

# Handbuch „Inbetriebnahme“

## SMD bestückter Bausatz **OpenDCC GBM**

**Variante 1 – als Rückmeldesystem**

GBMboostV1.6  
GBMboostV1.8  
GBMboostV1.9

Bausatz geeignet für:



# 1. Inhaltsverzeichnis

<b>1.</b>	<b>Inhaltsverzeichnis .....</b>	<b>2</b>
<b>2.</b>	<b>Änderungsverzeichnis .....</b>	<b>4</b>
<b>3.</b>	<b>Hinweis .....</b>	<b>5</b>
<b>4.</b>	<b>Sicherheitshinweise .....</b>	<b>6</b>
<b>5.</b>	<b>Einleitung zum OpenDCC GBM .....</b>	<b>7</b>
5.1	<i>Funktionsbeschreibung .....</i>	7
5.2	<i>Hardware Version, Schaltplan und Bestückungsdruck.....</i>	8
5.2.1	<i>Hardware Version.....</i>	8
5.2.2	<i>Schaltplan und Bestückungsdruck.....</i>	8
<b>6.</b>	<b>Bestückung des GBMboost v1.6 .....</b>	<b>9</b>
6.1	<i>THT Bauteile einlöten .....</i>	9
6.2	<i>Betriebsmöglichkeiten (Master/Node) .....</i>	10
6.3	<i>Versorgung .....</i>	11
6.4	<i>Betriebsmodus Master .....</i>	12
<b>7.</b>	<b>Bestückung des GBMboost v1.8 und v1.9.....</b>	<b>14</b>
7.1	<i>THT Bauteile einlöten .....</i>	14
7.2	<i>Betriebsmöglichkeiten (Master/Node) .....</i>	16
7.3	<i>Versorgung .....</i>	17
7.4	<i>Betriebsmodus Master .....</i>	18
<b>8.</b>	<b>Statusanzeige am GBMboost .....</b>	<b>20</b>
<b>9.</b>	<b>Bestückung des GBM16T.....</b>	<b>21</b>
9.1	<i>THT Bauteile einlöten .....</i>	21
9.2	<i>Kontrolle .....</i>	22
<b>10.</b>	<b>Statusanzeige am GBM16T.....</b>	<b>24</b>
<b>11.</b>	<b>Weitere Informationen zum GBM .....</b>	<b>25</b>
11.1	<i>Konfiguration des USB-Chips mit dem OpenDCC Treiber .....</i>	26
11.1.1	<i>FTDI-Chip konfigurieren.....</i>	26
11.1.2	<i>Treiber am PC installieren .....</i>	27
11.2	<i>Firmware Update GBMboost .....</i>	28
11.2.1	<i>Update mit dem Programmer .....</i>	28
11.2.2	<i>Update mit dem Bootloader.....</i>	34
11.3	<i>Firmware Update am GBM16T .....</i>	38
11.3.1	<i>Update mit dem Programmer .....</i>	38
11.3.2	<i>Update mit dem Bootloader.....</i>	41
11.5	<i>Kalibrierung vom GBM16T.....</i>	43
11.6	<i>Verbindungskabel GBM16T / GBMboost.....</i>	45

<b>12. Konzept „BiDiBus als Variante 1“ .....</b>	<b>46</b>
12.1 Wie wird der GBMboost und GBM16T angeschlossen? .....	47
12.2 Anschlüsse am GBMboost .....	48
12.3 Anschlüsse am GBM16T .....	49
12.4 GBMboost als BiDiB-Interface .....	50
<b>13. Terminierung des BiDiBus .....</b>	<b>51</b>
13.1 Vorgehensweise der Terminierung .....	51
<b>14. Wissenswertes zur Unique-ID .....</b>	<b>53</b>

## 2. Änderungsverzeichnis

Version	Änderungsbeschreibung	Seite	geändert von	Datum
V1.12	Bilder geändert (DCC Anschlussbelegung vom GBMboost und GBM16T geändert) DCC1 und DCC2 waren vertauscht	Seite 48	C. Schörner	27.02.2013
V1.12	Diamex All AVR – als Programmer eingefügt	Seite 28	C. Schörner	27.02.2013
V1.12	Hinweis zum Potentialausgleich zwischen GBM16T und GBMboost mit Erklärung was in Verbindung mit der H-Brücke passiert	Seite 22	C. Schörner	27.02.2013
V1.12	Die Current LED in der Liste der GBMboost Statusanzeige erweitert	Seite 20	C. Schörner	04.03.2013
V1.12	Erklärung der BiDiB-Seriennummer und Unique-ID eingefügt	Seite 53	C. Schörner	08.03.2013
V1.13	Verbindungskabel GBM16T / GBMboost herstellen	Seite 45	C. Schörner	18.03.2013
V1.16	In der Beschreibung "THT Teile Einlöten" die Terminierung + Abbildung ergänzt	Seite 9	C. Schörner	19.05.2013
V1.16	Neues Kapitel zur Terminierung des BiDiBus eingefügt	Seite 51	C. Schörner	19.05.2013
V1.17	In der Beschreibung "THT Teile Einlöten" beim GBM16T, wird darauf hingewiesen, dass die Stiftleiste J6 nicht benötigt wird und nicht bestückt werden sollte.	Seite 21	C. Schörner	22.06.2013
V2.0	Aufbau zur Variante 1 (GBM-Hardware v1.8) eingefügt	Seite 14	C. Schörner	11.10.2013
V2.1	technische Erklärung für den Grund des potenzialgetrennten Netzteils verbessert	Seite 22	C. Schörner	31.01.2014
V2.2	DCC-Terminierungsjumper ergänzt	Seite 15	C. Schörner	09.02.2014
V2.3	leichte Textänderungen	komplett	C. Schörner	18.12.2014
V2.4	FUSE-Einstellungen der Prozessoren angepasst	Seite 28	C. Schörner	01.02.2016
V2.5	Leichte Textänderungen und Hinweise zur HW V1.9	Komplett	C.Schörner	28.11.2016
V2.6	Jumper J54 Bezeichnung J3 aktualisiert	Seite 30	C. Schörner	08.08.2017
V2.7	Wertigkeit von CV70 (GBM16T Kalibrierung) hat sich geändert	Seite 44	C.Schörner	14.05.2019

## 3. Hinweis

Diese Bauanleitung beschreibt den Aufbau des **SMD bestückten GBM** aus der Selbstbaureihe von OpenDCC und Fichtelbahn.

Eine weiterführende Anleitung für die Inbetriebnahme des GBMs, finden Sie im Downloadbereich von Fichtelbahn.

Lesen Sie diese Bauanleitung **vor** Beginn des Zusammenbaus sorgfältig durch und beachten Sie die Sicherheitshinweise.

Diese Anleitung erhebt nicht den Anspruch auf ein kommerziell gefertigtes Produkt. Sie dient lediglich als Hilfe zum Aufbau des Bausatzes für interessierte Modellbahner und ausschließlich für den Eigenbedarf.

Diese Anleitung wurde sorgfältig geprüft und nach bestem Wissen erstellt. Es kann keine Gewähr für Vollständigkeit, Aktualität und Richtigkeit geleistet werden. Sollten Handelsnamen oder geschützte Bezeichnungen verwendet werden, so liegen alle Rechte beim Rechteinhaber.

Es wird keine Haftung jedweder Art übernommen, die aus der Nutzung dieser Anleitung, deren Inhalte oder deren Gebrauch herleitbar wäre. Der Nutzer dieser Anleitung erklärt sich mit Ingebrauchnahme damit einverstanden.

Die hier verwendete und teilweise beschriebene Software kann über die Internetseite von [www.opendcc.de](http://www.opendcc.de) und [www.fichtelbahn.de](http://www.fichtelbahn.de) heruntergeladen, frei benutzt, erweitert und verbessert werden. Alles Weitere zur Nutzung von Software, Hardware und Applikation, ist auf der Internetseite von OpenDCC und Fichtelbahn beschrieben. Der Nutzer und Anwender erklärt sich mit den dort beschriebenen Regelungen vorbehaltlos einverstanden.

**Eine kommerzielle Nutzung der Software oder von Teilen daraus ist nicht statthaft!**

Diese Bauanleitung darf keiner anderen Nutzung zugeführt werden, außer der bestimmungsgemäßen Anwendung zum Aufbau des GBM. Anderweitige Nutzung erfordert die Zustimmung des Autors, bzw. des Rechteinhabers der Internetseiten [www.opendcc.de](http://www.opendcc.de) und [www.fichtelbahn.de](http://www.fichtelbahn.de).

## 4. Sicherheitshinweise

Das in dieser Bauanleitung beschriebene Modul ist ein elektrisch betriebenes Gerät. Es sind alle beim Betrieb notwendigen Vorsichtsmaßnahmen zu treffen, die mit dem Umgang mit elektrischem Strom anzuwenden sind.

- Legen Sie an das Modul keinesfalls Netzspannung an.
- Verwenden Sie keinesfalls Schaltnetzteile von PCs. Diese Geräte sind nicht erdfrei, d.h. es können hier betriebsbedingt an den Gleisen und angeschlossenen Geräten hohe Spannungen auftreten – Lebensgefahr!
- Erden Sie keinesfalls leitfähige Teile ihrer Modellbahnanlage!
- Alle Schirmungen, Kabelschirme usw. sind ggf. wenn als notwendig erachtet auf einen gemeinsamen, erdfreien Punkt zusammen zu führen.
- Das fertige Modul ist ausschließlich mit Schutzkleinspannung und Schutztrennung zu betreiben.
- Modelleisenbahnen sind in der geläufigen Rechtsauffassung als Spielzeug eingestuft. Hier gelten besondere Bestimmungen.
- Zur Stromeinspeisung sind ausschließlich die im Handel erhältlichen Netzspeisegeräte mit der entsprechenden Zulassung zu verwenden.

Achten Sie beim Erwerb auf die entsprechende Klassifizierung des Netzgerätes. Näheres erfahren Sie unter [www.vde.de](http://www.vde.de).

### Bestimmungsgemäßer Gebrauch:

Das Modul ist dafür vorgesehen ausschließlich in Modellbahnanlagen, welche digital gesteuert werden sollen, zum Fahren, Schalten und Melden eingesetzt zu werden.

**Jeder andere Gebrauch ist nicht bestimmungsgemäß.**

Das Modul ist nicht dafür bestimmt von Kindern unter 14 Jahren zusammengebaut, eingebaut oder betrieben zu werden.

### Werkzeuge und Arbeitsmittel

Sie benötigen:

- Lötzinn 0,5 oder 0,3 mm Durchmesser
- ggf. Flussmittel
- Reinigungsmittel, Pinsel, 100% Isopropylalkohol
- Lupenleuchte, besser Mikroskop
- LötKolben 30 Watt, oder besser eine thermisch geregelte Lötstation.



## 5.2 Hardware Version, Schaltplan und Bestückungsdruck

### 5.2.1 Hardware Version

Vom SMD-bestückten GBM Bausatz gibt es zwei unterschiedliche Hardware Versionen v1.6 und v1.8. Im Juli 2013 wurde der erste SMD bestückte Bausatz mit der HW v1.6 gestartet. Im Zuge einer neuen Bestellaufgabe wurden ein paar minimale Hardware-änderungen an der Platine, die sich mit der Zeit als sinnvoll gezeigt haben, durchgezogen.

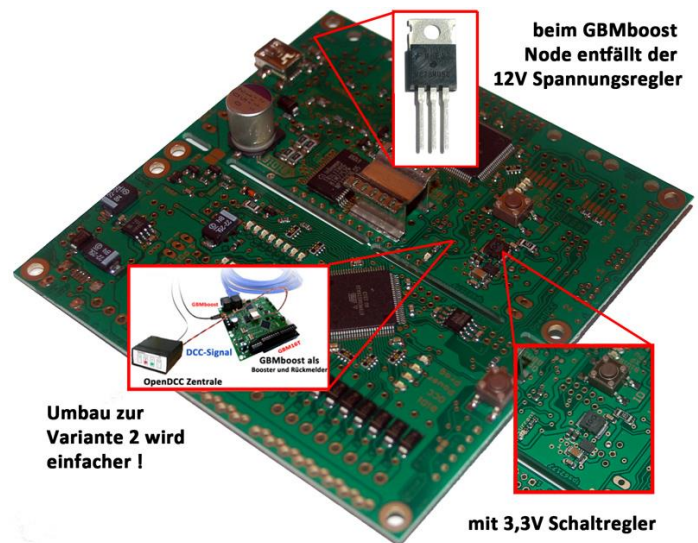


Abbildung 2: Hardwareänderung GBM v1.8

- **Der 3,3V Spannungsregler wurde durch einen 3,3V Schaltregler ersetzt.** Das bietet einen besseren Wirkungsgrad und eine geringere Wärmeentwicklung auf der Platine. Mit dieser Modifizierung stehen auch ausreichend Leistungsreserven für die Versorgung von weiteren Addon-Anwendungen bereit.
- **Der Hardwareumbau auf Variante 2 (externe Zentrale) wird mit der neuen Hardware Version etwas einfacher.** Es müssen jetzt nur noch die fehlenden Bauteile bestückt bzw. Lötjumper verändert werden. Die Aufbauanleitung für Variante 2 wurde um die neue Vorgehensweise erweitert.
- **Der 12V Spannungsregler wird nur beim GBMboost Master für die Xpressnet und BiDiBus-Versorgung benötigt.** Kommt der GBMboost als Node zum Einsatz, entfällt der Einbau des Spannungsreglers. Auf der neuen Platine v1.8 wurde dieses Versorgungskonzept leicht abgeändert.
- **weitere Bestückungs- und Hardwarefehler aus der FAQ sind eingeflossen.**

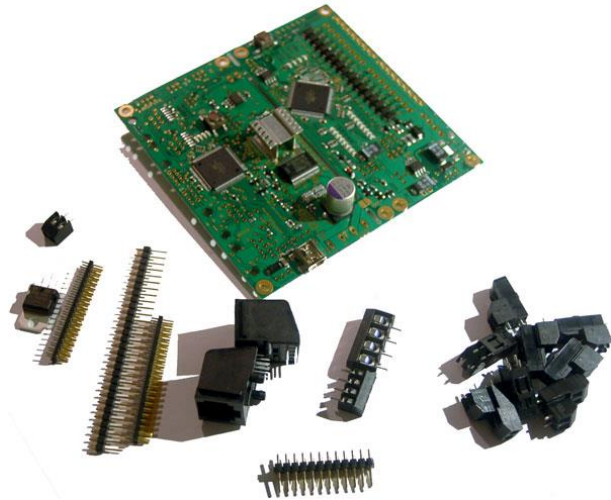
Diese Hardwareänderungen haben keinen Einfluss auf die bekannten Funktionen des GBMs oder dessen Firmware. Es können problemlos die bestehenden Hardware Versionen v1.6 und v1.8 miteinander zum Einsatz kommen.

Die Hardware Version v1.8 ersetzt ab dem **09.10.2013** die Hardware Version v1.6 vom SMD-bestückten GBM Bausatz. In diesem Handbuch finden Sie getrennt zu jeder Hardware Version eine ausführliche Aufbaubeschreibung.

### 5.2.2 Schaltplan und Bestückungsdruck

Den Schaltplan und den Bestückungsdruck finden Sie mit hoher PDF-Qualität im Downloadbereich vom GBM-Baustein auf der Fichtelbahn-Webseite

## 6. Bestückung des GBMboost v1.6



Im **Kapitel 6** geht es um die Bestückung der noch notwendigen bedrahteten Bauteile für den **GBMboost**.

Des Weiteren gibt es für den GBMboost zwei **Betriebsmöglichkeiten** (Master / Node) und verschiedene **Versorgungskombinationen** (USB, BiDiB...).

Diese Einstellungen werden im **Kapitel 6** ebenfalls behandelt.

Abbildung 3: Bauteile vom SMD-Bausatz v1.6

### 6.1 THT Bauteile einlöten

Im beiliegendem Säckchen finden Sie die noch zu bestückenden Teile.

**IC5** ... Der 12V Festspannungsregler ist zu bestücken bei einer Eingangsspannung von größer 12V an der Klemme X34. (Einbauichtung beachten ... siehe **Abbildung 4**) Wird für den Fahrbetrieb eine Eingangsspannung von 12V benötigt (abhängig von der Spurweite), dann entfällt der Einbau vom Spannungsregler IC5. Bei diesem Fall muss aber der Pin1 (Bohrloch) des IC5 mit einer Drahtbrücke zum Pin3 des Spannungsreglers IC5 verbunden werden.

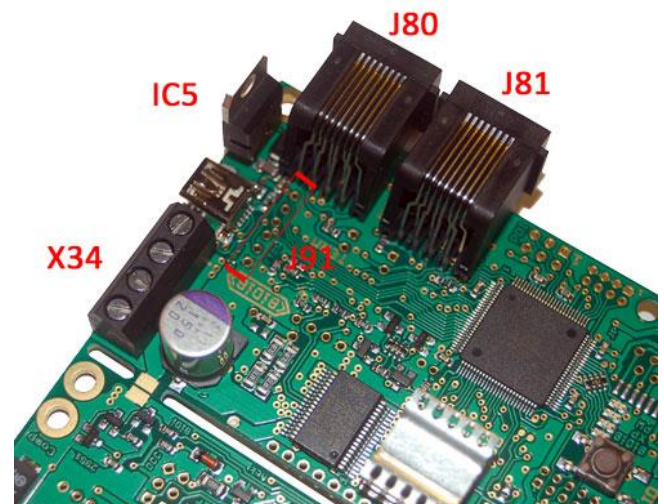


Abbildung 4: GBMboost v1.6 - Aufbau Teil 1

**J80, J81** ... Rj45 Buchse für den BiDiBus

**X34** ..... Anschlussklemme für die Versorgungsspannung und DCC-Booster Ausgang.

**J91**... Debugschnittstelle des GBMboost (optional)

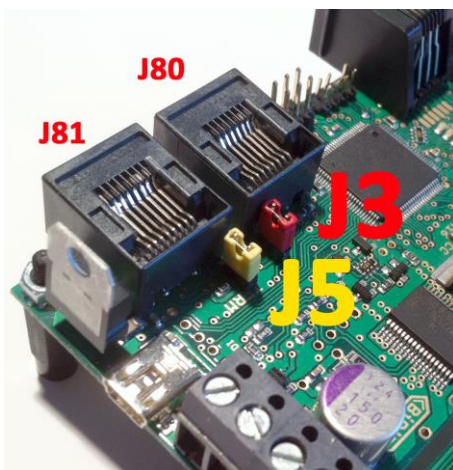


Abbildung 5: GBMboost v1.6 - Aufbau Teil 2

Auf der Oberseite werden die beiden RJ45-Buchsen eingelötet **J80** und **J81**.

Der SMD-Widertand **R30** (vorbestückt) versteckt sich neben dem **roten Jumper / Stiftleiste J3**. Beide Bauteile werden für die BiDiBus Terminierung benötigt. Weitere Informationen zur BiDiBus Terminierung finden Sie im **Kapitel 13**. Der **gelbe Jumper** ist für die DCC-Terminierung und sollte ebenfalls an beiden Enden der Busverkabelung geschlossen werden.

Die **Lötjumper SJ5** und **SJ6** müssen bei dieser Variante 1 (Belegtmelder) geöffnet bleiben und dürfen **nicht** geschlossen werden.

**Das führt sonst zu einem Kurzschluss!!**



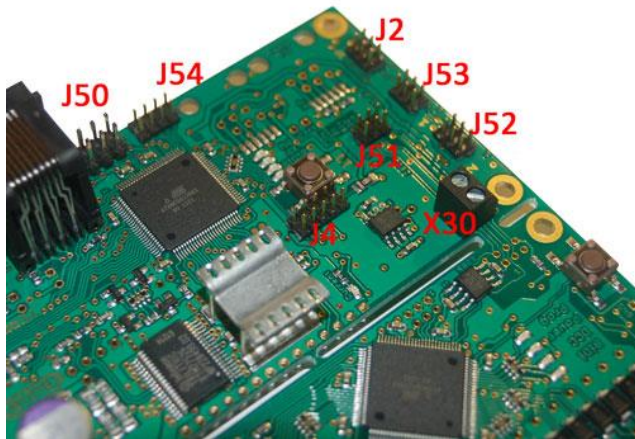


Abbildung 6: GBMboost v1.6 - Aufbau Teil 3

- 50... PDI Programmierschnittstelle
- J54... Einstellungen / für die Update Funktion am Master u. Zugriff auf die Debug-Schnittstelle
- J2... Notastaster Anschluss
- J51-J53 ... Anschluss für weitere GBM16T Module
- J4 ... Display Stiftleiste  
(optional - nicht im Bausatz enthalten)

## 6.2 Betriebsmöglichkeiten (Master/Node)

Mit dem Kauf eines **Master-** oder **Node-Device** im Fichtelbahn Shop wurde dieser Punkt „**Betriebsmöglichkeiten**“ für Sie schon ausgeführt und der Prozessor mit dem passenden Bootloader und Firmware ausgestattet. **Dieser Punkt ist für Sie nur als Information!**

Die Kommunikation auf dem BiDiBus findet auf der Basis von **Master** und **Slave** statt. Man hat am BiDiBus einen Master und zahlreiche weitere BiDiB-Baugruppen als Slave. Diese werden in der BiDiB-Sprache als **Node** bezeichnet!

