

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

Michelstadt (Bw)

OpenDCC - Zusammenfassung zum Bau -

Locobuffer II - Einstellungen -

Inhaltsverzeichnis

Unterlagen im WWW	2
Mein Aufbau – was habe ich weggelassen – was wurde geändert	2
Meine Inbetriebnahme.....	3
FTDI	3
FTDI-Treiber	3
Besonderheiten unter Win10 – meine Vorgehensweise	4
Fuses und Bootloader	5
OpenDCC-Zentrale	5
RocRail-Einstellungen (PC).....	5
OpenDCC-Konfiguration anpassen (PC)	7
LocoNET® - LocoBuffer II (Hans Deloof).....	8
RocRail-Einstellungen (PC).....	8
JMRI (DecoderPRO)-Einstellungen (PC)	9
Zusammenschaltung	10

Für Hinweise auf Fehler oder Ergänzungen ist der Autor dankbar.

Die Nennung von Marken- und Firmennamen geschieht in rein privater und nichtgewerblicher Nutzung und ohne Rücksicht auf bestehende Schutzrechte.

*Diese Zusammenstellung wurde nach bestem Wissen
und ohne Vollständigkeitsgarantie in der Hoffnung erstellt, dass sie nützlich ist.
Wenn sie nicht nützlich ist – dann eben nicht.*

Unterlagen im WWW

Im WWW gibt es zahlreiche Unterlagen, die sich mit dem Zusammenbau und der Inbetriebnahme des OpenDCC-Zentrale Z1 befassen.

Die nachfolgende Aufstellung erhebt keinen Anspruch auf Vollständigkeit:

https://opendcc.de/elektronik/opendcc/download.html	Downloadseite zum Projekt
https://www.opendcc.de/elektronik/opendcc/opendcc_hw_bau.html	Originaler Baubericht
https://www.li-pro.net/opendcc-zentrale.phtml	Verwendung und Anbau der Kühlkörper
https://modellbahnfreaks.de/dcc/zentrale.php	Baubericht

Mein Aufbau – was habe ich weggelassen – was wurde geändert

Schnittstellen:

Kein PonyProg es entfallen: D2, D6, D7, R27, R29, R30, X11
 Kein RS232 es entfallen: C6, C18, C19, C20, C21, IC2, X7
 Kein DMX es entfallen: IC3, X3

Ausbaustufe: IC5: ATmega 644P, V0.23.9 mit P50x-Emulation

Strombegrenzung: R7 = 11k → 2,0A
 R12 = 39k → 0,5A

Jumper: JP1, JP2 → offen (Intellibox-Modus)
 JP3 → 6pol. Programmierschnittstelle (ISP-Schnittstelle)
 JP4 → offen beim Programmieren, geschlossen im Betrieb
 JP5, JP6 → Stellung 2-3 für USB-Betrieb

