

Inhaltsverzeichnis

- [1 Vorbemerkungen](#)
 - [1.1 Verwendete Kürzel:](#)
- [2 Session Rules](#)
 - [2.1 MissionCode manager \(<kuid2:61392:8200>\)](#)
 - [2.2 Mission Code Stops and TimeTables Manager \(<kuid:61392:8300>\)](#)
 - [2.3 IT Enhanced Manager \(<kuid2:61392:8130>\)](#)
- [3 Driver Commands](#)
 - [3.1 IT SetPath \(<kuid2:61392:8111>\)](#)
 - [3.2 IT SetPathAndWaitActive \(<kuid2:61392:8113>\)](#)
 - [3.3 IT SetPathAndWaitNotRed \(<kuid2:61392:8112>\)](#)
 - [3.4 IT CancelEnhancedPath \(<kuid2:61392:8114>\)](#)
 - [3.5 ITMCAssignNextPath \(<kuid2:61392:8202>\)](#)
 - [3.6 AutoDriveToMCPathStop \(<kuid2:61392:8321>\)](#)
 - [3.7 SetMCTimeTable \(<kuid:61392:8330>\)](#)
 - [3.8 AutoDriveWithMCTimeTable \(<kuid2:61392:8322>\)](#)
 - [3.9 SetMissionCode \(<kuid2:61392:8201>\)](#)
 - [3.10 AddMissionCode \(<kuid2:61392:8203>\)](#)
 - [3.11 RemoveMissionCode \(<kuid2:61392:8204>\)](#)
- [4 Trackside und Trackmark Objects](#)
 - [4.1 Enhanced TRC3 invisible Interlocking Tower \(<kuid2:61392:8101>\)](#)
 - [4.2 EIT PathTrigger \(<kuid2:61392:8121>\) \(Trackmark\)](#)
 - [4.3 MCPPathStop \(<kuid2:61392:8311>\) \(Trackmark\)](#)
 - [4.4 Track Circuit Insulator \(<kuid2:>\) \(Trackmark\)](#)
 - [4.5 Track Circuit Detector \(<kuid2:>\) \(Trackmark\)](#)
- [5 ScriptLibraries\(Zubehör\)](#)
 - [5.1 EnhancedInterlockingTowers library \(<kuid2:61392:8190>\)](#)
 - [5.2 TRC 3 InterlockingTower library \(<kuid2:61392:8191>\)](#)
 - [5.3 MissionCodeLibrary \(<kuid2:61392:8290>\)](#)

EIT- und MC-System ist eine Übersicht zu den Assets aus dem Bereich Mission Code und Interlocking Tower. Antworten auf noch offene Fragen werden im Forum-Thread "EIT- und MC-System - Lexikonbeitrag - offene Fragen" diskutiert, solange sie noch nicht aktualisiert sind.

1 Vorbemerkungen

(Ausgangs-Version 2.1)

1.1 Verwendete Kürzel:

TM Trackmark (Wegpunkt)

PS Path Stop (Fahrstraßen-Haltepunkt)

IT Interlocking Tower (Stellwerk)

EIT Enhanced Interlocking Tower (Erweitertes Stellwerk, Verbessertes Stellwerk)

DC Driver Command (Fahrbefehl)

MC Mission Code (MC-Bezeichner, Aufgaben Bezeichner)

MCM Mission Code Manager (MC-Verwalter, Aufgaben Bezeichner Verwalter)

TT TimeTable (Zeit-Fahrplan)

MCS Mission Code Stop (MC-Stopp, MC-Haltepunkt)

ITEM IT Enhanced Manager (ITE-Verwalter)

TRC Trigger Rail Crossing (Bahnübergangssteuerung mit dem TRC V.3 System: Siehe [TRC V3 ! The Trigger Rail Crossing System](#) im [L'atelier de Vincent](#))

Quellen:

- http://online.ts2009.com/media...php/Driver_Commands_List (english)
- http://online.ts2009.com/media...Rule_List_with_Parameters (english)
- [Mission Codes Manager Rule](#) (english)
- [Mission Code Manager Regel](#) (deutsch)
- [Mission Code Stops and TimeTables Manager Rule](#) (english)
- [Mission Code Stops and TimeTables Manager Regel](#) (deutsch)
- http://online.ts2009.com/media...hanced_Interlocking_Tower (english)
- [Enhanced Interlocking Tower Manager Rule](#) (english)
- [Enhanced Interlocking Tower Manager Regel](#) (deutsch)

2 Session Rules

Diese Regeln müssen im Menu *Edit Session...* hinzugefügt werden und können dort bearbeitet werden.

