



デジタル鉄道模型で始める

# 自動運転入門

~ビジュアルプログラミングで簡単入門~

rev.2 (Sep 21, 2019)



# もくじ

- 1. デジタル鉄道模型 自動運転入門と応用
  - 1.0. 鉄道模型とデジタル
    - 1.0.1 デコーダとは
    - 1.0.2 コマンドステーションとは
    - 1.0.3 CVとは
    - 1.0.4 DCCに対応するには
  - 1.1. デジタル鉄道模型の面白さ
  - 1.2. 自動運転に必要な機器
  - 1.3. ここで使用する線路例と目的の動き
  - 1.4. 自動運転の動きを明確化する
- 2. 在線検出とセンサ
  - 2.1. センサの選び方
  - 2.2. センサの配置の仕方
  - 2.3. センサレールの準備
  - 2.4. センサの使い方
- 3. コマンドステーションと自動運転ソフトウェア
  - 3.1. コマンドステーションの準備
  - 3.2. Wi-Fiの接続
  - 3.3. ソフトの動かし方
  - 3.4. EducationPlatformの画面
  - 3.5. 全体配線図
- 4. Education Platformの使い方
  - 4.1 ブロックの操作の仕方
    - 4.1.1 ブロックをもってくる
    - 4.1.2 ブロックをくっつける
    - 4.1.3 ブロックを分ける
    - 4.1.4 ゴミ箱
    - 4.1.5 右クリックメニュー
  - 4.2 保存の仕方、読み出しの仕方
  - 4.3 待ち時間を考えよう
  - 4.4 線路にDCC電源を流す
  - 4.5 車両を動かそう
  - 4.6 車両の向きを変えて前後に動かそう
  - 4.7 ファンクションを使ってみよう
  - 4.8 ポイントを動かしてみよう
  - 4.9 センサと連動してみよう
  - 4.10 くりかえしブロックの注意
  - 4.11 せんろとセンサの動作チェック
  - 4.12 MP3を再生しよう
    - MP3ファイルの置き方
    - 使い方
    - 応用法
- 5. 自動運転プログラミングの準備

- 5.1. プログラミング機能の紹介
- 5.2. 自動運転プログラムを書いてみる
  - 5.2.1. センサを監視するブロック
  - 5.2.2. 準備する関数ブロック
  - 5.2.2. やま駅(センサアドレスは1)で動くブロック関数
  - 5.2.3. かわ駅(センサアドレスは2)で動くブロック関数
  - 5.2.3. 試運転
- 5.3. 応用編・ファンクションを組み合わせる
  - 5.3.1. ヘッドライト・テールライトを付ける
  - 5.3.2. 室内灯を付ける
  - 5.3.3. ファンクションF2を操作してサウンドを鳴らす
- 5.4. まとめ
  - 5.4.1. 全体の様子
  - 5.4.2. 写真とビデオ
  - 5.4.3. 機器の配線
  - 5.4.4. サンプルスクリプト
- 5.5 変数と関数を使う
  - 5.5.1 変数の使い方
  - 5.5.2 関数の使い方
- 6. ソフトウェアのアップデート方法
  - 6.1 アップデート手順の全体
  - 6.2 本体ファームウェアの設定・更新
  - 6.3 Webアプリのアップデート方法
    - 6.3.1 ツールを使う方法
    - 6.3.2 手動でアップデートされる場合
- 7. FAQ・質問回答
  - 7.1 作成できるプログラムのサイズに制限はありますか？
  - 7.2 DSbasicのプログラムやWindows用のDesktopStationSoftwareのイベントスクリプトをEducationPlatformで使用できるようにプログラムを変換できませんか？
  - 7.3 Education Platformを使って商業・ビジネス利用できますか？
  - 7.4 スマホやタブレットでも使用できますか？
  - 7.5 Education Platformを使用しているときは、他のスマホ等から操作できますか？
  - 7.6 命令が安定して鉄道模型に伝わらない
- 8. サポート
- 9. 最後に

# 1. デジタル鉄道模型 自動運転入門と応用

---

## 1.0. 鉄道模型とデジタル

日本では、線路に供給する電圧を0~12V(直流)で変えて、鉄道模型車両の中に入っているモーターの速度を変えて走らせるアナログDC式が主流です。しかし、欧米では、デジタル鉄道模型が既に主流になっています。デジタル鉄道模型とは、制御信号と電力を同時に送るため交流のパルス状の電圧を線路に流し、鉄道模型車両の中（もしくはポイントの中）にデコーダーと呼ばれる基板をモーターと線路（車輪,ソレノイド）の間に組み込み、コントローラーから信号と電力と一緒に受け取って、車両を動かす制御方式です。鉄道模型車両の電動車（モーター車）に、コンピュータが1台に1つずつ入っていると考えると差支えありません。コントローラーと車両間で通信ができるようになるため、たとえば線路上に複数台の車両がいても、個別に運転制御ができますし、常に電力を供給するので、速度が遅いときはヘッドライトが光らないということもありません。ポイントなどのアクセサリにも同様にデコーダーを組み込むことで、ポイントを操作する機器などが不要になる（省配線化）もできます。世界では、大きく分けてDCC(Digital Demand Control)方式と、メルクリン方式(Marklin Motorola, MM2, Marklin digital等)の2方式があります。メルクリン方式（メルクリン社の鉄道模型）は旧式のMM2と、現在のmfx方式の2種類があります。近年発売されているもののほとんどがmfx方式のものですが、mfx方式のデコーダ搭載の機関車は例外なく、MM2下位互換機能を持っています。つまり、MM2のコントローラで制御可能です。

デジタル化で同一線路上の複数の車両、ポイント、その他の機器毎に様々な制御が可能になります！アナログは、残念ながらただ走るだけ、光るだけで、サウンドやギミックは全くできません。配線もシンプルにすることは難しいです。

配線は簡単。コマンドステーションから出ているフィード線に、並列に配線をしていくだけです。デコーダはアドレスで区別します。アドレスには、車両アドレス(Loco)とポイントアドレス(Accessory)の2種類があり、全く別のもので扱われます。単に“アドレス”とだけ言うと、通常は車両アドレスのことを示します。

DCCを始めるにあたり、どのゲージ・スケールを選ぶかが重要です。諸外国ではHOゲージが主流なため、DCCのデコーダや搭載車両もHOの方が圧倒的に多いです。また、諸外国では世界シェアトップのメルクリン社と、それ以外で半分ほどのシェアに分かれています。メルクリン社のレールは三線式(通称AC)と特殊で、日本で一般的な車輪のみで集電する方式と互換性がありません。それ以外のメーカーは、車輪のみで集電するので車両の互換性があります。三線式はシェアもあるので、サードパーティから対応台車や車両が出ており、海外製の車両だけで楽しむ方にはあまり不都合はありません。また、集電不良に強い、レールの作りが頑丈という特徴もあり、子供でも楽しめるようになっています。しかしながら、国内の車両でも遊びたい方は、メルクリン社のレールシステムは採用しないほうが良いでしょう。日本のメーカーで入手性を考えると、HOではUNITRACKしかありません。NゲージであればトミックスFineTrackとKATOのUNITRACKのいずれかを選べば良いです。ゲージが一緒であれば、レールはどの会社のシステムを使っても車両は共通で動かせます。

### 1.0.1 デコーダとは

デコーダは、コマンドステーションからの命令を受け、モーターを動かしたり照明を光らせたり、ポイントを動かしたりと、スケールや機能ごとに製品が提供されています。日本で入手性の良いDCCデコーダを以下に挙げさせていただきます。2016年現在、DCCデコーダの電子工作では日本が最先端になりつつあります。電子工作ができる人は世界でも一番安く、自分でプログラミングできるデコーダを入手できます。

### 1.0.2 コマンドステーションとは

日本で入手性の良いDCCコマンドステーションを以下に挙げさせていただきます。基本的には欧米のメーカーのものが主流であり、唯一のKATOのコマンドステーションも、アメリカのデジトラックス社のOEM製品となっております。マニュアルは日本語化されているケースもありますので、使い方についてはそれほど戸惑う事は少ないと思われます。

コマンドステーションでは、車両の運転やポイントの制御を1台ですべて行えます。余計な配線は必要なく、2本の配線だけで電源と信号を送る事ができるので、シンプルになります。コマンドステーションの形状は様々ですが、多くの場合はスロットルと呼ばれる操作部分については、ケーブルで延長して動きやすくした製品を出しています。画面表示は、液晶のメーカーもあれば7セグで限定した表示としている場合もあります。パソコンとの接続については非対応のケースが多く、接続できる場合でもオプションのキットが必要なケースがほとんどです。

### 1.0.3 CVとは

DCCデコーダには、モードや設定があり、それを切り替えることで動きを直したり、調整したりすることができます。CVは、1-1024までありますが、DCCで必須と呼ばれるCV値以外は、各社まちまちで、数字の意味が異なるケースが多いです。設定の際は、必ず、デコーダのマニュアルを参照して設定しなければなりません。

CVを車両から読み出す場合、わざと微動させて電流を流す事で内部データを読み出します。この微動は異常ではありません。また、電流をあまり消費しないデコーダの場合、コマンドステーションの仕様によっては、うまく読み出せませんので注意が必要です。

CVはデコーダにパラメータを設定するために使用するための仕組みで、車両の特性に合わせて微調整することが出来ます。また、様々な機能の設定値を格納しておくことで、いろいろな機能を実現できます。CVは多機能にするための重要な機能になっている反面、デコーダを開発するメーカーの実装方法に制約があまりないため、メーカーごとにCV値がバラバラになっているケースが多々あります。CVを設定するには、デコーダの説明書をよく読んで、適切に値を設定することが求められます。CVの書き込みは、線路の上書き込みするための車両だけを載せて、行います。OPSモード(表 11.7参照)で行う場合は、他の車両が載っている状態でも書き込みは可能です。書き込みの際には、いったん、線路に信号を出すのを中断し、CV書き込みのための専用コマンドを与えてCV値書き込みを行います。書き込みに成功すると、一瞬だけ電流を流す操作(CV Acknowledge)を行います。この時、車両が一瞬動くような動きをします。車両への組み込み方によっては、電流を流す操作が出来ない場合があります。コマンドステーションによってはCV書き込み失敗と表示する場合があります。CVの読み込みは、流れる電流が大きい小さいか(CV Acknowledgeを利用)を複数回以上チェックして、値に読み替えることで検出します。Pagedモードでは、これを最大で256回繰り返すため読込は非常に時間がかかります。Directモードでは、ビット単位で読み込むため、8回から16回程度のため短時間で値の確認が可能です。CVの読み込みの際には、車両が連続して8回程度、ビクビクと微動します。これは故障ではなく、CV値を読み出すための動作になっています。DCCでは、CV Acknowledgeの際には60mA以上を流すことを規定しています・しかしながら、デコーダにモータを接続していない場合、60mAを流せるようになっていない場合、コマンドステーションが読込に失敗する場合があります。OPSモードで読込ができない理由としては、他の機関車が走行していたり動いている環境を想定しているため、電流値の変化の読み出しが困難だからです。このような場合に向けて、Railcomなどの特殊なCV値読み出し技術がありますが、対応していないデコーダも多々あることや、線路環境によってはうまく読み出せないケースやデコーダが誤作動することもあります。

