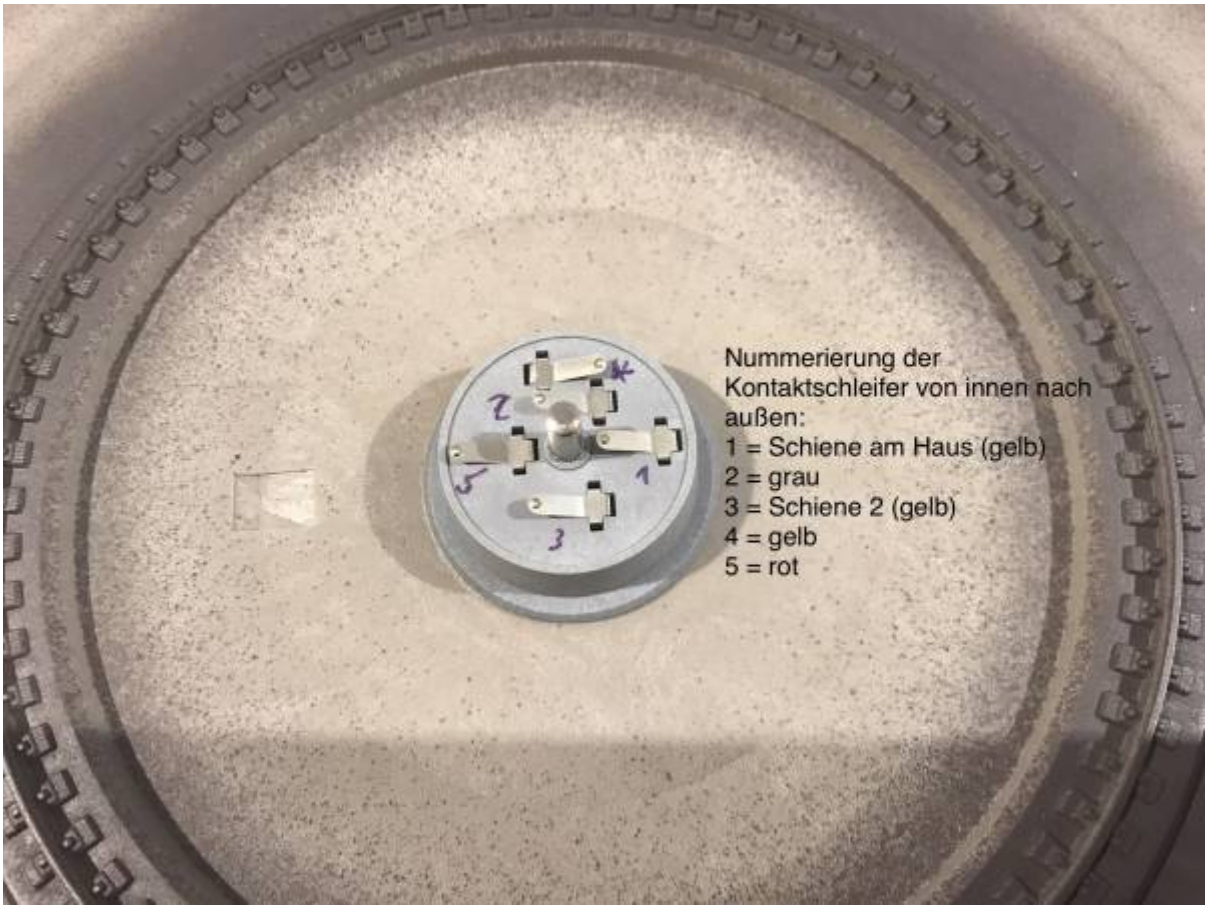
Beleuchtungseffekte und Signale

Bei der Steuerung von Beleuchtungen, Signalen sowie von weiteren Effekten auf der Drehbühne ist zu beachten, dass nicht unendlich viele Kontaktschleifer auf dem sog. Königsstuhl (Mitte der DS, wo die Bühne aufliegt) verfügbar sind, um die Steuerungssignale weiterzuleiten.

Bei der Fleischmann-Drehscheibe 6152 C gibt es z.B. 5 Schleifer. 2 werden für die Bühnenschienen benötigt (Schleifer 1 und 3), die anderen 3 können anderweitig genutzt werden (vgl. Abbildung), beispielsweise um 2 LEDs direkt zu betreiben oder mehrere LEDs sowie weitere Verbraucher über ein kleines Bus-System.

Die Messingachse, mit der die Bühne befestigt ist, könnte noch als 6. Schleifer benutzt werden.

Welche Möglichkeiten bei anderen Drehscheiben bestehen, hängt also stark von der Anzahl der verfügbaren Schleifer ab. Rein theoretisch könnte man einen der beiden Gleisanschlüsse „doppelt“ verwenden und sozusagen eine gemeinsame Masse nutzen. Damit würde man mit entsprechender Beschaltung einen zusätzlichen Schleifer gewinnen. Eine Beschriebung würde hier aber zu weit führen.