2.1 MissionCode manager (<kuid2:61392:8200>)

Der MCM erlaubt das Hinzufügen, Bearbeiten und verwalten von MC- Bezeichnern (im Original *strings*, was der programmiertechnische Begriff für eine Zeichenkette bzw. ein kleines Textstück ist), welche Zügen zugeordnet werden können, um sie an ihr Ziel zu bringen, in dem automatisch Fahrstraßen aus ITs ausgewählt werden.

Ein solcher MC-Bezeichner ist ein aus einer Liste von frei wählbarer MC- Bezeichner, welche vorher mit Hilfe der MMCM-Regel im Surveyor-Modus erfasst werden müssen. Nachdem alle MCs, die in der Session

benutzt werden sollen, erfasst sind, kann man einem oder mehreren vorhandenen Zügen diese MCs auf zwei Weisen zuordnen:

- Durch die Benutzung des Trains-Tabs bei der Bearbeitung im MCM,
- Mit dem *SetMissionCode*- bzw. *AddMissionCode*-DC.

Standardmäßig wird keinem Zug ein MC zugeordnet, aber ihm können einer oder mehrere Mission Codes zugeordnet werden. Alle MCs bleiben solange zugeordnet, bis die MC-Liste des Zuges mit dem *SetMissionCode*-DC bereinigt wird oder der MC mit dem *RemoveMissionCode*-DC entfernt wird. Die den Zügen zugeordneten MCs werden gespeichert und beim Laden der Session wieder hergestellt. Mehreren Zügen kann der gleiche MC parallel zugeordnet sein.

Nach der Festlegung der MC-Bezeichner kann man beim Bearbeiten des MCM jede Fahrstraße jedes vorhandenen ITs verfügbar machen, einschließlich möglicher Einstellungen (Options). Diese nun im MCM verfügbaren Fahrstraßen können, ggf. auch mehrfach verschiedenen, MCs zugeordnet werden.

Später im Driver Modus, wenn ein Zug mit Mission Code das Eingangssignal einer im MCM verfügbaren und dem Mission Code zugeordneten Fahrstraße erreicht, übernimmt der MCM an Stelle des Interlocking Tower die Verwaltung dieser Fahrstraße. Im Driver Modus funktioniert der MCM als Lauscher, der auf Mitteilungen der im MCM verfügbaren Interlocking Tower wartet, und mit ihnen die Fahrstraßenverwaltung abstimmt. Der MCM reagiert nur auf Fahrstraßen, die einem mission Code des Zuges zugeordnet sind.

Quelle:

http://online.ts2009.com/media....php/MissionCodes_Manager (english)

[Mission Code Manager Regel](#) (deutsch)

2.2 Mission Code Stops and TimeTables Manager (<[kuid:61392:8300](#)>)

Die *Mission Code Stops and TimeTables Manager*-Regel (MCS&TTM-Regel) erlaubt das Erzeugen, Bearbeiten und Verwalten von Haltezonen und Zeitfahrplänen, um einfach Züge einem Zeitfahrplan (TT) entlang einer Fahrstraßenfolge mit Hilfe von MCs zu realisieren.

Eine Stopzone ist eine Gruppe von Wegpunkten (TMs oder PSs), an denen ein Zug anhält, wenn er in seinem Zeitfahrplan (TT) vorgesehen ist. Eine Stopzone ist einfach ein logischer Name (Bezeichner, string), unter dem verschiedene PSs oder TMs zusammengefasst (aufgelistet) sind. Oft werden sprechende Stopzonennamen wie Stationsnamen oder Industrienamen verwendet, aber es kann auch jeder eindeutige Name dafür verwendet werden.

