

P50X-Interface für 3D-Modellbahn Studio

Version 0.4

UrBahn

26.11.2023

Inhaltsverzeichnis

1	Einleitung	3
2	Szenarien	4
2.1	Reale Anlage	4
2.2	Virtuelle Anlage	4
2.3	Verwendung mehrerer Monitore und PCs	4
3	Virtuelle serielle Schnittstellen	5
4	Konfiguration des Schnittstellenprogramms	7
4.1	Name der Konfigurationsdatei	7
4.1.1	Aufruf des Schnittstellenprogramms ohne weitere Zusätze	7
4.1.2	Aufruf des Schnittstellenprogramms mit Parameter	7
4.2	Fehlerhafte oder nicht existierende Konfigurationsdatei	7
4.3	Leere Start-Konfigurationsdatei	7
5	Aufruf des Schnittstellenprogramms	8
5.1	Erster Testaufruf	8
5.2	Testaufruf mit laufendem 3D-Modellbahn Studio	9
5.3	Separater Rechner für das 3D-Modellbahn Studio	10
6	Anlage der Crossreferenz-Dateien	11
6.1	LokDatei	11
6.2	Melderdatei	12
6.3	Weichendatei	13
7	Einbinden der seriellen Schnittstelle	15
7.1	Zuordnung der seriellen Schnittstellen	15
7.2	Eintrag in der Config-Datei	15
8	Testanlage	17
8.1	Testlok	17
8.2	Testanlage in der Modellbahnsteuerung	17
8.3	Testlok in der Modellbahnsteuerung	18
8.4	Eintrag in der Datei Loks.txt	19
8.5	Kontaktmelder	20
8.6	Eintrag in der Datei Melder.txt	20
8.7	Testlauf	21
9	Weichen und ähnliches	22
9.1	Einfache Weiche	22
9.2	Doppelkreuzungsweiche	23
9.3	Signale	24
9.4	Entkupplungsgleise und weitere Teile mit Antrieb	24
10	Sonstiges	25
10.1	Monitordatei für Serielle Schnittstelle	25
10.2	Ereignis- und Kommandoport	25
10.3	Baudrate	26
10.4	Ist/Soll-Geschwindigkeit	26

1 Einleitung

Das 3D-Modellbahn Studio erlaubt die Planung und Simulation komplexer virtueller Modellbahnanlagen mit unbegrenzt vielen Loks, Signalen und Weichen. Die aktuelle Version verfügt über eine Schnittstelle, über die Kommandos an das Programm geschickt und Ereignismeldungen vom Programm empfangen werden können.

Über ein geeignetes Schnittstellenprogramm können Modellbahnsteuerungen an das 3D-Modellbahn Studio angebunden werden - z.B. Rocrail mittels RocStudio (Details siehe z.B. <https://community.3d-modellbahn.de/wiki/steuerung/steuerung-%C3%BCber-rocrailrocstudio-r21/>).

Dieser Artikel beschreibt ein weiteres Schnittstellenprogramm, welches Modellbahnsteuerungen an das 3D-Modellbahn Studio anbindet, die das P50X-Protokoll verwenden (z.B. TrainController, ...).

Die logische Steuerung erfolgt ausschließlich von der Modellbahnsteuerung in Richtung 3D-Modellbahn Studio; die umgekehrte Richtung (3D-Modellbahn Studio kontrolliert die Modellbahnsteuerung) ist nicht vorgesehen¹. Hier wäre ein möglicher Erweiterungsbedarf zu klären.

Der Artikel ist als Schritt-für-Schritt-Anleitung angelegt, um in einem Fehlerfall den "Knackpunkt" besser orten zu können.

In den Beispielen wird der TrainController 5.8 mit der Einstellung für TAMS EasyControl verwendet, da der Autor diese besitzt und sämtliche Tests damit durchgeführt hat.

Da das Schnittstellenprogramm jedoch lediglich erwartet, daß P50X-Kommandos über eine serielle Schnittstelle ausgetauscht werden, sollte es auch mit weiteren Modellbahnsteuerungen funktionieren. Hier bitte ich um Test-Unterstützung weiterer Modellbahner, die entsprechende Programme besitzen.

Hinweis:

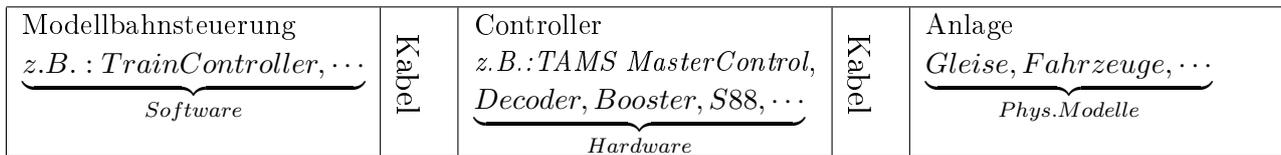
Dieser Text ist ausschließlich zur privaten Nutzung vorgesehen. Daher wird auf tangierte Copyrights / Urheberrechte nicht gesondert hingewiesen.

¹Z.B. wird das Umstellen einer Weiche im MBS nicht zur Modellbahnsteuerung übertragen, sehr wohl aber das Umstellen einer Weiche in der Modellbahnsteuerung zum MBS;

2 Szenarien

2.1 Reale Anlage

Eine reale Anlage lässt sich grob in folgende Blöcke gliedern:



2.2 Virtuelle Anlage

Die oben angeführte reale Anlage soll wie folgt auf eine virtuelle Anlage abgebildet werden:



2.3 Verwendung mehrerer Monitore und PCs

Die kleinste Lösung besteht darin, alle drei Softwareblöcke auf einen einzigen PC zu packen und über einen einzigen Monitor zu bedienen - schwierig, weil man nicht viel sieht.

Eine - aus meiner Sicht - vernünftige Lösung placiert die Softwareblöcke auf einem modernen PC, an dem zwei Monitore betrieben werden: ein Monitor für die Modellbahnsteuerung und ein weiterer Monitor für das 3D-Modellbahn Studio.

Die aufwändigste Lösung sieht pro Softwareblock einen eigenen PC mit Monitor vor (könnte sinnvoll sein bei Altgeräten mit mangelnder Performance).

Abhängig von der gewählten PC-Anzahl fallen die Interfaces zwischen den Softwareblöcken unterschiedlich aus:

- Liegen die Modellbahnsteuerung und das Schnittstellenprogramm auf unterschiedlichen PCs, werden die Programme wie in der realen Anlage über ein Kabel gekoppelt, z.B. ein serielles Kabel wie zwischen dem TrainController und der TAMS MasterControl;
- Liegen die Modellbahnsteuerung und das Schnittstellenprogramm auf demselben PCs, können virtuelle serielle Schnittstellen verwendet werden, z.B. das Programm com0com;
- Die Verteilung des Schnittstellenprogramms und des 3D-Modellbahn Studios auf einem oder zwei PCs hat lediglich eine unterschiedliche Adressierung zur Folge (Details in 5.3, Seite 10).

Der Autor verwendet *einen* PC mit *zwei* Monitoren.

3 Virtuelle serielle Schnittstellen

Sollten Sie die Modellbahnsteuerung und das Schnittstellenprogramm auf unterschiedlichen PCs laufen lassen und also ein Kabel für die Kopplung verwenden, können Sie diesen Abschnitt ignorieren.

Der open source-Treiber com0com für Windows eignet sich ausgezeichnet für die Aufgabe, die Modellbahnsteuerung (z.B. TrainController) mit dem Schnittstellenprogramm zu koppeln.

Die Konfiguration (siehe Dokumentation zu com0com) sollte folgendes Ergebnis haben:

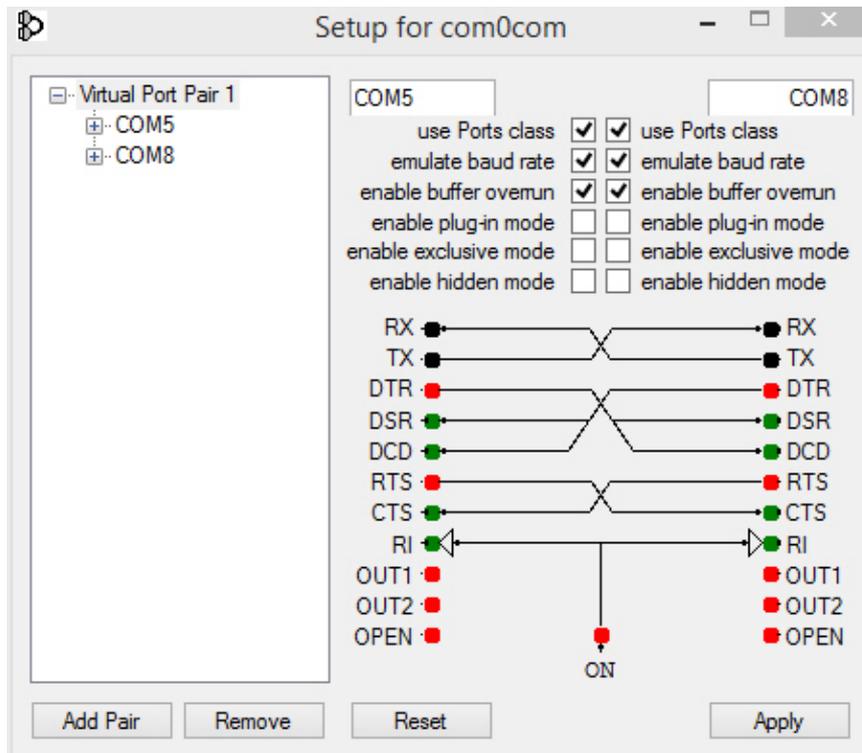


Abbildung 1: Setup for com0com

COM5 und *COM8* sind die hier beispielhaft verwendeten virtuellen seriellen Schnittstellen. Die Namen der Schnittstellen können Sie an Ihre Gegebenheiten anpassen.

Beim Autor ist die Modellbahnsteuerung TrainController an *COM8* und das Schnittstellenprogramm an *COM5* angeschlossen. Eine genaue Beschreibung folgt in 7.1 auf der Seite 15.

