



ニュースレター

Vol. 34 – No. 5

9~10月 2022

デジタルコンサルタント

Curtis Jeung

Rick Sinclair

最新ソフトウェアバージョン

Central Station 3 Version - 2.4.0(5)

Central Station 2 Version - 4.3.0(5)

Mobile Station 2 Version - 3.148

再び秋が訪れ、ホリデーシーズンに突入しようとしています。これは、手遅れになる前に車両を修理に出すための友好的なリマインダーです。多くの鉄道模型店は、修理の量が多いため、クリスマスに間に合うように修理を終えることができません。部品の取り寄せが必要な場合は、さらに時間が必要な場合があります。なので、早ければ早いほどよいのです。

Rickの記事では、MobileStation2を使用したターンテーブル(74861)の制御について詳しく説明しています。Curtis のトピックは「制御コンタクトについて」です。

前回の記事では、ターンテーブル(74861)をMobileStationで操作することに焦点を当てました。私は興奮したユーザーのような視点で解説してしまいました。つまり、マニュアルをあまり参照していませんでした。MobileStationの説明が記載されている操作説明図をオンラインで確認しましたが、プログラミングのセットアップ手順は見えていませんでした。

セットアップに関してさほど多くのことはありませんが、ユーザーがそれを知り、必要に応じて使用できるように、最小限の説明をしたいと思います。

ターンテーブルマニュアル

ターンテーブルは下位互換性があるため、MobileStationユーザーに門戸を開きます。現在、説明書は特定のコントローラー用に説明されているわけではありませんが、古いデジタル(6021 Control Unit/6040 keyboard)時代のコントローラー用に使われているようです。MobileStationは基本的にその発展形であるため、以下の手法は両方のコントローラーで機能します。

マニュアルの14ページに、新たに1番レールを設定する方法が説明されています。これは、レイアウトを変更したい場合や、扇形機関庫を追加したい場合に役立ちます。理由が何であれ、この手法を使用すると、ターンテーブルに触れる必要はありません。1番レールの再プログラミングは、MobileStationで行えます。

命令は簡単ですが、もう一度、メルクリンのウェブサイトからダウンロードして、マニュアルの15ページにあるMobileStationの命令を参照して下さい(図1)。

CS2/CS3	6021 (/CS2)	72760	MS2 address	Function
	end		225	Rotation is ended at the next track
	input	Button 1	225	Unlocking/Locking the turntable deck (on = unlock)
	clear		226	Continuing to run after STOP
	turn	Button 2	226	Rotating 180°
	step	Button 4	227	Deck rotates clockwise to the next track
	step	Button 3	227	Deck rotates counterclockwise to the next track
			228	Preselection of the rotation direction clockwise
			228	Preselection of the rotation direction counterclockwise
			229 - 236	Destination track to be stopped at
			236	Sound: Announcement of the turntable operator
			237	Operating sounds off
			237	Operating sounds on
			238	Sound signal: „(Locomotive) approaching”
			238	Sound signal: „(Deck) turning away”
			239	Light off (inside)
			239	Light on (inside)
			240	Light off (outside)
			240	Light on (outside)

* In the CS 3, you can stop at the tracks by touching the desired track on the display. No special buttons are provided for this.
Note: The turntable accepts no switching commands as long as the locking feature for the deck is open.

15

図1 MobileStationの命令をハイライト表示。

1番レールを設定する操作は次のようになります。

1. ブリッジを目的の1番レールまで回転させます。ブリッジのどちらの端が目的の1番レールにあるかは関係ありません。

2. 「STOP」ボタンを押してレール電源をオフにします。ここで、図2のようにMobileStationのアドレス225/226までスクロールします。

3. 「STOP」ボタンをもう一度押して、5秒以内に「Clear - Input - Clear」コマンドを入力します。

4. 説明書を見ると、これはキーボードのボタン「226Red - 225Green - 226Red」を押すことに相当します。

5. これにより、現在の位置が1番レールとして設定されます。



私たちの知識の一部は、ユーザーの質問に答えることで得られます。時には、理論をテストして物事を理解することもあります。ある時、スポークレールかブランクセグメントかどうかにかかわらず、各セグメントへの番号付けについて誰かと話しているときに、MobileStationによるターンテーブルの初期化についての議論が持ち上がり、これを調べることにしました。

