

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

**Michelstadt (Bw)**

**LocoIO**  
**Zusammenfassung der relevanten Dokumentationen**  
Version 8

**Inhaltsverzeichnis**

LocoIO – Betriebsanleitung (H.Deloof) .....	3
Befehlsübersicht (Ausschnitt aus der Loconetpersonaledition, Digitrax) ..	16
LocoHDL Module Konfiguration (H.Deloof) .....	17
LocoIO module software 1.48 – SV-Übersicht (H.Deloof) .....	30
wLocoIO-2 – Bauanleitung (W.Hückel) .....	34
LocoIO – Ergänzungen zu Konfiguration und Betrieb (M.Zimmermann) ..	54

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

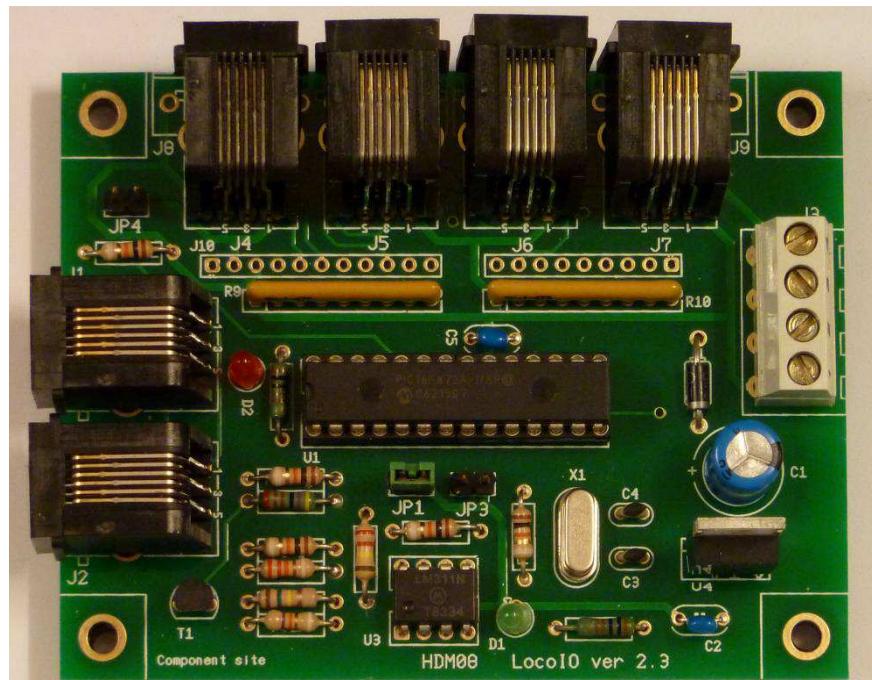
### Linkliste

Verwendete Links rund um die LocoIOs:

- 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](http://embeddedloconet.sourceforge.net/SV_Programming_Messages_v13_PE.pdf)
- Das Original: <http://www.locobuffer.com/LocoIO/LocoIO.pdf>
- Die SV-Übersicht:  
<http://wiki.rocrail.net/lib/exe/fetch.php?id=loconet-io-de&cache=cache&media=loconet:lio-sw:locio.pdf>
- Ergänzungen zu Konfiguration und Betrieb:  
<https://github.com/Kruemelbahn/LocoIO-Editor/blob/main/Documentation/LocoIO-Erg%C3%A4nzungen.pdf>

# *LocoIO*

## *Betriebsanleitung*



### HDM08

#### **Haftungsausschluss:**

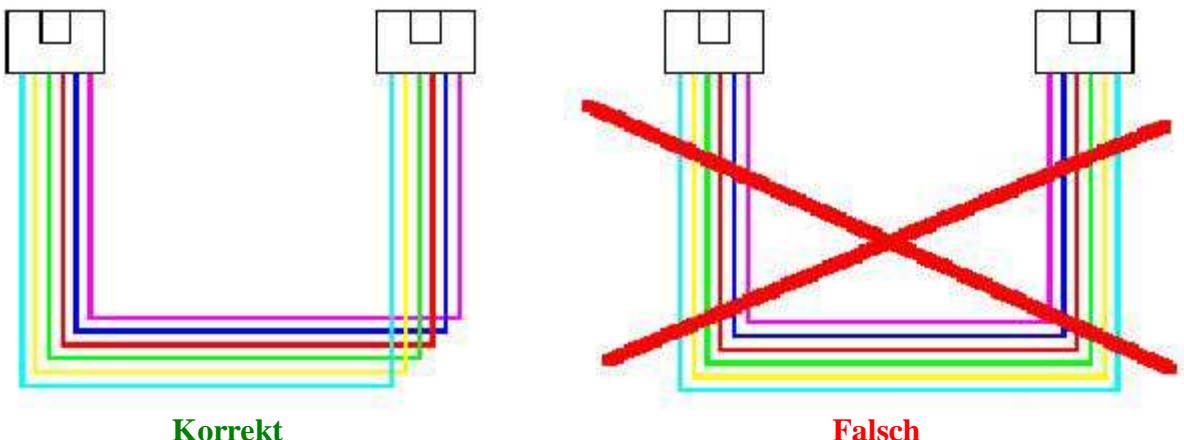
Die Benutzung von irgendwelchem Objekt, das auf diesem Site gekauft werden kann, oder irgendwelcher Prozedur auf diesem Site ist auf eigene Gefahr. Al diese Objekte und Prozeduren sind entwickelt für den persönlichen Gebrauch, und ich finde sie sehr nützlich. Deswegen will ich das hier teilen mit andere Modelleisenbahnliebhaber. Alle Objekte und Prozeduren sind getestet auf meinen Modelleisenbahnsystemen, ohne das es irgendwelche Schaden verursacht sind. Trotzdem garantiert das selbstverständlich nicht, dass alle Möglichkeiten und Prozeduren in allen Umständen oder Systemen funktionieren werde. Ich kann also selbstverständlich keine Haftung übernehmen, wenn diese Objekte oder Prozeduren in anderen Umständen oder Systemen benutzt werden. Verlassen Sie sich immer auf das eigene Urteilsvermögen und den gesunden Menschenverstand.

# LocoIO Modul Hardware 2.3

Dies ist ein Modul, 16 Bit Eingang/Ausgang, mit Anschluss zum Loconet. Es ist möglich das Modul zu verbinden mit Signalen, Schaltern, Stromsensoren, Druckknöpfe, ... . Nun auch mit blinkernden Ausgängen und 4-wege Signalausgang. Funktionellerweise macht dieses Gerät nur die Umsetzung eines 0 oder +5 Volt-Signals in eine Loconet-Mitteilung und/oder die Umsetzung einer Loconet-Kommando in ein 0 oder +5 Volt-Signal, und das für 16 unabhängige Bits. Es macht auch die Software „de-bounce“ von dem Eingang.

## **Loconet-Anschluss:**

Zum Anschließen am Loconet brauchen Sie einen 6 Faden-Kabel mit RJ12 Anschlusse. Es ist sehr wichtig, dass an beiden Enden des Kabels Pinne 1 an Pinne 1 angeschlossen ist.

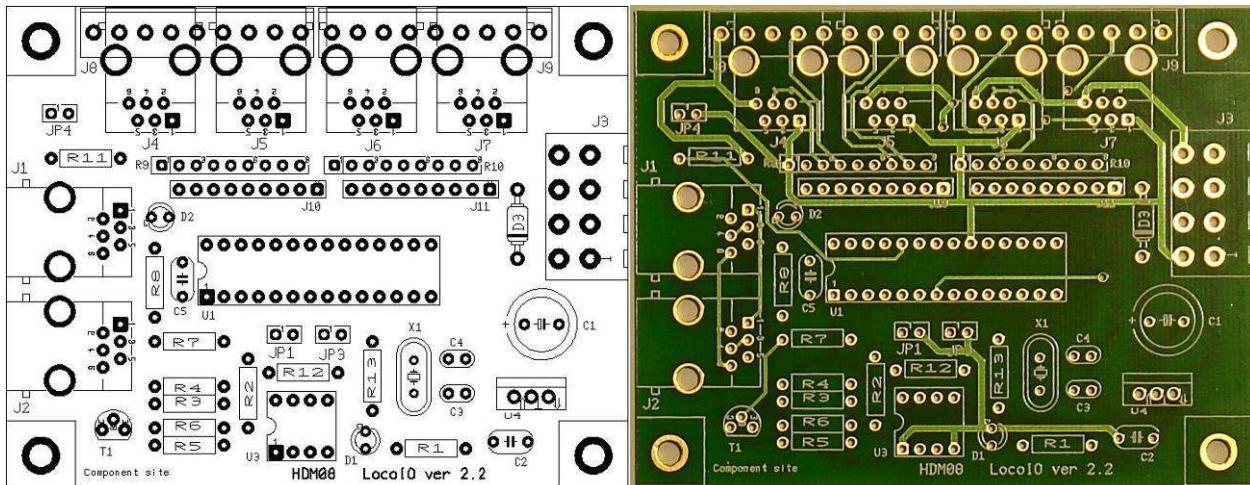


## Materialliste:

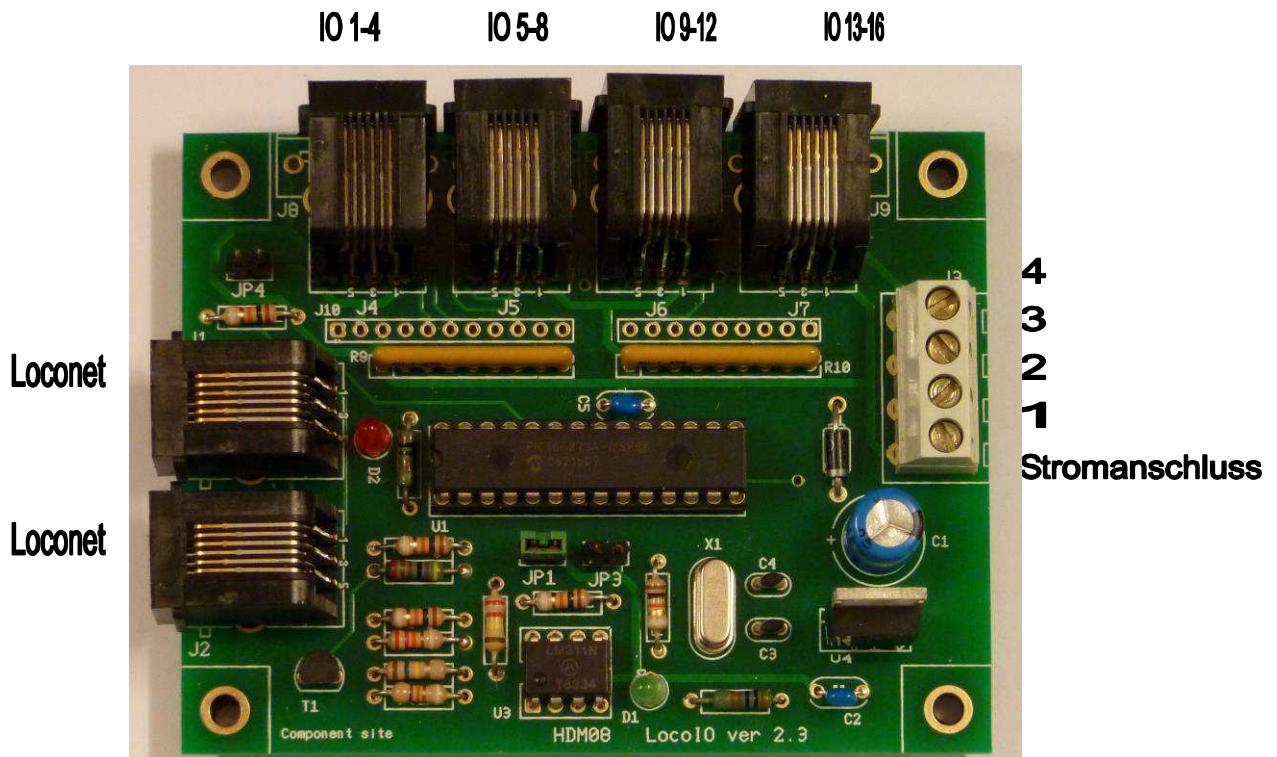
UT_GERÄT	UT_WERT	Refdes
Spannung Regler	7805	U4
Diode	1N4001	D3
Kondensator	100nF	C2, C5
Kondensator	15pF	C3, C4
Komparator IC	LM311N	U3
ELCO	100µF/25V	C1
Stecker	4 Pinne Platine Stecker (5,08)	J3
Stecker	9 Pinne	J10, J11
JUMPER2	2 Pinne	JP1, JP3, JP4
LED Ø3mm	Grün	(Option) D1
LED Ø3mm	Rot	D2
NPN Transistor	BC337	T1
PIC	PIC16F873-20/SP oder PIC16F873A-I/P	U1
Widerstand	10kΩ	R4, R11, R12, R13, R14
Widerstand	150kΩ	R6
Widerstand	1kΩ	R1, R8
Widerstand	220kΩ	R2
Widerstand	22kΩ	R3
Widerstand	47kΩ	R5
Widerstand	4k7Ω	R7
Widerstandspack	8x10kΩ (9 Pinnen SIL)	R9, R10
Stecker	RJ12	J1, J2
XTAL	Quartz 20MHz	X1
Stecker oder	RJ12	J4, J5, J6, J7
Stecker	9 Pinne Platine Stecker (3,81)	J8, J9

## Bemerkung:

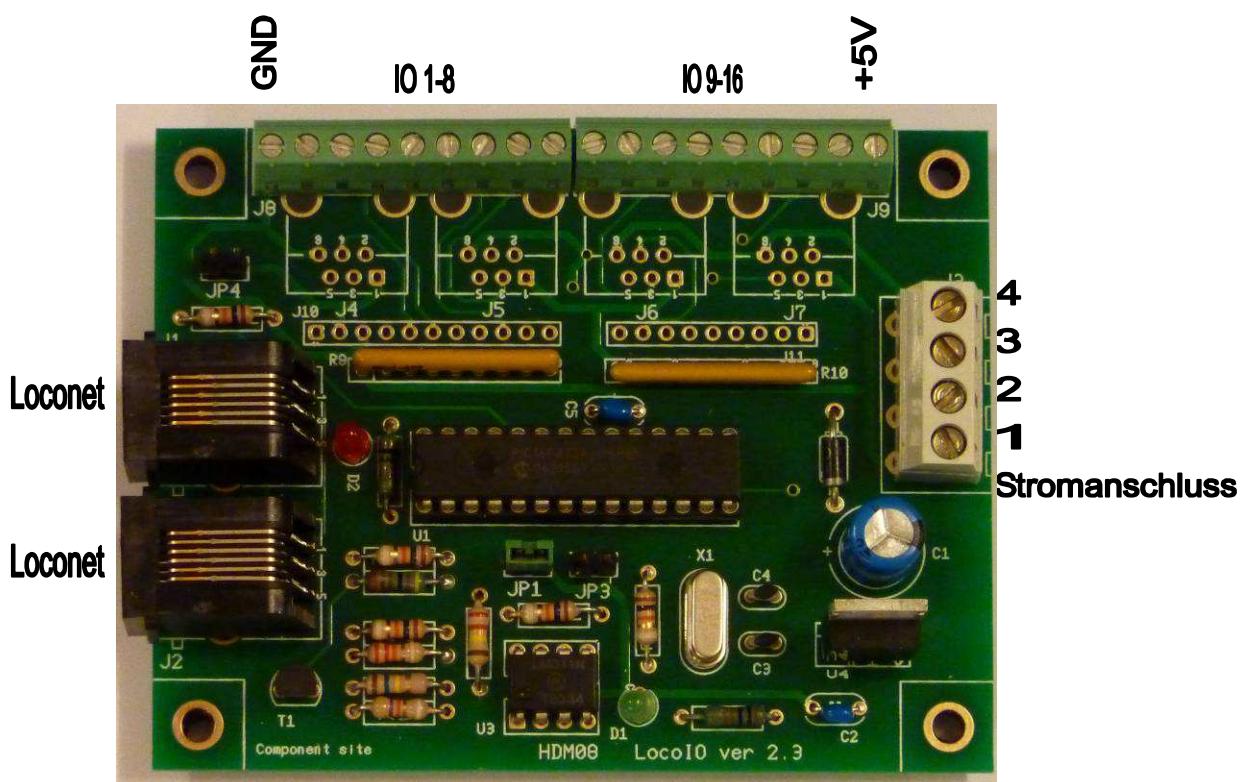
- Montieren Sie die PIC auf einem IC Basis, so dass Sie auch dieser PIC auf die LocoBuffer Modul programmieren können.
- Wenn Ihre XTAL Komponente aus Metall ist, müssen Sie dafür sorgen, dass es kein Kontakt gibt zwischen dem Metalloberfläche des XTAL und den Lötlöcher.
- Jede IO Pinne kann als Ausgang max. 20mA generieren.
- Das ganze Pack IO Pinnen soll zusammen max. 200mA generieren.
- Steckers J10 und J11 ist eine Option
- RJ12 Steckers J4, J5, J6,J7 können auch verfangen werden mit Platine Steckers J8 und J9.



Standardversion mit Steckern RJ12 für Extension Module und einfache Anschlüsse.



Möglichkeit, zum der Leitung Beziehungen mit Platine Steckern J8 und J9 oder mit Stift Anschlüsse J10 und J11 herzustellen



Anmerkung: Zwei Stecker RJ12 und 1 Platine Stecker ist auch möglich.

**Jumpereinstellungen:**

JP1	Aus	Eingang Status bei Stromanschluss und nach einer Loconet Trennung.
	An	Ohne Eingang Status.
JP3	Aus	normal
	An	nicht benutzt
JP4	Aus	normal
	An	nicht benutzt
		Diese Jumper ist vorbehalten für spätere Anwendung
		Diese Jumper ist vorbehalten für spätere Anwendung

**Anschlussmöglichkeiten J3 Stromanschluss:****Eingang:**

Pinne 2: 12V-15V DC Eingang  
 Pinne 4: GND Eingang

**Output:**

Pinne 1: 5V Ausgang  
 Pinne 4: GND Ausgang

**IO Anschlüsse:**

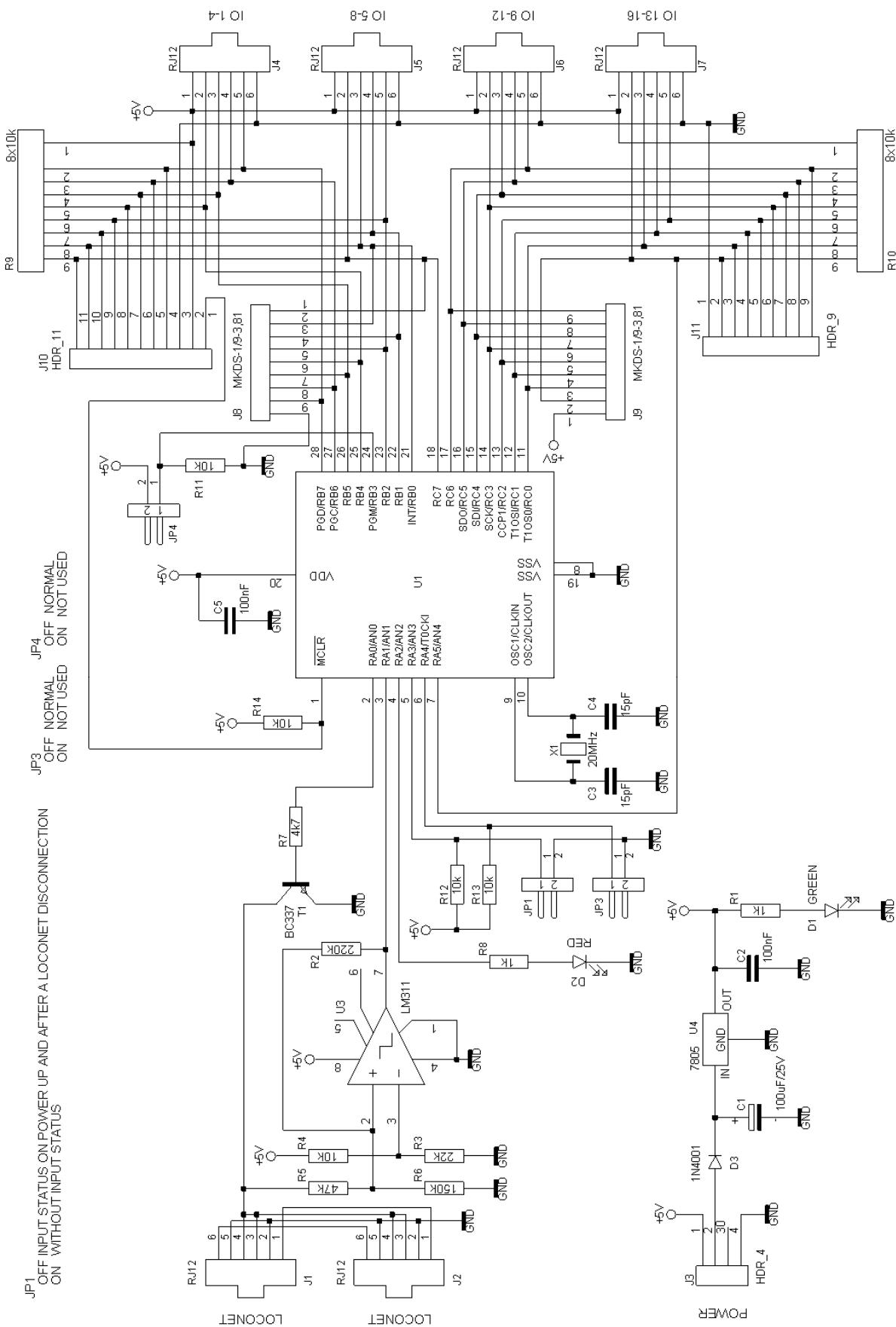
	<b>J4</b>	<b>J5</b>	<b>J6</b>	<b>J7</b>		<b>J8</b>	<b>J9</b>		<b>J10</b>	<b>J11</b>
Pin 1:	+5V	+5V	+5V	+5V		Pin 1:	IO-8	+5V	Pin 1:	VPP
Pin 2:	IO-4	IO-8	IO-12	IO-16		Pin 2:	IO-7	IO-16	Pin 2:	+5V
Pin 3:	IO-3	IO-7	IO-11	IO-15		Pin 3:	IO-6	IO-15	Pin 3:	GND
Pin 4:	IO-2	IO-6	IO-10	IO-14		Pin 4:	IO-5	IO-14	Pin 4:	IO-8
Pin 5:	IO-1	IO-5	IO-9	IO-13		Pin 5:	IO-4	IO-13	Pin 5:	IO-7
Pin 6:	GND	GND	GND	GND		Pin 6:	IO-3	IO-12	Pin 6:	IO-6
						Pin 7:	IO-2	IO-11	Pin 7:	IO-5
						Pin 8:	IO-1	IO-10	Pin 8:	IO-4
						Pin 9:	GND	IO-9	Pin 9:	IO-3
									Pin 10:	IO-2
									Pin 11:	IO-1

**Rot LED:**

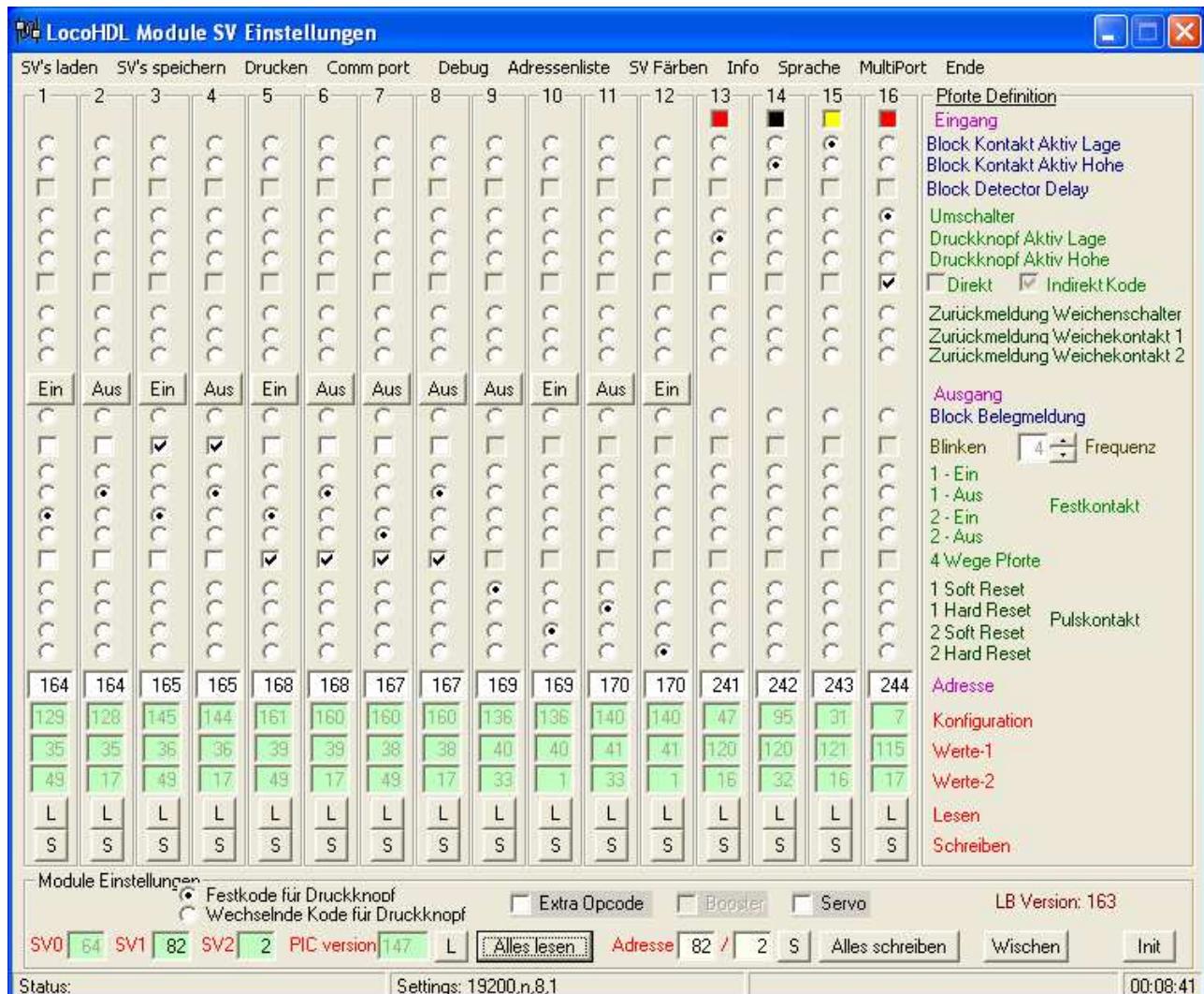
- An                   keine zentral Station oder keine Stromquelle auf Locobuffer mit J6 selektiert.
- Aus                 Loconet OK, kein Tätigkeitsanzeige
- Blitzen             Loconet Kommando Übertragung

**Bemerkungen:**

- Für eine Digitrax Command Station DB150 setzt JP1 An.
- Mit einer Gleichstrom-Versorgung ist der GND dieselben wie ein Intellibox oder ein Locobooster.



## LocoHDL Konfiguration Programm Vorbild eines LocoIO:

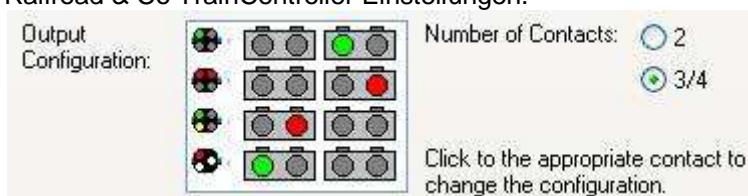


IO 1 und 2 ist ein rot/grün Signal (1=rot, 2= grün)

IO 3 und 4 ist ein blinkerndes rot/grün Signal

IO 5,6,7 und 8 ist ein 4-Wege-Signal

Railroad & Co TrainController Einstellungen.



IO 9 und 10 ist eine Weiche mit Spulen nur mit Softwarepulslänge

IO 11 und 12 ist eine Weiche mit Spulen mit Software und/oder Hardware Pulslänge

IO 13 ist einer Druckknopf aktiv niedrig, der OPC\_SW\_REQ (0xB0)-Kodes schicken

IO 14 ist eine Blockaufspürung aktiv hoch

IO 15 ist eine Blockaufspürung aktiv niedrig

IO 16 ist ein Kippschalter, der OPC\_SW REP (0xB1)-Kodes schicken

## Einige LocoIO Möglichkeiten

### Blockdetektor Eingang:

Schickt ein Kommando OPC\_INPUT REP (0xB2) zum Loconet.

Konfigurationsbyte = 31 (1FH)

*dann ist der Tor konfiguriert wie ein Blockaufspurungsmelder aktiv niedrig*

Konfigurationsbyte = 27 (1BH)

*dann ist der Tor konfiguriert wie ein Blockaufspurungsmelder aktiv niedrig verzögert ausschalten*

Konfigurationsbyte = 95 (5FH)

*dann ist der Tor konfiguriert wie ein Blockaufspurungsmelder aktiv hoch*

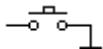
Konfigurationsbyte = 91 (5BH)

*dann ist der Tor konfiguriert wie ein Blockaufspurungsmelder aktiv hoch verzögert ausschalten*

Bemerkung: Der verzögert ausschalten hängt ab von der hergestellten blinkenden Rate.

Empfehlung: benutze der aktive niedrige Eingang. Auf den LocoIO gibt es bereits ein hochziehen „pull-up“ Widerstand womit Sie einfach das Tor benutzen können. Aktiv hoch ist eingebaut für bestimmte Module die nur ein hoher Eingang haben.

