

Dipl.-Ing. Michael Zimmermann

Buchenstr. 15

42699 Solingen

☎ 0212 46267

🌐 <https://kruemelsoft.hier-im-netz.de>

✉ BwMichelstadt@t-online.de

Michelstadt (Bw)

Nebenuhr-Statusanzeige

Hardware Version 1

Software Version 1

© 2024 – 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	Nebenuhr-Statusanzeige	3
1.1	Bedeutung der Anzeigen	3
1.2	Anschluss der Nebenuhr-Statusanzeige.....	3
2	Konfiguration.....	4
2.1	Übersicht aller verwendeten CVs	4
2.2	Tabelle der CVs	4
2.3	Inbetriebnahme mit der I ² C-LCD-Bedientafel	5
2.4	Menüstruktur	6
3	Software	7
3.1	HEX-Dateien	7
3.2	Quellcode	7
3.3	Den AVR flashen	7
3.4	Versionsgeschichte	7
4	Schaltpläne und Stücklisten	8
4.1	Nebenuhr-Statusanzeige	8
4.1.1	Übersicht	9
4.1.2	Prozessoreinheit.....	10
4.1.3	Stückliste Prozessoreinheit.....	11
4.2	Nebenuhr-Statusanzeige – WS2812B-Panel	13
4.2.1	Stückliste Nebenuhr-Statusanzeige – WS2812B- Panel	14
4.2.2	Alternative Anzeige durch ein (RGB-)LED-Band	14
4.2.2.1	Transistor-Schaltstufen	15
4.2.2.2	MOSFET-Schaltstufen	15
4.2.2.3	ULN 280X	16
4.3	Spannungsversorgung der Gesamtschaltung.....	16
4.3.1	Verwendung von WS2812B.....	16
4.3.2	Verwendung von (RGB-)LED-Bändern	17
4.4	Zeitanzeige mit einem TM1637 (optional).....	17
4.5	I ² C-LCD-Bedientafel (optional)	18
4.5.1	Stückliste I ² C-LCD-Bedientafel.....	19
5	Experten-Informationen.....	21
5.1	Kommunikation: LocoNET®-Telegramme	21

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 Nebenuhr-Statusanzeige

Zweck: Anzeige des Status der FastClock-Telegramme.

Durch einen Artikel in der FREMO-Zeitschrift HP1 Modellbahn Ausgabe 1/2005 Seite 14 "Philipp Masmeier - Läuft sie oder läuft sie nicht?" bin ich auf die Idee gekommen, eine solche „Uhrenampel“ nachzubauen. Im Artikel beschreibt der Autor Sinn und Zweck einer solchen „Uhrenampel“ – seine Hardware ist hier auf das RUT-System des FREMO ausgelegt.

Eine „Neuentwicklung“ für die Anwendung mit unserem System - die Basis ist hier das FastClock-Telegramm – war daher nur konsequent.

Die Nebenuhr-Statusanzeige arbeitet wie eine FastClock-Tochteruhr (siehe auch <https://github.com/Kruemelbahn/Nebenuhr>), die Uhrzeit kommt dabei immer aus einem FastClock-Telegramm über LocoNET® z.B. von

- RocRail zusammen mit OpenDCC Z1
siehe „Kapitel 2.5.1 Rocrail“ in der [Nebenuhrdokumentation](#)
- JMRI
siehe „Kapitel 2.5.2 JMRI“ in der [Nebenuhrdokumentation](#)
- oder von einer Zentrale, z.B.:
 - [Uhrenzentrale](#)

Anstelle der aktuellen Uhrzeit (Anzeige optional) wird mit diesem Gerät jedoch nur der Status angezeigt.

1.1 Bedeutung der Anzeigen

Über entsprechende Leuchten können drei verschiedene Status angezeigt werden:

- Grün = FastClock läuft im FastClock-Modus (Takt schneller als 1:1)
- Rot = FastClock angehalten (Uhr steht)
- Blau = FastClock läuft im Realtime-Modus (Takt = 1:1)

1.2 Anschluss der Nebenuhr-Statusanzeige

Die Nebenuhr-Statusanzeige wird an das LocoNET® angeschlossen. Für die Spannungsversorgung wird eine externe 12V=-Versorgung benötigt (siehe [Spannungsversorgung der Gesamtschaltung](#)).

2 Konfiguration

2.1 Übersicht aller verwendeten CVs

CV	Bedeutung
1	Eindeutige Identifikationsnummer 1...126, Standard = 1
2	Anzahl der angeschlossenen WS2812B: 1...255, Standard = 16
3	LED-Band-Helligkeit Rot: 0...255, Standard = 255
4	LED-Band-Helligkeit Grün: 0...255, Standard = 255
5	LED-Band-Helligkeit Blau: 0...255, Standard = 255
6	LED-Band-Helligkeit WS2812B: 0...255, Standard = 255
7	Softwareversion, (eigentlich) nur lesbar: Wird hier der Wert 0 eingetragen, so werden alle CVs auf ihren Standardwert zurückgesetzt. Anschließend sind alle CVs auf ihren richtigen Wert zu setzen (=neue Inbetriebnahme!)
8	18 = Kennung „Nebenuhr-Statusanzeige“, nur lesbar
9	Allgemeine Konfigurationen 1: Bit 0 = WS2812B: Farbe Weiß statt Blau verwenden Bit 1 = TM1637 angeschlossen Bit 2 ¹ = FastClock-Telegramm verwenden Bit 3 ² = FastClock läuft nach Initialisierung auch intern weiter Bit 4 ² = FastClock-Telegramme von JMRI ² unterstützen Bit 5 = --- Bit 6 = --- Bit 7 = --- Standard = 00011100 (=28) <i>Wird bei Erst-IBN eingestellt und sollte danach nicht mehr geändert werden.</i>

2.2 Tabelle der CVs

CV	Wert	Aktueller/mein Wert
1	1	
2	16	
3	255	
4	255	
5	255	
6	255	
7	1	
8	18	
9	00011100	

¹ Wird JMRI als Uhrenzentrale verwendet, sind in CV9 die Bits 2, 3 und 4 auf jeden Fall zu setzen.

² JMRI-Telegramme werden ab Software-Version 1 unterstützt

2.3 Inbetriebnahme mit der I²C-LCD-Bedientafel

Nicht jeder, der eine Nebenuhr sein Eigen nennt, braucht auch eine I²C-LCD-Bedientafel – da diese aber ggf. zur Inbetriebnahme oder Diagnose benötigt wird, sollte es wenigstens eine Bedientafel im gesamten System geben...

Übrigens: diese Bedientafel wird auch zur Konfiguration diverser Baugruppen verwendet – kommt also vielfältig zum Einsatz...

Eine Konfiguration vor dem ersten Einsatz der Nebenuhr-Statusanzeige ist normalerweise nicht erforderlich, da hier die Standardeinstellungen ausreichen. Mit Hilfe einer *I²C-LCD-Bedientafel* kann die Nebenuhr-Statusanzeige konfiguriert werden, für den eigentlichen Betrieb ist die *I²C-LCD-Bedientafel* nicht erforderlich.

