

## Bauanleitung

### BiDiB – S88 – Bridge V2.0

# OpenDCC

## Inhaltsverzeichnis:

Seite:

Titelseite.....	1
Inhaltsverzeichnis.....	2
Einleitung.....	3
Sicherheitshinweise, bestimmungsgemäßer Gebrauch.....	4
Werkzeuge und Arbeitsmittel.....	5
Vorbereitung.....	6
Aufbau.....	7
– Funktion	
– Platine	
– Schaltplan und Layouts	
– Bauteileliste	
– Aufbau in 5 Schritten	
Inbetriebnahme.....	15
Programmierung.....	16
Einstellungen.....	20
Einrichten am BiDiB.....	21
Anhang.....	23

# OpenDCC

## Einleitung:

Diese Bauanleitung beschreibt den Aufbau und Inbetriebnahme der S88-BiDiB-Bridge aus der Selbstbaureihe von OpenDCC.

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 versierte und interessierte Modellbahner ausschließlich für den Eigenbau.

Diese Anleitung wurde sorgfältig geprüft und nach bestem Wissen erstellt. Es kann kein Anspruch auf Vollständigkeit, Aktualität und Richtigkeit erhoben werden. Sollten Handelsnamen oder geschützte Bezeichnungen verwendet werden, so liegen alle Rechte beim Rechteinhaber.

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

Die hier verwendete und teilweise beschriebene Software kann auf der Internetseite von [www.opendcc.de](http://www.opendcc.de) als Download benutzt, erweitert und verbessert werden.

Alles Weitere zur Nutzung von Software, Hardware und Applikation, ist auf der Internetseite von OpenDcc beschrieben. Der Nutzer und Anwender erklärt sich mit den dort beschriebenen Regelungen vorbehaltlos einverstanden.

### **Eine kommerzielle Nutzung der Software oder Teile daraus ist nicht statthaft!**

Diese Bauanleitung darf keiner anderen Nutzung zugeführt werden, außer der bestimmungsgemäßen Anwendung zum Aufbau der S88-BiDiB-Bridge. Anderweitige Nutzung erfordert die Zustimmung des Autors, bzw. des Rechteinhabers der Internetseite [www.opendcc.de](http://www.opendcc.de).

Rechteinhaber: © Wolfgang Kufer, Mühldorf  
Autor : Alfred Ostertag, Schwäbisch Gmünd

# OpenDCC

## **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 PC's. 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 Schalten und Melden eingesetzt zu werden.

**Jeder andere Gebrauch ist nicht bestimmungsgemäß.**

Das Modul ist nicht dafür bestimmt von Kindern unter 15 Jahren zusammengebaut, eingebaut oder betrieben zu werden.

# OpenDCC

## Werkzeuge und Arbeitsmittel:

Sie benötigen:

- kleiner Seitenschneider
- Pinzette für kleinste Teile
- Lötzinn 0,5 oder 0,3 mm Durchmesser
- ggf. Flußmittel
- Reinigungsmittel, Pinsel, 100% Isopropylalkohol
- Löt- und Lötlösung
- Lupenleuchte, besser Kameramikroskop
- LötKolben 30 Watt, oder besser eine thermisch geregelte Lötstation.
- Lötspitzen mit Eignung zum Verarbeiten von SMD-Bauteilen.

Keine Angst vor den kleinen SMD - Teilen, Microcontroller und IC´s.

Durchhaltevermögen und eine ruhige Hand.....

### **Tipp:**

schauen Sie sich ein kleines Video an, wie die „Tausendfüßler“ von Atmel problemlos von Hand gelötet werden: [www.fichtelbahn.de](http://www.fichtelbahn.de) .

# OpenDCC

## **Vorbereitung:**

Nichts geht ohne systematisches Arbeiten.

Arbeitsmittel und Werkzeuge parat, Lötspitzen sauber ?

Nehmen Sie die zum Modul gehörigen Platinen. Prüfen Sie diese optisch unter Zuhilfenahme einer Lupenleuchte auf Fehler :

Leiterbahnen, Durchkontaktierungen, Bohrungen usw.

Reinigen Sie mit dem Isopropylalkohol die Oberflächen von Fett und sonst. Rückständen.

Prüfen Sie die erforderlichen Bauteile auf Vollständigkeit.

Planen Sie die einzelnen Arbeitsschritte und stellen Sie für die einzelnen Schritte die benötigten Bauteile bereit.

Soweit fertig ?

Na dann los !

# OpenDCC

## Aufbau:

### Funktion:

Was bauen wir den da zusammen ?

Um die bisher verwendeten S88 Rückmelder in das BiDiB -System funktional integrieren zu können wurde dieses Modul entwickelt.

Das Modul dient als Gateway. Die im Bestand vorhandenen S88-Melder können somit weiter verwendet werden.

Der im Konzept weiterverfolgte Aufbau mit S88-N Standard's erlaubt die Übertragung von RAILDATA über das Buffer IC U\$3 auf genormte Schnittstellen.

Weitergehende Informationen darüber erhalten Sie auf der Internetseite von [www.opendcc.de/info/s88-n/](http://www.opendcc.de/info/s88-n/)

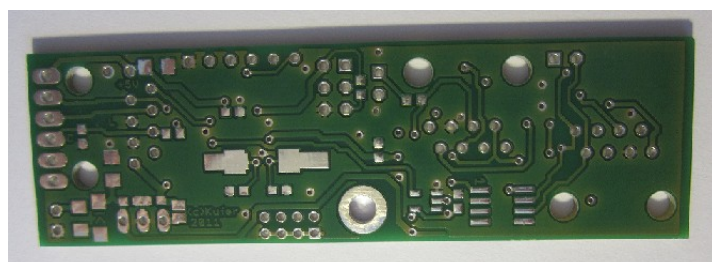
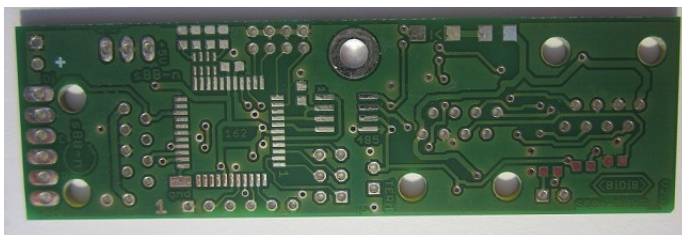
Die verschiedenen Anschluß- und Anwendungsmöglichkeiten werden in einer separaten Applikationsdokumentation ausführlich beschrieben.

### Eigenschaften:

- Ein S88-Strang, bis zu 16 Module zu je 8 Meldern.
- Automatisches Anmelden am BiDiBus
- Quittungsmode Secure-ACK implementiert (keine verlorenen Meldungen)
- Optional Bereitstellen des RAILDATA-Signals für S88-N.
- Anzeige des aktuellen Zustands (Power, Verbindung, Nachricht) mit LEDs
- Einstellbare Geschwindigkeit als S88-Busmaster
- Konfigurierbar per Jumper oder per USB (über FTDI-Kabel).
- S88-Anschluß: Dieser ist als Steckleiste oder S88-N ausgeführt.

