

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

Michelstadt (Bw)

LocoIO
Zusammenfassung der relevanten Dokumentationen
Version 9

Inhaltsverzeichnis

LocoIO – Ergänzungen zu Konfiguration und Betrieb (M.Zimmermann) 3

Für Experten (oder diejenigen, die Internes kennen möchten):

LocoIO module software 1.48 – SV-Übersicht (H.Deloof) 36

Befehlsübersicht (Digitrax, Ausschnitt aus der Loconetpersonaledition).. 40

Lesehinweis: Alle Dokumente beziehen sich – soweit möglich – auf die von mir verwendeten Versionen (also LocoHDL Version 3.5.2 bzw. LocoIO Version 1.48) und nicht auf die aktuell verfügbarsten Versionen. In verschiedenen Textabschnitten werden auch Screenshots der LocoHDL Version 4.0.6 zum Vergleich gezeigt.

LocoIO – Zusammenfassung der relevanten Dokumentationen

Versionsgeschichte:

vor Version 2	nicht nachgehalten
Version 2	22.11.2022 redaktionelle Überarbeitung und Hinzufügen von Screenshots aus LocoHDL 4.0.6
Version 3	23.11.2022 weitere redaktionelle Überarbeitungen
Version 4	08.06.2023 Ergänzungen (Expertenmodus und TwinCenter)
Version 5	06.08.2023 Ergänzungen zu „Einen Ausgang ... steuern - Umschalter“
Version 6	11.05.2024 Linkliste ergänzt
Version 7	20.09.2024 redaktionelle Überarbeitung, Links korrigiert
Version 8	17.01.2026 redaktionelle Überarbeitung, Links korrigiert
Version 9	26.01.2026 Dokument verschlankt

Linkliste

Verwendete Links rund um die LocoIOs:

- Das Original:
<https://web.archive.org/web/20220405130232/http://locobuffer.com/LocoIO/LocoIO.htm>
- LocoBuffer von Hans Deloof: <https://locohdl.synology.me/pageDE9.html>
- LocoIO von Hans Deloof: <https://locohdl.synology.me/pageDE8.html>
- LocoHDL von Hans Deloof: <https://locohdl.synology.me/pageDE7.html>
- LocoNET®-Spezifikation von Digitrax:
<https://www.digitrax.com/support/loconet/loconetpersonaledition.pdf>
http://embeddedloconet.sourceforge.net/SV_Programming_Messages_v13_PE.pdf
- Die SV-Übersicht:
<http://wiki.rocrail.net/lib/exe/fetch.php?id=loconet-io-de&cache=cache&media=loconet:lio-sw:locoio.pdf>
- Ergänzungen zu Konfiguration und Betrieb:
<https://github.com/Kruemelbahn/LocoIO/blob/main/Documentation/LocoIO-Erg%C3%A4nzungen.pdf>

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

Michelstadt (Bw)

LocoIO
Ergänzungen zu Konfiguration und Betrieb
 auch zusammen mit RocRail, JMRI oder dem TwinCenter

Inhaltsverzeichnis

Anstelle eines Vorwortes	2
Hinweis zum Programm LocoHDL und dem USB-LocoBuffer	3
Verschiedene LocoHDL-Versionen	4
Begriffe bei Hans Deloof – der Versuch einer Erläuterung	5
Konfigurieren – was heißt das?	7
Konfiguration des LocoBuffer in RocRail	7
Sinnvolle Reihenfolge – oder was ist zu tun?	8
Anschließen des LocoBuffer und des LocoIO-Moduls	9
Initialisierung des LocoIO-Moduls	9
Adresse für das LocoIO-Modul einstellen	10
Es geht los: LocoIO-Modul anwählen	14
Vertrauen ist gut – Kontrolle ist besser	15
Was macht man mit unbenutzten Anschlüssen?	15
Einen Ausgang mit einem Eingang steuern - Umschalter	16
Steuerung mit RocRail	17
Einen Ausgang mit einem Eingang steuern - Blockbelegtmeldung	19
Anzeige in RocRail	20
Steuerung mit RocRail	21
Wechselnde Kode für Druckknopf	22
Einen oder zwei Ausgänge mit zwei Eingängen steuern	24
Verwendung als Impulsausgang (anstelle eines Dauerkontaktes)	25
Steuerung mit RocRail	26
Steuerung mit dem TwinCenter / der Intellibox	28
Vier Eingänge – vier Ausgänge	29
Geht doch – oder?	30
Module/Schaltungen, die mit LocoIO angesteuert werden können	32
Konfigurieren mit JMRI	32
Zu guter Letzt ... speichern der Einstellungen	32
Eine Dokumentationshilfe – die Adressabelle	32
Linkliste	33

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.*

Anstelle eines Vorwortes

LocoIO war eine Initiative von [John Jabour](#), der für das LocoNET®-Protokoll ein auf einem PIC-Mikrocontroller basierendes frei konfigurierbares 16-fach Ein-/Ausgabe-Board entwickelt hat. Dieses Board stellt hierfür Ports des Mikrocontrollers zur Verfügung. Für die Ansteuerung von Relais, Weichen, Signalen usw. ist immer eine angepasste Hardware erforderlich.

Technische Daten eines LocoIO ¹	<i>LocoIO by H.Deloof</i>	<i>wLocoIO by W.Hückel</i>
Versorgungsspannung der LocoIO-Platine	12V=	5V=
max. Spannung an einen Port (=Anschluss)		5V
Strombelastung je Port (=Anschluss)		max. 25mA
Strombelastung insgesamt an allen Ports (=Anschlüssen)		max. 200mA
(opto)isolierter LocoNET®-Anschluss		nein

Dieses Dokument „*LocoIO - Ergänzungen zu Konfiguration und Betrieb*“ bezieht sich auf die LocoIO von Hans Deloof (<https://locohdl.synology.me/pageDE8.html>) und soll dem Anwender helfen, die Dokumentationen von Hans Deloof besser zu verstehen, sie ersetzt diese keinesfalls!



Die Kenntnis der Dokumentationen zu LocoHDL und LocoIO von Hans Deloof wird zwingend vorausgesetzt!

Grundlagen zu LocoIO sind auch in der Dokumentation „*wLocoIO-2 – Bauanleitung (W.Hückel)*“ in den Kapiteln 2, 3 und 8 nachzulesen, siehe Link in der Linkliste am Ende des Dokuments.

Und noch etwas:

- viele Beispielbilder in dieser Beschreibung aus LocoHDL beziehen sich oftmals auf ein LocoIO-Modul, was eigentlich die Ausnahme ist. Tatsächlich wird es in den meisten Fällen so sein, dass sich Ein- und Ausgänge auf verschiedenen LocoIO-Modulen befinden. Aber es geht eben auch so...
- Die in diesem Dokument beschriebenen Basiskonfigurationen sind i.d.R. auch in höheren PIC-Versionen verfügbar – im Zweifelsfall ist das Originalhandbuch in der zum LocoIO passenden Version ausschlaggebend!
- In dieser Dokumentation werden bei Bedarf Angaben und Hinweise aus den Dokumenten von Hans Deloof wiederholt – damit man nicht ständig hin und her blättern muss.
- LocoIO basiert auf einem Meldungssystem, d.h. nur *Signaländerungen* (also Signalwechsel an einem Eingang) erzeugen ein LocoNET®-Telegramm.
- LocoIO reagiert auch auf das LocoNET®-Telegramm OPC_GPON und sendet daraufhin den Status aller Eingänge mit Hilfe von OPC_SW_REQ(B0)- und OPC_INPUT_REP(B2)-Telegrammen.
- RocRail unterstützt die Module von Hans Deloof nicht. Da die verwendeten Telegramme aber mit denen des GCA50 identisch sind, funktioniert die Unterstützung dennoch.
- Wurde das LocoIO und / oder der zugehörige PIC-Prozessor direkt aus dem Shop von Hans Deloof bezogen, so ist dieses Dokument dennoch anwendbar – es werden dann lediglich andere Angaben zur LocoIO-PIC-Version angezeigt.

¹ Siehe auch Datenblatt des Prozessors PIC16F873

Hinweis zum Programm LocoHDL und dem USB-LocoBuffer

Das Programm LocoHDL reagiert empfindlich darauf, wenn der über die USB-Schnittstelle angeschlossene LocoBuffer kurze Spannungseinbrüche hat bzw. kurz von der Schnittstelle entfernt wird. Oftmals reicht dann noch nicht einmal der Neustart des Programmes LocoHDL und es muss der PC neu gestartet werden.

Nach Neustart des LocoBuffer (z.B. durch Aus- und Einschalten) oder abziehen des USB-Kabels ist die benutzte serielle Schnittstelle über USB nicht mehr gültig, da sich der LocoBuffer über USB nach dem Einschalten neu am System anmeldet. In diesem Fall ist das Filehandle (des Steuerprogramms, z.B. LocoHDL) für die geöffnete Schnittstelle unbrauchbar (es können keine Zeichen mehr versandt werden und es wird natürlich nichts mehr empfangen). Das kann zum "Hängenbleiben" des Steuerprogramms z.B. LocoHDL führen.



Wenn man am USB-LocoBuffer V3.0 den Jumper JP2 auf Stellung 2-3 stellt, erfolgt die Spannungsversorgung für die PICs auf dem LocoBuffer über die USB-Schnittstelle, Einflüsse der 12V-Versorgung spielen dann keine Rolle (... aber sehr wohl das Trennen des LocoBuffer vom PC...)

Verschiedene LocoHDL-Versionen

Eine mit LocoHDL 3.5.2 gespeicherte SV-Datei ist mit der neuen Version 4.0.6.29² nicht kompatibel, das habe ich so getestet:

*Version 4.0.6.29 kann die alten Dateien lesen
Version 3.5.2 kann die neuen Dateien nicht lesen.*

Mögliche Abhilfe: In der neuen SV-Datei³ steht in der ersten Zeile z.B. [177](#), [123](#). Löscht man diese Zeile, kann die so geänderte SV-Datei auch mit der alten Version gelesen werden (Angabe erfolgt ohne Gewähr, eine Änderung erfolgt auf eigenes Risiko!).

In LocoHDL Version 4.0.6.29² befindet sich für die Einstellung Festkode für Druckknopf bzw. Wechselnde Kode für Druckknopf ein Programmfehler: nach dem Laden einer SV-Datei oder nach dem Befehl „**Wische**“ ist die Anwahl ausgegraut und nicht mehr änderbar. Hier hilft aktuell nur das erneute Einlesen der Einstellungen direkt aus dem LocoIO.

Hat das LocoIO nur Ausgänge, ist eine falsche Einstellung eigentlich kein Problem, wird jedoch ein Ausgang zu einem Eingang umkonfiguriert und die Einstellung kann nicht auf Festkode für Druckknopf geändert werden, ist das für den betroffenen Eingang sehr wohl ein Problem!

Fehlerhafte Einstellungen sollten zudem nicht mit „**Wische**“ zurückgesetzt werden und müssen entweder

- erneut aus dem LocoIO eingelesen werden
- oder nach einem Programmneustart neu erstellt werden

Mögliche Abhilfe: In der SV-Datei³ steht in der Zeile zu SV 0⁴ z.B. [0](#), [2](#). Vom Wert der SV 0 (hier [2](#)) subtrahiert man den Wert [2](#) und trägt das Resultat als neuen Wert ein, im Beispiel also [0](#), [0](#). So wird beim Einlesen die Einstellung auf Festkode für Druckknopf geändert, die Einstellungen selbst bleiben auch nach dem erneuten Einlesen der geänderten SV-Datei in der Oberfläche ausgegraut. Diese Lösung ist daher nicht sinnvoll (höchstens temporär) nutzbar (Angabe erfolgt ohne Gewähr, eine Änderung erfolgt auf eigenes Risiko!).

² betroffen sind wahrscheinlich auch frühere Versionen, nicht jedoch Version 3.5.2 (das habe ich getestet)

³ Eine Änderung der SV-Datei wird mit einem einfachen Texteditor durchgeführt, z.B. mit dem Editor von Windows, mit Notepad o.ä. *Es ist unbedingt ratsam, vor einer Dateiänderung eine Sicherungskopie der zu ändernden Datei zu erstellen!*

⁴ Je nach Version der SV-Datei ist das die zweite oder dritte Zeile innerhalb der Datei. Die Zeile für SV 0 fängt auf jeden Fall mit [0](#), an.

Begriffe bei Hans Deloop – der Versuch einer Erläuterung

Einige Begriffe in der deutschen Oberfläche bzw. Anleitung von LocoHDL sind „gewöhnungsbedürftig“, daher versuche ich mich hier an einer *Klarstellung*:

Begriff	Erklärung
Pforte	Port
Pinne	Pin
Aktiv Lage	Active low
Aktiv Hohe	Active high
Verspätung	Delay
Druckknopf	Push button
Erdungskontakt	
Festkontakt	Fix contact
Wischen	Clear
Boden	GND (Masse)
Frequenz	Rate

Anschluss, entweder Eingang oder Ausgang
benötigt immer eine Adresse, die bei einem:
- Eingang die „Sensor“-Adresse
- Ausgang eine „Aktor“-Adresse ist.

Eingang schaltet nach GND (Masse)
meist ist zusätzlich ein externer⁵ Pull-Up-Widerstand z.B. 4,7kOhm erforderlich

Der Taster kann auch ein Relaiskontakt, Lichtschranke o.ä. sein...

Eingang schaltet nach +5V
der Pull-Down-Widerstand z.B. 4,7kOhm muss extern⁶ angebracht werden

Der Taster kann auch ein Relaiskontakt, Lichtschranke o.ä. sein...

Verzögerung, verzögert

Taster

Eingang schaltet nach GND (Masse)

Dauersignal
(im Gegensatz dazu:
Pulskontakt = oder

Setzt alle Eingabefelder auf einen Grundwert zurück bzw. löscht alle Ausgaben in einem Ausgabefenster.

Hier ist Vorsicht geboten:
Dieser Tastendruck sperrt aktuell die Eingabe Festcode / Wechselnde Kode für Druckknopf (siehe [hier](#))

GND (Masse)

Einstellung der Blinktaktrate:
0 = schnellster Takt (ca. 1Hz)
15 = langsamster Takt (ca. 0,25Hz)
Dieser Wert gilt für das gesamte LocoIO.

⁵ Auf den Platinen von Hans Deloop sind mir R9 und R10 Pull-Up-Widerstände vorgesehen, bei der Platine wLocoIO-2 müssen diese bei Bedarf extern angeschlossen werden.

⁶ Dann sind die Pull-Up-Widerstände R9 und R10 auf der Platine von Hans Deloop zu entfernen! Bei einem Mischbetrieb mit Aktiv Lage / Aktiv Hohe sind dann ggf. einzelne Pull-Up-Widerstände hinzuzufügen

Adresse	Address	<p><u>Hier ist Vorsicht geboten:</u></p> <p>In der Bedienoberfläche bezeichnet Adresse</p> <ul style="list-style-type: none">○ zum einen die Sensor- bzw. Aktoradresse○ zum anderen die Moduladresse
<u>Wechselnde Kode</u>	Alternating Code for Push Buttons	<p>mit jeder Tasterbetätigung am Eingang ändert der zugehörige Ausgang seinen Zustand:</p> <ul style="list-style-type: none">○ erster Tastendruck: Ausgang <i>Ein</i>○ zweiter Tastendruck: Ausgang <i>Aus</i>○ dritter Tastendruck: Ausgang <i>Ein</i>○ vierter Tastendruck: Ausgang <i>Aus</i>○ usw.

Konfigurieren – was heißt das?