**Deshalb muss man einen GBMboost zum Master machen!!**  
**Nur der Master hat eine Verbindung über USB zum PC!!**  
**Es darf auch nur ein Master am BiDiBus hängen!!**

### Der Unterschied zwischen Master und Slave (abgesehen von der unterschiedlichen Firmware):

Es sind zu bestücken auf der Rückseite für den Master:

**R57, R58** mit 4,7kOhm und **R100** mit 1kOhm

**Bemerkung:**

Die Bias-Widerstände R57, R58 können auch mit 1,5kOhm bestückt werden.

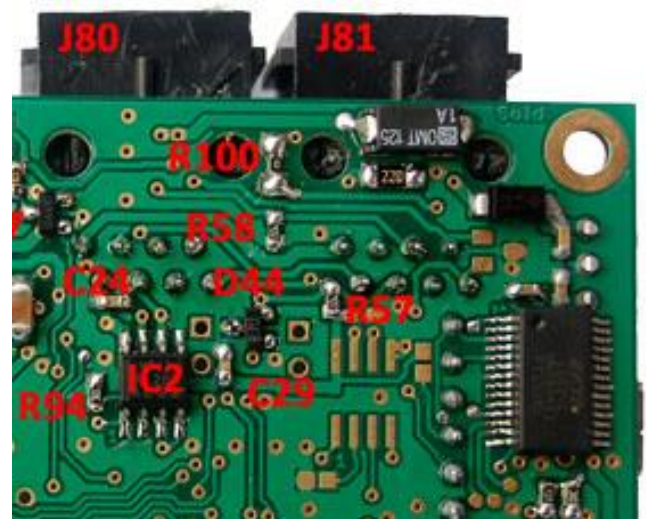


Abbildung 7: GBMboost v1.6 - Aufbau Teil 4



**Die Widerstände R57, R58 und R100 dürfen nur beim Master-Device bestückt werden.** Bei allen weiteren OpenDCC GBMs die als Node (Slave Device) am BiDiBus hängen bleiben die Pads unbestückt. Der R100 Widerstand ist ein Pullup Widerstand für das ACK-Netz. Die beiden anderen Widerstände sind BIAS Widerstände und sorgen für einen stabilen Arbeitspunkt für den RS485-Baustein.

## 6.3 Versorgung

In dieser **vorbestückten SMD-Variante** wurden die grundlegenden Kombinationen schon vorbestückt und Sie müssen nichts ändern! Dieser Punkt ist für Sie nur als Information.

Der GBMboost kann aus verschiedenen Quellen versorgt werden oder selber als Quelle für den BiDiBus dienen.

Im **Handbuch "Aufbau und Inbetriebnahme" Bausatz OpenDCC GBM** findet man eine ausführliche Bestückungserklärung.

**Ihr SMD-bestückter GBMboost ist vorbereitet für:**

**Versorgung über USB, über den BiDiBus und über eine externe Versorgungsquelle**

Bei Verwendung einer dieser Quellen ist kein Umbau notwendig!  
Man sollte aber dessen Leistungskapazität im Auge behalten!

**aus USB versorgt: (max. 500mA)**

nur geeignet für ein Modul als Interface oder Gleisbesetzmelder ohne Booster Funktion

**aus dem BiDiBus versorgt:**

nur geeignet für Gleisbesetzmelder ohne Booster Funktion. Die Voraussetzung dafür ist, dass ein GBMboost den BiDiBus speist (dafür wäre am besten der Master geeignet).

**aus einer externen Spannungsquelle: (empfehlenswert)**

geeignet für alle Einsatzbereiche und notwendig, wenn der GBMboost mit Booster Funktion betrieben wird. Das Netzteil (12V-20V DC) wird an der Klemme X34 Pin1/Pin2 angeschlossen.

## 6.4 Betriebsmodus Master

### Der GBMboost „Master“ muss den BiDiBus mit Spannung versorgen!

In diesem Fall muss man die Diode **D51** ausbauen und durch eine 500mA bis 1A SMD-Sicherung **F2** ersetzen (rechtes Bild).



**Eine einfachere Lösung** ist die beiliegende rückstellende Sicherung **F2** (im SMD-Bausatz beiliegend) parallel zu der Diode **D51** aufzulöten und umzuklappen.

Das linke Foto zeigt die parallel Sicherung (rundes gelbes Bauteil) zur Diode am Master GBMboost.

Abbildung 8: GBMboost v1.6 als Master Teil1

Dieser Umbau ist nur notwendig am GBMboost Master.



Abbildung 9: GBMboost v1.6 als Master Teil2

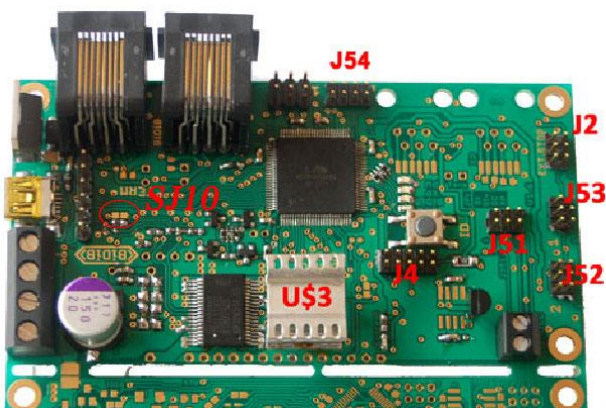


Der GBMboost „Master“ versorgt den BiDiBus mit einer 12V DC Spannung über die Sicherung **F2**. Das führt zu einem größeren Leitungsabfall am 12V Spannungsregler.

**Deshalb sollte der Spannungsregler am GBMboost „Master“ mit dem beiliegendem Kühlblech gekühlt werden.**

Abbildung 10: GBMboost v1.6 als Master Teil3

Durch die BiDiBus-Versorgung erhalten alle BiDiB-Baugruppen auch Ihre Spannungsversorgung. Hier sollte man den tatsächlichen Verbrauch im Auge behalten. Es ist empfohlen jede Baugruppe mit einer separaten Versorgungsspannung zu betreiben. Dadurch kann man eine Überlastung des BiDiBus vermeiden. Der GBMboost benötigt für seine Booster-Funktion eine externe Versorgung, nur dann kommt es zu einer Aktivierung der Booster Funktion!



Am **miniUSB** ausschließlich und **maximal 5 VDC** anlegen, sonst raucht der USB UART FT 235RL ab.



Abbildung 11: GBMboost v1.6 als Master Teil4

Je nach Einsatz können manche Stiftleisten weggelassen werden.  
Kurze Erklärung für welche Anwendung man die einzelnen Stiftleisten benötigt:

<b>J54</b>	Einstellungen	(bei Update u. DEBUG notwendig)
<b>J2</b>	externer Notastaster	(optional)
<b>J4</b>	Booster Display	(optional)
<b>J51</b>	Verbindung zum GBM16T (1)	
<b>J53</b>	Verbindung zum GBM16T (2)	
<b>J52</b>	Verbindung zum GBM16T (3)	
<b>SJ10</b>	DCC Verteilung, Betriebsmodus	(abhängig von den Varianten, in diesem Fall Var.1 offen)



Nach dem Aufbau, die Platine nochmals gründlich mit Alkohol von allen Lötückständen auf beiden Seiten reinigen. Anhand der Anleitung nochmals sorgfältig den gesamten Aufbau prüfen. **Alles OK ?** - dann geht's weiter.

Zur Kontrolle mittels strombegrenztem Netzteil die Betriebsspannung am X34 anlegen.  
(Polarität beachten / Beschriftung auf der Platinen Rückseite).

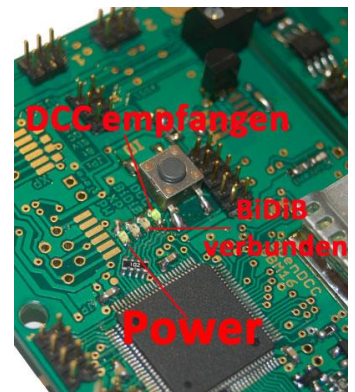


Abbildung 12: GBMboost v1.6 Betriebszustand

Ein **Master-Device** sowie ein **Node-Device** sind sofort einsatzbereit!

Im Normalfall flimmert jetzt die grüne **Power-LED** und die **DCC-LED** signalisiert, dass ein DCC Signal vorhanden ist. Bei jedem empfangenem und gesendetem BiDiB-Paket flackert die **BiDiB-LED**. Der Strom steigt beim Node auf ca. 70mA, beim Master auf ca. 120mA an.



**Damit haben Sie den GBMboost erfolgreich in Betrieb genommen.**

## 7. Bestückung des GBMboost v1.8 und v1.9



Im **Kapitel 7** geht es um die Bestückung der noch notwendigen bedrahteten Bauteile für den **GBMboost**.

Des Weiteren gibt es für den GBMboost zwei **Betriebsmöglichkeiten** (Master / Node) und verschiedene **Versorgungskombinationen** (USB, BiDiB...).

Diese Einstellungen werden im **Kapitel 7** ebenfalls erklärt.

Abbildung 13: Bauteile vom SMD-Bausatz v1.8

### 7.1 THT Bauteile einlöten

Im beiliegendem Säckchen finden Sie die noch zu bestückenden Teile.

**IC5 ... Der 12V Festspannungsregler ist nur zu bestücken beim GBMboost Master und bei einer Eingangsspannung von größer 12V an der Klemme X34.**

(Einbaurichtung beachten ... siehe **Abbildung 14**)

Wird für den Fahrbetrieb eine Eingangsspannung von 12V benötigt (abhängig von der Spurweite), dann entfällt der Einbau vom Spannungsregler IC5. In diesem Fall muss aber der Pin1 (Bohrloch) des Spannungsreglers IC5 mit einer Drahtbrücke zum Pin3 (Bohrloch) des Spannungsreglers IC5 verbunden werden.

Wird ein GBMboost Node realisiert, entfällt der Spannungsregler komplett und es wird auch keine Drahtbrücke notwendig.

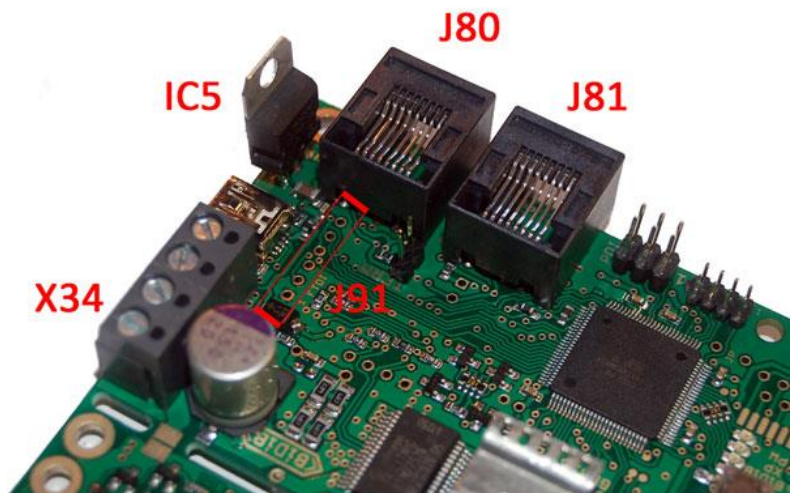


Abbildung 14: GBMboost v1.8 - Aufbau Teil 1

**J80, J81 ...** Rj45 Buchse für den BiDiBus

**X34 .....** Anschlussklemme für die Versorgungsspannung und DCC-Booster Ausgang.

**J91...** Debugschnittstelle des GBMboost (optional)

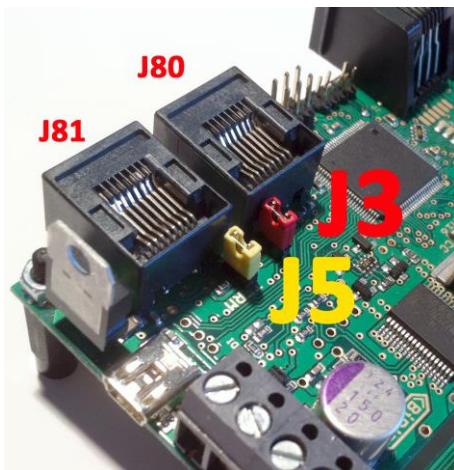


Abbildung 15: GBMboost v1.8 - Aufbau Teil 2

Auf der Oberseite werden die beiden RJ45-Buchsen eingelötet **J80** und **J81**.

Der SMD-Widertand **R30** (vorbestückt) versteckt sich neben dem **roten Jumper / Stiftleiste J3**. Beide Bauteile werden für die BiDiBus Terminierung benötigt. Weitere Informationen zur BiDiBus Terminierung finden Sie im **Kapitel 13**.

Der **gelbe Jumper J5** ist für die DCC-Terminierung und sollte ebenfalls an beiden Enden der Busverkabelung geschlossen werden.

Die Lötjumper **SJ5** und **SJ6** müssen bei dieser Variante 1 (Belegtmelder) geöffnet bleiben und dürfen **nicht** geschlossen werden.  
**Das führt sonst zu einem Kurzschluss!!**

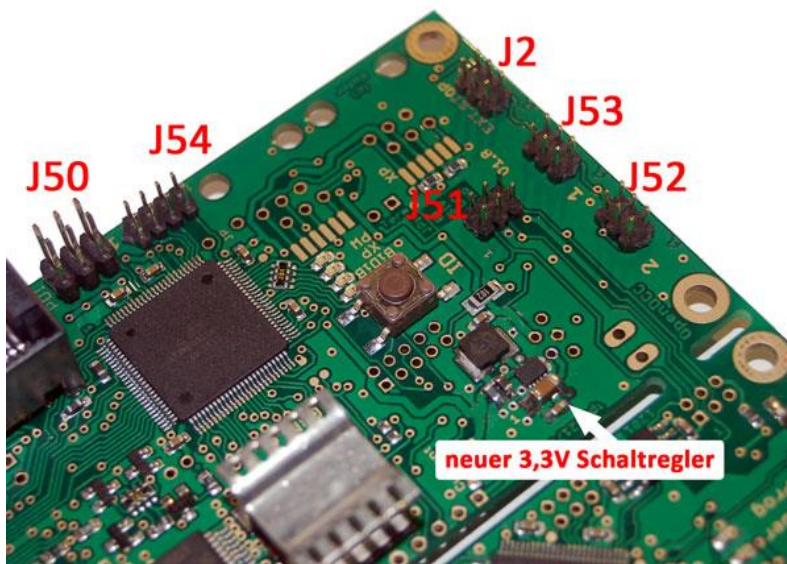


Abbildung 16: GBMboost v1.8 - Aufbau Teil 3

**J50...** PDI Programmierschnittstelle

**J54...** Einstellungen / für die Update Funktion am Master u. Zugriff auf die Debug-Schnittstelle

**J2...** Notastaster Anschluss

**J51-J53 ...** Anschluss für weitere GBM16T Module

## 7.2 Betriebsmöglichkeiten (Master/Node)

Die Kommunikation auf dem BiDiBus findet auf der Basis von **Master** und **Slave** statt. Man hat am BiDiBus einen Master und zahlreiche weitere BiDiB-Baugruppen als Slave. Diese werden in der BiDiB-Sprache als **Node** bezeichnet!

**Nur der Master hat eine Verbindung über USB zum PC!!**  
**Es darf auch nur ein Master am BiDiBus hängen!!**

### Der Unterschied zwischen Master und Slave

(abgesehen von der unterschiedlichen Firmware):

Der Master benötigt auf der Rückseite drei Widerstände **R100**, **R58** und **R57**. Diese Widerstände werden beim Node nicht benötigt.

Mit Hilfe von Lötjumpern wurde bei der Hardware Version v1.8 das Master / Node Verfahren verbessert. Jetzt müssen keine Widerstände entfernt bzw. aufgelötet werden, wenn man einen Wechsel vornehmen möchte.

Beim Master werden die Lötjumper **SJ57**, **SJ58** und **SJ100** geschlossen. Beim Node bleiben die drei Lötjumper geöffnet.

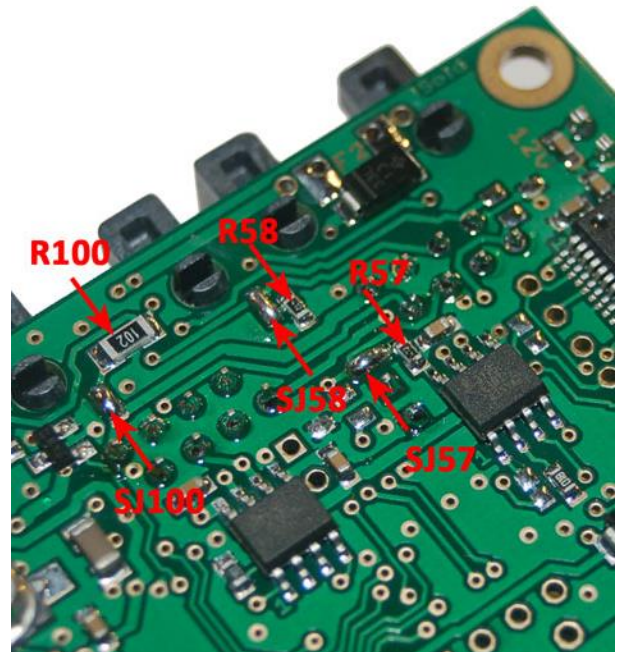


Abbildung 17: GBMboost v1.8 - Aufbau Teil 4

Beim Master werden die Lötjumper **SJ57**, **SJ58** und **SJ100** geschlossen. Beim Node bleiben die drei Lötjumper geöffnet. **Bitte überprüfen Sie diese Konfiguration bei der Erstinbetriebnahme der Baugruppe.**



Die Lötjumper **SJ57**, **SJ58** und **SJ100** dürfen nur beim **Master-Device geschlossen werden**. Bei allen weiteren OpenDCC GBMs die als Node (Slave Device) am BiDiBus hängen bleiben die Lötjumper geöffnet.

Der R100 Widerstand ist ein Pullup Widerstand für das ACK-Netz. Die beiden anderen Widerstände sind BIAS Widerstände und sorgen für einen stabilen Arbeitspunkt für den RS485-Baustein.

## 7.3 Versorgung

In dieser **vorbestückten SMD-Variante** wurden die grundlegenden Kombinationen schon vorbestückt und Sie müssen nichts verändern! Die weitere Erklärung ist für Sie nur als Information bei eventuellen Veränderungen abgedruckt.

Der GBMboost kann aus verschiedenen Quellen versorgt werden oder selber als Quelle für den BiDiBus dienen. Im **Handbuch "Aufbau und Inbetriebnahme" Bausatz OpenDCC GBM** findet man eine ausführliche Bestückungserklärung.