### Platine:

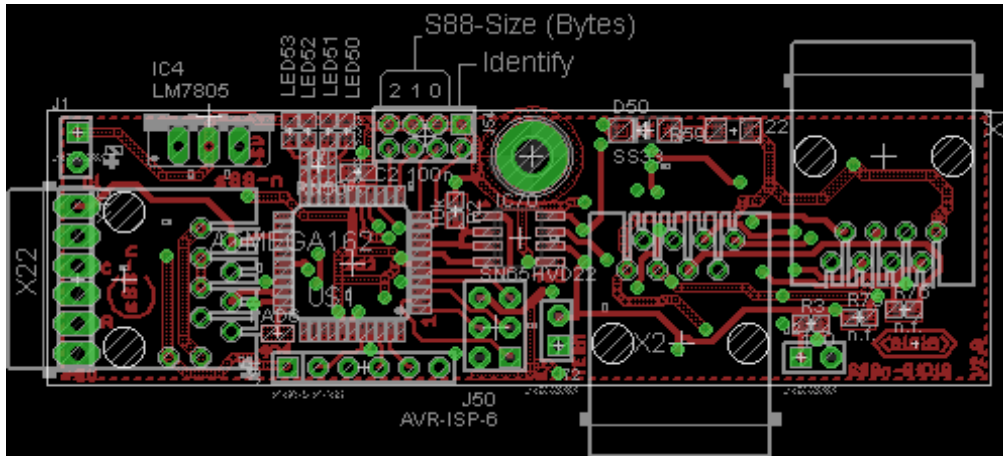
Ansicht der Ober- und Unterseite der Platine V2.0



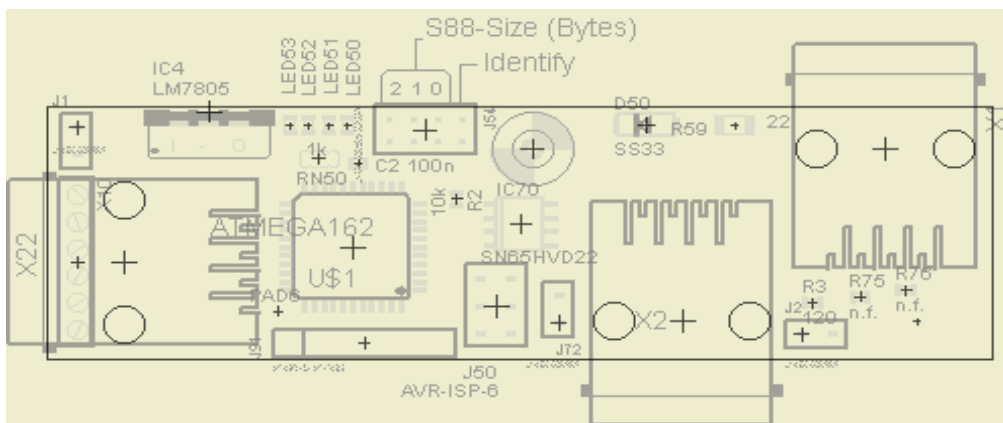
# OpenDCC

## Schaltplan, Layout und Bestückung:

Alle hier veröffentlichten Layouts und Schaltpläne können Sie auf der Internetseite downloaden.