### 1.0.4 DCCに対応するには

欧米の車両は、DCCにワンタッチで対応することが出来たり、最初からDCCの機能を搭載したものが売られています。

## 1.1. デジタル鉄道模型の面白さ

アナログでも、いくつもの単線の自動往復運転システムや関連した機器が販売されています。

単線の自動往復運転なんて、簡単で、自動運転の醍醐味を損なっているのでは？と思われると思います。でも、この単線だけで、どんな自動運転ができるのか考えてみると、非常に奥が深いことができます。また、単線でできないことは、他の配線では絶対にできません。つまり、基礎の基礎と考えることができるわけです。

今、日本国内で売られているアナログの単線の往復運転は、残念ながら多くのことはできません。アナログ車両は音は出さない、ライトはヘッドライト・テールライト、それに連動した室内灯程度です。本物の車両とは残念ながら似つかない世界で、そこを自分の頭の中で想像しながら補完する必要があります。でもDCCなら、いろいろなことができるようになります。しかも、車両ごとに、できることが違ってくるとい、いろいろな楽しみ方ができます。また、単線なら、どんなに狭い家でも、確保できます。HOでも、Gゲージでも、うまく工夫すればどうにかなるはずです。

つまり、アナログ鉄道模型では走るだけなので、どうしても飽きが来てしましますが・・・DCCなら話が大きく変わるので。

先程申しましたとおり、線路に走らせる車両によって、自動運転の動かし方は、バラエティに富んで変わってきます。また、線路の途中に人形を置いて線路内人達入りで緊急停車、といったストーリーを作っても良いかもしれません。緊急停車時は、DCCサウンドで急ブレーキや車内アナウンス1)など、様々な遊び方ができます。

このチャレンジでは、基本中の基本から丁寧にやらないといけないこと、考えるべきポイントを紹介し、最終的には、車両の中に入ったサウンドや照明などのギミックを組み合わせ、魅力的なストーリーをこの単線の自動往復運転の中に入れていきます。

## 1.2. 自動運転に必要な機器

DCCの自動運転とし、日本国内メーカーのものをで選んでいきます。最終的にオープンサウンドデータを書き込んだLokSoundデコーダ搭載のDCCサウンド車両を使用することとします。ここではキハ40 オープンサウンドデータ搭載車両を選んでいきます。

- コマンドステーション DSair2
- Fujigaya2 S88 Detector 8ch または Nucky S88 Train Detector (在線検出器、センサ)
- Windowsパソコン(Windows 7以降, 32bit or 64bit)
- USBケーブル(ミニ)
- ACアダプタ((12V~16V, DSair2用))
- フィーダ線
- UNITRACK HO((Nでも可))、他のメーカーの線路でももちろん可
- DCCサウンド車両 HO((サウンドが不要の場合は非サウンド車両でも可))

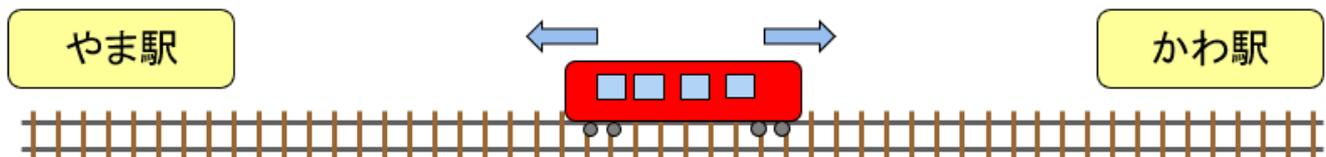
コマンドステーションDSair2、S88 Detector 8chは、[DesktopStationShop](#)で購入できます。

実際に本ページでは、以下の機器を使用しました。

機器・部品名	製品・仕様	価格・備考
コマンドステーション	DSair2	KIT 15800円 +FlashAir 6000円
在線検出器	Fujigaya2 S88 Detector 8ch	KIT 3800円
日本型信号機	Nucky 日本型信号機デコーダ	PCB 1000円+部品代
ACアダプタ	秋月電子 12V/2A	1000円
レール	HO UNITRACK HO	-
車両	トラムウェイ キハ40 500番台 HO	22800円
デコーダ	LokSoundV4 microデコーダ	16000円
USBケーブル	miniUSBケーブル(100円ショップなど)	108円
Ethernetケーブル	100Mbps対応のケーブル(S88-N用)	108円~
フィーダ線	KATO 24-825 延長コード等	207円

### 1.3. ここで使用する線路例と目的の動き

まず、駅が2つあって、その間を単線があるとします。やま駅と、かわ駅の間をディーゼル車が往復する例です。分岐もなく、ひたすら往復するだけのシンプルな配線です。この駅の間を、行ったり戻ったりします。



### 1.4. 自動運転の動きを明確化する

手動で運転するときには、人が目で見えて、駅に近づいたらブレーキを掛けて止める。到着したら反対方向に進行方向を切り替えて発車する、という動作になります。しかし、自動で行うためには、今説明した動作を、プログラミングしなければなりません。

今説明したことを、以下のように箇条書きにすると、プログラミングしないと見えないポイントが見えてきます。

#### 駅に近づいたら (条件1)

- ブレーキを掛けて (条件1の動作)
- 止める。(条件1の動作)

#### 到着したら (条件2)

- 反対方向に進行方向を切り替えて (条件2の動作)
- 発車する (条件2の動作)

以上を、運転パターンとして決定し、次はセンサを配置していきます。最初は、まずはただ動くだけの動作をプログラミングしていきますが、徐々に動きを増やしていき、自動運転の動きを充実化していきます。

## 2. 在線検出とセンサ

### 2.1. センサの選び方

機械の目になるのがセンサです。人には目があるので、目で見て判断できますが、目に相当するものを違うもので置換えてあげる必要があります。

鉄道模型のDCC自動運転が普及しているヨーロッパでは、たくさんのセンサを使用するため、複数のセンサを束ねて、1本の配線でセンサの信号をコマンドステーションに送る仕組みであるS88-Nが一般的に使用されます。

S88-Nに対応したセンサ機器のことを「S88デコーダ」と呼びます。デコーダという名前が付いていますが、DCCデコーダではなく、あくまでもセンサのデータを送るための機器という意味合いになります。機種によって、対応できるセンサの方式が様々で、自分の使い方にあったものを選択します。

DesktopStationでも、S88-Nを標準のセンサ方式として採用しています。DesktopStationでは、以下の日本製のS88デコーダ(在線検出センサ)を推奨しています。

- [Nucky S88 Train Detector](#) - 動力車の消費電力に反応するセンサ(電流式)
- [Fujigaya2 S88 Detector 8ch](#) - 車両の底に赤外線光を当ててその反射で車両に反応するセンサ(光式)
- [Fujigaya2 S88 Button Decoder](#) - ボタン操作でON/OFFを切り替えられるS88デコーダ

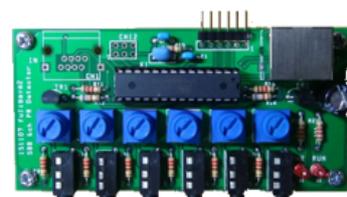
### 国産のS88デコーダ



- [Nucky](#)とFujigaya2からS88デコーダが販売されています。
- 本場のヨーロッパのメーカーや通販からも購入可能です。

#### Fujigaya2

光式と磁気式の2種類に対応したS88デコーダである「S88 Detector 8ch」を販売されています。気軽に在線検出するのに使いやすいです。



<https://seesaawiki.jp/fujigaya2/d/S88%20Detector%208ch>

#### Nucky

動力車の電流消費を検出できる電流式に対応したS88デコーダである「S88 Train Detector」を販売されています。信号や閉塞制御に最適。



[http://web.nucky.jp/dcc/s88n\\_train\\_detector/s88n\\_train\\_detector.html](http://web.nucky.jp/dcc/s88n_train_detector/s88n_train_detector.html)

どちらでも使用できますが、配線の引き回しについては使い勝手は光式の方が良いですが、車両によっては反応しないケースもあり、白い紙やセロテープ、銀紙を貼るといった細工が必要になる場合があります。電

流式は、動力車のみですが確実に検出できる一方で、線路の一部にGapを設ける（KATO等から販売されています）ことや、電流を検出するための配線をセンサまで繋げなければならない、レイアウト向きです。

- 電流式センサはレイアウト向き
- 光式センサはお座敷運転、サクッと動かしたい向き
- ボタン式は動作チェックなど

もちろん、レイアウトに光式センサを使うのもOKです。2つを組み合わせる高等なテクニックもあります。両方持っておけば、心強いでしょう。

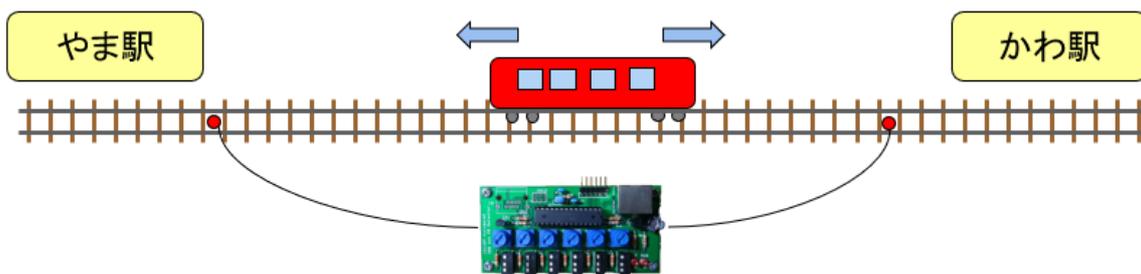
## 2.2. センサの配置の仕方

線路にセンサを配置する手法はたくさんあるのですが、ここではオーソドックスに、駅に停車させるという目的で2つのセンサを配置しました。

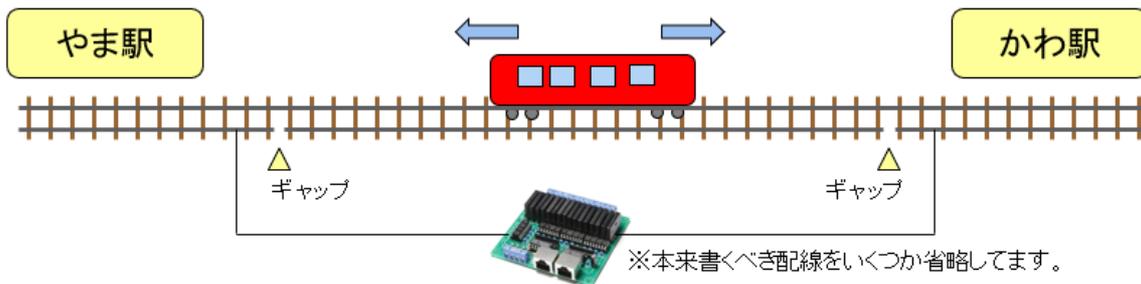
### どこにセンサを置くか？



#### 光式センサを配置する場合：



#### 電流式センサを配置する場合：

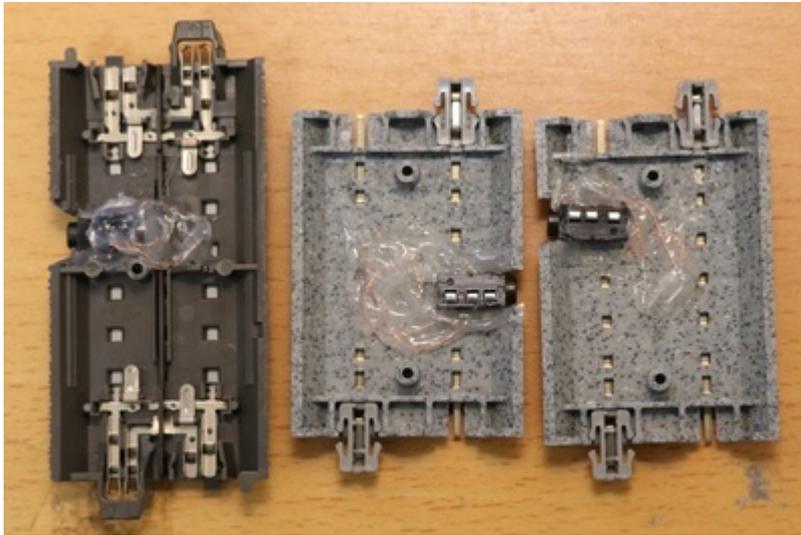


センサを駅の手前に置いたのは、このセンサが反応したらブレーキを掛けて止めるという動作を行うためのトリガ（きっかけ）にするためです。**ブレーキを掛ける時間や停止距離は、DCC車両のモータや特性によって異なります。**特にサウンド車は、サウンドと速度が連動するため、停止距離が長い傾向があります（本物みたいですね）。なので、特にこういった自動運転では、それほど速く走らせることはお勧めできません。ゆったりと走らせると良いでしょう。