Hinweise:

Der Meldung "Die digitale Signatur der für dieses Gerät erforderlichen Treiber kann nicht überprüft werden ..." kann begegnet werden zum Beispiel

- durch Ignorieren;
- durch Kommandozeileingabe von `bcdedit.exe -set TESTSIGNING ON` und Neustart;
- durch Verwendung der Version

https://code.google.com/p/powersdr-iq/downloads/detail?name=setup_com0co...

Die verwendeten seriellen Schnittstellen sollten aus *COM1* ... *COM9* sein, da die Schnittstel-

lennamen auf vier Stellen gekürzt werden.

4 Konfiguration des Schnittstellenprogramms

Informationen, die das Schnittstellenprogramm benötigt und nicht selbst ermitteln kann, müssen diesem mitgeteilt werden. Dies erfolgt über eine Konfigurationsdatei.

4.1 Name der Konfigurationsdatei

4.1.1 Aufruf des Schnittstellenprogramms ohne weitere Zusätze

In diesem Fall arbeitet das Schnittstellenprogramm mit einer Konfigurationsdatei, die im selben Verzeichnis liegt wie das Schnittstellenprogramm und den Namen *Config.txt* trägt.

4.1.2 Aufruf des Schnittstellenprogramms mit Parameter

Beim Aufruf des Schnittstellenprogramms können Sie den Namen (inklusive Pfad) einer Konfigurationsdatei angeben.

4.2 Fehlerhafte oder nicht existierende Konfigurationsdatei

Beim Zugriff auf eine fehlerhafte oder nicht existierende Konfigurationsdatei erfolgt die Fehlermeldung "*Fehler beim Zugriff auf die Konfigurationsdatei.*"

4.3 Leere Start-Konfigurationsdatei

Erzeugen Sie bitte unter Beachtung der oben angeführten Regeln mit einem ASCII-Texteditor eine leere Konfigurationsdatei, z.B.

```
#####  
#                                                                 #  
# Config-File für Testanlage                                     #  
#                                                                 #  
# 07.06.2015 ur                                               #  
#                                                                 #  
#####
```

Hinweis: Zeilen, die mit einem # beginnen, sind Kommentarzeilen.

5 Aufruf des Schnittstellenprogramms

5.1 Erster Testaufruf

Wir setzen jetzt voraus,

- daß ein leeres Unterverzeichnis zur Verfügung steht (im Beispiel `i:\UrBahnTest\`),
- daß das Schnittstellenprogramm `UrBahn.exe` in dieses Verzeichnis kopiert wurde,
- daß die leere Startkonfigurationsdatei in dieses Verzeichnis kopiert wurde.

Mit "cmd" öffnen wir ein Kommando-Fenster und wechseln in das Verzeichnis. Ein `dir` gibt:

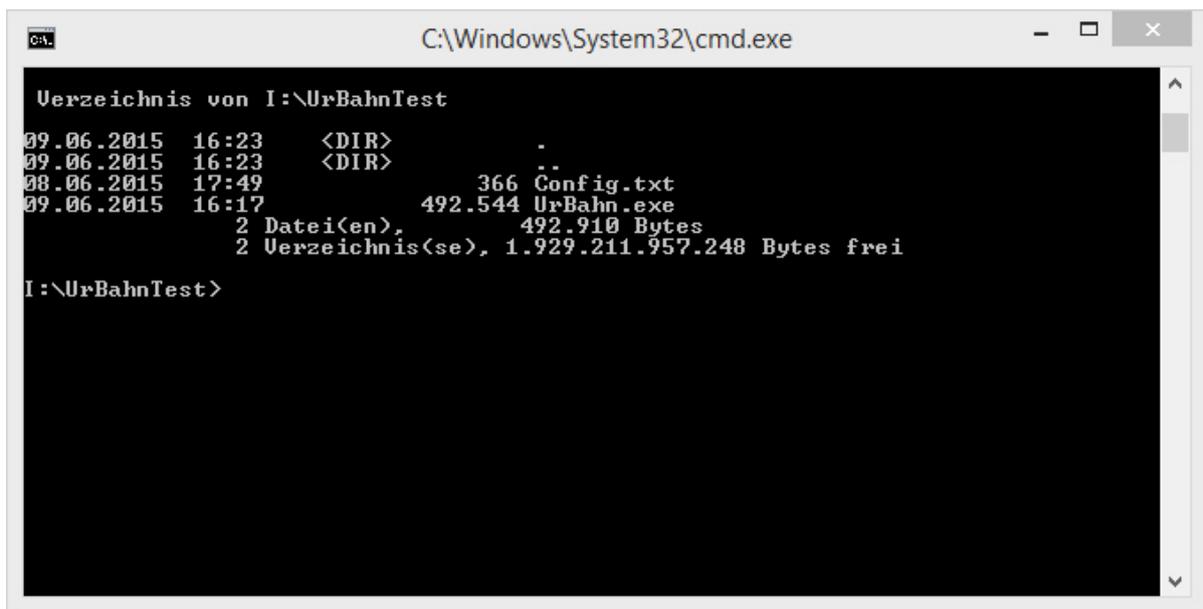
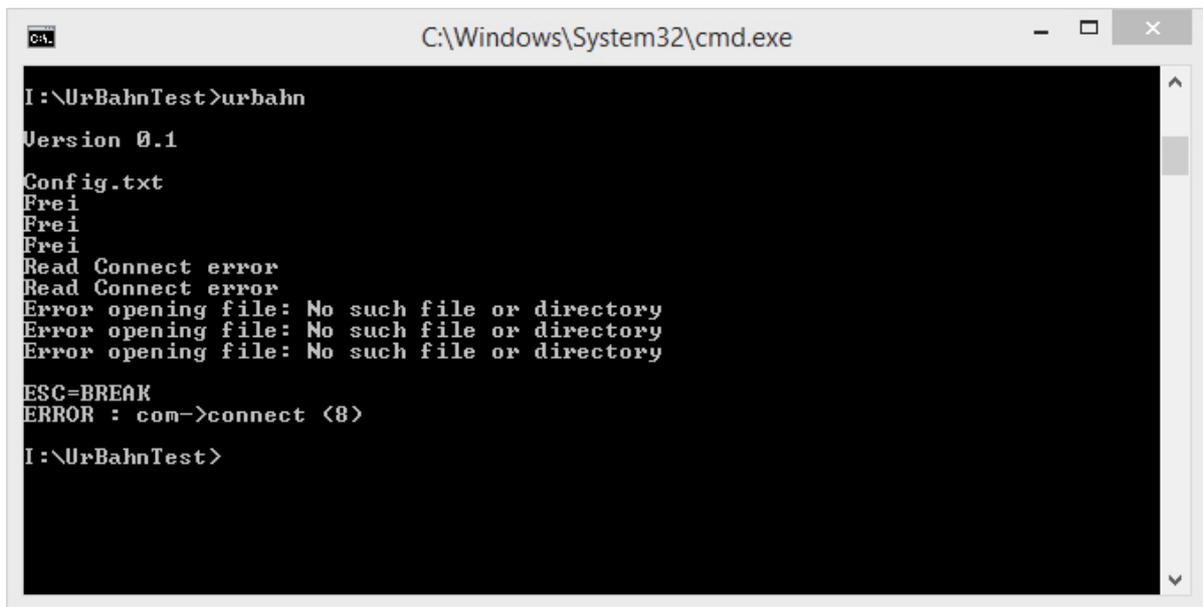


Abbildung 2: Anfangs-Unterverzeichnis

Aufruf von `UrBahn` gibt:



```
C:\Windows\System32\cmd.exe

I:\UrBahnTest>urbahn

Version 0.1

Config.txt
Frei
Frei
Frei
Read Connect error
Read Connect error
Error opening file: No such file or directory
Error opening file: No such file or directory
Error opening file: No such file or directory

ESC=BREAK
ERROR : com->connect <8>

I:\UrBahnTest>
```

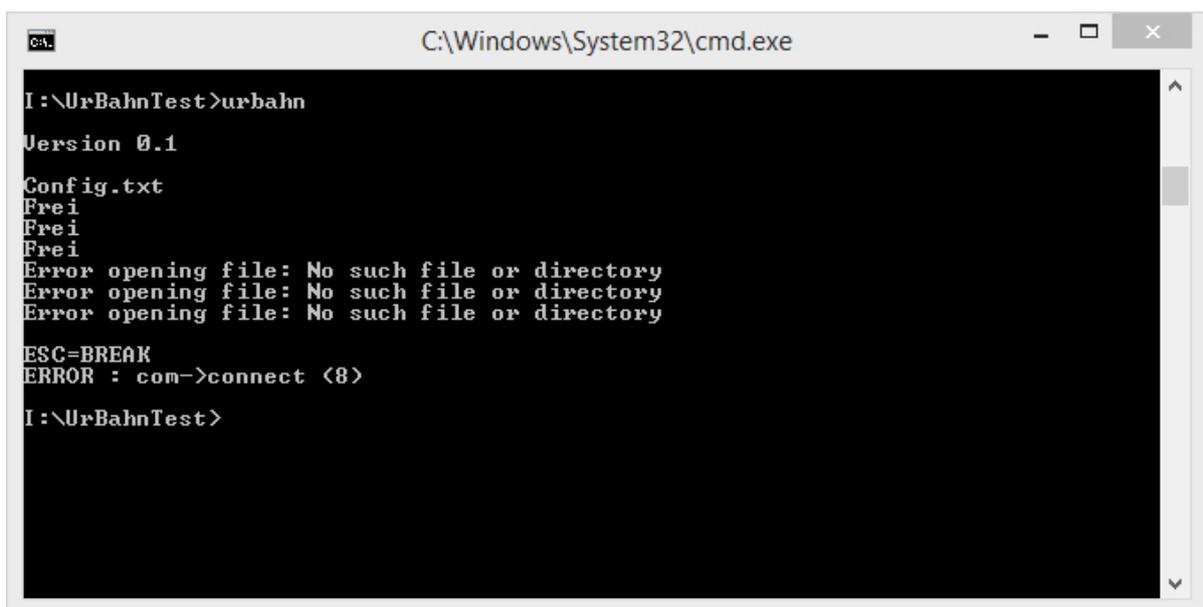
Abbildung 3: Erster Testaufruf

Außer der Programmversion und einer Reihe von Fehlermeldungen (dies ist hier völlig in Ordnung!) wird angezeigt, daß mit der `Config.txt` gearbeitet wird, die im selben Verzeichnis wie `UrBahn.exe` liegt.

