

# BiDiB auf der Modellbahn

BiDiB kennt einzelne Module (diese heißen Knoten), und solche Knoten können bestimmte Funktionen auf der Modellbahn erfüllen, z.B. Fahren, Schalten, Rückmelden, usw. Was genau ein Knoten kann, teilt er mittels sog. [Features](#) nach dem Verbinden dem PC-Programm automatisch mit.

## Prinzip

BiDiB bezeichnet erst mal nur das Protokoll, dieses Protokoll kann auf unterschiedlichen physikalischen Verbindungen implementiert sein. Gemeinsam ist allen Implementierungen, dass eine automatische Adressierung der Knoten erfolgt. Hierbei sind 3 Dinge wichtig:

- **IDENTIFY:** Das ist ein Taster / Stellknopf o.ä., mit diesem Taster wird eine deutlich erkennbare 'Markierung' (z.B. heftiges Blinken) des Knotens ausgelöst. Diese Markierung wird auch am PC angezeigt und kann dort wieder zurückgenommen werden oder auch von dort aus ausgelöst werden. Mittels Identify kann man als Anwender die Zuordnung der Anzeige in der Software und dem Hardwaremodule leicht nachvollziehen und kontrollieren.
- **USER\_NAME:** Einem BiDiB-Knoten kann man einen Benutzernamen geben (z.B. Booster\_HBF), dieser Name ist im Knoten gespeichert und wird von den Programmen dargestellt.
- **UNIQUE\_ID:** damit diese automatische Zuordnung funktioniert, benötigt jede Baugruppe eine eindeutige Kennung, die [UNIQUE\\_ID](#). Normalerweise wird diese Kennung beim Hersteller des Knoten fest eingebrannt. Die UNIQUE\_ID enthält neben Hersteller auch Informationen zu Produkt und zu angebotenen Funktionen des Knotens.

## Programme

BiDiB ist optimal geeignet, um mit dem PC, einen Kleincomputer (z.B. Raspberry Pi) oder einem Tablet Modellbahn zu steuern. Hierzu gibt es verschiedenste [Steuer- und Konfigurationsprogramme](#).

## Verbinden, Anschließen

Aktuell sind folgende Verbindungsmöglichkeiten für BiDiB verfügbar:

### USB

USB steht für Universal Serial Bus und ist der Standard für die Verbindung von Peripherie mit einem PC. Ein USB-Gerät meldet sich automatisch beim PC an, dieser sucht den [passenden Treiber](#) für das Gerät.

### BiDiBus

BiDiBus basiert auf RS485, das ist eine besonders robuste Standardtechnik für Industrieautomation. Mit Längen bis 400m sind damit auch ausgedehnte Anlagen möglich. Der BiDiBus muß linienförmig verlegt werden, an beiden Enden ist eine [BiDiBus Terminierung](#) erforderlich. BiDiBus wird mit preiswerten Patchkabeln verteilt. Auf diesem Kabel ist zugleich eine Verteilung des Gleissignals an die Booster integriert.

## Funktionen

Wie erfolgt die automatische Übermittlung der Funktionen einer Baugruppe/eines Knotens? Das erfolgt in zwei Stufen:

- Stufe 1: in der [UNIQUE\\_ID](#) wird die prinzipielle Fähigkeit mitgeteilt, eine bestimmte Funktionsart erfüllen zu können. Z.B. kann ein Knoten dem Bedien-Programm mitteilen, dass er Rückmelder enthält. Ein Knoten kann durchaus mehrere Funktionen zugleich erfüllen, er könnte z.B. ein Knoten für Fahren und Rückmelden sein.
- Stufe 2: die genauen Fähigkeiten (im Beispiel: die Zahl der Rückmelder und was diese können) werden dann über [Eigenschaftsfelder](#) (diese heißen 'features') mitgeteilt. In PC-Programmen wie z.B. [Monitor](#) kann man sich eine Übersicht der gemeldeten Features anzeigen lassen.

### Wie geht nun BiDiB mit diesen Funktionen um?

## Fahren

'Fahren' bedeutet das Steuern von Lokdekodern, also das Erzeugen der entsprechenden Gleisbefehle für Lokomotiven. Diese Gleisbefehle werden vom PC komplett beauftragt (also Geschwindigkeit, Loklicht, Geräuschfunktionen der Lok, usw). Diese Befehle werden vom Fahren-Knoten in DCC-Nachrichten umgesetzt und (intern oder am Bus) bereit gestellt. Die Befehle sind damit noch nicht auf dem Gleis oder bei der Lok, hierzu ist ein Gleisverstärker (Booster) erforderlich. Die Funktion Fahren wird z.B. vom [GBMBoost](#) bereitgestellt.

