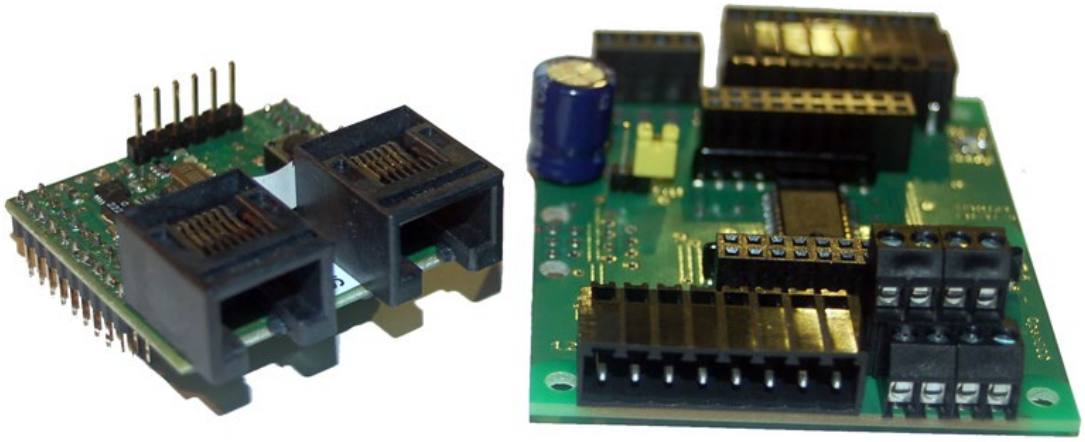
OneOC

Addon-Module for the „OneOC“ - Application



Manual

Assembly- and „How to use“ Instructions

Contents

Change log

Version	Changes	Page	Changed by	Date
v1.0	OneOC „Handbook/Manual“ (this document) written	complete	C. Schörner	03.11.2013
v1.1	Translate to English	complete	H. Falkenstein	21.04.2014
v1.2	Extended instructions for BiDiBonePlus	complete	C. Schörner	22.11.2014

Introduction

This document describes the assembly, configuration, and layout of the OneOC-plugin-device, which is a part of the DiY-series of OpenDCC and Fichtelbahn.

Please read this document carefully before assembling the kit and take notice of the security instructions.

Our products do not claim to be commercial products. This document is purely meant to be a means of help and information of how to assemble the kit for private use only.

This document was written in a thorough manner and to the best of the author's knowledge. The author neither warrants for completeness, nor up-to-datedness or correctness of this document. Wherever this document uses registered trade names or trademarked denominations all rights remain with the holder of those rights.

The author does not assume any liabilities of any kind for this instruction-document, its contents or any use of it. The user declares his agreement with this when using this document or when using the device.

The software that is used for the DIY-kit described herein can be downloaded from www.opendcc.de or www.fichtelbahn.de . It can be used freely. Feel free to change it and make it fit for your own use. You can find further details for the use of the software, DIY-kits (hardware), and hardware-applications on those webpages, too. The reader declares his full compliance with the rules and regulations that are stated on those webpages.

Commercial use of the software or parts of it is illegal and not allowed!

This document may not be used for anything else than as an assembly instruction for the BiDiBone-DIY-kit. The kits, its parts and the finished/assembled kits may not be used for anything else than the normal use. Any other use is prohibited and needs written consent of the author or the appropriate holder of rights of the webpages www.opendcc.de and www.fichtelbahn.de.

BiDiBone / BiDiBonePlus:
To increase the readability of the text, we have chosen not to use the term BiDiBone / BiDiBonePlus with each use and leave BiDiBone as the description. See the description no detailed indication of a difference, does the description or the function for both modules.

Safety Advice

The described module is an electric driven device
All necessary precautions have to be taken when using the device, also all generally necessary precaution for use of electric currents.

- Never use mains voltage with this device.
- Never use switching power supplies from PCs. Those power supplies are not ground-fee, that means there might be high voltages on the layouts, tracks and other connected devices – Danger To Life !
- Never ground-wire any conductive parts of your layout !
- All shielding and wire-shielding etc. can be put together at the same ground-free point, if regarded necessary.
- The finished module is only to be used with safety extra-low voltage and electrically protective separation.
- Model railroads are regarded as toys by law. Therefore special regulations apply.
- Only commercially available and certified powersupplies are to be used with this device.

Watch out for the appropriate certification when you buy a power supply. You can find more information about that on *www.vde.de*.

Specified normal operation:

The module is only to be used for digitally operated modeltrain layouts. This module is only to be used for switching, maneuvering and feedback.

Any other use is not a specified normal operation and is therefore not allowed.

The device is not meant to be assembled by, mounted by, nor operated by children under the age of 14 years.

Tools and Supplies of Work

You will need:

- solder 0,5 oder 0,3 mm in diameter
- flux melting agent in some cases
- cleaning solution, brushes, 100% Isopropanol
- illuminated magnifier, a microscope can be better
- soldering iron 30 Watt, or better a thermally regulated soldering iron station

1. General Information on OneOC

1.1 Description

The **OneOC** is a Base-Module (Base-Board). Look at it like a motherboard in your PC. OneOC needs a BiDiBone for operation. So the BiDiBone is like the processor in a PC. The OneOC-hardware-board is connected to the BiDiBone, then the appropriate application (firmware) is loaded onto the BiDiBone. That gives the BiDiBone-OneOC-combo all it needs to function. Just to make it clear: The base-module-board we are talking about in this manual and the corresponding firmware to be loaded onto the connected BiDiBone are both called OneOC.

Please use this manual as a comprehensive guide for assembly and use of the OneOC hardware module.

You can order the BiDiBone-module as SMD-premounted-DIY-kit or in a set with the OneOC-DIY-kit. The OneOC-Kit is offered in the Fichtelbahn-webshop and contains all special parts needed.(We translate “special parts” as all parts that are not part of the Reichelt-shopping-cart that is also shown in the Fichtelbahn-shop)

All standard parts can be ordered at the webshop of Reichelt. You can find a pre-filled shopping cart for that in the Fichtelbahn-webshop.

This document does not show you how to assemble the BiDiBone. You can find an assembly instruction for it on our Fichtelbahn webpage.

For the OneOC the BiDiBone and BiDiBonePlus can be used. It work both assemblies without a functional restriction.

1.2 Characteristics of OneOC

- connection to BiDiBone via two 20pin pin strips, 2,54mm grid
- Display port via a 6pin pin strip 2,54mm grid or RJ45 connector
- 20x galvanically isolated inputs (optocoupler) with common ground
- 1x control LED for status changes
- Supplied with 12V DC
- size: 59mm x 69mm

