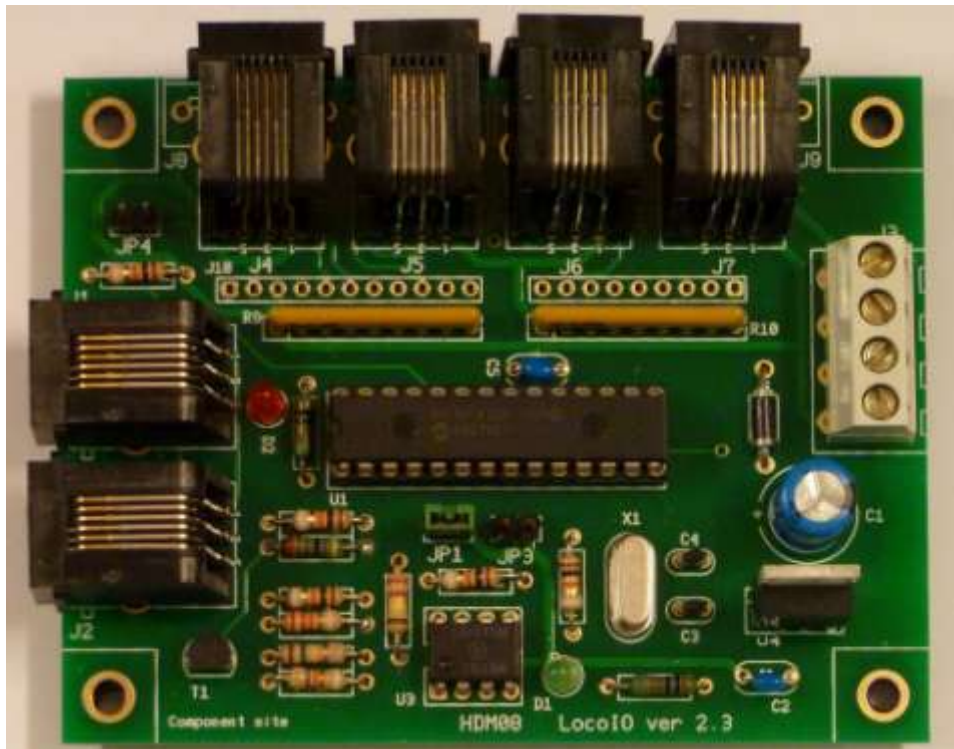


# *Loco10*

## *Users Manual*



## HDM08

### Liability disclaimer:

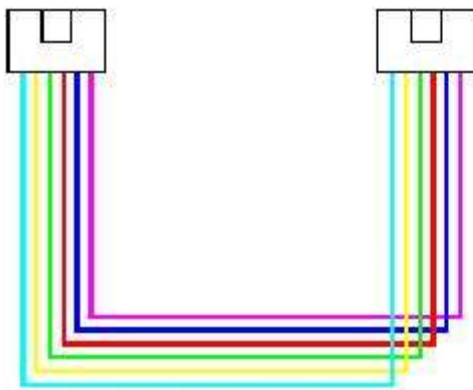
Use all items that can be bought and installation instructions that can be found on this site at your own risk. They have been developed for personal use, and I find them very useful. That is why I wish to share them with other model railroad hobbyists. All items and procedures have been tested and used on my own model railroad systems without causing any damage, but this does not necessarily imply that all modifications and procedures will work in any and all environments or systems. I cannot take any responsibility when items or procedures are used under different circumstances. Always use your own judgement and common sense!

# LocoIO module hardware 2.3

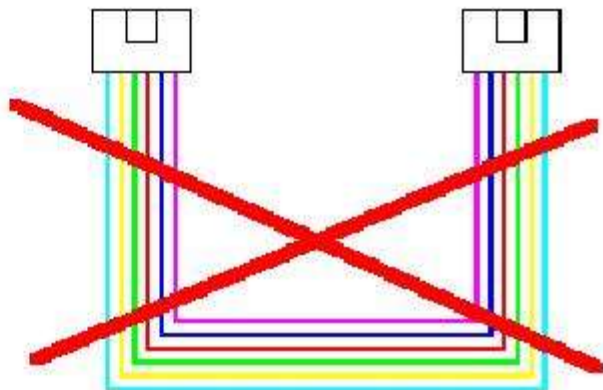
This is a 16 bits input/output module with connection to Loconet. It is possible to connect it with signals, switches, current sensors, push buttons, ... . Now also with blinking outputs and 4-way signal output. Basically what this device does is converts a 0 or +5 volt signal to a Loconet message and/or converts a Loconet message to a 0 or +5 volt signal, and this for 16 independent bits. It also does software de-bounce of the input.