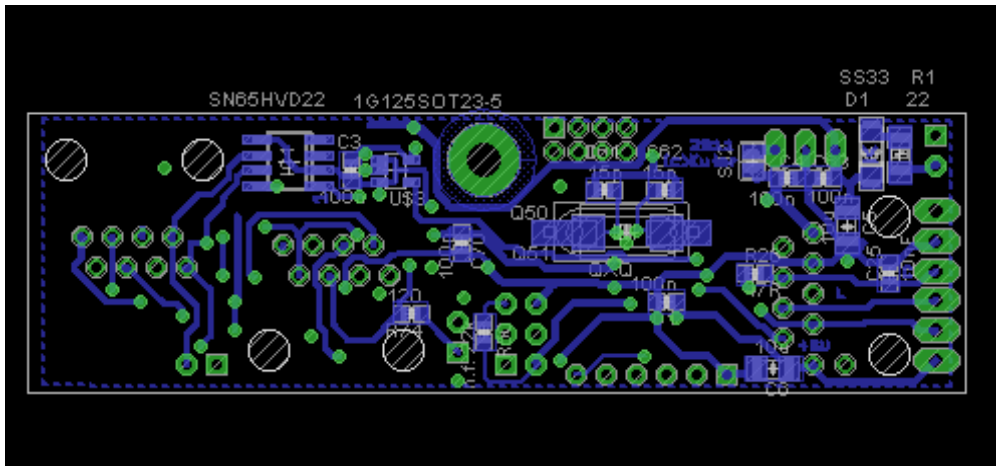
## Placement Bestückungsseite



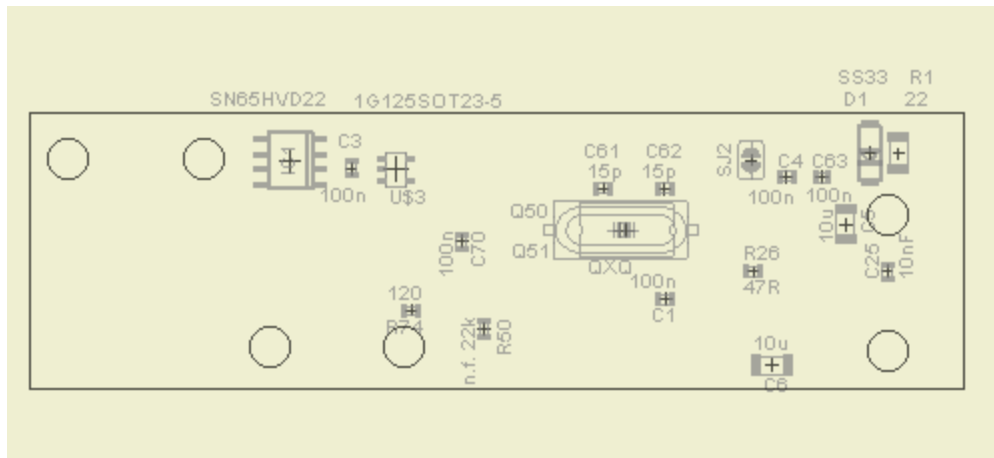


# OpenDCC

## Lötseite

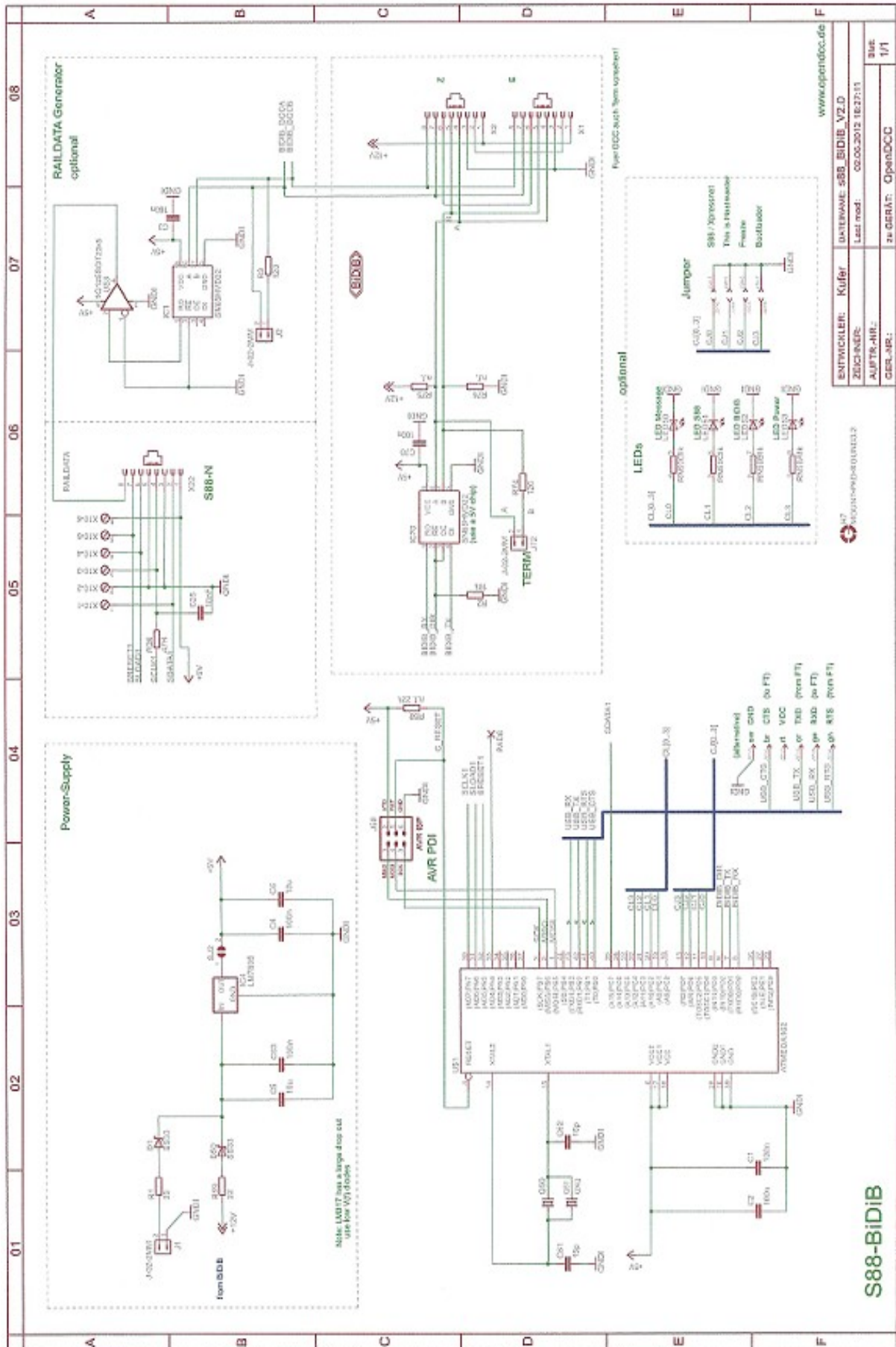


## Placement Lötseite



# OpenDCC

## Schaltplan:



# OpenDCC

## Bauteileliste:

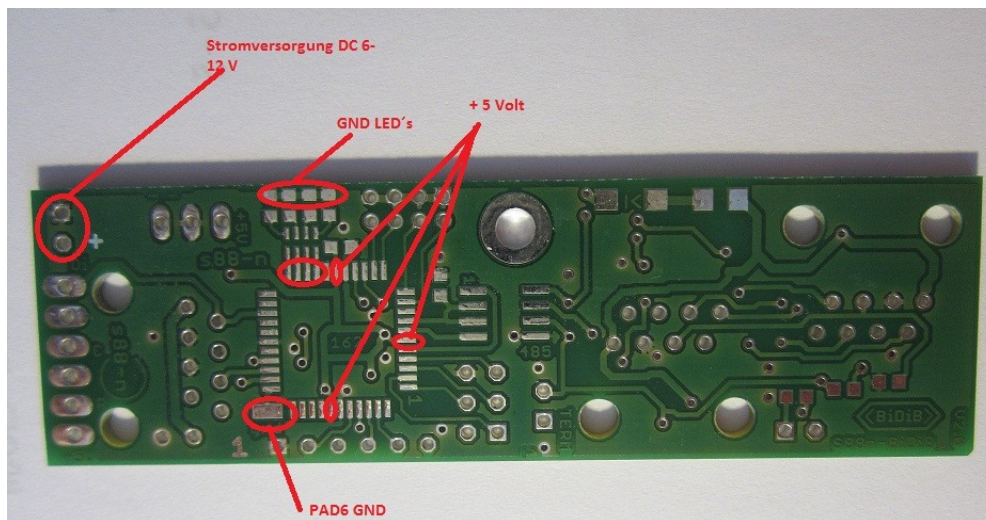
Part	Value	Package	Bezeichnung
C1	100 nF	0603	Kondensator X7R
C2	100 nF	0603	Kondensator X7R
C3	100 nF	0603	Kondensator X7R
C4	100 nF	0603	Kondensator X7R
C5	10 µF	0603	Kondensator X7R
C6	10 µF	0603	Kondensator X7R
C25	10 nF	0603	Kondensator
C61	15 pF	0603	Kondensator
C62	15 pF	0603	Kondensator
C63	100 nF	0603	Kondensator X7R
C70	100 nF	0603	Kondensator X7R
D1	SS33	DO214AC	Schottky Diode
D50	SS33	DO214AC	Schottky Diode
IC1	SN65HVD22D	SOIC8	Linear Technology RS 485
IC4	LM7805	TO 220	Spannungsregler
IC70	SN65HVD22D	SOIC8	Linear Technology RS 485
J1	HDR 100	Header - JP	Stiftkontakteleiste
J2	HDR 100	Header - JP	Stiftkontakteleiste
J50	HDR 100	Header - JP	Stiftkontakteleiste
J54	HDR 100	Header - JP	Stiftkontakteleiste
J72	HDR 100	Header - JP	Stiftkontakteleiste
J91	HDR 100	Header - JP	Stiftkontakteleiste
LED50	Low current 2 mA	0603	Orange ( Message )
LED51	Low current 2 mA	0603	Grün ( S88 )
LED52	Low current 2 mA	0603	Grün ( BiDiB )
LED53	Low current 2 mA	0603	Grün ( Power )
Q50	HC49UP 16MHz	SMD	Quarz
R1	22	1206	
R2	10k	0603	
R3	120	0603	
R26	47	0603	
R50	22k	0603	Not fitted
R59	22	1206	
R75	nn	0603	Not fitted
R76	nn	0603	Not fitted
RN50	1k	1206	Widerstandsnetzwerk
U\$1	Atmega 162	TQFP44	
U\$3	SN74LVC1G125	SOT 23-5	Single-gate-buffer
X1	AMP Con	RJ 45 8/8	520251-4
X2	AMP Con	RJ 45 8/8	520251-4
X22	AMP Con	RJ 45 8/8	520251-4