Überblick

Folgende Möglichkeiten bestehen:

1. Nutzung der beiden LED-Anschlüsse direkt auf der DS-Platine

- Der sketch steuert Hausbeleuchtung und Warnleuchte direkt an. Der Anschluss der beiden LEDs erfolgt über 3 Leitungen (1 x GND, 1 x Hausbeleuchtung, 1 x Warnleuchte/Flash). Hierfür können die 3 freien Schleifer des Königstuhls verwendet werden.
- Geeignete LED-Vorwiderstände auf der Platine sind notwendig.
- Keine Einschränkung i.V.m. DCC
- es ist jedoch keine Signalsteuerung möglich!

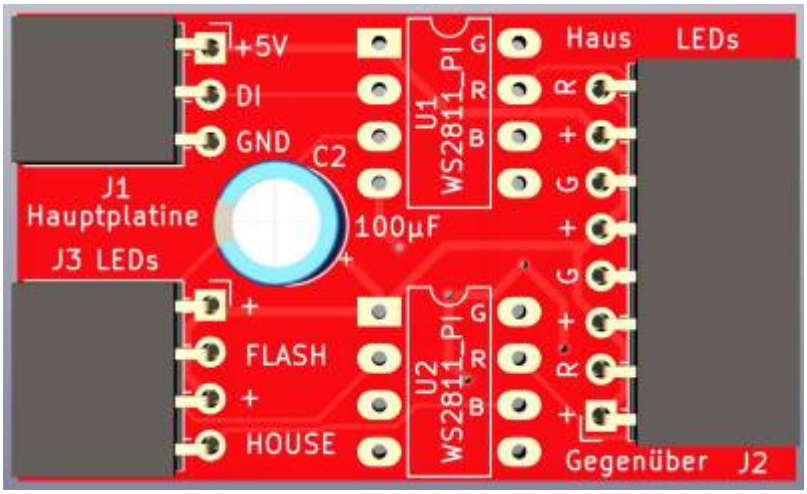
2. Bühnenplatine über FASTLed-Library für ein paar Effekte

- Der Turntable-Sketch steuert direkt LEDs über die FASTLED-Library an. Es sind je nach Platine auf der Bühne mehrere Effekte möglich (Signale, Servos, Beleuchtung).
- Bei Verwendung von mehreren Effekten u.U. Empfangsprobleme bei DCC-Signalen, da während der FASTLED.show()-Aktualisierung der DCC-Empfang beinträchtigt wird (Interrupt wird für einige µs abgeschaltet)!
- Dies funktioniert in der Praxis trotzdem sehr gut, da im Normalfall Hausbeleuchtung, Warnleuchte und Gleissperr-Lichtsignal ausreichen (6 Kanäle von 2 RGB-LEDs) und v.a. da die DCC-Befehle mehrfach von der Zentrale wiederholt/verschickt werden.
- Auf der Bühne wird eine kleine Platine benötigt zum Anschluss der Effekte an WS2811 (vgl. rote Bühnenplatine, sie eignet sich für LEDs, betrieben an 5V.)

3. Interface zur MobaLedLib

- Anbindung der DS-Platine an die MobaLedLib-Hauptplatine über den DCC-Nano
- Parametrierung der Effekte über Excel möglich (LED-Nano), sinnvoll bei komplexeren Effekten wie Rundum-Warnleuchte bestehend aus 4 LEDs, Servoansteuerung für Sperrsignale, Sound direkt auf der Bühne etc.
- keine Einschränkungen bei DCC-Empfang
- zusätzlicher Code hier im Sketch zum Senden von Befehlen an den DCC-Nano
- im DCC-Nano wird ein spezieller Sketch benötigt, der Empfangen dieser Befehle beherrscht
- DS-Nano und DCC-Nano müssen über 2 Leitungen miteinander verbunden sein (Pin D10 an RX vom DCC-Arduino plus GND)

Drehbühnenbeleuchtung und -effekte mit FASTLED-Library (rote Bühnen-Platine)



- Beleuchtung Bühnenhaus, 2 x 2 Signale mit zwei Signalbildern (parallel angeschlossen) und Warnblinken werden direkt von der Hauptplatine über die FASTLed-Lib gesteuert.
- Für diese 6 Kanäle werden 2 x WS2811 benötigt. Jeder hat 3 Farbkanäle RGB, die jeweils einen Verbraucher (LED) ansteuern können. Die Footprints für die WS2811 sind nicht geeignet für DIL-Chips, da die Pinbelegung nicht passt. Die Footprints sind vielmehr für kleine Abschnitte aus „Schokoladentafeln“ ausgelegt (WS2811-Platinchen).
- Über die Drehbühnenschleifkontakte werden von der Steuerung GND, +5V und DATA an eine kleine Platine direkt auf der Bühne weitergegeben
- Die Platine beinhaltet 2 x WS2811, 1 x 100 oder besser 470µF.
- Alle Verbraucher (LEDs) werden über 5V betrieben. Ggf. sind die Vorwiderstände bei Signalen anzupassen sowie Dioden zu entfernen, da die Signale i.d.R. auf 14V AC ausgelegt sind. Theoretisch können die Vorwiderstände weggelassen werden, da das WS2811 als Konstantstromquelle agiert! Erfahrungsgemäß sind weiße LEDs dann aber zu hell. Bei den Viessmann Lichtsignalen haben sich 4,7kOhm vor die weißen LEDs bewährt, bei den roten LEDs ist kein Vorwiderstand notwendig
- Achtung: Es werden keine RGB-LEDs sondern monochrome LEDs eingesetzt. Diese werden aber über einzelne RGB-Kanäle angesprochen!