車両によって停止距離が異なるということから、駅の正確な位置に止めたいと言う場合には、駅の手前、駅の中、限界ラインの部分などと、センサの数をどんどん増やさなければなりません。それは2つのセンサを使いこなした後で良いでしょう。センサを増やすと、できることも増えますが、考えなければいけない運転パターンが細かくなって複雑になります。

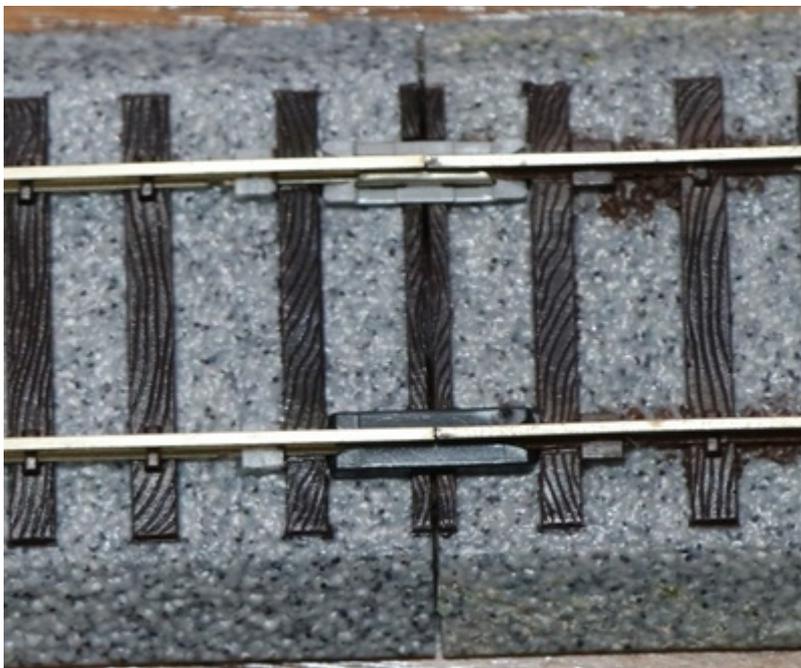
## 2.3. センサレールの準備

ここでは、お座敷でサクッと動かすことで始めますので、S88 Detector 8ch(光式)を使用することにします。線路にセンサを装着する必要があります。詳細は[電機屋の毎日](#)のブログを見ていただければと思いますが、少々半田付けとレールへの穴開け加工が必要となります。



電流式の場合には、UNITRACKですと、絶縁タイプのジョイナー(24-816 絶縁ジョイナー)を使うと、ギャップを切ることができます。また線を引き出す場合は、純正品のターミナルユニジョイナーを使うと、簡単にギャップを切った片側の線路から引き出せて、Nucky S88 TrainDetectorに接続できます。

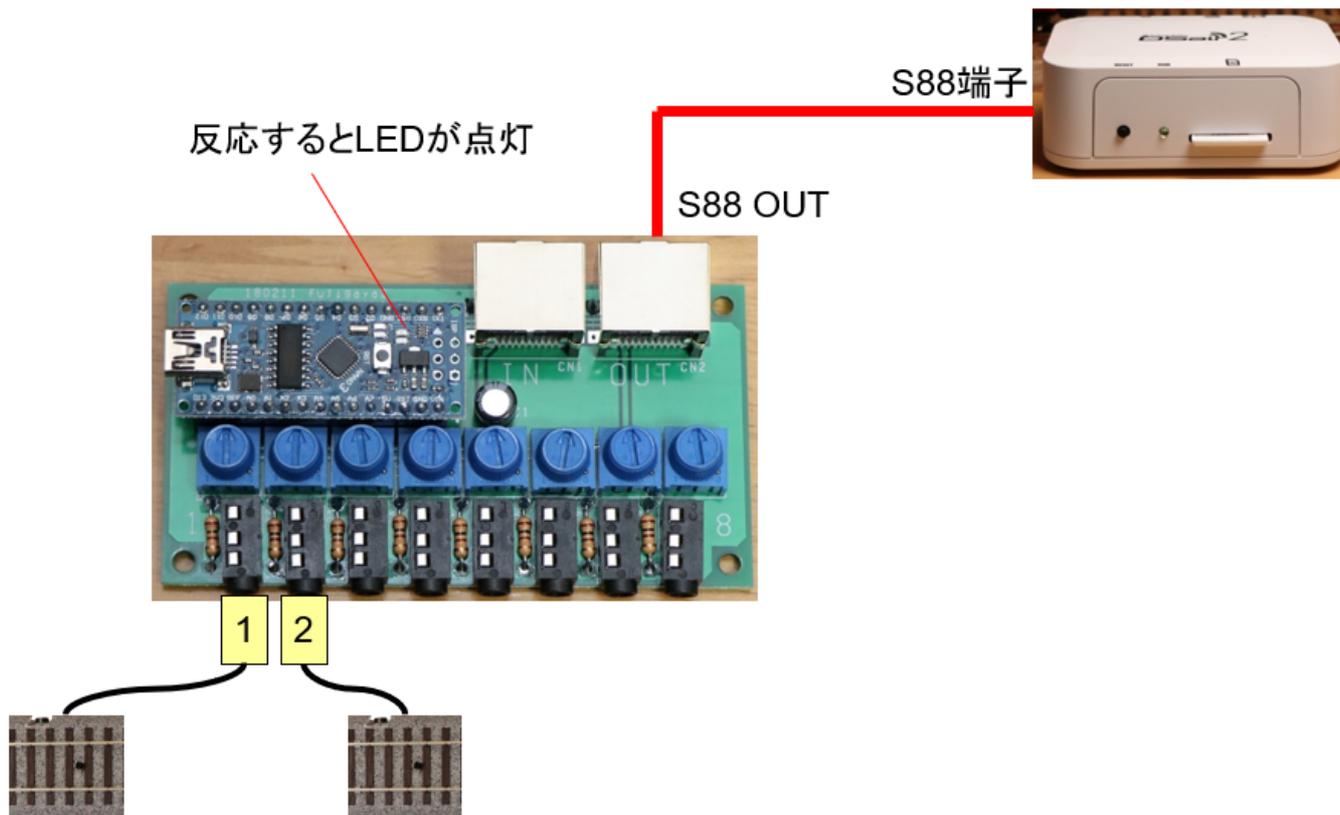
詳細は、[s88-N Train Detector用のギャップの切り方](#)を参照下さい。ここではUNITRACK向けに説明しています。他社線路の場合は、読み替えて参照下さい。



## 2.4. センサの使い方

S88 Detector 8chの使い方を説明します。配線は以下のように行って下さい。線路の上に、車両を通すか、指でセンサ部を触れない程度でふさぐと、反応します。

S88 Detector 8chにフォトリフレクタを使用する場合、フォトリフレクタは赤外線光を照射し、その反射の有無を検出するため、車両の内側が赤外線を反射しにくい素材の場合には、反応しないケースがあります。この場合は、セロテープなどを貼ってあげることで反応するようになる場合があります。



## 3. コマンドステーションと自動運転ソフトウェア

---

### 3.1. コマンドステーションの準備

DSair2を用意しましょう。DSair2は、ファームウェアr2m17以降(2019年7月リリース版)でブロック図プログラミングに対応しています。



DSair2を使用するには、以下の物品が必要です。

- DSair2本体
- FlashAir W-04 (DSair2本体に差し込みます)
- UNITRACK用フィーダ線(線路が異なる場合はその線路に対応するもの)
- ACアダプタ(たとえば12V/2A)



DSair2は以下のようなインターフェースを持っています。後面には、DCジャック、フィーダ出力、USBのコネクタがあります。



前面には、SDカードスロット、電源LED、リセットスイッチがあります。



側面にはS88-N コネクタがあります（要改造）。



S88改造方法の紹介 [https://desktopstation.net/wiki/doku.php/dsair2\\_s88](https://desktopstation.net/wiki/doku.php/dsair2_s88)

## 3.2. Wi-Fiの接続

ここではHuawei製のAndroid搭載スマートフォン・タブレットで説明します。スマートフォンやタブレット、PCのWi-Fi接続設定画面は各社によって異なります。お手持ちの機器のWi-Fi接続は、機器の説明書を参

考にして操作下さい。設定から「ネットワークの設定」等からWi-Fiを選択し、SSIDの中にある\*\*FlashAir\_(任意の英数字列)\*\*を探します。



見つかったらタップし、パスワードを“**12345678**”と入力し、接続します。問題が無ければ、Wi-Fiの接続が完了します。以後は、Chromeを使用して、\*\*<http://flashair/>と入力すれば、操作画面が表示されます。

