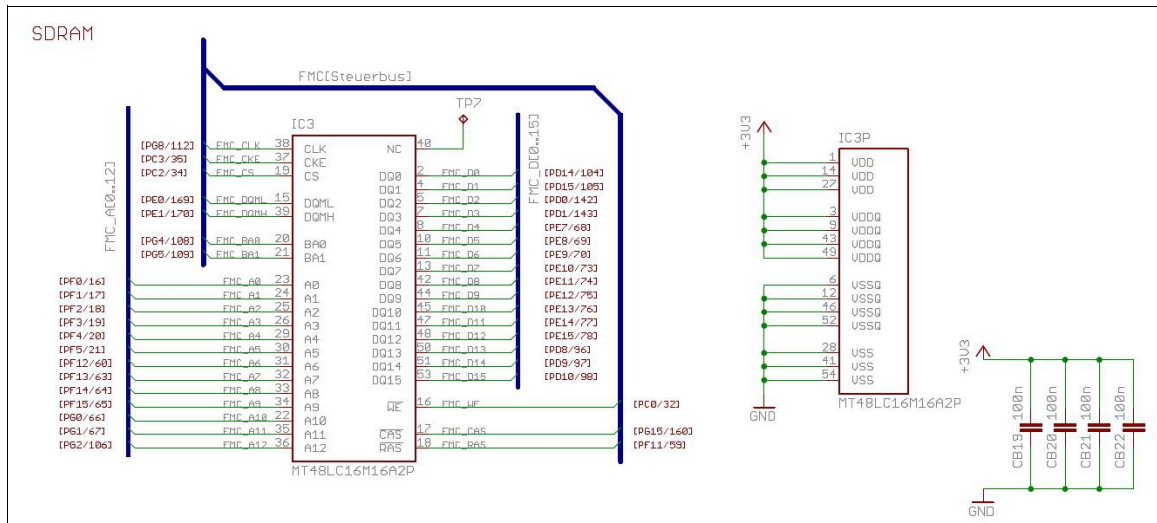
Ein paar GPIO-Pins der CPU liegen auf Testpads auf der TOP-Seite vom Board. Diese können entweder zum debuggen benutzt werden (Testpins für ein Oszi) oder es können dort „fliegend“ Leitungen angelötet werden.

## 1.14. Mechanik

Die Platine hat die Abmessungen von 100mm x 50mm.  
Die Befestigungsbohrungen haben einen Durchmesser von 3,2mm  
und einen Abstand von 92,5mm x 42,5mm (jeweils von der Lochmitte)



### 1.15. SDRAM



IC3 kann mit einem SDRAM bestückt werden.  
 (alle anderen Bauteile die hier gezeichnet sind, sind dann auch notwendig)

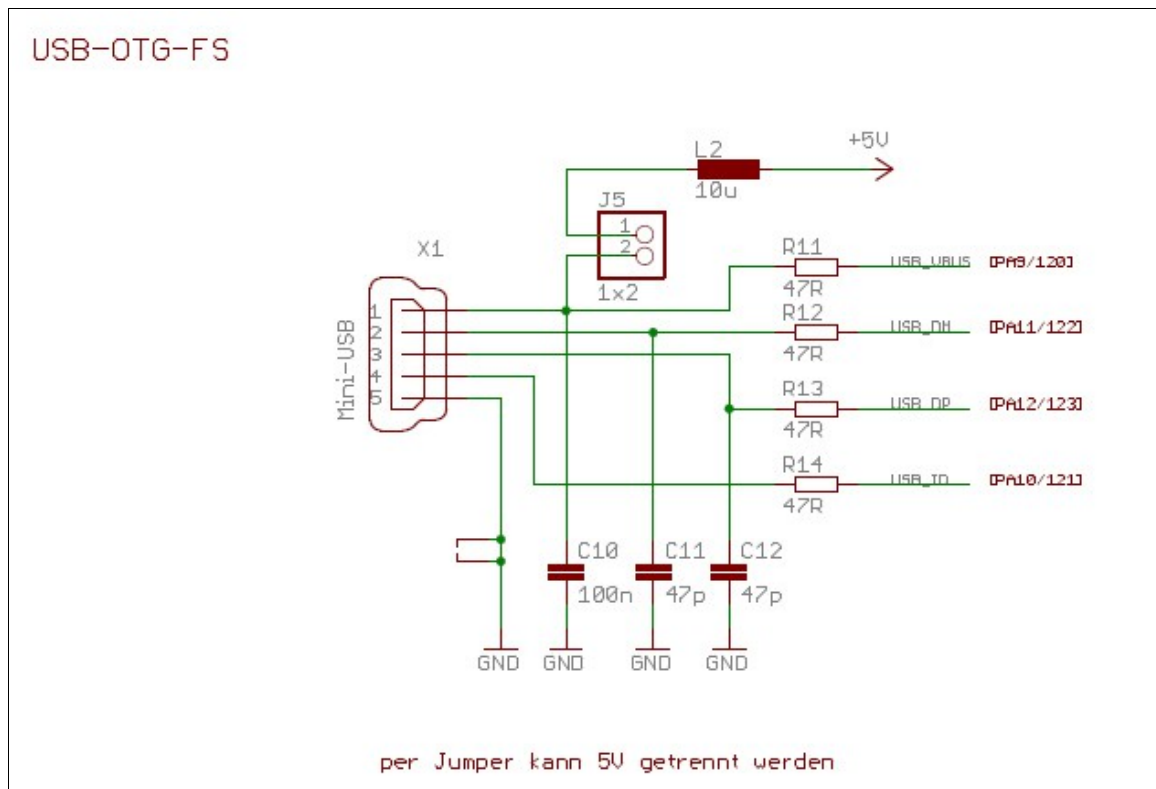
Das Gehäuse muss 54pin TSOP-II sein.

Im Schaltplan ist ein 256MBit Typ gezeichnet, es kann aber auch ein kleinerer Typ eingesetzt werden solange das Pinout passt und er einen 16bit Datenbus besitzt.

Für größere RAMs (mit Adressleitung A13) kann der Pin 40 (A13) über das Testpad TP7, theoretisch mit dem CPU-Pin PG3/107 verbunden werden (FMC\_A13).

Dieser liegt auf dem Testpad TP6.  
 (Bin mir aber nicht sicher ob das funktioniert)

## 1.16. USB-OTG



Falls USB-OTG benutzt werden soll, müssen die hier gezeigten Bauteile bestückt werden.

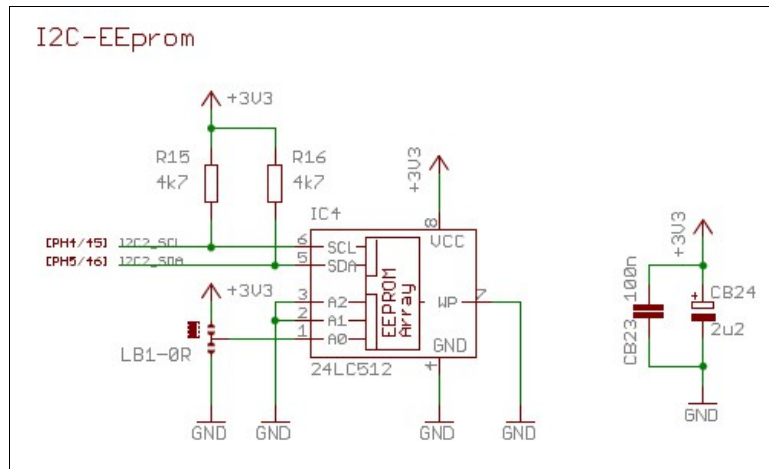
Über die USB-Buchse X1 kann das Board entweder mit einem PC verbunden werden (die CPU muss dann als USB-Device programmiert werden)

Oder es kann ein USB-Device an X1 angeschlossen werden (die CPU muss dann als USB-Host programmiert werden)

Je nach Anwendungsfall können die 5V über den Jumper J5 getrennt/verbunden werden.

Die CPU kann mit dem Bootloader über diese Buchse programmiert werden.

### 1.17. I2C-EEProm



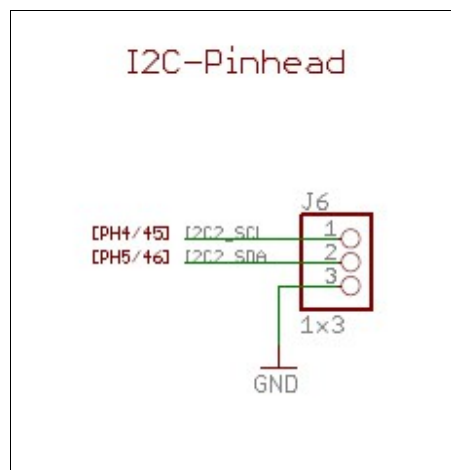
IC4 kann mit einem I2C-EEProm bestückt werden.  
 (alle anderen Bauteile die hier gezeichnet sind, sind dann auch notwendig)

Das Gehäuse muss ein SOIC-8 sein.

Im Schaltplan ist ein 512kBit Typ gezeichnet, es kann aber auch ein anderer Typ eingesetzt werden solange das Pinout passt.

Über die Lötbrücke LB1 kann eine von zwei I2C-Adressen eingestellt werden.

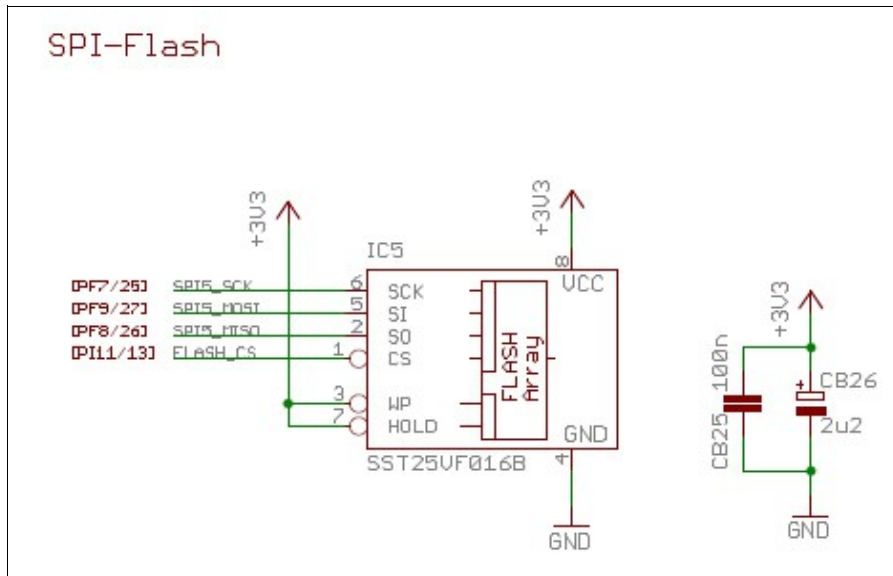
### 1.18. I2C-Pinhead



An diesem Pinhead können externe Bauteile an den I2C-Bus angeschlossen werden.