Folgende 2 x 3 Kanäle von 2 x WS2811 werden genutzt:

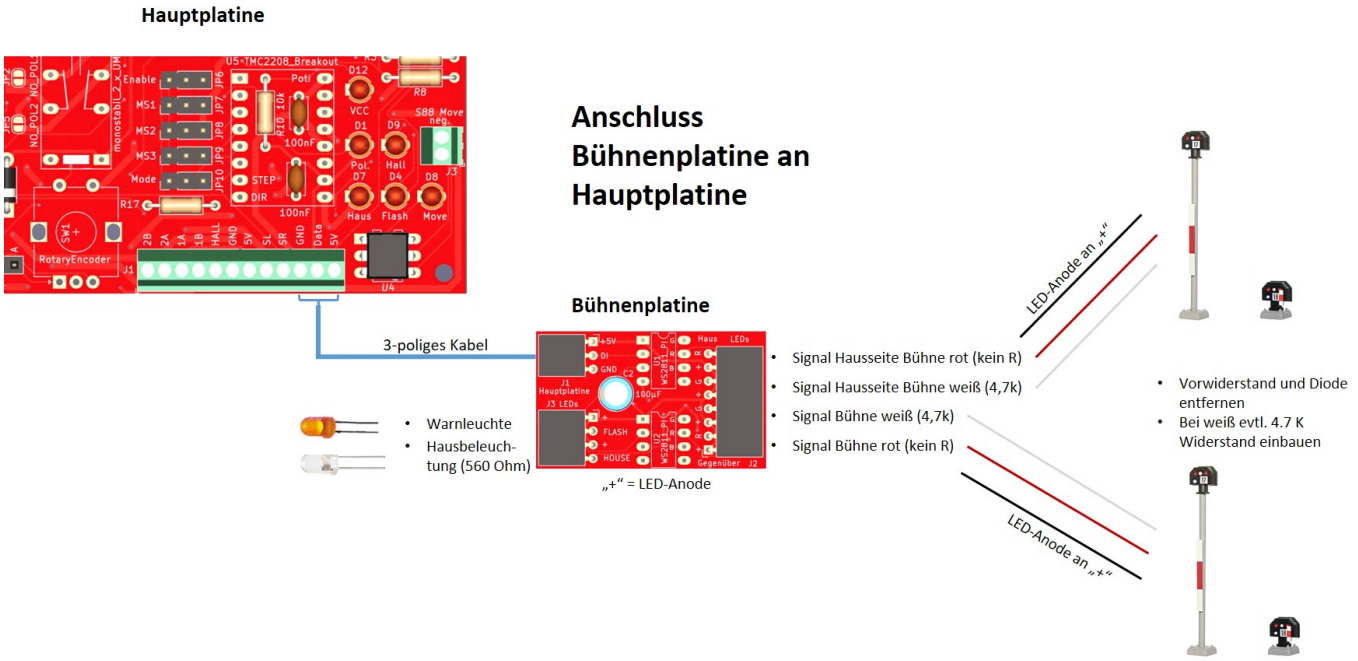
- Signal_rot_Gegenseite = Rot_1 –> Überblenden
- Signal_grün_Gegenseite = Gruen_1 –> dito.
- Warnleuchte = Blau_1 –> blinken / flashen

- Signal_rot_Haus = Rot_2 –> Überblenden
- Signal_weiß_Haus = Gruen_2 –> dito.
- Innenbeleuchtung Haus = Blau_2 –> Fade in/out oder Neonflackern

