Protocol – Description

Interface 10785

(Firmware – Version 1.6)

Modelleisenbahn GmbH, Plainbachstraße 4, A-5101 Bergheim, Austria
Tel.: +43 (0)5-7626-0, Fax: +43 (0)5-7626-1599
Web:www.roco.cc , Email:roco@roco.cc

1 General

This document describes the communication at the interfaces of the Roco Interface 10785.

1.1 Properties of the interface

The ROCO Interface is the connection device between the Roco Digital system and a PC. It expands the Roco Digital system with the feedback and programming track functions, with the option of programming and reading locomotive decoders using the PC. It also provides additional power for input devices and feedback devices on the RocoNet.
The interface transmits the control commands from the PC to the RocoNet and passes information from the RocoNet to the PC.

The PC's control commands can:

control locomotives
setting the course
programming locomotives (directly via the built-in Prog. track connection)
Reading locomotives (directly via the built-in Prog. track connection)

With these commands, the PC supports all the capabilities of the respective system:

- The Lokmaus2/R3 system can address 99 locomotive addresses and 256 switches.
- At the***multi***The MAUS system can address 9999 locomotive addresses and 1024 switches.
  (These limitations come from the respective master mouse, not the interface.)

The Lokmaus2/R3 does not support the full range of programming functions on the programming track connection of the interface.
The interface also allows three-digit inputs for programming. The interface can also read the decoders and display three-digit values.

The locomotive (or decoder) is programmed on a separate track, the programming track. The advantage of this technology is the safety that programming is carried out with reduced energy, so the locomotive receiver is largely protected from damage even if it is connected incorrectly. In addition, operation on the system can continue without interruption by programming. All other locomotives can remain on the system while a locomotive is being reset on the programming track.

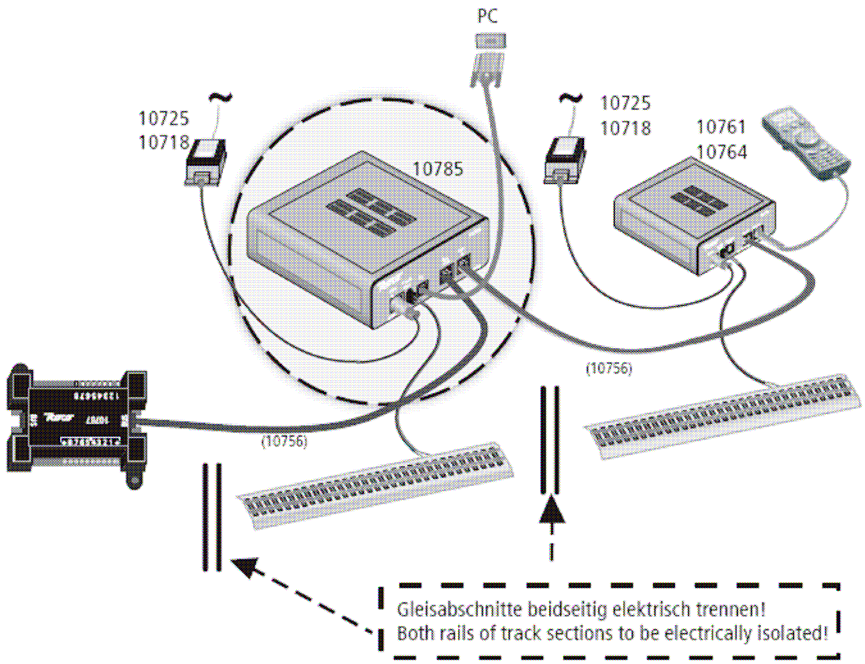
The PC receives the following information from RocoNet, if requested:

- status of locomotives
- Status of switches (last setting command)
- system status (operation, short circuit, etc.)
- Return message

1.2 Connecting the interface

The interface is connected to the RocoNet as a slave. The interface cannot be a master, which means that a master Lokmaus (or another master) must be present.
The interface receives its energy either from its own transformer, preferably the 10725 transformer, or directly from the RocoNet.
The interface can only program and read decoders if it is powered by its own transformer.
It is connected to the computer's serial interface (RS232) using the cable provided. For EMC reasons, only this cable may be used.

Fig. 1



The interface provides additional power for the RocoNet if its own transformer is connected. The RocoNet as a data bus, i.e. the information to the master and back from the master, simply runs through the interface. The power supply for the input devices on the RocoNet is normally provided by the amplifier. When the interface is used, only the devices connected between the main amplifier and the interface are supplied by the amplifier. After the interface, they are supplied with power by the latter. There is a built-in protective circuit to prevent the amplifier from feeding power back through the interface.

