

XML-Struktur einer CV-Definitionsdatei 2.0

Diese Seite beschreibt die zweite Struktur Version von CV-Definitionsdateien.

Die neue Version wird im ersten Release (Sep/Okt '16) vorerst nur bei den Decoder Definitionen Anwendung finden. (BiDiB-Node Definitionen bleiben unberührt)

Vorkonfigurierte Decoder Definitionen können zukünftig von der neuen Decoder Datenbank [Decoder DB](#) bezogen werden. Das neue Format wird in einem überarbeiteten Editor im neusten Release vom [BiDiB-Monitor](#) unterstützt.

Hinweise

* Alle XML-Tags und Attribute sind Case-Sensitiv im camelCase Format. Groß- und Kleinschreibung muss beachtet werden.

```
neu: <decoderFirmwareDefinition></decoderFirmwareDefinition> (alt:  
<VendorCV></VendorCV>)
```

1. Grundstruktur

Die XML-Datei besteht aus folgenden Teilen.

- 1) Informationen zur Version der Datei.
- 2) Informationen zur Firmware.
 - 2.1) Decoder auf denen diese Firmware installiert werden kann.
 - 2.2) Cv Angaben entsprechend der unterstützten Protokolle.
 - 2.2.1) Auflistung aller verfügbaren CVs.
 - 2.2.2) Einordnung der CVs in eine Baumstruktur.

```
<?xml version="1.0" encoding="utf-8" ?>  
<decoderFirmwareDefinition  
  xmlns="http://www.decoderdb.de/schema/decoderFirmware/1.1"  
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
  xmlns:ct="http://www.decoderdb.de/schema/commonTypes/1.1">  
  <version author="Max Musterman" lastUpdate="2014-12-09T00:00:00"  
  created="2014-12-09" />  
  <firmware version="34" versionExtension="" releaseDate="0001-01-01"  
  manufacturerId="157" manufacturerExtendedId="0">  
    <decoders>  
      <decoder name="N025" type="loco"/>  
    </decoders>  
    <protocols>  
      <protocol type="dcc" functions="0">  
        <cvs>  
          <ct:cv number="1" type="byte" possibleValues="1-127"  
  defaultValue="3" pomWriteExclude="true" mode="rw">
```

```
        <ct:description language="de" text="Adresse kurz"/>
        <ct:description language="en" text="Address short"/>
    </ct:cv>
    <ct:cv number="7" type="byte" mode="ro">
        <ct:description language="de"
text="Softwareversion"/>
        <ct:description language="en" text="Software Version
No." />
    </ct:cv>
    <ct:cv number="8" type="byte" mode="ro">
        <ct:description language="de"
text="Herstellerkennung"/>
        <ct:description language="en" text="Manufacturer ID"
help="Decoder reset mit '8'"/>
    </ct:cv>
</cvs>
<cvStructure>
    <category>
        <ct:description language="de" text="Informationen"/>
        <cvReference number="7"/>
        <cvReference number="8"/>
    </category>
    <category>
        ...
    </category>
</cvStructure>
</protocol>
</protocols>
</firmware>
</decoderFirmwareDefinition>
```

2. Dateiversion

Der Versions-Abschnitt liefert Information über die Datei.

```
<version
  createdBy="DecoderDB"
  creatorLink="www.decoderdb.de"
  author="Stephan Bauer"
  lastUpdate="2017-08-02T23:03:05"
  created="2016-12-30"/>
```

| Attribut | Beschreibung |
|-------------|---------------------------------------|
| createdBy | Ersteller der Datei (optional) |
| creatorLink | Link zum Ersteller (optional) |
| author | Author der Datei (optional) |
| created | Erstellungsdatum der Datei (optional) |

| Attribut | Beschreibung |
|------------|----------------------------------|
| lastUpdate | Letztes Änderungsdatum der Datei |

3. Firmware

Der Firmware-Abschnitt beinhaltet alle Informationen zur Firmware

```
<firmware
  version="34"
  versionExtension="sound"
  releaseDate="2010-12-30"
  manufacturerId="157"
  decoderDBLink="https://www.decoderdb.de/?decoderFirmwarePage=18">
  <decoders>
    <decoder name="N025" type="loco"/>
    <decoder name="T125" type="loco"/>
    <decoder name="T145" type="loco"/>
  </decoders>
</firmware>
```

| Attribut | Beschreibung |
|------------------|--|
| version | Versionsnummer der Firmware (optional) |
| versionExtension | Erweiterungsbeschreibung der Firmware |
| releaseDate | Veröffentlichungsdatum der Firmware |
| manufacturerId | Hersteller ID |
| decoderDBLink | DecoderDB Link zur Firmware |

3. CV definieren

Eine `<CV></CV>` Definition bildet immer einen Endknoten in der Baumstruktur.