Das Programm hat sich wegen gewisser Fehler unmittelbar selbst beendet.

5.2 Testaufruf mit laufendem 3D-Modellbahn Studio

Wir starten das 3D-Modellbahn Studio mit einer leeren Anlage (frei Platte) und wiederholen den Aufruf von `UrBahn.exe`:



```
C:\Windows\System32\cmd.exe

I:\UrBahnTest>urbahn

Version 0.1

Config.txt
Frei
Frei
Frei
Read Connect error
Read Connect error
Error opening file: No such file or directory
Error opening file: No such file or directory
Error opening file: No such file or directory

ESC=BREAK
ERROR : com->connect <8>

I:\UrBahnTest>
```

Abbildung 4: Testaufruf mit laufendem 3D-Modellbahn Studio

Die doppelte Fehlermeldung `Read Connect error` sollte verschwunden sein! Falls dies nicht

der Fall ist, läuft Ihr MBS möglicherweise auf einem separaten Rechner.

5.3 Separater Rechner für das 3D-Modellbahn Studio

Sie müssen dem Schnittstellenprogramm die Adresse des Rechners bekanntgeben, auf dem das MBS läuft.

Tragen Sie dazu die IP-Adresse des Rechners wie folgt in die Config-Datei ein:

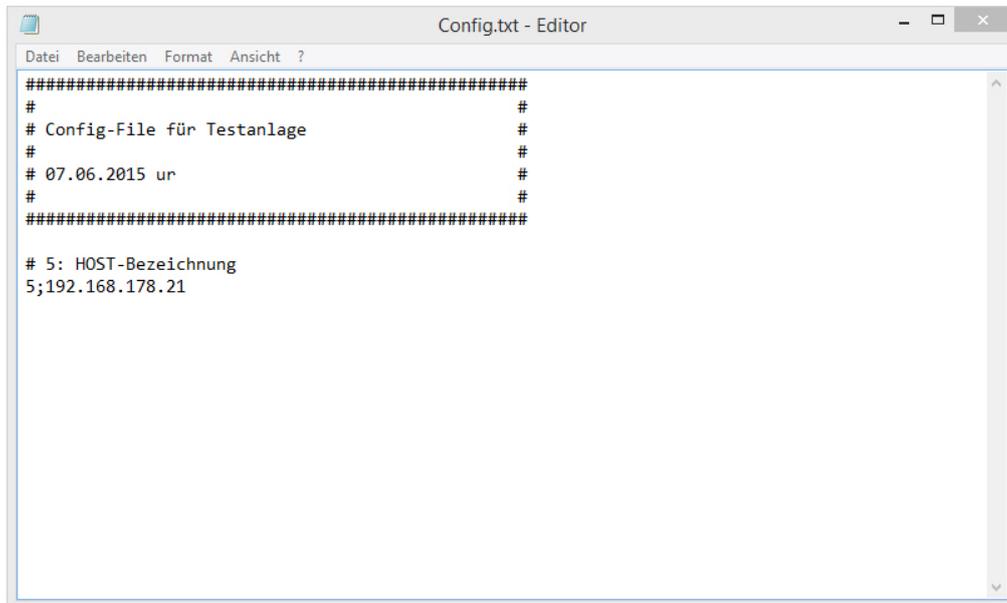


Abbildung 5: Config-File mit Eintrag für Netzkommunikation

Erklärung:

Die IP-Adresse ist die Nr. 5 in der Parameterliste. Weitere Parameter folgen (siehe z.B. 6.1 auf Seite 11).

Hinweise:

Gegebenenfalls müssen Sie UrBahn in der Firewall freischalten.

Im folgenden gehen wir davon aus, daß das MBS und UrBahn.exe auf einem gemeinsamen Rechner laufen und verzichten der Übersichtlichkeit halber auf den Eintrag Nr.5 in der Config-Datei.

Im nächsten Schritt beseitigen wir die drei Fehlermeldungen

"Error opening File: No such file or directory".

6 Anlage der Crossreferenz-Dateien

Da die P50X-Befehle mit Adressen arbeiten, das 3D-Modellbahn Studio hingegen symbolische Namen verwendet, müssen die Adressen in Namen übersetzt werden und umgekehrt. Dies erfolgt über Tabellen, die in Dateien abgelegt werden:

- eine Datei für Loks
- eine Datei für Weichen (, Signale, u.s.w.)
- eine Datei für Rückmelder.

Diese drei Dateien fehlen noch und sind der Auslöser für die Fehlermeldungen

"Error opening File: No such file or directory".

6.1 LokDatei

Legen Sie mit einem ASCII-Editor im erwähnten Unterverzeichnis eine Datei (z.B. `Loks.txt`) nach folgendem Muster an:

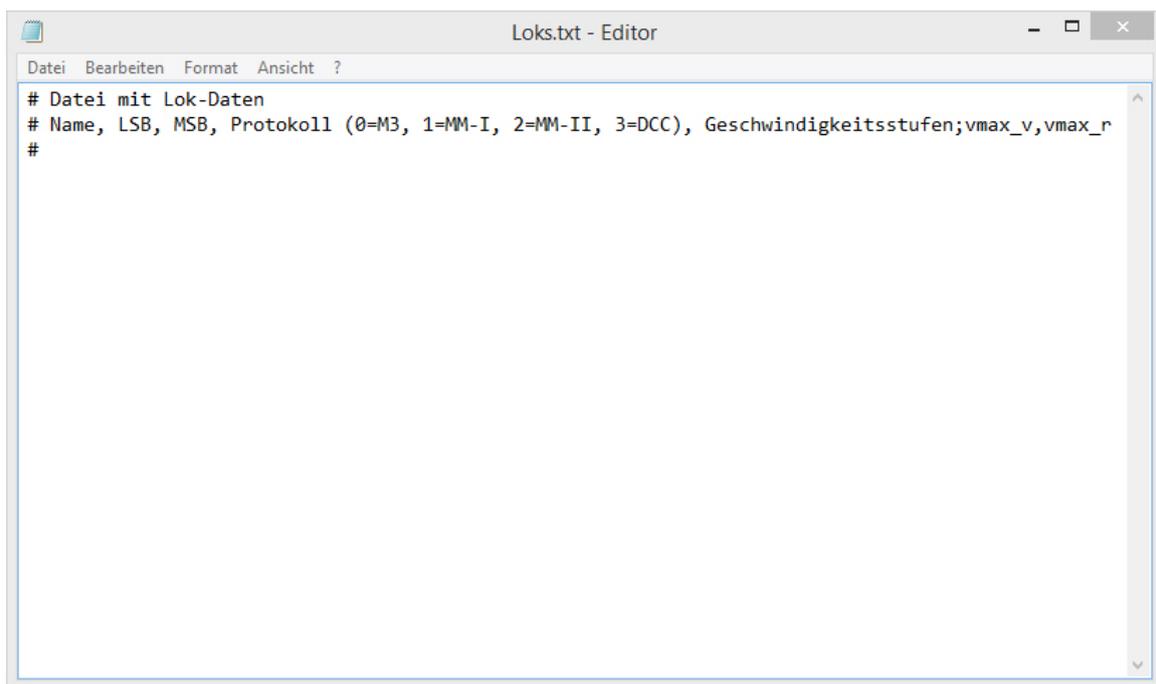


Abbildung 6: Leere Lokdatei

Erweitern Sie die Config-Datei um folgende Zeilen:

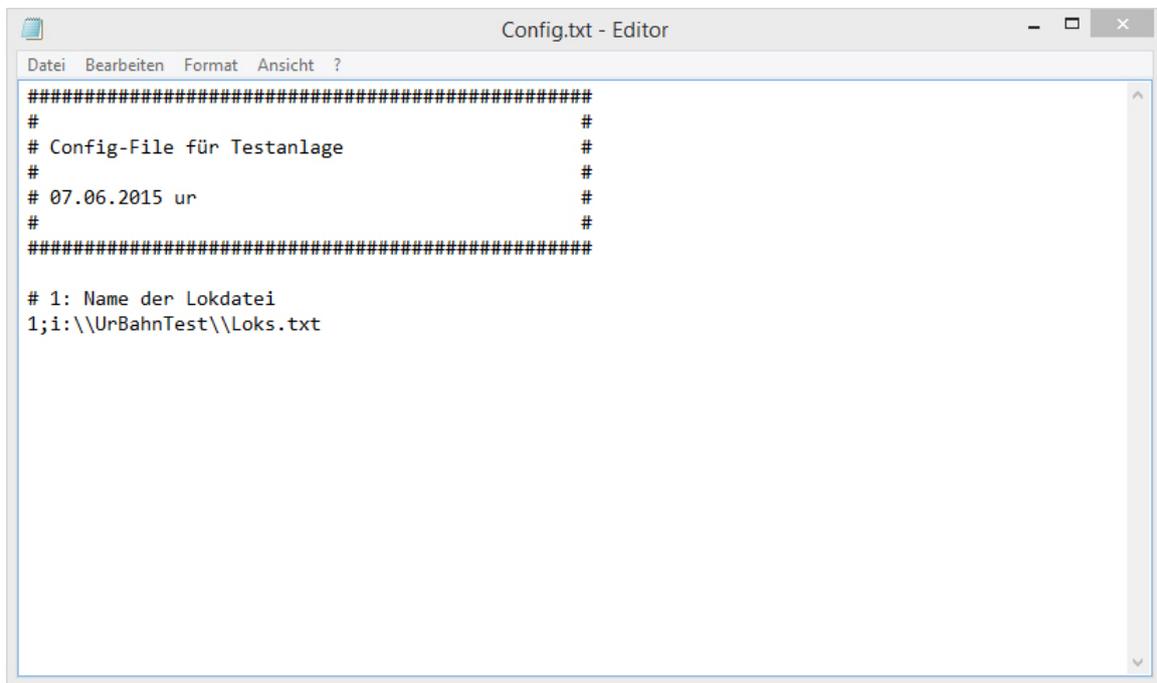


Abbildung 7: Config-File mit Eintrag für Lokdatei

Erklärung:

In der letzten Zeile steht die 1 dafür, daß der Pfad und Name der Lokdatei folgt.