The interface communicates with the PC via RS232 with**19200 baud**and pure software handshake.

On the RocoNet side, the command set corresponds to the XpressNET protocol[1] Version 3.0 from Lenz Elektronik GmbH, supplemented by special commands for organizing and evaluating feedback and for programming via the built-in programming output.

Connections:

RS232	4-pin Western socket
transformer in	Hollow plug socket 5.5 mm 6-pin
RocoNET in	Western socket RJ-14 (6P4C) 6-pin
RocoNET out	Western socket RJ-14 (6P4C) Socket for
programming track	Roco track connection plug

LED displays:

Comm	During normal operation, the green LED indicates the data traffic between the interface and the PC via the serial interface.
Prog	The red LED lights up during the programming process (reading or writing CV values) and flashes when there is no connection to the PC.

The green LED flashes together with the red LED if no master mouse is connected.

[1] ***XpressNET is a trademark of Lenz Elektronik GmbH, Hüttenbergstrasse 29 - D 35398 Gießen – Germany***

2 Communication

2.1 Communication with the master mouse

The interface uses the Lenz XpressNET° version 3.0 protocol to communicate with the master mouse and can be obtained from Lenz Elektronik GmbH.

The protocol is supplemented by special commands for organizing and evaluating the feedback and for programming via the built-in programming output. This additional command set is listed below. The interface autonomously handles the protocol routines and the error handshake on the bus. This is not made transparent to the PC. In the event of an error, an error message is sent to the PC if necessary or possible.

IMPORTANT:

In contrast to the ***multi***MAUS, the Lokmaus2/R3 does not support the entire range of commands of this protocol. The restrictions on this can be found in the documentation for the Lokmaus 2/R3. The same restrictions naturally also apply to the interface, as it relies on the Lokmaus2/R3 as master to implement the commands. It generates the track format and organizes communication on the bus and therefore cannot be bypassed by the interface.

Since the interface has its own fully functional programming output, it does not make sense to use the rudimentary programming functions of the Lokmaus 2/R3 for programming.

2.2 Communication of the interface with the PC

The interface communicates with the PC via RS232

Parameter:

Baud rate:	19200
Parity:	None
data bits:	8
stop bits:	1
Handshake:	Software

The command syntax is not compatible with other interfaces. However, the useful information is based on the commands as they are transmitted on the RocoNet, so that the interface does not have to perform complex conversions.
The command set between PC and interface is described separately below.

2.3 Data transfer PC↔Interface / RocoNet

The communication is based on a pure software handshake, ie after a data transfer the PC may only send something to the interface again if the processing has been confirmed by the interface or an error message has been sent back. An earlier second transfer may be lost and is therefore not permitted.

The data stream basically consists of the following components:

info byte The info byte indicates the type of data packet transmitted and is not forwarded to the RocoNet. Is not taken into account when calculating the XOR.

header The header is the first byte of the actual data transfer. Based on the XpressNet protocol version 3.0 from Lenz. This is the start of the data stream that is transmitted on the bus. Additional headers for feedback and programming are listed separately below.

Data The data following the header corresponds to the schema of the XpressNet protocol version 3.0 by Lenz. Additional commands for feedback and programming are listed separately below.

XOR The XOR byte is the last byte of the transmission and is calculated as the XOR of the header byte and the data bytes. The info byte is not taken into account.

2.3.1 PC→interface

The serial data transfer from the PC to the interface begins with sending a data packet to the interface. Based on the info byte, the interface's microcontroller decides how to proceed with the data packet (specified in HEX):

0x00	Data is transferred 1:1 to the RocoNet (without info byte)
0x10	Confirmation message for received transmission / error
0x2**X**	message concerns feedback and its configuration (see below)
0x4**X**	concerns programming (see below)

2.3.2 Interface→PC

Serial data transmission from the interface to the PC is normally the result of the transmission of a data packet to the interface. However, spontaneous transmission by the interface occurs with general call messages on the RocoNet and with the transmission of feedback information.

The PC uses the info byte to identify the origin of the data (indicated in HEX):

0x00	Data comes 1:1 from RocoNet or confirmation message for received transmission / error message
0x2**X**	concerns feedback and its configuration (see below)
0x4**X**	concerns programming / reading (see below)

All transfers must be acknowledged by the PC with confirmation. **error message (0x10)**

3 commands

3.1 Communication

3.1.1 PC→interface

3.1.1.1 Confirmation of data receipt

info
0x10

Each data packet from the interface must be acknowledged with the confirmation message 0x10!