- Der Stromsensor macht es einfach Züge, oder andere Strombenutzende Sachen, unaufhörlich auf zu spüren auf Teilstrecken Ihrer Bahnanlage.
- Erdungskontakte (aktiv niedrig) wie Reed-Kontakte, Kontaktschienen, ... und gibt Informationen über passierende Züge, spezifische Waggons mit magnetischen Streifen,...
- Mit einem Druckknopf zum Erdung können Sie Feedback nach dem Computer geben, so dass dieser dann irgendwelche Aktion machen kann, so wie z. B. dass starten einer Route oder Schema.



### Kippschaltereingang:

Schickt ein “Direkt” Kommando OPC\_SW\_REQ (0xB0) zum Loconet.

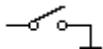
Konfigurationsbyte = 15 (0FH)

Oder

Schickt ein “Indirekt” Kommando OPC\_SW REP (0xB1) zum Loconet.

Konfigurationsbyte = 07 (07H)

- Mit nur einem Schalter und einem Eingangsanschluss können Sie ein Hoch (interne „pull-up“) oder Niedrig (Erdungsschalter). Damit können Sie ein rotes/grünes Signal machen, Weiche rechts oder links einstellen, ... Sehr praktisch auf einem Panel, aber begrenzt auf nur zwei Operationen.



### Druckkopfeingang:

Schickt ein “Direkt” Kommando OPC\_SW\_REQ (0xB0) zum Loconet.

Konfigurationsbyte = 47 (2FH) ) *dann ist der Tor konfiguriert wie ein Druckknopf aktiv niedrig*

Konfigurationsbyte = 111 (6FH) *dann ist der Tor konfiguriert wie ein Druckknopf aktiv hoch*

Oder

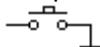
Schickt ein “Indirekt” Kommando OPC\_SW REP (0xB1) zum Loconet.

Konfigurationsbyte = 39 (27H) ) *dann ist der Tor konfiguriert wie ein Druckknopf aktiv niedrig*

Konfigurationsbyte = 103 (67H) *dann ist der Tor konfiguriert wie ein Druckknopf aktiv hoch*

Empfehlung: benutze der aktive niedrige Eingang. Auf den LocoIO gibt es bereits ein hochziehen „pull-up“ Widerstand womit Sie einfach das Tor benutzen können. Aktiv hoch ist eingebaut für bestimmte Module die nur ein hoher Eingang

- Mit Erdungskontakte (aktiv niedrig) mit einem Druckknopf können Sie Signale auf rot/grün/orange stellen,... Dies eist empfehlenswert wenn Sie Panels benutzen.



- Mit Erdungskontakte (aktiv niedrig) wie Reed-Kontakte, Kontaktschienen, ... können Sie Signale und Weichen ändern. Dies ist sehr praktisch, wenn Sie Ihre Bahnanlage ohne PC bedienen.

- Mit Wechselkode können Sie ein Signal auf rot/grün stellen, Weichen rechts oder links einstellen, . . . Sehr praktisch auf einem Panel, aber begrenzt auf nur zwei Operationen.

Bemerkung: dies ist nur empfehlenswert, wenn Sie alle Handlungen einfach visuell kontrollieren können.

### **Beschreibung von "Direkt" und "Indirekt" Kode.**

Beim "Direkt" Kode wird ein Kommando zum Loconet geschickt, die sofort anfangt mit der Bedienung der Weichen und Signale. Das ist die Normaleinstellung, wenn kein PC benutzt wird. Wenn man auch ein PC benutzt, dann empfängt dieser das Kommando für die Weiche oder das Signal, auch wenn der Software keine Zustimmung gibt für irgendwelche Änderung. Das Bildschirm wird also nicht übereinstimmen mit dem wirklichen Zustand der Weiche oder des Signals.

Beim "Indirekt" Kode wird ein Kommando zum Loconet geschickt. Das Kommando wird zuerst vom PC evaluiert, zum entscheiden, ob es durchgeführt werden soll. Zunächst wird der PC das Finalkommando, das die Weiche oder den Signal beeinflusst, zeigen.

#### **Bemerkung für „Railroad & Co Traincontroller“ –Benutzer:**

Sie müssen die folgende Option einstellen in RAILROAD.INI um dieser "Indirekt" Kodes akzeptieren zu können:

```
[Connections]
LNWatchTurnoutOutput = 1
```

Nach Empfang eines Indirekten Kodes und nach einer günstigen Evaluation sollte Traincontroller ein Direktkode schicken an Loconet. Leider tut er das NICHT.

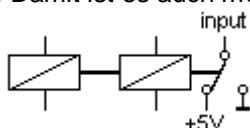
Deswegen sollten Sie am besten einen Indirekten Kode koppeln an einem Traincontroller Schalter oder Druckknopf, die danach bestimmte Handlungen ausführen können.

### **Zurückmeldungsschaltereingang, 2 Methoden:**

Schickt ein Kommando OPC\_SW\_REPLY (0xB1) zum Loconet.

Konfigurationsbyte = 23 (17H)

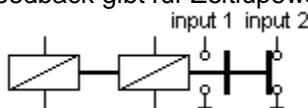
- Wenn wir einem Doppelschalter auf ein Weichenspule stellen, dann gibt dies zum PC die Realposition des Weiche. Damit ist es auch möglich die Weichen die von Hand gestellt wurden auf den PC zu folgen.



Schickt ein Kommando OPC\_SW\_REPLY (0xB1) zum Loconet.

Konfigurationsbyte = 55 (37H)

Mit dieser Konfiguration brauchen wir nur zwei Eingänge um ein Weiche zu kontrollieren. Der Vorteil ist, dass es so Feedback gibt für Zeitlupe weichemotoren.



Bemerkung für „Railroad & Co Traincontroller“ -Benutzer. Sie müssen die folgende Option einstellen in RAILROAD.INI um dieser Zurückmeldung Kodes akzeptieren zu können:

```
[Connections]
LNWatchTurnoutOutput = 1
```

## Blockbesetzmeldungsausgang:

Empfängt ein OPC\_INPUT REP (0xB2) vom Loconet.

Konfigurationsbyte = 192 (C0H) *Normalblockbesetzmeldung*

Konfigurationsbyte = 208 (D0H) *blinkernde Blockbesetzmeldung*

- Auf ein Panel können Sie LED stellen, die Sie eine Anweisung gibt über wo es Züge gibt auf Ihre Bahnanlage. Sie können 1 oder 2 LED direkt am LocoIO anschließen. Der Output ist max. 5V/20mA.



Wenn Sie mehr LED für einen Ausgang wünschen, dann müssen Sie ein Antriebsmodule benutzen.

## Fixkontakteausgang:

Empfängt ein OPC\_SW\_REQ (0xB0) vom Loconet.

Konfigurationsbyte = 128 (80H) *Normal Ausgang, Ausgang ist ausgeschaltet beim Start*

Konfigurationsbyte = 144 (90H) *blinkernder Ausgang, Ausgang ist ausgeschaltet beim Start*

Konfigurationsbyte = 160 (A0H) *4-wege Ausgang, LED ist ausgeschaltet beim Start*

Konfigurationsbyte = 176 (B0H) *blinkernder 4-wege Ausgang, LED ist ausgeschaltet beim Start*

Konfigurationsbyte = 129 (81H) *Normal Ausgang, Ausgang ist angeschaltet beim Start*

Konfigurationsbyte = 145 (91H) *blinkernder Ausgang, Ausgang ist angeschaltet beim Start*

Konfigurationsbyte = 161 (A1H) *4-wege Ausgang, LED ist angeschaltet beim Start*

Konfigurationsbyte = 177 (B1H) *blinkernder 4-wege Ausgang, LED ist angeschaltet beim Start*

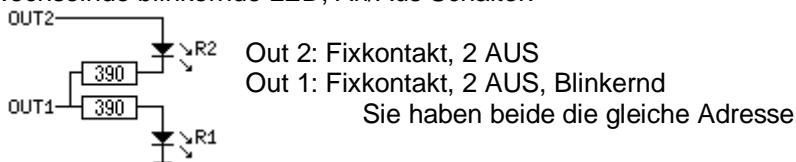
- Dieser Ausgang können Sie benutzen für Signale, Lampe,...

Sie können 1 oder 2 LED direkt am LocoIO anschließen. Der Ausgang ist max 5V/20mA.



Für kommerzielle Signale, die mehr Volt oder Lampe die mehr Strom brauchen, können Sie immer das Antriebs-Module oder Antriebs-Module 2 benutzen.

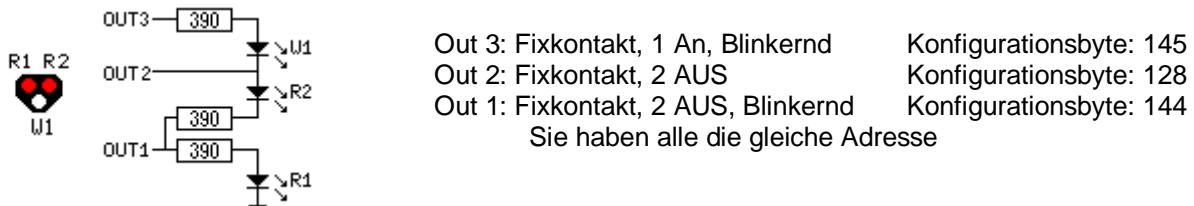
- Zwei wechselnde blinkende LED, An/Aus Schalter.



Konfigurationsbyte: 128

Konfigurationsbyte: 144

- Belgische Übergangssignale: zwei rote LED, wechselnd blinkend, oder weiß blinkend



Out 3: Fixkontakt, 1 An, Blinkernd

Konfigurationsbyte: 145

Out 2: Fixkontakt, 2 AUS

Konfigurationsbyte: 128

Out 1: Fixkontakt, 2 AUS, Blinkernd

Konfigurationsbyte: 144

Sie haben alle die gleiche Adresse

**Bemerkung:** Der Widerstand bei dem LED ist den minimalen empfohlenen Wert. Der Widerstandswert kann höher gestellt werden, wann der LED zu hell ist.

- Dieser Ausgang, zusammen mit dem Antriebs-Modul 2 für mehr Strom und Volt, ist sehr praktisch mit Wechselmotoren mit Endschalter.

## **Pulskontaktausgang:**

Empfängt ein OPC\_SW\_REQ (0xB0) vom Loconet.

Konfigurationsbyte = 136 (88H)

*Der Pulskontakt Ausgang wurde hoch gestellt von einem Softwarekommando vom Loconet und niedrig gestellt von einem zweiten Softwarekommando. Die Länge der Puls ist abhängig von der Zeit zwischen den zwei Kommandos von Steuerung Software. Wenn Sie ein Druckknopf benutzt, ist die Pulslänge gleich der Eindruckzeit des Druckknopfes.*

- Dieser Ausgang, zusammen mit dem Driver- Modul für mehr Strom und Volt, ist sehr praktisch mit Spulenmechanismen beim entkoppeln von Schienen. Sie bestimmen also wie lange des Entkoppelmechanismus aktiv ist, durch den zeit das Sie den Druckknopf eindrücken.

Konfigurationsbyte = 140 (8CH)

*Der Pulskontakt Ausgang wurde hoch gestellt von einem Softwarekommando vom Loconet. Die Länge des Pulses ist zwischen 1 und 2 Blinkpulse lang. Mit Druckknöpfen wird die Pulslänge nun bestimmt vom LocoIO. Wenn ein Softwarekommando zum zurückstellen des Ausgängen bevor die Hardwaresicherheit kommt, wird den Ausgang funktionieren als Konfigurationsbyte 136.*

- Dieser Ausgang, zusammen mit dem Driver-Module für mehr Strom und Volt, ist praktisch für Weichen. Der Ausgang bei dieser Konfiguration gibt eine Sicherheit, so dass die Pulse nicht durchbrennt, wenn das Zurückstellpulskommando nicht kommt.
- Wenn das Kommando für den Pulskontakt kommt von einem Reed-Relais, Kontaktschiene, ... ,dann wird die Spule einer Weiche nicht kontinuierlich aktiviert wenn einem Zug stillsteht auf dieser Reed-Relais oder Kontaktschiene. Dieser Pulslängesicherheit funktioniert als Endschalter für bestimmte Weichen.

## **Servo Motor Ausgang:**

Für analoge Servo Motoren betrieb, sehn sie in die Betriebsanleitung von LocoServo Module.

## **! EMPFEHLUNG:**

- Wenn die verwendete Version von LocoIO nicht die Option Vorwähler „Pforte nicht benutzt“ habe, dann wird es geraten unbenutzte Pforte als Ausgang zu definieren, und sie eine unbenutzte Adresse zu geben. Damit verhindern Sie, dass bei Störungen auf dem Modul, ungewünschte Codes auf Loconet gesandt werden.
- Geben Sie die Pforte, die als Servo Ausgang definiert ist, immer eine Adresse die nur einmal auf einem Modul auftritt.

## Die Geschichte der Versionen

### **LocoIO Version 1.35 bis Version 1.40**

- nicht mehr unterstützt

### **Neu in Version 1.41**

- Blockaufspürungskommandos sind geändert
- In vorherige Versionen hat die Blockaufspürung Konfigurationsbyte 15 (0FH) mit Wert 1 Bit 7 hoch. Alle Blockaufspürungen sind aktiv niedrig oder hoch abhängig von SV0 Bit 4.

- In Version 1.41 gibt es zwei Konfigurationsbytes für Blockaufspürung.  
31 (1FH) Tor ist ein Blockaufspürungseingang aktiv niedrig  
95 (5FH) Tor ist ein Blockaufspürungseingang aktiv hoch  
SV0 Bit 4 und Wert 1 Bit 7 ist nicht mehr benutzt.

Der Vorteil ist, dass jeder Blockaufspürungseingang nun separat aktiv niedrig oder hoch eingestellt werden kann.

- Der Druckknopf schickt nun zwei Kodes zum Loconet, eine beim Eindrücken des Knopfes und eine beim loslassen des Knopfes. Nun ist es möglich eine spule direkt zu aktivieren mittels Knöpfe auf einem Panel.
- Die Kodes die auf Loconet geschickt werden sind nun ganz gemäss die Loconet- Spezifikationen.

### **Neu in Version 1.42**

- Der Ausgang Konfigurationsbyte hat eine neue Funktion  
Konfigurationsbit 2 = 0 Softwarepulszurückstellung  
Konfigurationsbit 2 = 1 Hardwarepulszurückstellung

- Ergebnis:

- Konfigurationsbyte = 136 (88H)

Der Ausgang des Pulskontaktes ist hoch gestellt mittels eines Softwarekommandos vom Loconet und niedrig gestellt mittels eines zweiten Softwarekommandos. Die Pulslänge ist abhängig von der Zeit zwischen die zwei Kommandos von Steuerung Software. Wenn Sie ein Druckknopf benutzt, ist die Pulslänge gleich der Eindruckzeit des Druckknopfes.

- Konfigurationsbyte = 140 (8CH)

In Version 1.42 ist nun eine Sicherheit eingebaut in in LocoIO mit Konfigurationsbyte 140, so dass, wenn das Kommando nicht kommt um den Ausgang wieder niedrig zu stellen, der Ausgang doch niedrig gestellt wird. Die Länge des Pulses ist zwischen 1 und 2 Blinkpulse lang. Mit Druckknöpfen wird die Pulslänge nun bestimmt vom LocoIO. Wenn ein Softwarekommando zum zurückstellen des Ausgängen vor der Hardwaresicherheit kommt, dann wird der Ausgang funktionieren als Konfigurationsbyte 136.

### **Neu in Version 1.43**

- OPC\_SW\_REP (0xB1) kann nun geschickt werden für Kippschalter und Druckknöpfe.  
Indirekte Änderung von Schalter und Signale in Steuerung Software ohne direkte Kommandos zu schicken nach Schalter und Signale.  
Bemerkung für „Railroad & Co Traincontroller“ –Benutzer:  
Sie müssen die folgende Option einstellen in RAILROAD.INI um dieser Kodes zu akzeptieren:  
[Connections]  
LNWatchTurnoutOutput = 1
- Der Druckknopfeingang kann nun auch benutzt werden mit doppelte Kodes.  
So können Sie auch Schalter und Signale ändern ohne Druckknöpfe ein zu drücken.  
Bemerkung: Dies ist nur empfehlenswert, wenn Sie visuell kontrollieren können was Sie machen.

### **Neu in Version 1.44**

- Neue Konfigurationsbyte für Weichefeedback  
- Konfigurationsbyte = 23 (17H)

- So ist es auch möglich die Weiche von Hand zu ändern und das auf dem PC zu folgen.
- Wenn der Eingang niedrig gestellt wird, dann wird ein Richtungskode von der Weichezurückmeldung geschickt, wenn der Eingang hoch wird, dann wird die andere Kode von der Weichezurückmeldung zum Loconet geschickt. Nur ein Eingang ist benötigt um eine Weiche zu kontrollieren.

### **Neu in Version 1.45**

- Neue Konfigurationsbyte für Weichezurückmeldung.  
- Konfigurationsbyte = 55 (37H)

- So ist es auch möglich die Weiche von Hand zu ändern und das auf dem PC zu folgen.
- Mit dieser Konfiguration brauchen Sie zwei Eingänge für die Zurückmeldung einer Weiche. Es ist entwickelt als einem aktiven niedrigen Eingang. Wenn der Eingang niedrig gestellt wird, dann wird ein Richtungskode von der Weichezurückmeldung geschickt. Dies ist ein Vorteil, wenn Sie Zeitlupemotoren benutzen, weil Sie so die Zurückmeldungsschalter unabhängig von einander benutzen können.
- In vorherige Versionen gab es ein Timingproblem, wenn Eingänge zusammen mit blinkernden Ausgängen benutzt werden. Das ist nun aufgelöst.
- Es ist nun eine Sicherheit eingebaut zum vermeiden, dass beide Pulskontakte mit der gleichen Adresse zugleich aktiviert werden. Das verhindert, dass beide Pulse einer Weiche zugleich unter Strom stehen.
- Eingänge können außerhalb ihres spezifischen Loconet Kodes, auch ein Extra 4-byte Loconet Kode senden

- JP1 ermöglichen und sperren Funktion für Eingang Status.

## **Neu in Version 1.46**

- Außer dem normalen Adresse, hat die LocoIO nun auch eine Sub-Adresse. Diese kann benutzt werden um bestimmte Zonen einer Bahn eine feste Adresse zu geben, weil die Sub-Adressen benutzt werden für die unterschiedlichen LocoIO in dieser Zone. In einer Modularbahn ist es nun möglich jedes Klubmitglied eine LocoIO-Adresse zu geben, und zudem kann jedes Klubmitglied unterschiedlichen LocoIO mit unterschiedlichen Sub-Adressen benutzen.

## **Neu in Version 1.47**

- In diese Version werden die Pforte Einstellung erneuern, ohne Aus/An schalten von der power, mit LocoHDL Konfiguration Programme durch ein Schreiben von SV0 oder ein "Schreib Alles".

## **Neu in Version 1.48**

- Sub-Adresse mit MultiPort Kommando
- Wahlweise verzögert ausschalten für Blockabfragung

## **Neu in Version 1.49**

- Jeder Pforte kann ausgewählt werden, wie unbenutzt.
- Letzte Version für PIC 16F873(A)

## **Neu in Version 1.50**

- Für PIC 16F883
- Extra Opcode 2
- Doppelter Eingang
- Speziell Pforte für GO-IDLE-STOP Eingang und STATUS Ausgang.

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!

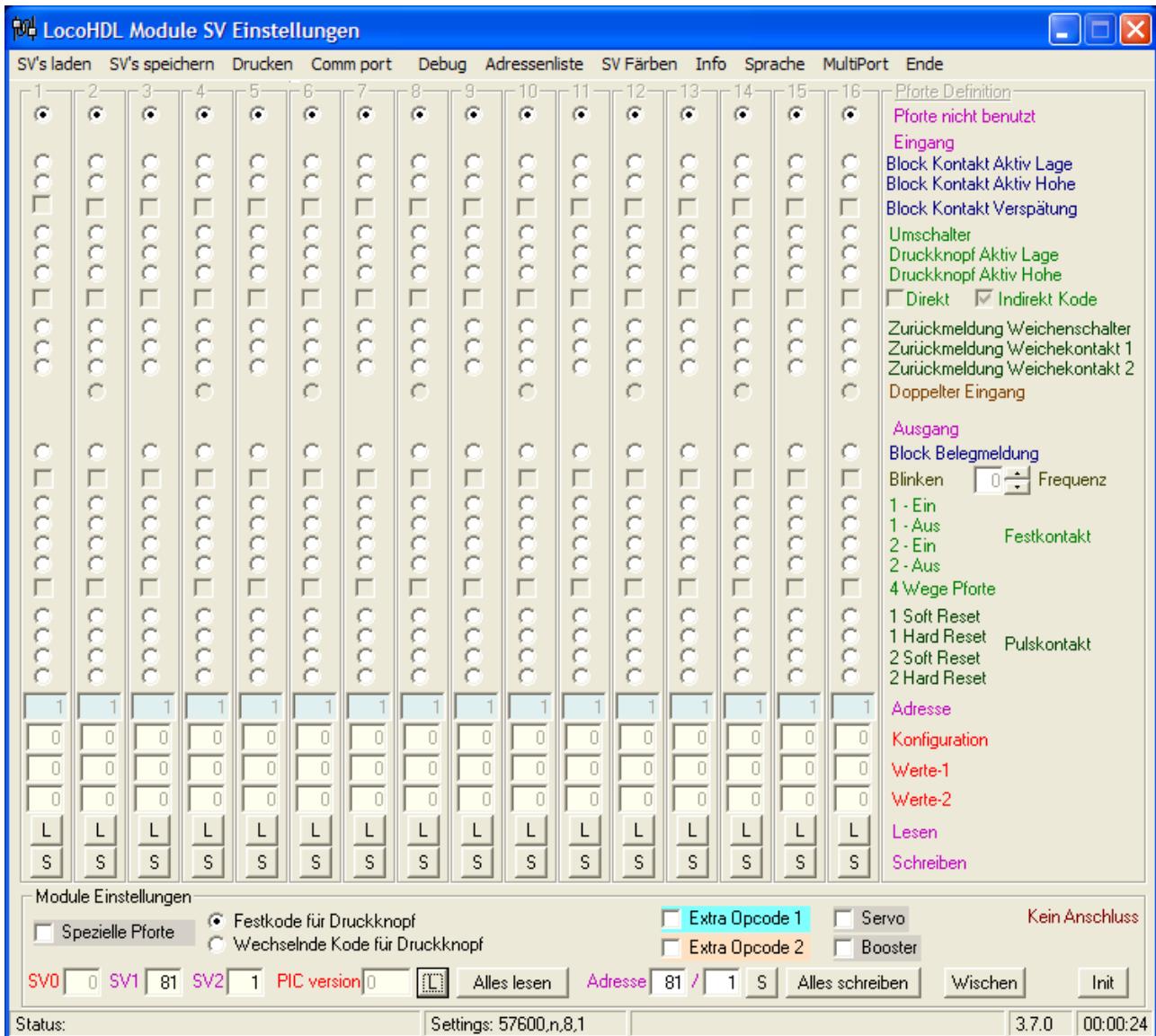
OPC\_SW REP 0xB1 ;Turnout SENSOR state REPORT NO  
 ; <0xB1>, <SN1>, <SN2>, <CHK> SENSOR state REPORT  
 <SN1> =<0,A6,A5,A4- A3,A2,A1,A0>, 7 ls adr bits. A1,A0 select 1 of 4 input pairs in a DS54  
 <SN2> =<0,1,I,L- A10,A9,A8,A7> Report/status bits and 4 MS adr bits.  
 this <B1> opcode encodes **input 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

# *LocoHDL Module*

## *Konfiguration*



### Haftungsausschluss:

Die Benutzung von irgendwelchem Objekt, das auf diesem Site gekauft werden kann, oder irgendwelcher Prozedur auf diesem Site ist auf eigene Gefahr. Al diese Objekte und Prozeduren sind entwickelt für den persönlichen Gebrauch, und ich finde sie sehr nützlich. Deswegen will ich das hier teilen mit andere Modelleisenbahnliebhaber. Alle Objekte und Prozeduren sind getestet auf meinen Modelleisenbahnsystemen, ohne das es irgendwelche Schaden verursacht sind. Trotzdem garantiert das selbstverständlich nicht, dass alle Möglichkeiten und Prozeduren in allen Umständen oder Systemen funktionieren werde. Ich kann also selbstverständlich keine Haftung übernehmen, wenn diese Objekte oder Prozeduren in anderen Umständen oder Systemen benutzt werden. Verlassen Sie sich immer auf das eigene Urteilsvermögen und den gesunden Menschenverstand.

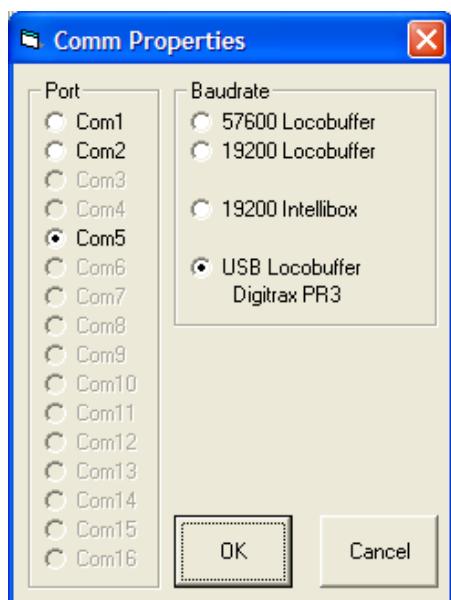
# LocoHDL Konfigurationsprogramm

Dieses Programm wird benutzt zum Konfigurieren, in einer einfachen Weise, von allen LocoHDL Modulen wie LocoIO, LocoServo und LocoBooster. Das Konfigurationsprogramm kommuniziert mit den unterschiedlichen Modulen über eine serielle Schnittstelle oder USB des PCs mit einem LocoBuffer über Loconet.

## Setup Konfigurationsprogramm:

Wenn Sie das LocoHDL Konfigurationsprogramm zum ersten Mal starten, wird es Ihnen fragen die Kommunikationstor (com Port)-Einstellungen ein zu geben.

- 1) Für LocoBuffer mit USB: selektieren "USB Locobuffer" und die Com Pforte.
- 2) Für LocoBuffer mit RS232: LocoBuffer soll in LocoBuffer Modus JP3: 2-3 eingestellt werden.  
Die Baudrate auf den LocoBuffer kann mit JP1 eingestellt werden.  
(Bestimmte älteren PC und tragbaren PC sind nicht geeignet für schnelle Kommunikation, und dort müssen Sie 19200 benutzen). Wählen Sie die Einstellungen die Sie mittels der LocoBuffer-Jumper eingestellt haben, und drücken OK.



3) Für Intellibox: LocoHDL Konfigurationsprogramm funktioniert auch auf der Intellibox software Version 1.5 oder höher mit  
IB Option: 1 Werte: 3 Interface Datenrate 19200 bps  
IB Option: 2 Werte: 4 Interface Syntax Loconet  
IB Option: 5 Werte: 1 Nummer of Stopbits 1  
IB Option: 6 Werte: 255 sperren Sie Tropfen von CTS, wenn Sie 'End' Zustand eintragen

Es gibt einige Begrenzungen, z. B. Sie können keine Adresseliste anfordern.

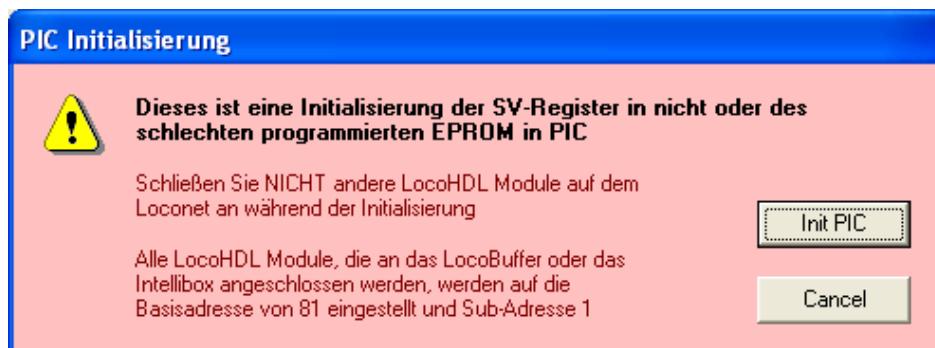
**Comm port** Mit diesem Knopf auf die Werkzeugeiste können Sie immer die Einstellungen ändern.

## Initialisierung des LocoHDL Moduls:

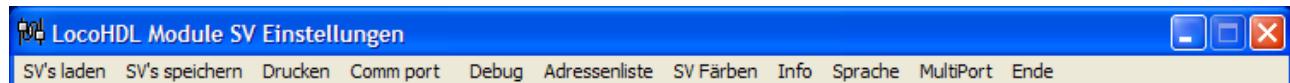
Wenn ein LocoHDL Modul programmiert ist mit einem HEX Datei in einem PIC Programmierer ohne die EPROM zu programmieren, dann gibt es nichts in die SV Register und dann hat das LocoHDL Modul keine Adresse. Darum müssen wir das Modul zuerst initialisieren.

**Schritt 1:** Sorge dafür dass es KEINE anderen Geräte auf die Loconet Anschlüsse gibt. Dies ist sehr wichtig, weil der LocoHDL Programme während der Initialisierung Mitteilungen verschickt, und das kann die Einstellungen von anderen LocoHDL Modulen vernichten.

**Schritt 2:** Starte das LocoHDL Konfigurationsprogramm und Drücke auf dem 'Init' Knopf



# Konfigurationsprogramm



## Werkzeugeistecknöpfe:

**SV's laden** **SV's speichern**

Die SV Information kann für jeden LocoIO gespeichert und geladen werden auf Ihren Computer.