Die notwendigen Pull-Up Widerstände (R15 und R16) dürfen nur einmal am Bus vorhanden sein. Falls es also schon externe Pull-Ups gibt, müssen R15, R16 entfernt werden.

### 1.19. SPI-Flash

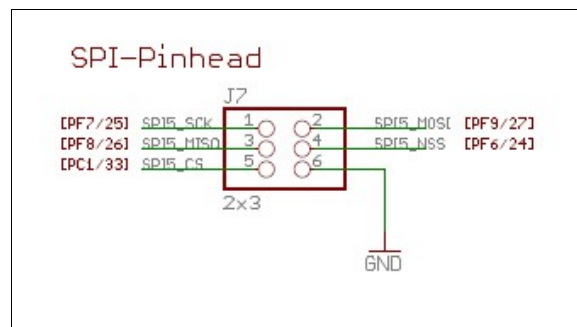


IC5 kann mit einem SPI-Flash bestückt werden.  
 (alle anderen Bauteile die hier gezeichnet sind, sind dann auch notwendig)

Das Gehäuse muss ein SOIC-8 sein.

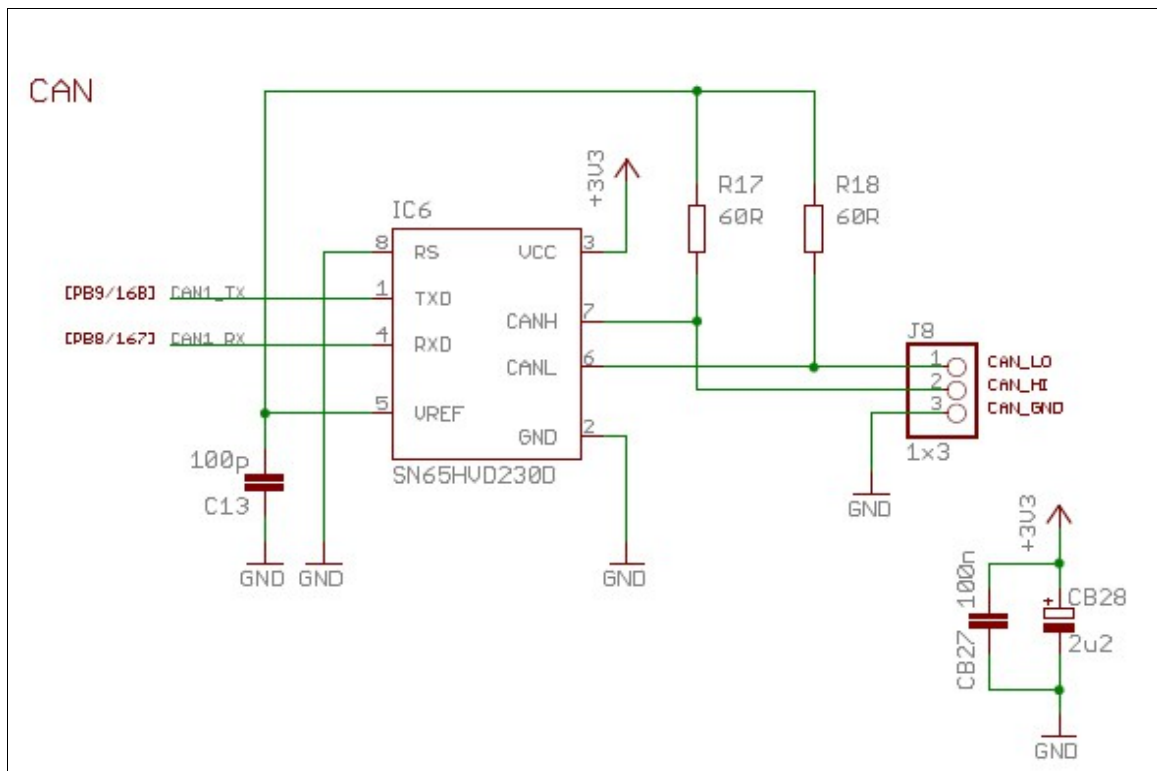
Im Schaltplan ist ein 16MBit Typ gezeichnet, es kann aber auch ein anderer Typ eingesetzt werden solange das Pinout passt.

### 1.20. SPI-Pinhead



An diesem Pinhead können externe Bauteile an den SPI-Bus angeschlossen werden.

## 1.21. CAN



IC6 kann mit einem CAN-Transceiver bestückt werden.  
(alle anderen Bauteile die hier gezeichnet sind, sind dann auch notwendig)

Das Gehäuse muss ein SO-8 (narrow) sein.

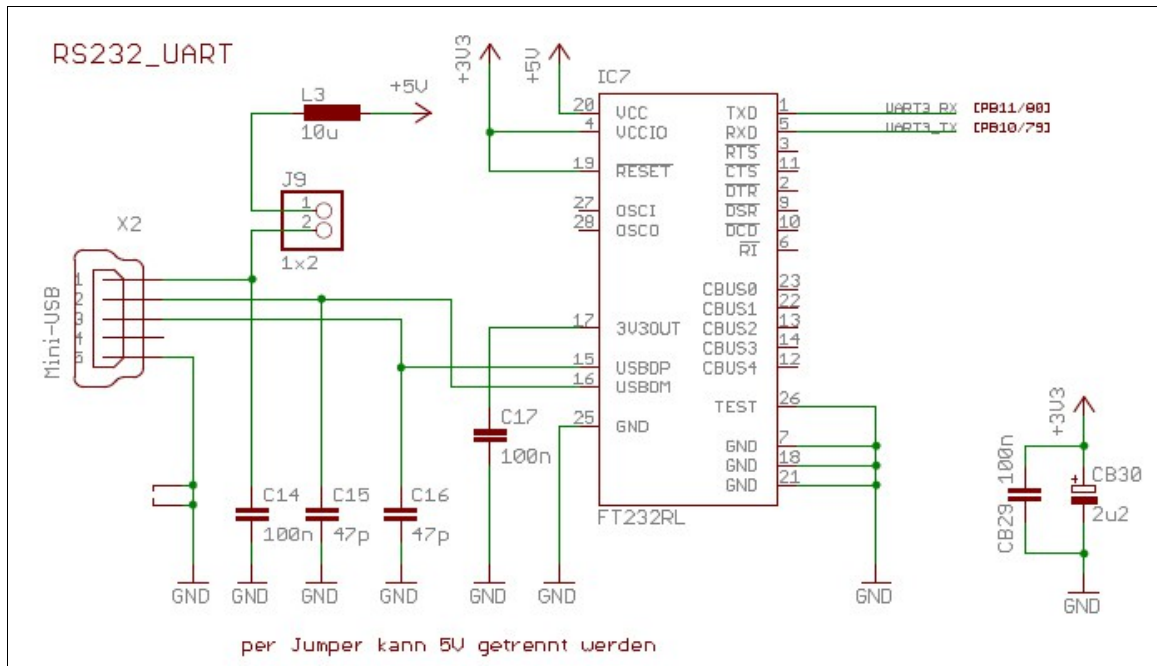
Über den 3pol Pinhead (J8) können externe Bauteile an den CAN-Bus angeschlossen werden.

Hinweis : den externen 120 Ohm Terminierungs-Widerstand am BUS nicht vergessen !

Die CPU kann mit dem Bootloader über diesen Pinhead programmiert werden.



### 1.22. RS232-UART



IC7 kann mit einem UART-FTDI-Chip bestückt werden.  
 (alle anderen Bauteile die hier gezeichnet sind, sind dann auch notwendig)

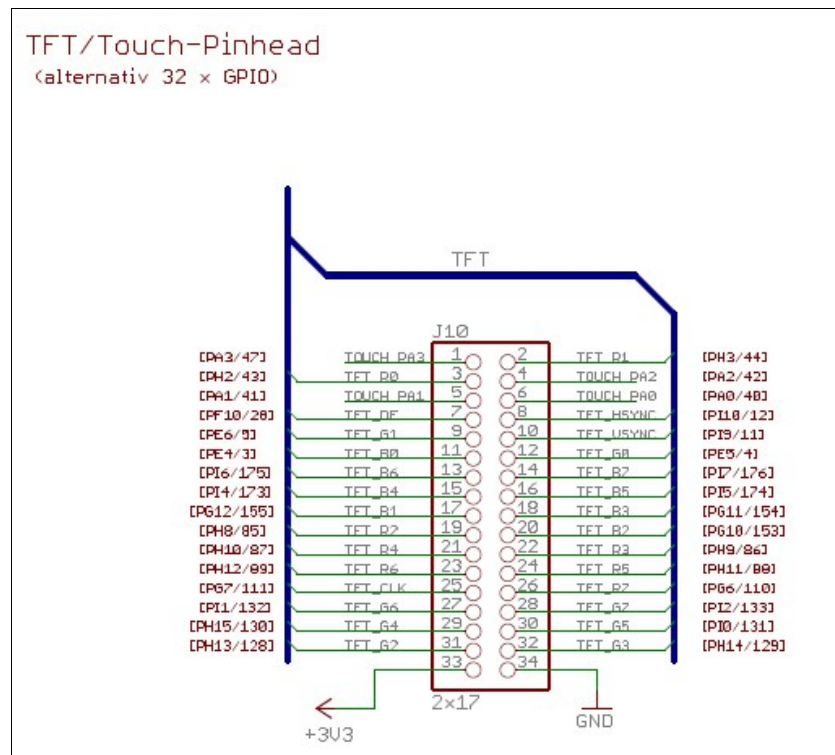
Das Gehäuse muss 28pin SSOP sein.

Über die USB-Buchse (X2) kann das Board mit einem PC verbunden werden  
 und die CPU kann dann per UART Daten austauschen. (virtueller COM-Port)

Je nach Anwendungsfall können die 5V über den Jumper J9 getrennt/verbunden werden.

Die CPU kann mit dem Bootloader über diese Buchse programmiert werden.