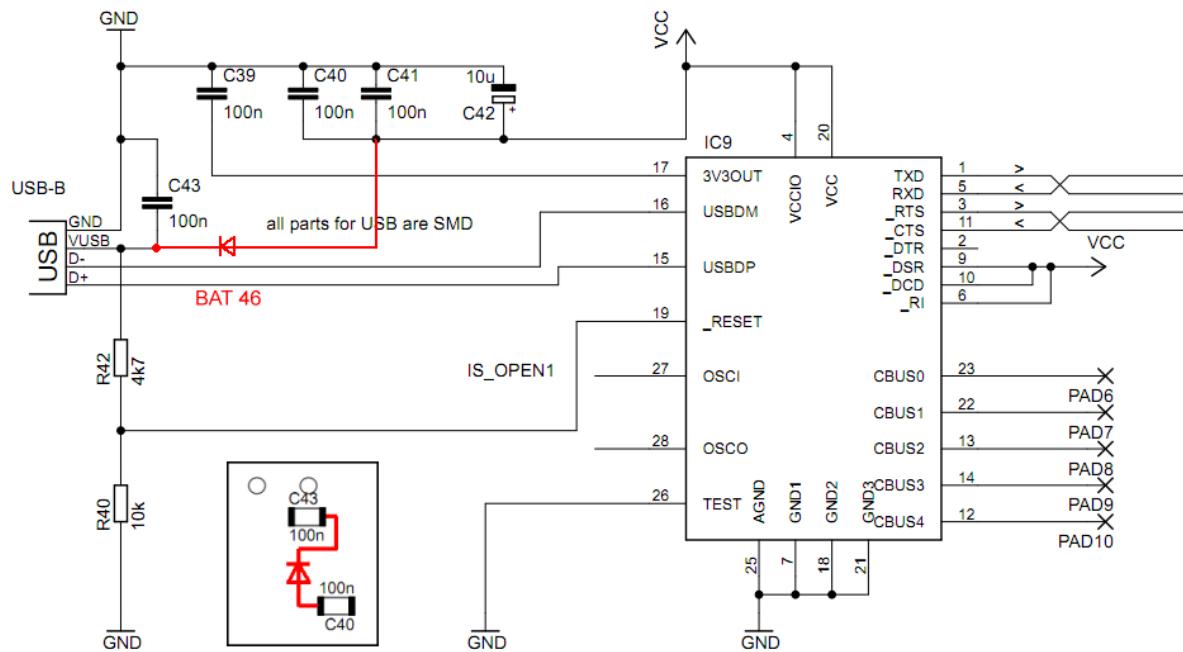
SJ1 → offen
 SJ2 → geschlossen
 SJ4 → geschlossen
 SJ6 → offen

Ergänzung für wUSBiso:

Der Originalaufbau der Hardware ist so gestaltet, dass die Zentrale von der USB-Schnittstelle nur die Signale verwendet, es erfolgt keine Versorgung mit der +5V-Leitung.

Wird jetzt die wUSBiso-Schaltung als Vorsatz verwendet, um eine Isolierung von PC und Zentrale zu erreichen (Stichwort: SELV), so fehlt dem Ausgangsteil von wUSBiso eine Spannungsversorgung.

Abhilfe schafft hier eine zusätzliche Schottky-Diode BAT46, diese lässt sich am besten auf der Unterseite zwischen den Anschlüssen von C40 und C43 platzieren.



Meine Inbetriebnahme

Die eigentliche Inbetriebnahme ist in den Unterlagen zur Zentrale gut beschrieben, hier meine Vorgehensweise:

FTDI

- FTDI_MProg31 (<https://www.ftdichip.com/Support/Utilities/MProg3.5.zip>) installieren
Achtung: die nachfolgende Erläuterung bezieht sich auf Version3.1!
- 15V an Zentrale anschließen
- USB-Kabel an die Zentrale und PC anschließen → Windows7 installiert einen passenden Treiber
- MProg starten
- OpenDCC_V1.2.ept (https://opendcc.de/elektronik/opendcc/OpenDCC_V1.2.ept) laden
- Button Suchen (der mit der Lupe) betätigen
- Button Speichern (der mit dem Blitz) betätigen

FTDI-Treiber

Insgesamt werden zwei USB-Treiber benötigt und installiert:

- **USB-IF OpenDCC V1.2** aus dem Verzeichnis **FTDI\Treiber**, Datei **ftdibus.inf**
- **USB Serial Port** aus dem Verzeichnis **FTDI\Treiber**, Datei **ftdiport.inf**

Besonderheiten unter Win10 – meine Vorgehensweise

(siehe auch: https://www.opendcc.de/elektronik/usb/opendcc_usb.html)

- Von <https://www.ftdichip.com> den aktuellen FTDI-Treiber für VCP (=VirtualComPort) herunterladen (aktuell: v2.12.28) und entpacken
- Von <https://www.opendcc.de> die zugehörigen Installationsdateien herunterladen und die Dateien im FTDI-Verzeichnis überschreiben.