## **Loconet connection:**

The connection to Loconet is with a 6 wire cable with RJ12 connectors. Important is that on the connector on both ends of the cable the pin1 to pin1 is connected.



**Correct**



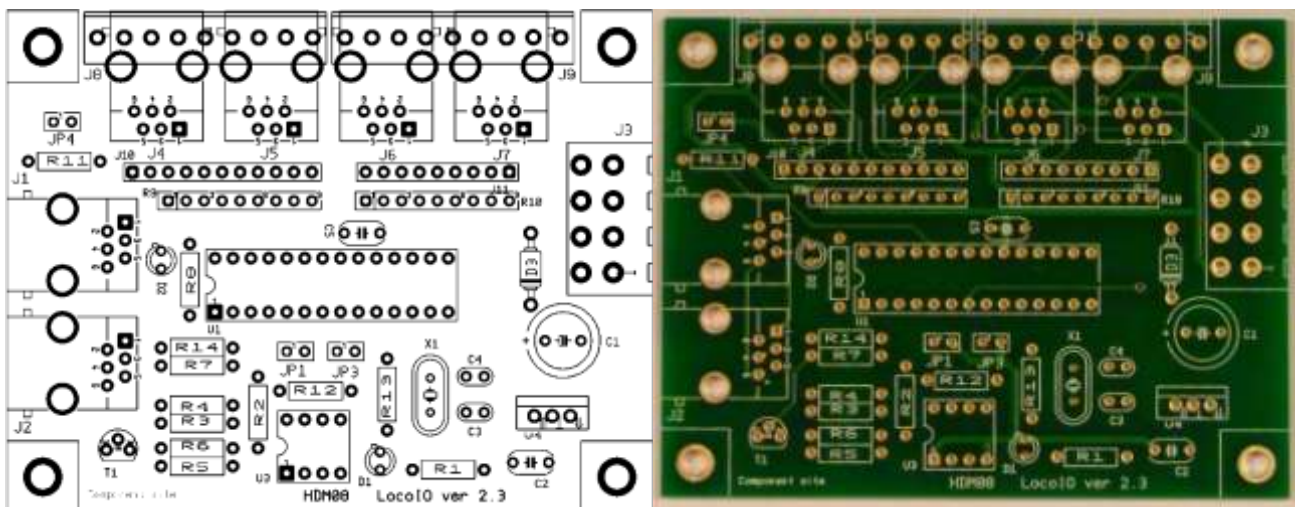
**Wrong**

## Bill of materials:

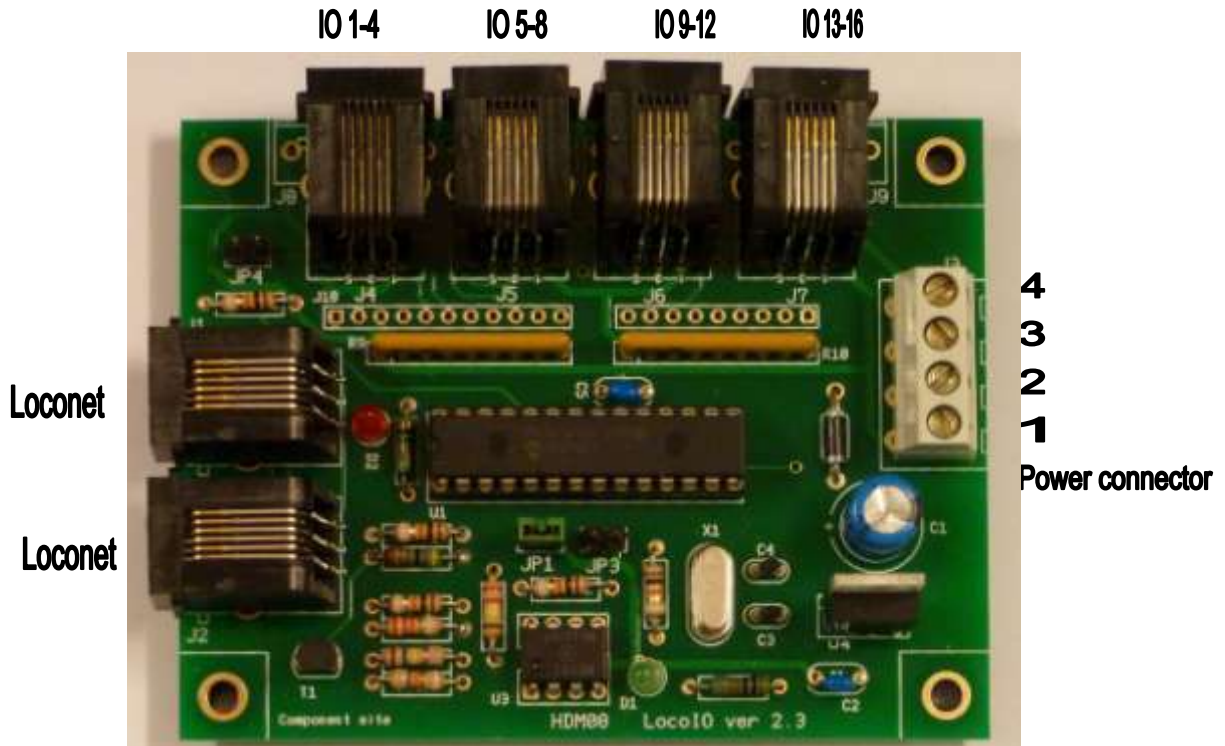
UT_DEVICE	UT_VALUE	Refdes
Voltage Regulator	7805	U4
Diode	1N4001	D3
Capacitor	100nF	C2, C5
Capacitor	15pF	C3, C4
Comparator IC	LM311N	U3
ELCO	100µF/25V	C1
HDR_4	4 pins print connector (5.08)	J3
HDR_9	9 pins	J10, J11
JUMPER2	2 pins	JP1, JP3, JP4
LED Ø3mm	Green	D1
LED Ø3mm	Red	D2
NPN transistor	BC337	T1
PIC	PIC16F873-20/SP or PIC16F873A-I/P	U1
Resistor	10kΩ	R4, R11, R12, R13, R14
Resistor	150kΩ	R6
Resistor	1kΩ	R1, R8
Resistor	220kΩ	R2
Resistor	22kΩ	R3
Resistor	47kΩ	R5
Resistor	4k7Ω	R7
Resistor pack	8x10kΩ (9 pins sil)	R9, R10
Connector	RJ12	J1, J2
XTAL	Quartz 20MHz	X1
Connector	RJ12	J4, J5, J6, J7
Or		
HDR_9	9 pins print connector (3,81)	J8, J9