Vor der Verwendung eines jeden LocoIO ist dieses zu konfigurieren (und damit ist weder das Aufspielen der Software für den PIC-Prozessor = „flashen“ oder „Brennen der Betriebssoftware“ noch das „Initialisieren“ gemeint):

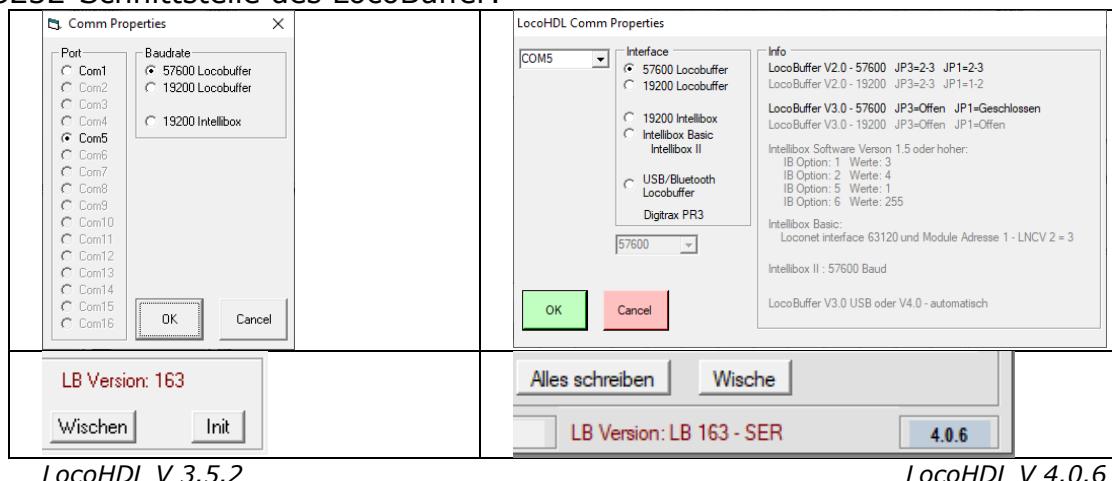
Konfigurieren bedeutet: jedem Anschluss wird gesagt, was er zu tun hat und wie er sich zu verhalten hat.

Ein LocoIO lässt sich auf zwei Arten konfigurieren:

- entweder direkt mit dem Programm [LocoHDL](#)
<https://locohdl.synology.me/LocoHDL/LocoHDL.zip> (aktuell: V4.0.6.29),
<https://locohdl.synology.me/LocoHDL%20Configuration%20DE.pdf>.

Mit LocoHDL wird als Schnittstelle zum LocoNET® unbedingt der LocoBuffer von Hans Deloof <https://locohdl.synology.me/pageDE9.html> benötigt.

Bei mir kommt der LocoBuffer in der Hardware Version V2.3 mit der Software Version **163** zum Einsatz⁷ – mit einem USB-Adapter zum Anschluss an die RS232-Schnittstelle des LocoBuffer:



- oder mit [RocRail](#) über das Menü Programmieren → LocoNet → LocoIO

Meine LocoIO haben im PIC die Software Version **1.48**, die Konfiguration der LocoIO erfolgt mit LocoHDL Version **3.5.2** (hiermit wurden auch die Screenshots erstellt – sofern nichts anderes angegeben). Neuere bzw. andere LocoHDL-Versionen wurden nicht (ausgiebig) getestet: es empfiehlt sich also, diese Kombination (Software Version **1.48** und LocoHDL Version **3.5.2**)⁸ auch so zusammen zu verwenden. Die Screenshots aus RocRail wurden mit der RocRail-Version **13775** erstellt.

Konfiguration des LocoBuffer in RocRail

Die Konfiguration des LocoBuffer in RocRail habe ich bereits im Dokument „*OpenDCC – Zusammenfassung zum Bau.pdf*“
<https://github.com/Kruemelbahn/OpenDCC/blob/main/Documentation/OpenDCC%20-%20Zusammenfassung%20zum%20Bau.pdf> beschrieben – bitte dort nachlesen.

⁷ Andere LocoBuffer-Versionen und -Arten wurden nicht getestet; je nach LocoBuffer-Bauart besitzen diese eine COM-, USB- oder Bluetooth-Schnittstelle

⁸ Höchste PIC-Version für LocoHDL **3.5.2** ist PIC-Version **1.48**, somit passt diese Kombination

Sinnvolle Reihenfolge – oder was ist zu tun?

- LocoIO anschließen (siehe [Anschließen des LocoBuffer und des LocoIO-Modules](#))
- Wenn bisher noch nicht geschehen: [Initialisierung der LocoIO-Module](#), zusammen mit Angabe der [Moduladresse](#).
- Jeden Anschluss einzeln konfigurieren:
Es sind zwei Angaben je Anschluss gemäß dem Motto „*Was bin ich?*“ und „*Wie heiße ich?*“ erforderlich:
 - ***Was bin ich?***
Auch wenn die Auswahl in einer Spalte (eine Spalte = ein Anschluss [„Pforte“]) viele Möglichkeiten bietet, ist es doch relativ einfach:
ein Anschluss ist entweder ein Eingang oder ein Ausgang. Hierbei sind verschiedene Ein- oder Ausgangsvarianten ausählbar – die Details werden weiter hinten in den einzelnen Abschnitten beschrieben.
 - ***Wie heiße ich?***
Die Adresse des Ein -bzw. Ausgangs, siehe [Sensoradresse](#), standardmäßiger Adressbereich 1...2048 bzw. 1...4096.
- Speichern der Eingaben nicht vergessen. Dies ist nicht für jeden Anschluss einzeln mit der zugehörigen Schaltfläche „**S**“ erforderlich, sondern kann auch für alle Anschlüsse gemeinsam mit der Schaltfläche „**Alles schreiben**“ erfolgen.

Eingaben

In der Eingabemaske gibt es drei Arten von Eingaben:

- **Runde Auswahlflächen** : hier kann innerhalb einer Gruppe (z.B. Spalte) immer nur eine Auswahl getroffen werden (die getroffene Auswahl ist erkennbar an einem Punkt:), jede andere Auswahl löscht die vorherige Auswahl
- **Eckige Auswahlflächen** : hier steht jede Auswahlmöglichkeit für sich, es können in einer Gruppe (z.B. Spalte) auch mehrere Auswahlen aktiv sein, aktive Auswahlen erkannt man am Haken .
- **Eingabefelder** **1**: hier werden Zahlen (= Adressen) eingegeben. Adressen können nur Zahlen sein: 0, 1, 2, 3, 4, 5, 6, 7, 8 und 9. Bitte 0 (Null) nicht mit dem Buchstaben ,o' oder ,O' verwechseln!

Ist eine Auswahl oder Eingabe nicht möglich, so ist die entsprechende Auswahl oder Eingabe ausgegraut, fehlerhafte Eingaben werden dem Anwender durch ein farbiges Feld innerhalb der Spalte angezeigt:

Farben von SV Register	
	Richtige SV Werte in PIC
	Richtige SV Werte geschrieben in PIC
	Falsche SV Werte in PIC
	Änderte SV Werte
	Richtige SV Werte von Bestand
	Falsche SV Werte
	Falsche SV Werte von Bestand
	Nicht verwendeter Kode in SV
	Doppeltes adresse für Servo in diesem Modul
	Nicht benutzte Pforte Adresse

Anschließen des LocoBuffer und des LocoIO-Moduls

1. den LocoBuffer
 - ❖ an 12V-Spannungsversorgung anschließen
 - ❖ und mit dem PC verbinden (das Programm LocoHDL wurde noch nicht gestartet!)
 2. das Programm LocoHDL starten
 - ❖ prüfen, ob der LocoBuffer von dem Programm erkannt wurde, d.h.:
 - es gibt keine Fehlermeldung beim Starten von LocoHDL
 - unten rechts im Fenster wird die Softwareversion des LocoBuffer angezeigt
- ACHTUNG**

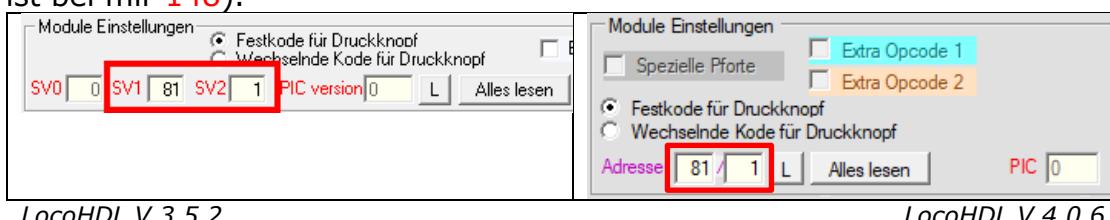
Solange der LocoBuffer von dem Programm LocoHDL nicht erkannt wird, kann ein LocoIO weder initialisiert noch konfiguriert werden und wir brauchen hier nicht weiterzumachen.

→ ggf. LocoHDL oder – wenn das nicht hilft – PC neustarten und zurück zu Schritt 1!
3. jetzt das zu initialisierende LocoIO
 - ❖ an – möglichst eine andere⁹ – 12V-Spannungsversorgung anschließen
 - ❖ über ein LocoNET®-Kabel (RJ12 an RJ12) das LocoIO mit dem LocoBuffer verbinden

Initialisierung des LocoIO-Moduls

Zusammenfassung (bzw. schnelle Übersicht):

- Für die allererste Initialisierung (Grundeinstellung und Einstellen der LocoIO-Moduladresse) darf nur der zur Initialisierung vorgesehene LocoIO am LocoNET® angeschlossen sein!
Hat ein LocoIO eine eigene eindeutige Moduladresse, so können zur weiteren Konfiguration auch mehrere LocoIO am LocoNET® angeschlossen sein, dann ist aber eine Änderung der LocoIO-Moduladresse (SV1 und SV2) nicht möglich.
- Schaltfläche „Init“ betätigen, den nachfolgenden Dialog mit „Init PIC“ bestätigen
- Standardmäßig ist nach der Initialisierung die Moduladresse „81/1“ gesetzt, die **PIC version** muss angezeigt werden (sie darf dann nicht mehr **0** sein und ist bei mir **148**).



- Als nächstes wird die Moduladresse geändert (siehe nächster Abschnitt)

⁹ Eine andere Spannungsversorgung – unabhängig von der des LocoBuffer – sorgt dafür, dass beim Anschließen eines neuen/anderen LocoIO die Spannungsversorgung des LocoBuffer stabil bleibt und mögliche Störungen an der USB-Schnittstelle verhindert werden. Das Programm LocoHDL reagiert recht empfindlich darauf, wenn der USB-Port kurzzeitig nicht verfügbar ist, während das Programm läuft.

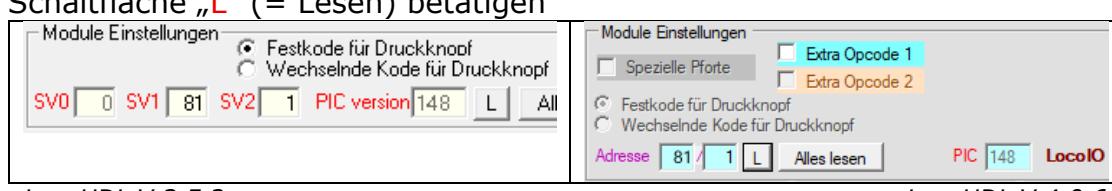
Adresse für das LocoIO-Modul einstellen

Da jedes LocoIO seine eigene Moduladresse bekommen muss, wird diese am besten direkt nach der Initialisierung im Feld **Adresse** (rechts unten, neben der Schaltfläche „**S**“) eingestellt:

- (meine) Basismoduladresse ist immer **60**
Jeder Modulist¹⁰ belegt bitte einen anderen Basismoduladressbereich; werden die LocoIO an einer stationären Anlage betrieben, so kann die Basismoduladresse bei **81** bleiben, lediglich die Submoduladressen müssen für jedes LocoIO unterschiedlich sein.
- Submoduladresse beginnend mit **1**

Bevor überhaupt Werte in das LocoIO geschrieben werden können, muss der zu konfigurierende LocoIO dem Programm LocoHDL bekannt gemacht werden – auch wenn nur ein einziger LocoIO angeschlossen ist:

- Aktuelle Modul-**Adresse** des angeschlossenen LocoIO (hier: **81 / 1**) im Adressfeld links unten (nahe der Schaltfläche „**L**“) eintragen
- Schaltfläche „**L**“ (= Lesen) betätigen

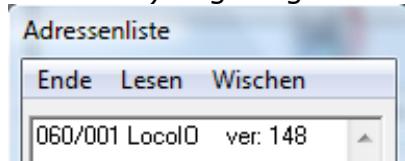


Wenn der LocoIO erkannt wurde, wird die **PIC version** angezeigt → jetzt kann konfiguriert werden

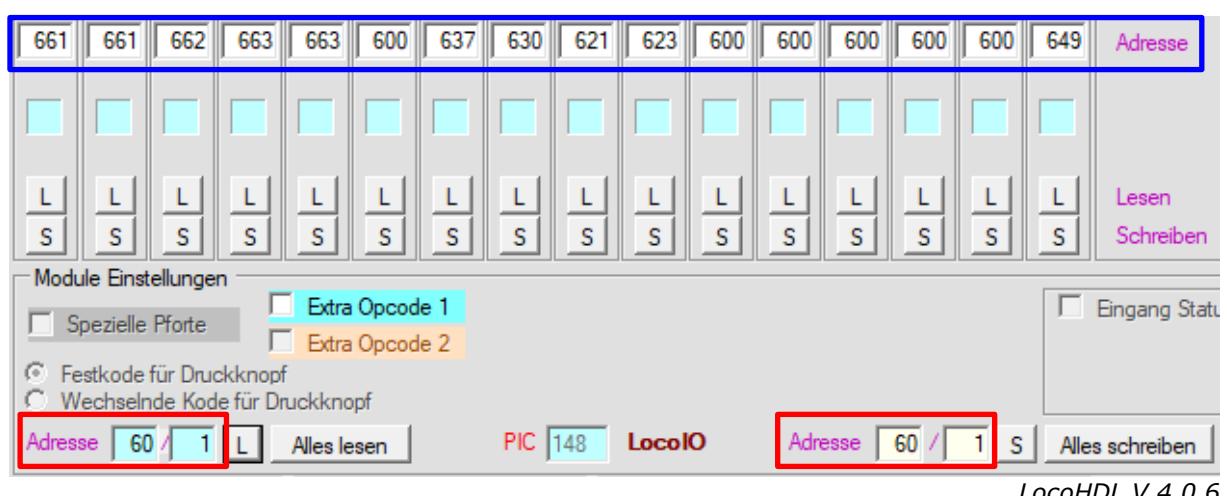
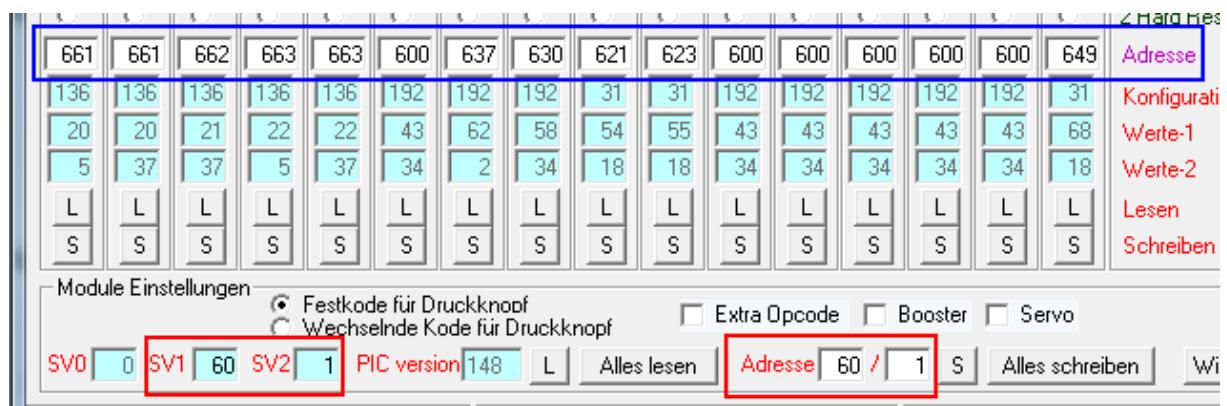
- Neue Modul-**Adresse** (hier: **60 / 1**) im Adressfeld rechts unten (nahe der Schaltfläche „**S**“) eintragen
- Schaltfläche „**S**“ (= Speichern) betätigen.