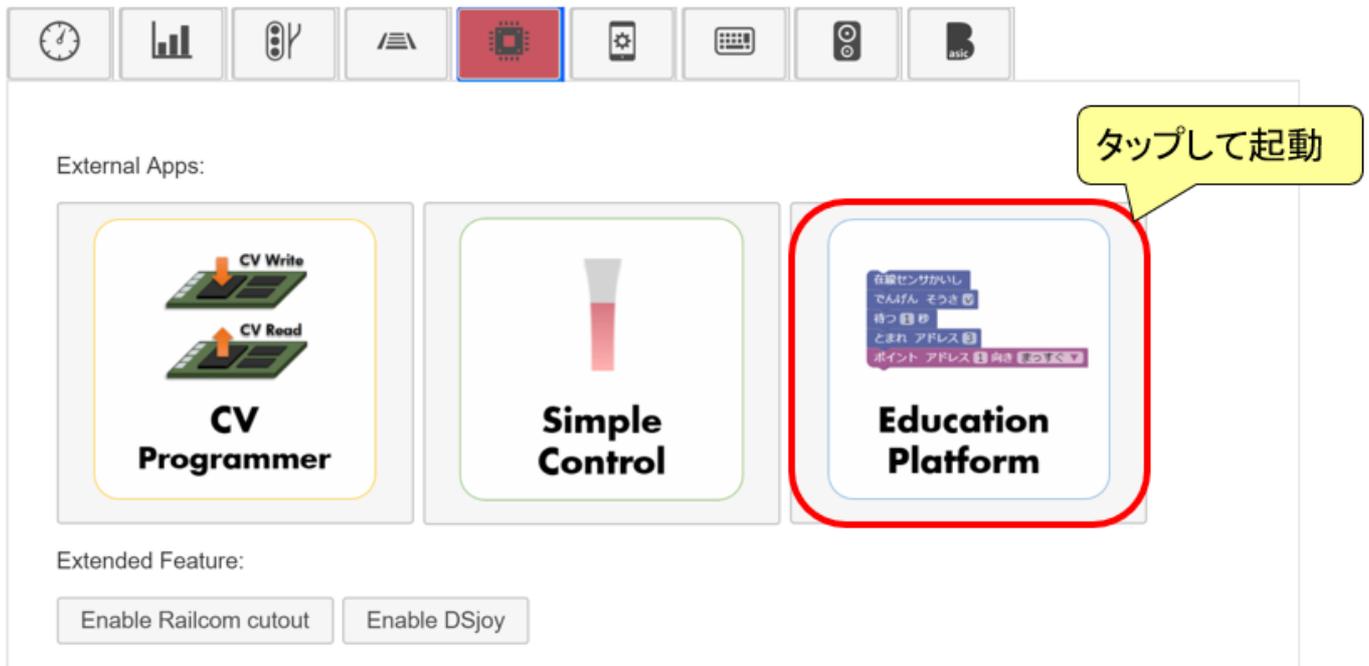
### 3.3. ソフトの動かし方

今回使用する自動運転プログラムソフトであるEducationPlatformは、DSair2のWebアプリの操作画面からICのアイコンのタブを選択して開くことができます。また、ブラウザから以下のURLを打ち込んで直接表示することもできます。

Education PlatformのアプリURL [http://flashair/SD\\_WLAN/block/index.html](http://flashair/SD_WLAN/block/index.html)

Education Platformは以下のサイトでも紹介しています。

Education Platformの説明ページ <https://desktopstation.net/wiki/doku.php/educationplatform>



### 3.4. EducationPlatformの画面

EducationPlatformを開くと、以下のような画面が表示されます。



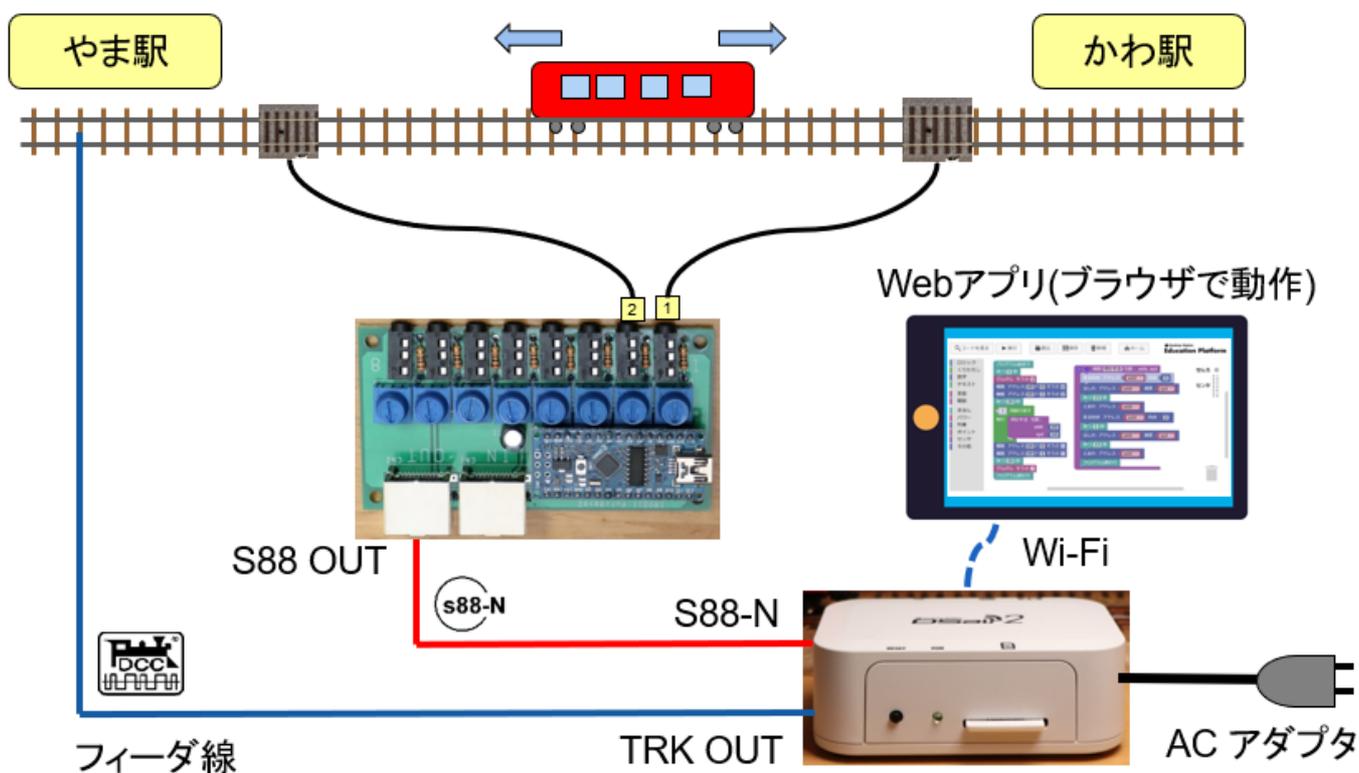
場所の名前	説明
広場	ブロックを置いてプログラミングしていく場所
引き出し	ブロックがしまっている場所
じょうたい	線路やセンサの状態を示すエリア
メニュー	ブロックなどを保存したりする機能のエリア

場所の名前	説明
-------	----

もどる	DSair2のメインアプリに戻ります
ゴミ箱	使わないブロックを捨てる場所
ズーム	広場の大きさを小さくしたり大きくしたりします。

### 3.5. 全体配線図

センサを含め、全体を配線した図を以下に示します。かわ駅側のセンサを1番(S88アドレス1)、やま駅側のセンサを2番(S88アドレス2)とここでは決めました。センサの数を増やす場合には、自分でどの番号に割り当てるか考えて、後述するイベントスクリプトのプログラムの中で対応付けを行って下さい。

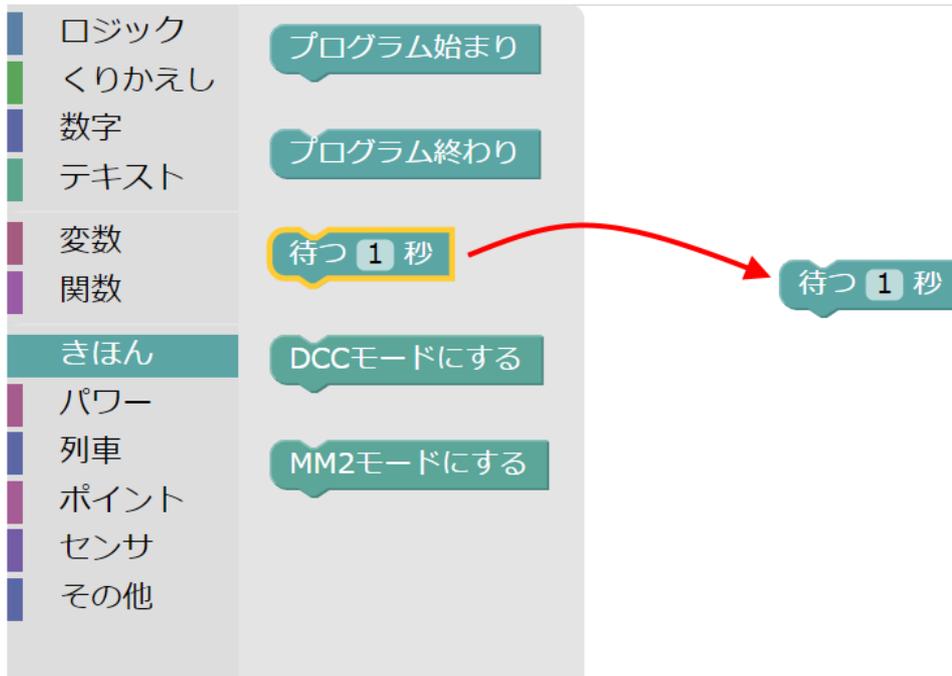


## 4. Education Platformの使い方

### 4.1 ブロックの操作の仕方

#### 4.1.1 ブロックをもってくる

ブロックは、引き出しから引っ張り出して、ひろばに置くことができます。



#### 4.1.2 ブロックをくっつける

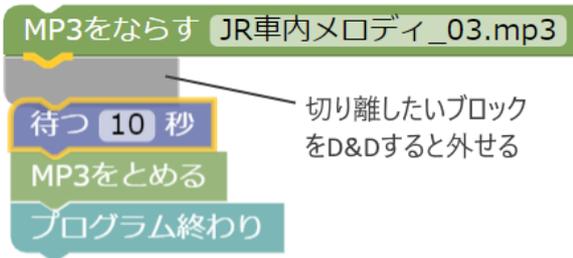
ブロックは上下にくっつけることができます。



ただし、横にくっつけるタイプのブロックもあります。横にくっつけるブロックは、じょうほうをブロックに教えるために使用します。

#### 4.1.3 ブロックを分ける

繋がったブロックを分けたい場合は、分けたいブロックをD&Dして引っ張ると外れます。



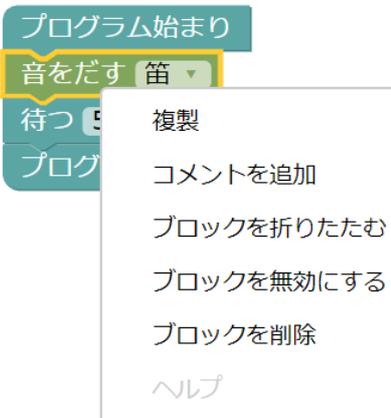
#### 4.1.4 ゴミ箱

右端のゴミ箱にブロックを入れると、ゴミ箱のふたが空いて、ブロックが消えてしまいます。要らないブロックは、ゴミ箱に入れましょう。



#### 4.1.5 右クリックメニュー

右クリックメニューを使うと、複製（同じブロックを作る）、指定のブロックを削除などができます。コピーアンドペーストも可能です。Ctrl+CとCtrl+Vがそのまま使用できます。



## 4.2 保存の仕方、読み出しの仕方

作ったブロック図は、XMLファイルとしてダウンロードして保管するか、ブラウザのメモリに保存できます。保存したXMLファイルは、読み出してまた表示できます。ブラウザのメモリに保存されたデータも読み出しできます。



保存場所	説明
ファイル	XMLファイル形式でダウンロードしたり、アップロードして使用できます。
メモリ	ブラウザが保持するメモリに保存します。リロードしても保持されます。

## 4.3 待ち時間を考えよう

Etcの「待つ」というブロックを使うと、指定した秒数、動きを止めることができます。車両を走り始めさせた場合、待つブロックを置かないと、次の動作が始まってしまい、うまく制御できなくなる場合があります。