## Remark:

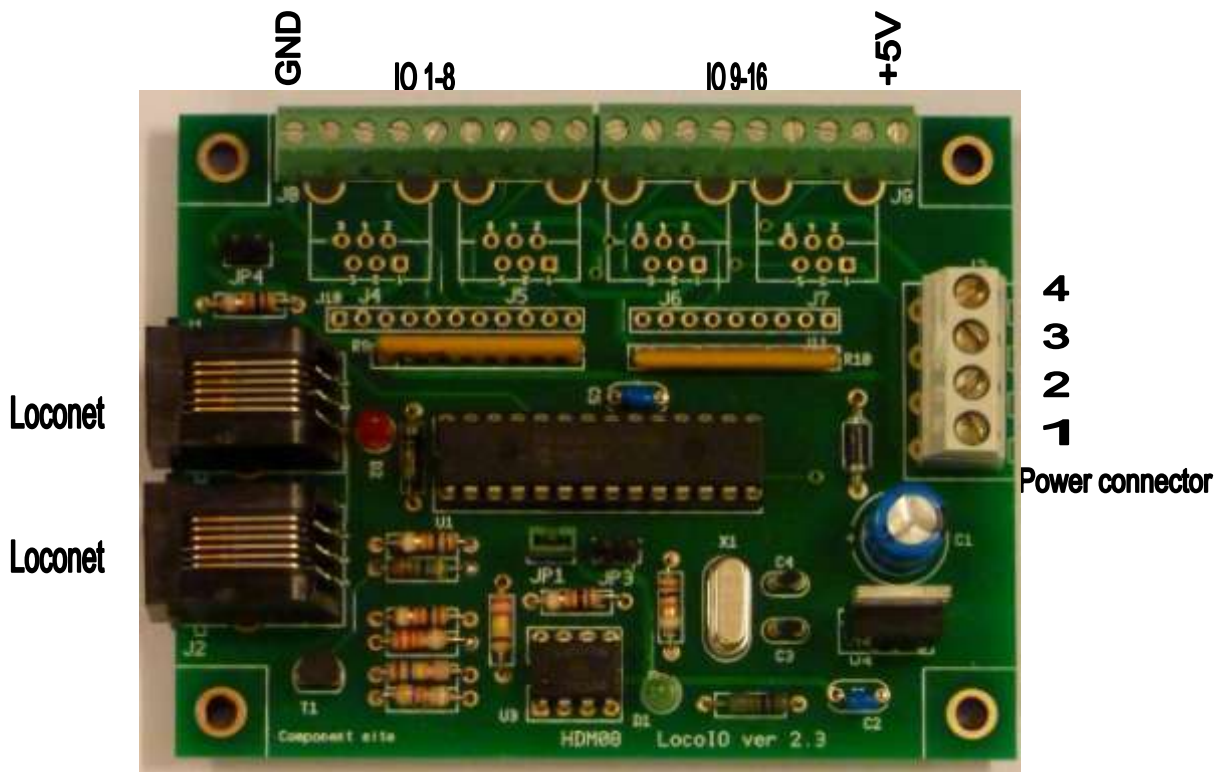
- Put the PIC on an IC socket then you can also program this PIC on the Locobuffer module.
- If your XTAL component is in metal, look that there is no contact between the metal surface of the XTAL and the solder holes.
- Every IO pin as output can give max. 20mA.
- The entire IO pins together max. 200mA
- Headers J10 and J11 are an option.
- RJ12 connectors J4, J5, J6, J7 can be replaced by print connectors J8 and J9.



Standard version with RJ12 connectors for extensie modules and simple connections.



Possibility to make wire connections with PCB connectors J8 and J9 or with pin connections J10 and J11



Remark: Two RJ12 connectors and 1 PCB connector is also possible.

### Jumper setting:

JP1	Off	Input status on power up and after a Loconet disconnection.	
	On	Without input status	
JP3	Off	normal	This jumper is reserved for maybe something later.
	On	not used	
JP4	Off	normal	This jumper is reserved for maybe something later.
	On	not used	