**Ende** Programmausgang Knopf

**Sprache** Sie können unterschiedliche Sprache wählen

**Drucken** Es ist möglich, die Daten der LocoHDL Module zu drucken.

**SV Färben** Hier finden Sie eine Erklärung von den Hintergrundfarben benutzt für die SV Registern.

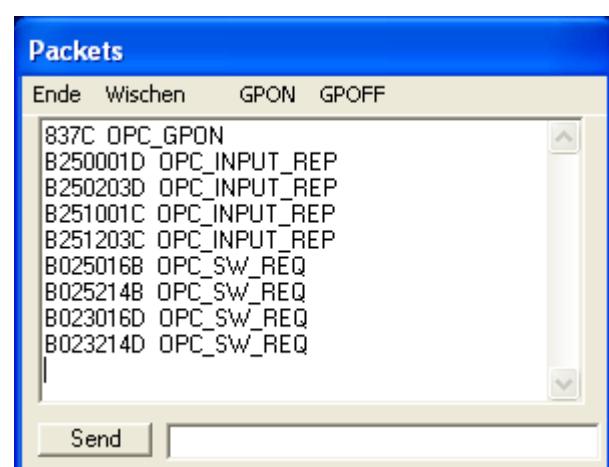


**Debug**

Mit diesem Knopf bekommen Sie ein zweiter Schirm, worauf die Loconet Pakete auf Ihren Loconet-Leine gezeigt werden.

Wenn Sie Erfahrung mit Loconet Kommandos haben, dann können Sie Unten im Schirm ein Paket schicken nach dem Loconet mit oder ohne die Kontrolliersumme.

(Dieser Schirm steht nicht zur Verfügung mit Intellibox Schnittstelle.)



**Adressenliste**

Ende Lesen Wischen

001/001 ver: 148 - LocoIO
002/001 ver: 148 - LocoIO
003/001 ver: 148 - LocoIO
<b>004/001 ver: 148 - LocoIO</b>
005/001 ver: 148 - LocoIO
006/001 ver: 148 - LocoIO
007/001 ver: 148 - LocoIO
008/001 ver: 148 - LocoIO
009/001 ver: 148 - LocoIO
010/001 ver: 148 - LocoIO
011/001 ver: 148 - LocoIO
012/001 ver: 148 - LocoIO
013/001 ver: 148 - LocoIO
014/001 ver: 148 - LocoIO
015/001 ver: 148 - LocoIO
016/001 ver: 102 - LocoServo
017/001 ver: 102 - LocoServo
018/001 ver: 148 - LocoIO
019/001 ver: 148 - LocoIO
020/001 ver: 148 - LocoIO
021/001 ver: 148 - LocoIO
022/001 ver: 148 - LocoIO
023/001 ver: 148 - LocoIO
024/001 ver: 148 - LocoIO
025/001 ver: 148 - LocoIO
026/001 ver: 102 - LocoServo
027/001 ver: 148 - LocoIO
028/001 ver: 148 - LocoIO
029/001 ver: 148 - LocoIO
030/001 ver: 148 - LocoIO
031/001 ver: 148 - LocoIO
032/001 ver: 148 - LocoIO
033/001 ver: 148 - LocoIO
034/001 ver: 102 - LocoServo
035/001 ver: 148 - LocoIO
081/001 ver: 150 - LocoIO
082/000 ver: 144 - LocoIO

**Adressenliste**

Mit dieser Knopf bekommen Sie ein zweiter Schirm das Sie eine Liste zeigt von allen LocoHDL Moduladressen die benutzt werden auf die angeschlossenen Loconet-Bahnen.

Wenn Sie zweimal klicken mit der Maus auf einer Adresse dann alle Register dieses Moduls werden gelesen.

Wenn Sie die Ctrl unten halten und dann die Maus anklicken auf einer Adresse dann lesen sie die Modul Register.

(Dieser Schirm steht nicht zur Verfügung mit Intellibox Schnittstelle.)

**Info**

**LocoHDL Module SV Konfigurations Programm**

Version 3.7.0  
Autor: Hans Deloof

Activ Zustand Kontacts: LocoIO Version 138 (nicht mehr unterstützt)  
Pulse Kontakte: LocoIO Version 139 (nicht mehr unterstützt)  
Blinken und Multi Signalstellung: LocoIO Version 140 (nicht mehr unterstützt)  
Abgesondert block Kontakt kodes: LocoIO Version 141  
Pulse ausgang mit Sicherheit: LocoIO Version 142  
Pulse ausgang mit kippen kode, shalter report code: LocoIO Version 143  
Weiche zurückmeldung Umschalter: LocoIO Version 144  
Weiche zurückmeldung kontakte, Extra Opcode option: LocoIO Version 145  
Sub-Adresse in LocoIO Adresse: LocoIO Version 146  
Analoge Servo Motor steuerung: LocoServo Version 147 - LocoServo Version 100  
Sub-Adresse in MultiPort Kommandos, Block Kontakt Verspätung: LocoIO Version 148  
4-Position Servos: LocoServo Version 101  
Block Kontakt Verspätung: LocoServo Version 102 - LocoBooster Version 002  
Pforte nicht benutzt Option: LocoServo Version 103 - LocoBooster Version 003 - LocoIO Version 149  
Extra Opcode 2 option, Special Ports: LocoServo Version 103 - LocoBooster Version 003 - LocoIO Version 150

---

Programm Abstuetzung LocoIO Version 141 bis 150  
Programm Abstuetzung Booster Version 001 bis 003  
Programm Abstuetzung LocoServo Version 100 bis 103  
Program Abstuetzung LocoBuffer Version 163, 164, 164(USB), Intellibox (oder Twin Center), Digitrax PR3

---

hans.deloof@compaqnet.be  
<http://users.telenet.be/deloof/index.html>

**Info** Information über LocoHDL configurationsprogrammversion

## LocoHDL Module Einstellungen:

Jedes Modul eine einzigartige anspreche Adresse für Loconet zuweisen.

Das LocoHDL Konfiguration Programm kann die Module immer danach ansprechen, gleichmäßig Sie während in Services eines Modelbahn.



LocoHDL Module enthalten auch Subadressen außerhalb der gewöhnlichen Adresse:

- Ein kann eine örtlich festgelegte Adresse zu den Bereichen in einem Plan geben und mit Subadressen innerhalb eines Bereichs.
- In einem Modul Plan können Sie eine Adresse den Vereinsmitgliedern bewilligen und dann kann jeder Vereinsmitglied einige LocoHDL Module mit einigen Subadressen benutzen.

Adresse (SV1) = 1 ... 79 oder 81 ... 127 (80 ist reserviert für LocoBuffer), Standard = 81 nach Initialisierung.

Subadresse (SV2) = 1 ... 126, Standard = 1 nach Initialisierung.

Insgesamt kann man also 15876 unabhängige Modulen ansprechen.



Sie können SV0 und "PIC Version" nicht direkt ändern. Diese sind nur zur Information.  
SV1 ist die Adresse des LocoHDL Moduls womit sind Kommunizieren.



Sie können ein neuer Adressewert von dem LocoHDL Modul eingeben, und es in die LocoHDL speichern durch auf dem "S" Knopf zu drücken. Diese neue Adresse wird angegeben in das LocoHDL Modul, wovon der aktuellen Adresse in SV1 uns Sub-Adresse in SV2 angedeutet ist.

Wenn Sie auf "Alles Schreiben" drücken, dann werden Sie den neuen Adressewert speichern und ALLE andere SV im LocoHDL Modul.

Wenn Sie mit einem anderen LocoHDL Modul kommunizieren wollen, müssen Sie ein neuer Adressewert eingeben im Textfeld SV1 und/oder Sub-Adresse in SV2.



Und auf dem "L" Knopf drücken.



Wenn Sie mit dem LocoHDL Modul kommunizieren, dann sind alle Textfelder grün (wenn die Werte korrekt sind) und werden die Werte von den Registern SV0, SV1, SV2 und die PIC Version des Programms gezeigt.

Wenn Sie "Alles Lesen" drücken, werden Sie alle SV Register des LocoHDL Moduls lesen.

Die SV Information wird dann automatisch übersetzt zu Pforte-Adresse und Funktion.



Es entfernt alle SV und Einstellungen und macht der Schirm fertig für eine neue Konfiguration.



Für Druckknöpfe können Sie aus zwei Methoden wählen.

Dieser Einstellung gilt für ALLE Druckknöpfe auf dem LocoHDL Modul.

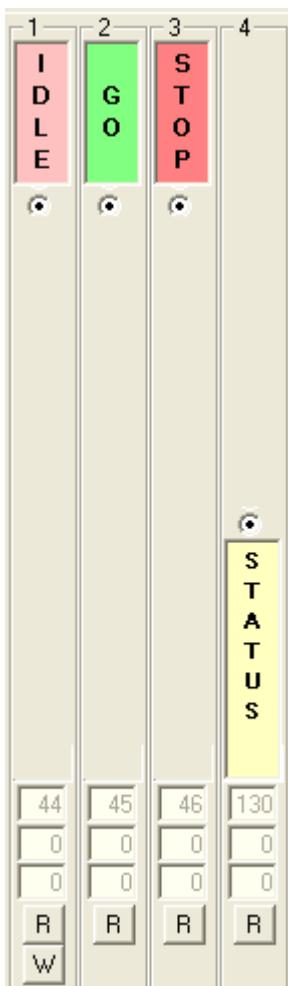
"Fest Kode" bedeutet, dass ein Druckknopf einen Ausgang an oder ausschalten kann. Das heißt, Sie benötigen 2 Druckknöpfe, um einen Ausgang zu schalten.

"Wechselnde Kode" bedeutet, dass jede Zeit die Taste betätigt wird, die Funktion wechselt zwischen einen Ausgang an oder ausschalten.

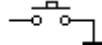
! Die "Fest Kode" Option wird geraten, weil in diesem Fall Sie immer sicher sind, was geschieht.

Special Ports

Die „Special Port“ Option gibt auf den ersten 4 Pforte (auf J4) eines LocoIO, LocoServo oder LocoBuffer Moduls die Möglichkeit, um eine spezielle Funktion zu gründen.



Pforte 1,2 und 3 sind Eingänge die können versehen werden mit einem Druckknopf.

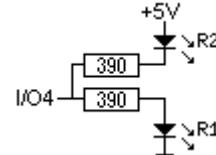


Am Betätigen des Druckknopfs der Pforte 1, schicken sie ein "IDLE" Kommando auf Loconet.

Am Betätigen des Druckknopfs der Pforte 2, schicken sie ein "GO" Kommando auf Loconet. Ein "GO" Kommando sagt im Loconet Netz, dass der Zug plan bereit ist zu benützen.

Am Betätigen des Druckknopfs der Pforte 3, schicken sie ein "STOP" Kommando auf Loconet. Die Zentrale wird den Strom der Verstärker abgeschalten. Sie können diesen Befehl als Notanschlag des Plans sehen.

Pforte 4 ist ein Ausgang, der von einem Stromkreis mit einer roten LED R2 und einer grünen LED R1 versehen werden kann.



Die grüne LED gibt die Anzeige, dass ein "GO" Kommando wird empfangen von Loconet. Die rote LED gibt die Anzeige, dass die "STOP" oder "IDLE" Kommando wird empfangen.

**Info:**

Die "GO" und "STOP" haben die gleiche Funktion wie die Knöpfe mit dem gleichen Namen auf einer Intellibox Zentrale.

<input checked="" type="radio"/>	Pforte nicht benutzt
<input type="radio"/>	Eingang
<input type="radio"/>	Block Kontakt Aktiv Lage
<input type="radio"/>	Block Kontakt Aktiv Hohe
<input type="radio"/>	Block Kontakt Verspätung
<input type="radio"/>	Umschalter
<input type="radio"/>	Druckknopf Aktiv Lage
<input type="radio"/>	Druckknopf Aktiv Hohe
<input type="radio"/>	Direkt <input checked="" type="checkbox"/> Indirekt Kode
<input type="radio"/>	Zurückmeldung Weichenschalter
<input type="radio"/>	Zurückmeldung Weichekontakt 1
<input type="radio"/>	Zurückmeldung Weichekontakt 2
<input type="radio"/>	Doppelter Eingang
Ausgang	
<input type="radio"/>	Block Belegmeldung
<input type="radio"/>	Blinken <input type="button" value="0"/> Frequenz
<input type="radio"/>	1 - Ein
<input type="radio"/>	1 - Aus
<input type="radio"/>	2 - Ein
<input type="radio"/>	2 - Aus
<input type="radio"/>	4 Wege Pforte
<input type="radio"/>	1 Soft Reset
<input type="radio"/>	1 Hard Reset
<input type="radio"/>	2 Soft Reset
<input type="radio"/>	2 Hard Reset

## Poort Einstellungen:

Die unterschiedlichen Pforten eine Funktion zuweisen.

Auf dem LocoIO sind 16 Pforten, auf dem LocoServo sind 4 Pforten und auf dem LocoBooster sind 4 Pforten dass allgemein und unabhängig von einander als Eingang oder Ausgang geschaltet werden kann.

Eine ausgedehnte Beschreibung des Einganges und des Ausganges kann Sie im LocoIO manuel unter Kapitel „LocoIO Möglichkeiten“ finden

## ! EMPFEHLUNG:

- Wenn die verwendete Version von LocoIO nicht die Option Vorwähler „Pforte nicht benutzt“ habe, dann wird es geraten unbenutzte Pforte als Ausgang zu definieren, und sie eine unbenutzte Adresse zu geben. Damit verhindern Sie, dass bei Störungen auf dem Modul, ungewünschte Codes auf Loconet gesandt werden.
- Geben Sie die Pforte, die als Servo Ausgang definiert ist, immer eine Adresse die nur einmal auf einem Modul auftritt.

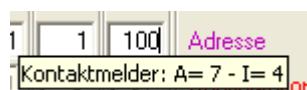
## Konfigurationsknöpfe:



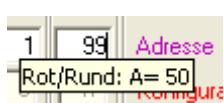
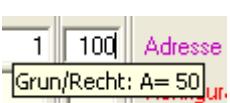
Die Blinkrate kann zwischen 0 und 15 gestellt werden.  
Wenn man diese Einstellung ändert, werden ALLE Blinkfunktionen auf dieses LocoHDL Modul geändert.



Dies ist die Adresse von einem Signal, Weiche, Blockdetektor, ...  
Dieses ist die Adresse des jeweiligen Pforten, das in der Zug Steuerung Software benutzt wird.  
Entsprechend den Loconet Spezifikationen können Signalen und weichen einer Adresse haben von 1 bis 2048.  
Belegmeldung oder Druckknöpfen haben ein Adresse bereich von 1 bis 4096.



Tooltip Text gibt es in Übereinstimmung mit Adresse und Eingang wie an S88.  
Beispiel: Loconet Adresse 100 = S88 Adresse 7 und Eingang 4



Tooltip Text eines Druckknopfs gibt die Adresse des Signals und des Weichen, die sie mit seiner Situation schaltet.



Tooltip Text des örtlich fest- oder Pulskontaktes zeigt mit an, welcher Druckknopf Sie diese schalten können.  
Beispiel: Loconet Adresse 100 Kontakt 1 = Druckknopf 199

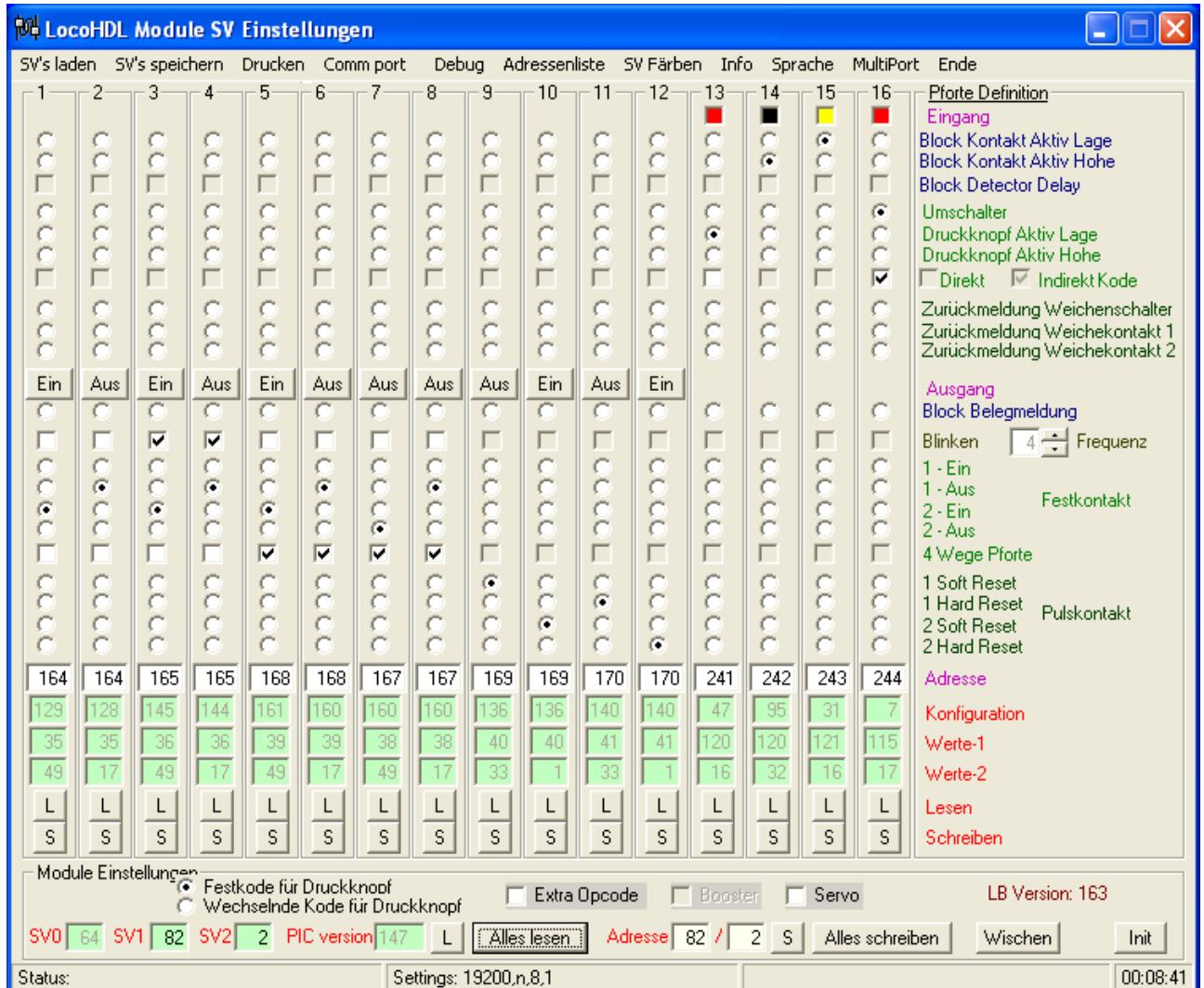


Für jedes Pforte oder LocoHDL Modul mit Adresse angedeutet in SV1/SV2, gibt es die Möglichkeit die Konfiguration, der Wert1 und der Wert2 des Registers zu lesen und zu speichern.

Wenn die SV Register einer Ausgabe gut gelesen sind, wird eine Testtaste erscheinen. Mit dieser Testtaste können Sie die Ausgabe ändern, um die Wirkung zu prüfen.



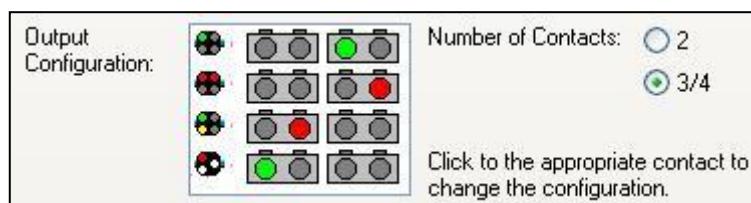
# LocoHDL Konfiguration Programm Vorbild eines LocoIO:



IO 1 und 2 ist ein rot/grün Signal (1=rot, 2= grün)

IO 3 und 4 ist ein blinkerndes rot/grün Signal

IO 5,6,7 und 8 ist ein 4-Wege-Signal



Railroad & Co TrainController Einstellungen.

IO 9 und 10 ist eine Weiche mit Spulen nur mit Softwarepulslänge

IO 11 und 12 ist eine Weiche mit Spulen mit Software und/oder Hardware Pulslänge

IO 13 ist einer Druckknopf aktiv niedrig, der OPC\_SW\_REQ (0xB0)-Kodes schicken

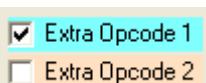
IO 14 ist eine Blockaufspürung aktiv hoch

IO 15 ist eine Blockaufspürung aktiv niedrig

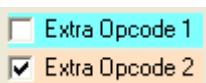
IO 16 ist ein Kippschalter, der OPC\_SW REP (0xB1)-Kodes schicken

# LocoIO, LocoServo und LocoBooster extra Op-kode Option für Eingang

Ein Opcode ist ein Loconet Kommando senden über das Loconet Netz



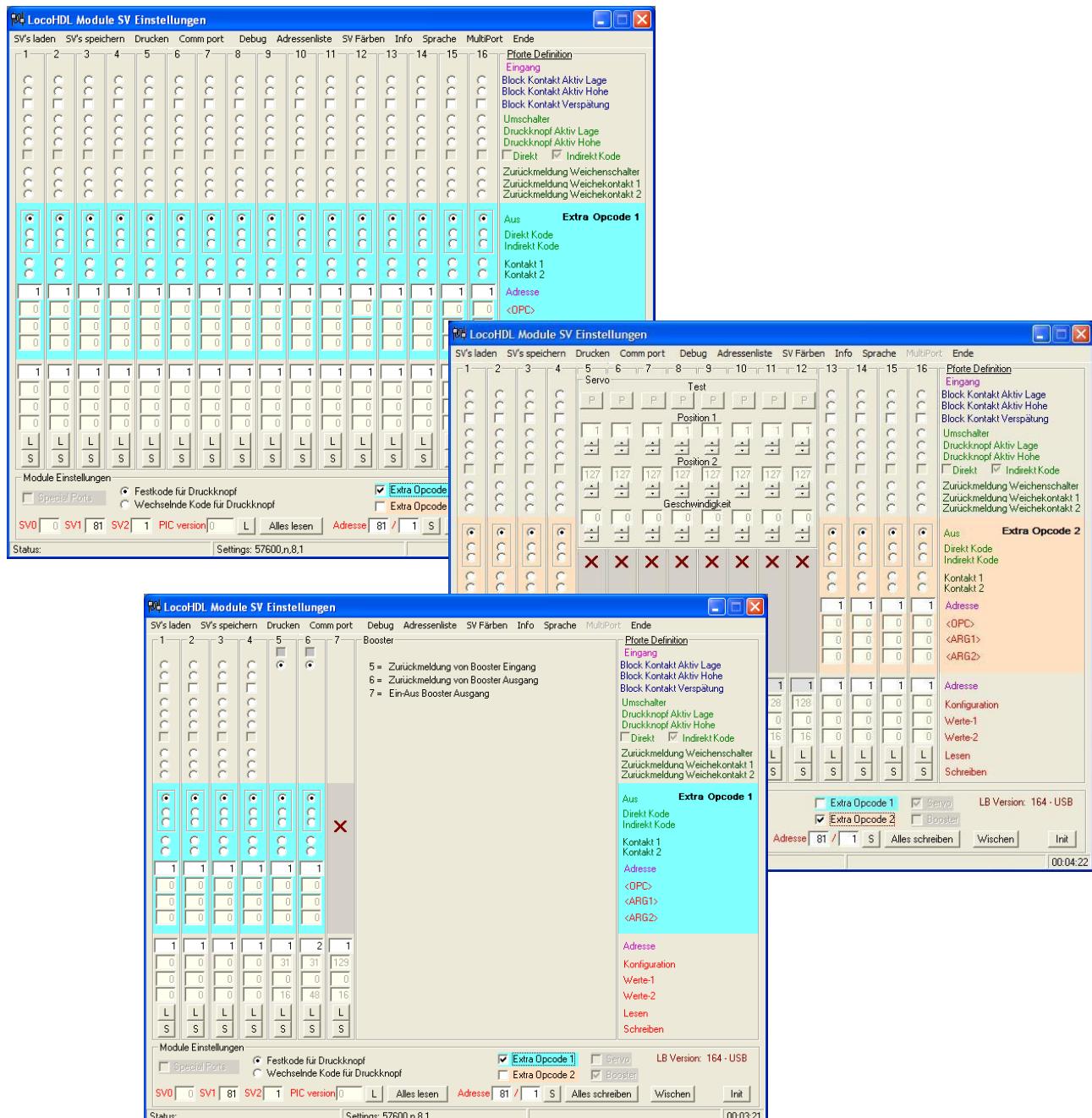
Mit diesem Knopf können sie ein fakultativer Anzeigeraum sehen mit Einstellungen von SV51 bis SV98. Beiliegend wird zusätzliches direktes oder indirektes Kommando nach Loconet geschickt. Das Extra Opcode 1 sendet, in dem Augenblick als der Druckknopf betätigt wird.

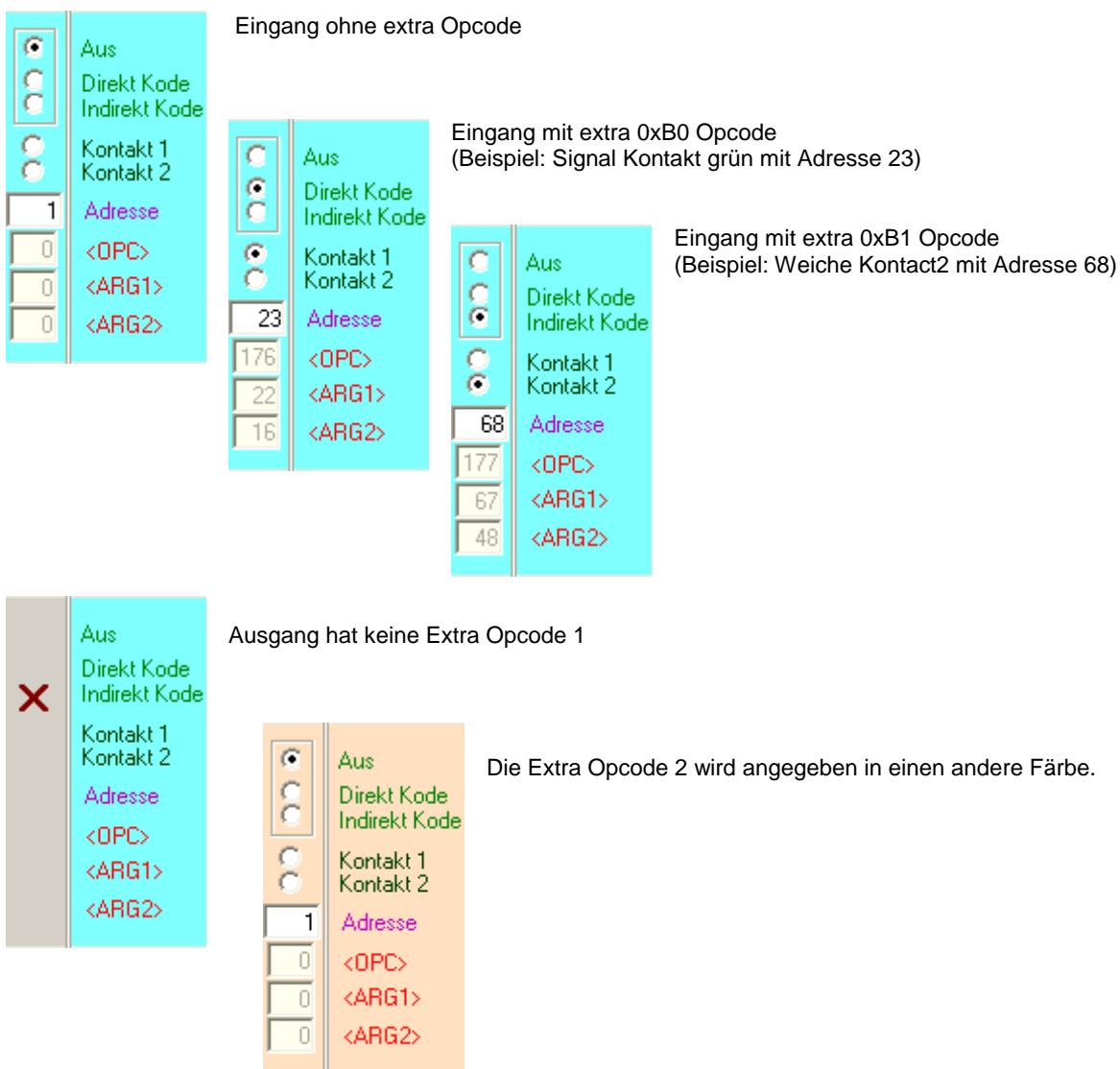


Mit diesem Knopf können sie ein fakultativer Anzeigeraum sehen mit Einstellungen von SV128 bis SV175. Das Extra Opcode 2 hat die gleiche Funktionalität wie das Extra Opcode 1 aber ist senden, wann der Druckknopf freigegeben wird.

Die Extraopcodes gibt die Möglichkeit an 1 Druckknopf, um 1, 2 oder 3 Loconet Kommandos zu geben.

Beispiel: mit dem Betätigen des Druckknopfs kann ein Weiche schalten, das Extra Opcode 1 einen zweiten Weiche schalten und am Freigeben des Druckknopfs kann das Extra Opcode 2 ein grünes Signal einstellen.