TTs werden im MCS&TTM erstellt und benannt und enthalten alle ihnen zugeordneten Stopzonen zusammen mit den Ankunfts- und Abfahrzeiten (arrival times and departure times). Die Ankunftszeit ist optional und wird nur als Hinweis auf die erwartete Ankunft betrachtet. Die Abfahrtszeit muss angegeben werden, außer für die Endstelle. Es gibt die Möglichkeit zu veranlassen, dass der Zug in der Ziel-Stopzone die Fahrtrichtung wechselt. Ebenfalls wird der MC festgelegt, welchem der Zug folgt, wenn er nach diesem Fahrplan fährt.

Nach der Festlegung der Stopzonen und Fahrpläne können Züge mit Hilfe des *SetMCTimeTable*-DC und dem *AutoDriveWithMCTimeTable*-DC veranlasst werden, einem solchen Fahrplan automatisch zu folgen. Wenn ein Zug bereits an einer Stopzone des zugeordneten Fahrplanes hält, folgt er dem Fahrplan ab dieser Stopzone. Der Zug wartet auf die Abfahrtszeit und ändert ggf. auch seine Fahrtrichtung.

(??? also Fahrtrichtungswechsel auch an Nicht-Endstellen? ???)

Das *AutoDriveWithMCTimeTable*-DC endet an der Endstelle des gesetzten Fahrplanes.

Mit Stopzonen und Zeitfahrplänen wird es möglich, mit nur zwei DCs eine Zeitfahrplan-Aufgaben für eine Session zu erstellen: Einem DC zum Zuweisen eines Zeitfahrplanes zum Zug (*SetMCTimeTable*) und einem DC, um automatisch diesem Zeitfahrplan zu folgen (*AutoDriveWithMCTimeTable*).

Das System ist einfach, um Personen-Verkehr mit Hilfe von Stationen als Stopzonen zu realisieren, aber man kann auch Gleisgruppen (yards) als Stopzonen für Gütertransporte ebenso wie Industrien einrichten.

Quelle:

http://online.ts2009.com/media...ps_and_TimeTables_Manager (english) [Mission Code Stops and TimeTables Manager Regel](#) (deutsch)

2.3 IT Enhanced Manager (<kuid2:61392:8130>)

Die *Enhanced Interlocking Tower Manager*-Regel (*IT Enhanced Manager*-Regel, *ITEM*-Regel) gehört zum EIT-Paket. Sie erlaubt im Surveyor-Modus die zentrale Verwaltung der in der Route enthaltenen Stellwerke und im Driver-Modus die Anzeige von Informationen zu Zustand der Stellwerke, Fahrstraßen und Züge. Die Regel arbeitet mit den Standard Stellwerken (ITs) und mit den Verbesserten Stellwerken (EITs).

Achtung! Die verschiedenen Bezeichnungen sind zwar unglücklich, aber werden in den Originaltexten leider so auch gemischt. Als Kürzel wird hier *ITEM*-Regel verwendet.

Im Surveyor-Modus sind die Informationen zu den Stellwerken die gleichen, wie die in den einzelnen Eigenschafts-Dialogen der ITs und EITs. Aber die *ITEM*-Regel bietet die Möglichkeit, das alles zentral zu verwalten.

Im Driver-Modus können die Anzeige-Informationen zu den Its, EITs, Fahrstraßen und Zügen hilfreich sein, aktuelle Zustände zu überblicken bzw. sogar Probleme mit Fahrstraßen zu erkennen.

Die Informationsanzeige im Driver-Modus findet man im Tools-Menü unter *Runtime monitors...* im Menüpunkt *View towers manager*.

Quelle:

http://online.ts2009.com/media...ocking_Tower_Manager_Rule (english)

[Enhanced Interlocking Tower Manager Regel](#) (deutsch)

3 Driver Commands

Diese Fahrkommandos werden den Fahrern entweder über das Menü Session Edit ... in der Driver-Regel vorgegeben und stehen dann zum Sessionstart (und ggf. dem Rücksetzen aller Kommandos) als Fahraufträge bereit oder können im Driver-Modus den Fahrern direkt gegeben werden.

3.1 IT SetPath (<kuid2:61392:8111>)

Setzt die eingestellte Fahrstraße aus einem Interlocking Tower (IT) oder einem Enhanced Interlocking Tower (EIT). Stammt die Fahrstraße aus einem EIT, kann sie mit dem *IT CancelEnhancedPath*-DC vor ihrer Benutzung freigegeben werden.