### Power connector J3 possibilities:

#### Input:

Pin 2: 12V-15V DC input  
Pin 4: GND input

#### Output:

Pin 1: 5V output  
Pin 4: GND output

### IO connectors:

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

### Red LED:

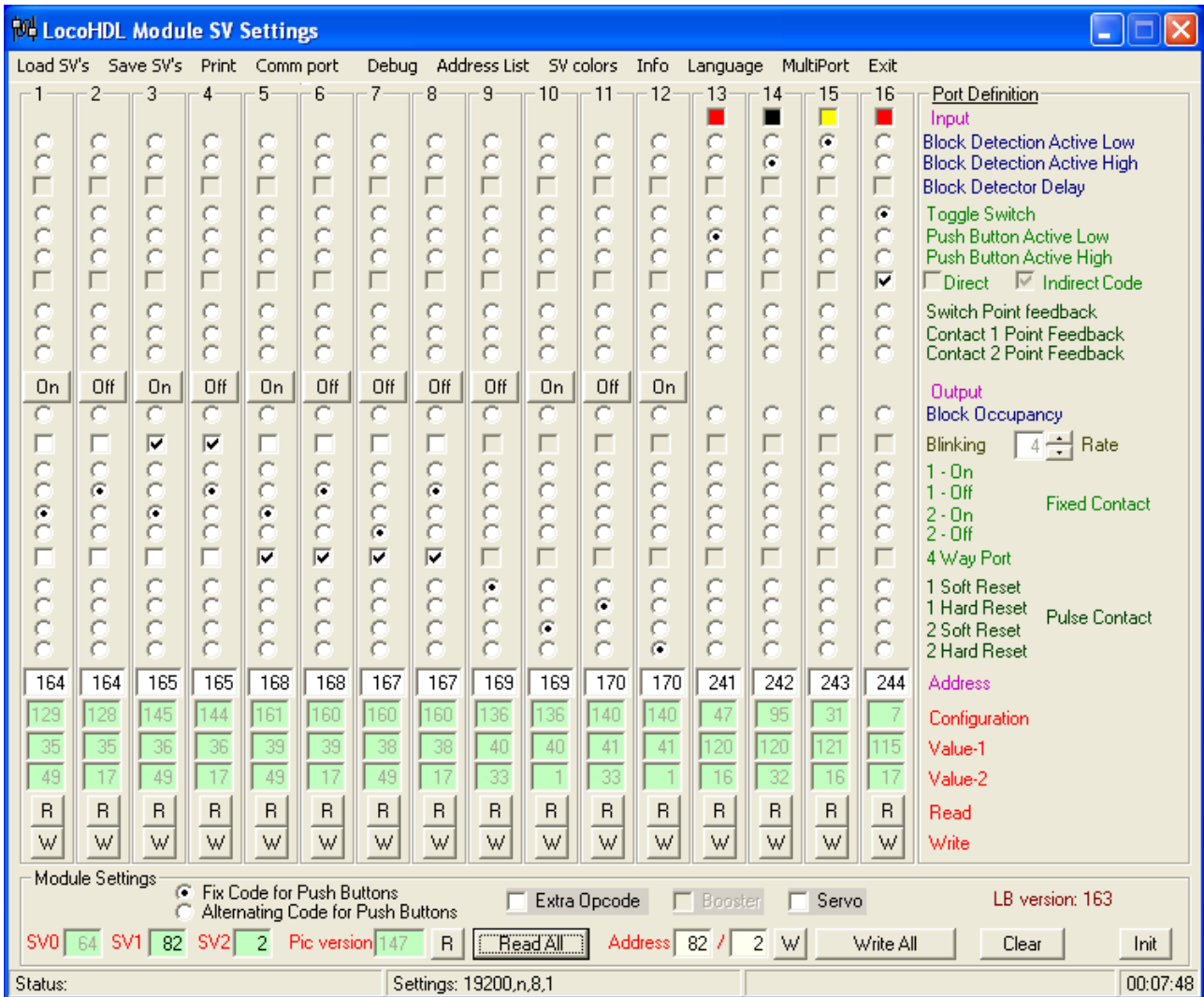
On	No central station connected or current source on Locobuffer with J6 selected.
Off	Loconet OK, no activity
Blinking	Loconet command transfer

### Remarks:

- User of a Digitrax Command Station DB150 need to set JP1 On.
- With a DC power supply is the GND the same as an Intellibox or Locobooster.

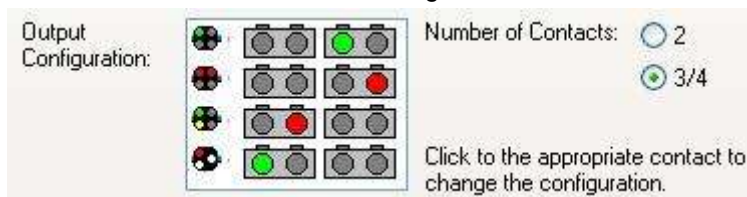


# LocoHDL configuration program example for LocoIO:



IO 1 and 2 is a red/green signal (1=red, 2= green) with address 164  
 IO 3 and 4 is a blinking red/green signal with address 165  
 IO 5,6,7 and 8 is a 4-ways signal with address 167 (168)

Railroad & Co TrainController settings.



IO 9 and 10 is a point with coils with only software pulse length  
 IO 11 and 12 is a point with coils with software and/or hardware pulse length  
 IO 13 is a push button active low that sends OPC\_SW\_REQ (0xB0) codes  
 IO 14 is block detection active high  
 IO 15 is block detection active low  
 IO 16 is a toggle switch that sends OPC\_SW\_REP (0xB1) codes

## Some LocoIO possibilities

### Block detector input:

Send a command OPC\_INPUT\_REP (0xB2) to Loconet.