**Ihr SMD-bestückter GBMboost ist vorbereitet für:**

Versorgung über USB und über eine externe Versorgungsquelle an X34

**aus USB versorgt: (max. 500mA)**

nur geeignet für ein Modul als Interface oder Gleisbesetzmelder ohne Booster Funktion

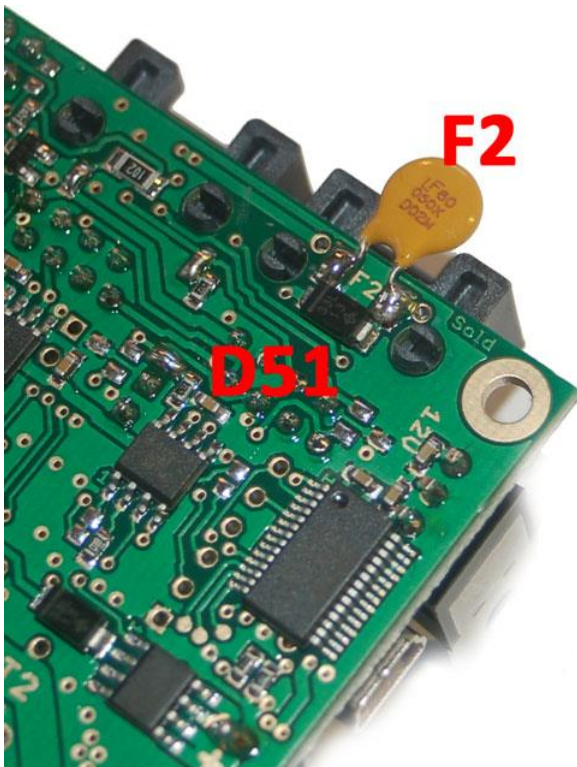
**aus dem BiDiBus versorgt:**

nur geeignet für Gleisbesetzmelder ohne Booster Funktion. Sie können mit dem Auflöten einer Sicherung F1 (500mA) und mit dem Überbrücken der Diode D51 die Versorgung aus dem BiDiBus aktivieren. Voraussetzung dafür ist, dass ein weiterer GBMboost den BiDiBus speist (dafür wäre am besten der Master geeignet).

**aus einer externen Spannungsquelle: (empfehlenswert)**

geeignet für alle Einsatzbereiche und notwendig, wenn der GBMboost mit Booster Funktion zum Einsatz kommt. Das Netzteil (12V-20V DC) wird an der Klemme X34 Pin1/Pin2 angeschlossen.

## 7.4 Betriebsmodus Master



**Der GBMboost „Master“ muss den BiDiBus mit Spannung versorgen!**

In diesem Fall muss man die Sicherung (500mA) **F2** einbauen.

Die **Abbildung 18** zeigt als **F2** eine Multifuse (gelbes Bauteil). Diese Multifuse ist Bestandteil des SMD-Bausatzes. Sie können hier auch eine 1A SMD-Sicherung einlöten.

**Dieser Umbau ist nur notwendig am GBMboost Master.**

Abbildung 18: GBMboost v1.8 als Master



Der GBMboost „Master“ versorgt den BiDiBus mit einer 12V DC Spannung über die Sicherung **F2**. Das führt zu einem größeren Leitungsabfall am 12V Spannungsregler.

**Deshalb sollte der Spannungsregler am GBMboost „Master“ mit dem beiliegendem Kühlblech gekühlt werden.**

Abbildung 19: GBMboost v1.8 Kühlblech

Durch die BiDiBus-Versorgung erhalten alle BiDiB-Baugruppen auch Ihre Spannungsversorgung. Hier sollte man den tatsächlichen Verbrauch im Auge behalten.

Falls das Kühlblech beim Aufstecken auf den Spannungsregler zu locker sitzt, können Sie mit einer Zange den Bügel des Kühlblechs zusammendrücken.

**Es ist empfohlen jede Baugruppe mit einer separaten Versorgungsspannung zu betreiben.** Damit kann man eine Überlastung des BiDiBus vermeiden. Der GBMboost benötigt für seine Booster-Funktion eine externe Versorgung, nur dann kommt es zu einer Aktivierung der Booster Funktion!



Am **miniUSB Port** ausschließlich und **maximal 5 VDC** anlegen, sonst raucht der USB UART FT 235RL (USB-Schnittstelle) ab.

Je nach Einsatz können manche Stiftleisten weggelassen werden.  
Eine kurze Erklärung für welche Anwendungen die einzelnen Stiftleisten vorhanden sind:

<b>J54</b>	Einstellungen	(bei Update u. DEBUG notwendig)
<b>J2</b>	externer Notastaster	(optional)
<b>J4</b>	Booster Display	(optional)
<b>J51</b>	Verbindung zum GBM16T (1)	
<b>J53</b>	Verbindung zum GBM16T (2)	
<b>J52</b>	Verbindung zum GBM16T (3)	
<b>SJ10</b>	DCC Verteilung, Betriebsmodus	(abhängig von den Varianten, in diesem Fall Var.1 offen)



Nach dem Aufbau, die Platine nochmals gründlich mit Alkohol von allen Lötückständen auf beiden Seiten reinigen. Anhand der Anleitung nochmals sorgfältig den gesamten Aufbau prüfen.  
**Alles OK ?** - dann geht's weiter.

Zur Kontrolle mittels strombegrenztem Netzteil die Betriebsspannung am X34 anlegen.  
(Polarität beachten / Beschriftung auf der Platinen Rückseite).

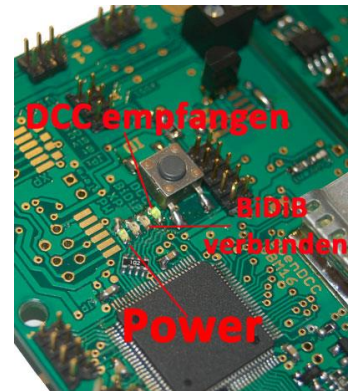


Abbildung 20: GBMboost v1.8 Betriebszustand






Im Normalfall flimmert jetzt die grüne **Power-LED** und die **DCC-LED** signalisiert, dass ein DCC Signal vorhanden ist. Bei jedem empfangenem und gesendetem BiDiB-Paket flackert die **BiDiB-LED**.

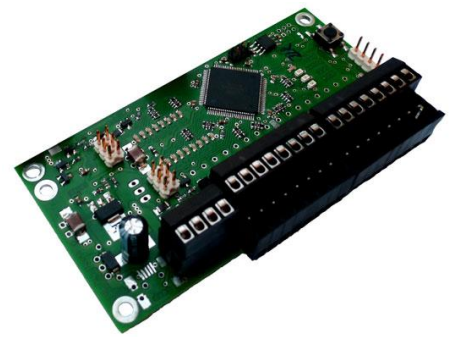
Der Strom steigt ohne angeschlossenen GBM16T beim Node auf ca. 40mA, beim Master auf ca. 80mA an.



**Damit haben Sie den GBMboost erfolgreich in Betrieb genommen.**

## 8. Statusanzeige am GBMboost

		Zustand der LED	Bemerkung
 <b>DCC</b>  <b>BiDiB</b>  <b>XP</b>  <b>PW</b>		<b>DCC LED flimmert</b>	DCC Signal wird erzeugt (beim Master) DCC Signal kommt über BiDiB (beim Node)
		<b>DCC LED aus</b>	Kein DCC Signal anliegend
		<b>BiDiB LED leuchtet</b> (beim erst verbinden)	Für BiDiB-Kommunikation bereit (beim Master) Mit dem BiDiBus verbunden (beim Node)
		<b>BiDiB blinkt kurzzeitig</b> (im Betrieb)	es findet eine Datenkommunikation statt (Zugriff auf den Baustein)
		<b>BiDiB LED aus</b>	aktuell keine BiDiB-Kommunikation / nicht mit BiDiB verbunden
		<b>XP LED aus</b>	normaler Status
		<b>XP LED an</b>	Bootloader gestartet
		<b>POWER LED flimmert</b>	GBMboost wird mit Betriebsspannung versorgt
		<b>Power LED blinkt</b>	Identifizierung gestartet (durch Taster oder Softwarefunktion)
		<b>DCC, BiDiB, XP und Power LED blinken</b>	kein eeprom-File vorhanden
		<b>XP und Power LED blinken</b>	keine Seriennummer vorhanden
		<i>Current LED in der Var. 1 leicht flimmernd, weil der Boosterausgang nicht belastet wird!</i>	
 <b>Current LED</b>		<b>Current LED flimmert</b>	LED zeigt mit steigender Helligkeit den aktuellen Verbrauch an.
		<b>Current LED blinkt</b>	Übertemperatur, Kurzschluss oder Überlast am Ausgang der Boosterendstufe



## 9. Bestückung des GBM16T

Auf dem zweiten Teil der Platine befindet sich der GBM16T mit seinen 16 Gleisbesetztkanälen. An dem GBMboost können noch zwei weitere GBM16T Module angeschlossen werden. (insgesamt drei GBM16T Module)

### 9.1 THT Bauteile einlöten

Im beiliegendem Säckchen finden Sie die noch zu bestückenden Teile.

**JP1...** PDI Programmierschnittstelle

**J6...** Settings Jumperleiste

Wird aktuell für den Betrieb nicht benötigt (empfohlen nicht zu bestücken).  
Beachten Sie bitte bei bestückter Jumperleiste, dass Sie nicht mit der Stiftleiste J1 verwechselt wird. Ein falsches Stecken kann zum Ausfall des Prozessors führen)

**J1...** Verbindung zum GBMboost

(nur notwendig, wenn der GBM16T nicht über den Platinensteg verbunden ist / getrennt vom GBMboost)

**X14..** Debugschnittstelle für Firmware Update

(wird benötigt, wenn ein Firmware Update ausgeführt werden soll ohne PDI-Programmer / aufgespielter Bootloader ist Voraussetzung)

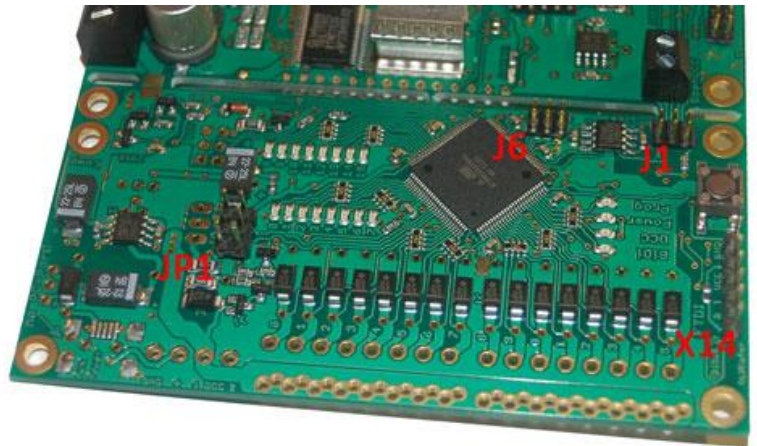
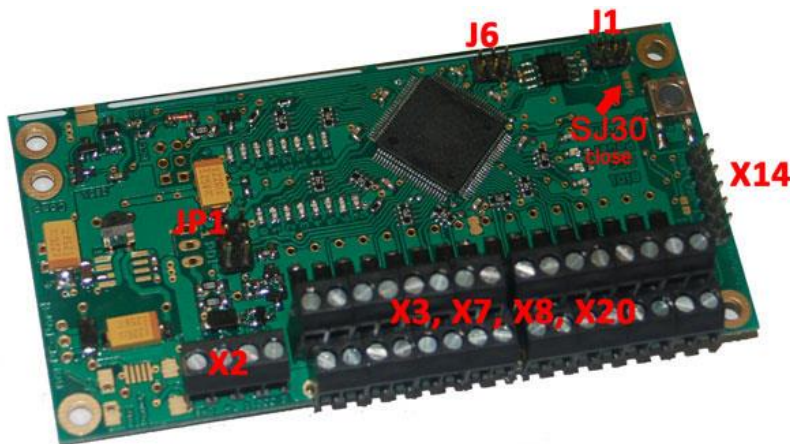


Abbildung 21: Aufbau von GBM16T Teil 1



**Fast geschafft!**

Nur noch die Anschlussklemmen **X2**, **X3**, **X7**, **X8** u. **X20**, die je nach verwendeter Anschlussklemme anders aussehen!!

Abbildung 22: Aufbau von GBM16T Teil 1

**J1** ist nur zu bestücken wenn das GBM16T Modul nicht über den Steg mit einem GBMboost verbunden ist. Die Verbindung wird in diesen Fall mit einem Flachbandkabel zum GBMboost hergestellt.



**SJ30 ist zu schließen bzw. bei der SMD-bestückten Version ist hier ein 22 Ohm**

**Widerstand eingebaut**, damit wird dem GBMboost signalisiert, dass ein Melder angeschlossen ist. Mit geöffnetem Lötjumper oder ohne dem 22 Ohm Widerstand, wird der **GBM16T** am **GBMboost** nicht erkannt.

Je nach Bedarf können für die Anschlussklemmen unterschiedliche Arten gewählt werden. Mehr Auskunft finden Sie auf der Webseite [www.fichtelbahn.de](http://www.fichtelbahn.de).

## Hinweis bei Verwendung mit einem Kehrschleifenmodul:

Für **X2** ist die Klemme RIA AKL 059-04 zu verwenden. Nur dann kann das Kehrschleifenmodul auf die Platine gesteckt werden. Bei höheren Klemmen ist das Kehrschleifenmodul nur abgesetzt mit einem Flachbandkabel möglich!

Je nach Einsatz können manche Stiftleisten weggelassen werden.

Kurze Erklärung für welche Anwendung die einzelnen Stiftleisten benötigt werden.

<b>J6</b>	Systemeinstellungen	(notwendig)
<b>J1</b>	Kommunikation zum GBMboost	(optional)
<b>JP1</b>	PDI Schnittstelle	(notwendig)
<b>SJ30</b>	<b>GBM16T Erkennung</b>	(geschlossen oder mit 22R bestückt)
<b>X14</b>	FTDI Schnittstelle für Firmware Update	(optional)
<b>X2</b>	5V DC (Ersatzspeisung) und DCC-Fahrstrom	(notwendig)
<b>X3, X7, X8, X20</b>	16 Gleisaugänge mit Masse	(notwendig)

## 9.2 Kontrolle

Der **GBM16T** hat zwei Spannungsversorgungen, die herkömmliche Versorgung über das DCC Signal und eine Hilfsspannung / Ersatzspeisung für die Funktion "**Belegtmeldung auch bei Boosterausfall**".

Zum Überprüfen der korrekten Funktionstüchtigkeit kann an **X2** (Pin1/Pin2) eine 5V Spannung (Ersatzspeisung) mit Strombegrenzung angelegt werden. Kommt es jetzt zu einem Lauflicht auf den Meldeleuchtdioden, dann funktioniert die Baugruppe.

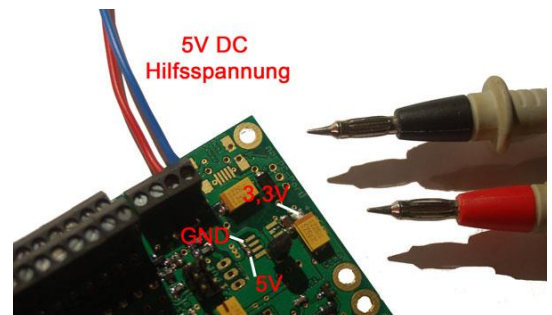


Abbildung 23: GBM16T Kontrolle

### Wichtig:

Die **Hilfsspannung / Ersatzspeisung** des GBM16T darf nicht von der gleichen **Masse-Quelle** des GBMboost stammen. d.h.

**Der GBMboost und der GBM16T müssen von zwei getrennten Netzteilen versorgt werden, sonst kommt es zu einem Potentialausgleich und Kurzschluss!!**

Durch die H-Brücke liegt das Bezugs-Potential des DCC-Signals nicht auf 0V des Netzteiles, sondern wird zwischen Minus und Plus des Netzteils hin- und her geschaltet. Somit entsteht ein Potentialunterschied zwischen „**GND**“ **GBM16T** und „**GND**“ **GBMboost**. Diese beiden Potentiale dürfen nicht zusammengebracht werden, deshalb ist für den GBM16T ein weiteres Netzteil notwendig mit erdfreien Bezug, wenn eine Belegtmeldung bei Boosterausfall gewünscht ist.

Beim Anlegen der Spannung flimmert die grüne **Power-LED** und die **DCC-LED** signalisiert durch ein blinken, dass noch kein DCC Signal empfangen wird.

Der **GBM16T** macht beim Systemstart einen Selftest, das ist zu erkennen durch das kurze Lauflicht an den Gleisstatus-LEDs.

Der Strom steigt auf der 5V Hilfsspannung auf ca. 60mA an.

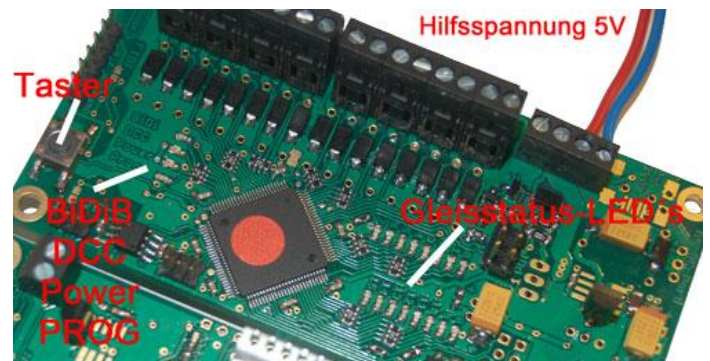


Abbildung 24: GBM16T Betriebsmodus



**Damit haben Sie den GBM16T erfolgreich in Betrieb genommen.**

## 10. Statusanzeige am GBM16T



Zustand der LED	Bemerkung
BiDiB LED aus	DCC Signal ohne RailCom-Cutout
BiDiB LED flimmert	DCC Signal mit RailCom-Cutout
DCC LED blinkt	Kein DCC Signal am Gleis nur Hilfsspannung
DCC LED flimmert	Die Gleisanschlüsse werden mit DCC versorgt
DCC LED leuchtet	Betriebsspannung anlegen mit gedrückten Taster führt zum Start des Bootloaders. Hier leuchtet solange der Taster gedrückt ist die DCC LED. Danach nur die Power LED!
POWER LED flimmert	GBM16T wird mit Betriebsspannung versorgt
POWER LED leuchtet	Bootloader aktiv nach dem Start mit gedrückten Taster
BiDiB, DCC, Power und Prog LED blinken	kein eeprom-File vorhanden
Gleis LED flimmert	Lokdecoder sendet RailCom-Informationen
Gleis LED leuchtet	Lokdecoder sendet keine RailCom-Informationen

## 11. Weitere Informationen zum GBM

Der GBMboost wird je nach Bestellung fertig als **Master** oder **Node** ausgeliefert. Das bedeutet, dass die passende Firmware mit der BiDiB-Seriennummer schon auf dem Baustein gespeichert ist und alle weiteren Konfigurationen und Kalibrierungen erfolgt sind. Das gleiche trifft auch beim GBM16T zu, dieser ist mit der aktuellen Firmware kalibriert und betriebsbereit.



Der **GBMboost** und **GBM16T** ist sofort einsatzbereit!

**Alle folgenden Beschreibungen sind als Information für ein Update oder für eine Änderung verfasst!**

### **HINWEIS:**

Sie müssen nur den OpenDCC USB-Treiber auf Ihrem PC-System installieren (siehe Kapitel 11.1.2 Treiber am PC installieren).

## 11.1 Konfiguration des USB-Chips mit dem OpenDCC Treiber

Der USB FTDI CHIP wird von Fichtelbahn schon mit den notwendigen Settings konfiguriert, dass Sie als Anwender nur noch den USB Treiber installieren müssen. Diesen finden Sie auf der Fichtelbahn Webseite.

Als Alternative zum Standard virtuellen Com-Port Treiber gibt es einen modifizierten OpenDCC GBM Treiber.

Am USB-Bus hat jeder Teilnehmer eine eindeutige VID und PID Kennung. Man spricht hier von einer Herstellerkennung und Produktkennung. Durch das USB-Branding von dem FTDI-Chip und dem Einsatz des passenden Treiber bekommt der GBMboost einen Namen im Gerätemanager.



### Das Branding gliedert sich in zwei Teilen:

- FTDI-Chip konfigurieren (wurde von Fichtelbahn schon durchgeführt)
- Treiber am PC installieren

### 11.1.1 FTDI-Chip konfigurieren

Damit in der Hardware nicht als „irgendein“ Port, sondern als OpenDCC GBM Device erkannt wird, muss man die Produktdaten in den Chip schreiben. Verbinden Sie den GBMboost über ein USB Kabel mit dem PC.

Der FTDI-Chip Hersteller bietet dazu ein kleines Tool mit dem Namen „FT Prog“ an. Link: [http://www.ftdichip.com/Support/Utilities/FT\\_Prog%20v2.6.8.zip](http://www.ftdichip.com/Support/Utilities/FT_Prog%20v2.6.8.zip)

**Info:** Das Programm, alle notwendigen Daten und Treiber sind im Fichtelbahn-Donwload zu finden.

- 1) Nach dem Öffnen des Tools auf die kleine Lupe klicken. Jetzt werden alle USB Devices eingelesen. Es bietet sich an, alle nicht gebrauchten USB Geräte abzustecken, dass es nicht zu einer Verwechslung kommt.