## Doppelter Eingang:

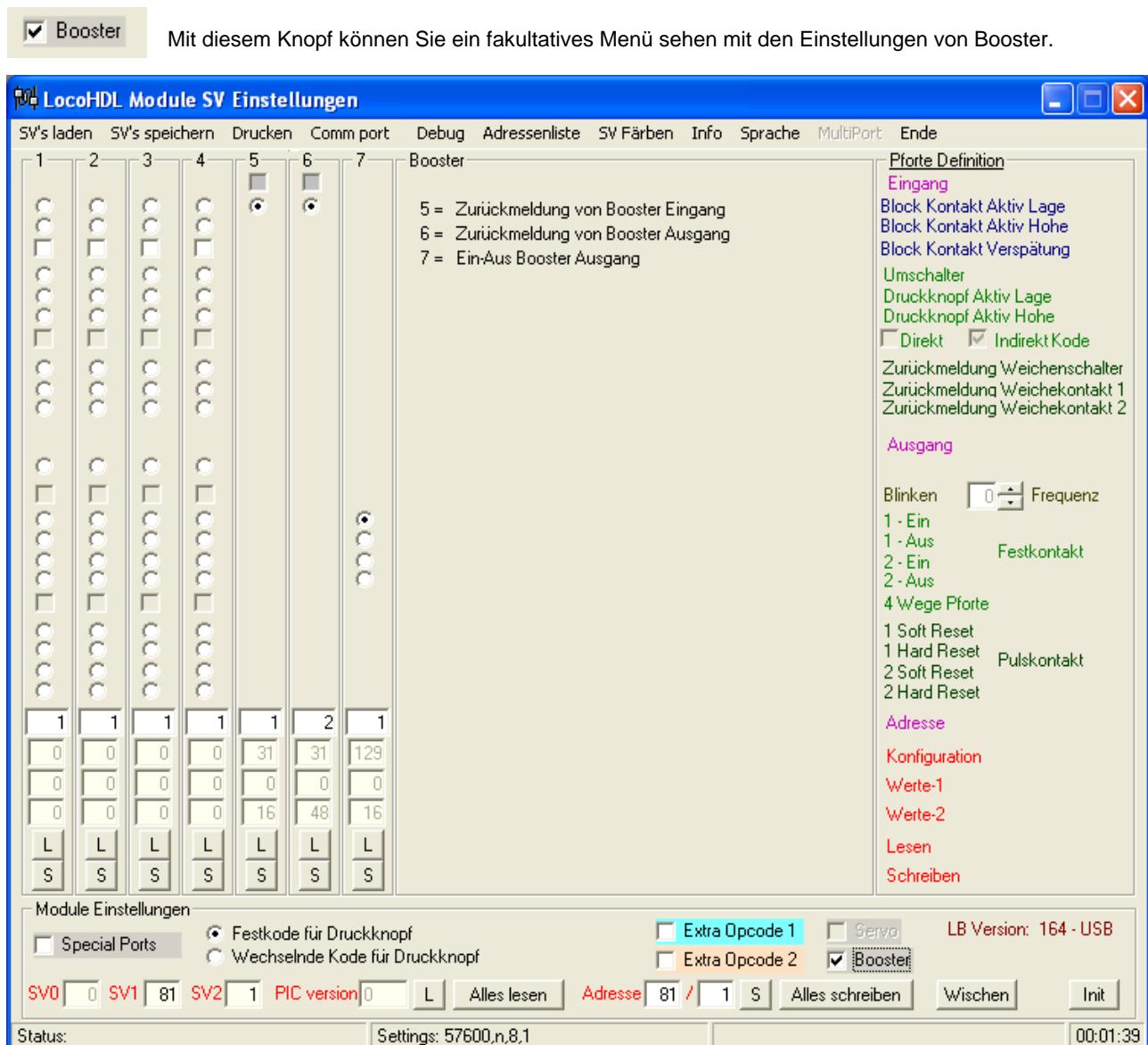
„Am doppelten Eingang“ wird ein gerade Eingang zu einem ungeraden Eingang verbunden (2 bis 1, 4 bis 3,..., 16 bis 15).



Vorbild mit Pforte 15 und 16, wie auf die Bild:

- Wenn Pforte 16 niedrig ist (0V), dann am aktiven Kommen von Pforte 15 wird ein Blockabfragungs-Code Aktive mit Adresse von Pforte 15 und das Extra Opcode 1 von Pforte 15 an Loconet zu gewesen
- Wenn Pforte 16 niedrig ist (0V), dann am inaktivierten Kommen von Pforte 15 wird ein Blockabfragungs-Code Inaktive mit Adresse von Pforte 15 und das Extra Opcode 1 von Pforte 15 an Loconet zu gewesen
- Wenn Pforte 16 hoch ist (5V), dann am aktiven Kommen von Pforte 15 wird ein Blockabfragungs-Code Aktive mit Adresse von Pforte 15 und das Extra Opcode 1 von Pforte 16 an Loconet zu gewesen
- Wenn Pforte 16 hoch ist (5V), dann am inaktivierten Kommen von Pforte 15 wird ein Blockabfragungs-Code Inaktive mit Adresse von Pforte 15 und das Extra Opcode 1 von Pforte 16 an Loconet zu gewesen

## LocoHDL Module Konfiguration für L-Booster und N-Booster



Was die Funktionalität betrifft sind die ersten 4 Pforten identisch mit dem LocoIO.

Pforte 5 ist fest eingestellt auf eine Belegmeldung, und meldet, dass es ein Eingabesignal gibt für den Booster.

Pforte 6 ist fest eingestellt auf eine Belegmeldung, und meldet, dass das Ausgabesignal OK ist.

Das bedeutet, dass es kein Kurzschluss gibt, und ein Eingangssignal anwesend ist. Das bedeutet aber nicht, dass der Boosterausgang darum angeschaltet ist (siehe Pforte 7).

Pforte 7 ist eine Fest Kontakt-Ausgang, die den Boosterausgang EIN oder AUS schaltet mittels eines Relais.

Das Relais wird NICHT angeschaltet, wenn bei einem Fest Kontakt-Ausgang = EIN eine oder mehrere Belegmeldungen von Pforte 5 und Pforte 6 abwesend sind oder eines OPC\_GPOFF (0x82) oder OPC\_IDLE (0x85) Befehl empfangen ist. Mit einem "1-An Fest Kontakt" oder "2-An Fest Kontakt" wird den Booster, nach dem Start und bei Empfang eines Eingabesignals, automatisch den Ausgabe anschalten.

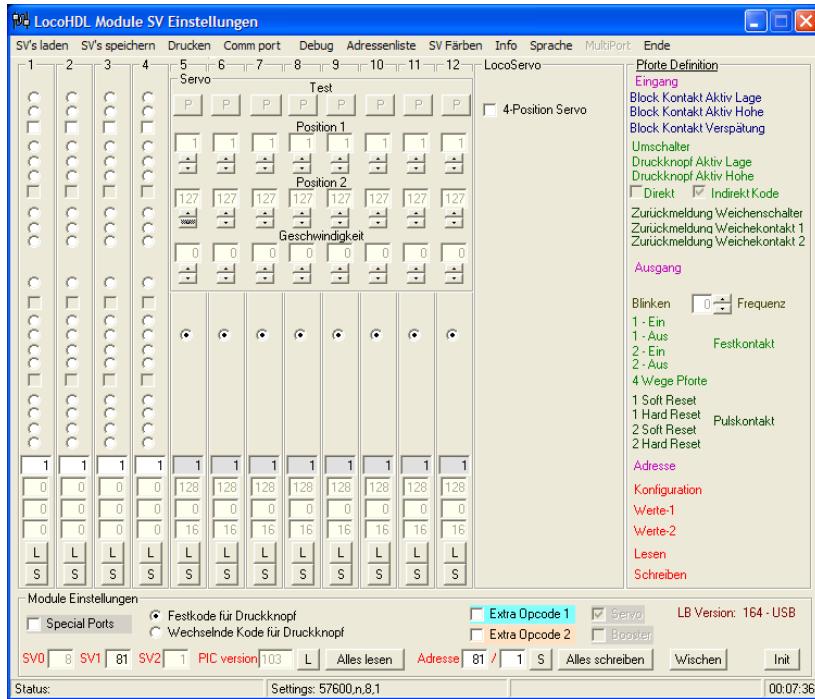
Mit einem "1-Aus Fest Kontakt" or "2-Aus Fest Kontakt" muss den Booster mittels eines Kommandos angeschaltet werden.

Einige der Möglichkeiten für Pforte 7:

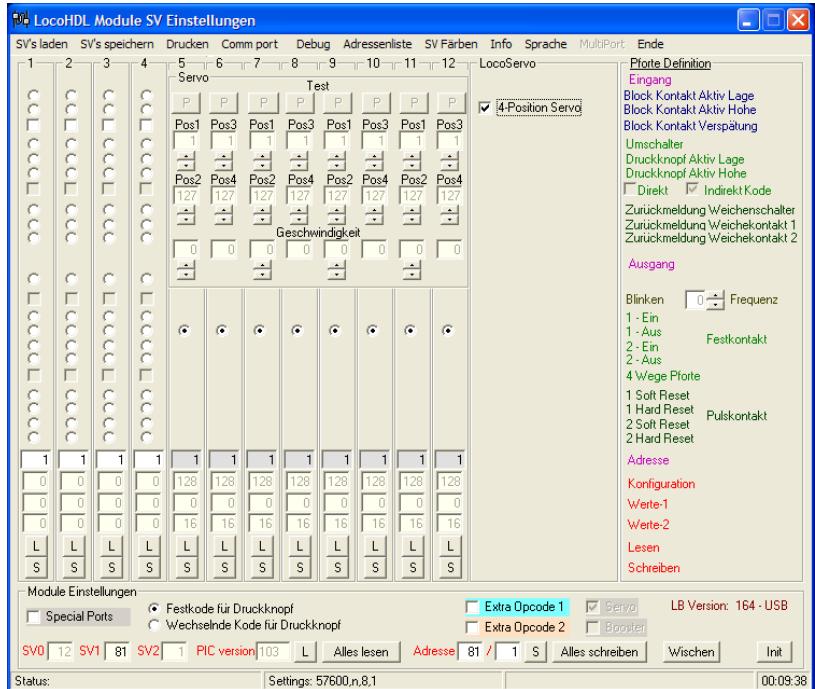
- Sie kann gekoppelt werden an einem Signal, um den Strom einer Bahn zu sperren.
- Individuelle Bahnstücke unterbrechen in Notfalle.

# LocoHDL Konfiguration für LocoServo Module

Servo Wenn die Registers von einem LocoServo bereits gelesen würde, wird diese Taste automatisch vorgewählt.



Das Servo kann in einen Winkel von 90° von Position 1 = 1 in Position 2 = 127 sich drehen.  
Die Position 1 und die Position 2 können innerhalb dessen 90° hergestellt werden mit einem Wert von 1 bis 127.  
Die Servo Motoren werden von einer Position auf die andere durch einen Report "fest Kontaktes" verschoben.  
Versetzung kann mit 4 unterschiedlichen Geschwindigkeiten geschehen.



Am Voreinstellen des Servos 4-Position, können 4 Servos zum LocoServo gefahren werden.

## LocoServo Module

Komplette Erklärung und Anschluss der Servo Motor zu LocoIO können im Servo Module Betriebsanleitung gefunden werden.

Hans Deloof

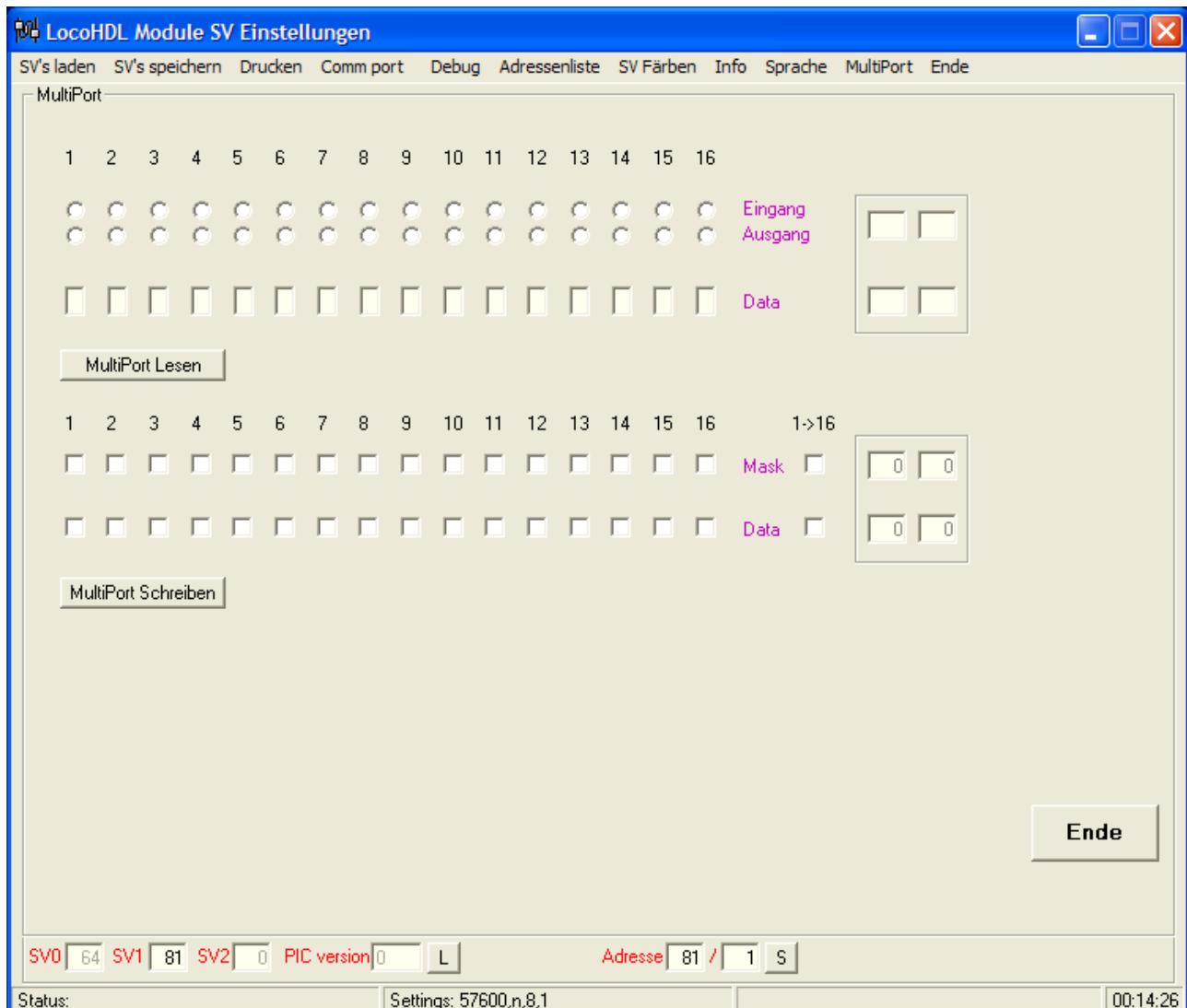
hans.deloof@compaqnet.be

<http://users.telenet.be/deloof/index.html>

Version 24/08/2012

# LocoHDL Configuration MultiPort Kommando test

Dieses arbeitet nur auf LocoIO Modulen.



MultiPort erlaubt Ihnen, alle Tore mit dort Funktionalität zu lesen und einiges oder alle Tore gleichzeitig zu schreiben. Ein MultiPort Befehl Gebrauch eine OPC\_PEER\_XFER Anzeige.

Verwenden der gelesenen Portfunktion: Der Befehl ist "04 gelesene Tore"

Die zurückgebrachten Werte sind:

Die gleiche Direktion trifft auf die hohe Maske (D7) und das hohe Datenbyte (D8) für Tore 9 bis 16 zu.

D5 niedrige Maske: Tor 1-8 mit 1-Tor ist Ausgang, 0 -Tor ist Eingang

D6 niedrige Daten: Tor 1-8 mit Wert von Tordaten

D7 hohe Maske: Tor 9-16 mit 1-Tor ist Ausgang, 0 -Tor ist Eingang

D8 hohe Daten: Tor 9-16 mit Wert von Tordaten

Verwenden der Schreibentorfunktion: Der Befehl ist "03 schreiben Tore"

Die Funktion Schreiben arbeitet nur auf Tore, die als Ausgang definiert werden.

Um diese Funktion zu verwenden, schicken Sie dem LocoIO 4 Bytes.

Diese Bytes werden in das D5 bis D8 Bytes der OPC\_PEER\_XFER Anzeige gesendet.

Das erste Byte (D5) ist die niedrige Maske und die Diagramme zu den Toren 1 bis 8.

Wenn eine Spitze dann eingestellt wird, wird das Tor mit den Daten im niedrigen Datenbyte (D6) aktualisiert.

Wenn die Bit null in ihr hat, wird das Tor nicht geändert.

Die gleiche Funktion trifft auf die hohe Maske (D7) und das hohe Datenbyte (D8) für Tore 9 bis 16 zu.

# 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

### **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  
 2 / 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  
 0 1 1 1 0 1 1  
 0 1 1 1 0 1 1  
 0 1 1 1 0 1 1  
 0 1 1 1 0 1 1  
 0 1 1 1 0 1 1

## **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

**Dipl.-Ing. Wolfgang Hückel**

Am Brunnen 1

42781 Haan

☎ +49 2129 2860

[mailto:wolfgang\\_hueckel@hotmail.com](mailto:wolfgang_hueckel@hotmail.com)

**wLocoIO-2**

Modelleisenbahn-Steuerung über das Loconet

Version 1.6

30. Mai 2019

**Wichtige Hinweise**

wLocoIO-2 ist für den Einsatz auf Modelleisenbahnanlagen vorgesehen. Diese Anleitung beschreibt die Funktion, den Aufbau und die Inbetriebnahme der Schaltung.

Der Autor dieser Anleitung übernimmt aber keine Haftung für den Aufbau und die Funktion von wLocoIO-2 sowie beliebige Schäden, die aus oder in Folge Aufbau oder Betrieb von wLocoIO-2 entstehen.

## 1 Inhaltsverzeichnis

1	Inhaltsverzeichnis.....	2
2	Was macht man mit LocoIO? .....	3
3	Loconet.....	3
4	Wozu gibt es dann wLocoIO-2?.....	4
5	Schaltung.....	4
	Abweichungen vom Original .....	6
	Schaltungsbeschreibung.....	6
6	Platine.....	7
	Bestückung .....	7
	Farbkodes für die Widerstände.....	8
	Bauteile.....	9
	Programmieren des Mikrocontrollers .....	9
7	Aufbau .....	9
8	Inbetriebnahme.....	10
	Initialisierung .....	11
	Schaltungskonfiguration.....	12
	Normaler Betrieb.....	18
9	Quellen .....	19
10	Versionierung des Dokuments .....	20

## 2 Was macht man mit LocoIO?

Der Begriff „LocoIO“ ist bei der Modelleisenbahn schon recht alt. Bereits im Jahre 2003 veröffentlichte John Jabour unter diesem Namen eine Schaltung, mit der man über das Loconet elektrisch betriebenes Modellbahnzubehör steuern kann. Das interessante dabei war, dass diese Schaltung in vielen Punkten zur Laufzeit konfigurierbar ist (ähnlich wie bei Digital-Decodern). Herzstück der Schaltung ist ein PIC-Mikrocontroller von Microchip. Das Loconet-Interface ist einfach und robust, arbeitet sehr zuverlässig und kommt völlig ohne Abgleich aus. LocoIO hat 16 Anschlüsse, jeder Anschluss kann über die Konfiguration entweder ein Eingang oder ein Ausgang sein. Außerdem wird konfiguriert, mit welchem anderen Loconet-Anschluss er korrespondieren soll, dazu wird dem Anschluss noch eine Adresse zugewiesen.

An einen Eingang schließt man einen Sensor an. Das ist ein elektrisches Gerät, das die Information „Sensor ist nicht aktiv“ oder „Sensor ist aktiv“ liefert. Beispiele für Sensoren sind Schalter und Rückmelder. Wenn sich der Zustand des Sensors ändert, wird die neue Information an die dem Anschluss zugeordnete Adresse geschickt.

An einen Ausgang schließt man einen Aktor an. Das ist ein elektrisches Gerät, das die Information des Anschlusses weiterverarbeitet. Beispiele für Aktoren sind Weichen, Signale und Lampen, aber auch Kontrollleuchten.

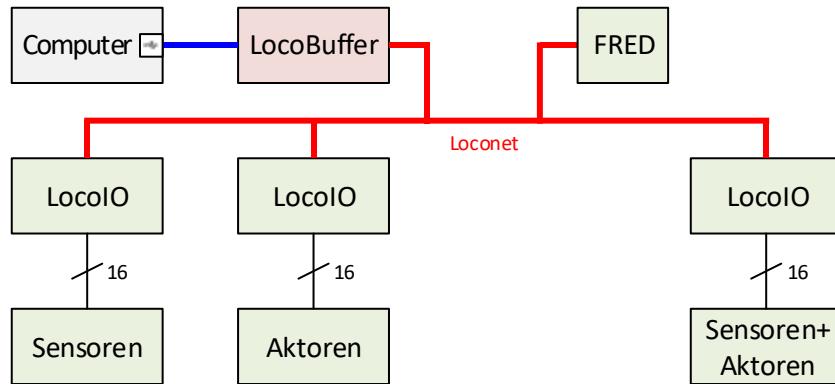
Leider hat John Jabour das Projekt nicht weiter gepflegt, und er wollte auch nicht das PIC-Programm veröffentlichen. Später hat Hans de Loof (auch als HDL bekannt) die Schaltung von John Jabour als Basis für ein ganzes Schaltungssystem übernommen. Das System ist so ausgelegt, dass damit viele der von der Modellbahn-Industrie angebotenen Komponenten gesteuert werden können. Außerdem hat er alles kommerzialisiert und bietet über die Firma Modelbouw HET SPOOR Platinen, Bausätze und fertig aufgebaute Bausteine an. Die Firmware für die Mikrocontroller konnte man anfangs auch von seinen Internetseiten herunterladen, um die Mikrocontroller selbst zu programmieren. Für die neuesten Produkte steht aber leider meistens keine Firmware mehr zur Verfügung, da muss man fertig programmierte PICs kaufen. Das HDL-System wird immer wieder mal weiter ausgebaut.

LocoIO wird auch von verschiedenen Modellbahn-Steuerprogrammen unterstützt. Die Einstellungen können der Schaltung meistens direkt aus dem Steuerprogramm zugewiesen werden. Wen das interessiert, der muss sich allerdings bei den Herstellern der Programme informieren, hier kann das alles nicht behandelt werden.

## 3 Loconet

Was ist denn das Loconet überhaupt? In diesem Rahmen kann nur eine kleine Einführung gegeben werden, denn eine komplette Erklärung würde den Rahmen dieser Beschreibung überschreiten. Wer mehr wissen will, der sei auf die Original-Beschreibung verwiesen.

Das Loconet wurde von dem amerikanischen Modellbahnhersteller Digitrax erfunden. Es handelt sich um ein einfaches Netzwerk zur Verbindung von Modelleisenbahn-Komponenten, das auf Grund der langsamen Datenübertragung nur sehr geringe Anforderungen an die technische Ausführung stellt. Deshalb kann das Loconet auch in fast beliebiger Netz-Topologie aufgebaut werden, ausgenommen ist nur der Ring. Über die Loconet-Datenleitung werden von Digitrax festgelegte Informationen geschickt, die jeder Teilnehmer auswertet. Wenn die Daten für die Schaltung bestimmt sind, die die Informationen gerade liest, reagiert sie entsprechend, ansonsten wird es ignoriert. So kann man bei einem entlang einer Modellbahnanlage verlegten Loconet sehr leicht fernbediente Steuerungen vornehmen wie zum Beispiel ein weiter weg liegendes Einfahrtsignal eines Bahnhofs oder mehrere irgendwo auf der Strecke befindlichen Schranken, die von einer Schrankenwärter-Betriebsstelle bedient werden.



Über eine entsprechende Schnittstelle wie zum Beispiel der LocoBuffer kann das Loconet auch mit einem Computer verbunden werden. So lassen sich mit einem handelsüblichen Modellbahn-Steuerprogramm über das Loconet Weichen und Signale stellen (Aktoren) und Schalter und Rückmelder abfragen (Sensoren). Es unterstützen allerdings nicht alle auf dem Markt angebotenen Steuerprogramme das Loconet, hier muss man also genau hinschauen.

Der Computer-Anschluss ist für die eigentliche Funktion von LocoIO während des Betriebs nicht notwendig, den benötigt man nur zur Initialisierung und zur Konfiguration.

## 4 Wozu gibt es dann wLocoIO-2?

Da das Original nicht kommerziell vertrieben wurde, und die Produkte von de Loof bei größerer Stückzahl in Summe doch recht teuer sind, wurde LocoIO in einer vereinfachten Variante nachgebaut. Zur Unterscheidung vom Original wurde die Schaltung wLocoIO getauft. Die erste Version war noch eine handgeätzte einseitige Platine mit sehr vielen Drahtbrücken. Vorteilhaft war, dass die Firmware für den Mikrocontroller zum Selbstprogrammieren zur Verfügung stand. Das führte zum sofortigen Erfolg, nur der Aufbau der Schaltung war noch wegen der vielen Brücken recht zeitintensiv, und Platinen ätzen zu Hause ist auch nicht so toll.

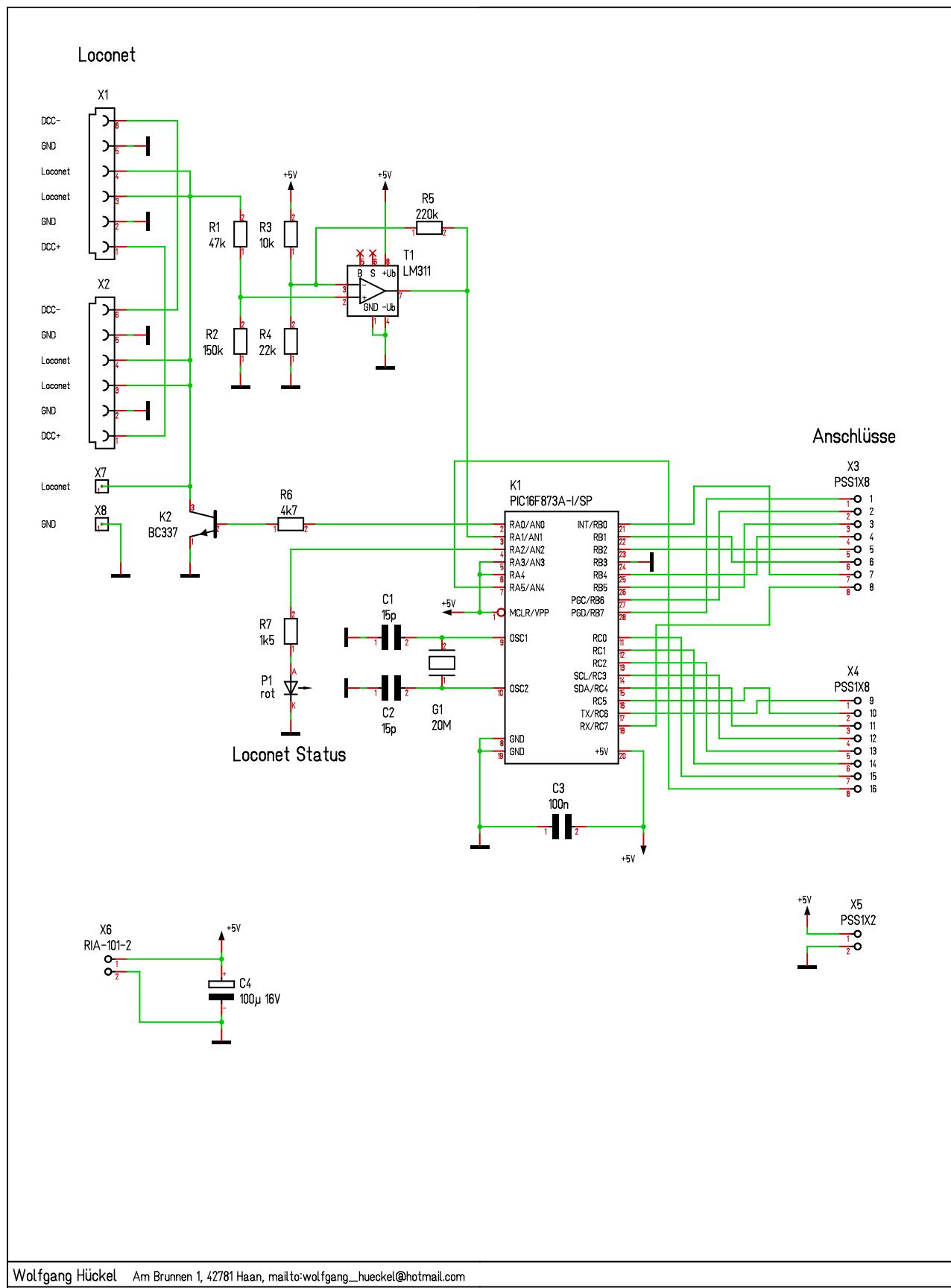
Daher wurde beschlossen, die Platine in größerer Stückzahl industriell fertigen zu lassen. So konnte sie auch zweiseitig werden, und die vielen Drahtbrücken entfielen. Um die Kosten für die Platinen möglichst gering zu halten, wurde die Fläche soweit es ging minimiert. Praktische Gründe veränderten dann noch ein paar Kleinigkeiten gegenüber dem Original, so dass schließlich der Name wLocoIO-2 gerechtfertigt war.

## 5 Schaltung

Die Schaltung wLocoIO-2 ist 100% kompatibel zur Schaltung LocoIO von John Jabour und zur Schaltung HDL LocoIO (HDM 08) von de Loof.

Der Mikrocontroller sollte mit der Firmware 1.48 von de Loof programmiert werden. Das ist die letzte Version, die mit dem letzten frei verfügbaren de Loof-Konfigurationsprogramm LocoHDL 3.5.2 funktioniert.

Es gab noch die 1.49 zum Selbstprogrammieren, aber LocoHDL 3.5.2 funktioniert damit nicht und de Loof stellt für diese Version kein Konfigurationsprogramm mehr zur Verfügung. Der Funktionsumfang der Version 1.48 ist aber für die gedachten Zwecke völlig ausreichend.