## Booster

Der Booster ist der notwendige Verstärker, welcher das Gleissignal mit Leistung versieht und aufs Gleis gibt. Für kleine Anlagen wird man einen Booster haben, bei größeren Anlagen können es auch schon mal 20 oder mehr Verstärker werden. Booster melden an den PC folgende Informationen zurück:

- Betriebszustand (An, Aus, Kurzschluß, Überhitzung, ...)
- aktueller Stromverbrauch

Für den stabilen Betrieb einer Anlage ist eine mikrosekundengenaue Synchronisation der Booster erforderlich, das wird automatisch durch die sichere, differentielle Übertragung des Steuersignals auf dem Buskabel gewährleistet. Der Booster erzeugt auch die Austastlücke für [Railcom](#).

## Schalten

Das 'Schalten' umfaßt das Betätigen von Zubehör am Gleis und neben dem Gleis. Zubehör am Gleis sind z.B. Signale, Weichen, Drehscheiben usw. Diese Objekte werden Accessory genannt. Signale können einen Begriff anzeigen, Weichen können eine bestimmte Stellung haben. Das ist der aktuelle Aspect (=Begriff) des Accessory. Wichtig dabei ist: Es kann immer nur ein Aspect ausgewählt werden.

Hier ein paar Beispiele:

- Eine einfache Weiche hat zwei Aspekte: gerade Fahrt und Abzweig.
- Ein Vorsignal hat (je nach Type) z.B. drei Aspekte: Vr0, Vr1, Vr2
- Eine Drehscheibe kann bis 48 Aspekte haben. Das sind dann die Abgänge. Dort wo das 'Häuschen' steht, dort ist der aktuelle Begriff.

BiDiB behandelt Zubehör über diese Accessory+Aspect Ebene. Man kann also z.B. einem Accessory befehlen, dass es den Aspect 0 einstellen soll. Es ist dann Sache des Bausteins, was er daraus macht. Manche Bausteine sind hier vom Anwender sehr flexibel konfigurierbar, andere bieten fertige Accessory z.B. zum Schalten von Weichen an. Hierzu wieder ein paar Beispiele:

- Das kann bei der [LightControl](#) ein [Servo](#) sein, der die Weiche 0 auf „Gerade“ stellt.
- Es kann aber auch ein Signal sein und hinter Aspect 0 ist ein Macro hinterlegt, das die LED-Ausganäge der LightControl so schaltet, das das Signalbild eines Vorsignals eingestellt wird.
- Bei einer Drehscheibe kann das auch das Anfahren der Bühne auf einen bestimmten Abgang (Aspect) sein.

Die meisten [Steuerprogramme](#) (RocRail, WDP, iTrain) verwenden diese Accessories zur Ansteuerung des „Zubehörs“.

Liste der Knoten mit 'Schalten'-Funktion:

- [OneServoTurn](#)
- [STμ](#)

'Schalten' kann mit unterschiedlichen Wirkelementen erfolgen, wie z.B. [Magnetartikel](#) oder [Servos](#).

## Rückmelden

'Rückmelden' bedeutet das Bereitstellen von Informationen über den Zustand und Aufenthaltsort von Loks und Zügen an das Steuerprogramm. Diese Funktion ist essentiell wichtig, damit automatisiert bzw. teilautomatisiert mit einem Steuerprogramm gefahren werden kann. Der Computer muß schließlich wissen, wo sich die Züge aufhalten. Rückmelden meint in BiDiB (im Gegensatz zu traditionellen Systemen) nicht nur den Transport von Kontakt- und/oder Belegtmelder, sondern auch von wichtigen Zusatzinformationen:

- Belegtmeldung: Die Anlage wird in Meldeabschnitte unterteilt, die Information, ob in einem bestimmten Abschnitt ein Fahrzeug vorhanden ist, wird an das Steuerprogramm geliefert.
- Adressmeldung: Es werden nicht nur Gleisbelegungen gemeldet, sondern mittels der per [railcom](#) übertragenen Informationen wird auch genau gemeldet, welche Lok mit welcher Adresse sich wo aufhält.
- Programmierinformationen: Mit railcom wird die sog. Programmierung auf dem Hauptgleis

(POM) erst richtig nutzbar: eine Lok kann überall auf der Anlage und während des laufenden Betriebes in ihren Eigenschaften eingestellt werden.