ターンテーブルを初期化してスポークレールとブランクセグメントを(内部的に)マッピングできれば、MobileStationは「ブランク」セグメントを選択しないだろうという理論でした。現在、MobileStationにはターンテーブルのグラフィック表示がないため、ブランクセグメントをジャンプできないことはわかっていたましたが、初期化した場合を確認したかったのです。

初期化

マニュアルを読むことで、1番レールを初期化できるという同じ理論で、それと同じくらい簡単にターンテーブルを初期化できました。プロセスは次のようになります。

1. ブリッジを1番レールにセットします。
2. 「STOP」ボタンを押します。ここで、図2のようにMobileStationのアドレス225までスクロールします。
3. もう一度「STOP」ボタンを押して、5秒以内にコマンド「End - Input - End」を入力します。これはキーボードのボタン「225Red - 225Green - 225Red」を押すことに相当します。

これが完了すると、ターンテーブルが回転し始め、スポークレールとブランクセグメントがマッピングされます。

シーケンスが完了したので、当然テストしたいと思いました。ブリッジの行き先としてブランクセグメントを選択しました。がっかりしたことに、それはブランクセグメントに行き、停止しました。

前と違いがないように思えたので、ターンテーブルをMobileStationで初期化する目的が何であったかに立ち返らなければなりません。ターンテーブルがブランクセグメントで停止するのを防ぐためのフィードバックがMobileStationにないのだろうかとか、ターンテーブルが「エラー」トーンを発してコマンドに従わないのだろうかと考えましたが、どちらも真実ではありませんでした。

もう一度考えた後、'step' コマンド (227 RedまたはGreen) を試しました。これにより、ブリッジが次のスポークレールに移動しました。ブリッジはブランクセグメントをスキップしました。

今では、初期化プロセスの利点はそれほどないことに気付きましたが、機関車を扇形機関庫に出し入れするときに便利です。

ターンテーブルサウンド

私たちの仕事の最も良い点の1つは、実験できることです。いくつかの実験は役に立ち、一部は役立ちません。私たちは失敗を人々に見せるのが好きです。他の人と失敗を共有するので、彼らが過ちを繰り返さないようにできます。以下は、ターンテーブルに関するそのようなシナリオです。

ターンテーブルのサウンドについて友人と話していました。具体的には、操作係のアナウンスを英語に変更できないかということでした。私は彼に、工場出荷時のデコーダーであるため、デコーダーを再プログラムすることはできないと言いました。

ここで、ダウンロードしてデコーダーにインストールできるデコーダーサウンドプロジェクトがあると思いつきました。ターンテーブルの保証修理を行う際に、標準的な21ピンデコーダーが搭載されていることを知っています。したがって、理論的には、メルクリンのWebサイトのプロファイルを使用してデコーダーをプログラムし、独自のオペレーターサウンドを追加することができます。少なくとも私の頭の中ではそうでした。

これをテストするために、メルクリンサーバーからサウンド/デコーダープロジェクトをダウンロードすることから始めました。オペレーターの音声は1つしかないため、音声ファイルを複数追加することはできませんでした。ターンテーブルプロファイルにファンクションボタンを「追加」する方法がないため、オペレーターサウンドがあった場所に独自のカスタム サウンドを挿入しました。

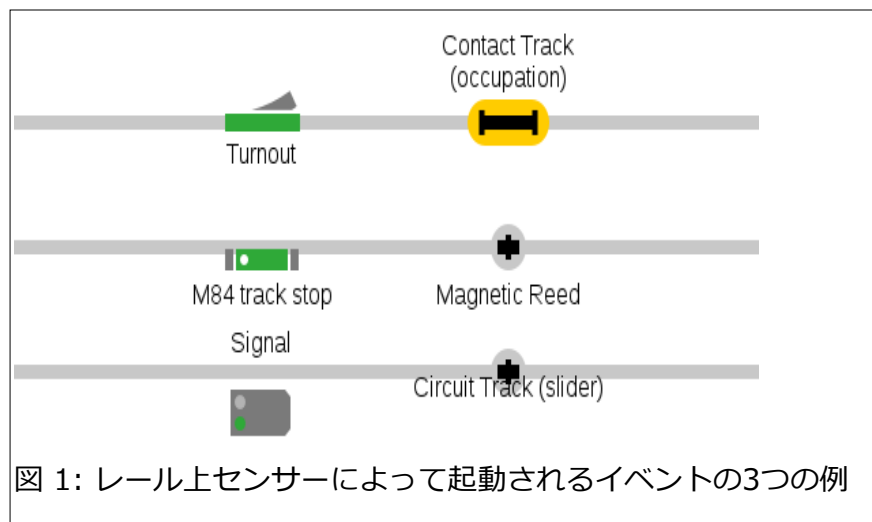
デコーダーをターンテーブルに取り付けてCS3に接続すると、残念ながら機関車として登録されました。デコーダーにアップロードしたカスタムサウンドを再生させることはできましたが、元のデコーダーのように制御することはできませんでした。スポークレールのインデックス作成や停止さえできませんでした。モーターは機関車のように回りました。次にmfxアクセサリとしてアイテムリストに追加しようとしたのですが、ターンテーブルはアクセサリとして読み込まれませんでした。