### 1.23. TFT-Pinhead



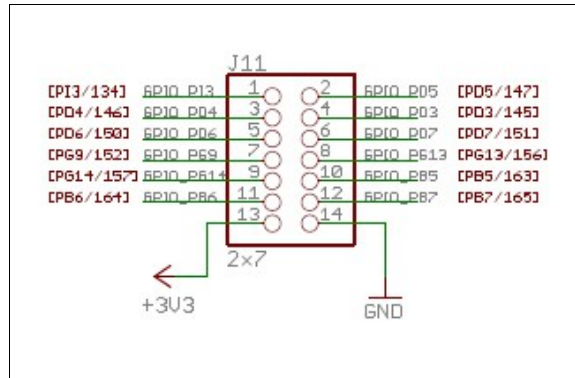
Über diesen Pinhead kann entweder ein TFT-Display (mit Touch) angeschlossen werden. Oder man kann den Pinhead benutzen um an 32 GPIO-Pins der CPU zu kommen.

Das Pinout vom Header ist willkürlich und passt zu keinem Display direkt.

Das bedeutet es muss immer ein Adapterkabel oder eine Adapterplatine zum Anschluss eines Displays benutzt werden.

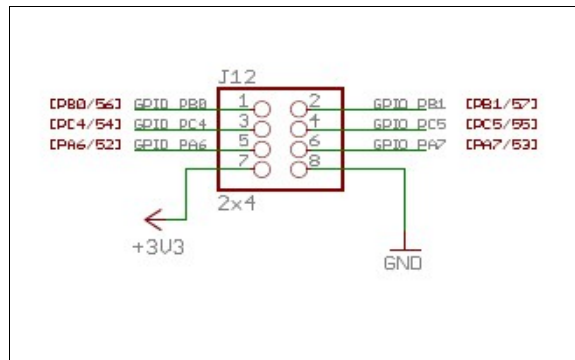
Hinweis : Ich werde wahrscheinlich noch eine Adapterplatine herstellen lassen mit einer "Standard Belegung" für einen 40pol. FPC-Stecker um ein 4,3Zoll oder 7Zoll Display mit Touch anschließen zu können. (Ich werde in meinem Web-Blog darüber schreiben)

### 1.24. GPIO-Pinhead (Digital)



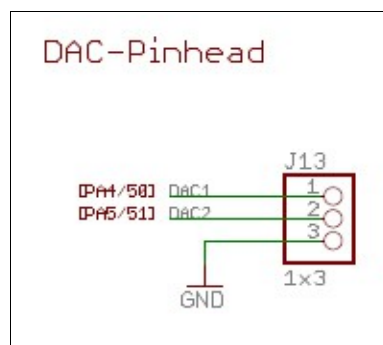
An diesem Pinhead liegen 12 Digitale GPIOs der CPU.

### 1.25. GPIO-Pinhead (Analog)



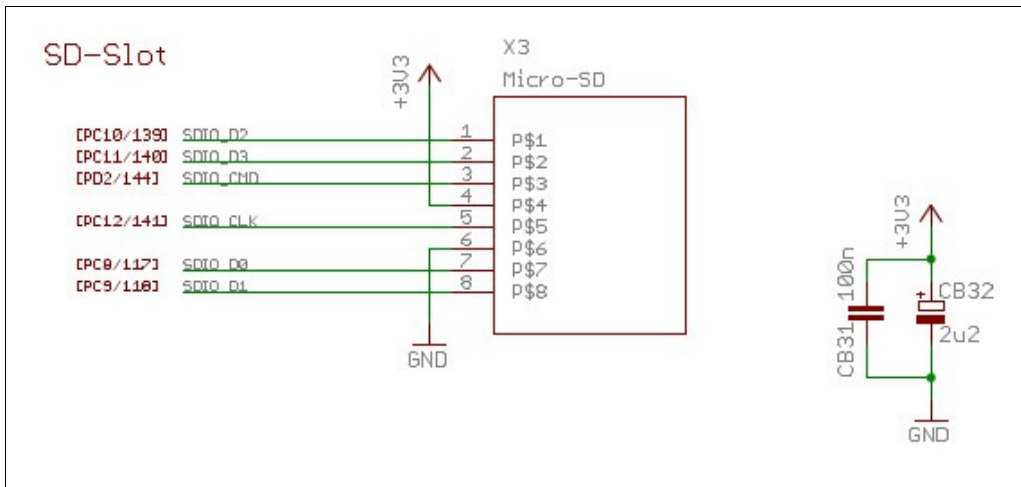
An diesem Pinhead liegen 6 Analoge (oder Digitale) GPIOs der CPU.

### 1.26. DAC-Pinhead



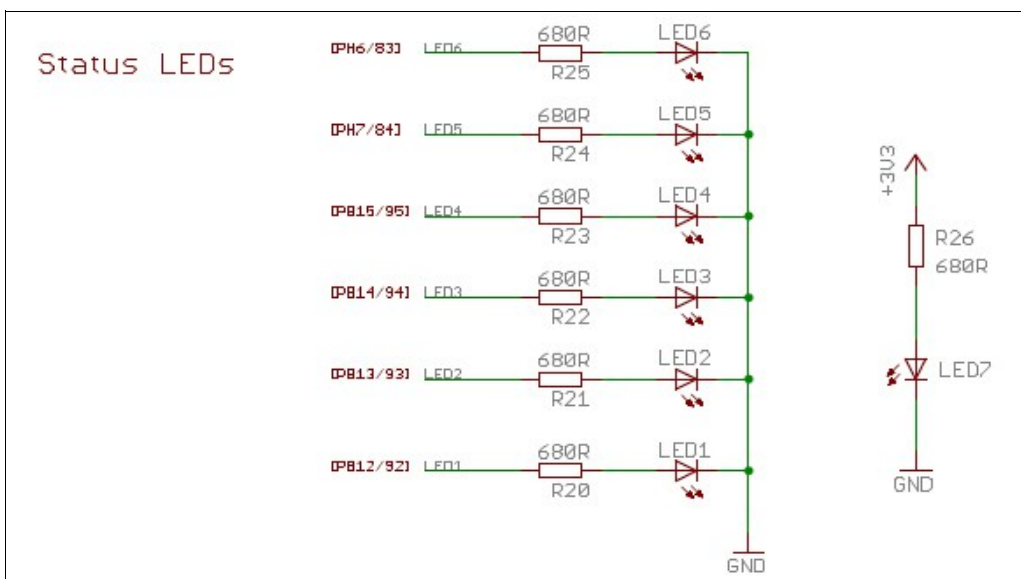
An diesem Pinhead liegen die beiden DAC-Kanäle der CPU.

### 1.27. SD-CARD



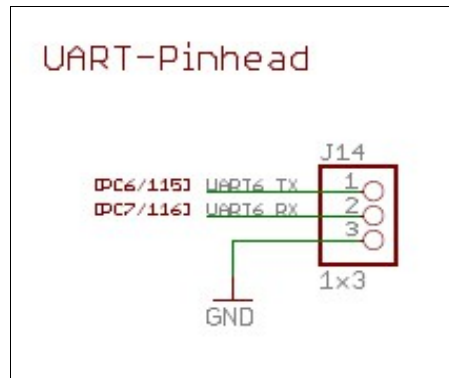
An X3 kann ein 8pol Kartenslot für Mikro-SD Karten bestückt werden.

### 1.28. Status LEDs



Es können bis zu 7 Status LEDs bestückt werden.  
 LED1 bis LED6 können über die CPU gesteuert werden.  
 LED7 dient als Power-LED

### 1.29. UART-Pinhead



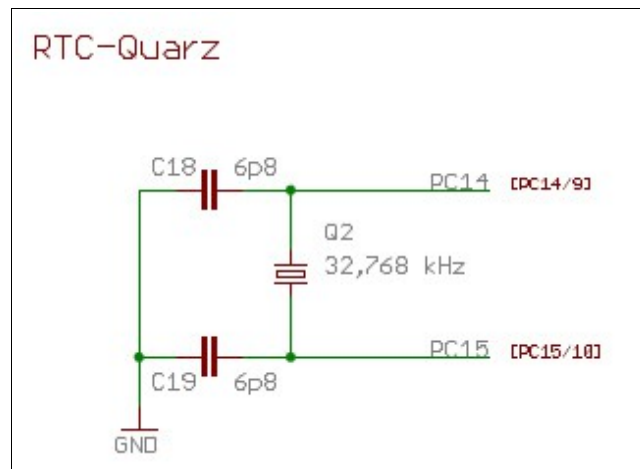
An diesem Pinhead liegt eine 3,3V UART der CPU.

Pin1 = TX der CPU (muss mit RX vom PC verbunden werden)

Pin2 = RX der CPU (muss mit TX vom PC verbunden werden)

Dieser Header dient als Debug-Schnittstelle bei der Inbetriebnahme vom Board.  
Jede Testsoftware von mir gibt über diesen Stecker Status-Meldungen aus.

### 1.30. RTC-Quarz



Falls eine RTC realisiert werden soll, können Q2, C18, C19 bestückt werden.