3.1.2 Interface→PC

3.1.2.1 Receive data packet

info			XOR
0x00	**0x01**	**0x00**	**0x01**

Data packet received, XOR of the packet OK, serial input buffer free. Interface response to all commands from the PC, with the exception of the confirmation message 0x10.

3.1.2.2 Serial input buffer full

info			XOR
0x00	**0x61**	**0x81**	**0xE0**

The previous command could not yet be transmitted to the master, the packet just sent by the PC is not processed and is discarded, the packet must be sent again by the PC.

3.1.2.3 XOR errors

info			XOR
0x00	**0x01**	**0x01**	**0x00**

The XOR of the data packet just received is incorrect. The packet is discarded and must be sent again by the PC.

3.2 Programming

3.2.1 PC→interface

3.2.1.1 Writing a CV

info	header	CV write	# CV	Value	xor track	XOR
0x40	**0xF4**	**0x7C**	**CV-1**	**byte**	**xor**	**XOR**

CV: CV to be written, starting at 0 = CV1 Range: 0x00 - 0xFF

Value: Value to be written to the CV. Range: 0x00 - 0xFF

xor track: XOR over the three bytes: "CV write", "#CV" and "value".

The command switches on the programming track and transmits reset packets and then programming data to the programming track according to NMRA.

3.2.1.2 Reading CV bit by bit

info	header	CV read	# CV	CV bit	xor track	XOR
0x41	**0xF4**	**0x78**	**CV-1**	**0xE8**	**xor**	**XOR**

CV: CV to be read, starting at 0 = CV1 Range: 0x00 - 0xFF

CV bit: Must always be 0xE8

xor track: XOR over the three bytes: "CV read", "#CV" and "CV-Bit".

The command switches on the programming track and transmits reset packets and then CV read data to the programming track according to NMRA.

3.2.1.3 Turn off the programming track

info		XOR
0x40	**0xF0**	**0xF0**

Switches off the programming track. The interface does not transmit any reset or programming data to the programming track. This is then without power.

3.2.2 Interface→PC

3.2.2.1 No voltage on the programming track

info			XOR
0x00	**0x01**	**0x02**	**0x03**

If a programming command is sent to the interface and no separate transformer is connected to the power supply, this error message is sent.

3.2.2.2 Answer: Write a CV

info	header	# CV	Value	XOR
0x42	**0xF2**	**CV-1**	**byte**	**XOR**

CV: CV that was written to.
Range: 0x00 - 0xFF

Value: Value written to the CV. Range: 0x00 - 0xFF

3.2.2.3 Answer: Read CV

info	header	# CV	Value	XOR
0x44	**0xF2**	**CV-1**	**byte**	**XOR**

CV: CV that was read.
Range: 0x00 - 0xFF

Value: Value read from the CV. Range: 0x00 - 0xFF

3.3 Feedback

3.3.1 PC→interface

3.3.1.1 Setting the feedback repetition rate

info	header	R-rate	XOR
0x21	**0xF1**	**RRR**	**XOR**

The master control unit (***locomotive***MOUSE or***multi***MAUS) cyclically fetches the data from the activated X-Bus participants by so-called data polling.

R-rate: Sets the repetition rate in relation to the data polling by the master. The value 3 means, for example, that the feedback procedure is carried out every third poll. The value 0x00 switches the feedback off.
Range: 0x00 - 0xFF

3.3.1.2 Specify the number of modules in a group

info	header	group	Number	XOR
0x22	**0xF2**	**GRP**	**ANZ**	**XOR**

Group: Select Group 0 (0x00) or Group 1 (0x01)
Range: 0x00 - 0x01

Number: Set the number of modules
Range: 0x00 - 0x0A

Specifies the number of modules in the feedback group. There is group 0 and group 1. They are queried alternately, provided group 1 is used at all. Each group can contain up to 10 modules.

3.3.1.3 Define info byte for the group's modules

info	header	group	Info_R	XOR
0x23	**0xF2**	**GRP**	**INFO**	**XOR**

Group: Select Group 0 (0x00) or Group 1 (0x01)
Range: 0x00 - 0x01

Info_R: The info byte for the selected group

It is placed on the bus as the second byte from the interface during the feedback procedure. Info bytes can be:

Info_R-Byte								
7	6	5	4	3	2	1	0	
0	0	0	G	0	0	0	0	Normal feedback
1	0	0	G	0	0	0	0	transfer software version
1	1	0	G	x	x	x	x	Program address, xxxx = address

G: group, 0 or 1

This info byte is transmitted by the interface until it is changed. The transmission of the info byte requires that the repetition rate is > 0, otherwise no feedback procedure is carried out!