Backslashes (\) müssen doppelt eingegeben werden.

6.2 Melderdatei

Legen Sie eine Datei (z.B. `Melder.txt`) nach folgendem Muster an:

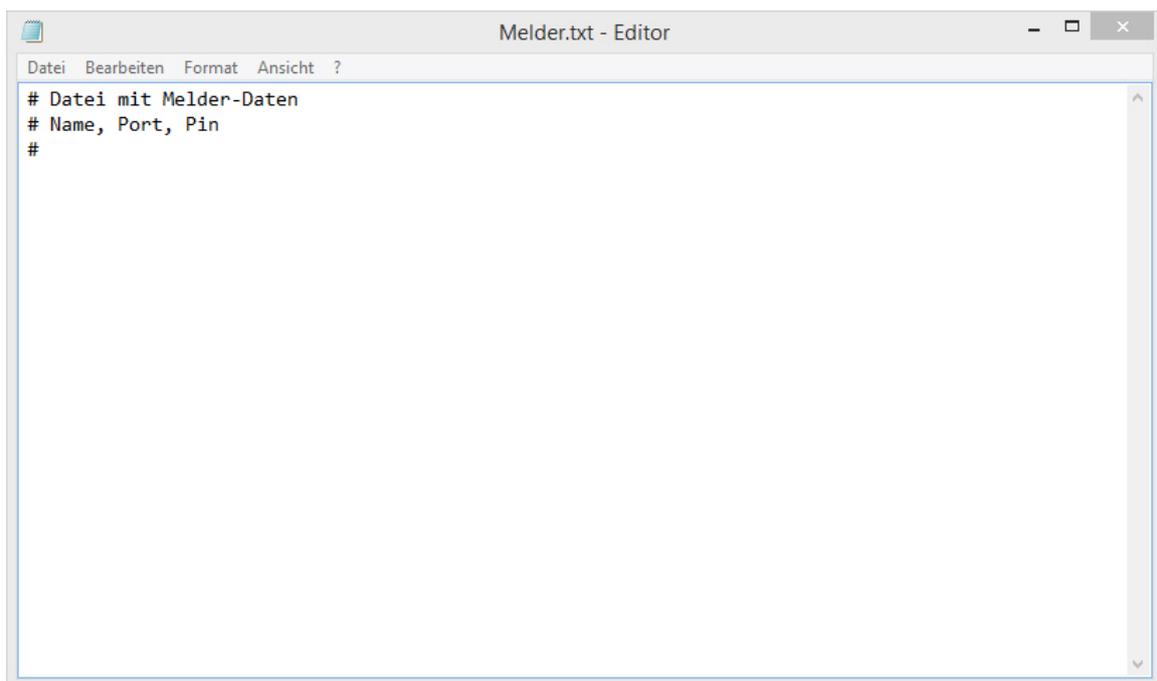


Abbildung 8: Leere Melderdatei

Tragen Sie den Namen dieser Datei als zweiten Parameter in der Config-Datei ein:

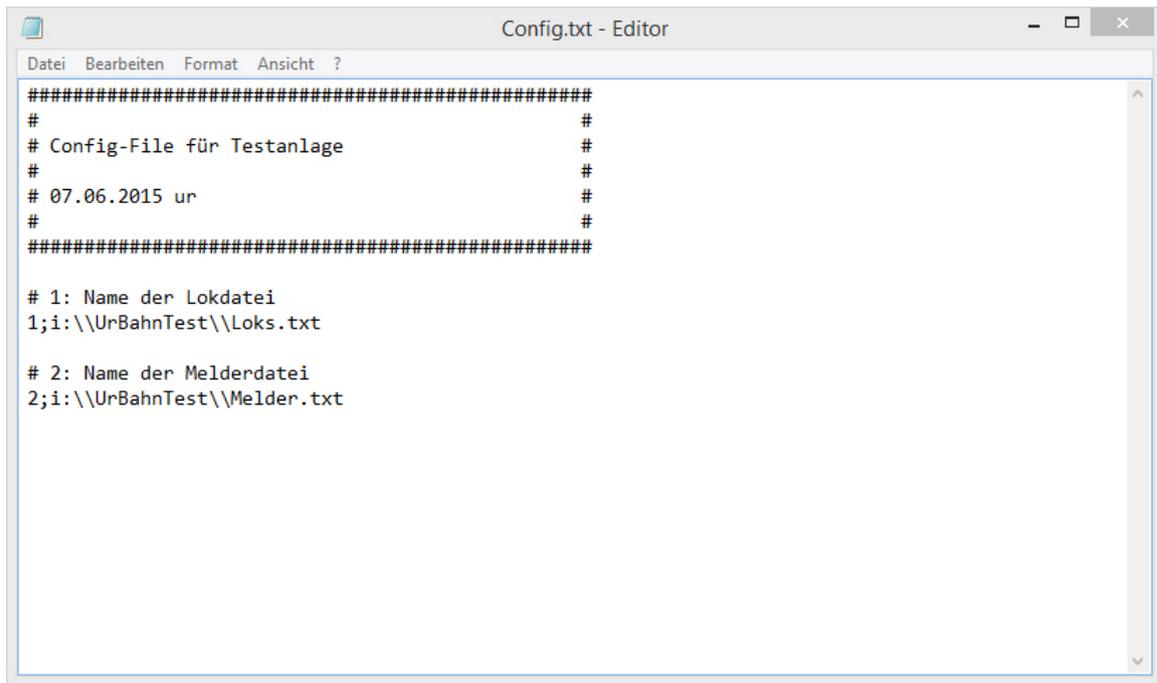


Abbildung 9: Config-File mit Eintrag für Melderdatei

6.3 Weichendatei

Legen Sie die Datei für Weichen an (z.B. `Weichen.txt`):

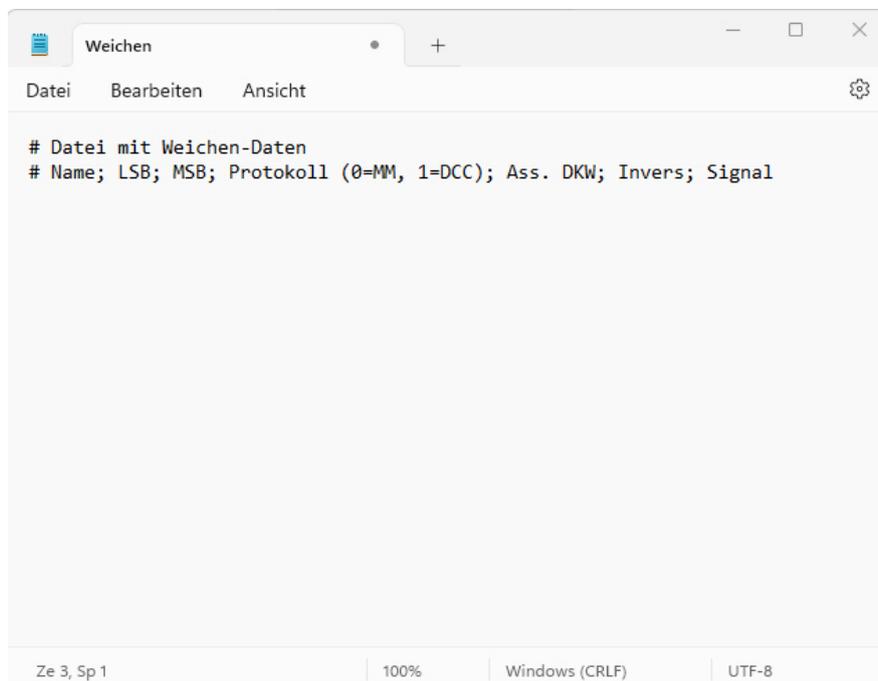


Abbildung 10: Leere Weichendatei

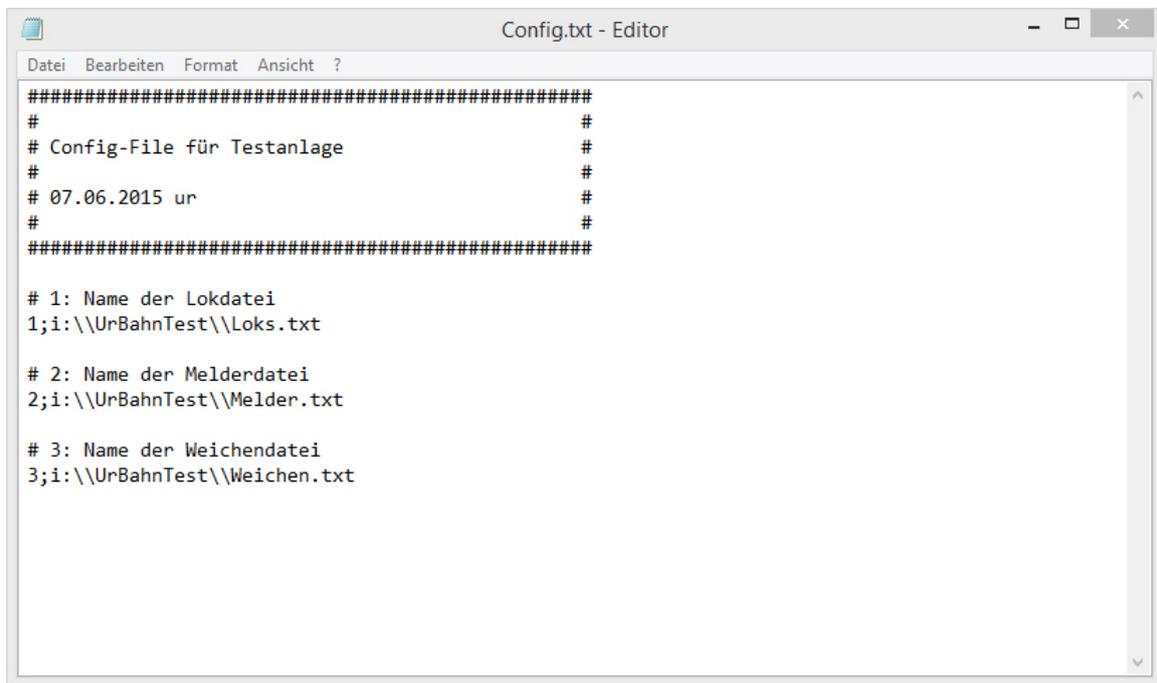


Abbildung 11: Config-File mit Eintrag für Weichendatei

Wir wiederholen den Aufruf von UrBahn.exe:

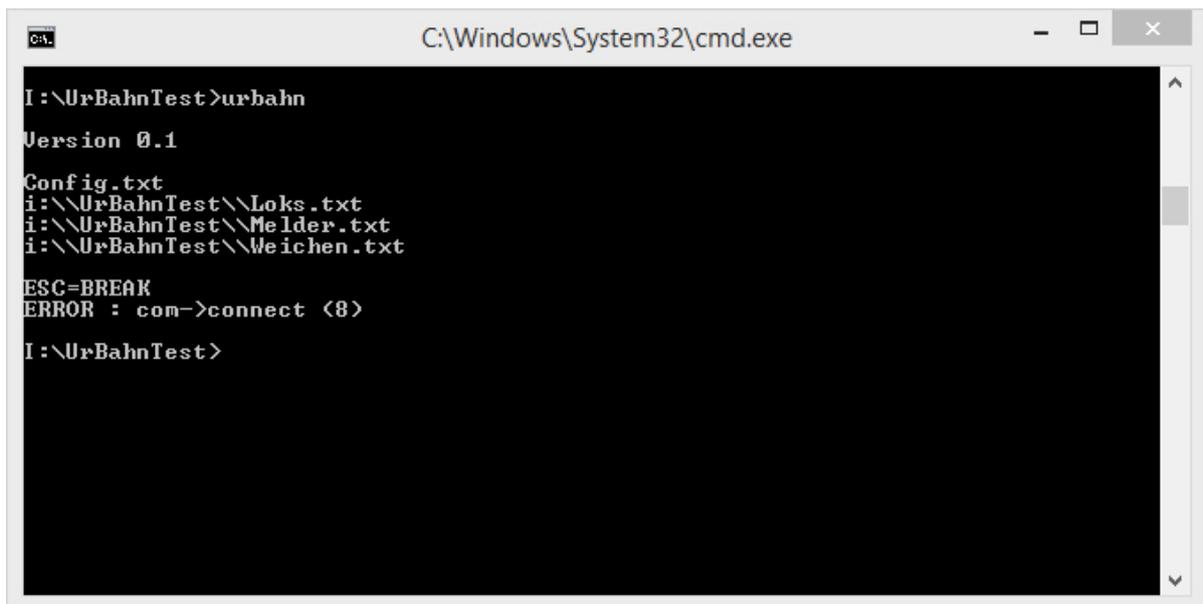


Abbildung 12: Aufruf von UrBahn.exe mit vorhandenen Konfigurationsdateien

Der letzte Fehler besagt, daß die serielle Schnittstelle (noch) nicht funktioniert. Dies beheben wir im Folgeschritt.

7 Einbinden der seriellen Schnittstelle

7.1 Zuordnung der seriellen Schnittstellen

Wir behandeln jetzt das Interface 1 in der Zeichnung unter 2.2 auf Seite 4. Dieses wird im Beispiel realisiert durch com0com:



Die Modellbahnsteuerung kommuniziert über eine serielle Schnittstelle (*COM_A*), die in der Modellbahnsteuerung eingestellt wird (z.B. COM8).

com0com stellt zwei intern verbundene serielle Schnittstellen zur Verfügung (*COM_B* und *COM_C*), von denen eine mit der Modellbahnsteuerung zu verbinden ist. Dies erfolgt dadurch, indem in com0com eine Schnittstelle genau so benannt wird wie die in der Modellbahnsteuerung ausgewählte Schnittstelle - im Beispiel also COM8 (siehe Abbildung 1 auf Seite 5).

COM_A muß also dieselbe Bezeichnung tragen wie COM_B.

Bleibt die Verbindung zwischen com0com und dem Schnittstellenprogramm (*COM_C* und *COM_D*).

Geben Sie in com0com der zweiten Schnittstelle einen Namen (COM1 bis COM9, sollte zweckmäßigerweise noch nicht belegt sein). Im Beispiel ist dies COM5.

7.2 Eintrag in der Config-Datei

Tragen Sie in der Config-Datei den Namen der seriellen Schnittstelle ein, über die das Schnittstellenprogramm mit com0com verbunden werden soll, im Beispiel also:

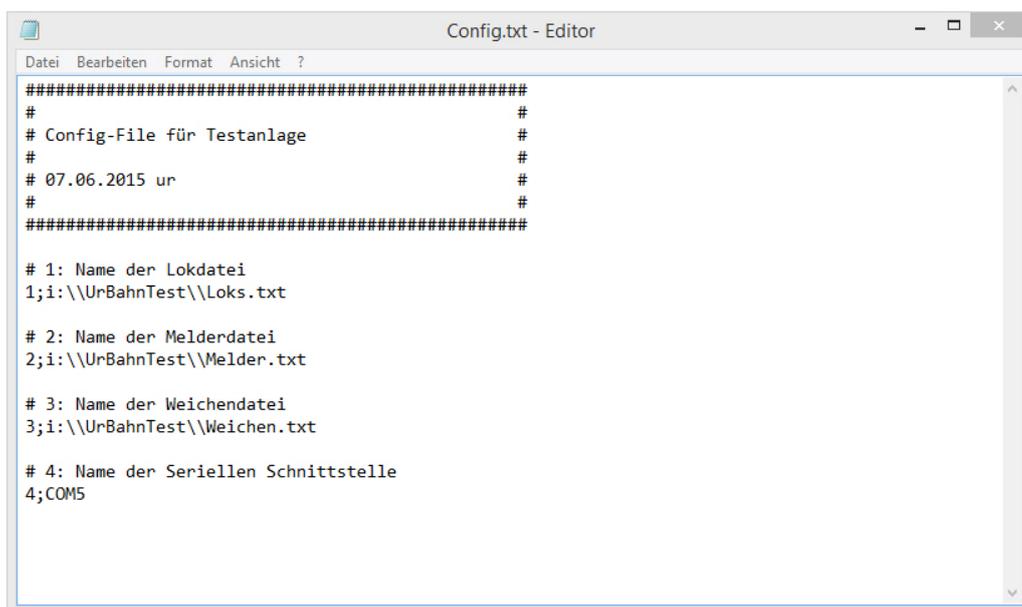
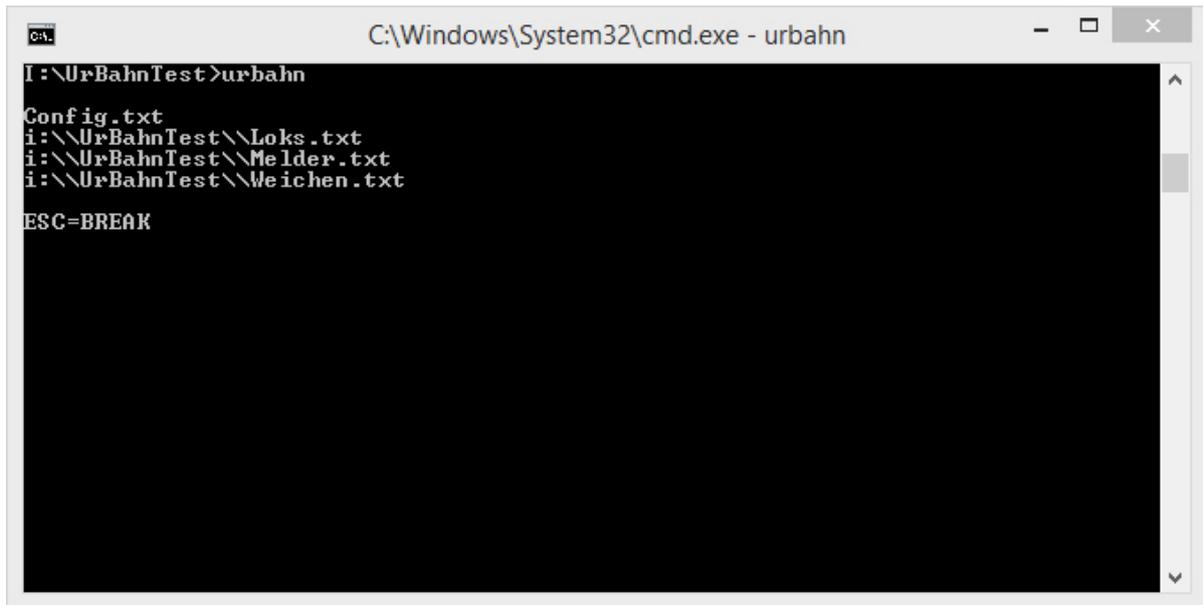


Abbildung 13: Config-File mit Eintrag für Serielle Schnittstelle

Ein weiterer Testaufruf von UrBahn.exe mit laufendem 3D-Modellbahn Studio zeigt keine Fehler mehr:



```
C:\Windows\System32\cmd.exe - urbahn
I:\UrBahnTest>urbahn
Config.txt
i:\UrBahnTest\Loks.txt
i:\UrBahnTest\Melder.txt
i:\UrBahnTest>Weichen.txt
ESC=BREAK
```

