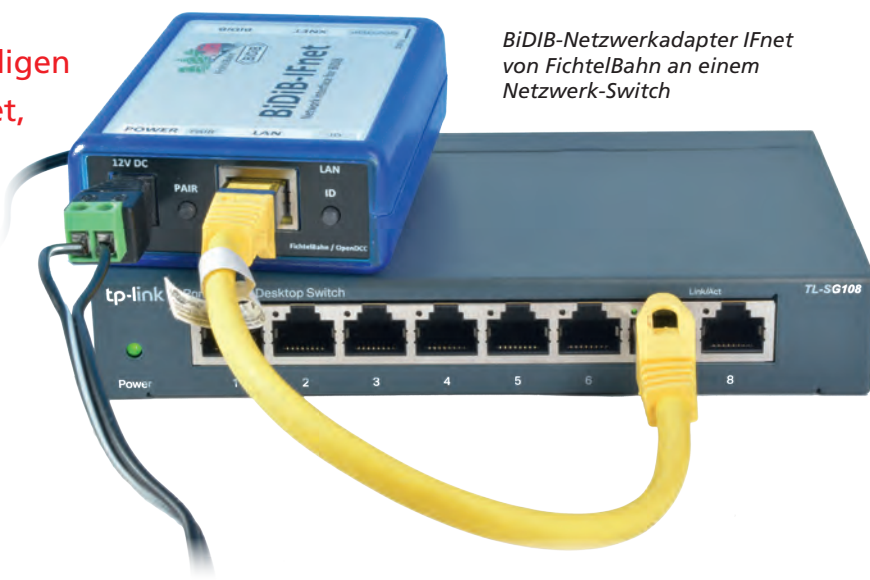


## Die Netzwerkvariante des BiDiB-Protokolls

# NET(T) MIT BIDIB

BiDiB ist im Vergleich zu den altherwürdigen Modellbahn-Bus-Systemen wie XpressNet, LocoNet und s88 noch ein recht junges System. Es wurde so konzipiert, dass auch bei vielen RailCom-Daten eine schnelle Übertragung problemlos möglich ist. Mit netBiDiB verbreitet sich nun auch eine Netzwerk-Version des Protokolls. Heiko Herholz erklärt die Struktur von BiDiB und zeigt, welche Vorteile netBiDiB hat.



*BiDiB-Netzwerkadapter IFnet von FichtelBahn an einem Netzwerk-Switch*

Vor etwa 13 Jahren hat mir Wolfgang Kufer zum ersten Mal seine Ideen zu BiDiB erläutert. Ich kann mich noch relativ gut daran erinnern, wie er mir die Struktur erläuterte und ich dann wissen wollte, wie die Hardware-Umsetzung aussieht. Die gab es aber damals noch gar nicht so konkret, denn BiDiB ist etwas anders als alle bisherigen Modellbahn-Bus-Systeme.

Diese kann man grob in zwei Gruppen unterteilen: Es gibt Modellbahn-Busse, bei denen erst die Hardware und ein Kommunikationsgrundgerüst da war, wie zum Beispiel der CAN-Bus. Dieser ist ein Industriebus, der vor allem im Automobil-Sektor zum Einsatz kommt. Die verschiedenen Modellbahn-CAN-Busse setzen darauf auf und übertragen ihre Modellbahndaten.

Die zweite Gruppe sind möglichst einfach gehaltene Übertragungssysteme, die speziell für die Modellbahn entwickelt wurden, wie zum Beispiel LocoNet oder RS-Bus. Beide Systeme kommen mit sehr wenig Hardware aus und sind dafür aber auch recht langsam. Außerdem decken sie nur die Anforderungen ab, die zum Zeitpunkt der Entwicklung vor mehr als 25 Jahren galten. Damals gab es noch keine automatische Anmeldung und auch kein RailCom. Damit gab es nur eine überschaubare Datenmenge, die über einen Modellbahnbus übertragen werden musste.

BiDiB ist nun ein komplett anderer Ansatz. Hier wurde ein Protokoll entwickelt, das aktuellen Modellbahn-Anforderungen folgt. Es klammert sich zunächst an keine feste Hardware und kann daher grundsätzlich über unterschiedliche Übertragungswege laufen. Vom Entwicklungsteam wurden drei Wege ausgewählt: USB/seriell, BiDiBus und Netzwerk.

