



NEWSLETTER

Vol. 32 – No. 1
January – February 2020

デジタルコンサルタント
Rick Sinclair
Curtis Jeung

Central Station 3 最新Version - 2.0.0 (1)
Central Station 2 最新Version - 4.2.9 (0)
Mobile Station 2 最新Version - 3.112

いっぱいのお食事とたくさんの家族とともに過ごした休日が終わり、寒い冬の日に戻って、模型部屋でレイアウトに取り組んでいます。友だちと一緒に過ごし、大きな目標に向かって小さな成果を積みあげていくのは素晴らしい時間です。

今号の最初の記事では、信号機の見方と配置について知りたい人のために信号機の現示の基本について説明します。2番目の記事は、マクロ(IFマクロ)を使用したCentral Station 3/3+でのイベントの作成に関するものです。

Märklinの信号機を理解する

2000年代初頭、色灯信号機の第1世代が発売された頃、Märklinはいくつかのトピックについてきちんとした「Webトレーニング」を行っていました。これは、(USAで)私たちが時々受けるオンライン安全運転講習とは異なり、トピックについて読んでからテストを受けるだけのオンラインチュートリアルでした。

テストに合格すると、証明書が印刷可能になり、メルクリンオタクであることの証明が誇らしげに表示されました。信じられないかもしれませんが、まだ何枚か持っています。

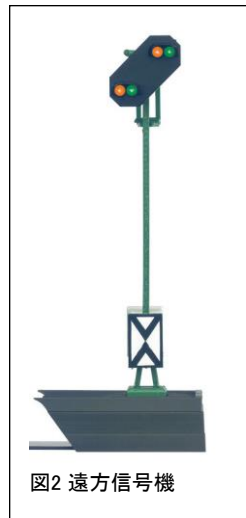
「Webトレーニング」のトピックの1つは、信号機に関するものでした。今日でも、学んだことを思い出すことができ、嬉しいです。間違った信号機を間違った場所に置くのではないかと心配している人々にとって、信号現示が何を意味するかについて書くのが良いと思いました。この記事では、H0デジタル色灯信号機のみを取り上げます。

注:実鉄道システムでは、緑の「進行」信号現示は列車の運転士に「掲示された制限速度で進む」ように指示しますが、本記事では全体を通してこのような詳細規定には言及しません。

信号機の基本

レイアウトでは、基本的に“Home”「主」信号機と“Distant”「遠方」信号機の2つの色灯信号機が使用されます。「主」信号機は、運転士がそれを見たときに「今」何をすべきかを伝えます。下の図1は、「主」信号機の写真です。

「遠方」信号機は、運転士に「次に」すべきであろうことを伝えます。図2は「遠方」信号機の画像です。遠方信号機は、単独で設置されるもの以外に「主」信号機のマストに架装されるもの(図3)もあります。



主信号機の種類

すべてのタイプの「主」信号機は、単に「主」信号機と呼ばれますが、理解を深めるために種類別の名前呼びます。

最も単純な信号機は「閉塞」信号機です(上の図1)。これには、赤:停止と緑:進行の現示があります。

次の信号は「場内」信号機です(図4)。この信号機には赤と緑のライトに加え、黄色のライトもあります。それらは、赤:停止、緑:進行、緑/黄:注意(減速進行)です。この信号機は、本線からの分岐がある駅の手前またはヤードの入り口で使用されます。

最後が、「出発」信号機です(図5)。これは最も複雑な信号機ですが、一度理解すると非常に簡単です。この信号には、赤、緑、黄、白のライトがあります。現示は他の信号機とほぼ同じですが、この信号にはそれらに加え、2つの赤いライトと2つの白いライトがあります。この信号機は、赤/赤:停止、緑:進行、緑/黄:注意(減速進行)、赤/白/白:(ヤード/駅構内での)入れ換え運転の進行許可、となります。入れ換え用の現示には制限があり、これについては後で「入換」信号機のところで説明します。