## 2. Aufbau und Inbetriebnahme

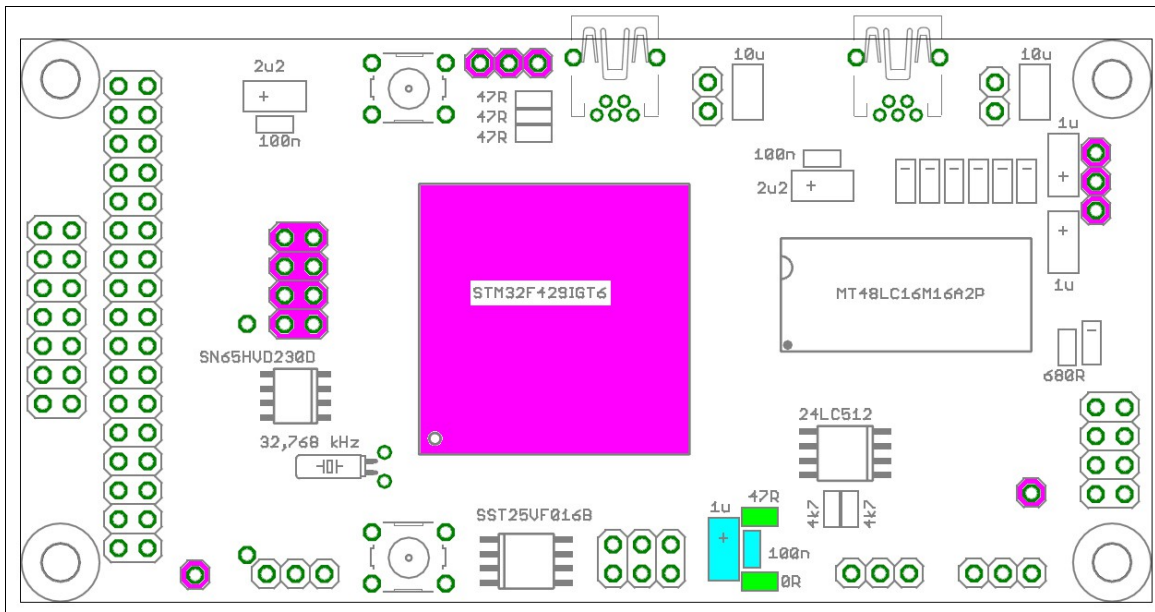
### 2.1. Tipps und Hinweise

- A.) Bitte zuerst das Board nur mit der "Minimalbestückung" aufbauen und dann testen ob es keine Fehler gibt. Dafür habe ich Testsoftware geschrieben.
- TST\_F429\_PowerOn : prüft ob die CPU läuft und die UART funktioniert.
  - TST\_F429\_PinCheck : prüft alle GPIOs auf Kurzschlüsse.
  - TST\_F429\_Terminal : dient (optional) zum prüfen mit dem Oszi auf Unterbrechungen.
- B.) Die Bauteile vom Board immer Modulweise bestücken und dann das Modul sofort testen. Ich habe für jedes Modul eine Testsoftware geschrieben die per UART den Status ausgibt. Damit grenzt sich die event. Fehlersuche auf ein einzelnes Modul ein.
- C.) Als erstes die CPU bestücken. Die noch leere Platine liegt dann noch waagrecht auf dem Tisch und man hat auf der Oberseite Platz zum löten.
- D.) Bei den anderen Bauteilen immer zuerst mit den kleinsten beginnen. Also zuerst die Widerstände (0603) und die Kondensatoren (0603) einlöten. Danach die größeren Bauteile.
- Sonst kann es passieren das man an die kleinen mit dem LötKolben nicht richtig drankommt, wenn ein größeres Bauteil direkt daneben sitzt.
- E.) Viel mit Flussmittel arbeiten. Das vereinfacht das löten und verhindert Kurzschlüsse.
- F.) Das Bauteil mit einer Pinzette fassen. Auf die Pads halten und mit dem LötKolben auf einer Seite fixieren. Die Verzinnung vom Pad reicht aus um das zu erledigen.
- Wenn das Bauteil fixiert ist, die andere Seite mit wenig Lötzinn anlöten, kurz abkühlen lassen und dann die fixierte Seite mit wenig Lötzinn nachlöten.
- G.) Die ICs (speziell die CPU) lötet man am einfachsten (finde ich) mit einer Lötdepotspitze und viel Flussmittel.
- H.) In der Anleitung sind alle Widerstände „grün“ markiert.  
Alle Kondensatoren sind „türkis“ markiert.  
Alle restlichen Bauteile sind „pink“ markiert

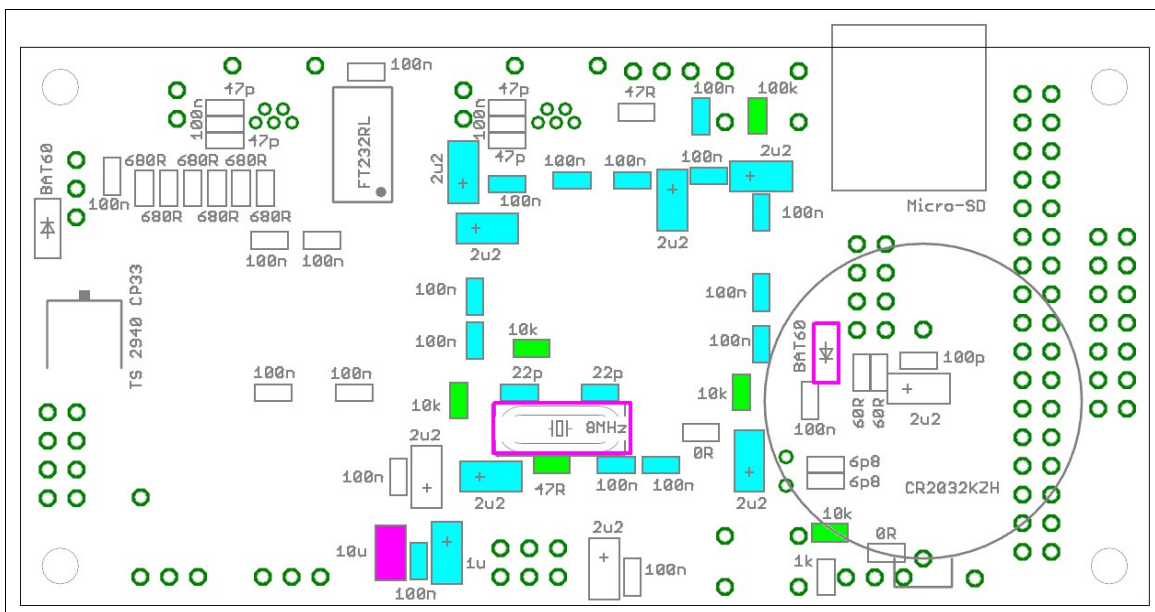
## 2.2. Minimal Bestückung

Zuerst sollten nur die notwendigsten Bauteile bestückt werden, um einen ersten Funktionstest durchführen zu können bei dem anderen Teile störend wirken könnten.

CPU :	IC1	Analog-Referenz :	R9, C8, C9
GND-Testpins :	J3, J4	Analog-GND :	R10
SWD/JTAG-Pinhead :	J1	Power + UART Pinhead :	J2, J14

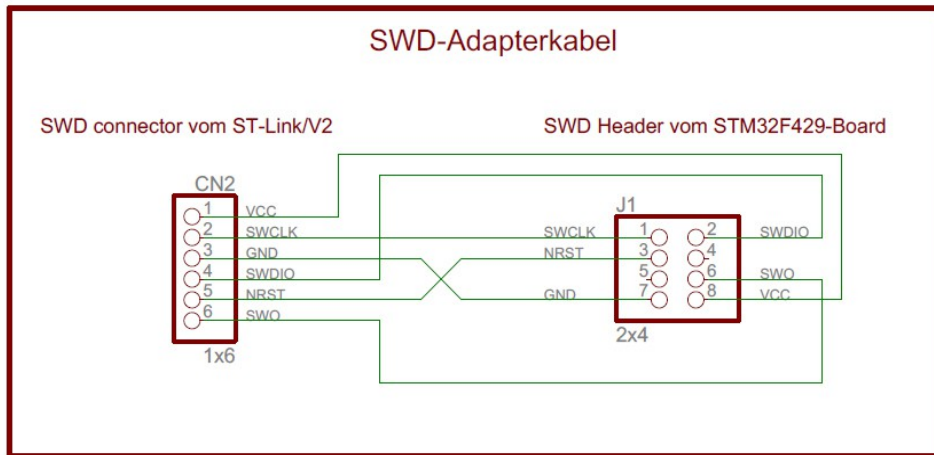


OSC_HSE :	C1, C2, Q1, R1	Analog-VDD :	L1, C6, C7
RESET :	R2, C3	Block-C :	CB1 bis CB15
BOOT :	R4, R5	VCAP :	C4, C5
PDR,BYPASS :	R6, R7	Diode :	D2



## 2.3. Adapterkabel für ST-Link/V2

Um das STM32F429-Board über einen ST-Link/V2 per SWD programmieren/debuggen zu können, muss ein Adapterkabel angefertigt werden.



Hinweis : Falls ein STM-Discovery-Board zum programmieren benutzt wird, kann dieses umgebaut werden damit es über das SWD-Adapterkabel auch das STM32F429-Board mit Strom versorgt. (Der VCC-Pin vom CN2 muss dazu an 3,3V angeschlossen werden)

## 2.3. Erster Funktionstest

### 1. Check der Stromaufnahme mit einem Labornetzteil :

Über den Pinhead J2 das Board mit 3,3V versorgen und am Netzteil langsam den Strom erhöhen. Mit der minimal Bestückung sollten nicht mehr als ca. 12 mA fließen (gemessen mit noch nicht programmierter CPU).

### 2. Check der UART-Funktion per Software :

Über den Pinhead J1 kann jetzt die Testsoftware (TST\_F429\_PowerOn) eingespielt werden und mit einem Terminalprogramm am PC über den Pinhead J14 geprüft werden. (UART-Setting = 115200 Baud, 8N1)

Die Status-Meldung „F429-Board COM6 ok“ muss zyklisch angezeigt werden. Jeder String an die CPU wird per Echo zurückgegeben.

### 3. Check der GPIOs auf Kurzschlüsse per Software :

Über den Pinhead J1 die Testsoftware (TST\_F429\_PinCheck) einspielen. Der Status wird über den Pinhead J14 ausgegeben. (UART-Setting = 115200 Baud, 8N1)

Geprüft werden alle GPIOs auf Kurzschluss gegen GND, VCC oder untereinander. Alle Tests müssen mit „OK“ beendet werden.

**Hinweis : Diese Software wirklich nur mit der „Minimal Bestückung“ laufen lassen. Ansonsten kann es zu Fehlern oder Kurzschlüssen kommen.**



**4. Check aller GPIOs per Software und Oszi auf Unterbrechungen (optional) :**

Mit der PC-Software : Discovery-Terminal (ab Version 1.8)  
und der CPU Testsoftware (TST\_F429\_Terminal)

können alle zugänglichen (freien) GPIO-Pins auf Funktion bzw. Unterbrechungen geprüft werden.

Die Software kann jeden beliebigen Pin als Ausgang schalten und ein HI oder LO-Signal  
(oder ein Rechteck-Signal) ausgeben, per Oszi kann das Ergebnis überprüft werden.

Mit der Software können auch die ADC-Kanäle überprüft werden.