Die Installation schlägt hier jetzt allerdings fehl - wir müssen Win10 etwas überlisten:

Die Treibersignierung muss abgeschaltet werden!

- **Secure Boot** im BIOS deaktivieren
(im Zweifelsfall bei Google nachschlagen,
Stichwort **win 10 treibersignierung abschalten**)
- die Eingabeaufforderung mit Administratorrechten öffnen
- die folgenden Befehle eingeben und mit **Enter** bestätigen
 - o BCDEDIT -Set LoadOptions DDISABLE_INTEGRITY_CHECKS
 - o BCDEDIT -Set TESTSIGNING ON
- Computer neu starten.

Nach dem Neustart können die beiden Treiber installiert werden:

- **USB-IF OpenDCC V1.2** aus dem Verzeichnis **FTDI\Treiber xxxx**, Datei **ftdbibus.inf**
- **USB Serial Port** aus dem Verzeichnis **FTDI\Treiber xxxx**, Datei **ftdiport.inf**

Dazu jeweils die Datei auswählen, Kontextmenü mit der rechten Maustaste öffnen und **installieren** auswählen, Fragen mit „ja, ich will...“ beantworten ☺.

Zum Schluss wird die Treibersignierung wieder aktiviert:

- die Eingabeaufforderung mit Administratorrechten öffnen
- die folgenden Befehle eingeben und mit **Enter** bestätigen
 - o BCDEDIT -Set LoadOptions EENABLE_INTEGRITY_CHECKS
 - o BCDEDIT -Set TESTSIGNING OFF
- Computer neu starten.

Fuses und Bootloader

Für das Programmieren des Controllers hilft diese Batch-Datei:

```
@echo setzen der Fuses
pause
avrdude -c avrisp -p m644p -P com3 -U lfuse:w:0xCE:m
avrdude -c avrisp -p m644p -P com3 -U hfuse:w:0xDC:m
avrdude -c avrisp -p m644p -P com3 -U efuse:w:0xFD:m

@echo kontrollieren der Fuses
pause
avrdude -c avrisp -p m644p -P com3 -U lfuse:r:-:i
avrdude -c avrisp -p m644p -P com3 -U hfuse:r:-:i
avrdude -c avrisp -p m644p -P com3 -U efuse:r:-:i

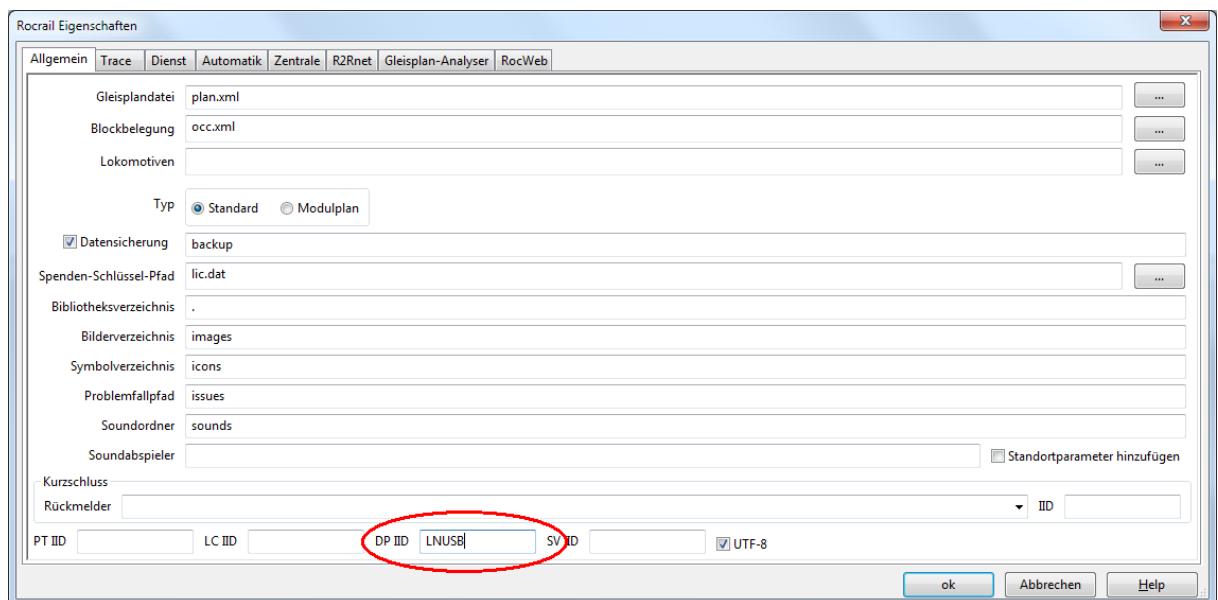
@echo Bootloader flashen
pause
avrdude -c avrisp -p m644p -P com3 -U flash:w:bootloader.hex:i ↵
-U eeprom:w:bootloader.eep:i

@echo FirmWare flashen (p50x-Emulation)
pause
avrdude -c avrisp -p m644p -P com3 -U flash:w:OpenDCC_XP.hex:I ↵
-U eeprom:w:OpenDCC_XP.eep:i
pause
```