遠方信号機

前述したように、遠方信号機はスタンドアロンの場合もあれば、主信号機のマストに取り付けられている場合も有ります(図2/3)。私は、運転士が次の主信号機を見ることができないような地形でない限り、マストに遠方信号機が付いた主信号機を使用するのが好きです。

遠方信号機には、緑/緑:進行、緑/黄:減速準備、黄/黄:停止準備の3つの現示があります。

マストに遠方信号機を追加すると、複雑となり、当然のことながら、読み取りが難しくなります。

注目すべきことの1つは、主信号機が赤の場合、マスト架装の遠方信号機がオフになることです。これにより、運転士は、主信号が停止現示ときに緑/緑または緑/黄色の現示を「進行」と見間違えません。



入換信号機

入換(ヤード)信号機は、その名の通りヤード(構内)用です(図6)。これにより、運転士は、ヤードで仕事を進めることが安全かどうかを知ることができます。2つの現示は、赤/赤:停止または白/白:入換進行です。ヤード信号機は、マスト架装または地上の低い台座に配置されます。

「出発」信号機の話に戻ります。ヤード信号現示が、駅/ヤードの出発信号機に組み込まれています。この理由は、入れ換え運転が必要な場合、それを許可する時に不測の事態の恐れがあるためです。したがって、ヤード信号は出発信号機に組み込まれます。

理解して頂きたいルールがあります。入れ換え用の線路がない場合、(入換現示付き)出発信号機の必要はなく、場内信号機で良いです。本線に合流する支線駅で入れ換え用の線路がない場合、(駅の手前に本線列車を止めることのできる場内信号機がないと)本線に出て入れ換えをすることはありえません。したがって、場内信号機を出発信号機として使えると思います。

入れ換えのルール#1は「本線を侵さない」ことを常に忘れないでください。



信号機設置ルール

さて、先ほど説明したとおり、信号機を配置するのは簡単です。

閉塞信号機:本線上の1つのブロック(閉塞区間)から次のブロックへの進入を制御します。

場内信号機:ヤードまたは駅への入場。ポイントの分岐側進入のときにも使用されます。

出発信号機:ヤードまたは駅から本線への出発。

入換信号機:ヤード内での入れ換え運転を許可します。

遠方信号機:次の主信号機で予期される現示を運転士に知らせる。

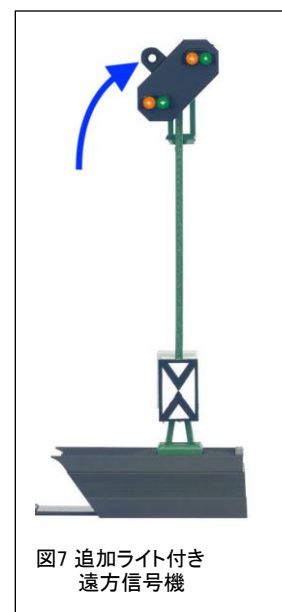
もうひとつの遠方信号機

1つの注意点は、もうひとつ遠方信号機(76481)があることです。これは、スタンドアロン型のみです。この信号機には白色ライトがあります(図7)。このライトは、遠方信号機と主信号機との距離が通常より短いことを意味します。

この記事が信号機をこれから設置しようと考えている方のお役に立てば幸いです。いつでもお気軽にご質問ください。

Enjoy your hobbies!