Configuration byte = 31 (1FH) *then port is block detection input active low*

Configuration byte = 27 (1BH) *then port is block detection input active low and switch off delay*

Configuration byte = 95 (5FH) *then port is block detection input active high*

Configuration byte = 91 (5BH) *then port is block detection input active high and switch off delay*

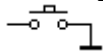
Remark: The switch off delay depends on the established blinking rate.

Advice: Use the active low input. On the LocoIO is already a pull-up resistor that makes it easy to use the port. Active high is implemented just for some modules that give only an active high output.

-The Current Sensor makes it easy to continuous detect trains or current consuming things on section of your layout.

-Contacts to ground (active low) as Reed contacts, contact rails, ... and gives information of passing trains, specific wagons with magnetic strips,...

-With a Push button to ground you can also use it to give a feedback to the computer that then take some action as starting a route or schedule.



### Toggle switch input:

Send a "Direct" command OPC\_SW\_REQ (0xB0) to Loconet.

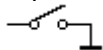
Configuration byte = 15 (0FH)

Or

Send an "Indirect" command OPC\_SW\_REP (0xB1) to Loconet.

Configuration byte = 07 (07H)

-With a single switch and one input connection you can make a High (internal pull-up) or Low (switch to ground). With this you can make a signal red/green, set points right or left, ... Useful on a panel, but limited to only 2 operations.



### Push button input:

Send a "Direct" command OPC\_SW\_REQ (0xB0) to Loconet.

Configuration byte = 47 (2FH) *then port is push button input active low*

Configuration byte = 111 (6FH) *then port is push button input active high*

Or

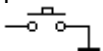
Send an "Indirect" command OPC\_SW\_REP (0xB1) to Loconet.

Configuration byte = 39 (27H) *then port is push button input active low*

Configuration byte = 103 (67H) *then port is push button input active high*

Advice: Use the active low input. On the LocoIO is already a pull-up resistor that makes it easy to use the port. Active high is implemented just for some modules that give only an active high output.

-Contacts to ground (active low) with a push button, then you can set a signal to red/green/orange, ... This has the preference for panel operating.



-Contacts to ground (active low) as Reed contacts, ... you can change signals and points. This is useful if you work without PC layout. Do not connect Contact rails directly to an Input without optical isolation (example: HDM10 as Voltage detector).

-With alternated code you can make a signal red/green, set points right or left, ... Useful on a panel, but limited to only 2 operations.

Remark: This is only advised if you have a visual control about what you are doing.

### Description of "Direct" and "Indirect" code.

With the "Direct" code a command will be issued to Loconet, which will immediately start controlling those points and signals.

This is the usual setting when no PC is used. If one does use a PC, it will receive the command for the point or the signal, even if the software does not permit the changes. In that case the view on the screen will not be consistent with the real state of the point or signal.

With the "Indirect" code a command will be issued to Loconet, which is first evaluated by the PC, to decide whether or not it will be executed. Next the PC will display the actual command that is sent to Loconet which will influence the point or signal.

### Remark for Railroad & Co Traincontroller users:

You need to set the following option in RAILROAD.INI to be able to accept "Indirect" codes:

```
[Connections]
```

```
LNWatchTurnoutOutput = 1
```

**After reception of an Indirect code, and after a positive evaluation Traincontroller should send a Direct code to Loconet. Unfortunately it does NOT.**

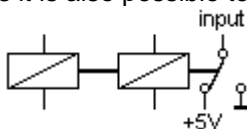
**It is therefore best to link an indirect code to a Traincontroller switch or push button, which can then perform certain actions.**

### Switch feedback input, 2 methods:

Send an OPC\_SW\_REP (0xB1) to Loconet.

Configuration byte = 23 (17H)

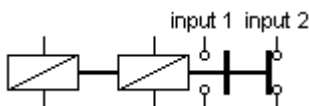
-If we place a double switch on a point coil than can this give to the PC the real position of the point. With this it is also possible to change the points with hand and follow this on the PC.



Send an OPC\_SW\_REP (0xB1) to Loconet.

Configuration byte = 55 (37H)

With this configuration we need two inputs to verify a point. The advantage is feedback for slow motion point motors.