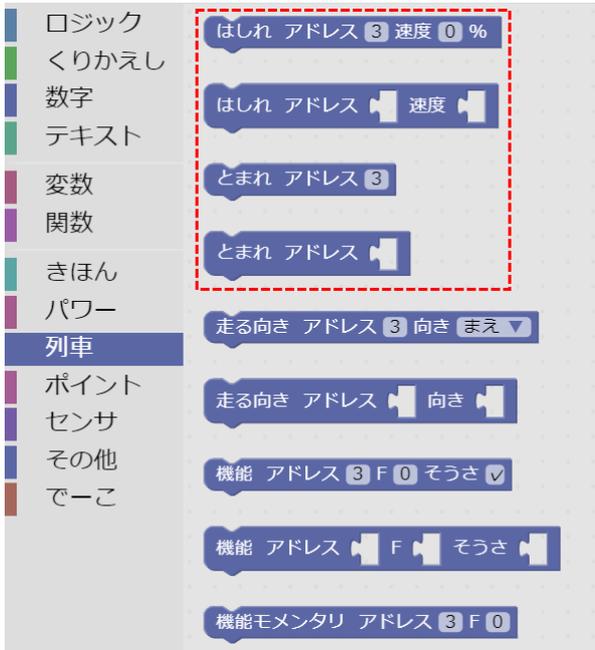
## 4.4 線路にDCC電源を流す

最初は、安全のため線路に電流を流しません。「でんげんそうさ」ブロックを使って、線路に電源を流す操作をしてください。



## 4.5 車両を動かそう

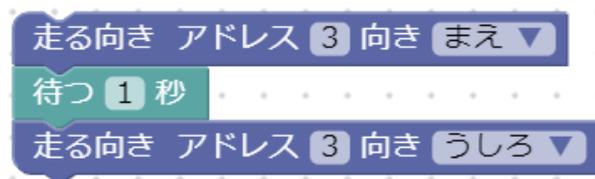
車両を走らせて、止めるブロックは以下の通りです。



はしれブロックで、速度を0にすると、とまれと同じ意味になります。また、速度は100が最高速度(100%)です。0が停止です。

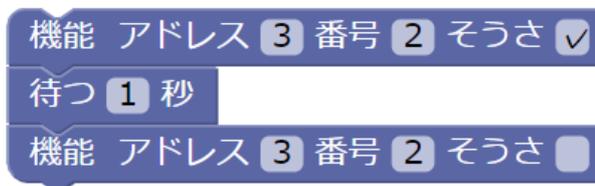
## 4.6 車両の向きを変えて前後に動かそう

車両の向きは、1=正方向(FWD)、2=逆方向(REV)で変更できます。



## 4.7 ファンクションを使ってみよう

車両のファンクションを操作します。



**ファンクションの番号**    **割り付けられている機能**

F0	前照灯
F1	サウンド機能のON/OFF
F2	警笛
F3以降	デコーダの説明書を参照ください

## 4.8 ポイントを動かしてみよう

ポイントを動かす事もできます。ポイントのアドレスは、車両のアドレスと種類が違いますので、同じ番号が来ても問題ありません。

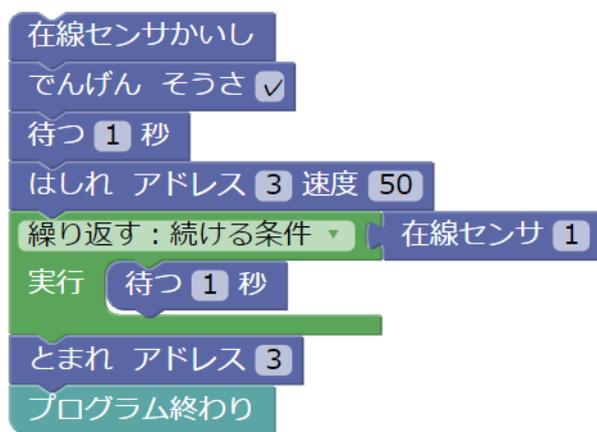
ポイントアドレスは、1～2044まで使用できます。向きは、まっすぐ(緑色)とぶんき(赤色)の2つです。

ポイント アドレス 1 向き まっすぐ ▾

## 4.9 センサと連動してみよう

在線検出センサを使う場合、必ず最初に「在線センサかいし」のブロックを置きます。このブロックを置くことで、センサが動くようになります。通常は動かないようになっています。

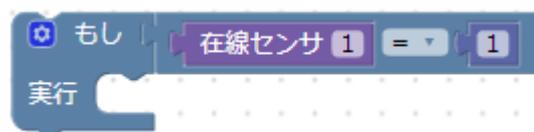
「在線センサ」のブロックを使う事で、センサが反応している(=列車が線路上にいる)と判断して、様々な動きが実現できます。在線センサは、16個まで使用できます。



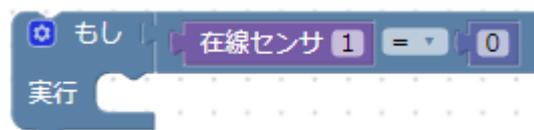
センサは、画面右側に16個の○で表現されており、赤丸のときは非在線、緑丸のときは在線という扱いになります。ただし、センサの配線の仕方や設定によって意味合いが変わる場合もあります。

センサーの判定の仕方は、列車がいるときは1、いないときは0としています。

たとえば、「もし」ブロックに使う場合、「センサ1に列車がいる」ときは



となります。「センサ1に列車がない」ときは



となります。

## 4.10 くりかえしブロックの注意

くりかえしブロックの中には、必ず、「待つ」ブロックがあるようにします。でないと、処理が動き続けてパングしてしまい、ブラウザがフリーズしてしまうことがあります。



## 4.11 せんろとセンサの動作チェック

画面右側には、せんろとセンサの○が表示されています。線路に電流を流しているときは、せんろに赤丸が表示されます。電流が流れていないときは緑丸になります。

センサにおいては、1～16の16カ所のS88デコーダのセンサを表示することができます。1台で8個のセンサを出力するS88デコーダの場合は2台分、1台で16個のセンサを出力するS88デコーダの場合は1台分となります。赤丸は非在線、緑丸は在線となります。



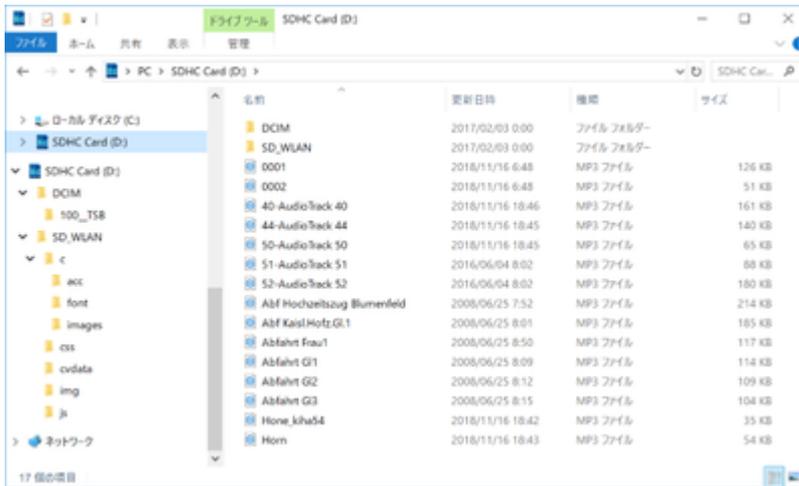
## 4.12 MP3を再生しよう

DSair2に挿入するFlashAirは、16GB～64GBの広大なフラッシュメモリを内蔵しており、ここにお好きなMP3ファイルを置くと、スマホのブラウザ上から簡単に再生することができます。しかも、スマホ上にファイルを置く必要はありません。再生命令を出した瞬間に自動的にFlashAirからスマホにダウンロードされて再生されます。

※スマホ上のファイルは再生できません。

### MP3ファイルの置き方

DSair2に使用しているFlashAirカードを、PCに差し込んでください。再生したいMP3ファイルをSDカードのルートフォルダ以下（フォルダの中においても可）に置いてください。



サポートの都合や説明をシンプルにするためにMP3ファイル以外はサポートしません。WAVやOGG、その他ファイルを置いても拡張子フィルタを使っているため表示されずは無視されます。また、ファイルサイズはなるべく小さくしてください。MP3ファイルサイズが10MB以上になると再生までにダウンロードする時間が掛かってしまいます。

## 使い方

「MP3をならす」ブロックを使って再生します。ファイル名は、FlashAirのSDカードのルート(一番上のドライブのフォルダ)にある場合は、ファイル名を書くだけで再生できます。大文字小文字は全て正確に入れて下さい。フォルダの区切りはスラッシュ「/」を使います。

MP3をならす test.mp3

停止させたい場合は、「音を止める」ブロックを使ってください。

音を止める

## 応用法

あくまでも一例ですが、Webサウンド機能を使って、より模型を楽しく遊ぶことができます。

- 発車メロディCDを購入し、MP3ファイルにしてSDカードにコピーすると、好きなタイミングで再生できます！より臨場感のある運転のために！
- バッテリー内蔵タイプのBluetoothスピーカーを使うと好きなところから音を出せます。
- ファイル数やファイルサイズに制限はありません。ファイルサイズが大きいと再生までに時間が掛かる程度です。

## 5. 自動運転プログラミングの準備

---

### 5.1. プログラミング機能の紹介

Education Platformアプリは、視覚的(ビジュアル)なプログラミング機能を提供します。

S88のセンサが動いたタイミング(=イベント)に応じて、自由に車両やポイントを動かす手順を記載して、自動運転を実現します。

基本的な書き方は、上から順に、車両やポイントにさせたい動きをブロックで表現していきます。たとえば、スピードをどれくらいにする、ファンクションを操作する、進行方向を変える、ポイントの方向を切り替える・・・などです。ブロックは、左側の引き出しから取り出すことができます。

ブロックの長さには特に制約はありません、非常に長いブロックを作っても構いませんが、ミスなどが発生しやすくなるので、「関数ブロック」などを活用して、なるべくブロックの塊を複数に分けるようにすることがコツとなります。

DCCでは複数の列車を同時に動かすことや、ポイントの操作ができますから、色々な動きを機械にやらせることができます。逆に言うと、いろんなことを機械に代わりにやらせることから、間違えて動きを書いてしまうと取り返しのつかない動きをしたりします。たとえば、車両のアドレスを逆にしてしまったとか、速度が早すぎた、など、いろいろなミスによって発生するトラブルが想定されます。