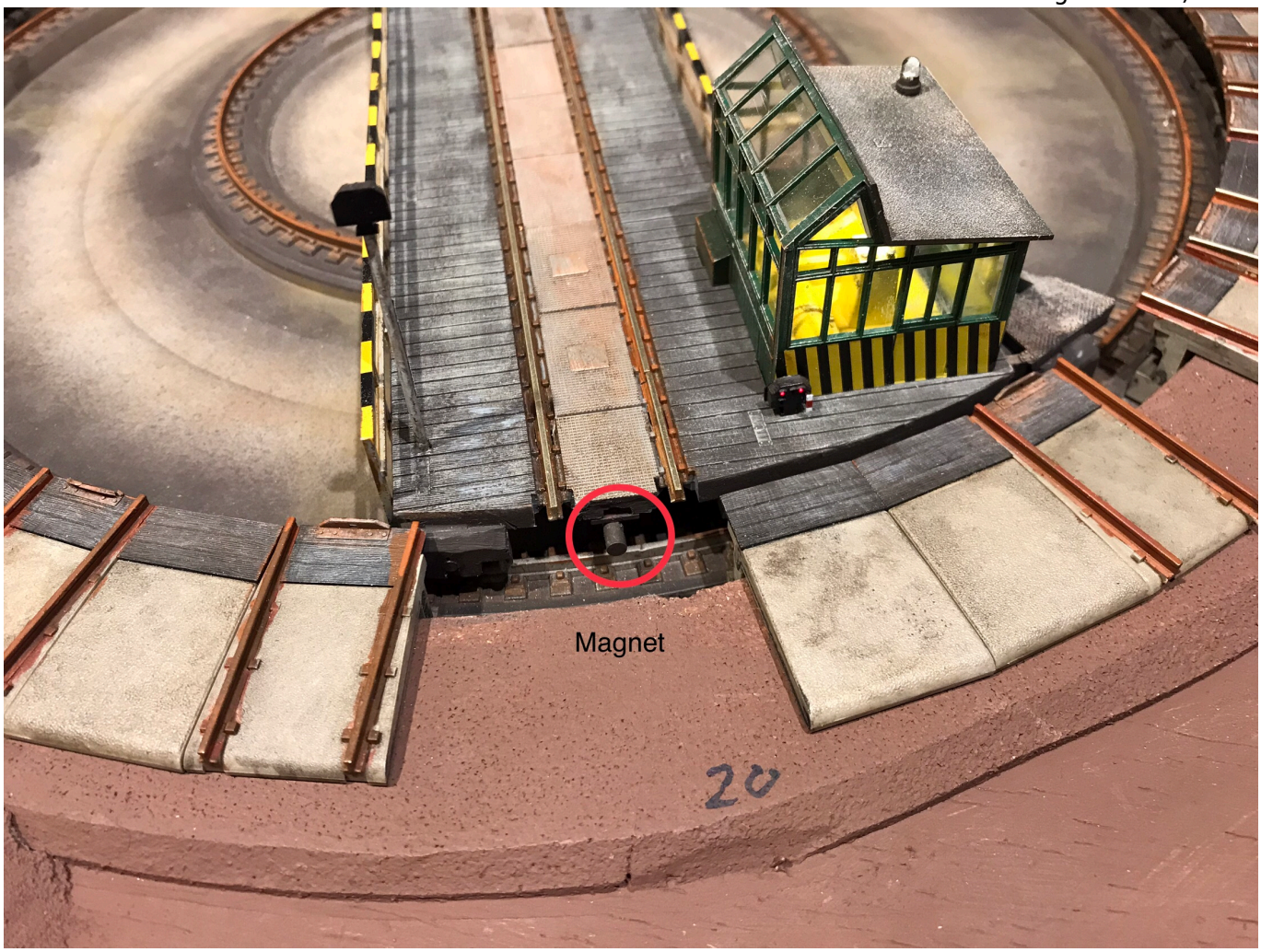
Anschluss der Bühnenplatine

Die folgende Abbildung zeigt, wie die Bühnenplatine an die Hauptplatine angeschlossen wird. Genau genommen geht das 3-polige Kabel von unten an die Schleiferanschlüsse des Drehscheiben-Königstuhl´s. Über die Schleifer gelangen die Signale auf die standardmäßig eingebaute

Drehscheibenplatine und von dort aus auf die rote Bühnenplatine.



So werden die Signale auf der Bühne verbaut (auf dem Hausdach befindet sich eine orangene LED als Warnleuchte und unter dem Hausdach wurde eine weiße LED für die Innenbeleuchtung montiert):



Das niedrige Gleissignal (rechts vor dem Bühnenhaus) ist nach außen gerichtet, das hohe Sperrsignal

(links vom Gleis) nach innen.

Die beiden Signale am jeweiligen Bühnenende werden parallel angeschlossen.