この失敗を克服できるかどうかを確認するために、いくつかの実験を行う必要があります。あるいは、負けを認めざるを得ないかもしれません。

Enjoy your hobbies!
Rick Sinclair

制御コンタクトとは

忠実な読者なら、イベント(CS3/3+)プログラミングに関する私の記事を読んだことがあるでしょう。イベントを使用すると、ボタンを1回タッチするか、コンタクトレールを使用することで、多くのもの(CSではアイテムと呼ばれる)をアクティブにすることができます。めったに議論されないイベントの1つの機能として、'Controlling Contacts (制御コンタクト)'があります。この記事では、制御コンタクトの機能と、それらをイベントおよびイベントトリガーで使用方法について説明します。初心者向けに、まずイベントトリガーとは何かを説明します。これらは、S88モジュールに接続されるレール上のセンサーです。それらは、(3線式システムの)シュー、磁気リードスイッチ、または在線を検出するコンタクトレールで、イベントを起動するレール上の「トリガー」センサーです。



センサーはS88モジュールに信号を送信し、S88はレイアウト上のどのセンサーがオン(またはオフ)したかをコントローラー(CS3/3+、CS2)に伝えますが、どの機関車がセンサーを踏んだかまでは通知できません。特定のセンサーがオン(またはオフ)したことをコントローラーに通知するだけです。起動されるイベントとその動作の例を図1に示します。

注意! : 例は単にトリガーセンサーが何ができるかを示しているだけであり、機能の詳細を示しているわけではありません。



図1で、最初のセンサーの例はコンタクトレールです。表示されている3つのセンサーのうち、在線検知に使用できるのはこれだけです。

このトリガーセンサーは、図2に示す例で使用できます。列車がこのコンタクトレールに接触すると、(列車の後方にある)ポイントを切り換えて、入ってくる後続列車の進路を変更します。

同様に、図3と図4では、トリガーセンサーにより、m84による停止区間や信号機による停止区間などを制御できます。また、3つの例は固定されたアクセサリを制御することに注意してください。それらをレールまたはレイアウト制御イベントと呼びます。