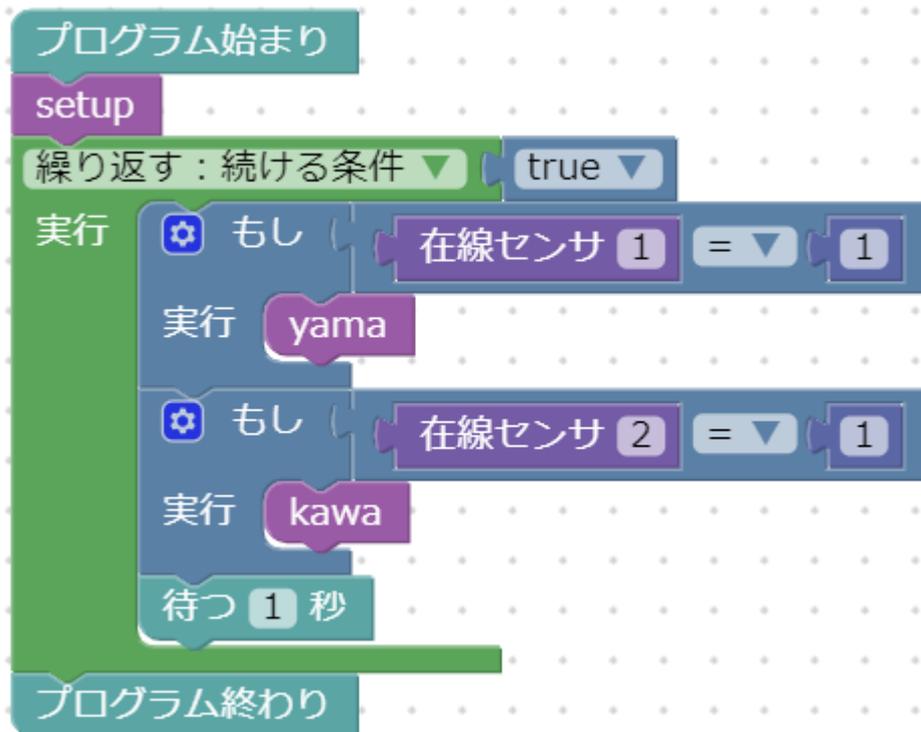
トラブルは絶対に起きます。なので、少しずつ慣れていって、トラブルがあってもすぐに対処できるような技術を少しずつ付けていくことで、自動運転のスキルがアップしていきます。

### 5.2. 自動運転プログラムを書いてみる

既に今までの説明の中で、線路にセンサを2カ所指定したと思います。S88 Detector 8chのポート1と2にセンサを繋げて下さい。このポート1,2が、そのままセンサアドレス1と2となります。なお、かわ駅側のセンサが1、やま駅側のセンサが2とします。センサの位置を逆にすると、そのままでは動きがおかしくなるので注意して下さい。

#### 5.2.1. センサを監視するブロック

かわ駅→やま駅の方向がFWD(順方向)として定義し、イベントスクリプトを書きました。



### 5.2.2. 準備する関数ブロック

この関数ブロックは、線路の電源オンと、在線センサをスタートさせる処理が入ります。後述するファンクションでサウンドやヘッドライト・テールライトを付ける場合にも、ここでブロックを足す方が良いでしょう。

ここでは、実際に車両を走らせています。



### 5.2.2. やま駅(センサアドレスは1)で動くブロック関数



### 5.2.3. かわ駅(センサアドレスは2)で動くブロック関数



### 5.2.3. 試運転

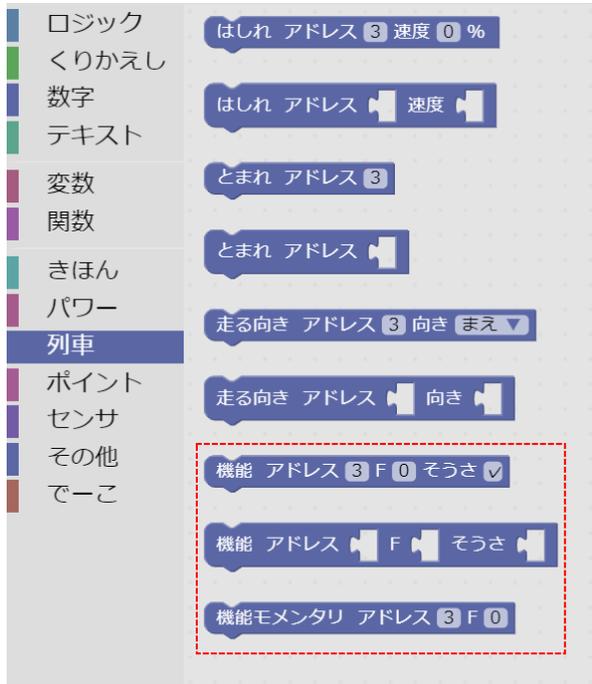
まずは、センサが反応するか、チェックします。指で押しても良いですし、実際に車両を走らせても良いでしょう。

センサが反応していたら、動かすボタンを押して、実際に試運転してみましょう。

緊急事態が起きたときは、止めるボタンを押すと停止します。

## 5.3. 応用編・ファンクションを組み合わせる

実際に往復運転ができましたでしょうか？ これだけでも楽しいと思いますが、DCCならではの機能である、「ファンクション」を使って、車両から音を出したりギミックを使って自動運転にもっと広がりを持たせましょう。



ここでは、[オープンサウンドデータ](#)を使ったキハ40(トラムウェイ製HO,LokSoundV4を搭載加工済,加工説明は[こちら](#)。

### 5.3.1. ヘッドライト・テールライトを付ける

ヘッドライト・テールライトはF0です。F0をONにすると、ヘッドライト・テールライトが点灯ようになります。なお、車両によってはテールライトが搭載されていないもの（ダミーになっているもの）があります。

機能ブロックを使用します。ここではアドレス3に対して操作しています。



### 5.3.2. 室内灯を付ける

ヘッドライト・テールライトと同じく、オープンサウンドデータのキハ40ではF3が室内灯になっているので、ファンクション操作を行います。F3=ONとします。



### 5.3.3. ファンクションF2を操作してサウンドを鳴らす

走行音を鳴らしたり、警笛を鳴らす、駅でアナウンスや笛の合図を行います。

警笛はF2ですので、以下のように操作すると、警笛が鳴ります。

機能モメンタリ アドレス 3 F 2

この"機能モメンタリブロック"は、内部でファンクションをOFF→ON→OFFの操作を行うことで実現しています。手動で行うと、3つのブロックを置かなければならないため、簡単に動かす事ができます。