Am I²C-Anschluss der Nebenuhr-Statusanzeige kann zu jeder Zeit – auch im bereits laufenden Betrieb – die I²C-LCD-Bedientafel angeschlossen bzw. entfernt werden.

Über diese Bedientafel können

- die CVs ausgelesen bzw. geändert werden,
- weitere Diagnosen durchgeführt werden.

Nach dem Anschließen der Bedientafel (bzw. nach dem Einschalten der Nebenuhr mit angeschlossener Bedientafel) erscheint auf dem Display die folgende Information:

Nebenuhr-Status
Version 1

Durch Drücken einer beliebigen Taste gelangt man zur Auswahl der einzelnen Inbetriebnahme- bzw. Diagnosemöglichkeiten.

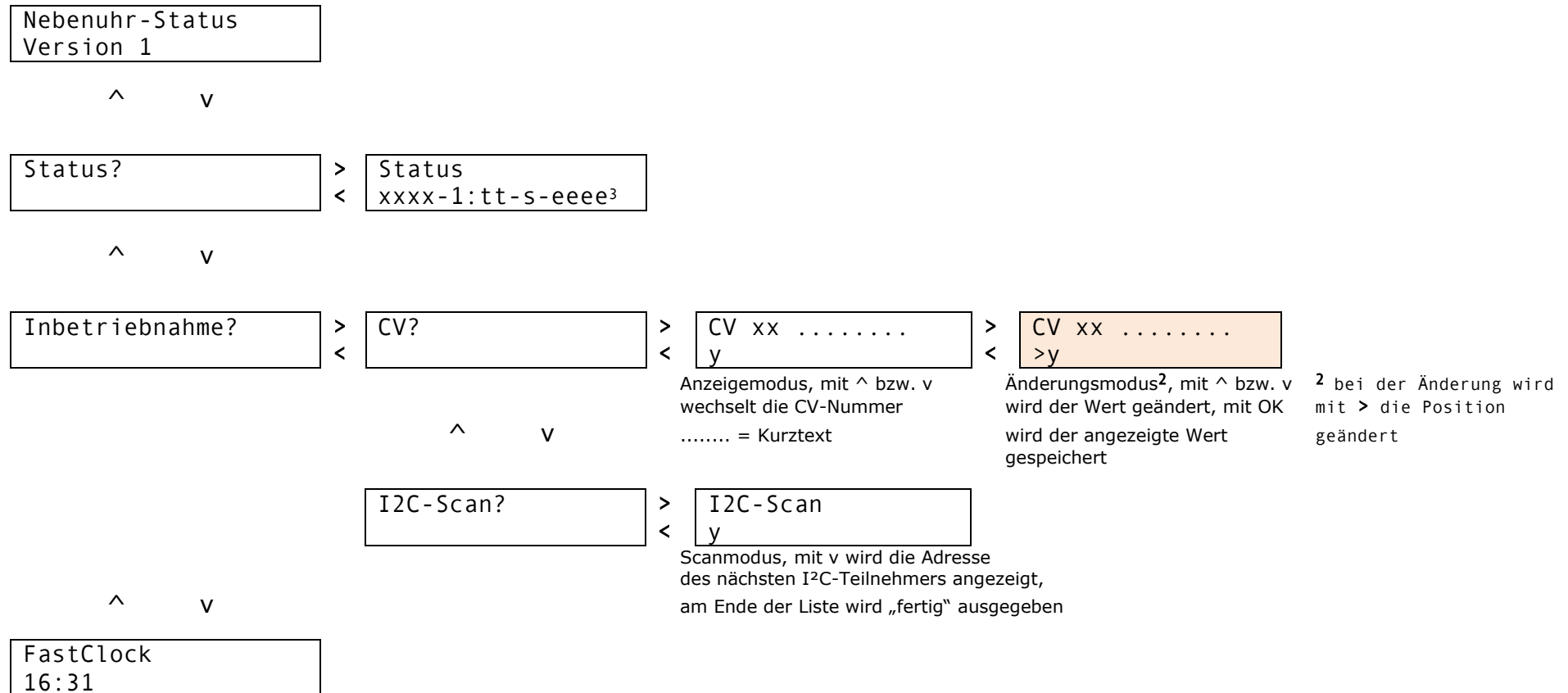
Für die vier kreuzförmig angeordneten Auswahl Tasten gilt:

- < beendet die aktuelle Auswahl, es wird nichts geändert bzw. gespeichert
- > aktiviert diese Auswahl
- ^ wechselt zur vorherigen Auswahl
- v wechselt zur nächsten Auswahl

Die Taste **OK** wird für Bestätigungen oder Speicherfunktionen benötigt.

2.4 Menüstruktur

(nachfolgend dargestellte Menü-Struktur ist für das LCD-Bedientafel gültig)



³ xxxx = Anzahl empfangener FastClock-Telegramme; tt = an der Zentrale eingestellter Teiler (Hinweis: z.B. 10:50 wird als 1:05 angezeigt, die genauere Einstellung der Uhrzentrale kann hier nicht angezeigt werden; s = Sync-Wert (0 oder 1); eeee = Angabe *Even* oder *Odd* des Minutenwertes

3 Software

Der Prozessor benötigt eine Software, um seine Aufgabe zu erfüllen.
Die Software wurde mit der Entwicklungsumgebung für die frei verfügbare [Arduino-IDE](#) erstellt.
Die gesamte Software ist gemäß der zugehörigen Lizenz verfügbar.

3.1 HEX-Dateien

Im GitHub-Repository befindet sich im Ordner „Hexfiles“ (<https://github.com/Kruemelbahn/Nebenuhr-Statusanzeige/tree/main/Hexfiles>) die bereits mit dem Quellcode kompilierte HEX-Datei. Diese Hex-Datei kann mit einem AVR-Programmiergerät auf den Prozessor geladen werden (siehe [Kapitel 3.3 Den AVR flashen](#)).

3.2 Quellcode

Der Quellcode im Hauptverzeichnis (<https://github.com/Kruemelbahn/Nebenuhr-Statusanzeige>) 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.

Die Kompilierung erfolgt für das Board „Arduino UNO“.

Für eine erfolgreiche Kompilierung sind nachfolgende Arduino-Bibliotheken erforderlich:

Arduino-Library	(Link)
Adafruit_NeoPixel	https://github.com/adafruit/Adafruit_NeoPixel
LocoNET®	http://mrrwa.org/loconet-interface/
MemoryFree	http://www.arduino.cc/playground/Code/AvailableMemory
TM1637	https://github.com/avishorp/TM1637

HeartBeat

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

Die Software unterstützt aktuell insgesamt 4 Panel = 16 LEDs vom Typ WS2812B.
Bei anderer LED-Anzahl ist die Software anzupassen:

```
const uint16_t NUMPIXELS = 16 //Anzahl der angeschlossenen WS2812B
```

3.3 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 (<https://www.ulrichradig.de/>).