Rick Sinclair (リック)



CS3/3+ でのマクロを使ったイベント作成:IF Macro (IFマクロ)

CS(2または3)で行うルートプログラムまたはイベントスクリプトでの最強の機能の1つは、条件付きプログラミングができることです。条件付きプログラミングを使用すると、「線路上の」占有状態を確認でき、システムがその状態に基づいて決定を下せるようになります。たとえば、目的地「A」へのルートが別の列車でブロックされている場合、目的地「B」に進みます。プログラマーの専門用語では、これは If > Then文として知られています。

CS2の場合、条件はメモリスクリプト編集ウィンドウの‘Ext.’ ボタンによってアクセスされるリストで設定できました。CS3では、いくつかの方法で条件を作成できます。単純な条件は、イベントステップに直接書き込むことができます。もう一つの方法は、このニュースレター記事のトピックであるイベントマクロを使用して作成できます。

さまざまなマクロがリストされており、基本的に、条件付きプログラムの作成を容易にするために用意されたプレースホルダー(テンプレート)です。今回は「IFマクロ」について説明するとともに、レイアウトを制御する方法について説明します。

制限無しで作業するには、まず高度なイベントの設定をオンにする必要があります。図1は、“Systems > CS3/CS3-1 > Track(運転)”メニューにある設定を示しています。ここで、“Events Extended Mode(イベント拡張モード)”という項目にチェックを入れます。

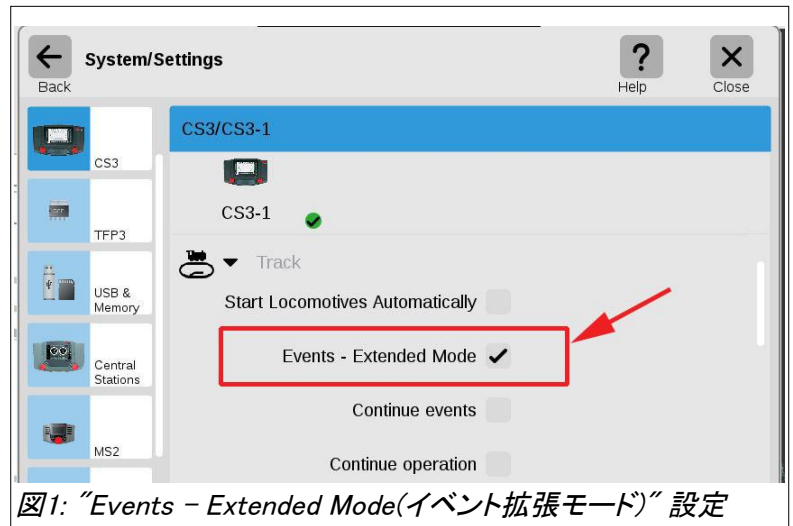


図1: “Events - Extended Mode(イベント拡張モード)” 設定

CS3システムメニュー内にこの項目が表示されない場合は、システムの更新を行う必要がある可能性があります。また、この項目は、以前のシステムバージョンから場所が移動している可能性もあります。

イベントスクリプトで「条件」を使用する場合、最も簡単な方法ではマクロを必要としませんが、さまざまなマクロを理解するための基礎となりますので、説明します。ここで、図2では、ポイント(W 2L)の周囲に2つのコンタクトがあります: コンタクト ‘K1’ はイベント起動用で、‘K2’ は在線検出用です。

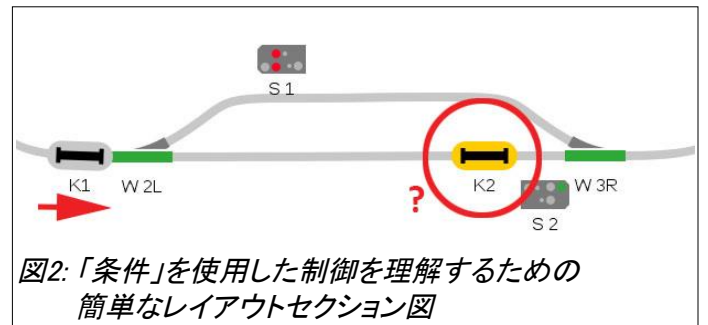


図2: 「条件」を使用した制御を理解するための簡単なレイアウトセクション図

図3は、コンタクト ‘K1’ で起動されるイベントスクリプトを示しています。イベントの最初のステップで、コンタクト ‘K2’ の状態を調べています(アイコン上の疑問符で示されます)。コンタクトのアイコンの状態(=在線)とレイアウトでの状態とが一致する場合、後続のステップが実行されます。この場合、ポイント ‘W 2L’ を分岐に切り換えて、‘K2’ のある占有された線路から列車を迂回させます。条件が一致しない場合、スクリプトは次のアクションを実行せずに終了します。