Abbildung 14: Finaler Testlauf

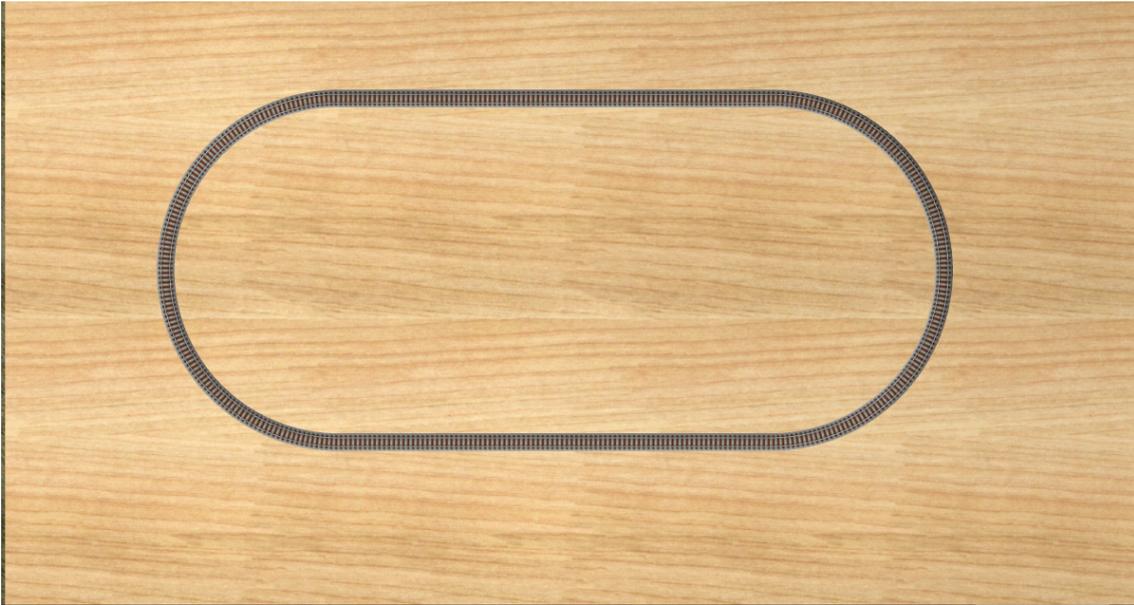
UrBahn.exe wird durch Drücken der Escape-Taste beendet.

Das System ist jetzt komplett.

Im folgenden setzen wir eine Testanlage auf.

8 Testanlage

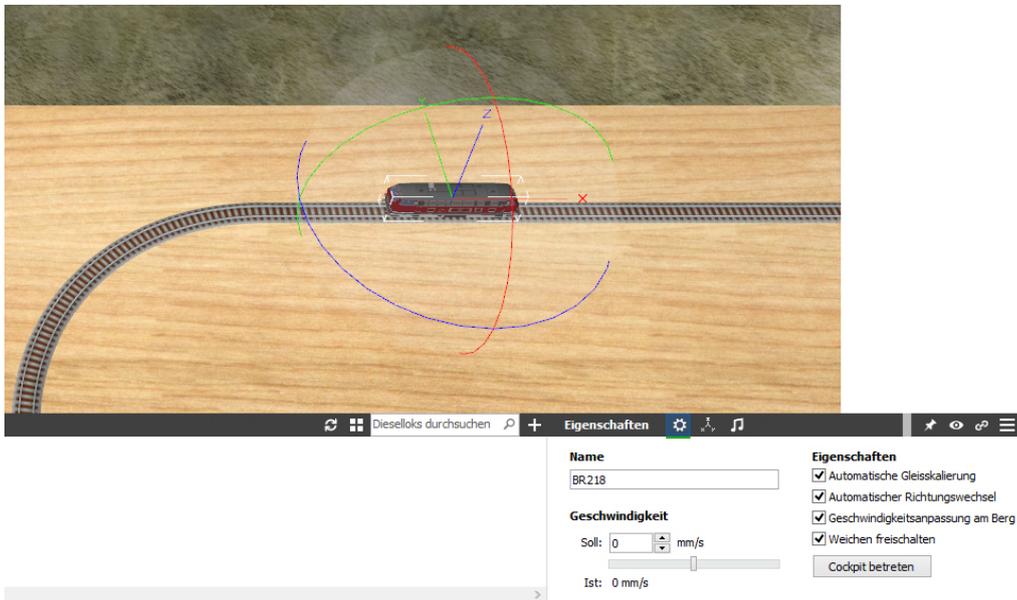
Erstellen Sie im 3D-Modellbahn Studio ein einfaches Gleisoval, z.B.:



Speichern Sie die Anlage z.B. unter dem Namen "UrBahnTest".

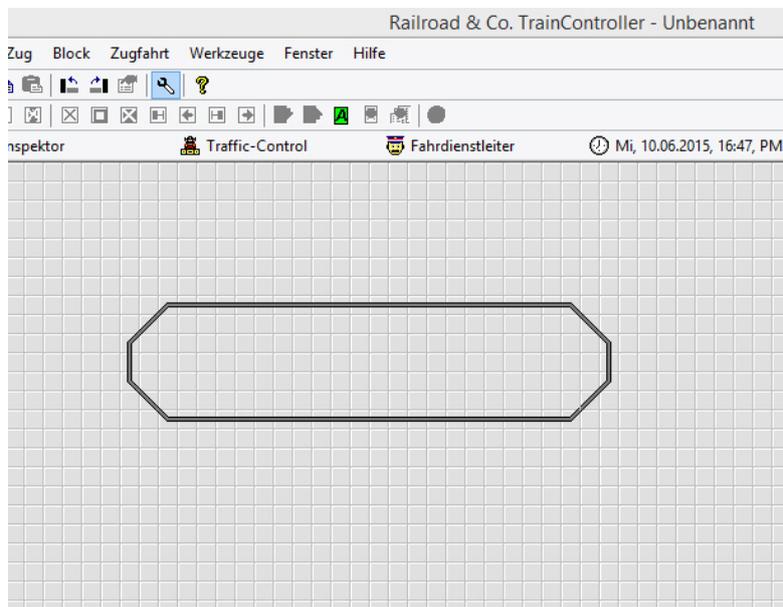
8.1 Testlok

Setzen Sie auf das Gleis eine Lok Ihrer Wahl und geben ihr einen einfachen Namen, z.B.:



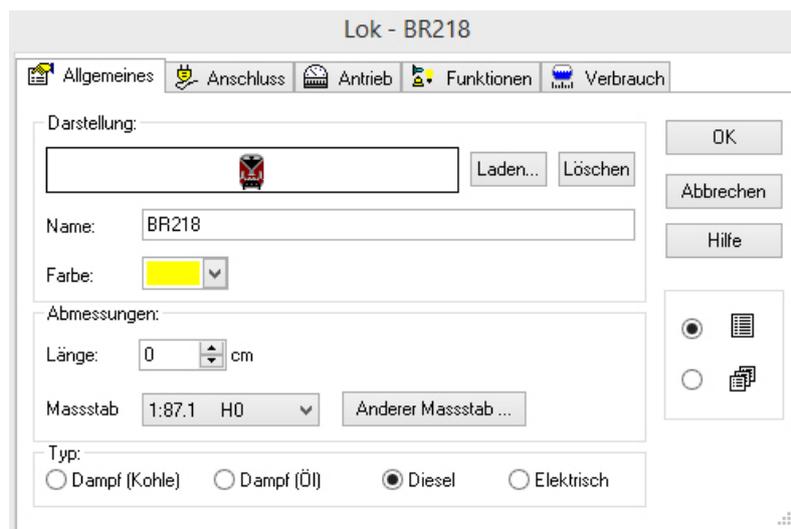
8.2 Testanlage in der Modellbahnsteuerung

Bauen Sie die Anlage in Ihrer Modellbahnsteuerung nach - im TrainController 5.8 z.B.:

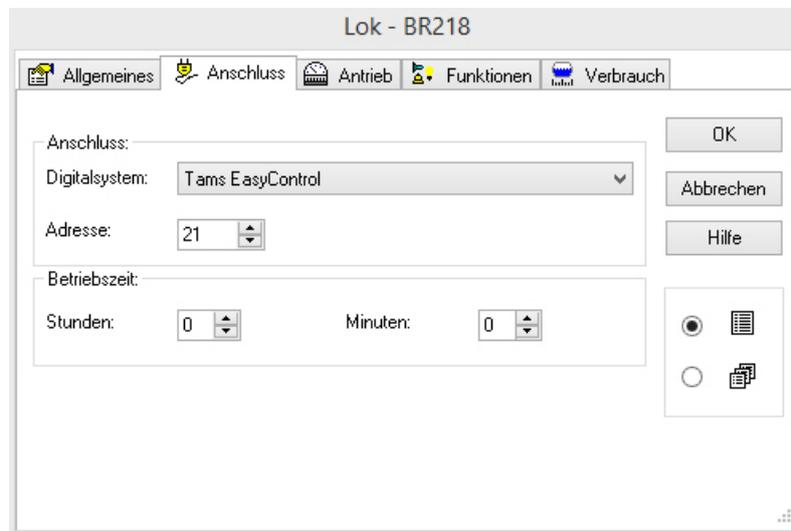


8.3 Testlok in der Modellbahnsteuerung

Legen Sie die Lok Ihrer Wahl in der Modellbahnsteuerung an - im TrainController 5.8 z.B.:

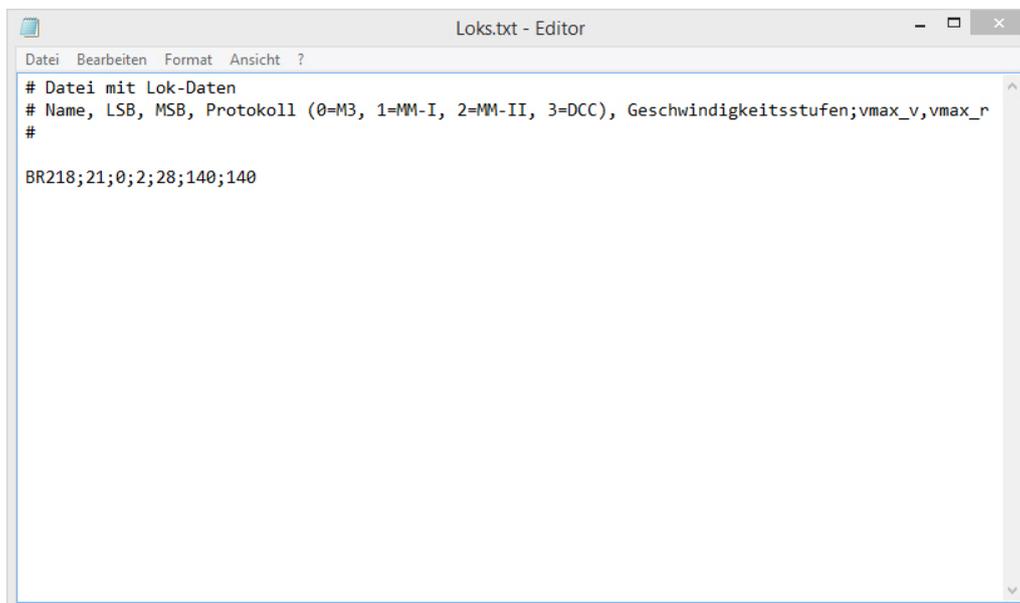


Geben Sie ihr eine Adresse, z.B. 21:



8.4 Eintrag in der Datei Loks.txt

Erweitern Sie die Datei Loks.txt um folgenden Eintrag:



Der Eintrag besagt:

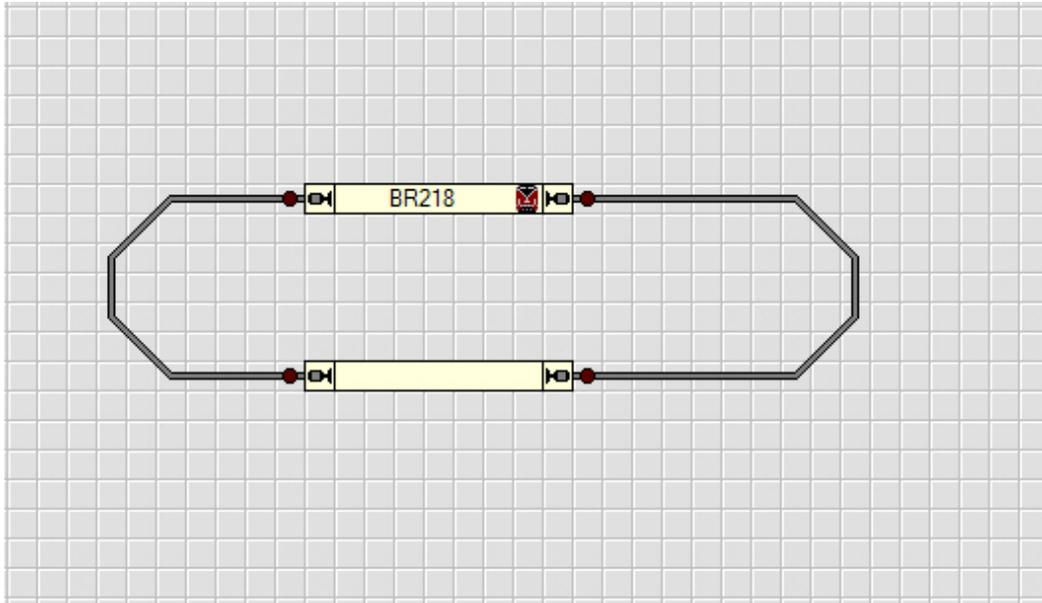
- Es gibt eine Lok mit dem Namen BR218;
- sie hat die Adresse 00 (MSB) 21 (LSB);
- sie versteht das Protokoll MM-II;
- sie hat 28 Geschwindigkeitsstufen;
- sie hat vorwärts wie rückwärts eine Maximalgeschwindigkeit von 140km/h.

8.5 Kontaktmelder

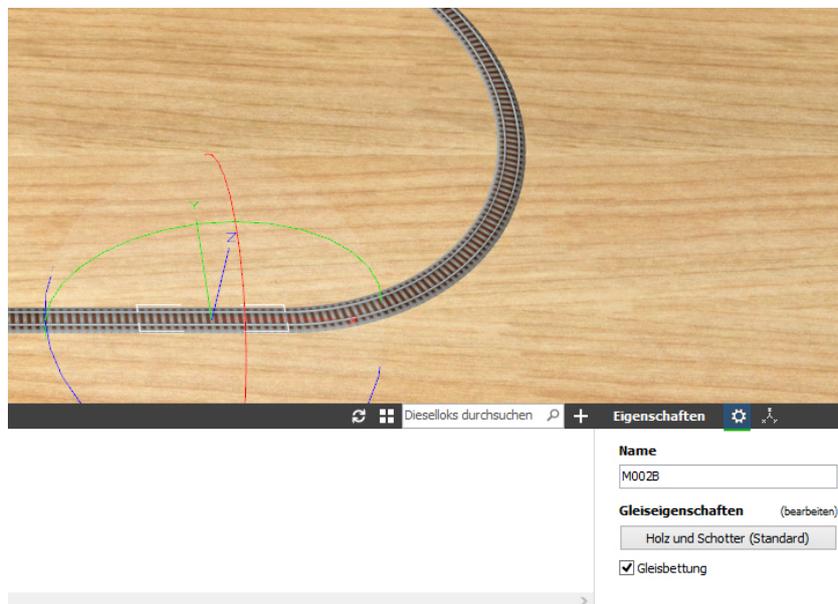
Spezifisch für TrainController:

Legen Sie zwei Traffic-Boxen an und ordnen Sie diesen je zwei Kontaktmelder zu - für den Block 1 z.B. die Melder M001A und M001B.

Setzen Sie die Lok in den Block1:



Geben Sie im 3D-Modellbahn Studio den korrespondierenden Gleisen den Namen des entsprechenden Melders, z.B.:



8.6 Eintrag in der Datei Melder.txt

Tragen Sie in der Datei Melder.txt die Melder mit ihren Namen und Adressen ein:

```
Melder.txt - Editor
Datei Bearbeiten Format Ansicht ?
# Datei mit Melder-Daten
# Name, Port, Pin
#
M001A;1;1
M001B;1;2
M002A;1;3
M002B;1;4
```

8.7 Testlauf

Speichern Sie alle Dateien ab und führen folgende Aktionen durch:

- Starten Sie das 3D-Modellbahn Studio muß mit der Anlage UrBahnTest;
- Starten Sie UrBahn.exe;
- Starten Sie Ihre Modellbahnsteuerung mit der Anlage UrBahnTest;
- die Modellbahnsteuerung muß online geschaltet sein.

Jetzt können Sie die Lok der 3D-Modellbahnsteuerung mit Ihrer Modellbahnsteuerung kontrollieren:

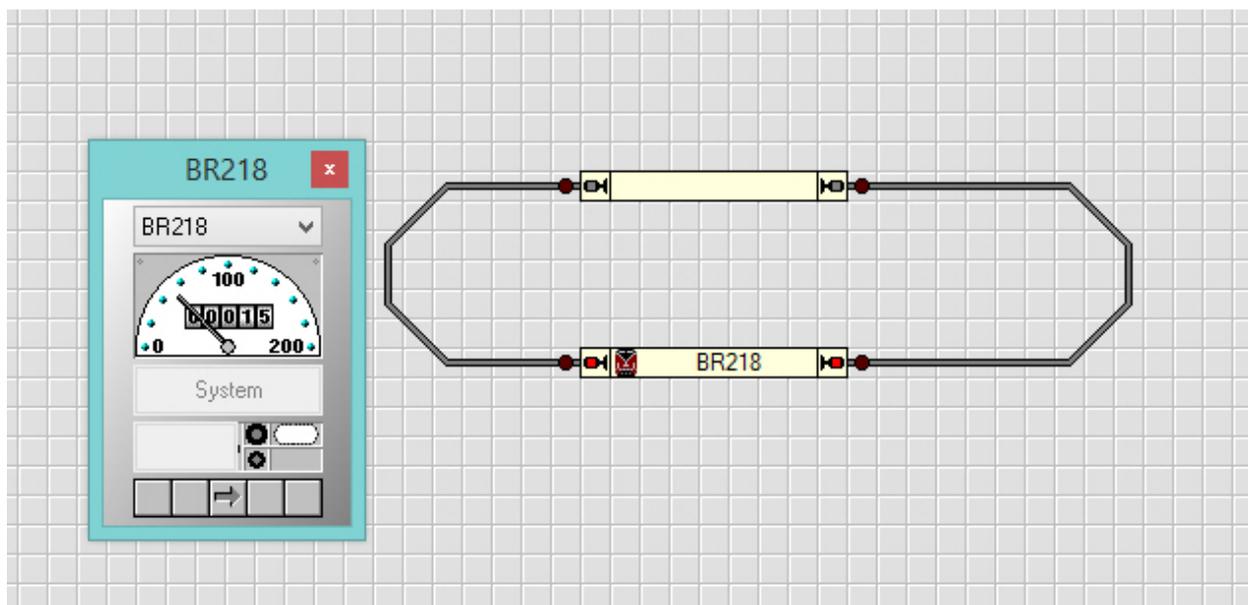
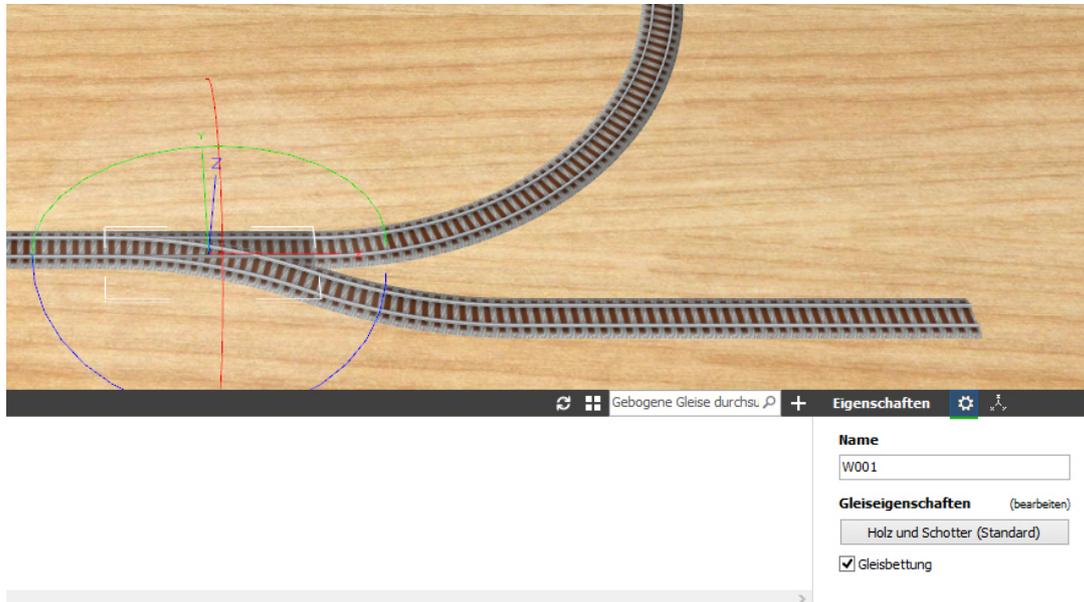