1.3 Wiring diagram, lay-out, and list of parts

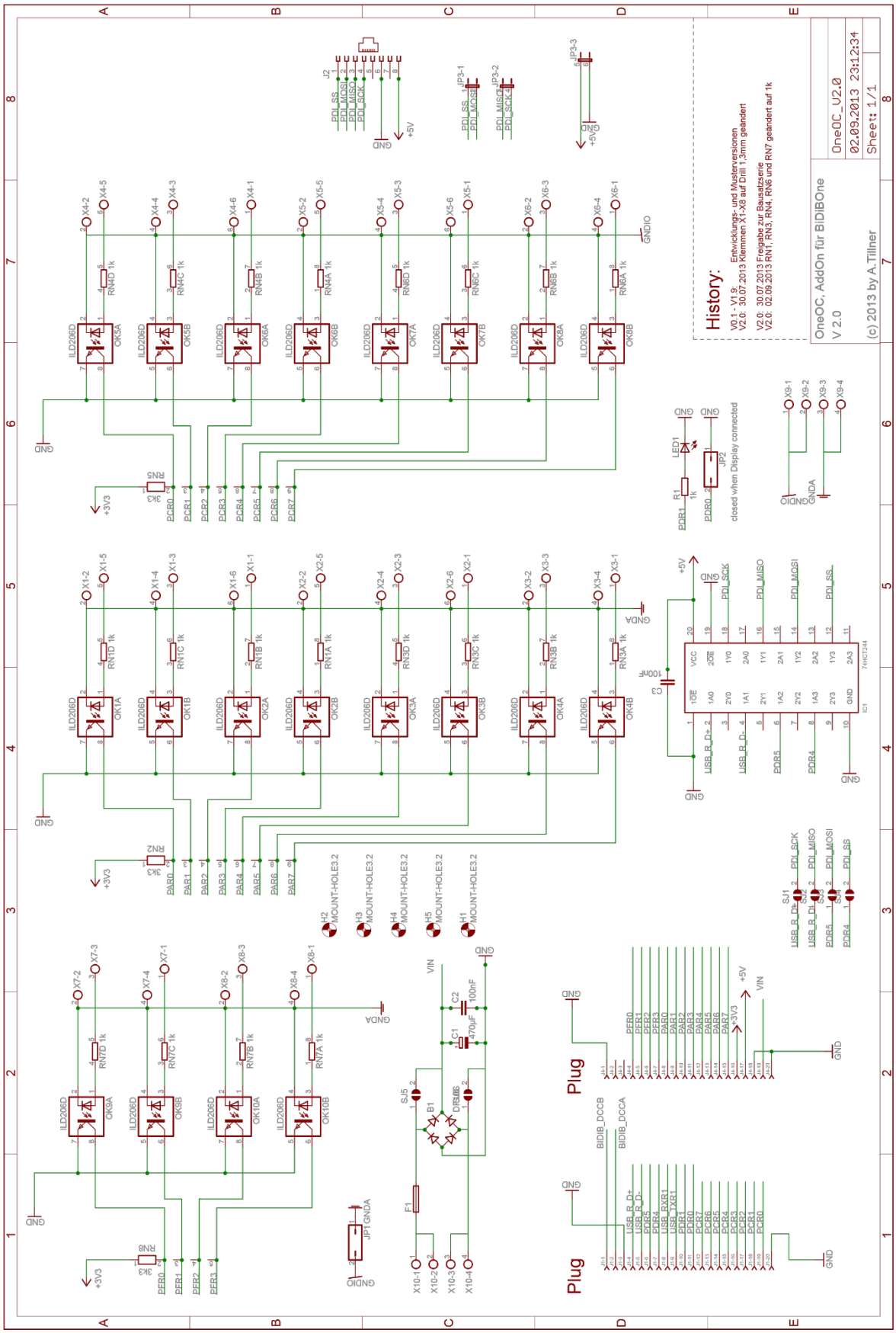


Figure 1: Wiring Diagram OneOC

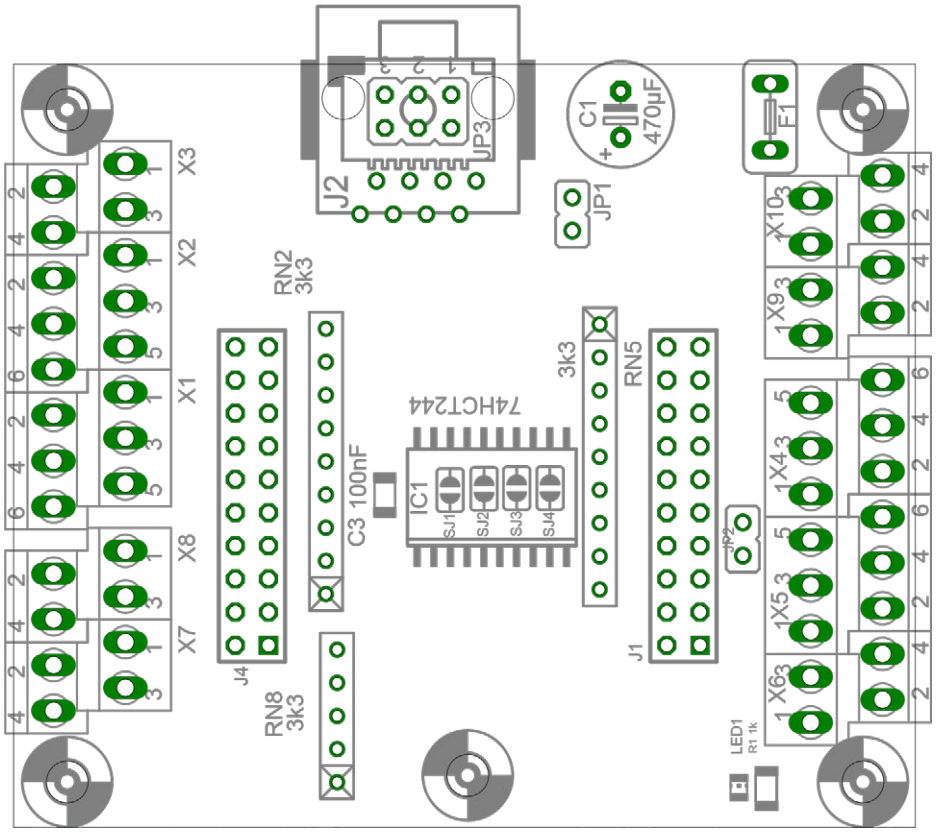


Figure 2: Layout TOP OneOC

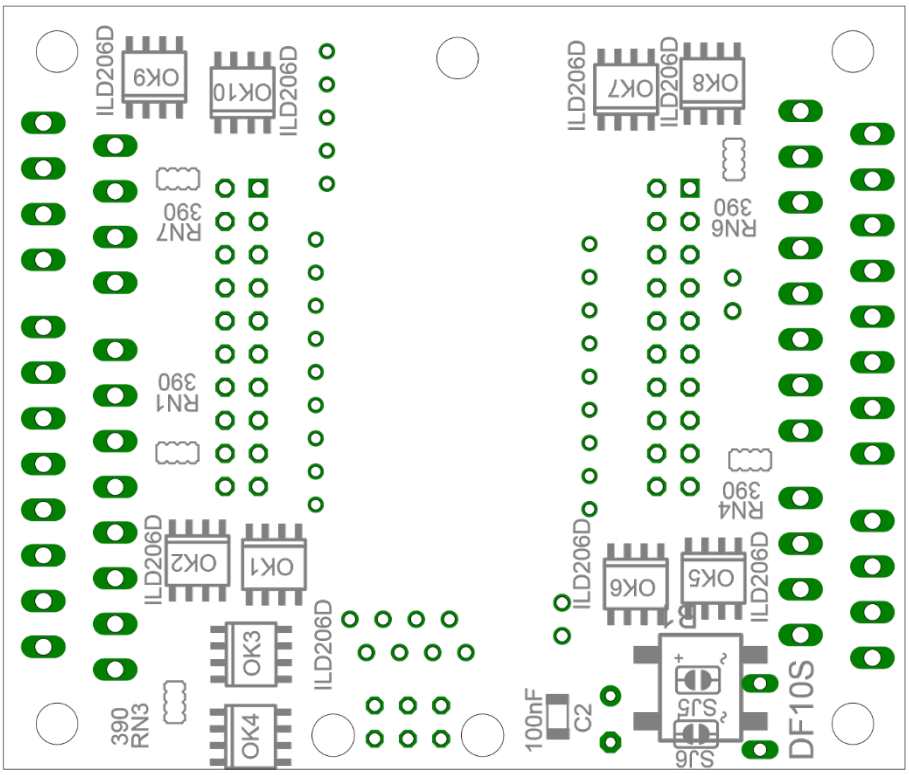


Figure 3: Layout Bottom OneOC

You can find high-res PDF-files for download on the Fichtelbahn.de webpage of:

- Wiring diagram
- Layout
- List of Parts

2. Assembling the OneOC

You will need the list of parts and the layout-print. Please download both from from the Fichtelbahn website. The Layout is part of the wiring plan found there.

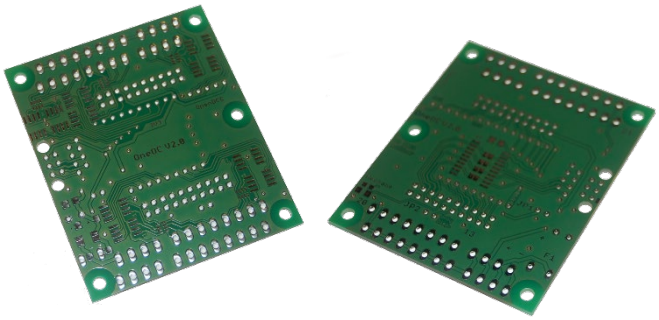
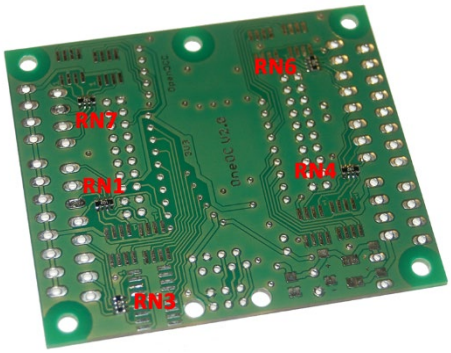


Figure 4 shows the front and back of the OneOC circuit board for correct orientation.

Figure 4: OneOC circuit boards



Assembly, step 1:

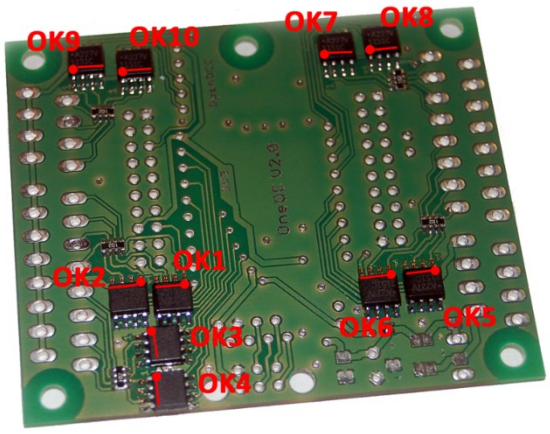
Let's start with the most difficult parts of this kit, the parallel resistors **RN1, RN3, RN4, RN6, and RN7.**

Figure 5:Mounting the resistors

Check what you have done:

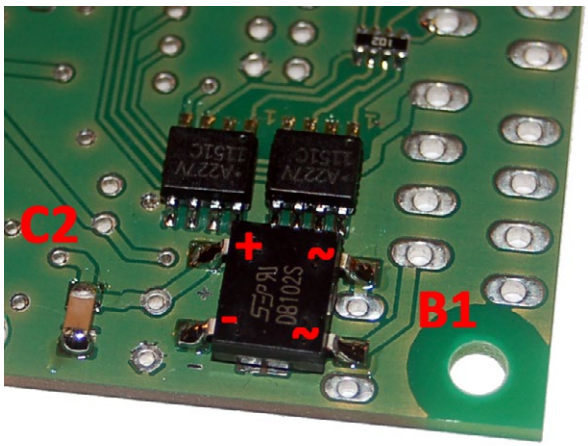
Please check the correct placement with a magnifying glass or a multimeter. Use the last and next pad for a multimeter check. The ohmic resistance should show 1kOhm.