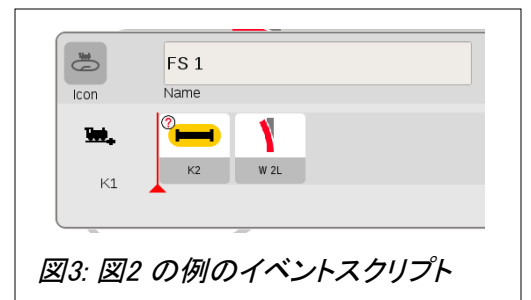


図3: 図2 の例のイベントスクリプト

‘IF’ マクロ

注目したいのは、前の「条件付き」スクリプトには選択肢が1つしかなく、アクションを完了するか、完全にキャンセルするだけです。アクションをキャンセルする代わりに、別のアクションを実行したい場合があります。このタイプの操作では、イベントエディターの編集ライン内で単純にプログラムすることはできません。このようなアクションを実行するために、CS3ソフトウェア内にマクロが用意されています。

マクロは基本的に事前に作成されたルーチンであり、埋めるためのプレースホルダー(空欄)を含みます。今回、最初に紹介するマクロは ‘IF’ マクロ(オプションリストの ‘Macro IF (If-Makro)’)です。

マクロの追加

マクロを追加するには、イベントエディターの ‘Add (追加)’ ボタンをクリックします。選択可能なさまざまなマクロのリストが表示されます(図4)。

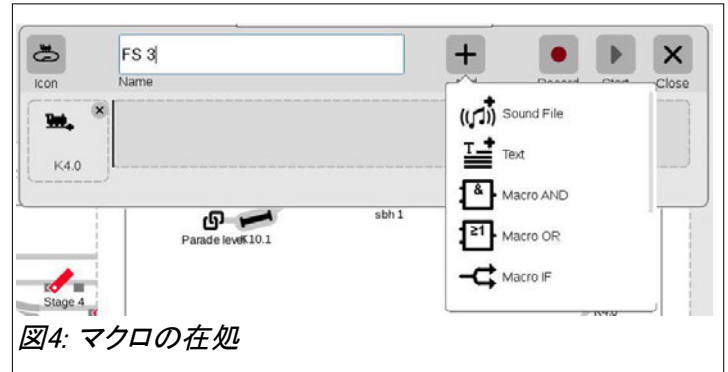


図4: マクロの在処

マクロを選択すると、イベント編集ライン内にマクロステップが挿入されます(図5)。これをクリックすると、マクロ設定パネルが表示されます。ただし、これはマクロの条件やアクションを設定するためのものではありません。それらを設定するには、イベントリストに戻る必要があります。

図6にイベントリストを示します(わかりやすくするためにリストを簡略化しました)。左のイベントは、図5でマクロが追加されたイベントです。これをマクロプレースホルダーと考えてください。右のイベントは、左のイベントに追加されたマクロの中身です。つまり、イベントにマクロを追加すると、イベントリストに追加のイベント(マクロイベント)が作成されます。



図5: イベント編集ラインに追加されたマクロステップ

マクロを設定するには、マクロイベントを見つけてクリックし、イベントエディターで ‘Macro IF’ (IFマクロ)イベントを開く必要があります。図7にイベントエディターが開いた状態を示します。ステップ編集ラインには、‘If container’、‘Then container’、‘Else container’ の3つのコンテナ(空欄)があります。プログラマー用語では、If/Then/Else文として知られるものです。