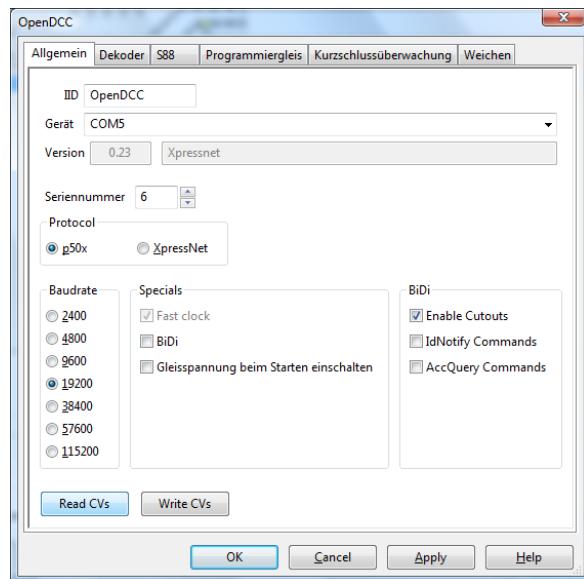
OpenDCC-Zentrale

RocRail-Einstellungen (PC)

- RocRail und RocView starten
- RocRail-Eigenschaften → Reiter Allgemein
- DP IID = **LNUSB**



- RocRail-Eigenschaften → Reiter Zentrale
Neue Zentrale OpenDCC erstellen, siehe hier:
https://opendcc.de/elektronik/opendcc/config_mit_rocrail.html
 - o COM-Port einstellen
(JMRI-Linux: /dev/opendcc/opendcc)
Achtung: nach der Einstellung / dem Wechsel des COM-Portes ist RocRail neu zu starten!
 - o Baudrate auf 19200Bd stellen.
Wichtig: diese Baudrate ist auch direkt am COM-Port einzustellen:
Systemsteuerung → Geräte-Manager → Anschlüsse (COM & LPT)
(Hierzu sind Admin-Rechte erforderlich!)
 - o Button „ReadCV“ drücken → es sollte Version 0.23 Xpressnet angezeigt werden.