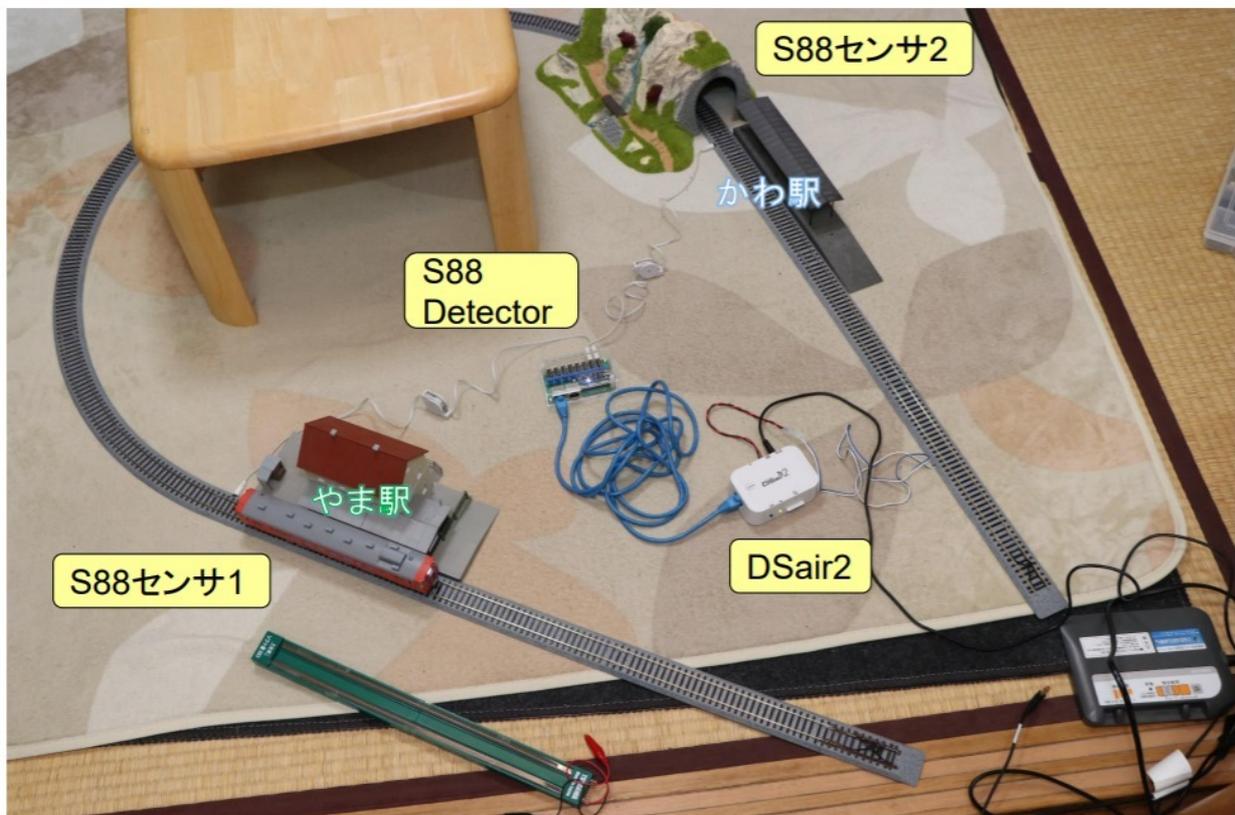
## 5.4. まとめ

上記をまとめたイベントスクリプトを、以下に掲載しております。

### 5.4.1. 全体の様子

#### 実際の配線

#### Desktop Station



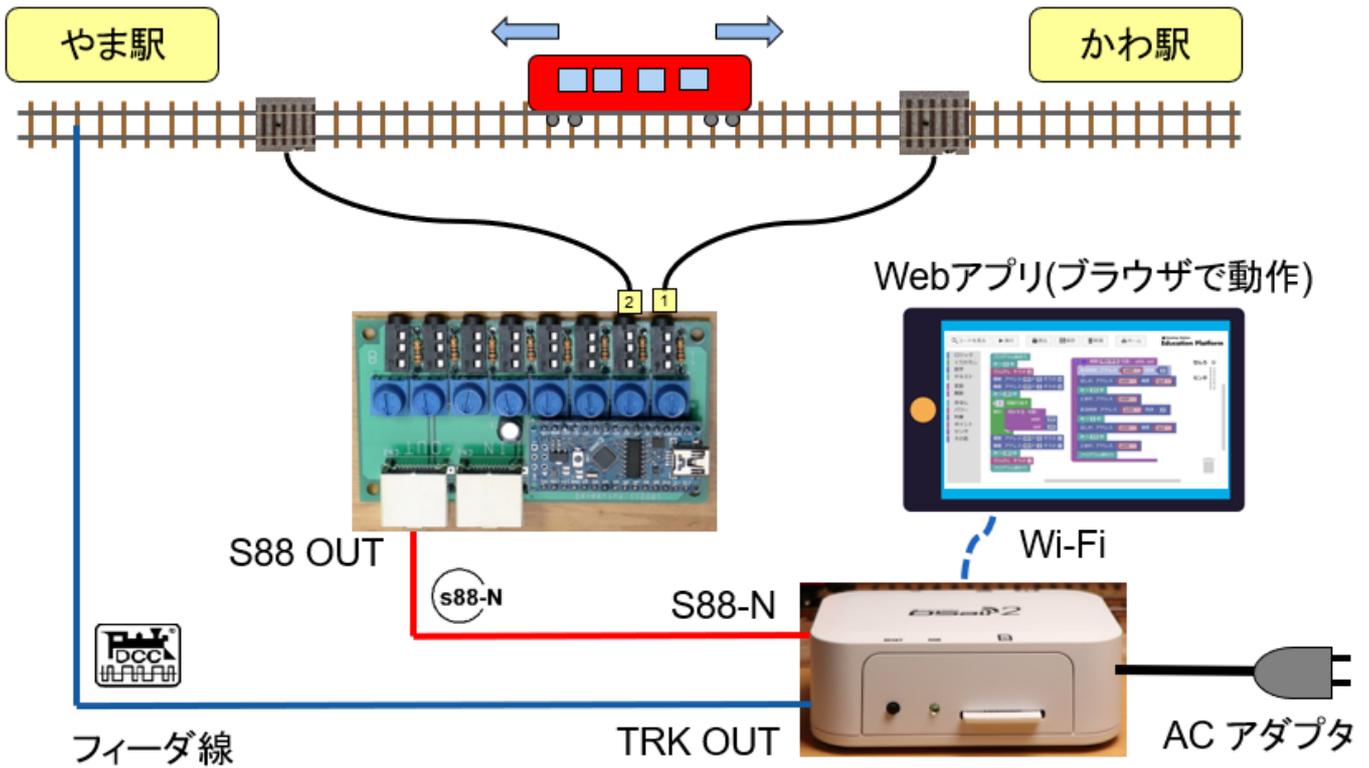
2

### 5.4.2. 写真とビデオ

Youtubeに自動運転動画をアップロードしています。

DCCサウンド車両をビジュアルプログラミングで自動運転 <https://www.youtube.com/watch?v=73IMQIAzxtY>

### 5.4.3. 機器の配線



### 5.4.4. サンプルスクリプト

```

プログラム始まり
setup
繰り返す: 続ける条件 true
実行
  もし 在線センサ 1 = 1
  実行 yama
  もし 在線センサ 2 = 1
  実行 kawa
  待つ 1 秒
  うーん と言う
  だーこちゃんが帰る
プログラム終わり

関数 yama
やま駅に着いたね と言う
とまれ アドレス 40
待つ 10 秒
走る向き アドレス 40 向き まえ
機能モメンタリ アドレス 40 F 16
待つ 23 秒
はしれ アドレス 40 速度 15 %
待つ 35 秒

関数 kawa
かわ駅に着いたね と言う
とまれ アドレス 40
待つ 10 秒
走る向き アドレス 40 向き うしろ
機能モメンタリ アドレス 40 F 16
待つ 23 秒
はしれ アドレス 40 速度 15 %
待つ 35 秒

関数 setup
音をだす きてき
でんげん オン
在線センサかいし
とまれ アドレス 40
機能 アドレス 40 F 0 そうさ
機能 アドレス 40 F 1 そうさ
機能 アドレス 40 F 3 そうさ
おはよう! と言う
待つ 20 秒
はしれ アドレス 40 速度 15 %
待つ 1 秒
    
```

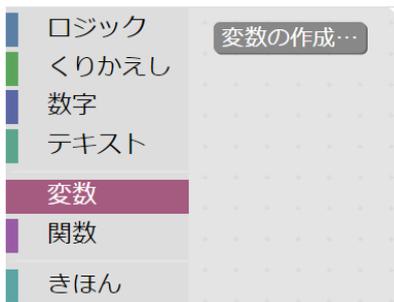
## 5.5 変数と関数を使う

変数と関数を使うと、より表現の幅が広がります。

項目	解説
変数	数字や文字を自由に入れられる箱のようなもの。名前を付けて、使い回すことができる。
関数	命令をまとめたもの。名前を付けて、使い回すことができる。

### 5.5.1 変数の使い方

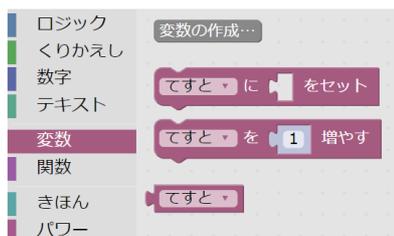
変数の作り方は簡単です。引き出しの「変数」の中に、変数の作成ボタンを押します。



以下のように変数の名前を付ける画面が出ます。日本語、英数字が使用できます。ただし、名前の最初に数字は付けなくて下さい。



変数を作成すると、引き出しに変数进行操作するブロックが3つ増えます。



たとえば、変数に10を掛ける処理を作る場合は、以下のように「数字」の引き出しにあるブロックを組み合わせることで実現できます。



### 5.5.2 関数の使い方

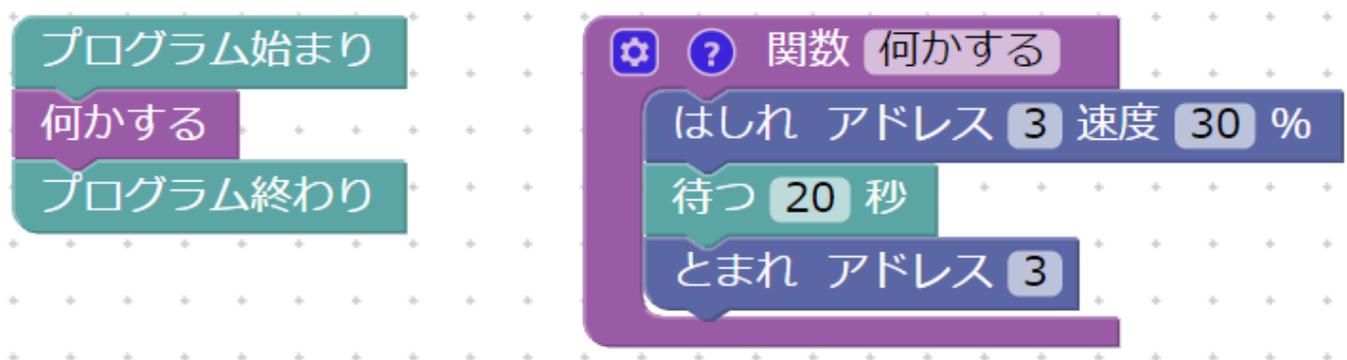
関数の引き出しから、関数のブロックを取り出します。



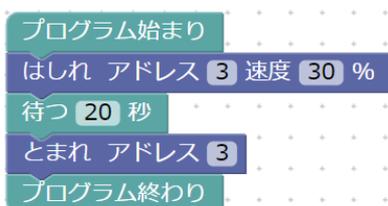
「何かする」ブロックが引き出しに増えました。



たとえば、以下のようにブロックを置くと、関数「何かをする」が動きます。



実は、以下の書き方と同じ事です。「何かをする」という関数に中身を置換えると、同じ事を何度も書かずに、「何かをする」ブロックを持ってくるだけで同じ事ができます。



以上の関数の機能を使い事で、処理をシンプルに分かりやすくまとめることができます。よく使う処理は、関数にして使い回してしまうことが、自動運転プログラムを効率よく作っていく上では重要になります。

## 6. ソフトウェアのアップデート方法

### 6.1 アップデート手順の全体

DSair2は大きく2つのソフトウェアで構成されています。ハードウェアを直接制御する本体ファームウェアと、Webサーバーやブラウザ用のWebアプリ、Wi-Fi機能を司るFlashAir(Wi-Fi SDカード)の2つについて、ご自分で設定する必要があります。

アップデート作業は、2つの作業を実施して頂くことになります。

ソフトの種類	役割
本体ファームウェア	DCC,Marklin,PWMの制御,管理,安全機能
FlashAir(Wi-Fi SDカード) Webアプリ	Wi-Fi機能,Webアプリの配信サーバー,HTMLや画像データの管理

### 6.2 本体ファームウェアの設定・更新

まず、CH340G USBドライバを普段使用するWindows PCにインストールして下さい。なお、Windows10以降はドライバなしで認識します。

CH340G USBドライバ [https://desktopstation.net/wiki/doku.php/install\\_driver](https://desktopstation.net/wiki/doku.php/install_driver)

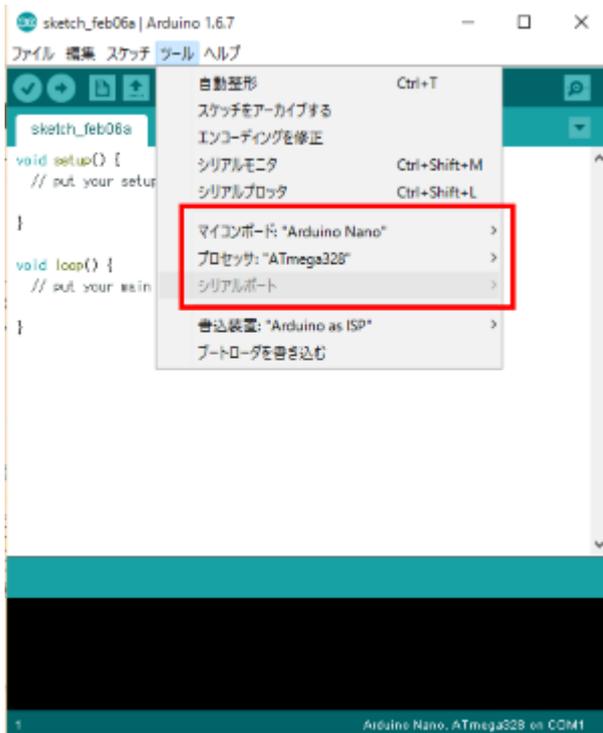
Arduino IDE(無料)も予めダウンロードし、インストールして下さい。バージョンは、1.8.5以降を推奨します。

Arduino IDE <https://www.arduino.cc/en/main/software>

ファームウェア・Webアプリから、最新のファームウェア(スケッチ)を解凍し、分かりやすいところに格納します。

ファームウェア・Webアプリ [https://desktopstation.net/wiki/doku.php/dsair2\\_firmware](https://desktopstation.net/wiki/doku.php/dsair2_firmware)

Arduino IDEを起動し、DSairFirmwareスケッチ(DSairFirmware.ino)を開きます。



ここで、DSair2をUSBケーブルを使ってPCに繋げて下さい。ACアダプタは接続する必要はありません。

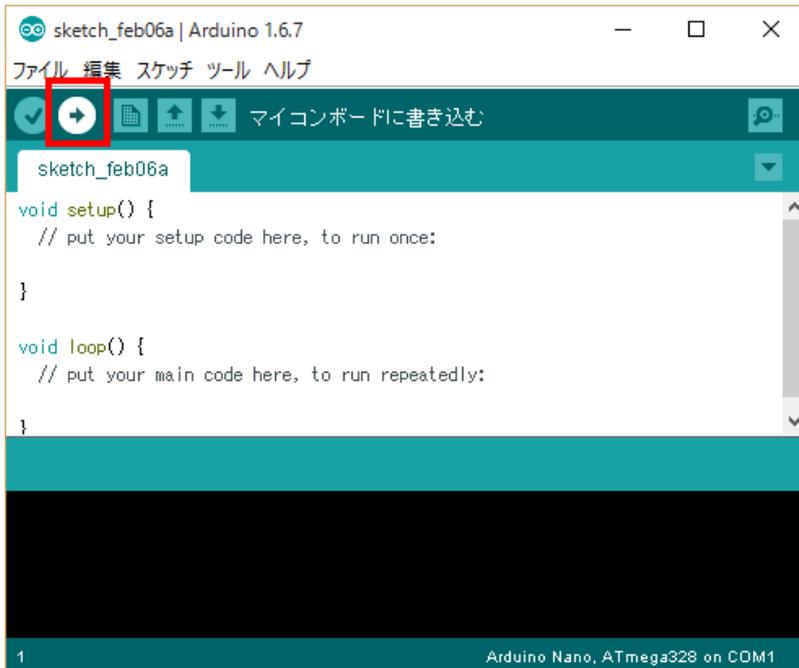
正常にPCで認識し、COMポートが割り付けられることを確認して下さい。

この後、Arduino IDEでボード設定を行います。ボードの種類はArduino nanoとし、プロセッサはATMEGA328Pとしますが注意点があります。

Arduino IDE 1.8.7以降では、ボードをArduino nanoにした後、プロセッサをATMEGA328P(Old Bootloader)に設定して下さい。デフォルトのままですと、書き込みが出来ません。ただし、1.8.5までのArduino IDEではデフォルトのまま書き込みが行えます。

