

Dipl.-Ing. Michael Zimmermann
Buchenstr. 15
42699 Solingen
☎ 0212 46267
🌐 <https://kruemelsoft.hier-im-netz.de>
✉ BwMichelstadt@t-online.de

Michelstadt (Bw)

DISPA

Hardware Version 1
Software Version 2.7

© 2020 – heute Michael Zimmermann



Wichtige Hinweise

Die hier beschriebenen elektrischen Schaltungen sind nur für den Einsatz auf Modelleisenbahnanlagen vorgesehen. Der Autor dieser Anleitung übernimmt keine Haftung für Aufbau und Funktion von diesen Schaltungen bei unsachgemäßer Verwendung sowie für beliebige Schäden, die aus oder in Folge Aufbau oder Betrieb dieser Schaltungen entstehen.

Für Hinweis auf Fehler oder Ergänzungen ist der Autor dankbar.

Ein Nachbau ist nur zum Eigenbedarf zulässig, die kommerzielle Nutzung Bedarf der schriftlichen Zustimmung des Autors.

Inhalt

1	DISPA	3
2	Anschluss und Bedienung.....	4
2.1	Anschluss	4
2.2	... und Bedienung.....	4
2.3	FREDI-Diagnose	5
2.3.1	SV auslesen und anzeigen	5
2.3.2	FREDI-Test	7
2.4	QR-Code-Leser.....	8
2.4.1	Aufbau und Beispiel für einen QR-Code.....	9
2.4.2	Verwendeter QR-Code-Leser	10
2.5	Funktionstastenmodi.....	10
2.5.1	Erläuterung.....	10
2.5.2	Setzen der Funktionstastenmodi	11
2.5.3	Zurücksetzen aller Funktionstastenmodi	11
3	Hardware.....	12
4	Software	12
4.1	HEX-Dateien	12
4.2	Quellcode	12
4.2.1	Die Funktion „notifyThrottleId“	13
4.2.2	DispaSetup.....	14
4.3	Den AVR flashen	14
4.4	Versionsgeschichte.....	14
5	Schaltpläne und Stücklisten	15
5.1	Prozessorplatine („LN-Universal“).....	17
5.1.1	Stückliste Prozessorplatine.....	18
5.2	Konstantstromquelle.....	20
5.2.1	Stückliste Konstantstromquelle.....	20
5.3	LCD-Anzeige-Einheit	21
5.3.1	Stückliste LCD-Anzeige-Einheit.....	22
5.4	I ² C-OLED-Anzeige-Einheit	23
5.4.1	Stückliste I ² C-OLED-Anzeige-Einheit.....	24
5.5	Tastatur-Anschluss	26
5.5.1	Stückliste Tastatur-Anschluss	27
6	Experten-Informationen.....	28
6.1	Kommunikation: LocoNET®-Telegramme	28

All Schematic and Board are licensed under a
 Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License,
 see <<http://creativecommons.org/licenses/by-nc-sa/3.0/legalcode>>.

This program is free software: you can redistribute it and/or modify
 it under the terms of the GNU General Public License as published by
 the Free Software Foundation, either version 3 of the License, or
 (at your option) any later version.

This program is distributed in the hope that it will be useful,
 but WITHOUT ANY WARRANTY; without even the implied warranty of
 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 GNU General Public License for more details.

You should have received a copy of the GNU General Public License
 along with this program. If not, see <<http://www.gnu.org/licenses/>>.

1 DISPA

Fahrzeuge müssen einem Handregler (FRED/FREDI usw.) zugewiesen werden, damit dieser sie steuern kann. Zuweisung?

Zuweisung (neudeutsch: Dispatchen) bedeutet, dass die gewünschte Lokadresse in der Zentrale zum 'Übernehmen' ausgewählt wird. Der nächste Handregler, der die Übernahmetasten (i.d.R. eine Tastenkombination) betätigt, erhält dann diese Lok zugewiesen (heißt: die Zentrale merkt sich dann, welcher Regler für diese Lok zuständig ist, und leitet die Befehle dieses Reglers an die entsprechende Lok weiter).

Im FREMO gab es dazu früher eine Box (**FR**emos **AN**ehmliche **Z**uordnungsbox), die aber nicht mehr verfügbar ist.

Aktuell wird im FREMO in der Regel jeder Lok vor einem oder am Fahrtag vor dem eigentlichen Betrieb ein Handregler zugewiesen.

Mittlerweile existieren in unserer Modulgruppe so viele DCC-fähige Lokomotiven, dass es nicht möglich ist, für jede Lok einen eigenen Handregler zu haben.

Daraufhin hatte ich einen neuen **FRANZ** auf PIC-Basis entwickelt.

Und die Zeit schreitet voran...

In der FREMO-Zeitschrift Hp1 (Heft 2/2011 Seite 22) wurde **DISPA** von Heiko Herholz vorgestellt – ein neuer **FRANZ** auf AVR-Basis, einfach zu bedienen.

Die Zuordnung der Handregler zu den Lokomotiven erfolgt nach wie vor direkt über die Zentrale – oder eben mit dem neuen **DISPA**.

Und was ist dann mit dem **FRANZ**? Der kann natürlich weiterhin verwendet werden. **DISPA** dagegen bietet andere Möglichkeiten (werden alle **FRANZ** - Möglichkeiten überhaupt benötigt?), ist dafür daher auch einfacher zu bedienen. Ein Pluspunkt für **DISPA** ...

2 Anschluss und Bedienung

2.1 Anschluss ...

DISPA ist ein StandAlone-Gerät:

- autark, weil er eine eigene Spannungsversorgung (9V-Blockbatterie) hat
- und wird **niemals** mit einem LocoNET® verbunden.

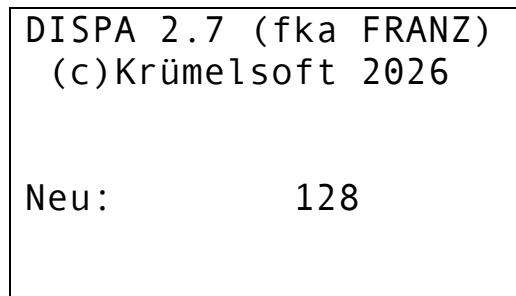
Ist ein Handregler mit **DISPA** verbunden und ist / wird **DISPA** eingeschaltet und am Handregler leuchtet keine LED, so kann es an

- zu geringer Batteriespannung liegen (→ Batterie austauschen oder externe Spannungsversorgung verwenden).
- einem fehlerhaften Handregler liegen. Vom **DISPA** aus erfolgt die Spannungsversorgung über Pin 6 der LocoNET®-Buchse. In diesem Fall ist die Diode D2 im Handregler (hier dem FREDI) zu prüfen (kalte Lötstelle?)! Es wurden bewusst Pin 1 und Pin 6 nicht miteinander verbunden: wenn **DISPA** irrtümlich mit dem LocoNET® der Anlage verbunden würde, gäbe es auf den RailSync-Leitungen einen Kurzschluss!

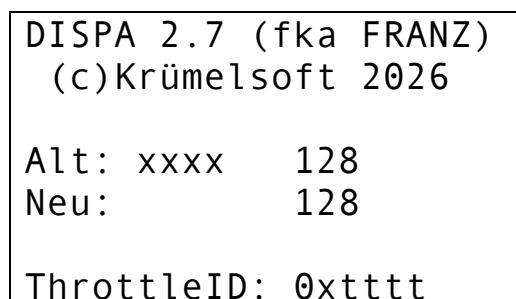
2.2... und Bedienung

Die Bedienung von **DISPA** ist denkbar einfach:

Nach dem Einschalten erscheint auf dem Display¹ die folgende Information:



Ist vor dem Einschalten bereits ein zugewiesener Handregler angeschlossen, so werden weitere Informationen² angezeigt:



¹ Es wird die Anzeige auf dem OLED dargestellt,
die Anzeige auf einem LCD enthält nur die Zeilen zu „Alt...“ und „Neu...“

² xxxx = alte Adresse auf dem Handregler, tttt = Handregler-Kennung im Hexadezimal-Format

Spätestens nach dem Einschalten wird dann ein Handregler angeschlossen. Hat dieser bereits ein Fahrzeug zugeordnet, so ist das Fahrzeug zunächst an die Zentrale (notfalls auch an den **DISPA**) zurückzugeben³.

Die weiteren Schritte sind einfach:

- gewünschte Lok-Adresse mit der Tastatur eingeben,
diese wird hinter dem Text Neu angezeigt
 - o mit * kann die Eingabe gelöscht werden
 - o mit # kann die Fahrstufe (Voreinstellung: 128) geändert werden
- mit dem Handregler die eingegebene Adresse und Fahrstufe übernehmen³
- fertig 😊

2.3 FREDI-Diagnose

Allgemein gilt im Diagnosemodus:

- mit der Taste * kann der FREDI-Diagnose-Modus jederzeit beendet werden, **DISPA** kehrt dann zum Dispatch-Modus zurück.