Eine CV wird mit folgendem XML-Block definiert:

```
<CV Number="0" Type="Byte" Min="0" Max="255" Values="" Mode="ro" >
  <Description Lang="de" Text="Version" Help="Zeigt die EEPROM Version" />
  <Description Lang="en" Text="Version" Help="Shows the EEPROM version" />
</CV>
```

| Datentyp (Type) | Beschreibung |
|-----------------|--|
| Byte | Wertebereich 0 - 255 (8 Bit) |
| Bit | Es können bis zu 8 Bits definiert werden, Anzeige als Checkboxen |
| Radio | Es können bis zu 255 Werte definiert werden, Anzeige als Radio-Buttons |
| DccLongAddr | Lange Lok-Adresse |
| DccAccAddr | Lange Zubehör-Adresse |

| Datentyp (Type) | Beschreibung |
|-----------------|--|
| Int | Wertebereich 0 - 65.535 (16 Bit, little endian) |
| SignedByte | Wertebereich -128 - 127 (8 Bit) |
| Long | Wertebereich 0 - 4.294.967.295 (32 Bit, little endian) |
| DccAddrRG | Für BiDiB, siehe unten |
| GBM16TReverser | Für BiDiB, siehe unten |

| Attribut | Beschreibung |
|--------------|---|
| Number | Die Nummer der CV |
| Type | siehe oben |
| Min | Minimalwert für diese CV |
| Max | Maximalwert für diese CV |
| Default | Standardwert der CV |
| Low | Bei Type „Int“, „DccLongAddr“ und „DccAccAddr“ wird hier die CV des Lowbyte eingetragen |
| High | Bei Type „Int“, „DccLongAddr“ und „DccAccAddr“ wird hier die CV des Lowbyte eingetragen |
| Index31 | Wert für Index-CV 31 |
| Index32 | Wert für Index-CV 32 |
| Activebits | Type „Bit“: Einschränkung auf einzelne Bits, durch ; getrennt |
| Mode | Änderbarkeit: „rw“ = read/write, „ro“ = read only, „wo“ = write only, „w“ = write, „h“ = hide |
| Rebootneeded | CV-Änderung wird erst bei einem Neustart des Knoten wirksam - „true“ (BiDiB) |

<Description> definiert den beschreibenden Text der CV:

| Attribut | Beschreibung |
|----------|---|
| Lang | Sprache des beschreibenden Textes. (de, en, ...) |
| Text | Beschreibung der CV in der jeweiligen Sprache. |
| Help | Hilfebeschreibung der CV in der jeweiligen Sprache. |

Gültige Werte vorgeben : Bei den CV-Typen „Bit“ und „Byte“, können dem Anwender, mit **Min** und **Max** Werte für die CV vorgegeben werden.

Wenn **Values** eine, durch „;“ getrennte, Liste von Werten enthält, werden **Min** und **Max** nicht berücksichtigt.

3.1. CV-Type "Byte"

Durch Mausklick auf eine CV vom Type „**Byte**“, wird im unteren, rechten Bereich eine Eingabemaske für den Bytewert angezeigt.

The screenshot shows a tree view on the left with the following structure:

- Zeitanzeige
 - Format der Zeitanzeige 87 0 0 rw
- Einstellungen für DOGM163
 - ASCII_Code für Zeichen, wenn ... 88 0 0 rw
 - ASCII-Code für Zeichen, wenn ... 89 0 0 rw
- Parameter der Messung (highlighted)
 - Masstab, z.B. HO = 87, TT = 120 84 87 87 rw (highlighted)
 - Min. Zeit (ms) zwischen zwei Meldu... 85 0 0 rw
 - Zeit (ms) zwischen den Portabfragen 86 0 0 rw
- Messstrecke 0

On the right, the 'Byte - CV 84' configuration window is open, showing:

- Bytewert:
- Gültige Werte:
- CV84:

Über die Attribute **Min**, **Max** und **Values** können die möglichen Eingaben vorgegeben werden.

Beispiele für **Type=„Byte“**:

Hier wird die Eingabe nicht eingegrenzt. Alle Werte von 0 bis 255 sind möglich. Der Standard-Wert ist 5.

```
<CV Number="51" Type="Byte" Min="0" Max="255" Default="5" Mode="rw">
  <Description Lang="de" Text="Test-CV"/>
  <Description Lang="en" Text="CV for some tests"/>
</CV>
```

Hier wird die Eingabe über die **Min**- und **Max**-Werte auf 3 bis 20 eingegrenzt.

```
<CV Number="51" Type="Byte" Min="3" Max="20" Mode="rw">
  <Description Lang="de" Text="Test-CV"/>
  <Description Lang="en" Text="CV for some tests"/>
</CV>
```

Hier wird die Eingabe über **Values** auf die Werte 4, 6, 13 und 20 beschränkt. **Min** und **Max** werden ignoriert.