- Eine Kontrolle kann z.B. mit „**Adressenliste**“ – „**Lesen**“ erfolgen, hier sollte dann als Ausgabe die soeben eingestellte Adresse (z.B. „**060/001** LocoIO ver: 148“) angezeigt werden:



¹⁰ Wenn in diesem Dokument von Modulen bzw. Modulisten die Rede ist, so bezieht sich das darauf, dass ich meine Module zusammen mit anderen Modellbahnherrn in einer Modularanlage betreibe...

Anmerkungen zum Begriff „Adresse“:

- Die LocoIO-Adresse (Moduladresse, **rote** Umrahmungen)
 - hat nichts mit der Sensor¹¹- oder Aktor¹²adresse (**blaue** Umrahmung) zu tun und ist davon völlig unabhängig!
 - wird verwendet, um LocoIO-Module in einem Netzwerk zu identifizieren und zu konfigurieren.
 - LocoIO-Basisadressen (SV1 bzw. der Wert vor dem „/“) liegen im Bereich von 1...79 bzw. 81...127 (die Adresse 80 ist für den LocoBuffer reserviert!)
 - LocoIO-Subadressen (SV2 bzw. der Wert hinter dem „/“) liegen im Bereich von 1...126.
 - Innerhalb eines LocoNET® bzw. an einer (Modul)Anlage müssen alle LocoIO unterschiedliche Moduladressen haben, d.h. die Kombination von LocoIO-Basis- und Submoduladresse muss eindeutig sein und darf nur einmal vorkommen (sonst kann ein LocoIO innerhalb eines LocoNET® nicht angesprochen werden. Der Telegrammverkehr selbst wird hierdurch jedoch nicht beeinträchtigt).
 - An einer (Modul)Anlage ist es sinnvoll, die LocoIO-Moduladressen entsprechend vorhandener Anlagenteile und Modulbesitzer strukturiert zu vergeben.

¹¹ Sensor = Schalter, Taster, Relaiskontakt, Rückmeldung einer Lichtschranke usw., allgemein = Eingang

¹² Aktor = LED, Glühlampe, Relais usw., allgemein = Ausgang

- Sensor- und Aktoradressen (blaue Umrahmung)
 - Sensoradressen (Eingänge) liegen im Bereich von 1...2048 (bzw. 4096)
 - Aktoradressen (Ausgänge) liegen im Bereich von 1...2048
 - Sensor- und Aktoradressen werden in den Meldungstelegrammen auf dem LocoNET® zur Identifizierung der angeschlossenen Sensoren und Aktoren benötigt.
 - Sensor- oder Aktoradressen können mehrfach verwendet werden (gegenseitige Beeinflussung, z.B. mehrere Sensoren schalten den gleichen Ausgang von unterschiedlichen Orten).

Werden jetzt LocoIO in einer Modulanlage an einem gemeinsamen LocoNET® betrieben, ist unbedingt auf die Adressverteilung der Sensoren und Aktoren zu achten, um nicht Aktoren des Kollegen zu beeinflussen oder fremde Sensorbefehle entgegen zu nehmen!

 - Bei einfachen Rückmeldern (Blockkontakte usw.) gibt es eine 1:1-Zuordnung: einem Aktor ist ein Sensor zugeordnet:
 - Ausgang *x* wird von Eingang *x* gesteuert.
 - Einem Aktor (Ausgang) sind bei Signalen, Weichen usw. in der Regel immer zwei Sensoren (Eingänge) zugeordnet:
 - Ausgang 1 wird von Eingang 1 und Eingang 2 gesteuert
 - Ausgang 2 wird von Eingang 3 und Eingang 4 gesteuert

usw.

allgemein gilt dann hier für eine Adresse:

 - Ausgang *x* wird von Eingang $(2*x)-1$ und Eingang $2*x$ gesteuert

Die von mir verwendeten Adressen liegen im Bereich:

- LocoIO-Basismoduladressen 60...69
 - Ausgänge von 600...699 und 1600...1699
 - Eingänge von 600...699 und 1600...1699 bzw. von 1199...1398 und 3199...3398



Beim Einsatz auf einer (stationären) Anlage ist es sinnvoll, sich über eine Aufteilung der Adressen (sowohl für die LocoIO als auch für die Sensoren und Aktoren) Gedanken zu machen – so bleibt die Übersicht erhalten.

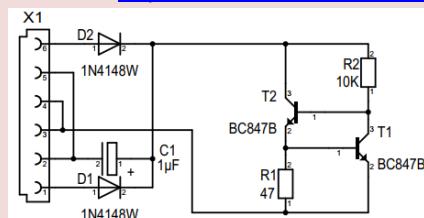


Werden die LocoIO in einem gemeinsamen LocoNET® an einer Modulanlage betrieben und liegen keine modulübergreifenden Gründe vor, dann belegt bitte jeder Modulist¹³ einen anderen Adressbereich sowohl für die LocoIO als auch für die Sensoren und Aktoren!

¹³ Wenn in diesem Dokument von Modulen bzw. Modulisten die Rede ist, so bezieht sich das darauf, dass ich meine Module zusammen mit anderen Modellbahnhern in einer Modulanlage betreibe...



Werden die LocoIO in einem eigenen / separaten LocoNET® betrieben, so ist dafür Sorge zu tragen, dass eine 15mA-Stromeinprägung vorhanden ist; im Zweifelsfall über die nachfolgend dargestellte Schaltung (siehe auch hier: http://dcc-mueller.de/loconet/lpull_d.htm):



Es geht los: LocoIO-Modul anwählen

Vor dem Konfigurieren ist die LocoIO-Moduladresse des LocoIO auszuwählen, das konfiguriert werden soll.

Zur Erinnerung: nach der Initialisierung können alle LocoIO angeschlossen sein, die Grundinitialisierung mit Moduladressvergabe haben wir ja bereits gemacht!

Um zu wissen, welche LocoIO überhaupt im LocoNET® verfügbar sind, verschaffen wir uns einen Überblick:

- mit „**Adressenliste**“ – „**Lesen**“ werden alle LocoIO gesucht und aufgelistet, hier sollte dann als Ausgabe z.B. „**060/001 LocoIO ver: 148**“ angezeigt werden:



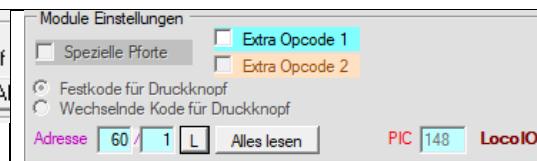
Bei leerer Liste ist dann eine ausgiebige Fehlersuche angesagt...

Bevor überhaupt Werte in das LocoIO geschrieben werden können, muss der zu konfigurierende LocoIO dem Programm LocoHDL bekannt gemacht werden – auch wenn nur ein einziger LocoIO angeschlossen ist, also:

- entweder das gewünschte LocoIO aus der Adressliste mit einem Doppelklick auswählen, das LocoIO-Modul wird so angewählt und direkt ausgelesen
- oder die gewünschte Modul-**Adresse** des angeschlossenen LocoIO (hier: **60 / 1**, standardmäßig nach der Initialisierung – wenn noch keine Moduladresse vergeben wurde: **81 / 1**) im Moduladressfeld links unten (SV1 und SV2 nahe der Schaltfläche „**L**“) eintragen
- **Schaltfläche „Alles lesen“** betätigen



LocoHDL V 3.5.2



LocoHDL V 4.0.6

- Nach kurzer Zeit werden die aktuellen Einstellungen dieses LocoIO angezeigt, auch die **PIC version** – jetzt kann jeder Anschluss konfiguriert / angepasst / geändert werden
- Nach einer Änderung das Speichern nicht vergessen: Schaltfläche „**S**“ am jeweiligen Anschluss bzw. „**Alles Schreiben**“!

Hinweis: Da im LocoNET® jedes LocoIO einen eigene Moduladresse hat, kann eine spätere Adressänderung für einen Sensor oder Aktor auch dann erfolgen, wenn bereits zusätzliche LocoIO am LocoNET® angeschlossen sind.

Vertrauen ist gut – Kontrolle ist besser

Nach einer Konfiguration können natürlich die Status der Eingänge überprüft oder die Ausgänge testweise gesetzt werden.

Eingänge: hier zeigt ein kleines Kästchen oben an jedem Anschluss den Status an:

grau		Status unbekannt
schwarz		Eingang nicht betätigt
gelb		Block Kontakt betätigt
grün		Druckknopf niedrige Adresse betätigt
rot		Druckknopf hohe Adresse betätigt

Ausgänge: können testweise über die Schaltfläche „**Ein**“ bzw. „**Aus**“ geschaltet werden.

Was macht man mit unbenutzten Anschlüssen?

Unbenutzte Anschlüsse erzeugen ggf. unnötige Telegramme, die das LocoNET® belasten bzw. sogar blockieren können!

Daher gilt für nicht verwendete Anschlüsse:

entweder

- als Ausgänge konfigurieren. Dabei verwende ich als Ausgangsadresse hierbei immer den Wert 600.

oder

- den Anschluss als Eingang 'Active low' definieren und dann unbedingt mit einem Pull-Up-Widerstand¹⁴ (z.B. 4,7kOhm) nach +5V verbinden (siehe [hier](#)).

oder

- neuere LocoHDL und PIC-Versionen ermöglichen auch die Auswahl „**Pforte nicht benutzt**“:



¹⁴ Auf den Platinen von Hans Deloof sind mir R9 und R10 Pull-Up-Widerstände vorgesehen, bei der Platine wLocoIO-2 müssen diese bei Bedarf extern angeschlossen werden.

Einen Ausgang mit einem Eingang steuern - Umschalter

(Umschalter und Festkontakteausgang)

(Telegramm: OPC_SW_REQ(B0), Adressbereich: 1...4096)

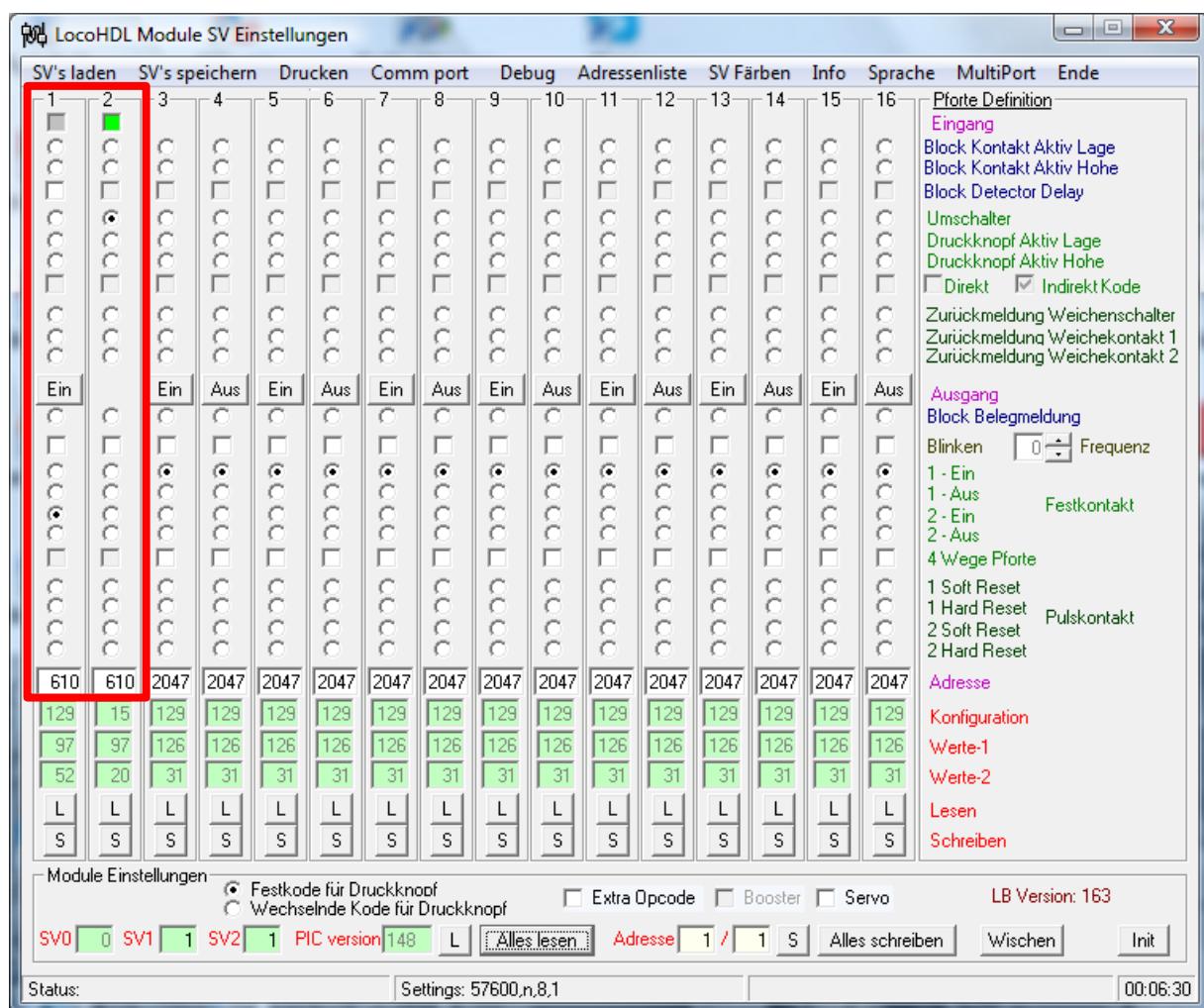
Einen Ausgang, der einem Eingang folgt, kann mit der Funktion Umschalter realisiert werden.

Um dieses Verhalten zu erreichen, wird der Eingang als Umschalter eingestellt, der Ausgang als 1 – Ein, 1 – Aus, 2 – Ein oder 2 – Aus.

Hierbei bezeichnet

- 1 den normalen Ausgang, 2 den zu 1 invertierten Ausgang
- Ein bzw. Aus den Schaltzustand des Ausganges nach dem Einschalten des LocoIO.