2.3.1 SV auslesen und anzeigen

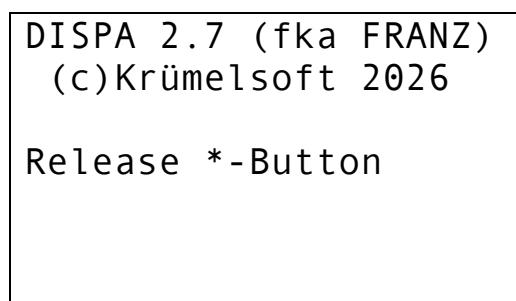
FREDIs ab Software-Version 1.8 haben interne Konfigurationswerte (SV, **System-Variable**)⁴ die mit geeigneter Hard- und Software ausgelesen und geändert werden können. JMRI⁵ ist beispielsweise eine geeignete Software. Die ist aber nicht immer zur Hand...

Mit der neuen **DISPA**-Software Version 2.1 ist es möglich, zu Diagnosezwecken diese SVs auszulesen und anzuzeigen. Diese Funktion ist in **DISPA**-Version 2.1 verfügbar, wenn der **DISPA** über ein OLED verfügt, und ab **DISPA**-Version 2.2 auch mit LCD⁶.

Für **DISPA**-Version 2.3 wurde das Auslesen um die SVs 8, 9 und 10 erweitert, für **DISPA**-Version 2.4 das Auslesen von SV 18 bis SV 34.

Zur Nutzung der Funktion ist

- der **DISPA** auszuschalten
- die Taste * zu betätigen und
- anschließend der **DISPA** einzuschalten und
- dann die Taste * mindestens für zwei Sekunden gedrückt zu halten (bis die Anzeige wie folgt aussieht):



³ Für die allgemeine Bedienung der Handregler wird auf die zugehörige Dokumentation verwiesen

⁴ Zur Bedeutung der SVs: siehe *FrediSvUsage.pdf* (<https://groups.io/g/fremodcc/files/FREDI/FrediSvUsage.pdf>)

⁵ Siehe auch: Krümelbahn Info 11 - JMRI - Universalwerkzeug für die Modellbahn

⁶ Es wird die Anzeige auf dem OLED dargestellt, bei einem LCD werden jeweils nur zwei SVs auf einer Seite angezeigt

- nach dem Loslassen der Taste * erscheint:

DISPA 2.7 (fka FRANZ)
(c)Krümelsoft 2026

Connect FREDI

- jetzt ist der FREDI (wenn er noch nicht eingesteckt ist) innerhalb von 15 Sekunden einzustecken
- der Auslesefortschritt der SVs ist durch die laufenden Punkte erkennbar:

DISPA 2.7 (fka FRANZ)
(c)Krümelsoft 2026

Connect FREDI

...

- scheitert das Auslesen, erscheint der Text Communication Error
- ist das Auslesen erfolgreich, erscheint Seite 1 der SV-Übersicht⁷:

FREDI-SV - Part 1

SV 2 : 22
SV 3, 4: 0x494B
SV 5 : Atmega328P
SV 6 : Atmega328P

⁷ Die angezeigten SV-Werte sind hier beispielhaft und abhängig vom verwendeten FREDI

- mit den Tasten **1** bis **5** (OLED, bei LCD bis **9**) kann zwischen den SV-Anzeigeseiten umgeschaltet werden:

FREDI-SV - Part 2	FREDI-SV - Part 3
SV 8, 9: 3 SV10 : 128 FS SV11 : operating	SV12 : 10000110 Matrix-FREDI (SWD) One Shift button Forw-Off-Rev-Switch
FREDI-SV - Part 4	FREDI-SV - Part 5
SV13,14: 2.2 SV15-17: 20.08.23 SV43 : no OPC_IMM SV44 : no BRAKE	SV18...34 1111111 F01234567890123456

2.3.2 FREDI-Test

Ab **DISPA**-Version 2.3 besteht die Möglichkeit, die Funktion des FREDI zu testen: es werden Geschwindigkeit, Richtung und der Status der Funktionen F0...F16 angezeigt.

- Wenn der FREDI eine gültige Fahrzeugadresse hat, kann vom Diagnosemodus ([Kapitel 2.3.1 SV auslesen und anzeigen](#)) ausgehend mit der Taste **#** auf eine Testseite gewechselt werden. Hier werden in der oberen Zeile Geschwindigkeit, Richtung und in der untersten Zeile der Status der Funktionen F0...F16 angezeigt:

FREDI-SV - Test
< θ >
1111111
F01234567890123456
.....

- mit den Tasten **1** bis **5** (OLED, bei LCD bis **9**) kann wieder zu den SV-Anzeigeseiten gewechselt werden.

2.4 QR-Code-Leser

Die FREDI-Zuweisung (Dispatchen) wird jetzt einfacher!

Mit der neuen **DISPA**-Version 2.4 ist es möglich, einen QR-Code-Leser anzuschließen und so einen QR-Code zu lesen, der Angaben zum Fahrzeug enthält:

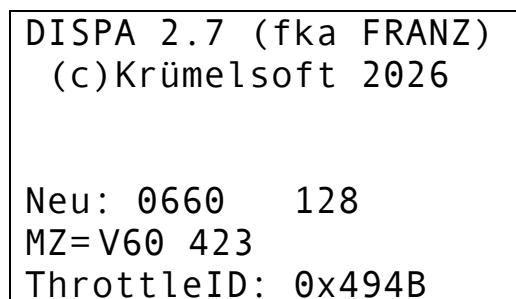
- Fahrzeugadresse
- Anzahl der Fahrstufen (Angabe optional, ohne Angabe = unverändert, es wird die aktuelle Eingabe verwendet)
- Fahrzeugbesitzer (Angabe optional)
- Fahrzeugname (Angabe optional)
- Funktionstastenmodus⁸ (Angaben optional, ohne Angabe = der aktuelle Modus der Funktionstasten bleibt unverändert)

Nach dem Lesen⁹ eines gültigen QR-Codes werden die Daten an den FREDI übertragen, ein anschließendes Dispatchen am FREDI ist nicht mehr erforderlich.

Nach der Datenübertragung

- bleiben die Status der LED unverändert
- und der FREDI kann vom **DISPA** getrennt und an das LocoNET® der Zentrale angeschlossen werden
- fertig 😊

Nach einem Scan sieht die Anzeige im OLED beispielhaft so aus:



⁸ Siehe [Kapitel 2.5 Funktionstastenmodi](#)

⁹ Die Bedienung des QR-Lesers ist abhängig vom Modell

2.4.1 Aufbau und Beispiel für einen QR-Code

Ausgangsbasis für einen QR-Code ist ein Text, der dann in den QR-Code gewandelt wird, z.B.:

```
loco.owner=MZ&loco.name=V100 1088&loco.address=610&loco.steps=128&loco.f2=1
```

Schlüsselwort	Wert	Anmerkungen
<code>loco.address</code>	<code>610</code>	Gültige DCC-Adresse im Bereich 1 bis 9999
<code>loco.steps</code>	<code>128</code>	Optional, zulässige Werte: <code>128A, 28A, 128, 28, 14, 28M</code> , default ist <code>128</code>
<code>loco.owner</code>	<code>MZ</code>	Optional, es empfiehlt sich, hier bekannte Kürzel (z.B. jeweils der erste Buchstabe des Vor- und Nachnamens) zu verwenden
<code>loco.name</code>	<code>V100 1088</code>	Optional, der Text ist möglichst auf 12 bis 18 Zeichen zu begrenzen (Anzeigemöglichkeiten des verwendeten Displays)
<code>loco.fx</code>	<code>1</code>	Optional, zulässige Zahlen <code>x</code> hinter dem <code>f</code> im Schlüsselwort: <code>0</code> bis <code>16, 255</code> Zulässige Werte: <code>0</code> = Funktionstaster wirkt, solange er gedrückt wird (‘on while pressed’, Tastfunktion), <code>1</code> = Funktionstaster wechselt bei jedem Tastendruck seinen Status (‘toggle’, Schaltfunktion), default ist <code>1</code> Es sind mehrere <code>loco.fx</code> Schlüsselworte zulässig <code>x=255</code> wirkt auf alle Funktionstaster gemeinsam; so bewirkt z.B. <code>loco.f255=1</code> , dass alle Funktionstaster auf default (<code>1</code>) gesetzt werden

Schlüsselwort und *Wert* werden durch `=` getrennt, mehrere Schlüsselworte durch ein `&`. Unbekannte Schlüsselworte werden beim Einlesen ignoriert. Die Reihenfolge der Angaben „*Schlüsselwort* `=` *Wert*“ ist beliebig.

Aufgrund der Datenstruktur sind

- die Trennzeichen `=` und `&`,
- die identifizierenden Buchstabenkombinationen:
`o.a, o.f, o.n, o.o` und `o.s`
- sowie Umlaute

innerhalb der Werte nicht zulässig.

Der QR-Code für das obige Beispiel sieht dann so aus:



Zur Erstellung von QR-Codes gibt es im Internet zahlreiche Werkzeuge (auch online), für Microsoft-Office ist ein Addin (QR4Office) verfügbar.