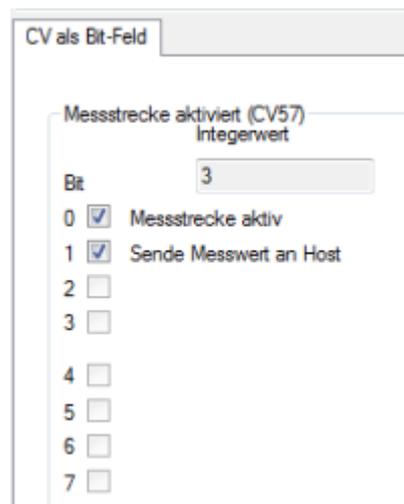
```
<CV Number="51" Type="Byte" Min="3" Max="20" Values="4;6;13;20" Mode="rw">
  <Description Lang="de" Text="Test-CV"/>
  <Description Lang="en" Text="CV for some tests"/>
</CV>
```

3.2. CV-Type "Bit"

```
<CV Number="4" Type="Bit" Default="0" Mode="rw" >
  <Description Lang="de" Text="Messstrecke aktiviert" />
  <Description Lang="en" Text="measuring track is aktive"/>
  <Bit Number="0">
    <Description Lang="de" Text="Messstrecke aktiv" Help="" />
    <Description Lang="en" Text="measuring track aktive" Help="" />
  </Bit>
  <Bit Number="1">
    <Description Lang="de" Text="Sende Messwert an Host"/>
    <Description Lang="en" Text="Send result to host"/>
  </Bit>
```

</CV>

Ist der Type „Bit“ für die CV angeben, werden für jedes Bit Checkboxen angezeigt. Es können mehrere Bits aktiviert sein.



Über das Tag <Description> (siehe oben) kann für jedes Bit eine Beschreibung und ein Hilfetext hinterlegt werden.

| Attribut | Beschreibung |
|----------|----------------------------|
| Number | Bit-Nummer beginnend mit 0 |

3.3. CV-Type "Bit" mit Gruppierung

In dem „normalen“ CV-Type „Bit“ können alle Bits gleichzeitig angeklickt werden. Für eine Auswahlliste ist es aber sinnvoll, immer nur eine Option zuzulassen. Wählt man eine Option an, werden die anderen „abgewählt“ (Radiobutton). Um beide Möglichkeiten zu kombinieren, wurde das Attribut „Group“ eingeführt.

```
<CV Number="81" Type="Bit" Default="2" Mode="rw" >
  <Description Lang="de" Text="Anzeige-Hardware"/>
  <Description Lang="en" Text="Display-Hardware"/>
  <Bit Number="0" Group="0">
    <Description Lang="de" Text="Kein Display angeschlossen"/>
    <Description Lang="en" Text="no display connected"/>
  </Bit>
  <Bit Number="1" Group="0">
    <Description Lang="de" Text="20 LEDs (Portstatus)"/>
    <Description Lang="en" Text="20 LEDs for Portstatus"/>
  </Bit>
  <Bit Number="2" Group="1">
    <Description Lang="de" Text="LCD DOGM163 16 x 3"/>
    <Description Lang="en" Text="LCD DOGM163 16 x 3"/>
  </Bit>
  <Bit Number="3" Group="1">
    <Description Lang="de" Text="7-Segment LED ( vierstellig )"/>
    <Description Lang="en" Text="7-Segment LED ( four digits )"/>
  </Bit>
  <Bit Number="7">
```

```

<Description Lang="de" Text="Ausgabe im Debug-Interface (FTDI)"/>
<Description Lang="en" Text="Output in Debug-Interface (FTDI)"/>
</Bit>
</CV>

```

Das Attribut „**Group**“ gibt an, welche Bits als Auswahlliste verwendet werden. Die anderen Bits können, wie im CV-Type „**Bit**“, weiter an- und abgeklickt werden.

Im Beispiel oben werden 2 Gruppen definiert. Damit werden

- die Bits 0-1 und 2-3 als Radiobutton verwendet.
- Bit 7 ist weiterhin eine Checkbox.

CV als Bit-Feld

Anzeige-Hardware (CV81)

Integerwert

| Bit | | Selection |
|-----|---|---|
| 0 | <input type="checkbox"/> | 0 <input type="radio"/> Kein Display angeschlossen |
| 1 | <input type="checkbox"/> | 1 <input type="radio"/> 20 LEDs (Portstatus) |
| 2 | <input type="checkbox"/> | 2 <input checked="" type="radio"/> LCD DOGM163 16x 3 |
| 3 | <input type="checkbox"/> | 3 <input type="radio"/> 7-Segment LED (vierstellig) |
| 4 | <input type="checkbox"/> | 4 <input type="radio"/> |
| 5 | <input type="checkbox"/> | 5 <input type="radio"/> |
| 6 | <input type="checkbox"/> | 6 <input type="radio"/> |
| 7 | <input checked="" type="checkbox"/> Ausgabe im Debug-Interface (FTDI) | 7 <input type="radio"/> |

Annotations:
▶ Bit 0-3 als Radiobutton
▶ Bit 4-7 "anklickbar"

3.4. CV-Type "Radio"

CV as Radiobutton

Angezeigte Daten (CV82)