図3: リードスイッチトリガー

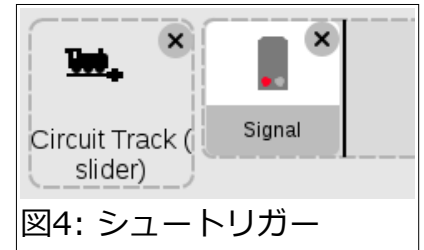


図4: シュートリガー

イベント制御

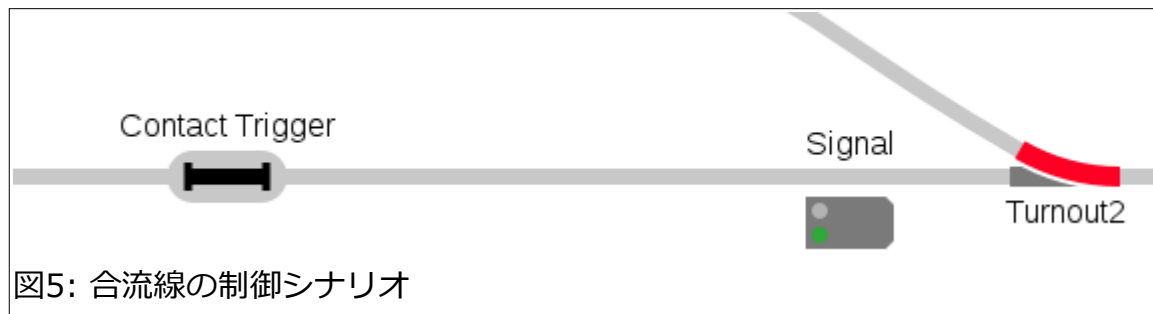


図5: 合流線の制御シナリオ

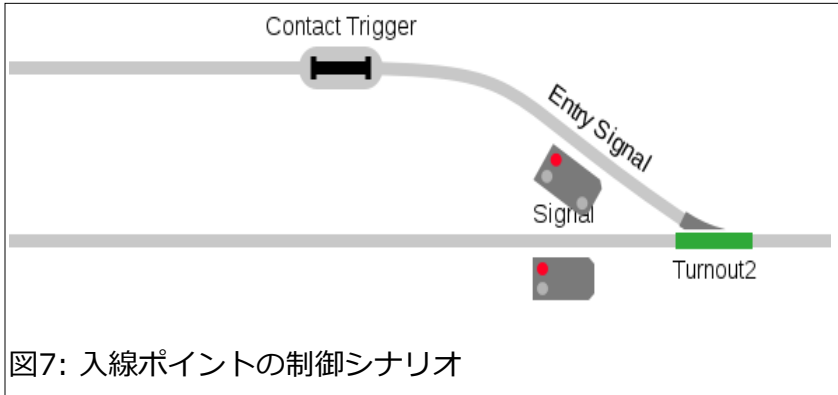


図6: ポイントを使用したイベント条件

イベントには、ポイントの切り換えや信号機の現示の変更などのアクション以外のものを含めることができます。CS3の「拡張モード」機能を設定すると、イベントシーケンス内に「条件」ステップを含めることができます。これらは「制御」ステップとして定義できます。イベントシーケンス内に配置すると、制御因子として機能するためです。制御の内容は、イベントを終了するか、または継続するかです。イベントシーケンス内のそれらが配置されている位置で条件判断します。それらは「?」アイコンによって識別されます。条件ステップの左上隅にあります。次の例で、条件ステップについて説明します。

最初の例(図5)では、'Turnout2' の状態をチェックして、ポイントが進入口に設定されているかどうかを確認します。そうでないなら、列車を止めるよう赤信号にします。イベントは 'Contact Trigger' によって起動されます。イベントプログラムを図6に示します。'Turnout2' という名前の付いた条件ステップによって状態が評価されていることがわかります。条件が「真」(分岐状態)の場合、信号機を「停止」にします。図6のイベントは、列車が 'Contact Trigger' に到達したときに開始されます。

信号機を条件として使用する同様の例を示します (図7、図8)。



上の例では、'contact trigger' で、信号機が停止現示かどうかを評価します。「真」の場合、イベントはポイントを待避線側にします。また、列車が通過できるように、'Entry Signal'を「緩速進行」現示にします。

制御コンタクト

ここまで、2つの異なる要素について説明してきました。トリガー要素 (センサー)と制御要素(条件ステップ)です。それらに加え、2つの形式を持つ別の要素があります。

それらは 'Controlling Contacts(制御コンタクト)' と呼ばれます。ただし、1つの形式はCS3内で明示されていませんが、もう1つは明示されています。これを理解するために、使用方法の類似点を説明し、次に2番目の形式の独自性を紹介します。

1つ目の制御コンタクトは、コンタクトを条件ステップとして使用します。前の例では、信号機とポイントの状態を条件としてイベントを制御していました。しかし、信号機が赤かどうかを評価するよりも、在線検知センサーを評価する方が信頼性が高いでしょう(信号機は、列車が近くにあることを正確に示さない場合があります)。そこで、コンタクトレールを条件として使用します。違いは、図9と図10のイベントの例で確認できます。

この方法を使用すると、コンタクトレールをイベントの実行条件とすることができます。図10で、コンタクトレールが「占有されていない」(灰色)と表示されている場合、続くイベントステップは実行されません。

この例は、デジタルニュースレーターの以前の記事で既におなじみの制御コンタクトのスタイルです。

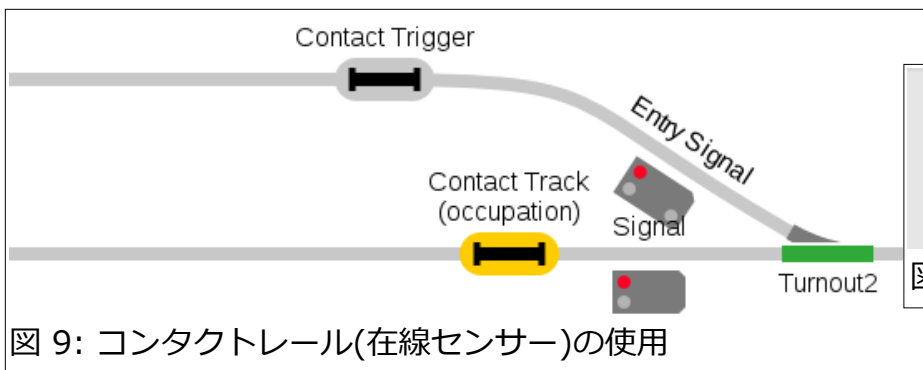


図 9: コンタクトレール(在線センサー)の使用



図10:制御コンタクトの例(図8と比較)