RocRail.ini:

```
<digint iid="OpenDCC" lib="opendcc" device="COM5" bps="19200"
    fbmod="6" poweratstartup="false" restricted="false"
    stress="false" libpath=".">
    <opendcc lib="p50x" bidi="false" fastclock="true"/>
</digint>
```

OpenDCC-Konfiguration anpassen (PC)

Nicht alle Konfigurationswerte lassen sich über RocRail ändern:

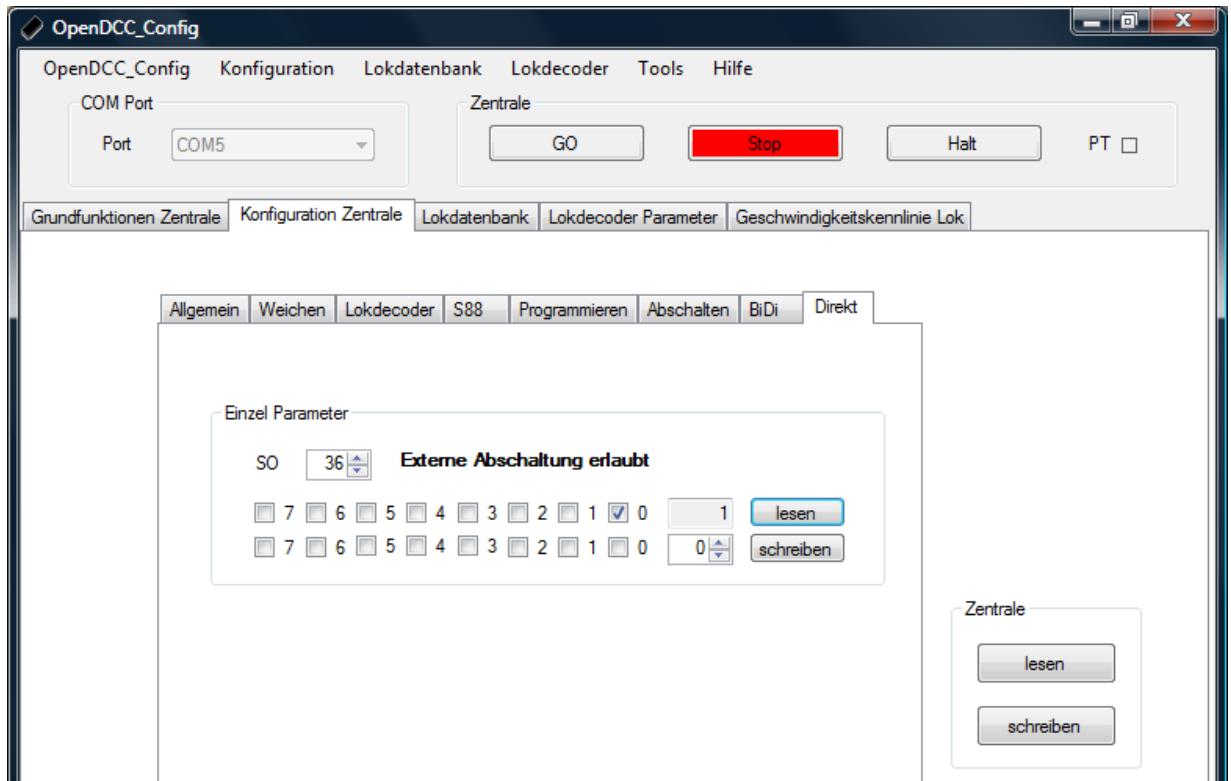
→ hier hilft das OpenDCC-ConfigTool.

Nach dem Anschluss der Zentrale an den PC, der Installation und dem Programmstart:

- COM-Port auswählen und über
- Menü OpenDCC_Config → Verbinden

die Verbindung zur Zentrale herstellen.

Auf dem Tab-Reiter Konfiguration Zentrale und hier auf dem Tab-Reiter Direkt können SO#-Werte gelesen und geschrieben werden.