(per Poti eine Spannung von 0V bis 3V anlegen)

Hinweis : folgende Pins sollten nicht geprüft werden (weil sie benutzt bzw. belegt sind)

PA13, PA14, PA15	: belegt vom JTAG/SWD
PB2	: belegt von Pull-Down R5
PB3,PB4	: belegt von JTAG/SWD
PC6, PC7	: belegt von UART-6
PH0, PH1	: belegt von Quarz Q1

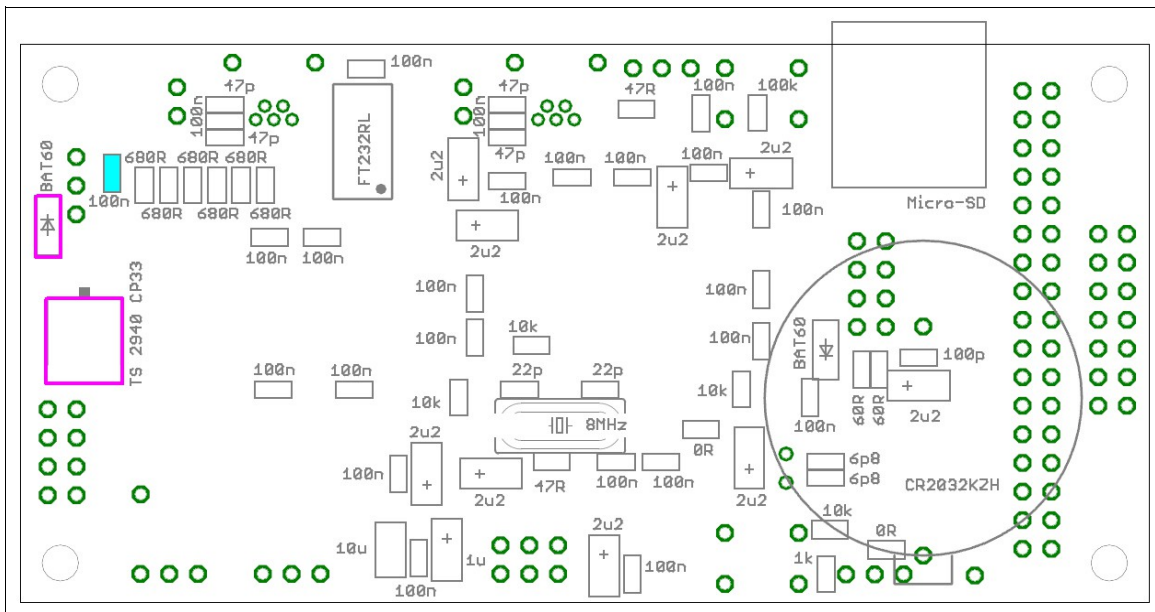
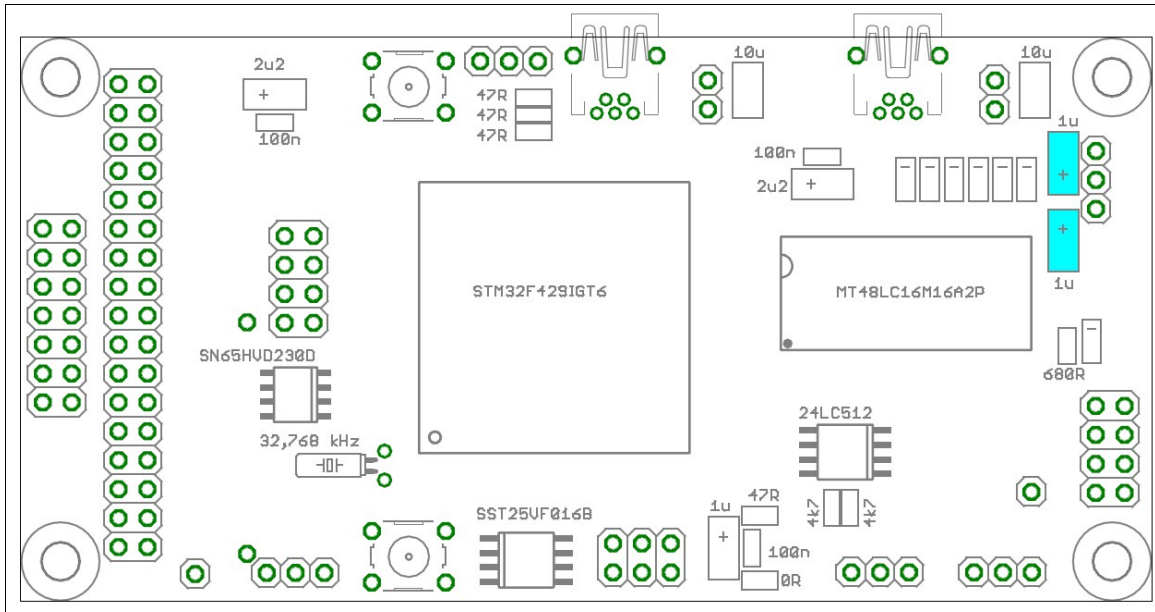


## 2.5. Spannungs-Regler

Falls das Board auch über externe 5V versorgt werden soll (z.B. über das USB-Kabel) dann folgende Bauteile bestücken :

TOP : CB16, CB18

BOTTOM : IC2, D1, CB17



### Check der Funktion :

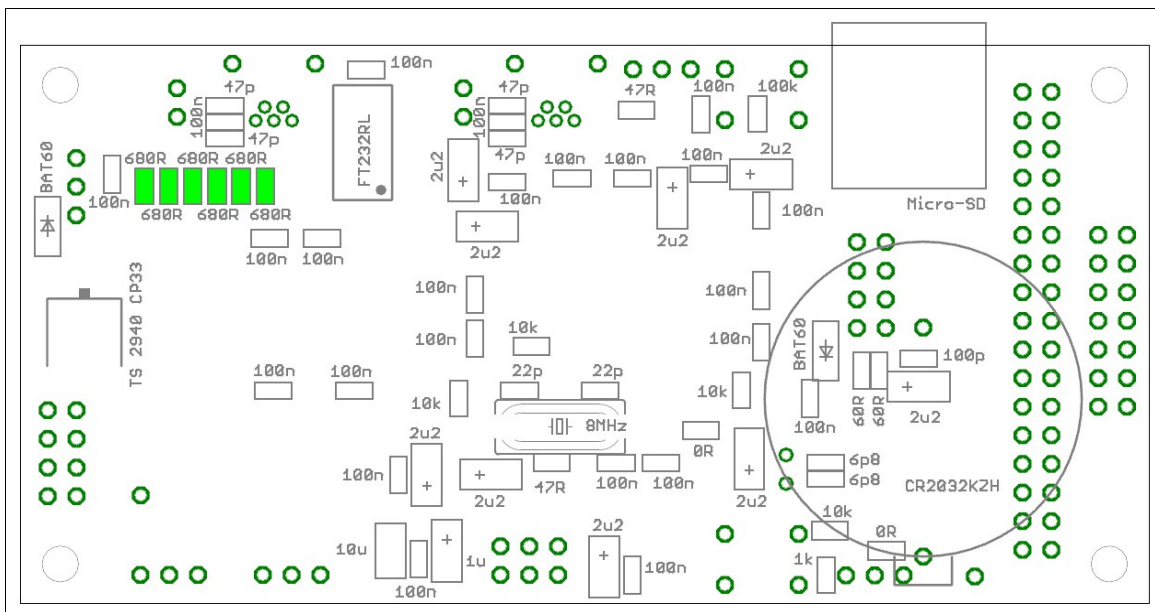
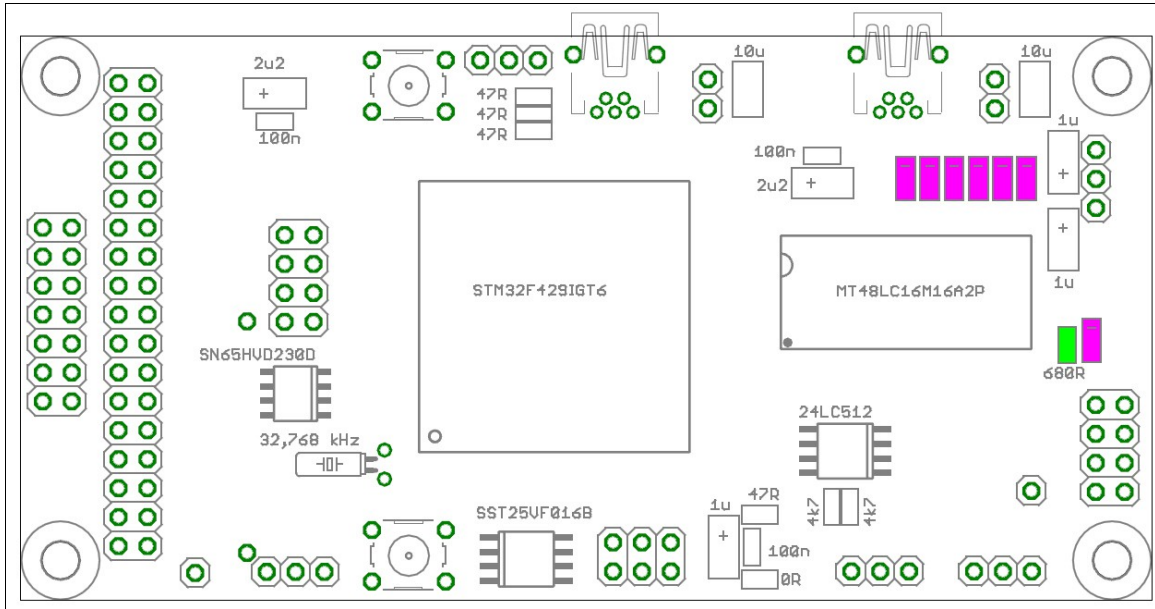
Über den Pinhead J2 das Board mit 5V versorgen und per Multimeter die 3V nachmessen.

## 2.6. Status LEDs

Falls Status LEDs benötigt werden :

TOP : LED1 bis LED7, R26

BOTTOM : R20 bis R25.