Integer Wert

| Auswahl | | Auswahl |
|---------|--|--------------------------|
| 0 | <input type="radio"/> Keine Daten anzeigen | 8 <input type="radio"/> |
| 1 | <input type="radio"/> Modelbahnuhr | 9 <input type="radio"/> |
| 2 | <input type="radio"/> Messwert der Strecke aus CV83 | 10 <input type="radio"/> |
| 3 | <input checked="" type="radio"/> Alle Messergebnisse | 11 <input type="radio"/> |
| 4 | <input type="radio"/> Nur Portstatus | 12 <input type="radio"/> |
| 5 | <input type="radio"/> Portstatus und Modelbahnuhr | 13 <input type="radio"/> |
| 6 | <input type="radio"/> | 14 <input type="radio"/> |
| 7 | <input type="radio"/> | 15 <input type="radio"/> |

Beispiel:

```
<CV Number="82" Type="Radio" Default="0" Mode="rw" >
  <Description Lang="de" Text="Anzeige Daten"/>
  <Radio Value="0">
    <Description Lang="de" Text="Keine Daten anzeigen"/>
  </Radio>
  <Radio Value="1">
    <Description Lang="de" Text="Modelbahnuhr"/>
  </Radio>
  <Radio Value="2">
    <Description Lang="de" Text="Messwert der Strecke aus CV83"/>
  </Radio>
  <Radio Value="3">
    <Description Lang="de" Text="Alle Messergebnisse"/>
  </Radio>
  <Radio Value="4">
    <Description Lang="de" Text="Nur Portstatus"/>
  </Radio>
  <Radio Value="32">
    <Description Lang="de" Text="Portstatus und Modelbahnuhr"/>
  </Radio>
</CV>
```

3.5. CV-Type "Radio" mit Gruppierung

Mit Hilfe des Attributes „Group“ können Radio-Buttons zu Gruppen aufgeteilt werden. Alle Radios, die keiner Gruppe zugewiesen sind, bilden die letzte Gruppe. Die ausgewählten Radios der Gruppen werden addiert.

```
<CV Number="47" Type="Radio" Default="0" Mode="rw">
  <Description Lang="de" Text="Effekte Ausgang E (Lötpad E)"/>
  <Radio Value="0" Group="0">
    <Description Lang="de" Text="Aus"/>
  </Radio>
  <Radio Value="1" Group="0">
    <Description Lang="de" Text="Mars-Licht (pulsierend)"/>
  </Radio>
  <Radio Value="2" Group="0">
    <Description Lang="de" Text="Blitzlicht (Strobe)"/>
  </Radio>
  <Radio Value="7" Group="1">
    <Description Lang="de" Text="Abblenden wenn Fahr-St. > 0"/>
  </Radio>
  <Radio Value="8" Group="0">
    <Description Lang="de" Text="Blinken Phase B"/>
  </Radio>
  <Radio Value="9" Group="1">
    <Description Lang="de" Text="Zufalls-Licht"/>
  </Radio>
```

```

<Radio Value="12" Group="1">
  <Description Lang="de" Text="Kupplungsfunktion"/>
</Radio>
<Radio Value="32">
  <Description Lang="de" Text="Aus, wenn rückwärts"/>
</Radio>
<Radio Value="64">
  <Description Lang="de" Text="Aus, wenn vorwärts"/>
</Radio>
</CV>

```

CV as Radiobutton

Effekte Ausgang E (Lötpad E) (CV47)

Integer value 66

Offset : Aus, wenn vorwärts (bei Steuerung durch F1 bis F12) (64)

Selection

- 0 Ein/Aus
- 1 Mars-Licht (pulsierend)
- 2 Blitzlicht (Strobe)
- 3 z.Z. nicht benutzt
- 4 Blinken Phase A

Selection

- 8 Blinken Phase B
- 9 Zufalls-Licht
- 10 z.Z. nicht benutzt
- 11 z.Z. nicht benutzt
- 12 Kupplungsfunktion

3.6. CV-Type "DccLongAddr"

```

<CV Number="17" Type="DccLongAddr" Min="192" Max="231" Low="18" High="17"
Mode="rw">
  <Description Lang="de" Text="Lange Adresse"/>
</CV>
<CV Number="18" Type="DccLongAddr" Min="0" Max="255" Low="18" High="17"
Mode="rw">
  <Description Lang="de" Text="Lange Adresse"/>
</CV>

```

3.7. CV-Type "DccAccAddr"

```

<CV Number="1" Type="DccAccAddr" Min="1" Max="63" Low="1" High="9"
Mode="rw">
  <Description Lang="de" Text="Decoder-Adresse"/>
</CV>
<CV Number="9" Type="DccAccAddr" Min="0" Max="7" Low="1" High="9" Mode="rw">
  <Description Lang="de" Text="Decoder-Adresse"/>
</CV>

```

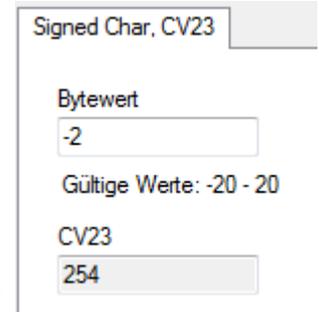
3.8. CVs über Indexregister CV31 und CV32 ansprechen

3.9. CV-Type "SignedByte" (BiDiB)

Durch Mausklick auf eine CV vom Type „**SignedByte**“, wird im unteren, rechten Bereich eine Eingabemaske für den Bytewert angezeigt. Im Unterschied zum Type „**Byte**“ können hier auch negative Werte eingegeben werden. Ansonsten sind die Attribute wie beim Type „**Byte**“ zu verwenden.