通信ポートは、DSair2が認識したポートを指定して下さい。ほとんどのケースでは、1つしか表示されていないはずですが、この場合は、認識された1つを選択すれば問題ありません。

アップロードボタン（メニューのアップロードでも可）で書き込みを行います。



## 6.3 Webアプリのアップデート方法

SDカード内の隠しフォルダSD\_WLAN内のファイルを、最新版のファームウェアのSD\_WLANフォルダからコピーすることでアップデートが行えます。Windows向けに、隠しフォルダなどを意識せずにアップデートをワンクリックで行うアップデートツールも提供しています。

FlashAirのW-04をご自分で購入された場合は、ファームウェアのバージョンに注意してください。一部の流通品は、初期バージョンの4.00.00であるため、正常に動作しません。アップデートツールが東芝からWebにて提供されていますので、先にアップデート作業を行ってください。バージョンが4.00.01以降の場合、ファームウェアアップデートは不要です。

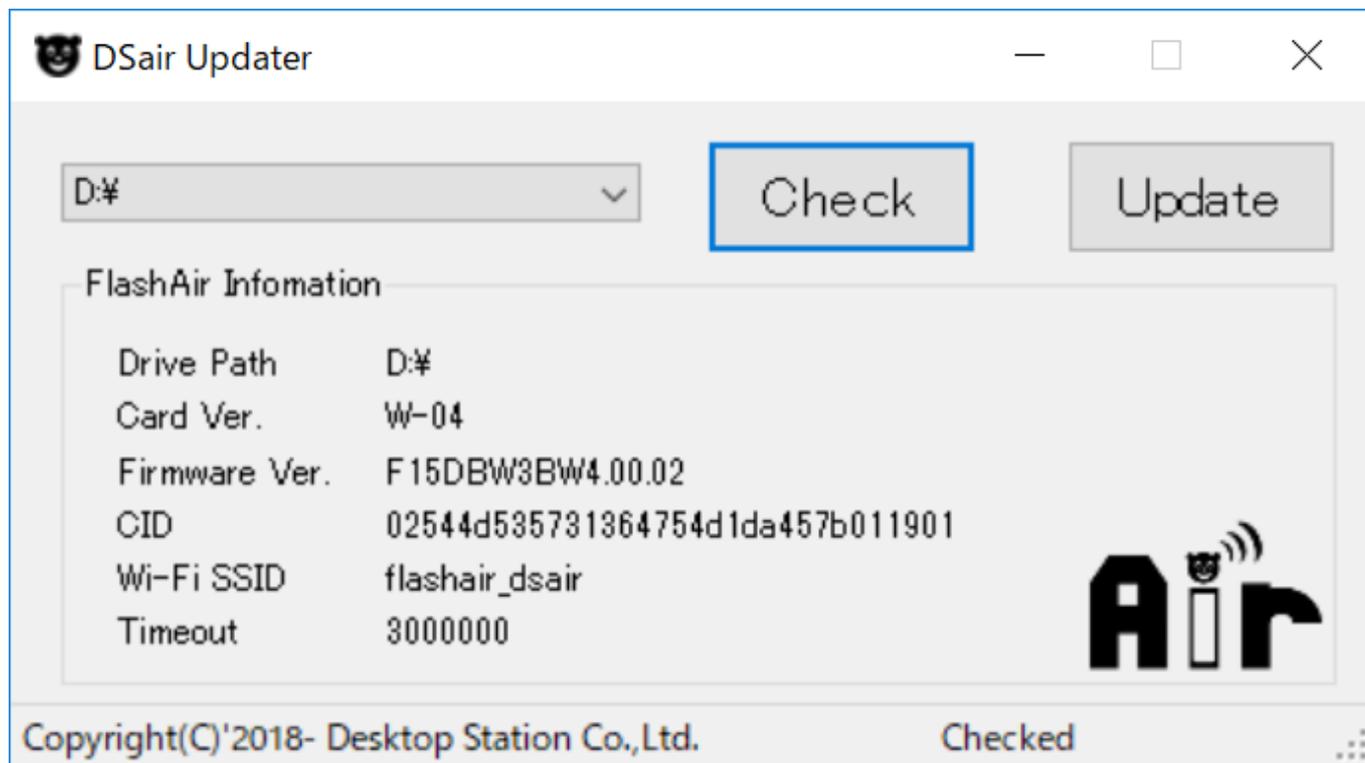
富士通製ノートPCの内蔵SDカードスロットや、外付けのSDカードリーダーで、W-04のファームウェアアップデートに失敗する報告があります。原因は、自動的に省電力モードに移行してしまうためです。外部電源を供給可能なUSBハブを使用すると、問題は起きないことを確認していますが、全てのハブやSDカードリーダーの組み合わせで検証しておりません。ファームウェアアップデートには十分、ご注意ください。なお、通常のファイルの読み書きにおいては問題は起きません。なお、もし失敗してカードが使いなくなった場合には、東芝メモリにサポートを依頼してください。なお、並行輸入版ではサポートを受けられないため、故障したら廃棄するしかありません。国内の正規販売店で購入して、アップデートを行ってください。

### 6.3.1 ツールを使う方法

Windows向けの専用ツールが提供されています。以下のページに記載の手順で、操作すると、自動でDSair2用のWebアプリが導入されます。なお、W-04専用で、4.00.01以降のファームウェアでなければ動作しません。4.00.00の場合は、東芝メモリ(現キオクシア社)の提供するファームウェアアップデートツールで4.00.01以降(どれでもかまいません) にアップデートしてください。

FlashAir内Webアプリのアップデートツール

[https://desktopstation.net/wiki/doku.php/flashair\\_update\\_tool](https://desktopstation.net/wiki/doku.php/flashair_update_tool)



### 6.3.2 手動でアップデートされる場合

下記で頒布しているFlashAirアプリをダウンロードします。

ファームウェア・Webアプリ [https://desktopstation.net/wiki/doku.php/dsair2\\_firmware](https://desktopstation.net/wiki/doku.php/dsair2_firmware)

パソコンにFlashAirをSDカードスロットに差し込んだら、SD\_WLANという隠しフォルダにアクセスし、以下のファイルを上書き保存します。たとえば、EドライブにFlashAirのドライブが出たら、パスを打ち込む画面にE:/SD\_WLAN/と直打ちすると移動できます。隠しフォルダ・ファイルを表示するように設定変更ことも検討ください。

CONFIGという隠しテキストファイルがあるので、テキストエディタで開きます。SD\_WLANフォルダ同様、普通には見えないので、隠しファイルを表示するように設定変更します。または、E:¥SD\_WLAN¥CONFIG と入力してENTERを押すと、メモ帳で編集できるようになります。

以下のように行を追加して「APPAUTOTIME=3000000」を足します。この追加の命令は、省電力モードに移行するまでの時間を通常よりも長くする設定です。もし忘れると、操作中に自動で無線通信が停止し、操作ができなくなる場合があります。

```
[Vendor]
CIPATH=/DCIM/100__TSB/FA000001.JPG
APPMODE=4
APPNETWORKKEY=*****
VERSION=F15DBW3BW4.00.00
CID=*****
PRODUCT=FlashAir
VENDOR=TOSHIBA
APPAUTOTIME=3000000
MASTERCODE=*****
LOCK=1
```

---

以上の作業で、設定は完了です。SD\_WLANフォルダに、ダウンロードしたZIPファイル内のSD\_WLAN以下ファイル・フォルダを全てコピーした後、取り外して、DSair2のSDカードスロットに差し込んでください。

## 7. FAQ・質問回答

---

### 7.1 作成できるプログラムのサイズに制限はありますか？

プログラムサイズの制限はありません。メモリとCPU性能が許す限り、使用できます。

### 7.2 DSbasicのプログラムやWindows用のDesktopStationSoftwareのイベントスクリプトをEducationPlatformで使用できるようにプログラムを変換できませんか？

現状はできません。技術的には可能です。

### 7.3 Education Platformを使って商業・ビジネス利用できますか？

Education Platformを使った講義、講演、デモンストレーション、博物館のジオラマへの組み込みなどは、非営利・営利にかかわらず自由に行えます。ただし、Education Platform自体を販売する行為はおやめ下さい。

DesktopStationでは、鉄道模型ジオラマ向けの現地据え付け・運用・メンテナンスサービスは生業として行っておりません。しかしながら、当社の製品や技術に精通した提携業者を紹介させていただきます。

### 7.4 スマホやタブレットでも使用できますか？

Education Platformはブラウザで動作しますので、PCだけでなくスマホやタブレットの画面操作だけでも、プログラミングできます。ただし、効率が良くないため、USBキーボードやマウスを接続して操作することをお勧めします。なお作成したデータは、スマホやタブレットの中にも保存できますし、XMLファイルとしてダウンロードできますので、ファイル管理も可能です。

### 7.5 Education Platformを使用しているときは、他のスマホ等から操作できますか？

Education Platformの内部処理は、DSair2のWebアプリ標準に準拠しています。

DSair Wi-Fi Specification [https://desktopstation.net/wiki/doku.php/dsair\\_wifi\\_specification](https://desktopstation.net/wiki/doku.php/dsair_wifi_specification)

Education Platformで自動運転プログラム動作中に、他のアプリから自由に操作できます。ただし、電源OFFや車両操作は自由に行えますので、自動運転対象の車両を他のアプリから操作すると、想定しない動きになる場合があります。

### 7.6 命令が安定して鉄道模型に伝わらない

Education Platformの自動運転プログラムは、ブラウザ上のWebアプリが処理しています。よって、Wi-Fiの無線通信が安定している環境で使用することが前提となります。

特に以下のような環境では、Wi-Fi無線の混雑状況をスマホの設定画面などで確認して、必要に応じて中継器の使用や、スマホをなるべくDSair2に近づけて利用下さい。

- 展示会の会場(無線通信機器が多用されるメーカーフェア等)
- たくさんの人が集まる環境(スマホ等を持っている人が多い環境)

## 8. サポート

---

デジタル鉄道模型フォーラムをご利用下さい。デジタル鉄道模型フォーラムはDesktopStationが管理運営しており、登録・年会費無料をご利用頂けます。

デジタル鉄道模型フォーラム <https://desktopstation.net/bb/>

## 9. 最後に

---

Education Platformは超大規模なレイアウトの制御には向きませんが、ご自宅で遊ぶ程度の中小のレイアウトには問題無くご利用頂けます。ここで得られたスキルを、さらにパワーアップさせて、夢を実現して頂ければ幸いです。