### Check der Funktion :

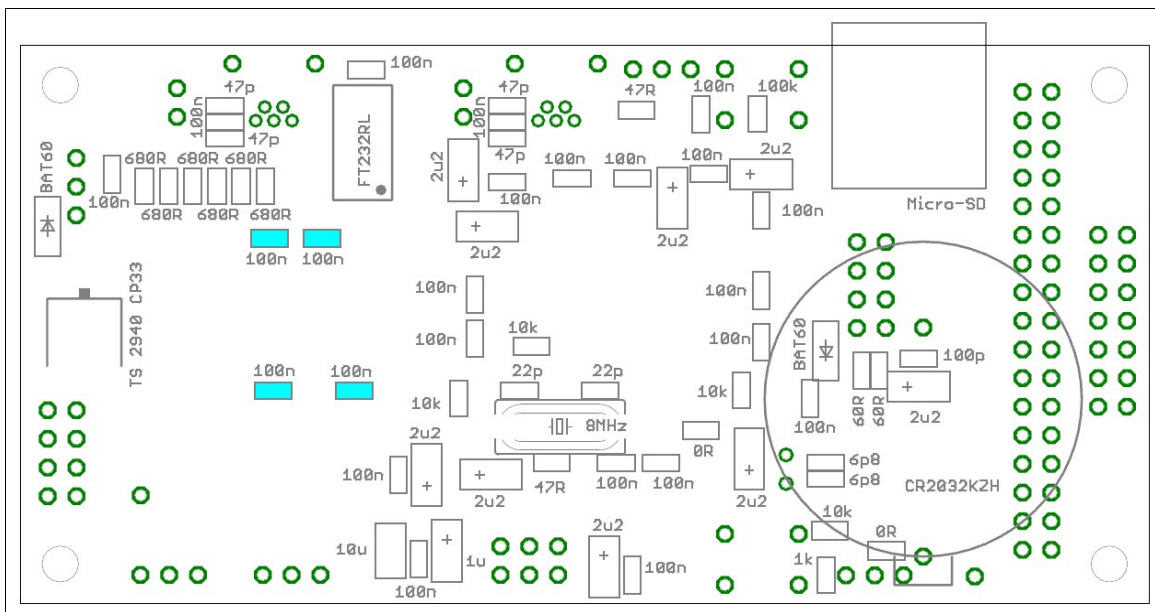
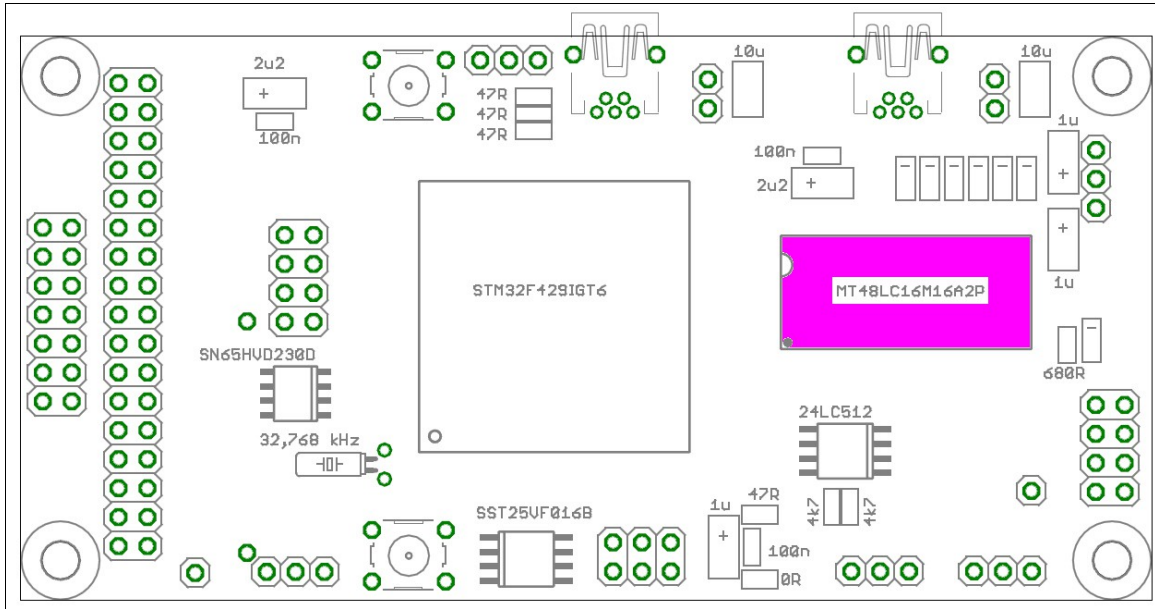
LED7 muss leuchten, sobald das Board mit Spannung versorgt wird

Für LED1 bis LED6 gibt es die Testsoftware (TST\_F429\_LED)

## 2.7. SDRAM

Falls das externe SDRAM benötigt wird :

TOP : IC3    BOTTOM : CB19 bis CB22



### Check der Funktion :

Für das SDRAM gibt es die Testsoftware (TST\_F429\_SDRAM)  
 Der Status wird per UART über J14 ausgegeben.

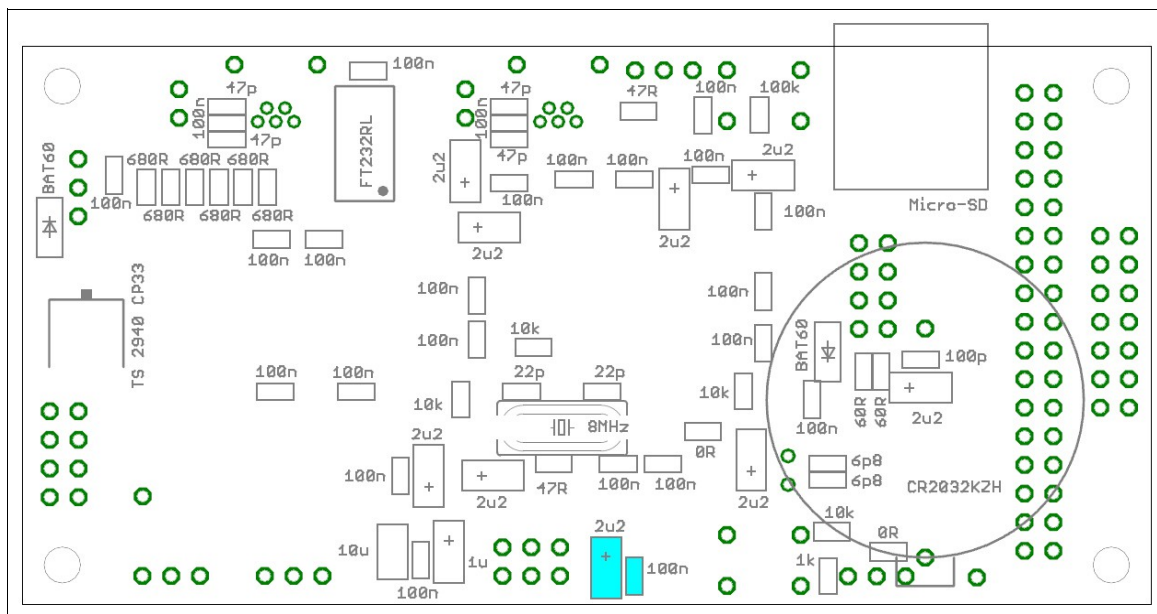
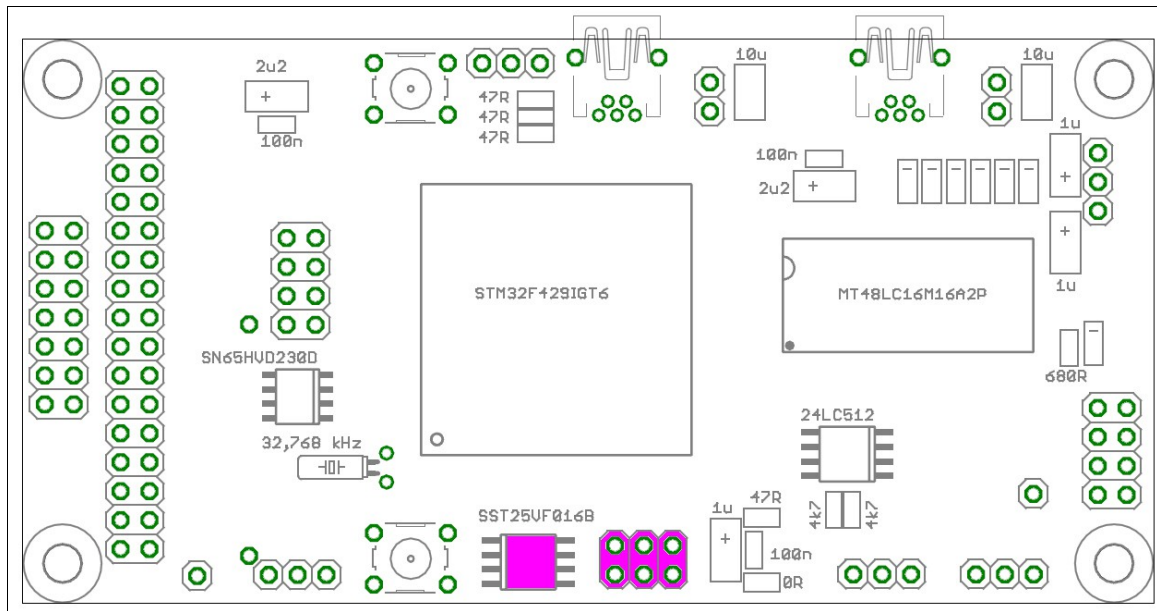




## 2.10. SPI

Falls das externe SPI-Flash benötigt wird :  
 TOP : IC5    BOTTOM : CB25, CB26 bestücken.

Falls der SPI-BUS extern benötigt wird : TOP : J7



### Check der Funktion :

Für das SPI-Flash gibt es die Testsoftware (TST\_F429\_SPI\_FLASH)

Der Status wird per UART über J14 ausgegeben.