3.3.2 Interface→PC

3.3.2.1 Feedback

info	header	Info_2	Data_1	Data_2	Data_X	AD value	XOR
0x20	**0xFX**	**INFO**	**RMM_1**	**RMM_2**	**RMM_X**	**ADW**	**XOR**

Info_2: Here the info byte specified by the PC is transferred

RMM_1 - RMM_X: Data of the individual feedback modules in**ascending** address order.

AD value: Value of the built-in AD converter, which measures the power consumption on the programming track.

Outputs the feedback information to the PC. This transmission first contains the info byte of the transmission, which identifies it as a feedback, then the header and then a second info byte. This is the info byte for the feedback transmitters, which shows what type of feedback it is. The feedback is sent cyclically at intervals of approx. 1 second or as soon as the status of the feedback transmitters changes!

If a feedback device does not respond, the interface sets the data byte of the feedback device to 00h. Starting with the first missing feedback device, all data bytes of the remaining feedback devices are filled with 00h.

3.4 Additional commands

3.4.1 PC→interface

3.4.1.1 Query the FW version of the interface

info	header	XOR
0x00	**0xFF**	**0xFF**

With this message the PC can query the FW version number from the interface.

3.4.2 Interface→PC

3.4.2.1 SW version

info	header	version	compiler	XOR
0x00	**0x02**	**version**	**Comp.No.**	**XOR**

Version: Version number in BCD format xxxxyyyy, where xxxx corresponds to the “major version” and yyyy corresponds to the “minor version”.

Compiler: compiler number

With this message, the interface tells the PC which software version number and which compiler variant was used to create the software for the PC.

With the coding in BCD format, the software versions up to 15.15 are displayed.

Examples: 0x12 corresponds to version 01.02
0x2e corresponds to version 02.14
0xb3 corresponds to version 11.03

3.5 Command list

3.5.1 PC→interface

command	info	header	Byte 1	Byte 2	Byte 3 B	yte 4 B	yte 5 B	yte 6
confirmation of data receipt	0x10							
Writing a CV	0x40	0xF4	0x7C	# CV	Value	xor	XOR	
read CV	0x41	0xF4	0x78	# CV	0xE8	xor	XOR	
Turn off Prg track	0x40	0xF0	0xF0					
Set refresh rate	0x21	0xF1	rate	XOR				
Set the number of modules	0x22	0xF2	Grp	number	XOR			
Set Info Byte	0x23	0xF2	Grp	info	XOR			
query software version	0x00	0xFF	0xFF					

3.5.2 Interface→PC

command	info	header	Byte 1	Byte 2	Byte 3 B	yte 4 B	yte 5 B	yte 6
Receive data package	0x00	0x01	0x00	0x01				
input buffer full	0x00	0x61	0x81	0xE0				
XOR error	0x00	0x01	0x01	0x00				
No excitement about the program.	0x40	0xF0	0xF0					
Answer: Write a CV	0x42	0xF2	# CV	Value	XOR			
Answer: Read CV	0x44	0xF2	# CV	Value	XOR			
Return message	0x20	0xFX	info	Data1	Data2	dataN	ADW	XOR
SW version	0x00	0x02	version	Co.Nr.	XOR			

4 examples

In the following examples, "PC:" at the beginning of the line means that this data is sent from the PC, "In:" means that the data was sent from the interface. All values are shown in hexadecimal.

4.1 Connection setup and initialization

The example shows how to initialize the interface at the beginning.

PC: 10	***At the beginning, the PC should send the confirmation***	
PC: 10	***byte 3 times***	
PC: 10		
PC: 40 f0 f0	***For safety reasons, the programming track***	
	should be switched off	
In: 00 01 00 01		
PC: 10		
PC: 21 f1 01 f0	***Set the repeat rate of the feedback to 1***	
In: 00 01 00 01		
PC: 10		
PC: 23 f2 00 00 f2	***Set info byte for feedback in Grp 0***	
In: 00 01 00 01		
PC: 10		
PC: 23 f2 01 10 e3	***Set info byte for feedback in Grp 1***	
In: 00 01 00 01		
PC: 10		
PC: 00 21 81 a0	***Everything On***	
In: 00 01 00 01		
PC: 10		
In: 00 61 01 60	***Broadcast "Alles An"***	
PC: 10		
PC: 22 f2 00 03 f1	Number of feedback devices in Grp 0 (	in this case 3)
In: 00 01 00 01		
PC: 22 f2 01 00 f3	Number of feedback devices in Grp 1 (	in this case 0)
In: 00 01 00 01		

4.2 Programming feedback devices

In this example, a feedback device is programmed to address 1 in group 0. The feedback device to be programmed must be disconnected from the RocoNet before programming. The feedback device may only be reconnected once the info bytes for both groups have been set to programming.
After the programming process, the info byte for both groups must be reset to the normal value.