Beispiele für **Type=„SignedByte“**:

Hier wird die Eingabe über die **Min**- und **Max**-Werte auf den Bereich -20 bis 20 begrenzt.



Signed Char, CV23

Bytewert
-2

Gültige Werte: -20 - 20

CV23
254

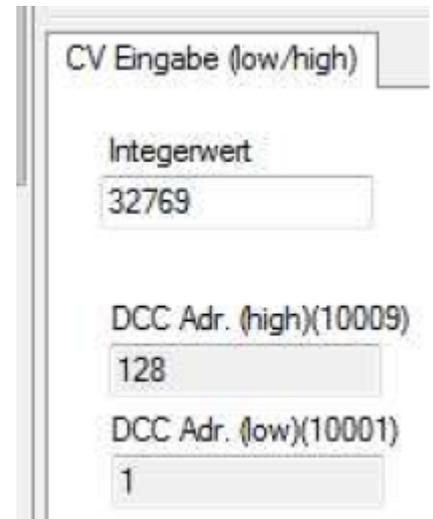
```
<CV Number="23" Type="SignedByte" Min="-20" Max="20" Default="" Mode="rw">  
  <Description Lang="de" Text="Temperaturoffset: 0=standard" />  
  <Description Lang="en" Text="Temperature offset: 0 default"/>  
</CV>
```

3.10. CV-Type "Int"

Der Type „Int“ setzt sich immer aus zwei CV's zusammen und bilden Low-Byte und High-Byte.

Beim Klick auf eines der beiden CV wird geprüft, ob die Partner CV vorhanden ist.

Wenn ja, wird im rechten, unteren Bereich eine Eingabemaske angezeigt.



CV Eingabe (low/high)

Integerwert
32769

DCC Adr. (high)(10009)
128

DCC Adr. (low)(10001)
1

Beide CV's müssen mit den Attribute **Low** und **High** aufeinander verweisen.

Beispiel:

```
<CV Number="1" Type="Int" Min="" Max="" Low="1" High="9" Mode="rw" >  
  <Description Lang="de" Text="DCC Adr. (low)" />  
  <Description Lang="en" Text="DCC Addr. (low)"/>
```

```

</CV>
<CV Number="9" Type="Int" Min="" Max="" Low="1" High="9" Mode="rw" >
  <Description Lang="de" Text="DCC Adr. (high)" />
  <Description Lang="en" Text="DCC Adr. (high)"/>
</CV>

```

3.11. CV-Type "DccAddrRG" (BiDiB)

Der Type „DccAddrRG“ setzt sich immer aus zwei CV's zusammen. Die erste CV ist angegeben, die zweite ist die nächsthöhere CV. Dieser Type arbeitet ähnlich wie „Int“.

Die eingegebene Integer-Zahl wird hier aber in den oberen 15 Bit gespeichert. Optisch wird der Wert also verdoppelt. Es ist also nur ein Wert zwischen 0 und 32767 möglich.

Das untere Bit der Lowbyte CV wird aber extra über zwei Radio-Button „R“ und „G“ gesetzt. Der gültige Eingabebereich wird über die Attribute **Min** und **Max** der Lowbyte CV bestimmt. Beide CV's müssen mit den Attribute **Low** und **High** aufeinander verweisen.

Beispiel:

```

<CV Number="59" Type="DccAddrRG" Min="0" Max="32767" Mode="rw" >
  <Description Lang="de" Text="Inhibit On, Addr"/>
  <Description Lang="en" Text="Inhibit On, Addr"/>
</CV>

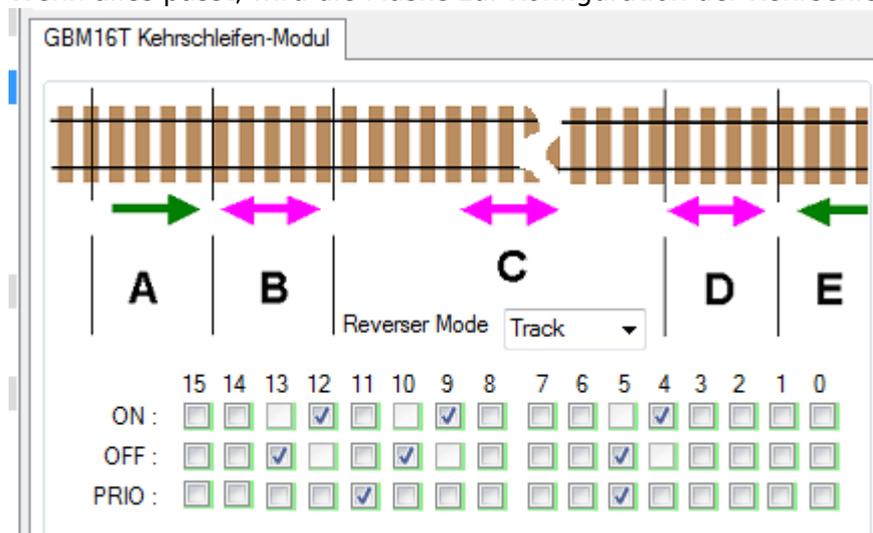
```