Assembly, step 2:



Now solder the ten optocouplers to the back-side (underside) of the board. Watch out for correct placement of parts **OK1-OK10**.
Pin1 is marked with a dot or a dash. This mark can be found on the optocouplers as well.

Figure 6: Mounting the optocouplers



Assembly, step **3:**

Now it is time to mount the commutator for the 12VDC to the board.

Watch out for correct placement of the commutator **B1, Figure 7** shows the correct polarity. The blocking capacitor **C2** is mounted next to the commutator.

Figure 7: 12V power supply

Display-function (optional) **(not yet available and not a part of the kit):**

Option: Solder another blocking capacitor C3 to the and a level switcher IC1 to the board. Watch out for the pin1-mark on the IC1.

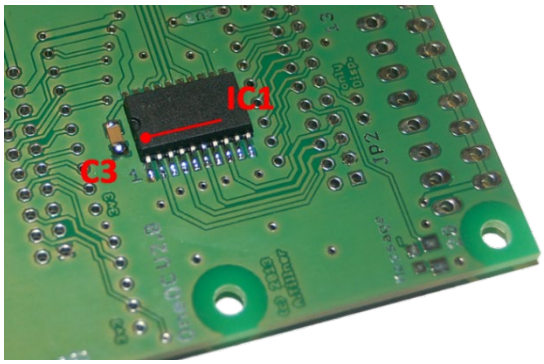


Figure 8: Level-converterfor display (optional)

Assembly, step 4:

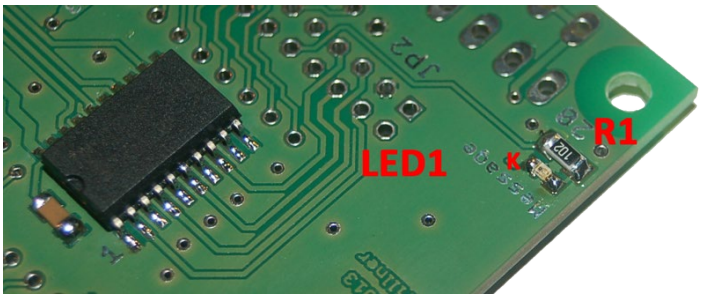


Figure 9: message-LED

Any change of inputs is shown by flashing of the **LED1. Watch for correct placement again.** The cathode of **LED1** marked with a „K“ on the corresponding figure Resistor **R1 is the pre-resistor for the** Message LED.

Assembly, step 5:

Mount the 8x network resistors **RN2** and **RN5** as well as the 4x **RN8** to the front of the circuit board.

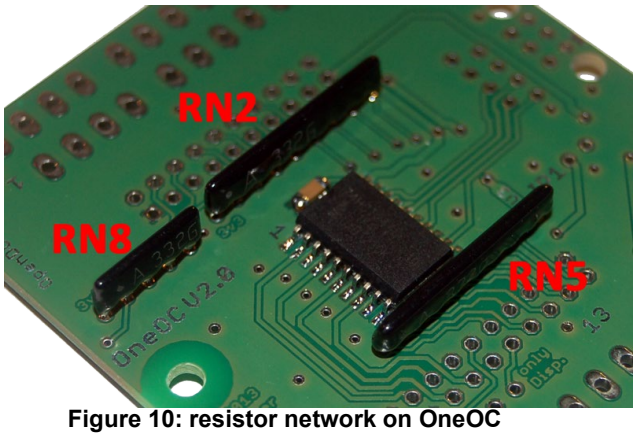
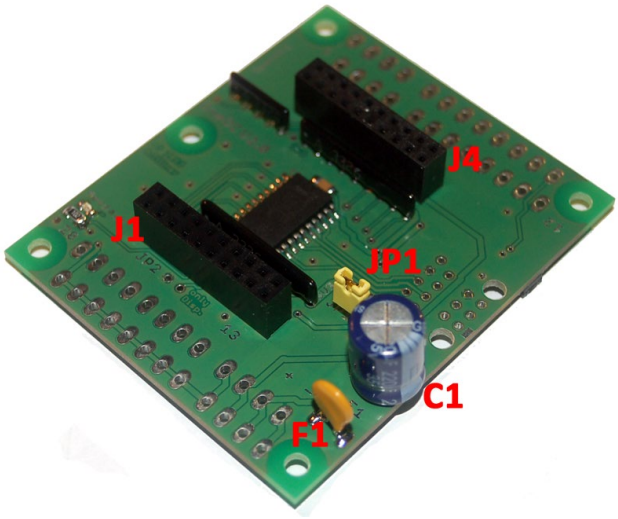


Figure 10: resistor network on OneOC



Assembly, step 6:

As a last step, mount fuse **F1**, cpapcitor **C1**, pinstrips **J1** and **J4** (for BiDiBone) and jumper **JP1**.

Figure 11: pin strips for BiDiBone

3. Connecting Terminal of OneOC

OneOC can be assembled with two different concepts of connection.
See page 14 for a little help to choose the one you like or need.

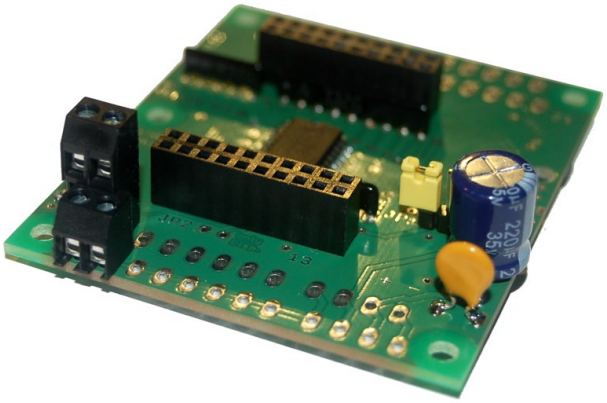


Figure 12: OneOC with double-elevated connecting terminal

Concept A:

The 20 ground-based inputs 0 – 19 and the 12V input use the **double-elevated connecting terminal.**

With the help of this connecting terminal every input can be controlled via its ground. The circuit board is used as ground busbar and ground potential.

Concept B:

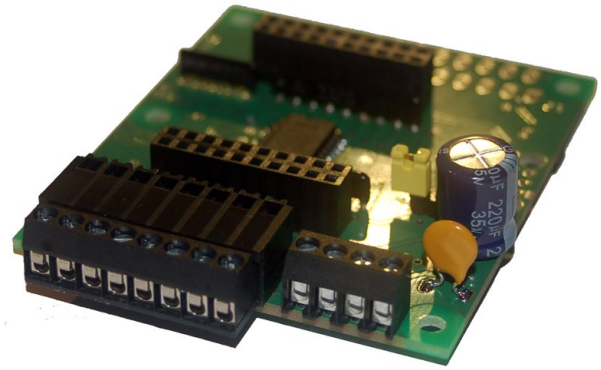


Figure 13: OneOC with RiaPlug „to the side“

With concept B we use the **RiaPlug connector „to the side**“. The supply voltage and the common ground is realized via a **4pin screw connector**.
With tis concept we only connect the input of the module. The external common ground busbar is connected to the 4pin screw connector. (an external ground busbar is needed)

The OneOC module is capable of a combination of both concepts.

You have succesfully finished the assembly.