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

3.4 Versionsgeschichte

V1 29.03.2025 initiale Erstellung

4 Schaltpläne und Stücklisten

Es wurden hier bereits vorhandene Platinen eingesetzt und für die Nebenuhr-Statusanzeige verwendet.

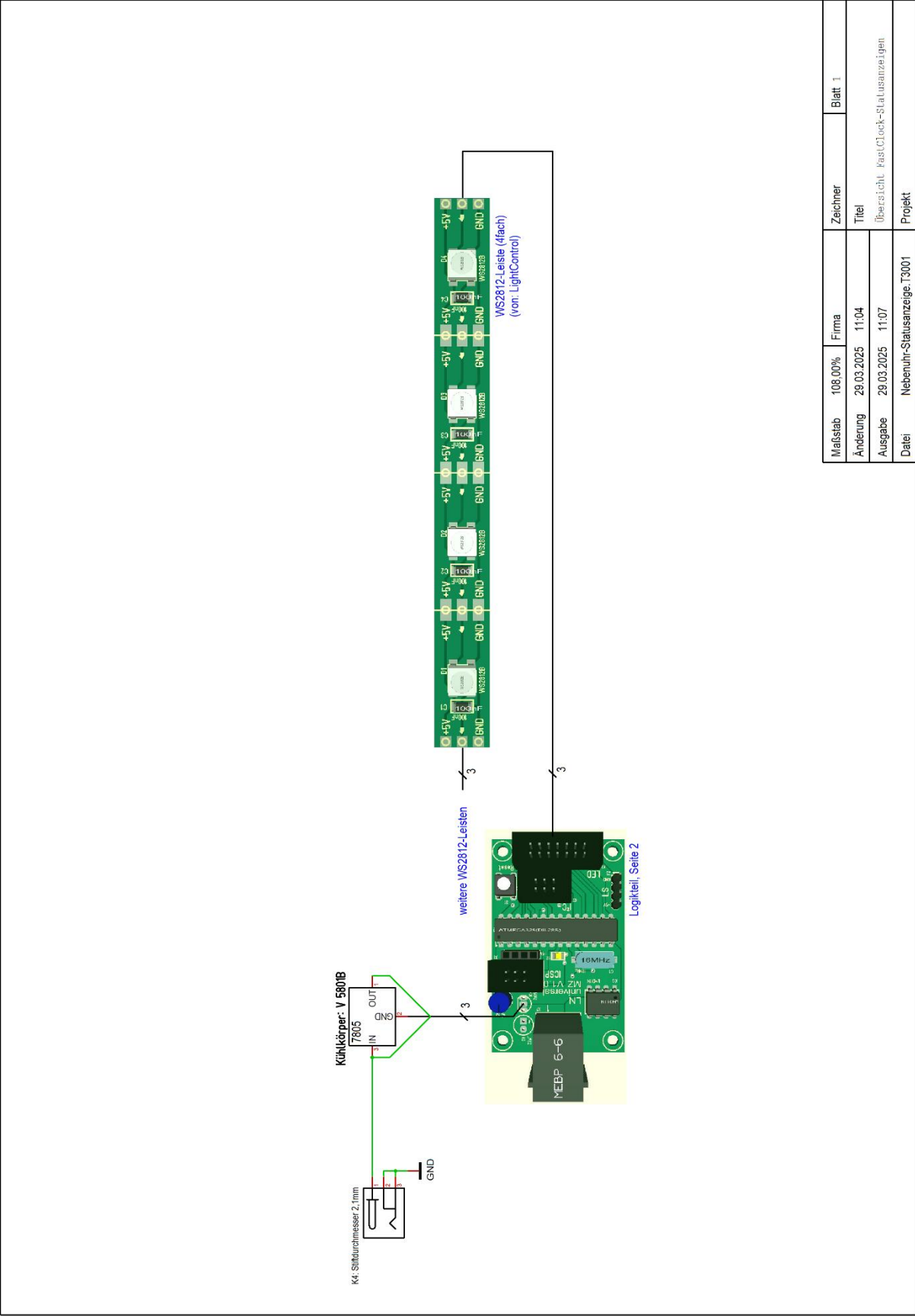
Bestellnummern beziehen sich, wenn nicht anders angegeben, auf den Lieferanten Reichelt (<https://reichelt.de>). Es kann nicht sichergestellt werden, dass die in den Stücklisten genannten Bestellnummern aktuell sind, diese können geändert worden bzw. der Artikel nicht mehr lieferbar sein.

4.1 Nebenuhr-Statusanzeige

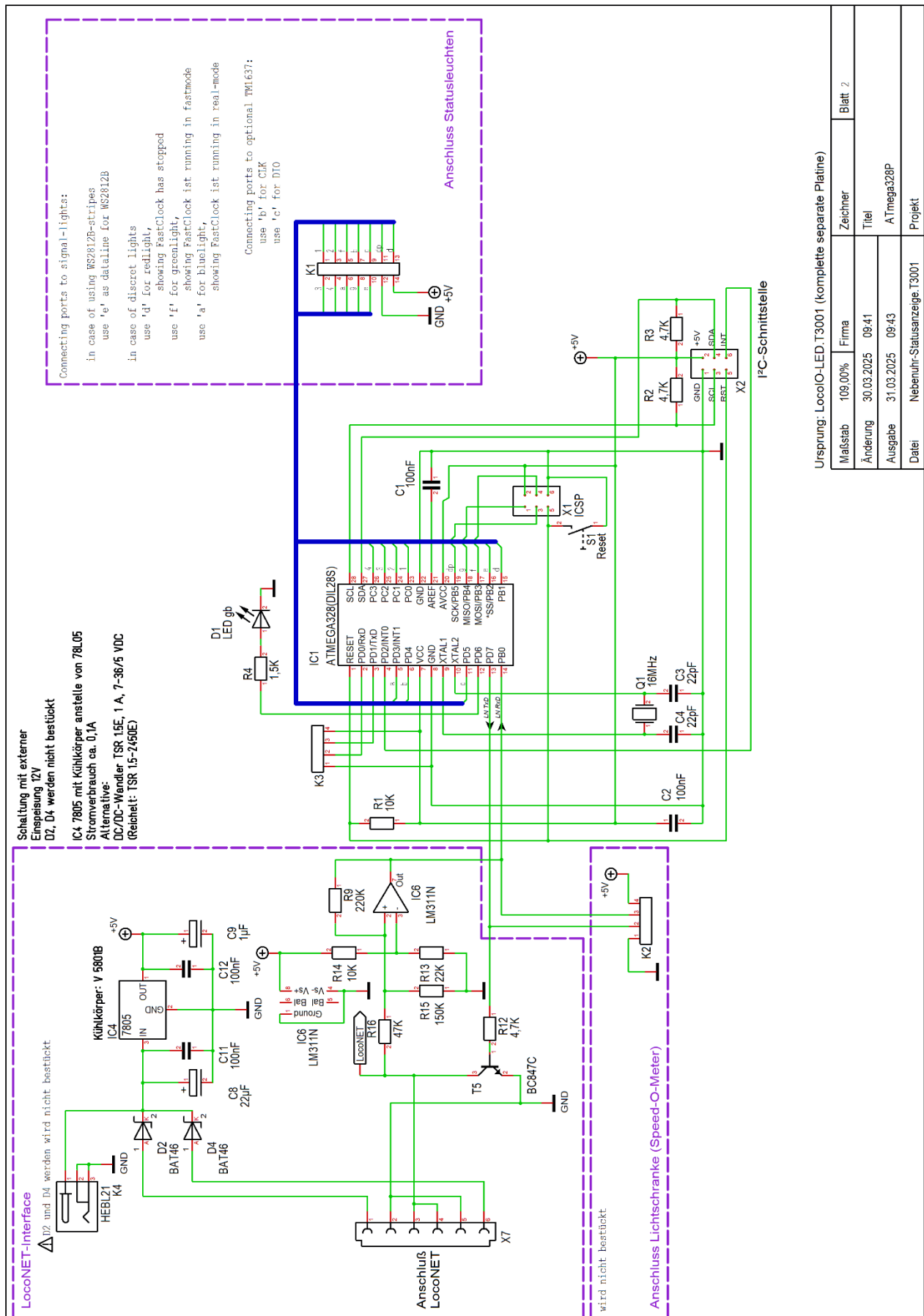
Die Nebenuhr-Statusanzeige besteht aus insgesamt zwei verschiedenen Komponenten:

- der Prozessorplatine „LN-Universal“
- vier WS2812B-Panel

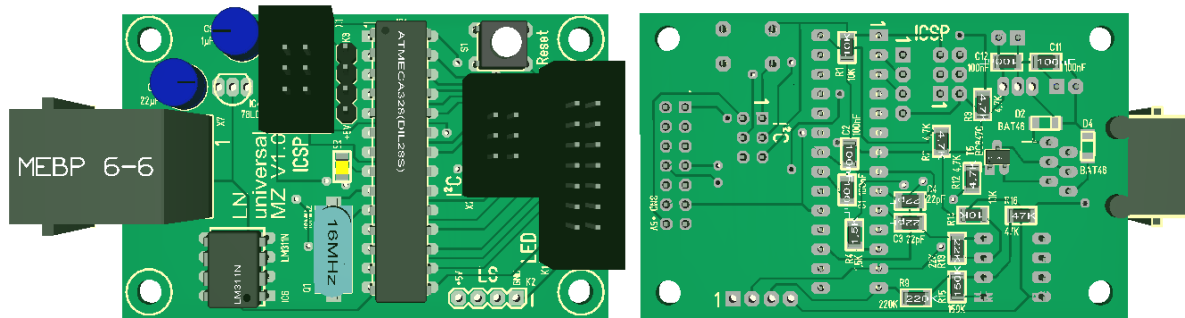
4.1.1 Übersicht



4.1.2 Prozessoreinheit



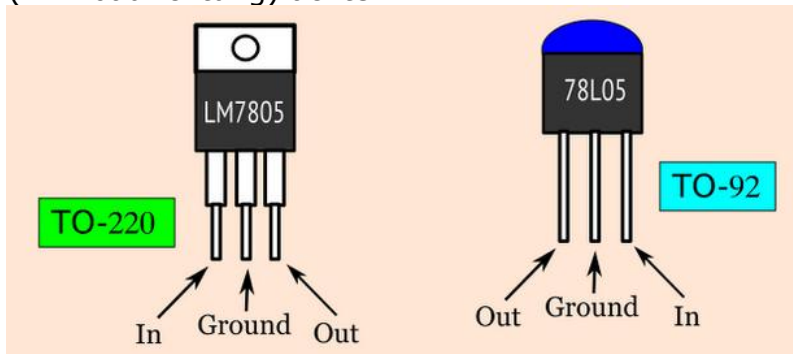
4.1.3 Stückliste Prozessoreinheit



Anzahl	Bauteil	Bestellnummer (Reichelt)	Anmerkung
			Platine 65mm * 40mm, doppelseitig
4	C1, C2, C11, C12	X7R-G1206 100N	
2	C3, C4	NPO-G1206 22P	
1	C8	RAD 22/16	RM 2,54
1	C9	RAD 1/100	RM 2,54
1	D1	SMD-LED 1206 GE	
1	IC1	ATMEGA 328P-PU	
1	IC1	GS 28P-S	
1	IC6	LM 311 P	
1	IC6	GS 8P	
1	K1	WSL 14G	
1	K1	PFL 14	
1	K3	SL 1X40G 2,54	Es werden insgesamt vier Stifte benötigt, die Leiste enthält 40 Stifte.
1	Q1	16,0000-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	
2	X1, X2	PFL 6	
1	X7	MEBP 6-6S	
1	D5	1N 4001	
1	IC4	µA 7805	an Stelle von µA 78L05, extern auf Kühlkörper befestigen
1	IC4	V 5801B	Alternativ auch: TSR 1.5-2450E
1	K4	HEBL 21	Hohlbuchse 2,1mm für 12V-Einspeisung

Hinweise:

- Die externe 12V-Gleichspannungsversorgung wird über die Hohlbuchse (K4, Ø-Mittenstift 2,1mm) eingespeist, der Mittenstift ist der ,+'-Anschluss $\ominus \text{---} \oplus$. Die Hohlbuchse wird über D5 an die Platine angeschlossen.
- anstelle des $\mu A78L05$ ist ein $\mu A7805$ (LM7805) mit Kühlkörper zu verwenden, hierbei ist unbedingt auf die Anschlussbelegung (=Einbaurichtung) achten!