3.12. CV-Type "GBM16TReverser" (BiDiB)

Dieser Type ist nur für einen GBM16T zugelassen.

Beim Anklicken einer CV-Zeile vom diesem Type, prüft der Monitor, ob es sich um eine CV für die Kehrschleifenkonfiguration eines GBM16T handelt und ob die UID zu einem GBMBoost passt. Wenn nicht, ist diese CV nicht änderbar.

Wenn alles passt, wird die Maske zur Konfiguration der Kehrschleife angezeigt.



Beispiel:

```
<CV Number="46" Type="GBM16TReverser" Min="0" Max="255" Mode="rw" >  
  <Description Lang="de" Text="Kehrschleife Modus" />  
  <Description Lang="en" Text="Reverser Mode"/>  
</CV>
```

3.13. CV-Type "Long" (BiDiB)

The screenshot shows a tree view of CVs under 'Abgänge Einstellungen'. The tree is expanded to show 'Abgang 1'. The following table represents the data shown in the tree view:

| CV Name | CV Number | Min | Max | Mode |
|-----------------------------------|-----------|------------|------------|------|
| Polarität 0 = normal / 1 = inv... | 153 | 0 | 0 | rw |
| Schritt Position (low) | 154 | 429493473 | 429493473 | gt |
| Schritt Position (low)-0 | 154 | 225 | 225 | rw |
| Schritt Position (low)-1 | 155 | 140 | 140 | rw |
| Schritt Position (low)-2 | 156 | 153 | 153 | rw |
| Schritt Position (low)-3 | 157 | 25 | 25 | rw |
| Polarität 0 = normal / 1 = inv... | 158 | 0 | 0 | rw |
| Schritt Position (low) | 159 | 4278255513 | 4278255513 | gt |
| Schritt Position (low)-0 | 159 | 153 | 153 | rw |
| Schritt Position (low)-1 | 160 | 255 | 255 | rw |
| Schritt Position (low)-2 | 161 | 0 | 0 | rw |
| Schritt Position (low)-3 | 162 | 255 | 255 | rw |

The detail panel on the right shows 'Beschreibung: BiDiB-' and 'CV Long'. It contains input fields for 'Value' (4278255513), 'CV 159' (153), 'CV 160' (255), 'CV 161' (0), and 'CV 162' (255).

Durch Mausklick auf eine CV vom Type „**Long**“, wird im unteren, rechten Bereich eine Eingabemaske für den Wert angezeigt.

Der Type „**Long**“ besteht aus vier aufeinander folgenden CV's. Beginnend mit der CV-Nummer die im Attribut **Number** angegeben ist.

Beispiele für **Type=„Long“**:

Hier wird der Long-Wert in den CV's 154, 155, 156 und 157 abgebildet.

```
<CV Number="154" Type="Long" Min="0" Max="4294967295" Mode="rw">  
  <Description Lang="de" Text="Schritt Position (low)"/>  
  <Description Lang="en" Text="Step position (low)"/>  
</CV>
```

4. Templates

Innerhalb der Tags **<Templates></Templates>** (mit s) können Templates definiert werden, die es vereinfachen, mehrfach vorkommende CV-Strukturen, in der Baumstruktur abzubilden. Jedes Template wird mit dem Tag **<Template></Template>** (ohne s) definiert. Hier ein Beispiel für die CV's der LED-Ports der LightControl)

```

<Template Name="LED">
  <CV Number="0" Type="Byte" Min="-" Max="-" Low="-" High="-" Values="-"
Mode="rw">
    <Description Lang="de-DE" Text="LED: Einstellung der Stromquelle"/>
    <Description Lang="en-EN" Text="LED: courant source setup"/>
  </CV>
  <CV Number="1" Type="Byte" Min="-" Max="-" Low="-" High="-" Values="-"
Mode="rw">
    <Description Lang="de-DE" Text="LED: Helligkeit für Zustand 'aus'"/>
    <Description Lang="en-EN" Text="LED: light intensity at status
'off'"/>
  </CV>
  <CV Number="2" Type="Byte" Min="-" Max="-" Low="-" High="-" Values="-"
Mode="rw">
    <Description Lang="de-DE" Text="LED: Helligkeit für Zustand 'an'"/>
    <Description Lang="en-EN" Text="LED: light intensity at status
'on'"/>
  </CV>
  <CV Number="3" Type="Byte" Min="-" Max="-" Low="-" High="-" Values="-"
Mode="rw">
    <Description Lang="de-DE" Text="LED: Dimmzeit in Richtung 'aus', 0 =
schnell / 255 = langsam"/>
    <Description Lang="en-EN" Text="LED: time to dim, direction 'off', 0
= fast / 255 = slow" />
  </CV>
  <CV Number="4" Type="Byte" Min="-" Max="-" Low="-" High="-" Values="-"
Mode="rw">
    <Description Lang="de-DE" Text="LED: Dimmzeit in Richtung 'an', 0 =
schnell / 255 = langsam"/>
    <Description Lang="en-EN" Text="LED: time to dim, direction 'on', 0
= fast / 255 = slow" />
  </CV>
</Template>