Ein- und Ausgang haben hierbei die gleiche Adresse, im nachfolgenden Beispiel wird „610“ verwendet:



Einsatzzweck z.B.:

- Überall da, wo ein Schaltsignal für die Dauer der Betätigung erforderlich ist

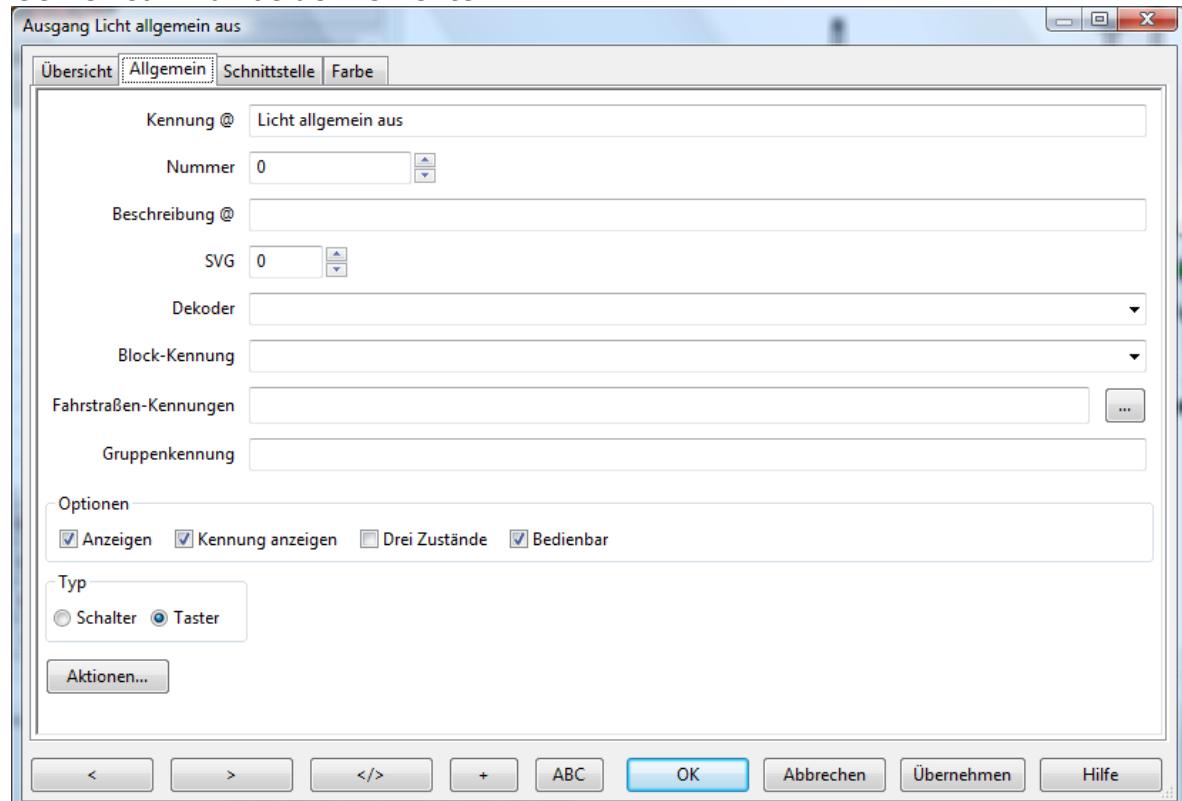
Anmerkungen:

- Soll die Signalwirkung invertiert werden, ist dies nur am Ausgang möglich (z.B. Tausch von 2 – Ein mit 1 – Ein)

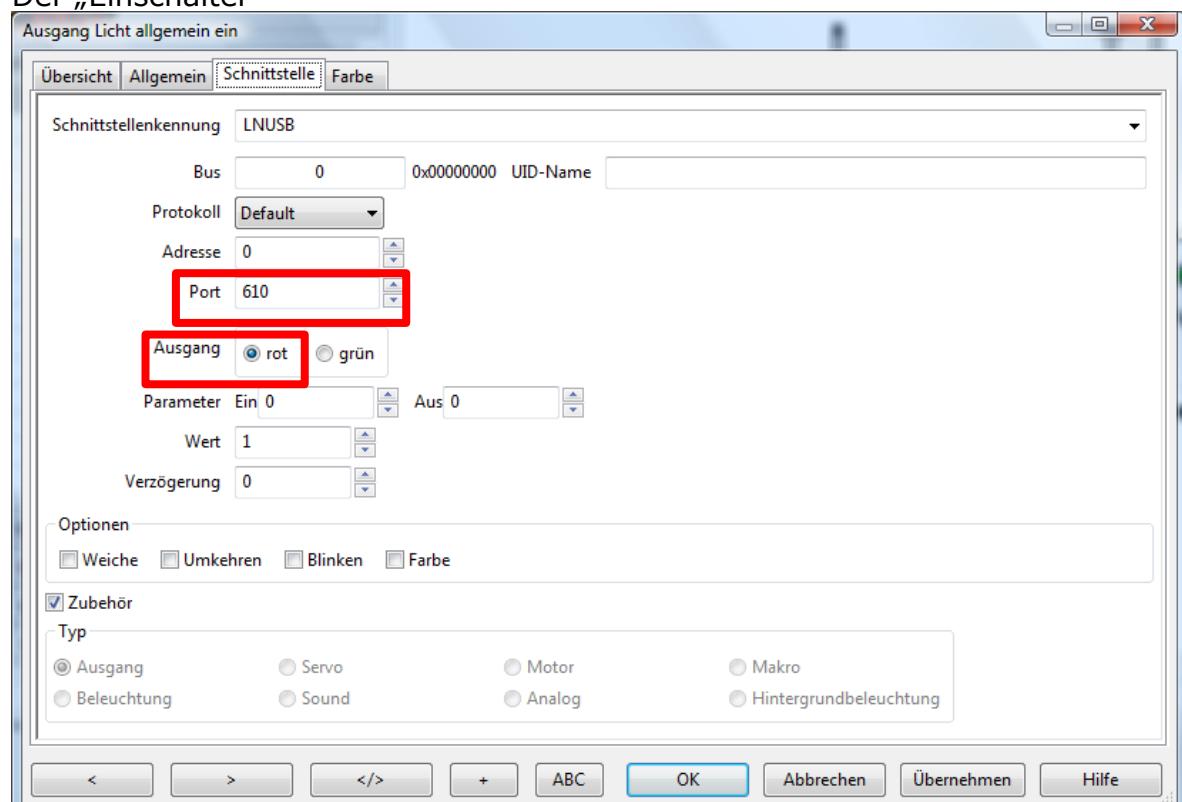
Steuerung mit RocRail

Um einen *Umschalter* in RocRail zu steuern, werden in RocRail zwei „Zubehör“-Elemente projektiert:

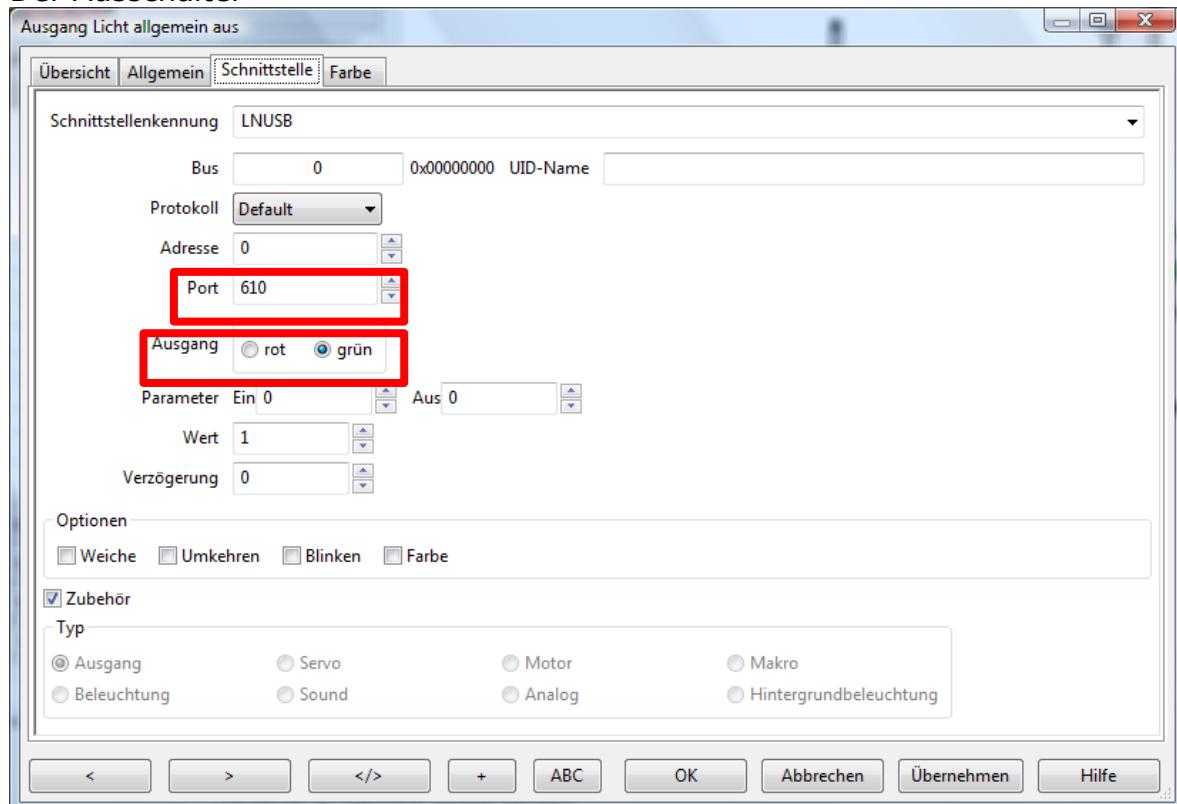
Gemeinsam für beide Elemente:



Der „Einschalter“



Der Ausschalter



Hier ist es wichtig:

- die Schnittstellenkennung des LocoNET® (bei mir: LNUSB) einzutragen
- die Adresse des Rückmelders (z.B.: 610 aus dem obigen Beispiel) einzutragen und
- ein Ausgang auf „rot“ den anderen auf „grün“ zu stellen

Expertenmodus:

- Es werden die LocoNET®-Telegramme B0 gesendet
- Beispiel für Adresse 610:
 - beim Druck auf die Taste / der Betätigung (Einschalten) des Umschalters wird B0 61 14 gesendet (610 - rot)
 - beim Loslassen der Taste / dem Zurückstellen (Ausschalten) des Umschalters wird B0 61 34 gesendet (610 - grün)

Einen Ausgang mit einem Eingang steuern - Blockbelegtmeldung

(Blockdetektor Eingang und Blockbesetzmeldungsausgang)

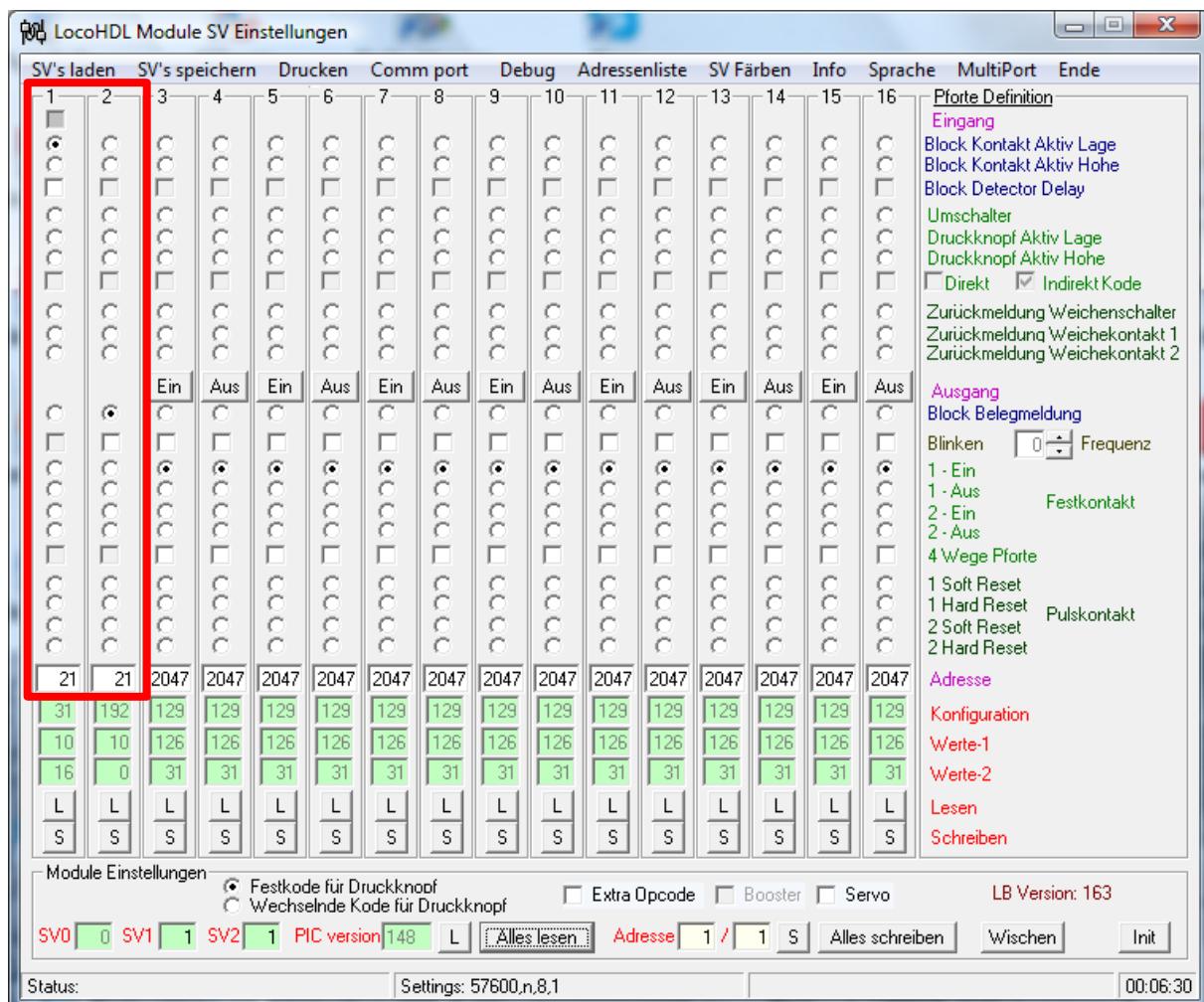
(Telegramm: OPC_INPUT REP(B2), Adressbereich: 1...2048)

Eine weitere Möglichkeit, wie ein Ausgang einem Eingang folgt, ist die Verwendung der **Blockbelegtmeldung**:

solange der Eingang betätigt ist, ist auch der Ausgang eingeschaltet.

Um dieses Verhalten zu erreichen, wird der Eingang als Block Kontakt eingestellt, der Ausgang als Block Belegmeldung.