制御コンタクトのもうひとつの形式は、CS3のメニューで明示されます(図11)。これまでに説明した例と使い方は非常に似ていますが、明確な違いがあります。'Controlling contacts (制御コンタクト)'は、ソフトウェアコンタクトです。換言すれば、この形式の制御コンタクトは、物理的センサーとs88モジュールを必要としません。

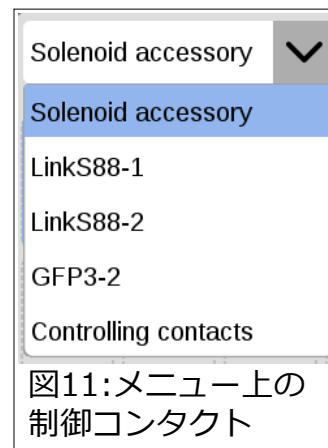


図11:メニュー上の制御コンタクト

問題は、「物理的にレイアウトにないコンタクトをどのように作成するか」です。これを説明すると、物理デバイスを必要としない制御デバイスです。信号機、ポイント、ライト、センサーは必要ありません。それをコンタクトと呼ぶのは少し間違った名前の付け方です。

この形式の制御コンタクトのもう1つの違いは、切り換え可能であることです。つまり、イベントステップでオンまたはオフを切り換えることができます。ポイントや信号機を切り換えるのと同様です。物理的なs88コンタクトもオン・オフを切り換えることができますが、列車による作動を妨げるため、そうしたくないでしょう。制御コンタクトは、本質的にレールセンサーや列車に無関係な制御デバイスです(センサー・モジュール接続を必要としません)。

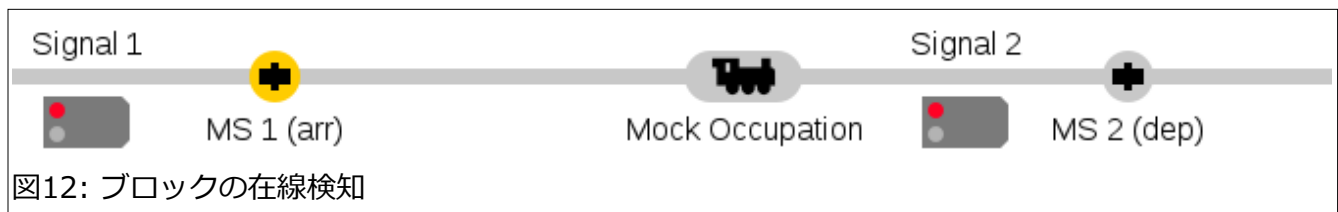
制御コンタクトの使用

正直なところ、いつ制御コンタクトを使用するかは、その必要性を見つけた結果にすぎない場合があります。それらを最初に使用したのは、2022年のデジタルニュースレター-No.2で書いた機関車追跡の記事でした。この使用例は、一般的なユーザーのスキルをはるかに超えています。それ以来、私はより簡単な利用方法を探してきました。ここでそれらを紹介します。

在線検知

制御コンタクトの最初の有用な使用法は、2線式システムのユーザーに適用されると思います。メルクリンの3線式システムでは、読者の多くは在線検知センサーに精通しているはずですが、2線式システム(Trix、LGBなど)の場合、在線検知は実現不可能ではないにしても、理想的ではありませんでした。センサーは、磁気リードスイッチまたは電流検出のみに限定されていました。これらは、オンになるとすぐにオフになる瞬間的なトリガーセンサーです。制御コンタクトの使用は、2線式システムで継続的在線検知を模倣する方法です。

図12 は、2線式システムで在線検知するために必要なレイアウト設定例です。



上の例は、閉塞制御するための最小限のアプローチです。1つの到着点センサー 'MS 1 (arr)'、1つの出発点センサー 'MS 2 (dep)'、および制御コンタクト 'Mock Occupation' が必要です。到着点センサーは、'Mock Occupation' コンタクトをオンにし、ブロックの在線を示します。出発点センサーは 'Mock Occupation' コンタクトをオフにし、ブロックの空きを示します。