4. Connector terminal assignment of OneOC

4.1 OneOC Terminal Assignment:

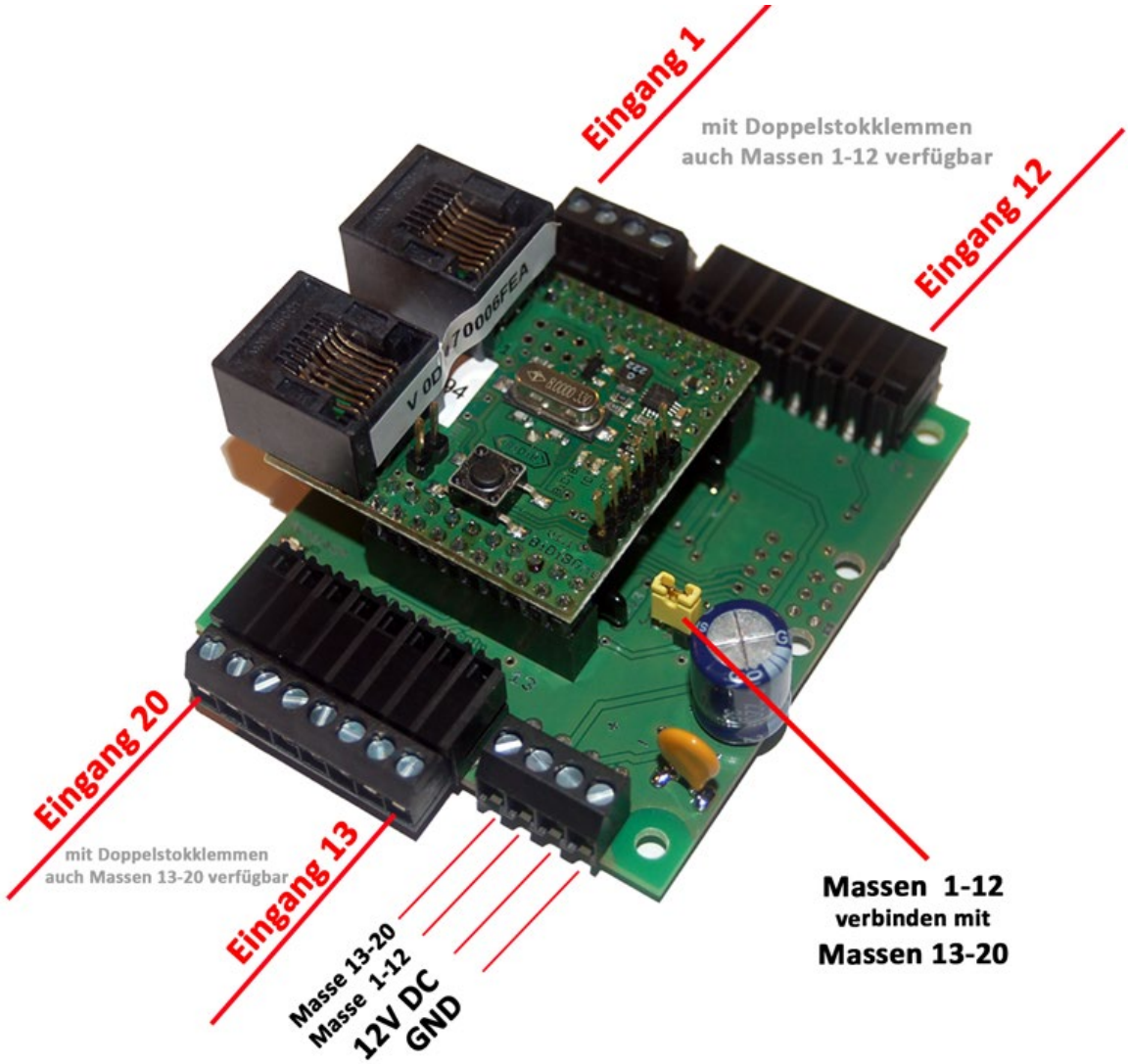


Abbildung 14: OneOC Connector assignment

The OneOC needs an external 12V power to supply the BiDiBone.

5. OneOC Status-LEDs

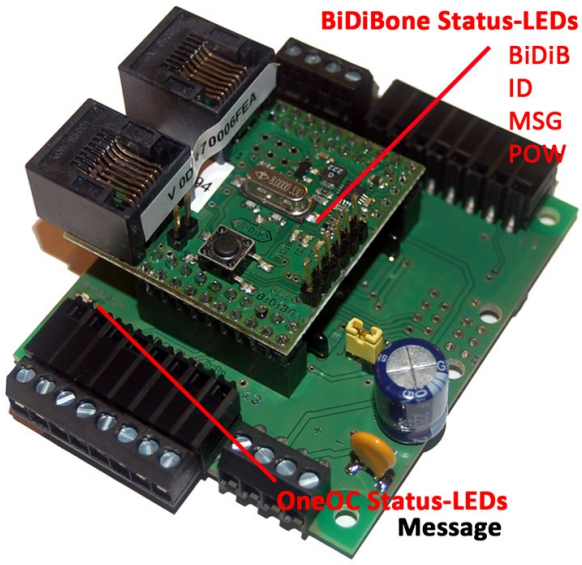


Figure 15: OneOC Status-LEDs

5.1 BiDiBone Status-LEDs:

BiDiB
ID
MSG
POW

Zustand der LED	Bemerkung
BiDiB LED lights	Connection to the BiDiBus established
BiDiB blinks shortly	Data is received and sent
BiDiB LED off	No connection to the BiDiBus
ID LED blinks fast	Identifying (via software or button)
MSG LED	--- no function ----
POW LED flickers	BiDiBone is connected to power
BiDiB, ID, MSG u. POW flicker	No eeprom-file present
BiDiB and POW flicker	No serial number present

5.2 OneOC Status-LEDs:

Message

Zustand der LED	Bemerkung
Message LED flickers	Status change on one of the inputs

6. Implementing the OneOC

Step 1:

Figure 16 shows the BiDiBone mounted to the base-board OneOC. Mind the correct alignment of the two modules, especially that the BiDiBone is not misplaced for one pin row.

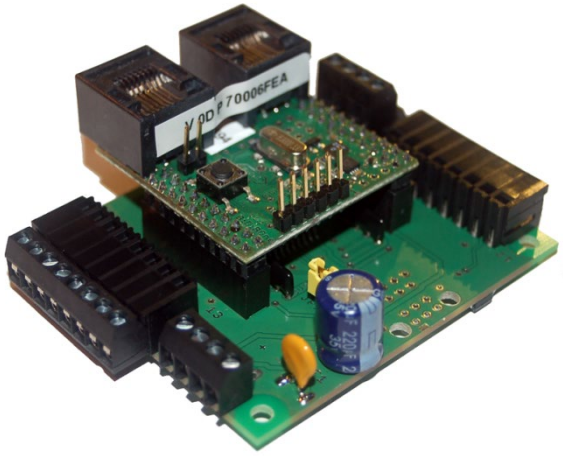


Figure 16: OneOC with BiDiBone

Step 2:

Now we connect the base-module OneOC to 12VDC power. **figure 17** shows the correct polarity.

Power consumption rises to around 20mA.

The **POW-LED on the BiDiBone** lights up (page 15)

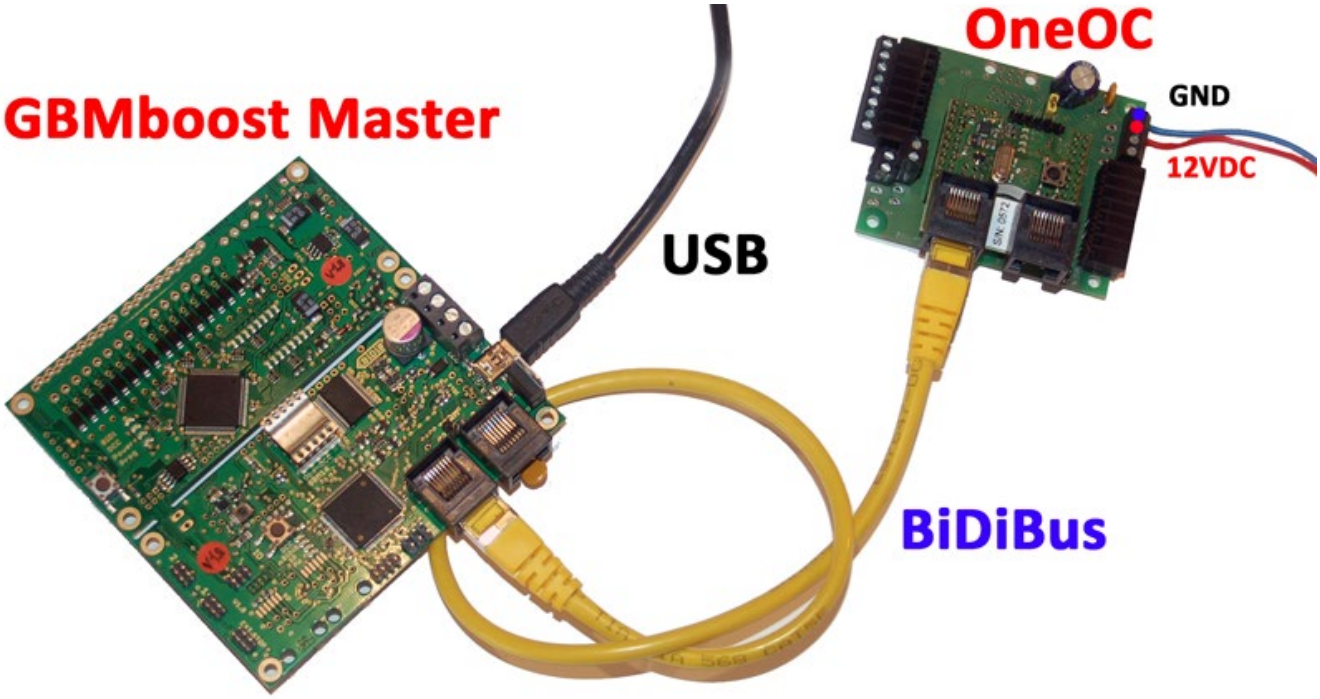


figure 17: Connection to the BiDiBus

Step 3:

If both are true (the power rises to around 20mA and the BiDiBone POW-led lights up) you may connect the GMBboost with your PC. Then connect the BiDiBone to the GBMbost with an RJ45 wire.