Ein- und Ausgang haben hierbei die gleiche Adresse, im nachfolgenden Beispiel wird „21“ verwendet:



Einsatzzweck z.B.:

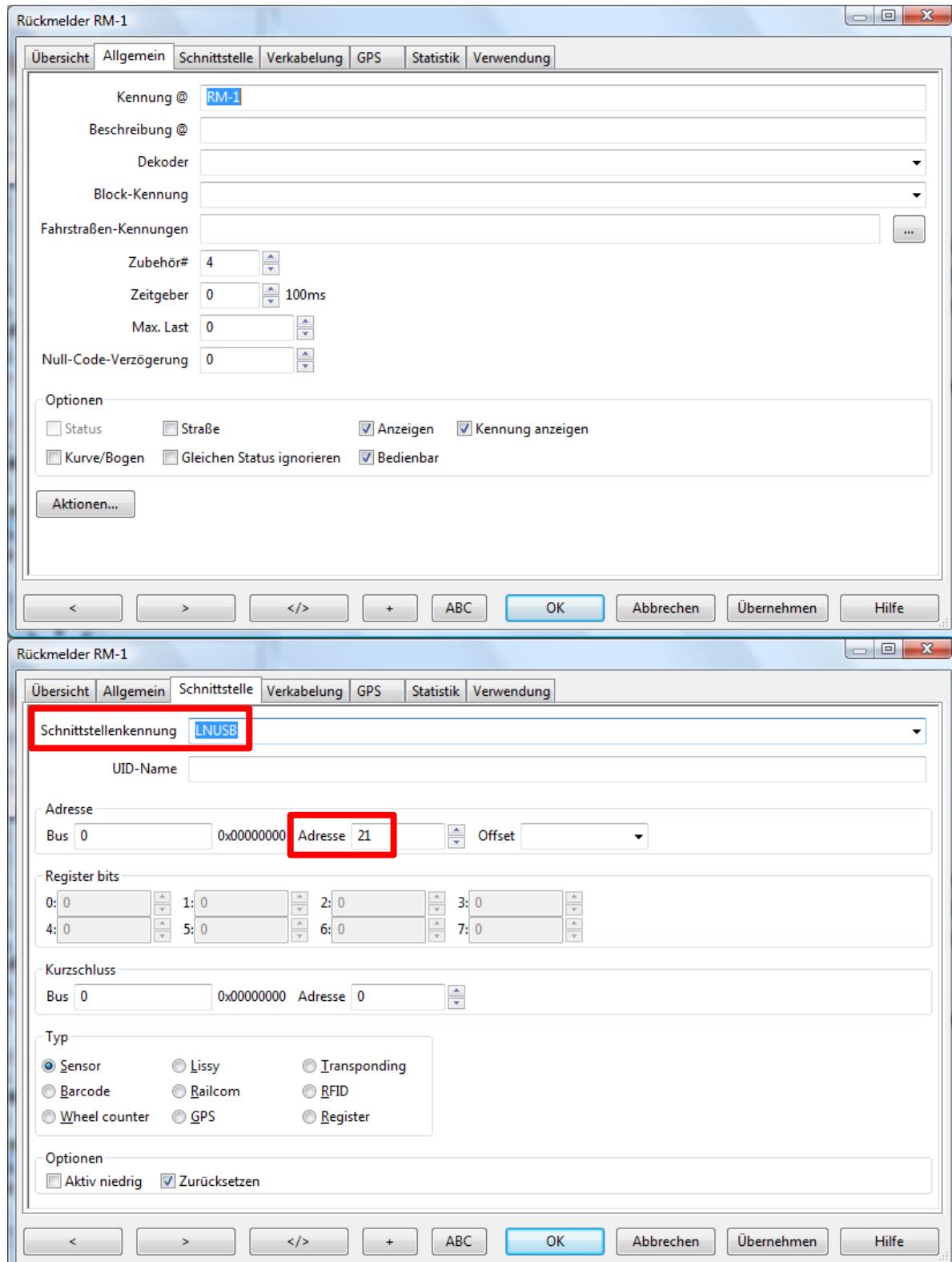
- Positionserkennung eines Zuges

Anmerkungen:

- Block Belegmeldung reagiert nur auf Block Kontakt (bzw. Block Kontakt steuert nur Block Belegmeldung).
- Soll die Signalwirkung invertiert werden, ist dies nur am Eingang möglich (Tausch von Block Kontakt Aktiv Lage mit Block Kontakt Aktiv Hohe)

Anzeige in RocRail

Um den aktuellen Zustand einer *Blockbelegtmeldung* in RocRail zu sehen, wird in RocRail ein „Rückmelder“ projektiert:



Hier ist es wichtig:

- die Schnittstellenkennung des LocoNET® (bei mir: LNUSB) einzutragen und
- die Adresse des Rückmeters (z.B.: 21 aus dem obigen Beispiel) einzutragen

Steuerung mit RocRail

Die Blockbelegtmeldung verwendet das Telegramm: OPC_INPUT REP(B2).
Ich habe es nicht geschafft, ein Element so im Gleisplan zu platzieren und zu konfigurieren, dass z.B. das Einschalten einer Beleuchtung möglich ist.

→ Hier ist demzufolge ein **Umschalter** zu verwenden!

Wechselnde Kode für Druckknopf

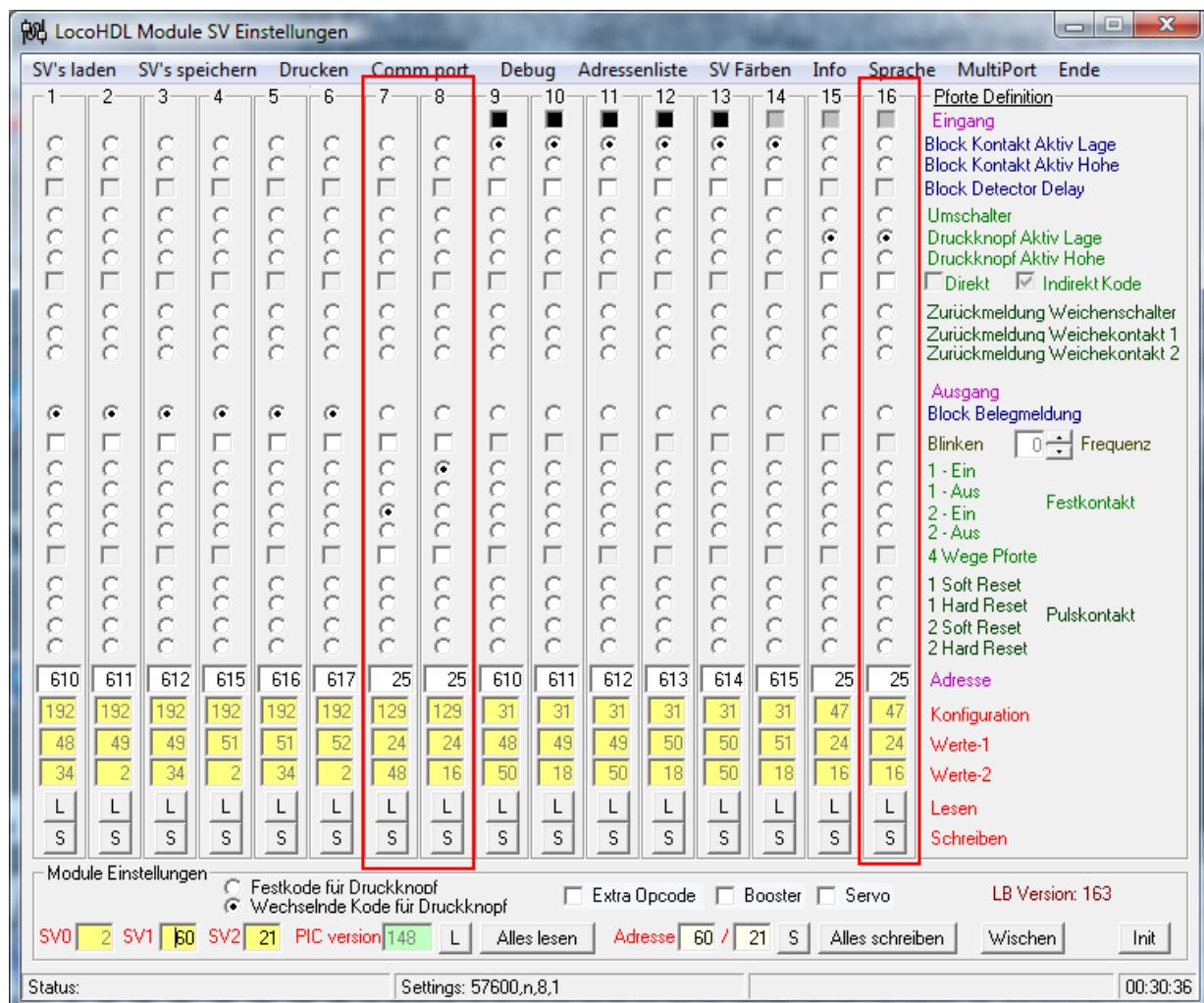
(Druckknopf Aktiv Lage und Druckknopf Aktiv Hohe)

(Telegramm: OPC_SW_REQ(B0), Adressbereich: 1...4096)

Hierbei wird ein Ausgang von einem Eingang gesteuert. Dieser Eingang schaltet bei der ersten Betätigung den Ausgang ein und bei der nächsten Betätigung wieder aus: der Ausgang wechselt somit mit jeder Tasterbetätigung am Eingang seinen Zustand (auch bekannt als Nachttischlampenschaltung):

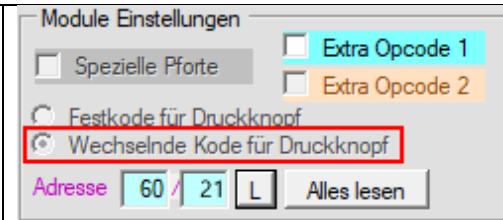
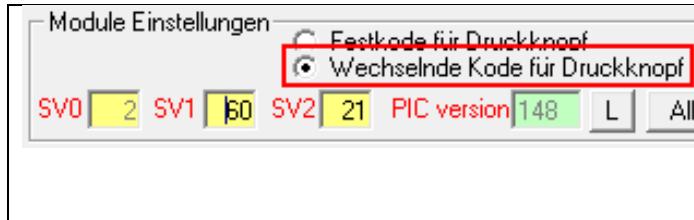
- erster Tastendruck: Ausgang Ein
- zweiter Tastendruck: Ausgang Aus
- dritter Tastendruck: Ausgang Ein
- vierter Tastendruck: Ausgang Aus
- usw.

Im unteren Beispiel schaltet Eingang 16 die Ausgänge 7 und 8 die jeweils umgekehrt zueinander schalten (wenn Ausgang 7 ein ist, ist Ausgang 8 aus und umgekehrt).



Wichtig:

Es muss die Einstellung Wechselnde Kode für Druckknopf aktiviert sein:



- Diese Einstellung wirkt auf **ALLE** Druckknopf-Eingänge an diesem LocoIO!
- Diese Einstellung kann nur aktiviert werden, wenn wenigstens ein Anschluss als Eingang definiert wurde.

In LocoHDL Version 4.0.6.29¹⁵ befindet sich für diese Einstellung ein Programmfehler, eine Erläuterung hierzu befindet sich [hier](#).

¹⁵ betroffen sind wahrscheinlich auch frühere Versionen, nicht jedoch Version 3.5.2

Einen oder zwei Ausgänge mit zwei Eingängen steuern

(Druckknopfeingang und Festkontaktausgang)

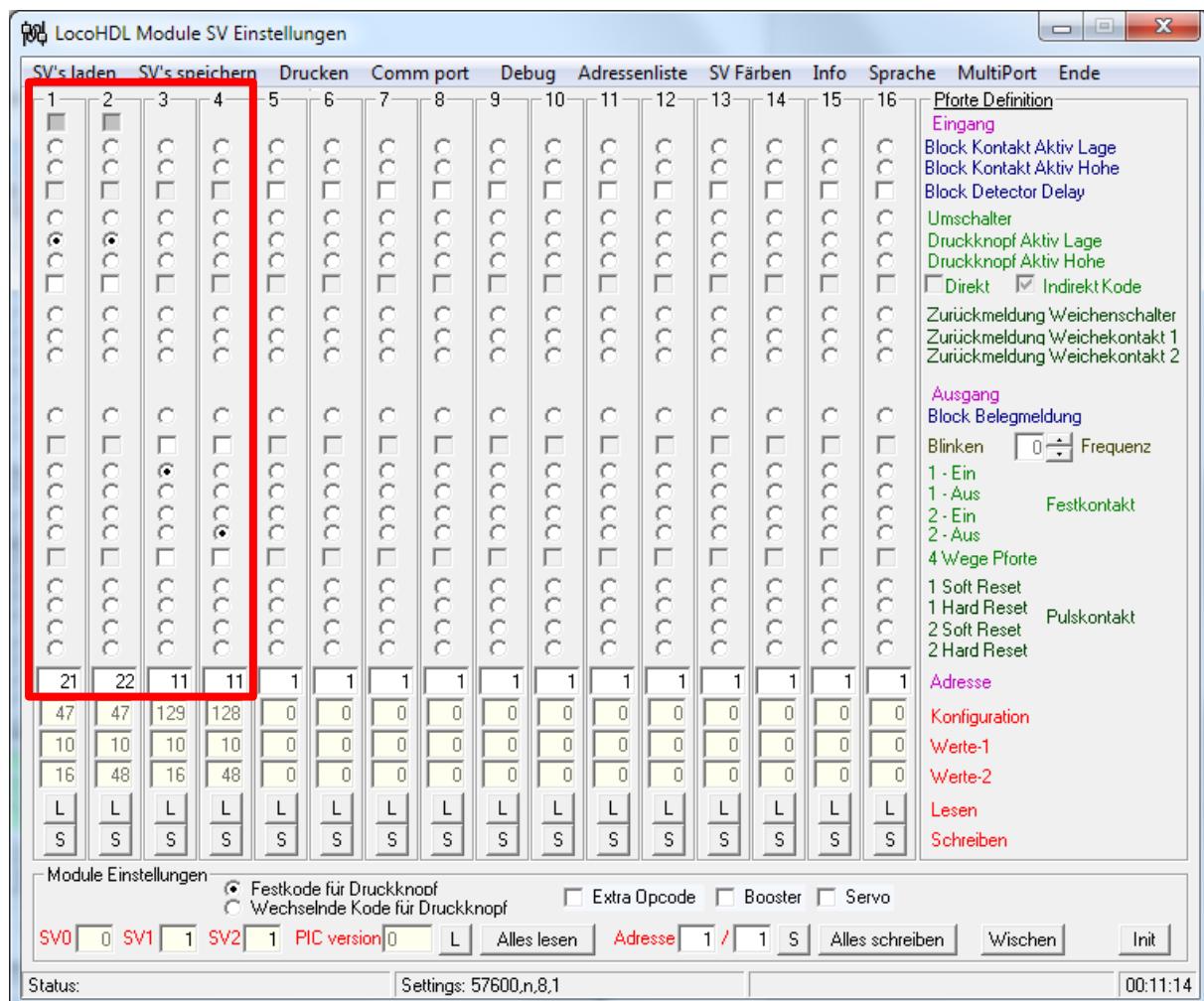
(Telegramm: OPC_SW_REQ(B0), Adressbereich: 1...4096)

Hierbei werden zwei untereinander abhängige Ausgänge von zwei Eingängen betätigt. Dabei schaltet

- Eingang 1 den Ausgang 3 **ein** (und Ausgang 4 **aus**),
- Eingang 2 den Ausgang 3 **aus** (und Ausgang 4 **ein**).

Die niedrigere Eingangsadresse ist hier bei dem Befehl „grün“ bzw. „gerade“ (**Grun/Recht**) zugeordnet, die höhere Eingangsadresse somit dem Befehl „rot“ bzw. „abzweigend“ (**Rot/Rund**).

Die Ausgänge bleiben geschaltet, auch wenn die Eingänge nicht mehr betätigt sind.



Schalttabelle:

Port betätigt	Zustand Port 3	Zustand Port 4
1	1	0
2	0	1

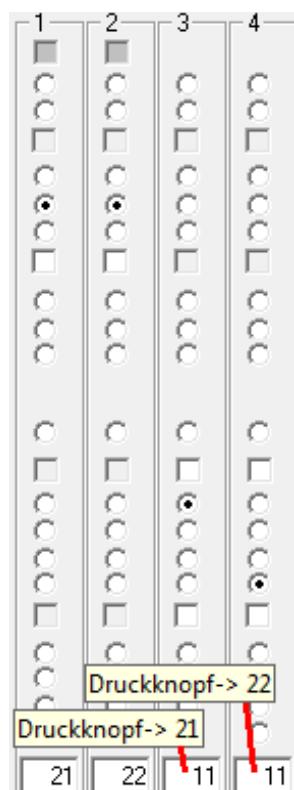
Für diese Anwendung werden immer zwei Eingänge benötigt: Port 1 schaltet Port 3 ein und Port 2 schaltet ihn wieder aus. Mit der Definition für Port 4 programmiert man einen zweiten Ausgang, der sich genau umgekehrt zum ersten Ausgang (Port

3) verhält. Wird ein solcher Ausgang nicht benötigt, ist Port 4 für andere Aufgaben frei verwendbar – dieses Beispiel funktioniert also auch, wenn man nur Port 1, 2 und 3 programmiert.