Remark for Railroad & Co Traincontroller users. You need to set the following option in RAILROAD.INI to be able to accept this Switch feedback codes:

```
[Connections]
```

```
LNWatchTurnoutOutput = 1
```

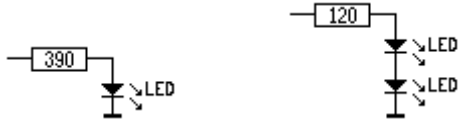
## Block occupancy output:

Receive an OPC\_INPUT\_REP (0xB2) from Loconet.

Configuration byte = 192 (C0H) *normal block detection output*

Configuration byte = 208 (D0H) *blinking block detection output*

-On a panel you can place LED that gives you an indication where trains are in your layout.  
Direct to LocoIO you can connect 1 or 2 LED's. The output is max 5V/20mA.



If you want more LED's to an output then you must use the Driver Module.

## Fixed contact output:

Receive an OPC\_SW\_REQ (0xB0) from Loconet.

Configuration byte = 128 (80H) *normal output, output off at power up*

Configuration byte = 144 (90H) *blinking output, output off at power up*

Configuration byte = 160 (A0H) *a 4-ways output, led off at power up*

Configuration byte = 176 (B0H) *blinking 4-ways output, led off at power up*

Configuration byte = 129 (81H) *normal output, output on at power up*

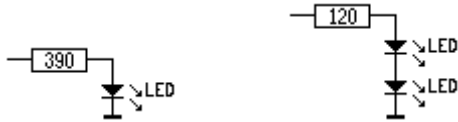
Configuration byte = 145 (91H) *blinking output, output on at power up*

Configuration byte = 161 (A1H) *a 4-ways output, led on at power up*

Configuration byte = 177 (B1H) *blinking 4-ways output, led on at power up*

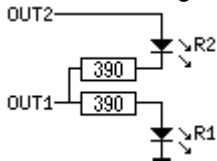
-This output you can use for signals, lamps,...

Direct to LocoIO you can connect 1 or 2 LED's. The output is max 5V/20mA.



For commercial signals who needs more voltage or lamps who needs more current, you can use the Driver Module or Driver Module 2.

-Two alternated blinking LED's, on and off switch.



Out2: Fixed contact, 2 Off

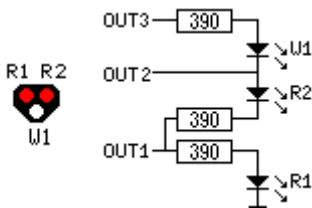
Out1: Fixed contact, 2 Off, Blinking

Both have the same address

Configuration byte: 128

Configuration byte: 144

-Belgian road cross signals: two red LED's alternated blinking or white blinking



Out3: Fixed contact, 1 On, Blinking

Out2: Fixed contact, 2 Off

Out1: Fixed contact, 2 Off, Blinking

All three have the same address

Configuration byte: 145

Configuration byte: 128

Configuration byte: 144

**Remark:** The resistor with the LED is the minimum recommended value. The resistor value can be higher if the LED is bright to high.

-This output together with the Driver Module 2, for more Current and voltage, is useful with point motors with end switches.

## **Pulse contact output:**

Receive an OPC\_SW\_REQ (0xB0) from Loconet.

Configuration byte = 136 (88H)

*The pulse contact output is been set high by a software command from Loconet and set low by a second software command. The length of the pulse depends of the time between the two commands given by Train software. Or by a button is the length of the pulse the time that we push the button.*

-This output together with the Driver Module, for more Current and voltage, is useful with coil mechanism for decouple rails. You will that the decouple mechanism is active during the time you push the button.

Configuration byte = 140 (8CH)

*The pulse contact output is been set high by a software command from Loconet. The length of the pulse is then between 1 and 2 blink pulse length. Or with a button does LocoIO now define the length of the pulse. If a software command for reset the output comes before the hardware security, then the output works as configuration byte 136.*

-This output together with the Driver Module, for more Current and voltage, is useful with points. The output with this configuration gives you a security that the coil is not burning out if the end pulse command is not coming.

- If the command for the Pulse Contact is coming from a reed relay, contact rail,... then the coil of a point is not continue activated if the train stand still on this reed relays or contact rail. This pulse length security works as an end switch on some points.