```

Hier bitte nicht wundern, das die CV's mit den Nummern 0 bis 4 definiert sind. Die benötigten CV-Nummern werden später in der Baumstruktur, per Offset definiert.

5. Baumstruktur definieren

Die eigentliche Baumstruktur wird innerhalb des Tags **<CVDefinition></CVDefinition>** definiert. Mit dem Tag **<Node></Node>** wird ein Knoten in der Baumstruktur angelegt. Mit dem Tag **<Description>** innerhalb von **<Node></Node>** wird der Text des Knoten hinterlegt. Innerhalb von **<Node></Node>** werden per Template (Siehe 4.) oder CVRef die CV's abgebildet.

Beispiel für einen einfachen Knoten vom GBMBoost:



| Beschreibung | CV |
|-------------------------------------|----|
| GBMBoost Master, Booster, Interface | |
| Allgemeine Daten | |
| Parser Modus | 1 |
| DCC Herstellerkennung | 2 |
| Produkt-ID 1 | 3 |

```
<Node>
  <Description Lang="de" Text="Allgemeine Daten"/>
  <Description Lang="en" Text="Basis"/>
  <CVRef Number="1" />
  <CVRef Number="29" Values="8"/>
  ...
</Node>
```

5.1. Knoten mit Template

Um ein Template zu verwenden, kann das **<Node>**-Tag mit folgenden Attributen versehen werden:

| Attribut | Beschreibung |
|----------|--|
| Template | Name des Templates. |
| Count | Angabe, wie oft das Template wiederholt werden soll. |
| Offset | Angabe, mit welcher CV-Nummer die Nummerierung der CV's beginnt. |
| Next | Multiplikator des Offset das jeweilige Template (Count). |

Beispiel:

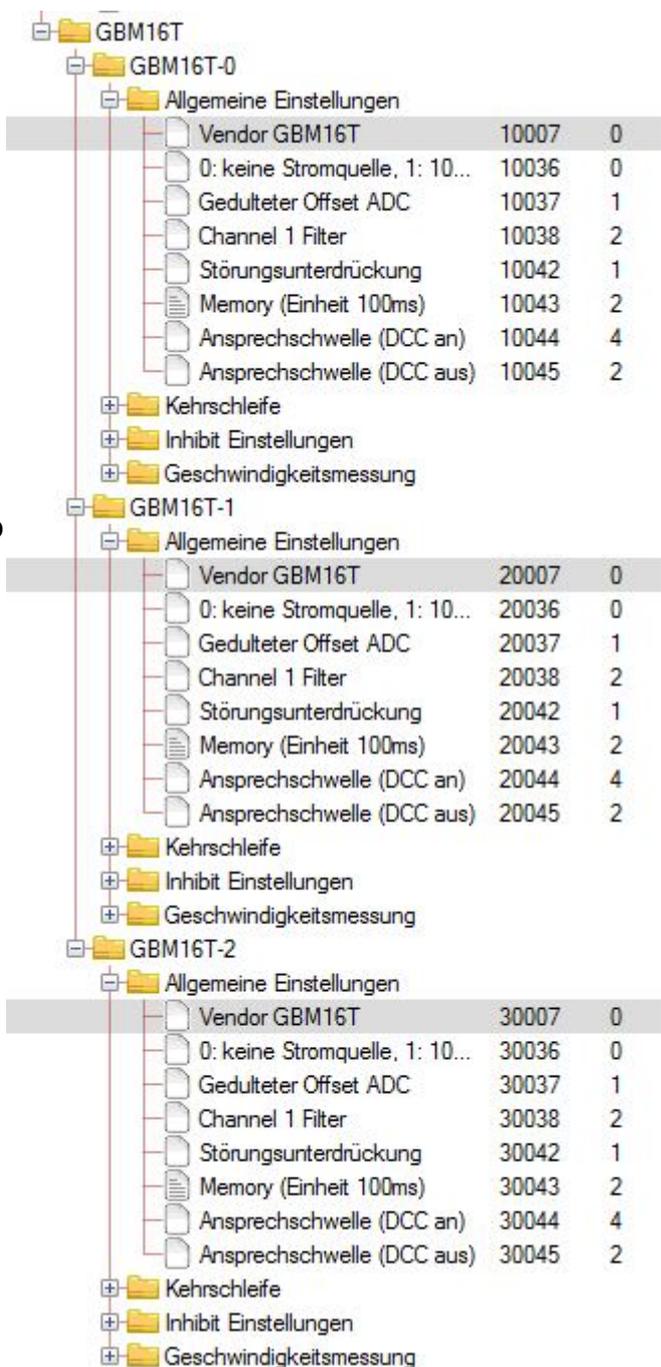
```
<Node>
  <Nodetext Lang="de" Text="GBM16T"/>
  <Nodetext Lang="en" Text="GBM16T"/>
  <Node Offset="10000" Template="GBM16T" Count="3" Next="10000">
    <Description Lang="de" Text="GBM16T-%%d"/>
    <Description Lang="en" Text="GBM16T-%%d"/>
  </Node>
</Node>
```