## ***Abweichungen vom Original***

Die de Loof-Schaltung wurde in einigen Punkten vereinfacht:

- Die lokale Erzeugung der 5 V-Spannung ist komplett entfallen, weil meistens mehrere solcher Platinen an einer Stelle eingesetzt werden, dann ist es sinnvoller, die Betriebsspannung zentral bereitzustellen. Für eine Einzelplatine kann man auch besser ein handelsübliches 5 V-Netzteil verwenden, das ist weniger Aufwand als ein Modellbahnttransformator mit nachgeschaltetem 5 V-Regler.
- Es wurde auf die Leuchtdiode zur Anzeige der Existenz der Betriebsspannung verzichtet. Wenn die Schaltung einmal läuft, schaut da eh keiner mehr hin.
- Für den Anschluss der Leitungen von und zur Modellbahnanlage sind bei de Loof neben den Einzelleitungen auch noch RJ 12-Buchsen möglich, um die Erweiterungsplatinen anzuschließen. Das wird bei wLocoIO-2 nicht gebraucht, und diese RJ 12-Buchsen wurden weggelassen, um die Platinenfläche zu verkleinern.
- Auf die Möglichkeit des Einbaus von Pull-Up-Widerständen an den Anschlüssen X3 und X4 wurde verzichtet, um Platinenfläche einzusparen.

## ***Schaltungsbeschreibung***

Bei X6 wird die 5 V-Versorgungsspannung eingespeist. Der Kondensator C4 ist ein Puffer gegen kurzzeitige Instabilitäten der Versorgung.

Das Loconet wird bei X1 und X2 angeschlossen. Es gibt zwei Buchsen, damit das Netzwerk bequem von einer Platine nur nächsten weitergereicht werden kann. Bei einer festen Installation kann man auch auf die Steckverbindung verzichten und X1 und X2 weglassen. Das Loconet wird dann über die Lötnägel X7 und X8 angeschlossen. Der Operationsverstärker T1 und die fünf Widerstände R1 bis R5 bilden den Loconet-Empfänger, der die Loconet-Daten in für den Mikrocontroller verständliche Logik-Pegel umwandelt. R6 und K2 bilden den Loconet-Ausgang.

Die Leuchtdiode P1 und der dazugehörige Vorwiderstand R7 zeigen die Aktivität oder Fehler auf dem Loconet an.

Herzstück der Schaltung ist der Mikrocontroller K1, programmiert mit der dazugehörigen Firmware. Der Kondensator C3 ist wieder ein Puffer. Der Mikrocontroller benötigt bei der vorgesehenen Betriebsfrequenz von 20 MHz drei externe Bauteile, den Quarz G1 und den beiden Kondensatoren C1 und C2.

An den Anschlussleisten X3 und X4 stehen die 16 Anschlüsse von und zur Modellbahnanlage zur Verfügung. Auf Grund der Leiterbahn-Anordnung liegen jetzt die Leitungen 9-16 links von 1-8. Das darf man aber nicht anders benennen, sonst stimmt die Anschlussnummern nicht mehr mit der Firmware überein.

Manchmal benötigt man auch die 5 V-Spannung an Bausteinen, die direkt an der Anlage eingebaut werden, diese Spannung kann man an der Anschlussleiste X5 abgreifen.

## 6 Platine

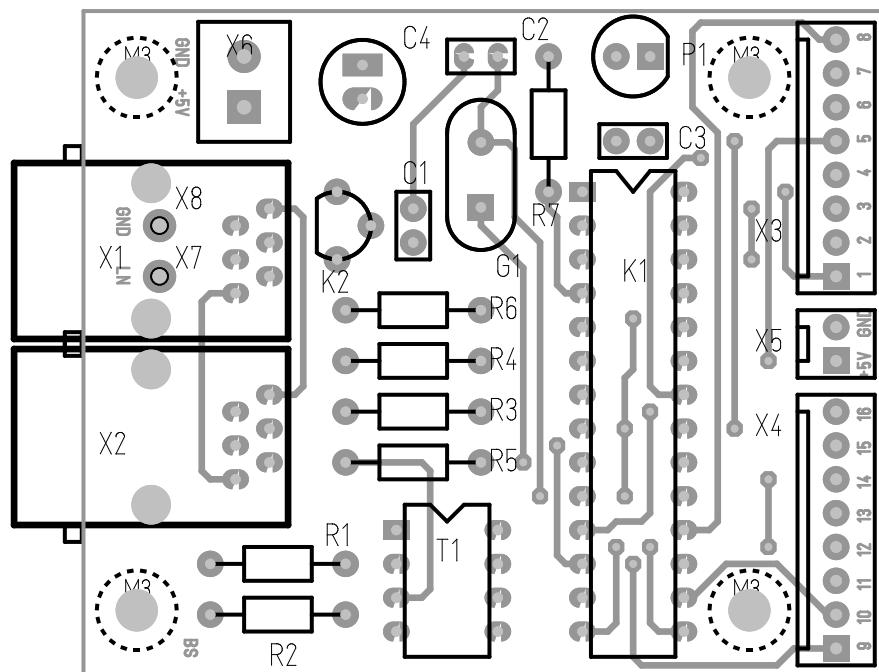
Auf der Platine sind auf beiden Seiten Leiterbahnen vorhanden. Zum Zeitpunkt der ersten industriellen Fertigung der Platine war der Bestückungsdruck noch sehr teuer, deshalb wurde darauf verzichtet. Wichtige Informationen wie die Kennzeichnung der Anschlüsse wurden aber trotzdem auf den Leiterbahnseiten „beschriftet“. Für die zweite Auflage konnte ein anderer Platinen-Lieferant gefunden werden, der auch den Bestückungsaufdruck ohne Mehrkosten dabei hat.

Die Größe der Platine beträgt 60 x 50 mm, sie ist 1,55 mm dick. Die Kupferstärke beträgt 35 µm. Beide Leiterbahnseiten sind mit Lötstopplack überzogen, so dass nur die Lötstellen offen sind. Das erleichtert die Montage der Bauteile.

Es sind vier Befestigungsbohrungen mit 3,2 mm Ø vorhanden.

### Bestückung

Alle Bauteile werden auf der hier abgebildeten Platinenseite BS (= Bestückungsseite) montiert.



## Stückliste

Widerstände					
R1	Widerstand	47 kΩ	Größe 0207		
R2	Widerstand	150 kΩ	Größe 0207		
R3	Widerstand	10 kΩ	Größe 0207		
R4	Widerstand	22 kΩ	Größe 0207		
R5	Widerstand	220 kΩ	Größe 0207		
R6	Widerstand	4,7 kΩ	Größe 0207		
R7	Widerstand	1,5 kΩ	Größe 0207		
Kondensatoren					
C1	Kondensator	15 pF	Keramik RM 2,54		
C2	Kondensator	15 pF	Keramik RM 2,54		
C3	Kondensator	100 nF	Vielschicht-Keramik RM 2,54		
C4	Elektrolytkondensator	100 µF 16 V	radial RM 2,54		
Halbleiter					
K1	Mikrocontroller	PIC 16 F 873 A-I/SP			
K1	IC-Fassung	DIL 28 S			
K2	Transistor	BC 337-40			
P1	Leuchtdiode	rot 3 mm Ø 2 mA			
T1	Operationsverstärker	LM 311			
T1	IC-Fassung	DIL 8			
sonstige Bauteile					
G1	Quarz	20 MHz HC 18 oder HC 49 U/S			
X1	Buchse	Western RJ 12 (alternativ Lötnägel X7 und X8)			
X2	Buchse	Western RJ 12 (alternativ Lötnägel X7 und X8)			
X3	Stiftleiste	PSS oder SL 1 x 8 RM 2,54			
X4	Stiftleiste	PSS oder SL 1 x 8 RM 2,54			
X5	Stiftleiste	PSS oder SL 1 x 2 RM 2,54			
X6	Anschlussklemme	RIA 1 x 2 RM 3,81 oder 2 Lötnagel 1 mm Ø			
X7	Lötnagel	1 mm Ø, nur bei Verzicht auf X1 und X2			
X8	Lötnagel	1 mm Ø, nur bei Verzicht auf X1 und X2			
Platine	wLocoIO-2				

## Farbkodes für die Widerstände

Wert	5 % Toleranz					1 % Toleranz				
	Braun	Grün	Rot	Gold	Braun	Grün	Schwarz	Braun	Braun	
1,5 kΩ	Braun	Grün	Rot	Gold	Braun	Grün	Schwarz	Braun	Braun	
4,7 kΩ	Gelb	Lila	Rot	Gold	Gelb	Lila	Schwarz	Braun	Braun	
10 kΩ	Braun	Schwarz	Orange	Gold	Braun	Schwarz	Schwarz	Rot	Braun	
22 kΩ	Rot	Rot	Orange	Gold	Rot	Rot	Schwarz	Rot	Braun	
47 kΩ	Gelb	Lila	Orange	Gold	Gelb	Lila	Schwarz	Rot	Braun	
150 kΩ	Braun	Schwarz	Gold	Gold	Braun	Grün	Schwarz	Orange	Braun	
220 kΩ	Rot	Rot	Gold	Gold	Rot	Rot	Schwarz	Orange	Braun	

Eine Toleranz von 5% reicht aus. Gegen die Verwendung von 1%-Metallfilmwiderständen gibt es aber keine Einwände.

## Bauteile

Alle Bauteile sind in der Regel im Elektronikhandel erhältlich.

Den Quarz gibt es in verschiedenen Gehäusehöhen, Grundform und Anschlussdrähte sind gleich. Das Gehäuse HC 18 ist deutlich höher. Der größere Quarz ist zwar im Prinzip genauer, aber das spielt hier überhaupt keine Rolle. Das Gehäuse HC 49 U/S kann bedenkenlos verwendet werden.

Die Lötnägel heißen bei einigen Lieferanten auch RTM-Stifte.

Die Buchsen X1 und X2 für das Loconet können bei einer Festinstallation auch durch die Lötnägel X7 und X8 ersetzt werden, nur X2 lässt sich zusätzlich montieren.

Die Anschlussleisten X3, X4 und X5 sind im Raster 2,54 mm gehalten. Es passen entweder verdrehsichere PSS-Steckverbinder hinein, oder wenn keine trennbare Verbindung benötigt wird, setzt man dort einreihige Stiftleisten SL ein. Solche Leisten haben je nach Hersteller und Lieferant verschiedene Bezeichnungen, wichtig ist also, dass das Rastermaß stimmt.

Der Anschluss der Energieversorgung X6 ist im Raster 3,81 mm gehalten, da verwendet man entweder eine zweipolige RIA-Steckverbindung oder setzt Lötnägel ein.

Die beiden Integrierten Schaltungen T1 und K1 sollten mit einer Steckfassung versehen werden. Die Verwendung einer Fassung hat keine schädlichen Auswirkungen auf die Funktion der Schaltung. Wenn man für K1 keine Fassung einsetzen will, dann muss der Mikrocontroller vor dem Einbau programmiert werden!

## Programmieren des Mikrocontrollers

Zum Programmieren des Mikrocontrollers benötigt man ein entsprechendes PIC-Programmiergerät, das den Typ 16F873A unterstützt.

Die Firmware de Loof LocoO PIC 1.48 liegt im HEX-Format (als Text-Datei) vor. Üblicherweise kann ein PIC-Programmiersystem eine solche HEX-Datei einlesen. Was genau zu tun ist, um die Firmware in den Mikrocontroller zu bekommen, ist bitte der Anleitung des Programmiergerätes zu entnehmen.

## 7 Aufbau

Es wird vorausgesetzt, dass man im Aufbau elektronischer Schaltungen und vor allem im Umgang mit dem Lötkolben geübt ist. An Werkzeug wird für den Aufbau neben einem Lötkolben mit sehr feiner Spitze noch ein kleiner Seitenschneider benötigt, vorzugsweise ohne oder nur mit geringer Wate. Für den einen oder anderen Handgriff könnten noch eine kleine Flachzange und eine Pinzette hilfreich sein.

Überstehende Drähte werden mit einem kleinen Seitenschneider so kurz wie möglich abgeschnitten. Immer mit so wenig Lötzinn wie möglich und nur kurzzeitig festlöten.

Aus Erfahrung empfiehlt es sich, beim Bestücken mit den niedrigsten Bauteilen zu beginnen.

Es geht also mit den Widerständen los. Die Anschlussdrähte werden nahe am Gehäuse um 90° umgebogen und durch die Lötaugen gesteckt. Wer es besonders schön machen will, achtet darauf, dass die Farbringe alle gleich ausgerichtet sind. Von der anderen Seite her etwas festziehen, bis der Widerstand auf der Platine aufliegt. Wenn es nicht passt, laufen die Drähte nicht sauber durch das Loch.

Jetzt sind die IC-Fassungen an der Reihe. Beim Einlöten der Fassungen sollte man darauf achten, die Fassungen richtig herum einzubauen. K1 und T1 haben auf einer Seite eine Markierung in Form einer Kerbe oder eines Punktes am Gehäuse. Die Fassungen haben auch so eine Markierung, die sollte so zu liegen kommen wie im Bestückungsplan ersichtlich. Dann weiß man hinterher, wie herum der Baustein in die Fassung eingesetzt werden muss.

Es folgen die Kondensatoren. Hier muss man die Drähte voraussichtlich nicht biegen, da sie schon im passenden Abstand aus dem Bauteil herauskommen.

Bei dem Elektrolytkondensator bitte aufpassen, dass er richtig herum eingebaut wird, der Minus-Anschluss gehört in das rechteckige Lötauge. Auch hier muss man die Drähte wohl nicht mehr biegen, da sie schon richtig aus dem Bauteil herauskommen.

Das Quarzgehäuse besteht komplett aus Metall. Die Platine hat an der Stelle, an der die Anschlussdrähte des Quarzes durch die Platine gesteckt werden, auch Lötaugen auf der Bauteilseite, das ist dann direkt unter dem Quarzgehäuse. Deshalb muss man darauf achten, dass das Metallgehäuse keinen Kurzschluss zwischen den Anschlüssen verursacht. Die Platine hat zwar Lötstopplack, aber sicherer ist es, das Quarzgehäuse nicht vollständig auf die Platine zu drücken, sondern so einen halben Millimeter Abstand zu lassen. Als Abstandshalter kann man provisorisch ein Stückchen Pappe unterlegen, das nach dem Festlöten wieder entfernt wird.

Bei der Leuchtdiode ebenfalls bitte aufpassen, dass sie richtig herum eingebaut wird, die Katode (der Minus-Anschluss) gehört in das rechteckige Lötauge. Das ist in der Regel der kürzere Anschlussdraht, oder es gibt auf der Seite dieses Anschlussdrahtes am unteren Gehäuserand eine abgeflachte Stelle. Wer unsicher ist, testet die Anschlüsse mit einem Durchgangsprüfer aus: ist die Diode richtig angeschlossen, leuchtet sie, und dann ist der Minus-Anschluss an der Katode. Die Drähte nicht biegen, da sie schon richtig aus dem Bauteil herauskommen.

Als nächstes werden die Buchsen eingelötet. Wer keine einbauen und stattdessen Lötnägel nehmen will, darf diesen Absatz überspringen. Jede RJ 12-Buchse bitte vorsichtig in die Platine einsetzen, die beiden dicken Halteklemmen am Gehäuse könnten etwas schwer in die Löcher gehen. Wenn es gar nicht passt, muss man die Löcher etwas aufreiben. Beim Einsetzen der Buchse aufpassen, dass von den Anschlussdrähten keines umknickt. Beim Einlöten die Buchse leicht dagegenhalten, damit sie gut auf der Platine aufliegt.

Beim Montieren der Lötnägel bitte unbedingt eine Zange oder einen anderen Hilfsvorrichtung zum Halten des Lötnagels nehmen, sonst gibt es heiße Finger. Den Lötnagel gut gegenhalten, sonst liegt er nicht richtig auf der Platine auf. Die Spitze des Lötnagels gehört übrigens nach unten...

Wenn für X3, X4 und X5 PSS-Steckverbinder eingebaut werden, ist es sinnvoll, die Verriegelung nach innen zu setzen, damit die Stifte von außen zugänglich bleiben.

## 8 Inbetriebnahme

Nach dem Aufbau und einer frischen PIC-Programmierung muss die Schaltung erst initialisiert werden. Anschließend ist sie noch zu konfigurieren. Beides wird man wohl mit einem Computer machen, dazu muss man das Loconet über einen LocoBuffer mit dem Computer verbinden.

Für Windows PCs kann man zum Beispiel mit der de Loof-Software LocoHDL 3.5.2 arbeiten (die hier genannten Beispiele beziehen sich auf die englischsprachige Version, weil die deutsche Übersetzung an einigen Stellen sehr schnell zu Verständnisschwierigkeiten führen kann). Das Programm wird einmal für die Verbindung mit dem LocoBuffer eingestellt, dann steht das Loconet am LocoBuffer zur Verfügung. Bitte die

15 mA-Stromeinprägung nicht vergessen, sonst funktioniert das Loconet nicht. Wer ganz schlau war, der hat seinen LocoBuffer bereits mit einer solchen zuschaltbaren Stromquelle ausgerüstet.

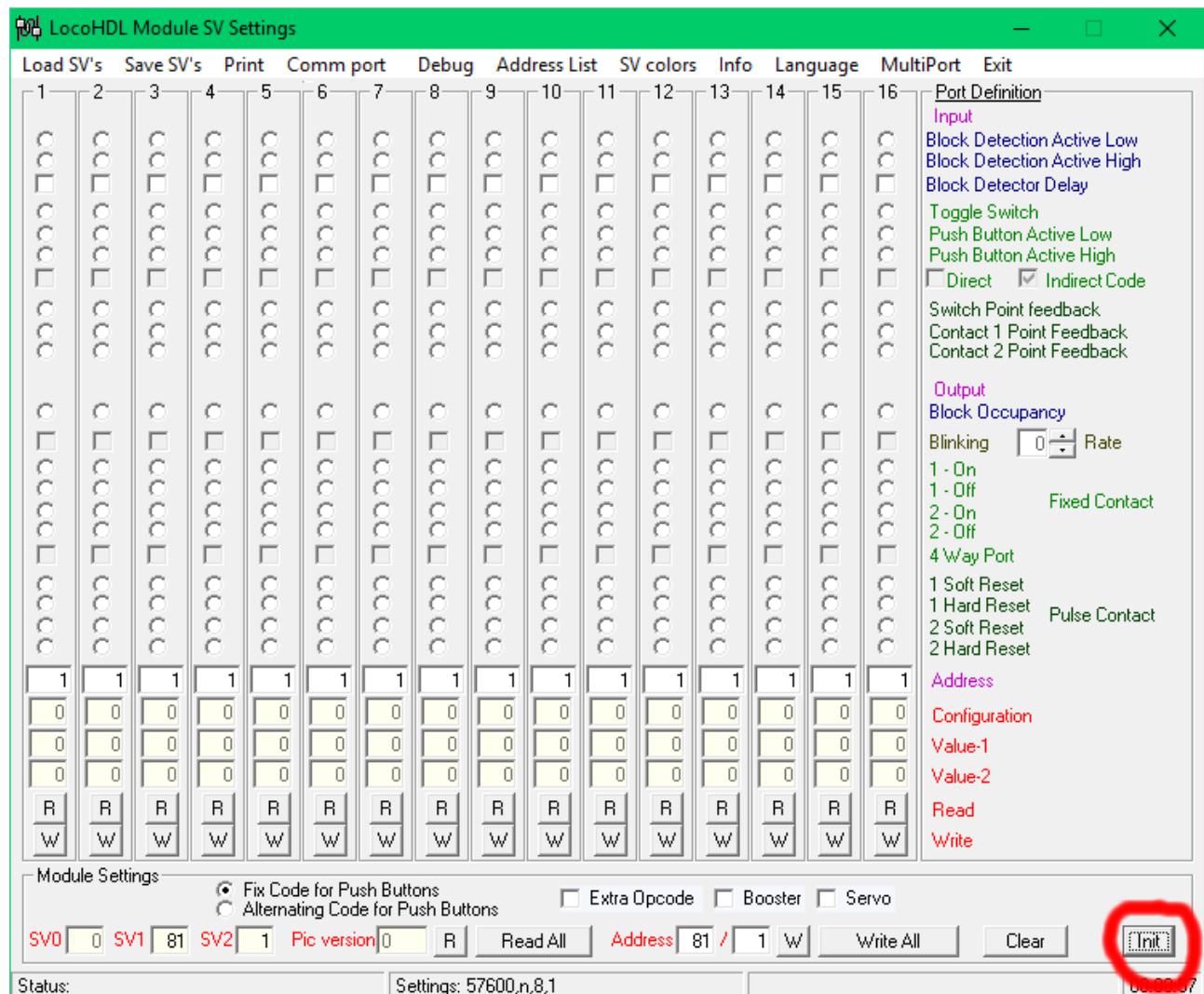
## Initialisierung

Jede wLocoIO-Platine muss nach der Programmierung des PIC erst einmal initialisiert werden. Durch die Initialisierung wird der Mikrocontroller mit allen für LocoIO notwendigen Daten versehen.

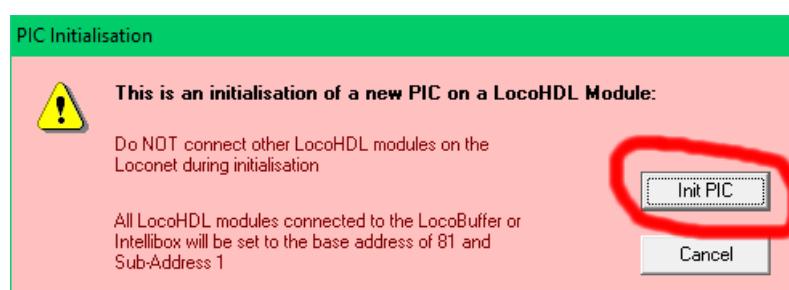
### Wichtig:

**Während Initialisierung darf immer nur eine einzige LocoIO-Schaltung am Loconet hängen!**

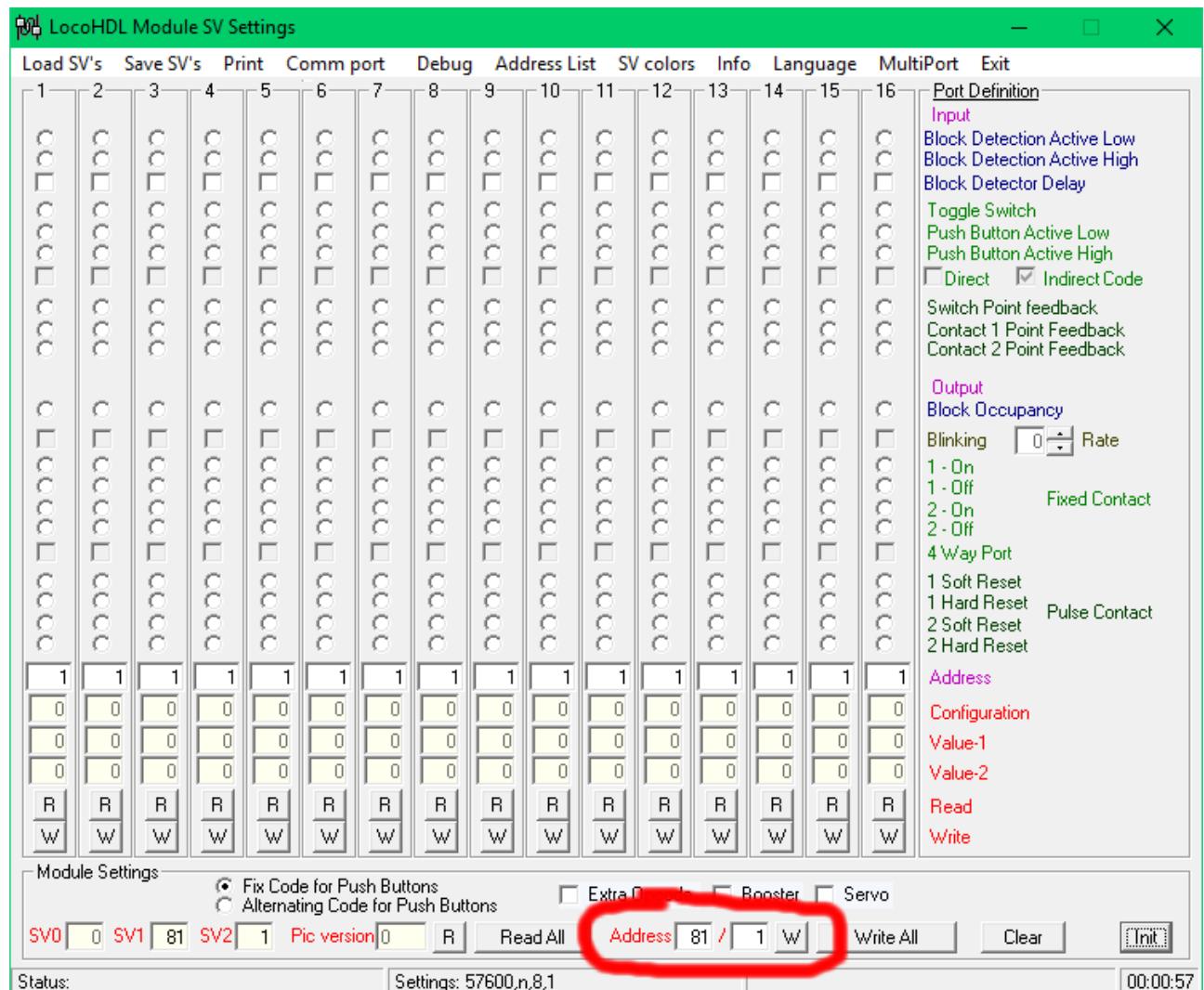
wLocoIO wird als einzige Schaltung am Loconet des LocoBuffers angeschlossen.



Zuerst drückt man bei LocoHDL auf die Schaltfläche „Init“.



Den anschließenden Sicherheitshinweis bestätigt man durch Drücken von „Init PIC“. Der PIC wird jetzt für die Arbeit als LocoIO-Baustein vorbereitet.



Jede LocoIO-Schaltung muss noch eine eindeutige Kennung haben. Durch die Initialisierung erhält die Schaltung die Kennung SV1=81 und SV2=1. Bei LocoHDL heißt diese Kennung Moduladresse. Die durch die Initialisierung vergebene Kennung sollte man sofort ändern, da durch die Initialisierung immer der gleiche Wert eingetragen wird. In die beiden Felder Address 81 / 1 werden also die gewünschten Werte eintragen. Der erste Wert kann 1..79 oder 81..127 sein (80 ist für den LocoBuffer reserviert). Der zweite Wert darf 1..126 sein. Es wird empfohlen, den ersten Wert als Gruppe zu verwenden, das heißt, der Wert ist bei allen LocoIO-Schaltungen gleich und mit den zweiten Wert bei jeder Schaltung durchzuzählen. Durch Drücken von W wird die neue Kennung in die Schaltung übertragen.

Bei den Rheinisch-Bergischen-Modulisten gibt es bereits eine Verteil-Ordnung für die Kennungen, falls LocoIO dort am allgemeinen Anlagen-Loconet angeschlossen werden soll. So wird verhindert, dass Adressen doppelt vorkommen, das wird nämlich nicht funktionieren. Auskunft dazu erteilt Michael Zimmermann.

Die Initialisierung ist abgeschlossen.

## Schaltungskonfiguration

Zur Konfiguration stehen verschiedene Möglichkeiten zur Verfügung. Auch hier kann man LocoHDL verwenden. Das Modellbahn-Steuerprogramm RocRail bietet ebenfalls einen Konfigurationsdialog für LocoIO

an. Und es gibt bestimmt noch weitere Möglichkeiten. Die folgende Beschreibung verwendet wieder LocoHDL.

Es wird vorausgesetzt, dass eine LocoIO-Schaltung auf die Kennung 40.1 und eine zweite auf die Kennung 40.2 initialisiert wurde.

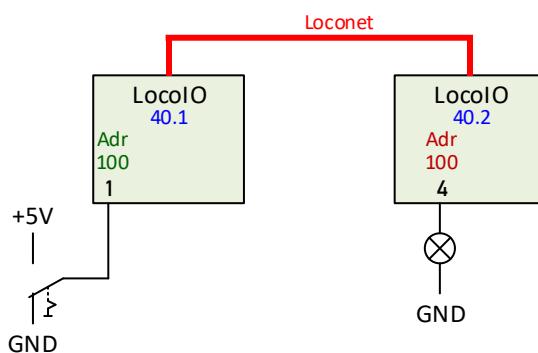
Beim Konfigurieren dürfen alle LocoIO am Loconet angeschlossen sein. Mit Address List bekommt man eine Liste der vorhandenen Schaltungen.

Die Kennung der Schaltung, die man konfigurieren will, muss in die Felder SV1 und SV2 eingetragen werden. Anschließend liest man die Daten der Schaltung durch Drücken auf R (oder Read All) ein.

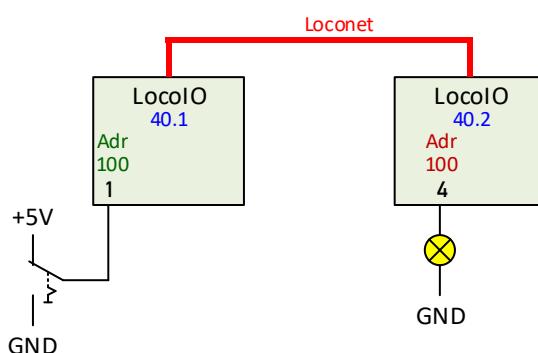
Jetzt kann man mit der Konfiguration der einzelnen Anschlüsse beginnen. Da das aber je nach Anwendungsfall sehr komplex sein kann, wird das in dieser Bauanleitung nicht ausführlich behandelt. de Loof hat in seinen Handbüchern zu LocoHDL und LocoIO etliche Informationen zusammengetragen, die kann man auf seiner Internetseite nachlesen. Weiterhin hat auch Michael Zimmermann darüber hinaus gehende Informationen zusammengestellt.

Konfigurationsbeispiel für einen Schalter, der eine fernbedient eine Lampe ein- und ausschaltet:

Für die Lampe sollte man eine LED mit Vorwiderstand nehmen, denn ein Ausgang darf maximal nur bis 25 mA belastet werden!