**Wichtig:**  
nicht betroffene Devices mit „Close Device“ schließen, sonst werden diese ebenfalls mit editiert.

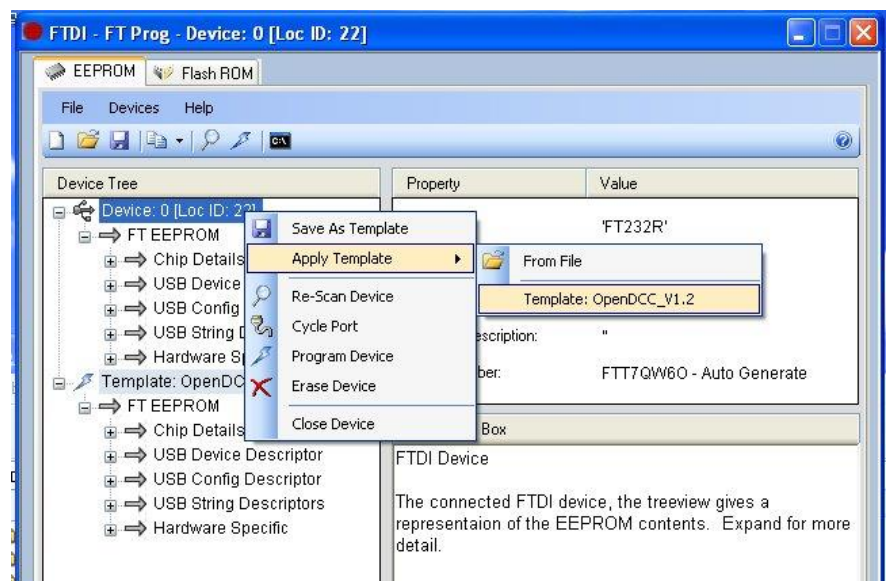


Abbildung 25: FTDI Chip konfigurieren Teil 1

2) Im nächsten Schritt laden Sie das modifizierte Template „GBMboost“ aus dem Ordner, dann auf Device mit rechts klicken / Apply Template / From File und das Template „GBMboost“ aus dem Downloadordner laden.

3) Um die modifizierten Einstellungen auf den Chip zu übertragen klicken Sie in der Statusleiste auf den **Blitz**.

Vergewissern Sie sich, dass nur das eine Device ausgewählt ist, das auch für das Branding vorgesehen ist. Mit einem Klick auf **Program** wird das Template übertragen auf den FTDI Chip.

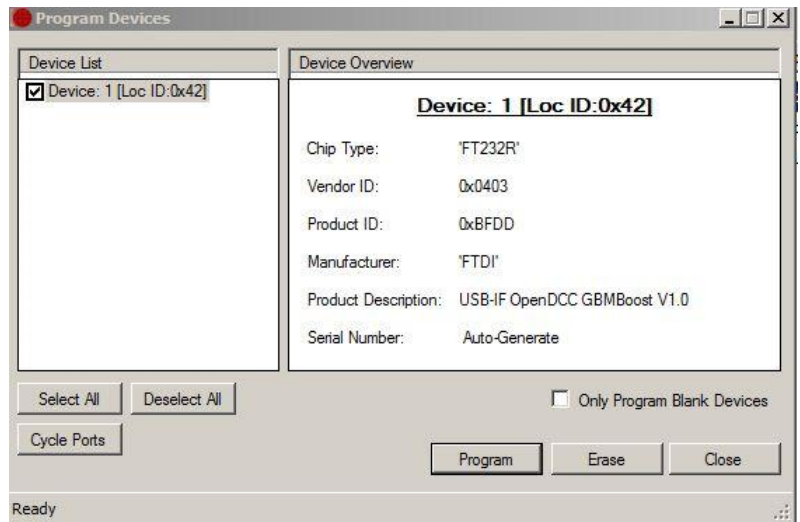
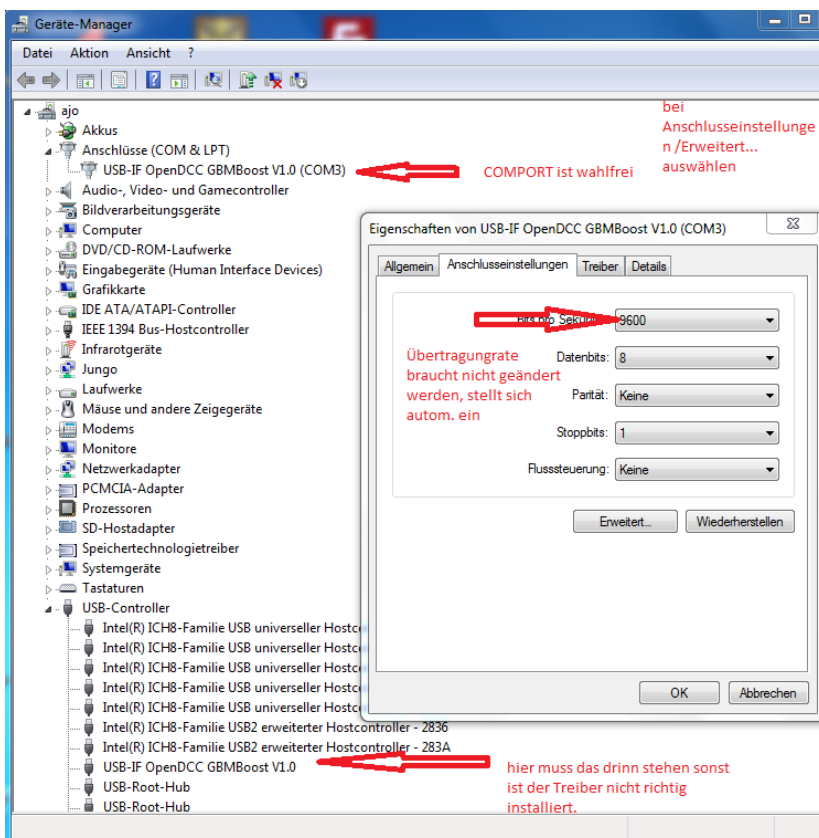


Abbildung 26: FTDI Chip konfigurieren Teil 2

**Der Schritt „FTDI-Chip“ konfigurieren ist somit abgeschlossen!**

## 11.1.2 Treiber am PC installieren

Trennen Sie den GBMboost von der USB-Versorgung u. schließen Sie ihn erneut an!



Je nach verwendetem Betriebssystem muss der FTDI-Treiber nach-installiert werden.

Den passenden Treiber finden Sie ebenfalls im Download-Ordner.

**Überprüfen Sie die erfolgreiche Erkennung des virtuellen COM-Ports im Gerätemanager.**

Abbildung 27: FTDI Treiber installieren

**Damit ist auch der zweite Teil des USB-Branding abgeschlossen.**

## 11.2 Firmware Update GBMboost

Dieses Kapitel ist als Ergänzung beigefügt und erklärt, wie man ein Firmware Update am GBMboost ausführen kann! Dies ist hilfreich um den bestehenden GBMboost einer neuen Entwicklung / Firmware anzupassen!

### Zwei Programmierverfahren führen ans Ziel:

- Die direkte Programmierung d.h. einspielen der Firmware ins EEPROM und in den Flash mit Hilfe eines Programmers, also die \*.hex und \*.eep – Files
- oder über das komfortable Bootloader-Prinzip.

### 11.2.1 Update mit dem Programmer

Der ATXmega wird mittels PDI bespielt, das ist eine Zweidraht-Schnittstelle. Der bisher übliche SPI-Adapter (wie z.B. ponyprog) kann nicht verwendet werden.

#### Verwendet werden kann:

- **AVRISPmkII:** Hierzu ist eine aktuelle Version von AVR Studio erforderlich, vor Benutzung unbedingt im Programmiermenu die Firmware des AVRISP updaten.
- **STK600:** Bei STK600 wird vom 6-poligen blauen PDI-Stecker eine 1:1 Verbindung auf das Board gezogen. Wichtig: Am STK600 muss der VTARGET-Jumper geöffnet werden!
- **JTAGICE mkII und JTAGICE mkII-CN**  
Bei dem AVR JTAGICE mkII ist die Data (PDI) mit dem JTAG Pin 9 zu verbinden.  
Bei dem JTAGICE mkII-CN (Clone) ist die Data (PDI) mit dem JTAG Pin 3 zu verbinden.

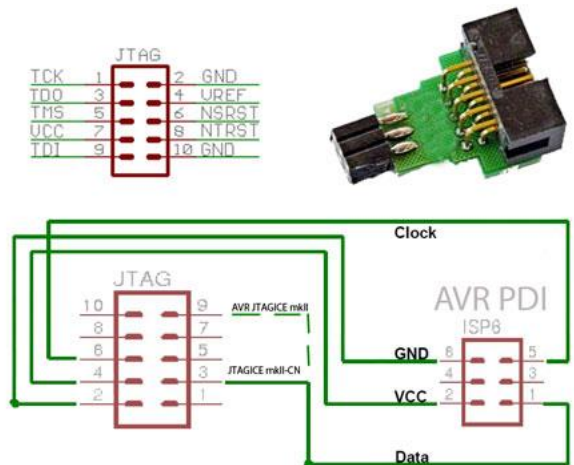


Abbildung 28: PDI / ISP Schnittstelle

### Der DIAMEX ALL AVR:



Der **DIAMEX ALL AVR** ist ein sehr preiswerter Programmer für das komplette BiDiB-Projekt. Über die PDI-Schnittstelle können all unsere ATX-Prozessoren programmiert werden.

Bezugsquelle Reichelt - [Artikel-Nr.: DIAMEX ALL AVR](#)

Der Programmierer ist genial, weil er viele Anwendungsbereiche abdecken kann und auch die programmierende Baugruppe mit Spannung versorgen könnte.

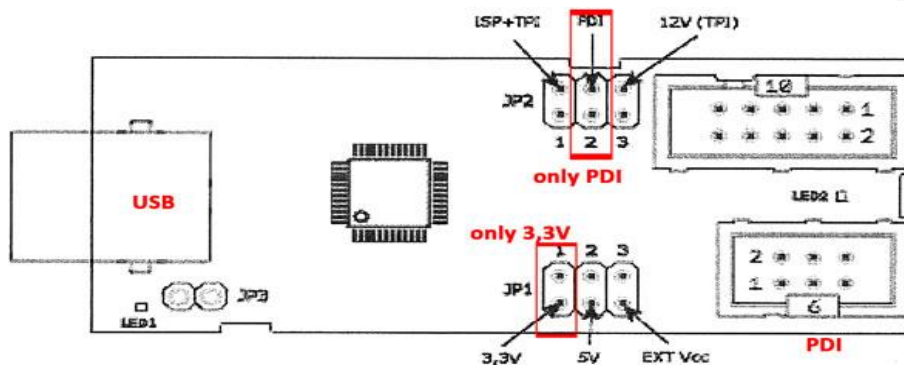


Abbildung 29: Diamex ALL AVR Programmierer

Deshalb ist vor der Programmierung absolut wichtig, dass die Jumperstellungen überprüft werden um eine Zerstörung des Programmiers und Prozessors zu vermeiden.

**Es darf nur ein Jumper auf PDI und 3,3V stecken.** Eine externe VCC Versorgung ist nicht empfohlen, weil bei Missachtung von Erdpotenzialen (anliegendes DCC Signal) der Programmierer zerstört wird, **deshalb den Jumper EXT VCC nicht stecken.**

## Wichtig:

**Den Jumper nicht auf 5V stecken, der Prozessor würde diese Betriebsspannung nicht überleben, seine Betriebsspannung liegt bei 3,3V.**

Der **DIAMEX ALL AVR** wird im AVR Studio unter dem **Device AVRISP mkII** erkannt.

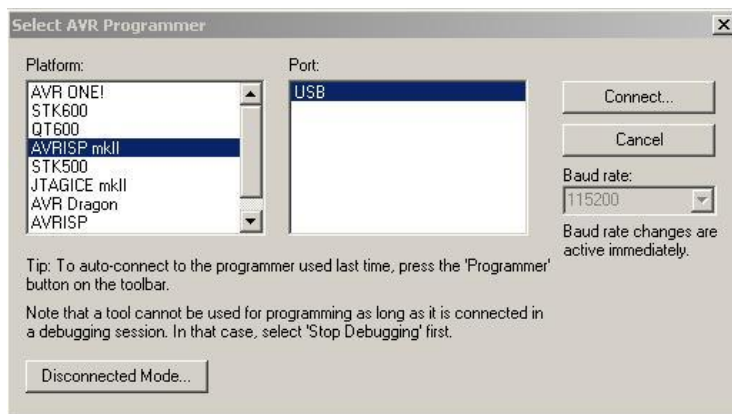


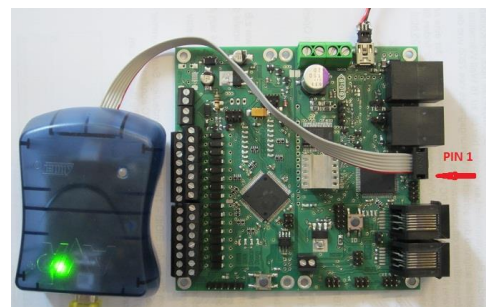
Abbildung 30: AVR Studio – Programmierer Auswahl

## Hinweis zur Stromversorgung während des Programmierens mit dem PDI Adapter:

Der GBMboost benötigt zum flashen seine eigene Spannungsversorgung und sollte nicht über den Programmierer gespeist werden. Die nachstehende Erläuterung und Screenshots vom Programmierverlauf geben die Version 4 von AVR Studio wieder.

### Verbinden Sie den GBMboost mit der Stromversorgung.

Auf richtige Polung achten, und Strombegrenzung auf 100 mA einstellen. Unprogrammiert darf der Stromverbrauch nicht über 30mA liegen. Den Programmieradapter mit dem PDI - Eingang **J50** verbinden, unbedingt auf richtige Positionierung von PIN 1 achten! PIN 1 des Programmierkabels erkennen Sie an der roten Ader.



## Die aktuelle Software steht im Internet von OpenDCC zum Download bereit.

Laden Sie die benötigten Versionen je nach beabsichtigtem Einsatz:

- Master-SW für den Betrieb als Master/Interface
- Knoten-SW (Node) für den Betrieb als Knoten ohne Interfacefunktion
- den Bootloader (identisch für Master und Node)
- die OpenDCC - BiDiB Seriennummer

Diese Dateien sind gepackt. Mit einem gebräuchlichen Entpacker wie z.B. WinRAR oder Winzip entpacken und am besten in einem separaten Ordner speichern.

## Zur Firmware:

Für alle Baugruppen, welche [BiDiB](#) unterstützen, ist die Vergabe einer eindeutigen Produktkennung zwingend erforderlich anhand einer **Seriennummer**.

**Weitere Informationen zur BiDiB-Seriennummer finden Sie am Anfang der Anleitung unter dem Punkt Einleitung:**

**„Wissenswertes zur BiDiB-Seriennummer / Unique-ID“**

## Zum Bootloader:

Der Bootloader ist ein Bestandteil der Firmware und kann nicht weggelassen werden.

Der GBM verfügt aber auch über einen doppelten Bootloader (FTDI oder BiDiB).

Dieser wird aktiviert, wenn während des Einschaltens des GBMboost Moduls der Taster gedrückt gehalten wird. Die Firmware startet dann den Bootloader, wobei der Jumper **J54, Position J3** festlegt, welche Schnittstelle Verwendung findet:

- **J3 gesteckt**, Boot über die **FTDI-Schnittstelle**:  
Hier findet die gleiche Schnittstelle wie für das Debuginterface Verwendung, allerdings ist die Baudrate auf 19200 (8N1) eingestellt. Der Boot über die FTDI-Schnittstelle wird aktiviert, wenn man vor dem Einschalten des GBM den Programmierertaster drückt und gedrückt hält. Wenn man die Taste wieder loslöst, meldet sich der Bootloader mit 'GBM\_Bootloader V??.??' über die FTDI-Schnittstelle. Nun kann man Befehl an den Bootloader senden, wobei jede Eingabe mit <cr> (Enter-Taste) abgeschlossen wird.
- **J3 offen**, Boot über die **BiDiBus-Schnittstelle**:  
Der Bootloader meldet sich selbständig am BiDiBus an und erwartet ein Firmware-Update. Dieses kann dann über die BiDiBus-Schnittstelle mit dem **BiDiB-Wizard Tool** durchgeführt werden.

## Programmieren mit dem AVR Studio:

Starten Sie das AVR Studio und bauen Sie eine Verbindung auf. Der Bildschirm müsste dann nach erfolgreichem Anmelden und Verbindung über USB so aussehen:

Wählen Sie ATXmega128A1 aus und lesen Sie die Signatur für den ATXmega128 aus:  
0x1E0x 97 0x4C.

Der Programmiermodus muss PDI gewählt werden.

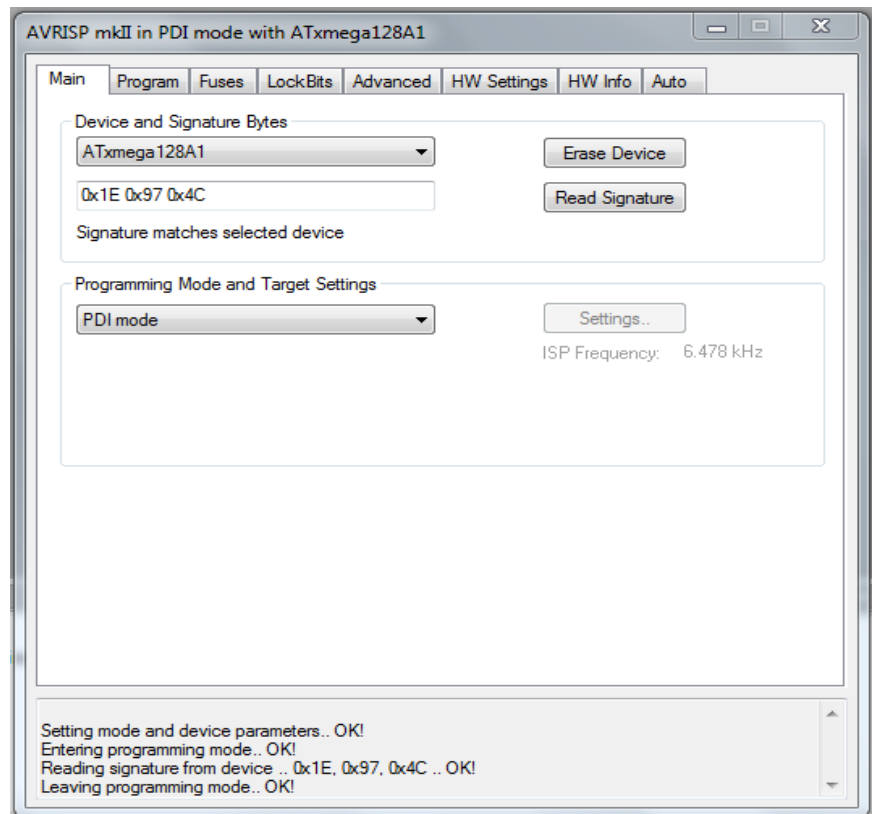
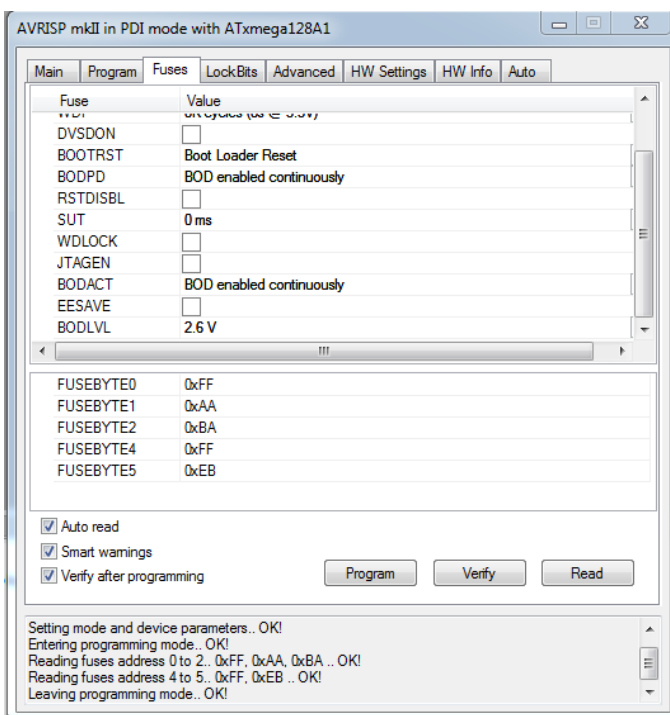


Abbildung 31: AVR Studio – Main Fenster

Als Nächstes werden die Fuses eingestellt. Dazu den Reiter „FUSES“ auswählen. Hier stellt man den Betriebsmodus des Mikrocontrollers ein. Gehen Sie hier sehr sorgfältig mit den Eingaben um! Ein „verfuster“ Atmel lässt sich nur schwer wiederbeleben.