# OpenDCC

## Schritt 1 Stromversorgung:

Die zuvor mit Alkohol gereinigte und geprüfte Platine bekommt nun die ersten Bauelemente.

Beginnen wir auf der Bestückungsseite mit den Kondensator C2, den Widerständen , R3, R56. Die Diode D50 und den Längsregler IC4 LM7805. Bei der Diode auf richtige Einbaulage achten, die Markierung – ein Strich an der Gehäuseseite markiert die Anode. Die Stiftleiste J1 wird ebenfalls bestückt.



### Tipp:

Vor einlöten des Längsreglers die SMD-LED's LED50 orange, Led 51 grün, LED 52 grün, LED 53 grün, sowie das Widerstandsnetzwerk RN50 bestücken. Da kommt man später auch wegen des Microcontrollers mehr schwer bei. Bei den LED's auf richtige Polung achten.

Am Besten mit Netzteil 1,8 Volt einstellen und so die Polarität prüfen. Auf der Platine sind die äußeren Löt pads mit GND verbunden.

Auf der **Lötseite** bestücken wir R1, D1, C4, C5, C6 und C63, die Lötbrücke SJ2 wird geschlossen.

Zum Test legen wir an der Stiftleiste J1 mit einem Labornetzteil eine Gleichspannung von 6-12 Volt an. Polarität beachten! Der Pluspol Pin2 ist auf der Platine mit + markiert. Strombegrenzung am Netzteil auf 20 mA begrenzen.

Messen wir jetzt am Löt pad 6 (GND) und den Pin 6,17 und 36 der Löt pads des Microcontrollers 5 Volt, dann haben wir bis jetzt alles richtig gemacht.

Die LED's prüfen wir ebenfalls an den Pin 19, 20, 21 und 22 des Microcontrollers bei eingebautem RN50 durch anlegen von + 5 Volt, GND J1 Pin1.

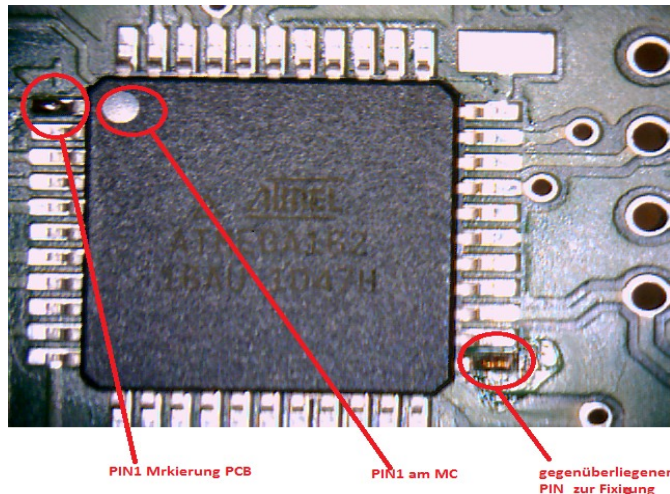
# OpenDCC

## Schritt 2 Einbau des Microcontrollers:

Nun kommt der Atmega 162 auf die Platine.

Auf richtige Positionierung von PIN 1 achten, dieser ist als kleiner Punkt auf der Oberseite des Gehäuses markiert.

Ein Lötpad mit wenig Lötzinn versehen, den Käfer lagerichtig auflegen und verlöten. Jetzt kann noch korrigiert werden. Gegenüberliegendes Lötpad mit dem MC-PIN verlöten – schon ist alles fixiert.



Unter Zugabe von Flußmittel und wenig Lötzinn eine Seite nach der Anderen verlöten, dabei den LötKolben zügig von einer Seite zur Anderen ziehen.

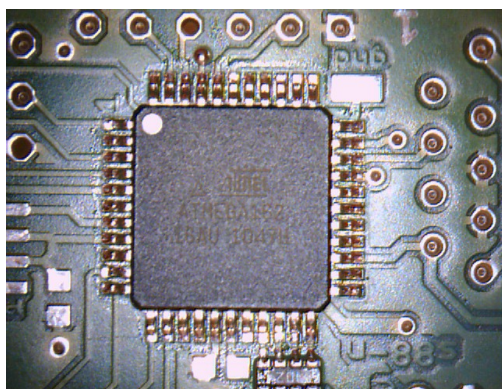
Das sieht schwieriger aus als es den Anschein hat. Entschlossen handeln.

Auf Kurzschlüsse achten, diese ggf. mit der Löt-sauglitze entfernen.

Nicht zu lange an einzelnen Pin's rumbruzeln, sonst stirbt der Controller am Hitzschlag ! Für genügend Auskühlzeit sorgen.

Lötreste und Flußmittel mit Alkohol reinigen.

Eine gute Film - Anleitung zum Löten von MC finden Sie bei [www.fichtelbahn.de](http://www.fichtelbahn.de)

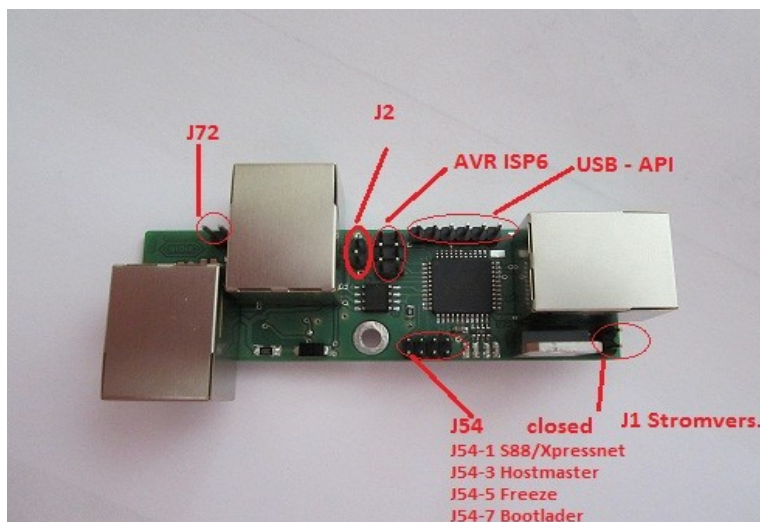


Fertig verlöteter Atmel.