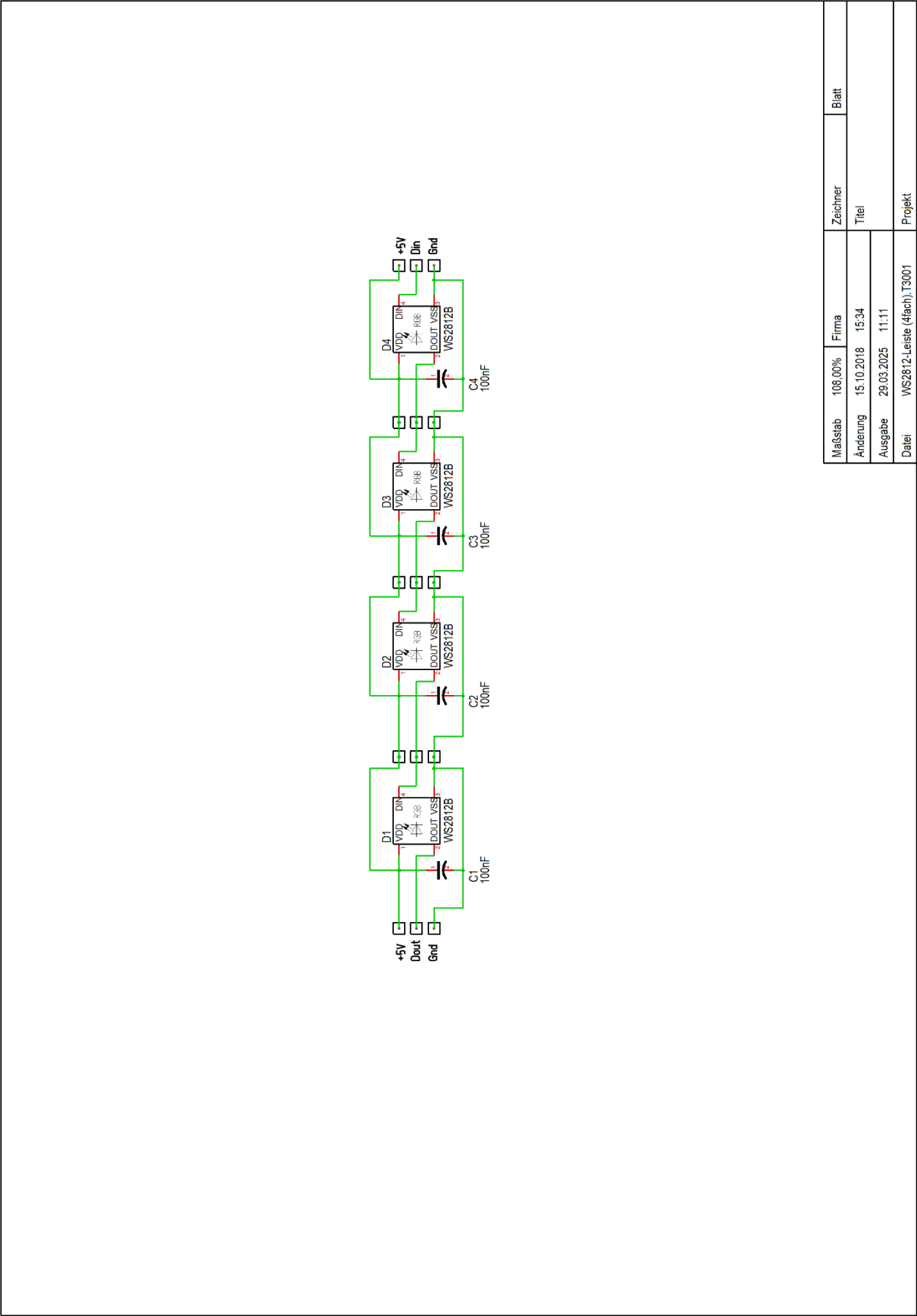
Alternativ kann auch ein 5V-Schaltregler verwendet werden, z.B.:

- o DollaTek 5V 1A Mini-Reglerblock (z.B. Amazon ASIN: B081JMJZG6)
- o Reichelt: TSR 1.5-2450E (DC/DC-Wandler TSR 1.5E, 1 A, 7-36/5 VDC)

Beide sind pin kompatibel zum LM7805 im TO220-Gehäuse.

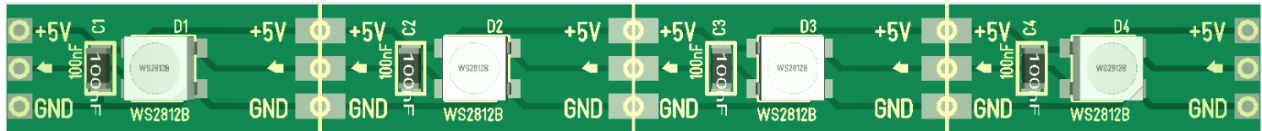
- D2, D4 werden nicht bestückt.
- Es ist eine Verbindung von der Prozessorplatine zur Statusanzeige herzustellen:
 - o WS2812B: siehe [Stückliste Nebenuhr-Statusanzeige – WS2812B-Panel](#)
 - o Andere Anzeigen: siehe [Verwendung von \(RGB-\)LED-Bändern](#)
- Der Stecker X1 (ICSP-Anschluss) wird zum Aufspielen der Software verwendet. Ist dieser nicht bestückt, muss zum Aufspielen der Software jedes Mal der Prozessor (IC1) aus seiner Fassung entfernt und anschließend wieder eingesetzt werden.
- Der Stecker X2 (I²C-Anschluss) wird nicht benötigt. Es empfiehlt sich jedoch die Bestückung, um die I²C-LCD-Bedientafel anschließen zu können, damit bei Bedarf CVs geändert werden können oder im Fehlerfall Diagnoseinformationen ausgelesen werden können.

4.2 Nebenuhr-Statusanzeige – WS2812B-Panel



Maßstab	108,00%	Firma	Zeichner	Blatt
Änderung	15.10.2018	15:34	Titel	
Ausgabe	29.03.2025	11:11		
Datei	WS2812-Leiste (4fach).T3001			
Projekt				

4.2.1 Stückliste Nebenuhr-Statusanzeige – WS2812B- Panel



Anzahl	Bauteil	Bestellnummer (Reichelt)	Anmerkung
			Platine 100mm * 10mm, einseitig
4	D1...D4		WS2812B Bauform 5050, (3,3...)5V, mit integriertem Controller (als einzelne LED nicht bei Reichelt erhältlich...)
4	C1, C2, C11, C12	X7R-G1206 100N	

(diese Tabelle zeigt die Stückliste für **ein** Panel und ist mit der Anzahl der verwendeten Panels zu multiplizieren!)

Der Anschluss des WS2812B-Panels an die Prozessorplatine erfolgt über K1 auf der Prozessorplatine (im obigen Panel-Bild auf der rechten Seite):

- K1 Pin 10 = Datenleitung = ◀
- K1 Pin 12 = GND (Masse / 0V)
- K1 Pin 14 = +5V

An der (im obigen Panel-Bild) linken Seite des WS2812B-Panels können weitere Panel angeschlossen werden.

Das Panel kann an den markierten Stellen nach Bedarf gekürzt werden.

Die Software unterstützt aktuell insgesamt 4 Panel = 16 LEDs vom Typ WS2812B. Bei anderer LED-Anzahl ist die Software anzupassen, siehe hier: [Quellcode](#).

4.2.2 Alternative Anzeige durch ein (RGB-)LED-Band

Alternativ zum WS2812B-Panel werden auch (RGB-)LED-Bänder (bzw. RGBW-LED-Bänder) unterstützt.

Anstelle der Farbe Blau kann auch die Farbe Weiß verwendet werden, ins besonders beim Einsatz von LED-Bändern mit Einzelfarben oder einem RGBW-LED-Band.