Der erstellte QR-Code kann dann ausgedruckt werden auf:

- Frediketten
- Fahrzeugkarten
- einem einfachen Blatt
- als Grafikdatei
- ...

Der Name der verwendeten *Schlüsselworte* ist nicht willkürlich gewählt: diese werden auch beim QR-Code für den wiFred¹⁰ (<https://us-trains.de/?p=510>) verwendet, der QR-Code für einen wiFred kann daher auch beim FREDI verwendet werden. Ohne http-Vorspann ist der FREDI-QR-Code jedoch nicht für den wiFred nutzbar.

2.4.2 Verwendeter QR-Code-Leser

Zum Lesen eines QR-Codes gibt es einige Lesegeräte, die an einen AVR (bzw. Arduino) angeschlossen werden können.

Grundvoraussetzung für den zu verwendenden QR-Code-Leser ist zum einen eine 5V-Spannungsversorgung, zum anderen eine TTL-kompatible Schnittstelle, da der verwendete AVR zwar eine USB-Schnittstelle unterstützt, dies aber nur als USB-Slave und nicht als USB-Master.

Der Anschluss des QR-Code-Lesers an **DISPA** erfolgt über eine zusätzliche SUB-D9-Buchse X4 ([Kapitel 5.1 Prozessorplatine \("LN-Universal"\)](#)), es werden die Anschlüsse +5V, GND, RxD und TxD verwendet:

- RxD vom **DISPA** kommt dann an TxD vom QR-Code-Leser und
- TxD vom **DISPA** kommt dann an RxD vom QR-Code-Leser

Für den Datenaustausch über die UART-Schnittstelle ist die Schnittstelle im **DISPA** auf 9600Bd 8N1¹¹ eingestellt. Als abschließendes Zeichen sollte der QR-Code-Leser ein Zeilenendezeichen (CR oder LF) senden. Der QR-Code-Leser ist entsprechend zu parametrieren.

QR-Code-Lesegeräte benötigen laut Hersteller deutlich mehr Strom¹² als ein FREDI – der Spannungsregler IC4 auf der Prozessorplatine ist unbedingt anzupassen (siehe [Kapitel 5.1.1 Stückliste Prozessorplatine](#))! Da die Lebensdauer einer 9V-Blockbatterie endlich ist, ist es ggf. günstiger, die 9V- bzw. 12V- Einspeisung über K6 (siehe [Kapitel 5.2 Konstantstromquelle](#)) zu nutzen. Meine Test-Scans mit der 9V-Blockbatterie waren erfolgreich, wie lange die Batterie hält, muss sich zeigen... .

2.5 Funktionstastenmodi

2.5.1 Erläuterung

Auch wenn es kaum bekannt und wenig genutzt ist: eine Funktionstaste im FREDI kann entweder als Taster (Funktion ist aktiv, solange die Taster gedrückt ist) oder als Schalter¹³ (Funktion wechselt bei jedem Tastendruck seinen Zustand) definiert und genutzt werden.

Diese Funktionsmodi sind immer FREDI-bezogen und wirken fahrzeugunabhängig, d.h. sie bleiben bei jedem Dispatchen unverändert, wenn keine Änderungen über den QR-Code oder JMRI (siehe auch [Krümelbahn Info 11 - JMRI - Universalwerkzeug für die Modellbahn](#), Seite 21) vorgenommen werden. Eine Vereinheitlichung der Zuordnung von Funktionen zu Funktionstasten ist daher sicherlich sinnvoll.

¹⁰ Beispiel für einen QR-Code als Text für den wiFred (die http-Adresse variiert):

<http://192.168.1.102/index.html?loco.address=610&loco.direction=2&loco.longAddress=on&loco=1>

¹¹ 8 Datenbits, ohne Paritätsbit, 1 Stopbit

¹² Mein QR-Code-Leser ist von WaveShare (https://www.waveshare.com/wiki/Barcode_Scanner_Module), passt in das Gehäuse TEKO 10007 und wurde hier gekauft:

<https://www.welectron.com/Waveshare-1D-2D-Barcode-Scanner-Module>

Von mir gemessener Stromverbrauch: im Standby ca. 5mA, für die Scandauer ein Peak von bis zu 150mA

¹³ Defaulteinstellung

2.5.2 Setzen der Funktionstastenmodi

Die Grundeinstellung für jede Funktionstaste an einem FREDI ist „Schalter“. Soll jetzt eine Funktionstaste als „Taster“ verwendet werden, so geht das

- einerseits über **JMRI** (siehe auch [Krümelbahn Info 11 - JMRI - Universalwerkzeug für die Modellbahn](#), Seite 23)
- andererseits über die Angabe im QR-Code:
 - o `loco.fx=0` verwendet künftig die Funktionstaste **x** als Taster
 - o `loco.fx=1` verwendet künftig die Funktionstaste **x** als Schalter

Innerhalb des QR-Codes können alle Funktionen F0 bis F16 angegeben werden (Sonderfall F255 (**x=255**), siehe oben).

2.5.3 Zurücksetzen aller Funktionstastenmodi

Was dann passieren kann: die Funktionstastenmodi sind „durcheinander“ und passen nicht mehr zum nächsten Fahrzeug – dann hilft am besten das Zurücksetzen aller Funktionstastenmodi mit diesem QR-Code:

```
loco.f0=1&loco.f1=1&loco.f2=1&loco.f3=1&loco.f4=1&loco.f5=1&loco.f6=1&  
loco.f7=1&loco.f8=1&loco.f9=1&loco.f10=1&loco.f11=1&loco.f12=1&  
loco.f13=1&loco.f14=1&loco.f15=1&loco.f16=1
```



...und ab V2.7 mit diesem QR-Code:

```
loco.f255=1
```



3 Hardware

Die entsprechenden Schaltbilder sind im [Kapitel 5 Schaltpläne und Stücklisten](#) zu finden.

Für **DISPA** gibt es keine „zentrale“ Platine: es wurden Platinen aus bestehenden Komponenten verwendet und miteinander verbunden.

Aus diesem Grund heraus sind auch Bauteilnamen mitunter mehrfach vergeben: Bauteilnamen beziehen sich deshalb immer auf die Platine, die im entsprechenden Kapitel beschrieben wird. Seitenangaben auf den Schaltplanseiten beziehen sich auf den Schaltplan und nicht auf dieses Dokument.

Alle Platinen sind professionell gefertigt und haben einen beidseitigen Bestückungsaufdruck, auf Bestückungspläne und -anleitungen wird daher verzichtet.

Viele Bauteile sind in der SMD-Variante verbaut, um die Platine kompakt gestalten zu können. SMD-Bauteile sind in der Stückliste farbig hervorgehoben.

Praxis für das Löten von SMD-Bauteilen sollte vorhanden sein.

4 Software

Der Prozessor benötigt eine Software, um seine Aufgabe zu erfüllen.

Die originale Software (<https://sourceforge.net/p/embeddedloconet/svn/HEAD/tree/old-cvs/trunk/apps/Dispa/>) des FREMO wurde erweitert und auf die Entwicklungsumgebung für die frei verfügbare [Arduino-IDE](#) sowie auf die verwendeten Platinen angepasst (geänderter Anschluss der Tastatur, optional: Ansteuerung eines OLED).

Die gesamte Software ist gemäß der zugehörigen Lizenz verfügbar.

4.1 HEX-Dateien

Im GitHub-Repository befinden sich im Ordner „Hexfiles“ (<https://github.com/Kruemelbahn/DISPA/tree/main/Hexfiles>) die bereits mit dem Quellcode kompilierten HEX-Dateien. Diese Hex-Dateien können mit einem AVR-Programmiergerät auf den Prozessor geladen werden (siehe [Kapitel 4.3 Den AVR flashen](#)).

Die Namensgebung der Dateien bezeichnet die Ausbaustufe der Software:

- „lcd“: kennzeichnet die Verwendung mit einem LCD
- „oled“: kennzeichnet die Verwendung mit einem OLED

4.2 Quellcode

Der Quellcode im Hauptverzeichnis (<http://www.github.com/Kruemelbahn/DISPA>) ist genau wie meine zugehörigen Bibliotheken unter GitHub verfügbar.

Der Quellcode wird nur benötigt, wenn

- Man neugierig ist
- Oder den Quellcode ändern und somit neu kompilieren möchte.
Zum Kompilieren wird die aktuelle Arduino-IDE benötigt.

Ansonsten reichen die Hex-Dateien völlig aus.

In der Datei DISPA.ino kann die Art der Anzeige festgelegt werden:

```
#define LCD      Ausgabe auf LCD, nicht zusammen mit OLED
#define OLED     Ausgabe auf OLED, nicht zusammen mit LCD
```

Die Kompilierung erfolgt für das Board „Arduino UNO“. Für eine erfolgreiche Kompilierung sind nachfolgende Arduino-Bibliotheken erforderlich:

Arduino-Library	(Link)
Adafruit-GFX-Library_master	https://github.com/adafruit/Adafruit-GFX-Library
Bounce	https://www.pjrc.com/teensy/td_libs_Bounce.html
LocoNET®	https://mrrwa.org/loconet-interface/
HeartBeat	
I2CKeypad	
LCDPanel	erfordert: Adafruit-GFX-Library
OLEDPanel	

(Bibliotheken, die grün hinterlegt sind, stehen in meinem [GitHub](#) zur Verfügung.)