### PLUG & PLAY

Bei BiDiB besitzt jeder Teilnehmer eine eindeutige Identifikationsnummer (Unique ID). Diese besteht aus einer Hersteller-Kennung, einer Produktkennung und einer Seriennummer. Grundsätzlich war es hier vorgesehen, die Hersteller-ID zu verwenden, die auch in Lok- und Fahrzeugdecodern genutzt wird. Da aber diese ID vom amerikanischen Modellbahnverband NMRA nur noch an Hersteller vergeben wird, die derartige Decoder mit DCC-Protokoll herstellen und von der NMRA zertifizieren lassen, gibt es Hersteller wie FichtelBahn, die keine derartige ID besitzen, da sie gar keine entsprechenden Decoder herstellen.

Daher erfolgt eine Verwaltung der IDs durch das BiDiB-Team. Alle Hersteller- und Produkt-IDs sind auf [www.bidib.org](http://www.bidib.org) dokumentiert. Auf dieser Seite wird auch das

BiDiB-Protokoll zum Nachlesen bereitgestellt. Bei Detailfragen kann man sich hier auch zu einem BiDiB-Forum anmelden und weitere Hilfe erhalten.

Wird ein neuer BiDiB-Client an ein BiDiB-System gesteckt oder ein komplettes System eingeschaltet, so melden sich alle Clients mit ihrer Unique ID an und übertragen dabei gleich einige Informationen, wie die Produktart (zum Beispiel: Rückmelder, Booster). Die Organisation von BiDiB-Elementen erfolgt in einer Knotenpunkt-Struktur, bei der die Knoten in unterschiedlichen Hierarchie-Ebenen liegen können. Über Hubs können Knoten miteinander verbunden werden. BiDiB-Nachrichten werden mit einer CRC-Prüfsumme und einer zusätzlichen Indizierung abgesichert. So ist nachvollziehbar, ob alle Nachrichten angekommen sind.

Im BiDiB-Protokoll sind Nachrichten derzeit in elf verschiedene Bereiche aufgliedert. Neben der Unterscheidung in verschiedene funktionale Bereiche für den Modellbahnbetrieb wie Melder, Booster und Schalten ist auch ein Bereich für Firmware-Updates vorhanden. BiDiB lässt sich sehr einfach mit dem PC auf dem aktuellen Stand halten. Mittels der für alle Betriebssysteme erhältlichen kostenlosen Software BiDiB-Wizard lassen sich Firmware-Updates auf alle BiDiB-Knoten aufspielen. Die

Software kann dabei sowohl für Tams- als auch für FichtelBahn-Komponenten eingesetzt werden.