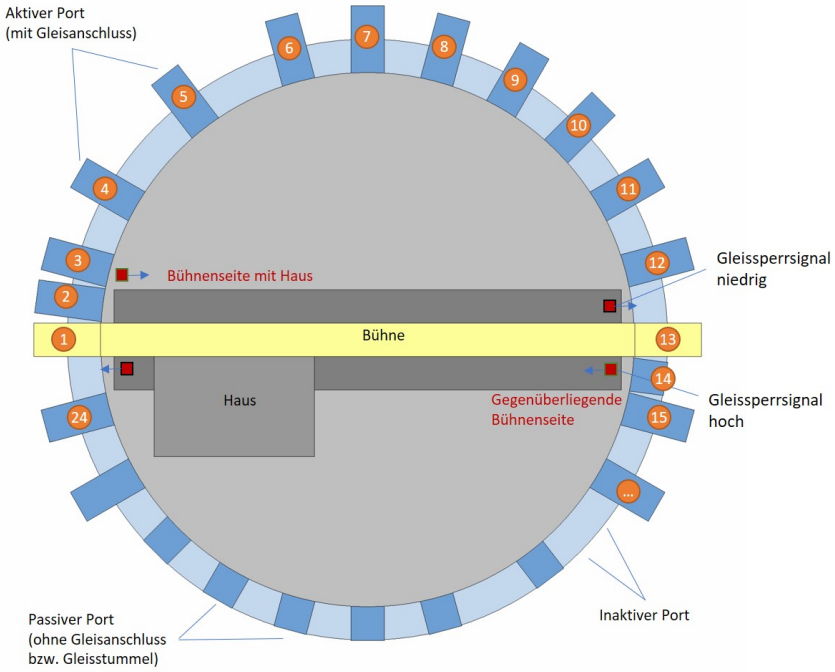
Ansteuerungslogik von Signalen

Folgende Steuerungslogik ist implementiert:

- Wenn sich die Bühne in Bewegung setzt, gehen alle Signale auf rot.
- Am Zielport angekommen wird bei #define ADVANCED_SIGNAL_CONTROL = 1 geprüft, wo das jeweilige Bühnenende steht: Steht das Bühnenende an einem aktiven Port (mit Gleisanschluss) wird das Signal auf weiß (= freie Fahrt) gestellt, bei blinden und inaktiven Ports bleiben die Signal auf rot (= Stopp). Hierfür muss pro Port im PORT_TYPE angegeben werden, ob es ein aktiver Port ist oder nicht. Bei ADVANCED_SIGNAL_CONTROL = 0 werden am Zielport einfach alle Signale auf weiß gestellt.
- Wenn man die Bühne im Nirwana stoppt (Drücken der Encoder-Taste während einer Fahrt oder entsprechender DCC-Befehl) oder wenn man eine Bewegung mit dem Poti beendet, bleiben alle Signale auf rot, da die Bühne i.d.R. nicht exakt an einem gültigen Port steht.

Fleischmann/Märklin Drehscheibe

- 48 Ports im 7.5° Raster
- Jeder aktive Port hat ein passives (blindes) Gegenüber (dort sind dann kleine Gleisstücke/ -stummel angebracht)
- Inaktive/unbenutzte Ports sind wahlweise nicht anfahrbar (nur per Poti)
- Im Beispiel rechts werden nur 24 Ports benutzt
- Auf der DS stehen 4 Lichtsignale, 2 jeweils nach außen gerichtet, 2 nach innen.
- Die nach innen gerichteten (hohen) Signale verhindern die Abfahrt von der Bühne,
- Die nach außen gerichteten (niedrigen) Gleissperrsignale die Auffahrt auf die Bühne
- Bei Bewegung der DS sind alle Signale auf rot.
- Bei Stillstand der DS sind nur die Signale auf weiß (= grün), wo das Bühnenende auf einen gültigen Gleisabgang/Port zeigt, an dem also ein echtes Gleis angeschlossen ist.
- Bei passiven Gleisabgängen/Ports ohne Gleis wird rot angezeigt
- Die zwei Signale am jeweiligen Bühnenende werden paarweise parallel zusammengeschaltet.
- Es gibt bei der FLM-Drehscheibe eine Seite mit Haus und eine ohne.



Drehbühnenbeleuchtung und -effekte mit MobaLedLib

Bei umfangreicheren Beleuchtungsaktionen (> 10 Verbraucher, Servos, separate Soundmodule etc.) muss der Drehscheiben-Nano mit einer MobaledLib-Hauptplatine kommunizieren, und zwar über den DCC-Nano.

Auf den DCC-Nano muss dafür ein spezieller Sketch geladen werden, damit diese Kommunikation funktioniert („23_A.DCC_Interface_HS_MP_mod.ino“, vgl. Abbildung unten). Dieser Sketch ist beim

User Domapi erhältlich. Im Drehscheiben-Nano selbst wird eine uni-direktionale serielle Schnittstelle aktiviert (SoftwareSerialTX Library).

Immer wenn im DS-Sketch LEDs oder Servos etc. geschaltet werden sollen, wird ein entsprechender Befehl an den DCC-Nano gesendet. Der DCC-Nano puffert die DS-Befehle und schickt sie und andere DCC-Kommandos an den LED-Nano zur Verarbeitung. Das Senden an den LED-Nano erfolgt nur dann, wenn ein Enable-Pin aktiviert ist. Falls deaktiviert, dann werden gerade die LEDs aktualisiert und die Schnittstelle ist abgeschalten.

Das Senden des DS-Nano an den DCC-Nano unterliegt keinen Restriktionen, d.h. sobald etwas im DS-Sketch ausgelöst wird, plappert die Schnittstelle los!