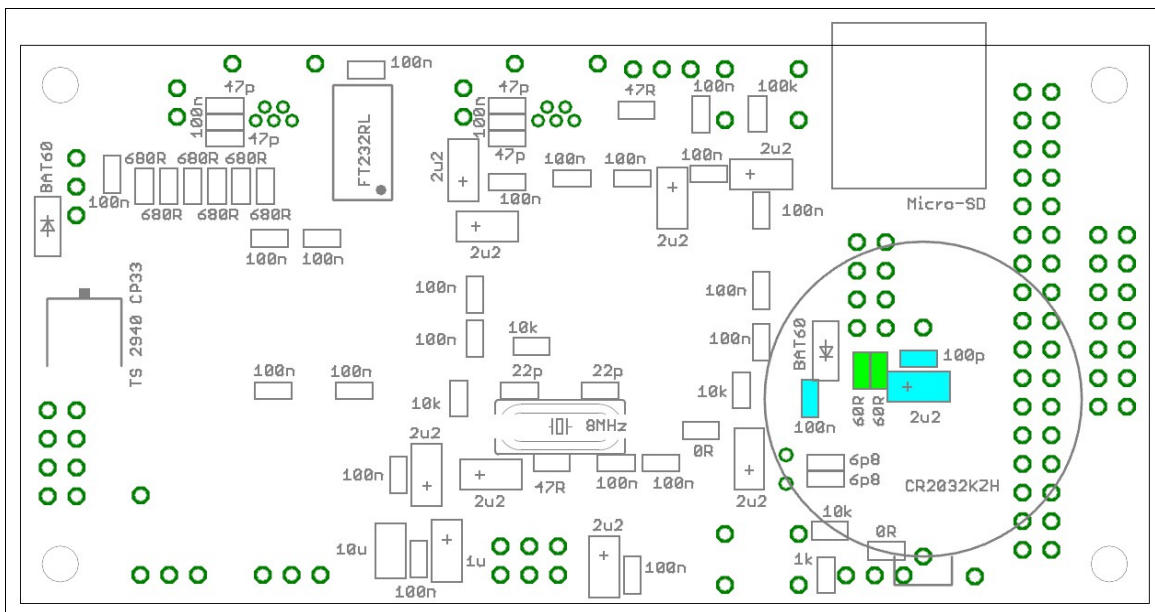
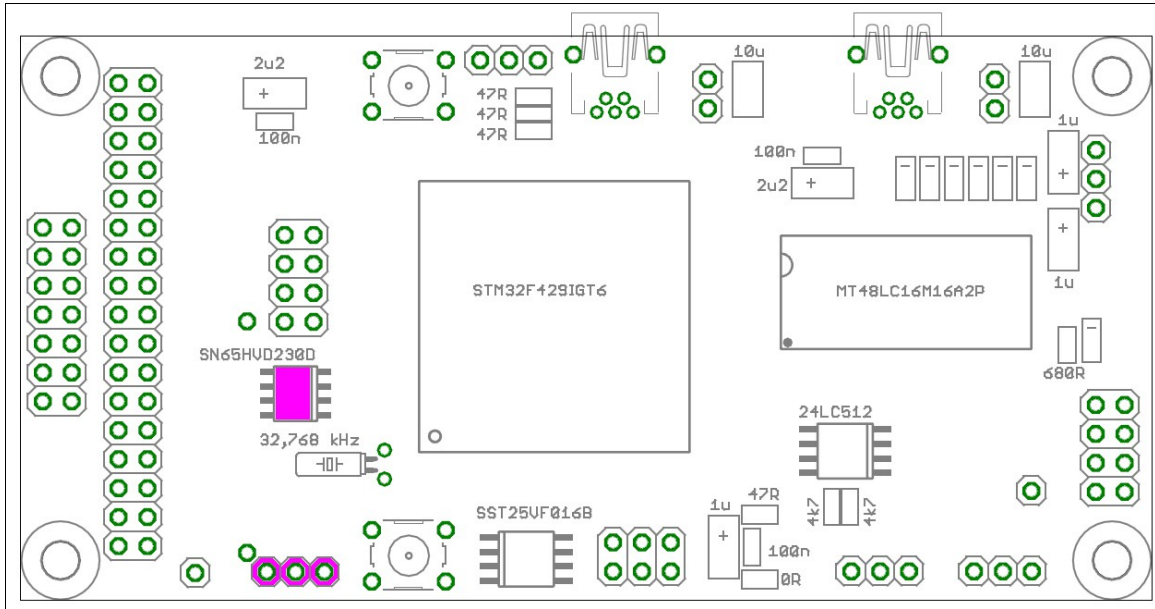


## 2.11. CAN

Falls die CAN-Schnittstelle benötigt wird :

TOP : IC6, J8

BOTTOM : C13, R17, R18, CB27, CB28



### Check der Funktion :

Für den CAN-Bus gibt es die Testsoftware (TST\_F429\_CAN)

Der Status wird per UART über J14 ausgegeben.

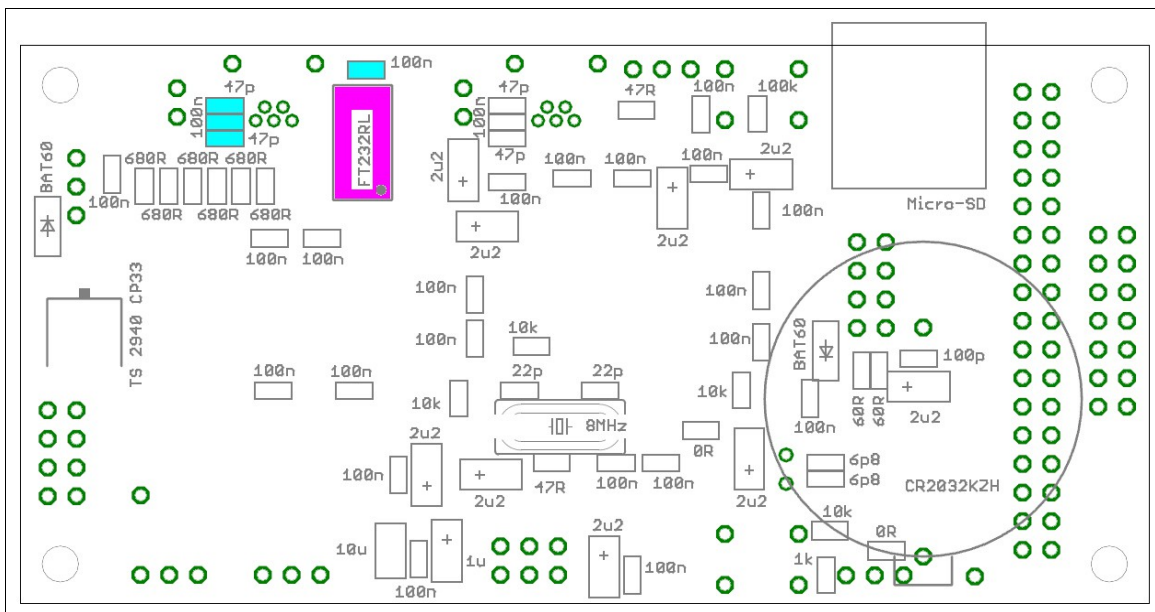
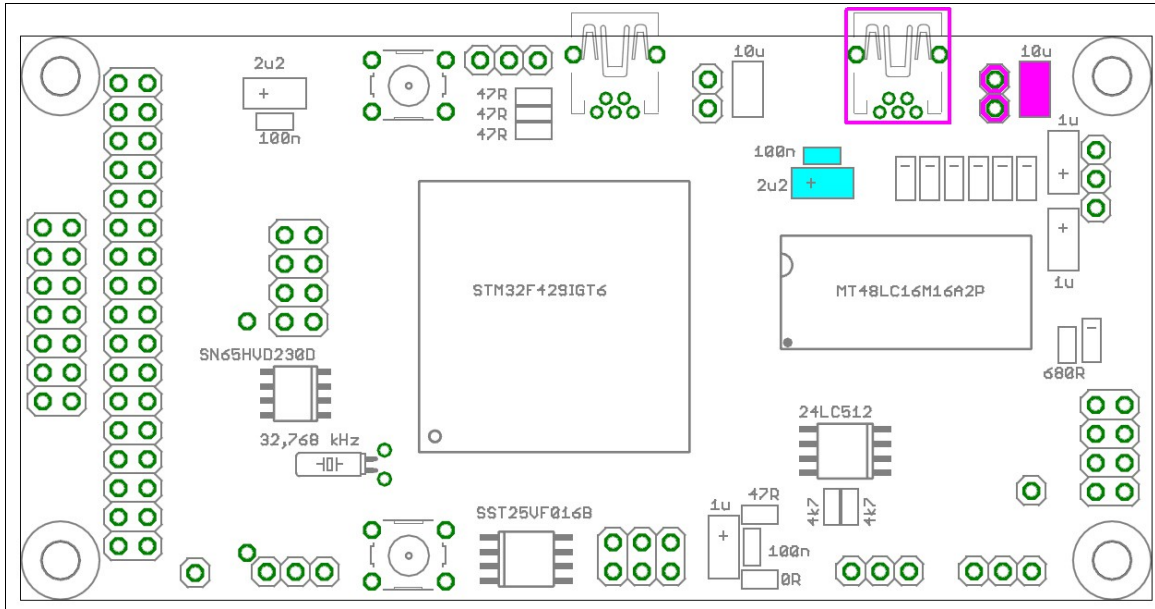
Hinweis : zum testen wird noch eine zweite CPU mit CAN-Transceiver benötigt.  
Diese muss CAN-Telegramme empfangen und senden können.

## 2.12. RS232-UART

Falls die UART-Schnittstelle benötigt wird :

TOP : X2, L3, J9, CB29, CB30

BOTTOM : IC7, C14 bis C17



### Check der Funktion :

Für die RS232-UART gibt es die Testsoftware (TST\_F429\_RS232)

Mit dieser kann eine UART Verbindung zu einem PC aufgebaut werden.

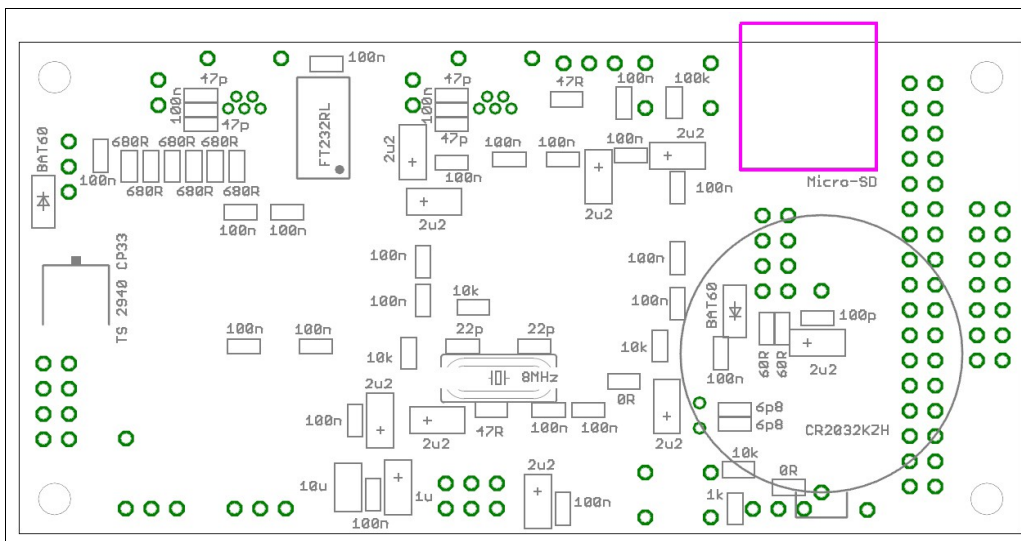
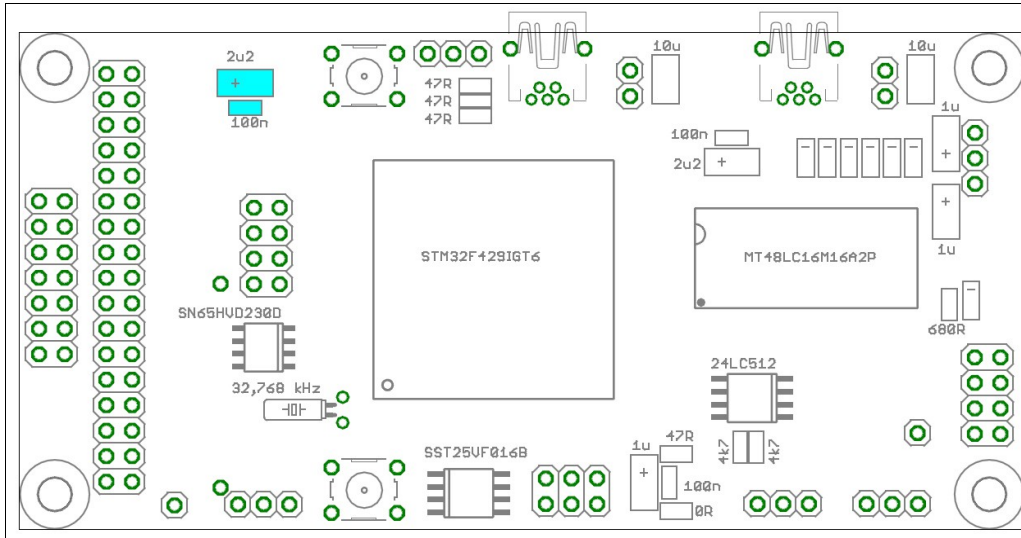
Der Status wird per UART über J14 ausgegeben.