# OpenDCC

## Schritt 3 restliche Bauteile auf Bestückungsseite:

Nun werden die restlichen passiven Bauelemente R2, R3 eingebaut, R75 und 76 werden nicht bestückt. Bei IC70 auf richtige Einbaulage achten. Die Stiftleisten J1, J2 für den Netzwerkschluß, J50 für das Programmierinterface AVR-ISP6, J54 für die Betriebsartenwahl, J72 für den BIDIB Busanschluß einbauen. Für spätere Optionen J91 als USB Port für Updates und Kommunikation mit der in der Software enthaltenen API, auf die mittels geeignetem FTDI Kabel zugegriffen werden kann, um Einstellungen z.B. CV's zu verändern.



Bezeichnung der externen Anschlüsse und Jumper

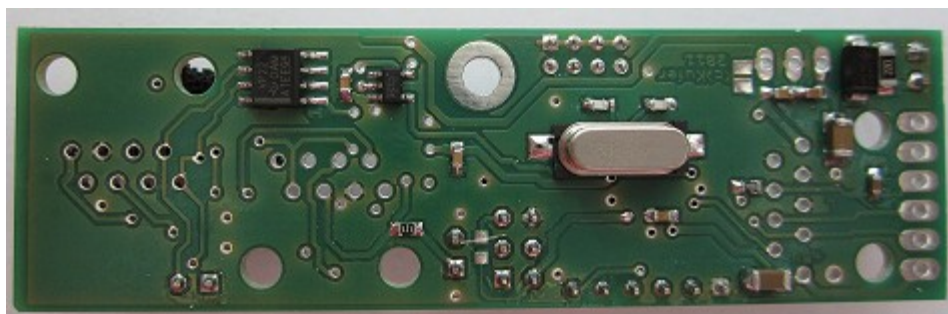
Zum Abschluß der Baustelle oben wird die Platine gründlich mit Pinsel und Alkohol gereinigt.

Alle Bauteile nochmals abschließend gründlich auf Fehler kontrollieren.

## Schritt 4 restliche Bauteile auf der Lötseite:

Hier wiederum erst die passiven Bauteile aufbringen, R50 wird nicht bestückt. Anschließend das Quarz Q50 mit 16 MHz, und nun IC1, den RS 485 Treiber SN65HVD22D. Auf richtige Positionierung von PIN 1 achten. IC U\$3 folgt als Nächstes, dieser Bustreiber stellt die Option RAILDATA zur Verfügung.

Jetzt nochmals alles sorgfältig prüfen und die Unterseite ebenfalls gründlich von Lötückständen befreien.



Ansicht Lötseite bestückt

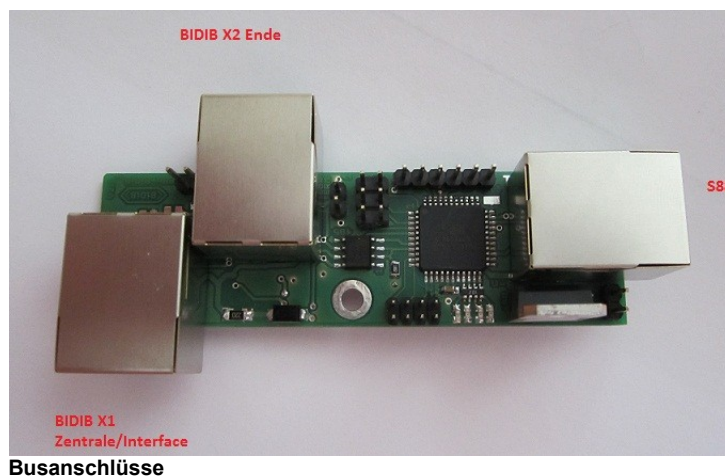
# OpenDCC

## Schritt 5 Anschlüsse und Anzeigen

Im letzten Schritt werden die beiden RJ 45 Buchsen X1 und X2 für den BiDiBus eingelötet. Die Kontaktausbildung ist hier 8/8, also mit 8 PIN Vollbestückung. Für die S88-Anbindung gibt es 2 Alternativen:

- als S88n mit RJ45 Buchse wie für den BiDiBus X22
- oder für Xpressnet

In diesem Fall muss dann anstelle der RJ45 Buchse eine 6 polige Stiftleiste verwendet werden. Diese Ausführung ist jedoch optional.



## Die Betriebsanzeigen

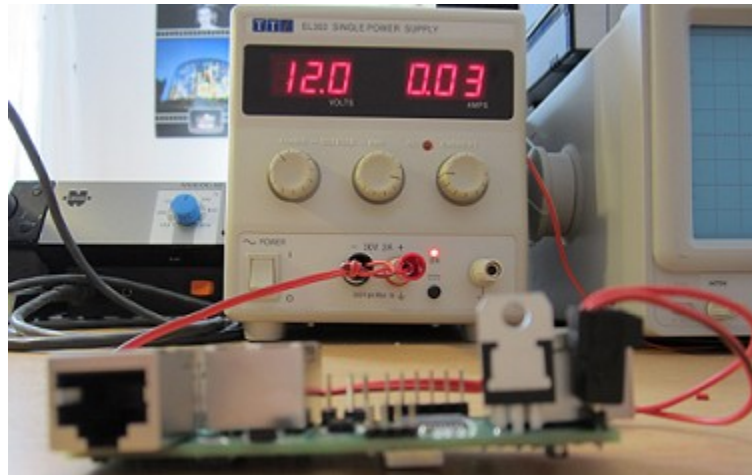
- LED53, grün, Power leuchtend: alles okay, schnell blinkend: Identify-Mode
- LED52, grün, BIDIB dauern leuchtend: Modul ist am Bus verbunden.  
aus: Keine Busverbindung möglich.
- LED51, grün, S88
- LED50, orange, MESSAGE: LED leuchtet bei einer übertragenen Nachricht kurz

# OpenDCC

## Inbetriebnahme:

Legen wir an J2 eine Gleichspannung mit unserem Labornetzteil mit Strombegrenzung 50 mA / 6 Volt an und steigern die Spannung langsam bis auf 12 Volt.

Der maximale Strom sollte bei 12 Volt ca. 27 - 30mA betragen.



Der Längsregler IC 4 LM7805 darf dabei warm werden. Die Temperatur sollte hier jedoch nicht über 65 °C ansteigen (siehe Datenblatt). Das kann sich je nach Anzahl der angeschlossenen S88 Module noch ändern. Ggf. muss hier noch ein Kühlkörper vorgesehen werden. Bei maximaler Last werden hier bis zu 15 Watt Wärmelast abgeführt.