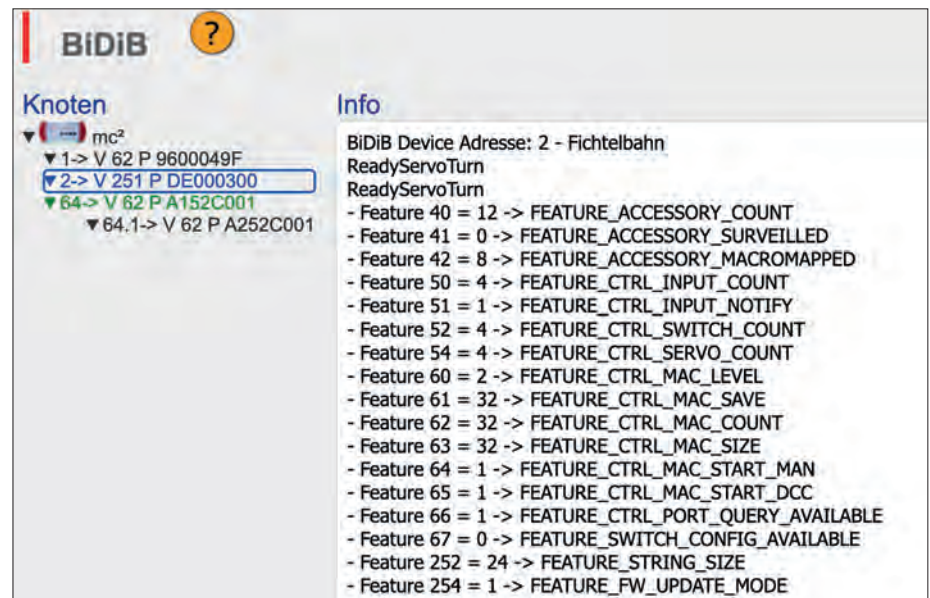
## BIDIBUS

Seit etlichen Jahren sind inzwischen BiDiB-Bausteine bei Tams und Fichtelbahn erhältlich. Diese sprechen untereinander BiDiB und sind über den BiDiBus miteinander verbunden. Genutzt wird hier das serielle RS-485 Protokoll. Dieses kommt zwar auch beim XpressNet zum Einsatz, wird aber bei BiDiBus komplett anders genutzt. Hier läuft die Datenübertragung mit sehr schnellen 500 kBaud ab. Das ist etwa achtmal schneller als auf dem XpressNet und im Vergleich mit Loconet sogar dreißigfach. Selbst im Vergleich mit den schnellen CAN-Bus-Systemen von ESU, Märklin und ZIMO erreicht BiDiBus noch die doppelte Geschwindigkeit.

Die Knotenstruktur des BiDiB-Protokolls ermöglicht dabei mindestens 1000 Teilnehmer auf einer Reichweite von rund 400 Metern. An einem BiDiBus-Knoten sind entsprechend der RS485-Spezifikation bis zu 32 Teilnehmer möglich. Durch Hubs und weitere Knoten ist BiDiB aber nahezu beliebig kaskadierbar. Die Übertragung der BiDiB-Daten aus dem BiDiBus an einen Steuerungsrechner erfolgt bisher über eine serielle- oder USB-Schnittstelle. Die Kommunikation läuft hierbei bevorzugt mit schnellen 1 MBaud. Mit dem Erscheinen der Digitalzentrale mc<sup>2</sup> von Tams und dem IFnet von FichtelBahn ist aber nun eine direkte Umsetzung der BiDiBus-Nachrichten auf das Netzwerk möglich. So lassen sich bei Bedarf größere Übertragungsgeschwindigkeiten realisieren.

## NETBIDIB

Bereits von Anfang an war auch der Versand von BiDiB-Nachrichten über das Netzwerk vorgesehen. Dies hat einige Vorteile, so fällt der bei Windows-Rechnern schwierige Umgang mit Treibern für die USB-Schnittstelle weg und nervige Dinge wie die Suche nach dem richtigen ComPort oder dass immer derselbe USB-Port genutzt werden muss, sind nicht mehr nötig. Der Anlagenrechner kann dank der Netzwerk- anbindung an einem anderen Ort stehen als die Modellbahnanlage. Als kleine und angenehme Nebeneffekte sind Dinge



The screenshot shows the BiDiB web interface. On the left, under 'Knoten', there is a tree view of nodes: 'mc<sup>2</sup>' (expanded), '1-> V 62 P 9600049F', '2-> V 251 P DE000300' (highlighted), '64-> V 62 P A152C001', and '64.1-> V 62 P A252C001'. On the right, under 'Info', it shows 'BiDiB Device Adresse: 2 - Fichtelbahn' and 'ReadyServoTurn'. Below that is a list of features: '- Feature 40 = 12 -> FEATURE\_ACCESSORY\_COUNT', '- Feature 41 = 0 -> FEATURE\_ACCESSORY\_SURVEILLED', '- Feature 42 = 8 -> FEATURE\_ACCESSORY\_MACROMAPPED', '- Feature 50 = 4 -> FEATURE\_CTRL\_INPUT\_COUNT', '- Feature 51 = 1 -> FEATURE\_CTRL\_INPUT\_NOTIFY', '- Feature 52 = 4 -> FEATURE\_CTRL\_SWITCH\_COUNT', '- Feature 54 = 4 -> FEATURE\_CTRL\_SERVO\_COUNT', '- Feature 60 = 2 -> FEATURE\_CTRL\_MAC\_LEVEL', '- Feature 61 = 32 -> FEATURE\_CTRL\_MAC\_SAVE', '- Feature 62 = 32 -> FEATURE\_CTRL\_MAC\_COUNT', '- Feature 63 = 32 -> FEATURE\_CTRL\_MAC\_SIZE', '- Feature 64 = 1 -> FEATURE\_CTRL\_MAC\_START\_MAN', '- Feature 65 = 1 -> FEATURE\_CTRL\_MAC\_START\_DCC', '- Feature 66 = 1 -> FEATURE\_CTRL\_PORT\_QUERY\_AVAILABLE', '- Feature 67 = 0 -> FEATURE\_SWITCH\_CONFIG\_AVAILABLE', '- Feature 252 = 24 -> FEATURE\_STRING\_SIZE', and '- Feature 254 = 1 -> FEATURE\_FW\_UPDATE\_MODE'.