Der Anschluss eines (RGB-)LED-Band an die Prozessorplatine erfolgt über K1 auf der Prozessorplatine:

- K1 Pin 5 = Grün = FastClock läuft im FastClock-Modus (Takt schneller als 1:1)
- K1 Pin 6 = Blau = FastClock läuft im Realtime-Modus (Takt = 1:1)
- K1 Pin 12 = GND (Masse / 0V)
- K1 Pin 13 = Rot = FastClock angehalten (Uhr steht)
- K1 Pin 14 = +5V

Da der verwendete Prozessor nicht in der Lage ist,

- Ströme von mehr als 20mA je Anschluss auszugeben oder
- Spannungen von mehr als 5V zu schalten,

sind unbedingt Verstärkerstufen für LED-Bänder einzusetzen (siehe Vorschläge in den nachfolgenden Abschnitten)⁴.

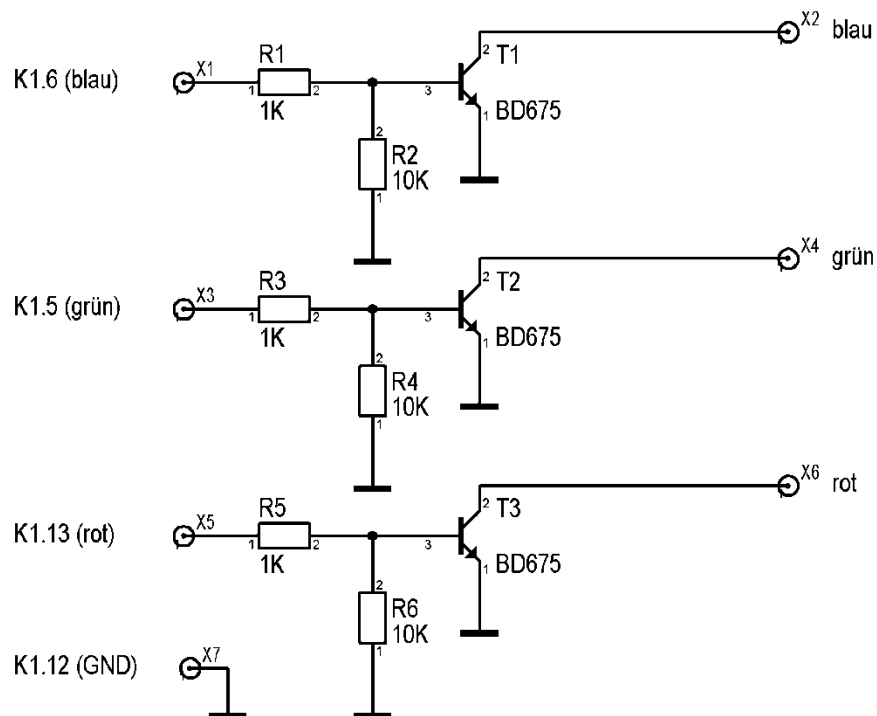
⁴ Daten meines RGB-LED-Bandes (V-TAC: [LED Strip SMD5050 30 LEDs RGB IP20](#))

Länge 5m = 50 Abschnitte à 100mm, 3 LED je Abschnitt = 30 LEDs/m = 150 LEDs

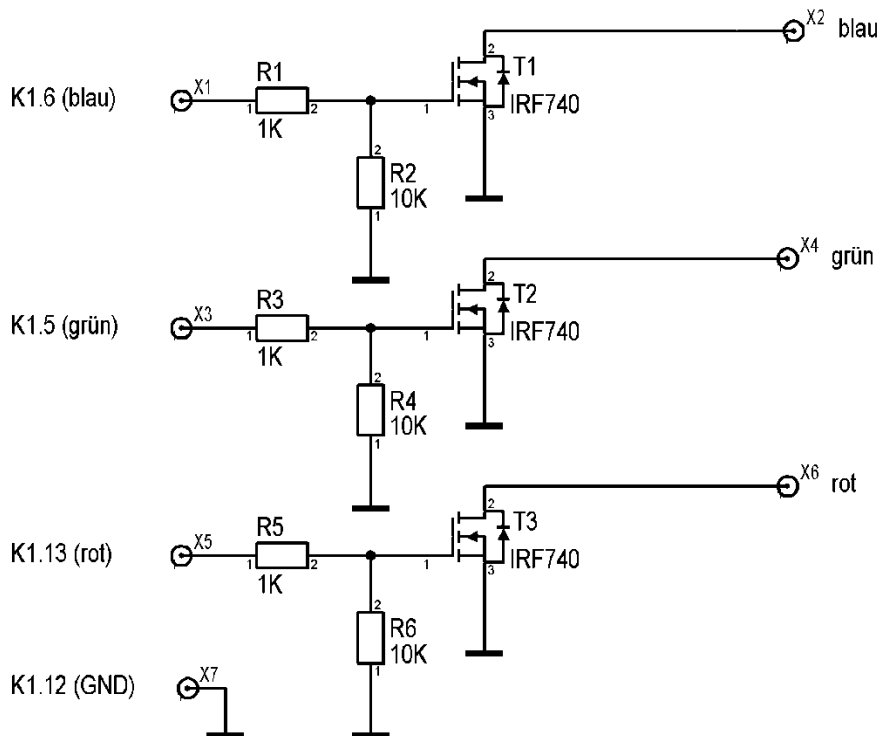
4,8W/m = 24W/5m = 0,16W/LED = 13mA/LED bei 12V-Versorgungsspannung

Gemessen wurden bei Anschluss aller 3 Farben (ergibt dann zusammen weiß) ca. 1A bei 12V = ca. 7mA/LED