Abbildung 15: Testlauf

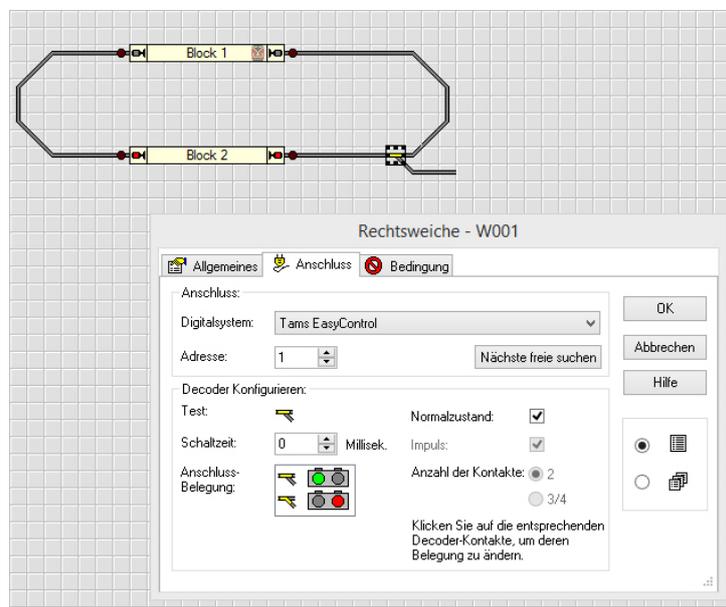
9 Weichen und ähnliches

9.1 Einfache Weiche

Fügen Sie in dem 3D-Modellbahn Studio eine Weiche in das Gleisbild ein und geben Sie ihr einen spezifischen Namen, z.B. W001:



Legen Sie die entsprechende Weiche in Ihrer Modellbahnsteuerung an - z.B. im TrainController:



Der Datei Weichen.txt fügen Sie folgenden Eintrag hinzu:

```

# Datei mit Weichen-Daten
# Name; LSB; MSB; Protokoll (0=MM, 1=DCC); Ass. DKW; Invers; Signal
#
# W001; 1; 0; 0; 0; 0; 0
W001;1

```

Die einzelnen Parameter:

Name	Frei vergebbarer Name der Weiche
LSB	Unterer Adressteil
MSB	Oberer Adressteil
Protokoll	Motorola oder DCC
Ass. DKW	Adresse der assoziierten Doppelkreuzungsweiche (siehe Folgeabschnitt)
Invers	Damit können die Steuersignale invertiert werden
Signal	Parameter wird für den MBP benötigt.

Erklärung:

In der auskommentierten (mit # beginnend) Zeile "# W001;..." steht die *Langform* (gut zu lesen, aber eben lang);

Da alle numerischen Parameter, die nicht explizit angegeben werden, mit 0 vorbesetzt sind, hat die *Kurzform* in der darunter stehenden Zeile die gleiche Bedeutung[, ist aber kürzer].

UrBahn.exe muß neu gestartet werden, um den neuen Dateieinhalt einzulesen. Anschließend können Sie die Weiche über Ihre Modellbahnsteuerung schalten.

9.2 Doppelkreuzungsweiche

Doppelkreuzungsweichen mit **einem** Antrieb werden wie Weichen behandelt.

Doppelkreuzungen mit zwei Antrieben haben i.d.R. zwei Adressen, die beispielhaft wie folgt in der Datei Weichen.txt verknüpft werden:

```

DKW003;13;0;0;14
DKW003A;14;0;0;13

```

Erklärung:

DKW003 mit der Adresse 13 hat die Sekundäradresse 14; dahinter verbirgt sich die virtuelle DKW003A mit der Adresse 14 und der Sekundäradresse 13.

Beide Zeilen sind verknüpft und gestatten so die Steuerung einer DKW mit zwei Antrieben.

Bemerkung:

Es scheint so, daß die Märklin-DKWs in der MBS mit **zwei** Antrieben angelegt sind, obwohl sie realiter **einen** Antrieb haben (jedenfalls meine - zugegeben - fossilen Teile). Es gibt aber im Forum eine Anleitung, wie sie passend modifiziert werden können.

9.3 Signale

Das Signal-Kommando 402: Setzt den Zustand eines Signals zeigte bei mir keine Wirkung. Daher bin ich folgenden Weg gegangen:

- In der Modellbahnsteuerung wird das Signal an eine virtuelle Weiche gebunden;
- diese virtuelle Weiche wird in der MBS angelegt und in Weichen.txt deklariert;
- in der MBS wird das Signal mit der virtuellen Weiche verbunden.

Signale werden also logisch wie 'umgedungelte' Weichen behandelt.

Nachtrag:

Das Signal-Kommando 402 funktioniert jetzt, wenn in der Weichendatei das Signal als solches gekennzeichnet ist (Parameter "Signal" = 1).

9.4 Entkupplungsgleise und weitere Teile mit Antrieb

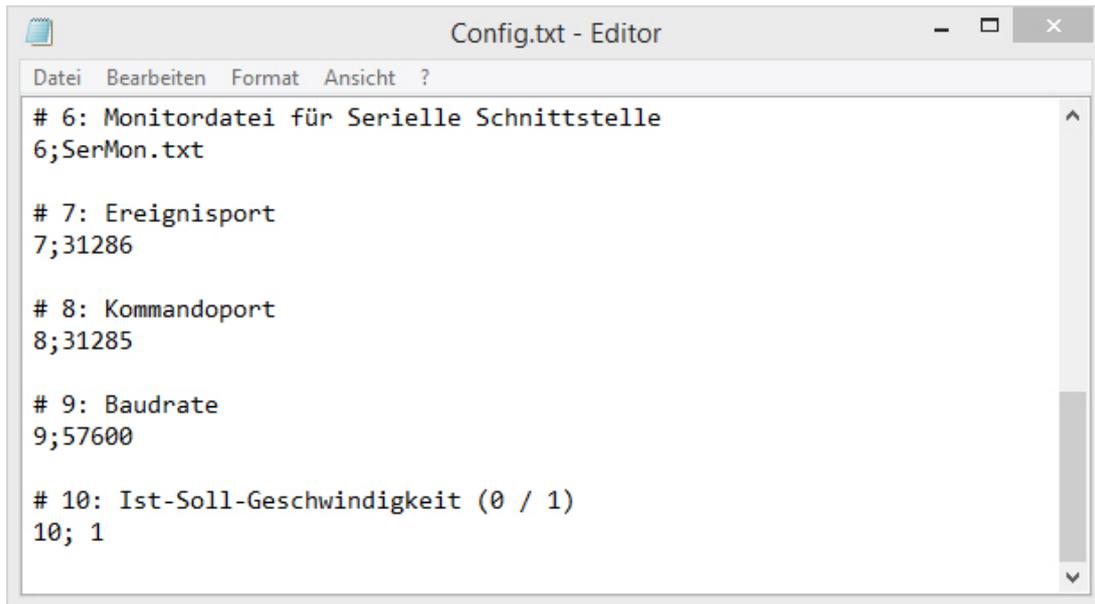
Diese werden ebenfalls als Weichen angesteuert.

Die Auswirkungen in der MBS sind individuell zu programmieren (z.B. Abkuppeln einer Lok in der MBS durch Betätigen eines Entkupplungsgleises in der Modellbahnsteuerung).

Hier wäre der Bedarf zu klären.

10 Sonstiges

Bei Bedarf können Sie die Config-Datei um folgende Einträge erweitern:



Wenn Sie dies nicht tun, verwendet das Programm Default-Werte.

10.1 Monitordatei für Serielle Schnittstelle

```
# 6: Monitordatei für Serielle Schnittstelle
6;SerMon.txt
```

Wenn Sie hier einen Dateinamen eintragen (z.B. „SerMon.txt“), wird die serielle Kommunikation in dieser Datei protokolliert.

Hinweis: Eine Protokollierung benötigen Sie normalerweise nicht. Daher sollten die Zeilen üblicherweise wie folgt aussehen:

```
# 6: Monitordatei für Serielle Schnittstelle
6;
```

10.2 Ereignis- und Kommandoport

```
# 7: Ereignisport
7;31286
```

```
# 8: Kommandoport
8;31285
```

Das 3D-MBS verwendet zur Zeit genau diese Ports. Sie sollten die Einträge nicht ändern (höchstens weglassen).

10.3 Baudrate

9: Baudrate
9;57600

Bei Verwendung eines USB-Anschlusses lassen Sie den Eintrag weg oder verwenden die Baudrate 57600 baud.

Bei Verwendung einer seriellen Schnittstelle können Sie die Baudrate vorgeben.

10.4 Ist/Soll-Geschwindigkeit

10: Ist-Soll-Geschwindigkeit (0 / 1)
10; 1

Bei einer 0 wird die vom Kontrollprogramm übertragene Geschwindigkeit direkt umgesetzt.

Bei einer 1 wird "Masse" simuliert - die Fahrzeuge beschleunigen und bremsen *geschmeidiger*.