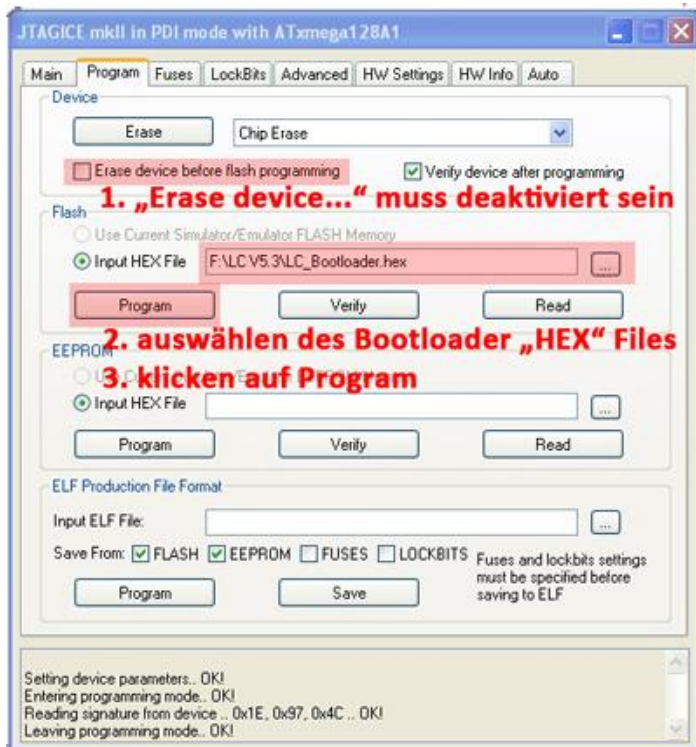


### Zusammengefasst:

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

Abbildung 32: AVR Studio – Fuse Fenster

Ist alles ordnungsgemäß eingestellt, meldet AVR Studio OK!



Das Kästchen "**Erase device before flash programming**" muss deaktiviert sein sonst wird später beim laden der Firmware der bestehende Bootloader gelöscht. Jetzt wählen Sie das **Bootloader-HEX** File aus Ihrem Downloadordner aus und mit Klick auf "**Program**" wird der Bootloader auf den GBMboost übertragen

### evtl. Kontrolle:

Sie könnten den Programmiervorgang abbrechen und mit einem Neustart des GBMboost und anschließenden **drücken auf den Taster** wird der Bootloader gestartet. Dieser findet keine Firmware und bleibt je nach JP 3 im Boot-Modus für BiDiB oder FTDI stehen. Der Zustand wird durch ein leuchten der LED signalisiert.

Abbildung 33: AVR Studio – Program Fenster Teil 1

Als nächster Schritt benötigen wir die eigentliche Firmware, das sind die beiden Dateien **\*000 .hex** und **\*001.hex**.

**Die Datei \*000.hex gehört in den Pfad "Flash". Die Datei \*001.hex in den Pfad "EEPROM".**



Der **GBMboost** kann als Master oder als Node verwendet werden. Der Unterschied liegt in der Firmware, **deshalb muss hier die passende Master oder Node Firmware übertragen werden.**

Mit Klick auf **Program** wird die Firmware auf den Mikrocontroller übertragen.

Nicht vergessen diese Prozedur bei **Flash** und **EEPROM** einzeln auszuführen. Die beiden Dateien werden nicht automatisch übertragen.

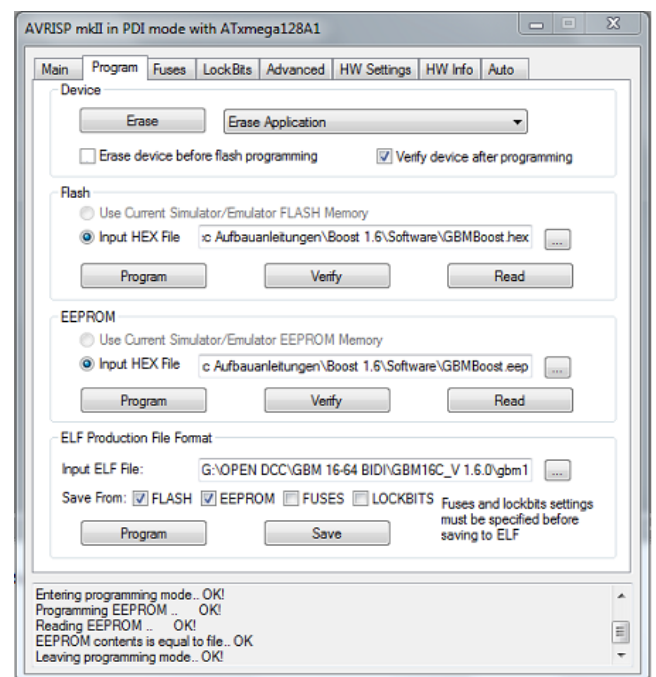


Abbildung 34: AVR Studio – Program Fenster Teil 2

**Dieser Schritt ist nur notwendig, wenn die Seriennummer aus dem USER SIGNATURE BEREICH gelöscht wurde:**

Als letzten Schritt generieren wir uns eine kostenlose Seriennummer mit dem BiDiB Seriennummer Tool auf [http://www.opendcc.de/elektronik/bidib/opendcc\\_bidib.html](http://www.opendcc.de/elektronik/bidib/opendcc_bidib.html)

Dazu ist eine Registrierung im Opendcc Forum notwendig. Mit Auswahl des passenden Produkts und einer Bemerkung kann das Seriennummer File generiert werden. Diese Datei kann gespeichert werden auf Ihrem PC mit Klick auf das gefärbte Kürzel **.eep**.

**OpenDCC - BiDiB Seriennummer**

Um eine Seriennummer zu erstellen oder einzusehen, benötigen Sie einen gültigen Zugang zum Forum. Bitte melden Sie sich nun mit den Anmeldedaten des Forums an:

Username:

Password:

Abbildung 35: BiDiB Seriennummer Generator

Die Seriennummer ist ebenfalls ein **eeprom-File** oder ein **serial\_000.hex File** das im Punkt EEPROM ausgewählt werden muss und auf den GBMboost übertragen wird mit **Klick auf Program**.

**Nicht vergessen die Seriennummer auf der Baugruppe zu vermerken.**



## Fehlermeldungen:

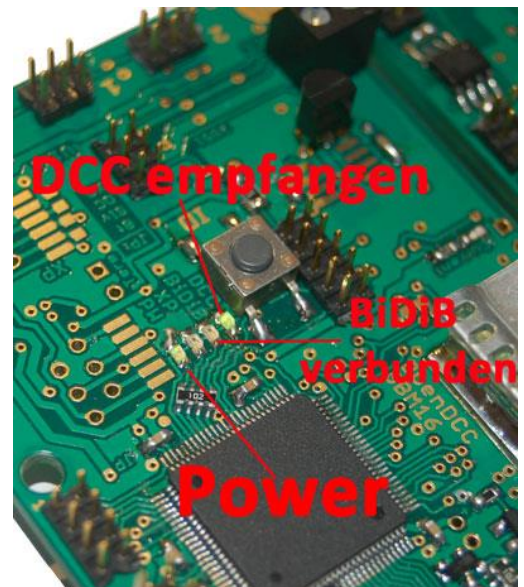
**Die vier grünen Status LEDs blinken jetzt hektisch**

- das eeprom File wurde vergessen zu übertragen auf den GBMboost

**Die zwei unteren grünen Status LEDs blinken jetzt hektisch**

- die Seriennummer wurde nicht übertragen auf den GBMboost

Im Normalfall flimmert jetzt eine grüne **Power-LED** und die **DCC-LED** signalisiert, dass ein DCC Signal empfangen wird. Bei jedem empfangenem und gesendetem BiDiB-Paket flackert die **BiDiB-LED**.  
Der Strom steigt nun auf 60mA – 70mA an.



**Damit haben Sie den GBMBoost erfolgreich aktualisiert.**

## 11.2.2 Update mit dem Bootloader

### Hinweis:

Das Update-Verfahren ist unterteilt in **Master** und **Node** (Slave). **Bitte beachten!**

### beim Master:



Das **BiDiB-Wizard Tool** zum Aktualisieren der Firmware oder zum erstmaligen aufspielen funktioniert beim Master nicht!

Der **GBMboost Master** besitzt eine USB-Schnittstelle die man für das Firmware Update mit Hilfe eines Terminalprogramms nutzen kann.

Dazu muss der GBMboost über USB mit dem PC verbunden werden.

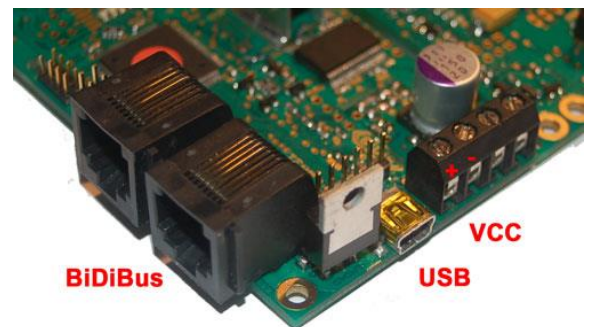
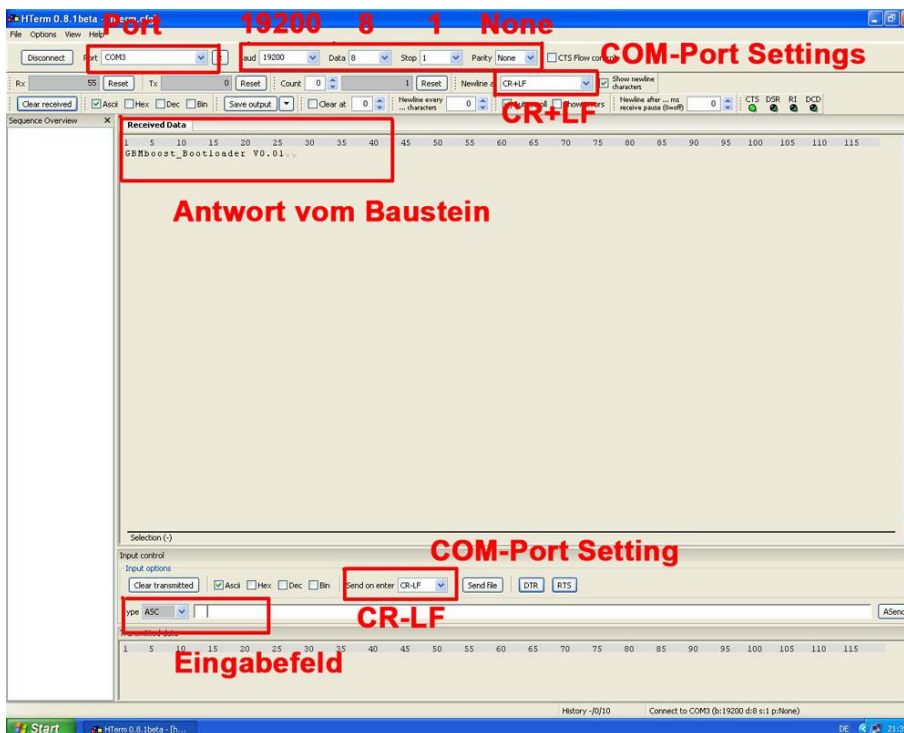


Abbildung 36: GBMboost Anschlussbelegung



Im Gerätemanager ihres Betriebssystems wird GBMboost als neues virtuelles COM-Interface erkannt. Für die Kommunikation mit dem GBMboost muss diese COM-Schnittstelle und das Terminalprogramm auf **19200 Baud und 8N1** betrieben werden.

Abbildung 37: GBMboost Master Update Teil 1

Der **GBMboost** besitzt einen Bootloader für FTDI (Serial) und BiDiB. Durch den setzen des Jumper **J54, Position J3** wird der FTDI(Serial) Bootloader ausgewählt. **Im Fall des Masters** ist das die USB-Schnittstelle.

**Bei einem Node** der nicht über BiDiB sein Update erhält und kein FTDI Chip vorhanden ist (USB-Schnittstelle), muss die Debug-Schnittstelle verwendet werden mit einem FTDI-Kabel. Hierzu muss noch Jumper J0 gesetzt werden.



Abbildung 38: GBMboost Bootloader Jumper

Halten die den **Taster** auf dem GBMboost gedrückt und schalten die Versorgungsspannung für den GBMboost ein. Jetzt bauen sie eine Verbindung zu dem Baustein auf, indem Sie auf den Button **Connect** im Terminalprogramm **hterm** klicken.

Jetzt schicken Sie ein **"?"** und bestätigen dies mit **Enter**.

**Der GBMboost antwortet mit "GBMboost\_Bootloader V?"**

### Schritt1:

Jetzt senden Sie ein **f** und bestätigen mit **Enter**.

Die Eingabe erfolgt im Eingabefeld des Terminalprogramms.

**Der GBMboost antwortet mit einem Punkt.**

### Schritt2:

Jetzt klicken Sie auf den Button **"Send file"** und wählen die passende flash Firmware-datei (\*hex oder \*000.hex) und bestätigen mit Start.

Der erste Teil der Firmware wird übertragen auf den GBMboost, sichtbar durch die zahlreichen Punkte im Terminalprogramm.

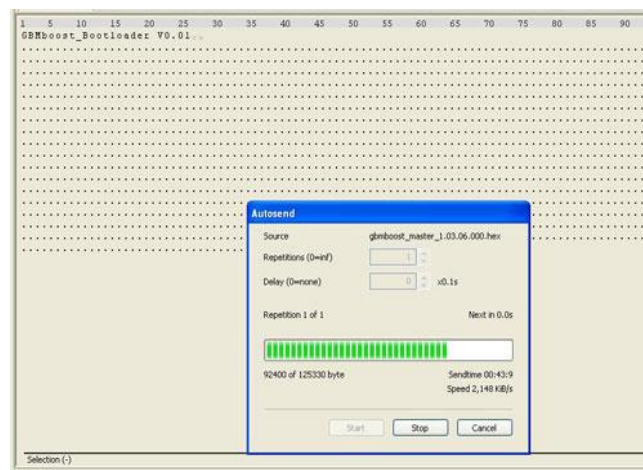


Abbildung 39: GBMboost Master Update Teil 2

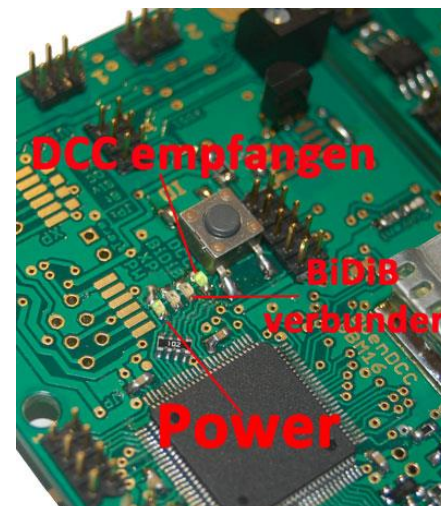
### Schritt3:

Jetzt müssen Sie noch das eeprom übertragen. Dazu senden Sie ein **e** und bestätigen das mit einem **Enter**. Der GBMboost antwortet mit einem Punkt. Sie wählen das passende eeprom File (\*.eeprom oder \*001.hex) und bestätigen mit **Start**. Es folgen wieder ein paar Punkte.

**Dieser Schritt ist nur notwendig, wenn die Seriennummer aus dem USER SIGNATURE BEREICH gelöscht wurde:**

Jetzt muss noch die gezogene Seriennummer übertragen werden (siehe Punkt 4.2 Seriennummer). Dazu senden Sie erneut ein **e** und bestätigen mit **Enter**. Wählen jetzt die gezogene Seriennummer und bestätigen mit **Start**.

**Nicht vergessen die Seriennummer auf der Baugruppe zu vermerken.**



Der **GBMboost** ist erfolgreich aktualisiert oder in Betrieb genommen. Nach dem trennen der Versorgungsspannung, entfernen des Jumpers **J3** und erneutem anschließen der Versorgungsspannung flimmert die **Power LED**.

## beim Node:

Grundsätzlich funktioniert beim **GBMboost Node** auch die beschriebene Variante vom GBMboost Master, aber die Funktion **Firmware Update** über das BiDiB-Wizard Tool ist erheblich benutzerfreundlicher.

### **Der Vorteil ist:**

Der GBMboost kann im eingebauten Zustand über das Tool aktualisiert werden, man spricht auch von einer Fernwartung.

In der Knotenliste von dem BiDiB-Wizard Tool finden Sie den GBMboost Baustein und alle weiteren BiDiB-Komponenten aufgelistet, für eine Konfiguration oder einem Firmware Update.

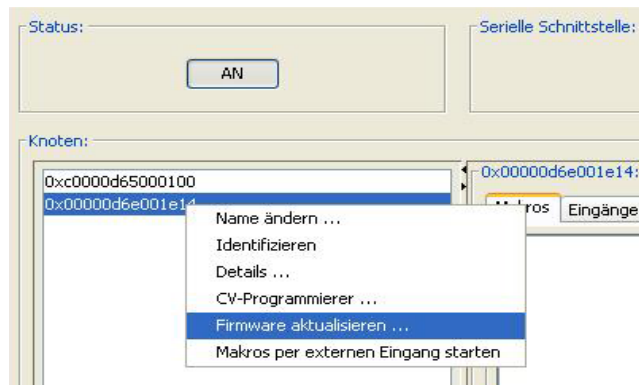


Abbildung 40: GBMboost Node Update Teil 1

Der GBMboost wird an die Versorgungsspannung gelegt und mit dem BiDiBus verbunden. **Bei erstmaligem flashen sind alle Status-LEDs aus.**

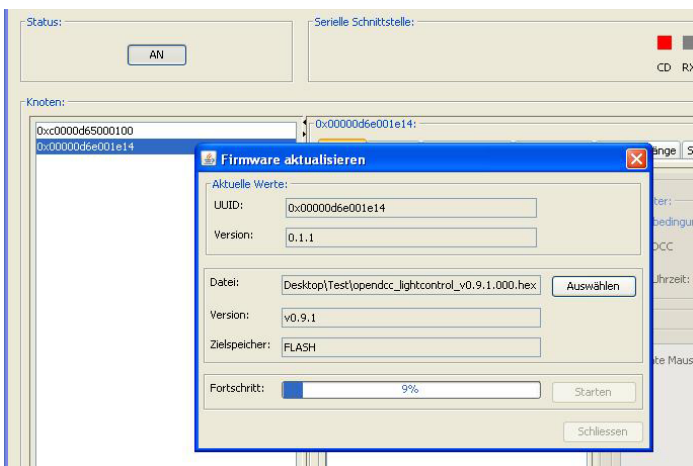


Abbildung 41: GBMboost Node Update Teil 2

Das **BiDiB-Wizard Tool** wird geöffnet und in der Knotenliste sollte die Seriennummer des GBMboost Masters (Interfaces) sichtbar sein.

Bei drücken des Tasters auf den neuen GBMboost erleuchtet die zweite grüne LED und im Tool erscheint eine neue Nummer ... **der neue GBMboost ist erwacht.**

Mit der rechten Maustaste auf die Seriennummer des neuen GBMboosts klicken und im Kontextmenü die Aktion **"Firmware aktualisieren"** auswählen.

Im nächsten Schritt muss der Ordner ausgewählt werden mit den drei Firmwaredateien (siehe oben). Diese Dateien müssen in der Reihenfolge **\*000.hex**, **\*001.hex** und **\*\*serial.001.hex** übertragen werden. Mit schließen des Fensters ist der GBMboost Node auf die neue Firmware aktualisiert und bereit.