- Zustandsmeldungen: Eine Lok liefert Informationen über ihren aktuellen Betriebszustand (Temperatur, Geschwindigkeit) und über das Gleissignal selbst (Verschmutzungssensor).

## Beleuchten

Knoten mit der Funktion 'Schalten' können auch Licht schalten. Hierzu kennt BiDiB bei den Port folgende Typen:

- SWITCH (SPORT = Schaltport): einfache Schalter, mögliche Zustände An oder Aus.
- LIGHT (LPORT = Lightport): Lichtausgänge, diese können Dimmen (ab und auf, auf mit Neonröhrensimulation) sowie Blinken.
- BACKLIGHT: = Hintergrundlicht: Lichtausgang, einstellbarer Zustand ist die Zielhelligkeit. Diese sind optimal geeignet, um eine Tag-Nacht Simulation zu realisieren.

Der im [BiDiB-Wizard](#) integrierte [DMX Modeler](#) ermöglicht die Planung und Realisierung von Beleuchtungsszenarien.

Beleuchtungen sind oft ablaufgesteuert (abends gehen nach und nach die Lichter in der Stadt an), diese Abläufe sollen aber nicht starr sein. BiDiB definiert hierfür [Makros](#), diese ermöglichen das Zusammenfassen von Beleuchtungsaktionen zu Schrittfolgen und bieten damit die Grundlage für eine lebendige Ausgestaltung der Modellbahn.

Folgende BiDiB-Module sind besonders für Beleuchtungsaufgaben geeignet: [LightControl](#), [OneDMX](#), [MoBaLiSt](#), [NeoControl](#)

## Programmieren

Unter 'Programmieren' versteht der Modellbahner i.d.R. das permanente Verändern von Einstellungen in Modulen oder Lokdekodern. Einstellungen werden in sog. Konfigurationsvariablen (CV = configuration variable) gespeichert, bei DCC-Dekodern gibt es für die wichtigsten Einstellungen [genormte Adressen](#). \* **Einstellungen von CVs in BiDiB-Knoten** BiDiB hat eine schnelles und einheitliches Verfahren, um auf CVs zuzugreifen. Die PC-Tools wie [BiDiB-Monitor](#) und [BiDiB-Wizard](#) verwenden [XML-Definitionsdateien](#), um menügeführt und übersichtlich die Einstellungen in den Knoten vorzunehmen.

### • Einstellungen von CVs in Lokdekodern

Bei Lokdekodern gibt es zwei Methoden zu Lesen und Schreiben von CVs:

1. Servicemode: das ist eine langsame, adresslose Methode, dabei werden einzelne Bits jeweils mit Stromverbrauchserhöhung des Dekoders übertragen. Für Servicemode ist deshalb ein separater, von der Anlage getrennter Anschluß des Dekoders erforderlich, das sog. [Programmiergleis](#).
2. Programmieren auf dem Hauptgleis: Hierbei werden die bidirektionalen Fähigkeiten von railcom benutzt: Die entsprechenden Befehle werden im 'Fahren'-Mode an den ausgewählten Lokdekoer geschickt, dieser beantwortet die Konfiguration mit einer railcom-Antwort. Das 'Melden'-System leitet die Antwort zurück an den PC. Bei einer flächendeckenden Ausrüstung der Anlage mit BiDiB-Melder wie z.B. den [GBMBoost](#) funktioniert diese Methode überall auf der

Anlage, ein extra Programmiergleis ist nicht mehr nötig.

## Kompatibilität

BiDiB bietet eine bisher unerreichte Qualität und Zuverlässigkeit der Modellbahnsteuerung. Jetzt hängt das natürlich nicht nur am Bussystem, sondern auch an den anderen eingesetzten Komponenten. Die Entwickler und Anwender von BiDiB testen und prüfen diese Komponenten:

- [Lokdekoder](#) Hier richtet sich unter Augenmerk besonders auf die Unterstützung von [railcom](#)-Funktionen.
- [Magnetartikel](#) Diese benötigen teils unterschiedlich starke Treiber.
- [Servos](#)

From:

<https://forum.opendcc.de/wiki/> - **BiDiB Wiki**

Permanent link:

<https://forum.opendcc.de/wiki/doku.php?id=loesungen>

Last update: **2016/07/05 10:52**