### 2.13. SD-CARD

Falls die SD-Karte benötigt wird :

TOP : CB31, CB32

BOTTOM : X3



#### Check der Funktion :

Für die SD-Karte gibt es die Testsoftware (TST\_F429\_SDIO)

Der Status wird per UART über J14 ausgegeben.

Hinweis : Der SD-Halter ist einer dessen Deckel man öffnen muss um die Karte einzulegen. Dazu den Deckel etwas in Richtung Platinenrand schieben und dann aufklappen.

Jetzt die SD-Karte einlegen (auf die Ausrichtung achten) den Deckel schließen und wieder Richtung "Lock" schieben, bis er einrastet.

Der SD-Halter kann nur bei geöffnetem Deckel eingelötet werden.

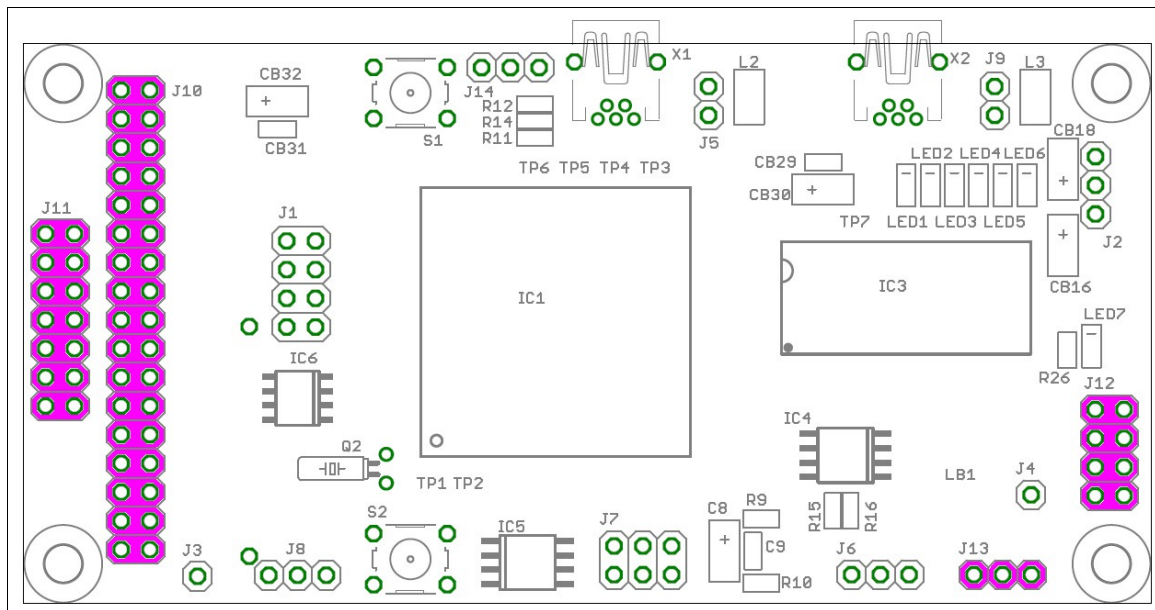
## 2.14. Sonstige Header

Falls ein TFT-Display benötigt wird (oder die 32 GPIOs der CPU) : J10 bestücken.

Falls die 12 Digitalen GPIOs der CPU benötigt werden : J11 bestücken.

Falls die 6 Analogen GPIOs der CPU benötigt werden : J12 bestücken.

Falls die 2 DAC-Kanäle der CPU benötigt werden : J13 bestücken.



### Check der Funktion :

Für die GPIOs gibt es die Testsoftware (TST\_F429\_Terminal)

Mit der passenden PC-Software Discovery-Terminal (ab Version 1.8) können alle GPIOs mit einem Oszi geprüft werden.

Hinweis : **Falls eine TFT-Adapterplatine auf den Header J10 gesteckt werden soll, ist es sinnvoll den Header J11 in der 90° Variante zu bestücken.**

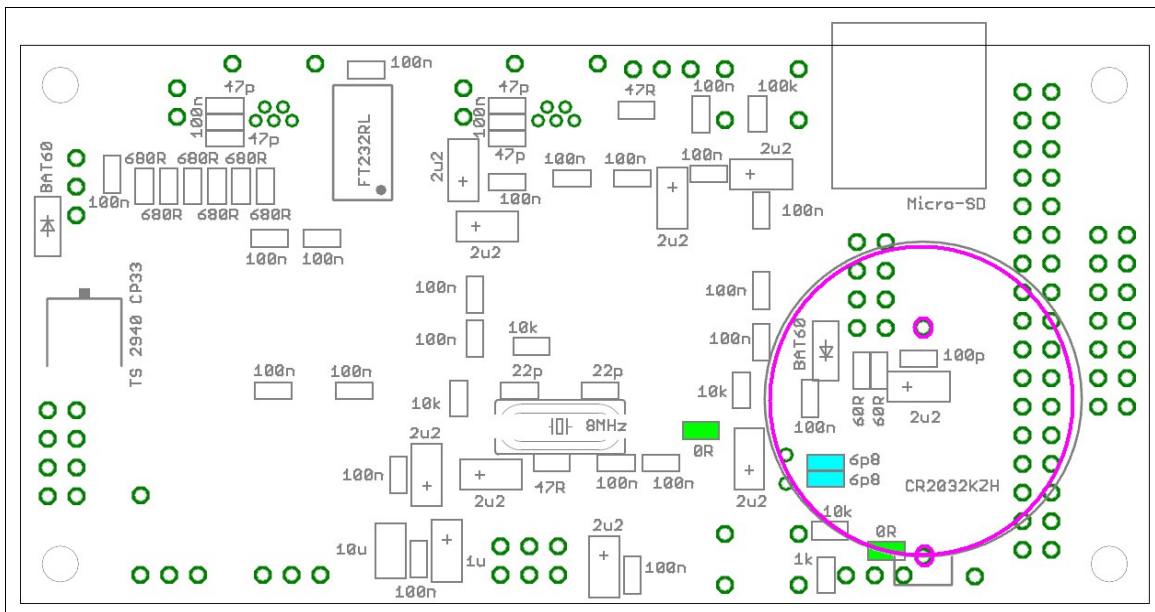
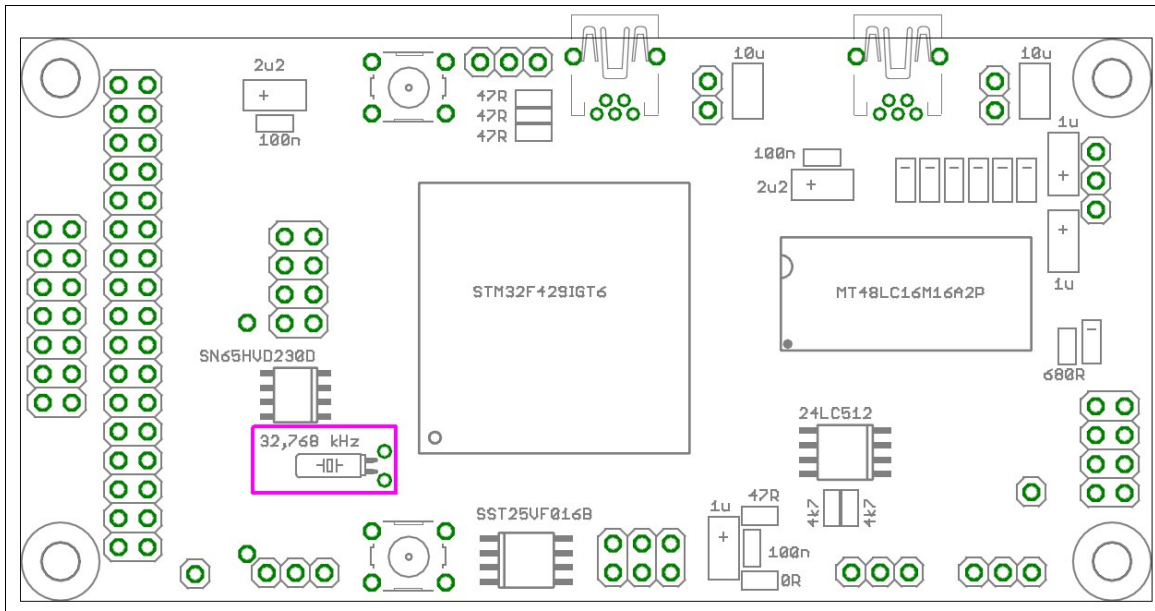
Dann kommt man auch bei gesteckter Adapterplatine noch an die Pins von J11. (weil die Adapterplatine den Header J11 von oben verdeckt)

### 2.15. RTC und Backup-Batterie

Falls der RTC-Quarz benötigt wird : TOP : Q2      BOTTOM : C18, C19 bestücken.

Falls die Backup-Batterie benötigt wird : BOTTOM : R19, B1 bestücken.

Falls die Backup-Batterie nicht benötigt wird : BOTTOM : R8 bestücken.



Hinweis :      Der Batterie-Halter (B1) verdeckt einige Bauteile und Header-Pins.  
 Diese Bauteile müssen zuerst bestückt werden.

### 3. Anhang

Das STM32F429-Projekt beinhaltet noch folgende Dokumente :

Schaltplan : F429\_Board\_SCH\_v100.pdf

Pinout der CPU : Pinout\_CPU\_v100.pdf

Portliste der CPU : Portlist\_CPU\_v100.pdf

Pinout der Header : Pinout\_Header\_v100.pdf

Bauteileliste : Bauteileliste\_v100.pdf

Bauteileplatzierung :

Oberseite (Namen) Top\_Names.pdf

Oberseite (Werte) Top\_Values.pdf

Unterseite (Namen) Bottom\_Names.pdf

Unterseite (Werte) Bottom\_Values.pdf