In obigem Beispiel ist SO#36 auf 1 gesetzt:

36	<p>Hier kann eingestellt werden, ob OpenDCC über den externen Eingang (Ctrl In) abgeschaltet wird.</p> <p>0: kein Abschalten über diesen Eingang 1: bei Anliegen einer Spannung von mind. 5V wird der Gleisausgang abgeschaltet - nur bei regulärem Betrieb, nicht bei Programmiermodus. Die Zentrale bleibt nach dem Abschalten in diesem Zustand und wird entweder über Taster oder PC wieder freigegeben. Zugleich wird bei abgeschalteter Zentrale der Ctrl-Out aktiviert - damit kann eine Abschaltmeldung als Open Collector Verknüpfung verdrahtet werden.</p> <p><i>Hinweis: das kann naturgemäß nicht zusammen mit der Weichenrückmeldung oder DMX funktionieren, da die Ein- und Ausgänge nur einmal verwendet werden können.</i></p>
----	--

Download des OpenDCC-ConfigTools:

https://www.opendcc.de/elektronik/opendcc/Setup_Config_Tool_V0.1.8.zip

Informationen zu den SO#:

https://www.opendcc.de/elektronik/opendcc/opendcc_doc_so.html

Informationen zum P50Xb-Format:

https://www.opendcc.de/elektronik/opendcc/opendcc_doc_ib.html

LocoNET® - LocoBuffer II (Hans Deloof)

RocRail-Einstellungen (PC)

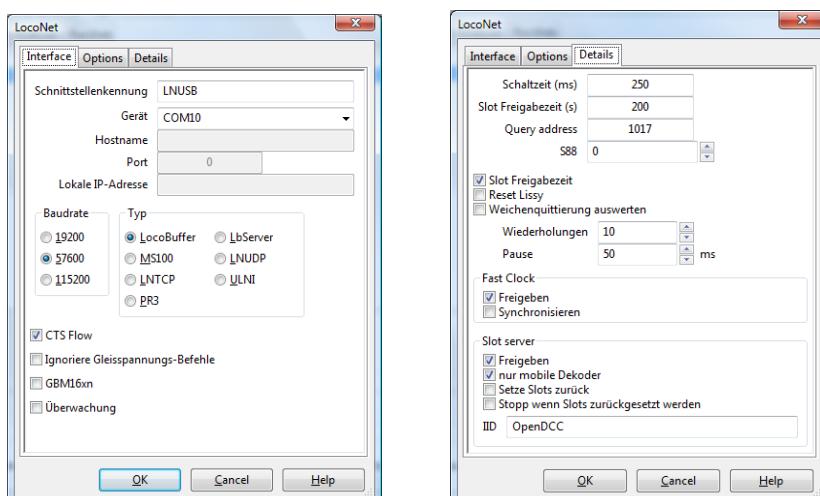
An mein LocoNET® werden auch LocoIO-Platinen angeschlossen. Um diese zu konfigurieren, wird der LocoBuffer II von Hans Deloof (<https://locohdl.synology.me/pageDE9.html>) benötigt – also kommt dieser auch zum Einsatz. Da aber mein bereits vor längerer Zeit gebauter LocoBuffer (Software LB163.hex) noch eine RS232-Schnittstelle hat, wird ein RS232-USB-Konverter benötigt. Wichtig: unbedingt Konverter mit originalem FTDI-Chipsatz verwenden, z.B.: DIGITUS DA-70156 von Reichelt https://www.reichelt.de/de/de/shop/produkt/usb_2_0_konverter_a_stecker_auf_rs-232-99617?search=DIGITUS%2520DA-70156&.

Um eine vollwertiges LocoNET® zu erhalten, wird das DCC-Signal der Zentrale an den LocoBuffer angeschlossen, die Spannungsversorgung (+15V) für den LocoBuffer kommt ebenfalls von der Zentrale.

Wird jetzt die [wUSBiso](#)-Schaltung als Vorsatz verwendet, um eine Isolierung von PC und Interface zu erreichen, fehlt dem Ausgangsteil von [wUSBiso](#) (Richtung LocoBuffer) eine Spannungsversorgung. Die hier erforderlichen 5V werden vom LocoBuffer genommen.