Folgendes ist zu beachten:

- Aktuell ist im Sketch vorgesehen: Steuerung der Hausbeleuchtung, Ansteuerung von 2 x 2 LED-Licht-Signalen mit zwei Signalbildern (je 2 Signale sind parallel angeschlossen) und des Warnblinkers über die MobaLedLib (und damit ebenfalls über FASTLed-Lib.)
- Senden von Befehlen an den DCC-Nano (dort muss ein spezieller Sketch ge-flashed werden).
- Es werden alle Hauptplatinen unterstützt, bei Version 1.7 werden das auf der Unterseite vorgesehene Puffer-IC3 und ein Widerstand benötigt.
- Version 1.8x wurde noch nicht getestet - hier wird ein „größeres“ Puffer-IC verwendet (kein SMD-Bauteil), was die gleiche Funktion hat wie IC3 auf der 1.7er Version. Daher sollte die 1.8er Platine ebenfalls integriert werden können.
- Die DS-Hauptplatine muss über 2 Kabel mit der MLL-Hauptplatine verbunden werden: Data und GND

Rote DS-Hauptplatine:

- Data (Pin D10 des Nano) und GND sind auf der rechten Seite der großen grünen Anschlussklemme verfügbar. Hierüber wird die MLL-Hauptplatine angeschlossen.
- Ein Pullup-Widerstand (R2, 22 kOhm) muss eingelötet werden.
- R1 (100 Ohm) wird nicht benötigt.
- Der JumperJP3 MobaLedLib ist zu schließen.
- JP4 muss offen bleiben.
- Optokoppler U2 und U3 werden nicht benötigt.
- Pin TurnFlash (A3) kann weiterhin die zugehörige LED auf der Platine ansteuern.
- Pin D10 (Haus) übernimmt das Data-Senden an den DCC-Arduino; daher kann die LED auf der sw. Platine nicht mehr sinnvoll verwendet werden (auslöten, macht aber auch nichts, wenn sie drin bleibt, sie flackert dann halt!)

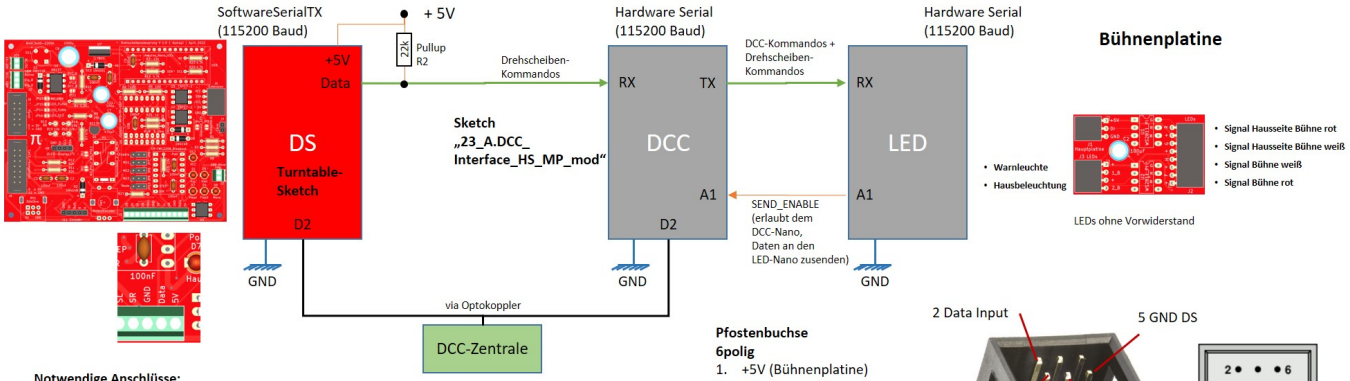
Exkurs - Schwarze DS-Hauptplatine:

- Als Senden-Pin wird Pin D10 des Nano verwendet. Auf der schwarzen Platine muss dann der Jumper DIR_LIGHT geschlossen werden.
- ein 22 kOhm Pullup-Widerstand muss eingelötet werden.
- „Senden“ (= Data) kann am Connector Con 3, Pin 8 abgegriffen werden. R10 muss überbrückt sein.
- GND und +5V (für den Pullup-Widerstand) können ebenfalls von Con 3 verwendet werden oder von den Potianschlüssen.
- Die Optokoppler OK3 und OK4 dürfen nicht bestückt sein!
- Pin TurnFlash (A3) kann weiterhin die LED auf der Platine ansteuern.
- Pin D10 (Haus) übernimmt das Senden an den DCC-Arduino; daher kann die LED auf der schwarzen Platine nicht mehr sinnvoll verwendet werden (auslöten, macht aber auch nichts, wenn sie drin bleibt, sie flackert dann halt!)

- Über die Drehbühnenschleifkontakte werden von der MLL-Hauptplatine GND, +5V und DATA an eine geeignete Platine direkt auf der Bühne weitergegeben. Data Out ist i.d.R. nicht möglich, da zu wenig Schleifer verfügbar sind (die Messingachse könnte hierfür genutzt werden)! Die DS-Effekte sollten daher die letzten in der Kette/Excelliste sein, da danach keine Weiterleitung des DATA-Signal mehr möglich ist.
- Die zusätzliche Platine auf der Bühne beinhaltet die notwendigen WS2811 oder WS2812 (es kann die rote Bühnenplatine verwendet werden oder eine selbst designte).
- Alle Verbraucher (LEDs) werden mit 5V betrieben. Ggf. sind die Vorwiderstände bei Signalen anzupassen, da i.d.R. auf 14V AC ausgelegt.
- Für Erweiterungen, z.B. weitere Effekte wie zusätzliche Beleuchtungen, Sound direkt auf der Bühne etc. müssen ggf. größere Anpassungen im Sketch gemacht werden, wenn diese Effekte von Drehscheibenbewegungen abhängen. Dafür muss man schon etwas tiefer einsteigen und programmieren können.