We will refer to the BiDiB-Monitor software from now on in this documentation

All following steps can be done with any hostprogram that supports the update functionality of BiDiB (like BiDiB-Wizard)

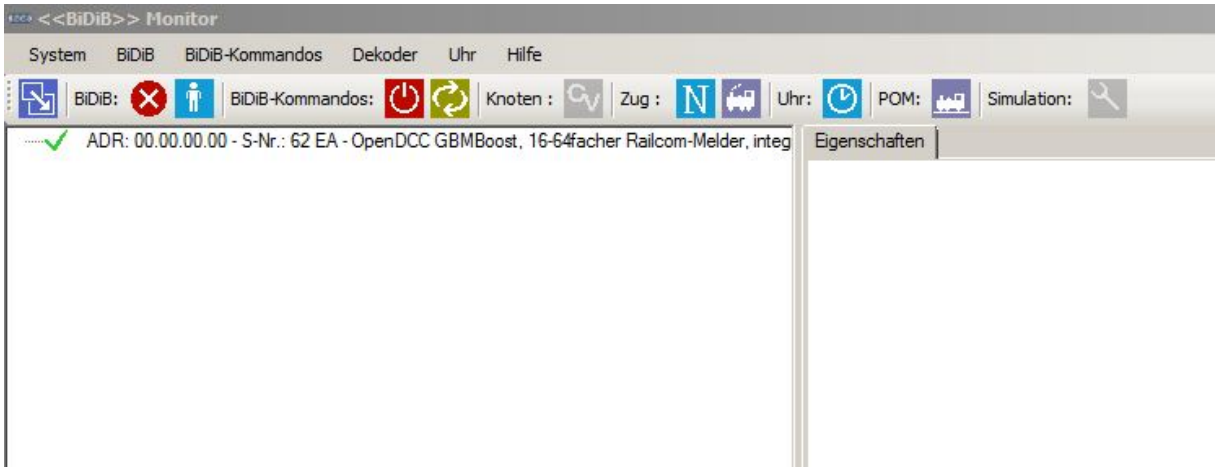


Figure 18: BiDiB-Monitor recognizes the GBMboost Master

The list of nodes shows the GBMboost Master. (**Figure 18**). The BiDiBone is not shown in this list yet. The BiDiBone needs a confirmation that it may connect to the BiDiBus**. For that, please press the button on the BiDiBone!**

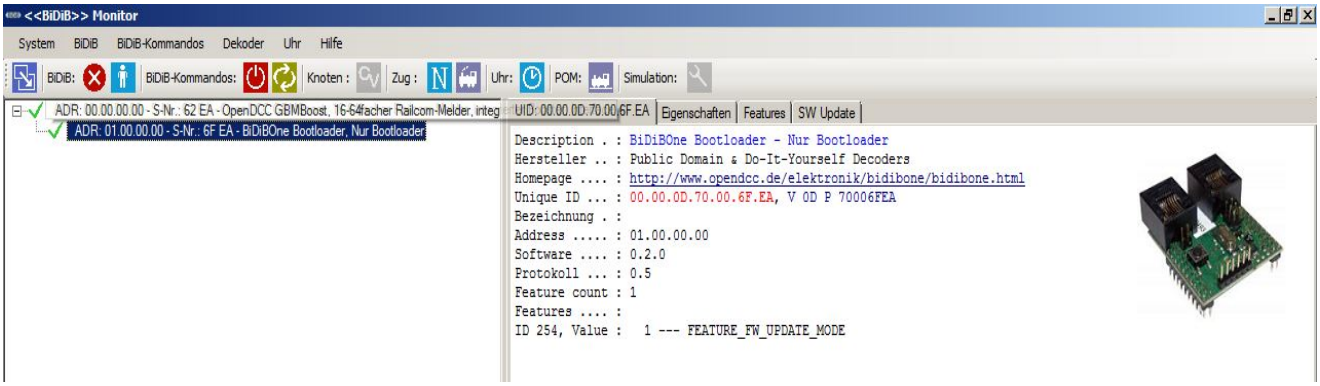


Figure 19: BiDiBone is showing up in the list

The SMD-premounted BiDiBone comes with a bootloader installed. With this it is able to connect to the BiDiBus. After you pressed the button it is recognized by the hostprogram as “BiDiBone with bootloader only”.

This means your assembly of BiDiBone and OneOC was successful!

Step 4:

Now it is time to download the application firmware. You can find it on the Fichtelbahn site. This firmware is loaded onto the BiDiBone via the BiDiBus turns the “BiDiBone with bootloader only” into “OneOC” in the node list.

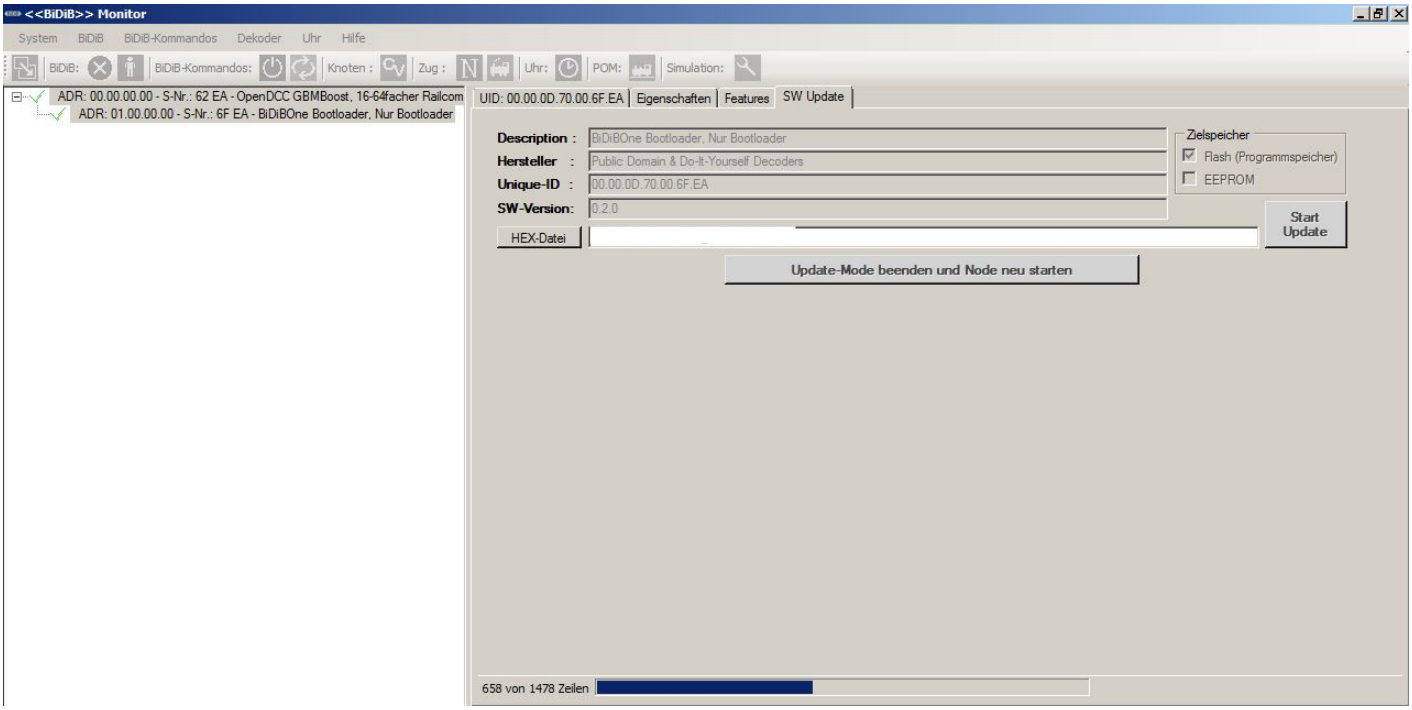


Figure 19: BiDiB-Monitorruns a firmware update on BiDiBone

In the node list click on “BiDiBone bootloader only” and open the tab “SW Update” in the right window. Follow the instructions of the update tool.

Start the firmware update with the file *.000.hex and carry on with file *.001.hex.

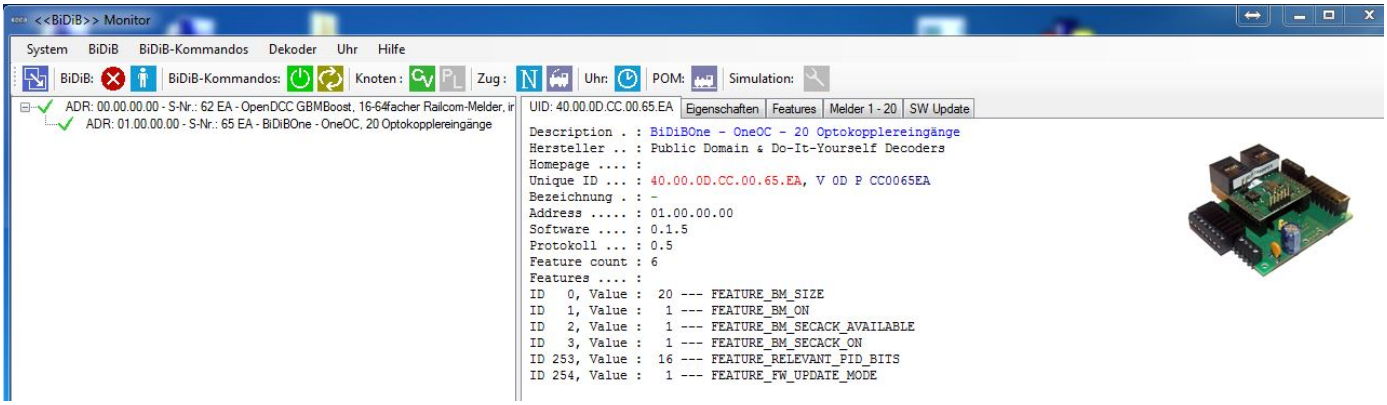


Figure 20: BiDiB-Monitor with OneOC

After the firmware update has finished the tool reboots the nodes and recognizes the new node „BiDiBone OneOC“

So now the OneOC has started to work properly.

7. Applying the OneOC

20x Optocoupler inputs

The node OneOC provides 20 inputs with ground connection for feedback applications.

Figure 14 shows the pin assignment of the 20 ground connected inputs.

Reed-contacts, switches , push-buttons, Hall-sensors or photoelectric relays can give feedback to a hostprogram.

Inputs 1 to 12 and **Inputs13 to 20** have a separat ground bar and therefore can be connected to two different potential.

Both grounds can also be connected by jumper **JP1**.