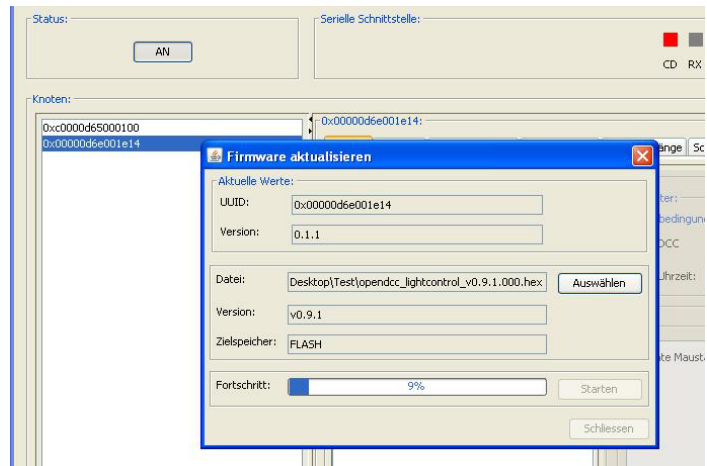


Abbildung 42: GBMboost Node Update Teil 3

## Hinweis:

**Die Seriennummer muss nur übertragen werden, wenn Sie aus dem USER SIGNATURE BEREICH gelöscht wurde.**



**Der GBMboost ist erfolgreich aktualisiert.**

## 11.3 Firmware Update am GBM16T

### Zwei Programmierverfahren führen ans Ziel:

- Die direkte Programmierung d.h. einspielen der Firmware ins EEPROM und in den Flash mit Hilfe eines Programmers, also die \*.hex und \*.eep – Files
- oder über das komfortable Bootloader-Prinzip.

### 11.3.1 Update mit dem Programmer

Weitere Informationen im Umgang mit einem Programmer finden Sie auf Seite 28.

Der GBM16T benötigt zum Flashen seine eigene Spannungsversorgung und soll nicht über den Programmer gespeist werden.

Verbinden Sie den **GBM16T** mit der Stromversorgung. Auf richtige Polung achten, und Strombegrenzung auf 100 mA einstellen. Unprogrammiert darf der Stromverbrauch nicht über 20mA liegen.

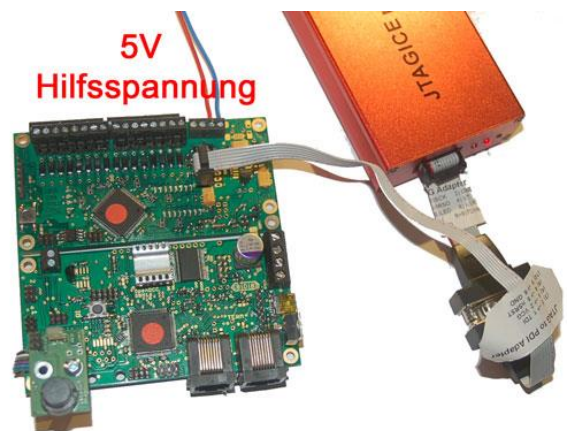


Abbildung 43: Update GBM16T Anschluss

Den Programmieradapter mit dem PDI - Eingang **J1** verbinden, unbedingt auf richtige Positionierung von PIN 1 achten!

PIN 1 des Programmierkabels erkennen Sie an der roten Ader.

#### **WICHTIG:**

Der Programmieradapter verbindet die GND des PCs mit der des GBM16T.

Daher ist absolut sicherzustellen, dass:

- **der GBMBoost / GBM16T komplett erdfrei gestellt ist**
- **kein DCC Signal am GBM16T anliegt (SJ5 und SJ6)**
- **keine USB-Verbindung vom GBMboost zum PC besteht**

dies gilt insbesondere bei der Programmierung über den PDI-Port, wenn die Brücken **SJ5** und **SJ6** bei der Erstprogrammierung schon geschlossen sind. Ein späteres Firmware Update über das FTDI-Kabel (Bootloader Variante) kann gefahrlos erfolgen, hier ist keine Maßnahme notwendig.

## Programmieren mit dem AVR Studio:

Starten Sie das AVR Studio und bauen Sie eine Verbindung auf. Der Bildschirm müsste dann nach erfolgreichem Anmelden und Verbindung über USB so aussehen:

Wählen Sie **ATXmega128A1** aus und lesen Sie die Signatur für den ATXmega128 aus:

0x1E0x 97 0x4C.

Der Programmiermodus muss PDI gewählt werden.

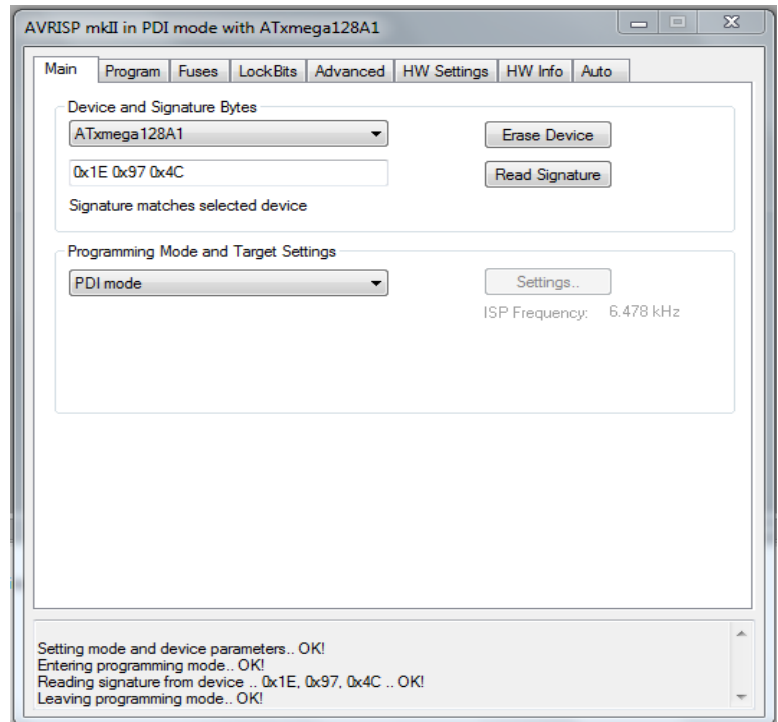


Abbildung 44: AVR-Studio - Update GBM16T Teil1

Als Nächstes werden die Fuses eingestellt. Dazu den Reiter „FUSES“ auswählen. Hier stellt man den Betriebsmodus des Mikrocontrollers ein. Gehen Sie hier sehr sorgfältig mit den Eingaben um! Ein „verfuster“ Atmel lässt sich nur schwer wiederbeleben.

### Nicht sichtbare Einstellungen:

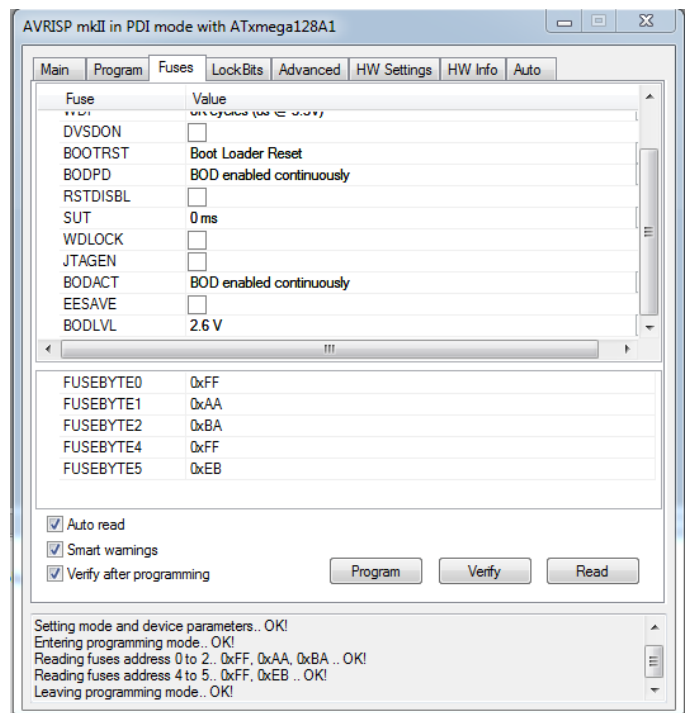
BODLEVEL: auf 2,1 Volt

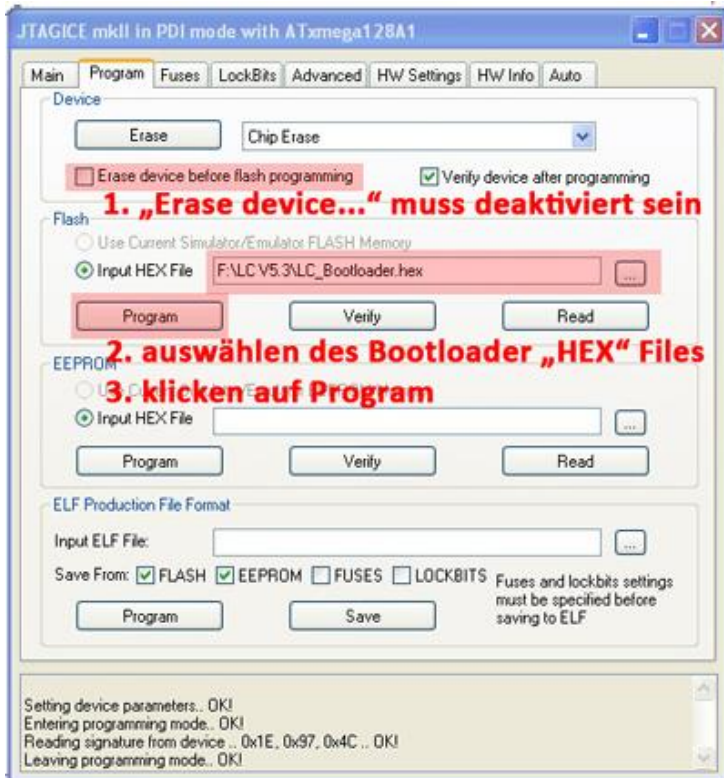
### Zusammengefasst:

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

Abbildung 45: AVR-Studio - Update GBM16T Teil2

Ist alles ordnungsgemäß eingestellt, meldet AVR Studio OK!



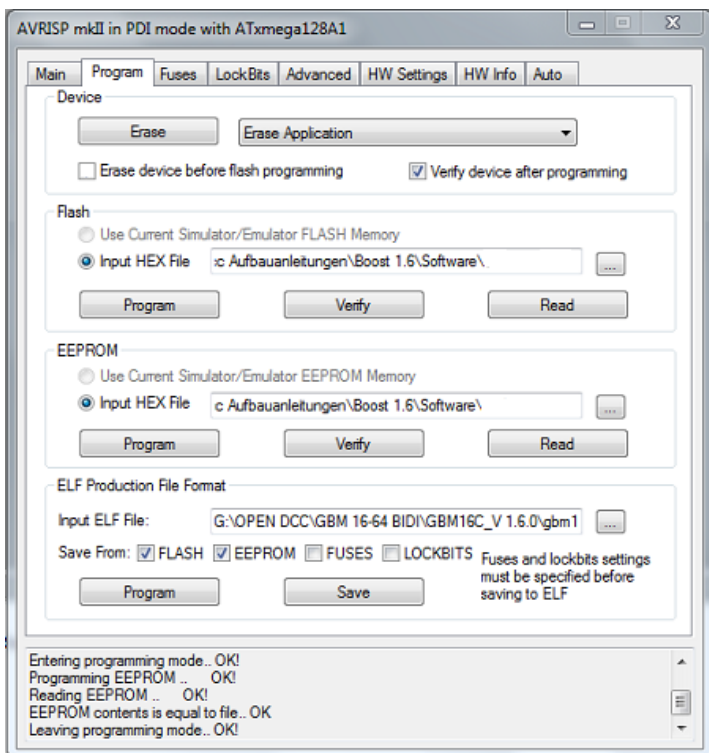


Das Kästchen "**Erase device before flash programming**" muss deaktiviert sein sonst wird später beim laden der Firmware der bestehende Bootloader gelöscht. Jetzt wählen Sie das **xboot\_gbm16t.hex** File aus Ihrem Downloadordner aus und mit Klick auf "**Program**" wird der Bootloader auf den GBM16T übertragen.

#### evtl. Kontrolle:

Sie könnten den Programmiervorgang abbrechen und mit einem Neustart des GBM16T und anschließenden **drücken auf den Taster** wird der Bootloader gestartet. Dieser findet keine Firmware und bleibt im Boot-Modus für FTDI stehen. Der Zustand wird durch ein leuchten der LED signalisiert.

Abbildung 46: AVR-Studio - Update GBM16T Teil3



Als nächster Schritt benötigen wir die Firmware, das sind die beiden Dateien gbm16t\_\*.hex und gbm16t\_\*.eep.

Die Datei **gbm16t\_\*.hex** gehört in den Pfad "Flash".

Die Datei **gbm16t\_\*.eep** in den Pfad "EEPROM".

Mit Klick auf Program wird die Firmware auf den Mikrocontroller übertragen. Nicht vergessen diese Prozedur bei Flash und EEPROM einzeln auszuführen. Die beiden Dateien werden nicht automatisch übertragen.

Abbildung 47: AVR-Studio - Update GBM16T Teil4

#### Hinweis nur als INFO:

Bei der SMD bestückten Version ist die 22 Ohm Version realisiert worden!!



Es gibt eine gbm16t\_22ohms Version und eine gbm16t\_5R6ohms Version. Die Firmware ist abhängig von der gewählten Widerstandsbestückung **R103 bis R119**. Mit der 5R6 Ohm Version ist keine Ersatzmessung (Belegtmeldung auch bei Boosterausfall) möglich, dafür ist die Hardware Railcom konform.

## 11.3.2 Update mit dem Bootloader

Der GBM16T kann über den Programmer programmiert werden, das ist sinnvoll bei einer Erstinbetriebnahme.

**Bei einem Update würden wir die Bootloader Variante vorschlagen.**

Der **GBM16T** besitzt eine Debugschnittstelle **X14** mit der man einen direkten Zugriff auf den Baustein bekommt. Diese Schnittstelle verwenden wir für das **Firmware Update**.

Der GBM16T kann nur über dieses beschriebene Verfahren aktualisiert oder in Betrieb genommen werden, wenn nicht die Programmer - Variante gewählt wird.

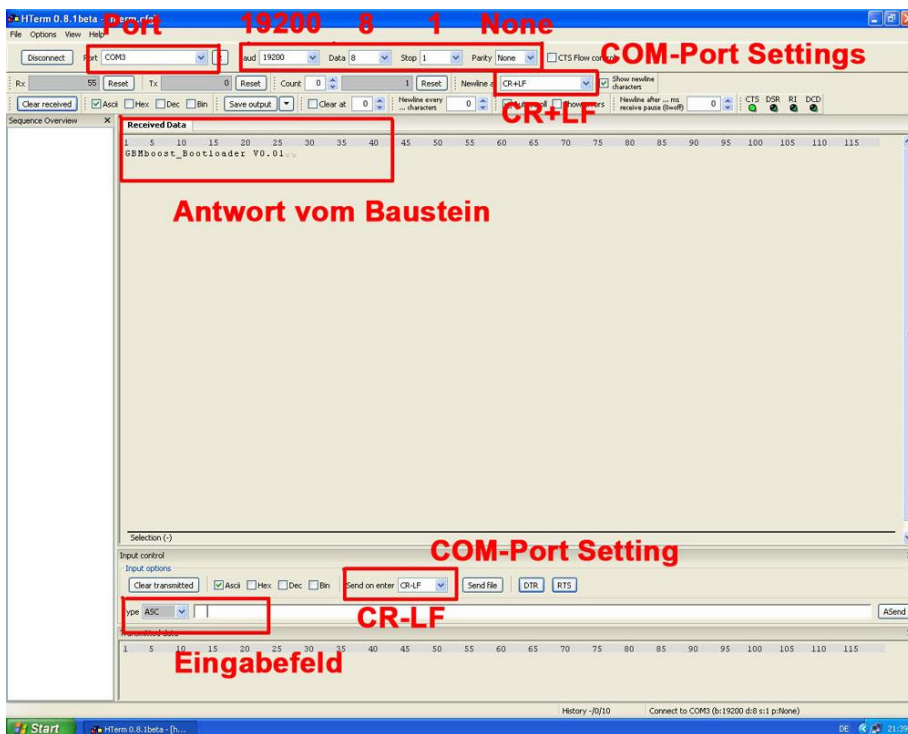


**Das BiDiB-Wizard Tool zum Aktualisieren der Firmware funktioniert am GBM16T nicht!**

### FDTI-RS232-TTL-Kabel



Sie benötigen ein **FDTI-RS232-TTL** Kabel (kostet 25 Euro) und ein Terminalprogramm (z.B. hterm).



Im Gerätemanager ihres Betriebssystems wird das TTL-Kabel als ein neues virtuelles COM-Interface erkannt. Für die Kommunikation mit dem GBM16T muss die COM-Schnittstelle und das Terminalprogramm auf **19200 Baud und 8N1** betrieben werden.

Abbildung 48: Bootloader - Update GBM16T Teil1

Jetzt bauen Sie eine Verbindung zu dem Baustein auf, indem Sie auf den Button **Connect** im Terminalprogramm „**hterm**“ klicken.

Halten die den **Taster** auf dem GBM16T gedrückt und schalten die Ersatzspeisung für den GBM16T ein.

**Der GBM16T antwortet mit "GBM16T\_Bootloader V?"**

## Schritt1:

Jetzt senden Sie ein **f** und bestätigen mit **Enter**. Die Eingabe erfolgt im Eingabefeld des Terminalprogramms.

**Der GBM16T antwortet mit einem Punkt.**

## Schritt2:

Jetzt klicken wir auf den Button "**Send File**" und wählen die passende Flash Firmware-datei (\*hex oder \*000.hex) und bestätigen mit **Start**.

Der erste Teil der Firmware wird übertragen auf den GBM16T, sichtbar durch die zahlreichen Punkte im Terminalprogramm.

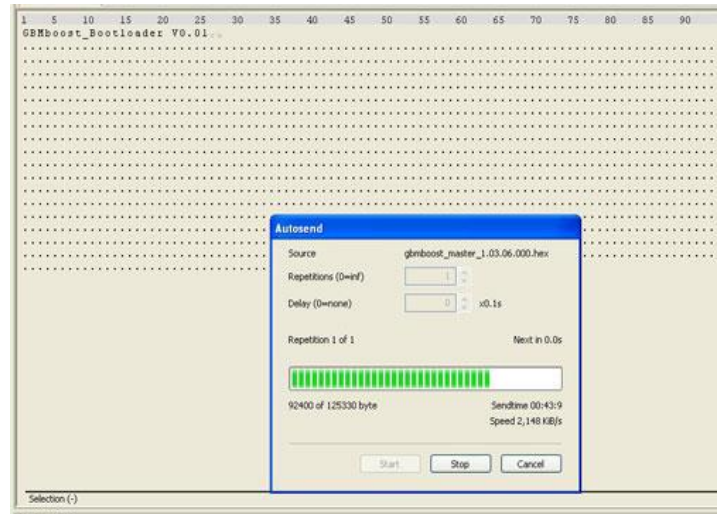


Abbildung 49: Bootloader - Update GBM16T Teil2

## Schritt3:

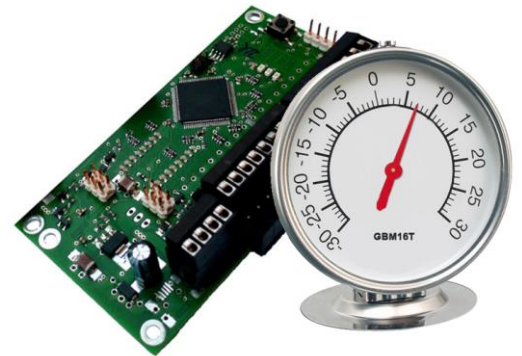
Jetzt müssen wir noch das eeprom übertragen. Dazu senden wir ein **e** und bestätigen das mit einem **Enter**. Der GBM16T antwortet mit einem Punkt. Wir wählen das passende eeprom File (\*.eeprom oder \*001.hex) und bestätigen mit **Start**. Es folgen wieder ein paar Punkte.