Einsatzzweck z.B.:

- Weichensteuerung
- Signalsteuerung für zweibegriffige Signale
- Überall da, wo ein Ausgang mit zwei Tastern (oder einem Umschalter) eingeschaltet werden soll

Der Zusammenhang zwischen den Ein- und Ausgangsadressen wird auch im Tooltipp angezeigt: steht der Cursor im Adressfeld eines Ausganges, zeigt der Tooltipp die Adresse des zugehörigen Einganges an:



Verwendung als Impulsausgang (anstelle eines Dauerkontaktes)

Im Beispiel oben bleibt der Ausgang (bzw. die Ausgänge) eingeschaltet, auch wenn die schaltenden Eingänge wieder ausgeschaltet sind (das Eingangssignal wird also quasi gespeichert).

Sind an den Ausgängen Spulenantriebe ohne Endabschaltung angeschlossen, können diese durchbrennen, wenn sie dauerhaft eingeschaltet sind. Dem kann man unter Verwendung eines Impulsausgangs vorbeugen:

- Soft Reset bedeutet: der Ausgang bleibt nur für die Dauer des Tastendruckes eingeschaltet.
- Hard Reset bedeutet: der Ausgang bleibt nur für die Dauer von 1 oder 2 Blinkimpulsen¹⁶ eingeschaltet, unabhängig von der Dauer des

¹⁶ die Schaltdauer des Ausgangs (Dauer des (Blink-)Impulses) hängt von der Blinken Frequenz ab:



Tastendruckes. Diese Einstellung ist dann zu verwenden, wenn das auslösende Ereignis ein Dauersignal erzeugt (z.B. ein Zug löst eine Lichtschranke oder einen Gleiskontakt aus und bleibt dann in der Lichtschranke oder auf dem Gleiskontakt stehen).

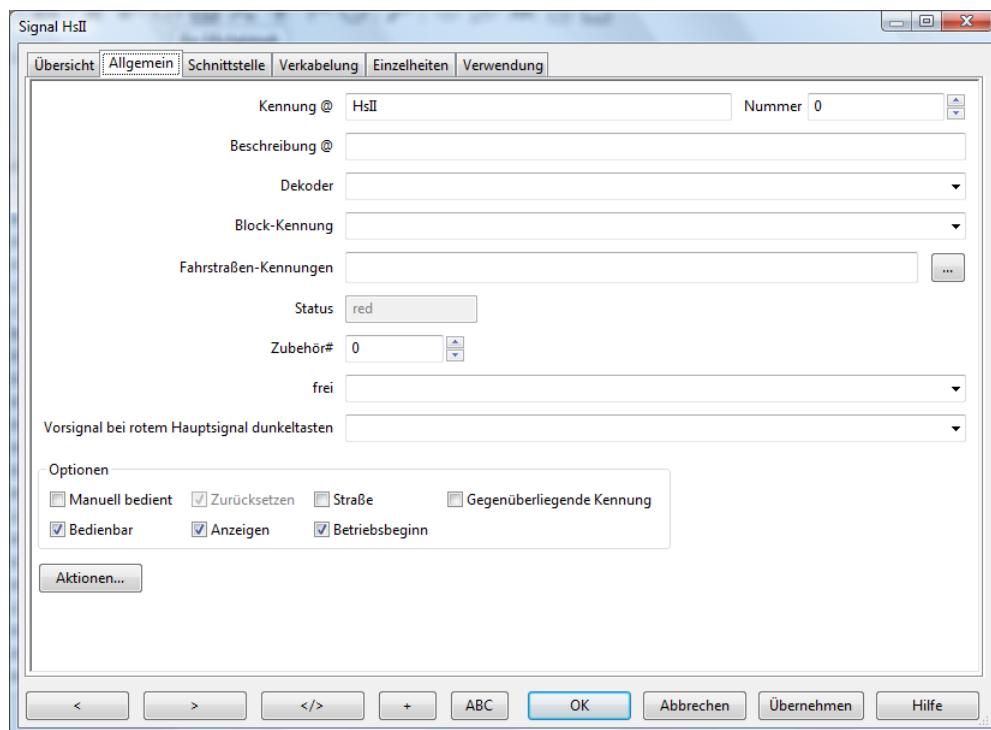
Schalttabelle:

Port betätigt	Zustand Port 3	Zustand Port 4
1		0
2	0	

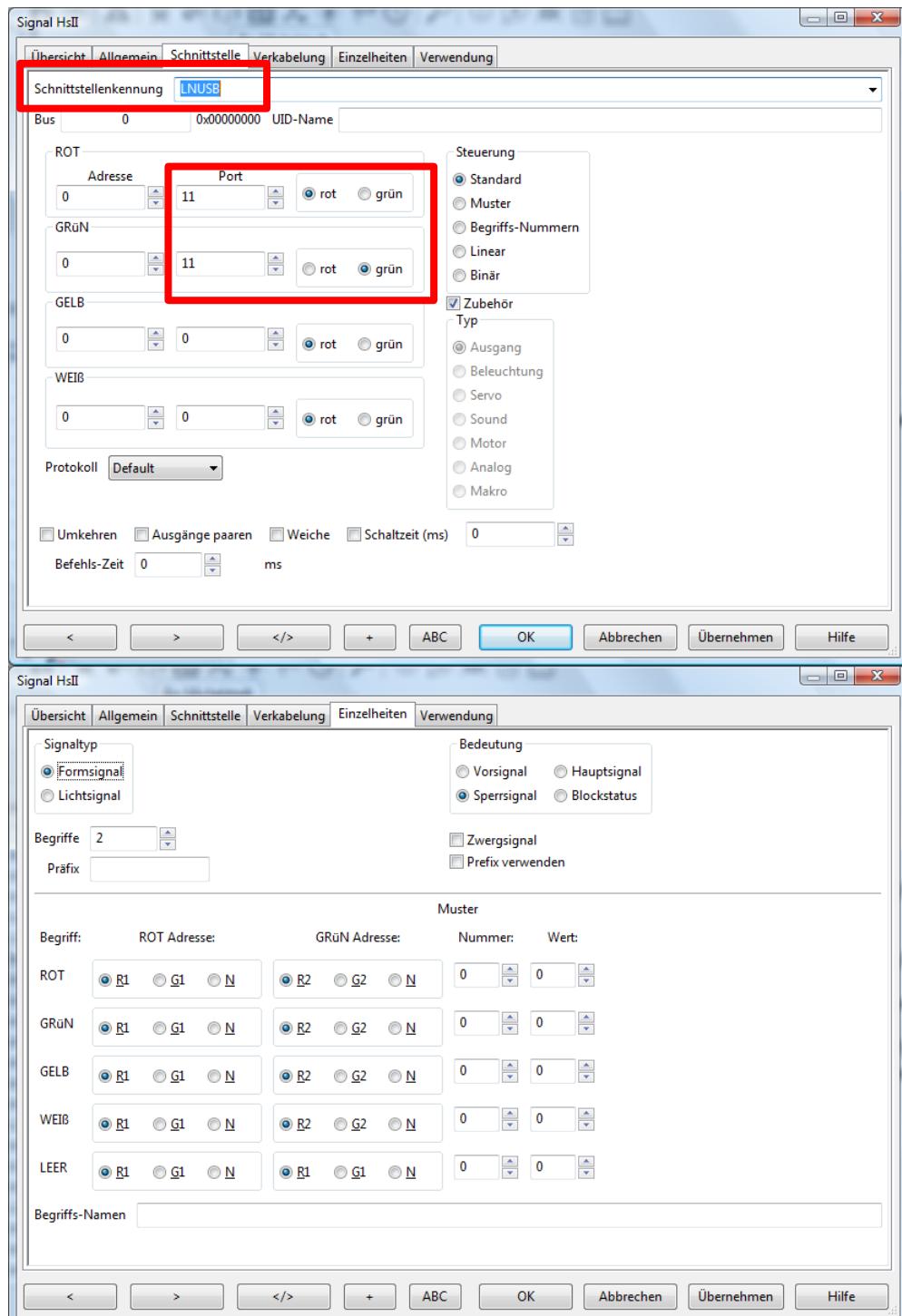
Soft Reset bzw. Hard Reset lässt sich nicht zusammen mit 4 Wege Pforte verwenden.

Steuerung mit RocRail

Um eine Weiche oder ein Signal zu steuern, ist das zu steuernde Element im Gleisplan zu projektieren:



Hier bedeutet 0 die schnellste Frequenz (1Hz) / kürzeste Impulsdauer (0,5s), 15 die niedrigste Frequenz (ca. 0,25Hz) / längste Impulsdauer (2s).



Hier ist es wichtig:

- die Schnittstellenkennung des LocoNET® (bei mir: LNUSB) einzutragen und
- die Adressen (für ROT und GRÜN) des Signals / der Weiche (z.B.: 11 aus dem obigen Beispiel) einzutragen

Hinweis: ein externes Stellen der Weiche / des Signals ändert die Anzeige (**rot** / **grün**) in RocRail nicht!

Steuerung mit dem TwinCenter / der Intellibox

Im Keyboard-Mode können mit dem TwinCenter / der Intellibox ebenfalls Schaltvorgänge für Signale und Weichen durchgeführt werden:

```
lok# |////|lok#
?|////| ?
```

Taste **mode** so oft betätigen, bis:

Keyboard
Mode

wechselt zu

```
lok# |////|lok#
?|////| ?
```

Taste **menu** betätigen

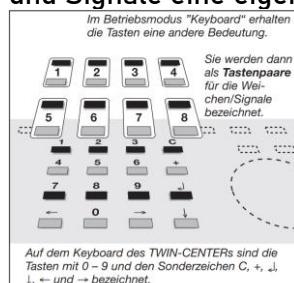
Keyboard
Adr.: - ...

Startadresse (Bereich 1...2000) eingeben und mit
Taste **←** übernehmen
(Abbruch über Taste **menu**)

```
lok# |////|lok#
?|////| ?
```

Es können immer 8 Weichen und Signale gestellt werden
(Startadresse **bis** Startadresse + 8).

Im Keyboard-Modus haben die Tasten für das Stellen von Weichen und Signale eine eigene Bedeutung:



Expertenmodus:

- Es werden die LocoNET®-Telegramme **B0** gesendet
- Beispiel für Adresse 660:
 - beim Druck auf Taste „1 rot“ wird **B0 13 15** gesendet (660 - rot)
 - beim Loslassen der Taste „1 rot“ wird **B0 13 05** gesendet (660 - rot aus)
 - beim Druck auf Taste „1 grün“ (≈ Taste 4) wird **B0 13 35** gesendet (660 - grün)
 - beim Loslassen der Taste „1 grün“ (≈ Taste 4) wird **B0 13 25** gesendet (660 - grün aus)

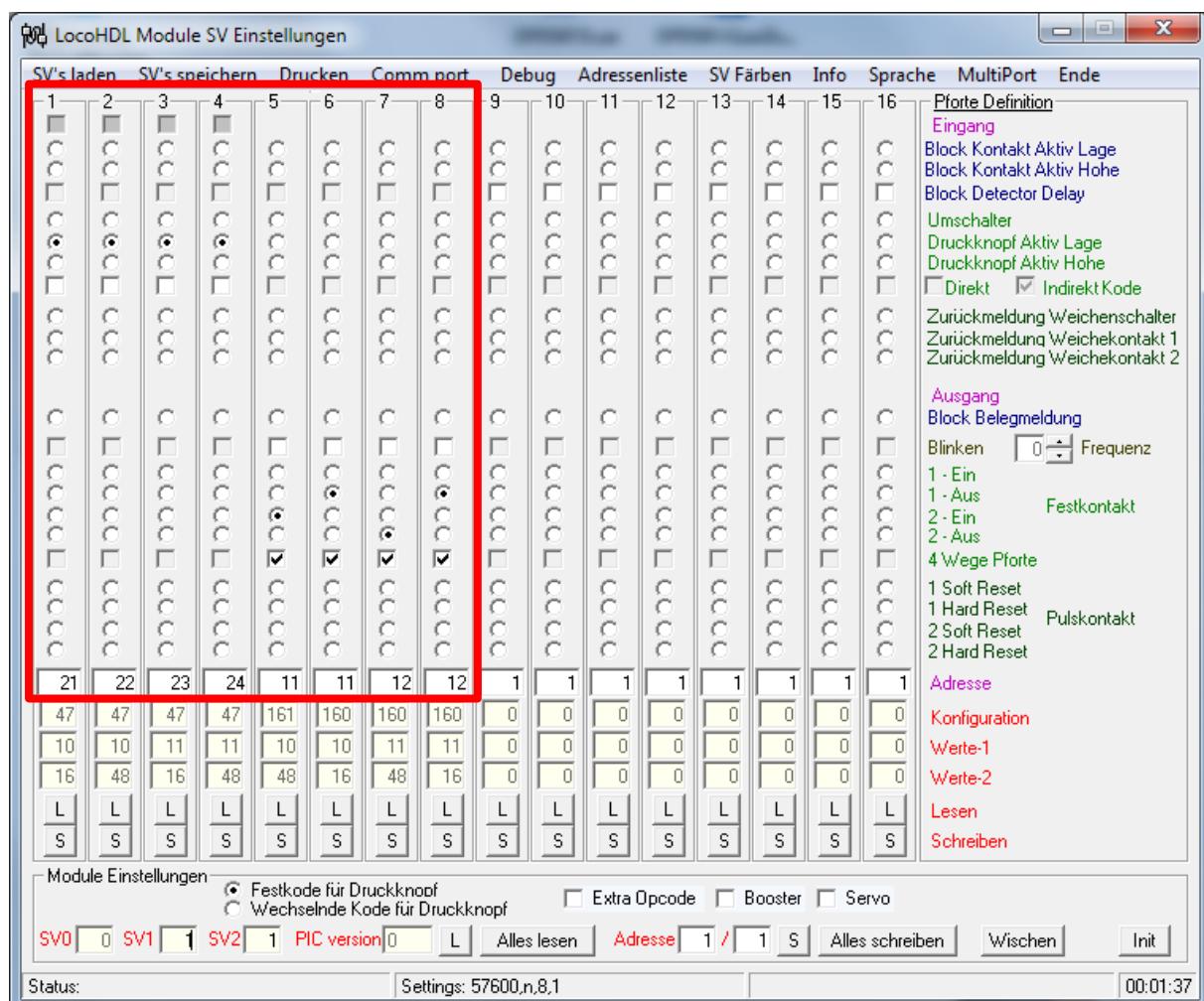
Vier Eingänge – vier Ausgänge

(Druckknopfeingang und Festkontaktausgang)

(Telegramm: OPC_SW_REQ(B0), Adressbereich: 1...4096)

Dies ist im Prinzip die Erweiterung von „Einen oder zwei Ausgänge mit zwei Eingängen steuern“ auf drei bzw. vier zusammengehörige Ein- und Ausgänge.