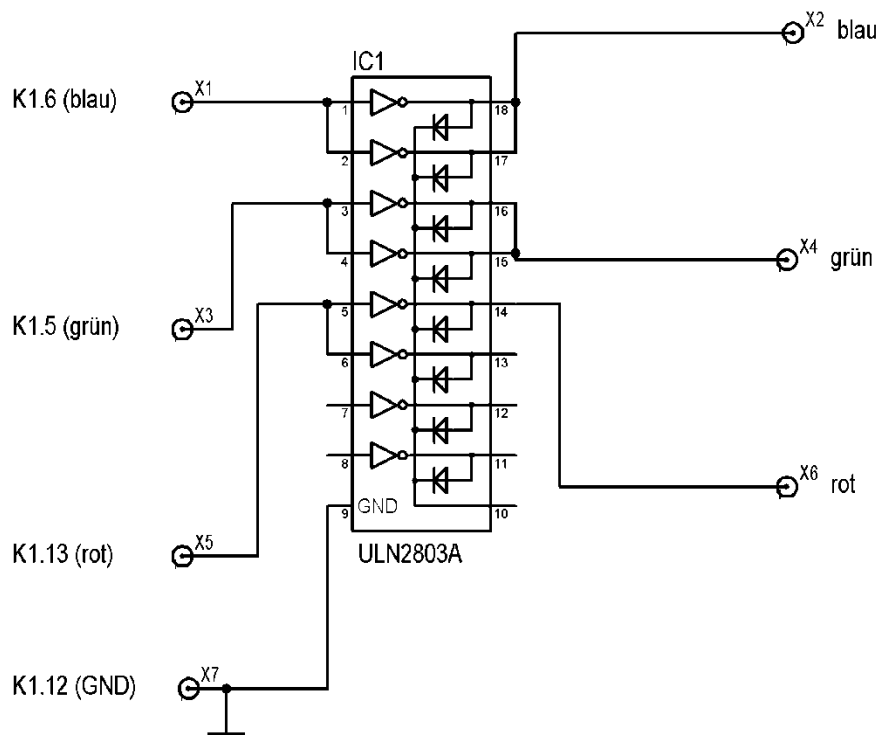
4.2.2.1 Transistor-Schaltstufen



4.2.2.2 MOSFET-Schaltstufen



4.2.2.3 ULN 280X



4.3 Spannungsversorgung der Gesamtschaltung

Grundsätzlich ist die Versorgung über den LocoNET®-Anschluss Aufgrund des hohen Strombedarfs nicht möglich.

4.3.1 Verwendung von WS2812B

eine WS2812B benötigt⁵

- 1,5mA für die Basisversorgung
- und zusätzlich 10,5mA für die volle Lichtstärke **einer** Farbe

Daraus ergibt sich (aufgerundet):

- ein Bedarf von 12mA je LED je Farbe

d.h. ein WS2812B-Panel mit 4 Leds benötigt:

- 48mA für die Farbe Rot bzw. Grün bzw. Blau
- 132mA für die Farbe Weiß (= Rot + Grün + Blau)

und die Formel:

$$I_{ges} = \text{Anzahl der LEDs} * (10,5\text{mA} * \text{Anzahl Farben} + 1,5\text{mA})$$

Versorgung durch 5V anstelle von 12V

Wird anstelle der 12V-Versorgung eine 5V-Versorgung verwendet, so kann

- o IC4 (7805) mit Kühlkörper

entfallen, am IC4 ist dann Anschluss 1 mit Anschluss 3 zu brücken. Eine Diode als Verpolungsschutz in der Einspeisung sollte nicht fehlen.

Für den Anschluss des 5V-Netzteils kann ebenfalls eine Hohlbuchse verwendet werden. Um eine fehlerhafte Einspeisung mit 12V zu verhindern, wird hier eine

⁵ gemessen...

Hohlbuchse mit Ø-Mittenstift **2,5mm** empfohlen (HEBL 25)⁶. Auch hier ist der Mittenstift der ‚+‘-Anschluss $\ominus \text{---} \text{---} \oplus$. Die Hohlbuchse wird über D5 an die Platine angeschlossen.

4.3.2 Verwendung von (RGB-)LED-Bändern

Da diese Anzeigen benötigen oftmals eine Versorgungsspannung von 12V. Die oben vorgeschlagenen Schaltstufen sind für die auftretenden Ströme anzupassen. Da es eine Vielzahl von (RGB-)LED-Bändern gibt und die Anzahl der tatsächlich verwendeten LEDs unbekannt ist, sind für die Dimensionierung die Datenblätter zu Rate zu ziehen. Gegebenenfalls sind auch Kühlkörper erforderlich.

Versorgung durch 5V anstelle von 12V

Ist bei 12V-LED-Bändern nicht sinnvoll.

4.4 Zeitanzeige mit einem TM1637 (optional)



Beschreibung:

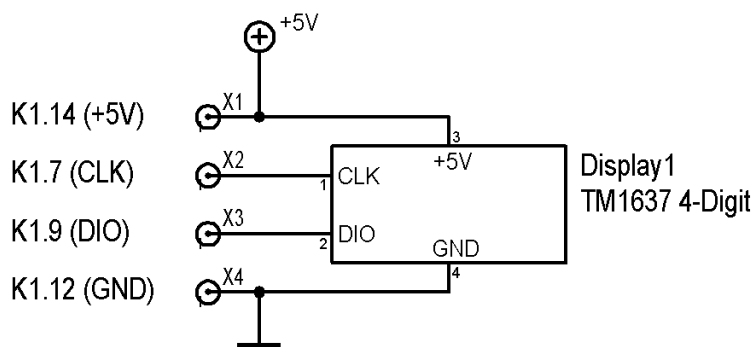
Ein Modul mit 4 Stellen aufgeteilt in eine 7-Segment-Anzeige. Der Treiber IC ist TM1637. Es kann zur Anzeige von Ziffern, Buchstaben etc. verwendet werden.

Details:

- 4 Stellen rot alpha numerische Anzeige
- 8 einstellbare Leucht-Level
- Eingangsspannung: 3.3-5V DC
- Stromverbrauch (bei 5V): 30-80mA
- Interface level kann bei 5V oder 3.3V liegen
- Abmessungen: ca. 42x24x12mm
- Gewicht: ca. 8g

Der Anschluss des TM1637⁷ an die Prozessorplatine erfolgt über K1 auf der Prozessorplatine:

- K1 Pin 7 = CLK-Signal zum TM1637
- K1 Pin 9 = DIO-Signal zum TM1637
- K1 Pin 12 = GND (Masse / 0V)
- K1 Pin 14 = +5V



Die Verwendung dieser Anzeige muss über CV9 Bit 1 aktiviert werden.