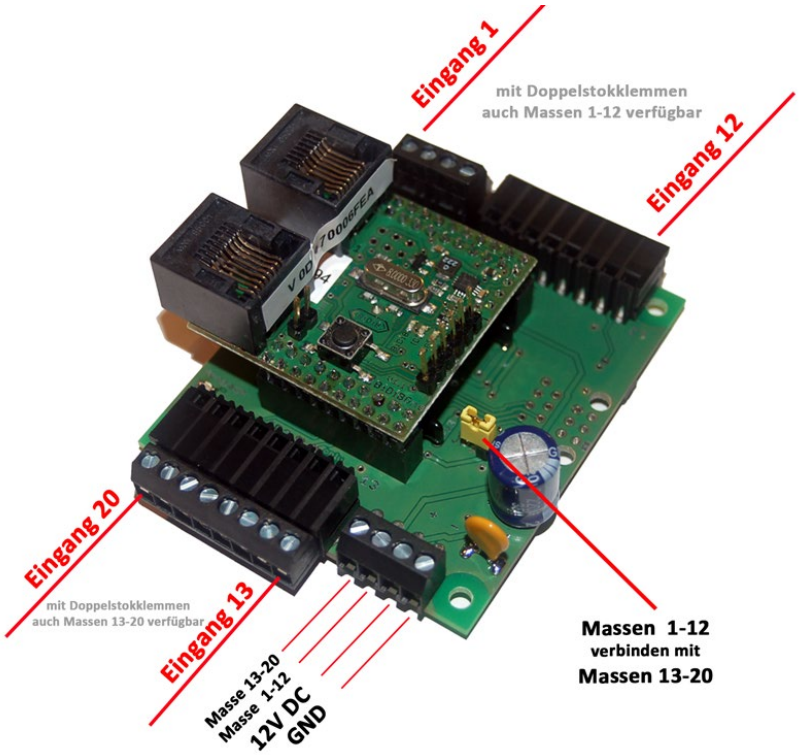


Figure 21: Inputs on OneOC

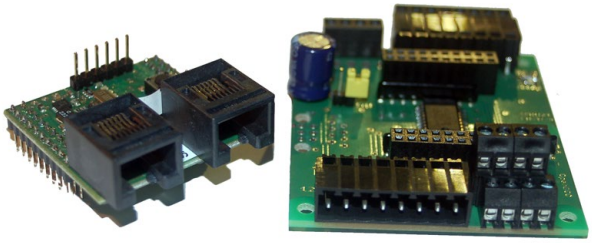
Figure 22: Inputs shown in BiDiB-Monitor

The inputs are captured and shown as normal occupied signals in the hostprogram. (**Figure 22**)
The inputs of the OneOC are attributed via its Unique-ID. This Unique-ID is allocated to an element in the PC-control-program.

Further information about Unique-ID is provided in chapter 9.

8. Further Information about the module

All following advice is meant for an update or change of the firmware!



8.1 Firmware Update with Programmer on BiDiBone

This chapter explains how to do a firmware update with the help of a programmer. A description of the recommended procedure is listed on page 16. Using a programmer in BiDiBone is an alternative to that.

The ATXmega is programmed with PDI, which is a 2-wire connector.
The SPI-Adapter used before (like ponyprog) cannot be used.

You can use:

- **AVRISPmkII**: in order to use that you need an up-to-date of AVR Studio. Update the firmware of the AVRISP before proceding.
- **STK600**: The STK600 is connected from its 6pin blue PDI-plug 1:1 to our board. Important Notice: VTARGET-jumper on the STK600 must be opened!
- **JTAGICE mkII** and JTAGICE **mkII-CN**
  Connect AVR JTAGICE mkII Data (PDI) with the JTAG Pin 9 .
  JTAGICE mkII-CN (Clone) Data (PDI) is connected to JTAG Pin 3.

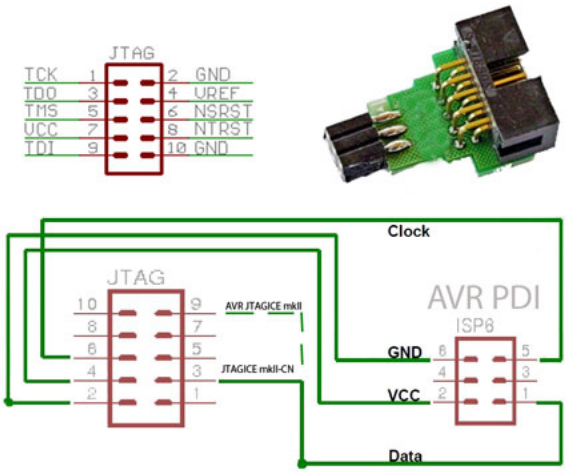


Figure 23: PDI / ISP interface

The **DIAMEX ALL AVR:**

The **DIAMEX ALL AVR** is a reasonably priced programmer for the entire BiDiB-project. All our ATX processors can be programmed via its PDI interface.

Buy it at Reichelt - **Part-Nr.: DIAMEX ALL AVR**

That prgrammer bears many advantages because it covers many appliances and can supply power to the assembly to be programmed.

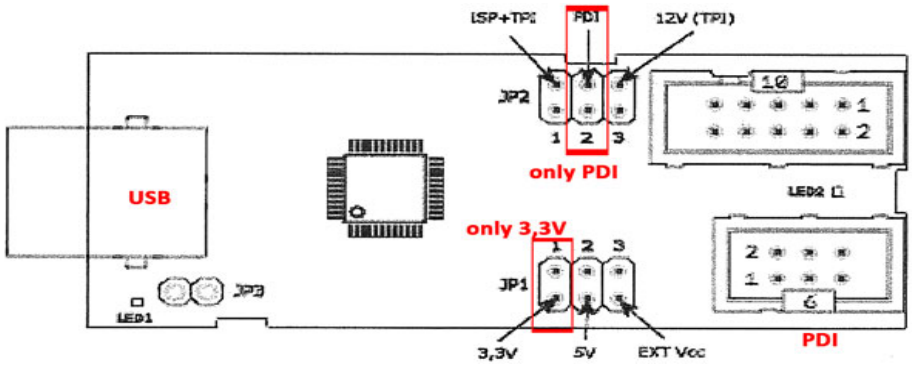


Figure 24: Diamex ALL AVR Programmer

It is essential to check the jumper settings before programming to prevent destruction of both programmer and processor.
There must only be ONE jumper on PDI and 3.3V. External power supply is not recommended.

Very Important:

Don't place the jumper on 5V. This would kill the processor. Its operation voltage is 3.3V!

Place the PDI-plug on the unmounted PDI-interface oft he BiDiBone. A longer pin strip helps you establish a connection.

The **DIAMEX ALL AVR shows up in** AVR Studio as **Device AVRISP**.

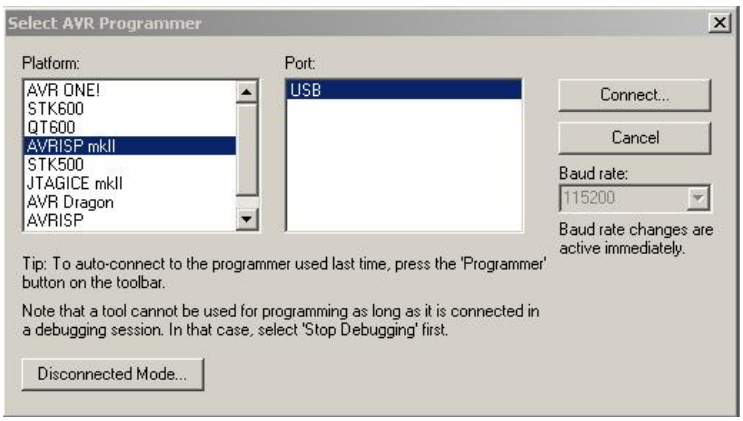


Figure 25: AVR Studio – Programmer choice

Start the AVR Studio and establish connection. After you connected and logged in successfully the screen should look like this:

Chose ATXmega128D3 and read out the signatur of the ATXmega128D3:

0x1E 0x97 0x48.