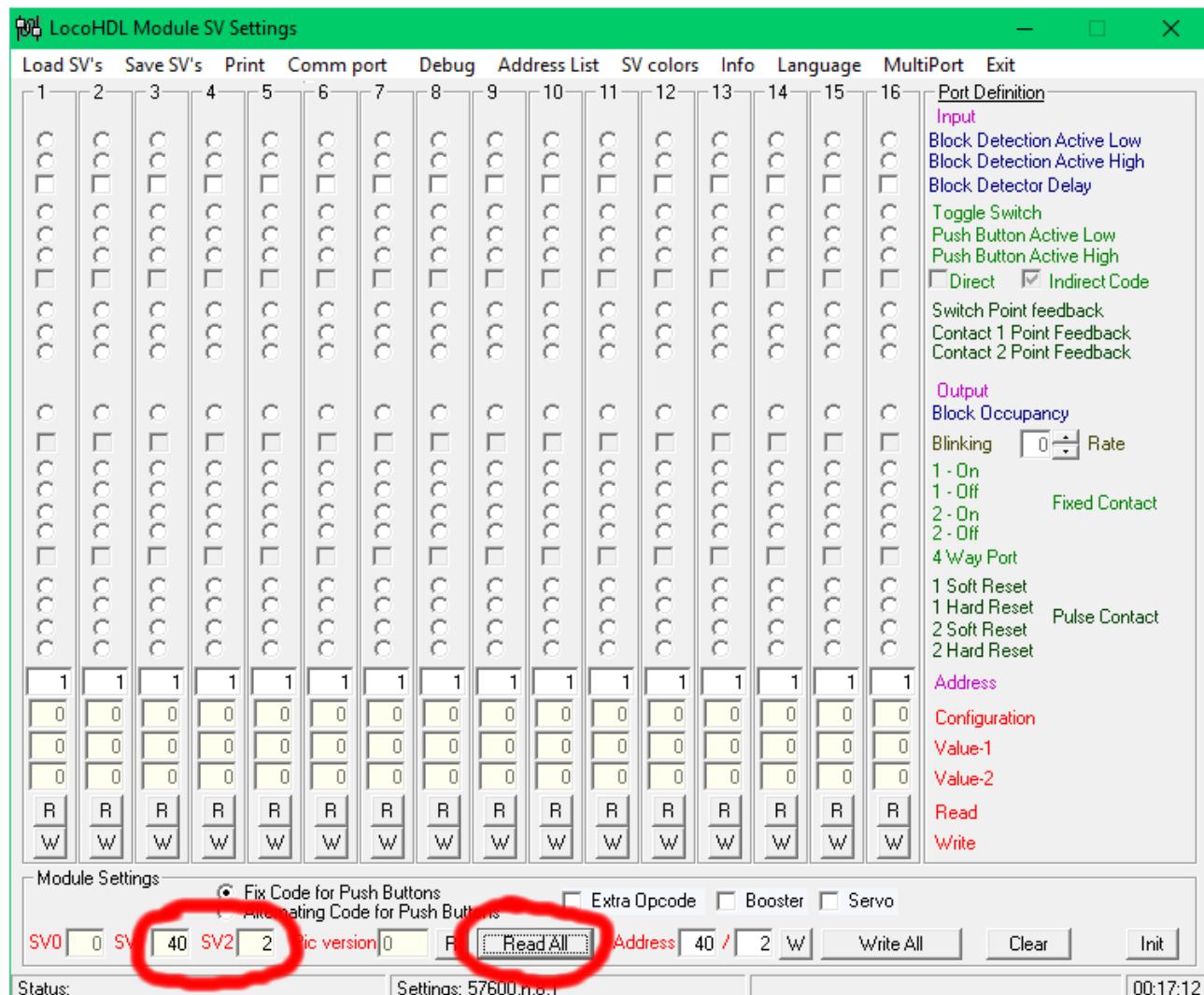
Der Anschluss 1 von LocoIO 40.1 soll als Eingang konfiguriert werden, der den Zustand des angeschlossenen Schalters an die Adresse 100 weitergibt. Ist also der Eingang wie dargestellt logisch 0, ist auch der Ausgang logisch 0, und die Lampe leuchtet nicht.



Wird der Schalter umgelegt, und der Eingang wird logisch 1, geht auch der Ausgang auf logisch 1, und die Lampe leuchtet.

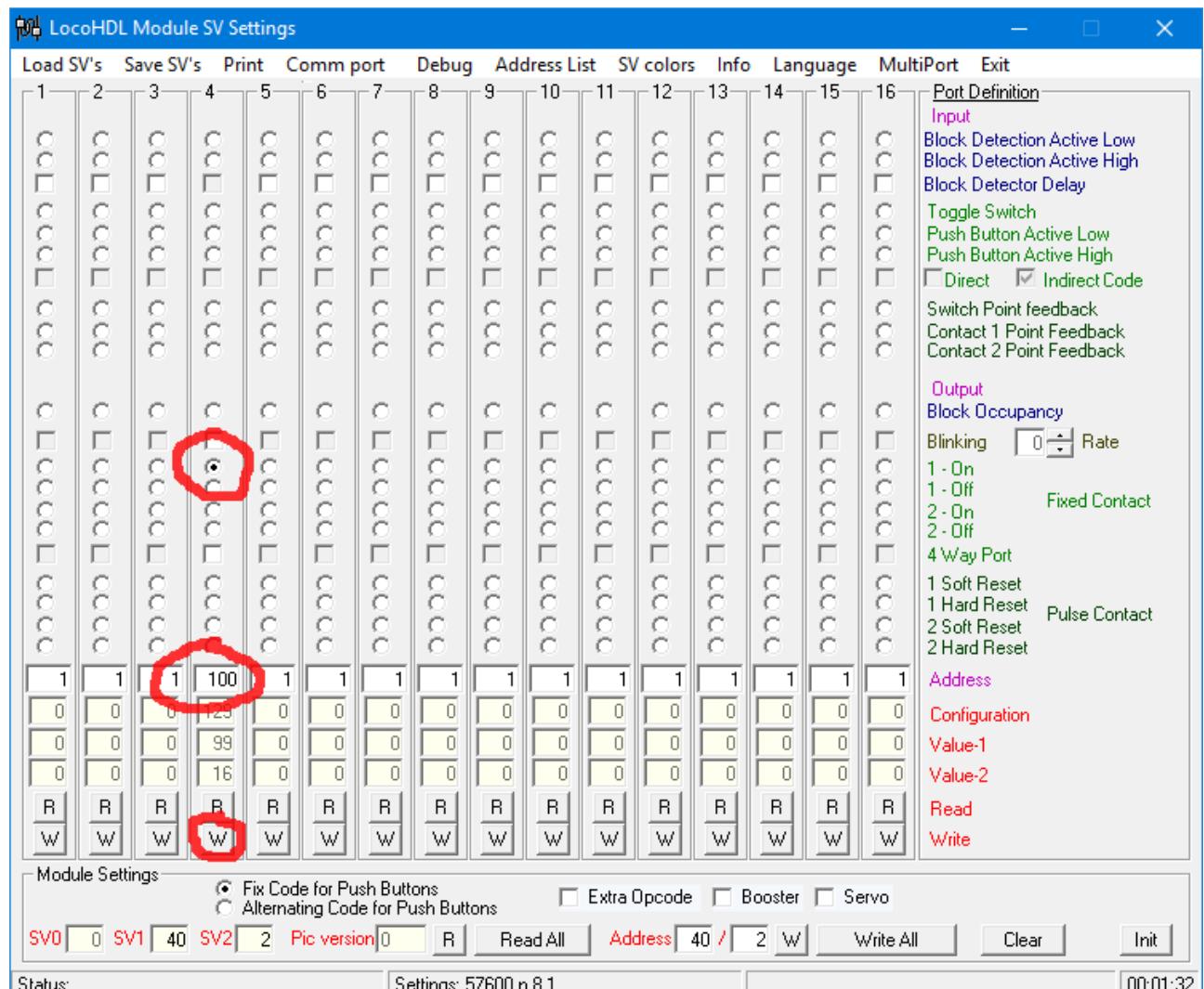
Schalter und Lampe werden mit den LocoIO-Anschlüssen verbunden.

Der Ausgang wird zuerst konfiguriert. Die Reihenfolge ist prinzipiell egal, aber hier wird es jetzt so gemacht.

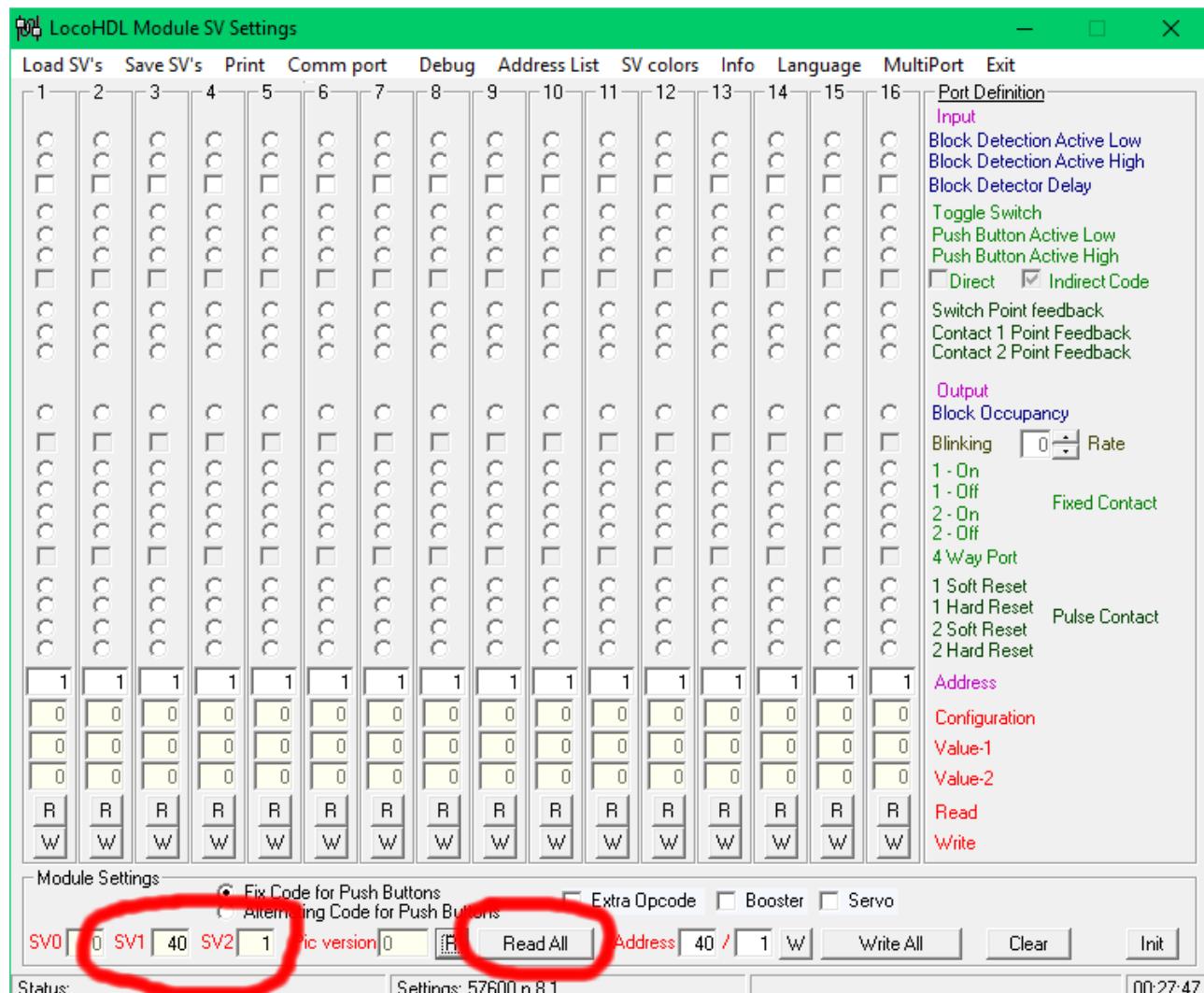


Zuerst muss bei SV1 der Wert 40 und bei SV2 der Wert 2 eingetragen werden. Danach auf die Schaltfläche Read All drücken. Kurz darauf wird die aktuelle Konfiguration angezeigt.

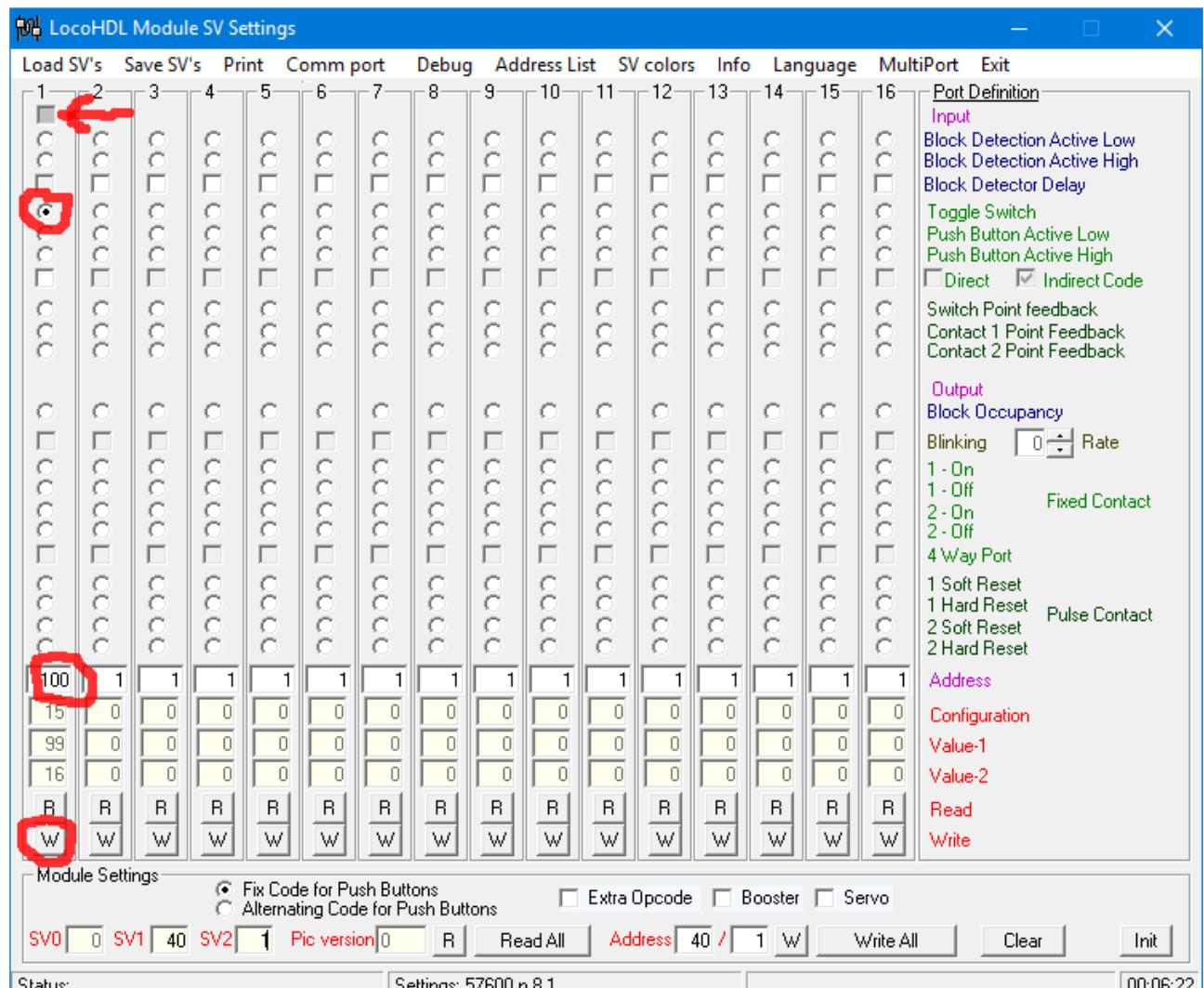
Jetzt kommt der Anschluss an die Reihe. Für jeden Anschluss gibt es im Programmfenster eine Spalte. Die Nummer gehört zu dem Anschluss mit der gleichen Nummer auf der Platine. Der Anschluss 4 soll ein direkt schaltender Ausgang werden, der über die Adresse 100 angesprochen wird.



In der Spalte 4 wird der Anschluss als Ausgang durch Klicken auf 1-On eingestellt, der Punkt erhält dabei eine Markierung. Im Feld Address wird der Wert 100 eingetragen. Dann auf die Schaltfläche W (für Write) in dieser Spalte drücken. Nach kurzer Zeit wird die Übernahme der neuen Einstellung durch gelbe Felder quittiert. Anschließend kann man mit erneutem Lesen durch Drücken von R (für Read) das Setzen der Einstellung überprüfen, die Felder werden dann grün. Jetzt kann der Ausgang mit Hilfe der On/Off-Schaltfläche bereits getestet werden. Dieser Schalter wird erst nach dem erneuten Lesen sichtbar.



Jetzt wird der Eingang konfiguriert. Zuerst muss bei SV1 der Wert 40 und bei SV2 der Wert 1 eingetragen werden. Auf Read All drücken. Nach kurzer Zeit wird die aktuelle Konfiguration angezeigt.



Dann in der Spalte 1 für den Anschluss 1 auf Toggle Switch drücken. Dort erscheinen dann ein Punkt und ganz oben der Status-Melder des Eingangs. Im Feld Address den Wert 100 eintragen. Dann auf die Schaltfläche W in dieser Spalte drücken. Nach kurzer Zeit wird die Übernahme der neuen Einstellung durch gelbe Felder quittiert. Anschließend kann man mit erneutem Lesen durch Drücken von R (für Read) das Setzen der Einstellung überprüfen, die Felder werden dann grün. Der Statusmelder zeigt den Schalterzustand des Eingangs an, das Feld sollte schwarz sein, wenn der Schalter auf GND liegt, und gelb, wenn er auf +5V liegt.

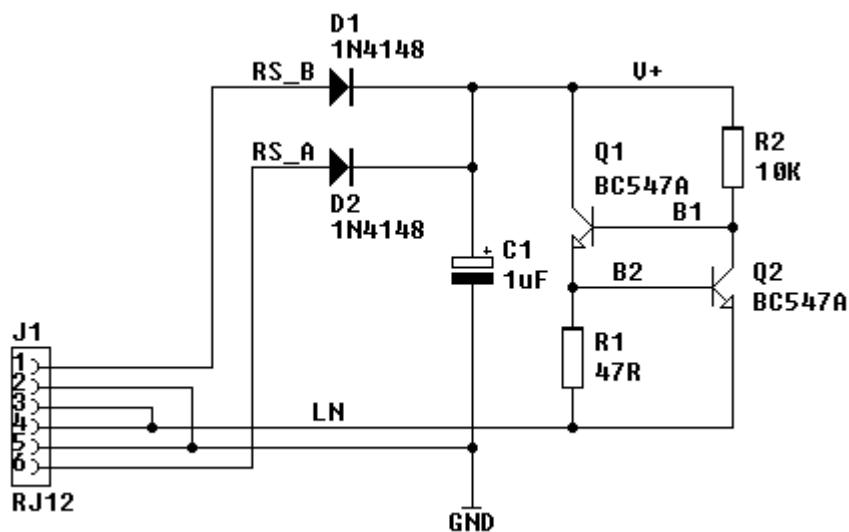
Die Beispielkonfiguration ist damit abgeschlossen.

## Normaler Betrieb

Die Schaltung ist nach Anschluss von Energieversorgung und Loconet sofort betriebsbereit. Die Schaltung ist wartungsfrei.

Hier noch einmal der Hinweis, dass ein einzelner Ausgang maximal 25 mA Laststrom ziehen darf. Für eine LED reicht das aus (noch besser: immer 2 mA-LEDs nehmen). Insgesamt dürfen alle Ausgänge einer Gruppe (also 1..8 oder 9..16) zusammen aber nur maximal 200 mA Laststrom ziehen. Und alle Ausgänge zusammen dürfen 250 mA nicht überschreiten! Wenn die angeschlossenen Verbraucher mehr Strom benötigen, muss man Verstärker verwenden.

Bei Betrieb an einer lokalen Loconet-Leitung darf man nicht die 15 mA-Stromeinprägung vergessen, sonst funktioniert das Loconet nicht.



Das Bild wurde dem entsprechenden Artikel von DCC Müller entnommen. Michael Zimmermann hat für eine solche Stromquelle auch eine Platine gemacht.

## 9 Quellen

Bei allen Internet-Verknüpfungen ist immer nur die Hauptseite angegeben, weil die eigentlichen Ziele manchmal umziehen, das kann man in einem Dokument wie diesem nicht ständig nachpflegen. Gegebenenfalls muss man also ein wenig navigieren.

### **John Jabour**

LocoIO

<http://www.locobuffer.com>

### **Hans de Loof**

Loconet als Modellbahn-Netzwerk

<http://users.telenet.be/delooft>

### **Modelbouw HET SPOOR**

HDL-Platinen und -Bausätze

<http://www.spoorshop.com>

### **Michael Zimmermann**

Krümelbahn

<http://www.kruemelsoft.privat.t-online.de>

### **Digitrax, Inc.**

Loconet Personal Edition

<http://www.digitrax.com>

### **Reichelt Elektronik**

Bauteil-Lieferant

<http://www.reichelt.de>

### **Microchip**

Datenblatt PIC 16 F 873 A

<http://www.microchip.com>

### **DCC Müller**

15 mA-Stromquelle für das Loconet

[http://www.dcc-mueller.de/loconet/lpull\\_d.htm](http://www.dcc-mueller.de/loconet/lpull_d.htm)

## **10 Versionierung des Dokuments**

vor 1.6	nicht nachgehalten
1.6 30.5.2019	redaktionelle Überarbeitung und Korrektur der Inbetriebnahme

Dipl.-Ing. Michael Zimmermann  
 Buchenstr. 15  
 42699 Solingen  
 ☎ 0212 46267  
 <https://kruemelsoft.hier-im-netz.de>  
 [BwMichelstadt@t-online.de](mailto: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
Grundlagen zu LocoIO .....	4
Konfiguration des LocoBuffer in RocRail .....	4
Hinweis zum USB-LocoBuffer und der Software LocoHDL .....	4
Initialisierung der LocoIO-Module .....	5
Adresse für die LocoIO-Module einstellen .....	6
Deloof'sche Begriffe – der Versuch einer Erklärung .....	10
Es geht los: LocoIO anwählen.....	11
Vertrauen ist gut –Kontrolle ist besser .....	12
Einen Ausgang mit einem Eingang steuern - Umschalter.....	13
Steuerung mit RocRail.....	14
Einen Ausgang mit einem Eingang steuern - Blockbelegtmeldung .....	16
Anzeige in RocRail.....	17
Steuerung mit RocRail .....	18
Wechselnde Kode für Druckknopf .....	19
Einen oder zwei Ausgänge mit zwei Eingängen steuern .....	21
Verwendung als Impulsausgang (anstelle eines Dauerkontaktes) .....	22
Steuerung mit RocRail .....	23
Steuerung mit dem TwinCenter / der Intellibox .....	25
Vier Eingänge – vier Ausgänge .....	26
Geht doch – oder? .....	27
Konfigurieren mit JMRI .....	29
Zu guter Letzt ... speichern der Einstellungen .....	29
Eine Dokumentationshilfe – die Adressstabelle .....	29
Linkliste .....	30

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

Diese Anmerkungen zu *Inbetriebnahme und Betrieb* beziehen sich auf die LocoIO von Hans Deloof (<https://locohdl.synology.me/pageDE8.html>) und sollen dem Anwender helfen, die bereits bestehende Dokumentation besser zu verstehen, sie ersetzt diese keinesfalls!

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 und wie er sich zu verhalten hat.**

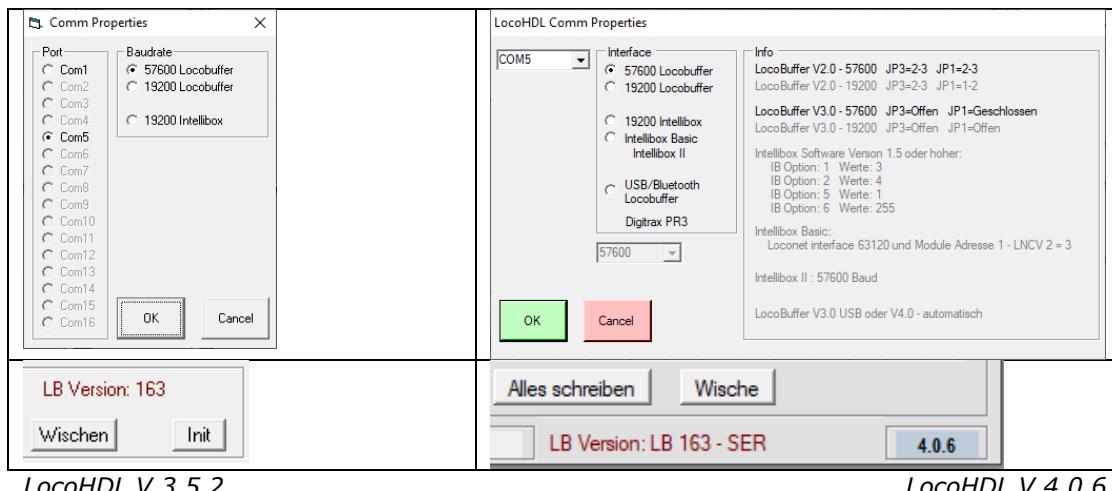
Ein LocoIO lässt sich auf zwei Arten konfigurieren:

- Entweder direkt mit der Software LocoHDL

<https://locohdl.synology.me/LocoHDL/LocoHDL.zip> (aktuell: V4.0.6.27),  
<https://locohdl.synology.me/LocoHDL/LocoHDL%20Configuration%20DE.pdf>.

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

Bei mir kommt der LocoBuffer von Hans Deloof in der Hardware Version V2.3 mit der Software Version **1.63** zum Einsatz<sup>1</sup> – mit einem USB-Adapter zum Anschluss an die RS232-Schnittstelle des LocoBuffers:



- 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**)<sup>2</sup> auch so

<sup>1</sup> Andere LocoBuffer-Versionen wurden nicht getestet; je nach LocoBuffer-Bauart besitzen diese auch zwei PIC-Prozessoren

<sup>2</sup> Höchste PIC-Version für LocoHDL **3.5.2** ist PIC-Version **1.48**

zusammen zu verwenden. Die Screenshots aus RocRail wurden mit der RocRail-Version **13775** erstellt.

Und noch etwas:

- 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.
- 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...
- Wenn in dieser Ergänzung von Modulen bzw. Modulisten die Rede ist, so bezieht sich das darauf, dass ich meine Module zusammen mit anderen Modellbahnhern in einer Modulanlage betreibe...
- RocRail unterstützt die HDL-Module 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.  
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!

## Grundlagen zu LocoIO

Weitere Grundlagen zum LocoIO sind in der Anleitung zum wLocoIO-2 (enthalten in: der *LocoIO (Zusammenfassung).pdf* [https://github.com/Kruemelbahn/LocoIO-Editor/blob/main/Documentation/LocoIO%20\(Zusammenfassung\).pdf](https://github.com/Kruemelbahn/LocoIO-Editor/blob/main/Documentation/LocoIO%20(Zusammenfassung).pdf)) in den Kapitel 2, 3 und 8 beschrieben.

## Konfiguration des LocoBuffer in RocRail

Die Konfiguration des LocoBuffer in RocRail habe ich bereits im Dokument *OpenDCC – Zusammenfassung zum Bau.pdf* (<https://magentacloud.de/s/-RfKmC8bpINXcJeN>) beschrieben – bitte dort nachlesen.

## Hinweis zum USB-LocoBuffer und der Software LocoHDL

Die Software 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...)

## Initialisierung der LocoIO-Module

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.

1. den LocoBuffer

- ❖ an 12V-Spannungsversorgung anschließen
- ❖ und mit dem PC verbinden

2. das Programm LocoHDL starten

- ❖ prüfen, ob der LocoBuffer von der Software erkannt wurde, d.h.:
  - es gibt keine Fehlermeldung beim Starten von LocoHDL
  - unten rechts im Fenster wird die Softwareversion des LocoBuffer angezeigt

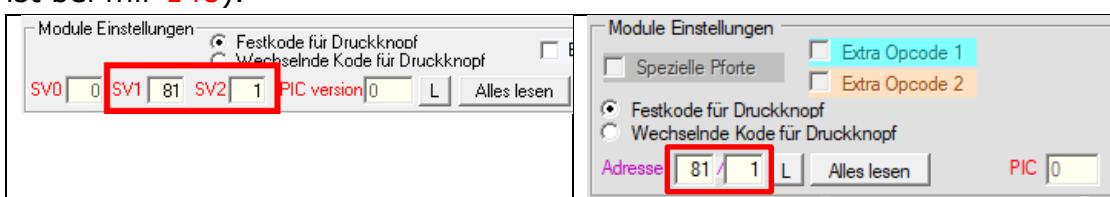


Solange der LocoBuffer von der Software LocoHDL nicht erkannt wird, kann ein LocoIO weder initialisiert noch konfiguriert werden und wir brauchen hier nicht weiterzumachen.  
→ ggf. PC neustarten und zurück zu Schritt 1!

3. jetzt das zu initialisierende LocoIO

- ❖ an – möglichst eine andere<sup>3</sup> – 12V-Spannungsversorgung anschließen
- ❖ über ein (kurzes) LocoNET®-Kabel (RJ12 an RJ12) das LocoIO mit dem LocoBuffer verbinden

- Button „Init“ betätigen, 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; ist bei mir **148**).



LocoHDL V 3.5.2

LocoHDL V 4.0.6

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

---

<sup>3</sup> 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. Die Software LocoHDL reagiert recht empfindlich darauf, wenn der USB-Port kurzzeitig nicht verfügbar ist, während das Programm läuft.