Notwendige Bibliotheken können über die Arduino-IDE hinzugefügt werden.

Mit dem Kompilieren entsteht eine Hex-Datei, die vor der Inbetriebnahme der Schaltung in den ATMEGA 328 geflashed (gebrannt) wird.

4.2.1 Die Funktion „notifyThrottleId“

Ab V2.7 steht die Funktion „notifyThrottleId“ zur Verfügung.

Diese Funktion wird nach der Anzeige der FREDI-Id (*Throttle-ID*) aufgerufen, um weitere Angaben anzugeben, z.B. das Kürzel des FREDI-Besitzers.

Funktionsdefinition in Dispa.ino:

```
=====
 * Function can be used for additional handling with ThrottleID, e.g. display ThrottleOwner if known
 * function is actually called inside 'lcd_ThrottleId()', regardless if LCD or OLED is defined

 * should return true, if owner could be displayed
 */
extern boolean notifyThrottleId(const uint16_t uiThrottleId) __attribute__((weak));
=====
```

Durch diese Definition mit der Angabe „__attribute__((weak))“ kann die Funktion existieren, muss es aber nicht.

Mögliche Implementierung, z.B. in einer neuen Datei „NotifyThrottleId.ino“:

```
//== notifyThrottleId for personal use with DISPA --- NOT FOR PUBLIC / COMMON USE --- ==
static boolean notifyThrottleId(const uint16_t uiThrottleId)
{
    return false;
}
```

Rückgabewert:

- **false**, wenn die Funktion nicht oder fehlerhaft ausgeführt wurde
- **true**, wenn die Funktion erfolgreich ausgeführt wurde

In den aktuellen Hex-Dateien wird diese Funktion *nicht* aufgerufen. Die Erstellung der Funktion bleibt somit dem Anwender überlassen (da ich die Zuordnung zu den mir bekannten Besitzern nicht öffentlich mache).

4.2.2 DispaSetup

DispaSetup.ino ist ein kleines Arduino-Programm, mit dem OLED und Tastatur auf korrekten Anschluss getestet werden können. Für den Betrieb von **DISPA** wird das Testprogramm nicht benötigt.

4.3 Den AVR flashen

Hierzu kann jeder AVR-Brenner verwendet werden, der diesen Prozessor unterstützt; meine Prozessoren brenne ich mit *AVRDude* und *USB AVR Prog* von U.Radig (<http://www.ulrichradig.de/>).

Die Fuses sind wie folgt zu setzen: Ifuse = 0xFF; hfuse = 0xDE; efuse = 0xFD

4.4 Versionsgeschichte

V2		initiale Erstellung aus den Original-Sourcen des FREMO
	30.11.2022	Schaltplan- und Handbuchkorrekturen
	08.12.2022	Handbuchkorrekturen
	25.01.2022	Korrektur des LCD-Anschluss (Schaltplan und Platine): LCD Pin4 (RS) (vorher offen) wird an K1 Pin8 angeschlossen, LCD Pin5 (R/W) wird an GND angeschlossen (vorher an K1 Pin8)
	25.04.2023	Hinweise zu LCD hinzugefügt
	09.12.2023	Kapitel 4 „Software“ aktualisiert
V2.1	17.01.2024	neue Funktion für die FREDI-Diagnose, Kapitel 2.3 hinzugefügt
V2.2	19.01.2024	FREDI-Diagnose auch mit LCD
	11.05.2024	Liste der Bibliotheken korrigiert
V2.3	12.08.2024	Anzeige der aktuellen Lokadresse und Fahrstufe (SV8...10) in der FREDI-Diagnose FREDI-Test (anzeigen von Geschwindigkeit, Fahrtrichtung und Status der Funktionen F0...F16)
	21.08.2024	Kapitel 4 erweitert
V2.4	09.09.2024	Anzeige der Funktionstastenmodi (SV18...34) in der FREDI-Diagnose Hinweise zum Anschluss der Tastatur an die I ² C-OLED-Anzeige hinzugefügt Verwendung eines QR-Code-Readers möglich (bedingt eine kleine Hardwareergänzung)
V2.5	21.09.2024	Bugfix: Begrenzung der Fahrzeugadresse im QR-Code von 1 bis 9999 Neu: Übertragung der Fahrstufen auch ohne (gültige) Fahrzeugadresse Handbuchkorrekturen
V2.6	27.09.2024	Bugfix für das Übertragen von kurzen Adressen aus dem QR-Code an den FREDI
	14.10.2024	redaktionelle Korrekturen
	11.11.2024	redaktionelle Ergänzung in Kapitel 2.1 „Anschluss...“
	03.12.2024	Angaben zu Fuses hinzugefügt
	10.12.2025	Grafik Seite 16 ausgetauscht
V2.7	27.02.2026	Software-Codecleaning, neue QR-Funktion „loco.f255=x“, neue Funktion „notifyThrottleId“

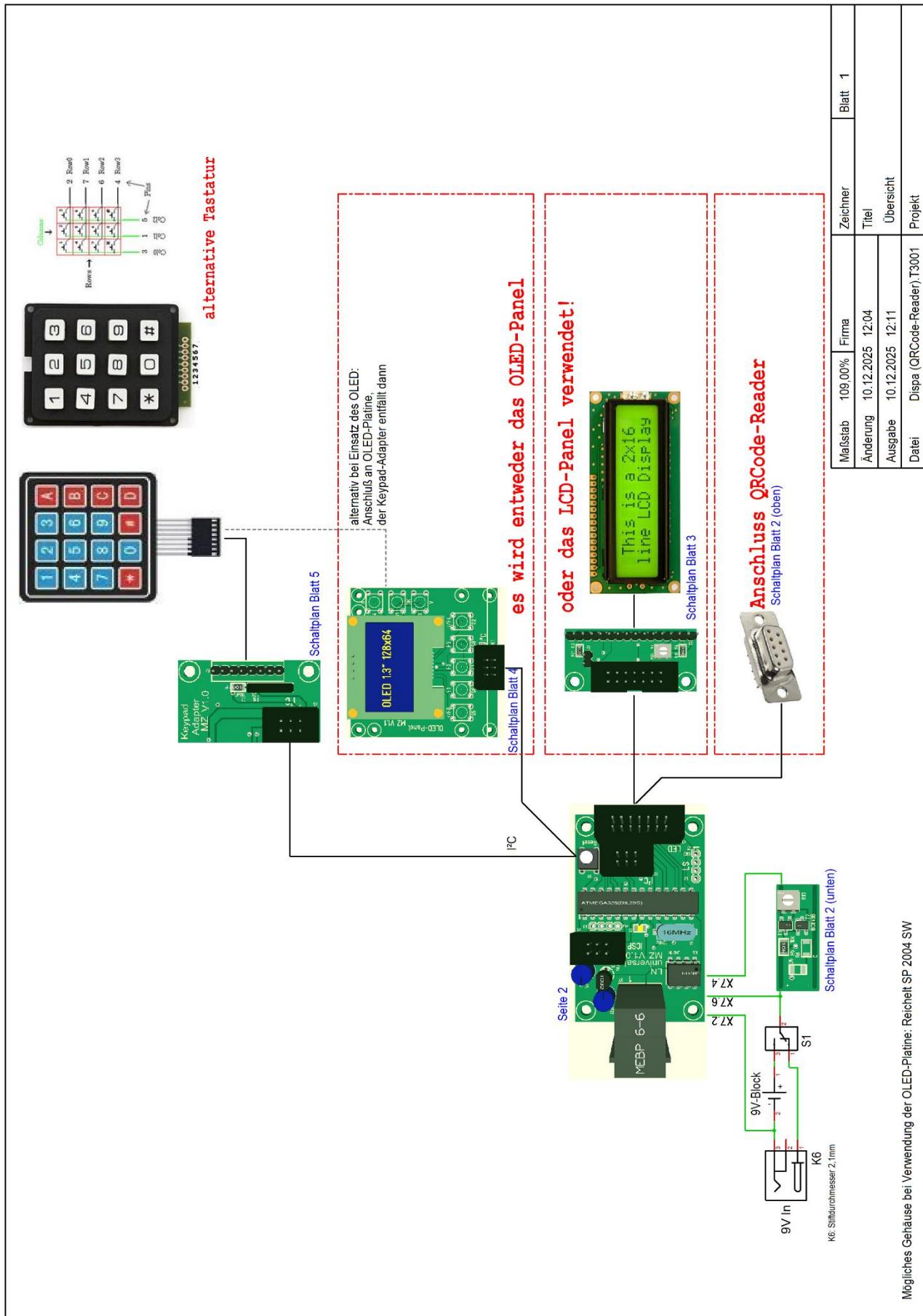
5 Schaltpläne und Stücklisten

Es wurden hier bereits vorhandene Platinen eingesetzt und für DISPA verwendet. Bestellnummern beziehen sich, wenn nicht anders angegeben, auf den Lieferanten Reichelt (<https://reichelt.de>). Es kann nicht sichergestellt werden, dass die in den Stücklisten genannten Bestellnummern aktuell sind, diese können geändert worden bzw. der Artikel nicht mehr lieferbar sein.