PC: 23 f2 00 c1 33	***Set info byte to programming address 1 for feedback in Grp 0***
In: 00 01 00 01	
PC: 10	
PC: 23 f2 01 c1 32	***Set info byte to programming address 1 for feedback in Grp 1***
In: 00 01 00 01	
PC: 10	

Feedback sensor was plugged in and programmed

PC: 23 f2 00 00 f2	***Reset info byte for feedback in Grp 0***
In: 00 01 00 01	
PC: 10	
PC: 23 f2 01 10 e3	***Reset info byte for feedback in Grp 1***
In: 00 01 00 01	
PC: 10	
In: 20 f3 00 00 00 f3	***Cyclic feedback from the interface***
PC: 10	

4.3 Reading CV

4.3.1 Reading individual CVs

In this example, CV 29 is read.

PC: 41 f4 78 1c e8 8c f4	***Read CV 29***
In: 00 01 00 01	
PC: 10	
In: 44 f2 1c 06 e8	***CV 29 has the value 0x06***
PC: 10	
PC: 40 f0 f0	***Turn off the programming track***

4.3.2 Reading multiple CVs

In this example, CVs 1 - 5 are read.

PC: 41 f4 78 00 e8 90 f4	***Read CV 1***
In: 00 01 00 01	
PC: 10	
In: 44 f2 00 03 f1	***CV 1 has the value 0x03***
PC: 10	
PC: 41 f4 78 01 e8 91 f4	***Read CV 2***
In: 00 01 00 01	
PC: 10	
In: 44 f2 01 03 f0	***CV 2 has the value 0x03***
PC: 10	
PC: 41 f4 78 02 e8 92 f4	***Read CV 3***
In: 00 01 00 01	
PC: 10	
In: 44 f2 02 04 f4	***CV 3 has the value 0x04***
PC: 10	
PC: 41 f4 78 03 e8 93 f4	***Read CV 4***
In: 00 01 00 01	
PC: 10	
In: 44 f2 03 03 f2	***CV 4 has the value 0x03***
PC: 10	
PC: 41 f4 78 04 e8 94 f4	***Read CV 5***
In: 00 01 00 01	
PC: 10	
In: 44 f2 04 3f c9	***CV 5 has the value 0xf3***
PC: 10	
PC: 40 f0 f0	***Turn off the programming track***

4.4 Writing a CV

In this example, CV 1 and CV 2 are written.

PC: 40 f4 7c 00 03 7f f4	***Write the value 0x03 in CV 1***
In: 00 01 00 01	
PC: 10	
In: 42 f2 00 03 e8	***In CV 1 the value 0x03 was written***
PC: 10	
PC: 40 f4 7c 01 01 7c f4	***Write the value 0x01 in CV 2***
In: 00 01 00 01	
PC: 10	
In: 42 f2 01 01 f2	***In CV 2 the value 0x01 was written***
PC: 10	
PC: 40 f0 f0	***Turn off the programming track***

5 Legal, Disclaimer

The company Modelleisenbahn GmbH expressly declares that under no circumstances shall it be legally liable for the content of this document or for any additional information provided in this document.

The legal responsibility lies exclusively with the user of the provided data or the publisher of the respective additional information.

Modelleisenbahn GmbH, Plainbachstraße 4, A-5101 Bergheim, Austria, expressly assumes no liability for any damage resulting from the use of the information provided or from the non-use of the information provided.

Modelleisenbahn GmbH, Plainbachstraße 4, A-5101 Bergheim, Austria, assumes no liability for the topicality, correctness, completeness or quality of the information provided. Liability claims relating to material, immaterial or immaterial damage caused by the use or non-use of the information provided are fundamentally excluded.

Modelleisenbahn GmbH, Plainbachstraße 4, A-5101 Bergheim, Austria, reserves the right to change, add to or delete the information provided without prior notice.

All brands and trademarks mentioned within the document and possibly protected by third parties are subject without restriction to the provisions of the applicable trademark law and the ownership rights of the respective registered owners.

The copyright for published information created by Modelleisenbahn GmbH, Plainbachstraße 4, A-5101 Bergheim, Austria, remains in any case solely with Modelleisenbahn GmbH, Plainbachstraße 4, A-5101 Bergheim, Austria.

Any reproduction or use of the information provided in other electronic or printed publications is not permitted without express consent.

Should parts or individual formulations of the disclaimer not, no longer or not completely correspond to the applicable legal situation, the remaining parts of the disclaimer remain unaffected in their content and validity.