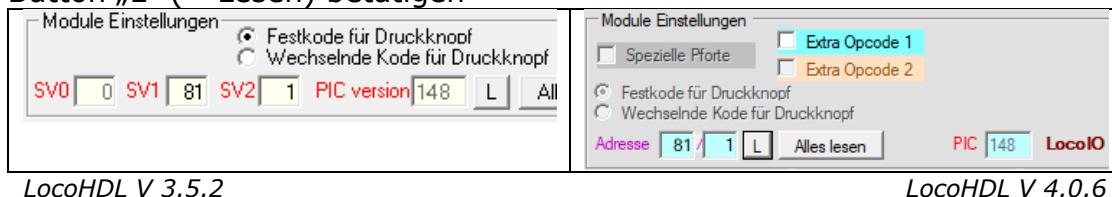
## Adresse für die LocoIO-Module einstellen

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

- (meine) Basismoduladresse ist immer **60**  
Jeder Modulist 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**

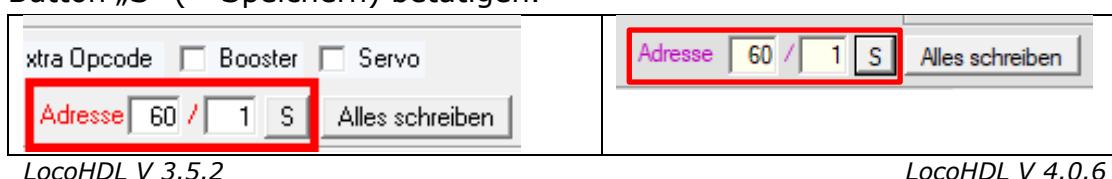
Bevor überhaupt Werte in das LocoIO geschrieben werden können, muss der zu konfigurierende LocoIO der Software bekannt gemacht werden – auch wenn nur ein einziger LocoIO angeschlossen ist:

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

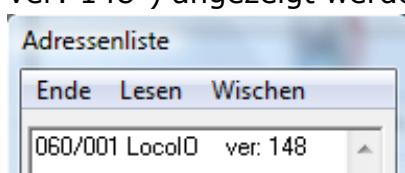


Wenn der LocoIO erkannt wurde, wird die **PIC version** angezeigt → jetzt kann konfiguriert werden

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



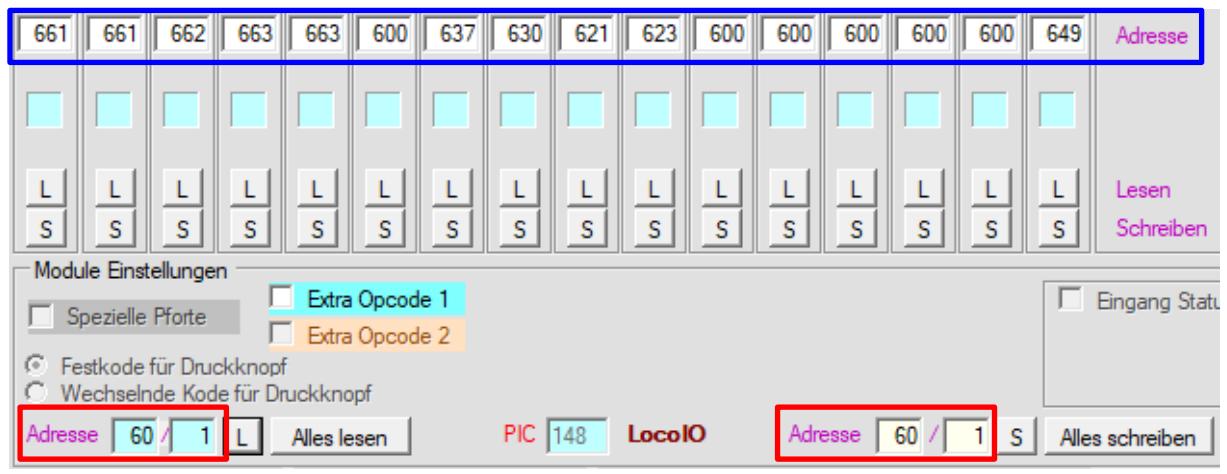
- Eine Kontrolle kann z.B. mit „**Adressenliste**“ – „**Lesen**“ erfolgen, hier sollte dann als Ausgabe die soeben eingestellte Adresse (z.B. „**060/001** LocoIO ver: 148“) angezeigt werden:



### Anmerkungen zum Begriff „Adresse“:



LocoHDL V 3.5.2



LocoHDL V 4.0.6

- Die LocoIO-Adresse (Moduladresse, **rote** Umrahmungen)
    - hat nichts mit der Sensor<sup>4</sup>- oder Aktor<sup>5</sup>adresse (**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.
- An einer (Modul)Anlage ist es sinnvoll, die Moduladressen entsprechend vorhandener Anlagenteile und Modulbesitzer strukturiert zu vergeben.

<sup>4</sup> Sensor = Schalter, Taster, Relaiskontakt, Rückmeldung einer Lichtschranke usw., allgemein = Eingang

<sup>5</sup> 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:

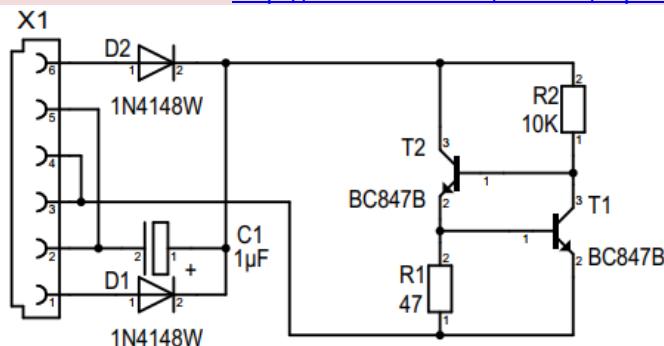
- LocoIO-Basismoduladressen 60...69
- Ausgänge von 600...699 und 1600...1699
- Eingänge von 600...699 und 1600...1699 bzw. von 1199...1398 und 3199...3398



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



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/lnpull\\_d.htm](http://dcc-mueller.de/loconet/lnpull_d.htm)):



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.

- Nicht angeschlossene Eingänge erzeugen ggf. unnötige Telegramme, die das LocoNET® belasten bzw. sogar blockieren!  
Daher gilt für nicht verwendete Anschlüsse:

**entweder**

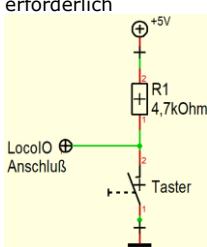
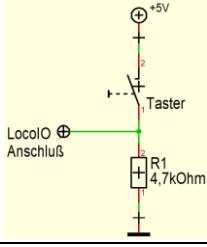
- als Ausgänge konfigurieren. Dabei verwende ich als Ausgangsadresse hierbei immer den Wert 600.

**oder**

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

## Deloof'sche Begriffe – der Versuch einer Erklärung

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

Begriff	Erklärung
Pforte	Port
Pinne	Pin Anschluss, entweder Eingang oder Ausgang benötigt immer eine Adresse, die bei einem: - Eingang die „Sensor“-Adresse - Ausgang eine „Aktor“-Adresse ist.
Aktiv Lage	Active low <b>Eingang schaltet nach GND (Masse)</b> meist ist zusätzlich ein externer Pull-Up-Widerstand z.B. 4,7kOhm erforderlich
	 Der Taster kann auch ein Relaiskontakt o.ä. sein...
Aktiv Hohe	Active high <b>Eingang schaltet nach +5V</b>
	 Der Taster kann auch ein Relaiskontakt o.ä. sein...
Verspätung	Delay Verzögerung, verzögert
Druckknopf	Push button Taster
Erdungskontakt	
Festkontakt	Fix contact Dauersignal (im Gegensatz dazu: Pulskontakt =  oder  Stand: 21.09.2024

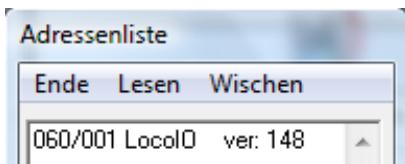
## Es geht los: LocoIO anwählen

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:



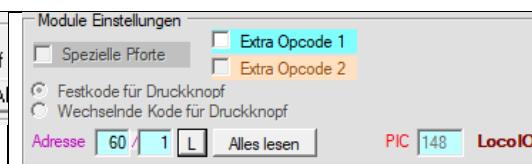
Bei leerer Liste ist dann eine ausgiebige Fehlersuche angesagt...

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

- Aktuelle 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 dem Button „L“) eintragen
- Button „**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: Button „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	grey	Status unbekannt
schwarz	black	Eingang nicht betätigt
gelb	yellow	Block Kontakt betätigt
grün	green	Druckknopf niedrige Adresse betätigt
rot	red	Druckknopf hohe Adresse betätigt

Ausgänge: können testweise über den Button „Ein“ bzw. „Aus“ geschaltet werden.

## Einen Ausgang mit einem Eingang steuern - Umschalter

(Umschalter und Festkontaktausgang)

(Telegramm: OPC\_SW\_REQ(B0), Adressbereich: 1...4096)

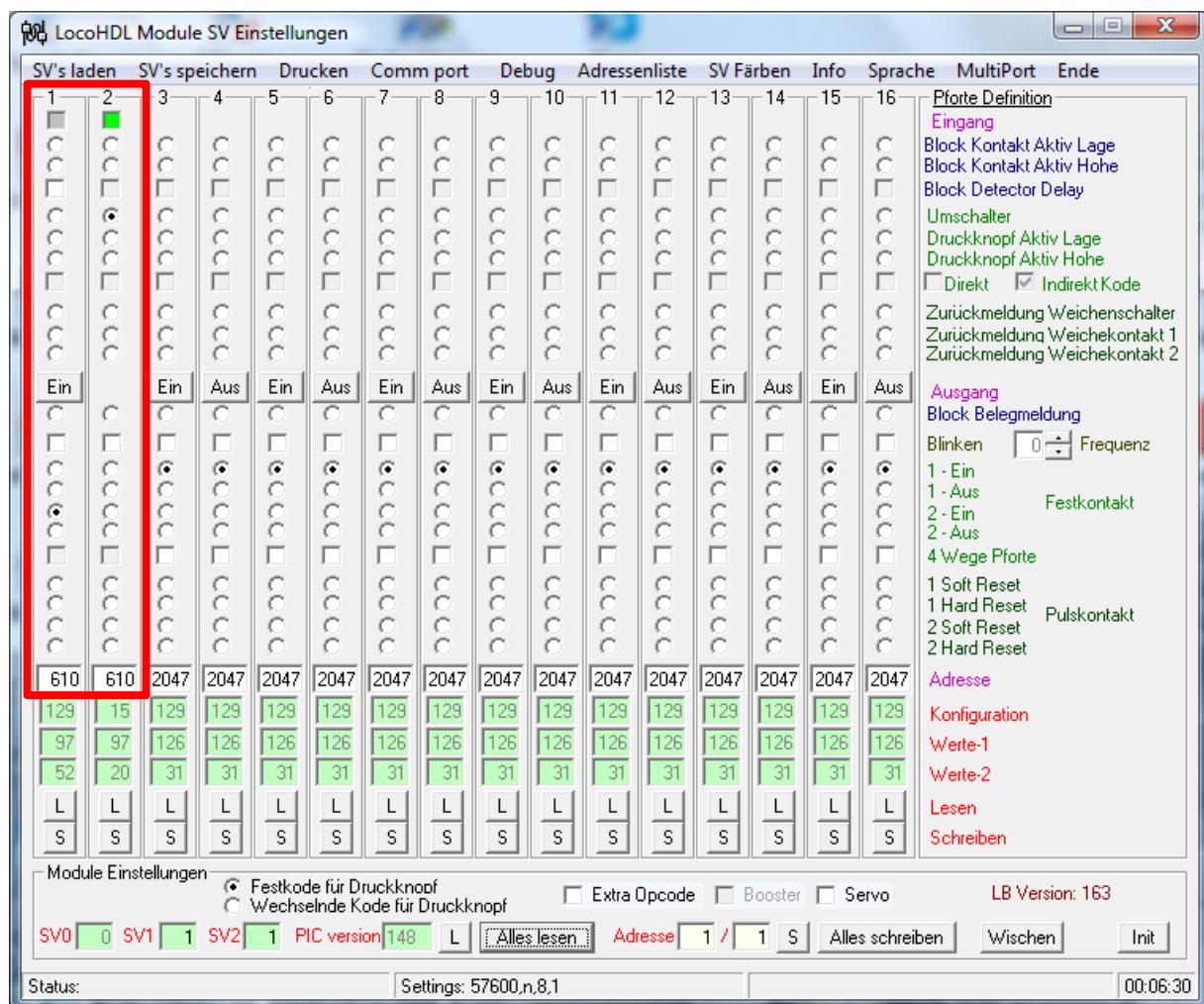
Einen Ausgang, der einem Eingang folgt, kann mit der Funktion Umschalter realisiert werden.

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

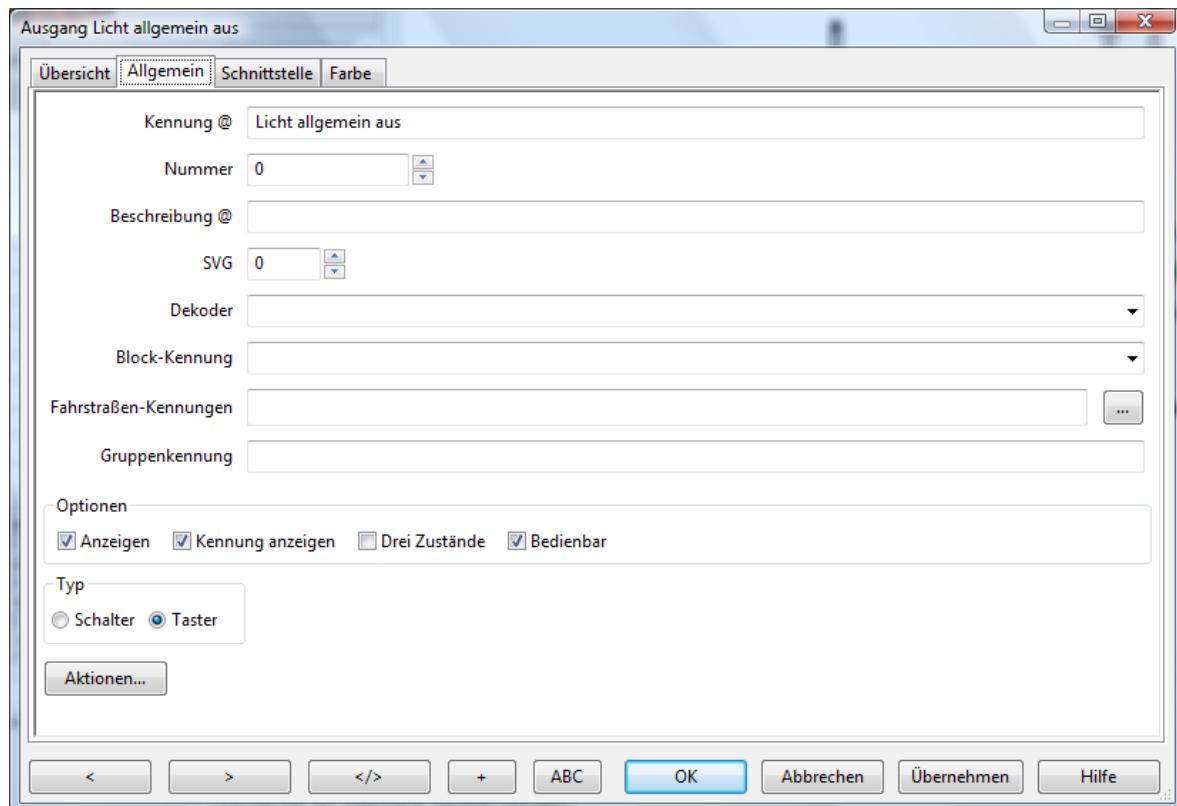
Anmerkungen:

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

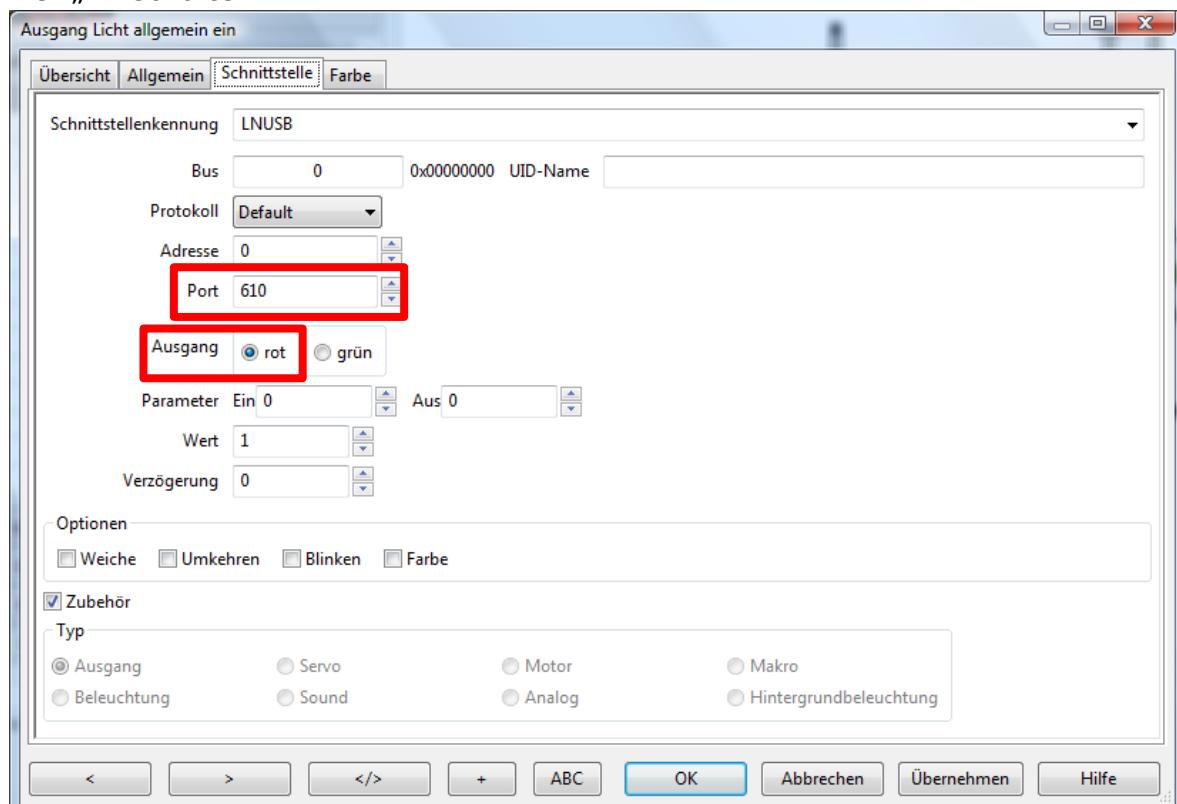
## Steuerung mit RocRail

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

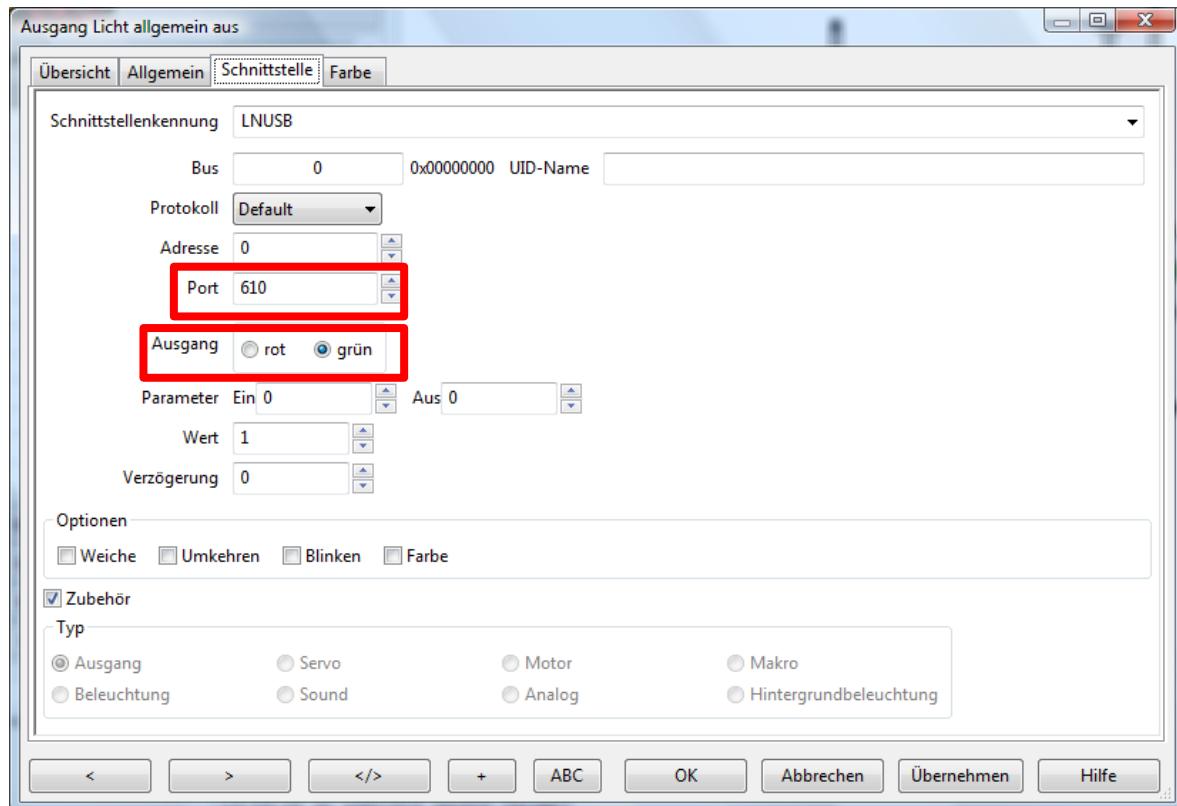
Gemeinsam für beide Elemente:



## 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)

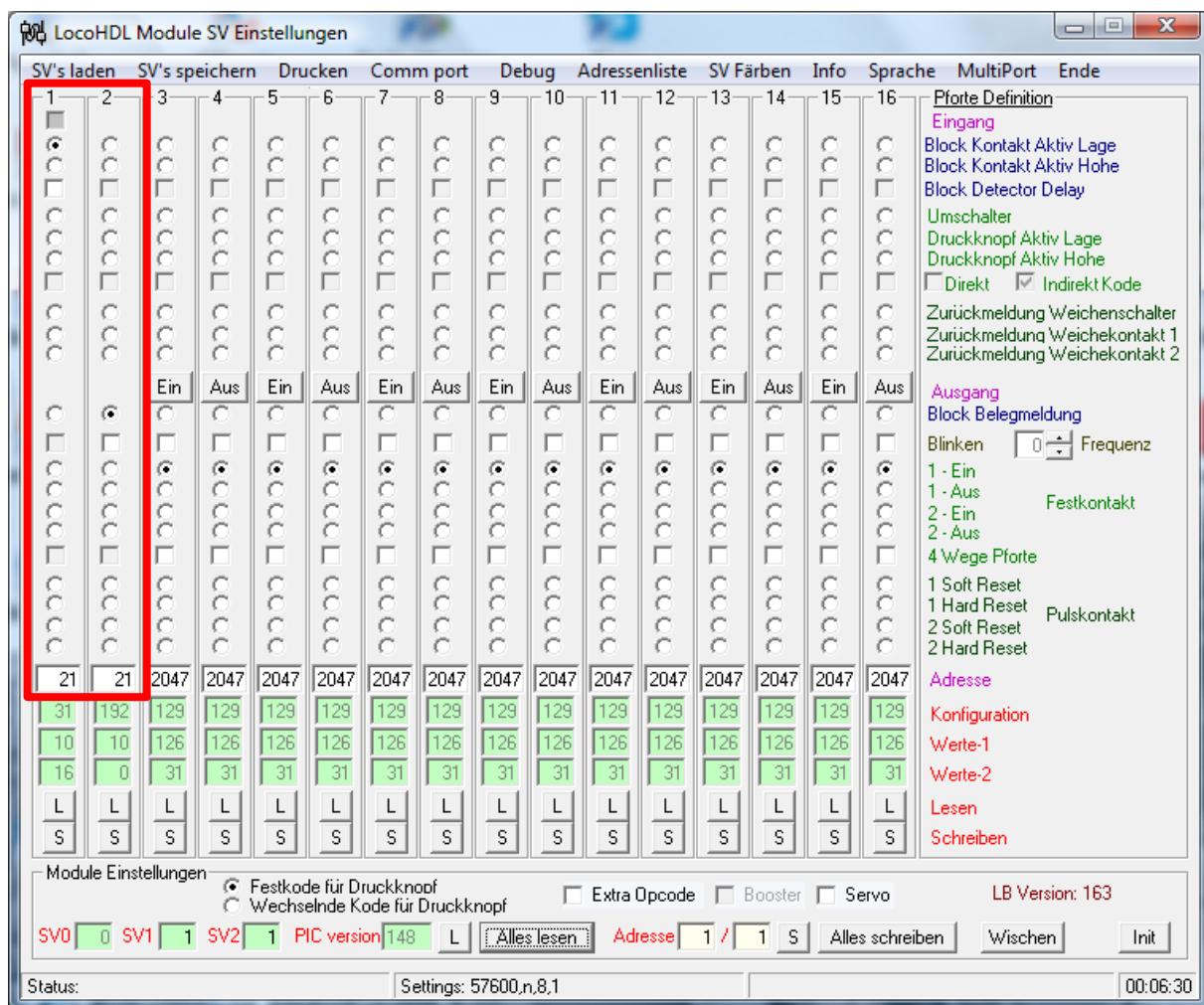
(Telegramm: OPC\_INPUT REP(B2), Adressbereich: 1...2048)

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

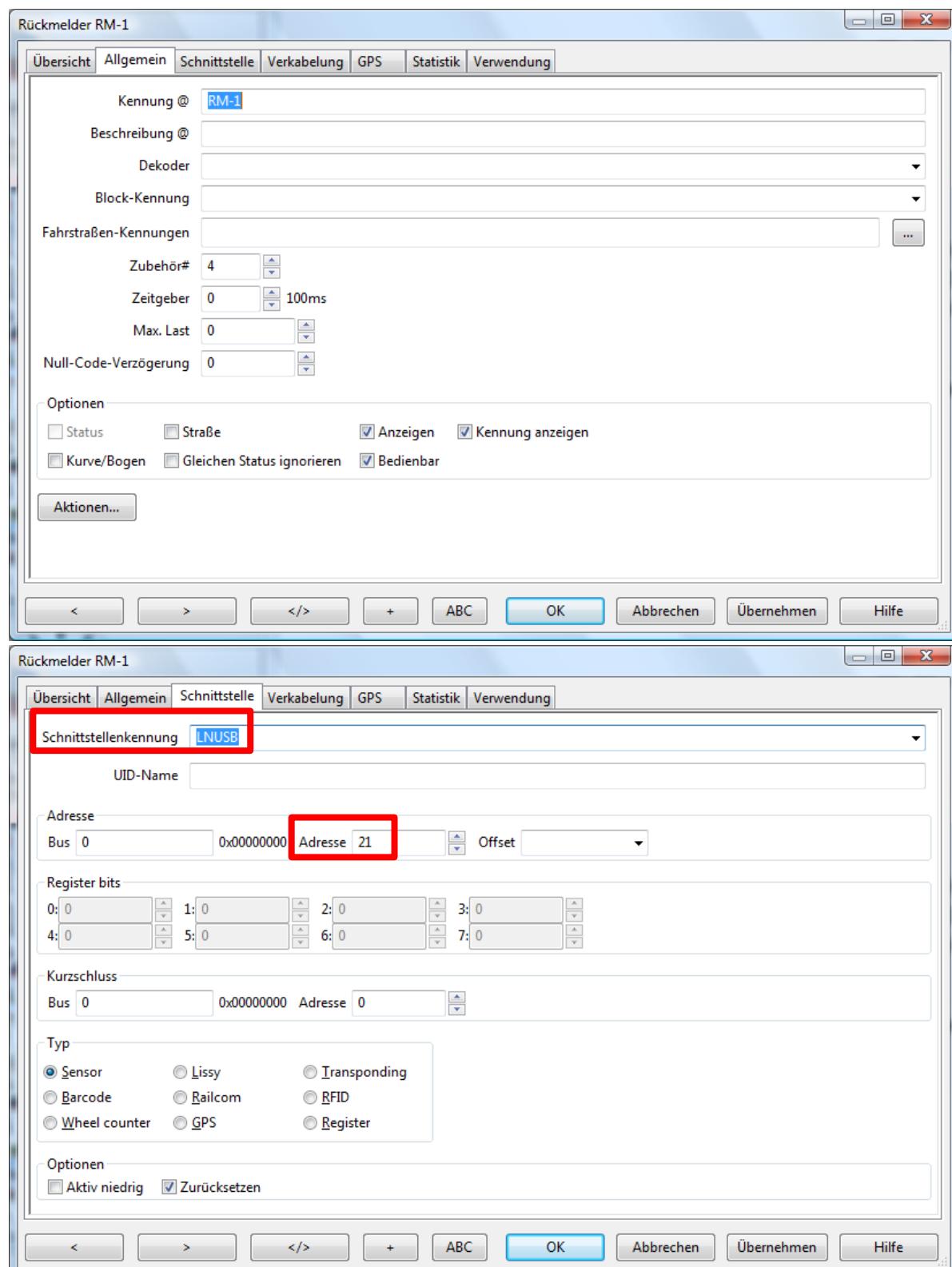
### Anmerkungen:

- 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 Rückmelders (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!