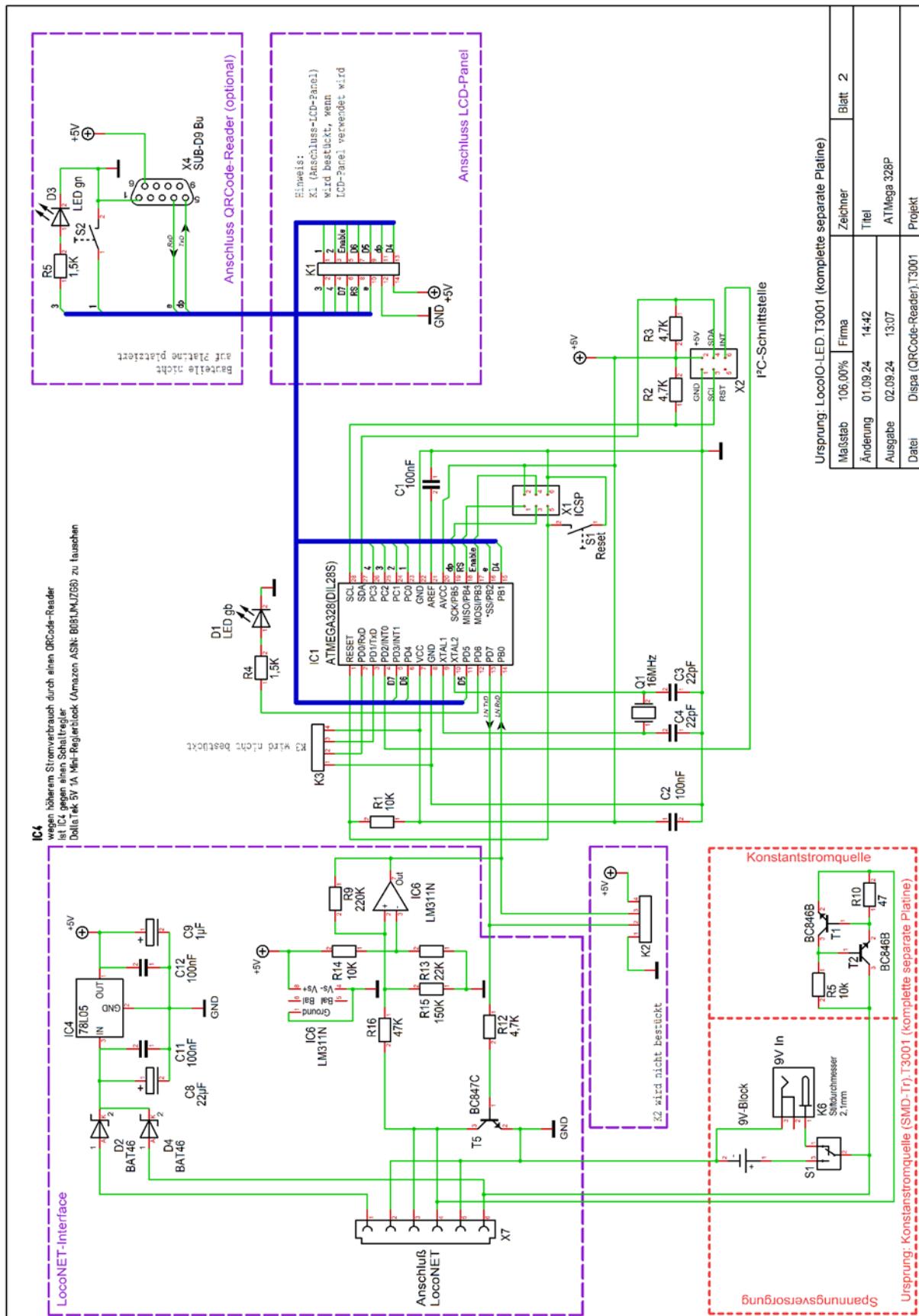
DISPA besteht aus insgesamt vier Komponenten:

- der Prozessorplatine „LN-Universal“
- der Anzeigeeinheit.
Hier kann entweder
 - o eine LCD-Anzeige-Einheit oder
 - o eine I²C-OLED-Anzeige-Einheitverwendet werden.
- der Tastatur-Platine für den Anschluss einer Tastatur
Wird die I²C-OLED-Anzeige verwendet, kann diese für den Anschluss der Tastatur verwendet werden, die separate Tastatur-Platine wird dann nicht benötigt.
- einer 15mA-Konstantstromquelle

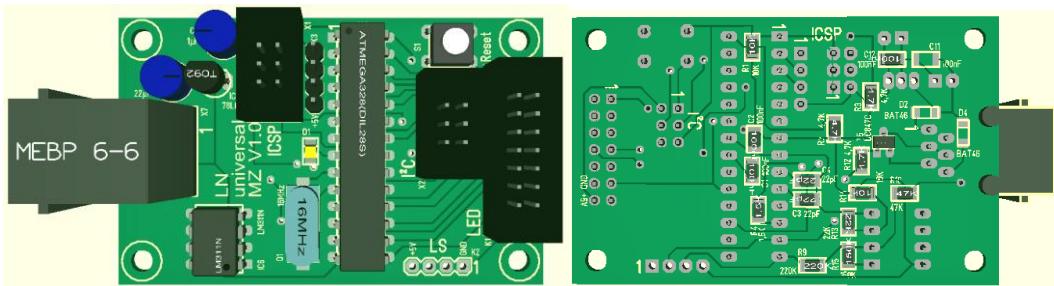
Bei Verwendung der OLED-Anzeige Platinen passt alles in das Gehäuse „SP 2004 SW“.



5.1 Prozessorplatine („LN-Universal“)



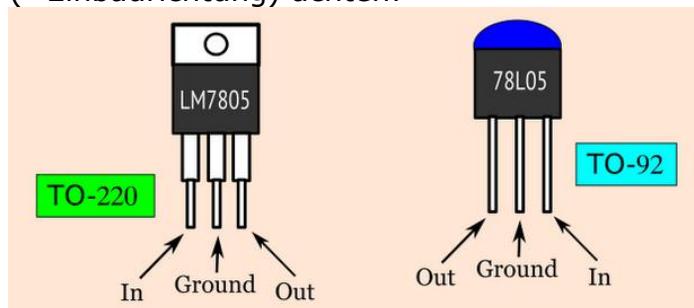
5.1.1 Stückliste Prozessorplatine



Anzahl	Bauteil	Bestellnummer (Reichelt)	Anmerkung
			Platine 64mm * 40mm, doppelseitig
3	C1, C2, C12	X7R-G1206 100N	
2	C3, C4	NPO-G1206 22P	
1	C8	RAD 22/25	RM 2,5; Ø6mm
1	C11	RAD 1/63	RM 2,5; Ø6mm
1	D1	SMD-LED 1206 GE	
2	D2, D4	BAT 46 SMD	
1	IC1	ATMEGA 328P-PU	
1	IC1	GS 28P-S	
1	IC4	µA 78L05	Bei Einsatz der LCD-Anzeige-Einheit und der Verwendung von dessen Hintergrundbeleuchtung ist hier anstelle des µA78L05 ein µA7805 zu verwenden! Dabei auf die Pinbelegung achten, diese unterscheiden sich, siehe Hinweise weiter unten!
1	IC6	LM 311 DIP	
1	IC6	GS 8P	
1	K1	WSL 14G	Werden nur bei Verwendung der LCD-Anzeige-Einheit benötigt und bestückt.
1	K1	PFL 14	
1	K3	SL 1X40G 2,54	Es werden insgesamt vier Stifte benötigt, die Leiste enthält 40 Stifte.
1	Q1	16,00000-HC49-SMD	
2	R1, R14	SMD 1/4W 10K	
3	R2, R3, R12	SMD 1/4W 4,7K	
1	R4	SMD 1/4W 1,5K	
1	R9	SMD 1/4W 220K	
1	R13	SMD 1/4W 22K	
1	R15	SMD 1/4W 150K	
1	R16	SMD 1/4W 47K	
1	S1	TASTER 3301	Kurzhub-Taster flach
1	T5	BC 847C SMD	
2	X1, X2	WSL 6G	
1	X2	PFL 6	
1	X4	SUB-D9-Einbaubuchse	Optional, Anschluss eines QR-Code-Lesers
1	X7	MEBP 6-6S	
1		DEBO BARC SCAN	Optional, QR-Code-Leser, identisch mit dem von mir verwendeten QR-Code-Leser

Hinweise:

- Es sind Verbindungen von der Prozessorplatine zu den einzelnen Platinen herzustellen:
 - o Bei Verwendung der LCD-Anzeige-Einheit:
K1(Prozessorplatine) nach K1(LCD-Anzeige-Einheit) mit einem 14poligen Flachbandkabel.
Dabei auf die Pinbelegung achten, diese unterscheiden sich!
 - o Bei Verwendung der I²C-OLED-Anzeige-Einheit:
 - X2(Prozessorplatine) nach
 - K1(I²C-OLED-Anzeige-Einheit)
 mit einem 6poligen Flachbandkabel.
 - o Der Anschluss der Tastatur erfolgt von
 - X2(Prozessorplatine) nach
 - X5(Tastatur-Platine)
 mit einem 6poligen Flachbandkabel.
Dieses Kabel kann bei Verwendung der I²C-OLED-Anzeige-Einheit mit einem weiteren Stecker PFL 6 bestückt werden (Stichwort „Buskabel“), somit gibt es dann ein Flachbandkabel mit drei PFL-6-Steckern zur Verbindung von X2 mit K1 und X5 (die Verbindungsreihenfolge ist dabei beliebig...)
- Ein (optionaler) QR-Code-Leser wird an X4 angeschlossen, X4 wird über ein Flachbandkabel an K1 angeschlossen: es werden die Anschlüsse +5V, GND, RxD und TxD verwendet:
 - o RxD vom **DISPA** kommt dann an TxD vom QR-Code-Leser und
 - o TxD vom **DISPA** kommt dann an RxD vom QR-Code-Leser
- D3, S2 und R5 (befinden sich nicht auf der Platine¹⁴), sind für spätere Anwendungen vorgesehen und werden aktuell nicht benötigt.
- Bei Einsatz der LCD-Anzeige-Einheit und der Verwendung von dessen Hintergrundbeleuchtung oder dem Einsatz eines QR-Code-Lesers ist wegen des erhöhten Stromverbrauchs anstelle des µA78L05 ein µA7805 (LM7805) zu verwenden, hierbei auf die Anschlussbelegung (=Einbaurichtung) achten!

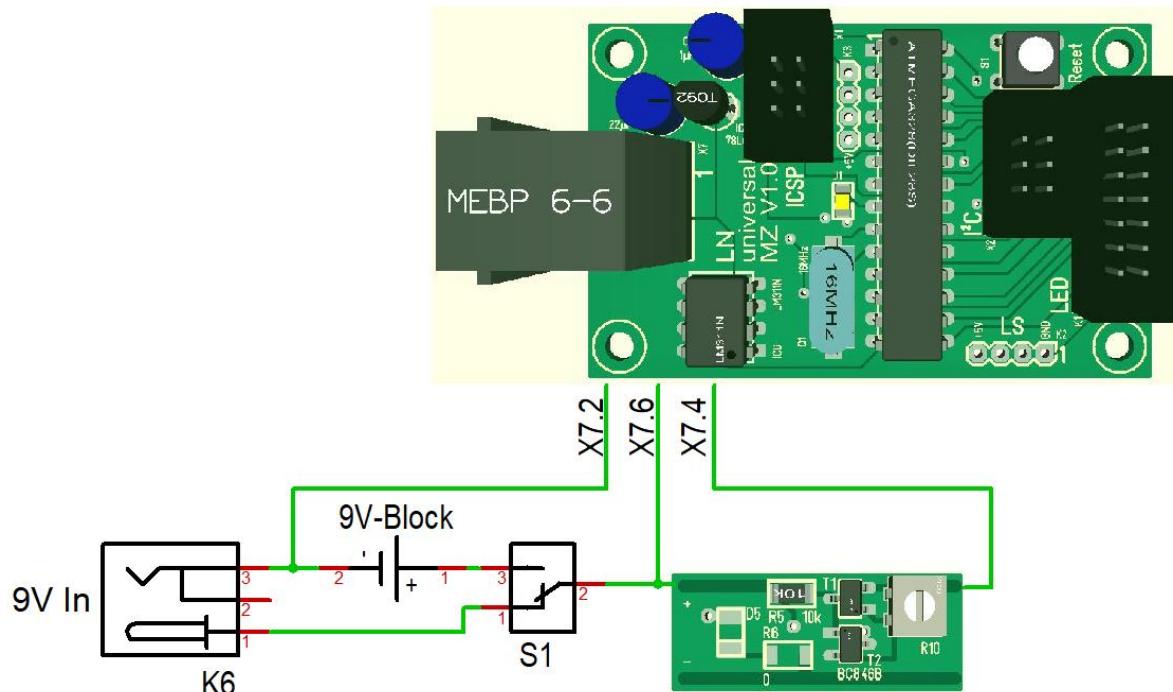


- Alternativ kann auch ein 5V-Schaltregler verwendet werden, z.B.:
 - o DollaTek 5V 1A Mini-Reglerblock (Amazon ASIN: B081JMZG6)
 - o Reichelt: [TSR 1.5-2450E](#) DC/DC-Wandler TSR 1.5E, 1 A, 7-36/5 VDC
 Beide sind pinkompatibel zum LM7805 im TO220-Gehäuse.

¹⁴ ... können aber bei Verwendung meiner OLED-Platine sehr einfach auf dieser angebracht werden:
S2 an Position von S1, D3 und R5 an Position von S4, die Verbindungen von S1 und S4 zu IC1 sind zu trennen

5.2 Konstantstromquelle

Die erforderliche Verdrahtung der Einspeisebuchse (K6), der 9V-Blockbatterie (bzw. der externen Spannungsversorgung), des Ein-Aus-Schalters (S1) und der Konstantstromquelle ergibt sich aus nachfolgender Zeichnung:



K6: Stiftdurchmesser 2,1mm

Anzahl	Bauteil	Bestellnummer (Reichelt)	Anmerkung
1	K6	HEBL 21	Anschluss der externen Spannungsversorgung
1	S1	MS 166	
1		CLIP HQ9V	Anschluss für 9V-Blockbatterie
1		EN POW 9V	9V-Blockbatterie

5.2.1 Stückliste Konstantstromquelle

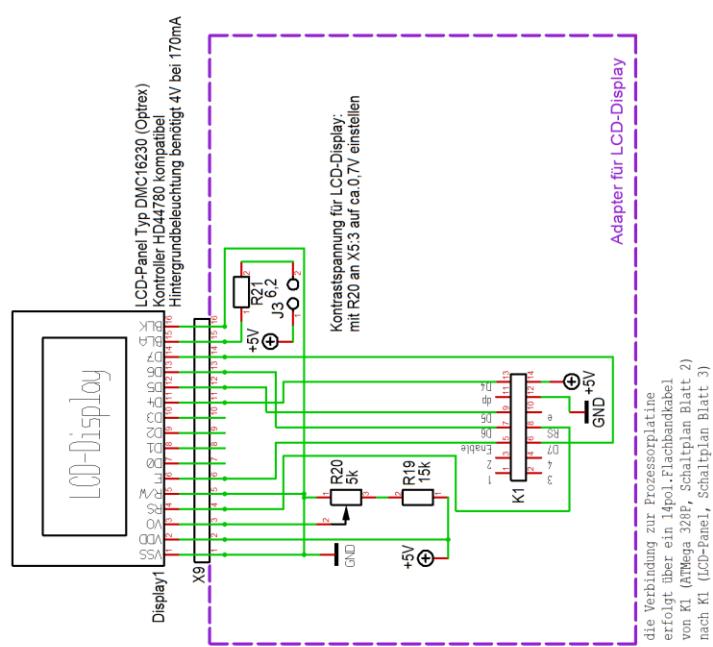
Anzahl	Bauteil	Bestellnummer (Reichelt)	Anmerkung
			Platine 25mm * 10mm, doppelseitig
1	R5	SMD 1/4W 10K	
1	R10	SMD 1/4W 47	Anstelle des Trimmers. Wird ein Trimmer 23A-200 verwendet, so ist der Konstantstrom auf 15mA einzustellen
2	T1, T2	BC 846B SMD	

Hinweis:

- D5 und R6 werden nicht bestückt
 - Der Anschluss erfolgt mit Einzeldrahtverbindungen.
- X7.x bedeutet: Lötanschluss x von Bauteil X7 (MEBP 6-6S)

5.3 LCD-Anzeige-Einheit

es wird entweder das LCD-Panel oder das OLED-Panel verwendet!

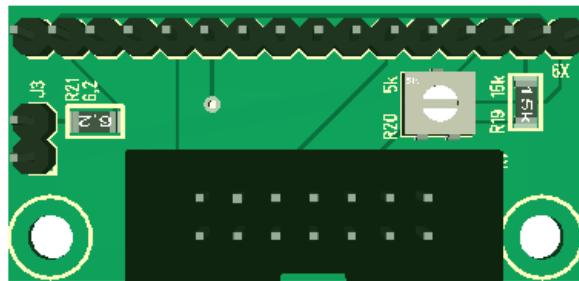


Hinweis: Aufbau / Verwendung wenn LCD-Panel verwendet wird

Ursprung: Franz-Prozessor.T3001 (Ausschnitt)

Maßstab	106.00%	Firma	Zeichner	Blatt 3
Änderung	25.01.23	10:45		
Ausgabe	25.01.23	10:48	Title	Adapter für LCD-Panel
Datum				Projekt
				Dispa T3001

Die LCD-Anzeige-Einheit wird für die Anzeige benötigt.