Hierbei werden vier untereinander abhängige Ausgänge von vier Eingängen betätigt. Dabei schaltet jeder Eingang einen Ausgang aktiv, die anderen Ausgänge werden inaktiv geschaltet.



Schalttabelle:

Port betätigt	Zustand Port 5	Zustand Port 6	Zustand Port 7	Zustand Port 8
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1

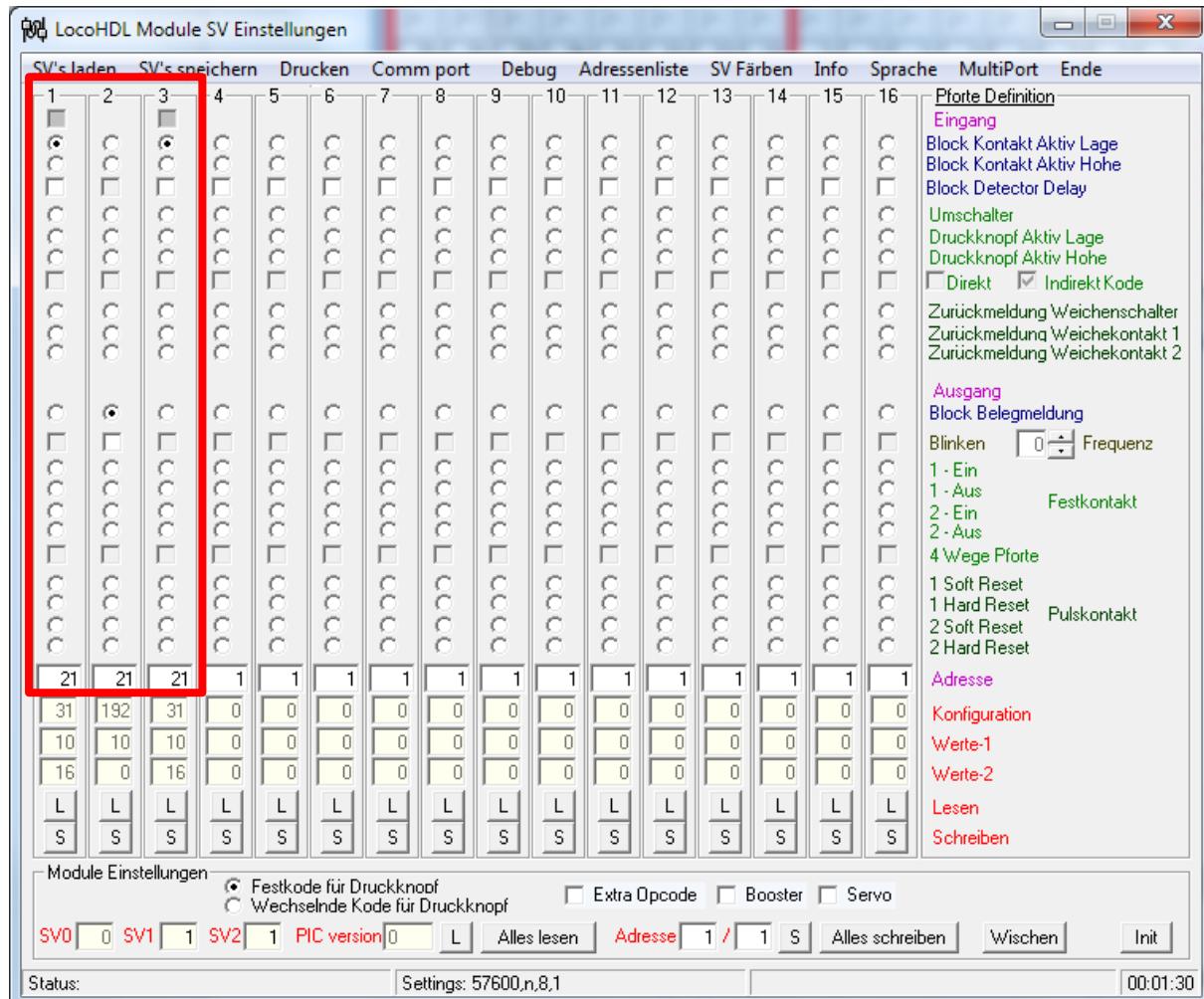
Einsatzzweck z.B.:

- Signalsteuerung für drei- oder vierbegriffige Signale

Geht doch – oder?

Kann man eigentlich einen Ausgang von mehreren Eingängen steuern – oder einen Ausgang vervielfachen?

Man kann - bedingt.



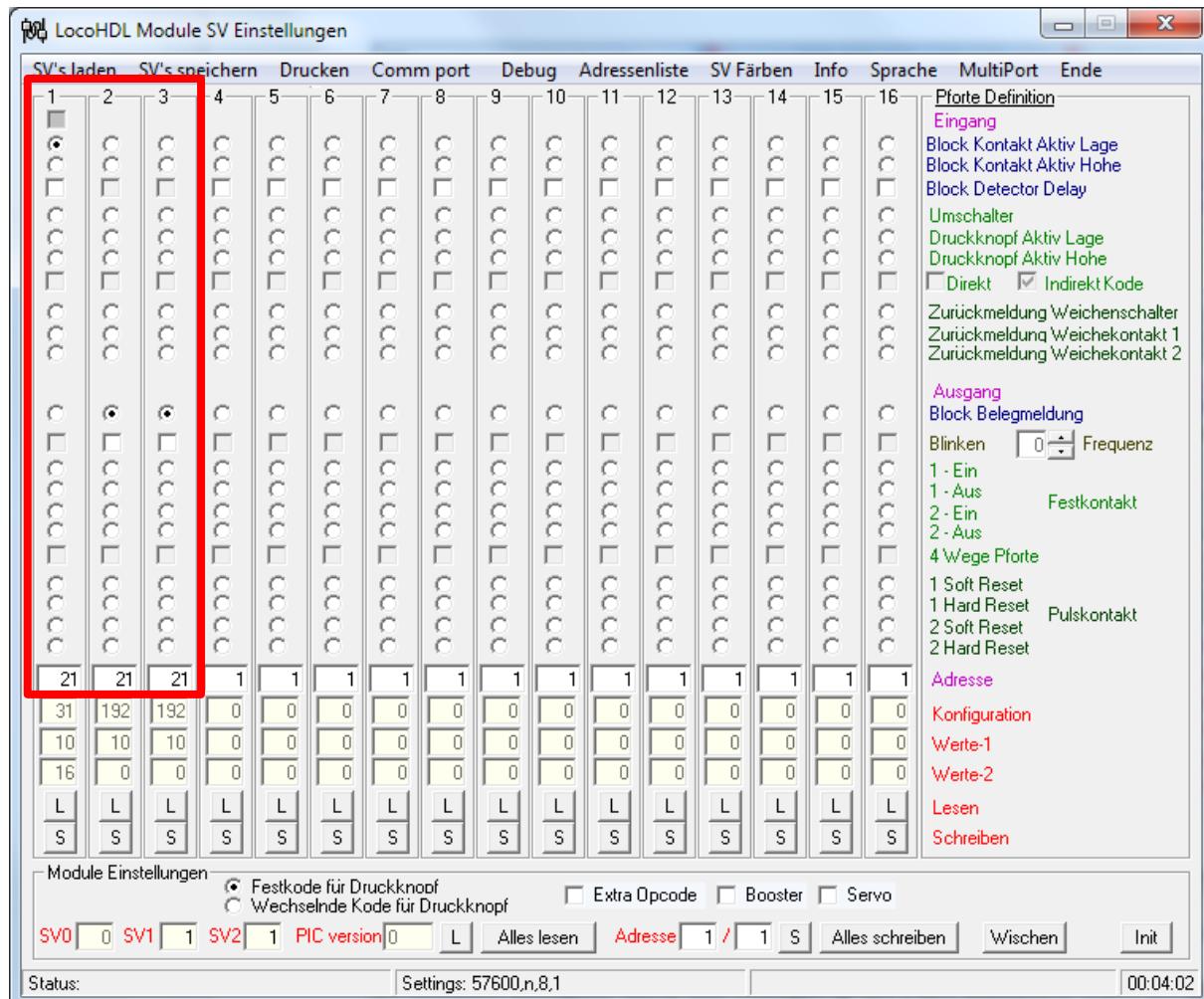
In diesem Beispiel wird der Ausgang an Port 2 sowohl vom Eingang an Port 1 als auch vom Eingang an Port 3 ein- bzw. ausgeschaltet – die beiden Eingänge wirken also wie eine Oder-Verknüpfung auf einen Ausgang.

Einsatzzweck z.B.:

- Anlagenbeleuchtung von zwei verschiedenen Orten mit jeweils einem Schalter einschalten

Anmerkung: Wird Port 2 sowohl von Port 1 als auch von Port 3 eingeschaltet, dann wird Port 2 wieder ausgeschaltet, wenn einer der beiden Ports wieder ausgeschaltet wird.

Was auch geht: ein Eingang steuert zwei Ausgänge gleichzeitig, siehe hier:



In diesem Beispiel werden die Ausgänge an Port 2 und Port 3 vom Eingang an Port 1 ein- bzw. ausgeschaltet – die beiden Ausgänge werden also gleichzeitig vom Eingang Port 1 geschaltet.

Einsatzzweck z.B.:

- Von einem Ort die verteilte Anlagenbeleuchtung einschalten

Module / Schaltungen, die mit LocoIO angesteuert werden können

Einige von mir entwickelte Module / Schaltungen können direkt an ein LocoIO angeschlossen und von diesen gesteuert werden. Aktuell sind dies:

FotoMichel: <https://github.com/Kruemelbahn/FotoMichel>

LichtComputer: <https://github.com/Kruemelbahn/LightControl>

Servo: <https://github.com/Kruemelbahn/Servo>

Servo2: <https://github.com/Kruemelbahn/Servo2>

Signal-Z: <https://github.com/Kruemelbahn/Signal-Z>

Details hierzu sind in den zugehörigen Dokumentationen zu finden.

Konfigurieren mit JMRI

Die LocoIO-Module können auch über das Programm JMRI (<https://www.jmri.org/>) konfiguriert werden. Hierbei ist zwar nach wie vor ein LocoBuffer¹⁷ erforderlich – allerdings nicht unbedingt der von Hans Deloop.

Nachlesen kann man das alles hier:

[Krümelbahn Info 11 - JMRI - Universalwerkzeug für die Modellbahn](#)

Zu guter Letzt ... speichern der Einstellungen

Einstellungen sollen nachvollziehbar bleiben – und nachschlagbar sein. Darum: „SV's speichern“ (in der Menüleiste) nicht vergessen!

Die hierbei erzeugte Datei dient nicht nur zum Nachsehen – bei einem Modul/Prozessortausch können diese einmal gespeicherten Einstellungen geladen und auf das neue Modul gespeichert werden. Die lästige Suche nach dem „wie war das noch gleich bei diesem Anschluss“ entfällt.

Hinweis: Eine mit LocoHDL 3.5.2 gespeicherte SV-Datei ist mit der neuen Version 4.0.6.29¹⁸ nicht kompatibel, siehe auch [hier](#).

Nach dem Speichern erstelle ich zusätzlich einen Screenshot der Einstellungen. Vorteil hierbei: man kann die aktuellen Einstellungen auch ohne LocoHDL nachschlagen.

Eine Dokumentationshilfe – die Adresstabelle

Um eine Übersicht über die Belegung und Verwendung der LocoIO-Adressen zu haben, benutze ich eine Exceltabelle, in der diese Informationen eingetragen sind:

Michelstädter Module			Adresse			Ausgänge			Eingänge				
Beschreibung			Block	rot	grün	Typ	Block	rot	grün	Typ	Block	rot	grün
		Dummy	600			B	✓			B			
Brücke	RM	Brücke Schranken RM	601			B	10-1			B	2-15		
Brücke		Brücke Freigabe Schranken öffnen	602			1E	2-16			U	10-8		
			...										
	Licht	Licht allgemein	630					
			...										
Brücke		Brücke Schranken	660	1319	1320	*S		2-1	2-2	D*		10-15	10-16

¹⁷ Ich verwende z.B. einen LocoBuffer mit einem Arduino-Nano mit zusätzlicher Hard- und Software (<https://github.com/Kruemelbahn/LocoBuffer-Nano>)

¹⁸ betroffen sind wahrscheinlich auch frühere Versionen, nicht jedoch Version 3.5.2 (das habe ich getestet)

Meine Notation:

- Eine Aufteilung in eine dreiteilige „Beschreibung“ erleichtert eine Suche bzw. eine Filterung
- unter „Adresse“ wird die benutzte Sensor- bzw. Aktoradresse eingetragen,
 - Block ist die Adresse bei Verwendung als Block Belegmeldung, Umschalter oder Druckknopf
 - rot bzw. grün die Adresse für die Verwendung bei Signalen oder Weichen
- in den Spalten „Ausgänge“ bzw. „Eingänge“ wird das Modul mit dem verwendeten Anschlussdaten eingetragen. 10-1 bedeutet demnach
 - Modul mit der Basismoduladresse 60, Submoduladresse 10, Anschluss 1
 - B bedeutet bei mir Block Belegmeldung. Die verwendeten Buchstaben und Zahlen leiten sich von der Kurzinfo auf der rechten Dialogseite des LocoHDL-Programms ab.

Wenn es dann zu jedem Ausgang wenigstens einen Eingang gibt, weiß ich, dass das zusammenpasst. Und bei einer Fehlersuche erkenne ich, wo ich suchen muss...