## **Servo output:**

For analogue Servo engines, look at the Servo module manual.

## **! RECOMMENDATION:**

- If the used version of LocoIO has not the option selection "Port not used", then it is been advised to define unused ports as output ports, and give them an unused address. Then you prevent that interference on the module will send out undesirable codes on Loconet.

- Always give ports that are used as Servo output an address that occurs only once on a module.

# History of the versions

## LocoIO version 1.35 until version 1.40

- no longer supported

## New in version 1.41

- Block detection command are changed

-In previous versions the block detection has configuration byte 15 (0FH) with value 1 bit 7 high. All block detection's are active low or high depends on SV0 bit 4.

-In version 1.41 are two configuration bytes for block detection.

31 (1FH) then port is block detection input active low

95 (5FH) then port is block detection input active high

SV0 bit 4 and value 1 bit 7 is no longer used.

The advantage of this is, every block detection input can now been set separately to active low or high.

- Push button send now two codes on loconet, one for pushing the button and one if you release the button. Now it is possible to direct activate a coil from buttons on a panel.

- The codes send on loconet are now full as discribed in the loconet specificaties.

## New in version 1.42

- The output configuration byte has a new function

configuration bit 2 = 0 software pulse reset

configuration bit 2 = 1 hardware pulse reset

- Result:

- configuration byte = 136 (88H)

The pulse contact output is been set high by a software command from Loconet and set low by a second software command. The length of the pulse depends of the time between the two commands given by a Train software. Or by a button is the length of the pulse the time that we push the button.

- configuration byte = 140 (8CH)

The version 1.42 have now a security builds in LocoIO with configuration byte 140) that, if the command to reset is not coming, the output reset. The length of the pulse is then between 1 and 2 blink pulse length. Or with a button does LocoIO now define the length of the pulse. If a software command for reset the output comes before the hardware security, then the output works as configuration byte 136.

## New in version 1.43

- OPC\_SW\_REP (0xB1) can now been send for toggle switch and push button.

Indirect changing of switches and signals in Train software without the switches and signals send direct commands.

Remark for Railroad & Co Traincontroller users:

You need to set in RAILROAD.INI the follow option to accept this codes:

[Connections]

LNWatchTurnoutOutput = 1

- The push button input can now also work with double codes.

With this it is possible to change switches and signals with only push button.

Remark: This is only advised if you have a visual control about what you are doing.

## New in version 1.44

- New configuration byte for point's feedback

- configuration byte = 23 (17H)

-With this it is also possible to change the points with hand and follow this on the PC.

-If the input becomes low then one direction code of the point feedback is send, if the input becomes high the other code of the point feedback is send on Loconet. Only one input is needed to check a point.

## New in version 1.45

- New configuration byte for point's feedback

- configuration byte = 55 (37H)

-With this it is also possible to change the points with hand and follow this on the PC.

-With this configuration you need two inputs to feedback a point. It is designed as an active low input. If the input becomes low it send a code of the point feedback. The advantage of using this with slow motion motors is that the feedbacks switches can be independent from each other.

-A timing problem in previous versions that exist if inputs are used together with blinking outputs, are now solved.

-There is now a security build in to prevent that both pulse contacts with the same address are activated together.

This prevent that both coils of a point have current at the same time.

- Entrances can send outside their specific Loconet code, also an extra 4-byte Loconet code.

-JP1 enable and disable function for input status.

### **New in version 1.46**

- Apart from the normal address LocoIO now has a sub-address as well. This can be used to give certain areas of a track a fixed address and use the different sub-addresses for the LocoIO within a zone. In a modular rail track the different club members can be given one LocoIO address and on top of that every club member can use several LocoIO's with different sub-addresses.

### **New in version 1.47**

- In this version will the ports be refreshed , without switching the power off /on. This can be done with LocoHDL configuration program by writing of 'SV0' or a "Write All".

### **New in version 1.48**

- Sub-addresses in Multiport Command
- Optional switch off delay for block detection

### **New in version 1.49**

- Each port can be select as unused
- The last version for PIC 16F873(A)

### **New in version 1.50**

- For PIC 16F883
- Extra Opcode 2
- Double Input
- Special Ports for GO-IDLE-STOP inputs and STATUS output.