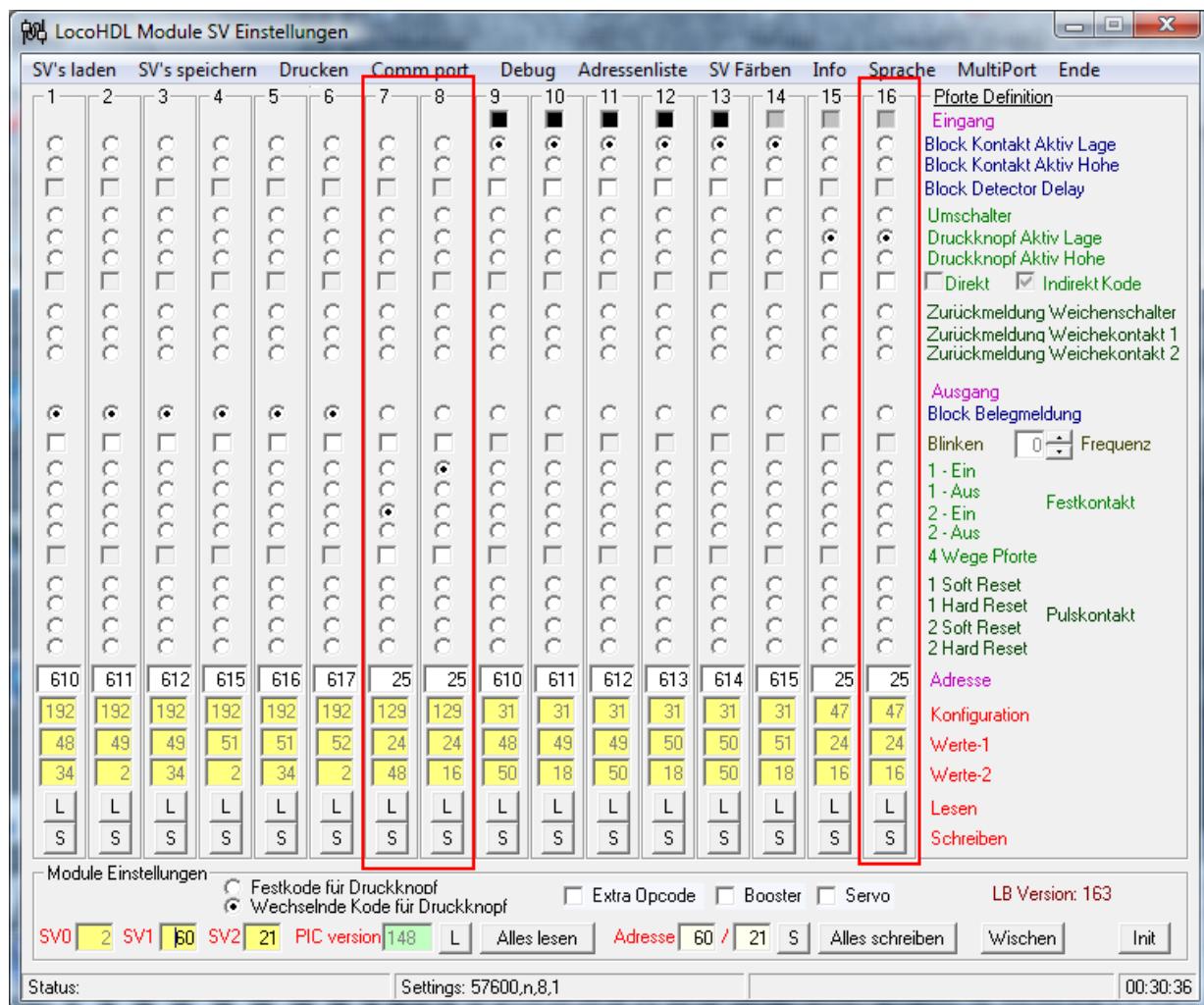
## Wechselnde Kode für Druckknopf

(Druckknopf Aktiv Lage und Druckknopf Aktiv Hohe)

(Telegramm: OPC\_SW\_REQ(B0), Adressbereich: 1...4096)

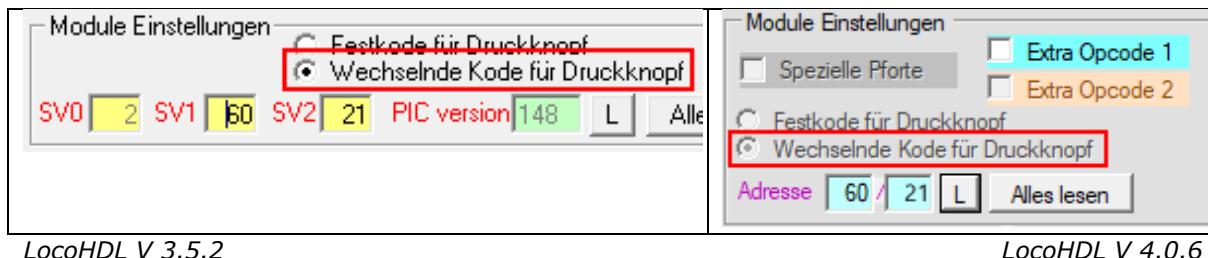
Hierbei wird ein Ausgang von einem Eingang gesteuert. Dieser Eingang schaltet bei der ersten Betätigung den Ausgang ein und bei der nächsten Betätigung wieder aus: der Ausgang wechselt somit mit jeder Tasterbetätigung am Eingang seinen Zustand Ein → Aus → Ein → Aus → ....

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!

## Einen oder zwei Ausgänge mit zwei Eingängen steuern

(Druckknopfeingang und Festkontaktausgang)

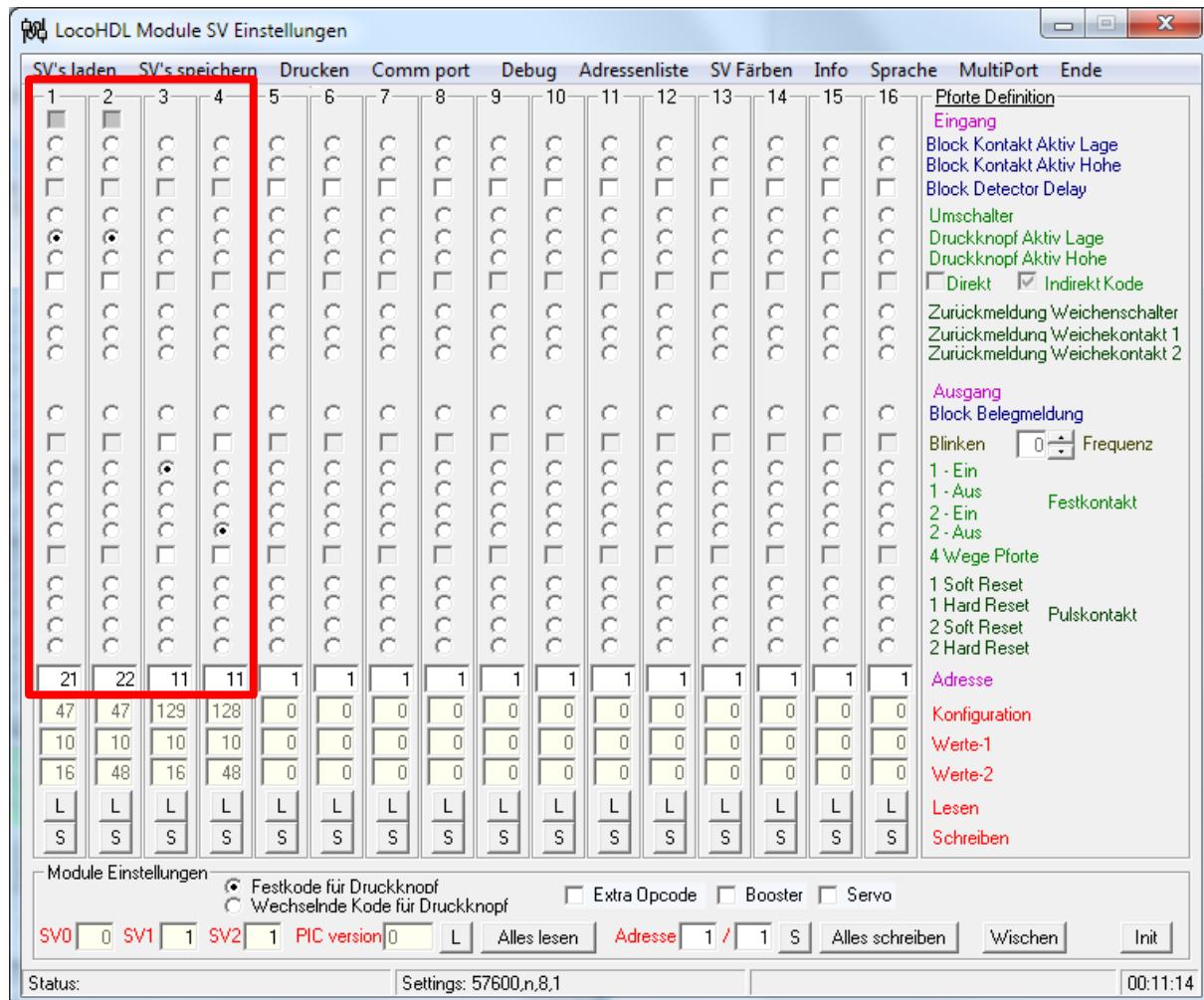
(Telegramm: OPC\_SW\_REQ(B0), Adressbereich: 1...4096)

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:

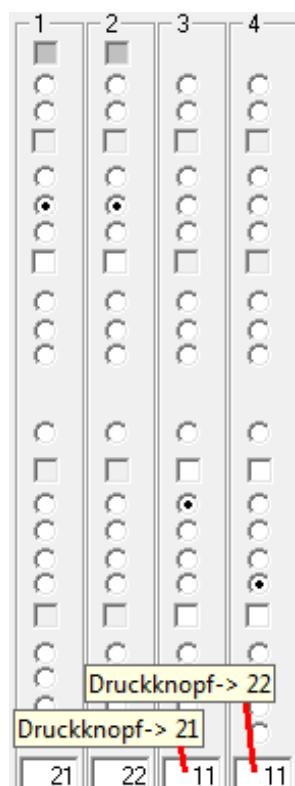
Port betätigt	Zustand Port 3	Zustand Port 4
1	1	0
2	0	1

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. 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 Blinkimpulsen<sup>6</sup> eingeschaltet, unabhängig von der Dauer des Tastendruckes. Diese Einstellung ist dann zu verwenden, wenn das auslösende Ereignis ein Dauersignal erzeugt (z.B. ein Zug löst eine Lichtschranke aus und bleibt dann in der Lichtschranke stehen).

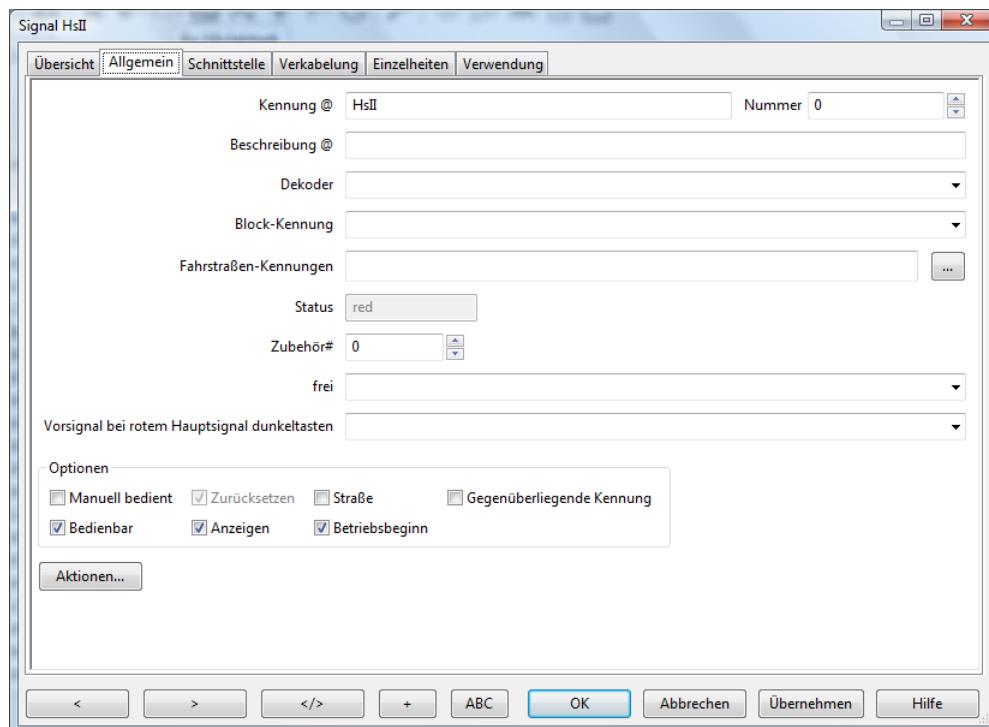
Schalttabelle:

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

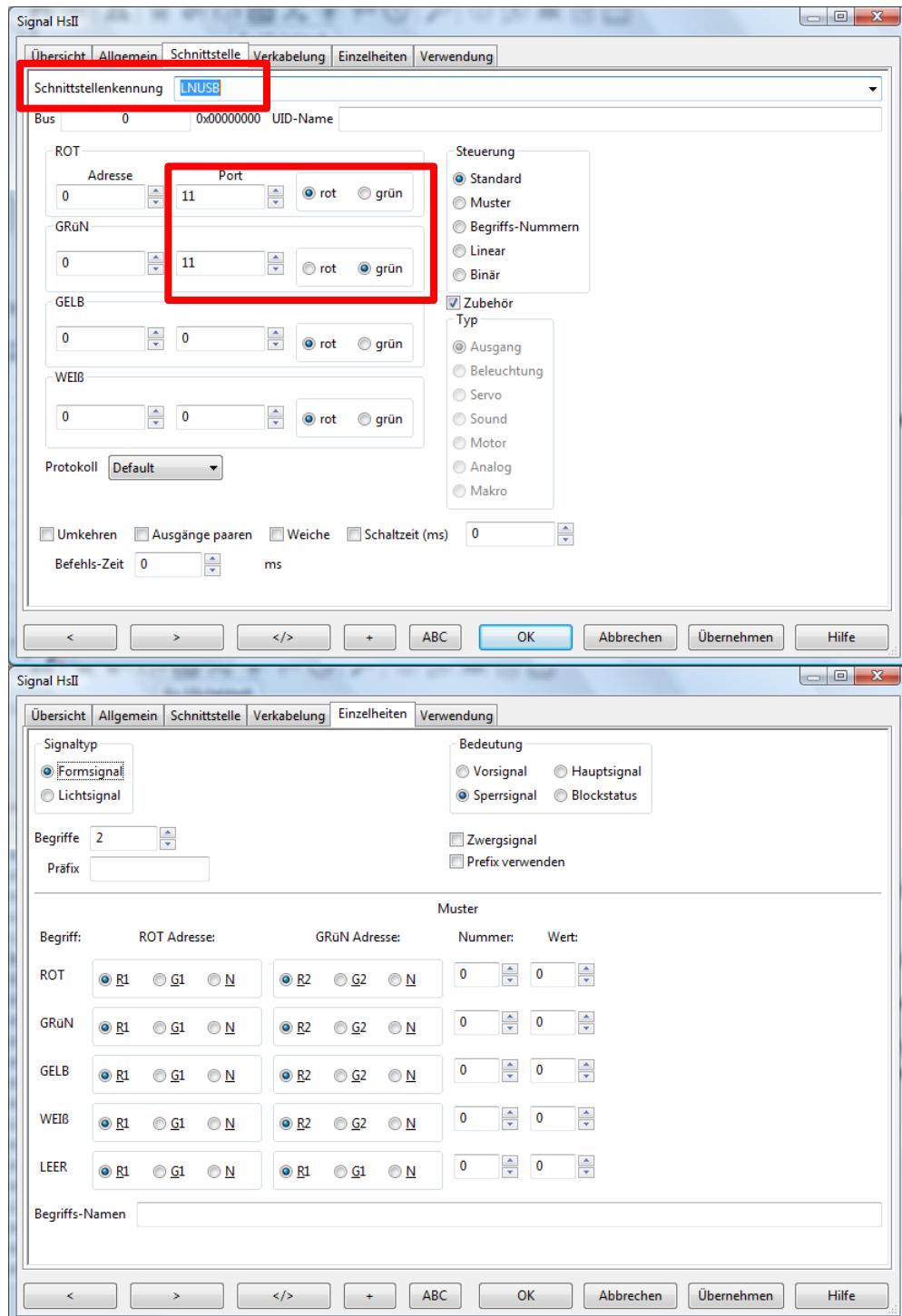
Um eine Weiche oder ein Signal zu steuern, ist das zu steuernde Element im Gleisplan zu projektieren:



<sup>6</sup> die Schaltdauer des Ausgangs (Dauer des (Blink-)Impulses) hängt von der Blinken Frequenz ab:



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

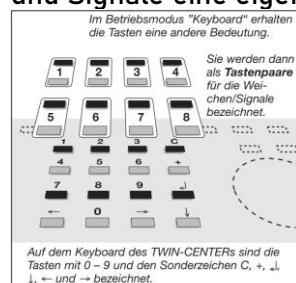
Keyboard  
Adr.: .... - ...

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)

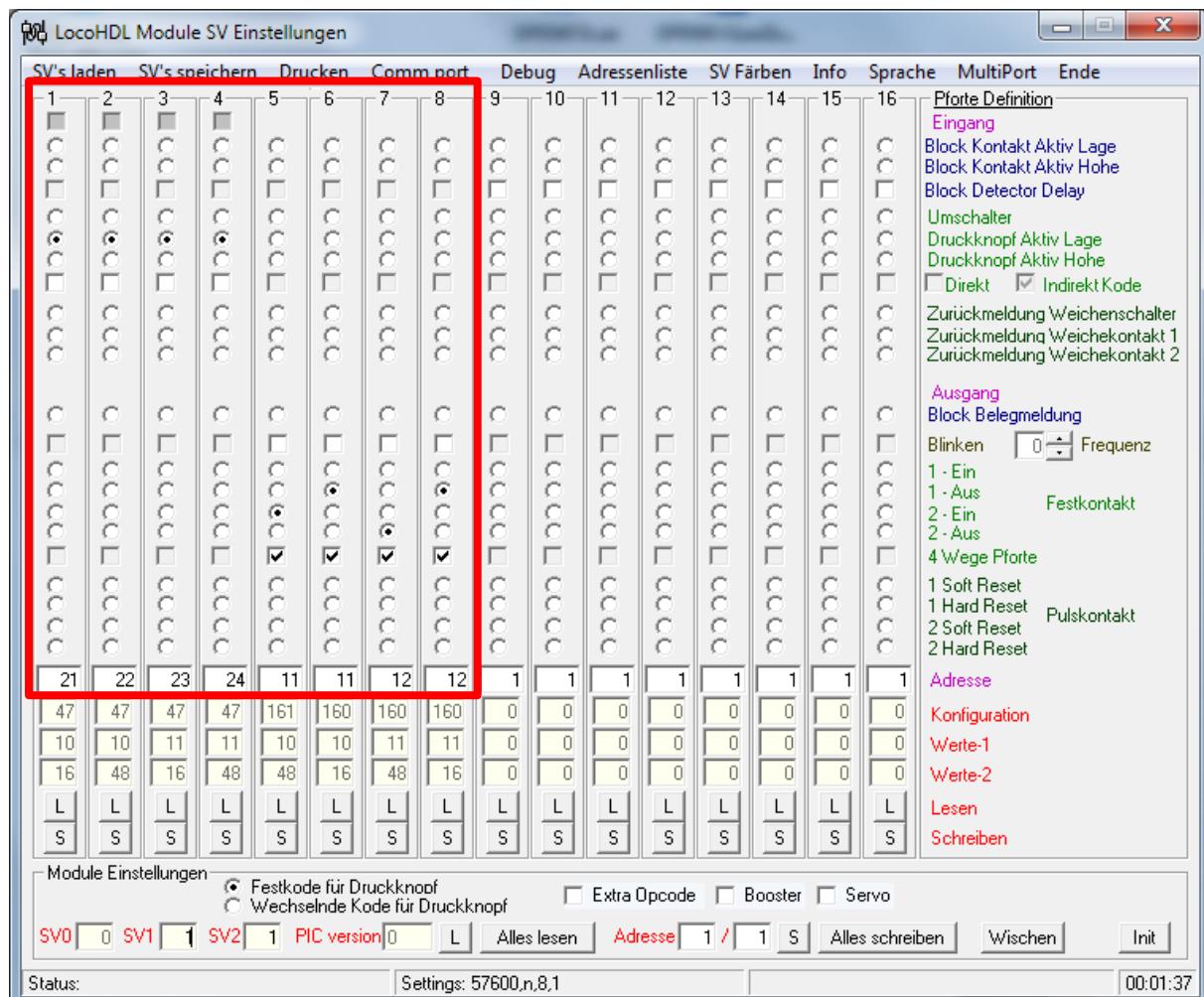
## Vier Eingänge – vier Ausgänge

(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

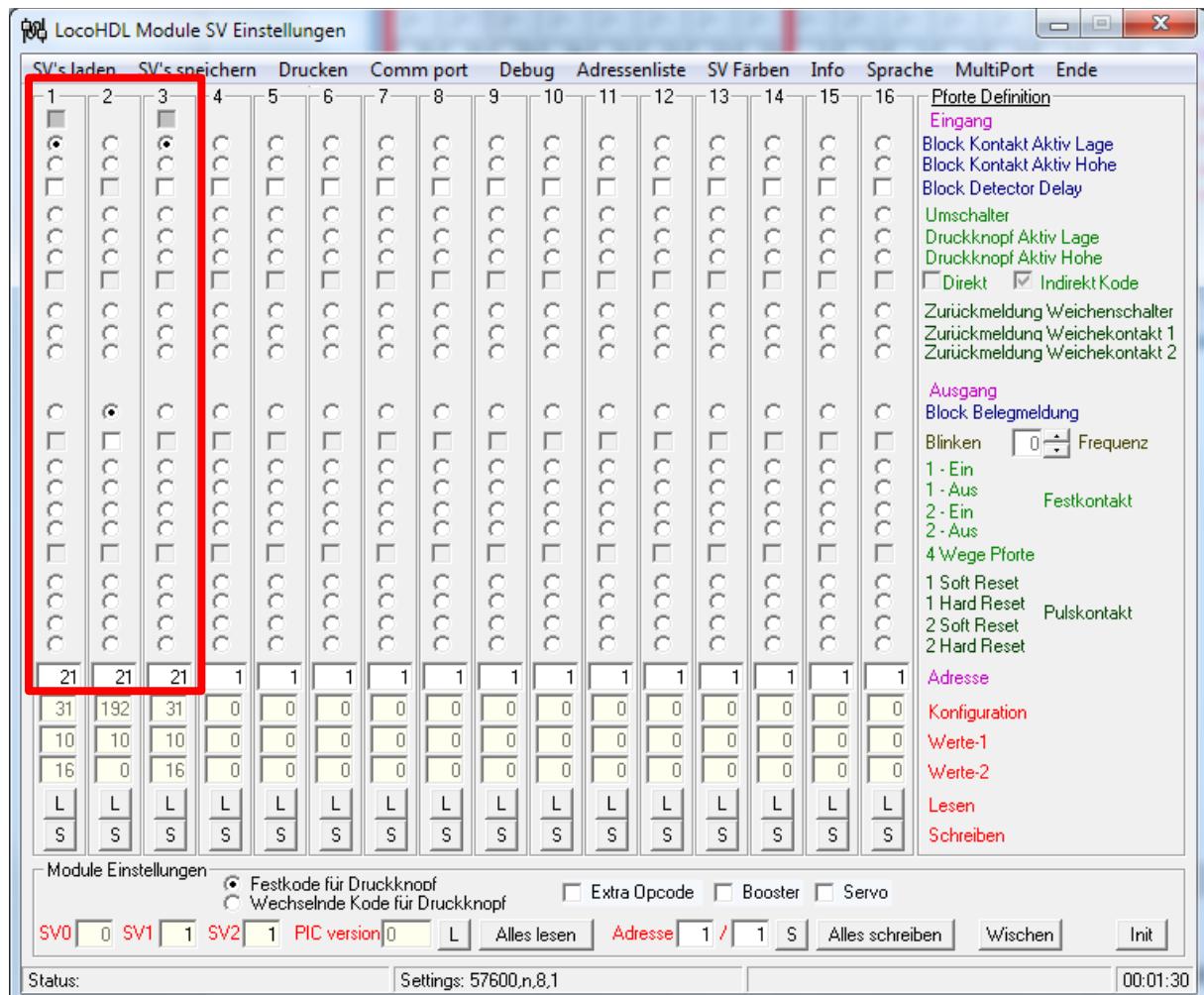
Einsatzzweck z.B.:

- 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.



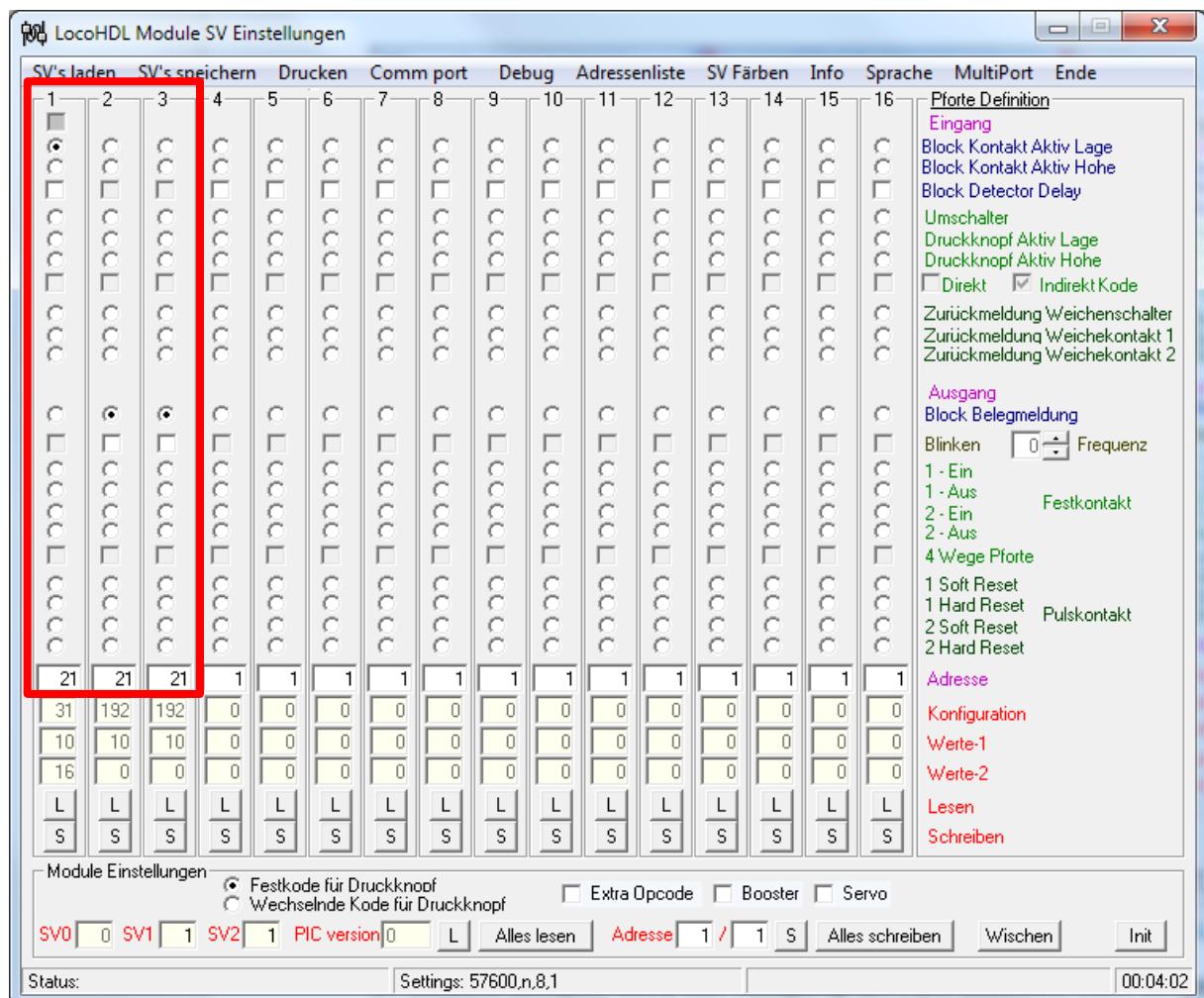
In diesem Beispiel wird der Ausgang an Port 2 sowohl vom Eingang an Port 1 als auch vom Eingang an Port 3 ein- bzw. ausgeschaltet – die beiden Eingänge wirken also wie eine Oder-Verknüpfung auf einen Ausgang.

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

## Konfigurieren mit JMRI

Die LocoIO-Module können auch über die Software JMRI (<https://www.jmri.org/>) konfiguriert werden. Hierbei ist zwar nach wie vor ein LocoBuffer<sup>7</sup> 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: „**SVs speichern**“ nicht vergessen!

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

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.

## 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

Meine Notation:

- Eine Aufteilung in eine dreiteilige „Beschreibung“ erleichtert eine Suche bzw. eine Filterung

<sup>7</sup> Ich verwende z.B. einen LocoBuffer mit einem Arduino-Nano mit zusätzlicher Hard- und Software (<https://github.com/Kruemelbahn/LocoBuffer-Nano>)

- unter „Adresse“ wird die benutzte Sensor- bzw. Aktoradresse eingetragen,
  - o Block ist die Adresse bei Verwendung als Block Belegmeldung, Umschalter oder Druckknopf
  - o 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
  - o Modul mit der Basismoduladresse 60, Submoduladresse 10, Anschluss 1
  - o B bedeutet bei mir Block Belegmeldung. Die verwendeten Buchstaben und Zahlen leiten sich von der Kurzinfo auf der rechten Dialogseite der HDL-Software 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:

- 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](http://embeddedloconet.sourceforge.net/SV_Programming_Messages_v13_PE.pdf)
- Das Original: <http://www.locobuffer.com/LocoIO/LocoIO.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-Editor/blob/main/Documentation/LocoIO\\_Erg%C3%A4nzungen.pdf](https://github.com/Kruemelbahn/LocoIO-Editor/blob/main/Documentation/LocoIO_Erg%C3%A4nzungen.pdf)
- Hinweis zum Umgang mit JMRI:  
<https://github.com/Kruemelbahn/Infoletter/blob/main/Krümelbahn%20Info%2011%20-%20JMRI%20-%20Universalwerkzeug%20für%20die%20Modellbahn.pdf>