Linkliste

Verwendete Links rund um die LocoIOs:

- Das Original:
<https://web.archive.org/web/20220405130232/http://locobuffer.com/LocoIO/LocoIO.htm>
- LocoHDL von Hans Deloof: <https://locohdl.synology.me/pageDE7.html>
- LocoIO von Hans Deloof: <https://locohdl.synology.me/pageDE8.html>
- LocoBuffer von Hans Deloof: <https://locohdl.synology.me/pageDE9.html>
- wLocoIO-2:
<https://1drv.ms/b/c/a13c984302a24415/ERVEogJDmDwggKHNBQAAAABGX3rws76FhAw2bXBd0WxxA>
- LocoNET®-Spezifikation von Digitrax:
<https://www.digitrax.com/support/loconet/loconetpersonaledition.pdf>
http://embeddedloconet.sourceforge.net/SV_Programming_Messages_v13_PE.pdf
- Die SV-Übersicht:
<http://wiki.rocrail.net/lib/exe/fetch.php?id=loconet-io-de&cache=cache&media=loconet:lio-sw:locoio.pdf>
- Ergänzungen zu Konfiguration und Betrieb (dieses Dokument im WWW):
<https://github.com/Kruemelbahn/LocoIO/blob/main/Documentation/LocoIO-Erg%C3%A4nzungen.pdf>
- Hinweis zum Umgang mit JMRI:
<https://github.com/Kruemelbahn/Infoletter/blob/main/Kr%C3%BCmelbahn%20Info%2011%20-%20JMRI%20-%20Universalwerkzeug%20f%C3%BCr%20die%20Modellbahn.pdf>

LocoIO module software 1.48

LocoIO SV Programmeer overzicht

LocoIO SV tabel

SV	Functie	PIC Pin/JP-Pin	SV	Functie	SV	Functie	
0	LocoIO Set-up						
1	LocoIO Laag Adres						
2	LocoIO Subadres						
3	Configuratie byte poort 1	28	J4-5	51	<OPC> poort 1		
4	Getal-1 - poort 1			52	<ARG1> poort 1		
5	Getal-2 - poort 1			53	<ARG2> poort 1		
6	Configuratie byte poort 2	27	J4-4	54	<OPC> poort 2		
7	Getal-1 - poort 2			55	<ARG1> poort 2		
8	Getal-2 - poort 2			56	<ARG2> poort 2		
9	Configuratie byte poort 3	26	J4-3	57	<OPC> poort 3		
10	Getal-1 - poort 3			58	<ARG1> poort 3		
11	Getal-2 - poort 3			59	<ARG2> poort 3		
12	Configuratie byte poort 4	25	J4-2	60	<OPC> poort 4		
13	Getal-1 - poort 4			61	<ARG1> poort 4		
14	Getal-2 - poort 4			62	<ARG2> poort 4		
15	Configuratie byte poort 5	23	J5-5	63	<OPC> poort 5	101	Servo 1 Pos 1
16	Getal-1 - poort 5			64	<ARG1> poort 5	102	Servo 1 Pos 2
17	Getal-2 - poort 5			65	<ARG2> poort 5	103	Snelheid Servo 1
18	Configuratie byte poort 6	22	J5-4	66	<OPC> poort 6	104	Servo 2 Pos 1
19	Getal-1 - poort 6			67	<ARG1> poort 6	105	Servo 2 Pos 2
20	Getal-2 - poort 6			68	<ARG2> poort 6	106	Snelheid Servo 2
21	Configuratie byte poort 7	21	J5-3	69	<OPC> poort 7	107	Servo 3 Pos 1
22	Getal-1 - poort 7			70	<ARG1> poort 7	108	Servo 3 Pos 2
23	Getal-2 - poort 7			71	<ARG2> poort 7	109	Snelheid Servo 3
24	Configuratie byte poort 8	18	J5-2	72	<OPC> poort 8	110	Servo 4 Pos 1
25	Getal-1 - poort 8			73	<ARG1> poort 8	111	Servo 4 Pos 2
26	Getal-2 - poort 8			74	<ARG2> poort 8	112	Snelheid Servo 4
27	Configuratie byte poort 9	17	J6-5	75	<OPC> poort 9	113	Servo 5 Pos 1
28	Getal-1 - poort 9			76	<ARG1> poort 9	114	Servo 5 Pos 2
29	Getal-2 - poort 9			77	<ARG2> poort 9	115	Snelheid Servo 5
30	Configuratie byte poort 10	16	J6-4	78	<OPC> poort 10	116	Servo 6 Pos 1
31	Getal-1 - poort 10			79	<ARG1> poort 10	117	Servo 6 Pos 2
32	Getal-2 - poort 10			80	<ARG2> poort 10	118	Snelheid Servo 6
33	Configuratie byte poort 11	15	J6-3	81	<OPC> poort 11	119	Servo 7 Pos 1
34	Getal-1 - poort 11			82	<ARG1> poort 11	120	Servo 7 Pos 2
35	Getal-2 - poort 11			83	<ARG2> poort 11	121	Snelheid Servo 7
36	Configuratie byte poort 12	14	J6-2	84	<OPC> poort 12	122	Servo 8 Pos 1
37	Getal-1 - poort 12			85	<ARG1> poort 12	123	Servo 8 Pos 2
38	Getal-2 - poort 12			86	<ARG2> poort 12	124	Snelheid Servo 8
39	Configuratie byte poort 13	13	J7-5	87	<OPC> poort 13		
40	Getal-1 - poort 13			88	<ARG1> poort 13		
41	Getal-2 - poort 13			89	<ARG2> poort 13		
42	Configuratie byte poort 14	12	J7-4	90	<OPC> poort 14		
43	Getal-1 - poort 14			91	<ARG1> poort 14		
44	Getal-2 - poort 14			92	<ARG2> poort 14		
45	Configuratie byte poort 15	11	J7-3	93	<OPC> poort 15		
46	Getal-1 - poort 15			94	<ARG1> poort 15		
47	Getal-2 - poort 15			95	<ARG2> poort 15		
48	Configuratie byte poort 16	7	J7-2	96	<OPC> poort 16		
49	Getal-1 - poort 16			97	<ARG1> poort 16		
50	Getal-2 - poort 16			98	<ARG2> poort 16		
99-100, 125-127	Reserve						

100 = Software Version im PIC

Dateiaufbau: <sv-nr>,<sv-wert>

177 = ??? (was bedeutet der Wert '123' ? nicht verfügbar in V3.5.2...)

LocoIO SV0 Setup

- Bit 7-4 Knipper frequentie
- Bit 3 0 = Normale Poorten
- 1 = Poort 5 tot 12 zijn Servo motor uitgangen
- Bit 2 0 = 2-positie Servo
- 1 = 4-positie Servo (alleen voor LocoServo)
- Bit 1 0 = Vaste code voor drukknoppen
- 1 = Wisselende code voor drukknoppen
- Bit 0 0 = normaal
- 1 = Poort hernieuwing

LocoIO SV1 LocoIO laag adres

SV1 = 1 ... 79 of 81 ... 127 (80 is gereserveerd voor Locobuffer), Default = 81 (0x51) na initialisatie
Opmerking: het LocoIO hoog adres is vast ingesteld op 1

LocoIO SV2 LocoIO Subadres

SV2 = 1 ... 126, Default = 1

LocoIO Configuratie byte

- Bit 7 0 = Ingang
- Bit 6 0 = Blok detector of Drukknop actief laag
1 = Blok detector of Drukknop actief hoog
- Bit 5 0 = Switch
1 = Push button
- Bit 4 0 = Switch of Push button
1 = Blok detector
- Bit 3 0 = OPC_SW REP (0xB1) report code
1 = OPC_SW REQ (0xB0) request code
- Bit 2 0 = uitschakel vertraging van Blok detector
1 = normale uitschakeling van Blok detector
- Bit 1 Niet gebruikt (1 = default)
- Bit 0 Niet gebruikt (1 = default)

- Bit 7 1 = Uitgang
- Bit 6 0 = Contact
1 = Blok detector
- Bit 5 0 = Normaal
1 = Multi
- Bit 4 0 = Normaal
1 = Knipperen
- Bit 3 0 = Vast contact
1 = Puls contact
- Bit 2 0 = software puls reset
1 = hardware puls reset
- Bit 1 Niet gebruikt (0 = default)
- Bit 0 0 = Bij opstarten vast contact hoog
1 = Bij opstarten vast contact laag

LocoIO Getal-1

	Val
Bit 7	0
Bit 6	A7
Bit 5	A6
Bit 4	A5
Bit 3	A4 1
Bit 2	A3
Bit 1	A2
Bit 0	A1

4 3 2 1 0
1 0 0 0 0

7 B

0111 1011

LocoIO Getal-2

- 8 Bit 7 Niet gebruikt (0 = default)
- Bit 6 0 = normaal
1 = Wissel terugmelding
- Bit 5 A0 voor blok detectors
0 = Contact 1
1 = Contact 2
- 1 Bit 4 0 = Puls contact
1 = Blok detectors, vast contact
- 8 Bit 3 A11
- 4 Bit 2 A10
- 2 Bit 1 A9
- 1 Bit 0 A8

Out 4 3 2 1 0
In 7 6 5 4 3 2 1 0
GND 0 1 2 3 4 5 6 7
Vcc 8 9 10 11 12 13 14 15

LcoIO <OPC>, <ARG1>, <ARG2>

Ingangen kunnen buiten hun specifieke Loconet code, ook een extra 4-byte Loconet code sturen.

Dit moeten de 3 eerste bytes zijn van een "4- byte Message Opcode" zoals beschreven in de Loconet Specificaties.

De vierde byte, de checksum, wordt er in de LcoIO software automatisch aan toe gevoegd

Als de <OPC> een waarde heeft lager dan 128 (0x80) dan wordt de code niet meegestuurd.

Het is mogelijk bijvoorbeeld met drukknop om een wissel en een signaal met verschillend adres samen te activeren.

Programmeren van SV's

De SV's in LcoIO kunnen geprogrammeerd worden met het LcoIOconfiguratie programma.

Of als je zelf een programma maakt kan je gebruik maken van Loconet OPC_PEER_XFER boodschappen.

Commandos voor instellen van SV's

01 – SV schrijven

02 – SV lezen

PC naar LcoIO loconet boodschap (OPC_PEER_XFER)

<u>Code</u>	<u>SV lezen</u>	<u>SV schrijven</u>
0xE5	Opcode voor OPC_PEER_XFER	0 Opcode voor OPC_PEER_XFER
0x10	Boodschap lengte	1 Boodschap lengte
SRCL	0x50 laag adres byte van Locobuffer	2 0x50 laag adres byte van Locobuffer
DSTL	LcoIO laag adres	3 LcoIO laag adres
DSTH	0x01 (vast LcoIO hoog adres)	4 0x01 (vast LcoIO hoog adres)
PXCT1		5
D1	Commando = 0x02	6 Commando = 0x01
D2	SV nummer	7 SV nummer
D3	0x00	8 0x00
D4	0x00	9 Data
PXCT2		10
D5	LcoIO Subadres	11 LcoIO Subadres
D6	0x00	12 0x00
D7	0x00	13 0x00
D8	0x00	14 0x00
CHK	Checksum	Checksum

LcoIO naar PC beedschap (OPC_PEER_XFER)

<u>Code</u>	<u>SV lezen</u>	<u>SV schrijven</u>
0xE5	Opcode voor OPC_PEER_XFER	0 Opcode voor OPC_PEER_XFER
0x10	Boodschap lengte	1 Boodschap lengte
SRCL	LcoIO laag adres	2 LcoIO laag adres
DSTL	0x50 laag adres byte van Locobuffer	3 0x50 laag adres byte van Locobuffer
DSTH	0x01 high adres byte van Locobuffer	4 0x01 high adres byte van Locobuffer
PXCT1	Hoog order bit van LcoIO versie	5 Hoog order bit van LcoIO versie
D1	Originele Commando = 0x02	6 Originele Commando = 0x01
D2	SV nummer aanvraag	7 SV nummer aanvraag
D3	Laagste 7 bits van LcoIO versie	8 Laagste 7 bits van LcoIO versie
D4	0x00	9 0x00
PXCT2	Hoog order bit van aanvraag data	A Hoog order bit van geschreven data
D5	LcoIO Subadres	B LcoIO Subadres
D6	Aanvraag_Data	C 0x00
D7	Aanvraag Data + 1	D 0x00
D8	Aanvraag Data + 2	E Geschreven Data —
CHK	Checksum	F Checksum

Commandos voor instelling van multi poort (MP)

! MP commandos werken NIET op Booster I/O of LocoServo I/O

- 03 – MP schrijven
- 04 – MP lezen

PC naar LocoIO loconet boodschap (OPC_PEER_XFER)

- 0 0xE5 Opcode voor OPC_PEER_XFER
- 1 0x10 Boodschap lengte
- 2 SRCL 0x50 laag adres byte van Locobuffer
- 3 DSTL LocoIO laag adres
- 4 DSTH 0x01 (vast LocoIO hoog adres)
- 5 PXCT1
 - D1 Commando
 - D2 0x00
 - D3 0x00
 - D4 LocoIO Subadres
- 10 PXCT2 Hoog order bits van mask en data
 - D5 laag mask (voor MP schrijven) of 0x00 (voor MP lezen)
 - D6 laag data (voor MP schrijven) of 0x00 (voor SV lezen)
 - D7 hoog mask (voor MP schrijven) of 0x00 (voor MP lezen)
 - D8 hoog data (voor MP schrijven) of 0x00 (voor SV lezen)
- CHK Checksum

LocoIO naar PC boodschap (OPC_PEER_XFER)

- 0xE5 Opcode voor OPC_PEER_XFER
- 0x10 Boodschap lengte
- SRCL LocoIO laag adres
- DSTL 0x50 laag adres byte van Locobuffer
- DSTH 0x01 high adres byte van Locobuffer
- PXCT1 Hoog order bit van LocoIO versie
- D1 Originele Commando
- D2 0x00
- D3 Laagste 7 bits van LocoIO versie
- D4 LocoIO Subadres
- PXCT2 Hoog order bits van mask en data
 - D5 laag mask
 - D6 laag data
 - D7 hoog mask
 - D8 hoog data
- CHK Checksum

Laag mask: Poort 1-8 met 1-poort is uitgang, 0-poort is ingang

Laag data: Poort 1-8 met waarde van poort data

Hoog mask: Poort 9-16 met 1-poort is uitgang, 0-poort is ingang

Hoog data: Poort 9-16 met waarde van poort data

Voorbeeld:

Als de laag mask is gezet op 170 (0xAA) (10101010) en de waarde van de laag data is gezet op 240 (0xF0) (11110000) dan gebeurt het volgende

Laag mask	Laag data	poort	wat gebeurt
1	1	1	veranderd naar 1
0	1	2	geen verandering
1	1	3	veranderd naar 1
0	1	4	geen verandering
1	0	5	veranderd naar 0
0	0	6	geen verandering
1	0	7	veranderd naar 0
0	0	8	geen verandering

OPC_INPUT REP 0xB2 ; General SENSOR Input codes NO
 ; <0xB2>, <IN1>, <IN2>, <CHK> Remark for LocoIO by deLoof (V1.48)
"X" has to be set to 0, otherwise didn't react when receiving this telegram
deLoof will also set "X" to 0 when sending this telegram
 <IN1> =<0,A6,A5,A4- A3,A2,A1,A0>, 7 ls adr bits. A1,A0 select 1 of 4 inputs pairs in a DS54
 <IN2> =<0,X,I,L- A10,A9,A8,A7> Report/status bits and 4 MS adr bits.
 "I"=0 for DS54 "aux" inputs and 1 for "switch" inputs mapped to 4K SENSOR space.
 (This is effectively a least significant adr bit when using DS54 input configuration)
 "L"=0 for input SENSOR now 0V (LO) , 1 for Input sensor >=+6V (HI)
 "X"=1, control bit , 0 is RESERVED for future!

OPC_SW REP 0xB1 ;Turnout SENSOR state REPORT NO
 ; <0xB1>, <SN1>, <SN2>, <CHK> SENSOR state REPORT
 <SN1> =<0,A6,A5,A4- A3,A2,A1,A0>, 7 ls adr bits. A1,A0 select 1 of 4 input pairs in a DS54
 <SN2> =<0,1,I,L- A10,A9,A8,A7> Report/status bits and 4 MS adr bits.
 this <B1> opcode encodes **input levels** for turnout feedback
 "I" =0 for "aux" inputs (normally not feedback), 1 for "switch" input used for turnout
 feedback for DS54 ouput/turnout # encoded by A0-A10
 "L" = 0 for this input 0V (LO), 1= this input > +6V (HI)
 alternately;
 <SN2> =<0,0,C,T- A10,A9,A8,A7> Report/status bits and 4 MS adr bits.
 this <B1> opcode encodes current **OUTPUT levels**
 "C"= 0 if "Closed" ouput line is OFF, 1="closed" output line is ON (sink current)
 "T"=0 if "Thrown" output line is OFF, 1="thrown" output line is ON (sink I)

OPC_SW REQ 0xB0 ;REQ SWITCH function NO
 ; <0xB0>, <SW1>, <SW2>, <CHK> REQ SWITCH function
 <SW1> =<0,A6,A5,A4- A3,A2,A1,A0>, 7 ls adr bits. A1,A0 select 1 of 4 input pairs in a DS54
 <SW2> =<0,0,DIR,ON- A10,A9,A8,A7> Control bits and 4 MS adr bits.
 ,DIR=1 for Closed/GREEN, =0 for Thrown/RED
 ,ON=1 for Output ON, =0 FOR output OFF

Note-, Immediate response of <0xB4><30><00> if command failed, otherwise no response