Ist dies nicht der Fall, kein Stromfluss vorhanden dann prüfen Sie ob SJ2 geschlossen ist, andernfalls nochmals die Schaltung prüfen.

Die LED's 50 – 53 sind nicht angesteuert und bleiben dunkel.

## Programmierung:

Die erforderliche Software steht in aktueller Form auf der Internetseite von OpenDCC zum Download bereit.

Was Sie noch benötigen:

Zum programmieren einen Programmer für das Einspielen der Software wie z.B. der AVRISP-MK II mit aktuellem Softwarestand für die ISP Programmierung, sowie das AVR Studio in beliebiger Version (ab V 4.18)

Die gezeigten Screenshots des Programmierverlaufes geben die Version 4 von AVR Studio wieder.

Auf die Installation und Anwendung der Programmierertools kann hier nicht weiter eingegangen werden. Hierzu besuchen Sie die Website von AVR.



# OpenDCC

Weiter benötigen Sie für den BiDiB – Teilnehmer eine eindeutige Seriennummer. Diese könne Sie selbst auf der Internetseite von OpenDCC abrufen. Voraussetzung für das Login sind gültige Anmeldedaten, die identisch zum OpenDCC – Forum sind.

### OpenDCC - BiDiB Seriennummer

**Neue Seriennummer generieren**

Produktkennung

Bemerkung

**Ihre Seriennummern**

Datum	Produkt	Bemerkung	Seriennummer	
02.06.2012, 16:46	BiDiBus-S88-Bridge 1	s88-Bridge_1	:0201fe0000	<a href="#">.eep</a>
14.05.2012, 20:01	LightControl 1, 32 LEDs, 16 Port, 4 Servo	LControl_1	:0207fe0000	<a href="#">.eep</a>

[Abmelden](#)

Selektieren Sie die gewünschte Produktkennung also S88-BiDiB-Bridge

Das genaue Procedere entnehmen Sie der ausführlichen Beschreibung auf der Webseite.

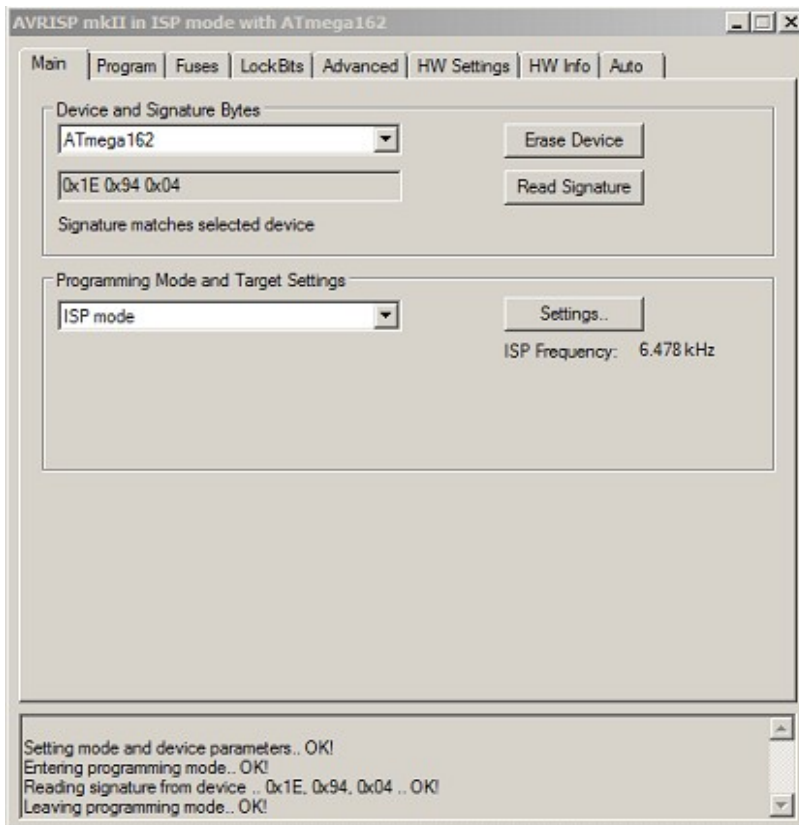
Die erhaltene Seriennummer speichern Sie ab und vermerken diese unverlierbar auf der Hardware.

# OpenDCC

## Einspielen der Software:

Verbinden Sie das Modul mit der Stromversorgung und den Programmieradapter mit der ISP – Stifteleiste auf der Platine, beachten Sie auf richtige Polung (PIN 1)

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

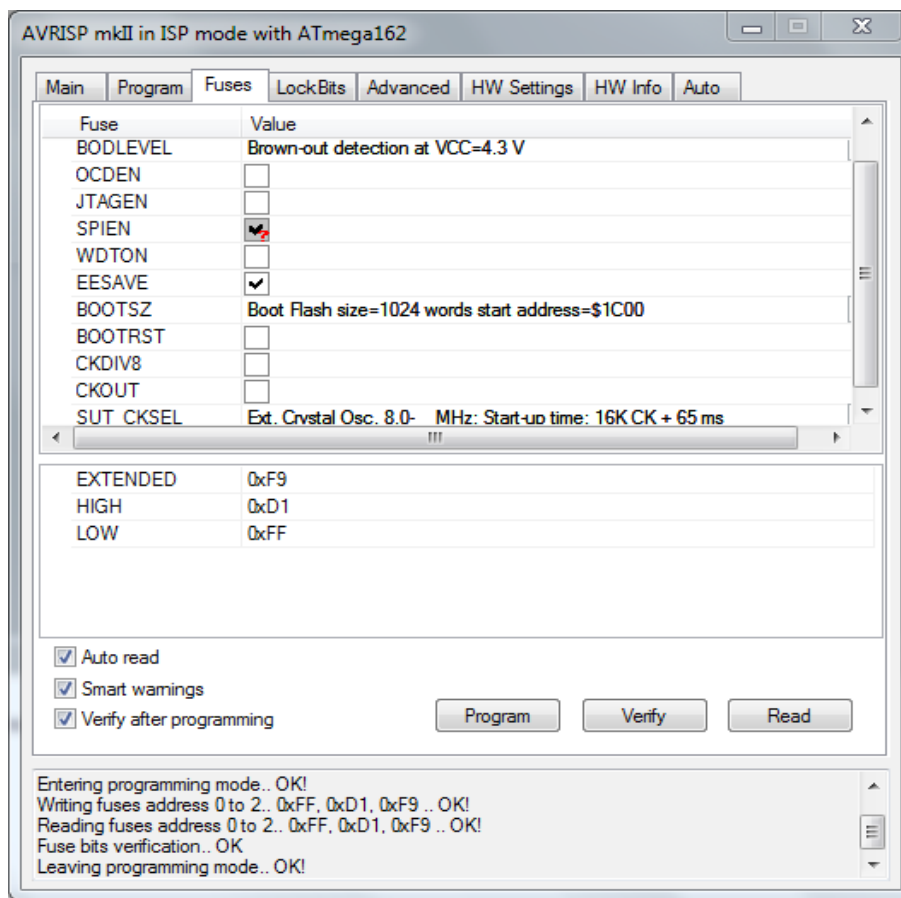


Wählen Sie Atmega162 und lesen Sie die Signatur des Microcontrollers aus.  
Der Programmiermodus ist auf ISP mode einzustellen und die Frequenz z.B. auf 6.478 kHz.