Siehe: „[Zusammenschaltung](#)“

- RocRail und RocView starten
- RocRail-Eigenschaften → Reiter Zentrale
Neue Zentrale LNUSB erstellen:
 - COM-Port einstellen
(JMRI-Linux: `/dev/opendcc/LocoBuffer`)
Achtung: nach der Einstellung / dem Wechsel des COM-Portes ist RocRail neu zu starten!
 - Type = LocoBuffer
 - Baudrate auf 57600Bd stellen.
Wichtig: diese Baudrate ist auch direkt am COM-Port des PC einzustellen:
Systemsteuerung → Geräte-Manager → Anschlüsse (COM & LPT)
(Hierzu sind Admin-Rechte erforderlich!)
 - CTS-Flow aktivieren
- Reiter Details
 - Slotserver: freigeben für nur mobile Dekoder
 - IID = OpenDCC

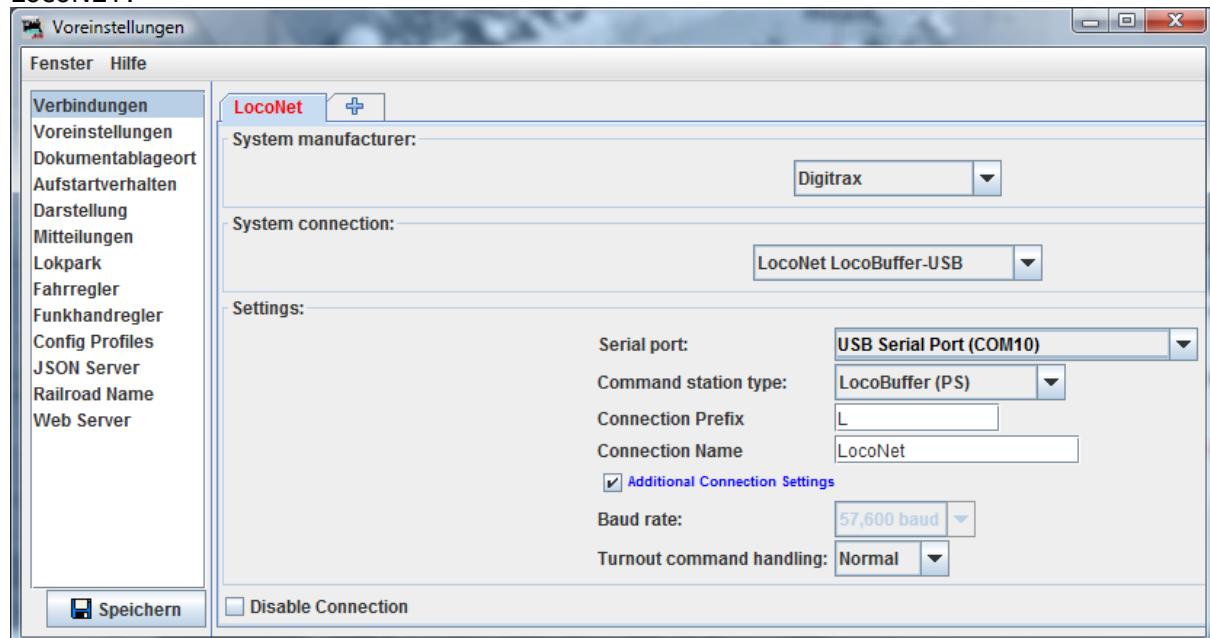


RocRail.ini:

JMRI (DecoderPRO)-Einstellungen (PC)