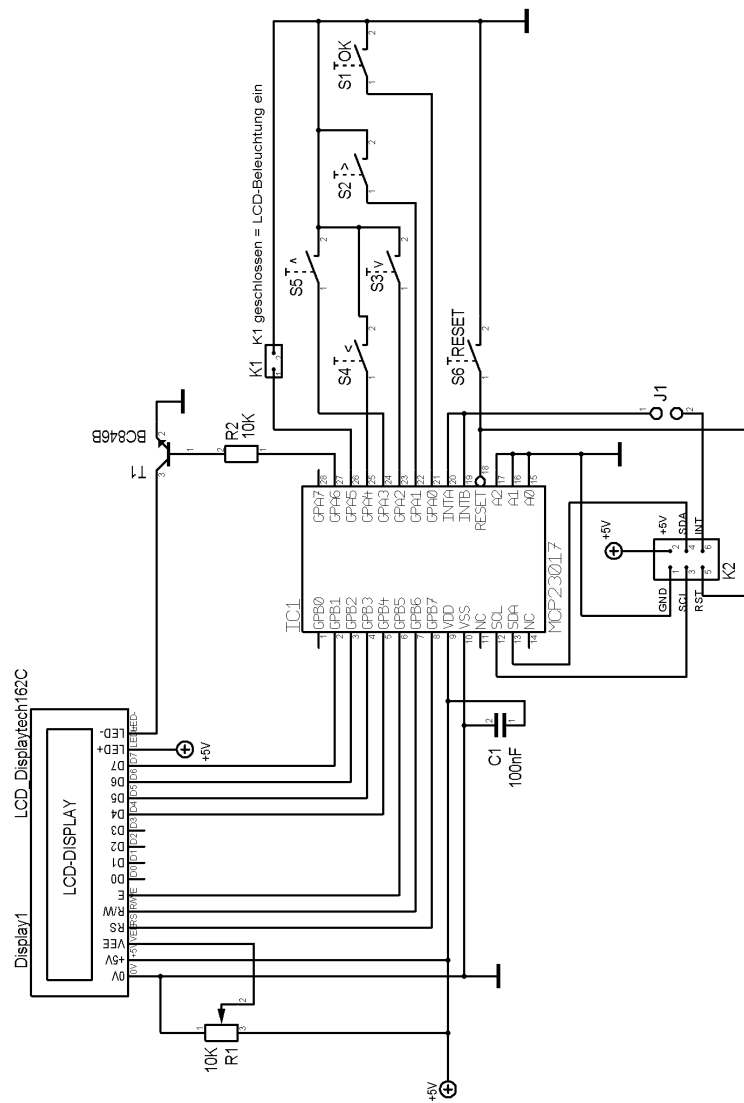
Versorgung durch 5V anstelle von 12V

- ohne Statusanzeige oder bei Verwendung von WS2812B für die Statusanzeige: siehe [Verwendung WS2812B](#)
- bei Verwendung von (RGB-)LED-Bändern nicht sinnvoll

⁶ dann passt der dünne 2,1mm-Hohlstecker für 12V nicht in die dicke 2,5mm-Hohlbuchse für 5V

⁷ Erhältlich z.B. bei [AZ-Delivery](#)

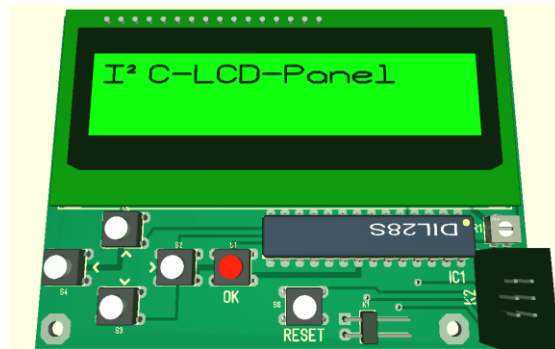
4.5 I²C-LCD-Bedientafel (optional)



	Masstab	107,00%	Firma	Zeichner	Blatt
	Änderung	09.06.17	16:17	Titel	
	Ausgabe	09.06.17	16:20		
	Datum	I2C-LCD-Anzeige (SMD), T3001			
	Projekt				

Nicht jeder, der eine Nebenuhr sein Eigen nennt, braucht auch eine I²C-LCD-Bedientafel – da diese aber ggf. zur Inbetriebnahme oder Diagnose benötigt wird, sollte es wenigstens eine Bedientafel im gesamten System geben...

Übrigens: diese Bedientafel wird auch zur Konfiguration diverser Baugruppen verwendet – kommt also vielfältig zum Einsatz...



Die komplette LCD-Anzeigeeinheit gibt es z.B. bei Reichelt:

<https://www.reichelt.de/de/de/arduino-shield-display-lcd-kit-16x2-blau-weiss-arduino-shd-lcd-p159967.html>
(ARDUINO SHD LCD)

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>

4.5.1 Stückliste I²C-LCD-Bedientafel

Anzahl	Bauteil	Bestellnummer (Reichelt)	Anmerkung
			Platine 84mm * 60mm, doppelseitig
1	C1	X7R-G1206 100N	
1	Display1	LCD 162C LED	Anschluss über MPE 094-1-016 und mit SL 1X40G 2,54 sinnvoll
1	IC1	MCP 23017-E/SP	I ² C-Adresse: 0x20
1	IC1	GS 28P-S	
1	K1	SL 1X40G 2,54	Es werden insgesamt zwei Stifte benötigt, eine Leiste enthält 40 Stifte. Auch möglich: SL 1X40W 2,54
1	K2	WSL 6G	Auch möglich: WSL 6W
1	R1	23A-10K	
1	R2	SMD 1/4W 10K	
6	S1...S6	TASTER 3301	Kurzhubtaster
1	T1	BC 847C SMD	

Hinweise:

- J1 bleibt offen
- An K1 kann ein Schalter (Schließer) zur Steuerung der LCD-Beleuchtung angeschlossen werden.
- Es wird empfohlen, das Display mit 16 Stiften aus SL 1X40G 2,54 zu bestücken, auf der Platine wird dann als Gegenstück die Buchsenleiste MPE 094-1-016 (beides nicht in der Stückliste oben enthalten) verwendet. Das Display selbst kann mit Gewindeschrauben M2 an der Platine befestigt werden und so bei Bedarf problemlos ausgetauscht werden.
- *Für die Verwendung des AdaFruit-RGB-LCD-Shields (I²C-Adresse: 0x20) gilt:*
 - o *Das Shield ist zur direkten Verwendung mit einem Arduino vorgesehen: der I²C-Anschluss (K2) ist mit Einzeldrähten herzustellen (siehe die zugehörige Anleitung).*
 - o *Das Shield besitzt keinen Anschluss K1: ein Schalter bzw. Drahtbrücke ist direkt zwischen Pin 26 des MCP23017 und GND anzuschließen.*

Meine I²C-LCD-Anzeige-Einheit habe ich in ein Gehäuse aus zwei Halbschalen (Bestellnummer bei Reichelt: SD 10 SW HALB) mit einem seitlichen SUB-D9-Stecker für den Anschluss an den I²C-Bus montiert.

Die Anzeigeeinheit ist auf diese Art universell auch für andere Anwendungen (Relaisblock, Stellwerk, Intervaluino, AVR-Sound, LocoIO-SV-Editor) einsetzbar.



Der Anschluss der I²C-Bedientafel an das FastClock-Modul kann komfortabel über Flachbandkabel erfolgen.

In meinem Fall habe ich den I²C-Anschluss mit einem SUB-D9-Stecker über ein Stück Flachbandkabel verbunden:



Das Anzeige-Modul ist so über den SUB-D9-Stecker an andere Geräte (z.B. mein Stellwerk oder meinen Intervaluino) angeschlossen werden.

5 Experten-Informationen

5.1 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...

LocoNET®-FastClock empfängt und sendet Telegramme mit den OP-Codes

- OPC_PEER_XFER 0xE5
- OPC_SL_RD_DATA 0xE7
- OPC_WR_SL_DATA 0xEF

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,

verwendet das ‚Format 2‘ und folgt nicht der Empfehlung 2.2.6) [Standard SV/EEPROM Locations](#) für die Verwendung von SV1...SV3.

Die Unterstützung der OPC_PEER_XFER-Telegramme ermöglicht es, die CVs auch mit dem Tool „DecoderPro®“ von JMRI (<https://www.jmri.org/>) auslesen und einstellen zu können, passende XML-Dateien und eine Anleitung sind verfügbar.