Als Nächstes werden die Fuses eingestellt. Gehen Sie hier sehr sorgfältig mit den Eingaben um. Ein „verfuster“ Atmel lässt sich nur schwer wiederbeleben.  
Ausführliche Informationen zur Einstellung der Fuses auf der OpenDCC Seite.

Das Lowbyte erhält den Wert 0xFF  
Das Highbyte                    0xD1  
Das Extend.Byte                0xF9

# OpenDCC



**Wichtig : JTAG darf nicht gesetzt sein.**

**Haken bei EESAVE wird gesetzt, sonst wird beim Einspielen der Seriennummer der Speicher des EEPROM's überschrieben.**

Ist alles ordnungsgemäß eingestellt, so erhalten Sie OK!

Die erforderlichen Dateien erhalten Sie als gepackte Datei als Download  
Diese Datei in einem eigenen Verzeichnis z.B. mit winZip entpacken.

Für die Programmierung finden Sie im erzeugten Unterordner „Default“  
zwei Dateien mit den Endungen \*.HEX und \*.eep.

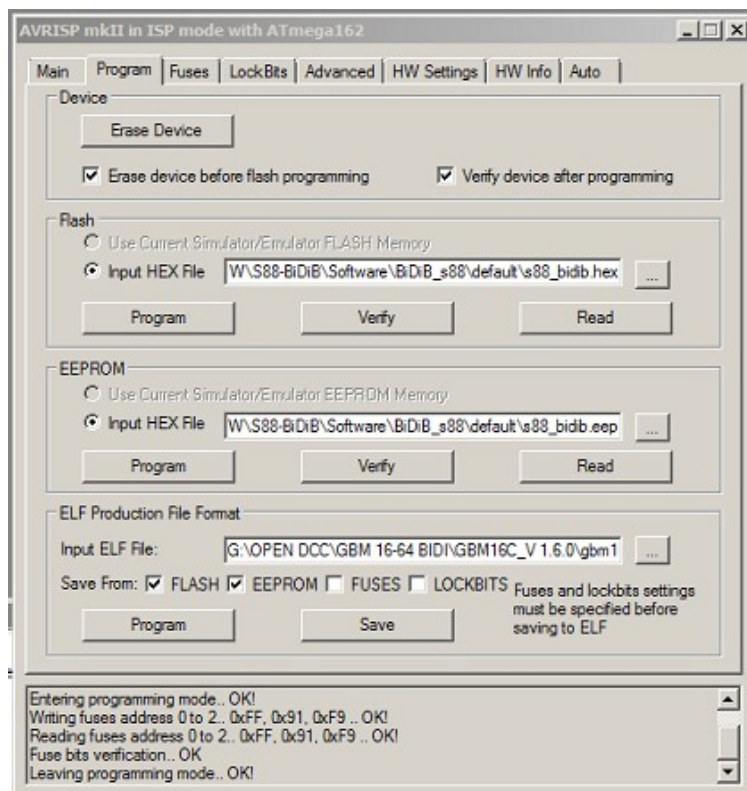
Nur diese Dateien werden in den Flash oder in das EEPROM des Microcontrollers  
geladen, die weiteren Dateien z.B. mit der Bezeichnung Sourcecode sind nicht  
geeignet.

Dann wird die „s88\_bidib.hex“ Datei in den Flash – Speicher eingespielt, die LED's  
53 und 52 blinken jetzt schnell.

In das EEPROM laden wir nun die Datei mit der Benennung „s88\_bidib.eep“.

Noch immer blinken die beiden LED's und warten auf die Seriennummer, die jetzt  
ebenfalls in das EEPROM geladen wird.

# OpenDCC



War das Einspielen der Software erfolgreich, meldet AVR Studio OK!

In diesem Zustand darf nur die grüne LED 53 Power leuchten.  
Wird der BiDiBus verbunden leuchtet die grüne LED 52 BiDiB.  
Die grüne LED 51 signalisiert das Vorhandensein des S88-Busses.  
Die orange LED 50 Message leuchtet kurz bei Datenverkehr auf.

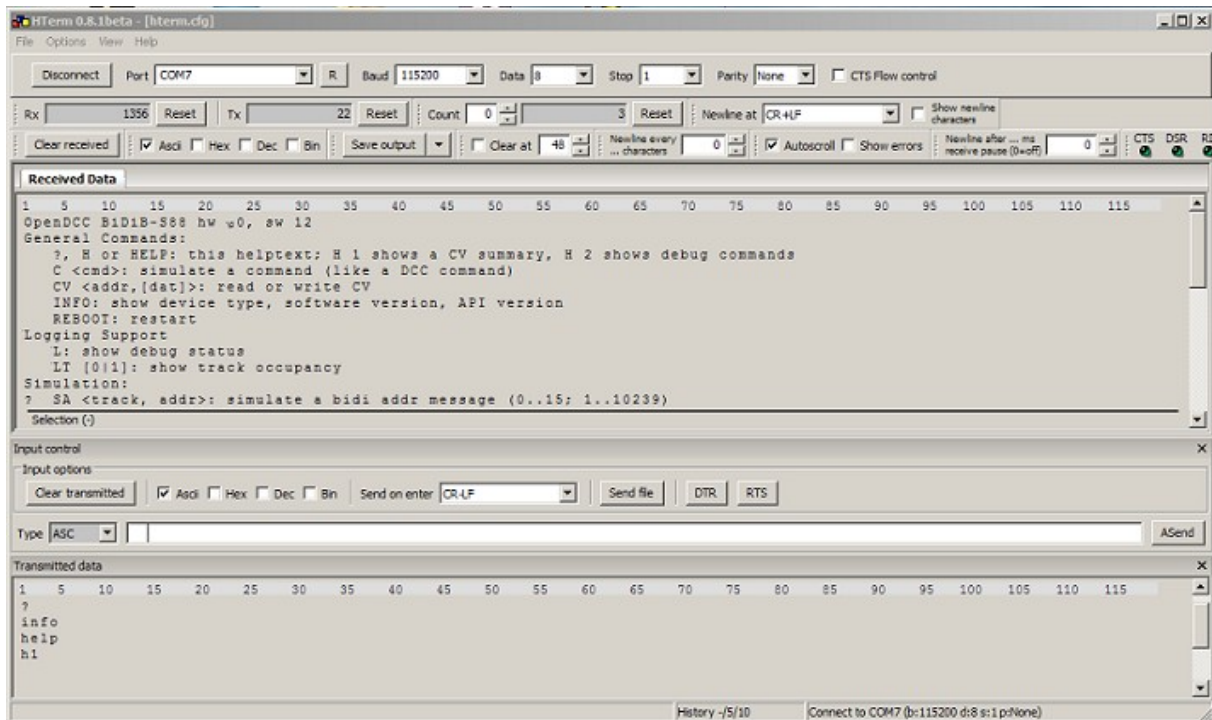
Sind diese Funktionen positiv geprüft ist der Baustein betriebsbereit.