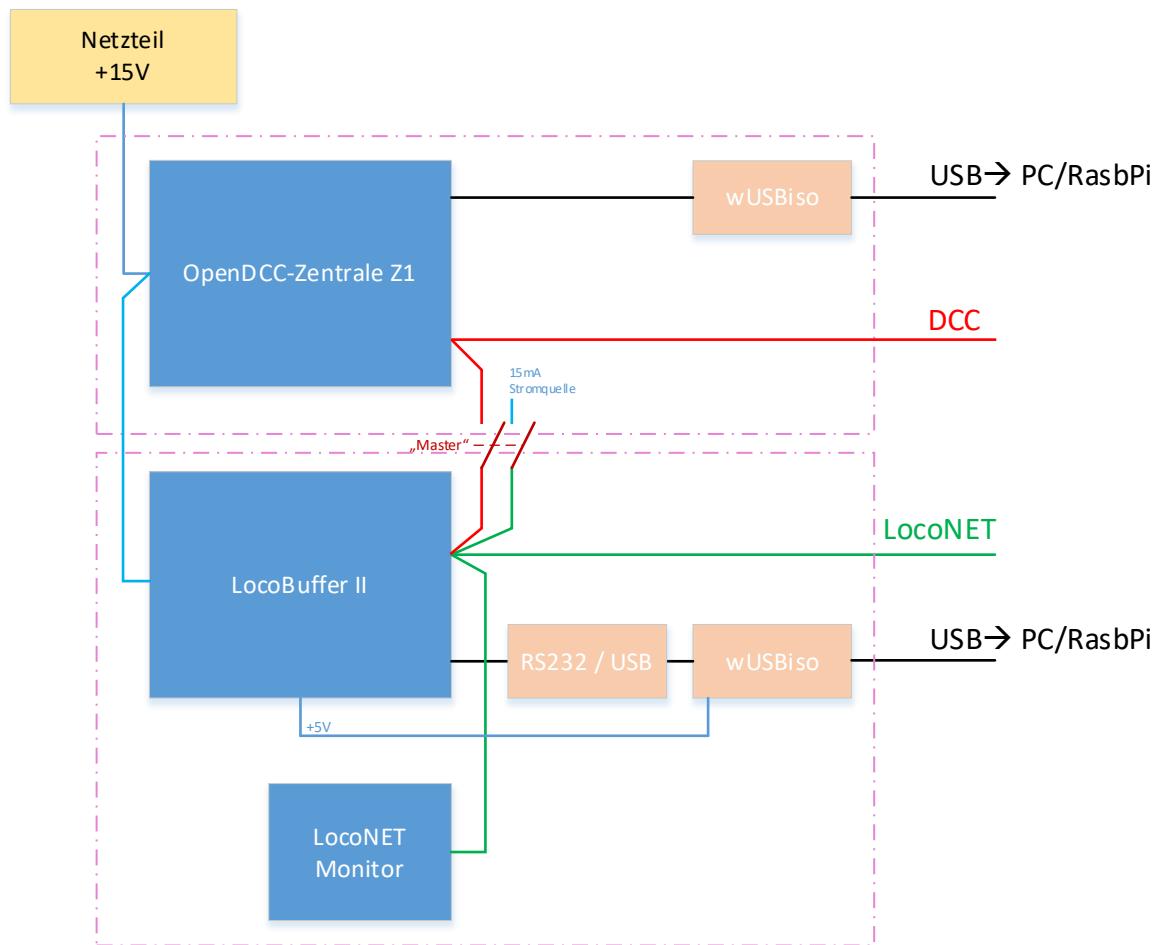
Serial port auf dem Raspberry: **ttyUSB0** oder **ttyUSB1** – je nachdem, welches Gerät wo angeschlossen ist.

LocoNET:



Zentrale:
(fehlt noch...)

Zusammenschaltung



Das DCC-Signal der Zentrale wird über den Schalter „Master“ an das LocoNET® weitergeleitet:

- „Master ein“ bedeutet: DCC-Signal von OpenDCC-Zentrale und 15mA-Stromquelle für das LocoNET® sind aktiv
- „Master aus“ entkoppelt das DCC-Signal von der OpenDCC-Zentrale und deaktiviert die 15mA-Stromquelle