Hier wird das Template GBM16T angezogen. Es wird 3 mal angewendet. Beim ersten mal wird ein Offset von 10000 verwendet. Wenn also im Template die erste CV mit Nummer 0 definiert ist, wird im Baum diese CV als 10000 dargestellt.

Das zweite mal wird der Wert von Next auf das Offset addiert.

Also wird CV 0 hier mit 20000 dargestellt. Das dritte mal dann mit 30000.

Wie im **Nodetext**-Tag zu sehen ist, kann die Variable **%%d** als durchlaufende Nummer (Index, Count) verwendet werden, um den Knoten einen entsprechenden Namen zu geben. Die Variable **%%p** kann gleich wie %%d verwendet werden, ist aber um +1 höher als der Index.



5.2. Repeater

Mit dem **Repeater**-Tag kann eine *for-Schleife* für CV-Values definiert werden.

Beispiel:

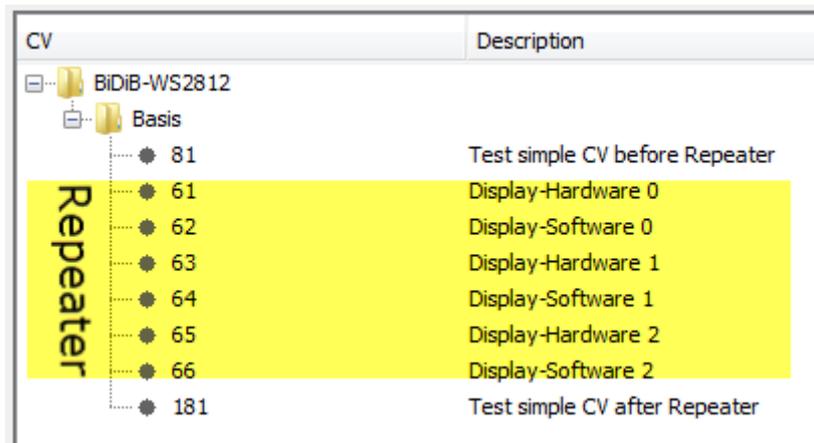
```
<Repeater Count="28">
  <CV Number="67" Type="Byte" Min="0" Max="255" Default="" Mode="rw">
    <Description Lang="de" Text="Fahrstufe %%d"/>
    <Description Lang="en" Text="Speedstep %%d"/>
  </CV>
</Repeater>
```

```
<Node>
  <Nodetext Lang="de" Text="Basis Daten" />
  <Nodetext Lang="en" Text="Basis" />
  <CV Number="81" Type="Radio" Mode="rw">
    <Description Lang="de" Text="Test einfache CV vor Repeater" />
    <Description Lang="en" Text="Test simple CV before Repeater" />
  </CV>

  <Repeater Offset="60" Count="3" Next="2">
    <CV Number="1" Type="Byte" Min="" Max="" Mode="rw">
      <Description Lang="de" Text="Anzeige-Hardware %%d" />
      <Description Lang="en" Text="Display-Hardware %%d" />
    </CV>
    <CV Number="2" Type="Byte" Min="" Max="" Mode="rw">
      <Description Lang="de" Text="Anzeige-Software %%d" />
      <Description Lang="en" Text="Display-Software %%d" />
    </CV>
  </Repeater>

  <CV Number="181" Type="Radio" Mode="rw">
    <Description Lang="de" Text="Test einfache CV nach Repeater" />
    <Description Lang="en" Text="Test simple CV after Repeater" />
  </CV>
</Node>
```

Die nachfolgende Abbildung zeigt das Resultat des Beispiels oben.



| CV | Description |
|-----|--------------------------------|
| 81 | Test simple CV before Repeater |
| 61 | Display-Hardware 0 |
| 62 | Display-Software 0 |
| 63 | Display-Hardware 1 |
| 64 | Display-Software 1 |
| 65 | Display-Hardware 2 |
| 66 | Display-Software 2 |
| 181 | Test simple CV after Repeater |

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

Permanent link: <https://forum.opendcc.de/wiki/doku.php?id=monitor:cv-definitionsfileneu&rev=1527110356>

Last update: 2018/05/23 23:19