**Für den GBM16T wird keine Seriennummer benötigt.**

## 11.5 Kalibrierung vom GBM16T

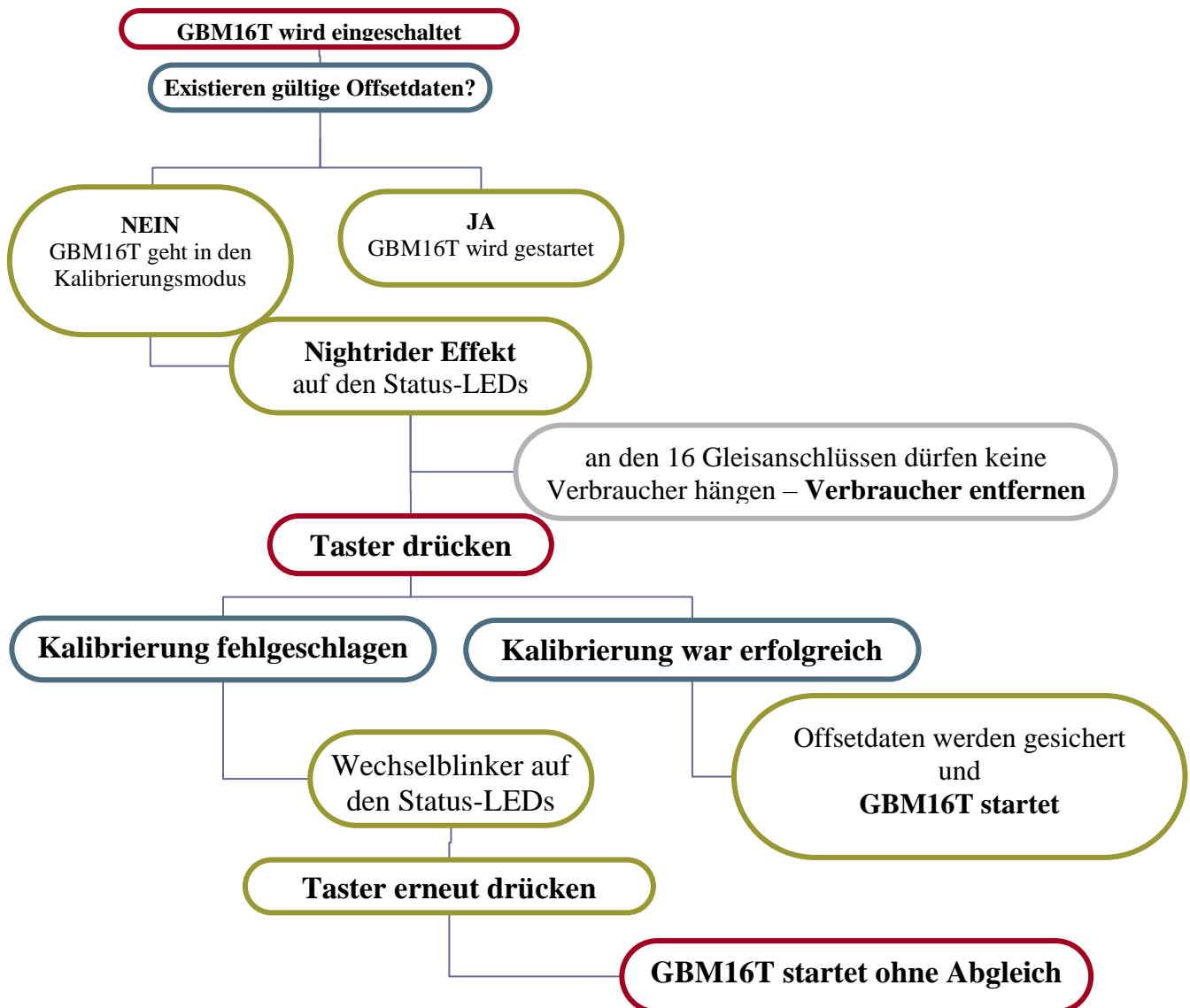
Der GBM16T arbeitet mit einer hohen Empfindlichkeit um auch bei 5V Hilfsspannung einen Verbraucher auf dem Gleis erkennen zu können. Das hat zur Folge, dass seine Genauigkeit und den damit verbundenen Wandleroffset einmalig mit der Realität abgeglichen werden muss. (Kalibrierung).

Diese Kalibrierung muss nur einmalig durchgeführt werden und wird vom GBM16T automatisch in einem geschützten Bereich (User Signature) abgelegt.



**Der Wert bleibt dem GBM16T auch nach einem „Chip Erase“ erhalten.**  
Man kann den Wert nur mit „Erase User Signatur“ löschen.

### Der GBM16T besitzt eine Kontrollroutine:



## HINWEIS:

Ist die Offset-Messung fehlgeschlagen, startet der GBM16T nach jedem Neustart im Kalibrierungsmodus. **Eine Kalibrierung ist für den GBM16T notwendig.**

## erneute Kalibrierung durchführen nach erfolgreicher Kalibrierung:

Eine nachträgliche Kalibrierung kann man auslösen, indem man den Wert des CV70 auf 1 setzt.

Der **GBM16T** ist erfolgreich kalibriert und in Betrieb genommen, wenn nach dem trennen und erneutem anschließen der Hilfsspannung / Ersatzspeisung die **Power LED** flimmert und die **DCC-LED** hektisch blinkt.

## Fehlermeldungen:

### Die vier Status LED´s blinken jetzt hektisch

- das eeprom File wurde vergessen zu übertragen auf den GBM16T

### Die vier Status LED´s blinken im Nightrider - Effekt

- der GBM16T ist im Kalibrierungsmodus und wartet auf die Bestätigung

### Die vier Status LED´s haben einen Wechselblinker

- die Kalibrierung ist fehlgeschlagen

**Ursache:** an den Gleisanschlüsse befinden sich noch Verbraucher

## 11.6 Verbindungskabel GBM16T / GBMboost

Das Verbindungskabel vom GBM16T zum GBMboost stellen wir uns selbst her. Wir schneiden das Kabel auf die gewünschte Länge zu und legen dieses in die ersten Pfostenbuchse ein. Die Buchse zeigt mit den Öffnungen nach *unten*, der rote Draht befindet sich auf der linken Seite. Pfostenbuchse zusammendrücken, evtl. zwischen den Fingern, am besten jedoch mit einer kleinen Rohrzange (250 mm). Eine weitere Möglichkeit besteht darin, die Buchse im Schraubstock vorsichtig zusammenzupressen. (Das Optimum wäre die Verwendung einer passenden Konfektionszange, die genannten preiswerteren Hilfsmittel genügen jedoch vollauf.)



Fertiggestelltes Kabel



Die Bauteile des Kabels



**Bei der Herstellung der Leitung ist nur wichtig, dass die Pin1 Markierung auf dem einen Stecker identisch mit der Pin1 Markierung auf dem anderen Stecker ist. Man spricht hier auch von einer 1:1 Verbindung.**

Dadurch ergibt sich ein Z-förmiges Gebilde oder ein U-förmiges Gebilde, wenn beide Stecker in eine Richtung zeigen.



**Wie lange darf das Kabel zwischen GBMboost und GBM16T sein?**

Die Verbindung zwischen GBM16T und GBMboost ist ein 3,3V TTL Signal. Lange Leitungswege führen zu Reflexionen. **Das bedeutet:**

Für einen störungsfreien Betrieb sollten die Reflexionen innerhalb der Schaltzeiten abgeklungen sein.

**Für das Projekt empfehlen wir eine maximale Leitungslänge von 1m (100cm).**

Daraus ergibt sich eine Spannweite von 2m, die für eine Verkabelung unter einer Anlage ausreichen sollte. **GBM16T <--1m Leitung--> GBMboost <--1mLeitung--> GBM16T.**

**Diese Leitungslänge habe wir getestet und dabei keine Reflexionen festgestellt!**

**Anschlussalternative mit dem Rastermaß 2,54mm:**



Eine kostengünstige Lösung wäre eine RM2,54 Stiftleiste auf die bestehende RM2,00 Stiftleiste Huckepack aufzulöten. Mit diesem Behilf können jetzt beide Platinen mit handelsüblichen RM 2,54mm Flachbandleitungen und Pfostenstecker verbunden werden.

## 12. Konzept „BiDiBus als Variante 1“

In dieser ersten **Variante 1** (GBMboost als Belegtmelder) gibt es einen **Master GBMboost** als Interface und zugleich als Belegtmelder für die max. drei angeschlossenen GBM16T Module. Alle weiteren **GBMboost werden zum Node** (Slave) und sind nur Belegtmelder für ihre GBM16T Module (Gleisfühler).

**Der einzige Unterschied** bei den GBMboosts ist bei dieser Variante 1, dass der Master die Master-Firmware benötigt und die Nodes mit der Node-Firmware geflasht werden müssen.

Eine DCC-Zentrale mit integriertem Booster oder wie im unteren Bild erkennbar, eine OpenDCC Z1 mit separaten Booster erzeugt die DCC-Fahrbeefehle und den Fahrstrom.

Wenn man hier eine RailCom - taugliche Zentrale und Booster verwendet, erhält man eine Rückmeldung mit RailCom - Information (Lokadresse, Fahrtrichtung...) von dem fahrenden Objekt. Bei nicht RailCom - fähiger Peripherie erhält man die klassische Belegtmeldung!!

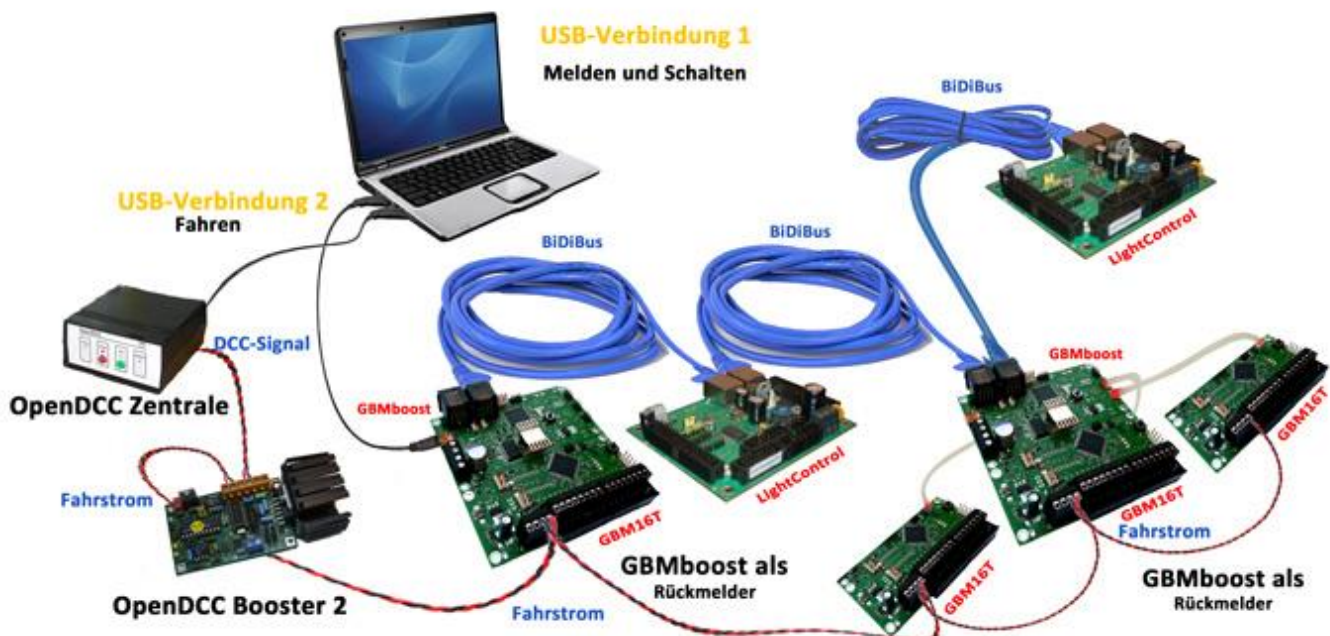


Abbildung 50: BiDiBus Topologie Variante 1



Die Lötjumper **SJ5** und **SJ6** müssen bei dieser Variante 1 (Belegtmelder) geöffnet bleiben und dürfen **nicht** geschlossen werden.  
Das führt sonst zu einem Kurzschluss!!



### Hinweis:

Beim Anschluss von weiteren **GBM16T Modulen** an einem GBMboost Master oder Node werden diese erst nach dem Neustart des betroffenen GBMboost erkannt. Die TTL-Verbindung zwischen GBMboost und GBM16T ist **nicht Hot-Plug** fähig.

## 12.1 Wie wird der GBMboost und GBM16T angeschlossen?

Die **Abbildung 51** zeigt die komplette Verkabelung des GBMs in der Variante 1:

### Spannungsversorgung:

Die Versorgung von **GBMboost** und **GBM16T** darf nicht vom gleichen Massebezug stammen, d.h. man benötigt zwei getrennte Netzteile.

Der **GBMboost** kann in dieser Ausbavariante 1 direkt von der USB-Quelle versorgt werden. Es ist aber empfohlen eine Spannungsversorgung von 12V-20V anzulegen um eine BiDiBus-Versorgung zu gewährleisten.

Der **GBM16T** kann mit einer Hilfsspannung / Ersatzspeisung versorgt werden. (5V DC mit 1A genügt). Die Ersatzspeisung kann im Inselbetrieb aufgebaut werden. Das bedeutet, man kann alle GBM16T Module die am gleichen GBMboost angeschlossen sind, von einem 5V Netzteil versorgen.

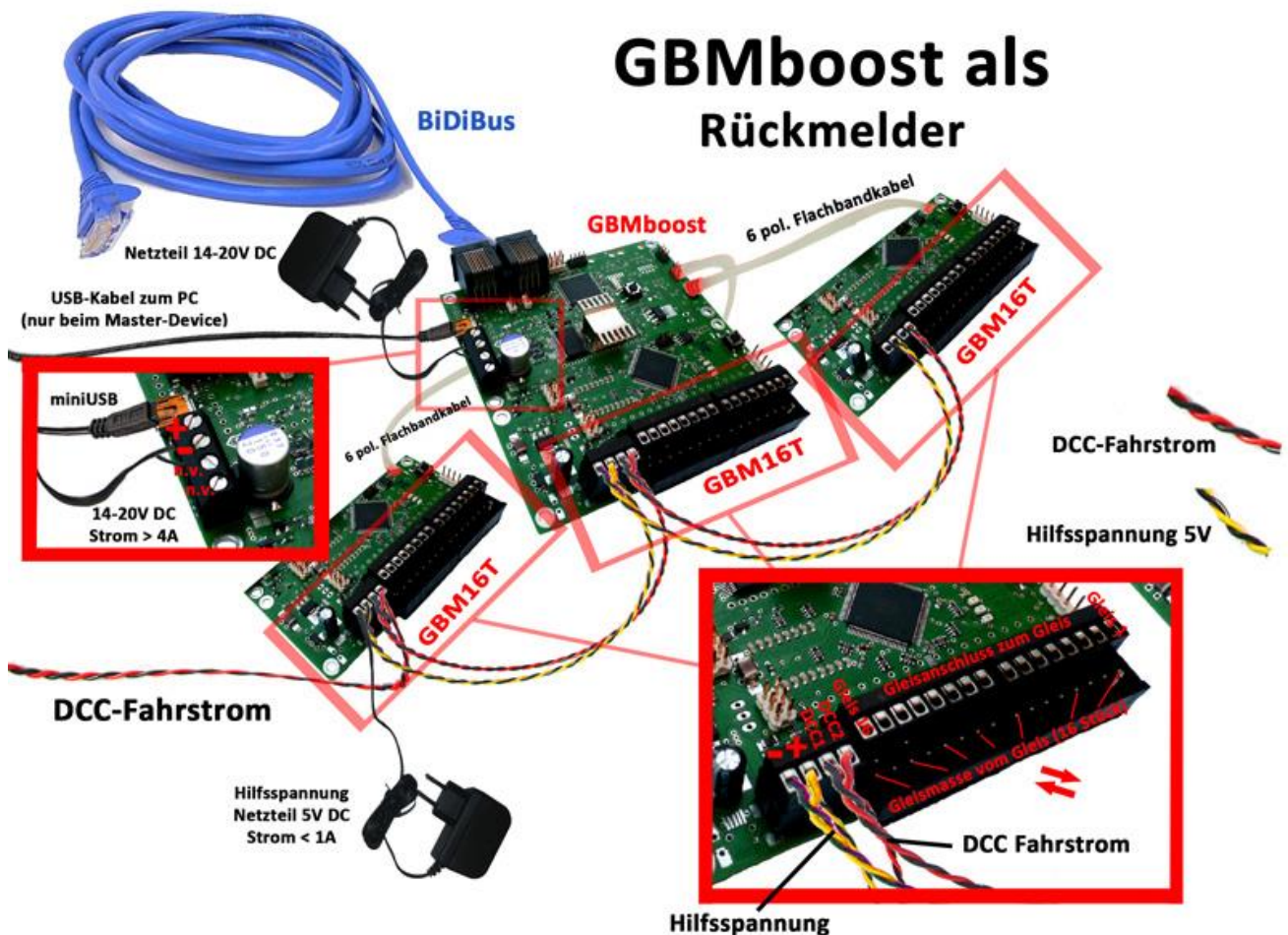


Abbildung 51: Anschlussplan für Variante 1



### Wichtig:

Es dürfen **nicht alle GBM16T Module** von einem 5V Netzteil versorgt werden!!  
**Den Inselbetrieb je GBMboost beachten!!!**

## 12.2 Anschlüsse am GBMboost

### Master:

- **USB Verbindung zum PC**  
(miniUSB - Buchse)
- **DCC-Input von externer Zentrale**  
(X30 – nicht abgebildet)

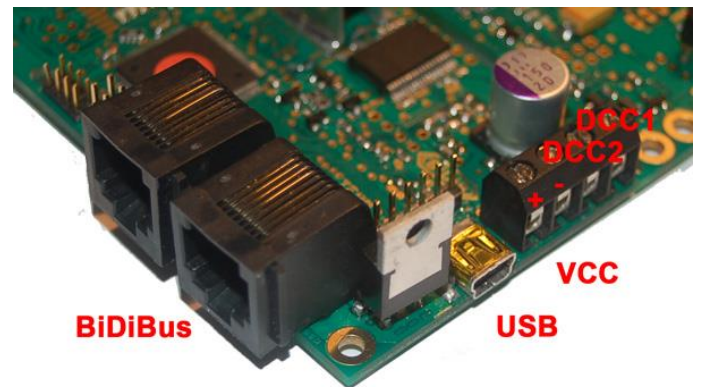


Abbildung 52: Anschlüsse GBMboost Ansicht 1

### Master und Node:

- **alle BiDiBus - Baugruppen werden in Reihe miteinander verbunden über die RJ45 Buchse** (der Master GBMboost kann auch Mittig, am Ende sowie am Anfang des BiDiBus seinen Platz finden)
- **Bei externer Versorgung wird die Klemme X34 Pin1 und Pin2 verwendet.** (Polarität beachten, Kennzeichnung auf Platinen Rückseite)
- **Der DCC-Fahrstrom vom internen Booster kann über die Klemme X34 zum GBM16T Klemme X2 geführt werden.** Eine elegantere Lösung ist, die beiden Lötjumper SJ5 und SJ6 anzuschließen.
- **Die weiteren GBM16T Module werden über die Klemme X2 vom GBM16T zum GBM16T Klemme X2 versorgt.** (siehe Anschlussbild oben)



Abbildung 53: Anschlüsse GBMboost Ansicht 2

### 12.3 Anschlüsse am GBM16T

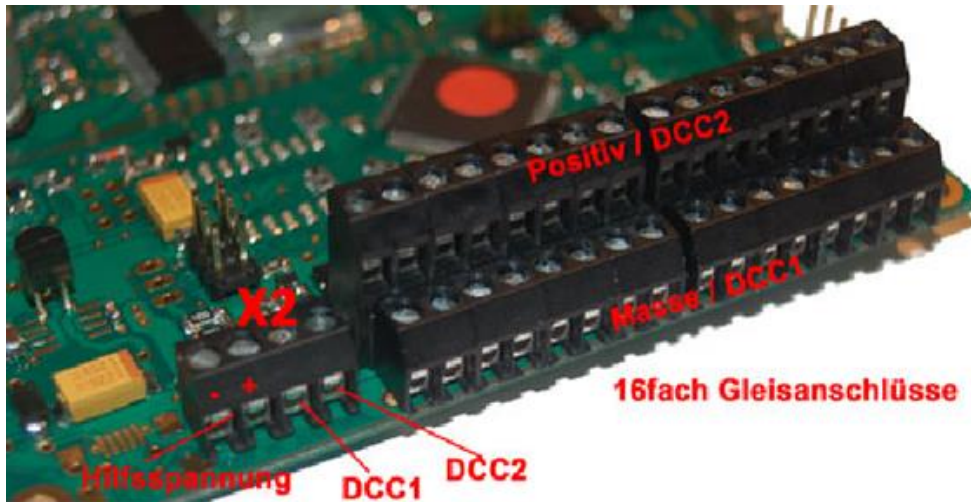


Abbildung 54: Anschlüsse GBM16T

X2 pin1:           5V GND  
 X2 Pin2:           5V Plus  
 X2 Pin3:           DCC1  
 X2 Pin4:           DCC2

X3, X7, X8, X20:   16x Gleisanschlüsse mit oder ohne Masse

## 12.4 GBMboost als BiDiB-Interface

Nach der erfolgreichen Inbetriebnahme des GBMboost als Master (Master-Widerstände), kann der Baustein mit dem PC über USB verbunden werden. **Mit dieser Aktion** und der aufgespielten Master-Firmware wird der GBMboost zum Master-Device am BiDiBus.