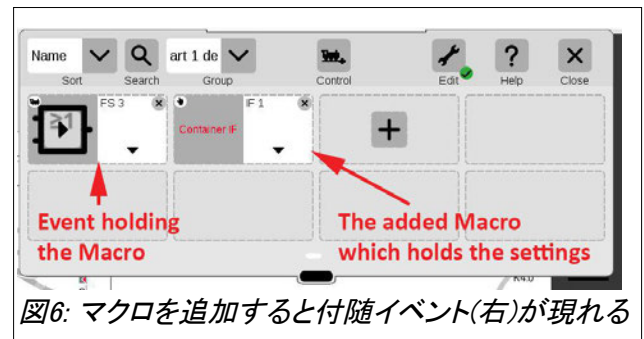


図6: マクロを追加すると付随イベント(右)が現れる

If/Then/(Else) 論理の理解

if/then/else文の説明を容易にするには、実際のアプリケーションでどのように機能するかを示すことが最善です。そこで、説明用に私のレイアウトのセクションを使用します(図8)。

この例では、信号停止区間‘sbh1’または‘sbh2’のいずれかの前で常に列車が待っています。今、待機中の列車は、コンタクト‘K3.0’で在線検知されています。従って、次の列車は、コンタクト‘K2.0’により「空いている」ことを示されているもう一方の線路に入らなければなりません。

プログラミング思考で考えずに、普通の思考プロセスで考えます。‘K3.0’線に列車がある場合、ポイント(sbh t.26)を直進に切り換える必要があります。‘K2.0’線が占有されている場合は、ポイントを分岐側に切り換える必要があります。これを考察すれば、どちらのコンタクトが占有状態かを見て、ポイントを切り換えればいいことになります。ここで、別の見方をすると、‘K3.0’線の在線／非在線に応じて、ポイントを直進または分岐に切り換えることもできます。即ち、

[IF] ‘K3.0’線が占有されている場合は、

[THEN] ‘sbh t.26’を直進に切り換え、

[ELSE] それ以外の場合は‘sbh t.26’を分岐させます。

これで‘if, then, else’文と呼ばれる意味がわかったと思います。この例(図9)では、1つのコンタクト‘K3.0’のみを見て、ポイントの切り換え方向を決定しています。すなわち、‘K3.0’線が空いている場合は、そちらに分岐するようにポイントを切り換えます。

どちらの制御例も同じことを達成し、レイアウトを制御する方針に応じて、それぞれに意味があります。違いは、後者は1つのイベントスクリプト、コンタクト、およびマクロのみが必要なことです。前者は、2つのイベントスクリプトと2つのコンタクトが必要です。ここでは、マクロを使った単一コンタクトによる制御に焦点を当てています。

イベントスクリプトエディター(図9)に戻り、2つのイベント部品を前の段落で説明したステップコンテナに配置しました。これらの2つの部品は、‘Track condition’ (在線状態)と‘Controlled element’ (制御対象)です。‘K3.0’コンタクト条件が‘Container If’に配置されていることがわかります。制御対象のポイント‘sbh t.26’は、‘Container THEN’と‘Container ELSE’の両方に配置され、切り換え方向が適切に設定されています。

「IFマクロ」の機能を再度説明します。在線状態を見て、条件が一致するかどうかに基づいてアクションを実行します。条件が一致しない場合、代替アクションを実行します。マクロは‘K3.0’の状態を読み取ります。それが占有状態の場合(コンタクトのアイコンが黄色の場合)、次にポイントを直進に設定します。それ以外の場合は、分岐に設定します。