センサーを配置する場所は、列車の長さや、在線をどういうタイミングで検知するかによって異なります。

たとえば、上記の状況では、列車の最後尾がまだブロック内にある間に、MS2が在線情報をオフにする場合があります (センサーは通常、先頭の機関車によってオン・オフされるため)。これらのイベントを図13と図14に示します。

そして、図9・図10の例のように、'Mock Occupation' 制御コンタクトを在線条件として使用して、同様のイベントを作成できます。



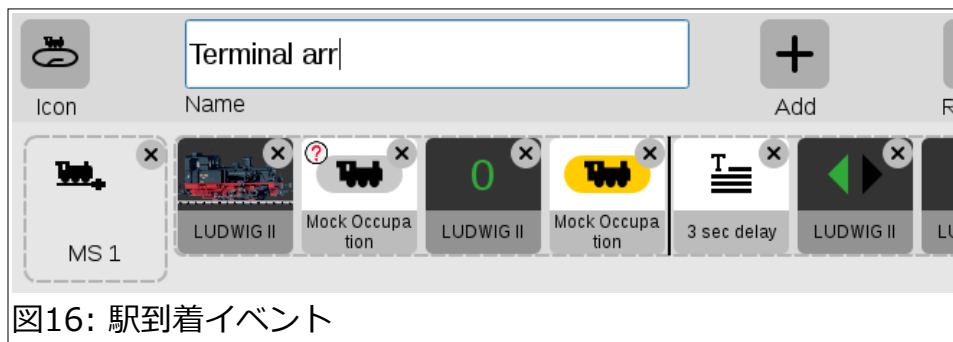
2 線式シャトル運転

CS3のシャトルマクロと、私が書いたシャトルシステムは、シャトルの終点でコンタクトレールを使用することを前提としています。シャトル駅で瞬間的なセンサーを使用すると、問題が発生します。図15は、イベントスクリプトと駅のセンサーのレイアウトを示しています。



つまり、(リードスイッチをオンにする)磁石が列車を停止させるイベントを起動した時点で、磁石は既にセンサーを行き過ぎています。列車が方向転換すると、磁石がイベントを再び起動し、終着駅を離れる前に列車を停止させます。問題は、列車がリードスイッチの上で絶えず前後に方向転換するため、どこにも行けないことです。

考えられる解決策は、制御コンタクトを作成して、駅のセンサーが一方向にしかアクティブ化されないようにすることです。図15のイベントスクリプトは、終点駅に入る列車用です。したがって、駅は空いている前提です。変更されたイベントを図16に示します。制御コンタクトがオフかどうかを確認する条件が含まれています。次に、2ステップ後、イベントは制御コンタクトをオンにします。制御コンタクトの2つの用途に注意することが重要です。



1つは条件('?')アイコンで、もう1つは制御(アクション)です。アイコンをクリックすると、アクションまたは条件を選択できます。このイベントは、制御コンタクトがオフの場合にのみ実行されます。

列車が出発するときに、同じコンタクト(MS 1)によって起動される別のイベントが必要です。このイベントを図17に示します。列車が終点駅を出発するとき、列車が駅に戻ったときにオフになっているように、制御コンタクトをリセットする必要があります。



制御コンタクトの使用は、物理的なS88コンタクトを必要としない(または望まない)場合に便利です。それらは切り換え可能であるため、レール上センサーが適切でない場合に役立ちます。全体として、これらは2線式システムでの在線検知の難しさの良い回避策です。これらの要素の探索を楽しんでいただければ幸いです。

Curtis Jeung

今後の予定 :

- **Trainfest**
November 12-13
Wisconsin State Fair Park, Milwaukee (West Allis) Wisconsin
- **ONLINE Webinar**
October 19, 2022
Delayed 1 week due to factory trip

デジタル、技術、および製品関連について、リックとカーティスに質問するには:

Phone: 650-569-1318 Hours: 6:00am – 9:00pm PST. Monday through Friday.

E-mail: digital@marklin.com

Märklin Digital Club · 1406 Creek Trail Drive · Jefferson City, MO 65109

©2022 Rick Sinclair, Curtis Jeung@Märklin, Inc.
Translated by H.Inoue@The Maerklin Chuff-Chuff Club of Nagoya.