Programming mode hast o be set to PDI

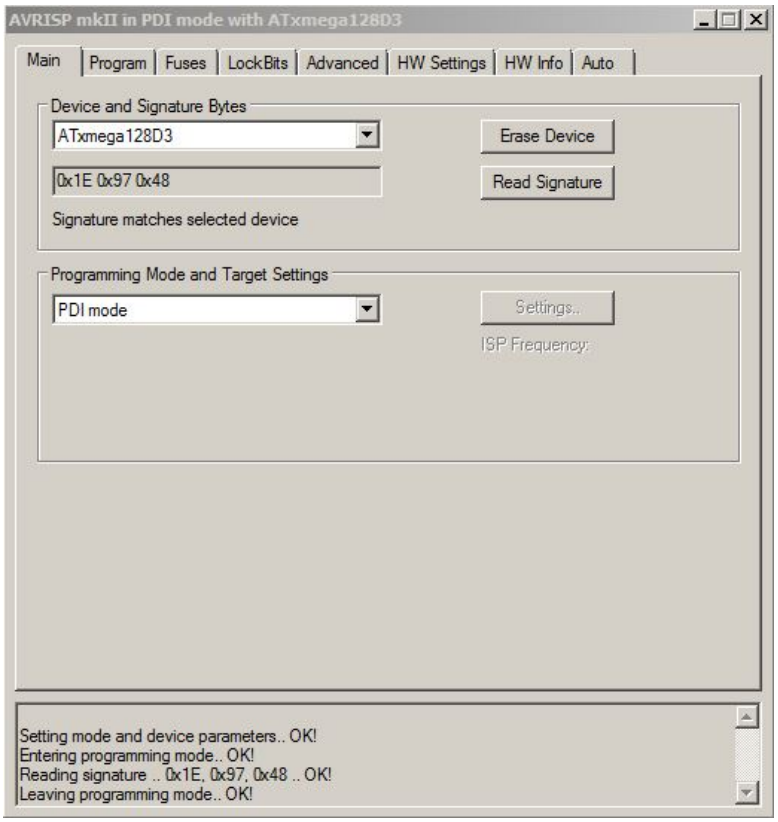
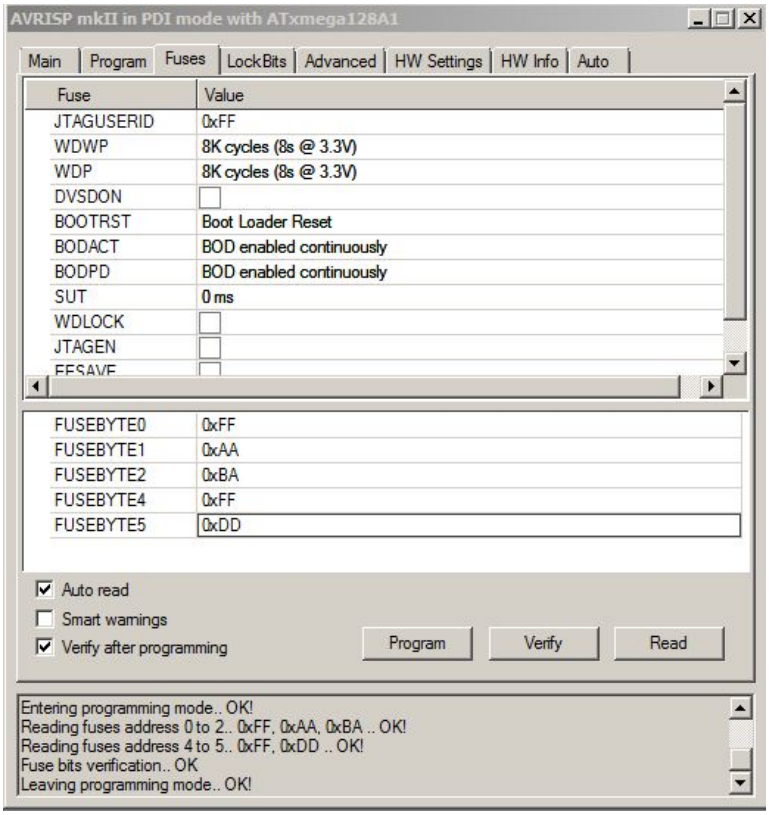


Figure 26: AVR Studio – Main Window



Set the Fuses. Click on the „FUSES“ tab. This is used to choose the work mode of the microcontroller. Be careful with the settings.

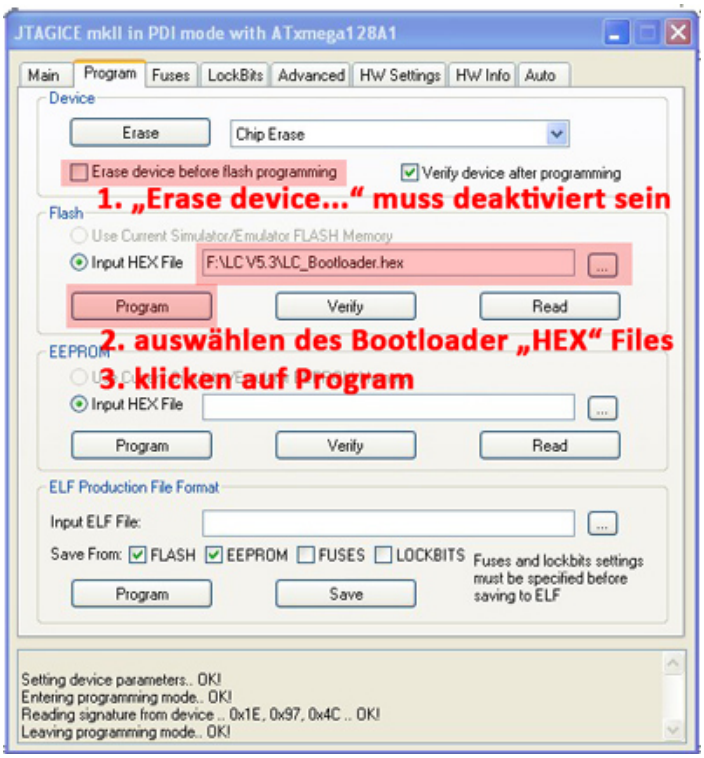
Hidden Settings:
BODLEVEL: 2,1 Volt

Summary:

FuseByte 0: 0xFF
FuseByte 1: 0xAA
FuseByte 2: 0xBA
FuseByte 4: 0xFF
FuseByte 5: 0xED

Figure 27: AVR Studio – Fuse Window

If you did set everything correctly, AVR Studio returns „OK“!



The checkbox **"Erase device before flash programming"** must be deactivated to prevent the deletion of the former bootloader.

Now choose the **Bootloader-HEX** File from your download folder and click **"Program"** to load the bootloader to the BiDiBone

Checking (optional):
You could stop the programming procedure, restart the BiDiBone and press the push-button to start the bootloader.

Figure 28: AVR Studio – Program Window Part 1

Now we need the firmware itself, files ***000 .hex** and ***001.hex**.

The file *000.hex belongs to path "Flash".
The file *001.hex to path "EEPROM".

When you click “**Program**” the firmware is sent to the microcontroller.

Flash and **EEPROM** are done separately. Both files are NOT sent automatically.

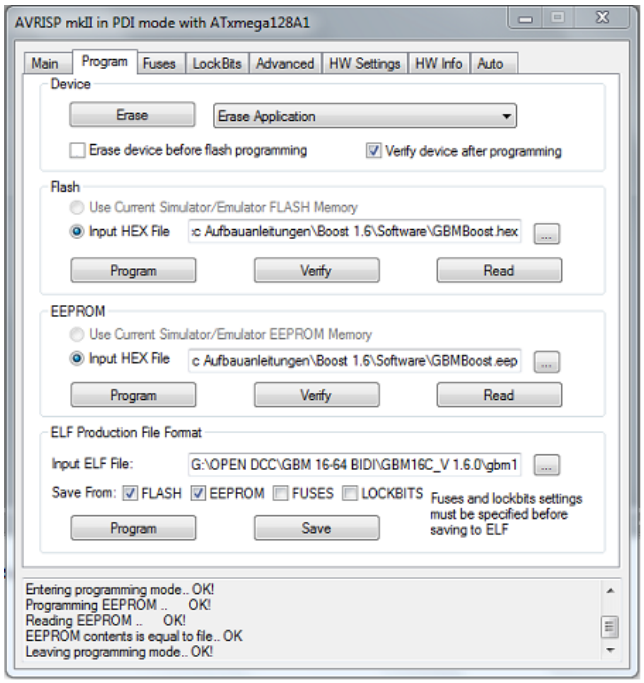


Figure 29: AVR Studio – Program Window Part 2

The following procedure is only necessary if the serial number has been deleted from USER SIGNATURE AREA:

Finally we have to create a serial number with the free BiDiB serial number creator on ***http://www.opendcc.de/elektronik/bidib/opendcc_bidib.html***

You need to register in the OpenDCC forum. With chosing the suitable product abd a remark a fitting serial number file can be created. Save the file to your pc with a click on the colored ".eep".

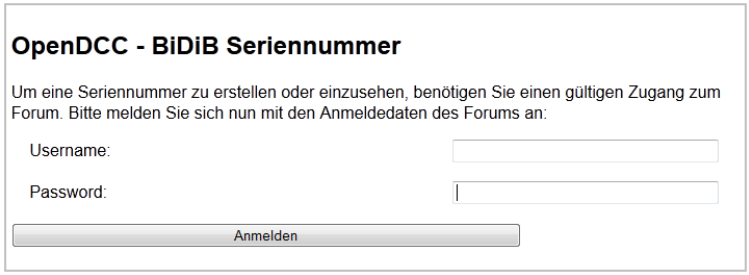


Figure 30: BiDiB Serialnumber Creator

The serial number is an eeprom-file or a **serial_000.hex file**. It has to be selected in EEPROM and transferred to the BiDiBone. Do this with clicking on "**Program**".

Don't forget to note the serial number on the module.

9. The Unique-ID

Figure 32 shows a **Unique-ID** sticker on a BiDiBus Device.
All SMD-premounted kits purchased on Fichtelbahn-Shop and are part of a BiDiBus are all delivered with their Serial Number programmed into them. This serial number is a part of the Unique ID. Therefore the complete Unique ID is printed to the back of the kit's board.

Figure 32: Unique ID on the board

What is a Unique-ID?
The Unique Id is a combination of the 16bit manufacturer's ID and a 32bit manufacturer-specific number, like product-index or serial-number.

V = VID (ManufacturerID)
OD = DIY-project
P = PID (ProductID)
6800 = GBMboost Master
6700 = GBMboost Node
6B00 = LightControl
0029 turns into **2900** = serial number

Unique-ID is provided in HEX-format.