*BiDiB-Knoten-Ansicht auf dem Web-Interface einer Digitalzentrale mc<sup>2</sup> von Tams. Am BiDiBus sind hier drei Knoten angeschlossen: Als Knoten 1 ein Schaltdecoder SD.34-BiDiB von Tams, als Knoten 2 ein ReadyServoTurn von FichtelBahn und als Knoten 64 der interne BiDiB-Hub der mc<sup>2</sup>. An diesem wiederum hängt als Knoten 64.1 der interne s88-Adapter zum Einlesen der Rückmeldemodule auf dem s88-Bus der mc<sup>2</sup>.*

*Alle Fotos und Screenshots: Heiko Herholz*

wie die galvanische Trennung von Anlage und PC schon automatisch vorhanden.

Bei netBiDiB kommt das Netzwerk-Transportprotokoll TCP zum Einsatz. Portnummern werden aus dem dynamischen Bereich verwendet. Bei Servern ist der Port 62875 voreingestellt. Dank mDNS wird ein netBiDiB-Server im Netzwerk automatisch gefunden. Ein umständliches Hantieren mit IP-Adressen und Ports entfällt. Der Verbindungsaufbau besteht bei netBiDiB aus vier Phasen:

- Discovery: BiDiB-Knoten werden im Netzwerk gesucht und ihre grundlegenden Eigenschaften sowie IP-Adresse und Portnummer ermittelt.
- Linkaufbau: Ein Kommunikationskanal zwischen zwei netBiDiB-Teilnehmern wird geöffnet.
- Pairing: Zwei Teilnehmer machen sich grundsätzlich miteinander bekannt und definieren ihre Verbindung als zulässig.
- Kontrolle: Aufbau einer Steuerverbindung zwischen zwei Teilnehmern. Ein Host kann ab jetzt den bei ihm angemeldeten Knoten steuern.

Beim Aushandeln der Transportverbindung kommen lokale Nachrichten zum

Einsatz, die zwar grundsätzlich den normalen Nachrichtenaufbau für BiDiB aufweisen, aber nur lokal verwendet und nicht an andere Knoten weitergeroutet werden.

BiDiB lässt sich auch netzweise kaskadieren. So ist es möglich, für große Anlagen mehrere BiDiB-Teilnetze aufzubauen und diese über einen Hub zusammenzuführen.

## SCHNELLES NETZ

Als direkt für den Modellbahn-Einsatz konzipiertes System ist BiDiB besonders leistungsfähig. Mit dem Einsatz von netBiDiB werden weitere Möglichkeiten geschaffen, die einen professionellen Einsatz auch bei großen Anlagen ermöglichen.

BiDiB ist zwar nicht bei den etablierten Normungsverbänden genormt, hat sich aber dennoch auf eine breite Basis gestellt. Aus dem Kreis der Entwickler heraus wurde der Verein BiDiB & Tool e.V. gegründet, der es sich zur Aufgabe gemacht hat, die Weiterentwicklung des BiDiB-Protokolls zu sichern, die ID-Datenbank zu pflegen und auf die Konformität der BiDiB-Produkte untereinander zu achten.

*Heiko Herholz*

## BIDIB-PROTOKOLL

<https://bidib.org/protokoll/intro.html>