3.2 IT SetPathAndWaitActive (<kuid2:61392:8113>)

Setzt die eingestellte Fahrstraße aus einem Interlocking Tower (IT) oder einem Enhanced Interlocking Tower (EIT) und wartet bis die Fahrstraße aktiv wird. Stammt die Fahrstraße aus einem EIT, kann sie mit IT CancelEnhancedPath Command vor ihrer Benutzung freigegeben werden.

3.3 IT SetPathAndWaitNotRed (<kuid2:61392:8112>)

Setzt die eingestellte Fahrstraße aus einem Interlocking Tower (IT) oder einem Enhanced Interlocking Tower (EIT) und wartet bis das nächste Signal nicht mehr Halt zeigt. Stammt die Fahrstraße aus einem EIT, kann sie vor ihrer Benutzung mit *IT CancelEnhancedPath* aus der Liste des Zuges entfernt werden.

3.4 IT CancelEnhancedPath (<kuid2:61392:8114>)

Löst die im DC eingestellte Fahrstraße aus einem EIT für den Zug auf, welche durch eines der drei *IT SetPathXXX*-DCs gesetzt wurde. Ein Zug kann mehrere Fahrstraßen gelistet haben.

?????????? ??????????

Werden die Fahrstraßen aus der Liste des Zuges automatisch entfernt oder muss man explizit das Löschkommando benutzen?

?????????? ??????????

3.5 ITMCAssignNextPath (<kuid2:61392:8202>)

Dieser Befehl löst eine Suche nach weiteren, im MCM vorhandenen passenden EIT-Fahrstraßen aus, welche dann der MCM versucht nach einem Auswahlalgorithmus zu aktivieren. Die Suche erfolgt in Fahrtrichtung. Die Entfernung, bis zu der Fahrstraßen und Folge-Fahrstraßen gesucht werden, ist begrenzt auf die im MCM eingestellte Länge (Standardwert 10km). Die gleiche Aktivität für den Zug löst ein *EIT PathTrigger* aus. Das Auswahlverfahren wird bei der Beschreibung der [MCM-Regel](#) ausführlich dargestellt.

3.6 AutoDriveToMCPathStop (<kuid2:61392:8321>)

Lässt den Zug zum angegebenen MCPS automatisch fahren. Das funktioniert für alle Züge, so ähnlich wie die *Drive to* und die *Navigate to* Befehle. Allerdings wird für die Zugführung zum Ziel das Auswahlverfahren benutzt, welches bei der Beschreibung der [MCM-Regel](#) ausführlich dargestellt wird.

3.7 SetMCTimeTable (<[kuid](#):61392:8330>)

Ordnet den angegebenen Zeitfahrplan aus dem MCS&TTM als aktuellen Zeitfahrplan dem Zug zu.

3.8 AutoDriveWithMCTimeTable (<kuid2:61392:8322>)

Startet die Abarbeitung des aktuell zugeordneten Zeitfahrplans aus dem MCS&TTM.

3.9 SetMissionCode (<kuid2:61392:8201>)

Entfernt alle dem Zug zugeordneten Mcs und fügt als einzigen den angegebenen MC (eine Liste aus fahrstraßen von Its oder EITs) hinzu. Der MC muss im MCM erstellt sein. Er kann auch dort dem Zug zugeordnet werden.

3.10 AddMissionCode (<kuid2:61392:8203>)

Fügt dem Zug einen MC (eine Liste aus Fahrstraßen von Its oder EITs) hinzu, indem er ihn der Liste der bereits zugeordneten Mcs hinzufügt. Der MC muss im MCM erstellt sein. Er kann auch dort dem Zug zugeordnet werden. Wenn der Zug noch keinen MC hat, verhält sich der Befehl wie das *SetMissionCode*-DC.

3.11 RemoveMissionCode (<kuid2:61392:8204>)

Entfernt den angegebenen MC aus der Liste der dem Zug zugeordneten Mcs. Der MC muss im MCM erstellt sein und mit dem *SetMissionCode*-DC, dem *AddMissionCode*-DC oder im MCM dem Zug zugeordnet worden sein.