How to use Unique-ID
As the name says, Unique-ID is an absolutely unique number. Through this number the module can be identified, no matter where it is mounted or where it is placed within the bus. In simple words: The Unique ID is like a phone number in a phone book. The hostprogramm provides the names for the individual „extensions". So UniqueId is the link between the label of the model on a PC and the module itself.

Example:
A BiDiBone is placed under a train station on your layout and carries the Unique-ID **0D 6B001234**.
In the hostprogram it is labeled „Station West". The BiDiB-system tells the hostprogram that 0D6B001234 can be reached under extension 3. So if anything changes in Station West, the hostprogramm knows it has to call extension 3. There you go !
So no more fiddling with addresses or dip-switches.!

Why should i use the UniqueId sticker?
What is the use of this piece of information?

All BiDiB-tools and up-to-date PC-controlprograms communicate with and control all modules via this number. There is no more “DCC-Address X” kind of thing anymore, all commands will be sent to the node’s UniqueID and its ports.
That means, coming from the other end of the chain, the UniqueID of the BiDiB-module needs to be published to the control-program in order to configure a new action at the output of a decoder (like LightControl) or the allocation of a detector in your layout (like GBM). .

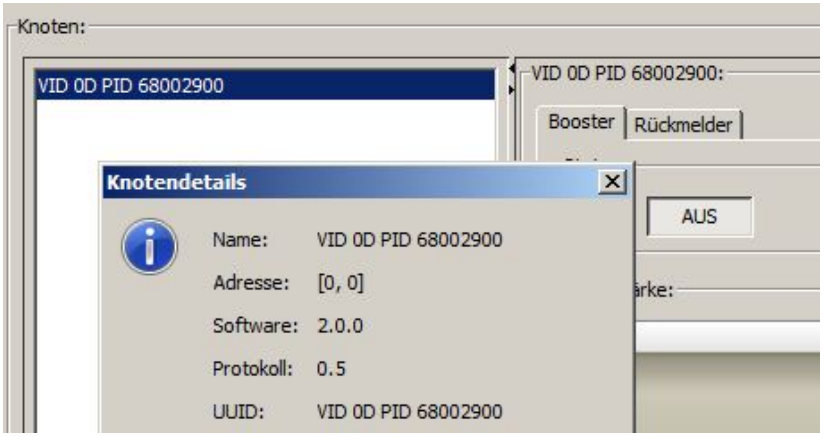


Figure 33: Unique ID im BiDiB-Tool

Hardware-Serialnumber:
The sticker on the back of the board contains the serialnumber (S/N) of the hardware and has nothing to do with its UniqueID. The S/N is used for registering the module with our internal and shop system
.

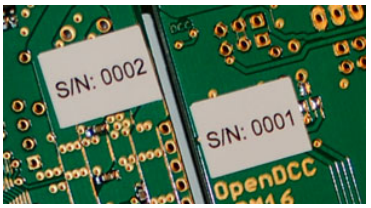


Figure 31: Hardware S/N on module

Where is the UniqueID stored on the module itself ?

This is split into two parts actually. The VID and PID are already programmed to the Flash/EEPROM-firmware and is sent to the module, yet it does not include the serial number. That means the firmware can be loaded onto the processor at will as well as updating it. If a serial-number is already present (in case of a firmware update), it will not be overwritten by an update. If the module does not detect a serial number at start, LEDs will blink and the module won’t work. **The firmware does not work without a serial number !**

(Exception: GBMboost as Master: A temporary or „emergency“ serial number is created with identifier 0100. It can easily be updated with a valid s/n at any time)

The serial number is loaded to the device as the third step of the process to put the device into use. It will be put on the EEPROM first.

At startup every module runs the following tests:
The user signature area is scanned for a serial number. If there is none, the serial number form the EEPROM is stored in the user signature area and the module starts working. If there already is a serial number stored in the user signature area, the one stored in EEPROM will be discarded and the s/n from the user signature area will be used.

The s/n stored in the user signature area will „survive“ a CHIP ERASE. It can only be deleted through a USER Signature ERASE.
In case you want to exchange the s/n on a BiDiBus-device you must do a USER Signature ERASE first. But keep in mind that all calibration values will be lost, because they are stored in the same area.
So a USER Signature ERASE should only be done for a very good reason !

I accidentially erased the serial number. How can I get it installed again ?

1. NEW serial number: Get a new serial number from the serial-number-generator and program it into the BiDiBus-device after deleting the USER signature area.
Important: Do not forget to replace the obsolete UniqueID on the sticker with the new one.
The s/n generator is located at http://www.opendcc.de/elektronik/bidib/opendcc_bidib.html

2. I like to use the „old“ UniqueID on the sticker again: In order to do that you need to contact Fichtelbahn-support at support@fichtelbahn.de . Do not forget to provide the **Unique-ID from the sticker and the corresponding hardware-serial number**.
Beware: This only works for SMD-premounted kits. If you created your Unique-ID yourself (for the non-SMD-premounted kits) you need to search the generator’s history for the correct number !

10. Terminating the BiDiBus

The **BiDiBus** is established through a RS485-2 wire connection that was especially engineered for high-speed and long-range data transfer. Its use is becoming more and more common in professional industrial applications. Wires can be stretched over a distance of more than 500 metres with high data transfer rates.

To guaranty both speed and data accuracy for those great distances it is necessary to terminate the bus. We call it “terminating the communications bus”. (RT1 in right figure)

If the length of the cable is below 5 metres no termination is necessary.

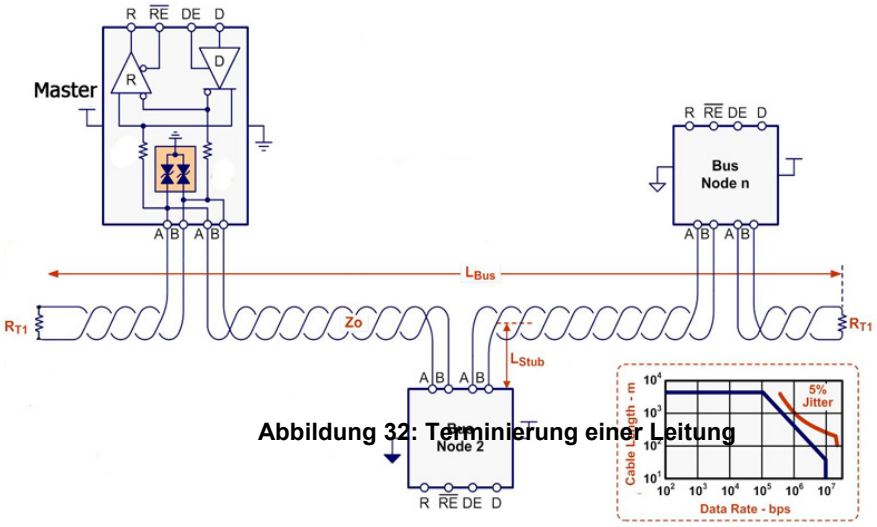


Abbildung 32: Terminierung einer Leitung

10.1 How to terminate the bus

The BiDiBus hast o be terminated at both ends with a 120 Ohm resistor. That means that the nodes at all ends of the bus have to be terminated, even with branched buses. That is done by placing a resistor between **BiDiB_A** and **BiDiB_B.**

All our BiDiB-nodes (GBM, LightControl, BiDiBone, s88-BiDiB-Interface....) are prepared for that to make it easier. You just have to place the termination jumper

Attention:
The jumper must only be set at the last nodes at the ends of the bus. In case you branched the BiDiBus with OneHub and generated dead-ends those have to be terminated, too.

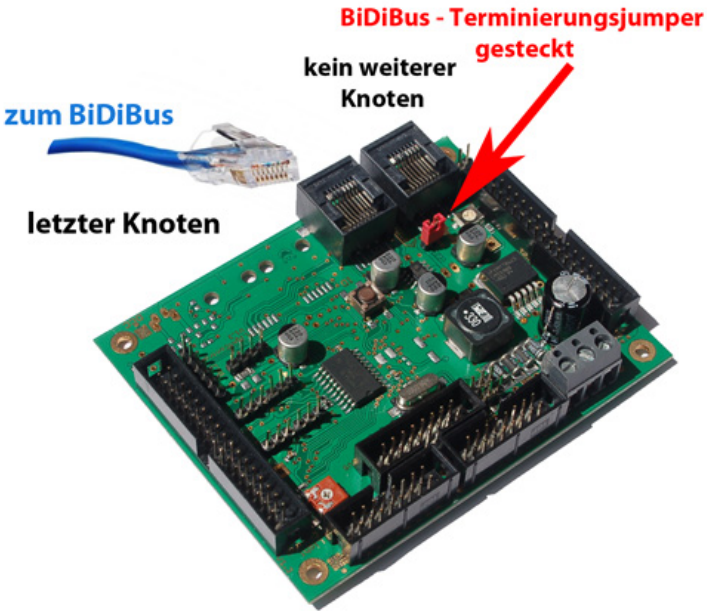


Figure 33: Terminating of a module

We higly appreciate your input. If you have suggestions on how to improve the document or should you have found any errors please let us know.

There is no warranty on this instruction and software nor do we guaranty the function of any module.We will not take any responsibility for any damage that the user or any third party inflicts or suffers by or from the use of the software or hardware. Under no circumstances can we held responsible for loss of sales or profit or any other financial or property loss during use or caused by the use of software or hardware.

The support-forum is a great source of information in case you have question
(www.opendcc.de/forum)

fichtelbahn.de
Christoph Schörner
Ahornstraße 7
D-91245 Simmelsdorf

support@fichtelbahn.de