## Einstellungen:

Im normalen Betrieb ist auf dem USB-Port ein Kommando-Interface (API) implementiert, über den mit dem Microcontroller kommuniziert werden kann. Die API arbeitet mit seriellem Protokoll mit den Parametern: 115200 Baud, 8 Bit, keine Parity, ein Stoppbit, (8N1).

Über den USB – Port J91 kann mittels eines FTDI - Kabels und eines geeigneten Terminalprogrammes z.B. Hterm.exe, der Betriebszustand des Prozessors ausgelesen werden, sowie verschiedene Einstellungen an den CV's vorgenommen werden.

# OpenDCC



API – Befehle:

Aufrufen mit Help, H oder ?.  
Danach einfach den Helptext lesen.

## Einstellungen am Jumper J 54:

Am J 54 können die Bereiche der einzelnen zuweisbaren S88 – Stänge definiert werden, oder der Zugriff auf die API ermöglicht.

S88-Sizes ( Bytes )

2	1	0	Identify
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<b>Taster: Identify, API</b>
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	s88-Size: 8 Bit
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	s88-Size: 16 Bit
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	s88-Size: 24 Bit
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	s88-Size: 32 Bit
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	s88-Size: 48 Bit
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	s88-Size: 64 Bit
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	s88-Size: 80 Bit
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	s88-Size: <b>Definiert durch CV, default 128 Bit</b>

Informationen für weitergehende Anwendungsmöglichkeiten finden Sie auf der Internetseite von OpenDCC.

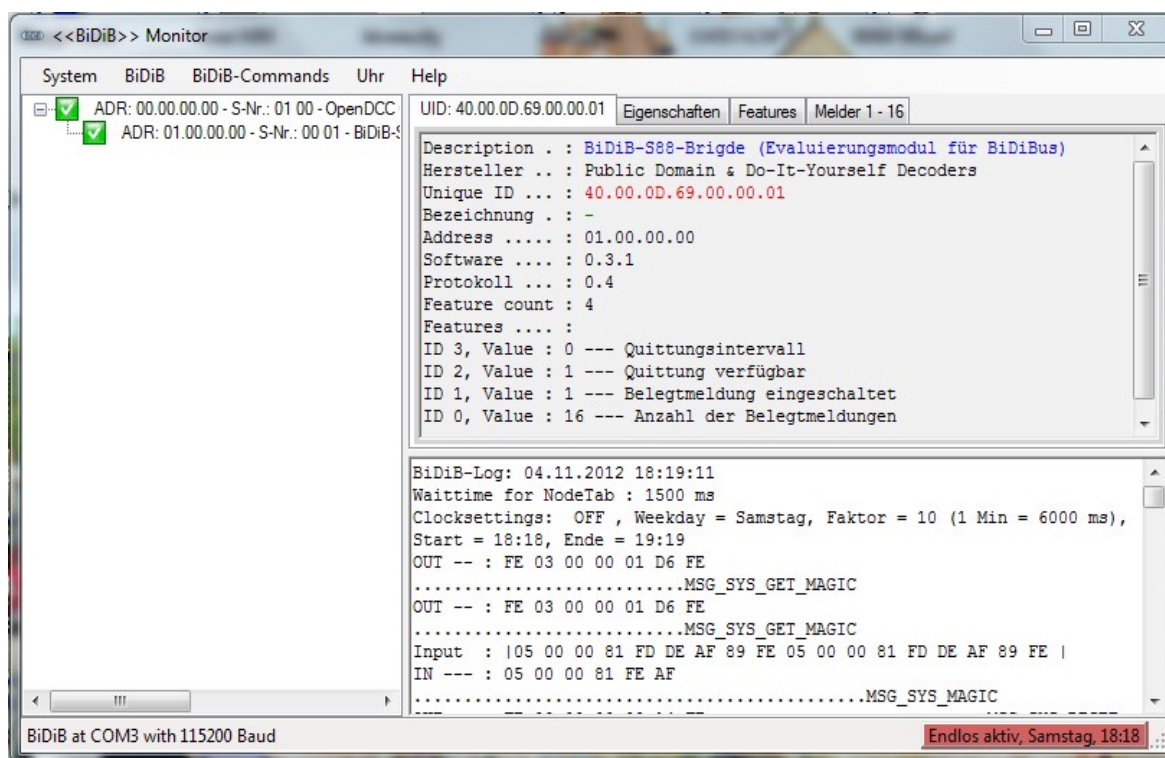
# OpenDCC

## Einrichten am BiDiB:

zur Kontrolle der Funktionsfähigkeit des Moduls kann der BiDiB – Monitor genutzt werden.

Die BiDiB – S88-Bridge in die BiDiB – Topologie koppeln ( X1 und X2 ), dann über den BiDiB – Monitor Verbindung zum Netzwerk herstellen.

Ist alles richtig zusammengebaut, programmiert und verdrahtet sollte das Ergebnis so aussehen:



Die BiDiB S88-Bridge meldet sich selbst am Bus an. Gut erkennbar als unterstes Device im Adressbaum.

Die weiteren Kenngrößen werden im linken Fenster angezeigt und ermöglichen die Kontrolle der Seriennummer oder des Softwarestands usw.

Haben Sie bereits einen s88 Rückmelder an die s88 Eingangsbuchse X 22 angeschlossen, den Reiter Melder 1-16 gewählt, so kann jetzt jeder einzelne Rückmeldeeingang auf Funktion getestet werden.

Weitere Möglichkeiten zur Anwendung des BiDiB – Monitors entnehmen Sie bitte der Internetseite von OpenDCC und dem Forum.

# OpenDCC

## **Anhang:**

Damit wären wir am Ende dieser Bauanleitung.

In einem separaten Applikationsmanual werden Programmierung der CV's, Anwendung der API, Einstellungsvarianten, Anschluss und Verdrahtungsmöglichkeiten ausführlich beschrieben.

Weitergehende Fragen bei Problemen oder Anregungen, Verbesserungsvorschlägen wenden Sie sich bitte ausschließlich an das OpenDCC-Forum.

Verwenden Sie stets die aktuellsten Softwarestände.

**Arbeiten Sie aktiv mit, für Verbesserungsvorschläge und Hinweise auf Fehler sind wir sehr dankbar.**