**Es darf am BiDiBus nur ein Master-Device vorkommen.**

Alle weiteren GBMboost Module sind Nodes (Slave-Devices) und benötigen die Node-Firmware. Diese Nodes werden mit einem RJ45 Kabel in Reihe mit dem Master GBMboost verkabelt.



**Ein Node darf nicht mit dem PC über ein USB-Kabel verbunden werden!**

Der virtuelle COM-Port wird als weiterer COM-Port eingetragen (also z.B. COM10 oder Com7), die Anwendung wird dann auf diesen COM-Port eingestellt.

Am Beispiel des **BiDiB-Wizard Tool**, muss in den Einstellungen des Programms der passende COM-Port gewählt werden.

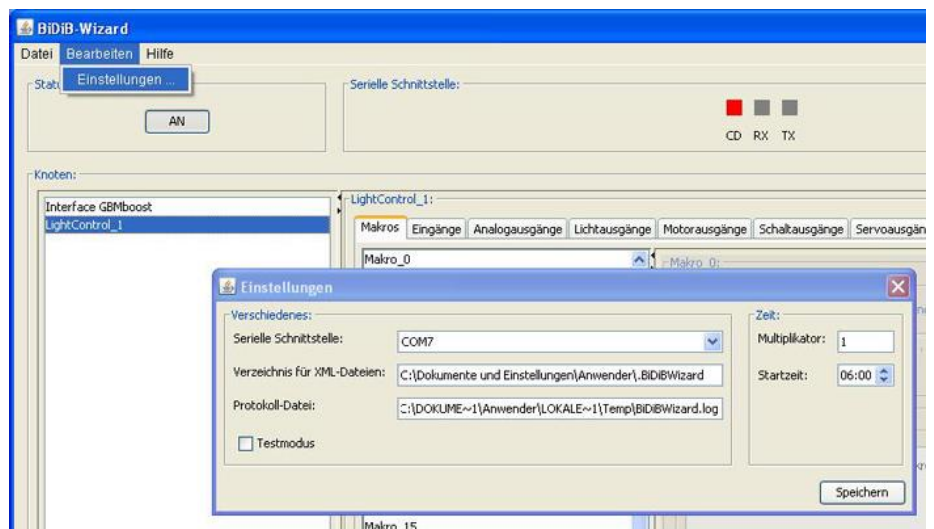


Abbildung 55: GBMboost als BiDiB-Interface / BiDiB-Wizard

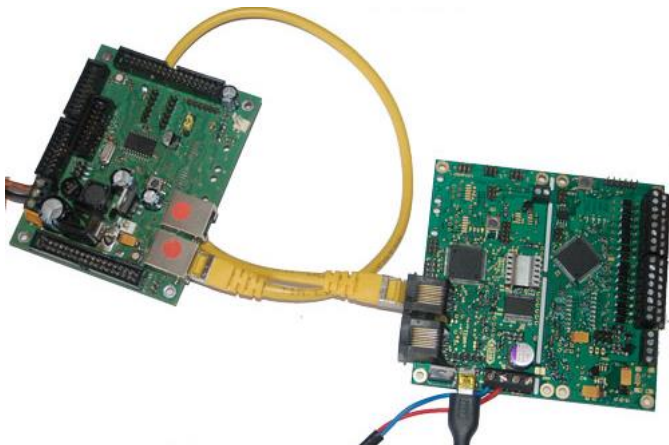


Abbildung 56: GBMboost als BiDiB-Interface / Anschluss

Der **GBMboost** ist jetzt bereit für eine Verbindung mit einer BiDiB-Baugruppe.

Wenn Sie eine BiDiB-Baugruppe mit dem GBMboost verbinden erleuchtet auf dem anzuschließenden BiDiB-Module sowie auf dem GBMboost die BiDiB-Status LED.

Die **Abbildung 56** zeigt die **LightControl** an dem **GBMboost**. Die erfolgreiche Verbindung über den **BiDiBus** wird auch in der Knotenliste im **BiDiB-Wizard Tool** in der **Abbildung 55** angezeigt.

## 13. Terminierung des BiDiBus

Der **BiDiBus** besteht aus einer RS485-2 Draht Verbindung die speziell für Hochgeschwindigkeits-Datenübertragungen über große Entfernungen entwickelt worden ist und eine zunehmende Verbreitung in industriellen Anwendungen findet. Dank diesen Eigenschaften kann eine Kabellänge von über 500 Meter mit hohen Datenübertragungsraten realisiert werden.

Um bei diesen hohen Übertragungsgeschwindigkeiten und Kabellängen noch eine fehlerfreie Kommunikation zu gewährleisten ist eine Terminierung des BiDiBusses notwendig um Reflexionen zu verhindern.

Man spricht auch davon, dass man den Kommunikationsbus abschließen muss.

(RT1 in der Abbildung)

**Bei kurzen Kabellängen unter 5m kann auf die Terminierung verzichtet werden.**

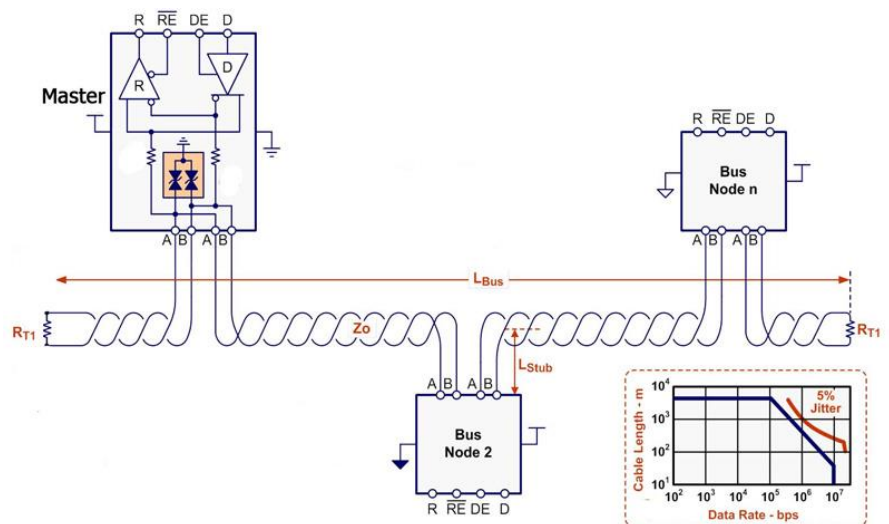


Abbildung 57: technische Darstellung einer Terminierung

### 13.1 Vorgehensweise der Terminierung

Der BiDiBus muss an jedem Ende (vorne und hinten) mit einem Widerstand mit dem Wert von 120 Ohm abgeschlossen werden.

Das bedeutet, dass jeder Knoten der am Ende des BiDiBusses (auch bei einem verzweigten Bussystem) seinen Platz findet, ein Widerstand zwischen **BiDiB\_A** und **BiDiB\_B** erhalten sollte.

Auf allen unseren BiDiB-Knoten (GBM, LightControl, BiDiBone, s88-BiDiB-Interface....) haben wir das schon vorgesehen und somit für die Anwender vereinfacht. Der Anwender muss nur am letzten Knoten im Bus den Terminierungsjumper stecken.

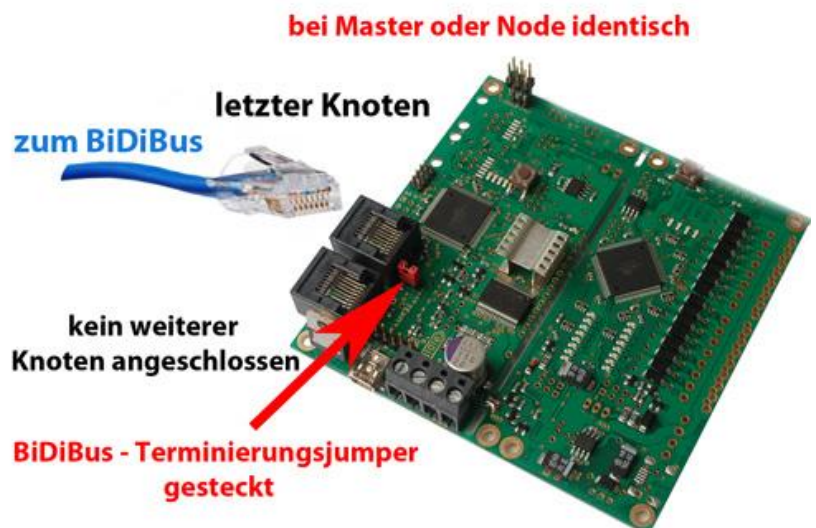


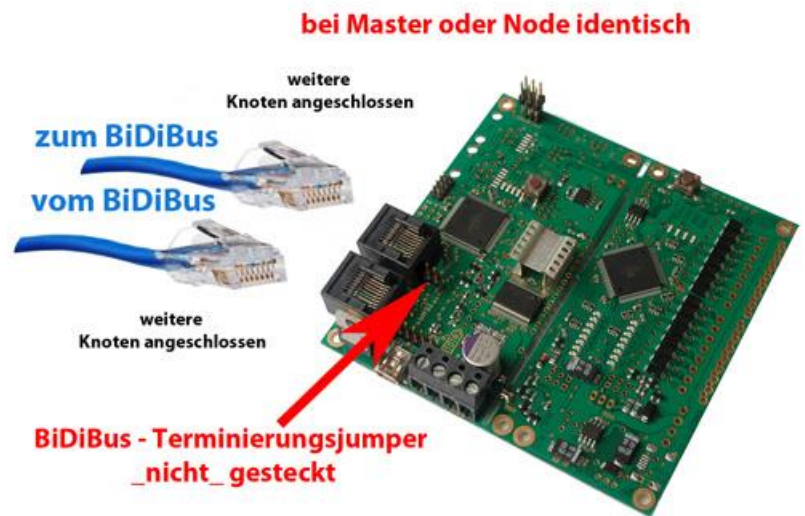
Abbildung 58: Terminierung am GBMboost

## Hinweis:

Der Jumper darf nur am letzten BiDiB-Knoten im BiDiBus (Anfang und Ende der kompletten Busleitung) gesteckt werden. Besteht der BiDiBus aus einem verzweigten Bus (Stichleitungen) mit Hilfe von BiDiBus Erweiterungsmodulen (Hubs), dann muss der Bus an allen offenen Enden abgeschlossen werden.

Befindet sich der Knoten im Bus und an beiden Enden sind weitere Knoten angeschlossen, dann wird der Terminierungsjumper nicht gesteckt.

**Die Terminierung ist unabhängig ob es sich um einen GBMboost Master oder GBMboost Node handelt. Der GBMboost Master kann sich am Anfang, am Ende oder mitten im BiDiBus befinden.**



## 14. Wissenswertes zur Unique-ID



Die **Abbildung 59** zeigt eine **Unique-ID** auf einem BiDiBus Device.  
Alle SMD bestückten BiDiB-Baugruppen die über den Fichtelbahn-Shop bezogen werden und am BiDiBus hängen werden mit einer aufgespielten BiDiB-Seriennummer ausgeliefert. Diese Seriennummer ist ein Bestandteil der Unique-ID und deshalb wird auch auf der Rückseite des Bausteins diese komplette Unique-ID abgedruckt.

**Abbildung 59: Unique ID auf Baugruppe**

### Was ist eine Unique-ID?

Die vom Hersteller in dem Baustein fest programmierte, eindeutige Kennung, bestehend aus 16 Bit Herstellerkennung und 32 Bit herstellerspezifischer Nummer (z.B. Produktindex und Seriennummer).

**V** = VID (Herstellerkennung)  
**OD** = Selbstbauprojekte  
**P** = PID (Produktkennung)  
**6800** = GBMboost Master  
**6700** = GBMboost Node  
**6B00** = LightControl  
 0029 wird zu **2900** = Seriennummer

Die Angaben der Unique-ID sind in HEX.

### Wozu dient die Unique-ID?

Die Unique-ID ist eine absolut einmalige Nummer, mit dieser Nummer wird eine Baugruppe unabhängig von ihrem Einbauort und ihrem Platz am Bus gefunden.

Das heißt: Das BiDiB-System führt ein Art 'Telefonbuch', unter welchem Anschluss welche Baugruppe erreicht werden kann. Das Hostprogramm vergibt dann Namen für die einzelnen Anschlüsse. Die Unique-ID ist hier das Verbindungsglied zwischen der Bezeichnung am PC und der Baugruppe.

### Beispiel:

Ein BiDiBone ist unter dem Bahnhof verbaut und hat die Unique-ID **0D 6B001234**.  
 Im Hostprogramm wird sie als HBF-West bezeichnet. Das BiDiB System meldet dem Hostprogramm: 0D6B001234 findest Du unter Anschluss 3. Wenn man jetzt in HBF-West was umstellt, guckt das Hostprogramm im Telefonbuch nach: aha, ich soll 3 anrufen.

**Somit hat man nichts mehr mit Adressen und Dipschalter zu tun.**

## Warum sollte man diese Nummer aufkleben? Für was könnte man diese Angabe gebrauchen?

Alle Tools aber auch die aktuellen PC-Steuerprogramme kommunizieren und verwalten die Bausteine über diese Nummer. Es gibt jetzt keine "DCC - Adresse XX" mehr, sondern der Befehl wird an den Knoten \*Unique-ID X\* und dessen Port gesendet.

Der Umkehrschluss ist, dass für die Konfiguration einer neuen Aktion an dem Ausgang eines Decoders (z.B. LightControl) oder der Zuordnung von einem Melder im Gleisbild (z.B. GBM), die passende Unique-ID des BiDiB-Bausteins dem Programm mitgeteilt werden muss.

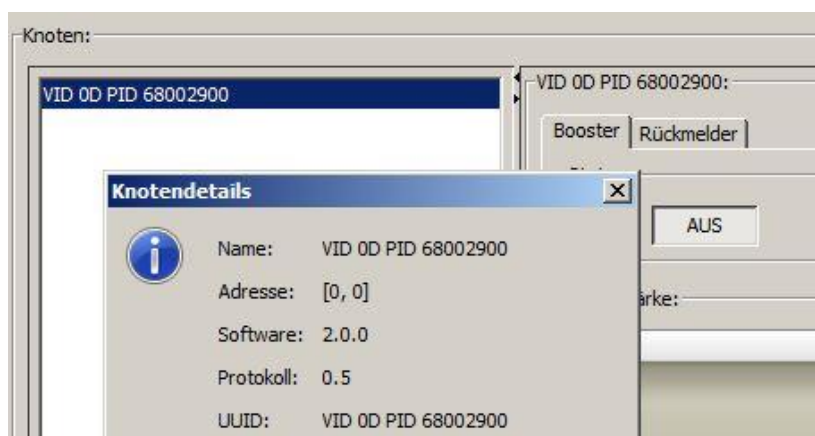


Abbildung 60: Unique ID im BiDiB-Tool

### Hardware-Seriennummer:

Die aufgeklebte Seriennummer mit S/N auf der Rückseite der Baugruppe ist eine Hardware-Seriennummer und hat nichts mit der Unique-ID Seriennummer für die BiDiB-Baugruppen zu tun. Mit dieser Nummer ist Ihr Baustein bei uns registriert.

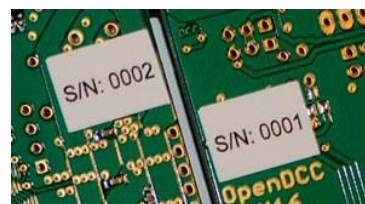


Abbildung 61: Hardware S/N auf Baugruppe

### Wo wird die Unique-ID auf dem Baustein gespeichert?

Das Verfahren hier ist etwas zweigeteilt. In der Flash/EEPROM-Firmware Datei ist die VID und PID bereits integriert und wird auf den Baustein übermittelt, jedoch ist noch keine Seriennummer enthalten. Die **Firmware Dateien** können daher problemlos auf den Prozessor geladen werden oder auch **aktualisiert** werden. Eine bereits vorhandene **Seriennummer wird dabei nicht überschrieben**. Wenn der Baustein beim Starten noch keine geladene Seriennummer erkennt, blinkt ein Fehlercode und der Betrieb wird blockiert. **Ohne Seriennummer läuft die Firmware nicht!**

*(Ausnahme beim GBMboost Master: hier wird eine Notfall-S/N erzeugt mit der Kennung: 0100. Diese S/N kann jederzeit mit dem Nachschieben der gültigen S/N überschrieben werden)*

Die Seriennummer wird immer als dritter Akt der Inbetriebnahme nachgeschoben und landet als erstes im EEPROM des Bausteins.

## **Bei jedem Neustart führt der Baustein folgende Überprüfung aus:**

Er schaut in dem User Signatur Bereich, ob schon eine Seriennummer vorhanden ist. Ist keine Seriennummer vorhanden wird die Seriennummer aus dem EEPROM hier abgelegt und der Baustein startet seinen Betrieb. Ist im Bereich der USER Signatur schon eine Seriennummer vorhanden wird die neuaufgespielte Seriennummer vom EEPROM verworfen und die schon hinterlegte Seriennummer aus dem User Signatur Bereich verwendet.

Die hinterlegte Seriennummer im USER Signatur Bereich bleibt auch erhalten bei einem CHIP ERASE und kann nur gelöscht werden mit einem USER Signatur ERASE.

Falls man die Seriennummer auf einem BiDiBus tauglichen Device tauschen müsse, dann funktioniert das nur über ein vorheriges USER Signatur ERASE.

Das hat aber auch zur Folge, dass vorhandene Kalibrierungswerte verloren gehen, weil diese ebenfalls in diesem Bereich gesichert werden.

**Dies sollte nicht ohne speziellen Grund durchgeführt werden.**

## **Ich habe aus Versehen die aufgedruckte Seriennummer gelöscht.**

### **Wie kann diese wieder aufgespielt werden?**

1. Man zieht eine neue Seriennummer über den Generator, spielt diese nach dem Löschen des USER Signatur Bereiches auf das BiDiBus Device.

**Wichtig:** Jetzt nicht vergessen den bestehenden Aufkleber durch die neue Unique-ID zu ersetzen. Link zum Generator: [http://www.opendcc.de/elektronik/bidib/opendcc\\_bidib.html](http://www.opendcc.de/elektronik/bidib/opendcc_bidib.html)

2. Man möchte die vorhandene (aufgedruckte) Unique-ID weiterverwenden, weil diese schon in den Gleisbildern meines PC-Programms hinterlegt wurde.

In diesem Fall kontaktieren Sie den Fichtelbahn-Support **mit der Angabe der aufgedruckten Unique-ID Nummer und der zugehörigen Hardware-Seriennummer.**

Dies funktioniert nur bei den SMD-vorbestückten Bausteinen! Bei eigenständig gezogenen Seriennummern müssen Sie in der Historie Ihrer gezogenen Seriennummern im Generator, nach der passenden Nummer suchen.

Für Verbesserungsvorschläge und Hinweise auf Fehler sind wir sehr dankbar.

Auf die Bauanleitung und Software gibt es keine Haftung auf Schäden oder eine Funktionsgarantie. Ich hafte nicht für Schäden, die der Anwender oder Dritte durch die Verwendung der Software oder Hardware verursachen oder erleiden. In keinem Fall hafte ich für entgangenen Umsatz oder Gewinn oder sonstige Vermögensschäden die bei der Verwendung oder durch die Verwendung dieser Programme oder Anleitungen entstehen können.

**Bei Rückfragen steht Ihnen unser Support-Forum gerne zur Verfügung!**  
([www.opendcc.de](http://www.opendcc.de))

## Kontakt:

**FichtelBahn**  
Christoph Schörner  
Am Dummersberg 26  
D-91220 Schnaittach

[support@fichtelbahn.de](mailto:support@fichtelbahn.de)



### © 2019 Fichtelbahn

Alle Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung vorbehalten.  
Vervielfältigungen und Reproduktionen in jeglicher Form bedürfen der schriftlichen Genehmigung durch Fichtelbahn.  
Technische Änderungen vorbehalten.

Rechteinhaber:  
Autor:  
Bilder/Grafik:

© Wolfgang Kufer, Mühldorf  
Christoph Schörner, Schnaittach  
Christoph Schörner, Schnaittach