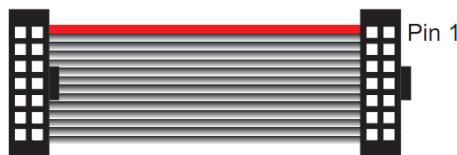


Ein passendes (HD44780-kompatibles) LCD-Modul („LCD 162C LED“) gibt es z.B. bei Reichelt:
<https://www.reichelt.de/lcd-modul-2x16-h-5-6mm-ge-gn-m-bel--lcd-162c-led-p31653.html>

5.3.1 Stückliste LCD-Anzeige-Einheit

Anzahl	Bauteil	Bestellnummer (Reichelt)	Anmerkung
			Platine 20mm * 42mm, doppelseitig, Platine nicht verfügbar!
1	Display1	LCD 162C LED	Anschluss über MPE 094-1-016 und mit SL 1X40G 2,54 sinnvoll
1	J3	SL 1X40W 2,54	Es werden insgesamt 18 Stifte benötigt, eine Leiste enthält 40 Stifte. Auch möglich: SL 1X40G 2,54
1	K7	WSL 14G	Auch möglich: WSL 14W
1	R19	SMD 1/4W 15K	
1	R20	23A-5,0K	Wird zur Kontrasteinstellung der LCD- Anzeige benötigt.
1	R21	SMD 1/4W 6,2	
1	X9	SL 1X40W 2,54	(siehe J3)

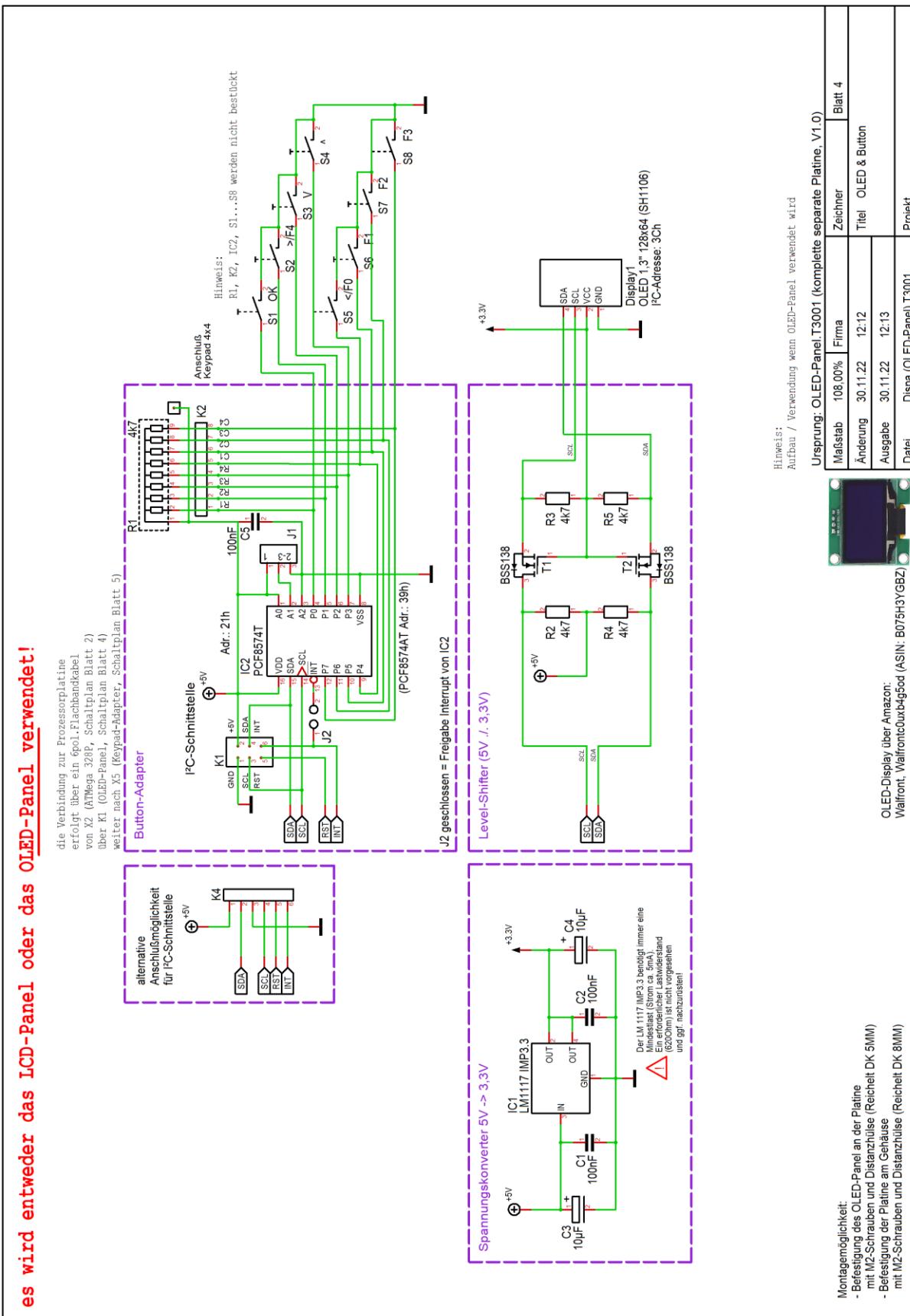
Der Anschluss der LCD-Bedientafel an die Prozessorplatine kann komfortabel über Flachbandkabel erfolgen:



Hinweis:

- Die in einem **Franz** verwendete LCD-Anzeige kann hier als *Display1* verwendet werden.

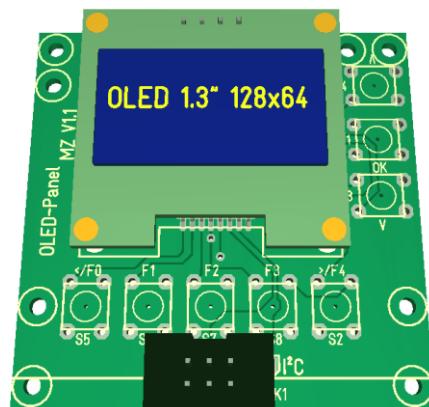
5.4 I²C-OLED-Anzeige-Einheit



Die I²C-OLED-Anzeige-Einheit wird für die Anzeige benötigt.

Vorteil der I²C-OLED-Anzeige-Einheit:

- geringere Größe
- mehr Anzeigemöglichkeiten (Informationen)



5.4.1 Stückliste I²C-OLED-Anzeige-Einheit

Anzahl	Bauteil	Bestellnummer (Reichelt)	Anmerkung
			Platine 54mm * 53mm, doppelseitig
3	C1, C2, C5	X7R-G1206 100N	
2	C3, C4	TAJ 3516 10/16	
1	Display1	OLED 1,3" 128x64 (SH1106)	(z.B. bei Amazon: ASIN: B075H3YGBZ) Auf die Anschlussreihenfolge ist zu achten!
1	IC1	LM1117 IMP3.3	
1	IC2	PCF 8574 T bzw. PCF 8574 AT	Siehe Hinweise! I ² C-Adresse: 0x21 ('T'-Version) bzw. 0x39 ('A'-Version)
1	K1	WSL 6W	Auch möglich: WSL 6G Anschluss I ² C: entweder über K1 oder K4
1	K1	PFL 6	
1	K4	SL 1X40G 2,54	Alternativer I ² C-Anschluss, wenn K1 nicht verwendet wird. Es werden insgesamt sechs Stifte benötigt, eine Leiste enthält 40 Stifte. Auch möglich: SL 1X40W 2,54 Anschluss I ² C: entweder über K4 oder K1
1	R1	SIL 9-8 4,7K	Siehe Hinweise!
4	R2, R3, R4, R5	SMD 1/4W 4,7K	
2	T1, T2	BSS 138 SMD	

Hinweise:

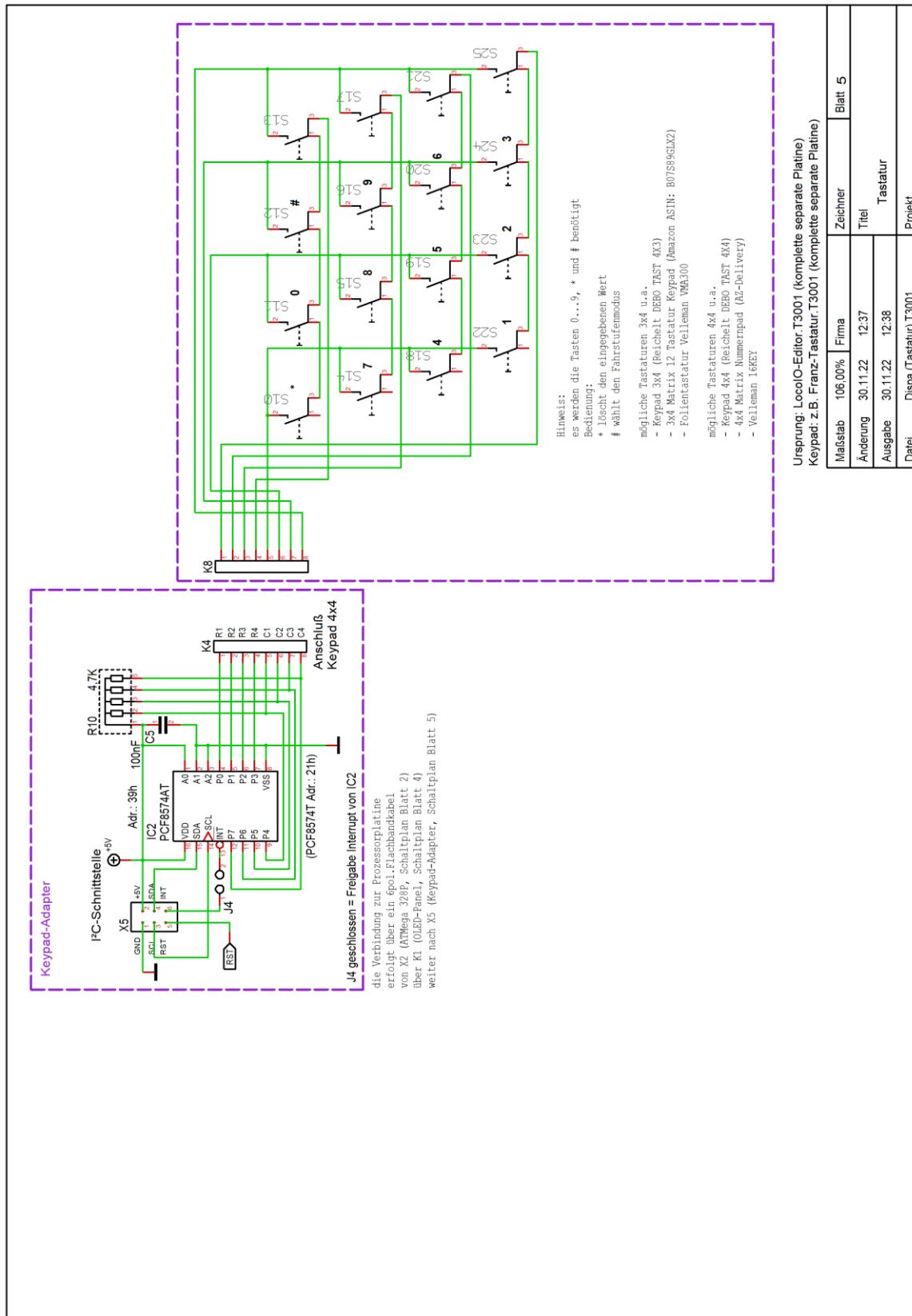
- J1 bleibt unverändert, J2 bleibt offen
- Bei Verwendung des OLED kann der Keypad-Adapter entfallen, wenn
 - o IC2 und das Widerstandsnetzwerk R1 bestückt werden
 - o Bei R1 werden nur 4 Widerstände genutzt:
Der gemeinsame Anschluss wird am quadratischen Lötpunkt angeschlossen, die Anschlüsse der ersten vier Widerstände werden abgeschnitten (oder hochgebogen), die vier unteren Widerstände an den unteren Lötpunkten eingelötet: d.h.: zwischen dem quadratischen Lötpunkt und dem Widerstandsnetzwerk bleiben vier Anschlüsse ungenutzt (nicht angeschlossen).
 - o An K2 wird die Tastatur angeschlossen (siehe auch Hinweise zur Tastatur in [Kapitel 5.5.1 Stückliste Tastatur-Anschluss](#))
- Wird der (zusätzliche) Keypad-Adapter (siehe [Kapitel 5.5 Tastaturanschluss](#)) genutzt, entfällt auf der I²C-OLED-Anzeige die Bestückung von IC2, K2 und dem Widerstandsnetzwerk R1.
- Taster S1...S8 werden in keiner Anwendungsvariante bestückt
- Das Display hat zum Anschluss vier Stifte. Es wird empfohlen, das Display über eine 4polige Buchsenleiste (BL 1X20G 2,54; kürzen) zu verbinden. Das Display selbst kann mit Gewindeschrauben M2 und Abstandshülsen (Höhe 5mm) an der Platine befestigt werden und so bei Bedarf problemlos ausgetauscht werden.
- **Die OLED-Anzeige gibt es mit abweichender Belegung der vier Stifte. Bitte unbedingt auf die Reihenfolge achten und ggf. Verdrahtung anpassen!**
- Anstelle von K1 (WSL 6) kann auch K4 (Stiftleiste 6polig) verwendet werden, dann kann auch die Platine bei Bedarf im unteren Teil um 4mm gekürzt werden.

Der Anschluss der I²C-OLED-Anzeige-Einheit an die Prozessorplatine kann komfortabel über Flachbandkabel erfolgen (siehe auch diesen Hinweis in [Kapitel 5.5.1 Stückliste Prozessorplatine](#)):

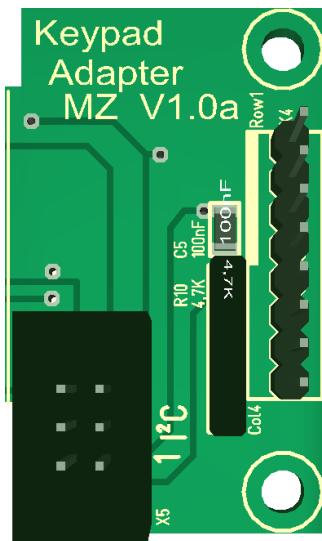


5.5 Tastatur-Anschluss

Bei Verwendung der I²C-OLED-Anzeige kann der Tastatur-Anschluss mit dieser separaten Platine („Keypad-Adapter“) entfallen, wenn stattdessen die Tastatur an die I²C-OLED-Anzeige angeschlossen wird (siehe [Kapitel 5.4 I²C-OLED-Anzeige-Einheit](#)).



5.5.1 Stückliste Tastatur-Anschluss



Anzahl	Bauteil	Bestellnummer (Reichelt)	Anmerkung
			Platine 23mm * 40mm, doppelseitig, V1.0
1	C5	X7R-G1206 100N	
1	IC2	PCF 8574 T bzw. PCF 8574 AT	I ² C-Adresse: 0x21 ('T-Version) bzw. 0x39 ('A'-Version)
1	K4	SL 1X40G 2,54	Es werden insgesamt acht Stifte benötigt, die Leiste enthält 40 Stifte.
1	R10	SIL 5-4 10K	
1	X5	WSL 6G	X5 kann auch mit WSL 6W bestückt werden, wenn die Platine separat verwendet wird
1	X5	PFL 6	
1			Keypad, siehe nachfolgende Hinweise

Hinweise:

- es werden die Tasten 0...9, * und # benötigt
- mögliche Tastaturen 3x4, u.a.
 - o Keypad 3x4 (Reichelt [DEBO TAST 4X3](#))
 - o 3x4 Matrix 12 Tastatur Keypad (Amazon ASIN: B07S89GLX2)
 - o Folientastatur Velleman VMA300
- mögliche Tastaturen 4x4, u.a.
 - o Keypad 4x4 (Reichelt [DEBO TAST 4X4](#))
 - o 4x4 Matrix Nummernpad (AZ-Delivery)
 - o Velleman 16KEY
 - o Tastatur des **Franz**

Der Anschluss des Tastatur-Adapters an die Prozessorplatine kann komfortabel über Flachbandkabel erfolgen (siehe auch diesen Hinweis in [Kapitel 5.5.1 Stückliste Prozessorplatine](#)):



6 Experten-Informationen

6.1 Kommunikation: LocoNET®-Telegramme

Die genaue Kenntnis der verwendeten Telegramme ist nur für Diagnosezwecke erforderlich und dient hier zusätzlich als Dokumentation. Weil – irgendwo muss ich das ja beschreiben...

DISPA empfängt und sendet Telegramme mit verschiedenen OP-Codes:

- Beim Dispatchen
 - o OPC_MOVE_SLOTS 0xBA
 - o OPC_RQ_SL_DATA 0xBB
 - o OPC_LOCO_ADR 0xBF
 - o OPC_SL_RD_DATA 0xE7
 - o OPC_WR_SL_DATA 0xEF
- Im Diagnose-Modus
 - o OPC_PEER_XFER 0xE5
- Im FREDI-Test-Modus werden nachfolgende Telegramme empfangen:
 - o OPC_LOCO_SPD 0xA0
 - o OPC_LOCO_DIRF 0xA1
 - o OPC_LOCO SND 0xA2
 - o OPC_LOCO_F912 0xA3
 - o OPC_UHLI_FUN 0xD4
 - o OPC_IMM_PACKET 0xED

Die Telegramme werden in der LocoNET®-Spezifikation

<https://www.digitrax.com/support/loconet/loconetpersonaledition.pdf> beschrieben,

das Telegramm für OPC_PEER_XFER ist hier

http://embeddedloconet.sourceforge.net/SV_Programming_Messages_v13_PE.pdf beschrieben
und verwendet das **Format 2**, folgt jedoch nicht der Empfehlung **2.2.6) Standard SV/EEPROM Locations** für die Verwendung von SV1...SV3.