Überblick rote Drehscheiben-Platine an MLL-Hauptplatine

Drehscheibensteuerung (rote domapi-Platine) an MLL-Hauptplatine (V1.7)



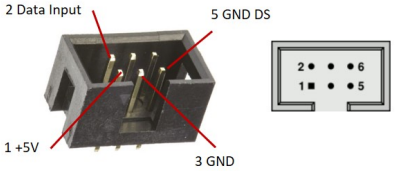
Notwendige Anschlüsse:

- DS-Platine und DCC-Nano müssen über 2 Leitungen miteinander verbunden sein (Ausgang Data von roter DS-Platine an RX-Pin vom DCC-Arduino und GND).
  - GND kann vom Pin „GND“ der vorderen grünen Klemmleiste verwendet werden. Kabel an Pin 5 des 6-poligen Wannensteckers auf der MLL-Hauptplatine anschließen (an dem normalerweise die WS281x hängen)
  - Data wird ebenfalls von der grünen Klemmleiste abgegriffen und an den RX-Pin des DCC-Nano angeschlossen
- Auf der roten DS-Platine muss der Jumper MobaLedLib geschlossen werden (es geht aber auch offen). R1 sollte ausgelötet sein, es geht aber auch mit dem 100 Ohm.
- R2, 22k Pullup-Widerstand ist einzulöten (geht aber auch ohne).

- Eine eigene oder die rote Bühnenplatine wird folgendermaßen mit der MLL-Platine verbunden:
  - +5V → Wannenstecker Pin 1
  - GND → Wannenstecker Pin3
  - DI (Data In) → Wannenstecker Pin 2

Pfostenbuchse
6polig

1. +5V (Bühnenplatine)
2. Data Input (Bühne)
3. GND (Bühne)
4. -
5. GND (DS-Platine)
6. -

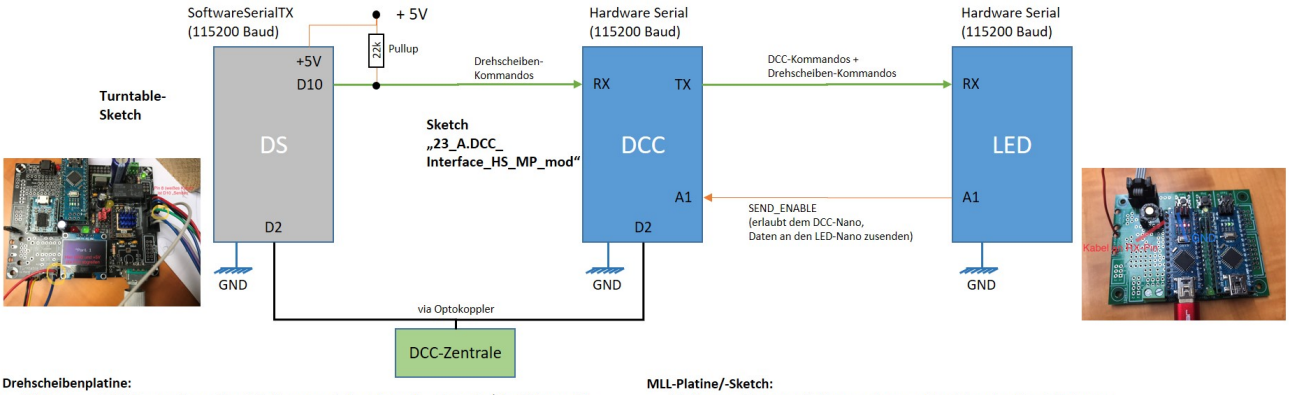


MLL-Platine/-Sketch:

- Auf Version 1.7 IC3/R9 bestücken, damit kann der Sketch das Senden an den LED-Nano abschalten, der Empfang von Drehscheibenbefehle ist aber immer noch möglich! Auf diese Weise kann der LED-Nano geflashed werden, auch wenn der DCC-Nano aktiv eingesteckt ist.
- Es muss ein angepasster Sketch auf den DCC-Nano geflashed werden (23_A.DCC_Interface_HS_MP_mod)

Überblick schwarze Drehscheiben-Platine an MLL-Hauptplatine

Drehscheibensteuerung an MLL-Hauptplatine



Drehscheibenplatine:

- DS-Nano und DCC-Nano müssen über 2 Leitungen miteinander verbunden sein (Pin D10 vom DS-Nano an RX-Pin vom DCC-Arduino plus GND).
- Auf der schwarzen DS-Platine muss der Jumper DIR_LIGHT geschlossen werden. R10 sollte überbrückt/eingelötet sein (0 Ohm); es geht aber auch mit dem 100 Ohm.
- Das "Senden" kann am Connector Con 3, Pin 8 abgegriffen werden. Von dort ein Kabel an den MLL-DCC-Nano RX-Pin löten (am besten von unten).
- GND und +5V können ebenfalls von Con 3 verwendet werden oder von den U$7-Anschlüssen links vom OLED.
- Ein 10 - 22 k Pullup-Widerstand ist vorzusehen (geht aber auch ohne).
- OK3 und OK4 sollten nicht bestückt sein!
- Pin TurnFlash (A3) kann weiterhin die LED auf der Platine ansteuern
- Pin D10 (Haus) übernimmt das Senden an den DCC-Arduino; daher kann die LED auf der sw. Platine nicht mehr sinnvoll verwendet werden
- Um das Flashen des DCC-Nanos nicht zu beeinträchtigen, wird die SoftwareSerialTX-Schnittstelle nur kurz für den Sendevorgang geöffnet.

MLL-Platine/-Sketch:

- Für die verschiedenen Platinenversionen gelten folgenden Einschränkungen:
  - Am besten ist die Version 1.7 geeignet: Dort IC3/R9 bestücken, damit kann der Sketch das Senden an den LED-Nano abschalten, der Empfang von Drehscheibenbefehle ist aber immer noch möglich! Auf diese Weise kann der LED-Nano geflashed werden, auch wenn der DCC-Nano aktiv eingesteckt ist.
  - Bei den beiden anderen Versionen sind i.d.R. Widerstände angeschlossen, um das Flashen des LED-Nano zu ermöglichen (das funktioniert aber nur ohne Drehscheibenanbindung!):
    - Version 1.6: der 3,9 kOhm Pulldown-Widerstand ist bereits auf der Platine vorgesehen.
    - Version 1.5: auf der Unterseite der Platine wurde meist ein 3,9 kOhm Widerstand eingelötet.
  - Achtung: Diese Widerstände helfen jedoch n i c h t , wenn der Drehscheiben-Nano/-Sketch an die MLL angebunden wird. Hier muss die Hardware Serial immer aktiv sein, um Drehscheibenbefehle zu empfangen.
  - Das bedeute dann aber auch, dass der LED-Nano bei V1.5 und V1.6 Platinen nicht geflashed werden kann, wenn der DCC-Nano eingesteckt ist!
- Es muss eine angepasster Sketch auf den DCC-Nano geflashed werden (23_A.DCC_Interface_HS_MP_mod)
- Kabel vom DS-Nano Pin D10 an den RX-Pin des DCC-Nanos löten.
- GND am besten von Pin 6 der ICSP verwenden (unten links)

Ansteuerungslogik von Signalen

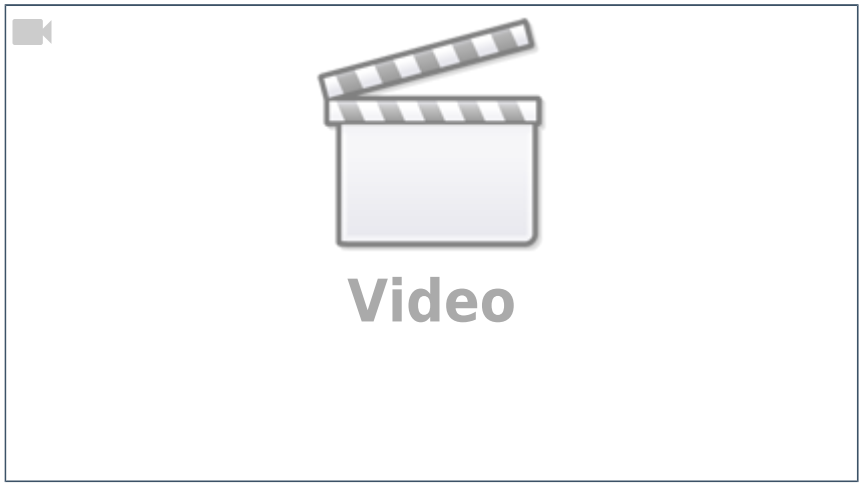
Siehe oben.

Ansteuerung von Gleissperr-Formsignalen über die MobaLedLib

Bernd hat hier dokumentiert, wie er Weinert Gleissperrsignale mit Servos über die MLL an der Drehscheibensteuerung betreibt. Er nutzt die Messingachse, um das Data-out-Signal der WS2811-Kette zurück an weitere Verbraucher weiterzuleiten.

Gleissperr-Formsignale mit MobaLedLib

Hier ein kurzes Video:



From:
https://wiki.mobaledlib.de/ - **MobaLedLib Wiki**

Permanent link:
https://wiki.mobaledlib.de/anleitungen/bauanleitungen/locoturn_v10/150_locoturn_beleuchtung_signale?rev=1686851499

Last update: **2023/06/15 18:51**