図7: IFマクロの編集ライン

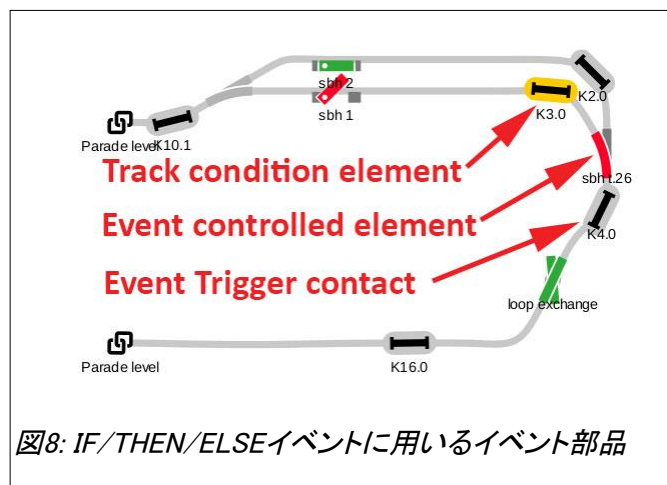


図8: IF/THEN/ELSEイベントに用いるイベント部品

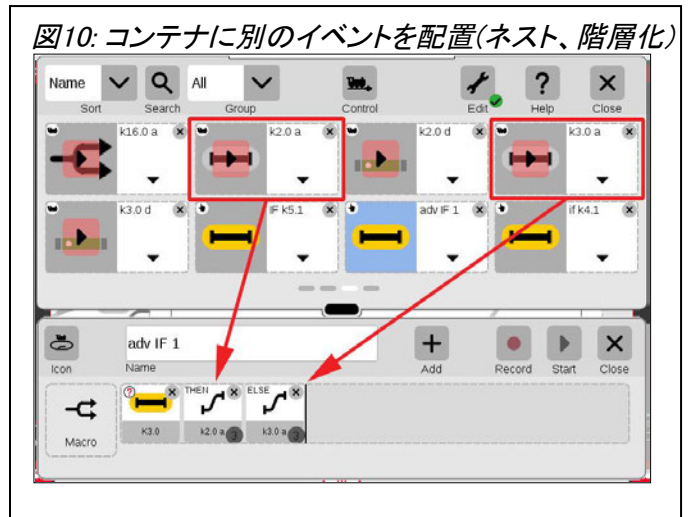


図9: IF, THEN, ELSE コンテナにあてはめたイベント部品

‘IF’・‘THEN’ コンテナについての補足

THENコンテナとELSEコンテナはそれぞれ1つのイベント部品のみを保持でき、前述の例では、1つの部品のみで済みましたが、複数の制御項目を必要とする場合は、THENおよびELSEコンテナに別のイベントを配置できます。例えば、信号機とポイントを同時に切り換えたい場合などです。

図10では、2つの別イベントをコンテナに配置しました。これは「ネスト(階層化)」と呼ばれます。IFマクロは、THENまたはELSEコンテナに配置された別イベントも実行できます。



イベントをネストする場合、そのイベントに独自の「条件」が含まれる場合は、コンテナに配置する際に注意してください。それは例えば、システムをロックする恐れのある競合を引き起こす「条件」を追加するような可能性です。それを行うことはできますが、それを行う理由を認識して、予期しない結果に対処できるようにする必要があります。

条件付きイベントの仕組みを理解することは、高度な自動化イベント作成の基本です。最初の例でのIf/Then制御は、(IFマクロを使った)If/Then/Else制御とは、制御したい部品の数異なります。必要な制御結果は1つだけですか、それとも別の制御結果も必要ですか？これらの2つの選択肢を試してみれば、レイアウトを制御する方針と、制御がどのように判断されるかがわかります。

CS3には他にも多くのマクロが含まれていますが、紹介するにはもう少し調査が必要であり、その間、IFマクロだけで達成できる高度な制御が数多くあると思います。

As always, have fun!
Curtis Jeung (カーティス)

今後の出展予定 :

Rocky Mountain Train Show
March 7-8, 2020
Denver Mart in Denver, Colorado

デジタル、技術、および製品関連の質問についてRickとCurtisに問い合わせるには :

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

E-mail: digital@marklin.com

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