4 Trackside und Trackmark Objects

Diese Objekte werden beim Bauen der Routen und Session der Anlage hinzugefügt. In den Eigenschaften können mögliche Einstellungen vorgenommen werden.

4.1 Enhanced TRC3 invisible Interlocking Tower (<kuid2:61392:8101>)

Der EIT ist ein Trackobjekt, mit dessen Hilfe die in den Eigenschaften angelegten Fahrstraßen verwaltet werden können. Er ist eine Erweiterung bzw. Verbesserung gegenüber den Standard ITs.

Anmerkung: TRC3 bezieht sich auf das Trigger Rail Crossing System (Siehe [TRC V3 ! The Trigger Rail Crossing System](#) im [L'atelier de Vincent](#))

4.2 EIT PathTrigger (<kuid2:61392:8121>) (Trackmark)

Dieser Trigger löst eine Suche nach weiteren, im MCM vorhandenen passenden EIT-Fahrstraßen aus, welche dann der MCM versucht nach einem Auswahlalgorithmus zu aktivieren. Die Suche erfolgt in Fahrtrichtung. Die Entfernung ist begrenzt auf die im MCM eingestellte Länge (Standardwert 10km). Die gleiche Aktivität für den Zug löst ein *ITMCAssignNextPath*-DC aus.

Zur Beschreibung des Suchvorganges siehe [3.5. ITMCAssignNextPath](#).

4.3 MCPathStop (<kuid2:61392:8311>) (Trackmark)

Dieses Trackside-Objekt ist ein spezieller Wegpunkt, der zur Kennzeichnung von MC StopZonen genutzt wird und ist mögliche Zielangabe für das *AutoDriveToMCPathStop*-DC.

4.4 Track Circuit Insulator (<kuid2:>) (Trackmark)

Ein *Track Circuit Insulator* begrenzt Gleisabschnitte, für die eine Gleisbesetzt-Information gewünscht wird. Ein in diesem Bereich vorhandener *Track Circuit Detector* erkennt dann, ob der Gleisabschnitt besetzt ist, oder eben nicht.

?????????? ??????????

Wie erhält man diese Information?

?????????? ??????????

4.5 Track Circuit Detector (<kuid2:>) (Trackmark)

Ein in einem durch *Track Circuit Insulator* begrenzten Gleisabschnitt vorhandener *Track Circuit Detector* überachtet, ob dieser Gleisabschnitt besetzt ist oder nicht.

?????????? ??????????

Wie erhält man diese Information?

?????????? ??????????

5 ScriptLibraries(Zubehör)

Diese Bibliotheken müssen mit installiert werden, damit die Assets, welche Scripte oder andere Zusammenhänge daraus benutzen, auch wie vorgesehen funktionieren können. Solche Bibliotheken sind dann innvoll, wenn mehrere Assets oder an mehreren Stellen die gleichen Scripte oder Zusammenhänge benutzt werden. Damit lassen sich also bereits vorhandene Dinge wiederverwenden.

5.1 EnhancedInterlockingTowers library (<kuid2:61392:8190>)

Diese Bibliothek enthält die zum Betreiben der EITs nötigen Funktionalitäten. Sie wird nur intern verwendet und ist für den Benutzer nicht sichtbar.

5.2 TRC 3 InterlockingTower library (<kuid2:61392:8191>)

Diese Bibliothek enthält zum Betreiben der ITs und EITs nötige Funktionalitäten. Sie wird nur intern verwendet und ist für den Benutzer nicht sichtbar.

Anmerkung: TRC 3 bezieht sich auf das Trigger Rail Crossing System (Siehe [TRC V3 ! The Trigger Rail Crossing System](#) im [L'atelier de Vincent](#))

5.3 MissionCodeLibrary (<kuid2:61392:8290>)

Die MC Bibliothek enthält zum Betreiben des MC-Systems nötige Funktionalitäten. Sie wird nur intern verwendet und ist für den Benutzer nicht sichtbar.

P.S. Alle Inhalte wurden von mir nach bestem Wissen und Gewissen zusammengetragen, ohne Anspruch auf Vollständigkeit und Richtigkeit. Die Quellen der übersetzten bzw. interpretierten Texte sind als Links beigefügt.